

POTIONCRAFT

El Videojuego



Alicia Cámara Casares
acamarc02@educarex.es
2024/2025

ÍNDICE

1. OBJETIVO y RESULTADOS DE APRENDIZAJE	2
2. DESCRIPCIÓN DEL VIDEOJUEGO	2
2.1. Requisito funcional - Inicio de la aplicación	4
2.1.1. Nueva partida	4
2.1.2. Cargar partida	4
2.2. Requisito funcional - Programa principal	4
4. SOBRE EL PLAGIO O USO DE IA	8
5. ENTREGA	8
6. CALIFICACIÓN	9

1. OBJETIVO y RESULTADOS DE APRENDIZAJE

Potioncraft es un juego de simulación y gestión donde el jugador asume el papel de un alquimista que administra su propia tienda de pociones, que creará mediante la combinación de ingredientes. Dichos ingredientes los puede conseguir comparándolos a los comerciantes que cada cierto tiempo visitan su tienda. En este proyecto vamos a implementar una versión simplificada del juego, partiendo de los requisitos que se detallan en posteriores apartados.

Esta actividad está orientada a evaluar la consecución de los siguientes resultados de aprendizaje del módulo de Acceso a Datos:

- **RA3. Gestiona la persistencia de los datos identificando herramientas de mapeo objeto relacional (ORM) y desarrollando aplicaciones que las utilizan.**

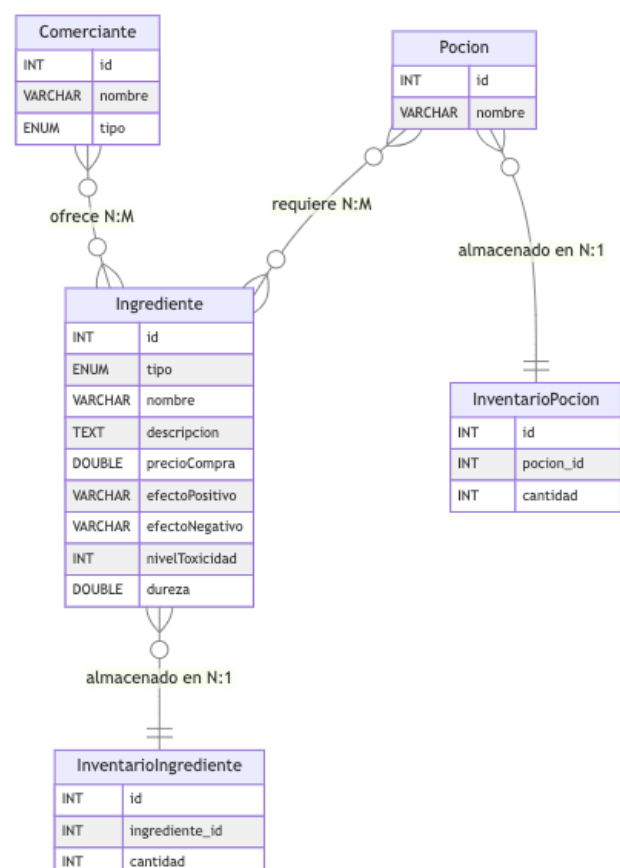
Para ello se emplearán los criterios de evaluación asociados a ellos, tal y como se recogen en el RD405/2023 y supondrá del 35% de la calificación de los mismos.

2. DESCRIPCIÓN DEL VIDEOJUEGO

Nuestra aplicación, Potioncraft, debe gestionar la información de los comerciantes que visitan el laboratorio para vender sus ingredientes y la información de las pociones que se pueden fabricar.

El diagrama entidad/relación de la base de datos potioncraft es el siguiente:

Cada **Poción** está formada por un identificador único y un nombre. Por otro lado, de cada **Ingrediente** conocemos su identificador único, el tipo (hierba, seta o mineral), nombre, etc.



Una Poción se fabrica a partir de un conjunto de Ingredientes, relación que se establece en la tabla intermedia **Ingredientes_Pociones**. Por ejemplo, la poción Curativa necesita una unidad de Flor acuática, una unidad de Espina dorada y una unidad de Seta marrón. Ten en cuenta que necesitamos tener una unidad de TODOS los ingredientes que conforman una poción para poder fabricarla.

De los **Comerciantes** conocemos su identificador único, nombre y tipo (herborista, recolector de setas, mercader o minero). El tipo de comerciante determina el tipo de ingrediente que puede vender:

- Los herboristas solo ofrecen ingredientes de tipo Hierba.
- Los recolectores de setas se limitan a vender ingredientes de tipo Seta.
- Los mineros ofrecen Minerales.
- Los mercaderes ambulantes venden cualquier tipo de ingrediente.

No obstante, de estas restricciones no te tienes que preocupar ya que están establecidas en la base de datos: la tabla **ComercianteIngrediente** registra los ingredientes que venden cada uno de los comerciantes (tabla intermedia resultante de la relación N:M entre Comerciante e Ingrediente).

Las cinco tablas mencionadas hasta ahora ya se proporcionan con datos en el script `potioncraft.sql` y su contenido no debe modificarse durante la ejecución del programa. Por otro lado, almacenamos los Ingredientes que ha comprado el usuario y su cantidad en la tabla **InventarioIngrediente**, al igual que almacenamos las Pociones que ha fabricado en la tabla **InventarioPocion**. Estas dos tablas deberás modificarla cuando ocurran determinados eventos, que será explicados a continuación.

Requisitos imprescindibles:

- **No se cargarán las tablas en memoria** para todo el programa, como hacíamos en StardamValley. En este proyecto, cada vez que necesitemos cierta información debemos hacer una consulta a la tabla correspondiente. El incumplimiento de este requisito supondrá una **penalización de -6 puntos**.
- En memoria solo debemos almacenar la **reputación** del jugador. Inicialmente, la reputación del jugador estará a 0. Dicho valor se almacenará en un **fichero de**

acceso aleatorio al cerrar la aplicación, de forma que se pueda recuperar el estado para volver a jugar.

- En este juego consideramos que **el oro (dinero) es infinito**, por lo que podemos comprar y vender libremente sin necesidad de tener en cuenta el presupuesto.

2.1. Requisito funcional - Inicio de la aplicación

Al igual que en la versión anterior, al iniciar la aplicación, se debe verificar si existe un estado guardado. Si no existe un estado guardado (es decir, no existe el fichero de acceso aleatorio que guarda la reputación), directamente se iniciará una nueva partida. En caso contrario, se presentará el siguiente menú:

```
===== BIENVENIDO A POTIONCRAFT =====  
1. Nueva partida  
2. Cargar partida
```

2.1.1. Nueva partida

Al iniciar una nueva partida, se eliminará el fichero de acceso aleatorio que almacena la reputación. Además, se limpiará el contenido de las tablas `InventarioIngrediente` e `InventarioPocione`. Tras ello, se mostrará el menú principal del juego.

2.1.2. Cargar partida

Carga en memoria la reputación almacenada en el fichero de acceso aleatorio y muestra el menú principal del juego.

2.2. Requisito funcional - Programa principal

Se debe ofrecer el siguiente menú con las opciones:

1. **CREAR POCIONES:** muestra al usuario las pociones que puede elaborar con los ingredientes que dispone, dándole la posibilidad de elegir una de ellas. Recuerda que en la tabla `InventarioIngrediente` se almacenan los ingredientes del usuario y la cantidad disponible. Al seleccionar una poción, el programa la produce (solo una unidad, aunque tenga ingredientes suficientes para fabricar más) y la añade al

inventario del usuario, al mismo tiempo que reduce el stock de los ingredientes utilizados. Ten en cuenta que, si el stock del Ingrediente ha quedado a 0, debe eliminarse el registro de ese ingrediente de la tabla `InventarioIngrediente`. Además, actualiza la reputación del jugador de acuerdo con la siguiente condición:

- a. Si la suma de los efectos positivos de los ingredientes supera a la suma de los efectos negativos, se incrementa la reputación en 1. En caso contrario, se resta 1 a la reputación.

La salida debe ser similar a la siguiente, informando del número de efectos negativos y positivos así como de la nueva reputación del jugador.

```
===== Pociones disponibles para fabricar =====  
1. Agrandar  
  - Blood Ruby (Tipo: MINERAL), - Seta loca (Tipo: SETA),  
2. Destreza  
  - Espina sangrienta (Tipo: HIERBA),  
0. Cancelar  
Seleccione el número de la poción que desea fabricar: 1  
Has seleccionado fabricar: Agrandar  
Reputación actualizada: -1 (Positivos: 0, Negativos: 1)  
Poción fabricada con éxito: Agrandar
```

2. La opción **VENDER POCIONES** permite al jugador vender TODAS las pociones que ha creado. El precio de cada poción se calcula como la suma de los precios de los ingredientes que la componen, más una comisión del 10% por la mano de obra (la comisión debe aplicarse al precio final de venta de la poción). Recuerda eliminar los registros de la tabla `InventarioPocion`.

La salida debe ser similar a la siguiente, informando del oro ganado con cada poción vendida y del oro total que se ha ganado:

```
===== Vendiendo todas las pociones disponibles =====  
Has vendido 1 unidades de Agrandar por 260,04 monedas de oro.  
Has vendido 1 unidades de Destreza por 35,64 monedas de oro.  
Total de ganancias: 295,68 monedas de oro.
```


3. La opción **COMPRAR INGREDIENTES** selecciona un comerciante aleatorio de entre los disponibles en la tabla Comerciante. De todos los ingredientes que puede vender ese comerciante (tabla ComercianteIngrediente), solo pondrá a la venta en esta visita 5 de ellos, seleccionados de forma aleatoria.

Se ofrecerá un menú al usuario que permita comprar todos los ingredientes que quiera hasta que se seleccione la opción de salir. Recuerda que al comprar un ingrediente, debes añadirlo a la tabla InventarioIngrediente; si el ingrediente ya existía, deberás actualizar la cantidad disponible.

La salida debe ser similar a la siguiente, informando del comerciante que visita, del oro gastado en cada ingrediente y del oro total gastado en esa compra al comerciante:

Visita: Aldric Hojaverde (HERBORISTA)

===== Menú de compra =====

Ingredientes disponibles:

1. Espina sangrienta (Tipo: HIERBA) - Precio: 32,40
2. Palo espinoso (Tipo: HIERBA) - Precio: 72,80
3. Espina dorada (Tipo: HIERBA) - Precio: 38,80
4. Romero druídico (Tipo: HIERBA) - Precio: 22,60
5. Terraria (Tipo: HIERBA) - Precio: 13,60
0. Salir

Seleccione el número del ingrediente que desea comprar: **1**

Has seleccionado: Espina sangrienta (Tipo: HIERBA). Precio unitario: 32,40

¿Cuántas unidades desea comprar? **3**

Has comprado 3 unidades de Espina sangrienta por 97,20 monedas de oro.

===== Menú de compra =====

Ingredientes disponibles:

1. Espina sangrienta (Tipo: HIERBA) - Precio: 32,40
2. Palo espinoso (Tipo: HIERBA) - Precio: 72,80
3. Espina dorada (Tipo: HIERBA) - Precio: 38,80
4. Romero druídico (Tipo: HIERBA) - Precio: 22,60
5. Terraria (Tipo: HIERBA) - Precio: 13,60
0. Salir

Seleccione el número del ingrediente que desea comprar: **0**

Fin de la visita de Aldric Hojaverde

ORO TOTAL INVERTIDO: 97 oros.

4. La opción **MOSTRAR ESTADÍSTICAS DEL JUEGO** muestra la reputación del jugador, los ingredientes y cantidades de los que dispone y las pociones y cantidades que tiene.

REPUTACIÓN: -1

INVENTARIO INGREDIENTES:

- Ingrediente: Blood Ruby (Tipo: MINERAL) - Cantidad: 1
- Ingrediente: Shadow Chanterelle (Tipo: SETA) - Cantidad: 2
- Ingrediente: Flor esponjosa (Tipo: HIERBA) - Cantidad: 2
- Ingrediente: Espina sangrienta (Tipo: HIERBA) - Cantidad: 3

INVENTARIO POCIONES:

- Poción: [Agrandar](#) - Cantidad: 1

5. La opción **MOSTRAR COMERCIANTES** muestra el nombre y tipo de los comerciantes junto con el número de ingredientes que comercializa.

Opción seleccionada: MOSTRAR COMERCIANTES

Comerciante: Thalassa Hierbasueños (Tipo: HERBORISTA) - Ingredientes que puede vender: 26
Comerciante: Fynn Bosquefungi (Tipo: RECOLECTOR_SETAS) - Ingredientes que puede vender: 13
Comerciante: Darion Mercaluz (Tipo: MERCADER) - Ingredientes que puede vender: 48
Comerciante: Aldric Hojaverde (Tipo: HERBORISTA) - Ingredientes que puede vender: 26
Comerciante: Drogran Rocaviva (Tipo: MINERO) - Ingredientes que puede vender: 9
Comerciante: Luciano Diamante (Tipo: MERCADER) - Ingredientes que puede vender: 48

6. La opción **MOSTRAR POCIONES** muestra el nombre de las pociones y los ingredientes necesarios para fabricarlas.

Opción seleccionada: MOSTRAR POCIONES

Poción: Ácido

Ingredientes necesarios:

- Healer's Heather (Tipo: HIERBA)
- Blood Ruby (Tipo: MINERAL)

Poción: Adhesivo

Ingredientes necesarios:

- Seta mohosa (Tipo: SETA)
- Cristal vital (Tipo: MINERAL)

7. La opción **SALIR** guarda la reputación en el fichero de acceso aleatorio y finaliza la ejecución del programa.

3. SUGERENCIA ORDEN DE IMPLEMENTACIÓN

Esto solo es una recomendación y no es necesario seguir este orden:

1. Lectura comprensiva del enunciado y desarrollo del diagrama de clases.
2. Determinar las relaciones entre tablas y las necesidades de bidireccionalidad.
3. Implementación de las clases y sus atributos. Anotación de las entidades.
4. Menú para cargar nueva partida y lectura/escritura del fichero de acceso aleatorio.
5. Implementar el menú principal.
6. Implementar las opciones 4 al 6 para comprobar que funciona la conexión.
7. Implementar la opción “Comprar ingredientes”
8. Implementar la opción “Crear pociones”
9. Implementar la opción “Vender pociones”
10. Fase de validación y pruebas.

A partir del paso 4, deberías comprobar que funciona antes de pasar al siguiente paso.

4. SOBRE EL PLAGIO O USO DE IA

Este trabajo está destinado a evaluar el nivel de adquisición de conocimientos de los alumnos de manera individual. Por lo tanto, la detección de material no original (plagio o copia) no permitirá dicha evaluación y acarreará una calificación de 0 (cero). Por los mismos motivos, la detección del uso de Inteligencia Artificial para la realización del trabajo acarreará una calificación de 0 (cero). El alumno podrá ser requerido para defender el proyecto en el formato que el profesor considere.

5. ENTREGA

Se deberá entregar el proyecto de IntelliJ en un archivo comprimido con el nombre del alumno. Para ello deberás usar la opción **FILE-> EXPORT -> PROJECT TO ZIP FILE**.

Además deberás entregar una pequeña memoria en la que indiques cuál ha sido tu estrategia para desarrollar el trabajo, los principales retos que se te han planteado y cómo los has resuelto. Incluye también un comentario sobre el nivel de dificultad de la tarea y si crees que te ha servido para aplicar/afianzar conocimientos.

6. CALIFICACIÓN

Se tendrán en cuenta los siguientes aspectos:

Criterio	Puntuación máx.
Corrección, organización y limpieza del código. Uso correcto y adecuado de JPA. Uso correcto de la Programación Orientada a Objetos.	2 ptos
Inicio: nueva partida / cargar partida	0.5 ptos
Funcionalidad “Crear pociones”	2.5 ptos
Funcionalidad “Vender pociones”	1.5 ptos
Funcionalidad “Comprar ingredientes”	2 ptos
Funcionalidad “Mostrar estadísticas del juego”	0.5 ptos
Funcionalidad “Mostrar comerciantes”	0.5 ptos
Funcionalidad “Mostrar pociones”	0.5 ptos

El objetivo de este trabajo es evaluar el uso de JPA e Hibernate, por lo que si no se utilizan dichas herramientas o se trabaja con las colecciones en memoria, la calificación será 0.