

# VUE PROJECT



Mihai-Sebastian Burghelea  
2023-2024

# INDEX

<b>Instal·lacions:</b>	<b>3</b>
TAILWIND CSS	3
Heroicons:	4
Vue-Router:	4
Axios	4
<b>Components</b>	<b>5</b>
Navbar	5
Footer	5
App.vue	6
<b>HOME</b>	<b>6</b>
<b>Anime.vue i Manga.vue</b>	<b>7</b>
<b>AnimeDetail.vue i MangaDetail.vue</b>	<b>9</b>
<b>UserDetail.vue</b>	<b>10</b>
<b>Index.js</b>	<b>12</b>
<b>Problemes</b>	<b>13</b>
API	13

# Instal·lacions:

## TAILWIND CSS

```
alumne@sebastian:~/Escriptori/Programacio/animetaill$ npm install -D tailwindcss postcss autoprefixer
added 116 packages, and audited 147 packages in 4s

37 packages are looking for funding
  run `npm fund` for details

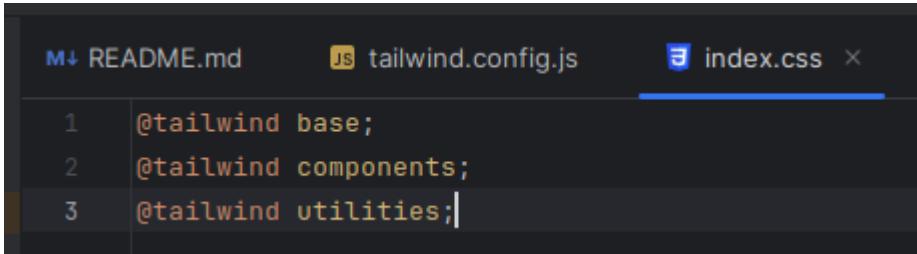
found 0 vulnerabilities

alumne@sebastian:~/Escriptori/Programacio/animetaill$ npx tailwindcss init -p
Created Tailwind CSS config file: tailwind.config.js
Created PostCSS config file: postcss.config.js
alumne@sebastian:~/Escriptori/Programacio/animetaill$
```



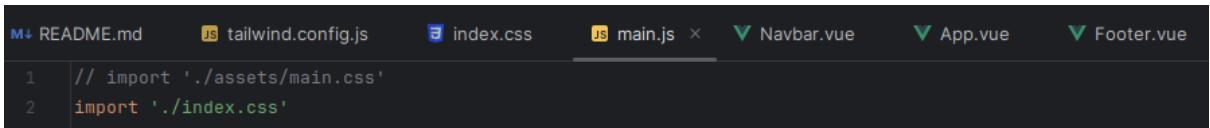
A screenshot of a code editor showing the `tailwind.config.js` file. The code defines a configuration object with properties `content`, `theme`, and `plugins`. The `content` array includes files like `index.html` and `src/**/*.{vue,js,ts,jsx,tsx}`. The `theme` object has an `extend` property.

```
1  /** @type {import('tailwindcss').Config} */
2  no usages
3  export default {
4    content: [
5      "./index.html",
6      "./src/**/*.{vue,js,ts,jsx,tsx}",
7    ],
8    theme: {
9      extend: {},
10    },
11    plugins: [],
12  }
```



A screenshot of a code editor showing the `index.css` file. It contains three lines of CSS starting with `@tailwind` followed by `base`, `components`, and `utilities`.

```
1  @tailwind base;
2  @tailwind components;
3  @tailwind utilities;
```



A screenshot of a code editor showing the `main.js` file. It imports `./assets/main.css` and `./index.css`.

```
1 // import './assets/main.css'
2 import './index.css'
```

## Heroicons:

```
alumne@sebastian:~/Escriptori/Programacio/animetail$ npm install @heroicons/vue

added 1 package, and audited 148 packages in 3s

37 packages are looking for funding
  run `npm fund` for details

found 0 vulnerabilities
```

## Vue-Router:

```
alumne@sebastian:~/Escriptori/Programacio/animetail$ npm install vue-router@4

up to date, audited 148 packages in 1s

37 packages are looking for funding
  run `npm fund` for details

found 0 vulnerabilities
```

## Axios

```
alumne@sebastian:~/Escriptori/Programacio/animetail$ npm install axios

added 9 packages, and audited 158 packages in 1s

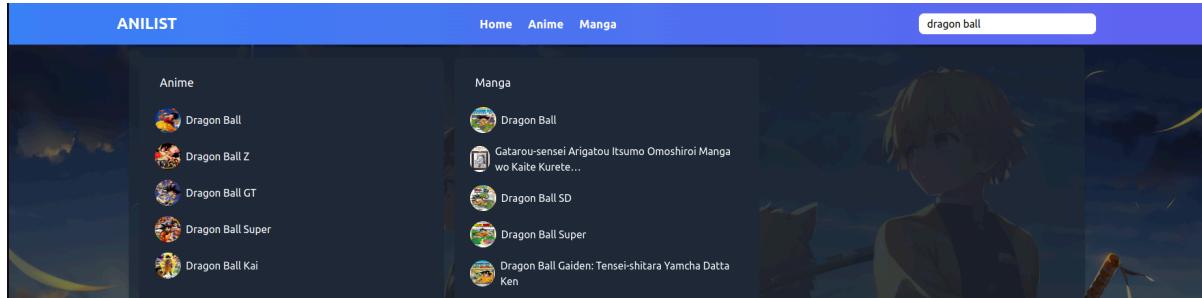
38 packages are looking for funding
  run `npm fund` for details

found 0 vulnerabilities
```

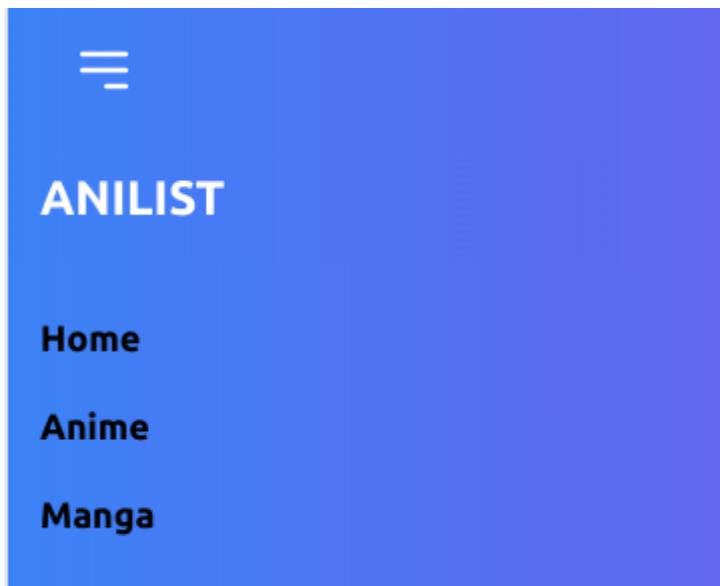
# Components

## Navbar

La navbar conté una barra de búsqueta amb resultats com anime, manga i usuaris.



La navbar té un drawer que l'ha fa responsive.



## Footer



## App.vue

A app.vue es a on es fa la crida dels components i de les vistes.

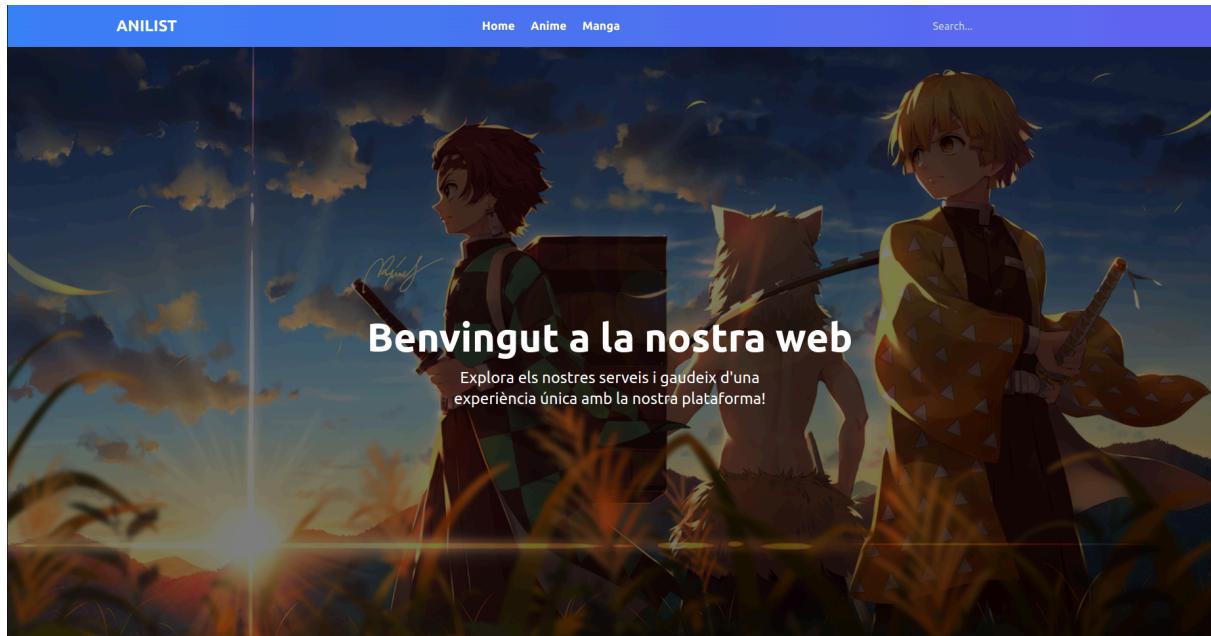
```
<template>
  <Navbar :transparent="isTransparent"></Navbar>
  <router-view @update-navbar-style="handleNavbarStyle"></router-view>
  <Footer></Footer>
</template>

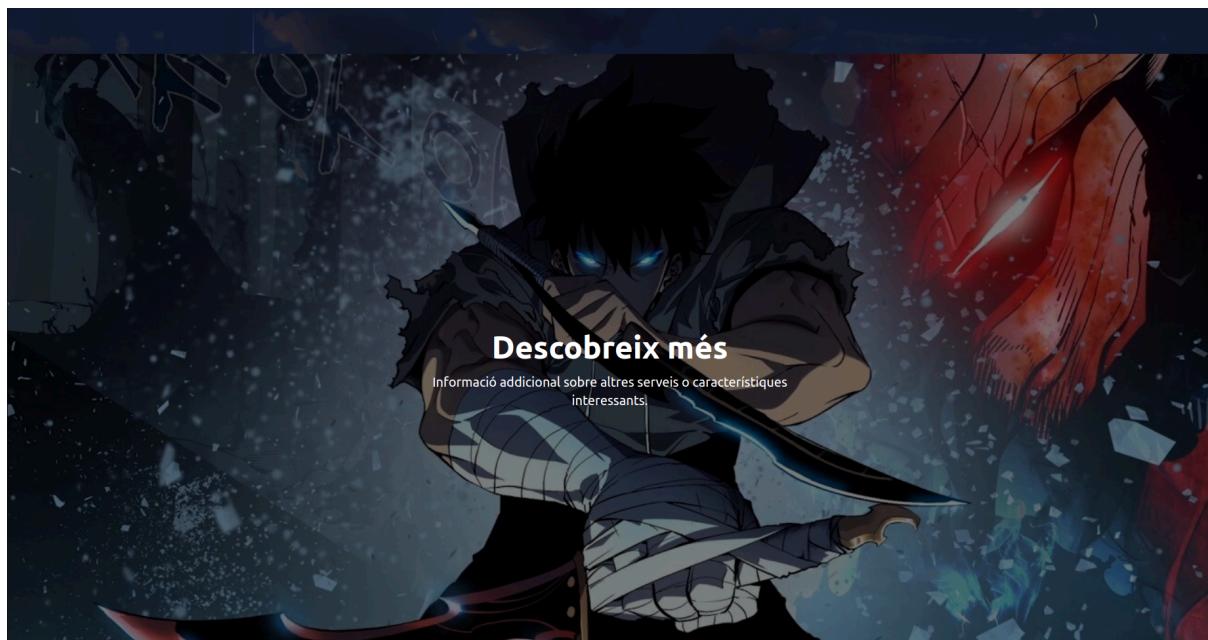
<script setup>
import { ref } from 'vue';
import Navbar from "@/components/Navbar.vue";
import Footer from "@/components/Footer.vue";

const isTransparent = ref({ value: false });

usage 🔔 Mihai-Sebastian
function handleNavbarStyle({ transparent }) {
  isTransparent.value = transparent;
}
</script>
```

## HOME





## Anime.vue i Manga.vue

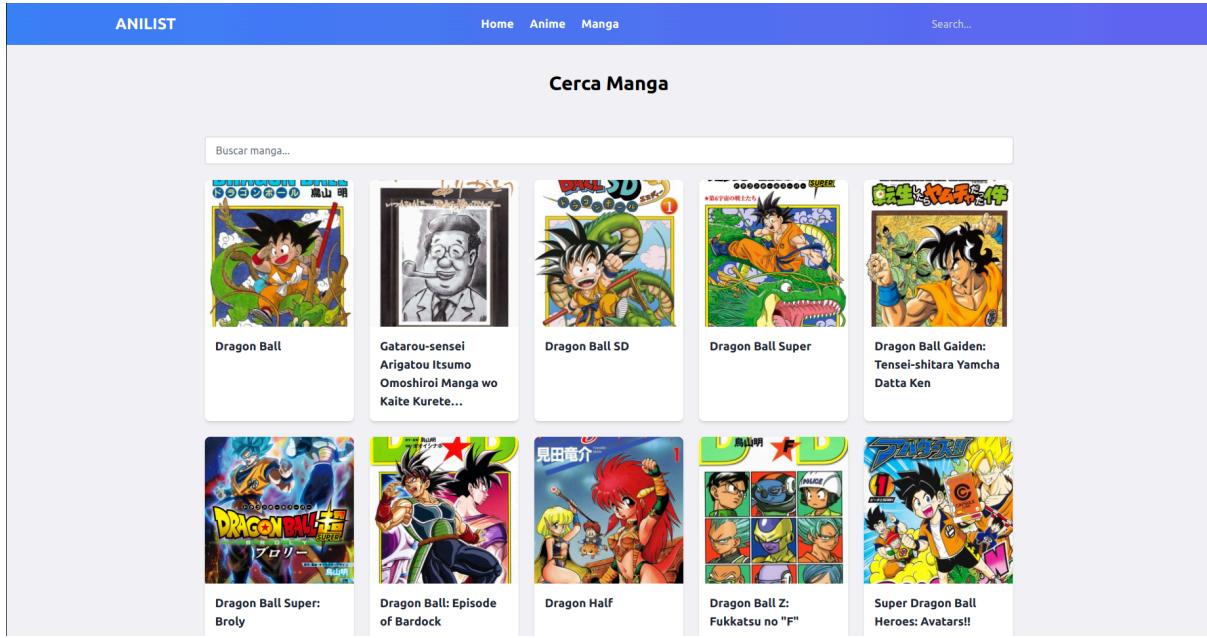
ANILIST

Home Anime Manga Search...

### Cerca Anime

Buscar anime...

ONE PIECE	THE ONE PIECE	ONE PIECE (Movie)	ONE PIECE: Episode of Merry - Mou Hitori no Nakama no Monogatari	ONE PIECE STAMPEDE
ONE PIECE: Episode of Nami - Koukaishi no	ONE PIECE: ROMANCE DAWN STORY	Noroi no One Piece	ONE PIECE: Heart of Gold	ONE PIECE THE MOVIE: Karakuri-jou

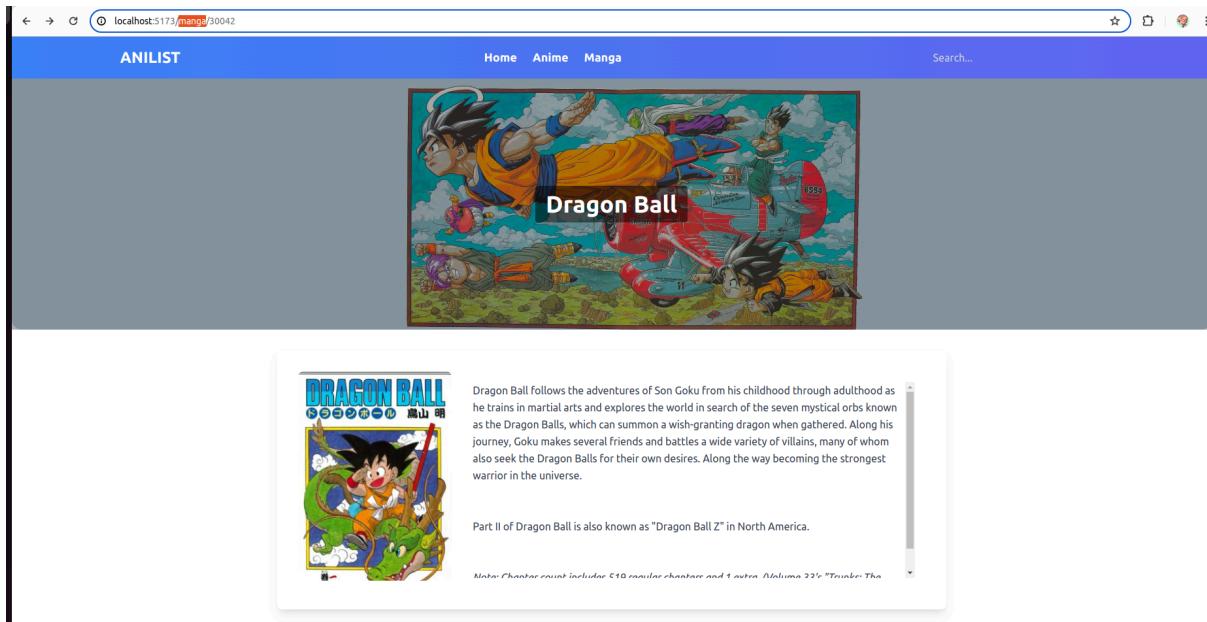


Les 2 pàgines son iguals però a una es fa la crida als type ANIME i als altres al type MANGA

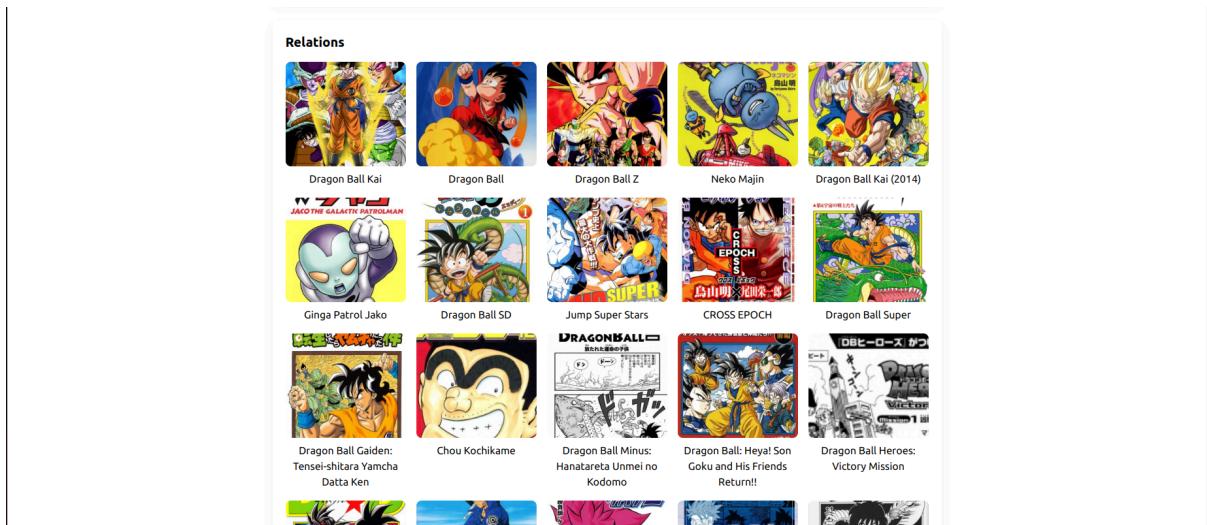
```
const searchQueryText = `query ($search: String) {
    Page {
        media (search: $search, type: ANIME, isAdult: false) {
            id
        }
    }
}
```

```
const searchQueryText = `query ($search: String) {
    Page {
        media (search: $search, type: MANGA, isAdult: false) {
            id
        }
    }
}
```

# AnimeDetail.vue i MangaDetail.vue



Es pot fer click a una relació i depenent de si es manga o anime va cap a MangaDetail o AnimeDetail.



The screenshot shows the ANILIST application interface. At the top, there is a navigation bar with links for Home, Anime, Manga, and a search bar. The main content area features a large banner image for 'Dragon Ball Z' with several characters from the show. Below the banner, there is a smaller thumbnail image of the same characters. To the right of the thumbnail, there is a descriptive text block:

Goku is back with his new son, Gohan, but just when things are getting settled down, the adventures continue. Whether he is facing enemies such as Freeza, Cell, or Boo, Goku is proven to be an elite of his own and discovers his race, Saiyan. He meets many new people, gaining allies as well as enemies, as he still finds time to raise a family and be the happy-go-lucky Saiyan he is.

(Source: Anime News Network)

## UserDetail.vue

Al buscar un usuari va cap a UserDetail.vue

The screenshot shows the ANILIST application interface, specifically the UserDetail.vue page for a user named 'Bardock21'. The top navigation bar is visible, and the search bar contains the username 'bardock21'. The main content area is divided into three sections: 'Anime', 'Manga', and 'Users'. The 'Anime' section lists two entries: 'Dragon Ball: Episode of Bardock' and 'Dragon Ball Z: Tatta Hitori no Saishuu Kessen - Freeza ni Idonda Z Senshi Son Goku no Chi'. The 'Manga' section lists one entry: 'Dragon Ball: Episode of Bardock'. The 'Users' section lists one entry: 'Bardock21'. Below these sections, there is a larger image of a character from the anime, with the text 'Username: Bardock21' displayed above it.

Aquí agafa les llistes que té l'usuari i les mostra segons l'usuari les tingue catalogades.

localhost:5173/user/932815

Dropped

BORUTO: NARUTO NEXT GENERATIONS    JoJo no Kimyou na Bouken: Diamond wa Kudskēnai    Chuunibyou demo Koi ga Shitai!    Kawaii dake ja Nai Shikimori-san    Kanojo mo Kanojo Season 2

Watching

ONE PIECE    Boku no Hero Academia    SPYxFAMILY Season 2    Kaijuu 8-gou    Mushoku Tensei II: Isekai Ittara Honki Dasu Part 2

Kimetsu no Yaiba:  
Hashira Geiko-hen

# Index.js

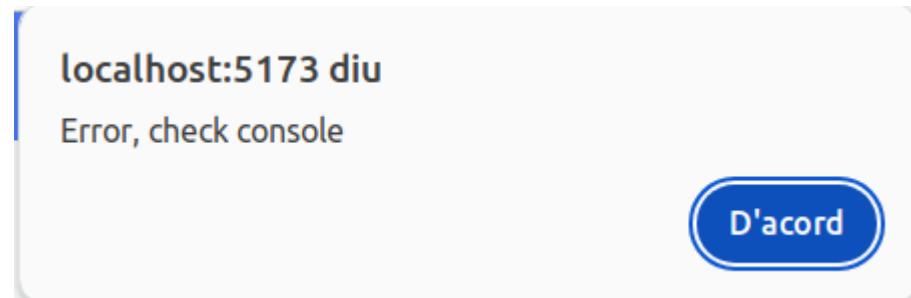
Aquí ficaré les rutes per al redirecccionament.

```
1 import { createRouter, createWebHistory } from 'vue-router'
2 import App from '@/App.vue';
3 import Anime from '@/views/Anime.vue';
4 import Manga from '@/views/Manga.vue';
5 import Home from '@/views/Home.vue';
6 import AnimeDetail from '@/views/AnimeDetail.vue';
7 import MangaDetail from '@/views/MangaDetail.vue';
8 import UserDetail from '@/views/UserDetail.vue';
9
10 const router : Router = createRouter( options: {
11     history: createWebHistory(import.meta.env.BASE_URL),
12     routes: [
13         {
14             path: '/',
15             name: 'home',
16             component: Home
17         },
18         {
19             path: '/anime',
20             name: 'anime',
21             component: Anime
22         },
23         {
24             path: '/manga',
25             name: 'manga',
26             component: Manga
27         },
28         {
29             path: '/anime/:id',
30             name: 'AnimeDetail',
31             component: AnimeDetail,
32             props: true
33         },
34         {
35             path: '/manga/:id',
36             name: 'MangaDetail',
37             component: MangaDetail,
38             props: true
39         },
40         {
```

# Problemes

## API

Com es async cada vegada que enviem una lletra envia una petició, això fa que si l'usuari escriu molt ràpid o borra se peta la pàgina.



Per a solucionar el problema afegeixo un debounce, que el que fa és enviar la petició si detecta que has estat 300 mil·lisegons sense tocar el teclat.

```
// Crear una función debounced para la búsqueda
const debouncedSearch = debounce(searchAnime, wait: 300);

//+ usages - Mihai-Sebastian
export function debounce(func, wait) {
  let timeout;
  return function (...args) : void {
    clearTimeout(timeout);
    timeout = setTimeout( handler: () => func.apply(this, args), wait);
  };
}
```

Un altre error es que per defecte no sortia contingut, així que li passo una busqueda per defecte quan l'usuari no escriu res o entra per primera vegada.

```
if (searchQuery.value.trim() === '') {
  // Si la barra de búsqueda está vacía, mostrar el contenido predeterminado
  fetchAnime( searchText: 'One Piece');
```