

Introducción a Vite JS

Actualmente, **Vite** soporta la creación tanto de proyectos que no utilizan frameworks (también conocidos como “vanilla”), como proyectos que utilizan **Vue**, **React**, **Preact**, **Svelte** o **Lit** (tanto usando Javascript como Typescript).

Vite es una herramienta de frontend que te ayudará a crear tus proyectos de forma agnóstica (sin atarte a ningún framework en particular) y que su desarrollo y construcción final sea lo más sencilla y cómoda posible. Está desarrollada por **Evan You**, el creador de **Vue**.

Vamos a crear un proyecto con Vite

En primer lugar, necesitaremos tener instalado **NodeJS**. Si necesitamos descargar o instalar Node.js, podremos hacerlo desde [AQUÍ](#).

Utilizaremos **npm** para comenzar el proyecto. Escribiremos desde una terminal:

```
$ npm create vite@latest mi-proyecto
```

```
Need to install the following packages: create-vite
Ok to proceed? (y) y
```

El comando **npm create vite** en nuestro ejemplo, creará una carpeta llamada **mi-proyecto** que sería la carpeta del proyecto.

Como vemos, nos avisa que es necesario instalar el paquete **create-vite** (en este caso porque no lo tiene instalado. **create-vite** es el asistente encargado de crear la aplicación web.

Al responder afirmativamente (**y**), nos aparecerá un menú para seleccionar que plantilla queremos utilizar (al elegir una, luego nos preguntará si queremos la versión javascript o typescript).

Plantillas de Vite

Si quisiéramos saltar ese menú interactivo, podríamos indicar un parámetro opcional **-template** para especificar directamente la plantilla a utilizar:

```
$ npm create vite mi-proyecto -- --template vanilla
```

```
o
```

```
$ npm create vite mi-proyecto -- --template react-ts
```

Las templates actualmente disponibles pueden ser: vanilla, vue, react, preact, lit o svelte. Y siempre deberemos elegir si queremos la versión Javascript o TypeScript.

Si no se aclara, la versión por defecto es Javascript, mientras que si se agrega **-ts** detrás del template resultará en TypeScript, como puede verse en la segunda línea de comando del cuadro de código anterior.

Hecho esto, veremos que la tarea de creación del proyecto es prácticamente instantánea. Esto es así porque **Vite** no realiza la instalación de paquetes automáticamente, por lo que luego tendremos que hacerlo nosotros manualmente:

```
$ cd mi-proyecto
```

```
$ npm install
```

Ahora sí, ya tendremos el proyecto listo para arrancarlo con el script **npm run dev** y comenzar a desarrollar. Tengamos en cuenta que en todos los proyectos que generemos tendremos un script **dev** que lanza el **modo de desarrollo de Vite** y, por otro lado, tendremos un script **build** que lanza el **modo de producción de Vite**, que es el que genera la subida a la carpeta **dist**, que es la que irá a producción.

Archivo vite.config.js

En nuestros proyectos de **Vite** tendremos un archivo vite.config.js de configuración donde podremos establecer algunos detalles del funcionamiento del empaquetador en el proyecto actual. Su estructura general es la siguiente:

```
import { defineConfig } from "vite"; export
```

```
default defineConfig({
```

```
  root: ".", base:
```

```
  "/", publicDir:
```

```
  "public", build: {
```

```
    outDir: "dist",
```

```
    assetsDir: "assets"
```

```
  },
```

```
  plugins: []
```

```
});
```

Dependiendo de la plantilla utilizada, este archivo de configuración puede ser diferente o, incluso, no existir (como es en el caso de los proyectos de Javascript vanilla).

Referencias: Para seguir estudiando: [guía de Vite](#)