



UNIVERSITAT
POLITÈCNICA
DE VALÈNCIA



etsinf Escuela Técnica Superior de Ingeniería **Informática**

Escuela Técnica Superior de Ingeniería Informática.

Desarrollo de un Servidor de Internet: Gestión de Competiciones Deportivas.

PROYECTO FIN DE CARRERA.

Titulación

Ingeniería Técnica en Informática de Gestión

Presentado por

Abdelmajid Berkane

Dirigido por

Hassan Mohamed, Houcine

Valencia, Septiembre 2011

Índice

1. Introducción	6
1.1 Internet	6
1.2 Tecnologías	7
1.2.1 ASP .NET	7
1.2.2 Visual Studio .NET 2008	10
1.3 Entorno.....	11
2. Descripción General del Sistema	13
2.1 Requisitos de Gestión de Información	16
2.2 Requisitos Funcionales	18
• Dar de alta un equipo.....	19
• Añadir un jugador a un equipo.....	19
• Definir un delegado de un equipo.....	19
• Dar de alta un deporte.....	20
• Dar de alta un responsable de competiciones.....	20
• Dar de alta una competición por puntos.....	20
• Dar de alta una competición por posiciones.....	21
• Cerrar inscripción de una competición.....	21
• Inscribir un equipo en una competición.....	21
• Crear un partido en una competición.....	22
• Introducir resultado de un partido.....	22
• Actualizar la clasificación tras un partido jugado.....	23
2.3 Requisitos de Interfaz de Usuario.....	24
2.3.1 Usuarios Anónimos	24
2.3.2 Usuarios Jugadores	24

2.3.3	Usuarios Responsables de competiciones	24
3.	Modelación de la web	26
3.1	Modelo de Navegación	27
3.1.1	Diagrama de Usuarios	29
3.1.2	Mapas Navegacionales.....	29
3.1.2.1	Mapas Navegacionales del usuario Anónimo.....	29
3.1.2.2	Mapas Navegacionales del usuario Jugador	29
3.1.2.3	Mapas Navegacionales del usuario Responsable de la competición	29
4	Implementación	47
4.1	Arquitectura	47
4.2	Diseño de la Base de Datos	48
4.3	Patrones de Traducción	49
4.3.1	Anónimo	49
4.3.2	Jugador	49
4.3.3	Responsable de la competición	49
4.4	Contexto de Navegación y Relación de Conntexto	49
4.4.1	Anónimo	49
4.4.2	Jugador	49
4.4.3	Responsable de la competición	49
4.5	Ejemplos	53
5	Diseño Relacionl de BD	57
5.1	Introducción.....	57
5.2	Diseño Logico.....	57
6.	Conclusión	66
7.	Bibliografía.....	67
7.	ANEXO A: Confugyración de la aplicación en el Servidor.....	69

Tabla de ilustraciones

<i>Ilustración 1: Aplicaciones Software para la Web.</i>	12
<i>Ilustración 2: Los resultados de la competición por posición (Equipo E10 contra Equipo E15).</i>	16
<i>Ilustración 3: Diagrama de Clases (UML) del sistema GCD.</i>	28
<i>Ilustración 4: Diagrama de usuarios del GCD.</i>	29
<i>Ilustración 5: Mapa de Navegación del usuario Anónimo.</i>	30
<i>Ilustración 6: Subsistema Clasificaciones-competición (usuario Anónimo).</i>	31
<i>Ilustración 7: Mapa de Navegación del usuario Jugador.</i>	32
<i>Ilustración 8: Subsistema Clasificaciones-competición (usuario Jugador).</i>	33
<i>Ilustración 9. Mapa Navegacional del usuario Responsable de la Competición.</i>	35
<i>Ilustración 10: Detalle del Subsistema Competiciones (Responsable Competiciones).</i>	36
<i>Ilustración 11: Detalle del Subsistema Deportes (Responsable Competiciones).</i>	37
<i>Ilustración 12: Detalle subsistema Equipos (Responsable de competiciones)</i>	38
<i>Ilustración 13: Detalles subsistema Partidos (Responsable de competiciones)</i>	38
<i>Ilustración 14: Detalles subsistema Clasificaciones(Responsable de competiciones)</i>	38
<i>Ilustración 15: Detalle subsistema Otros (Responsable de competiciones)</i>	39
<i>Ilustración 16: Relación de contexto (Usuario Anónimo)</i>	40
<i>Ilustración 17: Enlace contextual de servicio de competiciones por posición del usuario Anónimo.</i>	41
<i>Ilustración 18: Enlace contextual de servicio de competiciones por puntos del usuario Anónimo.</i>	42
<i>Ilustración 19: Contexto Equipo del usuario Jugador.</i>	43
<i>Ilustración 20: Enlace contextual de servicio de del contexto Partidos del usuario Jugador.</i>	44
<i>Ilustración 21: Enlace contextual de servicio del contexto Competiciones por puntos del usuario Jugador.</i> ...	45
<i>Ilustración 22: Enlace contextual de servicio del contexto Competiciones por posición del usuario Jugador.</i> .	46
<i>Ilustración 23: Esquema de Bases de datos de la aplicación GCD.</i>	48
<i>Ilustración 24: Esquema de las tablas de Bases de datos de la aplicación GCD.</i>	57
<i>Ilustración 25: Esquema de la tabla de Bases de datos DEPORTE.</i>	58
<i>Ilustración 26: Esquema de la tabla de Bases de datos EQUIPO.</i>	59
<i>Ilustración 27: Esquema de la tabla de Bases de datos PARTIDO.</i>	60
<i>Ilustración 28: Esquema de la tabla de Bases de datos CLASIFICACIÓN.</i>	60
<i>Ilustración 29: Esquema de la tabla de Bases de datos COMPETICIÓN.</i>	62
<i>Ilustración 30: Esquema de la tabla de Bases de datos COMPETICIÓN POR POSICIÓN.</i>	63
<i>Ilustración 31: Esquema de la tabla de Bases de datos COMPETICIÓN POR PUNTOS.</i>	63
<i>Ilustración 32: Esquema de la tabla de Bases de datos USUARIO.</i>	64
<i>Ilustración 33: Esquema de la tabla de Bases de datos RESPONSABLE DE LA COMPETICIÓN.</i>	64
<i>Ilustración 34: Esquema de la tabla de Bases de datos JUGADOR</i>	65
<i>Ilustración 35: Esquema de la tabla de Bases de datos JUGADOR_EQUIPO</i>	65

1. Introducción

1.1 Internet

El Internet, es un sistema mundial de redes de computadoras, por medio del cual un usuario en cualquier computadora puede, en caso de contar con los permisos apropiados, acceder información de otra computadora y poder tener una comunicación directa con otros usuarios en otras computadoras.

Fue concebido por la agencia de nombre **ARPA** (Advanced Research Projects Agency) del gobierno de los Estados Unidos en el año de 1969 y se le conocía inicialmente como **ARPANET**. El propósito original fue crear una red que permitiera a los investigadores en un Campus poder comunicarse a través de los sistemas de cómputo con investigadores en otras Universidades.

Hoy en día, el Internet es un medio de comunicación público, cooperativo y autosuficiente en términos económicos, accesible a cientos de millones de personas en el mundo entero. Físicamente, el Internet usa parte del total de recursos actualmente existentes en las redes de telecomunicaciones. Técnicamente, lo que distingue al Internet es el uso del protocolo de comunicación **TCP/IP** (Transmission Control Protocol/Internet Protocol).

En la última década, el uso de internet ha evolucionado mucho. Dentro del término *Marketing en Internet*, El internet se ha usado para comercializar productos y servicios que satisfagan las necesidades de los clientes. Los especialistas en Internet han diseñado programas y campañas para atraer a clientes potenciales hacia el *Sitio Web* y convencerlos de registrar sus nombres o adquirir productos...

Por otro lado aparece el término *World Wide Web*, o simplemente **Web**. Es el componente más usado en el Internet, una fuente inagotable del conocimiento humano.

Usando el Web, se tiene acceso a millones de páginas de información. La exploración en el Web se realiza por medio de un software denominado *Browser* o *Explorador*. La apariencia de un Sitio Web puede variar ligeramente dependiendo del explorador que use. Así mismo, las versiones más recientes disponen de una

funcionalidad mucho mayor tal como animación, realidad virtual, sonido y música...

1.2 Tecnologías

1.2.1 ASP .NET

Cada cierto tiempo se produce una revolución en la tecnología empleada en el desarrollo de aplicaciones. El primer gran cambio que vivimos fue el paso de la programación en **MS-DOS** a la programación Windows **Win16**, más tarde con la evolución de los sistemas operativos, el segundo cambio, las aplicaciones **Win32**, y ahora se está viviendo la tercera de las revoluciones, lo que Microsoft llama **.NET Framework** o más comúnmente **.NET**.

Microsoft lanzo esta nueva tecnología como respuesta a tecnología *Java de Sun*. El Framework **.NET** tiene grandes similitudes con la plataforma Java. Centrándonos en el concepto Framework **.NET**, es un marco en donde nuestras aplicaciones correrán. Nuestras aplicaciones ya no corren directamente bajo el sistema operativo si no que corren bajo este armazón o marco.

Los elementos principales del **.NET Framework** son:

- CLR (*Common Language Runtime*)
- El conjunto de clases del **.NET Framework**
- ASP.NET
- Los servicios Web
- Remoting
- Windows Forms

El **CLR** es el motor de ejecución de las aplicaciones **.NET**, lo que en Java sería la máquina virtual de Java, este motor se encarga de ejecutar todo el código **.NET** para ello ha de ser en dicho lenguaje. El CLR es el encargado de convertir este lenguaje intermedio en lenguaje máquina del procesador, esto normalmente se hace en tiempo real por un compilador JIT (*Just-In-Time*) que lleva incorporado el

CLR.

El conjunto de clases del .NET Framework es la piedra angular de cualquier desarrollador de .NET, es un conjunto de clases, interfaces, tipos que simplifican y optimizan el desarrollo de aplicaciones .NET además de proporcionar acceso a la funcionalidad del sistema.

ASP.NET es la parte del .NET Framework dedicada al desarrollo web. A través del servidor web **IIS** (Internet Information Services) nuestras aplicaciones ASP.NET se ejecutarán bajo el CLR y podremos usar el conjunto de clases del .NET Framework para desarrollarlas, obteniendo así una versatilidad y una potencia nunca antes conseguida en las aplicaciones ASP.

También son destacables los servicios web, que nos permitirán comunicarnos a través de Internet entre diferentes ordenadores, incluso entre distintos sistemas. Así como .NET *Remoting* que nos permite tener objetos en máquinas remotas e invocarlos desde otras máquinas. Y las *Windows Forms*, parte del .NET Framework que permite crear aplicaciones en el más clásico de los sentidos.

Centrándonos en ASP.NET, se puede afirmar que se ha construido bajo los siguientes principios:

- Facilidad de desarrollo
- Alto rendimiento y escalabilidad
- Mejora fiabilidad
- Fácil distribución e instalación

Suena bastante bien, aparentemente todo el mundo busca estas metas, vamos a ver cómo se ha intentado acercar a cada una de ellas.

Facilidad de desarrollo:

ASP.NET introduce un nuevo concepto, los "*Server Controls*", que permiten a modo de etiquetas HTML tener controles manejados por el servidor que identifican el navegador usado adaptándose para cada uno de ellos. Tareas tediosas como la validación de datos se convierten en fáciles y sencillas.

Posibilidad de elección del lenguaje de programación. Puedes elegir el

lenguaje de programación que más le guste, por defecto lleva integrado **C#**, **VB.NET** y **J#**, pero podría usar otro lenguaje.

Independencia de la herramienta de desarrollo. Puedes utilizar desde el *Bloc de Notas*, hasta la sofisticada y potente Visual Studio .NET, pasando por la gratuita *Web Matrix*. Y lo mejor de todo es la biblioteca de clases que lleva incorporada, ya no necesitarás obtener componentes de otras empresas para por ejemplo enviar un email, hacer "upload" de un fichero o generar gráficos en tiempo de ejecución.

Alto rendimiento y escalabilidad:

El código es compilado para ser ejecutado en el CLR. Puedes optar por tenerlo en el servidor precompilado o dejar que el servidor lo compile la primera vez que lo ejecute. El resultado es de 3 a 5 veces superior en velocidad que las antiguas páginas ASP.

Rico sistema de caché. El uso adecuado del potente caché incorporado aumenta considerablemente el rendimiento y la escalabilidad de la aplicación. La caché te permitirá cachear desde páginas completas a partes completas, pasando por conjuntos de datos extraídos de la base de datos.

ASP.NET está preparado para poder tener granjas de servidores web para sitios con alto volumen de tráfico y repartir la carga entre distintos servidores.

Mejora de la fiabilidad:

ASP.NET es capaz de detectar pérdidas de memoria, problemas con bloqueos y protección ante caídas. Entre otras cosas, es capaz de detectar aplicaciones web que pierden memoria, arrancando otro proceso limpio con una nueva instancia de la aplicación para cerrar la que pierde memoria liberando así la memoria perdida.

Fácil distribución e instalación:

Una aplicación *ASP.NET* se instala tan fácilmente como copiando los ficheros que la componen. No es necesario registrar ningún componente, tan solo copiar los ficheros al web.

Se puede recompilar la aplicación o enviar nuevos ficheros sin necesidad de reiniciar la aplicación ni el servidor web.

ASP .NET ha sido el lenguaje usado para la implementación del aspecto dinámico de la aplicación Web, acompañado del *HTML* que se ha usado para la implementación del aspecto estático de las páginas web y *JavaScript* usado para implementar procesos más específicos de la aplicación Web.

1.2.2 Visual Studio .NET 2008

Visual Studio .NET 2008 es un **IDE** (Integrated Development Environment), entorno integrado de desarrollo, desarrollado por Microsoft para ser usado bajo el sistema operativo *Microsoft Windows*.

El nuevo *framework .Net 3.5* está diseñado para aprovechar las ventajas que ofrecen los nuevos sistemas operativos – a partir de "Windows Vista" – a través de sus subsistemas "Windows Communication Foundation" **WCF** y "Windows Presentation Foundation" **WPF**. El primero tiene como objetivo la construcción de aplicaciones orientadas a servicios mientras que el último apunta a la creación de interfaces de usuario más dinámicas que las conocidas hasta el momento.

A las mejoras de desempeño, escalabilidad y seguridad con respecto a las anteriores versiones, se agregan entre otras, las siguientes novedades.

La mejora en las capacidades de Pruebas Unitarias permiten ejecutarlas más rápido independientemente de si lo hacen en el entorno IDE o desde la línea de comandos. Se incluye además un nuevo soporte para diagnosticar y optimizar el sistema a través de las herramientas de pruebas de Visual Studio. Con ellas se podrán ejecutar perfiles durante las pruebas para que ejecuten cargas, prueben procedimientos contra un sistema y registren su comportamiento; y utilizar herramientas integradas para depurar y optimizar.

Con *Visual Studio Tools for Office VSTO* integrado con Visual Studio 2008 es posible desarrollar rápidamente aplicaciones de alta calidad basadas en la interfaz de usuario (UI) de Office que personalicen la experiencia del usuario y mejoren su productividad en el uso de Word, Excel, PowerPoint, Outlook, Visio, InfoPath y Project. Una completa compatibilidad para implementación con *ClickOnce* garantiza el entorno ideal para una fácil instalación y mantenimiento de las soluciones Office.

Visual Studio 2008 permite incorporar características del nuevo Windows Presentation Foundation sin dificultad tanto en los formularios de Windows existentes como en los nuevos. Ahora es posible actualizar el estilo visual de las aplicaciones al de Windows Vista debido a las mejoras en *Microsoft Foundation Class Library* (MFC) y *Visual C++*. *Visual Studio 2008* permite mejorar la interoperabilidad entre código nativo y código manejado por .NET. Esta integración más profunda simplificará el trabajo de diseño y codificación.

LINQ (*Language Integrated Query*) es un nuevo conjunto de herramientas diseñado para reducir la complejidad del acceso a Base de Datos, a través de extensiones para C++ y Visual Basic así como para Microsoft .NET Framework. Permite filtrar, enumerar, y crear proyecciones de muchos tipos y colecciones de datos utilizando todas la misma sintaxis, prescindiendo del uso de lenguajes especializados como *SQL* o *XPath*.

Visual Studio 2008 ahora permite la creación de soluciones multiplataforma adaptadas para funcionar con las diferentes versiones de *.Net Framework: 3.5*.

.NET 3.5 incluye biblioteca *ASP.NET AJAX* para desarrollar aplicaciones web más eficientes, interactivas y altamente personalizadas que funcionen para todos los navegadores más populares y utilicen las últimas tecnologías y herramientas Web, incluyendo *Silverlight* y *Popfly*.

1.3 Entorno

El proyecto consiste en el diseño y la implementación de una aplicación Web para la gestión de competiciones deportivas on-line, se trata particularmente de la gestión de competiciones deportivas en el ámbito universitario, por lo tanto la información y las ideas que se han usado en la aplicación están relacionadas de alguna forma con la Universidad Politécnica de Valencia (U.P.V.), albergados en una web ficticia: www.GCD.upv.es

El sitio Web puede verse como un modelo estándar de cualquiera aplicación software, usando el web como la interfaz de usuario con la cual interactúa el usuario comprende todos los puntos de contacto entre los usuarios y la aplicación: introducir datos, visualizar información, etc.

La aplicación Web se encuentra dividida en dos partes bien diferenciadas,

una parte pública de acceso a los usuarios anónimos y una parte privada para los usuarios registrados (que son los Jugadores y los Responsables de las competiciones como veamos más tarde).

A la parte pública podrá entrar cualquier usuario que acceda a la Web desde un navegador (Internet Explorer, Mozilla Firefox, etc.).

La parte privada tiene opciones añadidas a los del usuario anónimo, en esta parte solo pueden entrar los usuarios registrados, donde los jugadores podrán gestionar todo lo que está relacionado con sus equipos mientras que los responsables de las competiciones se encargan de administrar las competiciones (ya veremos esas funcionalidades con mas detalles en los posteriores apartados).

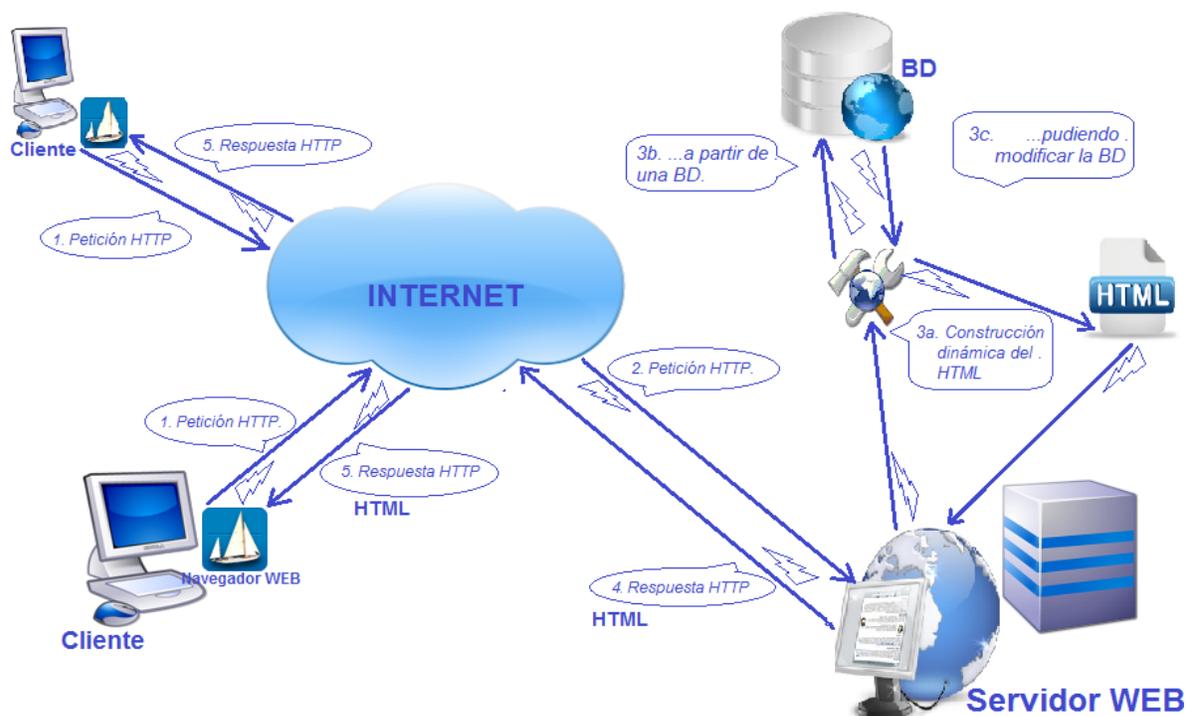


Ilustración 1: Aplicaciones Software para la Web.

2. Descripción General del Sistema

Este sistema debe permitir gestionar competiciones deportivas. Para ello, el sistema deberá mantener información sobre las competiciones en diferentes deportes, equipos y partidos jugados y por jugar, para obtener las clasificaciones de estas competiciones.

Al sistema se podrán conectar tantos usuarios anónimos, jugadores y responsables de competiciones.

Los anónimos sólo podrán visualizar las competiciones en marcha y las clasificaciones en cada competición. Esta clasificación indicará una posición de los equipos en la competición.

Los jugadores podrán gestionar sus equipos, visualizar sus clasificaciones actuales (la posición que ocupan en alguna competición) así como el resto de clasificaciones de competiciones en las que no participan, proponer partidos en competiciones por posición (aunque no se ha implementado) y ver el calendario de partidos por jugar.

Los responsables de competiciones podrán crear competiciones, crear el calendario inicial de competiciones por puntos (indicando los partidos) y consultar la clasificación provisional de cada competición para sacar reportes.

Pensando en poder aplicarlo en ámbitos Universitarios, se desea extender las capacidades del sistema para que se pueda otorgar a los jugadores (alumnos) un índice de actividad, para premiar con créditos de libre elección a aquellos que estén involucrados activamente en diferentes competiciones y participen como delegados de estas competiciones.

También se premiará a los ganadores de las diferentes competiciones, promovidas por la Universidad (competiciones oficiales).

Las competiciones

El interés principal del sistema es su sistema doble de competiciones:

Por puntos y por posición.

Competiciones por puntos (Las clásicas)

El responsable de las competiciones establece los puntos que obtiene cada equipo por ganar, perder o empatar un partido y, en base a los equipos que se han apuntado a la competición, establece un calendario de partidos (a priori fijo) en los que se realizará dicha competición.

El equipo ganador será el que haya ganado más puntos en sus partidos predefinidos.

Competiciones por posición

Se pretende establecer un modelo de competiciones más abierto y dinámico.

En este tipo de competiciones, el responsable sólo debe indicar la distancia máxima entre las posiciones actuales de los equipos, y la fecha de inicio y fin de la competición.

Mientras esta competición no haya finalizado, podrán inscribirse nuevos equipos (pasarán a obtener la posición más baja), a pesar que la competición haya empezado. En esta modalidad de competición no hay partidos predefinidos, de manera que, mientras dure la competición, cualquier equipo podrá proponer un partido contra otro equipo que participe en la competición, siempre que su diferencia de posiciones no sea mayor que la distancia máxima entre posiciones indicada en la competición.

Así, por ejemplo, en una competición con una distancia máxima entre posiciones de 5 puestos, un equipo que ocupe la posición 10 en la clasificación puede retar a cualquier equipo que esté entre la 5ª y la 15ª posición. Cuando se juega un partido en esta modalidad de competición por posición, la posición de cada equipo se actualiza en base a las posiciones de los equipos en la competición.

Supongamos que los equipos E1 y E2, que se han inscrito a una competición por posición juegan un partido, y gana E1 (no importa que juegue como local o como visitante).

- Si la posición en la competición de E1 es mayor que la de E2, E1 gana el número de posiciones que estén indicadas en la competición como *posiciones al ganar* (típicamente 1 posición) y E2 pierde una posición, que ocupa el equipo que iba tras él en la clasificación.
- Si la clasificación de E1 es menor que la de E2, E1 gana la posición que tenía E2, y E2 pierde tantos puestos como se indiquen en la competición *posiciones al no ganar*. Todos los equipos a los que E1 ha adelantado actualizarán su clasificación (perdiendo una posición) para dejar una clasificación global correcta.

Por ejemplo, supongamos la siguiente configuración, para una competición basada en posiciones:

- ✓ Distancia máxima entre posiciones = 5;
- ✓ Posiciones al Ganar = 2;
- ✓ Posiciones al No Ganar=3;

Supongamos que existen 20 equipos (E1 a E20) que ocupan la posición de la 1 a la 20 en la clasificación, respectivamente. Supongamos que se ha jugado un partido entre el equipo E10 y el E15 (válido según la distancia máxima entre posiciones). Hay tres opciones:

- a) El equipo E10 gana. Al ir delante en la clasificación, ganará 2 posiciones, asando a ocupar la posición 8. Esto provocará que los equipos E8 y E9 pierdan una posición cada uno. E15, por su parte, perderá sólo una posición, ya que iba detrás en la competición. Ocupará la posición 16 el quipo E16, ahora actualizará su clasificación para subir a la posición 15.
- b) El equipo E15 gana. Al ir detrás en la clasificación, pasará a ocupar la posición de E10, puesto 10. E10, por su parte, bajará 3 posiciones (al perder), y ocupará la posición 13. El resto de equipos se moverán acorde a los cambios. E13 y E14 perderán una posición, y pasarán a ocupar las posiciones 14 y 15 respectivamente.
- c) Hay un empate. E15 se queda como está. E10 pierde 3 posiciones al no ganar (pasa a la posición 13). E11, E12 y E13 ganan una posición cada uno, pasando a la posición 10, 11 y 12, respectivamente.

La siguiente tabla muestra los resultados en cada caso.

Posición en la competición	Equipos	a) Gana E10	b) Gana E15	c) Empate
8	E08	↑ ↑ E10		
9	E09	↓ E08		
10	E10	↓ E09		
11	E11		↑ ↑ ↑ ↑ ↑ E15	↑ E11
12	E12			↑ E12
13	E13			↑ E13
14	E14		↓ ↓ ↓ E10	↓ ↓ ↓ E10
15	E15	↑ E16	↓ E13	
16	E16	↓ E15	↓ E14	
17	E17			

↑ : El equipo sube una posición.

↓ : El equipo baja una posición.

Ilustración 2: Los resultados de la competición por posición (Equipo E10 contra Equipo E15).

Además, estos dos tipos de competiciones (tanto las de por posición como por puntos), podrán ser competiciones oficiales o no oficiales. Las competiciones oficiales se diferencian de las no oficiales porque sus ganadores reciben un número créditos de libre elección que se indica en la competición.

2.1 Requisitos de Gestión de Información

El sistema deberá almacenar información sobre las Competiciones. Una competición regula un calendario de partidos entre equipos, de los que al final ganará uno.

Una competición pertenece a un DEPORTE, tiene un responsable, y de esta competición se debe mantener información sobre su nombre, fecha de inicio y de fin, y el número de jugadores que deben poseer los equipos.

Existen dos tipos de competición:

- Por puntos
- Y por posición.

✓ Competiciones por puntos

Cada equipo posee una clasificación, que indica los puntos que lleva obtenidos según las partidas que haya ganado, perdido o empatado hasta el momento. Cada competición por puntos debe establecer el número de puntos que se obtiene tras ganar, perder o empatar un partido de esta competición.

✓ Competiciones por posición

Cada equipo posee una clasificación en forma de posición dentro de la competición. Cada vez que se gana o pierde un partido, esta clasificación (posición) se actualiza en base a la clasificación del otro equipo y el valor de 'posiciones al ganar' o 'posiciones al no ganar' (para casos de empate o pérdida) que tenga establecida esta competición. Cada competición por posición puede establecer además un valor de 'distancia máxima entre posiciones', el cual se usa para no permitir que dos equipos con clasificación muy distante jueguen partidos (y por tanto, conseguir partidas entre equipos del mismo nivel), favoreciendo así el juego entre equipos con un nivel parecido.

Los Jugadores deben organizarse en Equipos para participar en las competiciones. De cada jugador es necesario únicamente su nombre y email, y puede formar parte de cuantos equipos quiera. De un equipo se necesita mantener el nombre, y el jugador que actuará como delegado (sólo puede ser un jugador). En un equipo puede haber un número indeterminado de jugadores. Los equipos se inscriben en las competiciones (en tantas como se quieran inscribir), y poseen una clasificación en cada una de ellas.

El sistema debe mantener información sobre las Partidas entre los diferentes equipos de una competición. Una partida tiene una fecha, hora y lugar de juego, y se establece a un equipo como local y otro como visitante (en muchas competiciones, esto no aportará información adicional). Cuando una partida se

juegue, se indicará el marcador (como un texto) y, o bien se debe marcar como partido empatado (habrá deportes donde esto no sea posible), o bien se indicará el equipo ganador.

Por último, el sistema debe mantener información sobre los Responsables de competiciones. Cada responsable puede estar al cargo de varias competiciones, y se debe mantener información sobre su nombre y email.

2.2 Requisitos Funcionales

- Registrar un jugador
- Dar de alta un equipo
- Añadir un jugador a un equipo
- Definir un delegado de un equipo
- Dar de alta un deporte
- Dar de alta un responsable de competiciones
- Dar de alta una competición por puntos
- Dar de alta una competición por posiciones
- Cerrar inscripción de una competición
- Inscribir un equipo en una competición
- Crear un partido
- Introducir resultado de un partido
- Actualizar la clasificación tras un partido jugado
- Asignar créditos por actividades deportivas a un jugador
- Asignar créditos por ganar una competición oficial a un equipo

A continuación se describe cada una de estas operaciones con más detalle:

- **Registrar un jugador**

Descripción: Da de alta un jugador

Entradas: Nombre, email y contraseña

Efecto: Crea un nuevo usuario jugador

Restricciones: Ninguna

- ***Dar de alta un equipo***

Descripción: Crea un nuevo equipo

Entradas: Nombre, Jugador

Efecto: Crea un nuevo equipo, le añade el jugador que se indica, y lo marca como delegado (por defecto).

Restricciones: Ninguna

- ***Añadir un jugador a un equipo***

Descripción: Añade un nuevo jugador a un equipo

Entradas: Equipo, Jugador

Efecto: Añade al Jugador al Equipo indicado.

Restricciones: El Jugador no debe estar ya en el equipo.

- ***Definir un delegado de un equipo***

Descripción: Asigna a un jugador de un equipo la tarea de delegado

Entradas: Equipo, Jugador

Efecto: Asigna a Jugador el rol de delegado del Equipo. Si ya existía uno (por defecto siempre existe uno), lo cambiará (quitará el anterior y asignará el nuevo).

Restricciones: El Jugador debe ser parte del equipo para hacerlo delegado

- ***Dar de alta un deporte***

Descripción: Crea un nuevo deporte en el que podrán organizarse competiciones

Entradas: Nombre

Efecto: Crea un nuevo deporte

Restricciones: Ninguna

- ***Dar de alta un responsable de competiciones***

Descripción: Crea un nuevo responsable de Competiciones

Entradas: Nombre, email, contraseña

Efecto: Crea un nuevo responsable de competiciones

Restricciones: Ninguna

- ***Dar de alta una competición por puntos***

Descripción: Crea una competición por puntos

Entradas: Nombre, fecha inicio, fecha fin, núm. Jugadores por equipo, Responsable de Competiciones, puntos al ganar, puntos al perder, puntos al empatar

Efecto: Crea una nueva competición por puntos, y le asigna un responsable de competiciones. La competición se queda abierta para poder inscribir equipos y partidas.

Restricciones: Ninguna

- ***Dar de alta una competición por posiciones***

Descripción: Crea una competición por posiciones

Entradas: Nombre, fecha inicio, fecha fin, núm. Jugadores por equipo, Responsable de Competiciones, posiciones al ganar, posiciones al no ganar

Efecto: Crea una nueva competición por posición, y le asigna un responsable de competiciones. La competición se queda abierta para poder inscribir equipos y partidas.

Restricciones: Ninguna

- ***Cerrar inscripción de una competición***

Descripción: Cierra la inscripción de una competición

Entradas: Competición

Efecto: Deja la competición cerrada, para que no puedan inscribirse más equipos o partidos en ella

Restricciones: La Competición debe estar abierta

- ***Inscribir un equipo en una competición***

Descripción: Inscribir a un equipo en una competición

Entradas: Equipo, Competición

Efecto: Se añade el equipo a la competición. Si la competición es por puntos, se marca una clasificación inicial a 0 puntos. Si la competición es por ranking, se marca como clasificación inicial un nuevo ranking (el último, más bajo).

Restricciones: La competición debe estar abierta para inscripciones, y no haberse superado la fecha fin de la competición en el momento de la

inscripción. Además, el número de jugadores del equipo debe coincidir con el número de jugadores de la competición. Este equipo no debe estar ya en esta competición.

- **Crear un partido en una competición**

Descripción: Crea un partido en una competición entre dos equipos

Entradas: Fecha, hora, lugar, Competición, Equipo Local, Equipo Visitante

Efecto: Se crea un equipo entre los dos equipos en esta competición

Restricciones: La competición debe estar abierta para inscripciones, y no haberse superado la fecha fin de la competición en el momento de la inscripción. Los dos equipos deben pertenecer a la competición indicada, y deben ser dos equipos diferentes.

- **Introducir resultado de un partido**

Descripción: Marca un partido como jugado, y provoca la actualización de la clasificación

Entradas: Partida, marcador, empate, Equipo Ganador

Efecto: Marca la partida como jugada, establece el marcador y en función de éste, indica si hay un empate (booleano) o el Equipo Ganador. Esta operación desencadenará la actualización de la clasificación de la competición.

Restricciones: Se debe cumplir que (empate=false AND Equipo Ganador) ó (empate=true), es decir, o hay empate, o se indica el equipo ganador. El Equipo Ganador debe ser uno de los dos equipos del Partido (Equipo Local ó Equipo Visitante).

- **Actualizar la clasificación tras un partido jugado**

Descripción: Calcula la siguiente clasificación de una competición tras jugar un partido

Entradas: Partido Jugado

Efecto: En función del tipo de competición, se actualiza la clasificación de los equipos que han jugado el partido. Para una descripción completa, ver la sección “Las competiciones” en la Introducción de este documento.

Restricciones: La clasificación final de todos los equipos de la competición donde se ha jugado el partido debe ser coherente.

- **Asignar créditos por actividades deportivas a un jugador**

Descripción: Asigna un número de créditos indicado a un jugador.

Entradas: Jugador, número de créditos.

Efecto: Se le incrementarán los créditos acumulados al jugador.

Restricciones: Ninguna.

- **Créditos por ganar una competición oficial a un equipo**

Descripción: Asigna un número de créditos indicado por la competición a todos los jugadores del equipo indicado.

Entradas: Equipo, Competición.

Efecto: Se le incrementarán los créditos acumulados a cada jugador del equipo.

Restricciones: La Competición ha de ser oficial.

2.3 Requisitos de Interfaz de Usuario

2.3.1 Usuarios Anónimos

Estos usuarios podrán ver información sobre los deportes en los que se desarrollan competiciones, las competiciones disponibles (las que están activas), y las clasificaciones actuales en cada competición.

2.3.2 Usuarios Jugadores

Estos usuarios podrán gestionar sus equipos (crear nuevo equipo, añadir jugadores, definir delegado), y visualizar las posiciones/clasificaciones actuales que ocupan en sus equipos en las diferentes competiciones.

Tras seleccionar una competición en la que se está inscrito, se podrá observar el calendario de partidos jugados y los que quedan por jugar. Se podrá seleccionar cualquiera de ellos para ver más detalles. En caso de los partidos por jugar, se podrá además introducir el resultado, para indicar que se ha jugado el partido.

Además, se debe permitir que los jugadores busquen equipos contrincantes en competiciones de tipo ranking, para crear partidos con ellos. Se recomienda separar la gestión de competiciones por tipo (puntos/posiciones) para facilitar a los jugadores gestionar su participación en éstas por medio de los equipos.

2.3.3 Usuarios Responsables de Competiciones

Estos usuarios son los responsables de dar de alta los deportes y las competiciones, crear el calendario de partidos (dar de alta partidos) para las

competiciones por puntos, y cerrar las competiciones por puntos para que no se puedan inscribir más equipos.

Las competiciones por posiciones las dejarán abiertas hasta que finalice la competición para permitir la inscripción de nuevos equipos y partidos. Deben poder ver en cualquier momento la clasificación de cada competición para sacar reportes.

3. Modelación de la web

Para desarrollar aplicaciones web de calidad, surge la Ingeniería Web, que es la rama de la Ingeniería del Software que estudia los procesos, métodos, técnicas y recursos esenciales para el desarrollo de aplicaciones web de calidad.

La Ingeniería Web propone: El desarrollo a partir de modelos conceptuales de la web y el establecimiento de procesos de desarrollo adecuados, proporcionando una visión de la web basada en las siguientes características:

- **La gestión de servicios y contenidos:**

Los Modelos Conceptuales Web deben permitir describir:

- La visibilidad sobre la información.
- La accesibilidad a la ejecución de servicios, de nuestro sistema software.

- **Estructuras de Navegación**

Los Modelos Conceptuales Web deben estructurar la accesibilidad de cada usuario a los diferentes contenidos y servicios.

- **Guías de Presentación**

Los Modelos Conceptuales Web deben recoger las pautas de presentación de contenidos, servicios, estructuras de acceso, etc.

- **Gestión de Usuarios y políticas de personalización**

Los Modelos Conceptuales Web deben incorporar los usuarios del sistema y categorizarlos además de definir reglas de adaptación de la aplicación a las necesidades de cada uno de ellos.

En el proyecto se ha adoptado los conceptos que propone el método OOWS (*Object-Oriented Web Solutions*) -es un método orientado a objetos para programación web- para modelar nuestra aplicación web y definir el proceso de desarrollo que vamos a seguir, dividiendo el proceso de desarrollo en dos grandes pasos: Especificación conceptual de la aplicación: usando modelos conceptuales OOWS para describir las características de nuestra página web y Desarrollo de la solución final con ASP.NET, C#, JavaScript y PhotoShop.

3.1 Modelo de Navegación

El modelo de navegación especifica las características navegacionales de las aplicaciones web (acceso a información y ejecución de servicios) y define la estructura de los diferentes tipos de usuarios y como estos usuarios acceden al sistema. Siguiendo este modelo, hemos diseñado el **Diagrama de Usuarios** y los **Mapas Navegacionales** (especifican las propiedades navegacionales y proporciona una vista global del sistema para cada tipo de usuario tanto como definen la estructura del *Web Site*).

Diagrama UML de clases.

El sistema total está representado en el siguiente diagrama de clases (Según el Modelo UML: *Unified Modeling Lenguaje*).

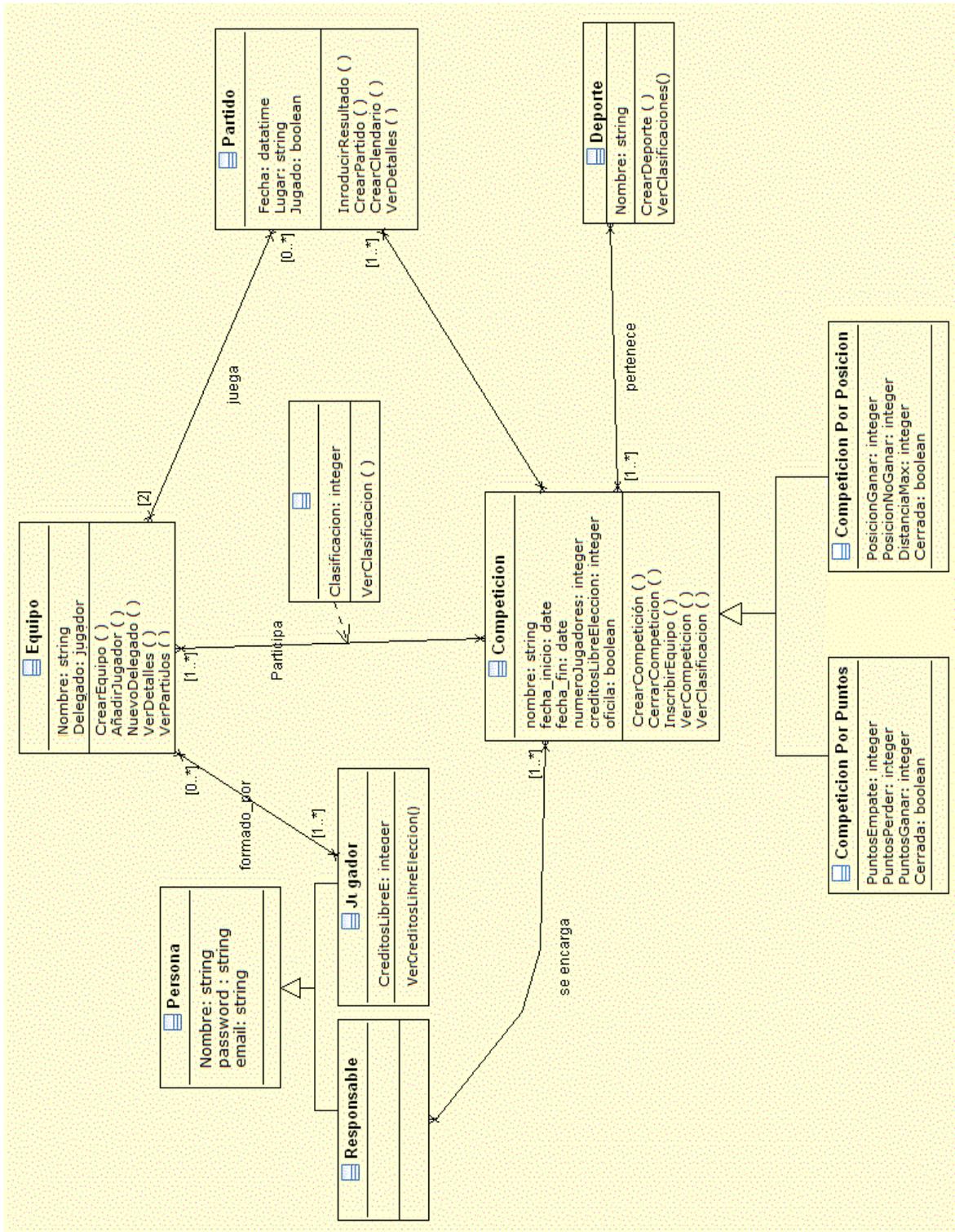


Ilustración 3: Diagrama de Clases (UML) del sistema GCD.

3.1.1 Diagrama de Usuarios

En el proyecto, se ha detectado tres tipos de usuarios: los anónimos, los jugadores y los responsables de las competiciones, en función del tipo de acceso que tengan con el sistema.

- **Los anónimos:** Estos usuarios al conectarse al sistema no necesitan identificarse y sus permisos son muy reducidos.
- **Los jugadores:** Este tipo de usuarios necesitan identificarse al conectarse al sistema, tendrán acceso a más zonas y funcionalidades del sistema que los usuarios anónimos.
- **Los responsables de las competiciones:** Son usuarios de tipo registrado también pero tienen más privilegios que los jugadores, se encargan de gestionar el sistema y mantenerlo (por ej. dando de altas, eliminando o/ gestionando competiciones, deportes, calendarios de partidos, etc.).

La ilustración (4), muestra el diagrama de usuarios del sistema GCD, y como están organizados dichos usuarios.

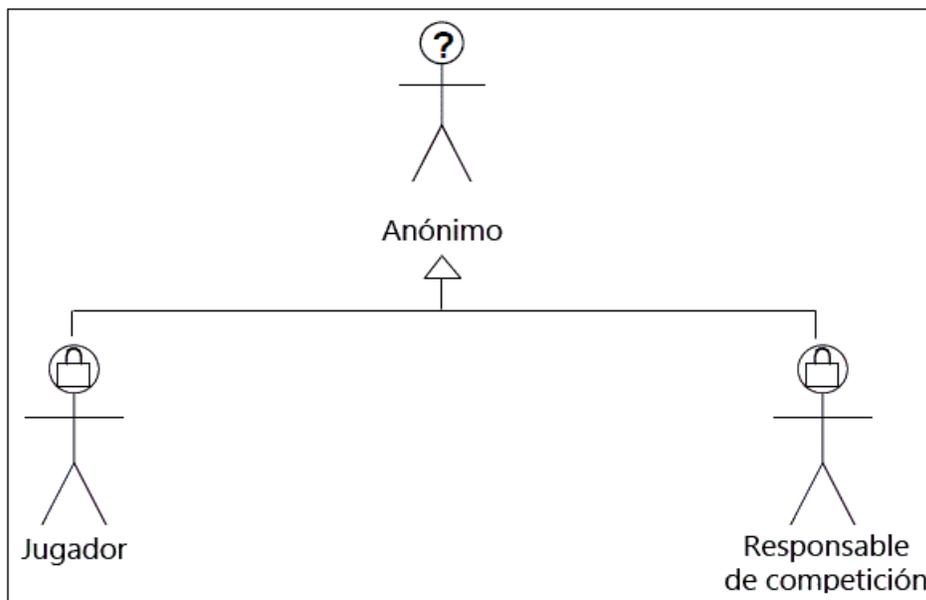


Ilustración 4: Diagrama de usuarios del GCD.

3.1.2 Mapas Navegacionales.

3.1.2.1 Mapa de Navegación del usuario Anónimo.

La ilustración (5), muestra el mapa de navegación del usuario *Anónimo* tal y como ha sido diseñada nuestra aplicación de gestión de competiciones deportivas GCD.

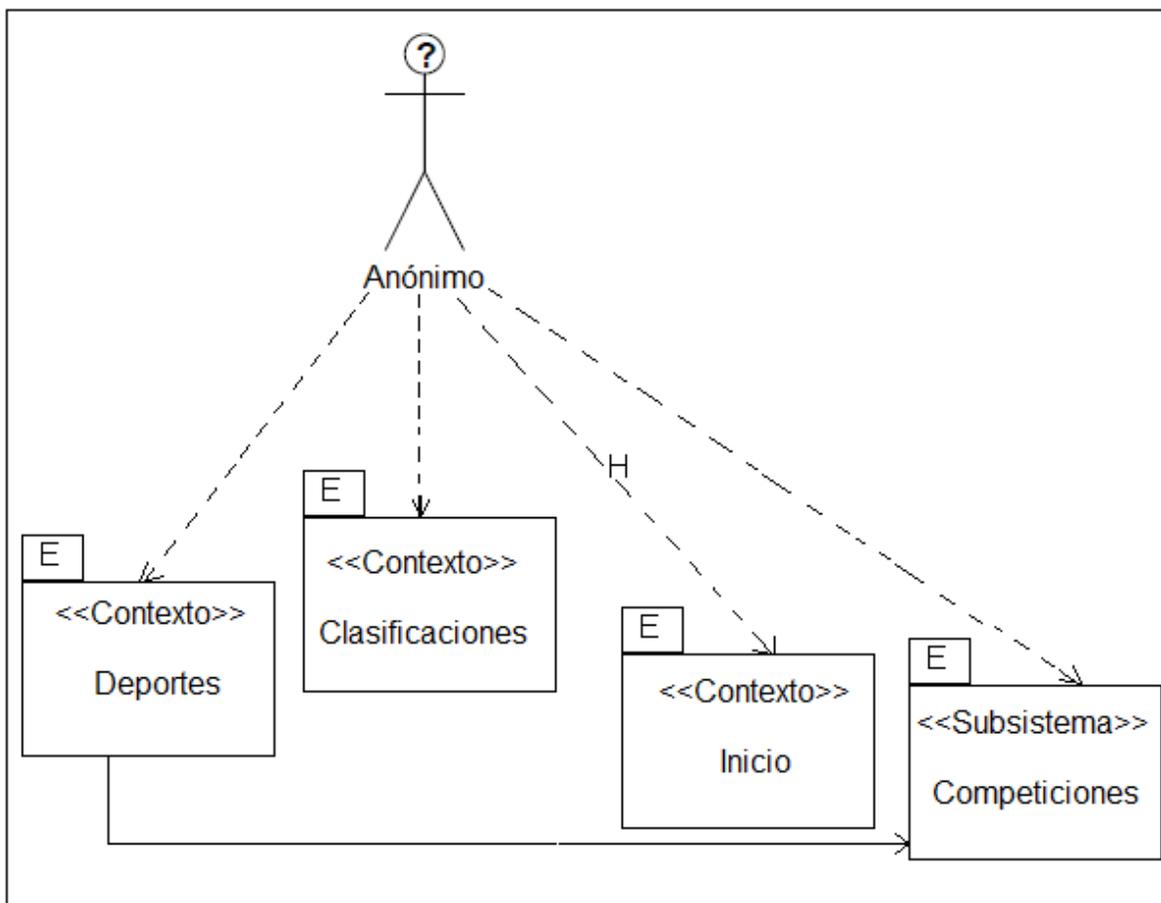


Ilustración 5: Mapa de Navegación del usuario Anónimo.

El usuario *Anónimo* tiene acceso a cuatro zonas de la aplicación: la zona del *Inicio*, *Deportes*, *Clasificaciones* y al subsistema *Competiciones* que está formada por dos módulos diferentes: *Competiciones por puntos* y *Competiciones por posición*, la ilustración (6) muestra como está dividido el subsistema *Competiciones*.

El usuario *Anónimo* dentro del subsistema *Competiciones* tiene acceso tanto al contexto de navegación de exploración *Competiciones por puntos* como al contexto de navegación de exploración *Competiciones por posiciones*, y de allí se puede visualizar los detalles de las clasificaciones de todas las competencias de nuestra bases de datos, este concepto está representado en la ilustración (6) por el contexto de secuencia Clasificaciones-Competición.

El usuario *Anónimo* cuando se conecta a la página web (*GCD: Gestión de competencias deportivas*) se dirige directamente a la contexto inicio (la página de Inicio), es lo que refleja la letra "H" de la ilustración (5).

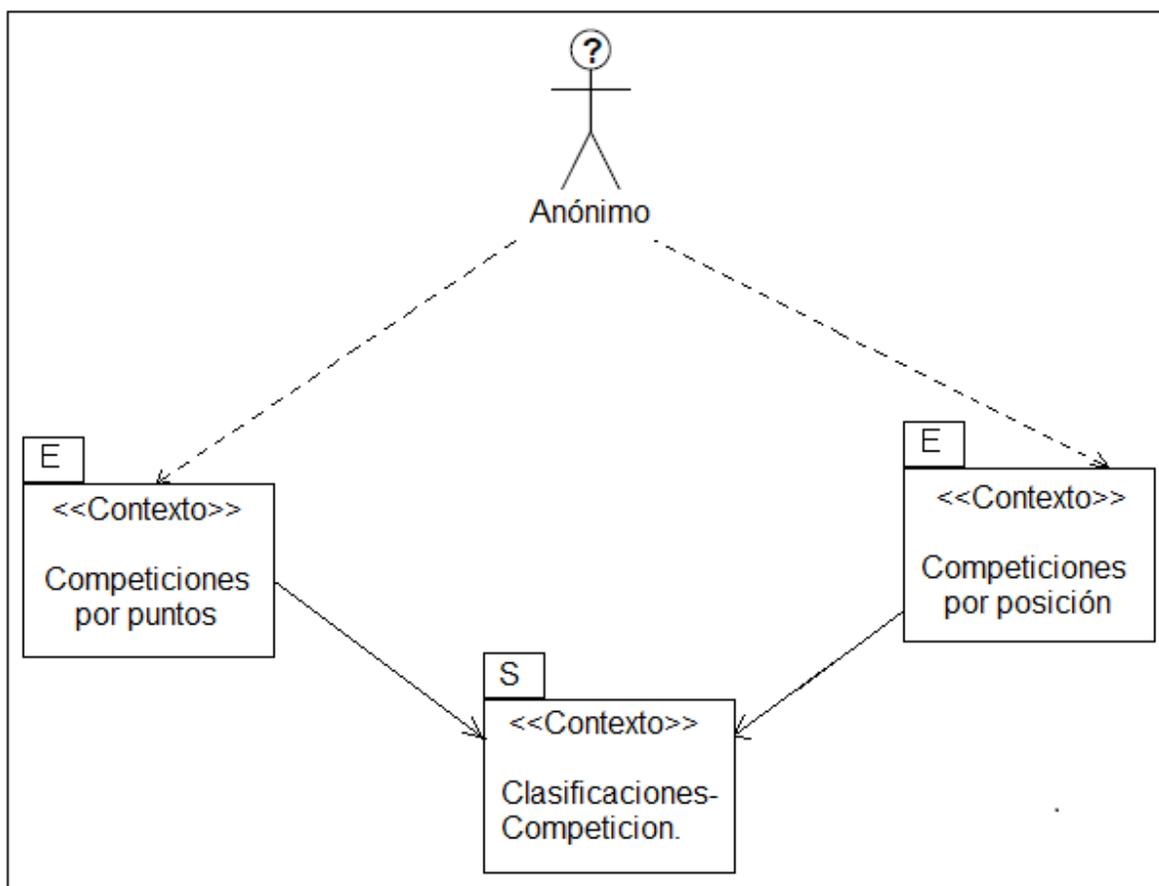


Ilustración 6: Subsistema Clasificaciones-competición (usuario Anónimo).

El usuario *Anónimo* al acceder al sistema se conecta directamente a la página *Inicio*, a partir de la cual accede a los Deportes, Competiciones (tanto a las Competiciones por Puntos como a las Competiciones por Posición) y Clasificaciones.

3.1.2.2 Mapa de Navegación del usuario Jugador.

La ilustración (7), muestra el mapa de navegación del usuario *Jugador* tal y como ha sido diseñada nuestra aplicación de gestión de competiciones deportivas GCD.

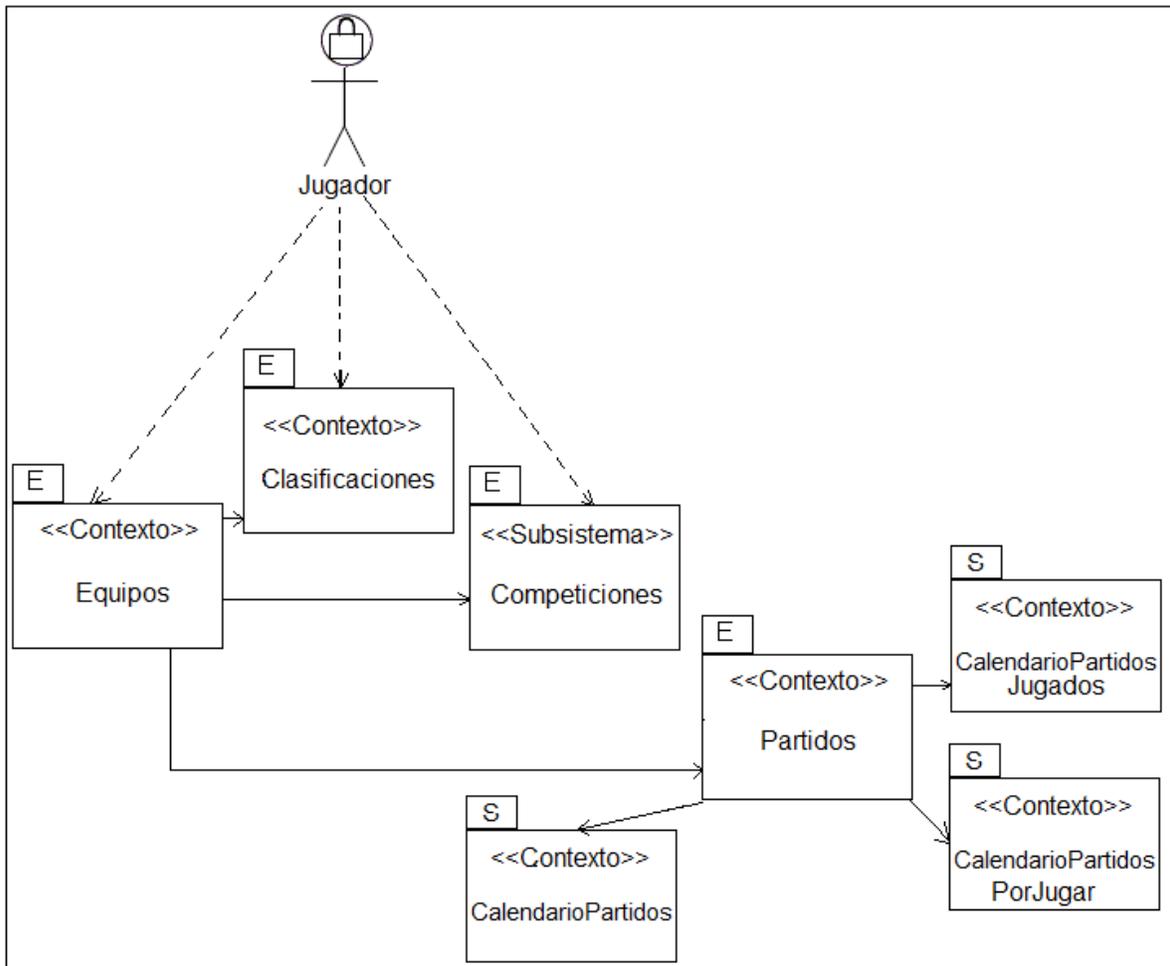


Ilustración 7: Mapa de Navegación del usuario Jugador.

El usuario *Jugador* tiene acceso a varias zonas de la aplicación; accede a zonas y servicios especiales a que no puede acceder el usuario Anónimo.

El usuario *Jugador* accede a los contextos Navegacionales de exploración *Equipos*, *Clasificaciones*, *Partidos* y el subsistema *Competiciones* desde el menú

principal que aparece en la página de *Inicio* del usuario *Jugador*. Se puede acceder al mismo contexto de navegación desde varios caminos y cada camino de estos ofrece un servicio a medida en función de donde lo accedemos:

- ✓ Cuando accedemos a los contextos *Partidos* y *Clasificaciones* y al subsistema *Competiciones* desde el contexto de navegación *Equipos*, accedemos a los partidos, clasificaciones y competiciones (o bien competiciones por puntos o bien competiciones por posición) respectivamente de un equipo en concreto, el equipo que tenemos seleccionado, es un equipo en los cuales juega el usuario *Jugador* que está conectado al sistema
- ✓ Cuando accedemos al contexto *Clasificaciones* y el subsistema *Competiciones* desde el menú principal de la página de *Inicio* del usuario *Jugador*, accedemos a todas las clasificaciones y competiciones (o bien competiciones por puntos o bien competiciones por posición) del usuario *Jugador* conectado al sistema.

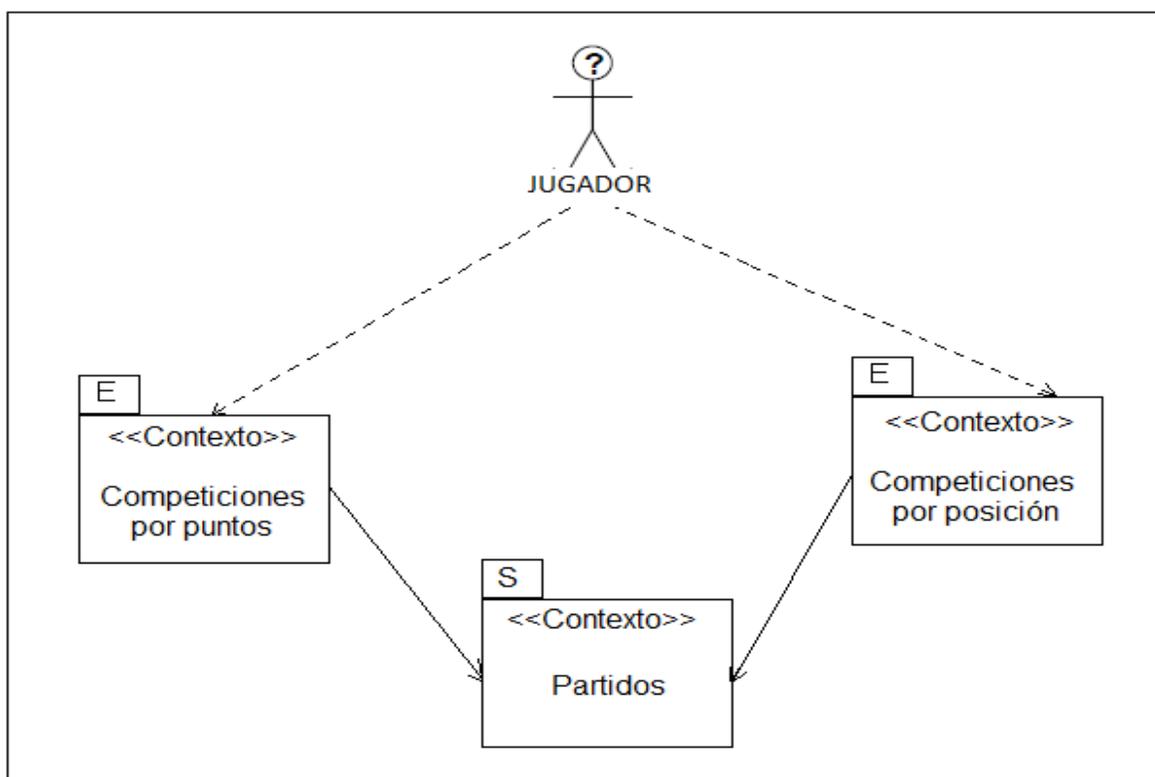


Ilustración 8: Subsistema Clasificaciones-competición (usuario Jugador).

La ilustración (8) muestra como está dividido el subsistema *Competiciones*. El usuario *Jugador* dentro del subsistema *Competiciones* tiene acceso tanto al

contexto navegacional *Competiciones por puntos* como al contexto navegacional *Competiciones por posiciones*, y desde allí se puede visualizar los detalles de los partidos de la competición que tiene seleccionada.

3.1.2.3 Mapa de Navegación del usuario Responsable de la competición.

La ilustración (9), muestra el mapa de navegación del usuario *Responsable de las competiciones* tal y como ha sido diseñada nuestra aplicación de gestión de competiciones deportivas GCD.

El usuario *Responsable de las competiciones* tiene acceso a varias zonas de la aplicación; aparte de que accede a zonas restringidas y ofrece servicios especiales que no están disponibles ni para el usuario *Anónimo* ni para el usuario *Jugador*, gestiona también las competiciones.

El usuario *Responsable de las competiciones* accede a varios contextos Navegacionales, en su caso aparecen todos como subsistemas; *Competiciones, Equipos, Partidos, Deportes, Clasificaciones y Otros*, organizados cada uno de los cuales como un bloque independiente con sus propios servicios.

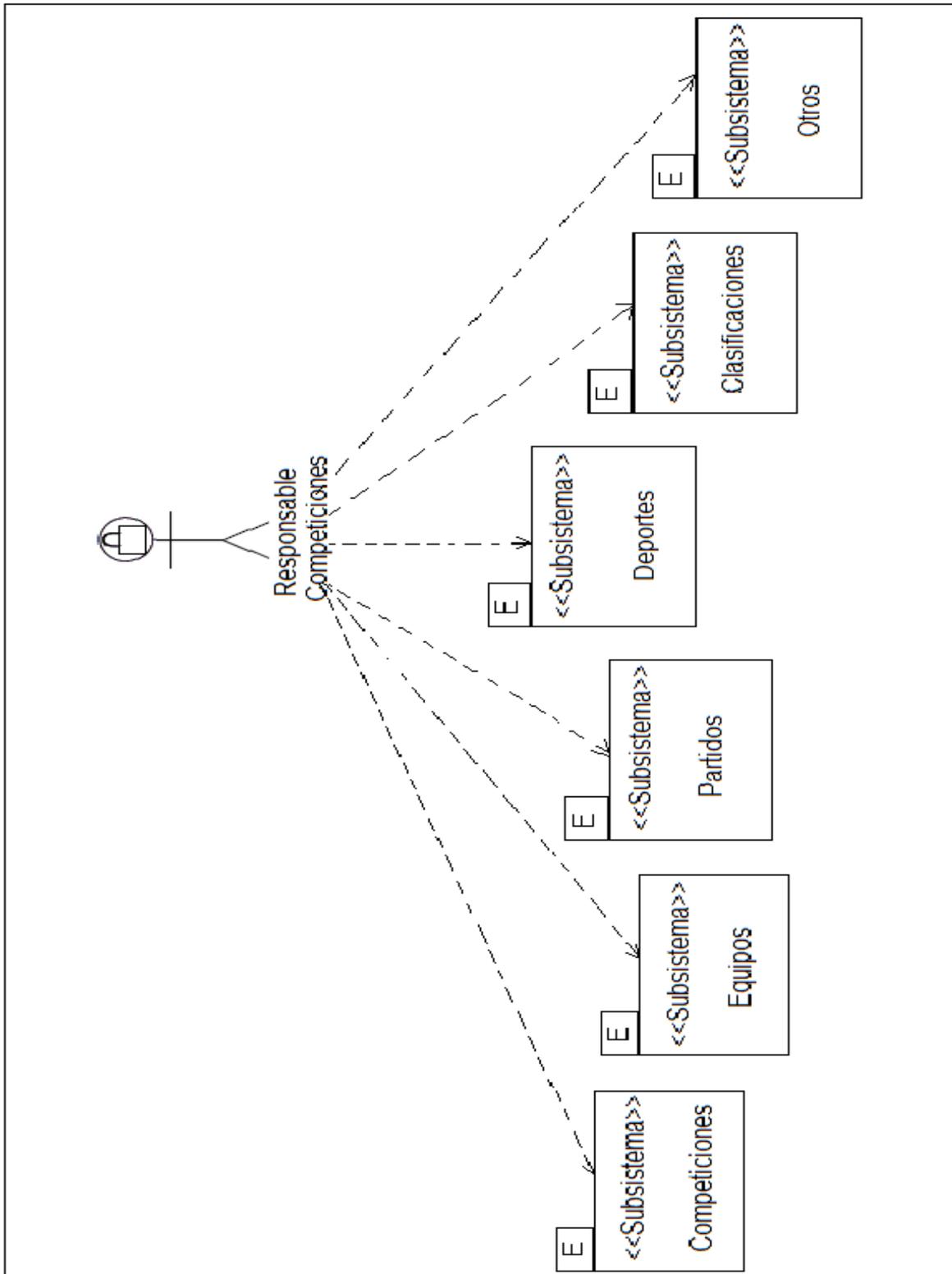


Ilustración 9. Mapa Navegacional del usuario Responsable de la Competición.

A continuación, en la ilustración 10 se muestra el subsistema de competiciones según como la ve el usuario *Responsable de Competiciones*, este subsistema define diferentes funciones que operan con los módulos de las *Competiciones por Puntos* y las *Competiciones por Contexto*, a partir del contexto *Competiciones* se puede llegar a las competiciones por puntos, competiciones por posiciones o clasificaciones y operar con sus componentes.

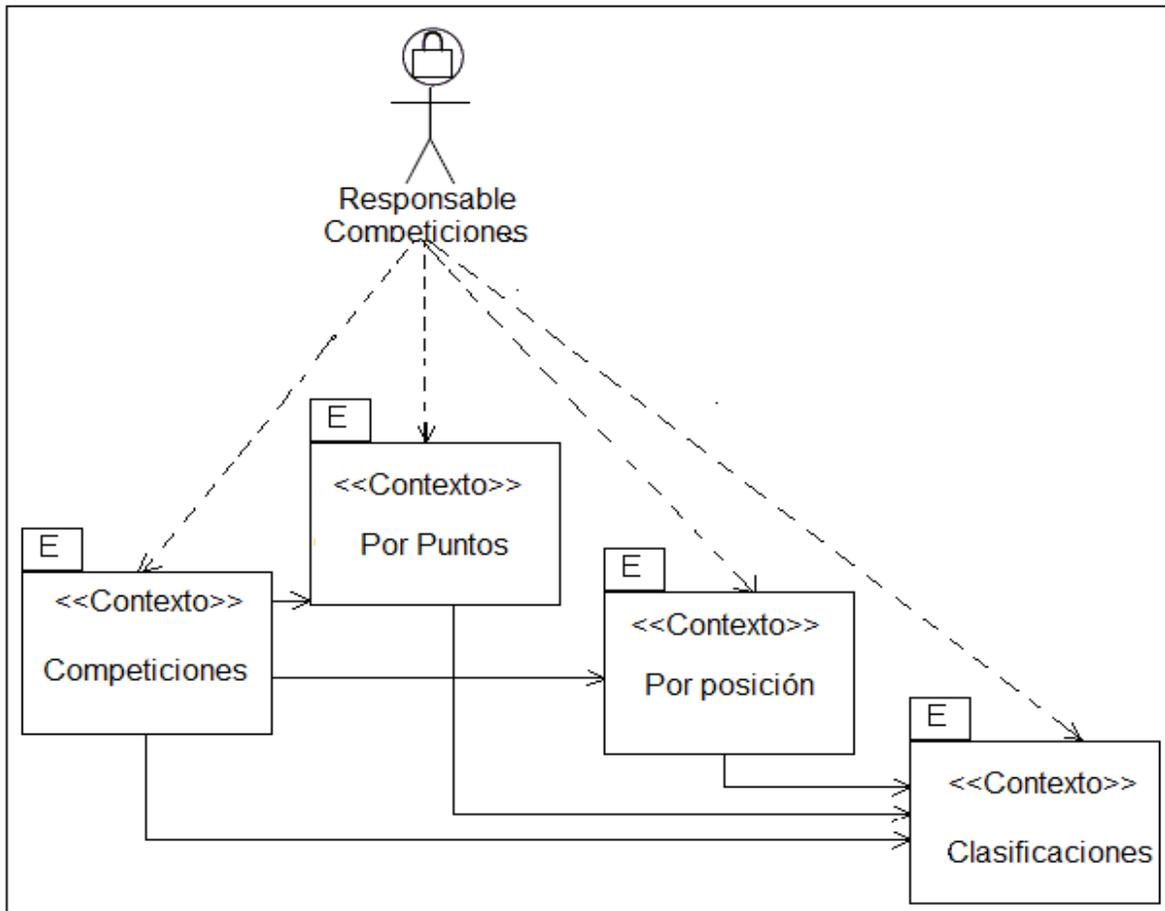


Ilustración 10: Detalle del Subsistema Competiciones (Responsable Competiciones).

La ilustración 11, muestra la vista del subsistema *Deportes* para el usuario *Responsable*.

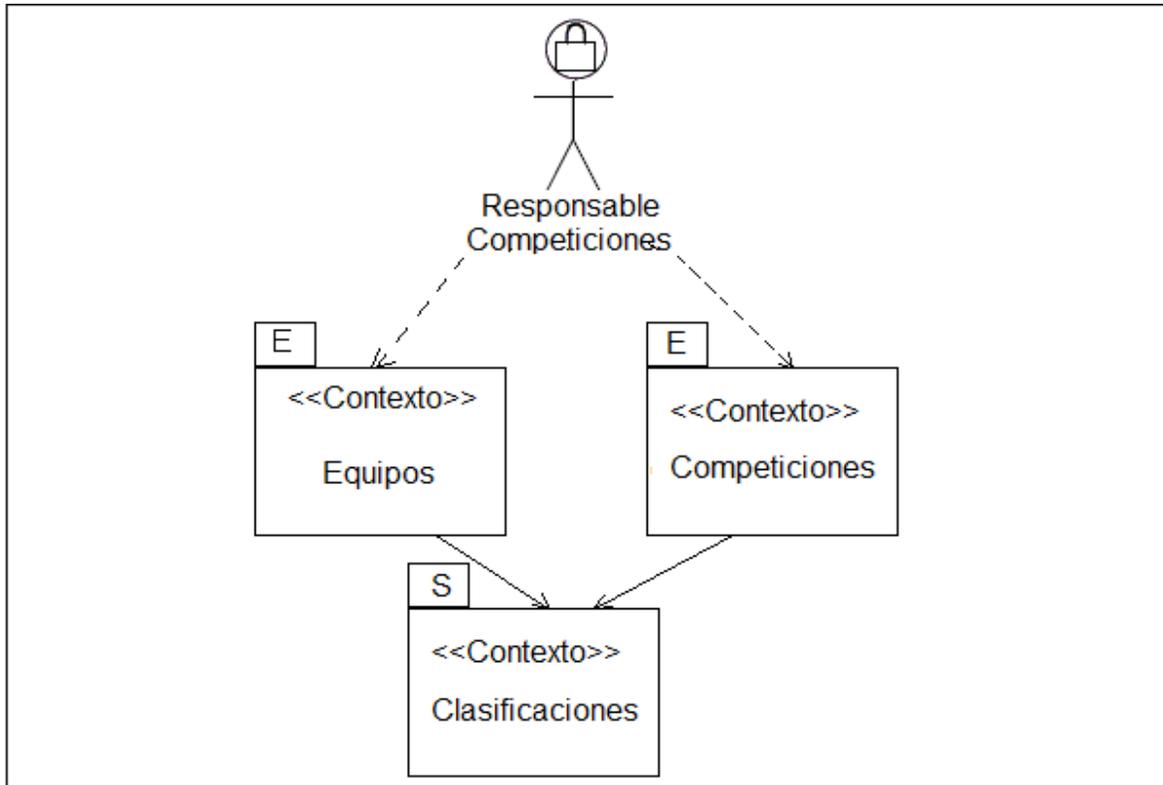


Ilustración 11: Detalle del Subsistema Deportes (Responsable Competiciones).

El Responsable de competición dentro de este subsistema siempre se explora al contexto Clasificaciones, partiendo tanto de las Competiciones como de Equipos.

La ilustración 12 muestra el subsistema del usuario *Responsable*, partiendo del modulo *Equipos* se explora tanto al detalle del equipo seleccionado como el calendario de partidos de dicho equipo.

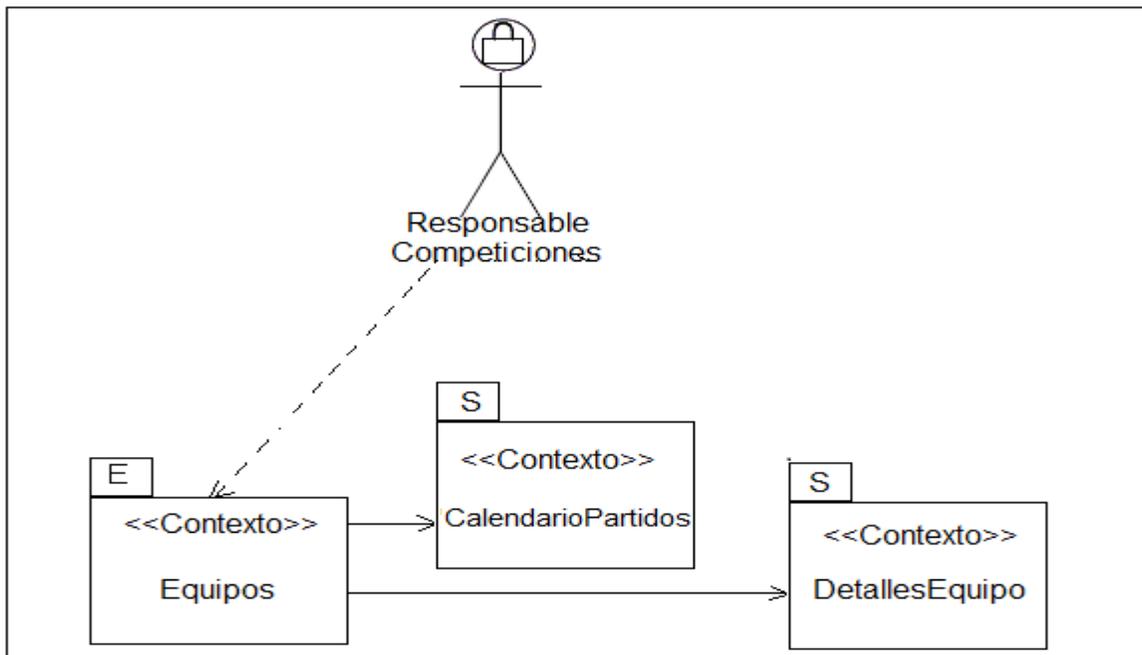


Ilustración 12: Detalle subsistema Equipos (Responsable de competencias)

El Responsable de la competición puede en cualquier momento acceder a los Partidos de un Equipo o una Competición (Ver la ilustración 13).

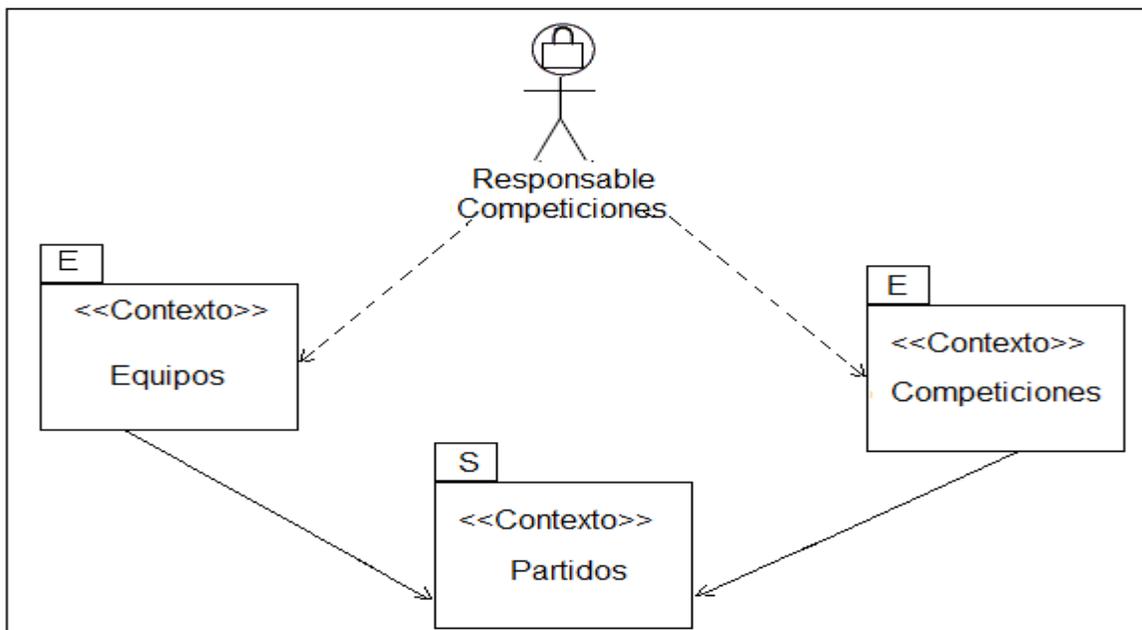


Ilustración 13: Detalles subsistema Partidos (Responsable de competencias)

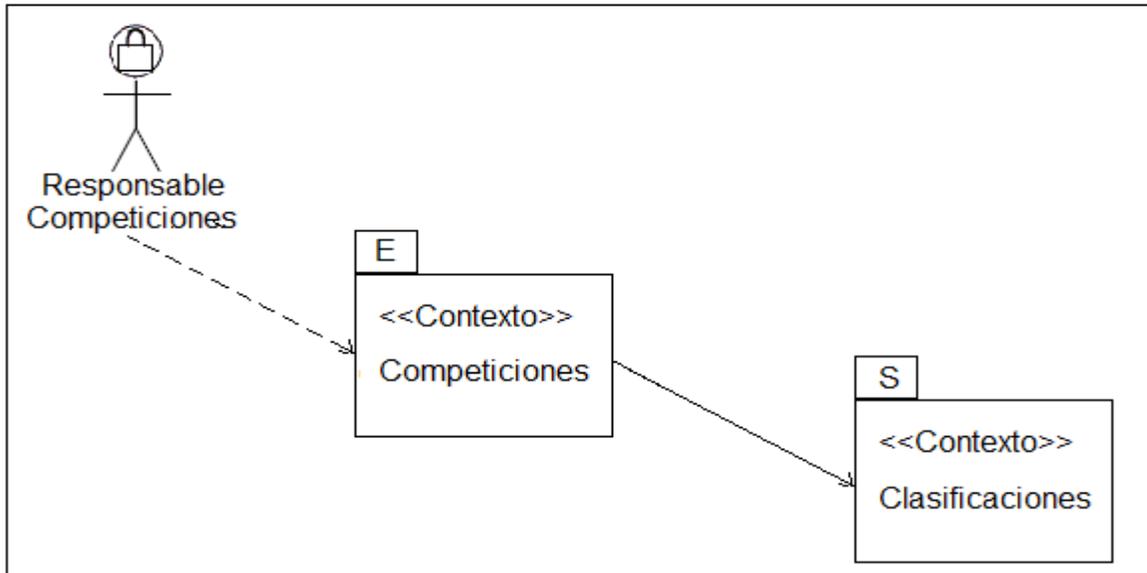


Ilustración 14: Detalle subsistema Clasificaciones (Responsable de competiciones)

La ilustración 14 muestra el detalle del subsistema Clasificaciones. El *Responsable* una vez esta en el modulo Clasificaciones puede acceder a una de ellas seleccionando una competición.

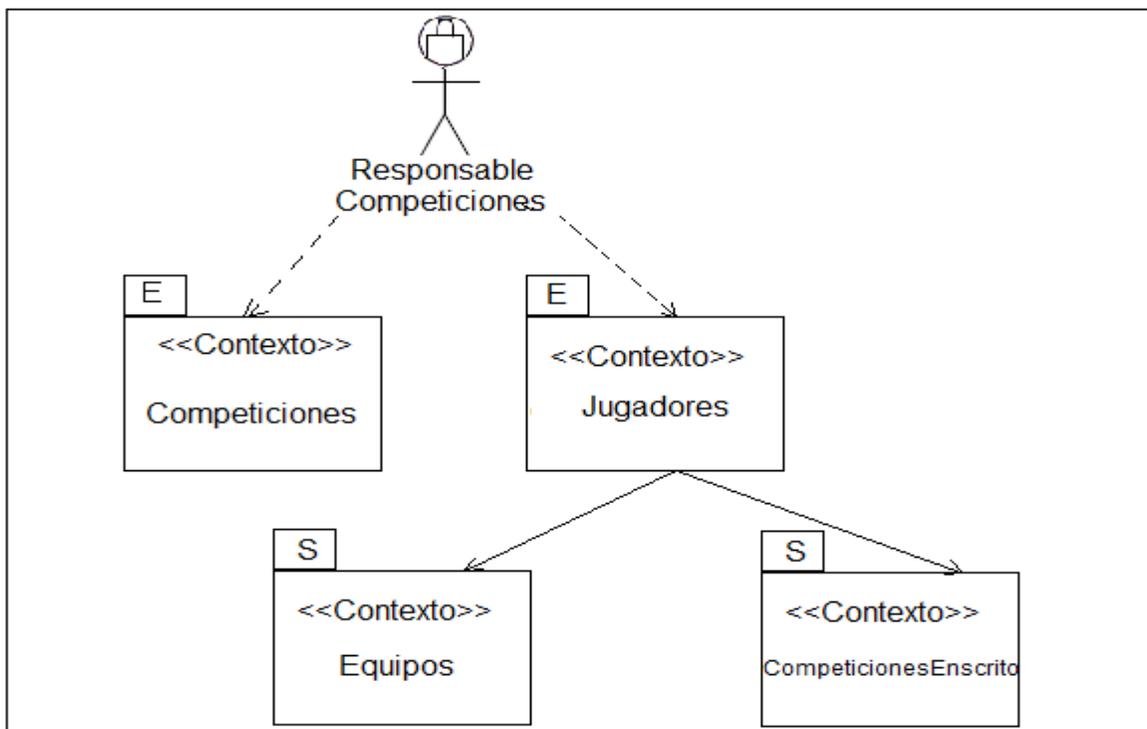


Ilustración 15: Detalle subsistema Otros (Responsable de competiciones)

En el subsistema Otros (ilustración 15), se ha juntado las funciones que no están definidas bajo ningún contexto relevante, contiene operaciones especiales.

A continuación se muestra los diagramas del contexto navegacional tanto del usuario Anónimo como los del usuario Responsable de las competencias el usuario Jugador:

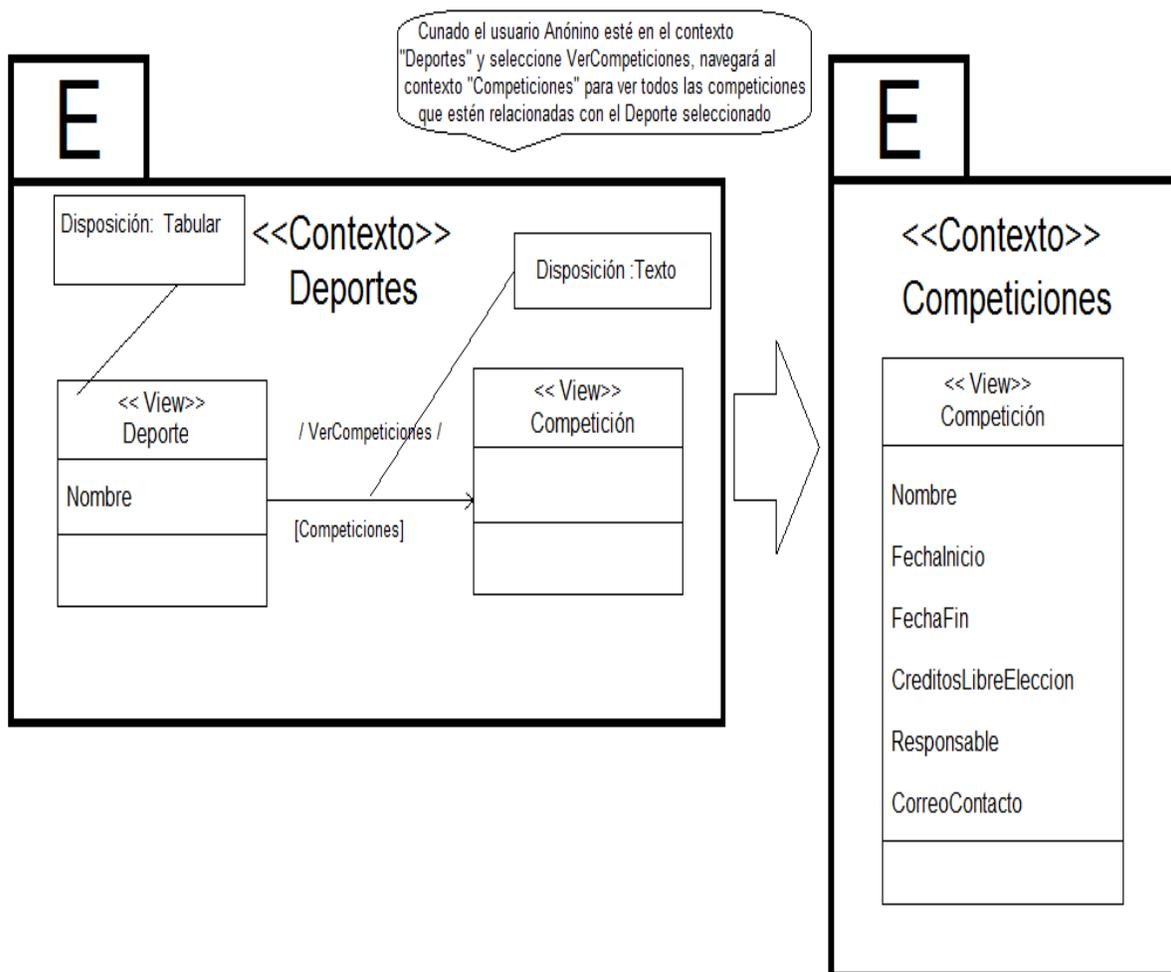


Ilustración 16: Relación de contexto del usuario anónimo.

Cuando el usuario Anónimo esté en el contexto *Deportes* y seleccione *VerCompeticiones*, navega al contexto *Competiciones* para ver todas las

competiciones que estén relacionadas con el *Deporte* seleccionado (Ver la ilustración 16).

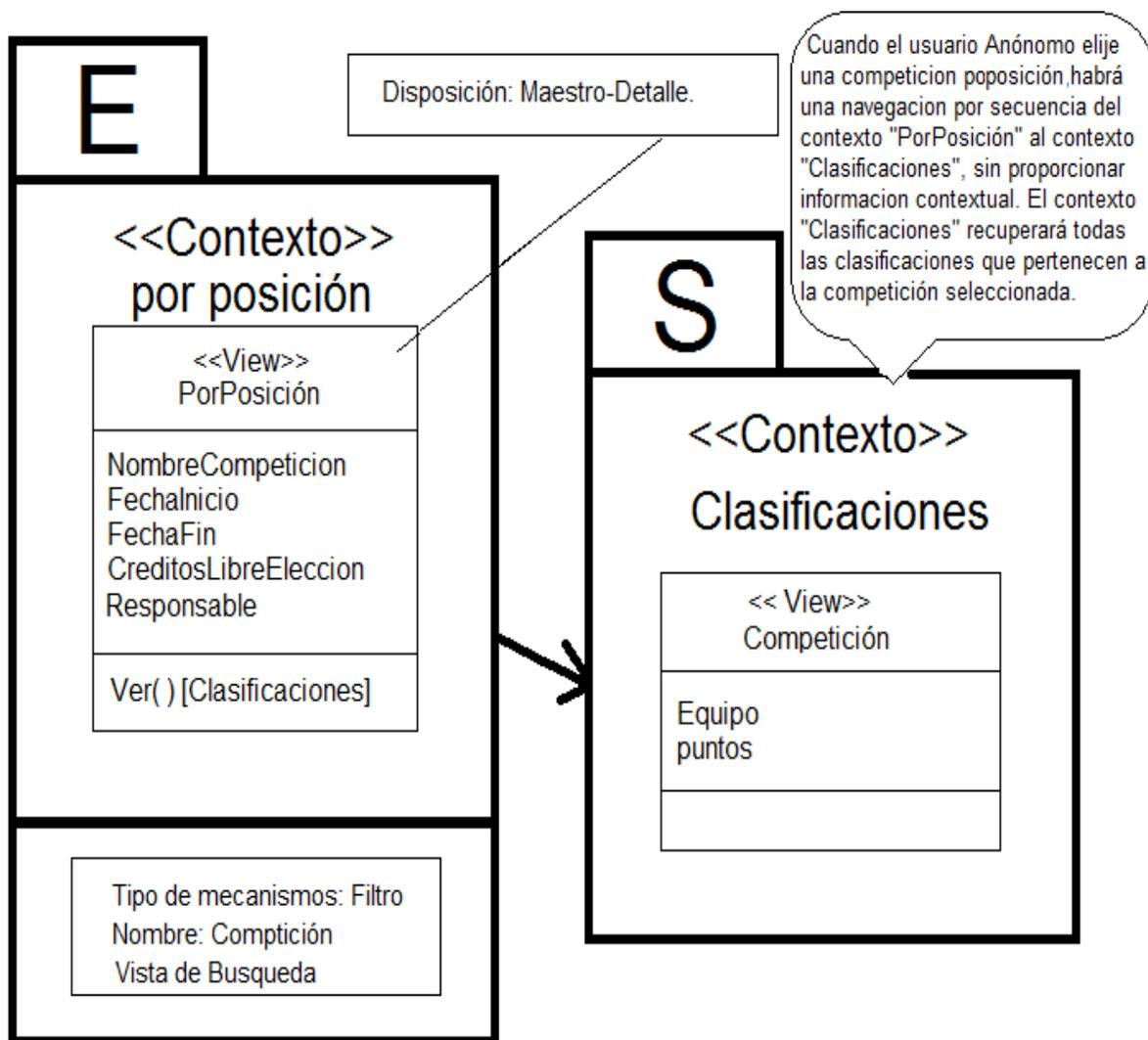


Ilustración 17: Enlace contextual de servicio de competiciones por posición del usuario Anónimo.

La ilustración 17, muestra que el usuario *Anónimo* cuando elije una competición por posición, habrá una navegación por secuencia del contexto *PorPosición* al contexto *Clasificaciones* recuperará todas las clasificaciones que pertenecen a la competición seleccionada.

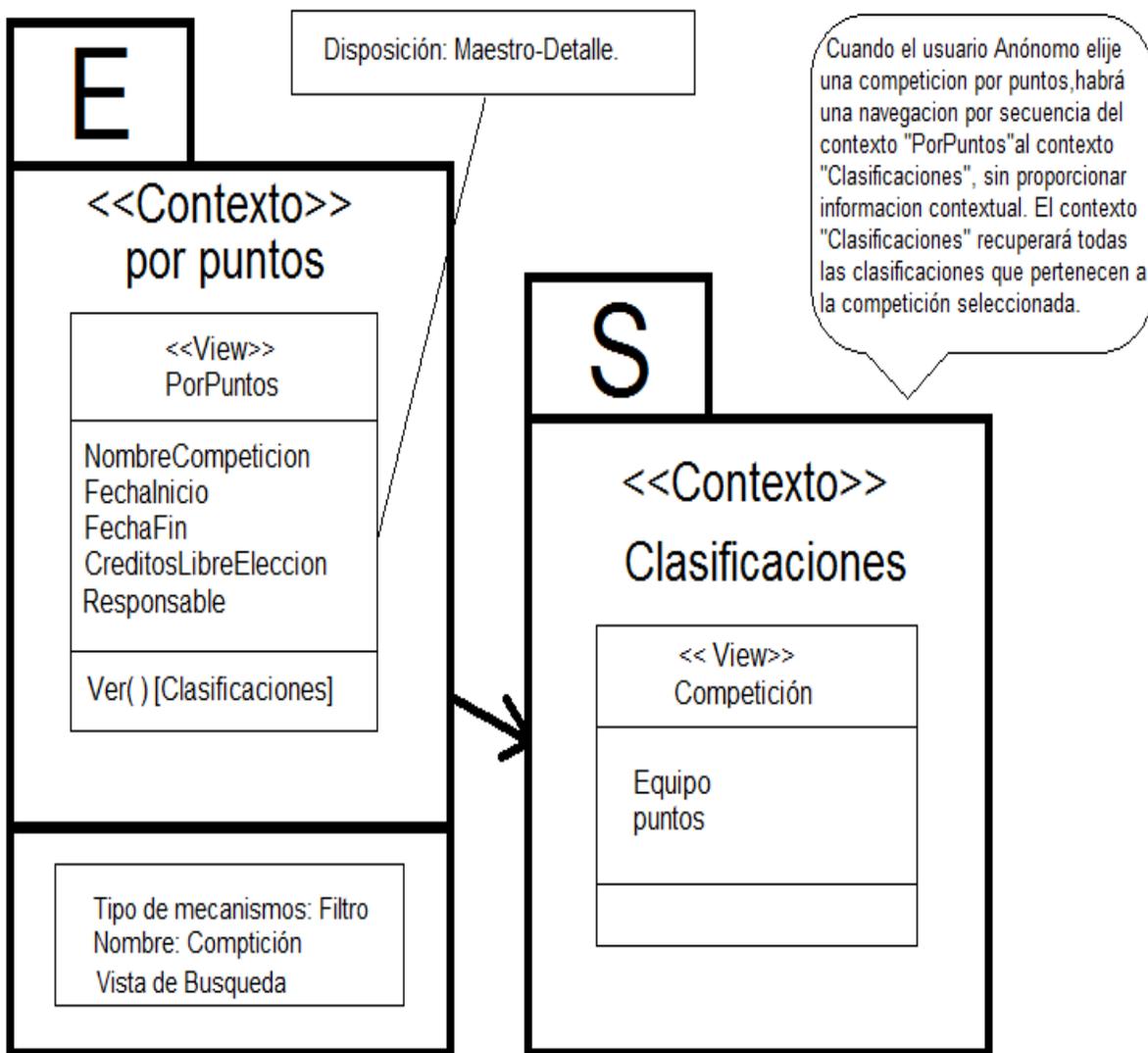


Ilustración 18: Enlace contextual de servicio de competiciones por puntos del usuario Anónimo.

La ilustración 18 muestra que el usuario *Anónimo* elije una competición por puntos, habrá una navegación por secuencia del contexto *PorPuntos* al contexto *Clasificaciones* sin proporcionar información contextual. El contenido *Clasificaciones* recuperará todas las clasificaciones que pertenecen a la competición seleccionada (filtra por la competición tal y como se muestra en la ilustración 18).

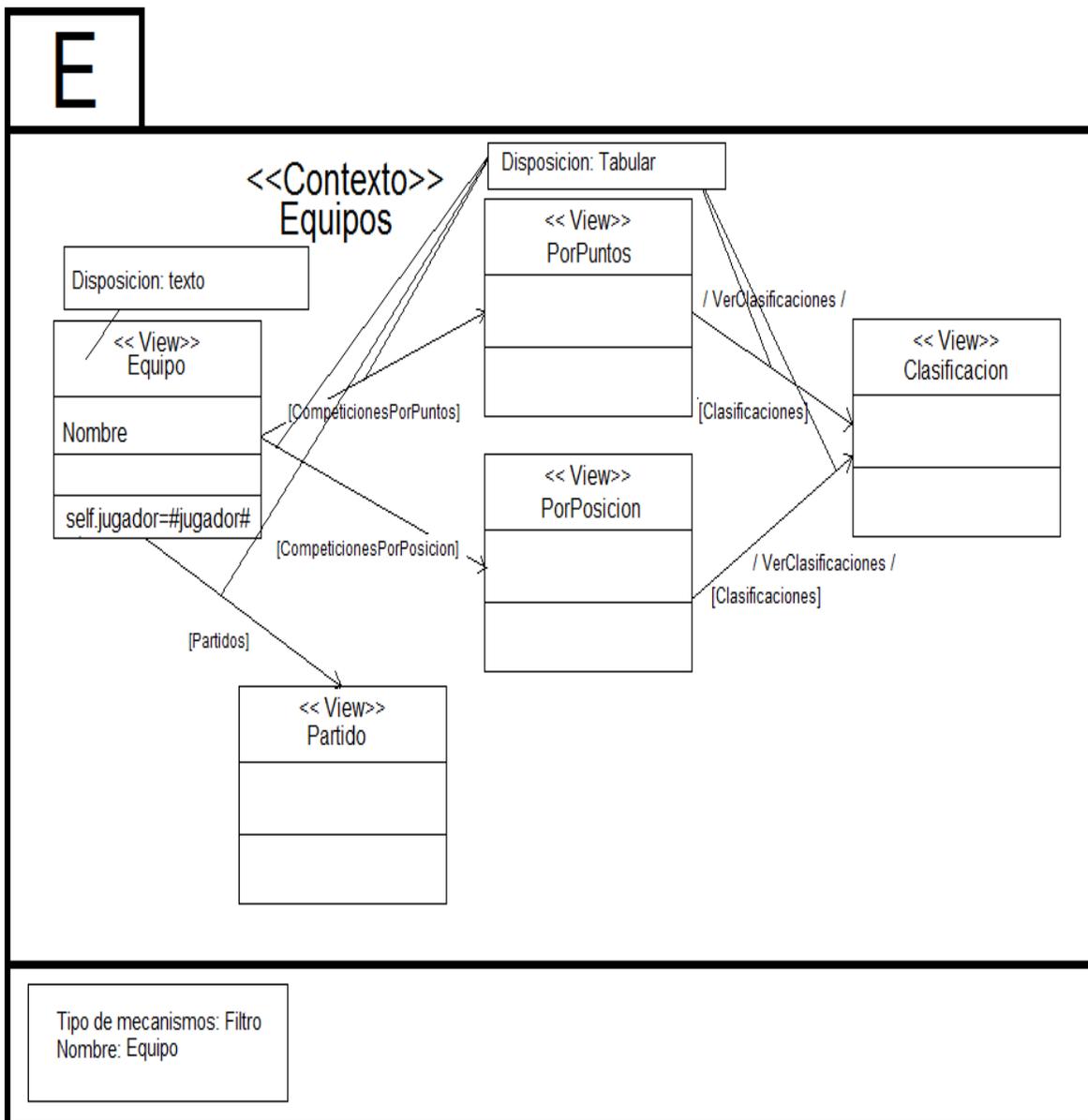


Ilustración 19: Contexto Equipo del usuario Jugador.

La ilustración 19, muestra el contexto *Equipo* del usuario *Jugador*. Desde la página *Equipo* se consultan las competiciones por puntos, competiciones por posición, partido y clasificaciones. Las competiciones y los partidos se filtran por el nombre del Equipo, y las clasificaciones se filtran por el nombre de la competición que por su vez ha sido filtrado por el nombre del Equipo. el resultado de las consultas se muestra en tablas.

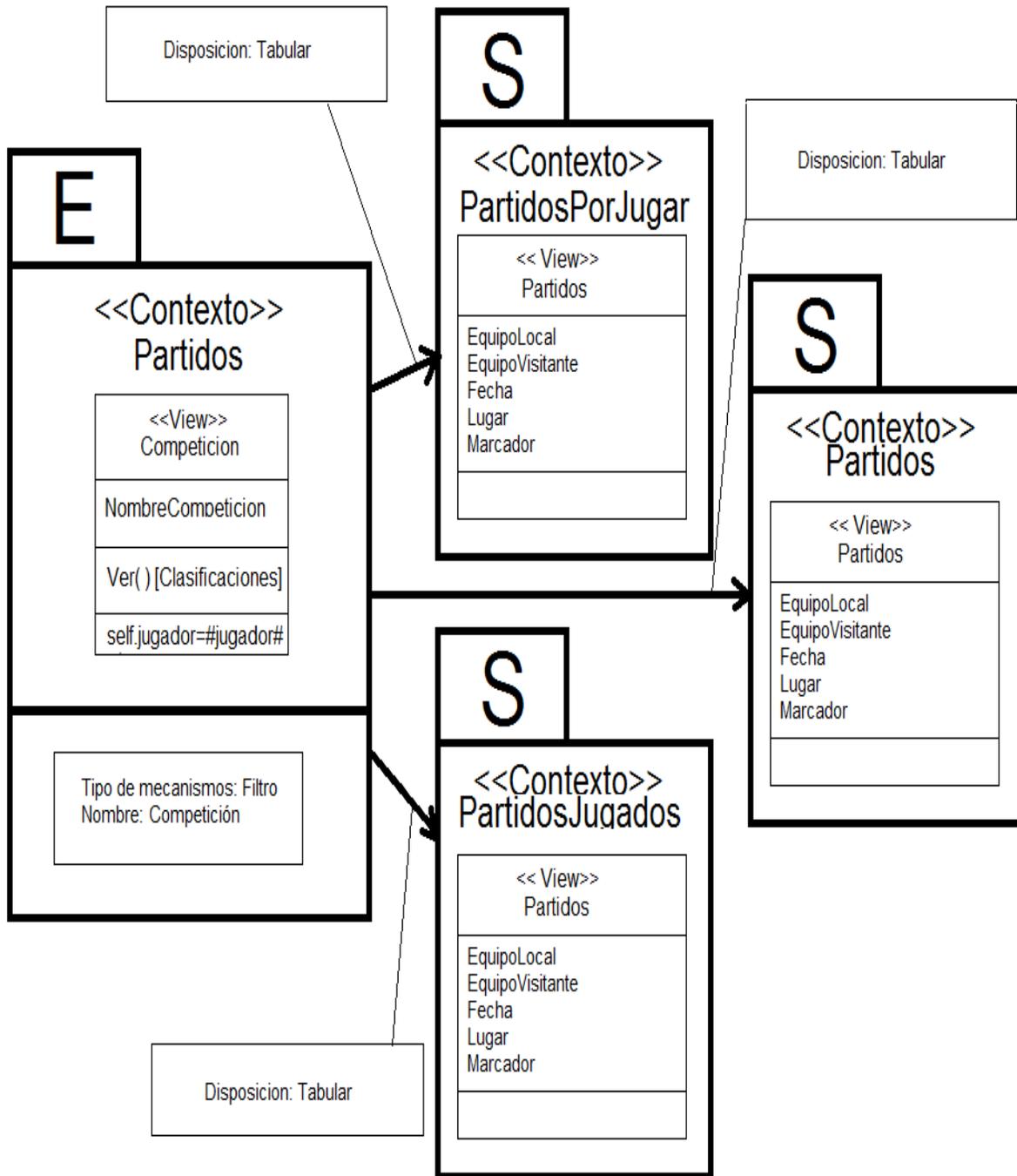


Ilustración 20: Enlace contextual de servicio del contexto Partidos del usuario Jugador.

La ilustración 20 muestra el enlace contextual de servicio *Partidos* del usuario *Jugador*, filtrando por el nombre de la competición. Desde dicho contexto se puede

explorar tanto los partidos jugados como los partidos que faltan por jugar de la competición seleccionada ordenando la información de dichos partidos en tabla (Equipo local, equipo visitante, la fecha del partido, el lugar y el marcador).

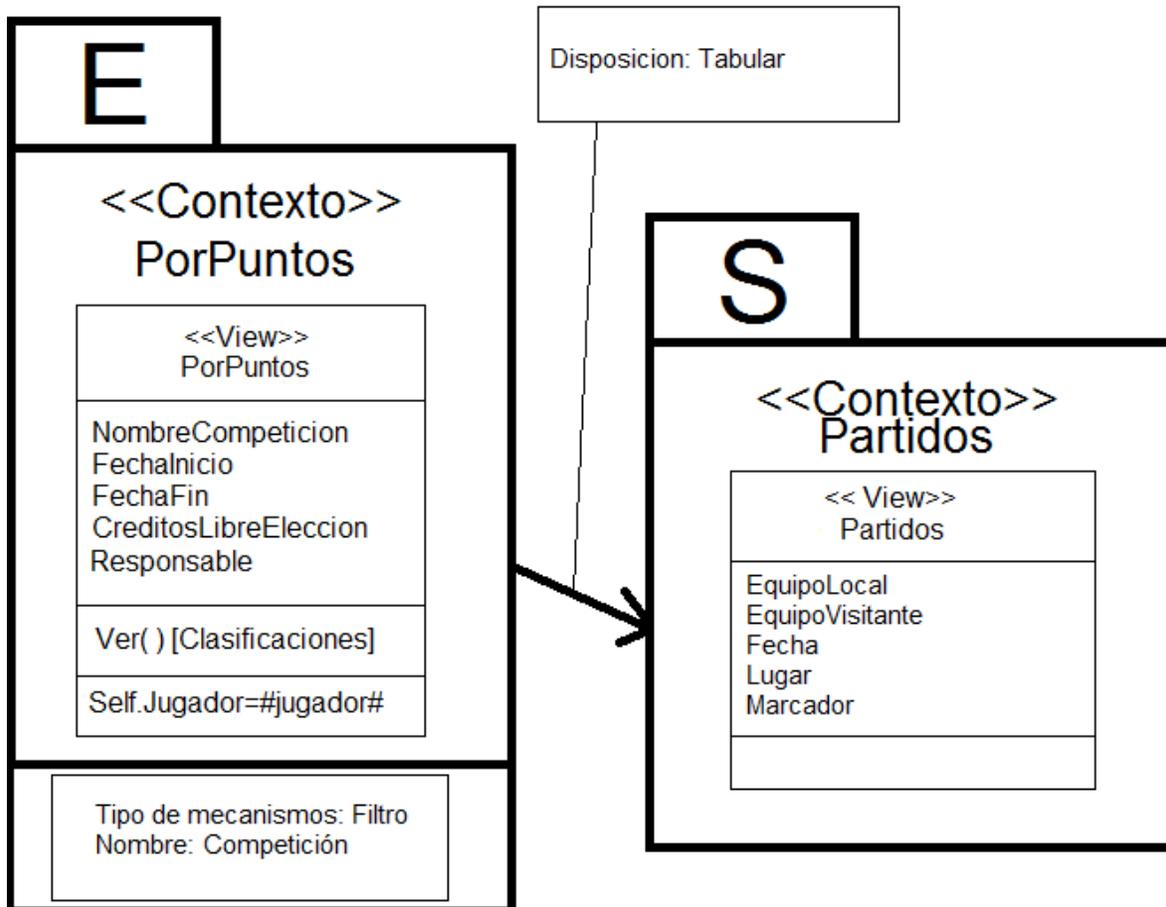


Ilustración 21: Enlace contextual de servicio del contexto Competiciones por puntos del usuario Jugador.

En la ilustración 21, se muestra el enlace contextual de servicio del contexto *Competiciones por puntos* del usuario *Jugador*;

La instrucción *Self.Jugador = #jugador#* : Significa que cuando el usuario *jugador* selecciona *ver()* que aparece en la esquema del contexto *Conexión por puntos* solo visualiza las competiciones donde el jugador está inscrito mostrando

datos de los partidos de dicha competición (Equipo local, equipo visitante, fecha, lugar y el marcador).

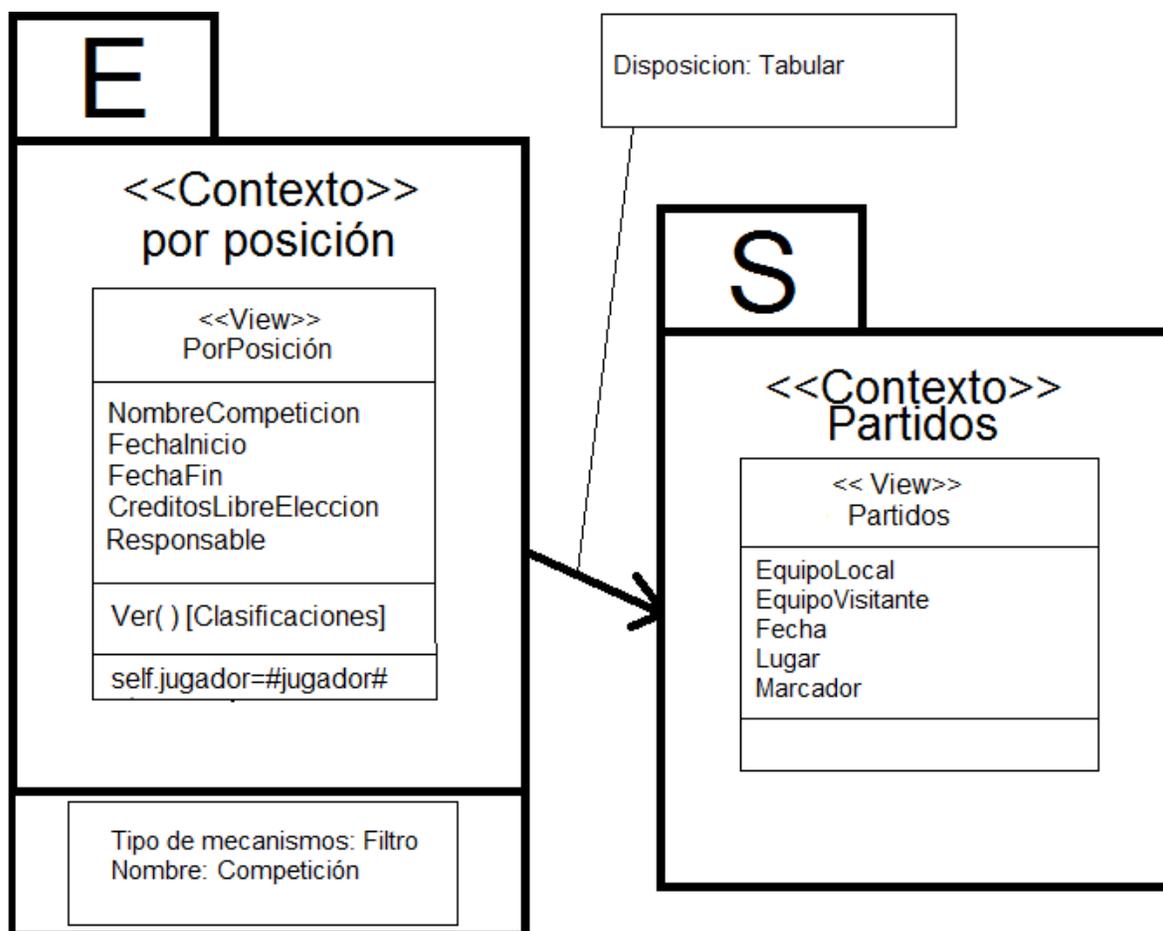


Ilustración 22: Enlace contextual de servicio del contexto Competiciones por posición del usuario Jugador.

En la ilustración 22, se muestra el enlace contextual de servicio del contexto *Competiciones por posición* del usuario *Jugador*;

La instrucción *Self.Jugador = #jugador#* : Significa que cuando el usuario *jugador* selecciona *ver()* que aparece en la esquema del contexto *Conexión por posición* solo visualiza las competiciones donde el jugador está inscrito mostrando datos de los partidos de dicha competición (Equipo local, equipo visitante, fecha, lugar y el marcador).

4. Implementación

4.1 Arquitectura

En el desarrollo del sistema se ha utilizado una arquitectura de Cliente/Servidor de tres capas: la capa de presentación, la capa de la lógica del negocio y la capa de acceso de datos.

En la capa de presentación se han desarrollado tanto los formularios de entrada de datos, como las páginas que muestran la información , así como las diferentes vistas de cada página, que tienen cada uno de los tres tipos de usuarios

En la capa de la lógica del negocio se ha desarrollado la funcionalidad del sistema, así como todos los métodos para conectarse, desconectarse, insertar, actualizar, eliminar y recuperar información de la base de datos, todo usando el lenguaje de programación *C#* que ofrece el entorno de programación *Visual Studio*.

Para la capa de acceso de datos se ha utilizado *Microsoft SQL Server Management Studio Express*.

4.2 Diseño de la Base de Datos

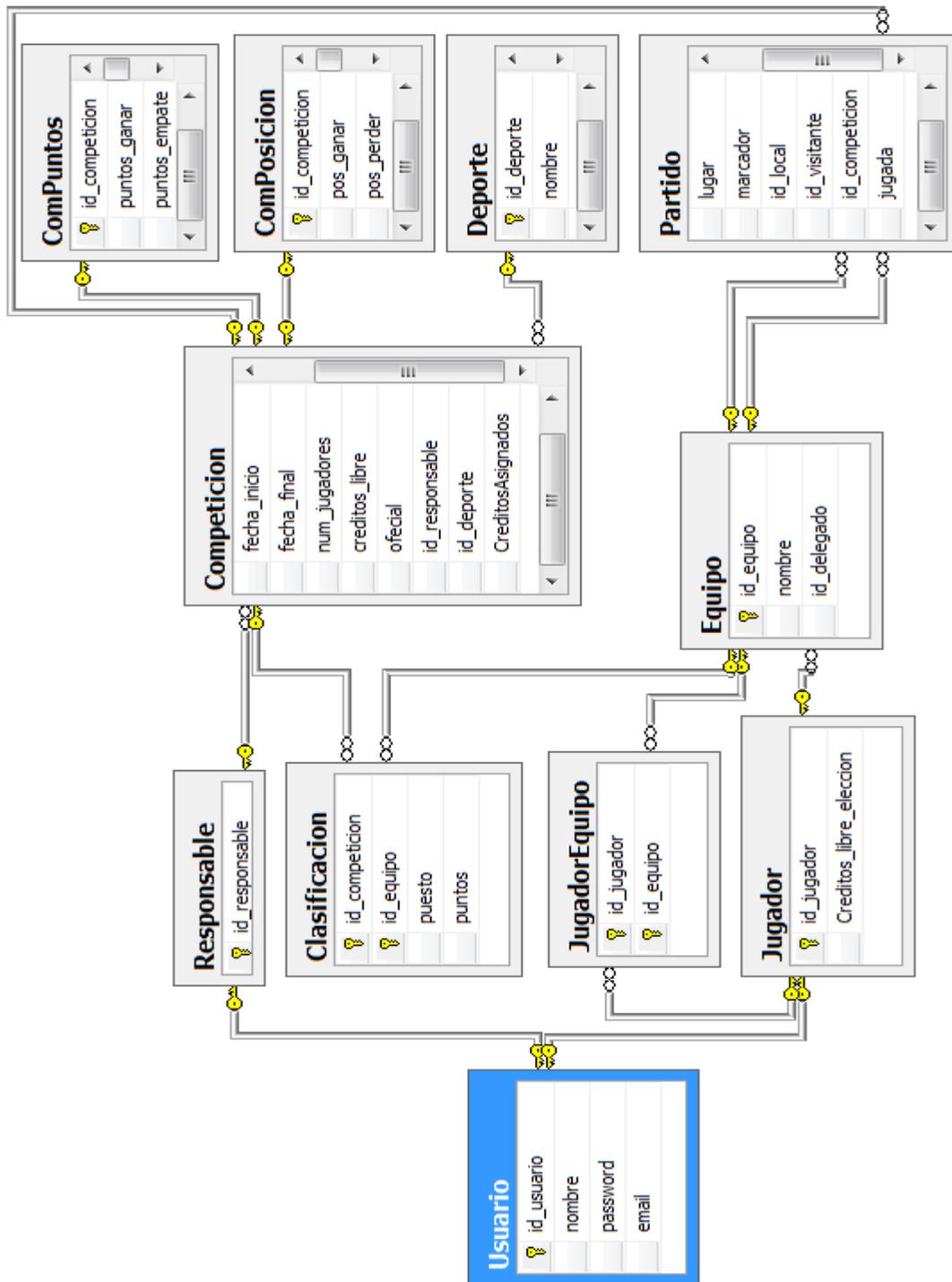


Ilustración 23: Esquema de Bases de datos de la aplicación GCD.

4.3 Patrones de Traducción

Mapa Navegacional: Se convierte cada nodo del mapa navegacional en un tipo de página.

4.3.1 Anónimo

Inicio: Es la página principal a donde entran los usuarios no registrados esta siempre accesible y sin navegación a otras páginas, es una página de noticias generales sobre alguna novedad en el mundo de deporte y las competiciones.

Deportes: Página de información desde la cual podemos navegar a las competiciones de un deporte, y que esta siempre accesible.

Competiciones por puntos y Competiciones por posición: Página de información desde la cual se puede navegar a la clasificación de una competición.

Clasificación: Página de información.

Registrarse: Página de Entrada de datos, siempre esta accesible.

Iniciar Sesión: Página de Entrada de datos, siempre esta accesible.

4.3.2 Jugador

Inicio: Es una página de accesibilidad sin navegación a otras páginas y que esta siempre accesible y por defecto, no es tan diferente que la página de Inicio del usuario anónimo.

Competiciones por puntos, Competiciones por posición: Página de información desde la cual se puede navegar al calendario de partidos de las

competiciones en que están inscritos los equipos donde participa el usuario Jugador que está conectado en este momento.

Partidos: Página de información, desde la cual se puede añadir el resultado (entrada de datos).

Gestión Equipo: Página de información y entrada de datos. Y lleva todas las operaciones que gestionan los equipos.

Mis datos: Página de información, donde se puede visualizar los datos del usuario Jugador que está conectado con la posibilidad de cambiar la contraseña.

4.3.3 Responsable de las competiciones

Inicio: Es una página por defecto con enlaces a todas las operaciones del usuario Responsable de la Competición. Está dividida en seis bloques principales, para que el Responsable tenga una vista global del sistema.

Estos bloques son:

Competiciones: este bloque define las diferentes operaciones que se pueden realizar a nivel de competiciones: como introducir, visualizar, modificar y crear Competiciones.

Equipos: contiene todas las operaciones relacionadas con Equipo.

Clasificaciones: en este bloque se pueden consultar las clasificaciones de las competiciones que están a cargo del responsable conectado.

Partidos: un bloque con todas las acciones relacionadas con Partidos (crear, modificar y consultar).

Deportes: permite la creación de un nuevo deporte, y consultar otros deportes existentes.

Otros: define unas operaciones especiales; como asignar créditos de libre elección, confirmar los partidos que proponen los equipos dentro del ámbito competiciones “Ranking” y finalmente MisDatos.

4.4 Contexto de Navegación y Relación de Contexto

4.4.1 Anónimo

Inicio: No muestra información relevante, suele mostrar alguna noticia de la actualidad.

Deportes: Se consultan todos los deportes presentes en la base de datos, y se muestra el nombre del deporte con un enlace a las competiciones del deporte. Clase directora: deporte, disposición de la información: Tabular Vertical, ordenado Ascendentemente. El enlace a competiciones se realiza a través del enlace "Select".

Competiciones por puntos, Competiciones por posición: Se muestra la información de una competición, si la competición no ha finalizado. Clase directora: competición, disposición de la información: Maestro-detalle, ordenado Ascendentemente, agrupados por tipo de competición.

Clasificación: Muestra la clasificación de una competición, que no ha finalizado. Clase directora: competición, disposición de la información: tabular.

4.4.2 Jugador

Inicio: No muestra información relevante, suele mostrar alguna noticia de la actualidad.

Competiciones por puntos, Competiciones por posición: Se muestra la información de una competición. Clase directora: competición, disposición de la información: Tabular. Aparecen enlaces que permiten ver los

calendarios de partidos (Partidos Jugados, Partidos por jugar y otro para todos los partidos)

Clasificación: Muestra la clasificación de una competición. En la misma página aparecen las clasificaciones de las competiciones tanto por posición como por puntos. Clase directora: competición, disposición de la información: Tabular.

Partidos: Muestra la información referente a un partido. Clase directora: Partido. Con la posibilidad de invocar el servicio introducir resultado.

Gestión Equipo: Muestra varias opciones y servicios que se pueden hacer a nivel del equipo, algunas de esas opciones muestra un listado de todos los equipos de los cuales forma parte el jugador, se puede invocar el servicio de crear equipo desde esta página. Clase directora: equipo, disposición de la información: Tabular vertical.

Mis datos: Muestra la información de un usuario. Clase directora: jugador, disposición de la información: Registro. El servicio de cambiar contraseña, se invoca con un *botónlink* con el nombre “Cambiar”.

4.4.3 Responsable de las competiciones

Inicio: Es una página por defecto con enlaces a todas las operaciones del usuario Responsable de la Competición. Está dividida en seis bloques principales, para el Responsable tenga una vista global del sistema.

Estos bloques son:

Competiciones: con todas las operaciones que se puede hacer a nivel de competiciones: introducir, visualizar, modificar y crear Competiciones.

Equipos: con las operaciones a nivel de Equipo.

Clasificaciones: con las operaciones para consultar clasificaciones de todas las competiciones que están a cargo del responsable conectado.

Partidos: con todas las operaciones relacionadas a Partidos (crear, modificar y consultar).

Deportes: con las operaciones que se hacen a nivel de Deportes (consultas y creación).

Otros: con operaciones especiales como asignar créditos de libre elección, confirmar los partidos que proponen los equipos dentro del ámbito de las competiciones Ranking y MisDatos.

4.5 Ejemplos

Una vista del usuario Anónimo:

Gestión de Competición Deportiva.



[Inicio](#)

[Inicio](#) [Deportes](#) [Competiciones](#) [Clasificaciones](#) [Regístrate](#) [Iniciar Sesión](#)

Tercer Puesto Para El Equipo Platu 25 de la Universitat Politècnica de València.



Tercera posición para la UPV , tras el Fernax y PIX 1. El equipo patrocinado por Héctor Vilanova alumno de la Escuela de Arquitectura, junto con Marta Carratalá, alumna de la misma escuela, y los profesores Juan Antonio Liébres, Alberto Sanchis, del Departamento de Sistemas Informáticos y Computación, y Alicia Martínez, profesora del Departamento de Construcciones Arquitectónicas, han quedado tercer clasificados en la clase Platu 25 en el XIII Trofeo S.M.

La Reina y XXIV Copa Almirante Marcial Sanchez-Bercaistegui, celebrado del 1 al 3 de julio en aguas del Real Club Náutico de Valencia.

El Trofeo S.M. la Reina cuenta con un gran prestigio en el panorama náutico nacional y es una de las grandes citas deportivas de la ciudad. En el aspecto deportivo, el Trofeo de S.M. La Reina se disputa en tres campos de regata: Alpha, Bravo y Charlie. Al salir del Real Club Náutico de Valencia, se ubica el campo Charlie, en el que han regateado los Platu 25, justo frente a la playa de Pinedo

1ª Victoria En El Campeonato Universitario de la UPV Europa voleibol



Un partido disputado a las 8:30 de la mañana, fuera del horario oficial, lo que provoca una protesta oficial por parte del equipo de la UPV.

A pesar de los inconvenientes (balón nuevo y el 2º árbitro era local) y de la ola de calor que esta azotando Serbia, en los últimos días, el equipo valenciano se alzo con la victoria, en un disputado encuentro.

El resultado final del encuentro fue 1-3, favorable ala UPV.

El 2º set fue tuvo una muy buena salida a partir por parte de equipo local que fue respondido con un gran juego por parte de las jugadoras valencianas.

El primer set fue muy igualado hasta el punto 15 y a partir de ahí tomo una leve ventaja el equipo de la UPV que le sirvió para ganar el set.

En el tercer set, la UPV acusó el bajón físico debido al calor y al esfuerzo realizado en los sets anteriores y las jugadoras serbias desplegaron un juego más combinativo.

El 4º set y definitivo el equipo español hizo gala de un juego más combinativo y ganó a un partido emocionante.

El equipo femenino de voleibol de la Universitat Politècnica de València se encuentra en Serbia disputando el XI Campeonato de Europa Universitario

El partido que abrió el XI Campeonato de Europa Universitario fue el que enfrentó al equipo de la UPV con el equipo anfitrión, Universidad de Kragujevac, Serbia.

Una vista del usuario Jugador:

Gestión de Competición Deportiva.

Inicio [Gestión equipo](#) [Competiciones](#) [Partidos](#) [Clasificaciones](#) [Mis Datos](#) [Berkan Abdelmajid](#)
[Cerrar sesión](#)

[Inicio](#) >> [Introducir resultado de partidas jugadas](#)

Introducir resultado de partidas jugadas

Buscar por nombre competición

Lista de competiciones:

Lista de partidos pendientes de introducir el resultado

<u>Equipo local</u>	<u>Equipo Visitante</u>	<u>Fecha</u>	<u>Lugar</u>	<u>Opciones</u>
ARQ Fútbol	CAM Fútbol	08/01/2011 18:30:00	UPV-Pista1	Introducir el resultado
CAM Fútbol	EUI Fútbol	10/01/2011 18:30:00	UPV-Pista1	Introducir el resultado
CAM Fútbol	ARQ Fútbol	12/01/2011 18:30:00	UPV-Pista1	Introducir el resultado

Introduce el resultado del partido:

CAM Fútbol <> ARQ Fútbol

1 0 x 2

[Validar resultado](#)

(*) 1: ha ganado el equipo local (que se encuentra a la izquierda).
(*) 2: ha ganado el equipo visitante (que se encuentra a la derecha).
(*) X: El equipo local y el equipo visitante han empatado.

© Copyright Gestión De Competiciones Deportivas - UPV-2010 - abdel@ulov.es

Una vista del usuario Responsable de competiciones:

The screenshot displays the 'Gestión de Competición Deportiva' web application interface. The header features the title 'Gestión de Competición Deportiva.' and a navigation bar with 'Inicio' and 'Maya Berkane Cerrar sesión'. The main content area is organized into several functional blocks:

- COMPETICIONES:** Includes options like 'Ver Competiciones', 'Competiciones por puntos', 'Competiciones por posición', 'Clasificaciones de competiciones', 'Cerrar inscripción en una competición', 'Dar de alta a una competición', and filters for 'Por posición' and 'Por puntos'.
- EQUIPOS:** Includes 'Inscribir un equipo en una competición', 'Detalles de un equipo', and 'Calendario de partidos de un equipo'.
- PARTIDOS:** Includes 'Crear calendario de partidos', 'Ver calendario de partidos', and 'Detalles partido'.
- DEPORTES:** Includes 'Crear deporte' and 'Ver clasificaciones de un deporte'.
- CLASIFICACIONES:** Includes 'Ver clasificaciones de una competición' and 'Actualizar clasificaciones tras jugar un partido'.
- OTROS:** Includes 'Partidos Propuestos (Competiciones por posición)' with a notification icon, 'Asignar créditos de libre elección a jugadores', 'Asignar créditos de libre elección al equipo ganador', and 'Hacer datos'.

The footer contains the text: '© Copyright: Gestión De Competiciones Deportivas - UPA-2010 - abber@ul.ups.edu.ec'.

5 Diseño relacional de la base de datos

5.1 Introducción

El diseño lógico consiste en transformar el esquema conceptual que se encuentra descrito usando un cierto modelo de datos, a estructuras y transacciones descritas en términos del modelo de datos en el cual se base el sistema de gestión de base de datos que se vaya a utilizar. Ya que el modelo de datos elegido es el modelo relacional se deberá realizar este proceso para obtener finalmente un esquema relacional y un conjunto de transacciones sobre las relaciones de este esquema.

A continuación se profundizará en cada una de las entidades y relaciones que forman el diagrama Entidad-Relación de sistema.

5.2 Diseño lógico

En este apartado se van a mostrar las diferentes tablas que estarán contenidas en la base de datos diseñada. Cada una de las tablas realizará una función indispensable para el correcto funcionamiento de la base de datos. Las tablas que forman esta base de datos son:

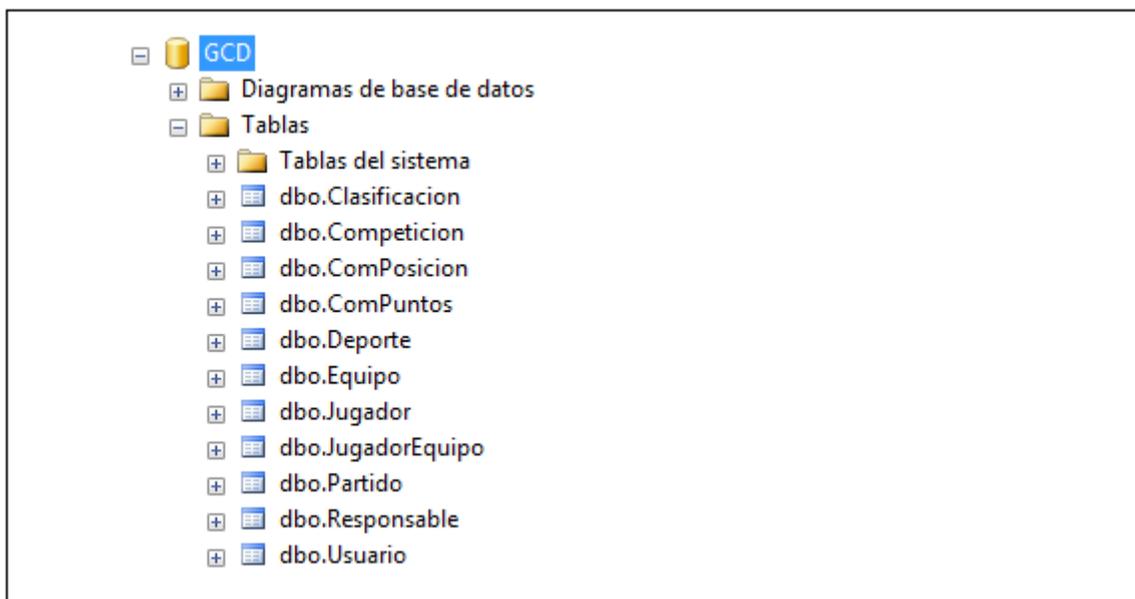


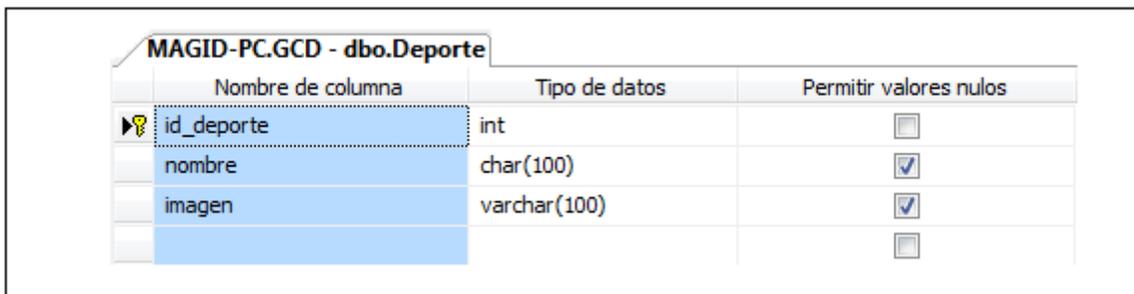
Ilustración 24: Esquema de las tablas de Bases de datos de la aplicación GCD.

- **dbo.Usuario.**
- **dbo.Responsable.**
- **dbo.Jugador.**
- **dbo.Competicion.**
- **dbo.ComPosicion.**
- **dbo.ComPuntos.**
- **dbo.Deporte.**
- **dbo.Clasificacion.**
- **dbo.Partido.**
- **dbo.Equipo.**
- **dbo.JugadorEquipo**

Ahora se analizará una a una cada tabla y dentro de cada una de ellas los campos que la componen y su significado:

Tabla dbo.Deporte: Esta tabla almacena todos los deportes de la base de datos. Sus campos son:

- **id_deporte:** Código numérico auto generado único para cada deporte.
- **nombre:** Nombre del deporte.
- **imagen:** Almacena el nombre del fichero de la imagen del deporte (100x60).



The screenshot shows a table structure for 'dbo.Deporte' with the following columns:

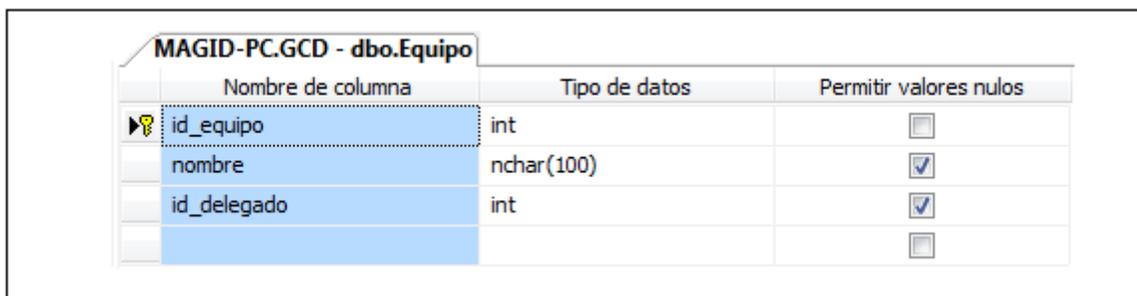
Nombre de columna	Tipo de datos	Permitir valores nulos
id_deporte	int	<input type="checkbox"/>
nombre	char(100)	<input checked="" type="checkbox"/>
imagen	varchar(100)	<input checked="" type="checkbox"/>
		<input type="checkbox"/>

Ilustración 25: Esquema de la tabla de Bases de datos DEPORTE.

Tabla dbo.Equipo: Esta tabla almacena todos los equipos inscritos en diferentes competiciones del sistema. Sus campos son:

- **id_equipo:** Código numérico auto generado único para cada equipo.
- **nombre:** Nombre del equipo.

- **id_delegado:** Almacena el identificador del delegado del equipo, tienes que ser un jugador.



Nombre de columna	Tipo de datos	Permitir valores nulos
 id_equipo	int	<input type="checkbox"/>
nombre	nchar(100)	<input checked="" type="checkbox"/>
id_delegado	int	<input checked="" type="checkbox"/>
		<input type="checkbox"/>

Ilustración 26: Esquema de la tabla de Bases de datos EQUIPO.

Tabla dbo.Partido: Esta tabla almacena todos los partidos de los equipos inscritos en diferentes competiciones del sistema. Sus campos son:

- **id_partido:** Código numérico auto generado único para cada partido.
- **fecha:** Almacena la fecha cuando se juega el partido.
- **lugar:** Almacena el lugar donde se juega el partido
- **marcador:** Almacena el resultado del partido.
- **id_local:** Almacena el identificador del equipo local.
- **id_visitante:** Almacena el identificador de equipo visitante.
- **id_competicion:** Almacena el identificador de la competición donde están inscritos los equipos (local y visitante) que juegan el partido.
- **jugada:** Es valor binario, almacena si el partido se ha jugado o está por jugar.

MAGID-PC.GCD - dbo.Partido		
Nombre de columna	Tipo de datos	Permitir valores nulos
 id_partido	int	<input type="checkbox"/>
fecha	smalldatetime	<input checked="" type="checkbox"/>
lugar	nchar(100)	<input checked="" type="checkbox"/>
marcador	nchar(100)	<input checked="" type="checkbox"/>
id_local	int	<input checked="" type="checkbox"/>
id_visitante	int	<input checked="" type="checkbox"/>
id_competicion	int	<input checked="" type="checkbox"/>
jugada	bit	<input checked="" type="checkbox"/>

Ilustración 27: Esquema de la tabla de Bases de datos PARTIDO.

Tabla dbo.Clasificacion: Esta tabla recoge información sobre las clasificaciones de los equipos que participan en las competiciones. Sus campos son:

- **id_competicion:** Código numérico, Forma parte de la clave primaria de la tabla **dbo.Clasificacion** junto con el **id_equipo**.
- **id_equipo:** Código numérico, Forma parte de la clave primaria de la tabla **dbo.Clasificacion** junto con el **id_competicion**.
- **puesto:** Un valor numérico, recoge la información sobre el puesto donde se encuentra el equipo si la competición es por posición.
- **puntos:** Un valor numérico, recoge la información sobre los puntos que obtenga un equipo si la competición es por puntos.

MAGID-PC.GCD - dbo.Clasificacion		
Nombre de columna	Tipo de datos	Permitir valores nulos
 id_competicion	int	<input type="checkbox"/>
 id_equipo	int	<input type="checkbox"/>
puesto	int	<input checked="" type="checkbox"/>
puntos	int	<input checked="" type="checkbox"/>
		<input type="checkbox"/>

Ilustración 28: Esquema de la tabla de Bases de datos CLASIFICACIÓN.

Tabla dbo.Competicion: Esta tabla recoge información sobre las competiciones de que dispone la BD. Sus campos son:

- **id_competicion:** Numérico auto generado único para cada competición.
- **Nombre:** Nombre de la competición.
- **Fecha_inicio:** Almacena la fecha de inicio de la competición.
- **Fecha_fin:** Almacena la fecha cuando se terminará la competición.
- **Num_jugadores:** Es un valor numérico, recoge la información sobre el número de los jugadores permitidos en una competición.
- **Créditos_libre:** Es un valor numérico, recoge la información sobre los créditos de libre elección que se asignarán al equipo que gana la competición.
- **Oficial:** Es un valor lógico, almacena si la competición es oficial o no.
- **Id_responsable:** Almacena el identificador del responsable de la competición. El valor de esa entrada tiene que ser igual a un valor que existe en la tabla **dbo.Responsable**.
- **Id_deporte:** Almacena el identificador del deporte de dicha competición. El valor de esa entrada tiene que ser igual a un valor que existe en la tabla **dbo.Deporte**.
- **CreditosAsignados:** Es un valor Booleano, cuando esta activado significa que se ha asignado los créditos de libre elección de dicha competición.

Nombre de columna	Tipo de datos	Permitir valores nulos
id_competicion	int	<input type="checkbox"/>
Nombre	nchar(100)	<input type="checkbox"/>
fecha_inicio	smalldatetime	<input checked="" type="checkbox"/>
fecha_final	smalldatetime	<input checked="" type="checkbox"/>
num_jugadores	int	<input checked="" type="checkbox"/>
creditos_libre	int	<input checked="" type="checkbox"/>
ofecial	bit	<input checked="" type="checkbox"/>
id_responsable	int	<input checked="" type="checkbox"/>
id_deporte	int	<input checked="" type="checkbox"/>
CreditosAsignados	bit	<input checked="" type="checkbox"/>
		<input type="checkbox"/>

Ilustración 29: Esquema de la tabla de Bases de datos COMPETICIÓN.

Tabla dbo.ComPosicion: Esta tabla recoge información sobre las competiciones que por posición. Sus campos son:

- **Id_competicion:** Almacena el identificador de una competición por posición. El valor de esa entrada tiene que ser igual a un valor que existe en la tabla **dbo.Competicion**.
- **pos_ganar:** Valor numérico, almacena el numero de posiciones que ha y que avanzar cuando el equipo gana un partido dentro una competición por posición
- **pos_perder:** Valor numérico, almacena el numero de posiciones que ha y que volver atrás cuando el equipo pierde un partido dentro una competición por posición.
- **distancia_max:** Valor numérico, almacena la diferencia de posiciones de los equipos en una competición por posición. Mientras que el valor distancia es mayor la diferencia de posiciones de los equipos, se puede solicitar partidos, en caso contrario no se puede.
- **cerrada:** Es un valor lógico, se activa cuando se cierra la inscripción de los equipos en la competición, y esta a valor 'False' cuando todavía se puede inscribirse.

Nombre de columna	Tipo de datos	Permitir valores nulos
id_competicion	int	<input type="checkbox"/>
pos_ganar	int	<input checked="" type="checkbox"/>
pos_perder	int	<input checked="" type="checkbox"/>
distancia_max	int	<input checked="" type="checkbox"/>
cerrada	bit	<input checked="" type="checkbox"/>
		<input type="checkbox"/>

Ilustración 30: Esquema de la tabla de Bases de datos COMPETICIÓN POR POSICIÓN.

Tabla dbo.ComPuntos: Esta tabla recoge información sobre las competiciones que por puntos. Sus campos son:

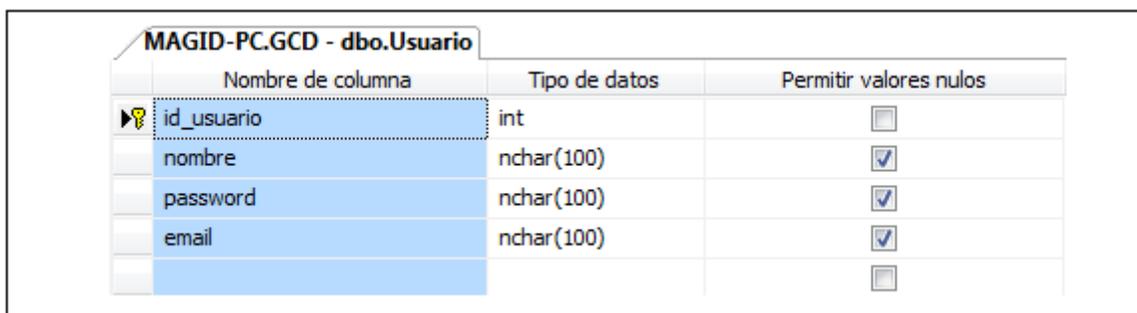
- **Id_competicion:** Almacena el identificador de una competición por puntos. El valor de esa entrada tiene que ser igual a un valor que existe en la tabla **dbo.Competicion**.
- **puntos_ganar:** Es un valor numérico, almacena los puntos que se asignan al equipo ganador.
- **puntos_perder:** Es un valor numérico, almacena los puntos que se asignan al equipo perdedor.
- **puntos_empate:** Es un valor numérico, almacena los puntos que se asignan a los equipos cuando se empatan.
- **cerrada:** Es un valor lógico, se activa cuando se cierra la inscripción de los equipos en la competición, y esta a valor 'False' cuando todavía se puede inscribirse.

Nombre de columna	Tipo de datos	Permitir valores nulos
id_competicion	int	<input type="checkbox"/>
puntos_ganar	int	<input checked="" type="checkbox"/>
puntos_empate	int	<input checked="" type="checkbox"/>
puntos_perder	int	<input checked="" type="checkbox"/>
cerrada	bit	<input checked="" type="checkbox"/>
		<input type="checkbox"/>

Ilustración 31: Esquema de la tabla de Bases de datos COMPETICIÓN POR PUNTOS.

Tabla dbo.Usuario: Esta tabla recoge información sobre todos los usuarios del sistema. Sus campos son:

- **Id_usuario:** Numérico auto generado único para cada usuario.
- **nombre:** Nombre de usuario, (nombre y apellidos)
- **password:** Una cadena de caracteres, almacena la contraseña del usuario.
- **email:** Una cadena de caracteres, almacena el correo electrónico del usuario.

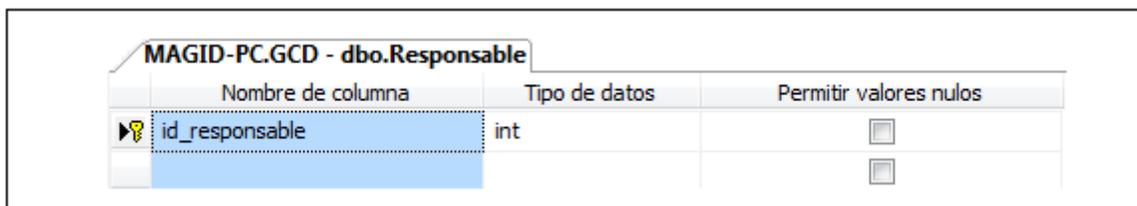


Nombre de columna	Tipo de datos	Permitir valores nulos
id_usuario	int	<input type="checkbox"/>
nombre	nchar(100)	<input checked="" type="checkbox"/>
password	nchar(100)	<input checked="" type="checkbox"/>
email	nchar(100)	<input checked="" type="checkbox"/>
		<input type="checkbox"/>

Ilustración 32: Esquema de la tabla de Bases de datos USUARIO.

Tabla dbo.Usuario: Esta tabla recoge información sobre todos los usuarios del sistema que son responsables de competiciones. Sus campos son:

- **id_responsable:** Almacena el identificador del responsable de la competición. El valor de esa entrada tiene que ser igual a un valor que existe en la tabla **dbo.Usuario**.



Nombre de columna	Tipo de datos	Permitir valores nulos
id_responsable	int	<input type="checkbox"/>
		<input type="checkbox"/>

Ilustración 33: Esquema de la tabla de Bases de datos RESPONSABLE DE COMPETICIONES.

Tabla dbo.Usuario: Esta tabla recoge información sobre todos los usuarios del sistema que son Jugadores. Sus campos son:

- **id_jugador:** Almacena el identificador del jugador. El valor de esa entrada tiene que ser igual a un valor que existe en la tabla **dbo.Usuario**.
- **Creditos_libre_eleccion:** Valor numérico, almacena los créditos de libre elección que se van acumulando para un jugador

Nombre de columna	Tipo de datos	Permitir valores nulos
id_jugador	int	<input type="checkbox"/>
Creditos_libre_eleccion	int	<input checked="" type="checkbox"/>

Ilustración 34: Esquema de la tabla de Bases de datos JUGADOR.

Tabla dbo.JugadorEquipo: Esta tabla, es una tabla de asociación, relaciona la tabla de Jugador con la tabla de Equipo. Sus campos son:

- **id_jugador:** Almacena el identificador del jugador. El valor de esa entrada tiene que ser igual a un valor que existe en la tabla **dbo.Usuario**, y forma parte de la clave de la tabla **dbo.JugadorEquipo** junto con **id_equipo**.
- **id_equipo:** Almacena el identificador del equipo. El valor de esa entrada tiene que ser igual a un valor que existe en la tabla **dbo.Equipo**, y forma parte de la clave de la tabla **dbo.JugadorEquipo** junto con **id.jugador**.

Nombre de columna	Tipo de datos	Permitir valores nulos
id_jugador	int	<input type="checkbox"/>
id_equipo	int	<input type="checkbox"/>

Ilustración 35: Esquema de la tabla de Bases de datos JUGADOR_EQUIPO.

6. Conclusiones

Después de llevar a cabo la realización de este proyecto se puede llegar a las siguientes conclusiones:

La red actualmente es un sitio accesible para todo el mundo y con un coste económico muy bajo.

Por otro lado, desde el punto de vista del desarrollador de la aplicación, Internet también contribuyó a la elevación de su rendimiento y llevar a cabo sus tareas diarias de implementar y desarrollar software. Una gran cantidad de información y tutoriales que le sirven de ayuda se ponen a disposición del programador para que pueda hacer sus tareas, consiguiendo buenos resultados en poco tiempo.

En cuanto a la utilización de la tecnología .NET, la conclusión a la que se ha llegado es que en la actualidad las tecnologías libres como PHP están empezando a ganar la partida a las propietarias (.NET) no solo por el coste que llevan asociadas éstas sino también por la gran cantidad de información y ejemplos prácticos que se encuentran en la red. Sin embargo una de las ventajas que continua teniendo .NET con relación a PHP es la facilidad y rapidez con la que se realizar aplicaciones profesionales sin tanto esfuerzo a través de la herramienta de desarrollo Visual Studio .NET

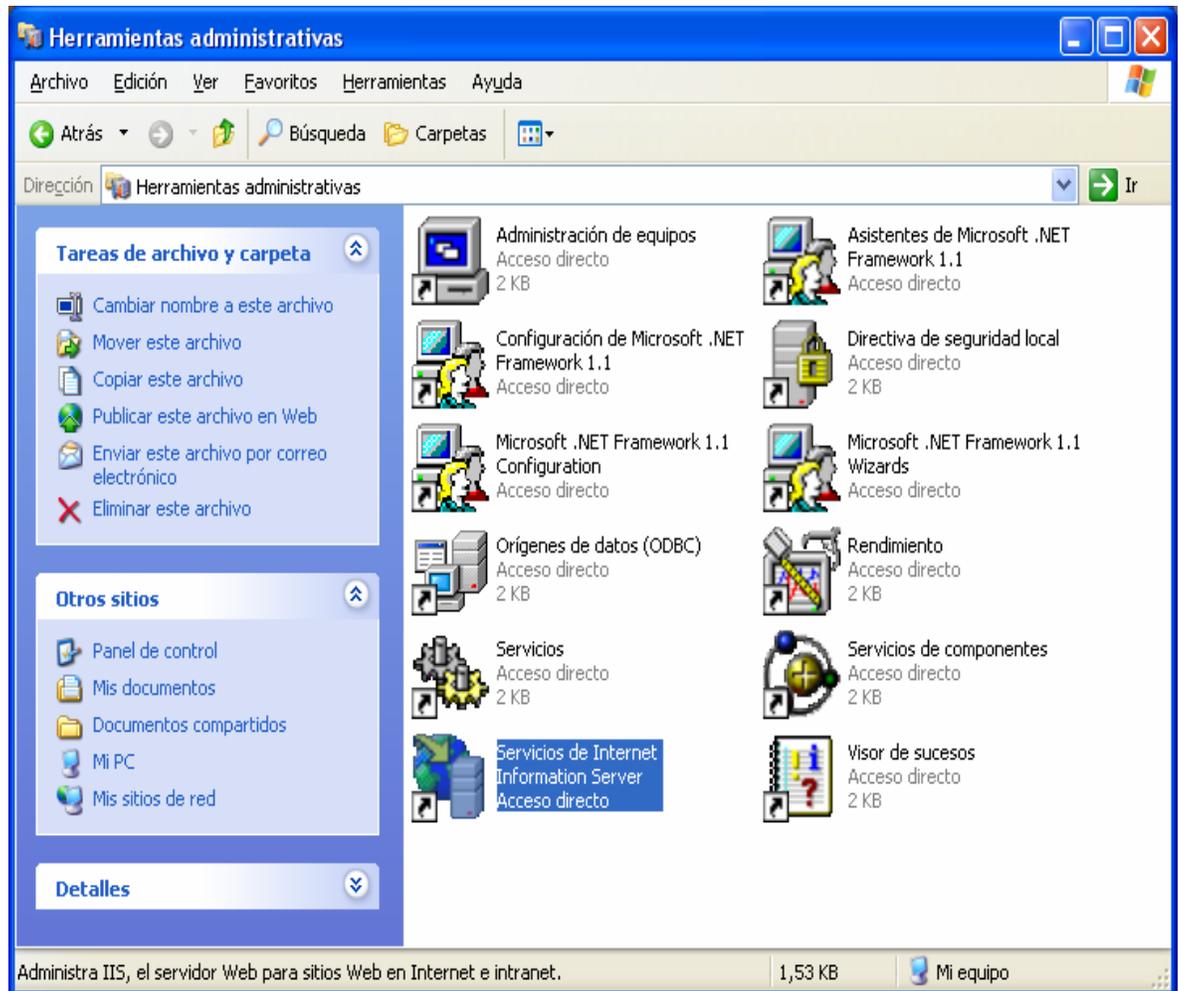
7. Bibliografía y Referencias

- [1] **ASP .net** - <http://es.gotdotnet.com/quickstart/aspplus>
 - Tutorial de ASP.NET con ejemplos prácticos de la tecnología .NET
- [2] **MSDN Microsoft** - <http://msdn2.microsoft.com/es-es/default.aspx>
 - Material de ayuda para los desarrolladores de aplicaciones bajo software y tecnologías de Microsoft
- [3] **El Guille** - <http://www.elguille.info/NET/ASPNET/indiceASPNET.aspx>
 - Tutorial y ejemplos desarrollados en .NET
- [4] **DesarrolloWeb** – <http://www.desarrolloweb.com>
 - Tutoriales y ejemplos prácticos de HTML, CSS, ASP, Javascript, .NET ...
- [5] **IEEE** - <http://www.ieee.org>
 - IEEE standard 830-IEEE Guide to Software Requirements Specifications
- [6] **Wikipedia** - <http://es.wikipedia.org>
 - Búsqueda de información sobre las tecnologías empleadas para la realización del proyecto
- [7] **Google** – <http://www.google.es>
 - Búsquedas generales de información de imágenes de los diferentes tipos deportes.
- [8] **TuWebAhora** – <http://www.tuwebahora.mex.tl>
 - Algunas informaciones y definiciones de los términos técnico usados en la documentación.
- [9] **EscenarioIdearios** – <http://escenarios.ideario.es>
 - La imagen usada en el logo de la aplicación la parte derecha.
- [10] **WebEstilo** – <http://www.webestilo.com>
 - Algunas informaciones y definiciones de los términos técnico usados en la documentación.
- [11] **UPV** – <http://www.upv.es>
 - Imágenes usadas en la página de inicio de la aplicación y enlaces a otras páginas.
- [12] **PAW** –_DESARROLLO DE APLICACIONES EN ENTORNOS WEB - ITIG/ITIS, 5867
 - La idea del proyecto fue cogida de las practicas de la asignatura **APW** (una parte del anunciado de las practicas de 2010 para documentar el caso de uso), Sin embargo se ha ampliado la funcionalidad y se han modificado los requisitos del sistema para que sea global y aplicable para cualquier tipo de competición deportiva.

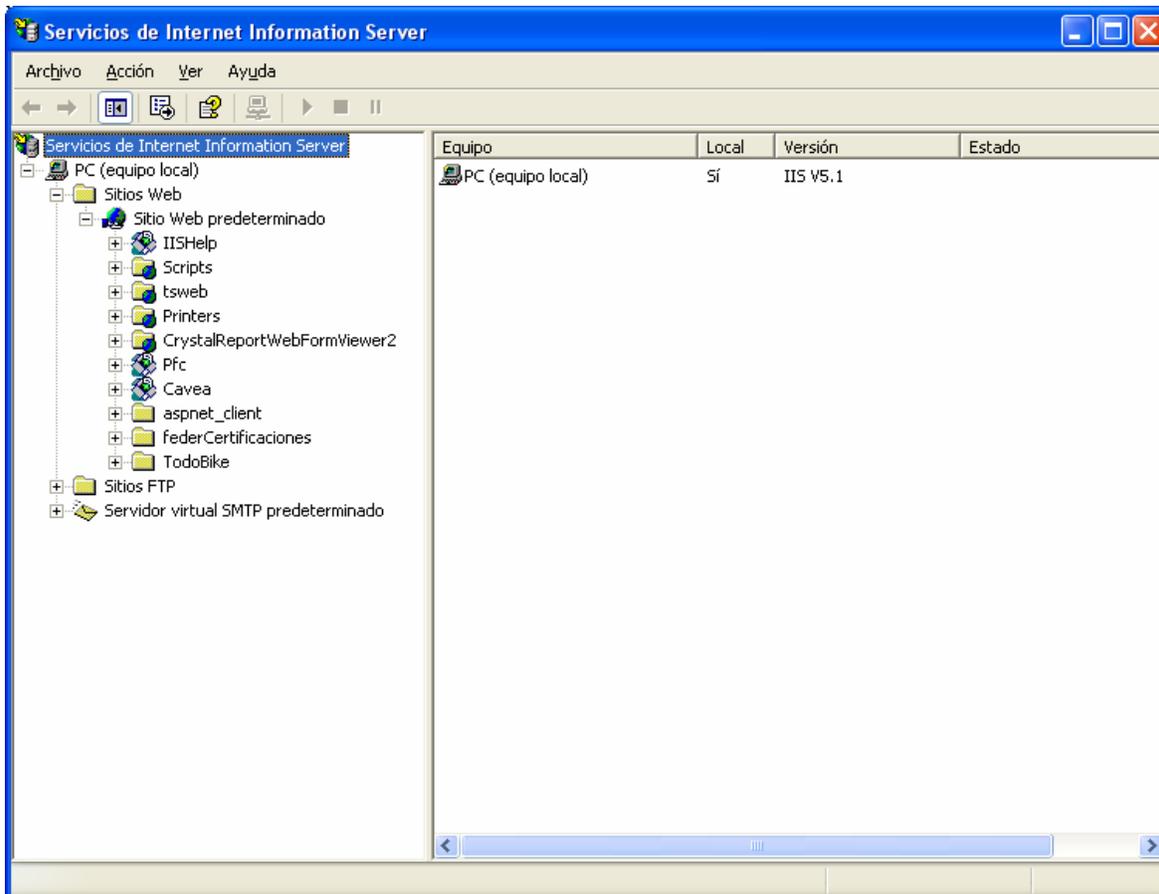
ANEXOS

ANEXO A: Configuración de la aplicación en el Servidor

- Los pasos necesarios para poner en funcionamiento la aplicación Web son los siguientes:
- Disponer de un ordenador que funcione bajo Windows 2000 o posterior con Microsoft Internet Information Server instalado y ejecutándose para que realice las funciones de servidor Web.
- Instalar (si no se encuentra ya instalado por alguna de las actualizaciones automáticas de Windows) Microsoft .NET Framework 1.0 o superior para que interprete la lógica de la aplicación Web desarrollada bajo ASP .NET.
- Disponer de conexión a Internet en el ordenador que actúe de servidor Web para que la aplicación sea accesible a todos los usuarios de Internet.
- Abrir el Explorador de archivos de Windows y crear dentro de la ruta **C:\inetpub\wwwroot** una carpeta llamada GSD que contendrá la aplicación Web en cuestión. *(IIS almacena en esa ruta local todas las aplicaciones Web publicadas en él)*
- Después de haber copiado todos los archivos en ubicación correspondiente, el siguiente paso es publicar la Web. Para ello se deberá abrir Microsoft Internet Information Server e indicárselo. El IIS se encuentra entre las Herramientas Administrativas de Windows. Para abrirlo se debe ir al menú Inicio, acceder a Configuración, luego Panel de Control, seleccionar Herramientas Administrativas.



Al abrir el icono de Administración de servicios de Internet, se abre la ventana principal del IIS...



Esta ventana proporciona gran cantidad de información. De toda ella se puede destacar:

- **Nombre del equipo local:** Es el nombre que tiene el equipo local (donde está funcionando este servidor de Internet).
- **Servicios disponibles:** IIS proporciona los servicios de Web, FTP y SMTP (correo electrónico).
- **Sitio Web predeterminado:** Es el sitio Web a partir del cual se deberán colgar las páginas Web que se quieran publicar.
- **Botones de Stop, Play y Pause:** Indican en qué estado está cada uno de los componentes del IIS. Si el servidor IIS está detenido o en pausa, cuando se acceda a páginas Web alojadas en la máquina, estas no se mostrarán.

IIS utiliza la figura de los directorios virtuales para publicar en la Web. Es decir, para exportar el contenido de un directorio en nuestra máquina a Internet, lo que se

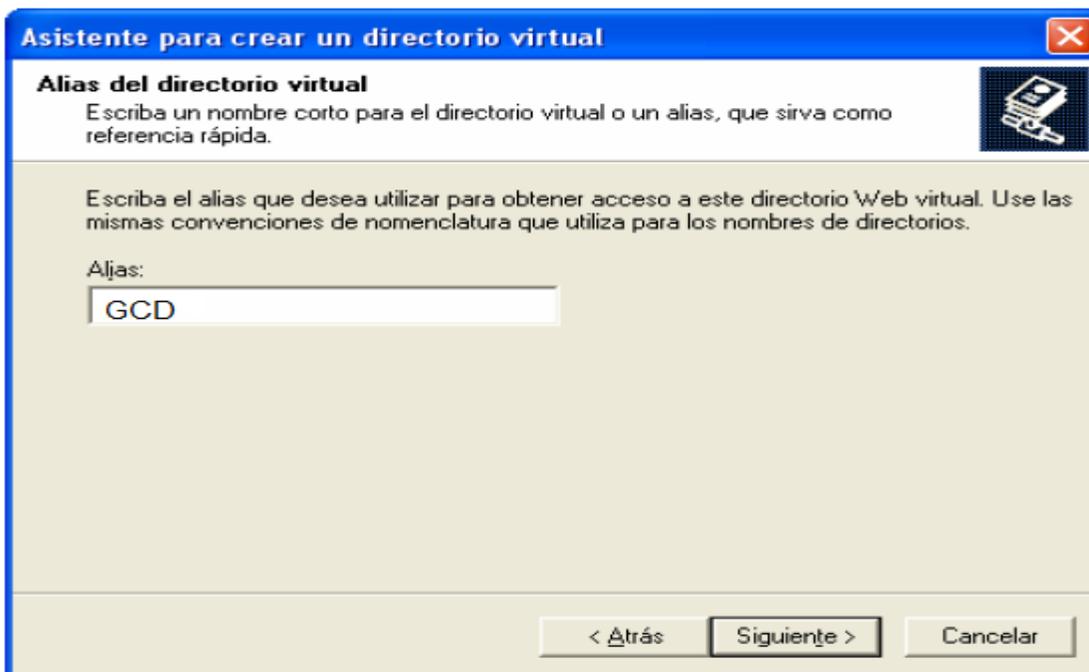
deberá hacer es crear un directorio virtual (GCD) en el servidor Web que apunte al directorio físico. El directorio virtual se deberá crear en la carpeta Sitio Web predeterminado de la ventana del IIS.

Para crear un directorio virtual se deberá abrir el IIS, desplegar sitios Web y darle al botón derecho sobre Sitio Web predeterminado y seleccionar Nuevo y Directorio Virtual.

Se mostrará un asistente para la creación de un nuevo directorio virtual...



Se nos preguntará qué nombre se le quiere dar al nuevo directorio virtual. El nombre a informar será GCD...

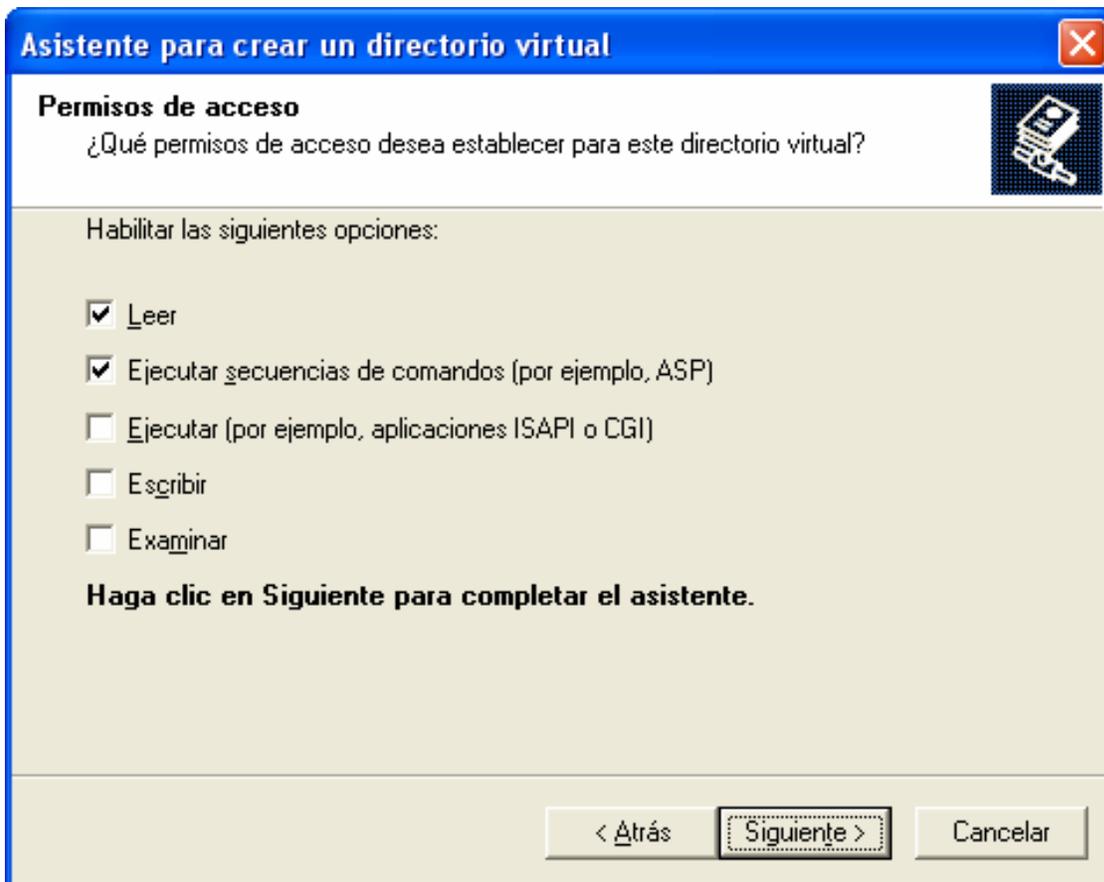


El siguiente paso será especificar el directorio físico donde están ubicadas o se ubicarán las páginas Web...



El último paso es establecer algunas opciones de configuración del directorio.

El asistente muestra la siguiente ventana...



Por motivos de seguridad, es recomendable dejar sólo estas dos opciones seleccionadas cuando sólo se publiquen páginas HTML con ASP .NET.

Finalmente, se puede establecer cuál será la página por defecto que se visualizará si se entra al sitio Web creado. Para ello, haciendo clic con el botón derecho sobre el directorio virtual, se accederá a sus Propiedades y en la pestaña "Documentos" se podrá establecer cuál será el documento (página Web) por defecto (predeterminado). En nuestro caso añadiremos Index.aspx a la lista para no tener que introducirlo a la hora de entrar en el sitio Web.