# - Laravel With Npm Using Admin lte template-----------

**Instruction (installation):**

**1** install laravel using this command (via Composer ): (download composer https://getcomposer.org/)

composer create-project --prefer-dist laravel/laravel blog

**2** And run this code

npm install (this command help ho install all depandancies: package.json)

**3** Now we are using adminlte template

(note: template boostrap and laravel using boostrap version should be same? If not then your design while be mess up, we use starter page : https://adminlte.io/themes/dev /AdminLTE/starter.html)

Npm install [admin-lte@v3.0.0-alpha.2](mailto:admin-lte@v3.0.0-alpha.2) --save (<https://adminlte.io/docs/2.4/installation>)

(**note:**  run this code **npm run watch** this code compile all file and dependences and build one block and run pragram )

**4** Now your server (php artisan serve) and also create auth

**5** Now put this code on **app/providers/appServiceProvider**

public function boot()

{

Schema::DefaultStringLength(191);

}

**6** Now migrate your database

php artisan migrate

**7** And also create auth to access backend

Php artisan make:auth

**8** error in console

<https://laravel.com/docs/csrf#csrf-x-csrf-token>

To solve it just copy this code on master.blade.php

<!-- CSRF Token -->

<meta name="csrf-token" content="{{ csrf\_token() }}">

# - Now Adding css and js file in a Template-----------

<link rel="stylesheet" href="/css/app.css">

<script src="/js/app.js"></script>

Here all css and js file is in this two file. (create master.blade.php file and copy paste starter.html code to master.blade.php and delete all css and jst link and copy paste that tow css js file on master page)

**1** For **js** just copy and paste this code to boostrap.js file (resources/assets/js:boostrap.js)

require('admin-lte'); this name should be same as in the dependences name

**2** For **css** file using this code on app.scss (resources/sass:app.scss)

@import '~admin-lte/dist/css/adminlte.css';

**3** Calling images (create img folder on your public folder)

./img/logo.png

**4**

# - Adminlte dependencies -----------

**1 fontawesome**

**To** use **fontawesome** we have to install it on our project using **npm** by this code

npm install @fortawesome/fontawesome-free (https://fontawesome.com/how-to-use)

(**note**: now its install you can see it on package.json file )

**2** how to use it

Copy this code to your **app.scss (**resources/sass/app.scss**)**

$fa-font-path:"../webfonts";

//fontawesome

@import "~@fortawesome/fontawesome-free/scss/fontawesome.scss";

@import "~@fortawesome/fontawesome-free/scss/solid.scss";

@import "~@fortawesome/fontawesome-free/scss/brands.scss";

(**note:**  now fontawesome work)

# -------- Vue component ---------------------------------

**Installing Vue component by npm (**https://router.vuejs.org/installation.html#direct-download-cdn**)**

**1** npm install vue-router

(**note:** run this code **npm run watch.**  This code recompile all file and built one file and run program)

**2** <router-view></router-view>

Copy this code and paste into master.blade.php

(**note:** component matched by the route will render here)

**3** importing (resources/js/app.js)

window.Vue = require('vue');

import VueRouter from 'vue-router'

Vue.use(VueRouter)

**3** Configure the JavaScript or route configuration (resources/js/app.js)

let routes = [

{ path: '/dashboard', component: require('./components/Dashboard.vue') },

{ path: '/developer', component: require('./components/Developer.vue') },

{ path: '/users', component: require('./components/Users.vue') },

{ path: '/profile', component: require('./components/Profile.vue') }

]

**4** Register Route (resources/js/app.js)

const router = new VueRouter({

routes // short for `routes: routes`

})

**5**  At last (add router ) app.js

const app = new Vue({

el: '#app',

router

});

(**note:** here **#app**  is address where router is looking so)

In master page add id app like this

<div class="wrapper" id="app">

.

.

<router-view></router-view>

.

.

</div><!-- /.wrapper -->

**Now it work**

**6** How to use it, working with route (master.blade.php)

<router-link to="/dashboard" class="nav-link">

<i class="nav-icon fas fa-tachometer-alt blue"></i>

<p>

Dashboard

</p>

</router-link>

(**note**: **a:router** **href:to** and **dashboard** is route name)

**7** Managing **url**

For this we have to mode history in **app.js** here is code

const router = new VueRouter({

mode:'history', //pre home/profile now only /profile

routes // short for `routes: routes`

})

(**note: before** : localhost:8000#/profile , **after:** localhost/profile)

**But** here is problem if we refresh it, it’s not work

**To** solve it

//this when u copy url and paste they show error page but with the help of this its work.

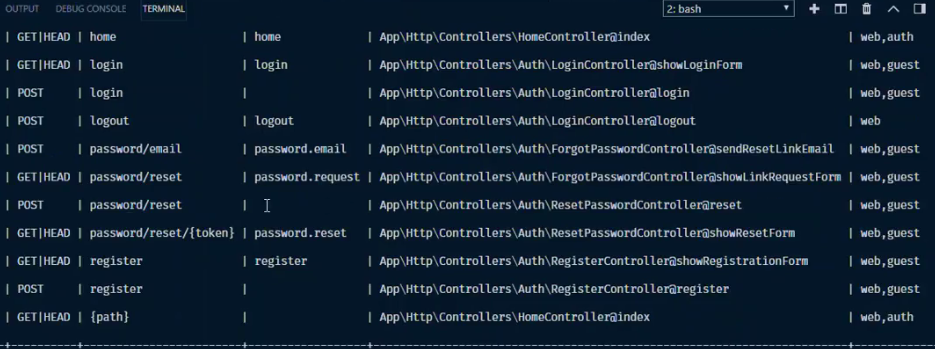
**Route::get('{path}',"HomeController@index")->where( 'path','([A-z\d-\/\_.]+)?' );**

Now its sloved

# ------ Crud Operation Vue component-----------------------------------

**Run this code**

**php artisan route:list**



**(note:**  this list help which **action** is execute on which **function)**

**For logout** -> it should be post request to the logout **url**

**1 logout**

***Here is a code***

<li class="nav-item">

<a class="nav-link" href="{{ route('logout') }}"

onclick="event.preventDefault();

document.getElementById('logout-form').submit();">

<i class="nav-icon fa fa-power-off red"></i>

<p>

{{ \_\_('Logout') }}

</p>

</a>

<form id="logout-form" action="{{ route('logout') }}" method="POST" style="display: none;">

@csrf

</form>

</li>

# -------Other Dependences-----------------------------------

**1 vform (**search **vform** on github.com result -> github.com/cretueusebiu/vform**)**

**(vform:** ajax form with validation and api features and many more**)**

**1 To install**

npm i axios vform

**2 importing**

Paste this code after the window.vue like this

import Vue from 'vue' or window.Vue = require('vue');

import { Form, HasError, AlertError } from 'vform'

window.Form = Form;

Vue.component(HasError.name, HasError)

Vue.component(AlertError.name, AlertError)

**3 How to use users.vue**

i<div class="form-group">

           <textarea v-model="form.bio" name="bio"

           placeholder="short bio for user (optinal)"

           class="form-control" :class="{ 'is-invalid': form.errors.has('bio') }"></textarea>

           <has-error :form="form" field="bio"></has-error>

           </div>

export default {

data() {

return{

// this new Form is instance from window.Form = Form

form: new Form({

id: '',

name: '',

email: '',

password: '',

type: '',

bio: '',

photo: ''

})

}

},

(**note:** v-model =”form.bio” => data(){…

form:new Form({id:’’,name:’’,bio:’’,})})

**5 sending Resources or data, create function**

<form @submit.prevent="createUser">

**And**

Create **createUser** function

Methods:{

createUser(){

this.form.post('api/user')

//there .then if success case and .catch is if any error occure

.then(()=>{

Fire.$emit('AfterCreated'); // creating custom event

$('#addNew').modal('hide') // this is simple method

toast({

type: 'success',

title: 'created successfully'

})

this.$Progress.finish();

})

.catch(()=>{

});

}

}

**And**

Go to the routes/api.php

Route::middleware('auth:api')->get('/user', function (Request $request) {

return $request->user();

});

**And** create api controller **usercontroller** (cause we using api modal )

php artisan make:controller API/UserController –api

**And copy this code to api.php**

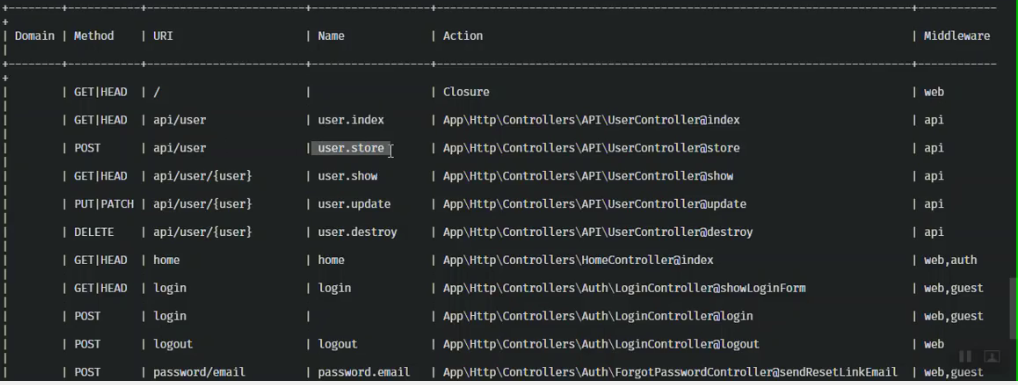
// we create api controller using this command php artisan make:controller API/UserController --api

//thats why we are using this type of route

//route value should be in array couse it only accept array value in api mode

Route::apiResources(['user'=>'API\UserController']);

**(note:**  this list help which **action** is execute on which **function)**

**Here,**  to save data we use **@store** function app/http/***Controller/API/userController***

use App\User; // mvc we have to add this line of code use App\User;

use Illuminate\Support\Facades\Hash; // using Hash fuciton on psw so we have to call it

public function store(Request $request)

{

// validation

$this->validate($request,[

'name' => 'required|string|max:191',

'email' => 'required|string|email|max:191|unique:users',

'type' => 'required|string|max:20',

'password' => 'required|string|min:6'

]);

// checking weather it is working or not

//return['message'=>'i have your data'];

//return $request->all(); // it is show all data

return User:: create([

'name' => $request['name'],

'email' => $request['email'],

'type' => $request['type'],

'bio' => $request['bio'],

'photo' => $request['photo'],

'password' => Hash::make($request['name']),

]);

}

**BUT BEFORE THIS app/User.php**

protected $fillable = [

'name', 'email', 'password', 'bio', 'photo', 'type'

];

# -------Your Database -----------------------------------

**1** database name is

laravel\_npmAdminlte