**LAPORAN TUGAS AKHIR KEAMANAN APLIKASI BESERTA PEJELASANNYA**



**DIBUAT OLEH:**

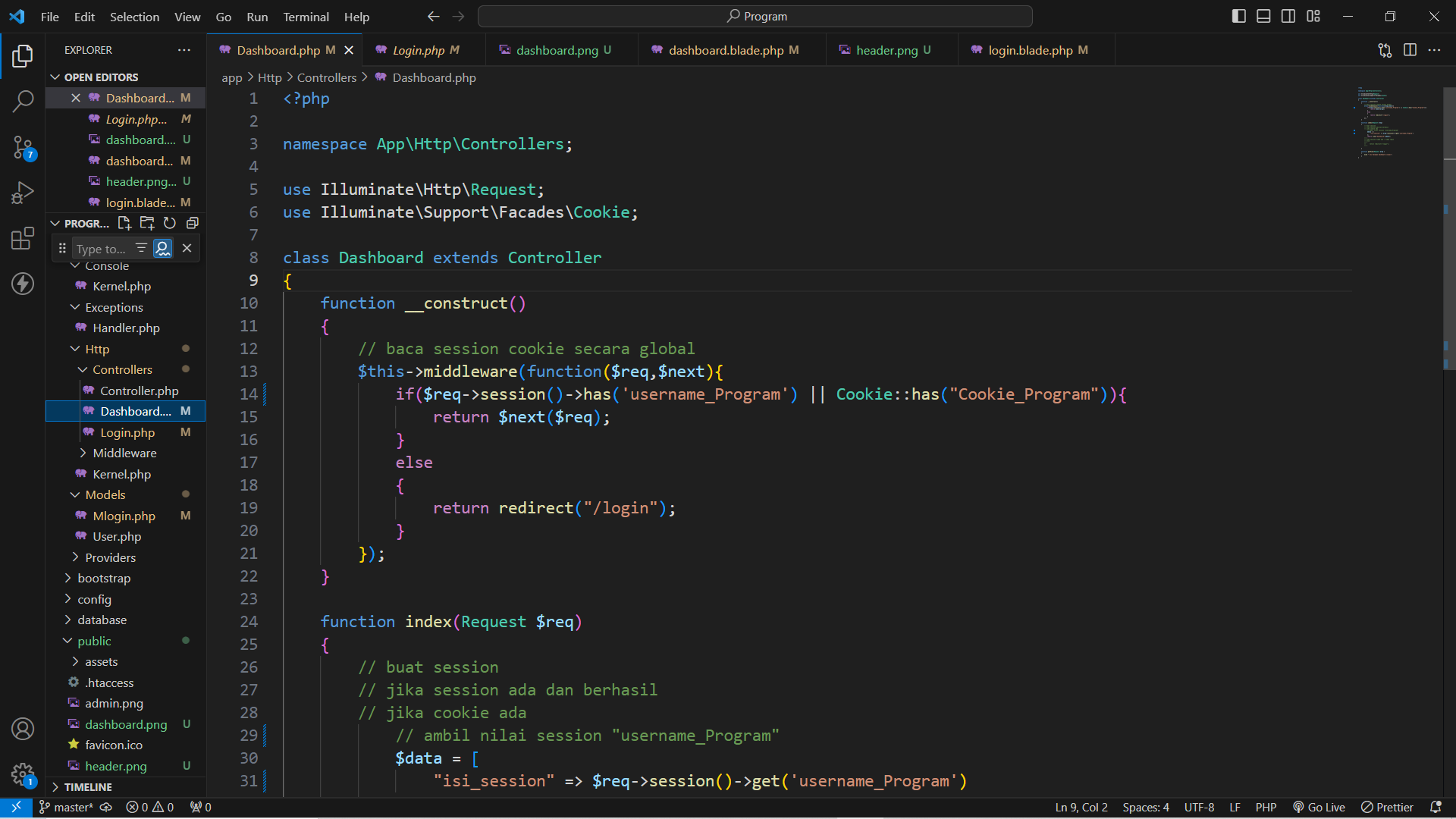
**Muhammad Tegar Tirta Laksana – 20313012**

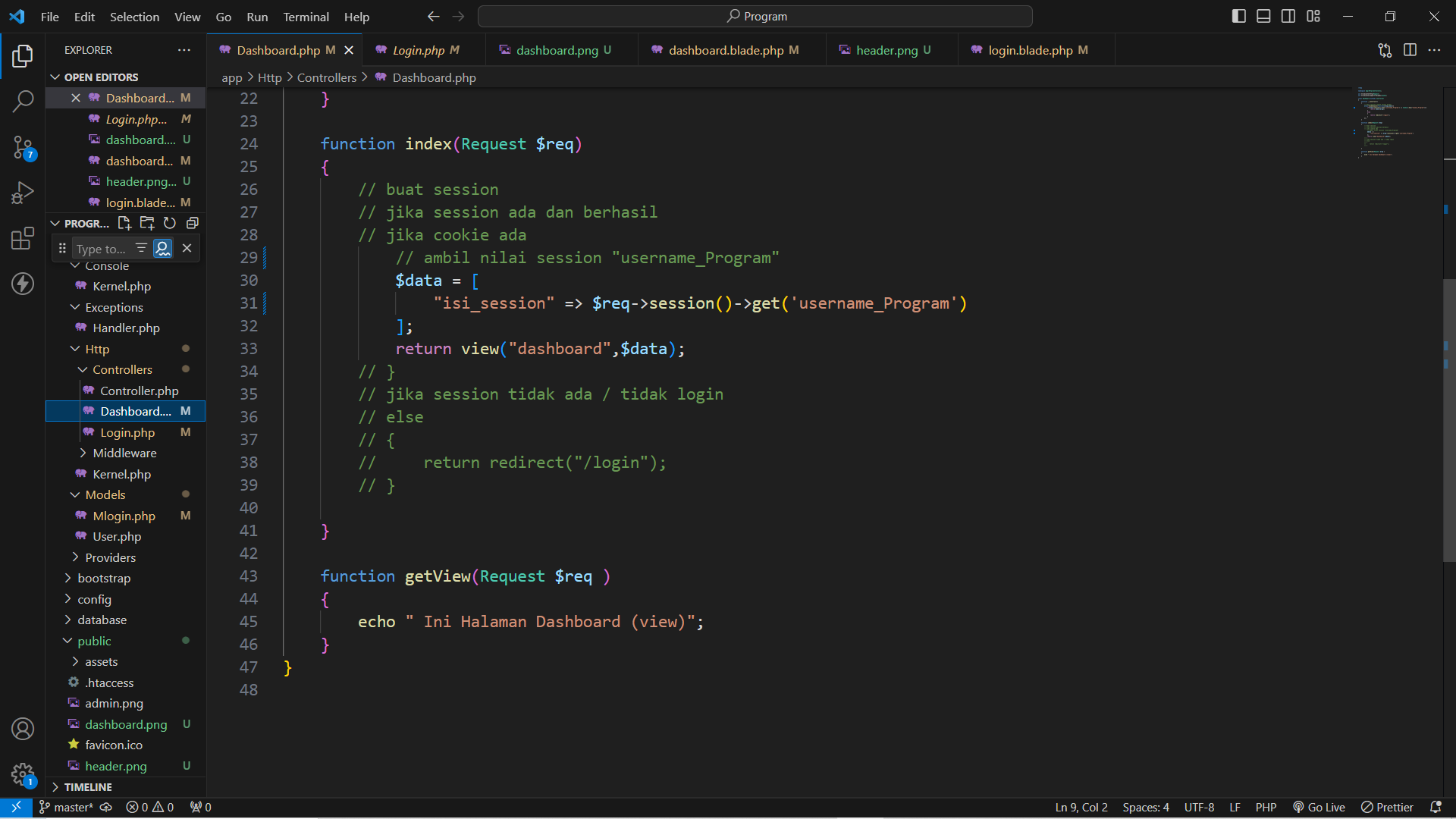
**UNIVERSITAS TEKNOKRAT INDONESIA**

**BANDAR LAMPUNG**

**2024**

1. Controller Dashboard





Middleware:

Konstruktor mengandung middleware yang memeriksa apakah session "username\_Program" ada atau cookie "Cookie\_Program" ada. Jika salah satunya ada, maka request diteruskan; jika tidak, pengguna diarahkan ke halaman login.

Fungsi index:

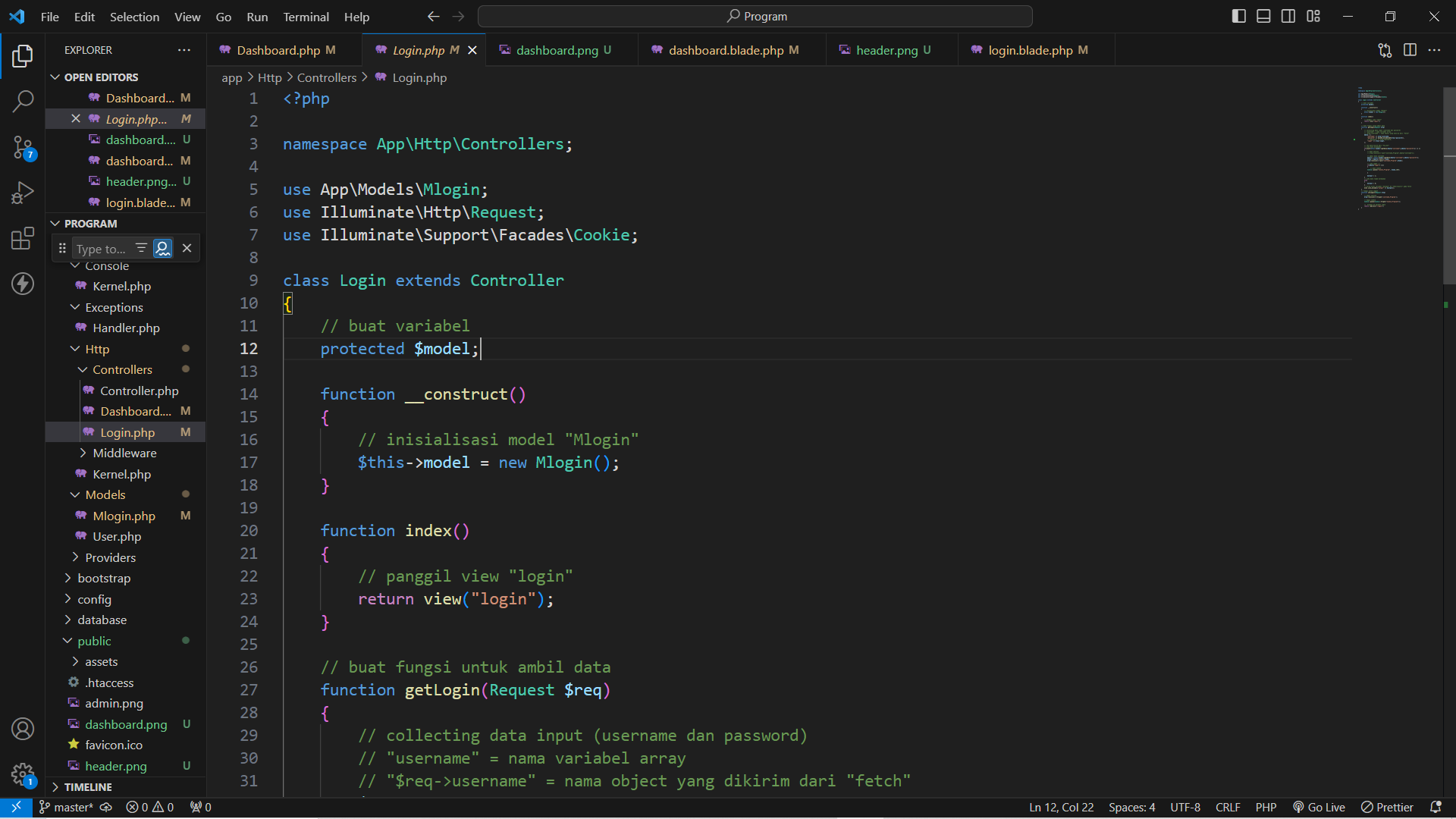
Fungsi ini memperoleh nilai dari session "username\_Program" dan melewatkan data tersebut ke view "dashboard". Data ini dapat digunakan di view untuk menampilkan informasi yang sesuai.

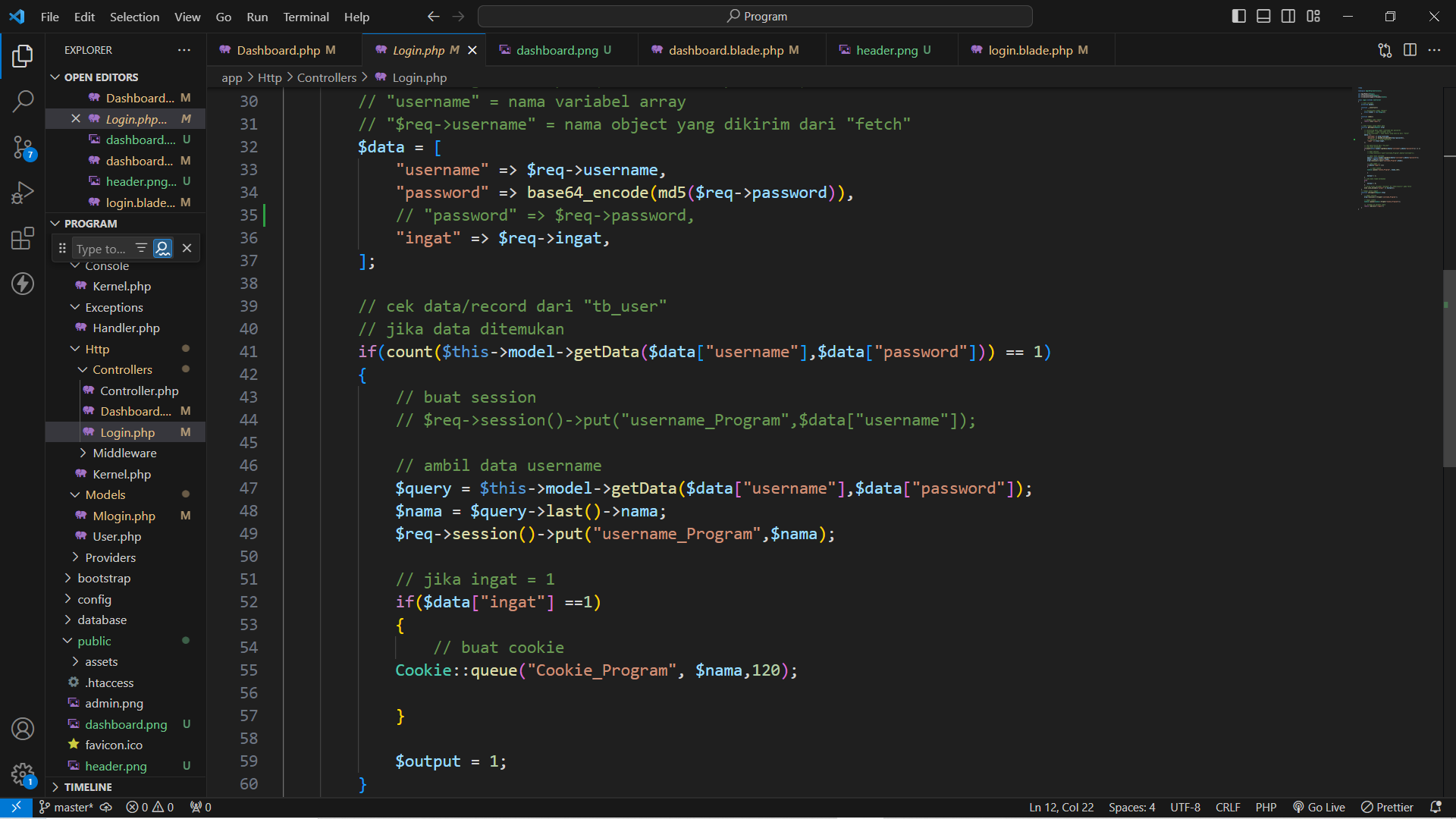
Fungsi getView:

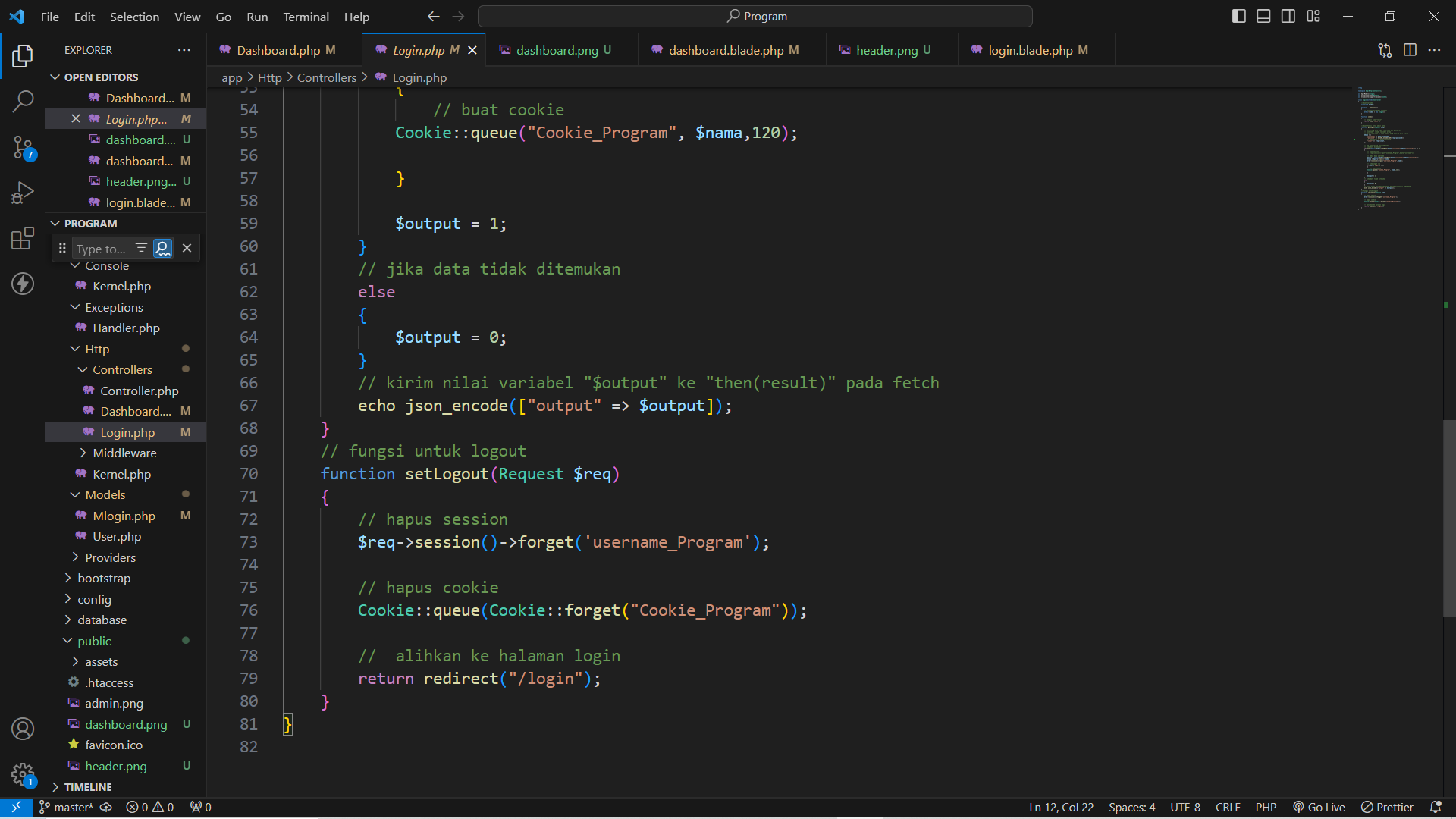
Fungsi ini hanya mencetak pesan bahwa itu adalah halaman Dashboard (view). Ini bisa diubah atau diperluas sesuai dengan kebutuhan Anda.

Semoga penjelasan ini membantu Anda memahami kode tersebut. Jika Anda memiliki pertanyaan lebih lanjut, jangan ragu untuk bertanya!

1. /serarch pada thunder client







Tentu, saya akan memberikan penjelasan yang lebih singkat:

Bagian 1: Inisialisasi Variabel dan Konstruktor

- `protected $model`: Variabel untuk menyimpan instance model "Mlogin".

- Konstruktor menginisialisasi model "Mlogin".

Bagian 2: Fungsi `index`

- Menampilkan view "login" saat endpoint diakses.

Bagian 3: Fungsi `getLogin`

- Mengelola proses login.

- Mengumpulkan data input (username dan password).

- Enkripsi password dengan `base64\_encode(md5($req->password))`.

- Memeriksa data dari model "Mlogin".

- Jika data ditemukan, membuat sesi ("username\_Program") dan, jika perlu, cookie ("Cookie\_Program").

- Mengirim hasil akhir sebagai respons JSON.

Bagian 4: Fungsi `setLogout`

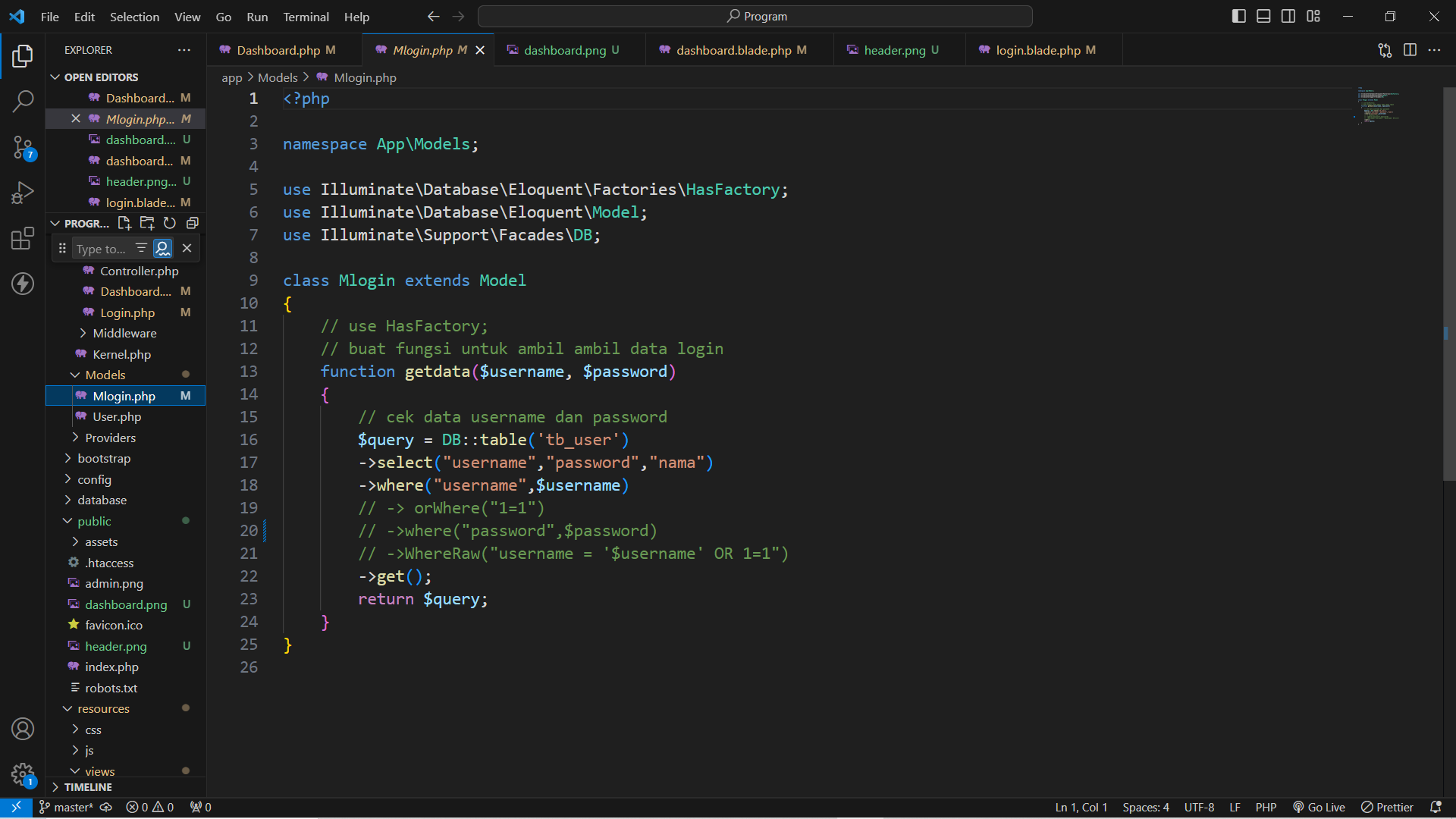
- Mengelola proses logout.

- Menghapus sesi ("username\_Program") dan cookie ("Cookie\_Program").

- Mengarahkan pengguna ke halaman login.

Semoga penjelasan yang lebih singkat ini membantu! Jika ada pertanyaan lebih lanjut, silakan ditanyakan.

1. /detail pada thunder client

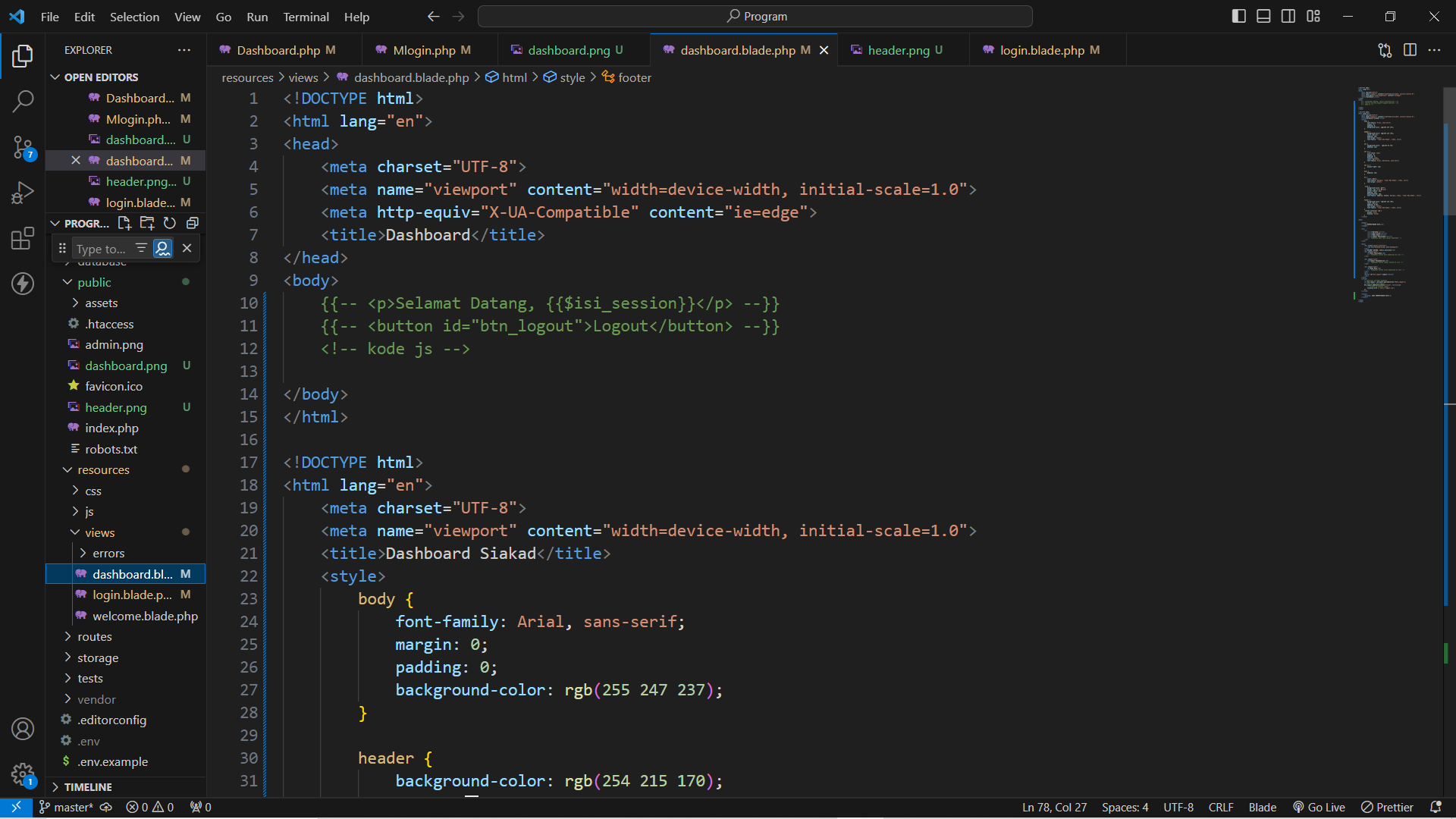
Namespace dan Class:

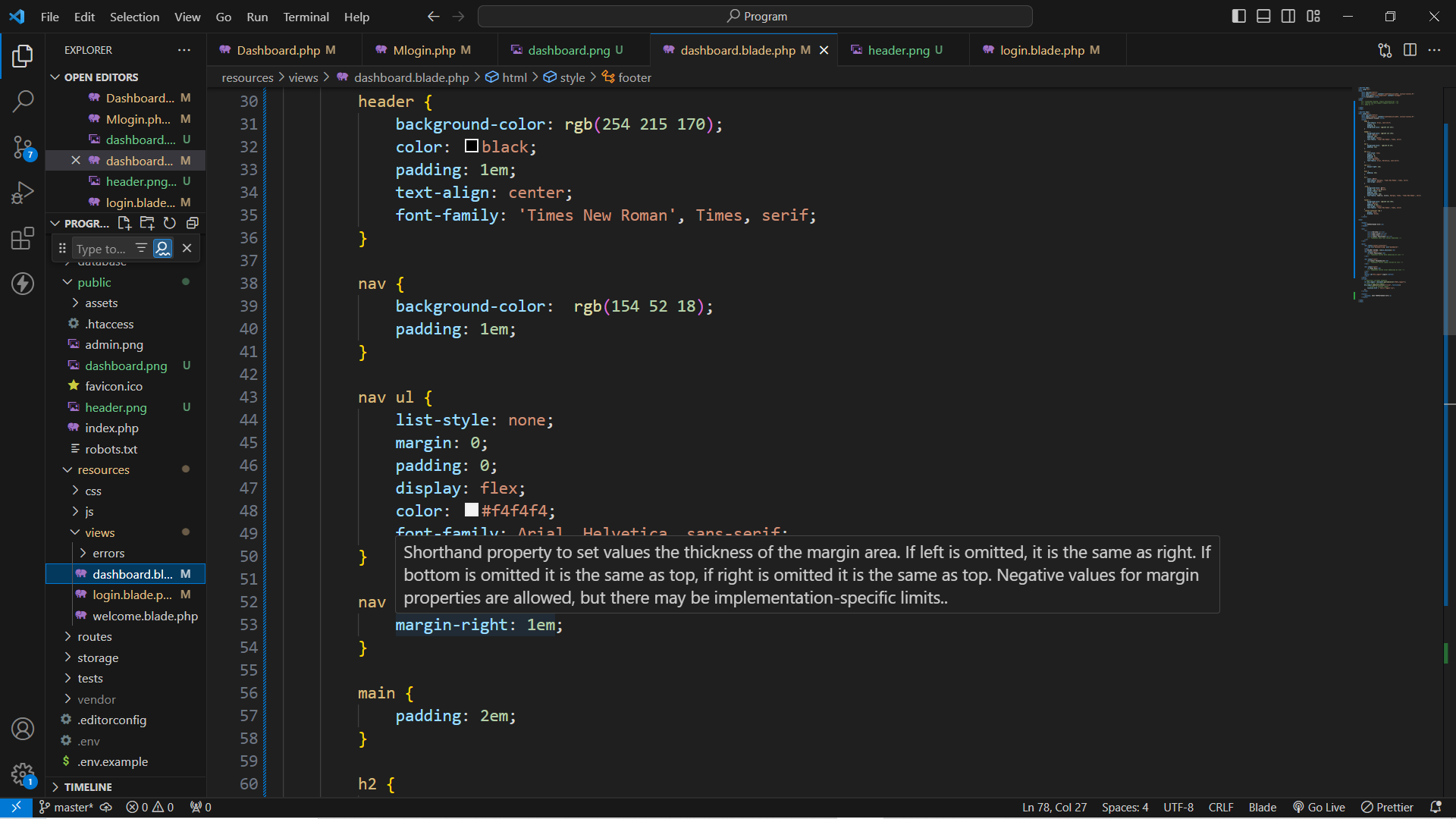
* Mendefinisikan namespace untuk model (App\Models).
* Menggunakan trait HasFactory (meskipun trait ini di-comment-out), yang umumnya digunakan untuk factory model, tetapi di sini tidak diperlukan.
* Meng-extend kelas dasar Model dari Laravel.
* Menggunakan fasad DB untuk berinteraksi dengan database.

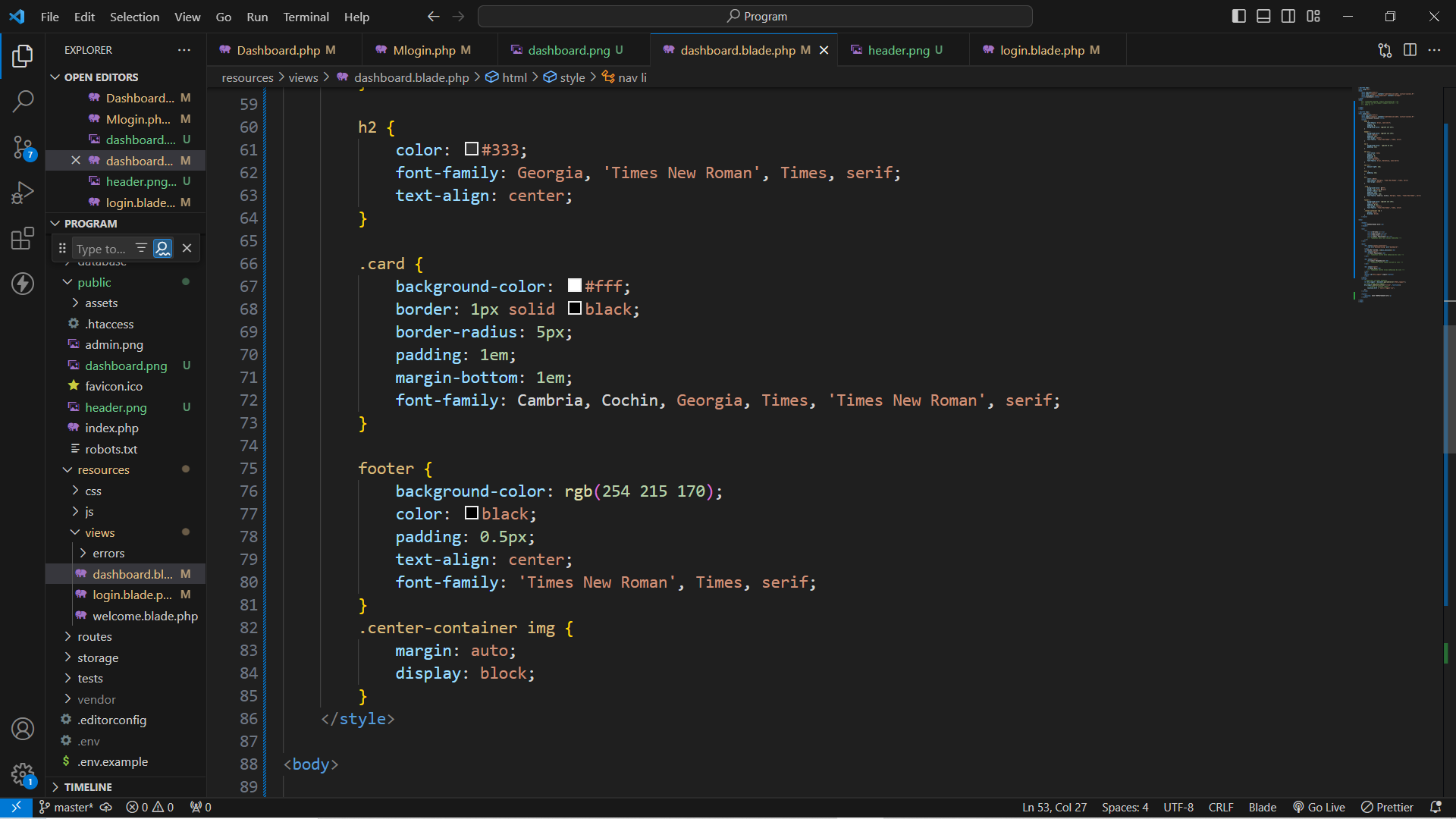
Fungsi getdata:

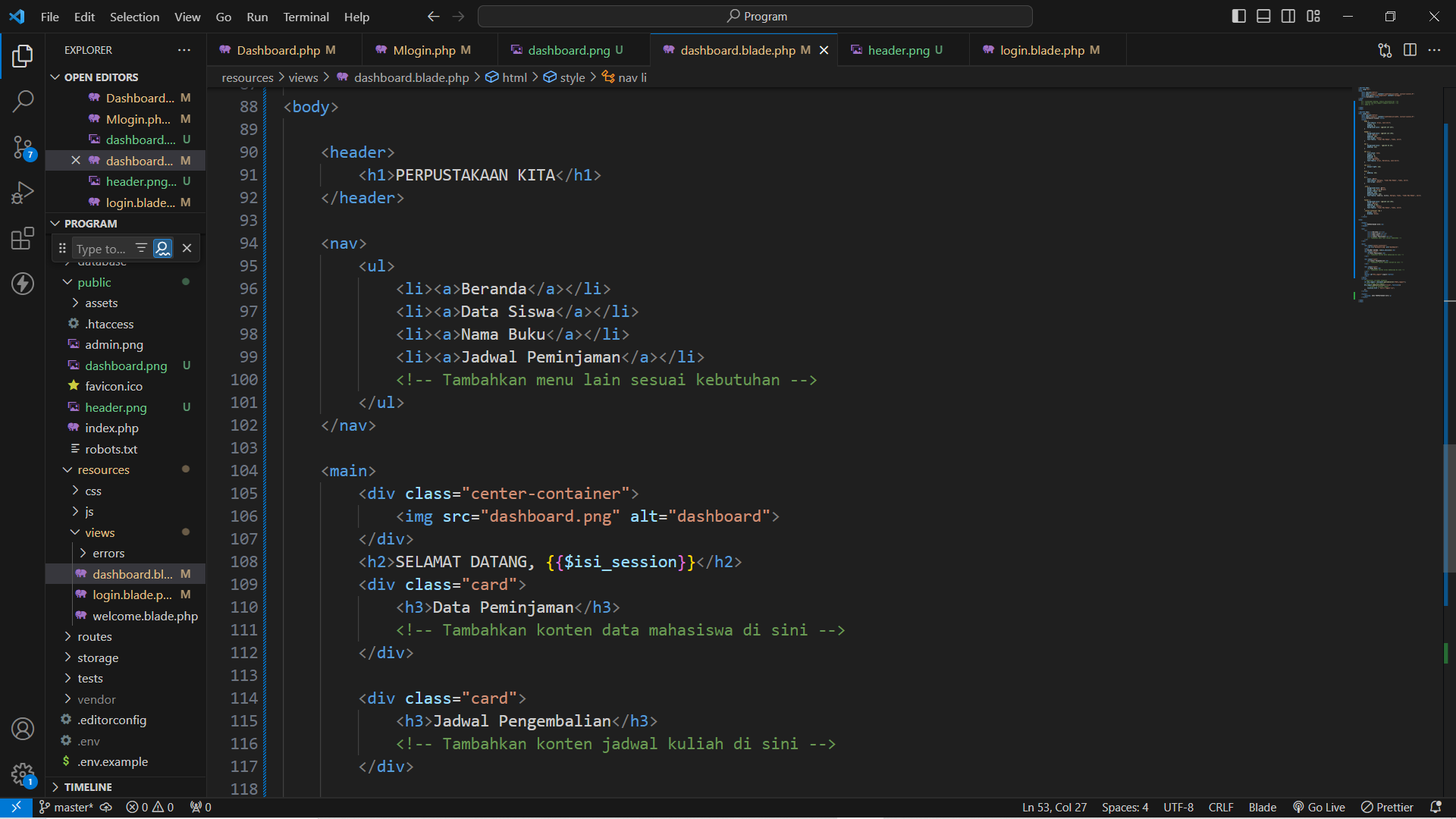
* Fungsi ini mengambil data dari tabel tb\_user berdasarkan username yang diberikan.
* Menggunakan fasad DB untuk membuat query builder.
* Select kolom username, password, dan nama.
* Melakukan kondisi where untuk mencocokkan username.
* Catatan: Ada beberapa bagian yang di-comment-out, yang mungkin merupakan percobaan untuk menangani situasi tertentu, tetapi saat ini tidak aktif.

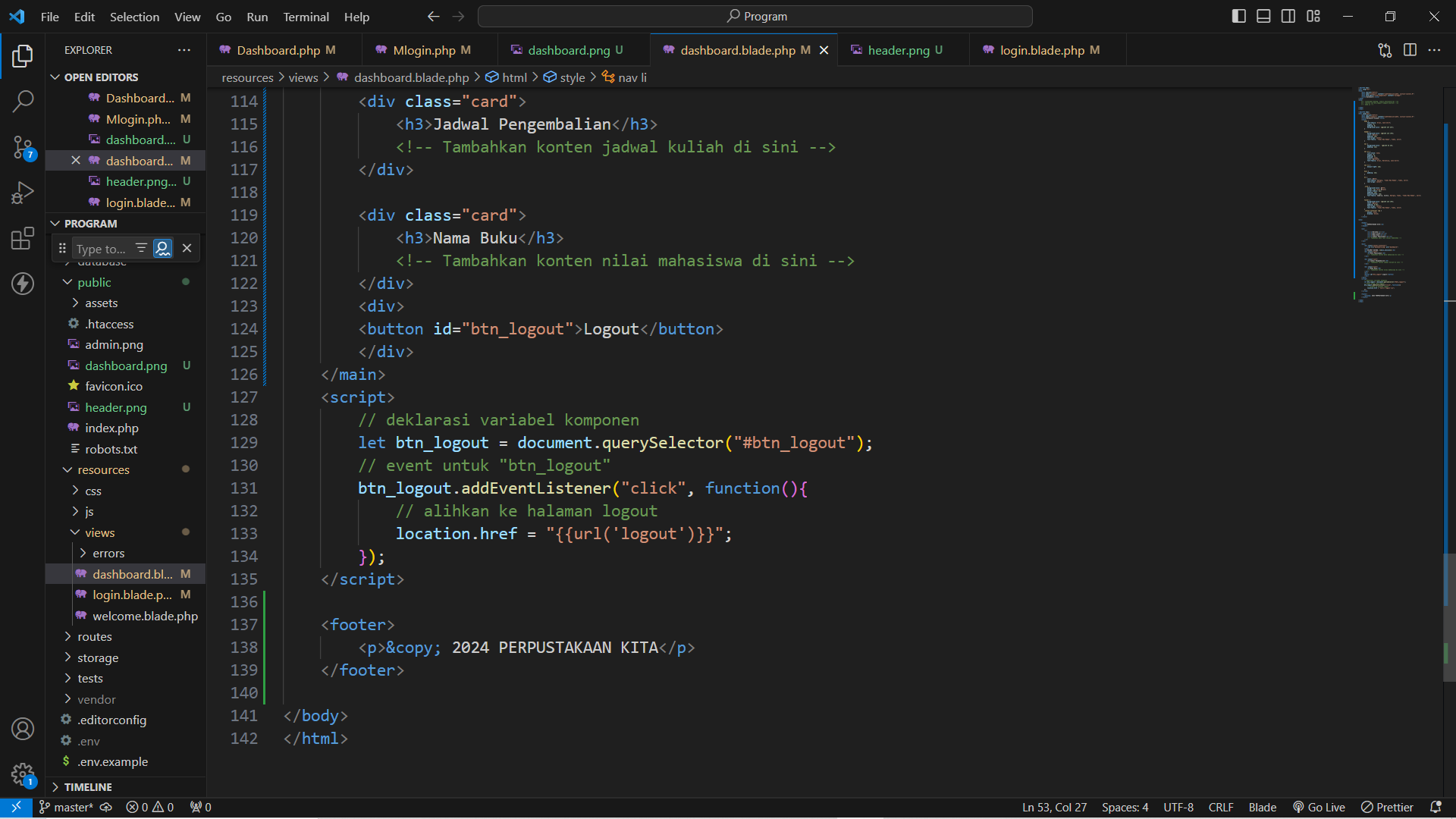
1. Dashboard.blade.php











Tentu, berikut adalah penjelasan singkat mengenai struktur dan elemen-elemen utama pada kode HTML dan CSS yang Anda berikan tanpa menyertakan codingan:

HTML:

1. Tag `<header>`:

- Berisi judul besar "PERPUSTAKAAN KITA" yang ditempatkan di tengah halaman.

2. Tag `<nav>`:

- Menyediakan area untuk menu navigasi, termasuk daftar menu seperti "Beranda", "Data Siswa", "Nama Buku", dan "Jadwal Peminjaman".

3. Tag `<main>`:

- Menyajikan konten utama halaman, termasuk gambar dashboard, sambutan "SELAMAT DATANG", dan beberapa kartu (cards) untuk informasi seperti "Data Peminjaman", "Jadwal Pengembalian", dan "Nama Buku".

4. Tag `<footer>`:

- Berisi informasi hak cipta dan tahun, misalnya "&copy; 2024 PERPUSTAKAAN KITA".

CSS:

1. Styling Umum:

- Menetapkan gaya umum untuk elemen-elemen HTML, termasuk warna latar belakang dan font-family.

2. Styling Header:

- Mengatur gaya untuk elemen header, seperti warna latar belakang, warna teks, dan padding.

3. Styling Navigation (Nav):

- Menetapkan tata letak dan tampilan untuk elemen navigasi, termasuk warna latar belakang dan warna teks.

4. Styling Main Content:

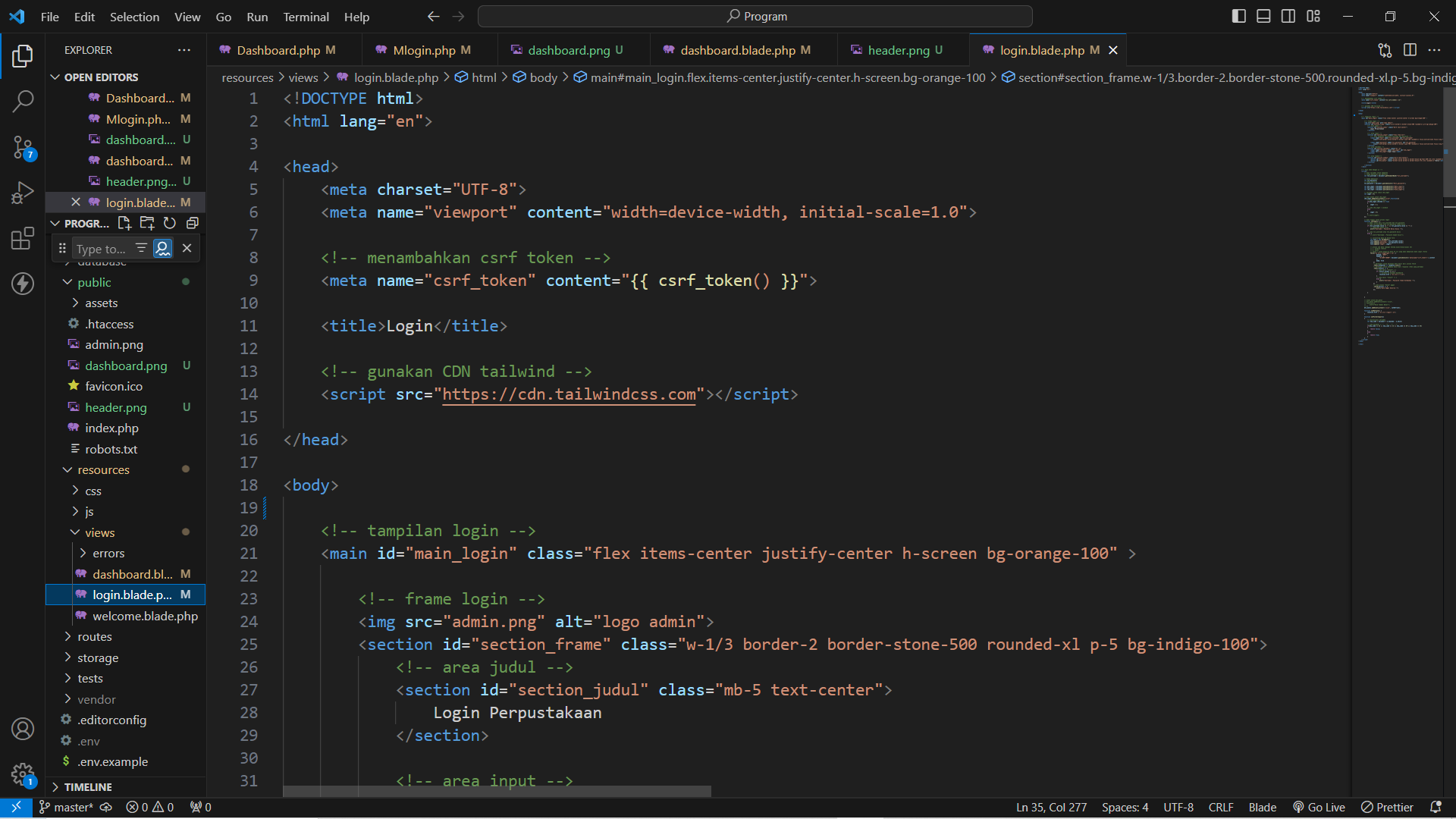
- Memberikan padding untuk elemen main dan mengatur gaya kartu (cards) yang berisi informasi peminjaman, pengembalian, dan buku.

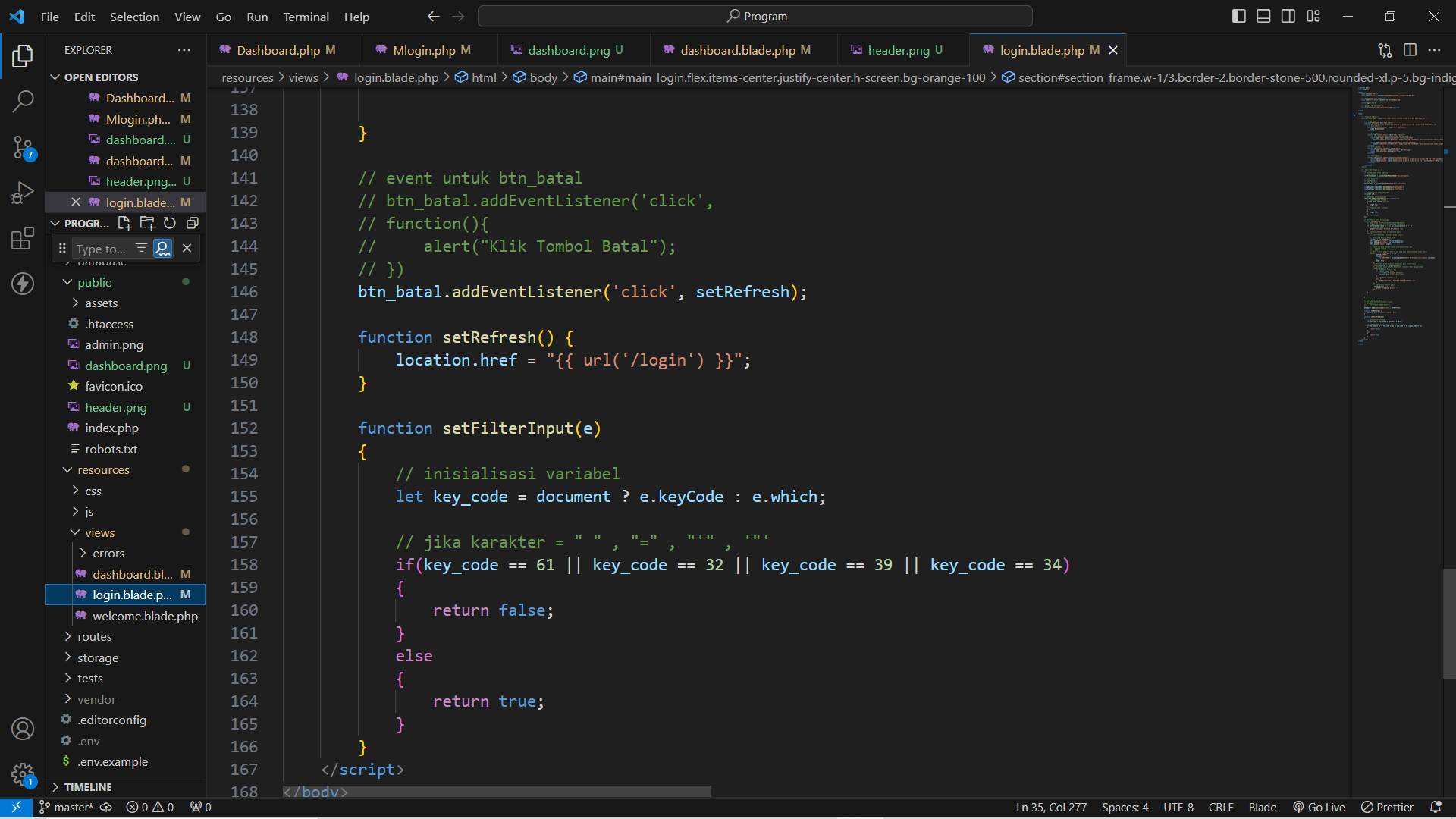
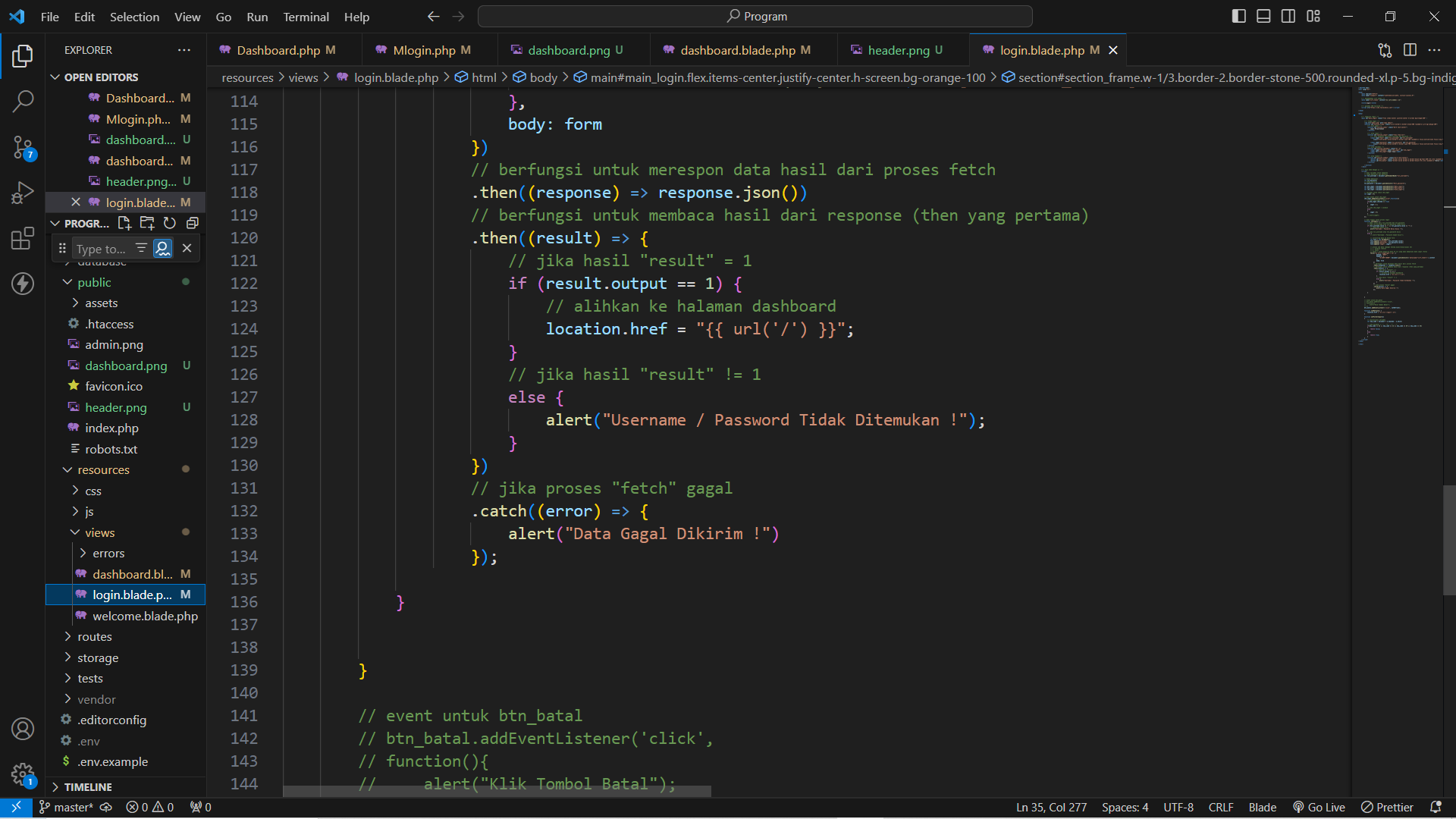
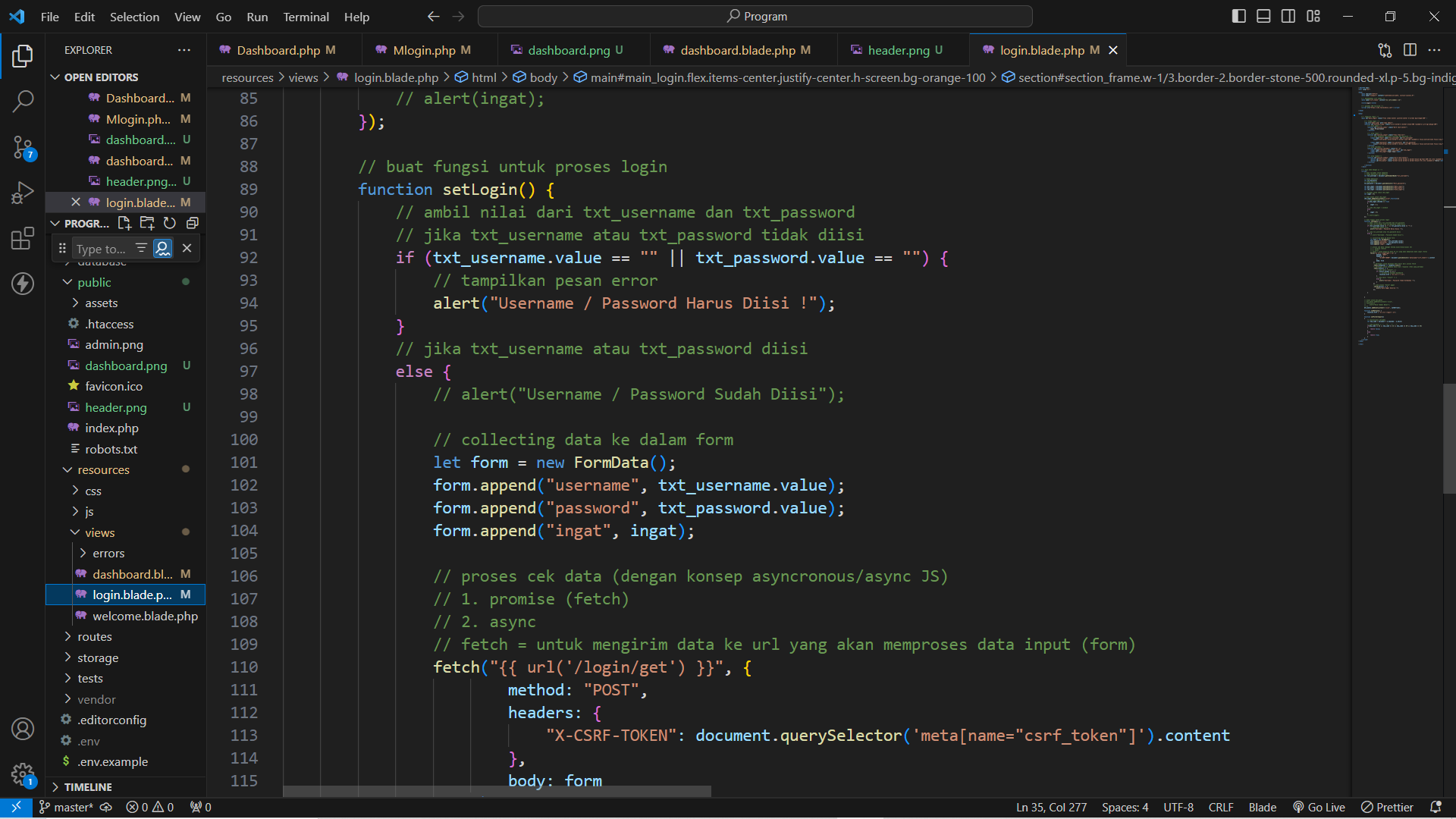
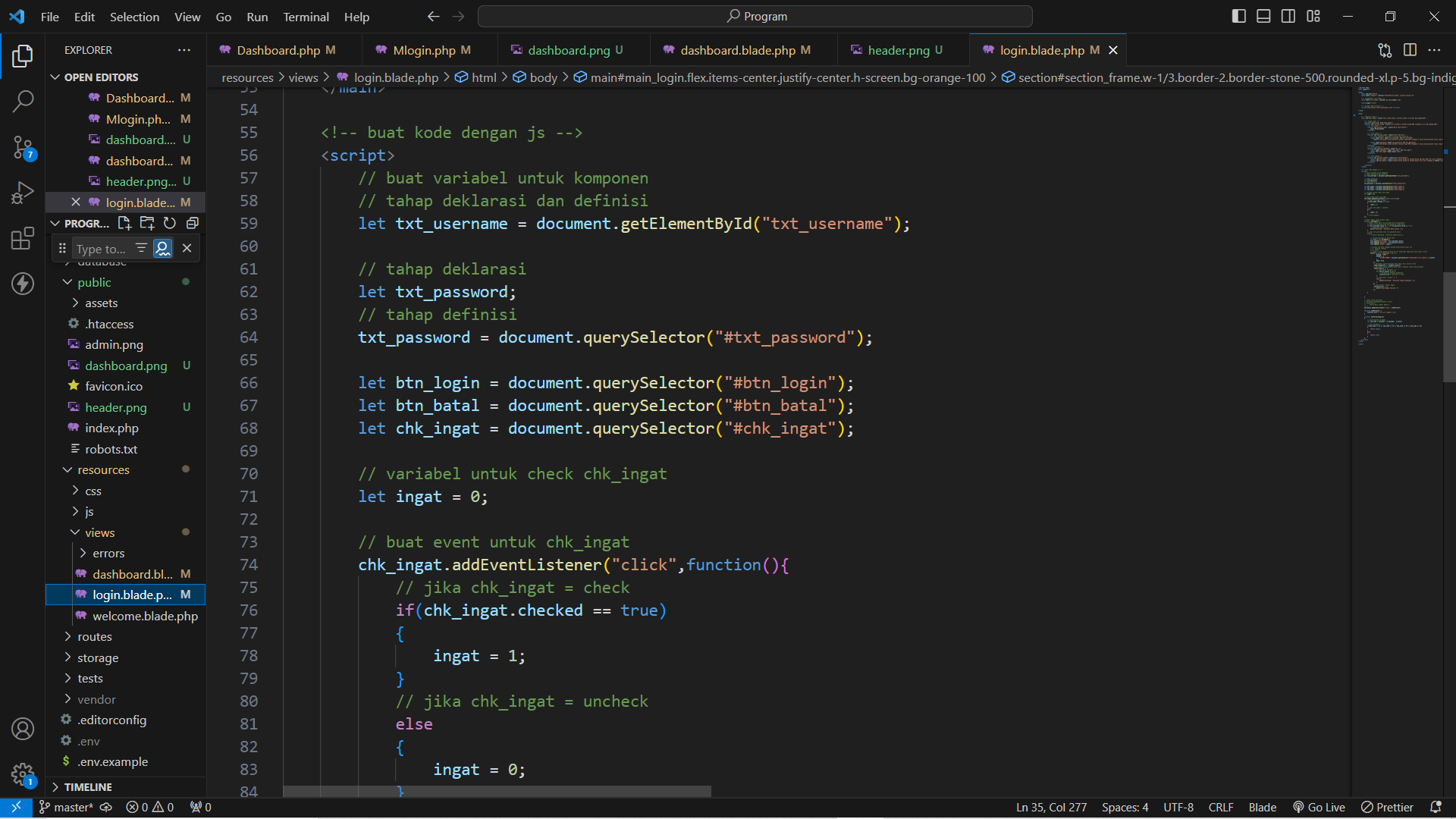
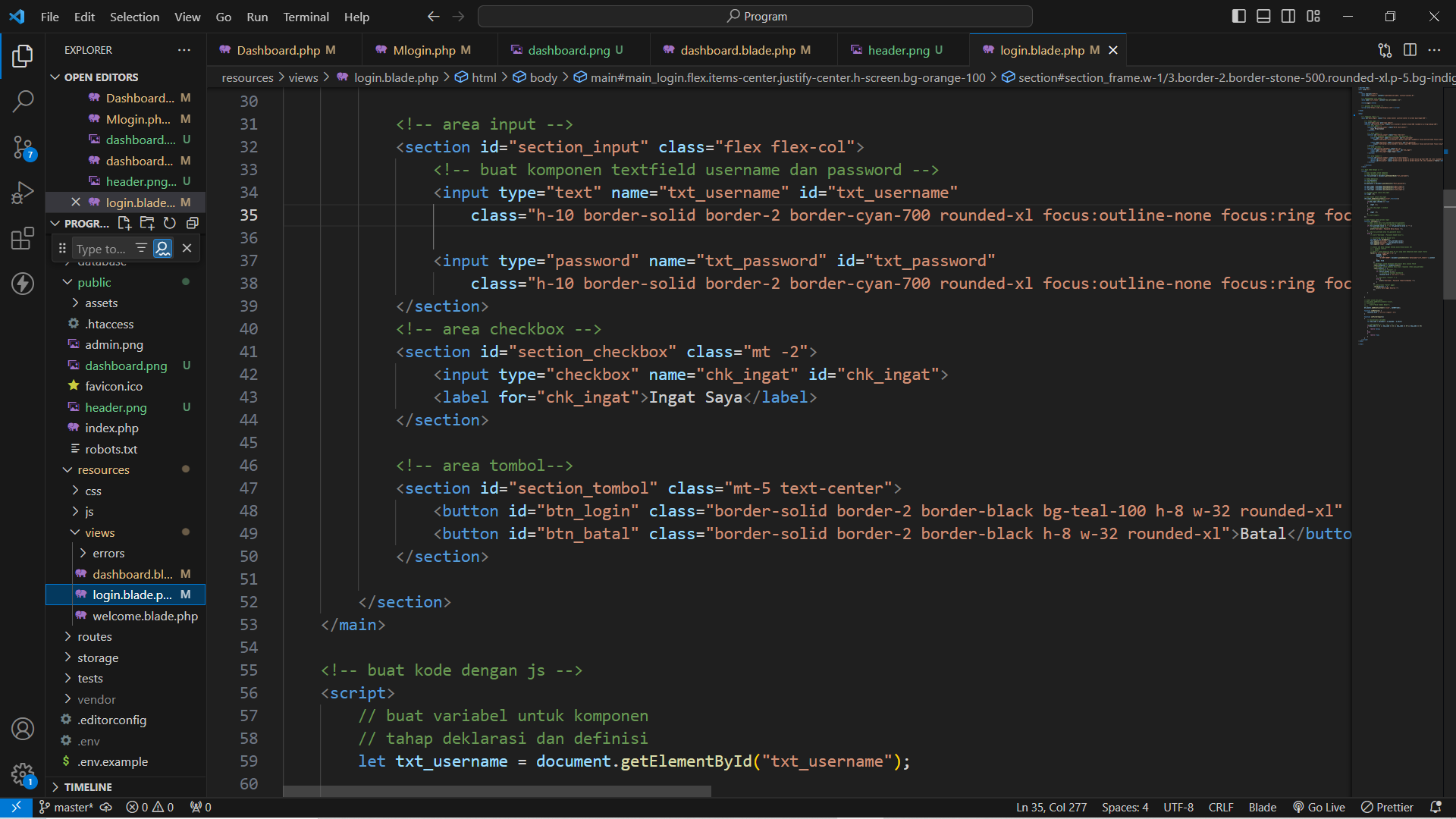
5. Styling Footer:

- Menetapkan tata letak dan tampilan untuk elemen footer, termasuk warna latar belakang dan warna teks.

Dengan kombinasi elemen-elemen ini, Anda dapat membuat tata letak halaman dashboard perpustakaan yang sederhana dan informatif.

1. Login.blade.php





Tentu, berikut adalah penjelasan singkat mengenai struktur dan fungsionalitas dari kode :

HTML:

1. Tag `<main>`:

- Menyajikan tampilan utama formulir login.

- Berisi elemen-elemen seperti gambar logo admin, judul "Login Perpustakaan", input username dan password, checkbox "Ingat Saya", dan tombol Login serta Batal.

JavaScript:

1. Deklarasi Variabel:

- Deklarasi variabel untuk elemen-elemen seperti `txt\_username`, `txt\_password`, `btn\_login`, `btn\_batal`, dan `chk\_ingat`.

2. Event Listener `chk\_ingat`:

- Membuat event listener untuk checkbox "Ingat Saya".

- Mengubah nilai variabel `ingat` berdasarkan status checkbox (tercentang atau tidak tercentang).

3. Fungsi `setLogin()`:

- Dipanggil saat tombol "Login" ditekan.

- Mengumpulkan data dari input username, password, dan status checkbox.

- Mengirim data menggunakan metode POST dengan fetch API.

- Jika respons dari server adalah 1, maka pengguna akan dialihkan ke halaman dashboard. Jika tidak, tampilkan pesan kesalahan.

4. Event Listener `btn\_batal`:

- Dipanggil saat tombol "Batal" ditekan.

- Memanggil fungsi `setRefresh()` untuk mengarahkan kembali ke halaman login.

5. Fungsi `setRefresh()`:

- Mengarahkan kembali ke halaman login.

6. Fungsi `setFilterInput(e)`:

- Mencegah karakter spasi, sama dengan, kutip satu, dan kutip ganda saat dimasukkan pada input username dan password.

Catatan:

- Terdapat penggunaan fetch API untuk mengirim data ke server secara asynchronous.

- Validasi input dilakukan dengan mengecek apakah username dan password kosong.

- Penggunaan checkbox "Ingat Saya" untuk menyimpan nilai dalam variabel `ingat`.

Dengan kombinasi elemen HTML dan fungsi JavaScript, formulir login ini memiliki beberapa fitur seperti validasi input, penggunaan checkbox, dan pengiriman data ke server.

Hasil codingan :

