

AVMASTER

Proyecto Fin de Ciclo (PFC)

Desarrollo de Aplicaciones Multiplataforma

Martín García Ramos

Versión: Final Marzo 2024

Índice

1.- Descripción del proyecto y ámbito de implantación:.....	3
1.A. - ¿Qué has desarrollado y por qué?	3
1.B. - ¿Quién lo usará?	3
1.C. - ¿Qué tecnologías has usado?.....	3
1.D. - ¿Se prevén cambios en el futuro?	3
2. – Temporalización del proyecto y fases de desarrollo	4
2.A - PLANIFICACIÓN:	4
2.A.1. PERT:.....	4
2.A.2. GANTT:	5
2.B. - RETOS ENCONTRADOS:	6
3. – Recursos de Hardware y software	7
3.A. - Requisitos de Hardware:	7
3.A.1. Requisitos del navegador:	7
3.A.2. Requisitos para ejecutar el servidor de ReactJS:.....	7
3.B. - Requisitos software:	8
3.A.1. Requisitos del navegador:	8
3.A.2. Requisitos para ejecutar el servidor de ReactJS:.....	8
4. – Arquitectura software y de sistemas	10
4.A. - Diagrama de Secuencia:	10
4.B. - Diagrama de Navegación:	11
4.C. - Diagrama de Jerarquía de componentes en cada página:	11
5. – Descripción de datos	14
6. – Documentación adjunta.....	15
6.A. GITHUB:.....	15
6.B. DIAGRAMAS:.....	15
6.C. TRAILER:.....	15

1.- Descripción del proyecto y ámbito de implantación:

1.A. - ¿Qué has desarrollado y por qué?

Se ha desarrollado una aplicación web que tiene como objetivo ser un lugar seguro en el que poder registrar los contenidos audiovisuales que te gustan, así como ver detalles interesantes del mismo.

1.B. - ¿Quién lo usará?

Forofos de series y películas.

1.C. - ¿Qué tecnologías has usado?

HTML, CSS, JavaScript, React y Bootstrap.

1.D. - ¿Se prevén cambios en el futuro?

Si esta pregunta va orientada a saber si la app es susceptible a ser ampliada y mejorada, sí, se podría mejorar y ampliar fácilmente.

Pero si esta pregunta consulta sobre mi intención personal acerca de si quiero ampliarla. No, no está previsto ampliarla más, desde un inicio la aplicación fue pensada para desarrollarse y ampliarse todo lo posible dentro del rango de tiempo asignado para que el 3/12/2023 la aplicación tuviera todas las funcionalidades implementadas hasta ese momento terminadas.

2. – Temporalización del proyecto y fases de desarrollo

2.A - PLANIFICACIÓN:

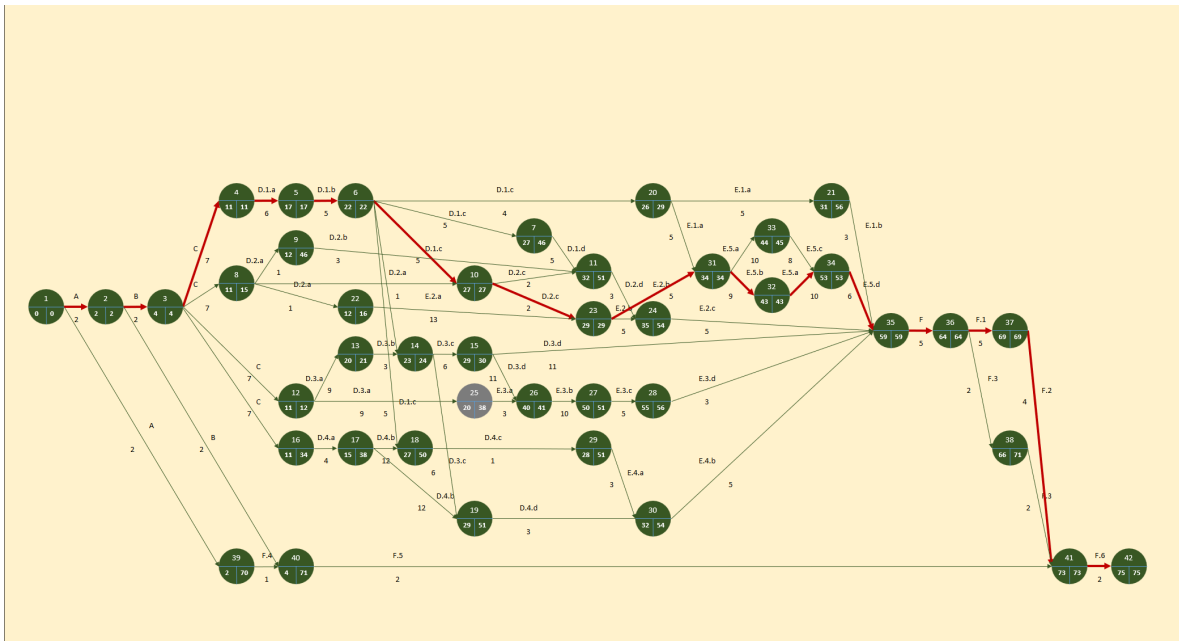
Planificación y descripción de las actividades en los diagramas PERT y GANTT. Debido a que se permiten iteraciones dinámicas, estos diagramas se han localizado en el Excel adjunto a este pdf.

No obstante dejo aquí algunas capturas para dejar constancia de ello:

2.A.1. PERT:

Para ver en mejor detalle el Pert mirar el Excel adjunto.

Nº	ABREVIATURA	TAREA	FECHA INICIO	FECHA FIN	PREDECESOR PARA INICIARSE	DURACIÓN TOTAL (En días)
1	A	Brainstorming	21/10/2023	22/10/2023	—	2,0
2	B	Planificación	23/10/2023	24/10/2023	A	2,0
3	C	Previa	24/10/2023	30/10/2023	B	7,0
4	D	Fase de películas	26/10/2023	03/12/2023		
	D.1	Página Base				
4	D.1.a	Creación de componente React con el HTML: /Buscador	21/10/2023	26/10/2023	C	6,0
5	D.1.b	Creación de estilos CSS para web inicial: /Buscador	23/10/2023	27/10/2023	'D.1.a'	5,0
6	D.1.c	Implementar API para llamarla cuando se realice una búsqueda.	29/10/2023	01/11/2023	'D.1.b'	4,0
7	D.1.d	Implementar estilos CSS para la lista de resultados de la búsqueda.	02/11/2023	06/11/2023	'D.1.c'	5,0
8	D.2	Página Info-item:				
	D.2.a	Creación de componente React con el HTML: Info-item	21/10/2023	21/10/2023	C	1,0
9	D.2.b	Creación de estilos CSS para Información-item	24/10/2023	26/10/2023	'D.2.a'	3,0
10	D.2.c	Adaptar el API para Información-item	27/10/2023	28/10/2023	'D.1.c' & 'D.2.a'	2,0
11	D.2.d	Adaptar estilos CSS para la lista de resultados.	29/10/2023	31/10/2023	'D.1.d' & 'D.2.b' & 'D.2.c'	3,0
12	D.3	Página Cartelera:				
	D.3.a	Creación de componente React con el HTML: Cartelera	21/10/2023	29/10/2023	C	9,0
13	D.3.b	Creación de estilos CSS para Cartelera.	25/10/2023	27/10/2023	'D.3.a'	3,0
14	D.3.c	Adaptar API para llamarla cuando se muestre el listado.	05/11/2023	10/11/2023	'D.1.c' & 'D.3.b'	6,0
15	D.3.d	Adaptar estilos CSS para la lista de resultados.	29/10/2023	08/11/2023	'D.3.c'	11,0
16	D.4	Página Próximamente:				
	D.4.a	Creación de componente React con el HTML: Próximamente	21/10/2023	24/10/2023	C	4,0
17	D.4.b	Creación de estilos CSS para la lista de resultados.	23/10/2023	03/11/2023	'D.4.a'	12,0
18	D.4.c	Adaptar API para llamarla cuando se muestre el listado.	25/10/2023	25/10/2023	'D.1.c' & 'D.4.b'	1,0
19	D.4.d	Adaptar estilos CSS para la lista de resultados.	26/10/2023	28/10/2023	'D.4.b' & 'D.3.c'	3,0
20	E	Fase Series:				
	E.1	Página base:				
20	E.1.a	Adaptar API para detectar series.	28/10/2023	01/11/2023	'D.1.c'	5,0
21	E.1.b	Implementar filtros para separar series de películas.	03/11/2023	05/11/2023	'E.1.a'	3,0
22	E.2	Página Info:				
	E.2.a	Adaptar /Crear subsección HTML: Información-item para que soporte serie	30/10/2023	11/11/2023	'D.2.a'	13,0
23	E.2.b	Adaptar el API para información-item.	25/11/2023	29/11/2023	'D.2.c' & 'E.2.a'	5,0
24	E.2.c	Adaptar los estilos CSS de Información-item	27/11/2023	01/12/2023	'D.2.d' & 'E.2.b'	5,0
25	E.3	Página Cartelera:				
	E.3.a	Cambio de nombre a novedades	21/11/2023	23/11/2023	'D.3.a'	3,0
26	E.3.b	Adaptación de estilos CSS	24/11/2023	03/12/2023	'D.3.d' & 'E.3.a'	10,0
27	E.3.c	Adaptar API para trackear también series.	28/11/2023	02/12/2023	'E.3.b'	5,0
28	E.3.d	Adaptar estilos CSS para la lista de resultados.	30/10/2023	01/11/2023	'E.3.c'	3,0
29	E.4	Página próximamente:				
	E.4.a	Adaptar API para trackear también series.	01/11/2023	03/11/2023	'D.4.c'	3,0
30	E.4.b	Adaptar estilos CSS para la lista de resultados.	04/11/2023	08/11/2023	'D.4.d' & 'E.4.a'	5,0
31	E.5	Página Listas:				
	E.5.a	Implementar listas de películas y series	19/11/2023	28/11/2023	E.1.a' & 'E.2.b'	10,0
32	E.5.b	Adaptar estilos CSS para la lista de resultados.	25/11/2023	03/12/2023	'E.5.a'	9,0
33	E.5.c	Implementar filtrado por géneros de series	26/11/2023	03/12/2023	'E.5.a'	8,0
34	E.5.d	Adaptar estilos CSS para el menú y el filtrado por género.	28/11/2023	03/12/2023	'E.5.b' & 'E.5.c'	6,0
35	F	Fase final	29/11/2023	03/12/2023	D.3.d' & 'E.1.b' & 'E.2.c' & 'E.3.d' & 'E.4.b' & 'E.5.d'	5,0
36	F.1	Test&Debugging	29/11/2023	03/12/2023	'F'	5,0
37	F.2	Creación de Readme	30/11/2023	03/12/2023	'F.1'	4,0
38	F.3	Creación de Trailer de proyecto	03/12/2023	04/12/2023	'F.1'	2,0
39	F.4	Creación de Diagrama CPM-Pert	03/12/2023	03/12/2023	'A'	1,0
40	F.5	Creación de Diagrama Gantt	04/12/2023	05/12/2023	'B' & 'F.4'	2,0
41	F.6	Creación de memoria	05/12/2023	06/12/2023	'F.2' & 'F.3' & 'F.5'	2,0
42	F.7	Entrega	03/12/2023	03/12/2023	'F.6'	1,0
						204,0 / 43



2.A.2. GANTT:

Para ver en mejor detalle el Gantt mirar el Excel adjunto.

Nº	ABREVIATURA	TAREA	FECHA INICIO	FECHA FIN	PREDECESOR PARA INICIARSE	DURACIÓN TOTAL (En días)
1	A	Brainstorming	21/10/2023	22/10/2023	—	2.0
2	B	Planificación	23/10/2023	24/10/2023	A	2.0
3	C	Previa	30/10/2023	30/10/2023	B	7.0
4	D	Fase de películas	26/10/2023	03/12/2023		
	D.1	Página Base				
4	D.1.a	Creación de componente React con el HTML: /Buscador	21/10/2023	26/10/2023	C	6.0
5	D.1.b	Creación de estilos CSS para web inicial: /Buscador	23/10/2023	27/10/2023	'D.1.a'	5.0
6	D.1.c	Implementar API para llamarla cuando se realice una búsqueda.	29/10/2023	01/11/2023	'D.1.b'	4.0
7	D.1.d	Implementar estilos CSS para la lista de resultados de la búsqueda.	02/11/2023	06/11/2023	'D.1.c'	5.0
	D.2	Página Info-Item:				
8	D.2.a	Creación de componente React con el HTML: Info-item	21/10/2023	21/10/2023	C	1.0
9	D.2.b	Creación de estilos CSS para Información-item	24/10/2023	26/10/2023	'D.2.a'	3.0
10	D.2.c	Adaptar el API para Información-item	27/10/2023	30/10/2023	'D.1.c' & 'D.2.a'	2.0
11	D.2.d	Adaptar estilos CSS para la lista de resultados.	29/10/2023	31/10/2023	'D.1.d' & 'D.2.b' & 'D.2.c'	3.0
	D.3	Página Cartelera:				
12	D.3.a	Creación de componente React con el HTML: Cartelera	21/10/2023	29/10/2023	C	9.0
13	D.3.b	Creación de estilos CSS para Cartelera.	25/10/2023	27/10/2023	'D.3.a'	3.0
14	D.3.c	Adaptar API para llamarla cuando se muestre el listado.	05/11/2023	10/11/2023	'D.1.c' & 'D.3.b'	6.0
15	D.3.d	Adaptar estilos CSS para la lista de resultados.	29/10/2023	08/11/2023	'D.3.c'	11.0
	D.4	Página Próximamente:				
16	D.4.a	Creación de componente React con el HTML: Próximamente	21/10/2023	24/10/2023	C	4.0
17	D.4.b	Creación de estilos CSS para la lista de resultados.	23/10/2023	03/11/2023	'D.4.a'	12.0
18	D.4.c	Adaptar API para llamarla cuando se muestre el listado.	25/10/2023	25/10/2023	'D.1.c' & 'D.4.b'	1.0
19	D.4.d	Adaptar estilos CSS para la lista de resultados.	26/10/2023	28/10/2023	'D.4.b' & 'D.3.c'	3.0
	E	Fase Series:	29/10/2023	03/12/2023		
	E.1	Página base:				
20	E.1.a	Adaptar API para detectar series.	28/10/2023	01/11/2023	'D.1.c'	5.0
21	E.1.b	Implementar filtros para separar series de películas.	03/11/2023	05/11/2023	'E.1.a'	3.0
	E.2	Página Info:				
22	E.2.a	Adaptar /Crear subsección HTML: Información-item para que soporte serie	30/10/2023	11/11/2023	'D.2.a'	13.0
23	E.2.b	Adaptar el API para información-item.	25/11/2023	29/11/2023	'D.2.c' & 'E.2.a'	5.0
24	E.2.c	Adaptar los estilos CSS de Información-item	27/11/2023	01/12/2023	'D.2.d' & 'E.2.b'	5.0
	E.3	Página Cartelera:				
25	E.3.a	Cambio de nombre a novedades	21/11/2023	23/11/2023	'D.3.a'	3.0
26	E.3.b	Adaptación de estilos CSS	24/11/2023	03/12/2023	'D.3.d' & 'E.3.a'	10.0
27	E.3.c	Adaptar API para trackear también series.	28/11/2023	02/12/2023	'E.3.b'	5.0
28	E.3.d	Adaptar estilos CSS para la lista de resultados.	30/10/2023	01/11/2023	'E.3.c'	3.0
	E.4	Página próximamente:				
29	E.4.a	Adaptar API para trackear también series.	01/11/2023	03/11/2023	'D.4.c'	3.0
30	E.4.b	Adaptar estilos CSS para la lista de resultados.	04/11/2023	08/11/2023	'D.4.d' & 'E.4.a'	5.0
	E.5	Página Listas:				
31	E.5.a	Implementar listas de películas y series	19/11/2023	28/11/2023	'E.1.a' & 'E.2.b'	10.0
32	E.5.b	Adaptar estilos CSS para la lista de resultados.	25/11/2023	03/12/2023	'E.5.a'	9.0
33	E.5.c	Implementar filtrado por géneros de series	26/11/2023	03/12/2023	'E.5.a'	8.0
34	E.5.d	Adaptar estilos CSS para el menú y el filtrado por género.	28/11/2023	03/12/2023	'E.5.b' & 'E.5.c'	6.0
35	F	Fase final	29/11/2023	03/12/2023	D.3.d' & 'E.1.b' & 'E.2.c' & 'E.3.d' & 'E.4.b' & 'E.5.d'	5.0
36	F.1	Test&Debugging	29/11/2023	03/12/2023	'F'	5.0
37	F.2	Creación de Readme	30/11/2023	03/12/2023	'F.1'	4.0
38	F.3	Creación de Trailer de proyecto	03/12/2023	04/12/2023	'F.1'	2.0
39	F.4	Creación de Diagrama CPM-Pert	03/12/2023	03/12/2023	'A'	1.0
40	F.5	Creación de Diagrama Gantt	04/12/2023	05/12/2023	'B' & 'F.4'	2.0
41	F.6	Creación de memoria	05/12/2023	06/12/2023	'F.2' & 'F.3' & 'F.5'	2.0
42	F.7	Entrega	03/12/2023	03/12/2023	'F.6'	1.0
						204.0 / 43

2.B. - RETOS ENCONTRADOS:

Esencialmente, el tener que desarrollar el proyecto en una tecnología desconocida en el momento del inicio del proyecto, provocó que a lo largo del desarrollo se tuvieran que rehacer diversas secciones varias veces y que el proyecto fuera más laberíntico de lo que ocurriría si se tuviera una planificación correcta.

Además por otro lado, al usar una tecnología tan nueva como ReactJS, he tenido que enfrentarme a diversos problemas con el software externo, uno de ellos por ejemplo fue detectar porqué se producía el error que se indica a continuación cuando se accedía a la aplicación tras haber seleccionado otra pestaña en el navegador y volver a la pestaña donde está la aplicación.

Uncaught runtime errors:

x

ERROR

Cannot redefine property: googletag

TypeError: Cannot redefine property: googletag

at Function.defineProperty (<anonymous>)

at <anonymous>:1:196

at <anonymous>:1:524

Tras muchas pruebas y concluir, que el error no se debía a mi propio proyecto, hayé la solución en inhabilitar la extensión de navegador: Adblocker Stands. Por lo visto, es un error conocido desde Marzo de 2024.

3. – Recursos de Hardware y software

Al tratarse de una web, los únicos requisitos de hardware necesarios son aquellos que se demandan para ejecutar debidamente un navegador como Chrome o Firefox, estos son:

3.A. - Requisitos de Hardware:

3.A.1. Requisitos del navegador:

3.A.1.a. Mínimos:

- Procesador: Procesador Intel Pentium 4 o AMD Athlon 64.
- Memoria RAM: 1 GB.
- Espacio en disco duro: 100 MB de espacio libre.
- Resolución de pantalla: 1280 x 1024.

3.A.1.b. Recomendados:

- Procesador: Procesador Intel Core i3 o AMD Ryzen 3, o superior.
- Memoria RAM: 4 GB o más para un rendimiento óptimo.
- Espacio en disco duro: 500 MB de espacio libre o más para la instalación y almacenamiento en caché.
- Resolución de pantalla: 1920 x 1080 o superior para una experiencia de navegación de alta definición.

3.A.2. Requisitos para ejecutar el servidor de ReactJS:

3.A.2.a. Requisitos Mínimos:

- Procesador: Procesador compatible con arquitectura x86 o x64.
- Memoria RAM: 2 GB o más.
- Espacio en disco duro: Al menos 100 MB de espacio libre para la instalación de Node.js y ReactJS, aunque el espacio requerido dependerá del tamaño de tus proyectos y las dependencias que utilices.

3.A.2.b. Requisitos Recomendados:

- Procesador: Procesador de al menos 2 núcleos, preferiblemente con capacidad para multihilos (Hyper-Threading o similar) para mejorar el rendimiento en tareas de compilación y ejecución.

- Memoria RAM: 4 GB o más para manejar proyectos de tamaño mediano a grande de manera eficiente.
- Disco duro: Un disco SSD (Solid State Drive) puede mejorar significativamente los tiempos de compilación y carga del servidor debido a su velocidad de acceso más rápida en comparación con un disco duro HDD (Hard Disk Drive).

3.B. - Requisitos software:

Cualquier navegador debería de ser capaz de ejecutar la web indistintamente, aunque esta fue testeada con Chrome y Firefox, así que cualquier navegador que use el motor web: Blink o el motor Gecko, así como tener el intérprete de JavaScript V8 o superior.

3.A.1. Requisitos del navegador:

3.A.1.a. Mínimos:

- Sistema operativo: Windows 7 o posterior, macOS X 10.10 o posterior, Ubuntu 14.04 o posterior, Debian 8 o posterior, openSUSE 13.3 o posterior, Fedora Linux 24 o posterior.

3.A.1.b. Recomendados:

- Sistema operativo: La última versión disponible es recomendada para garantizar la seguridad y la compatibilidad con las últimas tecnologías web.

3.A.2. Requisitos para ejecutar el servidor de ReactJS:

3.A.2.a. Requisitos Mínimos:

- Node.js: ReactJS se ejecuta sobre Node.js, por lo que necesitarás instalar Node.js en tu sistema. Se recomienda tener al menos la versión LTS (Long Term Support) para garantizar la estabilidad y compatibilidad. Puedes obtener Node.js desde su sitio web oficial: nodejs.org.

3.A.2.b. Requisitos Recomendados:

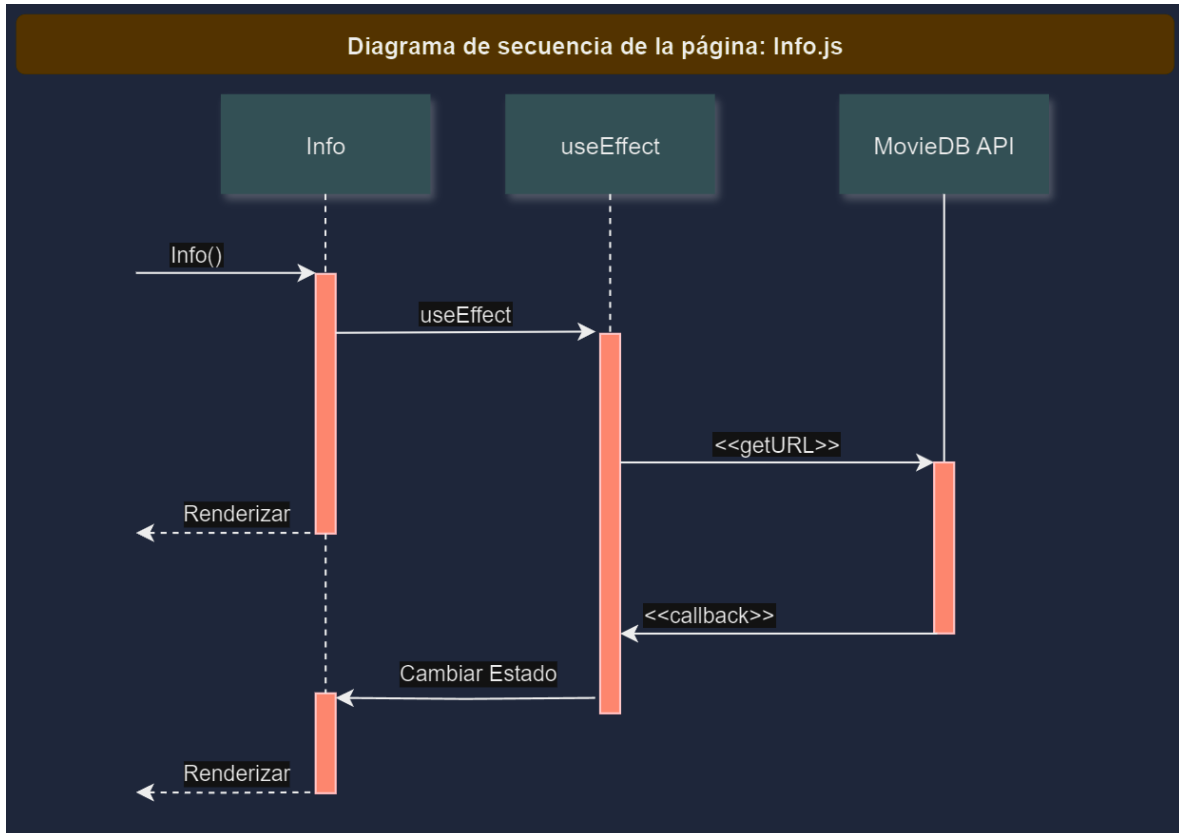
- Gestor de paquetes npm: npm es el gestor de paquetes de Node.js y se utiliza para instalar y administrar las dependencias de los proyectos de ReactJS. Se recomienda tener la última versión de npm instalada junto con Node.js.
- Editor de código: Se recomienda utilizar un editor de código con características que faciliten el desarrollo en ReactJS, como Visual Studio Code, Atom o Sublime Text.

- Sistema de control de versiones: Utilizar un sistema de control de versiones como Git para gestionar el código fuente de tus proyectos.

4. – Arquitectura software y de sistemas

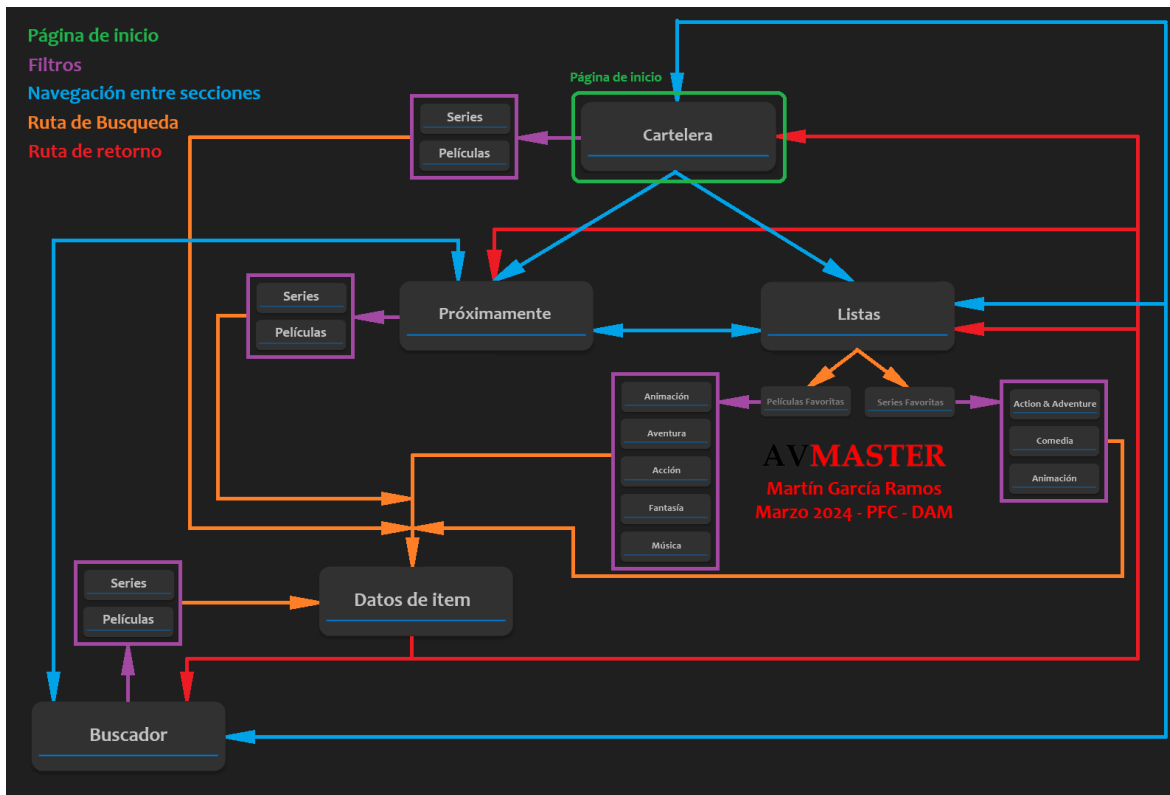
Mi proyecto al ser una web en ReactJS, no tiene “clases” o una orientación a objetos persé, tiene algo parecido a lo que se les denomina componentes y helpers; que en función a su propósito se le puede subdividir en helpers, pages, etc....

4.A. - Diagrama de Secuencia:



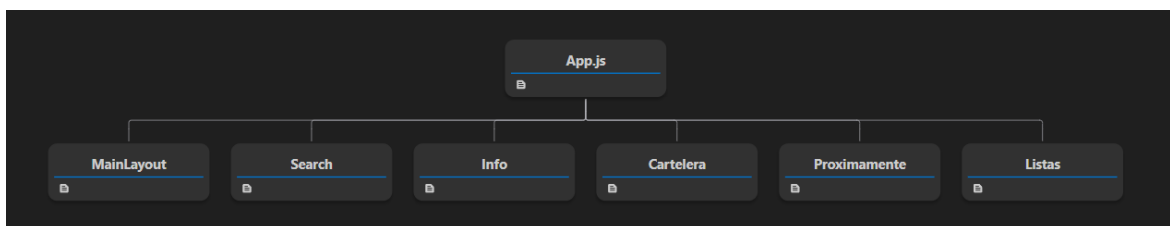
En este diagrama se puede ver cómo se envía Info(), a su vez Info llama al useEffect y este hará una petición asíncrona para obtener los datos de la película. Y mientras tanto, el Info devolverá el renderizado de la páginas de información. Cuando el MovieDB devuelva los datos, el useEffect hará una petición para cambiar el estado de la variable película y por ende se vuelve a realizar una petición para volver a renderizar la página de Info con los nuevos datos actualizados.

4.B. - Diagrama de Navegación:



4.C. - Diagrama de Jerarquía de componentes en cada página:

El proyecto consta de una aplicación que se reparte sus funcionalidades en los siguientes pages y layout, siendo llamados estos desde App.js mediante el Route:



El MainLayout determina las características de la barra superior de la aplicación.

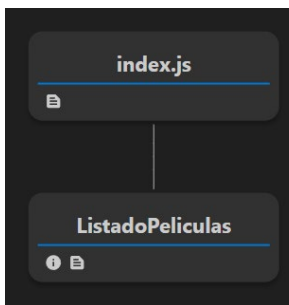
El Search por su parte, determina las propiedades del recuadro de búsqueda superior en su index que a su vez conecta con el componente ListadoPeliculas, que es el que se encarga de listar cada uno de los ítems, bien sean películas o series que se estén devolviendo en este caso como resultado de la búsqueda.



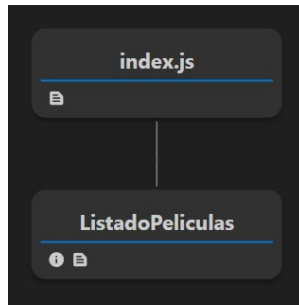
El Info por su parte, se encarga de obtener información sobre una película o serie de televisión desde una API, y luego renderiza esta información utilizando el componente DatosDePelicula.



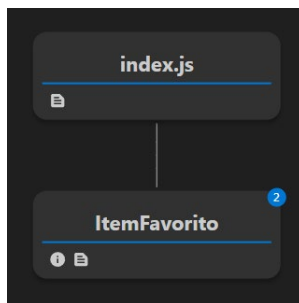
El Cartelera muestra una lista de películas o series dependiendo del tipo seleccionado por el usuario, haciendo solicitudes a una API y mostrando los resultados utilizando componentes personalizados. El usuario puede cambiar entre películas y series seleccionando el tipo deseado; conectando a su vez con el componente DatosDePelicula para mostrar los datos del ítem seleccionado según proceda.



En Próximamente, gracias a su index, se muestra una lista de películas o series próximamente disponibles, haciendo solicitudes a una API y mostrando los resultados utilizando el componente de ListadoPelículas que también se reutiliza en Search y Cartelera. El usuario puede cambiar entre películas y series seleccionando el tipo deseado.



En Listas se encarga de la presentación general de las listas de favoritos y las opciones de filtrado, mientras que el ItemFavorito se encarga de representar cada elemento favorito individual y proporcionar opciones para interactuar con ellos, como eliminarlos de la lista.



En todas las páginas, siempre se puede volver a la página de inicio iterando con los respectivos botones del MainLayout.

5. – Descripción de datos

No tengo mucho tiempo para extenderme con las muchas partes del código que me parecen interesantes, pero una que fue especialmente tediosa de realizar fue la del helper `banderasPaises.js`

Quería que cuando el usuario accediera a la información de una película o una serie apareciera el país de procedencia, pero el api que usaba devolvía los idiomas todos en inglés, así que tomé la decisión de usar emojis para así evitarme el tener que buscar todas las banderas una a una. En el archivo aún hoy día se pueden ver comentados la primera versión en español del diccionario que hice, así como su posterior traducción al inglés del listado. Esto lo hice así por que asegurándome tener todos los países del mundo ordenados alfabéticamente en español luego si algún país tenía una mala traducción o una sentencia incorrecta lo podría localizar fácilmente.

Lamentablemente, y aunque la idea parecía ser una buena idea en un principio lo cierto es que en la práctica, los emojis solo mostraban dos letras, y en muchos casos como el de Kosovo (xk) el emoji muestra letras que tal vez corresponden al nombre del país en su propia lengua más que en español o en inglés.

Por esa causa, tuve que tomarme las molestias de buscar todas las banderas de todos los países del mundo, tanto estados soberanos como no, con la finalidad de que apareciera la bandera en lugar del emoji con las dos letras.

La idea era que al poner el cursor encima de la bandera apareciera un simple texto emergente aclaratorio mostrando el nombre completo del país, pero como no tenía más tiempo al final lo dejé así:

```
1 export const banderaPais = {
2   "Afghanistan":{
3     "icono": "Af",
4     "bandera": "/imagenes/banderas/Afghanistan_2.svg.png"
5   },
6   "Åland":{
7     "icono": "Ax",
8     "bandera": "/imagenes/banderas/Åland.svg.png"
9   },
10  "Albania":{
11    "icono": "AL",
12    "bandera": "/imagenes/banderas/Albania.svg.png"
13  },
14  "Germany":{
15    "icono": "DE",
16    "bandera": "/imagenes/banderas/Germany.svg.png"
17  },
18  "Andorra":{
19    "icono": "AD",
20    "bandera": "/imagenes/banderas/Andorra.svg.png"
21  },
22 }
```

Siendo esto llamado desde el componente: `DatosDePelicula.js` (Línea 223-225)

```
1 {
2   /* PAIS */
3   [div className="mt-4"]
4     <p>País: </p> <ul>{props.pelicula.production_countries.map(e => { return <BanderaPais pais={e} key={e.iso_3166_1} /> })}</ul>
5   </div>
6 }
7 </React>
```

En resumen, una parte, sobre el papel sencilla que me supuso un trabajo inesperadamente arduo y repetitivo.

6. – Documentación adjunta

6.A. *GITHUB:*

https://github.com/AGuyLearningCode/pfc_2023

6.B. *DIAGRAMAS:*

Están adjuntos en el zip que se entregó en classroom así como también están subidos en la carpeta Docs del proyecto en el mismo repositorio de GitHub.

6.C. *TRAILER:*

En el siguiente enlace está disponible en alta calidad.

[https://drive.google.com/file/d/1vjVgibddjwg5xZZsF7gHrEj4gn_Wn5ks/view?usp=drive link](https://drive.google.com/file/d/1vjVgibddjwg5xZZsF7gHrEj4gn_Wn5ks/view?usp=drive_link)

No obstante, también está subido al repositorio de Github una versión comprimida.