

Laravel Api with JWT Authentication

1 preparation

database configuration

add extension to composer json
"tymon/jwt-auth": "0.5.*"

composer update

add provider and alias to config app.php
provider:
Tymon\JWTAuth\Providers\JWTAuthServiceProvider::class,
alias:
'JWTAuth' => 'Tymon\JWTAuth\Facades\JWTAuth'::class,
'JWTFactory' => 'Tymon\JWTAuth\Facades\JWTFactory'::class,

add middleware to app/http/kernel.php
'jwt.auth' => \Tymon\JWTAuth\Middleware\GetUserFromToken::class,
'jwt.refresh' => \Tymon\JWTAuth\Middleware\RefreshToken::class,

vendor publish
php artisan vendor:publish --provider="Tymon\JWTAuth\Providers\JWTAuthServiceProvider"

2 configuration

php artisan migrate

Edit api route

```
Route::post('login', 'AuthenticateController@login');  
Route::post('register', 'AuthenticateController@register');  
  
Route::group(['prefix' => 'v1', 'middleware' => ['jwt.auth']], function() {  
    Route::resource('user', 'UserController');  
});
```

Create controllers

```
php artisan make:controller AuthenticateController  
php artisan make:controller UserController
```

3 implementation

Edit those two controllers

AuthenticateController

```
<?php  
  
namespace App\Http\Controllers;  
  
use Illuminate\Http\Request;  
use App\Http\Requests;  
use JWTAuth;  
use App\User;  
use Hash;  
  
class AuthenticateController extends Controller  
{  
    public function __construct(User $user)  
    {  
        $this->user = $user;  
    }  
  
    public function login(Request $request)  
    {  
        $credentials = $request->only(['email', 'password']);  
  
        if (!$token = JWTAuth::attempt($credentials))  
        {  
            return response()->json(['error', 'Invalid credential'], 401);  
        }  
  
        return response()->json(compact('token'));  
    }  
  
    public function register(Request $request)  
    {  
        $credentials = $request->only(['name', 'email', 'password']);  
        $credentials = [  
            'name' => $credentials['name'],  
            'email' => $credentials['email'],  
            'password' => Hash::make($credentials['password'])  
        ];  
  
        try {  
            $user = $this->user->create($credentials);  
        } catch (Exception $e) {  
            return response()->json(['error' => 'User already exist'], 409);  
        }  
        $token = JWTAuth::fromUser($user);  
  
        return response()->json(compact('token'));  
    }  
}
```

UserController

```
<?php  
  
namespace App\Http\Controllers;  
  
use Illuminate\Http\Request;  
use App\Http\Requests;  
use App\User;  
  
class UserController extends Controller  
{  
    public function __construct(User $user)  
    {  
        $this->user = $user;  
    }  
  
    public function index()  
    {  
        return $this->user->all();  
    }  
  
    public function show($id)  
    {  
        $data = $this->user->where('id', $id)->get();  
        return response()->json(compact('data'), 200);  
    }  
}
```

4 test

php artisan serve

test with rest client