

Wijdan Ardra Fulvian

6706223137

D3IF 46-04

## Praktikum 7

- Http
- Kernel.php

```
1 <?php
2
3 // Wijdan Ardra Fulvian 6706223137 46-04
4
5 namespace App\Http;
6
7 use Illuminate\Foundation\Http\Kernel as HttpKernel;
8
9 1 reference | 0 implementations
10 class Kernel extends HttpKernel
11 {
12     /**
13      * The application's global HTTP middleware stack.
14      * These middleware are run during every request to your application.
15      *
16      * @var array<int, class-string|string>
17      */
18     0 references
19     protected $middleware = [
20         // \App\Http\Middleware\TrustHosts::class,
21         \App\Http\Middleware\TrustProxies::class,
22         \Illuminate\Http\Middleware\HandleCors::class,
23         \App\Http\Middleware\PreventRequestsDuringMaintenance::class,
24         \Illuminate\Foundation\Http\Middleware\ValidatePostSize::class,
25         \App\Http\Middleware\TrimStrings::class,
26         \Illuminate\Foundation\Http\Middleware\ConvertEmptyStringsToNull::class,
27     ];
28
29     /**
30      * The application's route middleware groups.
31      *
32      * @var array<string, array<int, class-string|string>>
33      */
34     0 references
35     protected $middlewareGroups = [
36         'web' => [
37             \App\Http\Middleware\EncryptCookies::class,
38             \Illuminate\Cookie\Middleware\AddQueuedCookiesToResponse::class,
39             \Illuminate\Session\Middleware\StartSession::class,
40             \Illuminate\View\Middleware\ShareErrorsFromSession::class,
41             \App\Http\Middleware\VerifyCsrfToken::class,
42             \Illuminate\Routing\Middleware\SubstituteBindings::class,
43         ],
44         'api' => [
45             // \Laravel\Sanctum\Http\Middleware\EnsureFrontendRequestsAreStateful::class,
46             \Illuminate\Routing\Middleware\ThrottleRequests::class, 'api',
47             \Illuminate\Routing\Middleware\SubstituteBindings::class,
48             \Laravel\Sanctum\Http\Middleware\EnsureFrontendRequestsAreStateful::class,
49             'throttle:api',
50             \Illuminate\Routing\Middleware\SubstituteBindings::class,
51         ],
52     ];
53
54     /**
55      * The application's middleware aliases.
56      * Aliases may be used instead of class names to conveniently assign middleware to routes and groups.
57      *
58      * @var array<string, class-string|string>
59      */
60     0 references
61     protected $middlewareAliases = [
62         'auth' => \App\Http\Middleware\Authenticate::class,
63         'auth.basic' => \Illuminate\Auth\Middleware\AuthenticateWithBasicAuth::class,
64         'auth.session' => \Illuminate\Session\Middleware\AuthenticateSession::class,
65         'cache.headers' => \Illuminate\Http\Middleware\SetCacheHeaders::class,
66         'can' => \Illuminate\Auth\Middleware\Authorize::class,
67         'guest' => \App\Http\Middleware\RedirectIfAuthenticated::class,
68         'password.confirm' => \Illuminate\Auth\Middleware\RequirePassword::class,
69         'precognitive' => \Illuminate\Foundation\Http\Middleware\HandlePrecognitiveRequests::class,
70         'signed' => \App\Http\Middleware\ValidateSignature::class,
71         'throttle' => \Illuminate\Routing\Middleware\ThrottleRequests::class,
72         'verified' => \Illuminate\Auth\Middleware\EnsureEmailIsVerified::class,
73     ];
74 }
```

- **Controllers**
  - ApiController.php

- AuthController.php

```

1 <?php
2
3 // Wijdan Andra Fulvian 6706223137 46-04
4
5 namespace App\Http\Controllers;
6
7 use App\Http\Controllers\Controller;
8 use App\Models\User;
9 use Illuminate\Http\Request;
10 use Illuminate\Support\Facades\Auth;
11 use Illuminate\Support\Facades\Hash;
12
13 > references | 0 implementations
14 class AuthController extends Controller
15 {
16     //
17
18     > reference | 0 overrides
19     public function register(Request $request)
20     {
21         $validatedData = $request->validate($rules: [
22             'name' => 'required|string|max:255',
23             'email' => 'required|string|email|max:255|unique:users',
24             'password' => 'required|string|min:8',
25         ]);
26         $user = User::create([
27             'name' => $validatedData['name'],
28             'email' => $validatedData['email'],
29             'password' => Hash::make($validatedData['password']),
30         ]);
31         $token = $user->createToken('auth_token')->plainTextToken;
32         return response()->json([
33             'access_token' => $token,
34             'token_type' => 'Bearer',
35         ]);
36     }
37
38     > reference | 0 overrides
39     public function login(Request $request)
40     {
41         if (!Auth::attempt($credentials: $request->only($keys: 'email', 'password'))) {
42             return response()->json([
43                 'message' => 'Invalid login details',
44                 'status' => 401;
45             ], $status: 401);
46         }
47         $user = User::where('email', $request['email'])->firstOrFail();
48         $token = $user->createToken('auth_token')->plainTextToken;
49         return response()->json([
50             'access_token' => $token,
51             'token_type' => 'Bearer',
52         ]);
53     }
54 }

```

- Controller.php

```

1 <?php
2
3 // Wijdan Andra Fulvian 6706223137 46-04
4
5 namespace App\Http\Controllers;
6
7 use Illuminate\Foundation\Auth\Access\AuthorizesRequests;
8 use Illuminate\Foundation\Validation\ValidatesRequests;
9 use Illuminate\Routing\Controller as BaseController;
10
11 > references | 3 implementations
12 class Controller extends BaseController
13 {
14     use AuthorizesRequests, ValidatesRequests;
15 }

```

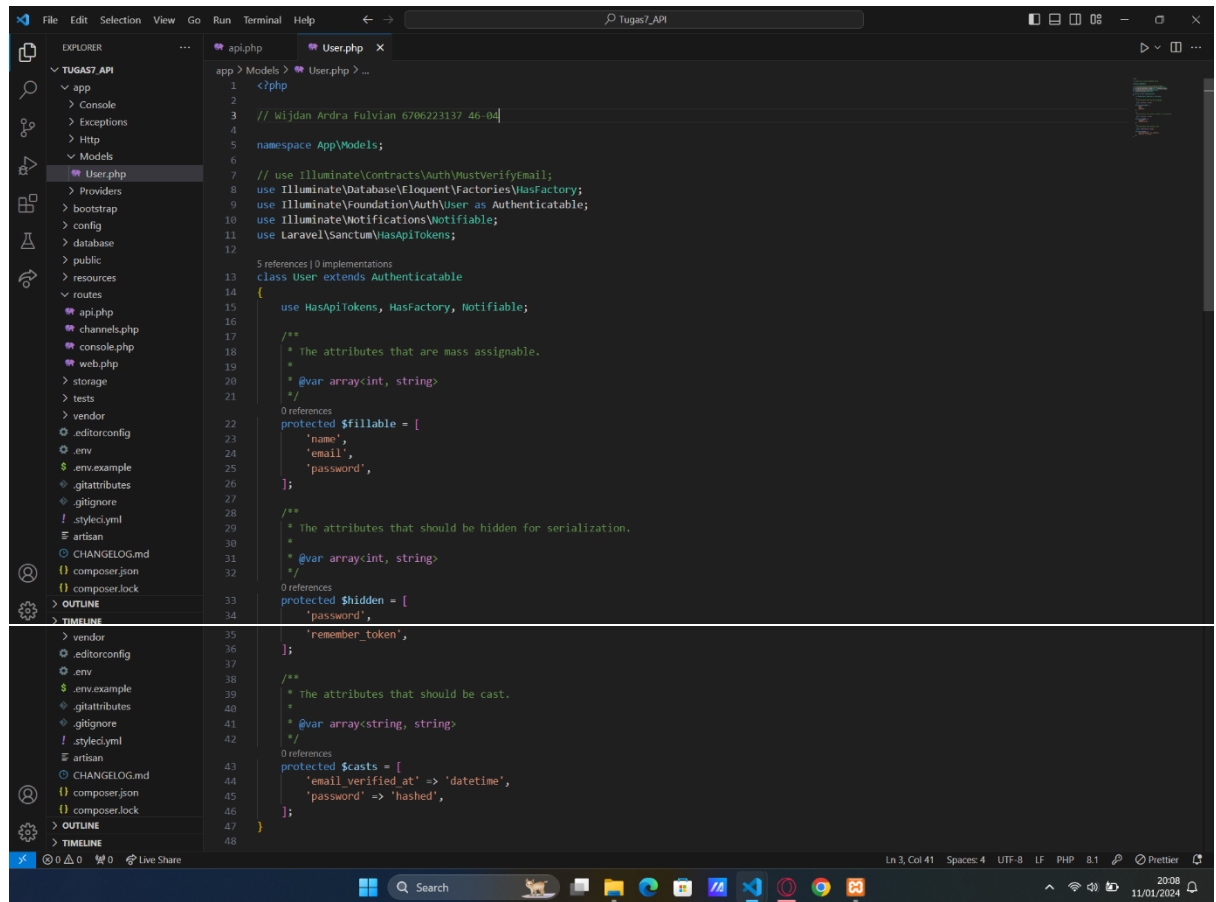
- UserController.php

```

1  <?php
2
3  // Wijdan Andra Fulvian 6706223137 46-04
4
5  namespace App\Http\Controllers;
6
7  use App\Http\Controllers\Controller;
8  use Illuminate\Http\Request;
9  use Illuminate\Support\Facades\DB;
10
11  class UserController extends Controller
12  {
13      /**
14       * Display a listing of the resource.
15       */
16      0 references | 0 overrides
17      public function index()
18      {
19          //
20      }
21
22      /**
23       * Show the form for creating a new resource.
24       */
25      0 references | 0 overrides
26      public function create()
27      {
28          //
29      }
30
31      /**
32       * Store a newly created resource in storage.
33       */
34      0 references | 0 overrides
35      public function store(Request $request)
36      {
37          //
38      }
39
40      /**
41       * Display the specified resource.
42       */
43      0 references | 0 overrides
44      public function show(string $id)
45      {
46          //
47      }
48
49      /**
50       * Show the form for editing the specified resource.
51       */
52      0 references | 0 overrides
53      public function edit(string $id)
54      {
55          //
56      }
57
58      /**
59       * Update the specified resource in storage.
60       */
61      0 references | 0 overrides
62      public function update(Request $request, string $id)
63      {
64          //
65      }
66
67      /**
68       * Remove the specified resource from storage.
69       */
70      0 references | 0 overrides
71      public function destroy(string $id)
72      {
73          //
74      }
75
76      2 references | 0 overrides
77      public function getUsers() {
78          $users = DB::table('users')
79              ->select(
80                  columns: 'name',
81                  'email',
82              )
83              ->orderBy( column: 'name', direction: 'asc')
84              ->get();
85          return response()->json( data: $users, status: 200);
86      }
87  }

```

- Models
- User.php

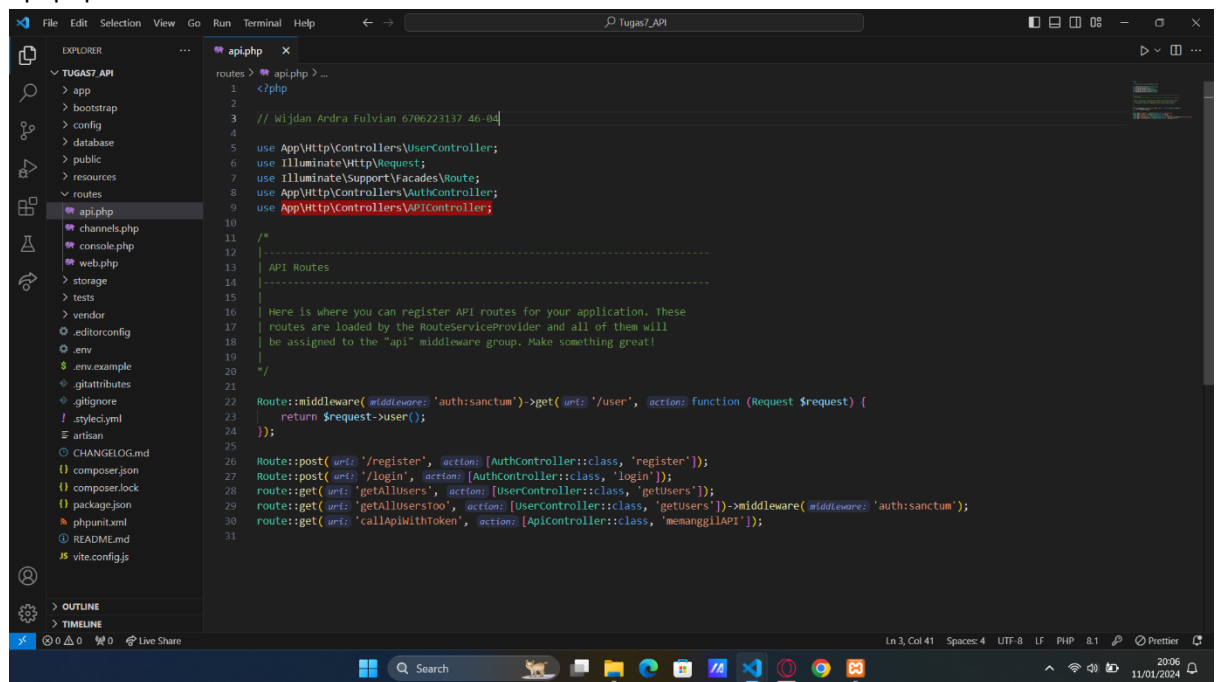


```

1 <?php
2
3 // Wijdan Andra Fulvian 6706223137 46-04
4
5 namespace App\Models;
6
7 // use Illuminate\Contracts\Auth\MustVerifyEmail;
8 use Illuminate\Database\Eloquent\Factories\HasFactory;
9 use Illuminate\Foundation\Auth\User as Authenticatable;
10 use Illuminate\Notifications\Notifiable;
11 use Laravel\Sanctum\HasApiTokens;
12
13 5 references | 0 implementations
14 class User extends Authenticatable
15 {
16     use HasApiTokens, HasFactory, Notifiable;
17
18     /**
19      * The attributes that are mass assignable.
20      *
21      * @var array<int, string>
22      */
23     0 references
24     protected $fillable = [
25         'name',
26         'email',
27         'password',
28     ];
29
30     /**
31      * The attributes that should be hidden for serialization.
32      *
33      * @var array<int, string>
34      */
35     0 references
36     protected $hidden = [
37         'password',
38         'remember_token',
39     ];
40
41     /**
42      * The attributes that should be cast.
43      *
44      * @var array<string, string>
45      */
46     0 references
47     protected $casts = [
48         'email_verified_at' => 'datetime',
49         'password' => 'hashed',
50     ];
51 }

```

- routes
- api.php

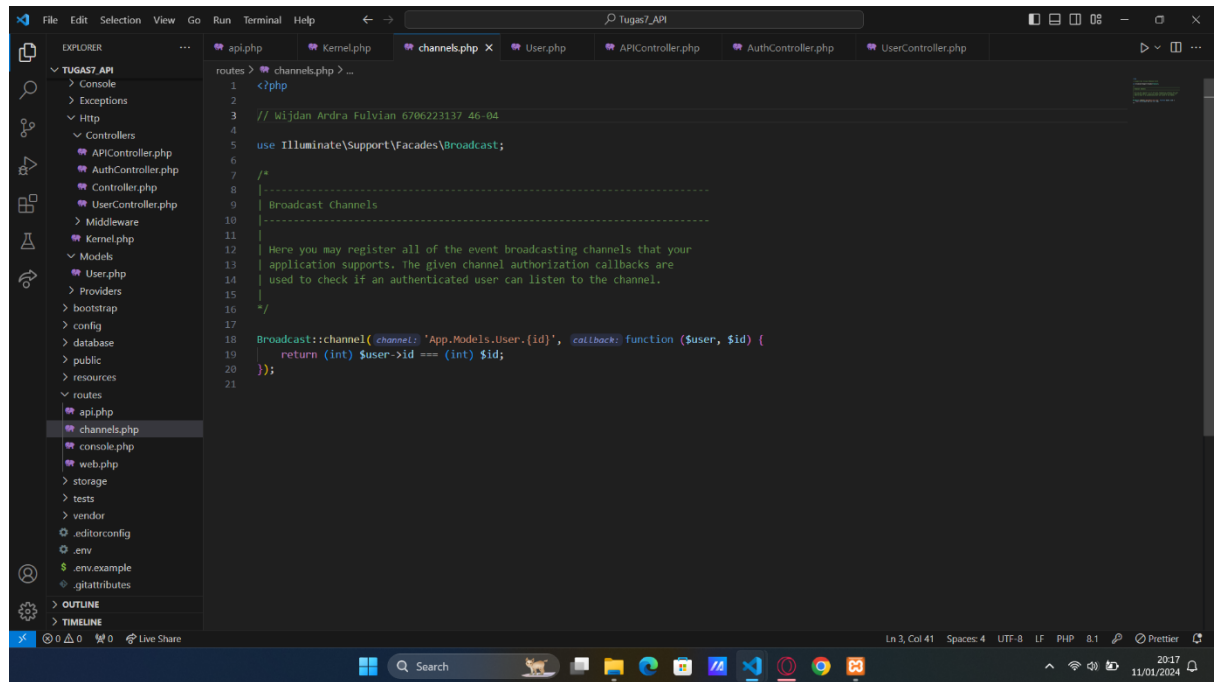


```

1 <?php
2
3 // Wijdan Andra Fulvian 6706223137 46-04
4
5 use App\Http\Controllers\UserController;
6 use Illuminate\Http\Request;
7 use Illuminate\Support\Facades\Route;
8 use App\Http\Controllers\AuthController;
9 use App\Http\Controllers\APIController;
10
11 /*
12 |-----|
13 | API Routes
14 |-----|
15 |
16 | Here is where you can register API routes for your application. These
17 | routes are loaded by the RouteServiceProvider and all of them will
18 | be assigned to the "api" middleware group. Make something great!
19 |
20 */
21
22 Route::middleware('auth:sanctum')->get('user', function (Request $request) {
23     return $request->user();
24 });
25
26 Route::post('register', [AuthController::class, 'register']);
27 Route::post('login', [AuthController::class, 'login']);
28 route::get('getAllUsers', [UserController::class, 'getUsers']);
29 route::get('getAllUserstoo', [UserController::class, 'getUsers'])->middleware('auth:sanctum');
30 route::get('callApiWithToken', [APIController::class, 'menanggilAPI']);
31

```

- channels.php

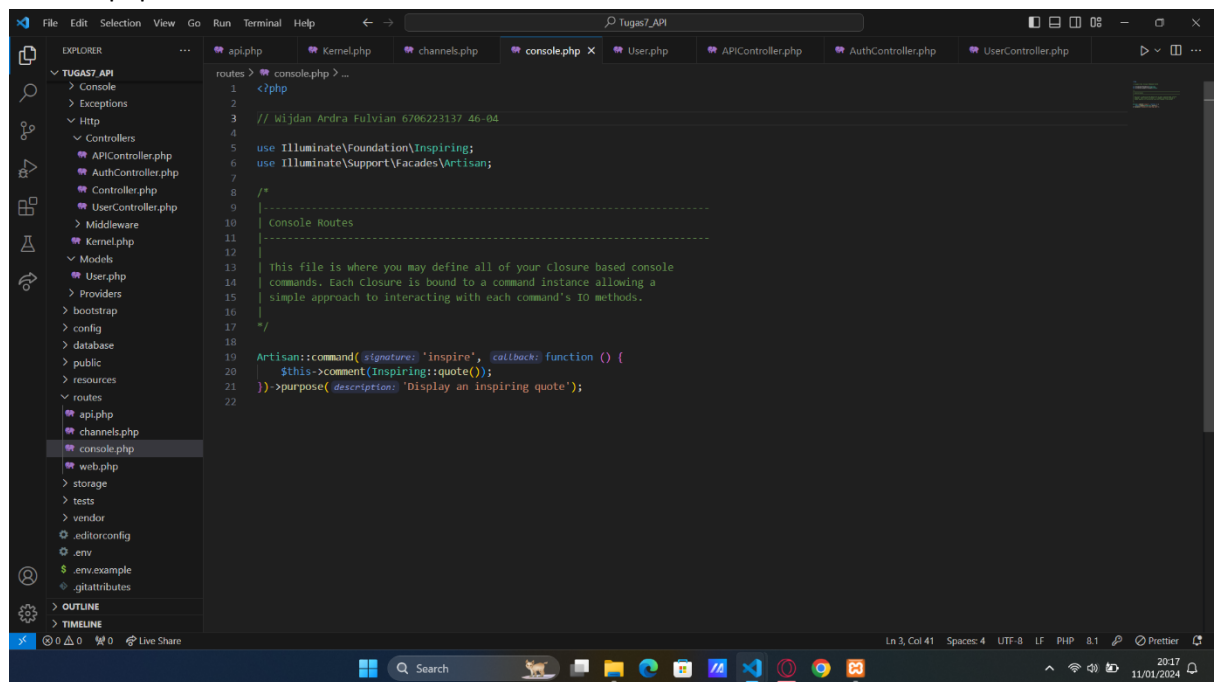


The screenshot shows the VS Code editor with the 'channels.php' file open. The Explorer sidebar on the left shows the project structure for 'TUGAS7\_API', with 'channels.php' selected under the 'routes' directory. The main editor area displays the following PHP code:

```
1 <?php
2
3 // Wijdan Andra Fulvian 6706223137 46-04
4
5 use Illuminate\Support\Facades\Broadcast;
6
7 /*
8 |-----
9 | Broadcast Channels
10 |-----
11 |
12 | Here you may register all of the event broadcasting channels that your
13 | application supports. The given channel authorization callbacks are
14 | used to check if an authenticated user can listen to the channel.
15 |
16 */
17
18 Broadcast::channel('App.Models.User.{id}', function ($user, $id) {
19     return (int) $user->id === (int) $id;
20 });
21
```

The status bar at the bottom indicates the file is at Line 3, Column 41, with 4 spaces, using UTF-8 encoding, LF line endings, PHP 8.1, and the Prettier formatter. The system clock shows 20:17 on 11/01/2024.

- console.php



The screenshot shows the VS Code editor with the 'console.php' file open. The Explorer sidebar on the left shows the project structure for 'TUGAS7\_API', with 'console.php' selected under the 'routes' directory. The main editor area displays the following PHP code:

```
1 <?php
2
3 // Wijdan Andra Fulvian 6706223137 46-04
4
5 use Illuminate\Foundation\Inspiring;
6 use Illuminate\Support\Facades\Artisan;
7
8 /*
9 |-----
10 | Console Routes
11 |-----
12 |
13 | This file is where you may define all of your Closure based console
14 | commands. Each Closure is bound to a command instance allowing a
15 | simple approach to interacting with each command's IO methods.
16 |
17 */
18
19 Artisan::command('inspire', function () {
20     $this->comment(Inspiring::quote());
21 })->purpose('Display an inspiring quote');
22
```

The status bar at the bottom indicates the file is at Line 3, Column 41, with 4 spaces, using UTF-8 encoding, LF line endings, PHP 8.1, and the Prettier formatter. The system clock shows 20:17 on 11/01/2024.

- web.php

```
routes > web.php > ...
1 <?php
2
3 // Wijdan Andra Fulvian 6706223137 46-04
4
5 use App\Http\Controllers\APIController;
6 use Illuminate\Support\Facades\Route;
7
8 /*
9 |-----
10 | Web Routes
11 |-----
12 |
13 | Here is where you can register web routes for your application. These
14 | routes are loaded by the RouteServiceProvider and all of them will
15 | be assigned to the "web" middleware group. Make something great!
16 |
17 */
18
19 Route::get('/', function () {
20     return view('welcome');
21 });
22 route::get('callApiWithToken', actions [APIController::class, 'memanggilAPI']);
23
```