

Fase-II Proyecto Final Programación en la Web

MyGameLog

Daniel Esteban Rueda Barrera - 2221927

José Augusto Quintero Lobo - 2202348

Daniel Fernando Leal Ayala - 2191430

Kevin Farith García Chaparro - 2204078

Universidad Industrial de Santander

Escuela de Ingeniería en Sistemas e Informática

Ingeniero: Manuel Guillermo Flórez Becerra, Profesor Planta

Programación en la Web

Bucaramanga, Santander, Colombia

27 de noviembre de 2024

Índice

1. Objetivos de la Aplicación web

- Objetivo general
- Objetivos específicos

2. Imagen de la Primera Página del Sitio

- Captura de pantalla de la página de inicio.

3. Estructura Jerárquica del Sitio en el Disco

- Descripción de la estructura de carpetas y archivos del proyecto.
- Explicación de las carpetas public, src, components , pages , entre otras.

4. Usuarios y Credenciales

5. Modelo Relacional de la BD

6. Lista de Funcionalidades Implementadas

- Funcionalidades del AppWeb
- Funcionalidades Implementadas en la fase II
- funcionalidades solicitadas pero no implementadas

7. Conclusiones

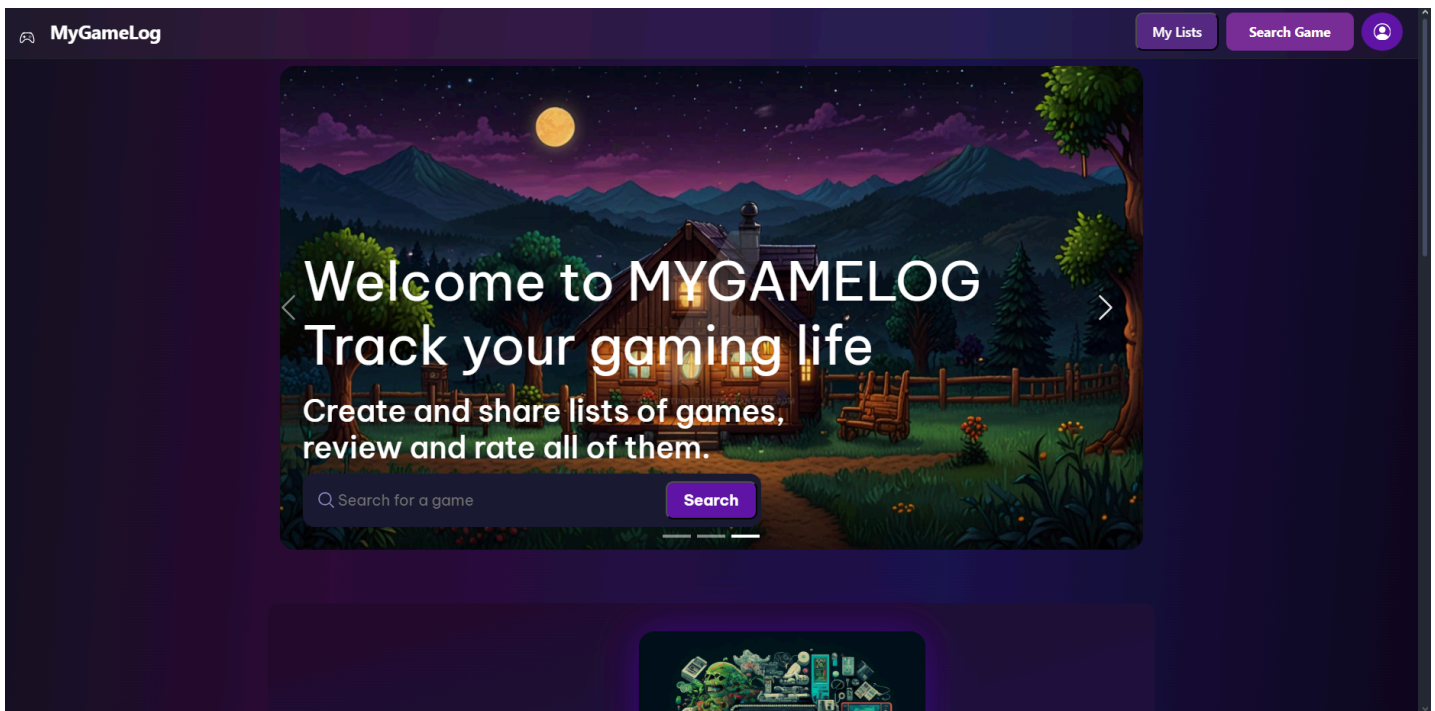
8. Bibliografía

1. Objetivos de la Aplicación web

- **Objetivo General:** Proporcionar a los usuarios una plataforma organizada para registrar y hacer seguimiento de su progreso en videojuegos, ayudando a los usuarios a tener un registro de sus videojuegos jugados, que están jugando y que planean jugar, además de la posibilidad de calificarlos.
- **Objetivos Específicos:**
 - ❖ Crear una interfaz interactiva y amigable usando React.
 - ❖ Implementar un sistema de registro de juegos donde el usuario pueda agregar, editar y eliminar entradas.
 - ❖ Implementar un sistema de navegación que permita a los usuarios desplazarse entre las secciones "Inicio", "Buscar Juegos", "Listas" y "Login", asegurando una experiencia de usuario fluida y accesible.
 - ❖ Facilitar la búsqueda de juegos dentro de la aplicación.
 - ❖ Hacer uso de principios de diseño web como el responsive design para accesibilidad en dispositivos móviles y de escritorio en nuestra aplicación web.

- ❖ Implementar una BD que guarde datos de los usuarios, los juegos y las listas de cada usuario con sus respectivos registros de videojuegos

2. Imagen de la Primera Página del Sitio

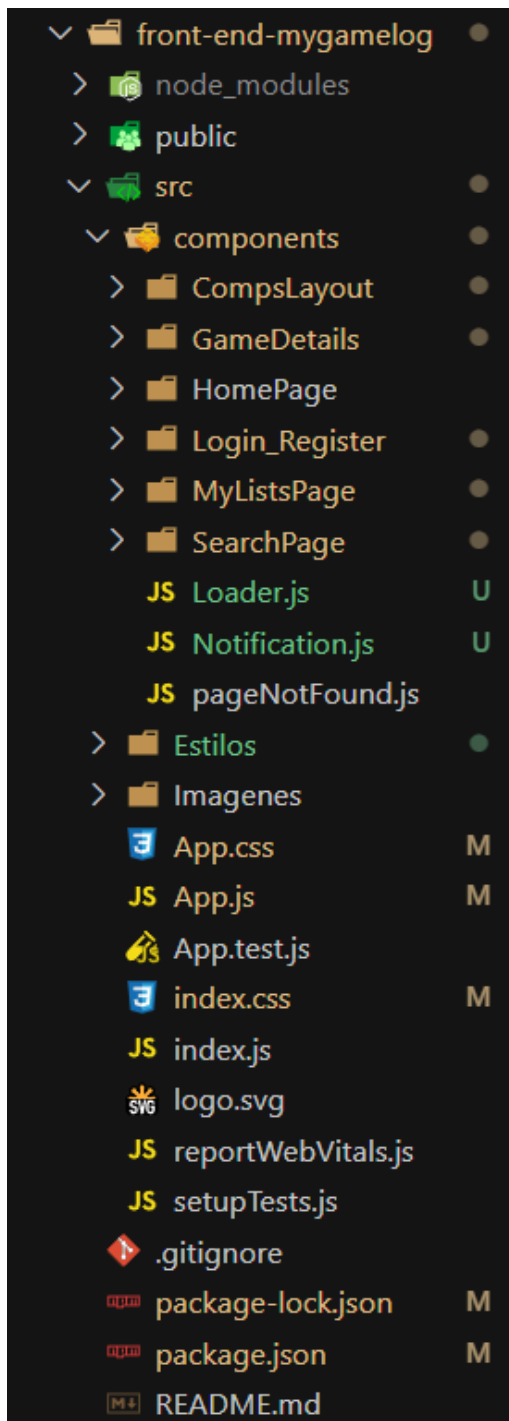


3. Estructura Jerárquica del Sitio en el Disco

Carpetas de front-end y back-end de la aplicación

```
> 📁 back-end-mygamelog  
> 📁 front-end-mygamelog
```

Descripción de la estructura de carpetas y archivos del **Front-End**.



|— **node_modules**

| |— **_Módulos y dependencias de npm_**

|— **public**

| |— **game-favicon.png** - Icono para el navegador

| |— **index.html** - HTML principal del proyecto

| |— **manifest.json** - Archivo de configuración para Progressive Web App (PWA)

| |— **robots.txt** - Indicaciones para bots de motores de búsqueda

|— **src**

| |— **components** - Componentes de React reutilizables divididos en carpetas

| |— **Estilos** - Carpeta para archivos CSS (hojas de estilo)

| |— **App.css** - Estilos principales del componente App

| |— **App.js** - Componente principal de la aplicación

| |— **App.test.js** - Archivo de pruebas para el componente App

| |— **index.css** - Estilos globales del proyecto

| |— **index.js** - Punto de entrada principal de la aplicación React

| |— **logo.svg** - Logo en formato SVG usado en la app

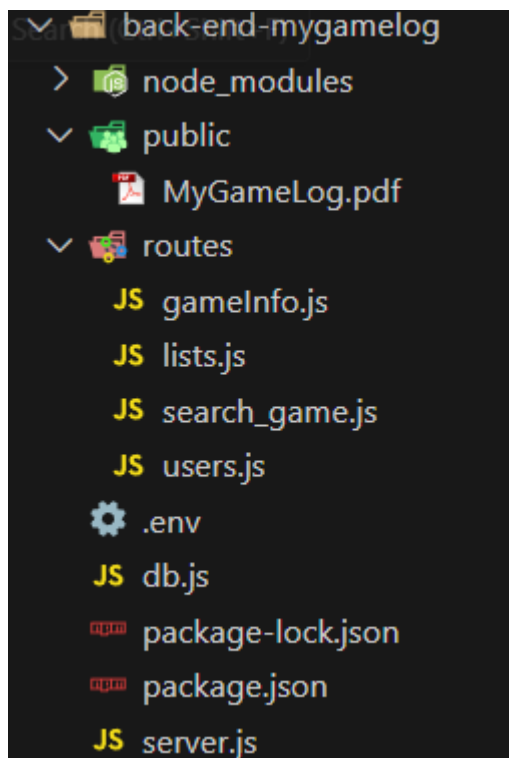
| |— **reportWebVitals.js** - Archivo para medir el rendimiento de la aplicación

| |— **setupTests.js** - Archivo de configuración para pruebas

|— **.gitignore** - Archivo para ignorar archivos y carpetas en Git

- |— **package-lock.json** - Detalles exactos de versiones de dependencias
- |— **package.json** - Configuración y dependencias del proyecto
- |— **README.md** - Documentación básica del proyecto

Descripción de la estructura de carpetas y archivos del **Back-End**



|— **node_modules**

| |— **_Módulos y dependencias de npm_**

|— **public**

| |— **MyGameLog.pdf** - este mismo archivo PDF

|— **routes**

| |— **gameInfo.js** - Ruta para obtener los datos de un juego específico de la BD

| |— **lists.js** - Contiene rutas relacionadas con las listas como: obtener las listas de un usuario específico, añadir un juego a una lista específica, eliminar juego de una lista, editar puntuación de un juego concreto.

| |— **search_game.js** - Ruta para buscar juegos en la barra de búsqueda

| |— **users.js** - Contiene la ruta para registrar a un usuario en BD y la ruta para verificar y hacer el login de este.

|— **.env** - Variable de ambiente usada para crear un Json Web Token JWT al logear a un usuario

|— **db.js** - Realiza la conexión con la BD y define las tablas de la BD como objetos además de sus relaciones para ser usadas en las distintas rutas

|— **package-lock.json** - Detalles exactos de versiones de dependencias

|— **package.json** - Configuración y dependencias del proyecto

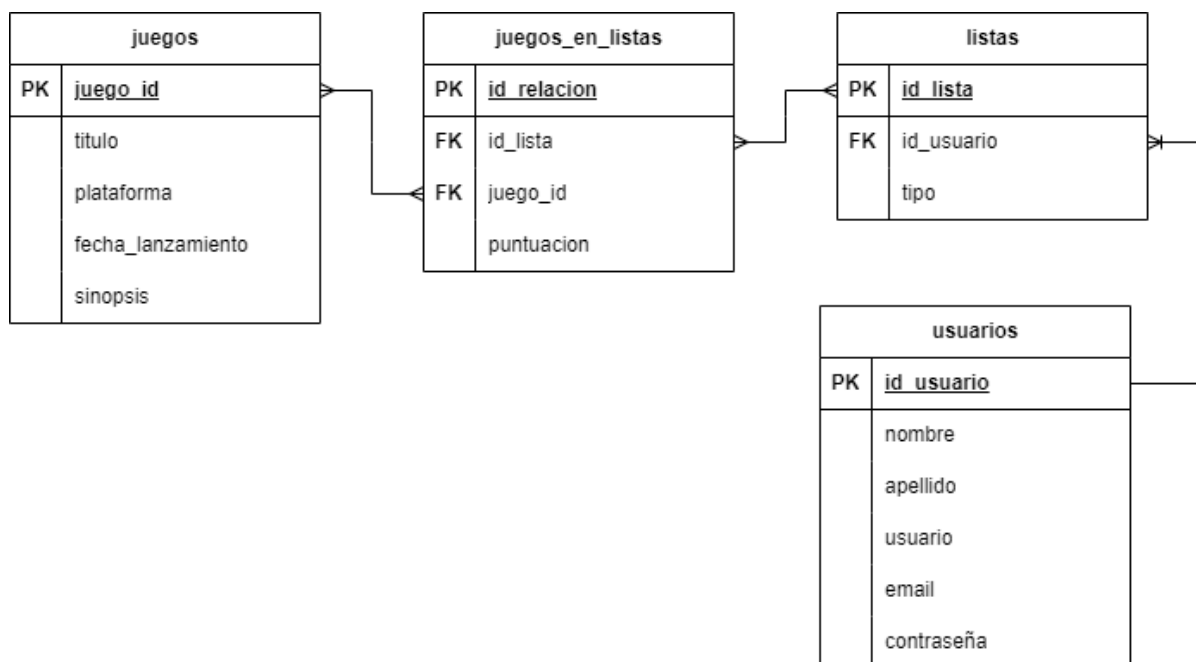
|— **server.js** - Archivo principal que crea el servidor en el puerto 8000, usa las rutas y se sincroniza con la BD.

4. Usuarios y credenciales para acceder al sitio web

MyGameLog solo cuenta con un tipo de usuario, que es el usuario normal que puede utilizar el servicio de guardar juegos en listas, además cuenta con un sistema de registro sencillo y simple para que cualquiera cree una cuenta y se logee para que pruebe todas las funcionalidades. Sin embargo se proporciona un usuario ya creado para que solo tenga que loguearse en el formulario de login para acceder a la página, pero si desea puede crear su propio usuario en el formulario de registro. Este es el usuario proporcionado en formato: usuario | contraseña:

usuariovisitante@gmail.com | 123456789

5. Modelo Relacional de la BD



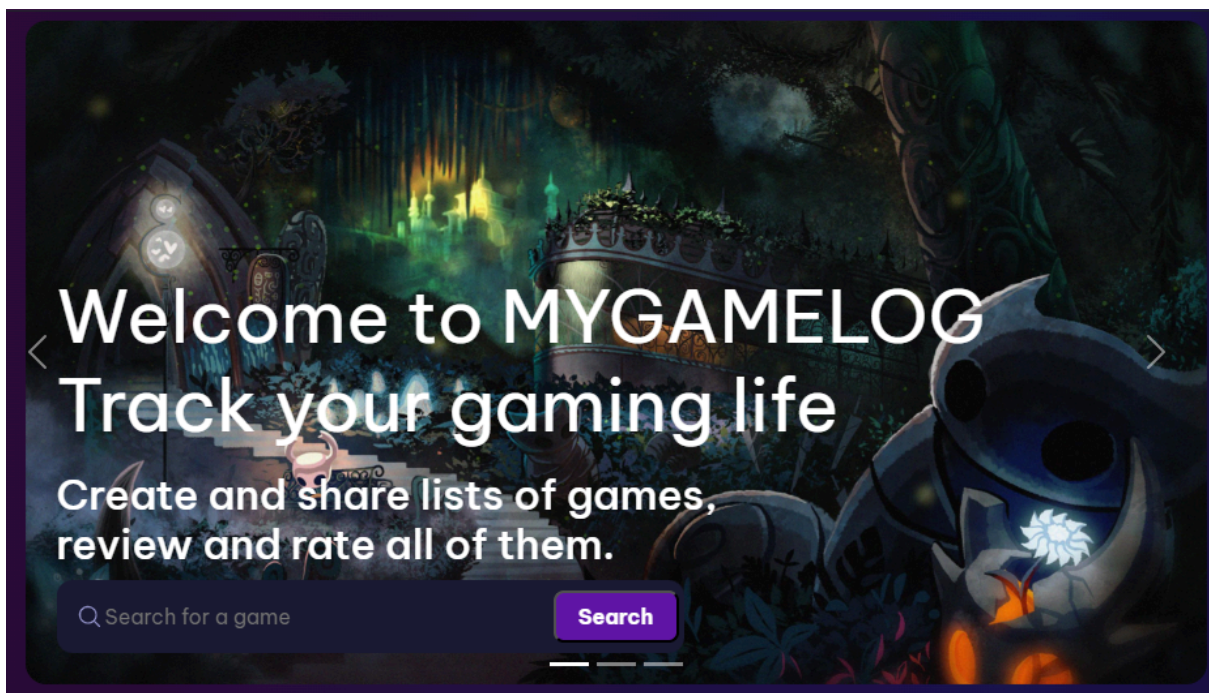
Para encender la base de datos MySQL hecha en XAMPP se debe una vez estando en la máquina virtual proporcionada con el usuario “admin” y contraseña “villabona” abrir XAMPP ejecutándolo como administrador y encender Apache y MySql. Se puede visualizar la BD accediendo desde el navegador en la máquina virtual a localhost y dándole click a

phpmyadmin, ahí se encuentra la BD llamada “proyecto” con las tablas mostradas en el diagrama ya implementadas y pobladas.

6. Explicación de Funcionalidades Implementadas

Funcionalidades implementadas en la fase I:

- **Inicio:** La pantalla de bienvenida en donde se implementó un carrusel con distintas imágenes de temáticas de videojuegos que realizan una transición cada x segundos utilizando el efecto “fade” (también se puede hacer la transición manualmente con las flechas), para darle un dinamismo a la página. Además dentro del mismo carrusel se implementó una barra de búsqueda que debe permitir realizar las consultas de los distintos videojuegos que los usuarios en general han comentado y puntuado.



- **Barra de Menú:** La barra de menú principal permanece fija y tiene los siguientes elementos en forma de botón: MyList , Search Game, icono de cuenta y por último **Mygame log**.

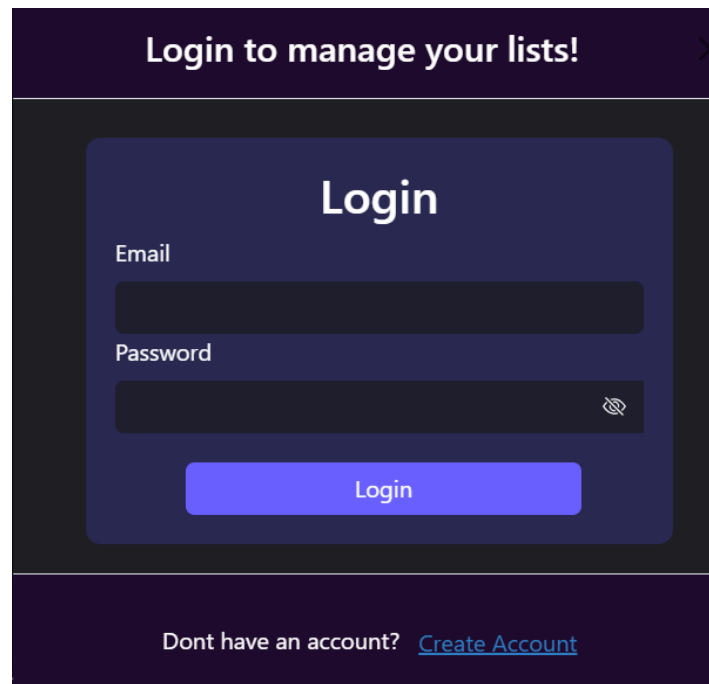


- **My List:** Se incorporó un botón que nos redirecciona a un apartado en forma de lista de los principales videojuegos con las puntuaciones de los usuarios, donde el usuario tendrá la opción de guardar algún videojuego de interés y puntuar para su lista personal, pero tendrá que loguearse para guardarlo en su cuenta (cuando la base de datos esté funcionando se podrán visualizar la lista como tal, ya que necesitamos los elementos).
- **Search Game:** Hace parte del menú principal de la página y su función es desplegar una barra de búsqueda en la parte superior de la página de forma centrada y con esto el usuario podrá realizar consultas de los distintos videojuegos de interés concretamente. Reemplaza el carrusel cuando se le hace click navegando a otra ruta. Como ejemplo en lo que se implementa la BD de videojuegos, se puede probar buscando “Pokemon” o “Fifa” y viendo los resultados.



- **Login:** Se implementó un botón en el menú principal con el típico icono de cuenta, donde al hacer click aparece una ventana emergente sobre el contenido de la página principal (se utilizó un modal) del login con username y password , además de tener en el mismo apartado una opción de crear

cuenta que al hacer click reemplaza el login por Register y ahí el usuario creará la respectiva cuenta de la forma más sencilla posible. Por último dicho register se incorporó un link de vuelta al login por si el usuario se equivocó y pueda devolverse rápidamente. Se hizo el uso de rutas privadas y públicas tanto para antes del login como para el MyList que es una función que se puede realizar después del login.



The image shows a login form titled "Login to manage your lists!". The form is centered on a dark background. It has a title "Login" in white. Below the title are two input fields: "Email" and "Password". The "Email" field is a simple text input. The "Password" field is a text input with a toggle icon (an eye) to its right. Below the input fields is a blue button labeled "Login". At the bottom of the form, there is a link that says "Dont have an account? [Create Account](#)".

Register on our website!

Register

First Name

Last Name

Username

Email

Password


Register

Already have an account?


[Login Here!](#)

- **MyGameLog:** El logo de la página junto con el nombre se incorporó donde si le hacemos click nos devolverá a la página principal donde está el carrusel, por ejemplo: si estamos en search Game o My Lists podemos devolvernoss a la página de inicio o principal con solo clicar el Mygame log.
- **Pie de página:** Se añadieron los principales medios de contactos como facebook, Twitter X , Reddit y discord. También se incorporó el contáctenos de los miembros del equipo, teniendo en cuenta los datos básicos y las fotos.


Our Team




Daniel Rueda
Developer
danielsteban1006@gmail.com



Jose Quintero
Developer
josequintero020730@gmail.com





Daniel Leal
Developer
daniellealaya102@gmail.com





Kevin García
Developer
kefagacha@gmail.com

Social Links

Facebook 

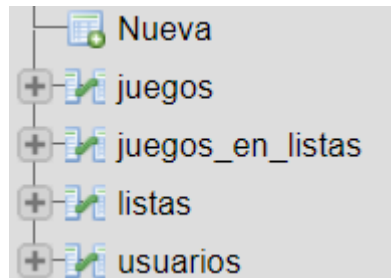
Twitter 

Reddit 

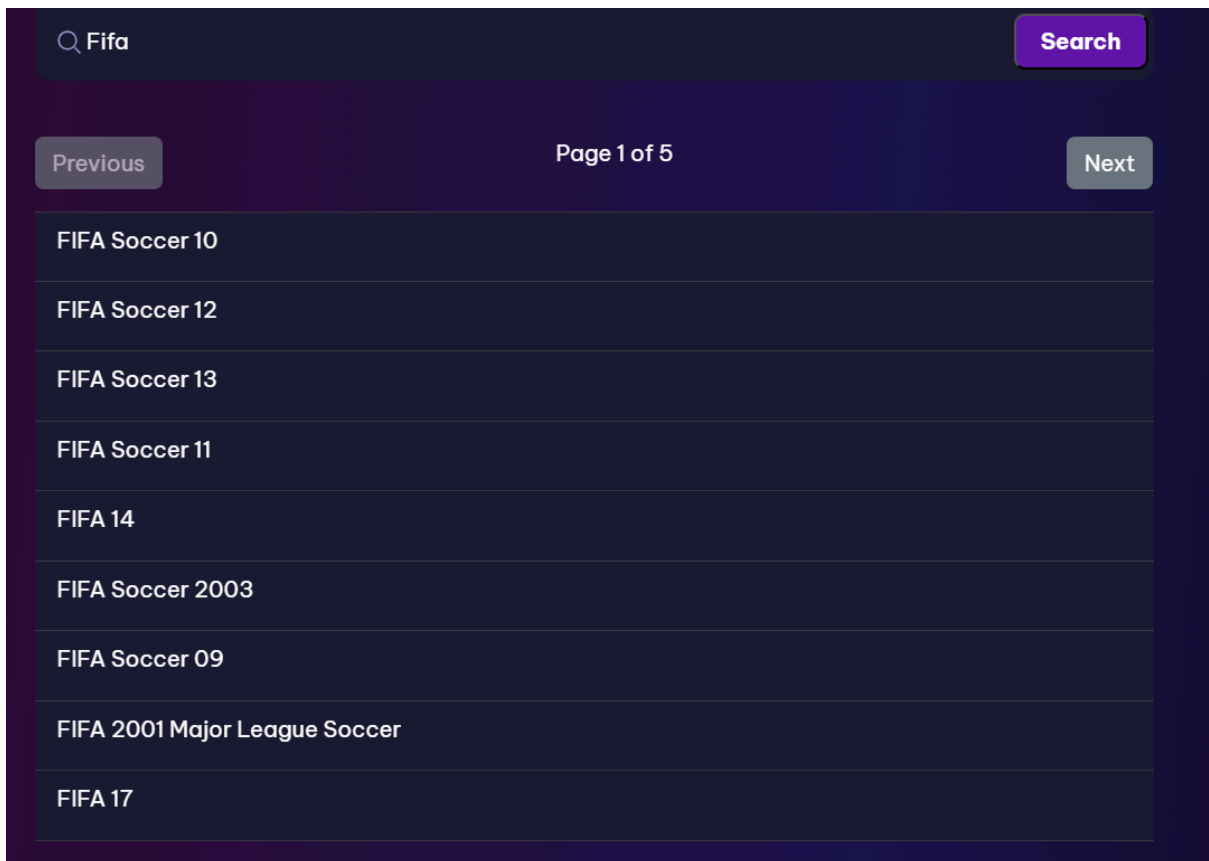
Discord 

Nuevas Funcionalidades implementadas en la Fase II

- **Implementación de la BD:** Se construyó la Base de Datos siguiendo el modelo lógico anterior usando MySQL en XAMPP, haciendo la respectiva conexión con el backend



- **Implementación del servidor Back-End:** Usando node y express.js se creó el servidor en el puerto 8000 junto con las rutas y conexiones a la bd necesarias
- **Búsqueda de juegos guardados en la BD:** Ahora con la BD implementada, es posible buscar más de 12000 juegos en la barra de búsqueda implementada en la fase I los cuales se muestran a modo de lista paginada de manera inteligente según la cantidad de resultados, además cada elemento de la lista al darle click redirige al usuario a una subpágina sobre el juego clickeado.



- **Sección de Información de Cada Juego y Agregación a Lista:** Al darle click al nombre de un juego que sale como resultado de la búsqueda mostrada en el apartado anterior redirige al usuario a una página única para cada juego en el que se muestra la información más relevante sobre el juego clickeado, como su título, año de lanzamiento, plataforma en la que está disponible y su sinopsis. Además se muestran 3 botones para agregar ese juego a una de las 3 listas que tiene cada usuario, lo cual solo funcionará si hay un usuario logeado.

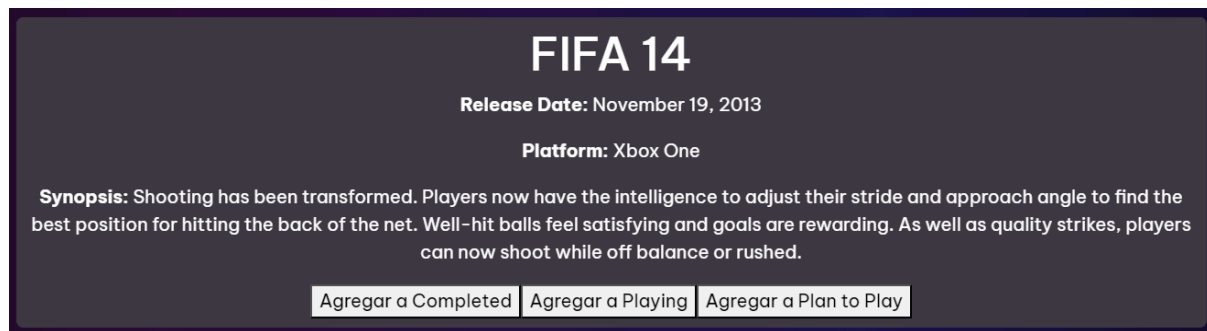
Pokemon Y

Release Date: October 12, 2013

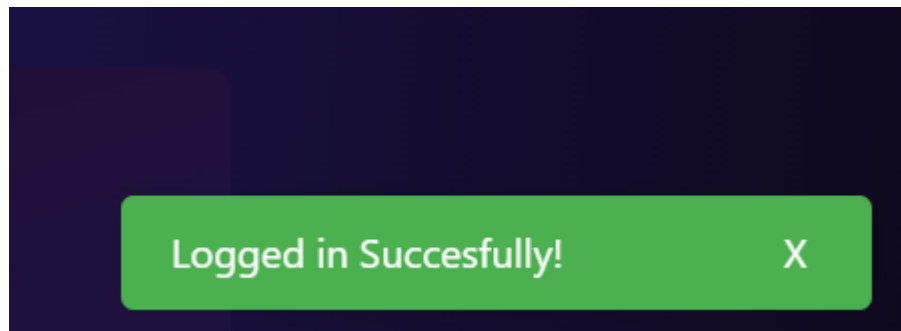
Platform: 3DS

Synopsis: Pokemon X and Y are the first mainline Pokemon games made for the 3DS and feature a mix of Pokemon old and new as well as a trio of new starting Pokemon - a watery toad Froakie, a fiery deer Fennekin, and a grass monster Chespin.








[Agregar a Completed](#) [Agregar a Playing](#) [Agregar a Plan to Play](#)



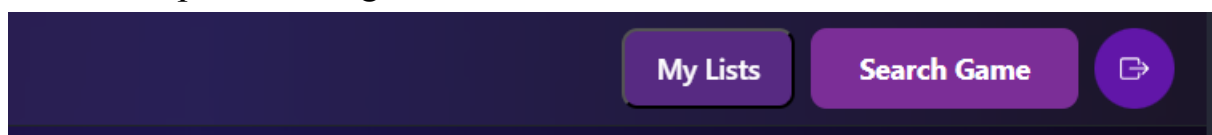
- **Función de Login y Registro completada:** ya con la BD es posible tener un sistema de login funcional, permitiendo a los usuarios acceder a secciones como My Lists únicamente si están logueados, mostrando una notificación en pantalla cuando se loguea con éxito.



- **Función de gestión de listas de cada usuario:** cada usuario, al loguearse, podrá acceder a la sección My Lists, donde verá las 3 listas con sus respectivos juegos que el usuario haya agregado, al agregar un juego se le puede asignar una puntuación a cada juego del 0 al 10, la cual se guarda automáticamente en BD al solar el focus del campo de texto y también se podrá eliminar un juego de la lista en la que se encuentra dando click al icono de basura junto a cada juego en las listas.

viven06's Lists		
Completed	Playing	Plan to Play
<div>  The Elder Scrolls V: Skyrim 7,00 </div> <div>  Mario & Luigi: Dream Team 7,50 </div> <div>  Fire Emblem: Three Houses 10,00 </div> <div>  Persona 5 Royal 8,60 </div>	<div>  Pokemon Black Version 8,00 </div> <div>  Donkey Kong 64 7,00 </div>	<div>  Portal 8,00 </div>

- **Botón de cerrar sesión:** una vez el usuario se registra y se loguea, el botón con icono de persona en la parte superior derecha de la barra de búsqueda será reemplazado por un botón con icono de salida, el cual cierra la sesión del usuario eliminando el JWT y volviendo al estado inicial de la página previo al login.



- **Icono de carga al recargar página:** al recargar la página se mostrará un loader o círculo dando vueltas indicando que la página está cargando.



En cuanto a las hojas de estilo y los script de JS cada archivo está de forma independiente y se tuvo en cuenta una buena organización de los mismos con los respectivos comentarios.

- **Ver el PDF de documentación desde el servidor:** se puede accediendo a localhost:8000/pdf ver este mismo archivo pdf ya que está alojado en la carpeta public del Back-End

Funcionalidades no implementadas:

Las siguientes funcionalidades no pudieron ser implementadas por falta de tiempo en el desarrollo del proyecto

- **Captcha:** Se planeaba implementar un captcha para evitar ingreso de bots a la página en el formulario de registro y login.
- **Función para recuperar contraseña:** se planeaba implementar una función de “olvide mi contraseña” la cual mandaba al email registrado un código el cual el usuario debía escribir para recuperar su contraseña.

7. Conclusiones

1. Se logró conseguir el objetivo general de crear una plataforma para registrar, seguir y evaluar el progreso en videojuegos. Con una estructura sólida de navegación y una interfaz intuitiva desarrollada en React, se ha facilitado una experiencia de usuario amigable y organizada.
2. El diseño responsive garantiza accesibilidad desde cualquier dispositivo, maximizando la usabilidad y adaptabilidad del sitio. La organización del contenido en rutas públicas y privadas contribuye a una experiencia de usuario lógica y fluida, y la inclusión de una página de error 404 mejora la navegación, evitando confusión.

3. La estructura de carpetas y archivos organizada y bien documentada, división entre front y back-end, incluyendo componentes, estilos y scripts separados, permite una mayor escalabilidad y mantenimiento del proyecto. Esto también facilita la colaboración y revisión del código por parte de terceros.
4. Las funcionalidades clave fueron implementadas, como el carrusel, el login modal, y la barra de búsqueda, el sistema de login y registro y manejo de sesiones, la búsqueda de juegos y la gestión de listas, sin embargo hubieron pequeñas funcionalidades que no se pudo implementar por cuestiones de tiempo, lo cual representa una mejora para futuras versiones del proyecto.
5. Al aplicar principios como usabilidad y accesibilidad, el proyecto MyGameLog refleja un enfoque centrado en el usuario, permitiendo una experiencia de navegación coherente y personalizada, especialmente con opciones de login que segmentan contenido relevante para usuarios registrados y visitantes.
6. La implementación del servidor, la Base de Datos y la conexión entre estos y el Front-end fue hecha de manera satisfactoria mostrando un comportamiento y funcionamiento adecuado.

8. Bibliografía

JavaScript. (s/f). MDN Web Docs. Recuperado el 27 de octubre de 2024, de <https://developer.mozilla.org/es/docs/Web/JavaScript>

React. (s/f-a). React.dev. Recuperado el 27 de octubre de 2024, de <https://react.dev/>

React. (s/f-b). Npm. Recuperado el 27 de octubre de 2024, de <https://www.npmjs.com/package/react>

React bootstrap. (s/f). Github.io. Recuperado el 27 de octubre de 2024, de <https://react-bootstrap.github.io/>

W3Schools online web tutorials. (s/f). W3schools.com. Recuperado el 27 de octubre de 2024, de <https://www.w3schools.com/>

Nielsen Norman Group. (2020). Usability 101: Introduction to Usability. Nielsen Norman Group. Recuperado el 27 de octubre de 2024, de <https://www.nngroup.com/articles/usability-101-introduction-to-usability/>

Mozilla Developer Network (MDN). (2023). Accessibility - MDN Web Docs. Mozilla Foundation. Recuperado el 27 de octubre de 2024, de <https://developer.mozilla.org/en-US/docs/Learn/Accessibility>

Google Developers. (2022). Web Fundamentals: Responsive Design. Google. Recuperado el 27 de octubre de 2024, de <https://developers.google.com/web/fundamentals/design-and-ux/responsive/f>