



UNIVERSIDAD
POLITECNICA
DE VALENCIA

Escuela Técnica Superior de Ingeniería Informática



Diseño e implementación de una Base de Datos para la gestión de eventos audiovisuales

PROYECTO FINAL DE CARRERA

AUTOR: Rafael Castañeda Lomas

DIRECTORA: Laura Mota Herranz

Julio de 2012

1	Presentación del proyecto y objetivos	5
2	Metodología de desarrollo	6
2.1	Análisis de restricciones en el nivel conceptual.....	10
2.2	Análisis de restricciones en el nivel lógico	11
2.3	Especificación de transacciones	11
3	Análisis	13
3.1	Descripción del problema.....	13
3.2	Requisitos de información.....	14
3.3	Requisitos de procesos	16
4	Diseño conceptual.....	17
4.1	Diagrama de Entidad-Relación	17
4.2	Análisis de restricciones externas en el nivel conceptual	26
5	Diseño lógico.....	28
5.1	Esquema lógico relacional	28
5.2	Análisis de restricciones en el nivel lógico	32
5.2.1	Tabla resumida de restricciones.....	34
5.3	Especificación de transacciones	35
6	Desarrollo de la aplicación en Oracle Forms 6i y Oracle Database XE.....	43
6.1	Montaje y conexión.....	43
6.2	Jerarquía de módulos.....	44
6.3	Documentación de los formularios	45
7	Implementación	50
7.1	Rellenado de listas.....	50
7.2	Botones.....	51
7.3	Procedimientos y funciones	53

8	Manual de usuario	63
9	Conclusiones.....	78
9.1	Trabajo realizado y valoración personal	78
10	Bibliografía	79

1 Presentación del proyecto y objetivos

El desarrollo del proyecto se centra en la realización una base de datos que almacene información sobre eventos audiovisuales. El objetivo será la creación de la base de datos y una interfaz que permita principalmente las siguientes operaciones:

- Mantenimiento de todos los objetos representados en la base de datos.
- Gestión de cines, salas y eventos que allí se visualicen.
- Gestión de los contenidos a visualizar.

2 Metodología de desarrollo

En la metodología para el desarrollo de aplicaciones que se propone se utiliza el *Modelo Entidad-Relación* para la realización del diseño conceptual y el *Modelo Relacional* para el diseño lógico y está basada en el *análisis de las restricciones de integridad* que se detectan a nivel conceptual, análisis que se integra en las fases de diseño de la base de datos por lo que a continuación se van a recordar estas fases extendiéndolas con nuevas tareas.

El modelo Entidad-Relación es un modelo de datos que se utiliza en la fase de diseño conceptual de una base de datos y que permite representar, en lo que se llama *diagrama ER*, las estructuras que constituyen el contenido del sistema de información junto con restricciones de distintos tipos que limitan las ocurrencias válidas de las mismas (de identificación, valor no nulo, etc.). Para ello hace uso, fundamentalmente, de tres conceptos: entidad, atributo y relación, representados respectivamente por rectángulos, óvalos y rombos. Evidentemente, este modelo no es capaz de representar *exactamente* cualquier sistema de información; las propiedades que se le escapan al modelo, se representan mediante la definición de las llamadas *restricciones añadidas*. Aunque existen algunas propuestas de lenguajes formales para de especificación de estas restricciones, en esta metodología se propone el uso del lenguaje natural.

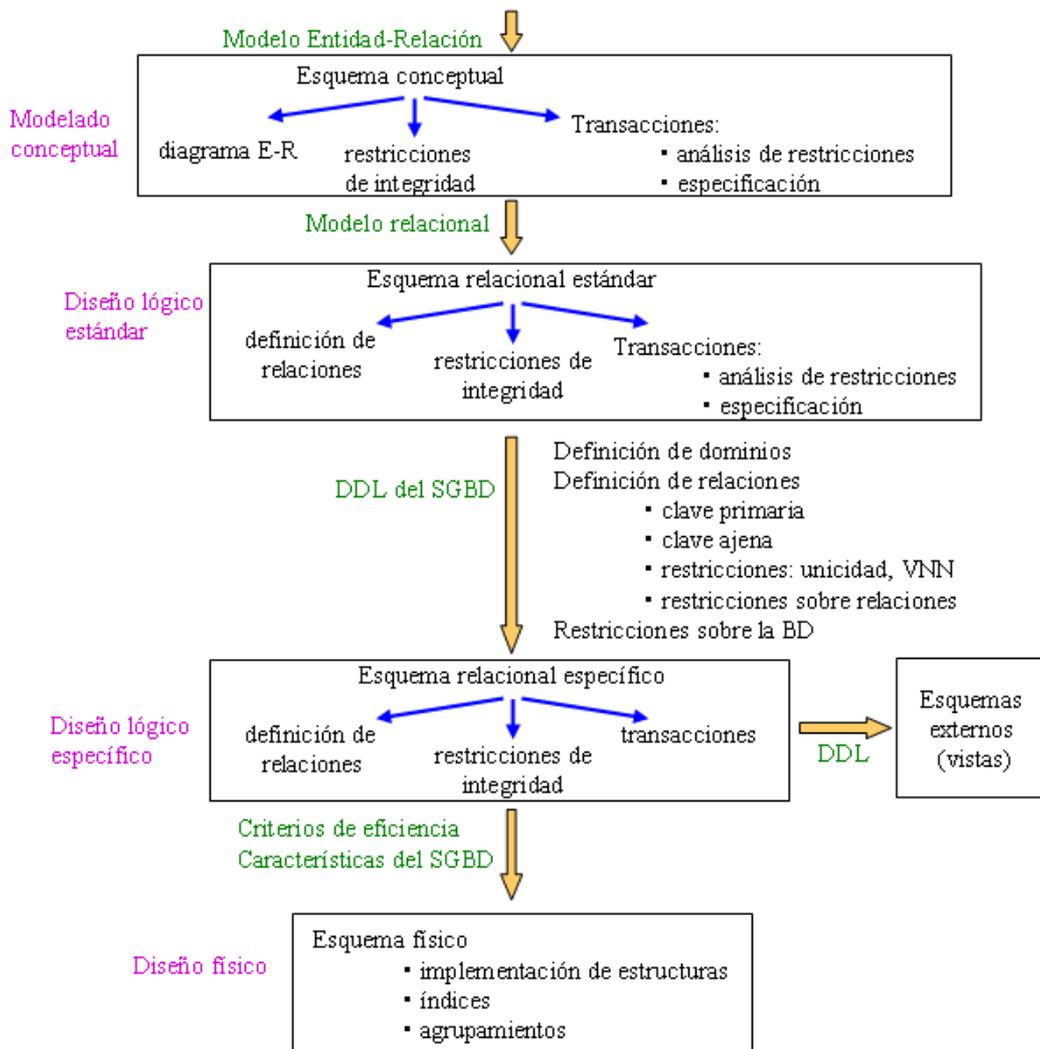
Se pueden distinguir dos tipos de restricciones añadidas:

- *Restricciones externas (R_{ex})*: representan propiedades del sistema de información que no han podido representarse en el diagrama ER. Un tipo especial de estas restricciones son las llamadas *dinámicas* ya que no definen estados válidos del sistema sino transiciones de estados válidos, es decir, son restricciones que no deben ser siempre ciertas sino cuando ocurre algo determinado.
- *Restricciones de atributos derivados (R_{ad})*: definen cómo se puede calcular el valor de un atributo derivado incluido en el diagrama ER.

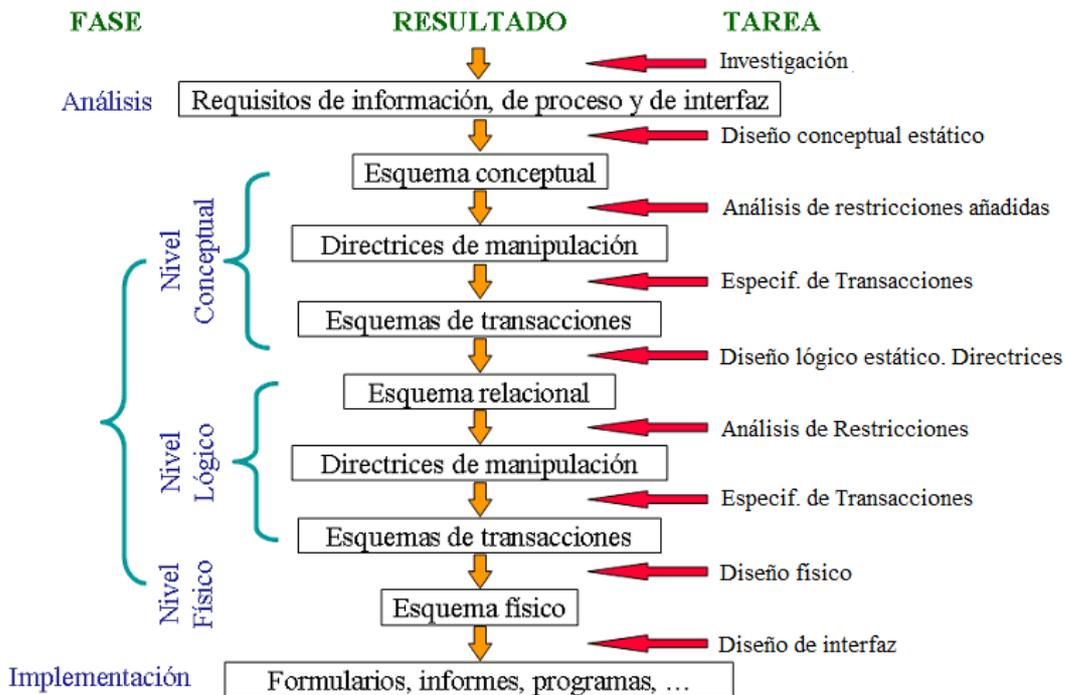
La fase que sigue al diseño conceptual es el diseño lógico. En esta fase se realiza la transformación (traducción) del esquema conceptual en términos del modelo relación (relaciones, claves primarias, claves ajenas, etc.). La mayoría de los textos que abordan este tema consideran que el diseño lógico consta de dos partes; por una parte, el *diseño lógico estándar* en el que se utilizan las estructuras del modelo relacional (i.e. relaciones) y, por otra parte, el *diseño lógico específico* en el que, el resultado del diseño estándar es ajustado al sistema de gestión de bases de datos comercial que se haya elegido. Cuanto más fiel sea este sistema al modelo de datos en que se basa, menor será el trabajo a realizar en el diseño específico.

Una vez realizado el diseño de la base de datos, hay que abordar el problema de diseñar las aplicaciones que la manipulan. Una *aplicación de bases de datos* es un programa que interactúa con una base de datos en algún punto de su ejecución.

Las fases del diseño de una base de datos comentadas en el tema anterior se presentan en el siguiente cuadro de forma aún más detallada teniendo en cuenta la metodología que se va a proponer.



En el esquema anterior, se pueden detectar las distintas tareas que deberán realizarse para la obtención final de la aplicación. Estas tareas se muestran en la siguiente figura en la que también se especifica el resultado de cada una de esas tareas.



De forma resumida a continuación se comenta cada una de las tareas a realizar:

- Investigación: el objetivo de esta fase 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. Como resultado genera los siguientes conjuntos de requisitos:
 - Requisitos de información: describen los datos que debe almacenar el sistema.
 - Requisitos de procesos: describen el comportamiento que debe tener el sistema, es decir qué se desea que realice el sistema.
 - Requisitos de interfaz: describen cómo se desea que se comporte el sistema durante la ejecución de la aplicación.

Los dos primeros conjuntos de requisitos se deben tener en cuenta en las fases de diseño de la base de datos mientras que los requisitos de interfaz serán relevantes durante el desarrollo de la aplicación (es por esto que estos requisitos pueden ser considerados más adelante ya que no se utilizan en las fases iniciales).

Aunque existen algunas metodologías para ayudar en la realización de esta tarea de investigación, no se va a presentar ninguna específica. La forma más elemental de abordar esta tarea consiste en hablar con los que serán los usuarios finales para descubrir cómo es el sistema y cómo quieren que sea. El

estudio de la documentación manipulada por la empresa, si es que ya está funcionando, también puede ser de ayuda (facturas, fichas de clientes, listado de productos, etc.).

- Diseño conceptual estático: es la parte del diseño conceptual que se centra en la obtención del diagrama entidad relación y en la detección de propiedades del sistema que no pueden expresarse en el propio diagrama.
- Análisis de restricciones añadidas (en el nivel conceptual).
- Especificación de las transacciones (en el nivel conceptual): esta tarea consiste en la descripción de las transacciones que van a permitir satisfacer los requisitos de procesos descubiertos en la fase de análisis en términos de los objetos del esquema conceptual, para ello, las restricciones de integridad tendrán que ser tenidas en cuenta. Dado que, sin demasiada dificultad, esta tarea puede posponerse al nivel lógico, no se realizará. Sin embargo, en caso de desear especificar las transacciones a este nivel, en el nivel lógico la especificación de las transacciones quedaría reducida a una traducción de las aquí diseñadas.
- Diseño lógico estático: parte del diseño lógico que consiste en la obtención de un esquema relacional a partir del diagrama entidad relación y en la especificación de las restricciones de integridad necesarias para que el esquema sea correcto. Estas restricciones, denominadas restricciones de transformación (R_t) surgen en los casos en que el diagrama de Entidad-Relación no puede ser representado fielmente mediante el esquema lógico y necesita de algunos ajustes para abordar exactamente la estructura plasmada en el diagrama de Entidad-Relación.
- Análisis de restricciones (en el nivel lógico). El análisis de restricciones incluye las restricciones del nivel conceptual además de las restricciones de transformación. En los apartados siguientes se detallan las fases más relevantes de la metodología como son el análisis de restricciones en el nivel conceptual, el análisis de restricciones en el nivel lógico y la especificación de transacciones.
- Especificación de las transacciones (en el nivel lógico).
- Diseño físico: esta tarea tiene como objetivo definir cómo almacenar la información (físicamente) y definir caminos de acceso para mejorar el rendimiento de las aplicaciones.
- Especificaciones de interfaz.

2.1 Análisis de restricciones en el nivel conceptual

La tarea del análisis de restricciones consiste en detectar para cada una de las restricciones añadidas, cuáles son las operaciones relevantes y en decidir qué tratamiento aplicar, esto es, decidir qué hacer, cuando ocurra alguna de esas operaciones. Este tratamiento se integrará en las transacciones.

Aunque el análisis de las restricciones añadidas podría posponerse al nivel lógico, se dificultaría mucho; es por eso que se propone realizarlo a nivel conceptual ya que el esquema sobre el que se analizan las restricciones es mucho más expresivo que el esquema relacional.

El análisis de estas restricciones consta de tres actividades:

- 1) Averiguar qué objetos del diagrama (entidades o relaciones) se ven afectados por la restricción.
- 2) Descubrir qué operaciones sobre esos objetos son *relevantes* para la restricción. Se dice que una operación es relevante para una restricción si su ejecución puede generar una base de datos en la que la restricción no se cumpla (es decir que se viole). En el caso de las restricciones dinámicas la operación relevante suele ir implícita en la propia restricción.
- 3) Elegir el tratamiento a realizar en caso de que se detecte una operación relevante. Este tratamiento puede ser de tres naturalezas:
 - *Restrictivo*: definiendo qué condición debe cumplirse para que la operación sea aceptada. En caso de que no se cumpla, la operación será rechazada. Este tratamiento tendrá la forma: “*Comprobar...*”
 - *Restaurador*: añadiendo operaciones a ejecutar. Este tratamiento tendrá la forma: “*Insertar...*” o “*Borrar...*” o “*Modificar...*” y también puede llevar condiciones asociadas.
 - *Prohibitivo*: este tratamiento no implica que la operación no pueda ser ejecutada nunca, sino que no se permite que sea ejecutada por el usuario. Sin embargo, sí que puede aparecer en las transacciones, por ejemplo, al incluirse tratamientos restauradores. Este tratamiento siempre tendrá la forma: “*Prohibido*”.

Es importante darse cuenta que, aunque existen intentos de automatizar la detección de operaciones relevantes para una restricción dada, la metodología que aquí se propone realiza esta tarea de forma *manual*, es decir el diseñador debe estudiar cuidadosamente la restricción y *descubrir* qué operaciones pueden ponerla en peligro.

2.2 Análisis de restricciones en el nivel lógico

En el paso del diagrama entidad relación a un esquema relacional, es posible que aparezcan nuevas restricciones dado que el modelo relacional de datos es menos expresivo en general que el modelo entidad relación. A estas restricciones se les llamará *restricciones de transformación* (R_t).

En el análisis de restricciones a nivel lógico deberán tratarse dos tipos de restricciones:

- Restricciones externas o de atributo derivado: en este caso hay que adaptarse el análisis realizado a nivel conceptual al esquema relacional obtenido. Hay que determinar para cada operación relevante a nivel conceptual cuáles son las operaciones equivalentes a nivel lógico adaptando adecuadamente el tratamiento propuesto.
- Restricciones de transformación: dado que son nuevas restricciones, el análisis hay que realizarlo desde el principio.
 - Restricción de existencia: En general, cuando una entidad sufre una restricción de existencia respecto a una relación que en el esquema relacional está representada en una relación (léase *tabla*) distinta a la suya, esta restricción no se puede representar y debe incluirse una restricción de transformación.
 - Generalización total y solapada
 - Generalización parcial y disjunta
 - Generalización total y disjunta

2.3 Especificación de transacciones

Esta tarea consiste en la descripción de las transacciones que van a permitir satisfacer los requisitos de procesos descubiertos en la fase de análisis, para ello, las restricciones de integridad tendrán que ser tenidas en cuenta.

La metodología que se propone para diseñar las transacciones es, pues, la siguiente:

1. Analizar con profundidad el requisito de proceso.
2. Identificar las relaciones del esquema relacional involucradas en el proceso y el tipo de manipulación que van a sufrir, es decir, las operaciones que se va a ejecutar sobre ellas.
3. Identificar las restricciones de integridad afectadas por el proceso al ser relevantes para ellas, algunas de las operaciones detectadas en el paso anterior.
4. Aplicar el tratamiento descrito en el análisis de las restricciones. Si alguno de estos tratamientos supone la ejecución de alguna operación restauradora, entonces:

- comprobar que la operación restauradora no está ya incluida en la transacción, y
- volver al paso 3.

5. Especificar la transacción.

La especificación de la transacción se puede realizar de dos formas distintas:

- Con un lenguaje algorítmico fácil de entender en el que se pueden incluir si se desea instrucciones de control, de selección, de iteración, etc. Las operaciones sobre la base de datos se deben especificar en SQL (SELECT INSERT, DELETE, UPDATE). Además es aconsejable introducir alguna instrucción que de manera sencilla controle los posibles errores que se puedan producir durante la ejecución de la transacción (la instrucción EXCEPCIÓN significará la interrupción y aborto de la transacción).
- En lenguaje natural describiendo qué deberá hacer la transacción. También se pueden incluir sentencias de SQL (sobre todo SELECT) y de nuevo es aconsejable describir cómo se gestionarán los errores.

3 Análisis

3.1 Descripción del problema

Una empresa dedicada a la comercialización y mantenimiento de proyectores de cine precisa de una aplicación informática para la difusión de los contenidos que se vayan a exhibir en los cines en los que distribuye su producto; además de proporcionar una plataforma a dichos cines para la gestión de los contenidos que se vayan a emitir en sus salas.

De este modo, es necesario crear un Sistema de Información que gestione los datos de todos los cines a los que la empresa distribuye sus proyectores, así como la programación que en ellos se exhiba.

Para ello, se necesita almacenar toda la información relativa a los cines: identificador, cif, nombre, teléfono, dirección, logo corporativo, así como su latitud y su longitud para geolocalizarlos en el mapa. Además, se deberá tener en cuenta que un cine podrá disponer de una o más salas y que de éstas será necesario conocer su nombre, capacidad, hora de apertura de la sala, tiempo de descanso entre sesiones y si dispone de soporte digital o 3D.

En dichas salas se exhibirán una serie de contenidos que podrán ser retransmisiones en directo o películas. De dichos contenidos se desea almacenar: identificador, título, director, duración, año, edad recomendada, género al que pertenece, país de producción, distribuidora, si el contenido se emite en versión original, si se emite en digital o en 3D, el cartel anunciador, el trailer y su web oficial, una breve descripción del contenido, la fecha a partir de la cual se podrá emitir dicho contenido y si el mismo es o no en directo. Los contenidos en directo podrán ser culturales: conciertos, operas, obras de teatro, ballets; o acontecimientos deportivos. En el caso de los primeros se almacenará el nombre del compositor y el de los intérpretes además del lugar desde dónde se emite; en el caso de los deportes tan sólo se guardará el lugar de origen y el deporte a exhibir. Por otro lado, tendremos las películas que se exhibirán en los cines del que tan sólo guardaremos su 'imdb'.

Los contenidos programados en la parrilla de cada sala se denominarán eventos y de ellos se almacenará su identificador, el cine en el que se programan, la sala en cuestión, el contenido programado, la fecha y hora de inicio y quién lo programó. No puede haber eventos programados para una fecha anterior al estreno del contenido. Del mismo modo, no puede haber dos eventos que se solapen en el espacio y en el tiempo.

Se desea tener información de todas las provincias españolas, de las que se guardará su identificador y su nombre. Además, también se desea tener codificadas todas las ciudades españolas que tengan más de 10.000 habitantes.

De este modo, las tareas a realizar por la aplicación serán las siguientes: carga inicial de datos de las tablas ciudad, provincia, país, género y distribuidor; alta y modificación de contenidos; alta y modificación de cines y salas; programación de eventos en los

cines. También se necesita poder publicar la cartelería de un determinado cine de modo que se obtenga la relación de las películas a exhibir.

3.2 Requisitos de información

Cine

- Datos personales: cif y nombre.
- Datos de contacto: dirección, código postal, teléfono, móvil, website, latitud y longitud.
- Logo del cine.

Sala

- Nombre de la sala y cine al que pertenece.
- Capacidad.
- Soporte: digital y 3D.
- Hora de apertura y tiempo de descanso entre sesiones.

Contenido

- Título.
- Características: género, año, duración, país, distribuidor, edad mínima y descripción.
- Trailer, imagen y website.
- Soporte: digital, 3D, versión original.
- Fecha de estreno.
- Emisión en directo.

Evento

- Contenido a emitir.
- Sala donde es emitido.
- Momento de emisión.

Película

- Imdb.

Directo

- Indica qué contenidos son emitidos en directo.

Cultural

- Tipología.

- Intérprete y compositor.
- Lugar desde el que se emite el contenido.

Deportivo

- Tipología.
- Lugar desde el que se emite el contenido.

Distribuidor

- Nombre.

Género

- Nombre.

Ciudad

- Nombre.
- Provincia.
- Latitud y longitud.

Provincia

- Nombre.

País

- Nombre.
- Código.

3.3 Requisitos de procesos

P1: Alta de un nuevo cine. El usuario podrá dar de alta nuevos cines introduciendo todos los datos necesarios para la creación de un nuevo cine.

P2: Modificación de un cine. El usuario tendrá la capacidad de modificar los datos referentes a un cine.

P3: Alta de una sala. El usuario podrá introducir los datos necesarios para la creación de una sala que estará ligada al cine.

P4: Modificación de una sala. El usuario podrá modificar los datos correspondientes a una sala. Esto puede ser debido al cambio en el soporte audiovisual de la sala, la capacidad de la misma o la hora de apertura de dicha sala.

P5: Introducción de un nuevo contenido. El usuario introducirá los datos correspondientes al contenido en cuestión (ya sea película o directo).

P6: Modificación de contenidos. La información acerca de cualquier contenido podrá ser modificada.

P7: Añadir un nuevo evento. El usuario podrá incluir contenidos dentro del calendario de exhibición de una determinada sala.

P8: Modificar un evento. El usuario tendrá la capacidad de modificar la exhibición de un evento.

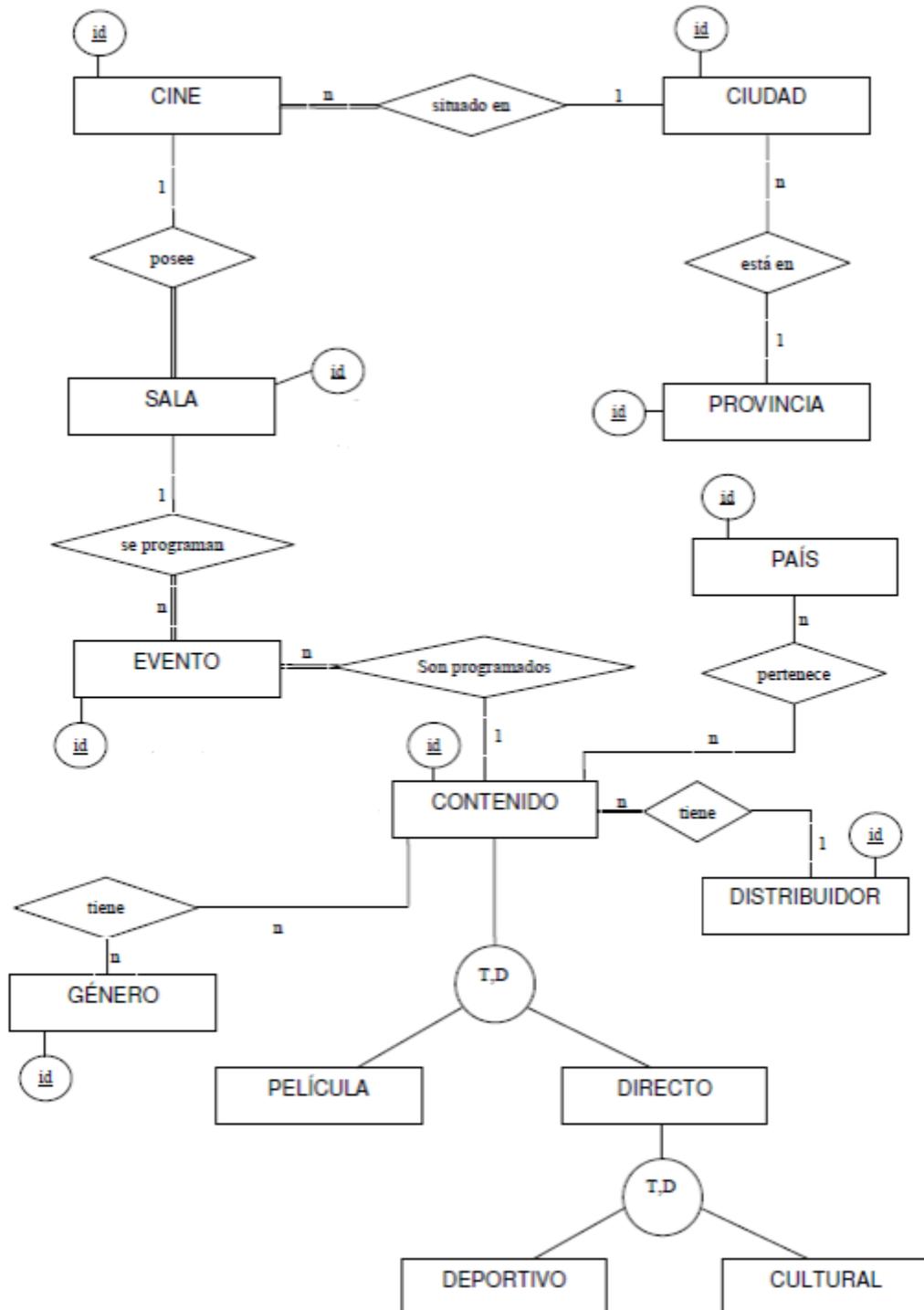
P9: Gestión de la cartelería. El usuario podrá gestionar la cartelería de uno o todos los cines con el fin de publicar los horarios finales de los pases de dicho(s) cine(s) hasta una determinada fecha.

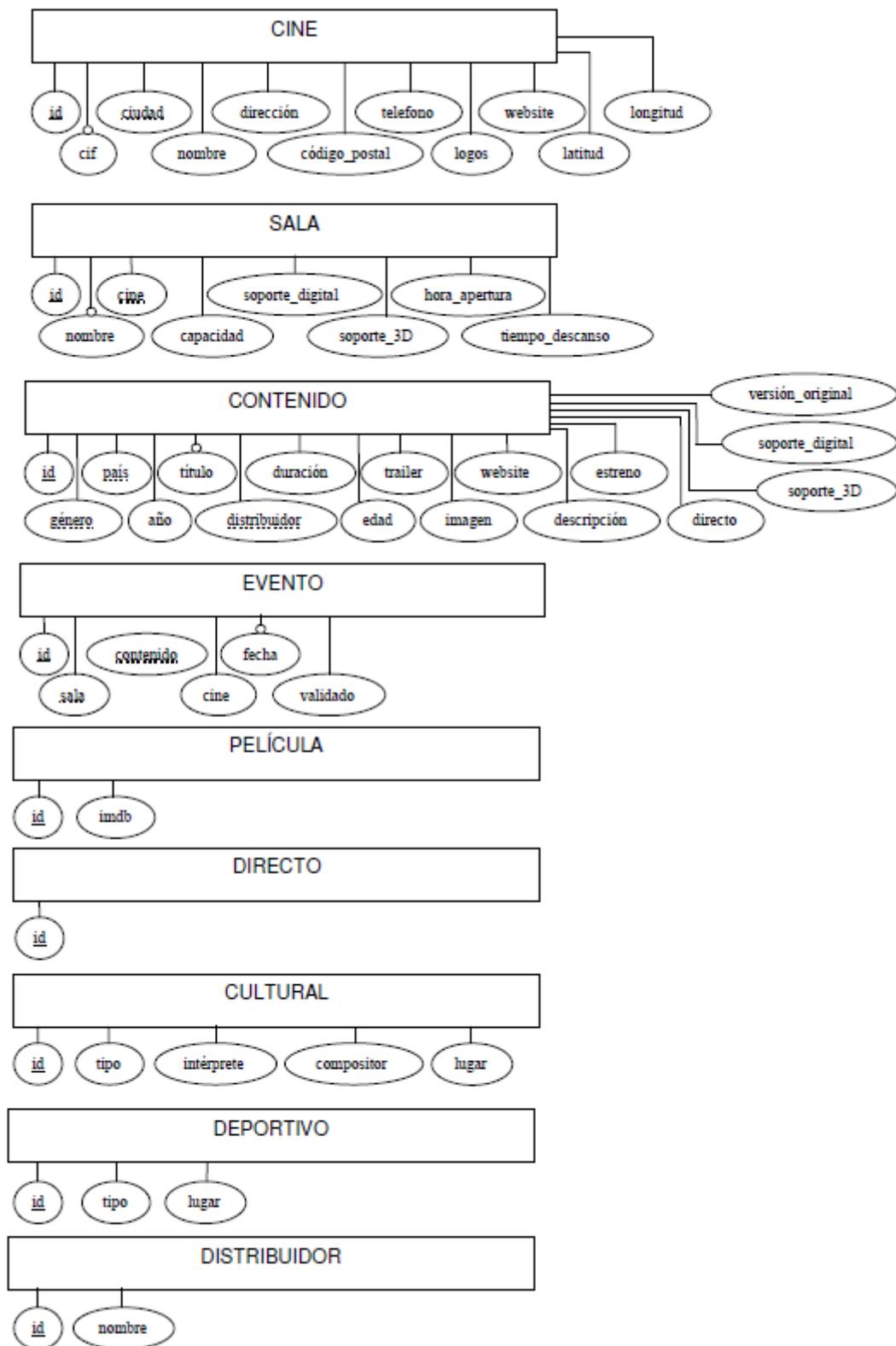
P10: Consulta de cartelería. El usuario podrá consultar qué contenidos se exhiben en un determinado cine, dónde hacen un contenido determinado y los horario de los cines.

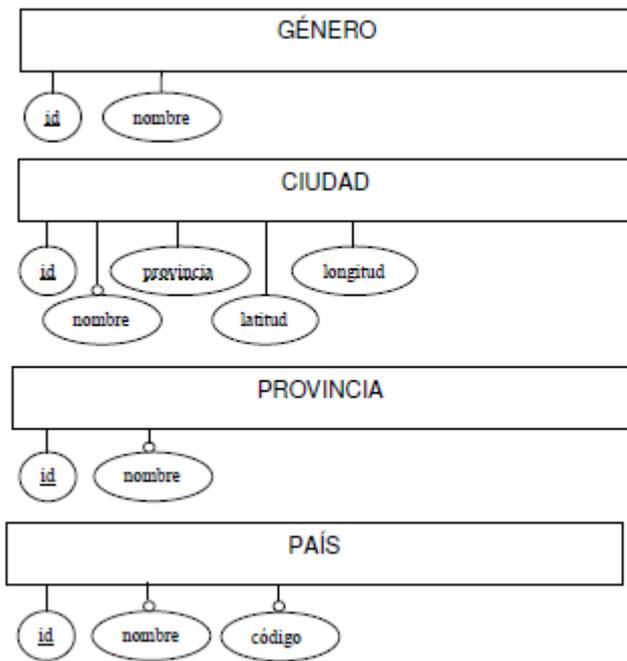
P11: Carga inicial: Al iniciar el sistema se realizará una carga de datos las tablas de ciudad, provincia, país, distribuidor y género.

4 Diseño conceptual

4.1 Diagrama de Entidad-Relación







Listado de restricciones:

RE1: No pueden programarse dos eventos en la misma sala a la misma hora.

RE2: No se puede modificar un evento si ya ha sido publicado en la cartelería.

RE3: No se pueden modificar eventos de fechas anteriores a la actual.

RE4: No se pueden programar eventos antes de la fecha de estreno.

A continuación se describen los atributos de cada objeto, indicando el dominio de datos asignado y las restricciones que posee cada atributo.

Cine (entidad):

<i>Atributo</i>	<i>Dominio</i>	<i>Restricción</i>
Id	Entero	Identificación
Cif	Texto	No nulo
Nombre	Texto	
Ciudad	Entero	No nulo
Dirección	Texto	
Código_postal	Entero	
Telefono	Entero	
Logo	BLOB	
Website	Texto	
Latitud	Texto	
Longitud	Texto	

Sala (entidad):

<i>Atributo</i>	<i>Dominio</i>	<i>Restricción</i>
Id	Entero	Identificación
Nombre	Texto	No nulo
Cine	Entero	No nulo

Capacidad	Entero	
Soporte_digital	Entero	
Soporte_3D	Entero	
Hora_apertura	Texto	
Tiempo_descanso	Entero	

Contenido (entidad):

<i>Atributo</i>	<i>Dominio</i>	<i>Restricción</i>
Id	Entero	Identificación
Título	Texto	No nulo
Género	Entero	
Año	Entero	
País	Entero	
Duración	Entero	No nulo
Distribuidor	Entero	No nulo
Edad	Entero	
Trailer	Texto	
Imagen	BLOB	
Website	Texto	
Descripción	Texto	
Estreno	Texto	

Directo	Entero	
Soporte_3D	Entero	
Soporte_digital	Entero	
Version_original	Entero	

Evento (entidad):

<i>Atributo</i>	<i>Dominio</i>	<i>Restricción</i>
Id	Entero	Identificación
Cine	Entero	
Sala	Entero	No nulo
Contenido	Entero	No nulo
Fecha	Entero	No nulo
Validado	Entero	No nulo

Película (entidad):

<i>Atributo</i>	<i>Dominio</i>	<i>Restricción</i>
Id	Entero largo	Identificación
Imdb	Texto	

Directo (entidad):

<i>Atributo</i>	<i>Dominio</i>	<i>Restricción</i>
Id	Entero	Identificación

Cultural (entidad):

<i>Atributo</i>	<i>Dominio</i>	<i>Restricción</i>
Id	Entero	Identificación
Tipo	Texto	
Intérprete	Texto	
Compositor	Texto	
Lugar	Texto	

Deportivo (entidad):

<i>Atributo</i>	<i>Dominio</i>	<i>Restricción</i>
Id	Entero largo	Identificación
Tipo	Texto	
Lugar	Texto	

Distribuidor (entidad):

<i>Atributo</i>	<i>Dominio</i>	<i>Restricción</i>
Id	Entero	Identificación
Nombre	Texto	

Género (entidad):

<i>Atributo</i>	<i>Dominio</i>	<i>Restricción</i>
Id	Entero	Identificación
Nombre	Texto	

Ciudad (entidad):

<i>Atributo</i>	<i>Dominio</i>	<i>Restricción</i>
Id	Entero	Identificación
Nombre	Texto	No nulo
Provincia	Entero	
Latitud	Entero	
Longitud	Entero	

Provincia (entidad):

<i>Atributo</i>	<i>Dominio</i>	<i>Restricción</i>
Id	Entero	Identificación
Nombre	Texto	No nulo

País (entidad):

<i>Atributo</i>	<i>Dominio</i>	<i>Restricción</i>
Id	Entero	Identificación
Nombre	Texto	No nulo
Código	Texto	No nulo

Situado_en (relación): sin atributos.

Posee (relación): sin atributos

Está_en (relación): sin atributos

Se_programan (relación): sin atributos

PFC: Diseño e implantación de una Base de Datos para la gestión de eventos audiovisuales

Son_programados (relación): sin atributos

Pertenece (relación): sin atributos

Tiene (relación): sin atributos

Tienen (relación): sin atributos

4.2 Análisis de restricciones externas en el nivel conceptual

RE1	Para la misma sala no pueden programarse dos eventos en la misma franja horaria.
Insertar en <i>Evento</i>	Comprobar que no exista un <i>Evento</i> programado en la sala indicada en la misma franja horaria, incluyendo el tiempo de descanso indicado para la sala en cuestión.
Modificar en <i>Evento</i>	Modificar momento en Evento

RE2	Un evento no puede ser ni modificado ni borrado una vez dicho evento haya sido publicado en la cartelería.
Modificar atributos en <i>Evento</i>	Comprobar si el campo <i>validado</i> de <i>Evento</i> no está activado para el evento en cuestión.
Borrar en <i>Evento</i>	Comprobar si el campo <i>validado</i> de <i>Evento</i> no está activado para el evento en cuestión.

RE3	No se puede ni borrar ni modificar un evento antiguo.
Modificar en <i>Evento</i>	Comprobar si la <i>fecha</i> del <i>Evento</i> es anterior a la fecha actual.
Borrar en <i>Evento</i>	Comprobar si la <i>fecha</i> del <i>Evento</i> es anterior a la fecha actual.

RE4	No se puede ni insertar ni modificar un evento con una fecha anterior a la fecha de estreno del contenido a programar.
-----	--

Insertar en <i>Evento</i>	Comprobar si la <i>fecha_estreno</i> del <i>Contenido</i> a programar es posterior a la fecha actual.
Modificar momento en <i>Evento</i>	Comprobar si la <i>fecha_estreno</i> del <i>Contenido</i> a programar es posterior a la fecha actual.

5 Diseño lógico

5.1 Esquema lógico relacional

Cine(id: number, cif: varchar2(20), nombre: varchar2(4000), dirección: varchar2(4000), ciudad: number, codigo_postal: number, teléfono: number, website: varchar2(250), latitud: varchar2(100), longitud: varchar2(100), logos: BLOB)

CP: {id}

CAj: {ciudad} \Rightarrow Ciudad

Borrado restrictivo

Modificación en cascada

VNN: {cif, ciudad}

UNI: {cif}

Sala(id: number, cine: number, nombre: varchar2(4000), capacidad: number, soporte_digital: number, soporte_3D: number, hora_apertura: varchar2(100), tiempo_descanso: number)

CP: {id}

CAj: {cine } \Rightarrow Cine

Borrado en cascada

Modificación en cascada

VNN: {nombre, cine_id}

Contenido(id: number, título: varchar2(500), duración: number, edad: number, país: number, género: number, año: number, distribuidor: number, descripción: varchar2(4000), trailer: varchar2(500), website: varchar2(500), soporte_digital: number, soporte_3D: dom_3d, version_original: number, directo: number, estreno: varchar2(100), imagen: BLOB)

CP: {id}

CAj: {distribuidor} \Rightarrow Distribuidor

Borrado restrictivo

Modificación en cascada

CAj: {género} \Rightarrow Género

Borrado a nulos

Modificación en cascada

CAj: {país} \Rightarrow País

Borrado a nulos

Modificación en cascada

VNN: {título}

Película(id: number, imdb: varchar2(200))

CP: {id}

CAj: {id} \Rightarrow Contenido

Borrado en cascada

Modificación en cascada

Directo(id: number)

CP: {id}

CAj: {id} \Rightarrow Contenido

Borrado en cascada

Modificación en cascada

Cultural(id: number, tipo: varchar2(200), compositor: varchar2(200), interprete: varchar2(1000), lugar: varchar2(200))

CP: {id}

CAj: {id} \Rightarrow Directo

Borrado en cascada

Modificación en cascada

Deportivo(id: number, tipo: varchar2(200), lugar: varchar2(200))

CP: {id}

CAj: {id} \Rightarrow Directo

Borrado en cascada

Modificación en cascada

Distribuidor(id: number, nombre: varchar2(200))

CP: {id}

Género(id: number, nombre: varchar2(200))

CP: {id}

Ciudad(id: number, nombre: varchar2(4000), provincia: number, latitud: number(3,6), longitud: number(3,6))

CP: {id}

CAj: {provincia} \Rightarrow Provincia

Borrado a nulos

Modificación en cascada

VNN: {nombre}

Provincia(id: number, nombre: varchar2(4000))

CP: {id}

VNN: {nombre}

País(id: number, código: varchar2(5), nombre varchar2(200))

CP: {id}

UNI: {codigo}

VNN: {nombre, código}

Evento(id: number, cine: number, sala: number, contenido: number, fecha: number, validado: number)

CP: {id}

CAj: {contenido} \Rightarrow Contenido

Borrado restrictivo

Modificación en cascada

CAj: {sala} \Rightarrow Sala

Borrado en cascada

Modificación en cascada

VNN: {fecha, sala, contenido}

Restricciones:

Rt1: No pueden programarse dos eventos en la misma sala a la misma hora.

Rt2: No se puede modificar un evento si ya ha sido publicado en la cartelera.

Rt3: No se pueden modificar eventos de fechas anteriores a la actual.

Rt4: No se pueden programar eventos antes de la fecha de estreno.

5.2 Análisis de restricciones en el nivel lógico

RE1: No pueden programarse dos eventos en la misma sala a la misma hora.

Conceptual	Lógico	Tratamiento
Insertar en <i>Evento</i>	<ul style="list-style-type: none"> • Insertar <i>Evento</i> 	Comprobar que fecha + duración del contenido a programar + tiempo_descanso <> eventos ya programados
Modificar en Evento	<ul style="list-style-type: none"> • Modificar momento en Evento 	Comprobar que fecha + duración del contenido a programar + tiempo_descanso <> eventos ya programados

RE2: No se puede modificar un evento si ya ha sido publicado en la cartelería.

Conceptual	Lógico	Tratamiento
Modificar atributos en <i>Evento</i>	<ul style="list-style-type: none"> • Modificar <i>Evento</i> 	Comprobar que <i>validado</i> <>1
Borrar en <i>Evento</i>	<ul style="list-style-type: none"> • Borrar <i>Evento</i> 	Comprobar que <i>validado</i> <>1

RE3: No se pueden modificar eventos de fechas anteriores a la actual.

Conceptual	Lógico	Tratamiento
Modificar en <i>Evento</i>	<ul style="list-style-type: none"> • Modificar <i>Evento</i> 	Comprobar que <i>fecha</i> < <i>fecha_actual</i>
Borrar en <i>Evento</i>	<ul style="list-style-type: none"> • Borrar <i>Evento</i> 	Comprobar que <i>fecha</i> < <i>fecha_actual</i>

RE4: No se pueden programar eventos antes de la fecha de estreno.

Conceptual	Lógico	Tratamiento
Modificar momento en <i>Evento</i>	<ul style="list-style-type: none">• Modificar <i>Evento</i>	Comprobar que <i>fecha_estreno</i> (Contenido) < <i>fecha_actual</i>
Borrar en <i>Evento</i>	<ul style="list-style-type: none">• Borrar <i>Evento</i>	Comprobar que <i>fecha_estreno</i> (Contenido) < <i>fecha_actual</i>

Rt1-4: El tratamiento es análogo a las RE 1-4.

5.2.1 Tabla resumida de restricciones

Tabla	Operación relevante	Nº restricción
Evento	Insertar	RE1, RT1
Evento	Modificar atributos	RE2, RT2
Evento	Borrar	RE2, RT2
Evento	Modificar	RE3, RT3
Evento	Borrar	RE3, RT3
Evento	Modificar momento	RE4, RT4
Evento	Borrar	RE4, RT4

5.3 Especificación de transacciones

Proceso 1: Alta de un nuevo cine

Análisis

En el alta de un nuevo cine se ha de introducir su cif y la ciudad en el que está ubicado como datos imprescindibles.

Relaciones y operaciones involucradas en el proceso

Insertar en Cine.

Aplicar tratamiento

El campo cif tiene una restricción de unicidad.

Transacción

INICIO

Si (cif== nulo) o (ciudad == nulo) entonces (*mensaje de error*)

Sino

Insertar cine;

Commit; (en caso de infringir la restricción de unicidad del cif
aparecerá error al hacer commit)

FinSi

FIN

Proceso 2: Modificación de un cine

Análisis

En la modificación de un cine también se tendrá que comprobar que tras las modificaciones realizadas los campos cif y ciudad están rellenos.

Relaciones y operaciones involucradas en el proceso

Modificar en Cine.

Aplicar tratamiento

El campo cif tiene una restricción de unicidad.

Transacción

INICIO

Si (cif== nulo) o (ciudad == nulo) entonces (*mensaje de error*)

Sino

Commit; (en caso de infringir la restricción de unicidad del cif
aparecerá error al hacer commit)

FinSi

FIN

Proceso 3: Alta de una nueva sala

Análisis

En el alta de una nueva sala se ha de introducir su nombre como dato imprescindible.

Relaciones y operaciones involucradas en el proceso

Insertar en Sala.

Transacción

INICIO

Si (nombre== nulo) entonces (*mensaje de error*)

Sino

Insertar sala;

Commit;

FinSi

FIN

Proceso 4: Modificación de una sala

Análisis

En la modificación de una sala se tendrá que comprobar que tras las modificaciones realizadas el campo nombre está relleno.

Relaciones y operaciones involucradas en el proceso

Modificar en Sala.

Transacción

INICIO

Si (nombre== nulo) entonces (*mensaje de error*)

Sino

Commit;

FinSi

FIN

Proceso 5: Introducción de un nuevo contenido

Dependiendo del tipo de contenido (película, cultural o deportivo) elegido se realizarán una serie de comprobaciones u otras.

Relaciones y operaciones involucradas en el proceso

Insertar en Contenido, insertar en Película, insertar en Cultural, insertar en Deportivo.

Transacción

INICIO

Si (titulo== nulo) o (distribuidor== nulo) o (duracion== nulo) entonces (*mensaje de error*)

Sino

Insertar Contenido;

Commit;

Si es Pelicula

Insertar Pelicula;

Commit;

Fin si

Si es Pelicula

Insertar Deportivo;

Commit;

Fin si

Si es Cultural

Insertar Cultural;

Commit;

Fin si

FinSi

FIN

Proceso 6: Modificación de un contenido

Dependiendo del tipo de contenido (película, cultural o deportivo) elegido se realizarán una serie de comprobaciones u otras.

Relaciones y operaciones involucradas en el proceso

Modificar en Contenido, modificar en Película, modificar en Cultural, modificar en Deportivo.

Transacción

INICIO

Si (titulo== nulo) o (distribuidor== nulo) o (duracion== nulo) entonces (*mensaje de error*)

Sino

Commit;

Si es Pelicula

Commit;

Fin si

Si es Pelicula

Commit;

Fin si

Si es Cultural

Commit;

Fin si

FinSi

FIN

Proceso 7: Añadir un nuevo evento

En el alta de un nuevo evento se ha de introducir la fecha, la sala y el contenido como datos imprescindibles.

Relaciones y operaciones involucradas en el proceso

Insertar en Evento.

Restricciones afectadas

Insertar en Evento: RE1

Transacción

INICIO

Si (fecha== nulo) o (sala== nulo) o (contenido== nulo) entonces (*mensaje de error*)

Sino

Comprobar fecha;

Comprobar disponibilidad horaria;

Si disponibilidad horaria

Insertar Evento;

Commit;

FinSi;

FinSi

FIN

Proceso 8: Modificar un evento

En la modificación de un evento se ha comprobar la existencia de la fecha, la sala y el contenido; así como

Relaciones y operaciones involucradas en el proceso

Insertar en Evento.

Restricciones afectadas

Modificar en Evento: RE2, RE3.

Transacción

INICIO

Si (fecha== nulo) o (sala== nulo) o (contenido== nulo) entonces (*mensaje de error*)

Sino

Comprobar fecha;

Comprobar disponibilidad horaria;

Si disponibilidad horaria entonces Commit;

FinSi;

FinSi

FIN

Proceso 9: Gestión de la cartelera

Para gestionar la cartelera bastará con chequear eventos ya programados en los diferentes cines/salas con el fin de certificar qué se va a difundir en cada uno de ellos.

Relaciones y operaciones involucradas en el proceso

No hay.

Transacción

INICIO

Si evento_marcado entonces Validar

FIN

Proceso 10: Consulta de la cartelera

Para consultar la cartelera bastará con seleccionar el cine y la sala correspondientes.

Relaciones y operaciones involucradas en el proceso

No hay.

Transacción

INICIO

Seleccionar cine;

Seleccionar sala;

FIN

Proceso 11: Carga inicial

Antes de inicializar el sistema, se deberán rellenar una serie de tablas que servirán de apoyo para la correcta gestión de toda la aplicación.

Relaciones y operaciones involucradas en el proceso

No hay.

Transacción

INICIO

Insertar en Ciudad;

Insertar en Provincia;

Insertar en País;

Insertar en Distribuidor;

Insertar en Género;

FIN

6 Desarrollo de la aplicación en Oracle Forms 6i y Oracle Database XE

Para el desarrollo de esta aplicación se ha decidido utilizar el servidor de base de datos gratuito que provee Oracle en su página web www.oracle.com, así como de la herramienta para la gestión de formularios Oracle Forms 6i.

6.1 Montaje y conexión

A la hora de realizar una correcta conexión y con el fin de evitar una serie de errores comunes en el momento de levantar el servidor, se van a describir una serie de pasos imprescindibles para que todo se desarrolle correctamente.

1 - Instalar primero el Developer Forms 6i

2 - Instalar Oracle XE (Esto con el objetivo de que en las variables de entorno debe de estar primero el path de Oracle XE)

3 - Deben asegurarse que en el registro de windows en:

HKEY_LOCAL_MACHINE\SOFTWARE\ORACLE la variable NET80 quede con C:\ORANT\NET80

4 - En la ruta C:\ORANT\NET80\ADMIN encontraran el TNSNAMES.ORA y SQLNET.ORA hagan backup de estos archivos y reemplacen estos archivos por los que instala Oracle XE que es en la ruta

C:\oraclexe\app\oracle\product\10.2.0\server\NETWORK\ADMIN

5 - Por ultimo deben de ejecutar lo siguiente desde una ventana de DOS

```
C:\>sqlplus /nolog
```

```
SQL> connect / as sysdba
```

```
Conectado.
```

```
SQL> shutdown immediate
```

```
SQL> startup restrict
```

```
SQL> alter database character set internal_use UTF8;
```

```
SQL> shutdown immediate
```

```
SQL> startup
```

Nota: Esto es para que al momento de conectar a Oracle XE no dé el error de windows:

Para conectarse deben de hacerlo con:

User Name: system

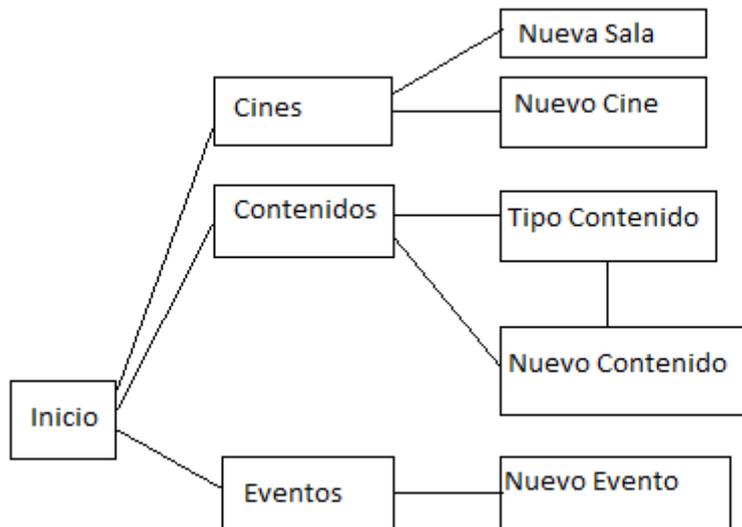
Password : (Password que colocaron al momento de instalar Oracle XE)

database : XE

Siguiendo todos estos pasos, no nos encontraremos con ciertos errores de conexión que provocan no pocos quebraderos de cabeza y que, aunque no lo parezca, no son triviales.

6.2 Jerarquía de módulos

A continuación se presenta un gráfico que visualiza los formularios principales desarrollados para la aplicación



6.3 Documentación de los formularios

- Inicio

Tablas: Ninguna

Controles:

- Botón 'Cines' que nos envía al formulario de 'Cines'.
- Botón 'Contenidos' que nos envía al formulario de 'Contenidos'.
- Botón 'Eventos' que nos envía al formulario de 'Eventos'.
- Botón 'Salir' que nos saca de la aplicación.

- Cines

Tablas: Cine y Sala.

Controles:

- Un listado en el que se muestran los cines existentes con su cif y nombre.
- Un listado en el que se muestran las salas asociadas a cada cine seleccionado en la parte superior, mostrando el nombre de la sala y su capacidad.
- Botón 'Añadir' en la parte superior que nos envía al formulario de Nuevos Cines.
- Botón 'Modificar' en la parte superior que nos envía al formulario de Nuevos Cines y permitirá modificar el cine que previamente haya sido marcado con el check correspondiente.
- Botón 'Consultar' en la parte superior que nos envía al formulario de Nuevos Cines y permitirá consultar el cine que previamente haya sido marcado con el check correspondiente.
- Botón 'Eliminar' en la parte superior que elimina todos aquellos cines cuyo check correspondiente haya sido marcado, a su vez, todas las salas que dependan de ese cine también serán eliminadas.
- Botón 'Volver' que nos envía de nuevo a la pantalla de Inicio.
- Botón 'Añadir' en la parte inferior que nos envía al formulario de Nuevas Salas.
- Botón 'Modificar' en la parte inferior que nos envía al formulario de Nuevas Salas y permitirá modificar la sala que previamente haya sido marcada con el check correspondiente.

- Botón 'Consultar' en la parte inferior que nos envía al formulario de Nuevas Salas y permitirá consultar la sala que previamente haya sido marcada con el check correspondiente.
- Botón 'Eliminar' en la parte inferior que elimina todas aquellas salas cuyo check correspondiente haya sido marcado.

- Nuevo Cine
Tabla: Cines.
Controles:
 - Ocho campos para los diferentes datos a rellenar.
 - Lista desplegable con las ciudades de la tabla ciudades.
 - Campo BLOB que permitirá guardar la imagen del logo del cine.
 - Botón 'Examinar' que permite la navegación por nuestro sistema de datos para subir la imagen que se desee.
 - Botón 'Añadir' que permitirá añadir un nuevo cine.
 - Botón 'Modificar' que permitirá modificar los datos de un cine existente.
 - Botón 'Cancelar' que nos devuelve a la pantalla de Cines.

- Nueva Sala
Tabla: Salas.
Controles:
 - Cinco campos para los diferentes datos a rellenar.
 - Lista desplegable con los tipos de soporte digital.
 - Lista desplegable con los tipos de soporte 3D.
 - Botón 'Añadir' que permitirá añadir una nueva sala.
 - Botón 'Modificar' que permitirá modificar los datos de una sala existente.
 - Botón 'Cancelar' que nos devuelve a la pantalla de Cines.

- Contenidos
Tablas: Contenido.

Controles:

- Un listado en el que se muestran los contenidos existentes con su título, nombre y las características técnicas del contenido.
- Botón ‘Añadir’ en la parte superior que nos envía al formulario de Tipos de Contenido.
- Botón ‘Modificar’ que nos envía al formulario de Nuevo Contenido y permitirá modificar el contenido que previamente haya sido marcado con el check correspondiente.
- Botón ‘Consultar’ que nos envía al formulario de Nuevo Contenido y permitirá consultar el contenido que previamente haya sido marcado con el check correspondiente.
- Botón ‘Eliminar’ que elimina todos aquellos contenidos cuyo check correspondiente haya sido marcado.
- Botón ‘Volver’ que nos envía de nuevo a la pantalla de Inicio.

- Tipo de Contenido

Tablas: Ninguna

Controles:

- Botón ‘Película’ que nos envía al formulario de Nuevo Contenido.
- Botón ‘Deportivo’ que nos envía al formulario de Nuevo Contenido.
- Botón ‘Cultural’ que nos envía al formulario de Nuevo Contenido.
- Botón ‘Volver’ que nos envía al formulario de Contenido.

- Nuevo Contenido

Tabla: Contenido, Deportivo, Cultural y Película.

Controles:

Dependiendo del botón pulsado en el tipo de contenido, tendremos visibles unos campos u otros. En total, y sin tener en cuenta el botón pulsado en el formulario anterior, la pantalla Nuevo Contenido tiene los siguientes campos:

- Doce campos para los diferentes datos a rellenar.
- Lista desplegable con las distribuidoras existentes en la tabla Distribuidor.

- Lista desplegable con los países existentes en la tabla País.
- Lista desplegable con los géneros existentes en la tabla Géneros.
- Tres listas desplegables con las características técnicas del contenido.
- Campo BLOB que permitirá guardar la imagen del cartel del contenido.
- Botón ‘Examinar’ que permite la navegación por nuestro sistema de datos para subir la imagen que se desee.
- Botón ‘Añadir’ que permitirá añadir un nuevo contenido.
- Botón ‘Modificar’ que permitirá modificar los datos de un contenido existente.
- Botón ‘Cancelar’ que nos devuelve a la pantalla de Contenido.

- Eventos

Tablas: Evento, Cines y Salas.

Controles:

- Lista desplegable con los cines existentes en la tabla Cines.
- Lista desplegable con las salas existentes en la tabla Salas dependiendo del Cine que haya sido seleccionado en la lista anterior.
- Un listado en el que se muestran los eventos programados con su fecha de emisión y el contenido a visualizar, dependiendo del cine y la sala seleccionados en la parte superior.
- Botón ‘Añadir’ que nos envía al formulario de Nuevo Evento.
- Botón ‘Modificar’ que nos envía al formulario de Nuevo Evento y permitirá modificar el evento que previamente haya sido marcado con el check correspondiente.
- Botón ‘Consultar’ que nos envía al formulario de Nuevo Evento y permitirá consultar el evento que previamente haya sido marcado con el check correspondiente.
- Botón ‘Eliminar’ que elimina todos aquellos eventos cuyo check correspondiente haya sido marcado.
- Botón ‘Validar’ que marcará como validados todos aquellos eventos con el check correspondiente haya sido marcado.
- Botón ‘Volver’ que nos envía de nuevo a la pantalla de Inicio.

- Nuevo Evento

Tabla: Evento.

Controles:

- Lista desplegable con los contenidos existentes en la tabla de Contenidos.
- Campos de cine, sala y fecha en la que se programará el evento.
- Botón 'Añadir' que permitirá añadir un nuevo Evento.
- Botón 'Modificar' que permitirá modificar los datos de un evento existente.
- Botón 'Cancelar' que nos devuelve a la pantalla de Eventos.

7 Implementación

Debido a que la funcionalidad de todos los botones es análoga en cada uno de los formularios, con el fin de mejorar la usabilidad de la aplicación, a continuación se van a detallar las diferentes funcionalidades tanto de botones, como de rellenado interno de listas y de los procedimientos internos utilizados en los diferentes formularios.

Por tanto, sólo mostraremos el funcionamiento de cada uno de ellos sin la necesidad de especificarlos para todos y cada uno de los formularios para no caer en la redundancia.

7.1 Rellenado de listas

- Lista de países

Para rellenar la lista de países se deben seguir los siguientes pasos:

- Dentro de las propiedades del elemento de lista declararemos que es un elemento de base de datos indicando a que columna de base de datos apunta dicha lista.
- Crearemos una función denominada `llena_lista_pais` que servirá para recuperar todos los elementos de la tabla País y que nos servirán para rellenar la lista. Para ello, utilizaremos el siguiente código:

```
Cursor c1 is
    Select nombre, id from país order by nombre;
Begin
    Clear_list(lista);
    Open c1;
    Loop
        Contador := contador + 1;
        Fetch c1 into v_desc, v_cod;
        If c1%notfound then
            If contador = 1 then message('No se encontraron datos'); end if;
            Exit;
        End if;
        Add_list_element(lista, contador, v_desc, v_cod);
    End loop;
    Close c1;
End;
```

Por último, llamaremos a la función cuando abramos el formulario en el `When_new_form_instance`.

7.2 Botones.

- Botón Añadir

Funciona lanzando el disparador `when_button_pressed` y dentro de éste se ejecuta el código:

```
Global_modos := 'I';  
Call_form('..\nuevo_cine.fmx');
```

- Botón Modificar

Funciona lanzando el disparador `when_button_pressed` y dentro de éste se ejecuta el código:

```
Marcados := comprobar_marcados;  
  
If marcados = 1 then  
    Global_modos := 'M';  
    Registro_seleccionado;  
    Limpiar_checks;  
    Call_form('..\nuevo_cine.fmx');  
End if;
```

- Botón Consultar

Funciona lanzando el disparador `when_button_pressed` y dentro de éste se ejecuta el código:

```
Marcados := comprobar_marcados;  
  
If marcados = 1 then  
    Global_modos := 'C';  
    Registro_seleccionado;  
    Limpiar_checks;  
    Call_form('..\nuevo_cine.fmx');  
End if;
```

- Botón Eliminar

Funciona lanzando el disparador `when_button_pressed` y dentro de éste llama a la función `proc_eliminar`.

- Botón Examinar

Funciona lanzando el disparador `when_button_pressed` y dentro de éste se ejecuta el código:

```
Declare  
    C_archivo varchar2(1000);  
    C_dir varchar2(1000);  
    Dirname varchar2(255);
```

```
V_filename varchar2(255);  
Begin  
  Tool_env.getvar('c:\', dirname);  
  Dirname := dirname || '\bin\bmp';  
  V_filename := get_file_name (dirname,Null, 'All files  
    (*.*)|*.*|' || 'Bitmap Files (*.bmp)|*.bmp|');  
  If v_filename is not null then  
    Read_image_file(v_filename, 'ANY', 'CINE.LOGOS');  
  End if;  
  :GLOBAL.Ruta := v_filename;  
  Message(:global.ruta);  
  Pause;  
End;
```

7.3 Procedimientos y funciones

- Procedimiento Recupera_Registro

Este procedimiento se utiliza en todos los formularios donde se recupera un registro para su posterior modificación o consulta.

```
If :global.modos = 'C' or :global.modos = 'M' then
    Execute_query;
End if;
```

- Procedimiento Rellena_titulo

Este procedimiento se utiliza en el formulario de Nuevo_Evento y como el nombre indica rellena el campo del titulo que se haya seleccionado de la lista de contenidos.

```
Select titulo into :contenido.titulo_nom from contenido
where id = :global.numcontenido;
:contenido.titulo := :global.numcontenido;
```

- Procedimiento Recupera_Nombre_Cine

Este procedimiento se utiliza en el formulario de Nuevo_Evento y como el nombre indica rellena el campo correspondiente con el nombre del cine seleccionado.

```
Select nombre into :control.cine from cine
where id = :global.numcine;
```

- Procedimiento Rellena_Sysfecha

Este procedimiento se utiliza en el formulario de Nuevo_Evento y guarda los valores de la fecha actual para su posterior comparación con el fin de controlar las restricciones especificadas anteriormente.

```
Select to_number(to_char(sysdate, 'DD')) into :global.sysdia from dual;
Select to_number(to_char(sysdate, 'MM')) into :global.sysmes from dual;
Select to_number(to_char(sysdate, 'YYYY')) into :global.sysaño from dual;
```

- Función Fecha_Antigua

Esta función devuelve un 1 si la fecha que se le pasa es anterior a la actual, y 0 si es posterior.

```
Anterior := 0;
If (:evento.año = :global.sysaño) then
```

```
    If (:evento.mes < :global.sysmes) then
        Anterior := 1;
    Elsif (:evento.mes = :global.sysmes) then
        If (:evento.dia < :global.sysdia) then
            Anterior = 1;
        End if;
    End if;
End if;
Return anterior;
```

- **Función Calcula_Días**

Esta función calcula los días que han pasado de un año, será utilizada dentro de otras funciones para realizar cálculos más complejos.

```
FUNCTION calcula_tiempo(anyo number, mes number, dia number, hora number, minuto
number) RETURN NUMBER IS
total number := 0;
cuantos_anyos number := 0;
cuantos_dias number := 0;
BEGIN
    cuantos_anyos := anyo - 2011;
    cuantos_dias := calcula_dias(mes,dia);
    total := (cuantos_anyos * 365 * 24 * 60) +
        (cuantos_dias * 24 * 60) +
        (hora * 60) + minuto;
    return total;
END;
```

- **Función Calcula_Tiempo**

Esta función calculará periodos de tiempo con la finalidad de decidir si un evento es o no programable según las restricciones especificadas.

```
FUNCTION calcula_tiempo(anyo number, mes number, dia number, hora number, minuto
number) RETURN NUMBER IS
total number := 0;
cuantos_anyos number := 0;
cuantos_dias number := 0;
BEGIN
    cuantos_anyos := anyo - 2011;
```

```
    cuantos_dias := calcula_dias(mes,dia);
    total := (cuantos_años * 365 * 24 * 60) +
             (cuantos_dias * 24 * 60) +
             (hora * 60) + minuto;
    return total;
END;
```

- Procedimiento Cuando_acaba

Este procedimiento calcula la hora de finalización de un evento que se intenta programar.

```
PROCEDURE cuando_acaba IS
    numminutos number;
    numhoras number;
    numdias number;
    nummes number;
    numanyo number;
    horas_acum number := 0;
    dias_acum number := 0;
    mes_acum number := 0;
    anyo_acum number := 0;
BEGIN
    -- Calculamos lo que dura el contenido en horas y minutos
    select trunc(:global.metraje/60)
    into numhoras
    from dual;
    select mod(:global.metraje,60)
    into numminutos
    from dual;

    -- Calculamos el horario de fin de evento
    if (:evento.minutos + numminutos > 60) then
        horas_acum := 1;
        :global.minutofin := :evento.minutos + numminutos - 60;
    else
        :global.minutofin := :evento.minutos + numminutos;
    end if;
    if (:evento.horas + numhoras + horas_acum > 23) then
```

```
        dias_acum := 1;
        :global.horafin := :evento.horas + numhoras - 23;
else
        :global.horafin := :evento.horas + numhoras + horas_acum;
end if;

if (dias_acum = 0) then
        :global.diafin := :evento.dia;
        :global.mesfin := :evento.mes;
        :global.anyofin := :evento.anyo;
else
        if (:evento.dia < 28) then
                :global.diafin := :evento.dia + dias_acum;
                :global.mesfin := :evento.mes;
                :global.anyofin := :evento.anyo;
        elsif (:evento.dia = 28) then
                if (:evento.mes = 2) then
                        mes_acum := 1;
                        :global.diafin := 1;
                        :global.mesfin := :evento.mes + mes_acum;
                        :global.anyofin := :evento.anyo;
                else
                        :global.diafin := :evento.dia + dias_acum;
                        :global.mesfin := :evento.mes;
                        :global.anyofin := :evento.anyo;
                end if;
        elsif (:evento.dia = 29) then
                :global.diafin := :evento.dia + dias_acum;
                :global.mesfin := :evento.mes;
                :global.anyofin := :evento.anyo;
        elsif (:evento.dia = 30) then
                if ((:evento.mes = 4) or
                (:evento.mes = 6) or
                (:evento.mes = 9) or
                (:evento.mes = 11)) then
                        mes_acum := 1;
                        :global.diafin := 1;
                end if;
        end if;
end if;
```

```

        :global.mesfin := :evento.mes + mes_acum;
        :global.anyofin := :evento.anyo;
    else
        :global.diafin := :evento.dia + dias_acum;
        :global.mesfin := :evento.mes;
        :global.anyofin := :evento.anyo;
    end if;
elseif (:evento.dia = 31) then
    mes_acum := 1;
    if (:evento.mes = 12) then
        anyo_acum := 1;
        :global.diafin := 1;
        :global.mesfin := 1;
        :global.anyofin := :evento.anyo + anyo_acum;
    else
        :global.diafin := 1;
        :global.mesfin := :evento.mes + mes_acum;
        :global.anyofin := :evento.anyo;
    end if;
end if;
end if;
END;
```

- **Función Es_programable**

Esta función devuelve un 1 si un evento es programable dependiendo de las restricciones establecidas o 0 en caso contrario.

```
FUNCTION es_programable RETURN NUMBER IS
    cabe number := 1;  contador number := 0;  cont_id number := 0;  cont_dur number := 0;
    ev_id number := 0;  ev_hor number := 0;  ev_min number := 0;  ev_dia number := 0;
    ev_mes number := 0;  ev_anyo number := 0;  numhoras number := 0;  numminutos
number := 0;
    numdias number;  nummes number;  numanyo number;  horas_acum number := 0;
    dias_acum number := 0;  mes_acum number := 0;  anyo_acum number := 0;
    hor_fin number := 0;  min_fin number := 0;  dia_fin number := 0;  mes_fin number := 0;
    anyo_fin number := 0;  ev_cur_ini number := 0;  ev_cur_fin number := 0;  ev_act_ini
number := 0;
    ev_act_fin number := 0;
```

```
cursor c1 is
select c.id, c.duracion, e.id, e.horas, e.minutos, e.dia, e.mes, e.anyo
from contenido c, evento e
where
    e.contenido = c.id and
    e.sala = :global.num sala and
    e.id <> :global.num evento
order by e.anyo, e.mes, e.dia, e.horas, e.minutos;

BEGIN
open c1;
loop
    contador := contador + 1;
fetch c1 into cont_id, cont_dur, ev_id, ev_hor, ev_min, ev_dia, ev_mes, ev_anyo;
if c1%notfound then
    if contador = 1 then
        message('No se encontraron datos');
    end if;
    exit;
end if;
-- Calculamos lo que dura el contenido en horas y minutos
        select trunc(cont_dur/60)
        into numhoras
        from dual;
        select mod(cont_dur,60)
        into numminutos
        from dual;

    horas_acum := 0;
    días_acum := 0;
    mes_acum := 0;
    anyo_acum := 0;

    -- Calculamos el horario de fin de evento
    if (ev_min + numminutos > 60) then
        horas_acum := 1;
```

```
        min_fin := ev_min + numminutos - 60;
else
        min_fin := ev_min + numminutos;
end if;
if (ev_hor + numhoras + horas_acum > 23) then
        dias_acum := 1;
        hor_fin := ev_hor + numhoras - 23;
else
        hor_fin := ev_hor + numhoras + horas_acum;
end if;

if (dias_acum = 0) then
        dia_fin := ev_dia;
        mes_fin := ev_mes;
        anyo_fin := ev_anyo;
else
        if (ev_dia < 28) then
                dia_fin := ev_dia + dias_acum;
                mes_fin := ev_mes;
                anyo_fin := ev_anyo;
        elsif (ev_dia = 28) then
                if (ev_mes = 2) then
                        mes_acum := 1;
                        dia_fin := 1;
                        mes_fin := ev_mes + mes_acum;
                        anyo_fin := ev_anyo;
                else
                        dia_fin := ev_dia + dias_acum;
                        mes_fin := ev_mes;
                        anyo_fin := ev_anyo;
                end if;
        elsif (ev_dia = 29) then
                dia_fin := ev_dia + dias_acum;
                mes_fin := ev_mes;
                anyo_fin := ev_anyo;
        elsif (ev_dia = 30) then
                if ((ev_mes = 4) or
```

```

        (ev_mes = 6) or
        (ev_mes = 9) or
        (ev_mes = 11)) then
            mes_acum := 1;
            dia_fin := 1;
            mes_fin := ev_mes + mes_acum;
            anyo_fin := ev_anyo;
        else
            dia_fin := ev_dia + dias_acum;
            mes_fin := ev_mes;
            anyo_fin := ev_anyo;
        end if;
    elsif (ev_dia = 31) then
        mes_acum := 1;
        if (ev_mes = 12) then
            anyo_acum := 1;
            dia_fin := 1;
            mes_fin := 1;
            anyo_fin := ev_anyo + anyo_acum;
        else
            dia_fin := 1;
            mes_fin := ev_mes + mes_acum;
            anyo_fin := ev_anyo;
        end if;
    end if;
end if;
-- Variables de inicio/fin del evento a insertar/modificar
-- :evento.{minutos, horas, dia, mes, anyo} // :global.{minutofin, horafin,
diafin, mesfin, anyofin}
-- Variables de inicio/fin de los eventos recogidos por el cursor
-- ev_min, ev_hor, ev_dia, ev_mes, ev_anyo // min_fin, hor_fin, dia_fin,
mes_fin, anyo_fin
-- Caso 1 (peli nueva empieza después que acabe la del cursor) --> OK
-- Caso 2 (peli nueva acaba antes que empiece la del cursor) -- > OK
-- Caso 3 (peli nueva termina después que empiece la del cursor y termina
antes que termine esta última) -- > KK
-- Caso 4 (peli nueva empieza antes que empiece la del cursor y termina
después que termine esta última) -- > KK

```

```
-- Caso 5 (peli nueva empieza después que empiece la del cursor y termina
después que esta última) -- > KK
```

```
-- Caso 6 (peli nueva empieza después que empiece la del cursor y termina
antes que esta última) -- > KK
```

```
ev_cur_ini := calcula_tiempo(ev_ano, ev_mes, ev_dia, ev_hor, ev_min);
ev_cur_fin := calcula_tiempo(ano_fin, mes_fin, dia_fin, hor_fin, min_fin);
ev_act_ini                                     :=
calcula_tiempo(:evento.ano, :evento.mes, :evento.dia, :evento.horas, :evento.minutos);
ev_act_fin                                     :=
calcula_tiempo(:global.anyofin, :global.mesfin, :global.diafin, :global.horafin, :global.minutofin);

if (ev_act_ini > ev_cur_fin) then cabe := 0;
elsif (ev_act_fin < ev_cur_ini) then cabe := 0;
elsif (ev_act_ini < ev_cur_ini) then
    if (ev_act_fin < ev_cur_ini) then
        cabe := 1;
        return cabe;
        exit;
    elsif (ev_act_fin > ev_cur_fin) then
        cabe := 1;
        return cabe;
        exit;
    end if;
elsif (ev_act_ini > ev_cur_ini) and (ev_act_ini < ev_cur_fin ) then
    cabe := 1;
    return cabe;
    exit;
end if;
end loop;
close c1;
return cabe;
END;
```

- **Función Comprobar_disponibilidad**

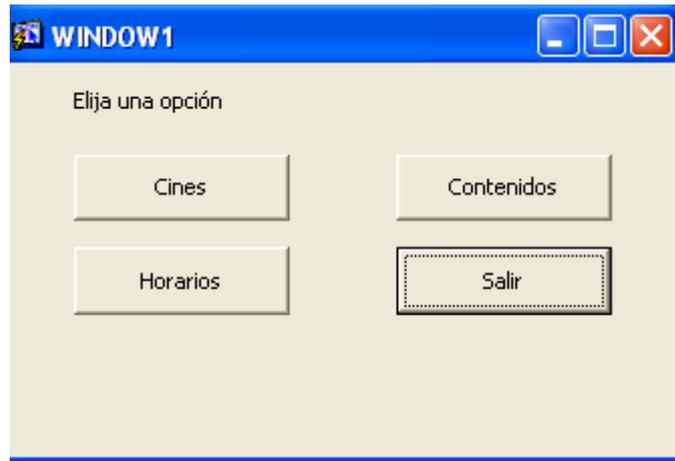
Esta función decide si un evento puede programarse o no en un determinado horario.

```
FUNCTION comprobar_disponibilidad RETURN NUMBER IS
disponible number := 0;
BEGIN
    --Recuperar duración contenido
    If :global.modo = 'I' then
        Select duración into :global.metraje
        From contenido where id = :contenido.titulo;
    Else
        Select duración into :global.metraje
        From contenido where id = :global.numcontenido;
    End if;
    --Calcular fecha fin de evento
    Cuando_acaba;
    Disponible := es_programable;
    Return disponible;
END;
```

8 Manual de usuario

Con la finalidad de mejorar el aprendizaje de los usuarios de la aplicación, se ha intentado, en la medida de lo posible, que todas los formularios tengan un aspecto semejante. Además, también se ha intentado que la funcionalidad de los botones sea la misma independientemente de las tablas implicadas en cada transacción.

Pantalla inicial



El menú principal consta de cuatro botones:

- El primer botón, Cines, nos envía al formulario de Cines/Salas.
- El segundo botón, Contenidos, nos envía al formulario de Contenidos.
- El tercer botón, Horarios, nos envía al formulario de Eventos.
- El cuarto botón, Salir, sale de la aplicación.

Formulario Cines/Salas

Listado de Cines	
Cif	Nombre
<input type="checkbox"/> A12345678	Cines A
<input type="checkbox"/> A98765432	Cines B
<input type="checkbox"/> B38765432	Cines C
<input type="checkbox"/> Z32365432	Cines D
<input type="checkbox"/> B11112222	Cines Grandes
<input type="checkbox"/> A987654321	Cines PP
<input type="checkbox"/>	

Salas	
Nombre	Capacidad
<input type="checkbox"/> Sala 4	255
<input type="checkbox"/> 1	500
<input type="checkbox"/> 2	200
<input type="checkbox"/> 3	150
<input type="checkbox"/>	

El bloque superior muestra el listado de cines existente en la aplicación, cada uno de los cines que allí aparecen lleva asociado un campo de check para marcar el cine en cuestión.

A la derecha del listado de cines aparecen una serie de botones, la funcionalidad se describe a continuación:

- Botón Añadir: nos envía al formulario de Nuevo_Cine, dicho formulario aparece en blanco tal y como muestra la figura.

The screenshot shows a Windows application window titled "WINDOW1". Inside the window is a form for adding a new cinema. The form has the following fields and controls:

- Cine**: A label above the form.
- Cif**: A text input field.
- Nombre**: A text input field.
- Dirección**: A text input field.
- Ciudad**: A dropdown menu.
- Código Postal**: A text input field.
- Teléfono**: A text input field.
- Website**: A text input field.
- Latitud**: A text input field.
- Longitud**: A text input field.
- Logos**: A label above a large empty rectangular area.
- Examinar**: A button located to the right of the Logos area.
- Añadir**: A button located on the right side of the form.
- Cancelar**: A button located below the Añadir button.

Es obligatorio rellenar, al menos, los campos de cif y elegir una ciudad del listado, tras realizar esta acción el botón *Añadir* nos permitirá el almacenamiento de un nuevo cine, siempre y cuando se respete la restricción de unicidad para el cif.

El botón *Examinar* permite buscar en nuestro sistema de archivos una imagen para asociarla al nuevo cine como logo del mismo.

Mediante el botón de *Cancelar* volveremos a la pantalla de Cines.

- Botón *Modificar*: tras chequear en alguno de los cines del listado, nos envía al formulario de Nuevo_Cine, dicho formulario aparecerá con los datos almacenados previamente para dicho cine.

The screenshot shows a Windows application window titled "WINDOW1". The window contains a form for editing a cinema record. The form has the following fields and values:

- Cif: A12345678
- Nombre: Cines A
- Dirección: C/Mayor 15
- Ciudad: Valencia
- Código Postal: 46008
- Teléfono: 961234567
- Website: www.miweb.com
- Latitud: 0,0
- Longitud: 40,0

There is a "Logos" section with a button labeled "Examinar" and a small image of a penguin. On the right side of the form, there are two buttons: "Modificar" and "Cancelar".

Como se puede observar, la diferencia con respecto al botón de Añadir, además de la recuperación de datos, el botón que aquí aparece es el de *Modificar*, pudiendo realizar la acción que dicho botón indica.

El funcionamiento del resto de botones es análogo al caso anterior.

- Botón Consultar: tras chequear en alguno de los cines del listado, nos envía al formulario de Nuevo_Cine, dicho formulario aparecerá con los datos almacenados previamente para dicho cine, pero como se puede observar en la figura siguiente aquí tanto el botón de *Modificar* como el de *Examinar* están deshabilitados puesto que hemos entrado mediante la opción de consulta.

WINDOW1

Cine

Cif A12345678

Nombre Cines A

Dirección C/Mayor 15

Ciudad Valencia Código Postal 46008 Teléfono 961234567

Website www.miweb.com

Latitud 0.0 Longitud 40.0

Logos

Examinar

Modificar

Cancelar

- Botón Eliminar: elimina los cines del listado cuyos checks hayan sido marcados, a su vez, todas las salas asociadas a cada uno de los cines marcados también serán eliminadas.
- Botón Volver: vuelve a la pantalla de Inicio.

El bloque inferior muestra el listado de salas asociado al cine seleccionado del listado de cines, cada una de las salas que allí aparecen lleva asociado un campo de check para marcar la sala en cuestión.

A la derecha del listado de salas aparecen una serie de botones, la funcionalidad se describe a continuación:

- Botón Añadir: nos envía al formulario de Nueva_Sala, dicho formulario aparece en blanco tal y como muestra la figura.

The screenshot shows a window titled "WINDOW1" with a form for adding a new cinema hall. The form is titled "Sala" and contains the following fields and controls:

- Cine: Cines A
- Nombre: (empty text box)
- Capacidad: (empty text box)
- Soporte Digital: (dropdown menu)
- Soporte 3d: (dropdown menu)
- Hora Apertura: (empty text box)
- Tiempo Descanso: (empty text box)

On the right side of the form, there are two buttons: "Añadir" and "Cancelar".

Es obligatorio rellenar el campo de nombre, tras realizar esta acción el botón *Añadir* nos permitirá el almacenamiento de una nueva sala.

Mediante el botón de *Cancelar* volveremos a la pantalla de Cines.

- Botón Modificar: tras chequear en alguna de las salas del listado, nos envía al formulario de Nuevo_Sala, dicho formulario aparecerá con los datos almacenados previamente para dicha sala.

The screenshot shows a window titled "WINDOW1" with a form for modifying an existing cinema hall. The form is titled "Sala" and contains the following fields and controls:

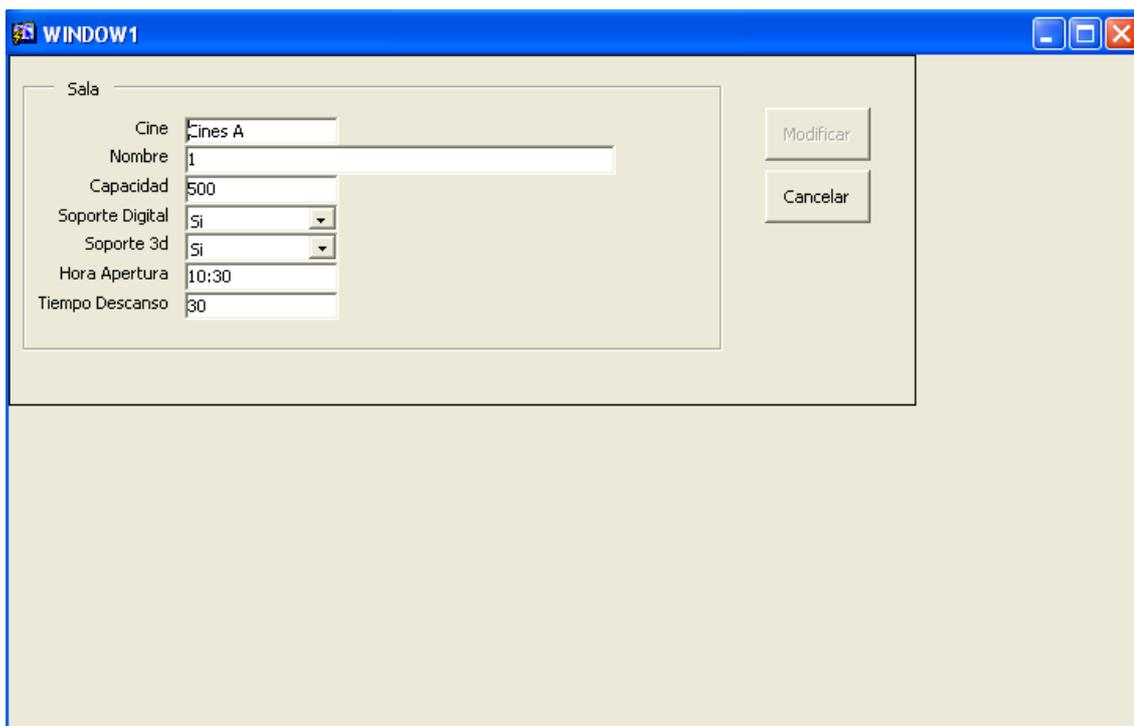
- Cine: Cines A
- Nombre: 1
- Capacidad: 500
- Soporte Digital: Si
- Soporte 3d: Si
- Hora Apertura: 10:30
- Tiempo Descanso: 30

On the right side of the form, there are two buttons: "Modificar" and "Cancelar".

Como se puede observar, la diferencia con respecto al botón de Añadir, además de la recuperación de datos, es que el botón que aquí aparece es el de *Modificar*, pudiendo realizar la acción que dicho botón indica.

El funcionamiento del resto de botones es análogo al caso anterior.

- Botón Consultar: tras chequear en alguna de las salas del listado, nos envía al formulario de Nuevo_Sala, dicho formulario aparecerá con los datos almacenados previamente para dicha sala, pero como se puede observar en la figura siguiente aquí el botón de *Modificar* está deshabilitado puesto que hemos entrado mediante la opción de consulta.



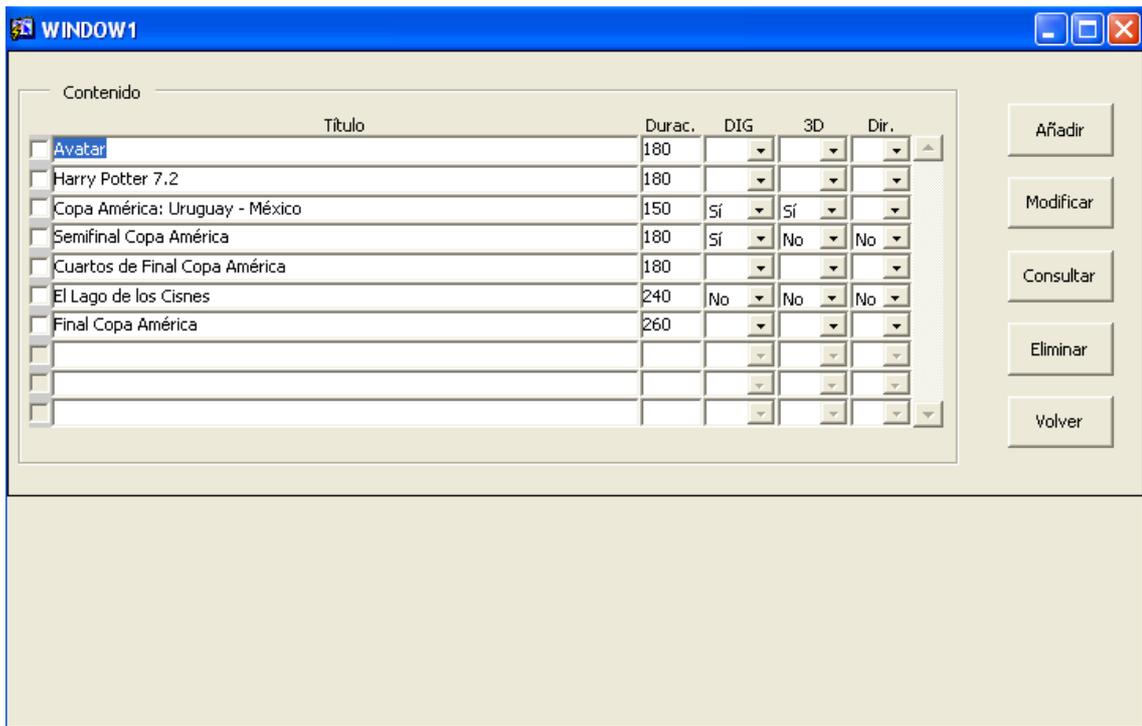
The screenshot shows a Windows application window titled "WINDOW1". Inside the window, there is a form titled "Sala". The form contains the following fields and values:

Field	Value
Cine	Cines A
Nombre	1
Capacidad	500
Soporte Digital	Si
Soporte 3d	Si
Hora Apertura	10:30
Tiempo Descanso	30

To the right of the form, there are two buttons: "Modificar" and "Cancelar". The "Modificar" button is disabled (grayed out).

- Botón Eliminar: elimina las salas del listado cuyos checks hayan sido marcados.

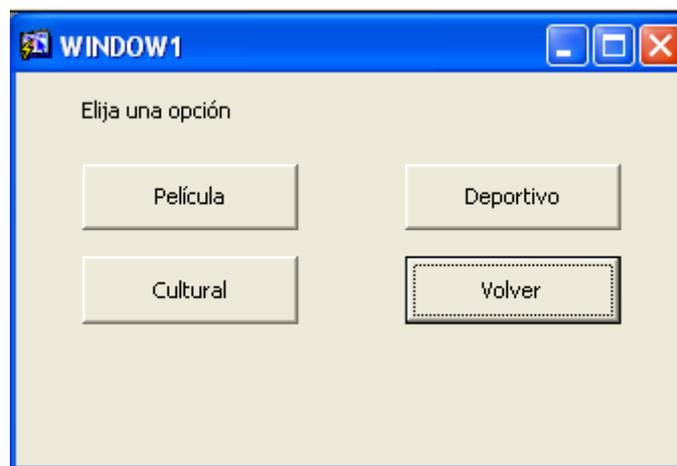
Formulario Contenidos



Se muestra el listado de contenidos existente en la aplicación, independientemente del tipo al que pertenezcan, cada uno de los contenidos que allí aparecen lleva asociado un campo de check para marcar el contenido en cuestión.

A la derecha del listado aparecen una serie de botones, la funcionalidad se describe a continuación:

- Botón Añadir: nos envía al formulario de Tipos_de_Contenido, tal y como muestra la figura.



El botón de *Película* nos envía al formulario de Nuevo_Contenido, pero mostrando los campos específicos para ese tipo de contenido y ocultando aquellos que no le son propios.

The screenshot shows a window titled 'WINDOW1' with a 'Contenido' tab. The form contains the following fields and controls:

- Título: Text input field.
- Duración: Text input field.
- Edad: Text input field.
- Año: Text input field.
- Distribuidor: Text input field.
- Descripción: Text input field.
- Trailer: Text input field.
- Website: Text input field.
- Versión Original: Dropdown menu.
- Directo: Dropdown menu.
- Imagen: Large empty rectangular area.
- Examinar: Button.
- Soporte Digital: Dropdown menu.
- Soporte 3d: Dropdown menu.
- Género: Dropdown menu.
- País: Dropdown menu.
- IMDB: Text input field.
- Añadir: Button.
- Cancelar: Button.

El botón de *Deportivo* nos envía al formulario de Nuevo_Contenido, pero mostrando los campos específicos para ese tipo de contenido y ocultando aquellos que no le son propios.

The screenshot shows a window titled 'WINDOW1' with a 'Contenido' tab. The form contains the following fields and controls:

- Título: Text input field.
- Duración: Text input field.
- Edad: Text input field.
- Año: Text input field.
- Distribuidor: Text input field.
- Descripción: Text input field.
- Trailer: Text input field.
- Website: Text input field.
- Versión Original: Dropdown menu.
- Directo: Dropdown menu.
- Imagen: Large empty rectangular area.
- Examinar: Button.
- Soporte Digital: Dropdown menu.
- Soporte 3d: Dropdown menu.
- Género: Dropdown menu.
- País: Dropdown menu.
- Tipo: Text input field.
- Lugar: Text input field.
- Añadir: Button.
- Cancelar: Button.

El botón de *Cultural* nos envía al formulario de Nuevo_Contenido, pero mostrando los campos específicos para ese tipo de contenido y ocultando aquellos que no le son propios.

The screenshot shows a Windows application window titled "WINDOW1" with a standard title bar (minimize, maximize, close buttons). The main content area is a form titled "Contenido". The form is organized into several sections:

- Top Section:** Fields for "Título", "Duración", "Edad", "Año", and "Distribuidor".
- Middle Section:** Fields for "Descripción", "Trailer", "Website", "Versión Original", and "Directo".
- Right Section:** Fields for "Soporte Digital", "Soporte 3d", "Género", and "País".
- Bottom Section:** Fields for "Tipo", "Lugar", "Compositor", and "Intérpretes".

Buttons are located on the right side of the form: "Añadir" (top), "Cancelar" (middle), and "Examinar" (bottom, near the "Tipo" field). A large empty rectangular area is present on the left side of the form, likely for an image or logo.

Para cada uno de los tres tipos es obligatorio rellenar los campos de 'Título', 'Duración' y 'Distribuidor', tras realizar esta acción el botón *Añadir* nos permitirá el almacenamiento de un nuevo contenido, del tipo previamente elegido.

El botón *Examinar* permite buscar en nuestro sistema de archivos una imagen para asociarla al nuevo contenido como cartel del mismo.

Mediante el botón de *Cancelar* volveremos a la pantalla de Cines.

El botón de *Volver* nos envía de vuelta al formulario de Contenido.

- Botón *Modificar*: tras chequear en alguno de los contenidos del listado, nos envía al formulario de Nuevo_Contenido, dicho formulario aparecerá con los datos almacenados previamente para dicho contenido y, tal y como ocurría con el formulario de Nuevo_Cine con el botón *Modificar* en lugar del de *Añadir*.

The screenshot shows a window titled 'WINDOW1' with a content management form. The form is titled 'Contenido' and contains the following fields and controls:

- Título:** Avatar
- Duración:** 180
- Edad:** 13
- Año:** 2010
- Distribuidor:** Aurum
- Descripción:** (empty text area)
- Trailer:** (empty text area)
- Website:** (empty text area)
- Versión Original:** No
- Imagen:** (empty image area)
- Soporte Digital:** Si
- Soporte 3d:** Si
- Género:** Aventuras
- País:** Estados Unidos
- IMDB:** 122456

Buttons: 'Modificar' and 'Cancelar' are located on the right side of the window. An 'Examinar' button is located below the image area.

El funcionamiento del resto de botones es análogo al caso anterior.

- Botón Consultar: tras chequear en alguno de los contenidos del listado, nos envía al formulario de Nuevo_Contenido, dicho formulario aparecerá con los datos almacenados previamente para dicho contenido, pero como se puede observar en la figura siguiente aquí tanto el botón de *Modificar* como el de *Examinar* están deshabilitados puesto que hemos entrado mediante la opción de consulta.

This screenshot is identical to the one above, showing the 'WINDOW1' content management form for 'Avatar'. The only difference is that the 'Modificar' and 'Cancelar' buttons on the right side of the window are now disabled, appearing in a greyed-out state.

Para los tipos Cultural y Deportivo el funcionamiento es completamente idéntico que para el tipo Película, con la única diferencia, anteriormente descrita, de algunos campos que son propios para unos tipos determinados.

- Botón Eliminar: elimina los contenidos del listado cuyos checks hayan sido marcados.

Botón Volver: vuelve a la pantalla de Inicio.

The screenshot shows a Windows application window titled "WINDOW1". The window contains a form with the following elements:

- Contenido:** A dropdown menu with "Avatar" selected.
- Evento:** Five input fields for "Horas" (20), "Minutos" (30), "Dia" (10), "Mes" (9), and "Año" (2012).
- Cine:** An input field containing "Cines A".
- Sala:** An input field containing "1".
- Buttons:** "Añadir" and "Cancelar".

Una vez seleccionado el contenido que se desea programar, se deberá rellenar la hora y la fecha en la que se desea programar el nuevo evento, tras realizar esta acción el botón *Añadir* nos permitirá el almacenamiento de un nuevo evento, siempre y cuando no contravenga las restricciones que previamente se han anotado. En este caso la aplicación realizará directamente el cálculo y le mostrará al usuario un mensaje de error en el caso de que dicho evento no pueda ser programado a esa hora para ese cine y sala.

Mediante el botón de *Cancelar* volveremos a la pantalla de Eventos.

- Botón *Modificar*: tras chequear en alguno de los eventos del listado, nos envía al formulario de *Nuevo_Evento*, dicho formulario aparecerá con los datos almacenados previamente para dicho evento y, tal y como ocurría con el formulario de *Nuevo_Cine* con el botón *Modificar* en lugar del de *Añadir*. Del mismo modo, que con el botón de *Añadir*, en caso de que las modificaciones realizadas creen algún solapamiento horario, la aplicación avisará al usuario, no realizando ninguna modificación de los datos previamente guardados.

WINDOW 1

Contenido

Avatar

Evento

Horas Minutos Dia Mes Año

20 30 10 6 2012

Cine Cines A

Sala 1

Modificar

Cancelar

El funcionamiento del resto de botones es análogo al caso anterior.

- Botón Consultar: tras chequear en alguno de los eventos del listado, nos envía al formulario de Nuevo_Evento, dicho formulario aparecerá con los datos almacenados previamente para dicho evento, aquí el botón de *Modificar* está deshabilitado puesto que hemos entrado mediante la opción de consulta.
- Botón Eliminar: elimina los eventos del listado cuyos checks hayan sido marcados.
- Botón Validar: marca como validados aquellos eventos cuyos checks hayan sido marcados.

Botón Volver: vuelve a la pantalla de Inicio.

9 Conclusiones

9.1 Trabajo realizado y valoración personal

Problemas encontrados y soluciones...

La aplicación realizada ofrece un alto nivel de cumplimiento de los objetivos y requerimientos del cliente, así como de la planificación y las fases establecidas en ésta.

La realización del proyecto de final de carrera ha servido para poder aplicar parte de los conocimientos aprendidos a lo largo de la carrera.

Construir y modificar la aplicación ha ayudado a comprender más exactamente cada una de las fases de desarrollo y las transformaciones que sufre el diseño en el paso entre una fase y la siguiente, transformaciones que hasta la fecha sólo se habían contemplado teóricamente.

Al escoger la plataforma Oracle Forms para el desarrollo he tenido que profundizar en el lenguaje utilizado, para así tener un mejor control de la aplicación. Como ventajas de la plataforma he de señalar el amplio uso de ésta en multitud de ámbitos laborales, como inconveniente señalaré la dificultad implícita que tiene tanto el montaje de un servidor de base de datos y la incompatibilidad durante el desarrollo entre las distintas versiones del producto (el desarrollo tiene incompatibilidades con los sistemas de 64-bits).

Por todo esto, considero que realizar este proyecto ha sido una experiencia muy positiva que me ha enseñado a valorar en su justa medida el desarrollo de toda aplicación (por pequeña que ésta pueda parecer) y la atención mostrada en todas sus fases y en especial en el análisis.

10 Bibliografía

Apuntes de Bases de Datos.

Apuntes de Desarrollo de aplicaciones de bases de datos.

<https://forums.oracle.com>

www.vbaexpress.com

<http://www.orafaq.com>