



Judul Tugas Besar BBK2DAB3 – Pengembangan Aplikasi Web

Tuliskan judul tugas besar yang akan dikerjakan pada Semester Ganjil Tahun Akademik 2025/2026. Judul harus memberikan gambaran yang jelas mengenai sistem yang akan dikembangkan serta tidak bersifat ambigu.

"Perancangan dan Pengembangan Smart Book System Berbasis Website sebagai Media Membaca Buku Digital Interaktif untuk Meningkatkan Minat Baca dan Akses Literasi Mahasiswa"

Nama Ketua dan Anggota Kelompok

Tuliskan nama seluruh anggota kelompok beserta NIM, nama lengkap, dan kelas masing-masing dengan format **NIM – Nama Lengkap – Kelas**.

1. **102022400122 – Michael William Setiawan Wee – SI-48-06 (Ketua Kelompok)**
2. **102022400020 – Muhammad Lukman Hakim – SI-48-06**
3. **102022400157 – Farhan Davin Rinaldi – SI-48-06**
4. **102022400252 – M. Surya Ilyasa – SI-48-06**
5. **102022400028 – Fadli Razzan Adabi – SI-48-06**

A. Latar Belakang

Pada bagian latar belakang pengembangan sistem, sebaiknya disajikan beberapa poin penting berikut untuk memberikan penjelasan yang logis mengenai kebutuhan dan urgensi pengembangan sistem:

A.1. Urgensi, Permasalahan atau Kebutuhan Nyata

Uraikan kondisi dan permasalahan aktual yang menjadi dasar perlunya pengembangan sistem. Pada bagian ini dapat disertakan gambar atau foto sebagai pendukung.

Minat baca di kalangan mahasiswa dan masyarakat saat ini cenderung mengalami penurunan seiring dengan berkembangnya media digital yang bersifat instan. Banyak mahasiswa lebih memilih sumber informasi singkat dibandingkan membaca buku secara utuh. Selain itu, keterbatasan akses terhadap buku fisik serta kurangnya platform membaca buku digital yang terintegrasi dan nyaman menjadi permasalahan tersendiri.

Platform membaca buku yang tersedia saat ini umumnya belum menyediakan pengalaman membaca yang interaktif, terstruktur, dan sesuai dengan kebutuhan akademik mahasiswa. Oleh karena itu, diperlukan sebuah sistem berbasis website yang mampu memfasilitasi kegiatan membaca buku digital secara efektif.

Urgensi, Permasalahan atau Kebutuhan Nyata Saat ini, tingkat literasi mahasiswa seringkali terhambat oleh aksesibilitas terhadap buku fisik yang terbatas, harga buku referensi yang mahal, serta perpustakaan fisik yang memiliki jam operasional terbatas.



Selain itu, metode membaca konvensional dianggap kurang menarik bagi generasi digital karena tidak adanya interaksi langsung (seperti *highlighting* atau pencatatan digital) yang terintegrasi. Mahasiswa membutuhkan platform yang tidak hanya menyediakan bahan bacaan, tetapi juga fitur interaktif untuk mendukung proses belajar..

A.2. Dampak Permasalahan yang Dihadapi

Jelaskan konsekuensi atau risiko yang muncul apabila permasalahan tidak ditangani. Pada bagian ini dapat disertakan gambar ilustrasi atau foto pendukung.

Apabila permasalahan tersebut tidak ditangani, maka minat baca mahasiswa ataupun masyarakat akan terus menurun dan berdampak pada kualitas pemahaman materi akademik. Selain itu, mahasiswa akan kesulitan mengelola aktivitas membaca, seperti melacak progres bacaan dan menemukan buku yang relevan dengan kebutuhan studi mereka..

A.3. Tujuan Pengembangan Sistem

Bagian ini menjelaskan urgensi pengembangan sistem dari aspek strategis dan operasional serta risiko yang berpotensi timbul apabila pengembangan tidak dilaksanakan. Penjelasan dapat dilengkapi dengan gambar atau foto pendukung.

Tujuan dari pengembangan sistem ini adalah untuk merancang dan membangun aplikasi Smart Book System berbasis website yang menyediakan fasilitas membaca buku digital secara interaktif. Sistem ini diharapkan dapat meningkatkan minat baca mahasiswa serta mempermudah akses literasi melalui fitur-fitur pendukung membaca menyediakan platform Smart Book System yang dapat diakses 24/7 di mana saja. Secara operasional, sistem ini bertujuan untuk mempermudah pencarian referensi melalui integrasi API buku global, menyediakan fitur membaca interaktif (catatan & penanda), serta membangun komunitas literasi melalui fitur ulasan.

A.4. Pentingnya Solusi Berbasis Sistem Informasi

Jelaskan dasar pertimbangan pemilihan solusi berbasis teknologi informasi sebagai pendekatan paling efektif dan efisien.

Solusi berbasis sistem informasi dipilih karena mampu memberikan kemudahan akses tanpa batasan waktu dan tempat. Sistem berbasis web juga memungkinkan pengelolaan data buku dan pengguna secara terpusat, sehingga lebih efisien, terstruktur, dan mudah dikembangkan di masa mendatang. Pentingnya Solusi Berbasis Sistem Informasi Pendekatan berbasis website dipilih karena tidak memerlukan instalasi aplikasi tambahan di perangkat pengguna.



(cross-platform), sehingga dapat diakses melalui laptop maupun smartphone. Teknologi web memungkinkan integrasi real-time dengan API eksternal (seperti Google Books) dan penyimpanan data cloud yang efisien untuk progres membaca pengguna, yang jauh lebih efektif dibandingkan pengelolaan perpustakaan manual...

B. Gambaran Umum Sistem

Bagian ini menjelaskan gambaran umum sistem yang akan dikembangkan, spesifikasi teknis termasuk perangkat dan alat pengembangan yang digunakan, ilustrasi pengguna sistem, struktur menu, serta mekanisme akses pengguna terhadap sistem.

.Sistem Smart Book System merupakan aplikasi berbasis website yang dirancang untuk meningkatkan minat baca dan akses literasi mahasiswa melalui buku digital interaktif. Sistem ini memberikan pengalaman membaca yang nyaman, terstruktur, dan dilengkapi fitur pendukung belajar.

1. Spesifikasi Teknis

- 1) Bahasa Pemrograman: PHP (Laravel Framework)
- 2) Database: MySQL
- 3) Front-end: Bootstrap / Tailwind CSS
- 4) Integrasi API: Google Books API untuk referensi global, Internal API untuk , review, dashboard, login, logout, review, dan notifikasi.
- 5) Platform: Web-based, cross-platform (dapat diakses melalui laptop, tablet, maupun smartphone tanpa instalasi aplikasi tambahan).

2. Aktor Pengguna

- 1) Admin:
 - a) Mengelola data pengguna dan buku digital.
 - b) Mengawasi aktivitas membaca, ulasan..
 - c) Mengakses laporan statistik sistem dan user.
- 2) User / Mahasiswa:
 - a) Membaca buku digital.
 - b) Menambahkan ulasan, meminjam/mengembalikan buku.

3. Mekanisme Akses

1. Pengguna harus mendaftar dan login terlebih dahulu.
2. Sistem menyediakan akses buku lokal (database internal) dan buku global melalui open library telkom university API.
3. Data catatan, aktivitas membaca, dan review tersimpan secara real-time di server melalui internal API.

4. Struktur Menu Utama



1. Fitur register dan login
 - A. Register akun baru untuk user dan login setelah membuat akunnya.
 2. Dashboard dan manajemen user
 - a. Ringkasan buku, progres baca, menambah buku baru, mengedit akun pengguna, statistik laporan sistem seperti user, berapa lama membaca, buku tersedia dan lainnya (untuk admin).
 3. Pencarian Buku dan pengelompokan buku berdasarkan AI (AI Engine)
 - a. Filter kategori, bahasa, tahun terbit; menampilkan buku dari database lokal dan API global.
 4. Membuat Event
 - a. Fitur baca interaktif, highlight teks, add note, bookmark..
 5. Notifikasi , Riwayat Aktivitas, Rating
 - 1) Fitur notif, ulasan buku, penilaian bintang.
 6. Pencarian dan pengelompokan buku berdasarkan kategori
 - A. Pengingat membaca, alert buku baru, timeline aktivitas, log aktivitas harian/bulanan.
5. Fitur Pendukung
1. CRUD data buku dan pengguna secara aman dan terstruktur.
 2. Ekspor laporan dalam format PDF atau Excel (misal: laporan aktivitas, katalog buku).
 3. Integrasi API eksternal dan internal untuk data buku secara global dan manajemen fitur yang interaktif.
 4. Notifikasi real-time melalui Web Push API untuk menjaga konsistensi aktivitas membaca.
6. Manfaat Sistem
1. Memberikan akses literasi tanpa batas waktu dan lokasi.
 2. Mendukung pembelajaran interaktif dan personalisasi melalui catatan, highlight, dan rekomendasi buku.
 3. Menumbuhkan motivasional reading habit
 4. Memudahkan monitoring literasi mahasiswa bagi pihak kampus atau admin.

C. Ruang Lingkup Sistem

Bagian ini menjelaskan ruang lingkup menu, fitur, dan API yang tersedia serta digunakan dalam sistem. Penetapan ruang lingkup disesuaikan dengan jumlah anggota yang terlibat.

Ruang lingkup sistem Smart Book System mencakup seluruh menu, fitur, dan API yang dikembangkan untuk mendukung pengalaman membaca buku digital interaktif bagi



mahasiswa. Penetapan ruang lingkup disesuaikan dengan jumlah anggota tim dan fokus pengembangan fitur yang saling terintegrasi.

Ruang lingkup sistem Smart Book System meliputi:

- Registrasi dan login pengguna
- Dashboard Admin dan Manajemen user
- Rekomendasi AI
- Membuat Event
- Rating dan ulasan
- Pencarian dan pengelompokan buku berdasarkan kategori

1. Fitur Utama Sistem

1. Registrasi dan Login Pengguna

- a. Pengguna dapat membuat akun baru dan login menggunakan email/nama.
- b. Login aman menggunakan enkripsi password dan autentikasi token (JWT/Sanctum).
- c. Fungsi logout dan manajemen profil tersedia untuk mengubah data pribadi atau password.

2. Dashboard Admin dan Manajemen User

- a. Admin dapat melihat daftar pengguna, menambah, mengubah, atau menghapus akun.
- b. Dashboard admin berfungsi untuk melihat jumlah pengguna, menambah buku baru ke sistem, melihat total buku/katalog, melihat status peminjaman, statistik aktivitas dan progres literasi setiap pengguna, download laporan CSV, ekspor data user, buku dan lainnya.

3. Rekomendasi AI

- a. Buku disimpan di database internal atau diambil dari Google Books API.

4. Membuat event

- a. Menyediakan tampilan membaca interaktif.
- b. User dapat menambahkan catatan, menandai halaman, dan menyorot teks.
- c. Riwayat bacaan tersimpan otomatis untuk memudahkan pelacakan progres membaca.

5. Pencarian dan Pengelompokan Buku

- a. Mesin pencari buku dengan filter kategori, bahasa, dan tahun terbit.
- b. Menampilkan hasil dari database internal maupun API eksternal.
- c. Mempermudah mahasiswa menemukan referensi sesuai kebutuhan akademik.

6. Rating, Review, dan Kolom Diskusi

- a. User dapat memberikan rating bintang dan ulasan pada buku yang dibaca..



2. Integrasi API

1. **Open library telkom:** Untuk pencarian dan informasi metadata buku global.
2. **Internal API:** Untuk manajemen user, dashboard admin, review, dan aktivitas pengguna.
3. **Gemini AI API:** Untuk rekomendasi buku yang bagus

3. Batasan Sistem

1. Aplikasi hanya tersedia dalam versi web (cross-platform), belum menyediakan versi mobile native.
2. Konten buku yang tersedia tergantung lisensi API eksternal dan koleksi internal.
3. Sistem tidak menangani transaksi pembelian buku digital; fokus pada literasi dan akses referensi.

4. Output / Keluaran Sistem

1. Laporan aktivitas pengguna (PDF/Excel).
2. Katalog buku pribadi dan ringkasan catatan (PDF).
3. Ringkasan ulasan buku atau log aktivitas bulanan.

Berikan gambaran sketsa / tampilan desain awal dari halaman awal sampai dengan halaman masing-masing fitur yang dikerjakan oleh masing-masing anggota kelompoknya.

D. Pembagian Tugas

Bagian ini menjelaskan latar belakang pemilihan setiap menu, tujuan penggunaannya, serta keterkaitannya dengan permasalahan yang diidentifikasi sehingga menu tersebut dinilai penting untuk dikembangkan. Selain itu, dijelaskan spesifikasi operasi CRUD yang mencakup struktur data dan atribut (field) yang digunakan, sumber dan dasar pemilihan API, serta jenis keluaran berupa distribusi berkas dalam bentuk ekspor atau cetak yang dihasilkan sistem.

D.1. Ketua Kelompok (102022400122 – Michael William Setiawan Wee – SI-48-06)

Bagian ini memuat lima pembahasan utama, yaitu: (1) latar belakang dan tujuan pemilihan menu, (2) spesifikasi fungsi dan keterkaitannya dengan permasalahan, (3) rancangan operasi CRUD beserta struktur data dan; (4) sumber API yang digunakan dilengkapi dengan alasannya, serta (5) jenis keluaran sistem berupa distribusi berkas hasil ekspor atau cetak.

MENU 1

MENU 1: LOGIN, LOGOUT, SESSION, REGISTER
--



Poin Pembahasan	Uraian
1. Latar Belakang & Tujuan	<p>1. Latar Belakang dan Tujuan Pemilihan Menu</p> <p>Latar Belakang: Keamanan data dan pengelolaan hak akses merupakan fondasi utama aplikasi Smart Book System . Tanpa sistem autentikasi yang baik, personalisasi fitur seperti “riwayat baca” dan akses ke dashboard admin tidak dapat dilakukan. Sistem harus mampu membedakan hak akses antara Admin dan User, serta menjaga data sensitif agar tidak disalahgunakan.</p> <p>Tujuan:</p> <ol style="list-style-type: none">1. Menyediakan autentikasi aman melalui login dan logout.2. Mengatur session agar hanya pengguna terdaftar yang dapat mengakses fitur sistem.3. Menyediakan Dashboard Admin untuk monitoring statistik sistem secara real-time.4. Mempermudah manajemen data pengguna melalui fitur CRUD User.
2. Spesifikasi Fungsi	Menu Register berfungsi untuk memungkinkan pengguna baru membuat akun pada sistem Smart Book System . Pengguna diminta untuk mengisi data pribadi seperti, nama lengkap, email, password, dan konfirmasi password.



	<p>Sistem akan melakukan validasi untuk memastikan data yang dimasukkan valid, seperti format email yang benar, unik, dan password memenuhi syarat keamanan.</p> <p>Setelah berhasil mendaftar, akun pengguna disimpan ke dalam database users dengan role default User, dan pengguna dapat langsung melakukan login. Fungsi ini membantu memperluas akses literasi digital, memastikan hanya pengguna terdaftar yang dapat mengakses fitur membaca buku digital dan interaktif. Menu register juga dilengkapi dengan fitur validasi kesalahan input, misalnya jika email sudah terdaftar atau password tidak sesuai ketentuan.</p> <p>Menu Login berfungsi sebagai pintu masuk pengguna ke sistem Smart Book System. Pengguna yang telah terdaftar memasukkan email dan password. Sistem akan memverifikasi data melalui database users dan membuat session aktif jika login berhasil.</p> <p>Login juga membedakan hak akses berdasarkan role pengguna:</p> <ul style="list-style-type: none">• Admin: Memiliki akses penuh ke dashboard, manajemen user, manajemen buku, dan laporan sistem.• User: Hanya dapat mengakses fitur membaca buku, wishlist, ulasan, pinjam/kembalikan buku. <p>Jika login gagal, sistem menampilkan notifikasi kesalahan. Session aktif digunakan untuk menjaga keamanan data, memastikan</p>
--	--



	<p>pengguna hanya bisa mengakses fitur sesuai haknya, dan mencegah akses tidak sah. Menu login juga mendukung integrasi dengan fitur logout untuk mengakhiri session.</p>
3. Rancangan CRUD & Struktur Data	<p>Tabel Utama:</p> <p>1. Fields / Atribut:</p> <ul style="list-style-type: none">a. id : INT (PK, AI) : Primary key, auto incrementb. name : VARCHAR(100) :Nama lengkap penggunac. email : VARCHAR(100) :Email pengguna, unikd. password : VARCHAR(255) :Password terenkripsi (hash)e. role : ENUM('admin','user') :Hak akses penggunaf. avatar : VARCHAR(255) :Foto profil (opsional)g. created_at : TIMESTAMP : Tanggal akun dibuath. updated_at : TIMESTAMP : Tanggal terakhir data diperbarui <p>Struktur Data Tabel users</p> <ul style="list-style-type: none">1. id : INT (PK, AI)<ul style="list-style-type: none">a. Primary key, auto increment, sebagai identitas unik tiap pengguna.2. name : VARCHAR(100)<ul style="list-style-type: none">a. Nama lengkap pengguna.3. email : VARCHAR(100)<ul style="list-style-type: none">a. Email pengguna, unik, digunakan untuk login dan notifikasi sistem.4. password : VARCHAR(255)<ul style="list-style-type: none">a. Password terenkripsi (hash), digunakan untuk autentikasi login.5. role : ENUM('admin','user')



- a. Hak akses pengguna; **admin** untuk pengelolaan sistem, **user** untuk pembaca buku.
6. **avatar** : VARCHAR(255)
 - a. Foto profil pengguna (opsional).
7. **created_at** : TIMESTAMP
 - a. Tanggal dan waktu akun dibuat.
8. **updated_at** : TIMESTAMP
 - a. Tanggal dan waktu terakhir data diperbarui.

Struktur Data Tabel Session / Aktivitas Login

1. **session_id** : VARCHAR(255)
 - a. ID unik untuk tiap session pengguna.
2. **user_id** : INT (FK)
 - a. ID pengguna yang terkait dengan session (mengacu ke tabel **users**).
3. **login_time** : TIMESTAMP
 - a. Waktu login berhasil.
4. **logout_time** : TIMESTAMP
 - a. Waktu logout / session berakhir.
5. **status** : ENUM('active','inactive')
 - a. Status session, **active** saat login, **inactive** saat logout.

Operasi CRUD:

1. Menu Register

1. **Create:**
 - a. Menambahkan user baru ke tabel **users** setelah validasi email, dan password.
 - b. Role default = **user**.
2. **Read:**
 - a. Tidak ada operasi read langsung pada register, validasi dilakukan sebelum insert.
3. **Update:**
 - a. Tidak ada update di menu register.
4. **Delete:**
 - a. Tidak ada delete di menu register.



	<p>2. Menu Login & Session</p> <ol style="list-style-type: none">Read:<ol style="list-style-type: none">Mengecek keberadaan user berdasarkan email dan password.Jika valid → buat session aktif untuk menyimpan info user (id, role, nama).Session Management:<ol style="list-style-type: none">Session menyimpan data login sementara untuk mengontrol hak akses.Logout menghapus session → akses ke fitur dibatasi.Create / Update / Delete:<ol style="list-style-type: none">Tidak ada operasi langsung di tabel users (kecuali update password). <p>3. Menu Logout</p> <ol style="list-style-type: none">Delete:<ol style="list-style-type: none">Menghapus session aktif saat pengguna logout.Mencegah akses tidak sah ke dashboard dan menu lainnya.
<p>4. Sumber API</p>	<p>Internal REST API (Authentication Service).</p> <p>Alasan: Internal REST API Authentication Service adalah layanan API internal yang digunakan untuk mengelola proses autentikasi dan otorisasi pengguna dalam sistem Smart Book System. Layanan ini berfungsi menjaga keamanan data login, manajemen session, serta pembatasan hak akses sesuai role pengguna, baik admin maupun user.</p> <p>Fitur Register memungkinkan pengguna baru untuk membuat akun dengan mengirimkan data berupa nama lengkap, email, dan password. API akan melakukan validasi terhadap format email, keunikan, dan kekuatan password sebelum menyimpan data ke tabel users dengan role default user. Setelah registrasi berhasil, sistem mengirimkan respons berupa konfirmasi pembuatan akun.</p>



	<p>Fitur Login menerima kredensial pengguna berupa email beserta password. API kemudian memverifikasi kecocokan data dengan tabel users. Jika valid, API akan membuat session atau token autentikasi (JWT/Sanctum) yang digunakan untuk mengontrol hak akses pengguna. Sistem mengembalikan informasi pengguna beserta token yang dapat digunakan untuk mengakses fitur dalam sistem sesuai role yang dimiliki.</p> <p>Fitur Logout digunakan untuk mengakhiri session aktif pengguna. Ketika pengguna melakukan logout, API akan menandai session/token sebagai tidak aktif sehingga akses ke dashboard dan menu lain akan dibatasi. Hal ini memastikan bahwa data pengguna tetap aman dan mencegah akses tidak sah setelah sesi berakhir.</p> <p>Manajemen Session memungkinkan sistem untuk memeriksa status session atau token saat ini, menentukan apakah pengguna masih login, dan mengatur hak akses berdasarkan role. Dengan cara ini, pengguna yang login dapat menggunakan fitur sesuai izin mereka, sementara keamanan data tetap terjaga.</p> <p>Penggunaan Internal REST API dipilih karena data sensitif seperti password dan session harus dikelola secara internal agar aman, serta memungkinkan integrasi real-time antara front-end dan back-end tanpa ketergantungan pada layanan pihak ketiga. Pendekatan ini juga memudahkan pengelolaan autentikasi secara efisien dan terstruktur, sehingga sistem dapat berjalan stabil dan aman.</p>
5. Jenis Keluaran (Ekspor/Cetak)	<p>Fitur Register, Login, Logout, dan Session pada sistem Smart Book System dapat menghasilkan dokumen fisik atau berkas ekspor secara langsung, karena fungsinya lebih berfokus pada autentikasi dan manajemen akses pengguna secara real-time. Namun, data yang dikumpulkan melalui aktivitas ini dapat dimanfaatkan untuk pembuatan laporan oleh admin.</p>



Poin Pembahasan	Uraian
1. Latar Belakang & Tujuan	<p>1. Latar Belakang dan Tujuan Pemilihan Menu</p> <p>Latar Belakang: Dashboard Admin dan Manajemen User dipilih karena merupakan inti dari sistem pengelolaan perpustakaan digital yang memerlukan pengawasan terpusat, kontrol akses berbasis peran, dan efisiensi dalam operasi administratif.</p> <p>Tujuan:</p> <ol style="list-style-type: none">1. Memberikan admin dan super_admin akses cepat untuk memantau statistik sistem (pengguna, buku, peminjaman).2. Memisahkan hak akses berdasarkan peran (<i>role</i>) untuk keamanan dan relevansi fungsi.3. Memfasilitasi pengelolaan pengguna (tambah, edit, hapus, ubah peran) dalam satu antarmuka terstruktur.4. Menyediakan navigasi yang intuitif dari dashboard ke modul lain (misal: ke katalog buku atau daftar pengguna).
2. Spesifikasi Fungsi	<p>Fungsi Dashboard Admin dirancang untuk menampilkan statistik ringkasan seperti jumlah pengguna, total buku, dan buku yang dipinjam, yang membantu admin memantau kondisi perpustakaan secara cepat. Fitur ini mengatasi permasalahan kesulitan dalam memperoleh gambaran menyeluruh sistem tanpa harus membuka banyak halaman. Sementara itu, fungsi Manajemen User menyediakan tabel dengan kolom ID, nama, email, peran, tanggal daftar, dan aksi yang memungkinkan admin melihat, menambah, mengubah, atau menghapus pengguna. Modul ini mengatasi masalah pengelolaan data pengguna yang terfragmentasi dengan menyatukannya dalam satu antarmuka yang dapat</p>



	<p>disaring dan dicari. Kedua fungsi ini saling terkait melalui navigasi langsung dari dashboard ke daftar pengguna, sehingga mempercepat akses administratif.</p>
3. Rancangan CRUD & Struktur Data	<p>Rancangan CRUD :</p> <p>Read: Mengambil data dari tabel users, books, personal_notes, dan user_activities untuk ditampilkan dalam dashboard. Selain itu Operasi CRUD pada Manajemen User dirancang dengan struktur data tabel "users" yang terdiri dari ID, nama, email, kata sandi terenkripsi, peran (user, admin,), serta tanggal pembuatan dan pembaruan. Operasi Create memungkinkan admin menambahkan pengguna baru melalui formulir dengan validasi email unik. Operasi Read menampilkan daftar pengguna dalam tabel dengan pagination dan filter berdasarkan peran. Operasi Update memungkinkan perubahan data pengguna seperti nama, email, atau peran melalui antarmuka edit dengan batasan tertentu (misalnya admin tidak dapat mengubah peran super_admin). Operasi Delete memberikan opsi penghapusan akun dengan konfirmasi dan pencegahan penghapusan akun sendiri. Seluruh operasi ini dilindungi oleh validasi session dan hak akses berbasis peran.</p> <p>Struktur data yang diakses:</p> <ul style="list-style-type: none">• users: id, name, email, role, created_at, updated_at• books: id, title, authors, published_date, thumbnail_url• personal_notes: id, user_id, book_id, note_content, page_number• user_activities: id, user_id, activity_type, description, created_at
4. Sumber API	<p>Internal REST API (Authentication Service).</p> <p>Alasan:</p> <p>Sistem ini menggunakan beberapa API utama yang dirancang dengan pendekatan RESTful untuk mendukung fungsionalitas Dashboard Admin dan Manajemen User. API</p>



	<p><code>/api/dashboard/stats</code> digunakan untuk mengambil data statistik dengan alasan agar dashboard dapat menampilkan informasi real-time tanpa perlu memuat ulang halaman. API <code>/api/users</code> menyediakan operasi CRUD lengkap untuk manajemen pengguna, dipilih karena kemudahannya dalam berintegrasi dengan tabel frontend dan fleksibilitas dalam penanganan data. API <code>/api/export/users/csv</code> digunakan untuk mengekspor data pengguna dan buku ke format CSV, dengan alasan memenuhi kebutuhan backup dan analisis data di luar sistem. API <code>/api/auth/logout</code> bertujuan untuk mengakhiri session secara aman. Pemilihan API ini didasarkan pada prinsip modularitas, keamanan melalui middleware autentikasi, dan kompatibilitas dengan arsitektur sistem yang terpisah antara frontend dan backend.</p>
5. Jenis Keluaran (Ekspor/Cetak)	<p>Sistem menghasilkan beberapa jenis keluaran untuk mendukung kebutuhan administratif, terutama dalam bentuk distribusi berkas. Ekspor data pengguna dan buku tersedia dalam format CSV yang dapat diunduh melalui tombol "Export CSV", menghasilkan file dengan kolom terstruktur seperti, nama, email, peran, dan tanggal daftar untuk keperluan arsip atau analisis eksternal. Laporan juga dapat dicetak langsung dari halaman Manajemen User dalam format PDF atau tampilan cetak browser, memungkinkan dokumentasi fisik untuk kebutuhan administrasi. Selain itu, sistem menyediakan notifikasi operasional berupa pesan teks yang muncul setelah tindakan CRUD, memberi umpan balik langsung kepada pengguna. Keluaran statistik visual dalam dashboard, meski bukan berkas, berfungsi sebagai laporan ringkas yang dapat dipantau secara real-time untuk pengambilan keputusan.</p>

D.2. Anggota Kelompok – 1 (102022400020 – Muhammad Lukman Hakim – SI-48-06)

Bagian ini memuat lima pembahasan utama, yaitu: (1) latar belakang dan tujuan pemilihan menu, (2) spesifikasi fungsi dan keterkaitannya dengan permasalahan, (3) rancangan operasi CRUD beserta struktur data dan; (4) sumber API yang digunakan dilengkapi dengan alasannya, serta (5) jenis keluaran sistem berupa distribusi berkas hasil ekspor atau cetak.



Poin Pembahasan	Uraian
1. Latar Belakang & Tujuan	<p>Latar Belakang: Menu Pencarian dan Pengelompokan Buku dipilih karena koleksi buku fisik maupun e-book lokal seringkali terbatas dan sulit diperbarui secara manual. Di sisi lain, mahasiswa membutuhkan akses cepat dan luas ke ribuan referensi global yang valid dan relevan. Oleh karena itu, diperlukan sistem pencarian yang terintegrasi dan efisien.</p> <p>Tujuan:</p> <ul style="list-style-type: none">• Menyediakan mesin pencari buku yang terhubung ke database lokal dan API global secara real-time.• Memudahkan mahasiswa dalam menemukan referensi buku sesuai kebutuhan akademik atau minat.• Menyediakan fitur filter berdasarkan kategori, bahasa, dan tahun terbit agar hasil pencarian lebih spesifik dan relevan.• Mengintegrasikan koleksi lokal dengan sumber eksternal untuk memperluas jangkauan referensi.
2. Spesifikasi Fungsi	<p>Sistem menyediakan fitur pencarian buku berdasarkan judul dengan memanfaatkan integrasi OpenLibrary API. Melalui fitur ini, pengguna dapat memperoleh data buku seperti judul, tanggal baca, penulis, rating buku, dan sampul buku secara otomatis. Buku hasil pencarian dapat disimpan ke dalam jurnal atau koleksi pribadi pengguna, sehingga memudahkan pencatatan dan pengelolaan referensi. Selain itu, sistem mendukung pengelolaan data buku yang tersimpan melalui fungsi tambah, ubah, dan hapus. Seluruh data buku yang dikelola bersifat personal dan terpisah antar pengguna karena sistem menerapkan mekanisme autentikasi untuk menjaga keamanan dan privasi data.</p>
3. Rancangan CRUD & Struktur Data	<p>Operasi CRUD:</p>



	<ol style="list-style-type: none">1. Create: Pengguna menambahkan buku hasil pencarian ke daftar favorit.2. Read: Sistem menampilkan daftar buku favorit milik pengguna (sinopsis, pengarang, rating).3. Update: Pengguna mengubah informasi tambahan pada buku yang disimpan.4. Delete: Pengguna menghapus buku dari daftar favorit. <p>Tabel Utama: books (Buku Favorit/Lokal) Struktur Data: id, user_id, title, author, isbn, publish_year, cover_url.</p>
4. Sumber API	<p>Google Books API.</p> <p>Alasan: API ini bersifat publik, gratis, dokumentasinya lengkap, dan menyediakan metadata buku yang sangat kaya (termasuk cover dan sinopsis) yang sangat dibutuhkan untuk menarik minat baca mahasiswa.</p>
5. Jenis Keluaran (Ekspor/Cetak)	<p>Jenis keluaran sistem berupa distribusi berkas hasil yang menampilkan data buku atau laporan jurnal secara terstruktur. Keluaran ini memungkinkan pengguna atau admin untuk melihat, membagikan, atau mencetak informasi buku sesuai kebutuhan, termasuk judul, penulis, kategori, ISBN, rating, dan status peminjaman.</p>

D.3. Anggota Kelompok – 3 (102022400157 – Farhan Davin Rinaldi – SI-48-06)

Bagian ini memuat lima pembahasan utama, yaitu: (1) latar belakang dan tujuan pemilihan menu, (2) spesifikasi fungsi dan keterkaitannya dengan permasalahan, (3) rancangan operasi CRUD beserta struktur data dan; (4) sumber API yang digunakan dilengkapi dengan alasannya, serta (5) jenis keluaran sistem berupa distribusi berkas hasil ekspor atau cetak.

Poin Pembahasan	Uraian
1. Latar Belakang & Tujuan	Latar Belakang: Fitur ini dikembangkan untuk mengatasi keterbatasan pencarian buku tradisional yang kaku, di mana pengguna kesulitan



	<p>menemukan rekomendasi yang relevan di tengah banyaknya katalog buku.</p> <p>Tujuan Utama:</p> <ul style="list-style-type: none">• Menyediakan rekomendasi buku atau topik yang personal, akurat, dan sangat kontekstual berdasarkan <i>query</i> bahasa alami yang dimasukkan pengguna.• Menciptakan pengalaman interaktif yang meniru konsultasi dengan pustakawan sungguhan.• Meningkatkan keterlibatan pengguna dengan platform Smartbook melalui fitur eksplorasi konten yang inovatif.
2. Spesifikasi Fungsi	<p>Fungsi Input Query: Halaman ini harus mampu menerima dan memvalidasi masukan teks bebas yang panjang (pertanyaan atau permintaan rekomendasi) dari pengguna di <i>input area</i>.</p> <p>Fungsi Pemrosesan AI: Memicu pengiriman <i>query</i> tersebut ke <i>backend</i> yang menjalankan algoritma kecerdasan buatan (seperti Model Bahasa Besar atau Mesin Rekomendasi).</p> <p>Fungsi Personalisasi: <i>Backend</i> AI harus menggunakan konteks pengguna yang <i>login</i> (Michael), seperti riwayat bacaan atau <i>rating</i> sebelumnya, untuk menghasilkan rekomendasi yang lebih sesuai.</p> <p>Fungsi Keluaran: Menampilkan hasil rekomendasi dalam format yang terstruktur dan mudah dibaca (misalnya, daftar judul, penulis, dan alasan rekomendasi) segera setelah pemrosesan selesai.</p> <p>Fungsi Status Interaktif: Setelah menekan Minta Rekomendasi, harus ada indikator visual (<i>loading state</i>) yang memberitahu pengguna bahwa <i>query</i> sedang diproses.</p>
3. Rancangan CRUD & Struktur Data	<p>Fungsi utama dari halaman AI Librarian ini adalah memicu layanan, bukan mengelola data buku secara langsung, sehingga operasi CRUD (Create, Read, Update, Delete) lebih berfokus pada log interaksi dan pengambilan data.</p> <p>Operasi READ terjadi secara implisit di mana sistem harus membaca data buku dari katalog utama</p>



	<p>Smartbook untuk dicocokkan dengan permintaan AI, dan kemudian data buku yang relevan akan disajikan kembali sebagai hasil rekomendasi kepada pengguna. Operasi CREATE terjadi saat pengguna memasukkan dan mengirimkan permintaan teksnya; sistem akan menciptakan atau mencatat log interaksi atau sesi <i>query</i> baru yang berisi teks permintaan pengguna untuk tujuan audit dan pelatihan model AI di masa depan.</p> <p>Untuk operasi UPDATE dan DELETE, keduanya tidak diperlukan secara langsung pada <i>query</i> rekomendasi ini, namun dapat diterapkan pada <i>log</i> interaksi. Operasi UPDATE dapat digunakan jika pengguna memberikan umpan balik (misalnya, menyatakan rekomendasi itu baik atau buruk), yang kemudian akan memperbarui preferensi pengguna untuk rekomendasi berikutnya. Sementara itu, operasi DELETE dapat digunakan untuk menghapus riwayat interaksi tertentu antara pengguna dan AI Librarian.</p> <p>Adapun Struktur Data Keluaran utama yang akan dihasilkan oleh API setelah pemrosesan AI adalah sebuah objek atau <i>array</i> yang berisi detail buku yang direkomendasikan. Struktur ini mencakup ID unik buku (bookId), Judul buku, Nama Penulis, Rating Rata-rata buku di platform, dan elemen yang paling penting, yaitu Alasan atau Ulasan Singkat (reasoning) yang dihasilkan oleh AI untuk menjelaskan mengapa buku tersebut sesuai dengan <i>query</i> pengguna.</p>
4. Sumber API	<p>Fitur AI Librarian membutuhkan pemanggilan API yang mengirim dan menerima data dari mesin rekomendasi cerdas.</p> <ul style="list-style-type: none">• API Utama: POST /api/ai/recommendation<ul style="list-style-type: none">○ Metode: POST (Karena mengirim data input baru).○ Payload (Data Terkirim): Berisi objek yang mencakup teks permintaan (query: "Saya suka buku Harry



	<p>Potter...") dan ID pengguna (<code>userId: Michael</code>).</p> <ul style="list-style-type: none">○ Fungsi Backend: API ini adalah <i>gateway</i> yang memicu proses AI, yang kemudian melakukan pencarian data dan menghasilkan rekomendasi.● API Pendukung: GET <code>/api/user/preference</code><ul style="list-style-type: none">○ Tujuan: Digunakan oleh <i>backend</i> AI untuk menarik data riwayat dan preferensi pengguna untuk meningkatkan kualitas personalisasi rekomendasi.
5. Jenis Keluaran (Ekspor/Cetak)	<p>Keluaran utama adalah tampilan visual daftar rekomendasi, dengan opsi untuk mengelola hasil tersebut:</p> <ul style="list-style-type: none">● Keluaran Standar: Daftar rekomendasi buku yang tersusun rapi dan ditampilkan di layar web setelah pemrosesan selesai.● Opsi Tindak Lanjut Digital:<ul style="list-style-type: none">○ Simpan ke Daftar Bacaan: Tombol di samping setiap rekomendasi untuk langsung menambahkannya ke Perpustakaan Saya pengguna.○ Berbagi: Opsi untuk mengirim daftar rekomendasi tersebut melalui <i>email</i> atau pesan.● Opsi Fisik: Mencetak halaman hasil atau mengekspor daftar rekomendasi sebagai berkas PDF sederhana.

D.4. Anggota Kelompok – 4 (102022400252 – M. Surya Ilyasa – SI-48-06)

Bagian ini memuat lima pembahasan utama, yaitu: (1) latar belakang dan tujuan pemilihan menu, (2) spesifikasi fungsi dan keterkaitannya dengan permasalahan, (3) rancangan operasi CRUD beserta struktur data dan; (4) sumber API yang digunakan dilengkapi dengan alasannya, serta (5) jenis keluaran sistem berupa distribusi berkas hasil ekspor atau cetak.



Poin Pembahasan	Uraian
1. Latar Belakang & Tujuan	<p>Latar Belakang: Membaca buku seringkali menjadi aktivitas soliter. Untuk meningkatkan literasi dan interaksi antar pengguna, diperlukan sebuah wadah informasi mengenai kegiatan komunitas seperti bedah buku atau pameran.</p> <p>Tujuan: Memberikan kemudahan bagi pengguna aplikasi <i>smartbook</i> untuk mengetahui jadwal, lokasi, dan detail acara literasi yang akan datang, sehingga terbangun ekosistem pembaca yang lebih aktif.</p>
2. Spesifikasi Fungsi	<p>Menyajikan tabel informasi acara literasi (ID, Nama, Tanggal, Lokasi, Deskripsi). Seringkali pembaca ketinggalan info mengenai acara bedah buku atau diskon pameran buku karena informasi yang tersebar. Fitur ini memusatkan informasi tersebut dalam satu platform aplikasi jurnal buku.</p>
3. Rancangan CRUD & Struktur Data	<p>Rancangan CURD:</p> <ul style="list-style-type: none">● Create: Admin/Sistem menambahkan event baru (misal: Workshop Menulis).● Read: Menampilkan daftar event dalam bentuk tabel seperti pada gambar (ID 1, 2, 3).● Update: Mengubah detail lokasi atau tanggal jika terjadi perubahan jadwal.● Delete: Menghapus event yang sudah selesai atau dibatalkan. <p>Tabel Utama: events (Kegiatan Buku) struktur Data: id, nama_event, tanggal, lokasi, dan deskripsi.</p>
4. Sumber API	<p>Internal API (Comment System).</p> <p>Alasan: Penggunaan API internal memungkinkan fleksibilitas dalam mengelola data event yang bersifat lokal atau spesifik untuk komunitas pengguna <i>smartbook</i>. Data dapat diambil secara <i>asynchronous</i> (misal menggunakan Fetch API atau Axios) agar halaman tidak perlu dimuat ulang saat data diperbarui.</p>



5. Jenis Keluaran (Ekspor/Cetak)	Sistem menyediakan fitur untuk mengekspor daftar event ke dalam format PDF atau CSV. Tujuan: Memudahkan pengguna untuk menyimpan jadwal secara luring (offline) atau bagi penyelenggara untuk mencetak daftar acara sebagai selebaran di lokasi kegiatan (seperti Perpustakaan Kota atau Telkom sesuai data pada gambar).
---	--

D.5. Anggota Kelompok - 5 (102022400028 – Fadli Razzan Adabi – SI-48-06)

Bagian ini memuat lima pembahasan utama, yaitu: (1) latar belakang dan tujuan pemilihan menu, (2) spesifikasi fungsi dan keterkaitannya dengan permasalahan, (3) rancangan operasi CRUD beserta struktur data dan; (4) sumber API yang digunakan dilengkapi dengan alasannya, serta (5) jenis keluaran sistem berupa distribusi berkas hasil ekspor atau cetak.

Poin Pembahasan	Uraian
1. Latar Belakang & Tujuan	Latar Belakang: Mahasiswa sering kali kesulitan menentukan kualitas buku sebelum membacanya dan kurang memiliki ruang untuk mengekspresikan opini terhadap literatur yang telah diselesaikan. Tujuan: Membangun ekosistem literasi yang interaktif dengan memfasilitasi pertukaran ulasan antar-mahasiswa, sehingga mempermudah pengambilan keputusan dalam memilih bahan bacaan yang berkualitas.
2. Spesifikasi Fungsi	Fitur ini mencakup sistem Bintang (Rating) skala 1-10, kolom Ulasan Tekstual , dan fitur Like/Helpful pada ulasan orang lain. Fungsi ini bertujuan untuk meningkatkan keterlibatan komunitas dan memberikan validasi sosial terhadap koleksi buku di perpustakaan.
3. Rancangan CRUD & Struktur Data	Tabel Utama: book_reviews



	<p>Struktur Data: id (PK), book_id (FK), user_id (FK), rating (integer), review_text (text), created_at, updated_at.</p> <p>Operasi CRUD:</p> <ol style="list-style-type: none">1. Create: Mahasiswa mengirimkan rating dan ulasan setelah selesai meminjam atau membaca buku.2. Read: Menampilkan rata-rata rating di halaman detail buku dan daftar ulasan dari pengguna lain.3. Update: Memberikan kesempatan bagi pengguna untuk menyunting ulasan jika terjadi perubahan opini atau perbaikan tulisan.4. Delete: Memungkinkan pengguna menghapus ulasan mereka sendiri atau admin menghapus ulasan yang melanggar ketentuan komunitas.
4. Sumber API	<p>Internal API (Review Engine) dan Sentiment Analysis API (Opsional). Alasan: Internal API digunakan untuk mengelola data ulasan secara real-time, sedangkan Sentiment Analysis dapat digunakan untuk mengategorikan ulasan secara otomatis menjadi "Positif", "Netral", atau "Negatif" agar mudah dipahami oleh pembaca lain.</p>
5. Jenis Keluaran (Ekspor/Cetak)	<p>Katalog Ulasan Terpopuler (PDF): Ringkasan buku dengan rating tertinggi dan ulasan paling inspiratif yang dapat dicetak sebagai rekomendasi bacaan mingguan di area perpustakaan fisik.</p>

E. Mockup

Bagian ini menyajikan gambaran sketsa antarmuka dalam bentuk mockup. Pembuatan mockup dapat menggunakan Balsamiq atau perangkat lain sesuai kebutuhan.

E.1. Ketua Kelompok – Menu #1 (102022400122 – Michael William Setiawan Wee)

Bagian ini menggambarkan lima pembahasan utama sebagaimana diuraikan pada Poin D, yang disusun oleh setiap mahasiswa, baik ketua maupun anggota. Uraian dilengkapi dengan ilustrasi mockup tata letak menu yang dikembangkan, contoh data yang ditampilkan, rancangan operasi CRUD, pemanfaatan API, serta komponen lain yang dinilai penting untuk ditampilkan.



Tuliskan uraian pada bagian ini.

MENU REGISTER, LOGIN, LOGOUT, SESSION

Bagian Login:

TATA LETAK MENU:

1. Struktur halaman Login dengan elemen: Judul "Login", instruksi "Masuk ke akun Anda", input Email, input Password, tombol Login, link Lupa Password
2. Struktur halaman Daftar dengan: Judul "Daftar", deskripsi, tombol "Register Sekarang!"
3. Pemisahan visual menggunakan garis horizontal

1. Wireframe lengkap semua halaman:

- Login Page
- Register Page
- User Dashboard
- Admin Dashboard
- Forgot Password Page

2. Layout komponen:

- Header/Navigation



- Sidebar menu
- Main content area
- Form controls
- Action buttons

3. User flow diagram:

- Login → Dashboard
- Register → Login
- Forgot Password → Reset

CONTOH DATA YANG DITAMPILKAN:

1. Label: "Email", "Password"
2. Teks tombol: "Login", "Lupa password?", "Register Sekarang!"
3. Instruksi pengguna: "Masuk ke akun Anda", "Belum punya akun?..."

RANCANGAN OPERASI CRUD:

Ilustrasi: Alur Create, Read, Update, Delete untuk setiap entitas

1. Create: Tombol "Register Sekarang!" (membuat user baru)
2. Read: Form login membaca data user dari database
3. Update: (implisit) Update session status saat login/logout
4. Delete: (implisit) Hapus session saat logout

Untuk Tabel USERS:

1. CREATE (Register)

1. Input: name, email, password, confirm_password
2. Validasi: email unique, password strength
3. Default: role='user', created_at=now()

2. READ (Login & Profile)

1. SELECT * FROM users WHERE email = ? AND password = ?
2. Return: id, name, email, role, avatar

3. UPDATE (Edit Profile)

1. Admin: bisa update role, status
2. User: bisa update name, avatar, password
3. SET updated_at = now()

4. DELETE (User Management)

1. Soft delete: update status='deleted'



2. Atau hard delete dengan konfirmasi

PEMANFAATAN API :

1. Authentication Service API Endpoints:
2. POST /api/auth/login
3. GET /api/auth/me
4. POST /api/auth/forgot-password

KOMPONEN LAIN :

- Security concern: Password field (harus disembunyikan)
- User guidance: Teks instruksi yang jelas
- Error handling: (implisit) Validasi input

REGISTER:

TATA LETAK MENU:

Halaman Daftar dirancang dengan tata letak vertikal yang terstruktur dan intuitif. Di bagian atas terdapat judul "Daftar" dengan ukuran font besar sebagai header utama, dilanjutkan dengan kalimat pengantar "Buat akun baru Anda". Formulir pendaftaran menampilkan empat field input yang masing-masing ditandai dengan indikator kotak centang: Nama Lengkap, Email, Password, dan Ulangi Password.



Setiap field memiliki placeholder yang jelas. Di bawah formulir terdapat garis pemisah horizontal, kemudian tombol aksi utama "Buat Akun" dengan penekanan visual. Bagian bawah halaman menampilkan pesan "Sudah Punya Akun?" beserta penjelasan singkat, diakhiri dengan tautan "Login Sekarang!" yang mengarahkan pengguna ke halaman login. Alur pengguna dimulai dari halaman beranda atau login, mengisi formulir pendaftaran, kemudian setelah berhasil akan diarahkan otomatis ke halaman login dengan email yang sudah terisi.

CONTOH DATA YANG DITAMPILKAN:

Pengguna akan mengisi data pribadi seperti nama lengkap (contoh: "Michael William Setiawan Wee"), alamat email (contoh: "michael@gmail.com"), password (contoh: "michael123!"), dan konfirmasi password. Sistem akan menampilkan validasi real-time, seperti centang hijau ketika format email valid atau pesan "Email sudah terdaftar" jika tidak tersedia. Untuk password, akan muncul indikator kekuatan yang menunjukkan tingkat keamanan (lemah/sedang/kuat) beserta checklist persyaratan seperti minimal 8 karakter, mengandung huruf besar dan kecil, angka, serta simbol. Setelah pendaftaran berhasil, data disimpan ke tabel users dengan password terenkripsi, role default "user", dan timestamp pendaftaran.

RANCANGAN OPERASI CRUD:

Operasi utama pada halaman Daftar adalah CREATE (membuat data pengguna baru). Proses dimulai dengan validasi input: nama harus minimal 3 karakter, email harus unik dan berformat valid, password harus memenuhi kriteria keamanan, dan konfirmasi password harus sama. Jika validasi gagal, sistem menampilkan pesan error spesifik di setiap field. Jika sukses, password di-hash menggunakan algoritma bcrypt sebelum disimpan ke database bersama data lain seperti role default "user" dan timestamp. Sistem juga membuat log aktivitas pendaftaran yang mencatat IP address dan perangkat pengguna. Tidak ada operasi READ, UPDATE, atau DELETE langsung pada halaman ini, tetapi admin dapat mengelola data pengguna melalui dashboard terpisah.

PEMANFAATAN API:

Halaman Daftar Akun menggunakan API internal untuk berbagai fungsi. Endpoint utama adalah POST `/api/auth/register` yang menerima data nama, email, password, dan konfirmasi password, kemudian mengembalikan respons sukses dengan data pengguna dan redirect ke login, atau respons error dengan detail validasi gagal. API pendukung lain termasuk GET `/api/auth/check-email` untuk memeriksa ketersediaan email secara real-time, dan POST `/api/auth/check-password-strength` untuk mengevaluasi kekuatan password. Semua komunikasi API menggunakan token autentikasi dan enkripsi HTTPS. API juga terintegrasi dengan layanan email untuk mengirim welcome message dan dengan sistem analitik untuk melacak sumber pendaftaran.

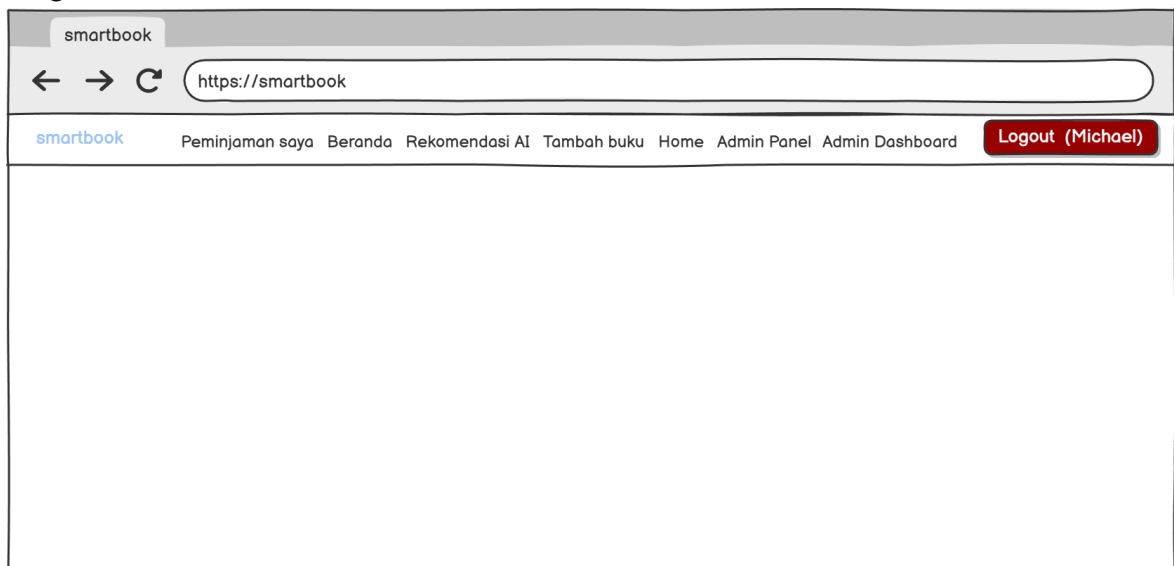
KOMPONEN PENTING LAINNYA:



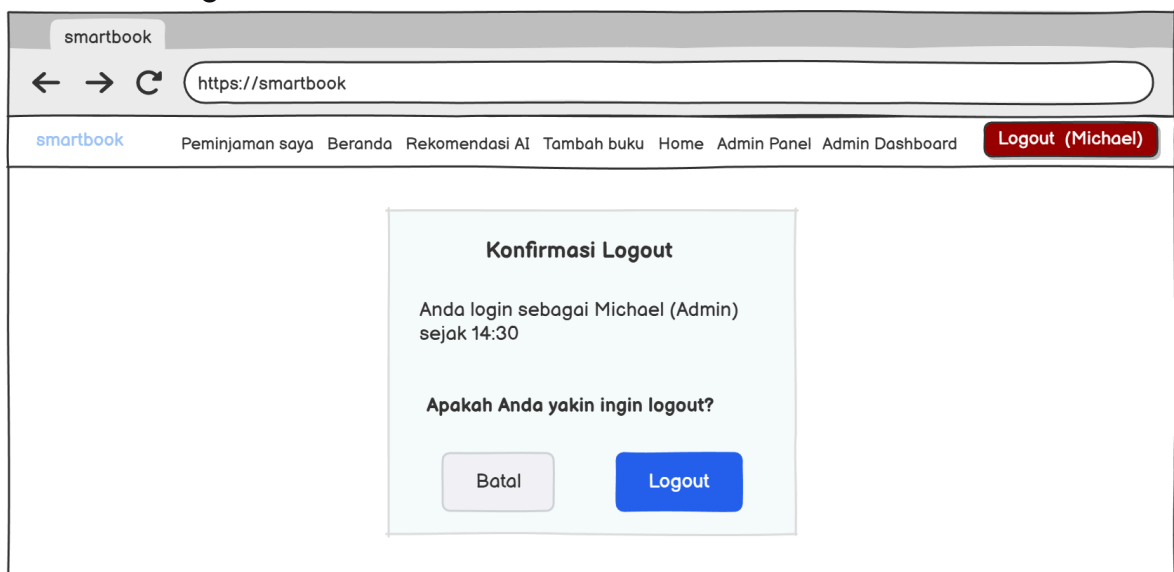
Komponen keamanan meliputi hashing password dengan bcrypt, proteksi CSRF token, rate limiting (maksimal 5 percobaan, jika salah password atau lainnya maka diharuskan menunggu selama 30 detik), dan sanitation input untuk mencegah XSS. Sistem validasi mencakup pengecekan format email, blacklist domain email sementara, serta aturan kompleksitas password. Penanganan error dirancang untuk memberikan pesan yang jelas dan membantu, seperti "Password harus mengandung huruf besar" atau "Email sudah terdaftar". Fitur pengalaman pengguna termasuk loading indicator saat proses pendaftaran, autofocus pada field berikutnya, dan pengisian otomatis email di halaman login setelah pendaftaran berhasil. Setelah pendaftaran, sistem secara otomatis membuat akun pengguna.

LOGOUT:

-bagian admin

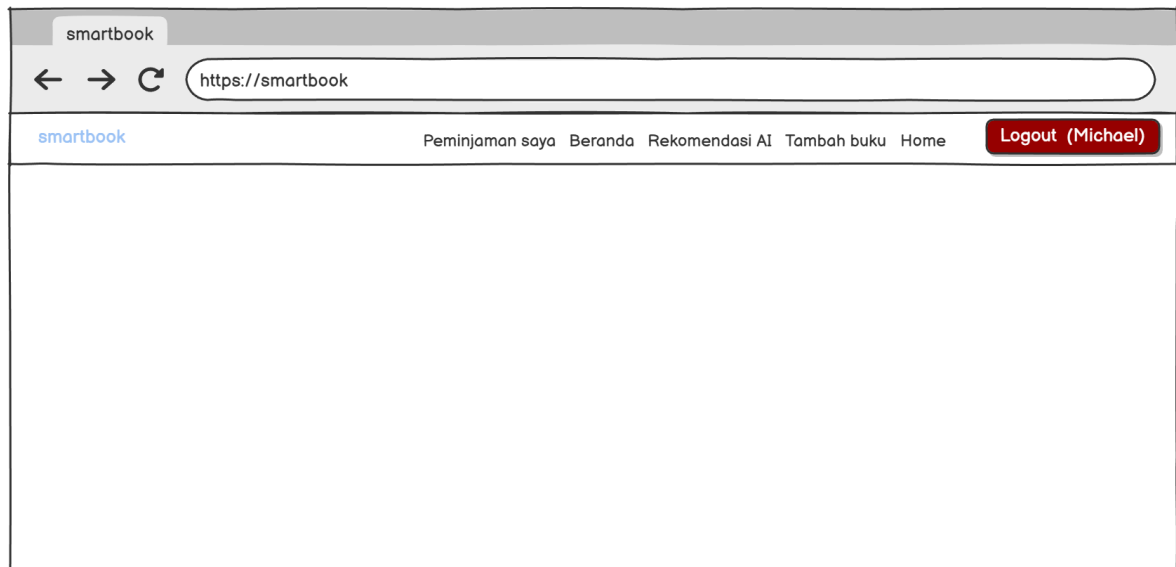


-konfirmasi logout admin

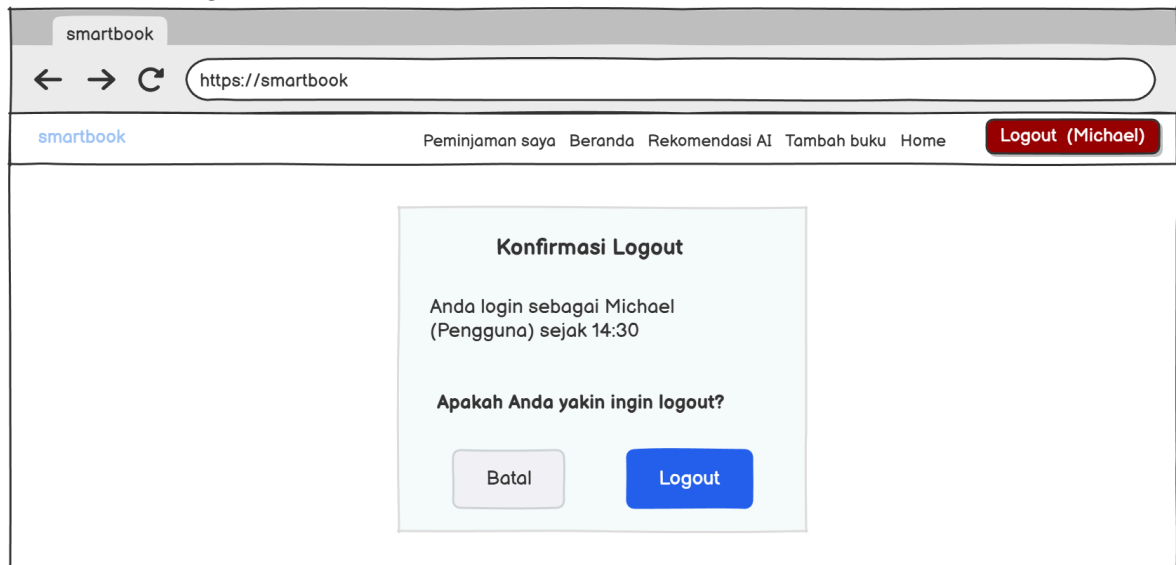




-BAGIAN USER BIASA



-konfirmasi logout



TATA LETAK:

Tombol Logout ditempatkan di bagian paling kanan atas sidebar navigasi dashboard, terpisah dari menu-menu fungsional lainnya dengan jarak visual yang jelas. Tombol ini menampilkan teks "Logout" diikuti nama pengguna dalam tanda kurung "(Michael)" untuk personalisasi dan konfirmasi identitas. Posisinya yang strategis di bagian bawah mencegah kesalahan klik, sementara tetap mudah diakses. Ketika diklik, tombol ini akan memicu modal konfirmasi atau langsung menjalankan proses logout tergantung pengaturan keamanan. Desain



menggunakan warna atau ikon yang berbeda (biasanya merah atau abu-abu) untuk membedakannya dari tombol aksi lainnya.

CONTOH DATA YANG DITAMPIKAN:

Saat pengguna mengklik logout, sistem mencatat data session yang akan diakhiri: session_id (contoh: "sess_abc123xyz456"), user_id (101), waktu logout (2024-01-15 15:45:30), durasi session (75 menit), dan aktivitas terakhir. Sistem juga menampilkan notifikasi konfirmasi berisi informasi "Anda login sebagai Michael sejak 14:30" dan konfirmasi "Apakah Anda yakin ingin logout?". Setelah logout berhasil, ditampilkan pesan "Logout berhasil. Anda akan diarahkan ke halaman login dalam 3 detik" beserta timer countdown. Data statistik yang dicatat meliputi total waktu session, jumlah aksi yang dilakukan, dan perangkat yang digunakan.

RANCANGAN OPERASI CRUD:

Operasi utama pada logout adalah UPDATE dan DELETE. Operasi UPDATE mengubah status session dari "active" menjadi "inactive" di tabel sessions, mengisi kolom logout_time dengan timestamp saat ini, dan memperbarui durasi session. Operasi DELETE menghapus token autentikasi dari penyimpanan client-side (localStorage/cookies) dan menghapus data session dari memory server. Sistem juga melakukan operasi READ terakhir untuk mencatat statistik session sebelum diakhiri. Tidak ada operasi CREATE pada proses logout, tetapi sistem dapat membuat log aktivitas logout di tabel audit logs untuk keperluan monitoring keamanan.

PEMANFAATAN API:

Endpoint utama logout adalah POST /api/auth/logout yang menerima token autentikasi di header Authorization. API ini akan memvalidasi token, mengupdate status session di database, membatalkan token JWT (jika menggunakan blacklist), dan mengembalikan respons sukses. Endpoint pendukung termasuk POST /api/auth/logout-all untuk mengakhiri semua session aktif pengguna dari berbagai perangkat, dan GET /api/auth/session/validate untuk memeriksa validitas token sebelum proses logout. Semua request logout wajib menyertakan CSRF token untuk mencegah serangan logoutsilang. API juga mengirimkan notifikasi ke sistem monitoring tentang aktivitas logout.

KOMPONEN LAINNYA:

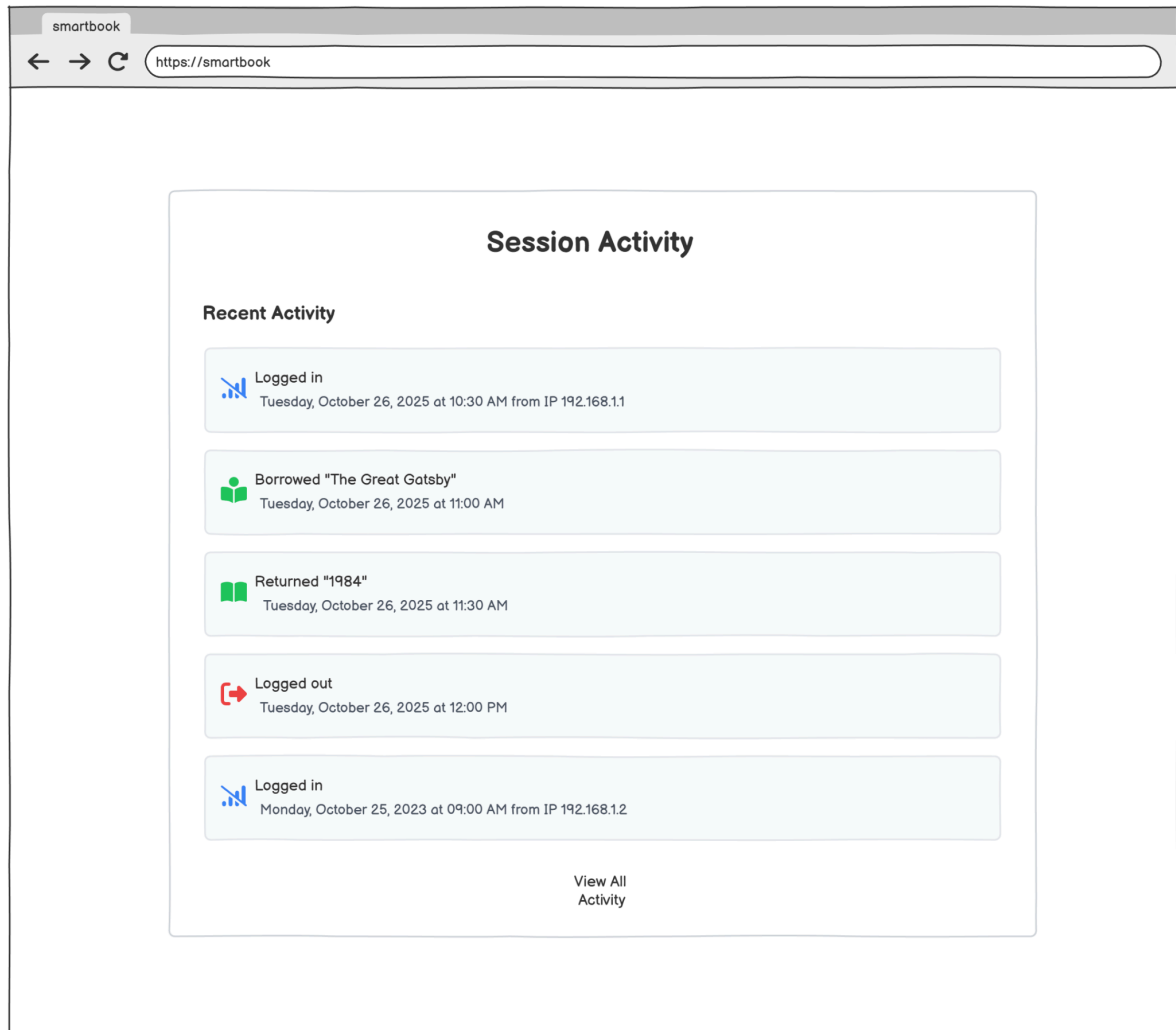
Keamanan logout meliputi: invalidasi token sepenuhnya (bukan hanya menghapus dari client), pencatatan audit trail dengan IP address dan alasan logout, pencegahan replay attack dengan token sekali pakai, dan konfirmasi sebelum logout jika ada perubahan data yang belum disimpan. Fitur user experience termasuk: auto-redirect ke halaman login setelah 3 detik, opsi "Batalkan" pada modal konfirmasi, feedback visual selama proses logout, dan remember me untuk login kembali. Sistem juga menyediakan fitur "Logout dari Semua Perangkat" untuk keamanan tambahan, notifikasi email jika logout dari perangkat baru, dan



pembersihan cache sensitif setelah logout. Untuk admin, tersedia fitur force logout pengguna lain dari dashboard monitoring session.

SESSION:

TATA LETAK:



Halaman Session Activity memiliki desain yang bersih dan terstruktur. Di bagian atas terdapat header utama "# Session Activity" yang menunjukkan fokus halaman. Di bawahnya, subheader "## Recent Activity" mengelompokkan aktivitas terkini. Konten utama berupa daftar aktivitas dalam format bullet points dengan timeline kronologis terbalik (terbaru di atas). Setiap aktivitas menampilkan: jenis aksi (Logged in, Borrowed, Returned, Logged out), detail spesifik (judul buku untuk peminjaman/pengembalian, IP address untuk login), tanggal dan waktu lengkap dengan hari, tanggal, bulan, tahun, dan jam, serta informasi tambahan seperti alamat IP. Di bagian bawah daftar, terdapat tautan "View All Activity" yang berfungsi sebagai navigasi ke halaman riwayat lengkap. Layout ini memprioritaskan readability dengan spacing yang cukup antara item.

CONTOH DATA YANG DITAMPILKAN:



Data yang ditampilkan mencakup empat jenis aktivitas: Login (dengan timestamp "Tuesday, October 26, 2025 at 10:30 AM" dan IP address "192.168.11"), Peminjaman Buku ("Borrowed 'The Great Gatsby' at 11:00 AM"), Pengembalian Buku ("Returned '1984' at 11:30 AM"), dan Logout ("Logged out at 12:00 PM"). Setiap entri menunjukkan pola waktu yang logis: login → meminjam → mengembalikan → logout. Data juga menunjukkan aktivitas dari hari sebelumnya ("Monday, October 25, 2025 at 09:00 AM") dengan IP berbeda ("192.168.12"), mengindikasikan akses dari perangkat atau lokasi berbeda. Format data konsisten dengan detail hari, tanggal, bulan, tahun, dan jam yang lengkap.

RANCANGAN OPERASI CRUD:

Operasi utama pada halaman ini adalah READ untuk menampilkan riwayat aktivitas. Sistem membaca data dari tabel `activity_logs` dengan query: `SELECT * FROM activity_logs WHERE user_id = ? ORDER BY timestamp DESC LIMIT 5`. Data diambil berdasarkan `user_id` dari session yang sedang aktif, diurutkan dari terbaru, dan dibatasi 5 item untuk tampilan awal. Operasi CREATE terjadi di latar belakang setiap kali pengguna melakukan aksi (login, logout, borrow, return) yang menambahkan record baru ke tabel `activity_logs`. Tidak ada operasi UPDATE atau DELETE yang tersedia untuk pengguna biasa di halaman ini, tetapi admin mungkin memiliki kemampuan untuk mengelola log. Tautan "View All Activity" mengarah ke operasi READ yang lebih komprehensif dengan pagination.

PEMANFAATAN API:

API endpoint utama adalah GET `/api/activity/recent` yang mengembalikan 5 aktivitas terbaru dalam format JSON. Response mencakup: `{"activities": [{"id": 1, "type": "login", "description": "Logged in from IP 192.168.11", "timestamp": "2025-10-26T10:30:00Z", "metadata": {"ip": "192.168.11", "device": "Chrome/Windows"}}, ...], "total": 15, "view_all_url": "/activity/full"}`. Endpoint GET `/api/activity/full` menyediakan akses ke seluruh riwayat dengan parameter pagination (`?page=1&limit=20`). API POST `/api/activity/log` digunakan internal sistem untuk mencatat aktivitas baru setiap event terjadi. Untuk keamanan, semua endpoint activity memerlukan autentikasi dan hanya mengembalikan data milik pengguna yang bersangkutan (kecuali untuk admin).

KOMPONEN PENTING LAIN:

Sistem pencatatan aktivitas mencakup komponen audit trail yang mencatat setiap aksi penting dengan timestamp, IP address, user agent, dan status. Keamanan diimplementasikan dengan validasi ownership data, sanitasi output untuk mencegah XSS, dan pembatasan akses log sensitif. Fitur pengguna termasuk filter berdasarkan jenis aktivitas (hanya login, hanya peminjaman, dll.), pencarian berdasarkan tanggal atau kata kunci, dan ekspor data ke format CSV atau PDF. Optimasi performa menggunakan database indexing pada kolom `user_id` dan `timestamp`, caching untuk data yang tidak sering berubah, dan lazy loading untuk riwayat lengkap. Notifikasi dapat diintegrasikan untuk aktivitas mencurigakan



seperti login dari IP yang tidak dikenal. Visualisasi data tambahan dapat menunjukkan statistik seperti "Rata-rata waktu session: 90 menit" atau "Buku paling sering dipinjam".

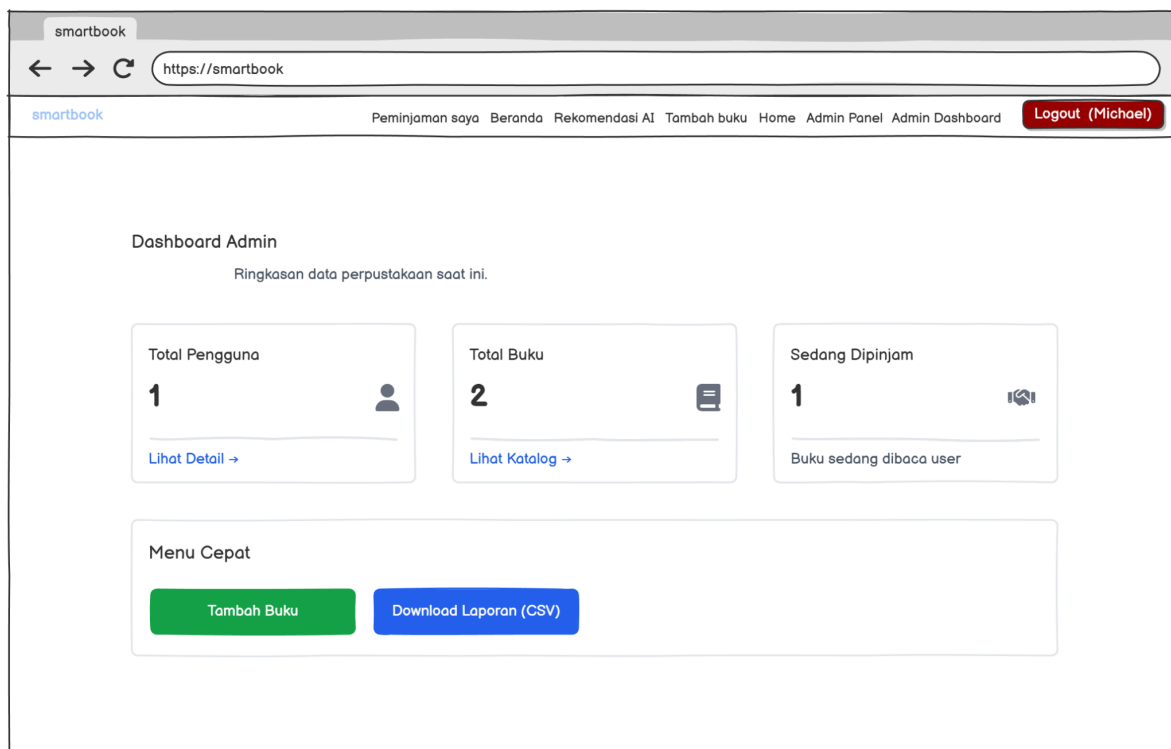
Tuliskan uraian pada bagian ini.

MENU DASHBOARD DAN MANAJEMEN USER

Bagian Login:

TATA LETAK MENU:

-DASHBOARD ADMIN



Dashboard Admin SmartBook memiliki layout terstruktur dengan navigation bar horizontal di bagian atas yang menampilkan logo "smartbook" di kiri dan delapan menu navigasi di kanan: Peminjaman saya, Beranda, Rekomendasi AI, Tambah buku, Home, Admin Panel, Admin Dashboard, dan Logout (Michael) yang dipersonalisasi. Di bawah navigation bar, terdapat header utama "Dashboard Admin" dengan deskripsi "Ringkasan data perpustakaan saat ini." Konten utama dibagi menjadi dua bagian: sebelah kiri menampilkan tiga kartu statistik dalam layout grid untuk Total Pengguna (1), Total Buku (2), dan Sedang Dipinjam (1) - masing-masing dengan tombol "Lihat Detail →" atau "Lihat Katalog →". Sebelah kanan menampilkan panel "Menu Cepat" dengan dua tombol aksi besar: "Tambah Buku" dan "Download Laporan (CSV)". Halaman diakhiri footer dengan copyright "© 2025 BookJournal. Dibuat dengan" dan dropdown indicator (▼).

CONTOH DATA YANG DITAMPILOKAN



Data yang ditampilkan pada dashboard admin mencakup metrik sistem: Total Pengguna = 1, Total Buku = 2, dan Buku Sedang Dipinjam = 1. Data ini real-time diambil dari database dengan query aggregate: `SELECT COUNT(*) FROM users WHERE role='user'`, `SELECT COUNT(*) FROM books WHERE status='available'`, dan `SELECT COUNT(*) FROM borrowings WHERE return_date IS NULL`. Setiap kartu statistik memiliki konteks tambahan seperti "Buku sedang dibaca user" untuk membantu interpretasi. Data pengguna Michael sebagai admin menunjukkan role-based access dengan kemampuan melihat semua data sistem. Menu cepat menyediakan akses langsung ke fungsi administratif dengan data contoh seperti format laporan CSV yang berisi kolom: Tanggal, Jenis Aktivitas, User ID, Buku ID, dan Status.

RANCANGAN OPERASI CRUD

Dashboard admin mengkonsolidasikan semua operasi CRUD untuk manajemen sistem. READ operasi menampilkan statistik agregat dari berbagai tabel (users, books, borrowings) dengan query optimized. CREATE operasi diakses melalui tombol "Tambah Buku" yang membuka form untuk menambahkan buku baru ke katalog dengan field: ISBN, Judul, Penulis, Kategori, Stok, dan Deskripsi. UPDATE operasi tersedia melalui link "Lihat Detail →" yang mengarah ke halaman edit pengguna atau "Lihat Katalog →" untuk edit buku. DELETE operasi diimplementasikan dengan soft delete melalui antarmuka manajemen yang terpisah dengan konfirmasi sebelum penghapusan. Dashboard juga mendukung operasi batch seperti download semua data dalam CSV melalui "Download Laporan (CSV)" yang melakukan export data dari multiple tables.

PEMANFAATAN API

Dashboard admin menggunakan API khusus dengan endpoint: GET /api/admin/dashboard/stats untuk mengambil statistik real-time, GET /api/admin/users untuk data pengguna dengan pagination, GET /api/admin/books untuk manajemen katalog, dan GET /api/admin/borrowings untuk monitoring peminjaman. API export POST /api/admin/export/csv menghasilkan laporan CSV dengan parameter format (users, books, borrowings, all). API create POST /api/admin/books untuk menambah buku baru dengan multipart form data termasuk upload cover image. Semua endpoint admin memerlukan autentikasi dengan role="admin" dan rate limiting yang ketat. WebSocket connection /ws/admin/updates menyediakan real-time updates untuk statistik tanpa refresh halaman.

KOMPONEN LAINNYA

Keamanan admin dashboard meliputi: dua faktor autentikasi untuk login admin, session timeout 15 menit, audit log untuk semua aksi administratif, dan IP whitelisting untuk akses admin. Fitur monitoring real-time mencakup: live counter untuk aktifitas pengguna, alert system untuk anomaly detection (seperti spike peminjaman), dan system health dashboard. Komponen utilitas termasuk: advanced search dengan filter multi-kriteria, bulk operations untuk manajemen



data massal, data visualization dengan chart.js untuk trend analysis, dan scheduled report generation. Backup system otomatis membuat snapshot data harian dengan opsi restore. Integration dengan analytics tools seperti Google Analytics untuk tracking penggunaan sistem. Responsive design memastikan dashboard dapat diakses dari tablet dan mobile dengan layout yang menyesuaikan.

-BAGIAN MANAJEMEN USER - ADMIN PANEL

The screenshot displays the 'Smart Book - Admin Panel' interface. At the top, there's a navigation bar with links: 'Peminjaman saya', 'Beranda', 'Rekomendasi AI', 'Tambah buku', 'Home', 'Admin Panel', and 'Admin Dashboard'. A 'Logout (Michael)' button is on the right. The main content area is titled 'Smart Book - Admin Panel' and 'Manajemen Pengguna'. Below this, it says 'Kelola akun pengguna, informasi login, logout, dan perangkat.' There's a 'Daftar Pengguna' section with a 'Tambah Pengguna' button. A table lists three users:

ID	Nama Pengguna	Email	Role	Terakhir Login	Terakhir Logout	Perangkat	Aksi
1	john_doe	john.doe@example.com	User	2023-11-19 10:30	2023-11-19 11:00	Desktop Chrome	Edit Hapus
2	jane_smith	jane.smith@example.com	User	2023-11-18 14:00	2023-11-18 15:30	Mobile Safari	Edit Hapus
3	Michael	michael@gmail.com	Admin	2025-11-19 09:00	2025-11-19 12:00	Desktop Firefox	Edit Hapus



The mockup shows a web browser window with the URL `https://smartbook`. The page title is "Edit Informasi Akun Pengguna". The form contains the following fields:

- Edit User**
 - Nama: John Doe
 - Email: john.doe@example.com
 - Role: Administrator
 - Status: Active
- User Details**

Field	Value
User ID:	12345
Last Login:	2025-10-27 10:30 AM
Created At:	2025-01-15 09:00 AM

Buttons: Save, Cancel

Footer: Akun Pengguna

-MOCKUP TATA LETAK MENU

Halaman Manajemen Pengguna SmartBook memiliki desain yang terstruktur dan profesional. Di bagian atas terdapat header utama "# Smart Book" sebagai identitas sistem, diikuti subheader "## Manajemen Pengguna" dengan deskripsi fungsi "Kelola akun pengguna, informasi login, logout, dan perangkat." Garis horizontal pemisah membatasi header dengan konten utama. Bagian inti halaman menampilkan tabel "Daftar Pengguna" dengan delapan kolom: ID, Nama Pengguna, Email, Role, Terakhir Login, Terakhir Logout, Perangkat, dan Aksi. Tabel menggunakan garis vertikal dan horizontal yang jelas untuk memisahkan kolom dan baris. Setiap baris data memiliki tombol "Edit" di kolom terakhir untuk aksi manajemen. Layout ini mengoptimalkan ruang vertikal untuk menampilkan maksimal data tanpa scrolling berlebihan. Selain itu terdapat bagian lanjutan ketika kita menekan tombol edit.

-CONTOH DATA YANG DITAMPILKAN

Tabel menampilkan tiga contoh pengguna dengan data lengkap: John Doe (ID: 1) dengan username "john_doe", email "john.doe@example.com", role "User", login terakhir "2023-11-19 10:30", logout terakhir "2023-11-19 11:00", perangkat "Desktop Chrome". Jane Smith (ID: 2) dengan username "jane_smith", email "jane.smith@example.com", role "User", login "2023-11-18 14:00", logout "2023-11-18 15:30", perangkat "Mobile Safari". Admin User (ID: 3) dengan



username "michael", email "michael@gmail.com", role "Admin", login "2025-11-19 09:00", logout "2025-11-19 12:00", perangkat "Desktop Firefox". Format data konsisten dengan timestamp menggunakan format "YYYY-MM-DD HH:MM" dan deskripsi perangkat menggabungkan device type dan browser.

Selain itu terdapat bagian edit yang menampilkan kolom untuk mengedit seperti nama, role dan lainnya.

-RANCANGAN OPERASI CRUD

Halaman ini mengkonsolidasi semua operasi CRUD untuk manajemen pengguna. READ dilakukan melalui query: `SELECT u.id, u.username, u.email, u.role, s.login_time, s.logout_time, s.device_info FROM users u LEFT JOIN sessions s ON u.id = s.user_id AND s.status = 'inactive' ORDER BY u.id`. CREATE pengguna baru tersedia melalui tombol "Tambah Pengguna" (implisit) di luar tabel. UPDATE diakses melalui tombol "Edit" di setiap baris yang membuka modal/form edit dengan field: username, email, role, status akun, dan reset password. DELETE implementasi menggunakan soft delete dengan mengubah status akun menjadi "nonaktif" atau toggle switch di halaman edit. Operasi batch seperti export data pengguna dan filter berdasarkan role/login status juga tersedia.

-PEMANFAATAN API

API endpoints untuk manajemen pengguna: GET `/api/admin/users` dengan parameter pagination (`?page=1&limit=20&role=user&search=john`), GET `/api/admin/users/{id}` untuk detail pengguna spesifik, POST `/api/admin/users` untuk membuat pengguna baru, PUT `/api/admin/users/{id}` untuk update data, DELETE `/api/admin/users/{id}` untuk soft delete, dan POST `/api/admin/users/{id}/reset-password` untuk reset password. API GET `/api/admin/users/export` menghasilkan CSV/Excel dengan semua data pengguna. WebSocket endpoint `/ws/admin/user-activity` memberikan real-time update untuk kolom login/logout terakhir. Semua endpoint memerlukan header `X-Admin-Token` dan validasi role admin.

-KOMPONEN LAIN

Fitur keamanan mencakup: audit log untuk setiap perubahan data pengguna, two-person approval untuk perubahan role ke admin, rate limiting untuk aksi massal, dan session validation sebelum operasi kritis. Fitur pencarian dan filter meliputi: search by username/email, filter by role (user/admin), filter by status (aktif/nonaktif), filter by last login date range, dan filter by device type. Fitur manajemen batch: select multiple users untuk aksi massal (aktif/nonaktif, reset password, assign role), import pengguna dari CSV, dan export data dengan custom columns. Monitoring features: last activity indicators dengan warna (hijau untuk online, abu-abu untuk offline >24 jam), login attempt history, dan device management untuk revoke access perangkat tertentu. Integration dengan email service untuk notifikasi perubahan akun dan welcome email untuk pengguna baru. Responsive table design dengan horizontal scroll untuk mobile view dan option untuk compact view.



E.2. Anggota Kelompok 1 – Menu #2 (102022400020 – Muhammad Lukman Hakim)

Bagian ini menggambarkan lima pembahasan utama sebagaimana diuraikan pada Poin D, yang disusun oleh setiap mahasiswa, baik ketua maupun anggota. Uraian dilengkapi dengan ilustrasi mockup tata letak menu yang dikembangkan, contoh data yang ditampilkan, rancangan operasi CRUD, pemanfaatan API, serta komponen lain yang dinilai penting untuk ditampilkan.

Menu Pencarian dan pengelompokan buku berdasarkan kategori

Menampilkan halaman utama pengelompokan buku berdasarkan kategori

menampilkan halaman cari dan tambahkan buku

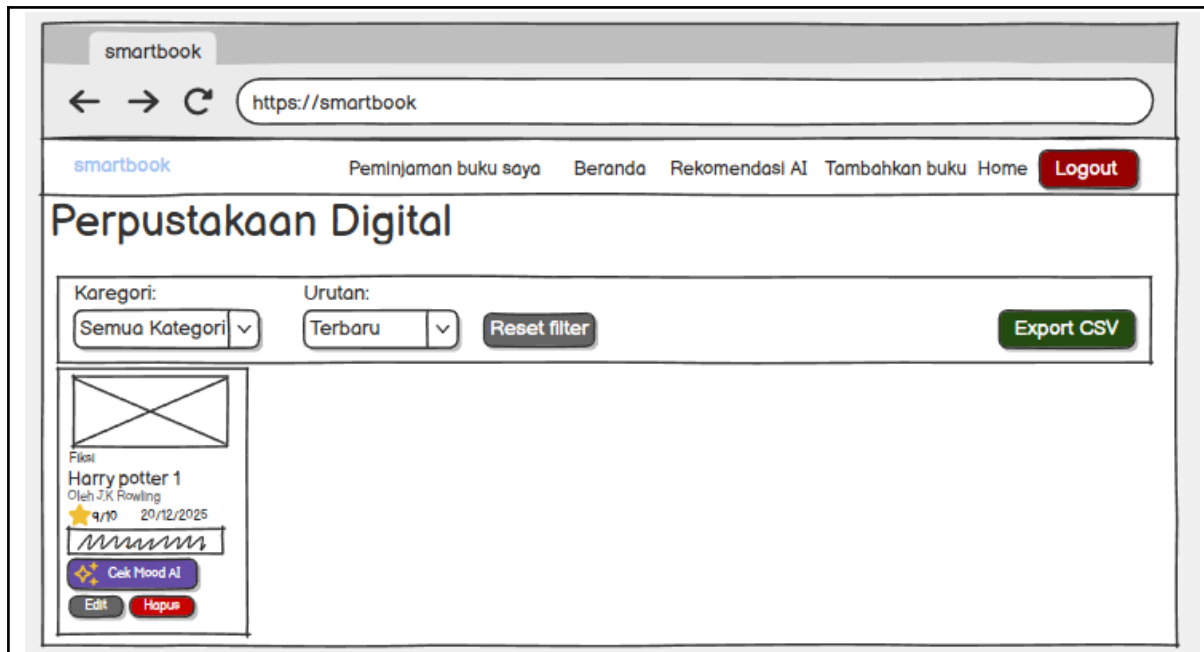


The screenshot shows a web browser window with the address bar displaying "https://smartbook". The page has a header with the "smartbook" logo and navigation links: "Peminjaman buku saya", "Beranda", "Rekomendasi AI", "Tambahkan buku", "Home", and a red "Logout" button. The main content area features a box titled "Cari & Tambah buku baru" with the subtitle "Cari buku berdasarkan judul untuk ditambahkan". Inside this box is a text input field labeled "Judul Buku" and a blue "Cari" button.

menampilkan buku yang sudah dicari dan bisa menambahkannya ke jurnal

The screenshot shows the same web browser window, but now displaying search results. The header and navigation links are identical. The main content area shows two book entries side-by-side. Each entry includes a placeholder image (a box with an 'X'), the title "Harry potter 1" and "Harry potter 2", the author "Oleh J.K Rowling", and the category "Fiksi" in a dropdown menu. Below the category is a "Rating" field with values "9" and "8" respectively. There is a "Catatan singkat" field with a wavy line icon. At the bottom of each entry is a "Tanggal Selesai Baca" field with dates "20/12/2025" and "29/12/2025" and a calendar icon. A blue button "Tambahkan ke jurnal" is located at the bottom of each book entry.

menampilkan halaman yang sudah ditambahkan buku ke jurnal



TATA LETAK:

menampilkan header berisi judul Perpustakaan Digital dan menu Beranda, Cari Buku, serta Jurnal, dengan area konten berupa sampul buku yang dikelompokkan berdasarkan kategori. Pada halaman pencarian, pengguna mengisi judul buku melalui kolom pencarian di bagian atas, lalu hasilnya ditampilkan dalam bentuk kartu sampul buku. Buku yang dipilih dapat langsung ditambahkan ke jurnal dengan opsi mengatur rating, menulis catatan singkat, dan menandai status selesai membaca. Tahap akhir menampilkan halaman jurnal berisi daftar buku yang telah disimpan, lengkap dengan informasi ringkas dan tombol ubah serta hapus, sehingga pengguna dapat mengelola koleksi buku secara mudah dan efisien.

Contoh data yang ditampilkan:

berupa informasi buku yang disajikan secara ringkas dan visual, meliputi sampul buku, judul Harry Potter, nama penulis J.K. Rowling, kategori fiksi, bahasa Inggris, serta rating 9. Pada tampilan detail dan jurnal, sistem juga menampilkan sinopsis singkat mengenai isi buku, catatan pribadi pengguna, serta status membaca (selesai atau belum selesai), sehingga pengguna dapat mengenali, menilai, dan mengelola buku dengan mudah.

Rancangan Operasi CRUD:

- Create dilakukan ketika pengguna menambahkan buku hasil pencarian ke jurnal, disertai pengisian rating, catatan singkat, dan status membaca.
- Read memungkinkan pengguna melihat daftar buku yang telah disimpan beserta detail seperti judul, penulis, sampul, rating, dan catatan.
- Update digunakan untuk mengubah data tambahan pada buku yang tersimpan, seperti memperbarui rating, catatan, atau status selesai membaca.
- Delete memungkinkan pengguna menghapus buku dari jurnal apabila sudah tidak diperlukan.



pemanfaatan API :

Pemanfaatan API dilakukan dengan mengintegrasikan Google Books API sebagai sumber data buku eksternal. API ini digunakan pada fitur pencarian buku untuk mengambil data secara real-time berdasarkan judul yang dimasukkan pengguna. Informasi yang diperoleh meliputi judul, penulis, tahun terbit, sinopsis, rating, dan sampul buku. Data dari API ditampilkan langsung pada hasil pencarian dan hanya disimpan ke database lokal ketika pengguna menambahkan buku ke jurnal, sehingga sistem tetap efisien dan data yang tersimpan relevan dengan kebutuhan pengguna.

komponen lain yang dinilai penting untuk ditampilkan.

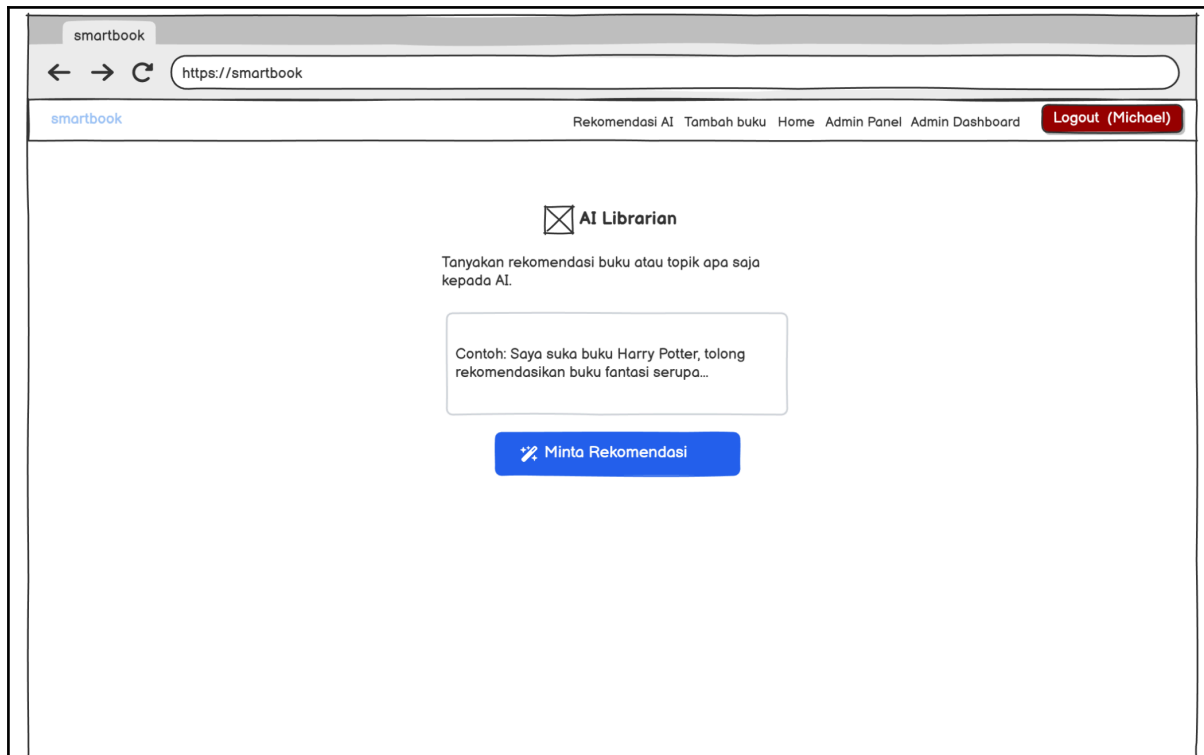
fitur filter buku berdasarkan kategori, bahasa, dan tahun terbit agar pencarian lebih cepat dan tepat. Sistem juga menampilkan status buku, seperti sudah disimpan atau belum serta selesai dibaca atau belum. Selain itu, tersedia tombol aksi yang jelas seperti tambah, ubah, dan hapus, serta notifikasi sederhana untuk memberi tahu pengguna saat data berhasil disimpan atau dihapus. Tampilan sampul buku yang konsisten dan menu yang mudah dipahami membuat sistem lebih nyaman digunakan.

E.3. Anggota Kelompok 2 – Menu #3 (102022400157 – Farhan Davin Rinaldi)

Bagian ini menggambarkan lima pembahasan utama sebagaimana diuraikan pada Poin D, yang disusun oleh setiap mahasiswa, baik ketua maupun anggota. Uraian dilengkapi dengan ilustrasi mockup tata letak menu yang dikembangkan, contoh data yang ditampilkan, rancangan operasi CRUD, pemanfaatan API, serta komponen lain yang dinilai penting untuk ditampilkan.

MENU : REKOMENDASI AI

-MOCKUP TATA LETAK



Tata letak halaman ini sangat minimalis dan fokus pada fungsi utama, yaitu interaksi dengan AI:

- **Header Navigasi:** Berisi logo smartbook dan tautan navigasi utama: Rekomendasi AI (halaman aktif), Tambah buku, Home, Admin Panel, Admin Dashboard, dan aksi akun (Logout (Michael)).
- **Area Utama (AI Librarian):** Berisi judul layanan dan instruksi.
 - **Judul:** Ditampilkan sebagai AI Librarian dengan ikon di atasnya.
 - **Instruksi:** Kalimat petunjuk: "Tanyakan rekomendasi buku atau topik apa saja kepada AI."
 - **Input Area:** Kotak teks besar tempat pengguna mengetik permintaan atau pertanyaan mereka. Terdapat contoh yang membantu pengguna memahami format masukan yang diharapkan.
 - **Tombol Aksi:** Tombol berwarna biru berlabel Minta Rekomendasi yang berfungsi untuk memproses permintaan.

- DATA YANG DITAMPILKAN

Data yang ditampilkan pada *mockup* ini sebagian besar adalah *placeholder* dan instruksi untuk pengguna:

- **Contoh Prompt (Permintaan):** "Contoh: Saya suka buku Harry Potter, tolong rekomendasikan buku fantasi serupa..."
- **Informasi Akun:** Status *login* ditunjukkan oleh nama pengguna (Michael) dan tombol Logout.



- **Tautan Administratif:** Kehadiran Admin Panel dan Admin Dashboard menyiratkan bahwa pengguna Michael memiliki hak akses administratif.

-RANCANGAN CRUD

Halaman ini tidak secara langsung melakukan operasi CRUD (Create, Read, Update, Delete) pada data buku atau ulasan, melainkan berfungsi sebagai antarmuka layanan yang memicu pemrosesan di *backend*.

- **CREATE (C) & READ (R) Tidak Langsung:**
 - Pengguna menciptakan (*Create*) sebuah *query* (permintaan teks) di *input area*.
 - Sistem membaca (*Read*) *query* tersebut.
 - *Backend AI* memproses *query* tersebut dan kemudian menghasilkan data rekomendasi baru, yang merupakan hasil READ yang dimodifikasi.
 - Rekomendasi yang dihasilkan akan ditampilkan di layar (kemungkinan di halaman atau area baru, atau di bawah tombol).
- **UPDATE (U) & DELETE (D):** Operasi ini tidak relevan dalam konteks halaman input *query AI* ini.

-PEMANFAATAN API

Halaman AI Librarian sepenuhnya bergantung pada satu atau lebih API *backend* yang menangani pemrosesan bahasa alami (NLP) dan algoritma rekomendasi:

- **API Pemrosesan Permintaan (Query API):** Saat pengguna menekan Minta Rekomendasi, *frontend* akan mengirim permintaan teks pengguna ke *backend* menggunakan metode **POST**.
 - **Endpoint Contoh:** **POST /api/ai/recommendation**
 - **Payload (Data Terkirim):** Berisi teks permintaan pengguna dan ID pengguna yang meminta.
- **Layanan AI (Backend):** *Backend* menerima permintaan, memprosesnya menggunakan model AI (seperti Model Bahasa Besar atau mesin rekomendasi), dan mencari data buku yang relevan dari *database*.
- **API Respon:** *Backend* mengirimkan kembali respons (response) yang berisi daftar rekomendasi buku (Judul, Penulis, *Rating*, tautan) dalam format terstruktur (misalnya, JSON).

-KOMPONEN PENTING LAIN

Interaksi *Real-time* / *Asynchronous*: Karena pemrosesan AI mungkin memakan waktu beberapa detik, antarmuka harus menunjukkan status *loading* setelah tombol ditekan, mencegah pengguna menekan berulang kali, dan menunjukkan bahwa permintaan sedang diproses.

User Context: Permintaan rekomendasi oleh AI kemungkinan menggunakan konteks



pengguna yang *login* (Michael), seperti riwayat bacaan dan *rating* yang pernah diberikan, untuk memberikan hasil yang lebih personal.

Admin Access: Keberadaan tautan Admin Panel dan Admin Dashboard menunjukkan bahwa halaman ini juga menangani aspek otorisasi, menampilkan tautan hanya kepada pengguna yang memiliki peran superuser.

E.4. Anggota Kelompok 3– Menu #4 (102022400252 – M. Surya Ilyasa)

Bagian ini menggambarkan lima pembahasan utama sebagaimana diuraikan pada Poin D, yang disusun oleh setiap mahasiswa, baik ketua maupun anggota. Uraian dilengkapi dengan ilustrasi mockup tata letak menu yang dikembangkan, contoh data yang ditampilkan, rancangan operasi CRUD, pemanfaatan API, serta komponen lain yang dinilai penting untuk ditampilkan.

Fitur Event / Kegiatan Buku

Daftar Event				
ID	Nama Event	Tanggal	Lokasi	Deskripsi
1	Bedah Buku "Filosofi Teras"	2025-12-01	Perpustakaan Kota	Diskusi mendalam tentang buku Filosofi Teras.
2	Workshop Menulis Kreatif	2025-12-10	Pusat Komunitas	Pelatihan dasar menulis cerita pendek.
3	Pameran Buku Tahunan	2025-01-15	Telkom	Pameran buku dari berbagai penerbit.

TATA LETAK:

Antarmuka fitur dikembangkan dengan struktur navigasi atas (navbar) yang mencakup menu "Peminjaman buku saya", "Beranda", "Rekomendasi AI", "Tambahkan buku", dan "Home", serta tombol "Logout" yang mencolok. Menu utama yang sedang aktif menampilkan tabel "Daftar Event" dengan desain bersih dan minimalis. Data disajikan dalam kolom-kolom teratur yang mencakup ID, Nama Event, Tanggal, Lokasi, dan Deskripsi untuk memudahkan keterbacaan pengguna.

Contoh data yang ditampilkan:

Sistem menampilkan daftar kegiatan literasi secara terstruktur dalam sebuah tabel yang memuat informasi aktual mengenai acara-acara komunitas mendatang. Contoh data pertama menunjukkan sebuah kegiatan Bedah Buku "Filosofi Teras" yang dijadwalkan pada tanggal 2025-12-01 di Perpustakaan Kota, dengan deskripsi yang



berfokus pada diskusi mendalam mengenai isi buku tersebut. Selanjutnya, terdapat entitas data untuk Workshop Menulis Kreatif yang akan diselenggarakan pada 2025-12-10 berlokasi di Pusat Komunitas, yang bertujuan memberikan pelatihan dasar bagi penulis cerita pendek pemula. Terakhir, sistem menyajikan informasi mengenai Pameran Buku Tahunan yang bertempat di Telkom pada 2025-01-15, yang mencakup pameran koleksi buku dari berbagai penerbit terkemuka.

rancangan operasi CRUD :

Fitur ini mengelola data melalui operasi CRUD yang terstruktur dalam tabel events:

- Create: Menambahkan kegiatan baru seperti workshop atau bedah buku ke dalam sistem.
- Read: Menampilkan daftar event dalam bentuk tabel informasi seperti yang terlihat pada mockup.
- Update: Memperbarui informasi tanggal, lokasi, atau deskripsi jika terdapat perubahan jadwal.
- Delete: Menghapus data acara yang sudah tidak relevan atau dibatalkan.

Struktur Data: Menggunakan kolom id (Integer, PK), nama_event (Varchar), tanggal (Date), lokasi (Varchar), dan deskripsi (Text).

pemanfaatan API :

Sumber API: Sistem menggunakan API internal untuk melakukan pengambilan data (*fetching*) dari database ke antarmuka pengguna secara dinamis. Hal ini memastikan data yang tampil (seperti daftar event di Telkom atau Perpustakaan Kota) selalu sinkron dengan data di server.

Komponen Penting:

Selain tabel utama, komponen penting lainnya adalah integrasi dengan sistem autentikasi (tombol Logout) untuk keamanan akses, serta responsivitas bar pencarian URL yang menunjukkan aplikasi berbasis web.

E.5. Anggota Kelompok 4 – Menu #5 (102022400028 – Fadli Razzan Adabi)

Bagian ini menggambarkan lima pembahasan utama sebagaimana diuraikan pada Poin D, yang disusun oleh setiap mahasiswa, baik ketua maupun anggota. Uraian dilengkapi dengan ilustrasi mockup tata letak menu yang dikembangkan, contoh data yang ditampilkan, rancangan operasi CRUD, pemanfaatan API, serta komponen lain yang dinilai penting untuk ditampilkan.

MENU : RATING dan Ulasan
-MOCKUP TATA LETAK MENU



smartbook

← → ↺ https://smartbook

smartbook

Peminjaman saya Beranda Rekomendasi AI Tambah buku Home Logout (Michael)

Rating dan Ulasan Buku

Rating dan Ulasan Buku

Detail Buku

Judul Buku Contoh

Penulis Contoh

Penerbit Contoh

Genre:

Fiksi Ilmiah

Tahun Terbit:

2023

Rating Rata-rata:

★★★★☆ 4.5 (120 ulasan)

Berikan Ulasan Anda

Rating Anda:

★★★★☆

Judul Ulasan:

Ulasan Anda:

Batal

Kirim Ulasan

Ulasan Lainnya

Pengguna A ★★★★★ 2 hari yang lalu

Buku yang sangat menarik!

Saya sangat menikmati membaca buku ini. Alur ceritanya bagus dan karakter-karakternya kuat.

Pengguna B ★★★★★ 1 minggu yang lalu

Cukup bagus, tapi bisa lebih baik.

Ceritanya menarik, tapi ada beberapa bagian yang terasa lambat. Secara keseluruhan, pengalaman membaca yang lumayan.

Pengguna C ★★★★★ 3 minggu yang lalu

Wajib dibaca!

Salah satu buku terbaik yang pernah saya baca tahun ini. Sangat direkomendasikan!

Rating dan Ulasan Buku Terpercaya

Tata letak halaman "Rating dan Ulasan Buku" tersusun secara hierarkis, mengalir dari informasi umum ke interaksi spesifik:

- Header Navigasi: Berisi logo Smartbook dan tautan penting seperti



Perpustakaan Saya, Beranda, dan aksi akun (Logout).

- **Detail Buku:** Menempati bagian atas konten utama, menampilkan informasi inti buku (Judul, Penulis, Genre) dan hasil agregasi data (Rating Rata-rata dari 120 ulasan).
- **Formulir Input ("Berikan Ulasan Anda"):** Area interaktif yang terdiri dari *input* untuk Rating Bintang, Judul Ulasan, dan *field* Ulasan Utama, dilengkapi tombol Kirim Ulasan dan Batal.
- **Daftar Ulasan ("Ulasan Lainnya"):** Menampilkan *scrollable list* dari semua ulasan pengguna lain, disusun secara kronologis atau berdasarkan relevansi.

-CONTOH DATA YANG DITAMPILKAN

Data yang ditampilkan dalam *mockup* ini dibagi menjadi data buku statis dan data ulasan dinamis:

- **Data Buku:** Judul Buku Contoh, Penulis Contoh, Penerbit Contoh, Genre Fiksi Ilmiah, dan Tahun Terbit 2023.
- **Data Agregasi:** Rating Rata-rata sebesar \$4.5\$ dengan jumlah total 120 ulasan.
- **Data Ulasan:** Setiap entri (misalnya, Pengguna A) menampilkan Rating Bintang (misalnya, 5 bintang), Judul Ulasan ("Buku yang sangat menarik"), Isi Ulasan, dan *timestamp* relatif ("8 hari yang lalu").

-RANCANGAN OPERASI CRUD

Operasi *Create, Read, Update, Delete* (CRUD) berpusat pada entitas **Ulasan** (\$Review\$) dan **Rating**:

- **READ (R):** Membaca detail buku, menghitung rating rata-rata, dan menampilkan daftar semua ulasan yang ada di bagian "Ulasan Lainnya". Ini adalah operasi yang paling sering terjadi.
- **CREATE (C):** Terjadi ketika pengguna mengisi dan mengirimkan formulir di bagian "Berikan Ulasan Anda". Data *rating* dan teks ulasan baru disimpan ke *database* dengan ID buku dan ID pengguna.
- **UPDATE (U):** (Implisit) Pengguna dapat mengubah *rating* atau isi ulasan yang telah mereka kirimkan sebelumnya (meskipun tidak ada tombol "Edit" yang terlihat, ini adalah fungsi standar).
- **DELETE (D):** (Implisit) Pengguna dapat menghapus ulasan yang telah mereka buat.

-PEMANFAATAN API

Aplikasi *frontend* akan berkomunikasi dengan *backend* melalui *Application Programming Interface* (API) untuk mengelola data:

- **API Pengambilan Data Buku/Rating:** Dipanggil menggunakan metode **GET** (contoh: `/api/books/{bookId}`) untuk memuat semua detail dan rating agregat saat halaman dibuka.
- **API Pengambilan Ulasan:** Dipanggil menggunakan **GET** (contoh: `/api/reviews?bookId={bookId}`) untuk memuat daftar ulasan di bagian "Ulasan Lainnya".
- **API Pengiriman Ulasan Baru:** Dipanggil menggunakan metode **POST** (contoh:



`/api/reviews`) ketika tombol **Kirim Ulasan** ditekan, mengirimkan data *rating* dan teks dari formulir.

-KOMPONEN PENTING LAIN

Selain tata letak dan operasi dasar, terdapat elemen non-visual penting yang mendukung fungsionalitas:

- **Sistem Otentikasi/Otorisasi:** Hanya pengguna yang sudah *login* (terlihat "Michael" di header) yang diizinkan untuk melihat, dan yang lebih penting, mengirimkan ulasan baru (operasi **CREATE**).
- **Validasi Form:** Harus ada validasi *client-side* dan *server-side* untuk memastikan *field* wajib (seperti *Rating* dan Isi Ulasan) tidak kosong sebelum data dikirim ke API **POST**.
- **Star Rating Component:** Komponen visual yang berfungsi ganda: menampilkan nilai rata-rata (*output*) dan memungkinkan pengguna memilih nilai *input* saat memberikan ulasan.
- **Penyortiran dan Pemuatan Data Ulasan:** Daftar ulasan biasanya diurutkan berdasarkan tanggal terbaru atau yang paling membantu. Jika jumlah ulasan sangat banyak, diperlukan mekanisme *Paginasi* atau *Infinite Scroll* untuk performa.

E.6. Anggota Kelompok 5 – Menu #6 ...

Bagian ini menggambarkan lima pembahasan utama sebagaimana diuraikan pada Poin D, yang disusun oleh setiap mahasiswa, baik ketua maupun anggota. Uraian dilengkapi dengan ilustrasi mockup tata letak menu yang dikembangkan, contoh data yang ditampilkan, rancangan operasi CRUD, pemanfaatan API, serta komponen lain yang dinilai penting untuk ditampilkan.

Tuliskan uraian pada bagian ini. Kosongkan apabila tidak memiliki anggota sampai 6.



Proposal Tugas Besar

BBK2DAB3 – Pengembangan Aplikasi Web

Sistem Informasi | Fakultas Rekayasa Industri | Telkom University