

# VIEWDEOGAME

PABLO FERNÁNDEZ FERNÁNDEZ 2DAW

## CONTENIDO

1. INTRODUCCIÓN .....	4
1.1. CONTEXTO.....	4
1.2. PRESENTACIÓN .....	4
1.3. OBJETIVOS.....	5
2. ANÁLISIS Y DISEÑO DE LA APLICACIÓN .....	5
2.1. ANÁLISIS DE REQUISITOS FUNCIONALES .....	5
2.1.1. INTRODUCCIÓN.....	5
2.1.2. TABLA DE REQUISITOS FUNCIONALES .....	5
2.2. DIAGRAMA DE CASOS DE USO .....	6
2.2.1. INTRODUCCIÓN .....	6
2.2.2. DIAGRAMAS Y DESCRIPCIÓN .....	7
2.3. DISEÑO DE INTERFAZ .....	9
2.3.1. INTRODUCCIÓN .....	9
2.3.2. DIAGRAMAS Y DESCRIPCIÓN .....	10
2.4. ESTRUCTURAS DE DATOS USADAS.....	23
2.4.1 INTRODUCCIÓN .....	23
2.4.2 DIAGRAMAS Y DESCRIPCIÓN .....	23
2.5. FASE DE PRUEBAS .....	25
2.5.1. PRUEBAS DE VALIDACIÓN DE CÓDIGO .....	25
2.5.2. PRUEBAS DE IMPLANTACIÓN EN DISTINTOS SOPORTES .....	27
3. IMPLEMENTACIÓN .....	28
3.1. TECNOLOGÍAS USADAS DURANTE EL DESARROLLO DEL PROYECTO .....	28
4. PUESTA EN MARCHA. DESPLIEGUE .....	34
4.1. GUIA DE DESPLIEGUE DEL PROYECTO .....	34
5. APLICACIÓN EN EL ÁMBITO EMPRESARIAL .....	40
5.1. IDEA DE NEGOCIO .....	40
5.2. PLANIFICACIÓN DE NECESIDADES DE RRHH .....	40
5.3. PLANIFICACIÓN DE NECESIDADES DE PRODUCCIÓN .....	40
5.4. PLANIFICACIÓN DE LA PREVENCIÓN DE RIESGOS LABORALES.....	40
5.5. PREVISIONES DE INVERSIÓN Y FINANCIACIÓN.....	41
5.6. ESTUDIO DE MERCADO Y MARKETING OPERATIVO.....	41
6. CONCLUSIÓN .....	41
4.1. VALORACIÓN DEL TRABAJO REALIZADO .....	41
4.2. POSIBLES AMPLIACIONES.....	42
7. BIBLIOGRAFÍA Y WEBGRAFÍA.....	42

8.ANEXO .....	43
1.MANUAL DE USUARIO .....	43
2.CÓDIGO COMPLETO DEL PROYECTO.....	59



## 1. INTRODUCCIÓN

### 1.1. CONTEXTO

Este documento detalla el proyecto de Pablo Fernández Fernández de 2do de Desarrollo de Aplicaciones Web: La página web ViewdeoGame.

### 1.2. PRESENTACIÓN

La página está dirigida al público perteneciente al sector de los videojuegos y está enfocada en un **sistema de valoración a través de capturas de pantalla** subidas por los propios usuarios, el objetivo es permitir que las personas puedan **compartir sus capturas favoritas de los videojuegos a los que hayan jugado**, además de permitir que otras personas puedan descubrir nuevos videojuegos a través de estas.

El archivo permite a gente subir videojuegos con una **portada, géneros, y las plataformas en las que está disponible** (esto permite actualizar rápidamente juegos que cambien su portada al salir en nuevas plataformas o que cambien de nombre al ser localizados en otro país). Una vez hayan sido subidos, **los usuarios pueden editar los géneros del juego, sus plataformas, el nombre y la carátula de este**. **El administrador podrá eliminarlos de la página** (permitiendo eliminar videojuegos que no existen, pero evitando que un usuario lo elimine por error), junto con todas sus capturas.

En la página principal aparecerá la **lista de videojuegos disponibles**, desde donde los usuarios pueden **buscar el videojuego por nombre, u ordenarlas según el número de capturas que tengan**. Una vez dentro de la página de cualquiera de los videojuegos, **el usuario puede subir sus propias capturas del videojuego**, y los otros usuarios las valorarán, además de poder **añadirla a su lista de capturas favoritas**. Las capturas subidas por cada usuario se verán en su página, al igual que sus favoritas, además de en la página del propio juego.

La valoración general de las capturas se realiza a través de un sistema de media entre la valoración de cada individual de cada usuario y el número de personas que la han valorado. Cada captura tendrá como datos su **nombre el usuario que la subió, su valoración, el número de gente que la tiene como favorita, el número de personas que la han valorado y la plataforma desde la que se subió** (permitiendo a las personas ver cómo cambia la calidad visual del juego dependiendo de su plataforma).

**Cada usuario podrá editar su correo electrónico y su nombre**, además de poder **eliminar el perfil junto con todas sus capturas y valoraciones**. También podrá **eliminar sus capturas individualmente**.

**El administrador puede eliminar cualquier captura, usuario y videojuego.**

### 1.3. OBJETIVOS

Relación de objetivos que persigue el proyecto:

- El usuario logueado puede **subir videojuegos** que le gusten
- **Los usuarios anónimos solo podrán ver la página principal, los detalles de los juegos, o el perfil de otros usuarios.** No podrán editar ni subir ni eliminar juegos ni capturas.
- **El usuario logueado puede ver y editar las páginas de videojuegos ya existentes, independientemente de si lo ha subido él.**
- **Los usuarios logueados podrán subir y eliminar sus propias capturas** además de **valorar las ajenas** y añadirlas a su lista de capturas favoritas.
- Los usuarios podrán **ver su perfil u el de otros usuarios**, donde se encontrarán su lista de capturas favoritas además de sus capturas subidas.
- **El administrador puede ver y eliminar cualquier cuenta de usuario, captura o videojuego.**

## 2. ANÁLISIS Y DISEÑO DE LA APLICACIÓN

### 2.1. ANÁLISIS DE REQUISITOS FUNCIONALES

#### 2.1.1. INTRODUCCIÓN

En esta sección se detallan en una tabla los requisitos funcionales que debe tener esta página, además de una pequeña descripción de estos.

#### 2.1.2. TABLA DE REQUISITOS FUNCIONALES

Requisito	Descripción
Registrarse	Cada visitante tendrá la opción de crear un usuario en la página
Login	Los visitantes deberían ser capaces de loguearse en la página utilizando el usuario respectivo
Ver página de videojuego	Los usuarios deberían ser capaces de ver la página correspondiente al videojuego que ellos elijan
Ver datos de captura	Se deberían poder ver los datos de una captura al hacer click en ella
Valorar captura	Se debería poder valorar la captura mediante las estrellas que aparecen en pantalla (1 estrella entera equivale a 1 punto, media equivale a 0.5)
Marcar captura como favorita	El usuario debería de ser capaz de añadir la captura correspondiente a favoritos

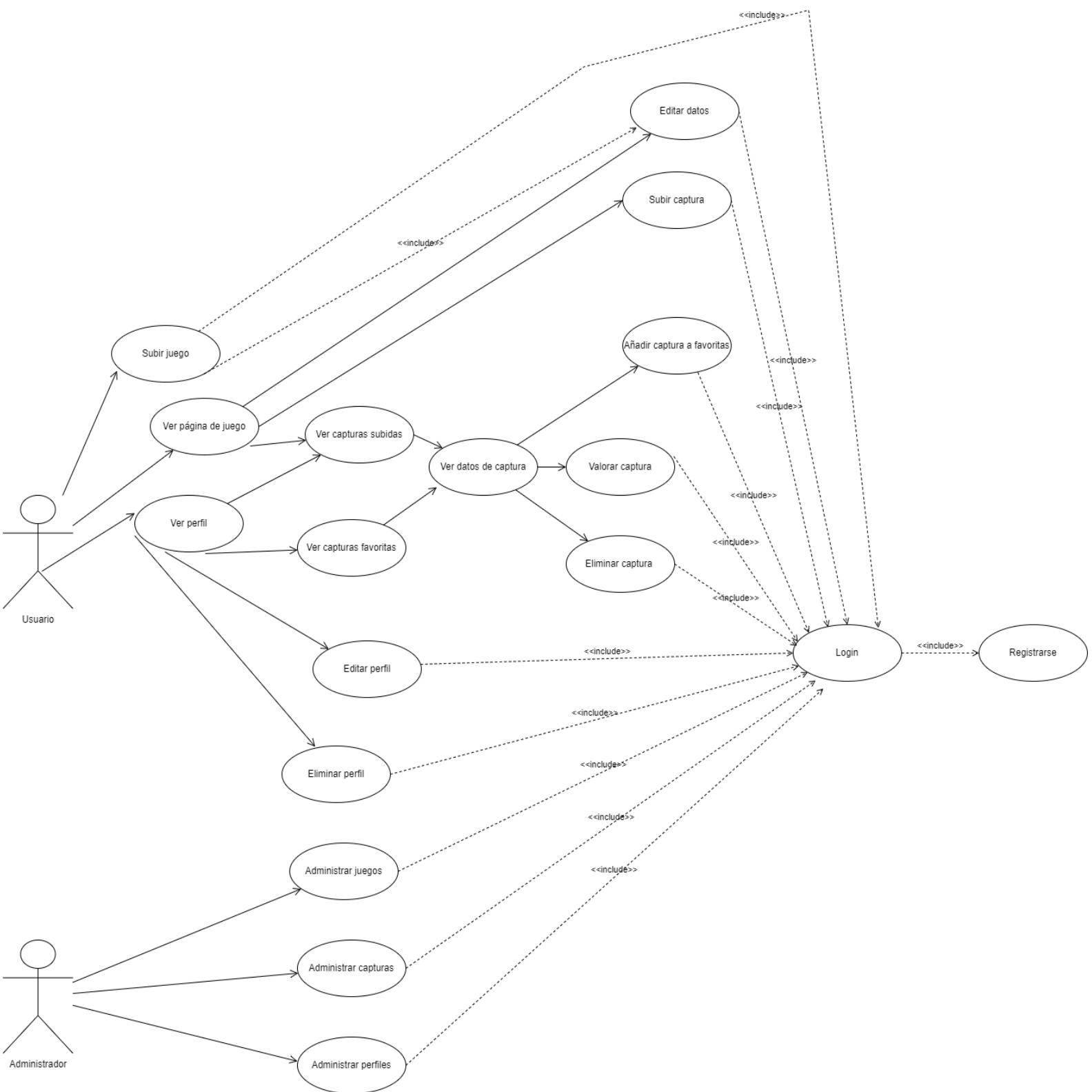
<b>Subir captura</b>	Si el usuario está logueado, debería ser capaz de subir una captura de pantalla a la página
<b>Eliminar captura</b>	El usuario debería poder eliminar las capturas subidas. El administrador debería de ser capaz de eliminar cualquier captura
<b>Buscar Perfil</b>	Se deberían poder buscar los perfiles de las diferentes personas registradas en la página, independientemente de si el usuario que está realizando la búsqueda lo está.
<b>Ver Perfil</b>	Se debería poder ver el perfil que ellos elijan, ya sea el suyo propio o el de otra persona
<b>Editar Perfil</b>	El usuario debería poder editar los datos de su propio perfil.
<b>Borrar Perfil</b>	El usuario debería poder borrar su propio perfil. El administrador debería de ser capaz de borrar todos los perfiles
<b>Subir juego</b>	Los usuarios logueados deberían de poder subir un videojuego a la página, con un nombre, géneros, plataformas y carátula.
<b>Editar juego</b>	Los usuarios logueados deberían poder editar los datos del videojuego
<b>Eliminar juego</b>	El administrador debería poder eliminar cualquier juego.

## 2.2. DIAGRAMA DE CASOS DE USO

### 2.2.1. INTRODUCCIÓN

En esta sección se mostrará el diagrama de casos de uso de la aplicación, con una explicación de cada uno de estos.

## 2.2.2. DIAGRAMAS Y DESCRIPCIÓN



## **Casos compartidos:**

- **Login:** Entrar en tu cuenta de la página, esto es necesario para: Subir y editar los datos del videojuego, añadir capturas a la lista de favoritos, subirlas, valorarlas y eliminarlas, y editar o eliminar tu perfil. En caso de que haga Login como administrador también podrá Administrar los juegos, capturas y perfiles.
- **Registrarse:** Crear una cuenta en la página web. Es necesario para hacer Login.

## **Usuario:**

- **Subir juego:** El usuario podrá subir un videojuego a la página web.
- **Editar datos:** Se podrá cambiar el nombre, caratula, plataformas y géneros del juego. Cuando un juego se sube por primera vez, es obligatorio introducir estos datos.
- **Ver página de juego:** El usuario podrá ver la página del juego que elija. En esta página se mostrará la caratula del juego, su nombre, los géneros a los que pertenece, las plataformas en las que está disponible y las capturas que se hayan subido.
- **Ver perfil:** Se podrá ver el perfil de cualquier usuario registrado en la web, ya sea el del propio usuario o el de otra persona.
- **Ver capturas subidas:** El usuario podrá ver las capturas subidas a la página web, si lo hace desde la página de un juego, verá las capturas pertenecientes a ese juego, si lo hace desde un perfil, verá las capturas subidas por ese usuario.
- **Ver capturas favoritas:** El usuario podrá ver todas las capturas que él u otro usuario hayan marcado como favoritas.
- **Subir captura:** Se podrá subir una captura a la página del juego correspondiente. La captura deberá tener la imagen, un nombre y una plataforma asignada.
- **Ver datos de captura:** El usuario podrá ver los datos de la captura que elija, siendo estos: su nombre, el usuario que la ha subido, la plataforma desde donde lo ha hecho, la cantidad de gente que la ha marcado como favorita, la cantidad de gente que la ha valorado, y su valoración media sobre cinco.
- **Añadir captura a favoritas:** El usuario podrá añadir la captura que elija a sus favoritas, siempre y cuando no la haya subido él.

- **Valorar captura:** El usuario le podrá dar una valoración sobre cinco a la captura que elija.
- **Eliminar captura:** El usuario podrá eliminar la captura que elija, siempre y cuando la haya subido él.
- **Editar perfil:** El usuario podrá editar el nombre y el correo electrónico de su propio perfil.
- **Eliminar perfil:** El usuario podrá eliminar su propio perfil de la página web, al hacerlo, también se eliminarán cualquier captura que haya subido y valoración que haya dado.

#### **Administrador:**

- **Administrar juegos:** El administrador podrá eliminar cualquier juego de la página web, junto con todas sus capturas.
- **Administrar capturas:** El administrador podrá eliminar cualquier captura de la página web., independientemente de quien la haya subido.
- **Administrar perfiles:** El administrador podrá eliminar cualquier perfil de la página web, independientemente de quien sea.

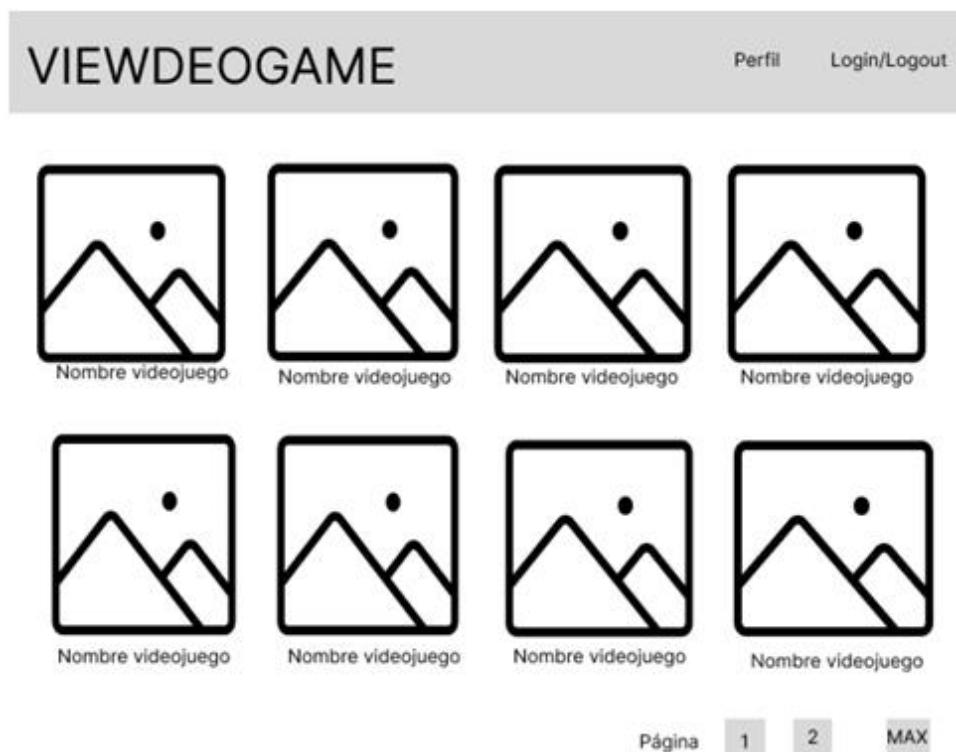
## 2.3. DISEÑO DE INTERFAZ

### 2.3.1. INTRODUCCIÓN

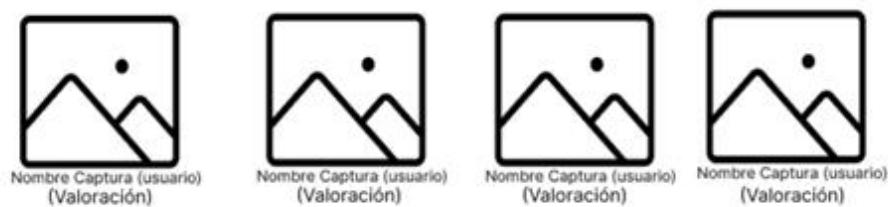
En esta sección se mostrará la interfaz general de la aplicación, además de una descripción de cada una y una explicación de los cambios que ha tenido en relación con la propuesta.

### 2.3.2. DIAGRAMAS Y DESCRIPCIÓN

#### PÁGINA DE INICIO



#### CAPTURAS DE LA SEMANA



Pablo Fernández Fernández . 2024

PÁGINA PRINCIPAL

Diseño Propuesto

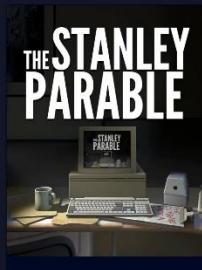
BUSCAR JUEGOS

BUSCAR PERFILES

## Disfruta de las vistas

Ya se han subido 18 capturas

Buscar...

Orden  
DESCENDENTECapturas:  
4Capturas:  
4Capturas:  
4Capturas:  
4

&lt; 1 2 &gt;

Pablo Fernández Fernández. 2024

La página ha recibido varios cambios: **las capturas más valoradas de la semana se han cambiado por un banner con una frase y el número de capturas totales que tiene la página**, pensé que este era un cambio pegaba más con la estética general de la página, ya que el banner utiliza una captura aleatoria de los juegos subidos, lo cuál va a ser algo recurrente en el resto de la página. **Se ha incluido una barra de búsqueda por nombre y un seleccionador para ordenar los juegos por orden ascendente o descendente según el número de capturas.**

**Se han cambiado las cartas de cada uno de los juegos para mostrar el número de capturas que tienen en lugar de su nombre**, este cambio se debe a que los videojuegos siempre incluyen su nombre en la portada del mismo, haciendo que incluir el nombre en la carta fuera redundante.

Se ha añadido al header una barra con las **opciones de buscar juegos** (la propia página de inicio) **y de buscar perfiles**. Una vez iniciemos sesión (como se verá en la sección de página de juego), se añadirá una opción para abrir el formulario de subir juego.

## PÁGINAS DE LOGIN Y REGISTRO



Email

Contraseña

**INICIAR SESIÓN**

¿Aún no tienes cuenta? [Regístrate](#)

[Volver a la página principal](#)

Pablo Fernández Fernández, 2024



Nombre de usuario

Email

Contraseña

Confirmar Contraseña

**REGISTRARSE**

¿Ya tienes cuenta? [Inicia sesión](#)

[Volver a la página principal](#)

Pablo Fernández Fernández, 2024

Las páginas de Login y de registro son muy similares la una a la otra, consistiendo en un **fondo oscuro** y en el centro un **formulario con el fondo azul**, el cual contiene el **logo de la página, unos campos para llenar con los datos del usuario y, debajo de estos, un botón para loguearse o registrarse**. Finalmente se ven unos **enlaces que varían según en cuál de las dos páginas estemos**.

Los cambios principales entre estas dos páginas son:

- La página de registro incluye los campos de **Nombre de usuario, contraseña y repetir contraseña**
- El primer enlace de la página de inicio envía al usuario a la página de inicio.

#### PÁGINA DE JUEGO

**VIEWDEOGAME**

Perfil      Login/Logout



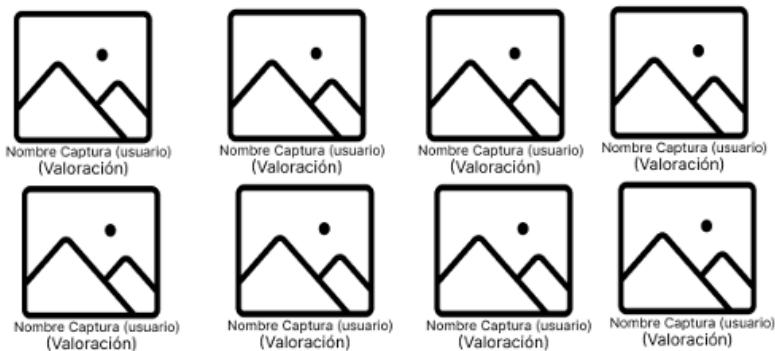
**NOMBRE VIDEOJUEGO**

Géneros  
Lorem ipsum dolor sit amet, consectetur adipiscing

Plataformas  
Lorem ipsum dolor sit amet, consectetur adipiscing

Descripción  
Lorem ipsum dolor sit amet, consectetur adipiscing elit. Phasellus mauris metus, dapibus in porta vel, fermentum id dui. Suspendisse potenti. Maecenas malesuada, libero sit amet mattis rhoncus, enim dolor malesuada dolor, non convallis nisi elit non augue.

#### CAPTURAS



Página 1 2 ... MAX

Pablo Fernández Fernández . 2024

PÁGINA DE VIDEOJUEGO

Diseño propuesto

 VIEWDEOGAME

PABLO CERRAR SESIÓN

SUBIR VIDEOJUEGO BÚSQUEDA JUEGOS BÚSQUEDA PERFILES



**Trails through daybreak**

**Generos:**  
JRPG  
Combate por turnos

**Plataformas:**  
Nintendo Switch  
PlayStation 5  
PC

SUBIR CAPTURA EDITAR DATOS ELIMINAR

**Capturas (4)**



© 2021 Nihon Falcom Corporation. All rights reserved.  
Pablo Fernández Fernández. 2024

La página de juego también ha pasado por algunos cambios:

El primero es que **la descripción del juego ahora ha sido sustituida por los botones de subir captura, editar juego** (los cuáles solo aparecerán si el usuario ha iniciado sesión) y **el botón de eliminar** (el cuál solo aparece si has iniciado sesión como administrador), este cambio se debe a que **consideré a la descripción como algo con una importancia menor en relación con el resto de las cosas, y necesitaba más espacio** para incluir los botones.

El segundo cambio es que **el nombre, valoración y el usuario de la captura ya no aparece debajo de estas**, este cambio, al igual del anterior, se debe a la **falta de espacio**, y consideré que sería una mejor idea **eliminar estas cosas e incluirlas todas en el modal que se verá en el siguiente punto**.

Se le ha añadido a la página una **imagen de fondo**, la cuál es una de las **capturas del juego elegida de manera aleatoria**.

---

#### DATOS DE CAPTURA

## Captura de prueba daybreak 4

De: pedro  
En: PC

Valoraciones:2

Favoritos:0



Nota Media:

**3.5/5**

Tu valoración:

Marcar como favorita

**ELIMINAR CAPTURA**

El modal que muestra los datos de captura consiste en el **nombre de la captura** como título, debajo de él se pueden ver el **usuario que la ha subido** y la **plataforma en la que lo ha hecho**.

Encima de la captura se pueden ver el **número de personas que la ha valorado** y el **número de personas que la ha marcado como favorita**.

Debajo de la captura se pueden ver la **nota media**, la **valoración con las estrellas que te permitirán darle una nota** (las estrellas aparecerán vacías por defecto o con la nota correspondiente si ya le has asignado una, y solo se verán si has iniciado sesión), el **check para marcarla como favorita** (el cuál solo se verá si has iniciado sesión y no has subido dicha captura) y el **botón para eliminar la captura** (el cuál solo se verá si has iniciado sesión y eres el que ha subido la captura, o si eres el administrador, como en el caso de la captura).

PERFIL DE USUARIO

VIEWDEOGAME

Perfil      Login/Logout



NOMBRE USUARIO

CAPTURAS MEJOR VALORADAS



Nombre Captura  
(Valoración)



Nombre Captura  
(Valoración)



Nombre Captura  
(Valoración)



Nombre Captura  
(Valoración)

CAPTURAS FAVORITAS



Nombre Captura (usuario)



Nombre Captura (usuario)



Nombre Captura (usuario)



Nombre Captura (usuario)

Página 1 2 ... MAX

PERFIL DE USUARIO

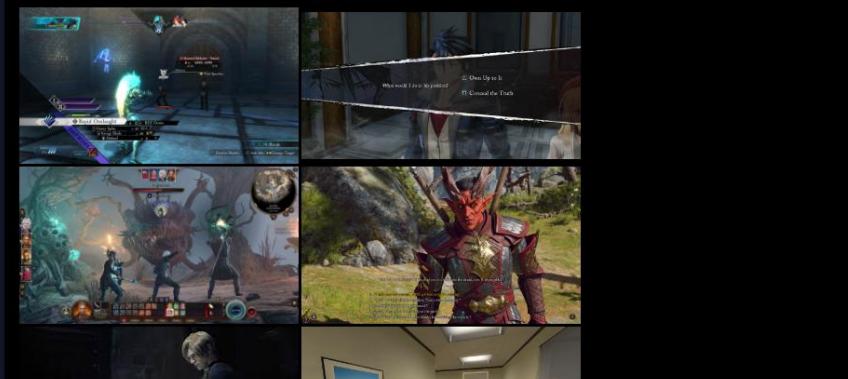
Diseño propuesto

[SUBIR VIDEOJUEGO](#)[BUSCAR JUEGOS](#)[BUSCAR PERFILES](#)

# PABLO

[ELIMINAR PERFIL](#)[EDITAR PERFIL](#)

## Capturas subidas (8)



## Capturas Favoritas (0)

Pablo Fernández Fernández. 2024

El perfil de usuario ha pasado por tres cambios:

El primero, **se ha eliminado la imagen del perfil**, debido a que no **se estaba utilizando en ningún otro sitio**, y el nombre ya era representación suficiente del perfil.

El segundo cambio es que se cambiaron las capturas destacadas por las que subió el usuario en general. Este cambio no ha tenido ningún motivo aparte de que consideré que **sería más adecuado que se enseñaran todas las capturas relacionadas con el perfil**.

El tercer y último cambio es que los expositores de capturas son más similares a los de la página de videojuego, esto se ha hecho para mantener un diseño consistente.

Debajo del nombre de usuario se pueden ver **dos botones añadidos**: El botón de **eliminar perfil** (el cuál solo aparecerá en caso de que seamos el administrador o el perfil sea el nuestro) y el botón de **editar perfil** (el cuál solo aparecerá si el perfil es el nuestro)

LISTA DE PERFILES

VIEWDEOGAME		Perfil	Login/Logout
Nombre			
Lorem		<a href="#">Eliminar</a>	
Página	1	2	MAX

Pablo Fernández Fernández . 2024

LISTA DE USUARIO (ADMINISTRADOR)

Diseño Propuesto

The screenshot shows a dark-themed web application interface. At the top, there's a header with the logo 'VIEWDEOGAME' on the left, a user profile icon with the name 'PABLO' and a 'CERRAR SESIÓN' (Logout) button on the right, and three navigation links: 'SUBIR VIDEOJUEGO', 'BUSCAR JUEGOS', and 'BUSCAR PERFILES'. Below the header is a search bar labeled 'Buscar Perfil' with a magnifying glass icon. Underneath the search bar is a table with three rows, each containing a user name: 'pablo', 'pedro', and 'sonia'. A small text '1-3 de 3' is visible at the bottom of the table. The footer contains the text 'Pablo Fernández Fernández, 2024'.

La lista de perfiles también ha tenido sus cambios:

En primer lugar, **los perfiles ahora se presentan en una tabla**. La razón de este cambio es que, **en la propuesta, la lista de usuarios estaba planeada como algo que solo el administrador podría ver**, pero al desarrollar el proyecto esto **cambió a que todo el mundo sea capaz de ver la lista de usuarios**, por lo que modifiqué el diseño a algo que considero que **encaja más con el resto de la página**.

El segundo es que he **movido el botón de eliminar de la lista de usuarios al perfil en sí**, esto es debido a que, **en un principio, solo el administrador iba a ser capaz de borrar perfiles**, pero al **cambiar esto a que los propietarios del perfil también pudieran eliminarlo**, decidí **cambiarlo de sitio para evitar que el propietario de perfil tenga que buscarlo** en lugar de simplemente entrar en él utilizando el botón del header. Además, esto hace que **el administrador se asegure de que usuario está borrando**.

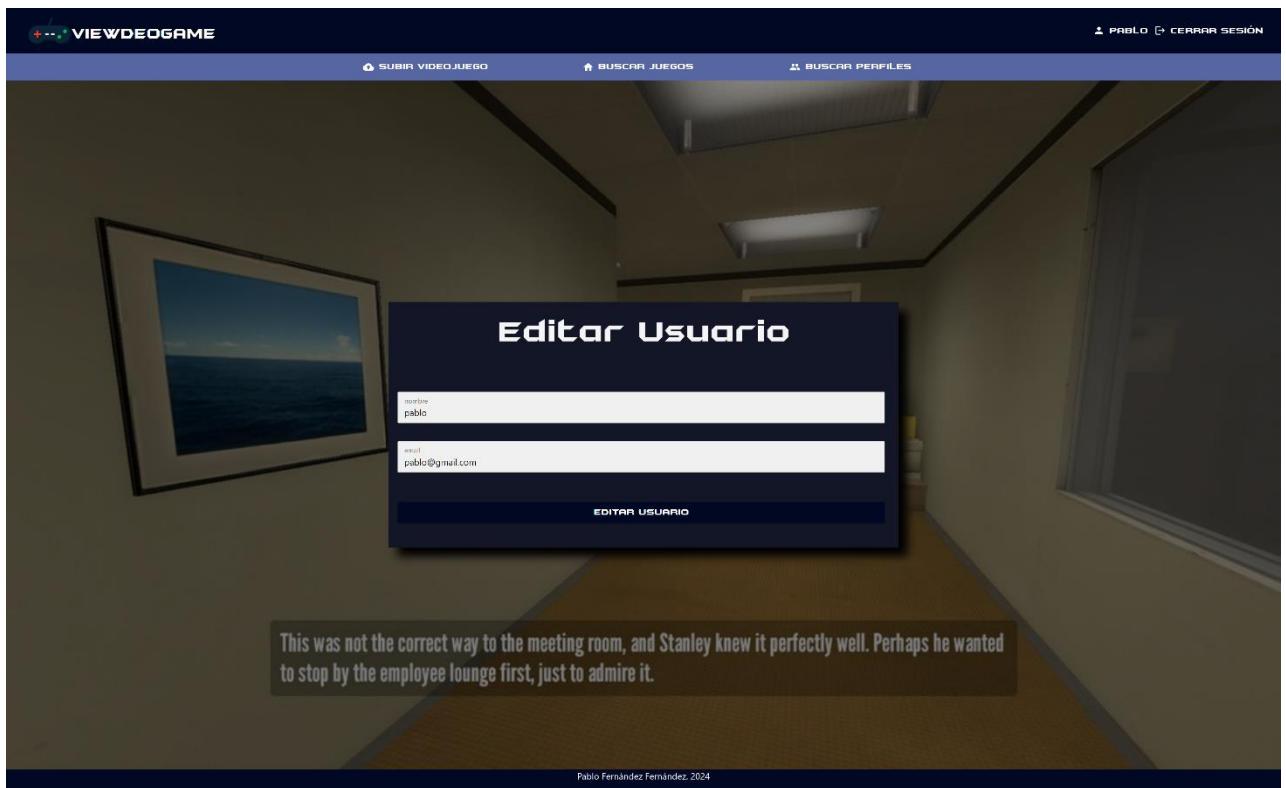
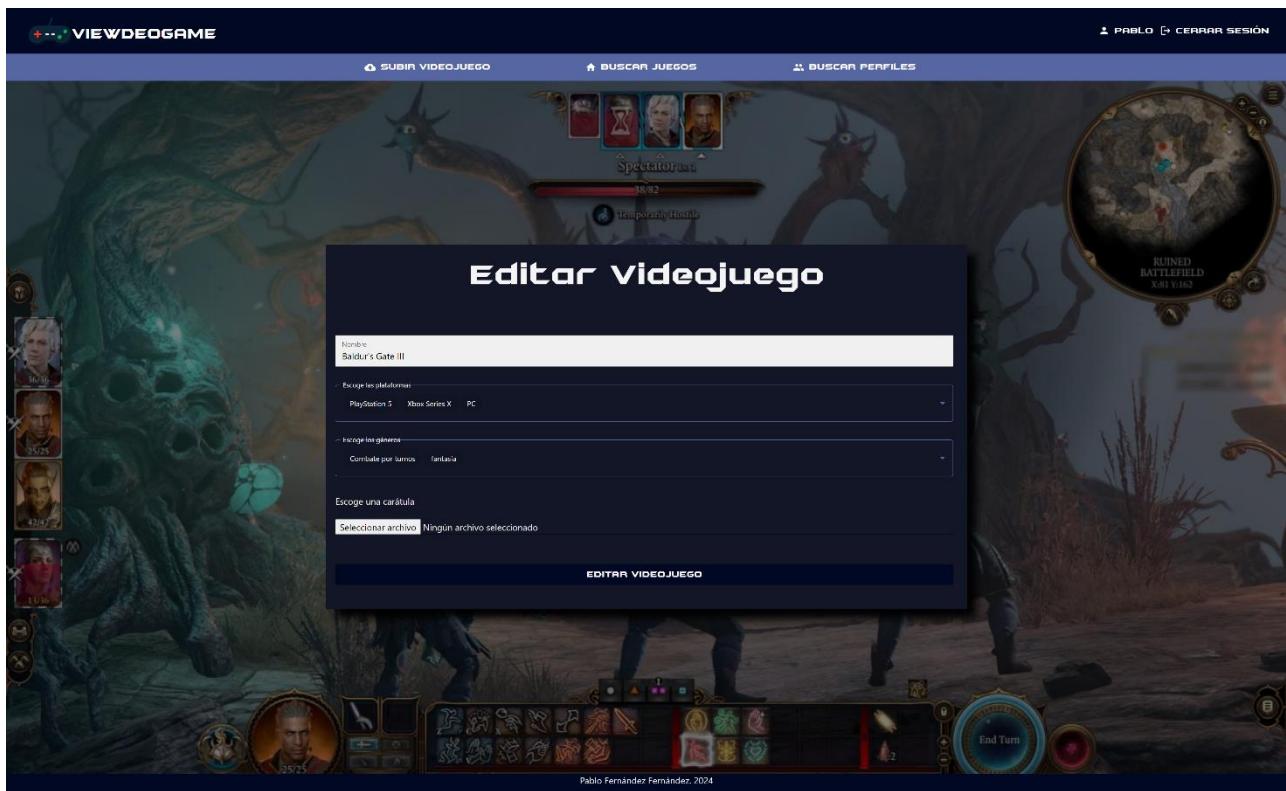
Se ha añadido una barra y un botón de búsqueda, esto es para que el usuario pueda encontrar más fácilmente el perfil que desea ver.

## FORMULARIOS

The image consists of two vertically stacked screenshots from a video game titled "VIEWDEOGAME".

**Top Screenshot:** A character with long white hair and a blue cap is shown from behind, holding a futuristic device. A dark overlay window titled "Subir Captura" (Upload Capture) is displayed. It contains fields for "Nombre" (Name), "Escoge una plataforma" (Choose a platform), and a file upload field showing "Seleccionar archivo Ningún archivo seleccionado" (Select file No file selected). A "SUBIR CAPTURA" button is at the bottom.

**Bottom Screenshot:** A character in a suit is standing in a hallway. A dark overlay window titled "Subir Videojuego" (Upload Game) is displayed. It contains fields for "Nombre" (Name), "Escoge las plataformas" (Choose platforms), "Escoge los géneros" (Choose genres), and a file upload field showing "Seleccionar archivo Ningún archivo seleccionado" (Select file No file selected). A "SUBIR VIDEOJUEGO" button is at the bottom. Below this window, a text box displays the message: "This was not the correct way to the meeting room, and Stanley knew it perfectly well. Perhaps he wanted to stop by the employee lounge first, just to admire it."



Todos los diseños de los formularios de la página siguen el mismo estilo, consistiendo en el formulario en sí en el centro de la pantalla, con un fondo azul oscuro y una sombra, y del fondo de toda la página en sí, el cuál es una captura aleatoria elegida del juego correspondiente en los formularios de Editar Videojuego y subir

**captura, del perfil en el caso del formulario de Editar Usuario, y de todos los juegos en el caso de Subir Videojuego.**

**El número de campos varía según el formulario:**

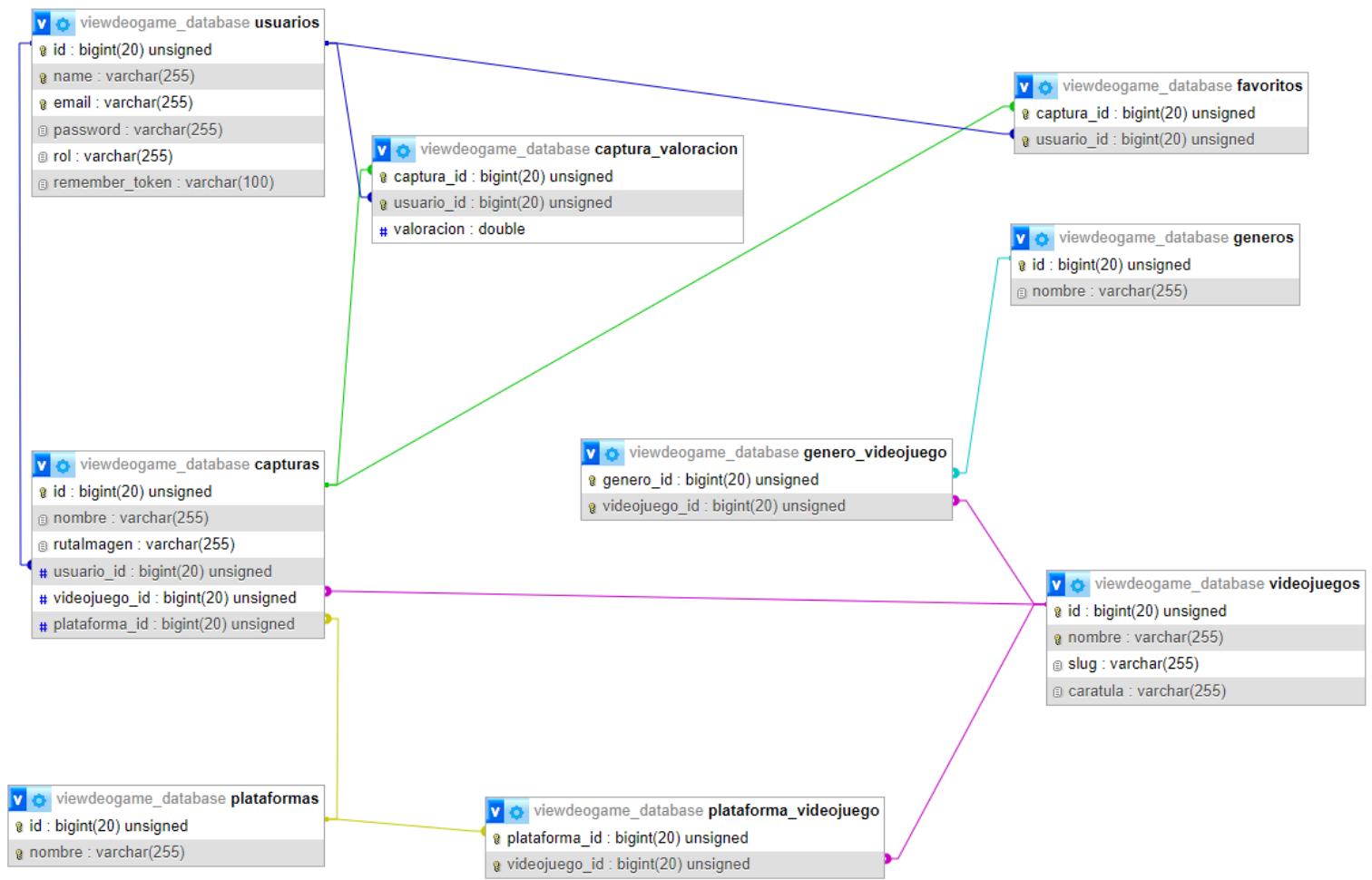
- **El formulario de subir captura** está formado por:
  - El campo de nombre (texto)
  - El campo de plataforma (select)
  - El campo de imagen (file)
- **Los formularios de subir y editar videojuego** están formados por:
  - El campo de nombre (texto)
  - El campo de plataformas (select)
  - El campo de géneros (select)
  - El campo de imagen (file)
- **El formulario de editar usuario** está formado por
  - El campo de nombre (texto)
  - El campo de email (email)

## 2.4. ESTRUCTURAS DE DATOS USADAS

### 2.4.1 INTRODUCCIÓN

En esta sección mostraré el diagrama representativo de mi base de datos, además de una explicación de sus diferentes tablas y relaciones.

### 2.4.2 DIAGRAMAS Y DESCRIPCIÓN



**De los videojuegos** nos interesa guardar un **id**, el **nombre**, un **slug** (para utilizar en la URL) y la **carátula** (la ruta donde esta está almacenada en el servidor).

**De los usuarios** nos interesa guardar el un **id**, el **nombre**, un **email**, la **contraseña** y su **rol** (admin o usuario normal)

**De las capturas** nos interesa guardar el **id**, un **nombre**, la **ruta** donde la imagen se encuentra en el servidor, el **id del usuario** que la subió, el **id del videojuego** al que pertenece y el **id de la plataforma** en la que se ha hecho.

**De las plataformas** nos interesa guardar un **id** y el **nombre**.

**De los géneros** nos interesa un **id** y el **nombre**.

**Un videojuego puede estar en varias plataformas, y una plataforma tiene varios juegos.**

**Un videojuego puede tener varias capturas, pero cada captura pertenece a un juego.**

**Un videojuego puede tener varios géneros, y un género puede pertenecer a varios videojuegos.**

Puedes sacar **varias capturas de diferentes juegos de la misma plataforma**, pero **cada captura tiene una plataforma asignada** (en la que se hizo).

**Un usuario puede subir varias capturas, y cada captura pertenece a un usuario, aunque una captura puede estar en las favoritas de varios usuarios, y un usuario puede tener varias capturas favoritas.**

**Una captura puede ser valorada por varios usuarios, y cada usuario puede valorar varias capturas.**

## 2.5. FASE DE PRUEBAS

### 2.5.1. PRUEBAS DE VALIDACIÓN DE CÓDIGO

Prueba	Resultado esperado	¿Éxito?
<b>Registrarse</b>	Se creará un usuario en la base de datos con su respectivo nombre, email, contraseña y rol.	Sí
<b>Login</b>	Nos loguearemos en la página web introduciendo el email y contraseña del usuario que acabamos de crear y se nos llevará a la página de inicio.	Sí
<b>Fetch videojuegos</b>	El frontend recibirá los videojuegos y sus capturas de la base de datos a través de un fetch al servidor	Sí
<b>Ordenar</b>	Los juegos aparecerán por defecto ordenados de manera descendente según su número de capturas, podremos hacer que ser ordenen de manera Ascendente.	Sí
<b>Filtrar Videojuegos</b>	Se filtrarán los videojuegos según su nombre, manteniéndose el orden que hayamos elegido.	Sí
<b>Acceder a página de videojuego</b>	Se accederá a la página de un videojuego haciendo click en su carta.	Sí
<b>Fetch Videojuego</b>	El frontend recibirá los datos del videojuego correspondiente y sus capturas a través de un Fetch al servidor.	Sí
<b>Ver datos de captura</b>	Hacer click en las capturas nos permitirá ver sus datos	Sí
<b>Valorar captura</b>	Usaremos las estrellas para asignar una valoración a la captura.	Sí
<b>Marcar como favorita</b>	Usaremos el check para añadir la captura a favoritas.	Sí
<b>Cambiar valoración</b>	Usaremos las estrellas para cambiar la valoración que le habíamos asignado	Sí

<b>Desmarcar como favorita</b>	La quitaremos de favoritas utilizando el check.	Sí
<b>Abrir formulario de subir captura</b>	Se nos mostrará el formulario para subir captura al pulsar el botón correspondiente	Sí
<b>Subir captura</b>	Se añadirá una captura a la base de datos con su respectivo nombre, plataforma, imagen, juego y usuario y se nos llevará a la página del videojuego de nuevo	Sí
<b>Eliminar Captura</b>	La captura se eliminará de la base de datos en cuanto pulsemos el botón de eliminar y confirmemos.	Sí
<b>Abrir formulario de Editar juego</b>	Se nos mostrará el formulario para editar el juego al pulsar el botón correspondiente.	Sí
<b>Editar juego</b>	Se actualizarán en la base de datos los datos del juego y se nos llevará a la página del videojuego de nuevo	Sí
<b>Fetch perfiles</b>	El frontend recibirá los datos de los perfiles correspondientes a través de un Fetch al servidor	Sí
<b>Buscar perfiles</b>	Podremos buscar los perfiles basándonos en su nombre	Sí
<b>Abrir perfil</b>	Hacer click en una de las líneas de la tabla correspondientes o en el nombre de perfil en el header nos mostrará la página del perfil correspondiente	Sí
<b>Fetch perfil</b>	El frontend recibirá los datos del perfil correspondiente con sus respectivas capturas favoritas y subidas a través de un Fetch al servidor	Sí
<b>Abrir formulario de editar Perfil</b>	Se nos mostrará el formulario de editar perfil al pulsar el botón correspondiente	Sí

<b>Editar perfil</b>	Se actualizarán en la base de datos los datos del usuario y se nos pedirá hacer login de nuevo	Sí
<b>Abrir formulario de subir videojuego</b>	Se nos mostrará el formulario de subir videojuego al pulsar el botón correspondiente.	Sí
<b>Subir videojuego</b>	Se añadirá un videojuego a la base de datos con su respectivo nombre, slug, plataformas, géneros y carátula, y se nos llevará a la página del videojuego de nuevo	Sí
<b>Eliminar Perfil</b>	Se eliminará al usuario de la base de datos y se nos llevará a la página de login	Sí
<b>Eliminar cualquier captura (administrador)</b>	Se eliminará la captura de la base de datos independientemente de quien la haya subido	Sí
<b>Eliminar cualquier perfil (administrador)</b>	Se eliminará el perfil de la base de datos, independientemente de quien sea, y nos llevará a la página de buscar perfiles	Sí
<b>Eliminar Videojuegos (administrador)</b>	Se eliminará el videojuego de la base de datos y nos devolverá a la página principal	Sí

#### 2.5.2. PRUEBAS DE IMPLANTACIÓN EN DISTINTOS SOPORTES

Dispositivo	Dimensiones	¿Éxito?
Monitor PC	2269 x 1226 (xl)	Sí
Nest Hub max	1280 x 800 (lg)	Sí
IPad pro	1024 x 1366 (md)	Sí

IPad Air	820 x 1180 (sm)	Sí
Samsung Galaxy s8+	360 x 740 (xs)	Sí

### 3. IMPLEMENTACIÓN

#### 3.1. TECNOLOGÍAS USADAS DURANTE EL DESARROLLO DEL PROYECTO

En este apartado procederé a listar las diferentes tecnologías que he utilizado durante el proyecto, además de dar una pequeña descripción de en qué consisten y una explicación de para qué las utilicé.

---

##### VISUAL STUDIO CODE



## Visual Studio Code

Visual Studio Code es un **editor de código disponible en todos los sistemas operativos Windows**.

Utilicé Visual Studio Code como entorno para **editar todo el código de la aplicación**, tanto **frontend** como **backend**, y para crear los archivos de **dockerfile** y **docker-compose**. El motivo por elegir Visual Studio Code por encima de cualquier otro editor de código es que **tengo mayor familiaridad con él**, además de que tiene **soporte integrado para JavaScript** y permite instalar **extensiones para incluir soporte de php, dockerfile e yml** y para **facilitar la escritura de código con varios snippets**. También incluye un **terminal** para poder introducir comandos, lo cual

me permitió realizar acciones como la instalación de paquetes de react o la creación de archivos en laravel de manera mucho más cómoda.

---

## COMPOSER



Composer es un **gestor de dependencias** que sirve para **instalar y actualizar las librerías que tu proyecto necesite simplemente declarándolas**.

Utilicé composer para **instalar Laravel y su librería: breeze** para gestionar la autenticación de usuarios.

---

## NPM



Npm es un **registro de software** utilizado para instalar y compartir paquetes.

Utilicé npm para **instalar todos los diferentes paquetes de react** que he utilizado en el desarrollo de la aplicación, además de para ejecutar los scripts de '**dev**', el cuál utilicé para probar y **debuggear la aplicación**, y '**build**', para **construir la aplicación una vez desplegada en el servidor**.

---

## VITE



Vite es una **herramienta de compilación** utilizada para proyectos de diferentes **librerías de JavaScript y TypeScript**.

Utilicé Vite para **compilar el frontend de mi aplicación** además de **generar la plantilla a partir de la cual lo desarollé**, además de para **importar los módulos de los diferentes paquetes de React** en las partes de la aplicación en las que necesitaba utilizarlos.

---

## LARAVEL



Laravel es un **framework de PHP** cuyo propósito es **proporcionar las bases de una aplicación web para poder facilitar su desarrollo**.

Utilicé laravel para **realizar todas las acciones del backend**, es decir, **manejar la base de datos, hacer comprobaciones de seguridad** (comprobar los tokens para evitar que un usuario sin los permisos necesarios o un usuario sin loguear realice

determinadas acciones), manejar las sesiones de usuario y responder a las peticiones del frontend enviando los datos u errores que se consideren oportunos. Laravel también incluye la interfaz de comandos “artisan”, la cual permite realizar acciones como migrar la base de datos o crear archivos destinados a determinadas funcionalidades con mucha más facilidad.

---

## REACT



React es una **librería de JavaScript** que sirve para **crear interfaces de usuario**.

Utilicé React para realizar todo el frontend de la aplicación, es decir, el **diseño de las páginas, los diferentes formularios y sus validaciones**, las **llamadas fetch al backend para obtener los datos necesarios** para cada una de las partes de la aplicación y para **subir, eliminar o actualizar registros en la base de datos**.

React utiliza para sus archivos la extensión **JSX**, la cual **combina JavaScript con HTML**, facilitando así el **crear páginas web dinámicas con el uso de objetos de JavaScript**.

---

MYSQL



MySQL es un sistema de base de datos relacional **open-source**.

Utilicé MySQL para **gestionar mi base de datos** utilizando el backend como medio.

---

DOCKER

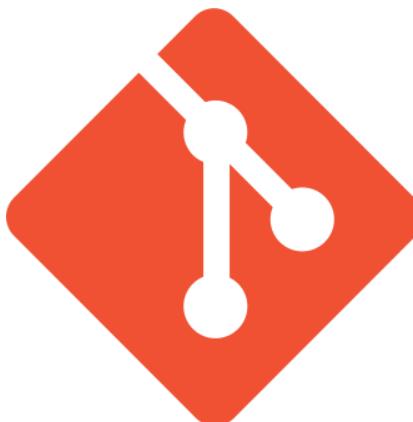


Docker es un **software de código abierto** utilizado para **desplegar aplicaciones utilizando contenedores**.

Utilicé docker para **desplegar en el servidor las diferentes partes de la aplicación en diferentes contenedores**, dentro de un mismo contenedor más grande que sirviera como conexión entre ellos, **haciendo uso de archivos Dockerfile y docker-compose**.

---

## GIT



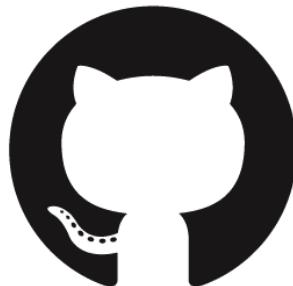
# git

Git es un **software de control de versiones** para llevar un **registro de los cambios realizados en archivos o carpetas**.

Utilicé git para **registrar los cambios que le hacía al backend y el frontend de la aplicación**, y después **subirlos a sus respectivos repositorios** en GitHub y **descargarlos en los diferentes ordenadores que utilizara para desarrollar la aplicación**.

---

## GITHUB



# GitHub

GitHub es una **plataforma para alojar proyectos mediante repositorios** utilizando Git.

Utilicé GitHub para **guardar el backend y el frontend de la aplicación** de manera que pudiera **continuar con el proyecto independientemente del ordenador que estuviera utilizando para desarrollar** en ese momento, y para que **los cambios que le hiciera a la página pudieran quedar almacenados** en caso de perder o borrar los archivos. También sirvió como **forma de acceder a anteriores versiones del proyecto y volver atrás si era necesario**.

## 4. PUESTA EN MARCHA. DESPLIEGUE

### 4.1. GUIA DE DESPLIEGUE DEL PROYECTO

En este punto detallaré como desplegar el proyecto en un servidor utilizando Docker.

Todos los archivos locales que utilicemos deben de estar en la misma carpeta, la cual será la que subamos al servidor.

#### PASO 1: CREAR LOS DOCKERFILE

Primero se crearán los **dockerfile** para cada una de las partes del proyecto, en estos instalaremos todo lo necesario para que el proyecto funcione y pueda ser accesible desde internet, además de ejecutar todos los comandos necesarios para ponerlo en marcha:

**DOCKERFILE FRONTEND:** Se utilizará la imagen de **node**. Se creará el **directorio para guardar los archivos**, se creará el **repositorio local y nos moveremos a él**, se le asignará el **repositorio remoto**, y se hará un **pull** de él. **Se instalarán los módulos necesarios** para el funcionamiento de la aplicación en el servidor y se realizará un **npm run build** para construirla. Se instalará **nginx** para utilizarlo como **proxy inverso**, se copiarán los **archivos de build** desde **dist** hasta **/var/www/html**, se copiarán los **archivos de configuración** de nginx y de la aplicación **desde nuestra carpeta local** a sus respectivos directorios y se **creará un enlace al archivo de configuración de la página desde sites-enabled** para poder acceder a ella. Por último, se ejecutará el comando “**nginx -g daemon-off**” para que nginx se ejecute continuamente.

```
FROM node:20.11.1
WORKDIR /app
RUN mkdir Viewdeogame-Frontend-Pablo-Fernandez
WORKDIR /app/Viewdeogame-Frontend-Pablo-Fernandez
RUN git init
RUN git remote add origin https://Pablobelmo:ghp_F60GiQBzuXPK5uaH0gOFCKSHcYpuAT1x0J2o@github.com/daw2-23-24-pdaw/Viewdeogame-Frontend-Pablo-Fernandez.git
RUN git pull origin main
RUN npm install
RUN npm install @rollup/rollup-linux-x64-gnu
RUN npm run build
RUN apt-get update && apt-get install nginx -y
RUN cp -r /app/Viewdeogame-Frontend-Pablo-Fernandez/dist/* /var/www/html/
COPY viewdeogame.conf /etc/nginx/sites-available/viewdeogame.conf
COPY nginx.conf /etc/nginx
RUN ln -s /etc/nginx/sites-available/viewdeogame.conf /etc/nginx/sites-enabled
EXPOSE 80
CMD ["nginx", "-g", "daemon off;"]
```

(Los pasos desde crear el directorio para los archivos hasta hacer el pull se pueden saltar simplemente clonando el repositorio, pero en el caso del frontend esto me dio problemas ya que no hacía pull del último commit, así que decidí iniciar el repositorio yo mismo)

```

server {
    listen 80;
    server_name pabloff24.iesmontenaranco.com www.pabloff24.iesmontenaranco.com;
    root /var/www/html;
    location / {
        try_files $uri /index.html;
    }
}

```

Ejemplo de archivo .conf del frontend

```

events {
    worker_connections 768;
    # multi_accept on;
}

http {
    ##
    # Basic Settings
    ##
    client_max_body_size 20M;
    sendfile on;
    tcp_nopush on;
    types_hash_max_size 2048;
    # server_tokens off;

    # server_names_hash_bucket_size 64;
    # server_name_in_redirect off;

    include /etc/nginx/mime.types;
    default_type application/octet-stream;

    ##
    # SSL Settings
    ##

    ssl_protocols TLSv1 TLSv1.1 TLSv1.2 TLSv1.3; # Dropping SSLv3, ref: POODLE
    ssl_prefer_server_ciphers on;

    ##
    # Logging Settings
    ##

    access_log /var/log/nginx/access.log;

    ##
    # Gzip Settings
    ##

    gzip on;

    # gzip_vary on;
    # gzip_proxied any;
    # gzip_comp_level 6;
    # gzip_buffers 16 8k;
    # gzip_http_version 1.1;
    # gzip_types text/plain text/css application/json application/javascript text/xml application/xml application/xml+rss text/javascript;

    ##
    # Virtual Host Configs
    ##

    include /etc/nginx/conf.d/*.conf;
    include /etc/nginx/sites-enabled/*;
}

```

Ejemplo de nginx.conf

**DOCKERFILE BACKEND:** Se utilizará la imagen de **php-apache**. Nos moveremos al directorio **/var/www/html**, ahí instalaremos **git**, **zlib1g-dev**, **libzip-dev** (estos dos últimos son necesarios para poder instalar zip) y las extensiones de php de docker para **zip**, **pdo** y **pdo\_mysql** (estos dos últimos son necesarios para poder acceder a la base de datos). **Copiaremos el archivo de configuración de php a /usr/local/etc/php/php.ini** (este paso es opcional, solo hazlo en caso de que quieras cambiar cosas de la configuración base de php). **Clonaremos el repositorio del backend en la carpeta y copiaremos el archivo .env** (será el mismo archivo que utilices en el backend pero con los cambios necesarios para que funcione en el servidor, como la base de datos o la url del frontend) **a la carpeta creada por la clonación**. Descargaremos **composer**. Nos moveremos hasta el backend, realizaremos un git pull (este paso es solo para asegurarse de que se copiaba el último commit), **actualizaremos composer e instalaremos los módulos correspondientes** y **utilizaremos el comando apache2-foreground para que apache se ejecute continuamente**. Otorgaremos permisos a los directorios necesarios, desactivaremos la página por defecto de apache, copiaremos el archivo de configuración de nuestra página a **/etc/apache2/sites-available** y la **activaremos**. Por último, **activaremos el mod “rewrite”** para evitar problemas al hacer peticiones a la URL.

```
FROM php:8.2-apache
WORKDIR /var/www/html
RUN apt-get update && apt-get install -y git \
    zlib1g-dev \
    libzip-dev
RUN docker-php-ext-install zip
RUN docker-php-ext-install pdo_mysql
COPY php.ini /usr/local/etc/php/php.ini
RUN git clone https://Pablobelmo:ghp_F60GiQBzuXPK5uaH0g0FCkSHcYpuAT1x0J2o@github.com/daw2-23-24-pdaw/Viewdeogame-Backend-Pablo-Fernandez.git
COPY .env /var/www/html/Viewdeogame-Backend-Pablo-Fernandez
RUN curl -sS https://getcomposer.org/installer | php -- --install-dir=/usr/local/bin --filename=composer
WORKDIR /var/www/html/Viewdeogame-Backend-Pablo-Fernandez
RUN git pull origin main
RUN composer update
RUN composer install
CMD ["apache2-foreground"]
RUN chown -R www-data:www-data /var/www/html/Viewdeogame-Backend-Pablo-Fernandez/storage /var/www/html/Viewdeogame-Backend-Pablo-Fernandez/bootstrap/cache
RUN chmod -R 775 /var/www/html/Viewdeogame-Backend-Pablo-Fernandez/storage
RUN chmod -R 775 /var/www/html/Viewdeogame-Backend-Pablo-Fernandez/bootstrap/cache
RUN chmod -R 777 /var/www/html/Viewdeogame-Backend-Pablo-Fernandez/public
RUN chmod -R 777 /var/www/html/Viewdeogame-Backend-Pablo-Fernandez/public/capturas
RUN chmod -R 777 /var/www/html/Viewdeogame-Backend-Pablo-Fernandez/public/caratulas
RUN a2dissite 000-default.conf
COPY web_laravel.conf /etc/apache2/sites-available
RUN a2ensite web_laravel.conf
RUN a2enmod rewrite
EXPOSE 80
```

```
<VirtualHost *:80>
    ServerAdmin pablo@example.com
    ServerName laravel.example.com
    DocumentRoot /var/www/html/Viewdeogame-Backend-Pablo-Fernandez/public/
    <Directory /var/www/html/Viewdeogame-Backend-Pablo-Fernandez/public/>
        AllowOverride All
        Require all granted
    </Directory>
    ErrorLog /var/log/apache2/error.log
    CustomLog /var/log/apache2/access.log combined
</VirtualHost>
```

Ejemplo de archivo de configuración del backend

**DOCKERFILE BASE DE DATOS:** Utilizaremos la última imagen de **mysql**. El dockerfile es muy simple, solo tenemos que copiar el **archivo sql de nuestra base de datos en /docker-entrypoint-initdb.d** para que lo ejecute al **crearse el contenedor**. El archivo sql debe de estar **dentro de otra carpeta dentro de la que estamos trabajando** (en este caso la carpeta en la que está el archivo es “db”)

```
FROM mysql:latest
ADD db /docker-entrypoint-initdb.d
EXPOSE 80
```

---

## PASO 2: CREAR EL DOCKER-COMPOSE.YML

Se creará un archivo **docker-compose**, en este se **indicarán los diferentes contenedores que se quieran crear**, y dentro de estos sus **nombres**, el **dockerfile** desde el que se quieren crear las imágenes junto con la carpeta en la que se encuentran (en este caso será “.” porque se encuentran en la misma carpeta), los **puertos** a los que los asignamos, el **volumen** en el que persistirán los datos, **si queremos que inicie cada vez que lo hace docker y, si dependen de otro contenedor, indicarlo**. En el contenedor de la base de datos también indicaremos el nombre de la base de datos, el usuario, la contraseña del usuario y la contraseña del root.

```
version: '3.3'
services:
  db:
    container_name: mysql_db
    build:
      dockerfile: Dockerfile-db
    context: .
    environment:
      MYSQL_DATABASE: viewdeogame-database
      MYSQL_USER: admin123
      MYSQL_PASSWORD: narancopablo
      MYSQL_ROOT_PASSWORD: root
    volumes:
      - db_data:/var/lib/mysql
    ports:
      - 3306:80
    restart: always
  laravel:
    container_name: Viewdeogame-Backend
    build:
      dockerfile: Dockerfile-backend
    context: .
    volumes:
      - laravel_data:/var/www/html
    ports:
      - 8000:80
    restart: always
    depends_on:
      - db
  react:
    container_name: Viewdeogame-Frontend
    build:
      dockerfile: Dockerfile-frontend
    context: .
    volumes:
      - react_data:/var/www/html
    ports:
      - 80:80
    restart: always
    depends_on:
      - laravel
  volumes:
    laravel_data:
    react_data:
    db_data:
```

---

### PASO 3: SUBIRLO Y DESPLEGARLO EN EL SERVIDOR

Lo primero que haremos será **subir los archivos de nuestra carpeta de despliegue** (en la que se mencionó al principio de este apartado que estarían todos los dockerfile, docker-compose y otros archivos necesarios) al servidor utilizando el comando “**scp -r \*ruta de la carpeta\* \*nombre de usuario\*@\*dominio\*:\***carpeta a la que queramos subir los archivos”” (es probable que el .env no se copie, y tengamos que subirlo individualmente con “**scp \*ruta de la carpeta\*/.env**”). **Nos conectaremos al servidor utilizando ssh en la consola de comandos**, una vez dentro, **nos moveremos a la carpeta en la que tengamos el servidor usando cd \*ruta correspondiente\***.

**Instalaremos docker si es necesario y utilizaremos el comando “docker login”** e introduciremos el nombre de usuario y contraseña de nuestra cuenta de docker (si no se tiene, se debe crear una para este paso). Una vez nos hayamos logueado, y estando en la misma carpeta en la que están todos los archivos, **utilizaremos el comando “docker-compose up -d”**, este comando utilizará el archivo docker-compose.yml para crear las imágenes, volúmenes y contenedores necesarios y desplegará el proyecto. **Una vez haya terminado el proceso podremos conectarnos a la web utilizando la url correspondiente.**

**Si hemos hecho cambios en el frontend y queremos que estos se reflejen**, solo tendremos que **entrar en el contenedor utilizando “docker exec -t \*nombre del contenedor\* bash”**, y una vez dentro hacer “**git pull**”. Después haremos “**npm run build**”, y cuando finalice introduciremos el comando “**cp -r dist/\* /var/www/html**”. **Para el backend** solo tendremos que utilizar “**docker exec -t \*nombre del contenedor\* bash**” y “**git pull**”. **Si hay problemas** a la hora de ver los cambios reflejados en la web, **reinicia los contenedores con “docker restart \*nombre del contenedor”**.

## 5. APLICACIÓN EN EL ÁMBITO EMPRESARIAL

### 5.1. IDEA DE NEGOCIO

Podría **publicitar mi página web por las redes sociales** ya sea a base de **pagar anuncios o compartiéndola yo mismo por lugares como Twitter**. Una vez mi página creciera en popularidad podría intentar **desarrollar otras páginas web y anunciarlas en esta**, aprovechando su popularidad para darles a estas nuevas páginas un impulso inicial.

Una vez tuviera suficiente dinero, podría **contratar a gente y formar una empresa desarrolladora en lugar de ser un desarrollador independiente**.

### 5.2. PLANIFICACIÓN DE NECESIDADES DE RRHH

En un principio podría contratar a un **moderador que se asegurara que se suben imágenes adecuadas** a los videojuegos correctos, y que **se encargara de eliminar las que no correspondan al videojuego** en el que se han subido y **borrar el usuario que las ha subido en caso de que se repita la incidencia regularmente**.

Una vez la página creciera, necesitaría contratar a más gente para **arreglar bugs, o mantener el servidor**. También podría **contratar a más moderadores** para que se encarguen de cosas como **asegurar que los videojuegos que se suben existen y tienen el nombre y carátula correctos**.

Si añadiera más funcionalidades a la página podría **contratar a otros desarrolladores para poder añadir más cosas en menor tiempo**.

Cuanto más creciera el negocio y más páginas se creasen, a más gente debería contratar para realizar los diferentes trabajos en las diferentes páginas.

### 5.3. PLANIFICACIÓN DE NECESIDADES DE PRODUCCIÓN

Necesitaría comprar **ordenadores** para poder **llevar a cabo el desarrollo de esta y otras páginas**, además de **otras máquinas** para poder **alojar los servidores** que sean necesarios.

También necesitaré otras cosas como **pantallas, teclados, ratones y escritorios** para que los empleados puedan llevar a cabo su trabajo en las páginas.

### 5.4. PLANIFICACIÓN DE LA PREVENCIÓN DE RIESGOS LABORALES

Necesitaría **monitores de buena calidad** para **reducir el cansancio en los ojos** debido a las varias horas que los desarrolladores estarán en frente de una pantalla, también necesitaría **una buena iluminación en la oficina**, para que sean capaces de **ver la pantalla**. Para **evitar lesiones en las manos y muñecas, necesitaría teclados y ratones ergonómicos**. Para evitar **lesiones en la espalda** por estar varias horas sentado, necesitaría **sillas cómodas**.

También podría hacer que los **nuevos empleados necesitaran pasar un curso de prevención de riesgos laborales**.

## 5.5. PREVISIONES DE INVERSIÓN Y FINANCIACIÓN

Podría pagar dinero para **publicitar la página en las redes sociales** o hacerlo yo mismo. También podría **pagar a influencers y otros tipos de personalidades mediáticas en internet para que la publiciten en sus vídeos o streams**.

Para financiar la página podría **poner anuncios en esta**, o incluso **abrir un patreon para que aquellos que deseen donar a la página pudieran hacerlo**, y a cambio hubiera **una recompensa para aquellos que hayan donado una cantidad específica** o más o un **nuevo añadido a la página en caso de que se alcance una meta de dinero donado**.

## 5.6. ESTUDIO DE MERCADO Y MARKETING OPERATIVO

El mercado objetivo de esta página serían **las personas que jueguen a videojuegos y que tengan acceso totalmente libre a internet**, por lo que estaría principalmente dirigida a **adolescentes y jóvenes adultos**. La mayoría de las páginas de videojuegos se dedican a la compraventa de estos o a reviews y opiniones de los juegos en sí, así que **centrar la página en las capturas de videojuegos sería una fácil manera de diferenciarla del resto**, además de **presentar a la gente una manera rápida de compartirlas**, aprovechando cosas como que **twitter ya no sea compatible con la funcionalidad de compartir de las consolas, debido a los cambios recientes en su API**.

El marketing operativo se centraría en lo ya dicho, **publicitarla agresivamente en las redes sociales como twitter, servidores de Discord, o subreddits de videojuegos** e intentar pagar a **influencers y personalidades mediáticas** para que la patrocinien. También se podría poner anuncios en páginas que permitan la publicidad de otras páginas web.

# 6. CONCLUSIÓN

## 4.1. VALORACIÓN DEL TRABAJO REALIZADO

Creo que mi trabajo y proyecto final ha sido el **adecuado debido al tiempo disponible para desarrollarlo**. He intentado **aprovechar todo el tiempo extra que tuviera** como por ejemplo **las horas que me dejaron para hacerlo durante las prácticas** o **las primeras semanas después de semana santa**, las cuales **aproveché para realizar cosas** que me pudieran llevar más tiempo luego **como la instalación de todo lo necesario para el backend y el frontend o la gestión de usuarios y arreglar problemas de CORS y csrf**.

## 4.2. POSIBLES AMPLIACIONES

En el caso de dedicar más tiempo a esta página, dos de las ampliaciones que se me ocurre serían **un sistema de comentarios y un sistema de mensajes entre usuarios**. Estas dos cosas **permitirían que los usuarios pudieran dar sus opiniones concretas de las capturas que valoren y comunicarse con otros usuarios para hablar de los videojuegos que les gustan o pasarse capturas entre ellos**.

Otra cosa podría ser un **sistema de seguimiento de usuarios o de videojuegos**, en el que **el usuario podría decidir seguir a otro usuario o a un videojuego, y recibir notificaciones cuando estos tuvieran alguna actualización o nuevas capturas**.

También podría utilizar la **API de IGDB**, la cuál me **permitiría conseguir directamente todos los datos de cualquier videojuego**, así pudiendo centrarme totalmente en la parte de las capturas en lugar de tener que añadir cosas como **editar y subir videojuegos**, la razón por la que no hice esto es que **IGDB requiere registrar la aplicación en Twitch developers para poder utilizarla**, algo que **podría generar dificultades debido a que es un proyecto de clase**. Además, **IGDB es de pago si quieres ganar dinero con la página web**, así que incluso **si siguiera con ella, sería algo que no implementaría salvo que supiera al 100% si voy o quiero ganar beneficios**.

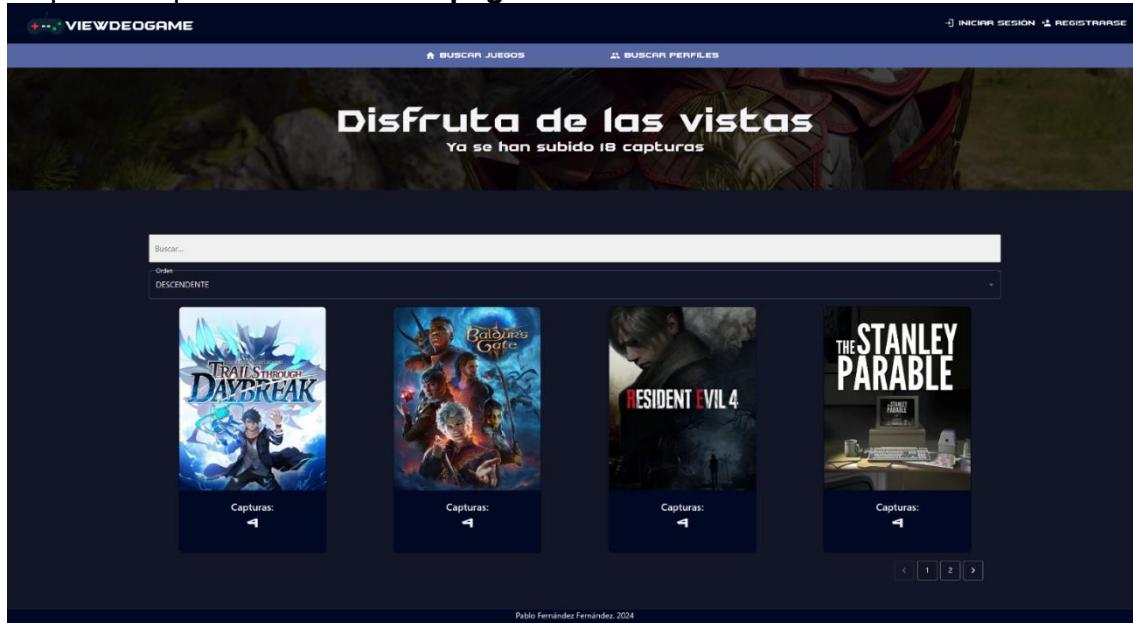
## 7. BIBLIOGRAFÍA Y WEBGRAFÍA

- **Composer:** <https://getcomposer.org/>
- **Laravel:** <https://laravel.com/>
- **Laravel-Breeze:** <https://github.com/laravel/breeze>
- **Documentación de Laravel:** <https://laravel.com/docs/11.x/readme>
- **Node:** <https://nodejs.org/en>
- **NPM:** <https://www.npmjs.com/>
- **Vite:** <https://vitejs.dev/>
- **React:** <https://es.react.dev/>
- **Documentación de React:** <https://es.react.dev/learn>
- **MUI:** <https://mui.com/>
- **Material icons:** <https://mui.com/material-ui/material-icons/>
- **Documentación de MUI:** <https://mui.com/material-ui/getting-started/>
- **React-Cookie:** <https://www.npmjs.com/package/react-cookie>
- **React-Router:** <https://reactrouter.com/en/main>
- **React-notifications:** <https://www.npmjs.com/package/react-notifications>
- **Styled Components:** <https://styled-components.com/>

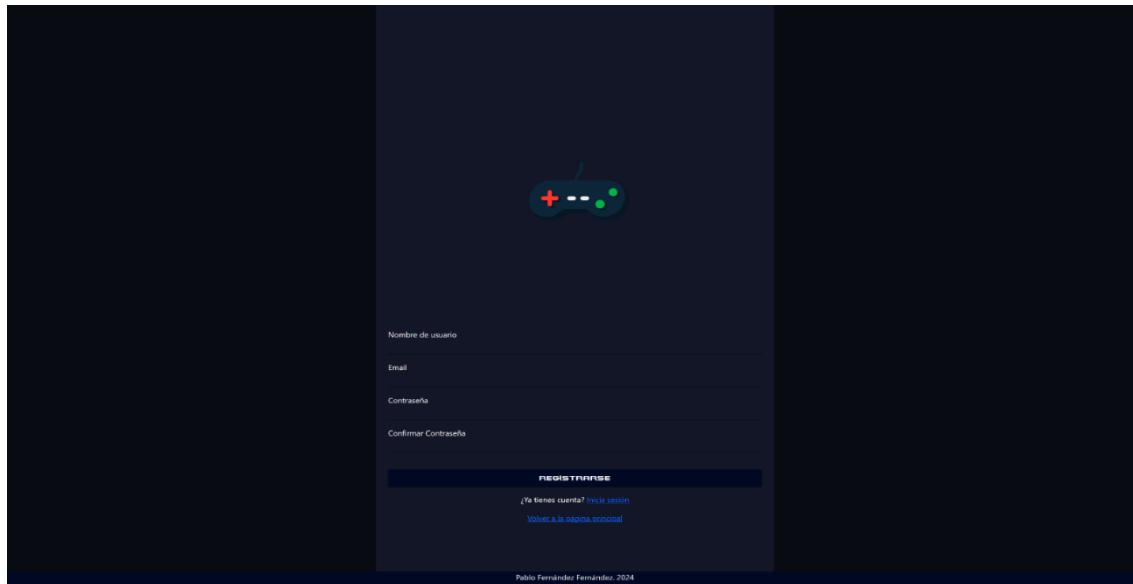
## 8.ANEXO

### 1.MANUAL DE USUARIO

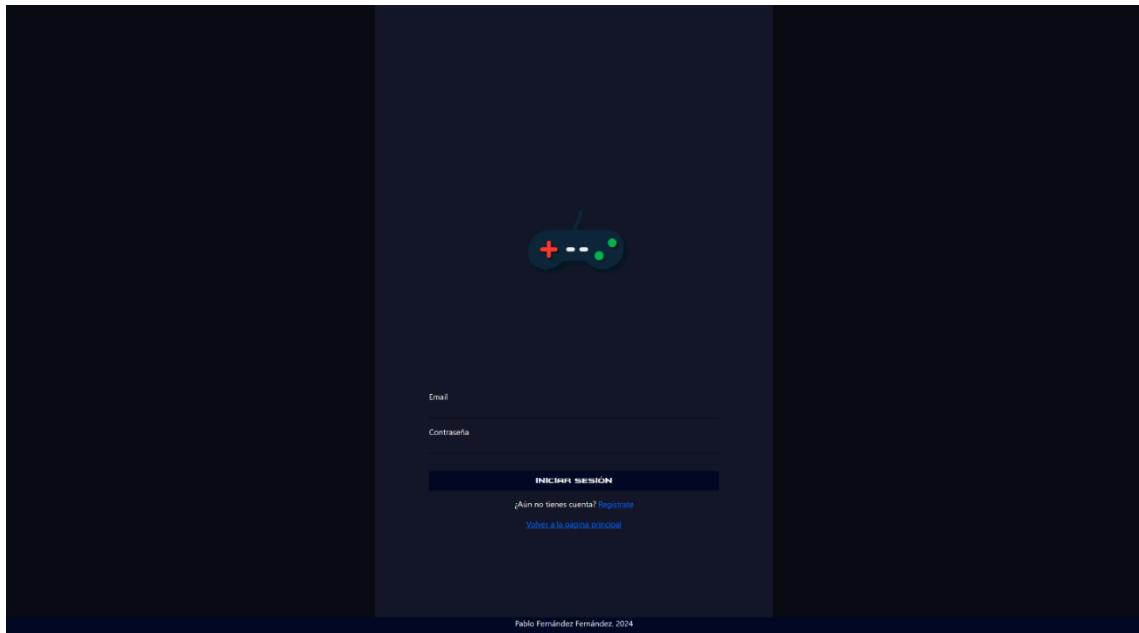
Lo primero que veremos será la página de inicio:



Desde aquí podremos realizar varias acciones dependiendo del botón que pulsemos. Empezaremos con el botón de **registrarse**, situado en la **parte derecha del header**:



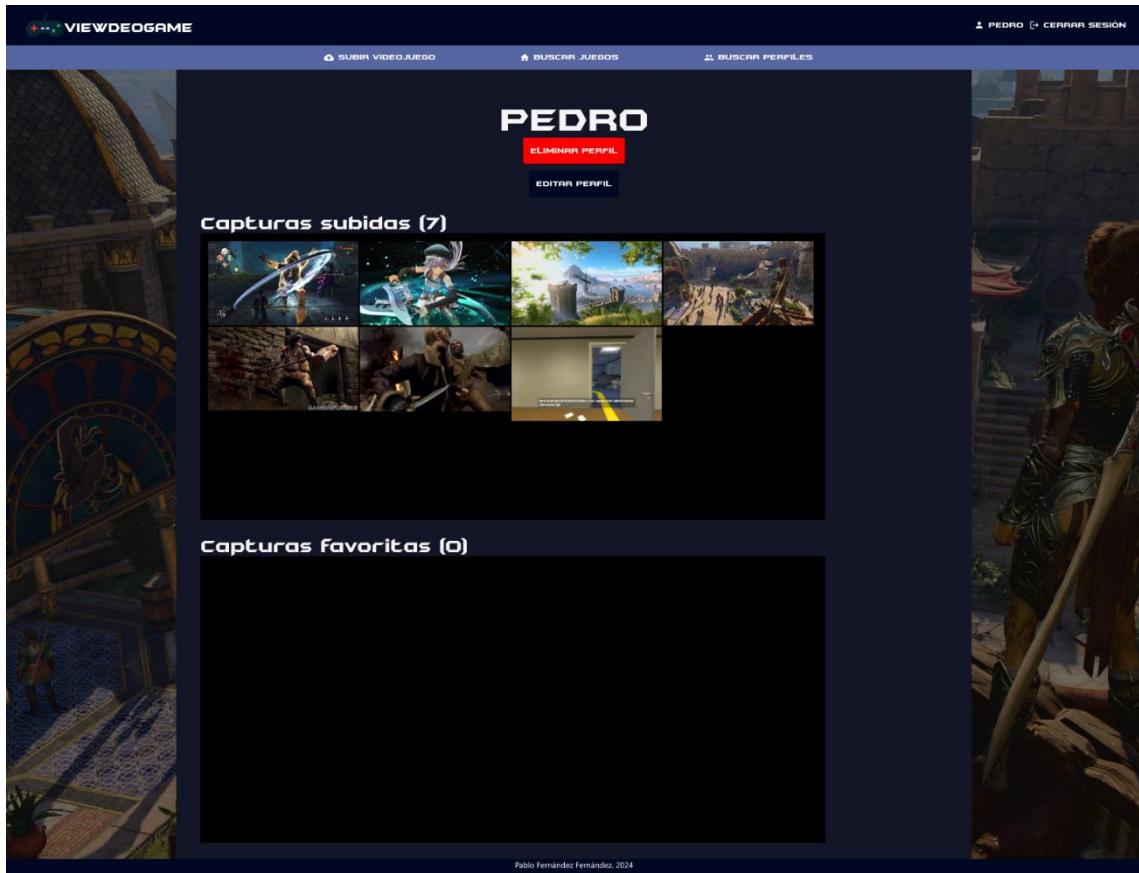
Nos aparecerá un formulario para crear un usuario en la página, en él tendremos que introducir un **nombre de usuario** (el cual no puede superar los 20 caracteres), un **Email** y una **contraseña** (la cual deberá superar los 8 caracteres) la cual tendremos que confirmar. Tras rellenar todos los campos correctamente y pulsar el botón de registrarse, se nos enviará a la página de Login.



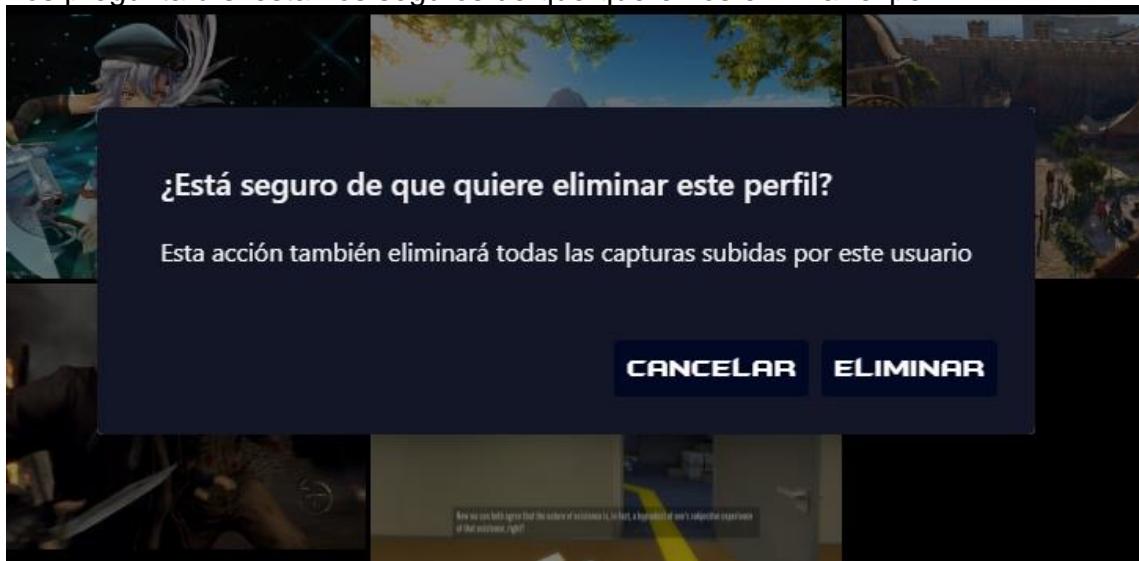
En este formulario deberemos introducir el Email y contraseña que hayamos indicado al registrarnos anteriormente. Una vez hayamos introducido los datos correctos, hacemos click en "iniciar sesión" y nos devolverá a la página de inicio.



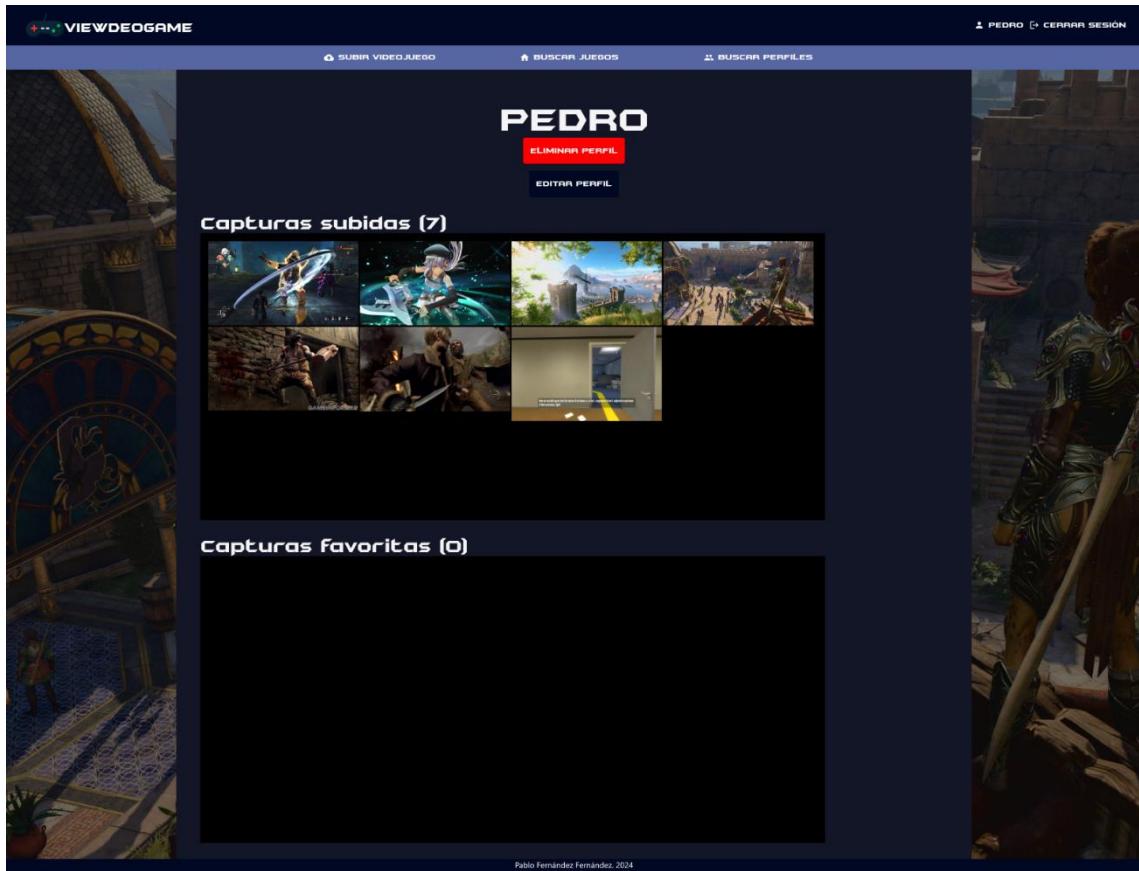
Como se puede ver, los botones de "iniciar sesión" y "registrarse" ahora han cambiado por nuestro nombre de usuario y "cerrar sesión". Como su nombre indica, pulsar el botón de cerrar sesión cerrará la sesión del usuario con el que nos hayamos logueado y nos devolverá a la página de inicio. Pulsar el botón con nuestro nombre de usuario nos llevará a nuestro perfil.



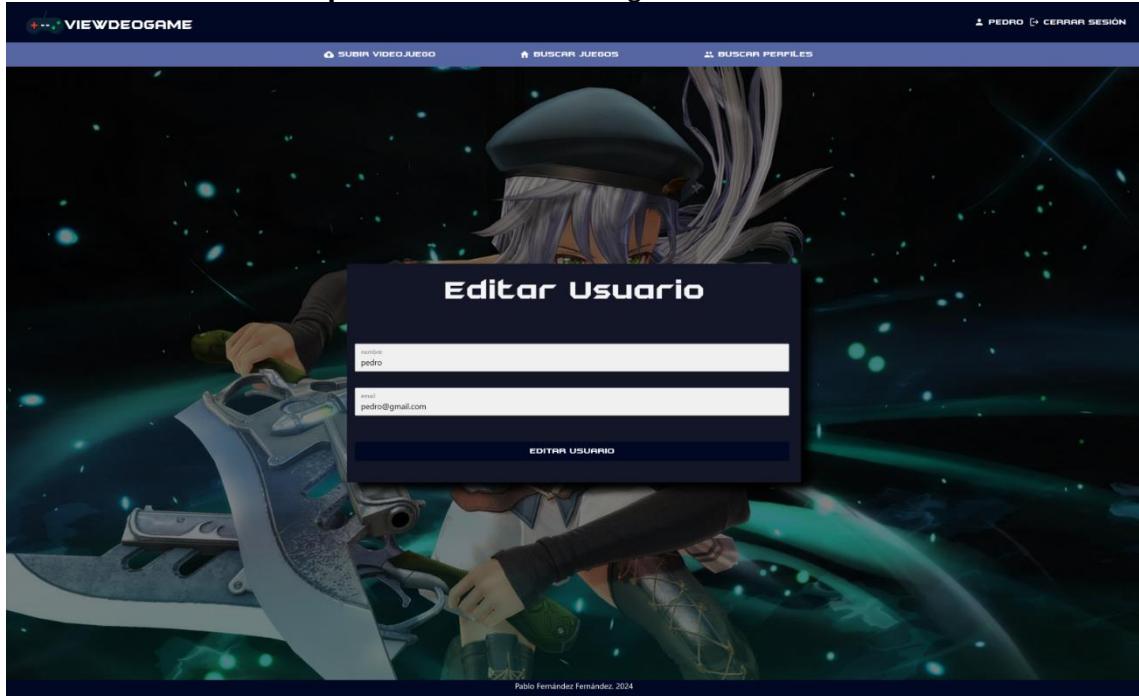
Debajo de nuestro nombre de usuario se pueden ver los botones de **"eliminar perfil"** y **"editar perfil"**. Pulsar el botón de **eliminar perfil** nos abrirá un aviso en el que nos preguntará si estamos seguros de que queremos eliminar el perfil.



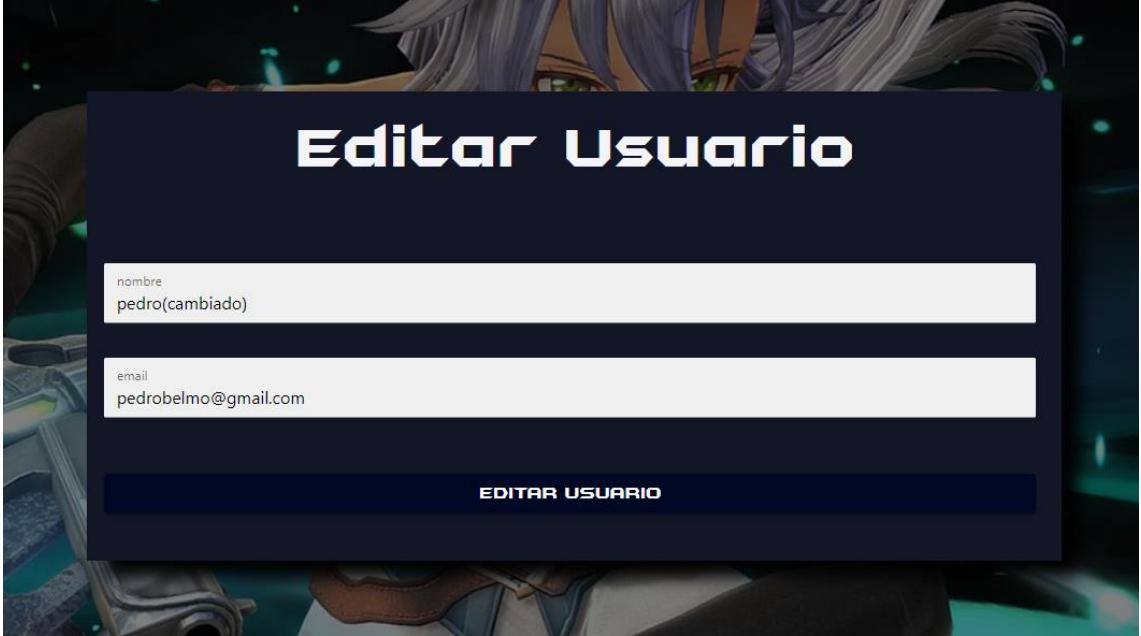
**Si elegimos “eliminar”, nuestro usuario se borrará** y nos enviará de vuelta a la página de login, si pulsamos cancelar, el diálogo se cerrará y continuaremos en la página de usuario.



Pulsar el botón “editar perfil” nos abrirá el siguiente formulario:



En este formulario nos aparecerán los campos de **nombre de usuario** y **email** con sus valores actuales. Si decidimos cambiar cualquiera de los dos valores, o ambos, y **pulsamos el botón de "editar usuario"**, se nos llevará de vuelta a la página de **Iniciar sesión**, donde tendremos que introducir de nuevo nuestro nuevo email (si lo hemos cambiado) y nuestra contraseña.



# Editar Usuario

nombre  
pedro(cambiado)

email  
pedrobelmo@gmail.com

**EDITAR USUARIO**

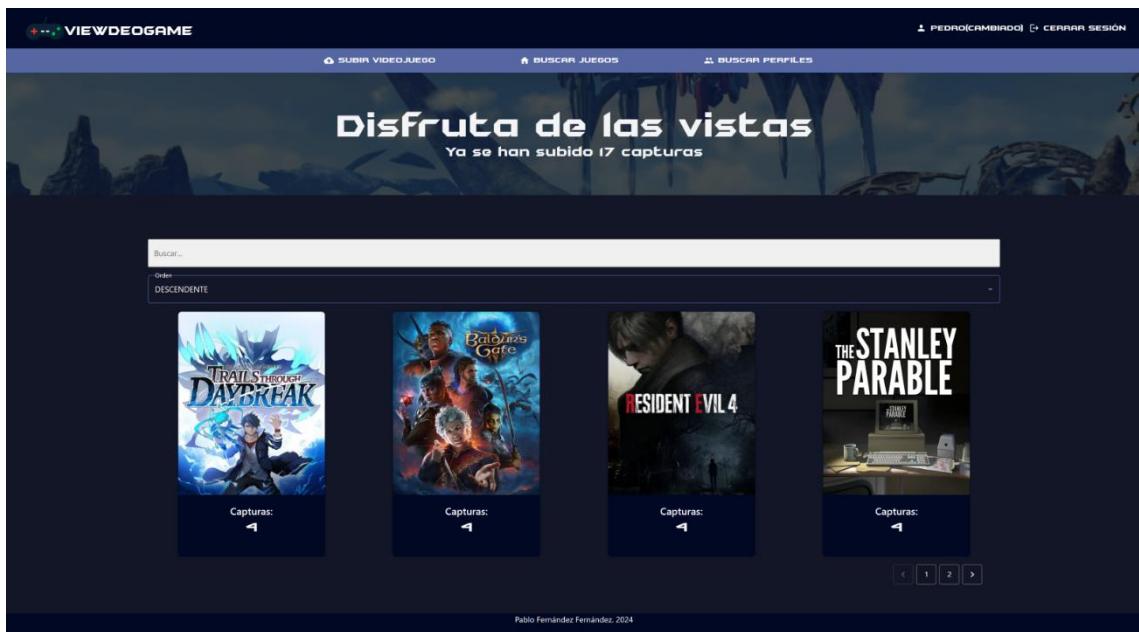
Email  
pedrobelmo@gmail.com

Contraseña  
.....|

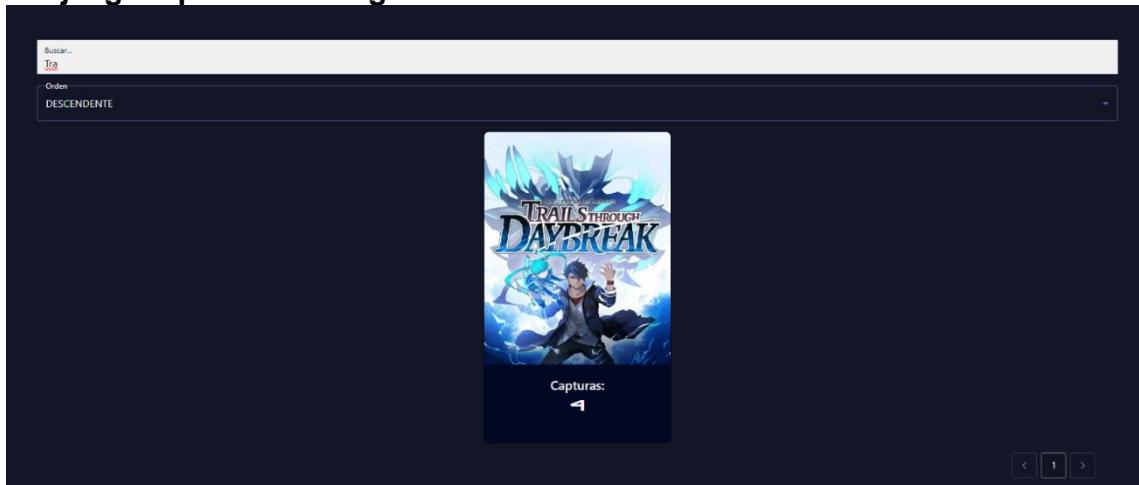
**INICIAR SESIÓN**

¿Aún no tienes cuenta? [Regístrate](#)

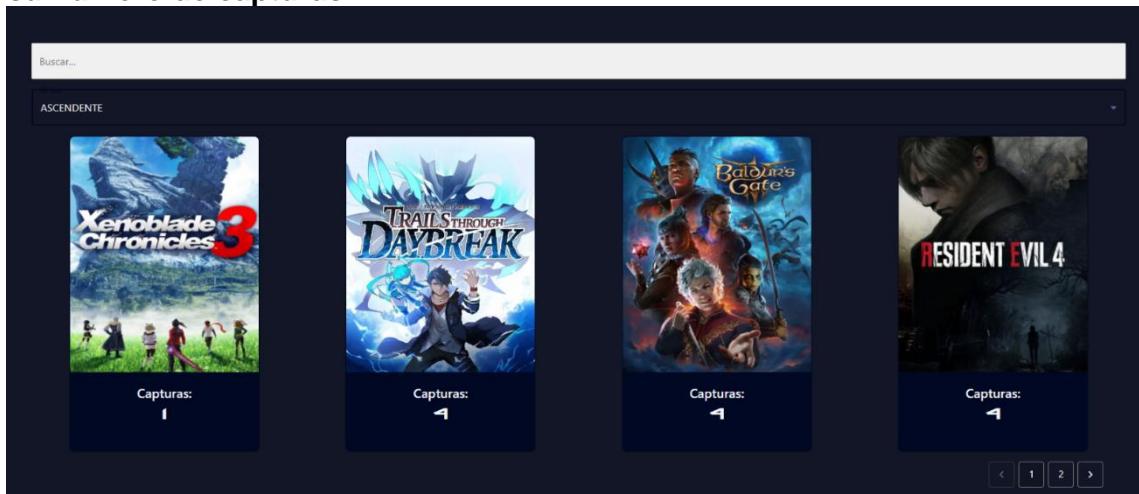
[Volver a la página principal](#)



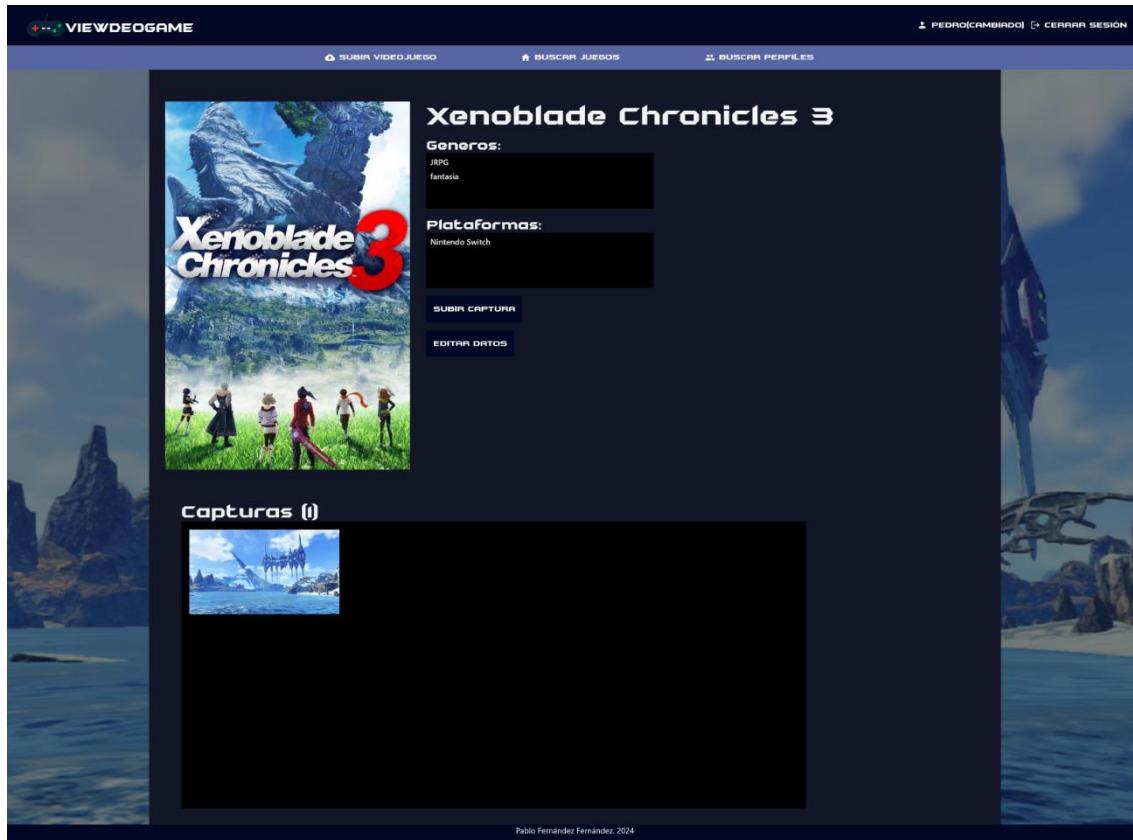
De nuevo en la página de inicio, veremos un campo de texto que pone **"Buscar..."** y un select que pone **"Descendente"**, modificar el campo de búsqueda filtrará los juegos que se ven según su nombre.



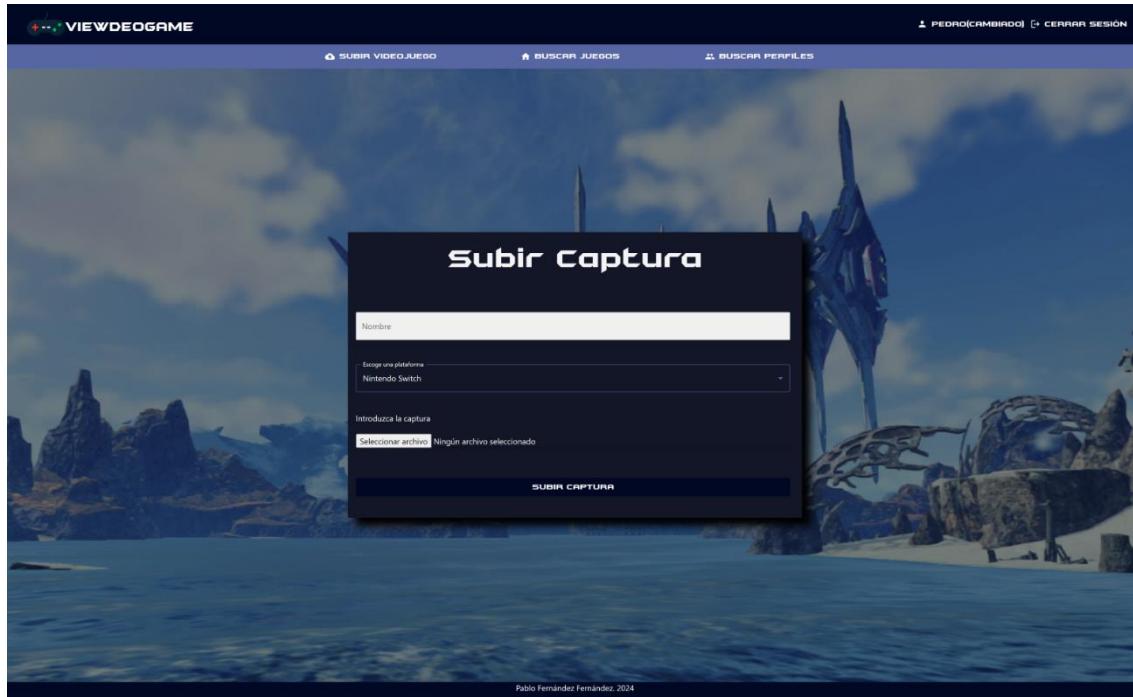
Modificar el select cambiará el orden en el que se muestran los juegos, según su número de capturas.



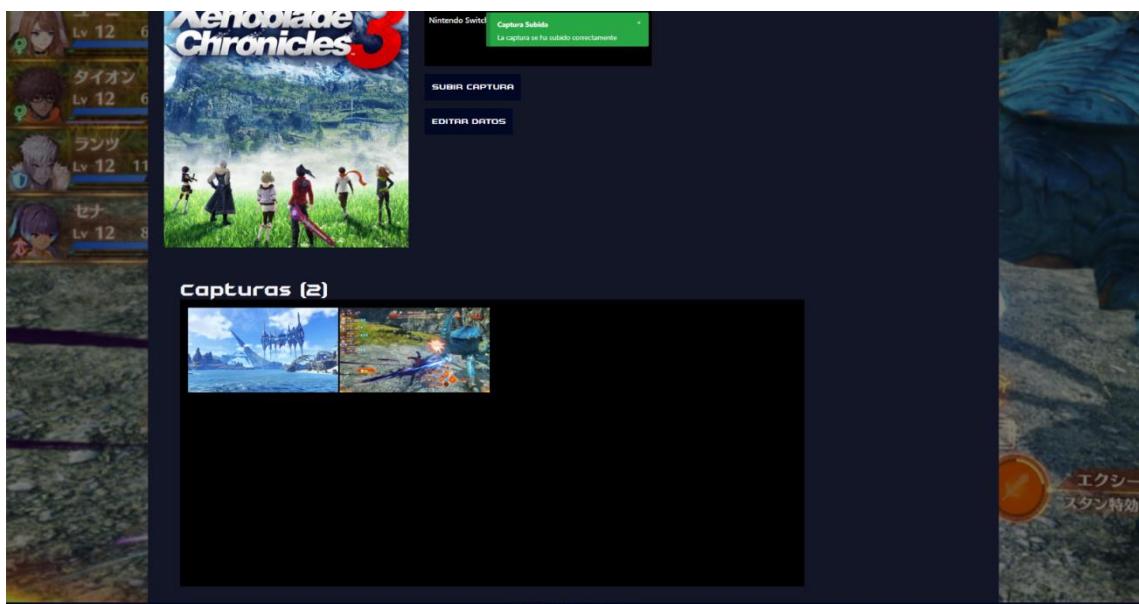
Pulsar en cualquiera de los juegos, nos llevará a su página.



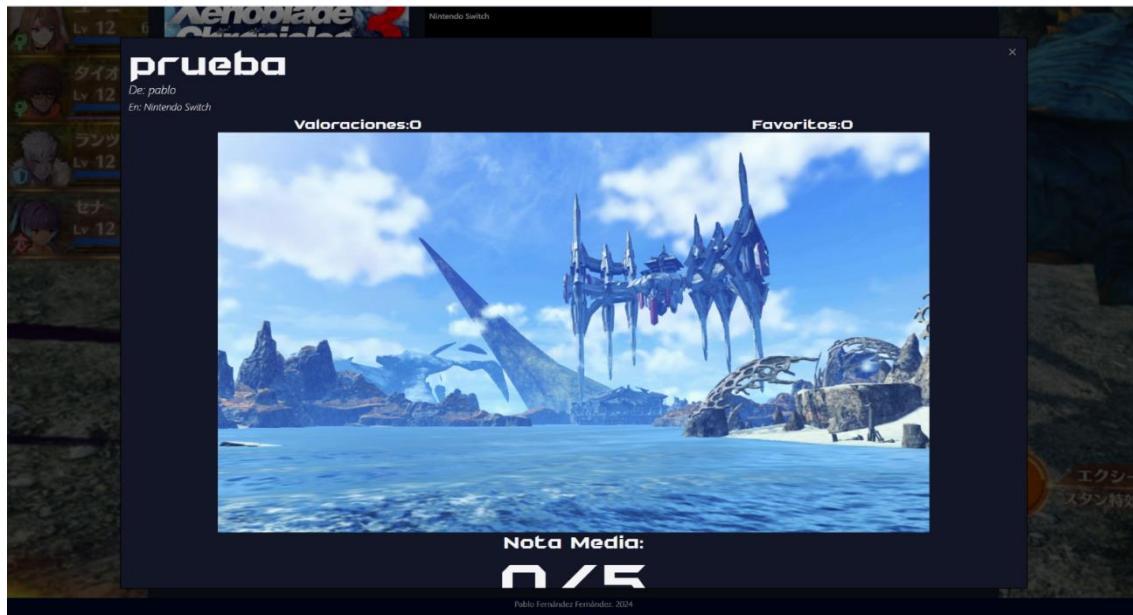
Lo primero que veremos debajo de las plataformas del juego serán los botones de subir captura y editar datos. Pulsar en **"subir captura"** nos llevará a un nuevo formulario:



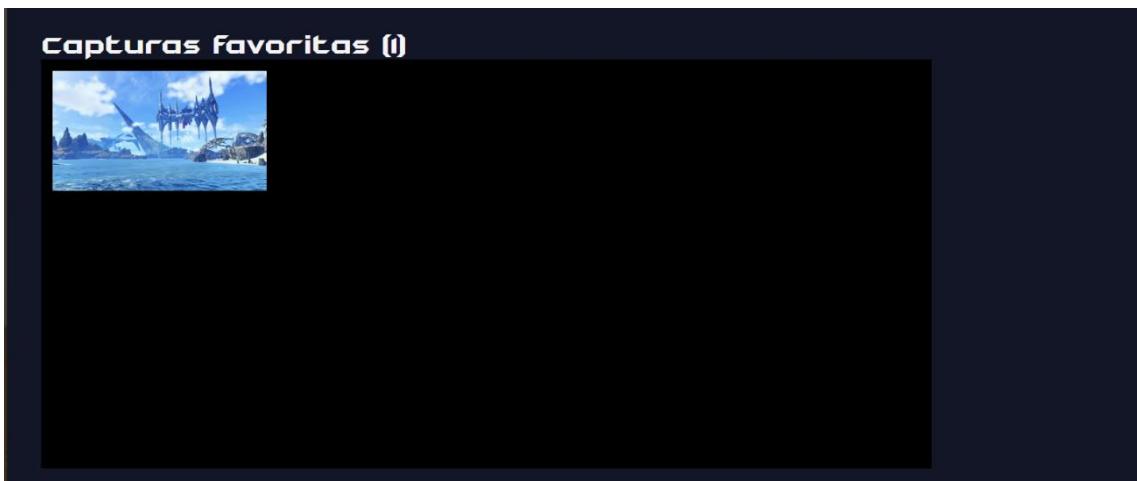
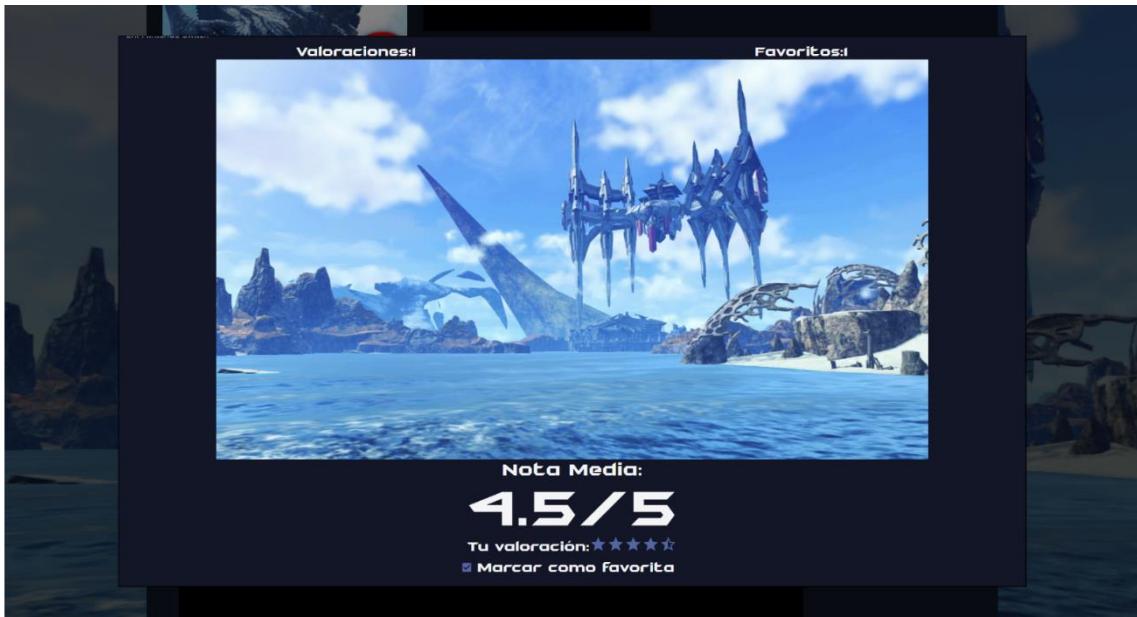
En el formulario se nos pedirán un **nombre**, una **plataforma** (Si solo hay una plataforma disponible se elegirá automáticamente) y la **imagen de la captura**. Una vez hayamos introducido los datos y **pulsemos el botón de "subir captura"**, se nos **devolverá a la página del juego**, y ahora aparecerá la captura que hayamos subido.



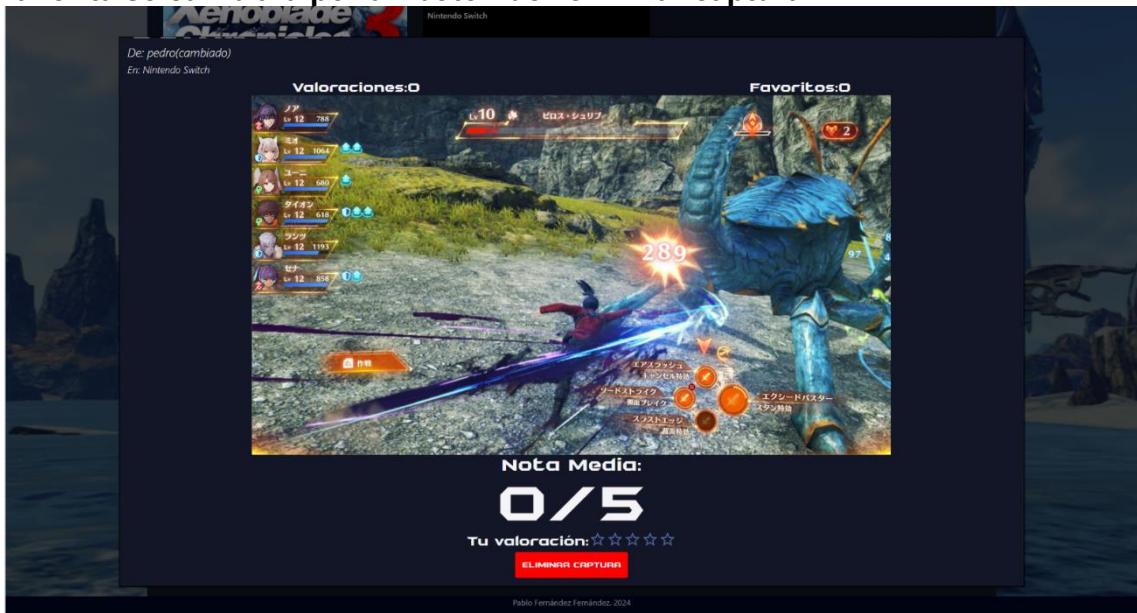
Si pulsamos en cualquiera de las capturas, se abrirá el siguiente modal:



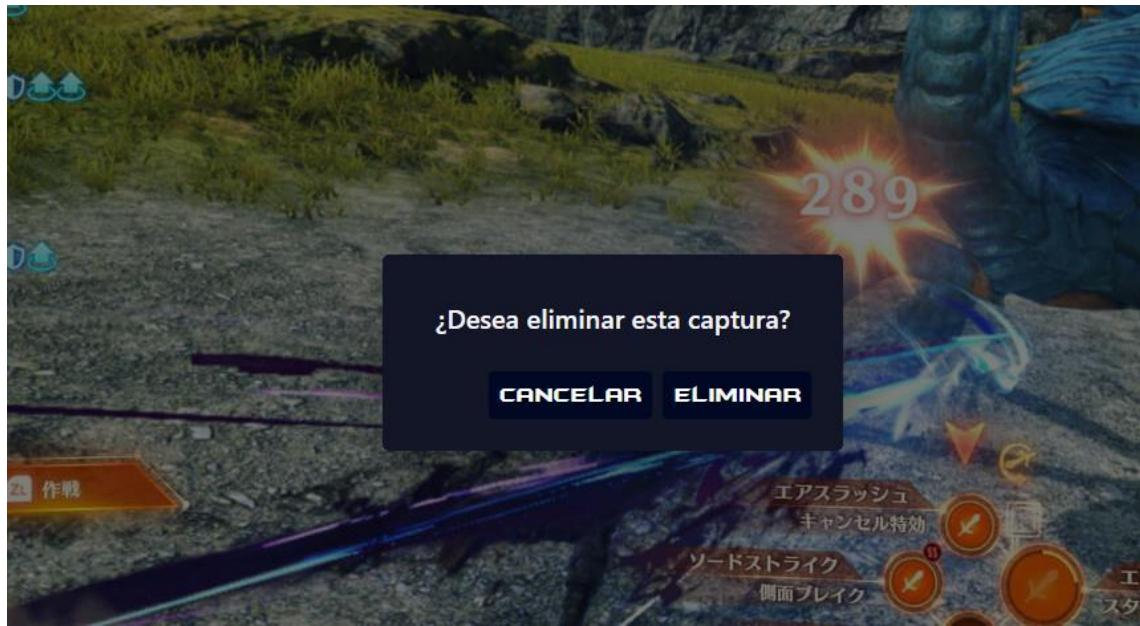
Veremos cinco estrellas y una casilla que indica si la queremos marcar como favorita. Cambiar las estrellas modificará la nota que le deseemos dar a esta captura, y marcarla como favorita hará que aparezca en la sección de "Capturas favoritas" en nuestro perfil.



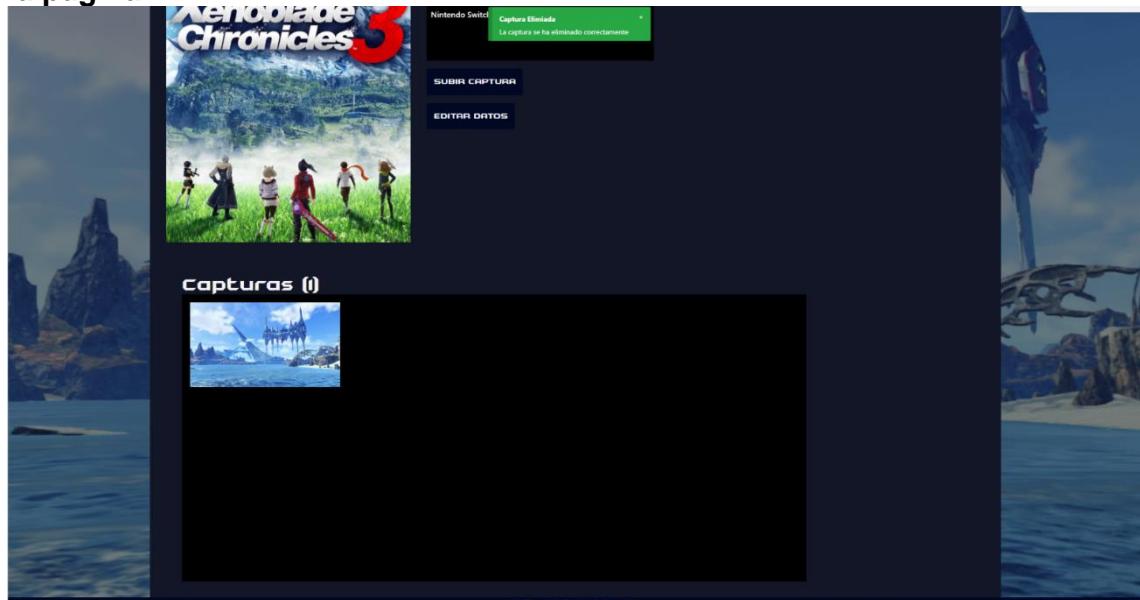
Si abrimos la captura que nosotros hemos subido, la casilla de marcar como favorita se cambiará por un botón de "eliminar captura".



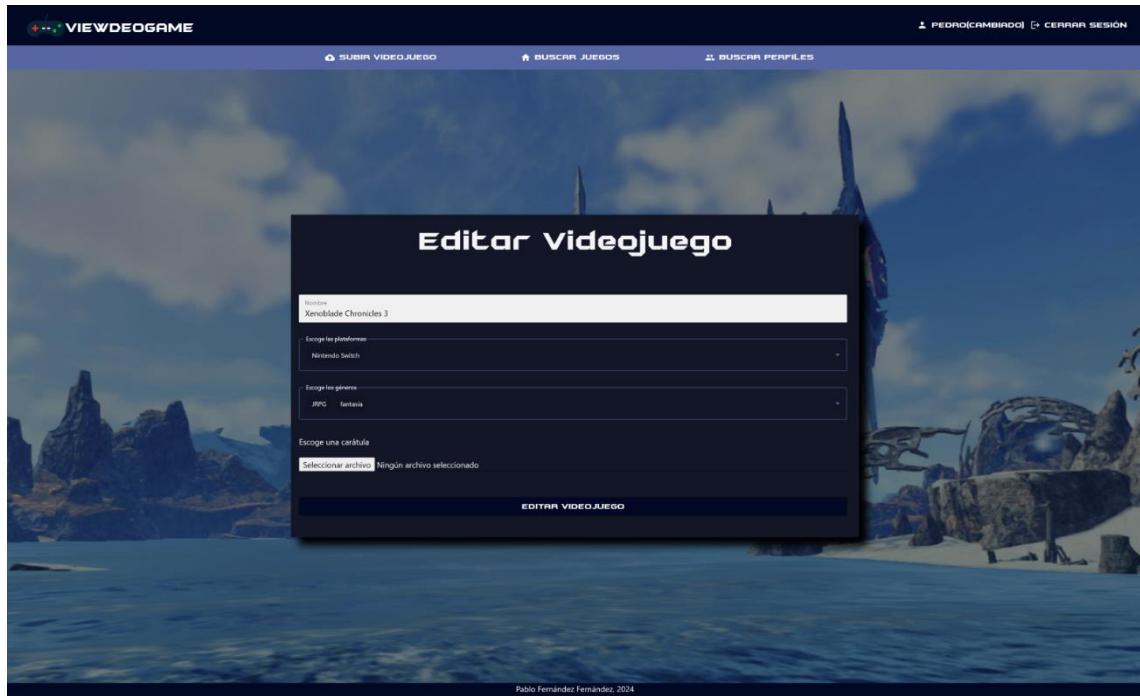
Hacer click en él nos abrirá un diálogo que nos pedirá confirmación:



Pulsar en cancelar cerrará el diálogo, pulsar en eliminar borrará la captura de la página:



Si pulsamos el botón de "editar datos", se nos abrirá el siguiente formulario:



En el siguiente formulario se ve un campo de texto indicando **el nombre**, un select indicando **las plataformas**, otro select indicando **los géneros**, y un input indicando **la carátula**, cada uno rellenado con los datos correspondientes. Si decidimos modificar cualquiera de los datos y pulsamos en **"editar videojuego"**, se nos devolverá a la página del juego con los nuevos datos reflejados:





## Xenoblade 3

### Generos:

JRPG  
fantasía  
Terror

### Plataformas:

Nintendo Switch

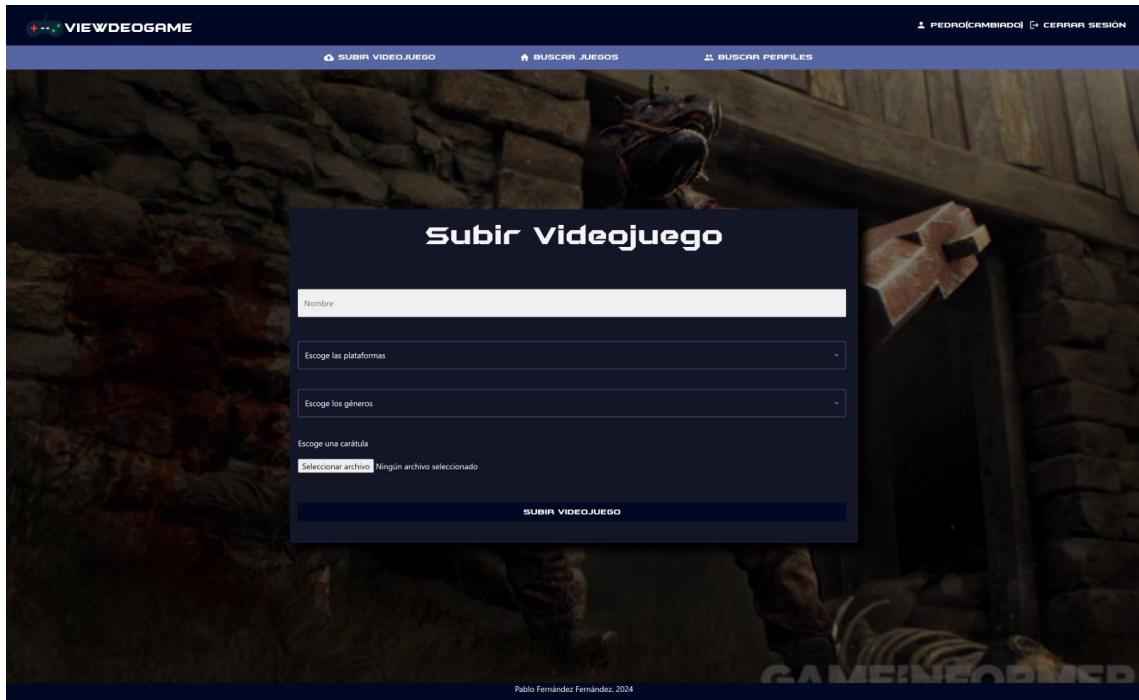
[SUBIR CAPTURA](#)

[EDITAR DATOS](#)

Desde donde estamos, en la barra de acciones, haremos click en "subir videojuego":

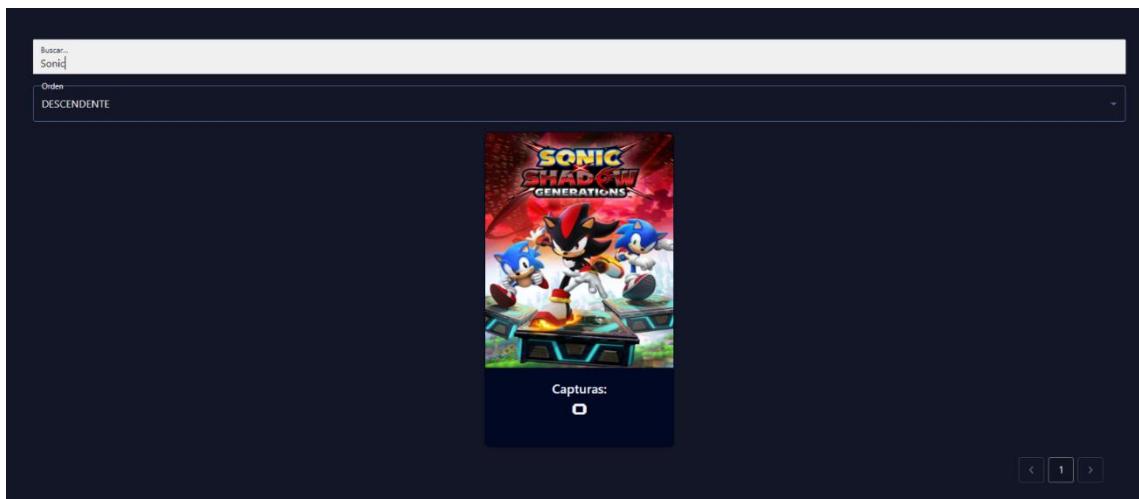
A screenshot of a website interface for submitting video games. At the top, there are three navigation links: 'SUBIR VIDEOJUEGO' (with a camera icon), 'BUSCAR JUEGOS' (with a house icon), and 'BUSCAR PERFILES' (with a user icon). Below this is a search bar with the placeholder 'BÚSQUEDA'. The main content area displays the 'Xenoblade 3' game entry. It includes the game's cover art, its title 'Xenoblade 3', and sections for 'Generos:' (listing JRPG, fantasía, and Terror) and 'Plataformas:' (listing Nintendo Switch). There are also buttons for 'SUBIR CAPTURA' and 'EDITAR DATOS'.

Hacerlo abrirá el siguiente formulario:



**Los campos son los mismos que los del formulario de editar videojuego**, con la diferencia de que esta vez están vacíos, rellenarlos y pulsar en **"subir videojuego"** subirá un nuevo videojuego a la página, y nos devolverá a la página de inicio, donde podremos encontrarlo.

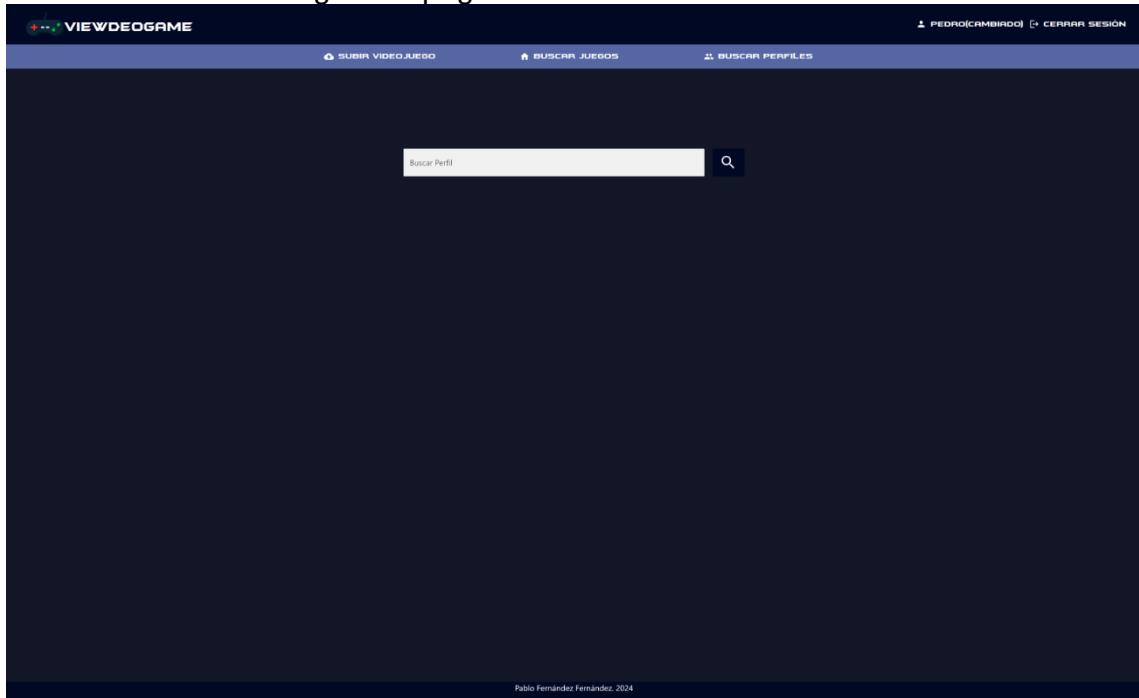




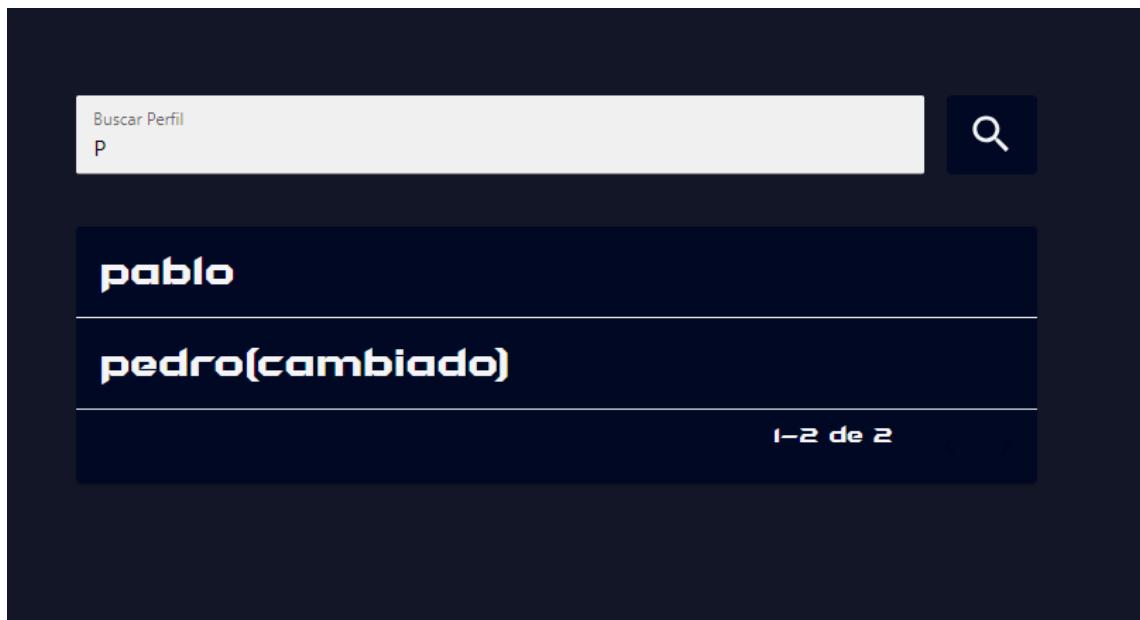
Ahora, en la barra de acciones, haremos click en "buscar perfiles":



Esto nos llevará a la siguiente página:



Si introducimos cualquier valor en la barra de búsqueda y luego hacemos click en el botón o pulsamos enter, nos aparecerán los usuarios cuyo nombre coincide con la palabra que hayamos introducido:



Hacer click en cualquiera de ellos nos llevará a su perfil:

A screenshot of a user profile page for "pablo". The page has a dark theme. At the top, there is a navigation bar with links for "VIEWVIDEOGAME", "SUBIR VIDEOJUEGO", "BUSCAR JUEGOS", and "BUSCAR PERFILES". On the right side of the top bar, there is a user icon labeled "PEDRO(CAMBIADO)" and a "CERRAR SESIÓN" button. The main content area is titled "PABLO". Below the title, there is a section titled "Capturas subidas (7)" which contains a grid of seven screenshot thumbnails. Below this, there is a section titled "Capturas favoritas (0)". At the bottom of the page, there is a footer with the text "Pablo Fernández Fernández, 2024".

## 2.CÓDIGO COMPLETO DEL PROYECTO

- **Frontend:** <https://github.com/daw2-23-24-pdaw/Viewdeogame-Frontend-Pablo-Fernandez>
- **Backend:** <https://github.com/daw2-23-24-pdaw/Viewdeogame-Backend-Pablo-Fernandez>