Enter The Vue CLI

Moving to the vue cli, what to expect, how to stay sane and enjoy vue's exciting development environment

Vue cli

Out-of-the-box support for Babel, TypeScript, ESLint, PostCSS, PWA, Unit Testing & End-to-end Testing. The plugin system allows the community to build and share reusable solutions to common needs. Vue CLI is fully configurable without the need for ejecting. This allows your project to stay up-to-date for the long run. Create, develop and manage your projects through an accompanying graphical user interface.

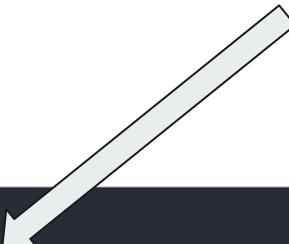Translation - Vue.js on a more complex level but simple if you **follow a guide...**

Step 1 - Install

You will need to install the vue-cli installer on your computer as a global. 'global' means you will be able to access the vue-cli from any project on your computer.

Pls use npm

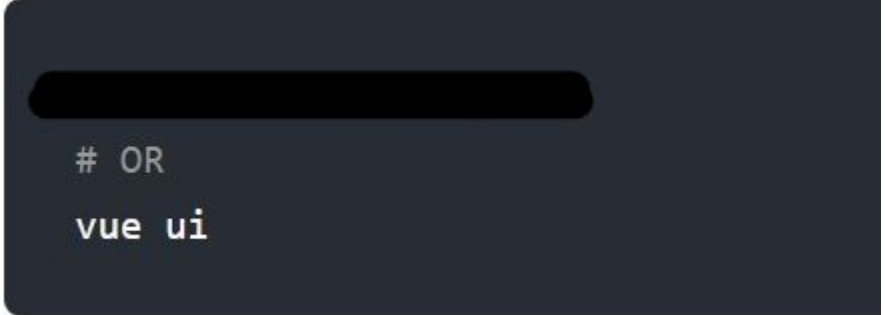Install:

```
npm install -g @vue/cli
# OR
yarn global add @vue/cli
```

Step 2 - Launch Vue's UI tool

Vue has created an awesome GUI tool that allows you to create projects with ease. To use it run **vue ui** after vue has finished installing.
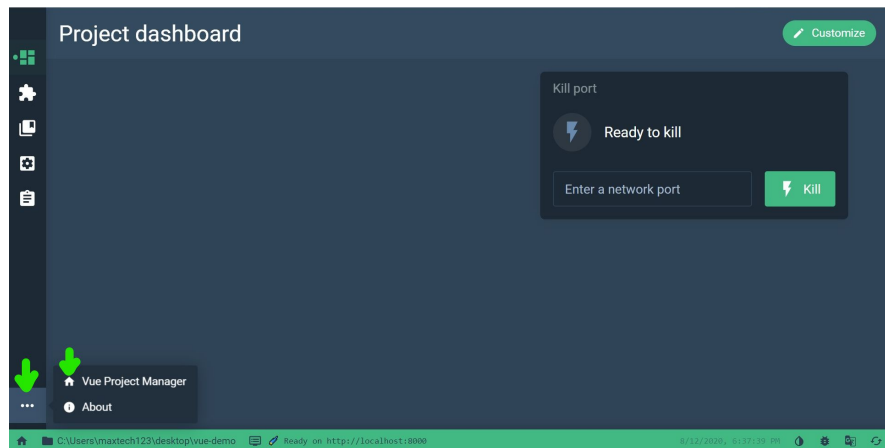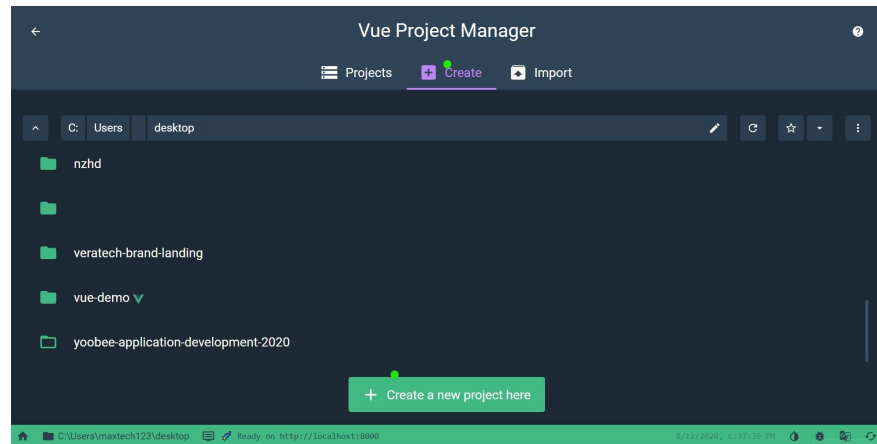
Create a project:

```
# OR
vue ui
```

Step 3 - Launch Vue's UI tool

Vue's ui opens in the browser, to create a new project click on the links indicated in the picture.

Step 4 - Create a new project

Click create then choose a place on your computer to create a new project. Finally click the create a new project button
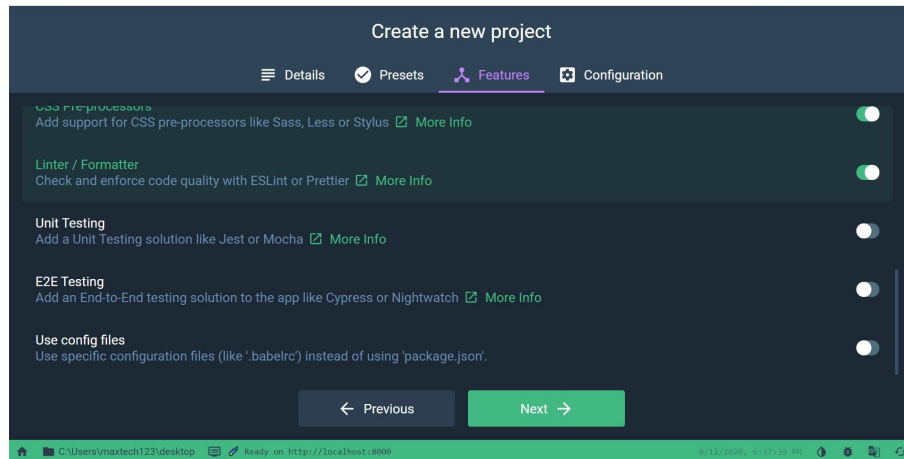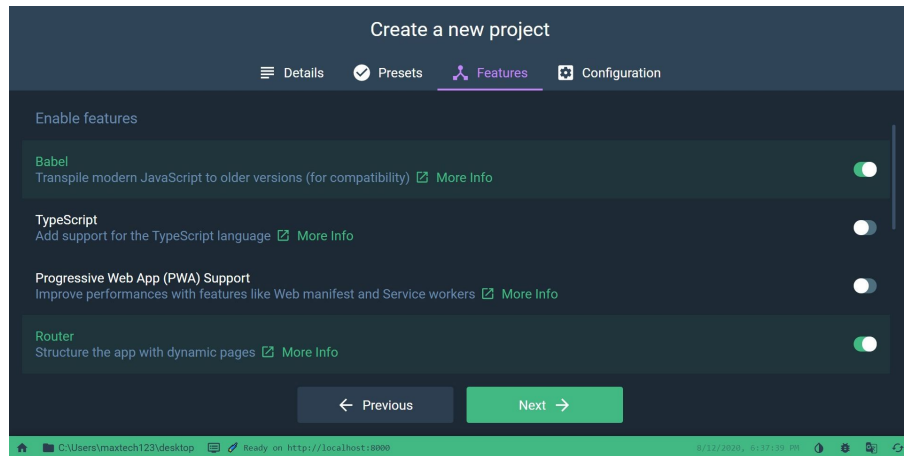
Step 5 - Naming options

Name the folder and select npm as your package manager. Do not worry about the additional options further down the page. Click **next.**
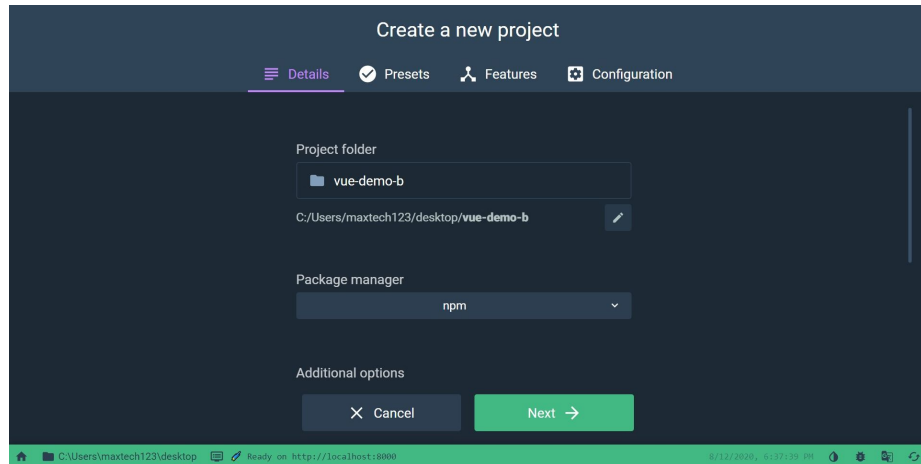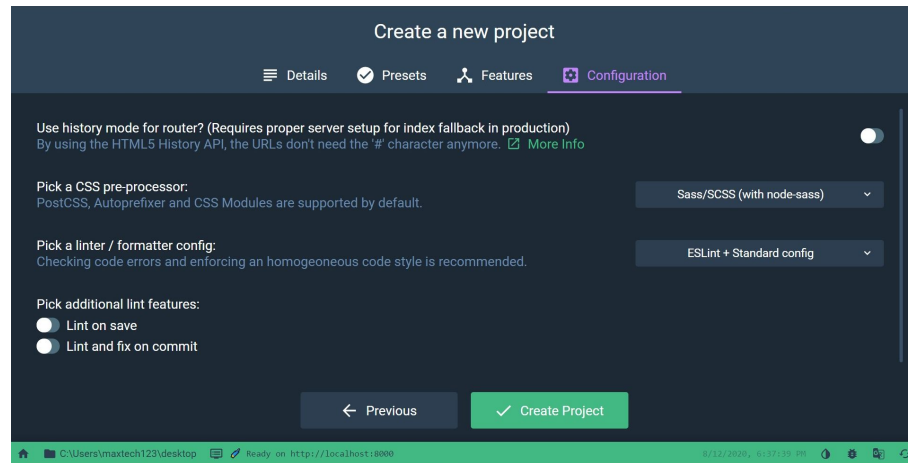
Step 7 - Select Presets

Turn **all options on** *except for Typescript, PWA support, Unit Testing, E2E, and config files === off*.

Step 9 - Select Presets con't

Choose node-sass and you preferred style guide for javascript. Leave all other features the same. Ignore creating a preset name at the next step...

Create a new project

≡ Details   ✓ Presets   ⚹ Features   ⚙ Configuration

Use history mode for router? (Requires proper server setup for index fallback in production)
By using the HTML5 History API, the URLs don't need the '#' character anymore. ↗ More Info

Pick a CSS pre-processor: Sass/SCSS (with node-sass) ⌄
PostCSS, Autoprefixer and CSS Modules are supported by default.

Pick a linter / formatter config: ESLint + Standard config ⌄
Checking code errors and enforcing an homogeneous code style is recommended.

Pick additional lint features:
  ○ Lint on save
  ○ Lint and fix on commit

                                    ← Previous   ✓ Create Project

⌂ ▸ C:\Users\maxtech123\desktop   ▣ ✎ Ready on http://localhost:8000   8/12/2020, 8:37:19 PM
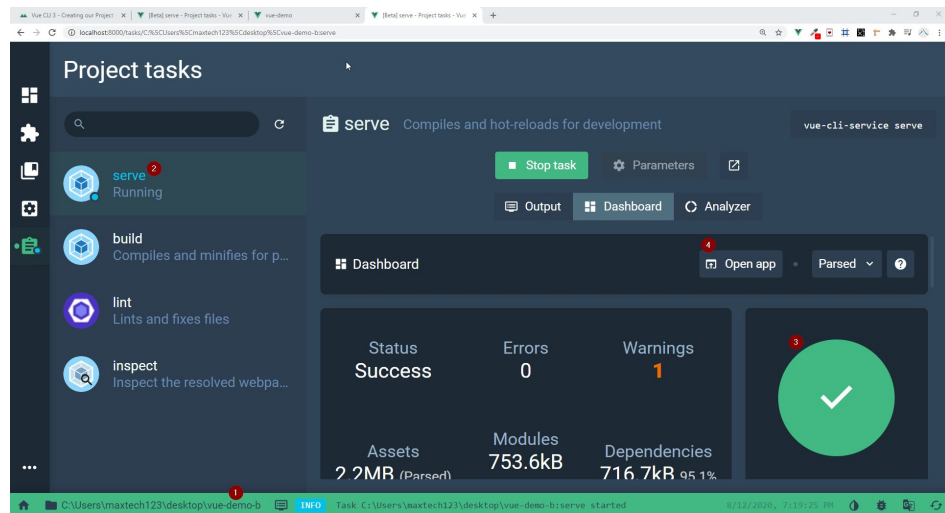
Step 10 - Use project tasks

Choose node-sass and you preferred style guide for javascript. Leave all other features the same. Ignore creating a preset name at the next step...

1. The current dir and application in focus
2. Click to serve the application locally
3. Shows the status of the application serve
4. Click to launch the app on the right port in the browser

Step 11 - Congratulations

You made it, the latest, greatest, modern version of front end development is now installed and ready to roll. Next… how do we actually navigate through it, and use it.
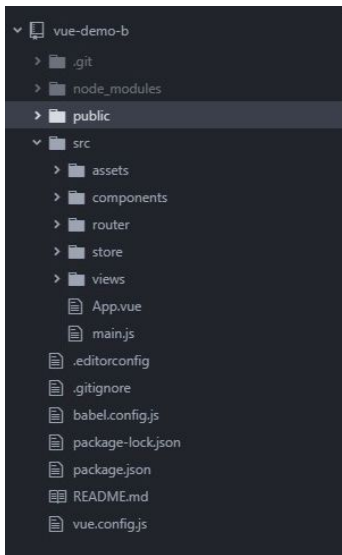
1

Vue's hello world page

Welcome to Your Vue.js App

For a guide and recipes on how to configure / customize this project,
check out the vue-cli documentation.

Installed CLI Plugins

babel   router   vuex   eslint

```
∨ 🖥 vue-demo-b
  > 📁 .git
  > 📁 node_modules
  > 📁 public
  ∨ 📁 src
    > 📁 assets
    > 📁 components
    > 📁 router
    > 📁 store
    > 📁 views
      📄 App.vue
      📄 main.js
    📄 .editorconfig
    📄 .gitignore
    📄 babel.config.js
    📄 package-lock.json
    📄 package.json
    📖 README.md
    📄 vue.config.js
```

2

How it all looks in Atom

Other useful links

Intro Vids (Video 2):
https://www.vuemastery.com/courses/real-world-vue-js/vue-cli

Creating a project:
(Need to do `npm install -g @vue/cli`) first
https://cli.vuejs.org/guide/creating-a-project.html#vue-create