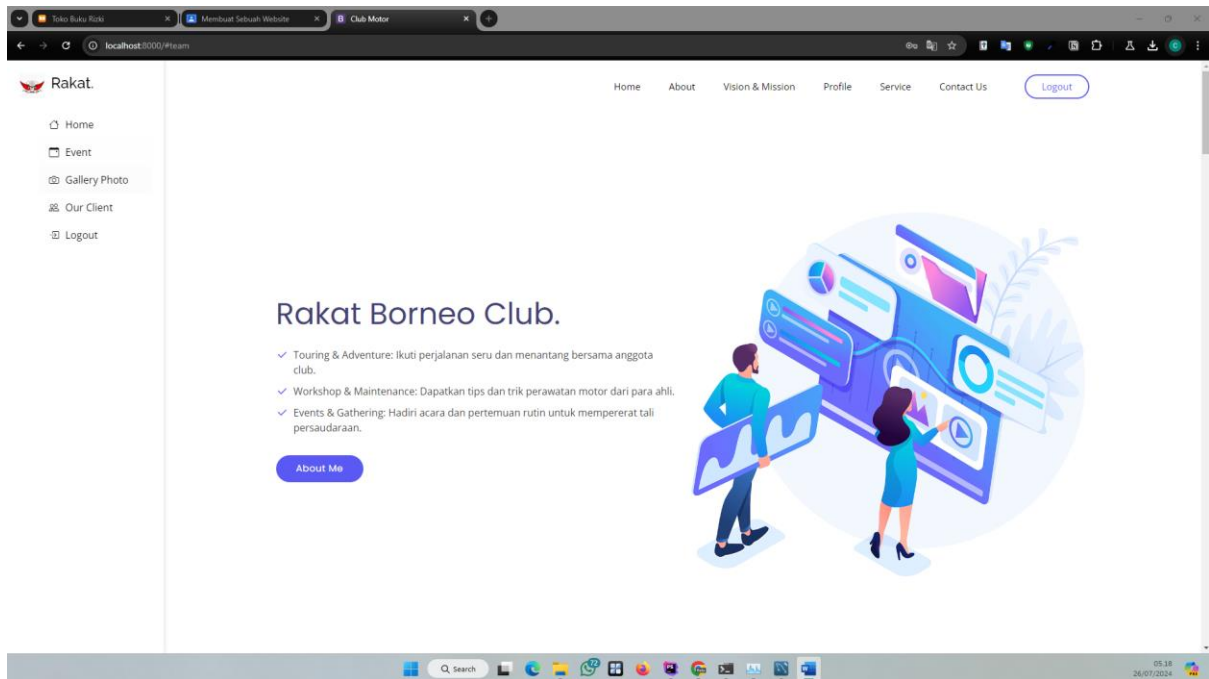


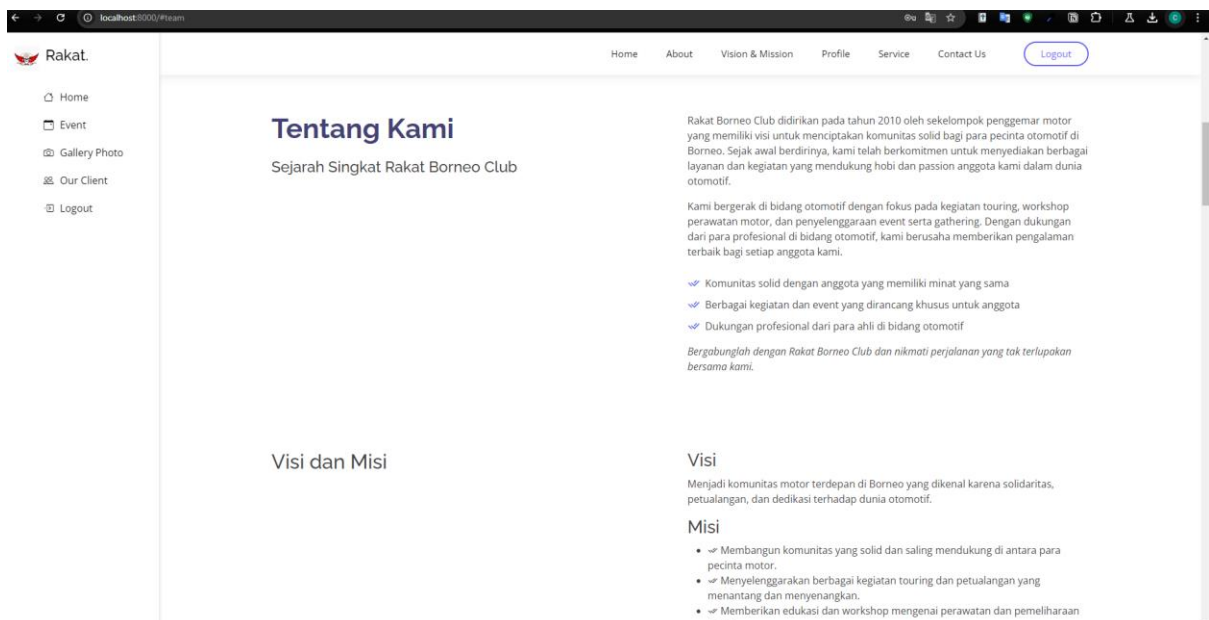
NAMA : Muhammad Rizki Firdaus

NPM : 2210010466

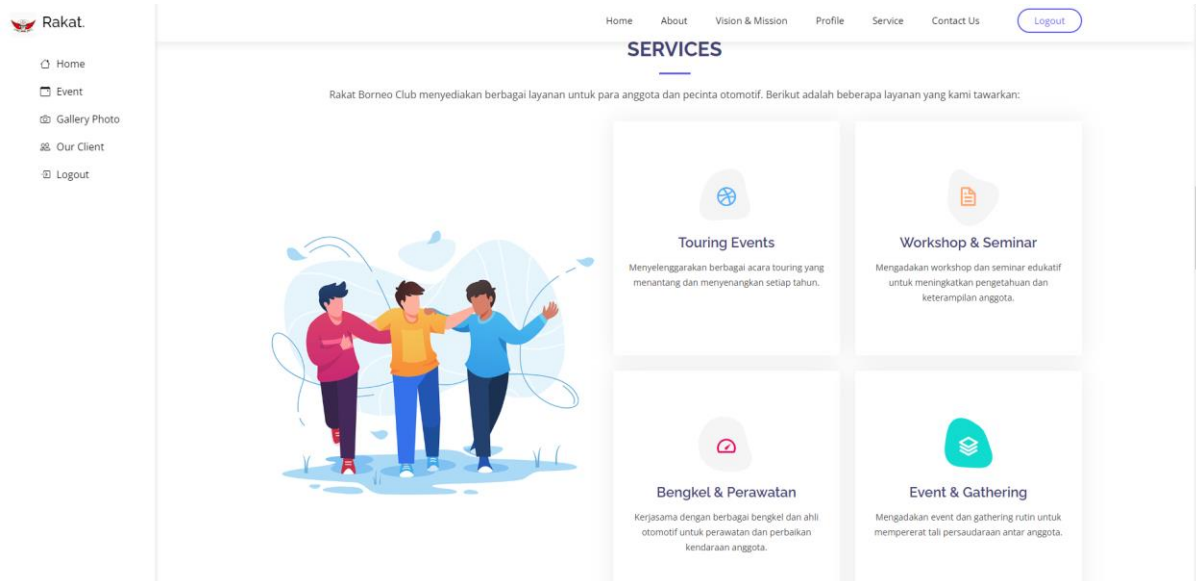
UNIVERSITAS : UNISKA



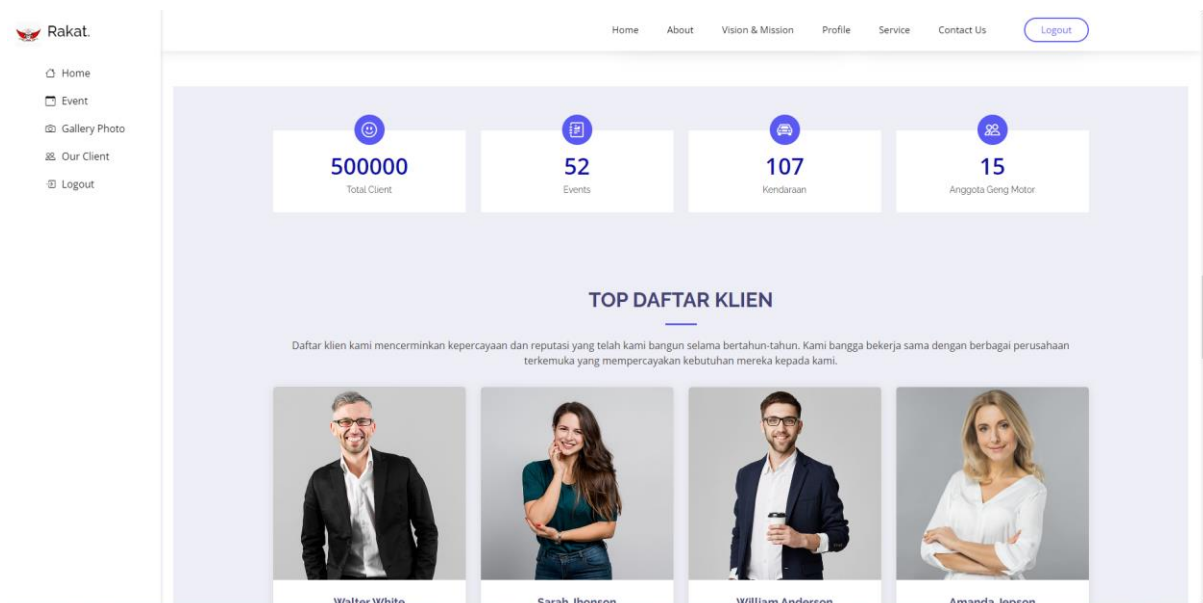
Tampilan Depan Website



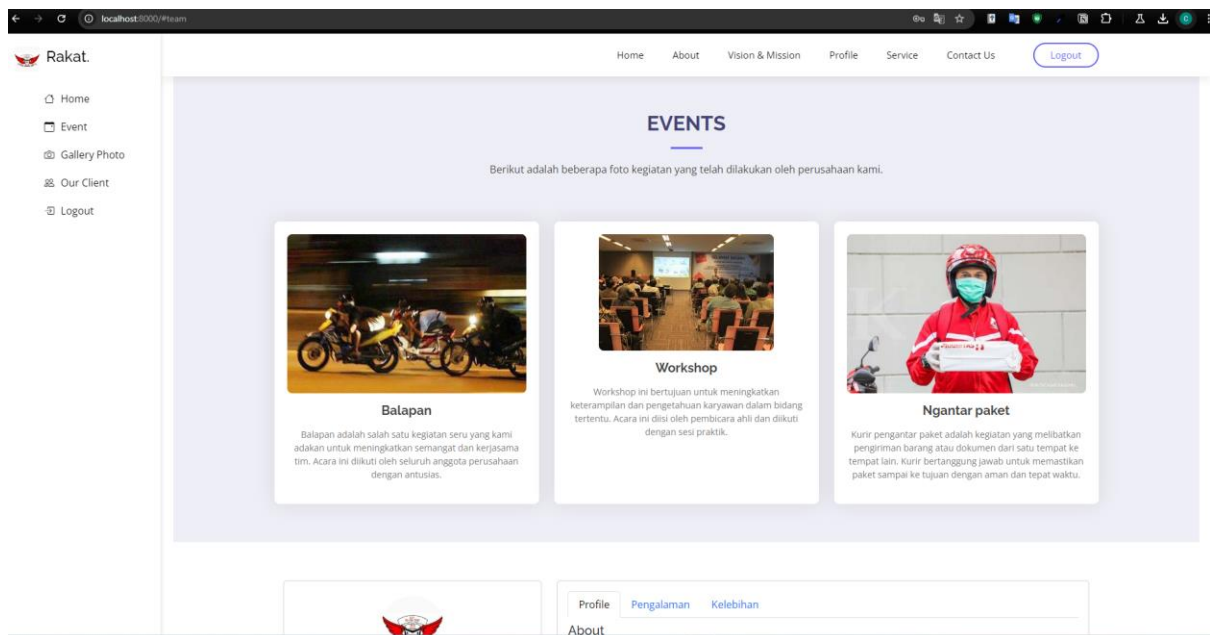
Tampilan About Us



Tampilan Service/ Jasa yang ada di Perusahaan



Tampilan List Client



Tampilan Events

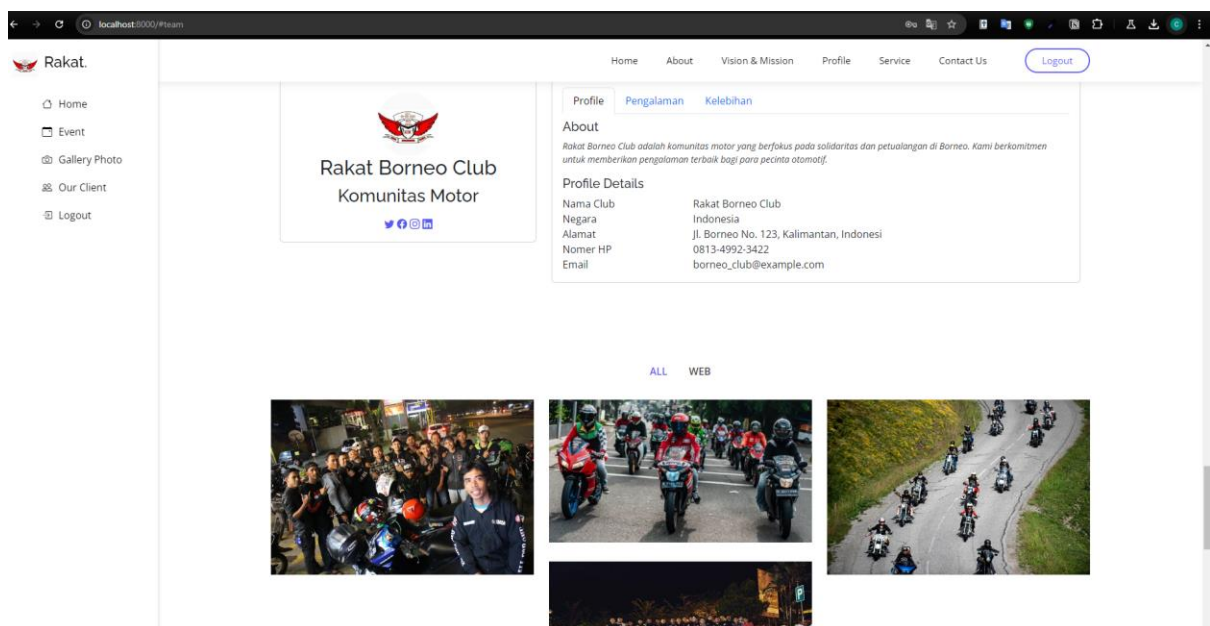
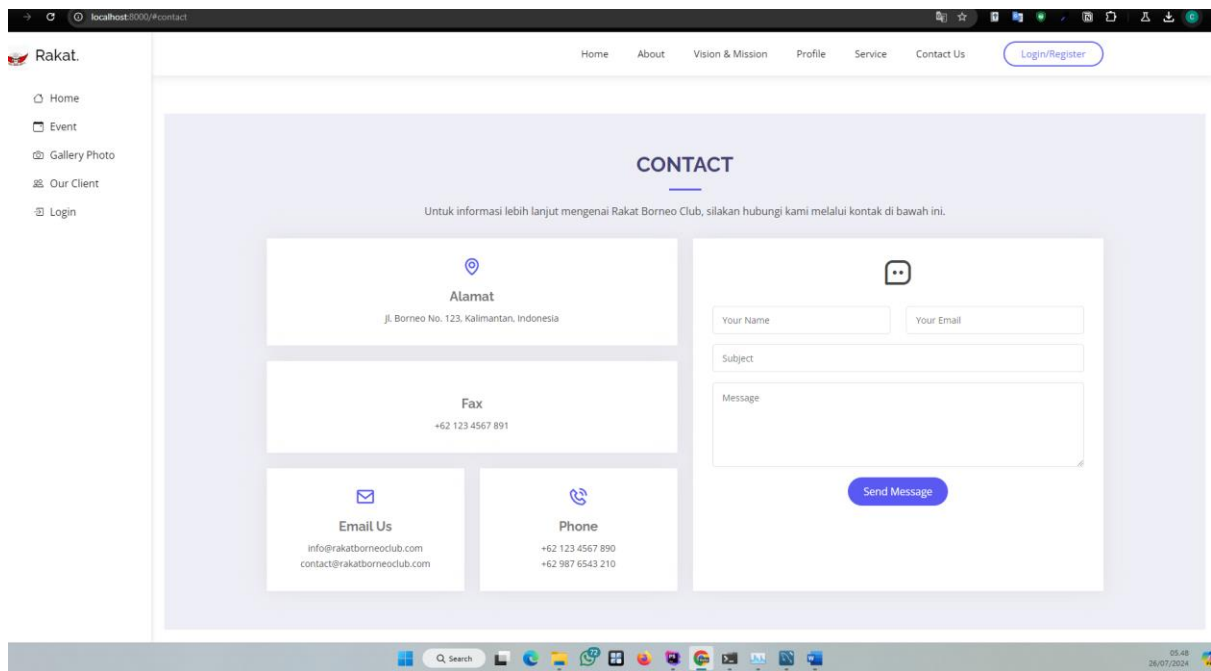
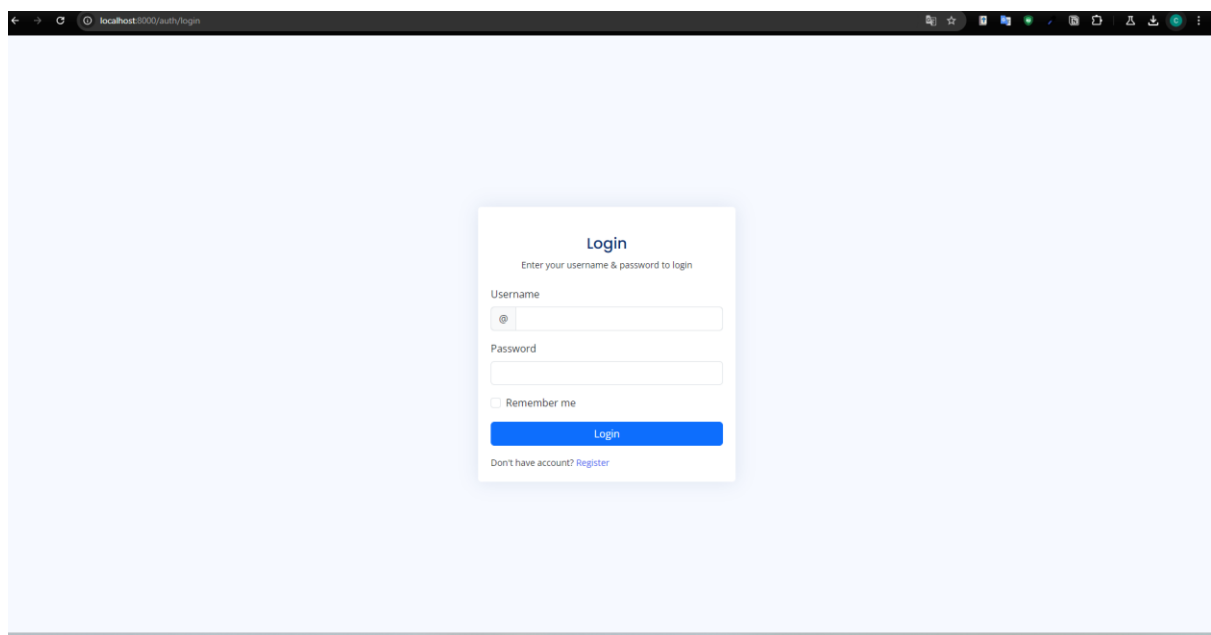


Image Galery dan Biodata

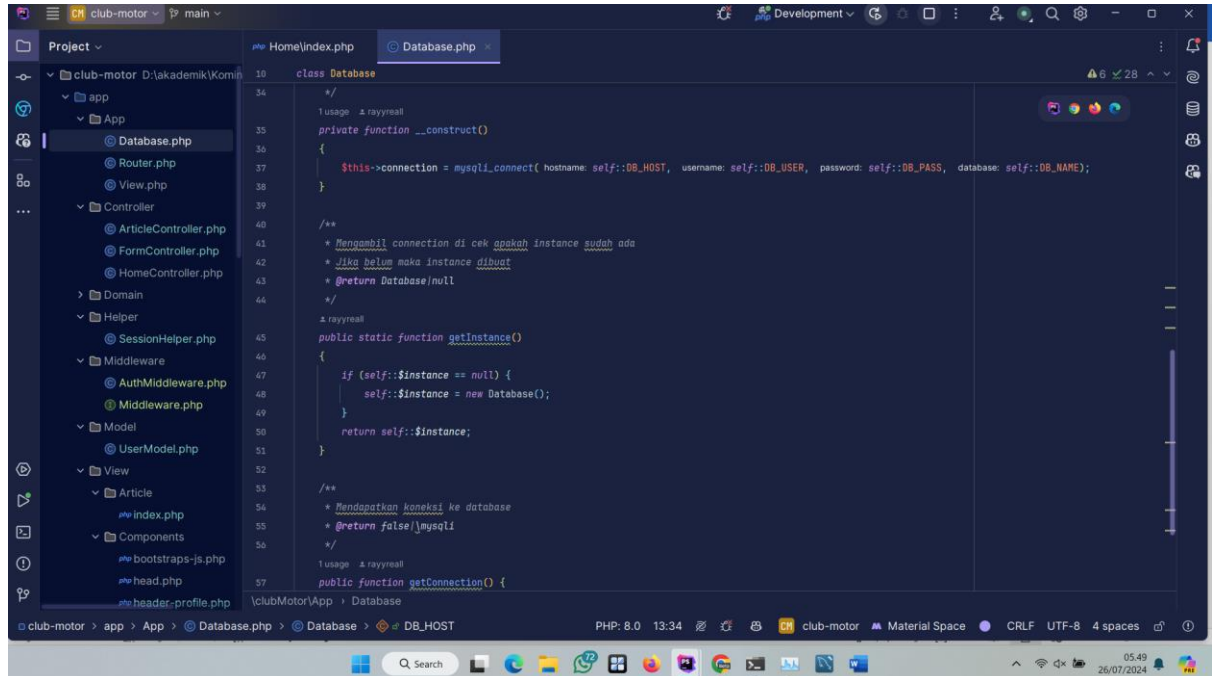


Halaman Contacts



Coding:

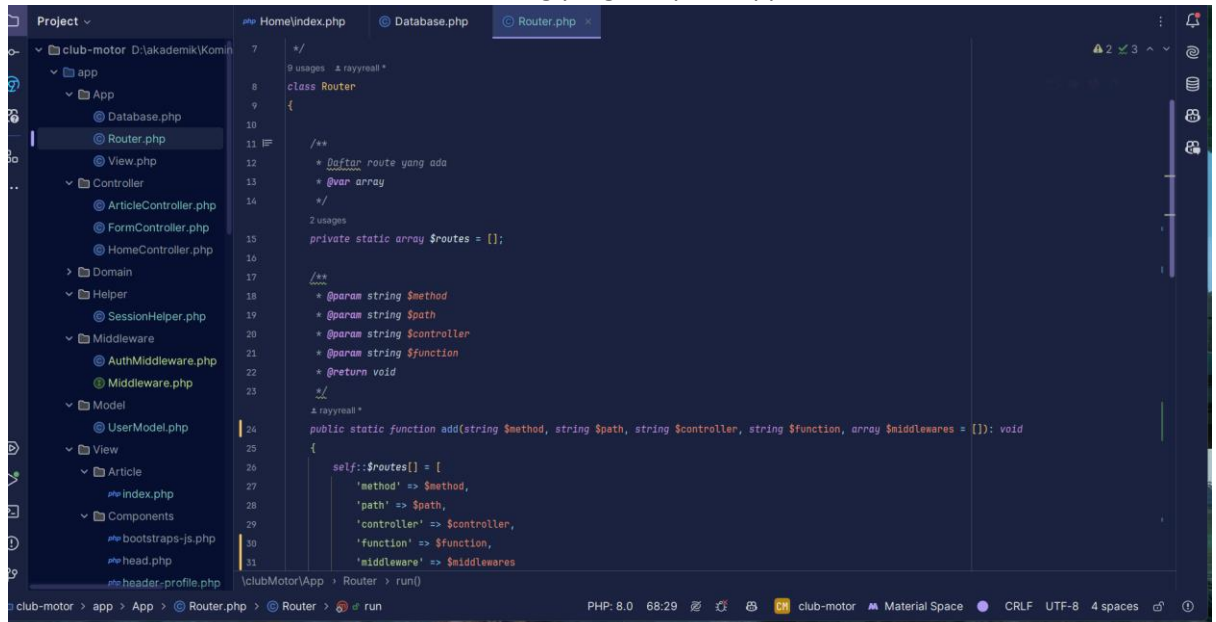
1. Koneksi ke database di App\Database.php



The screenshot shows a code editor with the 'club-motor' project open. The file 'Database.php' is selected in the 'App' directory. The code defines a 'Database' class with a private constructor, a static 'getInstance()' method, and a 'getConnection()' method. The 'getConnection()' method uses 'mysqli_connect' to establish a database connection.

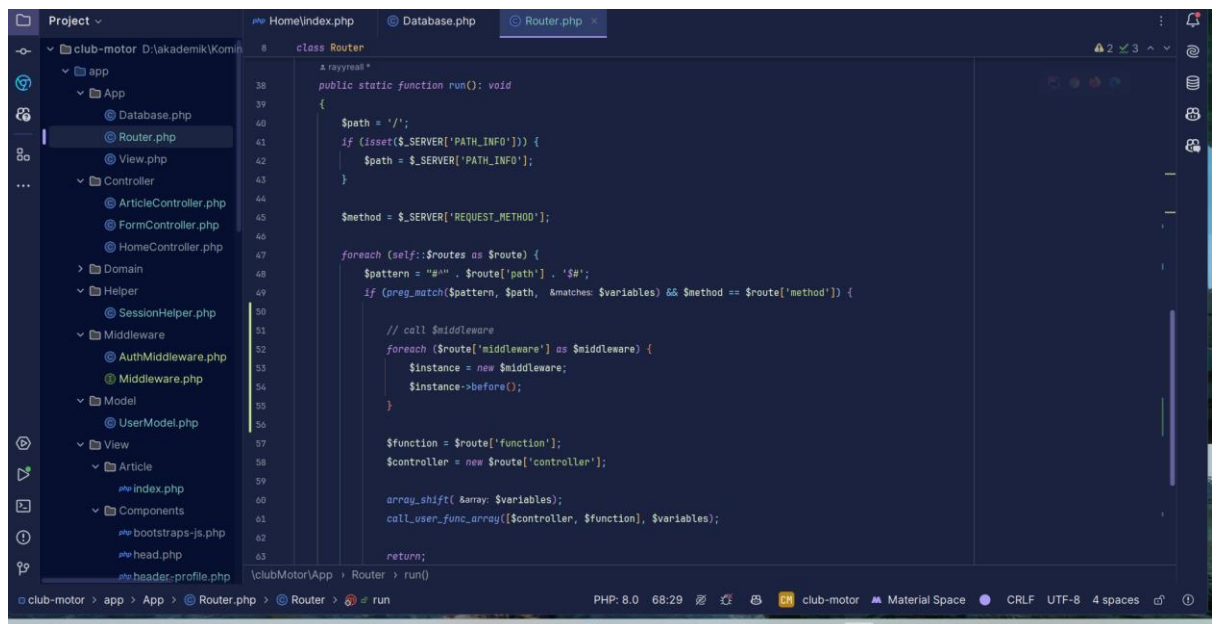
```
18 class Database
19 {
20     //
21     // usage & rayreal
22     private function __construct()
23     {
24         $this->connection = mysqli_connect( hostname: self::DB_HOST, username: self::DB_USER, password: self::DB_PASS, database: self::DB_NAME);
25     }
26
27     /**
28      * Mengambil connection di cek apakah instance sudah ada
29      * jika belum maka instance dibuat
30      * @return Database|null
31      */
32     //
33     // usage & rayreal
34     public static function getInstance()
35     {
36         if (self::$instance == null) {
37             self::$instance = new Database();
38         }
39         return self::$instance;
40     }
41
42     /**
43      * Mendapatkan koneksi ke database
44      * @return false|mysqli
45      */
46     //
47     // usage & rayreal
48     public function getConnection() {
49
50     }
51 }
```

2. Class Router untuk melakukan handle Routing yang ada pada aplikasi



The screenshot shows a code editor with the 'club-motor' project open. The file 'Router.php' is selected in the 'App' directory. The code defines a 'Router' class with a private static array '\$routes', a 'add()' method, and a 'run()' method. The 'add()' method adds a new route to the '\$routes' array, and the 'run()' method handles the routing.

```
7 //
8 // usage & rayreal *
9 class Router
10 {
11     //
12     // Daftar route yang ada
13     // @var array
14     //
15     // usage
16     private static array $routes = [];
17
18     /**
19      * @param string $method
20      * @param string $path
21      * @param string $controller
22      * @param string $function
23      * @return void
24      */
25     //
26     // usage & rayreal *
27     public static function add(string $method, string $path, string $controller, string $function, array $middlewares = []): void
28     {
29         self::$routes[] = [
30             'method' => $method,
31             'path' => $path,
32             'controller' => $controller,
33             'function' => $function,
34             'middlewares' => $middlewares
35         ];
36     }
37
38     //
39     // usage & rayreal *
40     public function run()
41     {
42
43     }
44 }
```



The screenshot shows an IDE with a project named 'club-motor'. The file explorer on the left shows the project structure, including folders for 'app', 'Controller', 'Domain', 'Helper', 'Middleware', 'Model', and 'View'. The 'Router.php' file is selected in the 'app' folder. The main editor displays the code for the 'Router' class, which is a rayyreal class. The code includes a 'run()' method that handles incoming requests by matching patterns, calling middleware, and executing the appropriate controller function. The status bar at the bottom indicates the file is 'club-motor' and the editor is in 'Material Space' mode.

```
class Router
{
    public static function run(): void
    {
        $path = '/';
        if (isset($_SERVER['PATH_INFO'])) {
            $path = $_SERVER['PATH_INFO'];
        }

        $method = $_SERVER['REQUEST_METHOD'];

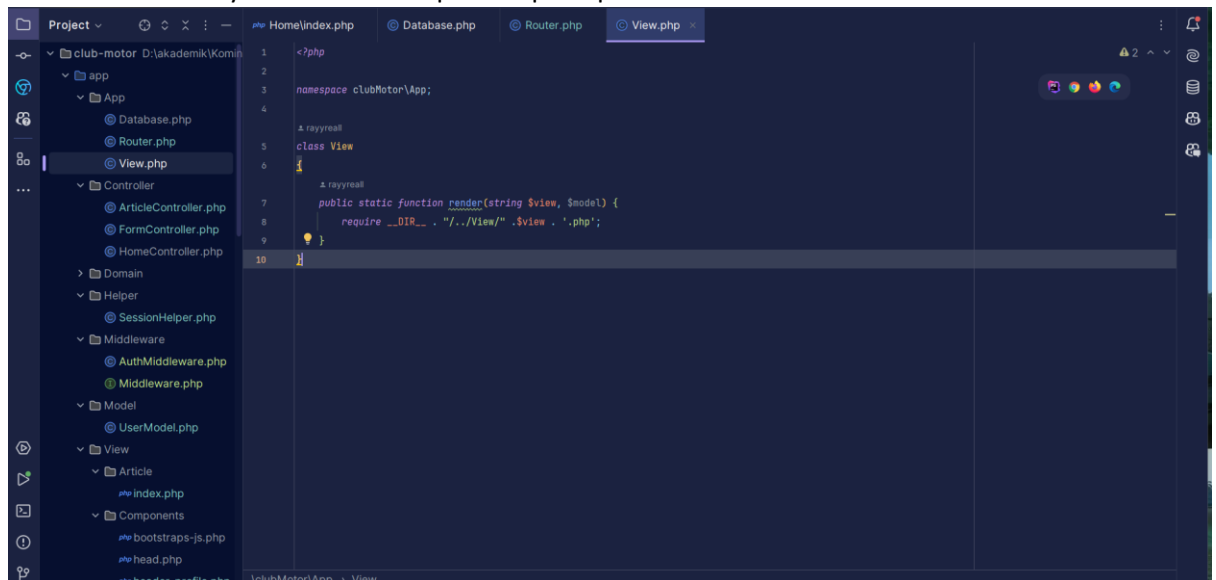
        foreach (self::$routes as $route) {
            $pattern = "#^" . $route['path'] . "$#";
            if (preg_match($pattern, $path, &matches: $variables) && $method == $route['method']) {
                // call $middleware
                foreach ($route['middleware'] as $middleware) {
                    $instance = new $middleware;
                    $instance->before();
                }

                $function = $route['function'];
                $controller = new $route['controller'];

                array_shift(&array: $variables);
                call_user_func_array([$controller, $function], $variables);

                return;
            }
        }
    }
}
```

3. Membaca directory View untuk tampilan depan aplikasi



The screenshot shows the same IDE with the 'View.php' file selected in the 'app' folder. The main editor displays the code for the 'View' class, which is a rayyreal class. The code includes a 'render()' method that takes a view name and a model, and renders the view by including the corresponding PHP file. The status bar at the bottom indicates the file is 'club-motor' and the editor is in 'Material Space' mode.

```
<?php
namespace clubMotor\App;

class View
{
    public static function render(string $view, $model) {
        require __DIR__ . "/../View/" . $view . ".php";
    }
}
```

4. Controller Untuk Handle pada Logic Aplikasi

club-motor > main

Project

- club-motor
- app
 - App
 - Database.php
 - Router.php
 - View.php
 - Controller
 - ArticleController.php
 - FormController.php
 - HomeController.php
 - Domain
 - Helper
 - SessionHelper.php
 - Middleware
 - AuthMiddleware.php
 - Middleware.php
 - Model
 - UserModel.php
 - View
 - Article
 - index.php
 - Components
 - bootstrap.js.php
 - head.php
 - header-profile.php

Home/index.php

```
<?php
namespace clubMotor\Controller;

use clubMotor\App\View;

class HomeController
{
    public function index(): void
    {
        View::render( view: "Home/index", model: null);
    }
}
```

club-motor > app > Controller > HomeController.php > HomeController

PHP: 8.0 7:7 club-motor Material Space CRLF UTF-8 4 spaces

club-motor > app

Project

- club-motor
- app
 - App
 - Database.php
 - Router.php
 - View.php
 - Controller
 - ArticleController.php
 - FormController.php
 - HomeController.php
 - Domain
 - Helper
 - SessionHelper.php
 - Middleware
 - AuthMiddleware.php
 - Middleware.php
 - Model
 - UserModel.php
 - View
 - Article
 - index.php
 - Components
 - bootstrap.js.php
 - head.php
 - header-profile.php

Home/index.php

FormController.php

```
class FormController
{
    public function login(): void
    {
        View::render( view: "Form/login", []);
    }

    public function register(): void
    {
        SessionHelper::startSession();
        View::render( view: "Form/register", []);
    }

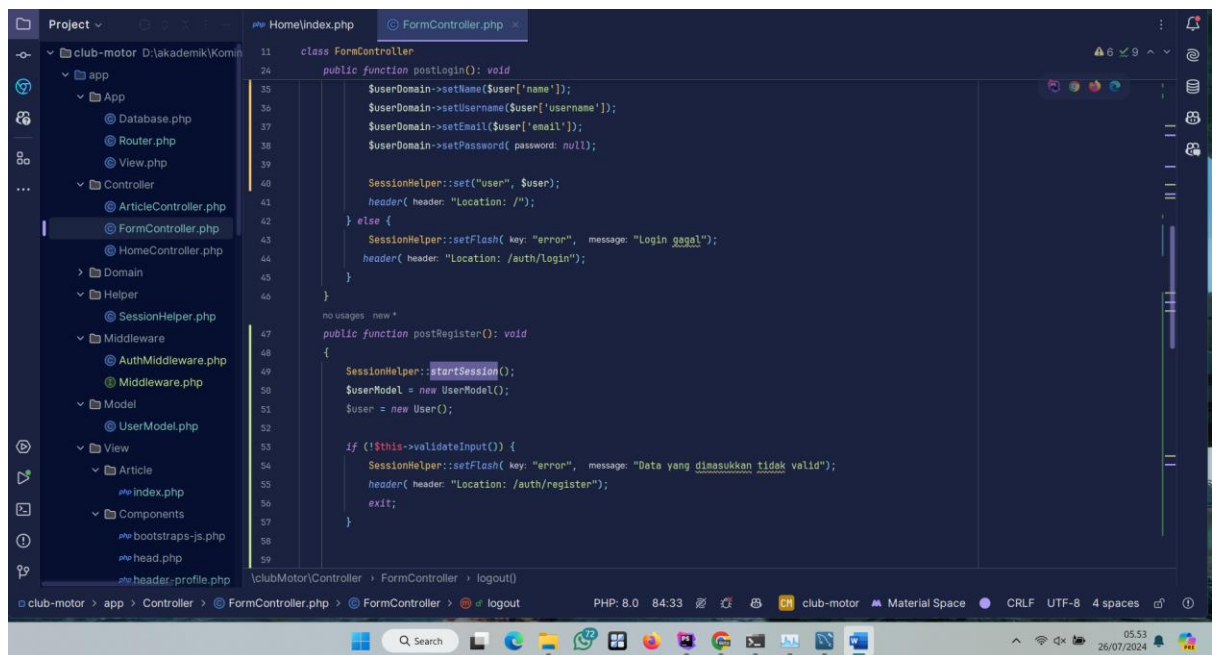
    public function postLogin(): void
    {
        SessionHelper::startSession();
        UserModel = new \clubMotor\Model\UserModel();

        $user = $userModel->findByUsernameAndPassword($_POST['username'], $_POST['password']);

        if ($user) {
            $userDomain = new User();
            $userDomain->setId((int) $user['id']);
            $userDomain->setName($user['name']);
            $userDomain->setUsername($user['username']);
            $userDomain->setEmail($user['email']);
            $userDomain->setPassword( password: null);
        }
    }
}
```

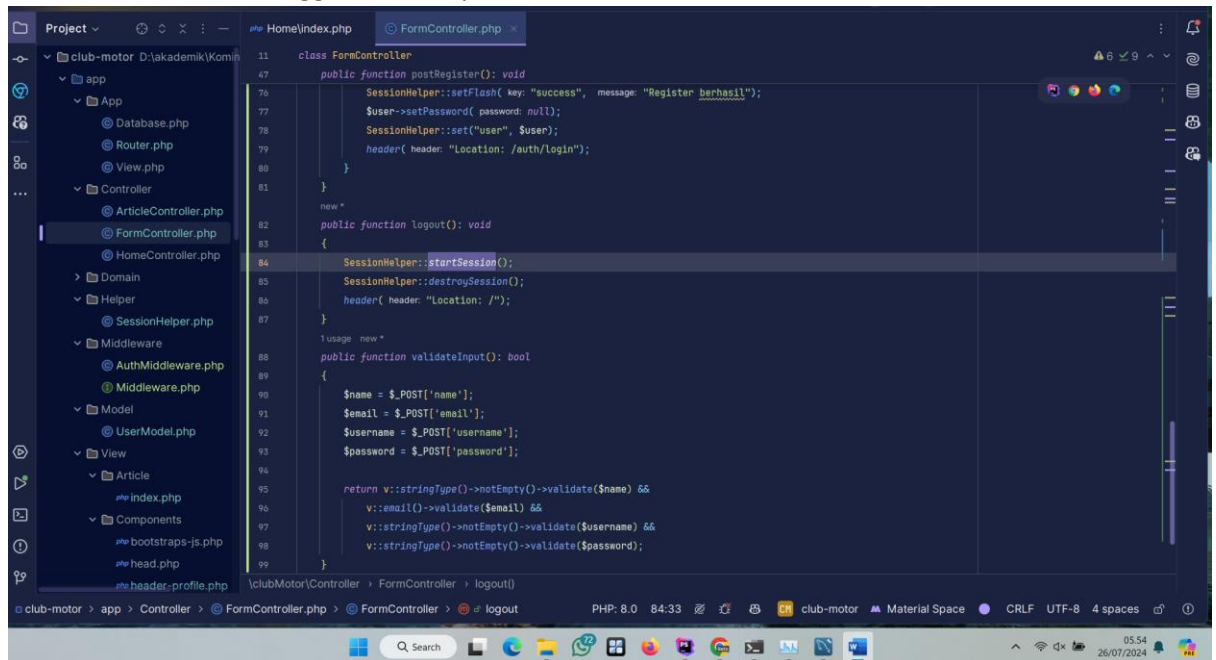
club-motor > app > Controller > FormController.php > FormController > logout

PHP: 8.0 84:33 club-motor Material Space CRLF UTF-8 4 spaces



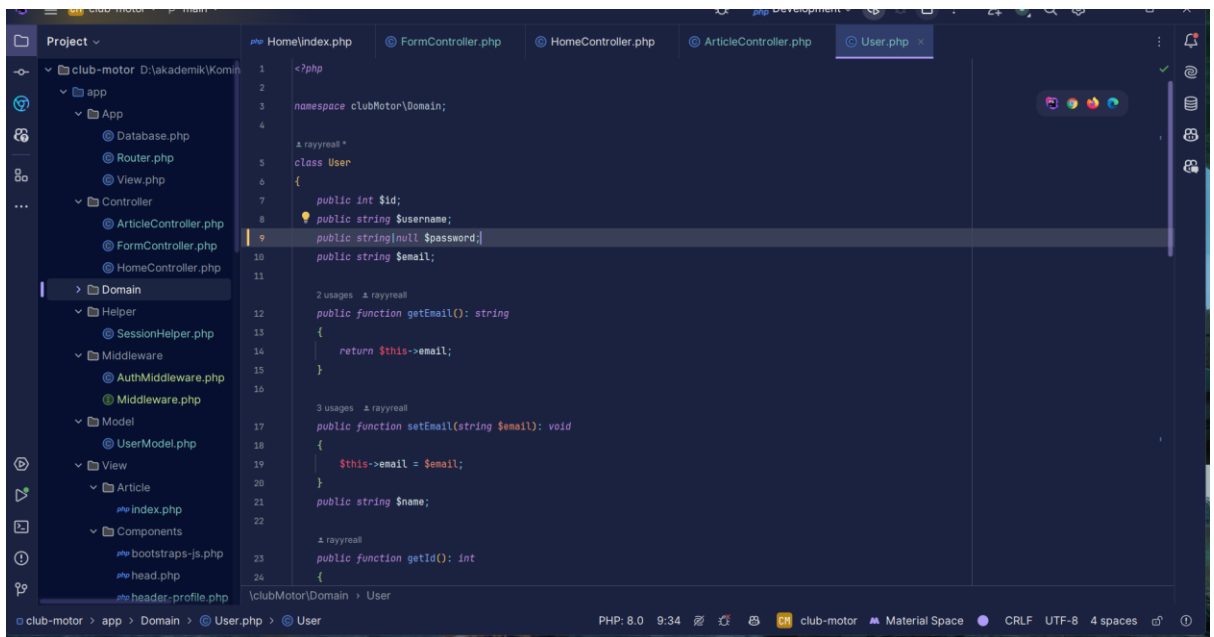
```
11 class FormController
24     public function postLogin(): void
35     {
36         $userDomain->setName($user['name']);
37         $userDomain->setName($user['username']);
38         $userDomain->setEmail($user['email']);
39         $userDomain->setPassword($password);
40
41         SessionHelper::set("user", $user);
42         header( header: "Location: /");
43     } else {
44         SessionHelper::setFlash( key: "error", message: "Login gagal");
45         header( header: "Location: /auth/login");
46     }
47 }
48
49 no usages new *
50 public function postRegister(): void
51 {
52     SessionHelper::startSession();
53     $userModel = new UserModel();
54     $user = new User();
55
56     if (!$this->validateInput()) {
57         SessionHelper::setFlash( key: "error", message: "Data yang dimasukkan tidak valid");
58         header( header: "Location: /auth/register");
59         exit;
60     }
61 }
```

5. Melakukan Validasi Menggunakan Respect\Validation



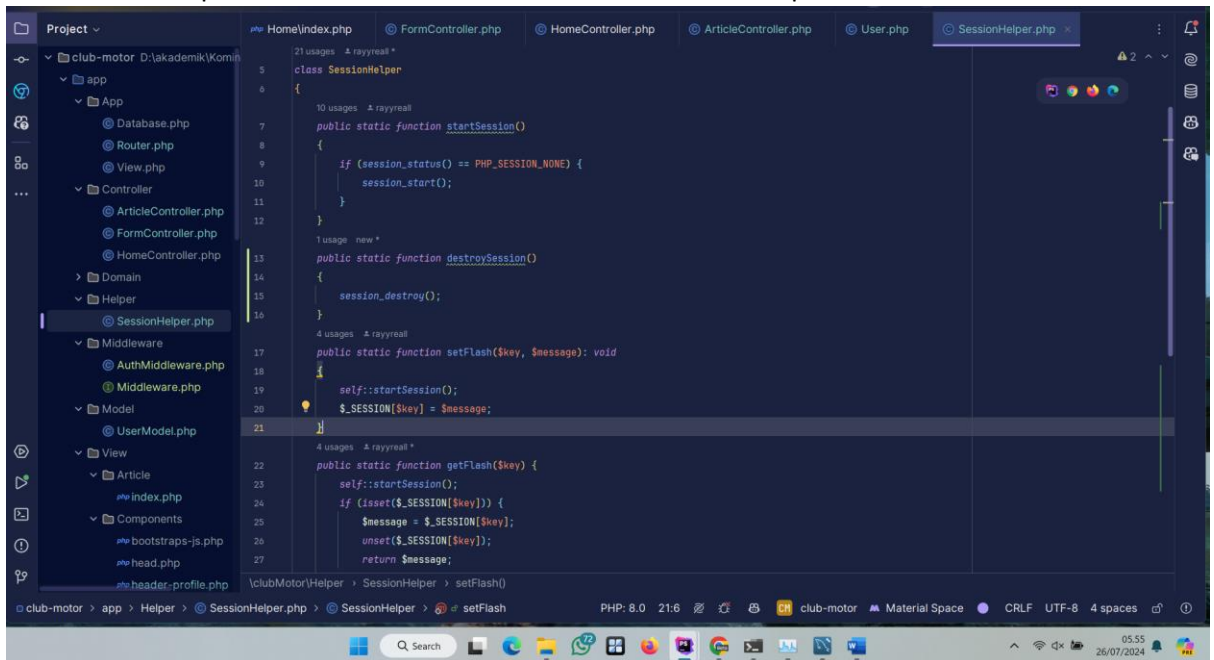
```
76     SessionHelper::setFlash( key: "success", message: "Register berhasil");
77     $user->setPassword($password);
78     SessionHelper::set("user", $user);
79     header( header: "Location: /auth/login");
80 }
81
82 new *
83 public function logout(): void
84 {
85     SessionHelper::startSession();
86     SessionHelper::destroySession();
87     header( header: "Location: /");
88 }
89
90 1 usage new *
91 public function validateInput(): bool
92 {
93     $name = $_POST['name'];
94     $email = $_POST['email'];
95     $username = $_POST['username'];
96     $password = $_POST['password'];
97
98     return v::stringType()->notEmpty()->validate($name) &&
99         v::email()->validate($email) &&
100         v::stringType()->notEmpty()->validate($username) &&
101         v::stringType()->notEmpty()->validate($password);
102 }
```

6. Folder domain untuk membuat Kontrak Untuk data User yang di ambil dari database

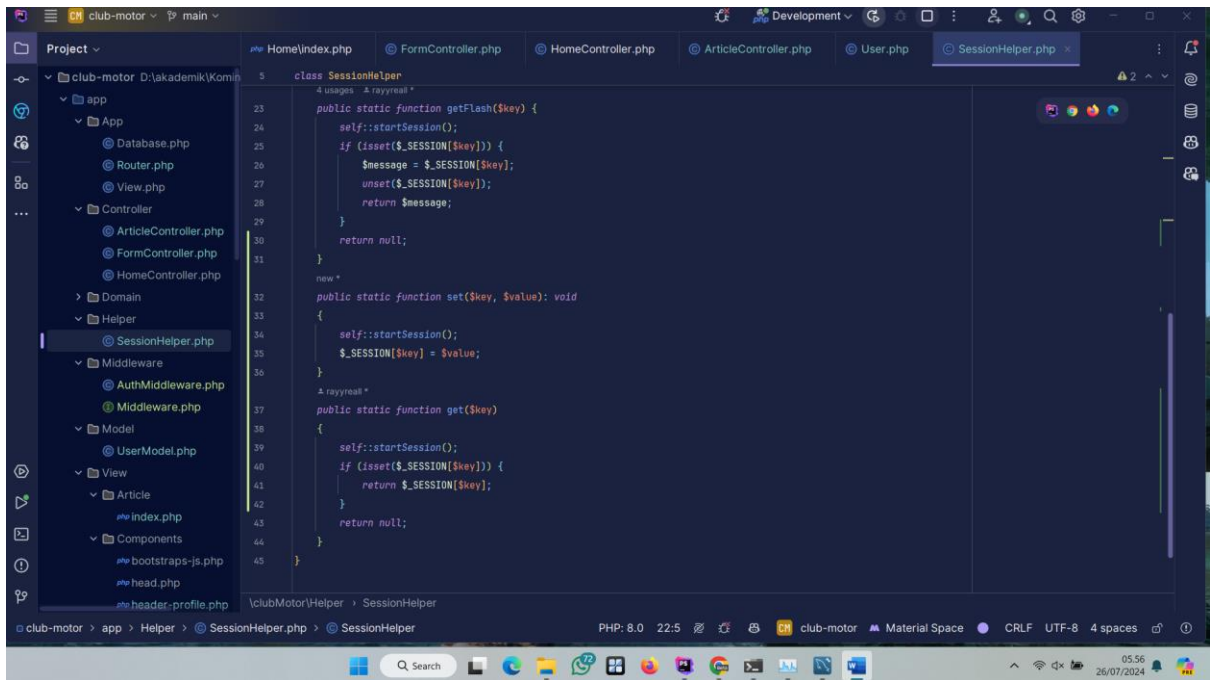


```
1 <?php
2
3 namespace clubMotor\Domain;
4
5 < rayreal >
6 class User
7 {
8     public int $id;
9     public string $username;
10    public string|null $password;
11    public string $email;
12
13    2 usages < rayreal >
14    public function getEmail(): string
15    {
16        return $this->email;
17    }
18
19    3 usages < rayreal >
20    public function setEmail(string $email): void
21    {
22        $this->email = $email;
23    }
24
25    public string $name;
26
27    < rayreal >
28    public function getId(): int
29    {
30    }
```

7. Class SessionHelper Untuk Membantu melakukan handle terhadap session



```
1 < rayreal >
2 class SessionHelper
3 {
4     10 usages < rayreal >
5     public static function startSession()
6     {
7         if (session_status() == PHP_SESSION_NONE) {
8             session_start();
9         }
10    }
11
12    1 usage < rayreal >
13    public static function destroySession()
14    {
15        session_destroy();
16    }
17
18    4 usages < rayreal >
19    public static function setFlash($key, $message): void
20    {
21        self::startSession();
22        $_SESSION[$key] = $message;
23    }
24
25    4 usages < rayreal >
26    public static function getFlash($key) {
27        self::startSession();
28        if (isset($_SESSION[$key])) {
29            $message = $_SESSION[$key];
30            unset($_SESSION[$key]);
31            return $message;
32        }
33    }
```

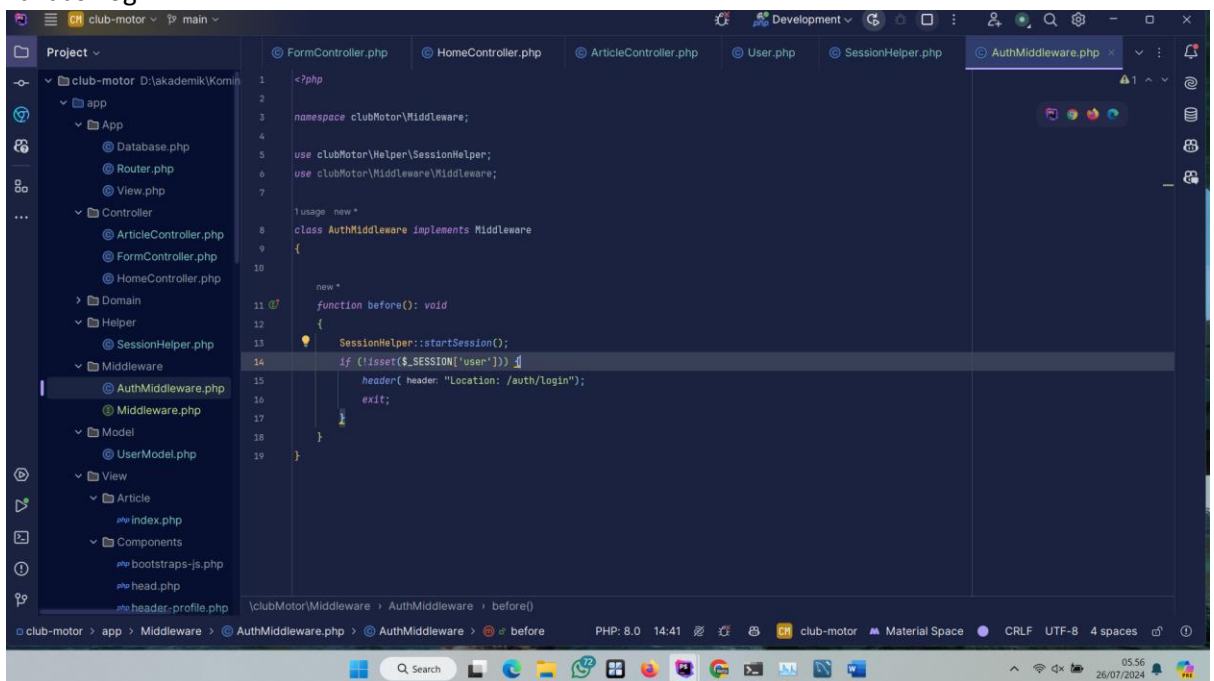


```
class SessionHelper
{
    public static function getFlash($key) {
        self::startSession();
        if (isset($_SESSION[$key])) {
            $message = $_SESSION[$key];
            unset($_SESSION[$key]);
            return $message;
        }
        return null;
    }

    public static function set($key, $value): void
    {
        self::startSession();
        $_SESSION[$key] = $value;
    }

    public static function get($key)
    {
        self::startSession();
        if (isset($_SESSION[$key])) {
            return $_SESSION[$key];
        }
        return null;
    }
}
```

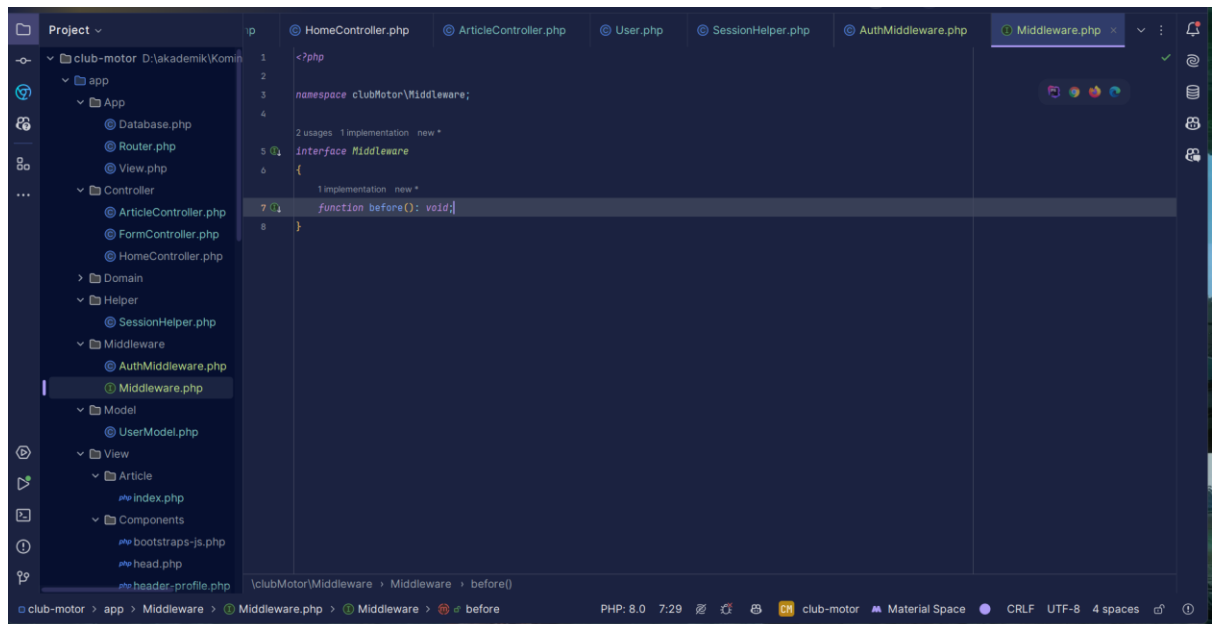
8. Middleware Auth, otomatis dijalankan berdasarkan pemanggilan routing disini melakukan validasi login



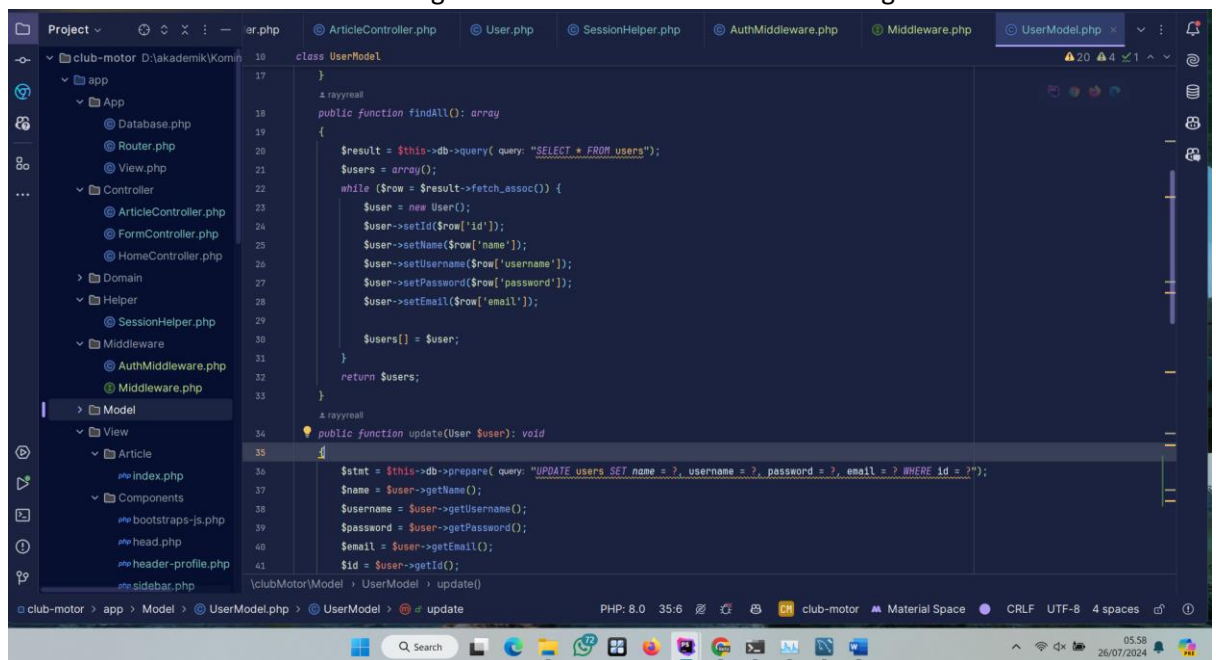
```
<?php
namespace cClubMotor\Middleware;

use cClubMotor\Helper\SessionHelper;
use cClubMotor\Middleware\Middleware;

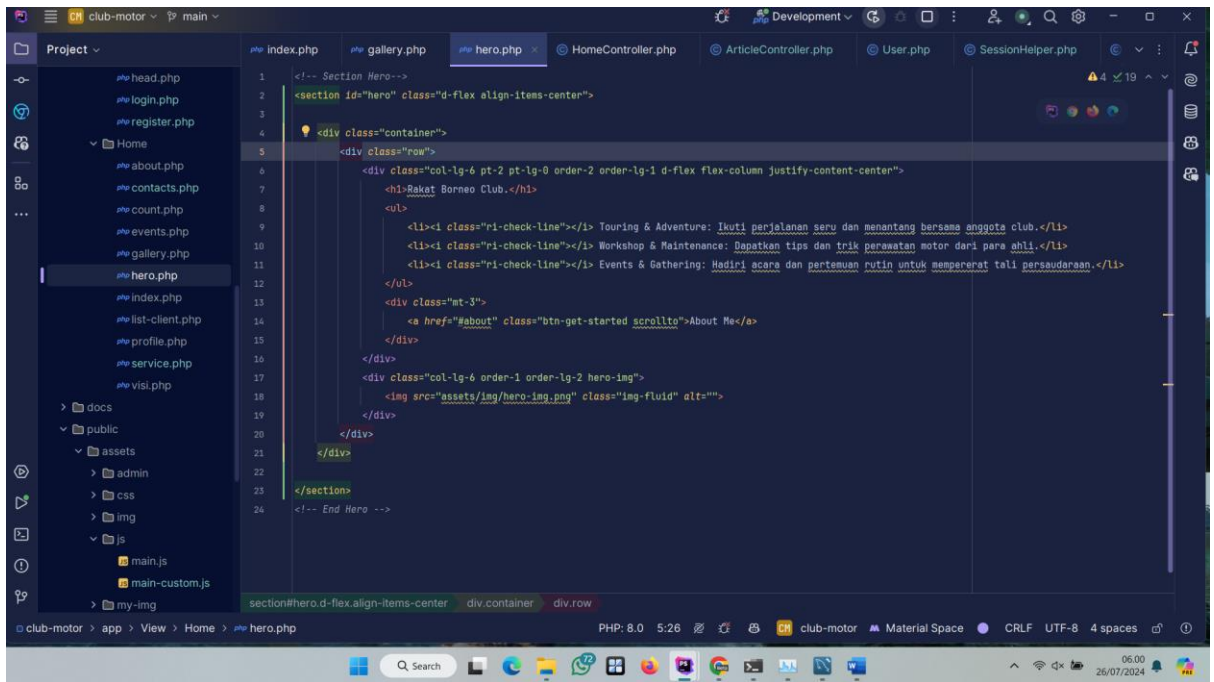
class AuthMiddleware implements Middleware
{
    function before(): void
    {
        SessionHelper::startSession();
        if (!isset($_SESSION['user'])) {
            header('Location: /auth/login');
            exit;
        }
    }
}
```



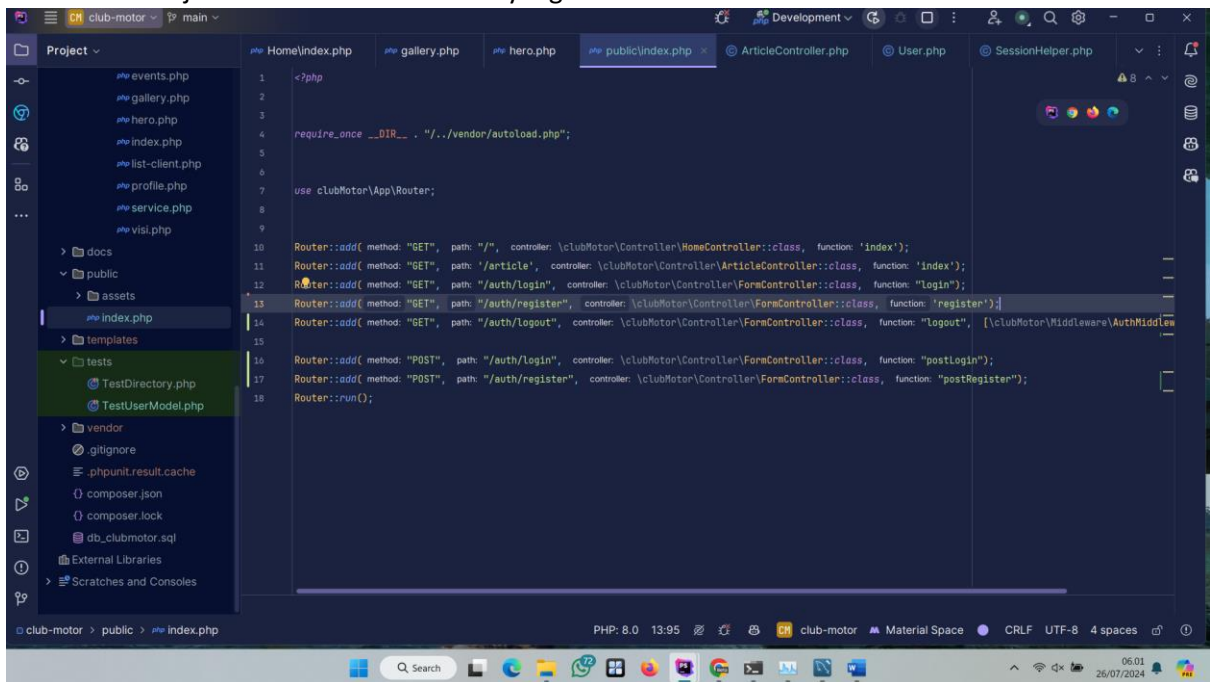
9. Folder model untuk Melakukan Pengambilan data dari database Untuk Login



10. View Untuk Tampilan Aplikasi berikut beberapa Example bagian Isi View



11. File untuk menjalankan keseluruhan code yang dibuat tadi



12. Struktur folder untuk menyimpan asset tambahan seperti gambar, css dll

