

Implementasi *The Movie DataBase* API Untuk Sistem Informasi Film Berbasis *Mobile*

Toba Amiruddin Sitorus^{1*}, Ledy Elsera Astrianty²

Informatika, Universitas Teknologi Yogyakarta, Yogyakarta, Indonesia

*e-mail *Corresponding Author*: 5210411207.toba@student.uty.ac.id

Abstract

The vast selection of films frequently makes it challenging for users to choose a movie that aligns with their preferences, impacting the overall user experience. This study aims to develop a mobile-based film information application integrated with the The Movie DataBase (TMDB) Application Programming Interface (API), providing personalized movie recommendations tailored to user preferences. The development method follows the Waterfall approach, which includes requirement analysis, design, implementation using Flutter, Flask, MySQL, and black-box testing. Key features of the application include genre-based movie recommendations, movie search, synopsis display, and bookmark and favorites functionality. The testing results show that the application successfully provides users with a fast, accurate, and interactive experience, effectively helping them find movies according to their preferences.

Keywords: Information System; Film; TMDB; Mobile

Abstrak

Banyaknya pilihan film yang tersedia kerap menyulitkan pengguna dalam menentukan film yang cocok dengan preferensi mereka, ini berpengaruh pada pengalaman pengguna yang tidak optimal. Penelitian ini bertujuan untuk mengembangkan aplikasi informasi film berbasis mobile yang mengintegrasikan *The Movie DataBase* (TMDB) *Application Programming Interface* (API), guna menyediakan rekomendasi film yang disesuaikan dengan preferensi pengguna. Metode pengembangan menggunakan pendekatan Waterfall yang mencakup analisis kebutuhan, desain, implementasi menggunakan Flutter, Flask, dan MySQL, serta pengujian menggunakan metode black-box testing. Fitur utama aplikasi meliputi rekomendasi film berbasis genre, pencarian film, tampilan sinopsis, serta bookmark dan favorit. Hasil pengujian menunjukkan bahwa aplikasi ini berhasil memberikan pengalaman pengguna yang cepat, akurat, dan interaktif, serta efektif bagi pengguna untuk menemukan film sesuai dengan preferensi mereka.

Kata Kunci: Sistem Informasi; Film; TMDB; Mobile

1. Pendahuluan

Film sebagai media komunikasi massa memiliki dampak bagi masyarakat. Film merupakan bentuk pesan yang terdiri dari tanda-tanda dan simbol yang membentuk sistem dan makna tertentu, hal ini dapat ditafsirkan berbeda oleh berbagai orang berdasarkan dari referensi dan kemampuan berpikir orang tersebut [1]. Film menjadi semakin populer seiring perjalanan waktu. Perkembangan dalam popularitas ini diikuti oleh mulainya ketertarikan masyarakat dalam menonton film. Namun demikian dengan perkembangan pesat di dunia perfilman ini, masyarakat juga membutuhkan informasi untuk menentukan film apa yang akan ditonton yang menjadi latar belakang penelitian ini [2]. Dengan banyaknya pilihan film yang terus bertambah setiap tahun, pengguna sering kali membutuhkan sistem yang mampu menyaring informasi secara efisien. Integrasi dengan TMDB API memungkinkan aplikasi ini menyajikan data yang relevan, terkini, dan terpercaya, sehingga dapat memenuhi kebutuhan pengguna akan informasi yang lengkap, mulai dari sinopsis hingga rating film.

Klasifikasi genre yang ada saat ini masih tergolong umum karena belum mampu menangkap preferensi genre yang berbeda antara satu penonton dengan lainnya. Misalnya, penonton yang menyukai film action bernuansa komedi memiliki preferensi berbeda dibandingkan dengan penonton yang menyukai komedi bernuansa action. Oleh karena itu,

diperlukan sistem informasi rekomendasi yang dapat menyesuaikan genre sesuai dengan preferensi pengguna [3]. Saat ini, meskipun klasifikasi genre pada platform informasi film sudah tersedia, sistem tersebut masih terlalu umum dan belum mampu menangkap preferensi spesifik penonton, seperti kombinasi genre tertentu.

Setiap individu memiliki preferensi yang beragam dalam memenuhi kebutuhannya, terutama dalam mencari hiburan, seperti menonton film, sebagai cara untuk mengalihkan perhatian dari masalah, kesibukan, dan rutinitas sehari-hari. Kepuasan individu juga dapat dipengaruhi oleh faktor lingkungan di sekitarnya [4]. Untuk mengakomodasi preferensi individu yang beragam, dibutuhkan sistem informasi yang mampu memberikan rekomendasi film sesuai kebutuhan pengguna, baik untuk hiburan maupun tujuan profesional. Dengan integrasi TMDb API, sistem informasi ini menyajikan informasi real-time seperti sinopsis, rating, dan genre, serta dilengkapi fitur personalisasi seperti rekomendasi berdasarkan preferensi, bookmark, dan daftar favorit.

Menurut survei yang telah dilakukan dengan penyebaran kuisioner ke 72 responden dari kalangan pecinta film, hasil survei menunjukkan bahwa sebesar 88,6% orang merasa kebingungan untuk menentukan film apa yang hendak ditonton, sebesar 84% orang telah mengatakan kebutuhan akan film meningkat sewaktu pandemi [5]. Banyaknya pilihan film yang tersedia, membuat pengguna sering merasa kesulitan dalam menentukan film untuk ditonton yang sesuai dengan minat mereka masing-masing. Sistem informasi ini juga bertujuan untuk mengembangkan aplikasi berbasis mobile yang memberikan rekomendasi film yang disesuaikan dengan preferensi individu, serta menyediakan fitur seperti *bookmark* dan daftar favorit.

2. Tinjauan Pustaka

Berdasarkan penelitian yang dilakukan oleh Diana dan Irfan Ardianto dalam karya mereka berjudul "Aplikasi Katalog Trailer Film Cinema 21 Berbasis Android", penulis berupaya menyajikan berbagai informasi, seperti teks, gambar, katalog film, serta data bioskop. Penelitian ini juga hanya memberikan informasi umum terkait teks trailer film [6].

Penelitian yang dilakukan oleh Sari Noorlima Yanti dan Erni Rihyanti dalam jurnal berjudul "Penerapan Rest API untuk Sistem Informasi Film Secara Daring" mengungkapkan bahwa sistem informasi tidak hanya berfungsi untuk menyajikan data selama masa promosi film, tetapi juga mampu menampilkan informasi seperti rating, pendapatan, dan video pembuatan film [7].

Penelitian lain yang berjudul "Perancangan Sistem Informasi Film Berbasis WEB" oleh Abdul Muni, dan Khairul Ihwan menunjukkan bahwa selain menyediakan informasi tentang genre film, website ini juga berfungsi untuk membantu pengguna dalam memilih jenis file yang akan diunggah [3].

Berdasarkan penelitian dengan judul "Penerapan Rest API Menggunakan Retrofit Untuk Sistem Informasi Film Berbasis Android (Studi Kasus: Sinopsis Film)" oleh Robbie Christover, Sugiyatno, dan Herlawati menjelaskan bahwa aplikasi mobile berbasis Android yang menggunakan Rest API dengan *Retrofit* dapat digunakan untuk mencari sinopsis film [8].

Penelitian berjudul "Pemanfaatan API Dalam Menampilkan Data Dinamis Untuk Sistem Informasi Film" oleh Tri Adi Ningsih dan I Made Widiarth merancang sebuah aplikasi berbasis website yang menyajikan informasi mengenai dunia perfilman, baik domestik maupun internasional. Sistem ini tidak hanya menampilkan judul film, tetapi juga menyediakan sinopsis serta rating film yang diinginkan [9].

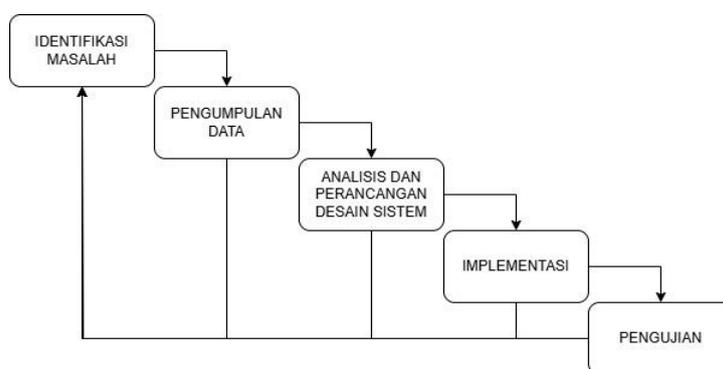
Tabel 1 Hasil Perbandingan Tinjauan Pustaka

Penulis	Fitur
[6]	Informasi film, katalog film, informasi bioskop, dan informasi teks trailer film.
[7]	Informasi film, Informasi promosi film (rating, pendapatan, dan video).
[3]	Informasi film, pencarian film.
[8]	Informasi film, pencarian film.
[9]	Informasi film, pencarian film.
Peneliti	Informasi film, pencarian film, preferensi genre, penanda <i>bookmark</i> dan favorit film, dan katalog film.

Dalam perbandingan dengan jurnal sebelumnya, implementasi sistem informasi ini memberikan pengalaman yang lebih lengkap dan interaktif bagi pengguna. Pertama, yaitu fitur untuk menambahkan preferensi genre film untuk personalisasi rekomendasi film setiap pengguna, lalu fitur menampilkan informasi tentang sinopsis film sehingga pengguna bisa memperoleh informasi lebih lanjut tentang film yang diusulkan. Fitur *bookmark* dan favorit akan memungkinkan pengguna untuk menandai suatu film atau menyimpan film favorit mereka.

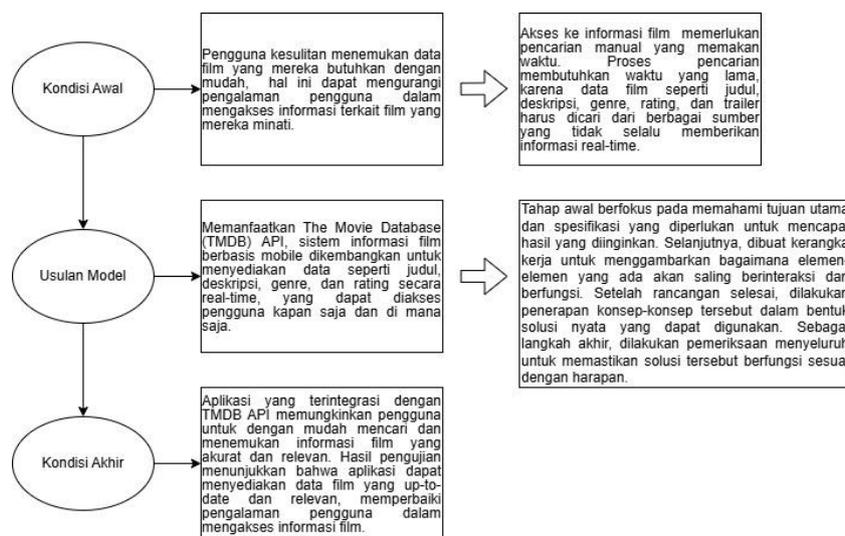
3. Metodologi

Metode *Waterfall* merupakan pengembangan perangkat lunak yang bersifat linier dan terstruktur, di mana prosesnya dibagi menjadi tahapan-tahapan yang jelas dan terdefinisi dengan baik. Satu per satu tahapan harus diselesaikan sepenuhnya sebelum melanjutkan ke tahap berikutnya [10]. Tahapan pengembangan sistem informasi dilakukan melalui serangkaian tahap, mulai dari analisis kebutuhan, perancangan sistem, implementasi, hingga pengujian. Setiap tahapan itu saling berkaitan dan memiliki tujuan untuk memastikan bahwa sistem yang dibangun memenuhi kebutuhan pengguna serta dapat beroperasi secara efektif dan efisien.



Gambar 1 Metode Waterfall

Penelitian ini dimulai dengan mengidentifikasi kondisi awal yang menggambarkan permasalahan yang dihadapi pengguna. Selanjutnya, penelitian mengusulkan model yang relevan sebagai solusi untuk permasalahan tersebut. Lalu, penelitian ini bertujuan untuk mencapai kondisi akhir yang diharapkan, sesuai dengan tujuan yang telah ditentukan.



Gambar 2 Kerangka Penelitian

3.1. Identifikasi Masalah

Industri film saat ini mengalami perkembangan yang sangat pesat, dengan ratusan film baru yang dirilis setiap tahunnya di seluruh dunia. Kemajuan teknologi yang pesat juga telah memungkinkan para penonton untuk dengan mudah mengakses berbagai platform streaming

dan situs web untuk menonton film favorit mereka. Banyaknya pilihan yang tersedia justru membuat penonton merasa bingung dan kesulitan untuk menemukan film yang benar-benar menarik atau sesuai dengan mood mereka. Fitur pencarian yang disediakan oleh platform streaming terkadang tidak cukup membantu dalam mempersempit pilihan, sehingga pengguna harus menghabiskan waktu lama untuk memilih film yang tepat, yang pada akhirnya bisa mengurangi pengalaman menonton mereka.

Selain itu, meskipun banyak film dengan kualitas dan alur cerita yang luar biasa, tidak banyak sistem informasi rekomendasi film yang sesuai dengan preferensi genre pengguna. Sistem yang ada sering kali tidak dapat mempersonalisasi pengalaman menonton berdasarkan genre favorit atau kecenderungan pengguna, sehingga rekomendasi yang diberikan cenderung umum dan kurang relevan. Hal ini menyebabkan banyak film berkualitas tinggi yang mungkin sesuai dengan selera pengguna tidak terlihat, karena sistem rekomendasi tidak dapat menggali potensi preferensi individu dengan baik.

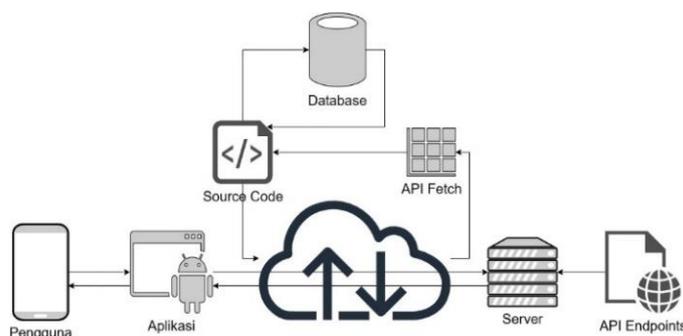
3.2. Pengumpulan Data

Sistem memerlukan data tentang berbagai film, termasuk judul, genre, tahun rilis, sinopsis, dan informasi lainnya yang relevan. Sistem mungkin perlu berintegrasi dengan API eksternal untuk mengimpor data tentang film terbaru, atau informasi lain yang diperlukan. Pemilihan sumber data adalah langkah yang penting, menggunakan API film dari pihak ketiga yang sudah memiliki reputasi sangat baik seperti *The Movie Database* API merupakan langkah yang tepat. API film pihak ketiga umumnya menyediakan beragam informasi film, termasuk judul, genre, pemeran, peringkat, dan ulasan. Selain itu, sistem juga membutuhkan preferensi genre film dari pengguna. Preferensi ini dikumpulkan melalui interaksi pengguna dengan aplikasi dan menjadi data penting untuk mendukung personalisasi rekomendasi film. Informasi ini digunakan untuk menyesuaikan rekomendasi yang diberikan, sehingga aplikasi dapat memberikan hasil yang relevan dengan minat dan kebutuhan pengguna.

3.3. Analisis dan Perancangan Sistem

1) Arsitektur Model

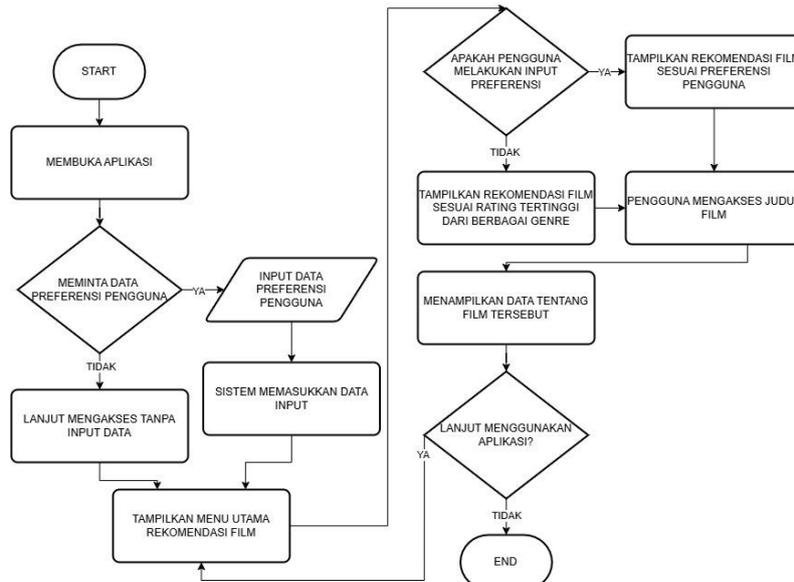
Pada gambar arsitektur model berikut kita dapat melihat bagaimana aplikasi bekerja yaitu *smartphone* sebagai *device* yang digunakan oleh client dalam mengakses aplikasi, akan meminta data ke *web service* yang dituju, *web service* kemudian mengumpulkan data dari api yang disediakan oleh pihak ketiga untuk data tersebut diproses sesuai dengan data preferensi pengguna, lalu diberikan kembali ke *smartphone* pengguna sebagai respon dari permintaan data.



Gambar 3 Arsitektur Model

2) Flowchart Sistem

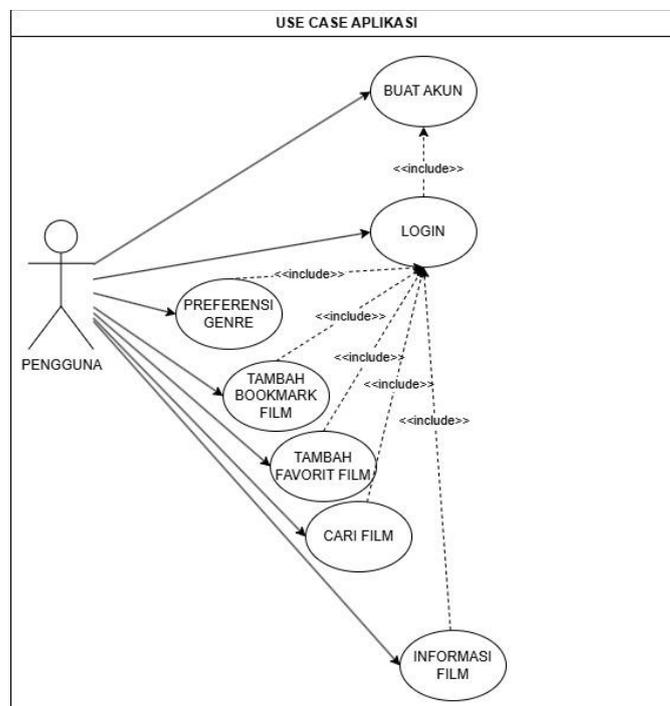
Flowchart adalah gambaran simbolis dari sebuah algoritma atau prosedur untuk menyelesaikan masalah. Dengan *flowchart*, pengguna menjadi lebih mudah memeriksa bagian-bagian yang mungkin tidak sengaja terlewat dalam proses analisis masalah. Selain itu, *flowchart* juga berfungsi sebagai alat komunikasi antara pemrogram dalam tim proyek. *Flowchart* mempermudah pemahaman terhadap alur logika yang kompleks dan panjang [11].



Gambar 4 Flowchart Sistem

3) Use Case Diagram

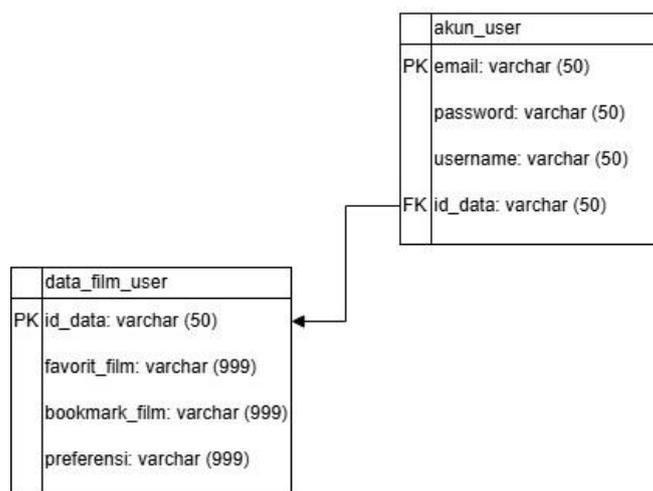
Use case diagram adalah diagram yang menunjukkan interaksi antara aktor dan use case dalam sebuah sistem. Diagram ini berfungsi untuk membantu proses analisis dan desain sistem. Use case merepresentasikan tindakan atau aktivitas yang dilakukan oleh aktor, yang digambarkan dalam bentuk elips horizontal. Aktor sendiri merupakan entitas yang berperan dalam berinteraksi dengan sistem, baik berupa individu maupun organisasi yang saling bertukar informasi. [12]. Use case menggambarkan fungsi-fungsi utama yang disediakan oleh aplikasi dan bagaimana pengguna menggunakan aplikasi tersebut untuk mencapai tujuan tertentu.



Gambar 5 Use Case Diagram

4) Desain Struktur Tabel

Connolly, dan Begg berpendapat, database adalah sekumpulan data yang saling terhubung secara logis dan deskriptif, yang dirancang untuk memenuhi kebutuhan penyimpanan informasi dalam suatu organisasi. Database memberikan kemudahan dalam penyimpanan data dengan format yang mandiri, dan dapat beradaptasi dengan baik. Hal ini karena database didefinisikan secara terpisah dari program aplikasi yang menggunakannya, sehingga pengembangan lingkup database tidak mempengaruhi program-program yang mengaksesnya [13]. Desain tabel database ini dirancang untuk mengelola informasi akun pengguna dan data terkait film yang disukai, *bookmark*, serta preferensi film mereka.



Gambar 6 Desain Tabel

3.4. Implementasi

Aplikasi ini dikembangkan menggunakan Flutter untuk antarmuka pengguna, MySQL sebagai basis data, dan Flask sebagai *backend* untuk mengelola logika dan komunikasi data.

Flutter ialah *software development kit* (SDK) yang dikembangkan oleh Google dengan lisensi open source, digunakan untuk membuat aplikasi mobile berkinerja tinggi untuk iOS dan Android dari satu codebase. Tujuan utamanya adalah untuk memungkinkan pengembang menciptakan aplikasi yang memiliki performa tinggi dan terasa alami di berbagai platform [14]. *Structured Query Language* (SQL) adalah bahasa pemrograman yang berfungsi untuk mengelola, mengakses, dan memodifikasi data dalam sistem basis data relasional. Di sisi lain, MySQL merupakan sistem manajemen basis data (DBMS) yang memanfaatkan SQL sebagai jembatan antara aplikasi perangkat lunak dan database dalam sistem. MySQL juga adalah salah satu DBMS yang paling banyak digunakan saat ini [15]. Flask, sebuah *framework* web berbasis Python, sering dimanfaatkan untuk mengembangkan aplikasi web secara cepat dan efisien. Dengan sifatnya yang ringan dan fleksibel, Flask menjadi pilihan favorit bagi pengembang yang menginginkan kendali penuh terhadap struktur aplikasi web yang mereka buat [16].

3.5. Pengujian

Metode *black box testing* adalah teknik pengujian yang digunakan untuk menemukan kesalahan dalam sebuah sistem aplikasi dengan cara menguji fungsionalitasnya tanpa memperhatikan struktur internal atau detail kode yang digunakan. termasuk kesalahan pada fungsi sistem aplikasi dan fitur-fitur aplikasi yang hilang. Dengan kata lain, *black box testing* merupakan cara untuk menguji fungsionalitas sebuah sistem aplikasi [17]. Pengujian *black box* bertujuan untuk memverifikasi bahwa setiap fungsi dalam aplikasi berjalan dengan benar tanpa memperhatikan implementasi internal. Fokus pengujian terletak pada masukan dan keluaran dari setiap komponen.

4. Hasil dan Pembahasan

4.1. Hasil

Gambar di bawah ini adalah halaman awal ketika pengguna baru mendaftarkan akun di aplikasi ini, atau nantinya dapat diakses lagi jika ingin menambahkan, atau merubah preferensi

genre. Disini jika pengguna menekan *card* genre yang tampil maka genre tersebut akan tercatat sebagai genre preferensi pengguna, dan jika ditekan lagi maka akan menghapus genre tersebut.



Gambar 7 Halaman Preferensi Genre

Pada gambar 8 berikut adalah tampilan awal dari aplikasi, disini adalah bagian yang menampilkan informasi umum akan dunia film secara menyeluruh, yaitu akan rilis, rilis terbaru, populer sekarang ini, populer sepanjang masa, dan rekomendasi khusus pengguna.



Gambar 8 Halaman Utama

Gambar di dibawah ini merupakan isi halaman jika anda menekan tombol lainnya. Halaman ini digunakan untuk menampilkan informasi lebih lengkap dari informasi yang tampil di halaman home. Sebagai contoh, pada contoh halaman berikut merupakan halaman "lainnya" rekomendasi film dari genre yang disukai pengguna, pengguna dapat memilih genre favorit mereka agar dapat ditampilkan di halaman tersebut. Jika tidak ada film yang cocok dengan minat

pengguna, pengguna tinggal menarik widget kebawah untuk me-*refresh* hasil film yang lainnya dari genre yang telah dipilih.



Gambar 9 Halaman Lainnya

Pada gambar 11 berikut, merupakan tampilan informasi film, dengan *header* yang mencakup ikon kembali, nama aplikasi, dan ikon berbagi. Di bawahnya, terdapat poster film, judul, dan informasi dasar seperti rating usia, durasi, genre, asal negara, dan aktor utama.



Gambar 10 Halaman Informasi Film

Pada halaman ini kita bisa melakukan pencarian terhadap judul film yang kita ingin cari, hal ini memudahkan pengguna ketika hendak mengakses judul film yang tidak terdapat pada halaman utama atau hanya sekedar mencari informasi tentang suatu film.



Gambar 11 Halaman Pencarian

Pada gambar berikut kita dapat melihat halaman *favorit*. Pada halaman ini tersimpan data film yang ditandai favorit oleh pengguna, dengan adanya halaman *favorit* ini, pengguna dapat dengan cepat mengakses film-film yang mereka sukai tanpa perlu mencari melalui katalog atau melakukan pencarian ulang.



Gambar 12 Halaman Favorit

Gambar berikut ini adalah halaman *bookmark* didalam aplikasi, halaman ini memungkinkan pengguna untuk menandai film yang mereka sukai atau ingin tonton ulang di masa mendatang, Dengan fitur ini, pengguna dapat dengan cepat melihat dan memilih film yang tersimpan.



Gambar 13 Halaman Bookmark

4.2. Pengujian Sistem

Pada tahap pengujian ini peneliti menggunakan metode *black box*. Metode *Black Box Testing* adalah teknik pengujian yang digunakan untuk menemukan kesalahan dalam sebuah sistem aplikasi dengan cara menguji fungsionalitasnya tanpa memperhatikan struktur internal atau detail kode yang digunakan. Berikut adalah rumus yang digunakan [18]:

Ket:

f = frekuensi berhasil

n = jumlah uji yang dilakukan

p = Persentase

Rumus:

$$P = \frac{f}{n} \times 100 \quad (1)$$

Tabel 2 Hasil Pengujian Black Box

No	Aktivitas	Hasil yang diharapkan	Hasil yang didapat	kesimpulan
1	Mendapatkan <i>card</i> film.	Aplikasi menampilkan berbagai <i>card</i> dari film.	Dapat menampilkan <i>card</i> film.	Sesuai.
2	Masuk ke halaman pemilihan preferensi bagi pengguna yang baru mendaftar	Aplikasi dapat menampilkan halaman pemilihan preferensi	Dapat menampilkan halaman pemilihan preferensi	Sesuai.
3	Masuk ke halaman informasi film ketika <i>card</i> ditekan.	Aplikasi dapat menampilkan informasi tentang film tersebut.	Aplikasi menampilkan informasi film.	Sesuai.
4	Aplikasi mendapatkan rekomendasi film sesuai dengan preferensi genre pengguna.	Sistem informasi menampilkan rekomendasi film berdasarkan preferensi genre pengguna.	Aplikasi berhasil mendapatkan hasil dari preferensi genre pengguna.	Sesuai.
5	Masuk ke halaman lainnya ketika menekan tombol "lainnya" dan	Aplikasi akan berhasil masuk dan	Aplikasi berhasil masuk ke halaman tersebut, dan	Sesuai.

6	menampilkan informasi sesuai <i>section</i> nya. Berpindah dari satu menu ke yang lainnya.	menampilkan informasi yang sesuai. Aplikasi akan berpindah dengan benar.	menampilan informasi yang sesuai. Aplikasi berpindah dengan benar.	Sesuai.
7	Menambahkan film ke daftar favorit.	Aplikasi dapat menambahkan film ke daftar favorit.	Berhasil menambahkan ke daftar favorit.	Sesuai.
8	Menambahkan fim ke daftar <i>bookmark</i> .	Aplikasi menambahkan film ke daftar <i>bookmark</i> .	Berhasil menambahkan ke daftar <i>bookmark</i> .	Sesuai.
9	Melakukan pencarian film.	Aplikasi dapat melakukan pencarian film.	Berhasil mencari film dari input pengguna.	Sesuai.

Dari tabel hasil pengujian di atas, kita dapat menghitung bobot pengujian menggunakan persamaan (1), yaitu:

$$P = \frac{10}{10} \times 100 = 100\%$$

Berdasarkan hasil tersebut, tingkat keberhasilan yang dicapai adalah 100%, ini menandakan aplikasi sudah berjalan sesuai dengan perintah yang diberikan.

4.3. Pembahasan

Pengujian yang dilakukan menunjukkan bahwa fitur-fitur tersebut berfungsi dengan baik, seperti kemampuan aplikasi untuk menampilkan *card* film, memberikan informasi detail tentang film, dan menambahkan film ke daftar favorit atau *bookmark* sesuai preferensi pengguna. Aplikasi ini juga memberikan pengalaman pengguna yang baik dengan menyediakan akses mudah dan cepat yang disajikan secara langsung, dapat membantu pengguna dalam membuat keputusan yang lebih cepat.

Sistem informasi ini menawarkan solusi yang lebih lengkap dan interaktif dibandingkan dengan penelitian-penelitian terdahulu dalam hal pengembangan sistem informasi berbasis *mobile* untuk film. Pada penelitian Sari Noorlima Yanti, dan Erni Rihyanti dalam jurnal "Penerapan Rest API untuk Sistem Informasi Film Secara Daring" [7], disana ia hanya memberikan fitur-fitur dasar seperti pencarian film, informasi tentang *top rated*, *popular*, *now playing*, dan *upcoming*, namun penelitian ini menambahkan elemen penting yang memperkaya pengalaman pengguna. Fitur seperti personalisasi rekomendasi berdasarkan preferensi genre, penambahan sinopsis film, serta penanda *bookmark* dan favorit, menjadikan aplikasi ini lebih efektif dalam memberikan rekomendasi yang relevan dan memperbaiki cara pengguna mengakses dan mengelola informasi film yang mereka pilih. Dengan fitur-fitur tambahan seperti personalisasi berdasarkan preferensi genre dan penanda *bookmark* serta favorit, aplikasi ini tidak hanya memberikan informasi film, tetapi juga memberikan kontrol lebih besar kepada pengguna dalam mengelola pilihan film mereka.

5. Kesimpulan

Penggunaan *The Movie Database* (TMDB) API dalam pengembangan sistem informasi film berbasis *mobile* dapat mempermudah pengguna dalam mengakses informasi film secara *real-time* seperti judul, deskripsi, genre, *rating*, dan *trailer*. Proses pengembangan aplikasi dilakukan dengan metodologi perangkat lunak yang terstruktur, memastikan setiap bagian dari sistem berfungsi sesuai harapan. Pada tahap pengujian juga menunjukkan bahwa data yang disediakan oleh TMDB API relevan dan sesuai kebutuhan pengguna, data yang diberikan oleh TMDB API juga tampil secara konsisten dan akurat, mendukung kebutuhan pengguna akan informasi film yang lebih lengkap.

Setelah Implementasi sistem informasi ini, pengembangan lebih lanjut juga dapat dilakukan dengan memperluas jangkauan pengguna dengan menambahkan versi websitenya. Hal ini bertujuan untuk memperluas lingkup target pengguna, sehingga tidak hanya pengguna *smartphone* yang dapat mengakses informasi tersebut, tetapi juga pengguna perangkat lainnya

seperti laptop, atau komputer desktop. Pengembangan tersebut juga termasuk desain dan implementasi *website* yang responsif untuk memastikan pengalaman pengguna yang optimal.

Daftar Referensi:

- [1] M. R. I. Y. A. Pertiwi, "Analisis Resepsi Interpretasi Penonton terhadap Konflik Keluarga dalam Film "Dua Garis Biru", *Junal AUDIENS*, vol. 1, no. 1, pp. 1-8, 2020.
- [2] A. H. S. Ciaputra, "Rekomendasi Pemilihan Film Dengan Hybrid Filtering Dan K-Nearest Neighbor," *Jurnal Rekayasa Informasi*, vol. 9, no. 2, pp. 101-109, 2020.
- [3] A. I. K. Muni, "Perancangan Sistem Informasi Film Berbasis WEB," *Jurnal Teknik Industri UNISI*, vol. 5, no. 2, pp. 1-5, 2021.
- [4] M. S. Y. Rizky, "Preferensi Penonton Terhadap Film Indonesia," *Communicare*, vol. 4, no. 1, pp. 15-34, 2019.
- [5] J. Lemantara, "Sistem Pendukung Keputusan Penentuan Film Berdasarkan Minat dengan Metode Weighted Product," *Jurnal Riset Sistem Informasi dan Teknik Informatika*, vol. 8, no. 2, pp. 587-600, 2023.
- [6] A. I. Diana., "Aplikasi Katalog Trailer Film Cinema 21 Berbasis Android," *Jurnal Komputer, Informasi, dan Teknologi (KOMITEK)*, vol. 2, no. 2, pp. 461-470, 2022.
- [7] S. R. E. Yanti, "Penerapan Rest API untuk Sistem Informasi Film Secara Daring," *Jurnal Informatika Universitas Pamulang*, vol. 6, no. 1, pp. 195-201, 2021.
- [8] R. S. H. Christover, "Penerapan Rest API Menggunakan Retrofit Untuk Sistem Informasi Film Berbasis Android (Studi Kasus: Sinopsis Film)," *Journal of Students Research in Computer Science*, vol. 3, no. 2, pp. 159-170, 2022.
- [9] T. W. I. Ningsih, "Pemanfaatan Api Dalam Menampilkan Data Dinamis Untuk Sistem Informasi Film," *Jurnal Elektronik Ilmu Komputer Udayana*, vol. 12, no. 1, pp. 31-44, 2023.
- [10] E. R. F. M. D. R. F. Wahyuni, "SDLC Big Bang dan Waterfall: Perbandingan Pendekatan Dalam Pengembangan Perangkat Lunak," *Nuansa Informatika: Jurnal Teknologi dan Informasi*, vol. 18, no. 2, pp. 41-45, 2024.
- [11] A. M. C. K. Zaki. A.M., "Sistem Pendukung Keputusan Penerimaan Tenaga Kerja Menggunakan Metode Naive Bayes," pp. 399-407, 2020.
- [12] F. A. A. S. W. M. Arianti. T., "Perancangan Sistem Informasi Perpustakaan Menggunakan Diagram Uml (Unified Modelling Language)," *Jurnal Ilmiah Komputer Terapan dan Infromasi*, vol. 1, no. 1, pp. 19-25, 2022.
- [13] W. B. Alfia. N.Y., "Perancangan Aplikasi Retensi Data Pada Database MySQL (Studi Kasus: PT. Telkomsigma)," *JUSIBI (Jurnal Sistem Informasi dan E-Bisnis)*, vol. 2, no. 3, pp. 364-374, 2020.
- [14] S. C. G. Tjandra, "Pemanfaatan Flutter dan Electron Framework pada Aplikasi Inventori dan Pengaturan Pengiriman Barang," *Journal Of Information System, Graphics, Hospitality and Technology*, vol. 2, no. 2, pp. 76-81, 2020.
- [15] F. Silalahi, *Manajemen Database MySQL (Structured Query Language)*, Semarang: Yayasan Prima Agus Teknik, 2022, pp. 3-4.
- [16] Dianta. IA., *Logika dan Algoritma untuk Merancang Aplikasi Komputer*, Semarang: Yayasan Prima Agus Teknik, 2021, pp. 10-11.
- [17] N. L. A. S. A. Arofiq, "Pengujian Sistem Schedule Planning Produksi Dengan Metode BlackBox Testing pada PT.Smartfren Telecom TBK Untuk Pemula," *Teknobis: Teknologi, Bisnis, dan Pendidikan*, vol. 1, no. 1, pp. 71-79, 2023.
- [18] R. P. J. B. A. Zahro, "Pengujian Aplikasi Seluler WAMSY (Warehouse Management System) Pada Raw Material Warehouse Menggunakan Metode Black Box," *Jurnal Teknis ITS*, vol. 10, no. 2, pp. 195-200, 2021.