

# **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 terbaru 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. Menerapkan 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 Ansible 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

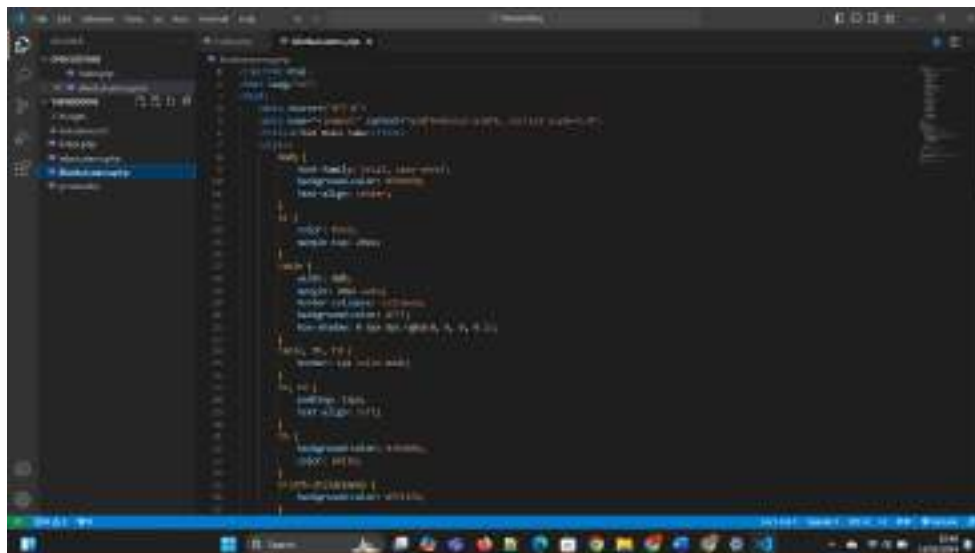
Tujuan praktikum : memberikan pemahaman dan keterampilan dalam menginstal dan mengkonfigurasi web stack yang umum digunakan dalam pengembangan web.

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**

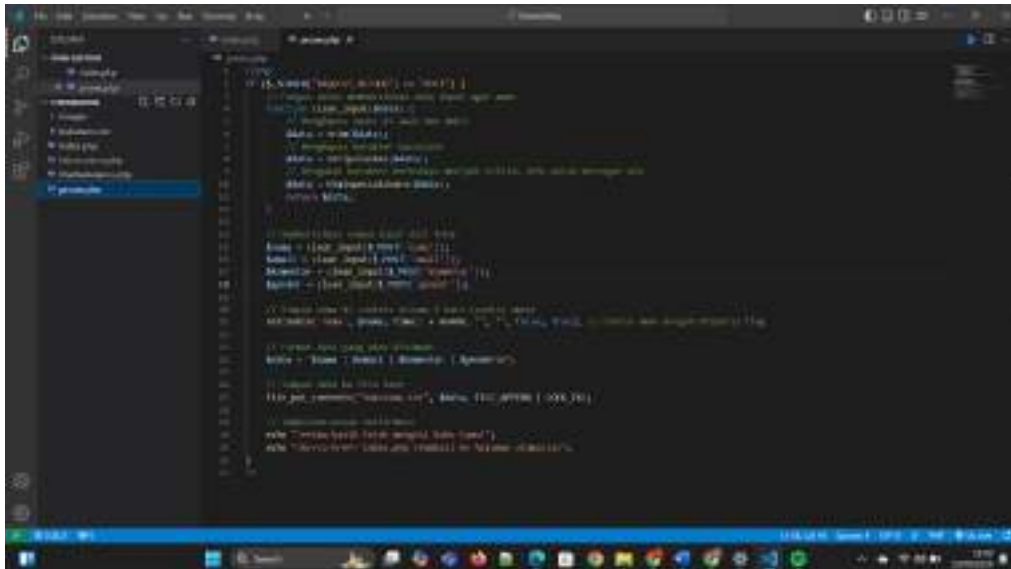
### **lihatbukutamu.php**



### **PHP 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**



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 &lt;) 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.

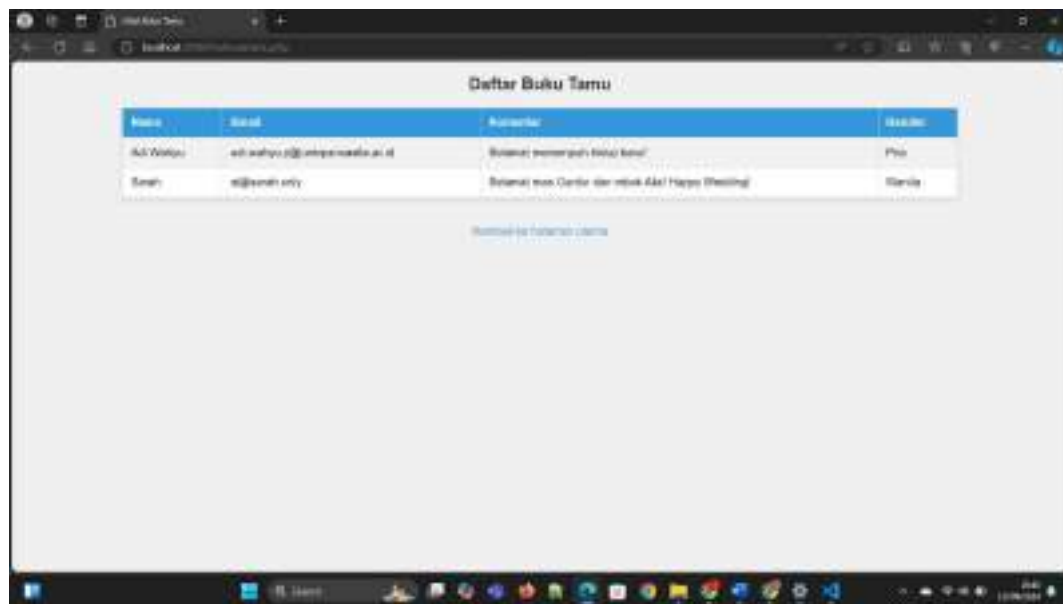
### Landing page



### Form Input

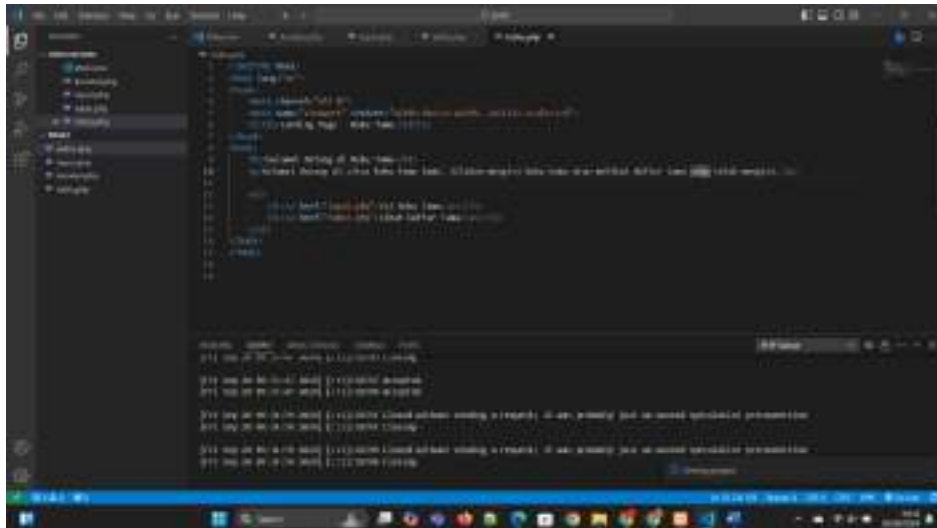
A screenshot of a web browser displaying a guest registration form. The page has a light gray background. At the top, the title "Isi Buku Tamu" is centered in a bold, black font. Below the title is a white form box with a thin gray border. Inside the form box, there are four input fields: "Nama:" (Name), "Email:", "Komentar:" (Comment), and "Gender:". The "Nama:" field is a single-line text input. The "Email:" field is a single-line text input. The "Komentar:" field is a multi-line text area. The "Gender:" field has two radio buttons labeled "Pria" (Male) and "Wanita" (Female). Below the form box is a blue button with white text that says "Daftar". At the bottom of the page, there is a small, faint link that says "Kontak kami melalui WhatsApp". The browser's address bar shows "localhost:3000". The Windows taskbar is visible at the bottom of the screen.

halaman daftar buku tamu.



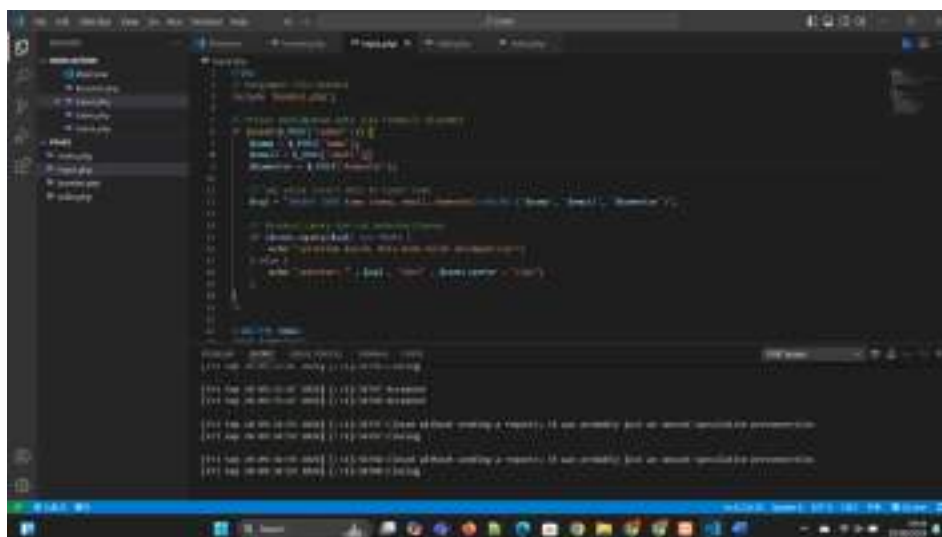
## PRAKTIKUM 3

Index.php



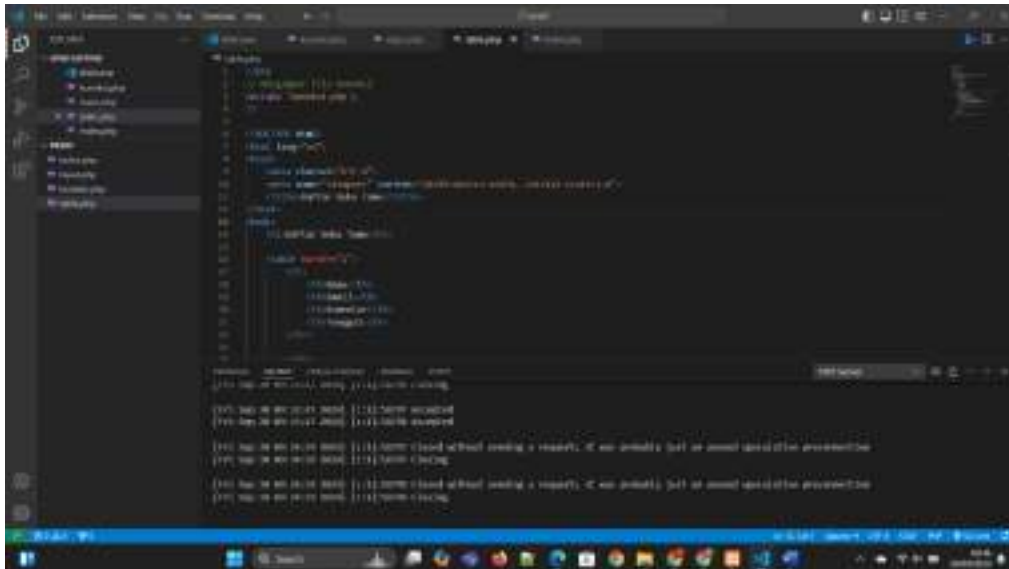
Membuat halaman landing sederhana untuk buku tamu. Terdapat judul, paragraf sambutan, dan dua tautan: satu untuk mengisi buku tamu (input.php) dan satu lagi untuk melihat daftar tamu (tabel.php).

### Input.php



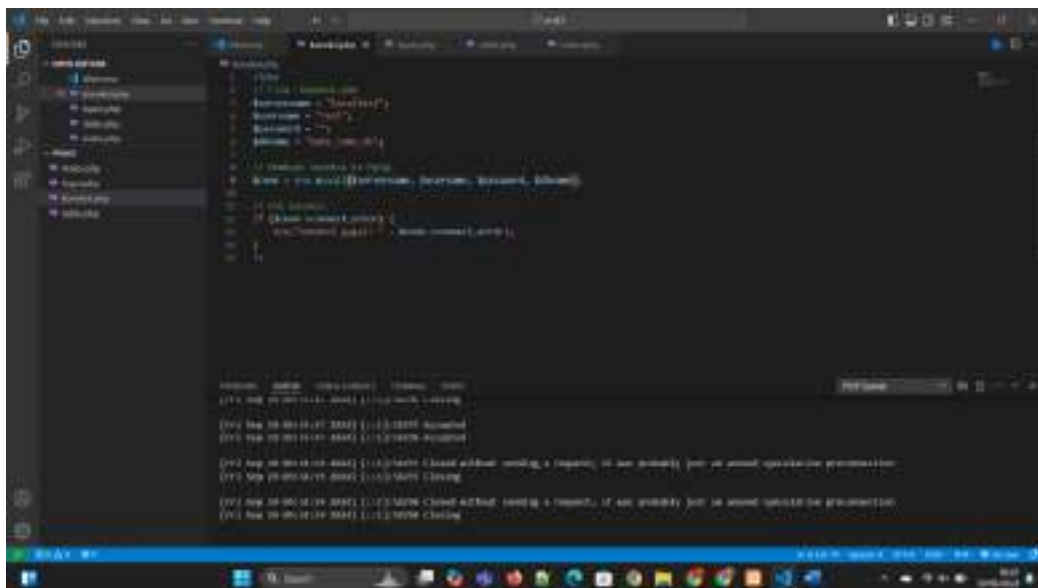
Kode PHP in menginput data buku tamu. Ketika formulir disubmit, data nama, email, dan komentar dikirimkan dan disimpan ke dalam tabel tamu di database melalui query SQL. Jika berhasil, muncul pesan konfirmasi; jika gagal, ditampilkan pesan error. Halaman HTML berisi formulir untuk mengisi data tamu dan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

### table.php



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.

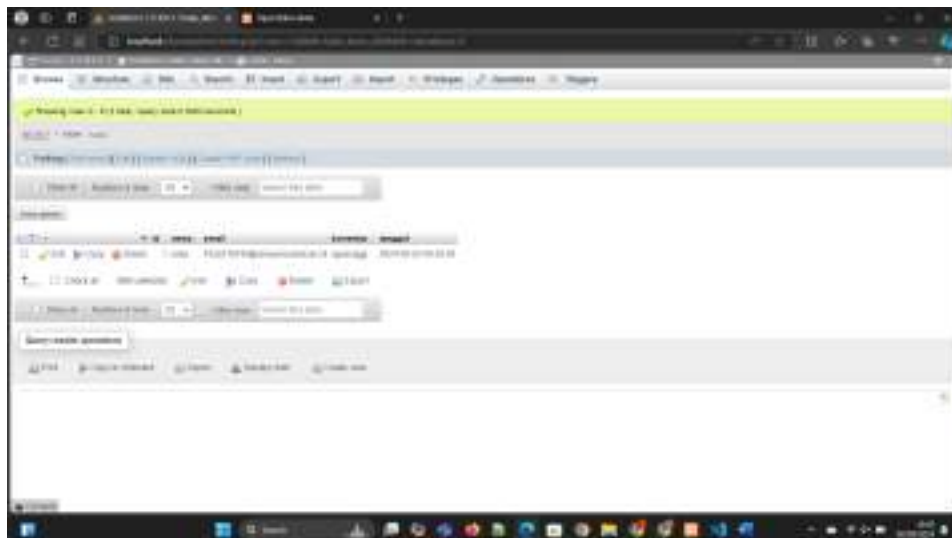
#### koneksi.php



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().

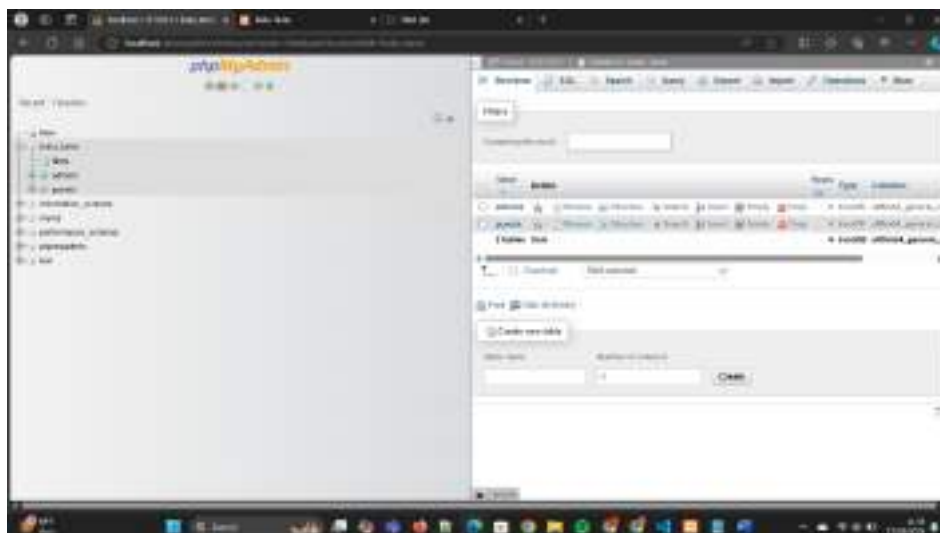
**Berhasil menginput buku tamu**





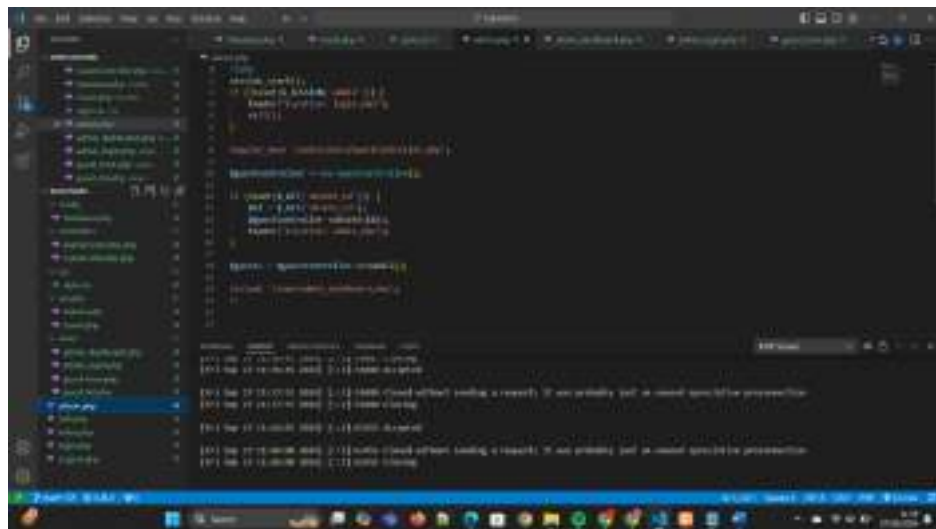
## PRAKTIKUM 4

### 1. Database di MySQL

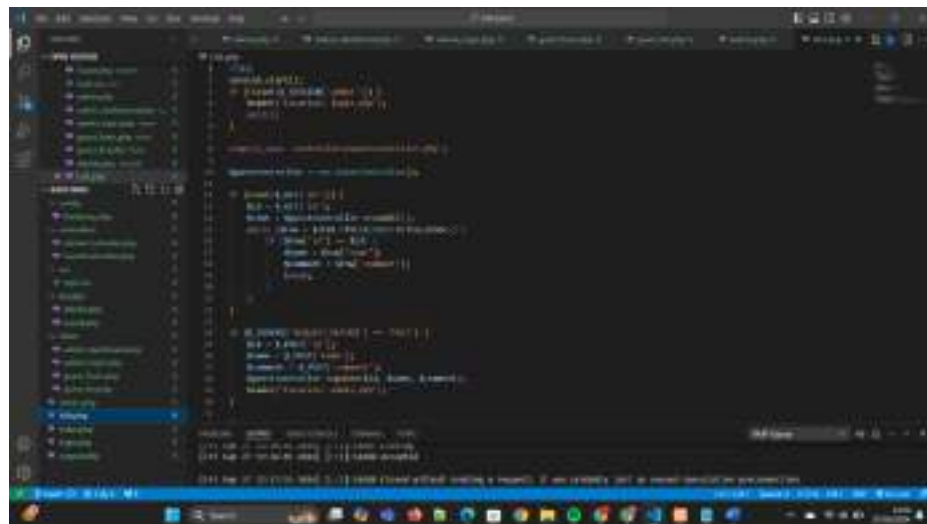


### 2.Struktur Direktori Proyek

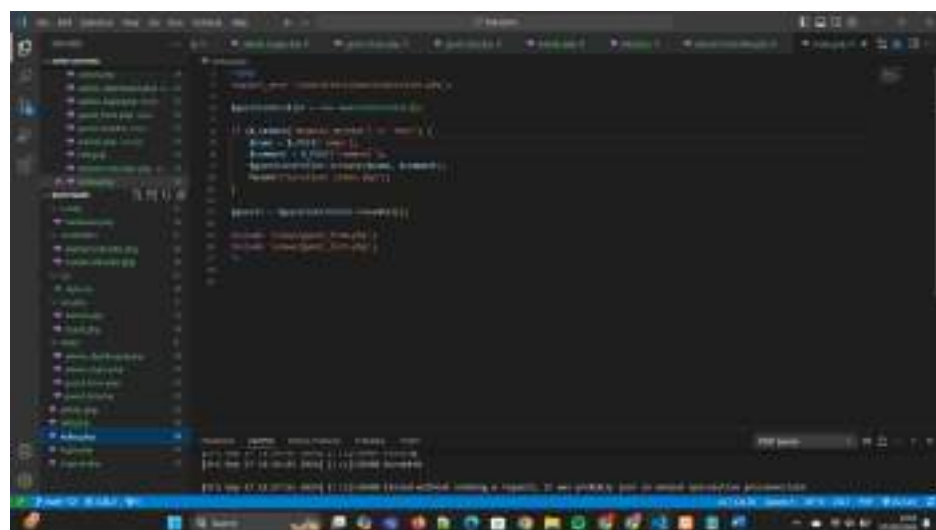
admin.php



Edit.php



## Index.php



```
1 session_start();
2 session_unset();
3 session_destroy();
4 header('Location: login.php');
5 exit();
6
7 // Fungsi untuk login
8 function login($username, $password) {
9     $username = trim($username);
10    $password = trim($password);
11
12    // Cek apakah username dan password ada
13    if ($username == '' || $password == '') {
14        return false;
15    }
16
17    // Cek apakah username dan password benar
18    if ($username == 'admin' && $password == 'admin') {
19        // Login berhasil
20        session_regenerate_id(true);
21        session_set_cookie_params(0);
22        session_start();
23        $_SESSION['username'] = $username;
24        $_SESSION['password'] = $password;
25        header('Location: index.php');
26        exit();
27    }
28
29    // Login gagal
30    return false;
31}
32
33 // Fungsi untuk logout
34 function logout() {
35     session_unset();
36     session_destroy();
37     header('Location: login.php');
38     exit();
39}
```

Logout.php

```
1 session_start();
2 session_unset();
3 session_destroy();
4 header('Location: login.php');
5 exit();
6
7 // Fungsi untuk login
8 function login($username, $password) {
9     $username = trim($username);
10    $password = trim($password);
11
12    // Cek apakah username dan password ada
13    if ($username == '' || $password == '') {
14        return false;
15    }
16
17    // Cek apakah username dan password benar
18    if ($username == 'admin' && $password == 'admin') {
19        // Login berhasil
20        session_regenerate_id(true);
21        session_set_cookie_params(0);
22        session_start();
23        $_SESSION['username'] = $username;
24        $_SESSION['password'] = $password;
25        header('Location: index.php');
26        exit();
27    }
28
29    // Login gagal
30    return false;
31}
32
33 // Fungsi untuk logout
34 function logout() {
35     session_unset();
36     session_destroy();
37     header('Location: login.php');
38     exit();
39}
```

SS Hasil Bukutamu

Selamat Datang di Buku Tamu

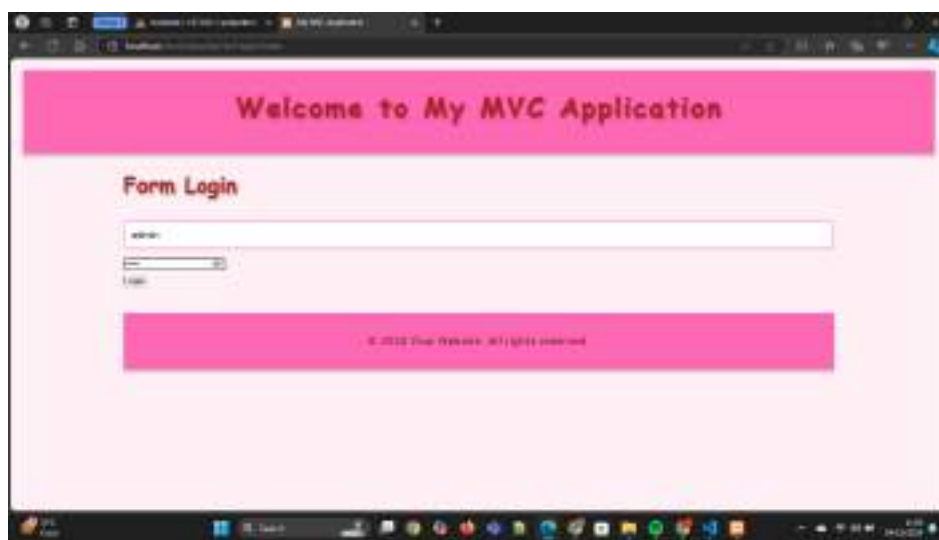
Daftar Komentar

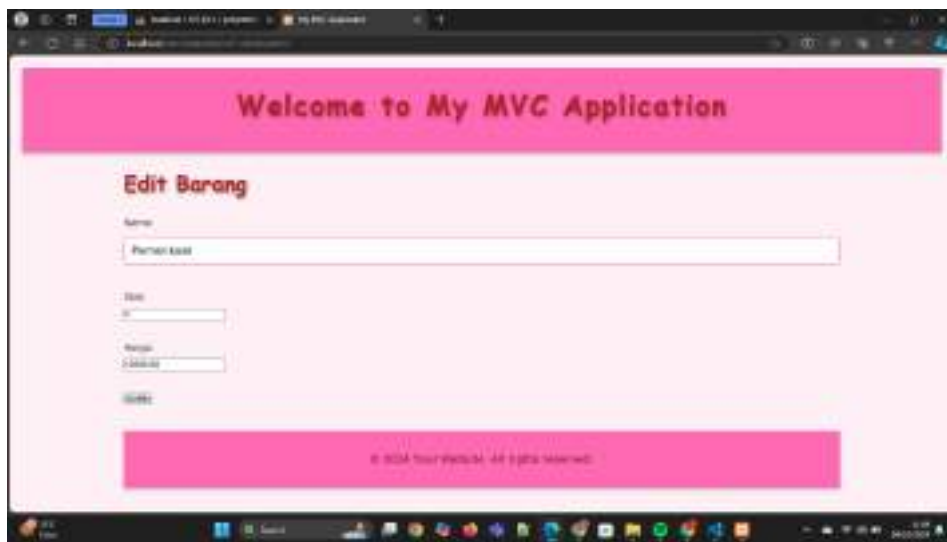
Nama	Komentar	Waktu
Adi	menyakit	2024-07-07 10:00:00

## SS Halaman Login.php



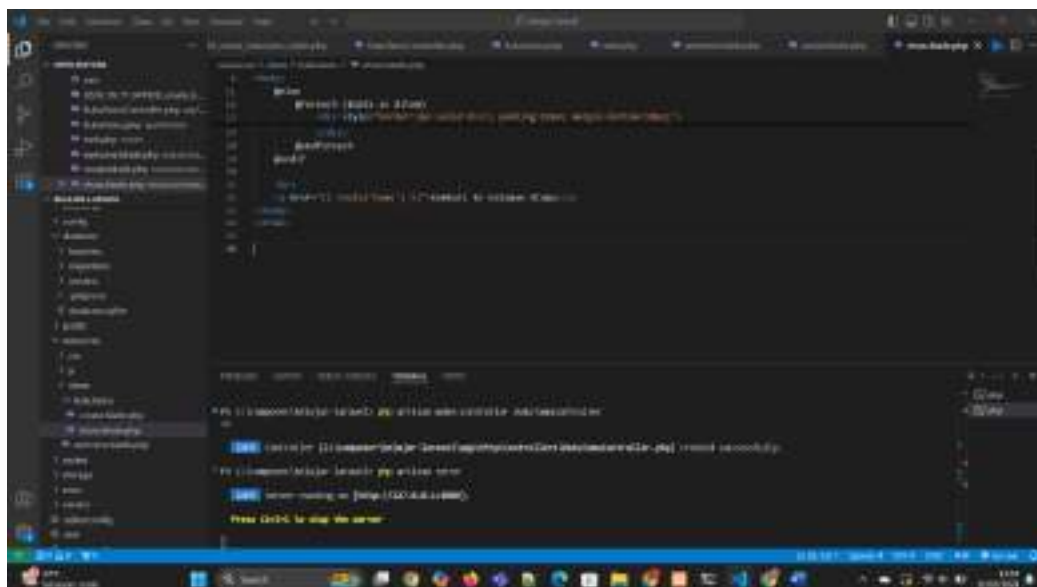
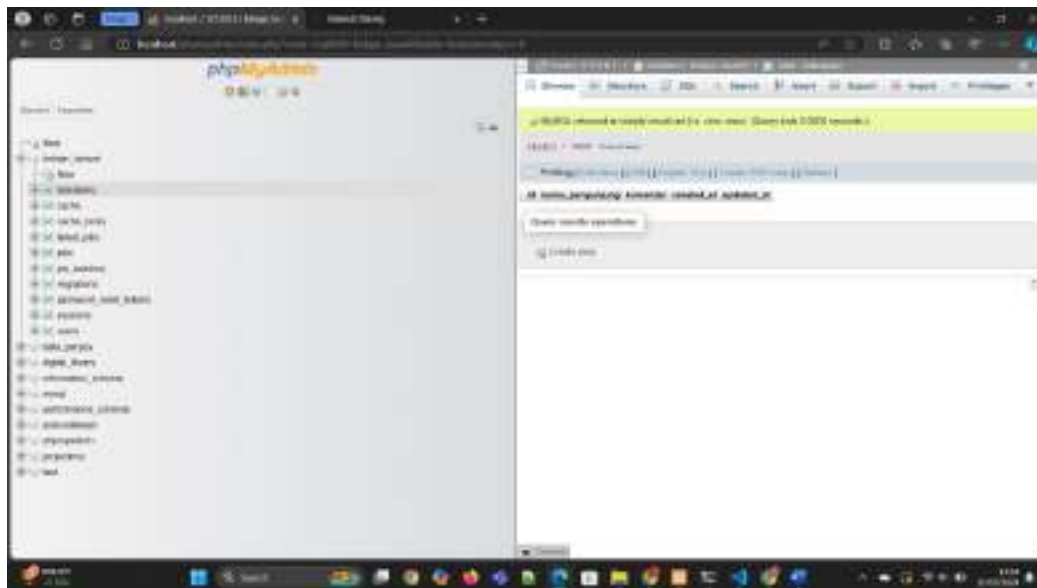
## PRAKTIKUM 5





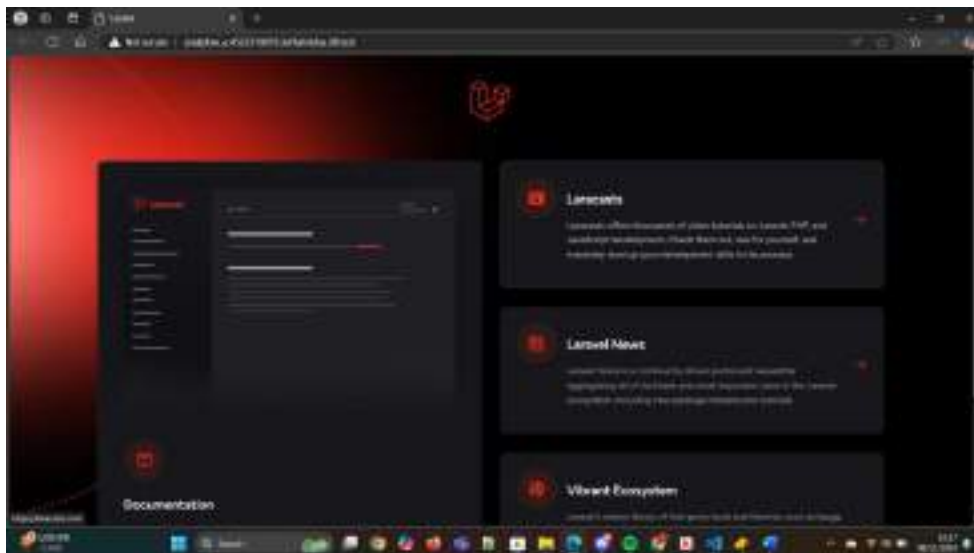
## PRAKTIKUM 6



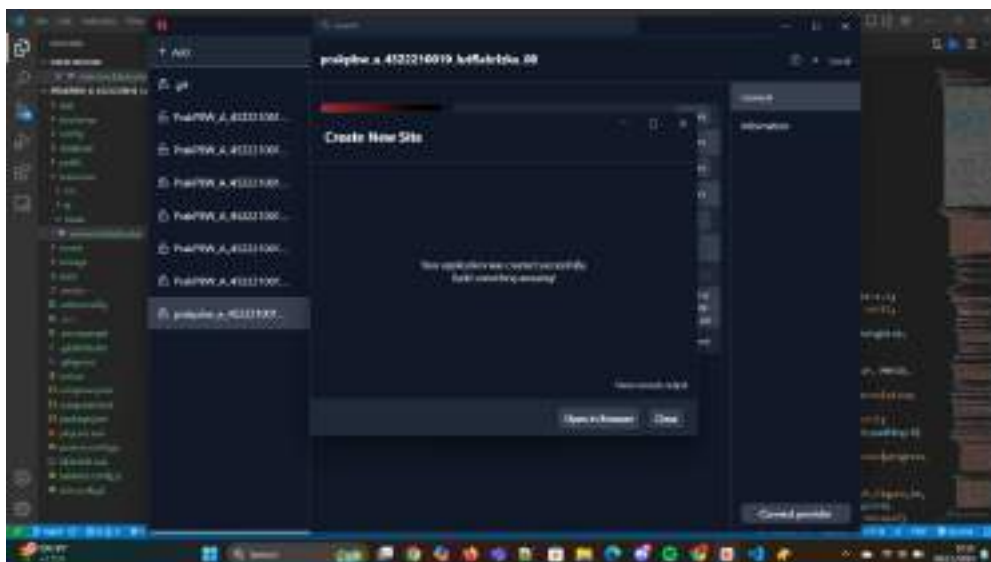


## PRAKTIKUM 7

- Bagaimana tampilan utama?



- Sukses Herd

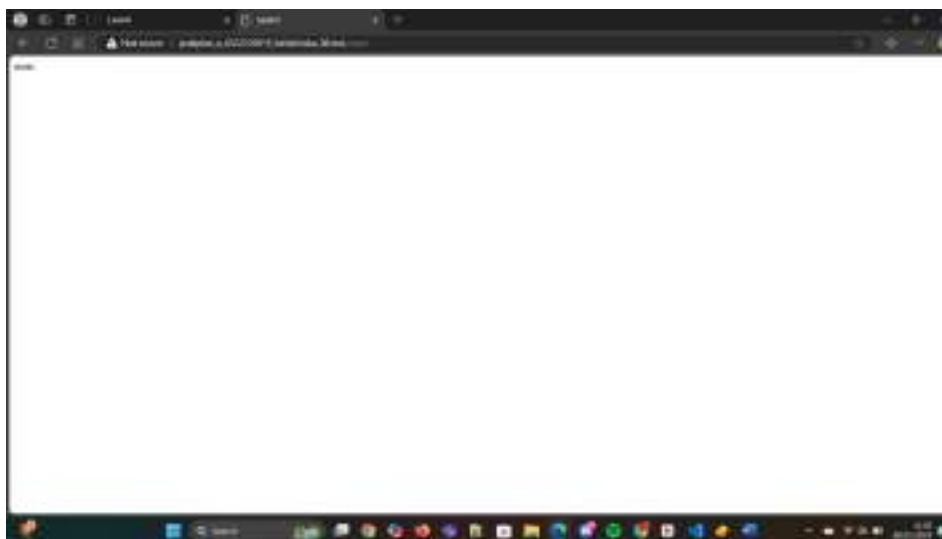
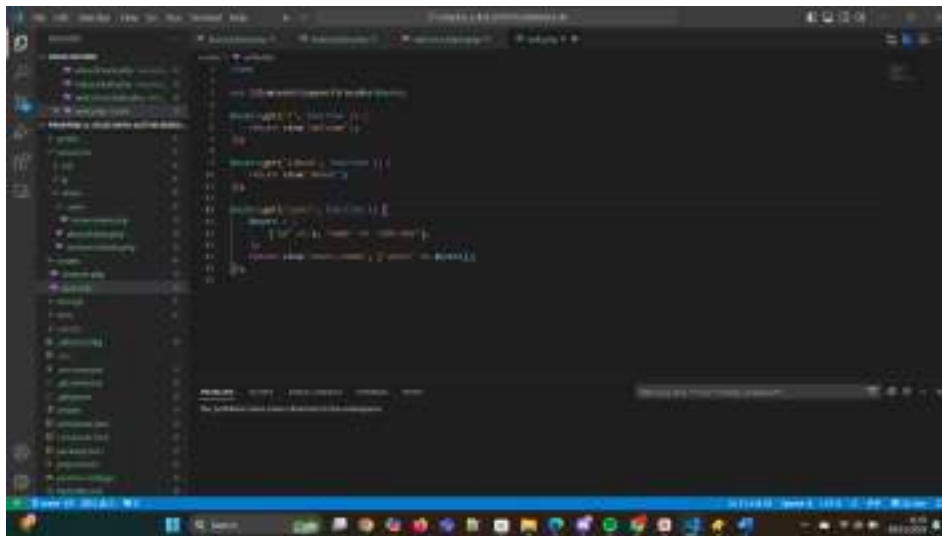


- Blade About





- Tampilan users.index



- Bagaimana tampilan HomePage

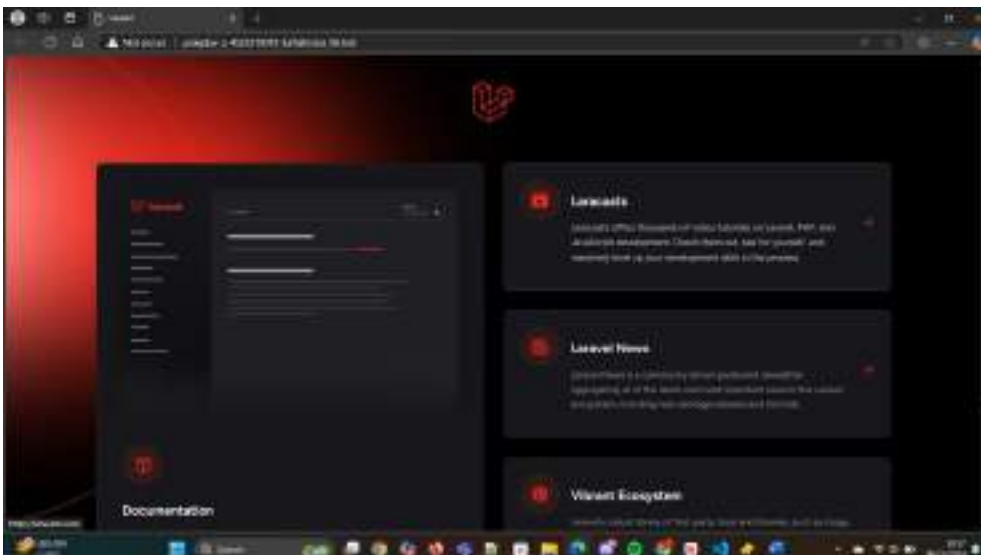
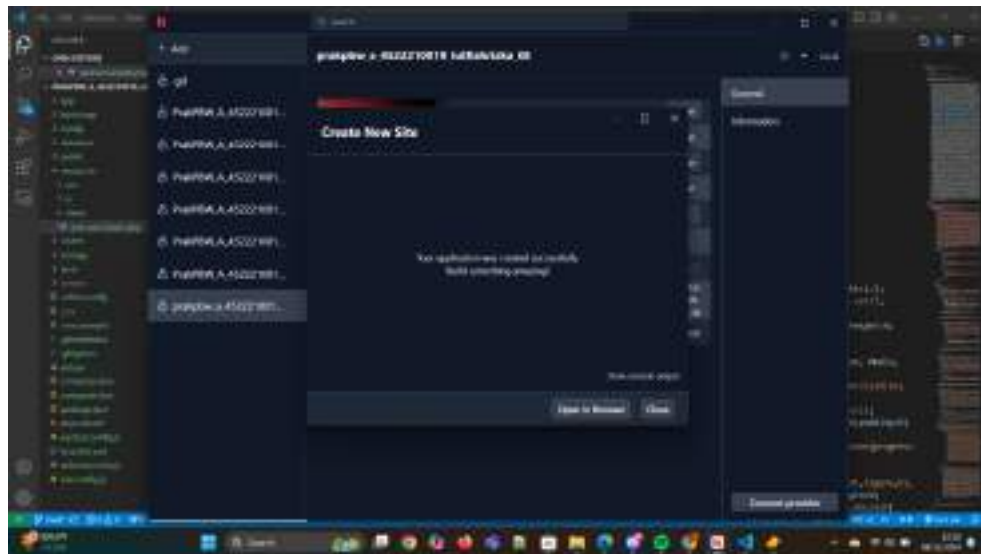


- Setelah Styling Tailwind CSS

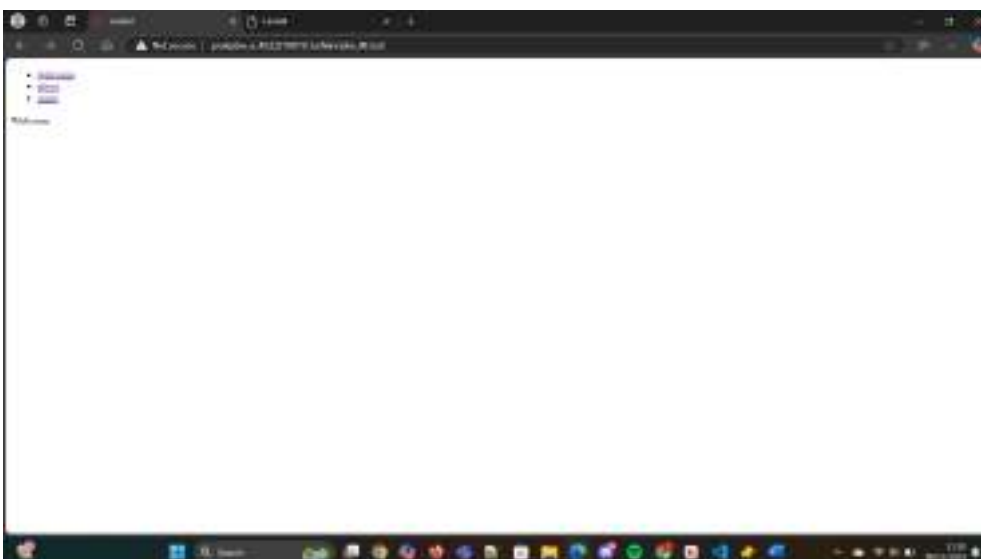




## Mengcreate Package Laravel Menggunakan Herd



## Tampilan HomePage

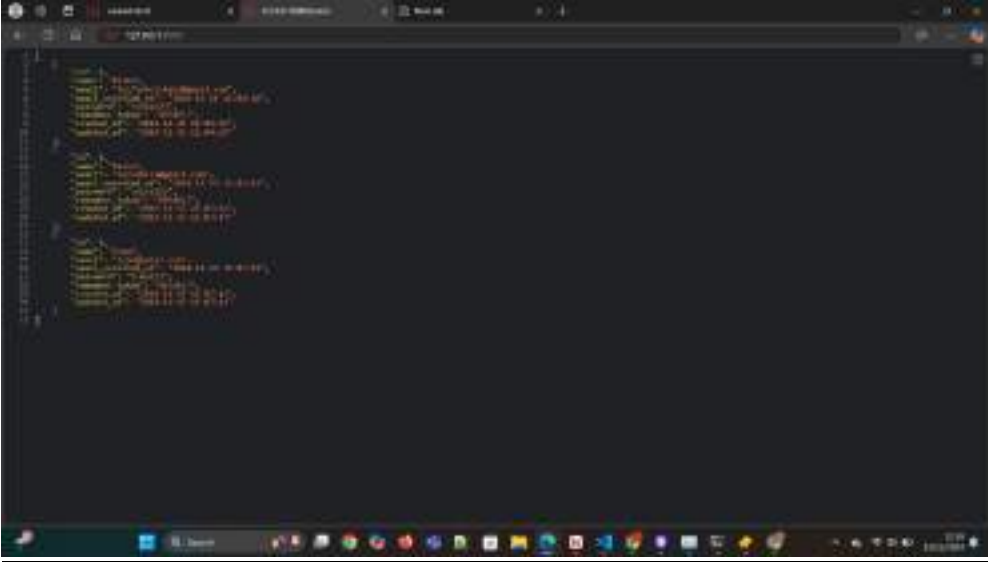


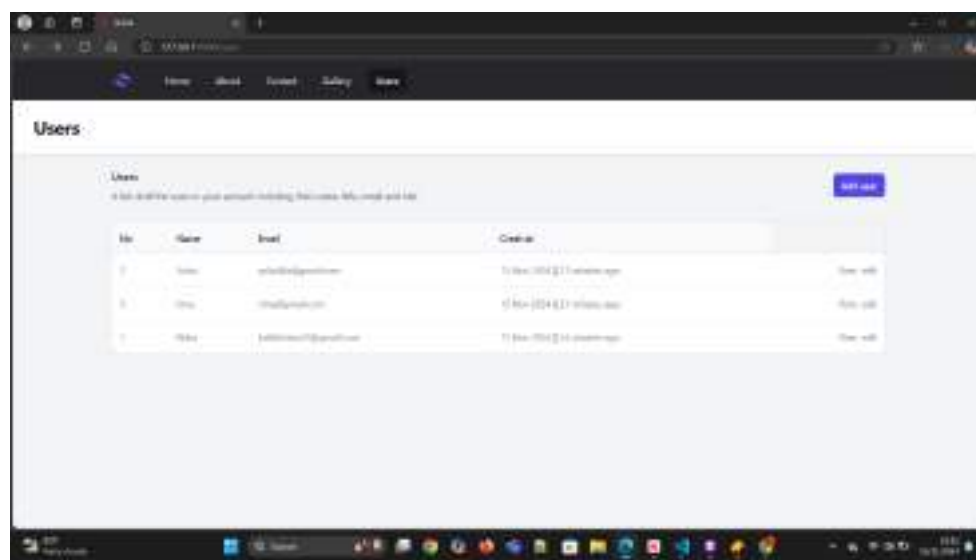
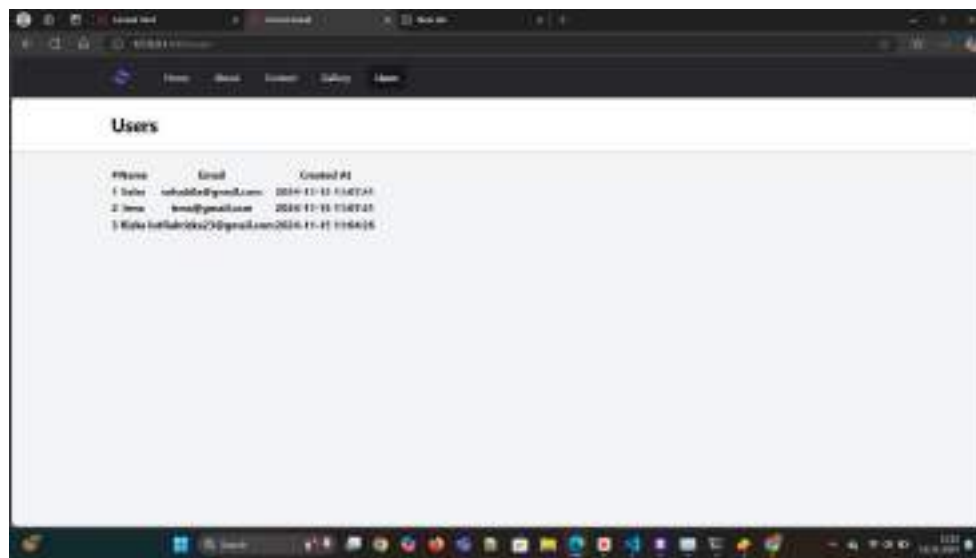
## Styling CSS Tailwind

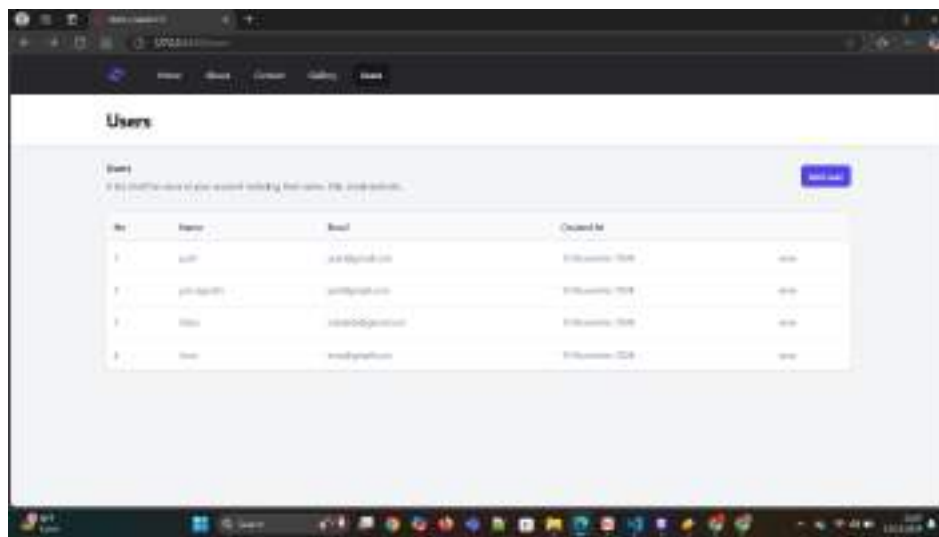
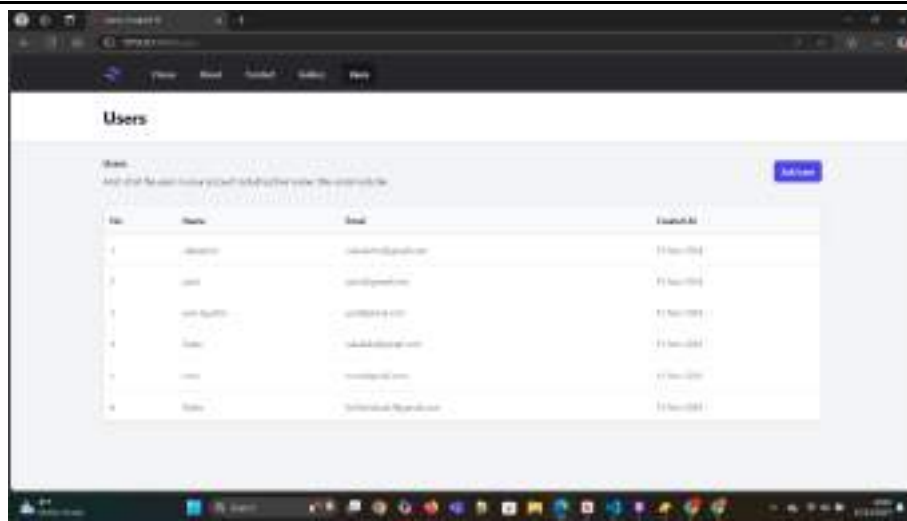


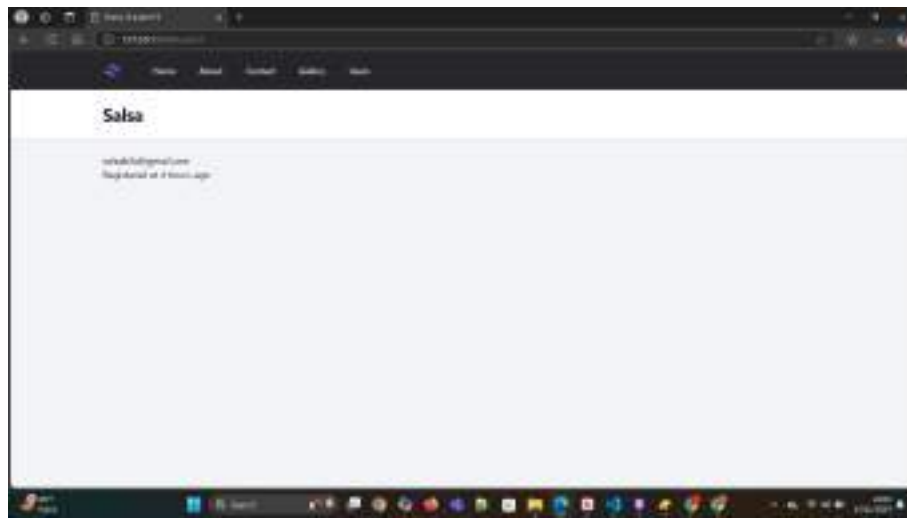
## PRAKTIKUM 9 VID 7-12

## Menampilkan Data dari Database

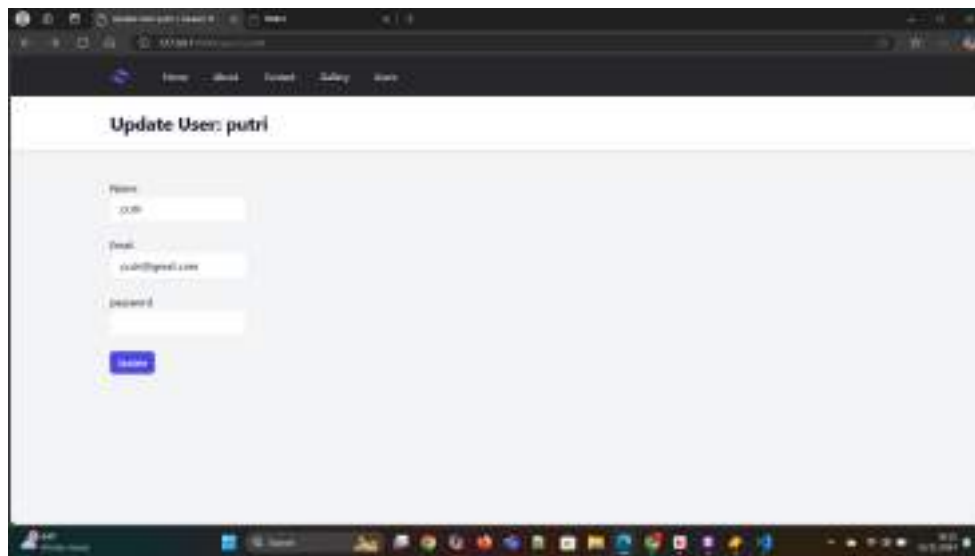




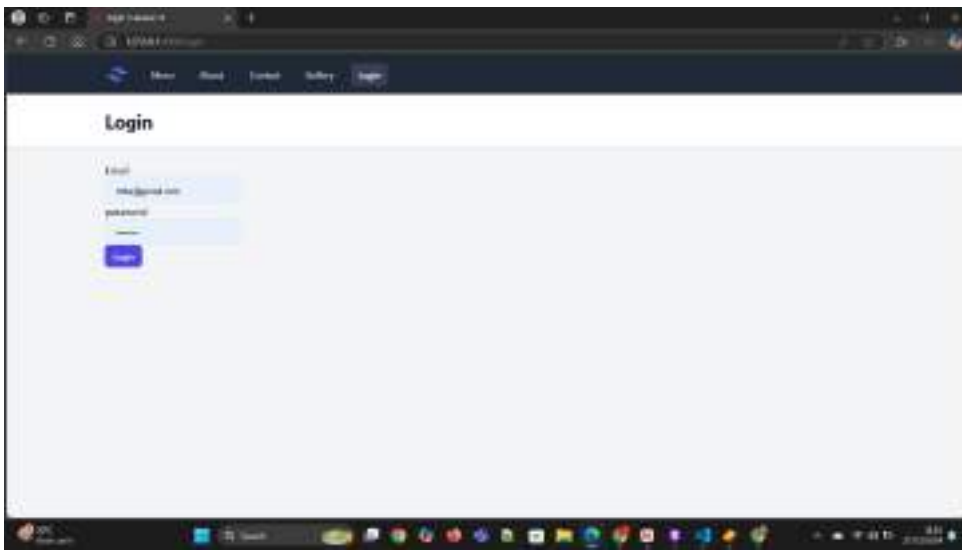
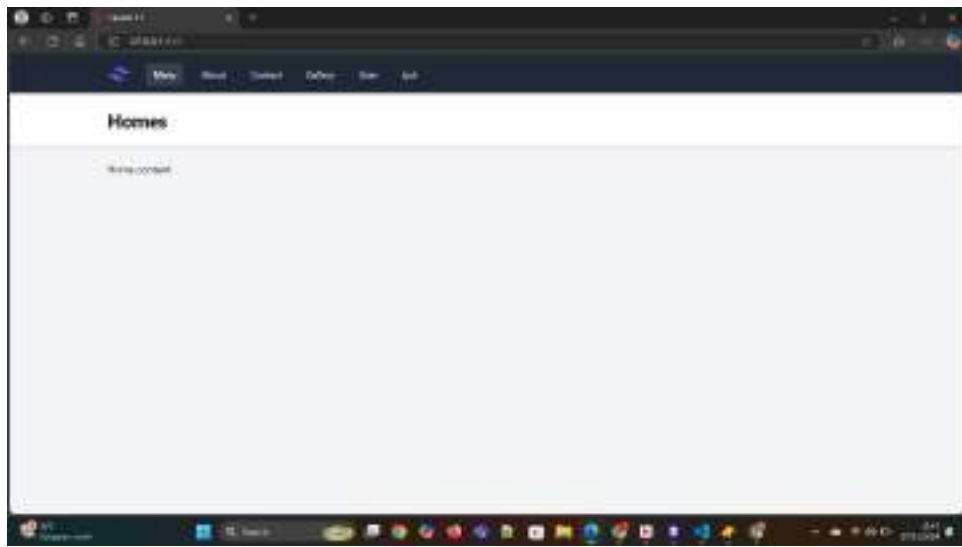




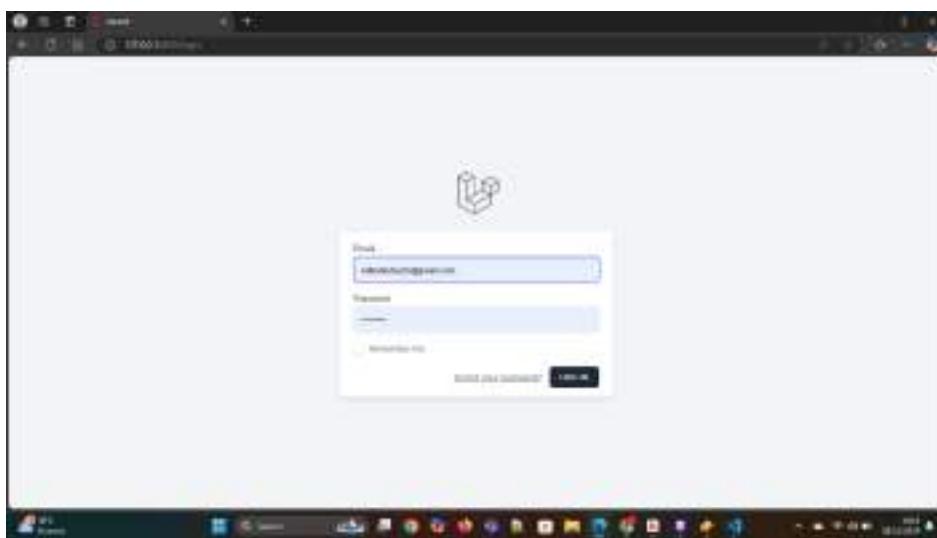
## PRAKTIKUM 10 VID 13-17

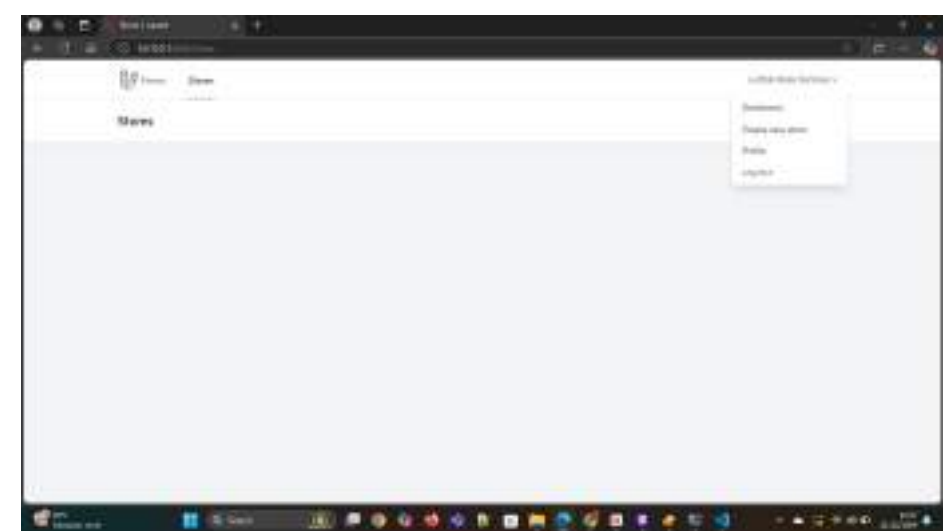
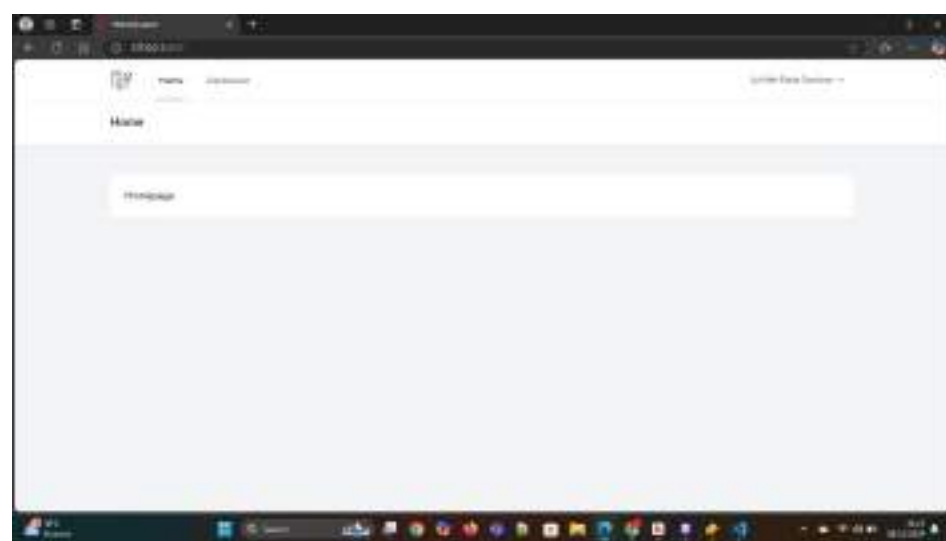
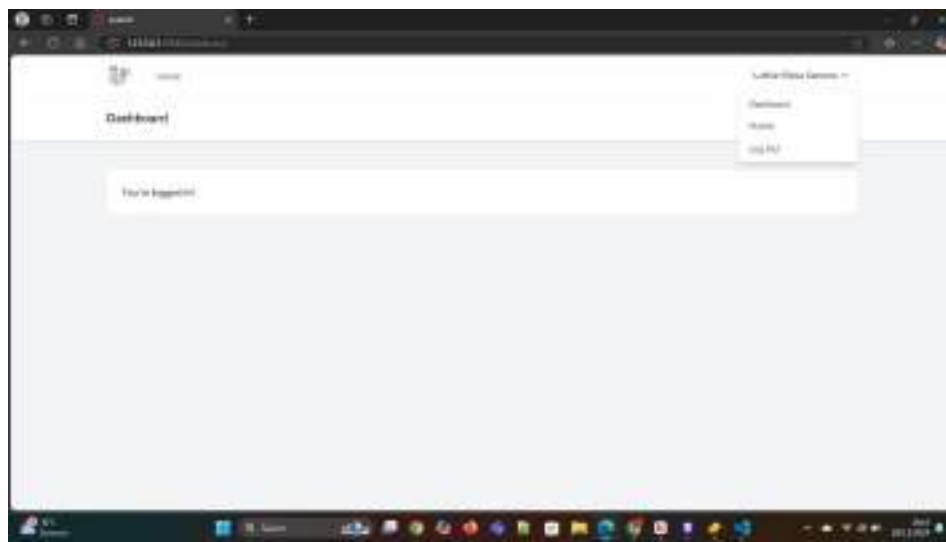


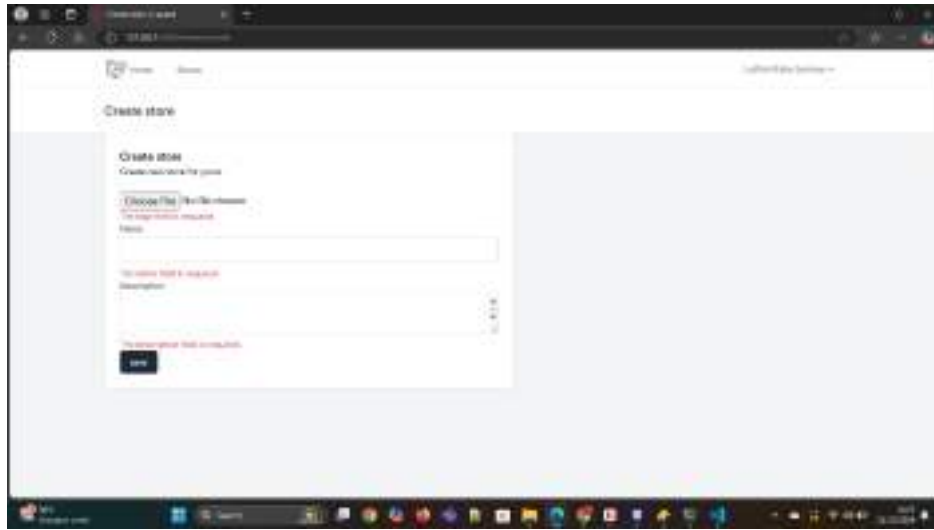




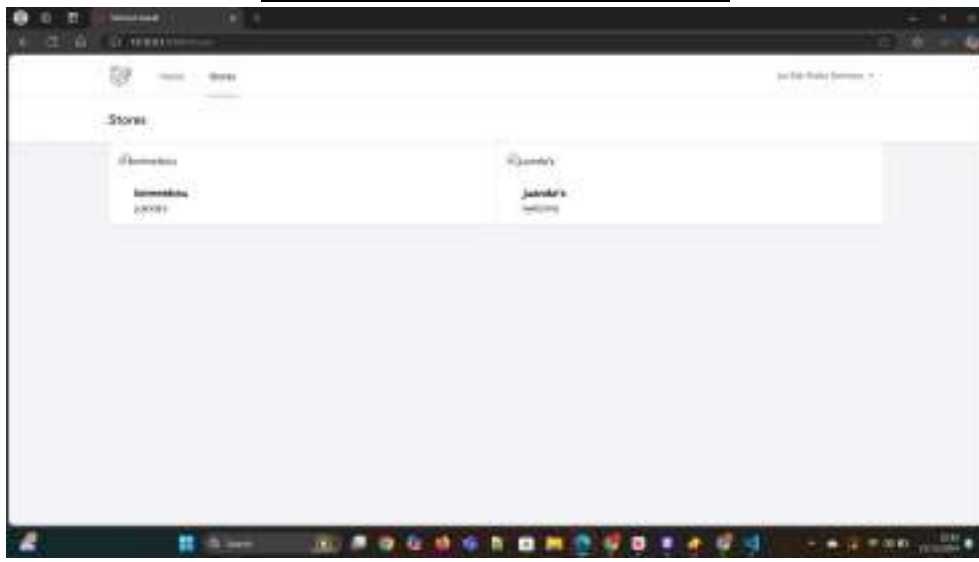
## PRAKTIKUM 11 VID 19-21



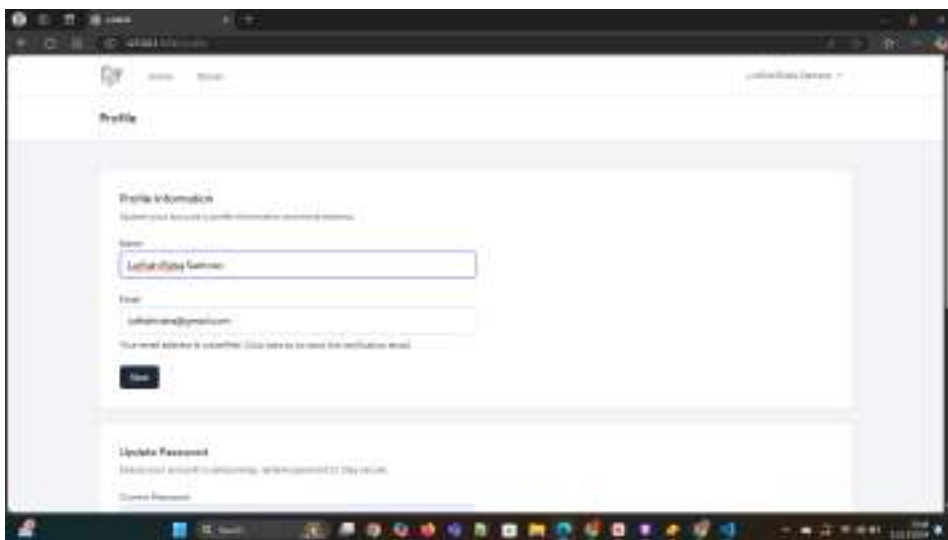
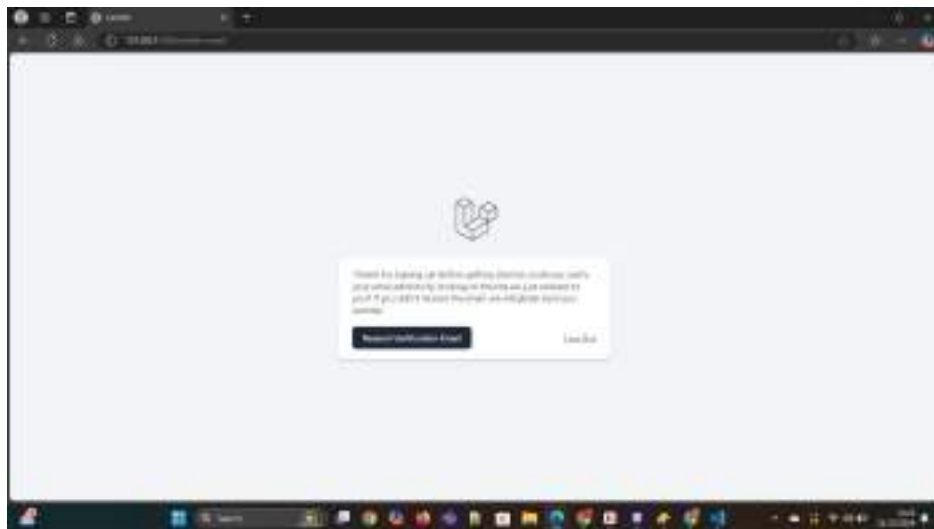


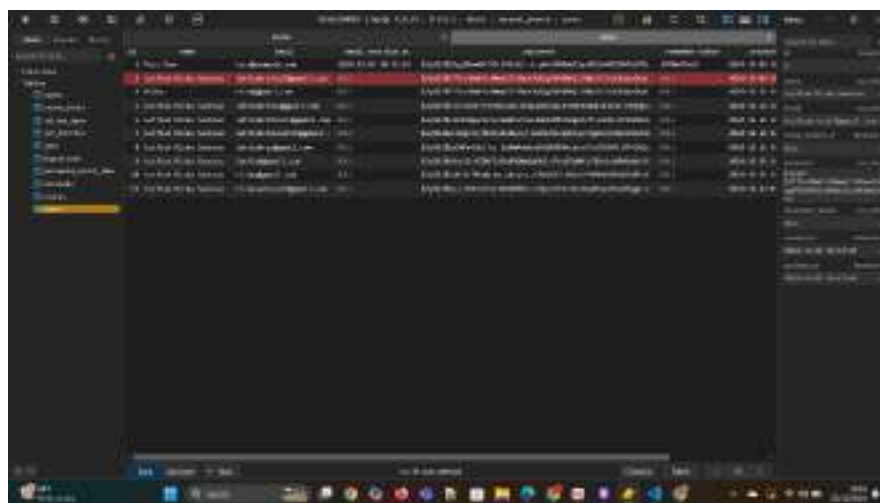


## PRAKTIKUM 12 VID 22

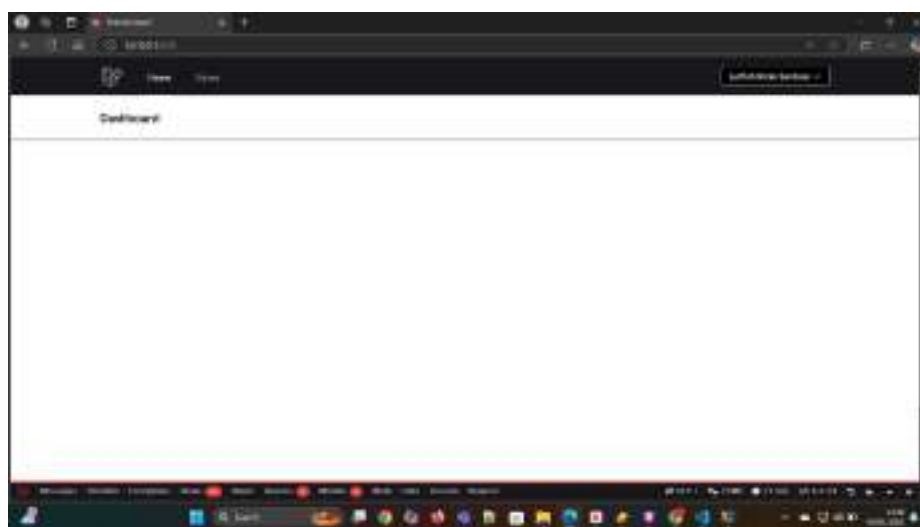


## PRAKTIKUM 13 VID 23-24





## PRAKTIKUM 14 VID 26



## PRAKTIKUM 15 Vid 27-31

