


Kelas : 2KS2
NIM : 222313332
Nama : Raihan Taufiqurrahman Zaki
Tanggal : 14 Juli 2025
Tanda Tangan : 

LAPORAN PROYEK AKHIR INDIVIDU

“HALOPUS”

BAB I

PENDAHULUAN

A. Latar Belakang

Saat ini perkembangan teknologi informasi berkembang sangat cepat dan mulai banyak dimanfaatkan di berbagai bidang, termasuk dalam pengelolaan perpustakaan. Sistem perpustakaan yang masih menggunakan cara manual dinilai kurang efektif dan tidak sesuai lagi dengan kebutuhan zaman sekarang. Beberapa masalah yang sering terjadi dalam pengelolaan manual di antaranya sulitnya mencari data, risiko dokumen fisik rusak atau hilang, proses administrasi yang lama, dan sulitnya memantau aktivitas perpustakaan.

Karena itu, dibutuhkan sebuah sistem informasi yang bisa membantu proses pengelolaan perpustakaan menjadi lebih cepat, aman, dan teratur. Salah satu solusi yang tepat adalah menggunakan sistem berbasis web, karena bisa diakses dari mana saja, memudahkan penyimpanan dan pencarian data, serta tampilannya bisa dibuat lebih mudah digunakan oleh admin maupun anggota perpustakaan.

B. Tujuan Pengembangan

Berdasarkan latar belakang tersebut, dikembangkanlah Halopus sebagai solusi sistem informasi perpustakaan berbasis web yang bertujuan untuk:

1. Mengotomatisasi proses pengelolaan data perpustakaan

2. Meningkatkan efisiensi dan efektivitas layanan perpustakaan
3. Menyediakan akses yang mudah dan cepat bagi pengguna
4. Memfasilitasi monitoring dan pelaporan aktivitas perpustakaan secara real-time
5. Mengurangi risiko human error dalam pengelolaan data

C. Deskripsi Web

Halopus merupakan sistem informasi perpustakaan berbasis web yang dirancang khusus untuk membantu pengelolaan data buku, data anggota, serta proses peminjaman dan pengembalian buku secara terintegrasi. Aplikasi ini ditujukan untuk digunakan oleh petugas atau admin perpustakaan serta para anggota perpustakaan di lingkungan institusi pendidikan, organisasi, atau komunitas yang memiliki kebutuhan pengelolaan koleksi buku secara sistematis dan terstruktur.

1. Pengguna (User)

Halopus mengimplementasikan sistem multi-level user dengan tiga jenis pengguna yang memiliki hak akses berbeda:

a. Super Admin

- 1) Memiliki hak akses penuh terhadap seluruh fitur sistem
- 2) Dapat mengelola akun admin lain
- 3) Memiliki akses ke semua data dan laporan sistem

b. Admin Biasa (Admin dan Staff)

- 1) Memiliki akses terhadap fitur-fitur pengelolaan perpustakaan
- 2) Dapat mengelola data buku, anggota, dan transaksi
- 3) Tidak memiliki akses untuk mengelola akun admin lain

c. Anggota

- 1) Dapat meminjam dan mengembalikan buku (harus melalui admin jika ingin mengembalikan buku)
- 2) Dapat memantau riwayat peminjaman secara mandiri
- 3) Memiliki akses terbatas sesuai dengan kebutuhan pengguna layanan

2. Data

Data yang dikelola dalam Halopus mencakup:

- a. Data Katalog Buku : judul, penulis, penerbit, tahun terbit, kategori, ISBN, jumlah eksemplar, lokasi rak
- b. Data Kategori Buku : klasifikasi buku berdasarkan subjek atau genre
- c. Data Anggota : nama lengkap, email, nomor telepon, alamat, kode anggota, tipe anggota, status keanggotaan, tanggal bergabung
- d. Data Admin : informasi akun admin dan hak akses
- e. Data Transaksi : riwayat peminjaman dan pengembalian buku, termasuk tanggal, status, dan denda

3. Fitur

Halopus menyediakan berbagai fitur yang mendukung operasional perpustakaan modern:

a. Fitur Umum:

- 1) Sistem autentikasi
- 2) Dashboard dengan ringkasan data dan statistic
- 3) Search (termasuk livenessearch) dan filter data untuk pencarian yang efisien
- 4) Responsive design yang dapat diakses dari berbagai perangkat

b. Fitur untuk Admin:

- 1) Manajemen data buku (CRUD)
- 2) Manajemen data anggota dan keanggotaan (CRUD)
- 3) Proses peminjaman dan pengembalian buku
- 4) Manajemen kategori buku
- 5) Laporan dan riwayat transaksi
- 6) Monitoring aktivitas sistem

c. Fitur Khusus Super Admin:

- 1) Manajemen akun admin lain

d. Fitur untuk Anggota:

- 1) Pencarian katalog buku
- 2) Riwayat peminjaman pribadi
- 3) Informasi keanggotaan

D. Deskripsi Web

Untuk keperluan pengujian dan demonstrasi Halopus, tersedia akun-akun berikut:

Jenis Akun	Username/email	Password	Keterangan
Super Admin	admin	admin123	Akses penuh sistem
Admin Biasa	new_staff	staff123	Akses terbatas admin
Anggota	k.yoojung@stis.ac.id	tes123456	Akses anggota perpustakaan

E. Manfaat Web

Dengan implementasi Halopus, diharapkan dapat memberikan manfaat sebagai berikut:

1. Peningkatan Efisiensi: Optimasi proses administrasi perpustakaan, mengurangi waktu dan tenaga yang diperlukan
2. Akurasi Data: Mengurangi risiko human error dalam pengelolaan data
3. Kemudahan Akses: Interface yang user-friendly memudahkan pengguna dalam mengakses layanan perpustakaan
4. Monitoring Real-time: Kemampuan untuk memantau aktivitas perpustakaan secara real-time
5. Skalabilitas: Sistem yang dapat berkembang sesuai dengan kebutuhan institusi
6. Keamanan Data: Implementasi sistem keamanan yang melindungi data perpustakaan

7. Layanan Mandiri: Anggota dapat mengakses informasi dan layanan secara mandiri tanpa bantuan petugas

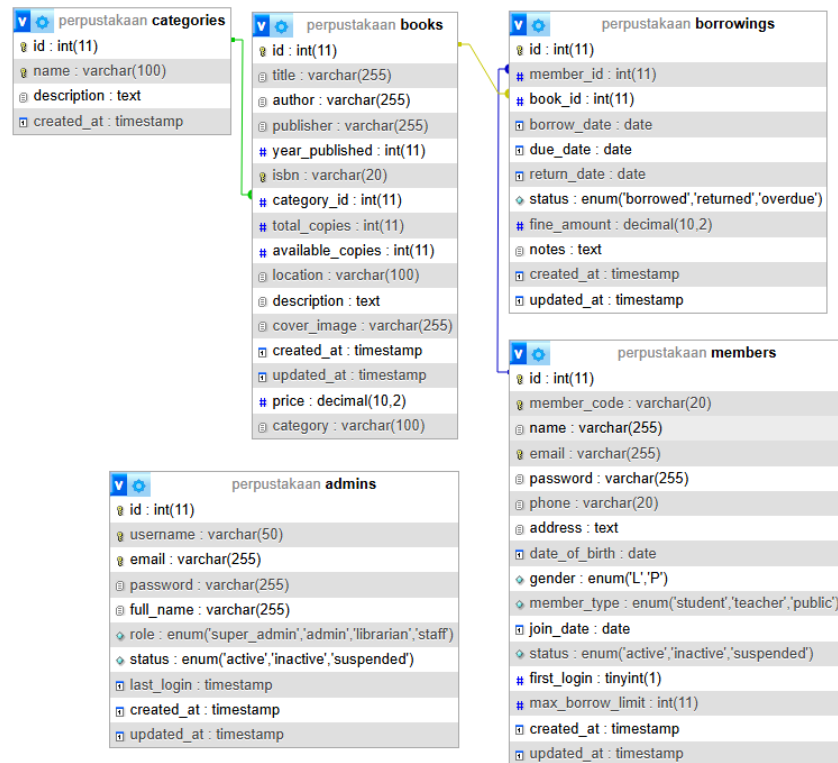
Melalui pengembangan Halopus, diharapkan dapat terwujud sistem pengelolaan perpustakaan yang modern, efisien, dan sesuai dengan kebutuhan era digital saat ini.

BAB II

ISI

A. Model Data (Skema Database)

Sistem informasi perpustakaan Halopus menggunakan database MySQL dengan 5 tabel utama yang saling terhubung. Skema database dapat dilihat pada gambar berikut:



Gambar 2.1 Skema Database Halopus

Penjelasan Tabel:

1. categories

Menyimpan data kategori buku yang ada di perpustakaan.

Kolom:

- id: primary key
- name: nama kategori buku
- description: deskripsi kategori
- created_at: waktu pembuatan data

2. books

Menyimpan data detail buku di perpustakaan.

Kolom:

- a. id: primary key
- b. title: judul buku
- c. author: nama penulis
- d. publisher: nama penerbit
- e. year_published: tahun terbit
- f. isbn: kode ISBN
- g. category_id: relasi ke tabel categories
- h. total_copies: jumlah total eksemplar
- i. available_copies: jumlah eksemplar yang masih tersedia
- j. location: lokasi buku
- k. description: deskripsi buku
- l. cover_image: gambar cover buku
- m. created_at, updated_at: waktu buat dan ubah
- n. price: harga buku
- o. category: nama kategori buku (opsional)

3. members

Menyimpan data anggota perpustakaan.

Kolom:

- a. id: primary key
- b. member_code: kode anggota
- c. name, email, password, phone, address: data diri anggota
- d. date_of_birth: tanggal lahir
- e. gender: jenis kelamin (L/P)
- f. member_type: tipe anggota (student, teacher, public)
- g. join_date: tanggal bergabung
- h. status: status akun (active, inactive, suspended)
- i. first_login: info login pertama kali
- j. max_borrow_limit: maksimal buku yang bisa dipinjam
- k. created_at, updated_at: waktu buat dan ubah

4. admins

Menyimpan data akun admin.

Kolom:

- a. id: primary key
- b. username, email, password, full_name: data admin
- c. role: jenis admin (super_admin, admin, librarian, staff)
- d. status: status akun (active, inactive, suspended)
- e. last_login: waktu login terakhir
- f. created_at, updated_at: waktu buat dan ubah

5. borrowings

Menyimpan data transaksi peminjaman buku.

Kolom:

- a. id: primary key
- b. member_id: relasi ke members
- c. book_id: relasi ke books
- d. borrow_date: tanggal pinjam
- e. due_date: batas pengembalian
- f. return_date: tanggal buku dikembalikan
- g. status: status transaksi (borrowed, returned, overdue)
- h. fine_amount: nominal denda jika ada
- i. notes: catatan tambahan
- j. created_at, updated_at: waktu buat dan ubah

Dalam database Halopus, terdapat beberapa jenis relasi antar tabel. Berikut penjelasan relasinya:

1. categories → books

a. Jenis Relasi: One to Many

b. Penjelasan:

Satu kategori bisa dimiliki oleh banyak buku, tetapi satu buku hanya bisa memiliki satu kategori.

c. Relasi:

categories.id → books.category_id

2. members → borrowings

a. Jenis Relasi: One to Many

b. Penjelasan:

Satu anggota dapat melakukan banyak transaksi peminjaman, tetapi satu transaksi peminjaman hanya dimiliki oleh satu anggota.

c. Relasi:

members.id → borrowings.member_id

3. books → borrowings

a. Jenis Relasi: One to Many

b. Penjelasan:

Satu buku bisa dipinjam oleh banyak anggota pada waktu berbeda (dalam banyak transaksi peminjaman), tapi satu transaksi peminjaman hanya terkait dengan satu buku.

c. Relasi:

books.id → borrowings.book_id

4. Admins

Tidak memiliki relasi ke tabel lain secara langsung, karena hanya bertugas mengelola data dan aktivitas melalui sistem

B. Validasi

1. Java Script

a. Main.js (0 Error dan 0 Warning)

```

1  /*jshint esversion: 11 */
2
3  // Fungsionalitas sidebar
4  document.addEventListener('DOMContentLoaded', () => {
5      // Ambil elemen-elemen terkait sidebar
6      const hamburgerMenu = document.getElementById("hamburgerMenu");
7      const sidebar = document.getElementById("sidebar");
8      const sidebarOverlay = document.getElementById("sidebarOverlay");
9      const sidebarClose = document.getElementById("sidebarClose");
10
11      // Fungsi untuk membuka dan menutup sidebar
12      function toggleSidebar() {
13          sidebar.classList.toggle("active");
14          sidebarOverlay.classList.toggle("active");
15          hamburgerMenu.classList.toggle("active");
16          document.body.style.overflow = sidebar.classList.contains("active") ? "hidden" : ""
17      }
18
19      // Fungsi untuk menutup sidebar
20      function closeSidebar() {
21          sidebar.classList.remove("active");
22          sidebarOverlay.classList.remove("active");
23          hamburgerMenu.classList.remove("active");
24          document.body.style.overflow = "";
25      }
26
27      // Event klik pada tombol hamburger untuk toggle sidebar
28      if (hamburgerMenu) {
29          hamburgerMenu.addEventListener("click", toggleSidebar);
30      }
31
32      // Event klik pada tombol close sidebar
33      if (sidebarClose) {
34          sidebarClose.addEventListener("click", closeSidebar);
35      }
36
37      // Event klik di area overlay untuk menutup sidebar
38      if (sidebarOverlay) {
39          sidebarOverlay.addEventListener("click", closeSidebar);
40      }
41
42      // Event tekan tombol Escape untuk menutup sidebar
43      document.addEventListener("keydown", (e) => {
44          if (e.key === "Escape" && sidebar.classList.contains("active")) {
45              closeSidebar();
46          }
47      });
48
49      // Auto-hide alert setelah 5 detik
50      setTimeout(() => {
51          const alerts = document.querySelectorAll(".alert");
52          alerts.forEach(alert => {
53              alert.style.opacity = "0";
54              alert.style.transform = "translateY(-20px)";
55              setTimeout(() => alert.remove(), 300);
56          });
57      }, 5000);
58  });

```

CONFIGURE

Metrics

There are 7 functions in this file.

Function with the largest signature take 1 arguments, while the median is 0.

Largest function has 14 statements in it, while the median is 3.

The most complex function has a cyclomatic complexity value of 4 while the median is 1.

b. Animasi.js (0 Error dan 0 Warning)

```

1  /*jshint esversion: 11 */
2  document.addEventListener('DOMContentLoaded', function () {
3      // Ambil elemen menu sidebar dan item-item di dalamnya
4      const menuItems = document.querySelectorAll('.sidebar-menu li a');
5      const sidebarMenu = document.getElementById('sidebarMenu');
6
7      // Efek hover dock pada menu sidebar
8      sidebarMenu.addEventListener('mouseover', function (e) {
9          const targetItem = e.target.closest('li');
10         if (!targetItem) return;
11
12         const allItems = Array.from(sidebarMenu.querySelectorAll('li'));
13         const targetIndex = allItems.indexOf(targetItem);
14
15         allItems.forEach((item, index) => {
16             const distance = Math.abs(index - targetIndex);
17             const link = item.querySelector('a');
18
19             if (distance === 0) {
20                 link.style.transform = 'translateX(15px) scale(1.08)';
21                 link.style.zIndex = '10';
22             } else if (distance === 1) {
23                 link.style.transform = 'translateX(8px) scale(1.04)';
24                 link.style.zIndex = '5';
25             } else if (distance === 2) {
26                 link.style.transform = 'translateX(4px) scale(1.02)';
27                 link.style.zIndex = '2';
28             } else {
29                 link.style.transform = 'translateX(0) scale(1)';
30                 link.style.zIndex = '1';
31             }
32         });
33     });
34
35     // Reset efek dock saat kursor keluar dari sidebar
36     sidebarMenu.addEventListener('mouseleave', function () {
37         const allLinks = sidebarMenu.querySelectorAll('li a');
38         allLinks.forEach(link => {
39             link.style.transform = 'translateX(0) scale(1)';
40             link.style.zIndex = '1';
41         });
42     });
43
44     // Efek klik menu (tambah class active dan animasi klik)
45     menuItems.forEach(item => {
46         item.addEventListener('click', function (e) {
47             menuItems.forEach(i => i.classList.remove('active'));
48             this.classList.add('active');
49             this.style.transform = 'scale(0.95)';
50             setTimeout(() => {
51                 this.style.transform = '';
52             }, 150);
53         });
54     });
55
56     // Simulasi pengecekan role user
57     const hasRole = (role) => {
58         return role === 'super_admin';
59     };
60
61     // Menyembunyikan menu admin jika bukan super_admin
62     const adminMenu = document.querySelector('.admin-only');
63     if (!hasRole('super_admin')) {

```

CONFIGURE

Metrics

There are 10 functions in this file.

Function with the largest signature take 2 arguments, while the median is 1.

Largest function has 11 statements in it, while the median is 2.

The most complex function has a cyclomatic complexity value of 4 while the median is 1.

c. Pencarian.js (0 Error dan 0 Warning)

<pre>1 /*jshint esversion: 11 */ 2 // Kelas untuk fitur pencarian buku 3 class BookSearch { 4 constructor() { 5 // Ambil elemen input, form, dan container hasil pencarian 6 this.searchInput = document.getElementById("searchInput"); 7 this.searchResults = document.getElementById("searchResults"); 8 this.searchForm = document.getElementById("searchForm"); 9 this.init(); 10 } 11 12 // Inisialisasi event listener 13 init() { 14 // Event submit form pencarian 15 if (this.searchForm) { 16 this.searchForm.addEventListener("submit", (e) => { 17 e.preventDefault(); 18 this.performSearch(); 19 }); 20 } 21 22 // Event input realtime saat user mengetik 23 if (this.searchInput) { 24 let timeout; 25 this.searchInput.addEventListener("input", (e) => { 26 clearTimeout(timeout); 27 timeout = setTimeout(() => { 28 if (e.target.value.length >= 3) { 29 this.performLiveSearch(e.target.value); 30 } else { 31 this.clearResults(); 32 } 33 }, 300); 34 }); 35 } 36 } 37 38 // Pencarian buku via AJAX saat submit form 39 async performSearch() { 40 const query = this.searchInput?.value.trim(); 41 if (!query) return; 42 43 try { 44 const response = await fetch("pages/cari_buku.php?search=\${encodeURIComponent(45 query 46)}"); 47 const data = await response.json(); 48 this.displayResults(data); 49 } catch (error) { 50 console.error("Search error:", error); 51 } 52 53 // Pencarian live saat user mengetik (minimal 3 karakter) 54 async performLiveSearch(query) { 55 try { 56 const response = await fetch("pages/cari_buku.php?search=\${encodeURIComponent(57 query 58)}"); 59 const data = await response.json(); 60 this.displayLiveResults(data); 61 } catch (error) { 62 console.error("Live search error:", error); 63 } 64 } 65 66 // Tampilkan hasil pencarian utama di halaman 67 } 68 }</pre>	<p>CONFIGURE</p> <p>Metrics</p> <p>There are 16 functions in this file.</p> <p>Function with the largest signature take 1 arguments, while the median is 1.</p> <p>Largest function has 13 statements in it, while the median is 2.5.</p> <p>The most complex function has a cyclomatic complexity value of 3 while the median is 1.</p> <p>Two unused variables</p> <p>196 bookSearch</p> <p>200 selectBook</p>
---	--

d. Validasi.js (0 Error dan 0 Warning)

<pre>1 /*jshint esversion: 11 */ 2 // Fungsi validasi input wajib diisi 3 function validateRequired(value, fieldName) { 4 if (!value value.trim() === "") { 5 return `\${fieldName} harus diisi`; 6 } 7 return null; 8 } 9 10 // Fungsi validasi format email 11 function validateEmail(email) { 12 const emailRegex = /^[^\s@]+@[^\s@]+\.[^\s@]+\$/; 13 if (!emailRegex.test(email)) { 14 return "Format email tidak valid"; 15 } 16 return null; 17 } 18 19 // Fungsi validasi angka beserta batas minimal dan maksimal 20 function validateNumber(value, fieldName, min = null, max = null) { 21 if (isNaN(value) value === "") { 22 return `\${fieldName} harus berupa angka`; 23 } 24 25 const num = Number.parseFloat(value); 26 if (min !== null && num < min) { 27 return `\${fieldName} minimal \${min}`; 28 } 29 if (max !== null && num > max) { 30 return `\${fieldName} maksimal \${max}`; 31 } 32 return null; 33 } 34 35 // Fungsi validasi tahun terbit 36 function validateYear(year) { 37 const currentYear = new Date().getFullYear(); 38 const yearNum = Number.parseInt(year); 39 if (yearNum < 1900 yearNum > currentYear) { 40 return "Tahun harus antara 1900 dan \${currentYear}"; 41 } 42 return null; 43 } 44 45 // Setup validasi realtime saat input blur atau diketik 46 function setupRealtimeValidation() { 47 const inputs = document.querySelectorAll(".form-control"); 48 49 inputs.forEach((input) => { 50 input.addEventListener("blur", function () { 51 validateField(this); 52 }); 53 54 input.addEventListener("input", function () { 55 clearFieldError(this); 56 }); 57 }); 58 } 59 60 // Fungsi validasi per field saat event terjadi 61 function validateField(field) { 62 const value = field.value.trim(); 63 const fieldName = field.getAttribute("data-field-name") field.name;</pre>	<p>CONFIGURE</p> <p>Metrics</p> <p>There are 13 functions in this file.</p> <p>Function with the largest signature take 4 arguments, while the median is 1.</p> <p>Largest function has 19 statements in it, while the median is 4.</p> <p>The most complex function has a cyclomatic complexity value of 11 while the median is 2.</p>
---	---

C. Screenshot Fitur


Berikut adalah fitur yang tersedia dalam Halopus (umum):

1. Halaman Login

a. Deskripsi:

Fitur untuk masuk ke dalam sistem menggunakan akun admin atau anggota.

b. Screenshot:



The screenshot displays the admin login interface for 'Perpustakaan Digital'. The page has a dark purple background. At the top, the title 'Perpustakaan Digital' is in yellow, followed by the instruction 'Silakan login untuk melanjutkan' in white. Below this, there are two tabs: 'Admin' (highlighted in red) and 'Anggota'. The login form includes a 'Username' label in red above a dark blue input field, and a 'Password' label in red above another dark blue input field. A prominent pink button labeled 'Login sebagai Admin' is positioned below the password field. At the bottom, a thin horizontal line separates the footer, which contains the copyright notice '© 2025 Perpustakaan Digital'.

Gambar 2.2 Halaman Login Admin

Perpustakaan Digital

Silakan login untuk melanjutkan

Admin

Anggota

Email

Password

Login sebagai Anggota

Belum memiliki akun? [Daftar di sini](#)

© 2025 Perpustakaan Digital

Gambar 2.3 Halaman Login Anggota

Daftar Anggota Baru

Silakan isi formulir di bawah untuk mendaftar

Nama Lengkap *

Kim yoo-Jung

Email *

k.yoojung@stis.ac.id

Password * **Konfirmasi Password ***

.....

Nomor Telepon *

08111111111

Tanggal Lahir *

22/09/1999

Alamat

Distrik Seongdong, Seoul, Korea Selatan

Jenis Kelamin *

☒ Laki-laki ☐ Perempuan

Daftar Sekarang

Sudah memiliki akun? [Login di sini](#)

Gambar 2.4 Halaman Daftar Anggota

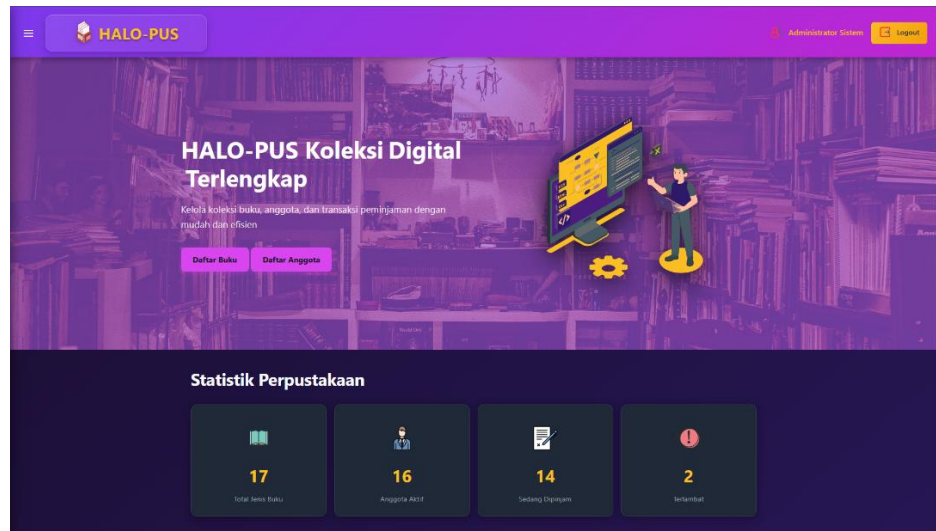
Berikut adalah fitur yang tersedia dalam Halopus (admin):

1. Dashboard

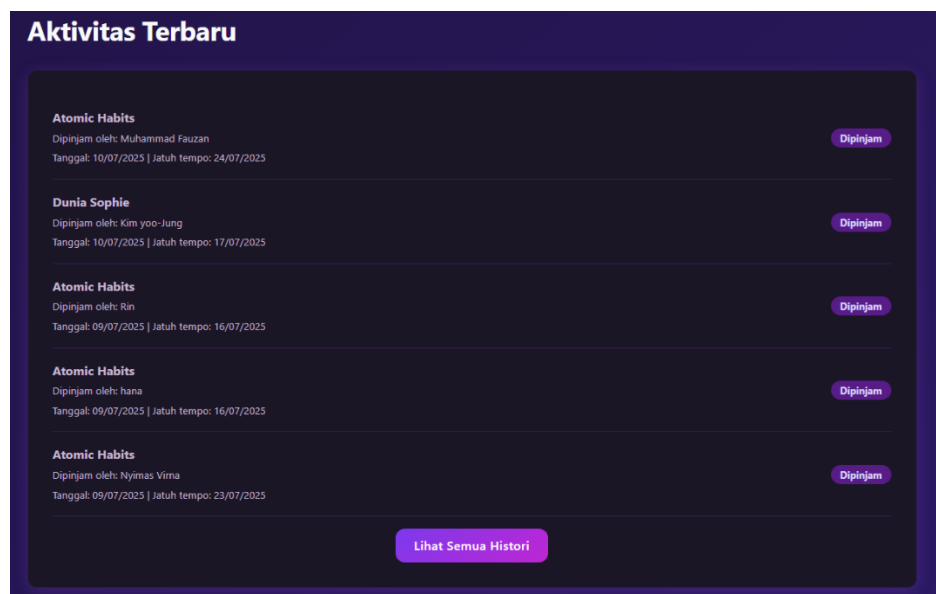
a. Deskripsi:

Menampilkan ringkasan data penting seperti total anggota, buku, peminjaman aktif, peminjaman terlambat, dan aktivitas terbaru serta beberapa aksi cepat.

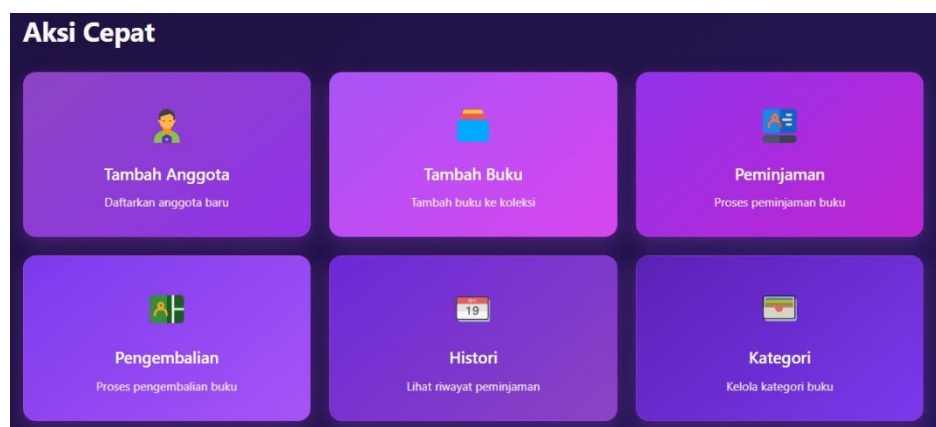
b. Screenshot:



Gambar 2.5 Halaman Dashboard



Gambar 2.6 Dashboard Aktivitas Terbaru



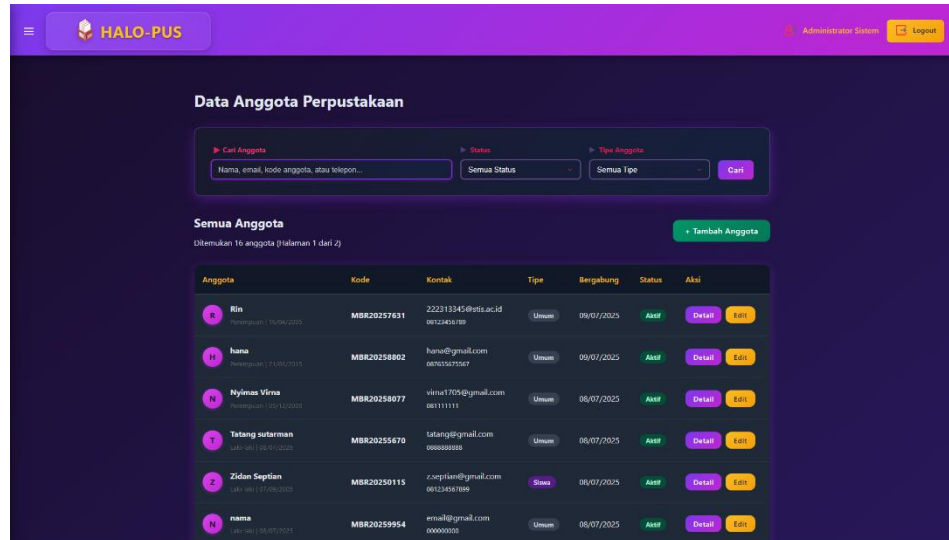
Gambar 2.7 Dashboard Aksi Cepat

2. Data Anggota

a. Deskripsi:

Fitur untuk melihat, menambah, dan mengedit data anggota.

b. Screenshot:



Gambar 2.8 Halaman Data Anggota

Tambah Anggota Baru

Nama Lengkap * **Email ***

Nomor Telepon * **Tanggal Lahir ***

Jenis Kelamin * **Tipe Anggota ***

Alamat *


Pengaturan Password *

☒ Password Otomatis
Gunakan kode anggota sebagai password default

☐ Password Kustom
Tentukan password sendiri

Daftarkan Anggota **Kembali ke Data Anggota** **Dashboard**

Gambar 2.9 Halaman Tambah Anggota Baru



Rin

MBR20257631

AKTIF

Email:
222313345@stis.ac.id

Telepon:
08123456789

Jenis Kelamin:
Perempuan

Tanggal Lahir:
16/04/2005

Tipe Anggota:
Umum

Bergabung:
09/07/2025

Alamat:
Jalan Otista 3

Edit Pinjam Buku Kembali

1

TOTAL PEMINJAMAN

1

SEDANG DIPINJAM

0

SUDAH DIKEMBALIKAN

Rp 0

TOTAL DENDA

Buku yang Sedang Dipinjam

Atomic Habits

oleh James Clear

Dipinjam: 09/07/2025

Jatuh tempo: 16/07/2025

DIPINJAM

Histori Peminjaman Terbaru

Atomic Habits

oleh James Clear

09/07/2025

DIPINJAM

Lihat Semua Histori

Gambar 2.10 Halaman Detail Data Anggota

Edit Data Anggota

► Nama Lengkap *

Kim yoo-Jung

► Email *

k.yoojung@stis.ac.id

► Nomor Telepon *

081111111111

► Tanggal Lahir *

22/09/1999

► Jenis Kelamin *

☐ Laki-laki
 ☒ Perempuan

► Tipe Anggota *

Umum (Maks 2 buku)

► Status *

Aktif

► Kode Anggota

MBR20250169

Kode anggota tidak dapat diubah

► Alamat *

Distrik Seongdong, Seoul, Korea Selatan

Perbarui Data Anggota

Lihat Detail

Kembali ke Data Anggota

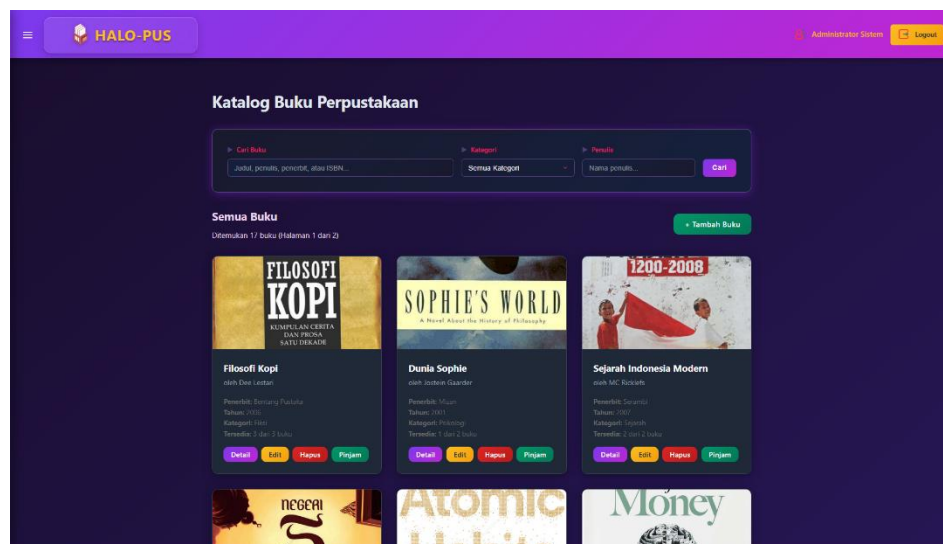
Gambar 2.11 Halaman Edit Data Anggota

3. Katalog Buku

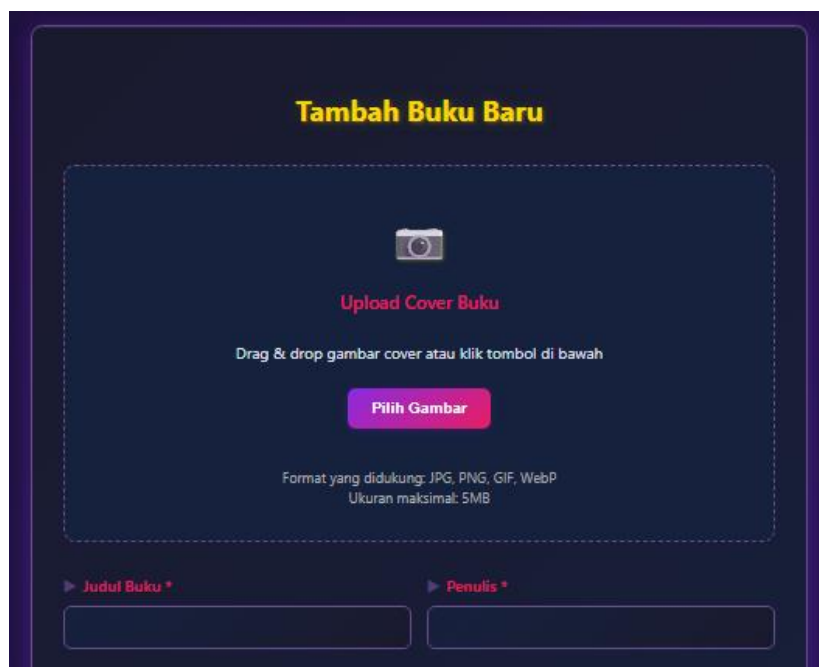
a. Deskripsi:

Menampilkan daftar buku perpustakaan dengan fitur pencarian, filter, tambah, edit, dan hapus buku.

b. Screenshot:



Gambar 2.12 Halaman Katalog Buku



Penerbit *

Tahun Terbit *

2025

ISBN

978-xxxx-xxxx-xxxx

Kategori *

Pilih Kategori

Jumlah Buku *

1

Lokasi Rak

Contoh: Rak A1

Jumlah total buku yang akan ditambahkan

Lokasi penyimpanan buku di perpustakaan

Deskripsi

Masukkan deskripsi singkat tentang buku...

Tambah Buku

Kembali ke Katalog

Dashboard

Gambar 2.13 Halaman Tambah Buku Baru

Judul:

Laskar Pelangi

Penulis:

Andrea Hirata

Penerbit:

Bentang Pustaka

Tahun Terbit:

2005

ISBN:

9789793062792

Kategori:

Fiksi

Lokasi:

Rak A1

Ketersediaan:

1 dari 3 buku

Deskripsi:

Novel tentang perjuangan anak-anak Belitung untuk mendapatkan pendidikan

Edit

Hapus

Pinjam

Kembali

4

Total Peminjaman

2

Sedang Dipinjam

2

Kali Dikembalikan

Sedang Dipinjam Oleh

Diah Ayu Nur Rahmadani

TERLAMBAT

MBR20257450

Dipinjam: 03/07/2025

Jatuh tempo: 10/07/2025

Muhammad Fauzan

TERLAMBAT

MBR20253038

Dipinjam: 27/06/2025

Jatuh tempo: 11/07/2025

Histori Peminjaman Terbaru

Diah Ayu Nur Rahmadani

TERLAMBAT

MBR20257450

03/07/2025

Muhammad Fauzan

TERLAMBAT

MBR20253038

27/06/2025

Kim yoo-Jung


DIKEMBALIKAN

MBR20250169


27/06/2025

Gambar 2.14 Halaman Detail Buku

Edit Buku



Cover Buku



Cover saat ini

Drag & drop gambar cover baru atau klik tombol di bawah

Pilih Gambar Baru

Format yang didukung: JPG, PNG, GIF, WebP
Ukuran maksimal: 5MB
Kosongkan jika tidak ingin mengubah cover

Judul Buku *

Laskar Pelangi

Penulis *

Andrea Hirata

Penerbit *

Bentang Pustaka

Tahun Terbit *

2005

ISBN

9789793062792

Kategori *

Fiksi

Jumlah Buku *

3

Lokasi Rak

Rak A1

Deskripsi

Novel tentang perjuangan anak-anak Belitung untuk mendapatkan pendidikan

Perbarui Buku

Lihat Detail

Kembali ke Katalog

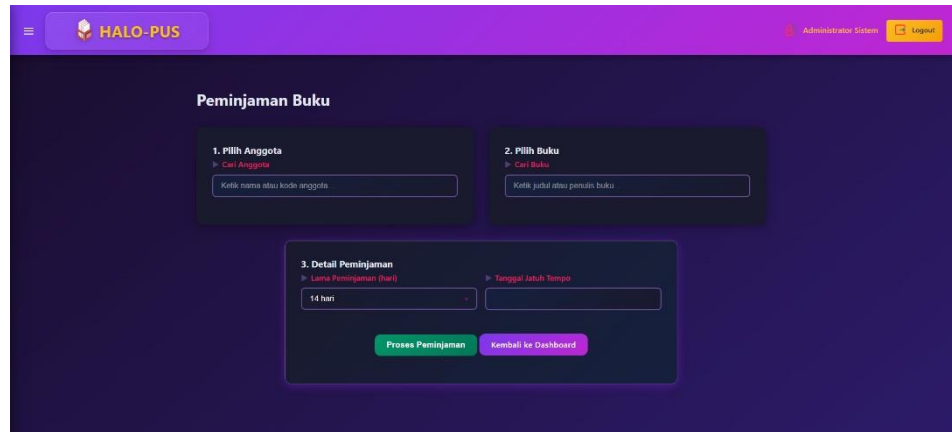
Gambar 2.15 Halaman Edit Buku

4. Peminjaman

a. Deskripsi:

Admin bisa melakukan peminjaman buku atas nama anggota, dengan menentukan tanggal pengembalian.

b. Screenshot:



The screenshot shows a web application interface for book borrowing. At the top, there is a purple header with the logo 'HALO-PUS' and a 'Logout' button. Below the header, the main content area is titled 'Peminjaman Buku'. It contains three sections: 1. 'Pilih Anggota' with a dropdown menu and a text input field for the member's name or ID. 2. 'Pilih Buku' with a dropdown menu and a text input field for the book title or ID. 3. 'Detail Peminjaman' with a dropdown menu for the loan type (currently set to '14 hari') and a date input field for the return date. At the bottom of the form, there are two buttons: 'Proses Peminjaman' (green) and 'Kembali ke Dashboard' (purple).

Gambar 2.16 Halaman Peminjaman Buku



The screenshot shows a live search results page for books. The title is '2. Pilih Buku' with a sub-header 'Cari Buku'. A search input field contains the text 'sa'. Below the input field, there is a list of book results. The first result is 'Belajar Python untuk Pemula' by Muhammad Ihsan, with 2 books available, categorized under 'Sains & Teknologi' and located in 'Rak C2'. The second result is 'Kimia Dasar' by Hardjono Sastrohamidjojo, with 4 books available, categorized under 'Sains & Teknologi' and located in 'Rak C1'. The third result is 'Rich Dad Poor Dad' by Robert Kiyosaki, with 2 books available. A vertical scrollbar is visible on the right side of the results list.

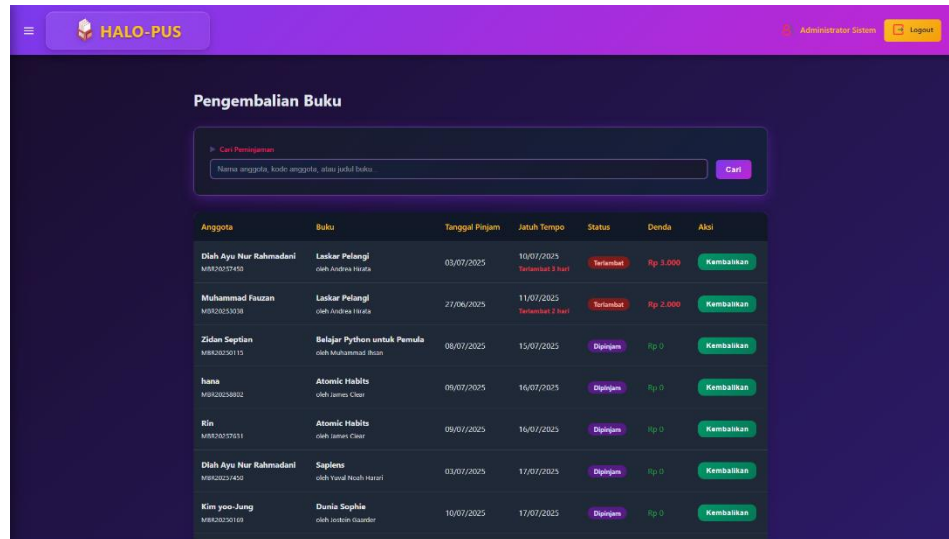
Gambar 2.17 Livesearch

5. Pengembalian

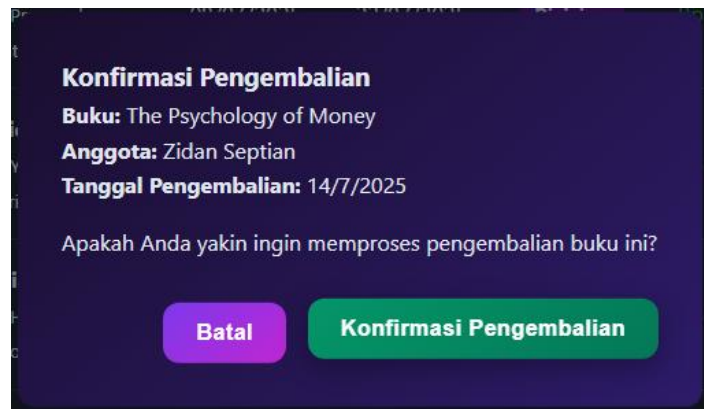
a. Deskripsi:

Admin melakukan konfirmasi pengembalian buku dan perhitungan denda jika lewat batas waktu.

b. Screenshot:



Gambar 2.18 Halaman Pengembalian Buku



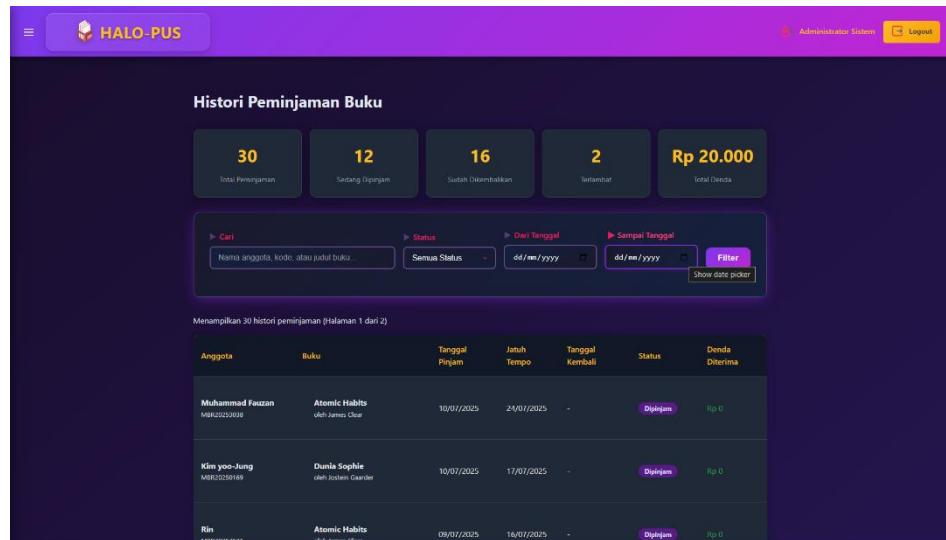
Gambar 2.19 Konfirmasi Pengembalian

6. Histori

a. Deskripsi:

Admin dapat melihat aktivitas peminjaman buku.

b. Screenshot:



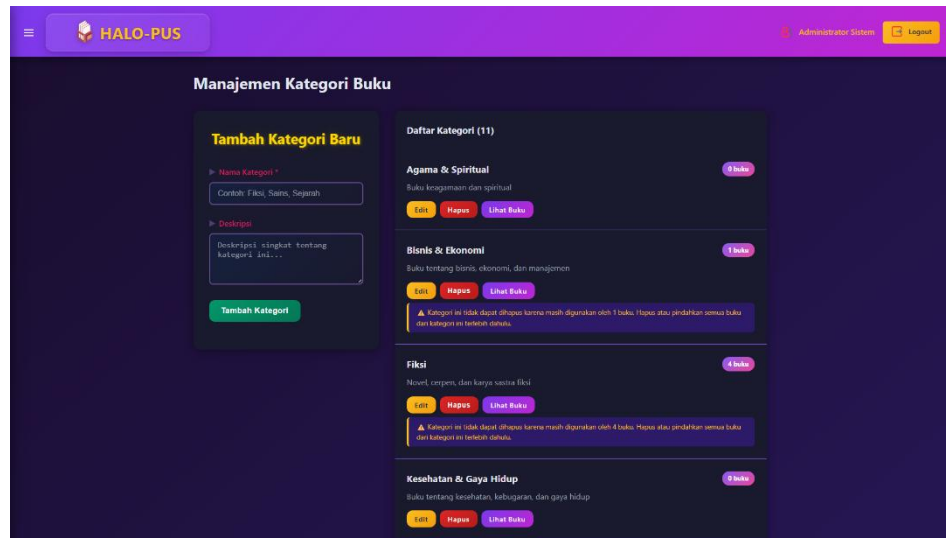
Gambar 2.20 Halaman Histori

7. Kategori

a. Deskripsi:

Admin dapat menambah, edit, dan hapus kategori buku.

b. Screenshot:



Gambar 2.21 Halaman Kategori Buku



Edit Kategori

► **Nama Kategori ***

Agama & Spiritual

► **Deskripsi**

Buku keagamaan dan spiritual

Perbarui Kategori

Batal

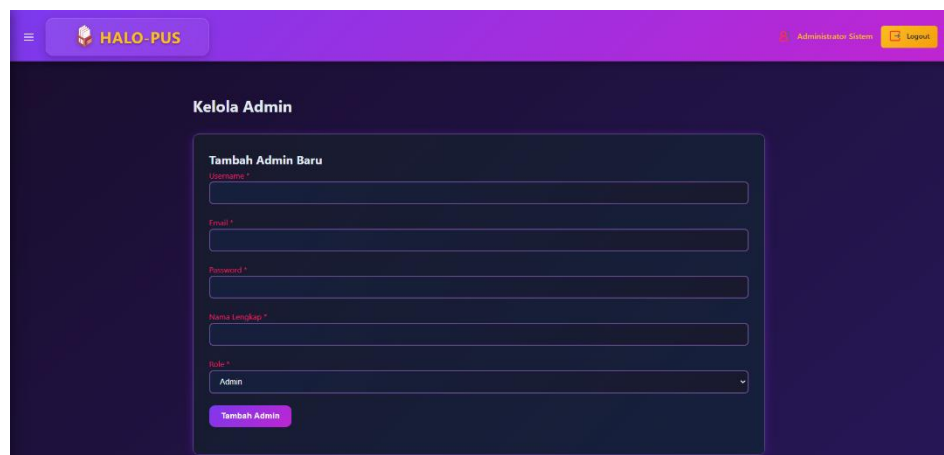
Gambar 2.22 Edit Kategori

8. Kelola Admin (Super Admin)

a. Deskripsi:

Fitur khusus super admin untuk menambah, aktif/nonaktifkan, atau suspend akun admin lain.

b. Screenshot:



HALO-PUS Administrator Sistem Logout

Kelola Admin

Tambah Admin Baru

Username

Email *

Password *

Nama Lengkap *

Role *

Admin

Tambah Admin

Gambar 2.23 Halaman Kelola Admin

Daftar Admin						
Username	Nama Lengkap	Email	Role	Status	Last Login	Aksi
old_staff	Staff Perpustakaan	staff1@perpustakaan.com	Staff	Tidak Aktif	Belum pernah	Tidak Aktif
inspector	Inspektur	inspector@perpustakaan.com	Admin	Tidak Aktif	Belum pernah	Tidak Aktif
client	Klien Reguler	client@perpustakaan.com	Pustakawan	Aktif	Belum pernah	Aktif
new_staff	Staff Perpustakaan	staff2@perpustakaan.com	Staff	Aktif	13/07/2025 14:34	Aktif
tester	Penguji	tester@perpustakaan.com	Admin	Tidak Aktif	07/07/2025 21:01	Tidak Aktif
saya	Peretas Handal	saya@gmail.com	Admin	Suspended	27/06/2025 05:46	Suspende
guest	Tamu Perpustakaan	guest@perpustakaan.com	Staff	Tidak Aktif	07/07/2025 21:29	Tidak Aktif
admin	Administrator Sistem	admin@perpustakaan.com	Super	Aktif	13/07/2025 15:27	Current User
librarian	Pustakawan Utama	librarian@perpustakaan.com	Pustakawan	Aktif	07/07/2025 23:34	Aktif
staff	Staff Perpustakaan	staff@perpustakaan.com	Staff	Tidak Aktif	Belum pernah	Tidak Aktif

Gambar 2.24 Daftar Admin

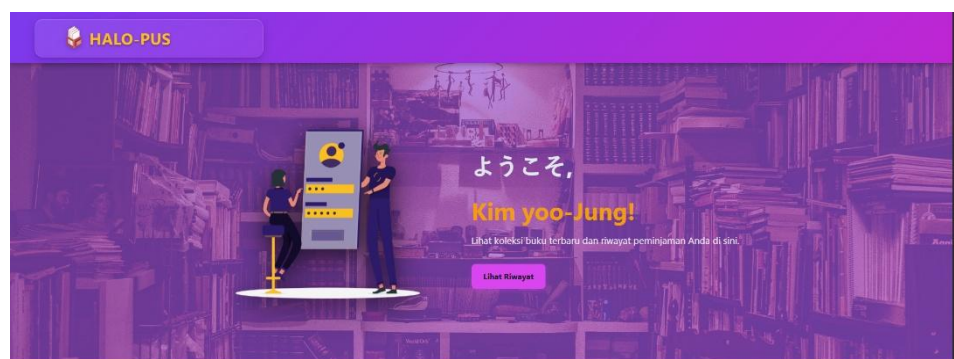
Berikut adalah fitur yang tersedia dalam Halopus (anggota):

1. Dashboard

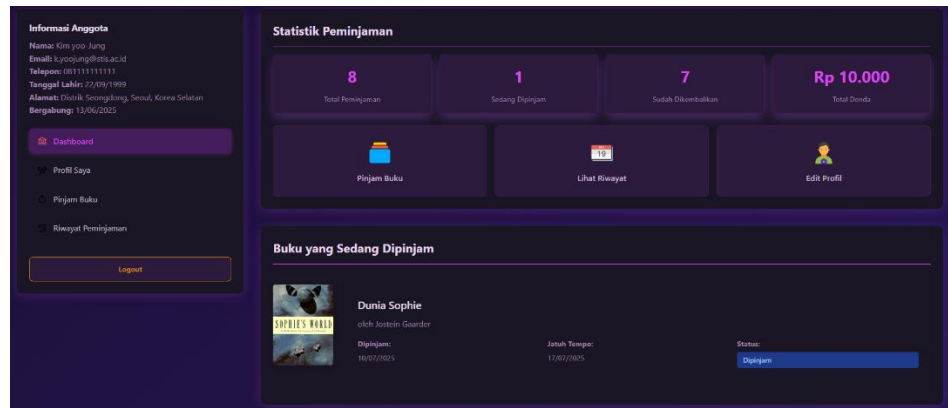
a. Deskripsi:

Menampilkan ringkasan akun anggota, statistik peminjaman, dan riwayat peminjaman.

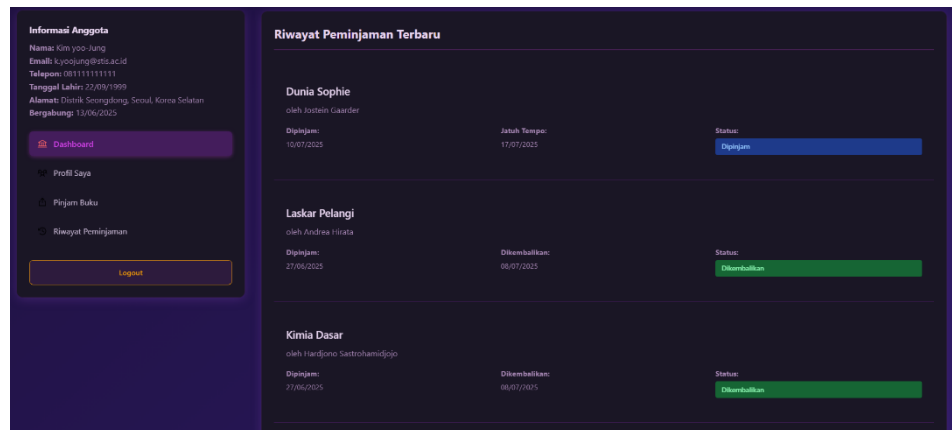
b. Screenshot:



Gambar 2.25 Halaman Dashboard



Gambar 2.26 Informasi, Statistik dan Buku yang sedang dipinjam



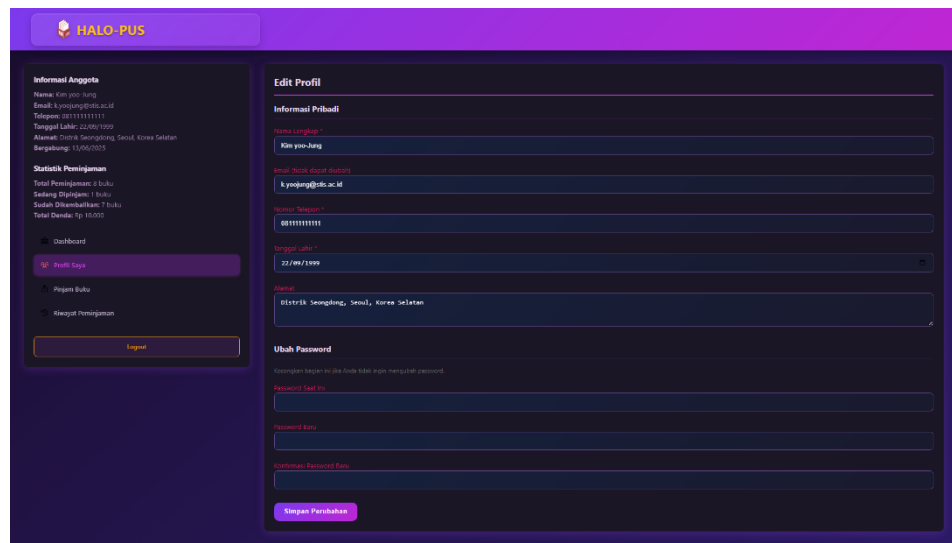
Gambar 2.27 Riwayat Peminjaman

2. Profil Saya

a. Deskripsi:

Anggota bisa melihat dan mengedit data pribadi, termasuk ubah password.

b. Screenshot:



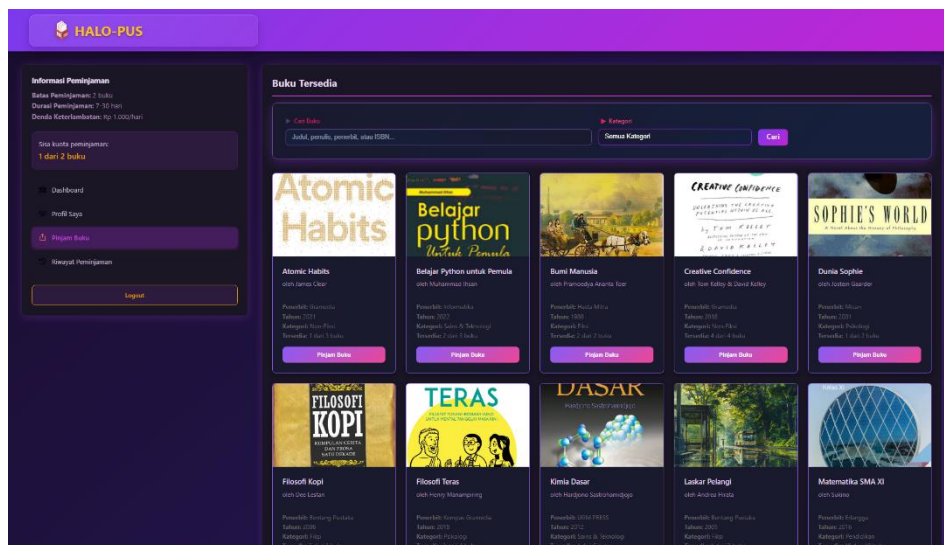
Gambar 2.28 Halaman Profil Saya

3. Pinjam Buku

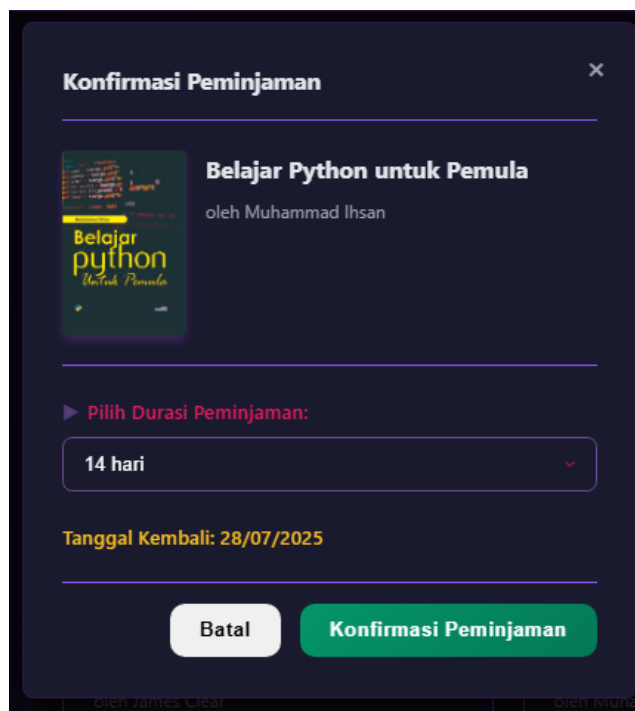
a. Deskripsi:

Anggota bisa meminjam buku secara online tanpa harus ke admin.

b. Screenshot:



Gambar 2.29 Halaman Pinjam Buku



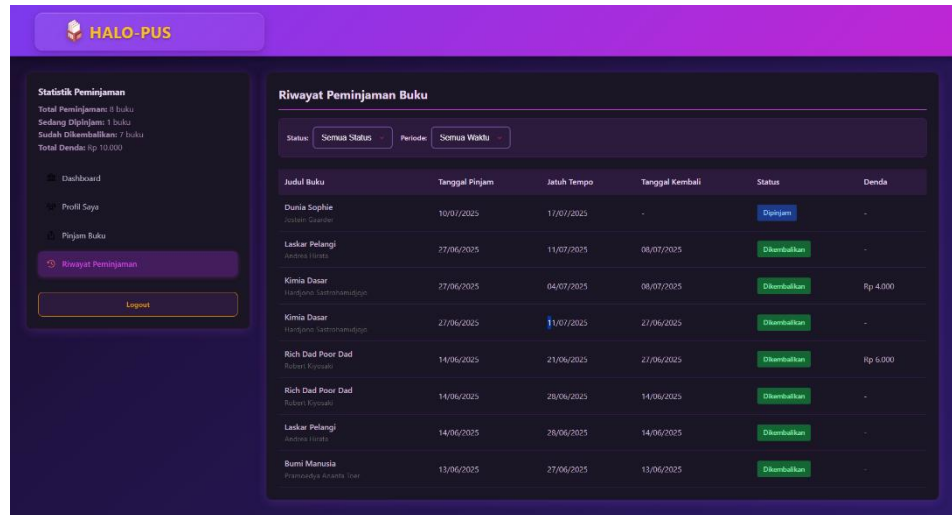
Gambar 2.30 Halaman Konfirmasi

4. Riwayat Peminjaman

a. Deskripsi:

Menampilkan daftar buku yang pernah dipinjam, statusnya, dan jumlah denda.

b. Screenshot:



Judul Buku	Tanggal Pinjam	Jatuh Tempo	Tanggal Kembali	Status	Denda
Dunia Sophie Heidi Pitlor	10/01/2025	11/01/2025	-	Dipinjam	-
Laskar Pelangi Andrea Hirata	27/06/2025	11/07/2025	08/07/2025	Dikembalikan	-
Kimia Dasar Hartono Satromandjaja	27/06/2025	04/07/2025	08/07/2025	Dikembalikan	Rp 4.000
Kimia Dasar Hartono Satromandjaja	27/06/2025	11/07/2025	27/06/2025	Dikembalikan	-
Rich Dad Poor Dad Robert Kiyosaki	14/06/2025	21/06/2025	27/06/2025	Dikembalikan	Rp 6.000
Rich Dad Poor Dad Robert Kiyosaki	14/06/2025	28/06/2025	14/06/2025	Dikembalikan	-
Laskar Pelangi Andrea Hirata	14/06/2025	28/06/2025	14/06/2025	Dikembalikan	-
Bumi Manusia Pramoedya Ananta Toer	13/06/2025	27/06/2025	13/06/2025	Dikembalikan	-

Gambar 2.31 Halaman Riwayat Peminjaman

D. Contoh Penggunaan Sistem

1. Contoh 1: Admin melakukan peminjaman buku

- Login sebagai admin
- Masuk ke menu Peminjaman
- Pilih anggota
- Pilih buku
- Atur tanggal pengembalian
- Simpan transaksi

2. Contoh 2: Anggota meminjam buku via web

- Login sebagai anggota
- Masuk ke menu Pinjam Buku
- Cari buku yang diinginkan
- Klik Pinjam

BAB III

PENUTUP

A. Kesimpulan

Berdasarkan hasil pengembangan dan implementasi aplikasi Halopus, dapat disimpulkan bahwa sistem informasi perpustakaan berbasis web ini mampu membantu proses pengelolaan data buku, data anggota, peminjaman, dan pengembalian buku secara lebih efektif dan efisien dibandingkan metode manual. Aplikasi ini menyediakan fitur-fitur yang dapat diakses oleh dua jenis pengguna, yaitu admin dan anggota, dengan hak akses dan layanan yang berbeda sesuai peran masing-masing.

Dengan adanya sistem ini, proses pencarian data buku dan anggota menjadi lebih cepat, proses administrasi peminjaman dapat dilakukan dengan lebih rapi dan terstruktur, serta pengelolaan data anggota dan buku dapat dipantau secara realtime melalui dashboard. Selain itu, anggota juga diberikan kemudahan untuk meminjam buku secara mandiri melalui akun masing-masing.

B. Saran

Agar sistem ini dapat terus berkembang dan memenuhi kebutuhan pengguna di masa depan, berikut beberapa saran pengembangan:

1. Menambahkan fitur notifikasi pengingat tanggal pengembalian buku melalui email atau SMS.
2. Mengembangkan fitur cetak laporan peminjaman dan pengembalian dalam format PDF atau Excel.
3. Menyediakan fitur upload cover buku secara otomatis saat menambah data buku.
4. Menambahkan transaksi sekaligus beberapa buku dalam satu kali pinjam.
5. Meningkatkan tampilan antarmuka agar lebih modern dan responsif untuk berbagai perangkat mobile.

C. Tantangan yang Dihadapi

Selama proses pengembangan Halopus, terdapat beberapa tantangan yang dihadapi, antara lain:

1. Perancangan database relasional yang harus dapat mengakomodasi berbagai jenis data sekaligus memastikan relasi antar tabel berjalan optimal.

2. Implementasi sistem login multi-role, di mana perlu memastikan bahwa hak akses setiap pengguna berjalan sesuai peran yang ditentukan.
3. Pengaturan validasi data agar sistem tidak menerima data ganda atau data yang tidak sesuai format.
4. Pengujian fitur-fitur dinamis seperti live search, filter, dan penghitungan denda otomatis yang memerlukan debugging cukup lama untuk memastikan berjalan stabil.
5. Pengelolaan tampilan antarmuka agar dapat berjalan lancar di berbagai ukuran layar dan perangkat.

Meskipun menghadapi beberapa kendala, seluruh fitur utama yang dirancang dapat diimplementasikan dan berjalan sesuai dengan kebutuhan yang telah direncanakan.