

Introduction

Lux - VueJS Clean Admin Dashboard Template. Built with [Vuetify Components Library](#) (The most popular component library for VueJS), VueCLI, Vue Router and Vuex. It's a next generation, modern and easy to customize template.

Features

- VueJS - Modern Javascript Framework
- Vuex Integration
- Vue Router
- Vue i18n
- Styling with SASS
- Theme with SASS variables
- Fully Responsive
- Dark version very customizable
- Plug and play icon packs
- Extensive UI Component Library
- Modern Calendar Component
- Chat Application Template
- Email Application Template
- ToDo Application Template
- Kanban Application Template
- Authentication Pages / SignIn, SignUp, Forgot Password, Reset Password
- Utility Pages / Maintenance, Coming Soon, Errors
- Ecommerce Pages / List, Details, Orders, Cart
- User Pages / List, Edit
- Dashboard Charts and Widgets Examples
- No JQuery
- RTL (Right To Left)
- ECharts and Apex Charts

Browser Support

This template is built to work with the latest desktop and mobile browsers.

- Chrome
- Firefox
- Edge
- Safari 10+

Requirements

System Requirements


Node.js

Please install the NodeJS from <https://nodejs.org> , LTS version if possible.

Yarn

(optional)

Yarn is dependency manager like NPM.

It's used because of [yarn workspaces](#)  so we install all the packages dependencies without repeating multiple node_modules in our project.

Install Yarn from <https://classic.yarnpkg.com/en/docs/install> .

Packages

Each package is and encapsulated, individual implementation of this template.

Demo

Location: `packages/vue`

Full examples just like the preview for this template.

Starter

Location: `packages/starter`




















Boilerplate version so you can quickly start a fresh new application.

Laravel

Location: `packages/laravel-starter`

Laravel boilerplate version so you can quickly start a fresh new Lux Laravel application.

Libraries

- [VuetifyJS](#) 
- [Vue](#) 
- [Vue-Router](#) 
- [Vuex](#) 
- [Vue-i18n](#) 
- [Vue-head](#) 
- [Vue-emoji-picker](#) 
- [Vue-echarts](#) 
- [Vue-apexcharts](#) 
- [Vue Draggable](#) 
- [Vue Google Tag](#) 
- [Vue2-google-map](#) 
- [Vue Shortkey](#) 
- [Flag Icon Css](#) 
- [Animate.css](#) 
- [Tiptap](#) 
- [Moment](#) 
- [Lodash](#) 
- [ApexCharts](#) 

Installation

- [Vue - Lux Admin](#)
 - [Install Dependencies](#)
 - [Development Server](#)
- [Laravel 8 - Lux Admin](#)
 - [Laravel Installation and dependencies](#)
 - [Quick Start](#)
 - [Development](#)
 - [Relevant folders to implement Lux on Laravel 8](#)

Vue - Lux Admin

Prerequisites

Make sure you have [Node.js](#) and [NPM CLI](#) installed.

We recommend [Yarn](#) for ease of installation.

Get the template

Download the template .zip and unzip to a folder of your choosing.

Inside that folder we will need to install the dependencies before running the development server.

TIP

There are dot files inside the folders like `.gitignore` , and when moving folders sometimes they are left behind, make sure to use the correct method to copy all the files.

Install Dependencies

(If you installed [Yarn](#)) Install all the dependencies for all the packages

```
yarn
```

sh

Or install only the dependencies for a specific package with NPM:

```
cd packages/vue
# or
# cd packages/starter

# install dependencies
npm install
```

sh

Development Server

On each package it's possible to start the respective development server with the command:

```
cd packages/{ package name }

# run the development server
yarn dev
# or for npm
npm run dev
```

sh

Open <http://localhost:8080> in the browser.

WARNING

Your app may run on a different port than 8080 because it could be already in use by other application. To be sure, you can check correct open port in the terminal you ran the command

```
yarn dev || npm run dev .
```

Laravel 8 - Lux Admin

packages/laravel-starter

Laravel Installation and dependencies

See: <https://laravel.com/docs/8.x>

Quick Start

For all the dependencies that need to be setup PHP and Laravel, please consult the [Laravel Documentation](#)

```
cd packages/laravel-starter

# install laravel dependencies
composer install

# install package.json dependencies
yarn # or npm install
```

sh

Development

On one terminal:

```
php artisan serve
```

sh

And on another terminal:

```
yarn watch
```

sh

Open <http://127.0.0.1:8000> in the browser.

Your app may run on a different port than 8000 because it could be already in use by other application. To be sure, you can check correct open port in the console you ran the command `php artisan serve` .

Relevant folders to implement Lux on Laravel 8

The files and folders listed bellow are the places changed to make the Vuetify / Laravel integration work.

Files	Description
package.json	NPM Dependencies
webpack.mix.js	Webpack Compile Frontend Assets
resources/js/*	Vuetify Application
resources/sass/*	Vuetify Variables and overrides
resources/view/spa.blade.php	Basically Index.html
app/Http/Controllers/SpaController.php	Serve spa.blade.php
routes/web.php	(SPA) Redirect all requests to SpaController

Folder Structure

vue	
public	
images	
favicon.ico	
index.html	-> Main HTML
robots.txt	
src	
apps	
board	
components	-> Board App – Components
pages	-> Board App – Pages
routes	-> Board App – Router
store	-> Board App – Store
BoardApp.vue	-> Board App Layout
chat	-> Chat App
email	-> Email App
todo	-> Todo App
assets	
scss	-> Styles
vuetify	-> Vuetify Variables and overrides
theme.scss	-> Main Theme SCSS
components	-> Custom Theme Components
configs	-> Theme Configurations
filters	-> Filters
layouts	-> Application Layouts
pages	
auth	-> Auth Pages – SignIn, SignUp, etc
dashboard	-> Dashboard Pages – Analytics Pages
error	-> Error Pages – 404, 500, etc
ui	-> UI Pages – Demo with examples
users	-> Users Pages – List, edit
utility	-> Utility Pages – Maintenance, coming s
BlankPage.vue	-> Blank page
plugins	-> Vue Plugins
router	-> Routes for Vue-Router
store	-> Main Vuex Store
translations	-> I18n Translations
App.vue	-> Main Application Vue
main.js	-> Main Application JS
.eslintrc.js	-> ESLint Configuration
.gitignore	-> Git Ignore
README.md	-> Readme
babel.config.js	-> Babel Configurations
package.json	-> NPM Package Information
vue.config.js	-> VueCLI Configuration File

Template Configurations

There are many things you can change on this template, from the primary color to the layout or RTL setup. Checkout bellow what are the available configurations:

The configurations are all located under the folder: `src/configs`

theme.js

globalTheme

- Type: `'light' | 'dark'`
- Default: `'light'`

Defines the global theme colors for the application and all it's components

menuTheme

- Type: `'global' | 'light' | 'dark'`
- Default: `'global'`

Theme colors for the side menu

toolbarTheme

- Type: `'global' | 'light' | 'dark'`
- Default: `'global'`

Main toolbar theme colors

isToolbarDetached

- Type: `boolean`
- Default: `false`

Detach toolbar from top

isContentBoxed

- Type: `boolean`
- Default: `false`

Wraps pages content with a max-width

isRTL

- Type: `boolean`
- Default: `false`

Application is right to left

dark

- Type: `Object`

Dark theme colors

light

- Type: `Object`

Light theme colors

navigation.js

menu

- Type: `Array<Object>`

Side menu (main menu)

footer

- Type: `Array<Object>`

Side menu footer links

toolbar.js

apps

- Type: `Array<Object>`

Quickmenu for applications

user

- Type: `Array<Object>`

User dropdown menu

currencies.js

currency

- Type: `Object`

Starting currency

availableCurrencies

- Type: `Array<Object>`

Available currencies to select on the application

locales.js

locale

- Type: `Object`

Starting locale

fallbackLocale

- Type: `string`

When translation is not available fallback to that locale

availableLocales

- Type: `Array<Object>`

Available locales to select on the application

analytics.js

gald

- Type: `String`
- Env: `VUE_APP_GMAPS_KEY`

Google analytics id

maps.js

key

- Type: `String`
- Env: `VUE_APP_GA_ID`

Google Maps API Key

time.js

Time format display to use on the application by zone

icons.js

Array with all the icon libraries being used on the application. They will by automatically pushed to the `index.html` header at the application start.

Internationalization

Translation Files

This template uses the plugin [vue-i18n](#).

The translation files are located in `src/translations`, they are simple objects that can be used with ease in the Vue components.

To add or remove locales just go to the `src/configs/locales.js` and adjust what are the languages you wish to display.

Configurations

Locales (translations) - `src/configs/locales.js`

Property	Type	Description
locale	String	Pre selected locale
fallbackLocale	String	When translation is not available fallback to that
availableLocales	Array	Locales to select on the application

Time - `src/configs/time.js`

Time format display to use on the application by zone

Style Customizations

A more deep customization would be looking and changing the actual SASS variables that will reflect on the Vuetify build.

They can be found in: `assets/scss/vuetify/variables`

Global - `_global.scss`

From border radius root size to small corrections on Vuetify components, that's where you will find them.

Light Theme - `_light.scss`

Light mode theme colors and variables.

Dark Theme - `_dark.scss`

Dark mode theme colors and variables.

Font - `_font.scss`

Fonts and typography styling.

Mixins - `_mixins.scss`

Mixins helpers - Media queries for the Vuetify display breakpoints.

Shadows - `_elevations.scss`

Editing will change how the vuetify elements cast their shadow and the class `elevation- $\{n\}$` will affect elements.

Additional Class Helpers - `_utilities.scss`

Extra css classes to help you build your application

Build and deploy

Build

Command to create the distribution build of the Vue Application:

From the root of the template:

```
# Compiles and minifies for production on `package/vue`
yarn build:vue
# or for npm
npm run build:vue

# Compiles and minifies for production on `package/starter`
yarn build:starter
# or for npm
npm run build:starter
```

sh

To build a specific package just:

```
cd package/{name}

yarn build
# or for npm
npm run build
```

sh

This command will generate the `dist` folder and that's the distribution build. The compacted application ready to be served from an HTTP server.

Deploy

After you get the `dist` build you just need to:

- Upload it to the HTTP server you wish to use
- Configure the server in order to rewrite all requests (not file requests) to the `index.html` so it is served as a SPA and it's handling all the requests from the clients.

RTL

To use Right To Left in this template you just need to:

- Open `src/configs/theme.js`
- Edit `isRTL` to true and save

Components

What are the possible configurations for the components?

VuetifyJS Components

For an extensive guide on the components, please visit <https://vuetifyjs.com>[↗].

Linting

Disable linting

To remove ESLint validations from the project

```
# in the folder root project you want to disable
yarn remove eslint-plugin-vue @indielayer/eslint-config-vue

# or if you use NPM
npm remove eslint-plugin-vue @indielayer/eslint-config-vue
```

sh

More information

For more information on the rules, please visit <https://github.com/indielayer/eslint-config> 

For more documentation on ESLint, please visit <https://eslint.org/> 

Robots.txt

File: `public/robots.txt`

Disallow all robots (default)

To exclude all robots from the entire website

```
User-agent: *  
Disallow: /
```

Allow all

To allow all robots complete access to crawl your website

```
User-agent: *  
Disallow:
```

More information

For more documentation on robots.txt, please visit <https://www.robotstxt.org/robotstxt.html> 

Support

Support is available to help and answer your questions via:

- Email at support@indielayer.com
- Discord channel **#premium-theme-support** at <https://community.vuetifyjs.com/> 