



Programación con Herramientas Modernas

Trabajo práctico 2019

Dominio y precondiciones

El dominio sigue siendo el mismo de la entrega 0. Asumimos que están implementadas las interfaces de usuario de los casos de uso de la entrega anterior.

Parte 1: Persistencia

La aplicación deberá implementar la persistencia mediante un componente de acceso a datos (Repositorio o Home) que trabaje contra un motor de base de datos relacional. Dado que en las próximas entregas lo volveremos a utilizar, lo más recomendable es que simplemente cambie el componente de acceso a datos de la forma más dinámica posible.

El grupo podrá elegir entre

- crear las entidades en un motor de base de datos relacional, utilizar un script de manipulación de datos mediante SQL y hacer el mapeo mediante JDBC
- utilizar un framework de mapeo OR/M (JPA o Hibernate) y generar el juego de datos mediante un script SQL o bien mediante llamadas a queries del lado de objetos (en los tests o en el Bootstrap que provee Arena).

Reponsabilidad grupal

- Definición del modelo de objetos
- Definición de los componentes de acceso a datos
- Definición del script inicial de datos con la siguiente información:
 - Películas, al menos 8 y 1 Saga.
 - Al menos 3 funciones para cada película.
 - Usuarios 5 al menos.
 - Entradas al menos dos por usuario.

Responsabilidad individual

- **Integrante 1:** Consulta de películas y funciones disponibles.
- **Integrante 2:** Consulta de panel de control.
- **Integrante 3:** Login y búsqueda de amigos.

Cada integrante debe ser responsable de encontrar los componentes adecuados para cada caso de uso asignado: application model si correspondiere, objetos de dominio que colaboran, controllers y vista.

Quedan excluidos de la persistencia el carrito de compra que se continuará manejando en memoria y la sugerencias tanto de películas como amigos continúan siendo valores dummies.

En caso de haber menos integrantes se conversará con el tutor los casos de uso a entregar.

Parte 2: Componentes en la base de datos

Se pide que implementen componentes en la base de datos para resolver los siguientes requerimientos:

1. conocer las entradas de películas de acción (nombre, año, valoración) que se dan para una determinada sala.
2. llevar un control de las modificaciones que se hagan sobre una película, de manera de saber: a) la fecha en la que se hizo la modificación, b) el nombre anterior, c) el nombre nuevo
3. saber cuáles son las películas que tienen más de 3 entradas.
4. evitar que el saldo de un usuario tome un valor nulo en la base (por fuera de la interfaz de usuario).

Debe decidir cómo modelar cada uno de los casos. Las opciones son:

- vistas
- triggers
- stored procedures
- constraints

Cada integrante será responsable de un punto y deberá justificar su decisión y explicar su resolución en la entrega presencial.