**Video 1(install)**

1.Define a Route:

{

Path:’/home’

Component:Home

}

2.Render the router:

<router-view></router-view>

3.Manual navigation:

Inside a vue component:

this.$router.push(‘/home’)//navigate to <http://yourdomain.com/home>

this.$router.back()//go back in the browser history back

**Vuex**

Spa state for each component and way to manage this state inside component and many piece of state component to be used

so vuex is used to store centalized this state so it could be shared among many components more than once.

Working:in vue component it dispatch an **action** it talk to backend API do calculation and then send a commit and this commit will **mutate** the state and override the state with new instances then inside component **getters** are used to get values from this piece of state.

Eg:

Import Vuex from ‘vuex’;

Const store = new Vuex.Store({

State:{},

Getters:{},

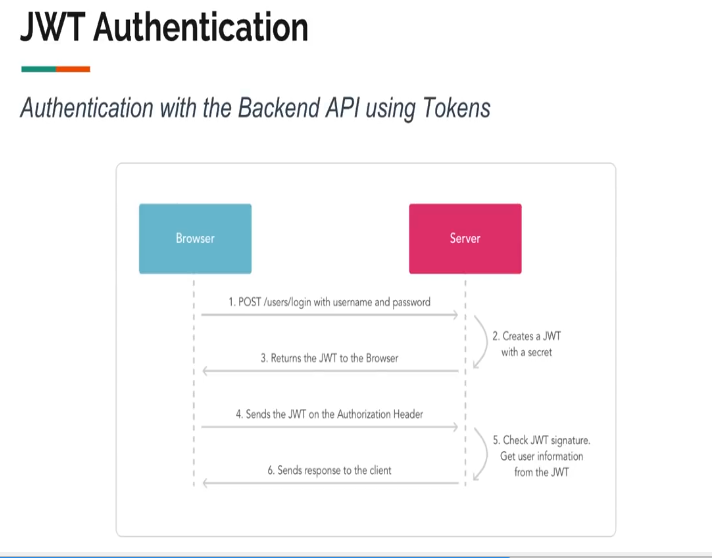
Actions:{},

Mutation:{}

})

**JWT Authentication**

Communicate with backend Api



**Validate.JS**

Javascript validation library

const constraints = { //constraints will get back error

name:{

presents: true

},

email:{

presense:true,

email:true//email is true other wise it will pass error

}

}

Import validate from ‘validate.js’

const errors = validate(data,constraints); //constraints will get error above

If(errors){

//passed

}

**Video 2**

1. **npm install --save vuex vue-router validate.js**
2. **npm i**
3. remove div and style part from welcome.php
4. resources->assets->js->bootstrap.js
5. resources->assets->js->app.js
6. js->route.js
7. ExampleComponent.vue change to MainApp.vue
8. npm run watch
9. Create Js->store.js

**Video 3**

1. Table migration
2. Jwt Authentication
3. Tymondesigns/jwt-auth
4. Quickstart->user model
5. Every request in server will go through web.php
6. RouteServiceProvider-> mapApiRoutes() request through api so url: Vuelar.dev/api/…..

‘middleware- = > ‘api’ no needed already declared in RouteServiveProvider.

1. Copy everything from jwt page and now add in AuthController.php and add api in **auth('api')** everywhere

protected function respondWithToken($token)

    {

        return response()->json([

            'access\_token' => $token, //after user authenticated user will have return with user data it have below code added

            'user' => $this->guard()->user(),//added does same thing like expire\_in in last

            'token\_type' => 'bearer',

            'expires\_in' => auth('api')->factory()->getTTL() \* 60  //api added

        ]);

    }

    //added override

    public function guard()

    {

        return Auth::Guard('api');

    }

In Command prompt

PS C:\wamp64\www\Vuelar> php artisan tinker

Psy Shell v0.9.12 (PHP 7.3.5 — cli) by Justin Hileman

>>> $user = new App\User;

=> App\User {#3131}

>>> $user->name = "Test"

=> "Test"

>>> $user->email = "test@gmail.com"

=> "test@gmail.com"

>>> $user->password = bcrypt('123456')

=> "$2y$10$w/OnQV80TtwrN7nrgWEk8e50KfXqI9H0UM94n561HmBZ9FcZ6P2DW"

>>> $user->save()

In **Postman** <Url:Post>localhost:8000/Vuelar.dev/api/auth/login

Body: {

    "email": "test@gmail.com",

    "password": "123456"

}

From bootstrap.js copy to Header:key-X-Requested-With value-XMLHttpRequest

Error: message :Auth not found

Sol: \Auth in AuthController.php in

public function guard()

    {

        return Auth::Guard('api');

    }

**Video 4(API and Vuex)**

1. Create asset->js->helpers->auth.js