

PROJECTE

Creem el projecte de vue:

```
alumni@insebre-com:~$ npm init vite paisos
Need to install the following packages:
create-vite@5.2.3
Ok to proceed? (y) y

> npx
> create-vite paisos

✓ Select a framework: > Vue
✓ Select a variant: > TypeScript

Scaffolding project in /home/alumni/paisos...

Done. Now run:

  cd paisos
  npm install
  npm run dev

alumni@insebre-com:~$ cd paisos/
alumni@insebre-com:~/paisos$ npm install

added 43 packages, and audited 44 packages in 21s

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

found 0 vulnerabilities
alumni@insebre-com:~/paisos$ npm run dev

> paisos@0.0.0 dev
> vite

VITE v5.2.11  ready in 223 ms

→ Local:   http://localhost:5173/
→ Network: use --host to expose
→ press h + enter to show help
```

Si accedim al projecte, podem veure la web inicial de vue:



Vite + Vue

count is 0

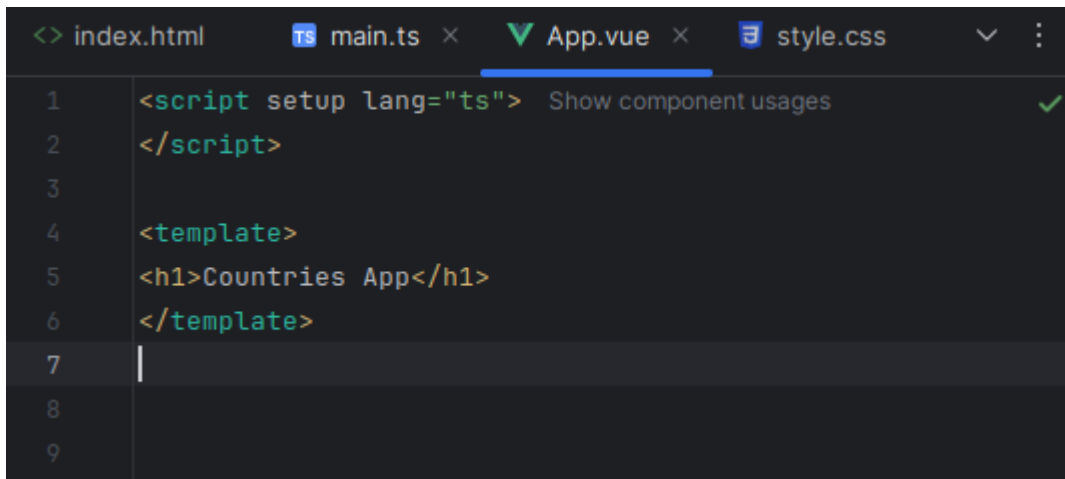
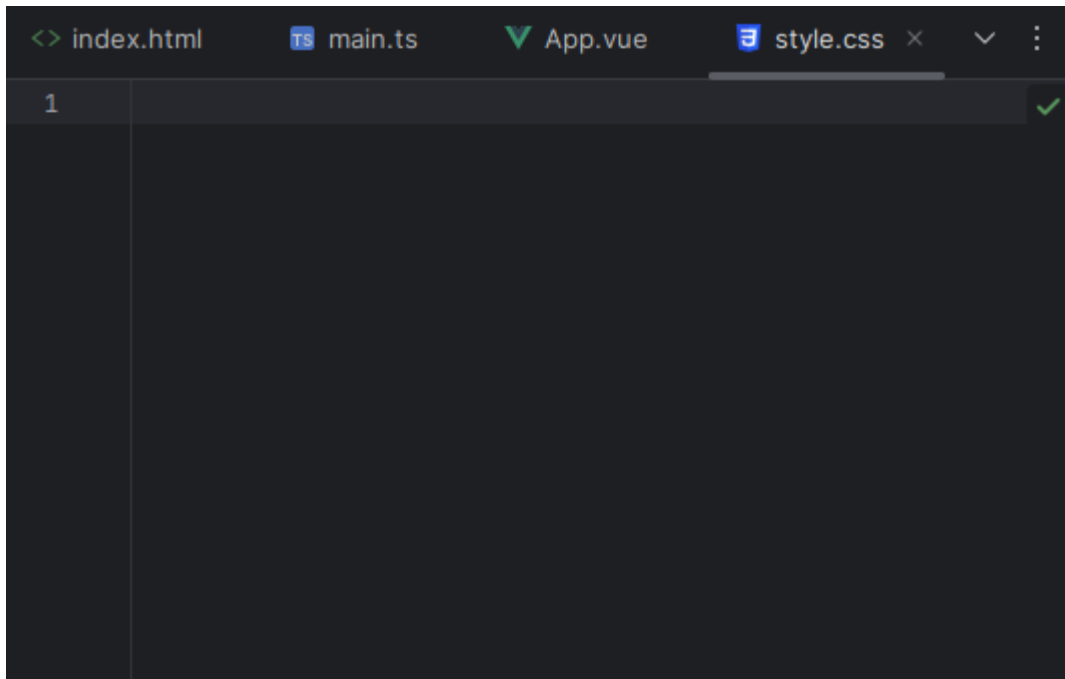
Edit `components/HelloWorld.vue` to test HMR

Check out [create-vue](#), the official Vue + Vite starter

Install [Volar](#) in your IDE for a better DX

Click on the Vite and Vue logos to learn more

Deixem App.vue i style.css “sense res” per iniciar el nostre projecte:



Ens instal·lem el Framework de Tailwind Css per ajudar a la creació d'estils de la nostra web:

Enllaç: <https://tailwindcss.com/docs/guides/vite>

```
alumne@insebre-com:~/paisos$ sudo npm install -D tailwindcss@latest postcss@latest autoprefixer@latest
[sudo] contrasenya per a alumne:

added 114 packages, and audited 158 packages in 7s

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

found 0 vulnerabilities
```

```
alumine@insebre-com:~/paisos$ sudo npx tailwindcss init -p

Created Tailwind CSS config file: tailwind.config.js
Created PostCSS config file: postcss.config.js
```

Ens tindria que apareixer el tailwind.config.js:

```
JS tailwind.config.js
```

Configurem aquest fitxer segons la documentació:

```
/** @type {import('tailwindcss').Config} */
no usages
export default {
  content: [
    "./index.html",
    "./src/**/*.vue",
  ],
}
```

```
App.vue style.css
@tailwind base;
@tailwind components;
@tailwind utilities;
```

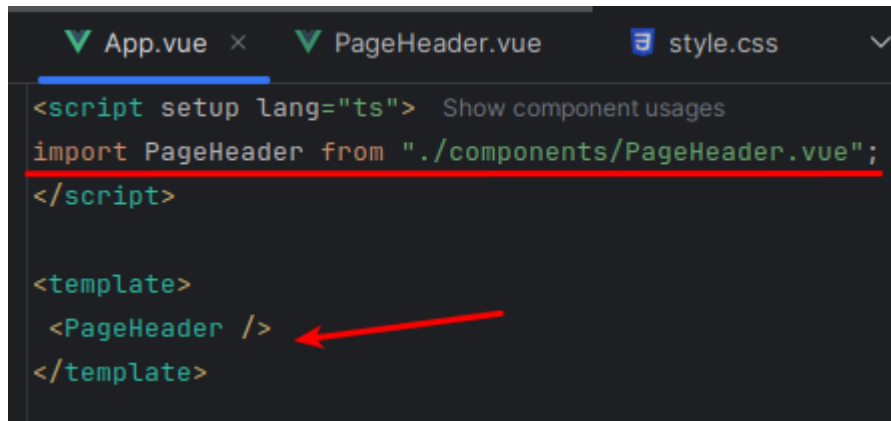
Un cop fet això, crearem el nostre primer component, per fer-ho crearem un nou document Vue dintre la carpeta de components:

```
components
  HelloWorld.vue
  PageHeader.vue
```

Aquí podem afegir el que vulguem:

```
App.vue PageHeader.vue
<template> Show component usages
<div>
  <h1>Countries App</h1>
  <h2>App to search countries</h2>
</div>
</template>
```

I si el volem mostrar, el tindrem que importar a App.vue u cridar-lo:



```
<script setup lang="ts"> Show component usages
import PageHeader from "./components/PageHeader.vue";
</script>

<template>
  <PageHeader />
</template>
```

Per descarregar la API dels països, fem el següent:

Enllaç: <https://restcountries.com/>

Ens descarquem el mòdul axios que ens permet fer crides http:

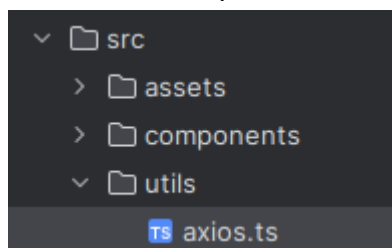
```
insebre-com% sudo npm add axios

added 9 packages, and audited 167 packages in 1s

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

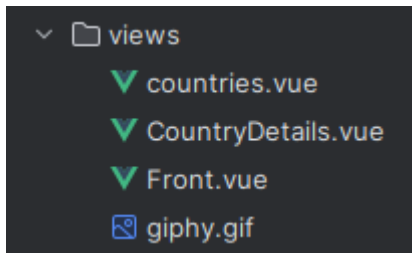
found 0 vulnerabilities
```

Ara, dintre la carpeta de /src, creem la carpeta utils amb el fitxer axios.ts dintre:

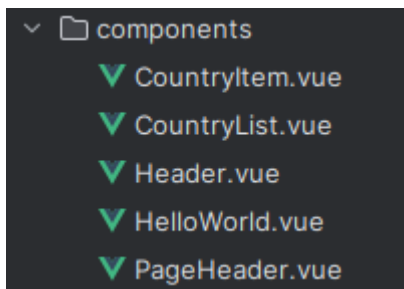


```
<template>
<div class="text-center py-24">
  <h1 class="text-8x1 font-bold tracking-widest
text-sky-700">Countries App</h1>
  <h6 class="mt-4 text-1x">App to search countries</h6>
</div>
</template>
```

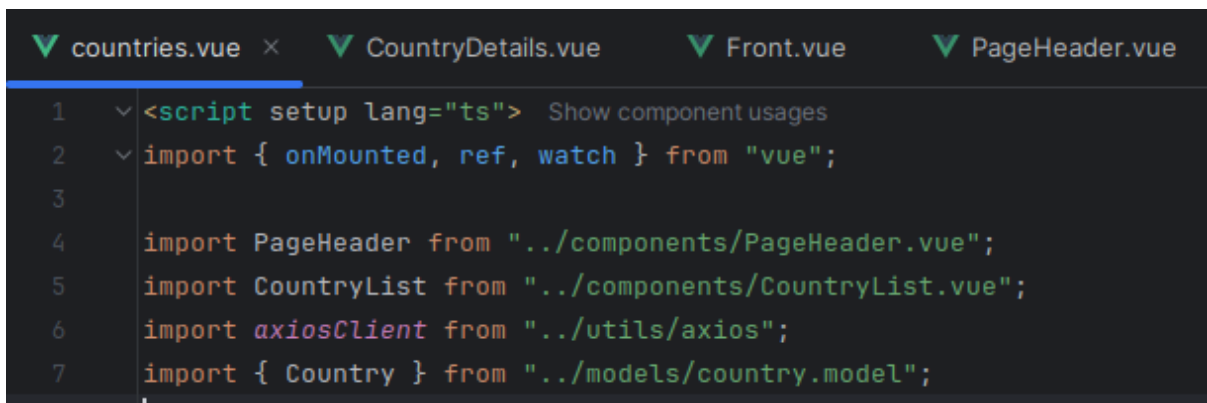
He creat un component per cada vista que m'agradaria integrar a la web i les he ficat a "views":



Dintre la carpeta “components” he ficat tots els components a importar:



Per exemple a la vista de countries:



Per enllaçar les pàgines, creem la carpeta “router → index.js” i allí fiquem les pàgines a importar, en aquest cas la de front i countries. Definim l’arrel com al Front i countries:

```

import { createRouter, createWebHistory } from 'vue-router';
import Front from "../views/Front.vue";
import countries from "../views/countries.vue";

const router : Router = createRouter({ options: {
  history: createWebHistory(import.meta.env.BASE_URL),
  routes: [
    {
      path: '/',
      name: 'Front',
      component: Front
    },
    {
      path: '/countries',
      name: 'Countries',
      component: countries
    },
  ],
});
Show usages
export default router

```

Dintre App.vue, cridem el router-view:

```

<template> Show component usages
  <router-view></router-view>
</template>

<script setup>
  import Front from "../views/Front.vue";
</script>

<style>
</style>

```

CONTINGUT:

App.vue:

```
App.vue x
1  <template> Show component usages
2  <router-view ></router-view>
3  </template>
4
5  <script setup>
6  import Front from "../views/Front.vue";
7  </script>
8  💡
9  <style>
10 </style>
```

Countries.vue:

countries.vue x

```
1  <script setup lang="ts"> Show component usages
2  import { onMounted, ref, watch } from "vue";
3
4  import PageHeader from "../components/PageHeader.vue";
5  import CountryList from "../components/CountryList.vue";
6  import axiosClient from "../utils/axios";
7  import { Country } from "../models/country.model";
8
9  const countries = ref<Country[]>( value: []);
10 const search = ref( value: "");
11 const filteredCountries = ref<Country[]>( value: []);
12 const page = ref( value: 1);
13 const itemsPerPage = ref( value: 12);
14 const paginatedCountries = ref<Country[]>( value: []);
15 const totalItems = ref( value: 0);
16
17 Show usages
18 const fetchCountries = async () => {
19   try {
20     const { data } = await axiosClient.get( url: "/all");
21     countries.value = data;
22     totalItems.value = countries.value.length;
23   } catch (error) {
24     console.log(error);
25   }
26 };
27
28 Show usages
29 const filterCountries = () => {
30   filteredCountries.value = countries.value.filter((country) =>
31     country.name.common.toLowerCase().includes(search.value.toLowerCase())
32 );
33 };
34
35 Show usages
36 const sliceCountries = (currentCountries: Country[]) => {
37   const start = (page.value - 1) * itemsPerPage.value;
38   const end = page.value * itemsPerPage.value;
39   paginatedCountries.value = currentCountries.slice(start, end);
40 };
41
```

Show usages

```
const changePage = (newPage: number) => {
  page.value = newPage;
};

onMounted( hook: () => {
  fetchCountries();
});

watch( sources: [countries, page, filteredCountries], cb: () => {
  sliceCountries(
    currentCountries: filteredCountries.value.length <= 0 && search.value === ""
      ? countries.value
      : filteredCountries.value
  );
});
</script>
```

```
<template>
  <PageHeader />
  <div class="container max-w-screen-lg mx-auto px-6">
    <div class="mb-8">
      <input
        type="text"
        class="border border-gray-300 rounded w-full p-1 px-4"
        placeholder="Seacrh by country name"
        v-model="search"
        @input="filterCountries"
      />
    </div>
    <div class="mb-8 flex justify-center space-x-6">
      <button
        :disabled="page <= 1"
        :class="{ 'opacity-50': page <= 1 }"
        @click="changePage( newPage: page - 1 )"
        class="border border-gray-300 rounded px-2 py-0.5 hover:bg-gray-200"
      >
```

```

        Previous
      </button>
      <button
        :disabled="page >= totalItems / itemsPerPage"
        :class="{ 'opacity-50': page >= totalItems / itemsPerPage }"
        @click="changePage( newPage: page + 1)"
        class="border border-gray-300 rounded px-2 py-0.5 hover:bg-gray-200"
      >
        Next
      </button>
    </div>
    <CountryList :countries="paginatedCountries" />
  </div>
</template>

```

Front.vue:

```
1 <template> Show component usages
2   <div class="front-page">
3     <div><button @click="redirectToApp">{{ pageTitle }}</button></div>
4   </div>
5 </template>
6
7 <script setup lang="ts">
8 import { useRouter } from 'vue-router';
9
10 const router = useRouter();
11 const pageTitle = "COUNTRIES";
12
13 Show usages
14 const redirectToApp = () => {
15   if (router) {
16     // Redirigir a la ruta /países
17     router.push('/countries');
18   } else {
19     console.error("router is undefined");
20   }
21 };
22
23 <style scoped>
24 @import url("https://fonts.googleapis.com/css?family=Source+Sans+Pro");
25
26 .front-page {
27   color: #fff;
28   height: 100vh;
29   overflow: hidden;
30   /* Añade la URL del GIF como fondo */
31   background-image: url('https://media.giphy.com/media/bXhiABcqQ6T3W/giphy.gif');
32   /* Ajusta la propiedad background-size según sea necesario */
33   background-size: cover;
34   background-repeat: no-repeat;
35   display: flex;
36   align-items: center;
37   justify-content: center;
38 }
```

```

.front-page div {
  background: 50% 100% / 50% 50% no-repeat
    radial-gradient(ellipse at bottom, #fff, transparent, transparent);
  -webkit-background-clip: text;
  background-clip: text;
  color: transparent;
  font-size: 10vw;
  font-family: "Source Sans Pro", sans-serif;
  animation: reveal 3000ms ease-in-out forwards 200ms,
    glow 2500ms linear infinite 2000ms;
}

@keyframes reveal {
  80% {
    letter-spacing: 8px;
  }
  100% {
  }
}

@keyframes glow {
  40% {
    text-shadow: 0 0 8px #fff;
  }
}
</style>

```

CountryItem.vue:

```
CountryItem.vue ×
1  <script setup lang="ts"> Show component usages
2  import { defineProps } from "vue";
3  import { Country } from "../models/country.model";
4
5  Show usages
6  interface Props {
7    country: Country;
8  }
9
10 defineProps<Props>();
11
12 </script>
13
14 <template>
15   <div class="h-56 sm:h-36">
16     
21   </div>
22   <h4 class="text-xl font-bold text-gray-500">{{ country.name.common }}</h4>
23   <p class="text-sm text-gray-400">
24     Capital:
25     <span v-for="capital in country.capital" class="font-bold">{{
26       capital
27     }}</span>
28   </p>
29   <p class="text-sm text-gray-400">
30     Population:
31     <span class="font-bold">{{ country.population?.toLocaleString() }}</span>
32   </p>
33   <p class="text-sm text-gray-400">
34     Region:
35     <span class="font-bold">{{ country.region }}</span>
36   </p>
37 </template>
```

CountryList.vue:

```
CountryList.vue ×
1  <script setup lang="ts"> Show component usages
2  import { defineProps } from "vue";
3  import { Country } from "../models/country.model";
4  import CountryItem from "../CountryItem.vue";
5
6  Show usages
7  interface Props {
8    countries: Country[];
9  }
10 defineProps<Props>();
11 </script>
12
13 <template>
14   <div class="grid grid-cols-1 sm:grid-cols-2 md:grid-cols-3 gap-3">
15     <div
16       v-for="country in countries"
17       class="border border-gray-300 rounded p-3"
18     >
19       <CountryItem :country="country" />
20     </div>
21   </div>
22 </template>
```

Header.vue:

```

▼ Header.vue ×
1 <header class="bg-gray-900"> Show component usages
2 <nav class="mx-auto flex max-w-7xl items-center justify-between p-6 lg:px-8" aria-label="Global">
3   <div class="flex lg:flex-1">
4     <a href="#" class="-m-1.5 p-1.5">
5       <span class="sr-only">Your Company</span>
6       
7     </a>
8   </div>
9   <div class="flex lg:hidden">
10    <button type="button" class="-m-2.5 inline-flex items-center justify-center rounded-md p-2.5 text-gray-400">
11      <span class="sr-only">Open main menu</span>
12      <svg class="h-6 w-6" fill="none" viewBox="0 0 24 24" stroke-width="1.5" stroke="currentColor" aria-hidden="true">
13        <path stroke-linecap="round" stroke-linejoin="round" d="M3.75 6.75h16.5M3.75 12h16.5M3.75 16.5h16.5" />
14      </svg>
15    </button>
16  </div>
17  <div class="hidden lg:flex lg:gap-x-12">
18    <a href="#" class="text-sm font-semibold leading-6 text-white">Product</a>
19    <a href="#" class="text-sm font-semibold leading-6 text-white">Features</a>
20    <a href="#" class="text-sm font-semibold leading-6 text-white">Marketplace</a>
21    <a href="#" class="text-sm font-semibold leading-6 text-white">Company</a>
22  </div>
23  <div class="hidden lg:flex lg:flex-1 lg:justify-end">
24    <a href="#" class="text-sm font-semibold leading-6 text-white">Log in <span aria-hidden="true">&rarr;</span></a>
25  </div>
26 </nav>
27 <!-- Mobile menu, show/hide based on menu open state. -->
28 <div class="lg:hidden" role="dialog" aria-modal="true">
29   <!-- Background backdrop, show/hide based on slide-over state. -->
30   <div class="fixed inset-0 z-10" />
31   <div class="fixed inset-y-0 right-0 z-10 w-full overflow-y-auto bg-gray-900 px-6 py-6 sm:max-w-sm sm:ring-1 sm:ring-white/10">
32     <div class="flex items-center justify-between">
33       <a href="#" class="-m-1.5 p-1.5">
34         <span class="sr-only">Your Company</span>
35         
36       </a>
37       <button type="button" class="-m-2.5 rounded-md p-2.5 text-gray-400">
38         <span class="sr-only">Close menu</span>
39         <svg class="h-6 w-6" fill="none" viewBox="0 0 24 24" stroke-width="1.5" stroke="currentColor" aria-hidden="true">
40           <path stroke-linecap="round" stroke-linejoin="round" d="M6 18L18 6M6 6L18 18" />
41         </svg>
42       </button>
43     </div>
44     <div class="mt-6 flow-root">
45       <div class="-my-6 divide-y divide-gray-500/25">
46         <div class="space-y-2 py-6">
47           <a href="#" class="-mx-3 block rounded-lg px-3 py-2 text-base font-semibold leading-7 text-white hover:bg-gray-800">Product</a>
48           <a href="#" class="-mx-3 block rounded-lg px-3 py-2 text-base font-semibold leading-7 text-white hover:bg-gray-800">Features</a>
49           <a href="#" class="-mx-3 block rounded-lg px-3 py-2 text-base font-semibold leading-7 text-white hover:bg-gray-800">Marketplace</a>
50           <a href="#" class="-mx-3 block rounded-lg px-3 py-2 text-base font-semibold leading-7 text-white hover:bg-gray-800">Company</a>
51         </div>
52         <div class="py-6">
53           <a href="#" class="-mx-3 block rounded-lg px-3 py-2.5 text-base font-semibold leading-7 text-white hover:bg-gray-800">Log in</a>
54         </div>
55       </div>
56     </div>
57   </div>
58 </div>
59 </header>

```

PageHeader.vue:


```
PageHeader.vue ×
1 <template> Show component usages
2 <div class="text-center py-24">
3   <h1 class="text-8x1 font-bold tracking-widest text-sky-700">Countries App</h1>
4   <h6 class="mt-4 text-1x">App to search countries</h6>
5 </div>
6 </template>
7
```

Index.js:

```
JS index.js ×
1 import { createRouter, createWebHistory } from 'vue-router';
2 import Front from "../views/Front.vue";
3 import countries from "../views/countries.vue";
4
5 const router : Router = createRouter( options: {
6   history: createWebHistory(import.meta.env.BASE_URL),
7   routes: [
8     {
9       path: '/',
10      name: 'Front',
11      component: Front
12    },
13    {
14      path: '/countries',
15      name: 'Countries',
16      component: countries
17    },
18  ],
19 });
20 export default router
```