

Trabajo de Fin de Grado

*Grado Superior
en Desarrollo
de Aplicaciones WEB*

***Aplicación web
de compra-venta
de videojuegos***

Óscar Ponce de León Álvarez

Introducción, justificación y objetivos

Actualmente todos tenemos asumido que estamos en la era online. Todo lo puedes encontrar en internet con el simple hecho de abrir un buscador. A nivel personal, un concepto, una noticia, un producto o una curiosidad. A nivel laboral tienes a tu alcance y al de los demás productos, publicidad y comunicación con una comunidad muy amplia y diversa.

A día de hoy es muy difícil encontrar un negocio o empresa que no se encuentre en este mundo online y que, además, haya creado su propio servicio para anunciarse o vender sus productos, sean del tipo que sean.

Son muchos los tipos de ventas online que nos podemos encontrar:

- Catálogos online: no son en sí tiendas, pero permiten presentar los productos del negocio de forma ordenada y accesible dando la posibilidad de promocionarlos al mismo tiempo.
- Tiendas virtuales: son sitios web creados y diseñados específicamente para vender productos o servicios. Una de sus principales características es que siempre está disponible las 24 horas los 7 días de la semana.
- Blogs con tienda online: es similar a las tiendas virtuales con la peculiaridad de que todo es más cercano, familiar, artesanal. Todo lo que se utiliza viene de personas que han comprado o han vendido los productos.
- Dropshipping: más que una tienda virtual se puede considerar un método para vender el producto en una página como intermediario. Uno de los ejemplos que a día de hoy sigue en auge es Amazon.
- Tiendas en redes sociales: Es un método que últimamente está en auge. Los usuarios de las redes sociales utilizan sus perfiles tanto de Instagram, Twitter, Facebook, etc., para mostrar sus productos y luego utilizan aplicaciones o páginas externas para su venta.

Aquella persona o grupo de ellas que por necesidad o por gusto quiera dar a conocer su negocio o quiera entrar en el terreno de la venta de productos cuenta con la gran suerte de poder elegir entre una de estas opciones ya que el mundo de las ventas online es muy amplio y tiene muchas salidas.

El punto de partida del actual trabajo es crear un servicio de compra-venta de productos, aplicable a un tipo de empresa de videojuegos.

El objetivo general es crear una aplicación web de compra-venta de videojuegos.

Para ello, se tendrá que:

1. Generar una base de datos desde cero en la que poder almacenar todos los datos.
2. Gestionar la aplicación web desde la interfaz gráfica.
3. Realizar compras y ventas mediante el acceso de un usuario con dinero digital.

Índice

| | |
|---|----|
| Introducción, justificación y objetivos | 2 |
| Índice | 4 |
| Índice de ilustraciones | 5 |
| Resumen del trabajo | 6 |
| Módulos Formativos aplicados en el trabajo | 7 |
| Herramientas/Lenguajes utilizados en el trabajo | 8 |
| Fases del proyecto. | 12 |
| Conclusiones..... | 25 |
| Bibliografía..... | 26 |

Índice de ilustraciones

| | |
|--|----|
| Ilustración 1 Logo de MySQL..... | 8 |
| Ilustración 2 Logo de Eclipse IDE | 9 |
| Ilustración 3 Logo de XAMPP | 9 |
| Ilustración 4 Logo de Bootstrap | 10 |
| Ilustración 5 Logo de Spring Boot | 10 |
| Ilustración 6 Ejemplo de Java sacado de la aplicación..... | 11 |
| Ilustración 7 Ejemplo de SQL sacado de la creación de la base de datos | 11 |
| Ilustración 8 Ejemplo de HTML sacado de la aplicación | 11 |
| Ilustración 9 Ejemplo de HTML sacado de la aplicación | 11 |
| Ilustración 10 Modelo Entidad-Relación..... | 12 |
| Ilustración 11 Diagrama de casos de uso..... | 17 |
| Ilustración 12 Diagrama de casos de uso..... | 17 |
| Ilustración 13 Resultado final de la aplicación..... | 20 |
| Ilustración 14 Paquete con los controladores | 21 |
| Ilustración 15 Paquetes de los modelos | 22 |
| Ilustración 16 Login.jsp..... | 24 |

Resumen del trabajo

La aplicación web consta de un diseño bastante minimalista e intuitivo que permite fácilmente moverse entre las distintas páginas.

En principio, cuenta con la posibilidad de ver el catálogo de productos e inspeccionarlos sin necesidad de registrarse. Si el usuario quisiera hacer una compra o venta de algún producto existe también la posibilidad de hacerlo mediante registro en la página.

La forma de registrarse contempla también varios tipos de perfiles que controlan distintas funciones dentro de la página.

Estos perfiles son los siguientes:

- Perfil de cliente
- Perfil de propietario
- Perfil de administrador

Como cliente, una vez que se inicia sesión, se puede modificar los datos por si se ha introducido un error o se ha equivocado al registrarse. Ahora siendo cliente de la página tiene la aplicación permite la compra y la venta de productos.

Para la venta y compra de productos, todo se hace desde el carrito de la compra o desde la misma página del producto, no sin antes tener que meter dinero digital a través de la web.

Como propietario, se tiene la capacidad de poder añadir los videojuegos y sus géneros y, además, se tiene también la posibilidad de cambio si en un futuro el producto se quiere cambiar.

Al ser propietario le llegan la lista de las compras y ventas que se han hecho en la página.

Como administrador, puedes administrar los usuarios, saber sus datos y poder cambiar sus perfiles o si se tiene algún problema con dicho usuario eliminarlo.

Módulos Formativos aplicados en el trabajo

Para hacer la aplicación web se utilizarán los conocimientos adquiridos en las asignaturas de Programación, Base de datos, Lenguaje de Marcas, Desarrollo web entorno servidor, Desarrollo web entorno cliente, Despliegue de aplicaciones y Diseño de interfaces web.

Para el desarrollo del Back-end se ha utilizado todos los conocimientos adquiridos en programación sobre el uso de Java, gracias a Base de datos se ha podido utilizar usar esos conocimientos para crear una base de datos SQL donde poder almacenar todos los datos, por último, pero no menos importante, gracias a desarrollo web entorno servidor he podido utilizar los conocimientos adquiridos sobre servidores, sobre todo centrándome en SpringBoot y el como conectar todo lo anterior.

Para el desarrollo del Front-end se ha utilizado los conocimientos de Lenguaje de marcas, Desarrollo web entorno cliente y diseño de interfaces web, para la creación de la estructura HTML responsive, con su CSS, la creación de formularios, etc.

Cabe destacar que, aunque no haya ninguna asignatura sobre Bootstrap, se ha estudiado el uso del mediante la documentación de la propia página principal.

Y por último para el despliegue de la aplicación se ha hecho uso de Despliegue de aplicaciones, hemos utilizado Tomcat para desplegar la aplicación en formato .war.

Herramientas/Lenguajes utilizados en el trabajo

Para la realización de este proyecto se ha utilizado una serie de frameworks y programas, todos de uso gratuito por cualquier público.

MySQL



Ilustración 1 Logo de MySQL

MySQL es un sistema de gestión de base de datos relacionales de código abierto y basado en el lenguaje SQL. Es una aplicación que funciona en todas las plataformas tanto en Windows como en Linux.

El funcionamiento de MySQL se basa en el modelo cliente-servidor. El servidor que utiliza MySQL funciona de dos maneras distintas, como un programa independiente y de manera que se pueda unir a otras aplicaciones.

Este programa permite almacenar y acceder a los datos guardados en ese espacio, también permite la replica de datos y sus usuarios pueden manejarlo mediante el SQL.

Por último, utiliza una seguridad que utiliza contraseñas encriptadas y permite controlar los privilegios de los usuarios y los accesos.

Como dato interesante, MySQL es la base de datos de bastantes sitios web conocidos como Twitter, Facebook o YouTube.

Eclipse IDE



Ilustración 2 Logo de Eclipse IDE

Eclipse es un entorno de desarrollo integrado creado para el desarrollo del software, servicios web, etc. Es un entorno que permite el uso de varios lenguajes, aunque como está desarrollado en Java, este es el lenguaje principal

Eclipse dispone de una interfaz bastante simple e intuitiva al usarlo y gracias a su capacidad de poder instalar plugins, esto lo diferencia de otros IDEs donde no se pueden instalar más funcionalidades

Por último, Eclipse permite el uso de bastantes frameworks para el desarrollo de interfaces gráficas para el usuario, servicios Eclipse

XAMPP



Ilustración 3 Logo de XAMPP

XAMPP es una herramienta de desarrollo que permite a los diseñadores web y programadores probar sus propios programas en un servidor local creado en sus propios ordenadores.

Entre sus ventajas, se puede decir que es un servidor web bastante fácil de usar y no requiere mucha complicación al configurarlo, además esta disponible para todas las plataformas y es gratuito.

Bootstrap

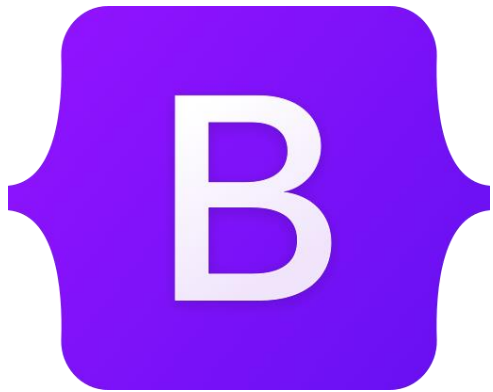


Ilustración 4 Logo de Bootstrap

Bootstrap es uno de los frameworks mas usados en el mundo del desarrollo web. Se centra en el CSS y permite la creación de paginas web con una interfaz completamente limpia y completamente responsive (Se adapta a todo tipo de pantallas, tanto de ordenadores como de móviles).

Entre sus ventajas se encuentra su soporte (dado que tiene una documentación muy extensa sobre su utilización, pero fácil de comprender), la compatibilidad con los principales navegadores de hoy en día (ya que todos lo detectan) y por último permite la integración con JavaScript, lo que permite el uso de botones con efectos, tarjetas, tipografías diferentes, etc.

Spring Boot



Ilustración 5 Logo de Spring Boot

Spring es un framework de código abierto que permite y facilita la creación de aplicaciones en Java, Kotlin y groovy.

Es conocido por sus múltiples módulos los cuales añaden, muchas más funcionalidades como por ejemplo el acceso a datos a base de datos o crear controladores web.

Spring Boot permite la construcción rápida de aplicaciones de spring y que sea mucho más fácil de configurar. Gracias a esto, Spring Boot nos permite enfocarnos únicamente en el código a la hora de la creación.

Tiene por defecto un servidor de aplicaciones embebido y un gestor de dependencias.

En cuanto a los lenguajes de programación, se han usado los siguientes:

Java

Java es un lenguaje de programación orientado a objetos. Gracias a la máquina virtual de Java (JVM), este lenguaje es interpretado y ejecutado. Gracias a Java, se pueden crear distintos programas en muchos dispositivos diferentes, permitiendo la ejecución de la misma aplicación en distintos dispositivos.

```
// Método que permite al cliente crear usuarios
@PostMapping("/registro")
public String procRegistro(Usuario usuarioaux, Perfil perfil, RedirectAttributes rattr) {

    // Creamos usuario
    Usuario usuario = new Usuario();
    usuario.setUsuario(usuarioaux.getUsuario());
    usuario.setNombre(usuarioaux.getNombre());
    usuario.setApellido(usuarioaux.getApellido());
    usuario.setContraseña("{noop}" + usuarioaux.getContraseña());
    usuario.setDireccion(usuarioaux.getDireccion());
    usuario.setEmail(usuarioaux.getEmail());
    usuario.setFechaAlta(new Date());
    usuario.setTelefono(usuarioaux.getTelefono());
    usuario.setDinero(0.00);

    // Añadimos una lista de perfil de usuario
    List<Perfil> lista = new ArrayList<Perfil>();
    Perfil p = pdao.verPerfil(perfil.getIdperfil());
    lista.add(p);
    usuario.setPerfiles(lista);

    // lo creamos
    int reg = udao.crear(usuario);
    if (reg == 0) {
        System.out.println("Usuario no añadido");
        rattr.addFlashAttribute("mensaje", "El usuario no se ha creado/Nombre de usuario ya existe");
    } else {
        System.out.println("Usuario añadido");
        rattr.addFlashAttribute("mensaje", "El usuario se ha creado");
    }

    return "redirect:/";
}
```

Ilustración 6 Ejemplo de Java sacado de la aplicación

SQL

El SQL es un lenguaje de consulta estructurado que permite la manipulación de datos de una base de datos. Es utilizado en muchas empresas que contienen sus datos en una base de datos.

```
CREATE TABLE VIDEOJUEGOS
(IDVIDEOJUEGO INTEGER PRIMARY KEY AUTO_INCREMENT,
TITULO VARCHAR(200) NOT NULL,
DESCRIPCION TEXT NOT NULL,
DISTRIBUIDORA VARCHAR(200) NOT NULL,
PRECIO DOUBLE(9,2),
IMAGEN VARCHAR(255),
IDGENERO INTEGER NOT NULL,
FOREIGN KEY(IDGENERO) REFERENCES GENERO(IDGENERO)
);
```

Ilustración 7 Ejemplo de SQL sacado de la creación de la base de datos

HTML y CSS

HTML significa HyperText Markup Language, que es un lenguaje de marcas reconocido a nivel mundial. Con este lenguaje se crea la mayoría de páginas web que existen a día de hoy. El navegador se encarga de interpretar el lenguaje y mostrarlo de forma clara y limpia.

```
<div class="col-md-12">
  <div class="form-check">
    <input class="custom-control-input" type="checkbox" value=""
      id="invalidCheck" required> <label
        class="custom-control-label fs-5" for="invalidCheck"> Acepto
        los terminos y condiciones </label>
    </div>
  </div>
```

Ilustración 8 Ejemplo de HTML sacado de la aplicación

Para el aspecto del HTML se utiliza el CSS, es un lenguaje de marcas que permite definir el aspecto de una página web y su código se combina con el HTML, lo que hace que un sitio web, o al menos el apartado visual este completo.

Fases del proyecto.

Modelo de datos

Para la creación de la base de datos, primero se ha tenido que saber de qué se iba a hacer la base de datos. En este caso, es sobre la venta y compra de videojuegos. Dada esta necesidad necesitamos una base de datos en donde poder guardar la información del usuario, los videojuegos, los géneros de los videojuegos, las compras, las ventas y por ultimo los perfiles de los usuarios.

A partir de esta premisa, se va a desarrollar un modelo entidad-relación de estas entidades, atributos y relaciones.

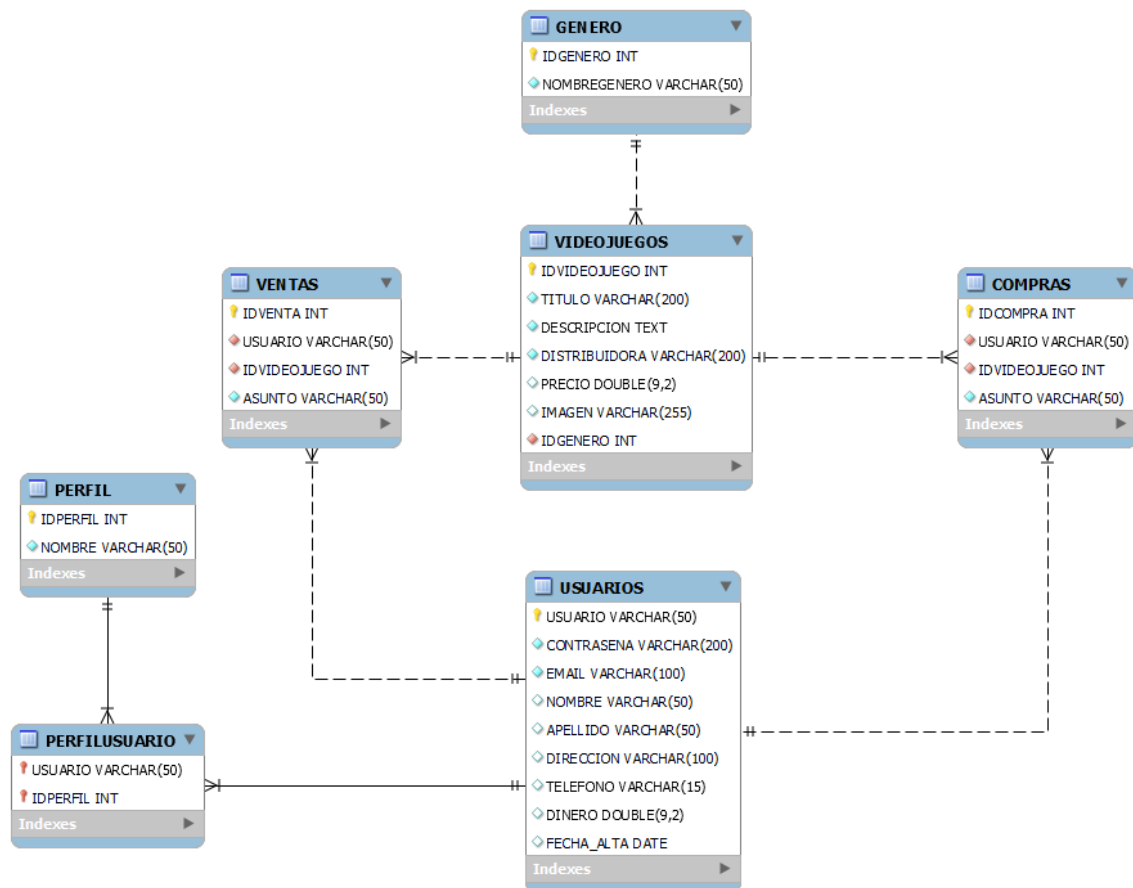


Ilustración 10 Modelo Entidad-Relación

Una vez definido este modelo, se pasará a crear y explicar las tablas de la base de datos.

Tabla usuarios

La tabla usuario contendrá todos los datos del usuario registrado.

| Campo | Tipo | Nulo | Clave primaria | Único | Predeterminado |
|------------|---------------|------|----------------|-------|----------------|
| Usuario | Varchar (50) | No | Si | Si | |
| Contraseña | Varchar (200) | No | No | No | |
| Email | Varchar (100) | No | No | No | |
| Nombre | Varchar (50) | Si | No | No | |
| Apellido | Varchar (50) | Si | No | No | |
| Dirección | Varchar (100) | Si | No | No | |
| Teléfono | Varchar (15) | Si | No | No | |
| Dinero | Double | Si | No | No | |
| Fecha_alta | Date | Si | No | No | |

Los campos en la tabla contienen la siguiente información:

- Usuario: nombre elegido por el usuario que permite iniciar sesión en la aplicación y con el que se identificara.
- Contraseña: contraseña de la cuenta
- Email: email del usuario
- Nombre: nombre del usuario
- Apellido: apellidos del usuario
- Dirección: dirección donde vive el usuario
- Teléfono: teléfono del usuario
- Dinero: dinero del usuario con el que comprara los videojuegos
- Fecha_alta: fecha de alta en la plataforma del usuario

Tabla perfil

Tabla con la información de los distintos perfiles en la aplicación

| Campo | Tipo | Nulo | Clave primaria | Único | Predeterminado |
|----------|--------------|------|----------------|-------|----------------|
| IdPerfil | Integer | No | Si | Si | Auto_increment |
| Nombre | Varchar (50) | No | No | No | |

Los campos de la tabla contienen la siguiente información:

- IdPerfil: id de los perfiles creados
- Nombre: Nombre de los perfiles creados

Tabla perfil usuario

Tabla con la que relacionaremos los usuarios con un perfil en concreto

| Campo | Tipo | Nulo | Clave primaria | Único | Predeterminado |
|----------|--------------|------|----------------|-------|----------------|
| Usuario | Varchar (50) | No | Si | Si | |
| IdPerfil | Integer | No | Si | Si | |

Los campos de la tabla contienen la siguiente información:

- Usuario: Usuario del usuario
- IdPerfil: Id del perfil asignado

En esta tabla encontramos dos claves primarias pero que a la vez son dos claves foráneas que hacen referencia a las tablas de usuario y de Perfil.

Tabla géneros

Tabla con los géneros a los que pertenece los distintos videojuegos

| Campo | Tipo | Nulo | Clave primaria | Único | Predeterminado |
|--------------|--------------|------|----------------|-------|----------------|
| IdGenero | Integer | No | Si | Si | Auto_increment |
| NombreGenero | Varchar (50) | No | No | No | |

Los campos de la tabla contienen la siguiente información:

- IdGenero: Id del genero
- NombreGenero: Nombre del genero

Tabla Videojuegos

Tabla con todos los datos de los videojuegos

| Campo | Tipo | Nulo | Clave primaria | Único | Predeterminado |
|---------------|---------------|------|----------------|-------|----------------|
| IdVideojuego | Integer | No | Si | Si | Auto_increment |
| Título | Varchar (200) | No | No | No | |
| Descripción | Text | No | No | No | |
| Distribuidora | Varchar (200) | No | No | No | |
| Precio | Double (9,2) | Si | No | No | |
| Imagen | Varchar (250) | Si | No | No | |
| IdGenero | Integer | No | No | No | |

Los campos de la tabla contienen la siguiente información:

- IdVideojuego: Id del videojuego
- Título: Nombre del videojuego
- Descripción: Descripción sobre el videojuego
- Distribuidora: Empresa que distribuye el videojuego o creadora
- Precio: Dinero que cuesta el videojuego
- Imagen: Imagen con el que se representara el videojuego
- IdGenero: Id con el género del videojuego

En la tabla encontramos una clave foránea en IdGenero la cual hace referencia a la tabla género.

Tabla compras

Tabla que contendrá los datos de las compras

| Campo | Tipo | Nulo | Clave primaria | Único | Predeterminado |
|--------------|--------------|------|----------------|-------|----------------|
| IdCompra | Integer | No | Si | Si | Auto_increment |
| Usuario | Varchar (50) | No | No | No | |
| IdVideojuego | Integer | No | No | No | |
| Asunto | Varchar (50) | No | No | No | |

Los campos de la tabla contienen la siguiente información:

- IdCompra: Id de la compra
- Usuario: Usuario que compra el videojuego
- IdVideojuego: Id del Videojuego
- Asunto: Asunto donde pondrá que se ha hecho la compra del videojuego

En la tabla podemos encontrar dos claves foráneas que pertenecen a Usuario y a IdVideojuego, estas hacen referencia a las tablas de los videojuegos y de los usuarios

Tabla ventas

Tabla que contendrá los datos de las ventas

| Campo | Tipo | Nulo | Clave primaria | Único | Predeterminado |
|--------------|--------------|------|----------------|-------|----------------|
| IdVenta | Integer | No | Si | Si | Auto_increment |
| Usuario | Varchar (50) | No | No | No | |
| IdVideojuego | Integer | No | No | No | |
| Asunto | Varchar (50) | No | No | No | |

Los campos de la tabla contienen la siguiente información:

- IdVenta: Id de la venta
- Usuario: Usuario que vende el videojuego
- IdVideojuego: Id del Videojuego
- Asunto: Asunto donde pondrá que se ha hecho la venta del videojuego

En la tabla podemos encontrar dos claves foráneas que pertenecen a Usuario y a IdVideojuego, estas hacen referencia a las tablas de los videojuegos y de los usuarios

Casos de uso

Una vez que tenemos creada la base de datos tenemos que ver cómo será la interacción entre el usuario o cliente con la aplicación y para ello se verán los casos de uso.

El diagrama de casos de usos se compone de dos partes principalmente, los actores y el caso de uso.

- El caso de uso es una acción o tarea que realiza el actor o el sistema.
- Los actores son las personas o entidades que participan en estas acciones.

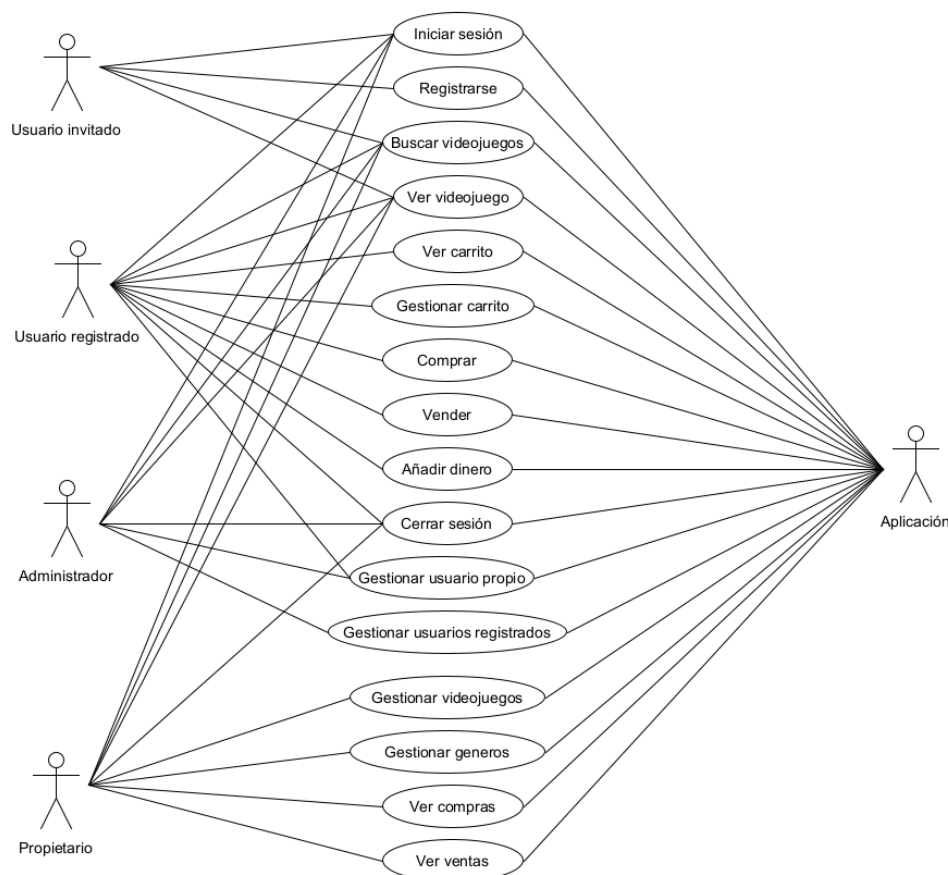


Ilustración 11 Diagrama de casos de uso

El diagrama de casos de uso se utilizará para mostrar el funcionamiento de un sistema ante su interacción con otros actores o sistemas.

En el diagrama se pueden observar 5 actores: El usuario invitado, el usuario registrado, el propietario, el administrador y el propio sistema.

El usuario que no está registrado puede registrarse en la aplicación, iniciar sesión en ella y buscar videojuegos.

El usuario invitado puede iniciar sesión en la aplicación, buscar videojuegos, añadir dinero, modificar los datos de su usuario, comprar y vender videojuegos

El administrador puede iniciar sesión en la aplicación, cerrar la sesión, y administrar los usuarios registrados.

El propietario puede iniciar sesión en la aplicación, cerrar sesión en la aplicación y administrar la creación y modificación de los videojuegos y géneros, y la visualización de las compras y ventas hechas en la aplicación.

La aplicación interactúa con todas estas tareas, al ser ella misma la que esta compuesta de estas tareas.

Se va a explicar los casos de uso de todos los actores:

El usuario invitado

1. Acceder a la aplicación
2. Una vez accede a la aplicación tendrá tres opciones: Como primera opción podrá registrarse en la que aparecerá un formulario donde tendrá que rellenar unos datos. Si no existe ese usuario, el sistema lo creará. Como segunda opción podrá hacer la búsqueda de videojuegos, que dependiendo del tipo de búsqueda escogerá una opción u otra y finalmente el sistema le mostrará el resultado. Como tercera opción tenemos la de iniciar sesión, en la que se introducirá el usuario y contraseña, y el sistema comprobará si existe. Si existe, el usuario habrá iniciado sesión.

El usuario registrado

1. Acceder a la aplicación
2. Una vez accede a la aplicación e inicia sesión, verá un menú el cual tendrá varias opciones. El usuario podrá ver y modificar sus datos al registrarse por si se ha equivocado o quiere cambiar algo. Podrá hacer la misma búsqueda de videojuegos que los usuarios invitados y una vez este mirando el producto podrá añadirlo al carrito o vender un producto de ese mismo tipo. Para poder hacer la compra tiene la opción de añadir dinero, en la que le aparecerá un formulario para los datos de su tarjeta y la cantidad que desea meter. Una vez se tenga el dinero deseado, se puede ir a ver el carrito de la compra y realizar el pedido.
3. Cuando el usuario haya acabado todas estas tareas podrá cerrar sesión y lo devolverá a la página de inicio.

El propietario

1. Acceder a la aplicación
2. Una vez accede a la aplicación e inicia sesión, el propietario podrá modificar sus datos como un usuario normal, añadir videojuegos y poder modificar sus datos en caso de que haga falta. Podrá hacer lo mismo con los géneros de los videojuegos. Por último, podrá visualizar todas las ventas y compras hechas en la aplicación.
3. Por último, el propietario podrá cerrar sesión y lo devolverá a la página de inicio.

El administrador

1. Accede a la aplicación
2. Una vez accede a la aplicación e inicia sesión, el administrador podrá ver quién está registrado en la página, y si hace falta modificarlos y eliminarlos.
3. Por último, el administrador podrá cerrar sesión y lo devolverá a la página de inicio.

Aplicación

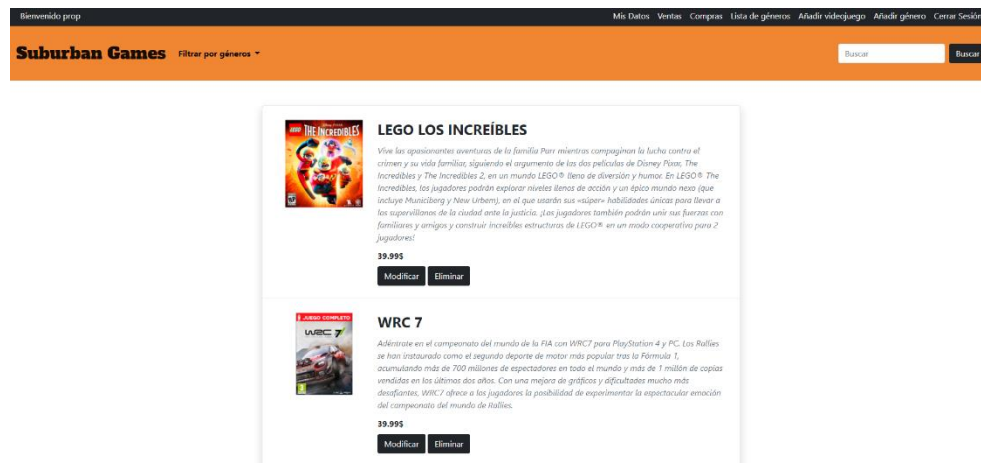


Ilustración 13 Resultado final de la aplicación

Para la aplicación se ha creado un proyecto Spring Boot para que se haga más fácil el diseño de los controladores. Hay que comentar que al ser un proyecto Maven se han añadido unas dependencias aparte para el uso del jstl.

Lo primero que se ha hecho ha sido configurar el proyecto para que se pueda conectar a la base de datos y para la seguridad de los usuarios. Esto último se ha configurado en un fichero Java dentro del paquete “com.ite.videojuegos.configuracion”.

Para organizar el resto de este proyecto se ha aplicado el modelo vista-controlador (MVC). Este modelo se utiliza sobre aplicaciones y lo divide en tres partes: el modelo, la vista y el controlador.

El modelo es la parte donde se trabaja con los datos, contiene los métodos necesarios para acceder a esos datos y modificarlos.

Las vistas contienen el código que hace que se pueda crear la interfaz de usuario, en este caso se ha utilizado JSP en vez de HTML.

El controlador es el enlace entre el modelo y las vistas, contiene el código necesario para que se pueda realizar una compra, registrarse, etc.

Los controladores

Estos controladores se encuentran dentro del paquete “com.ite.videojuegos.controlador”. Se divide en 4 controladores principales, uno para cada perfil (cliente, propietario, Administrador) y el restante para las acciones generales.

En el Controlador de inicio podemos encontrarnos los métodos principales de la aplicación.

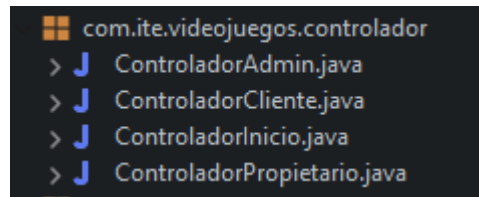


Ilustración 14 Paquete con los controladores

Contiene los métodos para las siguientes acciones:

- Inicializar algunos elementos como lo son el carrito o los videojuegos
- Iniciar sesión y registrarse con un usuario y contraseña
- Ver los videojuegos en detalle
- La búsqueda de videojuegos mediante la búsqueda por géneros o la búsqueda por el título

El controlador del cliente se encarga de todas las acciones que necesita el usuario con perfil de cliente.

Contiene los métodos para las siguientes acciones:

- Ver y modificar los datos del usuario que ha iniciado sesión
- Poder añadir dinero para que el usuario pueda comprar un videojuego
- Acceder al carrito y la capacidad de añadir videojuegos al carrito y eliminarlos
- Comprar y vender videojuegos

El controlador del propietario se encargará de las acciones que hacen los usuarios con perfil de propietario

Contiene los métodos para las siguientes acciones:

- Añadir, modificar y eliminar los videojuegos
- Añadir, modificar y eliminar los géneros
- Ver una lista de géneros
- Ver una lista con las compras realizadas
- Ver una lista con las ventas realizadas

El controlador del administrador se encarga de las acciones que hacen los usuarios con perfil de administrador.

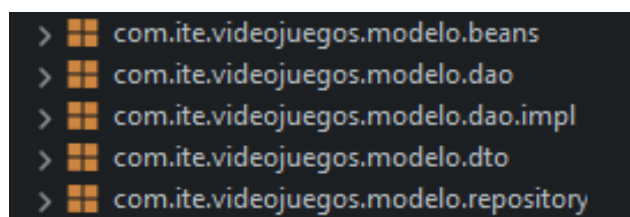
Contiene los métodos para las siguientes acciones:

- Ver una lista con todos los usuarios registrados en la aplicación
- Modificar los perfiles de los usuarios
- Eliminar usuarios

Los modelos

Los modelos se encuentran dentro del paquete “com.ite.videojuegos.modelo”. Este se divide en 5 partes.

- En el paquete modelo.beans se encuentran los modelos de datos de los videojuegos, usuarios, compras, etc.
- En el paquete modelo.repository se encuentran las interfaces que hacen que conecten los métodos con la base de datos. En el archivo VideojuegoRepository se encuentran secuencias JPQL que sirven para acceder de una manera especial a la base de datos, en este caso personalizan una consulta a la base de datos.
- En el paquete modelo.dao vamos a encontrar las interfaces de los modelos de datos que contienen los nombres de los métodos.
- En el paquete modelo.dao.impl se va a encontrar la implementación de las interfaces en el anterior paquete. En los métodos acceden a la base de datos a través de los repositorios creados en el otro paquete, gracias a esto podemos guardar, modificar, eliminar o visualizar los datos que hay guardados
- Finalmente, en el paquete modelo.dto se encuentra una clase que sirve para subir imágenes y guardarla en una dirección.



```
> com.ite.videojuegos.modelo.beans
> com.ite.videojuegos.modelo.dao
> com.ite.videojuegos.modelo.dao.impl
> com.ite.videojuegos.modelo.dto
> com.ite.videojuegos.modelo.repository
```

Ilustración 15 Paquetes de los modelos

Las vistas

Las vistas se encuentran en las carpetas “vistas” dentro de la carpeta “WEB-INF”. Todas las vistas están creadas en JSP (Java Server Pages).

Estas vistas son:

- añadirDinero.jsp: es una vista del perfil del cliente donde se muestra un formulario que hay que rellenar para introducir dinero en la aplicación.
- añadirGenero.jsp: es una vista del perfil del propietario que muestra un formulario para registrar un género.
- añadirVideojuego.jsp: es una vista del perfil del propietario que muestra un formulario para registrar un videojuego.
- cabecera.jsp: es una de las vistas principales del programa donde viene el menú de opciones y la barra de navegación, esta incluida en todas las vistas.
- confirmarVenta.jsp: es una vista del perfil del cliente donde se muestra una confirmación para que pueda vender el videojuego.
- detalleVideojuego.jsp: se muestran los datos de los videojuegos y a su vez las opciones de añadirlo al carrito o venderlo.
- footer.jsp: es el footer que aparece en todas las vistas.
- inicio.jsp: es la vista principal de la aplicación, se muestra la lista con los videojuegos que se están vendiendo.
- listaCompras.jsp.: es una vista del perfil del propietario, en ella puede mirar una lista de las compras realizadas
- listaVentas.jsp: es una vista del perfil del propietario, en ella puede mirar una lista de las ventas realizadas
- listaGeneros.jsp: es una vista del perfil del propietario, en ella se puede ver los géneros con la posibilidad de eliminarlo o modificarlos.
- login.jsp: es una vista donde aparece un formulario para iniciar sesión.
- misdatos.jsp: es una vista donde aparece los datos del usuario con el que se ha iniciado sesión con posibilidad a modificarlos.
- modificarDatos.jsp: vista que permite modificar los datos del usuario con el que se ha iniciado sesión.

- modificarGenero.jsp es una vista del perfil del propietario, que permite la modificación de los géneros.
- modificarUsuario.jsp: es una vista del perfil del administrador, permite la modificación del perfil de cualquier usuario.
- modificarVideojuego.jsp: es una vista del perfil del propietario, que permite la modificación de cualquier dato de un videojuego.
- registro.jsp: es la vista que permite acceder a un formulario para registrarse en la aplicación.
- usuarios.jsp: es una vista del perfil del administrador que permite ver una lista con todos los usuarios registrados.
- verCarro.jsp: es una vista del perfil del cliente, donde se puede ver todos los elementos que se han añadido al carrito con posibilidad de realizar una compra.

```

<%@ page language="java" contentType="text/html; charset=UTF-8"
    pageEncoding="UTF-8"%>
<%@ taglib uri="http://java.sun.com/jsp/jstl/core" prefix="c"%>
<%@ taglib prefix="sec"
    uri="http://www.springframework.org/security/tags"%>

<!DOCTYPE html>
<html>
<head>
<meta charset="UTF-8">
<title>Iniciar sesión</title>
</head>
<body class="d-flex flex-column min-vh-100">
<jsp:include page="cabecera.jsp"></jsp:include>
<div class="container rounded bg-white mt-5 mb-5">
<div class="row justify-content-center">
<div class="col-md-5 border-right">
<div class="p-3 py-5">
<form action="/Login" method="post">
<div>
<div class="d-flex justify-content-between align-items-center mb-3">
<p class="text-right fs-1 fw-bold fst-italic"
style="color: #f28d52;">Iniciar sesión</p>
</div>
<div class="row mt-3">
<div class="col-md-12 input-group-lg">
<label class="labels fs-5">Usuario</label><input type="text"
class="form-control" name="username">
</div>
<div class="col-md-12 input-group-lg">
<label class="labels fs-5">Contraseña</label><input
type="password" class="form-control input-lg" name="password">
</div>
<div class="mt-5 text-center">
<input class="btn btn-dark btn-lg" type="submit"
value="Iniciar sesión">
</div>
</div>
</div>
</form>
</div>
</div>
</div>
<jsp:include page="footer.jsp"></jsp:include>
</body>
</html>

```

Ilustración 16 Login.jsp

El despliegue

Finalmente, para el despliegue de la aplicación se ha utilizado XAMPP. Gracias al Tomcat que trae por defecto, se ha creado un servidor. Por otra parte, para que el proyecto pudiera funcionar en Tomcat, se ha tenido que modificar un par de opciones en las propiedades del proyecto y pasarlo de formato .jar a formato .war.

Conclusiones.

El desarrollo de una aplicación web parece difícil, pero una vez que estás metido y trabajando dentro, es más fácil de lo que parece.

Hay que destacar que hay planificar bien la estructura y prestar atención tanto a la programación como al diseño de interfaz, ya que un pequeño fallo puede desatar muchos problemas y errores que son difíciles de solucionar.

Gracias a este trabajo he podido aumentar mis conocimientos sobre el desarrollo web, lo que ha hecho que me guste ese mundo mucho más de lo que lo hacía ya. Ha sido muy satisfactorio, entretenido e interesante el estudio de tecnologías que no había tocado nunca, como Bootstrap.

Los objetivos planteados inicialmente se han cumplido todos, e incluso en un futuro se puede intentar profundizar más en la aplicación.

Estas mejoras podrían ser de diferentes tipos:

- Mejorar el formulario de venta
- Soporte con el administrador
- Dotar de más funciones al administrador
- Añadir la opción de notificaciones
- Crear un filtro mas extenso para la búsqueda de videojuegos
- Mejorar la interfaz de usuario ya existente

Bibliografía

Documentación de Bootstrap:

<https://getbootstrap.com/docs/5.0/getting-started/introduction/>

Documentación de la asignatura de Desarrollo Web Entorno Servidor

Documentación de la asignatura de Diseño de Interfaces Web

Documentación de la asignatura de Despliegue de aplicaciones

Documentación de la asignatura de Desarrollo Web Entorno Cliente

Documentación de la asignatura de Lenguaje de Marcas

Documentación de la asignatura de Base de datos