

## Programación Web

Carlos Lucena Robles [f92luroc@uco.es](mailto:f92luroc@uco.es)

Miguel Raigón Jiménez [i92rajim@uco.es](mailto:i92rajim@uco.es)

Kamal Abdelkader Haddu [i02abhak@uco.es](mailto:i02abhak@uco.es)

Pablo Roldán Puebla [i92ropup@uco.es](mailto:i92ropup@uco.es)

Paloma Romero Delgado [i92rodep@uco.es](mailto:i92rodep@uco.es)

---

## Informe Práctica 2

### 1. Decisiones de diseño e implementación

- 1.1. Cambios respecto a la práctica 1 en cuanto a nivel de grupo:
  - 1.1.1. Mejora en la planificación gracias al uso de Jira Software.
  - 1.1.2. Mayor rapidez en la resolución de fallos en la implementación y toma de decisiones.
- 1.2. Cambios respecto a la práctica 1 en cuanto a nivel de código:
  - 1.2.1. Uso de clases auxiliares para el main.
  - 1.2.2. Todos los mensajes de entrada/salida con el usuario se encuentran en la capa de visualización.
  - 1.2.3. Eliminación del fichero de excepciones de usuario ya que los controles que se hacían en él, correo electrónico válido, fecha de nacimiento válida y mayoría de edad de un usuario, se han implementado en el gestor de usuarios.
  - 1.2.4. Eliminación de los ficheros de excepciones de kart y pista ya que los controles que se hacían en ellos se han llevado a cabo a la hora de introducir por teclado, es decir, en el display de gestor de pista. Entre estos controles están tipo de kart, dificultad de la pista y estado de kart y pista.
- 1.3. Creación de la base de datos:
  - 1.3.1. Se dispone de las tablas Bono, Kart, Pista, Reserva y Usuario con sus correspondientes claves y atributos. En un principio se decidió hacer las tablas de reserva Infantil, Familiar y Adultos. En cambio, se eliminaron porque no es recomendable hacerlo de esta manera puesto que la mayoría de atributos son comunes y estos subtipos no se relacionan entre sí. Por tanto, ahora se dispone de la tabla del supertipo, Reserva, con los atributos del supertipo, los subtipos y el atributo tipo\_Reserva.
  - 1.3.2. Para la reserva de Bono se tiene una tabla independiente. Esta tabla tiene como clave foránea la id de reserva.
  - 1.3.3. Cada vez que se carga el script de la base de datos se hace el borrado de todas las tablas.
- 1.4. Se han añadido las funcionalidades:

- 1.4.1. Eliminar un usuario a partir de su correo electrónico.
- 1.4.2. Búsqueda de un usuario en concreto a partir de su correo electrónico.
- 1.4.3. Listado de todos los karts.
- 1.5. Otras decisiones:
  - 1.5.1. Se ha decidido que un usuario pueda realizar más de una reserva para el mismo día. La otra opción contemplada era no permitir que un usuario haga una nueva reserva teniendo ya una para el mismo día.
  - 1.5.2. Respecto a la factoría se ha mantenido como estaba, con los productos reserva infantil, familiar y adultos.
  - 1.5.3. Se ha establecido un valor de defecto para la fecha de inscripción, en vez de null, ya que se han tenido problemas al coger datos de la base de datos siendo la fecha null.
  - 1.5.4. Para la fecha de las reservas se ha considerado de tipo Date debido a que trabajar con la fecha en tipo String supone más problemas.

## 2. Dificultades

Las dificultades encontradas a lo largo de la realización de la práctica han sido las siguientes:

1. Error al intentar crear la base de datos sin el modelo relacional previo. La solución fue realizar el modelo relacional y modificar la base de datos creada ya que no se contemplaba bien las relaciones y faltaban atributos en las tablas.
2. Pensamos hacer uso de triggers para controlar que si la reserva es de modalidad individual, un idReserva sólo puede aparecer una única vez en Bono, y si la reserva es de modalidad Bono sólo puede aparecer cinco veces como máximo. Sin embargo, los triggers se encontraban bloqueados. Por lo que este control se ha implementado en el código.
3. Se ha considerado no poder modificar el correo electrónico de los usuarios. Debido a que correo es clave primaria, y si el usuario tiene reservas hechas salta error ya que deja de existir al modificarlo. Se solucionará para la siguiente práctica.
4. A pesar de habernos planificado mejor, con más tiempo podríamos haber pulido más el código y haber implementado la modificación de reservas y el borrado de reserva individual.

## 3. Referencias

1. MySQL Documentation Team (2022). "MySQL 8.0 Reference Manual"  
<<https://dev.mysql.com/doc/refman/8.0/en/enum.html>>
2. ChuWiki (2006). "Java:Base de datos"  
<[https://chuwiki.chuidiang.org/index.php?title=Categor%C3%ADa:Java:Base\\_de\\_datos](https://chuwiki.chuidiang.org/index.php?title=Categor%C3%ADa:Java:Base_de_datos)>
3. W3Schools (1999-2022). "SQL CHECK Constraint"  
<[https://www.w3schools.com/sql/sql\\_check.asp](https://www.w3schools.com/sql/sql_check.asp)>
4. Usuario stalepretzel, (2008) . "StackOverFlow"  
<<https://stackoverflow.com/questions/380057/foreign-key-not-working-in-mysql-why-can-i-insert-a-value-thats-not-in-the-for>>