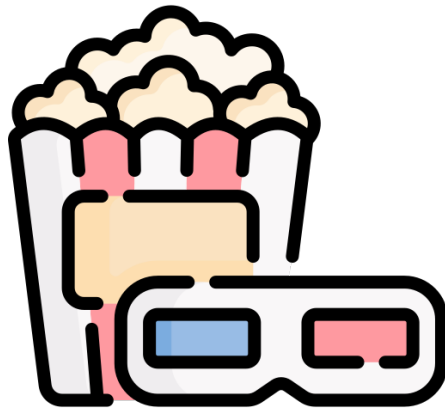


Trabajo TFG



REPLAY

**Aplicación web para la visualización
y búsqueda de información sobre
Películas y Series**

Autor: Imanol Seara Montes

Tutor: Sandra Pereira Álvarez

Fecha: 12/2023

Github: <https://github.com/7ONAMI/replay-fct>

Web: <https://replay-fct.000webhostapp.com>

Indice:

Contenido

Objetivo de Proyecto.....	3
Estudio de mercado.....	4
Presupuesto.....	5
Planificación	6
Documentación técnica.....	7
Herramientas empleadas	9
Problemas Encontrados.....	11
Propuestas de mejoras	12
Manual Tecnico de Instalación	13
Instalación en Server	13
Manual de usuario	14
Pruebas Realizadas.....	16
Conclusión	20

Objetivo de Proyecto

Dada la cantidad de contenido audiovisual del que disponemos hoy en día, gracias a servicios como Netflix, Prime Video, HBO Max etc. Mas que nunca se necesita una fuente de información de este contenido. El objetivo de este proyecto es crear una página web que liste la información de películas y series usando la API de TheMovieDB. Tendrá tres tipos de listas, las mejor valoradas, las mas populares y los últimos estrenos. Tambien podrá realizar búsquedas. La página debe de ser responsive y estar en español.

Objetivos específicos:

- Crear una web utilizando la API themovieDB.
- Proporcionar una interfaz web fácil de usar y accesible.
- Ofrecer información completa y actualizada sobre las series y películas.
- Separar en tres tipos de listas las películas y series.
- Permitir a los usuarios realizar búsquedas.
- Visionar todos los detalles de cada película o serie.
- Poder compartir la URL de la película o de una búsqueda.

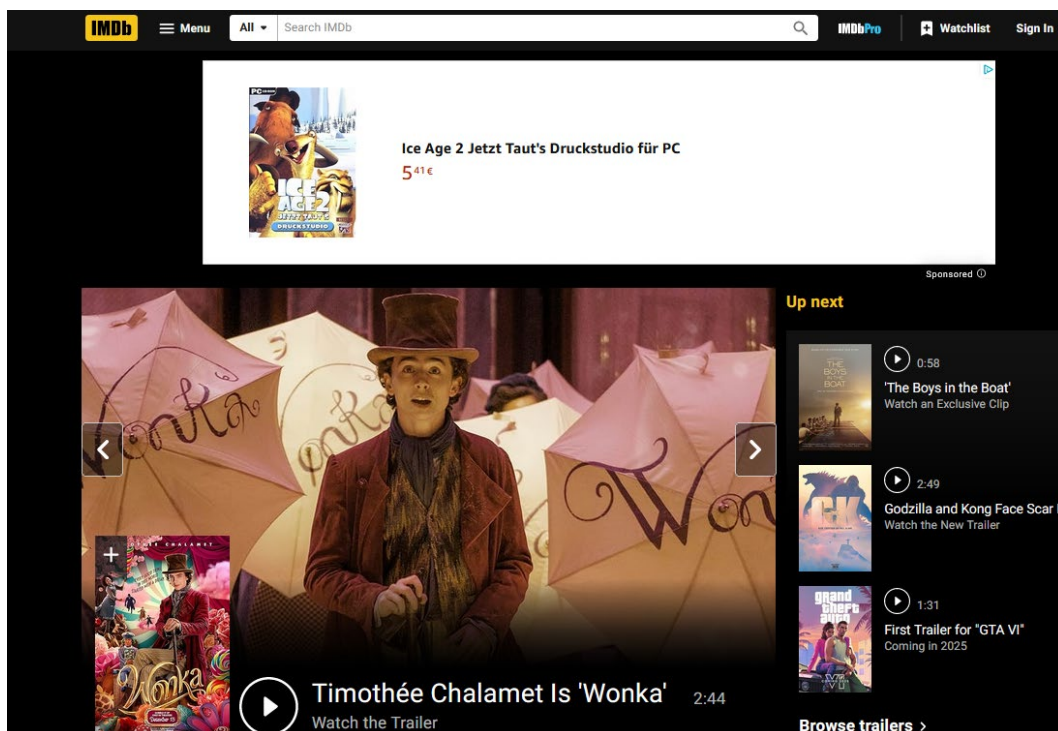
Estudio de mercado

El mercado de las páginas web de películas y series es muy competitivo. Existen muchas webs que ofrecen este servicio como IMDb o JustWatch.



Sin embargo, estas páginas web suelen estar en inglés y no siempre son fáciles de usar. La página web que se propone en este proyecto tienen como objetivo cubrir una necesidad que existe en el mercado:

Una web de consulta de información de películas y series en español que sea fácil de usar y ofrezca una buena experiencia de usuario.



Presupuesto

El presupuesto de este proyecto es de 2515 euros. Este presupuesto se destinará a la compra de un dominio y hosting, la así como a la contratación de servicios profesionales y una futura traducción a más idiomas.

El presupuesto está echo en base al mantenimiento de un año de dicha web, tanto de recursos como de asistencia.

Categoría	Descripción	Presupuesto
Desarrollo web	Desarrollo de front-end y del back-end	1000 €
API	Contratación de API por 1 año	800 €
Hosting	Contratación por 1 año	200 €
Dominio	Contratación por 1 año	15 €
Otros costes	Diseño, traducción y otros recursos	500€
Total:		2515 €

Planificación

La planificación de este proyecto se divide en tres fases diferentes.

El proyecto se planificará en tres fases:

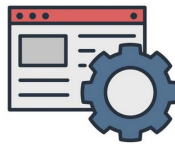
- **Fase 1:** Diseño y desarrollo



- En esta fase se Diseñará la web usando Figma como herramienta y también se estudia el funcionamiento de la API. A continuación se desarrolla el proyecto utilizando Vite para montarlo y React para la interfaz junto con JavaScript.

Duración: 15 horas

- **Fase 2:** Pruebas y depuración



- Se realizarán pruebas manuales tras lo cual se inicia la depuración del código, luego se realizaran las pruebas automatizadas y otra vez depuración de código.

Duración: 20 horas

- **Fase 3:** Lanzamiento de la página web



- Búsqueda de un hosting adecuado, contratación del Dominio y subida del proyecto garantizando el debido funcionamiento de este.

Duración; 5 horas

Documentación técnica

La API de **TMDb** es una **API RESTful**, que es una interfaz de comunicación entre sistemas que utiliza el protocolo de transferencia de hipertexto (HTTP) para obtener datos. Es gratuita y proporciona acceso a una gran cantidad de datos sobre películas, series, y otros aspectos de la industria del entretenimiento. La API siempre devuelve la información en un objeto JSON.

Para utilizar la API TMDB, primero debes registrarte para obtener una clave de API. Una vez tengas la clave puedes comenzar a realizar solicitudes. Todas ellas se realizan mediante el protocolo HTTP.

Las operaciones de la API de TMDB se dividen en 3 categorías:

Operaciones de Recursos: Estas operaciones permiten acceder a datos sobre recursos específicos como películas o series. Ej:

<https://api.themoviedb.org/3/movie/157336>

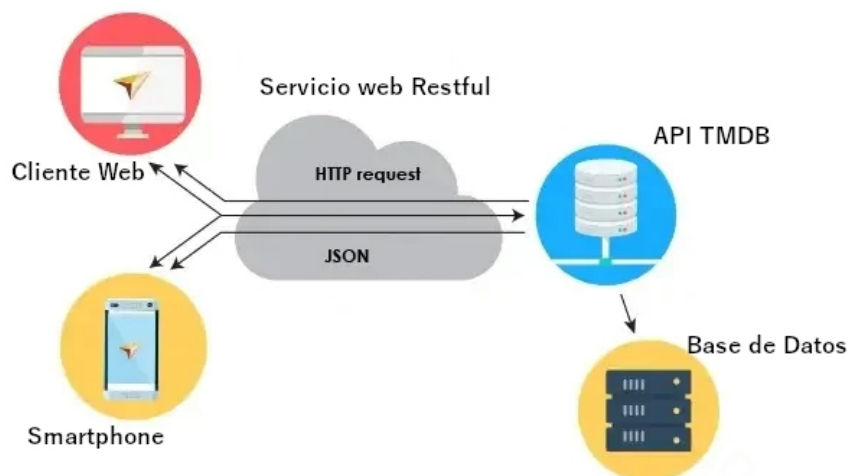
Operaciones de Colecciones: Estas operaciones permiten acceder a colecciones de recursos. Ej:

<https://api.themoviedb.org/3/movie/popular>

Operaciones de Búsqueda: Estas operaciones permiten realizar búsquedas de recursos. Ej:

<https://api.themoviedb.org/3/search/multi?query=batman>

Arquitectura de Sistema:



Diseño conceptual de funcionamiento:

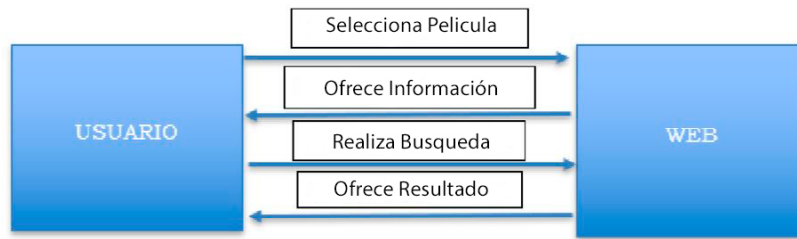


Diagrama de clases. Este diagrama muestra las clases y objetos que componen la API y las relaciones entre ellos.

El diagrama muestra las siguientes clases principales:

- **Movie:** Representa una película. Tiene propiedades para el título, la fecha de lanzamiento, el género, la calificación y otras información.
- **TVShow:** Representa un programa de televisión. Tiene propiedades similares a las de Movie, además de propiedades para las temporadas y los episodios.
- **Person:** Representa una persona relacionada con el entretenimiento. Tiene propiedades para el nombre, la fecha de nacimiento, la ocupación y otras información.

```
class Movie {
  id: integer
  title: string
  year: integer
  runtime: integer
  rating: number;
  posterPath: string;
  overview: string;
  genres: List<Genre>
  actors: List<Actor>
}

class Genre {
  id: integer
  name: string
}

class Actor {
  - id: integer
  - name: string
  - birthdate: Date
  - known_for: List<Movie>
}

class TVShow {
  title: string;
  releaseDate: string;
  genres: List<Genre>
  rating: number;
  runtime: number;
  posterPath: string;
  overview: string;
  actors: List<Actor>
  seasons: Array<Season>;
}

class Season {
  number: number;
  episodes: Array<Episode>;
}

class Episode {
  title: string;
  airDate: string;
  runtime: number;
  posterPath: string;
  overview: string;
}
```


Herramientas empleadas

Las herramientas empleadas en este proyecto son las siguientes:

- **Git y GitHub:** Para el control de versiones.

Git es un sistema de control de versiones distribuido que permite a los desarrolladores colaborar en proyectos de forma eficiente. GitHub es una plataforma de alojamiento de código fuente que facilita el uso de Git.



- **React:** Para el desarrollo de la interfaz de usuario.

Es una biblioteca de JavaScript para crear interfaces de usuario. Es muy popular y permite reutilizar código de forma eficiente.



- **JavaScript:** Para la programación de la lógica de la página web.

Es un lenguaje de programación interpretado que se utiliza para crear contenido dinámico en webs.



- **HTML5:** Para el marcado de la página web.



- **CSS:** Para el estilo de la página web.



- **Vite:** Para la creación del proyecto y la compilación de la página web.

Es un framework web que permite crear aplicaciones web de forma rápida y sencilla. Utiliza Webpack, para compilar y servir aplicaciones web de forma rápida y eficiente.



Problemas Encontrados

La mayor parte de los problemas encontrados son con el uso de React y la API TheMovieDB.

Problema con el enrutamiento para la vista detalles de las películas y series.

El ID de películas y series se solapan, esto quiere decir que repiten numero, no van juntos, para que al copiar un enlace y pegarlo sirva el mismo resultado, no podría ser `"/detalles/id-pelicula-o-serie"` pues no se sabría si es serie o película.

Solución: Pasar por props tanto el tipo de servicio audiovisual como el id.

`"/detalles/:tipo/:id"`  `"/detalles/movie/23132"`

Problema con la búsqueda de Peliculas y series

Tras una búsqueda si se resetea o se carga otra sección, debe limpiarse el "search" pero la información es escasa de como resetearlo desde otro componente.

Solución: utilizando `useSearchParams` para crear la query y la función `.delete` para resetearlo.

Problema con el JSON de búsqueda

El JSON con el resultado de la búsqueda de películas y series vienen con un array de 20 objetos. Si hay mas resultados el JSON tiene una propiedad de página.

Solución: Utilizar una dependencia para la carga automática de mas objetos al hacer scrool. Con un spinner de carga. La dependencia es `InfiniteScroll`

Propuestas de mejoras

En el futuro, se podrían realizar las siguientes mejoras en la página web:

- Añadir más funciones, como la posibilidad de ver tráilers o leer críticas de las películas y series.
- Mejorar la experiencia de usuario, por ejemplo, añadiendo más filtros para buscar películas y series, como filtro por géneros.
- Añadir un sistema de recomendaciones en base a una película para que los usuarios puedan descubrir nuevas películas y series que les pueda gustar.
- Añadir un login y que los usuarios puedan guardar sus películas favoritas y las que ya hayan visto.
- Añadir un sistema de comentarios para que los usuarios puedan compartir sus opiniones sobre las películas y series.
- Añadir mas información sobre el Casting, como películas en las que trabajó, una imagen...
- Traducir la página web a más idiomas.

Manual Tecnico de Instalación

Previamente debe tener instalado node y npm (u otro gestor de paquetes). Para la instalación del proyecto, desde **GitHub** se debe clonar el repositorio con **GIT**.

En una terminal navega a la carpeta donde quieras clonar el repositorio y ejecuta:

```
Git clone https://github.com/7ONAMI/replay-fct
```

Luego Debes instalar las dependencias utilizando el comando:

```
npm install
```

Una vez instaladas las dependencias inicializa el proyecto con el comando:

```
npm run dev
```

Instalación en Server

Para instalar el proyecto en un servidor, primero debemos compilarlo. Para ello ejecutamos el comando:

```
npm run build
```

Este comando compila el proyecto y lo guarda en una carpeta llamada **dist**

Luego copiamos la carpeta en nuestro servidor y listo.

Manual de usuario



Dispone de una barra de navegación (1) para seleccionar los diferentes tipos de listas (populares, mejor valoradas y estrenos). Debajo se encuentra la barra de búsqueda (2) para series y películas. Dos botones (3) para seleccionar entre ver la lista de películas o series. Si haces click sobre la tarjeta de la película (4) te llevará a la lista de detalles. Donde podrá ver toda la información sobre dicha película o serie.111

Vista Detalles



Trolls 3: Todos juntos

Estreno 2023-10-12  7.2 Duración 1h 32m

Generos Animación, Familia, Música, Fantasía, Comedia

Nuevos trolls se unen al espectáculo.

Cuando Floyd, uno de los hermanos de Branch, es secuestrado por una pareja de maléficas estrellas del pop para aprovecharse de su talento musical, Branch y Poppy tendrán que embarcarse en un viaje tan aterrador como emotivo para reunir a la familia y rescatar a Floyd de un destino aún peor que el desolador olvido de los fans.

Director Walt Dohrn

Main Cast Anna Kendrick, Justin Timberlake, Camila Cabello, Eric André, Amy Schumer, Andrew Rannells, Daveed Diggs, Troye Sivan, Kid Cudi, Zosia Mamet, Zoëy Deschanel

Pruebas Realizadas

Pruebas de caja blanca:

Hacen referencia a aquellas que se centran en los detalles procedimentales del software, es decir, el flujo de ejecución del código.

No se detallan estas pruebas ya que se realizaron según se desarrollo el proyecto y no son significativas.

Pruebas de caja negra:

Son Aquellas que se encargan de validar los datos de entrada y la salida generada. A continuación, se detallan los casos de prueba realizados:

Prueba-1	Cambiar Lista top
Descripción	El usuario cambia el tipo de lista en el navbar
Salida Esperada	Se actualiza la lista de películas a esa lista
Resultado	Ok

Prueba-2	Cambiar entre Películas y series
Descripción	El usuario cambia de Películas a series y viceversa, pulsando el botón de cada elemento
Salida Esperada	Se actualiza la lista con series al pulsar el botón de series y con películas al pulsar el botón de películas
Resultado	Ok

Prueba-4	Ver detalles de películas
Descripción	El usuario abre los detalles de la película al pulsar sobre ella
Salida Esperada	Se cambia a la sección /detalles/movie/id con el id de la película y se muestran los detalles
Resultado	Ok

Prueba-5	Ver detalles de series
Descripción	El usuario abre los detalles de la serie al pulsar sobre ella
Salida Esperada	Se cambia a la sección /detalles/tv/id con el id de la serie y se muestran los detalles
Resultado	Ok

Accesibilidad:

Validador: <https://jigsaw.w3.org/css-validator/>

Resultado:

Comprobar Accesibilidad por:

URL Subir Marcados

Dirección:

Sin Problemas

19 Problemas posibles
La mayoría derivados del hosting
(imagen del logo que no se carga)

Revisión de accesibilidad

Revisión de la accesibilidad (Direclines: [WCAG 2.0 \(Nivel AA\)](#))

Problemas conocidos (0) Problemas probables 0(0) Posibles problemas (19) Validación HTML Validación CSS

1.1 Alternativas de texto: Proporcionar alternativas de texto para cualquier contenido no-texto

Criterios de éxito 1.1.1 Contenido no texto (A)

Comprobar 16: El texto Alt no está vacío y la imagen puede ser decorativa.

◆ **Línea 17, columna 381:**

```


Resultado:

Taw

[ES] [EN] [PT]

Resumen

**2 Problemas**  
en 1 criterios de éxito  
**Son necesarias correcciones**

Perceptible 2

Operable 0

Comprendible 0

Robusto 0

**13 Advertencias**  
en 7 criterios de éxito  
**Es necesario revisar manualmente**

Perceptible 2

Operable 5

Comprendible 6

Robusto 0

**18 No verificados**  
en 18 criterios de éxito  
**Comprobación completamente manual**

Perceptible 4

Operable 8

Comprendible 5

Robusto 1

Recurso: <https://replay-fct.00webhostapp.com/> Fecha: 12/12/2023 04:13 Pautas WCAG 2.1 Nivel del análisis: AA Tecnologías: HTML, CSS

modo que puedan percibirlos.

| Pauta                                                     | Nivel | Resultado | Problemas | Advertencias |
|-----------------------------------------------------------|-------|-----------|-----------|--------------|
| 1.1-Textos alternativos                                   |       |           | 0         | 2            |
| 1.1.1 - Contenido no textual                              | A     |           |           | 2            |
| 1.2-Medios basados en el tiempo                           |       |           | 0         | 0            |
| 1.2.1 - Solo audio y solo video (grabaciones)             | A     | na        |           |              |
| 1.2.2 - Subtítulos (pregrabados)                          | A     | na        |           |              |
| 1.2.3 - Audiodescripción o Medio Alternativo (Pregrabado) | A     | na        |           |              |
| 1.2.4 - Subtítulos (en directo)                           | AA    | na        |           |              |
| 1.2.5 - Descripción auditiva (Pregrabada)                 | AA    | na        |           |              |
| 1.3-Adaptable                                             |       |           | 2         | 0            |
| 1.3.1 - Información y relaciones                          | A     |           | 2         |              |
| 1.3.2 - Secuencia con significado                         | A     |           |           |              |
| 1.3.3 - Características sensoriales                       | A     |           |           |              |
| 1.4-Distinguible                                          |       |           | 0         | 0            |
| 1.4.1 - Uso del color                                     | A     |           |           |              |
| 1.4.2 - Control del audio                                 | A     | na        |           |              |
| 1.4.3 - Contraste (Mínimo)                                | A     |           |           |              |
| 1.4.4 - Redimensionamiento del texto                      | AA    |           |           |              |
| 1.4.5 - Imágenes de texto                                 | AA    |           |           |              |

# Conclusión

---

La página web de películas y series que se propone en este proyecto es un proyecto viable que tiene el potencial de cubrir una necesidad que existe en el mercado. La página web utiliza tecnologías modernas que garantizan su fiabilidad y escalabilidad.

El proyecto se ha planificado de forma adecuada para el desarrollo por parte de una única persona, y se ha previsto un presupuesto suficiente para su ejecución. La documentación técnica de la API y del código facilita el mantenimiento y la evolución de la página web en el futuro.

Las propuestas de mejoras que se han planteado podrían mejorar aún más la calidad de la página web.

A nivel personal, he podido desarrollar mis habilidades y adquirir Experiencia. Aprendiendo a utilizar nuevas tecnologías como la librería de React y Node js con sus dependencias. Además, he aprendido el uso de las APIs, en concreto los la API TheMovieDB y sus dificultades.

| Desarrollo personal       | Dificultades                 |
|---------------------------|------------------------------|
| Desarrollo de habilidades | Componentes y Hooks de React |
| Experiencia adquirida     | Uso de la API TheMovieDB     |
| Aprendizaje de React      | Dependencias de Node js      |
| Experiencia con APIs      |                              |