

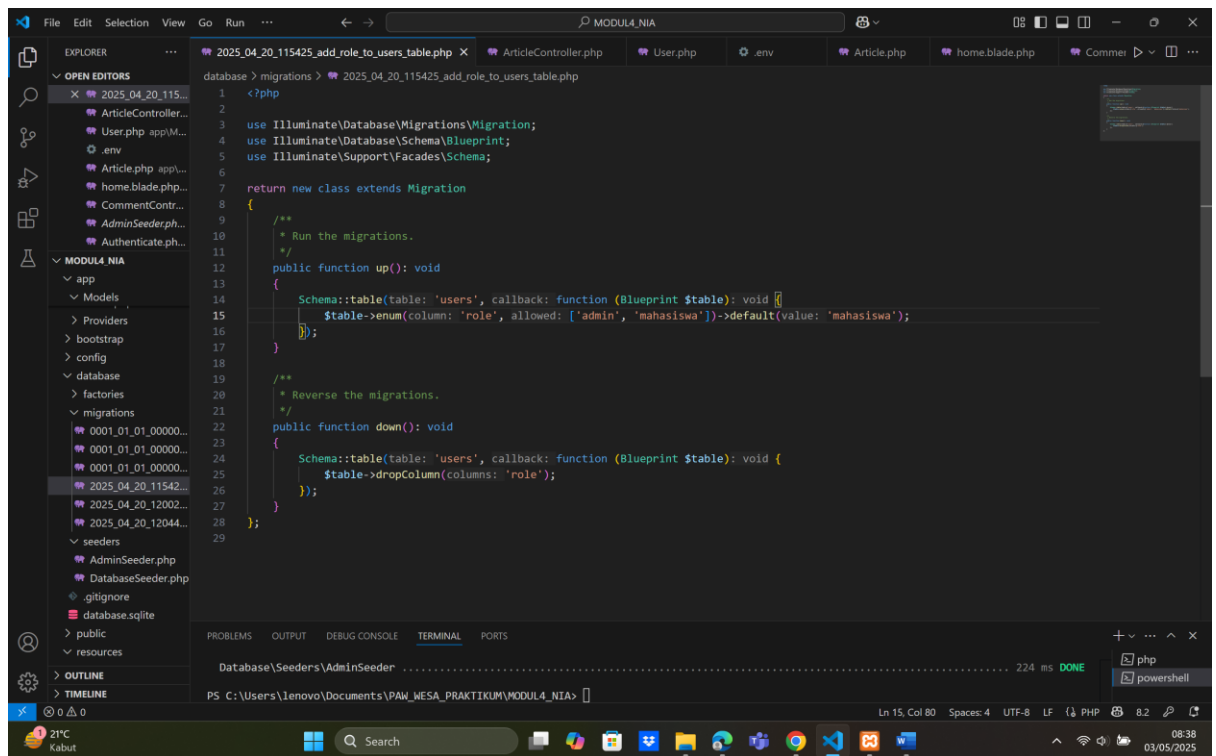
NAMA : RAHMANIA ANGGRAINI

KELAS : SI-47-08

NIM : 102022300034

## JURNAL MODUL 4

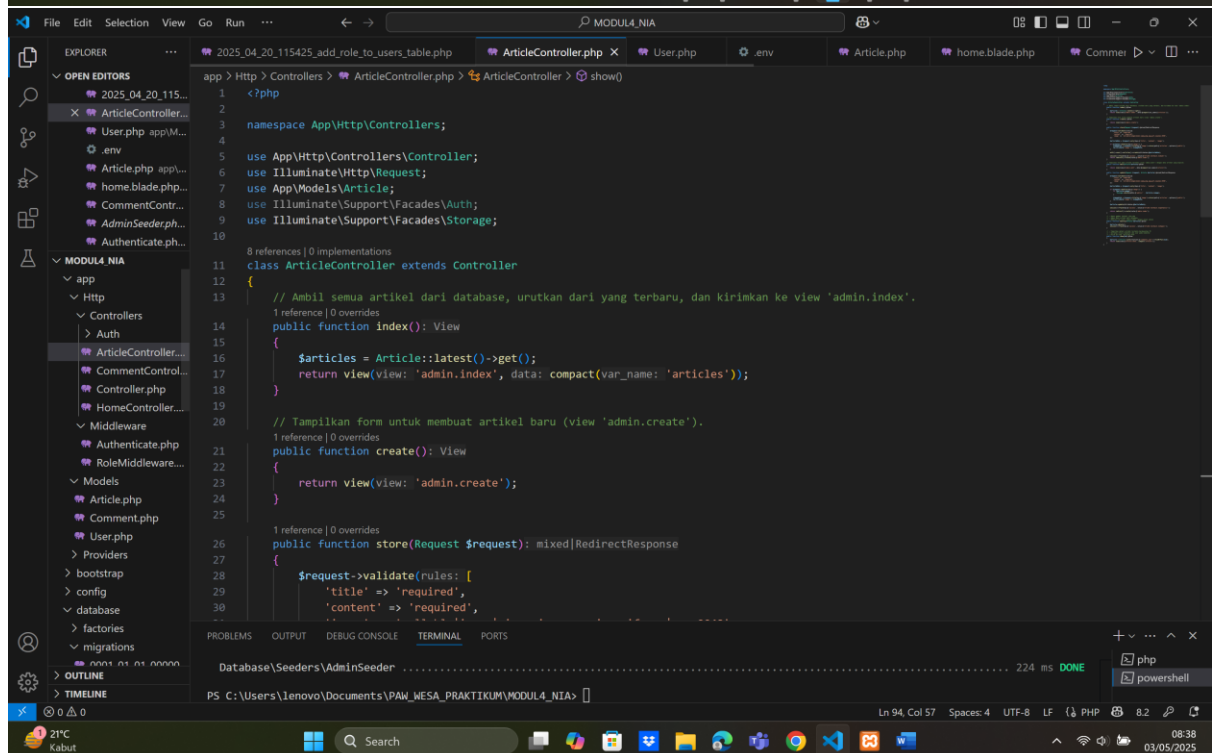
1.



The screenshot shows the Visual Studio Code editor with a project named 'MODUL4\_NIA'. The Explorer sidebar on the left shows the file structure, including 'migrations' and 'seeds' folders. The main editor window displays the file '2025\_04\_20\_115425\_add\_role\_to\_users\_table.php' within the 'migrations' directory. The code is a PHP migration class that extends 'Migration'. It includes comments for running and reversing migrations. The 'up()' method uses 'Schema::table()' to add a 'role' column to the 'users' table, with a default value of 'mahasiswa'. The 'down()' method uses 'Schema::table()' to drop the 'role' column. The bottom status bar shows the file is at line 15, column 80, and the terminal output indicates a successful execution of the 'Database\Seeders\AdminSeeder' command.

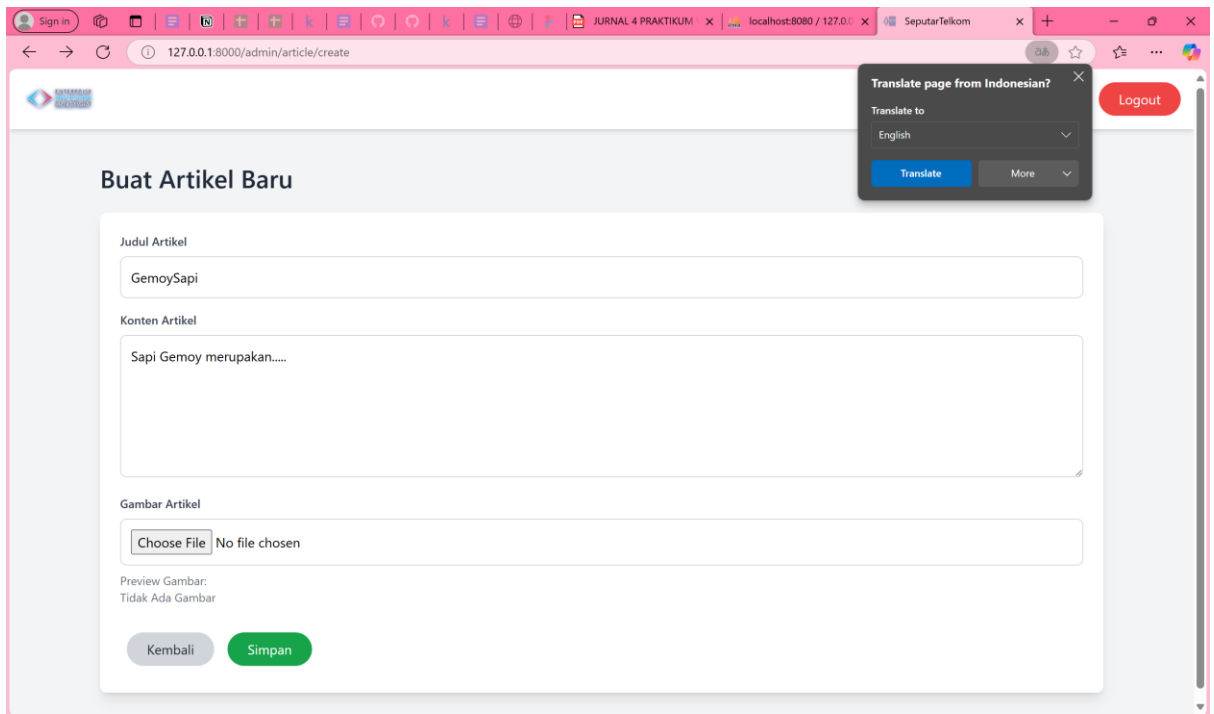
```
1 <?php
2
3 use Illuminate\Database\Migrations\Migration;
4 use Illuminate\Database\Schema\Blueprint;
5 use Illuminate\Support\Facades\Schema;
6
7 return new class extends Migration
8 {
9     /**
10      * Run the migrations.
11      */
12     public function up(): void
13     {
14         Schema::table('users', callback: function (Blueprint $table): void {
15             $table->enum('role', allowed: ['admin', 'mahasiswa'])->default(value: 'mahasiswa');
16         });
17     }
18
19     /**
20      * Reverse the migrations.
21      */
22     public function down(): void
23     {
24         Schema::table('users', callback: function (Blueprint $table): void {
25             $table->dropColumn('role');
26         });
27     }
28 };
29
```

2.

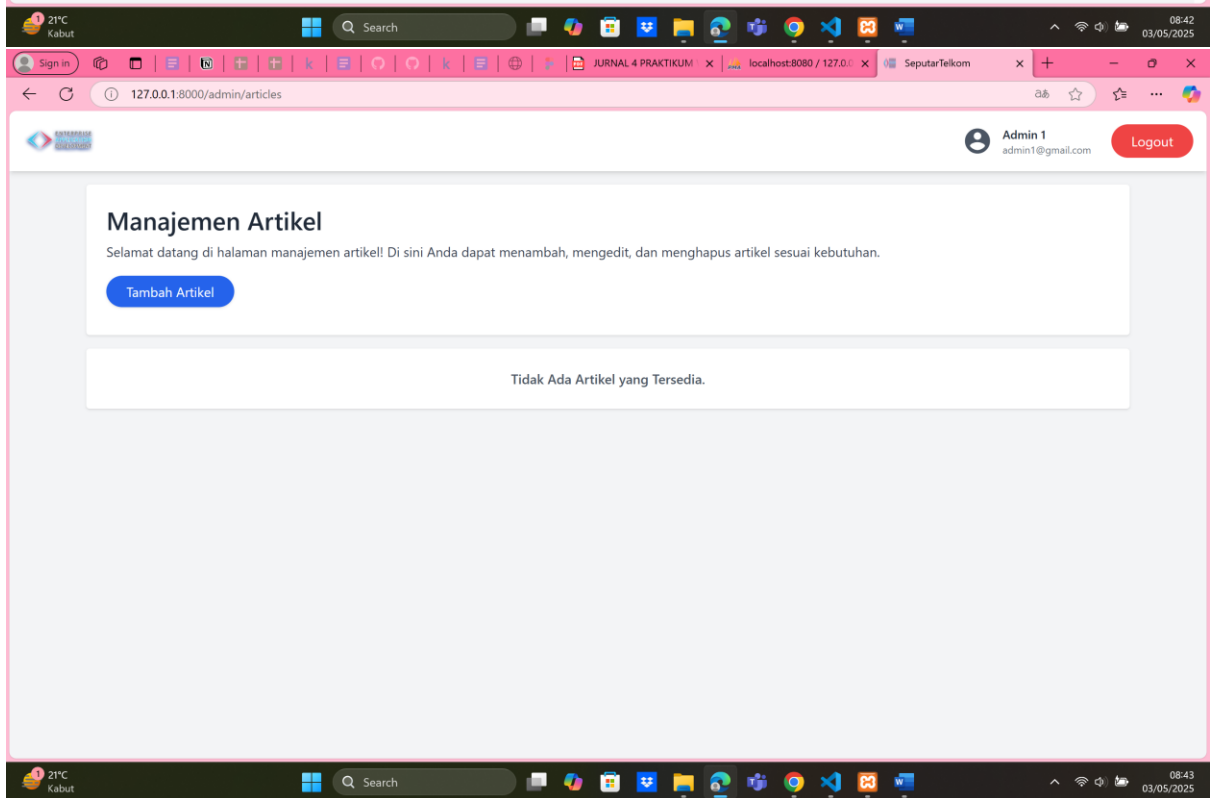


The screenshot shows the Visual Studio Code editor with the same project 'MODUL4\_NIA'. The Explorer sidebar shows the 'Controllers' folder. The main editor window displays the file 'ArticleController.php' within the 'Controllers' directory. The code is a PHP controller class that extends 'Controller'. It includes comments for the 'index()' and 'create()' methods. The 'index()' method uses 'Article::latest()' to get the latest articles and returns a view. The 'create()' method returns a view for creating a new article. The 'store()' method uses '\$request->validate()' to validate the request data. The bottom status bar shows the file is at line 94, column 57, and the terminal output indicates a successful execution of the 'Database\Seeders\AdminSeeder' command.

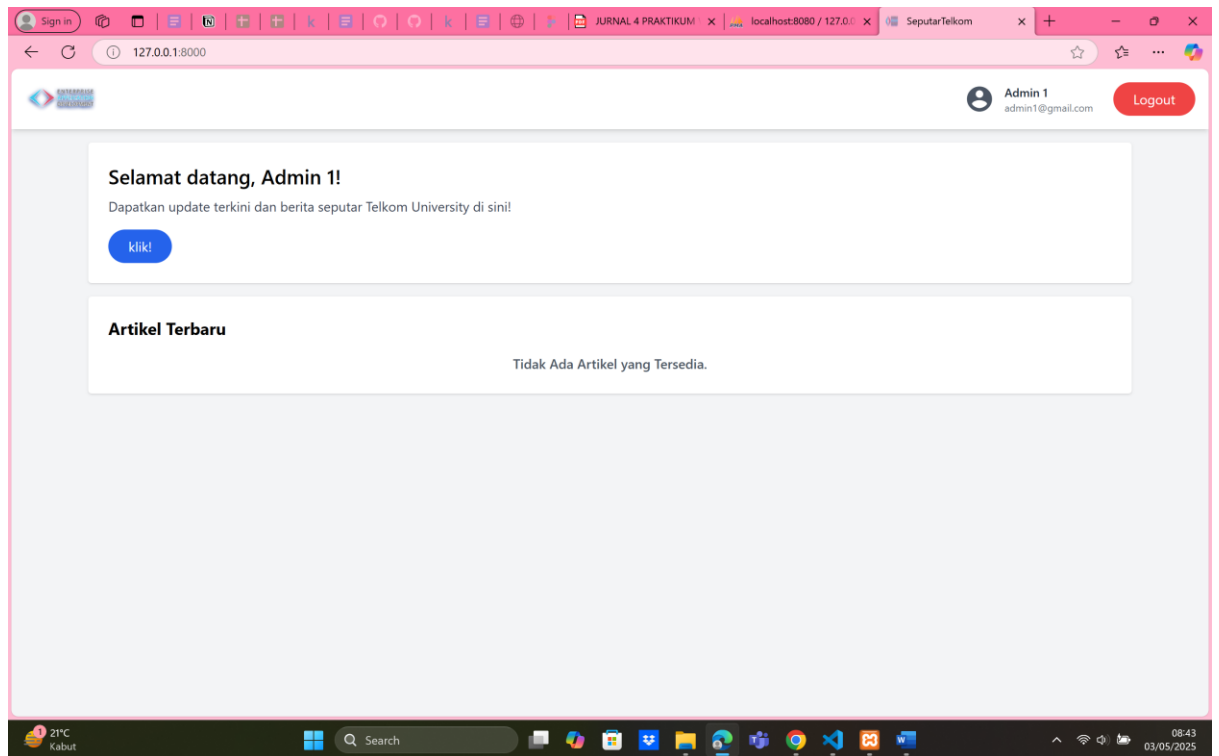
```
1 <?php
2
3 namespace App\Http\Controllers;
4
5 use App\Http\Controllers\Controller;
6 use Illuminate\Http\Request;
7 use App\Models\Article;
8 use Illuminate\Support\Facades\Auth;
9 use Illuminate\Support\Facades\Storage;
10
11 class ArticleController extends Controller
12 {
13     // Ambil semua artikel dari database, urutkan dari yang terbaru, dan kirimkan ke view 'admin.index'.
14     public function index(): View
15     {
16         $articles = Article::latest()->get();
17         return view('admin.index', data: compact('articles'));
18     }
19
20     // Tampilkan form untuk membuat artikel baru (view 'admin.create').
21     public function create(): View
22     {
23         return view('admin.create');
24     }
25
26     public function store(Request $request): mixed|RedirectResponse
27     {
28         $request->validate(rules: [
29             'title' => 'required',
30             'content' => 'required',
31         ]);
32     }
33 }
```



3.



4.



5.