لإنشاء **نظام إدارة مشاريع الفرق** باستخدامLaravel ، مع الجداول المطلوبة والربط بينها، اتبع هذه الخطوات التفصيلية:

.1إعداد المشروع وربطه بقاعدة البيانات:

إنشاء مشروع Laravel جديد:

١. افتح المحطة (terminal) وقم بإنشاء مشروع Laravel جديد:

```
composer create-project --prefer-dist laravel/laravel
team_project_management
```

بعد إنشاء المشروع، ادخل إلى المجلد:

```
cd team_project_management
```

إعداد الاتصال بقاعدة البيانات في ملف .env:

```
DB_CONNECTION=mysql

DB_HOST=127.0.0.1

DB_PORT=3306

DB_DATABASE=team_project_management

DB_USERNAME=root

=DB_PASSWORD
```

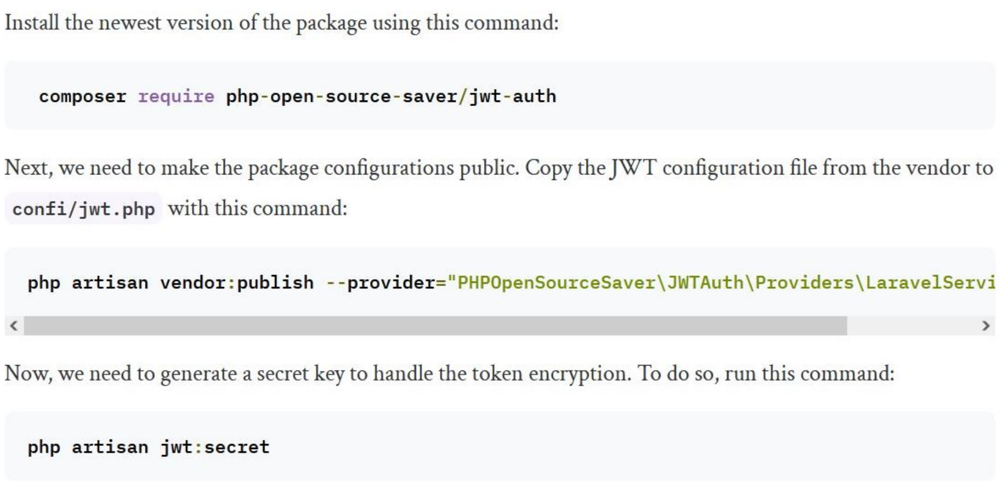
أنشئ قاعدة بيانات في PhpMyAdmin بنفس الاسم الذي حددته في .env:

قم بتشغيل التطبيق للتحقق من أنه يعمل بشكل صحيح:

```
php artisan serve
```

نقوم بتطبيق Authenticationباستخدام JWT

وفق الخطوات التالية



Install the newest version of the package using this command:

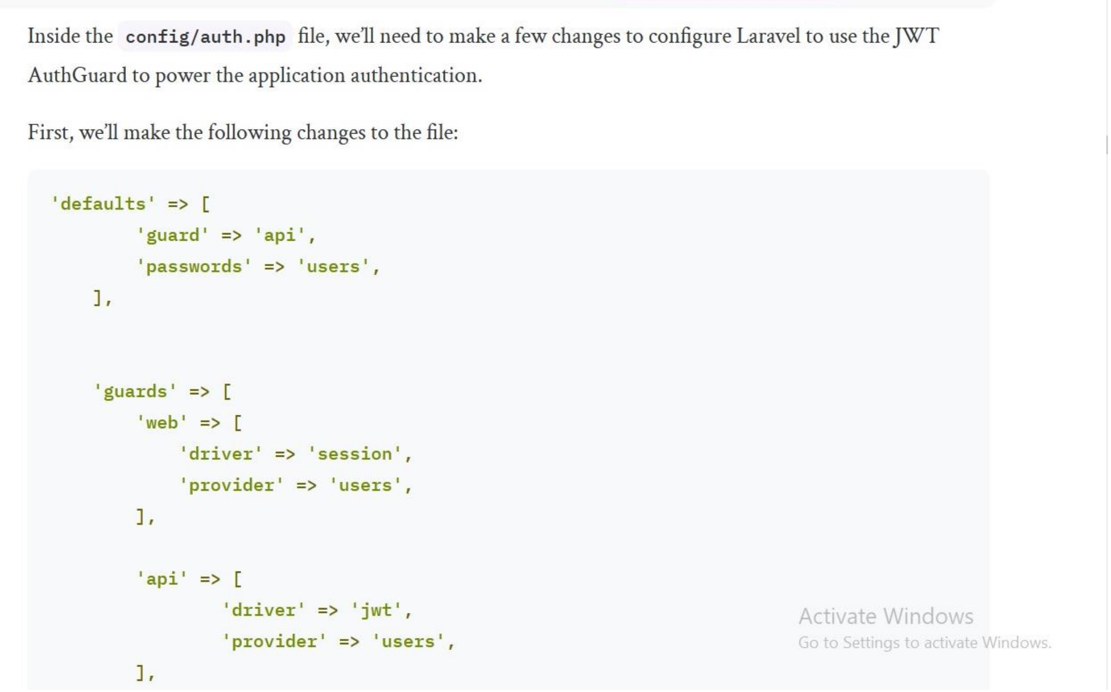
```
composer require php-open-source-saver/jwt-auth
```

Next, we need to make the package configurations public. Copy the JWT configuration file from the vendor to `confi/jwt.php` with this command:

```
php artisan vendor:publish --provider="PHPOpenSourceSaver\JWTAuth\Providers\LaravelServi
```

Now, we need to generate a secret key to handle the token encryption. To do so, run this command:

```
php artisan jwt:secret
```

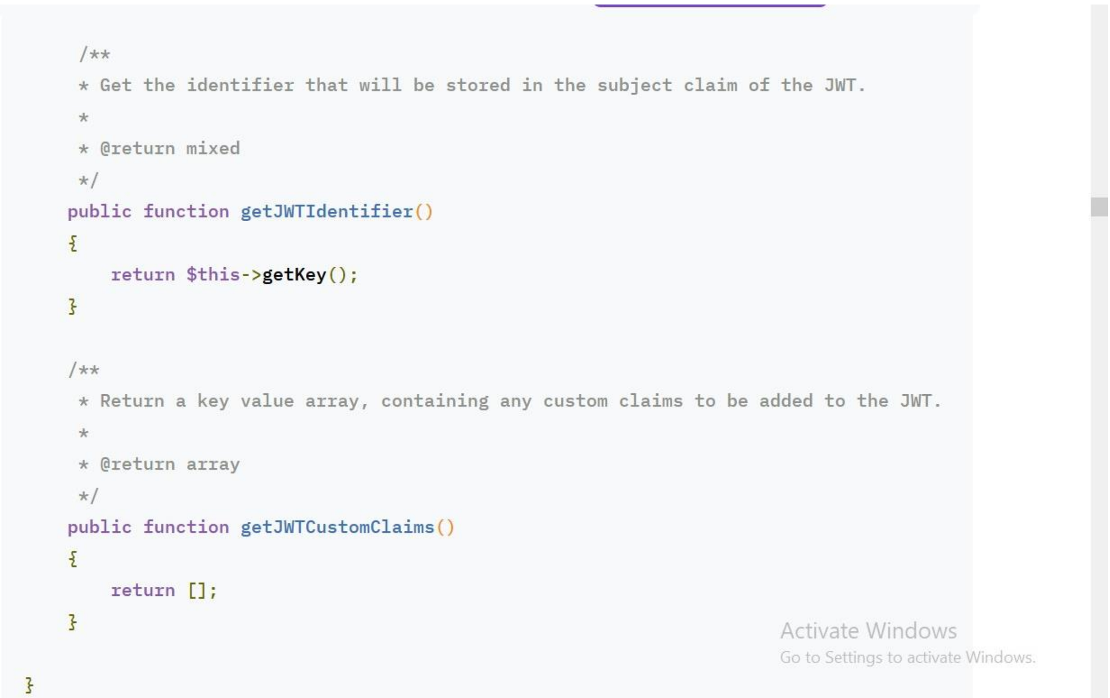


Inside the `config/auth.php` file, we'll need to make a few changes to configure Laravel to use the JWT AuthGuard to power the application authentication.

First, we'll make the following changes to the file:

```
'defaults' => [
        'guard' => 'api',
        'passwords' => 'users',
    ],


    'guards' => [
        'web' => [
            'driver' => 'session',
            'provider' => 'users',
        ],

        'api' => [
                'driver' => 'jwt',
                'provider' => 'users',
        ],
```

نضيف الدوال التالية الى model user



```
     /**
     * Get the identifier that will be stored in the subject claim of the JWT.
     *
     * @return mixed
     */
    public function getJWTIdentifier()
    {
        return $this->getKey();
    }

    /**
     * Return a key value array, containing any custom claims to be added to the JWT.
     *
     * @return array
     */
    public function getJWTCustomClaims()
    {
        return [];
    }

}
```

ننشأ الكونترولر الخاص بال Authentication بالتعليمة التالية

php artisan make:controller AuthController

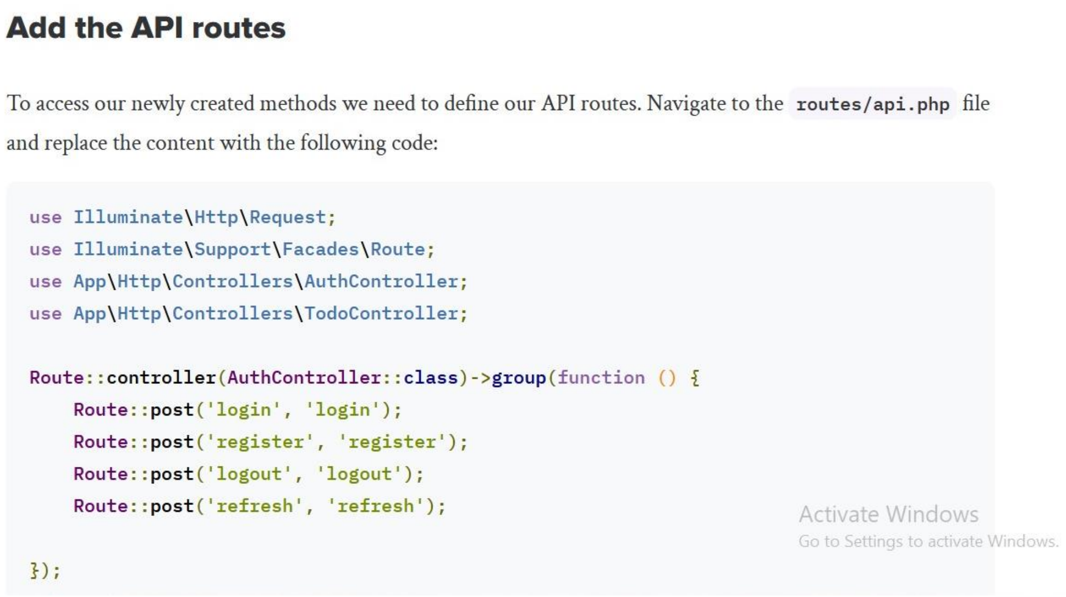
ومن ثم نضيف عليه ما يلي

```
namespace App\Http\Controllers;

use Illuminate\Http\Request;

use Illuminate\Support\Facades\Auth;

use Illuminate\Support\Facades\Hash;

use App\Models\User;


class AuthController extends Controller

{


    public function __construct()

    {

        $this->middleware('auth:api', ['except' => ['login','register']]);
```

```
}

public function login(Request $request)
{
    $request->validate([
        'email' => 'required|string|email',
        'password' => 'required|string',
    ]);
    $credentials = $request->only('email', 'password');

    $token = Auth::attempt($credentials);
    if (!$token) {
        return response()->json([
            'status' => 'error',
            'message' => 'Unauthorized',
        ], 401);
    }

    $user = Auth::user();
    return response()->json([
            'status' => 'success',
            'user' => $user,
            'authorisation' => [
                'token' => $token,
                'type' => 'bearer',
            ]
        ]);

}

public function register(Request $request){
    $request->validate([
        'name' => 'required|string|max:255',
        'email' => 'required|string|email|max:255|unique:users',
        'password' => 'required|string|min:6',
    ]);

    $user = User::create([
        'name' => $request->name,
        'email' => $request->email,
        'password' => Hash::make($request->password),
    ]);

    $token = Auth::login($user);
    return response()->json([
        'status' => 'success',
        'message' => 'User created successfully',
        'user' => $user,
        'authorisation' => [
            'token' => $token,
            'type' => 'bearer',
        ]
    ]);
}

public function logout()
{
    Auth::logout();
    return response()->json([
        'status' => 'success',
        'message' => 'Successfully logged out',
    ]);
}

public function refresh()
{
    return response()->json([
        'status' => 'success',
        'user' => Auth::user(),
        'authorisation' => [
            'token' => Auth::refresh(),
            'type' => 'bearer',
        ]
    ]);
}
```

}

نضيف ال routs بالشكل التالي



Add the API routes

To access our newly created methods we need to define our API routes. Navigate to the `routes/api.php` file and replace the content with the following code:

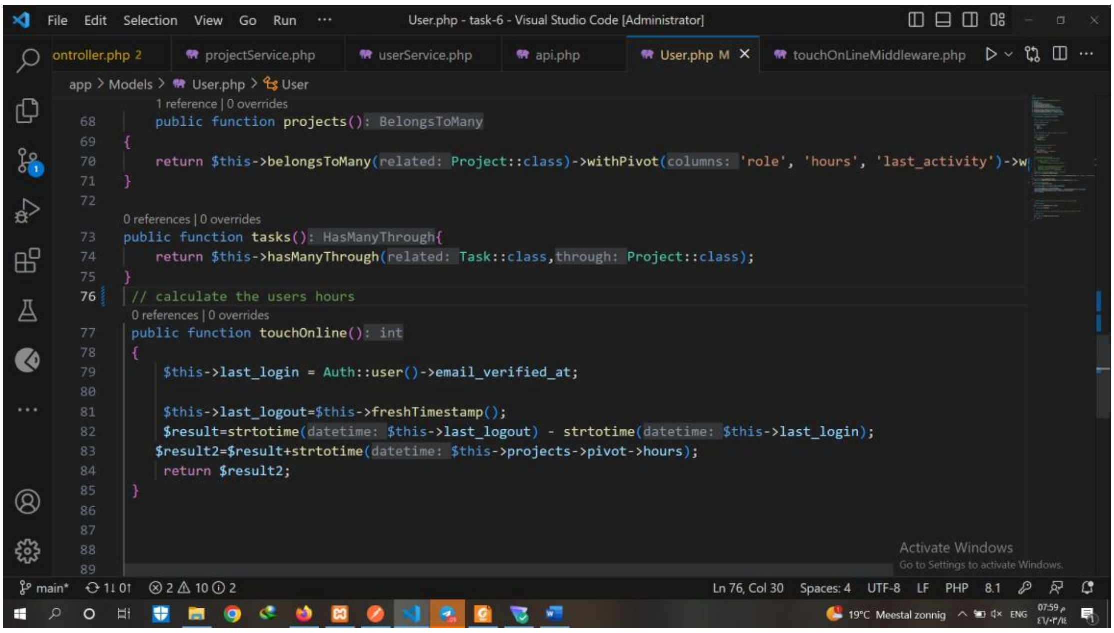
```php
use Illuminate\Http\Request;
use Illuminate\Support\Facades\Route;
use App\Http\Controllers\AuthController;
use App\Http\Controllers\TodoController;

Route::controller(AuthController::class)->group(function () {
    Route::post('login', 'login');
    Route::post('register', 'register');
    Route::post('logout', 'logout');
    Route::post('refresh', 'refresh');

});
```

ونتحقق من ما سبق من خلال البوست مان

ثم نقوم بتعديل . user Model

بإضافة العلاقة بين المستخدم والprojects

وأيضا إضافة دالة لحساب عدد ساعات المستخدم الحالي



File Edit Selection View Go Run ···
User.php - task-6 - Visual Studio Code [Administrator]
ontroller.php 2
projectService.php
userService.php
api.php
User.php M
touchOnLineMiddleware.php
app > Models > User.php > User
1 reference | 0 overrides
0 references | 0 overrides
0 references | 0 overrides

```php
    public function projects(): BelongsToMany
{
    return $this->belongsToMany(related: Project::class)->withPivot(columns: 'role', 'hours', 'last_activity')->w
}

public function tasks(): HasManyThrough{
    return $this->hasManyThrough(related: Task::class,through: Project::class);
}
 // calculate the users hours
 public function touchOnline(): int
 {
     $this->last_login = Auth::user()->email_verified_at;

     $this->last_logout=$this->freshTimestamp();
     $result=strtotime(datetime: $this->last_logout) - strtotime(datetime: $this->last_login);
    $result2=$result+strtotime(datetime: $this->projects->pivot->hours);
     return $result2;
 }
```

main* 1↓ 0↑ ⊗ 2 ⚠ 10 ⓘ 2
Ln 76, Col 30 Spaces: 4 UTF-8 LF PHP 8.1
Activate Windows
Go to Settings to activate Windows.
19°C Meestal zonnig ENG 07:59 م ٤/٠٣/١٤

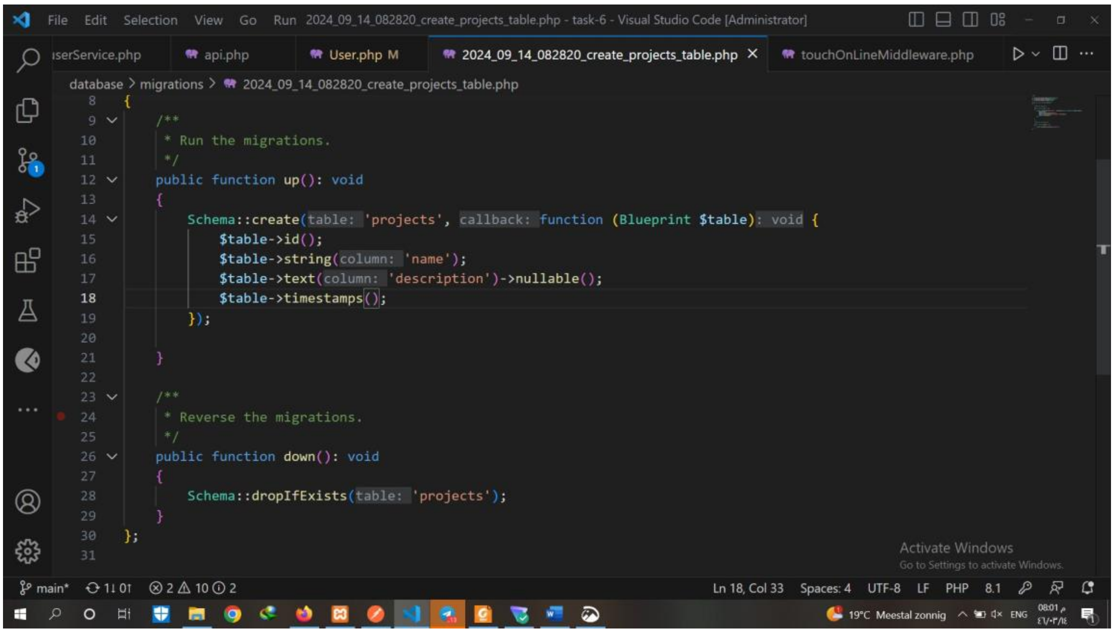
إنشاء الجداول:

إنشاء الجداول باستخدام الـ:Migration

١. إنشاء جدول :projects مع modl

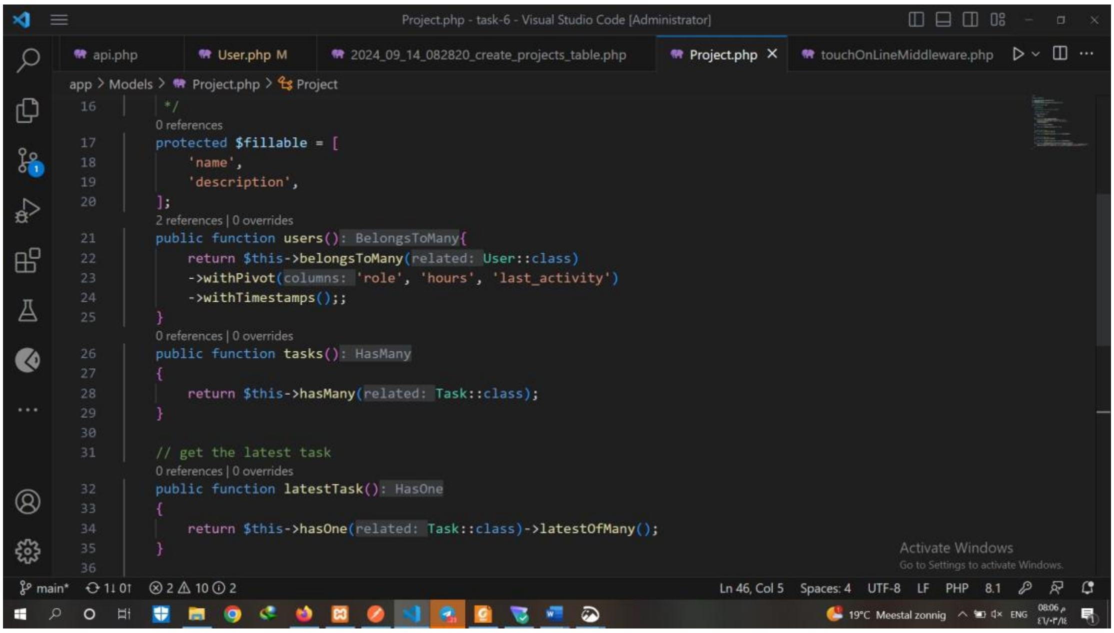
```
php artisan make:modle Project -m
```

في ملف الـ Migration الذي تم إنشاؤه، قم بإضافة الأعمدة المطلوبة:


2024_09_14_082820_create_projects_table.php - task-6 - Visual Studio Code [Administrator]
userService.php
api.php
User.php M
2024_09_14_082820_create_projects_table.php
touchOnLineMiddleware.php
database > migrations > 2024_09_14_082820_create_projects_table.php
main*
Ln 18, Col 33
Spaces: 4
UTF-8
LF
PHP
8.1

```php
{
    /**
     * Run the migrations.
     */
    public function up(): void
    {
        Schema::create(table: 'projects', callback: function (Blueprint $table): void {
            $table->id();
            $table->string(column: 'name');
            $table->text(column: 'description')->nullable();
            $table->timestamps();
        });

    }

    /**
     * Reverse the migrations.
     */
    public function down(): void
    {
        Schema::dropIfExists(table: 'projects');
    }
};
```

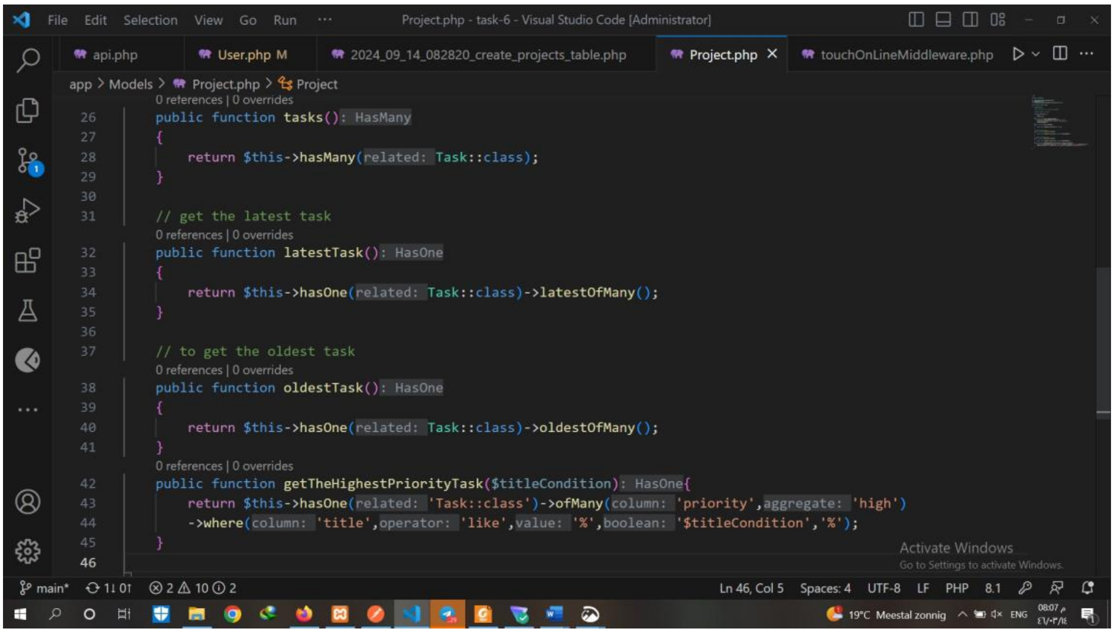
والتعديل على project Model


Project.php - task-6 - Visual Studio Code [Administrator]
api.php
User.php M
2024_09_14_082820_create_projects_table.php
Project.php
touchOnLineMiddleware.php
app > Models > Project.php > Project
main*
Ln 46, Col 5
Spaces: 4
UTF-8
LF
PHP
8.1

```php
    */
    0 references
    protected $fillable = [
        'name',
        'description',
    ];
    2 references | 0 overrides
    public function users(): BelongsToMany{
        return $this->belongsToMany(related: User::class)
        ->withPivot(columns: 'role', 'hours', 'last_activity')
        ->withTimestamps();;
    }
    0 references | 0 overrides
    public function tasks(): HasMany
    {
        return $this->hasMany(related: Task::class);
    }

    // get the latest task
    0 references | 0 overrides
    public function latestTask(): HasOne
    {
        return $this->hasOne(related: Task::class)->latestOfMany();
    }
```

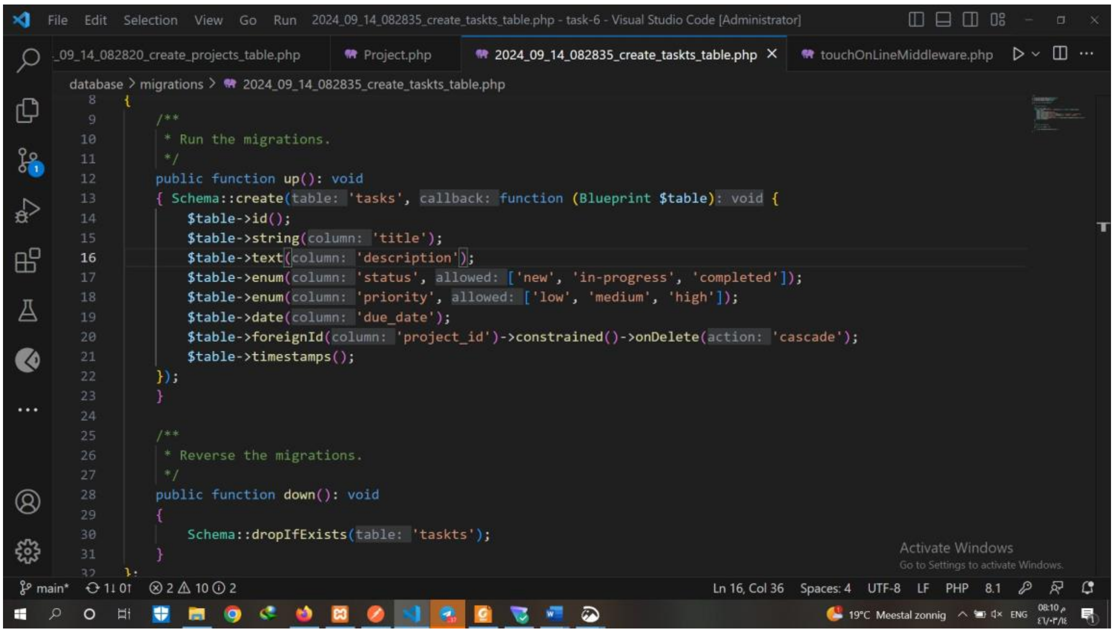
واضاافة العلاقة مع جدول المستخدمين

وانشاء دوال للوصول الى احدث مهمة واقدم مهمة

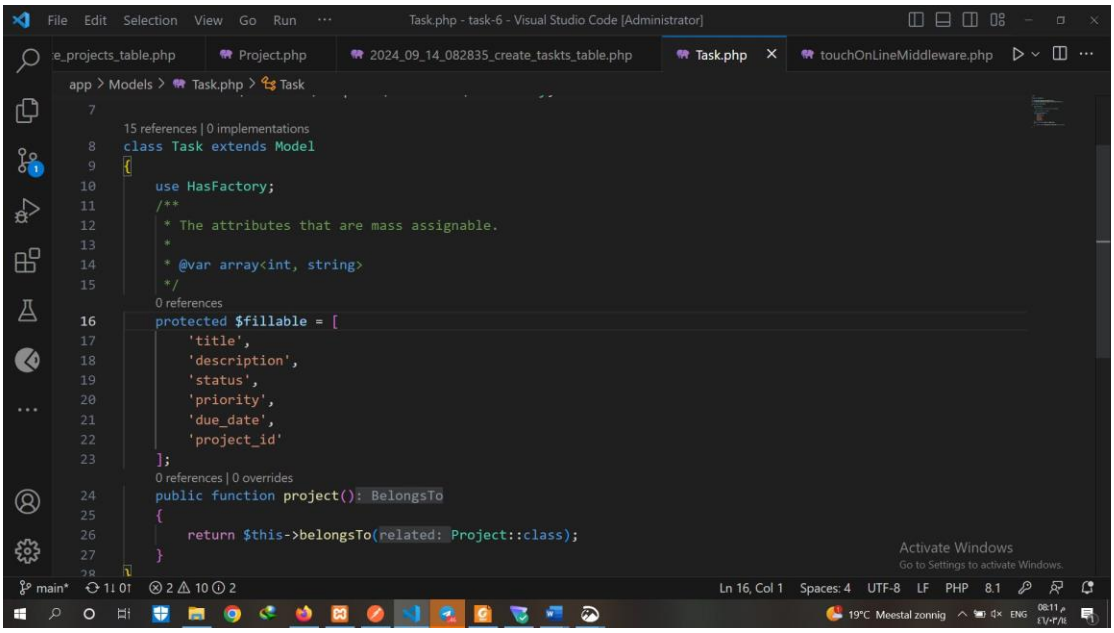

File Edit Selection View Go Run ···
Project.php - task-6 - Visual Studio Code [Administrator]
api.php
User.php M
2024_09_14_082820_create_projects_table.php
Project.php
touchOnLineMiddleware.php
app > Models > Project.php > Project
Activate Windows
Go to Settings to activate Windows.
main* 1↓ 0↑ ⊗ 2 ⚠ 10 ⓘ 2
Ln 46, Col 5 Spaces: 4 UTF-8 LF PHP 8.1
19°C Meestal zonnig ENG 08:07

```php
    0 references | 0 overrides
    public function tasks(): HasMany
    {
        return $this->hasMany(related: Task::class);
    }

    // get the latest task
    0 references | 0 overrides
    public function latestTask(): HasOne
    {
        return $this->hasOne(related: Task::class)->latestOfMany();
    }

    // to get the oldest task
    0 references | 0 overrides
    public function oldestTask(): HasOne
    {
        return $this->hasOne(related: Task::class)->oldestOfMany();
    }
    0 references | 0 overrides
    public function getTheHighestPriorityTask($titleCondition): HasOne{
        return $this->hasOne(related: 'Task::class')->ofMany(column: 'priority',aggregate: 'high')
        ->where(column: 'title',operator: 'like',value: '%',boolean: '$titleCondition','%');
    }
```

كما نقوم بإضافة maigrationو جدول المهام

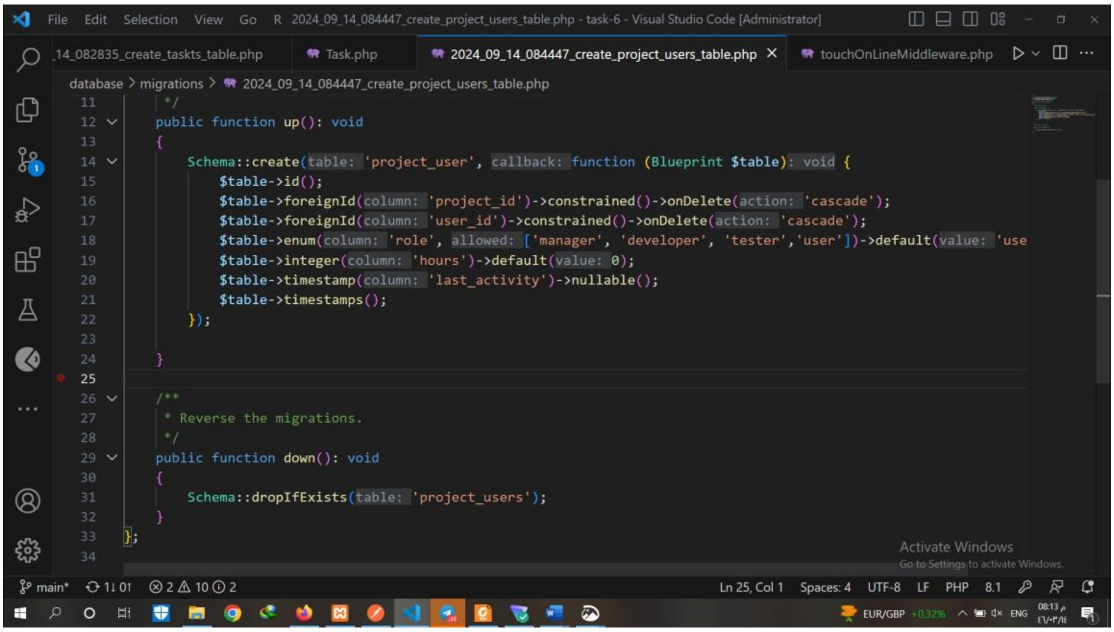

File Edit Selection View Go Run
2024_09_14_082835_create_taskts_table.php - task-6 - Visual Studio Code [Administrator]
_09_14_082820_create_projects_table.php
Project.php
2024_09_14_082835_create_taskts_table.php
touchOnLineMiddleware.php
database > migrations > 2024_09_14_082835_create_taskts_table.php
Activate Windows
Go to Settings to activate Windows.
main* 1↓ 0↑ ⊗ 2 ⚠ 10 ⓘ 2
Ln 16, Col 36 Spaces: 4 UTF-8 LF PHP 8.1
19°C Meestal zonnig ENG 08:10

```php
{
    /**
     * Run the migrations.
     */
    public function up(): void
    { Schema::create(table: 'tasks', callback: function (Blueprint $table): void {
        $table->id();
        $table->string(column: 'title');
        $table->text(column: 'description');
        $table->enum(column: 'status', allowed: ['new', 'in-progress', 'completed']);
        $table->enum(column: 'priority', allowed: ['low', 'medium', 'high']);
        $table->date(column: 'due_date');
        $table->foreignId(column: 'project_id')->constrained()->onDelete(action: 'cascade');
        $table->timestamps();
    });
    }

    /**
     * Reverse the migrations.
     */
    public function down(): void
    {
        Schema::dropIfExists(table: 'taskts');
    }
};
```


Task.php - task-6 - Visual Studio Code [Administrator]
e_projects_table.php
Project.php
2024_09_14_082835_create_taskts_table.php
Task.php
touchOnLineMiddleware.php
app > Models > Task.php > Task
Activate Windows
Go to Settings to activate Windows.
main*
Ln 16, Col 1
Spaces: 4
UTF-8
LF
PHP
8.1

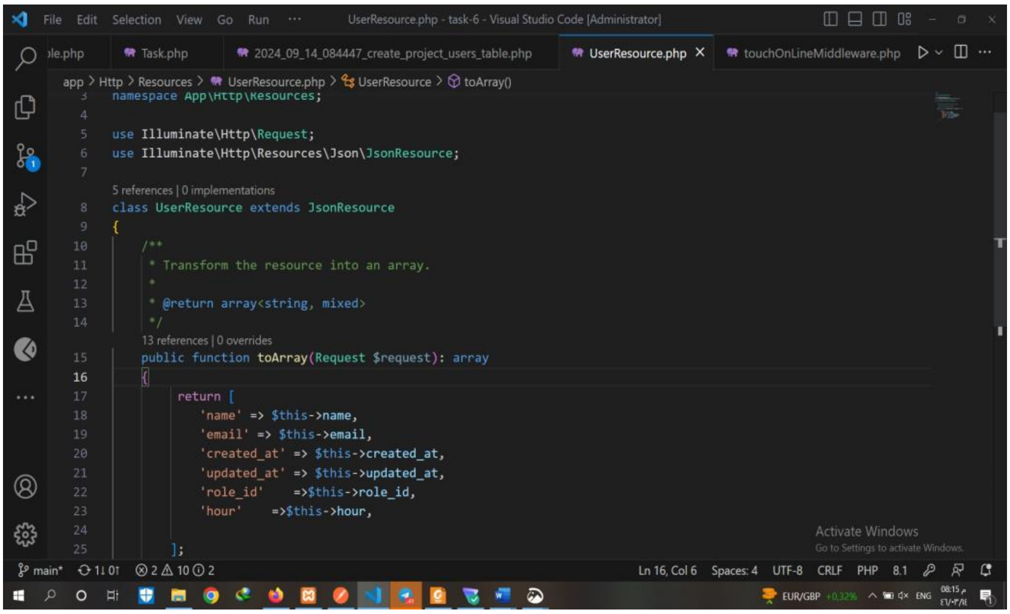
```
15 references | 0 implementations
class Task extends Model
{
    use HasFactory;
    /**
     * The attributes that are mass assignable.
     *
     * @var array<int, string>
     */
    0 references
    protected $fillable = [
        'title',
        'description',
        'status',
        'priority',
        'due_date',
        'project_id'
    ];
    0 references | 0 overrides
    public function project(): BelongsTo
    {
        return $this->belongsTo(related: Project::class);
    }
```

بعدها نقوم بانشاء pivot table

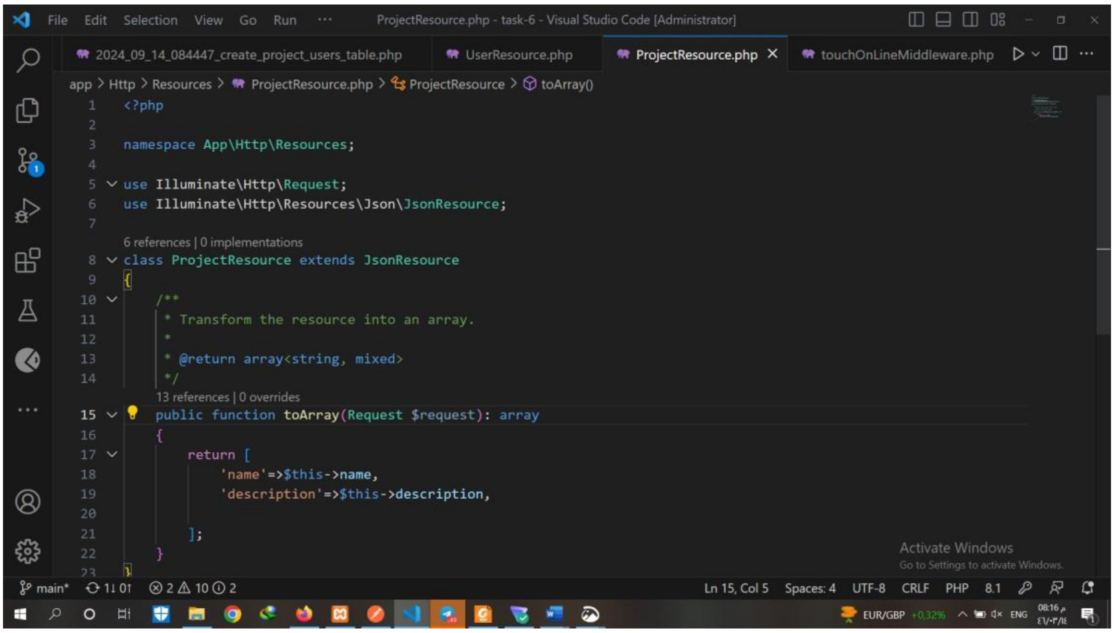

2024_09_14_084447_create_project_users_table.php - task-6 - Visual Studio Code [Administrator]
_14_082835_create_taskts_table.php
Task.php
2024_09_14_084447_create_project_users_table.php
touchOnLineMiddleware.php
database > migrations > 2024_09_14_084447_create_project_users_table.php
Activate Windows
Go to Settings to activate Windows.
main*
Ln 25, Col 1
Spaces: 4
UTF-8
LF
PHP
8.1

```
     */
    public function up(): void
    {
        Schema::create(table: 'project_user', callback: function (Blueprint $table): void {
            $table->id();
            $table->foreignId(column: 'project_id')->constrained()->onDelete(action: 'cascade');
            $table->foreignId(column: 'user_id')->constrained()->onDelete(action: 'cascade');
            $table->enum(column: 'role', allowed: ['manager', 'developer', 'tester','user'])->default(value: 'use
            $table->integer(column: 'hours')->default(value: 0);
            $table->timestamp(column: 'last_activity')->nullable();
            $table->timestamps();
        });

    }

    /**
     * Reverse the migrations.
     */
    public function down(): void
    {
        Schema::dropIfExists(table: 'project_users');
    }
};
```

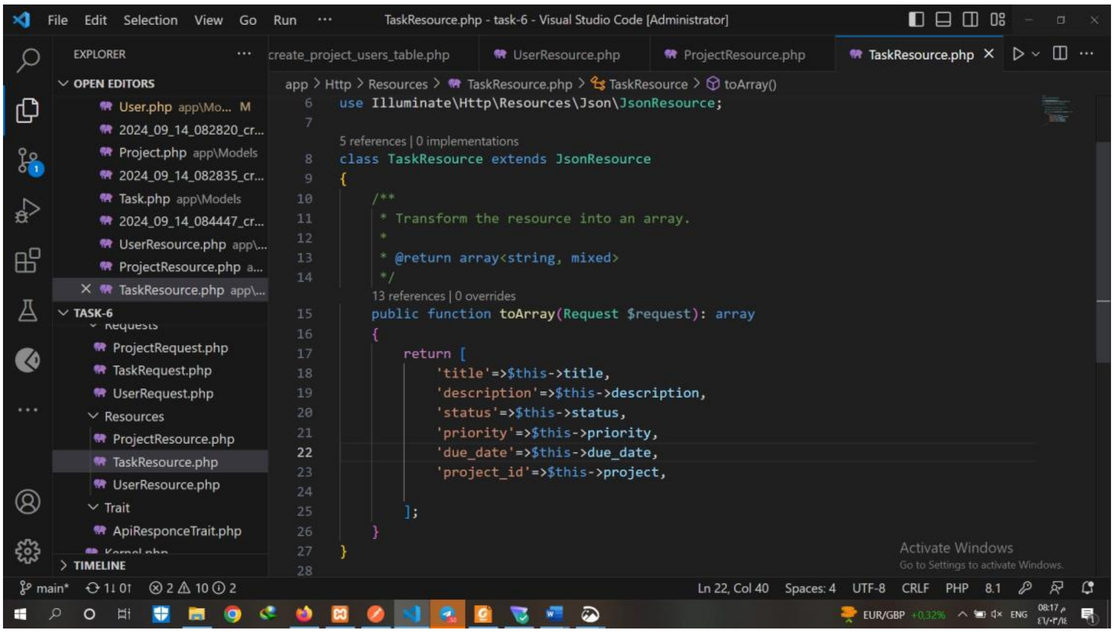
ونبدأ بانشاء resourcesلكل مودل


File Edit Selection View Go Run ···
UserResource.php - task-6 - Visual Studio Code [Administrator]
ole.php
Task.php
2024_09_14_084447_create_project_users_table.php
UserResource.php
touchOnLineMiddleware.php
app > Http > Resources > UserResource.php > UserResource > toArray()
5 references | 0 implementations
13 references | 0 overrides
Activate Windows
Go to Settings to activate Windows.
main* 1↓ 0↑ ⊗ 2 ⚠ 10 ⓘ 2
Ln 16, Col 6 Spaces: 4 UTF-8 CRLF PHP 8.1
EUR/GBP +0.32% ENG 08:15

```php
namespace App\Http\Resources;

use Illuminate\Http\Request;
use Illuminate\Http\Resources\Json\JsonResource;

class UserResource extends JsonResource
{
    /**
     * Transform the resource into an array.
     *
     * @return array<string, mixed>
     */
    public function toArray(Request $request): array
    {
        return [
            'name' => $this->name,
            'email' => $this->email,
            'created_at' => $this->created_at,
            'updated_at' => $this->updated_at,
            'role_id'    =>$this->role_id,
            'hour'    =>$this->hour,

        ];
```


File Edit Selection View Go Run ···
ProjectResource.php - task-6 - Visual Studio Code [Administrator]
2024_09_14_084447_create_project_users_table.php
UserResource.php
ProjectResource.php
touchOnLineMiddleware.php
app > Http > Resources > ProjectResource.php > ProjectResource > toArray()
6 references | 0 implementations
13 references | 0 overrides
Activate Windows
Go to Settings to activate Windows.
main* 1↓ 0↑ ⊗ 2 ⚠ 10 ⓘ 2
Ln 15, Col 5 Spaces: 4 UTF-8 CRLF PHP 8.1
EUR/GBP +0.32% ENG 08:16

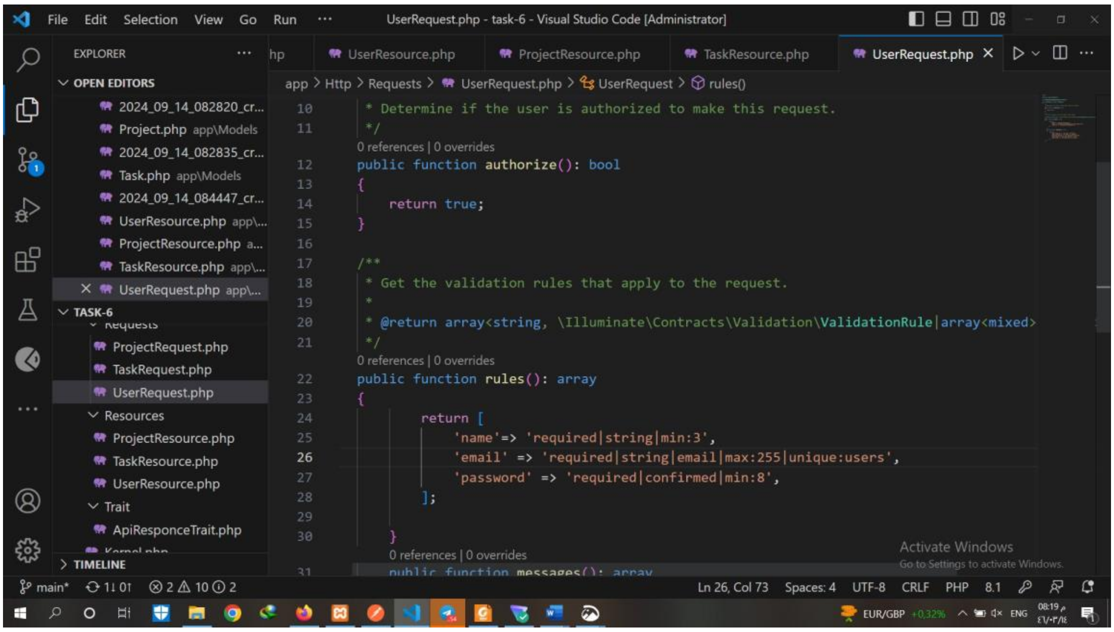
```php
<?php

namespace App\Http\Resources;

use Illuminate\Http\Request;
use Illuminate\Http\Resources\Json\JsonResource;

class ProjectResource extends JsonResource
{
    /**
     * Transform the resource into an array.
     *
     * @return array<string, mixed>
     */
    public function toArray(Request $request): array
    {
        return [
            'name'=>$this->name,
            'description'=>$this->description,

        ];
    }
}
```



TaskResource.php - task-6 - Visual Studio Code [Administrator]
app > Http > Resources > TaskResource.php > TaskResource > toArray()
Ln 22, Col 40 Spaces: 4 UTF-8 CRLF PHP 8.1

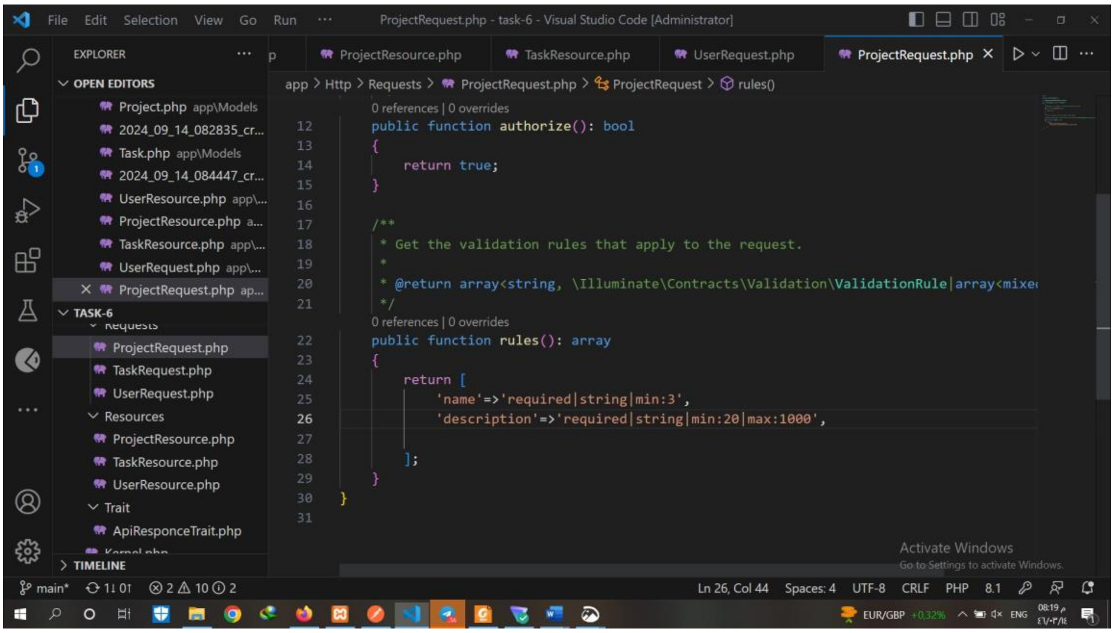
```php
use Illuminate\Http\Resources\Json\JsonResource;

class TaskResource extends JsonResource
{
    /**
     * Transform the resource into an array.
     *
     * @return array<string, mixed>
     */
    public function toArray(Request $request): array
    {
        return [
            'title'=>$this->title,
            'description'=>$this->description,
            'status'=>$this->status,
            'priority'=>$this->priority,
            'due_date'=>$this->due_date,
            'project_id'=>$this->project,

        ];
    }
}
```

ومن ثم نقوم بانشاء ال requestsمن اجل التحقق من صحة البيانات المدخلة



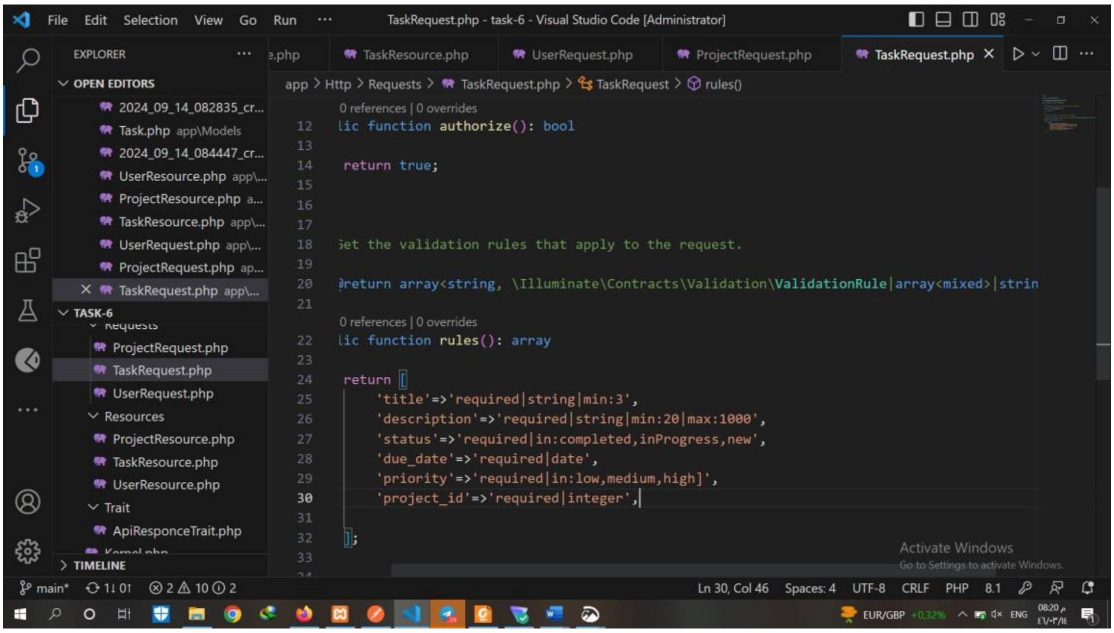
UserRequest.php - task-6 - Visual Studio Code [Administrator]
app > Http > Requests > UserRequest.php > UserRequest > rules()
Ln 26, Col 73 Spaces: 4 UTF-8 CRLF PHP 8.1

```php
     * Determine if the user is authorized to make this request.
     */
    public function authorize(): bool
    {
        return true;
    }

    /**
     * Get the validation rules that apply to the request.
     *
     * @return array<string, \Illuminate\Contracts\Validation\ValidationRule|array<mixed>
     */
    public function rules(): array
    {
        return [
            'name'=> 'required|string|min:3',
            'email' => 'required|string|email|max:255|unique:users',
            'password' => 'required|confirmed|min:8',
        ];

    }
    public function messages(): array
```



ProjectRequest.php - task-6 - Visual Studio Code [Administrator]
ProjectResource.php
TaskResource.php
UserRequest.php
ProjectRequest.php
app > Http > Requests > ProjectRequest.php > ProjectRequest > rules()

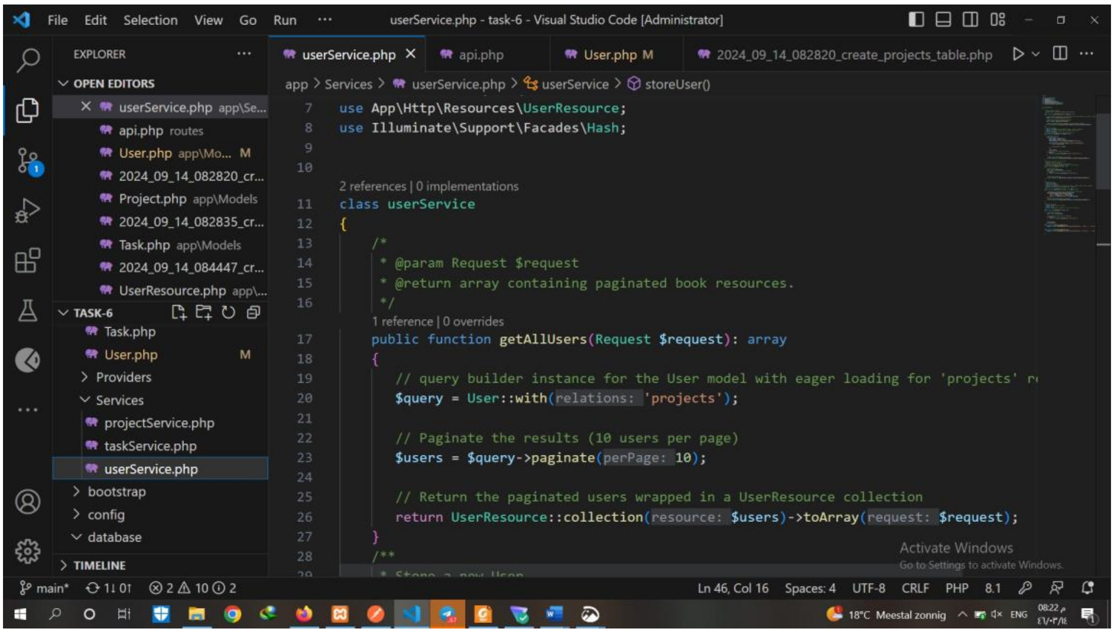
```php
    0 references | 0 overrides
    public function authorize(): bool
    {
        return true;
    }

    /**
     * Get the validation rules that apply to the request.
     *
     * @return array<string, \Illuminate\Contracts\Validation\ValidationRule|array<mixe
     */
    0 references | 0 overrides
    public function rules(): array
    {
        return [
            'name'=>'required|string|min:3',
            'description'=>'required|string|min:20|max:1000',

        ];
    }
}
```



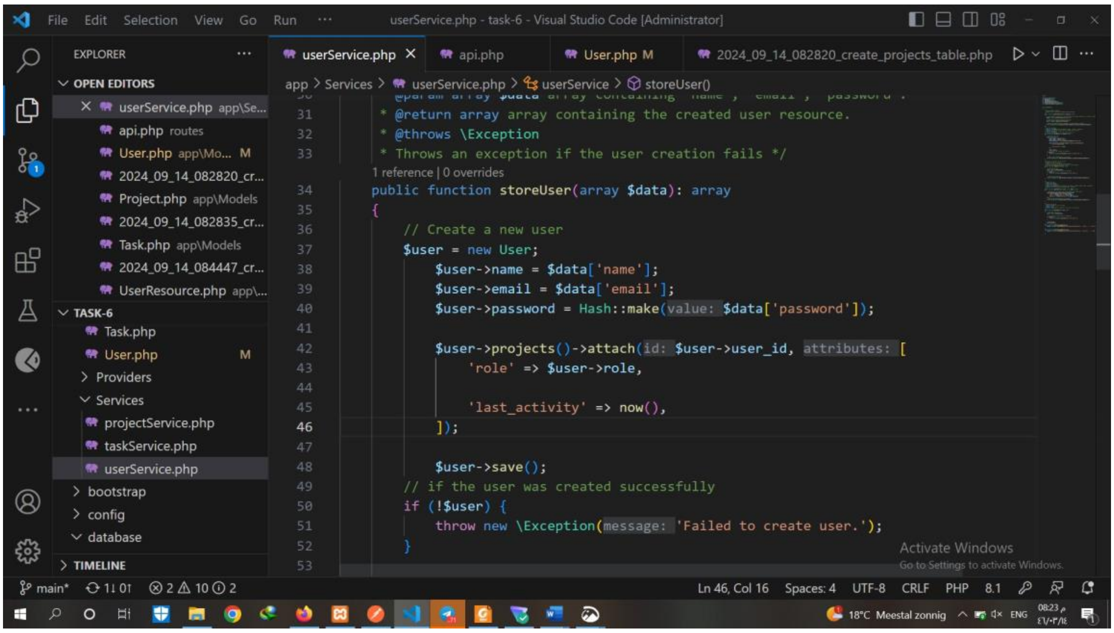
TaskRequest.php - task-6 - Visual Studio Code [Administrator]
TaskResource.php
UserRequest.php
ProjectRequest.php
TaskRequest.php
app > Http > Requests > TaskRequest.php > TaskRequest > rules()

```php
0 references | 0 overrides
lic function authorize(): bool

return true;



3et the validation rules that apply to the request.

@return array<string, \Illuminate\Contracts\Validation\ValidationRule|array<mixed>|strin

0 references | 0 overrides
lic function rules(): array

return [
     'title'=>'required|string|min:3',
     'description'=>'required|string|min:20|max:1000',
     'status'=>'required|in:completed,inProgress,new',
     'due_date'=>'required|date',
     'priority'=>'required|in:low,medium,high]',
     'project_id'=>'required|integer',

];
```

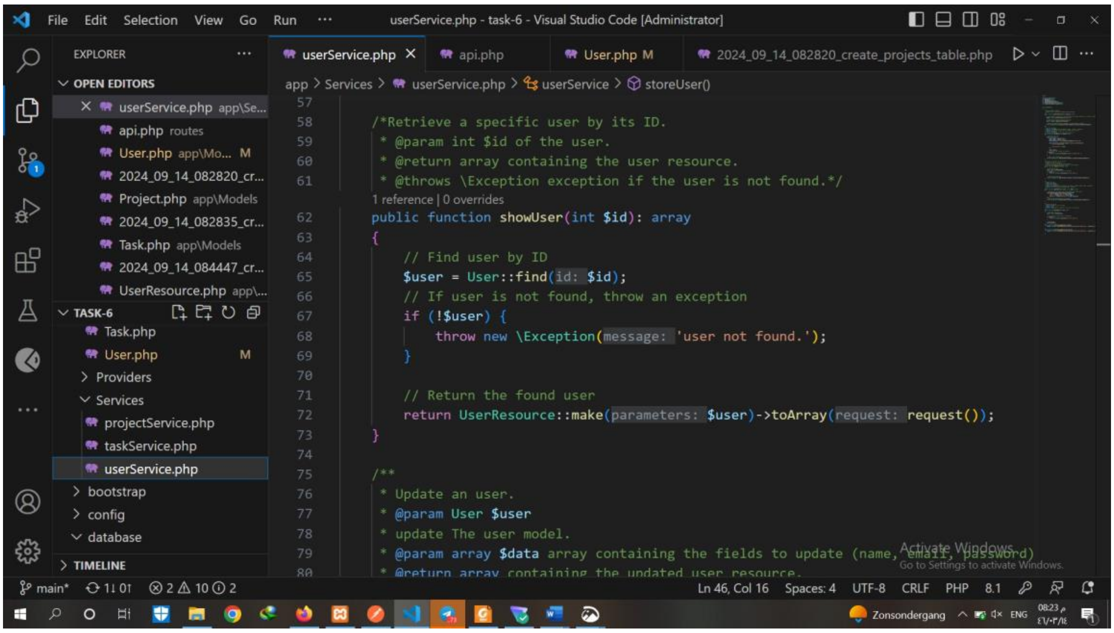
وننشأ ال services لنضع فيه عمليات ال crud ونخفف التعقيد داخل الكونترولر



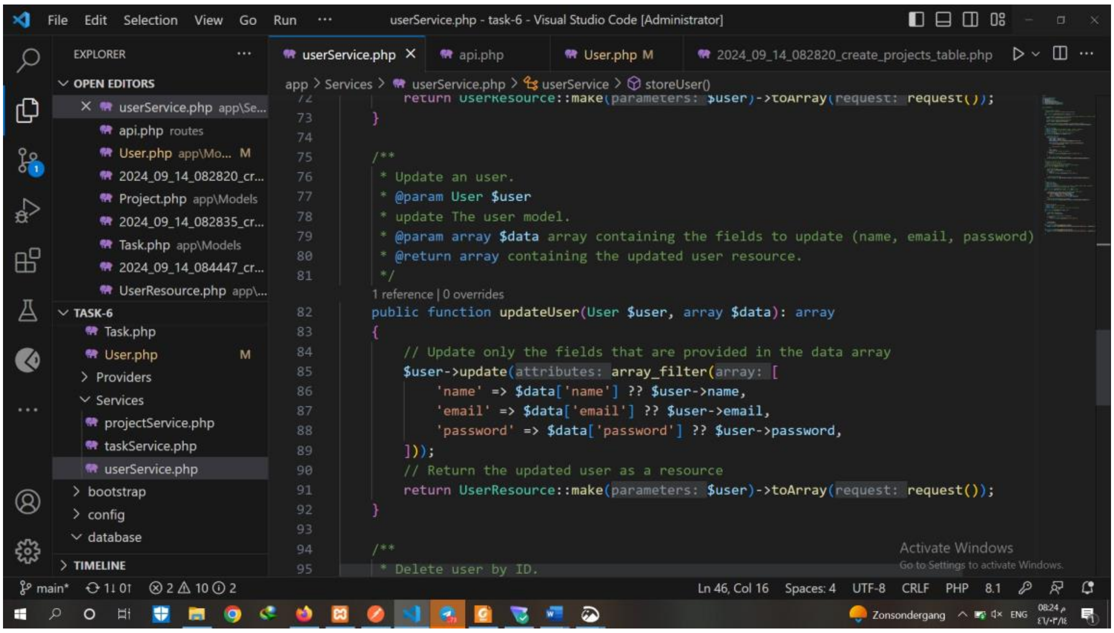
```php
use App\Http\Resources\UserResource;
use Illuminate\Support\Facades\Hash;


2 references | 0 implementations
class userService
{
    /*
     * @param Request $request
     * @return array containing paginated book resources.
     */
    1 reference | 0 overrides
    public function getAllUsers(Request $request): array
    {
        // query builder instance for the User model with eager loading for 'projects' r
        $query = User::with(relations: 'projects');

        // Paginate the results (10 users per page)
        $users = $query->paginate(perPage: 10);

        // Return the paginated users wrapped in a UserResource collection
        return UserResource::collection(resource: $users)->toArray(request: $request);
    }
    /**
     * Store a new User
```



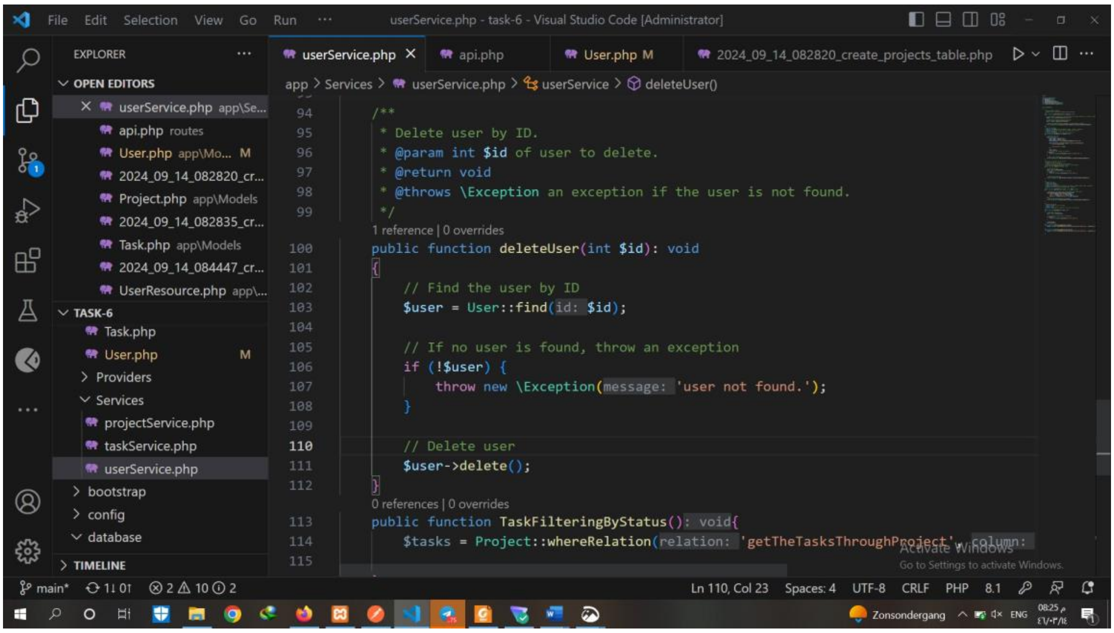
```php
     * @param array $data array containing 'name', 'email', 'password'.
     * @return array array containing the created user resource.
     * @throws \Exception
     * Throws an exception if the user creation fails */
    1 reference | 0 overrides
    public function storeUser(array $data): array
    {
        // Create a new user
        $user = new User;
            $user->name = $data['name'];
            $user->email = $data['email'];
            $user->password = Hash::make(value: $data['password']);

            $user->projects()->attach(id: $user->user_id, attributes: [
                'role' => $user->role,

                'last_activity' => now(),
            ]);

            $user->save();
        // if the user was created successfully
        if (!$user) {
            throw new \Exception(message: 'Failed to create user.');
        }
```



```php

    /*Retrieve a specific user by its ID.
     * @param int $id of the user.
     * @return array containing the user resource.
     * @throws \Exception exception if the user is not found.*/
    1 reference | 0 overrides
    public function showUser(int $id): array
    {
        // Find user by ID
        $user = User::find(id: $id);
        // If user is not found, throw an exception
        if (!$user) {
            throw new \Exception(message: 'user not found.');
        }

        // Return the found user
        return UserResource::make(parameters: $user)->toArray(request: request());
    }

    /**
     * Update an user.
     * @param User $user
     * update The user model.
     * @param array $data array containing the fields to update (name, email, password)
     * @return array containing the updated user resource.
```

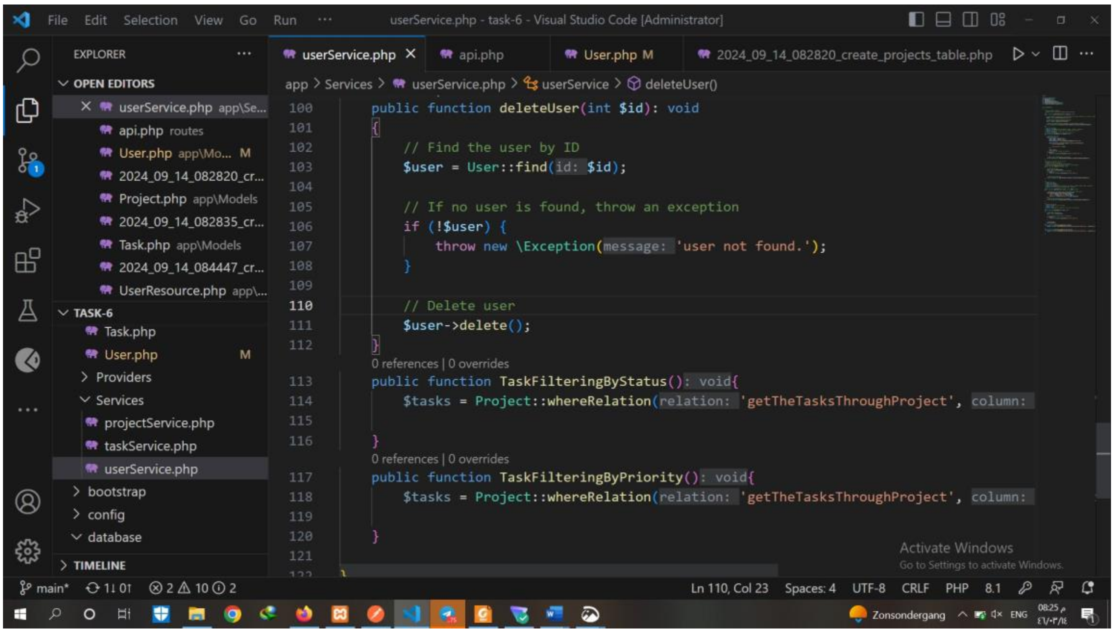


```php
        return UserResource::make(parameters: $user)->toArray(request: request());
    }

    /**
     * Update an user.
     * @param User $user
     * update The user model.
     * @param array $data array containing the fields to update (name, email, password)
     * @return array containing the updated user resource.
     */
    1 reference | 0 overrides
    public function updateUser(User $user, array $data): array
    {
        // Update only the fields that are provided in the data array
        $user->update(attributes: array_filter(array: [
            'name' => $data['name'] ?? $user->name,
            'email' => $data['email'] ?? $user->email,
            'password' => $data['password'] ?? $user->password,
        ]));
        // Return the updated user as a resource
        return UserResource::make(parameters: $user)->toArray(request: request());
    }

    /**
     * Delete user by ID.
```


userService.php - task-6 - Visual Studio Code [Administrator]
userService.php
api.php
User.php M
2024_09_14_082820_create_projects_table.php
app > Services > userService.php > userService > deleteUser()

```
/**
 * Delete user by ID.
 * @param int $id of user to delete.
 * @return void
 * @throws \Exception an exception if the user is not found.
 */
1 reference | 0 overrides
public function deleteUser(int $id): void
{
    // Find the user by ID
    $user = User::find(id: $id);

    // If no user is found, throw an exception
    if (!$user) {
        throw new \Exception(message: 'user not found.');
    }

    // Delete user
    $user->delete();
}
0 references | 0 overrides
public function TaskFilteringByStatus(): void{
    $tasks = Project::whereRelation(relation: 'getTheTasksThroughProject', column:
```

ننشأ دوال تحقق فلترة المهام حسب الأولوية والحالة

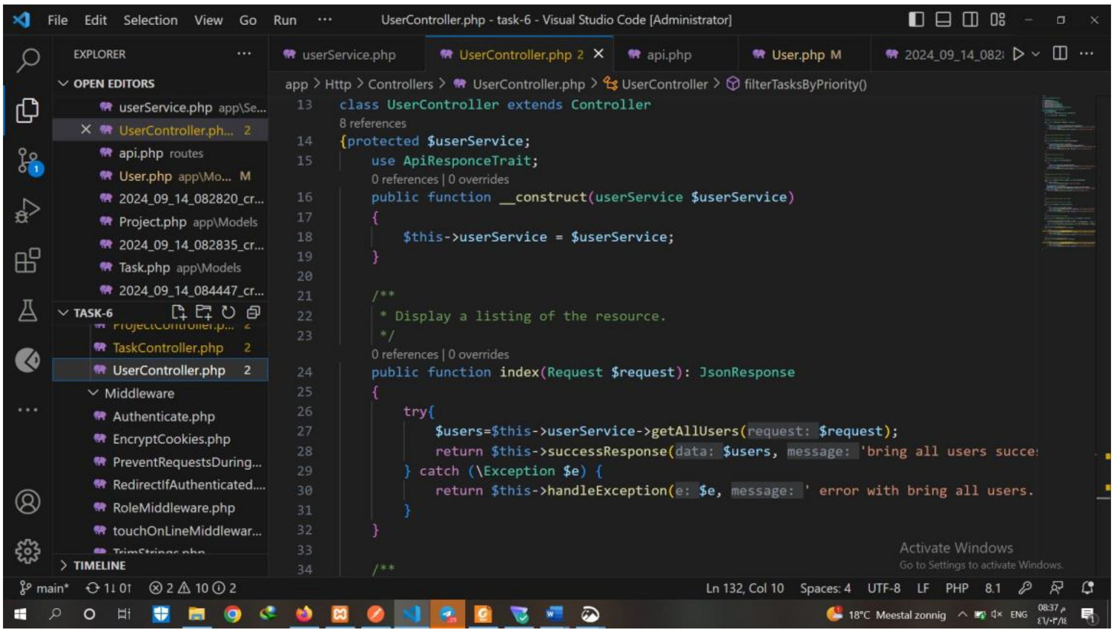

userService.php - task-6 - Visual Studio Code [Administrator]
userService.php
api.php
User.php M
2024_09_14_082820_create_projects_table.php
app > Services > userService.php > userService > deleteUser()

```
public function deleteUser(int $id): void
{
    // Find the user by ID
    $user = User::find(id: $id);

    // If no user is found, throw an exception
    if (!$user) {
        throw new \Exception(message: 'user not found.');
    }

    // Delete user
    $user->delete();
}
0 references | 0 overrides
public function TaskFilteringByStatus(): void{
    $tasks = Project::whereRelation(relation: 'getTheTasksThroughProject', column:

}
0 references | 0 overrides
public function TaskFilteringByPriority(): void{
    $tasks = Project::whereRelation(relation: 'getTheTasksThroughProject', column:

}
```

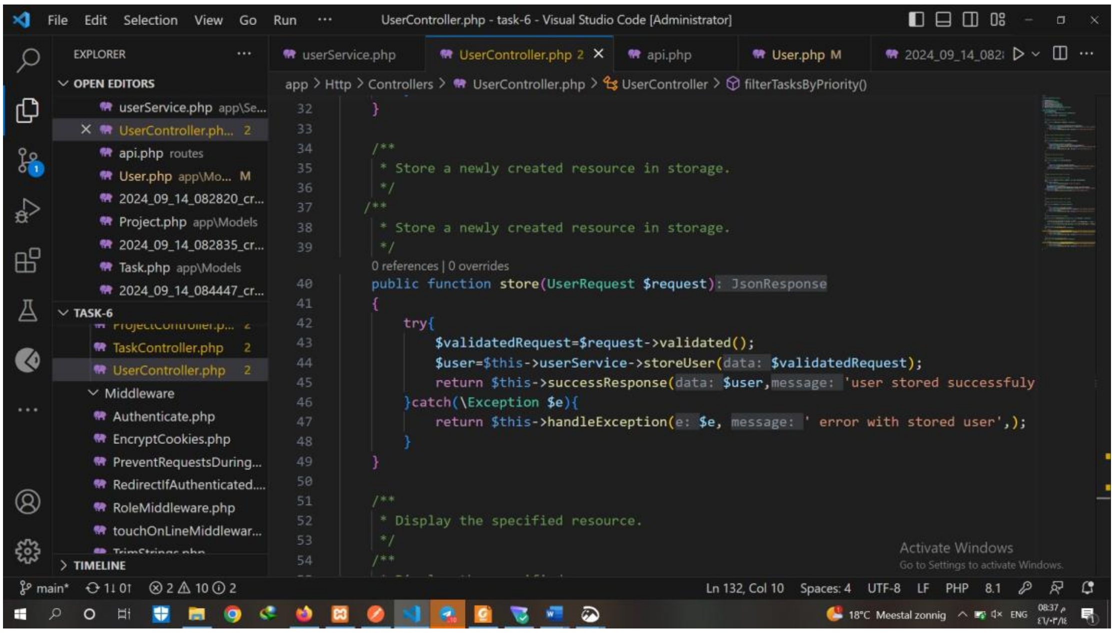
وبنفس الطريقة ننشأ services لكل من ال Task, project

ونقوم بانشاء الكونترولر بالتعليمة التالية

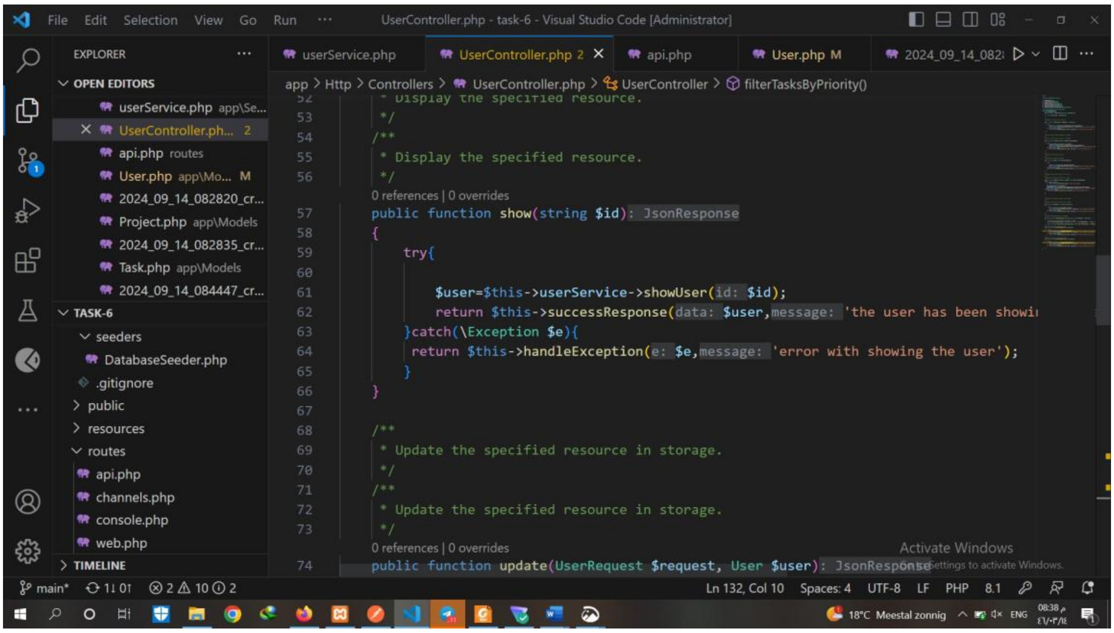
Php artisan make:controller User controller



```
class UserController extends Controller
8 references
{protected $userService;
    use ApiResponceTrait;
    0 references | 0 overrides
    public function __construct(userService $userService)
    {
        $this->userService = $userService;
    }

    /**
     * Display a listing of the resource.
     */
    0 references | 0 overrides
    public function index(Request $request): JsonResponse
    {
        try{
            $users=$this->userService->getAllUsers(request: $request);
            return $this->successResponse(data: $users, message: 'bring all users succe:
        } catch (\Exception $e) {
            return $this->handleException(e: $e, message: ' error with bring all users.
        }
    }

    /**
```

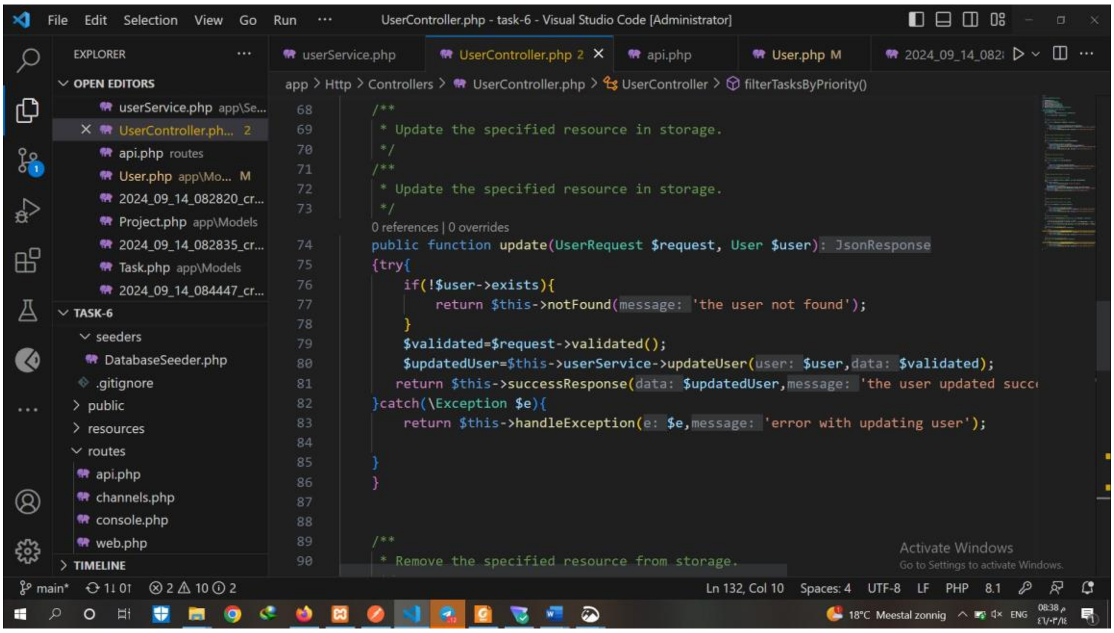


```
    }

    /**
     * Store a newly created resource in storage.
     */
   /**
     * Store a newly created resource in storage.
     */
    0 references | 0 overrides
    public function store(UserRequest $request): JsonResponse
    {
        try{
            $validatedRequest=$request->validated();
            $user=$this->userService->storeUser(data: $validatedRequest);
            return $this->successResponse(data: $user,message: 'user stored successfuly
        }catch(\Exception $e){
            return $this->handleException(e: $e, message: ' error with stored user',);
        }
    }

    /**
     * Display the specified resource.
     */
    /**
```



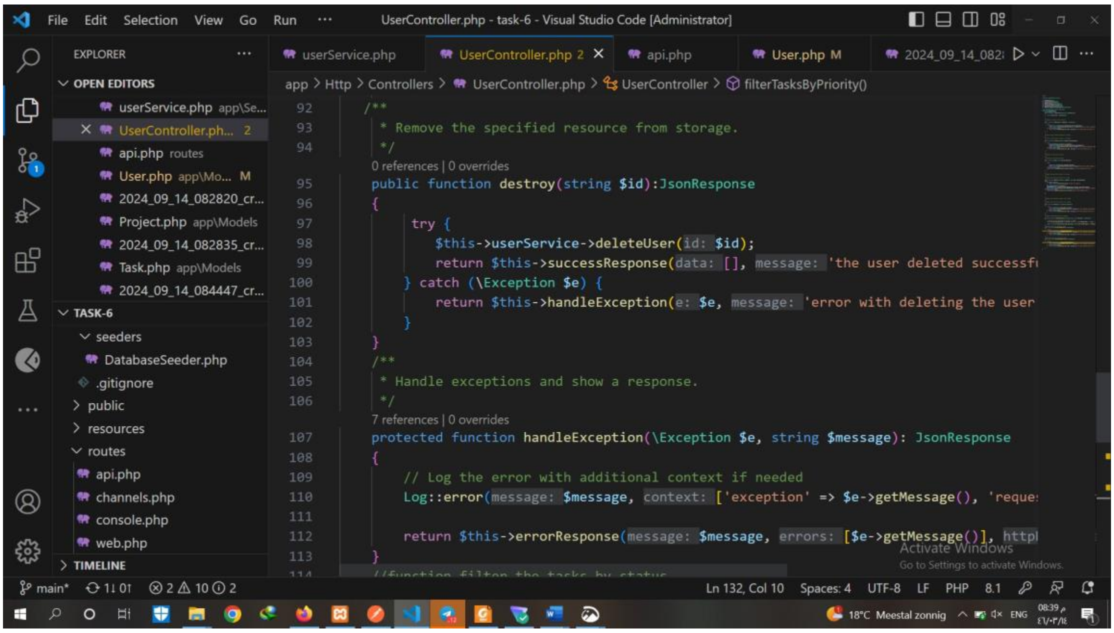
UserController.php - task-6 - Visual Studio Code [Administrator]
userService.php
UserController.php 2
api.php
User.php M
2024_09_14_082
app > Http > Controllers > UserController.php > UserController > filterTasksByPriority()
EXPLORER
OPEN EDITORS
TASK-6
TIMELINE
Ln 132, Col 10 Spaces: 4 UTF-8 LF PHP 8.1

```php
     * Display the specified resource.
     */
    /**
     * Display the specified resource.
     */
    0 references | 0 overrides
    public function show(string $id): JsonResponse
    {
        try{

            $user=$this->userService->showUser(id: $id);
            return $this->successResponse(data: $user,message: 'the user has been showi
        }catch(\Exception $e){
         return $this->handleException(e: $e,message: 'error with showing the user');
        }
    }

    /**
     * Update the specified resource in storage.
     */
    /**
     * Update the specified resource in storage.
     */
    0 references | 0 overrides
    public function update(UserRequest $request, User $user): JsonResponse
```



UserController.php - task-6 - Visual Studio Code [Administrator]
userService.php
UserController.php 2
api.php
User.php M
2024_09_14_082
app > Http > Controllers > UserController.php > UserController > filterTasksByPriority()
EXPLORER
OPEN EDITORS
TASK-6
TIMELINE
Ln 132, Col 10 Spaces: 4 UTF-8 LF PHP 8.1

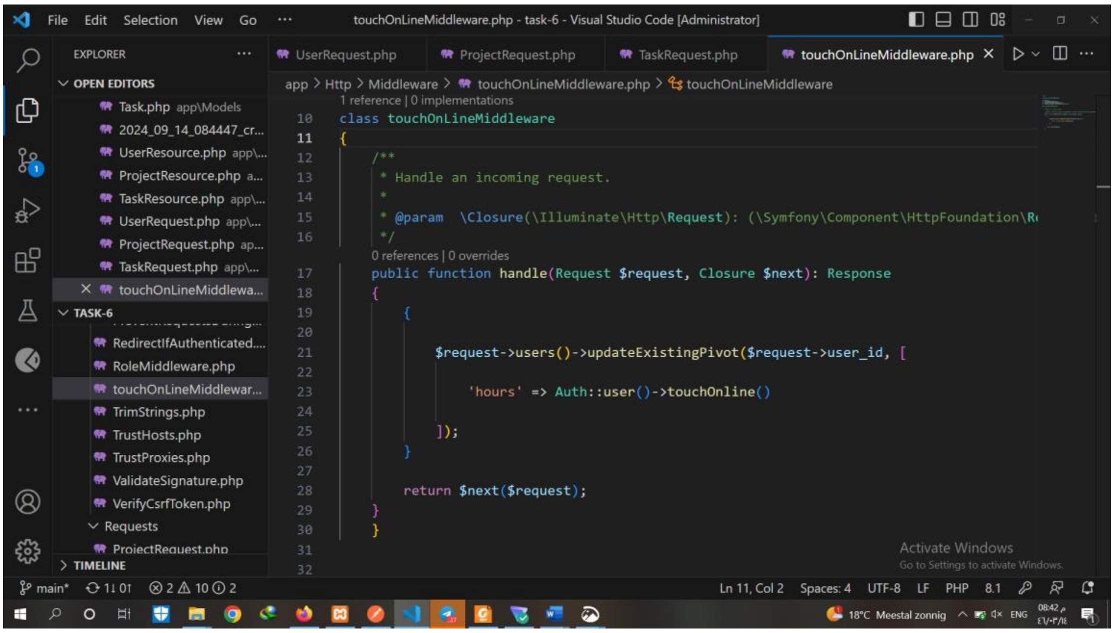
```php
    /**
     * Update the specified resource in storage.
     */
    /**
     * Update the specified resource in storage.
     */
    0 references | 0 overrides
    public function update(UserRequest $request, User $user): JsonResponse
    {try{
        if(!$user->exists){
            return $this->notFound(message: 'the user not found');
        }
        $validated=$request->validated();
        $updatedUser=$this->userService->updateUser(user: $user,data: $validated);
       return $this->successResponse(data: $updatedUser,message: 'the user updated succ
    }catch(\Exception $e){
        return $this->handleException(e: $e,message: 'error with updating user');

    }
    }


    /**
     * Remove the specified resource from storage.
```



```
/**
 * Remove the specified resource from storage.
 */
public function destroy(string $id):JsonResponse
{
    try {
        $this->userService->deleteUser(id: $id);
        return $this->successResponse(data: [], message: 'the user deleted successfu
    } catch (\Exception $e) {
        return $this->handleException(e: $e, message: 'error with deleting the user
    }
}
/**
 * Handle exceptions and show a response.
 */
protected function handleException(\Exception $e, string $message): JsonResponse
{
    // Log the error with additional context if needed
    Log::error(message: $message, context: ['exception' => $e->getMessage(), 'reque

    return $this->errorResponse(message: $message, errors: [$e->getMessage()], httpI
}
```

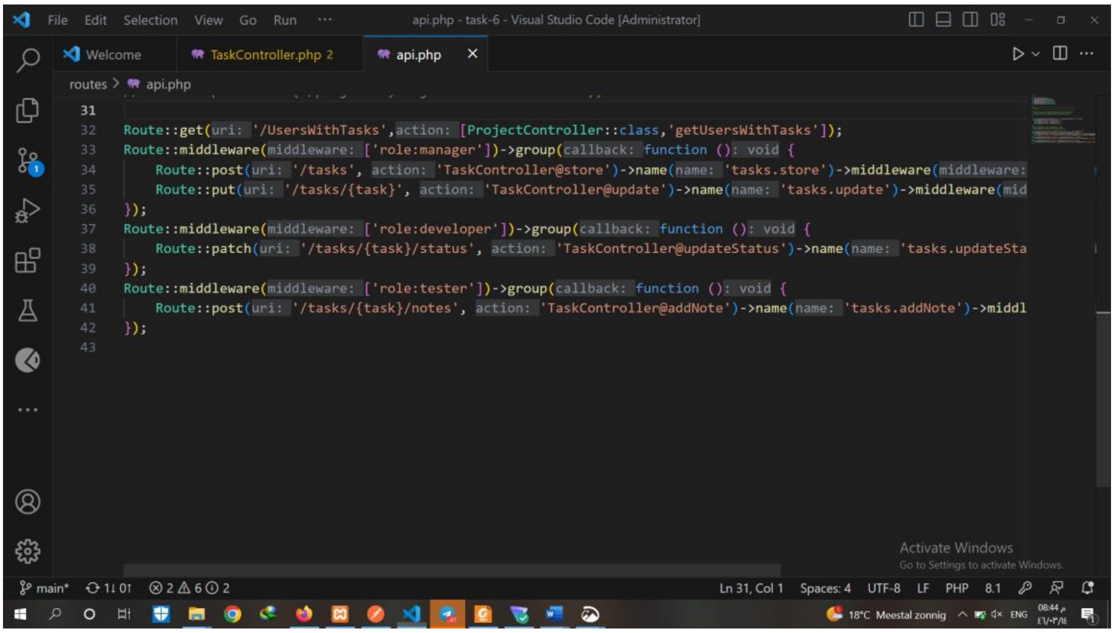
وبنفس الطريقة من اجل الProject, task

نقوم بانشاء ميدلوير من اجل حساب عدد ساعات المستخدم الحالي



```
class touchOnLineMiddleware
{
    /**
     * Handle an incoming request.
     *
     * @param  \Closure(\Illuminate\Http\Request): (\Symfony\Component\HttpFoundation\Re
     */
    public function handle(Request $request, Closure $next): Response
    {
        {

            $request->users()->updateExistingPivot($request->user_id, [

                'hours' => Auth::user()->touchOnline()

            ]);
        }

        return $next($request);
    }
}
```

وأخيرا نضيف ال routs



api.php - task-6 - Visual Studio Code [Administrator]
File Edit Selection View Go Run ···
Welcome
TaskController.php 2
api.php
routes > api.php
main* 1↓ 0↑ ⊗ 2 ⚠ 6 ⓘ 2
Ln 31, Col 1 Spaces: 4 UTF-8 LF PHP 8.1
Activate Windows
Go to Settings to activate Windows.
18°C Meestal zonnig ENG 08:44

```php

Route::get(uri: '/UsersWithTasks',action: [ProjectController::class,'getUsersWithTasks']);
Route::middleware(middleware: ['role:manager'])->group(callback: function (): void {
    Route::post(uri: '/tasks', action: 'TaskController@store')->name(name: 'tasks.store')->middleware(middleware:
    Route::put(uri: '/tasks/{task}', action: 'TaskController@update')->name(name: 'tasks.update')->middleware(mid
});
Route::middleware(middleware: ['role:developer'])->group(callback: function (): void {
    Route::patch(uri: '/tasks/{task}/status', action: 'TaskController@updateStatus')->name(name: 'tasks.updateSta
});
Route::middleware(middleware: ['role:tester'])->group(callback: function (): void {
    Route::post(uri: '/tasks/{task}/notes', action: 'TaskController@addNote')->name(name: 'tasks.addNote')->middl
});

```

وأخيرا نختبر المشروع عن طريق البوستمان