

RANCANG BANGUN SISTEM INFORMASI RIVIEW FILM DAN BUKU MENGGUNAKAN METODE PROTOTYPING BERBASIS WEB

M. Gali Almahdi^{1*}, M. Farel Revansha Buulolo², Syuhada Simbolon³

¹²³Prodi Ilmu Komputer, Universitas Negeri Medan, Jl. Willem Iskandar, Pasar V, Medan Estate, Kota Medan, Sumatera Utara.

Keywords:

web;
Film;
Sistem informasi

Abstrak. Perkembangan teknologi informasi telah meningkatkan konsumsi konten digital, khususnya film dan buku. Banyaknya pilihan konten sering membuat pengguna kesulitan menentukan film atau buku yang sesuai dengan minat mereka. Selain itu, informasi ulasan dan penilaian pengguna masih tersebar di berbagai platform sehingga kurang efektif sebagai bahan pertimbangan. Penelitian ini bertujuan untuk merancang dan membangun sistem informasi review film dan buku berbasis web bernama ReviewHub. Sistem ini memungkinkan pengguna mencari konten, memberikan ulasan dan rating, serta memperoleh rekomendasi berdasarkan kesamaan genre menggunakan metode *Content-Based Filtering*. Metode pengembangan yang digunakan adalah Prototyping, dengan tahapan analisis kebutuhan, perancangan, pembangunan prototipe, dan pengujian. Sistem dikembangkan menggunakan React dan Supabase. Pengujian dilakukan menggunakan *Black-Box Testing* pada fitur utama sistem. Hasil pengujian menunjukkan bahwa seluruh fungsi berjalan sesuai kebutuhan. Sistem yang dibangun dapat membantu pengguna menemukan, menilai, dan memperoleh rekomendasi film dan buku secara lebih mudah dan terintegrasi.



Copyright © 2026 The Author(s), Published by Jurnal Informatika dan Teknik Elektro Terapan. This is an open-access article distributed under the terms of the Creative Commons Attribution-NonCommercial 4.0 International License (CC BY-NC 4.0).

Abstract. The rapid development of information technology has increased the consumption of digital content, particularly movies and books. The large number of available options often makes it difficult for users to determine which movies or books match their interests. In addition, user reviews and ratings are still scattered across various platforms, making them less effective as a reference for decision-making. This study aims to design and develop a web-based information system for reviewing movies and books called ReviewHub. The system enables users to search for content, submit reviews and ratings, and receive recommendations based on genre similarity using the Content-Based Filtering method. The system was developed using the Prototyping method, which consists of requirements analysis, design, prototype development, and testing stages. The application was built using React and Supabase. Testing was conducted using Black-Box Testing on the main system features. The results show that all functions operate as expected. The developed system can help users find, evaluate, and obtain movie and book recommendations more easily through an integrated platform.

1. PENDAHULUAN

Perkembangan teknologi internet di Indonesia telah mengubah pola konsumsi hiburan masyarakat secara mendasar.

Berdasarkan laporan Asosiasi Penyelenggara Jasa Internet Indonesia (APJII), jumlah pengguna internet di Indonesia pada tahun 2024 mencapai 221,5 juta jiwa atau setara

79,5% dari total populasi nasional [1]. Pertumbuhan pengguna internet tersebut berbanding lurus dengan meningkatnya konsumsi konten digital, khususnya film melalui platform streaming dan buku dalam format digital (*e-book*), yang semakin mudah diakses melalui berbagai perangkat pintar [2]. Fenomena ini mendorong lahirnya budaya berbagi pendapat secara daring, di mana masyarakat tidak lagi berperan sebagai konsumen pasif, melainkan aktif menulis ulasan dan memberikan penilaian sebagai bentuk partisipasi dalam ekosistem digital.

Dalam perspektif kebahasaan, teks ulasan merupakan jenis teks yang berfungsi untuk menilai suatu karya dari berbagai aspek kelebihan dan kekurangannya, dengan tujuan memberikan pertimbangan dan rekomendasi kepada pembaca [3]. Namun, ledakan volume konten digital melahirkan permasalahan serius berupa *information overload*, yaitu kondisi ketika pengguna kesulitan menemukan konten yang relevan di antara jutaan pilihan yang tersedia secara daring. Mayoritas pengguna platform hiburan digital mengalami hambatan signifikan dalam proses pemilihan konten akibat tidak tersedianya sistem panduan rekomendasi yang memadai [4]. Permasalahan ini diperparah oleh ketiadaan platform terpusat yang secara bersamaan mengintegrasikan fitur ulasan pengguna dan mekanisme rekomendasi otomatis untuk dua jenis konten hiburan utama, yaitu film dan buku, dalam satu ekosistem yang kohesif.

Sejumlah penelitian telah dilakukan untuk mengatasi permasalahan rekomendasi konten digital berbasis *Content-Based Filtering* (CBF) [5]. Mengembangkan sistem rekomendasi film berdasarkan kemiripan deskripsi cerita menggunakan metode *Content-Based Filtering* dan menunjukkan bahwa sistem mampu memberikan rekomendasi yang relevan berdasarkan karakteristik konten [6]. Menerapkan metode *Content-Based Filtering* pada sistem rekomendasi buku digital dengan pembobotan TF-RF untuk meningkatkan relevansi hasil pencarian buku [7]. Membangun prototipe sistem rekomendasi film Indonesia menggunakan pendekatan *Content-Based Filtering* dan metode *Vector Space Model* yang mampu menghasilkan rekomendasi film serupa berdasarkan atribut konten [8]. Menunjukkan bahwa penerapan

teknologi web modern menggunakan framework berbasis JavaScript mampu menghasilkan layanan digital dengan performa tinggi dan antarmuka yang responsif.

Meskipun demikian, penelitian-penelitian sebelumnya umumnya hanya berfokus pada satu jenis konten, yaitu film atau buku secara terpisah. Selain itu, sebagian besar sistem hanya menyediakan fitur rekomendasi tanpa mengintegrasikan fasilitas ulasan pengguna secara terpadu dalam satu platform. Oleh karena itu, diperlukan suatu sistem informasi yang mampu mengintegrasikan fitur review dan rekomendasi untuk dua jenis konten sekaligus, yaitu film dan buku, sehingga pengguna dapat memperoleh pengalaman eksplorasi konten yang lebih efektif dan personal.

Berdasarkan permasalahan tersebut, penelitian ini bertujuan untuk merancang dan membangun sistem informasi ReviewHub berbasis web menggunakan metode *Content-Based Filtering* untuk menghasilkan rekomendasi film dan buku berdasarkan kesamaan genre. Sistem dikembangkan menggunakan React.js sebagai frontend dan Supabase sebagai backend serverless berbasis PostgreSQL. Adapun kontribusi penelitian ini adalah menyediakan platform terintegrasi untuk review dan rekomendasi film serta buku, menerapkan algoritma *Content-Based Filtering* berbasis genre, serta menghadirkan antarmuka *Single Page Application* (SPA) yang interaktif dan responsif.

2. TINJAUAN PUSTAKA

Sistem rekomendasi merupakan bagian dari teknologi kecerdasan buatan (Artificial Intelligence/AI) yang dirancang untuk menyaring informasi berdasarkan preferensi pengguna berdasarkan data atau perilaku pengguna lainnya. Sistem ini telah banyak digunakan dalam berbagai platform populer seperti Netflix, Amazon, dan Spotify, yang menawarkan rekomendasi produk, film, musik, hingga buku [8][9]. *content-based filtering* adalah metode pada sistem rekomendasi yang berfokus pada karakteristik item-item yang akan direkomendasikan berdasarkan preferensi atau minat pengguna [10]. *Single Page Application* (SPA) adalah teknologi yang bekerja di dalam browser yang tidak membutuhkan reload page saat digunakan,

Adapun manfaat dan tujuan menggunakan teknik Single Page Application (SPA) adalah agar dalam penyajian sebuah data tidak diperlukan waktu yang lama untuk menampilkan data yang sedang diminta (request)[11]

ReactJS adalah kerangka bahasa pemrograman dari java script yang digunakan untuk membangun antarmuka pengguna (user interface) pada aplikasi berbasis website. Karena bahasa pemrograman ini bersifat open-source banyak library yang dapat digunakan di internet untuk mempermudah pengembang dalam membangun aplikasi. ReactJS menggunakan pendekatan berbasis komponen yang memungkinkan pengembang memecah UI sehingga dapat digunakan kembali[12]

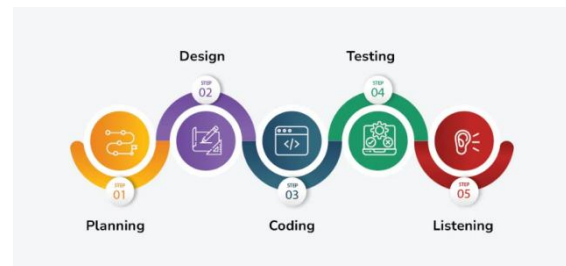
Supabase merupakan platform Backend as a Service berbasis PostgreSQL yang menyediakan database, autentikasi, API otomatis, real-time sync, serta Row-Level Security (RLS) untuk pengamanan data padat tingkat baris. Keunggulan Supabase terletak pada API otomatis, sinkronisasi real-time, serta penerapan RLS berbasis JWT[13]

Black Box Testing adalah metode pengujian perangkat lunak yang berfokus untuk memvalidasi fungsionalitas suatu aplikasi tanpa memerlukan pengetahuan tentang kode atau struktur internalnya. Metode ini mengevaluasi perangkat lunak berdasarkan input dan output yang diharapkan, dengan tujuan memastikan kepatuhan fungsional terhadap persyaratan yang ditentukan. Di antara kekurangan pengujian Black Box adalah bahwa mereka dapat memiliki tingkat kesulitan yang tinggi jika persyaratan untuk uji kasus tidak jelas. Selain itu, pengujian ini tidak efisien jika menguji kode yang sangat kompleks[14][15]

3. METODE PENELITIAN

Dalam proses pengembangan sistem diperlukan suatu metode atau model pengembangan perangkat lunak untuk mendukung jalannya pembangunan aplikasi. Salah satu metode yang dapat digunakan adalah Extreme Programming (XP). Metode ini dipilih karena mampu meningkatkan efisiensi, fleksibilitas, serta mempercepat proses pengembangan perangkat lunak melalui pendekatan iteratif dan berorientasi pada kebutuhan pengguna (Ananda Agung, 2024).

Extreme Programming terdiri dari empat tahapan utama, yaitu:



Gambar 3.1 Tahapan Pengembangan Sistem

1. Analisis Kebutuhan (Requirements Gathering)

Mengidentifikasi data konten (film dan buku), kebutuhan interaksi pengguna (registrasi, autentikasi, pemberian ulasan, dan rating), serta spesifikasi fungsional backend untuk manajemen data.

2. Desain (Design)

Menyusun arsitektur data relasional dan merancang alur kerja antarmuka (user interface layout) berbasis komponen untuk Single Page Application (SPA).

3. Pembangunan Prototipe (Build Prototype)

Melakukan implementasi pengodean pada sisi frontend menggunakan React.js dan integrasi basis data serverless melalui layanan Supabase.

4. Evaluasi dan Pengujian (User Evaluation & Testing)

Melakukan pengujian fungsionalitas menggunakan metode Black-box Testing untuk memastikan validitas logika sistem dan akurasi kalkulasi rekomendasi.

4. HASIL DAN PEMBAHASAN

4.1 Tahap Perencanaan (Planning)

4.1.1 Identifikasi Masalah

Permasalahan pengumpulan informasi ulasan film dan buku yang masih terfragmentasi, serta minimnya fitur rekomendasi personal yang adaptif terhadap preferensi pengguna. Untuk mengatasi

kendala tersebut, dirancang platform ReviewHub, sebuah sistem ulasan terintegrasi berbasis Single Page Application (SPA). Dalam pengembangannya, platform ini mengimplementasikan pustaka React.js dikombinasikan dengan build tool Vite pada sisi frontend. Penggunaan React.js dipilih karena efisiensi penanganan state dan kecepatan proses re-rendering komponen antarmuka yang dinamis. Visualisasi antarmuka murni dibangun menggunakan Vanilla CSS custom variables guna memastikan fleksibilitas desain tanpa membebani performa aplikasi. Untuk mendukung interaksi yang interaktif, digunakan pustaka React Router DOM untuk manajemen navigasi dan Framer Motion untuk efek animasi transisi halaman. Pada sisi backend, aplikasi ini menerapkan arsitektur serverless menggunakan platform Supabase dengan basis data relasional PostgreSQL. Pemanfaatan Supabase bertujuan untuk mempermudah manajemen data real-time.

4.1.2 Analisis Kebutuhan

Adapun kebutuhan fungsional pada sistem informasi ReviewHub adalah:

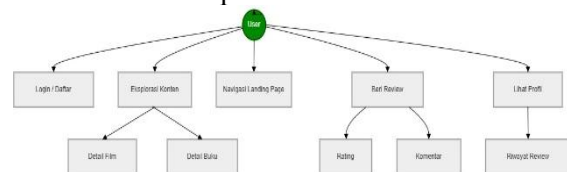
- Pengguna dapat melakukan registrasi dan login (Session-based authentication).
- Pengguna dapat mengelola data profil dan riwayat interaksi.
- Pengguna dapat mencari konten film atau buku berdasarkan judul dan kategori genre.
- Pengguna dapat memberikan ulasan (review) beserta penilaian (rating skala 1–5).
- Sistem dapat mengkalkulasi agregasi nilai ulasan secara dinamis.
- Sistem dapat menghasilkan rekomendasi film atau buku serupa secara otomatis melalui pencocokan genre menggunakan metode Content-Based Filtering.
- Admin dapat mengakses panel khusus untuk mengelola data konten, ulasan, dan pengguna.

4.2 Tahap Perancangan (Design)

4.2.1 Use Case Diagram

Perancangan use case diagram digunakan untuk memetakan interaksi yang terjadi antara pengguna (aktor) dengan fungsi-fungsi utama yang disediakan oleh sistem informasi ReviewHub. Berdasarkan alur navigasi pengguna (user navigation flow) yang telah dirancang, interaksi sistem dibagi menjadi dua kategori hak akses utama, yaitu pengguna umum (guest/user) dan pengguna dengan hak khusus (admin). Tahapan perancangan sistem ini secara visual dapat dilihat pada Gambar berikut ini:

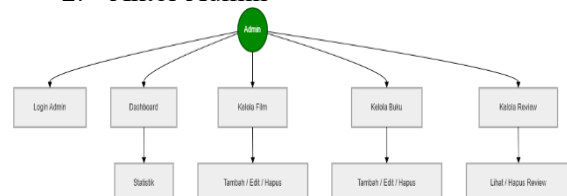
1. Aktor Pengguna (User/Guest) Mind Map User



Gambar 4.1 Mind Map User

Secara umum, Mind map User menggambarkan fitur-fitur utama yang dapat digunakan oleh pengguna biasa dalam sistem ReviewHub berupa pengguna dapat melakukan login atau daftar akun untuk masuk ke sistem. Setelah itu, pengguna dapat menjelajahi film dan menjelajahi buku yang tersedia di aplikasi. Pada bagian film dan buku, pengguna dapat membuka halaman detail film atau detail buku untuk melihat informasi lengkap seperti judul, deskripsi, rating, dan ulasan. Pengguna juga dapat memberi review berupa rating dan komentar terhadap film atau buku yang dipilih. Selain itu, pengguna dapat membuka profil untuk melihat data akun dan riwayat review yang pernah dibuat.

2. Aktor Admin



Gambar 4.2 Aktor Admin

Pada mind map Admin menjelaskan fitur-fitur utama yang hanya dapat diakses oleh administrator. Admin harus melakukan login

admin terlebih dahulu agar dapat masuk ke halaman pengelolaan sistem. Setelah masuk, admin dapat membuka dashboard untuk melihat statistik sistem, seperti jumlah film, buku, review, dan pengguna. Admin juga dapat mengelola film dan mengelola buku, yaitu menambahkan, mengedit, atau menghapus data konten. Selain itu, admin dapat mengelola review dengan melihat daftar review dari pengguna dan menghapus review jika diperlukan. pada platform.

4.2.2 Perancangan Algoritma Content-Based Filtering

Algoritma Content-Based Filtering (CBF) pada ReviewHub bekerja dengan menghitung skor kemiripan antara konten yang sedang dilihat pengguna dengan seluruh konten lain dalam basis data berdasarkan atribut genre. Sistem kemudian mengurutkan hasil berdasarkan skor tertinggi dan menampilkan sejumlah konten teratas sebagai rekomendasi. Implementasi algoritma CBF pada penelitian ini menerapkan mekanisme weighted scoring yang menggabungkan dua komponen penilaian: (1) jumlah genre yang beririsan (genre overlap) sebagai komponen utama, dan (2) bobot tambahan dari nilai rating agregat sebagai komponen sekunder.

4.2.3 Perancangan Sistem Agregasi Rating

Sistem agregasi rating dirancang untuk menghitung nilai rating rata-rata secara *real-time* dari seluruh ulasan yang masuk untuk setiap konten. Perhitungan dilakukan menggunakan formula rata-rata aritmetika standar:

$$\text{Rating}_{\text{avg}} = \frac{\sum \text{Rating}_i}{n}$$

$\sum \text{Rating}_i$ = Total akumulasi seluruh nilai rating dari semua ulasan

n = Jumlah ulasan yang masuk untuk konten tersebut.

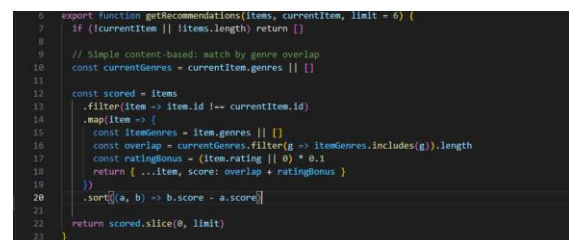
4.3 Build Ptototypr (Coding)

Pada tahap implementasi, seluruh rancangan logika, diagram alur, dan arsitektur basis data ditransformasikan ke dalam baris kode program menggunakan ekosistem React.js dan layanan basis data Supabase.

4.3.1 Implementasi Algoritma dan Logika Sistem

Proses kalkulasi rekomendasi dan kalkulasi statistik rating diwujudkan melalui skrip pemrograman JavaScript di sisi klien.

1. Mesin Rekomendasi (*Content-Based Filtering*): Implementasi fungsi Proses pengkodean fungsi rekomendasi diwujudkan pada berkas `lib/recommendation.js` melalui fungsi `getRecommendations()`. Fungsi ini menerima parameter berupa data katalog (`items`), identitas konten aktif (`currentId`), serta array kategori genre dari konten yang sedang diakses (`currentGenres`). Implementasi potongan kode algoritma tersebut disajikan pada Gambar.



```

export function getRecommendations(items, currentId, limit = 6) {
  if (!currentId || !items.length) return []

  // Simple content-based: match by genre overlap
  const currentGenres = currentId.genres || []

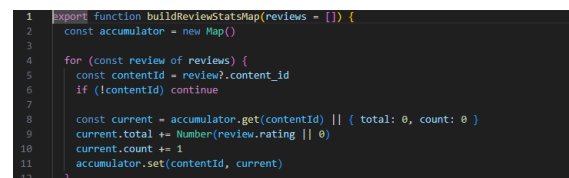
  const scored = items
    .filter(item => item.id !== currentId.id)
    .map(item => {
      const itemGenres = item.genres || []
      const overlap = currentGenres.filter(g => itemGenres.includes(g)).length
      const ratingBonus = (item.rating || 0) * 0.1
      return { ...item, score: overlap + ratingBonus }
    })
    .sort((a, b) => b.score - a.score)

  return scored.slice(0, limit)
}

```

Gambar 4.3 Potongan Kode Algoritma

2. Sistem Agregasi Rating: Fungsi `buildReviewStatsMap()` pada file `lib/reviewStats.js` diimplementasikan untuk mengumpulkan seluruh baris data ulasan dari tabel Supabase. Fungsi ini melakukan pemetaan (*mapping*) berdasarkan ID konten untuk menghitung jumlah total pengulas beserta nilai rata-rata (*mean*) rating secara dinamis.



```

export function buildReviewStatsMap(reviews = []) {
  const accumulator = new Map()

  for (const review of reviews) {
    const contentId = review?.content_id
    if (!contentId) continue

    const current = accumulator.get(contentId) || { total: 0, count: 0 }
    current.total += Number(review.rating || 0)
    current.count += 1
    accumulator.set(contentId, current)
  }
}

```

Gambar 4.3 Potongan Kode Pemetaan

4.3.2 Continuous Iteration

Selama proses konstruksi perangkat lunak, dilakukan pengelolaan kode sumber secara adaptif untuk memfasilitasi setiap perubahan komponen dan perbaikan *bug*.

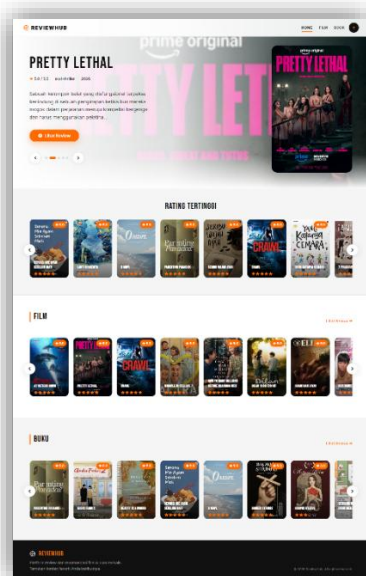
Penelitian ini memanfaatkan repositori Git berbasis GitHub sebagai media penyimpanan kontrol versi (*version control system*).

Setiap pembaharuan komponen antarmuka, optimasi kueri pencarian teks, maupun penyesuaian fungsionalitas *Single Page Application* (SPA) direkam melalui komit (*commit*) bertahap. Hal ini menjamin integritas kode sumber dan memudahkan pelacakan perubahan struktur aplikasi ReviewHub dari tahap awal hingga aplikasi siap disebar (deployment).

4.3.3 Implementasi Antarmuka Sistem (User Interface)

Berikut merupakan visualisasi dari halaman utama dan komponen-komponen penting yang telah berhasil diimplementasikan pada aplikasi ReviewHub:

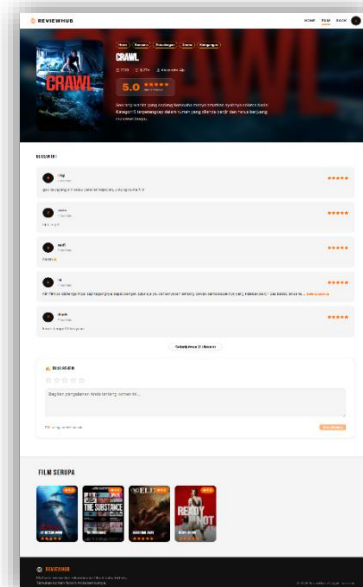
1. Halaman Beranda (*Home Page*)
Halaman ini merupakan antarmuka pertama yang diakses pengguna. Halaman beranda mengimplementasikan pengurutan popularitas (*Popularity Ranking*) untuk menampilkan film dan buku dengan nilai rating tertinggi dari hasil perhitungan agregasi ulasan.



Gambar 4.4 Halaman Beranda

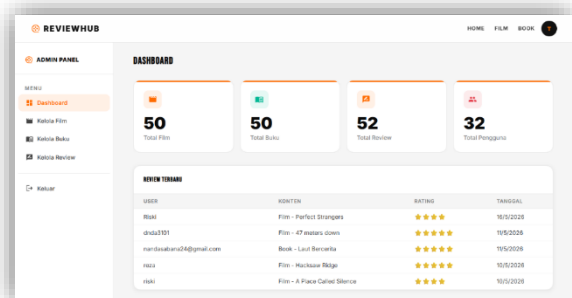
2. Halaman Detail Konten, Ulasan, rating dan Rekomendasi. Halaman ini

menampilkan informasi detail dari film atau buku yang dipilih. Pada bagian bawah halaman ini, fungsi mesin rekomendasi berbasis *Content-Based Filtering* menampilkan daftar konten serupa yang relevan bagi pengguna.



Gambar 4.5 Halaman Detail Konten

3. Halaman Dasbor Admin (*Admin Dashboard*). Antarmuka ini dirancang secara khusus untuk memfasilitasi pengguna yang memiliki tingkat hak akses sebagai pengelola (*Admin Role*). Melalui halaman ini, admin dapat melakukan pengawasan menyeluruh terhadap ekosistem data yang tersimpan di dalam basis data PostgreSQL Supabase.



Gambar 4.6 Halaman Dasbor Admin

4. Tahap Pengujian (*Testing*)

Pengujian fungsional pada sistem informasi ReviewHub dilakukan menggunakan metode *Black-Box Testing*. Metodologi ini berfokus pada pengamatan terhadap hasil masukan (*input*) dan keluaran (*output*) dari perangkat lunak guna memastikan seluruh fungsi berjalan normal sesuai dengan kebutuhan tanpa harus memeriksa struktur internal kode programnya.

Proses pengujian dilakukan pada komponen-komponen kritis sistem yang meliputi fungsi autentikasi, eksplorasi katalog, penulisan ulasan, keandalan mesin rekomendasi, serta kontrol panel admin. Hasil pengujian fungsionalitas tersebut dijabarkan pada tabel-tabel berikut:

a. Pengujian Halaman Utama (*Home Page*) dan Autentikasi

Bagian ini menguji kemampuan sistem dalam menyajikan data terpopuler (*Popularity Ranking*) serta keandalan manajemen sesi pengguna.

No	Skenario Pengujian	Hasil yang Diharapkan	Status
1	Mengakses halaman beranda tanpa masuk log (Guest Session).	Sistem berhasil memuat Single Page Application (SPA) dan menampilkan daftar film/buku terpopuler berdasarkan urutan rating tertinggi hasil agregasi.	Sesuai
2	Mengisi email dan kata kunci valid pada formulir login kemudian menekan tombol masuk.	Sistem memvalidasi token keamanan via Supabase Auth network dan mengubah status menjadi pengguna terautentikasi	Sesuai

		(User Session).	
3	Mengosongkan salah satu kolom pada formulir login dan menekan tombol masuk.	Sistem menolak proses autentikasi dan menampilkan pesan peringatan (error validation) pada antarmuka.	Sesuai

Tabel 4.1 Pengujian Halaman Utama

b. Hasil Pengujian Halaman Detail dan Rekomendasi

No	Skenario Pengujian	Hasil yang Diharapkan	Status
1	Menekan salah satu kartu (card) konten film atau buku pada halaman utama.	Sistem berpindah halaman menggunakan React Router DOM dan merender rincian deskripsi data konten secara lengkap.	Sesuai
2	Memeriksa seksi komponen rekomendasi di bagian bawah halaman detail.	Algoritma Content-Based Filtering berhasil melakukan komparasi array genre dan menampilkan maksimal 4 konten serupa berdasarkan Weighted Scoring tertinggi.	Sesuai
3	Memberikan penilaian bintang (skala 1-5) dan menulis	Data ulasan baru tersimpan ke dalam tabel PostgreSQL Supabase dan	Sesuai

	teks komentar pada kolom ulasan.	memicu pembaruan statistik rating secara dinamis.	
4	Mencoba menulis ulasan baru dalam kondisi belum melakukan login (Guest).	Sistem memblokir kolom masukan ulasan dan mengarahkan pengguna ke halaman autentikasi terlebih dahulu (Role Protection).	Sesuai

Tabel 4.2 Pengujian Halaman Detail dan Rekomendasi

c. Hasil Pengujian Halaman Dasbor Admin

No	Skenario Pengujian	Hasil yang Diharapkan	Status
1	Mengakses URL panel admin menggunakan akun dengan tingkat hak akses (role) 'Admin'.	Sistem berhasil melakukan validasi Auth Context dan membuka akses penuh ke halaman dasbor pengelolaan data.	Sesuai
2	Mengisi formulir data master baru (film atau buku) dan menekan tombol simpan.	Sistem berhasil melakukan operasi Create dan menyimpan records konten baru ke dalam database relasional.	Sesuai
3	Menekan tombol hapus pada salah satu baris daftar ulasan pengguna yang masuk.	Sistem menjalankan fungsi moderasi konten, menghapus ulasan yang dinilai melanggar, serta mengkalkulasi	Sesuai

		ulang average rating konten terkait.	
4	Mencoba mengakses paksa URL dasbor admin menggunakan akun dengan role 'User' biasa.	Sistem secara otomatis menolak permintaan, memblokir komponen antarmuka dasbor, dan melempar pengguna kembali ke halaman beranda.	Sesuai

Tabel 4.3 Pengujian Halaman Dasbor Admin

5. KESIMPULAN

Berdasarkan hasil penelitian yang telah dilakukan, sistem informasi review film dan buku berbasis web bernama ReviewHub berhasil dirancang dan dibangun menggunakan metode Prototyping. Sistem ini mampu menyediakan fitur pencarian konten, pemberian ulasan dan rating, serta rekomendasi film dan buku menggunakan metode *Content-Based Filtering* berdasarkan kesamaan genre.

Hasil pengujian menggunakan metode *Black-Box Testing* menunjukkan bahwa seluruh fitur utama sistem dapat berjalan sesuai dengan kebutuhan yang telah ditentukan, mulai dari proses autentikasi pengguna, pengelolaan ulasan, pencarian konten, hingga pengelolaan data oleh admin. Penggunaan React dan Supabase juga membantu dalam membangun sistem yang responsif dan mudah digunakan.

Dengan adanya sistem ReviewHub, pengguna dapat lebih mudah memperoleh informasi, memberikan penilaian, serta menemukan rekomendasi film dan buku dalam satu platform yang terintegrasi. Penelitian ini diharapkan dapat menjadi referensi untuk pengembangan sistem rekomendasi berbasis web yang lebih baik di masa mendatang.

UCAPAN TERIMA KASIH

Penulis mengucapkan terima kasih kepada pihak-pihak terkait yang telah memberi dukungan terhadap penelitian ini.

DAFTAR PUSTAKA

- [1] Asosiasi Penyelenggara Jasa Internet Indonesia, "Survei Penetrasi Internet Indonesia 2024," Jakarta, Indonesia, 2024. [Online]. Available: [APJII Survei Internet 2024](#)
- [2] R. A. Asmarantika, A. M. Prestianta, and N. Evita, "Pola konsumsi media digital dan berita online Gen Z Indonesia," *Jurnal Kajian Media*, vol. 6, no. 1, pp. 34–44, 2022.
- [3] R. A. Ananda and G. F. Nama, "Analisis dan Perancangan Layanan Streaming Film Berbasis Web Langganan Menggunakan Framework NextJS," *Jurnal Informatika dan Teknik Elektro Terapan (JITET)*, vol. 12, no. 1, pp. 352–359, 2024.
- [4] R. M. Harahap and A. N. Rachman, "Sistem rekomendasi film berdasarkan kemiripan deskripsi cerita menggunakan metode Content-Based Filtering," *Jurnal Informatika dan Teknik Elektro Terapan (JITET)*, vol. 13, no. 3, 2025. [Online]. Available: [PDF JITET CBF Film](#)
- [5] S. Rahmadhani, L. Hakim, and G. H. Wibowo, "Sistem rekomendasi penelusuran buku berbasis Content-Based Filtering dengan pembobotan TF-RF," *Jurnal Informatika Polinema*, vol. 10, no. 4, pp. 491–500, 2024. [Online]. Available: [PDF Jurnal Rekomendasi Buku](#)
- [6] D. T. Santoso, V. Atina, and D. Hartanti, "Prototipe sistem rekomendasi film Indonesia menggunakan pendekatan Content Based Filtering dan metode Vector Space Model," *Infotek: Jurnal Informatika dan Teknologi*, vol. 7, no. 2, pp. 444–455, 2024. [Online]. Available: [PDF Prototipe Sistem Rekomendasi Film Indonesia](#)
- [7] R. A. Ananda and G. F. Nama, "Analisis dan perancangan layanan streaming film berbasis web langganan menggunakan framework NextJS," *Jurnal Informatika dan Teknik Elektro Terapan (JITET)*, vol. 12, no. 1, pp. 352–359, 2024. [Online]. Available: [PDF JITET NextJS Streaming Film](#)
- [8] D. Velamentosa and E. Zuliarso, "Sistem Rekomendasi Film Menggunakan Metode Content-Based Filtering," *JATI (Jurnal Mhs. Tek. Inform.)*, vol. 9, no. 2, pp. 2918–2922, 2025, doi: 10.36040/jati.v9i2.13251.
- [9] S. A. Pratama, "Pengembangan Sistem Rekomendasi Buku Menggunakan Collaborative Filtering," *J. Komput.*, vol. 2, no. 2, pp. 81–86, 2024, doi: 10.70963/jk.v2i2.112.
- [10] T. Ridwansyah, B. Subartini, and S. Sylviani, "Penerapan Metode Content-Based Filtering pada Sistem Rekomendasi," *Math. Sci. Appl. J.*, vol. 4, no. 2, pp. 70–77, 2024, doi: 10.22437/msa.v4i2.32136.
- [11] M. F. Santoso, "Teknik Single Page Application (Spa) Layout Web Dengan Menggunakan React Js Dan Bootstrap," *J. Khatulistiwa Inform.*, vol. 9, no. 2, pp. 107–114, 2021, doi: 10.31294/jki.v9i2.11357.
- [12] M. Naufal Musyaffa, H. Fajri, and S. Hidayat Al Ikhsan, "Pembuatan Aplikasi Pemesanan Dan Pemasangan Kaca Film Mobil Berbasis Android Dan Website," *JATI (Jurnal Mhs. Tek. Inform.)*, vol. 8, no. 4, pp. 5878–5885, 2024, doi: 10.36040/jati.v8i4.9860.
- [13] I. G. Wahyu, W. Putra, E. M. Dharma, P. Trisna, and H. Permana, "IMPLEMENTASI RELATIONAL DATABASE DENGAN ROW-LEVEL SECURITY (RLS) PADA SISTEM INVENTORY MENGGUNAKAN SUPABASE DAN REACT NATIVE EXPO (STUDI KASUS BENGKEL SARI MERTA)," vol. 10, no. 2, pp. 2443–2448, 2026.
- [14] M. H. Alfrida, N. S. Salsabila, A. Wicaksono, and G. P. Mindara, "PENERAPAN BLACK BOX TESTING UNTUK EVALUASI FUNGSIONALITAS WEBSITE MAGGOPLAST," vol. 9, no. 1, pp. 169–176, 2025.
- [15] A. Samdono, A. P. Sari, and F. P. Aditiawan, "Pengujian Black Box Pada Sistem Informasi Stok Dan Penjualan Berbasis Website Menggunakan Metode Equivalence Partitioning (Studi Kasus: CV. Algani Karya Mandiri)," *JATI (Jurnal Mhs. Tek. Inform.)*, vol. 8, no. 1, pp. 880–885, 2024.