



UNIVERSIDAD
POLITECNICA
DE VALENCIA



Sistema de Información para la Gestión de Ligas de Billar

BILLARGEST

PROYECTO FINAL DE CARRERA

AUTOR:
Carlos Soriano Lorente

DIRECTOR:
José María Torralba Martínez

Valencia, Mayo 2010

Contenido:

1. RESUMEN

2. DOCUMENTO I – MEMORIA

- Capítulo 1.- Introducción, objeto, antecedentes y bases.
- Capítulo 2.- Plan de Sistemas de información y selección de alternativas.
- Capítulo 3.- Análisis del sistema de información.
- Capítulo 4.- Evaluación de riesgos.
- Capítulo 5.- Medición, estimación, presupuesto, oferta y programación temporal.
- Capítulo 6.- Diseño de sistema de información.
- Capítulo 7.- Plan de construcción y pruebas.
- Capítulo 8.- Plan de puesta en servicio.
- Capítulo 9.- Conclusiones.
- Bibliografía.

3. ANEXOS A LA MEMORIA

- ANEXOS 1 – Estudio Preliminar del software de gestión actual.
- ANEXOS 2 - Planificación de Sistemas de Información (Proceso PSI).
- ANEXOS 3 - Estudio de Viabilidad del Sistema (Proceso EVS).
- ANEXOS 4 - Análisis del Sistema de Información (Proceso ASI).
- ANEXOS 5 - Diseño del Sistema de Información (Proceso DSI).
- ANEXOS 6 - Construcción del Sistema de Información (Proceso CSI).
- ANEXOS 7 - Implantación y Aceptación del Sistema (Proceso IAS).
- ANEXOS 8 - Mantenimiento del Sistema de Información (Proceso MSI).
- ANEXOS 9 - Manual de usuario.
- ANEXOS 10 – Presupuesto.
- ANEXOS 11 - Diseño de la Base de Datos.
- ANEXOS 12 - Ejemplo de Pruebas de Interfaz.
- ANEXOS 12 – Pliego de condiciones.
- ANEXOS 13 – Evaluación de Riesgos.

Resumen. Índice:

1. Necesidad, problema, promotor, usuarios	2
2. Metodologías	3
3. Plazo de Realización	4
4. Herramientas de Realización.....	4
5. Funcionalidades de la aplicación.....	4
6. Pantalla principal.....	6
7. Precio final.....	6

1. Necesidad, problema, promotor, usuarios

La Federación de Billar de la Comunidad Valenciana (en adelante FBCV) contaba con numerosos problemas debidos a la antigüedad del software con el que se gestionaban los torneos, clubes y socios de la comunidad. Estos problemas iban desde la imposibilidad de actualizar los componentes hardware en los que funcionaba la aplicación, hasta limitaciones en la mayor parte de datos de entrada, y no solo eso, la obligada adaptación a la LOPD¹ al estar la federación en su ámbito de aplicación hacia que fuera necesario adoptar un paquete software que cumpliera las directrices de dicha ley.

Para conseguir la financiación necesaria, que no solo se limitaba al software, sino también a la actualización de hardware, se utilizó una subvención del Consell Valencià de L'Esport. La primera estimación de presupuesto que hizo la FCBV fue de siete mil euros (7.000€) por la actualización integra.

Una vez se aprobó el acta para la ampliación y actualización de los Sistemas de Información, el personal administrativo fue el encargado de contactar con empresas y particulares dispuestos a llevar a cabo el proyecto en cuestión. Se pensó en un primer momento en la empresa que había desarrollado la primera versión, pero tras quince años, la empresa había desaparecido. La siguiente opción fue la de intentar adquirir un paquete ya desarrollado, pero en el mercado no existía ninguna herramienta específica que cubriera todas las necesidades. Así pues, tras solicitar dos presupuestos, en los que la cuantía de una primera estimación superaba ampliamente los diez mil euros (10.000€) por el paquete completo, se plantea la compra por separado del material hardware y de la herramienta de gestión, que sería adaptada de otra federación de billar autonómica.

Uno de los socios de la FBCV es copropietario de una tienda de informática en la cual se encargarían los dos equipos que formarán parte del nuevo sistema de información. Las características básicas serían:

- Procesador Intel Dual Core E5200.
- 4GB Ram PC667.
- 1TB HD.
- Impresora Laser HP® 1022 Compatible PS5 y PS6.
- SO: Microsoft® Windows 7 Home, Compatible Framework 3.5.

El mismo socio es el que ofrece, antes de que las gestiones con la federación de otra comunidad avancen, la posibilidad de hacer el trabajo al que finalmente será el futuro desarrollador del proyecto, D. Carlos Soriano Lorente, y tras una primera estimación de costes, la cuantía destinada al paquete informático es aceptada por las dos partes.

De esta manera nace el proyecto BILLARGEST (**Sistema de Información para la Gestión de Ligas de Billar**) el cual permite llevar a cabo la gestión integral de los diferentes clubs, socios, torneos y ligas de Comunidad Valenciana. El principal cometido del nuevo programa es la actualización del antiguo paquete de software cubriendo los requisitos ya marcados anteriormente, y añadiendo nuevas funcionalidades y características acordes a los tiempos que corren.

¹ Ley Orgánica 15/1999, de 13 de diciembre, de Protección de Datos de Carácter Personal.

2. Metodologías

El estudio de cada proceso de desarrollo del proyecto se ha llevado a cabo con Métrica² v3. La decisión principal de usar esta metodología responde básicamente a que es la metodología estándar de planificación y desarrollo de sistemas de información de la administración central española, lo que asegura la calidad, la seguridad del paquete y sobre todo, la cantidad de recursos documentales que existen de esta metodología en español, al contrario que otras metodologías ágiles como Programación Extrema (XP) o Essential Unified Process (EssUP).

También se ha tenido en cuenta la posibilidad futura de venta a alguna federación nacional que requiera obligatoriamente el uso de esta metodología, como así lo requieren todas las organizaciones de la administración central española.

Debido a que el lenguaje usado para la programación del paquete será un lenguaje 100% orientado a objetos como es C# 3.5, se decide el uso del lenguaje de modelado UML³ 2.0. Ello conlleva el uso de diferentes diagramas que serán para el caso:

Diagramas de Estructura

- Diagrama de clases
- Diagrama de componentes
- Diagrama de paquetes

Diagramas de Comportamiento

- Diagrama de casos de uso

Las pruebas del ciclo de vida que se realizarán serán:

- Pruebas Unitarias
- Pruebas de Integración
- Pruebas Funcionales
- Pruebas de Rendimiento

La planificación del proyecto se realiza con diagramas de Grantt, y la estimación del tiempo de desarrollo se basa en la estimación por puntos de función ajustados.

² MÉTRICA V3: **Metodología de Planificación, Desarrollo y Mantenimiento de sistemas de información** (28/04/2010): <http://www.csi.map.es/csi/metrica3/>

³ UML V2.0: Unified Modeling Language (28/04/2010): <http://www.uml.org/>

3. Plazo de Realización

El proyecto se lleva a cabo de Septiembre de 2009 a Abril de 2010.

4. Herramientas de Realización

El paquete se ha realizado con el entorno de programación Microsoft® Visual Studio .NET 2008 usando el Framework 3.5 y el lenguaje de programación C#. La BB.DD. ha sido realizada con el entorno Visual Studio en formato Microsoft® Access 2007. También se ha usado la aplicación de generación de Reportes Crystal Reports® .NET y la herramienta de creación de paquetes autoinstalables InstallShield® 2010.

5. Funcionalidades de la aplicación

La aplicación contiene las siguientes funcionalidades:

Inicio de la aplicación: la aplicación contiene una pantalla de bienvenida que aparece durante el inicio y presenta datos acerca de la aplicación. Una vez mostrada la pantalla de inicio, se realizan las comprobaciones de licencias. En caso de que el usuario introduzca un código de licencia válido, este se almacena en los archivos de configuración de la aplicación y no se vuelve a solicitar la clave tras el primer arranque. En caso de no introducirlo, la pantalla no permite el acceso a la aplicación.

Gestión de temporadas: Desde esta opción se gestiona la base de datos y las temporadas de BILLARGEST. En la gestión de temporadas se incluye:

Nueva temporada: permite crear una nueva temporada para albergar nuevas competiciones. Crear una nueva temporada conlleva vaciar los listados de estadísticas del programa y las actas visibles, pero no los clubes y socios introducidos.

Abrir temporada: permite abrir o eliminar una temporada ya creada seleccionando el año.

Acta: Desde esta opción se gestionan solo las actas de la temporada abierta actualmente. En la gestión de actas se incluye:

Añadir: permite añadir una nueva acta a la Base de Datos.

Editar: permite editar un acta previamente introducida a la Base de Datos.

Borrar: permite borrar un acta previamente introducida a la Base de Datos.

Sancionar: permite sancionar algún club por cualquier tipo de irregularidad, como una alineación indebida.

Club: Desde esta opción se gestionan los clubes adscritos a la Federación. En la gestión de clubes se incluye:

Añadir: permite añadir un nuevo club a la Base de Datos.

Editar: permite editar un club previamente introducido en la Base de Datos.

Borrar: permite borrar un club previamente introducido en la Base de Datos.

Socio: Desde esta opción se gestionan los socios adscritos a los clubes dados de alta. En la gestión de socios se incluye:

Añadir: permite añadir un nuevo socio a la Base de Datos.

Editar: permite editar un socio previamente introducido en la Base de Datos.

Borrar: permite borrar un socio previamente introducido en la Base de Datos.

Imprimir Etiqueta: permite imprimir un carnet de socio con un formato previamente establecido.

Precio Licencia: permite establecer el precio de la licencia para socio Junior o socio Sénior.

Listados: Desde esta opción se visualizan e imprimen los listados del programa. Dentro de listados se incluyen las opciones de:

Clasificación: permite visualizar en pantalla un listado de la clasificación provisional por puntos de una liga.

Promedios: permite visualizar en pantalla un listado de la clasificación provisional por mayor promedio de una liga.

Opciones generales:

Imprimir: si se tiene abierto un formulario de entrada de datos se puede seleccionar la opción de menú Imprimir, que presenta una selección de opciones de impresión para el origen de datos actual.

Exportar: exportar cualquier formulario a un formato de archivo de hoja de cálculo de Microsoft[®] Excel (.xls).

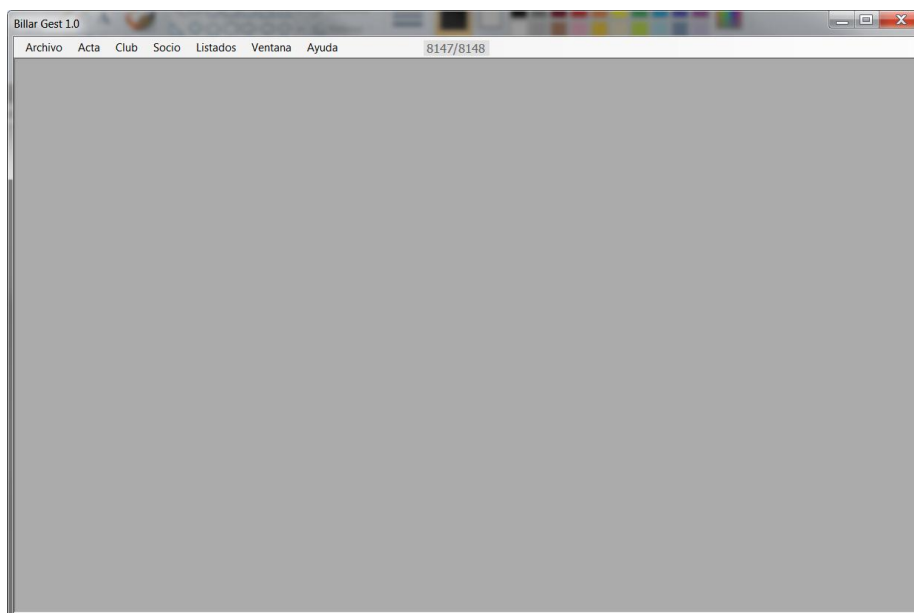
Otros: el marco de la aplicación también admite algunas características comunes de las aplicaciones para Windows. Entre ellas se incluyen:

-Un menú Ventana que le permite distribuir en cascada o en la forma que elija todas las ventanas abiertas.

-Un cuadro de diálogo Acerca de y elementos de menú del archivo de Ayuda.

6. Pantalla principal

A continuación se representa la pantalla principal de la aplicación, que contiene el acceso a las diferentes funciones de negocio. Desde de la pantalla principal se informará en todo momento de la temporada que se encuentra el usuario actualmente. Se podrá acceder a las opciones del menú haciendo **click** sobre ellas.



7. Precio final

El precio final del paquete serán tres mil novecientos noventa y cinco euros (3.995 €) IVA incluido.

Memoria. Índice:

1.	Introducción, Objeto, Antecedentes y Bases	3
1.1.	Introducción	3
1.2.	Objeto	4
1.3.	Promotor del proyecto	4
1.4.	Descripción de la situación actual	5
1.5.	Bases del proyecto	7
1.6.	Metodología general	7
1.7.	Estructura documental	8
1.8.	Plan de ampliación	9
2.	Plan de Sistemas de Información y selección de alternativas.....	10
2.1.	Situación actual	10
2.2.	Plan de Sistemas de Información	10
2.3.	Estudio y selección de alternativas	11
3.	Análisis del Sistema de Información	14
3.1.	Alcance del sistema	14
3.2.	Entorno tecnológico	14
3.3.	Estándares y normas	15
3.4.	Requisitos funcionales.....	15
3.5.	Análisis de clases	16
3.6.	Interfaces de usuario.....	17
4.	Evaluación de riesgos	18
4.1.	Riesgos identificados	18
4.2.	Planes de acción	18
4.3.	Planes de contingencia.....	19
5.	Medición, Estimación, Presupuesto, Oferta y Programación temporal	20
5.1.	Cálculo de puntos de función.....	20
5.2.	Descomposición del esfuerzo por fases y presupuesto	23
5.3.	Oferta al cliente	24
6.	Diseño del sistema de información	25
6.1.	Niveles de arquitectura	25
6.2.	Requisitos de diseño	26

6.3.	Excepciones	26
6.4.	Entorno tecnológico	26
7.	Plan de construcción y pruebas	27
7.1.	Plan de Construcción	27
7.2.	Alcance de las pruebas	27
7.3.	Plan de pruebas del sistema.....	27
8.	Plan de puesta en servicio.....	28
8.1.	Plan de formación	28
8.2.	Plan de manual de usuario.....	28
9.	Conclusiones	29
	Bibliografía	31

1. Introducción, Objeto, Antecedentes y Bases

1.1.Introducción

Una federación deportiva por su propia definición es la unión o alianza entre sociedades u organizaciones, que tiene como objeto la promoción, organización y reglamentación de uno o varios deportes en un ámbito territorial propio donde ejercerá sus actividades.

Es fácil suponer que para llevar a cabo todos los objetivos de cualquier federación deportiva, es necesario usar un conjunto de herramientas administrativas que permitan, de una manera segura, fácil y rápida, gestionar y organizar todo el volumen de información que generan dichas actividades.

Un sistema de información informatizado resuelve muchos de los problemas generados por el aumento de volumen de datos y la dispersión de información, y así lo creían también los responsables de la Federación de Billar de la Comunidad Valenciana (en adelante FBCV), que en 1996, tomaron la decisión por consenso, de implantar un sistema de información informatizado en su sede de la Calle Avellanas de Valencia. En el acta interna, se describían una serie de requisitos básicos que debía cumplir la aplicación, entre los cuales estaban:

- Una gestión completa de los socios y clubes de la Comunidad Valenciana. Los datos debían ser gestionados de una manera sencilla y rápida.
- El sistema debería poder gestionar las ligas anuales de todas las modalidades de Billar que engloba la FBCV. Asimismo debería poder gestionar todos los listados que hasta el momento se hacían a mano.
- Poder imprimir todos los datos almacenados en el programa en formato A4 y poder imprimir carnets en un formato de media cuartilla preestablecido.
- La aplicación debía funcionar en el S.O. MSDOS 6.22
- El coste aproximado de la implementación no debía superar las 800.000 pesetas incluidas peticiones hasta dos meses después de la implementación.
- El entorno de aplicación sería mono puesto y mono usuario.

El proceso de consultoría se llevó a cabo por la empresa Carbel Sistemas S.L., la cual llevó también la implementación del programa en tres meses. Asimismo, se adquirió un equipo informático basado en un procesador Intel[®] 486, el cual sigue todavía en funcionamiento, y una impresora matricial Hewlett Packard[®].

Estos requisitos, tras veinte años, han pasado a ser limitaciones de la propia aplicación actual. La FBCV se dio cuenta que su sistema de información no cumplía las exigencias que requerían los tiempos actuales. Los principales problemas que suponían una limitación de los sistemas actuales con los que trabajaban eran la nula capacidad de ampliación a nivel de software, que usaba formatos de ligas de hace quince años, junto con datos fijos que limitaban la cantidad de ligas, sus divisiones, etc., además de los problemas surgidos por el hardware, que eran todavía más costosos de solventar, como la sustitución de piezas de un equipo para el cual ya no se fabrican componentes, la dificultad de conseguir consumibles para una impresora específica y obsoleta, y la imposibilidad de sustituir ningún componente que no fuera establecido en la configuración inicial del programa.

1.2. Objeto

El proyecto BILLARGEST (**Sistema de Información para la Gestión de Ligas de Billar**) permite llevar a cabo la gestión integral de los diferentes clubs, socios, torneos y ligas de Comunidad Valenciana. El principal cometido del nuevo programa es la actualización del antiguo paquete de software cubriendo los requisitos ya marcados anteriormente, y añadiendo nuevas funcionalidades y características acordes a los tiempos que corren.

Comprende las siguientes líneas de gestión:

- El alta/baja/modificación de los clubs de la Comunidad Valenciana.
- El alta/baja/modificación de los socios, y la creación de las fichas de los jugadores, para la posterior impresión de los carnets de socios de la federación. Se debe tener en cuenta un "listado de fuerza", por el cual a la hora de asignar jugadores a los partidos es necesario que se cumpla un orden establecido.
- La posibilidad de manejar y editar los datos de todas las ligas anuales de forma rápida y sencilla, con objeto de saber en todo momento las clasificaciones en promedio de los jugadores y la clasificación en puntos de los diferentes equipos que participan en las ligas.
- Poder llevar un seguimiento semanal de partidos disputados, actas entregadas en la Federación y puntuaciones obtenidas en esas actas.
- La aplicación debe tener en cuenta las penalizaciones impuestas por los árbitros. El paquete debe controlar la asignación de árbitros a los diferentes encuentros.
- La gestión de las temporadas anuales, pudiéndose exportar/importar la información de la BB.DD. como ficheros propios del programa.
- La posibilidad de generar y exportar cualquier listado a formato Microsoft® Excel 2007 y la impresión de los mismos con vista previa y preselección de campos.

Toda la funcionalidad estará incluida en una sola aplicación. Además, todo el programa estará controlado por el personal de administración, no siendo necesario ningún tipo de control de acceso, aunque si se incluirá un control de acceso de licencia para evitar el uso indiscriminado o la posterior reventa.

Para la elaboración de los objetivos propuestos se han hecho necesarias entrevistas con el cliente así como el análisis en profundidad del paquete de gestión BILLGEST con el que actualmente trabaja la FBCV.

1.3. Promotor del proyecto

El promotor del proyecto es la *Real Federación Valenciana de Billar (FBCV)*, una federación deportiva que cuenta con más de 80 clubs y 2.000 socios adscritos. La gestión del sistema de información viene dada por un paquete software encargado en 1996, del cual se requiere una actualización por problemas de mantenimiento y limitaciones en su construcción inicial.

Debido a esos problemas, se ha estado compaginando el uso de la herramienta de gestión con la utilización de algunos paquetes ofimáticos e incluso formularios en papel escritos a mano. La solución para la gestión y los trámites que habitualmente hace el departamento de administración pasa por la integración de todos los casos de uso en una única herramienta.

Tras un periodo de evaluaciones de alternativas, se contacta con el actual responsable del análisis y desarrollo del software para realizar el proyecto. El personal de administración de la FBCV será el encargado de consultar las exigencias y requisitos con el responsable del análisis, además de la posterior revisión de funcionamiento.

Se plantea asimismo la posibilidad de ampliar los requisitos iniciales de la aplicación en un futuro próximo para llevar parte de la aplicación a la Web.

1.4. Descripción de la situación actual

El personal de gestión de la FCVB está compuesta por 16 personas, las cuales se reparten la gestión completa de la institución, el organigrama actual de la FBCV se corresponde con el siguiente organigrama:



Figura 1.1. Organigrama de la FBCV

Actualmente la responsabilidad de los sistemas de información recae sobre el secretario general, que es así mismo responsable de administración. Las peticiones del personal de administración son recogidas directamente por el secretario general que a su vez si lo considera oportuno las pasa a presidencia, y es el presidente el responsable de aprobar o denegar las peticiones y los pagos de mantenimiento/ampliación de los servicios de TIC, siempre en consenso con tesorería.

Administración recoge directamente las actas de las temporadas firmadas por los árbitros y son los encargados de publicar las estadísticas y clasificaciones semanalmente tanto en la Web como en el tablón de la federación. También es posible que si el jugador ha solicitado el envío por correo electrónico estas listas se le faciliten directamente en formato Excel.

En caso de que algún club impugne el resultado del acta oficial, esta se introduce en el programa y se envía al Comité de apelación para su revisión. Según su resolución, se modificará o no el acta.

Asimismo administración también es la encargada de dar de alta los clubes a principio de temporada, o los jugadores una vez recibido el formulario de inscripción y una vez pagada la cuota, con un importe variable según el tipo de club o el tipo de socio.

El comité de competición marca al principio de temporada las diferentes divisiones y grupos que conformarán la competición, el calendario, las jornadas que se jugarán, etc. etc., toda esta información es recogida por administración e introducida en el sistema de información de la federación.

El diagrama entidad relación básico¹ del Sistema de información es el siguiente:

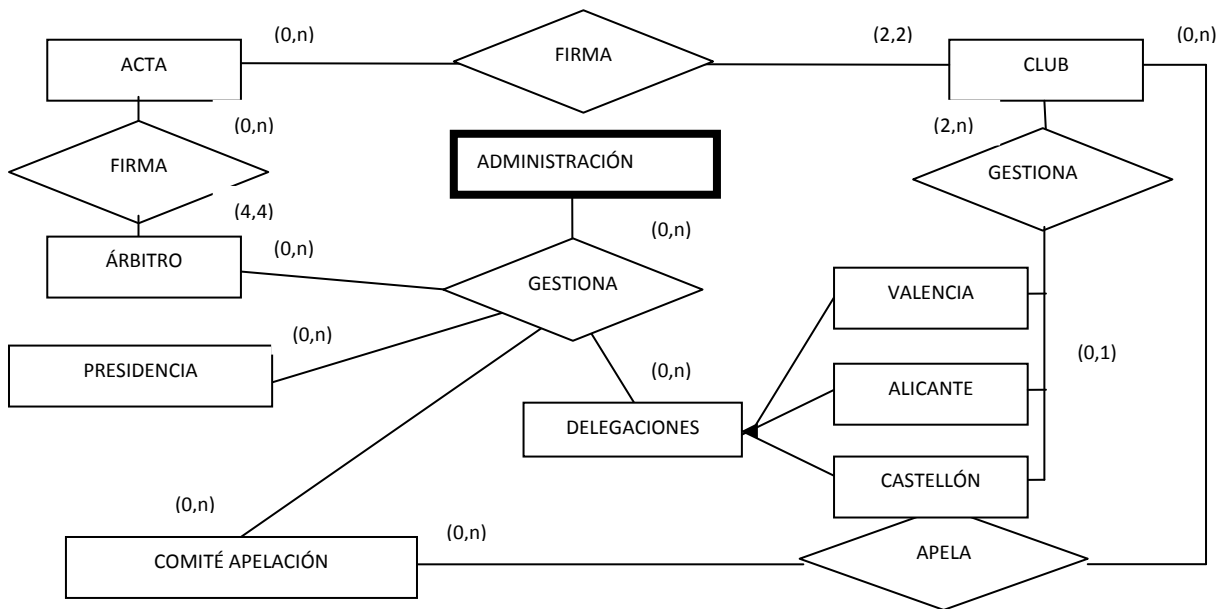


Figura 1.2. Diagrama entidad relación

Para automatizar y gestionar el sistema, se encuentra un único software de gestión llamado BILLGEST producido por la empresa Carbel Sistemas S.L. hace 16 años. El sistema de gestión esta centralizado en un único PC/IBM compatible 486 con MSDOS 6.22 y una impresora Epson EPL-6200N. Además se encuentra otro PC para las gestiones administrativas habituales (email, actualización de la Web, etc...) El programa a desarrollar no contempla en principio la integración de estas funcionalidades por lo que no serán objeto de estudio.

Así, tras un análisis preliminar se establece que carencias tiene el actual software de gestión y cuáles serán las nuevas características que se requieren por parte de administración para que las gestiones se realicen de una manera más eficiente (**ver ANEXOS 1 – Estudio Preliminar del software de gestión actual**).

¹ Modelo E-R Chen Original (1976): <http://www.csc.lsu.edu/~chen/>

1.5. Bases del proyecto

Como se ha comentado anteriormente el proyecto BILLARGEST consiste en llevar a cabo una aplicación para gestionar toda la información de la FBCV que se relacione con clubes, socios y temporadas, quedando fuera de ella todo lo que no sean estas tres entidades y sus relaciones.

Factores críticos de éxito:

-Es necesario disponer de una infraestructura básica para soportar la aplicación a desarrollar, esto es un equipo compatible con el Framework .NET 3.5 y un sistema operativo Microsoft® Windows en versión XP/2000 o Vista/W7. La aplicación se generará tanto para máquinas X86 como X64 habiendo una completa compatibilidad para los 2 tipos de tecnologías.

-Se facilitará gratuitamente un servicio de conexión remota para realizar pruebas telemáticas desde internet con la máquina host que albergará el programa, no obstante en caso de hacer uso de esta herramienta es necesario que personal del departamento esté presente en la máquina física y supervise la conexión remota.

-El departamento de administración debe jugar un papel clave a la hora de la toma de requisitos por parte del consultor, de manera que se facilite tanto información veraz relativa a la propia gestión actual, necesidades vigentes y carencias, e información del manejo actual del flujo de información. También se facilitará todo tipo de formularios e impresos relativos al sistema de información a desarrollar.

- Es necesario que se establezca un medio de comunicación continuo entre el cliente y el responsable del proyecto para evitar retrasos en los avances de la aplicación.

-Asimismo para poder resolver las incidencias de manera rápida y efectiva es necesario que el departamento de administración se implique en la parte de pruebas que corresponde exclusivamente al cliente.

1.6. Metodología general

La metodología general del proyecto es MÉTRICA® en su versión 3².

Esta metodología permite dar soporte al ciclo de vida del proyecto y permite alcanzar los siguientes objetivos:

"- Proporcionar o definir Sistemas de Información que ayuden a conseguir los fines de la Organización mediante la definición de un marco estratégico para el desarrollo de los mismos."

"- Dotar a la Organización de productos software que satisfagan las necesidades de los usuarios dando una mayor importancia al análisis de requisitos."

"- Mejorar la productividad de los departamentos de Sistemas y Tecnologías de la Información y las Comunicaciones, permitiendo una mayor capacidad de adaptación a los cambios y teniendo en cuenta la reutilización en la medida de lo posible."

"- Facilitar la comunicación y entendimiento entre los distintos participantes en la producción de software a lo largo del ciclo de vida del proyecto, teniendo en cuenta su papel y responsabilidad, así como las necesidades de todos y cada uno de ellos."

² MÉTRICA V3: Metodología de Planificación, Desarrollo y Mantenimiento de sistemas de información (28/04/2010): <http://www.csi.map.es/csi/metrica3/>

“- Facilitar la operación, mantenimiento y uso de los productos software obtenidos.”

Empleando esta metodología se asegura la calidad, la seguridad del paquete y sobre todo gran cantidad de recursos documentales en lengua española.

Los procesos que trata la metodología METRICA V3 y serán aplicados al proyecto son:

- 1) **“Proceso de Planificación de Sistemas de Información**, al no estar dentro del ámbito de la norma ISO 12.207 de Procesos del Ciclo de Vida de Software, se ha determinado a partir del estudio de los últimos avances en este campo, la alta competitividad y el cambio a que están sometidas las organizaciones.

El entorno de alta competitividad y cambio en el que actualmente se encuentran las organizaciones, hace cada vez más crítico el requerimiento de disponer de los sistemas y las tecnologías de la información con flexibilidad para adaptarse a las nuevas exigencias, con la velocidad que demanda dicho entorno.

La existencia de tecnología de reciente aparición, permite disponer de sistemas que apoyan la toma de decisiones a partir de grandes volúmenes de información procedentes de los sistemas de gestión e integrados en una plataforma corporativa. MÉTRICA Versión 3 ayuda en la planificación de sistemas de información facilitando una visión general necesaria para posibilitar dicha integración y un modelo de información global de la organización.”

- 2) **“Proceso de Desarrollo de Sistemas de Información**, para facilitar la comprensión y dada su amplitud y complejidad se ha subdividido en cinco procesos:

- ESTUDIO DE VIABILIDAD DEL SISTEMA (EVS).
- ANÁLISIS DEL SISTEMA DE INFORMACIÓN (ASI).
- DISEÑO DEL SISTEMA DE INFORMACIÓN (DSI).
- CONSTRUCCIÓN DEL SISTEMA DE INFORMACIÓN (CSI).
- IMPLANTACIÓN Y ACEPTACIÓN DEL SISTEMA (IAS).”

- 3) **“Proceso de Mantenimiento de Sistemas de Información**, cuyo objetivo es la obtención de una nueva versión de un sistema de información desarrollado con MÉTRICA, a partir de las peticiones de mantenimiento que los usuarios realizan con motivo de un problema detectado en el sistema o por la necesidad de una mejora del mismo.”

MÉTRICA Versión 3 facilita la toma de decisión y la realización de todas las tareas que comprende el desarrollo de un sistema de información.

1.7. Estructura documental

La estructura básica documental del proyecto³ corresponde al siguiente índice:

1. RESUMEN

2. DOCUMENTO I – MEMORIA

- Capítulo 1.- Introducción, objeto, antecedentes y bases.
- Capítulo 2.- Plan de Sistemas de información y selección de alternativas.
- Capítulo 3.- Análisis del sistema de información.
- Capítulo 4.- Evaluación de riesgos.

³ Norma UNE 157801 (2007): Criterios Generales para la elaboración de proyectos de Sistemas Informáticos

Capítulo 5.- Medición, estimación, presupuesto, oferta y programación temporal.
Capítulo 6.- Diseño de sistema de información.
Capítulo 7.- Plan de construcción y pruebas.
Capítulo 8.- Plan de puesta en servicio.
Capítulo 9.- Conclusiones.
Bibliografía.

ANEXOS A LA MEMORIA

ANEXOS 1 – Estudio Preliminar del software de gestión actual.
ANEXOS 2 - Planificación de Sistemas de Información (Proceso PSI).
ANEXOS 3 - Estudio de Viabilidad del Sistema (Proceso EVS).
ANEXOS 4 - Análisis del Sistema de Información (Proceso ASI).
ANEXOS 5 - Diseño del Sistema de Información (Proceso DSI).
ANEXOS 6 - Construcción del Sistema de Información (Proceso CSI).
ANEXOS 7 - Implantación y Aceptación del Sistema (Proceso IAS).
ANEXOS 8 - Mantenimiento del Sistema de Información (Proceso MSI).
ANEXOS 9 - Manual de usuario.
ANEXOS 10 – Presupuesto.
ANEXOS 11 - Diseño de la Base de Datos.
ANEXOS 12 - Ejemplo de Pruebas de Interfaz.
ANEXOS 12 – Pliego de condiciones.
ANEXOS 13 – Evaluación de Riesgos.

1.8. Plan de ampliación

Se considera la opción de ampliar la aplicación objeto de este PFC para dotar a la Web corporativa de toda la información manejada en el programa, para que los usuarios puedan acceder a ella directamente sin necesidad de ser reenviada por parte de administración.

Esta posible ampliación se integraría como una nueva función de negocio en el programa, usando la misma tecnología C# .NET, por lo que no habría ningún tipo de problema para su implementación. La conexión con la BB.DD. se realizaría mediante las propias cadenas de conexión proporcionadas por el Framework sin necesidad de usar un conector ODBC externo. En caso de llevarla a cabo, habría que replantear el sistema físico y operativo ya que sería necesario incluir en el sistema de información de la FBCV algún tipo de servidor.

Las peticiones adicionales se evaluarán y solo se implementarán si no requieren gran tiempo de realización, y siempre posteriormente a las incidencias, además serán gratuitas durante un mes, así pues no serán consideradas en ningún plan.

2. Plan de Sistemas de Información y selección de alternativas

2.1. Situación actual

Actualmente la Federación de Billar de la Comunidad Valenciana dispone de dos ordenadores personales los cuales son manejados por dos personas del departamento de administración. Estas personas son las encargadas de introducir y manejar los datos que circulan por el sistema de información de la Federación.

Estos ordenadores están conectados entre sí mediante un cable de par trenzado ya que uno de ellos realiza funciones de conexión compartida a internet.

Uno de ellos utiliza un sistema operativo Microsoft® Windows XP SP2, y el otro es manejado bajo un sistema Windows® 98 SE. Ambos tienen impresoras, una Epson matricial con drivers DOS específicos configurados en el programa de gestión actual y otra HP Laserjet® 1000 series PostScript 6 compatible.

Los datos que son referentes a la gestión de liga, clubes o socios, son manejados mediante el software de gestión actual BILLGEST, asimismo los datos generados por este software son tratados por el personal del departamento en otros formatos de tecnología más actual, como son listados Excel® 2007, y reenviados mediante programas de correo electrónico a los socios y clubes para que informen a quienes consideren oportuno.

Debido a las carencias del software actual, durante estos años se han ido creando nuevos formularios para dar servicios a sus clientes que no podían ofrecer con el programa de gestión. La mayoría han sido implementados con Macros Excel 2003. Estos formularios se utilizan como complemento al software BILLGEST.

También se hace uso de la herramienta FrontPage para la modificación de la página web, de la cual se realizan actualizaciones periódicamente.

2.2. Plan de Sistemas de Información

El departamento de administración requiere de mucho tiempo y esfuerzo en suplir las carencias que provoca el uso del programa actual de gestión BILLGEST. Las carencias principales son:

- Nula capacidad de ampliación y actualización.
- Falta de soporte.
- Limitada capacidad de ampliación de hardware y mantenimiento que de soporte al software.
- Formatos de ligas de hace 15 años.
- Limitaciones en los datos de entrada de la aplicación.
- Problemas de integridad en los datos.
- Problemas con el sistema de backup.
- Interfaz gráfica poco usable y manejable.
- Nula portabilidad.
- Alto coste en tiempo al tener que transformar los datos de salida de la aplicación en entregables a los asociados.

El presente proyecto trata de suplir todas estas carencias e integrarlas en una nueva aplicación que de servicio a la FBCV.

2.3. Estudio y selección de alternativas

Una vez expuestos los problemas de la aplicación actual y los requerimientos básicos por parte de administración a presidencia, y una vez aprobado el seguimiento del plan, se hace necesario seleccionar el tipo de herramienta que pueda suplir el programa actual.

Para ello se analizan diferentes alternativas:

Alternativa 1: Adaptación integral de un programa de gestión en producción ya existente de otra Federación de Billar

Esta alternativa fue una de las primeras consideraciones que se tuvo en cuenta a la hora de seleccionar un software adecuado que pudiera dar soporte al departamento. La Federación Catalana de Billar ya disponía de un software que podía suplir la mayor parte de las exigencias de administración de la FBCV. Este software era capaz de:

- Gestionar Clubes y Socios
- Gestionar Temporadas anuales
- Gestionar Reservas en las mesas de juego de la Federación
- Gestión de documentación y normativa asociada
- Registro de árbitros
- Registro contable

La aplicación, desarrollada por una empresa Catalana en exclusiva para la Federación Catalana de Billar (FCB), superaba en cierta medida las necesidades actuales de la FBCV, no obstante se hacía necesaria una pequeña remodelación para adaptarlo al sistema de liga interprovincial que se lleva a cabo en esta comunidad.

El factor limitante de esta alternativa es sin duda el precio, muy superior a las posibilidades con las que cuenta actualmente las arcas de la Federación Valenciana.

Alternativa 2: Ampliación del programa actual

La ampliación o actualización del programa actual BILLGEST fue la primera opción que se tuvo en cuenta para mejorar el sistema de información de la Federación.

Desgraciadamente la compañía creadora ya no existe, y por tanto se parte de un programa totalmente cerrado y sin posibilidad de edición. No obstante se piensa en la posibilidad de realizar módulos independientes que editen las entradas y salidas del programa.

2.1. Ampliación en la gestión de listados

La adaptación del módulo de exportación consiste en crear un convertidor de formatos de listados propietarios BILLGEST a formatos editables Microsoft® Office 2007. Para llevar a cabo esta tarea se piensa en la posibilidad de encargar la realización de Macros Visual Basic v6 a la empresa que desarrolló el software para la FCB. La ventaja clara de esta opción es el coste relativamente bajo.

2.2. Creación de un módulo independiente para la gestión de carnets de socios

De manera similar a la ampliación anterior, se piensa en crear otro pequeño módulo que integre la impresión de carnets de socios, mediante una BB.DD. que funcione de manera independiente de la BB.DD. principal, ya que después del pre análisis no se observa posibilidad de acceso a las BB.DD. de BILLGEST. El problema radica en la posible redundancia de datos.

Alternativa 3: Programa sustitutorio íntegro

3.1. Aplicaciones comerciales

Las herramientas de gestión de ligas son muy comunes, y es posible desde adquirir aplicaciones comerciales como es el paquete "Competiciones deportivas V.10" de la empresa Sagois S.L.U. a usar servicios Web de Gestión de Ligas como el que ofrece Esportalia S.L.

Concretamente se estudia el programa de Competiciones deportivas V.10 de la empresa Sagois S.L.U. cuyas principales funciones de negocio son:

CALENDARIO

- Creación y gestión de campeonatos de liga (a 1 o 2 vueltas) y copa, así como fases previas y finales.
- Realiza el sorteo del campeonato de manera automática a 1 o 2 vueltas con posibilidad de establecer la primera jornada de manera manual. Es posible establecer todas las jornadas de manera manual.
- Genera los enfrentamientos de copa. Permite generar previas de liga y final de copa.
- Creación de calendarios, pudiendo seleccionar entre calendarios completos, por jornadas, o de un rango de jornadas. Posibilidad de mostrar la clasificación actual en la parte inferior de la página.
- Gestión profesional de horarios: Creación de calendarios con un número ilimitado de campos de juego. Además, ahora es posible reservar el campo de juego por parte de usuarios, y el programa tiene en cuenta estas reservas a la hora de generar el calendario para evitar coincidencias horarias.

CLASIFICACIONES

- Generación de clasificaciones de equipos y jugadores.
- Equipos: clasificación general, deportividad, equipos menos goleados.
- Jugadores: Goleadores, Deportividad y equipos menos goleados.

SANCIONES

- Imprime las actas de los partidos, informándonos de qué jugadores están sancionados.
- Gestión profesional de árbitros y de sanciones.
- Listas negras de DNI: permiten la detección automática, por medio del DNI, de un jugador al que no se le permita inscribirse en un campeonato.

INTERNET

- Exportación automática de equipos, calendarios y clasificaciones a HTML.

- Posibilidad de subir las páginas a un servidor privado.
- Creación de copias de seguridad, con posibilidad de subirlas a internet.

OTROS

- Nueva base de datos de usuarios con la posibilidad de importarlos a los equipos a lo largo de los años.
- Posibilidad de almacenar equipos y jugadores para poder utilizarlos en campeonatos posteriores.
- Creación de hojas de inscripción.
- Creación de carnets.
- Gestión Económica (Sólo versión élite).

El coste de la aplicación entra dentro de la partida destinada a la actualización del software de la FBCV, pero no así el coste de la ampliación para adaptarlo a las necesidades específicas de la FBCV, como es la adopción de listas de fuerza y generación de las diferentes divisiones.

3.2 Desarrollo desde cero de la aplicación

Consiste en el desarrollo de una sola aplicación que disponga de información centralizada de todas las funciones requeridas para mantener el sistema de información de la FBCV. La aplicación correspondería por tanto a una *actualización* de la herramienta de gestión de la que se dispone actualmente.

3. Análisis del Sistema de Información

3.1. Alcance del sistema

El alcance del sistema debe cubrir toda la operatoria de gestión de clubes, de las tres provincias de la Comunidad Valenciana, la gestión de los Socios de los respectivos clubes, y el tratamiento de las ligas que se juegan anualmente. El límite del alcance radica por tanto, en las entidades externas de Club y Socio. Si bien la gestión estará centralizada por el departamento de administración, habrá que fijar en el catálogo de requisitos las exigencias particulares de cada una de estas dos entidades.

Adicionalmente, el sistema de información contempla la actualización de equipos, una nueva gestión de BB.DD. y la inclusión de nuevos paquetes ofimáticos para dar soporte a las extensiones del sistema de gestión de ligas a implantar.

También es importante destacar que el desarrollo no contempla el acceso al programa de gestión con distintos roles. Lo que si entra dentro del alcance es la posibilidad de exportación de datos en un formato propio, lo que permitirá la instalación en diferentes equipos de la FBCV, siempre trabajando sobre una única versión de BB.DD.

El resto de entidades de la FBCV quedan fuera del ámbito del programa y no se desarrollarán.

3.2. Entorno tecnológico

El entorno tecnológico que soportará la carga del sistema de información corresponde a las siguientes configuraciones:

Equipos de administración (2):

- Caja MicroATX
- Microprocesador Intel Dual Core E5200
- Placa base Asus mod. P S775
- Disco duro 1TB
- Memoria RAM 4GB 677 Mhz
- Red Ethernet 10/100/1000
- Impresora HP 1022
- Monitor TFT HP 22" Mod. W2216v
- S.O. Microsoft® Windows 7 Home

Equipos de red:

- Switch 4 puertos 10/100/1000 SMC Mod. 24C

3.3. Estándares y normas

Metodología de desarrollo: El proyecto se realizará siguiendo las directrices de Métrica en su versión 3.

Protección de datos: Se establecerán los ficheros conforme a la Ley Orgánica de Protección de Datos de 1999, de esta manera los ficheros con datos de carácter personal tendrán que inscribirse en la Agencia de Protección de datos, responsabilidad de la FBCV, y su finalidad será la gestión de los contenidos y comunicaciones comerciales relacionadas con la actividad de la Federación. La adopción de la LOPD será llevada a cabo por una empresa externa.

Implementación: La implementación se realizará mediante el lenguaje C# .NET3.5 y el entorno de desarrollo de Microsoft® Visual Studio 2008. La BB.DD. será desarrollada en lenguaje Microsoft® Access 2007. Gracias al empleo de esta tecnología, se consigue un desarrollo ágil, potente y orientado 100% a objetos.

Modelo de desarrollo: Prototipado evolutivo, se construye una serie de grandes versiones sucesivas de un producto. El modelo evolutivo asume que los requerimientos no son completamente conocidos al inicio del proyecto. Una vez se crea el primer desarrollo, los usuarios lo usan, y proveen retroalimentación a los desarrolladores. Basada en esta retroalimentación, la especificación de requerimientos es actualizada, y una segunda versión del producto es desarrollada y desplegada. El proceso se repite indefinidamente.

Modelo de BB.DD.: Modelo entidad-relación de bases de datos.

Análisis y diseño: Se usarán los modelos contenidos en Lenguaje de Modelado Unificado UML⁴ v2. La técnica para la captura de requisitos potenciales de la actualización de requisitos es el modelo de Casos de Uso y para la especificación se usa la Guía IEEE Std. 830-98⁵.

Plataforma y Arquitectura: La plataforma del sistema será Microsoft® Windows Framework 3.5, con arquitectura de tres capas y un nivel.

3.4. Requisitos funcionales

La captura de requisitos tras el estudio de los formularios de la FBCV, las entrevistas con el personal de administración y el estudio preliminar de su anterior sistema de gestión BILLGEST se resumen en:

Gestión de Temporadas:

- Gestión de temporadas. Alta, baja y modificación.
- Copias de seguridad. Apertura, Cierre, Salvado.
- Especificación de número de grupos y divisiones por club.
- Acceso directo a diferentes tipos de listado.
- Exportación de listados a formato compatible Microsoft® Excel.
- Impresión de listados por cualquier impresora compatible con Microsoft® Windows.
- Información relativa a los clubes enfrentados y socios que las disputan.
- Gestión de actas de temporadas. Alta, baja y modificación.
- Sanciones por ruptura de listas de fuerza.
- Gestión de árbitros.

⁴ UML V2.0: Unified Modeling Language (28/04/2010): <http://www.uml.org/>

⁵ Estándar IEEE para la especificación de requisitos de software (28/04/2010): http://www.ctr.unican.es/asignaturas/is1/IEEE830_esp.pdf

Gestión de Socios:

- Gestión de socios. Alta, baja y modificación.
- Fichero de Datos personales.
- Tarifas de Socio (Junior / Sénior).
- Listas de Fuerza.
- Bajas temporales, bajas definitivas.
- Diseño e impresión de carnets por cualquier impresora compatible con Microsoft®.
- Acceso directo a diferentes tipos de listado.
- Exportación de listados a formato compatible Microsoft® Excel.
- Impresión de listados por cualquier impresora compatible con Microsoft®.

Gestión de Clubes:

- Gestión de clubes. Alta, baja y modificación.
- Apartado de solicitudes a la FBCV.
- Acceso directo a diferentes tipos de listado.
- Exportación de listados a formato compatible Microsoft® Excel.
- Impresión de listados por cualquier impresora compatible con Microsoft®.

3.5. Análisis de clases

Socio: Persona asociada al club que paga una cuota anual a cambio de los derechos de participación en los diferentes partidos de la liga. Además, tiene derecho de acceso a las salas de billar de sus respectivos clubs, entrenamientos guiados, revista mensual, descuentos en material,... Un socio puede o no jugar las ligas propuestas por la FBCV.

Club: Sociedad deportiva que consta de nombre, presidente o responsable y esta dado de alta en la Federación de su comunidad. Puede optar a subvenciones locales.

Temporada: Corresponde con el calendario que va desde Septiembre de un año hasta Julio del año siguiente.

Acta: Documento escrito con los resultados e incidencias de los encuentros. Cuando se aprueba (firma por el director del encuentro y los respectivos capitanes), se incluye en la lista de actas aprobadas las cuáles se usan para modificar la clasificación general.

Lista de fuerza: Lista con el orden preferente de los jugadores de cada club.

División: Cada uno de los grupos que componen la liga.

Grupo: Cada uno de los subgrupos que componen las Divisiones, formados por un máximo de doce equipos.

Jornada: Corresponde a los partidos que se llevan a cabo cada cierto periodo de tiempo.

Billar Francés 3 bandas: Modalidad de Billar donde se anota una carambola o punto si la bola golpea 3 bandas antes de completar la carambola.

Billar Americano Pool Continuo: Modalidad de Billar donde gana el jugador que sea el primero en introducir un número determinado de bolas acordado previamente.

Billar Libre: Corresponde a cualquier modalidad que no sean las 2 anteriores.

3.6. Interfaces de usuario

Para el desarrollo no se contempla la posibilidad del uso de diferentes roles en la aplicación, por lo que solo habrá un tipo de acceso predeterminado con todas las opciones disponibles.

Básicamente, el programa dispondrá de menús y submenús con las principales funciones de negocio en la parte superior. En la zona de trabajo, estarán incluidas todas las ventanas emergentes desplegadas al seleccionar los menús.

4. Evaluación de riesgos

4.1. Riesgos identificados

Riesgo es la Condición o evento incierto, que de darse, tiene efecto en al menos un objetivo del proyecto (ámbito, calendario, coste o productividad). Por tanto se hace necesario abordar, preparar y planificar las actividades de gestión de riesgos del proyecto para reducir los efectos adversos que puedan provocarse.

Para la clasificación de riesgos se utilizarán las tablas propuestas por TeraQuest⁶ que proporcionan una visión global de riesgos del proyecto (**ver ANEXOS 10 – Evaluación de Riesgos**).

Una vez identificados los riesgos, se calcula⁷ el efecto en euros que supone la exposición a dicho riesgo. De esta manera se tendrá una visión global del posible impacto económico en el proyecto.

4.2. Planes de acción

Algunas de las medidas que se van a llevar a cabo en el proyecto BillGEST para evitar la exposición de riesgos son:

- Durante todo el desarrollo del proyecto, deberá realizarse por el Jefe de Proyecto un seguimiento del cumplimiento de estándares de la calidad y de la normativa ISO en cuanto a desarrollo de proyectos informáticos se refiere y calidad en los mismos.
- Para evitar que la inexperiencia del cliente produzca fracaso del proyecto, se realizarán entrevistas con el mismo de forma periódica, de manera que se le va enseñando lo que se encuentra realizado y se le pide opinión para poder encontrar requisitos cambiantes en las fases más tempranas del proyecto. Además, a través de estas reuniones se le ofrece poca familiarización con el producto, haciéndosele más sencilla su utilización cuando se encuentre finalizado.
- A la hora de introducir datos históricos en la aplicación, se consultará con el cliente si dicho proceso desea que sea llevado a cabo por nuestra organización con el consiguiente aumento en el coste o si por el contrario será realizado por otra persona, evitando así riesgos de aumentos de presupuesto injustificados.
- Debido a la inexperiencia del cliente, se proporcionará un registro de todos los acontecimientos ocurridos en el sistema, de forma que si alguno de estos ocasiona un fallo, a través de esta documentación, el desarrollador podrá saber los pasos que ocasiono el fallo.
- Se le entregará al cliente una documentación del producto final, sobre la que se podrá encontrar también un manual de usuario con la descripción completa del uso de la herramienta. Como complemento se les ofrecerá a los usuarios de la aplicación un curso de formación que será impartido por el responsable del proyecto.

⁶ TeraQuest, Riesgos genéricos del proyecto (Diciembre, 1999): <http://ais.msu.edu/internal/projectmgmt/documents/ProjectRiskFactors.pdf>

⁷ Para el cálculo de probabilidades se usará una versión simplificada de la gestión de riesgos: RISK MANAGEMENT REPORTING (NASA, 2009): <http://standards.gsfc.nasa.gov/gsfcd-std/gsfcd-std-0002.pdf>

El cálculo del coste se basa en el coste aproximado en caso de que el riesgo se cumpliera completamente (Amenaza signo – Oportunidad signo +)

- Para asegurar que el producto final tiene la máxima calidad, se buscarán casos de prueba con valores límite.

4.3. Planes de contingencia

Para establecer el plan de contingencia se van a tener en cuenta los riesgos con mayores posibilidades de afectar a la evolución del proyecto.

- Si por la implantación de la nueva herramienta se producen cambios en los flujos de datos del cliente, se le enseñara al cliente cual es este flujo de datos, como se origina y cuál es el nuevo comportamiento que tienen.
- En caso de que durante la realización del proyecto, aparezcan problemas de implementación los cuáles el responsable del proyecto no sea capaz de resolver, se fomentará un contrato con una consultora experta en tecnología .NET que cubrirá las insuficiencias técnicas de la empresa mediante la impartición de algún curso.
- En caso de que se produzca un defecto grave en el sistema, el responsable se compromete a hacerse cargo del coste de dicho problema sin coste para la FBCV. Esto supone que la calidad y los procesos que la controlan deben ser inspeccionados de forma cuidadosa.
- Si no se cumpliesen las fechas marcadas como consecución de tareas, se respetarán los tiempos asignados a cada tarea y se retrasará la fecha de entrega final del proyecto tantos días como se haya retrasado la suma de todas las tareas.
- En caso de que cambie la reglamentación de los campeonatos de billar, las modificaciones que requiera la aplicación por estas causas deberán ser tenidas en cuenta como ampliaciones o modificaciones del proyecto, y en ningún caso su coste será asumido por el proveedor.

5. Medición, Estimación, Presupuesto, Oferta y Programación temporal

5.1. Cálculo de puntos de función

Para medir el tamaño del software se usan los denominados Puntos de Función definidos por Allan Albrecht en 1979⁸. A través de ellos se puede medir la funcionalidad de una aplicación determinada independientemente de la tecnología utilizada para su construcción. A través de este método, se calculan los puntos de función de la aplicación, apareciendo:

Puntos de Función Sin Ajustar (PFSA): 176
 Puntos de Función Ajustados: 135,72

Para realizar su cálculo se utilizan las siguientes tablas:

Ficheros Lógicos Internos			
Nombre fichero	TERregistro	TEDato	Complejidad
Temporada	2	5	B
Clasificacion_club	0	11	B
Clasificacion_jugador	0	12	B
Socio	1	12	B
Club	0	10	B
Acta	4	14	B
Temporada_Club	0	2	B
Socio_Acta	1	10	B
Total Ficheros Lógicos Internos		Baja	8
		Media	0
		Alta	0

Figura 4.1. Ficheros Lógicos Internos

Siendo:

TED (tipo elemental de datos): es un único campo, no recursivo y reconocible por el usuario en un FLI o FIE.

TER (tipo elemental de registro): subgrupo de datos elementales reconocible por los usuarios en un FLI o FIE. Existen dos tipos de subgrupos, opcional, cuando los usuarios los pueden usar o no, y obligatorios, cuando los usuarios han de utilizar al menos un subgrupo.

Ficheros Externos de Interfaz			
Nombre fichero	TERregistro	TEDato	Complejidad
ClubFederacion	4	25	M
ClubMinisterio	8	30	A
Total Ficheros Externos de Interfaz		Baja	0

⁸ Allan J. Albrecht and John E. Gaffney, "Software Function, Lines of Code, and Development Effort Prediction: A Software Science Validation", IEEE Transactions on Software Engineering, vol SE-9, No 6, (Noviembre, 1983). Ejemplo de aplicación en: <http://computing.unn.ac.uk/staff/CGPV1/downloadables/fpa.pdf>

Media	1
Alta	1

Figura 4.2. Ficheros Externos de Interfaz

Entradas Externas				
Evento	Nombre del proceso	RefAFich	TEData	Complejidad
1,1	Datos nuevo socio	2	12	M
1,2	Datos editar socio	3	12	A
1,3	Datos nueva temporada	4	15	A
1,4	Datos nuevo club	4	17	A
1,5	Datos editar club	2	14	M
1,6	Datos nueva acta	5	23	A
1,7	Datos editar acta	5	26	A
1,8	Datos sancionar	3	25	A
Total Entradas Externas			Baja	0
			Media	2
			Alta	6

Figura 4.3. Entradas externas

Siendo:

RAF (referencia a fichero): Un fichero Lógico Interno (ILF) leído o mantenido por una transacción o un fichero de Interfaz externo (FIE) leído por una transacción.

Salidas Externas				
Evento	Nombre del proceso	RefAFich	TEData	Complejidad
2,1	Promedio de clubes	3	15	M
2,2	Promedio de jugadores	3	18	M
2,3	Visualización de jornadas	4	20	A
2,4	Consulta clasificación	2	16	M
2,5	Listado temporadas	1	11	B
2,6	Listado actas	1	10	B
2,7	Listado socios	1	5	B
2,8	Listado clubes	1	11	B
Total Salidas Externas			Baja	4
			Media	3
			Alta	1

Figura 4.4. Salidas externas

Consultas Externas		Entrada		Complejidad Entrada
Evento	Nombre del proceso	RefAFich	TEData	
3,1	Elegir categoría	1	3	B
3,2	Introducir división	1	4	B
3,3	Introducir jornada	1	24	M
3,4	Introducir grupo	1	6	B
3,5	Introducir club	1	50	M
3,6	Introducir jugador	1	50	M

Figura 4.5. Consultas externas 1

Consultas Externas		Salida		Complejidad Salida
Evento	Nombre del proceso	RefAFich	TEData	
3,1	Elegir categoría	1	1	B

3,2	Introducir división	1	1	B
3,3	Introducir jornada	1	1	B
3,4	Introducir grupo	1	1	B
3,5	Introducir club	1	1	B
3,6	Introducir jugador	1	1	B

Figura 4.6. Consultas externas 2

Consultas Externas		
Evento	Nombre del proceso	Complejidad Entradas+Salidas.
3,1	Elegir categoría	B
3,2	Introducir división	B
3,3	Introducir jornada	M
3,4	Introducir grupo	B
3,5	Introducir club	M
3,6	Introducir jugador	M
Total Consultas Externas		3
		3
		0

Figura 4.7. Consultas externas 3

#	Factor de Complejidad	Valor (0..5)
1	Comunicación de Datos.	3
2	Proceso Distribuido.	1
3	Objetivos de Rendimiento	0
4	Configuración de Explotación compartida	1
5	Tasa de Transacciones	0
6	Entrada de Datos EN-LÍNEA	2
7	Eficiencia con el Usuario Final	1
8	Actualizaciones EN-LÍNEA	1
9	Lógica del Proceso Interno Compleja	0
10	Reusabilidad del Código	1
11	Contempla la Conversión e Instalación	1
12	Facilidad de Operación	0
13	Instalaciones Múltiples	1
14	Facilidad de Cambios	0
Factor de Complejidad Total (FCT)		12

Figura 4.8. Factores de complejidad

Tipo Elemento	Dificultad	Peso	Cantidad	Total	Total Elemento
Ficheros Internos	Simple	7	8	56	
	Media	10	0	0	
	Compleja	15	0	0	
	Total Ficheros Internos				
Ficheros de Interfaz	Simple	5	0	0	
	Media	7	1	7	
	Compleja	10	1	10	
	Total Ficheros de Interfaz				
Entradas	Simple	3	0	0	
	Media	4	2	8	

	Compleja	6	6	36	
	Total Puntos de Función Entradas				44
Salidas	Simple	4	4	16	
	Media	5	3	15	
	Compleja	7	1	7	
	Total Salidas				38
Consultas:	Simple	3	3	9	
	Media	4	3	12	
	Compleja	6	0	0	
	Total Consultas				21
Total Puntos de Función Sin Ajustar (PFSA)					176

Figura 4.9. Puntos de función sin ajustar

	VFA		
Puntos de función Ajustados	0,77		135,52
Estimación	135,52	C#	474,32 horas

A la vista de dichos puntos de función se puede estimar el esfuerzo de dicho proyecto en torno a 474,32 horas para desarrollarlo sobre un lenguaje de alto nivel y quinta generación como es C# 3.5 (aproximadamente 3,5 horas⁹ por punto de función ajustado).

5.2. Descomposición del esfuerzo por fases y presupuesto

Una vez calculado el coste en horas para el desarrollo de la aplicación se establecen las tablas de coste unitario horario según actividad y presupuesto general.

Actividad	Coste € / hora
Planificación del Sistema	20 €
Estudio de Viabilidad	20 €
Análisis del Sistema	20 €
Diseño del Sistema	18 €
Construcción del Sistema	18 €
Implantación y pruebas	18 €
Documentación	18 €
Formación	15 €

Figura 4.10. Coste/hora según actividad

⁹ Según los datos del estudio del Software Productivity Research (2009): <http://www.gotdotnet.com/team/compare/petshop.aspx>

Concepto	Horas	Porcentaje Final	Precio unitario / hora (IVA excluido)
Planificación del Sistema	48,85	10,30%	977 €
Estudio de Viabilidad	27,79	5,86%	555,8 €
Análisis del Sistema	64,55	13,61%	1.291 €
Diseño del Sistema	55,68	11,74%	1.002,2 €
Construcción del Sistema	124,74	26,30%	2.245,3 €
Implantación y pruebas	74,27	15,66%	1.336,8 €
Documentación	49,94	10,53%	898,92 €
Formación	28,45	6,00%	426,75 €
	BASE IMPONIBLE TOTAL DE LA OFERTA		8.733,77 €
	IVA		1.397,40 €
	PRECIO TOTAL DE LA OFERTA		10.131,17 €
	PRECIO MÁXIMO ESCENARIO HIPOTÉTICO <=		10.500 €

4.11. Oferta económica

5.3. Oferta al cliente

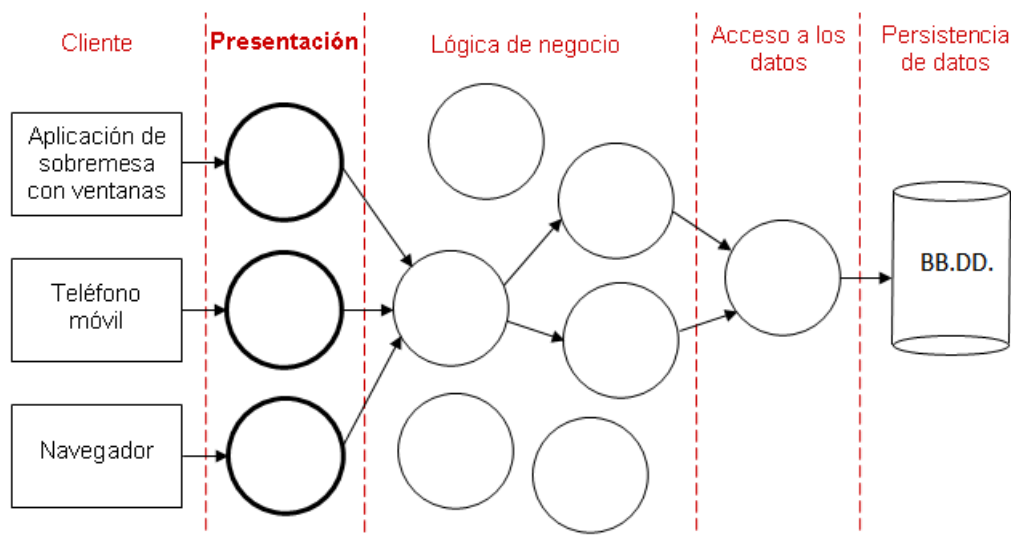
Tras el estudio en profundidad de los costes de la implantación del sistema y el margen de beneficio requerido, se asume por tanto un precio de venta de doce mil euros (12.000 €) IVA incluido.

Debido a la posibilidad de comercializar la aplicación para otras Federaciones (en principio existen 17 clientes potenciales de los cuales uno parece estar interesado), se plantea la posibilidad de rebajar el precio final del paquete, que quedaría en tres mil novecientos noventa y cinco euros (3.995 €) IVA incluido.

6. Diseño del sistema de información

6.1. Niveles de arquitectura

La arquitectura lógica del sistema es una arquitectura multicapa, compuesta por 3 capas¹⁰ (presentación, lógica de negocio y acceso a datos):



6.1. Capas de una arquitectura multicapa.

1- Capa de presentación: es la que ve el usuario (también se la denomina "capa de usuario"), presenta el sistema al usuario, le comunica la información y captura la información del usuario en un mínimo de proceso (realiza un filtrado previo para comprobar que no hay errores de formato). Esta capa se comunica únicamente con la capa de negocio. También es conocida como interfaz gráfica y debe tener la característica de ser "amigable" (entendible y fácil de usar) para el usuario.

2- Capa de negocio: es donde residen los programas que se ejecutan, se reciben las peticiones del usuario y se envían las respuestas tras el proceso. Se denomina capa de negocio porque es aquí donde se establecen todas las reglas que deben cumplirse. Esta capa se comunica con la capa de presentación, para recibir las solicitudes y presentar los resultados, y con la capa de datos, para solicitar al sistema de gestión de BB.DD. para almacenar o recuperar datos de él. También se consideran aquí los programas de aplicación.

3- Capa de datos: es donde residen los datos y es la encargada de acceder a los mismos. Está formada por un gestor de base de datos relacionales (S.G.B.D.) Access 2007, que realiza todo el almacenamiento de datos, recibe solicitudes de almacenamiento o recuperación de información desde la capa de negocio.

De esta manera, se consigue un programa con una mayor escalabilidad, que permitirá en un futuro la ampliación de funcionalidad sin perder la integridad principal lógica de la aplicación.

La arquitectura física será una arquitectura de un nivel¹¹, compuesta simplemente de una estación de trabajo que dará soporte a todas las capas lógicas.

¹⁰ Arquitectura multicapa Wikipedia (2010): <http://es.wikipedia.org/wiki/Cliente-servidor>

¹¹ Programación por capas, Wikipedia (2010): http://es.wikipedia.org/wiki/Programación_por_capas

6.2. Requisitos de diseño

El diseño orientado a objetos con el que se desarrollara el sistema requiere de una plataforma compatible. En este caso, la plataforma de desarrollo es .NET de Microsoft[®], su entorno de desarrollo Visual Studio 2008, y el lenguaje de desarrollo, orientado 100% a objetos, será C#.

El desarrollo será iterativo e incremental, de la siguiente manera:

- Obtener los máximos requisitos posibles en el tiempo fijado. Ordenarlos y fijar los más importantes.

- Obtener los máximos casos de uso y actores posibles. Ordenarlos por importancia (más importantes para el usuario, más difíciles de implementar o que ayuden a definir lo máximo posible la arquitectura) y detallar sólo los primeros en la lista.

- Obtener la arquitectura para esos primeros casos de uso.

- Diseño detallado para esos primeros casos de uso, siempre con un plazo.

- Codificar y probar.

Volver a iterar con nuevos requisitos y casos de uso surgidos, etc., etc....

La programación orientada a objetos permite concebir los programas de una manera bastante intuitiva y cercana a la realidad, aumentando la productividad, la reutilización de código y la mantenibilidad¹².

6.3. Excepciones

Las excepciones provocadas por el sistema serán todas debidamente tratadas por el programa. Es posible que en algún caso la excepción provoque que el sistema se vuelva inestable y no se pueda recuperar, por lo que todo error llevará un código marcado con el que se identificará claramente el problema y se podrá recurrir a una tabla para su posterior solución de incidencia.

No obstante se intentará en la medida de lo posible realizar un buen filtrado de datos de entrada para evitar muchas de las excepciones de error.

6.4. Entorno tecnológico

Como se ha explicado anteriormente, el sistema se prevé instalar sobre una arquitectura de un solo nivel físico, esto es todas las capas de la aplicación radican en una sola estación de trabajo.

Para la correcta ejecución del software asociado al proyecto es necesario que el sistema posea las librerías del Framework 3.5 de Microsoft[®]. Estas librerías son totalmente gratuitas y de libre descarga. No obstante, se incluirán en el programa y se auto instalarán en caso de no estar previamente descargadas.

¹² Miguel Angel Álvarez, Programación orientada a objetos en PHP (2009): <http://www.desarrolloweb.com/articulos/1540.php>

7. Plan de construcción y pruebas

7.1. Plan de Construcción

El paquete se realiza con el entorno de programación Microsoft® Visual Studio .NET 2008 usando el Framework 3.5 y el lenguaje de programación C# 3.5.

Las pruebas unitarias se realiza con la herramienta Open Source NUnit así como con las propias herramientas incluidas en el entorno de programación.

La BB.DD. se realiza con el entorno Visual Studio en formato Access 2007. La aplicación para la generación de reportes (para los carnets de socio) es CrystalReports® .NET

La herramienta de creación de paquetes autoinstalables corresponderá con el paquete InstallShield® 2010.

7.2. Alcance de las pruebas

Las pruebas del ciclo de vida que se realizarán serán:

- Pruebas Unitarias
- Pruebas de Integración
- Pruebas Funcionales
- Pruebas de Rendimiento

7.3. Plan de pruebas del sistema

Para el plan de pruebas se realizan pruebas de verificación de requerimientos y pruebas del sistema que incorpora el software.

Los pasos para la realización de las pruebas son:

- Revisar la verificabilidad del requerimiento.
- Especificar el criterio de verificación.
- Hacer visible las propiedades o elementos del software necesarios para verificar el cumplimiento del requerimiento.
- Hacer controlable los elementos del software necesarios para llevar a cabo las pruebas.
- Elaborar el plan de pruebas.
- Ejecutar el plan de pruebas y reportar sus resultados.

Cada caso de prueba se detallará en un documento que recogerá todas las pruebas de una determinada funcionalidad de la aplicación.

8. Plan de puesta en servicio

8.1. Plan de formación

El plan de formación se basará en sesiones dirigidas al personal del departamento donde se actualizará el sistema de información.

El plan se dividirá en sesiones teórico/prácticas, donde se expondrán casos reales de uso y se explicará el funcionamiento completo de la aplicación.

Para el plan se hará uso de material de formación como son las diapositivas electrónicas.

8.2. Plan de manual de usuario

El manual de usuario se compondrá básicamente de:

- Introducción
- Instalación y desinstalación del paquete
- Descripción del entorno de trabajo
- Funciones del programa
- Solución de errores

9. Conclusiones

Varios han sido los factores que han conseguido que el proyecto se implante con éxito y que sea una plataforma robusta y fiable de gestión para la FBCV. Los principales factores son los siguientes:

1. Negociación

En este factor se determinan las dos variables principales del proyecto, el tiempo y el costo. Es clave ya que identifica si el proyecto tiene los argumentos para ser exitoso, o están en riesgo el alcance y las expectativas de ambas partes. Gracias a que el tiempo no era una prioridad para la FBCV y los requisitos eran muy concretos, no existieron grandes problemas en la negociación, y las exigencias fueron perfectamente aceptadas tanto por desarrollador como por cliente.

2. Tecnología

En este factor se determina sobre qué plataforma tecnológica se desarrolla el proyecto. La selección de la tecnología puede darse por muchos factores, desde el factor económico, hasta el factor de imposición por el cliente o por la empresa desarrolladora. Se tuvo claro desde un primer momento que el desarrollo se haría con una plataforma 100% orientada a objetos y de última tecnología. También era necesario que el desarrollador tuviera experiencia con el lenguaje, y que fuera una solución de bajo coste de desarrollo. Así pues se optó por la versión académica gratuita de Microsoft® Visual Studio 2008 y por varias herramientas de libre distribución que acompañan al paquete. Al cumplir perfectamente para el cliente una implantación con esta tecnología, el proyecto no tuvo problemas al respecto.

3. Metodología

La selección de la metodología de desarrollo adecuada depende del tiempo, costo y tecnología seleccionada. Hay que decidir que metodología de trabajo hay que usar según las características del proyecto para poder cumplir las expectativas funcionales y de negocio esperadas. En el proyecto se han usado metodologías altamente testadas y relacionadas con la plataforma de desarrollo elegida, que además han demostrado ser efectivas y cumplir con éxito en muchos proyectos de T.I.C. Estas metodologías eran conocidas de antemano por el desarrollador lo que agilizó en gran medida el tiempo de arranque de los procesos. Tampoco habían exigencias al respecto por parte del cliente.

4. Recursos

El último factor del cual depende el éxito del proyecto son los recursos que estarán involucrados, es decir, las personas y sus respectivos perfiles de conocimientos y experiencia en el tipo de proyecto, metodología de trabajo y tecnología. Todo el peso del proyecto recayó sobre una persona, y si bien existía el riesgo de bloqueo, también agilizaba el desarrollo ya que no se producían problemas de comunicación y mala planificación entre los integrantes del equipo.

Para la FBCV la puesta en marcha del software BILLARGEST ha permitido suplir todas las carencias que arrastraban por el uso de su antiguo programa de gestión:

- Mejorada interfaz gráfica, más intuitivo y flexible.
- Adecuación a las nuevas normativas de la FBCV.
- Adecuación a las nuevas exigencias del personal de administración.
- Independencia del hardware con respecto al software.
- Posibilidad de actualización futura.
- Soporte y asistencia.

El presente proyecto además servirá de base para las futuras ampliaciones que pretende realizar la FBCV sobre el actual programa, y para conseguir que su herramienta en un futuro mantenga como mínimo los estándares actuales de calidad y fiabilidad.

Bibliografía¹³

- [AENOR] AENOR (2007): Norma UNE 157801 "Criterios generales para la elaboración de proyectos de Sistemas de Información".
- [CompeticionesV10] (2009) "Competiciones deportivas versión 10" <http://www.sagois.com>
- [FBCV] [1998] Federación de Billar de la Comunidad Valenciana (1998): "Estatutos de la Federación".
- [Gutiérrez] Javier J. Gutiérrez, María J. Escalona, Manuel Mejías y Antonia M. Reina (2006) "Modelos de pruebas" <http://www.lsi.us.es/~javierj/publications/MDA14.pdf>.
- [IBM] IBM (2010) "Best practices for programming in C" http://www.ibm.com/developerworks/aix/library/au-hook_duttaC.html
- [ISO 9241-210:2010] (2009) "Ergonomics of human-system interaction" -- Part 210: http://www.usabilitynet.org/tools/r_international.htm#13407.
- [Lacalle] Alberto Lacalle (2009) "Importancia de una buena interfaz" <http://albertolacalle.com/hci/interfaz.htm>
- [Lorente] Diana Lorente Marí (2007) "Diseño de una aplicación informática para una empresa del sector de transporte marítimo de mercancías".
- [Marcelo] Marcelo, Julián (2004) Planificación de esfuerzos, costes y recursos. Apuntes de la asignatura. ETSII. UPV.
- [Métrica] MÉTRICA. VERSIÓN 3 Consejo Superior de Administración Electrónica. Ministerio de Administraciones Públicas. (2010) "Metodología de Planificación, Desarrollo y Mantenimiento de sistemas de información". www.csi.map.es/csi/metrica3/.
- [Montesa] Montesa Andrés, Jose Onofre (2009): "Planificación de proyectos informáticos". Apuntes.
- [Prácticas .NET] Microsoft (2010) "Patterns & Practices" [http://msdn.microsoft.com/es-es/practices/default\(en us\).aspx](http://msdn.microsoft.com/es-es/practices/default(en%20us).aspx).
- [Pressman] Pressman, Roger S. (2006). "Ingeniería del software: un enfoque práctico" McGraw-Hill.
- [Sánchez] Sánchez, Juan (2009): UML 2.0 Apuntes.
- [SO/IEC DTR 9126-4] (2001) "Software Engineering - Product quality - Part 4: Quality in use metrics" http://www.usabilitynet.org/tools/r_international.htm#9126-4.

¹³ En base a la norma ISO 690-2 (11-15-1997): <http://biblioteca.ucv.cl/herramientas/citaselectronicas/iso690-2/iso690-2.html>

Anexo I. Índice:

1.	Problemática general	3
2.	Estudio de características.....	3
2.1.	ACTA.....	4
2.1.1.	Introducción de acta.	4
2.1.2.	Modificación/Consulta/Baja de Actas.....	5
2.1.3.	Listado de Actas.....	5
2.1.4.	Mantenimiento Etiquetas de carácter general.	6
2.1.5.	Impresión de Etiquetas.	7
2.2.	LISTA DE FUERZA	7
2.2.1.	Introducción / Modificación de Listas de Fuerza.	8
2.2.2.	Consulta.....	8
2.2.3.	Listado.	8
2.3.	SOCIO.....	9
2.3.1.	Edición de socio.....	9
2.3.2.	Listado de socio	10
2.3.3.	Listado para Seguridad Social.....	11
2.3.4.	Licencias	11
2.3.5.	Clasificación.....	12
2.3.6.	Mejor Promedio	13
2.3.7.	Lista Subs.....	14
2.4.	CLUB	15
2.4.1.	Edición de Club.....	15
2.4.2.	Listado de Club	16
2.4.3.	Clasificación de club	16
2.5.	UTIL.....	17
2.5.1.	Reconstruir	18

2.5.2.	Configurar Listado	18
2.5.3.	Consulta Histórico	18
2.5.4.	Copia de Seguridad.....	19
2.5.5.	Cambiar Año	19
2.5.6.	Vaciar Año	19
2.5.7.	Finalizar	19
3.	Conclusión	20

Estudio preliminar del software de gestión actual

1. Problemática general

El programa BillGest es el software de gestión utilizado en la FBCV desde 1995 hasta la actualidad. Cumple la mayor parte de las exigencias de la FBCV, y ha demostrado robustez durante todo este tiempo. Pero actualmente existen diversos problemas surgidos principalmente por su antigüedad:

- Imposibilidad de exportar datos a formato compatible Microsoft® para su edición.
- Limitaciones en la cantidad de divisiones/grupos.
- Entorno poco amigable.
- Imposibilidad de ampliar o mantener el actual equipo hardware.
- Incompatibilidad con Microsoft® Windows Vista/Windows 7 (X86 / X64).
- Características desfasadas y no adaptadas a nuevas normativas (LOPD).
- Copias de seguridad inestables y poco eficientes
- Fichas con un formato de papel anticuado.
- Funcionalidades en desuso.
- No admite ejecuciones en un entorno de red.
- Imposibilidad de contactar con el servicio técnico (no existe la empresa desarrolladora en la actualidad).
- Poca o nula flexibilidad con drivers.
- Formato propietario cerrado de ficheros.

2. Estudio de características

Como parte de la recogida de información para la evaluación de requisitos se realiza un pequeño análisis del actual software, para averiguar que características se conservarán idénticas, cuales se ampliarán, cuales se eliminarán y según el caso si alguna reducirá su funcionalidad. Todo el documento se ha realizado con la supervisión del personal de administración que maneja habitualmente el programa.

Básicamente el programa se divide en cinco secciones con diferentes subsecciones:

1. ACTA	-Introducción de actas -Modificación, consulta y baja de actas -Listado acta -Mantenimiento Etiquetas de carácter general -Impresión Etiquetas
2. LISTA DE FUERZA	-Introducción/Modificación de Lista de Fuerza -Consulta -Listado
3. SOCIO	-Edición socios -Listado socios -Listado seguridad social -Impresión de licencias -Clasificación de jugadores -Mejor promedio -Lista socios SUB-15, SUB-17 y SUB-21
4. CLUB	-Alta/Modificación/Consulta/Baja club -Listado de clubs -Listado de clasificación de clubs
5. UTIL	-Reconstruir (Reíndice de BB.DD.)

	<ul style="list-style-type: none"> -Config. Listados y Etiquetas -Config. del Histórico de avisos -Copia seguridad Diskete -Cambiar año -Vaciar año -Finalizar
--	----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

2.1. ACTA

En esta sección se gestiona todo lo relacionado con las actas:



Figura 1. Menú de acta

2.1.1. Introducción de acta.

Introduccion de acta es la primera opción del programa, esto es así ya que es la opción que se utiliza más frecuentemente. Las actas una vez recibidas en formato impreso, en caso de estar bien cumplimentadas y firmadas, se introducen en el sistema.

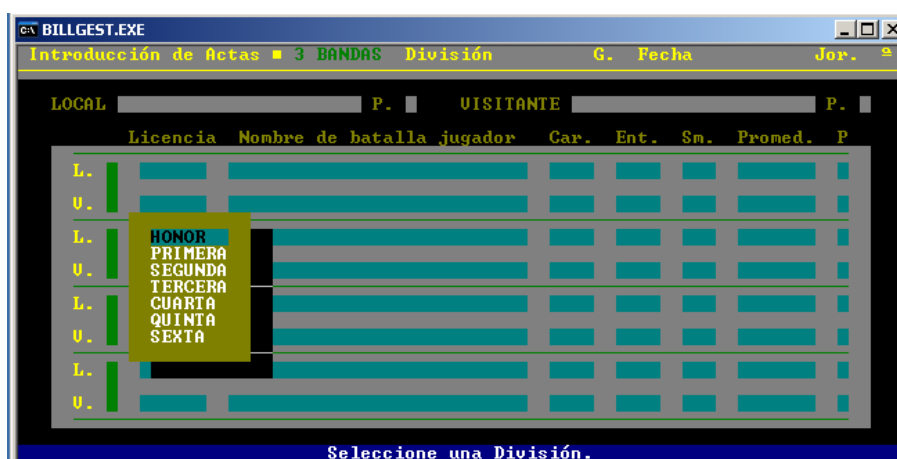


Figura 2. Introducción de actas

Campos de entrada	-Liga -División -Grupo -Fecha Acta -Jornada -Club Local -Club Visitante -Número Licencia Jugador -Carambolas -Entradas -SM (serie mayor). Usado para desempates.
Modificaciones	-Se debe introducir además el árbitro de cada encuentro -Se debe poder introducir el jugador por Licencia o por Nombre completo, con campos autocompletables

2.1.2.Modificación/Consulta/Baja de Actas.

El programa permite la consulta, la modificación y la baja de las actas introducidas en el sistema.

- La modificación gestiona los mismos datos que el Alta.
- La consulta permite buscar las actas para acceder a un listado concreto. Equivale al 1.3 -> SELECC. UNA.
- La baja elimina de la BB.DD. el acta.

Modificaciones: El funcionamiento debe ser similar, centralizando las búsquedas de los listados.

2.1.3.Listado de Actas.

En el listado de actas se puede consultar todas las cctas introducidas en el sistema. El primer criterio de búsqueda corresponde a las opciones:

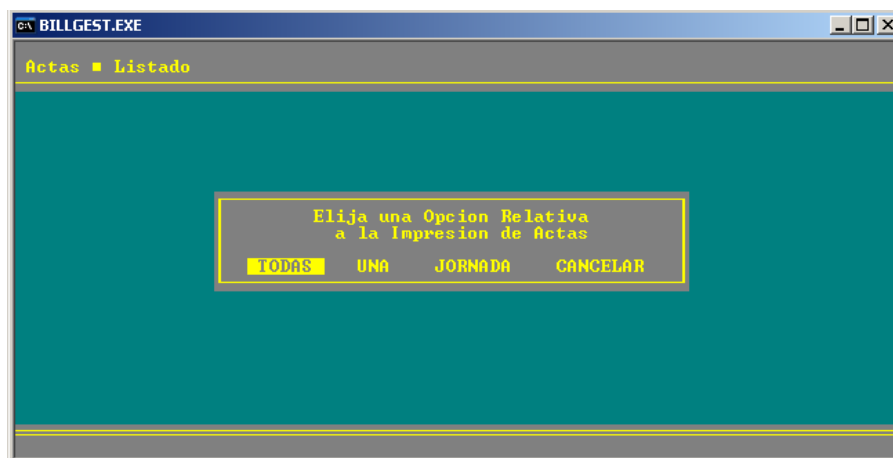


Figura 3. Listado de Actas

- TODAS : Se muestra un listado con todas las actas de ese año.
- UNA : Se selecciona un acta concreta introduciendo la Liga-> Division-> Grupo -> Club vis.
- JORNADA : Se muestra un listado con todas las actas de una jornada en concreto.

Modificaciones: El funcionamiento debe ser similar, centralizando las búsquedas de los listados.

2.1.4. Mantenimiento Etiquetas de carácter general.

Las etiquetas impresas actúan como carnets de socios. La sección 1.4 permite introducir, editar y borrar los datos a imprimir en las Etiquetas. Las Etiquetas actúan independientemente del socio, por lo que se necesita una gestión aparte.



Figura 4. Introducción de Etiquetas

Modificaciones: Las Etiquetas se imprimirán directamente desde el listado de socios, no habrá diferentes clasificaciones en los tipos de etiquetas, y los datos de los socios serán automáticamente los datos generados para las Etiquetas, por lo que desaparece el mantenimiento de estas y todo lo relacionado.

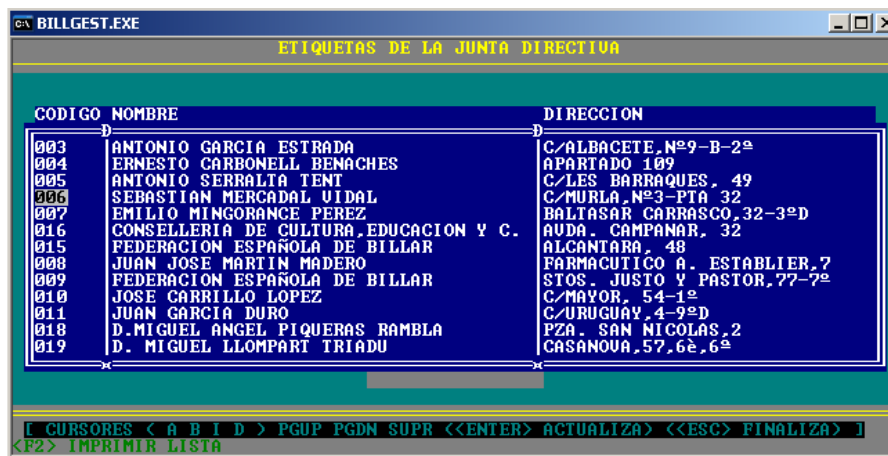


Figura 5. Edición de Etiquetas

2.1.5. Impresión de Etiquetas.

El programa permite la impresión de las Etiquetas introducidas en el apartado anterior. Además permite seleccionar cuales se quieren imprimir, seleccionarlas por diferentes clubes, y que cantidad hacer.



Figura 6. Impresión de Etiquetas

Modificaciones: La impresión se realizará directamente desde los reportes generados para los socios. El funcionamiento sera similar, pero se gestionará todo mediante el listado general de socios y la selección de las diferentes filas. El formato y las etiquetas físicas imprimibles se volverán a hacer con un nuevo diseño, por lo que no es necesario conservar el formato anterior.

2.2. LISTA DE FUERZA

En esta sección se gestiona todo lo relacionado con las Listas de Fuerza:



Figura 7. Menú de Lista de Fuerza

2.2.1.Introducción / Modificación de Listas de Fuerza.

El programa permite la impresión de las etiquetas introducidas en el apartado anterior. Además permite seleccionar cuáles se quieren imprimir, seleccionarlas por diferentes clubes, y que cantidad hacer.

Campos de entrada	-
Modificaciones	-

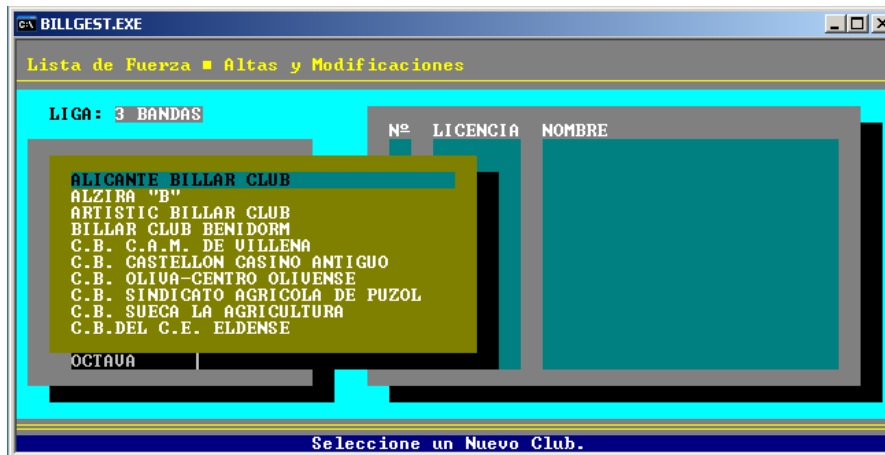


Figura 8. Modificación de Listas de Fuerza

2.2.2.Consulta.

La consulta de la Lista de Fuerza permite entrar en el menu de Edición sin opcion a editar. De esta manera se puede ver directamente la lista de un club.

Modificaciones: La opción quedará integrada en el futuro listado y no tiene sentido desarrollarla.

2.2.3.Listado.

El listado de la Lista de Fuerza permite mostrar todas las listas introducidas. El criterio de pre-selección es el mismo que en el apartado 1.3, TODAS (lista todas las actas), UNA (Selección de una por club, = 2.2) o SELECCIÓN (Selección de varias de un listado de todas).

Modificaciones: La opción quedará integrada en el listado

2.3. SOCIO

En esta sección se gestiona todo lo relacionado con la gestión de socios y sus listados:



Figura 9. Menú de socio

2.3.1. Edición de socio

El programa permite la gestión (alta – modificación – consulta- baja) de los socios de los diferentes clubes, así como de personal de la propia federación.

Para introducir un socio, es necesario primero introducir un código. El programa revisa que los tres primeros caracteres del código correspondan a un club. En caso de no serlo, da un mensaje de error.



Figura 8. Paso uno de edición de socio

Si el número es válido, y la licencia ya existe, muestra la plantilla de la siguiente pantalla con los campos rellenos listos para edición. En caso de no existir, la plantilla aparece con los campos vacíos lista para introducir un nuevo socio.

El campo búsqueda permite buscar un socio por un nombre concreto. Si existe, la licencia se rellenará automáticamente.

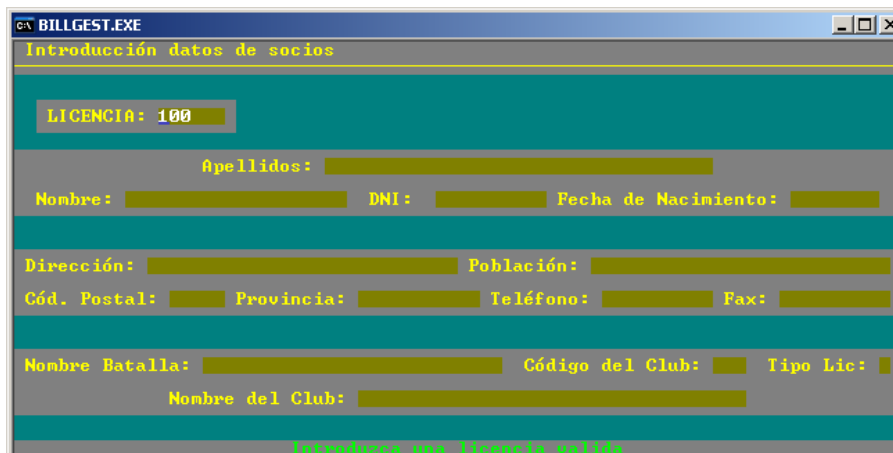


Figura 8. Edición de socio ventana principal

Los datos Nombre Batalla, Código de Club, Tipo de Licencia (JUNIOR/SENIOR) y Nombre de Club se completan automáticamente al introducir el número de licencia de 6 caracteres.

Campos de entrada	-Número de Licencia -Nombre -Apellidos -DNI -Fecha nacimiento -Dirección -Población -CP -Provincia -Teléfono -Fax
Modificaciones	-El código de Socio no tendrá correlación con el código de Club. La relación se hará por un campo en la tabla independiente - Se necesita incluir el campo Email y el campo Licencia en Vigor - Se elimina el campo Fax - Se intercambia la opción independiente de listas de fuerza por el orden de socio.

2.3.2. Listado de socio

Se selecciona el tipo de query para la extracción del listado y posteriormente se permite la impresión de los mismos.



Figura 9. Listado de socio

Modificaciones: La opción quedará integrada en el listado general de socios, no habrá distinción para las búsquedas.

2.3.3. Listado para Seguridad Social

El programa permite mostrar un listado de licencias por rango para las gestiones administrativas en la seguridad social.



Figura 10. Listado para la S.S.

Modificaciones: La opción quedará integrada en el listado general de socios, no habrá distinción para las búsquedas.

2.3.4. Licencias

Licencias permite imprimir los carnets de socios. Esta opción permite realizar una preselección antes de acceder al listado de impresión de etiquetas del apartado 1.4.

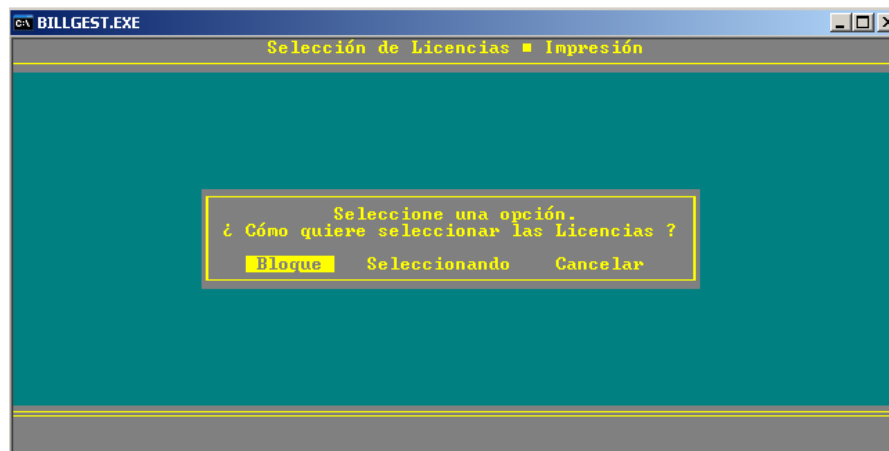
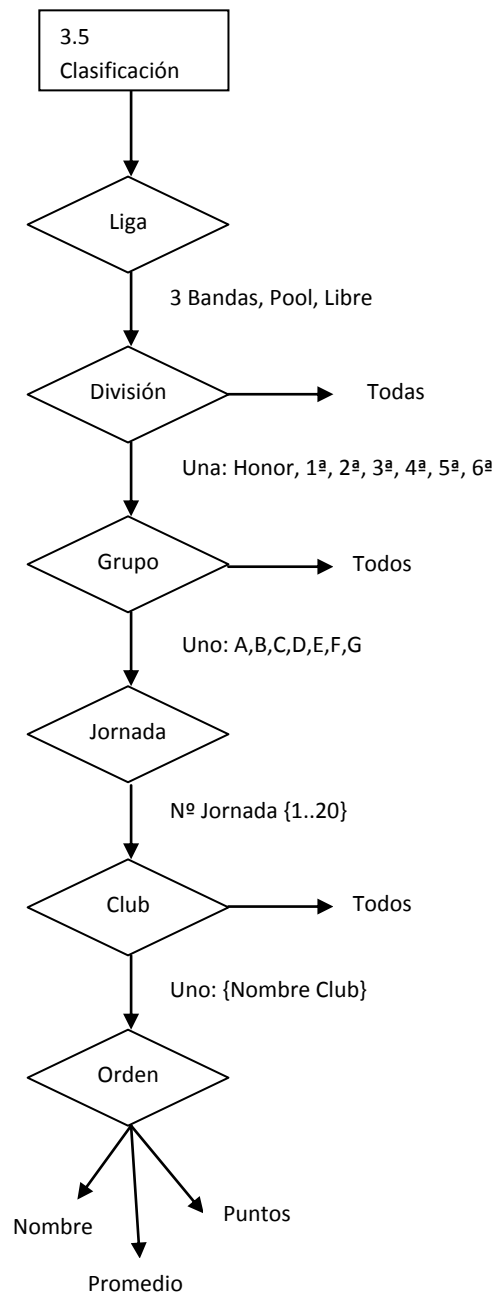


Figura 11. Selección de Licencias

Modificaciones: La opción quedará integrada en el detalle de Socios, se reconstruirá y la selección se realizará filtrando un listado general mediante lista desplegable.

2.3.5. Clasificación

Uno de los requisitos elementales del software es la generación de tablas de clasificación, tanto para jugadores como para clubs. En este apartado se pueden mostrar listados de clasificaciones en base a criterios seleccionados. La selección se realiza en base al siguiente diagrama de actividad¹:



¹ Diagrama de actividad: http://es.wikipedia.org/wiki/Diagrama_de_actividades



Figura 12. Clasificación de jugadores

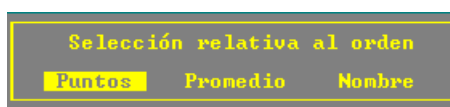


Figura 13. Selección de orden

Modificaciones: La Clasificación por Jugadores se filtrará de dos maneras. La primera sera por menú, existirán diferentes opciones del menú general para acceder a las clasificaciones de jugadores por tipo de Liga. El segundo filtro, que corresponderá a los pasos 2 y 3, se realizará mediante lista desplegable, la cual solo mostrará las opciones realmente existentes (si no existen grupos G, no se mostrará esa opcion en la lista). El resto de las selecciones se incluyen implícitamente en los campos del listado a mostrar, los cuales servirán para restringir la impresión de los determinados socios.

2.3.6. Mejor Promedio

El mejor promedio permite mostrar un listado de socios clasificados por promedios de mayor a menor. Mediante unos criterios se puede seleccionar que campos se quieren mostrar y el orden. Posteriormente estos promedios se imprimen y se distribuyen a los diferentes clubs.



Figura 14. Mejor promedio

Modificaciones: La opción de mejor promedio queda implícita en el listado general de clasificación mediante una columna, la cual puede reordenar la lista al ser pulsada, por lo que se evita esta función de negocio.

2.3.7. Lista Subs

Otra de las características de los listados de socio es la posibilidad de seleccionar socios por categoría Sub-15 (menor de 15 años antes de iniciar la temporada), Sub-17 y Sub-21.

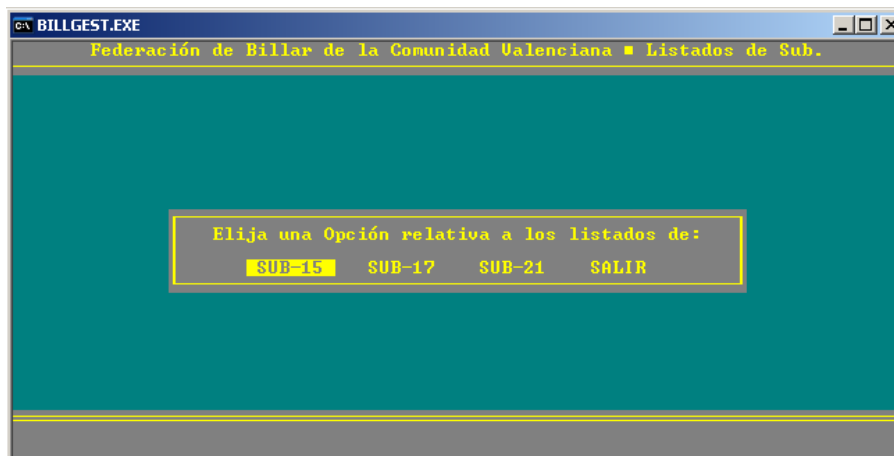


Figura 15. Listados Sub.

Modificaciones: Esta opción en la practica no se utiliza, por lo que queda descartada su implimentación (no obstante es posible reordenar los listados por fecha de nacimiento).

2.4. CLUB

En esta sección se gestiona todo lo relacionado con la gestión de clubes y sus listados:

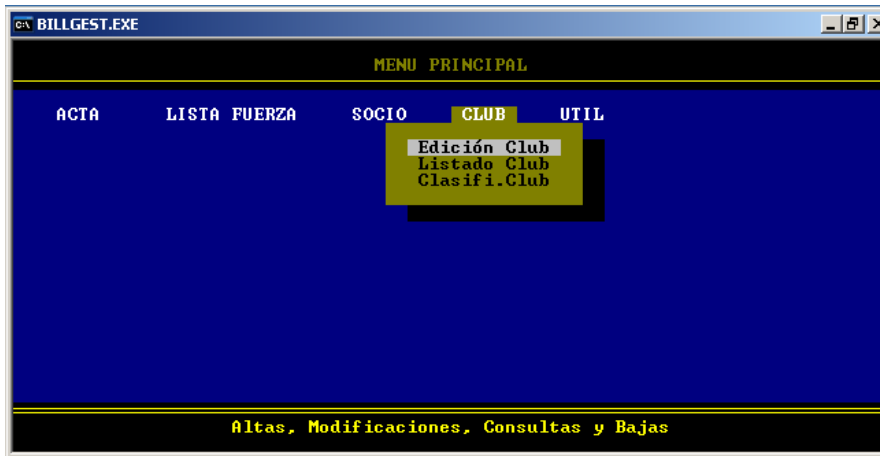


Figura 16. Menú de club

2.4.1. Edición de Club

Edición de Club permite las Altas, Modificaciones, Consultas y Bajas de los Clubes en la BB.DD.

En primera instancia se introduce un código de tres dígitos, en caso de existir el club, la plantilla de la siguiente pantalla aparece con los campos rellenos y es editable. En caso de no existir, la pantalla aparece con campos vacíos.

El campo búsqueda permite introducir un nombre de club y comprobar si existe o no.



Figura 17. Edición de club, primer paso

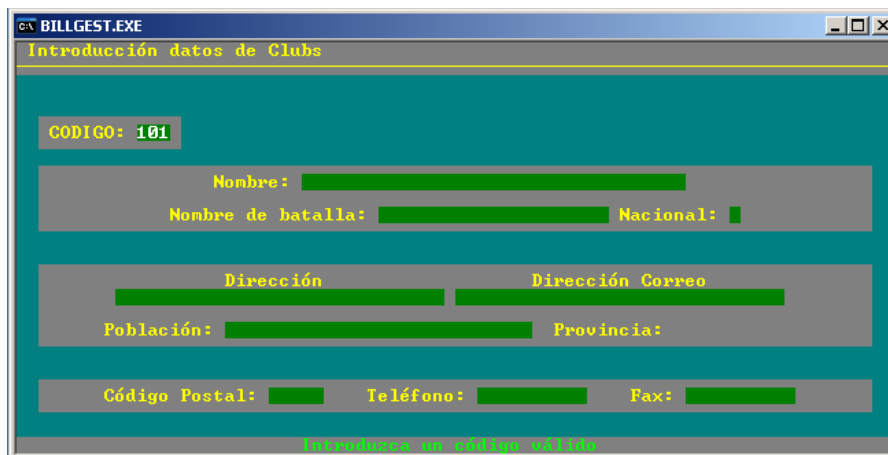


Figura 18. Edición de club ventana principal

Campos de	-Código
-----------	---------

entrada	-Nombre -Nombre de Batalla -Nacional -Dirección -Dirección de Correo -Población -Provincia -CP -Teléfono -Fax
Modificaciones	-Se necesita incluir el campo CIF, Web, Email - Se necesita diferenciar entre dirección jurídica y dirección de correspondencia. - Se elimina la opción Internacional

2.4.2.Listado de Club

EL Listado de clubes permite imprimir un listado con todos los clubes inscritos en la BB.DD. La opción imprime automáticamente el listado.

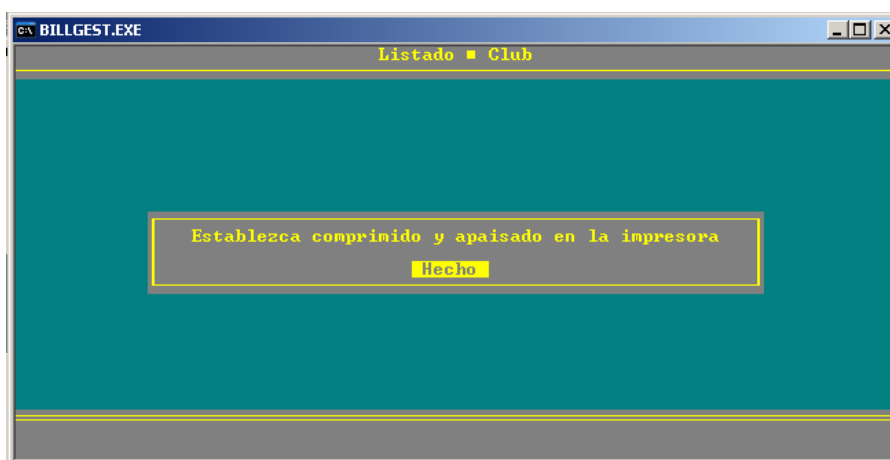


Figura 19. Listado de club

Modificaciones: Esta opción sera un clon de los listados de socios pero para clubes. El formato será completamente diferente.

2.4.3.Clasificación de club

La Clasificación de club es una opción pareja a la clasificación de socios pero el filtrado solamente contiene los pasos LIGA -> DIVISION -> GRUPO -> JORNADA. El sistema calcula los puntos de los clubes en base a las

actas introducidas y los muestra en un listado por clubes. Se seguirán los pasos de socios par realizar este listado.



Figura 20. Clasificación de club

2.5. UTIL

En esta sección se gestionan las diferentes utilidades, opciones y configuraciones del programa:



Figura 21. Menú de Util

2.5.1.Reconstruir

Reconstruir permite reindexar una BB.DD. que ha sufrido un cierre inesperado. Esto podría suceder por cortes repentinos de luz, cuelgues del programa o cuelgues propios del S.O., actualmente la tecnología de BB.DD. realiza estas gestiones automáticamente y no es necesario incluir esta función de negocio en el programa.

2.5.2.Configurar Listado

Mediante esta opción es posible configurar el formato para la impresión de los listados de toda la aplicación.



Figura 22. Configuración de listado

Modificaciones: Esta opción no se implementará.

2.5.3.Consulta Histórico

La consulta del histórico de avisos es la opción mediante la cual se muestran los avisos marcados en las actas de una determinada liga en una determinada temporada.

Modificaciones: Actualmente la opción no se utiliza, por lo que no será implementada.

2.5.4. Copia de Seguridad

Mediante esta opción podemos realizar el backup de los datos de clubs, socios y las actas de la temporada actual, y también recuperar estos datos. El sistema solicita que opción queremos elegir, y posteriormente exige entre uno y dos diskettes de formato 3½", ya sea para guardar o para restaurar la BB.DD. Es importante recalcar que en caso de restaurar una copia, no se podrá deshacer el cambio y se perderían los datos actuales.



Figura 23. Copia de seguridad introducción de Diskettes



Figura 24. Copia de seguridad aviso

Modificaciones: Esta opción deberá ser implementada, adecuando las características a las nuevas funcionalidades otorgadas por Windows[®]. Existirá la opción de guardado de datos y recuperación con un formato específico de fichero, que restaurará las copias internas del programa. En principio la opción de deshacer no se implementará.

2.5.5. Cambiar Año

Cambiar Año permite abrir y cargar cualquier temporada guardada en una copia de seguridad.

2.5.6. Vaciar Año

Vaciar año borra los datos de las actas de una temporada concreta. Esto afecta por tanto a los listados de clasificaciones que pasan a tomar valor cero. Hay que tener en cuenta que ni los datos de socio ni los de club quedan afectados.

2.5.7. Finalizar

Sale del programa al entorno MS.DOS.

3. Conclusión

Tras el análisis del programa se tiene una primera toma de requisitos que será complementada con formularios y entrevistas con el personal de administración. Básicamente el nuevo programa debe ofrecer los servicios básicos mostrados aquí, eliminar los servicios que han caído en desuso, y proporcionar nuevas herramientas que faciliten la gestión al personal.

Por otra parte, para el tratamiento de errores se intentará en la medida de lo posible proporcionar recuperabilidad, ya que el software actual, a pesar de ser bastante robusto, con cada fallo no controlado sale inmediatamente al entorno deshaciendo todos los cambios.

Anexo II -1. Índice:

1.	Inicio del Plan de Sistemas de Información.....	3
1.1.	Análisis de la necesidad del PSI	3
1.2.	Identificación del alcance del PSI	4
1.3.	Determinación de responsables.....	5
2.	Definición y Organización del PSI	6
2.1.	Especificación del ámbito y alcance	6
2.2.	Organización del PSI	9
2.3.	Definición del plan de trabajo	10
2.4.	Comunicación del plan de trabajo	11
3.	Estudio de la información relevante	11
3.1.	Selección y análisis de antecedentes	11
3.2.	Valoración de antecedentes.....	12
4.	Identificación de requisitos	12
4.1.	Estudio de los procesos del PSI	12
4.2.	Análisis de las necesidades de información	13
5.	Estudio de los sistemas de información actuales.....	14
5.1.	Alcance y objetivos del estudio de los sistemas de información actuales.....	14
5.2.	Análisis de los sistemas de información actuales	15
6.	Diseño del modelo de sistemas de información	15
6.1.	Diagnostico de la situación actual	15
6.2.	Definición del modelo de sistemas de información	16
7.	Definición de la arquitectura tecnológica	18
7.1.	Identificación de las necesidades de infraestructura tecnológica	18
7.2.	Selección de la arquitectura tecnológica	18
8.	Definición del plan de acción	20
8.1.	Definición de proyectos a realizar	20
8.2.	Elaboración del plan de mantenimiento	22
9.	Revisión y aprobación del PSI.....	22
9.1.	Aprobación del PSI	23

• Plan de Sistemas de Información (PSI)

El Plan de Sistemas de Información tiene como objetivo la obtención de un marco de referencia para el desarrollo de sistemas de información que responda a los objetivos estratégicos de la organización. Este marco de referencia consta de:

- “- Una descripción de la situación actual, que constituirá el punto de partida del Plan de Sistemas de Información. Dicha descripción incluirá un análisis técnico de puntos fuertes y riesgos, así como el análisis de servicio a los objetivos de la organización.
- Un conjunto de modelos que constituya la arquitectura de información.
- Una propuesta de proyectos a desarrollar en los próximos años, así como la prioridad de realización de cada proyecto.
- Una propuesta de calendario para la ejecución de dichos proyectos.
- La evaluación de los recursos necesarios para los proyectos a desarrollar en el próximo año, con el objetivo de tenerlos en cuenta en los presupuestos. Para el resto de proyectos, bastará con una estimación de alto nivel.
- Un plan de seguimiento y cumplimiento de todo lo propuesto mediante unos mecanismos de evaluación adecuados.”

Es fundamental que la alta dirección de la organización tome parte activa en la decisión del Plan de Sistemas de Información con el fin de posibilitar su éxito. La dirección debe convencer a sus colaboradores más directos de la necesidad de realización del plan; de su apoyo de forma constructiva, mentalizándose de que la ejecución del mismo requerirá la utilización de unos recursos de los cuales son responsables.

La presentación del Plan de Sistemas de Información y la constitución del equipo supone el arranque del proyecto y es fundamental que las más altas instancias de la organización estén implicadas en ambos, dando el apoyo necesario y aportando todo tipo de medios. Explicar el plan a las personas de la organización y a las unidades organizativas afectadas sobre las que recaerá el Plan, el apoyo de los altos directivos y la cualificación de los recursos de las distintas unidades implicadas, serán factores críticos de éxito del Plan de Sistemas de Información.

El nivel de detalle con el que se hará el estudio de la situación actual dependerá de la existencia de documentación actual, de si hay personas que conozcan dicha documentación y de la predisposición a una sustitución total o parcial por sistemas de información nuevos. En cualquier caso, como paso previo para detectar aspectos importantes que puedan afectar a la organización, es necesario investigar sus puntos fuertes, áreas de mejora, riesgos y amenazas posibles y hacer un diagnóstico de los mismos.

Para la elaboración del Plan de Sistemas de Información se estudian las necesidades de información de los procesos de la organización afectados por el Plan, con el fin de definir los requisitos generales y obtener modelos conceptuales de información. Por otra parte se evalúan las opciones tecnológicas y se propone un entorno.

Tras analizar las prioridades relacionadas con las distintas variables que afectan a los sistemas de información, se elabora un calendario de proyectos con una planificación lo más detallada posible de los más inmediatos. Además, se propone una sistemática para mantener

actualizado el Plan de Sistemas de Información para incluir en él todos los cambios necesarios, garantizando el cumplimiento adecuado del mismo.

1. Inicio del Plan de Sistemas de Información

“El objetivo de esta actividad es determinar la necesidad del Plan de Sistemas de Información y llevar a cabo el arranque formal del mismo, con el apoyo del nivel más alto de la organización. Como resultado, se obtiene una descripción general del Plan de Sistemas de Información que proporciona una definición inicial del mismo, identificando los objetivos estratégicos a los que apoya, así como el ámbito general de la organización al que afecta, lo que permite implicar a las direcciones de las áreas afectadas por el Plan de Sistemas de Información.”

Además, se identifican los factores críticos de éxito y los participantes en el Plan de Sistemas de Información, nombrando a los máximos responsables.

Tareas propuestas para esta actividad:

- 1.1. Análisis de la necesidad del PSI
- 1.2. Identificación del alcance del PSI
- 1.3. Determinación de responsables

1.1. Análisis de la necesidad del PSI

Objetivo:

“Se analizan las expectativas de las áreas que han planteado la necesidad de llevar a cabo el Plan de Sistemas de Información, así como los productos finales esperados. Una vez verificado que las necesidades de la organización se deben cubrir con un Plan de Sistemas de Información, se toma la decisión de su inicio.”

Aprobación del inicio del PSI:

El departamento de administración y su responsable el secretario general de la organización ha decidido impulsar la puesta en marcha de un nuevo Plan de Sistemas de Información de actualización y apoyo a los procesos existentes en la organización.

Esta decisión cuenta con el apoyo explícito del presidente de la Federación, y se basa en la necesidad de modernizar los procesos de la organización, dotándolos de la tecnología que permita un adecuado, coherente y preciso tratamiento de una información cada vez más voluminosa y crítica. Este apoyo de la Dirección se considera especialmente necesario, máxime teniendo en cuenta la novedad que un proyecto de este tipo representa para la empresa, al no existir precedentes en cuanto a Planes de Sistemas de Información o estudios similares.

1.2. Identificación del alcance del PSI

Objetivo:

“Se define el ámbito del Plan de Sistemas de Información en términos de procesos de la organización afectados y, como consecuencia, las direcciones de las áreas implicadas. Se determinan los objetivos estratégicos de la organización que deben ser considerados en el Plan de Sistemas de Información, así como aquellos aspectos que la dirección considera factores críticos de éxito para el mismo.”

Ámbito y Objetivos del PSI:

El Comité de Dirección está formado por el Presidente, Vicepresidente 1º, Vicepresidente 2º y los Directores de todos los Departamentos. Tras una reunión se decide que el plan de Sistemas recaiga sobre los procesos manejados por el departamento de administración en exclusiva, por tanto el resto de departamentos no variarán la definición de su sistema de información, cierto es que en cierta medida el impacto en administración será patente en otros departamentos, indirectamente.

Los objetivos estratégicos y sus factores críticos de éxito a conseguir con la elaboración del plan de sistemas de información son:

Eliminar carga del departamento de Administración	Rapidez y facilidad de uso de la actualización del paquete de gestión.
Agilidad en la toma de decisiones	Conexión directa del sistema de información con los respectivos responsables de los departamentos
Aumento del número de clubes asociados	Dotar de más servicios a la federación, publicitar mas la actividad y establecer una política de competencias
Reducción del tiempo en la publicación de resultados	Automatización del sistema para las actualizaciones de resultados y su difusión.
Dotar de mayores facilidades al acceso de información para clubes y socios	Publicar la información deseada con la mayor rapidez y que esta información pueda ser dispuesta por cualquiera de sus clubes y asociados.
Remodelación de la web y conexión directa y automática con el software de gestión	Auto actualización de la web corporativa en base a los datos introducidos en el sistema de información
Facilitar el mantenimiento de los sistemas	Facilidad para la compra y sustitución del material hardware y software.
Gran capacidad de almacenamiento de información y seguridad a todos los niveles	El volumen de información y su seguridad no debe ser un factor limitante para la consecución del plan de sistemas.

Para ello se ha puesto en marcha un Plan Operativo cuyos primeros objetivos consisten en la compra de hardware informático que posibilite la adopción del nuevo paquete software.

Asimismo es necesario el apoyo firme de Dirección en todo el proyecto, y es necesario que el volumen de trabajo crezca de forma controlada, a fin de evitar que un aumento de trabajo obligue a los responsables de los Departamentos a reducir la dedicación exigida por las tareas del Plan encomendadas.

1.3. Determinación de responsables

Objetivo:

“Delimitado el ámbito del Plan de Sistemas de Información, se implica a las unidades organizativas afectadas, informándoles de la decisión y solicitando su participación en el estudio que se va a iniciar. En sesiones de trabajo con las distintas unidades se determinan los principales responsables del Plan de Sistemas de Información a los que seguidamente se les debe comunicar su nombramiento y solicitar su aceptación.”

Las personas seleccionadas serán los participantes en la Dirección del Plan de Sistemas de Información. También se determina la necesidad de apoyo en la función de seguimiento que determine el Plan de Sistemas de Información. Dicha necesidad depende de la amplitud del Plan de Sistemas de Información y de la duración prevista para el mismo. Si se considera necesario, en esta actividad se proponen los responsables de dicho seguimiento.

Responsables del PSI:

La responsabilidad en primera instancia del Plan de Sistemas de Información recae sobre el departamento de administración, que es el principal objetivo en la reestructuración del plan. Asimismo se decide nombrar responsables en caso de que se necesite colaboración a los directores de cada departamento.

- Director del Plan de Sistemas de Información: Sra. Dña. Cristina González (Secretario General).
- Representantes de las áreas:

PRESIDENCIA:

D. Antonio Ortiz Martínez

ADMINISTRACION:

Dña. Cristina González

TESORERIA/CONTABILIDAD

Dña. Cristina González

DIRECCION DEPORTIVA:

D. Alejandro Gil Martínez

COMITE JUECES-ARBITR:

D. Angel Gutiérrez Alcaraz

COMITE DE COMPETICION:

D. Angel Gutiérrez Alcaraz

COMITE DE APELACION:

D. Julio Martínez Cusí

DELEGADO ALICANTE:

D. Luis M. González de la Vega

DELEGADO CASTELLON:

D. Jesús Gallen Naches

DELEGADO DE POOL:

D. Antonio Morales Fernández

2. Definición y Organización del PSI

En esta actividad se detalla el alcance del plan, se organiza el equipo de personas que lo va a llevar a cabo y se elabora un calendario de ejecución. Todos los resultados o productos de esta actividad constituirán el marco de actuación del proyecto más detallado que en PSI 1 en cuanto a objetivos, procesos afectados, participantes, resultados y fechas de entrega.

Tareas propuestas para esta actividad:

- 2.1. Especificación y ámbito de alcance
- 2.2. Organización del PSI
- 2.3. Definición del plan de trabajo
- 2.4. Comunicación del plan de trabajo

2.1. Especificación del ámbito y alcance

Objetivo:

“De manera más concreta que en la actividad Inicio del Plan de Sistemas de Información (PSI 1), en esta tarea se describe el ámbito de los procesos de la organización a considerar. Igualmente, se definirá el alcance, es decir, los objetivos específicos del Plan de Sistemas de Información. Puede ser necesario determinar distintos objetivos para cada proceso del proyecto. Los responsables de los distintos procesos de la organización afectados por el Plan de Sistemas de Información participarán de forma activa en la definición de los objetivos, sin perder de vista los resultados de la actividad anterior.”

Descripción general de procesos afectados y catalogo de objetivos:

Se especifican los objetivos a alcanzar por cada departamento:

1) Administración

Proceso: La Sra. González, introduce los clubes y socios que formalizan su alta con el formulario pertinente a través de las delegaciones provinciales o directamente en un programa no estándar con base de datos propia y que corre bajo MS-DOS, llamado BILLGEST.

Objetivos: Agilizar el trabajo, actualizar formatos y disponer de una BB.DD. de formato Microsoft® Office editable con posibilidades de exportación, además se debe mantener una BB.DD. común que permita mejorar la coherencia en la información.

Proceso: El personal de administración imprime los listados semanalmente desde BILLGEST y los envía por correo ordinario a las direcciones físicas incluidas en la BB.DD. de BILLGEST.

Objetivos: Simplificar el trabajo del personal automatizando el proceso de envío y cambiar a largo plazo el envío en papel por formato electrónico.

Proceso: El personal de administración imprime los carnets de Socio anualmente una vez todos los socios están formalizados por sus respectivos clubes y son recibidos los impresos de inscripción.

Objetivos: El diseño completo del carnet será impreso automáticamente al mismo tiempo que los datos para evitar tener que disponer de carnets vírgenes preestablecidos, asimismo se debe conseguir agilizar el envío de carnets a los clubes.

Proceso: La Sra. González cierra la temporada una vez todas las ligas y partidos han finalizado y guarda una copia de seguridad en diskettes de 3 ½ de la misma.

Objetivos: Poder realizar backups en cualquier medio físico y agilizar la conclusión de temporadas con guardados automáticos.

Proceso: El personal de administración manda a los Socios información sobre nuevos eventos, torneos y descuentos ofrecidos por la FBCV.

Objetivos: Aumentar la capacidad de envío de publicidad en formato electrónico de la FBCV con envíos masivos pero especializados con información que considere oportuna el secretario general.

Proceso: Administración recibe las actas de las temporadas firmadas por los árbitros que la han disputado con un formulario estándar impreso.

Objetivos: Conseguir a largo plazo el uso de firma electrónica digital para que los propios árbitros puedan firmar sus actas y enviarlas en formato electrónico al sistema de información de la FBCV.

Proceso: Administración recibe una solicitud por parte de un socio adscrito a un club para recibir personalmente por email las tablas clasificatorias y promedios.

Objetivos: El Socio debe poder tramitar electrónicamente sus solicitudes y en base a sus peticiones debe remitírsele toda la información requerida.

Proceso: Administración recibe un formulario de impugnación por parte del comité de apelación.

Objetivos: De la misma manera el Comité de Apelación debería, usando su intranet, introducir en el sistema las impugnaciones que lleven a cabo los clubes.

Proceso: La Sra. González envía al comité de dirección deportiva la lista con los clubes y socios adscritos a principio de temporada, con su situación regularizada.

Objetivos: El software de gestión debe filtrar las altas con fallos y defectos de forma, asimismo una vez comprobadas y validadas por el personal de administración, estas deben enviarse en formato electrónico a Dirección Deportiva para su alta final.

Proceso: Administración recibe el acta de temporada con los cambios si los hubiere respecto a otros años, ligas a disputar, calendario de competición, etc. etc.

Objetivos: La información debe introducirse en el sistema de una manera rápida y fácil, con formatos preestablecidos y con preferencias por usuario.

Proceso: Los ordenadores de administración están interconectados mediante un cable de par trenzado, 2 tarjetas de red, y no comparten impresoras.

Objetivos: La conexión de los ordenadores a internet debe ser totalmente independiente, además las impresoras deben estar compartidas.

2) Comité de Apelación

Proceso: El Comité de Apelación recibe un recurso por parte de un Club mediante una carta dirigida al director del comité por asuntos relacionados con la competición. El comité aprueba o desestima el recurso e informa al remitente y en caso de modificar alguna acta también se lo comunica al personal de Administración.

Objetivos: Las apelaciones deben tramitarse de manera electrónica, introduciendo los cambios si los hubiese directamente en el sistema evitando el trabajo de administración.

3) Delegaciones

Proceso: La delegación provincial recibe una solicitud de diversa índole por parte de un club.

Objetivos: Las delegaciones provinciales deben estar interconectadas con la Federación central por medio de una red privada virtual corporativa. Todos los procesos deben centralizarse en un mismo sistema de información.

4) Presidencia

Proceso: El presidente o sus vicepresidentes aprueban o rechazan una propuesta.

Objetivos: Las propuestas, además de firmarse en un acuerdo escrito e introducidas en los estatutos, deben ser incluidas en el sistema de información de la FBCV.

5) Tesorería

Proceso: Operaciones con los flujos contables y libros de cuentas

Objetivos: Se debe eliminar todas las operaciones en papel y habilitar un software de gestión que permita llevar una contabilidad electrónica de manera segura. Las aprobaciones y denegaciones por parte de Tesorería también deben ser comunicadas de manera electrónica a los departamentos afectados.

6) Dirección deportiva

Proceso: Se decide la modificación de alguna norma establecida en los estatutos.

Objetivos: Toda modificación debe introducirse en el sistema de información y por tanto modificar automáticamente las opciones de las actas clubes y ligas del software de gestión de ligas sin necesidad de personal externo de administración.

2.2. Organización del PSI

Objetivo:

“En esta tarea se tratan cuestiones relacionadas con la organización del trabajo para llevar a cabo el Plan de Sistemas de Información. Se seleccionan los participantes, valorando el número y perfil de profesionales de Sistemas y Tecnologías de la Información y Comunicaciones (STIC) necesarios en función de los objetivos perseguidos. Asimismo, se determinan las funciones de los responsables de la dirección y seguimiento del Plan de Sistemas de Información. Adicionalmente, se concretan aspectos logísticos relacionados con el material, salas de reuniones, estándares de documentación, etc.”

Equipo de trabajo:

El equipo de trabajo que realizara el Plan de Sistemas de Información corresponde a:

1) Jefe de Proyecto de PSI

Nombre: Sra. Cristina González.

Cargo: Secretario General

Conocimientos Tecnologías de la Información: Medio

Funciones:

- Coordinar el Plan de Sistemas de Información.
- Participar en las tareas que le sean encomendadas en el Plan.
- Crear el calendario de seguimiento del PSI.
- Asegurar la consecución de objetivos y tareas del Plan.
- Participar en todas las reuniones relacionadas con el PSI.
- Detectar necesidades de formación de los usuarios participantes.
- Decidir sobre las posibles modificaciones técnicas del proyecto.
- Aprobar los resultados de la realización del PSI.
- Introducción de los datos correspondientes en el sistema.

2) Responsables Funcionales

Son los respectivos Jefes de los departamentos.

Conocimientos T.I.: Bajo
 Funciones:

- Participar en las tareas que le sean encomendadas en el PSI.
- Participar en todas las reuniones relacionadas con el PSI de su departamento.
- Coordinar al personal de su departamento.

3) Consultores Externos

Equipo técnico que elaborará los productos a entregar.
 Conocimientos T.I.: Alto

Funciones:

- Captación de información in situ o diferida mediante entrevistas, formularios e impresos con el personal requerido.
- Evaluación de la información recogida.
- Elaboración de informes y productos especificados en el PSI.

2.3. Definición del plan de trabajo

Objetivo:

“El objetivo de esta tarea es determinar todos los productos finales del Plan de Sistemas de Información, así como la fecha prevista de obtención y entrega de los mismos. Es necesario planificar las distintas actividades y estimar los tiempos requeridos para llevarlas a cabo, teniendo en cuenta la disponibilidad de los usuarios del Plan de Sistemas de Información. Se deben considerar también los factores críticos de éxito, identificados en la actividad anterior y recogidos en la descripción general de procesos de la organización afectados, ya que pueden condicionar la elaboración del plan de trabajo.”

Se detallan las actividades, asignando participantes, tiempos y responsables de cada una de ellas, los resultados esperados y el plan de trabajo a seguir.

Plan de trabajo:

El plan de trabajo y el calendario detallado para la realización del PSI se especifica a continuación:

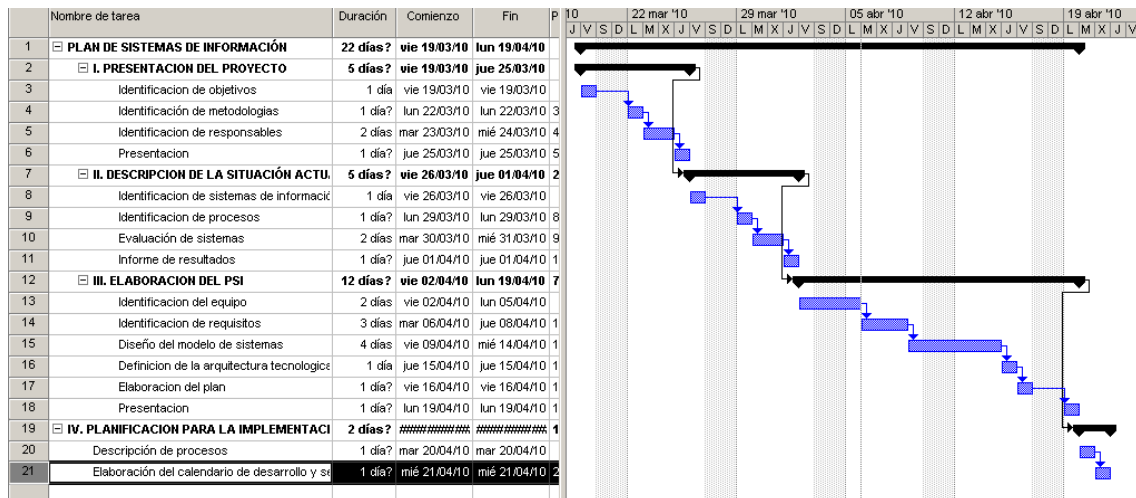


Figura 2.1. Plan de trabajo

2.4. Comunicación del plan de trabajo

Objetivo:

“Una vez definido el plan de trabajo se comunica a los usuarios del Plan de Sistemas de Información con el fin de que sea aceptado. Esto permite que los usuarios conozcan el método de trabajo a seguir, los resultados a obtener y la dedicación necesaria por su parte.”

Aceptación del plan de trabajo:

El Plan de trabajo es comunicado a los usuarios del Plan de Sistemas de Información a fin de darlo a conocer y obtener su aceptación.

3. Estudio de la información relevante

El objetivo de esta actividad es recopilar y analizar todos los antecedentes generales que puedan afectar a los procesos y a las unidades organizativas implicadas en el Plan de Sistemas de Información, así como a los resultados del mismo. Pueden ser de especial interés los estudios realizados con anterioridad al Plan de Sistemas de Información, relativos a los sistemas de información de su ámbito, o bien a su entorno tecnológico, cuyas conclusiones deben ser conocidas por el equipo de trabajo del Plan de Sistemas de Información.

Tareas propuestas para esta actividad:

- 3.1. Selección y análisis de antecedentes
- 3.2. Valoración de antecedentes

3.1. Selección y análisis de antecedentes

Objetivo:

“Se seleccionan las fuentes de información y documentación a considerar en este estudio, teniendo en cuenta todos aquellos antecedentes de interés: plan estratégico de sistemas de información, estudios previos, plan general informático, etc. y se analiza el contenido de la información anterior. En el inicio y organización del Plan de Sistemas de Información se habrá orientado sobre la existencia de estos antecedentes, para facilitar al equipo de trabajo el desarrollo de esta actividad.

Asimismo, se debe entrevistar a las personas de la organización que puedan aportar información adicional sobre antecedentes que deban ser considerados en el Plan de Sistemas de Información, al margen de la documentación disponible. La información recogida se tiene también en cuenta en la valoración de los mismos.”

Análisis de antecedentes:

Una vez analizadas las unidades organizativas se entrevista a la Sra. Cristina González, responsable de Administración y Secretario General de la FBCV, que es la persona responsable del sistema de información actual de la Federación.

De la entrevista se concluye que no existe ningún tipo de estudio previo sobre Sistemas de Información, y que la informatización de la organización se produjo de forma no planificada e independiente para cada tarea, siempre con ayuda exterior.

3.2. Valoración de antecedentes

Objetivo:

“Se realiza la valoración de los antecedentes analizados en la tarea anterior y las conclusiones se recogerán en el catálogo de requisitos. La realización de esta valoración ayudará a establecer términos de referencia en cuanto a estándares, procedimientos, normativas, etc., si es que existen.”

Antecedentes:

Como se ha especificado anteriormente no existe ningún estudio previo, por lo que el Plan de Sistemas de Información ha de empezar desde cero.

4. Identificación de requisitos

El objetivo final de esta actividad va a ser la especificación de los requisitos de información de la organización, así como obtener un modelo de información que los complemente.

Para conseguir este objetivo, se estudia el proceso o procesos de la organización incluidos en el ámbito del Plan de Sistemas de Información. Para ello es necesario llevar a cabo sesiones de trabajo con los usuarios, analizando cada proceso tal y como debería ser, y no según su situación actual, ya que ésta puede estar condicionada por los sistemas de información existentes.

Del mismo modo, se identifican los requisitos de información, y se elabora un modelo de información que represente las distintas entidades implicadas en el proceso, así como las relaciones entre ellas. Por último, se clasifican los requisitos identificados según su prioridad, con el objetivo de incorporarlos al catálogo de requisitos del Plan de Sistemas de Información.

Tareas propuestas para esta actividad:

- 4.1. Estudio de los procesos del Plan de Sistemas de Información
- 4.2. Análisis de las necesidades de información

4.1. Estudio de los procesos del PSI

Objetivo:

“Se estudia cada proceso de la organización incluido en el ámbito del Plan de Sistemas de Información. Para cada uno de ellos, es necesario identificar las actividades o funciones, la información implicada en ellas y las unidades organizativas que participan en el desarrollo de cada actividad.

Para obtener esta información es necesario llevar a cabo sesiones de trabajo con los usuarios implicados en cada uno de los procesos a analizar. Una vez contrastadas las conclusiones, se elabora el modelo correspondiente a cada proceso. Si existe relación entre los distintos modelos, se unifican en la medida de lo posible, con el fin de proporcionar una visión global en el contexto de la organización y facilitar una identificación de requisitos más objetiva.”

Modelo de procesos de la organización:

El modelo de procesos que se desarrollara contemplara solo la herramienta de gestión de ligas socios y clubes de la FBCV, dejando al margen otro tipo de herramientas contables o de mejora de procesos de otras unidades organizativas.

Después de llevar a cabo reuniones y recopilación de información se acuerda el estudio de los procesos dentro de administración:

1. **Gestión de Temporadas:** correspondiente a la gestión de torneos.
2. **Gestión de Socios:** permite gestionar la operativa respecto a socios y personal.
3. **Gestión de Clubes:** permite gestionar la operativa respecto a clubes adheridos.

4.2. Análisis de las necesidades de información

Objetivo:

“Mediante sesiones de trabajo, se identifican las necesidades de información de cada uno de los procesos analizados en la actividad anterior. Se elabora un modelo de información que refleje las principales entidades y relaciones existentes entre ellas. Todo esto se realiza con la perspectiva de lo que debe ser el proceso en cuanto a sus actividades y funciones, así como a la información de entrada y salida para cada una de ellas.”

Los resultados del análisis realizado en esta tarea son la base para la identificación de requisitos.

Necesidades de información:

Requisitos funcionales correspondientes a:

Gestión de Temporadas:

- Apertura y cierre de temporadas.
- Copias de seguridad
- Especificación de número de grupos y divisiones por club.
- Listado general o por promedios
- Impresión de listados
- Información relativa a los clubes enfrentados y socios que las disputan
- Información respecto a las puntuaciones obtenidas
- Número limitado preestablecido de jornadas en las que se introducirán actas.
- Sanciones y árbitros

Gestión de Socios:

- Datos personales
- Socios junior o sénior y sus diferentes tarifas
- Listas de Fuerza
- Bajas temporales, bajas definitivas
- Diseño de carnets
- Impresión de carnets
-

Gestión de Clubes:

- Solicitudes a la FBCV
- Datos de los clubes como clubes deportivos

5. Estudio de los sistemas de información actuales

El objetivo de esta actividad es obtener una valoración de la situación actual al margen de los requisitos del catálogo, apoyándose en criterios relativos a facilidad de mantenimiento, documentación, flexibilidad, facilidad de uso, etc. En esta actividad se debe tener en cuenta la opinión de los usuarios, ya que aportarán elementos de valoración, como por ejemplo, su nivel de satisfacción con cada sistema de información.

Se seleccionan los sistemas de información actuales que son objeto del análisis y se lleva a cabo el estudio de los mismos con la profundidad y el detalle que se determine conveniente en función de los objetivos definidos para el Plan de Sistemas de Información. Este estudio permite, para cada sistema, determinar sus carencias y valorarlos. Esta valoración se utilizará en la actividad Diseño del Modelo de Sistemas de Información (PSI 6), donde se analizará la cobertura de los sistemas de información actuales con respecto a los requisitos.

Tareas propuestas para esta actividad:

- 5.1. Alcance y objetivos del estudio de los sistemas de información actuales
- 5.2. Análisis de los sistemas de información actuales

5.1. Alcance y objetivos del estudio de los sistemas de información actuales

Objetivo:

“A partir de la descripción general de los procesos de la organización afectados por el Plan de Sistemas de Información se determina qué sistemas de información actuales se encuentran dentro del ámbito del Plan de Sistemas de Información. Se seleccionan, de los sistemas existentes, los que deben ser analizados, así como los objetivos del estudio de cada uno. De esta forma, se establece el dominio de sistemas de información de la organización a considerar. También se tienen en cuenta los objetivos definidos para el Plan de Sistemas de Información, en función de los cuales se establece la amplitud y profundidad con la que se deberá desarrollar esta actividad.”

Los objetivos del Plan de Sistemas de Información se completan con los objetivos definidos en esta tarea para el estudio de los sistemas de información actuales. La información relativa a los sistemas de información que dan actualmente soporte a los procesos afectados por el Plan de Sistemas de Información, se obtiene mediante sesiones de trabajo con los usuarios y el apoyo del personal informático que se considere necesario.

Sistemas de Información que afectados en el PSI:

Actualmente el único sistema de información que entra dentro del PSI son los procesos relacionados con el departamento de Administración. El principal objetivo radica en la actualización y mejora de los servicios relacionados con esta unidad.

5.2. Análisis de los sistemas de información actuales

Objetivo:

“En esta tarea se lleva a cabo el estudio de los sistemas de información actuales afectados por el PSI. Para cada sistema de información se recogen, al menos, las características básicas relativas a datos, software de aplicación, procesos de la organización a los que da soporte y de qué forma lo hace, flexibilidad, carencias, riesgos y posibles amenazas.

En función del tipo de sistema de información y de los objetivos de su estudio se recopila además, para cada uno de ellos, información procedente de diversos puntos de vista (la opinión de usuarios de los sistemas de información, de analistas de desarrollo, de personal de operación, etc.).”

Descripción general de los sistemas de información actuales:

El único sistema de información actual que dispone la FBCV está basado en el software de Gestión BILLGEST, del cual hay un estudio realizado en el **ANEXO 1 – Estudio Preliminar del software de gestión actual**

El resto de procesos involucrados en el PSI no disponen de una infraestructura actualmente y no serán analizados.

6. Diseño del modelo de sistemas de información

El objetivo de esta actividad es identificar y definir los sistemas de información que van a dar soporte a los procesos de la organización afectados por el Plan de Sistemas de Información. Para ello, en primer lugar, se analiza la cobertura que los sistemas de información actuales dan a los requisitos recogidos en el catálogo elaborado en las actividades Estudio de la Información Relevante (PSI 3) e Identificación de Requisitos (PSI 4). Esto permitirá efectuar un diagnóstico de la situación actual, a partir del cual se seleccionan los sistemas de información actuales considerados válidos, identificando las mejoras a realizar en los mismos.

Por último, se definen los nuevos sistemas de información necesarios para cubrir los requisitos y funciones de los procesos no soportados por los sistemas actuales seleccionados.

Teniendo en cuenta los resultados anteriores, se elabora el modelo de sistemas de información válido para dar soporte a los procesos de la organización incluidos en el ámbito del Plan de Sistemas de Información.

Tareas propuestas para esta actividad:

- 6.1. Diagnóstico de la situación actual
- 6.2. Definición del modelo de Sistemas de Información

6.1. Diagnóstico de la situación actual

Objetivo:

“Para llegar a un diagnóstico sobre la situación actual, se tiene en cuenta la valoración de los sistemas de información actuales realizada en la actividad Estudio de los Sistemas de Información Actuales (PSI

5), y se estudia la cobertura de requisitos que se tiene con ellos. Esto permite determinar los requisitos del catálogo no cubiertos por los sistemas de información actuales, estudiando su criticidad y prioridad. En paralelo, se analiza el modelo de información obtenido en la tarea Análisis de las Necesidades de Información (PSI 4.2). Se determina si existen entidades o relaciones del mismo, que no aparecen recogidas en la situación actual o que, estando recogidas, su tratamiento actual no responde a los nuevos requisitos. Como resultado del análisis anterior, se seleccionan los sistemas de información a conservar y se elabora, si procede, la relación de mejoras a realizar en cada uno de ellos para cubrir los requisitos que le afectan.”

Diagnóstico de la situación actual:

Se analizan las necesidades cubiertas y las no cubiertas del sistema de información actual:

1) Administración

- Entradas:

Clubes: Si

Socios: Si

Actas: Si

Impugnaciones: No

-Salidas:

Impresiones: Si

Carnets: Si

Listados: Si

Copias de seguridad: Si

Publicidad, CRM: No

- 2) Comité de Apelación: El proceso no se encuentra cubierto por ningún sistema de información.
- 3) Delegaciones: El proceso no se encuentra cubierto por ningún sistema de información.
- 4) Presidencia: El proceso no se encuentra cubierto por ningún sistema de información.
- 5) Tesorería: El proceso no se encuentra cubierto por ningún sistema de información.
- 6) Dirección deportiva: El proceso no se encuentra cubierto por ningún sistema de información.

6.2. Definición del modelo de sistemas de información

Objetivo:

“Esta tarea tiene como objetivo representar el conjunto de sistemas de información que da soporte a los procesos de la organización afectados, describiendo sus relaciones e interfaces, así como definir qué sistemas de información actuales se mantendrán con las mejoras propuestas, y qué sistemas de información nuevos cubrirán los requisitos no soportados por los sistemas de información actuales.

Para identificar cada sistema de información nuevo se analizan:

- Los sistemas de información actuales que se conservan.
- Los requisitos no cubiertos por los sistemas de información actuales. Se realiza una identificación inicial de sistemas de información, agrupando actividades homogéneas de los procesos de la organización afectados que actúan sobre información común.
- Diferentes tipos de sistemas de información (de gestión, de soporte a la toma de decisiones, especiales, etc.).
- Interfaces entre sistemas de información, con el objetivo de minimizarlas.
- Tecnología especial requerida, si procede.

Las conclusiones obtenidas de dicho análisis sirven para identificar cada sistema de información nuevo y elaborar el modelo de sistemas de información.

Por último y contando con la experiencia de los participantes en esta tarea, se obtiene el modelo de sistemas de información, que incluye un diagrama de representación de todos ellos, con sus conexiones e interfaces, y una descripción de cada sistema de información con el conjunto de actividades y requisitos que cubre, así como el tipo de sistema de que se trata.”

Modelo de sistemas de información:

Tras analizar los procesos se estima que el sistema de información general intra departamental quedará de la siguiente manera:

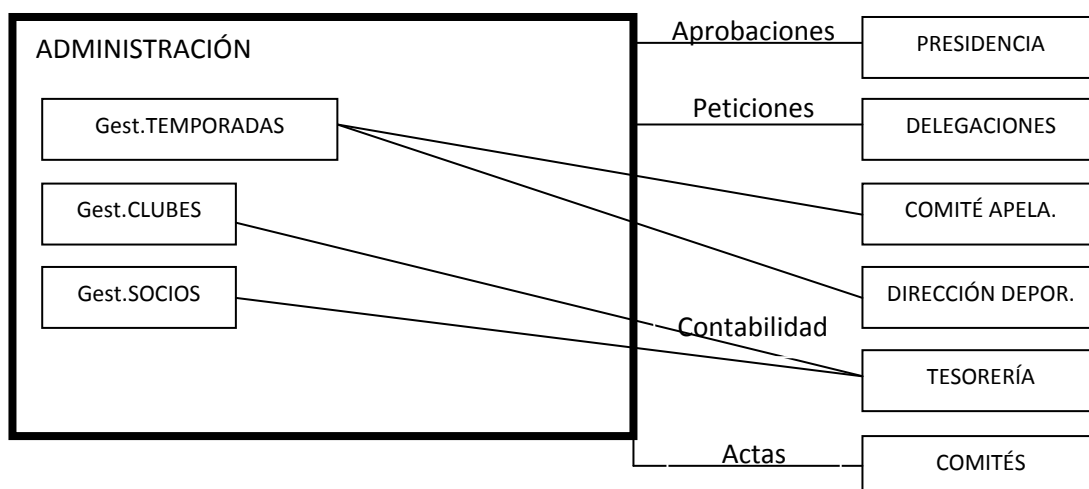


Figura 6.1. Diagrama de relación entre departamentos

Administración: El sistema de información del departamento de administración recoge todas las peticiones formuladas por el resto de departamentos.

- **Temporadas:** Actualización del Sistema de Gestión
- **Clubes:** Actualización del Sistema de Gestión
- **Socios:** Actualización del Sistema de Gestión

Comité de Apelación: Se conserva su sistema de información actual.

Delegaciones: Se conserva su sistema de información actual.

Presidencia: Se conserva su sistema de información actual.

Tesorería: Se conserva su sistema de información actual.

Dirección deportiva: Se conserva su sistema de información actual.

7. Definición de la arquitectura tecnológica

En esta actividad se propone una arquitectura tecnológica que de soporte al modelo de información y de sistemas de información incluyendo, si es necesario, opciones. Para esta actividad se tienen en cuenta especialmente los requisitos de carácter tecnológico, aunque es necesario considerar el catálogo completo de requisitos para entender las necesidades de los procesos y proponer los entornos tecnológicos que mejor se adapten a las mismas.

Tareas propuestas para esta actividad:

- 7.1. Identificación de las necesidades de infraestructura tecnológica
- 7.2. Selección de la arquitectura tecnológica

7.1. Identificación de las necesidades de infraestructura tecnológica

Objetivo:

“Esta tarea tiene el objetivo de analizar las necesidades de infraestructura tecnológica y proponer las alternativas viables desde el punto de vista tecnológico, para dar respuesta a dichas necesidades. Para ello, se comienza analizando el modelo de sistemas de información y el catálogo de requisitos, en especial los de carácter técnico. Se identifican las necesidades (entornos necesarios, conectividad y comunicaciones entre ellos, disponibilidad, servicios críticos, etc.).

A continuación se determinan las posibles alternativas de infraestructura tecnológica, definiendo los componentes, a alto nivel, y representando gráficamente cada una de ellas. Es necesario establecer la forma de gestionar la infraestructura tecnológica para responder a las necesidades identificadas. La visión aportada por los consultores de Tecnologías de la Información y Comunicaciones (TIC) debe ser de futuro, considerando la posible evolución de las distintas tecnologías candidatas, así como de las actualmente incorporadas en la organización. Es imprescindible contar, en este análisis, con la información relativa a los entornos tecnológicos de la situación actual, así como los estándares existentes en la organización.”

Identificación de necesidades:

Las necesidades de infraestructura respecto al antiguo sistema de información no variarán, por lo que simplemente se realizara una actualización de los sistemas para dar soporte al nuevo sistema de gestión a introducir.

7.2. Selección de la arquitectura tecnológica

Objetivo:

“Esta tarea está encaminada a la selección de una alternativa de plataforma tecnológica para determinar lo que llamaremos arquitectura tecnológica, que recoge la infraestructura más adecuada para dar soporte, en el contexto de la organización, al modelo de información y de sistemas de información propuesto.

Para cada alternativa, se debe analizar su impacto en la organización, así como los medios y el tiempo necesarios para su implantación. Se deben tener en cuenta los recursos tecnológicos actuales para evaluar los cambios necesarios.

Se realiza un estudio de cada propuesta, indicando ventajas e inconvenientes, así como el nivel de respuesta a las necesidades identificadas en la tarea anterior.

Por último, una estimación económica global puede ayudar a elegir la alternativa que va a ser propuesta, para la cual pueden incluirse opciones.”

Arquitectura tecnológica:

La infraestructura tecnológica con la que cuenta actualmente la FBCV corresponde a dos máquinas IBM PC X86 compatibles con sistema operativo Microsoft Windows XP y Microsoft Windows 98 SE respectivamente. Uno de ellos dispone de una impresora Epson de tipo matricial.

Los ordenadores están conectados entre sí por una red de tipo TCP/IP mediante un cable de par trenzado, y uno de ellos realiza la función de conexión compartida a internet.

Vistas las necesidades de infraestructuras tecnológicas basadas en la situación actual y los requisitos (no se establecerá un software de gestión para un entorno multicapa, multiusuario o multipuesto) se establece que:

1. **Acceso a Internet:** la única modificación corresponderá a la necesidad de independencia en las conexiones de internet, para ello se sustituirá el cable de par trenzado por un switch de 4 puertos 10/100mbps. La conexión de alta velocidad por cable de fibra óptica permanece inalterada.

La estructura general queda de la siguiente manera:

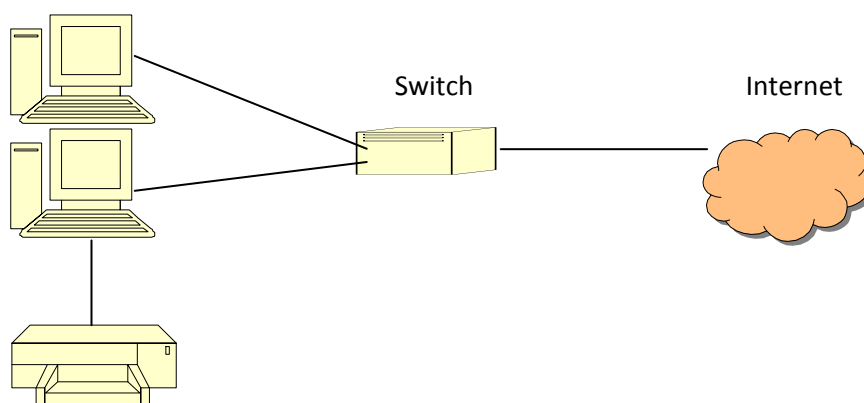


Figura 7.1. Estructura física

2. **Sistemas Operativos:** Debido principalmente a la existencia de licencias y al conocimiento de los usuarios, así como a la elección del sistema operativo de red, la decisión tomada en este aspecto es la de mantener los sistemas operativos en los equipos existentes, e introducir en los posibles nuevos equipos el sistema operativo de Microsoft más actual con el que cuenten, ya que los sistemas de gestión serán 100% compatibles con cualquier S.O. Microsoft.
3. **Hardware:** Se opta por hardware clónico analizando las prestaciones y los precios. Debido a que la experiencia de uso de este tipo de equipos en la empresa ha resultado exitosa no se observa la necesidad del cambio de equipos, Se reutilizarán los equipos existentes, adquiriendo los necesarios según Sistemas y aplicaciones.

8. Definición del plan de acción

En el Plan de Acción, que se elabora en esta actividad, se definen los proyectos y acciones a llevar a cabo para la implantación de los modelos de información y de sistemas de información, determinados en las actividades Identificación de Requisitos (PSI 4) y Diseño del Modelo de Sistemas de Información (PSI 6), con la arquitectura tecnológica propuesta en la actividad Definición de la Arquitectura Tecnológica (PSI 7). El conjunto de estos tres modelos constituye la arquitectura de información.

Dentro del Plan de Acción se incluye un calendario de proyectos, con posibles alternativas, y una estimación de recursos, cuyo detalle será mayor para los más inmediatos. Para la elaboración del calendario se tienen que analizar las distintas variables que afecten a la prioridad de cada proyecto y sistema de información. El orden definitivo de los proyectos y acciones debe pactarse con los usuarios, para llegar a una solución de compromiso que resulte la mejor posible para la organización. Por último, se propone un plan de mantenimiento para el control y seguimiento de la ejecución de los proyectos, así como para la actualización de los productos finales del Plan de Sistemas de Información.

Tareas propuestas para esta actividad:

- 8.1. Definición de proyectos a realizar
- 8.2. Elaboración del plan de mantenimiento

8.1. Definición de proyectos a realizar

Objetivo:

“Se determinan los proyectos y acciones necesarios para implantar la arquitectura de información propuesta, definiendo para cada proyecto los objetivos que cubre y cualquier observación que se considere relevante.”

A continuación, se asignan prioridades tratando de combinar diferentes criterios como:

- Relación con los objetivos considerados en el Plan de Sistemas de Información.
- Condicionantes técnicos que impliquen dependencias entre proyectos.
- Tiempo de implantación.
- Beneficios para la organización (tangibles e intangibles).
- Limitaciones y consideraciones relativas a la organización (impacto, necesidades de formación, etc.).
- Recursos disponibles a corto y medio plazo, tanto de las áreas de Sistemas de Información y Tecnologías de la Información como de los usuarios.
- Situación de riesgo de algunos de los sistemas actuales a sustituir o mejorar.
- Otros.

Después del estudio de todos los elementos anteriores, se propone el calendario de proyectos que mejor conjugue todas las restricciones analizadas previamente, estimando fechas de principio y fin de cada uno de ellos, así como los recursos necesarios para los más inmediatos. Se hará énfasis en los objetivos estratégicos soportados por el Plan de Sistemas de Información.

Por último, se completa el plan de proyectos considerando los factores críticos de éxito para llevar a cabo la propuesta, así como el plan de acciones necesarias, deducidas del análisis de dichos factores (planes de formación, gestión del cambio, etc.).

Las conclusiones obtenidas deben ser contrastadas y modificadas, si se estima conveniente, con las aportaciones de los usuarios.

Plan de proyectos:

Los criterios de decisión se establecerán en base a:

- 1) Prioridad
- 2) Dependencias
- 3) Coste

Por tanto, se implantarán los sistemas de información de la siguiente manera:

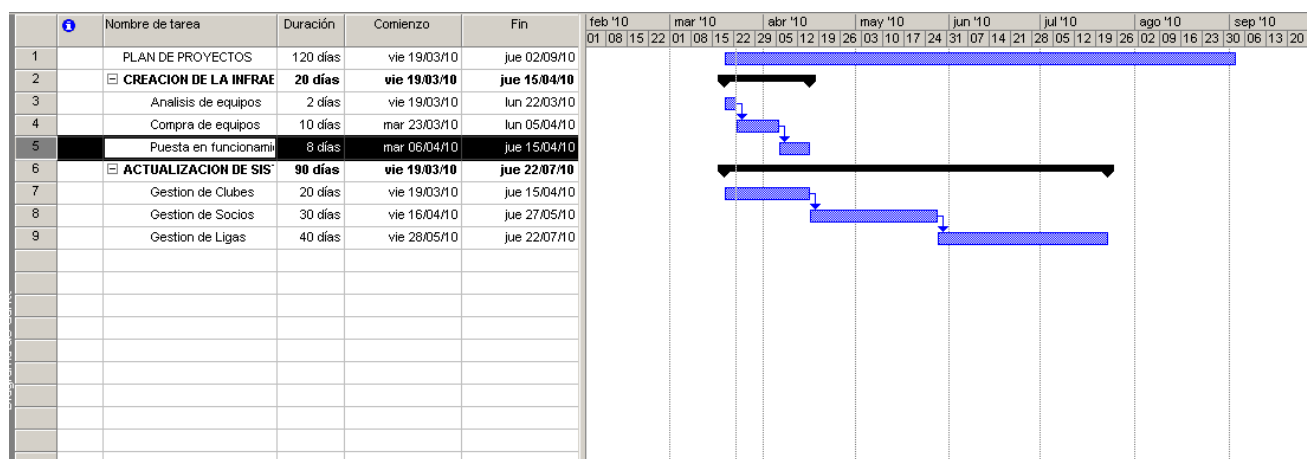


Figura 8.1. Plan de implantación

En primera instancia se implantará la infraestructura tecnológica que dará soporte a los proyectos del Plan de Sistemas de información. Consistirá en la compra de los nuevos equipos que actualizaran los equipos existentes, el switch y las impresoras acordadas.

Una vez instalada la base tecnológica se procederá a la implantación de los diferentes sistemas software estudiado. Al ser una única implantación de un programa cerrado no conllevará un calendario de implantación en fases en el cliente. El desarrollo contemplará en un primer momento la creación del modulo de clubes, posteriormente el de socios y en último lugar el modulo de ligas.

8.2. Elaboración del plan de mantenimiento

Objetivo:

“Una vez establecido el plan de acción, se deben definir las acciones que permitan mantener actualizado el Plan de Sistemas de Información a su terminación, y conocer el grado de avance de los proyectos que en él se han definido.”

Todo ello se denomina Plan de Mantenimiento del Plan de Sistemas de Información.

En este plan de mantenimiento, entre otros aspectos que se puedan considerar relevantes para la organización, se deben establecer los productos finales del Plan de Sistemas de Información que se van a mantener actualizados, el soporte para los mismos, y cuándo y por quién van a ser modificados, así como la frecuencia y situaciones en los que se debe revisar el plan de proyectos y los responsables de hacerlo.

También se determina a quiénes se informará, y con qué periodicidad, del grado de avance del plan establecido o de los cambios que en él se produzcan.

Plan de mantenimiento:

Se convocarán reuniones semanales para los grupos de trabajo correspondientes a cada tarea. Al final de estas reuniones se rellenará un acta donde se expondrán los avances conseguidos hasta el momento y las diferentes incidencias que hayan surgido.

Una vez este desarrollada una beta preliminar instalable, se convocara al personal de administración para las primeras pruebas e instalaciones. Es el personal de administración el responsable de valorar la implantación.

Las incidencias que vayan surgiendo serán informadas por los medios habituales, email o teléfono, al director del proyecto. Él es el encargado de solventarlas en el periodo más corto de tiempo posible.

En caso de que aparezcan nuevas necesidades, se estudiará cada caso concreto, en caso de ser modificaciones de poca envergadura se incluirán directamente en el proyecto. En otro caso, tendrán que ser evaluadas y presupuestadas por parte del implantador.

9. Revisión y aprobación del PSI

Esta actividad tiene como objetivo contrastar con los responsables de la dirección del Plan de Sistemas de Información la arquitectura de información y el plan de acción elaborados anteriormente, para mejorar la propuesta si se considera necesario y por último, obtener su aprobación final.

Tareas propuestas para esta actividad:

- 9.1. Aprobación del Plan de Sistemas de Información

9.1. Aprobación del PSI

Objetivo:

“Se entrega la propuesta final, y se solicita formalmente al Comité de Dirección del Plan de Sistemas de Información la aprobación de la misma. Por último, se debe informar de los resultados a las unidades organizativas participantes y a todas aquellas afectadas por los resultados del Plan de Sistemas de Información.”

Aprobación del PSI:

Una vez reunidos presidente y vicepresidentes se acuerda por unanimidad la aprobación del Plan de Sistemas de Información propuesto. Se comunica al departamento de Administración y a su responsable, para comenzar la consecución de los proyectos propuestos. A su vez administración lo comunica al Jefe del Proyecto.

Anexo II-2. Índice:

1.	Establecimiento del alcance del sistema.....	2
1.1.	Estudio de la solicitud	3
1.2.	Identificación del alcance del sistema.....	4
1.3.	Especificación del alcance del EVS	4
2.	Estudio de la situación actual.....	5
2.1.	Valoración del estudio de la situación actual.....	6
2.2.	Descripción de los sistemas de información actuales.....	6
2.3.	Realización del diagnóstico de la situación actual	8
3.	Definición de requisitos del sistema	8
3.1.	Catalogación de requisitos	8
4.	Estudio de alternativas de solución	11
4.1.	Preselección de alternativas de solución	11
4.2.	Descripción de las alternativas de solución	12
5.	Valoración de las alternativas	15
5.1.	Estudio de la inversión	15
5.2.	Estudio de los riesgos	16
6.	Selección de la solución	17
6.1.	Evaluación de las alternativas y selección.....	18

• Estudio de Viabilidad del Sistema (EVS)

“Mientras que el Plan de Sistemas de Información tiene como objetivo proporcionar un marco estratégico que sirva de referencia para los Sistemas de Información de un ámbito concreto de una organización, el objetivo del Estudio de Viabilidad del Sistema es el análisis de un conjunto concreto de necesidades para proponer una solución a corto plazo, que tenga en cuenta restricciones económicas, técnicas, legales y operativas. La solución obtenida como resultado del estudio puede ser la definición de uno o varios proyectos que afecten a uno o varios sistemas de información ya existentes o nuevos. Para ello, se identifican los requisitos que se ha de satisfacer y se estudia, si procede, la situación actual.

A partir del estado inicial, la situación actual y los requisitos planteados, se estudian las alternativas de solución. Dichas alternativas pueden incluir soluciones que impliquen desarrollos a medida, soluciones basadas en la adquisición de productos software del mercado o soluciones mixtas. Se describe cada una de las alternativas, indicando los requisitos que cubre.

Una vez descritas cada una de las alternativas planteadas, se valora su impacto en la organización, la inversión a realizar en cada caso y los riesgos asociados. Esta información se analiza con el fin de evaluar las distintas alternativas y seleccionar la más adecuada, definiendo y estableciendo su planificación.

Si en la organización se ha realizado con anterioridad un Plan de Sistemas de Información que afecte al sistema objeto de este estudio, se dispondrá de un conjunto de productos que proporcionarán información a tener en cuenta en todo el proceso.

Las actividades que engloba este proceso se recogen en la siguiente figura, en la que se indican las actividades que pueden ejecutarse en paralelo y las que precisan para su realización resultados originados en actividades anteriores.”

1. Establecimiento del alcance del sistema

“En esta actividad se estudia el alcance de la necesidad planteada por el cliente o usuario, o como consecuencia de la realización de un PSI, realizando una descripción general de la misma. Se determinan los objetivos, se inicia el estudio de los requisitos y se identifican las unidades organizativas afectadas estableciendo su estructura.

Se analizan las posibles restricciones, tanto generales como específicas, que puedan condicionar el estudio y la planificación de las alternativas de solución que se propongan.

Si la justificación económica es obvia, el riesgo técnico bajo, se esperan pocos problemas legales y no existe ninguna alternativa razonable, no es necesario profundizar en el estudio de viabilidad del sistema, analizando posibles alternativas y realizando una valoración y evaluación de las mismas, sino que éste se orientará a la especificación de requisitos, descripción del nuevo sistema y planificación.

Se detalla la composición del equipo de trabajo necesario para este proceso y su planificación. Finalmente, con el fin de facilitar la implicación activa de los usuarios en la definición del sistema, se identifican sus perfiles, dejando claras sus tareas y responsabilidades.”

Tareas propuestas para esta actividad:

- 1.1. Estudio de la solicitud
- 1.2. Identificación del alcance del sistema
- 1.3. Especificación del alcance del EVS

1.1. Estudio de la solicitud

Objetivo:

“Se realiza una descripción general de la necesidad planteada por el usuario, y se estudian las posibles restricciones de carácter económico, técnico, operativo y legal que puedan afectar al sistema. Antes de iniciar el estudio de los requisitos del sistema se establecen los objetivos generales del Estudio de Viabilidad, teniendo en cuenta las restricciones identificadas anteriormente.

Si el sistema objeto de estudio se encuentra en el ámbito de un Plan de Sistemas de Información vigente, se debe de tomar como referencia el catálogo de requisitos y la arquitectura de información resultante del mismo, como información adicional para la descripción general del sistema y determinación de los requisitos iniciales.”

Estudio de la situación actual:

La necesidad prioritaria para el cliente, que será el punto principal del objeto de estudio, es la actualización de su herramienta de gestión actual. Como se ha comentado anteriormente, la funcionalidad principal que debe aportar la actualización es la misma que la que aportaba la herramienta anterior, añadiendo algunas nuevas funciones y actualizando otras conforme a los nuevos estatutos, reglas y prioridades de la FBCV.

La funcionalidad por tanto radica en la gestión de los datos de los clientes, los cuales son los socios y clubes adscritos a la federación, y la gestión de las temporadas de ligas anuales que se desarrollan en la comunidad. Los datos que genere el programa serán tratados con otras herramientas las cuales no serán desarrolladas.

La restricción principal a la que la FBCV se enfrenta es la de carácter económico. La FBCV cuenta con unos ingresos muy limitados, dados en parte por la parte proporcional que cobra a los socios inscritos en clubes, como por los mismos clubes. Por otro lado, cuenta con una subvención anual del Consell Valencià de L'Esport. Con estos ingresos debe cubrir salarios de los componentes de la Federación, gastos derivados de los bienes muebles e inmuebles, etc. etc.

La aplicación además tendrá que tener una muy buena interfaz de usuario. La formación del personal que trabajara con la herramienta es baja, por tanto se hace necesario un entorno de fácil trabajo.

Los recursos físicos que soportaran la carga del programa de gestión superaran con creces los mínimos requisitos indispensables para el programa.

Con respecto a las restricciones legales, la única norma que restringirá el uso de los ficheros con datos sensibles será la LOPD¹. No habrá intercambio de ficheros con ninguna otra entidad, ni será necesario un control de acceso a la aplicación.

¹ LOPD: Ley Orgánica 15/1999 de Protección de Datos de Carácter Personal.

1.2. Identificación del alcance del sistema

Objetivo:

“Se analiza el alcance de la necesidad planteada y se identifican las restricciones relativas a la sincronización con otros proyectos, que puedan interferir en la planificación y futura puesta a punto del sistema objeto del estudio. Esta información se recoge en el catálogo de requisitos. Si el sistema pertenece al ámbito de un Plan de Sistemas de Información, se debe tener en cuenta la arquitectura de información propuesta para analizar el alcance del sistema e identificar los sistemas de información que quedan fuera del ámbito del estudio. Además, se estudia el plan de proyectos, para determinar las posibles dependencias con otros proyectos.

Una vez establecido el alcance, se identifican las unidades organizativas afectadas por el sistema, así como su estructura y responsables de las mismas. Para determinar los responsables se tiene en cuenta a quiénes afecta directamente y quiénes pueden influir en el éxito o fracaso del mismo.”

Estudio:

Para que el sistema de gestión se desarrolle correctamente es necesario que el soporte físico este instalado y funcionando en producción.

La extensión del proyecto se reduce al departamento de administración y su software de gestión, por tanto quedan excluidos del ámbito de estudio los sistemas de información de:

- Comité de Apelación
- Delegaciones
- Presidencia
- Tesorería
- Dirección deportiva

1.3. Especificación del alcance del EVS

Objetivo:

“En función del alcance del sistema y los objetivos del Estudio de Viabilidad del Sistema, se determinan las actividades y tareas a realizar. En particular, hay que decidir si se realiza o no el estudio de la situación actual y, en el caso de considerarlo necesario, con qué objetivo. Si el sistema pertenece al ámbito de un Plan de Sistemas de Información, los criterios que pueden orientar sobre la necesidad de llevar a cabo el estudio de la situación actual dependen de la arquitectura de información propuesta, en cuanto a la identificación de los sistemas de información actuales, implicados en el ámbito de este estudio, que se haya decidido conservar.

Se identifican los usuarios participantes de las distintas unidades organizativas afectadas para la realización del Estudio de Viabilidad del Sistema, determinando previamente sus perfiles y responsabilidades. Debe comunicarse el plan de trabajo a los usuarios identificados como implicados en el Estudio de Viabilidad, solicitando su aceptación y esperando su confirmación.”

Objetivos del estudio de la situación actual:

En vista a que el proyecto se basa en una actualización de un software de gestión creado con anterioridad, se hace indispensable el estudio de las herramientas anteriores a la nueva implantación.

Una vez estudiado el antiguo sistema, se decide, conjuntamente con el departamento afectado, que nuevas opciones contemplara el sistema a desarrollar, y cuáles deben ser modificadas.

También se identifica si el nuevo sistema abarcará más usuarios de los actuales, si modifica la gestión de otros departamentos, si contempla la eliminación de herramientas con las que se complementaba el actual sistema, si hay duplicidad de la información, etc. etc.

Los usuarios responsables de las unidades organizativas por tanto serán:

ADMINISTRACION/ TESORERIA/CONTABILIDAD**Dña. Cristina González**

Su principal responsabilidad es la de facilitar la captura de requisitos al equipo del proyecto, seguimiento del proyecto, introducción de los datos en el sistema y testing.

JEFE DE PROYECTO**D. Carlos Soriano Lorente**

Es el principal responsable del proyecto, la persona a la que deben remitirse todas las incidencias y comunicaciones.

Tras el estudio se determina que no afectara a nuevas áreas, ni modificara el uso de herramienta alguna excepto del sistema de gestión principal.

Una vez asignados los roles, se comunica al departamento de administración cuál es el plan de trabajo propuesto.

2. Estudio de la situación actual

“La situación actual es el estado en el que se encuentran los sistemas de información existentes en el momento en el que se inicia su estudio.

Teniendo en cuenta el objetivo del estudio de la situación actual, se realiza una valoración de la información existente acerca de los sistemas de información afectados. En función de dicha valoración, se especifica el nivel de detalle con que se debe llevar a cabo el estudio. Si es necesario, se constituye un equipo de trabajo específico para su realización y se identifican los usuarios participantes en el mismo.

Si se decide documentar la situación actual, normalmente es conveniente dividir el sistema actual en subsistemas. Si es posible se describirá cada uno de los subsistemas, valorando qué información puede ser relevante para la descripción.

Como resultado de esta actividad se genera un diagnóstico, estimando la eficiencia de los sistemas de información existentes e identificando los posibles problemas y las mejoras.”

Tareas propuestas para esta actividad:

- 2.1. Valoración del estudio de la situación actual
- 2.2. Descripción de los sistemas de información actuales
- 2.3. Realización del diagnóstico de la situación actual

2.1. Valoración del estudio de la situación actual

Objetivo:

“En función de los objetivos establecidos para el estudio de la situación actual, y considerando el contexto del sistema especificado en la descripción general del mismo, se identifican los sistemas de información existentes que es necesario analizar con el fin de determinar el alcance del sistema actual. Asimismo, se decide el nivel de detalle con el que se va a llevar a cabo el estudio de cada uno de los sistemas de información implicados. En el caso de haber realizado un Plan de Sistemas de Información que afecte a dicho sistema, se toma como punto de partida para este análisis la arquitectura de información propuesta. Para poder abordar el estudio, se realiza previamente una valoración de la información existente acerca de los sistemas de información afectados por el EVS. Se debe decidir si se realizan o no los modelos lógicos del sistema actual o si se describe el modelo físico, en función de los siguientes criterios:

- Si existen los modelos lógicos, y son fiables, se utilizan en la tarea Descripción de los Sistemas de Información Existentes (EVS 2.3)
- Si no existen dichos modelos, o no son fiables, se considera el tiempo de vida estimado para el sistema de información en función de la antigüedad, la obsolescencia de la tecnología o la falta de adecuación funcional para determinar si se obtienen los modelos lógicos y físicos del sistema actual o por el contrario no se elabora ningún modelo.

La información relativa a los sistemas de información que se decida analizar, se obtiene mediante sesiones de trabajo con los Directores de Usuarios y el apoyo de los profesionales de Sistemas y Tecnologías de la Información y Comunicaciones (STIC) que se considere necesario.”

Estudio:

Como se ha comentado anteriormente, el único sistema de información que va a ser estudiado en profundidad, será el que maneja el departamento de administración.

No existe ningún modelo lógico desarrollado para él, por tanto se deberá analizar y desarrollar el modelo.

Respecto al modelo físico, se especificará conjuntamente con el modelo lógico ya que debido a su limitada amplitud no se hace necesario un mayor detalle.

2.2. Descripción de los sistemas de información actuales

Objetivo:

“En esta tarea se describen los sistemas de información existentes afectados, según el alcance y nivel de detalle establecido en la tarea Valoración del Estudio de la Situación Actual (EVS 2.1), mediante sesiones de trabajo con los usuarios designados para este estudio.

Si se ha decidido describir los sistemas a nivel lógico, y si existe un conocimiento suficiente de los sistemas de información a especificar, puede hacerse directamente, aplicando las técnicas de modelización y siguiendo un método descendente. Si no se dispone del conocimiento suficiente, se construyen los modelos a partir de la descripción del modelo físico, es decir, de forma ascendente. Si se tiene que describir el modelo físico, se puede hacer mediante un Diagrama de Representación en el que se recojan todos los componentes físicos y sus referencias cruzadas. Otra opción es describir el modelo físico de forma más detallada, para lo que es necesaria la utilización de herramientas de tipo scanner.

Es conveniente indicar la localización geográfica y física actual de los módulos y datos de los sistemas de información afectados, evaluando al mismo tiempo la redundancia en las distintas unidades organizativas.”

Modelo:

Una vez realizadas las entrevistas con el responsable de administración, se elabora un pequeño modelo lógico² de las unidades de negocio afectadas, que complementa al mostrado en el Plan de Sistemas de Información:

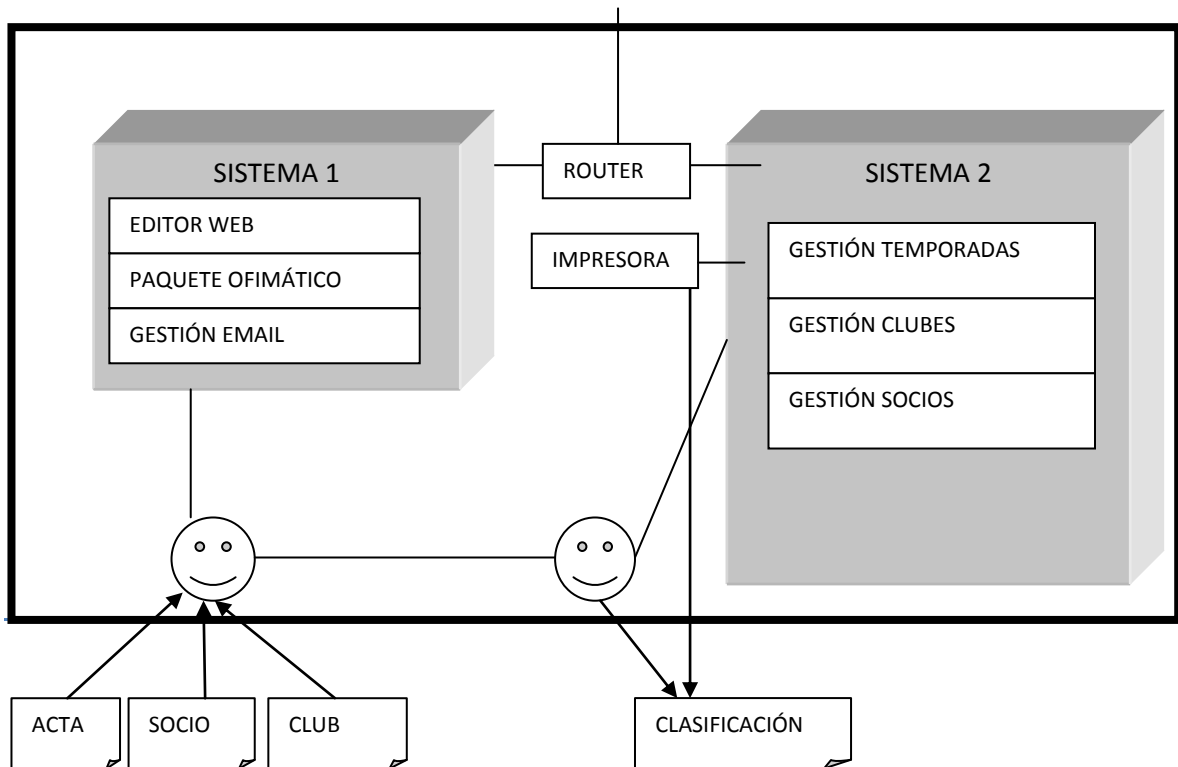


Figura 2.1. Diagrama de despliegue.

² Diagrama de despliegue UML 2.0 <http://advanceduml.wordpress.com/2009/01/30/deployment-diagrams-communication-paths/>

2.3. Realización del diagnóstico de la situación actual

Objetivo:

“Con el fin de elaborar el diagnóstico de la situación actual se analiza la información de los sistemas de información existentes, obtenida en la tarea anterior y se identifican problemas, deficiencias y mejoras. Estas últimas deben tenerse en cuenta en la definición de los requisitos.

En el caso de haber realizado un Plan de Sistemas de Información, se considera la valoración realizada sobre los sistemas de información actuales que pertenecen al ámbito de este estudio.

Si se ha tomado la decisión de no describir la situación actual, se realiza un diagnóstico global justificando esta decisión.”

Diagnóstico:

Partiendo de la información del Plan de Sistemas de Información, se decide que la estructura del sistema de información del departamento sea la misma que la actual. Los formularios e incidencias serán recogidos manualmente por el departamento, y serán ellos quienes introduzcan los datos en el sistema. De la misma manera, tampoco variará la gestión de la web corporativa ni se automatizarán envíos a socios o clubes.

Se decide por tanto que la ampliación únicamente afecte a la actualización de la plataforma a un entorno más actual, con más posibilidades de ampliación futura, más velocidad y flexibilidad en la Interfaz de Usuario, mayor tolerancia a errores, y modificando los requisitos en base a nuevas reglas federativas.

3. Definición de requisitos del sistema

“Esta actividad incluye la determinación de los requisitos generales, mediante una serie de sesiones de trabajo con los usuarios participantes, que hay que planificar y realizar.

Una vez finalizadas, se analiza la información obtenida definiendo los requisitos y sus prioridades, que se añaden al catálogo de requisitos que servirá para el estudio y valoración de las distintas alternativas de solución que se propongan.”

Tareas propuestas para esta actividad:

3.1. Catalogación de requisitos

3.1. Catalogación de requisitos

Objetivo:

“Se analiza la información obtenida en las sesiones de trabajo para la Identificación de Requisitos, definiendo y catalogando los requisitos (funcionales y no funcionales) que debe satisfacer el sistema, indicando sus prioridades.

Se incluirán también requisitos relativos a distribución geográfica y entorno tecnológico.”

Requisitos funcionales:

Especificación completa de requisitos, ampliando la expuesta en el Plan de Sistemas de Información:

Gestión de Temporadas:

- Apertura y cierre de temporadas.
Se deberá poder abrir y cerrar temporadas, eligiendo para cada club que ligas disputará.
- Copias de seguridad
Todas las temporadas podrán ser guardadas en formatos propios del programa de gestión.
- Especificación de número de grupos y divisiones por club.
Se amplía el número de grupos y divisiones actual, ahora deberá contemplar hasta seis grupos en cuatro divisiones.
- Listado general o por promedios
El programa generará listados de actas y listados de jornadas, pudiendo acceder directamente por promedio.
- Impresión de listados
Todos los listados que genere el programa deben poder ser imprimibles con una Impresora compatible Microsoft® Windows Vista/7.
- Gestión de actas
Se guardará información respecto a los encuentros entre los clubes que disputan una determinada liga. Una vez las actas estén introducidas en el sistema, el programa deber ser capaz de interpretar los resultados y mostrar los listados de puntuación referentes a las actas.
- Árbitros
Es necesario que el programa mantenga una relación de árbitros en los encuentros.
- Sanciones
Las sanciones podrán ser introducidas explícitamente o automáticamente por el sistema en caso de alineación indebida. Estas sanciones modificarán la puntuación y promedios.

Gestión de Socios:

- Datos personales
El sistema debe almacenar de una manera fiable y segura los datos de los socios de los clubes inscritos en la FBCV. Se mantienen campos y se modifican según lo expuesto en el anexo análisis preliminar del programa de gestión BILLGEST.
- Socios junior o sénior y sus diferentes tarifas
El programa tendrá una opción para introducir y modificar tarifas según será el socio junior o sénior. Estas tarifas quedaran expuestas luego en los carnets de socio.
- Listas de Fuerza
Cada usuario tendrá una posición en una lista de fuerza, por la cual ningún jugador en una posición más retrasada en esa lista podrá jugar en una posición anterior y viceversa. En caso de hacerlo, el programa detectará la situación.

- Bajas temporales, bajas definitivas
Es posible que el usuario se de de baja temporal (pagando una cuota mínima) o baja definitiva (fin de pago). El sistema debe contemplar las dos opciones.
- Carnets de Socio
Los carnets de socio de la FBCV serán sustituidos por los nuevos carnets que genere el programa. Estos carnets modificaran su tamaño y datos impresos conforme a las nuevas exigencias de la FBCV.

Gestión de Clubes:

- Solicitudes a la FBCV
Toda solicitud de alta de club en la FBCV, una vez aceptada por presidencia, y realizados los trámites oportunos en la Federación Española, será dada de alta en el sistema. Los campos de los clubes variaran respecto a las nuevas exigencias de la FBCV.

Requisitos no funcionales:

Se describen los requisitos no funcionales acordados con la FBCV.

- Seguridad
Como se ha comentado anteriormente, no existirá control de acceso más que el del propio sistema operativo.
- Normativa Interna
El programa debe adaptarse a los estatutos internos que la FBCV tiene respecto a la creación de ligas, desarrollo y datos de socios y clubes.
- Normativa Externa
Se hace necesaria la Inscripción de los ficheros en el Registro General de la Protección de Datos. Artículo 26 LOPD. Dicha inscripción será llevada a cabo por una empresa externa asignada por la FBCV.
- Interfaz Grafica
La interfaz grafica debe ser sencilla y de fácil adaptación respecto al viejo sistema. Por tanto, se mantendrán las tabulaciones típicas de MSDOS.
- Velocidad del sistema
La velocidad del sistema debe ser rápida y mantenerse estable para una gestión media aproximada de 3000 socios y 100 clubes.
- Mantenibilidad
El mantenimiento del sistema tanto hardware como software debe resultar cómodo, ágil y de fácil reemplazo.
- Escalabilidad
Debido a que existe la posibilidad de ampliaciones futuras a corto plazo, es necesario que el sistema tenga un sólido núcleo que permita añadir nuevas funcionalidades sin perjudicar la funcionalidad básica. Además estas nuevas funcionalidades deberán ser desarrolladas en la misma plataforma.

- Compatibilidad
Se asegura una perfecta compatibilidad con sistemas Microsoft® Windows 7 / Vista tanto X64 como X86.

4. Estudio de alternativas de solución

“Este estudio se centra en proponer diversas alternativas que respondan satisfactoriamente a los requisitos planteados, considerando también los resultados obtenidos en el Estudio de la Situación Actual (EVS 2), en el caso de que se haya realizado.

Teniendo en cuenta el ámbito y funcionalidad que debe cubrir el sistema, puede ser conveniente realizar, previamente a la definición de cada alternativa, una descomposición del sistema en subsistemas.

En la descripción de las distintas alternativas de solución propuestas, se debe especificar si alguna de ellas está basada, total o parcialmente, en un producto existente en el mercado. Si la alternativa incluye un desarrollo a medida, se debe incorporar en la descripción de la misma un modelo abstracto de datos y un modelo de procesos, y en orientación a objetos, un modelo de negocio y un modelo de dominio.”

Tareas propuestas para esta actividad:

- 4.1. Preselección de alternativas de solución
- 4.2. Descripción de las alternativas de solución

4.1. Preselección de alternativas de solución

Objetivo:

“Una vez definidos los requisitos a cubrir por el sistema, se estudian las diferentes opciones que hay para configurar la solución. Entre ellas, hay que considerar la adquisición de productos software estándar del mercado, desarrollos a medida o soluciones mixtas.

Dependiendo del alcance del sistema y las posibles opciones, puede ser conveniente realizar inicialmente una descomposición del sistema en subsistemas. Se establecen las posibles alternativas sobre las que se va a centrar el estudio de la solución, combinando las opciones que se consideren más adecuadas.”

Selección de alternativas de solución:

Las opciones de solución que podrían hacer frente a la configuración de requisitos actual corresponden a:

- 1 Aplicación comercial existente.
- 2 Aplicación hecha a medida.

4.2. Descripción de las alternativas de solución

Objetivo:

“Para cada alternativa propuesta, se identifican los subsistemas que cubre y los requisitos a los que se da respuesta.

Se deben considerar también aspectos relativos a la cobertura geográfica (ámbito y limitaciones) de procesos y datos, teniendo en cuenta a su vez la gestión de comunicaciones y control de red.

En la definición de cada alternativa, se propone una estrategia de implantación teniendo en cuenta tanto la cobertura global del sistema como la cobertura geográfica.

Si la alternativa incluye desarrollo se describe el modelo abstracto de datos y el modelo de procesos, y en el caso de Orientación a Objetos, el modelo de negocio y, opcionalmente, el modelo de dominio. Se propone el entorno tecnológico que se considera más apropiado para la parte del sistema basada en desarrollo y se describen los procesos manuales.

Si la alternativa incluye una solución basada en producto se analiza su evolución prevista, adaptabilidad y portabilidad, así como los costes ocasionados por licencias, y los estándares del producto. Igualmente se valora y determina su entorno tecnológico.”

Alternativas de solución:

1. Aplicación comercial existente

El principal problema con el que cuenta la FBCV a la hora de disponer de una alternativa comercial existente es que no existen en el mercado actualmente sistemas de gestión de ligas exclusivos para ligas de billar, lo que provoca que nunca se cumplan los requisitos al 100%. Incluso seleccionando varias herramientas tampoco se podría llegar a cubrir todos los requisitos.

No obstante se estudia alguna alternativa que pudiera dar soporte a la mayor parte de las funcionalidades requeridas, incluso ampliar otras.

1) Competiciones deportivas V.10 (Sagois S.L.U.)

CALENDARIO

- Creación y gestión de campeonatos de liga (a 1 o 2 vueltas) y copa, así como fases previas y finales.
- Realiza el sorteo del campeonato de manera automática a 1 o 2 vueltas con posibilidad de establecer la primera jornada de manera manual. Es posible establecer todas las jornadas de manera manual.
- Genera los enfrentamientos de copa. Permite generar previas de liga y final de copa.
- Creación de calendarios, pudiendo seleccionar entre calendarios completos, por jornadas, o de un rango de jornadas. Posibilidad de mostrar la clasificación actual en la parte inferior de la página.
- Gestión profesional de horarios: Creación de calendarios con un número ilimitado de campos de juego. Además, ahora es posible reservar el campo de juego por parte de usuarios, y el programa tiene en cuenta estas reservas a la hora de generar el calendario

para evitar coincidencias horarias, ... (esta opción no está disponible en la licencia educativa)

CLASIFICACIONES

- Generación de clasificaciones de equipos y jugadores.
- Equipos: clasificación general, deportividad, equipos menos goleados.
- Jugadores: Goleadores, Deportividad y equipos menos goleados.

SANCIONES

- Imprime las actas de los partidos, informándonos de qué jugadores están sancionados.
- Gestión profesional de árbitros y de sanciones.
- Listas negras de DNI: permiten la detección automática, por medio del DNI, de un jugador al que no se le permita inscribirse en un campeonato.

INTERNET

- Exportación automática de equipos, calendarios y clasificaciones a HTML.
- Posibilidad de subir las páginas a un servidor privado.
- Creación de copias de seguridad, con posibilidad de subirlas a Internet.

OTROS

- Nueva base de datos de usuarios con la posibilidad de importarlos a los equipos a lo largo de los años.
- Posibilidad de almacenar equipos y jugadores para poder utilizarlos en campeonatos posteriores.
- Creación de hojas de inscripción.
- Creación de carnets.
- Gestión Económica (Sólo versión élite).

REQUISITOS DEL SISTEMA

Windows XP (con Service Pack 2)

- Pentium III 700MHz o más
- 256MB de RAM

Windows Vista

- 800 MHz o más
- 512 MB de RAM

Mac OS X 10.5

- PowerPC G4 (867MHz+), G5, o Mac con procesador Intel
- 512 MB de RAM

Mac OS X 10.4.11

- PowerPC G4, G5 o Mac con procesador Intel
- 256 MB de RAM

2) PCLIGA 2000 (Manuel Dan Pozo, Freeware)

El programa permite introducir los enfrentamientos de cada jornada.

Para cada partido:

- Jugadores: Titulares y suplentes.
- Goleadores y amonestaciones.
- Árbitro, táctica de cada equipo.
- Fecha del encuentro, asistencia de público y un bloc de notas.

Para cada equipo:

- Todos los datos referentes al club (presidente, entrenador, estadio, nº socios, etc.).
- La plantilla de jugadores (pueden incluirse las fotos).
- Reflejar si el equipo está sancionado con puntos.

Para cada jugador:

- Nombre y apodo futbolístico.
- Fecha de nacimiento, foto, nacionalidad y altura.
- Demarcación, dorsal, ficha, cláusula de rescisión y año de fin de contrato.

Realizar una clasificación de la Liga, manejando diferentes posibilidades:

- Clasificación por puntos.
- Clasificación por puntos conseguidos en casa o fuera.
- Clasificación por goles a favor y en contra.
- Selección de un intervalo de jornadas o de fechas.
- Zonas en la clasificación: Liga de Campeones, ascenso, descenso, etc.
- Flechas indicativas de cambio de puesto de un equipo.
- Evolución de un equipo en la liga en varios parámetros.
- Evolución general de la liga en varios parámetros.

Estadísticas generales de la liga y de jugadores.

Tabla con todos los resultados

Pueden contrastarse los datos de los jugadores, es decir, todos los datos de un jugador relativos a la competición (goles, partidos jugados, etc.) se obtienen de los datos que se hayan introducido en los partidos, o bien, pueden introducirse los datos manualmente independientemente de que no se hayan introducido directamente en los enfrentamientos.

Extracción de todos los datos (Clasificaciones, Jornadas, Estadísticas) a formato HTML.

REQUISITOS DEL SISTEMA

Windows XP (con Service Pack 2)

- Pentium III 700MHz o más
- 256MB de RAM

Windows Vista

- 800 MHz o más
- 512 MB de RAM

2. Aplicación hecha a medida

La aplicación hecha a medida, al contrario que las aplicaciones comerciales, sería capaz de cubrir totalmente los requisitos exigidos por la Federación, esto es la gestión de ligas, clubes y socios con las características descritas anteriormente.

Para el caso de la aplicación hecha a medida los requisitos mínimos de sistema serían, teniendo en cuenta su plataforma con el Framework 3.5 de Microsoft®:

Windows XP SP2 / Vista / W7

- Pentium IV 2.4 GHz
- 512MB de RAM

Otro punto a destacar es la flexibilidad que proporciona este tipo de desarrollo, que además de cubrir necesidades actuales podría cubrir también necesidades futuras que todavía no se necesiten.

5. Valoración de las alternativas

“Una vez descritas las alternativas se realiza una valoración de las mismas, considerando el impacto en la organización, tanto desde el punto de vista tecnológico y organizativo como de operación, y los posibles beneficios que se esperan contrastados con los costes asociados. Se realiza también un análisis de los riesgos, decidiendo cómo enfocar el plan de acción para minimizar los mismos y cuantificando los recursos y plazos precisos para planificar cada alternativa.”

Tareas propuestas para esta actividad:

- 5.1. Estudio de la inversión
- 5.2. Estudio de los riesgos

5.1. Estudio de la inversión

Objetivo:

“Para cada alternativa de solución propuesta, se valora el impacto en la organización y se establece su viabilidad económica. Para ello, se realiza un análisis coste/beneficio que determina los costes del sistema y los pondera con los beneficios tangibles, cuantificables directamente, y con los beneficios intangibles, buscando el modo de cuantificarlos.”

Impacto y viabilidad económica:1 Aplicación comercial existente.

Existiría un gran impacto en la organización al usar una nueva aplicación comercial:

- Necesidad de nueva formación a usuarios.
- Imposibilidad de abarcar los requisitos, deberían usarse varias aplicaciones simultáneamente.
- El uso de varias aplicaciones provoca el aumento de recursos de los sistemas.
- Trato menos personalizado y mayor lentitud en la resolución de incidencias.

No obstante, las ventajas que aportaría esta opción, son varias, entre ellas el tiempo de implantación de las aplicaciones en el sistema, la fiabilidad, robustez, e inclusión de opciones no especificadas en los requisitos que pudieran ser interesantes al departamento.

El coste económico de la adopción de estas herramientas es variable, y depende mucho de las características finales que incluya la aplicación. Se asume que para que el sistema mejorara algunos aspectos del programa, el coste medio de los programas de gestión rondaría los quinientos euros (500€).

2 Aplicación hecha a medida.

El impacto en la organización de una aplicación desarrollada a medida básicamente otorgaría ventajas al sistema de información, a cambio de un desembolso económico a priori mayor. En todo caso, este desembolso entra dentro de la capacidad adquisitiva para una temporada de la FBCV. Sus características principales serían:

- La Interfaz de la nueva aplicación se adaptaría a la manera antigua de interactuar con el sistema por lo que la formación necesaria sería escasa.
- Requisitos funcionales y no funcionales 100% cubiertos.
- Rápida resolución de incidencias y peticiones.
- Posibilidad de preparar el sistema para cambios futuros.
- Tecnología actual.
- Cumplimiento de normas específicas.
- Informes personalizados

5.2. Estudio de los riesgos

Objetivo:

“Para cada alternativa se seleccionan los factores de situación que habrá que considerar, relativos tanto a la incertidumbre como a la complejidad del sistema. Se identifican y valoran los riesgos asociados y se determinan las medidas a tomar para minimizarlos.”

Riesgos:

1 Aplicación comercial existente.

Al utilizar una aplicación comercial se debe tener en cuenta que el riesgo asociado principal radica en que la funcionalidad propuesta no cubra la funcionalidad requerida por la FBCV. Se estudia para el caso de la aplicación comercial **Competiciones deportivas V.10 (Sagois S.L.U.)**.

- Al usar varias aplicaciones, se puede incurrir en duplicidad de datos.
- Difícil mantenimiento de datos.
- Problemas de incompatibilidad con normas preestablecidas.
- Difícil integración en el sistema.
- Interfaz de usuario en primera instancia poco amigable.

2 Aplicación hecha a medida.

- En caso de carga excesiva del proyecto de sistemas de información al personal de la Federación, pudieran darse retrasos en la implantación y puesta en marcha por los propios responsables del departamento.
- Es posible que la antigua interfaz del programa de gestión que va a ser adaptada no sea la interfaz más idónea a largo plazo.
- Incumplimiento de plazos de entrega por parte de los desarrolladores.
- Incumplimiento de presupuestos.

6. Selección de la solución

“Antes de finalizar el Estudio de Viabilidad del Sistema, se convoca al Comité de Dirección para la presentación de las distintas alternativas de solución, resultantes de la actividad anterior. En dicha presentación, se debaten las ventajas de cada una de ellas, incorporando las modificaciones que se consideren oportunas, con el fin de seleccionar la más adecuada. Finalmente, se aprueba la solución o se determina su inviabilidad.”

Tareas propuestas para esta actividad:

- 6.1. Evaluación de las alternativas y selección

6.1. Evaluación de las alternativas y selección

Objetivo:

“Una vez recibida la confirmación de qué alternativas van a ser presentadas para su valoración, se efectúa su presentación al Comité de Dirección, debatiendo sobre las ventajas e inconvenientes de cada una de ellas y realizando las modificaciones que sugiera dicho Comité, hasta la selección de la solución final.”

Selección:

Tras el estudio de las ventajas e inconvenientes de las opciones para sustituir la actual aplicación de gestión, se llega a la conclusión que la mejor propuesta es la creación de una herramienta partiendo de cero.

Anexo II-3. Índice:

1.	Definición del sistema	3
1.1.	Determinación del alcance del sistema.....	4
1.2.	Especificación de estándares y normas.....	4
2.	Establecimiento de requisitos	5
2.1.	Obtención de requisitos	5
2.2.	Especificación de casos de uso.....	9
2.3.	Validación de requisitos	16
3.	Identificación de subsistemas	16
3.1.	Identificación de subsistemas de análisis.....	17
3.2.	Integración de subsistemas de análisis	18
4.	Análisis de los casos de uso.....	18
4.1.	Identificación de clases asociadas a los casos de uso	18
5.	Análisis de clases	19
6.	Definición de interfaces de usuario	20
6.1.	Identificación de principios generales de la interfaz.....	21
6.2.	Especificación del comportamiento dinámico de la interfaz	23
6.3.	Especificación de formatos de impresión	25
7.	Especificación del plan de pruebas	26
8.	Aprobación del análisis de sistemas de información	27

• Análisis del sistema de información (ASI)

“El objetivo de este proceso es la obtención de una especificación detallada del sistema de información que satisfaga las necesidades de información de los usuarios y sirva de base para el posterior diseño del sistema.

Al ser MÉTRICA Versión 3 una metodología que cubre tanto desarrollos estructurados como orientados a objetos, las actividades de ambas aproximaciones están integradas en una estructura común.

En la primera actividad, Definición del Sistema (ASI 1), se lleva a cabo la descripción inicial del sistema de información, a partir de los productos generados en el proceso Estudio de Viabilidad del Sistema (EVS). Se delimita el alcance del sistema, se genera un catálogo de requisitos generales y se describe el sistema mediante unos modelos iniciales de alto nivel.

También se identifican los usuarios que participan en el proceso de análisis, determinando sus perfiles, responsabilidades y dedicaciones necesarias. Así mismo se elabora el plan de trabajo a seguir.

La definición de requisitos del nuevo sistema se realiza principalmente en la actividad Establecimiento de Requisitos (ASI 2). El objetivo de esta actividad es elaborar un catálogo de requisitos detallado, que permita describir con precisión el sistema de información, y que además sirva de base para comprobar que es completa la especificación de los modelos obtenidos en las actividades Identificación de Subsistemas de Análisis (ASI 3), Análisis de Casos de Uso (ASI 4), Análisis de Clases (ASI 5), Elaboración del Modelo de Datos (ASI 6), Elaboración del Modelo de Procesos (ASI 7) y Definición de Interfaces de Usuario (ASI 8).

Hay que hacer constar que estas actividades pueden provocar la actualización del catálogo, aunque no se refleja como producto de salida en las tareas de dichas actividades, ya que el objetivo de las mismas no es crear el catálogo sino definir modelos que soporten los requisitos.

Para la obtención de requisitos en la actividad Establecimiento de Requisitos (ASI 2) se toman como punto de partida el catálogo de requisitos y los modelos elaborados en la actividad Definición del Sistema (ASI 1), completándolos mediante sesiones de trabajo con los usuarios. Estas sesiones de trabajo tienen como objetivo reunir la información necesaria para obtener la especificación detallada del nuevo sistema. Las técnicas que ayudan a la recopilación de esta información pueden variar en función de las características del proyecto y los tipos de usuario a entrevistar. Entre ellas podemos citar las reuniones, entrevistas, *Joint Application Design (JAD)*, etc. Durante estas sesiones de trabajo se propone utilizar la especificación de los casos de uso como ayuda y guía en el establecimiento de requisitos. Esta técnica facilita la comunicación con los usuarios y en el análisis orientado a objetos constituye la base de la especificación. A continuación se identifican las facilidades que ha de proporcionar el sistema, y las restricciones a que está sometido en cuanto a rendimiento, frecuencia de tratamiento, seguridad y control de accesos, etc. Toda esta información se incorpora al catálogo de requisitos.

En la actividad Identificación de Subsistemas de Análisis (ASI 3), se estructura el sistema de información en subsistemas de análisis, para facilitar la especificación de los distintos modelos y la traza de requisitos. En paralelo, se generan los distintos modelos que sirven de base para el diseño. En el caso de análisis estructurado, se procede a la elaboración y descripción detallada del modelo de datos y de procesos, y en el caso de un análisis orientado a objetos, se

elaboran el modelo de clases y el de interacción de objetos, mediante el análisis de los casos de uso. Se especifican, asimismo, todas las interfaces entre el sistema y el usuario, tales como formatos de pantallas, diálogos, formatos de informes y formularios de entrada.

En la actividad Análisis de Consistencia y Especificación de Requisitos (ASI 9), se realiza la verificación y validación de los modelos, con el fin de asegurar que son:

- Completos, puesto que cada modelo obtenido contiene toda la información necesaria recogida en el catálogo de requisitos.
- Consistentes, ya que cada modelo es coherente con el resto de los modelos.
- Correctos, dado que cada modelo sigue unos criterios de calidad predeterminados en relación a la técnica utilizada, calidad de diagramas, elección de nombres, normas de calidad, etc.).

En la actividad Especificación del Plan de Pruebas (ASI 10), se establece el marco general del plan de pruebas, iniciándose su especificación, que se completará en el proceso Diseño del Sistema de Información (DSI).

La participación activa de los usuarios es una condición imprescindible para el análisis del sistema de información, ya que dicha participación constituye una garantía de que los requisitos identificados son comprendidos e incorporados al sistema y, por tanto, de que éste será aceptado. Para facilitar la colaboración de los usuarios, se pueden utilizar técnicas interactivas, como diseño de diálogos y prototipos, que permiten al usuario familiarizarse con el nuevo sistema y colaborar en la construcción y perfeccionamiento del mismo. En el siguiente gráfico se muestra la relación de actividades del proceso Análisis del Sistema de Información, tanto para desarrollos estructurados como para desarrollos orientados a objetos, distinguiendo las que se pueden realizar en paralelo de aquellas que han de realizarse secuencialmente”.

1. Definición del sistema

“Esta actividad tiene como objetivo efectuar una descripción del sistema, delimitando su alcance, estableciendo las interfaces con otros sistemas e identificando a los usuarios representativos. Las tareas de esta actividad se pueden haber desarrollado ya en parte en el proceso de Estudio de Viabilidad del Sistema (EVS), de modo que se parte de los productos obtenidos en dicho proceso para proceder a su adecuación como punto de partida para definir el sistema de información”.

Tareas propuestas para esta actividad:

- 1.1. Determinación del alcance del sistema
- 1.2. Especificación de estándares y normas

1.1. Determinación del alcance del sistema

Objetivo:

“En esta tarea se delimita el sistema de información, utilizando como punto de partida el modelo de procesos especificado en la descripción de la solución del proceso Estudio de Viabilidad del Sistema (EVS). Se indica qué procesos pertenecen al ámbito del Sistema de Información y se identifican las entidades externas al sistema que aportan o reciben información. Asimismo, se obtiene un modelo conceptual de datos identificando las entidades y relaciones que forman parte del sistema de información objeto de este análisis a partir del modelo abstracto de datos generado en la tarea Evaluación de Alternativas y Selección (EVS 6.2).

En el caso de análisis orientado a objetos, antes de la captura de requisitos empleando los casos de uso, puede ser conveniente establecer el contexto del sistema a partir del modelo de negocio obtenido en el proceso Estudio de Viabilidad del Sistema (EVS), y además, opcionalmente, del modelo de dominio. El modelo de negocio especifica los procesos a los que se quiere dar respuesta en el sistema de información, en forma de casos de uso de alto nivel, y el subconjunto de objetos del dominio requerido para ello”.

Modelo de negocio y dominio:

Como se ha comentado en el Anexo II – 3.1 el alcance del sistema cubrirá toda la operatoria de gestión de clubes, de las tres provincias de la Comunidad Valenciana, la gestión de los Socios de los respectivos clubes, y el tratamiento de las ligas que se juegan anualmente. El límite del alcance radica por tanto, en las entidades externas de club y socio.

Adicionalmente, el sistema de información contempla la actualización de equipos, una nueva gestión de BB.DD. y la inclusión de nuevos paquetes ofimáticos para dar soporte a las extensiones del sistema de gestión de ligas a implantar.

1.2. Especificación de estándares y normas

Objetivo:

“La realización de esta tarea permite considerar las referencias para el sistema de información en estudio, desde el punto de vista de estándares, normativas, leyes o recomendaciones, que deben tenerse en cuenta a lo largo de todo el proceso de desarrollo. El producto resultante se obtiene actualizando el catálogo de normas elaborado en el proceso Estudio de Viabilidad del Sistema (EVS), incorporando toda la información que, desde el punto de vista de la instalación, se considere necesario contemplar para la elaboración de los distintos productos del ciclo de vida”.

Catálogo de normas:

El catálogo de normas corresponderá a:

Metodología de desarrollo: El proyecto se realizará siguiendo las directrices de Métrica en su versión 3.

Protección de datos: Se establecerán los ficheros conforme a la Ley Orgánica de Protección de Datos de 1999, de esta manera los ficheros con datos de carácter personal tendrán que inscribirse en la Agencia de Protección de datos, responsabilidad de la FBCV, y su finalidad será la gestión de los contenidos y comunicaciones comerciales relacionadas con la actividad de la Federación.

Implementación: La implementación será llevada a cabo mediante el lenguaje C# .NET y el entorno de desarrollo de Microsoft[®] Visual Studio 2008. La BB.DD. será desarrollada en lenguaje Access. Gracias al empleo de esta tecnología, se consigue un desarrollo ágil, potente y orientado 100% a objetos.

Modelo de desarrollo: Prototipado evolutivo, se construye una serie de grandes versiones sucesivas de un producto. El modelo evolutivo asume que los requerimientos no son completamente conocidos al inicio del proyecto. Una vez se crea el primer desarrollo, los usuarios lo usan, y proveen retroalimentación a los desarrolladores. Basada en esta retroalimentación, la especificación de requerimientos es actualizada, y una segunda versión del producto es desarrollada y desplegada. El proceso se repite indefinidamente.

Modelo de BB.DD: Modelo entidad-relación de bases de datos

Análisis y diseño: Se usaran los modelos contenidos en UML 2.0¹. La técnica para la captura de requisitos potenciales de la actualización de requisitos será el modelo de Casos de Uso y para la especificación se usara la Guía IEEE Std. 830-98.²

Plataforma y Arquitectura: La plataforma del sistema será Microsoft[®] Windows Framework 3.5, y la arquitectura cliente de 1 capa.

2. Establecimiento de requisitos

“En esta actividad se lleva a cabo la definición, análisis y validación de los requisitos a partir de la información facilitada por el usuario, completándose el catálogo de requisitos obtenido en la actividad Definición del Sistema (ASI 1). El objetivo de esta actividad es obtener un catálogo detallado de los requisitos, a partir del cual se pueda comprobar que los productos generados en las actividades de modelización se ajustan a los requisitos de usuario.

Se propone como técnica de obtención de requisitos la especificación de los casos de uso de la orientación a objetos, siendo opcional en el caso estructurado. Dicha técnica ofrece un diagrama simple y una guía de especificación en las sesiones de trabajo con el usuario”.

Tareas propuestas para esta actividad:

- 2.1. Obtención de requisitos
- 2.2. Especificación de casos de uso
- 2.3. Validación de requisitos

2.1. Obtención de requisitos

Objetivo:

“En esta tarea comienza la obtención detallada de información mediante sesiones de trabajo con los usuarios, previamente identificados en la actividad Definición del Sistema (ASI 1). Se recoge información de los requisitos que debe cumplir el software. En la definición de los requisitos, que sirven de base para establecer los niveles de servicios del sistema, hay que tener en cuenta, si

¹ UML V2.0: Unified Modeling Language (28/04/2010) <http://www.uml.org/>

² Estándar IEEE para la especificación de requisitos de software (28/04/2010)

existen, las posibles restricciones del entorno, tanto hardware como software, que puedan afectar al sistema de información.

También se definen las prioridades que hay que asignar a los requisitos, considerando los criterios de los usuarios acerca de las funcionalidades a cubrir. Los principales tipos de requisitos que se deben especificar son, por ejemplo:

- Funcionales.
- Rendimiento.
- Seguridad.
- Implantación.
- Disponibilidad del sistema.

En el caso de orientación a objetos se especifican, además, los casos de uso asociados a los requisitos funcionales. Los casos de uso son una técnica de especificación de requisitos válida tanto en desarrollos estructurados como en orientación a objetos, aunque en este último caso se propone como técnica obligatoria al ser necesaria como referencia a lo largo de todo el ciclo de vida. En esta tarea se elabora el modelo de casos de uso, según las normas y estándares de la organización, identificando:

- Actores.
- Casos de uso.
- Breve descripción de cada caso de uso.

Catálogo de requisitos y casos de uso:

La representación de los casos de uso no coincide directamente con los requisitos funcionales, ya que la gestión de listados y actas tendrá una interfaz diferenciada y de acceso independiente, cuando en los requisitos están incluidas. No obstante, se agrupan por el catálogo de requisitos:

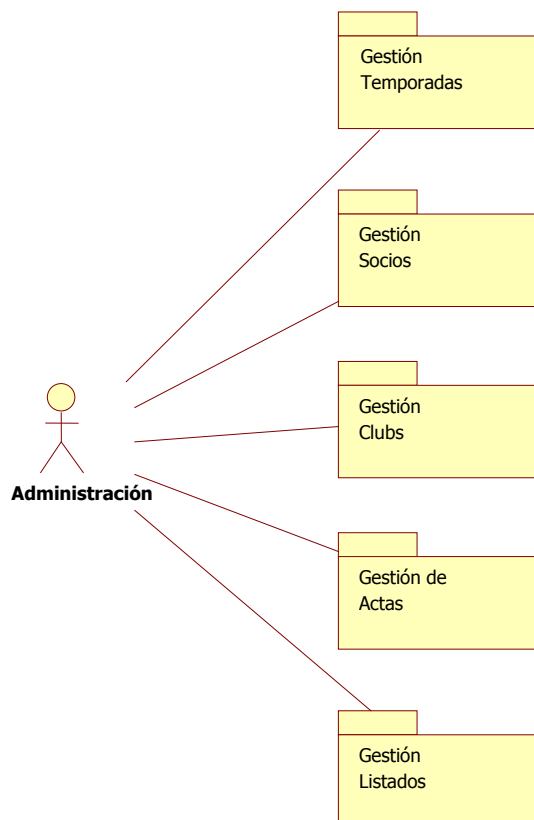


Figura 2.1. Diagrama de casos de uso

1) Requisito funcional Gestión de Temporadas

Crear nueva temporada:

- Entrada: Se introduce el año de la temporada, y para cada club, las ligas y divisiones en las que participará.
- Proceso: Se asignan los datos de entrada a un objeto de tipo temporada y se establece esa temporada como actual.
- Salida: Se generan los registros en la BB.DD. en las tablas de temporadas y clasificación.

Abrir temporada:

- Entrada: Se selecciona la temporada que se desea abrir de un desplegable.
- Proceso: Se asigna la temporada seleccionada al objeto temporada.
- Salida: Se actualiza la BB.DD. con el registro de la temporada actual para guardar el estado.

Añadir acta:

- Entrada: Se introducen los datos del acta.
- Proceso: Se genera un objeto de tipo acta con los datos introducidos.
- Salida: Se generan los registros de acta en la BB.DD. y se modifican los registros de clasificación.

Modificar acta:

- Entrada: Se modifican los datos representados en el dialogo.
- Proceso: Se asignan estos nuevos datos al objeto acta.
- Salida: Se actualizan en el registro los datos del acta y clasificaciones correspondientes.

Borrar acta:

- Entrada: Se selecciona del desplegable el acta a borrar.
- Proceso: Se asignan los datos del acta a borrar a un objeto.
- Salida: Se elimina en la BB.DD. el registro del acta en cuestión y se actualizan los datos de las clasificaciones.

Sancionar:

- Entrada: Se introducen los datos del club a sancionar y su puntuación a eliminar
- Proceso: Se calcula la puntuación y se asigna al objeto club
- Salida: Se actualizan en la BB.DD. los datos de la clasificación de ese club

Visualización de jornadas:

- Entrada: Se seleccionan las jornadas a visualizar
- Proceso:
- Salida: Se representan en pantalla las jornadas seleccionadas

Consultar clasificación:

- Entrada: Se selecciona una división y un grupo.
- Proceso:
- Salida: Se representan en pantalla la clasificación total de esa división y grupo

Requisito funcional 2) Gestión de Clubes

Añadir club:

- Entrada: Se introducen los datos del club a añadir
- Proceso: Se crea un nuevo objeto de tipo club y se asignan los datos introducidos
- Salida: Se crea un nuevo registro club en la BB.DD.

Editar club:

- Entrada: Se modifican los datos del club seleccionado
- Proceso: Se asignan estos nuevos datos al objeto club
- Salida: Se actualizan los datos del club en la BB.DD.

Borrar club:

- Entrada: Se selecciona el club a eliminar
- Proceso: Se asignan los datos del club a borrar a un objeto
- Salida: Se elimina el registro de la BB.DD. con el club, pero no con su clasificación ni sus actas

Imprimir club:

- Entrada: Se seleccionan los clubes a imprimir
- Proceso:
- Salida: Se imprimen los clubes seleccionados

Exportar club:

- Entrada: Se seleccionan los clubes a exportar
- Proceso:
- Salida: Se exportan los clubes seleccionados a Microsoft[®] Excel

Listado de clubes:

- Entrada: Se selecciona la división a listar
- Proceso:
- Salida: Se listan todos los clubes de la división seleccionada

Requisito funcional 3) Gestión de Socios**Añadir socio:**

- Entrada: Se introducen los datos del socio a añadir
- Proceso: Se crea un nuevo objeto de tipo socio y se asignan los datos introducidos
- Salida: Se crea un nuevo registro socio en la BB.DD.

Editar socio:

- Entrada: Se modifican los datos del socio seleccionado
- Proceso: Se asignan estos nuevos datos al objeto socio
- Salida: Se actualizan los datos del socio en la BB.DD.

Borrar socio:

- Entrada: Se selecciona el socio a eliminar
- Proceso: Se asignan los datos del socio a borrar a un objeto
- Salida: Se elimina el registro de la BB.DD. con el socio, pero no con su clasificación ni sus actas

Imprimir socio:

- Entrada: Se seleccionan los socios a imprimir
- Proceso:
- Salida: Se imprimen los carnets de los socios seleccionados

Exportar socio:

- Entrada: Se seleccionan los socios a exportar
- Proceso:
- Salida: Se exportan los socios seleccionados a Microsoft[®] Excel

Listado de socios:

- Entrada: Se selecciona la división a listar
- Proceso:
- Salida: Se listan todos los socios de la división seleccionada

Requisitos no funcionales:

Se describen los requisitos no funcionales especificados en el Anexo II - 3.3.3

- Seguridad
Como se ha comentado anteriormente, no existirá control de acceso más que el del propio sistema operativo.
- Normativa Interna
El programa debe adaptarse a los estatutos internos que la FBCV tiene respecto a la creación de ligas, desarrollo y datos de socios y clubes.
- Normativa Externa
Se hace necesaria la Inscripción de los ficheros en el Registro General de la Protección de Datos. Artículo 26 LOPD.
- Interfaz Grafica
La interfaz grafica debe ser sencilla y de fácil adaptación respecto al viejo sistema. Por tanto, se mantendrán las tabulaciones típicas de MSDOS[®].
- Velocidad del sistema
La velocidad del sistema debe ser rápida y mantenerse estable para una gestión media aproximada de 3000 socios y 100 clubes.
- Mantenibilidad
El mantenimiento del sistema tanto hardware como software debe resultar cómodo, ágil y de fácil reemplazo.
- Escalabilidad
Debido a que existe la posibilidad de ampliaciones futuras a corto plazo, es necesario que el sistema tenga un sólido núcleo que permita añadir nuevas funcionalidades sin perjudicar la funcionalidad básica. Además estas nuevas funcionalidades deberán ser desarrolladas en la misma plataforma.
- Compatibilidad
Se asegura una perfecta compatibilidad con sistemas Microsoft[®] Windows 7 / Vista tanto X64 como X86.

2.2. Especificación de casos de uso

Objetivo:

“Esta tarea es obligatoria en el caso de orientación a objetos, y opcional en el caso de análisis estructurado, como apoyo a la obtención de requisitos. El objetivo de esta tarea es especificar cada caso de uso identificado en la tarea anterior, desarrollando el escenario”.

Para completar los casos de uso, es preciso especificar información relativa a:

- Descripción del escenario, es decir, cómo un actor interactúa con el sistema, y cuál es la respuesta obtenida.
- Precondiciones y post condiciones.
- Identificación de interfaces de usuario.
- Condiciones de fallo que afectan al escenario, así como la respuesta del sistema (escenarios secundarios).”

Especificación de casos de uso:

1 – Gestión de temporadas

Caso de Uso	Crear nueva temporada.
Descripción	Se crea una nueva temporada que englobará un periodo de competición de septiembre del año seleccionado a junio del año siguiente.
Actor principal	Personal administración.
Actor secundario	-
Precondiciones	Debe existir algún club introducido en la BB.DD.
Post condiciones	Se crean los registros en la BB.DD. con los datos de una nueva temporada, los datos de clasificaciones vacios, y se almacena esa temporada como actual.
Flujo de Eventos	1 - El actor principal selecciona la opción nueva temporada del menú. 2 - Se selecciona el periodo en el que se desarrollará la temporada. 3 - Se selecciona para cada club, las divisiones y grupos donde jugará. 4 - Se crean los registros en la BB.DD.
Restricción síncrona	En 4) Si no hay clubes seleccionados, se avisa al usuario pidiendo que los seleccione.
Restricción asíncrona	- En cualquier momento el usuario puede cerrar la ventana de dialogo. - En cualquier momento el usuario puede cerrar el programa.

Caso de Uso	Abrir temporada.
Descripción	Se abre una temporada ya creada y se establece como temporada actual.
Actor principal	Personal administración.
Actor secundario	-
Precondiciones	Debe existir alguna temporada introducida en la BB.DD.
Post condiciones	Se guarda en la BB.DD. la temporada seleccionada como actual.
Flujo de Eventos	1 - El actor principal selecciona la opción abrir temporada del menú. 2 - Se hace doble clic en la temporada que se desea abrir. 3 - Se actualiza en la BB.DD. la temporada seleccionada como la actual.
Restricción síncrona	-
Restricción asíncrona	- En cualquier momento el usuario puede cerrar la ventana de dialogo. - En cualquier momento el usuario puede cerrar el programa.

Caso de Uso	Añadir acta.
Descripción	Una vez recibida el acta en formato impreso, comprobada la firma, se introduce en el sistema.
Actor principal	Personal administración.
Actor secundario	-
Precondiciones	Existe una temporada abierta como actual.
Post condiciones	Se introduce el acta en las tablas de la BB.DD.
Flujo de Eventos	1 - El actor principal selecciona la opción nueva acta del menú 2 - El personal de administración introduce los datos del acta en el dialogo 3 - Se introducen los datos del acta en la tabla de actas, y se modifican los datos de clasificación en las tablas de clasificación por club y jugadores.
Restricción síncrona	En 2) Si no se ha seleccionado la división y el grupo antes de escribir los clubes, el programa avisa que no puede continuar. En 2) Si el club local es igual que el club visitante, el sistema avisa del error. En 2) Si el jugador local es igual que el jugador visitante, el sistema avisa del error.
Restricción asíncrona	- En cualquier momento el usuario puede cerrar la ventana de dialogo. - En cualquier momento el usuario puede cerrar el programa.

Caso de Uso	Modificar acta
Descripción	Se abre un acta en modo edición para su modificación.
Actor principal	Personal administración.
Actor secundario	-
Precondiciones	Debe existir algún acta en la BB.DD.
Post condiciones	El acta modificada queda guardada en la BB.DD, así como las tablas de clasificación de jugadores y clubes.
Flujo de Eventos	1 - El actor principal selecciona la opción modificar acta del menú. 2 - Se debe seleccionar un acta del desplegable. 3 - El personal de administración modifica los datos del acta en el dialogo 4 - Se modifican los datos del acta en la tabla de actas, y se modifican los datos de clasificación en las tablas de clasificación por club y jugadores.
Restricción síncrona	En 3) Si no se ha seleccionado la división y el grupo antes de escribir los clubes, el programa avisa que no puede continuar. En 3) Si el club local es igual que el club visitante, el sistema avisa del error. En 3) Si el jugador local es igual que el jugador visitante, el sistema avisa del error.
Restricción asíncrona	- En cualquier momento el usuario puede cerrar la ventana de dialogo. - En cualquier momento el usuario puede cerrar el programa.

Caso de Uso	Borrar acta
Descripción	Se selecciona un acta a eliminar.
Actor principal	Personal administración.
Actor secundario	-
Precondiciones	Debe existir algún acta en la BB.DD.
Post condiciones	El acta seleccionada queda eliminada de la BB.DD.
Flujo de Eventos	1 - El actor principal selecciona la opción borrar acta del menú. 2 - Se debe seleccionar un acta del desplegable. 3 - El personal de administración elimina el acta seleccionada 4 - Se eliminan los datos del acta en la tabla de actas, y se modifican los datos de clasificación en las tablas de clasificación por club y jugadores.
Restricción síncrona	-
Restricción asíncrona	- En cualquier momento el usuario puede cerrar la ventana de dialogo. - En cualquier momento el usuario puede cerrar el programa.

Caso de Uso	Sancionar
Descripción	Se sanciona a un club por algún motivo reflejado en un acta de sanción
Actor principal	Personal administración.
Actor secundario	-
Precondiciones	Debe existir algún club en la BB.DD.
Post condiciones	
Flujo de Eventos	1 - El actor principal selecciona la opción sancionar del menú. 2 - Se debe seleccionar un club del desplegable. 3 - El personal de administración introduce el club a sancionar y los puntos 4 - Se restan los datos introducidos en las tablas de clasificación por club y jugadores.
Restricción síncrona	-
Restricción asíncrona	- En cualquier momento el usuario puede cerrar la ventana de dialogo. - En cualquier momento el usuario puede cerrar el programa.

Caso de Uso	Visualización de jornadas.
Descripción	Se muestra un Datagridview con las jornadas de la temporada.
Actor principal	Personal administración.
Actor secundario	-
Precondiciones	Hay una temporada abierta.
Post condiciones	Se representa el Datagridview en pantalla con las jornadas.
Flujo de Eventos	1 - El personal de administración selecciona la opción de visualizar jornadas. 2 - Se muestran en pantalla los datos de todas las jornadas.
Restricción síncrona	En 2) El usuario puede modificar los criterios de orden de la jornada pulsando en la cabecera del Datagridview. En 3) El usuario puede mandar a imprimir los datos del Datagridview.
Restricción asíncrona	- En cualquier momento el usuario puede cerrar la ventana de dialogo. - En cualquier momento el usuario puede cerrar el programa.

Caso de Uso	Consultar clasificación.
Descripción	Se consulta la clasificación por clubes o jugadores.
Actor principal	Personal administración.
Actor secundario	-
Precondiciones	Hay una temporada abierta.
Post condiciones	Se representa el Datagridview en pantalla con la clasificación seleccionada
Flujo de Eventos	1 - El personal de administración selecciona la opción de consultar clasificación. 2 - Se selecciona el tipo de consulta 2 - Se muestran en pantalla los datos de la clasificación filtrados por la selección.
Restricción síncrona	En 2) El usuario puede modificar los criterios de orden de la jornada pulsando en la cabecera del Datagridview.
Restricción asíncrona	- En cualquier momento el usuario puede cerrar la ventana de dialogo. - En cualquier momento el usuario puede cerrar el programa.

2- Gestión de clubes

Caso de Uso	Añadir club.
Descripción	Se introduce un nuevo club en la BB.DD.
Actor principal	Personal administración.
Actor secundario	-
Precondiciones	-
Post condiciones	Se crean los registros en la BB.DD. con los datos del nuevo club.
Flujo de Eventos	1 - El personal de administración selecciona la opción de introducir nuevo club. 2 - Se introducen los datos del nuevo club. 3 - Se crean los registros en la BB.DD.
Restricción síncrona	En 2) Si los datos introducidos tienen formato incorrecto se avisa al usuario y se marca el Textbox para que sea modificado.
Restricción asíncrona	- En cualquier momento el usuario puede cerrar la ventana de dialogo. - En cualquier momento el usuario puede cerrar el programa.

Caso de Uso	Modificar club.
Descripción	Se abre un club en modo edición para su modificación.
Actor principal	Personal administración.
Actor secundario	-
Precondiciones	Debe existir algún club en la BB.DD.
Post condiciones	El club modificado queda guardado en la BB.DD.
Flujo de Eventos	1 - El actor principal selecciona la opción modificar club del menú. 2 - Se debe seleccionar un club del desplegable. 3 - El personal de administración modifica los datos del club en el dialogo. 4 - Se modifican los datos del club en la tabla de clubes y clasificación.
Restricción síncrona	En 3) Si los datos introducidos tienen formato incorrecto se avisa al usuario y se marca el Textbox para que sea modificado.
Restricción asíncrona	- En cualquier momento el usuario puede cerrar la ventana de dialogo. - En cualquier momento el usuario puede cerrar el programa.

Caso de Uso	Borrar club.
Descripción	Se selecciona un club a eliminar.
Actor principal	Personal administración.
Actor secundario	-
Precondiciones	Se debe seleccionar un club del desplegable.
Post condiciones	El club seleccionado queda eliminado de la BB.DD. de clubes.
Flujo de Eventos	1 - El actor principal selecciona la opción borrar club del menú. 2 - Se debe seleccionar un club del desplegable. 3 - El personal de administración elimina el club seleccionado. 4 - Se eliminan los datos del club en la tabla de clubes, pero no se modifican los datos de clasificación en las tablas de clasificación por club y jugadores.
Restricción síncrona	-
Restricción asíncrona	- En cualquier momento el usuario puede cerrar la ventana de dialogo. - En cualquier momento el usuario puede cerrar el programa.

Caso de Uso	Imprimir club.
Descripción	Se envía a imprimir el club o clubes seleccionados.
Actor principal	Personal administración.
Actor secundario	-
Precondiciones	Debe existir algún club en la BB.DD.
Post condiciones	Los datos seleccionados son enviados a la Impresora correctamente.
Flujo de Eventos	1 - El actor principal selecciona la opción imprimir club del menú. 2 - Se debe seleccionar algún club del desplegable. 3 - Se configuran las opciones de impresión. 4- Se lanza la impresión.
Restricción síncrona	En 2) Si no se selecciona ningún club, no se permite continuar.
Restricción asíncrona	- En cualquier momento el usuario puede cerrar la ventana de dialogo. - En cualquier momento el usuario puede cerrar el programa.

Caso de Uso	Exportar club
Descripción	Se envía a Excel el club o clubes seleccionados.
Actor principal	Personal administración.
Actor secundario	-
Precondiciones	-
Post condiciones	Debe existir algún club en la BB.DD.
Flujo de Eventos	1 - El actor principal selecciona la opción exportar club del menú. 2 - Se debe seleccionar algún club del desplegable. 3 - Se lanza el proceso Excel junto con los datos de los clubes seleccionados.
Restricción síncrona	En 2) Si no se selecciona ningún club, no se permite continuar. En 3) Si no se encuentra el software Ms.® Excel instalado, se mostrará un mensaje de error.
Restricción asíncrona	- En cualquier momento el usuario puede cerrar la ventana de dialogo. - En cualquier momento el usuario puede cerrar el programa.

Caso de Uso	Listar clubes
Descripción	Se consulta el listado de clubes
Actor principal	Personal administración.
Actor secundario	-
Precondiciones	Debe existir algún club en la BB.DD.
Post condiciones	Se representa el Datagridview en pantalla con los clubes seleccionados
Flujo de Eventos	1 - El personal de administración selecciona la opción de listar clubes. 2 - Se selecciona el tipo de consulta 2 - Se muestran en pantalla los datos de los clubes filtrados por la selección.
Restricción síncrona	En 2) El usuario puede modificar los criterios de orden de los clubes pulsando en la cabecera del Datagridview. En 3) El usuario puede mandar a imprimir los datos del Datagridview.
Restricción asíncrona	- En cualquier momento el usuario puede cerrar la ventana de dialogo. - En cualquier momento el usuario puede cerrar el programa.

3 – Gestión de socios

Caso de Uso	Añadir socio.
Descripción	Se introduce un nuevo socio en la BB.DD.
Actor principal	Personal administración.
Actor secundario	-
Precondiciones	Existe algún club previamente introducido en la BB.DD.
Post condiciones	Se crean los registros en la BBDD con los datos del nuevo socio.
Flujo de Eventos	1 - El personal de administración selecciona la opción de introducir nuevo socio. 2 - Se introducen los datos del nuevo socio. 3 - Se crean los registros en la BB.DD.
Restricción síncrona	En 2) Si los datos introducidos tienen formato incorrecto se avisa al usuario y se marca el textbox para que sea modificado.
Restricción asíncrona	- En cualquier momento el usuario puede cerrar la ventana de dialogo - En cualquier momento el usuario puede cerrar el programa

Caso de Uso	Modificar socio.
Descripción	Se abre un socio en modo edición para su modificación.
Actor principal	Personal administración.
Actor secundario	-
Precondiciones	Debe existir algún socio en la BB.DD.
Post condiciones	El socio modificado queda guardado en la BB.DD.
Flujo de Eventos	1 - El actor principal selecciona la opción modificar socio del menú. 2 - Se debe seleccionar un socio del desplegable. 3 - El personal de administración modifica los datos del socio en el dialogo. 4 - Se modifican los datos del socio en la tabla de socios y clasificación
Restricción síncrona	En 3) Si los datos introducidos tienen formato incorrecto se avisa al usuario y se marca el textbox para que sea modificado.
Restricción asíncrona	- En cualquier momento el usuario puede cerrar la ventana de dialogo - En cualquier momento el usuario puede cerrar el programa

Caso de Uso	Borrar socio.
Descripción	Se selecciona un socio a eliminar.
Actor principal	Personal administración.
Actor secundario	-
Precondiciones	Se debe seleccionar un socio del desplegable.
Post condiciones	El socio seleccionado queda eliminado de la BB.DD de socios.
Flujo de Eventos	1 - El actor principal selecciona la opción borrar socio del menú. 2 - Se debe seleccionar un socio del desplegable. 3 - El personal de administración elimina el socio seleccionado. 4 - Se eliminan los datos del socio en la tabla de socios, pero no se modifican los datos de clasificación en las tablas de clasificación por club y jugadores.
Restricción síncrona	-
Restricción asíncrona	- En cualquier momento el usuario puede cerrar la ventana de dialogo - En cualquier momento el usuario puede cerrar el programa

Caso de Uso	Imprimir socio.
Descripción	Se envía a imprimir los carnets del socio o socios seleccionados.
Actor principal	Personal administración.
Actor secundario	-
Precondiciones	Debe existir algún socio en la BB.DD.
Post condiciones	Los datos seleccionados son enviados a la Impresora correctamente.
Flujo de Eventos	1 - El actor principal selecciona la opción imprimir carnets de socio del menú. 2 - Se debe seleccionar algún socio del desplegable. 3 - Se configuran las opciones de impresión. 4 - Se lanza la utilidad de reportes. 5- Se mandan imprimir los reportes.
Restricción síncrona	En 2) Si no se selecciona ningún socio, no se permite continuar.
Restricción asíncrona	- En cualquier momento el usuario puede cerrar la ventana de dialogo. - En cualquier momento el usuario puede cerrar el programa.

Caso de Uso	Exportar socio.
Descripción	Se envía a Excel el socio o socios seleccionados.
Actor principal	Personal administración.
Actor secundario	-
Precondiciones	-
Post condiciones	Debe existir algún socio en la BB.DD.
Flujo de Eventos	1 - El actor principal selecciona la opción exportar socio del menú. 2 - Se debe seleccionar algún socio del desplegable. 3 - Se lanza el proceso Excel junto con los datos de los socios seleccionados.
Restricción síncrona	En 2) Si no se selecciona ningún socio, no se permite continuar. En 3) Si no se encuentra el software Ms.® Excel instalado, se mostrará un mensaje de error.
Restricción asíncrona	- En cualquier momento el usuario puede cerrar la ventana de dialogo. - En cualquier momento el usuario puede cerrar el programa.

Caso de Uso	Listado socios.
Descripción	Se consulta el listado de socios
Actor principal	Personal administración.
Actor secundario	-
Precondiciones	Debe existir algún socio en la BB.DD.
Post condiciones	Se representa el Datagridview en pantalla con los socios seleccionados
Flujo de Eventos	1 - El personal de administración selecciona la opción de listar socios. 2 - Se selecciona el tipo de consulta 2 - Se muestran en pantalla los datos de los socios filtrados por la selección.
Restricción síncrona	En 2) El usuario puede modificar los criterios de orden de los socios pulsando en la cabecera del Datagridview. En 3) El usuario puede mandar a imprimir los datos del Datagridview.
Restricción asíncrona	- En cualquier momento el usuario puede cerrar la ventana de dialogo. - En cualquier momento el usuario puede cerrar el programa.

2.3. Validación de requisitos

Objetivo:

“Mediante esta tarea, los usuarios confirman que los requisitos especificados en el catálogo de requisitos, así como los casos de uso, son válidos, consistentes y completos”.

Validación:

Una vez presentados los requisitos del catalogo al responsable del proyecto del cliente, se confirman que son correctos y se validan para posterior ejecución.

3. Identificación de subsistemas

“El objetivo de esta actividad, común tanto para análisis estructurado como para análisis orientado a objetos, es facilitar el análisis del sistema de información llevando a cabo la descomposición del sistema en subsistemas. Se realiza en paralelo con el resto de las actividades de generación de modelos del análisis. Por tanto, se asume la necesidad de una

realimentación y ajuste continuo con respecto a la definición de los subsistemas, sus dependencias y sus interfaces”.

Tareas propuestas para esta actividad:

- 3.1. Determinación de subsistemas de análisis
- 3.2. Integración de subsistemas de análisis

3.1. Identificación de subsistemas de análisis

Objetivo:

“La descomposición del sistema en subsistemas debe estar, principalmente, orientada a los procesos de negocio, aunque también es posible adoptar otros criterios lógicos. Entre los criterios que pueden ayudar a su identificación, se encuentran los siguientes:

- Homogeneidad de procesos.
- Servicios comunes.
- Prioridad.
- Afinidad de requisitos.
- Localización geográfica.

En análisis estructurado, los subsistemas coinciden habitualmente con el primer nivel de descomposición del Diagrama de Flujo de Datos (diagrama 0), de modo que llevan implícita la definición de dependencia y de interfaz.

En análisis orientado a objetos, se identifican y definen las dependencias entre subsistemas analizando los elementos compartidos entre ellos o las interfaces entre subsistemas. En el caso de que se decida abstraer un subsistema para su análisis como una unidad con una funcionalidad concreta, se puede, opcionalmente, definir la interfaz de dicho subsistema para poder delimitar su comportamiento y utilización en el modelo general del sistema. Por tanto, se establece como obligatoria la asociación entre subsistemas indicando sólo la dependencia. Además, opcionalmente, se propone la especificación de la interfaz de subsistemas de análisis, y la definición del comportamiento del sistema. En ambos casos, se asignan los requisitos y casos de uso a cada uno de los subsistemas identificados, actualizando el catálogo de requisitos”.

Descripción de subsistemas de análisis:

Como se ha comentado el proyecto se centra en único sistema de gestión con tres subsistemas. La dependencia entre ellos se muestra a continuación:

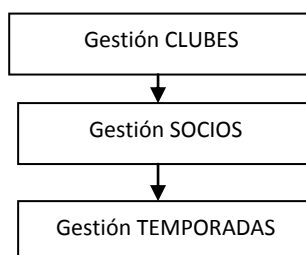


Figura 3.1. Diagrama de dependencias

3.2. Integración de subsistemas de análisis

Objetivo:

“El objetivo de esta tarea es la coordinación en la elaboración de los distintos modelos de análisis de cada subsistema, asegurando la ausencia de duplicidad de elementos y la precisión en la utilización de los términos del glosario. Esta tarea se realiza en paralelo con el resto de las actividades de elaboración de modelos del análisis, y permite tener una visión global y unificada de los distintos modelos.

Como consecuencia de la coordinación de modelos, se pueden identificar elementos comunes con posible implicación en la propia definición de subsistemas y en sus dependencias o interfaces”.

Integración:

Los subsistemas descritos anteriormente están totalmente integrados en un único sistema software por lo que no existirán duplicidades de ningún tipo.

4. Análisis de los casos de uso

“El objetivo de esta actividad, que sólo se realiza en el caso de Análisis Orientado a Objetos, es identificar las clases cuyos objetos son necesarios para realizar un caso de uso y describir su comportamiento mediante la interacción dichos objetos. Esta actividad se lleva a cabo para cada uno de los casos de uso contenidos en un subsistema de los definidos en la actividad Identificación de Subsistemas de Análisis (ASI 3). Las tareas de esta actividad no se realizan de forma secuencial sino en paralelo, con continuas realimentaciones entre ellas y con las realizadas en las actividades Establecimiento de Requisitos (ASI 2), Identificación de Subsistemas de Análisis (ASI 3), Análisis de Clases (ASI 5) y Definición de Interfaces de Usuario (ASI 8)”.

Tareas propuestas para esta actividad:

4.1. Identificación de clases asociadas a los casos de uso

4.1. Identificación de clases asociadas a los casos de uso

Objetivo:

“En esta tarea se comienzan a identificar los objetos necesarios para realizar el caso de uso, basándose en la especificación que tenemos del mismo. A partir del estudio del caso de uso, se extrae una lista de objetos candidatos a ser clases. Es posible que, inicialmente, no se disponga de la información necesaria para identificar todas, por lo que se hace una primera aproximación que se va refinando posteriormente, durante esta actividad y en el proceso de diseño. Además, algunos de los objetos representan mejor la información del sistema si se les identifica como atributos en vez de como clases. Para poder diferenciarlas, es necesario estudiar el comportamiento de esos objetos en el diagrama de interacción y además se debe tener en cuenta una serie de reglas, como puede ser el suprimir palabras no pertinentes, con significados vagos o sinónimos. Una vez definidas cada una de las clases, se incorporan al modelo de clases de la actividad Análisis de Clases (ASI 5), donde se identifican sus atributos, responsabilidades y relaciones”.

Identificación de clases asociadas:

Tras un primer análisis, las clases identificadas asociadas a los casos de uso son:

- Socio
- Club
- Temporada
- Acta

5. Análisis de clases

“El objetivo de esta actividad que sólo se realiza en el caso de Análisis Orientado a Objetos es describir cada una de las clases que ha surgido, identificando las responsabilidades que tienen asociadas, sus atributos, y las relaciones entre ellas. Para esto, se debe tener en cuenta la normativa establecida en la tarea Especificación de Estándares y Normas (ASI 1.3), de forma que el modelo de clases cumpla estos criterios, con el fin de evitar posibles inconsistencias en el diseño.

Teniendo en cuenta las clases identificadas en la actividad Análisis de los Casos de Uso (ASI 4), se elabora el modelo de clases para cada subsistema. A medida que avanza el análisis, dicho modelo se va completando con las clases que vayan apareciendo, tanto del estudio de los casos de uso, como de la interfaz de usuario necesaria para el sistema de información”.

Análisis de clases:

- Socio: Contiene los datos de los socios introducidos en el sistema.
- Club: Contiene los datos de los clubes introducidos en el sistema.
- Temporada: Contiene los datos de las temporadas introducidas en el sistema.
- Acta: Contiene las actas de las temporadas introducidas en el sistema.
- Clasificación Jugador: Contiene los datos de la competición del socio para cada temporada.
- Clasificación Club: Contiene los datos de la competición del club para cada temporada.
- Temporada Club: Contiene los datos de la relación entre temporada y club.

Se genera a continuación el diagrama de clases asociado:

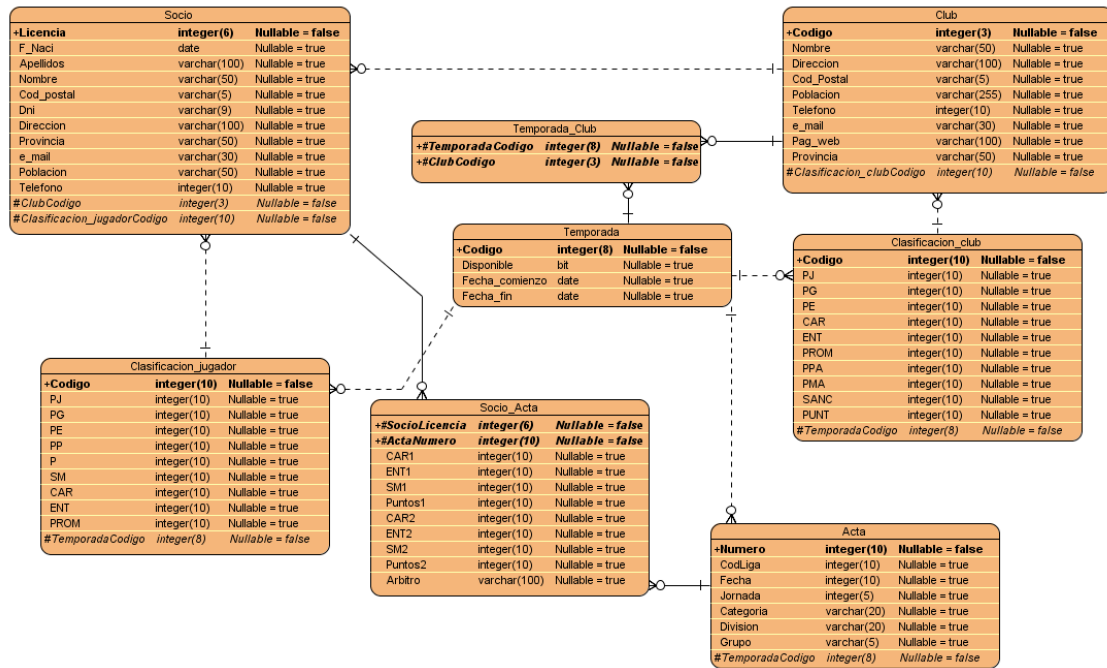


Figura 5.1. Diagrama de clases

6. Definición de interfaces de usuario

“En esta actividad se especifican las interfaces entre el sistema y el usuario: formatos de pantallas, diálogos, e informes, principalmente. El objetivo es realizar un análisis de los procesos del sistema de información en los que se requiere una interacción del usuario, con el fin de crear una interfaz que satisfaga todos los requisitos establecidos, teniendo en cuenta los diferentes perfiles a quienes va dirigido.

Al comienzo de este análisis es necesario seleccionar el entorno en el que es operativa la interfaz, considerando estándares internacionales y de la instalación, y establecer las directrices aplicables en los procesos de diseño y construcción. El propósito es construir una interfaz de usuario acorde a sus necesidades, flexible, coherente, eficiente y sencilla de utilizar, teniendo en cuenta la facilidad de cambio a otras plataformas, si fuera necesario.

Se identifican los distintos grupos de usuarios de acuerdo con las funciones que realizan, conocimientos y habilidades que poseen, y características del entorno en el que trabajan. La identificación de los diferentes perfiles permite conocer mejor las necesidades y particularidades de cada uno de ellos.

Asimismo, se determina la naturaleza de los procesos que se llevan a cabo (en lotes o en línea). Para cada proceso en línea se especifica qué tipo de información requiere el usuario para

completar su ejecución realizando, para ello, una descomposición en diálogos que refleje la secuencia de la interfaz de pantalla tipo carácter o pantalla gráfica.

Finalmente, se define el formato y contenido de cada una de las interfaces de pantalla especificando su comportamiento dinámico. Se propone un flujo de trabajo muy similar para desarrollos estructurados y orientados a objetos, coincidiendo en la mayoría de las tareas, si bien es cierto que en orientación a objetos, al identificar y describir cada escenario en la especificación de los casos de uso, se hace un avance muy significativo en la toma de datos para la posterior definición de la interfaz de usuario”.

Como resultado de esta actividad se genera la especificación de interfaz de usuario, como producto que engloba los siguientes elementos:

- Principios generales de la interfaz.
- Catálogo de perfiles de usuario.
- Descomposición funcional en diálogos.
- Catálogo de controles y elementos de diseño de interfaz de pantalla.
- Formatos individuales de interfaz de pantalla.
- Modelo de navegación de interfaz de pantalla.
- Formatos de impresión.
- Prototipo de interfaz interactiva.
- Prototipo de interfaz de impresión.

Tareas propuestas para esta actividad:

- 6.1. Identificación de principios generales de la interfaz
- 6.2. Especificación del comportamiento dinámico de la interfaz
- 6.3. Especificación de formatos de impresión

6.1. Identificación de principios generales de la interfaz

Objetivo:

“El objetivo de esta tarea es especificar los estándares, directrices y elementos generales a tener en cuenta en la definición de la interfaz de usuario, tanto para la interfaz interactiva (gráfica o carácter), como para los informes y formularios impresos. En primer lugar, se selecciona el entorno de la interfaz interactiva (gráfico, carácter, etc.), siguiendo estándares internacionales y de la instalación, y se determinan los principios de diseño de la interfaz de usuario, contemplando:

- Directrices generales en cuanto a la interfaz y aspectos generales de interacción.
- Principios de composición de pantallas y criterios de ubicación de los distintos elementos dentro de cada formato.
- Normas para los mensajes de error y aviso, codificación, presentación y comportamientos.
- Normas para la presentación de ayudas.

Hay que establecer criterios similares para la interfaz impresa:

- Directrices generales.
- Principios de composición de informes y formularios.
- Normas de elaboración, distribución y salvaguarda de la información.

Principios generales de la interfaz:

Se establecen las siguientes reglas básicas para la generación de la interfaz:

- Fuente de carácter de todo el programa: **Segoe UI; 9pt**
- Ventana principal contenedora:
 - Padre del resto de diálogos.
 - Apertura inicial maximizada.
 - Se muestra en la barra de tareas.
 - Se muestra su icono.
 - No modal.
 - Bordes de tamaño modificable.
 - Botones de Min, Max, Cerrar habilitados.
- Diálogos de interacción general:
 - Hijos de la ventana principal.
 - Apertura inicial maximizada.
 - No se muestran en la barra de tareas.
 - No se muestra su icono.
 - No modal.
 - Bordes de tamaño modificable.
 - Botones de Min, Max, Cerrar habilitados.
- Representación de listados: con el componente DataGridView.
- Si existe opción de exportación e impresión, siempre alineado a la derecha.
- Si existe opción de filtrado, siempre alineado a la izquierda.
- Todos los diálogos contendrán título en su barra superior.
- Se asigna un tamaño fijo para los botones ACEPTAR y CANCELAR en todos los formularios.
- Todos los botones podrán ser accedidos mediante atajos de teclado ALT+1ª letra del botón.
- Campos de texto y edición, siempre alineados a la izquierda.
- Menús incrustados en diálogos, siempre con marco descriptivo.
- Todos los cuadros de texto asociados a BB.DD. se rellenarán automáticamente al introducir los primeros caracteres.
- Las tabulaciones se harán según el estudio del análisis preliminar del software anterior.

Además de estas características generales, se asume específicamente que:

- Cada opción del menú permitirá acceder a una función de negocio.
- Una opción del menú permitirá restablecer los diálogos mostrados en pantalla.
- Se establece el diálogo ayuda como predeterminado.
- Se mantendrá visible la temporada actual en todo momento en la barra superior.

Como se ha comentado anteriormente, los diálogos no variarán según el perfil del usuario que trabaje con ellos.

6.2. Especificación del comportamiento dinámico de la interfaz

Objetivo:

“El objetivo de esta tarea es definir los flujos entre los distintos formatos de interfaz de pantalla, y también dentro del propio formato. Este comportamiento se describe mediante un modelo de navegación de interfaz de pantalla.

Para cada formato individual de pantalla o ventana, se establece la entrada lógica de los datos y las reglas de validación, incluyendo dependencia de valores (reflejo de los requisitos de validación de sistema).

Se analiza y determina la secuencia de acciones específicas para completar cada diálogo, tal y como se ejecuta en el ámbito de la interfaz, así como las condiciones que se deben cumplir para su inicio, y las posibles restricciones durante su ejecución. El comportamiento está dirigido y representado por los controles y los eventos que provocan su activación.

Se identifican aquellos diálogos o formatos considerados críticos para el correcto funcionamiento del sistema, basándose en el número de usuarios, frecuencia de uso, datos implicados, alcance de las funciones asociadas al diálogo, diálogos comunes a diferentes funciones, marco de seguridad establecido en los requisitos del sistema, etc.”.

Comportamiento dinámico:

Se comenta a continuación un ciclo completo de diálogos para la correcta gestión de la información del sistema.

Pantalla principal de la aplicación:

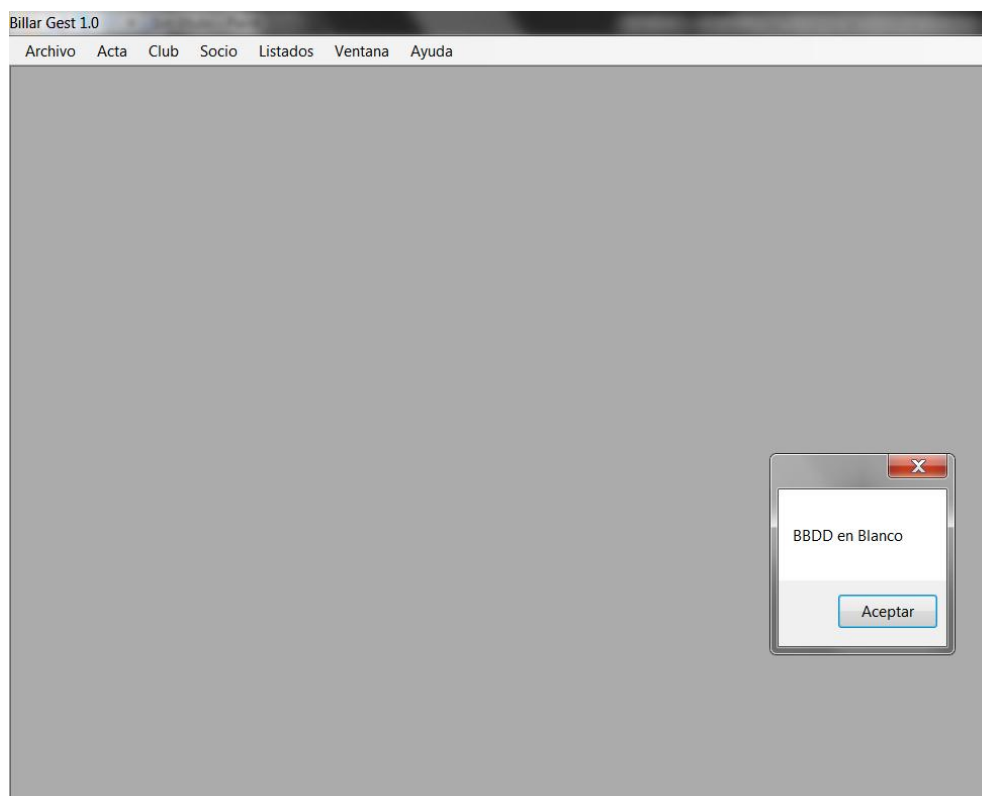


Figura 6.1. Ventana principal

El sistema arranca con una BB.DD. en blanco y sin ninguna temporada dada como actual. Es necesario introducir clubes (2) y jugadores (8) en el sistema para crear una temporada nueva, o abrir una temporada actual. El menú SOCIO permanece desactivado mientras no se introduzca ningún club.

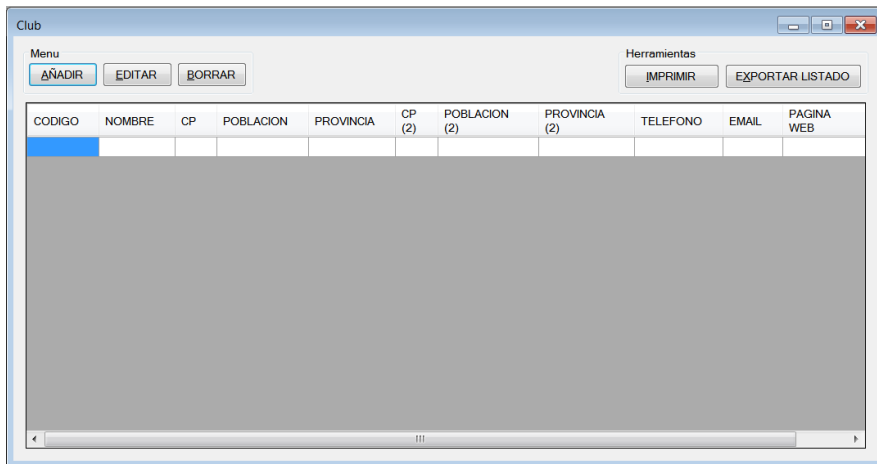


Figura 6.2. Ventana de lista de clubes

En el diálogo de gestión de clubes, se añade un club nuevo:

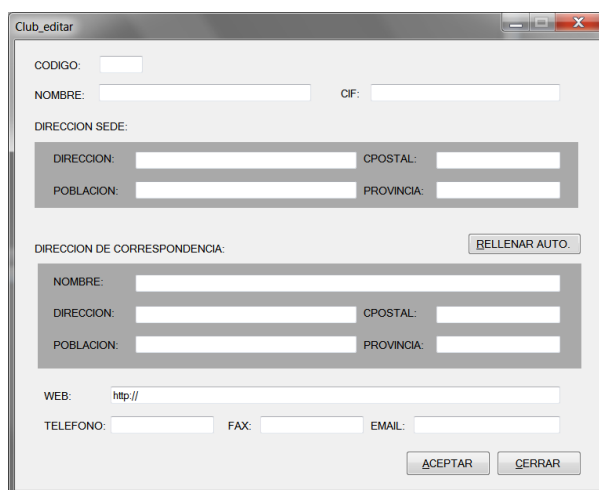


Figura 6.3. Ventana de detalle de club

Una vez se introduce un club en el sistema, se puede añadir un socio nuevo de la misma forma:

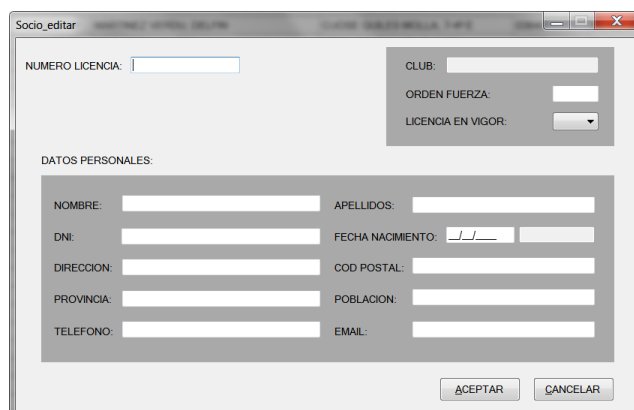


Figura 6.4. Ventana de detalle de socios

Una vez introducidos clubes y socios, ya se puede crear una nueva temporada:

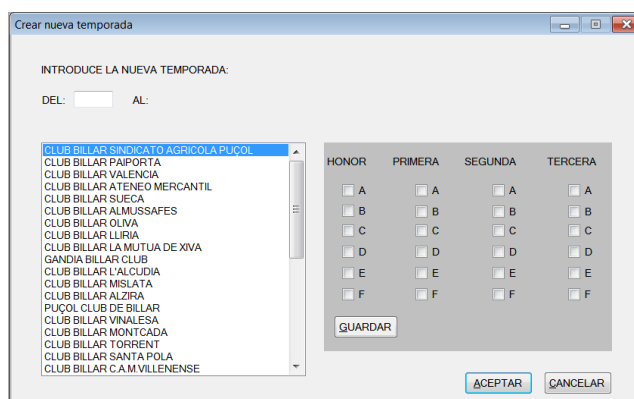


Figura 6.5. Ventana de creación temporadas

Se seleccionan los clubes y la temporada y el sistema crea las tablas de clasificación. A partir de ahora, ya se puede introducir actas en las jornadas de la temporada.

Se introducen las actas:

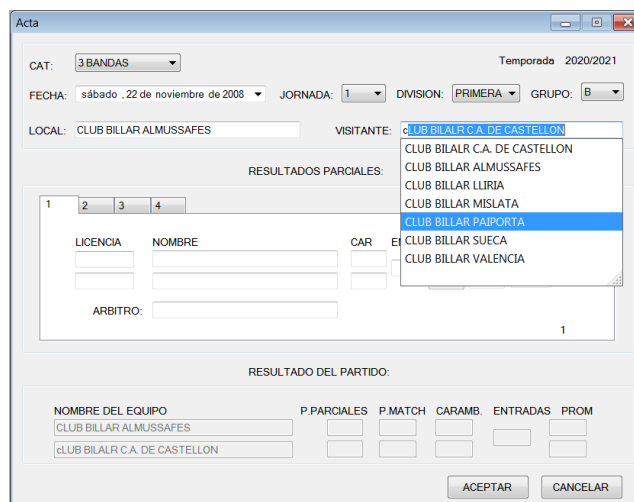


Figura 6.6. Ventana de alta de acta

Una vez introducidas actas en el sistema, ya es posible consultar los listados de clasificación de clubes y jugadores.

6.3. Especificación de formatos de impresión

Objetivo:

“El objetivo de esta tarea es especificar los formatos y características de las salidas o entradas impresas del sistema.

De acuerdo a los estándares establecidos en la tarea Especificación de Principios Generales de la Interfaz (ASI 8.1), se definen los formatos individuales de informes y formularios, estos últimos si son necesarios, así como sus características principales, entre las que se especifican la periodicidad, confidencialidad, procedimientos de entrega o difusión, y salvaguarda de copia.

Opcionalmente, se recomienda la utilización de prototipos”.

Formatos de impresión:

Se ha creado un dialogo para las opciones específicas de impresión de todos los listados:

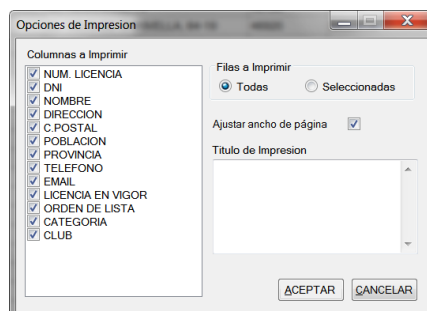


Figura 6.7. Ventana de detalles de impresión

El encargado de gestionar los formatos de impresión una vez establecidas las opciones es el propio sistema operativo. Es necesario que para su correcto funcionamiento la impresora sea compatible con PostScript v6.

7. Especificación del plan de pruebas

“En esta actividad se inicia la definición del plan de pruebas, el cual sirve como guía para la realización de las pruebas, y permite verificar que el sistema de información cumple las necesidades establecidas por el usuario, con las debidas garantías de calidad.

El plan de pruebas es un producto formal que define los objetivos de la prueba de un sistema, establece y coordina una estrategia de trabajo, y provee del marco adecuado para elaborar una planificación paso a paso de las actividades de prueba. El plan se inicia en el proceso Análisis del Sistema de Información (ASI), definiendo el marco general, y estableciendo los requisitos de prueba de aceptación, relacionados directamente con la especificación de requisitos.

Dicho plan se va completando y detallando a medida que se avanza en los restantes procesos del ciclo de vida del software, Diseño del Sistema de Información (DSI), Construcción del Sistema de Información (CSI) e Implantación y Aceptación del Sistema (IAS).

Se plantean los siguientes niveles de prueba:

- Pruebas unitarias.
- Pruebas de integración.
- Pruebas del sistema.
- Pruebas de implantación.
- Pruebas de aceptación.

En esta actividad también se avanza en la definición de las pruebas de aceptación del sistema. Con la información disponible, es posible establecer los criterios de aceptación de las pruebas incluidas en dicho nivel, al poseer la información sobre los requisitos que debe cumplir el sistema, recogidos en el catálogo de requisitos”.

Plan de pruebas:

Se establece un modelo de pruebas general que se describe con el siguiente diagrama³:

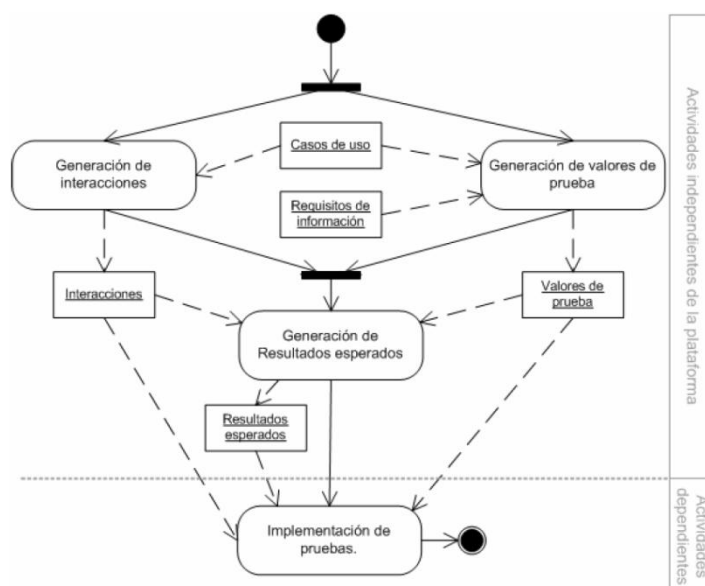


Figura 7.1. Diagrama de pruebas

Asimismo la documentación de las pruebas contendrá:

- Fecha de la prueba
- Persona responsable de la prueba
- Función o Interfaz a probar
- Nivel de ejecución (desarrollo – integración – producción)
- Si se requieren módulos externos de adicionales para el correcto funcionamiento de la prueba
- Tabla de casos de prueba, con un resumen de su resultado.
- Descripción de los casos de prueba resueltos.

Se ampliará la descripción de las pruebas en anexos sucesivos.

8. Aprobación del análisis de sistemas de información

“En esta tarea se realiza la presentación del análisis del sistema de información al Comité de Dirección, para la aprobación final del mismo.”

Aprobación del análisis del sistema de información:

Una vez presentado el análisis al presidente y vicepresidentes de la FBCV se establece que el sistema cumplirá los requisitos exigidos y se acepta la siguiente fase del proyecto.

³ Modelos de pruebas para el sistema (2006): <http://www.lsi.us.es/~javierj/publications/MDA14.pdf>

Anexo II-4. Índice:

1.	Definición de la arquitectura del sistema	3
1.1.	Definición de niveles de arquitectura	4
1.2.	Identificación de requisitos de diseño y construcción	6
1.3.	Especificación de excepciones	7
1.4.	Especificación de estándares y normas de diseño y construcción.....	9
1.5.	Especificación del entorno tecnológico	18
1.6.	Especificación de requisitos de operación y seguridad.....	19
2.	Diseño de clases	21
3.	Diseño físico de datos	38
4.	Diseño de la migración y carga inicial de datos	39
5.	Especificación técnica del plan de pruebas.....	40
6.	Especificación de requisitos de documentación de usuario	41

• Diseño del sistema de información (DSI)

“El objetivo del proceso de Diseño del Sistema de Información (DSI) es la definición de la arquitectura del sistema y del entorno tecnológico que le va a dar soporte, junto con la especificación detallada de los componentes del sistema de información. A partir de dicha información, se generan todas las especificaciones de construcción relativas al propio sistema, así como la descripción técnica del plan de pruebas, la definición de los requisitos de implantación y el diseño de los procedimientos de migración y carga inicial, éstos últimos cuando proceda.

Las actividades de este proceso se agrupan en dos grandes bloques.

- En un primer bloque de actividades, que se llevan a cabo en paralelo, se obtiene el diseño de detalle del sistema de información. La realización de estas actividades exige una continua realimentación. El sistema de información se estructura en subsistemas de diseño. Éstos a su vez se clasifican como de soporte o específicos, al responder a propósitos diferentes.

- Los subsistemas de soporte contienen los elementos o servicios comunes al sistema y a la instalación, y generalmente están originados por la interacción con la infraestructura técnica o la reutilización de otros sistemas, con un nivel de complejidad técnica mayor.

- Los subsistemas específicos contienen los elementos propios del sistema de información, generalmente con una continuidad de los subsistemas definidos en el proceso de Análisis del Sistema de Información (ASI).

También se especifica en detalle el entorno tecnológico del sistema de información, junto con su planificación de capacidades (capacity planning), y sus requisitos de operación, administración, seguridad y control de acceso.

El diseño detallado del sistema de información, siguiendo un enfoque estructurado, comprende un conjunto de actividades que se llevan a cabo en paralelo a la Definición de la Arquitectura del Sistema (DSI 1). El alcance de cada una de estas actividades se resume a continuación:

- Diseño de la Arquitectura de Soporte (DSI 2), que incluye el diseño detallado de los subsistemas de soporte, el establecimiento de las normas y requisitos propios del diseño y construcción, así como la identificación y definición de los mecanismos genéricos de diseño y construcción.

- Diseño de la Arquitectura de Módulos del Sistema (DSI 5), dónde se realiza el diseño de detalle de los subsistemas específicos del sistema de información y la revisión de la interfaz de usuario.

- Diseño Físico de Datos (DSI 6), que incluye el diseño y optimización de las estructuras de datos del sistema, así como su localización en los nodos de la arquitectura propuesta.

En el caso de Diseño Orientado a Objetos, conviene señalar que el diseño de la persistencia de los objetos se lleva a cabo sobre bases de datos relacionales, y que el diseño detallado del sistema de información se realiza en paralelo con la actividad de Diseño de la Arquitectura de Soporte (DSI 2), y se corresponde con las siguientes actividades:

- Diseño de Casos de Uso Reales (DSI 3), con el diseño detallado del comportamiento del sistema de información para los casos de uso, el diseño de la interfaz de usuario y la validación de la división en subsistemas.

- Diseño de Clases (DSI 4), con el diseño detallado de cada una de las clases que forman parte del sistema, sus atributos, operaciones, relaciones y métodos, y la estructura jerárquica del mismo. En el caso de que sea necesario, se realiza la definición de un plan de migración y carga inicial de datos.

Una vez que se tiene el modelo de clases, se comienza el diseño físico en la actividad Diseño Físico de Datos (DSI 6), común con el enfoque estructurado. Una vez finalizado el diseño de detalle, se realiza su revisión y validación en la actividad Verificación y Aceptación de la Arquitectura del Sistema (DSI 7), con el objeto de analizar la consistencia entre los distintos modelos y conseguir la aceptación del diseño por parte de los responsables de las áreas de Explotación y Sistemas.

- El segundo bloque de actividades complementa el diseño del sistema de información. En él se generan todas las especificaciones necesarias para la construcción del sistema de información:

- Generación de Especificaciones de Construcción (DSI 8), fijando las directrices para la construcción de los componentes del sistema, así como de las estructuras de datos.

- Diseño de la Migración y Carga Inicial de Datos (DSI 9), en el que se definen los procedimientos de migración y sus componentes asociados, con las especificaciones de construcción oportunas.

- Especificación Técnica del Plan de Pruebas (DSI 10), que incluye la definición y revisión del plan de pruebas, y el diseño de las verificaciones de los niveles de prueba establecidos. El catálogo de excepciones permite, de una forma muy ágil, establecer un conjunto de verificaciones relacionadas con el propio diseño o con la arquitectura del sistema.

- Establecimiento de Requisitos de Implantación (DSI 11), que hace posible concretar las exigencias relacionados con la propia implantación del sistema, tales como formación de usuarios finales, infraestructura, etc.

Finalmente, en la actividad de Presentación y Aprobación del Diseño del Sistema de Información (DSI 12), se realiza una presentación formal y aprobación de los distintos productos del diseño”.

1. Definición de la arquitectura del sistema

“En esta actividad se define la arquitectura general del sistema de información, especificando las distintas particiones físicas del mismo, la descomposición lógica en subsistemas de diseño y la ubicación de cada subsistema en cada partición, así como la especificación detallada de la infraestructura tecnológica necesaria para dar soporte al sistema de información.

El particionamiento físico del sistema de información se especifica identificando los nodos y las comunicaciones entre los mismos, con cierta independencia de la infraestructura tecnológica que da soporte a cada nodo.

Con el fin de organizar y facilitar el diseño, se realiza una división del sistema de información en subsistemas de diseño, como partes lógicas coherentes y con interfaces claramente definidas.

Se establece una distinción entre subsistemas específicos del sistema de información (en adelante, subsistemas específicos) y subsistemas de soporte, con la finalidad de independizar, en la medida de lo posible, las funcionalidades a cubrir por el sistema de información de la

infraestructura que le da soporte. En la mayoría de los casos, los subsistemas específicos provienen directamente de las especificaciones de análisis y de los subsistemas de análisis, mientras que los subsistemas de soporte provienen de la necesidad de interacción del sistema de información con la infraestructura y con el resto de los sistemas, así como de la reutilización de módulos o subsistemas ya existentes en la instalación.

Una vez identificados y definidos los distintos subsistemas de diseño, se determina su ubicación óptima de acuerdo a la arquitectura propuesta. La asignación de dichos subsistemas a cada nodo permite disponer, en función de la carga de proceso y comunicación existente entre los nodos, de la información necesaria para realizar una estimación de las necesidades de infraestructura tecnológica que da soporte al sistema de información. Este factor es especialmente crítico en arquitecturas multinivel o cliente/servidor, donde las comunicaciones son determinantes en el rendimiento final del sistema.

Se propone crear un catálogo de excepciones en el que se especifiquen las situaciones anómalas o secundarias en el funcionamiento y ejecución del sistema de información, y que se irá completando a medida que se avance en el diseño detallado de los subsistemas. En esta actividad también se establecen los requisitos, normas y estándares originados como consecuencia de la adopción de una determinada solución de arquitectura o infraestructura, que serán aplicables tanto en este proceso como en la Construcción del Sistema de Información (CSI)".

Tareas propuestas para esta actividad:

- 1.1. Definición de niveles de arquitectura
- 1.2. Identificación de requisitos de diseño y construcción
- 1.3. Especificación de excepciones
- 1.4. Especificación de estándares y normas de construcción
- 1.5. Especificación del entorno tecnológico
- 1.6. Especificación de requisitos de operación y seguridad

1.1. Definición de niveles de arquitectura

Objetivo:

"En esta tarea se describen los niveles de la arquitectura software, mediante la definición de las principales particiones físicas del sistema de información, representadas como nodos y comunicaciones entre nodos.

Se entiende por nodo cada partición física o parte significativa del sistema de información, con características propias de ejecución o función, e incluso de diseño y construcción.

Para facilitar la comprensión del sistema, se recomienda identificar como nodos los elementos de infraestructura más significativos de la arquitectura en la que se va a implementar el sistema de información. Los elementos que se aconseja especificar son los siguientes:

- Gestores de datos.
- Tipos de puesto cliente.
- Tipos de dispositivos de impresión.
- Monitores de teleproceso.
- Servidores.
- Comunicaciones.

La comunicación se expresa por una conexión entre nodos, indicando su carácter bidireccional o unidireccional, con las principales características de los protocolos o tipo de mensajes utilizados. La especificación de los niveles de la arquitectura se realiza con el detalle suficiente como para permitir un diseño dirigido hacia una solución concreta. En general, no es preciso indicar en cada nodo detalles relativos al hardware, capacidad, rendimiento o configuraciones de tolerancia a fallos, entre otros. Esta información se concreta en la tarea Especificación del Entorno Tecnológico (DSI 1.6).

Los criterios para diseñar la arquitectura se obtienen a partir de directrices tecnológicas o de integración, propias de la instalación, y del catálogo de requisitos del sistema de información. Es necesario tener en cuenta, especialmente, aspectos relacionados con:

- Usuarios: ubicación, movilidad, concurrencia, número, etc.
- Datos: variabilidad, volúmenes, necesidades de consolidación, seguridad, etc.
- Procesos: distribución, reutilización, concurrencia, carácter crítico, etc.”.

Niveles de arquitectura:

Se analizan en un primer lugar las ventajas de una arquitectura multinivel y se observa cuales de ellas influirían en la solución actual.

Recursos centralizados: debido a que el servidor es el centro de la red, puede administrar los recursos que son comunes a todos los usuarios, por ejemplo: una base de datos centralizada se utilizaría para evitar problemas provocados por datos contradictorios y redundantes. En el sistema actual solo habrá un usuario accediendo al sistema simultáneamente por lo que no es aplicable.

Seguridad mejorada: no importa la cantidad de puntos de entrada que permiten el acceso a los datos, ya que la seguridad principal esta centralizada. Actualmente el trabajo se desarrolla en un solo lugar físico, por lo que esta mejora tampoco es aplicable.

Administración al nivel del servidor: ya que los clientes no juegan un papel importante en este modelo, requieren por tanto menos administración. Siendo el administrador el mismo responsable que el usuario del software, no beneficiaría en la práctica.

Red escalable: gracias a esta arquitectura, es posible quitar o agregar clientes sin afectar el funcionamiento de la red y sin la necesidad de realizar mayores modificaciones. Sería un punto importante en futuras ampliaciones, pero por el coste se deja el desarrollo para más adelante.

Por tanto el sistema se basará en una arquitectura de un solo nivel, ya que se considera suficiente para el tipo de trabajo y la cantidad de transacciones diarias.

Se analizan las ventajas de una arquitectura de un solo nivel:

Menor coste: debido a la complejidad técnica del servidor, la solución de un nivel requiere menos tiempo de desarrollo y por tanto menor coste. Este punto es determinante en la configuración que se establecerá.

Comunicaciones: Al no existir comunicaciones entre diferentes maquinas se evita cualquier fallo que pueda provocar la paralización del sistema.

De esta manera, será una solución de tres capas (presentación, lógica del negocio, datos) que residirán en un solo ordenador (presentación + lógica + datos).

1.2. Identificación de requisitos de diseño y construcción

Objetivo:

“En esta tarea se realiza la especificación de los requisitos que están directamente relacionados con la adopción o diseño de una arquitectura o infraestructura concreta, y que pueden condicionar el diseño o la construcción del sistema de información.

Entre estos requisitos pueden estar los relacionados con lenguajes, rendimiento de los distintos elementos de la arquitectura, así como criterios de ubicación de módulos y datos en los distintos nodos.

Por tanto, como resultado de esta tarea se actualiza el catálogo de requisitos elaborado en el proceso Análisis de Sistemas de Información.”

Requisitos de diseño y construcción:

Como se ha especificado en el PSI, la arquitectura para el diseño y la construcción del sistema se basará en la plataforma .NET de Microsoft[®]. Sus ventajas a la hora de construir el sistema y diseñar prototipos son¹:

Código administrado: El motor en tiempo de ejecución de .NET realiza un control automático del código para que este sea seguro, es decir, controla los recursos del sistema para que la aplicación se ejecute correctamente.

Interoperabilidad multilinguaje: El código puede ser escrito en cualquier lenguaje compatible con .Net ya que siempre se compila en código intermedio.

Compilación just-in-time: El compilador JIT incluido en el Framework compila el código intermedio generando el código máquina propio de la plataforma. Se aumenta así el rendimiento de la aplicación al ser específico para cada plataforma.

Garbage collector: El motor en tiempo de ejecución proporciona un sistema automático de administración de memoria denominado recolector de basura (*garbage collector*). El MTE detecta cuándo el programa deja de utilizar la memoria y la libera automáticamente. De esta forma el programador no tiene que liberar la memoria de forma explícita aunque también sea posible hacerlo manualmente (mediante el método *dispose()* liberamos el objeto para que el recolector de basura lo elimine de memoria).

Seguridad de acceso al código: Se puede especificar que una pieza de código tenga permisos de lectura de archivos pero no de escritura. Es posible aplicar distintos niveles de seguridad al código, de forma que se puede ejecutar código procedente del Web sin tener que preocuparse si esto va a estropear el sistema.

Despliegue: Por medio de los ensamblados resulta mucho más fácil el desarrollo de aplicaciones distribuidas y el mantenimiento de las mismas. El Framework realiza esta tarea de forma automática mejorando el rendimiento y asegurando el funcionamiento correcto de todas las aplicaciones.

Además, se decide la utilización de una BB.DD. Ms.[®] Access 2007. Esta decisión viene condicionada por el hecho que la arquitectura será de un solo nivel. En otro caso, hubiera sido indispensable el uso de otro SGBD.

¹ Contadores de rendimiento .NET (2009): [http://msdn.microsoft.com/es-es/library/ms172525\(VS.90\).aspx](http://msdn.microsoft.com/es-es/library/ms172525(VS.90).aspx)

1.3. Especificación de excepciones

Objetivo:

“El objetivo de esta tarea es la definición de los comportamientos no habituales en el sistema, que reflejan situaciones anómalas o secundarias en el funcionamiento y ejecución del sistema de información. Para ello, se establece previamente el nivel de especificación de las mismas, así como los criterios de catalogación y clasificación.

Se propone su catalogación como ayuda para el diseño del sistema de información y como guía en la especificación técnica de las pruebas, al permitir la generación de algunos casos de prueba de forma inmediata. Dicho catálogo se va completando a partir de las actividades correspondientes al diseño detallado de los subsistemas.

Las excepciones se describen incluyendo, al menos, los siguientes conceptos:

- Tipo y descripción de la excepción.
- Condiciones previas del sistema de información.
- Elemento afectado (nodo, módulo, caso de uso).
- Respuesta del sistema de información.
- Elemento asociado a la respuesta esperada del sistema (módulo, clase, procedimiento, etc.).

Las excepciones que se proponen como obligatorias son las relacionadas con el funcionamiento general del sistema de información, habitualmente asociadas a:

- Nodos y comunicaciones del particionamiento físico del sistema de información. Este tipo de excepciones tiene lugar cuando no están disponibles los gestores de bases de datos o los recursos compartidos del sistema (representados como nodos), cuando se producen fallos en las comunicaciones entre nodos, etc.
- Rangos o valores no válidos en la entrada de datos, como pueden ser atributos obligatorios, con formatos específicos, etc.

Se recomienda, según el nivel de especificación que se establezca en cada caso, catalogar también las excepciones particulares que se identifiquen en las actividades del diseño de detalle”.

Catálogo de excepciones y errores:

En el sistema que se está desarrollando se distinguen diferentes tipos de excepciones según su nivel de procedencia. Todas las excepciones se irán incorporando a un catálogo conforme la construcción de la aplicación avance. En caso de ya existir esa excepción, se asigna un nuevo código y se añade al catálogo.

Excepciones de error:

- Excepciones por datos de entrada incorrectos sin acceso a BB.DD.

En este primer caso, para evitar este tipo de excepciones, al introducir un dato de entrada con un formato o valor incorrecto, el sistema localiza el cuadro de entrada y espera a que el usuario rectifique el texto. En caso de no hacerlo, el sistema no permite continuar (solo cancelar el diálogo).

A screenshot of a web form for data entry. The form contains several input fields: 'APELLIDOS:', 'FECHA NACIMIENTO:' (with the value '23/25/6256' and a red error icon), 'COD POSTAL:', 'POBLACION:', and 'EMAIL:'. A tooltip message 'La fecha es incorrecta' is displayed over the birth date field. At the bottom of the form are two buttons: 'ACEPTAR' and 'CANCELAR'.

Figura 1.1. Validación de datos de entrada

- Excepciones por datos de entrada incorrectos con acceso a BB.DD.

En este caso, al introducir un valor que requiere tener un campo dado previamente de alta en la BB.DD. el sistema responde con una ventana de mensaje advirtiendo de la necesidad de realizar una acción previa.

A screenshot of a 'Socio_editar' window. It contains several input fields: 'NUMERO LICENCIA:' (with value '6+2'), 'CLUB:', 'ORDEN FUERZA:', 'LICENCIA EN VIGOR:', 'NOMBRE:', 'DNI:', 'DIRECCION:', 'PROVINCIA:', 'TELEFONO:', and 'EMAIL:'. A modal error dialog box is overlaid on the form, containing the text 'El club todavia no existe, introduce primero un club valido' and an 'Aceptar' button. The main window also has 'ACEPTAR' and 'CANCELAR' buttons at the bottom.

Figura 1.2. Ventana de error por datos incorrectos

- Excepciones por accesos incorrectos a BB.DD.

En caso que la excepción no permita vuelta atrás, toda excepción irá asociada a un código, y estos códigos recogidos en una tabla Excel de códigos de excepción. El código contendrá un primer dígito indicando el tipo de excepción, otro con la función de negocio del fallo, y otro conjunto de dígitos en numeración secuencial según la excepción.

A screenshot of a 'Resultado' window. It displays the text 'Excepción controlada: 1-1-0001' and an 'ACEPTAR' button at the bottom.

Figura 1.3. Ventana con código de error

Excepciones generales:

Las excepciones no provocadas por datos de entrada, ya sean excepciones de cálculo, de proceso o de salida, se considerarán excepciones generales, y conllevan un código de manera similar al que se enlaza con las excepciones por accesos incorrectos a BB.DD.

Excepciones no controladas

Aunque en la medida de lo posible se intentará proteger toda excepción del programa, es posible que no se lleguen a controlar todas (faltas de ficheros y librerías, problemas con el gestor de BB.DD., etc. etc.). Por tanto como norma se establecerá deshabilitar cualquier tipo de depuración just-in-time que ofrece .NET para evitar que la aplicación pueda ser vulnerada. Se especificará en el manual de usuario que en caso de ocurrir una excepción no controlada, se contacte con el servicio de atención al cliente.

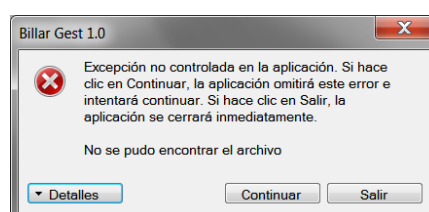


Figura 1.4. Excepción no controlada

1.4. Especificación de estándares y normas de diseño y construcción

Objetivo:

“En esta tarea se definen los estándares técnicos y de nomenclatura, normas y recomendaciones, que generalmente están relacionados con la adopción o diseño de una arquitectura o infraestructura tecnológica concreta, y que pueden condicionar el diseño o la construcción del sistema de información.

Como resultado de esta tarea, se actualiza el catálogo de normas obtenido en el proceso Análisis del Sistema de Información. La información recogida en el catálogo se debe tener en cuenta en la elaboración de los productos resultantes del diseño y construcción del sistema de información. El catálogo de normas es, por tanto, producto de entrada en todas las tareas, aunque por sencillez se omite la referencia al mismo.”

Se definirán por separado tanto los estándares de codificación completos como los estándares de diseño:

1. Estándares de codificación:

Un estándar de codificación comprende todos los aspectos de la generación de código. Al comenzar un proyecto de software, es necesario establecer un estándar de codificación para asegurarse de que todos los programadores del proyecto trabajen de forma coordinada. Cuando el proyecto de software incorpore código fuente previo, o bien cuando realice el mantenimiento de un sistema de software creado anteriormente, el estándar de codificación debería establecer cómo operar con la base de código existente.

Usar técnicas de codificación sólidas y realizar buenas prácticas de programación con vistas a generar un código de alta calidad es de gran importancia para la calidad del software y para obtener un buen

rendimiento. Para ello, se usarán las técnicas de codificación propuestas por Microsoft[®] para el desarrollo de software en .NET².

La adopción de un estándar de codificación sólo es viable si se sigue desde el principio hasta el final del proyecto de software.

El estándar además propone llevar a cabo revisiones del código a lo largo de todo el desarrollo para localizar defectos en el mismo, las revisiones también pueden afianzar los estándares de codificación de manera uniforme.

GENERALES

- Para conservar recursos se será muy selectivo en la elección del tipo de dato, asegurando que el tamaño de una variable no sea excesivamente grande. Por ejemplo en los ciclos *for* es mejor, en la mayoría de las veces utilizar un tipo de dato tipo *short* que *int*.
- Se mantendrá el tiempo de vida de las variables tan corto como sea posible, esto es muy importante por ejemplo cuando se utiliza un recurso finito como una conexión a una Base de Datos.
- Se mantendrá el alcance de las variables tan corto como sea posible, esto sirve para evitar confusiones, mejorar la mantenibilidad y además minimiza la dependencia, es decir si por algún motivo se comete el error de borrar una variable es más fácil de encontrar el error si esta tiene un alcance más pequeño.
- Se usarán los procedimientos y variables con un solo propósito, evitando crear procedimientos multipropósito que lleven a cabo una variedad de funciones no relacionadas.
- Dentro de una clase, se evitará el uso de variables públicas, en cambio se utilizarán procedimientos y propiedades que accedan a dichas variables (privadas), así se provee una capa de encapsulación y brinda la posibilidad de validar valores de cambio sobre las mismas, antes de manipularlas directamente.
- Cuando se utilicen objetos pooled de MTS (Microsoft Transaction Server), se adquirirán los recursos lo más tarde posible y se liberarán lo más pronto como sea posible, esto repercute en el manejo de recursos.
- Usar siempre un esquema transaccional como MTS o SQL Server y minimizar el alcance y la duración de las transacciones.
- No abrir conexiones a datos usando credenciales específicas de usuario, ya que estas no podrán ser reutilizadas por el pool de conexiones.
- Evitar el uso de conversión de tipos o casting ya que esto puede generar resultados imprevistos, sobre todo cuando dos variables están involucradas en una sentencia, utilizar el cast solo en situaciones triviales, cuando este no sea el caso asegurar de comentar la razón por la cual se hizo.
- Usar siempre rutinas de manejo de excepciones.
- Ser específico cuando se declaren objetos que puedan generar colisión, por ejemplo si se tiene dos métodos con el mismo nombre en diferentes namespaces escribirlos con el nombre completo incluyendo el del paquete.

² Técnicas de codificación y prácticas de programación .NET (2010): [http://msdn.microsoft.com/es-es/library/aa291596\(VS.71\).aspx](http://msdn.microsoft.com/es-es/library/aa291596(VS.71).aspx)

- Evitar el uso de variables en el ámbito de aplicación.
- Usar siempre sentencias *Select-Case* o *Switch* en lugar de utilizar sentencias if-then repetitivas.
- Liberar explícitamente las referencias a objeto. (*variable = nothing* ó *variable = null*)
- Siempre que sea posible utilizar polimorfismo en vez de cláusulas *Switch* o *Select*.

Las técnicas de codificación están divididas en tres secciones:

- a) Nombrado
- b) Documentación Interna (Comentarios)
- c) Formato

a) Nombrado:

Si se utiliza un nombre que evite referirse a la implementación se estará conservando la abstracción de la estructura ya que la implementación está sujeta a cambios, de esta manera se describe que hace la estructura y no como lo hace. Por ejemplo es más claro nombrar un procedimiento de acceso a datos *SeleccionarRegistro()* que *RealizarConsultaSelect()*, porque lo que importa (para que otra persona entienda el código) es que se supone que hace el método y no como lo hace.

Otra directiva es la de utilizar nombres tan largos como para que se entiendan pero a la vez tan cortos como para que no den información irrelevante, por ejemplo es mejor emplear *SeleccionarClub()* que *SeleccionarelClubdelaProvincia()*.

Estructuras (namespaces, procedimientos, clases, interfaces y propiedades)

- Los nombres de todas las estructuras de código deben ser en español.
- Los namespaces deben empezar por el código *BillarGest* seguido de la unidad de negocio y la funcionalidad, ej *Billargest.Actas.AccesoDatos*
- El nombre del ensamblado y el namespace root deben ser idénticos
- El nombre de la clase y el archivo fuente deben ser iguales.
- No se debe usar la notación húngara, la cual prefixa una abreviatura referente al tipo de objeto: *lblAceptar (label)*, *btnOK(Botón)*, etc.
- Evitar nombres imprecisos que permitan interpretaciones subjetivas, como por ejemplo *DefinirEsto()*, o bien *ytG8* para una variable.
- En la Programación Orientada a Objetos es redundante incluir nombres de clases en el nombre de las propiedades de clases, como por ejemplo *Billar.BillarSocio*, en su lugar, utilizar *Billar.Socio*, pues el nombre de la clase ya contiene dicha información.
- Utilizar la técnica verbo-sustantivo para nombrar procedimientos que ejecuten alguna operación en un determinado objeto, como por ejemplo *CalcularTemporada()*.

- Empezar los nombres de clase y propiedades con un nombre, por ejemplo *TemporadasAnyo*, la primera letra de cada palabra debe ser mayúscula.

- Empezar los nombres de interfaz con el prefijo "I", seguido de un nombre o una frase nominal, como *IComponente*, o con un adjetivo que describa el comportamiento de la interfaz, como *IPersistible*. No utilizar el subrayado "_" (con la excepción de las variables privadas), y utilizar lo menos posible las abreviaturas, ya que pueden causar confusiones.

Variables

- Las variables miembro se escriben con la primera letra de cada palabra en mayúscula a excepción de las variables miembro privadas. Las Variables internas o de bloque deben ir en minúscula. En el caso de las variables miembro privadas, se debe utilizar el prefijo "m_" sumado al nombre de la variable (*m_variable*).

- Es recomendable que las variable booleanas contengan una palabra que describa su estado: *puedeEliminarse*, *esGrande*, *tieneHijos*, etc. Y siempre se debe referir al estado verdadero: *tieneSocios* en cambio de *noTieneSocios*.

- Incluso para el caso de una variable de poco uso, que deba aparecer sólo en unas cuantas líneas de código, emplear un nombre descriptivo. Utilizar nombres de variables de una sola letra, como *i* o *j* sólo para índices (ciclos *for*).

- No utilizar números o cadenas literales, como por ejemplo *For i = 1 To 7*. En su lugar, emplear constantes con nombre, del tipo *For i = 1 To Enumeracion.length* para que resulten fáciles de mantener y comprender.

Parámetros

Los parámetros siguen el mismo estándar de las variables

Tablas

- Los nombres de las tablas serán en singular. Por ejemplo, *CLUB* en lugar de *CLUBES*

- Cuando se pongan nombres a las columnas de las tablas, no repetir el nombre de la tabla; por ejemplo, evitar un campo llamado *ClubNombre* de una tabla llamada *Club* (Igual que con las propiedades de una clase).

- No incorporar el tipo de datos en el nombre de una columna.

Varios

- Para términos largos o utilizados con frecuencia, utilizar abreviaturas para mantener las longitudes de los nombres dentro un límite razonable, por ejemplo, "HTML" en lugar de "Lenguaje de marcado de hipertexto". Además, asegurarse de que sus abreviaturas sean coherentes a lo largo de toda la aplicación. Si se utiliza indistinta y aleatoriamente "HTML" y "Lenguaje de marcado de hipertexto" en un mismo proyecto, provocará confusión.

- Minimizar el uso de abreviaturas; pero si las emplea, use coherentemente las que haya creado. Una abreviatura sólo debe tener un significado y, del mismo modo, a cada palabra

abreviada sólo debe corresponder una abreviatura. Por ejemplo, si utiliza "min." para abreviar "mínimo", hágalo siempre así, y no use también "min." para abreviar "minuto".

- Cuando se nombre un procedimiento que retorne un valor, incluir además de la acción, la entidad que será devuelta; *LeerArchivo*, *ConsultarSocio*, *ObtenerResumen*.
- Los archivos y los nombres de carpetas, al igual que los nombres de procedimientos, deben describir claramente su finalidad.
- Evitar reutilizar nombres para elementos diferentes, como por ejemplo una rutina llamada *ProcesoLavado()* y una variable *iLavado*.
- Evitar el uso de caracteres como "\$" o "%"
- No usar nombres que sean dependientes del tipo de variable, control o clase en cambio utilizar nombres que describan el propósito de la variable, control o propiedad.
- Evitar este comportamiento: *void Escribir(double doubleValor)*, en su lugar utilizar *void Escribir(double valor)*.

b) Comentarios:

Existen dos tipos de documentación de software: externa e interna. La documentación externa, como por ejemplo las especificaciones, los archivos de ayuda y los documentos de diseño, se debe mantener fuera del código fuente. La documentación interna está formada por los comentarios que los programadores escriben dentro del código fuente durante la fase de desarrollo.

Uno de los problemas de la documentación de software interna es garantizar que se mantengan y actualicen los comentarios al mismo tiempo que el código fuente. Aunque unos buenos comentarios en el código fuente no tienen ningún valor en el tiempo de ejecución, resultan valiosísimos para el programador que tiene que mantener el software.

Se seguirán las siguientes pautas para realizar comentarios:

- Los comentarios deben ser en español.
- Utilizar la función de documentación XML de C#. Cuando se modifique el código, mantener siempre actualizados los comentarios circundantes.
- Al principio de cada rutina, resulta útil hacer comentarios estándar, repetitivos, que indiquen el propósito de la rutina, las suposiciones y las limitaciones. Un comentario repetitivo podría consistir en una breve introducción que explicara por qué existe y qué puede hacer.
- Evitar añadir comentarios al final de una línea de código. Los comentarios de final de línea sí son apropiados al anotar declaraciones de variables. En este caso, alinear todos los comentarios de final de línea en la misma posición de tabulación.
- Evitar los comentarios recargados, como las líneas enteras de asteriscos. En su lugar, utilizar espacios para separar los comentarios y el código.
- Evitar rodear un bloque de comentarios con un marco tipográfico. Puede resultar agradable, pero es difícil de mantener.
- Antes de la implementación, quitar todos los comentarios temporales o innecesarios, para evitar cualquier confusión en la futura fase de mantenimiento.

- Si se necesita realizar comentarios para explicar una sección de código compleja, examinar el código para decidir si se debería volver a escribir. Siempre que sea posible, no documentar un código malo, volver a escribirlo. Aunque, por regla general, no debe sacrificarse el rendimiento para hacer un código más simple para el usuario, es indispensable un equilibrio entre rendimiento y mantenibilidad.
- Usar frases completas cuando se escriba comentarios. Los comentarios deben aclarar el código, no añadirle ambigüedad.
- Ir comentando al mismo tiempo que programa. Por otro lado, aunque se tuviera la oportunidad de revisar el código que se ha escrito, lo que parece obvio hoy es posible que seis semanas después no lo sea.
- Evitar usar comentarios superfluos o inapropiados, como comentarios divertidos al margen.
- Usar los comentarios para explicar el propósito del código. No se deben usar como si fueran traducciones literales.
- Para evitar problemas recurrentes, hacer siempre comentarios al depurar errores y solucionar problemas de codificación, especialmente cuando se trabaje en equipo.
- Hacer comentarios en el código que esté formado por bucles o bifurcaciones lógicas.
- Realizar los comentarios en un estilo uniforme, respetando una puntuación y estructura coherentes a lo largo de toda la aplicación.
- Separar los comentarios de sus delimitadores mediante espacios. Si se respetan estas normas, los comentarios serán más claros y fáciles de localizar si trabaja sin indicaciones de color.
- Cada declaración de variable importante debe incluir un comentario en la misma línea que describa el uso de la variable que se declara.
- Después de la secuencia de continuación de línea no puede escribirse un comentario en la misma línea.
- Todos los comentarios deben estar ortográficamente bien escritos, una escritura incorrecta demuestra un desarrollo negligente.
- Evitar comentarios que expliquen cosas obvias, en la mayoría de las cosas el código debe ser auto explicativo. Un buen código con buenas prácticas de nombrado no debe ser comentado.

c) Formato:

El formato hace que la organización lógica del código sea más clara. Se seguirán las siguientes pautas para establecer el formato:

- Evitar albergar múltiples clases en un solo archivo.
- Establecer un tamaño estándar de sangría (por ejemplo, cuatro espacios o una tabulación) y usarlo siempre. Alinear las secciones de código mediante la sangría predeterminada.
- Usar un único tipo de letra cuando publique versiones impresas del código fuente.
- Declarar una sola variable por línea.

- Incluir llaves en los bloques condicionales así sea de una sola sentencia.
- Agregar los namespaces en orden descendente empezando por los del sistema y terminado por los personalizados o de usuario.
- Una clase debe estar definida en orden descendente de la siguiente manera: Variables Miembro, Constructores, Enumeraciones, Estructuras o Clases anidadas, Propiedades y por último los Métodos.
- La secuencia de declaración de acuerdo a los modificadores de acceso debe ser la siguiente:

public > protected > internal > private

- Aislar la interfaz de implementación utilizando bloques Region.
- Alinear verticalmente las llaves de apertura y cierre de cualquier bloque:

```
Public void Main()  
{  
...  
}
```

- Establecer una longitud de línea máxima para el código.
- Utilizar espacios antes y después de los operadores siempre que eso no altere la sangría aplicada al código
- Usar espacios en blanco para organizar a su antojo secciones de código. De tal manera que se comprenda la segmentación del código.
- Cuando se tenga que dividir una línea en varias, aclarar que el código sigue en la línea de más abajo mediante un operador de concatenación colocado al final de cada línea, y no al principio.
- Siempre que sea posible, no colocar más de una instrucción por línea, a excepción de los bucles.
- Al escribir en HTML, establecer un formato estándar para las etiquetas y los atributos; como por ejemplo, las etiquetas siempre en mayúscula y los atributos en minúscula
- Cuando se escriba instrucciones SQL utilizar mayúsculas para las palabras clave: *SELECT*, *UPDATE*, *WHERE*, *FROM*, etc.
- Colocar las cláusulas SQL principales en líneas separadas, de modo que las instrucciones sean más fáciles de leer y editar:

```
SELECT Nombre, Apellido  
FROM Clientes  
WHERE Fecha= 'Now.Today';
```

- Cuando se aplique más de un atributo a una estructura colocarlos individualmente en una línea y no todos juntos y separados por comas.
- Siempre que sea posible inicializar las variables en el momento de la declaración (no es necesario en VisualBasic).
- Siempre utilizar los tipos nativos de cada lenguaje y no los del CTS.

- Evitar hacer unboxing y boxing secuencialmente, es decir, en la misma rutina.
- No comparar strings con "" o con *string.empty*, usar la propiedad *length* del *string*:

```
If(String.length == 0)
```

- Evitar las llamadas a métodos dentro de sentencias condicionales.
- Evitar utilizar recursividad, usar mejor ciclos for anidados.
- Usar el operador condicional ternario solo en casos triviales, cuando sean casos complejos evitarlo.
- Evitar comparar variables booleanas contra false o true, en vez de eso utilizar el operador de negación:

```
If(puedeEliminarse == true)if(puedeEliminarse=false)
```

Mejor usar:

```
If(puedeEliminarse)if(!puedeEliminarse)
```

2. Estándares de interfaz:

1. Consideraciones respecto al ratón:

La funcionalidad del mouse se utilizará para seleccionar componentes en pantalla y comandos del menú por parte del usuario.

El botón izquierdo del mouse será utilizado para seleccionar los componentes y los comandos del menú, siendo el único botón activado de este dispositivo.

2. Consideraciones respecto al teclado:

El teclado tendrá la funcionalidad del ingreso de datos. Asimismo, mediante la tecla *TAB* o la tecla *ENTER* el usuario podrá desplazar el foco de selección del teclado de componente en componente de forma ordenada.

3. Con lo referente a la ventana principal:

- Contendrá el menú principal.
- Permitirá enlaces con otras ventanas, para realizar actividades específicas.
- Contendrá un fondo de pantalla distintivo de las otras.
- El tamaño será de 1024x768 píxeles.
- La ventana iniciará maximizada por defecto y podrá ser minimizada.

4. Con referente a las ventanas se deben tener en cuenta las siguientes consideraciones acerca de sus propiedades:

- Start Position: Center Screen
- Form Border Style: Fixed Single
- Maximize Box: False
- Minimize Box: False

Todos los componentes serán tabulados a una misma distancia, a partir de margen izquierdo.

Para el cálculo de la separación entre los diferentes elementos de la ventana, se configurarán las opciones del menú Herramientas con los valores que se detallan a continuación en el campo Diseñador de Windows Forms – General:

- Layout Mod: Snap To Grid
- Show Grid: True
- Snap-To-Grid: False

A continuación, en las propiedades de la ventana se colocan los siguientes valores:

- Draw Grid: True
- Grid Size: 15 píxeles.

5. Con referente a la separación entre las etiquetas, las cajas de texto, las listas de selección simple y múltiple, se deberá considerar:

- La separación vertical entre cajas de texto, listas de selección simple y múltiple será de 12 píxeles.
- La separación vertical entre etiquetas será de 20 píxeles.
- La separación será horizontal entre etiquetas y demás controles (excepto control de marco): Se debe tomar en cuenta la alineación con las demás etiquetas del mismo grupo (alineación hacia la izquierda). Es decir, el espacio de separación será la mayor longitud de alguna de dichas etiquetas más 20 píxeles.

6. Con referente a la separación y alineación de un control de marco:

- La separación horizontal entre control de marco y límite izquierdo de la ventana: 10 píxeles.
- La separación horizontal entre etiquetas y control de marco: 20 píxeles.
- Para la alineación de los elementos contenidos por un control de marco, se deberá seleccionar todos los elementos mencionados y dar clic en la opción Centrar Horizontalmente que aparece en la barra de herramientas.

7. Con referente a los botones de pulsación se deben tener en cuenta las siguientes consideraciones:

- La separación horizontal entre botones será de 35 píxeles.
- La separación vertical entre botón y límite inferior de la ventana o marco que lo contiene será de 20 píxeles.
- El tamaño de los botones serán de 45x35 píxeles
- Las imágenes que tendrán los botones estarán centradas.

8. Con referente al elemento grilla (DataGridView) se deben tener en cuenta las siguientes consideraciones:

- Auto Size Columns Mode: All Cells
- Auto Size Rows Mode: All Cells
- Read Only: True (Todas las columnas)

1.5. Especificación del entorno tecnológico

Objetivo:

“En esta tarea se definen en detalle los distintos elementos de la infraestructura técnica que dan soporte al sistema de información, determinando la implementación concreta de los nodos y comunicaciones especificados en la tarea Definición de Niveles de Arquitectura (DSI 1.1).

Se propone agrupar los elementos de la infraestructura en los siguientes conceptos:

- Hardware: procesadores, unidades de almacenamiento, estaciones de trabajo, etc.
- Software: sistemas operativos, subsistemas, middleware, gestores de bases de datos, sistemas de ficheros, software de base, herramientas y utilidades de gestión propias del sistema, etc.
- Comunicaciones: diseño de la topología de la red, protocolos, nodos de red, etc.

La definición de los distintos elementos puede generar restricciones técnicas que afecten al diseño o construcción del sistema de información. Asimismo, se realiza una estimación de la planificación de capacidades (capacity planning) o se especifican los parámetros que Explotación y Sistemas precisen para realizar dicha planificación. Se indican, al menos, las necesidades previstas de:

- Almacenamiento: espacio en disco, espacio en memoria, pautas de crecimiento y evolución estimada del sistema de información, etc.
- Procesamiento: número y tipo de procesadores, memoria, etc.
- Comunicaciones: líneas, caudal, capacidades de elementos de red, etc.

Para poder determinar la planificación de capacidades, es necesario conocer los diseños detallados de los módulos / clases y escenarios, incluida la información de control en las comunicaciones, así como el diseño físico de datos optimizado, productos que se están generando en paralelo a esta actividad. También se tienen en cuenta, cuando proceda, las estimaciones de volúmenes de datos propios de la migración y carga inicial de datos”.

Entorno tecnológico:

Como se ha comentado anteriormente, la plataforma se creará siguiendo un modelo de tres capas pero con un nivel de arquitectura. Por tanto, no se requieren diferentes máquinas según el modelo de capa, todas harán las funciones de estación de trabajo.

Infraestructura entorno a:

- Hardware:

(2) Workstation hechas a medida, compuestas por:

- Caja MicroATX
- Microprocesador Intel Dual Core E5200
- Placa base Asus mod. P S775
- Disco duro 1TB
- Memoria RAM 4 GB 677 MHz
- Red Ethernet 10/100/1000
- Impresora HP 1022
- Monitor TFT HP 22” Mod. W2216v

- Software:

- S.O. Windows 7 Home

- Comunicaciones:

- Switch 4 puertos 10/100/1000 SMC Mod. 24C.
- Conexión ADSL.

Con respecto a la plataforma se especifican a continuación los requisitos mínimos:**- Requisitos de software:**

- Microsoft Windows XP Home o Microsoft Windows XP Professional, ambos con Service Pack 2 o posterior
- Microsoft Windows Server 2003 con Service Pack 1 o posterior
- Microsoft Vista
- Microsoft Windows Server 2008

Nota:

.NET Framework 3.5 solo admite IA64 en Microsoft Windows Server 2008.

- Requisitos de hardware:

- Mínimo: Pentium a 400 MHz, 96 MB
- Se recomienda: Pentium a 1 GHz o más rápido, 256 MB o más

1.6. Especificación de requisitos de operación y seguridad

Objetivo:

“El objetivo de esta tarea es definir los procedimientos de seguridad y operación necesarios para no comprometer el correcto funcionamiento del sistema y garantizar el cumplimiento de los niveles de servicios que exigirá el sistema en cuanto a la gestión de operaciones (procesos por lotes, seguridad, comunicaciones, etc.). Los niveles de servicio se especifican formalmente en el proceso Implantación y Aceptación del Sistema (IAS).

Tomando como referencia los requisitos establecidos para el sistema, y teniendo en cuenta la arquitectura propuesta y las características del entorno tecnológico definido en esta actividad, se lleva a cabo la definición de los requisitos de seguridad y control de acceso necesarios para garantizar la protección del sistema y minimizar el riesgo de pérdida, alteración o consulta indebida de la información. Para ello, se diseñan los procedimientos relacionados con:

- Acceso al sistema y a sus recursos (datos, transacciones, librerías, etc.).
- Mantenimiento de la integridad y confidencialidad de los datos.
- Control y registro de accesos al sistema (logs, certificación, etc.).
- Copias de seguridad y recuperación de datos y su periodicidad.
- Recuperación ante catástrofes.

Asimismo, se definen los requisitos de operación para los distintos elementos del sistema (módulos, clases, estructuras físicas de datos, sistemas de ficheros), que se están elaborando en paralelo a esta actividad, y se diseñan los procedimientos asociados relacionados con:

- Tratamiento en línea (franja horaria/periodos críticos, número máximo de usuarios, etc.).
- Tratamiento por lotes (periodicidad y secuencia de ejecución, interdependencias, petición de ejecución, etc.).
- Control y planificación de trabajos.
- Recuperación y reanudación de trabajos.
- Distribución de información generada por el sistema, tanto trabajos planificados o bajo petición.
- Control y seguimiento del correcto funcionamiento de los procedimientos de backup y recuperación utilizados habitualmente.

Requisitos de operación y seguridad:

El control de acceso al programa vendrá establecido por el propio control de acceso que proporciona el sistema operativo instalado, en este caso Windows 7. Se crearán credenciales para los usuarios que trabajen con el software de la aplicación. Los logs de acceso al sistema serán habilitados y se mantendrá el registro de errores activo.

Los usuarios, si bien podrán acceder con todos los derechos de administrador, tendrán instalado software de tipo antispyware y antivirus para evitar cualquier intrusión en el sistema, así como el control de aplicaciones proporcionado por defecto en Windows 7. Las actualizaciones del sistema operativo serán también automáticas por defecto.

Para el mantenimiento y copia de los datos de la aplicación, se integra una opción que permite el guardado de todas las tablas que maneja el sistema en cualquier ubicación. Este fichero tendrá una extensión **.billar*. La apertura y el salvado se realizarán con el habitual diálogo de Windows de la forma:

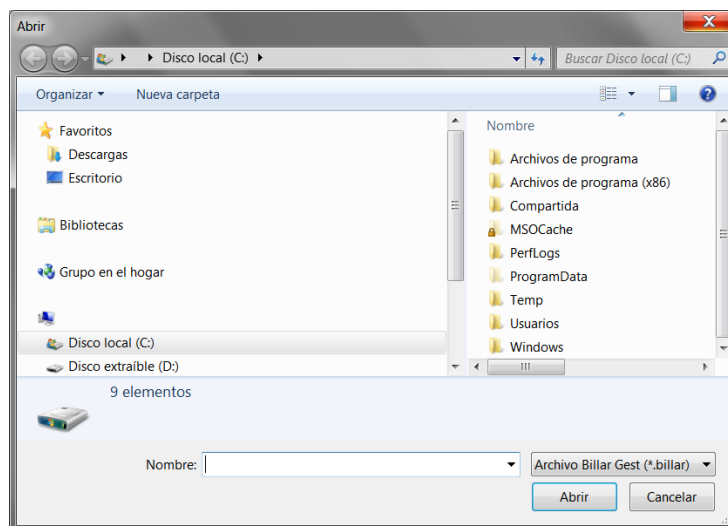


Figura1.5. Diálogo de apertura de fichero

No se establece por defecto el salvado de la BB.DD. cada periodo de tiempo, pero será posible habilitarlo si las exigencias del personal de administración lo requiere. Se establece como política interna del departamento el guardar un fichero de copia al finalizar cada día si se han introducido cambios en el sistema, este fichero será guardado en una unidad externa compatible.

2. Diseño de clases

“El propósito de esta actividad, que se realiza sólo en el caso de Diseño Orientado a Objetos, es transformar el modelo de clases lógico, que proviene del análisis, en un modelo de clases de diseño. Dicho modelo recoge la especificación detallada de cada una de las clases, es decir, sus atributos, operaciones, métodos, y el diseño preciso de las relaciones establecidas entre ellas, bien sean de agregación, asociación o jerarquía. Para llevar a cabo todos estos puntos, se tienen en cuenta las decisiones tomadas sobre el entorno tecnológico y el entorno de desarrollo elegido para la implementación.

Se identifican las clases de diseño, que denominamos clases adicionales, en función del estudio de los escenarios de los casos de uso, que se está realizando en paralelo en la actividad Diseño de Casos de Uso Reales (DSI 3), y aplicando los mecanismos genéricos de diseño que se consideren convenientes por el tipo de especificaciones tecnológicas y de desarrollo. Entre ellas se encuentran clases abstractas, que integran características comunes con el objetivo de especializarlas en clases derivadas. Se diseñan las clases de interfaz de usuario, que provienen del análisis. Como consecuencia del estudio de los escenarios secundarios que se está realizando, pueden aparecer nuevas clases de interfaz.

También hay que considerar que, por el diseño de las asociaciones y agregaciones, pueden aparecer nuevas clases, o desaparecer incluyendo sus atributos y métodos en otras, si se considera conveniente por temas de optimización.

La jerarquía entre las clases se va estableciendo a lo largo de esta actividad, a medida que se van identificando comportamientos comunes en las clases, aunque haya una tarea propia de diseño de la jerarquía.

Otro de los objetivos del diseño de las clases es identificar para cada clase, los atributos, las operaciones que cubren las responsabilidades que se identificaron en el análisis, y la especificación de los métodos que implementan esas operaciones, analizando los escenarios del Diseño de Casos de Uso Reales (DSI 3). Se determina la visibilidad de los atributos y operaciones de cada clase, con respecto a las otras clases del modelo.

Una vez que se ha elaborado el modelo de clases, se define la estructura física de los datos correspondiente a ese modelo, en la actividad Diseño Físico de Datos (DSI 6).

Además, en los casos en que sea necesaria una migración de datos de otros sistemas o una carga inicial de información, se realizará su especificación a partir del modelo de clases y las estructuras de datos de los sistemas origen.

Como resultado de todo lo anterior se actualiza el modelo de clases del análisis, una vez recogidas las decisiones de diseño.”

1) Gestión temporalada:

Se realiza la creación de una temporalada y se pueden abrir temporaladas existentes en la aplicación

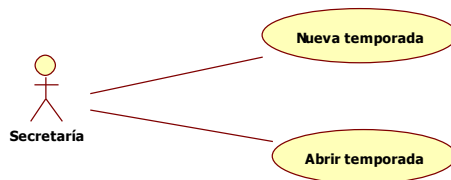


Figura 2.1. Gestión de temporalada

1.1 Crear nueva temporalada

El usuario puede crear una nueva temporalada introduciendo cada uno de los clubes que participaran y en qué división van a competir.

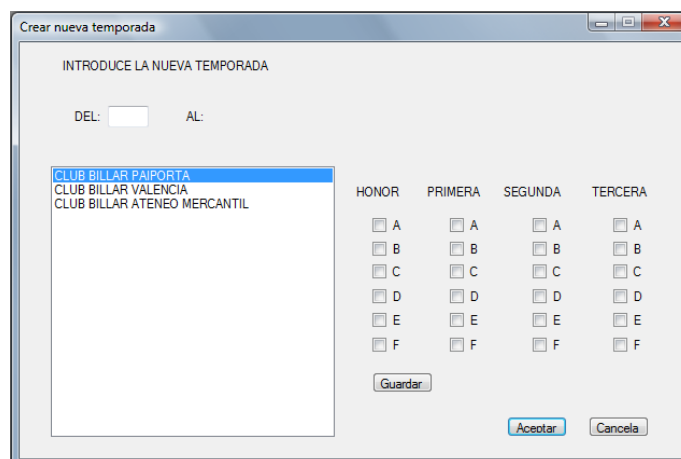


Figura 2.2. Crear nueva temporalada

Tipos elementales de datos:

Fecha comienzo temporalada:	date
Fecha fin temporalada:	date
Nombre del club:	varchar[50]
Cod_club:	integer[3]
Disponible	bit

Tipos derivados de datos:

Cod_temporada:	integer[8]
----------------	------------

Formado por los 8 dígitos que corresponden a los dos años de las fechas introducidas:

Vector de participación de temporaladas:	integer [6][4]
------------------------------------------	----------------

Ficheros accedidos:

- Temporada[lectura/escritura]
- Temporada_Club[lectura/escritura]
- Clasificación club[lectura/escritura]
- Clasificación jugador[lectura/escritura]

1.2 Abrir temporada

Se abre una temporada existente en el sistema. Se selecciona la temporada y se abre.

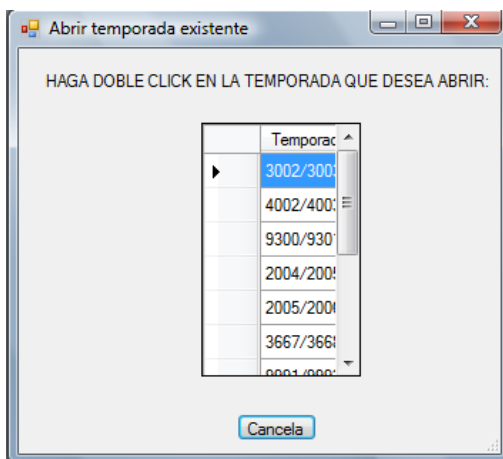


Figura 2.3. Abrir temporada existente

Ficheros accedidos:

- Temporada[lectura/escritura]
- Temporada_Club[lectura/escritura]
- Clasificación club[lectura/escritura]
- Clasificación jugador[lectura/escritura]

2) Gestión socios:

Se da de alta un nuevo socio, se pueden editar los valores de cada socio, se puede borrar un socio en caso de que este se dé de baja del club correspondiente y se imprimen las etiquetas a partir de las cuales se crean los carnets.



Figura 2.4. Gestión de socios

2.2 Nuevo socio

El usuario introduce los datos de cada socio. Previamente, es necesaria la introducción del club al que va a pertenecer el socio. El número de licencia debe ser el código de un club existente (3 cifras) concatenado con su número de jugador.

LICENCIA	NOMBRE	PROVINCIA	TELEFONO	COD_CLUB	EMAIL
102002	MARI ESCRICHE, ANTONIO	VALENCIA	96-3407994	102	a@a.es
101028	DASI SALVADOR, MIGUEL	VALENCIA	160 03 87	101	
101043	AVALLONE BELLVER, LUIS	VALENCIA	333 12 35	101	

Figura 2.5. Nuevo socio

Tipos elementales de datos:

Nombre: varchar[50]
 Apellidos: varchar[100]
 Fecha nacimiento: date
 DNI: varchar[9]
 Dirección: varchar[100]
 Provincia: varchar[50]
 Población: varchar[50]
 CP: varchar[5]
 Teléfono: integer[10]
 Email: varchar[30]
 Código del Club: integer[3]
 Código clasificación: integer[10]

Tipo derivado de datos:

Numero Licencia: 'cod_club'+integer[3]

Ficheros accedidos:

Socio [lectura/escritura]
 Club[lectura]

2.3 Editar socio

El usuario puede editar los datos de cada socio, el programa comprueba que si se cambia el número de licencia, no este repetido y además si se cambian los 3 primeros dígitos, comprueba que el club exista.



Figura 2.6. Editar socio

Tipos elementales de datos:

Nombre:	varchar[50]
Apellidos:	varchar[100]
Fecha nacimiento:	date
DNI:	varchar[9]
Dirección:	varchar[100]
Provincia:	varchar[50]
Población:	varchar[50]
CP:	varchar[5]
Teléfono:	integer[10]
Email:	varchar[30]
Código del Club:	integer[3]
Código clasificación:	integer[10]

Tipo derivado de datos:

Numero Licencia:	'cod_club'+integer[3]
------------------	-----------------------

Ficheros accedidos:

- Socio [lectura/escritura]
- Club[lectura]
- Socio_Acta[lectura/escritura]

2.4 Borrar socio

El usuario puede borrar uno de los socios que se encuentren en la aplicación, seleccionándolo sobre la tabla y presionando el botón borrar.

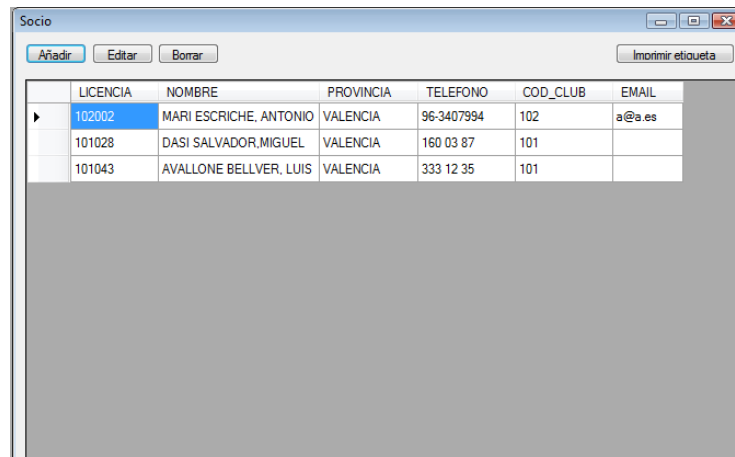


Figura 2.7. Borrar socio

Tipos elementales de datos:

Nombre: varchar[50]
 Apellidos: varchar[100]
 Fecha nacimiento: date
 DNI: varchar[9]
 Dirección: varchar[100]
 Provincia: varchar[50]
 Población: varchar[50]
 CP: varchar[5]
 Teléfono: integer[10]
 Email: varchar[30]
 Código del Club: integer[3]
 Código clasificación: integer[10]

Tipo derivado de datos:

Numero Licencia: 'cod_club'+integer[3]

Ficheros accedidos:

Socio [lectura/escritura]
 Socio_Acta[lectura/escritura]

2.5 Imprimir carnet socio

El usuario puede realizar la impresión de la etiqueta de los socios que se encuentren en la aplicación, seleccionándole sobre la tabla y presionando el botón *Imprimir etiqueta*.

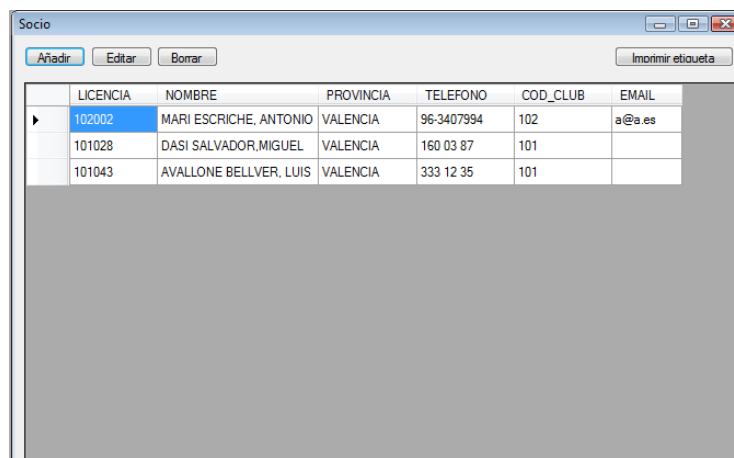


Figura 2.8. Imprimir etiqueta de socio

Tipos elementales de datos:

Nombre: varchar[50]
 Apellidos: varchar[100]
 Fecha nacimiento: date
 DNI: varchar[9]
 Dirección: varchar[100]
 Provincia: varchar[50]
 Población: varchar[50]
 CP: varchar[5]
 Teléfono: integer[10]
 Email: varchar[30]
 Código del Club: integer[3]
 Código clasificación: integer[10]

Tipo derivado de datos:

Numero Licencia: 'cod_club'+integer[3]

Ficheros accedidos:

Socio [lectura]

3) Gestión clubes:

Se pueden añadir clubs, se modifican las características de los existentes y se pueden borrar clubs.

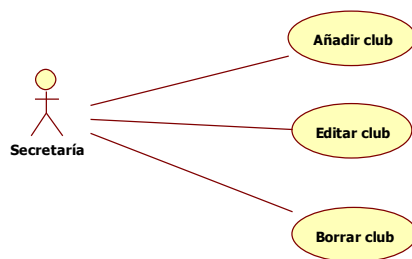


Figura 2.9. Gestión de clubes

3.1 Añadir club

El usuario introduce los datos de cada club. Si el código de club ya está en uso, al guardar el club el programa retorna un error.

Nota: De una parte, para que se pueda dar de alta un nuevo club, este debe encontrarse previamente introducido en la base de datos de la Federación Española de Billar. De otra parte, sería necesario que dicho club se encontrase registrado en la base de datos del Consell Valencia de l'Esport.

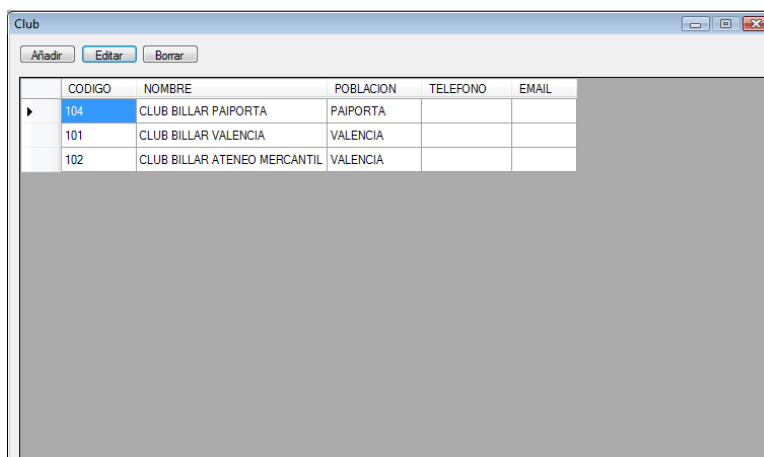


Figura 2.10. Añadir club

Tipos elementales de datos:

Código club: integer[3]
 Nombre: varchar[50]
 Dirección: varchar[10]
 CP: varchar[5]
 Población: varchar[50]
 Provincia: varchar[50]
 Teléfono: integer[10]
 Email: varchar[30]
 Pagina Web: varchar[100]
 Código clasificación: integer[10]

Ficheros accedidos:

Club [lectura/escritura]
 Clasificación_club[lectura/escritura]
 ClubFederacion[lectura]
 ClubMinisterio[lectura]

3.2 Editar club

El usuario puede editar los datos del club. Se comprueba que no haya ni códigos ni nombres repetidos. Además, si se modifica el código del club, se actualizan en cascada los datos de los socios, modificando sus tres primeras cifras y su campo código de club.

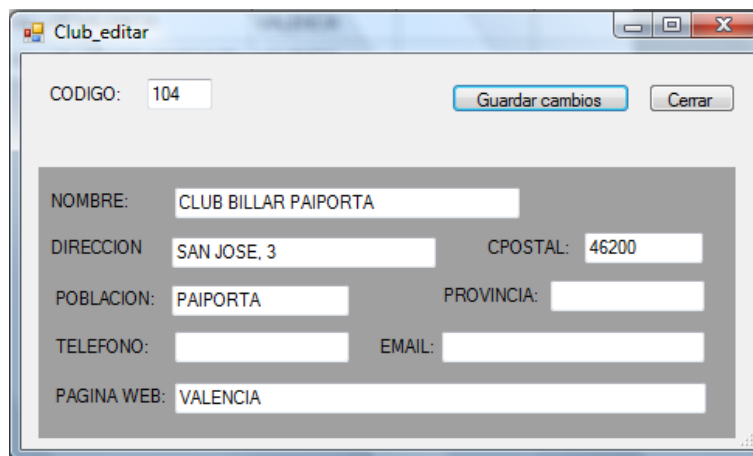


Figura 2.11. Editar club

Tipos elementales de datos:

Código club:	integer[3]
Nombre:	varchar[50]
Dirección:	varchar[10]
CP:	varchar[5]
Población:	varchar[50]
Provincia:	varchar[50]
Teléfono:	integer[10]
Email:	varchar[30]
Página Web:	varchar[100]
Código clasificación:	integer[10]

Ficheros accedidos

Club [lectura/escritura]
 Clasificación_club[lectura/escritura]

3.3 Borrar club

El usuario puede borrar uno de los clubs que se encuentren en la aplicación, seleccionándole sobre la tabla y presionando el botón borrar.

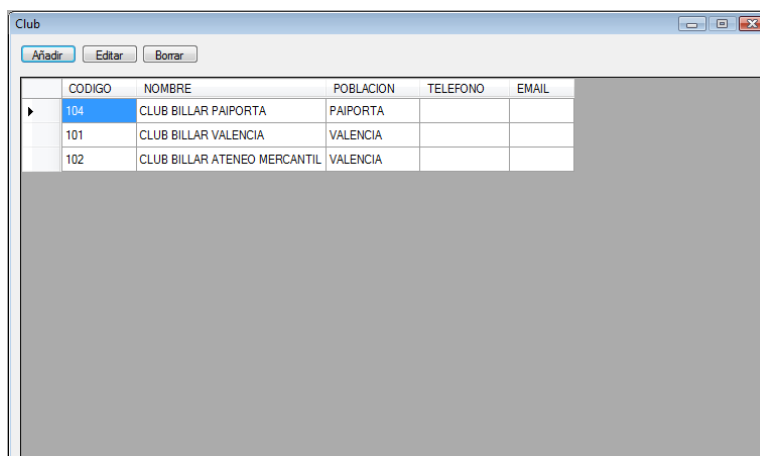


Figura 2.12. Borrar club

Tipos elementales de datos:

Código club: integer[3]
 Nombre: varchar[50]
 Dirección: varchar[10]
 CP: varchar[5]
 Población: varchar[50]
 Provincia: varchar[50]
 Teléfono: integer[10]
 Email: varchar[30]
 Pagina Web: varchar[100]
 Código clasificación: integer[10]

Ficheros accedidos:

Club [lectura/escritura]
 Clasificación_club[lectura/escritura]

4) Gestión actas:

Se crean actas nuevas de los partidos disputados y se pueden modificar las existentes. También se puede sancionar a los clubs o jugadores que no cumplan la normativa.

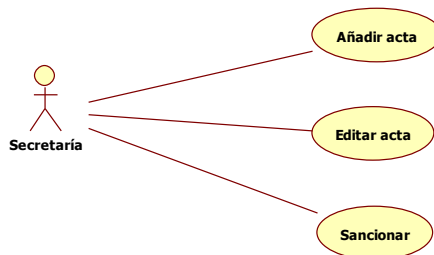


Figura 2.13. Gestión de actas

4.1 Añadir acta

El usuario introduce los datos que reflejan el resultado de un enfrentamiento entre dos clubes. Para ello, se introducen los resultados de las cuatro partidas disputadas.

Figura 2.14. Añadir acta

Tipos elementales de datos:

Categoría:	varchar[20]
Fecha:	date
Jornada:	integer[5]
División:	varchar[20]
Grupo:	varchar [5]
Club local:	varchar[50]
Club visitante:	varchar[50]
Número acta:	integer[10]
Temporada:	varchar[4]

Para cada uno de los resultados parciales:

Nombre del jugador 1:	varchar[50]
CAR del jugador 1:	integer[10]
ENT del jugador 1:	integer[10]
SM del jugador 1:	integer[10]
Nombre del jugador 2:	varchar[50]
CAR del jugador 2:	integer[10]
ENT del jugador 2:	integer[10]
SM del jugador 2:	integer[10]
Arbitro:	varchar[100]

Tipo derivado de datos:

Puntos del jugador 1:	integer[10]
Puntos del jugador 2:	integer[10]

Calculados a partir de los datos que haya en CAR,ENT y SM:

Numero Licencia jugador 1:	'cod_club'+integer[3]'
Numero Licencia jugador 2:	'cod_club'+integer[3]'

Ficheros accedidos:

- Acta [lectura/escritura]
- Socio[lectura]
- Socio_Acta[lectura/escritura]
- Club[lectura]
- Temporada[lectura]

4.2 Editar acta

El usuario introduce editará los datos que reflejan el resultado de un enfrentamiento entre dos clubes. Para ello, nos aparecen los resultados de los partidos disputados.

The screenshot shows a software window titled 'Acta_editar_detalle'. It contains several input fields and tables for editing match data. At the top, there are dropdown menus for 'CAT' (3 BANDAS), 'FECHA' (sábado, 22 de noviembre de 200...), 'JORNADA' (1), 'DIVISION' (HONOR), and 'GRUPO' (A). Below these are fields for 'Numero acta' (11) and 'Temporada' (1010/1011). The 'LOCAL' is 'CLUB BILLAR ATENEO MERCANTIL' and the 'VISITANTE' is 'CLUB BILLAR VALENCIA'. A section titled 'RESULTADOS PARCIALES' has a tabbed interface with tabs for frames 1, 2, 3, and 4. Below this is a table of player statistics:

LICENCIA	NOMBRE	CAR	ENT	PROM	SM	PUNTOS
102002	MARI ESCRICHE, ANTONIO	23	3	7,666	3	0
101028	DASI SALVADOR, MIGUEL	33		11	3	2

Below the player table is an 'ARBITRO' field with the value 'ccc'. At the bottom, a table shows 'RESULTADO DEL PARTIDO' with columns for 'NOMBRE DEL EQUIPO', 'P.PARCIALES', 'P.MATCH', 'CARAMB.', 'ENTRADAS', and 'PROM'.

NOMBRE DEL EQUIPO	P.PARCIALES	P.MATCH	CARAMB.	ENTRADAS	PROM
CLUB BILLAR ATENEO MERCANTIL	0	0	89	12	7,4166
CLUB BILLAR VALENCIA	8	2	132		11

Buttons for 'Guardar cambios' and 'Cancela' are located at the bottom right of the window.

Figura 2.15. Editar acta

Tipos elementales de datos:

- Categoría: varchar[20]
- Fecha: date
- Jornada: integer[5]
- División: varchar[20]
- Grupo: varchar [5]
- Club local: varchar[50]
- Club visitante: varchar[50]
- Número acta: integer[10]
- Temporada: varchar[4]

Para cada uno de los resultados parciales:

- Nombre del jugador 1: varchar[50]
- CAR del jugador 1: integer[10]
- ENT del jugador 1: integer[10]
- SM del jugador 1: integer[10]
- Nombre del jugador 2: varchar[50]
- CAR del jugador 2: integer[10]

NT del jugador 2: integer[10]
 SM del jugador 2: integer[10]
 Arbitro: varchar[100]

Tipo derivado de datos:

Puntos del jugador 1: integer[10]
 Puntos del jugador 2: integer[10]

Calculados a partir de los datos que haya en CAR,ENT y SM:

Numero Licencia jugador 1: 'cod_club'+integer[3]
 Numero Licencia jugador 2: 'cod_club'+integer[3]

Ficheros accedidos

Acta [lectura/escritura]
 Socio[lectura]
 Socio_Acta[lectura/escritura]
 Club[lectura]
 Temporada[lectura]

4.3 Borrar acta

El usuario puede borrar una de las actas que se encuentren en la aplicación, seleccionándola sobre la tabla y presionando el botón *Borrar*.

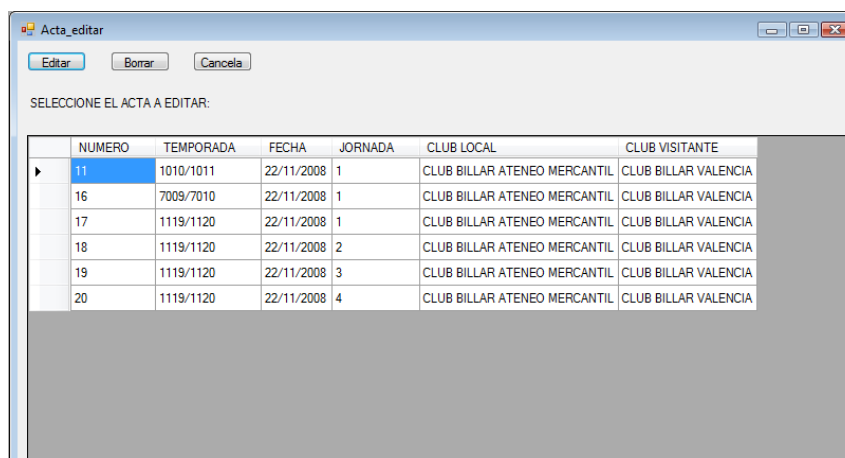


Figura 2.16. Borrar acta

Tipos elementales de datos:

Categoría: varchar[20]
 Fecha: date
 Jornada: integer[5]
 División: varchar[20]
 Grupo: varchar [5]
 Club local: varchar[50]
 Club visitante: varchar[50]
 Número acta: integer[10]
 Temporada: varchar[4]

Para cada uno de los resultados parciales:

Nombre del jugador 1:	varchar[50]
CAR del jugador 1:	integer[10]
ENT:	integer[10]
SM del jugador 1:	integer[10]
Nombre del jugador 2:	varchar[50]
CAR del jugador 2:	integer[10]
SM del jugador 2:	integer[10]
Puntos del jugador 1:	integer[10]
Puntos del jugador 2:	integer[10]
Arbitro:	varchar[100]

Tipo derivado de datos:

Numero Licencia jugador 1:	'cod_club'+integer[3]
Numero Licencia jugador 2:	'cod_club'+integer[3]

Ficheros accedidos:

- Acta [lectura/escritura]
- Socio[lectura]
- Socio_Acta[lectura/escritura]
- Club[lectura]
- Temporada[lectura]

4.4 Sancionar

Se introduce la modalidad de juego, categoría, el grupo, el club y los puntos con los que se le sancionará.

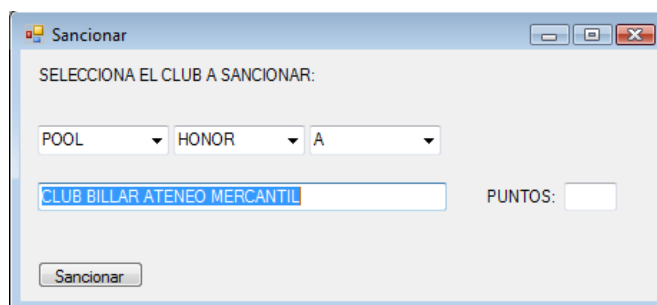


Figura 2.17. Sancionar

Tipos elementales de datos:

Categoría:	varchar[20]
División:	varchar[20]
Grupo:	varchar[5]
Club:	varchar[50]
Puntos:	integer[10]

Ficheros accedidos:

- Acta [lectura/escritura]
- Socio_Acta[lectura/escritura]

Club[lectura]

5) Gestión listados:

Se pueden ver consultar las estadísticas de los jugadores y de los clubs. También pueden exportarse a Excel.

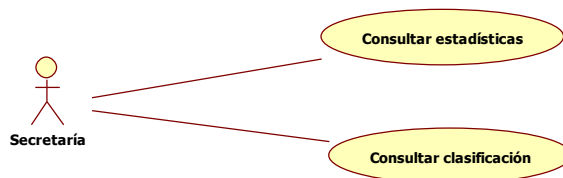


Figura 2.18. Gestión de listados

5.1 Consultar estadísticas

Permite visualizar las estadísticas de los clubs clasificados por divisiones y de los jugadores, según la opción que seleccionemos.

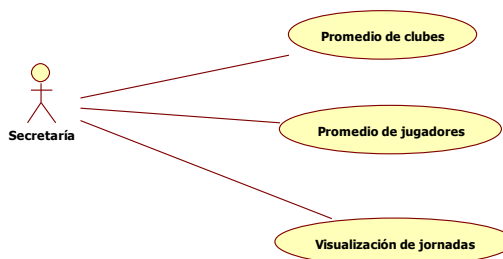


Figura 2.19. Consultar estadísticas

5.2 Promedio clubs

Seleccionamos la división de la cual queremos visualizar las estadísticas.

EQUIPO	PJ	PG	PE	PP	CAR	ENT	PROM	PPA	PMA	SANC	PUNT
CLUB BILLAR ATENEO MERCANTIL	1	0	1	0	144	4	36	4	1	0	1
CLUB BILLAR VALENCIA	1	0	1	0	144	4	36	4	1	0	1

Figura 2.20. Promedio de clubs

Tipos elementales de datos:

- Club: varchar[50]
- PJ: integer[10]
- PG: integer[10]
- PE: integer[10]

PP: integer[10]
 CAR: integer[10]
 ENT: integer[10]
 PROM: integer[10]
 PPA: integer[10]
 PMA: integer[10]
 SANC: integer[10]
 PUNT: integer[10]

Ficheros accedidos:

Clasificación_club[lectura]
 Club[lectura]
 Temporada[lectura]

5.3 Promedio jugadores

Seleccionamos la división de la cual queremos visualizar las estadísticas.

	NOMBRE	PJ	PG	PE	PP	P	SM	CAR	ENT	PROM
▶	MARI ESCRICHE, ANTONIO	8	-5	16	-3	6	10	398	30	13.266666666666666
	AVALLONE BELLVER, LUIS	4	-4	11	-3	3	6	56	6	9.333333333333333
	DASI SALVADOR, MIGUEL	4	-4	7	1	-1	4	44	24	1.8333333333333333

Figura 2.21. Promedio de jugadores

Tipos elementales de datos:

Nombre: varchar[50]
 PJ: integer[10]
 PG: integer[10]
 PE: integer[10]
 PP: integer[10]
 P: integer[10]
 CAR: integer[10]
 ENT: integer[10]
 PROM: integer[10]

Ficheros accedidos:

Clasificación_jugador [lectura]
 Socio[lectura]
 Temporada[lectura]

5.4 Visualización de jornadas

Permite al usuario visualizar las jornadas que se han disputado.

FECHA	LOCAL	PM	PC	VISITANTE
22/11/2008	CLUB BILLAR ATENEO MERCANTIL	0	89	CLUB BILLAR VALENCIA
22/11/2008	CLUB BILLAR ATENEO MERCANTIL	2	88	CLUB BILLAR VALENCIA
22/11/2008	CLUB BILLAR ATENEO MERCANTIL	2	254	CLUB BILLAR VALENCIA
22/11/2008	CLUB BILLAR ATENEO MERCANTIL	1	144	CLUB BILLAR VALENCIA
22/11/2008	CLUB BILLAR ATENEO MERCANTIL	2	144	CLUB BILLAR VALENCIA
22/11/2008	CLUB BILLAR ATENEO MERCANTIL	2	144	CLUB BILLAR VALENCIA

Figura 2.22. Visualización de jornadas

Tipos elementales de datos:

- Fecha: date
- Club local: varchar[50]
- PM: integer[10]
- PC: integer[10]
- Club visitante: varchar[50]
- PM: integer[10]
- PC: integer[10]
- PE: integer[10]

Ficheros accedidos:

- Acta[lectura]
- Club[lectura]
- Socio_acta[lectura]
- Temporada[lectura]

5.5 Consultar clasificación

Seleccionamos la categoría y la división de la cual queremos consultar la clasificación y presionamos sobre Ver clasificación.

EQUIPO	PJ	PG	PE	PP	CAR	ENT	PROM	PPA	PMA	SANC	PUNT
CLUB BILLAR ATENEO MERCANTIL	2	2	0	0	388	30	13.266666666666668	10	4	0	4
CLUB BILLAR VALENCIA	2	0	0	2	100	30	3.3333333333333335	7	0	0	0

Figura 2.23. Consultar clasificación

Tipos elementales de datos

- Club : varchar[50]
- PJ: integer[10]
- PG: integer[10]

PE:	integer[10]
PP:	integer[10]
CAR:	integer[10]
ENT:	integer[10]
PROM:	integer[10]
PPA:	integer[10]
PMA:	integer[10]
SANC:	integer[10]
PUNT:	integer[10]

Ficheros accedidos

Clasificación_club[lectura]

3. Diseño físico de datos

“En esta actividad se define la estructura física de datos que utilizará el sistema, a partir del modelo lógico de datos normalizado o modelo de clases, de manera que teniendo presentes las características específicas del sistema de gestión de datos concreto a utilizar, los requisitos establecidos para el sistema de información, y las particularidades del entorno tecnológico, se consiga una mayor eficiencia en el tratamiento de los datos.

También se analizan los caminos de acceso a los datos utilizados por cada módulo/clase del sistema en consultas y actualizaciones, con el fin de mejorar los tiempos de respuesta y optimizar los recursos de máquina.

Las tareas de esta actividad se realizan de forma iterativa y en paralelo con las realizadas en las actividades Definición de la Arquitectura del Sistema (DSI 1), dónde se especifican los detalles de arquitectura e infraestructura y la planificación de capacidades, Diseño de la Arquitectura de Soporte (DSI 2), dónde se determinan y diseñan los servicios comunes que pueden estar relacionados con la gestión de datos (acceso a bases de datos, ficheros, áreas temporales, sincronización de bases de datos, etc.), Diseño de Casos de Uso Reales y de Clases (DSI 3 y 4), para desarrollo orientado a objetos, y Diseño de la Arquitectura de Módulos del Sistema (DSI 5), para desarrollo estructurado, dónde se especifica la lógica de tratamiento y las interfaces utilizadas.

En el caso de diseño orientado a objetos, esta actividad también es necesaria. La obtención del modelo físico de datos se realiza aplicando una serie de reglas de transformación a cada elemento del modelo de clases que se está generando en la actividad Diseño de Clases (DSI 4).

Asimismo, en esta actividad hay que considerar los estándares y normas establecidos para el diseño aplicando, cuando proceda, los mecanismos genéricos de diseño identificados en la tarea Identificación de Mecanismos Genéricos de Diseño (DSI 2.2).”

Diseño físico de datos:

Para la consulta del diseño físico de la BB.DD. se remite al **anexo V - DISEÑO BB.DD.**

4. Diseño de la migración y carga inicial de datos

“Esta actividad sólo se lleva a cabo cuando es necesaria una carga inicial de información, o una migración de datos de otros sistemas, cuyo alcance y estrategia a seguir se habrá establecido previamente.

Para ello, se toma como referencia el plan de migración y carga inicial de datos, que recoge las estructuras físicas de datos del sistema o sistemas origen implicadas en la conversión, la prioridad en las cargas y secuencia a seguir, las necesidades previas de depuración de la información, así como los requisitos necesarios para garantizar la correcta implementación de los procedimientos de migración sin comprometer el funcionamiento de los sistemas actuales.

A partir de dicho plan, y de acuerdo a la estructura física de los datos del nuevo sistema, obtenida en la actividad Diseño Físico de Datos (DSI 6), y a las características de la arquitectura y del entorno tecnológico propuesto en la actividad Definición de la Arquitectura del Sistema (DSI 1), se procede a definir y diseñar en detalle los procedimientos y procesos necesarios para realizar la migración.

Se completa el plan de pruebas específico establecido en el plan de migración y carga inicial, detallando las pruebas a realizar, los criterios de aceptación o rechazo de la prueba y los responsables de la organización, realización y evaluación de resultados.

Asimismo, se determinan las necesidades adicionales de infraestructura, tanto para la implementación de los procesos como para la realización de las pruebas.

Como resultado de esta actividad, se actualiza el plan de migración y carga inicial de datos con la información siguiente:

- Especificación del entorno de migración.
- Definición de procedimientos de migración.
- Diseño detallado de módulos.
- Especificación técnica de las pruebas.
- Planificación de la migración y carga inicial.”

Diseño de la migración de datos:

Queda establecido en el contrato que la migración de datos la realizará el personal de la FBCV, si bien se prestará apoyo y soporte para resolver dudas sobre restricciones, campos obligatorios / no obligatorios, tipos de datos, etc., etc.

5. Especificación técnica del plan de pruebas

“En esta actividad se realiza la especificación de detalle del plan de pruebas del sistema de información para cada uno de los niveles de prueba establecidos en el proceso Análisis del Sistema de Información:

- Pruebas unitarias.
- Pruebas de integración.
- Pruebas del sistema.
- Pruebas de implantación.
- Pruebas de aceptación.

Para ello se toma como referencia el plan de pruebas, que recoge los objetivos de la prueba de un sistema, establece y coordina una estrategia de trabajo, y provee del marco adecuado para planificar paso a paso las actividades de prueba.”

Plan de pruebas:

Microsoft® Visual Studio 2008 permite la creación de un proyecto de pruebas unitarias, este proyecto se usará en cada función de negocio del programa con una configuración específica para cada módulo. El resto de pruebas se realizarán sin ningún programa de pruebas específico.

Para las pruebas de los diálogos, se tendrá en cuenta como mínimo:

- Cumplimiento de los estándares de diseño
- Ortografía y formato
- Entrada / Salida
- Rendimiento

En casos especiales estas pruebas serán ampliables.

Además, todas las pruebas de cada tipo se documentarán en un archivo de texto que contendrá los siguientes apartados:

- Fecha de la prueba
- Nombre de la función de negocio probada
- Resumen de las pruebas
- Configuración del probador
- Precondiciones
- Detalle de las pruebas
- Autor de la prueba
- Estado de las prueba
- Revisiones

En el **anexo VI – Pruebas Socio** se puede consultar un ejemplo aplicado de un documento de pruebas de interfaz unitaria para la aplicación.

6. Especificación de requisitos de documentación de usuario

“En esta tarea se recoge toda la información necesaria para la especificación de la documentación a entregar al usuario, que incluirá los manuales de usuario y, cuando proceda, los manuales de explotación. Para ello, es necesario definir, entre otros, los siguientes aspectos:

- Tipo de documentos y estándares a seguir en la elaboración de los mismos.
- Formato en el que se desarrollarán.
- Estructura.
- Soporte en el que se van a generar.
- Distribución y mantenimiento de la documentación y copias a editar.
- Control de versiones.”

Plan de documentación:

La documentación que se entregará una vez finalizada la implantación consistirá en el manual de usuario en formato PDF, que será incluido en una unidad CDROM. El manual se compondrá en su estructura básica de:

- Introducción
- Instalación y desinstalación del paquete
- Descripción del entorno de trabajo
- Funciones del programa
- Solución de errores

Anexo II-5. Índice:

1.	Preparación del entorno de generación y construcción	3
1.1.	Implantación de la base de datos física o ficheros.....	3
1.2.	Preparación del entorno de construcción.....	4
2.	Generación del código de los componentes y procedimientos.....	4
2.1.	Generación del código de componentes	5
3.	Ejecución de las pruebas unitarias	5
4.	Ejecución de las pruebas de integración.....	6
5.	Ejecución de las pruebas de sistema.....	6
6.	Elaboración de los manuales de usuario.....	7
7.	Definición de la formación de usuarios finales	8
8.	Construcción de los componentes y procedimientos de migración	8
9.	Aprobación del sistema de información	9

• Construcción del sistema de información (CSI)

“En este proceso se genera el código de los componentes del Sistema de Información, se desarrollan todos los procedimientos de operación y seguridad y se elaboran todos los manuales de usuario final y de explotación con el objetivo de asegurar el correcto funcionamiento del Sistema para su posterior implantación.

Para conseguir dicho objetivo, en este proceso se realizan las pruebas unitarias, las pruebas de integración de los subsistemas y componentes y las pruebas del sistema, de acuerdo al plan de pruebas establecido.

Asimismo, se define la formación de usuario final y, si procede, se construyen los procedimientos de migración y carga inicial de datos. Al ser MÉTRICA Versión 3 una metodología que cubre tanto desarrollos estructurados como orientados a objetos, las actividades de ambas aproximaciones están integradas en una estructura común.

El producto Especificaciones de Construcción del Sistema de Información, obtenido en la actividad de Generación de Especificaciones de Construcción (DSI 8), es la base para la construcción del sistema de información. En dicho producto se recoge la información relativa al entorno de construcción del sistema de información, la especificación detallada de los componentes y la descripción de la estructura física de datos, tanto bases de datos como sistemas de ficheros. Opcionalmente, incluye un plan de integración del sistema de información, en el que se especifica la secuencia y organización de la construcción de los distintos componentes.

En la actividad Preparación del Entorno de Generación y Construcción (CSI 1), se asegura la disponibilidad de la infraestructura necesaria para la generación del código de los componentes y procedimientos del sistema de información.

Una vez configurado el entorno de construcción, se realiza la codificación y las pruebas de los distintos componentes que conforman el sistema de información, en las actividades:

- Generación del Código de los Componentes y Procedimientos (CSI 2), que se hace según las especificaciones de construcción del sistema de información, y conforme al plan de integración del sistema de información
- Ejecución de las Pruebas Unitarias (CSI 3), dónde se llevan a cabo las verificaciones definidas en el plan de pruebas para cada uno de los componentes
- Ejecución de las Pruebas de Integración (CSI 4), que incluye la ejecución de las verificaciones asociadas a los subsistemas y componentes, a partir de los componentes verificados individualmente, y la evaluación de los resultados.

Una vez construido el sistema de información y realizadas las verificaciones correspondientes, se lleva a cabo la integración final del sistema de información en la actividad Ejecución de las Pruebas del Sistema (CSI 5), comprobando tanto las interfaces entre subsistemas y sistemas externos como los requisitos, de acuerdo a las verificaciones establecidas en el plan de pruebas para el nivel de pruebas del sistema. En la actividad Elaboración de los Manuales de Usuario (CSI 6), se genera la documentación de usuario final o explotación, conforme a los requisitos definidos en el proceso Diseño del Sistema de Información.

La formación necesaria para que los usuarios finales sean capaces de utilizar el sistema de forma satisfactoria se especifica en la actividad Definición de la Formación de Usuarios Finales (CSI 7).

Si se ha establecido la necesidad de realizar una migración de datos, la construcción y pruebas de los componentes y procedimientos relativos a dicha migración y a la carga inicial de datos se realiza en la actividad Construcción de los Componentes y Procedimientos de Migración y Carga Inicial de Datos (CSI 8)".

1. Preparación del entorno de generación y construcción

"El objetivo de esta actividad es asegurar la disponibilidad de todos los medios y facilidades para que se pueda llevar a cabo la construcción del sistema de información. Entre estos medios, cabe destacar la preparación de los puestos de trabajo, equipos físicos y lógicos, gestores de bases de datos, bibliotecas de programas, herramientas de generación de código, bases de datos o ficheros de prueba, entre otros.

Las características del entorno de construcción y sus requisitos de operación y seguridad, así como las especificaciones de construcción de la estructura física de datos, se establecen en la actividad Generación de Especificaciones de Construcción (DSI 8), y constituyen el punto de partida para la realización de esta actividad".

Tareas propuestas para esta actividad:

- 1.1 Implantación de la base de datos física o ficheros
- 1.2 Preparación del entorno de construcción

1.1. Implantación de la base de datos física o ficheros

Objetivo:

"En esta tarea hay que:

- Crear los elementos del sistema gestor de base de datos o sistema de ficheros
- Reservar el espacio de almacenamiento, definiendo, entre otros, los dispositivos físicos a emplear, tamaño de los bloques, tipo de registro físico, zona de desbordamiento, opciones de almacenamiento de datos, etc.
- Inicializar la base de datos o ficheros, cargando los datos considerados necesarios en el espacio de almacenamiento previamente definido".

Implantación de la base de datos:

Siguiendo las especificaciones descritas en el DSI, se crea la BB.DD. partiendo de cero mediante el programa Microsoft® Access 2007. Una vez creada, se realizan las primeras cargas para comprobar restricciones de integridad, y se asigna la cadena de conexión en el desarrollo principal del programa.

1.2. Preparación del entorno de construcción

Objetivo:

“En esta tarea se prepara el entorno en el que se construirán los componentes del sistema de información, contemplando aspectos tales como:

- Bibliotecas o librerías a utilizar
- Herramientas: generadores de código, editores, compiladores, verificadores sintácticos, montadores de enlace
- Puestos de trabajo
- Implementación de los procedimientos de operación y seguridad propios del entorno de construcción, de acuerdo a los requisitos de seguridad y operación establecidos en la tarea Especificación del Entorno de Construcción (DSI 8.1)”.

Preparación del entorno:

El desarrollo del software de gestión se realizará mediante la plataforma .NET. Para ello se hace necesaria la instalación de los siguientes paquetes:

- Microsoft[®] Visual Studio 2008 Service Pack 1 (SP1)
- Microsoft[®] .NET Framework 3.5 SP1

Se instalará en una maquina recomendada por Microsoft[®], esto es con las siguientes características:

- 1GB RAM
- procesador 2.2 GHz
- 600mb libres HD

La BBDD se realizará en formato Access 2007, lo que conlleva la instalación del paquete:

- Microsoft[®] Access 2007

Además para la generación de autoinstalables, se piensa que la herramienta gratuita de publicación proporcionada por Microsoft no es adecuada. Por tanto, se instalará adicionalmente:

- InstallShield[®] Premium 2010

El resto de herramientas necesarias para la elaboración del software, como es Crystal Report[®] .NET, vienen integradas por defecto en la instalación estándar de Visual Studio 2008.

2. Generación del código de los componentes y procedimientos

“El objetivo de esta actividad es la codificación de los componentes del sistema de información, a partir de las especificaciones de construcción obtenidas en el proceso Diseño del Sistema de Información (DSI), así como la construcción de los procedimientos de operación y seguridad establecidos para el mismo.

En paralelo a esta actividad, se desarrollan las actividades relacionadas con las pruebas unitarias y de integración del sistema de información. Esto permite una construcción

incremental, en el caso de que así se haya especificado en el plan de pruebas y en el plan de integración del sistema de información”.

Tareas propuestas para esta actividad:

2.1 Generación del código de componentes

2.1. Generación del código de componentes

Objetivo:

“En esta tarea se genera el código correspondiente a cada uno de los componentes del sistema de información, identificados en la tarea Definición de Componentes y Subsistemas de Construcción (DSI 8.2).

Para generar el código fuente se tienen en cuenta los estándares de nomenclatura, codificación y calidad utilizados por la organización y recogidos en el catálogo de normas.

Con el fin de verificar que el código fuente especifica de forma correcta el componente, se realiza su ensamblaje o compilación, verificando y corrigiendo los errores sintácticos, y el enlace del código objeto obtenido con las correspondientes bibliotecas”.

Generación del código fuente:

El código generado puede consultarse en los anexos código fuente incluidos en el **anexo VII – Código fuente**.

3. Ejecución de las pruebas unitarias

“En esta actividad se realizan las pruebas unitarias de cada uno de los componentes del sistema de información, una vez codificados, con el objeto de comprobar que su estructura es correcta y que se ajustan a la funcionalidad establecida.

En el plan de pruebas se ha definido el entorno necesario para la realización de cada nivel de prueba, así como las verificaciones asociadas a las pruebas unitarias, la coordinación y secuencia a seguir en la ejecución de las mismas y los criterios de registro y aceptación de los resultados”.

Ejecución de pruebas unitarias:

Para la ejecución de las pruebas unitarias se utiliza un proyecto de pruebas unitarias integrado de Visual Studio 2008. En este proyecto se lanzan las pruebas de las funciones y procedimientos del sistema desarrollado. Se utiliza además el nuevo soporte integrado de Visual Studio 2008 para diagnosticar y optimizar el sistema a través de las herramientas de pruebas.

4. Ejecución de las pruebas de integración

“El objetivo de las pruebas de integración es verificar si los componentes o subsistemas interactúan correctamente a través de sus interfaces, tanto internas como externas, cubren la funcionalidad establecida, y se ajustan a los requisitos especificados en las verificaciones correspondientes.

La estrategia a seguir en las pruebas de integración se establece en el plan de pruebas, dónde se habrá tenido en cuenta el plan de integración del sistema de información, siempre y cuando se haya especificado en la tarea Definición de Componentes y Subsistemas de Construcción (DSI 8.2).

Esta actividad se realiza en paralelo a las actividades Generación del Código de los Componentes y Procedimientos (CSI 2) y Ejecución de las Pruebas Unitarias (CSI 3). Sin embargo, es necesario que los componentes objeto de las pruebas de integración se hayan verificado de manera unitaria”.

Pruebas de integración:

Las pruebas de integración se realizan de forma implícita con el prototipo de la aplicación. Las pruebas se documentan en formularios tipo como se describió en el anexo DSI, y contienen:

- Fecha de la prueba
- Nombre de la función de negocio probada
- Resumen de las pruebas
- Configuración del probador
- Precondiciones
- Detalle de las pruebas
- Autor de la prueba
- Estado de las prueba
- Revisiones

5. Ejecución de las pruebas de sistema

“El objetivo de las pruebas del sistema es comprobar la integración del sistema de información globalmente, verificando el funcionamiento correcto de las interfaces entre los distintos subsistemas que lo componen y con el resto de sistemas de información con los que se comunica.

En la realización de estas pruebas es importante comprobar la cobertura de los requisitos, dado que su incumplimiento puede comprometer la aceptación del sistema por el equipo de operación responsable de realizar las pruebas de implantación del sistema, que se llevarán a cabo en el proceso Implantación y Aceptación del Sistema”.

Pruebas de sistema:

Las pruebas de sistema permiten verificar el conjunto completo de la aplicación. Se siguen los siguientes pasos para realizarlas:

- Revisar la verificabilidad del requerimiento
- Especificar el criterio de verificación
- Hacer visible las propiedades o elementos del software necesarios para verificar el cumplimiento del requerimiento
- Hacer controlable los elementos del software necesarios para llevar a cabo las pruebas
- Elaborar el plan de pruebas
- Ejecutar el plan de pruebas y reportar sus resultados

A su vez, cada prueba de sistema deberá contemplar las siguientes pruebas:

- **Pruebas funcionales:** si el requisito está contemplado correctamente en el programa.
- **Pruebas de comunicación:** si la comunicación entre componentes funciona correctamente.
- **Pruebas de rendimiento:** si el tiempo de respuesta de la aplicación es aceptable.
- **Pruebas de volumen:** si el rendimiento no se ve afectado con un gran volumen de datos.
- **Pruebas de sobrecarga:** si el sistema maneja excepciones provocadas por sobrecarga de tipos y recursos.
- **Pruebas de disponibilidad de datos:** si el sistema puede recuperarse ante fallos.
- **Pruebas de facilidad de uso:** si el sistema se adapta correctamente al usuario.
- **Pruebas de operación:** si los procedimientos de operación del sistema funcionan correctamente.
- **Pruebas de entorno:** si el software funciona correctamente en el entorno instalado.

6. Elaboración de los manuales de usuario

“El objetivo de esta tarea es elaborar la documentación de usuario, tanto usuario final como de explotación, de acuerdo a los requisitos establecidos en la tarea Especificación de Requisitos de Documentación de Usuario (DSI 11.1), y recogidos en el catálogo de requisitos. Los requisitos de documentación especifican aspectos relativos a los tipos de documentos a elaborar y estándares a seguir en la generación de los mismos, y para cada uno de ellos:

- Formato y soporte en el que se desarrollarán
- Estructura
- Distribución y mantenimiento de la documentación y número de copias a editar”.

Características del manual de usuario:

El manual de usuario se compondrá básicamente de:

- Introducción
- Instalación y desinstalación del paquete
- Descripción del entorno de trabajo
- Funciones del programa
- Solución de errores

El formato será Adobe® PDF y se entregará en la unidad CDROM con el resto del paquete. No obstante, en caso de requerirlo se podrá hacer entrega de copias impresas.

7. Definición de la formación de usuarios finales

“En esta actividad se establecen las necesidades de formación del usuario final, con el objetivo de conseguir la explotación eficaz del nuevo sistema. Para la definición de la formación hay que tener en cuenta las características funcionales y técnicas propias del sistema de información, así como los requisitos relacionados con la formación del usuario final, establecidos en la tarea Especificación de Requisitos de Implantación (DSI 11.2).

El producto resultante de esta actividad es la especificación de la formación de usuarios finales, que consta de los siguientes elementos:

- Esquema de formación
- Materiales y entornos de formación.

En el proceso Implantación y Aceptación del Sistema (IAS), se unifican las especificaciones de formación de cada sistema de información implicado en la implantación y se elabora un único plan de formación que esté alineado con el plan de implantación del sistema”.

Plan de formación:

El plan de formación se basará en sesiones dirigidas al personal del departamento donde se actualizará el sistema de información.

El plan se dividirá en dos sesiones teórico-prácticas, donde se expondrán casos reales de uso y se explicará el funcionamiento completo de la aplicación.

Para el plan se hará uso de material de formación como son las diapositivas electrónicas.

8. Construcción de los componentes y procedimientos de migración

“El objetivo de esta actividad es la codificación y prueba de los componentes y procedimientos de migración y carga inicial de datos, a partir de las especificaciones recogidas en el plan de migración y carga inicial de datos obtenido en el proceso Diseño del Sistema de Información.

Previamente a la generación del código, se prepara la infraestructura tecnológica necesaria para realizar la codificación y las pruebas de los distintos componentes y procedimientos asociados, de acuerdo a las características del entorno de migración especificado en el plan de migración y carga inicial de datos.

Finalmente, se llevan a cabo las verificaciones establecidas en la especificación técnica del plan de pruebas propio de la migración”.

Migración:

Como se ha especificado en el anexo DSI, queda establecido en el contrato que la migración de datos la realizará el personal de la FBCV, si bien se prestará apoyo y soporte para resolver dudas sobre restricciones, campos obligatorios / no obligatorios, tipos de datos, etc., etc...

9. Aprobación del sistema de información

“En esta tarea se recopilan los productos del sistema de información y se presentan al Comité de Seguimiento para su aprobación.”

Aprobación del sistema de información:

Tras la reunión establecida con el responsable del proyecto se decide aceptar el prototipo del sistema de información presentado y continuar con la implantación en el entorno del departamento.

Anexo II-6. Índice:

1.	Establecimiento del plan de implantación	3
1.1.	Definición del plan de implantación.....	3
2.	Incorporación del sistema al entorno de operación	4
2.1.	Preparación de la instalación	5
2.2.	Realización de la instalación.....	6
3.	Carga de datos al entorno de operación	6
3.1.	Migración y carga inicial de datos	7
4.	Pruebas de implantación del sistema	7
5.	Pruebas de aceptación del sistema.....	8
6.	Preparación del mantenimiento del sistema	8
7.	Presentación y aprobación del sistema	9
8.	Paso a producción	9

• Implantación y aceptación del sistema (IAS)

“Este proceso tiene como objetivo principal la entrega y aceptación del sistema en su totalidad, y la realización de todas las actividades necesarias para el paso a producción del mismo.

En primer lugar, se revisa la estrategia de implantación que ya se determinó en el proceso Estudio de Viabilidad del Sistema (EVS). Se estudia su alcance y, en función de sus características, se define un plan de implantación y se especifica el equipo que lo va a llevar a cabo. Conviene señalar la participación del usuario de operación en las pruebas de implantación, del usuario final en las pruebas de aceptación, y del responsable de mantenimiento.

Las actividades previas al inicio de la producción incluyen la preparación de la infraestructura necesaria para configurar el entorno, la instalación de los componentes, la activación de los procedimientos manuales y automáticos asociados y, cuando proceda, la migración o carga inicial de datos. Para ello se toman como punto de partida los productos software probados, obtenidos en el proceso Construcción del Sistema de Información (CSI) y su documentación asociada.

Se realizan las pruebas de implantación y de aceptación del sistema en su totalidad. Responden a los siguientes propósitos:

- Las pruebas de implantación cubren un rango muy amplio, que va desde la comprobación de cualquier detalle de diseño interno hasta aspectos tales como las comunicaciones. Se debe comprobar que el sistema puede gestionar los volúmenes de información requeridos, se ajusta a los tiempos de respuesta deseados y que los procedimientos de respaldo, seguridad e interfaces con otros sistemas funcionan correctamente. Se debe verificar también el comportamiento del sistema bajo las condiciones más extremas.
- Las pruebas de aceptación se realizan por y para los usuarios. Tienen como objetivo validar formalmente que el sistema se ajusta a sus necesidades. Asimismo, se llevan a cabo las tareas necesarias para la preparación del mantenimiento, siempre y cuando se haya decidido que éste va a efectuarse. En cualquier caso, es necesario que la persona que vaya a asumir el mantenimiento conozca el sistema, antes de su incorporación al entorno de producción.

Además hay que determinar los servicios (y niveles para cada uno) que requiere el sistema que se va a implantar, y el acuerdo que se adquiere una vez que se inicie la producción. Hay que distinguir entre servicios de gestión de operaciones (servicios por lotes, seguridad, comunicaciones, etc.) y servicios al cliente (servicio de atención al usuario, mantenimiento, etc.) que se deben negociar en cuanto a recursos, horarios, coste, etc. Se fija el nivel con el que se prestará el servicio como indicador de la calidad del mismo.

Conviene señalar que la implantación puede ser un proceso iterativo que se realiza de acuerdo al plan establecido para el comienzo de la producción del sistema en su entorno de operación. Para establecer este plan se tiene en cuenta:

- El cumplimiento de los requisitos de implantación definidos en la actividad Establecimiento de Requisitos (ASI 2) y especificados en la actividad Establecimiento de Requisitos de Implantación (DSI 11).

- La estrategia de transición del sistema antiguo al nuevo.

Finalmente, se realizan las acciones necesarias para el inicio de la producción”.

1. Establecimiento del plan de implantación

“En esta actividad se revisa la estrategia de implantación para el sistema, establecida inicialmente en el proceso Estudio de Viabilidad del Sistema (EVS). Se identifican los distintos sistemas de información que forman parte del sistema objeto de la implantación. Para cada sistema se analizan las posibles dependencias con otros proyectos, que puedan condicionar el plan de implantación.

Una vez estudiado el alcance y los condicionantes de la implantación, se decide si ésta se puede llevar a cabo. Será preciso establecer, en su caso, la estrategia que se concretará de forma definitiva en el plan de implantación.

Se constituye el equipo de implantación, determinando los recursos humanos necesarios para la propia instalación del sistema, para las pruebas de implantación y aceptación, y para la preparación del mantenimiento. Se identifican, para cada uno de ellos, sus perfiles y niveles de responsabilidad.”

Tareas propuestas para esta actividad:

1.1. Definición del plan de implantación

1.1. Definición del plan de implantación

Objetivo:

“La estrategia de implantación del sistema se habrá determinado en la tarea Evaluación de las Alternativas y Selección (EVS 6.2) del proceso Estudio de Viabilidad del Sistema (EVS), en función de la envergadura del sistema, es decir el número de sistemas de información implicados en la implantación y la cobertura geográfica, cuyo alcance depende de las características y complejidad de los sistemas de información que conforman el sistema objeto de la implantación.

Se revisan los requisitos de implantación (instalación, infraestructura, formación) establecidos en la tarea Especificación de Requisitos de Implantación (DSI 11.2) y los procedimientos implicados en la implantación, establecidos para cada uno de los sistemas de información en la tarea Especificación de Requisitos de Operación y Seguridad (DSI 1.7), con el fin de asegurar su adecuación a la estrategia global de implantación.

Una vez analizada la información anterior, se define un plan de implantación que permita calcular adecuadamente el esfuerzo y los recursos necesarios para llevar a cabo con éxito la implantación. Dicho plan debe contemplar todas las tareas relacionadas con:

- La formación necesaria para la implantación, tanto a usuarios finales como al equipo que se encarga de realizar las pruebas de implantación y aceptación del sistema.

- La preparación de la infraestructura necesaria para la incorporación del sistema al entorno de operación.
- La instalación de todos los componentes y procedimientos manuales y automáticos asociados a cada sistema de información implicado en la implantación”.

Plan de implantación:

Para establecer la implantación del nuevo sistema de información se comunica al responsable de la FBCV establecer una reunión donde se expondrán las siguientes fases del plan definido:

- Plan de formación:

Esta fase como se ha detallado anteriormente, contempla la formación que se dará a los futuros usuarios de la aplicación.

- Implantación del sistema:

Esta fase contempla los pasos de implantación del sistema de información en la FBCV.

- Migración de datos:

Para la continuidad del plan es necesario que la FBCV realice la correcta migración de datos.

- Aceptación:

Una vez migrados los datos, empezarán las cargas con datos en un entorno de preproducción. En caso de que la funcionalidad sea correcta y concuerde con los requisitos del plan de sistemas, se procede al paso a producción de la aplicación.

2. Incorporación del sistema al entorno de operación

“En esta actividad se realizan todas las tareas necesarias para la incorporación del sistema al entorno de operación en el que se van a llevar a cabo las pruebas de implantación y aceptación del sistema.

Mientras que las pruebas unitarias, de integración y del sistema se pueden ejecutar en un entorno distinto de aquél en el que finalmente se implantará, las pruebas de implantación y aceptación del sistema deben ejecutarse en el entorno real de operación. El propósito es comprobar que el sistema satisface todos los requisitos especificados por el usuario en las mismas condiciones que cuando se inicie la producción.

Por tanto, como paso previo a la realización de dichas pruebas y de acuerdo al plan de implantación establecido, se verifica que los recursos necesarios están disponibles para que se pueda realizar, adecuadamente, la instalación de todos los componentes que integran el sistema, así como la creación y puesta a punto de las bases de datos en el entorno de operación.

Asimismo, se establecen los procedimientos de explotación y uso de las bases de datos de acuerdo a la normativa existente en dicho entorno.

Tareas propuestas para esta actividad:

- 2.1. Preparación de la instalación
- 2.2. Realización de la instalación

2.1. Preparación de la instalación

Objetivo:

“En esta tarea se verifica que está disponible la infraestructura necesaria para configurar el entorno. Dicha infraestructura debe cumplir los requisitos de implantación (instalación e infraestructura) y tener en cuenta los procedimientos de seguridad y control de acceso (mantenimiento de la integridad y confidencialidad de los datos, control de accesos al sistema, copias de seguridad y recuperación de datos, etc.), y operación y administración del sistema (estándares, recuperación y reanudación de trabajos, planificación de trabajos, etc.).

Además, si alguno de los sistemas de información implicados en la implantación lleva implícita una migración de datos habrá que tener en cuenta, también, las características del entorno y los procedimientos propios de la migración establecidos en el plan de migración y carga inicial de datos, obtenido en la actividad Diseño de la Migración y Carga Inicial de Datos (DSI 9). Una vez comprobada la idoneidad de los distintos elementos relacionados con la infraestructura, se realiza la instalación del software de base necesario para la incorporación posterior de los componentes asociados a los sistemas de información implicados en la implantación.

También se debe asegurar que el equipo responsable de la realización de las pruebas de implantación y aceptación del sistema ha recibido la formación necesaria.”

Preparación de la instalación:

Una vez se comprueba que los nuevos equipos están disponibles, se realiza in situ la gestión de usuarios para crear las cuentas del personal de administración en los PC's del departamento.

Además, durante la gestión de la instalación y carga inicial, se instalara de manera provisional el software de gestión remota VNC[®], con el fin de evitar desplazamientos innecesarios a la FBCV. Este software, al contrario que el escritorio remoto proporcionado por Microsoft[®] Windows 7, permite ver in situ los movimientos que se realizan con el PC, de manera que siempre estarán controlados los movimientos que realiza el usuario remoto. Es muy importante que la ejecución remota sea con este tipo de servicio para evitar problemas de violación o pérdida de información por parte del cliente que puedan ser reclamados al desarrollador.

No se realiza ninguna instalación previa de otro software, debido a que se debe probar si el propio paquete instala automáticamente el software requerido para su ejecución. No obstante, se comprueba que la versión de Excel previamente instalada sea compatible para la exportación de listados.

2.2. Realización de la instalación

Objetivo:

“Se realiza la instalación de todos los componentes del nuevo sistema, incluidos los procedimientos manuales y automáticos, de acuerdo al plan de implantación y a su ubicación física, establecida en el proceso Diseño del Sistema de Información. Se deben tener en cuenta los estándares y normativas por los que se rige la organización en los entornos de operación.

Asimismo, se prepara el entorno de datos identificando los sistemas de información que forman parte del sistema objeto de la implantación. Para cada uno de ellos:

- Se crean las bases de datos a partir del esquema físico elaborado en el proceso de construcción.
- Se establecen los procedimientos de explotación y uso de las bases de datos, es decir, la normativa necesaria para la utilización de las bases de datos, actualización, consulta, etc.
- Se revisan los procedimientos necesarios para realizar las copias de seguridad de los datos y de restauración de las copias indicando su frecuencia, así como los procedimientos de consolidación y sincronización de la información, estos últimos cuando proceda.
- Se preparan las autorizaciones de acceso a los datos para los distintos perfiles de usuarios. Una vez comprobada la correcta instalación del nuevo sistema, se activan los procedimientos de operación, de administración del sistema, de seguridad y de control de acceso. Incluyen el arranque y cierre del sistema según la frecuencia establecida, la planificación de trabajos, su recuperación y reanudación, las autorizaciones de acceso al sistema según los distintos perfiles de usuario, etc.

Asimismo, si es necesaria una migración de datos se activarán también los procedimientos asociados”.

Instalación final:

En esta fase se realiza la instalación del paquete BILLARGEST 1.0. En caso de realizarse correctamente, quedara instalado en el sistema:

- Software de gestión BILLARGEST 1.0
- Microsoft® Framework 3.5
- Base de datos en formato Microsoft® Access 2007

3. Carga de datos al entorno de operación

“Teniendo en cuenta que los sistemas de información que forman parte del sistema a implantar pueden mejorar, ampliar o sustituir a otros ya existentes en la organización, puede ser necesaria una carga inicial y/o una migración de datos cuyo alcance dependerá de las características y cobertura de cada sistema de información implicado. Por tanto, la necesidad de una migración de datos puede venir determinada desde el proceso Estudio de Viabilidad del Sistema (EVS), en la actividad Selección de la Solución (EVS 6). Allí se habrá establecido la estrategia a seguir en la sustitución, evaluando las opciones del enfoque de desarrollo e instalación más apropiados para llevarlo a cabo.

En cualquier caso, en la actividad Diseño de la Migración y Carga Inicial de Datos (DSI 9) se habrán definido y planificado los procesos y procedimientos necesarios para llevar a cabo la

migración, realizándose su codificación en la actividad Construcción de los Componentes y Procedimientos de Migración y Carga Inicial de Datos (CSI 8)”.

Tareas propuestas para esta actividad:

3.1. Migración y carga inicial de datos

3.1. Migración y carga inicial de datos

Objetivo:

“Se realiza la carga inicial de datos del nuevo sistema, y se comprueba que ha finalizado correctamente. A continuación, si procede, se hace la migración de datos, activando los procedimientos correspondientes, para efectuar la transformación de los datos de la estructura existente a la nueva. Se lleva a cabo la depuración de los datos que no sean consistentes, hasta comprobar su correcta finalización”.

Migración y carga de datos:

Una vez instalado el paquete, se comunica a la FBCV que realice la migración de datos en el sistema. Una vez concluida, se pasa a las pruebas de implantación.

4. Pruebas de implantación del sistema

“La finalidad de las pruebas de implantación es doble:

- Comprobar el funcionamiento correcto del mismo en el entorno de operación.
- Permitir que el usuario determine, desde el punto de vista de operación, la aceptación del sistema instalado en su entorno real, según el cumplimiento de los requisitos especificados.

Para ello, el responsable de implantación revisa el plan de pruebas de implantación y los criterios de aceptación del sistema, previamente elaborados. Las pruebas las realizan los técnicos de sistemas y de operación, que forman parte del grupo de usuarios técnicos que ha recibido la formación necesaria para llevarlas a cabo”.

Pruebas de implantación del sistema:

Las pruebas de implantación correrán a cargo de la FBCV, siendo supervisadas por el responsable del departamento, y no serán documentadas. En caso que alguna prueba no sea satisfactoria, se comunicará al responsable de desarrollo del programa. Este será el encargado de gestionar la incidencia y resolverla en el tiempo más corto posible.

Las incidencias serán documentadas en un documento estándar de pruebas.

Una vez resuelta la incidencia, será necesario recargar por completo el programa. Para ello se hará uso del software de gestión remota VNC. En caso de que sea necesario la modificación de la BBDD, la responsabilidad de migrar los datos entre BBDD será del desarrollador de la aplicación.

5. Pruebas de aceptación del sistema

“Las pruebas de aceptación tienen como fin validar que el sistema cumple los requisitos básicos de funcionamiento esperado y permitir que el usuario determine la aceptación del sistema. Por este motivo, estas pruebas son realizadas por el usuario final que, durante este periodo de tiempo, debe plantear todas las deficiencias o errores que encuentre antes de dar por aprobado el sistema definitivamente.

Los Directores de los Usuarios revisan los criterios de aceptación, especificados previamente en el plan de pruebas del sistema, y dirigen las pruebas de aceptación final que llevan a cabo los usuarios expertos. A su vez, éstos últimos deben elaborar un informe que los Directores de los Usuarios analizan y evalúan para determinar la aceptación o rechazo del sistema”.

Pruebas de aceptación:

Las pruebas de aceptación también serán realizadas por el departamento de administración y tampoco serán documentadas. En caso de que el departamento encuentre algún tipo de problema, se remitirá al desarrollador de la aplicación.

En caso de que existiera alguna ambigüedad entre los requisitos y la funcionalidad desarrollada, se establecerán reuniones entre las dos partes para contemplar si se debe aceptar la modificación como una incidencia (problema de la aplicación) o de ser el caso, como una petición nueva. En caso de ser una petición, será necesario valorar si la petición es de alcance, lo que supondría que en caso de desarrollo habría que modificar los requisitos y revalorar el producto para su cobro.

6. Preparación del mantenimiento del sistema

“El objetivo de esta actividad es permitir que el equipo que va a asumir el mantenimiento del sistema esté familiarizado con él antes de que el sistema pase a producción. Para conseguir este objetivo, se ha considerado al responsable de mantenimiento como parte integrante del equipo de implantación. Por lo tanto, se habrá tenido en cuenta su perfil al elaborar el esquema de formación correspondiente.

Una vez que el responsable de mantenimiento ha recibido la formación necesaria y adquirido una visión global del sistema que se va a implantar, se le entregan los productos que serán objeto del mantenimiento. De esta manera, obtiene de una forma gradual un conocimiento profundo del funcionamiento y facilidades que incorpora el sistema, que van a permitirle acometer los cambios solicitados por los usuarios con mayor facilidad y eficiencia. Se reduce, en consecuencia, el esfuerzo invertido en el mantenimiento.

Es importante resaltar que la existencia de una configuración del software permite reducir el esfuerzo requerido y mejora la calidad general del software a mantener, aunque no garantiza un mantenimiento libre de problemas. Una pobre configuración del software puede tener un impacto negativo sobre su facilidad de mantenimiento”.

Plan de mantenimiento:

Se establece que el desarrollador se haga cargo sin coste del servicio de mantenimiento durante el primer mes desde la implantación definitiva, siempre que la petición no sea de alcance, es decir, no modifique los requisitos iniciales y el coste operativo de asumir la petición sea excesivamente grande, en tal caso, se considerará el cobro aparte. Las peticiones aceptadas a partir del primer mes serán a cargo del cliente, pero en ningún caso se entregarán fuentes o se proporcionará formación para el mantenimiento de la aplicación.

7. Presentación y aprobación del sistema

“Una vez que se han efectuado las pruebas de implantación y de aceptación, y que se ha fijado el acuerdo de nivel de servicio, el Comité de Dirección debe formalizar la aprobación del sistema. Para esto, se lleva a cabo una presentación general del sistema al Comité de Dirección y se espera la confirmación de su aprobación”.

Aprobación del sistema:

Se reúne el responsable de desarrollo junto con el responsable del departamento de administración para evaluar la implantación del sistema de información. Siendo la evaluación positiva, se determina el paso a producción del sistema.

8. Paso a producción

“Esta actividad tiene como objetivo establecer el punto de inicio en que el sistema pasa a producción, se traspa la responsabilidad al equipo de mantenimiento y se empiezan a dar los servicios establecidos en el acuerdo de nivel de servicio, una vez que el Comité de Dirección ha aprobado el sistema.

Para ello es necesario que, después de haber realizado las pruebas de implantación y de aceptación del sistema, se disponga del entorno de producción perfectamente instalado en cuanto a hardware y software de base, componentes del nuevo sistema y procedimientos manuales y automáticos.

En función del entorno en el que se haya llevado a cabo las pruebas de implantación y aceptación del sistema, habrá que instalar los componentes del sistema total o parcialmente. También se tendrá en cuenta la necesidad de migrar todos los datos o una parte de ellos. Una vez que el sistema ya está en producción, se le notifica al responsable de mantenimiento, al responsable de operación y al Comité de Dirección”.

Paso a producción:

Una vez finalizadas las pruebas de implantación y aceptación, empieza a operar el sistema en un entorno real. El viejo software deja de ser utilizado y se eliminan las viejas máquinas del entorno.

Anexo II-7. Índice:

1.	Registro de la petición.....	3
1.1.	Registro de la petición.....	3
1.2.	Asignación de la petición.....	4
2.	Análisis de la petición.....	5
3.	Seguimiento y evaluación de los cambios hasta aceptación	5
3.1.	Registro de la petición.....	6
3.2.	Aprobación y cierre de la petición	6

• Mantenimiento del sistema de información (MSI)

“El objetivo de este proceso es la obtención de una nueva versión de un sistema de información desarrollado con MÉTRICA Versión 3 ó Versión 2, a partir de las peticiones de mantenimiento que los usuarios realizan con motivo de un problema detectado en el sistema, o por la necesidad de una mejora del mismo.

En este proceso se realiza el registro de las peticiones de mantenimiento recibidas, con el fin de llevar el control de las mismas y de proporcionar, si fuera necesario, datos estadísticos de peticiones recibidas o atendidas en un determinado periodo, sistemas que se han visto afectados por los cambios, en qué medida y el tiempo empleado en la resolución de dichos cambios. Es recomendable, por lo tanto, llevar un catálogo de peticiones de mantenimiento sobre los sistemas de información, en el que se registren una serie de datos que nos permitan disponer de la información antes mencionada.

En el momento en el que se registra la petición, se procede a diagnosticar de qué tipo de mantenimiento se trata. Atendiendo a los fines, podemos establecer los siguientes tipos de mantenimiento:

Correctivo: son aquellos cambios precisos para corregir errores del producto software.

Evolutivo: son las incorporaciones, modificaciones y eliminaciones necesarias en un producto software para cubrir la expansión o cambio en las necesidades del usuario.

Adaptativo: son las modificaciones que afectan a los entornos en los que el sistema opera, por ejemplo, cambios de configuración del hardware, software de base, gestores de base de datos, comunicaciones, etc.

Perfectivo: son las acciones llevadas a cabo para mejorar la calidad interna de los sistemas en cualquiera de sus aspectos: reestructuración del código, definición más clara del sistema y optimización del rendimiento y eficiencia.

Estos dos últimos tipos quedan fuera del ámbito de MÉTRICA Versión 3 ya que requieren actividades y perfiles distintos de los del proceso de desarrollo.

Una vez registrada la petición e identificado el tipo de mantenimiento y su origen, se determina de quién es la responsabilidad de atender la petición. En el supuesto de que la petición sea remitida, se registra en el catálogo de peticiones de mantenimiento y continúa el proceso. La petición puede ser denegada. En este caso, se notifica al usuario y acaba el proceso.

Posteriormente, según se trate de un mantenimiento correctivo o evolutivo, se verifica y reproduce el problema, o se estudia la viabilidad del cambio propuesto por el usuario. En ambos casos se estudia el alcance de la modificación. Hay que analizar las alternativas de solución identificando, según el tipo de mantenimiento de que se trate, cuál es la más adecuada. El plazo y urgencia de la solución a la petición se establece de acuerdo con el estudio anterior.

La definición de la solución incluye el estudio del impacto de la solución propuesta para la petición en los sistemas de información afectados. Mediante el análisis de dicho estudio, la

persona encargada del Proceso de Mantenimiento valora el esfuerzo y coste necesario para la implementación de la modificación.

Las tareas de los procesos de desarrollo que va a ser necesario realizar son determinadas en función de los componentes del sistema actual afectados por la modificación. Estas tareas pertenecen a actividades de los procesos Análisis, Diseño, Construcción e Implantación. Por último, y antes de la aceptación del usuario, es preciso establecer un plan de pruebas de regresión que asegure la integridad del sistema de información afectado. La mejor forma de mantener el coste de mantenimiento bajo control es una gestión del Proceso de Mantenimiento efectiva y comprometida. Por lo tanto, es necesario registrar de forma disciplinada los cambios realizados en los sistemas de información y en su documentación. Esto repercutirá directamente en la mayor calidad de los sistemas resultantes”.

1. Registro de la petición

“El objetivo de esta actividad es establecer un sistema estandarizado de registro de información para las peticiones de mantenimiento, con el fin de controlar y canalizar los cambios propuestos por un usuario o cliente, mejorando el flujo de trabajo de la organización y proporcionando una gestión efectiva del mantenimiento.

Es importante asignar responsabilidades para evitar la realización de cambios que beneficien a un usuario, pero que produzcan un impacto negativo sobre otros muchos. Por tanto, es necesario que todas las peticiones de mantenimiento sean presentadas de una forma estandarizada, que permita su clasificación y facilite la identificación del tipo de mantenimiento requerido.

Una vez que la petición ha sido registrada, que ha determinado el tipo de mantenimiento y los sistemas de información a los que inicialmente puede afectar, se comprueba su viabilidad, de acuerdo a las prestaciones de mantenimiento establecidas para dichos sistemas de información”.

Tareas propuestas para esta actividad:

- 1.1. Registro de la petición
- 1.2. Asignación de la petición

1.1. Registro de la petición

Objetivo:

“Esta tarea tiene como objetivo registrar las peticiones que los usuarios solicitan con motivo de la detección de un problema o por la necesidad de una mejora. Se crea un catálogo que constituye un medio para la comunicación entre el usuario o cliente y el responsable de mantenimiento.

Este catálogo servirá de base para abordar, en tareas posteriores, el análisis de la petición, realizar la modificación solicitada y proporcionar datos estadísticos sobre peticiones recibidas o atendidas.

La información que debe incluir dicho registro se determina de acuerdo a las normas o estándares existentes en la organización para la recepción de peticiones de mantenimiento. En el caso de un error se debe incluir una completa descripción de las circunstancias que llevaron al fallo, adjuntando datos de entrada, listados, o cualquier otro material de soporte que se considere oportuno. Para peticiones de mejora se debe remitir una especificación de los requisitos a contemplar.

En cualquier caso, será imprescindible recoger la identificación, origen y tipo de petición, asignarle una prioridad inicial e incorporar una descripción, lo más precisa posible, que facilite su posterior análisis.”

Registro de peticiones:

Las peticiones se recogen por tres canales, ya sea por email directo al responsable del desarrollo, por teléfono, o en las diversas reuniones que se establecen para las consultas.

Las peticiones se clasifican en una hoja Excel en la cual se anexa toda la documentación recibida para la petición. Además, es necesario establecer el tipo de petición, la fecha, si es bloqueante o no, el módulo afectado y la urgencia de resolución.

1.2. Asignación de la petición

Objetivo:

“En esta tarea se determina el tipo de mantenimiento requerido por la petición catalogada, teniendo en cuenta toda la información que se ha registrado en la tarea anterior. Hay que identificar también los sistemas de información inicialmente afectados por petición.

A continuación, se comprueba que el servicio de mantenimiento, definido en el plan de mantenimiento para el sistema de información, cubre el tipo de mantenimiento que requiere la petición. Sobre la base de estos criterios, se acepta o rechaza la petición y se notifica a quién corresponda.

Si la petición es aceptada, se determina de quién es la responsabilidad de atender la solicitud para proceder a su estudio posterior.”

Asignación de peticiones:

El primer paso para el análisis de la petición es comprobar si se trata de un problema del cliente o es parte de la aplicación. En caso de ser problema del cliente se le notifica y se da por cerrada la incidencia. En caso de ser problema de la aplicación desarrollada, se analizará posteriormente el tipo de petición que corresponde.

2. Análisis de la petición

“En esta actividad se lleva a cabo el diagnóstico y análisis del cambio para dar respuesta a las peticiones de mantenimiento que han sido aceptadas en la actividad anterior.

Se analiza el alcance de la petición en lo referente a los sistemas de información afectados, valorando hasta qué punto pueden ser modificados en función del ciclo de vida estimado para los mismos y determinando la necesidad de desviar la petición hacia el proceso Estudio de Viabilidad del Sistema (EVS) o Análisis del Sistema de Información (ASI), en función del impacto sobre los sistemas de información afectados.

El enfoque de este estudio varía según el tipo de mantenimiento, teniendo en cuenta que en el caso de un mantenimiento correctivo que implique un error crítico debe abordarse el cambio de forma inmediata sin profundizar en el origen del mismo. No obstante, una vez reanudado el servicio, es imprescindible analizar el problema y determinar cuál es la solución definitiva”.

Análisis de la petición:

Una vez analizada la petición, se establece el tipo de petición (incidencia de software, petición de modificación o mejora y petición de alcance). La prioridad de resolución será la incidencia (problema de la aplicación), posteriormente se resolverán peticiones (modificaciones o mejoras de corto alcance), y en última instancia peticiones de alcance.

Durante el primer mes desde la puesta en marcha de la aplicación, las peticiones correrán a cargo del desarrollador. No obstante, en caso de ser una petición de alcance, se evaluará su coste, y solo se desarrollará de manera gratuita para el cliente dentro del plazo establecido, si además no hay incidencias ni peticiones menos prioritarias que resolver.

Fuera de plazo, las peticiones deberán ser evaluadas para su cobro.

3. Seguimiento y evaluación de los cambios hasta aceptación

“Se realiza el seguimiento de los cambios que se están llevando a cabo en los procesos de desarrollo, de acuerdo a los puntos de control del ciclo de vida del cambio establecidos en el plan de acción. Durante este seguimiento, se comprueba que sólo se han modificado los elementos que se ven afectados por el cambio y que se han realizado las pruebas correspondientes, especialmente las pruebas de integración y del sistema. Del resultado obtenido se hace una evaluación del cambio para la posterior aprobación.

Una vez finalizado el cambio en desarrollo, se realizan las pruebas de regresión que especificadas en la actividad anterior, comprobando que ningún sistema no modificado, pero con posibilidades de verse afectado, ha variado su comportamiento habitual. Se informa si ha habido incidencias con el fin de que se resuelvan del modo más conveniente. Se evalúan las pruebas.

La aprobación de la petición se realiza al finalizar las pruebas de regresión, y después de comprobar que todo lo que ha sido modificado o puede verse afectado por el cambio, funciona correctamente.

Con el cierre de la petición se podrán incluir en el catálogo, si se considera oportuno, parte de la información obtenida durante el proceso de mantenimiento que pueda facilitar posteriores análisis”.

Tareas propuestas para esta actividad:

- 3.1. Seguimiento de los cambios
- 3.2. Aprobación y cierre de la petición

3.1. Registro de la petición

Objetivo:

“Se hace el seguimiento del plan de acción de acuerdo a los puntos de control establecidos en la actividad anterior.

Se realiza el seguimiento de los cambios necesarios en los componentes de cada sistema de información afectado, así como en los productos asociados, siguiendo las actividades correspondientes a los procesos de Análisis, Diseño, Construcción e Implantación identificados en la actividad anterior.

Asimismo, se lleva a cabo el control de la planificación establecida, que abarca los siguientes aspectos:

- Realizar la traza de los cambios que la petición ha provocado a lo largo de los procesos de desarrollo implicados.
- Verificar que se han realizado satisfactoriamente las pruebas unitarias, de integración y del sistema que se consideraron necesarias para los componentes a modificar.
- Comprobar que sólo se ha modificado lo establecido y, en caso contrario, justificar el motivo.
- Asegurar que se han actualizado los productos correspondientes.
- Llevar el control de los distintos desarrollos existentes en paralelo sobre un mismo componente, con el fin de coordinar las modificaciones incluidas en cada uno de ellos, y asegurar que en el paso a producción se implantan correctamente”.

Registro de la petición:

Una vez realizada la modificación, se vuelve a realizar un conjunto de test de prueba para la función de negocio modificada. Toda actualización se documenta en las hojas de pruebas ya realizadas añadiendo un nuevo campo en la hoja.

3.2. Aprobación y cierre de la petición

Objetivo:

“Se aprueba formalmente la finalización de la petición de mantenimiento de acuerdo a los resultados obtenidos en la tarea anterior. Se actualiza el catálogo de peticiones registrando el cierre de la petición tratada.

Asimismo, para llevar un control del coste y al mismo tiempo evaluar la facilidad de mantenimiento, es conveniente registrar datos cuantitativos relativos al tiempo empleado en el análisis de la petición, en el estudio del impacto, resolución del cambio, recursos empleados, etc.

El registro de este tipo de información proporciona una base cuantitativa sobre la que tomar decisiones relativas a la eficacia de las técnicas y procedimientos de mantenimiento.”

Cierre de la petición:

Una vez resuelta la incidencia y realizadas las pruebas pertinentes, se modifica la hoja de control de incidencias, identificando que se encuentra resuelta, y añadiendo datos tales como tiempo de resolución, coste operativo, recursos o cualquier observación pertinente que se considere oportuna.

En caso de ser una petición con costo asociado, se acumula en una hoja aparte cuyo importe será cobrado una vez finalizado el mes.

Anexo III. Índice:

1. Introducción	2
2. Instalación y desinstalación del paquete	2
3. Descripción del entorno de trabajo	5
4. Funciones del programa.....	7
5. Solución de errores	18

Manual de usuario

1. Introducción

Bienvenido al manual de usuario del software de gestión de ligas de billar BILLARGEST v1.0. El presente manual contiene toda la información técnica necesaria para el uso del Sistema de Gestión de Ligas de Billar BILLARGEST, que le permitirá la gestión integral de Temporadas, Socios y Clubes de su Federación.

2. Instalación y desinstalación del paquete

Antes de proceder a su instalación, compruebe que su sistema cumple los requisitos mínimos del programa:

-Sistemas operativos compatibles: Microsoft® Windows Server 2003; Windows Server 2008; Windows Vista; Windows XP

-Procesador: procesador Pentium a 400 MHz o equivalente (mínimo); procesador Pentium a 1 GHz o equivalente (recomendado)

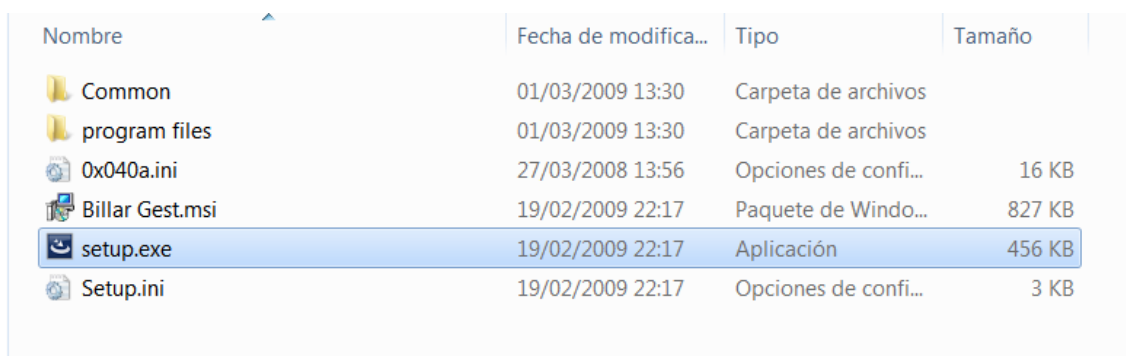
-RAM: 96 MB (mínimo); 256 MB (recomendado)

-Disco duro: se pueden necesitar hasta 150 MB de espacio disponible

-Unidad de CD o DVD: 2X

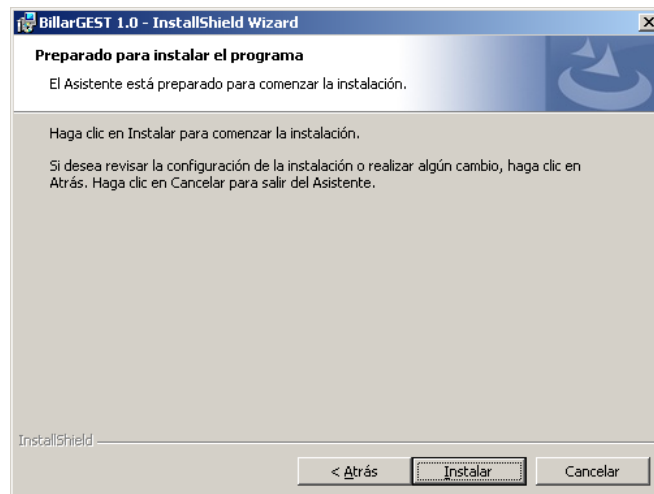
-Pantalla: 800 x 600, 256 colores (mínimo); color de alta densidad de 1024 x 768, 32 bits (recomendado)

Para comenzar la instalación haga **doble clic** en el icono **setup.exe** de su CDROM de instalación:

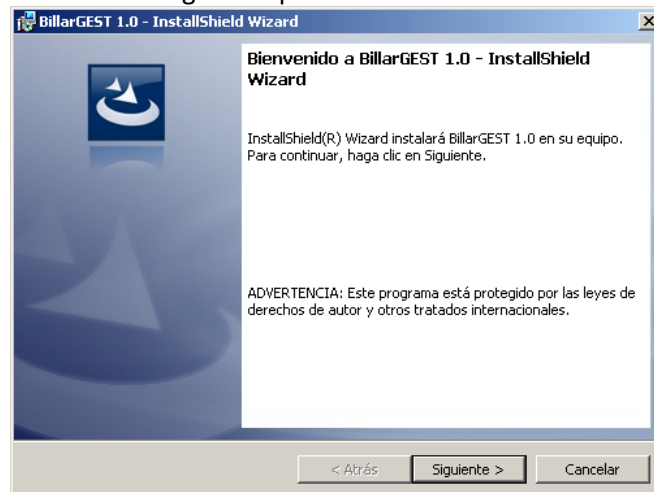


Nombre	Fecha de modifica...	Tipo	Tamaño
Common	01/03/2009 13:30	Carpeta de archivos	
program files	01/03/2009 13:30	Carpeta de archivos	
0x040a.ini	27/03/2008 13:56	Opciones de confi...	16 KB
Billar Gest.msi	19/02/2009 22:17	Paquete de Windo...	827 KB
setup.exe	19/02/2009 22:17	Aplicación	456 KB
Setup.ini	19/02/2009 22:17	Opciones de confi...	3 KB

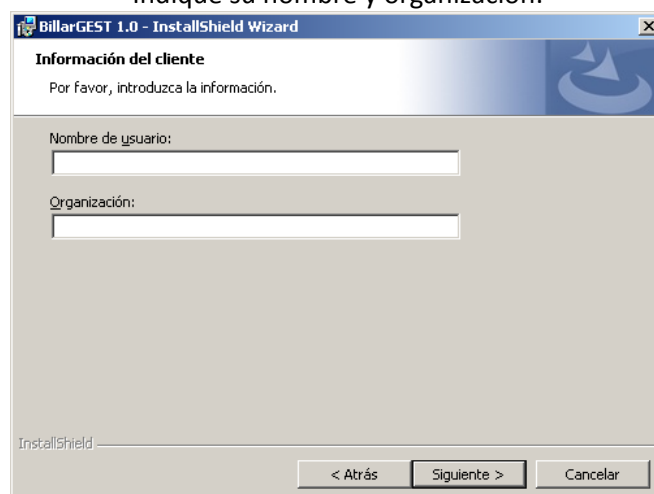
Una vez abierta la ventana de instalación, siga los pasos que se indican a continuación:



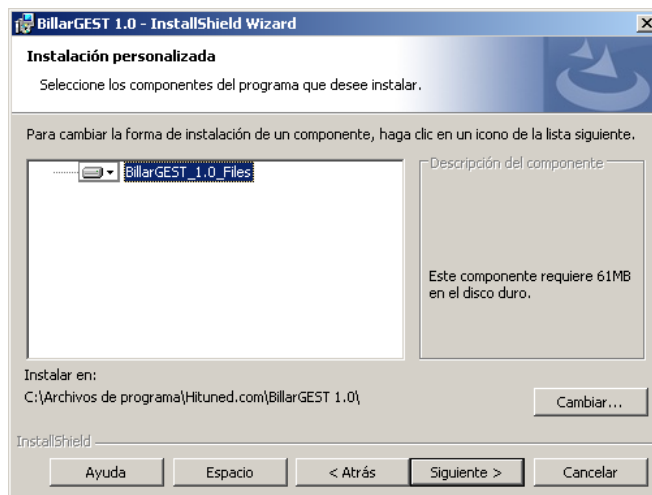
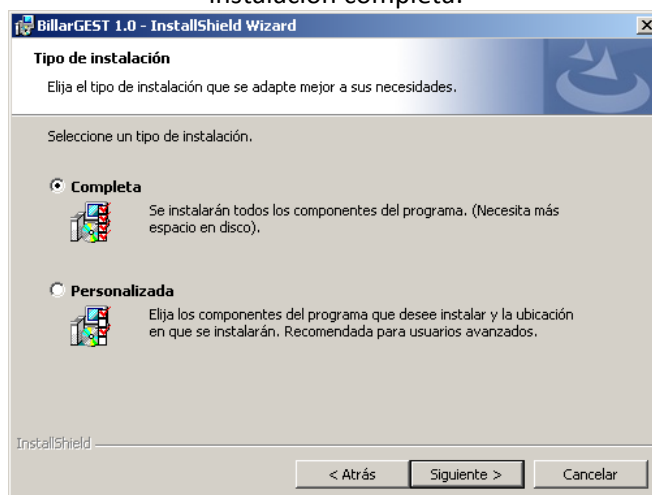
Pulse siguiente para iniciar la instalación:



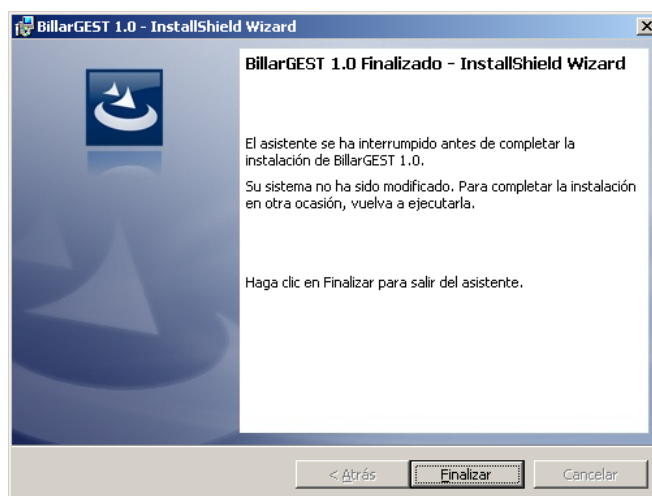
Indique su nombre y organización:



En caso que desee cambiar la ruta de instalación, pulse personalizada. En otro caso, seleccione instalación completa:



NOTA: En caso que decida cancelar la instalación, las carpetas temporadas y todos los archivos instalados serán borrados del sistema:



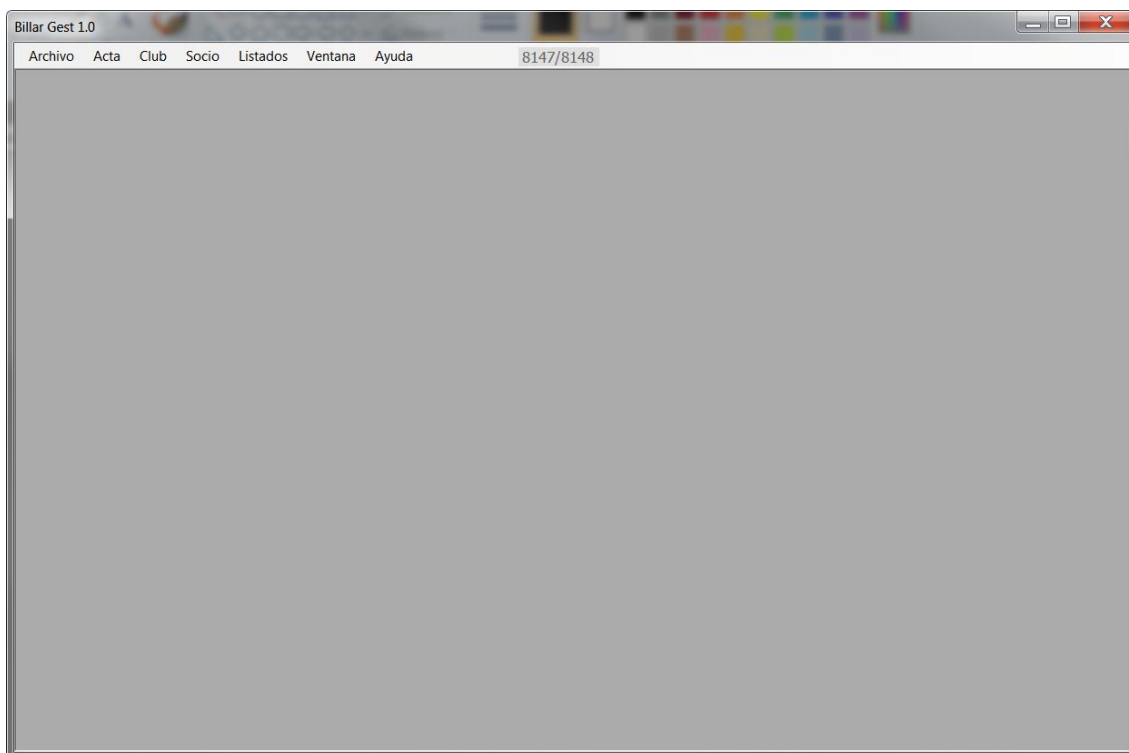
Una vez instalada, para ejecutar la aplicación, haga **doble clic** en el icono de su menú de inicio.

3. Descripción del entorno de trabajo

A continuación se muestran las principales características del entorno de BILLARGEST 1.0:

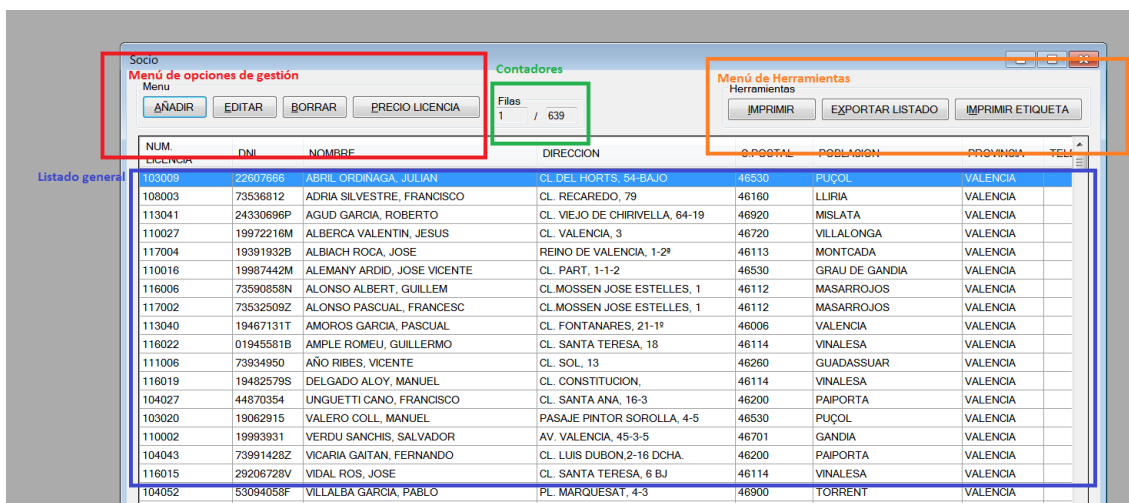
Pantalla principal: Desde aquí puede acceder a los diferentes menús de la aplicación.

Des de la pantalla principal se le informará en todo momento de la temporada que se encuentra actualmente. Usted puede acceder a las opciones del menú directamente haciendo **clic** sobre ellas.



Para salir de la aplicación pulse el botón del menú **Archivo** y a continuación **Salir**. También puede salir de la aplicación pulsando la cruceta superior derecha.

En cada menú de gestión usted encontrara las siguientes opciones predeterminadas:



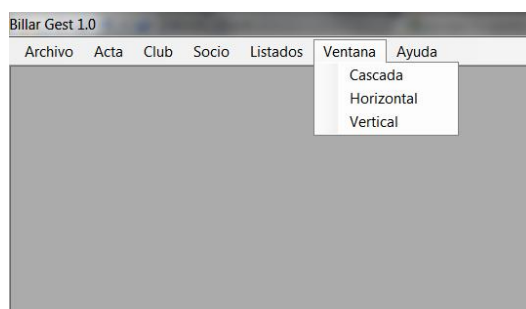
Menú de opciones de gestión: Contiene las opciones predeterminadas de gestión. Por orden, **Añadir, Editar, Borrar** y el resto de opciones específicas. Puede accederse directamente a la opción pulsando **ALT+Primer carácter**.

Contadores: El contador de listados permite contar un número de filas directamente del listado seleccionándolas con el ratón.

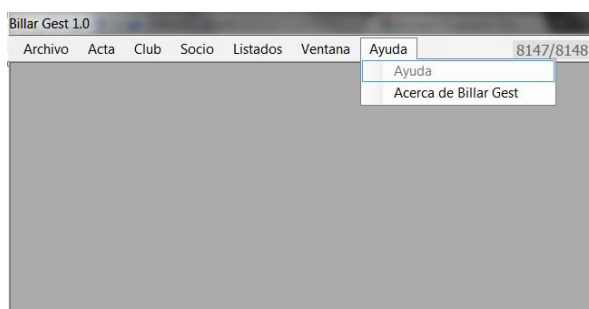
Menú de Herramientas: El menú de herramientas contiene las opciones de impresión y exportación de listados. Sus opciones por orden son: **Imprimir, Exportar listado** y el resto de opciones específicas. Puede accederse directamente a la opción pulsando **ALT+Primer carácter**.

Listado general: Corresponde al listado de la función seleccionada. Puede cambiar el orden de presentación de las columnas simplemente pulsando sobre ella.

Opciones de ayuda a la navegación: En todo momento usted puede maximizar, mover por el área de trabajo, o minimizar las ventanas de las opciones de menú del programa. En caso que necesite recolocar estas ventanas, se le aconseja que use la opción **Ventana** del menú de inicio. Seleccionando una de las tres opciones usted podrá recolocar las ventanas de la manera deseada.



En caso que necesite ayuda o soporte durante la ejecución del programa, acceda a la opción **Ayuda** del menú superior.

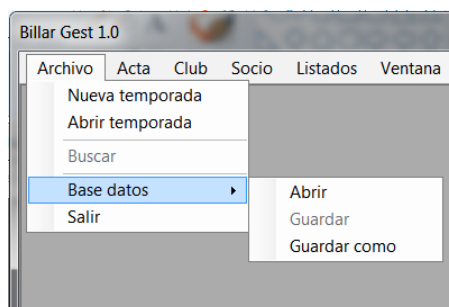


4. Funciones del programa

A continuación se explican de manera detallada las funciones que proporciona BILLARGEST para la gestión de su Federación de Billar.

Gestión de temporadas: Desde esta opción se gestiona la base de datos y las temporadas de BILLARGEST.

Una vez pulsado el botón **Archivo** en el menú superior, puede acceder al resto de opciones:

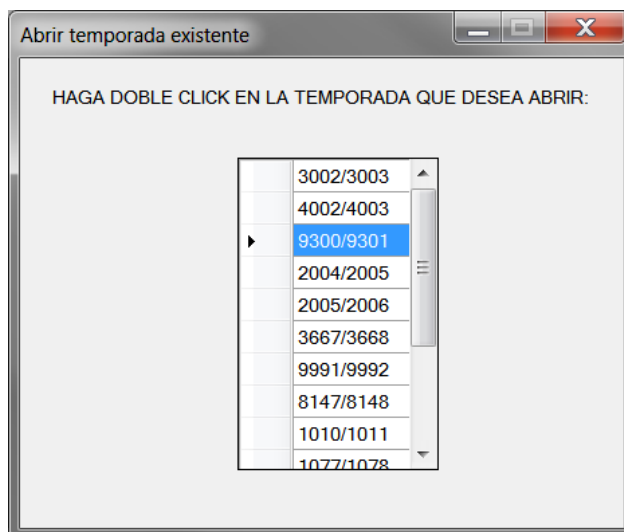


Nueva temporada: Permite crear una nueva temporada para albergar nuevas competiciones. Crear una nueva temporada conlleva vaciar los listados de estadísticas del programa y las actas visibles, pero no los clubes y socios introducidos.

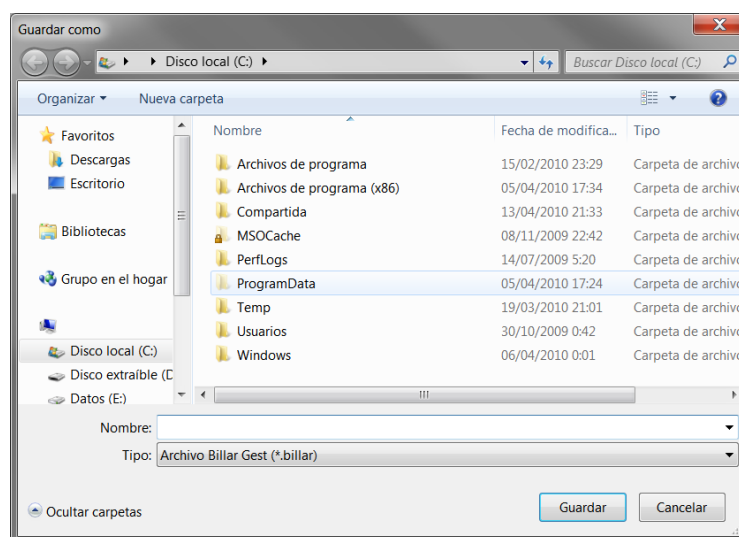
Para crear una nueva temporada, primero introduzca el año de la temporada. Posteriormente, para cada club, el grupo o grupos en los que participará. Pulse **Guardar** una vez seleccionados para pasar al siguiente club. Cuando estén todos los datos introducidos, pulse **Aceptar** para guardar los datos en la Base de Datos.

HONOR	PRIMERA	SEGUNDA	TERCERA
<input type="checkbox"/> A	<input type="checkbox"/> A	<input type="checkbox"/> A	<input type="checkbox"/> A
<input type="checkbox"/> B	<input type="checkbox"/> B	<input type="checkbox"/> B	<input type="checkbox"/> B
<input type="checkbox"/> C	<input type="checkbox"/> C	<input type="checkbox"/> C	<input type="checkbox"/> C
<input type="checkbox"/> D	<input type="checkbox"/> D	<input type="checkbox"/> D	<input type="checkbox"/> D
<input type="checkbox"/> E	<input type="checkbox"/> E	<input type="checkbox"/> E	<input type="checkbox"/> E
<input type="checkbox"/> F	<input type="checkbox"/> F	<input type="checkbox"/> F	<input type="checkbox"/> F

Abrir temporada: Podrá abrir una temporada ya creada, para ello seleccione el año de la temporada deseada y haga **doble clic** sobre ella.

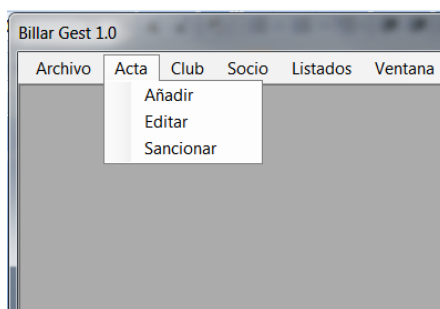


Base datos: Usted podrá gestionar las bases de datos desde esta opción. En caso de querer guardar una copia de seguridad integral de la base de datos, pulse **Guardar Como** e introduzca la ruta para guardar la Base de Datos.



Para Abrir una Base de Datos previamente guardada, pulse **Abrir** de la misma manera.

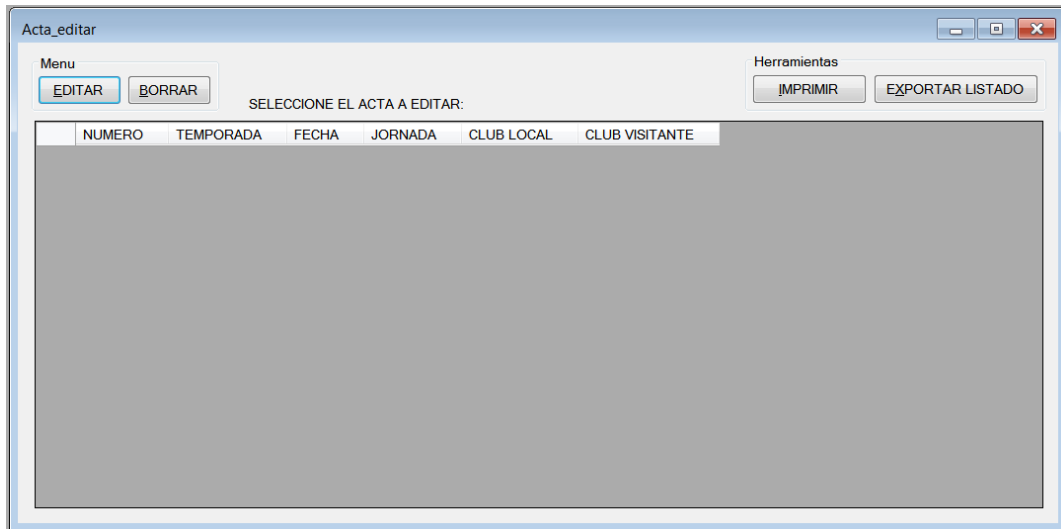
Acta: Desde esta opción se gestionan solo las actas de la temporada abierta actualmente.



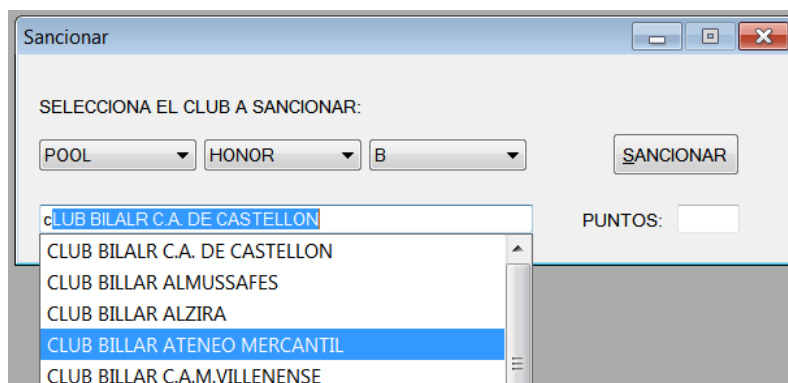
Añadir: Usted podrá añadir una nueva acta a la Base de Datos. Para ello pulse **Añadir** en el menú. Introduzca los datos del acta correspondiente y pulse **Aceptar**. Recuerde que puede saltar entre casillas pulsando la tecla **TAB** o la tecla **ENTER**.

Editar: Usted podrá editar un acta previamente introducida a la Base de Datos. Para ello pulse **Editar** en el menú. Una vez seleccione un acta pulse **Editar** para editar el acta.

Borrar: Usted podrá borrar un acta previamente introducida a la Base de Datos. Para ello pulse **Editar** en el menú. Seleccione un acta y pulse **Borrar** en el menú superior



Sancionar: Permite sancionar algún club por causas tales como alineación indebida o irregularidades. Seleccione el Club y los puntos a descontar de su clasificación global.



Club: Desde esta opción se gestionan los clubes adscritos a la Federación.

CODIGO	NOMBRE	CIF	DIRECCION	CP	POBLACION	PROVINCIA
103	CLUB BILLAR SINDICATO AGRICOLA PUÇOL		SAN JUAN, 93	46530	PUÇOL	VALENCIA
104	CLUB BILLAR PAIPORTA		SAN JORGE, 7 - BAJO	46200	PAIPORTA	VALENCIA
101	CLUB BILLAR VALENCIA		CONVENTO SANTA CLARA, 11	46002	VALENCIA	VALENCIA
102	CLUB BILLAR ATENEO MERCANTIL		PL. DEL AYUNTAMIENTO, 18	46002	VALENCIA	VALENCIA
105	CLUB BILLAR SUECA		RONDA PAIS VALENCIA,SIN	46410	SUECA	VALENCIA
106	CLUB BILLAR ALMUSSAFES		CL. POLIESPORTIU,SIN	46440	ALMUSSAFES	VALENCIA
107	CLUB BILLAR OLIVA		CL. MORERAS, 3	46780	OLIVA	VALENCIA
108	CLUB BILLAR LLIRIA		NTRA SRA. DE MONTIEL, 7	46160	LLIRIA	VALENCIA
109	CLUB BILLAR LA MUTUA DE XVVA		ANTONIO MACHADO, 6	46370	XVVA	VALENCIA
110	GANDIA BILLAR CLUB		CL. MIRAMAR, 13-BAJO	46701	GANDIA	VALENCIA
111	CLUB BILLAR L'ALCUDIA		PZ. TIRANT LO BLANCH,SIN	46250	L'ALCUDIA	VALENCIA
113	CLUB BILLAR MISLATA		POETA MIGUEL, HERNANDEZ, 11	46920	MISLATA	VALENCIA
114	CLUB BILLAR ALZIRA		ESCUELAS PIAS, 1	46600	ALZIRA	VALENCIA
115	PUÇOL CLUB DE BILLAR		SERRA LLARGA, 4	46530	PUÇOL	VALENCIA
116	CLUB BILLAR VINALESA		PZ. DEL CASTELL, 2	46114	VINALESA	VALENCIA

Añadir: Pulse la opción **Añadir** para añadir un nuevo club a la Base de Datos. Introduzca sus datos y pulse **Aceptar**. El botón **Rellenar Auto**. Permite rellenar automáticamente la dirección de correspondencia con los mismos datos de la dirección de sede. Recuerde que el código debe ser único, en caso de no serlo, el programa le avisará para que lo cambie.

Editar: Pulse la opción **Editar** para editar un club existente en la Base de Datos. Edite sus datos y pulse **Aceptar**. Recuerde que NO podrá editar el código numérico de club.

The 'Club_editar' window contains the following fields and sections:

- CODIGO:** 201
- NOMBRE:** ALICANTE BILLAR CLUB
- CIF:** G03305000
- DIRECCION SEDE:**
 - DIRECCION:** DEVESA, 4
 - CPOSTAL:** 03010
 - POBLACION:** ALICANTE
 - PROVINCIA:** ALICANTE
- DIRECCION DE CORRESPONDENCIA:** (with a 'RELLENAR AUTO.' button)
 - NOMBRE:** ALICANTE BILLAR CLUB
 - DIRECCION:** DEVESA, 4 BJ
 - CPOSTAL:** 03010
 - POBLACION:** ALICANTE
 - PROVINCIA:** ALICANTE
- WEB:** (empty field)
- TELEFONO:** 616697500
- FAX:** (empty field)
- EMAIL:** jose.gomez@ua.es
- Buttons:** ACEPTAR, CERRAR

Borrar: Pulse la opción **Borrar** para borrar un club existente en la Base de Datos. Recuerde que una vez borrado, no podrá volver a rescatar sus datos a menos que haya previamente guardado una copia de la Base de Datos.

Imprimir: En cualquier momento usted puede imprimir los datos del listado desde la opción **Imprimir** del menú de **Herramientas**. Seleccione las columnas que desea imprimir, las filas, si desea que se ajuste al ancho de la página de impresión y si desea introduzca un título de impresión. Pulse **Aceptar** para imprimir el listado.

The 'Opciones de Impresion' dialog box includes the following settings:

- Columnas a Imprimir:** A list of fields with checkboxes, all of which are checked: CODIGO, NOMBRE, CP, POBLACION, PROVINCIA, CP (2), POBLACION (2), PROVINCIA (2), TELEFONO, EMAIL, PAGINA WEB, and FAX.
- Filas a Imprimir:** Radio buttons for 'Todas' (selected) and 'Seleccionadas'.
- Ajustar ancho de página:** A checked checkbox.
- Título de Impresion:** An empty text area for entering a title.
- Buttons:** ACEPTAR, CANCELAR

Exportar Listado: En cualquier momento usted puede exportar los datos del listado a Microsoft © Excel desde la opción **Exportar** del menú de **Herramientas**.

Socio: Desde esta opción se gestionan los socios adscritos a los clubes dados de alta.

NUM LICENCIA	DNI	NOMBRE	DIRECCION	C.POSTAL	POBLACION	PROVINCIA	TELEFONO
109009	22607996	ADRI, EDOVAGGA, ALJUAN	CL DEL HORITO, 54-BAJO	46530	FUJOL	VALENCIA	
109003	73536812	ADRIA SILVESTRE, FRANCISCO	CL. RECAREDO, 79	46160	LLIBRA	VALENCIA	
113041	24330698P	AGUD GARCIA, ROBERTO	CL. VIEJO DE CHRVELLA, 84-19	46920	MSLATA	VALENCIA	
110027	18972216M	ALBERCA VALENTIN, JESUS	CL. VALENCIA, 3	46720	VILLALONGA	VALENCIA	
117004	19381932B	ALBACH ROCA, JOSE	REINO DE VALENCIA, 1-2P	46113	MONTCADA	VALENCIA	
110016	10967442M	ALEMANY ARDID, JOSE VICENTE	CL. PART. 1-1-2	46530	GRAU DE GANDIA	VALENCIA	
110006	73590558N	ALONSO ALBERT, GUILLEM	CL. MOSEN JOSE ESTELLES, 1	46112	MASARROJOS	VALENCIA	
117002	73032509Z	ALONSO PASCUAL, FRANCESC	CL. MOSEN JOSE ESTELLES, 1	46112	MASARROJOS	VALENCIA	
113040	19467131T	AMOROS GARCIA, PASCUAL	CL. FONTANARES, 21-1F	46006	VALENCIA	VALENCIA	
110022	01945581B	AMPLE ROMEU, GUILLEMO	CL. SANTA TERESA, 1B	46114	VINALESA	VALENCIA	
111006	73904950	AÑO RIBES, VICENTE	CL. SOL, 13	46260	GUADASSUAR	VALENCIA	
116019	19482579S	DELGADO ALOY, MANUEL	CL. CONSTITUCION,	46114	VINALESA	VALENCIA	
104027	44570254	UNGUETTI CANO, FRANCISCO	CL. SANTA ANA, 16-3	46200	PAIPORTA	VALENCIA	
105020	19592915	VALENO COLL, MANUEL	PASAJE PINTOR BOROLLA, 4-5	46030	FUJOL	VALENCIA	
110002	19999331	VERDU SANCHIS, SALVADOR	AV. VALENCIA, 45-3-5	46701	GANDIA	VALENCIA	
104043	73991426Z	VICARIA GAITAN, FERNANDO	CL. LUIS DUBON-2-16 DCHA.	46200	PAIPORTA	VALENCIA	
116015	29206728V	VIDAL ROS, JOSE	CL. SANTA TERESA, 6 BU	46114	VINALESA	VALENCIA	
104052	53094556F	VILLALBA GARCIA, PABLO	PL. MARQUESAT, 4-5	46900	TORRENT	VALENCIA	
104061	52748046F	VILLALBA SANZ, MANUEL	CL. SAN FRANCISCO, 62-2	46200	PAIPORTA	VALENCIA	
110010	19899479G	VILLORA MARTINEZ, JUAN RICARDO	CL. ENRIC VALOR, 20-5	46117	BETERA	VALENCIA	
109019	22517547C	ZACARES CARIS, LUIS MARIANO	AV. PERS Y VALERO, 122-6	46006	VALENCIA	VALENCIA	
109004	22589639Y	ZACARES FERRANDO, LUIS	AV. PERS Y VALERO, 122-6	46006	VALENCIA	VALENCIA	
110031		CAJEDON, FREDERIC				BELOGA	
208017	21630401	ABAD FERRER, RICARDO	CL. OLIVER, 75	03802	ALCOY	ALICANTE	
216013	74200118W	ABELLAN GOMEZ, ANTONIO	CL. VICENTE ALEXANDRE, 24-1F	03400	VILLENA	ALICANTE	
214024	X7205465W	AGUILAR ONCE, JOHNY ROMAN	CL. MADRID, 32-5 2DA.	03130	SANTA POLA	ALICANTE	
214025	X7205452N	AGUILAR ONCE, LICTEH ADRIMAN	CL. MADRID, 32-5 2DA.	03130	CREVILLENTE	ALICANTE	

Añadir: Pulse la opción **Añadir** para añadir un nuevo socio a la Base de Datos. Introduzca sus datos y pulse **Aceptar**. Recuerde que el número de licencia debe ser único, en caso de no serlo, el programa le avisará para que lo cambie.

Socio_editar

NUMERO LICENCIA:

CLUB:

ORDEN FUERZA:

LICENCIA EN VIGOR:

DATOS PERSONALES:

NOMBRE: APELLIDOS:

DNI: FECHA NACIMIENTO:

DIRECCION: COD POSTAL:

PROVINCIA: POBLACION:

TELEFONO: EMAIL:

ACEPTAR CANCELAR

Editar: Pulse la opción **Editar** para editar un socio existente en la Base de Datos. Edite sus datos y pulse **Aceptar**. Recuerde que NO podrá editar el número de licencia de socio ni el club al que pertenece.

The screenshot shows a window titled 'Socio_editar' with the following fields:

- NUMERO LICENCIA: 218008
- CLUB: CLUB BILLAR CASINO EL CAMP
- ORDEN FUERZA: 4
- LICENCIA EN VIGOR: Sí
- DATOS PERSONALES:
 - NOMBRE: MATEO
 - APELLIDOS: MAS VERDU
 - DNI: 21417703B
 - FECHA NACIMIENTO: 15/04/1959 SENIOR
 - DIRECCION: ADVA. GENERALITAT, 8-1ªA
 - COD. POSTAL: 03560
 - PROVINCIA: ALICANTE
 - POBLACION: EL CAMPELLO
 - TELEFONO: [empty]
 - EMAIL: [empty]

Buttons: ACEPTAR, CANCELAR

Borrar: Pulse la opción **Borrar** para borrar un socio existente en la Base de Datos. Recuerde que una vez borrado, no podrá volver a rescatar sus datos a menos que haya previamente guardado una copia de la Base de Datos.

Precio Licencia: Pulse la opción **Precio Licencia** para establecer el precio de la licencia para socio Junior o socio Sénior. Pulse **Aceptar** una vez introducidos los datos.

The screenshot shows a window titled 'Precio' with the following fields:

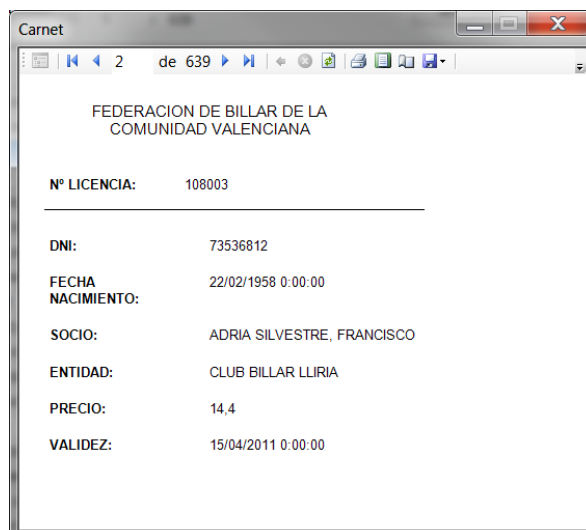
- JUNIOR:
 - ANTERIOR: 12,4
 - ACTUAL: [empty]
- SENIOR:
 - ANTERIOR: 14,4
 - ACTUAL: [empty]

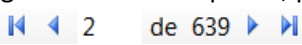




Buttons: ACEPTAR, CANCELAR

Imprimir: En cualquier momento usted puede imprimir los datos del listado desde la opción **Imprimir** del menú de **Herramientas**. Seleccione las columnas que desea imprimir, las filas, si desea que se ajuste al ancho de la página de impresión y si desea introduzca un título de impresión. Pulse **Aceptar** para imprimir el listado.

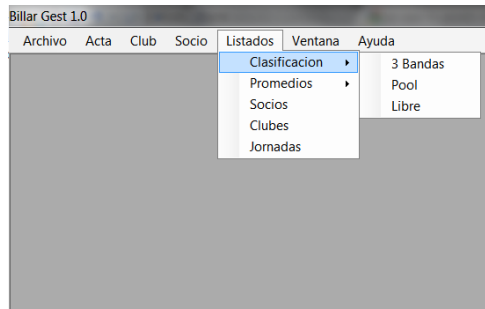
Exportar Listado: En cualquier momento usted puede exportar los datos del listado a Microsoft[®] Excel desde la opción **Exportar** del menú de **Herramientas**.

Imprimir Etiqueta: En cualquier momento usted puede imprimir un carnet de socio desde la opción **Imprimir Etiqueta** del menú de **Herramientas**.



Una vez generadas las etiquetas, puede navegar entre las etiquetas desde el menú superior  2 de 639 . Pulse el icono de **Impresión**  para imprimir las. También puede **Copiarlas**  o **Guardarlas**  en un directorio.

Listados: Desde esta opción se visualizan e imprimen los listados del programa.



Clasificación: Pulse la opción **Clasificación** para visualizar en pantalla un listado de la clasificación provisional por puntos de una liga. Usted puede ordenar el listado pulsando en la cabecera de la columna. Para seleccionar la liga a filtrar haga **click** en la liga deseada desde el menú **Filtrar**

The screenshot shows the 'Clasificación 3 bandas' window. It has a 'Filtrar' dropdown menu set to 'HONOR A'. Below the menu is a table with the following data:

	PJ	PG	PE	PP	CAR	ENT	PROM	PPA	PMA	SANC	PUNT
CLUB BILLAR ATENEO MERCANTIL	2	1	1	0	60	11	5,4545454545454541	9	3	0	3
CLUB BILLAR PAIPORTA	2	0	1	1	41	11	3,7272727272727271	7	1	0	1
CLUB BILLAR SUECA	0	0	0	0	0	0	0	0	0	0	0
CLUB BILLAR VALENCIA	0	0	0	0	0	0	0	0	0	0	0
CLUB BILLAR SINDICATO AGRICOLA PUÇOL	0	0	0	0	0	0	0	0	0	0	0

Promedios: Pulse la opción **Promedios** para visualizar en pantalla un listado de la clasificación provisional por mayor promedio de una liga. Usted puede ordenar el listado pulsando en la cabecera de la columna. Para seleccionar la liga a filtrar haga **click** en la liga deseada desde el menú **Filtrar**

The screenshot shows the 'Promedio de Clubes' window. It has a 'Filtrar' dropdown menu set to 'TODAS'. Below the menu is a table with the following header:

EQUIPO	PJ	PG	PE	PP	CAR	ENT	PROM	PPA	PMA	SANC	PUNT
--------	----	----	----	----	-----	-----	------	-----	-----	------	------

Socios / Clubes / Jornadas: Pulse la opción **Socios, Clubes o Jornadas** para visualizar en pantalla un listado de los socios guardados en la Base de Datos, clubes o actas por jornadas. Usted puede ordenar el listado pulsando en la cabecera de la columna. Para seleccionar el Club / Socio / Jornada a filtrar haga **click** en la liga deseada desde el menú **Filtrar**

NOMBRE	DNI	DIRECCION	POBLACION	CC
ORDIÑAGA, JULIAN	22607666	CL.DEL HORTS, 54-BAJO	PUÇOL	465
SILVESTRE, FRANCISCO	73536812	CL. RECAREDO, 79	LLIRIA	461
GARCIA, ROBERTO	24330696P	CL. VIEJO DE CHIRIVELLA, 64-19	MISLATA	465
RCA VALENTIN, JESUS	19972216M	CL. VALENCIA, 3	VILLALONGA	467
CH ROCA, JOSE	19391932B	REINO DE VALENCIA, 1-2º	MONTCADA	461
ANY ARDID, JOSE VICENTE	19987442M	CL. PART, 1-1-2	GRAU DE GANDIA	465
SO ALBERT, GUILLEM	73590858N	CL.MOSSEN JOSE ESTELLES, 1	MASARROJOS	461
SO PASCUAL, FRANCESC	73532509Z	CL.MOSSEN JOSE ESTELLES, 1	MASARROJOS	461
OS GARCIA, PASCUAL	19467131T	CL. FONTANARES, 21-1º	VALENCIA	46C
E ROMEU, GUILLERMO	01945581B	CL. SANTA TERESA, 18	VINALESA	461
RIBES, VICENTE	73934950	CL. SOL, 13	GUADASSUAR	462
ADO ALOY, MANUEL	19482579S	CL. CONSTITUCION,	VINALESA	461
ETTI CANO, FRANCISCO	44870354	CL. SANTA ANA, 16-3	PAIPORTA	462
RO COLL, MANUEL	19062915	PASAJE PINTOR SOROLLA, 4-5	PUÇOL	465
J SANCHIS, SALVADOR	19993931	AV. VALENCIA, 45-3-5	GANDIA	467
IA GAITAN, FERNANDO	73991428Z	CL. LUIS DUBON 2-16 DCHA.	PAIPORTA	462
ROS, JOSE	29206728V	CL. SANTA TERESA, 6 BJ	VINALESA	461

5. Solución de errores

En caso de que el programa devuelva algún tipo de error, se ruega comunique el código del error mostrado en la ventana al responsable del proyecto para resolver el problema en el periodo más corto de tiempo posible.

Puede remitir el error por correo electrónico a: carsolo2@fiv.upv.es .

Anexo IV. Índice:

1.	Medición por puntos de función ajustados	2
1.1.	Identificación de la frontera de cálculo.....	2
1.2.	Identificación de puntos de función de datos.....	3
1.2.1.	Ficheros Lógicos Internos	3
1.2.2.	Ficheros de Interfaz Externos.....	4
1.3.	Identificación de puntos de función de transacciones	5
1.3.1.	Entradas del exterior	5
1.3.2.	Salidas al exterior	6
1.3.3.	Consultas al exterior.....	6
1.4.	Determinación del factor de ajuste.....	7
1.5.	Estimación del esfuerzo requerido	9
2.	Presupuesto Final	10

Presupuesto

1. Medición por puntos de función ajustados

La técnica de puntos de función fue introducida por Alan Albrecht en 1979¹ y su propósito es medir el software cualificando la funcionalidad que proporciona externamente, basándose en el diseño lógico del sistema.

Los objetivos de los puntos de función son:

- Medir lo que el usuario pide y lo que el usuario recibe.
- Medir independientemente la tecnología utilizada en la implantación del sistema.
- Proporcionar una métrica de tamaño que de soporte al análisis de la calidad y la productividad.
- Proporcionar un medio para la estimación del software y proporcionar un factor de normalización para la comparación de distintos software. El análisis de los puntos de función se desarrolla considerando cinco parámetros básicos externos del sistema:

1. Entrada (EI, del inglés External Input).
2. Salida (EO, del inglés External Output).
3. Consultas (EQ, del inglés External Query).
4. Ficheros Lógicos Internos (ILF, del inglés Internal Logic File).
5. Ficheros Lógicos Externos (EIF, del inglés External Interface File).

Con estos parámetros, se determinan los puntos de función sin ajustar (PFsA). A este valor, se le aplica un factor de ajuste obtenido en base a unas valoraciones subjetivas sobre la aplicación y su entorno; es decir, las características generales del sistema.

1.1. Identificación de la frontera de cálculo

La frontera de cálculo marca el límite entre el proyecto o aplicación que se mide y las aplicaciones externas o el dominio del usuario. Su propósito es aclarar el marco del producto a medir, la propiedad de los datos, necesario para contar los puntos de función y la relación entre los procesos, necesario para el cálculo de los puntos de función.

Las reglas para el establecimiento de la frontera de cálculo son:

- La frontera se determina basándose en el *punto de vista del usuario*. Se centra en que es lo que el usuario puede describir y entender.

¹ Allan J. Albrecht and John E. Gaffney, "Software Function, Lines of Code, and Development Effort Prediction: A Software Science Validation", IEEE Transactions on Software Engineering, vol SE-9, No 6, November 1983.

- La frontera entre aplicaciones relacionadas se basa en las funciones de negocio independientes, como las ve el usuario, no basándose en conceptos técnicos
- En los proyectos de ampliación, la frontera inicial ha de estar de acuerdo a las fronteras actuales de la aplicación o aplicaciones a modificar.

El alcance del sistema por tanto deberá cubrir toda la operatoria de gestión de clubes, de las tres provincias de la Comunidad Valenciana, la gestión de los Socios de los respectivos clubes, y el tratamiento de las ligas que se juegan anualmente. El límite del alcance radicará en las entidades externas de club y socio.

1.2. Identificación de puntos de función de datos

Los puntos de función de datos se dividen en dos tipos:

1.2.1. Ficheros Lógicos Internos

Un fichero lógico interno (FLI) es un grupo de datos o información de control lógicamente relacionados e identificada por el usuario y mantenida dentro de la frontera de la aplicación. Se trata por tanto de una agrupación de datos lógica o identificable desde el punto de vista del usuario y que satisface un requerimiento específico del usuario. La agrupación de datos es mantenida por procesos de la aplicación en estudio, mediante un proceso elemental de la aplicación y no ha sido contada como un fichero de interfaz externo.

Una vez identificados los FLI hay que averiguar la complejidad de los mismos. Para ello, se usa la siguiente tabla:

Contribución Ficheros	Número de Campos o Atributos		
	1-19 TED	20-50 TED	51 + TED
1 TER	BAJA	BAJA	MEDIA
2 a 5 TER	BAJA	MEDIA	ALTA
6 TER.	MEDIA	ALTA	ALTA

Figura 1. Tabla de complejidad FLI

Siendo:

TED (tipo elemental de datos): es un único campo, no recursivo y reconocible por el usuario en un FLI o FIE.

TER (tipo elemental de registro): subgrupo de datos elementales reconocible por los usuarios en un FLI o FIE. Existen dos tipos de subgrupos, opcional, cuando los usuarios los pueden usar o no, y obligatorios, cuando los usuarios han de utilizar al menos un subgrupo.

Ficheros Lógicos Internos			
Nombre fichero	TERregistro	TEDato	Complejidad
Temporada	2	5	B
Clasificacion_club	0	11	B
Clasificacion_jugador	0	12	B
Socio	1	12	B
Club	0	10	B
Acta	4	14	B
Temporada_Club	0	2	B
Socio_Acta	1	10	B
Total Fichero Lógicos Internos		Baja	8
		Media	0
		Alta	0

Figura 2. Ficheros Lógicos Internos

1.2.2. Ficheros de Interfaz Externos

Un fichero de interfaz externo (FIE) es un grupo de datos o información de control lógicamente relacionados identificados por el usuario, referenciada por la aplicación, pero mantenida dentro de los límites de otra aplicación. Esto quiere decir que un FIE contado en una aplicación es un FLI en otra aplicación.

Por tanto, se trata de una agrupación de datos lógica o identificable desde el punto de vista del usuario y satisface un requerimiento específico del usuario; la agrupación de datos es referenciada, y externa, a la aplicación en estudio. Además, la agrupación **no es mantenida** mediante la aplicación en estudio, ha sido contada como un fichero lógico Interno en otra aplicación y no ha sido contada como un fichero lógico Interno de la aplicación en estudio.

Ficheros Externos de Interfaz			
Nombre fichero	TERregistro	TEDato	Complejidad
ClubFederacion	4	25	M
ClubMinisterio	8	30	A
Total Fichero Externos de Interfaz		Baja	0
		Media	1
		Alta	1

Figura 3. Ficheros Externos de Interfaz

1.3. Identificación de puntos de función de transacciones

Los puntos de función transaccionales cuentan la funcionalidad que obtiene el usuario de la aplicación. Contienen:

1.3.1. Entradas del exterior

Una entrada del exterior consiste en la recepción de información desde el exterior de la frontera de la aplicación. La información ha sido especificada por el usuario para cumplir la especificación de los procesos de negocio. Además, la información procesada puede o no mantener datos en algún FLI.

Para el proceso identificado es aplicable:

- La lógica de proceso es única y distinta de otras EE
- Los elementos de datos identificados son distintos de otras EE.

Para averiguar la complejidad se usará la siguiente tabla:

	1-4 TED	5-15TED	16 + TED
0-1 RAF	BAJA	BAJA	MEDIA
2 RAF	BAJA	MEDIA	ALTA
3 + RAF	MEDIA	ALTA	ALTA

Figura 4. Tabla de complejidad FIE

Siendo:

RAF (referencia a fichero): Un fichero Lógico Interno (ILF) leído o mantenido por una transacción o un fichero de Interfaz Externo (FIE) leído por una transacción.

Entradas Externas				
Evento	Nombre del proceso	RefAFich	TEDato	Complejidad
1,1	Datos nuevo socio	2	12	M
1,2	Datos editar socio	3	12	A
1,3	Datos nueva temporada	4	15	A
1,4	Datos nuevo club	4	17	A
1,5	Datos editar club	2	14	M
1,6	Datos nueva acta	5	23	A
1,7	Datos editar acta	5	26	A
1,8	Datos sancionar	3	25	A
Total Entradas Externas			Baja	0
			Media	2
			Alta	6

Figura 5. Entradas externas

1.3.2. Salidas al exterior

Es un proceso elemental que genera datos o información de control y la envía al exterior de la frontera de la aplicación.

Para averiguar la complejidad se usará la siguiente tabla:

	1-5 TED	6-19TED	20 + TED
0-1 RAF	BAJA	BAJA	MEDIA
2-3 RAF	BAJA	MEDIA	ALTA
4 + RAF	MEDIA	ALTA	ALTA

Figura 6. Tabla de complejidad SE

Salidas Externas				
Evento	Nombre del proceso	RefAFich	TEDato	Complejidad
2,1	Promedio de clubes	3	15	M
2,2	Promedio de jugadores	3	18	M
2,3	Visualización de jornadas	4	20	A
2,4	Consulta clasificación	2	16	M
2,5	Listado temporadas	1	11	B
2,6	Listado actas	1	10	B
2,7	Listado socios	1	5	B
2,8	Listado clubes	1	11	B
Total Salidas Externas			Baja	4
			Media	3
			Alta	1

Figura 7. Salidas externas

1.3.3. Consultas al exterior

Es un proceso elemental compuesto de una combinación entrada-salida que produce una recuperación de datos. La parte de la salida no contiene datos calculados.

Ningún FLI es mantenido durante el proceso.

Sera una consulta al exterior cuando:

- Una petición atraviesa la frontera del sistema.
- Una salida traspasa la frontera hacia el exterior de la aplicación.
- Se recuperan datos.
- Los datos recuperados no incluyen datos derivados.
- La petición de entrada y los resultados de la salida conforman juntas un proceso que es la menor unidad de proceso que tiene sentido para el usuario final.

Consultas Externas		Entrada		
Evento	Nombre del proceso	RefAFich	TEData	Complejidad Entrada
3,1	Elegir categoría	1	3	B
3,2	Introducir división	1	4	B
3,3	Introducir jornada	1	24	M
3,4	Introducir grupo	1	6	B
3,5	Introducir club	1	50	M
3,6	Introducir jugador	1	50	M

Figura 8. Consultas externas 1

Consultas Externas		Salida		
Evento	Nombre del proceso	RefAFich	TEData	Complejidad Salida
3,1	Elegir categoría	1	1	B
3,2	Introducir división	1	1	B
3,3	Introducir jornada	1	1	B
3,4	Introducir grupo	1	1	B
3,5	Introducir club	1	1	B
3,6	Introducir jugador	1	1	B

Figura 9. Consultas externas 2

Consultas Externas		
Evento	Nombre del proceso	Complejidad Entradas+Salidas.
3,1	Elegir categoría	B
3,2	Introducir división	B
3,3	Introducir jornada	M
3,4	Introducir grupo	B
3,5	Introducir club	M
3,6	Introducir jugador	M
Total Consultas Externas		3
		3
		0

Figura 10. Consultas externas 3

1.4. Determinación del factor de ajuste

Es un proceso elemental compuesto de una combinación entrada-salida que produce una recuperación de datos. La parte de la salida no contiene datos calculados. Sus características son:

- Se basa en 14 características generales del sistema.
- Completan la visión externa de la aplicación.
- No están recogidos en la funcionalidad de la aplicación.
- Toman un valor entre 0 y 5.

- 0 No presente, o sin influencia.
- 1 Influencia Incidental.
- 2 Influencia Moderada.
- 3 Influencia Media.
- 4 Influencia Significativa.
- 5 Influencia Generalizada y Fuerte.

Una vez aplicado el factor de complejidad se tiene que:

#	Factor de Complejidad	Valor (0..5)
1	Comunicación de Datos.	3
2	Proceso Distribuido.	1
3	Objetivos de Rendimiento	0
4	Configuración de Explotación compartida	1
5	Tasa de Transacciones	0
6	Entrada de Datos EN-LÍNEA	2
7	Eficiencia con el Usuario Final	1
8	Actualizaciones EN-LÍNEA	1
9	Lógica del Proceso Interno Compleja	0
10	Reusabilidad del Código	1
11	Contempla la Conversión e Instalación	1
12	Facilidad de Operación	0
13	Instalaciones Múltiples	1
14	Facilidad de Cambios	0
Factor de Complejidad Total (FCT)		12

Figura 11. Factores de complejidad

Tipo Elemento	Dificultad	Peso	Cantidad	Total	Total Elemento
Ficheros Internos	Simple	7	8	56	
	Media	10	0	0	
	Compleja	15	0	0	
	Total Ficheros Internos				
Ficheros de Interfaz	Simple	5	0	0	
	Media	7	1	7	
	Compleja	10	1	10	
	Total Ficheros de Interfaz				
Entradas	Simple	3	0	0	
	Media	4	2	8	
	Compleja	6	6	36	
	Total Puntos de Función Entradas				
Salidas	Simple	4	4	16	
	Media	5	3	15	
	Compleja	7	1	7	
	Total Salidas				
Consultas:	Simple	3	3	9	
	Media	4	3	12	
	Compleja	6	0	0	
	Total Consultas				
Total Puntos de Función Sin Ajustar (PFSA)					176

Figura 12. Puntos de función sin ajustar

	VFA		
Puntos de función Ajustados	0,77		135,52
Estimación	135,52	C#	474,32 horas
Puntos de Función Sin Ajustar (PFSA):		176	
Puntos de Función Ajustados:		135,72	

A la vista de dichos puntos de función se puede estimar el esfuerzo de dicho proyecto en torno a 474,32 horas para desarrollarlo sobre un lenguaje de alto nivel y quinta generación como es C# 3.5 (aproximadamente 3,5 horas² por punto de función ajustado).

1.5. Estimación del esfuerzo requerido

Una vez calculados los puntos de función es necesario aplicar esta medida a la valoración real de trabajo que requiere el proyecto.

Para ello, se dividen los puntos de función sin ajustar entre las horas estimadas, y se obtiene la productividad media:

Concepto	Horas	Porcentaje Final
Planificación del Sistema	48,85	10,30%
Estudio de Viabilidad	27,79	5,86%
Análisis del Sistema	64,55	13,61%
Diseño del Sistema	55,68	11,74%
Construcción del Sistema	124,74	26,30%
Implantación y pruebas	74,27	15,66%
Documentación	49,94	10,53%
Formación	28,45	6,00%

Figura 13. Productividad Media PF/HORAS

PRODUCTIVIDAD MEDIA: 176/356,3 = 0,49 Puntos de función por hora.

² Según los datos del estudio del Software Productivity Research (2009): <http://www.gotdotnet.com/team/compare/petshop.aspx>

2. Presupuesto Final

El objetivo del presupuesto es obtener una medida del coste económico del presente proyecto.

Se analizan primero los costes de los distintos perfiles del equipo de trabajo:

Actividad	Coste € / hora
Planificación del Sistema	20 €
Estudio de Viabilidad	20 €
Análisis del Sistema	20 €
Diseño del Sistema	18 €
Construcción del Sistema	18 €
Implantación y pruebas	18 €
Documentación	18 €
Formación	15 €

Figura 14. Coste/hora según actividad

El coste final será:

Concepto	Horas	Porcentaje Final	Precio unitario / hora (IVA excluido)
Planificación del Sistema	48,85	10,30%	977 €
Estudio de Viabilidad	27,79	5,86%	555,8 €
Análisis del Sistema	64,55	13,61%	1.291 €
Diseño del Sistema	55,68	11,74%	1.002,2 €
Construcción del Sistema	124,74	26,30%	2.245,3 €
Implantación y pruebas	74,27	15,66%	1.336,8 €
Documentación	49,94	10,53%	898,92 €
Formación	28,45	6,00%	426,75 €
	BASE IMPONIBLE TOTAL DE LA OFERTA		8.733,77 €
	IVA		1.397,40 €
	PRECIO TOTAL DE LA OFERTA		10.131,17 €
	PRECIO MÁXIMO ESCENARIO HIPOTÉTICO <=		10.500 €

Figura 15. Oferta económica

Tras el estudio en profundidad de los costes de la implantación del sistema y el margen de beneficio requerido, se asume por tanto un precio de venta de doce mil euros (12.000 €) IVA incluido.

Debido a la posibilidad de comercializar la aplicación para otras Federaciones (en principio existen 17 clientes potenciales de los cuales uno parece estar interesado), se plantea la posibilidad de rebajar el precio final del paquete, que quedaría en tres mil novecientos noventa y cinco euros (3,995 €) IVA incluido.

E:\Escritorio\Billgest 20100323\Billar66.accdb
 Tabla: Acta

domingo, 11 de abril de 2010
 Página: 1

Propiedades

DateCreated:	24/11/2008 21:39:17	DefaultView:	2
DisplayViewsOnSharePointSite			1
GUID:	{guid {156EC80B-A3BA-42CF-B396-BF0962D3D2A1}}	HideNewField:	Falso
LastUpdated:	04/12/2008 1:22:30	NameMap:	Datos binarios largos
OrderByOn:	Falso	OrderByOnLoad:	Verdadero
Orientation:	De izquierda a derecha	RecordCount:	0
TotalsRow:	Falso	Updatable:	Verdadero

Columnas

Nombre	Tipo	Tamaño
NUMERO	Entero largo	4
AggregateType: -1 AllowZeroLength: Falso AppendOnly: Falso Attributes: Tamaño fijo; Autoincremento CollatingOrder: 3082 ColumnHidden: Falso ColumnOrder: Predeterminado ColumnWidth: Predeterminado DataUpdatable: Falso GUID: {guid {B82D8F49-CC15-459C-A2CB-5A91DD11603B}} OrdinalPosition: 0 Required: Falso SourceField: NUMERO SourceTable: Acta TextAlign: General		
CODLIGA	Texto	255
AggregateType: -1 AllowZeroLength: Verdadero AppendOnly: Falso Attributes: Longitud variable CollatingOrder: 3082 ColumnHidden: Falso ColumnOrder: Predeterminado ColumnWidth: Predeterminado DataUpdatable: Falso DisplayControl: Cuadro de texto GUID: {guid {18460899-9669-46A4-A35D-CCA65CB1FD67}} IMEMode: 0 IMESentenceMode: 3 OrdinalPosition: 1 Required: Falso SourceField: CODLIGA SourceTable: Acta TextAlign: General UnicodeCompression: Verdadero		
TEMPORADA	Texto	255

E:\Escritorio\Billgest 20100323\Billar66.accdb
 Tabla: Acta

domingo, 11 de abril de 2010
 Página: 2

AggregateType: -1
 AllowZeroLength: Verdadero
 AppendOnly: Falso
 Attributes: Longitud variable
 CollatingOrder: 3082
 ColumnHidden: Falso
 ColumnOrder: Predeterminado
 ColumnWidth: Predeterminado
 DataUpdatable: Falso
 DisplayControl: Cuadro de texto
 GUID: {guid {E22815AE-9CB9-4BF7-8405-A0D874919D59}}
 IMEMode: 0
 IMESentenceMode: 3
 OrdinalPosition: 2
 Required: Falso
 SourceField: TEMPORADA
 SourceTable: Acta
 TextAlign: General
 UnicodeCompression: Verdadero

FECHA

Fecha/Hora

8

AggregateType: -1
 AllowZeroLength: Falso
 AppendOnly: Falso
 Attributes: Tamaño fijo
 CollatingOrder: 3082
 ColumnHidden: Falso
 ColumnOrder: Predeterminado
 ColumnWidth: Predeterminado
 DataUpdatable: Falso
 GUID: {guid {3DAD3266-CB0D-44BB-9362-42D932462E6E}}
 IMEMode: 0
 IMESentenceMode: 3
 OrdinalPosition: 3
 Required: Falso
 ShowDatePicker: Para fechas
 SourceField: FECHA
 SourceTable: Acta
 TextAlign: General

JORNADA

Entero largo

4

AggregateType: -1
 AllowZeroLength: Falso
 AppendOnly: Falso
 Attributes: Tamaño fijo
 CollatingOrder: 3082
 ColumnHidden: Falso
 ColumnOrder: Predeterminado
 ColumnWidth: Predeterminado
 DataUpdatable: Falso
 DecimalPlaces: Automático
 DisplayControl: Cuadro de texto
 GUID: {guid {26E26B58-6BDB-42EB-9DF1-82768835BBA3}}
 OrdinalPosition: 4

E:\Escritorio\Billgest 20100323\Billar66.accdb
 Tabla: Acta

domingo, 11 de abril de 2010
 Página: 3

Required: Falso
 SourceField: JORNADA
 SourceTable: Acta
 TextAlign: General

LICENCIA11 Texto 255

AggregateType: -1
 AllowZeroLength: Verdadero
 AppendOnly: Falso
 Attributes: Longitud variable
 CollatingOrder: 3082
 ColumnHidden: Falso
 ColumnOrder: Predeterminado
 ColumnWidth: Predeterminado
 DataUpdatable: Falso
 DisplayControl: Cuadro de texto
 GUID: {guid {78F113C7-5340-4CB7-B564-F57B7553ACAA}}
 IMEMode: 0
 IMESentenceMode: 3
 OrdinalPosition: 5
 Required: Falso
 SourceField: LICENCIA11
 SourceTable: Acta
 TextAlign: General
 UnicodeCompression: Verdadero

CAR11 Entero largo 4

AggregateType: -1
 AllowZeroLength: Falso
 AppendOnly: Falso
 Attributes: Tamaño fijo
 CollatingOrder: 3082
 ColumnHidden: Falso
 ColumnOrder: Predeterminado
 ColumnWidth: Predeterminado
 DataUpdatable: Falso
 DecimalPlaces: Automático
 DisplayControl: Cuadro de texto
 GUID: {guid {D3C69FDD-CB3B-4A21-9470-958107106234}}
 OrdinalPosition: 6
 Required: Falso
 SourceField: CAR11
 SourceTable: Acta
 TextAlign: General

SM11 Entero largo 4

AggregateType: -1
 AllowZeroLength: Falso
 AppendOnly: Falso
 Attributes: Tamaño fijo
 CollatingOrder: 3082
 ColumnHidden: Falso
 ColumnOrder: Predeterminado
 ColumnWidth: Predeterminado

E:\Escritorio\Billgest 20100323\Billar66.accdb
 Tabla: Acta

domingo, 11 de abril de 2010
 Página: 5

Attributes: Tamaño fijo
 CollatingOrder: 3082
 ColumnHidden: Falso
 ColumnOrder: Predeterminado
 ColumnWidth: Predeterminado
 DataUpdatable: Falso
 DecimalPlaces: Automático
 DisplayControl: Cuadro de texto
 GUID: {guid {585316F9-C2FA-4705-859F-1607E6D17A92}}
 OrdinalPosition: 10
 Required: Falso
 SourceField: SM12
 SourceTable: Acta
 TextAlign: General

ENT1 Entero largo 4

AggregateType: -1
 AllowZeroLength: Falso
 AppendOnly: Falso
 Attributes: Tamaño fijo
 CollatingOrder: 3082
 ColumnHidden: Falso
 ColumnOrder: Predeterminado
 ColumnWidth: Predeterminado
 DataUpdatable: Falso
 DecimalPlaces: Automático
 DisplayControl: Cuadro de texto
 GUID: {guid {C165640A-F7DA-4427-800D-6D9B18B9C13C}}
 OrdinalPosition: 11
 Required: Falso
 SourceField: ENT1
 SourceTable: Acta
 TextAlign: General

ARBITRO1 Texto 255

AggregateType: -1
 AllowZeroLength: Verdadero
 AppendOnly: Falso
 Attributes: Longitud variable
 CollatingOrder: 3082
 ColumnHidden: Falso
 ColumnOrder: Predeterminado
 ColumnWidth: Predeterminado
 DataUpdatable: Falso
 DisplayControl: Cuadro de texto
 GUID: {guid {14050CBD-DC88-47E9-AA58-E39FD64A0BCD}}
 IMEMode: 0
 IMESentenceMode: 3
 OrdinalPosition: 12
 Required: Falso
 SourceField: ARBITRO1
 SourceTable: Acta
 TextAlign: General
 UnicodeCompression: Verdadero

E:\Escritorio\Billgest 20100323\Billar66.accdb
 Tabla: Acta

domingo, 11 de abril de 2010
 Página: 6

LICENCIA21		Texto	255
AggregateType:	-1		
AllowZeroLength:	Verdadero		
AppendOnly:	Falso		
Attributes:	Longitud variable		
CollatingOrder:	3082		
ColumnHidden:	Falso		
ColumnOrder:	Predeterminado		
ColumnWidth:	Predeterminado		
DataUpdatable:	Falso		
DisplayControl:	Cuadro de texto		
GUID:	{guid {73C136F3-56D4-48C8-83F3-D9C869152092}}		
IMEMode:	0		
IMESentenceMode:	3		
OrdinalPosition:	13		
Required:	Falso		
SourceField:	LICENCIA21		
SourceTable:	Acta		
TextAlign:	General		
UnicodeCompression:	Verdadero		
CAR21		Entero largo	4
AggregateType:	-1		
AllowZeroLength:	Falso		
AppendOnly:	Falso		
Attributes:	Tamaño fijo		
CollatingOrder:	3082		
ColumnHidden:	Falso		
ColumnOrder:	Predeterminado		
ColumnWidth:	Predeterminado		
DataUpdatable:	Falso		
DecimalPlaces:	Automático		
DisplayControl:	Cuadro de texto		
GUID:	{guid {F2A385D5-7FCF-4FB5-AB97-E75247B2EA89}}		
OrdinalPosition:	14		
Required:	Falso		
SourceField:	CAR21		
SourceTable:	Acta		
TextAlign:	General		
SM21		Entero largo	4
AggregateType:	-1		
AllowZeroLength:	Falso		
AppendOnly:	Falso		
Attributes:	Tamaño fijo		
CollatingOrder:	3082		
ColumnHidden:	Falso		
ColumnOrder:	Predeterminado		
ColumnWidth:	Predeterminado		
DataUpdatable:	Falso		
DecimalPlaces:	Automático		
DisplayControl:	Cuadro de texto		
GUID:	{guid {46FE7C25-120A-452B-AD72-40CD2A84A81E}}		
OrdinalPosition:	15		

E:\Escritorio\Billgest 20100323\Billar66.accdb
 Tabla: Acta

domingo, 11 de abril de 2010
 Página: 7

Required: Falso
 SourceField: SM21
 SourceTable: Acta
 TextAlign: General

LICENCIA22 Texto 255

AggregateType: -1
 AllowZeroLength: Verdadero
 AppendOnly: Falso
 Attributes: Longitud variable
 CollatingOrder: 3082
 ColumnHidden: Falso
 ColumnOrder: Predeterminado
 ColumnWidth: Predeterminado
 DataUpdatable: Falso
 DisplayControl: Cuadro de texto
 GUID: {guid {A328A428-0FF4-45F3-8C33-4284563ED32C}}
 IMEMode: 0
 IMESentenceMode: 3
 OrdinalPosition: 16
 Required: Falso
 SourceField: LICENCIA22
 SourceTable: Acta
 TextAlign: General
 UnicodeCompression: Verdadero

CAR22 Entero largo 4

AggregateType: -1
 AllowZeroLength: Falso
 AppendOnly: Falso
 Attributes: Tamaño fijo
 CollatingOrder: 3082
 ColumnHidden: Falso
 ColumnOrder: Predeterminado
 ColumnWidth: Predeterminado
 DataUpdatable: Falso
 DecimalPlaces: Automático
 DisplayControl: Cuadro de texto
 GUID: {guid {242F3E8B-3247-42BA-82A0-192359E8BDE1}}
 OrdinalPosition: 17
 Required: Falso
 SourceField: CAR22
 SourceTable: Acta
 TextAlign: General

SM22 Entero largo 4

AggregateType: -1
 AllowZeroLength: Falso
 AppendOnly: Falso
 Attributes: Tamaño fijo
 CollatingOrder: 3082
 ColumnHidden: Falso
 ColumnOrder: Predeterminado
 ColumnWidth: Predeterminado

E:\Escritorio\Billgest 20100323\Billar66.accdb
 Tabla: Acta

domingo, 11 de abril de 2010
 Página: 8

DataUpdatable: Falso
 DecimalPlaces: Automático
 DisplayControl: Cuadro de texto
 GUID: {guid {A7578188-B497-43C2-9C37-A4FEEAA0049C}}
 OrdinalPosition: 18
 Required: Falso
 SourceField: SM22
 SourceTable: Acta
 TextAlign: General

ENT2 Entero largo 4

AggregateType: -1
 AllowZeroLength: Falso
 AppendOnly: Falso
 Attributes: Tamaño fijo
 CollatingOrder: 3082
 ColumnHidden: Falso
 ColumnOrder: Predeterminado
 ColumnWidth: Predeterminado
 DataUpdatable: Falso
 DecimalPlaces: Automático
 DisplayControl: Cuadro de texto
 GUID: {guid {21119CDB-B5E4-4AE7-B252-9748C7CFA330}}
 OrdinalPosition: 19
 Required: Falso
 SourceField: ENT2
 SourceTable: Acta
 TextAlign: General

ARBITRO2 Texto 255

AggregateType: -1
 AllowZeroLength: Verdadero
 AppendOnly: Falso
 Attributes: Longitud variable
 CollatingOrder: 3082
 ColumnHidden: Falso
 ColumnOrder: Predeterminado
 ColumnWidth: Predeterminado
 DataUpdatable: Falso
 DisplayControl: Cuadro de texto
 GUID: {guid {412F83AC-88AE-4328-93BA-CE3C08D33A9D}}
 IMEMode: 0
 IMESentenceMode: 3
 OrdinalPosition: 20
 Required: Falso
 SourceField: ARBITRO2
 SourceTable: Acta
 TextAlign: General
 UnicodeCompression: Verdadero

LICENCIA31 Texto 255

AggregateType: -1
 AllowZeroLength: Verdadero
 AppendOnly: Falso

E:\Escritorio\Billgest 20100323\Billar66.accdb
 Tabla: Acta

domingo, 11 de abril de 2010
 Página: 9

Attributes: Longitud variable
 CollatingOrder: 3082
 ColumnHidden: Falso
 ColumnOrder: Predeterminado
 ColumnWidth: Predeterminado
 DataUpdatable: Falso
 DisplayControl: Cuadro de texto
 GUID: {guid {81214511-E72E-4808-83BF-348D1077CC84}}
 IMEMode: 0
 IMESentenceMode: 3
 OrdinalPosition: 21
 Required: Falso
 SourceField: LICENCIA31
 SourceTable: Acta
 TextAlign: General
 UnicodeCompression: Verdadero

CAR31 Entero largo 4

AggregateType: -1
 AllowZeroLength: Falso
 AppendOnly: Falso
 Attributes: Tamaño fijo
 CollatingOrder: 3082
 ColumnHidden: Falso
 ColumnOrder: Predeterminado
 ColumnWidth: Predeterminado
 DataUpdatable: Falso
 DecimalPlaces: Automático
 DisplayControl: Cuadro de texto
 GUID: {guid {48CF952E-25F8-41A8-857B-57717A748F72}}
 OrdinalPosition: 22
 Required: Falso
 SourceField: CAR31
 SourceTable: Acta
 TextAlign: General

SM31 Entero largo 4

AggregateType: -1
 AllowZeroLength: Falso
 AppendOnly: Falso
 Attributes: Tamaño fijo
 CollatingOrder: 3082
 ColumnHidden: Falso
 ColumnOrder: Predeterminado
 ColumnWidth: Predeterminado
 DataUpdatable: Falso
 DecimalPlaces: Automático
 DisplayControl: Cuadro de texto
 GUID: {guid {9C6297BC-9DC0-40D9-9E8F-B3604DFFD542}}
 OrdinalPosition: 23
 Required: Falso
 SourceField: SM31
 SourceTable: Acta
 TextAlign: General

E:\Escritorio\Billgest 20100323\Billar66.accdb
 Tabla: Acta

domingo, 11 de abril de 2010
 Página: 10

<p>LICENCIA32</p> <p>AggregateType: -1 AllowZeroLength: Verdadero AppendOnly: Falso Attributes: Longitud variable CollatingOrder: 3082 ColumnHidden: Falso ColumnOrder: Predeterminado ColumnWidth: Predeterminado DataUpdatable: Falso DisplayControl: Cuadro de texto GUID: {F9F974AB-6D78-4AE4-8843-F5EDEB56C705} IMEMode: 0 IMESentenceMode: 3 OrdinalPosition: 24 Required: Falso SourceField: LICENCIA32 SourceTable: Acta TextAlign: General UnicodeCompression: Verdadero</p>	<p>Texto</p>	<p>255</p>
<p>CAR32</p> <p>AggregateType: -1 AllowZeroLength: Falso AppendOnly: Falso Attributes: Tamaño fijo CollatingOrder: 3082 ColumnHidden: Falso ColumnOrder: Predeterminado ColumnWidth: Predeterminado DataUpdatable: Falso DecimalPlaces: Automático DisplayControl: Cuadro de texto GUID: {5D58D62A-E9CF-4438-9B39-FC8ACBCE6CC4} OrdinalPosition: 25 Required: Falso SourceField: CAR32 SourceTable: Acta TextAlign: General</p>	<p>Entero largo</p>	<p>4</p>
<p>SM32</p> <p>AggregateType: -1 AllowZeroLength: Falso AppendOnly: Falso Attributes: Tamaño fijo CollatingOrder: 3082 ColumnHidden: Falso ColumnOrder: Predeterminado ColumnWidth: Predeterminado DataUpdatable: Falso DecimalPlaces: Automático DisplayControl: Cuadro de texto GUID: {AF395FD2-A52C-450B-8E8E-0F494BFD2759} OrdinalPosition: 26</p>	<p>Entero largo</p>	<p>4</p>

E:\Escritorio\Billgest 20100323\Billar66.accdb
 Tabla: Acta

domingo, 11 de abril de 2010
 Página: 11

Required: Falso
 SourceField: SM32
 SourceTable: Acta
 TextAlign: General

ENT3 Entero largo 4

AggregateType: -1
 AllowZeroLength: Falso
 AppendOnly: Falso
 Attributes: Tamaño fijo
 CollatingOrder: 3082
 ColumnHidden: Falso
 ColumnOrder: Predeterminado
 ColumnWidth: Predeterminado
 DataUpdatable: Falso
 DecimalPlaces: Automático
 DisplayControl: Cuadro de texto
 GUID: {guid {847DD992-5FF2-4B5E-AEA1-ED20AA725A6E}}
 OrdinalPosition: 27
 Required: Falso
 SourceField: ENT3
 SourceTable: Acta
 TextAlign: General

ARBITRO3 Texto 255

AggregateType: -1
 AllowZeroLength: Verdadero
 AppendOnly: Falso
 Attributes: Longitud variable
 CollatingOrder: 3082
 ColumnHidden: Falso
 ColumnOrder: Predeterminado
 ColumnWidth: Predeterminado
 DataUpdatable: Falso
 DisplayControl: Cuadro de texto
 GUID: {guid {5F8F114E-62C2-4CBE-B86D-E48717C8BFDB}}
 IMEMode: 0
 IMESentenceMode: 3
 OrdinalPosition: 28
 Required: Falso
 SourceField: ARBITRO3
 SourceTable: Acta
 TextAlign: General
 UnicodeCompression: Verdadero

LICENCIA41 Texto 255

AggregateType: -1
 AllowZeroLength: Verdadero
 AppendOnly: Falso
 Attributes: Longitud variable
 CollatingOrder: 3082
 ColumnHidden: Falso
 ColumnOrder: Predeterminado
 ColumnWidth: Predeterminado

E:\Escritorio\Billgest 20100323\Billar66.accdb
 Tabla: Acta

domingo, 11 de abril de 2010
 Página: 12

	DataUpdatable:	Falso		
	DisplayControl:	Cuadro de texto		
	GUID:	{guid {D0BE727F-C710-4296-AEEE-C3B57BC56BB5}}		
	IMEMode:	0		
	IMESentenceMode:	3		
	OrdinalPosition:	29		
	Required:	Falso		
	SourceField:	LICENCIA41		
	SourceTable:	Acta		
	TextAlign:	General		
	UnicodeCompression:	Verdadero		
CAR41			Entero largo	4
	AggregateType:	-1		
	AllowZeroLength:	Falso		
	AppendOnly:	Falso		
	Attributes:	Tamaño fijo		
	CollatingOrder:	3082		
	ColumnHidden:	Falso		
	ColumnOrder:	Predeterminado		
	ColumnWidth:	Predeterminado		
	DataUpdatable:	Falso		
	DecimalPlaces:	Automático		
	DisplayControl:	Cuadro de texto		
	GUID:	{guid {F984C514-C1DD-4C2A-8F47-EB583C5DA80B}}		
	OrdinalPosition:	30		
	Required:	Falso		
	SourceField:	CAR41		
	SourceTable:	Acta		
	TextAlign:	General		
SM41			Entero largo	4
	AggregateType:	-1		
	AllowZeroLength:	Falso		
	AppendOnly:	Falso		
	Attributes:	Tamaño fijo		
	CollatingOrder:	3082		
	ColumnHidden:	Falso		
	ColumnOrder:	Predeterminado		
	ColumnWidth:	Predeterminado		
	DataUpdatable:	Falso		
	DecimalPlaces:	Automático		
	DisplayControl:	Cuadro de texto		
	GUID:	{guid {19FBAAC1-B42A-4B16-AF3C-E7B83DE597A1}}		
	OrdinalPosition:	31		
	Required:	Falso		
	SourceField:	SM41		
	SourceTable:	Acta		
	TextAlign:	General		
LICENCIA42			Texto	255
	AggregateType:	-1		
	AllowZeroLength:	Verdadero		
	AppendOnly:	Falso		

E:\Escritorio\Billgest 20100323\Billar66.accdb
 Tabla: Acta

domingo, 11 de abril de 2010
 Página: 13

Attributes: Longitud variable
 CollatingOrder: 3082
 ColumnHidden: Falso
 ColumnOrder: Predeterminado
 ColumnWidth: Predeterminado
 DataUpdatable: Falso
 DisplayControl: Cuadro de texto
 GUID: {guid {2EB245DF-F49B-4792-A4E9-B9207170DD0C}}
 IMEMode: 0
 IMESentenceMode: 3
 OrdinalPosition: 32
 Required: Falso
 SourceField: LICENCIA42
 SourceTable: Acta
 TextAlign: General
 UnicodeCompression: Verdadero

CAR42 Entero largo 4

AggregateType: -1
 AllowZeroLength: Falso
 AppendOnly: Falso
 Attributes: Tamaño fijo
 CollatingOrder: 3082
 ColumnHidden: Falso
 ColumnOrder: Predeterminado
 ColumnWidth: Predeterminado
 DataUpdatable: Falso
 DecimalPlaces: Automático
 DisplayControl: Cuadro de texto
 GUID: {guid {CEC436E2-9809-4358-BAD1-66F58AFEFF7A}}
 OrdinalPosition: 33
 Required: Falso
 SourceField: CAR42
 SourceTable: Acta
 TextAlign: General

SM42 Entero largo 4

AggregateType: -1
 AllowZeroLength: Falso
 AppendOnly: Falso
 Attributes: Tamaño fijo
 CollatingOrder: 3082
 ColumnHidden: Falso
 ColumnOrder: Predeterminado
 ColumnWidth: Predeterminado
 DataUpdatable: Falso
 DecimalPlaces: Automático
 DisplayControl: Cuadro de texto
 GUID: {guid {25DF7FC2-3725-4A78-8E41-823426881E08}}
 OrdinalPosition: 34
 Required: Falso
 SourceField: SM42
 SourceTable: Acta
 TextAlign: General

E:\Escritorio\Billgest 20100323\Billar66.accdb
 Tabla: Acta

domingo, 11 de abril de 2010
 Página: 14

<p>ENT4</p> <p>AggregateType: -1 AllowZeroLength: Falso AppendOnly: Falso Attributes: Tamaño fijo CollatingOrder: 3082 ColumnHidden: Falso ColumnOrder: Predeterminado ColumnWidth: Predeterminado DataUpdatable: Falso DecimalPlaces: Automático DisplayControl: Cuadro de texto GUID: {guid {F3D4DA2A-0DEB-4C5D-8EAA-853B9A93B43C}} OrdinalPosition: 35 Required: Falso SourceField: ENT4 SourceTable: Acta TextAlign: General</p>	<p>Entero largo</p>	<p>4</p>
<p>ARBITRO4</p> <p>AggregateType: -1 AllowZeroLength: Verdadero AppendOnly: Falso Attributes: Longitud variable CollatingOrder: 3082 ColumnHidden: Falso ColumnOrder: Predeterminado ColumnWidth: Predeterminado DataUpdatable: Falso DisplayControl: Cuadro de texto GUID: {guid {A1746DDA-745D-458C-B28B-C9F8CCDFF57B}} IMEMode: 0 IMESentenceMode: 3 OrdinalPosition: 36 Required: Falso SourceField: ARBITRO4 SourceTable: Acta TextAlign: General UnicodeCompression: Verdadero</p>	<p>Texto</p>	<p>255</p>
<p>CLUB1</p> <p>AggregateType: -1 AllowZeroLength: Verdadero AppendOnly: Falso Attributes: Longitud variable CollatingOrder: 3082 ColumnHidden: Falso ColumnOrder: Predeterminado ColumnWidth: Predeterminado DataUpdatable: Falso DisplayControl: Cuadro de texto GUID: {guid {58F3BA5A-4033-42CA-8080-FDE19E3578B8}} IMEMode: 0 IMESentenceMode: 3</p>	<p>Texto</p>	<p>255</p>

E:\Escritorio\Billgest 20100323\Billar66.accdb
 Tabla: Acta

domingo, 11 de abril de 2010
 Página: 15

OrdinalPosition: 37 Required: Falso SourceField: CLUB1 SourceTable: Acta TextAlign: General UnicodeCompression: Verdadero			
PAR1	Entero largo		4
AggregateType: -1 AllowZeroLength: Falso AppendOnly: Falso Attributes: Tamaño fijo CollatingOrder: 3082 ColumnHidden: Falso ColumnOrder: Predeterminado ColumnWidth: Predeterminado DataUpdatable: Falso DecimalPlaces: Automático DisplayControl: Cuadro de texto GUID: {guid {4F85851F-42C2-483F-A67F-A47FD7B1EA3D}} OrdinalPosition: 38 Required: Falso SourceField: PAR1 SourceTable: Acta TextAlign: General			
MATCH1	Entero largo		4
AggregateType: -1 AllowZeroLength: Falso AppendOnly: Falso Attributes: Tamaño fijo CollatingOrder: 3082 ColumnHidden: Falso ColumnOrder: Predeterminado ColumnWidth: Predeterminado DataUpdatable: Falso DecimalPlaces: Automático DisplayControl: Cuadro de texto GUID: {guid {06A71804-5C66-4C8B-AE23-074ED1C2DDB5}} OrdinalPosition: 39 Required: Falso SourceField: MATCH1 SourceTable: Acta TextAlign: General			
CARAM1	Entero largo		4
AggregateType: -1 AllowZeroLength: Falso AppendOnly: Falso Attributes: Tamaño fijo CollatingOrder: 3082 ColumnHidden: Falso ColumnOrder: Predeterminado ColumnWidth: Predeterminado			

E:\Escritorio\Billgest 20100323\Billar66.accdb
 Tabla: Acta

domingo, 11 de abril de 2010
 Página: 16

DataUpdatable: Falso
 DecimalPlaces: Automático
 DisplayControl: Cuadro de texto
 GUID: {guid {17B687D4-253A-4875-A43F-04A52D8D965F}}
 OrdinalPosition: 40
 Required: Falso
 SourceField: CARAM1
 SourceTable: Acta
 TextAlign: General

CLUB2 Texto 255

AggregateType: -1
 AllowZeroLength: Verdadero
 AppendOnly: Falso
 Attributes: Longitud variable
 CollatingOrder: 3082
 ColumnHidden: Falso
 ColumnOrder: Predeterminado
 ColumnWidth: Predeterminado
 DataUpdatable: Falso
 DisplayControl: Cuadro de texto
 GUID: {guid {49159041-D842-4075-A3C5-7938233D87EA}}
 IMEMode: 0
 IMESentenceMode: 3
 OrdinalPosition: 41
 Required: Falso
 SourceField: CLUB2
 SourceTable: Acta
 TextAlign: General
 UnicodeCompression: Verdadero

PAR2 Entero largo 4

AggregateType: -1
 AllowZeroLength: Falso
 AppendOnly: Falso
 Attributes: Tamaño fijo
 CollatingOrder: 3082
 ColumnHidden: Falso
 ColumnOrder: Predeterminado
 ColumnWidth: Predeterminado
 DataUpdatable: Falso
 DecimalPlaces: Automático
 DisplayControl: Cuadro de texto
 GUID: {guid {F64D282F-F307-4AA4-B7FE-DD3ADC7782DC}}
 OrdinalPosition: 42
 Required: Falso
 SourceField: PAR2
 SourceTable: Acta
 TextAlign: General

MATCH2 Entero largo 4

AggregateType: -1
 AllowZeroLength: Falso
 AppendOnly: Falso

E:\Escritorio\Billgest 20100323\Billar66.accdb
 Tabla: Acta

domingo, 11 de abril de 2010
 Página: 17

Attributes: Tamaño fijo
 CollatingOrder: 3082
 ColumnHidden: Falso
 ColumnOrder: Predeterminado
 ColumnWidth: Predeterminado
 DataUpdatable: Falso
 DecimalPlaces: Automático
 DisplayControl: Cuadro de texto
 GUID: {guid {423DE690-8432-436E-8DDB-D4FDFBC1C41C}}
 OrdinalPosition: 43
 Required: Falso
 SourceField: MATCH2
 SourceTable: Acta
 TextAlign: General

CARAM2 Entero largo 4

AggregateType: -1
 AllowZeroLength: Falso
 AppendOnly: Falso
 Attributes: Tamaño fijo
 CollatingOrder: 3082
 ColumnHidden: Falso
 ColumnOrder: Predeterminado
 ColumnWidth: Predeterminado
 DataUpdatable: Falso
 DecimalPlaces: Automático
 DisplayControl: Cuadro de texto
 GUID: {8EE91ED0-EE22-43C5-99FC-8B53D318D122}}
 OrdinalPosition: 44
 Required: Falso
 SourceField: CARAM2
 SourceTable: Acta
 TextAlign: General

ENTCLUB Entero largo 4

AggregateType: -1
 AllowZeroLength: Falso
 AppendOnly: Falso
 Attributes: Tamaño fijo
 CollatingOrder: 3082
 ColumnHidden: Falso
 ColumnOrder: Predeterminado
 ColumnWidth: Predeterminado
 DataUpdatable: Falso
 DecimalPlaces: Automático
 DisplayControl: Cuadro de texto
 GUID: {F7F0F2F4-F31D-4741-ADC0-B4D1984C285D}}
 OrdinalPosition: 45
 Required: Falso
 SourceField: ENTCLUB
 SourceTable: Acta
 TextAlign: General

Relaciones

ClubActa

Club	Acta
NOMBRE	1 ∞ CLUB1
Attributes: RelationshipType:	Forzado; Actualizaciones en cascada; Eliminationes en cascada Uno a varios

ClubActa

Club	Acta
NOMBRE	1 ∞ CLUB2
Attributes: RelationshipType:	Forzado; Actualizaciones en cascada; Eliminationes en cascada Uno a varios

SocioActa

Socio	Acta
LICENCIA	1 ∞ LICENCIA12
Attributes: RelationshipType:	Forzado; Actualizaciones en cascada; Eliminationes en cascada Uno a varios

SocioActa

Socio	Acta
LICENCIA	1 ∞ LICENCIA21
Attributes: RelationshipType:	Forzado; Actualizaciones en cascada; Eliminationes en cascada Uno a varios

SocioActa

Socio	1	∞	Acta
LICENCIA			LICENCIA22
Attributes:		Forzado; Actualizaciones en cascada; Eliminationes en cascada	
RelationshipType:		Uno a varios	

SocioActa

Socio	1	∞	Acta
LICENCIA			LICENCIA31
Attributes:		Forzado; Actualizaciones en cascada; Eliminationes en cascada	
RelationshipType:		Uno a varios	

SocioActa

Socio	1	∞	Acta
LICENCIA			LICENCIA32
Attributes:		Forzado; Actualizaciones en cascada; Eliminationes en cascada	
RelationshipType:		Uno a varios	

SocioActa

Socio	1	∞	Acta
LICENCIA			LICENCIA41
Attributes:		Forzado; Actualizaciones en cascada; Eliminationes en cascada	
RelationshipType:		Uno a varios	

SocioActa

Socio	1	∞	Acta
LICENCIA			LICENCIA42
Attributes:		Forzado; Actualizaciones en cascada; Eliminationes en cascada	
RelationshipType:		Uno a varios	

E:\Escritorio\Billgest 20100323\Billar66.accdb
 Tabla: Acta

domingo, 11 de abril de 2010
 Página: 20

TemporadaActa

Temporada	Acta
Temporada	TEMPORADA
Attributes:	No forzado
RelationshipType:	Uno a varios

Índices de tabla

Nombre	Número de campos
ActaTemporada	1
Clustered:	Falso
DistinctCount:	7
Foreign:	Falso
IgnoreNulls:	Falso
Name:	ActaTemporada
Primary:	Falso
Required:	Falso
Unique:	Falso
Campos:	
TEMPORADA	Ascendente
ClubActa	1
Clustered:	Falso
DistinctCount:	1
Foreign:	Verdadero
IgnoreNulls:	Falso
Name:	ClubActa
Primary:	Falso
Required:	Falso
Unique:	Falso
Campos:	
CLUB1	Ascendente
ClubActa1	1
Clustered:	Falso
DistinctCount:	1
Foreign:	Verdadero
IgnoreNulls:	Falso
Name:	ClubActa1
Primary:	Falso
Required:	Falso
Unique:	Falso

E:\Escritorio\Billgest 20100323\Billar66.accdb
 Tabla: Acta

domingo, 11 de abril de 2010
 Página: 21

Campos:
 CLUB2 Ascendente

PrimaryKey 1
 Clustered: Falso
 DistinctCount: 21
 Foreign: Falso
 IgnoreNulls: Falso
 Name: PrimaryKey
 Primary: Verdadero
 Required: Verdadero
 Unique: Verdadero
 Campos:
 NUMERO Ascendente

SocioActa1 1
 Clustered: Falso
 DistinctCount: 3
 Foreign: Verdadero
 IgnoreNulls: Falso
 Name: SocioActa1
 Primary: Falso
 Required: Falso
 Unique: Falso
 Campos:
 LICENCIA12 Ascendente

SocioActa2 1
 Clustered: Falso
 DistinctCount: 1
 Foreign: Verdadero
 IgnoreNulls: Falso
 Name: SocioActa2
 Primary: Falso
 Required: Falso
 Unique: Falso
 Campos:
 LICENCIA21 Ascendente

SocioActa3 1
 Clustered: Falso
 DistinctCount: 4
 Foreign: Verdadero

E:\Escritorio\Billgest 20100323\Billar66.accdb
 Tabla: Acta

domingo, 11 de abril de 2010
 Página: 22

IgnoreNulls: Falso
 Name: SocioActa3
 Primary: Falso
 Required: Falso
 Unique: Falso
 Campos: LICENCIA22 Ascendente

SocioActa4 1
 Clustered: Falso
 DistinctCount: 1
 Foreign: Verdadero
 IgnoreNulls: Falso
 Name: SocioActa4
 Primary: Falso
 Required: Falso
 Unique: Falso
 Campos: LICENCIA31 Ascendente

SocioActa5 1
 Clustered: Falso
 DistinctCount: 4
 Foreign: Verdadero
 IgnoreNulls: Falso
 Name: SocioActa5
 Primary: Falso
 Required: Falso
 Unique: Falso
 Campos: LICENCIA32 Ascendente

SocioActa6 1
 Clustered: Falso
 DistinctCount: 1
 Foreign: Verdadero
 IgnoreNulls: Falso
 Name: SocioActa6
 Primary: Falso
 Required: Falso
 Unique: Falso

E:\Escritorio\Billgest 20100323\Billar66.accdb
 Tabla: Acta

domingo, 11 de abril de 2010
 Página: 23

Campos:
 LICENCIA41 Ascendente

SocioActa7 1
 Clustered: Falso
 DistinctCount: 4
 Foreign: Verdadero
 IgnoreNulls: Falso
 Name: SocioActa7
 Primary: Falso
 Required: Falso
 Unique: Falso
 Campos:
 LICENCIA42 Ascendente

Permisos de usuario

admin Eliminar; Leer permisos; Establecer permisos; Cambiar propietario, Leer definición; Escribir definición; Leer datos; Insertar datos; Actualizar datos; Eliminar datos

Permisos de grupo

Admins Eliminar; Leer permisos; Establecer permisos; Cambiar propietario, Leer definición; Escribir definición; Leer datos; Insertar datos; Actualizar datos; Eliminar datos

Users Eliminar; Leer permisos; Establecer permisos; Cambiar propietario, Leer definición; Escribir definición; Leer datos; Insertar datos; Actualizar datos; Eliminar datos

E:\Escritorio\Billgest 20100323\Billar66.accdb
 Tabla: Clasificacion_club

domingo, 11 de abril de 2010
 Página: 24

Propiedades

DateCreated:	23/11/2008 2:55:07	DefaultView:	2
DisplayViewsOnSharePointSite			1
GUID:	{guid {AB5C23D4-E628-45F3-8E65-D7625B53BCE5}}	HideNewField:	Falso
LastUpdated:	24/02/2009 2:27:24	NameMap:	Datos binarios largos
OrderBy:	[Clasificacion_club].[CODLIGA]	OrderByOn:	Verdadero
	,'		
	[Clasificacion_club].[CODIGO]		
OrderByOnLoad:	Verdadero	Orientation:	De izquierda a derecha
RecordCount:	201	TotalsRow:	Falso
Updatable:	Verdadero		

Columnas

Nombre	Tipo	Tamaño
CODIGO	Entero largo	4
AggregateType:	-1	
AllowZeroLength:	Falso	
AppendOnly:	Falso	
Attributes:	Tamaño fijo; Autoincremento	
CollatingOrder:	3082	
ColumnHidden:	Falso	
ColumnOrder:	1	
ColumnWidth:	Predeterminado	
DataUpdatable:	Falso	
OrdinalPosition:	0	
Required:	Falso	
SourceField:	CODIGO	
SourceTable:	Clasificacion_club	
TextAlign:	General	
EQUIPO	Texto	38
AggregateType:	-1	
AllowZeroLength:	Falso	
AppendOnly:	Falso	
Attributes:	Longitud variable	
CollatingOrder:	3082	
ColumnHidden:	Falso	
ColumnOrder:	2	
ColumnWidth:	3288	
DataUpdatable:	Falso	
DisplayControl:	Cuadro de texto	
GUID:	{guid {FC118F31-4CD4-4D78-9508-DBAE53C5FE27}}	
IMEMode:	0	
IMESentenceMode:	3	
OrdinalPosition:	1	
Required:	Verdadero	
SourceField:	EQUIPO	
SourceTable:	Clasificacion_club	
TextAlign:	General	
UnicodeCompression:	Falso	

E:\Escritorio\Billgest 20100323\Billar66.accdb
 Tabla: Clasificacion_club

domingo, 11 de abril de 2010
 Página: 25

<p>TEMPORADA</p> <p>AggregateType: -1 AllowZeroLength: Falso AppendOnly: Falso Attributes: Longitud variable CollatingOrder: 3082 ColumnHidden: Falso ColumnOrder: 3 ColumnWidth: Predeterminado DataUpdatable: Falso DisplayControl: Cuadro de texto IMEMode: 0 IMESentenceMode: 3 OrdinalPosition: 2 Required: Verdadero SourceField: TEMPORADA SourceTable: Clasificacion_club TextAlign: General UnicodeCompression: Verdadero</p>	<p>Texto</p>	<p>255</p>
<p>CODLIGA</p> <p>AggregateType: -1 AllowZeroLength: Falso AppendOnly: Falso Attributes: Longitud variable CollatingOrder: 3082 ColumnHidden: Falso ColumnOrder: 4 ColumnWidth: Predeterminado DataUpdatable: Falso DisplayControl: Cuadro de texto IMEMode: 0 IMESentenceMode: 3 OrdinalPosition: 3 Required: Verdadero SourceField: CODLIGA SourceTable: Clasificacion_club TextAlign: General UnicodeCompression: Verdadero</p>	<p>Texto</p>	<p>10</p>
<p>PJ</p> <p>AggregateType: -1 AllowZeroLength: Falso AppendOnly: Falso Attributes: Tamaño fijo CollatingOrder: 3082 ColumnHidden: Falso ColumnOrder: 5 ColumnWidth: Predeterminado DataUpdatable: Falso DecimalPlaces: Automático DisplayControl: Cuadro de texto GUID: {guid {113573AE-328C-468B-91AB-6613A60D1360}} OrdinalPosition: 4</p>	<p>Entero largo</p>	<p>4</p>

E:\Escritorio\Billgest 20100323\Billar66.accdb
 Tabla: Clasificacion_club

domingo, 11 de abril de 2010
 Página: 26

	Required:	Falso	
	SourceField:	PJ	
	SourceTable:	Clasificacion_club	
	TextAlign:	General	
PG			Entero largo 4
	AggregateType:	-1	
	AllowZeroLength:	Falso	
	AppendOnly:	Falso	
	Attributes:	Tamaño fijo	
	CollatingOrder:	3082	
	ColumnHidden:	Falso	
	ColumnOrder:	6	
	ColumnWidth:	Predeterminado	
	DataUpdatable:	Falso	
	DecimalPlaces:	Automático	
	DisplayControl:	Cuadro de texto	
	GUID:	{guid {489A7CA2-C9F6-433F-AA17-7F44E068D773}}	
	OrdinalPosition:	5	
	Required:	Falso	
	SourceField:	PG	
	SourceTable:	Clasificacion_club	
	TextAlign:	General	
PE			Entero largo 4
	AggregateType:	-1	
	AllowZeroLength:	Falso	
	AppendOnly:	Falso	
	Attributes:	Tamaño fijo	
	CollatingOrder:	3082	
	ColumnHidden:	Falso	
	ColumnOrder:	7	
	ColumnWidth:	Predeterminado	
	DataUpdatable:	Falso	
	DecimalPlaces:	Automático	
	DisplayControl:	Cuadro de texto	
	GUID:	{guid {F0B908C5-F30A-4ED7-B3AC-AC27FE2C3DF0}}	
	OrdinalPosition:	6	
	Required:	Falso	
	SourceField:	PE	
	SourceTable:	Clasificacion_club	
	TextAlign:	General	
PP			Entero largo 4
	AggregateType:	-1	
	AllowZeroLength:	Falso	
	AppendOnly:	Falso	
	Attributes:	Tamaño fijo	
	CollatingOrder:	3082	
	ColumnHidden:	Falso	
	ColumnOrder:	8	
	ColumnWidth:	Predeterminado	
	DataUpdatable:	Falso	
	DecimalPlaces:	Automático	

E:\Escritorio\Billgest 20100323\Billar66.accdb
 Tabla: Clasificacion_club

domingo, 11 de abril de 2010
 Página: 27

	DisplayControl: Cuadro de texto GUID: {guid {FD0A5DA3-AC46-4B0D-A62A-C00BFAB9CCEA}} OrdinalPosition: 7 Required: Falso SourceField: PP SourceTable: Clasificacion_club TextAlign: General		
CAR		Entero largo	4
	AggregateType: -1 AllowZeroLength: Falso AppendOnly: Falso Attributes: Tamaño fijo CollatingOrder: 3082 ColumnHidden: Falso ColumnOrder: 9 ColumnWidth: Predeterminado DataUpdatable: Falso DecimalPlaces: Automático DisplayControl: Cuadro de texto GUID: {guid {458D8725-C907-44EE-BC9D-263FCE7BA243}} OrdinalPosition: 8 Required: Falso SourceField: CAR SourceTable: Clasificacion_club TextAlign: General		
ENT		Entero largo	4
	AggregateType: -1 AllowZeroLength: Falso AppendOnly: Falso Attributes: Tamaño fijo CollatingOrder: 3082 ColumnHidden: Falso ColumnOrder: 10 ColumnWidth: Predeterminado DataUpdatable: Falso DecimalPlaces: Automático DisplayControl: Cuadro de texto GUID: {guid {71FF6F6A-B6A2-46E9-B869-EB509EBC0F6C}} OrdinalPosition: 9 Required: Falso SourceField: ENT SourceTable: Clasificacion_club TextAlign: General		
PROM		Doble	8
	AggregateType: -1 AllowZeroLength: Falso AppendOnly: Falso Attributes: Tamaño fijo CollatingOrder: 3082 ColumnHidden: Falso ColumnOrder: 11		

E:\Escritorio\Billgest 20100323\Billar66.accdb
 Tabla: Clasificacion_club

domingo, 11 de abril de 2010
 Página: 28

	ColumnWidth:	Predeterminado	
	DataUpdatable:	Falso	
	DecimalPlaces:	Automático	
	DisplayControl:	Cuadro de texto	
	GUID:	{guid {4949894C-14AA-4B6C-8828-7BD28F9E013B}}	
	OrdinalPosition:	10	
	Required:	Falso	
	SourceField:	PROM	
	SourceTable:	Clasificacion_club	
	TextAlign:	General	
PPA			Entero largo 4
	AggregateType:	-1	
	AllowZeroLength:	Falso	
	AppendOnly:	Falso	
	Attributes:	Tamaño fijo	
	CollatingOrder:	3082	
	ColumnHidden:	Falso	
	ColumnOrder:	12	
	ColumnWidth:	Predeterminado	
	DataUpdatable:	Falso	
	DecimalPlaces:	Automático	
	DisplayControl:	Cuadro de texto	
	GUID:	{guid {170602D5-983A-45D0-A99B-D7B669E4A604}}	
	OrdinalPosition:	11	
	Required:	Falso	
	SourceField:	PPA	
	SourceTable:	Clasificacion_club	
	TextAlign:	General	
PMA			Entero largo 4
	AggregateType:	-1	
	AllowZeroLength:	Falso	
	AppendOnly:	Falso	
	Attributes:	Tamaño fijo	
	CollatingOrder:	3082	
	ColumnHidden:	Falso	
	ColumnOrder:	13	
	ColumnWidth:	Predeterminado	
	DataUpdatable:	Falso	
	DecimalPlaces:	Automático	
	DisplayControl:	Cuadro de texto	
	GUID:	{guid {BF66C26A-4EC3-4143-8A25-761726828AE6}}	
	OrdinalPosition:	12	
	Required:	Falso	
	SourceField:	PMA	
	SourceTable:	Clasificacion_club	
	TextAlign:	General	
SANC			Entero largo 4
	AggregateType:	-1	
	AllowZeroLength:	Falso	
	AppendOnly:	Falso	
	Attributes:	Tamaño fijo	

E:\Escritorio\Billgest 20100323\Billar66.accdb
 Tabla: Clasificacion_club

domingo, 11 de abril de 2010
 Página: 29

CollatingOrder: 3082
 ColumnHidden: Falso
 ColumnOrder: 14
 ColumnWidth: Predeterminado
 DataUpdatable: Falso
 DecimalPlaces: Automático
 DisplayControl: Cuadro de texto
 GUID: {guid {E337FCBB-D97B-4902-850D-F9482AE192A0}}
 OrdinalPosition: 13
 Required: Falso
 SourceField: SANC
 SourceTable: Clasificacion_club
 TextAlign: General

PUNT	AggregateType: -1 AllowZeroLength: Falso AppendOnly: Falso Attributes: Tamaño fijo CollatingOrder: 3082 ColumnHidden: Falso ColumnOrder: Predeterminado ColumnWidth: Predeterminado DataUpdatable: Falso DecimalPlaces: Automático DisplayControl: Cuadro de texto GUID: {guid {65387BD8-1A97-4B53-B2F7-446AD3F12912}} OrdinalPosition: 14 Required: Falso SourceField: PUNT SourceTable: Clasificacion_club TextAlign: General	Entero largo	4
------	---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	--------------	---

Relaciones

ClubClasificacion_club

Club	Clasificacion_club
NOMBRE	1 ∞ EQUIPO
Attributes:	Forzado; Actualizaciones en cascada; Eliminations en cascada
RelationshipType:	Uno a varios

E:\Escritorio\Billgest 20100323\Billar66.accdb
 Tabla: Clasificacion_club

domingo, 11 de abril de 2010
 Página: 30

TemporadaClasificacion_club

Temporada	Clasificacion_club
Temporada	TEMPORADA
Attributes:	No forzado
RelationshipType:	Uno a varios

Índices de tabla

Nombre	Número de campos
Clasificacion_clubTemporada	1
Clustered:	Falso
DistinctCount:	31
Foreign:	Falso
IgnoreNulls:	Falso
Name:	Clasificacion_clubTemporada
Primary:	Falso
Required:	Falso
Unique:	Falso
Campos:	
TEMPORADA	Ascendente
ClubClasificacion_club	1
Clustered:	Falso
DistinctCount:	200
Foreign:	Verdadero
IgnoreNulls:	Falso
Name:	ClubClasificacion_club
Primary:	Falso
Required:	Falso
Unique:	Falso
Campos:	
EQUIPO	Ascendente
PrimaryKey	3
Clustered:	Falso
DistinctCount:	4343
Foreign:	Falso
IgnoreNulls:	Falso
Name:	PrimaryKey
Primary:	Verdadero
Required:	Verdadero
Unique:	Verdadero

E:\Escritorio\Billgest 20100323\Billar66.accdb
 Tabla: Clasificacion_club

domingo, 11 de abril de 2010
 Página: 31

Campos:	
EQUIPO	Ascendente
TEMPORADA	Ascendente
CODLIGA	Ascendente

Permisos de usuario

admin	Eliminar; Leer permisos; Establecer permisos; Cambiar propietario, Leer definición; Escribir definición; Leer datos; Insertar datos; Actualizar datos; Eliminar datos
-------	-----------------------------------------------------------------------------------------------------------------------------------------------------------------------

Permisos de grupo

Admins	Eliminar; Leer permisos; Establecer permisos; Cambiar propietario, Leer definición; Escribir definición; Leer datos; Insertar datos; Actualizar datos; Eliminar datos
Users	Eliminar; Leer permisos; Establecer permisos; Cambiar propietario, Leer definición; Escribir definición; Leer datos; Insertar datos; Actualizar datos; Eliminar datos

E:\Escritorio\Billgest 20100323\Billar66.accdb
 Tabla: Clasificacion_jugador

domingo, 11 de abril de 2010
 Página: 32

Propiedades

DateCreated:	23/11/2008 2:55:44	DefaultView:	2
DisplayViewsOnSharePointSite			1
GUID:	{guid {DBC3A27F-C192-4A27-9551-BB8BCB3B2A34}}	HideNewField:	Falso
LastUpdated:	27/12/2008 2:57:00	NameMap:	Datos binarios largos
OrderBy:	[Clasificacion_jugador].[CODLI GA]	OrderByOn:	Verdadero
OrderByOnLoad:	Verdadero	Orientation:	De izquierda a derecha
RecordCount:	10503	TotalsRow:	Falso
Updatable:	Verdadero		

Columnas

Nombre	Tipo	Tamaño
CODIGO	Entero largo	4
AggregateType:	-1	
AllowZeroLength:	Falso	
AppendOnly:	Falso	
Attributes:	Tamaño fijo; Autoincremento	
CollatingOrder:	3082	
ColumnHidden:	Falso	
ColumnOrder:	1	
ColumnWidth:	Predeterminado	
DataUpdatable:	Falso	
OrdinalPosition:	0	
Required:	Falso	
SourceField:	CODIGO	
SourceTable:	Clasificacion_jugador	
TextAlign:	General	
NOMBRE	Texto	255
AggregateType:	-1	
AllowZeroLength:	Falso	
AppendOnly:	Falso	
Attributes:	Longitud variable	
CollatingOrder:	3082	
ColumnHidden:	Falso	
ColumnOrder:	Predeterminado	
ColumnWidth:	Predeterminado	
DataUpdatable:	Falso	
DisplayControl:	Cuadro de texto	
GUID:	{guid {20E06BEA-C9D8-41C9-96EF-086D61A8CEB3}}	
IMEMode:	0	
IMESentenceMode:	3	
OrdinalPosition:	1	
Required:	Verdadero	
SourceField:	NOMBRE	
SourceTable:	Clasificacion_jugador	
TextAlign:	General	
UnicodeCompression:	Falso	
TEMPORADA	Texto	255

E:\Escritorio\Billgest 20100323\Billar66.accdb
 Tabla: Clasificacion_jugador

domingo, 11 de abril de 2010
 Página: 34

	Required:	Falso	
	SourceField:	PJ	
	SourceTable:	Clasificacion_jugador	
	TextAlign:	General	
PG			Entero largo 4
	AggregateType:	-1	
	AllowZeroLength:	Falso	
	AppendOnly:	Falso	
	Attributes:	Tamaño fijo	
	CollatingOrder:	3082	
	ColumnHidden:	Falso	
	ColumnOrder:	Predeterminado	
	ColumnWidth:	Predeterminado	
	DataUpdatable:	Falso	
	DecimalPlaces:	Automático	
	DisplayControl:	Cuadro de texto	
	GUID:	{guid {18CD582E-F679-4D8A-B580-07CC2AA70F2B}}	
	OrdinalPosition:	5	
	Required:	Falso	
	SourceField:	PG	
	SourceTable:	Clasificacion_jugador	
	TextAlign:	General	
PE			Entero largo 4
	AggregateType:	-1	
	AllowZeroLength:	Falso	
	AppendOnly:	Falso	
	Attributes:	Tamaño fijo	
	CollatingOrder:	3082	
	ColumnHidden:	Falso	
	ColumnOrder:	Predeterminado	
	ColumnWidth:	Predeterminado	
	DataUpdatable:	Falso	
	DecimalPlaces:	Automático	
	DisplayControl:	Cuadro de texto	
	GUID:	{guid {D97955A2-53A7-43A0-9007-10CB6425A876}}	
	OrdinalPosition:	6	
	Required:	Falso	
	SourceField:	PE	
	SourceTable:	Clasificacion_jugador	
	TextAlign:	General	
PP			Entero largo 4
	AggregateType:	-1	
	AllowZeroLength:	Falso	
	AppendOnly:	Falso	
	Attributes:	Tamaño fijo	
	CollatingOrder:	3082	
	ColumnHidden:	Falso	
	ColumnOrder:	Predeterminado	
	ColumnWidth:	Predeterminado	
	DataUpdatable:	Falso	
	DecimalPlaces:	Automático	

E:\Escritorio\Billgest 20100323\Billar66.accdb
 Tabla: Clasificacion_jugador

domingo, 11 de abril de 2010
 Página: 35

	DisplayControl: Cuadro de texto GUID: {guid {A2228219-456C-40AE-827A-22CB2CB220DB}} OrdinalPosition: 7 Required: Falso SourceField: PP SourceTable: Clasificacion_jugador TextAlign: General		
P		Entero largo	4
	AggregateType: -1 AllowZeroLength: Falso AppendOnly: Falso Attributes: Tamaño fijo CollatingOrder: 3082 ColumnHidden: Falso ColumnOrder: Predeterminado ColumnWidth: Predeterminado DataUpdatable: Falso DecimalPlaces: Automático DisplayControl: Cuadro de texto GUID: {guid {B00AE271-11B9-485A-9D77-91A1255A4F04}} OrdinalPosition: 8 Required: Falso SourceField: P SourceTable: Clasificacion_jugador TextAlign: General		
SM		Entero largo	4
	AggregateType: -1 AllowZeroLength: Falso AppendOnly: Falso Attributes: Tamaño fijo CollatingOrder: 3082 ColumnHidden: Falso ColumnOrder: Predeterminado ColumnWidth: Predeterminado DataUpdatable: Falso DecimalPlaces: Automático DisplayControl: Cuadro de texto GUID: {guid {CA5D0CB6-D7B8-44ED-9236-493FCFAE38E1}} OrdinalPosition: 9 Required: Falso SourceField: SM SourceTable: Clasificacion_jugador TextAlign: General		
CAR		Entero largo	4
	AggregateType: -1 AllowZeroLength: Falso AppendOnly: Falso Attributes: Tamaño fijo CollatingOrder: 3082 ColumnHidden: Falso ColumnOrder: Predeterminado		

E:\Escritorio\Billgest 20100323\Billar66.accdb
 Tabla: Clasificacion_jugador

domingo, 11 de abril de 2010
 Página: 36

	ColumnWidth:	Predeterminado	
	DataUpdatable:	Falso	
	DecimalPlaces:	Automático	
	DisplayControl:	Cuadro de texto	
	GUID:	{guid {A27FF2F6-EF3E-42D1-B419-DA2D14BF1505}}	
	OrdinalPosition:	10	
	Required:	Falso	
	SourceField:	CAR	
	SourceTable:	Clasificacion_jugador	
	TextAlign:	General	
ENT			Entero largo 4
	AggregateType:	-1	
	AllowZeroLength:	Falso	
	AppendOnly:	Falso	
	Attributes:	Tamaño fijo	
	CollatingOrder:	3082	
	ColumnHidden:	Falso	
	ColumnOrder:	Predeterminado	
	ColumnWidth:	Predeterminado	
	DataUpdatable:	Falso	
	DecimalPlaces:	Automático	
	DisplayControl:	Cuadro de texto	
	GUID:	{guid {AE41FAD7-8B53-494E-B4C6-1C8BB5947DD8}}	
	OrdinalPosition:	11	
	Required:	Falso	
	SourceField:	ENT	
	SourceTable:	Clasificacion_jugador	
	TextAlign:	General	
PROM			Doble 8
	AggregateType:	-1	
	AllowZeroLength:	Falso	
	AppendOnly:	Falso	
	Attributes:	Tamaño fijo	
	CollatingOrder:	3082	
	ColumnHidden:	Falso	
	ColumnOrder:	Predeterminado	
	ColumnWidth:	Predeterminado	
	DataUpdatable:	Falso	
	DecimalPlaces:	Automático	
	DisplayControl:	Cuadro de texto	
	GUID:	{guid {02ED3575-049C-4E31-9E6C-4940DAB1AAF8}}	
	OrdinalPosition:	12	
	Required:	Falso	
	SourceField:	PROM	
	SourceTable:	Clasificacion_jugador	
	TextAlign:	General	

Relaciones

E:\Escritorio\Billgest 20100323\Billar66.accdb
 Tabla: Clasificacion_jugador

domingo, 11 de abril de 2010
 Página: 37

TemporadaClasificacion_jugador

Temporada	Clasificacion_jugador
Temporada	1 ∞ TEMPORADA
Attributes:	Forzado; Actualizaciones en cascada; Eliminaciones en cascada
RelationshipType:	Uno a varios

Índices de tabla

Nombre	Número de campos
PrimaryKey	3
Clustered:	Falso
DistinctCount:	65265
Foreign:	Falso
IgnoreNulls:	Falso
Name:	PrimaryKey
Primary:	Verdadero
Required:	Verdadero
Unique:	Verdadero
Campos:	
NOMBRE	Ascendente
TEMPORADA	Ascendente
CODLIGA	Ascendente

TemporadaClasificacion_jugador	1
Clustered:	Falso
DistinctCount:	22
Foreign:	Verdadero
IgnoreNulls:	Falso
Name:	TemporadaClasificacion_jugador
Primary:	Falso
Required:	Falso
Unique:	Falso
Campos:	
TEMPORADA	Ascendente

Permisos de usuario

admin	Eliminar; Leer permisos; Establecer permisos; Cambiar propietario, Leer definición; Escribir definición; Leer datos; Insertar datos; Actualizar datos; Eliminar datos
-------	-----------------------------------------------------------------------------------------------------------------------------------------------------------------------

E:\Escritorio\Billgest 20100323\Billar66.accdb
Tabla: Clasificacion_jugador

domingo, 11 de abril de 2010
Página: 38

Permisos de grupo

Admins	Eliminar; Leer permisos; Establecer permisos; Cambiar propietario, Leer definición; Escribir definición; Leer datos; Insertar datos; Actualizar datos; Eliminar datos
Users	Eliminar; Leer permisos; Establecer permisos; Cambiar propietario, Leer definición; Escribir definición; Leer datos; Insertar datos; Actualizar datos; Eliminar datos

E:\Escritorio\Billgest 20100323\Billar66.accdb
 Tabla: Club

domingo, 11 de abril de 2010
 Página: 39

Propiedades

DateCreated:	06/11/2008 2:05:58	DefaultView:	2
DisplayViewsOnSharePointSite		1	FilterOnLoad: Falso
GUID:	{guid {E9364693-B096-4E47-8408-ED875E02F044}}	HideNewField:	Falso
LastUpdated:	24/08/2009 23:04:14	NameMap:	Datos binarios largos
OrderByOn:	Falso	OrderByOnLoad:	Verdadero
Orientation:	De izquierda a derecha	RecordCount:	38
TotalsRow:	Falso	Updatable:	Verdadero

Columnas

Nombre	Tipo	Tamaño
CODIGO	Texto	3
AggregateType: -1 AllowZeroLength: Falso AppendOnly: Falso Attributes: Longitud variable CollatingOrder: 3082 ColumnHidden: Falso ColumnOrder: Predeterminado ColumnWidth: Predeterminado DataUpdatable: Falso DisplayControl: Cuadro de texto GUID: {guid {56FC6A81-50CE-4013-AD07-D0BE304FC278}} IMEMode: 0 IMESentenceMode: 3 OrdinalPosition: 0 Required: Verdadero SourceField: CODIGO SourceTable: Club TextAlign: General UnicodeCompression: Falso		
NOMBRE	Texto	38
AggregateType: -1 AllowZeroLength: Falso AppendOnly: Falso Attributes: Longitud variable CollatingOrder: 3082 ColumnHidden: Falso ColumnOrder: Predeterminado ColumnWidth: Predeterminado DataUpdatable: Falso DisplayControl: Cuadro de texto GUID: {guid {96864BCF-AF60-4989-98BD-0BABFD50D025}} IMEMode: 0 IMESentenceMode: 3 OrdinalPosition: 1 Required: Verdadero SourceField: NOMBRE SourceTable: Club		

E:\Escritorio\Billgest 20100323\Billar66.accdb
 Tabla: Club

domingo, 11 de abril de 2010
 Página: 40

TextAlign:	General		
UnicodeCompression:	Falso		
DIRECCION		Texto	100
AggregateType:	-1		
AllowZeroLength:	Verdadero		
AppendOnly:	Falso		
Attributes:	Longitud variable		
CollatingOrder:	3082		
ColumnHidden:	Falso		
ColumnOrder:	Predeterminado		
ColumnWidth:	Predeterminado		
DataUpdatable:	Falso		
DisplayControl:	Cuadro de texto		
IMEMode:	0		
IMESentenceMode:	3		
OrdinalPosition:	2		
Required:	Falso		
SourceField:	DIRECCION		
SourceTable:	Club		
TextAlign:	General		
UnicodeCompression:	Falso		
COD_POST		Texto	5
AggregateType:	-1		
AllowZeroLength:	Verdadero		
AppendOnly:	Falso		
Attributes:	Longitud variable		
CollatingOrder:	3082		
ColumnHidden:	Falso		
ColumnOrder:	Predeterminado		
ColumnWidth:	Predeterminado		
DataUpdatable:	Falso		
DisplayControl:	Cuadro de texto		
GUID:	{guid {1E0FCB8B-C1FC-4298-AECD-04447718589A}}		
IMEMode:	0		
IMESentenceMode:	3		
OrdinalPosition:	3		
Required:	Falso		
SourceField:	COD_POST		
SourceTable:	Club		
TextAlign:	General		
UnicodeCompression:	Falso		
POBLACION		Texto	30
AggregateType:	-1		
AllowZeroLength:	Verdadero		
AppendOnly:	Falso		
Attributes:	Longitud variable		
CollatingOrder:	3082		
ColumnHidden:	Falso		
ColumnOrder:	Predeterminado		
ColumnWidth:	Predeterminado		
DataUpdatable:	Falso		

E:\Escritorio\Billgest 20100323\Billar66.accdb
 Tabla: Club

domingo, 11 de abril de 2010
 Página: 41

DisplayControl: Cuadro de texto
 GUID: {guid {8AA60F56-719F-44DD-A360-DEF2E6FD48DF}}
 IMEMode: 0
 IMESentenceMode: 3
 OrdinalPosition: 4
 Required: Falso
 SourceField: POBLACION
 SourceTable: Club
 TextAlign: General
 UnicodeCompression: Falso

TELEFONO Texto 10

AggregateType: -1
 AllowZeroLength: Verdadero
 AppendOnly: Falso
 Attributes: Longitud variable
 CollatingOrder: 3082
 ColumnHidden: Falso
 ColumnOrder: Predeterminado
 ColumnWidth: Predeterminado
 DataUpdatable: Falso
 DisplayControl: Cuadro de texto
 GUID: {guid {565E39CF-05A9-469F-834A-2C32AC36495C}}
 IMEMode: 0
 IMESentenceMode: 3
 OrdinalPosition: 5
 Required: Falso
 SourceField: TELEFONO
 SourceTable: Club
 TextAlign: General
 UnicodeCompression: Verdadero

EMAIL Texto 40

AggregateType: -1
 AllowZeroLength: Verdadero
 AppendOnly: Falso
 Attributes: Longitud variable
 CollatingOrder: 3082
 ColumnHidden: Falso
 ColumnOrder: Predeterminado
 ColumnWidth: Predeterminado
 DataUpdatable: Falso
 DisplayControl: Cuadro de texto
 GUID: {guid {90D52BEB-FD55-4FD9-86A5-9FA6BB34CE85}}
 IMEMode: 0
 IMESentenceMode: 3
 OrdinalPosition: 6
 Required: Falso
 SourceField: EMAIL
 SourceTable: Club
 TextAlign: General
 UnicodeCompression: Verdadero

PAGINA_WEB Texto 150

E:\Escritorio\Billgest 20100323\Billar66.accdb
 Tabla: Club

domingo, 11 de abril de 2010
 Página: 42

AggregateType: -1
 AllowZeroLength: Verdadero
 AppendOnly: Falso
 Attributes: Longitud variable
 CollatingOrder: 3082
 ColumnHidden: Falso
 ColumnOrder: Predeterminado
 ColumnWidth: 1500
 DataUpdatable: Falso
 DisplayControl: Cuadro de texto
 GUID: {guid {8361F2AF-525C-4844-ADE3-D9B9A0010963}}
 IMEMode: 0
 IMESentenceMode: 3
 OrdinalPosition: 7
 Required: Falso
 SourceField: PAGINA_WEB
 SourceTable: Club
 TextAlign: General
 UnicodeCompression: Verdadero

PROVINCIA Texto 20

AggregateType: -1
 AllowZeroLength: Verdadero
 AppendOnly: Falso
 Attributes: Longitud variable
 CollatingOrder: 3082
 ColumnHidden: Falso
 ColumnOrder: Predeterminado
 ColumnWidth: Predeterminado
 DataUpdatable: Falso
 DisplayControl: Cuadro de texto
 IMEMode: 0
 IMESentenceMode: 3
 OrdinalPosition: 8
 Required: Falso
 SourceField: PROVINCIA
 SourceTable: Club
 TextAlign: General
 UnicodeCompression: Falso

CIF Texto 15

AggregateType: -1
 AllowZeroLength: Verdadero
 AppendOnly: Falso
 Attributes: Longitud variable
 CollatingOrder: 3082
 ColumnHidden: Falso
 ColumnOrder: Predeterminado
 ColumnWidth: Predeterminado
 DataUpdatable: Falso
 DisplayControl: Cuadro de texto
 IMEMode: 0
 IMESentenceMode: 3
 OrdinalPosition: 9

E:\Escritorio\Billgest 20100323\Billar66.accdb
 Tabla: Club

domingo, 11 de abril de 2010
 Página: 43

Required: Falso
 SourceField: CIF
 SourceTable: Club
 TextAlign: General
 UnicodeCompression: Verdadero

CO_NOMBRE Texto 50

AggregateType: -1
 AllowZeroLength: Verdadero
 AppendOnly: Falso
 Attributes: Longitud variable
 CollatingOrder: 3082
 ColumnHidden: Falso
 ColumnOrder: Predeterminado
 ColumnWidth: Predeterminado
 DataUpdatable: Falso
 DisplayControl: Cuadro de texto
 IMEMode: 0
 IMESentenceMode: 3
 OrdinalPosition: 10
 Required: Falso
 SourceField: CO_NOMBRE
 SourceTable: Club
 TextAlign: General
 UnicodeCompression: Verdadero

CO_DIRECCION Texto 100

AggregateType: -1
 AllowZeroLength: Verdadero
 AppendOnly: Falso
 Attributes: Longitud variable
 CollatingOrder: 3082
 ColumnHidden: Falso
 ColumnOrder: Predeterminado
 ColumnWidth: Predeterminado
 DataUpdatable: Falso
 DisplayControl: Cuadro de texto
 IMEMode: 0
 IMESentenceMode: 3
 OrdinalPosition: 11
 Required: Falso
 SourceField: CO_DIRECCION
 SourceTable: Club
 TextAlign: General
 UnicodeCompression: Verdadero

CO_COD_POST Texto 5

AggregateType: -1
 AllowZeroLength: Verdadero
 AppendOnly: Falso
 Attributes: Longitud variable
 CollatingOrder: 3082
 ColumnHidden: Falso
 ColumnOrder: Predeterminado

E:\Escritorio\Billgest 20100323\Billar66.accdb
 Tabla: Club

domingo, 11 de abril de 2010
 Página: 45

AllowZeroLength:	Verdadero
AppendOnly:	Falso
Attributes:	Longitud variable
CollatingOrder:	3082
ColumnHidden:	Falso
ColumnOrder:	Predeterminado
ColumnWidth:	Predeterminado
DataUpdatable:	Falso
DisplayControl:	Cuadro de texto
IMEMode:	0
IMESentenceMode:	3
OrdinalPosition:	15
Required:	Falso
SourceField:	FAX
SourceTable:	Club
TextAlign:	General
UnicodeCompression:	Verdadero

Relaciones

ClubActa

Club	Acta
NOMBRE	1 ∞ CLUB1
Attributes:	Forzado; Actualizaciones en cascada; Eliminationes en cascada
RelationshipType:	Uno a varios

ClubActa

Club	Acta
NOMBRE	1 ∞ CLUB2
Attributes:	Forzado; Actualizaciones en cascada; Eliminationes en cascada
RelationshipType:	Uno a varios

E:\Escritorio\Billgest 20100323\Billar66.accdb
 Tabla: Club

domingo, 11 de abril de 2010
 Página: 46

ClubClasificacion_club

Club	Clasificacion_club
NOMBRE	1 ∞ EQUIPO
Attributes:	Forzado; Actualizaciones en cascada; Eliminationes en cascada
RelationshipType:	Uno a varios

Índices de tabla

Nombre	Número de campos
CODIGO	1
Clustered:	Falso
DistinctCount:	34
Foreign:	Falso
IgnoreNulls:	Falso
Name:	CODIGO
Primary:	Falso
Required:	Falso
Unique:	Verdadero
Campos:	
CODIGO	Ascendente

PrimaryKey	1
Clustered:	Falso
DistinctCount:	34
Foreign:	Falso
IgnoreNulls:	Falso
Name:	PrimaryKey
Primary:	Verdadero
Required:	Verdadero
Unique:	Verdadero
Campos:	
NOMBRE	Ascendente

Permisos de usuario

admin	Eliminar; Leer permisos; Establecer permisos; Cambiar propietario, Leer definición; Escribir definición; Leer datos; Insertar datos; Actualizar datos; Eliminar datos
-------	-----------------------------------------------------------------------------------------------------------------------------------------------------------------------

E:\Escritorio\Billgest 20100323\Billar66.accdb
Tabla: Club

domingo, 11 de abril de 2010
Página: 47

Permisos de grupo

Admins	Eliminar; Leer permisos; Establecer permisos; Cambiar propietario, Leer definición; Escribir definición; Leer datos; Insertar datos; Actualizar datos; Eliminar datos
Users	Eliminar; Leer permisos; Establecer permisos; Cambiar propietario, Leer definición; Escribir definición; Leer datos; Insertar datos; Actualizar datos; Eliminar datos

E:\Escritorio\Billgest 20100323\Billar66.accdb
 Tabla: Precio

domingo, 11 de abril de 2010
 Página: 48

Propiedades

DateCreated:	28/07/2009 23:24:39	DefaultView:	2
DisplayViewsOnSharePointSite			1
GUID:	{guid {77498A9F-AB6B-45BC-9633-9BE1536F04D8}}	HideNewField:	Falso
LastUpdated:	29/07/2009 0:41:50	NameMap:	Datos binarios largos
OrderByOn:	Falso	OrderByOnLoad:	Verdadero
Orientation:	De izquierda a derecha	RecordCount:	1
TotalsRow:	Falso	Updatable:	Verdadero

Columnas

Nombre	Tipo	Tamaño
Id	Entero largo	4
AggregateType: -1 AllowZeroLength: Falso AppendOnly: Falso Attributes: Tamaño fijo; Autoincremento CollatingOrder: 3082 ColumnHidden: Falso ColumnOrder: Predeterminado ColumnWidth: Predeterminado DataUpdatable: Falso OrdinalPosition: 0 Required: Falso SourceField: Id SourceTable: Precio TextAlign: General		
Junior	Simple	4
AggregateType: -1 AllowZeroLength: Falso AppendOnly: Falso Attributes: Tamaño fijo CollatingOrder: 3082 ColumnHidden: Falso ColumnOrder: Predeterminado ColumnWidth: Predeterminado DataUpdatable: Falso DecimalPlaces: 2 DisplayControl: Cuadro de texto Format: General Number OrdinalPosition: 1 Required: Verdadero SourceField: Junior SourceTable: Precio TextAlign: General		
Senior	Simple	4
AggregateType: -1 AllowZeroLength: Falso AppendOnly: Falso		

E:\Escritorio\Billgest 20100323\Billar66.accdb
 Tabla: Precio

domingo, 11 de abril de 2010
 Página: 49

Attributes:	Tamaño fijo
CollatingOrder:	3082
ColumnHidden:	Falso
ColumnOrder:	Predeterminado
ColumnWidth:	Predeterminado
DataUpdatable:	Falso
DecimalPlaces:	2
DisplayControl:	Cuadro de texto
Format:	General Number
OrdinalPosition:	2
Required:	Verdadero
SourceField:	Senior
SourceTable:	Precio
TextAlign:	General

Índices de tabla

Nombre	Número de campos
--------	------------------

Id	1
Clustered:	Falso
DistinctCount:	1
Foreign:	Falso
IgnoreNulls:	Falso
Name:	Id
Primary:	Falso
Required:	Falso
Unique:	Falso
Campos:	
Id	Ascendente

PrimaryKey	1
Clustered:	Falso
DistinctCount:	1
Foreign:	Falso
IgnoreNulls:	Falso
Name:	PrimaryKey
Primary:	Verdadero
Required:	Verdadero
Unique:	Verdadero
Campos:	
Id	Ascendente

E:\Escritorio\Billgest 20100323\Billar66.accdb
Tabla: Precio

domingo, 11 de abril de 2010
Página: 50

Permisos de usuario

admin	Eliminar; Leer permisos; Establecer permisos; Cambiar propietario, Leer definición; Escribir definición; Leer datos; Insertar datos; Actualizar datos; Eliminar datos
-------	-----------------------------------------------------------------------------------------------------------------------------------------------------------------------

Permisos de grupo

Admins	Eliminar; Leer permisos; Establecer permisos; Cambiar propietario, Leer definición; Escribir definición; Leer datos; Insertar datos; Actualizar datos; Eliminar datos
Users	Eliminar; Leer permisos; Establecer permisos; Cambiar propietario, Leer definición; Escribir definición; Leer datos; Insertar datos; Actualizar datos; Eliminar datos

E:\Escritorio\Billgest 20100323\Billar66.accdb
 Tabla: Socio

domingo, 11 de abril de 2010
 Página: 51

Propiedades

DateCreated:	06/11/2008 2:04:30	DefaultView:	2
DisplayViewsOnSharePointSite		1	FilterOnLoad: Falso
GUID:	{guid {030D1AB0-6401-45A9-A0EB-539465A708B9}}	HideNewField:	Falso
LastUpdated:	25/08/2009 23:43:07	NameMap:	Datos binarios largos
OrderByOn:	Falso	OrderByOnLoad:	Verdadero
Orientation:	De izquierda a derecha	RecordCount:	639
TotalsRow:	Falso	Updatable:	Verdadero

Columnas

Nombre	Tipo	Tamaño
LICENCIA	Texto	6
AggregateType:	-1	
AllowZeroLength:	Verdadero	
AppendOnly:	Falso	
Attributes:	Longitud variable	
CollatingOrder:	3082	
ColumnHidden:	Falso	
ColumnOrder:	Predeterminado	
ColumnWidth:	Predeterminado	
DataUpdatable:	Falso	
DisplayControl:	Cuadro de texto	
GUID:	{guid {1DA7564B-EB0B-4EF1-8BA7-6596F34940C4}}	
IMEMode:	0	
IMESentenceMode:	3	
OrdinalPosition:	0	
Required:	Verdadero	
SourceField:	LICENCIA	
SourceTable:	Socio	
TextAlign:	General	
UnicodeCompression:	Falso	
F_NACI	Fecha/Hora	8
AggregateType:	-1	
AllowZeroLength:	Falso	
AppendOnly:	Falso	
Attributes:	Tamaño fijo	
CollatingOrder:	3082	
ColumnHidden:	Falso	
ColumnOrder:	Predeterminado	
ColumnWidth:	Predeterminado	
DataUpdatable:	Falso	
GUID:	{guid {5E99362C-A532-4627-A68B-DEA67CE40057}}	
IMEMode:	0	
IMESentenceMode:	3	
OrdinalPosition:	1	
Required:	Falso	
ShowDatePicker:	Para fechas	
SourceField:	F_NACI	
SourceTable:	Socio	

E:\Escritorio\Billgest 20100323\Billar66.accdb
 Tabla: Socio

domingo, 11 de abril de 2010
 Página: 52

	TextAlign:	General		
APELLIDOS			Texto	35
	AggregateType:	-1		
	AllowZeroLength:	Verdadero		
	AppendOnly:	Falso		
	Attributes:	Longitud variable		
	CollatingOrder:	3082		
	ColumnHidden:	Falso		
	ColumnOrder:	Predeterminado		
	ColumnWidth:	3528		
	DataUpdatable:	Falso		
	DisplayControl:	Cuadro de texto		
	GUID:	{guid {C294E867-39AB-48BC-BE10-3C2397233863}}		
	IMEMode:	0		
	IMESentenceMode:	3		
	OrdinalPosition:	2		
	Required:	Falso		
	SourceField:	APELLIDOS		
	SourceTable:	Socio		
	TextAlign:	General		
	UnicodeCompression:	Falso		
NOM_BATA			Texto	55
	AggregateType:	-1		
	AllowZeroLength:	Verdadero		
	AppendOnly:	Falso		
	Attributes:	Longitud variable		
	CollatingOrder:	3082		
	ColumnHidden:	Falso		
	ColumnOrder:	Predeterminado		
	ColumnWidth:	Predeterminado		
	DataUpdatable:	Falso		
	DisplayControl:	Cuadro de texto		
	IMEMode:	0		
	IMESentenceMode:	3		
	OrdinalPosition:	3		
	Required:	Falso		
	SourceField:	NOM_BATA		
	SourceTable:	Socio		
	TextAlign:	General		
	UnicodeCompression:	Falso		
NOMBRE			Texto	20
	AggregateType:	-1		
	AllowZeroLength:	Verdadero		
	AppendOnly:	Falso		
	Attributes:	Longitud variable		
	CollatingOrder:	3082		
	ColumnHidden:	Falso		
	ColumnOrder:	Predeterminado		
	ColumnWidth:	Predeterminado		
	DataUpdatable:	Falso		
	DisplayControl:	Cuadro de texto		

E:\Escritorio\Billgest 20100323\Billar66.accdb
 Tabla: Socio

domingo, 11 de abril de 2010
 Página: 53

GUID: {guid {0D31ADCC-DE05-45F0-858E-68668904469F}}
 IMEMode: 0
 IMESentenceMode: 3
 OrdinalPosition: 4
 Required: Falso
 SourceField: NOMBRE
 SourceTable: Socio
 TextAlign: General
 UnicodeCompression: Falso

COD_CLUB Texto 3

AggregateType: -1
 AllowZeroLength: Verdadero
 AppendOnly: Falso
 Attributes: Longitud variable
 CollatingOrder: 3082
 ColumnHidden: Falso
 ColumnOrder: Predeterminado
 ColumnWidth: Predeterminado
 DataUpdatable: Falso
 DisplayControl: Cuadro de texto
 GUID: {guid {0CEAE19B-D378-4E67-B237-70F7CEBE6490}}
 IMEMode: 0
 IMESentenceMode: 3
 OrdinalPosition: 5
 Required: Falso
 SourceField: COD_CLUB
 SourceTable: Socio
 TextAlign: General
 UnicodeCompression: Falso

DNI Texto 10

AggregateType: -1
 AllowZeroLength: Verdadero
 AppendOnly: Falso
 Attributes: Longitud variable
 CollatingOrder: 3082
 ColumnHidden: Falso
 ColumnOrder: Predeterminado
 ColumnWidth: Predeterminado
 DataUpdatable: Falso
 DisplayControl: Cuadro de texto
 GUID: {guid {D3033BA5-DDE7-4C6C-9E83-16C3CC7E1767}}
 IMEMode: 0
 IMESentenceMode: 3
 OrdinalPosition: 6
 Required: Falso
 SourceField: DNI
 SourceTable: Socio
 TextAlign: General
 UnicodeCompression: Falso

DIRECCION Texto 50

AggregateType: -1

E:\Escritorio\Billgest 20100323\Billar66.accdb
 Tabla: Socio

domingo, 11 de abril de 2010
 Página: 54

AllowZeroLength: Verdadero
 AppendOnly: Falso
 Attributes: Longitud variable
 CollatingOrder: 3082
 ColumnHidden: Falso
 ColumnOrder: Predeterminado
 ColumnWidth: Predeterminado
 DataUpdatable: Falso
 DisplayControl: Cuadro de texto
 GUID: {guid {C4182A47-78AE-4C5D-8C01-64945493F3DD}}
 IMEMode: 0
 IMESentenceMode: 3
 OrdinalPosition: 7
 Required: Falso
 SourceField: DIRECCION
 SourceTable: Socio
 TextAlign: General
 UnicodeCompression: Falso

POBLACION Texto 27

AggregateType: -1
 AllowZeroLength: Verdadero
 AppendOnly: Falso
 Attributes: Longitud variable
 CollatingOrder: 3082
 ColumnHidden: Falso
 ColumnOrder: Predeterminado
 ColumnWidth: Predeterminado
 DataUpdatable: Falso
 DisplayControl: Cuadro de texto
 GUID: {guid {F389B3AE-6A8B-4CC9-AF38-16A501079E11}}
 IMEMode: 0
 IMESentenceMode: 3
 OrdinalPosition: 8
 Required: Falso
 SourceField: POBLACION
 SourceTable: Socio
 TextAlign: General
 UnicodeCompression: Falso

COD_POST Texto 5

AggregateType: -1
 AllowZeroLength: Verdadero
 AppendOnly: Falso
 Attributes: Longitud variable
 CollatingOrder: 3082
 ColumnHidden: Falso
 ColumnOrder: Predeterminado
 ColumnWidth: Predeterminado
 DataUpdatable: Falso
 DisplayControl: Cuadro de texto
 GUID: {guid {18BE9148-1E39-4BAD-B6A0-4E3EDD20512D}}
 IMEMode: 0
 IMESentenceMode: 3

E:\Escritorio\Billgest 20100323\Billar66.accdb
 Tabla: Socio

domingo, 11 de abril de 2010
 Página: 55

OrdinalPosition: 9
 Required: Falso
 SourceField: COD_POST
 SourceTable: Socio
 TextAlign: General
 UnicodeCompression: Falso

PROVINCIA Texto 11

AggregateType: -1
 AllowZeroLength: Verdadero
 AppendOnly: Falso
 Attributes: Longitud variable
 CollatingOrder: 3082
 ColumnHidden: Falso
 ColumnOrder: Predeterminado
 ColumnWidth: Predeterminado
 DataUpdatable: Falso
 DisplayControl: Cuadro de texto
 GUID: {guid {DDC751B7-D956-4D36-AE0B-1E2F094870D6}}
 IMEMode: 0
 IMESentenceMode: 3
 OrdinalPosition: 10
 Required: Falso
 SourceField: PROVINCIA
 SourceTable: Socio
 TextAlign: General
 UnicodeCompression: Falso

TELEFONO Texto 10

AggregateType: -1
 AllowZeroLength: Verdadero
 AppendOnly: Falso
 Attributes: Longitud variable
 CollatingOrder: 3082
 ColumnHidden: Falso
 ColumnOrder: Predeterminado
 ColumnWidth: Predeterminado
 DataUpdatable: Falso
 DisplayControl: Cuadro de texto
 GUID: {guid {D744EF57-AE02-441A-BF5A-745E71672E94}}
 IMEMode: 0
 IMESentenceMode: 3
 OrdinalPosition: 11
 Required: Falso
 SourceField: TELEFONO
 SourceTable: Socio
 TextAlign: General
 UnicodeCompression: Falso

EMAIL Texto 40

AggregateType: -1
 AllowZeroLength: Verdadero
 AppendOnly: Falso
 Attributes: Longitud variable

E:\Escritorio\Billgest 20100323\Billar66.accdb
 Tabla: Socio

domingo, 11 de abril de 2010
 Página: 56

CollatingOrder: 3082
 ColumnHidden: Falso
 ColumnOrder: Predeterminado
 ColumnWidth: 1788
 DataUpdatable: Falso
 DisplayControl: Cuadro de texto
 GUID: {guid {8BD6385E-E88F-4786-A30F-88D32D20724D}}
 IMEMode: 0
 IMESentenceMode: 3
 OrdinalPosition: 12
 Required: Falso
 SourceField: EMAIL
 SourceTable: Socio
 TextAlign: General
 UnicodeCompression: Falso

NACIONAL Texto 1

AggregateType: -1
 AllowZeroLength: Verdadero
 AppendOnly: Falso
 Attributes: Longitud variable
 CollatingOrder: 3082
 ColumnHidden: Falso
 ColumnOrder: Predeterminado
 ColumnWidth: Predeterminado
 DataUpdatable: Falso
 DisplayControl: Cuadro de texto
 GUID: {guid {6513893A-9BD2-4065-A1AC-FE1B6F888110}}
 IMEMode: 0
 IMESentenceMode: 3
 OrdinalPosition: 13
 Required: Falso
 SourceField: NACIONAL
 SourceTable: Socio
 TextAlign: General
 UnicodeCompression: Falso

ENVIGOR Sí/No 1

AggregateType: -1
 AllowZeroLength: Falso
 AppendOnly: Falso
 Attributes: Tamaño fijo
 CollatingOrder: 3082
 ColumnHidden: Falso
 ColumnOrder: Predeterminado
 ColumnWidth: Predeterminado
 DataUpdatable: Falso
 DisplayControl: 106
 Format: Yes/No
 OrdinalPosition: 14
 Required: Falso
 SourceField: ENVIGOR
 SourceTable: Socio
 TextAlign: General

E:\Escritorio\Billgest 20100323\Billar66.accdb
 Tabla: Socio

domingo, 11 de abril de 2010
 Página: 57

<p>ORDEN</p> <p>AggregateType: -1 AllowZeroLength: Falso AppendOnly: Falso Attributes: Tamaño fijo CollatingOrder: 3082 ColumnHidden: Falso ColumnOrder: Predeterminado ColumnWidth: Predeterminado DataUpdatable: Falso DecimalPlaces: Automático DisplayControl: Cuadro de texto OrdinalPosition: 15 Required: Falso SourceField: ORDEN SourceTable: Socio TextAlign: General</p>	<p>Entero largo</p>	<p>4</p>
<p>PRECIO</p> <p>AggregateType: -1 AllowZeroLength: Verdadero AppendOnly: Falso Attributes: Longitud variable CollatingOrder: 3082 ColumnHidden: Falso ColumnOrder: Predeterminado ColumnWidth: Predeterminado DataUpdatable: Falso DisplayControl: Cuadro de texto IMEMode: 0 IMESentenceMode: 3 OrdinalPosition: 16 Required: Falso SourceField: PRECIO SourceTable: Socio TextAlign: General UnicodeCompression: Verdadero</p>	<p>Texto</p>	<p>20</p>
<p>CATEGORIA</p> <p>AggregateType: -1 AllowZeroLength: Verdadero AppendOnly: Falso Attributes: Longitud variable CollatingOrder: 3082 ColumnHidden: Falso ColumnOrder: Predeterminado ColumnWidth: Predeterminado DataUpdatable: Falso DisplayControl: Cuadro de texto IMEMode: 0 IMESentenceMode: 3 OrdinalPosition: 17 Required: Falso SourceField: CATEGORIA</p>	<p>Texto</p>	<p>10</p>

E:\Escritorio\Billgest 20100323\Billar66.accdb
 Tabla: Socio

domingo, 11 de abril de 2010
 Página: 58

	SourceTable:	Socio	
	TextAlign:	General	
	UnicodeCompression:	Verdadero	
CLUB			Texto 255
	AggregateType:	-1	
	AllowZeroLength:	Verdadero	
	AppendOnly:	Falso	
	Attributes:	Longitud variable	
	CollatingOrder:	3082	
	ColumnHidden:	Falso	
	ColumnOrder:	Predeterminado	
	ColumnWidth:	Predeterminado	
	DataUpdatable:	Falso	
	DisplayControl:	Cuadro de texto	
	IMEMode:	0	
	IMESentenceMode:	3	
	OrdinalPosition:	18	
	Required:	Falso	
	SourceField:	CLUB	
	SourceTable:	Socio	
	TextAlign:	General	
	UnicodeCompression:	Verdadero	

Relaciones

SocioActa

	Socio		Acta
	LICENCIA	1 ∞	LICENCIA12
Attributes:	Forzado; Actualizaciones en cascada; Eliminationes en cascada		
RelationshipType:	Uno a varios		

SocioActa

	Socio		Acta
	LICENCIA	1 ∞	LICENCIA21
Attributes:	Forzado; Actualizaciones en cascada; Eliminationes en cascada		
RelationshipType:	Uno a varios		

SocioActa

Socio	1	∞	Acta
LICENCIA			LICENCIA22
Attributes:		Forzado; Actualizaciones en cascada; Eliminations en cascada	
RelationshipType:		Uno a varios	

SocioActa

Socio	1	∞	Acta
LICENCIA			LICENCIA31
Attributes:		Forzado; Actualizaciones en cascada; Eliminations en cascada	
RelationshipType:		Uno a varios	

SocioActa

Socio	1	∞	Acta
LICENCIA			LICENCIA32
Attributes:		Forzado; Actualizaciones en cascada; Eliminations en cascada	
RelationshipType:		Uno a varios	

SocioActa

Socio	1	∞	Acta
LICENCIA			LICENCIA41
Attributes:		Forzado; Actualizaciones en cascada; Eliminations en cascada	
RelationshipType:		Uno a varios	

SocioActa

Socio	1	∞	Acta
LICENCIA			LICENCIA42
Attributes:		Forzado; Actualizaciones en cascada; Eliminations en cascada	
RelationshipType:		Uno a varios	

E:\Escritorio\Billgest 20100323\Billar66.accdb
 Tabla: Socio

domingo, 11 de abril de 2010
 Página: 60

Índices de tabla

Nombre	Número de campos
PrimaryKey	1
Clustered:	Falso
DistinctCount:	647
Foreign:	Falso
IgnoreNulls:	Falso
Name:	PrimaryKey
Primary:	Verdadero
Required:	Verdadero
Unique:	Verdadero
Campos:	
LICENCIA	Ascendente

SocioLICENCIA	1
Clustered:	Falso
DistinctCount:	647
Foreign:	Falso
IgnoreNulls:	Falso
Name:	SocioLICENCIA
Primary:	Falso
Required:	Falso
Unique:	Falso
Campos:	
LICENCIA	Ascendente

SocioNOMBRE	1
Clustered:	Falso
DistinctCount:	209
Foreign:	Falso
IgnoreNulls:	Falso
Name:	SocioNOMBRE
Primary:	Falso
Required:	Falso
Unique:	Falso
Campos:	
NOMBRE	Ascendente

Permisos de usuario

admin	Eliminar; Leer permisos; Establecer permisos; Cambiar propietario, Leer definición; Escribir definición; Leer datos; Insertar datos; Actualizar datos; Eliminar datos
-------	-----------------------------------------------------------------------------------------------------------------------------------------------------------------------

E:\Escritorio\Billgest 20100323\Billar66.accdb
Tabla: Socio

domingo, 11 de abril de 2010
Página: 61

Permisos de grupo

Admins	Eliminar; Leer permisos; Establecer permisos; Cambiar propietario, Leer definición; Escribir definición; Leer datos; Insertar datos; Actualizar datos; Eliminar datos
Users	Eliminar; Leer permisos; Establecer permisos; Cambiar propietario, Leer definición; Escribir definición; Leer datos; Insertar datos; Actualizar datos; Eliminar datos

E:\Escritorio\Billgest 20100323\Billar66.accdb
 Tabla: Temporada

domingo, 11 de abril de 2010
 Página: 62

Propiedades

DateCreated:	25/11/2008 11:50:09	DefaultView:	2
DisplayViewsOnSharePointSite			1
GUID:	{guid {BDADBDE4-6C73-48FF-B1EB-9C6074B8DCC1}}	HideNewField:	Falso
LastUpdated:	28/07/2009 23:27:27	NameMap:	Datos binarios largos
OrderByOn:	Falso	OrderByOnLoad:	Verdadero
Orientation:	De izquierda a derecha	RecordCount:	15
TotalsRow:	Falso	Updatable:	Verdadero

Columnas

Nombre	Tipo	Tamaño
Id	Entero largo	4
AggregateType: -1 AllowZeroLength: Falso AppendOnly: Falso Attributes: Tamaño fijo; Autoincremento CollatingOrder: 3082 ColumnHidden: Falso ColumnOrder: Predeterminado ColumnWidth: Predeterminado DataUpdatable: Falso GUID: {guid {D22D5EAF-3145-4A7F-85FB-EEA5661C90F8}} OrdinalPosition: 0 Required: Falso SourceField: Id SourceTable: Temporada TextAlign: General		
Temporada	Texto	255
AggregateType: -1 AllowZeroLength: Verdadero AppendOnly: Falso Attributes: Longitud variable CollatingOrder: 3082 ColumnHidden: Falso ColumnOrder: Predeterminado ColumnWidth: Predeterminado DataUpdatable: Falso DisplayControl: Cuadro de texto GUID: {guid {B7CB3E30-7701-4BDC-8176-15DA09513235}} IMEMode: 0 IMESentenceMode: 3 OrdinalPosition: 1 Required: Falso SourceField: Temporada SourceTable: Temporada TextAlign: General UnicodeCompression: Verdadero		
Actual	Entero largo	4

E:\Escritorio\Billgest 20100323\Billar66.accdb
 Tabla: Temporada

domingo, 11 de abril de 2010
 Página: 63

AggregateType: -1
 AllowZeroLength: Falso
 AppendOnly: Falso
 Attributes: Tamaño fijo
 CollatingOrder: 3082
 ColumnHidden: Falso
 ColumnOrder: Predeterminado
 ColumnWidth: Predeterminado
 DataUpdatable: Falso
 DecimalPlaces: Automático
 DisplayControl: Cuadro de texto
 OrdinalPosition: 2
 Required: Falso
 SourceField: Actual
 SourceTable: Temporada
 TextAlign: General

Disponible

Entero largo

4

AggregateType: -1
 AllowZeroLength: Falso
 AppendOnly: Falso
 Attributes: Tamaño fijo
 CollatingOrder: 3082
 ColumnHidden: Falso
 ColumnOrder: Predeterminado
 ColumnWidth: Predeterminado
 DataUpdatable: Falso
 DecimalPlaces: Automático
 DisplayControl: Cuadro de texto
 OrdinalPosition: 3
 Required: Falso
 SourceField: Disponible
 SourceTable: Temporada
 TextAlign: General

Mayor

Texto

255

AggregateType: -1
 AllowZeroLength: Verdadero
 AppendOnly: Falso
 Attributes: Longitud variable
 CollatingOrder: 3082
 ColumnHidden: Falso
 ColumnOrder: Predeterminado
 ColumnWidth: Predeterminado
 DataUpdatable: Falso
 DisplayControl: Cuadro de texto
 IMEMode: 0
 IMESentenceMode: 3
 OrdinalPosition: 4
 Required: Falso
 SourceField: Mayor
 SourceTable: Temporada
 TextAlign: General
 UnicodeCompression: Verdadero

Relaciones

TemporadaActa

Temporada	Acta
Temporada	TEMPORADA
Attributes:	No forzado
RelationshipType:	Uno a varios

TemporadaClasificacion_club

Temporada	Clasificacion_club
Temporada	TEMPORADA
Attributes:	No forzado
RelationshipType:	Uno a varios

TemporadaClasificacion_jugador

Temporada	Clasificacion_jugador
Temporada	1 ∞ TEMPORADA
Attributes:	Forzado; Actualizaciones en cascada; Eliminations en cascada
RelationshipType:	Uno a varios

Índices de tabla

Nombre	Número de campos
PrimaryKey	1
Clustered:	Falso
DistinctCount:	48
Foreign:	Falso
IgnoreNulls:	Falso
Name:	PrimaryKey
Primary:	Verdadero
Required:	Verdadero
Unique:	Verdadero

E:\Escritorio\Billgest 20100323\Billar66.accdb
Tabla: Temporada

domingo, 11 de abril de 2010
Página: 65

Campos:
Temporada Ascendente

Permisos de usuario

admin	Eliminar; Leer permisos; Establecer permisos; Cambiar propietario, Leer definición; Escribir definición; Leer datos; Insertar datos; Actualizar datos; Eliminar datos
-------	-----------------------------------------------------------------------------------------------------------------------------------------------------------------------

Permisos de grupo

Admins	Eliminar; Leer permisos; Establecer permisos; Cambiar propietario, Leer definición; Escribir definición; Leer datos; Insertar datos; Actualizar datos; Eliminar datos
--------	-----------------------------------------------------------------------------------------------------------------------------------------------------------------------

Users	Eliminar; Leer permisos; Establecer permisos; Cambiar propietario, Leer definición; Escribir definición; Leer datos; Insertar datos; Actualizar datos; Eliminar datos
-------	-----------------------------------------------------------------------------------------------------------------------------------------------------------------------

E:\Escritorio\Billgest 20100323\Billar66.accdb
 Base de datos: E:\Escritorio\Billgest 20100323\Billar66.accdb

domingo, 11 de abril de 2010
 Página: 66

Propiedades

AccessVersion:	09.50	AllowBuiltInToolbars:	Verdadero
AllowDatasheetSchema:	Verdadero	AllowFullMenus:	Verdadero
AllowShortcutMenus:	Verdadero	AllowSpecialKeys:	Verdadero
AllowToolbarChanges:	Verdadero	ANSI Query Mode:	0
Auto Compact:	0	Build:	423
CheckTruncatedNumFields:	1	CollatingOrder:	3082
HasOfflineLists:	70	NavPane Category:	1
NavPane Closed:	0	NavPane Sort By:	1
NavPane View By:	0	NavPane Width:	215
Picture Property Storage:	0	ProjVer:	87
QueryTimeout:	60	RecordsAffected:	0
Show Values in Indexed:	1	Show Values in Non-Indexed:	1
Show Values in Remote:	0	Show Values Limit:	1000
ShowDocumentTabs:	Verdadero	StartupShowDBWindow:	Verdadero
StartupShowStatusBar:	Verdadero	Themed Form Controls:	1
Transactions:	Verdadero	Updatable:	Verdadero
UseAppIconForFrmRpt:	Falso	UseMDIMode:	0
Version:	12.0		

Permisos de usuario

admin	Eliminar; Leer permisos; Establecer permisos; Cambiar propietario, Abrir o ejecutar; Abrir en modo exclusivo
-------	--------------------------------------------------------------------------------------------------------------

Permisos de grupo

Admins	Eliminar; Leer permisos; Establecer permisos; Cambiar propietario, Abrir o ejecutar; Abrir en modo exclusivo
Users	Eliminar; Leer permisos; Establecer permisos; Cambiar propietario, Abrir o ejecutar; Abrir en modo exclusivo

Relaciones

ClubActa

Club	Acta
NOMBRE	1 ∞ CLUB1
Attributes:	Forzado; Actualizaciones en cascada; Eliminations en cascada
RelationshipType:	Uno a varios

ClubActa

Club	Acta
NOMBRE	1 ∞ CLUB2
Attributes:	Forzado; Actualizaciones en cascada; Eliminations en cascada
RelationshipType:	Uno a varios

ClubClasificacion_club

Club	Clasificacion_club
NOMBRE	1 ∞ EQUIPO
Attributes:	Forzado; Actualizaciones en cascada; Eliminations en cascada
RelationshipType:	Uno a varios

SocioActa

Socio	Acta
LICENCIA	1 ∞ LICENCIA12
Attributes:	Forzado; Actualizaciones en cascada; Eliminations en cascada
RelationshipType:	Uno a varios

SocioActa

Socio	1	∞	Acta
LICENCIA			LICENCIA21
Attributes:		Forzado; Actualizaciones en cascada; Eliminations en cascada	
RelationshipType:		Uno a varios	

SocioActa

Socio	1	∞	Acta
LICENCIA			LICENCIA22
Attributes:		Forzado; Actualizaciones en cascada; Eliminations en cascada	
RelationshipType:		Uno a varios	

SocioActa

Socio	1	∞	Acta
LICENCIA			LICENCIA31
Attributes:		Forzado; Actualizaciones en cascada; Eliminations en cascada	
RelationshipType:		Uno a varios	

SocioActa

Socio	1	∞	Acta
LICENCIA			LICENCIA32
Attributes:		Forzado; Actualizaciones en cascada; Eliminations en cascada	
RelationshipType:		Uno a varios	

SocioActa

Socio	1	∞	Acta
LICENCIA			LICENCIA41
Attributes:		Forzado; Actualizaciones en cascada; Eliminations en cascada	
RelationshipType:		Uno a varios	

SocioActa

Socio	1	∞	Acta
LICENCIA			LICENCIA42
Attributes:	Forzado; Actualizaciones en cascada; Eliminationes en cascada		
RelationshipType:	Uno a varios		

TemporadaActa

Temporada	Acta
Temporada	TEMPORADA
Attributes:	No forzado
RelationshipType:	Uno a varios

TemporadaClasificacion_club

Temporada	Clasificacion_club
Temporada	TEMPORADA
Attributes:	No forzado
RelationshipType:	Uno a varios

TemporadaClasificacion_jugador

Temporada	1	∞	Clasificacion_jugador
Temporada			TEMPORADA
Attributes:	Forzado; Actualizaciones en cascada; Eliminationes en cascada		
RelationshipType:	Uno a varios		

	CICLO DE PRUEBAS	
Pruebas Unitarias Interfaz Socio	Billargest	06/06/2010

Socio

RESUMEN DE LAS PRUEBAS

	Descripción	Retorno	Resultado
Paso 1	DataGridView	OK	Funciona correctamente.
Paso 2	Interfaz y estándares	OK	Funciona correctamente.
Paso3	Ortografía		
Paso 4	E/S		
Paso 5	Rendimiento		

(Deben documentarse resultados correctos e incorrectos de todas las posibles opciones de lanzamiento de la rutina ó códigos de acción).

CONFIGURACIÓN DEL PROBADOR

- No se utiliza ningún probador específico.

PRECONDICIONES

- No existe ninguna precondición específica.

CICLO DE PRUEBAS		
Pruebas Unitarias Interfaz Socio	Billargest	06/06/2010

DETALLE DE LAS PRUEBAS

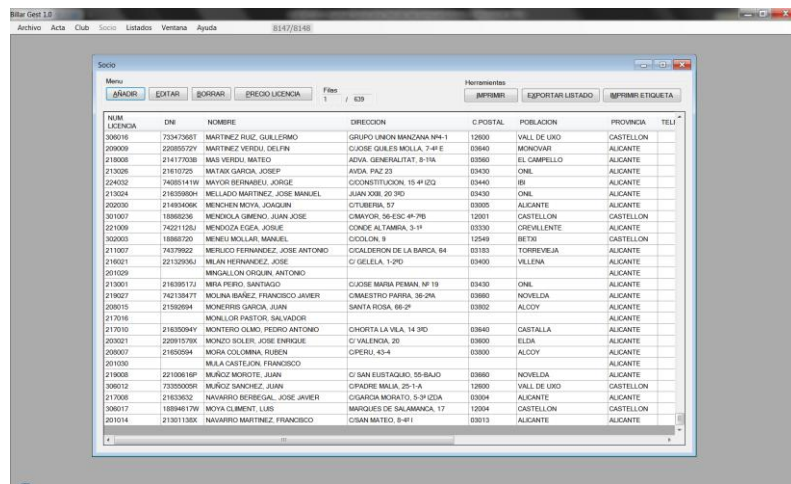
1. DATAGRIDVIEW

Se comprueba que el formato del DataGridView que se representa en la pantalla sea el correcto acorde a los estándares establecidos. Su navegación debe ser completa. El enlace a datos debe funcionar correctamente.

Entrada: Registros tabla Socio BB.DD.

LICENCIA	F_NACI	APELLIDOS	NOM_BATA	NOMBRE	COD_CLUB	DNI	DIRECCION	POBLACION	COD_POST
101001	13/05/1962	PASCUAL CORDIER		PASCUAL CORD JORGE	101	22548804H	CL. PADRE FERF VALENCIA	VALENCIA	46002
101005	02/10/1958	JARQUE PEREZ		JARQUE PEREZ, CARLOS ANGEL	101	22636803L	AV. 9 DE OCTUBRE QUART DE POB	VALENCIA	46930
101006	13/09/1962	GARCIA CIURANA		GARCIA CIURAN CARLOS ADRIAN	101	22539013W	AV. MAESTRO F VALENCIA	VALENCIA	46015
101007	17/03/1943	GARCIA LLORIS		GARCIA LLORIS, JOSE VICENTE	101	19488075Z	CL. PINTOR RIB EL VEDAT DE TC	VALENCIA	46901
101008	01/10/1962	LLONGO SORIANO		LLONGO SORIA, RAMON FRANC	101	24328295E	CL. CIUTAT DE VALENCIA	VALENCIA	46015
101009	11/12/1940	ROSELLO BELLVER		ROSELLO BELLV LUIS	101	19360935H	CL. JUAN DE CE VALENCIA	VALENCIA	46017
101010	25/11/1967	RUBIO VILA		RUBIO VILA, DC DOMINGO	101	22558629E	CL. MOLINA, 2- VALENCIA	VALENCIA	46006
101013	26/03/1973	LOPEZ BONET		LOPEZ BONET, I BENJAMIN	101	52689259P	CL. XATIVA, 1-P VALENCIA	VALENCIA	46002
101014	31/12/1964	GARCIA CATALAN		GARCIA CATAL MANUEL	101	18422591M	AV. PEREZ GALE VALENCIA	VALENCIA	46018
101015	28/03/1929	GIJARRO NEIRA		GIJARRO NEIRA EMILIO	101	19120876X	AV. TIRSO DE N VALENCIA	VALENCIA	46009
101016	25/03/1936	SERRA RODAS		SERRA RODAS, I EMILIO	101	19617205E	CL. JUAN RAMC VALENCIA	VALENCIA	46006
101017	25/04/1948	ROMERO IBAÑEZ		ROMERO IBAÑE EMILIO	101	19442120I	CL. MONTE LLA XIRIVELLA	VALENCIA	46950
101018	11/11/1965	HERRAIZ MARCHAL		HERRAIZ MARC CARLOS	101	19899548V	CL. ROMEU DE VALENCIA	VALENCIA	46006
101019	15/05/1948	GONZALEZ HEREDIA		GONZALEZ HER FRANCISCO	101	19862854P	CL. SAGUNTO, VALENCIA	VALENCIA	46009
101020	14/09/1958	GINER ESTIVALIS		GINER ESTIVALI JOSE RAMON	101	22635960G	CL. CUENCA, 12 VALENCIA	VALENCIA	46007
101021	22/07/1945	HONRUBIA FERNANDEZ		HONRUBIA FER PADRO	101	74468985E	CL. SOLIANA, 2 VALENCIA	VALENCIA	46013
101022	13/01/1963	JIMENEZ GARCIA		JIMENEZ GARCIA ANGEL	101	22688104F	PL. RAFAEL ATT MANISES	VALENCIA	46940
101023	25/11/1949	MONTALT BADIA		MONTALT BADI GABRIEL	101	19444598F	CL. ANTONIO SI FOIOS	VALENCIA	46134
101024	06/04/1950	BIOSCA CARBONELL		BIOSCA CARBO RAFAEL	101	19865566A	CL. ALCOY, 9-5 VALENCIA	VALENCIA	46004
101025	23/09/1974	RUIZ ROCH		RUIZ ROCH, EU EUGENIO	101	06250700I	CL. SOCUELLAN TOMELLOSO	VALENCIA	46014
101028	27/10/1942	MONTORO SANCHEZ		MONTORO SAN VICENTE	101	19389906F	PL. SAINETERO VALENCIA	VALENCIA	46014
101032	06/08/1952	MESA ALCAÑIZ		MESA ALCAÑIZ, JUAN JOSE	101	22612779F	CL. MIGUEL HE XIRIVELLA	VALENCIA	46950
101033	01/05/1957	MORENO VICENTE		MORENO VICET CRISTOBAL	101	22631825D	CL. ALMOINES, VALENCIA	VALENCIA	46007
101034	31/12/1972	MACIAN GALLEGO		MACIAN GALLE JOAQUIN	101	52686790T	CL. SAN VICENT MANISES	VALENCIA	46940
101035	12/01/1968	BOTELLA BRANT		BOTELLA BRAN RAFAEL	101	52725436V	CL. PUERTO RIC VALENCIA	VALENCIA	46006
102003	13/01/1956	REQUENA VITALES		REQUENA VITA JOSE FERNAND	102	19465235	CL. ARNICHES, VALENCIA	VALENCIA	46183
102006	28/10/1958	MARTI SOLAZ		MARTI SOLAZ, J JOAQUIN	102	22529560	CL. CORONEL N VALENCIA	VALENCIA	46019
102009	04/05/1940	GUTIERREZ ALCARAZ		GUTIERREZ ALC ANGEL	102	22844632	CL. RAMON Y C/ GODELLA	VALENCIA	46110
102010	24/12/1940	MUÑOZ HERRERO		MUÑOZ HERRE JESUS FRANCIS	102	19865190K	CL. AYELO DE N VALENCIA	VALENCIA	46026

Salida: DataGridView correctamente representado y con su formato correcto.



	CICLO DE PRUEBAS	
Pruebas Unitarias Interfaz Socio	Billargest	06/06/2010

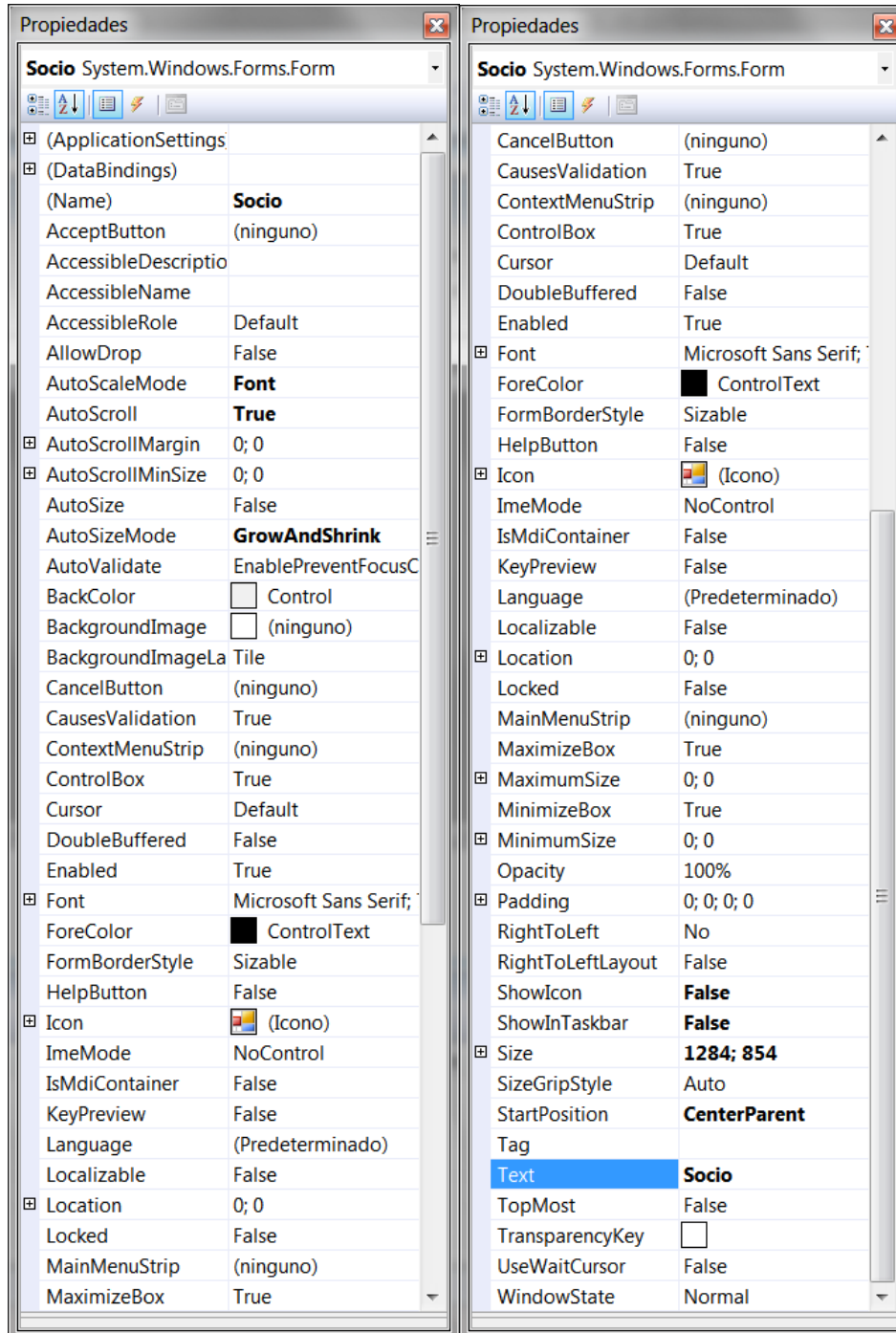
2. Interfaz y estándares

Se comprueba que la interfaz cumple con los estándares especificados en el conjunto de normas de la aplicación.

Entrada:

Salida: Diálogo socio correctamente representado.

<h1>CICLO DE PRUEBAS</h1>		
Pruebas Unitarias Interfaz Socio	Billargest	06/06/2010



	CICLO DE PRUEBAS	
Pruebas Unitarias Interfaz Socio	Billargest	06/06/2010

Autor de la prueba: **Carlos Soriano Lorente**
 Fecha de la prueba: **11/04/2010**
 Estado pruebas (PR-En progreso, CO- Completado) : PR
 Autor de la revisión: **yyyyyyyyy**
 Fecha de la revisión: **yy/yy/yyyy**
 Estado Revisión: (PR-En progreso, CO- Completado) :

(Archivar en C:\Billargest\Pruebas\XXX, siendo XXX la unidad de negocio a la que pertenece el elemento a probar)

Anexo VII. Índice:

1. Condiciones administrativas	2
1.1. Introducción	2
1.2. Objeto del contrato	2
1.3. Obligaciones	2
1.4. Garantía	3
1.5. Lugar de realización.....	4
1.6. Presupuesto.....	4
1.7. Forma de pago	4
1.8. Plazo de ejecución.....	4
1.9. Licencias	4
1.10. Penalizaciones	5
1.11. Entregables.....	5
2. Condiciones técnicas	6
2.1. Objetivo	6
2.2. Descripción funcional	6
2.3. Condiciones de realización	6
2.4. Equipo de trabajo	7
2.5. Requisitos	7
3. Seguridad y confidencialidad	9

Pliego de condiciones

1. Condiciones administrativas

1.1. Introducción

Sistema de Información para la Gestión de Ligas de Billar, en adelante BILLARGEST, es un programa de gestión administrativa que tiene como objetivo el seguimiento de las ligas anuales de Billar, el control de socios y el control de clubs adscritos a la Real Federación Valenciana de Billar de la Comunidad Valenciana, en adelante FBCV.

El presente documento de carácter comprensivo y obligatorio contiene las condiciones o cláusulas que se aceptan por contratante y contratista en el contrato de implantación del sistema BILLARGEST en la FBCV.

1.2. Objeto del contrato

El Objeto del presente documento es establecer los requerimientos básicos para la puesta en marcha del sistema BILLARGEST en el departamento de administración de la FBCV, destinado a cubrir las carencias del sistema que ya se encuentra en producción actualmente.

1.3. Obligaciones

Tanto el contratante, la FBCV, como el contratista, D. Carlos Soriano Lorente, se comprometen a ejecutar el contrato en las condiciones previstas en el presente documento.

Corresponde al contratante:

- Supervisar y validar la realización y el desarrollo de los trabajos.
- Reconocer los derechos de propiedad intelectual del adjudicatario sobre la aplicación contemplada en este contrato. El adjudicatario posee en exclusiva los derechos de explotación que pudieran derivarse de los trabajos realizados por un tiempo ilimitado.
- Dar conformidad a los resultados finales de los trabajos realizados.
- Emitir las certificaciones parciales de recepción de los mismos.
- Facilitar la aportación del equipo, personal o medios necesarios para la correcta implantación del sistema en la entidad.
- Velar por el cumplimiento y el nivel de calidad de los trabajos de desarrollo exigidos.

Corresponde a contratista o adjudicatario:

- Facilitar a las personas designadas por la FBCV la información y documentación que soliciten para disponer de un pleno conocimiento del desarrollo de los trabajos, así como de los eventuales problemas que puedan plantearse y de las tecnologías, métodos y herramientas utilizados para resolverlos.
- La presentación de un programa de trabajo.
- El contratista debe disponer de la capacidad técnica precisa para cubrir las obligaciones que se deriven del contrato.
- No proporcionar dato alguno de los trabajos contratados, ni publicar, total o parcialmente, el contenido de los mismos. En todo caso, el contratista será responsable de los daños y perjuicios que se deriven del incumplimiento de estas obligaciones, siendo además dicho incumplimiento causa de resolución del contrato.
- Responsabilizarse de la calidad técnica de los trabajos que desarrolle y de las prestaciones que realice, así como de las consecuencias que produzcan omisiones, errores, métodos inadecuados o conclusiones incorrectas en la ejecución del contrato.
- Informar a la FBCV de cuáles son las autorizaciones, licencias o cesiones para que ésta pueda conocer las limitaciones de su uso.
- Ejecutar el contrato en las condiciones previstas en el presente documento y en la oferta técnica presentada por él.

El desconocimiento del contrato en cualquiera de sus términos no eximirá de la obligación de su cumplimiento.

1.4. Garantía

Una vez finalizada la implantación del sistema el plazo de garantía ofertado por el licitador será de un año en todo lo referido al mantenimiento correctivo de los desarrollos realizados. En este plazo el adjudicatario se compromete a corregir con la mayor brevedad los posibles errores y problemas que surjan por el uso del software implantado.

Para la corrección de errores, una vez la notificación ha sido atendida por el contratista, se iniciarán las medidas correctivas y solo se detendrán una vez la incidencia ha sido resuelta, siempre que el contratante no exija dar prioridad a otra incidencia.

Asimismo, durante el primer mes tras la puesta en marcha, las peticiones de mejora y modificaciones correrán a cargo del adjudicatario.

Fuera de los plazos establecidos, las incidencias y peticiones remitidas al adjudicatario serán evaluadas y tratadas con nuevas cláusulas que no se establecen en el presente contrato.

1.5. Lugar de realización

Se determina para la fase de implantación que el lugar sea la sede central de la FBCV en la calle Avellanas, 14-2º, 46003 Valencia. Todo coste generado por desplazamiento correrá a cargo del contratista.

1.6. Presupuesto

El importe máximo de la presente contratación asciende a la cantidad de tres mil novecientos noventa y cinco euros, (3.995€), IVA y todo tipo de impuesto incluidos.

1.7. Forma de pago

Se le abonará al contratista los precios convenidos de la siguiente forma:

- 80 % del precio de adjudicación tras la implantación y a la firma de la aceptación del sistema.
- 20 % del precio de la adjudicación a la finalización del primer mes tras la implantación.

El abono del trabajo se realizará mediante transferencia bancaria a la cuenta especificada por el adjudicatario.

1.8. Plazo de ejecución

El proyecto se llevará a cabo desde el día indicado en el documento de formalización del contrato y su plazo de realización será de siete meses.

Se entregará en el contrato un cronograma pormenorizado en función de las fases del proyecto, con detalle de los trabajos a realizar en cada una de ellas.

1.9. Licencias

Se establece que todos los derechos de propiedad intelectual del presente proyecto corresponden en exclusiva al contratista. Dicha titularidad atribuye por tanto plena disposición y derecho exclusivo de explotación de los trabajos realizados al mismo, por tanto podrá hacer uso en otros proyectos de los desarrollos, documentación y trabajos realizados bajo el presente contrato.

El contratante no podrá difundir o aprovechar para cualquier medio de información, comunicación pública o reproducción, en cualquier lugar del mundo y sin límite temporal, los resultados o suministros de este Contrato.

Se establece además que el número de licencias del programa a implantar será de DOS unidades y por tanto podrá ser instalado en dos maquinas cualesquiera con nivel de arquitectura establecido en el apartado 2.4 de este mismo pliego. Estas licencias no tienen caducidad alguna en el tiempo y no se permite su posterior reventa ni utilización fuera de los límites de las condiciones establecidas en el documento contractual.

1.10. Penalizaciones

La FBCV podrá imponer penalizaciones por retraso, en la entrega de los trabajos objeto del contrato. La penalización aplicable con carácter general es del 5% del importe total del contrato por cada semana de retraso, a partir de la semana siguiente a la finalización del plazo de entrega.

Aquellos tiempos de respuesta de resolución que superen, para cada una de las incidencias tratadas, los tiempos comprometidos por el adjudicatario, serán susceptibles de aplicación de penalización, si dicho retraso es enteramente achacable al adjudicatario del presente expediente. En caso reiterativo, se podrá plantear la rescisión del contrato por incumplimiento continuado de los compromisos establecidos por contrato.

El importe de las penalizaciones se hará efectivo mediante deducción de las mismas en los pagos con fecha establecida en el siguiente pliego.

1.11. Entregables

El adjudicatario del contrato deberá comprometerse a la entrega de los siguientes documentos a lo largo de la ejecución del proyecto:

- Documento de requisitos.
- Plan de actividades.
- Plan de pruebas.
- Manual de usuario.

2. Condiciones técnicas

2.1. Objetivo

Constituye el objeto del presente pliego de condiciones técnicas para la contratación del proyecto de implantación del sistema de información para la gestión de ligas de billar BILLARGEST para la Federación de Billar de la Comunidad Valenciana (FBCV).

Este documento abarcará los aspectos técnicos y organizativos necesarios para asegurar la máxima calidad del proyecto presentado.

2.2. Descripción funcional

Las condiciones de los trabajos comprendidos en el presente pliego de condiciones técnicas, consiste en la realización del proyecto BILLARGEST que engloba las siguientes unidades de negocio:

- Gestión de Temporadas.
 - Gestión de Actas.
- Gestión de Clubes.
- Gestión de Socios.

Todos los módulos especificados además contendrán a su vez las funciones generales:

- Listados.
- Impresión.
- Exportación a Excel (excepto actas).

En el **ANEXO II - 3.Análisis del sistema de información** se encuentra una descripción detallada de los mismos.

2.3. Condiciones de realización

El adjudicatario presentará un plan de actividades en el cual se especificarán los horarios de prestación de los servicios, calendarios de disponibilidad del técnico, periodos de desarrollo, implantación y pruebas y de resolución de incidencias, así como el resto de información de servicio necesaria para la correcta colaboración entre los dos titulares del contrato.

El contratista se compromete a utilizar la metodología y pautas de trabajo especificadas en los anexos de requisitos entregados para garantizar la calidad de los servicios ofrecidos.

Además, habrá de incluirse la correspondiente formación del personal seleccionado por la administración de la FBCV, que se encargará de la posterior administración del sistema.

La aceptación del producto se realizará de manera formal a través de un documento que lo acredite una vez todas las pruebas de aceptación del producto queden concluidas y sean

satisfactorias para la FBCV. De la misma manera se realizará al finalizar el periodo de formación. Los detalles del proceso de formación se encuentran en el **ANEXO II - 5. Construcción del Sistema de Información.**

2.4. Equipo de trabajo

El contratante designará a un solo director técnico de proyecto que considere oportuno, el cual con relación a la prestación de los servicios objeto del presente contrato tendrá las siguientes funciones:

- Velar por el cumplimiento de las prestaciones exigidas y ofrecidas.
- Realizar las certificaciones parciales de servicios prestados.
- Otras actuaciones:
 - Informar al adjudicatario de cualquier deficiencia que observe en algún componente lógico, facilitando a la vez toda la información disponible sobre la incidencia.
 - Adoptar las medidas que fueren precisas, dentro de lo posible, con el fin de facilitar la determinación de los fallos y sus causas.
 - Adoptar las medidas que fuesen precisas para la utilización del sistema de acuerdo con las normas de empleo del adjudicatario.

Respecto al técnico responsable del proyecto por parte del adjudicatario, se precisa que tenga experiencia los siguientes conocimientos:

- Desarrollo de aplicaciones en entorno Microsoft® Visual Studio 2008 con Microsoft® Visual C# 3.5.
- Elaboración de Bases de Datos relacionales con formato Microsoft® Access.
- Construcción de diagramas y modelado.
- Experiencia en informática ofimática para la redacción de la documentación asociada.
- Capacidad de comunicación para la formación in situ a usuarios.

2.5. Requisitos

Requisitos técnicos específicos:

- Arquitectura

La arquitectura donde se implantará el sistema será una arquitectura de un solo nivel de acuerdo a las exigencias de la FBCV. Esta arquitectura corresponde a los sistemas en los que tanto el remitente de una solicitud como el receptor de la solicitud enviada por

cliente se encuentran en una misma máquina física. Esta configuración no podrá ser alterada en ningún caso.

- Herramienta de desarrollo y programación
 - Microsoft® Visual Studio 2008 Service Pack 1 (SP1)
 - Microsoft® .NET Framework 3.5 SP1
 - Instalador: InstallShield® Premium 2010
 - Generación de reportes: Crystal Report® .NET
- Herramientas de gestión de Base de Datos:
 - Microsoft® Access 2007
- Herramientas de documentación:
 - Microsoft® Word 2007
- Herramientas de gestión:
 - Microsoft® Project 2003

Requisitos técnicos generales:

Definición de los servicios que son necesarios para certificar la calidad de los productos:

- Funcionalidad: deberá asegurar la adecuación y la exactitud de las aplicaciones, así como interoperabilidad y seguridad de acceso.
- Mantenibilidad: se deberá asegurar que la aplicación es mantenible, es decir, con la documentación deberá poder ser analizada, cambiada y probada. Además se deberá asegurar que la aplicación sea estable.
- Eficiencia: se deberá asegurar que la aplicación es eficiente.
- Usabilidad: se validará que la aplicación se puede entender, aprender y operar.
- Fiabilidad: se validará que la aplicación tiene una suficiente tolerancia a fallos y capacidad de recuperación ante errores.
- Portabilidad: se validará que la aplicación ofrece unos niveles correctos de adaptabilidad, instalabilidad y coexistencia con otras aplicaciones.

Se asumen además todos los requisitos no funcionales descritos en el **ANEXO II - 2. Estudio de Viabilidad del Sistema.**

3. Seguridad y confidencialidad

La empresa adjudicataria y el personal encargado de la realización de las tareas guardarán secreto profesional sobre todas las informaciones, documentos y asuntos a los que tengan acceso o conocimiento durante la vigencia del contrato, estando obligados a no hacerlos públicos o enajenar cuantos datos conozcan como consecuencia o con ocasión de su ejecución, incluso después de finalizar el plazo contractual.

El adjudicatario se compromete a mantener estricta confidencialidad y a no revelar o ceder datos, ni aun para su conservación, o documentos proporcionados por la Administración o copia de los mismos, a terceros, para cualquier otro uso no previsto como necesario para el desempeño del proyecto, especialmente los datos de carácter personal.

Anexo VIII. Índice:

1. Evaluación de riesgos	2
1.1. Riesgos identificados	2
1.2. Exposición a riesgos	8
1.3. Planes de acción	12
1.4. Planes de contingencia	13

1. Evaluación de riesgos

1.1. Riesgos identificados

Riesgo es la Condición o evento incierto, que de darse, tiene efecto en al menos un objetivo del proyecto (ámbito, calendario, coste o productividad). Por tanto se hace necesario abordar, preparar y planificar las actividades de gestión de riesgos del proyecto para reducir los efectos adversos que puedan provocarse.

Para la clasificación de riesgos se utilizarán las tablas propuestas por TeraQuest^{©1} que proporcionan una visión global de riesgos del proyecto.

Factor ID	Descripción	Riesgo bajo	Riesgo medio	Riesgo alto	Descripción
Metas y objetivos					
1	Ajusta el proyecto a la estructura del cliente.	X			No se esperan cambios en la estructura del cliente.
2	Ajusta el proyecto a la estructura del proveedor.	X			No se esperan cambios para adaptarse a los roles requeridos para la realización del proyecto.
3	Workflow.		X		El producto podría ocasionar cambios en los flujos de datos del cliente.
Dirección del programa					
4	Influencias políticas.	X			No hay ninguna decisión impulsada por las políticas.
6	Establecimiento de fechas.	X			Las fechas se han establecido mediante un compromiso razonable entre proyecto y proceso.
7	Atractivo tecnológico.	X			Se trata de una tecnología madura.
8	Solución a Corto Plazo.			X	Se trata de un proyecto que puede variar debió al cambio de normas en la reglamentación.
Organización de Gestión					
9	Estabilidad de la organización.	X			La organización no puede sufrir cambios.
10	Organización de roles y responsabilidades.	X			Todos los roles radican en la misma persona.
11	Políticas y estándares.	X			El proyecto se desarrollará de acuerdo a estándares y políticas ampliamente

¹ TeraQuest, Riesgos genéricos del proyecto (Diciembre, 1999): <http://ais.msu.edu/internal/projectmgt/documents/ProjectRiskFactors.pdf>

					conocidas.
12	Apoyo a la Gestión.	X			El adjudicatario se encuentra comprometido con el éxito del proyecto.
13	Compromiso.	X			El compromiso es total.
14	Objetivos del proyecto.	X			Los objetivos están totalmente verificados por el cliente y son alcanzables por el adjudicatario.
Cliente					
15	Compromiso cliente.	X			El cliente de la aplicación ha descrito convenientemente todos los procesos y se encuentra comprometido.
16	Experiencia del cliente.			X	La experiencia del cliente es nula y eso provoca un alto nivel de incertidumbre.
17	Aceptación del cliente.		X		El cliente entiende el proyecto y los procesos pero puede pretender cambios.
18	Necesidades de formación del cliente.			X	El cliente precisa formación en la utilización del producto.
19	Justificación del cliente.		X		El cliente realizó una especificación completa pero existe la posibilidad de que la implantación del sistema produzca cambios en la especificación.
Parámetros del proyecto					
20	Tamaño del proyecto.	X			El tamaño del proyecto es pequeño.
21	Restricciones de hardware.	X			La aplicación puede ser soportada por un equipo comercial y no tiene problemas restricciones de tiempo.
22	Reutilización de componentes.	X			Ninguno de los componentes utilizados serán reutilizados.
23	Tamaño del presupuesto.		X		En caso de que el proyecto cambie de especificación, no está asegurado el presupuesto.
24	Restricciones de presupuesto.		X		El presupuesto debe adaptarse al tipo de organización que nos encontramos modelando.
25	Control de costos.		X		Los costos se encuentran controlados para los recursos establecidos.
26	Compromiso de entrega.	X			El compromiso por parte del adjudicatario es máximo, se cumplirán

					los plazos.
27	Compromiso de desarrollo.	X			Las fechas de cada fase han sido estudiadas y el compromiso es máximo.
Contenido del producto					
28	Estabilidad de los requisitos.		X		Los requisitos pueden cambiar.
29	Requisitos claros y completos.		X		Algunos requisitos podrían no encontrarse perfectamente especificados para su comprensión o no estar completos.
30	Facilidad de prueba.	X			El tamaño del proyecto y su complejidad baja, junto con trabajar con una tecnología conocida hace que el proyecto sea fácil de probar.
31	Dificultad de diseño.	X			Al tratarse de un proyecto de gestión, en el que el proceso a seguir está bien definido, el diseño no se complica en este caso.
32	Dificultad de implementación.	X			La formación tecnológica del desarrollador es alta y por tanto la implementación no es un riesgo elevado.
33	Dependencias del sistema.	X			A pesar de tener algunas dependencias, el sistema puede funcionar al completo sin ellas.
Despliegue					
34	Recursos de hardware para entregables.	X			Los recursos de que se disponen hacen que no sea un riesgo importante.
35	Respuesta de otros factores de rendimiento.	X			El proyecto no requiere de grandes recursos, los planificados serán suficientes.
36	Servicio de Impacto de cambios al cliente.	X			Los cambios pueden producir un impacto muy bajo sobre la herramienta.
37	Migración de Datos.	X			El proyecto no tiene migración de datos.
38	Enfoque experimental.	X			Es un proyecto con antecesor.
39	Integración externa o interna de interfaces.	X			La base de datos que se usará tiene una interfaz totalmente integrada con el entorno de desarrollo.

Proceso de desarrollo					
40	Análisis de alternativas.	X			Se estableció una comparación de los ciclos de desarrollo y coste hasta encontrar el que mejor se adaptaba al proyecto.
41	Compromiso en el proceso.		X		Los cambios son únicamente comunicados a los implicados.
42	Enfoque de aseguramiento de la calidad.	X			La calidad del producto final está asegurada debido al cumplimiento de estándares.
43	Documentación de desarrollo.	X			La documentación del proyecto estará disponible y completa.
44	Uso de la ingeniería de procesos.	X			Al realizar el desarrollo del proyecto se tiene en cuenta la ingeniería de procesos.
45	Temprana identificación de defectos.		X		Se incorporan técnicas para detección de defectos, pero pueden no ser suficientemente eficientes.
46	Seguimiento de defectos.		X		Existirá un enfoque evolutivo para el seguimiento de defectos.
47	Control de cambios en los productos.	X			Se utilizarán herramientas para gestionar cambios.
Entorno de desarrollo					
48	Instalación física.	X			La instalación es sencilla, no supone riesgo.
49	Plataforma de hardware.	X			Estable, sin cambios previstos, la capacidad es suficiente.
50	Disponibilidad de las herramientas.	X			La disponibilidad de las herramientas que se utilizarán es total y su documentación esta completa.
51	Apoyo de proveedores.		X		El apoyo por parte de los proveedores del entorno de desarrollo es nulo de manera directa, pero existe gran cantidad de apoyo de terceras partes.
52	Contrato establecido.			X	El contrato especifica cláusulas como el número de equipos sobre el que se puede instalar el software.
53	Recuperación de desastres.		X		Existen medidas de seguridad pero no hay un procedimiento de seguridad establecido.
Gestión del proyecto					

54	Enfoque PM (Project Management)	X			Los procesos de planificación y cumplimiento son llevados a cabo en el momento adecuado.
55	Comunicación PM.	X			La comunicación es clara en el equipo de desarrollo ya que está compuesto por un miembro.
56	Experiencia PM.		X		La experiencia es media sobre este tipo de proyectos.
57	Actitud PM.	X			La actitud es firme para conseguir el éxito del proyecto.
58	Autorización PM.	X			No es necesario implantar la gestión del liderazgo.
59	Apoyo de la PM.	X			El apoyo es máximo.
Equipo de desarrollo					
60	Disponibilidad de los miembros del equipo.		X		Es posible que el responsable del proyecto sea interrumpido en ocasiones de sus tareas.
61	Equipos con mezclas de habilidades.		X		El responsable del proyecto tiene solo unas ciertas habilidades sobre las que destaca.
62	Experiencia en aplicaciones.		X		La experiencia de desarrollo en proyectos de cualquier índole es media.
63	Experiencia con proyectos de hardware y software.	X			La experiencia sobre el hardware y el software del equipo es grande.
64	Experiencia con procesos.		X		El responsable tiene experiencia sobre procesos, pero quizás no la suficiente.
65	Formación del equipo.	X			La formación es continúa, aunque no demasiado necesaria para el tipo de desarrollo que se plantea.
66	Actitud y espíritu de equipo.	X			-
67	Productividad del equipo.	X			La productividad del responsable es alta.
68	Área de experiencia con la aplicación (de dominio).		X		La experiencia de desarrollo en este tipo de proyectos es media.
Tecnología					
69	Combinar la tecnología al proyecto.	X			La tecnología prevista para el proyecto es adecuada para los clientes y para

					su desarrollo.
70	Experiencia en la tecnología del equipo de proyecto.	X			La experiencia de desarrollo en la tecnología escogida es alta.
71	Disponibilidad de experto en la tecnología.	X			Expertos en tecnología de fácil acceso.
72	Madurez de la tecnología.	X			La tecnología está implantada en la industria desde hace bastante tiempo.
Mantenimiento					
73	Complejidad de diseño.	X			Baja complejidad, estructuralmente mantenible.
74	Personal de apoyo.			X	No existe personal de apoyo, el mantenimiento será llevado a cabo únicamente por el responsable.
75	Apoyo de proveedores.			X	No existe ningún tipo de apoyo por parte de los proveedores.
Total de categorías					13
Total de factores					75

Figura 4.1. Riesgos identificados

1.2. Exposición a riesgos

Una vez identificados los riesgos, se calcula² el efecto en euros que supone la exposición a dicho riesgo. De esta manera se tendrá una visión global del posible impacto económico en el proyecto.

Factor ID	Descripción	Probabilidad	Efecto (€)	Exposición
Metas y objetivos				
1	Ajusta el proyecto a la estructura del cliente	0,1	$0,1*(-500)=-50$	Baja
2	Ajusta el proyecto a la estructura del proveedor	0,1	$0,1*(-1000)=-100$	Baja
3	Workflow	0,3	$0,3*(-500)=-150$	Media
Dirección del programa				
4	Influencias políticas	0,1	$0,1*(-500)=-50$	Baja
6	Establecimiento de fechas	0,1	$0,1*(-1000)=-100$	Baja
7	Atractivo tecnológico	0,1	$0,1*(1000)=100$	Baja
8	Solución a Corto Plazo	0,7	$0,7*(1000)=700$	Alta
Organización de Gestión				
9	Estabilidad de la organización	0,1	$0,1*(1000)=100$	Baja
10	Organización de roles y responsabilidades	0,1	$0,1*(500)=50$	Baja
11	Políticas y estándares	0,1	$0,1*(500)=50$	Baja
12	Apoyo a la Gestión	0,1	$0,1*(500)=50$	Baja
13	Compromiso	0,1	$0,1*(500)=50$	Baja
14	Objetivos del proyecto	0,1	$0,1*(500)=50$	Baja
Cliente				
15	Compromiso cliente	0,1	$0,1*(-2000)=-200$	Baja
16	Experiencia del cliente	0,7	$0,7*(-2000)=-1400$	Alta
17	Aceptación del cliente	0,2	$0,2*(1000)=200$	Media

² Para el cálculo de probabilidades se usará una versión simplificada de la gestión de riesgos: RISK MANAGEMENT REPORTING (NASA, 2009): <http://standards.gsfc.nasa.gov/gsf-std/gsf-std-0002.pdf>
El cálculo del coste se basa en el coste aproximado en caso de que el riesgo se cumpliera completamente (Amenaza signo - Oportunidad signo +)

18	Necesidades de formación del cliente	0,8	$0,8*(-1000)=-800$	Alta
19	Justificación del cliente	0,5	$0,5*(-500)=-250$	Media
Parámetros del proyecto				
20	Tamaño del proyecto	0,1	$0,1*100=10$	Baja
21	Restricciones de hardware	0,1	$0,1*100=10$	Baja
22	Reutilización de componentes	0,1	$0,1*100=10$	Baja
23	Tamaño del presupuesto	0,3	$0,3*(-500)=-150$	Media
24	Restricciones de presupuesto	0,7	$0,7*(-1000)=700$	Media
25	Control de costos	0,5	$0,5*(-500)=-250$	Media
26	Compromiso de entrega	0,1	$0,1*1000=100$	Baja
27	Compromiso de desarrollo	0,1	$0,1*1000=100$	Baja
Contenido del producto				
28	Estabilidad de los requisitos	0,4	$0,4*(-1000)=400$	Media
29	Requisitos claros y completos	0,4	$0,4*(-1000)=400$	Media
30	Facilidad de prueba	0,1	$0,1*1000=100$	Baja
31	Dificultad de diseño	0,1	$0,1*1000=100$	Baja
32	Dificultad de implementación	0,1	$0,1*1000=100$	Baja
33	Dependencias del sistema	0,1	$0,1*1000=100$	Baja
Despliegue				
34	Recursos de hardware para entregables	0,1	$0,1*1000=100$	Baja
35	Respuesta de otros factores de rendimiento	0,1	$0,1*1000=100$	Baja
36	Servicio de Impacto de cambios al cliente	0,1	$0,1*(-1000)=-100$	Baja
37	Migración de Datos	0,1	$0,1*(-100)=-10$	Baja
38	Enfoque experimental	0,1	$0,1*(-500)=-50$	Baja
39	Integración externa o interna de interfaces	0,2	$0,2*300=60$	Baja
Proceso de desarrollo				
40	Análisis de alternativas	0,1	$0,1*500=50$	Baja

41	Compromiso en el proceso	0,4	$0,4*(-750)=-300$	Media
42	Enfoque de aseguramiento de la calidad	0,1	$0,1*1000=800$	Baja
43	Documentación de desarrollo	0,1	$0,1*500=50$	Baja
44	Uso de la ingeniería de procesos	0,1	$0,1*500=50$	Baja
45	Temprana identificación de defectos	0,8	$0,8*(-3000)=-2400$	Media
46	Seguimiento de defectos	0,8	$0,8*(-3000)=-2400$	Media
47	Control de cambios en los productos	0,1	$0,1*1000=100$	Baja
Entorno de desarrollo				
48	Instalación física	0,1	$0,1*1000=100$	Baja
49	Plataforma de hardware	0,1	$0,1*1000=100$	Baja
50	Disponibilidad de las herramientas	0,1	$0,1*1000=100$	Baja
51	Apoyo de proveedores	0,8	$0,8*(-1000)=-800$	Media
52	Contrato establecido	0,8	$0,8*(-1000)=-800$	Alta
53	Recuperación de desastres	0,3	$0,3*(-1500)=-450$	Media
Gestión del proyecto				
54	Enfoque PM	0,1	$0,1*1500=150$	Baja
55	Comunicación PM	0,1	$0,1*1500=150$	Baja
56	Experiencia PM	0,1	$0,1*1500=150$	Media
57	Actitud PM	0,1	$0,1*1500=150$	Baja
58	Autorización PM	0,1	$0,1*1500=150$	Baja
59	Apoyo de la PM	0,1	$0,1*1500=150$	Baja
Equipo de desarrollo				
60	Disponibilidad de los miembros del equipo	0,4	$0,4*(-2000)=-800$	Media
61	Equipos con mezclas de habilidades	0,6	$0,6*1000=600$	Media
62	Experiencia en aplicaciones	0,8	$0,8*500=400$	Media
63	Experiencia con proyectos de hardware y software	0,8	$0,8*500=400$	Baja
64	Experiencia con procesos	0,4	$0,4*(-500)=-200$	Media
65	Formación del equipo	0,1	$0,1*(-500)=-50$	Baja

66	Actitud y espíritu de equipo	0,1	$0,1*1000=100$	Baja
67	Productividad del equipo	0,1	$0,1*1000=100$	Baja
68	Área de experiencia con la aplicación (de dominio)	0,5	$0,5*1000=500$	Media
Tecnología				
69	Combinar la tecnología al proyecto	0,1	$0,1*1000=100$	Baja
70	Experiencia en la tecnología del equipo de proyecto	0,1	$0,1*1000=100$	Baja
71	Disponibilidad de experto en la tecnología	0,1	$0,1*(-1000)=-100$	Baja
72	Madurez de la tecnología	0,1	$0,1*100=10$	Baja
Mantenimiento				
73	Complejidad de diseño	0,2	$0,2*1000=200$	Baja
74	Personal de apoyo	0,8	$0,8*(-1000)=-800$	Alta
75	Apoyo de proveedores	0,8	$0,8*(-1000)=-800$	Alta

Figura 4.2. Exposición a riesgos

1.3. Planes de acción

Algunas de las medidas que se van a llevar a cabo en el proyecto BillGEST para evitar la exposición de riesgos son:

- Durante todo el desarrollo del proyecto, deberá realizarse por el Jefe de Proyecto un seguimiento del cumplimiento de estándares de la calidad y de la normativa ISO en cuanto a desarrollo de proyectos informáticos se refiere y calidad en los mismos.
- Para evitar que la inexperiencia del cliente produzca fracaso del proyecto, se realizarán entrevistas con el mismo de forma periódica, de manera que se le va enseñando lo que se encuentra realizado y se le pide opinión para poder encontrar requisitos cambiantes en las fases más tempranas del proyecto. Además, a través de estas reuniones se le ofrece poca familiarización con el producto, haciéndosele más sencilla su utilización cuando se encuentre finalizado.
- A la hora de introducir datos históricos en la aplicación, se consultará con el cliente si dicho proceso desea que sea llevado a cabo por nuestra organización con el consiguiente aumento en el coste o si por el contrario será realizado por otra persona, evitando así riesgos de aumentos de presupuesto injustificados.
- Debido a la inexperiencia del cliente, se proporcionará un registro de todos los acontecimientos ocurridos en el sistema, de forma que si alguno de estos ocasiona un fallo, a través de esta documentación, el desarrollador podrá saber los pasos que ocasiono el fallo.
- Se le entregará al cliente una documentación del producto final, sobre la que se podrá encontrar también un manual de usuario con la descripción completa del uso de la herramienta. Como complemento se les ofrecerá a los usuarios de la aplicación un curso de formación que será impartido por el responsable del proyecto.
- Para asegurar que el producto final tiene la máxima calidad, se buscarán casos de prueba con valores límite.

1.4. Planes de contingencia

Para establecer el plan de contingencia se van a tener en cuenta los riesgos con mayores posibilidades de afectar a la evolución del proyecto.

- Si por la implantación de la nueva herramienta se producen cambios en los flujos de datos del cliente, se le enseñara al cliente cual es este flujo de datos, como se origina y cuál es el nuevo comportamiento que tienen.
- En caso de que durante la realización del proyecto, aparezcan problemas de implementación los cuáles el responsable del proyecto no sea capaz de resolver, se fomentará un contrato con una consultora experta en tecnología .NET que cubrirá las insuficiencias técnicas de la empresa mediante la impartición de algún curso.
- En caso de que se produzca un defecto grave en el sistema, el responsable se compromete a hacerse cargo del coste de dicho problema sin coste para la FBCV. Esto supone que la calidad y los procesos que la controlan deben ser inspeccionados de forma cuidadosa.
- Si no se cumpliesen las fechas marcadas como consecución de tareas, se respetarán los tiempos asignados a cada tarea y se retrasará la fecha de entrega final del proyecto tantos días como se haya retrasado la suma de todas las tareas.
- En caso de que cambie la reglamentación de los campeonatos de billar, las modificaciones que requiera la aplicación por estas causas deberán ser tenidas en cuenta como ampliaciones o modificaciones del proyecto, y en ningún caso su coste será asumido por el proveedor.