

TRABAJO FIN UNIDAD 2 -Parques temáticos-

1.INTRODUCCIÓN

Este proyecto consiste en la elaboración de una aplicación que permita gestionar tres parques temáticos diferentes. Para la gestión de dichos parques temáticos diferentes se requerirá que cada uno de ellos disponga de su propia BBDD.

2.APLICACIÓN

La aplicación consiste en la gestión independiente de tres parques temáticos diferentes de Euskadi y deberá realiza para cada uno de ellos **mínimamente** lo siguiente:

- Visualización de un menú principal en el que se presenten los 3 parques temáticos diferentes en cuestión
- Visualización de los datos más relevantes de cada parque temático (nombre, fecha de apertura, dirección...)
- Visualización de un menú en el que se alojen mínimamente las siguientes opciones:
 - Listado de los espectáculos disponibles en cada uno de los parques temáticos diferentes, mostrando datos como el número del espectáculo, el nombre, el aforo, una descripción, el lugar donde se llevará a cabo y el coste (si lo hubiera). Además, se podrá consultar para cada espectáculo su fecha, horario y el empleado responsable del mismo.
 - Listado de los clientes que están dados de alta en cada espectáculo. De cada cliente se debe registrar el dni, el nombre, los apellidos y la edad. Además, por cada cliente se debe poder consultar los espectáculos a los que se ha inscrito y el horario en el que acudirá a dicho espectáculo.
 - Listado de los empleados del parque temático. Por cada empleado se deberá conocer su dni, nombre, su primer apellido, la fecha de nacimiento, la fecha de contratación, la nacionalidad, y el cargo que desempeña en el parque temático. Además se podrá consultar de qué espectáculo es responsable en cada momento.
- Alta de un nuevo empleado.
- Alta de un nuevo espectáculo.
- Alta de un nuevo cliente.
- Baja de un espectáculo existente. Para ello se da al alumno libertad para su gestión. Es decir, el alumno puede eliminar el espectáculo completamente de la BBDD o puede de alguna manera indicar que es un espectáculo antiguo pero permitiendo que igualmente se pudiera acceder a visualizar los clientes que disfrutaron de dicho espectáculo, su horario, quién fue el empleado responsable del mismo, etc.
- Baja de un empleado existente. Para ello se da al alumno libertad para su gestión. Es decir, el alumno puede eliminar el empleado completamente de la BBDD o puede de alguna manera indicar que es un ex empleado que ya no trabaja en el parque temático, pero permitiendo que igualmente se pudiera acceder a visualizar su información.
- Baja de un cliente existente. Para ello se da al alumno libertad para su gestión. Es decir, el alumno puede eliminar el cliente completamente de la BBDD o puede de alguna manera indicar que es un cliente que no tiene ningún espectáculo pendiente de disfrutar; es decir, que ya ha disfrutado de todos en los que se inscribió, pero permitiendo que igualmente se pudiera acceder a visualizar su información, así como los espectáculos a los que acudió.
- Modificación de los datos de un cliente.
- Modificación de datos de un espectáculo.

- Modificación de datos de un empleado
- Listado de los metadatos de los elementos de la una BBDD
 - De la BBDD (nombre, driver, url y cliente)
 - De las tablas de la BBDD (esquema, nombre de la tabla, clave primaria y las columnas que tiene la tabla)
 - Por cada columna de la BBDD (el nombre, el tipo, si es nula)
- La aplicación podrá utilizar la interfaz gráfica que desee, siendo java swing la más recomendada por ser la más conocida entre el alumnado.

3.BASE DE DATOS

El modelo E/R inicial deberá ser construido por el alumno basándose en las instrucciones dadas.

Entidades

La aplicación deberá contar **MÍNIMAMENTE** como base con las siguientes entidades:

- Espectáculo (número de la espectáculo, nombre, aforo, descripción, lugar, y el coste)
- Cliente (dni, nombre, apellidos y edad).
- Empleado (dni, nombre, primer apellido, fecha de nacimiento, fecha de contratación, nacionalidad, y cargo que desempeña).

Relaciones

Deberán ser implementadas por el alumno en función de sus objetivos de desarrollo.

Especificaciones funcionales

A través de la interfaz gráfica, la aplicación debe permitir mínimamente la realización de las siguientes tareas:

- Gestión de los espectáculos (altas, bajas, visualización de los espectáculos y sus horarios, visualización de los responsables de cada espectáculo).
- Gestión de los clientes (altas, bajas, visualización de los espectáculos a los que están inscritos y día y hora en las que se desarrolla dicho espectáculo).
- Gestión de los empleados (altas, bajas, visualización de los espectáculos de los que es responsable, años en la empresa...).
- Parque temático (visualización de los datos del parque temático).

4.REQUISITOS MÍNIMOS

A continuación se disponen los requisitos que se consideran de obligatorio cumplimiento para la realización del proyecto:

- La aplicación deberá realizar mínimamente lo que se muestra en el apartado 2. APLICACION.
- Cada parque temático utilizará una BBDD diferente para su gestión.
- La pantalla inicial dispondrá un menú de acceso para los diferentes parques temáticos.

5.ORIENTACIÓN PARA MEJORAS

A continuación se presentan algunas opciones para la mejora de la aplicación:

- **Bloque 1**
 - Gestión de otro parque temático más
- **Bloque 2**
 - Asociar bonificaciones a cada uno de los clientes. Cada bonificación deberá de tener un código, nombre y una descripción. Cada bonificación tendrá una cuantía económica asociada pero dependerá del cliente que haya recibido la bonificación. Además se almacenará la fecha en la que el cliente consigue dicha bonificación. Se visualizarán las bonificaciones por cada cliente así como el importe total que se ahorrará el cliente en el presente año.
- **Bloque 3**
 - Buscador de clientes en función de diversos criterios (nombre, apellido...)
 - Búsqueda de los empleados en función de diversos criterios (nombre, apellidos, cargo...)
 - Buscador de espectáculos en función de diversos criterios (nombre, código...)
- **Bloque 4**
 - Histórico de cliente/espectáculo: Crear una opción en el menú donde elegido un cliente se listen todos los espectáculos a los que ha acudido a lo largo de su vida.
 - Recaudación de actividades: Añadir una opción que calcule el dinero recaudado por el parque temático a través de los espectáculos ofertados.
 - Calcular la factura anual de un cliente determinado (sumando el coste de los espectáculos en los que está apuntado este año).
 - Mejora de la aplicación de manera gráfica.
 - Listado de los metadatos de los elementos de otra de las BBDD (tal y como está explicado en el punto 2. APLICACIÓN).
 - Pequeña memoria explicando las funcionalidades de la aplicación
- **Bloque 5**
 - Acceso a la aplicación mediante clave
 - Control de sesiones (quién ha accedido y cuándo)

6.CRITERIOS DE EVALUACIÓN

- El cumplimiento de los mínimos descritos otorgará al alumnado a acceder, **como máximo**, a un 6 en su calificación.
- Cualquier mejora propuesta por el alumno se considerará para aumentar esa nota hasta el máximo posible: 10.
- Si hubiera una propuesta diferente de las aquí contempladas, el profesor y el grupo de alumnos deberán acordar la puntuación de la misma.**
- De no ser así, **queda a criterio del profesor puntuar dichas propuestas.**
- La puntuación final del proyecto se dará en conjunto a todo el equipo, siendo **el propio equipo el encargado de repartir dicha puntuación entre los integrantes**. Ejemplo, si el grupo consta de 4 integrantes, la nota será sobre un máximo de 40 puntos, que tendrán que ser repartidos por los integrantes del equipo. Si el grupo constara de 3 integrantes, la puntuación máxima sería de 30 puntos.

7.PRESENTACIÓN DE LA APLICACIÓN

- El trabajo será presentado en clase al profesor explicando las diferentes funcionalidades que éste realiza, y detallando en cada momento la forma y el motivo de su elaboración.
- El trabajo deberá entregarse en un fichero .zip con el nombre grupoX.zip siendo X 1, 2 o 3 en función del número del grupo correspondiente.

•El trabajo deberá incluir:

- El repositorio donde se ha realizado el proyecto de Java
- El modelo Entidad-Relacion
- (Opcional) Una pequeña memoria explicando las funcionalidades de la aplicación.