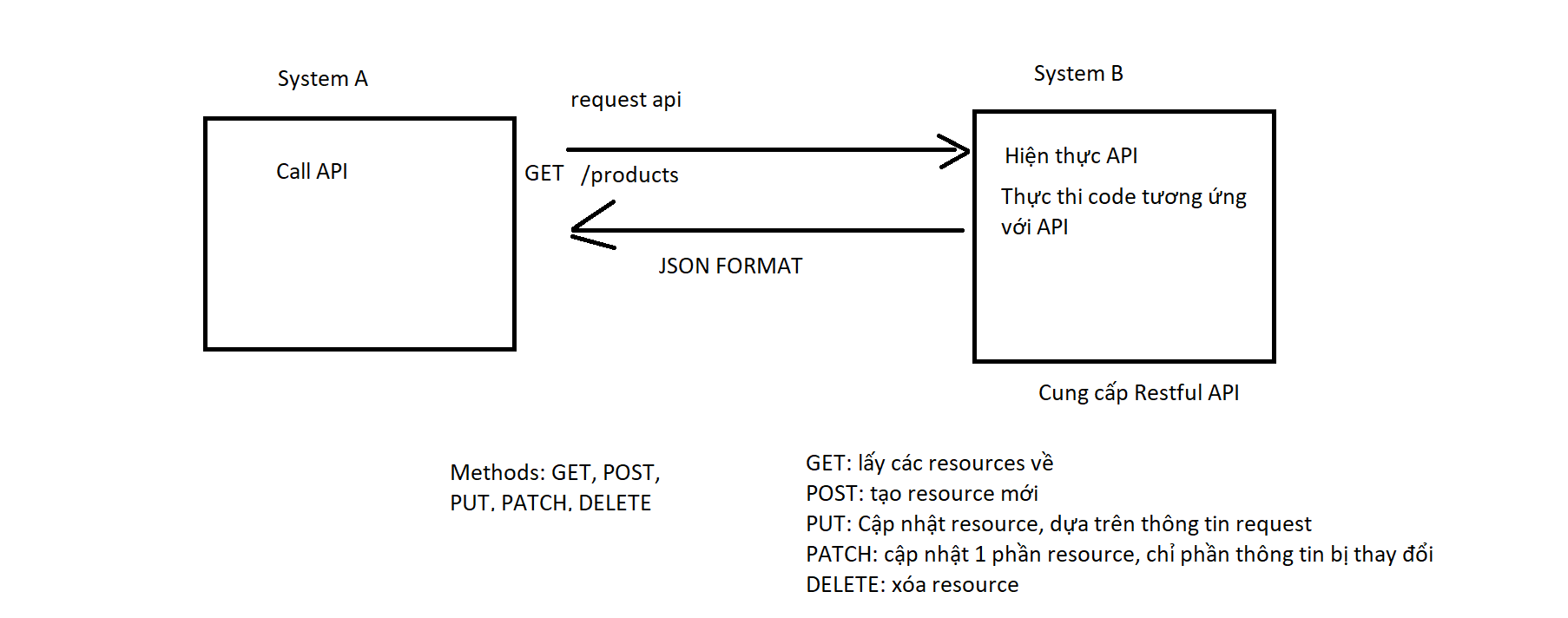
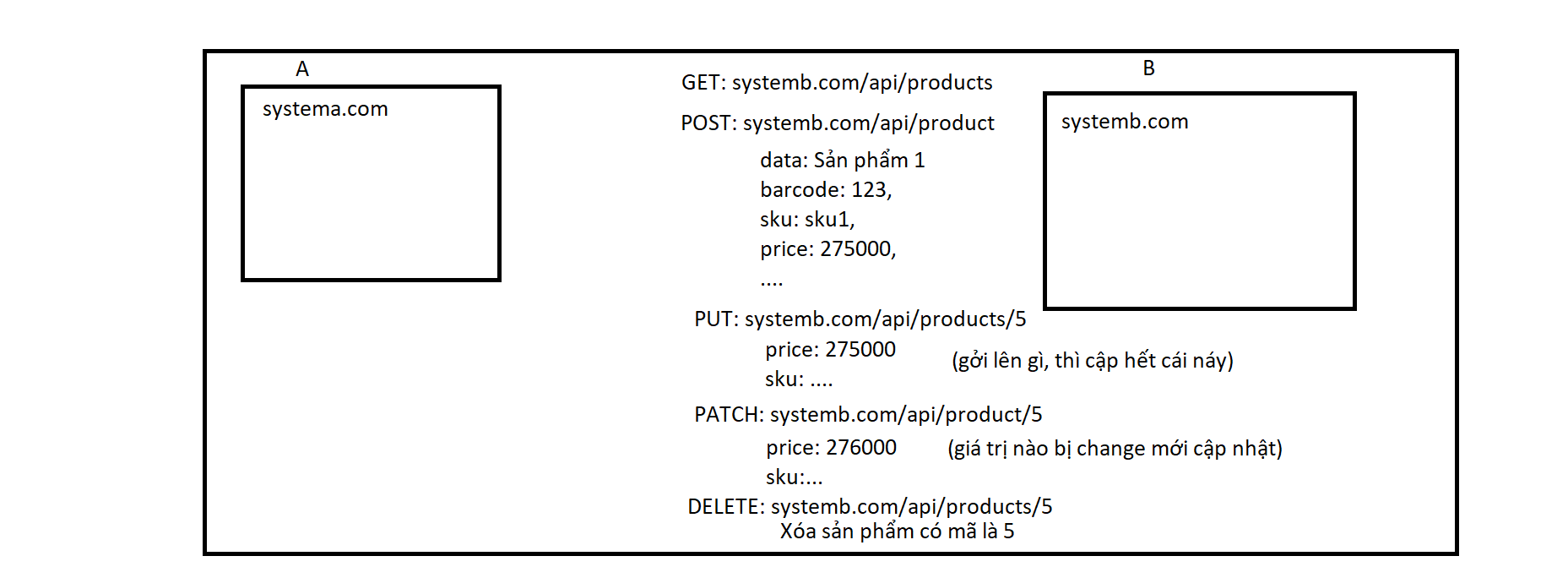
**RESTFUL API**

****

****

Tạo project qlsvapi bằng dòng lệnh bên dưới

composer create-project --prefer-dist laravel/laravel=8.5.9 qlsvapi

Tạo api.laravel.qlsv.com và trỏ vào thư mục qlsvapi (dùng virtual host)

Tạo database api-laravel-qlsv-k37 (lưu ý k37 sẽ thay đổi theo khóa học)

Chỉ định database trong file .env

DB\_CONNECTION=mysql

DB\_HOST=127.0.0.1

DB\_PORT=3306

DB\_DATABASE=api-laravel-qlsv-k37

DB\_USERNAME=root

DB\_PASSWORD=

Có 3 cách xác thực : HTTP Basic, OAuth2, JWT (Json Web Token)

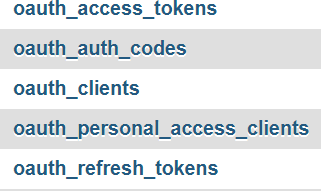
Sử dụng package passport để xác thực OAuth2

composer require laravel/passport

Chạy lệnh: php artisan migrate

Nó sẽ migrate trong thư mục vendor\laravel\passport\database\migrations

Sau khi migrate ta được kết quả



Vào config/app.php Add thêm dòng bên dưới vào providers

Laravel\Passport\PassportServiceProvider::class

Tiếp tục chạy lệnh php artisan passport:install

Nó sẽ tạo ra 2 khóa bảo mật và lưu trong cột secret bảng oauth\_clients

Trong file app/Models/User.php

use Laravel\Passport\HasApiTokens;

Trong class User của file app/Models/User.php

use HasFactory, Notifiable, HasApiTokens;

Trong app\Providers\AuthServiceProvider.php

<?php

namespace App\Providers;

use Illuminate\Foundation\Support\Providers\AuthServiceProvider as ServiceProvider;

use Illuminate\Support\Facades\Gate;

use Laravel\Passport\Passport;

class AuthServiceProvider extends ServiceProvider

{

    /\*\*

     \* The policy mappings for the application.

     \*

     \* @var array

     \*/

    protected $policies = [

        'App\Models\Model' => 'App\Policies\ModelPolicy',

    ];

    /\*\*

     \* Register any authentication / authorization services.

     \*

     \* @return void

     \*/

    public function boot()

    {

        $this->registerPolicies();

        Passport::routes();

    }

}

Trong file config/auth.php, cập guards:api từ token thành passport

'guards' => [

        'web' => [

            'driver' => 'session',

            'provider' => 'users',

        ],

        'api' => [

            'driver' => 'passport',

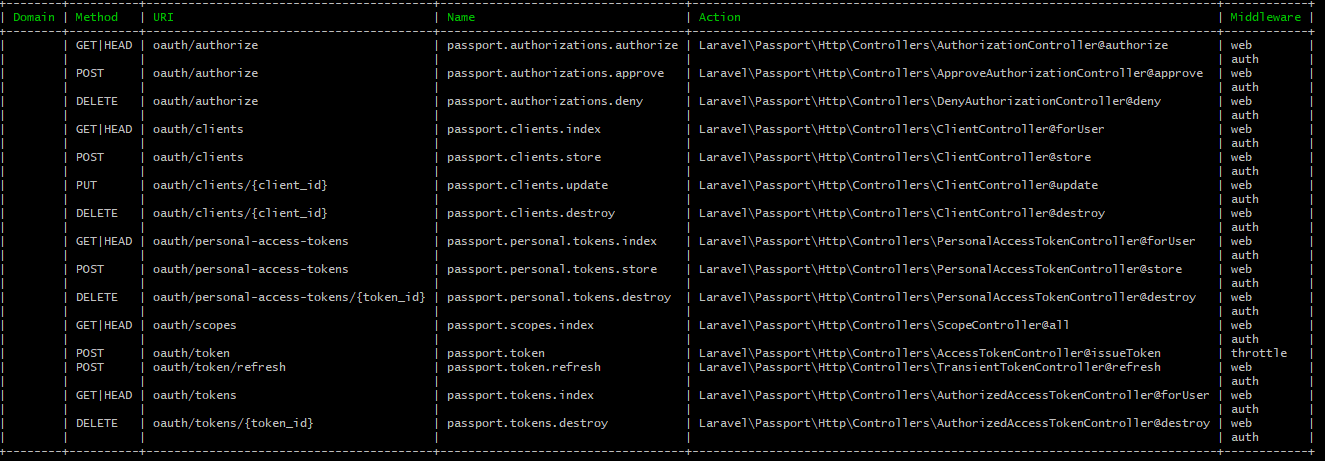
            'provider' => 'users',

            'hash' => false,

        ],

    ],

php artisan route:list --name="passport"



Tạo app\Http\Controllers\API\UserController.php

<?php

namespace App\Http\Controllers\API;

use App\Http\Controllers\Controller;

use App\Models\User;

use Illuminate\Http\Request;

use Illuminate\Support\Facades\Auth;

use Validator;

class UserController extends Controller

{

    public $successStatus = 200;

/\*\*

 \* login api

 \*

 \* @return \Illuminate\Http\Response

 \*/

    public function login()

    {

        if (Auth::attempt(['email' => request('email'), 'password' => request('password')])) {

            $user = Auth::user();

//Create token with name is MyApp

            $success['token'] = $user->createToken('MyApp')->accessToken;

$success['name'] = $user->name;

            return response()->json(['success' => $success], $this->successStatus);

        } else {

            return response()->json(['error' => 'Unauthorised'], 401);

        }

    }

/\*\*

 \* Register api

 \*

 \* @return \Illuminate\Http\Response

 \*/

    public function register(Request $request)

    {

        $validator = Validator::make($request->all(), [

            'name' => 'required',

            'email' => 'required|email',

            'password' => 'required',

            'confirm\_password' => 'required|same:password',

        ]);

        if ($validator->fails()) {

            return response()->json(['error' => $validator->errors()], 401);

        }

        $input = $request->all();

        $input['password'] = bcrypt($input['password']);

        $user = User::create($input);

        $success['token'] = $user->createToken('MyApp')->accessToken;

        $success['name'] = $user->name;

        return response()->json(['success' => $success], $this->successStatus);

    }

/\*\*

 \* user api

 \*

 \* @return \Illuminate\Http\Response

 \*/

    public function user()

    {

        $user = Auth::user();

        return response()->json(['success' => $user], $this->successStatus);

    }

}

Trong file routes\api.php

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

//     return $request->user();

// });

Route::post('login', UserController::class.'@login');

Route::post('register', UserController::class.'@register');

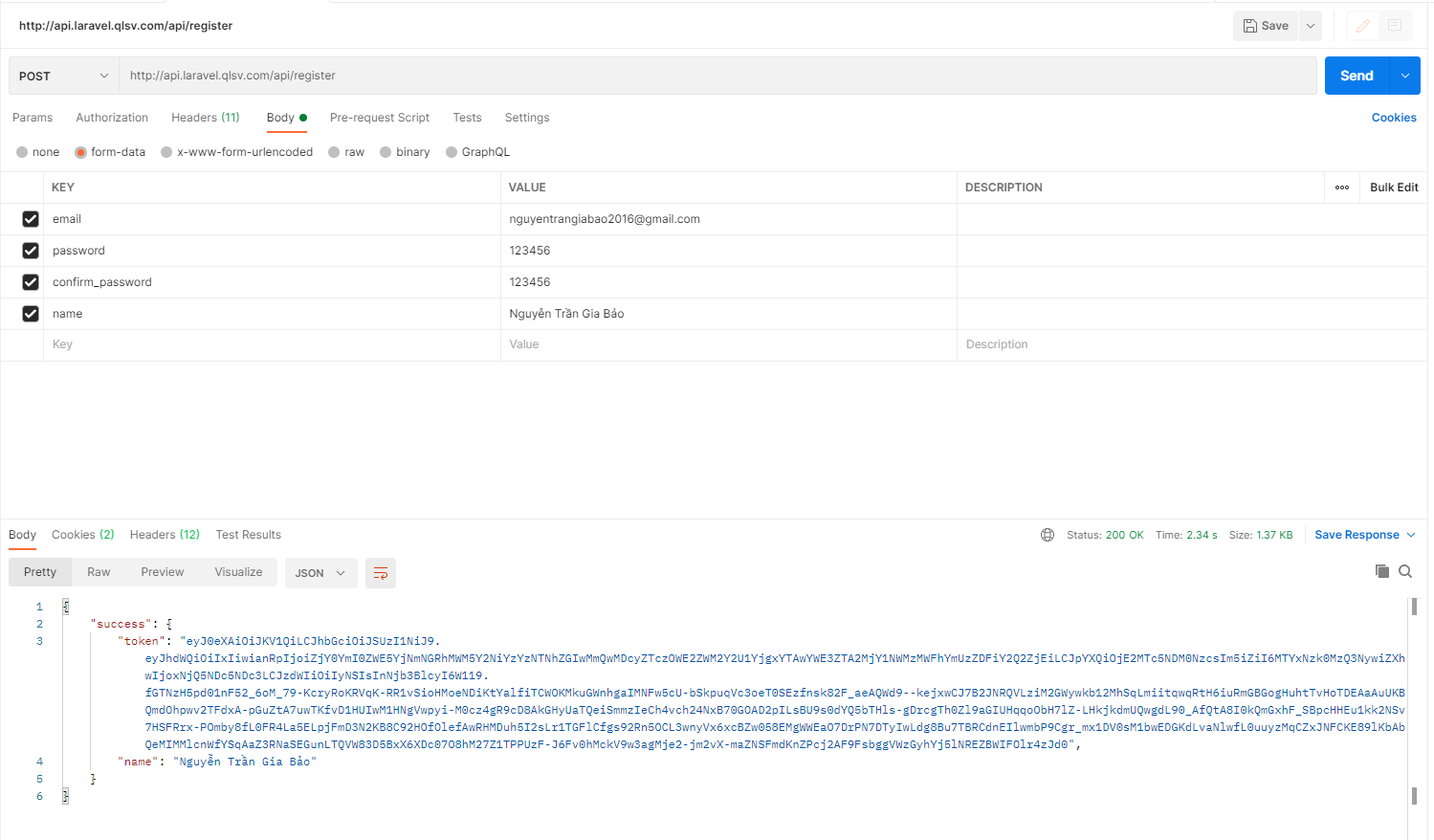
Route::group(['middleware' => 'auth:api'], function() {

    Route::get('/user', UserController::class.'@user');

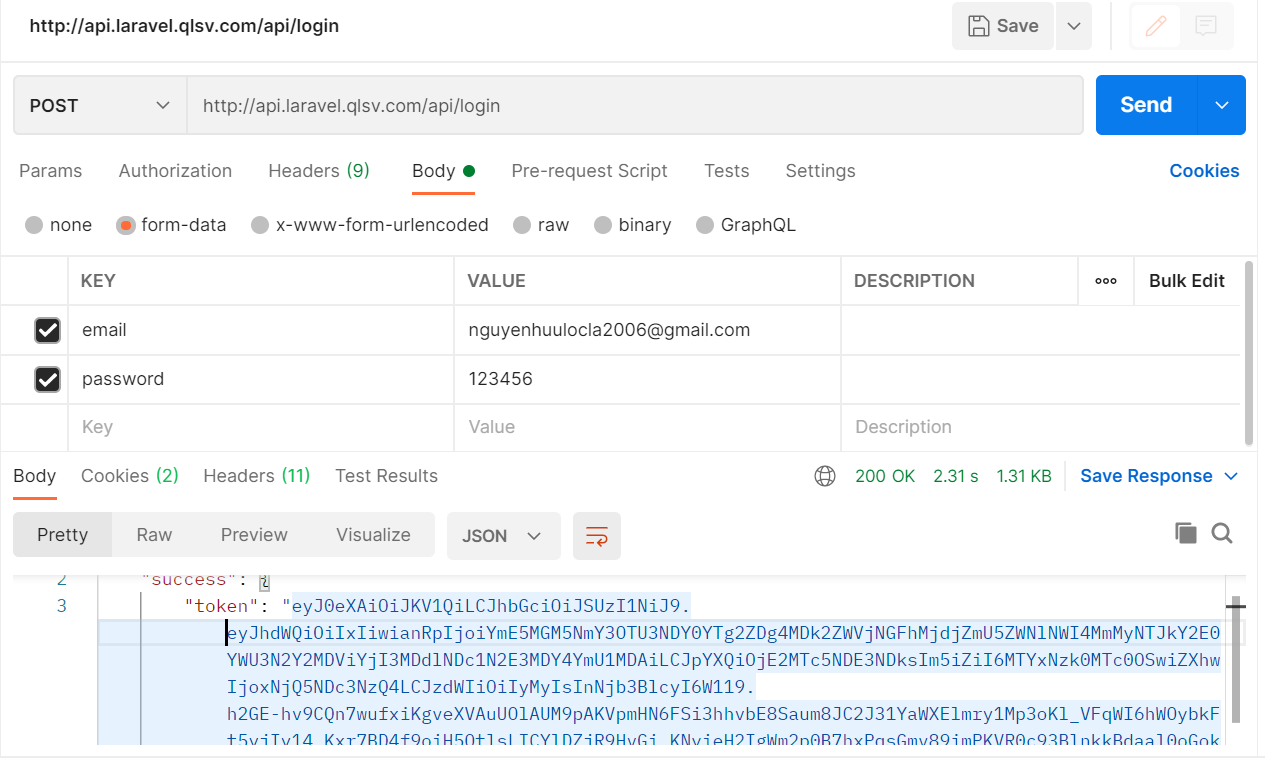
});

Sau đó dùng phần mềm POSTMAN để test api

Đăng ký:



Đăng nhập



Xem chi tiết thông tin của 1 user

