

LAMPIRAN

A. Lampiran 1 Surat Kesepakatan Bimbingan Skripsi

SURAT KESEPAKATAN BIMBINGAN SKRIPSI

Kami yang bertanda tangan di bawah ini :

Pihak Pertama

Nama : Ramanda Kholisandra
NIM : 20090015
Program Studi : Teknik Informatika

Pihak Kedua

Nama : Dwi Intan Afidah, S.T., M.Kom.
Status : Dosen
NIDN : 0629109301
Jabatan Fungsional : Asisten Ahli
Pangkat/Golongan : Penata Muda Tk. 1/III

Pada hari ini Rabu tanggal 27 Maret 2024 telah terjadi sebuah kesepakatan bahwa Pihak Kedua bersedia menjadi Pembimbing I Skripsi Pihak Pertama dengan **syarat adanya kemajuan pengerjaan skripsi yang dipresentasikan setiap dua minggu**. Adapun waktu dan tempat pelaksanaan disepakati antar pihak. Demikian kesepakatan ini dibuat dengan penuh kesadaran guna kelancaran penyelesaian Skripsi

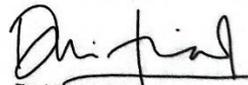
Tegal, 27 Maret 2024

Pihak Pertama



Ramanda Kholisandra

Pihak Kedua



Dwi Intan Afidah, S.T., M.Kom.

Mengetahui
Ketua Program Studi Sarjana Terapan Teknik Informatika



Dyah Aprilia, S.T., M.Kom.
NIPY 09.01.2025

SURAT KESEPAKATAN BIMBINGAN SKRIPSI

Kami yang bertanda tangan di bawah ini :

Pihak Pertama

Nama : Ramanda Kholisandra
NIM : 20090015
Program Studi : Teknik Informatika

Pihak Kedua

Nama : Ardi Susanto, S.Kom., M.Cs.
Status : Dosen
NIDN : 0629109301
Jabatan Fungsional : Asisten Ahli
Pangkat/Golongan : Penata Muda Tk. 1 – III/b

Pada hari ini Rabu tanggal 27 Maret 2024 telah terjadi sebuah kesepakatan bahwa Pihak Kedua bersedia menjadi Pembimbing II Skripsi Pihak Pertama dengan syarat Pihak Pertama wajib melakukan bimbingan Skripsi minimal 8 kali kepada Pihak Kedua. Adapun waktu dan tempat pelaksanaan disepakati antar pihak. Demikian kesepakatan ini dibuat dengan penuh kesadaran guna kelancaran penyelesaian Skripsi

Tegal, 27 Maret 2024

Pihak Pertama



Ramanda Kholisandra

Pihak Kedua



Ardi Susanto, S.Kom., M.Cs.

Mengetahui
Ketua Program Studi Sarjana Terapan Teknik Informatika



Dyah Apriliana, S.Kom.
NIPY 09.015.225

B. Lampiran 2 Surat Pernyataan Pengajuan HKI

SURAT PERNYATAAN

Yang bertanda tangan di bawah ini, pemegang hak cipta:

1. Nama : Ramanda Kholisandra
Kewarganegaraan : Indonesia
Alamat : Jalan Ahmad Yani Utara Gang Gianti Timur 1 RT.03/RW.014 Kecamatan Pemalang, Kabupaten Pemalang, Provinsi Jawa Tengah, 5213
2. Nama : Dwi Intan Af'idah, S.T., M.Kom.
Kewarganegaraan : Indonesia
Alamat : Desa Grinting RT.003/RW.001, Kecamatan Bulakamba, Kabupaten Brebes, Provinsi Jawa Tengah
2. Nama : Ardi Susanto, S.Kom., M.Cs.
Kewarganegaraan : Indonesia
Alamat : Margadana, Tegal

Dengan ini menyatakan bahwa:

1. Karya Cipta yang saya mohonkan :
Berupa : Program Komputer
Berjudul : Aplikasi Rekomendasi Film Asia Berdasarkan Sentimen Analisis Menggunakan Collaborative Filtering Berbasis Website
 - Tidak meniru dan tidak sama secara esensial dengan Karya Cipta milik pihak lain atau obyek kekayaan intelektual lainnya sebagaimana dimaksud dalam Pasal 68 ayat (2);
 - Bukan merupakan Ekspresi Budaya Tradisional sebagaimana dimaksud dalam Pasal 38;
 - Bukan merupakan Ciptaan yang tidak diketahui penciptanya sebagaimana dimaksud dalam Pasal 39;
 - Bukan merupakan hasil karya yang tidak dilindungi Hak Cipta sebagaimana dimaksud dalam Pasal 41 dan 42;
 - Bukan merupakan Ciptaan seni lukis yang berupa logo atau tanda pembeda yang digunakan sebagai merek dalam perdagangan barang/jasa atau digunakan sebagai lambang organisasi, badan usaha, atau badan hukum sebagaimana dimaksud dalam Pasal 65 dan;
 - Bukan merupakan Ciptaan yang melanggar norma agama, norma susila, ketertiban umum, pertahanan dan keamanan negara atau melanggar peraturan perundang-undangan sebagaimana dimaksud dalam Pasal 74 ayat (1) huruf d Undang-Undang Nomor 28 Tahun 2014 tentang Hak Cipta
2. Sebagai pemohon mempunyai kewajiban untuk menyimpan asli contoh ciptaan yang dimohonkan dan harus memberikan apabila dibutuhkan untuk kepentingan penyelesaian sengketa perdata maupun pidana sesuai dengan ketentuan perundang-undangan.
3. Karya Cipta yang saya mohonkan pada Angka 1 tersebut di atas tidak pernah dan tidak sedang dalam sengketa pidana dan/atau perdata di Pengadilan
4. Dalam hal ketentuan sebagaimana dimaksud dalam Angka 1 dan Angka 3 tersebut di atas saya / kami langgar, maka saya / kami bersedia secara sukarela bahwa:
 - a. permohonan karya cipta yang saya ajukan dianggap ditarik kembali; atau

- b. Karya Cipta yang telah terdaftar dalam Daftar Umum Ciptaan Direktorat Hak Cipta, Direktorat Jenderal Hak Kekayaan Intelektual, Kementerian Hukum Dan Hak Asasi Manusia R.I dihapuskan sesuai dengan ketentuan perundang-undangan yang berlaku.
- c. Dalam hal kepemilikan Hak Cipta yang dimohonkan secara elektronik sedang dalam berperkara dan/atau sedang dalam gugatan di Pengadilan maka status kepemilikan surat pencatatan elektronik tersebut ditangguhkan menunggu putusan Pengadilan yang berkekuatan hukum tetap.

Demikian Surat pernyataan ini kami buat dengan sebenarnya dan untuk dipergunakan sebagaimana mestinya.

Tegal, Juli 2024



(Ramanda Kholisandra)

A handwritten signature in black ink, appearing to read 'Dwi Intan Af'idah'.

(Dwi Intan Af'idah, S.T., M.Kom.)

A handwritten signature in black ink, appearing to read 'Ardi Susanto'.

(Ardi Susanto, S.Kom., M.Cs.)

C. Lampiran 3 Surat Pengalihan HKI

SURAT PENGALIHAN HAK CIPTA

Yang bertanda tangan di bawah ini:

1. Nama : Ramanda Kholisandra
Kewarganegaraan : Indonesia
Alamat : Jalan Ahmad Yani Utara Gang Gianti Timur 1 RT.03/RW.014 Kecamatan Pemalang, Kabupaten Pemalang, Provinsi Jawa Tengah, 5213
2. Nama : Dwi Intan Af'idah, S.T., M.Kom.
Kewarganegaraan : Indonesia
Alamat : Desa Grinting RT.003/RW.001, Kecamatan Bulakamba, Kabupaten Brebes, Provinsi Jawa Tengah
2. Nama : Ardi Susanto, S.Kom., M.Cs.
Kewarganegaraan : Indonesia
Alamat : Margadana, Tegal

Adalah **Pihak I** selaku pencipta, dengan ini menyerahkan karya ciptaan saya kepada:

Nama : Pusat Penelitian dan Pengabdian Masyarakat (P3M)
Alamat : Jl. Mataram No. 9 Pesurungan Lor Kota Tegal

Adalah **Pihak II** selaku Pemegang Hak Cipta berupa Program Komputer dengan judul "Aplikasi Rekomendasi Film Asia Berdasarkan Sentimen Analisis Menggunakan Collaborative Filtering Berbasis Website". untuk didaftarkan di Direktorat Hak Cipta dan Desain Industri, Direktorat Jenderal Kekayaan Intelektual, Kementerian Hukum dan Hak Asasi Manusia Republik Indonesia.

Demikian Surat pernyataan ini kami buat dengan sebenarnya dan untuk dipergunakan sebagaimana mestinya.

Pemegang Hak Cipta
Ketua P3M



(Dr. Aldi Budi Riyanta, S.Si., M.T)

Tegal, Juli 2024

Pencipta



(Ramanda Kholisandra)



(Dwi Intan Af'idah, S.T., M.Kom.)



(Ardi Susanto, S.Kom., M.Cs.)

D. Lampiran 4 Manual Book dan Dokumen Teknikal

MANUAL BOOK



Aplikasi Rekomendasi Film Asia Berdasarkan Sentimen Analisis
Menggunakan Collaborative Filtering Berbasis Website.

Oleh :

Ramanda Kholisandra

Dwi Intan Af'idah, S.Kom., M.Kom.

Ardi Susanto, S.Kom., M.Cs.

DAFTAR ISI

| | |
|------------------------------------|----|
| DAFTAR ISI..... | 2 |
| DAFTAR GAMBAR..... | 3 |
| 1. PENDAHULUAN..... | 4 |
| 1.1. Tujuan Pembuatan Dokumen..... | 4 |
| 1.2. Deskripsi Umum Aplikasi..... | 4 |
| 1.3. Deskripsi Dokumen..... | 4 |
| 2. PERANGKAT YANG DIBUTUHKAN..... | 4 |
| 2.1. Perangkat Lunak..... | 4 |
| 3. MENU DAN CARA PENGGUNAAN..... | 5 |
| 3.1. Struktur Menu..... | 5 |
| 3.2. Cara Penggunaan..... | 7 |
| User | 7 |
| 1. Halaman Login..... | 7 |
| 2. Halaman Lupa Password..... | 8 |
| 3. Halaman Reset Password..... | 9 |
| 4. Halaman Registrasi..... | 10 |
| 5. Halaman Verifikasi OTP..... | 11 |
| 6. Halaman Pilih Genre..... | 12 |
| 7. Halaman Home..... | 13 |
| 8. Halaman Detail..... | 14 |
| 9. Halaman Profil Pengguna..... | 15 |
| 10. Halaman Bookmark..... | 16 |
| 11. Halaman Review..... | 16 |
| Admin | 17 |
| 1. Halaman Login..... | 17 |
| 2. Halaman Dashboard..... | 17 |
| 3. Halaman Genres..... | 18 |
| 4. Halaman Movies..... | 18 |
| 5. Halaman Reviews..... | 19 |
| 6. Halaman Export Dataset..... | 20 |
| 7. Halaman Train Model..... | 21 |
| 8. Halaman User..... | 21 |

DAFTAR GAMBAR

| | |
|---|----|
| Gambar 1. Tampilan Halaman Login..... | 8 |
| Gambar 2. Tampilan Halaman Lupa Password..... | 9 |
| Gambar 3. Tampilan Halaman Reset Password..... | 10 |
| Gambar 4. Tampilan Halaman Registrasi..... | 11 |
| Gambar 5. Tampilan Halaman Verifikasi Email..... | 12 |
| Gambar 6. Tampilan Halaman Pilih Genre..... | 13 |
| Gambar 7. Tampilan Halaman Home..... | 14 |
| Gambar 8. Halaman Detail..... | 15 |
| Gambar 9. Halaman Profile..... | 15 |
| Gambar 10. Tampilan Halaman Bookmark..... | 16 |
| Gambar 11. Tampilan Halaman Review..... | 16 |
| Gambar 12. Tampilan Halaman Login Admin..... | 17 |
| Gambar 13. Tampilan Halaman Dashboard..... | 18 |
| Gambar 14. Tampilan Halaman Genres Admin..... | 18 |
| Gambar 15. Tampilan Halaman Movies Admin..... | 19 |
| Gambar 16. Tampilan Halaman Reviews Admin..... | 20 |
| Gambar 18. Tampilan Halaman Export Dataset Admin..... | 21 |
| Gambar 19. Tampilan Halaman Training Model..... | 21 |
| Gambar 20. Tampilan Halaman User Admin..... | 22 |

1. PENDAHULUAN

1.1. Tujuan Pembuatan Dokumen

Dokumen *user manual* ini dibuat untuk memberikan penjelasan dan gambaran atau sebagai panduan kepada pengguna mengenai penggunaan Aplikasi Rekomendasi Film Asia Berdasarkan Sentimen Analisis Menggunakan Collaborative Filtering Berbasis Website.

1.2. Deskripsi Umum Aplikasi

Aplikasi ini dirancang untuk memberikan sebuah rekomendasi film Asia berdasarkan kesamaan spesifikasi seorang *user* dengan *user* lainnya terhadap suatu film dan dengan mengimplementasikan sentiment analisis untuk pemberian rekomendasinya.

1.3. Deskripsi Dokumen

Dokumen ini dibuat untuk memberikan panduan penggunaan Aplikasi Rekomendasi Film Asia

1. Bagian 1

Bagian ini berisi informasi umum yang meliputi tujuan pembuatan dokumen, deskripsi umum sistem, dan deskripsi dokumen.

2. Bagian 2

Bagian ini berisi perangkat yang dibutuhkan untuk penggunaan aplikasi meliputi perangkat lunak.

3. Bagian 3

Bagian ini berisi panduan atau cara penggunaan.

2. PERANGKAT YANG DIBUTUHKAN

2.1. Perangkat Lunak

Untuk menjalankan aplikasi ini, berikut beberapa perangkat lunak yang dibutuhkan yaitu:

1. Sistem Operasi Android. Aplikasi ini harus dijalankan pada perangkat yang menggunakan sistem operasi Android.
2. Koneksi Internet. Untuk menjalankan aplikasi dengan lancar, diperlukan koneksi internet yang stabil.

3. Penyimpanan Data.

3. MENU DAN CARA PENGGUNAAN

3.1. Struktur Menu

Ada 2 pengguna dalam aplikasi Filmvibes yaitu *admin* dan *user*. Berikut struktur menu yang ada pada aplikasi Filmvibes adalah sebagai berikut:

User

1. Register

Menu ini digunakan untuk mendaftarkan akun baru. Pengguna harus mengisi informasi seperti nama, negara, provinsi, email, dan *password* untuk membuat akun.

2. Login

Menu ini digunakan untuk masuk ke aplikasi dengan akun yang sudah terdaftar. Pengguna harus memasukkan email dan kata sandi yang telah didaftarkan.

3. Verifikasi OTP

Menu ini berfungsi untuk menginputkan kode OTP agar akun *user* dapat terverifikasikan.

4. Lupa Password

Menu ini berfungsi bagi *user* yang lupa *password* dari akun mereka. Disini *user* akan disuruh memasukan email mereka yang nantinya akan ada email yang masuk ke email *user* yang berisi link untuk me reset *password* sang *user*.

5. Reset Password

Menu yang berfungsi untuk me reset *password user* dan menggantinya ke *password* baru.

6. Pilih Genre

Menu ini berfungsi untuk media rekomendasi bagi *user* baru. Dimana *user* akan disuruh memilih 5 genre film kesukaannya.

7. Home

Menu ini akan menampilkan rekomendasi film berdasarkan kesamaan referensi tontonan dengan *user* lain.

8. Detail

Menu ini menampilkan detail dari film seperti sinopsis film, rating film, judul film, dan *link streaming* film (opsional).

9. Profile

Menu ini digunakan untuk melihat dan mengedit informasi profil mereka, seperti nama, email, dan mengganti foto profile akun.

10. Bookmark

Menu yang menampilkan beberapa film rekomendasi yang sudah disimpan *user*.

11. Review

Menu ini berfungsi untuk menilai film rekomendasi dengan mengisi komentar, judul komentar, dan rating. Disini juga akan menampilkan komentar *user* lainnya.

Admin

1. Login

Menu ini digunakan untuk masuk ke dashboard dengan akun yang sudah terdaftar sebagai admin. Admin harus memasukkan email dan kata sandi yang telah didaftarkan.

2. Dashboard

Halaman ini merupakan halaman pembuka untuk halaman dashboard admin yang dimana disini akan menampilkan nama user yang login ke dashboard.

3. Menu Genres

Pada halaman ini akan menampilkan semua data genre yang ditampilkan pada halaman pilih genre saat user baru registrasi dan genre pada film

4. Menu Movies

Pada halaman ini akan menampilkan semua data genre yang ditampilkan pada halaman pilih genre saat user baru registrasi dan genre pada film.

5. Menu Reviews

Pada halaman ini akan menampilkan semua ulasan yang ditulis oleh *user* terhadap film yang diulasnya. Halaman juga akan menampilkan statistik film yang banyak diulas serta rata-rata rating yang diberikan oleh *user*.

6. Menu Export Dataset

Halaman ini akan menampilkan form untuk membuat jadwal pengekspor data menjadi dataset. Data akan dipilih apakah itu data film, ulasan, ataupun rating.

7. Menu Train Mode

Pada halaman ini akan menunjukkan form yang digunakan untuk melakukan pelatihan model rekomendasi dan model sentimen analisis.

8. Menu User

Pada halaman akan menampilkan form yang digunakan untuk mengupdate password akun administrator.

3.2. Cara Penggunaan.

User

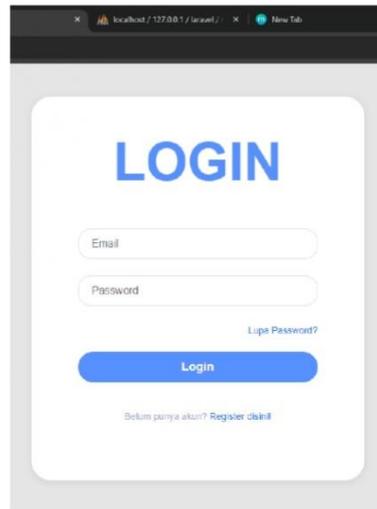
1. Halaman Login

Dihalaman login ini, anda dapat masuk kedalam aplikasi dengan memasukan *email* dan *password* yang sudah terdaftar dan terverifikasi. Tampilan halaman logion dapat dilihat pada Gambar 1. Berikut langkah-langkah nya:

- 1) Masukkan *email* dan *password*. Pastikan sudah terdaftar dan terverifikasi.
- 2) Klik button Login.
- 3) Aplikasi akan mengarahkan ke halaman *home*.

Catatan:

- 1) Jika anda belum memiliki akun, maka anda harus membuatnya terlebih dahulu dengan menekan link "Register".
- 2) Jika anda lupa *password* akun anda, maka anda bisa menekan link "Lupa Password"

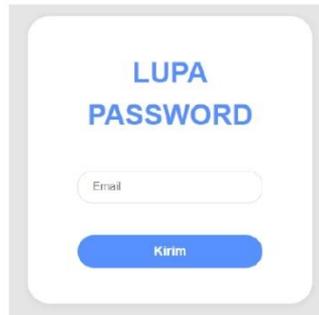


Gambar 1. Tampilan Halaman Login

2. Halaman Lupa Password

Dihalaman ini, *user* mengisikan email yang terdaftar untuk mereset *password* akunnya yang lupa. Tampilan tersebut dapat dilihat pada Gambar 2. Berikut langkah-langkah nya :

- 1) Terdapat inputan untuk *user* mengisinya dengan akun email yang lupa *password* nya.
- 2) *User* akan mendapatkan pesan email yang berisi link untuk mereset *passwordpassword* *user* yang lama.

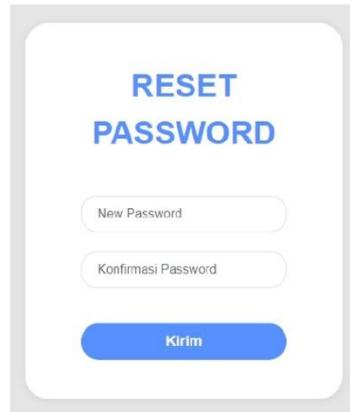


Gambar 2. Tampilan Halaman Lupa Password

3. Halaman Reset Password

Dihalaman ini, *user* mengisi *password* baru untuk mengubah *password* akunnya yang lupa. Tampilan tersebut dapat dilihat pada Gambar 3. Berikut langkah-langkah nya :

- 1) Terdapat inputan untuk *user* mengisinya dengan *password* dan konfirmasi *password*. Klik tombol "Kirim".
- 2) *User* akan dialihkan ke halaman login.



Gambar 3. Tampilan Halaman Reset Password

4. Halaman Registrasi

Pada halaman ini, anda akan diarahkan untuk membuat akun baru agar dapat masuk kedalam website. Tampilan tersebut dapat dilihat pada Gambar 4. Berikut langkah-langkah nya:

- 1) Terdapat empat formular untuk proses pendaftaran, masukkan informasi yang diperlukan seperti Nama, Kota, Provinsi, Email, dan Password. Klik button "Register"
- 2) Anda bisa login.
- 3) Setelah login, website akan menampilkan halaman otp.

Catatan:

- 1) Jika Anda sudah memiliki akun, Anda bisa klik text "Login"

The image shows a registration form with the following elements:

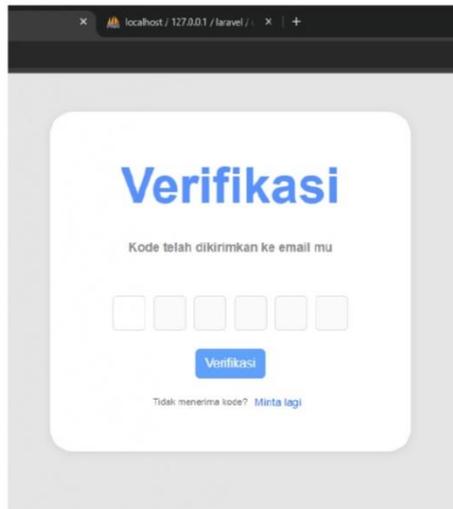
- REGISTER** (Title)
- Nama
- Pilih Negara
- Pilih Provinsi
- Email
- Password
- Repassword
-
- [Sudah punya akun? Login](#)

Gambar 4. Tampilan Halaman Registrasi

5. Halaman Verifikasi OTP

Halaman ini digunakan untuk memastikan bahwa alamat email yang Anda daftarkan valid dan dapat diakses oleh anda. Tampilan tersebut dapat dilihat pada Gambar 5. Berikut langkah-langkah nya:

- 1) Masukkan kode OTP yang sudah terkirim ke Email yang Anda gunakan.
- 2) Klik button “Kirim”
- 3) Aplikasi akan menampilkan halaman isi genre.



Gambar 5. Tampilan Halaman Verifikasi Email

6. Halaman Pilih Genre

Halaman ini digunakan untuk media rekomendasi bagi *user* baru. Dimana *user* akan disuruh memilih 5 genre film kesukaannya. Tampilan tersebut dapat dilihat pada Gambar 6. Berikut langkah-langkah nya:

- 4) *User* memilih 5 genre pada *checkbox* untuk media rekomendasi filmnya.
- 5) Klik button "Kirim"
- 6) Aplikasi akan menampilkan halaman isi *home*.

PILIH 5 GENRE FILM FAVORIT KAMU

| | |
|--|---------------------------------------|
| <input type="checkbox"/> 1. Action | <input type="checkbox"/> 2. Adventure |
| <input type="checkbox"/> 3. Comedy | <input type="checkbox"/> 4. Drama |
| <input type="checkbox"/> 5. Fantasy | <input type="checkbox"/> 6. Family |
| <input type="checkbox"/> 7. History | <input type="checkbox"/> 8. Horror |
| <input type="checkbox"/> 9. Misteri | <input type="checkbox"/> 10. Romance |
| <input type="checkbox"/> 11. Science Fiction | <input type="checkbox"/> 12. Thriller |
| <input type="checkbox"/> 13. Crime | |

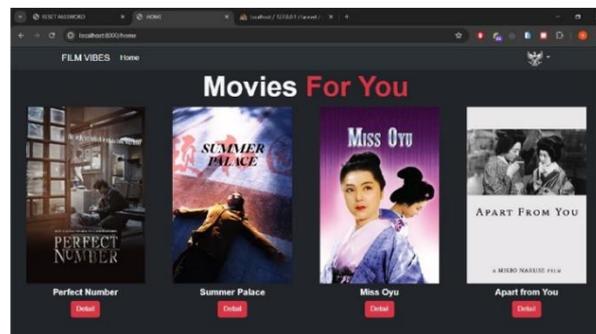
[KAMI](#)

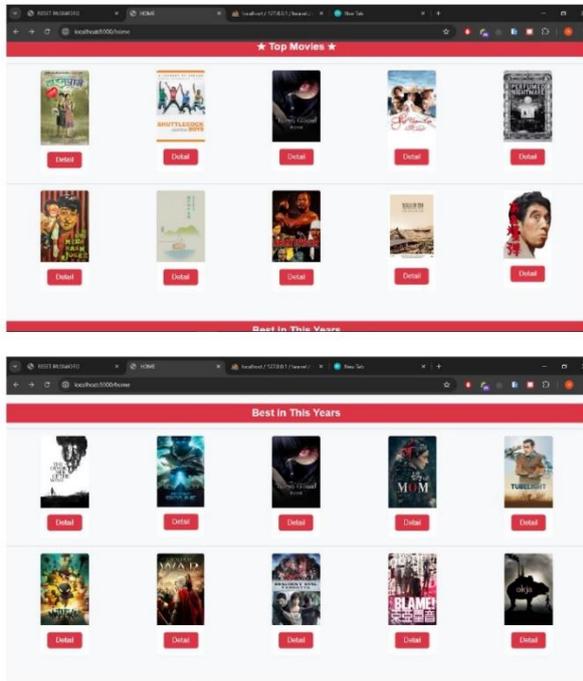
Gambar 6. Tampilan Halaman Pilih Genre

7. Halaman Home

Halaman *home* merupakan halaman yang dilihat setelah anda berhasil login ke aplikasi. Dalam halaman *home*, akan menampilkan beberapa rekomendasi judul film yang diambil berdasarkan kesamaan preferensi tontonan dengan *user* lain. Tampilan tersebut dapat dilihat pada Gambar 6. Berikut langkah-langkah yang harus dilakukan:

- 1) Klik tombol "Detail".
- 2) Website akan mengarahkan *user* ke menu detail dari film yang dipilih.

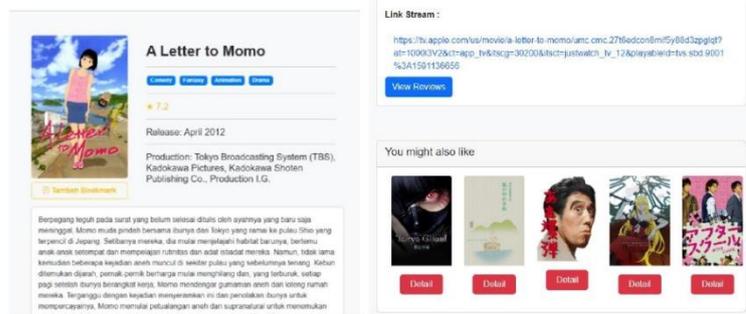




Gambar 7. Tampilan Halaman *Home*

8. Halaman Detail

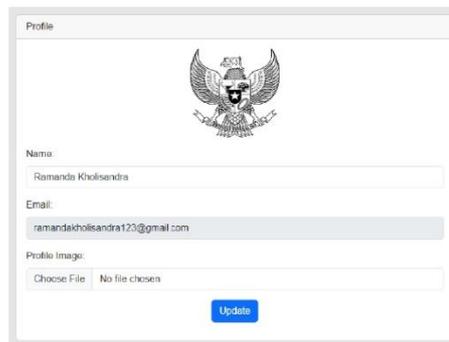
Halaman ini akan menampilkan detail terkait film seperti judul, genre, rating, tahun produksi, perusahaan pemproduksi, sinopsis, *link streaming* (opsional) dan rekomendasi film lainnya . Berikut tampilan pada halaman ini dapat dilihat pada Gambar 8.



Gambar 8. Halaman Detail

9. Halaman Profil Pengguna

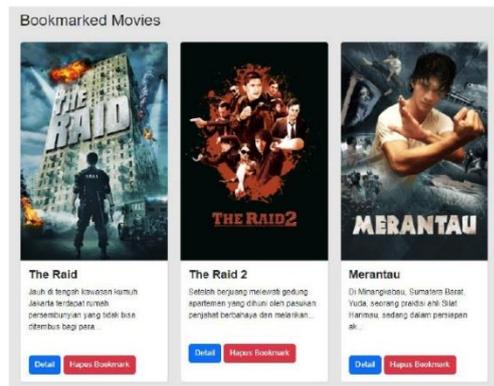
Halaman ini menampilkan detail dari profil Anda. Terdapat nama dan email yang anda gunakan serta fungsi untuk mengubah foto profil akun anda. Berikut tampilan halaman profile dapat dilihat pada Gambar 15.



Gambar 9. Halaman Profile

10. Halaman Bookmark

Halaman ini akan menampilkan beberapa film yang sudah di *bookmark* oleh *user*. Berikut tampilan halaman profile dapat dilihat pada Gambar 10.



Gambar 10. Tampilan Halaman *Bookmark*

11. Halaman Review

Pada halaman ini *user* bisa memberikan komentar, judul kometar, dan rating terhadap film hasil rekomendasi. Tampilan halaman ini bisa dilihat pada Gambar 11.

The image shows a form titled "REVIEW" for the movie "A Letter to Momo". The form has the following fields: "Judul Film" (with "A Letter to Momo" entered), "Rating" (with a dropdown menu), "Judul Review" (with an empty text box), and "Review" (with a large text area). There is a "Submit" button at the bottom left of the form. Below the form, it says "REVIEWS FOR A Letter to Momo".

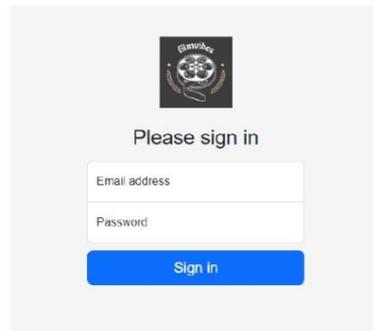
Gambar 11. Tampilan Halaman Review

Admin

1. Halaman Login

Dihalaman login ini, admin dapat masuk kedalam *dashboard* dengan memasukan *email* dan *password* yang sudah terdaftar. Tampilan halaman login admin dapat dilihat pada Gambar 12. Berikut langkah-langkah nya:

- 4) Masukkan *email* dan *password*. Pastikan sudah terdaftar.
- 5) Klik button Login.
- 6) Aplikasi akan mengarahkan ke halaman *dashboard*.

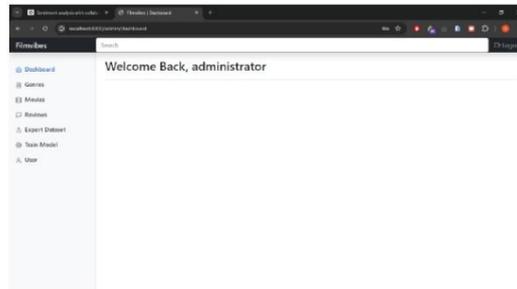


The image shows a login form with a logo at the top, the text "Please sign in", and two input fields for "Email address" and "Password". A blue "Sign In" button is located below the fields.

Gambar 12. Tampilan Halaman Login Admin

2. Halaman Dashboard

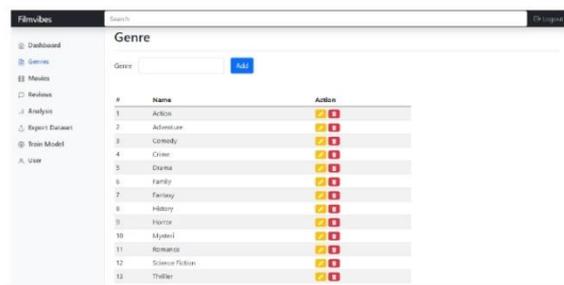
Halaman ini merupakan halaman pembuka untuk halaman *dashboard* admin yang dimana disini akan menampilkan nama *user* yang login ke *dashboard*. Tampilan gambar halaman ini dapat dilihat pada Gambar 13 berikut ini.



Gambar 13. Tampilan Halaman *Dashboard*

3. Halaman Genres

Pada halaman ini akan menampilkan semua data genre yang ditampilkan pada halaman pilih genre saat *user* baru registrasi dan genre pada film. Disini admin bisa mengubah, menambahkan, dan menghapus genre yang diinginkan. Tampilan halaman ini bisa dilihat pada Gambar 14 berikut.



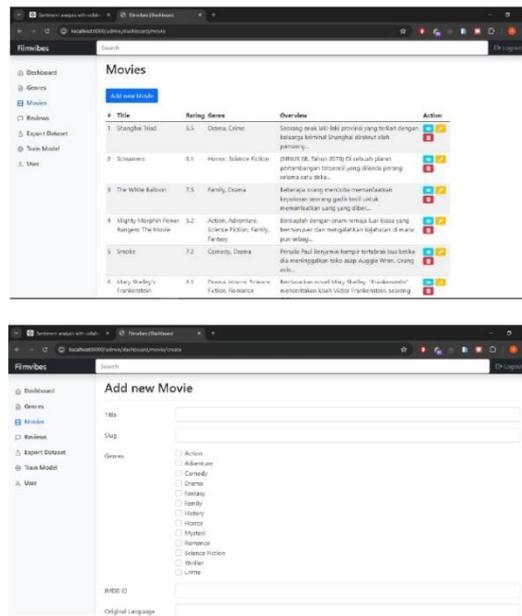
Gambar 14. Tampilan Halaman Genres Admin

4. Halaman Movies

Pada halaman ini akan menampilkan data semua film pada website. Di halaman ini juga, admin dapat menambahkan data film baru, mengubah data film, melihat data film, dan menghapusnya. Tampilan halaman ini bisa dilihat

pada Gambar 15 dibawah ini. Berikut langkah-langkah dalam menambahkan film baru :

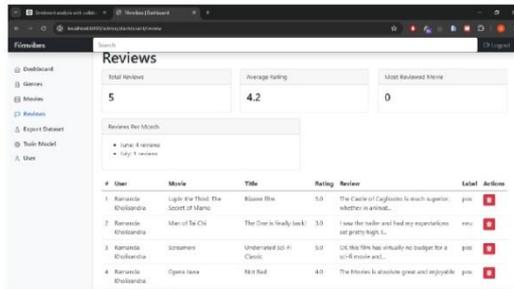
- 1) Tekan tombol “Add new movie”
- 2) Isi data film seperti judul, slug, genre, dan lain-lain.
- 3) Tekan tombol “Submit” untuk menyimpannya.



Gambar 15. Tampilan Halaman Movies Admin

5. Halaman Reviews

Pada halaman ini akan menampilkan beberapa informasi terkait data review semua *user* terhadap film yang mereka review. Disini ditampilkan data total review, rata-rata review, film yang banyak di review, dan akumulasi review per bulan. Admin juga dapat menghapus review yang tidak diinginkan. Tampilan halaman ini dapat dilihat pada Gambar 16 dibawah ini.

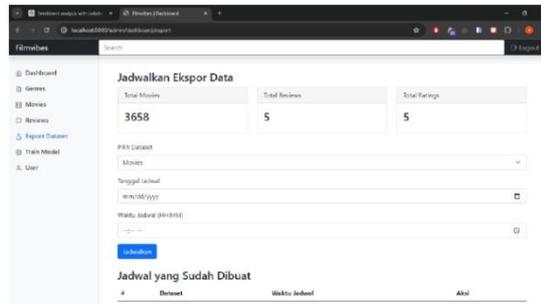


Gambar 16. Tampilan Halaman Reviews Admin

6. Halaman Export Dataset

Pada halaman ini akan menampilkan fitur untuk mengekspor data untuk nantinya dijadikan dataset yang akan di latih sebagai bahan sistem rekomendasi film. Tampilan untuk halaman ini dapat dilihat pada Gambar 18 dibawah ini. Berikut langkah-langkah untuk membuat jadwal ekspor data :

- 1) Pilih dataset yang akan di ekspor (Movies, Rating, dan Review)
- 2) Atur tanggal, bulan dan tahun penjadwalan.
- 3) Atur jam dan menit penjadwalan.
- 4) Klik tombol “jadwalkan”.
- 5) Task akan muncul pada table dibawah yang bisa dihapus.



Gambar 17. Tampilan Halaman *Export Dataset* Admin

7. Halaman Train Model

Halaman ini akan menampilkan fitur untuk melakukan *training* atau pelatihan model sistem rekomendasinya. Admin dapat menentukan nilai parameter *Number of Factors*, *Learning Rate*, dan *Regularization Term* untuk mendapatkan nilai evaluasi yang baik. Tampilan halaman *training model* dapat dilihat pada Gambar 19 dibawah ini.

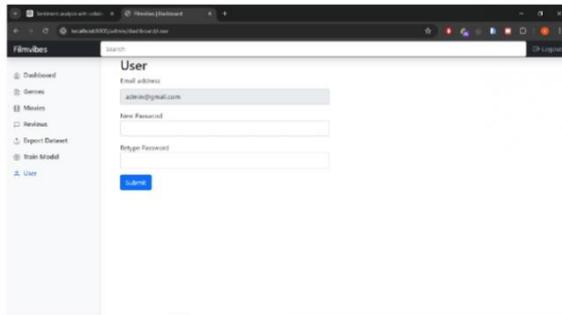


Gambar 18. Tampilan Halaman *Training Model*

8. Halaman User

Di halaman ini akan menampilkan fitur untuk mengganti *password* admin. Tampilan halaman ini dapat dilihat pada Gambar 20 dibawah ini. Berikut langkah-langkah untuk mengganti *password* admin :

- 1) Masukan *password* baru pada inputan.
- 2) Masukan kembali *password* untuk mengkonfirmasi.
- 3) Klik tombol “Submit”.



Gambar 19. Tampilan Halaman *User Admin*

DOKUMEN TEKNIS



Aplikasi Rekomendasi Film Asia Berdasarkan Sentimen Analisis
Menggunakan Collaborative Filtering Berbasis Website.

Oleh :

Ramanda Kholisandra

Dwi Intan Af'idah, S.Kom., M.Kom.

Ardi Susanto, S.Kom., M.Cs.

DAFTAR ISI

| | |
|-------------------------|----|
| DAFTAR ISI..... | 24 |
| DAFTAR GAMBAR..... | 25 |
| Profile..... | 27 |
| Latar Belakang..... | 27 |
| Spesifikasi Teknis..... | 30 |

DAFTAR GAMBAR

| | |
|---|----|
| Gambar 1.1. Source Code Tampilan Login..... | 9 |
| Gambar 1.2. Source Code Lupa Password..... | 10 |
| Gambar 1.3. Tampilan Source Code Reset Password..... | 11 |
| Gambar 1.4. Source Code Tampilan Registrasi..... | 12 |
| Gambar 1.5. Tampilan Source Code Verifikasi OTP..... | 13 |
| Gambar 1.6. Tampilan Source Code Pilih Genre..... | 14 |
| Gambar 1.7. Source Code Tampilan Home..... | 15 |
| Gambar 1.8. Source Code Tampilan Detail..... | 16 |
| Gambar 1.9. Source Code Tampilan Bookmark..... | 18 |
| Gambar 1.10. Source Code Tampilan Detail History..... | 19 |
| Gambar 1.11. Source Code Tampilan Kesimpulan Profile..... | 20 |
| Gambar 1.12. Tampilan Source Code Login Admin..... | 21 |
| Gambar 1.13. Tampilan Source Code Dashboard..... | 22 |
| Gambar 1.14. Tampilan Source Code Dashboard Genre..... | 24 |
| Gambar 1.15. Tampilan Source Code Dashboard Genre Edit..... | 24 |
| Gambar 1.16. Tampilan Source Code Dashboard Movies..... | 26 |
| Gambar 1.17. Tampilan Source Code Dashboard Movies Show..... | 27 |
| Gambar 1.18. Tampilan Source Code Dashboar Movies Create..... | 27 |
| Gambar 1.19. Tampilan Source Code Dashboard Movies Update..... | 28 |
| Gambar 1.20. Tampilan Source Code Dashboard Review..... | 29 |
| Gambar 1.21. Tampilan Source Code Dashboard Export Dataset..... | 31 |
| Gambar 1.22. Tampilan Source Code Training Model..... | 32 |
| Gambar 1.23. Tampilan Source Code Dashboard User..... | 33 |
| Gambar 2.1. Route Endpoint API..... | 34 |
| Gambar 2.2. Route Login..... | 35 |
| Gambar 2.3. Controller Lupa Password..... | 36 |
| Gambar 2.4. Source Code Reset Password..... | 37 |
| Gambar 2.5. Source Code Route Register..... | 38 |
| Gambar 2.6. Route Verify OTP..... | 39 |

| | |
|---|----|
| Gambar 2.7. Route Choose Genre | 40 |
| Gambar 2.8. Route Home | 41 |
| Gambar 2.9. Route Profile | 42 |
| Gambar 2.10. Route Review | 43 |
| Gambar 2.11. Route Bookmark | 44 |
| Gambar 2.12. Route Admin Login..... | 45 |
| Gambar 2.13. Route Admin Dashboard Genre | 46 |
| Gambar 2.14. Route Admin Dashboard Movies..... | 47 |
| Gambar 2.15. Route Admin Dashbard Review..... | 48 |
| Gambar 2. 16. Route Admin Dashboard Export Dataset..... | 49 |
| Gambar 2. 17. Route Training Model | 50 |
| Gambar 2. 18. Route Admin Dashboard User | 51 |

Profile

Filmvibes merupakan sebuah aplikasi website untuk merekomendasikan film khususnya di wilayah Asia dengan menggunakan algoritma Collaborative Filtering dan pemanfaatan sentiment analisis dari ulasan *user*. Sistem akan merekomendasikan berdasarkan kesamaan preferensi tontonan atau selera film seorang *user* dengan *user* lainnya. *User* akan memberikan review terhadap film yang nantinya akan digunakan sebagai media dataset untuk pembuatan model sistem rekomendasinya. Dengan ini, diharapkan *user* akan memiliki banyak referensi untuk menonton sebuah film tanpa terbatas dari satu preferensi saja seperti genre atau lainnya.

Latar Belakang

Film adalah sebuah kesenian gabungan antara seni sastra, seni peran, seni musik, dan seni rupa. Pada masa sekarang ini film telah menjadi salah satu hiburan favorit utama masyarakat. Jumlah film pertahun terhitung mencapai ribuan. Dilihat dari banyaknya jumlah penonton bioskop yang terus meningkat dari tahun ke tahun. Per 2018 angka jumlah penonton bioskop di Indonesia saja telah mencapai lebih dari 50 juta penonton dengan jumlah produksi film luar negeri hingga dalam negeri sebanyak hampir 200 judul film yang telah tayang di seluruh Indonesia. Setiap tahun, banyak film baru yang diproduksi oleh industri yang membuat beberapa orang bingung untuk memilih film mana yang layak untuk ditonton. Kelayakan sebuah film menjadi sebuah aspek penting untuk memberikan penilaian positif terhadap film, sehingga dapat menjadikan film tersebut laris di kalangan masyarakat luas dan penonton yang menonton juga akan merasakan kepuasan.

Pemilihan film yang bagus untuk ditonton menjadi sebuah masalah yang sering dijumpai. Kesesuaian genre, rating, alur cerita, pengambilan gambar, pemilihan musik, dan aspek lainnya merupakan sebuah poin penting yang mendasari film tersebut laris di kalangan masyarakat. Kelayakan sebuah film sangat mempengaruhi kepuasan penonton, karena kelayakan film secara langsung mempengaruhi pengalaman menonton dan reaksi emosional penonton. Saat penonton merasa terhubung secara emosional dengan cerita, karakter, atau tema yang disajikan dalam film, tingkat kepuasan mereka meningkat, dan mereka lebih cenderung akan merekomendasikan film tersebut kepada orang lain serta

ingin menontonnya kembali. Oleh karena itu, dibutuhkan sebuah aplikasi yang dapat merekomendasikan film berdasarkan selera dari sang penonton.

Beberapa penelitian menunjukkan bahwa kepuasan film dapat ulasan yang ditulis penonton tersebut di media sosial. Dari ulasan yang di tulis itu, dapat diambil beberapa informasi terkait aspek-aspek kepuasan sang penonton terhadap film tersebut. Penelitian terdahulu menunjukkan bahwa pembuatan sistem rekomendasi film dengan menggunakan metode collaborative filtering dapat membantu untuk mengambil beberapa data informasi dari komentar sang penonton untuk nantinya akan di kumpulkan sebagai media rekomendasi, film seperti apa yang disukai oleh sang penonton. Dengan begitu, penonton dapat mendapatkan rekomendasi film sesuai dengan beberapa kriteria yang disukai seperti genre, durasi film, dan lain sebagainya. Namun pada penelitian tersebut lingkupnya hanya berupa rating tertinggi.

Dari hasil penelitian sebelumnya, penggunaan metode untuk sistem rekomendasi film adalah dengan menggunakan sistem analisis sentiment dengan meneliti bagian ulasan penonton terkait film yang telah ditonton. Untuk algoritmanya sendiri akan menggunakan naive bayes dikarenakan, nilai akurasi dalam pelatihan model sistem yang terbilang cukup akurat. Dari model yang telah dilatih tersebut nantinya akan diimplementasikan kedalam aplikasi berbasis website untuk mendapatkan informasi berdasarkan ulasan yang ditulis oleh pengguna Setelah menonton film yang telah direkomendasikan. Model akan mendeteksi ulasan pengguna bernilai positif atau negative yang nantinya akan digunakan sebagai media untuk rekomendasi film untuk pengguna selanjutnya. Dengan demikian, diharapkan pengguna dapat lebih mudah menemukan film yang sesuai dengan kriteria tertentu dan dapat meningkatkan pengalaman menonton film secara keseluruhan.

Manfaat

1. Membangun aplikasi berbasis website yang memungkinkan pengguna untuk dengan mudah menemukan film-film Asia yang sesuai dengan preferensi mereka, meningkatkan kepuasan dalam pengalaman menonton.
2. Menyediakan aplikasi yang dapat merekomendasi film secara relevan berdasarkan analisis ulasan, pengguna sehingga dapat menghemat waktu untuk mencari film yang sesuai dengan selera pengguna.
3. Membantu dalam memperluas cakupan penonton untuk film-film Asia yang mungkin sebelumnya kurang dikenal dengan memberikan rekomendasi yang sesuai dengan preferensi individu pengguna.
4. Meningkatkan keterlibatan pengguna dengan platform melalui interaksi dengan aplikasi dan kontribusi ulasan, memperkuat komunitas pengguna dan meningkatkan relevansi rekomendasi film.

Spesifikasi Teknis

Spesifikasi Teknis, meliputi:

1. Modul Pengguna
2. *Source Code*

Berikut uraian spesifikasi untuk pembangunan aplikasi:

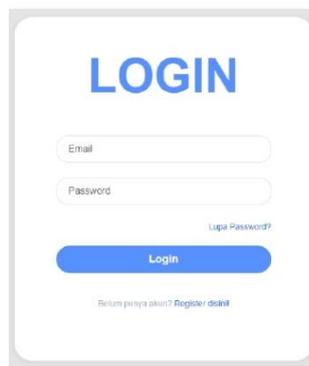
1. Flask Python
2. Visual Studio Code
3. Google Collaboration
4. Web Browser (Chrome, Edge)
5. Framework Laravel
6. XAMPP

Berikut uraian spesifikasi modul:

1. Modul untuk pengguna
2. *Source Code*
 - 1) *User Interface*.

1. Tampilan Login

Pada tampilan halaman login menggunakan form yang menggunakan method 'POST' dan mengirim data ke route 'login'. Memiliki 2 inputan yaitu email yang bertipe 'email' dan password yang bertipe 'password'. Pada login juga ada link 'Lupa Password' yang berfungsi bagi *user* yang lupa password akun mereka. Disana juga terdapat link 'Register' yang diperuntukan bagi *user* yang ingin membuat akun. Kedua inputan *user* yang dimasukan yaitu email dan password akan di olah di bagian *Controller* untuk di cek dan disamakan dengan data yang ada di database apakah ada data *user* yang sesuai data inputan. Tampilan *Source Code* halaman login dapat dilihat pada Gambar 1.1.

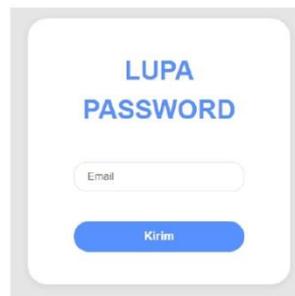
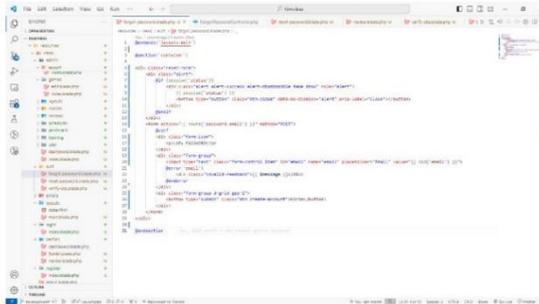


Gambar 1.1. Source Code Tampilan Login

2. Tampilan Lupa Password

Pada tampilan halaman lupa password, form menggunakan method 'POST' dan mengirim data ke route 'password.email'. Form ini memiliki satu inputan yaitu email yang bertipe 'text' dimana *user* dapat memasukkan email mereka untuk mendapatkan link reset password. Jika ada kesalahan input pada email, pesan kesalahan akan ditampilkan di bawah input email tersebut. Ketika user berhasil mengirim permintaan reset password, pesan sukses akan ditampilkan. Pada halaman ini juga terdapat judul 'LUPA PASSWORD' yang ditampilkan di bagian atas form. Tombol 'Kirim' digunakan untuk mengirim

permintaan reset password. Tampilan *Source Code* tersebut dapat dilihat pada Gambar 1.3.



Gambar 1.2. *Source Code* Lupa Password

3. Tampilan Reset Password

Pada tampilan halaman reset password, form menggunakan method 'POST' dan mengirim data ke route 'password.update'. Form ini memiliki dua inputan yaitu password baru dan konfirmasi password. Kedua inputan tersebut bertipe 'password' dimana user harus memasukkan password baru dan mengonfirmasi password baru mereka. Jika ada kesalahan dalam input, pesan kesalahan akan ditampilkan di atas form. Pada halaman ini juga terdapat judul 'RESET PASSWORD' yang ditampilkan di bagian atas form. Tombol 'Kirim'

digunakan untuk mengirim permintaan reset password. Tampilan *source code* halaman reset password dapat dilihat pada Gambar 1.3.

```

<code>
</code>

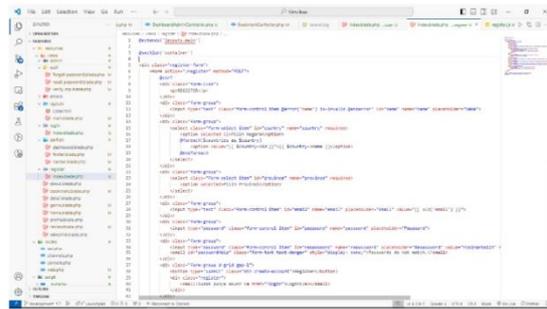
```

Gambar 1.3. Tampilan *Source Code* Reset Password

4. Tampilan Register

Tampilan registrasi ini menampilkan formulir yang terdiri dari empat input yaitu nama, kota, provinsi, email, dan password. Masing-masing dikelola oleh input yang bertipe 'text' untuk kolom nama, email, dan password dan input bertipe 'select' untuk kolom kota dan provinsi yang memiliki fungsi mengontrol teks yang diinput pengguna, serta divalidasi di bagian *Controller* menggunakan 'validate' untuk menyimpan dan memvalidasi inputan yang diisikan ke dalam form register diisi dengan benar. Halaman menyediakan tombol *Register* yang

terhubung ke fungsi 'store' pada *Controller* yang menerima inputan *user* dan mengirim data ke server. Dan juga tersedia teks link 'Login' jika pengguna sudah memiliki akun. *Source Code* tersebut dapat dilihat pada Gambar 1.1.



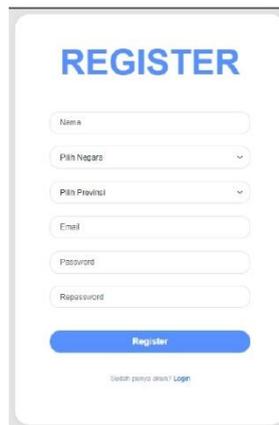
```
function register($nama, $password, $repassword, $email, $provinsi, $negara) {
    // Validasi input
    if (empty($nama) || empty($password) || empty($repassword) || empty($email) || empty($provinsi) || empty($negara)) {
        return false;
    }

    // Cek apakah email sudah terdaftar
    $query = "SELECT * FROM users WHERE email = '$email'";
    $result = mysqli_query($koneksi, $query);
    if (mysqli_num_rows($result) > 0) {
        return false;
    }

    // Cek apakah password dan repassword sama
    if ($password != $repassword) {
        return false;
    }

    // Hash password
    $password = password_hash($password, PASSWORD_DEFAULT);

    // Simpan data ke database
    $query = "INSERT INTO users (nama, password, email, provinsi, negara) VALUES ('$nama', '$password', '$email', '$provinsi', '$negara')";
    $result = mysqli_query($koneksi, $query);
    if ($result) {
        return true;
    } else {
        return false;
    }
}
```



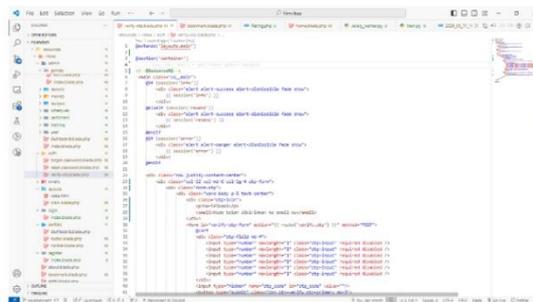
The image shows a web registration form with the following fields and elements:

- REGISTER** (Title)
-
-
-
-
-
-
-
- [Sudah punya akun? Login](#)

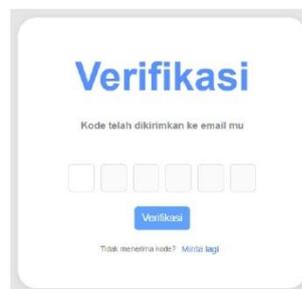
Gambar 1.4. *Source Code* Tampilan Registrasi

5. Tampilan Verifikasi OTP

Pada tampilan halaman verifikasi OTP, form menggunakan method 'POST' dan mengirim data ke route 'verify.otp'. Form ini memiliki enam inputan bertipe 'number' yang masing-masing dapat menampung satu digit kode OTP. Selain itu, terdapat juga tombol 'Verifikasi' untuk mengirim kode OTP yang dimasukkan. Jika ada pesan informasi, pesan resend, atau pesan kesalahan dari session, pesan tersebut akan ditampilkan di bagian atas form. Pada halaman ini juga terdapat judul 'Verifikasi' dan pesan 'Kode telah dikirimkan ke email mu' yang ditampilkan di bagian atas form. Terdapat pula form lain untuk mengirim permintaan pengiriman ulang kode OTP ke route resend.otp dengan tombol 'Minta lagi'. Tampilan *source code* halaman verifikasi OTP dapat dilihat pada Gambar 1.5.



The image shows a code editor window with a file explorer on the left and a code editor on the right. The code is written in PHP and includes a session check, a message display, a form with six number input fields, a 'Verifikasi' button, and a 'Minta lagi' link. The code uses Laravel-style syntax for session management and form rendering.



The image shows a web form titled 'Verifikasi'. Below the title, it says 'Kode telah dikirimkan ke email mu'. There are six empty input boxes for the OTP code. Below the boxes is a blue button labeled 'Verifikasi'. At the bottom, there is a link that says 'Tidak menerima kode? Minta lagi'.

Gambar 1.5. Tampilan *Source Code* Verifikasi OTP

6. Tampilan Pilih Genre

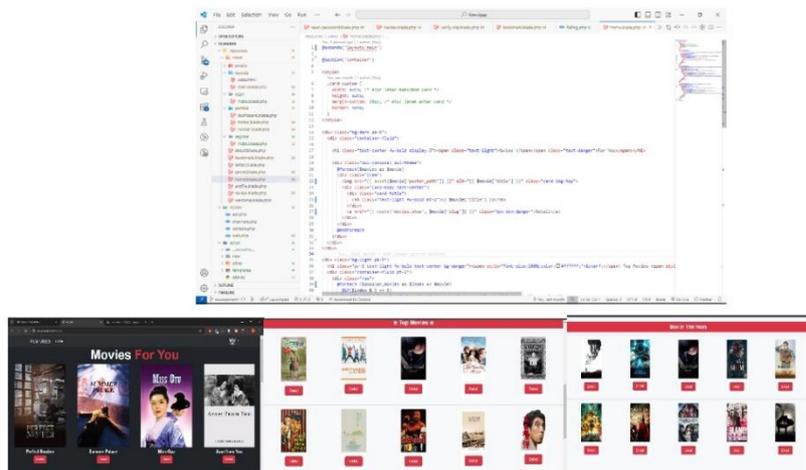
Pada tampilan halaman pemilihan genre, form menggunakan method 'POST' dan mengirim data ke route 'store.genre'. Form ini menampilkan beberapa *checkbox* yang berisi daftar genre film yang bisa dipilih oleh *user*, dengan maksimal lima genre favorit. Jika user memilih lebih dari lima genre, *checkbox* tambahan tidak akan dicentang berkat adanya *JavaScript* yang mengelola batasan tersebut. Jika ada pesan sukses dari *session*, pesan tersebut akan ditampilkan di bagian atas form. Pada halaman ini juga terdapat judul 'PILIH 5 GENRE FILM FAVORIT KAMU' di bagian atas form, dan tombol 'Kirim' digunakan untuk mengirim pilihan genre. Tampilan source code halaman pemilihan genre dapat dilihat pada Gambar 1.5.



Gambar 1.6. Tampilan *Source Code* Pilih Genre

7. Tampilan Home

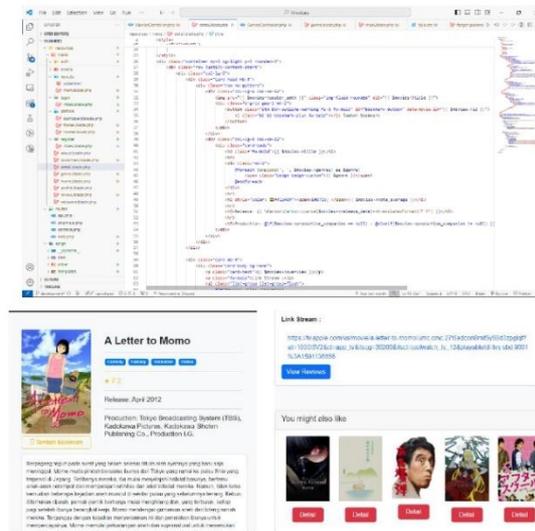
Pada tampilan halaman home, terdapat berbagai bagian yang menampilkan film untuk pengguna. Halaman ini dibagi menjadi tiga bagian utama: rekomendasi film untuk pengguna, film paling populer, dan film terbaik tahun ini. Bagian pertama adalah rekomendasi film untuk pengguna berdasarkan genre favorit mereka yang ditampilkan dalam carousel, di mana setiap item film memiliki poster, judul, dan tombol untuk melihat detail film tersebut. Bagian kedua adalah film paling populer yang ditampilkan dalam grid dengan masing-masing baris berisi lima film; setiap film ditampilkan dalam bentuk kartu kecil dengan poster film dan tombol detail. Bagian ketiga adalah film terbaik tahun ini, yang juga ditampilkan dalam grid dengan cara yang sama seperti bagian film populer. Tampilan *source code* tampilan home dapat dilihat pada Gambar 1.7.



Gambar 1.7. *Source Code* Tampilan Home

8. Tampilan Detail

Pada tampilan halaman detail, pengguna dapat melihat informasi lengkap mengenai film tertentu yang mereka pilih. Halaman ini menampilkan poster film, judul, genre, rata-rata suara, tanggal rilis, perusahaan produksi, dan deskripsi singkat tentang film tersebut. Terdapat juga tombol untuk menambahkan film ke daftar bookmark pengguna serta tautan untuk menonton film secara daring jika tersedia. Selain itu, pengguna dapat melihat lima film lain yang berasal dari negara produksi yang sama dan memiliki nilai rata-rata suara tertinggi, namun tidak termasuk film yang sedang ditampilkan, sebagai rekomendasi film serupa. *Source Code* tersebut dapat dilihat pada Gambar 1.8.

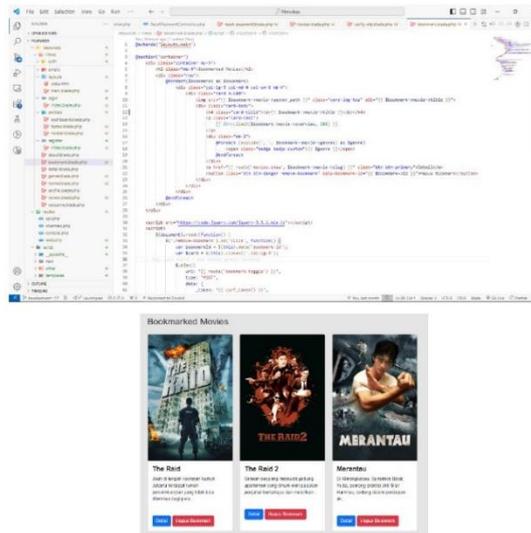


Gambar 1.8. *Source Code* Tampilan Detail

9. Tampilan Bookmark

Pada halaman bookmark, pengguna dapat melihat daftar film yang telah mereka tandai sebagai bookmark. Setiap film ditampilkan dengan judul, gambar poster, ringkasan singkat, genre, dan tombol untuk melihat detail film atau menghapus film dari daftar bookmark. Tombol "Hapus Bookmark" memungkinkan pengguna untuk menghapus film dari daftar bookmark mereka tanpa perlu meninggalkan halaman.

Halaman ini terhubung ke BookmarkController, yang dimana terdapat metode untuk menangani penampilan dan interaksi pengguna terkait daftar bookmark. Metode index mengambil daftar bookmark dari database untuk pengguna yang sedang terautentikasi, sementara metode toggle memungkinkan pengguna untuk menambahkan atau menghapus bookmark dengan bantuan AJAX. Selain itu, terdapat metode check yang digunakan untuk memeriksa apakah suatu film sudah ditandai sebagai bookmark oleh pengguna, yang memungkinkan penyesuaian tampilan tombol bookmark sesuai dengan status saat ini. Metode store mengelola penambahan bookmark baru ke dalam basis data, memastikan tidak ada duplikat bookmark untuk film yang sama. Dengan demikian, BookmarkController menyediakan fungsi yang penting untuk mengelola interaksi pengguna terkait bookmark secara efektif dan efisien. *Source Code* tersebut dapat dilihat pada Gambar 1.9.

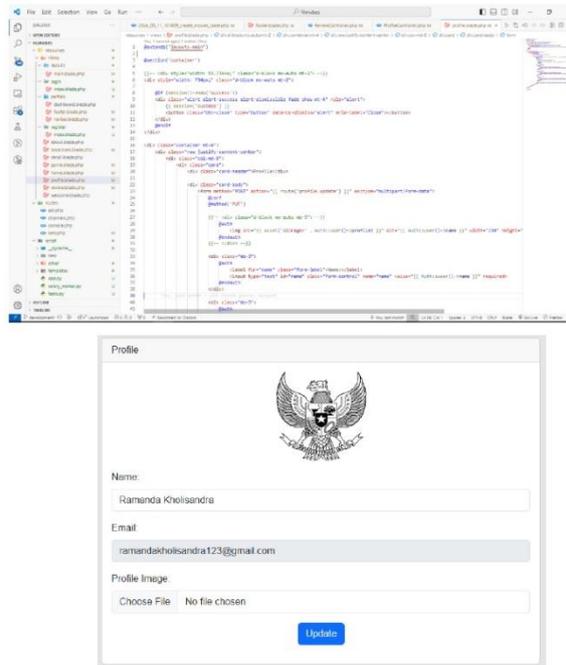


Gambar 1.9. Source Code Tampilan Bookmark

10. Tampilan Profile

Pada halaman profile, pengguna dapat melihat dan mengubah informasi profil mereka, termasuk nama, email (ditampilkan namun tidak dapat diubah), serta gambar profil. Terdapat juga tombol untuk memperbarui informasi profil dan pemberitahuan sukses setelah profil berhasil diperbarui. Di sisi ProfileController, terdapat metode show yang menampilkan halaman profil pengguna yang sedang terautentikasi dengan menggunakan data dari Auth::user(). Metode ini mengirimkan data tersebut ke halaman view profile untuk ditampilkan. Selain itu, metode update memproses pembaruan informasi profil berdasarkan data yang dikirimkan melalui form, termasuk validasi untuk memastikan nama diisi dan format foto profil yang benar. Proses ini juga mengelola pengunggahan gambar profil baru, menghapus foto profil lama jika ada, dan menyimpan foto profil baru ke dalam penyimpanan yang ditetapkan (public disk). Setelah berhasil diperbarui, pengguna akan diarahkan kembali ke

halaman profil dengan pemberitahuan sukses, memastikan fungsionalitas lengkap untuk mengelola dan memperbarui profil dengan mudah dan aman.. *Source Code* tersebut dapat dilihat pada Gambar 1.10.

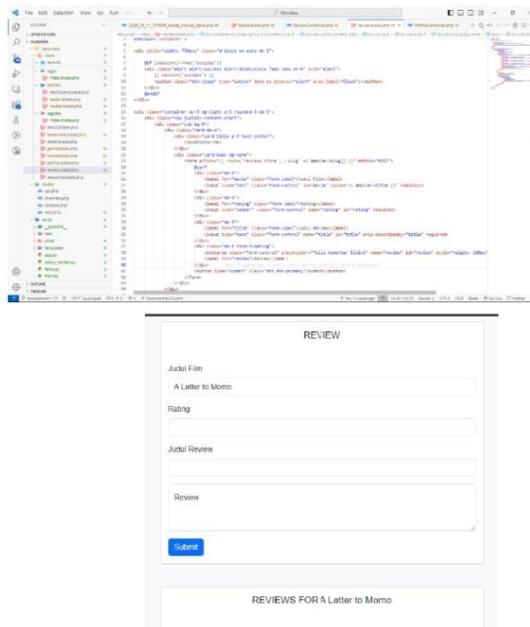


Gambar 1.10. *Source Code* Tampilan Detail History

11. Tampilan Review

Pada halaman `review.blade.php`, pengguna dapat menambahkan ulasan untuk sebuah film dengan mengisi rating, judul ulasan, dan komentar. Data yang diinputkan akan disubmit dan disimpan menggunakan metode `store` dalam `ReviewController`. Setelah berhasil disimpan, pengguna akan dialihkan kembali ke halaman dengan pemberitahuan sukses. `ReviewController` sendiri memiliki dua metode utama yaitu `index($slug)` yang digunakan untuk menampilkan semua

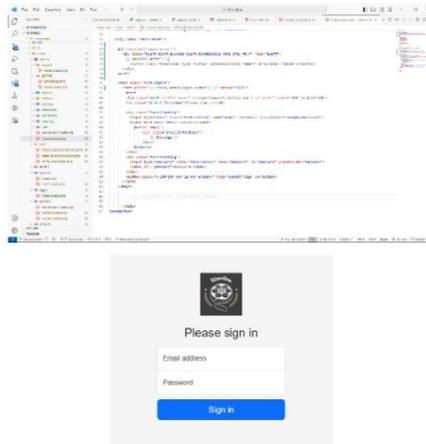
ulasan yang terkait dengan film yang dipilih berdasarkan slug. Metode ini memungkinkan pengguna untuk melihat ulasan yang sudah ada. Dan store(Request \$request, \$slug) yang memproses data yang dikirimkan dari form untuk menyimpan ulasan baru ke dalam database. Sebelum disimpan, metode ini melakukan validasi terhadap rating, judul ulasan, dan komentar. Selain itu, metode ini menggunakan Google Translate untuk menerjemahkan ulasan ke bahasa Inggris dan mengidentifikasi sentimen dari ulasan tersebut sebelum disimpan. Dengan ini, pengguna dapat berinteraksi dengan ulasan film secara dinamis, menambahkan ulasan baru, dan melihat ulasan yang sudah ada dengan mudah. *Source Code* tersebut dapat dilihat pada Gambar 1.11.



Gambar 1.11. *Source Code* Tampilan Kesimpulan Profile

12. Tampilan Login Admin

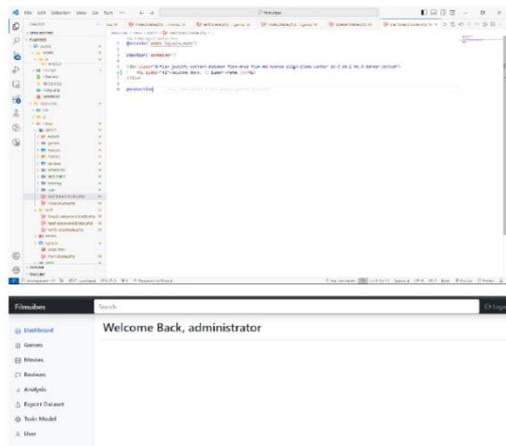
Halaman ini adalah halaman login untuk admin. Pada bagian atas halaman, terdapat penanganan pesan sukses atau error menggunakan session Laravel untuk memberikan feedback kepada pengguna setelah tindakan mereka. Form login terdiri dari dua input utama: email dan password. Input email menggunakan validasi bawaan Laravel untuk memastikan bahwa nilai yang dimasukkan adalah alamat email yang valid. Begitu juga dengan input password, yang diperlukan dan harus memenuhi panjang minimum yang ditentukan (dalam kasus ini, minimal 8 karakter). Saat form disubmit, data dikirimkan ke endpoint route 'admin.login.submit' menggunakan metode POST. Laravel memproses data ini dengan middleware CSRF protection yang diaktifkan secara otomatis, yang memastikan keamanan form dari serangan Cross-Site Request Forgery (CSRF). Jika pengguna gagal login (misalnya, karena salah memasukkan email atau password), pesan kesalahan akan ditampilkan di bagian atas form. Pesan ini menggunakan Bootstrap alert untuk memberikan tampilan yang jelas dan responsif kepada pengguna. *Source code* dapat dilihat pada Gambar 1.12.



Gambar 1.12. Tampilan *Source Code* Login Admin

13. Tampilan Dashboard

Halaman dashboard adalah bagian dari admin area yang menggunakan template `main.blade.php` sebagai layout utama. Pada bagian ini, terdapat sebuah header yang menyertakan judul "Welcome Back" diikuti dengan nama pengguna yang sedang login. Layout utama dari halaman ini dibangun dengan menggunakan Bootstrap CSS untuk struktur dasar, yang memberikan tata letak responsif dan desain yang konsisten. Bootstrap Icons digunakan untuk ikon-ikon yang mungkin diperlukan, seperti yang diintegrasikan melalui `main.blade.php`. *Source code* dashboard bisa dilihat pada Gambar 1.13.



Gambar 1.13. Tampilan Source Code Dashboard

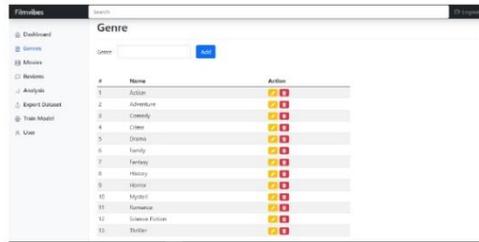
14. Tampilan Dashboard Genres

Halaman dashboard genre adalah bagian dari antarmuka admin yang digunakan untuk mengelola genre film. Halaman ini mengadopsi layout utama dari `main.blade.php` dalam admin, menampilkan daftar genre dalam sebuah tabel dengan opsi untuk menambah, mengedit, dan menghapus setiap entri genre. Form penambahan genre memungkinkan pengguna untuk memasukkan nama genre baru dan menyimpannya melalui aksi POST ke route 'admin.genre.store'.

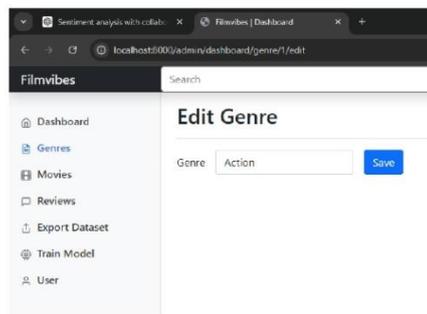
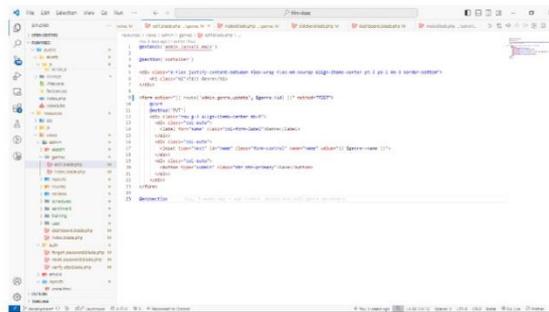
Setiap baris tabel menampilkan nomor urut, nama genre, serta tombol edit dan hapus. Tombol edit mengarahkan pengguna ke halaman edit.blade.php untuk mengubah detail genre yang sudah ada, sementara tombol hapus menghapus genre dengan metode DELETE. Teknologi yang digunakan meliputi PHP Blade Templating untuk integrasi data dinamis dari backend dengan tata letak HTML, Bootstrap untuk responsivitas dan tata letak UI, serta Laravel Routing untuk mengarahkan pengguna ke fungsi CRUD yang sesuai dalam model Genre. Halaman ini memfasilitasi administrasi efektif dan efisien dari genre film dalam sistem aplikasi Filmvibes.

Halaman edit genre menyediakan formulir untuk mengubah detail sebuah genre film yang sudah ada. Dengan menggunakan metode POST ke admin.genre.update, halaman ini menampilkan kolom input yang memuat nama genre yang sedang diedit. Pengguna dapat melakukan perubahan dan menyimpannya menggunakan tombol "Save". Implementasi menggunakan PHP Blade Templating untuk menampilkan formulir dengan data yang diisi sebelumnya dari model Genre, Bootstrap untuk tata letak dan elemen UI seperti kolom input dan tombol "Save", serta Laravel Routing untuk memproses perubahan yang dilakukan pada genre. Ini memungkinkan admin untuk mengelola dan memperbarui informasi genre film secara efisien dalam aplikasi Filmvibes. *Source code* dapat dilihat pada Gambar 1.14. dan Gambar 1.15. berikut ini.





Gambar 1.14. Tampilan *Source Code* Dashboard Genre



Gambar 1.15. Tampilan *Source Code* Dashboard Genre Edit

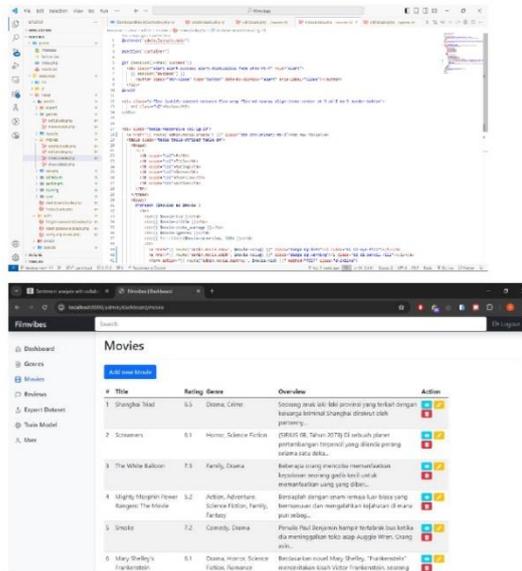
15. Tampilan Dashboard Movies

Halaman dashboard movies ini dirancang untuk memungkinkan pengelolaan film dalam aplikasi dengan menggunakan Laravel. Halaman utama menampilkan daftar film dengan opsi untuk menambah, melihat detail, mengedit, dan menghapus film. Desain halaman menggunakan Bootstrap untuk tampilan responsif yang baik. Setiap film ditampilkan dalam tabel dengan kolom-kolom yang mencakup informasi seperti judul, rating, genre, dan ringkasan singkat. Pengguna dapat menambahkan film baru dengan mengklik tombol "Add new Movie", yang mengarahkan mereka ke halaman formulir untuk mengisi detail film seperti judul, genre, IMDB ID, dan lainnya.

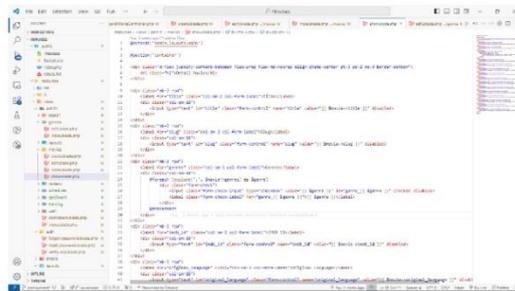
Untuk melihat detail film, pengguna dapat mengklik ikon mata untuk membuka halaman detail yang menampilkan informasi lebih lanjut seperti slug, bahasa asli, tanggal rilis, dan lainnya. Halaman ini juga memungkinkan pengguna untuk mengedit informasi film dengan mengklik ikon pensil, yang mengarahkan mereka ke formulir prapengisian dengan data film yang dapat diubah.

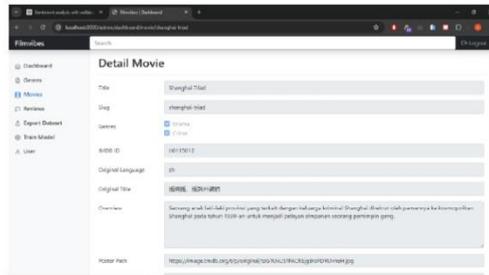
Proses penghapusan film dilakukan dengan mengklik ikon sampah yang mengarahkan pengguna ke konfirmasi penghapusan. Semua operasi ini diimplementasikan dengan menggunakan Laravel dan PHP untuk logika backend, sementara tampilan frontend menggunakan Blade templating engine untuk integrasi yang mulus. Fitur pagination disediakan secara otomatis oleh Laravel untuk menangani banyaknya data film, dengan tautan navigasi di bagian bawah tabel untuk navigasi yang mudah.

Secara keseluruhan, dashboard movies ini dirancang untuk memberikan pengalaman pengelolaan film yang intuitif dan efisien dalam aplikasi berbasis web menggunakan Laravel. *Source code* dapat dilihat pada Gambar 1.16., Gambar 1.17., Gambar 1.18. dan Gambar 1.19. berikut ini.

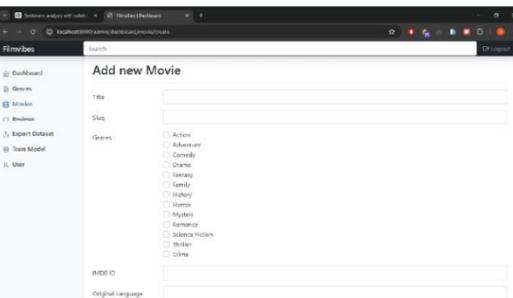
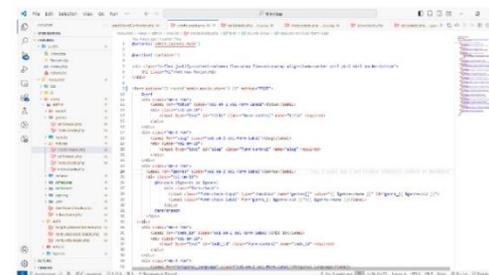


Gambar 1.16. Tampilan Source Code Dashboard Movies

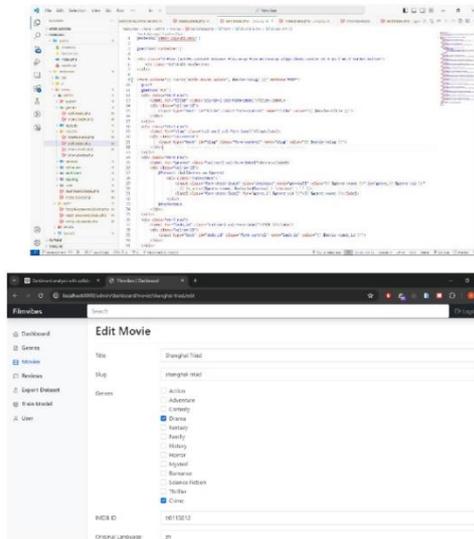




Gambar 1.17. Tampilan *Source Code* Dashboard Movies Show



Gambar 1.18. Tampilan *Source Code* Dashboard Movies Create



Gambar 1.19. Tampilan *Source Code* Dashboard Movies Update

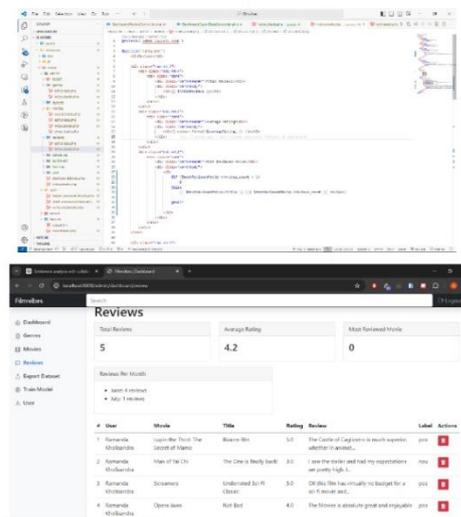
16. Tampilan Dashboard Reviews

Pada halaman ini, data ulasan film ditampilkan dengan detail statistik yang relevan. Saat halaman dimuat, `index()` method dari `DashboardReviewController` dijalankan. Proses ini melibatkan pengambilan data ulasan dengan informasi tambahan seperti pengguna yang melakukan ulasan dan film yang diulas, yang kemudian ditampilkan dengan sistem paginasi untuk memudahkan navigasi.

Selain daftar ulasan, beberapa statistik penting juga dihitung langsung dari basis data. Informasi seperti total jumlah ulasan (`totalReviews`), rata-rata rating ulasan (`averageRating`), dan film yang paling banyak mendapat ulasan (`mostReviewedMovie`) diambil dari model `Review` dan `Movies` menggunakan fitur Eloquent ORM Laravel. Selain itu, terdapat bagian yang menampilkan statistik jumlah ulasan per bulan (`reviewsPerMonth`). Data ini diambil dengan

mengelompokkan ulasan berdasarkan bulan pembuatan (`created_at`) dan menghitung jumlah ulasan untuk setiap bulan dalam sistem.

Hasil perhitungan dan data yang diambil kemudian dikirimkan ke view `admin.reviews.index` dengan menggunakan fungsi `compact()`, memastikan variabel yang diperlukan tersedia untuk ditampilkan. Dengan demikian, halaman dashboard dirancang untuk memberikan gambaran lengkap dan terperinci tentang aktivitas ulasan film dalam sistem administrasi, memberikan informasi yang berharga kepada pengguna admin. *Source code* dapat dilihat pada Gambar 1.20. berikut ini.



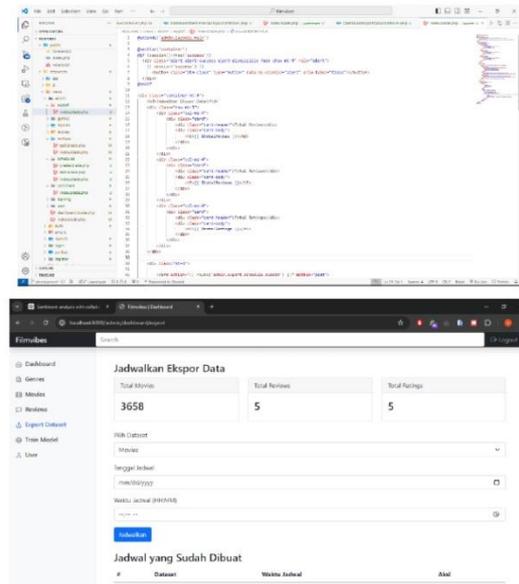
Gambar 1.20. Tampilan Source Code Dashboard Review

17. Tampilan Dashboard Export Dataset

Halaman dashboard ekspor data bertujuan untuk mengelola dan mengekspor data terkait film, ulasan, dan rating ke dalam file CSV. Halaman ini menggunakan layout utama dari admin ('admin.layouts.main') dengan konten utama yang disusun di dalam section 'container'. Ketika halaman dimuat, terdapat fitur notifikasi menggunakan session Laravel untuk menampilkan pesan sukses jika ada operasi yang berhasil dilakukan. Di bawah notifikasi, halaman menampilkan beberapa statistik utama seperti total film ('\$totalMovies'), total ulasan ('\$totalReviews'), dan total rating ('\$totalRatings'). Informasi-informasi ini ditampilkan dalam bentuk kartu yang disusun secara rapi menggunakan Bootstrap grid system.

Selanjutnya, halaman menyediakan formulir untuk jadwal ekspor data, yang memungkinkan pengguna untuk memilih jenis dataset (movies, reviews, atau ratings), tanggal jadwal, dan waktu jadwal. Setelah formulir disubmit, data jadwal disimpan ke dalam basis data menggunakan model 'Schedule'. Selain itu, perintah Artisan ('Artisan::call()') juga dijalankan untuk menjadwalkan eksekusi perintah eksportasi data sesuai dengan dataset dan waktu yang ditentukan. Di bagian bawah halaman, terdapat tabel yang menampilkan daftar jadwal ekspor yang sudah dibuat sebelumnya. Setiap entri tabel menampilkan dataset yang dijadwalkan untuk diekspor, waktu jadwal eksekusi, dan opsi untuk menghapus jadwal tersebut.

Secara keseluruhan, halaman ini dirancang untuk memberikan fungsionalitas lengkap dalam mengelola dan menjadwalkan proses ekspor data terstruktur dari aplikasi ke dalam format CSV, memberikan alat yang efektif bagi admin untuk mengatur dan mengawasi proses ini dengan mudah. *Source code* dapat dilihat pada Gambar 1.21. dibawah ini.



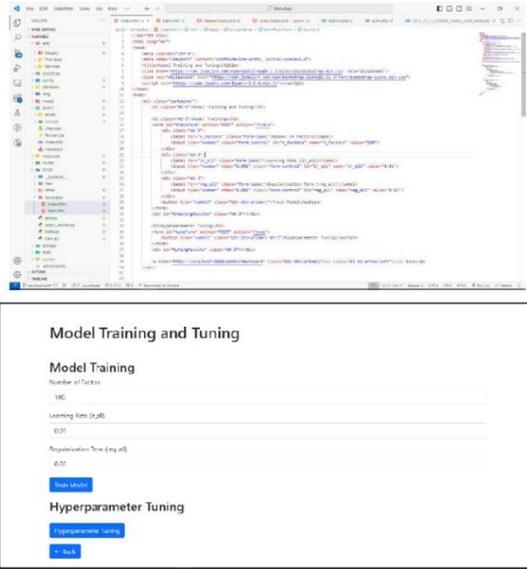
Gambar 1.21. Tampilan *Source Code* Dashboard Export Dataset

18. Tampilan Dashboard Training Model

Halaman dashboard training model ini dirancang untuk melakukan pelatihan dan penyetelan hiperparameter model untuk membangun sistem rekomendasi. Ketika halaman dimuat, ada formulir untuk pelatihan model dengan input jumlah faktor, learning rate, dan regularization term. Formulir ini memungkinkan pengguna untuk mengirimkan permintaan pelatihan model dengan menggunakan metode POST ke endpoint `/train`. Hasil dari pelatihan model akan ditampilkan di bawah formulir ini dalam elemen dengan id `trainingResults`.

Tombol "Back" di bagian bawah halaman mengarahkan pengguna kembali ke dashboard admin untuk navigasi yang mudah. Halaman ini dirancang untuk memudahkan penggunaan dan monitoring proses pelatihan serta

penyetelan model secara interaktif dan efisien dalam konteks pengembangan sistem rekomendasi. *Source code* dapat dilihat pada Gambar 1.23. berikut ini.



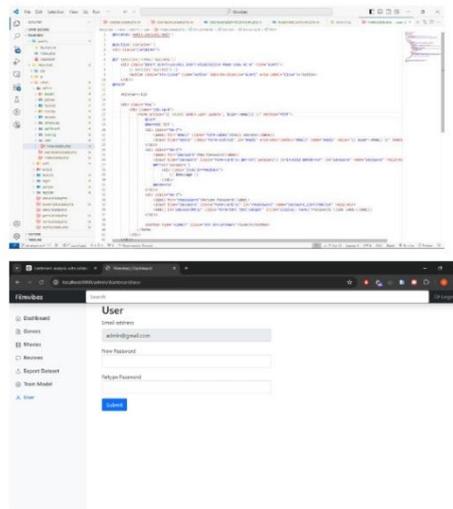
Gambar 1.22. Tampilan *Source Code* Training Model

19. Tampilan Dashboard User

Halaman dashboard user ini digunakan untuk mengelola pengguna admin, khususnya untuk mengubah password admin yang terautentikasi. Saat halaman dimuat, jika ada session flash success yang diset, maka akan muncul pesan sukses dalam bentuk alert di bagian atas halaman. Konten utama halaman ini terdiri dari sebuah form yang memungkinkan admin untuk mengubah password mereka sendiri. Form ini menggunakan metode PUT untuk mengirimkan data ke endpoint yang sesuai dengan route admin.user.update. Form tersebut terdiri dari input untuk alamat email (yang di-set menjadi readonly karena tidak bisa diubah), password baru, dan konfirmasi password baru.

Validasi dilakukan menggunakan Laravel validation rules untuk memastikan bahwa email ada dalam basis data dan password yang dimasukkan memenuhi syarat minimal karakter dan telah dikonfirmasi.

Di bagian JavaScript, jQuery digunakan untuk memastikan bahwa password yang diinputkan dan konfirmasi password cocok sebelum form dikirimkan. Ini membantu mencegah kesalahan pengguna dan memastikan bahwa hanya data yang valid yang dikirimkan untuk memperbarui password. Setelah form dikirimkan dan data diverifikasi serta password berhasil diubah, admin akan diarahkan kembali ke halaman indeks pengguna admin dengan pesan sukses bahwa password telah berhasil diubah. Hal ini memberikan pengalaman pengguna yang intuitif dan aman dalam mengelola akun mereka. *Source code* dapat dilihat pada Gambar 1.24. berikut ini.

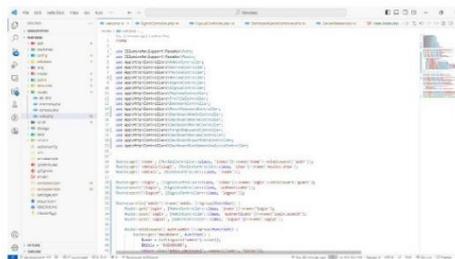


Gambar 1.23. Tampilan *Source Code* Dashboard User

2) Logic

1. Route Endpoint API

Bagian Kode ini mendefinisikan konfigurasi rute API di bagian laravel. Di dalam laravel, didefinisikan untuk mengelompokkan semua rute API. Ada berbagai rute yang diatur untuk menangani operasi otentikasi, pengelolaan pengguna, menampilkan rekomendasi, fitur untuk admin seperti penambahan data film, hapus review, export data, dan pelatihan model. Rute ini mencakup metode HTTP seperti GET, dan POST. Beberapa rute, seperti rute untuk mendapatkan profile pengguna, home, review, dan bookmark dilindungi oleh middleware *auth* dan *verified* untuk memastikan hanya pengguna yang sudah terverifikasi yang dapat mengaksesnya. Selain itu, middleware *admin* digunakan untuk mencegah user yang bukan admin masuk ke dashboard. Fungsi dari setiap rute diarahkan ke metode di dalam berbagai controller yang didefinisikan di dalam direktori controllers. *Source Code* tersebut dapat dilihat pada Gambar 2.1.

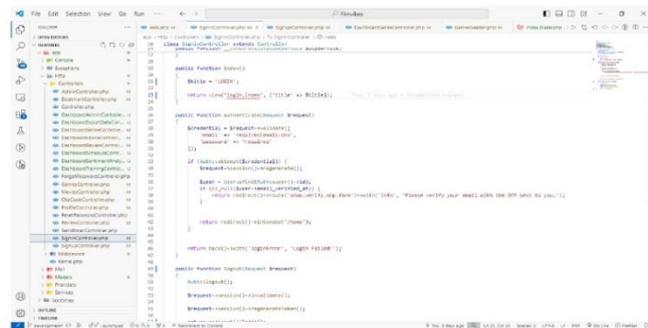


Gambar 2.1. Route Endpoint API

2. Route Login

Bagian ini merupakan fungsi login yang digunakan mengelola proses autentikasi pengguna dalam aplikasi Laravel. Ini mengimpor model User, class Request, layanan OtpCodeService, dan facade Auth. Layanan OtpCodeService digunakan untuk verifikasi kode OTP. Kode ini memiliki beberapa metode.

__construct() menginjeksi OtpCodeService untuk digunakan dalam metode lainnya. Metode index() menampilkan halaman login dengan variabel title 'LOGIN'. Metode authenticate(Request \$request) memvalidasi kredensial pengguna (email dan password). Jika autentikasi berhasil, sesi pengguna diregenerasi. Jika email pengguna belum diverifikasi, pengguna diarahkan ke halaman verifikasi OTP. Jika email telah diverifikasi, pengguna diarahkan ke halaman utama. Jika autentikasi gagal, pengguna dikembalikan ke halaman login dengan pesan kesalahan. Metode logout(Request \$request) mengelola proses logout. Pengguna dikeluarkan dari sesi, sesi dinonaktifkan, dan token sesi diregenerasi. Setelah itu, pengguna diarahkan kembali ke halaman login. Controller ini memastikan hanya pengguna terverifikasi yang dapat mengakses aplikasi. *Source Code* tersebut dapat dilihat pada Gambar 2.2.

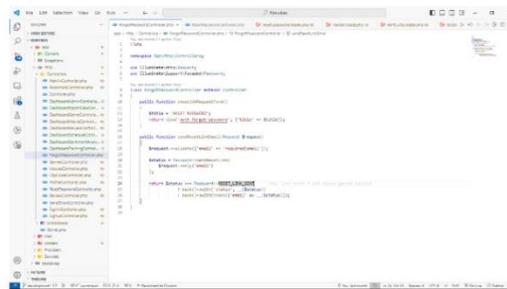


Gambar 2.2. Route Login

3. Route Forgot Password

Kode berfungsi mengatur proses reset kata sandi. Controller ini bernama 'ForgotPasswordController' dan memiliki dua metode utama. Metode pertama adalah 'showLinkRequestForm()'. Metode ini digunakan untuk menampilkan formulir permintaan tautan reset kata sandi. Di dalam metode ini, variabel 'Stitle' diinisialisasi dengan nilai 'RESET PASSWORD' dan kemudian halaman view 'auth.forgot-password' ditampilkan dengan data 'title' yang diberikan.

Metode kedua adalah `sendResetLinkEmail(Request \$request)`. Metode ini menerima objek `Request` yang berisi data permintaan dari pengguna. Pertama, metode ini memvalidasi bahwa field `email` wajib diisi dan harus dalam format email yang valid. Setelah validasi berhasil, metode ini memanggil `Password::sendResetLink()` untuk mengirimkan tautan reset kata sandi ke alamat email yang diberikan. Jika pengiriman tautan berhasil, pengguna akan diarahkan kembali ke halaman sebelumnya dengan pesan status yang sesuai. Jika gagal, pengguna akan diarahkan kembali dengan pesan kesalahan yang sesuai. *Source code* kode ini dapat dilihat pada Gambar 2.3 berikut ini.



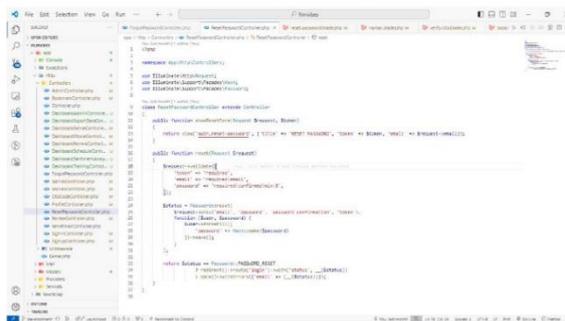
Gambar 2.3. *Controller* Lupa Password

4. Route Reset Password

Kode ini punya dua fungsi utama: `showResetForm` dan `reset`. Fungsi `showResetForm` digunakan untuk menampilkan form reset password. Fungsi ini menerima `Request` dan `token` sebagai parameter. Di dalam fungsi ini, form reset password ditampilkan dengan mengembalikan view `auth.reset-password`, yang juga mengirimkan beberapa data ke view tersebut, yaitu judul 'RESET PASSWORD', token, dan email yang diterima dari request.

Fungsi kedua, `reset`, digunakan untuk memproses permintaan reset password. Fungsi ini pertama-tama memvalidasi input dari request untuk memastikan bahwa token, email, dan password yang dikonfirmasi serta memiliki panjang minimal 8 karakter sudah benar. Setelah validasi, fungsi ini

memanggil metode `reset` dari `Password` facade yang mengelola proses reset password menggunakan data yang diterima dari request. Di dalam callback yang diberikan ke metode `reset`, password baru pengguna di-hash dan disimpan ke dalam database. Jika proses reset password berhasil, pengguna akan diarahkan ke halaman login dengan status berhasil. Sebaliknya, jika gagal, pengguna akan diarahkan kembali ke halaman sebelumnya dengan pesan error yang sesuai. *Source code* tersebut dapat dilihat pada Gambar 2.4.



Gambar 2. 4. *Source Code* Reset Password

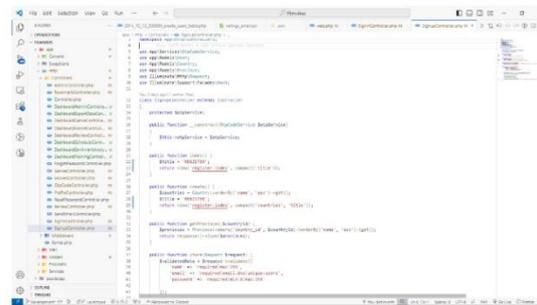
5. Route Register

Route ini bertanggung jawab untuk mengelola proses pendaftaran pengguna baru. Konstruktors dari controller ini menerima instance dari `OtpCodeService`, yang digunakan untuk mengirimkan OTP (One Time Password) setelah pengguna berhasil mendaftar. Fungsi `index` digunakan untuk menampilkan halaman registrasi dengan judul 'REGISTER'. Fungsi `create` mengambil daftar negara dari model `Country` dan mengurutkannya berdasarkan nama untuk ditampilkan dalam halaman registrasi. Fungsi ini juga mengembalikan halaman registrasi dengan judul 'REGISTER' dan data negara.

Fungsi `getProvinces` digunakan untuk mengembalikan daftar provinsi berdasarkan `country_id` yang diterima dari request dalam format JSON. Fungsi `store` adalah fungsi yang menangani penyimpanan data pengguna

setelah validasi data yang diterima dari request. Data yang validasi termasuk nama, email, dan password dengan aturan tertentu. Password dari pengguna di-hash menggunakan `Hash::make()` sebelum data pengguna disimpan ke dalam tabel `users`. Setelah penyimpanan data pengguna berhasil, OTP dikirimkan menggunakan `OtpCodeService` yang telah disiapkan sebelumnya. Pengguna kemudian diarahkan ke halaman login dengan pesan sukses untuk memeriksa email mereka untuk mendapatkan kode OTP guna memverifikasi email mereka.

Controller ini mengintegrasikan validasi input, penggunaan service eksternal (`OtpCodeService`), serta penggunaan model Laravel untuk manipulasi data pengguna dan negara. *Source code* dapat dilihat pada Gambar 2.5. berikut.



Gambar 2. 5. *Source Code* Route Register

6. Route Verify OTP

Route ini terhubung ke Controller OtpCodeController yang bertugas mengelola verifikasi dan pengiriman ulang OTP. Konstruktora menerima OtpCodeService untuk operasi OTP. showVerifyOtpForm menampilkan halaman verifikasi jika pengguna belum terverifikasi. verifyOtp memvalidasi OTP, memperbarui status verifikasi email pengguna, dan menghapus kode OTP yang digunakan. Fungsi resendOtp mengirim ulang OTP untuk pengguna yang

sudah login. Semua fungsi memanfaatkan otentikasi Laravel untuk memastikan pengguna yang sesuai dan mengarahkan ke halaman yang sesuai setelah operasi berhasil atau gagal. *Source Code* tersebut dapat dilihat pada Gambar 2.6.



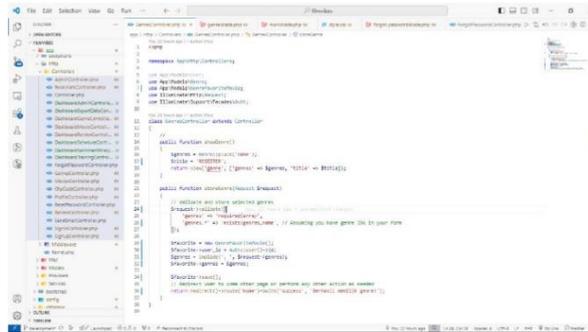
Gambar 2.6. Route Verify OTP

7. Route Choose Genre

Route ini terhubung ke Controller GenresController dalam Laravel yang bertanggung jawab untuk mengelola proses terkait genre film. Fungsi showGenre digunakan untuk menampilkan halaman dengan daftar genre yang diambil dari model Genre menggunakan metode pluck, yang mengambil nama-nama genre dan menampilkannya agar pengguna bisa memilih genre favorit mereka. Hal ini dilakukan untuk nantinya dapat dijadikan media rekomendasi film bagi pengguna baru.

Fungsi storeGenre mengelola penyimpanan genre yang dipilih oleh pengguna. Fungsi ini pertama-tama memvalidasi input dari request untuk memastikan bahwa genres yang dipilih merupakan array dan setiap elemen di dalamnya ada dalam tabel genres. Setelah validasi berhasil, controller membuat instance baru dari model GenreFavoriteMovie, menetapkan user_id dari pengguna yang saat ini login menggunakan Auth::user()->id, dan menggabungkan genre-genre yang dipilih menjadi satu string yang dipisahkan dengan koma. Data tersebut disimpan ke dalam database dan pengguna

diarahkan kembali ke halaman utama (home) dengan pesan sukses bahwa pemilihan genre berhasil. *Source Code* tersebut dapat dilihat pada Gambar 2.7.



Gambar 2.7. Route Choose Genre

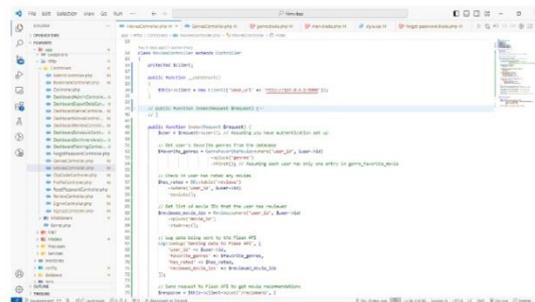
8. Route Home

Route ini terkoneksi ke Controller 'MoviesController' dalam Laravel mengelola tiga fungsi utama terkait dengan aplikasi film: 'index', 'show', dan konstruktor. Konstruktor menginisialisasi objek 'Client' dari Guzzle untuk berkomunikasi dengan API Flask yang berjalan di 'http://127.0.0.1:5000'. Fungsi 'index' digunakan untuk menampilkan halaman utama ('home') dengan rekomendasi film untuk pengguna yang sedang login. Fungsi ini mengambil genre favorit pengguna dari tabel 'GenreFavoriteMovie', mengecek apakah pengguna telah memberikan ulasan pada film, dan mengambil daftar ID film yang telah diulas oleh pengguna dari tabel 'Review'. Data ini kemudian dikirim sebagai payload JSON ke API Flask ('/recommend') untuk mendapatkan rekomendasi film. Selain rekomendasi film, fungsi ini juga mengambil daftar film populer berdasarkan rating ('vote_average') dan film terbaru berdasarkan tanggal rilis ('release_date').

Fungsi 'show' digunakan untuk menampilkan detail dari satu film berdasarkan 'slug' yang diterima sebagai parameter. Fungsi ini mengambil data film berdasarkan 'slug', lalu mengambil lima film lain dari negara produksi

yang sama dengan film yang sedang ditampilkan, diurutkan berdasarkan rating ('vote_average') secara menurun.

Kontroler ini memanfaatkan model Laravel untuk mengakses dan mengelola data film dan ulasan, serta menggunakan Guzzle untuk berkomunikasi dengan API Flask untuk mendapatkan rekomendasi film berdasarkan data pengguna yang disediakan. *Source Code* tersebut dapat dilihat pada Gambar 2.8.



Gambar 2.8. Route Home

9. Route Profile

Route ini terhubung ke Controller 'ProfileController' dalam Laravel yang bertanggung jawab untuk mengelola profil pengguna. Fungsi 'show' digunakan untuk menampilkan halaman profil pengguna dengan informasi dari pengguna yang sedang login, seperti nama dan judul 'PROFILE'. Fungsi ini menggunakan 'Auth::user()' untuk mengambil data pengguna yang sedang login. Fungsi 'update' mengelola pembaruan data profil pengguna berdasarkan data yang diterima dari request. Fungsi ini pertama-tama memvalidasi input yang diterima, memastikan bahwa nama pengguna diisi dan berupa string dengan panjang maksimal 255 karakter, serta opsional validasi foto profil dengan tipe file yang diizinkan (jpeg, png, jpg, gif) dan maksimum ukuran file 2MB. Setelah validasi berhasil, data nama pengguna diperbarui dan jika ada file foto

profil yang diunggah, foto profil lama dihapus dari penyimpanan dan foto profil yang baru diunggah disimpan. *Source Code* tersebut dapat dilihat pada Gambar 2.9.



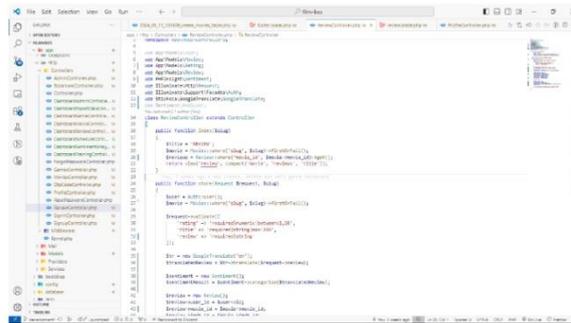
Gambar 2.9. Route Profile

10. Route Review

Route ini terhubung ke Controller 'ReviewController' yang mengelola proses terkait ulasan film. Fungsi 'index' digunakan untuk menampilkan halaman ulasan dengan informasi film berdasarkan 'slug' yang diterima sebagai parameter. Fungsi ini mengambil data film dari model 'Movies' berdasarkan 'slug', kemudian mengambil semua ulasan yang terkait dengan film tersebut dari model 'Review'. Data ini kemudian ditampilkan di halaman 'review' bersama dengan judul 'REVIEW'. Fungsi 'store' bertanggung jawab untuk menyimpan ulasan baru yang dikirimkan oleh pengguna untuk film tertentu. Fungsi ini pertama-tama mengambil pengguna yang sedang login menggunakan 'Auth::user()' dan mengambil data film berdasarkan 'slug'. Data yang dikirim dari request kemudian divalidasi untuk memastikan bahwa rating, judul ulasan, dan isi ulasan telah diisi dengan benar.

Setelah validasi berhasil, isi ulasan diterjemahkan ke bahasa Inggris menggunakan Google Translate dan kemudian dianalisis sentimennya menggunakan library Sentiment untuk menentukan label sentimen ulasan tersebut. Setelah itu, sebuah objek 'Review' baru dibuat dan disimpan ke dalam

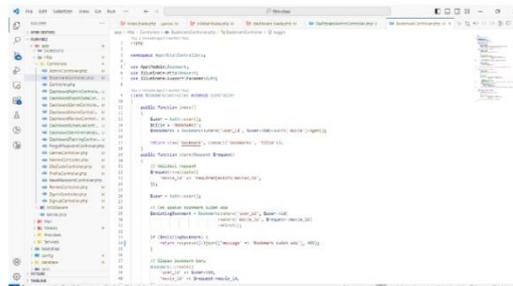
database dengan informasi pengguna, film, rating, judul, isi ulasan, dan label sentimen. Selain itu, rating pengguna juga disimpan dalam tabel 'Rating' untuk merekam rating yang diberikan oleh pengguna terhadap film tersebut. *Source Code* tersebut dapat dilihat pada Gambar 2.10.



Gambar 2.10. Route Review

11. Route Bookmark

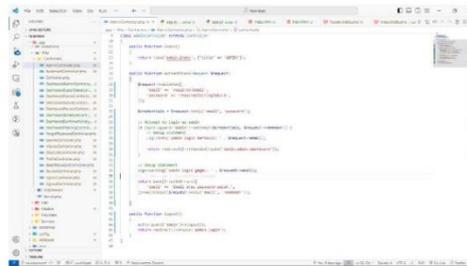
Route ini terhubung ke Controller BookmarkController yang mengelola penandaan film oleh pengguna dengan tiga fungsi utama. Fungsi index menampilkan daftar bookmark pengguna dengan data yang terhubung dengan model Movie. Fungsi store menyimpan bookmark baru setelah memvalidasi keberadaan film dan memastikan tidak ada duplikasi bookmark. Fungsi toggle memungkinkan pengguna untuk menambah atau menghapus bookmark dalam satu permintaan, sementara check memeriksa keberadaan bookmark untuk film tertentu. Controller ini memanfaatkan otentikasi Laravel dan model untuk mengelola data bookmark secara efisien. *Source code* tersebut dapat dilihat pada Gambar 2.11.



Gambar 2.11. Route Bookmark

12. Route Admin Login

Pada Route ini memiliki fungsi `authenticate` yang akan memproses proses otentikasi administrator. Admin diminta untuk memasukkan email dan kata sandi, yang kemudian divalidasi untuk memastikan email valid dan kata sandi memiliki panjang minimal 8 karakter. Setelah itu, fungsi mencoba untuk melakukan login sebagai administrator menggunakan `Auth::guard('admin')->attempt()`. Jika otentikasi berhasil, sebuah pesan log info dicatat dengan email administrator yang berhasil login, dan pengguna diarahkan ke halaman yang dimaksud (dalam kasus ini, dashboard admin). Jika otentikasi gagal, pesan log warning dicatat dengan email administrator yang gagal login, dan pengguna diarahkan kembali ke halaman sebelumnya dengan pesan error bahwa email atau kata sandi yang dimasukkan salah. Fungsi `logout` digunakan untuk proses logout administrator, yang menghapus sesi administrator dan mengarahkan pengguna kembali ke halaman login administrator. Controller ini memanfaatkan fitur otentikasi Laravel dengan penjaga (`guard`) khusus untuk administrator (`admin`), memastikan bahwa proses login dan logout administrator dapat dilakukan dengan aman dan efisien. *Source code* ini dapat dilihat pada Gambar 2.12. berikut ini.



Gambar 2.12. Route Admin Login

13. Route Admin Dashboard Genre

Route ini terhubung ke Controller DashboardGenreController dalam Laravel bertanggung jawab untuk mengelola data genre pada dashboard administrator. Fungsi index menampilkan daftar genre yang diurutkan berdasarkan nama secara ascending, yang kemudian ditampilkan melalui view admin.genres.index. Fungsi create menampilkan formulir untuk membuat genre baru melalui view admin.genres.create. Fungsi store menyimpan genre baru setelah memvalidasi bahwa nama genre diperlukan (required) melalui request. Jika validasi berhasil, genre baru dibuat dan pengguna diarahkan kembali ke halaman indeks genre administrator dengan pesan sukses. Fungsi edit menampilkan formulir untuk mengedit genre yang dipilih melalui view admin.genres.edit, dengan genre yang sudah ada dikirimkan sebagai data. Fungsi update memperbarui informasi genre yang dipilih setelah memvalidasi bahwa nama genre diperlukan (required) melalui request. Jika validasi berhasil, genre diperbarui dan pengguna diarahkan kembali ke halaman indeks genre administrator dengan pesan sukses. Fungsi destroy menghapus genre yang dipilih dari penyimpanan dan mengarahkan pengguna kembali ke halaman indeks genre administrator dengan pesan sukses bahwa genre telah berhasil dihapus. *Source code* dapat dilihat pada Gambar 2.14. berikut ini.



Gambar 2.13. Route Admin Dashboard Genre

14. Route Admin Dashboard Movies

Route ini terhubung ke Controller `DashboardMovieController` yang bertanggung jawab untuk mengelola data film dalam dashboard administrator. Fungsi `index` menampilkan daftar film dengan paginasi dan mengirimkannya ke view untuk ditampilkan. Fungsi `create` menyediakan formulir untuk membuat film baru dengan opsi genre yang diambil dari model `Genre`. Setelah memvalidasi input dari formulir, fungsi `store` menyimpan film baru dan mengarahkan kembali ke halaman indeks dengan pesan sukses.

Selain itu, controller ini juga menyediakan fungsi untuk menampilkan detail film, mengedit, memperbarui, dan menghapus film yang ada. Fungsi `show` menampilkan detail film berdasarkan slug, `edit` menyediakan formulir untuk mengubah informasi film, dan `update` memperbarui film setelah validasi. Fungsi `destroy` menghapus film dari database. Semua operasi ini memanfaatkan fitur-fitur Laravel seperti routing, validasi data, dan interaksi dengan model untuk menjaga integritas dan konsistensi data dalam aplikasi. *Source code* dapat dilihat pada Gambar 2.14. dibawah ini.



Gambar 2.14. Route Admin Dashboard Movies

15. Route Admin Dashboard Review

Route ini terhubung ke Controller 'DashboardReviewController' dalam yang digunakan untuk mengelola review film dalam dashboard administrator. Fungsi 'index' menampilkan daftar review dengan paginasi dan juga menyertakan statistik terkait, seperti total review yang ada, rata-rata rating dari semua review, dan film yang paling banyak direview. Selain itu, controller ini menghitung jumlah review per bulan untuk ditampilkan dalam bentuk grafik atau laporan.

Fungsi-fungsi lainnya termasuk 'edit', yang menampilkan formulir untuk mengedit review yang sudah ada, dan 'update' yang melakukan validasi terhadap input dari formulir dan menyimpan perubahan yang dilakukan. Fungsi 'destroy' menghapus review dari database setelah konfirmasi dari pengguna atau administrator. Semua operasi ini menggunakan fitur-fitur Laravel seperti validasi data, penggunaan model untuk mengakses dan mengelola data, serta pengaturan rute untuk navigasi antar halaman dalam aplikasi. *Source code* kode ini dapat dilihat pada Gambar 2.15. berikut ini.



Gambar 2.15. Route Admin Dashbard Review

16. Route Admin Dashboard Export Dataset

Kode ini berada di Controller 'DashboardExportDataController' yang bertanggung jawab untuk mengelola proses ekspor data terkait film, ulasan, dan peringkat ke dalam format file CSV. Fungsi 'exportReviews', 'exportMovies', dan 'exportRatings' masing-masing mengambil data dari model Review, Movies, dan Rating, lalu menyimpannya dalam file CSV di dalam direktori penyimpanan aplikasi. Setelah selesai, pengguna diarahkan kembali ke halaman admin dengan pesan sukses yang menandakan bahwa proses ekspor telah berhasil dilakukan. Fungsi 'scheduleExport' memungkinkan administrator untuk menjadwalkan ekspor data dengan memvalidasi tanggal, waktu, dan jenis dataset yang akan diekspor. Data jadwal disimpan dalam model Schedule, dan perintah Artisan digunakan untuk menjalankan proses ekspor data sesuai dengan jadwal yang ditentukan. Sementara itu, fungsi 'deleteSchedule' digunakan untuk menghapus jadwal ekspor yang sudah ada berdasarkan ID yang diberikan, dengan pengalihan kembali ke halaman admin setelah berhasil dihapus. Semua fitur ini dirancang untuk memberikan fleksibilitas dan efisiensi dalam manajemen data administratif pada aplikasi Laravel. *Source code* kode ini dapat dilihat pada Gambar 2.16. dibawah ini.

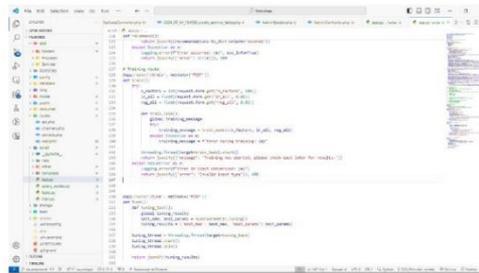


Gambar 2. 16. Route Admin Dashboard Export Dataset

17. Route Admin Dashboard Train Model

Route ini akan mengalihkan ke halaman website yang dibuat oleh aplikasi flask yang berfungsi untuk melakukan pelatihan model dengan menggunakan data dari file CSV yang tersimpan di dalam direktori aplikasi. Fungsi 'train_models' melakukan beberapa tugas utama, termasuk pelatihan algoritma Collaborative Filtering menggunakan SVD (Singular Value Decomposition) dari pustaka Surprise untuk merekomendasikan film berdasarkan rating pengguna. Selain itu, fungsi ini juga melatih sebuah model klasifikasi Multinomial Naive Bayes untuk analisis sentimen berdasarkan ulasan yang ada. Proses ini melibatkan penggunaan TF-IDF untuk vektorisasi teks dan pemrosesan teks sederhana seperti lemmatisasi dan penghapusan kata-kata stop words.

Setelah pelatihan selesai, model Collaborative Filtering, klasifikasi sentimen, dan vektorisasi TF-IDF disimpan dalam file .pkl untuk digunakan kembali. Proses ini mendukung aplikasi yang lebih kompleks dengan integrasi analisis sentimen dan rekomendasi berbasis pengguna. Hasilnya, penggunaan pustaka seperti Surprise dan integrasi dengan alat pemrosesan bahasa alami seperti NLTK menunjukkan pendekatan yang komprehensif dalam pengelolaan dan analisis data untuk sistem rekomendasi film. *Source code* ini dapat dilihat pada Gambar 2.17. dibawah ini.



Gambar 2. 17. Route Training Model

18. Route Admin Dashboard User

Route ini terhubung ke controller `DashboardAdminController` yang mengelola tindakan terkait administrator dalam aplikasi dengan memanfaatkan Laravel. Fungsi `index()` digunakan untuk menampilkan halaman utama administrator yang sedang login saat ini. Ini dilakukan dengan mengambil informasi pengguna yang diotorisasi menggunakan `Auth::user()` dan kemudian mengirimkan data tersebut ke tampilan yang sesuai untuk ditampilkan kepada pengguna. Fungsi `update()` bertanggung jawab untuk memperbarui kata sandi administrator. Proses dimulai dengan validasi input yang memastikan email dan password baru memenuhi persyaratan yang ditetapkan. Setelah validasi berhasil, pencarian dilakukan untuk mencocokkan email yang diberikan dengan catatan admin yang ada dalam basis data. Jika admin ditemukan, password baru di-hash menggunakan `Hash::make()` dan disimpan kembali ke basis data. Informasi ini juga dicatat dalam log aplikasi untuk jejak dan audit. Setelah pembaruan berhasil, pengguna diarahkan kembali ke halaman index dengan pesan sukses yang menyatakan bahwa password telah berhasil diperbarui. *Source code* dapat dilihat pada Gambar 2.18. dibawah ini.



Gambar 2. 18. Route Admin Dashboard User

E. Lampiran 5 Sertifikat HKI


REPUBLIK INDONESIA
KEMENTERIAN HUKUM DAN HAK ASASI MANUSIA

SURAT PENCATATAN CIPTAAN

Dalam rangka perlindungan ciptaan di bidang ilmu pengetahuan, seni dan sastra berdasarkan Undang-Undang Nomor 28 Tahun 2014 tentang Hak Cipta, dengan ini menerangkan:

Nomor dan tanggal permohonan : EC00202467530, 18 Juli 2024

Pencipta

Nama : **Ramanda Kholisandra**

Alamat : Jalan Ahmad Yani Utara Gang Gianti Timur 1 No. 28, Kelurahan Mulyoharjo Kec.Pemalang Kab.Pemalang Provinsi Jawa Tengah 52313, Pemalang, Pemalang, Jawa Tengah, 52313

Kewarganegaraan : Indonesia

Pemegang Hak Cipta

Nama : **Pusat Penelitian dan Pengabdian Masyarakat (P3M) Politeknik Harapan Bersama**

Alamat : Jalan Mataram No. 9, Pesurungan Lor, Kecamatan Margadana 52142, Margadana, Tegal, Jawa Tengah 52142

Kewarganegaraan : Indonesia

Jenis Ciptaan : **Program Komputer**

Judul Ciptaan : **APLIKASI REKOMENDASI FILM ASIA BERDASARKAN SENTIMEN ANALISIS MENGGUNAKAN COLLABORATIVE FILTERING BERBASIS WEBSITE**

Tanggal dan tempat diumumkan untuk pertama kali di wilayah Indonesia atau di luar wilayah Indonesia : 18 Juli 2024, di Tegal

Jangka waktu perlindungan : Berlaku selama 50 (lima puluh) tahun sejak Ciptaan tersebut pertama kali dilakukan Pengumuman.

Nomor pencatatan : 000642882

adalah benar berdasarkan keterangan yang diberikan oleh Pemohon.
Surat Pencatatan Hak Cipta atau produk Hak terkait ini sesuai dengan Pasal 72 Undang-Undang Nomor 28 Tahun 2014 tentang Hak Cipta.

a.n. MENTERI HUKUM DAN HAK ASASI MANUSIA
DIREKTUR JENDERAL KEKAYAAN INTELEKTUAL
u.b
Direktur Hak Cipta dan Desain Industri


IGNATIUS M.T. SILALAH
NIP. 196812301996031001



Disclaimer:
Dalam hal pemohon memberikan keterangan tidak sesuai dengan surat pernyataan, Menteri berwenang untuk mencabut surat pencatatan permohonan.

F. Lampiran 6 Lembar Bimbingan



**SARJANA TERAPAN TEKNIK INFORMATIKA
POLITEKNIK HARAPAN BERSAMA**

LEMBAR BIMBINGAN TUGAS AKHIR

Nama : Ramanda Kholisandra
Nim : 20090015
No. Ponsel : 085292326504
Judul TA : Aplikasi Rekomendasi Film Asia Berdasarkan Sentimen Analisis Menggunakan Collaborative Filtering Berbasis Website
Dosen Pembimbing I : Dwi Intan Afidah, S.T., M.Kom.

| No | Tanggal | Pemeriksaan | Perbaikan yang perlu dilakukan | Paraf Pembimbing |
|----|---------------|------------------|---|------------------|
| 1. | 5 April 2024 | Use Case Diagram | Melanjutkan pembangunan aplikasi dan belajar algoritma yang akan digunakan | |
| 2. | 19 April 2024 | Produk | Implementasikan desain ke source code dan implementasikan algoritma ke source code | |
| 3. | 3 Mei 2024 | Produk | Melakukan evaluasi perormfma model dan Mulai development tampilan website | |
| 4. | 23 Mei 2024 | Produk | Diperjelas bisnis proses aplikasi dan Melanjutkan development fitur yang diperlukan | |
| 5. | 4 Juli 2024 | Produk | Membuat halaman untuk training model pada sisi flask | |
| 6. | 12 Juli 2024 | Produk | Menampilkan film terbaru pada menu | |



**SARJANA TERAPAN TEKNIK INFORMATIKA
POLITEKNIK HARAPAN BERSAMA**

| | | | | |
|----|---------------|---------|--|--|
| 7. | 16 Juli 2024 | Laporan | <ul style="list-style-type: none">• tujuan dan manfaat diubah agar lebih spesifik• tabel diubah agar kelebihan dan kekurangan ditulis dalam kolom hasil saja• nama tabel diubah di atas kiri bukan dibawah tabel | |
| 8. | 17 Juli 2024 | Laporan | <ul style="list-style-type: none">• Penambahan sitasi• Penambahan isi kesimpulan | |
| 9. | 19 Juli 2024. | Laporan | Acc | |

Tegal, 17 Juli 2024
Dosen Pembimbing I

Dwi Intan Afidah, S.T., M.Kom.
NIPY. 11.020.470



**SARJANA TERAPAN TEKNIK INFORMATIKA
POLITEKNIK HARAPAN BERSAMA**

LEMBAR BIMBINGAN TUGAS AKHIR

Nama : Ramanda Kholisandra
Nim : 20090015
No. Ponsel : 085292326504
Judul TA : Aplikasi Rekomendasi Film Asia Berdasarkan Sentimen Analisis Menggunakan Collaborative Filtering Berbasis Website
Dosen Pembimbing II : Ardi Susanto, S.Kom., M.Cs.

| No | Tanggal | Pemeriksaan | Perbaikan yang perlu dilakukan | Paraf Pembimbing |
|----|-----------|----------------------------|---|------------------|
| 1. | 3-9-2024 | Konsultasi Perubahan Judul | Pemklaran logika pada Sistem | |
| 2. | 2-5-24 | Aplikasi - Rancangan | - Penambahan fitur - index | |
| 3 | 19-6-2024 | Aplikasi | - Selesaikan aplikasi - Buat laporan | |



**SARJANA TERAPAN TEKNIK INFORMATIKA
POLITEKNIK HARAPAN BERSAMA**

| No | Tanggal | Pemeriksaan | Perbaikan yang perlu dilakukan | Paraf Pembimbing |
|----|---------|-------------|--------------------------------|---|
| | | | |  |

Tegal, 17 Juli 2024
Dosen Pembimbing II



Ardi Susanto, S.Kom., M.Cs.
NIPY. 03.021.490