

LAPORAN AKHIR PBW A



4522210019 – Lutfiah Rizka Santoso

Dosen : Adi Wahyu Pribadi
S1-Teknik Informatika
Fakultas Teknik Universitas Pancasila
2024/2025

PRAKTIKUM 1

Menggunakan metode paket manajer system Ubuntu

1. Instal my SQL
2. Update system, tujuan nya memastikan semua paket system terbarukan sebelum instalasi

```
$ sudo apt update
```

3.menginstall web NGINX

4. Menginstal PHP dan modul untuk integrasi PHP dengan NGINX.

5. Menginstal sistem manajemen basis data MariaDB.

6. Mengonfigurasi NGINX agar bisa memproses file PHP.

Edit konfigurasi NGINX nya itu dengan : *sudo nano /etc/nginx/sites-available/default*
Lalu temukan bagian nya yg ini : *location ~ \.php\$ { include snippets/fastcgi-php.conf; fastcgi_pass unix:/var/run/php/php7.4-fpm.sock; fastcgi_param SCRIPT_FILENAME \$document_root\$fastcgi_script_name; include fastcgi_params; }* kemudian simpan dan tutup file

7. Mnerapkan konfigurasi NGINX dengan perintah : *sudo systemctl restart nginx*

8. kemudian bias langsung mengakses *http://localhost/info.php*

Menggunakan metode Docker

1.install docker untuk bias menjalankan container

2. buat file docker-compose.yml gunanya mengkonfigurasi layanan web stack compose

Dengan perintah :

mkdir my-webstack

cd my-webstack

nano docker-compose.yml

3.menyediakan file info.php menampilkan informasi PHP

4. jalankan *docker-compose up -d*

5.lalu buka *http://localhost/info.php*

Menggunakan Alat Otomatisasi konfigurasi (Ansible)

1.Install Anisble untuk otomatisasi konfigurasi.

2. membuat file playbook Ansible untuk menulis skrip mengatur web stack

Perintah : *mkdir ansible-webstack*

cd ansible-webstack

nano webstack.yml

3.Jalankan ansible-playbook webstack.yml

4.Akses *phpinfo()* di Browser

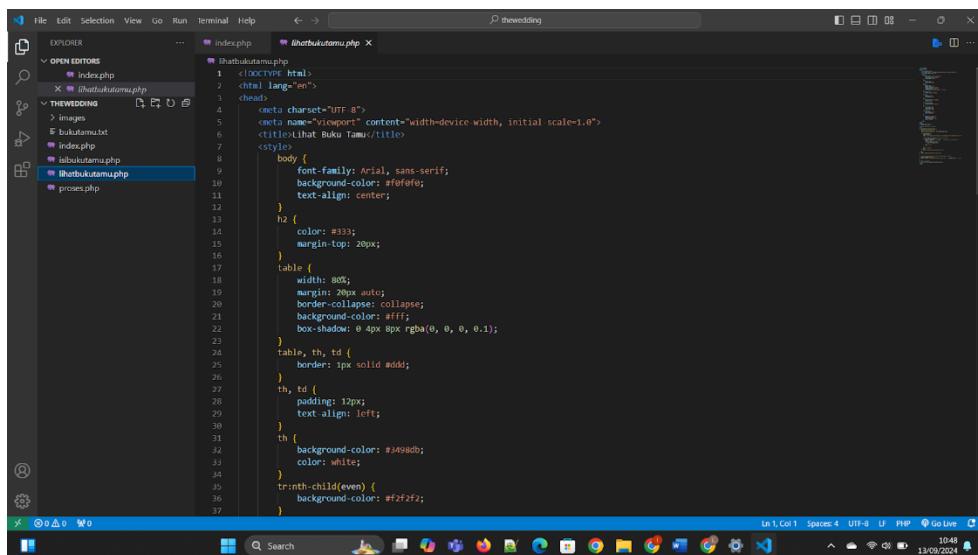
Tujuan praktikum : memberikan pemahaman dan keterampilan dalam menginstal dan mengkonfigurasi web stack yang umum digunakan dalam pengembangan web.

Teknologi yang Digunakan

1. NGINX Mengelola permintaan HTTP dari pengguna dan menyajikan file statis atau memproses permintaan dinamis melalui PHP.
2. PHP untuk Menangani logika aplikasi web, seperti pemrosesan formulir, interaksi dengan basis data, dan menghasilkan konten dinamis untuk pengguna.
3. MySQL/MariaDB:
Fungsi: Menyimpan dan mengelola data aplikasi web, seperti informasi pengguna, konten, dan konfigurasi.
4. Docker:
Fungsi: Mempermudah pengaturan dan pengelolaan lingkungan pengembangan dan produksi dengan mengisolasi aplikasi dalam kontainer yang dapat dipindahkan.
5. Ansible:
Fungsi: Mengotomatiskan proses instalasi dan konfigurasi perangkat lunak, mengurangi kebutuhan untuk konfigurasi manual dan meningkatkan konsistensi.

PRAKTIKUM 2

lihatbukutamu.php



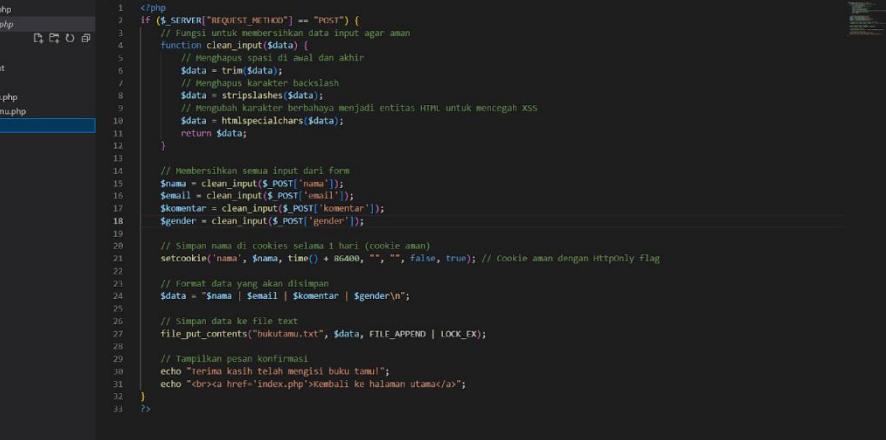
```

<?php
if (!file_exists("bukutamu.txt")) {
    echo "Belum ada data buku tamu.";
} else {
    $file = fopen("bukutamu.txt", "r");
    $data = fread($file, filesize("bukutamu.txt"));
    fclose($file);
    echo $data;
}
    
```

PHP untuk Menampilkan Data Buku Tamu:

- 1.Pemeriksaan File Kode PHP pertama mengecek apakah file bukutamu.txt ada menggunakan file_exists().
 - 2.Pembacaan File Jika file ada, file tersebut dibaca dengan file() dan setiap baris dipisahkan menggunakan karakter " | ".
 - 3.Tabel Dinamis PHP membuat tabel HTML secara dinamis berdasarkan data dari bukutamu.txt. Jika file kosong, akan ditampilkan pesan "Belum ada data buku tamu."
 - 4.Pengunjung Terakhir Jika ada cookie bernama 'nama', akan ditampilkan nama pengunjung terakhir di bawah tabel menggunakan \$_COOKIE['nama'].
- Navigasi: link ([<a>](#)) untuk kembali ke halaman utama (index.php).

proses.php



```
File Edit Selection View Go Run Terminal Help < - > thewedding

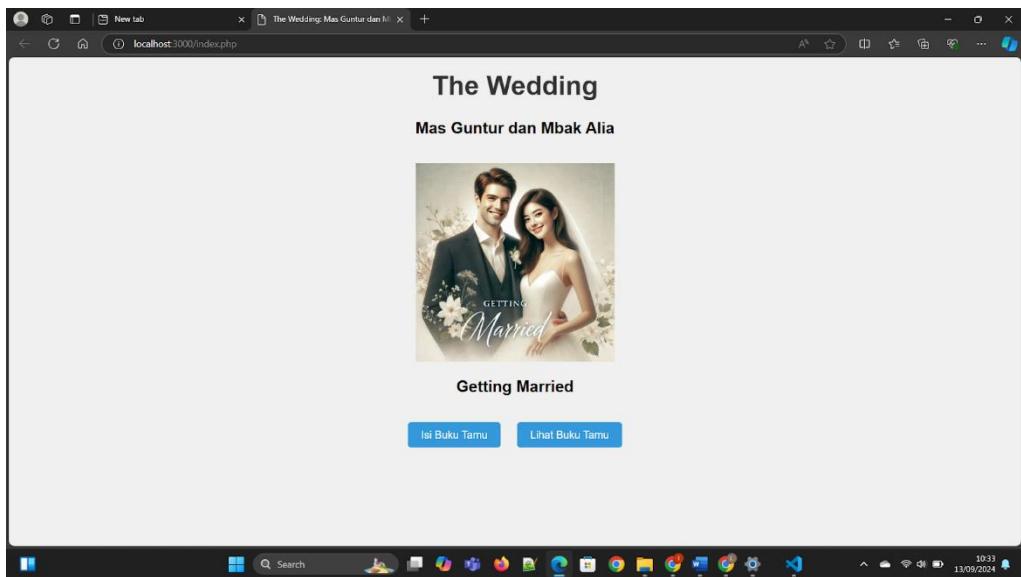
EXPLORER ... index.php proses.php x
OPEN EDITORS ... index.php proses.php x
THEWEDDING ... bukutamu.txt index.php isbukutamu.php libisbukutamu.php proses.php
index.php
proses.php
1 <?php
2 if ($_SERVER["REQUEST_METHOD"] == "POST") {
3     // Fungsi untuk memerlukan data input agar aman
4     function clean_input($data) {
5         // Menghapus spasi di awal dan akhir
6         $data = trim($data);
7         // Menghapus karakter backslash
8         $data = stripslashes($data);
9         // Mengubah karakter berbahaya menjadi entitas HTML untuk mencegah XSS
10        $data = htmlspecialchars($data);
11        return $data;
12    }
13
14    // Membersihkan semua input dari form
15    $nama = clean_input($_POST['nam']);
16    $email = clean_input($_POST['email']);
17    $komentar = clean_input($_POST['komentar']);
18    $gender = clean_input($_POST['gender']);
19
20    // Simpan nama di cookies selama 1 hari (cookie aman)
21    setcookie('nam', $nama, time() + 86400, "", "", false, true); // cookie aman dengan HttpOnly flag
22
23    // Format data yang akan disimpan
24    $data = $nama | $email | $komentar | $gender\n";
25
26    // Simpan data ke file text
27    file_put_contents("bukutamu.txt", $data, FILE_APPEND | LOCK_EX);
28
29    // Tampilkan pesan konfirmasi
30    echo "terima kasih telah mengisi buku tamu!";
31    echo "<br><a href='index.php'>Kembali ke halaman utama</a>";
32 }
33 ?>
```

Kode PHP di atas berfungsi untuk memproses data yang dikirim melalui form "Isi Buku Tamu".

1. Pengecekan Metode Request if (`$_SERVER["REQUEST_METHOD"] == "POST"`) mengecek apakah metode pengiriman data adalah POST. Ini memastikan bahwa script hanya berjalan jika form dikirimkan.
 2. Fungsi `clean_input()`
 - Fungsi ini digunakan untuk membersihkan data input yang diterima dari form agar lebih aman dari serangan seperti XSS (Cross-Site Scripting) dan memastikan data bebas dari karakter tak diinginkan.
 - `trim()` menghapus spasi di awal dan akhir string.
 - `stripslashes()` menghapus backslash.
 - `htmlspecialchars()` mengubah karakter khusus menjadi entitas HTML (misalnya < menjadi <) untuk mencegah eksekusi kode berbahaya.
 3. Membersihkan Data Input Setiap input dari form seperti nama, email, komentar, dan gender dibersihkan dengan memanggil fungsi `clean_input()`.
 4. Penyimpanan Nama ke dalam Cookie `setcookie('nama', $nama, time() + 86400, "", "", false, true);` menyimpan nama yang diisi oleh pengguna ke dalam cookie. Cookie diset agar berlaku selama 1 hari (86400 detik). Opsi `HttpOnly` (true di parameter terakhir) digunakan agar cookie hanya dapat diakses oleh server, meningkatkan keamanan.
 5. Format Data yang Akan Disimpan Data dari form disusun dalam format teks: `"$nama | $email | $komentar | $gender\n"`, di mana setiap field dipisahkan dengan simbol | .
 6. Menyimpan Data ke File `file_put_contents("bukutamu.txt", $data, FILE_APPEND | LOCK_EX);` menyimpan data yang sudah diformat ke file bukutamu.txt. Opsi `FILE_APPEND` memastikan bahwa data ditambahkan ke akhir file tanpa menghapus data yang sudah ada, sedangkan `LOCK_EX` memastikan bahwa file tidak diakses oleh proses lain saat sedang ditulis untuk menghindari masalah konkuren.
 7. Pesan Konfirmasi ini Jika data berhasil diproses, pesan "Terima kasih telah mengisi buku tamu!" ditampilkan. Terdapat juga link untuk kembali ke halaman utama (index.php).

I. Hasil screenshot tampilan landing page, form input, dan halaman daftar buku tamu.

Landing page



Form Input

A screenshot of a Microsoft Edge browser window showing a guest book form. The title bar says 'Isi Buku Tamu' and the address bar shows 'localhost:3000/isibukutamu.php'. The form is titled 'Isi Buku Tamu' and contains fields for 'Nama:' (Name), 'Email:', and 'Komentar:' (Comment). There is also a 'Gender:' section with radio buttons for 'Pria' (Male) and 'Wanita' (Female). A 'Submit' button is at the bottom right. Below the form is a link 'Kembali ke halaman utama' (Return to main page). The taskbar at the bottom shows various pinned icons.

halaman daftar buku tamu.

Daftar Buku Tamu			
Nama	Email	Komentar	Gender
Adi Wahyu	adi.wahyu.p@univpancasila.ac.id	Selamat menempuh hidup baru!	Pria
Sarah	si@sarah.only	Selamat mas Guntur dan mbak Alia! Happy Wedding!	Wanita

[Kembali ke halaman utama](#)

PRAKTIKUM 3

Index.php

```

<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Landing Page - Buku Tamu</title>
</head>
<body>
    <h1>Selamat datang di Buku Tamu</h1>
    <p>Selamat datang di situs buku tamu kami. Silakan mengisi buku tamu atau melihat daftar tamu yang telah mengisi.</p>
    <ul>
        <li><a href="input.php">isi Buku Tamu</a></li>
        <li><a href="tabel.php">lihat Daftar Tamu</a></li>
    </ul>
</body>
</html>

```

Membuat halaman landing sederhana untuk buku tamu. Terdapat judul, paragraf sambutan, dan dua tautan: satu untuk mengisi buku tamu (input.php) dan satu lagi untuk melihat daftar tamu (tabel.php).

Input.php

```

<?php
// Mengimpor file koneksi
include 'koneksi.php';

// Proses penyimpanan data jika formulir disubmit
if (isset($_POST['submit'])) {
    $nama = $_POST['nama'];
    $email = $_POST['email'];
    $komentar = $_POST['komentar'];

    // SQL untuk insert data ke tabel tamu
    $sql = "INSERT INTO tamu (nama, email, komentar) VALUES ('$nama', '$email', '$komentar')";

    // Eksekusi query dan cek keberhasilannya
    if ($conn->query($sql) === TRUE) {
        echo "<p>Terima kasih, data Anda telah disimpan!</p>";
    } else {
        echo "<p>Error: " . $sql . "<br>" . $conn->error . "</p>";
    }
}

<!DOCTYPE html>
<html>

```

Kode PHP ini menginput data buku tamu. Ketika formulir disubmit, data nama, email, dan komentar dikirimkan dan disimpan ke dalam tabel tamu di database melalui query SQL. Jika berhasil, muncul pesan konfirmasi; jika gagal, ditampilkan pesan error. Halaman HTML berisi formulir untuk mengisi data tamu dan tautan kembali ke halaman utama.

table.php

```

<?php
// Mengimpor file koneksi
include 'koneksi.php';
?>
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Daftar Buku Tamu</title>
</head>
<body>
    <h1>Daftar Buku Tamu</h1>
    <table border="1">
        <tr>
            <th>Nama</th>
            <th>Email</th>
            <th>Komentar</th>
            <th>Tanggal</th>
        </tr>
        <tr>
            <td colspan="4" style="text-align: center;">Belum ada data tamu.
        </tr>
    </table>
</body>

```

The screenshot shows the Visual Studio Code interface with the 'table.php' file open in the editor. The code defines an HTML table with four columns: Nama, Email, Komentar, and Tanggal. It includes a placeholder row for when there is no data. The output tab shows logs from a PHP server, indicating connections and acceptances. The taskbar at the bottom shows various application icons.

Menampilkan daftar tamu yang diambil dari database. Setelah mengimpor file koneksi (koneksi.php), query SQL digunakan untuk mengambil nama, email, komentar, dan tanggal dari tabel tamu. Data ditampilkan dalam tabel HTML, dan jika tidak ada tamu, ditampilkan pesan "Belum ada data tamu." Koneksi database ditutup setelah data ditampilkan.

koneksi.php

```

<?php
// File: koneksi.php
$servername = "localhost";
$username = "root";
$password = "";
$dbname = "buku_tamu_db";

// Membuat koneksi ke MySQL
$conn = new mysqli($servername, $username, $password, $dbname);

// Cek koneksi
if ($conn->connect_error) {
    die("Koneksi gagal: " . $conn->connect_error);
}
?>

```

The screenshot shows the Visual Studio Code interface with the 'koneksi.php' file open in the editor. The code creates a MySQL connection using the mysqli extension. It checks for errors and exits if they occur. The output tab shows logs from a PHP server, indicating connections and acceptances. The taskbar at the bottom shows various application icons.

Membuat koneksi ke database MySQL. Variabel berisi informasi server, username, password, dan nama database. Koneksi dibuat menggunakan objek mysqli, dan jika terjadi kesalahan, pesan error ditampilkan dengan die().

Berhasil menginput buku tamu

The screenshot shows the MySQL Workbench interface. A table named 'tamu' is displayed with one row of data:

	id	name	email	komentar	tanggal
	1	nzka	452211019@univpancasila.ac.id	nguenggg	2024-09-20 09:35:03

PRAKTIKUM 4

1. Database di MySQL

The screenshot shows the phpMyAdmin interface for the 'buku_tamu' database. It displays the structure of the database, including tables like 'admins', 'guests', and 'tables_sum'. A table named 'guests' is selected, showing its data.

2. Struktur Direktori Proyek

admin.php

Edit.php

```

File Edit Selection View Go Run Terminal Help ↻ → ⌂ bukutamu Database.php U guest.php U style.css U admin.php U admin_dashboard.php U admin_login.php U guest_from.php U guest_list.php U
OPEN EDITORS
  guestController.php con... U
  Database.php config U
  Guest.php models U
  style.css U
  admin.php U
  admin_dashboard.php v... U
  admin_login.php v... U
  guest_from.php v... U
  guest_list.php v... U
BUKUTAMU
  config Database.php
  controllers AdminController.php GuestController.php
  css # style.css
  models Admin.php Guest.php
  views admin_dashboard.php admin_login.php guest_from.php guest_list.php
  Admin.php
  Guest.php
  index.php
  login.php
  Logout.php
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
[Fri Sep 27 11:36:45 2024] [:1]:58488 Closing
[Fri Sep 27 11:36:45 2024] [:1]:58488 Accepted
[Fri Sep 27 11:37:55 2024] [:1]:58488 Closed without sending a request; it was probably just an unused speculative preconnection
[Fri Sep 27 11:37:55 2024] [:1]:58488 Closing
[Fri Sep 27 11:44:05 2024] [:1]:63451 Accepted
[Fri Sep 27 11:46:00 2024] [:1]:63451 Closed without sending a request; it was probably just an unused speculative preconnection
[Fri Sep 27 11:46:00 2024] [:1]:63451 Closing
PHP Server
Ln 1, Col 1 Spaces 4 UTF-8 CR/LF PHP Go Live

```

Edit.php

Index.php

```

File Edit Selection View Go Run Terminal Help ↻ → ⌂ bukutamu admin.php U admin_dashboard.php U admin_login.php U guest_from.php U guest_list.php U Admin.php U Edit.php U AdminController.php U index.php U
OPEN EDITORS
  Guest.php models U
  style.css U
  admin.php U
  admin_dashboard.php v... U
  admin_login.php v... U
  guest_from.php v... U
  guest_list.php v... U
  Admin.php
  Guest.php
  index.php
  login.php
  Logout.php
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
[Fri Sep 27 11:36:45 2024] [:1]:58488 Closing
[Fri Sep 27 11:36:45 2024] [:1]:58488 Accepted
[Fri Sep 27 11:37:55 2024] [:1]:58488 Closed without sending a request; it was probably just an unused speculative preconnection
PHP Server
Ln 1, Col 1 Spaces 4 UTF-8 CR/LF PHP Go Live

```

Index.php

Login.php

```

File Edit Selection View Go Run Terminal Help ↻ → ⌂ bukutamu admin_login.php U guest_from.php U guest_list.php U Admin.php U Edit.php U AdminController.php U index.php U
OPEN EDITORS
  admin.php U
  admin_dashboard.php v... U
  admin_login.php v... U
  guest_from.php v... U
  guest_list.php v... U
  Admin.php
  Guest.php
  index.php
  login.php
  Logout.php
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
[Fri Sep 27 11:36:45 2024] [:1]:58488 Closing
[Fri Sep 27 11:36:45 2024] [:1]:58488 Accepted
[Fri Sep 27 11:37:55 2024] [:1]:58488 Closed without sending a request; it was probably just an unused speculative preconnection
PHP Server
Ln 1, Col 1 Spaces 4 UTF-8 CR/LF PHP Go Live

```

Login.php

```

<?php
session_start();
require_once 'controllers/AdminController.php';

$admincontroller = new AdminController();

if ($_SERVER['REQUEST_METHOD'] == 'POST') {
    $username = $_POST['username'];
    $password = $_POST['password'];

    if ($admincontroller->login($username, $password)) {
        if (isset($_SESSION['admin']) == $username) {
            header("Location: admin.php");
        } else {
            $error = "username atau password salah!";
        }
    } else {
        include 'views/admin_login.php';
    }
}

if (isset($error)) {
    echo "<p style='color:red;'>$error</p>";
}
?

```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

[Fri Sep 27 11:36:45 2024] [::1]:58488 Accepted

[Fri Sep 27 11:37:55 2024] [::1]:58488 Closed without sending a request; it was probably just an unused speculative preconnection

PHP Server

Logout.php

```

<?php
session_unset();
session_destroy();
header("Location: login.php");
?

```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

[Fri Sep 27 11:36:45 2024] [::1]:58487 [200]: GET /index.php - Uncaught Error: Call to a member function prepare() on null in C:\xampp\htdocs\Prak-PBW\A\bukutamu\models\Guest.php:3

Stack traces:

#0 C:\xampp\htdocs\Prak-PBW\A\bukutamu\controllers\GuestController.php(23): Guest->readAll()

#1 C:\xampp\htdocs\Prak-PBW\A\bukutamu\models\Guest.php(10): GuestController->readAll()

#2 {main}

thrown in C:\xampp\htdocs\Prak-PBW\A\bukutamu\models\Guest.php on line 37

[Fri Sep 27 11:36:45 2024] [::1]:58487 Closing

[Fri Sep 27 11:37:55 2024] [::1]:58488 Closing

[Fri Sep 27 11:37:55 2024] [::1]:58488 Accepted

[Fri Sep 27 11:46:00 2024] [::1]:63451 Accepted

[Fri Sep 27 11:46:00 2024] [::1]:63451 Closed without sending a request; it was probably just an unused speculative preconnection

[Fri Sep 27 11:46:00 2024] [::1]:63451 Closed without sending a request; it was probably just an unused speculative preconnection

[Fri Sep 27 11:46:00 2024] [::1]:63451 Closing

PHP Server

SS Hasil Bukutamu

Selamat Datang di Buku Tamu

Komentar

Nama:

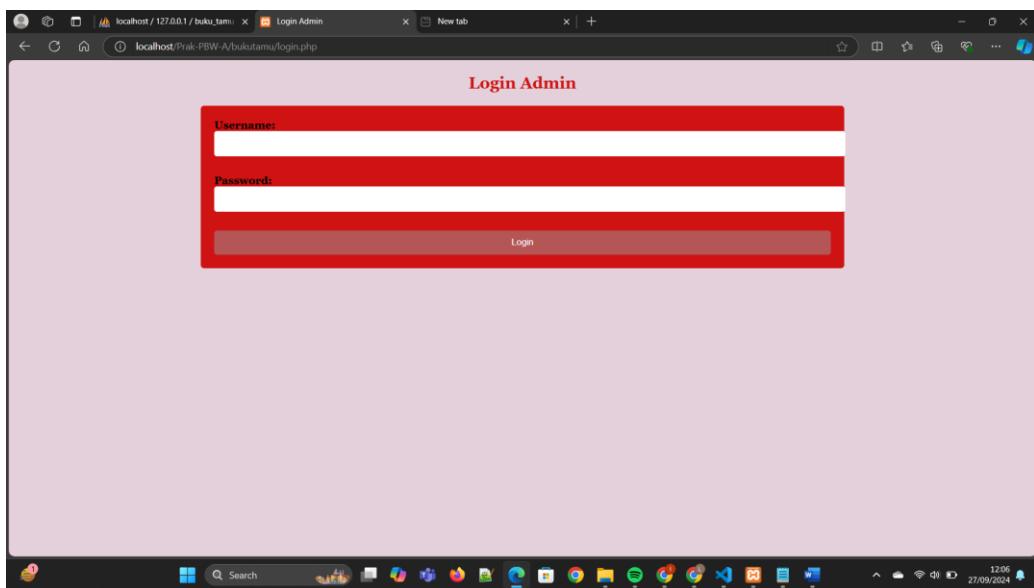
Komentar:

Kirim

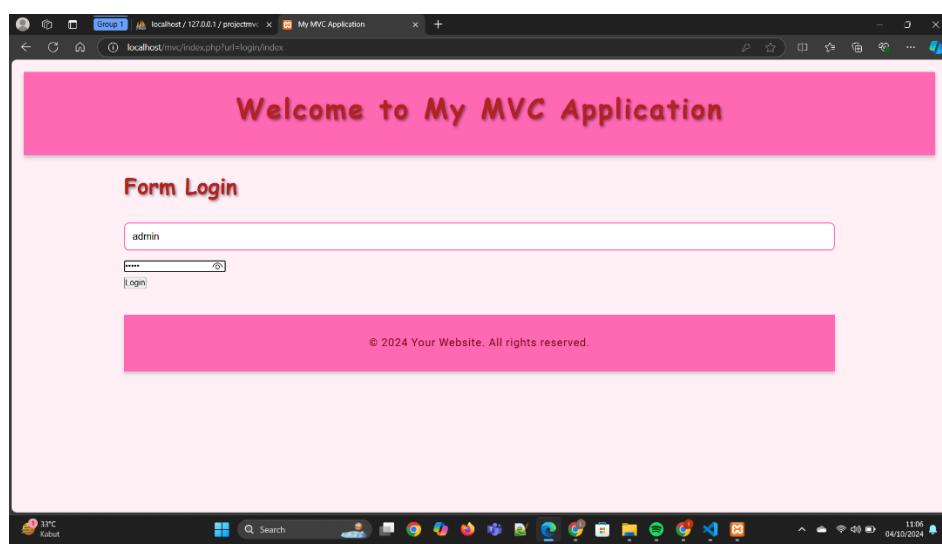
Daftar Komentar

Nama	Komentar	Waktu
rizka	semangatt	2024-09-27 12:05:17

SS Halaman Login.php



PRAKTIKUM 5



Welcome to My MVC Application

Halaman Admin

Tambah Barang | Logout

ID	NAMA	STOK	HARGA	AKSI
2	Permen karet	5	12000.00	Edit Hapus

© 2024 Your Website. All rights reserved.

Welcome to My MVC Application

Tambah Barang

Nama:

Stok:

Harga:

© 2024 Your Website. All rights reserved.

Welcome to My MVC Application

Edit Barang

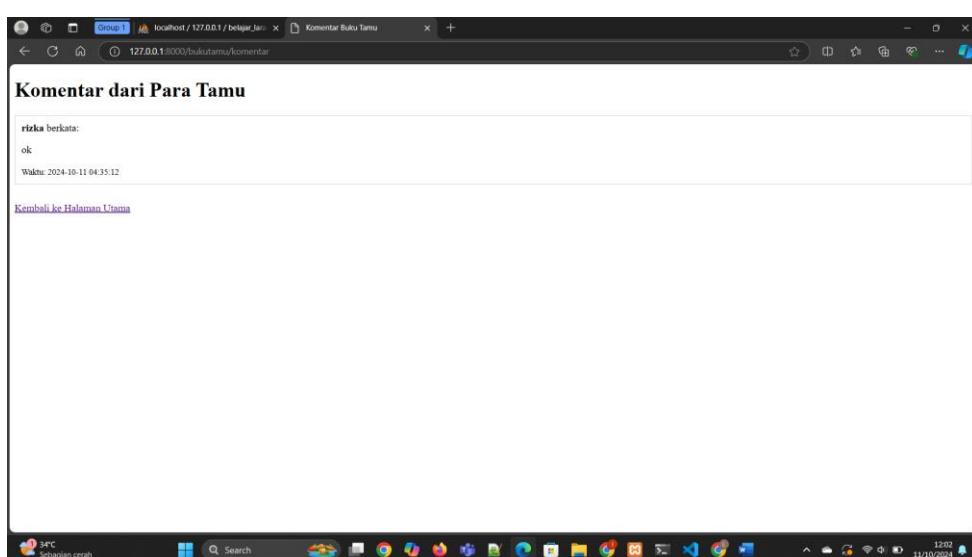
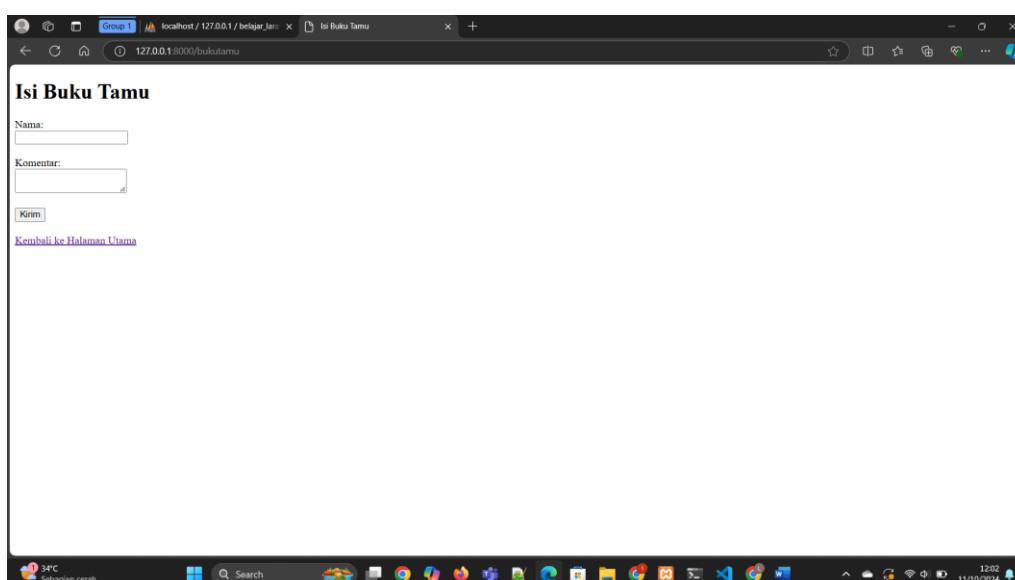
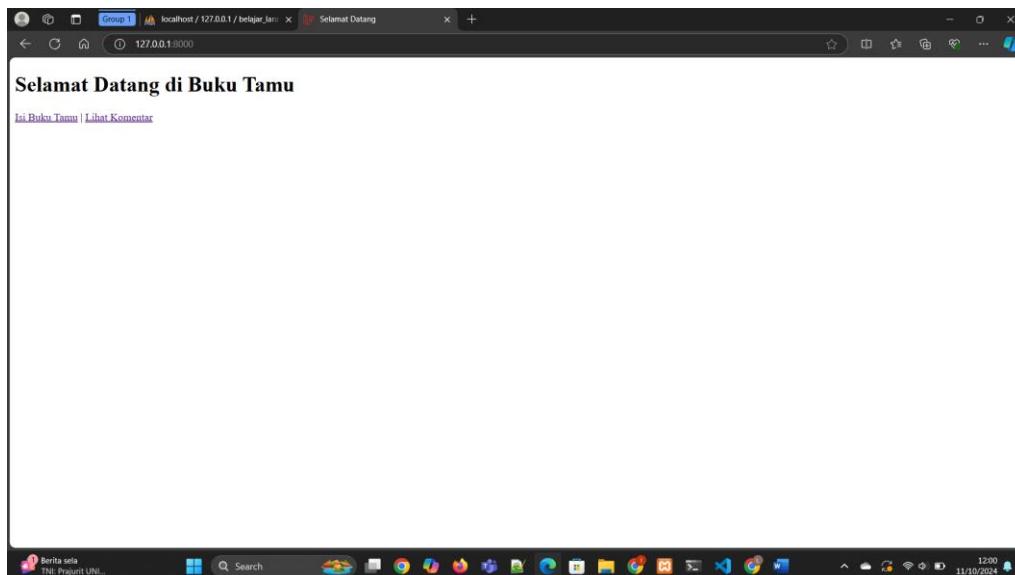
Nama:

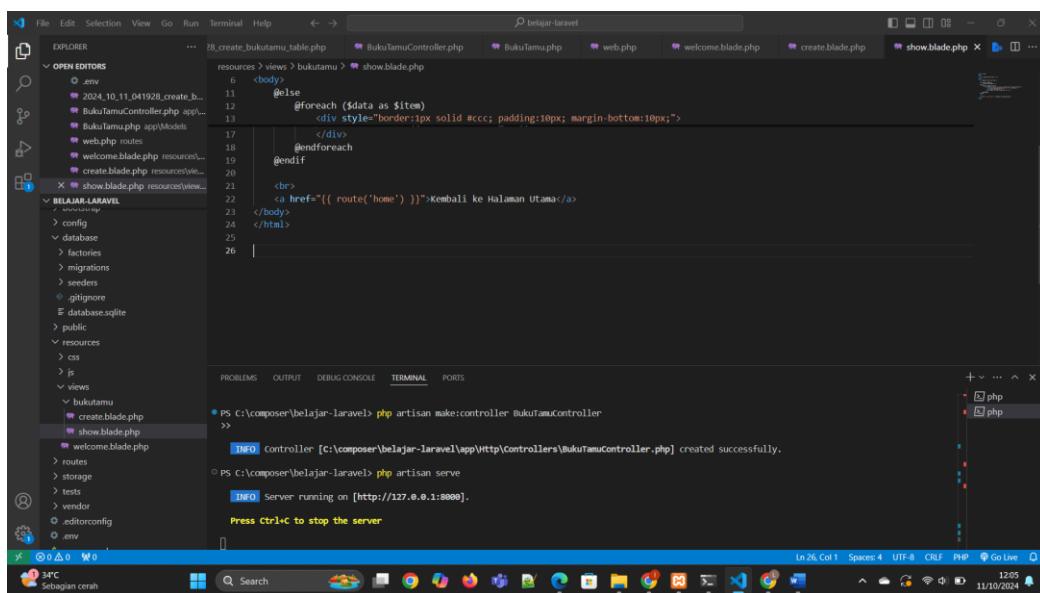
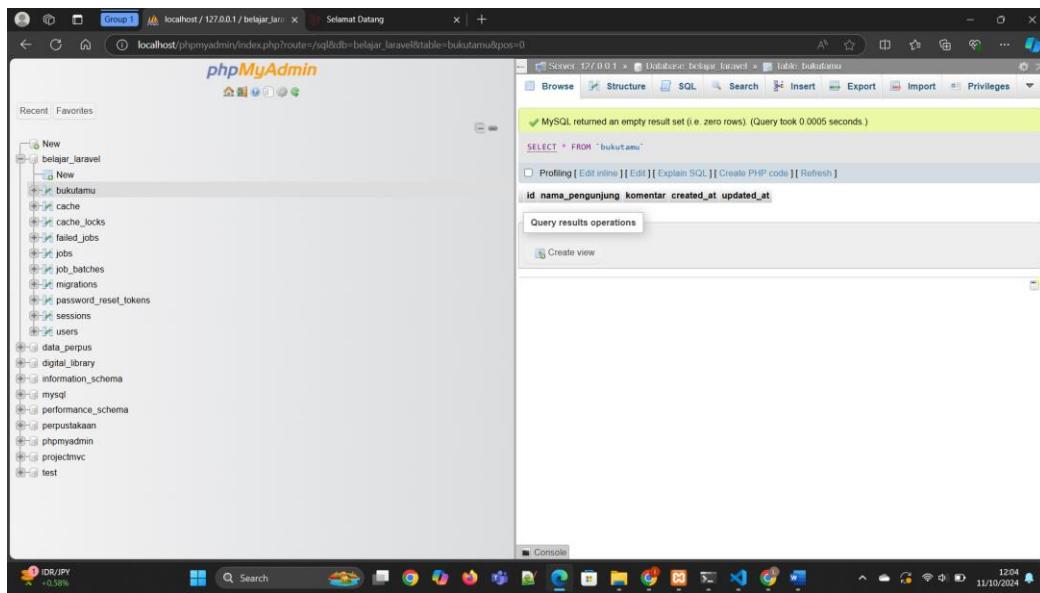
Stok:

Harga:

© 2024 Your Website. All rights reserved.

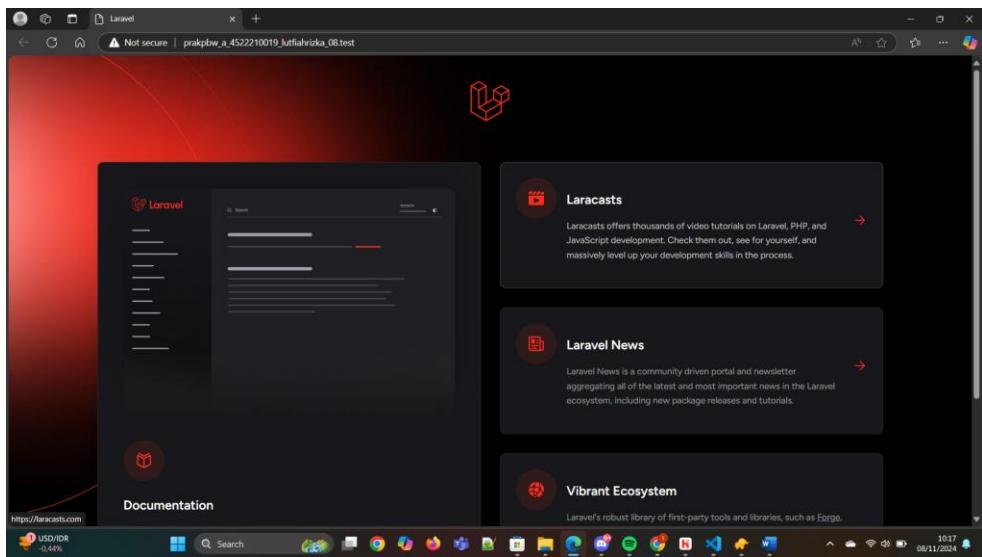
PRAKTIKUM 6



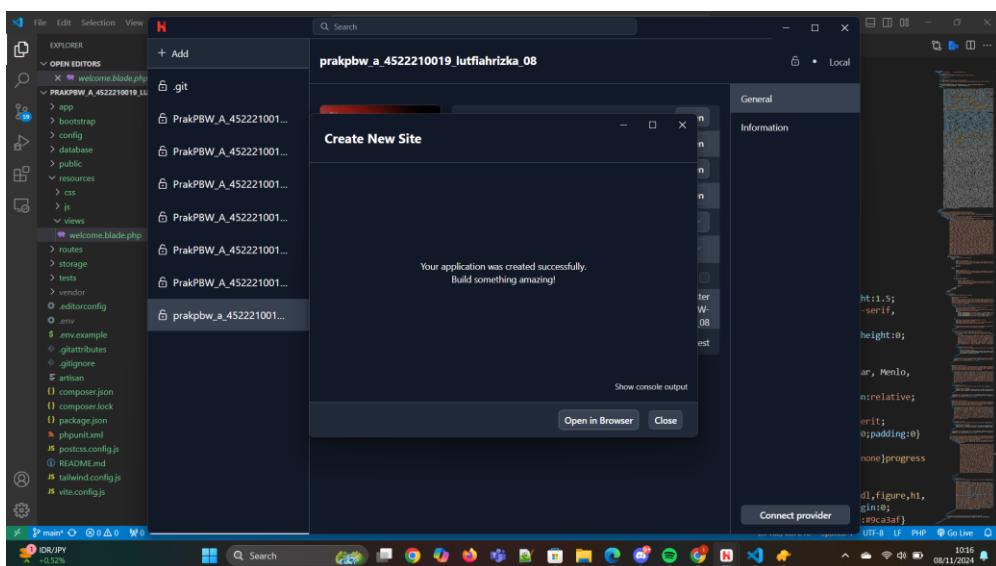


PRAKTIKUM 7

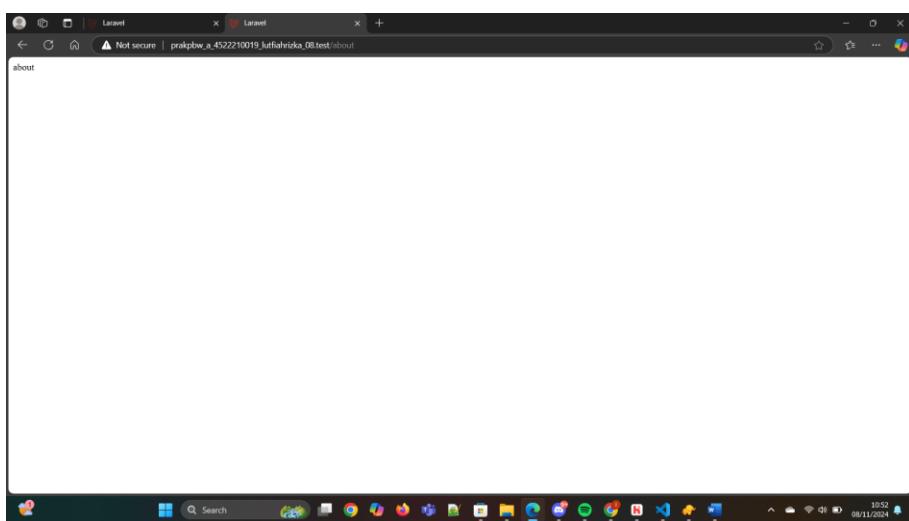
- Bagaimana tampilan utama?



- **Sukses Herd**



- **Blade About**



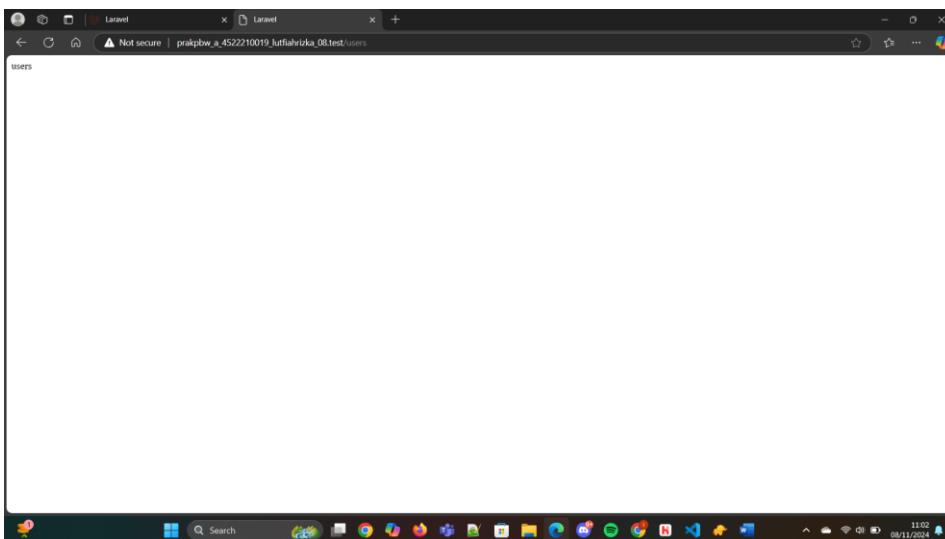
- Tampilan users.index

```

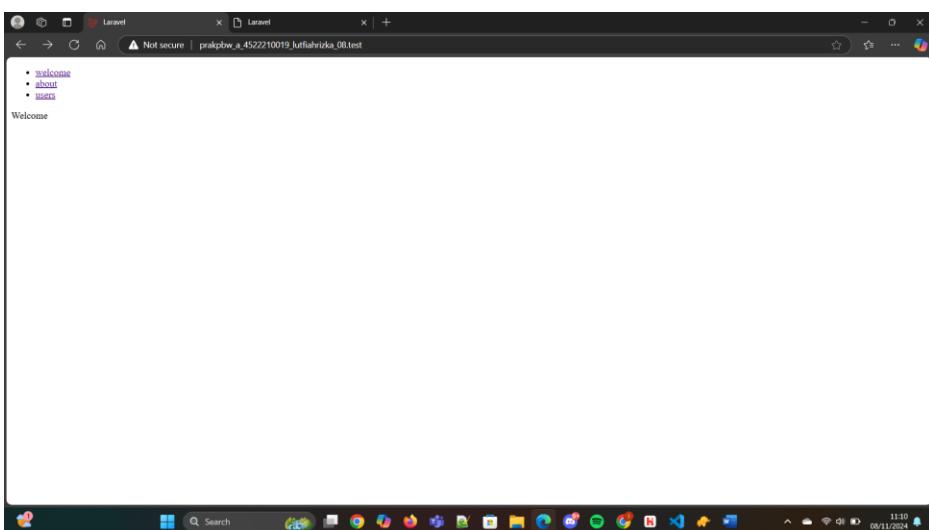
File Edit Selection View Go Run Terminal Help < - > prakpbw_a_4522210019_lutfiahrikza_08
EXPLORER routes > web.php
routes > web.php
1 <?php
2
3 use Illuminate\Support\Facades\Route;
4
5 Route::get('/', function () {
6     return view('welcome');
7 });
8
9 Route::get('/about', function () {
10    return view('about');
11 });
12
13 Route::get('users', function () {
14     $users = [
15         ['id' => 1, 'name' => 'John Doe'],
16     ];
17     return view('users.index', ['users' => $users]);
18 });
19

```

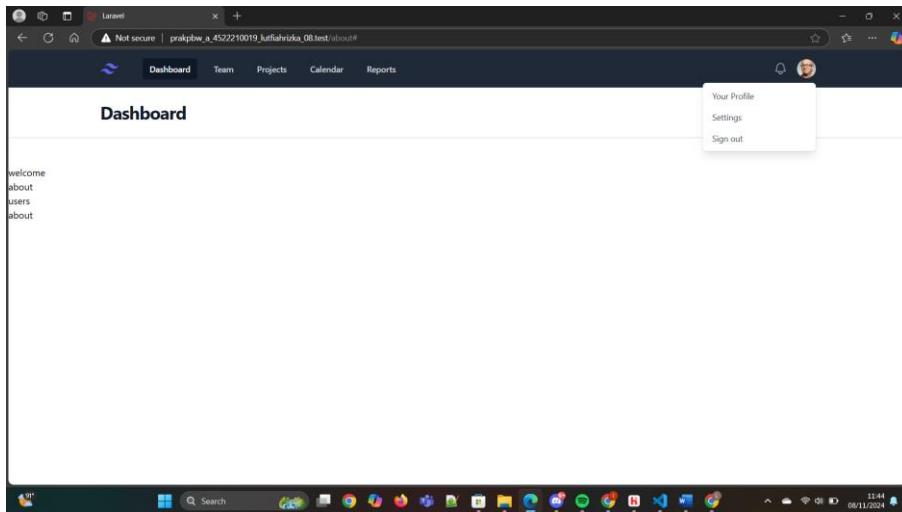
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL 08/11/2024



- Bagaimana tampilan HomePage



- Setelah Styling Tailwind CSS



- Pelajari penerapan otorisasi dengan Laravel Breeze! Dan buatkan rangkumannya.

Laravel Breeze adalah paket starter kit untuk Laravel yang menyediakan implementasi otentikasi dasar (login, register, reset password) dan juga otorisasi. Laravel Breeze menggunakan **Tailwind CSS** dan **Blade templates** atau **Inertia.js** (untuk Vue atau React) sebagai frontend.

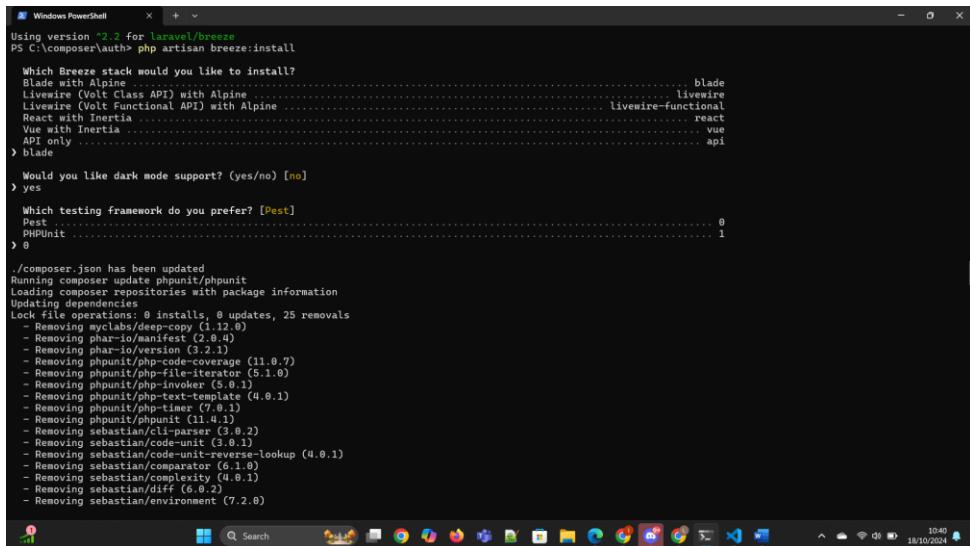
```
Windows PowerShell
PS C:\composer\auth> php artisan migrate
[INFO] Nothing to migrate.

PS C:\composer\auth> composer require laravel/breeze --dev
./composer.json has been updated
Loading composer repositories with package information
Updating composer repositories with package information
Updating dependencies
Lock file operations: 1 install, 0 updates, 0 removals
- Locking laravel/breeze (v2.2.2)
Writing lock file
Install dependencies from lock file (including require-dev)
Package operations: 1 install, 0 updates, 0 removals
INFO: Could not find files for the given pattern(s).
INFO: Could not find files for the given pattern(s).
INFO: Could not find files for the given pattern(s).
- Downloading laravel/breeze (v2.2.2)
  Installing laravel/breeze (v2.2.2): Extracting archive
Generating optimized autoload files
> illuminate\Foundation\ComposerScripts::postAutoloadDump
> php artisan package:discover --ansi
[INFO] Discovering packages.

laravel/breeze ..... DONE
laravel/pail ..... DONE
laravel/sail ..... DONE
laravel/tinker ..... DONE
nesbot/carbon ..... DONE
numenaduro/collision ..... DONE
numenaduro/termwind ..... DONE

78 packages you are using are looking for funding.
Use the 'composer fund' command to find out more!
> php artisan vendor:publish --tag=laravel-assets --ansi --force
[INFO] No publishable resources for tag [laravel-assets].
No security vulnerability advisories found.
Using version '^2.2' for laravel/breeze
PS C:\composer\auth> php artisan breeze:install
```

`composer require laravel/breeze --dev` digunakan untuk menginstal paket **Laravel Breeze** sebagai dependensi dalam mode pengembangan (development). Opsi `--dev` menandakan bahwa paket Breeze hanya akan digunakan selama pengembangan aplikasi dan tidak diperlukan di lingkungan produksi. Jadi, saat aplikasi dipindahkan ke server produksi, dependensi ini tidak akan disertakan, yang membantu menjaga ukuran aplikasi tetap kecil dan hanya memuat paket yang dibutuhkan untuk menjalankan aplikasi.



```
Windows PowerShell x + 
Using version ^2.2 for laravel/breeze
PS C:\composer\auth> php artisan breeze:install
Which Breeze stack would you like to install?
Blade with Alpine ..... blade
Livewire (Volt Class API) with Alpine ..... livewire
Livewire (Volt Functional API) with Alpine ..... livewire-functional
React with Inertia ..... react
Vue with Inertia ..... vue
API only ..... api
> blade
Would you like dark mode support? (yes/no) [no]
> yes
Which testing framework do you prefer? [Pest]
Pest ..... 0
PHPUnit ..... 1
> 0
./composer.json has been updated
Running composer update phpuunit/phpunit
Lock file operations: 0 installs, 0 updates, 25 removals
Updating dependencies
Lock file operations: 0 installs, 0 updates, 25 removals
- Removing myclabs/deep-copy (1.12.0)
- Removing phar-io/manifest (2.0.4)
- Removing phar-io/mimeservice (2.1.1)
- Removing phpuunit/php-code-coverage (11.0.7)
- Removing phpuunit/php-file-iterator (5.1.0)
- Removing phpuunit/php-invoker (5.0.1)
- Removing phpuunit/php-text-template (4.0.1)
- Removing phpuunit/php-timer (7.0.1)
- Removing phpuunit/php-token-stream (4.0.1)
- Removing sebastian/cli-parser (3.0.2)
- Removing sebastian/code-unit (3.0.1)
- Removing sebastian/code-unit-reverse-lookup (4.0.1)
- Removing sebastian/comparator (6.1.0)
- Removing sebastian/complexity (4.0.1)
- Removing sebastian/diff (6.0.2)
- Removing sebastian/environment (7.2.0)
```

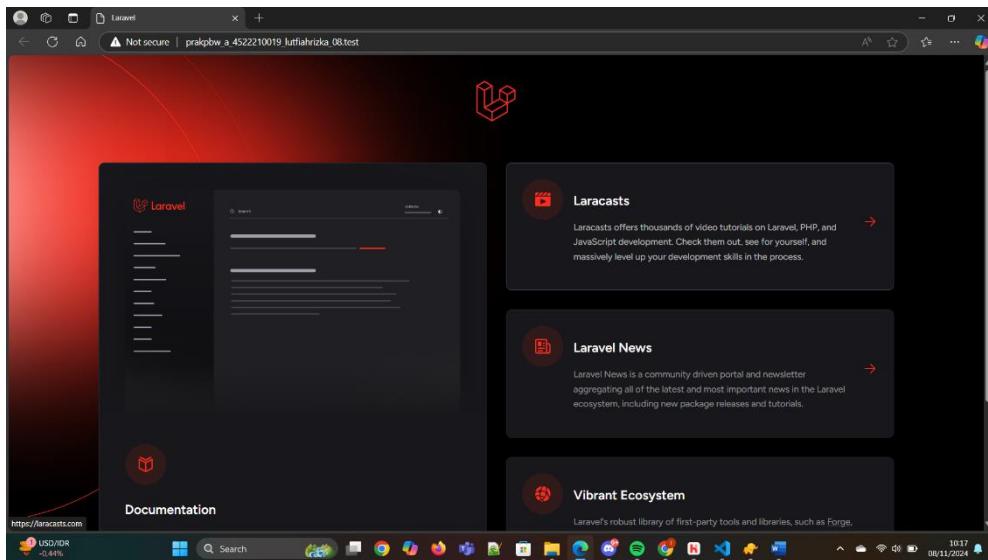
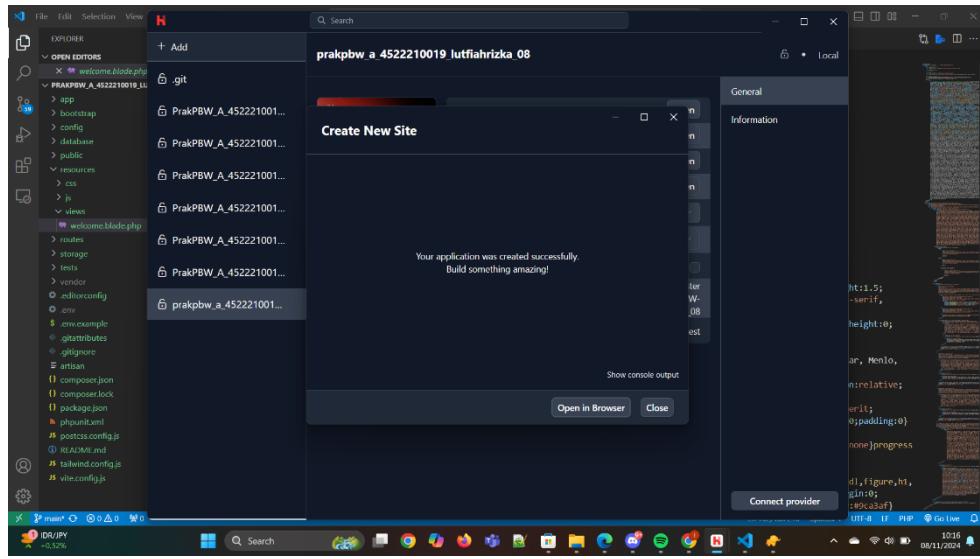
Perintah `php artisan breeze:install` digunakan untuk menginstal scaffolding atau kerangka dasar otentikasi yang disediakan oleh Laravel Breeze. Perintah ini akan membuat file dan struktur dasar untuk fitur-fitur otentikasi seperti login, registrasi, reset password, dan dashboard di aplikasi Laravel.

Proses instalasi ini secara otomatis menghasilkan beberapa file dan konfigurasi penting, seperti:

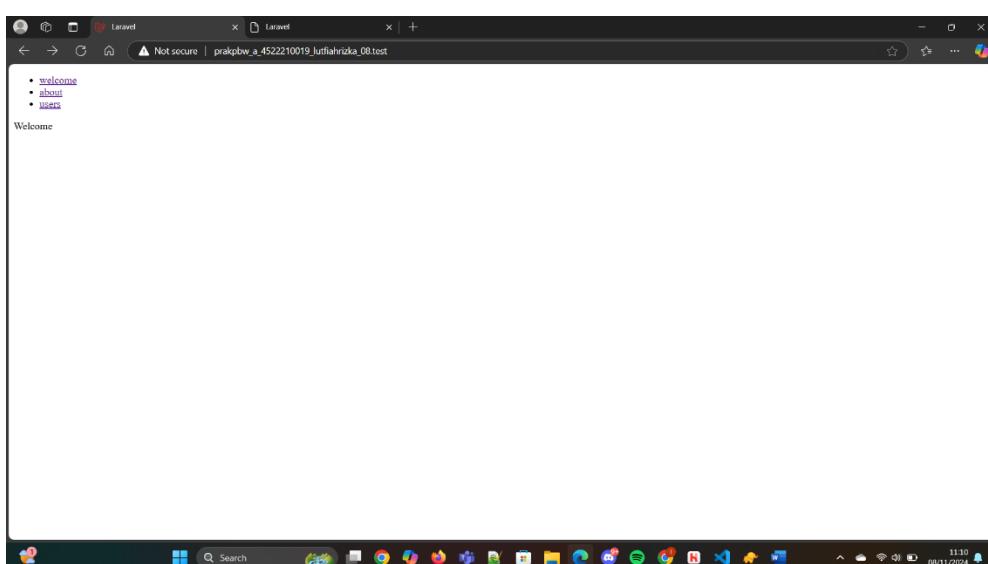
1. **Rute untuk otentikasi**, termasuk rute login, registrasi, dan reset password.
2. **Tampilan (views)** yang menggunakan Blade templates atau Inertia.js, tergantung pada konfigurasi yang dipilih, untuk menampilkan form login, register, dan halaman otentikasi lainnya.
3. **Controller yang menangani logika otentikasi**, seperti `AuthenticatedSessionController` untuk login, `RegisteredUserController` untuk registrasi, dan `PasswordResetController` untuk reset password.
4. **File validasi dan pengaturan form**, seperti form login, register, reset password, dan verifikasi email.
5. **Middleware otentikasi** yang akan membatasi akses ke rute tertentu untuk pengguna yang sudah terautentikasi.
 - a. Setelah menjalankan perintah ini, aplikasi Laravel Anda akan memiliki fitur otentikasi dasar yang siap digunakan, termasuk form login, registrasi, dan dashboard, lengkap dengan rute dan controller yang diperlukan untuk mengelola otentikasi pengguna.

PRAKTIKUM 8 VID 1-6

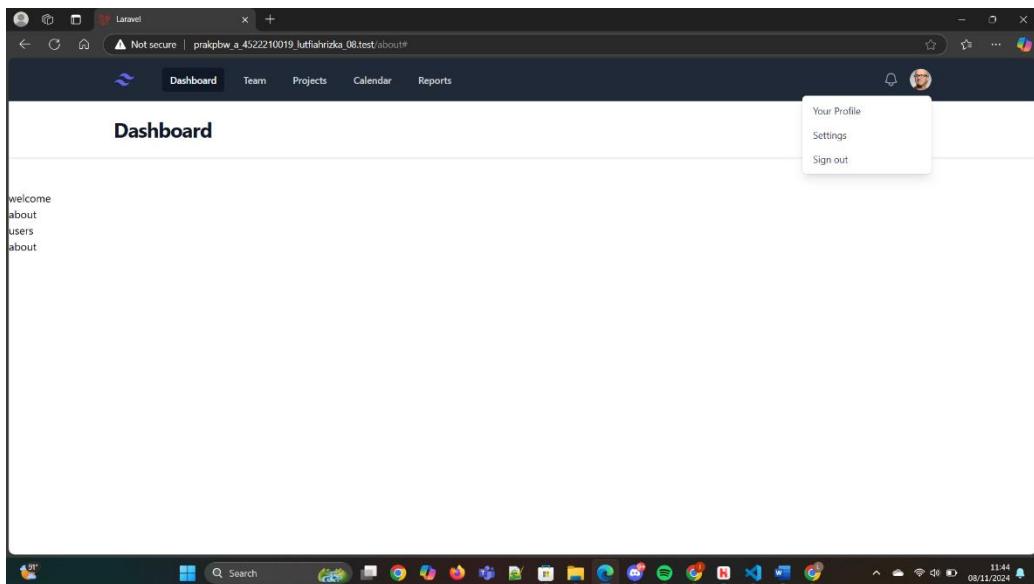
Mengcreate Package Laravel Menggunakan Herd



Tampilan HomePage



Styling CSS Tailwind



PRAKTIKUM 9 VID 7-12

Menampilkan Data dari Database

A screenshot of a web browser showing a JSON response from a database query. The URL in the address bar is "127.0.0.1:8000/users". The JSON data lists three user records with their respective details. The browser's address bar shows "Laravel Dev" and the system tray indicates the date as 15/11/2024 and the time as 11:39.

```
[{"id": 1, "name": "Rizka", "email": "lutfihrizka2@gmail.com", "created_at": "2024-11-15 11:04:26", "updated_at": "2024-11-15 11:04:26", "remember_token": "DEFAULT"}, {"id": 2, "name": "Salma", "email": "salma@mail@gmail.com", "created_at": "2024-11-15 11:07:41", "updated_at": "2024-11-15 11:07:41", "remember_token": "DEFAULT"}, {"id": 3, "name": "Irmawati", "email": "irmawati@gmail.com", "created_at": "2024-11-15 11:07:41", "updated_at": "2024-11-15 11:07:41", "remember_token": "DEFAULT"}]
```

A screenshot of a web browser window titled "Users" from a Laravel application. The URL is 127.0.0.1:8000/users. The page displays a table of users with columns: #, Name, Email, and Created At. There are three entries:

#	Name	Email	Created At
1	Salsa	salsabila@gmail.com	2024-11-15 11:07:41
2	Irma	irma@gmail.com	2024-11-15 11:07:41
3	Rizka	lutfiahrizka23@gmail.com	2024-11-15 11:04:26

A screenshot of a web browser window titled "Users" from a Laravel application. The URL is 127.0.0.1:8000/users. The page displays a table of users with columns: No, Name, Email, and Creat-at. There are three entries. A blue "Add user" button is located in the top right corner of the table header.

No	Name	Email	Creat-at	
2	Salsa	salsabila@gmail.com	15 Nov 2024 21 minutes ago	View Edit
3	Irma	irma@gmail.com	15 Nov 2024 21 minutes ago	View Edit
1	Rizka	lutfiahrizka23@gmail.com	15 Nov 2024 24 minutes ago	View Edit

Create User

Name	<input type="text" value="putri"/>
Email	<input type="text" value="putri@gmail.com"/>
password	<input type="password" value="*****"/>
<input type="button" value="Save"/>	

Users

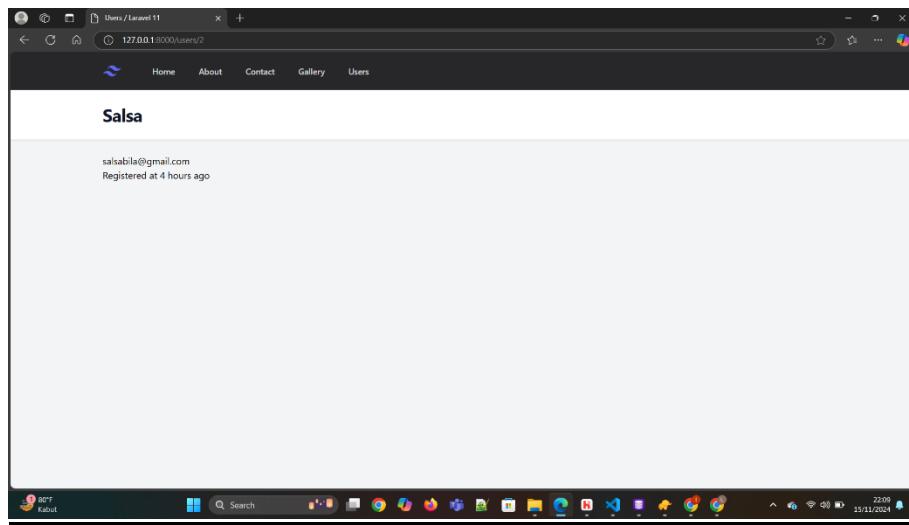
A list of all the users in your account including their name, title, email and role.

No	Name	Email	Created At
1	zakiadrim	zakiadrim@gmail.com	15 Nov 2024
2	putri	putri@gmail.com	15 Nov 2024
3	yesti agustin	yesti@gmail.com	15 Nov 2024
4	Salsa	salsabila@gmail.com	15 Nov 2024
5	Irma	irma@gmail.com	15 Nov 2024
6	Rizka	lutfahriza23@gmail.com	15 Nov 2024

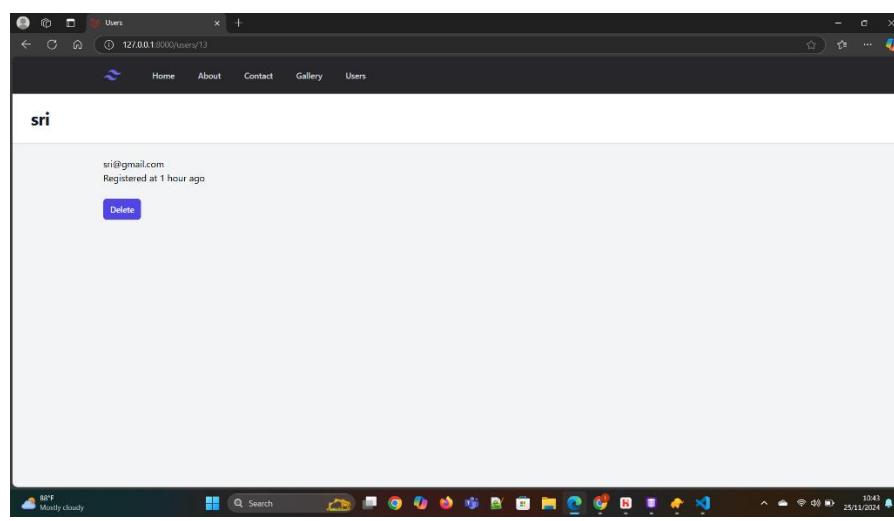
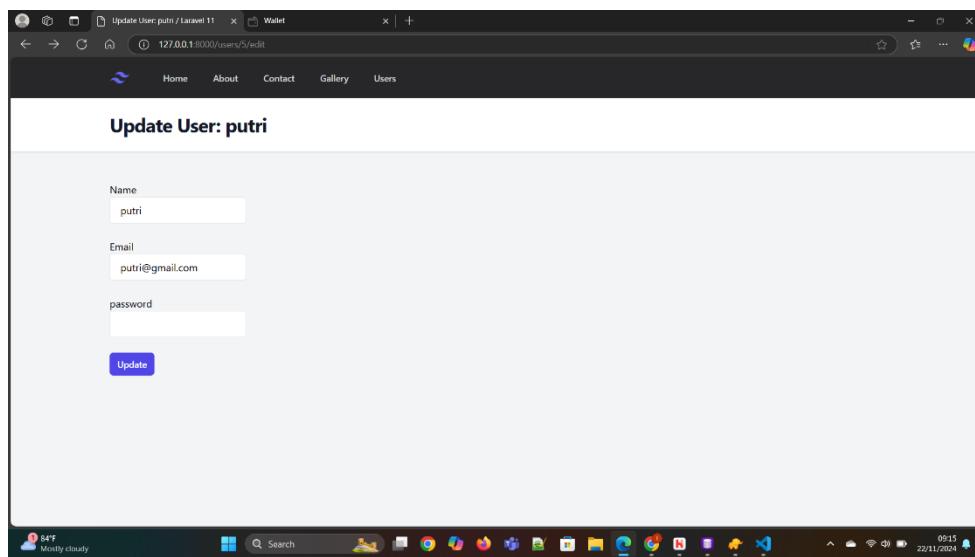
Users

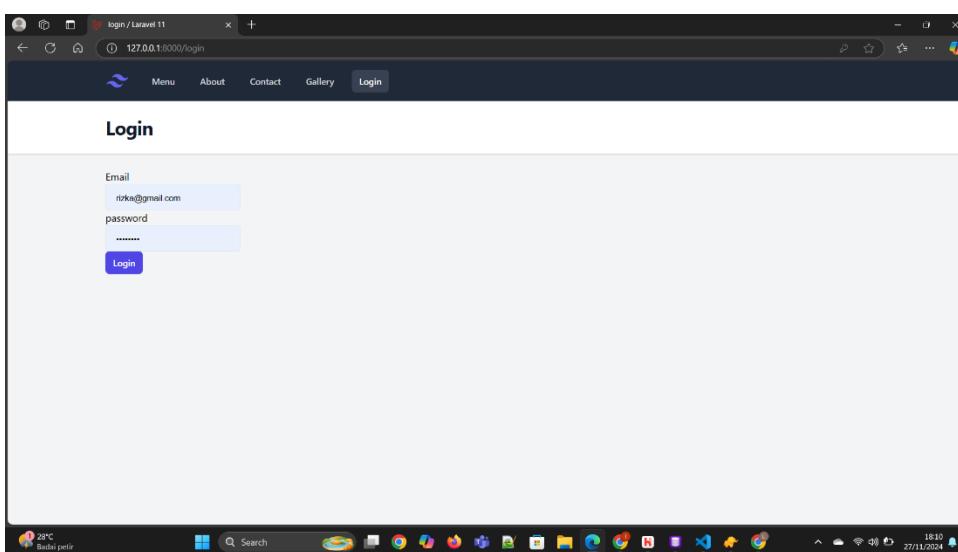
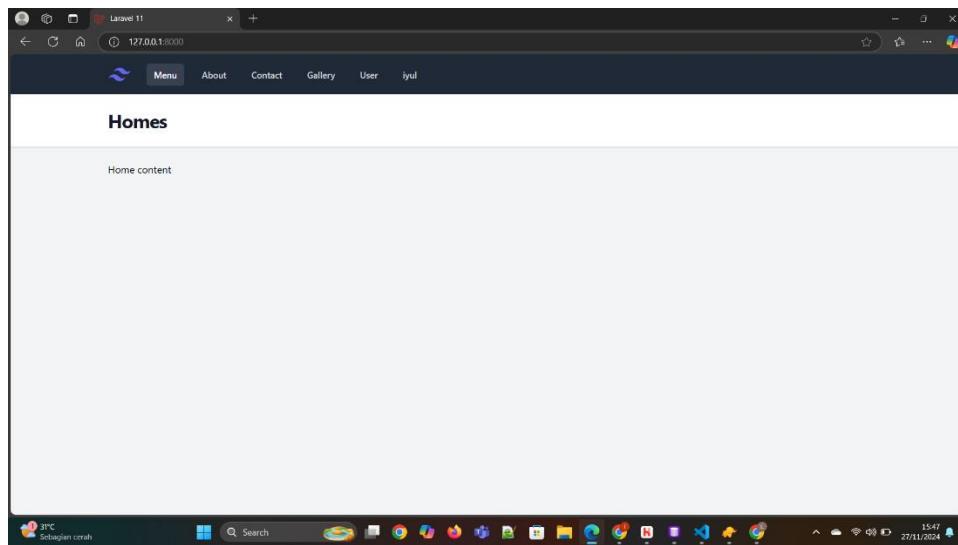
A list of all the users in your account including their name, title, email and role.

No	Name	Email	Created At	Action
1	putri	putri@gmail.com	15 November 2024	view
2	yesti agustin	yesti@gmail.com	15 November 2024	view
3	Salsa	salsabila@gmail.com	15 November 2024	view
4	Irma	irma@gmail.com	15 November 2024	view

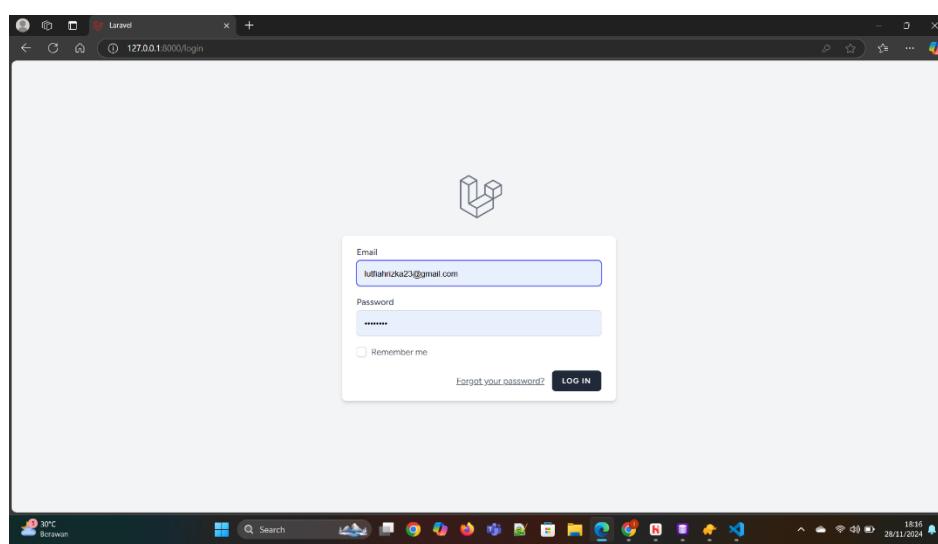


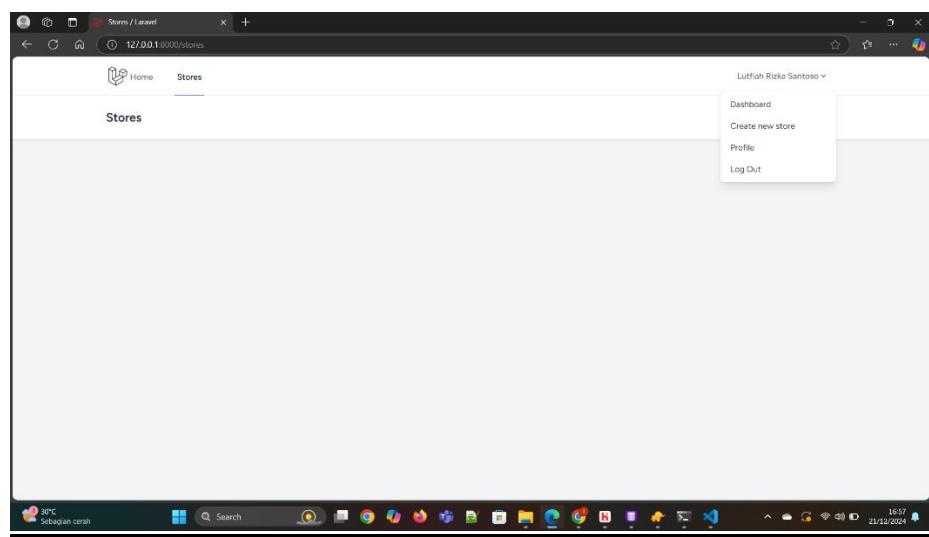
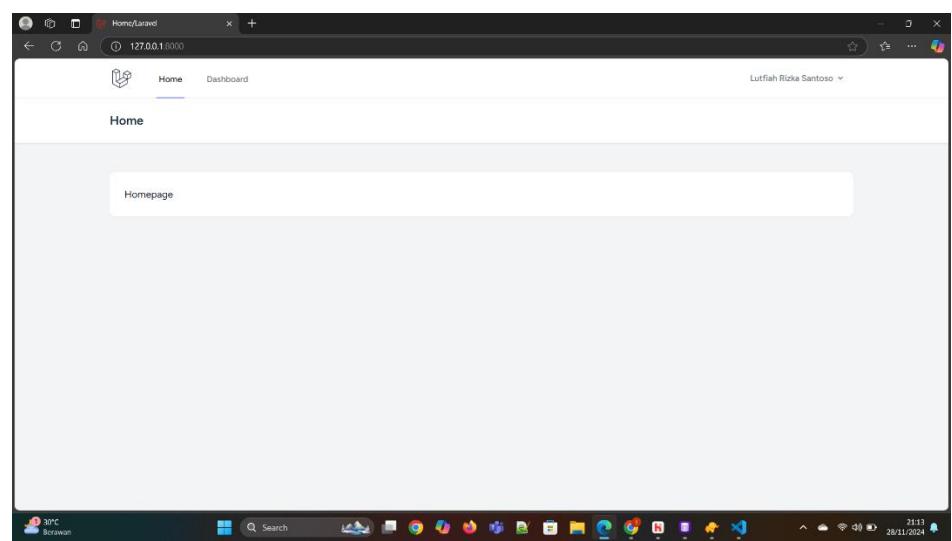
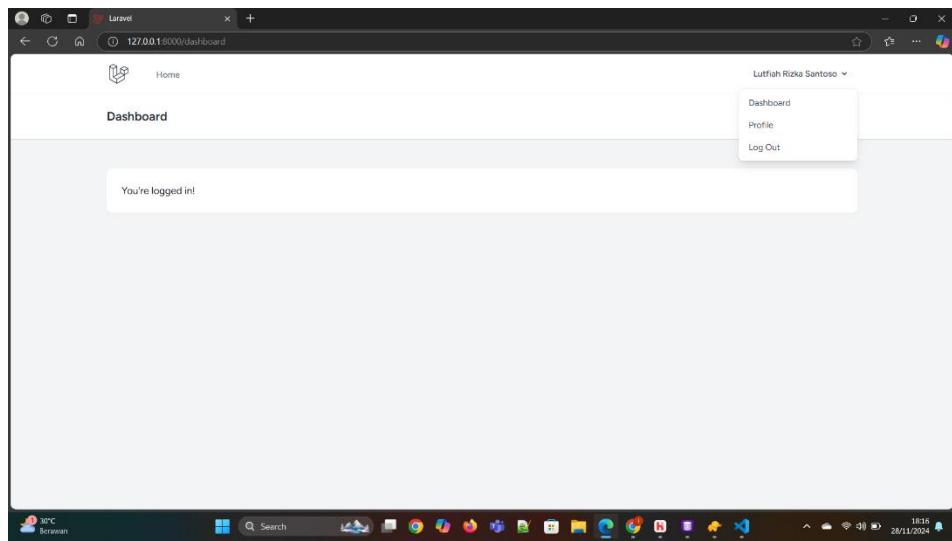
PRAKTIKUM 10 VID 13-17

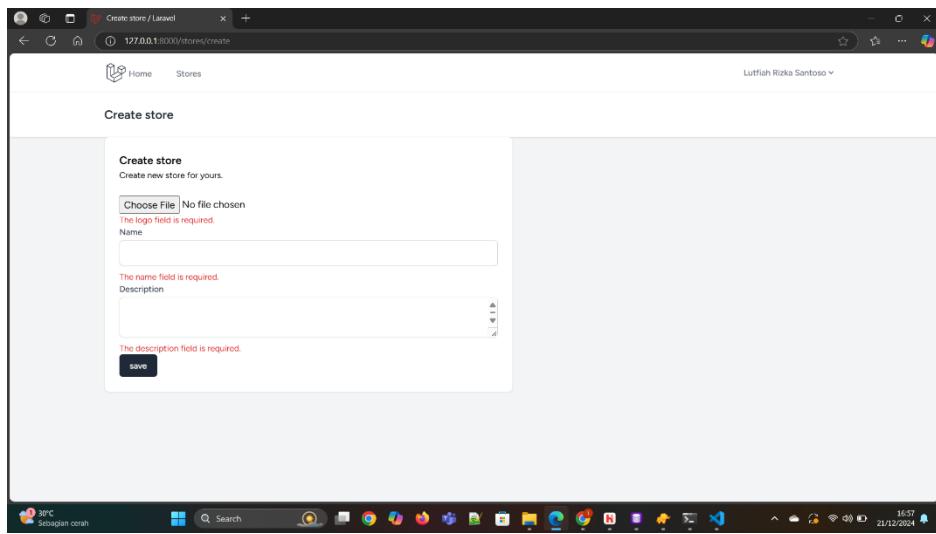




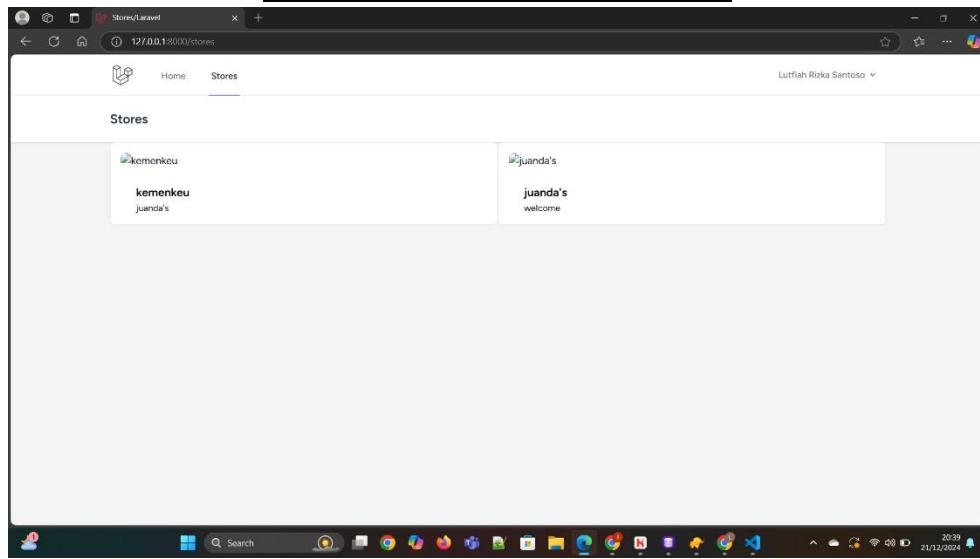
PRAKTIKUM 11 VID 19-21



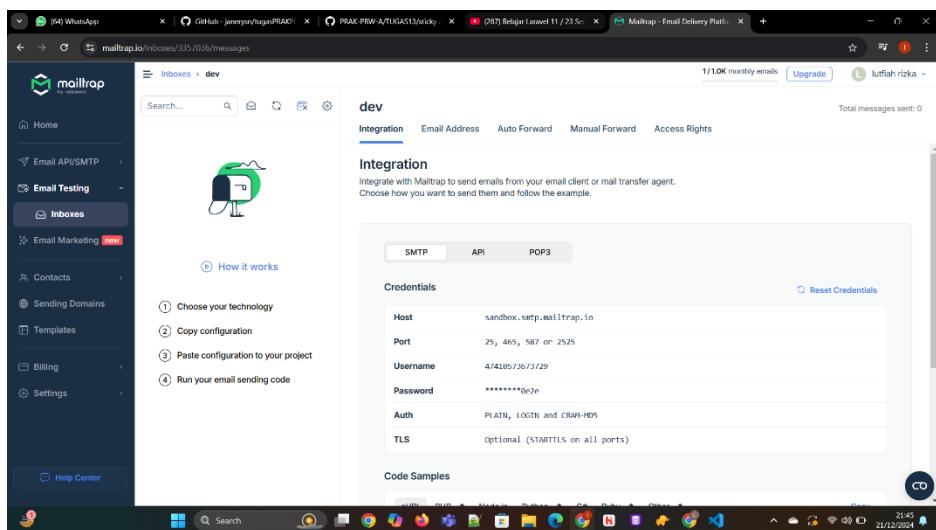
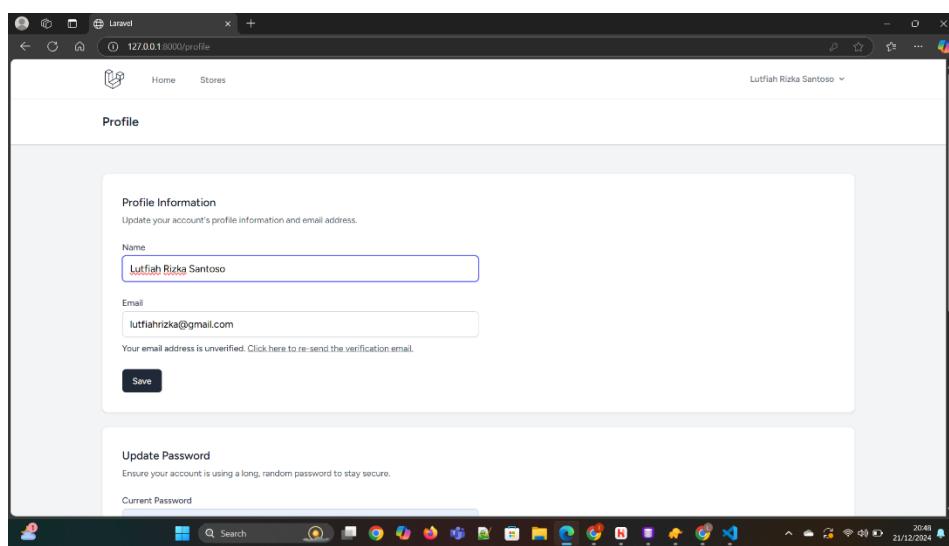
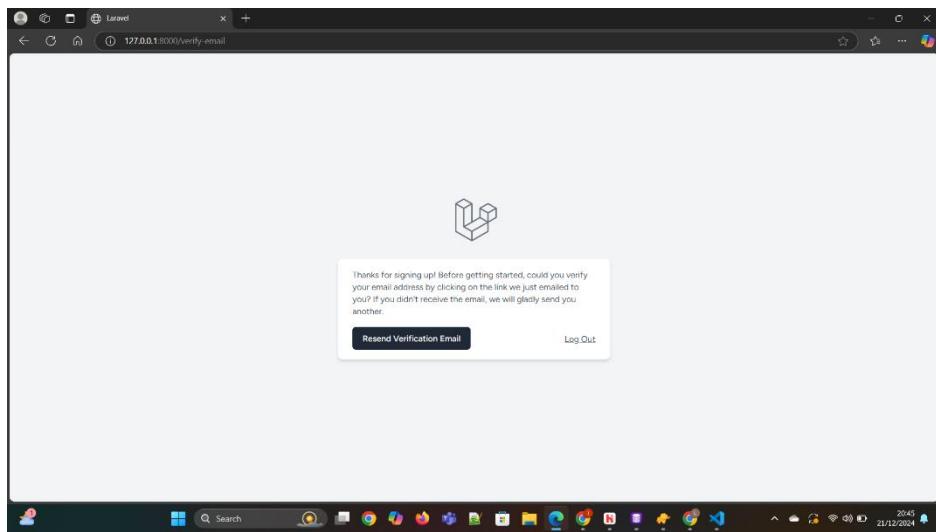


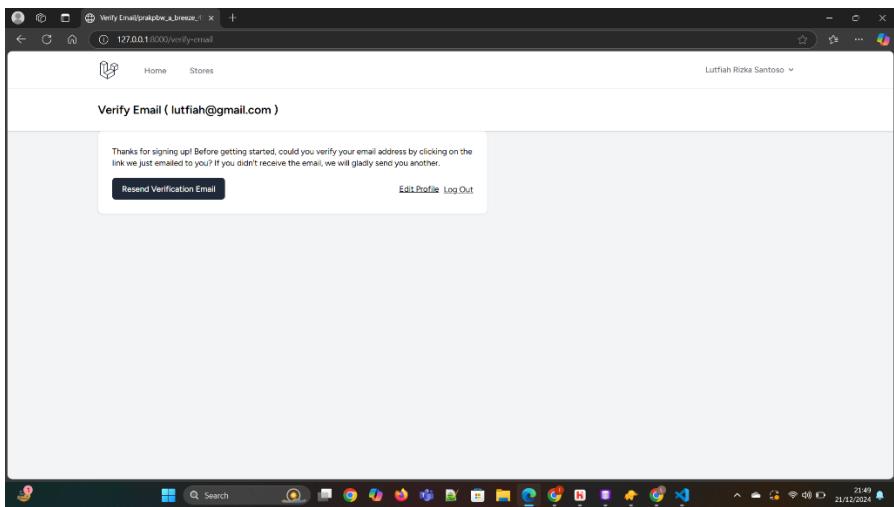


PRAKTIKUM 12 VID 22



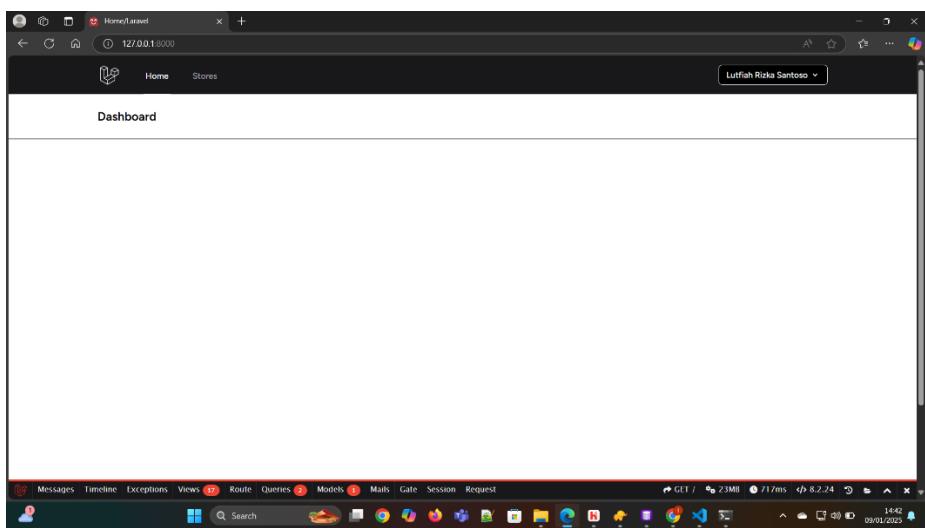
PRAKTIKUM 13 VID 23-24





DEVELOPMENT MySQL 8.0.39 - TiVo 2 : Merit : Laravel_Bootstrap : users										
Item	Queries	History	Stores	Users	Search for field...	Id	longname	remember_token	created_at	updated_at
1	Test user	lutfiah@gmail.com	2024-12-01 18:12:19	\$2y\$12\$8uqfBwefY17SiUQ2...L...p9L60000aTzvSD...m7D507XJThi	R0tWYe0IA	2024-12-01	M			
2	lutfiah Rizka Santoso	lutfiah123@gmail.com	2024-12-01 18:12:19	\$2y\$12\$TTIu3hEfjRmewTjHw...k8L...p9L60000aTzvSD...m7D507XJThi	M	2024-12-01	M			
3	Rizka	rizka@gmail.com	2024-12-01 18:12:19	\$2y\$12\$TTIu3hEfjRmewTjHw...k8L...p9L60000aTzvSD...m7D507XJThi	M	2024-12-01	M			
4	lutfiah Rizka Santoso	lutfiah123@gmail.com	2024-12-01 18:12:19	\$2y\$12\$TTIu3hEfjRmewTjHw...k8L...p9L60000aTzvSD...m7D507XJThi	M	2024-12-01	M			
5	lutfiah Rizka Santoso	lutfiah123@gmail.com	2024-12-01 18:12:19	\$2y\$12\$TTIu3hEfjRmewTjHw...k8L...p9L60000aTzvSD...m7D507XJThi	M	2024-12-01	M			
6	lutfiah Rizka Santoso	lutfiah123@gmail.com	2024-12-01 18:12:19	\$2y\$12\$TTIu3hEfjRmewTjHw...k8L...p9L60000aTzvSD...m7D507XJThi	M	2024-12-01	M			
7	lutfiah Rizka Santoso	lutfiah123@gmail.com	2024-12-01 18:12:19	\$2y\$12\$TTIu3hEfjRmewTjHw...k8L...p9L60000aTzvSD...m7D507XJThi	M	2024-12-01	M			
8	lutfiah Rizka Santoso	lutfiah123@gmail.com	2024-12-01 18:12:19	\$2y\$12\$TTIu3hEfjRmewTjHw...k8L...p9L60000aTzvSD...m7D507XJThi	M	2024-12-01	M			
9	lutfiah Rizka Santoso	lutfiah@gmail.com	2024-12-01 18:12:19	\$2y\$12\$TTIu3hEfjRmewTjHw...k8L...p9L60000aTzvSD...m7D507XJThi	M	2024-12-01	M			
10	lutfiah Rizka Santoso	rizka@gmail.com	2024-12-01 18:12:19	\$2y\$12\$TTIu3hEfjRmewTjHw...k8L...p9L60000aTzvSD...m7D507XJThi	M	2024-12-01	M			
11	lutfiah Rizka Santoso	rizka123@gmail.com	2024-12-01 18:12:19	\$2y\$12\$TTIu3hEfjRmewTjHw...k8L...p9L60000aTzvSD...m7D507XJThi	M	2024-12-01	M			

PRAKTIKUM 14 VID 26



PRAKTIKUM 15 Vid 27-31

