

DPPL-xx

DESKRIPSI PERANCANGAN PERANGKAT LUNAK

LIBRARY MANAGEMENT SYSTEM

untuk:

Bapak Cholid Fauzi

Dipersiapkan oleh:

Group A

Khairunida Khumaira 1301223216

Frederico De Rosari 1301220215

Shalisyah Amania Fatiha 1301223251

Putri Adinda Syakira 1301223073

Aaron Barichello 1301223337

Program Studi Teknologi Informasi

Fakultas Informatika

Jl. Telekomunikasi 1, Dayeuhkolot Bandung

Telkom University	Prodi S1- Teknologi Informasi Universitas Telkom	Nomor Dokumen		Halaman
		DPPL-xx <xx:no grp>		<#/>/<jml #
		Revisi	<nomor revisi>	Tgl: <isi tanggal>

Prodi S1 Informatika Tel-U	DPPL-XXX	Halaman 2 dari 70
Template dokumen ini dan informasi yang dimilikinya adalah milik Prodi S1 Informatika Tel-U dan bersifat rahasia. Dilarang me-reproduksi dokumen ini tanpa diketahui oleh Prodi S1 Informatika Tel-U.		

DAFTAR PERUBAHAN

Revisi	Deskripsi
A	
B	
C	
D	
E	
F	
G	

INDEX TGL	-	A	B	C	D	E	F	G
Ditulis oleh								
Diperiksa oleh								
Disetujui oleh								

Daftar Halaman Perubahan

Halaman	Revisi	Halaman	Revisi

Daftar Isi

Daftar Isi.....	3
Daftar Tabel.....	5
Daftar Gambar.....	6
1. Pendahuluan.....	6
1.1 Tujuan Penulisan Dokumen.....	7
1.2 Lingkup Masalah.....	7
1.3 Definisi dan Istilah.....	7
1.4 Referensi.....	7
2 Perancangan Global.....	8
2.1 Rencana Lingkungan Implementasi.....	8
2.2 Deskripsi Arsitektur Perangkat Lunak.....	8
3 Perancangan Rinci.....	10
3.1 Realisasi Use Case.....	10
3.1.1 Use Case #1 Registrasi Anggota.....	11
3.1.1.1 Use Case Scenario #1 Registrasi Anggota.....	11
3.1.1.2 UI Design dan Deskripsi Objek UI #1 Registrasi Anggota.....	12
3.1.1.3 Identifikasi Object dan Tipe Kelas #1 Registrasi Anggota.....	14
3.1.1.4 Sequence Diagram #1 Registrasi Anggota.....	15
3.1.2 Use Case #2 Login.....	15
3.1.2.1 Use Case Scenario #2 Login.....	15
3.1.2.2 UI Design dan Deskripsi Objek UI #2 Login.....	16
3.1.2.3 Identifikasi Object Baru & Tipe Kelas #2 Login.....	18
3.1.2.4 Sequence Diagram #2 Login.....	19
3.1.3 Use Case #3 Request Peminjaman Buku.....	19
3.1.3.1 Use Case Scenario #3 Request Peminjaman Buku.....	19
3.1.3.2 UI Design dan Deskripsi Objek UI #3 Request Peminjaman Buku.....	20
3.1.3.3 Identifikasi Object Baru & Tipe Kelas #3 Request Peminjaman Buku.....	23
3.1.3.4 Sequence Diagram #3 Request Peminjaman Buku.....	24
3.1.4 Use Case #4 Status Peminjaman Buku.....	24
3.1.4.1 Use Case Scenario #4 Status Peminjaman Buku.....	24
3.1.4.2 UI Design dan Deskripsi Objek UI #4 Status Peminjaman Buku.....	25
3.1.4.3 Identifikasi Object Baru & Tipe Kelas #4 Status Peminjaman Buku:.....	26
3.1.4.4 Sequence Diagram #4 Status Peminjaman Buku.....	27
3.1.5 Use Case #5 Pengembalian Buku.....	27
3.1.5.1 Use Case Scenario #5 Pengembalian Buku.....	27
3.1.5.2 UI Design dan Deskripsi Objek UI #5 Pengembalian Buku.....	28
3.1.5.3 Identifikasi Object Baru & Tipe Kelas #5 Pengembalian Buku.....	30
3.1.5.4 Sequence Diagram #5 Pengembalian Buku.....	30

3.1.6 Use Case #6 History Peminjaman Buku.....	31
3.1.6.1 Use Case Scenario #6 History Peminjaman Buku.....	31
3.1.6.2 UI Design dan Deskripsi Objek UI #6 History Peminjaman Buku.....	32
3.1.6.3 Identifikasi Object Baru & Tipe Kelas #6 History Peminjaman Buku.....	33
3.1.6.4 Sequence Diagram #6 History Peminjaman Buku.....	33
3.1.7 Use Case #7 Menyetujui atau Menolak Peminjaman Buku.....	33
3.1.7.1 Use Case Scenario #7 Menyetujui atau Menolak Peminjaman Buku.....	33
3.1.7.2 UI Design dan Deskripsi Objek UI #7 Menyetujui atau Menolak Peminjaman Buku.....	34
3.1.7.3 Identifikasi Object Baru & Tipe Kelas #7 Menyetujui atau Menolak Peminjaman Buku.....	37
3.1.7.4 Sequence Diagram #7 Menyetujui atau Menolak Peminjaman Buku.....	38
3.1.8 Use Case #8 Update Status Peminjaman Buku.....	38
3.1.8.1 Use Case Scenario #8 Update Status Peminjaman Buku.....	38
3.1.8.2 UI Design dan Deskripsi Objek UI #8 Update Status Peminjaman Buku.....	40
3.1.8.3 Identifikasi Object Baru & Tipe Kelas #8 Update Status Peminjaman Buku.....	42
3.1.8.4 Sequence Diagram #8 Update Status Peminjaman Buku.....	43
3.1.9 Use Case #9 Menyetujui Pengembalian Buku.....	43
3.1.9.1 Use Case Scenario #9 Menyetujui Pengembalian Buku.....	43
3.1.9.2 UI Design dan Deskripsi Objek UI #9 Menyetujui Pengembalian Buku.....	44
3.1.9.3 Identifikasi Object Baru & Tipe Kelas #9 Menyetujui Pengembalian Buku.....	46
3.1.9.4 Sequence Diagram #9 Menyetujui Pengembalian Buku.....	47
3.1.10 Use Case #10 Mencari Buku di Perpustakaan.....	47
3.1.10.1 Use Case Scenario #10 Mencari Buku di Perpustakaan.....	47
3.1.10.2 UI Design dan Deskripsi Objek UI #10 Mencari Buku di Perpustakaan.....	49
3.1.10.3 Identifikasi Object Baru & Tipe Kelas #10 Mencari Buku di Perpustakaan.....	51
3.1.10.4 Sequence Diagram #10 Mencari Buku di Perpustakaan.....	52
3.1.11.2 UI Design dan Deskripsi Objek UI #11 Melihat Laporan Pengunjung.....	53
3.1.11.3 Identifikasi Object Baru & Tipe Kelas #11 Melihat Laporan Pengunjung.....	54
3.1.11.4 Sequence Diagram #11 Melihat Laporan Pengunjung.....	55
3.2 Diagram Kelas Keseluruhan.....	56
3.3 Perancangan Detail Kelas.....	56
3.4 Perancangan Algoritma dan/atau Query.....	60
4 Matriks Kerunutan (Requirement Traceability Matrix).....	68

Setelah Daftar Isi Boleh ada Daftar Tabel dan Daftar Gambar

Daftar Tabel

Prodi S1 Informatika Tel-U	DPPL-XXX	Halaman 7 dari 70
Template dokumen ini dan informasi yang dimilikinya adalah milik Prodi S1 Informatika Tel-U dan bersifat rahasia. Dilarang me-reproduksi dokumen ini tanpa diketahui oleh Prodi S1 Informatika Tel-U.		

Daftar Gambar

1. Pendahuluan

1.1 Tujuan Penulisan Dokumen

Tujuan dari Dokumen Desain Perangkat Lunak Library Management Berbasis Web (DPPL) adalah untuk memberikan panduan rinci tentang struktur, fungsionalitas, dan karakteristik perangkat lunak yang dibuat. Dokumen ini menjelaskan detail teknis, keputusan arsitektur, algoritme, dan komponen utama perangkat lunak library management kami. DPPL ini digunakan oleh tim pengembang perangkat lunak untuk memahami rencana pengembangan dan pedoman yang harus diikuti selama proses pengembangan perangkat lunak.

1.2 Lingkup Masalah

Dalam penulisan DPPL ini, digunakan standar penulisan yang mengacu pada gaya penulisan teknis umum. Setiap pernyataan requirements dianggap memiliki prioritas yang sama kecuali dinyatakan sebaliknya. Dokumen ini mencakup semua aspek yang berkaitan dengan persyaratan perangkat lunak untuk Library Management Berbasis Web, termasuk antarmuka pengguna, fungsionalitas utama, dan ketergantungan pada faktor eksternal.

1.3 Definisi dan Istilah

Keyword	Definisi
DPPL	Dokumen yang berisi spesifikasi kebutuhan perangkat lunak yang akan dikembangkan.
User	Aktor yang menggunakan perangkat lunak tersebut untuk mendapatkan tempat management buku yang mereka punya.
Database	Data yang dikelola tersimpan di dalam server.
Perangkat	Sebuah objek yang mengacu pada komputer dan laptop beserta spesifikasi dan komponennya.
Admin	Aktor yang mengatur semua fungsi dan sistem dalam aplikasi website.
Sistem	Sarana yang dapat memberikan saran buku yang sesuai dengan preferensi pengguna

1.4 Referensi

Referensi yang digunakan dalam penulisan DPPL ini meliputi :

- [Draw.io](#)
- SKPL Library Management

2 Perancangan Global

2.1 Rencana Lingkungan Implementasi

Perangkat Lunak (PL) Sistem Manajemen Perpustakaan ini memiliki dua antarmuka utama, yaitu untuk pengunjung dan admin. Pengunjung dapat mencari buku dan login untuk mengakses laman menu anggota, dengan ketersediaan fitur status peminjaman dan riwayat peminjaman untuk anggota yang sudah login. Admin memiliki kontrol penuh terhadap manajemen anggota, peminjaman, pengembalian, serta pembuatan, pengeditan, dan penghapusan keanggotaan. Proses peminjaman buku melibatkan modul pencarian buku, login, peminjaman buku yang membutuhkan persetujuan admin, serta modul pengembalian buku dengan konfirmasi admin. Modul laporan pengunjung menyediakan informasi tentang aktivitas pengguna untuk membantu admin dalam pengambilan keputusan terkait manajemen perpustakaan. Keseluruhan, arsitektur ini memisahkan peran antara pengunjung dan admin, memberikan fungsionalitas yang terorganisir dan terkendali.

2.2 Deskripsi Arsitektur Perangkat Lunak

No	Nama Komponen	Keterangan
1	Pencarian buku	Modul ini memungkinkan pengguna untuk mencari buku dalam perpustakaan berdasarkan kriteria tertentu, seperti judul, penulis, atau kategori.
2	Login	Modul autentikasi yang memungkinkan pengguna untuk masuk ke sistem dengan menyediakan informasi login yang valid.
3	Status Peminjaman	Memberikan akses kepada anggota perpustakaan untuk melihat status peminjaman buku yang sedang berlangsung.
4	Riwayat Peminjaman	Modul ini memungkinkan anggota perpustakaan untuk melihat riwayat peminjaman buku yang pernah ia dilakukan.
5	Register Anggota	Modul untuk membuat akun anggota baru dalam sistem perpustakaan. Hanya dapat diakses oleh admin.

6	Update Status Peminjaman Buku	Modul yang mengelola pembaruan status peminjaman buku, misalnya, konfirmasi peminjaman.
7	Peminjaman buku	Modul untuk mengelola proses peminjaman buku yang ada di perpustakaan.
8	Konfirmasi Peminjaman Buku	Modul yang memproses konfirmasi peminjaman buku, Melibatkan anggota untuk memasukan password.
9	Pengembalian Buku	Modul untuk mengelola proses pengembalian buku ke perpustakaan.
10	Konfirmasi Pengembalian Buku	Modul yang memproses konfirmasi pengembalian buku, Melibatkan anggota untuk memasukan password.
11	Laporan Pengunjung	Mengumpulkan dan menyajikan data terkait penggunaan perpustakaan, seperti laporan statistik pengunjung.

3 Perancangan Rinci

3.1 Realisasi Use Case

Berisi TABEL USE CASE sebagai berikut :

No	Nama Use Case	Deskripsi Use Case
#1	Registrasi Anggota	Use Case ini memungkinkan calon anggota untuk mendaftarkan diri sebagai pengguna perpustakaan. Pengguna harus memasukkan informasi pribadi mereka ke dalam formulir pendaftaran dan menerima konfirmasi melalui email untuk mengaktifkan akun mereka.
#2	Login	Use Case ini memungkinkan pengguna yang telah terdaftar untuk masuk ke dalam sistem menggunakan username dan password yang telah ditetapkan saat registrasi. Setelah berhasil login, pengguna akan diarahkan ke dashboard utama.
#3	Request Peminjaman Buku	Use Case ini digunakan oleh pengguna untuk mengajukan peminjaman buku melalui sistem perpustakaan berbasis web.
#4	Status Peminjaman Buku	Use Case ini digunakan oleh pengguna untuk mengecek status terkini dari buku-buku yang sedang dipinjam atau dalam proses peminjaman.
#5	Pengembalian Buku	Use Case ini digunakan oleh pengguna untuk melakukan proses pengembalian buku yang telah dipinjam melalui sistem perpustakaan berbasis web.
#6	History Peminjaman Buku	Use Case ini memungkinkan pengguna untuk melihat riwayat lengkap dari semua buku yang telah dipinjam dari perpustakaan melalui sistem.
#7	Menyetujui atau Menolak Peminjaman Buku	Use Case ini digunakan oleh pustakawan untuk menyetujui atau menolak permintaan peminjaman buku yang diajukan oleh pengguna.
#8	Update Status Peminjaman Buku	Use Case ini digunakan oleh pustakawan untuk memperbarui status buku yang sedang dipinjam, seperti memperbarui tanggal pengembalian, menandai buku sebagai dikembalikan, atau memperbarui kondisi buku setelah pemeriksaan.
#9	Menyetujui Pengembalian Buku	Use Case ini digunakan oleh pustakawan untuk memverifikasi dan menyetujui pengembalian buku yang telah dilakukan oleh pengguna.
#10	Mencari Buku di Perpustakaan	Use Case ini memungkinkan pengguna untuk mencari dan menemukan buku yang tersedia dalam koleksi perpustakaan menggunakan sistem online.
#11	Melihat Laporan Pengunjung	Use Case ini digunakan oleh pustakawan untuk meninjau dan menganalisis data pengunjung yang telah masuk ke perpustakaan dalam periode tertentu.

3.1.1 Use Case #1 Registrasi Anggota

3.1.1.1 Use Case Scenario #1 Registrasi Anggota

Skenario Use Case #1 :

- i. Pre-Condition : Pengguna belum terdaftar sebagai anggota perpustakaan.
- ii. Use Case Description : Use Case ini memungkinkan individu untuk mendaftar sebagai anggota perpustakaan melalui sistem online, memungkinkan mereka untuk mengakses layanan dan sumber daya perpustakaan..
 - a. Primary Flow

No	Alur Utama
1	Aktor : Memilih opsi 'Daftar sebagai Anggota' pada website perpustakaan atau sistem online.
2	Sistem : Menampilkan form pendaftaran online untuk diisi.
3	Aktor : Mengisi form dengan informasi yang diperlukan seperti nama lengkap, alamat, email, dan informasi lain yang relevan.
4	Sistem : Memproses informasi yang dimasukkan dan melakukan verifikasi data.
5	Aktor : Mengunggah dokumen identifikasi atau dokumen lain yang diperlukan jika diminta.
6	Sistem : Mengkonfirmasi pendaftaran dan mengirim email verifikasi atau notifikasi ke pengguna.
7	Sistem : Menciptakan akun anggota dalam sistem dan memberikan akses ke layanan perpustakaan.

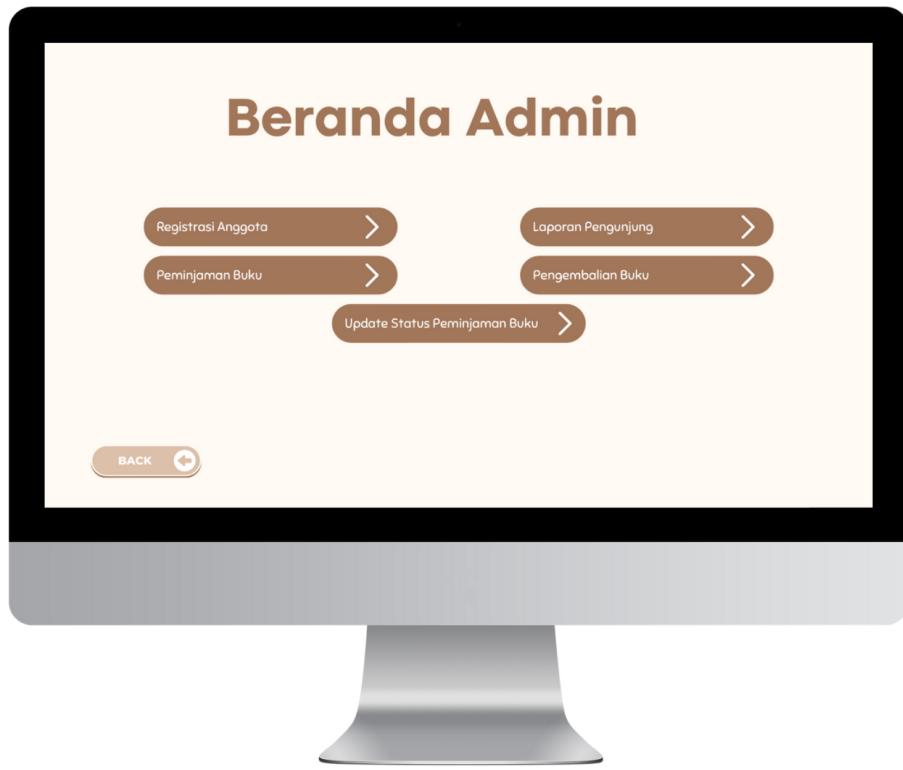
- b. Alternative Flow

No	Alur Alternatif
1	Jika ada kesalahan dalam mengisi form. Maka sistem menampilkan pesan kesalahan dan meminta pengguna untuk memperbaiki informasi yang salah.
2	Jika sistem gagal memproses pendaftaran. Maka sistem menampilkan pesan error dan menginstruksikan pengguna untuk mencoba kembali atau menghubungi perpustakaan untuk bantuan lebih lanjut.
3	Jika dokumen yang diunggah tidak memenuhi syarat. Maka sistem memberikan feedback tentang masalah dengan dokumen yang diunggah dan meminta pengguna untuk mengunggah dokumen yang sesuai.

- iii. Post-Condition : Pengguna berhasil terdaftar sebagai anggota perpustakaan dan dapat mengakses layanan yang tersedia.

3.1.1.2 UI Design dan Deskripsi Objek UI #1 Registrasi Anggota





Tabel Deskripsi Objek UI

ID. LAYA R	NAMA LAYAR	DESKRIPSI
P001	Registrasi	Menu Registrasi
P002	Registrasi berhasil	Registrasi berhasil
P003	Registrasi gagal	Registrasi gagal

UNTUK MASING – MASING ANTAR MUKA / PAGE dibuatkan spesifikasi detail
Antarmuka XXX: {diisi dengan no. layar atau no gambar rancangan antarmuka}

Page Registrasi Anggota

ID_OBJEK	JENIS	LABEL*	Keterangan**
Teks1	Text Box	Nama	Admin mengisi nama
Teks2	Text Box	Username	User mengisi username
Teks3	Text Box	Kata Sandi	User mengisi kata sandi
Teks4	Text Box	Email	Admin mengisi email
Teks5	Text Box	No Telepon	Admin mengisi no telepon
Teks6	Text Box	Jenis Kelamin	Admin mengisi jenis kelamin
Teks7	Text Box	Usia	Admin mengisi usia
Teks8	Text Box	Alamat	Admin mengisi alamat
Button1	Button	Submit	Jika diklik akan melakukan verifikasi data user.
Button2	Button	Back	Kembali ke halaman beranda Admin.

Page Registrasi Anggota berhasil

Prodi S1 Informatika Tel-U	DPPL-XXX	Halaman 15 dari 70
Template dokumen ini dan informasi yang dimilikinya adalah milik Prodi S1 Informatika Tel-U dan bersifat rahasia. Dilarang me-reproduksi dokumen ini tanpa diketahui oleh Prodi S1 Informatika Tel-U.		

Id_Objek	JENIS	LABEL*	Keterangan**
Teks1	Text Box	Nama	Admin mengisi nama
Teks2	Text Box	Username	User mengisi username
Teks3	Text Box	Kata Sandi	User mengisi kata sandi
Teks4	Text Box	Email	Admin mengisi email
Teks5	Text Box	No Telepon	Admin mengisi no telepon
Teks6	Text Box	Jenis Kelamin	Admin mengisi jenis kelamin
Teks7	Text Box	Usia	Admin mengisi usia
Teks8	Text Box	Alamat	Admin mengisi alamat
Button1	Button	Submit	Jika diklik akan melakukan verifikasi data user, jika valid Admin akan didirect ke halaman beranda Admin.
Button2	Button	Back	Kembali ke halaman beranda Admin.

Page Alternatif Registrasi Anggota gagal

Id_Objek	JENIS	LABEL*	Keterangan**
Teks1	Text Box	Nama	Admin mengisi nama
Teks2	Text Box	Username	User mengisi username
Teks3	Text Box	Kata Sandi	User mengisi kata sandi
Teks4	Text Box	Email	Admin mengisi email
Teks5	Text Box	No Telepon	Admin mengisi no telepon
Teks6	Text Box	Jenis Kelamin	Admin mengisi jenis kelamin
Teks7	Text Box	Usia	Admin mengisi usia
Teks8	Text Box	Alamat	Admin mengisi alamat
Button1	Button	Submit	Jika diklik akan melakukan verifikasi data user, jika tidak valid Admin akan didirect kembali ke halaman registrasi.
Button2	Button	Back	Kembali ke halaman beranda Admin.

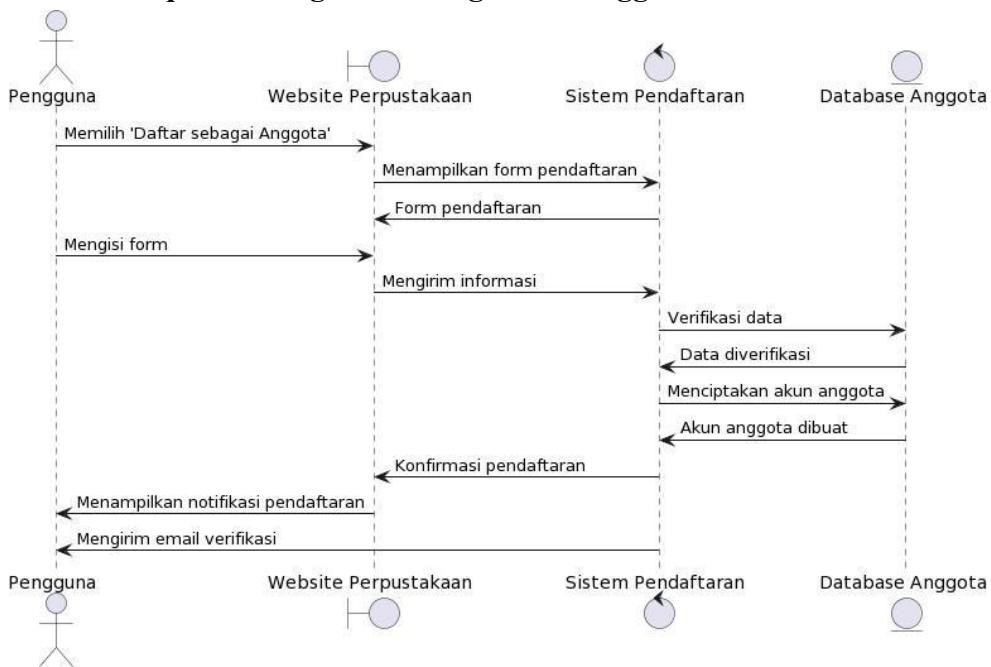
3.1.1.3 Identifikasi Object dan Tipe Kelas #1 Registrasi Anggota

TABEL OBJECT PERANCANGAN

No	Nama Object Baru	Jenis / Tipe Kelas
1	Page Register Anggota	Boundary
2	Page Register Anggota berhasil	Boundary
3	Page Register Anggota gagal	Boundary
4	Button Submit	Controller
5	Button Back	Controller
6	Database User	Entity

*Tipe kelas seperti Boundary(Interface), Entity(Database), Controller

3.1.1.4 Sequence Diagram #1 Registrasi Anggota



3.1.2 Use Case #2 Login

3.1.2.1 Use Case Scenario #2 Login

Skenario Use Case #2 :

- Pre-Condition : Pengguna berada di halaman login.
- Use Case Description : Use Case ini digunakan oleh pengguna untuk mengakses sistem dengan memasukkan kredensial yang valid.
 - Primary Flow

No	Alur Utama
1	Aktor : Memasukkan nama pengguna.
2	Sistem : Meminta kata sandi
3	Aktor : Memasukkan kata sandi.
4	Sistem : Memvalidasi kredensial.
5	Sistem : Menampilkan dashboard jika kredensial benar.
6	Sistem : Menampilkan pesan error jika kredensial salah.

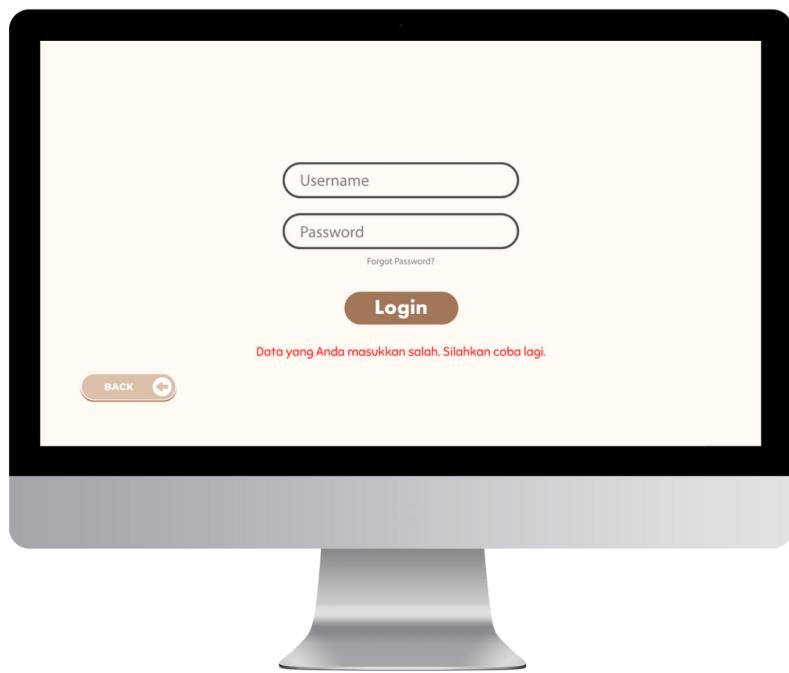
b. Alternative Flow

No	Alur Alternatif
1	Jika kredensial salah. Maka sistem menampilkan pesan error dan meminta pengguna untuk mencoba lagi.
2	Jika lupa sandi. Maka pengguna memilih opsi “Lupa Sandi”. Dan sistem memberikan instruksi untuk reset sandi melalui email.

- iii. Post-Condition : Pengguna berhasil masuk ke dashboard utama aplikasi manajemen perpustakaan.

3.1.2.2 UI Design dan Deskripsi Objek UI #2 Login





Tabel Deskripsi Objek UI

ID. LAYA R	NAMA LAYAR	DESKRIPSI
P001	Login	Menu login
P002	Login berhasil	Beranda Anggota ketika login berhasil
P003	Login gagal	Menu login ketika login gagal

UNTUK MASING – MASING ANTAR MUKA / PAGE dibuatkan spesifikasi detil
Antarmuka XXX: {diisi dengan no. layar atau no gambar rancangan antarmuka}

Page Login

Id_Objek	JENIS	LABEL*	Keterangan**
Teks1	Text Box	Username	User mengisi username
Teks2	Text Box	Password	User mengisi password
Button1	Button	Login	Jika diklik akan melakukan verifikasi data user.
Button2	Button	Back	Kembali ke halaman mencari buku.

Page Login berhasil

Id_Objek	JENIS	LABEL*	Keterangan**
Teks1	Text Box	Username	User mengisi username
Teks2	Text Box	Password	User mengisi password
Button1	Button	Login	Jika diklik akan melakukan verifikasi data user, jika valid user akan didirect ke beranda anggota.
Button2	Button	Back	Kembali ke halaman mencari buku.

Page Alternatif Login gagal

Id_Objek	JENIS	LABEL*	Keterangan**
Teks1	Text Box	Username	User mengisi username
Teks2	Text Box	Password	User mengisi password
Button1	Button	Login	Jika diklik akan melakukan verifikasi data user, jika tidak valid user akan didirect ke halaman login.
Button2	Button	Back	Kembali ke halaman mencari buku.

3.1.2.3 Identifikasi Object Baru & Tipe Kelas #2 Login

TABEL OBJECT PERANCANGAN

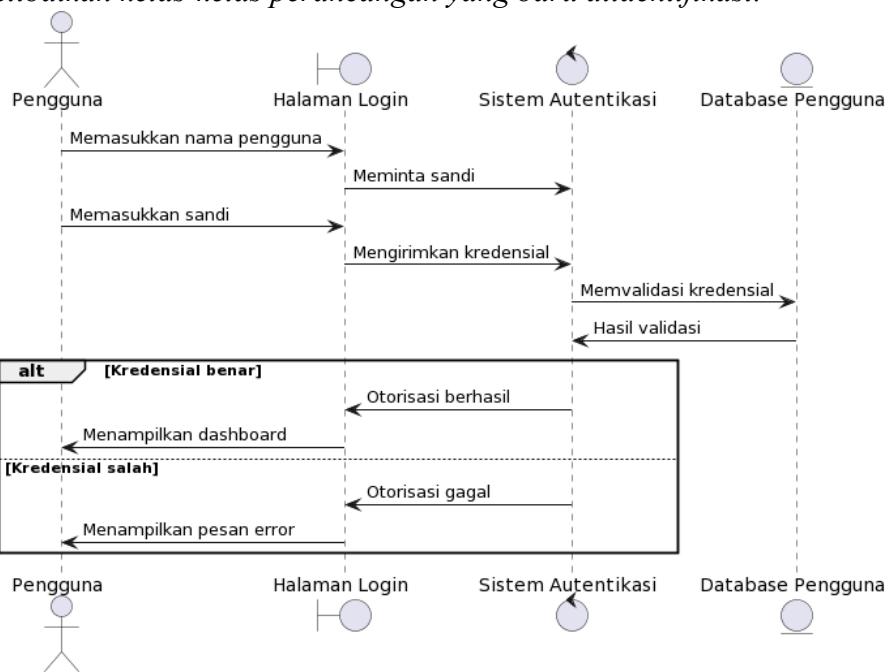
No	Nama Object Baru	Jenis / Tipe Kelas
1	Page Login	Boundary
2	Page Login berhasil	Boundary
3	Page Login gagal	Boundary
4	Button Login	Controller
5	Button Back	Controller
6	Database user	Entity

*Tipe kelas seperti Boundary(Interface), Entity(Database), Controller

3.1.2.4 Sequence Diagram #2 Login

Buatlah diagram sequence untuk masing - masing use case sesuai skenario.

Skenario harus melibatkan kelas-kelas perancangan yang baru diidentifikasi.



3.1.3 Use Case #3 Request Peminjaman Buku

3.1.3.1 Use Case Scenario #3 Request Peminjaman Buku

Skenario Use Case #3 :

- Pre-Condition : Pengguna telah berhasil login ke dalam sistem.
- Use Case Description : Use Case ini digunakan oleh pengguna untuk mengajukan peminjaman buku melalui sistem perpustakaan berbasis web.
 - Primary Flow

No	Alur Utama
1	Aktor : Memilih opsi 'Peminjaman Buku' dari menu utama.
2	Sistem : Menampilkan daftar buku yang tersedia untuk dipinjam
3	Aktor : Mencari dan memilih buku yang diinginkan.
4	Sistem : Menampilkan detail buku dan formulir peminjaman.
5	Aktor : Mengisi formulir peminjaman dengan informasi yang diperlukan (tanggal peminjaman, durasi, dll.)
6	Sistem : Mengonfirmasi detail peminjaman dan mengajukan request.
7	Sistem : Menyimpan request peminjaman dan memberikan notifikasi

berhasil kepada pengguna.

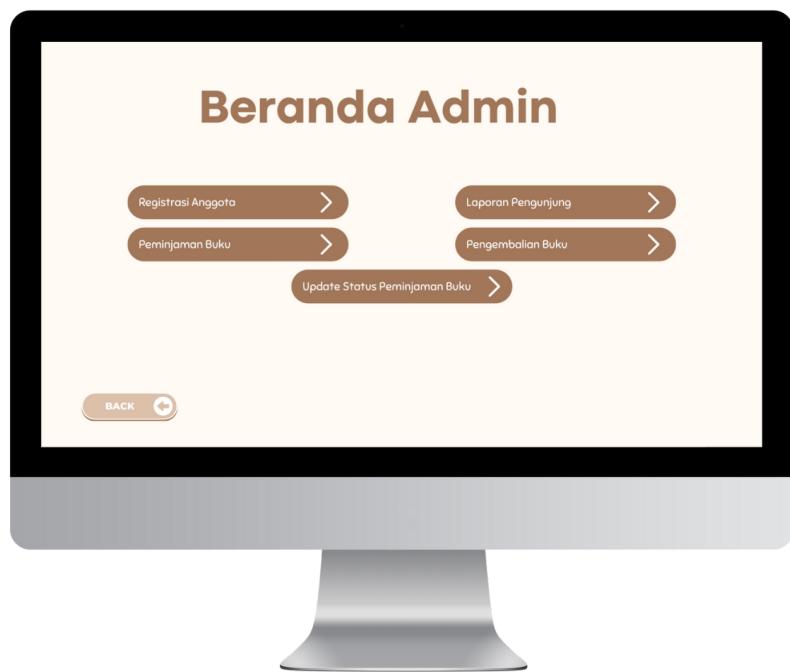
b. Alternative Flow

No	Alur Alternatif
1	Jika buku tidak tersedia. Maka sistem menampilkan pesan bahwa buku sedang dipinjam orang lain dan memberikan opsi untuk memasukkan diri ke daftar tunggu.
2	Jika pengguna melebihi batas peminjaman. Maka sistem menolak request peminjaman dan memberikan informasi bahwa pengguna telah melebihi batas jumlah peminjaman yang diizinkan.
3	Jika terjadi kesalahan sistem. Maka sistem menampilkan pesan error dan meminta pengguna untuk mencoba lagi nanti.

- iii. Post-Condition : Permintaan peminjaman buku telah tercatat di sistem dan menunggu persetujuan dari pustakawan.

3.1.3.2 UI Design dan Deskripsi Objek UI #3 Request Peminjaman Buku





Tabel Deskripsi Objek UI

ID. LAYA R	NAMA LAYAR	DESKRIPSI
P001	<i>Peminjaman Buku</i>	<i>Menu peminjaman buku.</i>
P002	<i>Peminjaman Buku berhasil</i>	<i>Halaman konfirmasi peminjaman buku ketika peminjaman buku berhasil.</i>
P003	<i>Peminjaman Buku gagal</i>	<i>Kembali ke menu peminjaman buku ketika peminjaman buku gagal.</i>

UNTUK MASING – MASING ANTAR MUKA / PAGE dibuatkan spesifikasi detil

Antarmuka XXX: {diisi dengan no. layar atau no gambar rancangan antarmuka}

Page Peminjaman Buku

Id_Objek	JENIS	LABEL*	Keterangan**
Teks1	Text Box	Username Anggota	Admin mengisi username anggota.
Teks2	Text Box	Nama Peminjam	Admin mengisi nama peminjam.
Teks3	Text Box	Tanggal Peminjaman	Admin mengisi tanggal peminjaman.
Teks4	Text Box	ID Buku	Admin mengisi id buku.
Teks5	Text Box	Judul Buku	Admin mengisi judul buku.
Teks6	Text Box	Tanggal Pengembalian	Admin mengisi tanggal pengembalian.
Button1	Button	Continue	Jika diklik akan melakukan verifikasi data user.
Button2	Button	Back	Kembali ke beranda Admin.

Page Peminjaman Buku berhasil

Id_Objek	JENIS	LABEL*	Keterangan**
Teks1	Text Box	Username Anggota	Admin mengisi username anggota.
Teks2	Text Box	Nama Peminjam	Admin mengisi nama peminjam.
Teks3	Text Box	Tanggal Peminjaman	Admin mengisi tanggal peminjaman.
Teks4	Text Box	ID Buku	Admin mengisi id buku.
Teks5	Text Box	Judul Buku	Admin mengisi judul buku.
Teks6	Text Box	Tanggal Pengembalian	Admin mengisi tanggal pengembalian.
Button1	Button	Continue	Jika diklik akan melakukan verifikasi data user, jika valid Admin akan didirect ke beranda Admin
Button2	Button	Back	Kembali ke beranda Admin.

Page Peminjaman Buku gagal

Id_Objek	JENIS	LABEL*	Keterangan**
Teks1	Text Box	Username Anggota	Admin mengisi username anggota.
Teks2	Text Box	Nama Peminjam	Admin mengisi nama peminjam.
Teks3	Text Box	Tanggal Peminjaman	Admin mengisi tanggal peminjaman.

Id_Objek	JENIS	LABEL*	Keterangan**
Teks4	Text Box	ID Buku	Admin mengisi id buku.
Teks5	Text Box	Judul Buku	Admin mengisi judul buku.
Teks6	Text Box	Tanggal Pengembalian	Admin mengisi tanggal pengembalian.
Button1	Button	Continue	Jika diklik akan melakukan verifikasi data user, jika tidak valid Admin akan didirect ke halaman peminjaman buku.
Button2	Button	Back	Kembali ke beranda Admin.

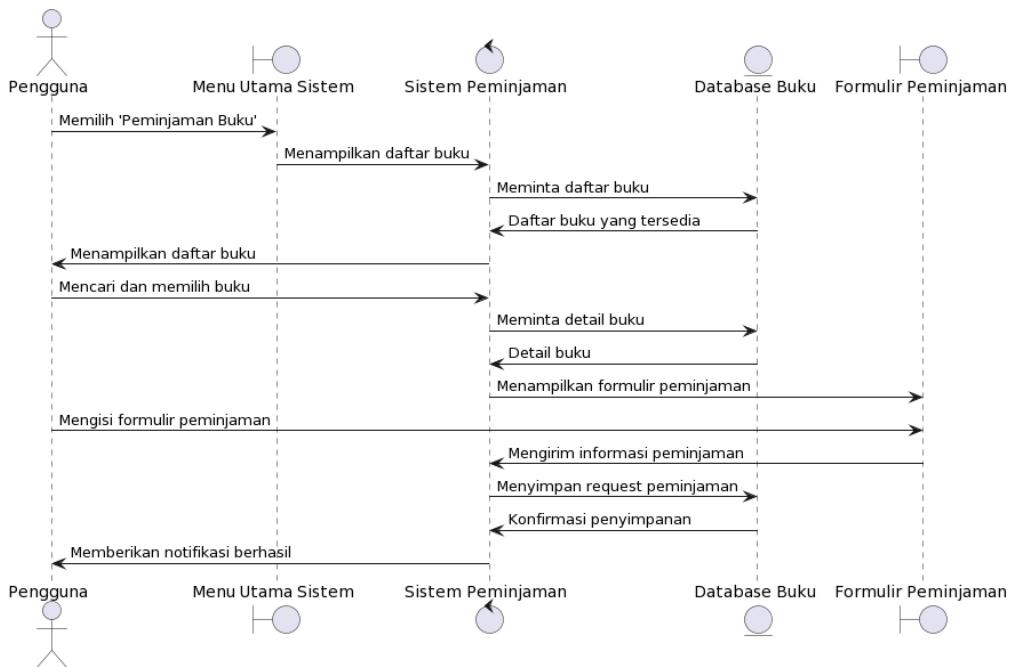
3.1.3.3 Identifikasi Object Baru & Tipe Kelas #3 Request Peminjaman Buku

TABEL OBJECT PERANCANGAN

No	Nama Object Baru	Jenis / Tipe Kelas
1	Page Peminjaman Buku	Boundary
2	Page Peminjaman Buku berhasil	Boundary
3	Page Peminjaman Buku gagal	Boundary
4	Button Continue	Controller
5	Button Back	Controller
6	Database User	Entity

*Tipe kelas seperti Boundary(Interface), Entity(Database), Controller

3.1.3.4 Sequence Diagram #3 Request Peminjaman Buku



3.1.4 Use Case #4 Status Peminjaman Buku

3.1.4.1 Use Case Scenario #4 Status Peminjaman Buku

Skenario Use Case #4 :

- Pre-Condition : Pengguna telah berhasil login dan memiliki buku yang sedang dipinjam atau telah mengajukan peminjaman.
- Use Case Description : Use Case ini digunakan oleh pengguna untuk mengecek status terkini dari buku-buku yang sedang dipinjam atau dalam proses peminjaman.
 - Primary Flow

No	Alur Utama
1	Aktor : Memilih opsi 'Status Peminjaman' dari menu utama atau dashboard.
2	Sistem : Menampilkan daftar semua peminjaman buku yang dilakukan oleh pengguna.
3	Aktor : Memilih peminjaman tertentu untuk melihat detail lebih lanjut.
4	Sistem : Menampilkan detail status peminjaman seperti tanggal peminjaman, tanggal pengembalian, status persetujuan, dan denda jika ada.

b. Alternative Flow

No	Alur Alternatif
1	Jika sistem gagal memuat data. Maka sistem menampilkan pesan error dan menyarankan pengguna untuk mencoba kembali atau menghubungi administrator sistem.
2	Jika tidak ada buku yang dipinjam. Maka sistem menampilkan pesan yang memberitahukan bahwa pengguna tidak memiliki buku yang sedang dipinjam atau dalam proses peminjaman.
3	Jika buku terlambat dikembalikan. Maka sistem menampilkan status terlambat beserta jumlah denda yang harus dibayar pengguna.

- iii. Post-Condition : Pengguna mendapatkan informasi detail tentang status peminjaman bukunya.

3.1.4.2 UI Design dan Deskripsi Objek UI #4 Status Peminjaman Buku



Tabel Deskripsi Objek UI

ID. LAYA R	NAMA LAYAR	DESKRIPSI
P001	Status Peminjaman Buku	Halaman status peminjaman buku.

UNTUK MASING – MASING ANTAR MUKA / PAGE dibuatkan spesifikasi detil
Antarmuka XXX: {diisi dengan no. layar atau no gambar rancangan antarmuka}

Page Status Peminjaman Buku

Id_Objek	LABEL*	Keterangan**
Button1	Menampilkan detail status peminjaman buku	Jika diklik user akan melihat detail status peminjam buku mereka.
Button2	Menyembunyikan detail status peminjaman buku	Jika diklik user akan menyembunyikan detail status peminjam buku mereka.
Button3	Back	Kembali ke beranda Anggota.

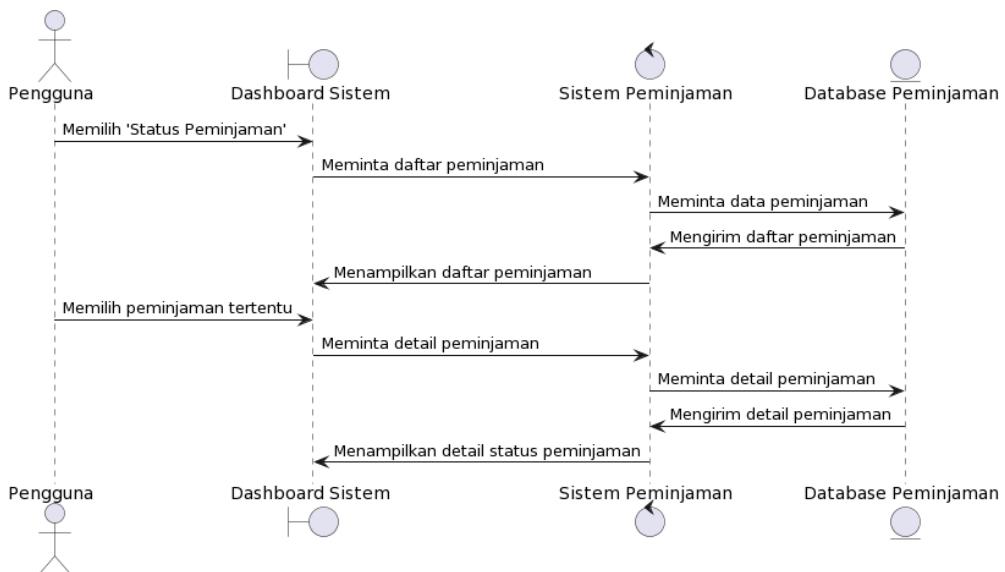
3.1.4.3 Identifikasi Object Baru & Tipe Kelas #4 Status Peminjaman Buku:

TABEL OBJECT PERANCANGAN

No	Nama Object Baru	Jenis / Tipe Kelas
1	Page Status Peminjaman Buku	Boundary
2	Button Up	Controller
3	Button Down	Controller
4	Button Back	Controller
5	Database User	Entity

*Tipe kelas seperti Boundary(Interface), Entity(Database), Controller

3.1.4.4 Sequence Diagram #4 Status Peminjaman Buku



3.1.5 Use Case #5 Pengembalian Buku

3.1.5.1 Use Case Scenario #5 Pengembalian Buku

Skenario Use Case #5 :

- Pre-Condition : Pengguna telah berhasil login dan memiliki buku yang sedang dipinjam.
- Use Case Description : Use Case ini digunakan oleh pengguna untuk melakukan proses pengembalian buku yang telah dipinjam melalui sistem perpustakaan berbasis web.
 - Primary Flow

No	Alur Utama
1	Aktor : Memilih opsi 'Pengembalian Buku' dari menu utama.
2	Sistem : Menampilkan daftar buku yang sedang dipinjam oleh pengguna.
3	Aktor : Memilih buku yang ingin dikembalikan.
4	Sistem : Menampilkan form pengembalian buku dengan detail peminjaman.
5	Aktor : Memverifikasi informasi pada form pengembalian dan mengkonfirmasi pengembalian.
6	Sistem : Memproses pengembalian dan mengupdate status buku.

b. Alternative Flow

No	Alur Alternatif
1	Jika buku rusak atau hilang. Maka pengguna menyatakan kondisi buku pada form pengembalian. Lalu, sistem mengevaluasi kondisi buku dan menentukan denda atau pengganti yang sesuai.
2	Jika sistem gagal memproses pengembalian. Maka sistem menampilkan pesan error dan menyuruh pengguna untuk mencoba lagi atau menghubungi pustakawan.
3	Jika terdapat denda keterlambatan. Maka sistem menginformasikan jumlah denda dan memberikan instruksi pembayaran.

- iii. Post-Condition : Buku yang dipinjam telah dikembalikan ke perpustakaan dan sistem telah memperbarui status peminjaman.

3.1.5.2 UI Design dan Deskripsi Objek UI #5 Pengembalian Buku





Tabel Deskripsi Objek UI

ID. LAYA R	NAMA LAYAR	DESKRIPSI
P001	<i>Pengembalian Buku</i>	<i>Menu pengembalian buku.</i>
P002	<i>Pengembalian Buku berhasil</i>	<i>Pengembalian Buku berhasil</i>
P003	<i>Alternatif Pengembalian Buku gagal</i>	<i>Pengembalian Buku gagal</i>

Page Pengembalian Buku

Id_Objek	LABEL*	Keterangan**
Teks1	Text Box	Admin mengisi username anggota.
Teks2	Text Box	Admin mengisi id buku.
Button1	Submit	Jika diklik data user akan diproses.
Button2	Back	Jika diklik ke beranda Admin.

Page Pengembalian Buku berhasil

Id_Objek	LABEL*	Keterangan**
Teks1	Text Box	Admin mengisi username anggota.
Teks2	Text Box	Admin mengisi id buku.
Button1	Submit	Jika diklik data user akan diproses, dan jika valid akan menampilkan detail data pengembalian buku.
Teks3	Text Box	Menampilkan detail data pengembalian buku.
Button2	Back	Jika diklik ke beranda Admin.

Page Pengembalian Buku gagal

Id_Objek	LABEL*	Keterangan**
Teks1	Text Box	Admin mengisi username anggota.
Teks2	Text Box	Admin mengisi id buku.
Button1	Submit	Jika diklik data user akan diproses, dan jika tidak valid akan kembali halaman pengembalian buku.
Button2	Back	Jika diklik ke beranda Admin.

3.1.5.3 Identifikasi Object Baru & Tipe Kelas #5 Pengembalian Buku

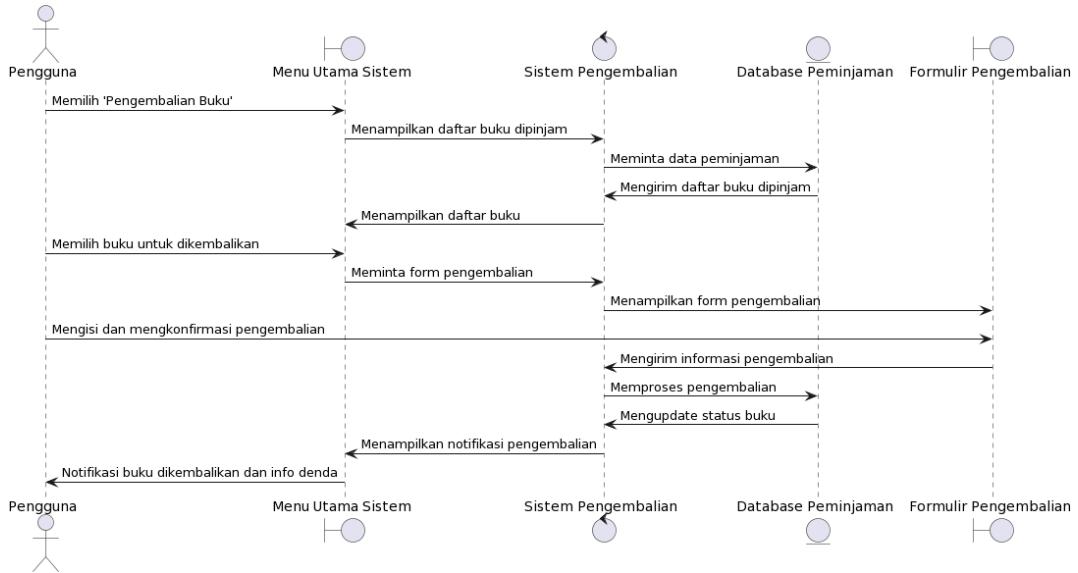
TABEL OBJECT PERANCANGAN

No	Nama Object Baru	Jenis / Tipe Kelas
1	Page Pengembalian Buku	Boundary
2	Page Pengembalian Buku berhasil	Boundary
3	Page Pengembalian Buku gagal	Boundary
4	Button submit	Controller
5	Button back	Controller
6	Database user	Entity

*Tipe kelas seperti Boundary(Interface), Entity(Database), Controller

3.1.5.4 Sequence Diagram #5 Pengembalian Buku

Prodi S1 Informatika Tel-U	DPPL-XXX	Halaman 32 dari 70
Template dokumen ini dan informasi yang dimilikinya adalah milik Prodi S1 Informatika Tel-U dan bersifat rahasia. Dilarang me-reproduksi dokumen ini tanpa diketahui oleh Prodi S1 Informatika Tel-U.		



3.1.6 Use Case #6 History Peminjaman Buku

3.1.6.1 Use Case Scenario #6 History Peminjaman Buku

Skenario Use Case #6 :

- Pre-Condition : Pengguna telah berhasil login ke dalam sistem.
- Use Case Description : Use Case ini memungkinkan pengguna untuk melihat riwayat lengkap dari semua buku yang telah dipinjam dari perpustakaan melalui sistem.
 - Primary Flow

No	Alur Utama
1	Aktor : Memilih opsi 'History Peminjaman' dari menu utama atau dashboard
2	Sistem : Memilih opsi 'History Peminjaman' dari menu utama atau dashboard.
3	Aktor : Jika diperlukan, pengguna dapat memilih transaksi tertentu untuk melihat detail lebih lanjut.
4	Sistem : Menampilkan detail lengkap dari transaksi peminjaman yang dipilih, termasuk riwayat denda dan catatan lainnya.

- Alternative Flow

No	Alur Alternatif
1	Jika tidak ada history yang tersedia. Maka sistem menampilkan pesan yang menginformasikan bahwa pengguna belum pernah melakukan

	peminjaman buku.
2	Jika terjadi kesalahan teknis saat memuat history. Maka sistem menampilkan pesan error dan menyarankan pengguna untuk mencoba lagi atau menghubungi dukungan teknis.

- iii. Post-Condition : Pengguna dapat meninjau semua transaksi peminjaman buku yang telah dilakukan.

3.1.6.2 UI Design dan Deskripsi Objek UI #6 History Peminjaman Buku



Tabel Deskripsi Objek UI

ID. LAYA R	NAMA LAYAR	DESKRIPSI
P001	Riwayat Peminjaman Buku	Halaman riwayat peminjaman buku.

Page Riwayat Peminjaman Buku

Id_Objek	LABEL*	Keterangan**
Teks1	Menampilkan detail data riwayat peminjaman buku	User akan melihat detail data riwayat peminjaman buku

Prodi S1 Informatika Tel-U	DPPL-XXX	Halaman 34 dari 70
Template dokumen ini dan informasi yang dimilikinya adalah milik Prodi S1 Informatika Tel-U dan bersifat rahasia. Dilarang me-reproduksi dokumen ini tanpa diketahui oleh Prodi S1 Informatika Tel-U.		

Id_Objek	LABEL*	Keterangan**
Button1	Back	Kembali ke beranda Anggota.

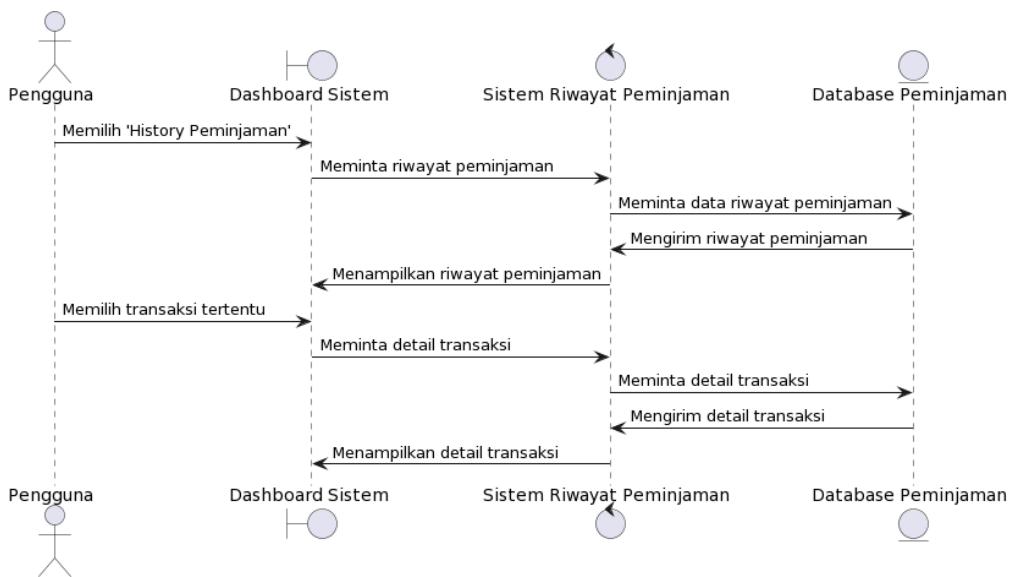
3.1.6.3 Identifikasi Object Baru & Tipe Kelas #6 History Peminjaman Buku

TABEL OBJECT PERANCANGAN

No	Nama Object Baru	Jenis / Tipe Kelas
1	Page History Peminjaman Buku	Boundary
2	Button Back	Controller

*Tipe kelas seperti Boundary(Interface), Entity(Database), Controller

3.1.6.4 Sequence Diagram #6 History Peminjaman Buku



3.1.7 Use Case #7 Menyetujui atau Menolak Peminjaman Buku

3.1.7.1 Use Case Scenario #7 Menyetujui atau Menolak Peminjaman Buku

Skenario Use Case #7 :

- Pre-Condition : Pustakawan telah berhasil login ke dalam sistem dan ada permintaan peminjaman buku yang tertunda.
- Use Case Description : Use Case ini digunakan oleh pustakawan untuk menyetujui atau menolak permintaan peminjaman buku yang diajukan oleh pengguna.
 - Primary Flow

No	Alur Utama
1	Aktor : Memilih opsi 'History Peminjaman' dari menu utama atau

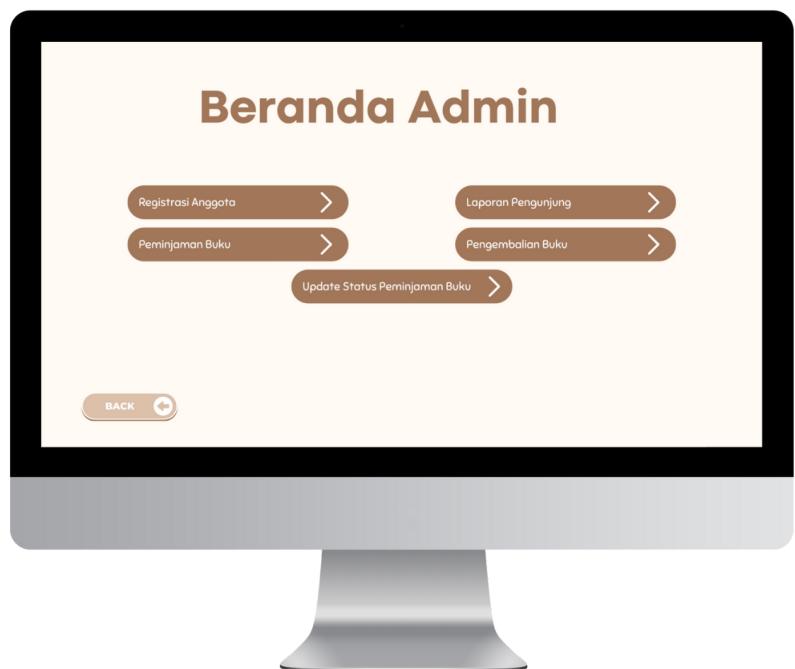
	dashboard
2	Sistem : Memilih opsi 'History Peminjaman' dari menu utama atau dashboard.
3	Aktor : Jika diperlukan, pengguna dapat memilih transaksi tertentu untuk melihat detail lebih lanjut.
4	Sistem : Menampilkan detail lengkap dari transaksi peminjaman yang dipilih, termasuk riwayat denda dan catatan lainnya.

b. Alternative Flow

No	Alur Alternatif
1	Jika buku tidak tersedia, maka pustakawan menolak permintaan dan sistem memberikan opsi kepada pengguna untuk dimasukkan ke daftar tunggu.
2	Jika informasi peminjam tidak lengkap atau tidak valid, maka pustakawan meminta pengguna melengkapi informasi yang diperlukan atau memverifikasi keabsahan data mereka.
3	Jika sistem mengalami kesalahan, maka sistem menampilkan pesan error dan pustakawan mencoba untuk memproses permintaan kembali atau menggunakan prosedur manual.

- iii. Post-Condition : Pengguna dapat meninjau semua transaksi peminjaman buku yang telah dilakukan.

3.1.7.2 UI Design dan Deskripsi Objek UI #7 Menyetujui atau Menolak Peminjaman Buku





Tabel Deskripsi Objek UI

ID. LAYA R	NAMA LAYAR	DESKRIPSI
P001	Konfirmasi Peminjaman Buku	Halaman konfirmasi peminjaman buku.
P002	Konfirmasi Peminjaman Buku berhasil	Jika permintaan peminjaman buku telah disetujui, maka Admin akan kembali ke beranda Admin.
P003	Konfirmasi Peminjaman Buku gagal	Jika permintaan peminjaman buku tidak disetujui, maka Admin akan kembali ke halaman konfirmasi peminjaman buku.

Page Konfirmasi Peminjaman Buku

Id_Objek	LABEL*	Keterangan**
Teks1	Text Box	Admin akan mengisi kata sandi
Button1	Submit	Jika diklik data user akan diproses.
Button2	Back	Kembali ke beranda Admin

Page Konfirmasi Peminjaman Buku berhasil

Id_Objek	LABEL*	Keterangan**
Teks1	Text Box	Admin akan mengisi kata sandi

Id_Objek	LABEL*	Keterangan**
<i>Button1</i>	<i>Submit</i>	<i>Jika diklik data user akan diproses, dan jika valid akan kembali ke beranda Admin.</i>
<i>Button2</i>	<i>Back</i>	<i>Kembali ke beranda Admin.</i>

Page Konfirmasi Peminjaman Buku gagal

Id_Objek	LABEL*	Keterangan**
<i>Teks1</i>	<i>Text Box</i>	<i>Admin akan mengisi kata sandi</i>
<i>Button1</i>	<i>Submit</i>	<i>Jika diklik data user akan diproses, dan jika tidak valid akan tetap berada di halaman konfirmasi peminjaman buku.</i>
<i>Button2</i>	<i>Back</i>	<i>Kembali ke beranda Admin.</i>

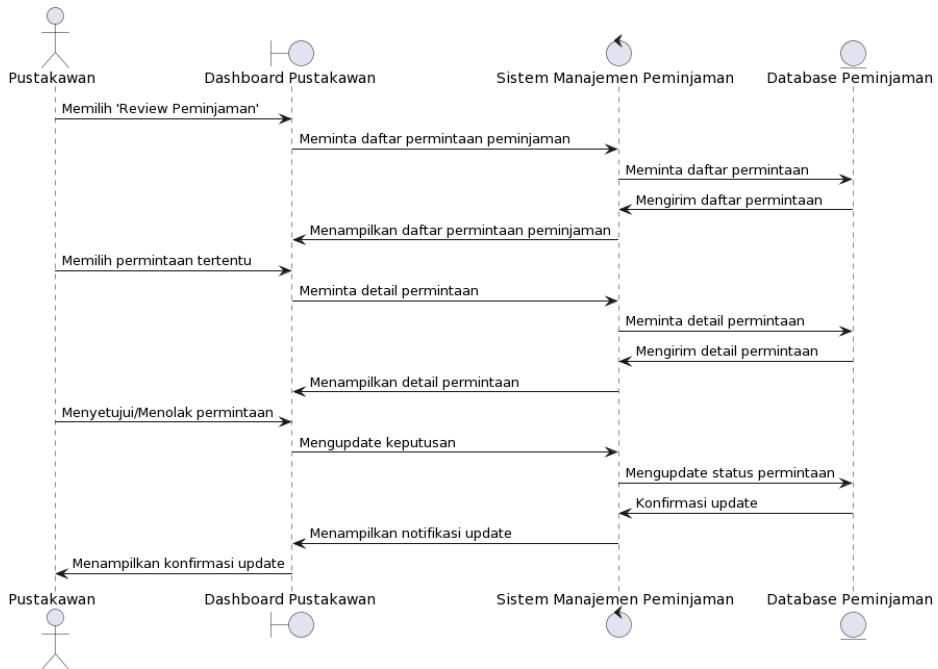
3.1.7.3 Identifikasi Object Baru & Tipe Kelas #7 Menyetujui atau Menolak Peminjaman Buku

TABEL OBJECT PERANCANGAN

No	Nama Object Baru	Jenis / Tipe Kelas
1	Page Konfirmasi Peminjaman Buku	Boundary
2	Page Konfirmasi Peminjaman Buku berhasil	Boundary
3	Page Konfirmasi Peminjaman Buku gagal	Boundary
4	Button Submit	Controller
5	Button Back	Controller
6	Database User	Entity

*Tipe kelas seperti Boundary(Interface), Entity(Database), Controller

3.1.7.4 Sequence Diagram #7 Menyetujui atau Menolak Peminjaman Buku



3.1.8 Use Case #8 Update Status Peminjaman Buku

3.1.8.1 Use Case Scenario #8 Update Status Peminjaman Buku

Skenario Use Case #8 :

- Pre-Condition : Pustakawan telah berhasil login ke dalam sistem.
- Use Case Description : Use Case ini digunakan oleh pustakawan untuk memperbarui status buku yang sedang dipinjam, seperti memperbarui tanggal pengembalian, menandai buku sebagai dikembalikan, atau memperbarui kondisi buku setelah pemeriksaan.
 - Primary Flow

No	Alur Utama
1	Aktor : Memilih opsi 'Update Status Peminjaman' dari dashboard pustakawan.
2	Sistem : Menampilkan daftar peminjaman buku yang aktif.
3	Aktor : Memilih peminjaman buku tertentu untuk diperbarui.
4	Sistem : Menampilkan form untuk memperbarui status peminjaman buku.
5	Aktor : Memasukkan pembaruan status, seperti perubahan tanggal pengembalian, kondisi buku, atau catatan lain yang relevan.

6	Sistem : Memverifikasi dan mengkonfirmasi perubahan yang dimasukkan
7	Sistem : Mengupdate informasi dalam sistem dan memberikan konfirmasi sukses kepada pustakawan.

b. Alternative Flow

No	Alur Alternatif
1	Jika buku tidak ditemukan dalam sistem, maka sistem menampilkan pesan error dan pustakawan memeriksa data fisik buku atau memasukkan informasi buku secara manual.
2	Jika terjadi kesalahan input, maka sistem menolak perubahan karena kesalahan format atau data yang tidak valid dan meminta pustakawan untuk memasukkan ulang informasi.
3	Jika sistem mengalami gangguan, maka sistem menampilkan pesan error dan pustakawan mencoba kembali nanti atau menghubungi dukungan IT.

- iii. Post-Condition : Status peminjaman buku telah diperbarui dalam sistem sesuai dengan kondisi aktual.

3.1.8.2 UI Design dan Deskripsi Objek UI #8 Update Status Peminjaman Buku





Tabel Deskripsi Objek UI

ID. LAYA R	NAMA LAYAR	DESKRIPSI
P001	Update Status Peminjaman Buku	Halaman riwayat peminjaman buku.
P002	Update Status Peminjaman Buku berhasil	Menampilkan detail data status peminjaman buku yang ter-update pada halaman update status peminjaman buku ketika update peminjaman buku berhasil.
P003	Update Status Peminjaman Buku gagal	Kembali ke halam update status peminjaman buku ketika ketika update status peminjaman buku gagal.

Page Update Status Peminjaman Buku

Id_Objek	LABEL*	Keterangan**
Teks1	Text box	Admin mengisi id buku.
Teks2	Text box	Admin mengisi tanggal peminjaman.
Teks3	Text box	Admin mengisi tanggal pengembalian.
Teks4	Text box	Admin mengisi kondisi buku.
Button1	Submit	Jika diklik data update status peminjaman buku diproses.
Button2	Back	Kembali ke beranda Admin.

Page Update Status Peminjaman Buku berhasil

Id_Objek	LABEL*	Keterangan**
Teks1	<i>Text box</i>	<i>Admin mengisi id buku.</i>
Teks2	<i>Text box</i>	<i>Admin mengisi tanggal peminjaman.</i>
Teks3	<i>Text box</i>	<i>Admin mengisi tanggal pengembalian.</i>
Teks4	<i>Text box</i>	<i>Admin mengisi kondisi buku.</i>
Button1	<i>Submit</i>	<i>Jika diklik data update status peminjaman buku diproses, dan jika valid maka Admin akan melihat data update status peminjaman buku yang terbaru.</i>
Teks5	<i>Menampilkan detail data update status peminjaman buku.</i>	<i>Admin melihat data update status peminjaman buku yang terbaru.</i>
Button2	<i>Back</i>	<i>Kembali ke beranda Admin.</i>

Page Update Status Peminjaman Buku gagal

Id_Objek	LABEL*	Keterangan**
Teks1	<i>Text box</i>	<i>Admin mengisi id buku.</i>
Teks2	<i>Text box</i>	<i>Admin mengisi tanggal peminjaman.</i>
Teks3	<i>Text box</i>	<i>Admin mengisi tanggal pengembalian.</i>
Teks4	<i>Text box</i>	<i>Admin mengisi kondisi buku.</i>
Button1	<i>Submit</i>	<i>Jika diklik data update status peminjaman buku diproses, dan jika tidak valid maka Admin akan didirect kembali ke halaman update status peminjaman buku yang terbaru.</i>
Button2	<i>Back</i>	<i>Kembali ke beranda Admin.</i>

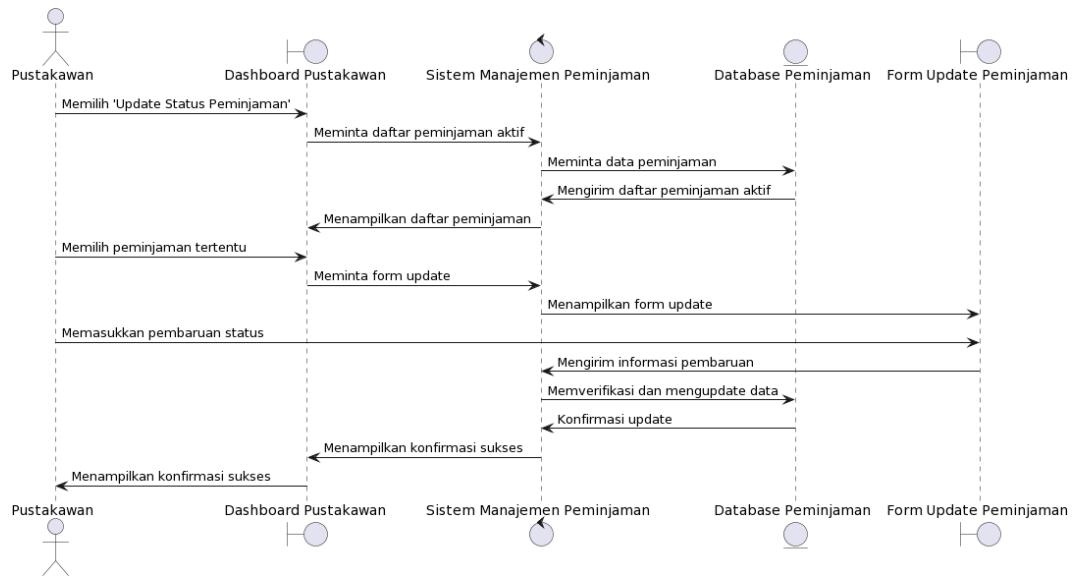
3.1.8.3 Identifikasi Object Baru & Tipe Kelas #8 Update Status Peminjaman Buku

TABEL OBJECT PERANCANGAN

No	Nama Object Baru	Jenis / Tipe Kelas
1	Page Update Status Peminjaman Buku	Boundary
2	Page Update Status Peminjaman Buku berhasil	Boundary
3	Page Update Status Peminjaman Buku gagal	Boundary
4	Button Submit	Controller
5	Button Back	Controller
6	Database User	Entity

*Tipe kelas seperti Boundary(Interface), Entity(Database), Controller

3.1.8.4 Sequence Diagram #8 Update Status Peminjaman Buku



3.1.9 Use Case #9 Menyetujui Pengembalian Buku

3.1.9.1 Use Case Scenario #9 Menyetujui Pengembalian Buku

Skenario Use Case #9 :

- Pre-Condition : Pustakawan telah berhasil login ke dalam sistem dan buku yang dipinjam telah fisik dikembalikan ke perpustakaan.
- Use Case Description : Use Case ini digunakan oleh pustakawan untuk memverifikasi dan menyetujui pengembalian buku yang telah dilakukan oleh pengguna.
 - Primary Flow

No	Alur Utama
1	Aktor : Memilih opsi 'Verifikasi Pengembalian Buku' dari dashboard pustakawan
2	Sistem : Menampilkan daftar buku yang dikembalikan dan menunggu verifikasi.
3	Aktor : Memilih buku yang dikembalikan untuk diverifikasi.
4	Sistem : Menampilkan form verifikasi pengembalian dengan detail peminjaman dan kondisi buku.
5	Aktor : Memeriksa buku secara fisik dan mengisi form verifikasi dengan status kondisi buku .
6	Sistem : Mengkonfirmasi kebenaran pengembalian dan kondisi buku.
7	Sistem : Mengupdate status peminjaman buku di sistem sebagai 'dikembalikan' dan mengakhiri peminjaman .

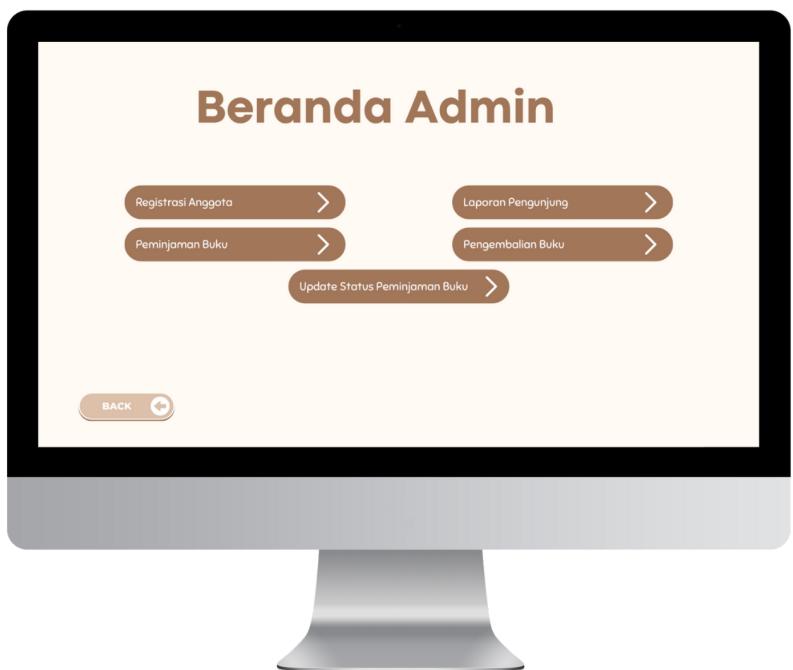
b. Alternative Flow

No	Alur Alternatif
1	Jika buku rusak atau hilang, maka pustakawan mencatat kerusakan atau kehilangan dalam sistem dan menetapkan denda atau pengganti sesuai kebijakan perpustakaan.
2	Jika buku dikembalikan terlambat, maka sistem menghitung denda berdasarkan kebijakan keterlambatan dan pustakawan menginformasikan kepada pengguna mengenai denda tersebut.
3	Jika sistem gagal memperbarui status, maka sistem menampilkan pesan error dan pustakawan mencoba memperbarui kembali atau menghubungi dukungan teknis.

- iii. Post-Condition : Buku yang dikembalikan telah diperiksa, statusnya diperbarui di sistem, dan proses peminjaman ditutup.

3.1.9.2 UI Design dan Deskripsi Objek UI #9 Menyetujui Pengembalian Buku





Tabel Deskripsi Objek UI

ID. LAYA R	NAMA LAYAR	DESKRIPSI
P001	Konfirmasi Pengembalian Buku	Halaman konfirmasi pengembalian buku.
P002	Konfirmasi Pengembalian Buku berhasil	Jika pengembalian buku berhasil, maka Admin akan kembali ke beranda Admin.
P003	Konfirmasi Pengembalian Buku gagal	Jika pengembalian buku gagal, maka Admin akan kembali ke halaman konfirmasi pengembalian buku.

Page Konfirmasi Pengembalian Buku

Id_Objek	LABEL*	Keterangan**
<i>Teks1</i>	<i>Text box</i>	<i>Admin mengisi kata sandi.</i>
<i>Button1</i>	<i>Submit</i>	<i>Jika diklik kata sandi diproses.</i>
<i>Button2</i>	<i>Back</i>	<i>Kembali ke beranda Admin.</i>

Page Konfirmasi Pengembalian Buku berhasil

Id_Objek	LABEL*	Keterangan**
<i>Teks1</i>	<i>Text box</i>	<i>Admin mengisi kata sandi.</i>
<i>Button1</i>	<i>Submit</i>	<i>Jika diklik kata sandi diproses, dan jika valid maka konfirmasi pengembalian buku telah disetujui dan Admin kembali ke beranda Admin.</i>
<i>Button2</i>	<i>Back</i>	<i>Kembali ke beranda Admin.</i>

Page Konfirmasi Pengembalian Buku gagal

Id_Objek	LABEL*	Keterangan**
<i>Teks1</i>	<i>Text box</i>	<i>Admin mengisi kata sandi.</i>
<i>Button1</i>	<i>Submit</i>	<i>Jika diklik kata sandi diproses, dan jika tidak valid maka konfirmasi pengembalian buku tidak disetujui dan Admin kembali ke halaman konfirmasi pengembalian buku.</i>
<i>Button2</i>	<i>Back</i>	<i>Kembali ke beranda Admin.</i>

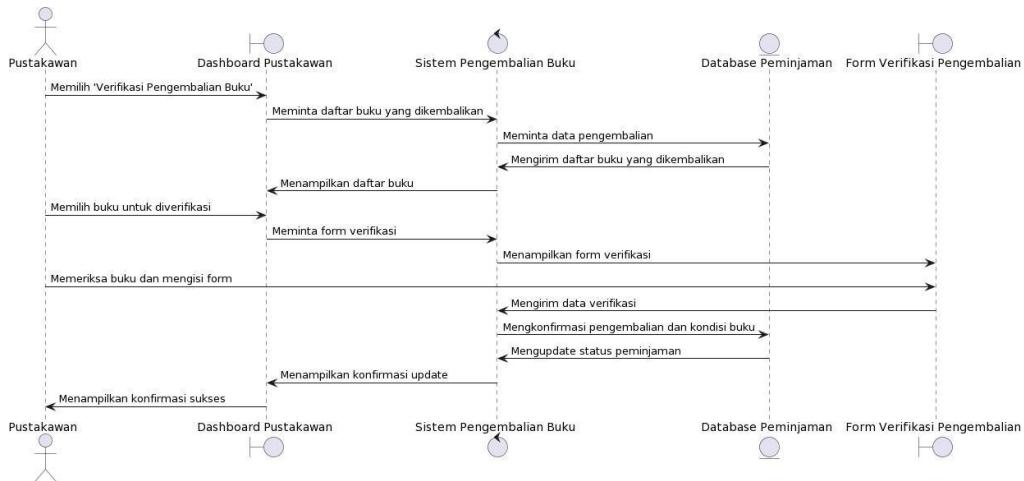
3.1.9.3 Identifikasi Object Baru & Tipe Kelas #9 Menyetujui Pengembalian Buku

TABEL OBJECT PERANCANGAN

No	Nama Object Baru	Jenis / Tipe Kelas
1	Page Konfirmasi Pengembalian Buku	Boundary
2	Page Konfirmasi Pengembalian Buku berhasil	Boundary
3	Page Konfirmasi Pengembalian Buku gagal	Boundary
4	Button Submit	Controller
5	Button Back	Controller

*Tipe kelas seperti Boundary(Interface), Entity(Database), Controller

3.1.9.4 Sequence Diagram #9 Menyetujui Pengembalian Buku



3.1.10 Use Case #10 Mencari Buku di Perpustakaan

3.1.10.1 Use Case Scenario #10 Mencari Buku di Perpustakaan

Skenario Use Case #10 :

- Pre-Condition : Pengguna telah berhasil login ke dalam sistem atau menggunakan interface publik sistem pencarian.
- Use Case Description : Use Case ini memungkinkan pengguna untuk mencari dan menemukan buku yang tersedia dalam koleksi perpustakaan menggunakan sistem online.
 - Primary Flow

No	Alur Utama
1	Aktor : Memasuki kata kunci pencarian dalam bar pencarian atau menggunakan fitur pencarian lanjutan.
2	Sistem : Menerima kata kunci dan melakukan pencarian dalam basis data perpustakaan.
3	Sistem : Menampilkan daftar buku yang sesuai dengan kriteria pencarian, termasuk judul, penulis, dan status ketersediaan.
4	Aktor : Jika diperlukan, memilih judul buku untuk mendapatkan informasi lebih detail.
5	Sistem : Menampilkan informasi detail buku seperti lokasi rak, ringkasan, dan informasi peminjaman..

- Alternative Flow

No	Alur Alternatif
1	Jika buku tidak ditemukan, maka sistem menampilkan pesan bahwa tidak ada hasil yang cocok dan menawarkan saran untuk memperluas kriteria pencarian.
2	Jika terjadi kesalahan teknis saat pencarian, maka sistem menampilkan pesan error dan menyuruh pengguna untuk mencoba kembali atau menghubungi bantuan.
3	Jika buku yang dicari sedang dipinjam, maka sistem menampilkan status peminjaman dan memberikan opsi untuk pengguna untuk menambahkan diri ke daftar tunggu atau reservasi.

- iii. Post-Condition : Pengguna mendapatkan daftar buku yang sesuai dengan kriteria pencarian mereka.

3.1.10.2 UI Design dan Deskripsi Objek UI #10 Mencari Buku di Perpustakaan





Tabel Deskripsi Objek UI

ID. LAYA R	NAMA LAYAR	DESKRIPSI
P001	Mencari Buku di Perpustakaan	Halaman mencari buku di perpustakaan..
P002	Mencari Buku di Perpustakaan berhasil	Jika mencari buku di perpustakaan berhasil, maka user akan melihat detail dari data buku yang dicari.
P003	Mencari Buku di Perpustakaan gagal	Jika mencari buku di perpustakaan gagal, maka user akan tetap berada di halaman mencari buku di perpustakaan..

Page Mencari Buku di Perpustakaan

Id_Objek	LABEL*	Keterangan**
Teks1	Text box	User akan mengisi judul buku yang dicari.
Button1	Search	Jika diklik judul buku diproses.

Page Mencari Buku di Perpustakaan berhasil

Id_Objek	LABEL*	Keterangan**
<i>Teks1</i>	<i>Text box</i>	<i>User akan mengisi judul buku yang dicari.</i>
<i>Button1</i>	<i>Search</i>	<i>Jika diklik judul buku diproses, dan jika valid maka pencarian buku di perpustakaan berhasil dan user akan melihat detail dari data buku yang dicari.</i>

Page Mencari Buku di Perpustakaan gagal

Id_Objek	LABEL*	Keterangan**
<i>Teks1</i>	<i>Text box</i>	<i>User akan mengisi judul buku yang dicari.</i>
<i>Button1</i>	<i>Search</i>	<i>Jika diklik judul buku diproses, dan jika tidak valid maka pencarian buku di perpustakaan gagal dan user akan tetap berada di halaman mencari buku di perpustakaan, dan menampilkan notifikasi bahwa buku tidak ditemukan.</i>

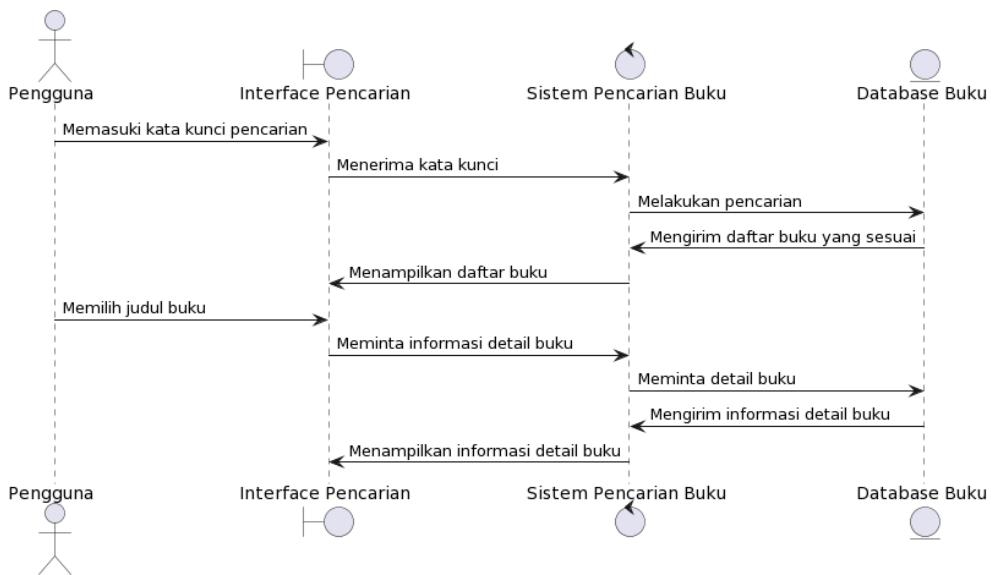
3.1.10.3 Identifikasi Object Baru & Tipe Kelas #10 Mencari Buku di Perpustakaan

TABEL OBJECT PERANCANGAN

No	Nama Object Baru	Jenis / Tipe Kelas
1	Page Mencari Buku di Perpustakaan	Boundary
2	Page Mencari Buku di Perpustakaan berhasil	Boundary
3	Page Mencari Buku di Perpustakaan gagal	Boundary
4	Button Search	Controller
5	Database Buku	Entity

*Tipe kelas seperti Boundary(Interface), Entity(Database), Controller

3.1.10.4 Sequence Diagram #10 Mencari Buku di Perpustakaan



3.1.11 Use Case #11 Melihat Laporan Pengunjung

3.1.11.1 Use Case Scenario #11 Melihat Laporan Pengunjung

Skenario Use Case #10 :

- Pre-Condition : Pustakawan telah berhasil login ke dalam sistem.
- Use Case Description : Use Case ini digunakan oleh pustakawan untuk meninjau dan menganalisis data pengunjung yang telah masuk ke perpustakaan dalam periode tertentu.
 - Primary Flow

No	Alur Utama
1	Aktor : Memilih opsi 'Laporan Pengunjung' dari dashboard pustakawan.
2	Sistem : Menampilkan pilihan untuk menentukan periode waktu laporan.
3	Aktor : Memasukkan rentang tanggal dan kriteria lainnya untuk laporan.
4	Sistem : Menghasilkan dan menampilkan laporan berdasarkan data pengunjung pada periode yang dipilih.
5	Aktor : Menganalisis data laporan yang disajikan dalam bentuk tabel atau grafik.
6	Sistem : Menyediakan opsi untuk mengunduh laporan dalam format yang diinginkan (PDF, Excel, dll.).

b. Alternative Flow

No	Alur Alternatif
1	Jika tidak ada data yang tersedia, maka sistem menampilkan pesan bahwa tidak ada catatan pengunjung untuk periode yang dipilih.
2	Jika terjadi kesalahan sistem saat menghasilkan laporan, maka sistem menampilkan pesan error dan pustakawan mencoba lagi atau menghubungi dukungan teknis.
3	Jika pustakawan membutuhkan data lebih detail, maka sistem menyediakan opsi untuk menyesuaikan kriteria laporan dan menghasilkan laporan yang lebih spesifik.

- iii. Post-Condition : Pustakawan mendapatkan laporan pengunjung yang dapat digunakan untuk analisis statistik atau keperluan manajemen perpustakaan.

3.1.11.2 UI Design dan Deskripsi Objek UI #11 Melihat Laporan Pengunjung



Tabel Deskripsi Objek UI

ID. LAYAR	NAMA LAYAR	DESKRIPSI
P001	<i>Melihat Laporan Pengunjung</i>	<i>Halaman melihat data laporan pengunjung.</i>

Page Melihat Laporan Pengunjung

Id_Objek	LABEL*	Keterangan**
Teks1	Data laporan pengunjung pada bulan pertama	Menampilkan data laporan jumlah pengunjung setiap minggunya pada setiap bulan pertama.
Teks2	Data laporan pengunjung pada bulan kedua	Menampilkan data laporan jumlah pengunjung setiap minggunya pada setiap bulan kedua.
Teks3	Data laporan pengunjung pada bulan ketiga	Menampilkan data laporan jumlah pengunjung setiap minggunya pada setiap bulan ketiga.
Button1	Back	Kembali ke beranda Admin.

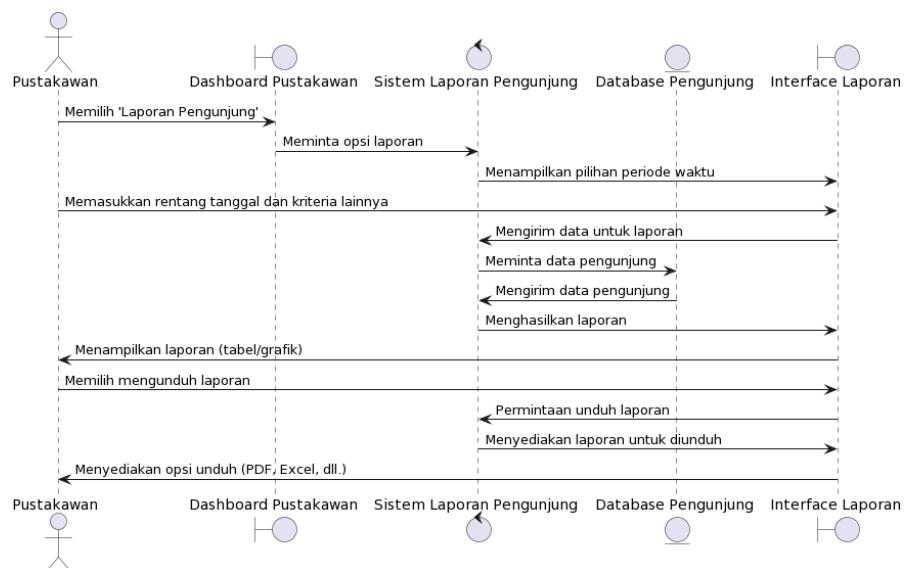
3.1.11.3 Identifikasi Object Baru & Tipe Kelas #11 Melihat Laporan Pengunjung

TABEL OBJECT PERANCANGAN

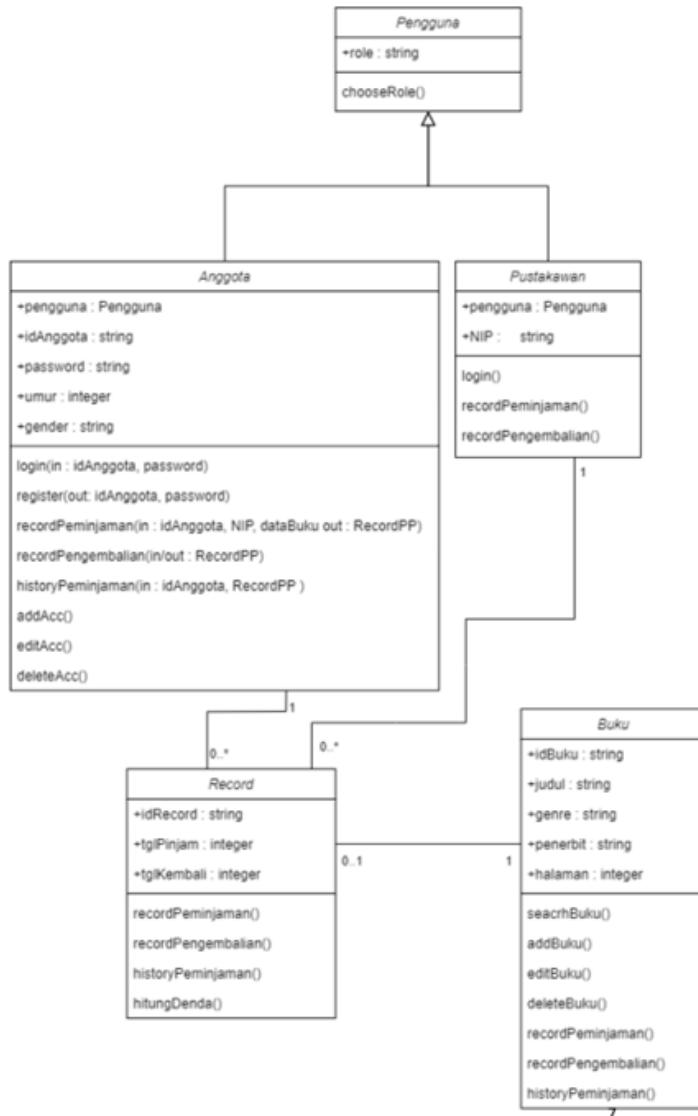
No	Nama Object Baru	Jenis / Tipe Kelas
1	Page Melihat Laporan Pengunjung	Boundary
2	Button Up	Controller
3	Button Down	Controller
4	Button Back	Controller
5	Database User	Entity

*Tipe kelas seperti Boundary(Interface), Entity(Database), Controller

3.1.11.4 Sequence Diagram #11 Melihat Laporan Pengunjung



3.2 Diagram Kelas Keseluruhan



3.3 Perancangan Detail Kelas

1. Pengguna (User)
 - Atribut:
 - o `role: string`
 - Behaviour:
 - o `chooseRole(): void`
2. Anggota (Member)

- Atribut:
 - o pengguna: Pengguna
 - o idAnggota: string
 - o umur: integer
 - o gender: string
- Behaviour:
 - o login(): void
 - o register(): void
 - o recordPeminjaman(): void
 - o recordPengembalian(): void
 - o historyPeminjaman(): void
 - o addAcc(): void
 - o editAcc(): void
 - o deleteAcc(): void

3. Pustakawan (Librarian)

- Atribut:
 - o pengguna: Pengguna
 - o NIP: string
- Behaviour:
 - o login(): void
 - o recordPeminjaman(): void
 - o recordPengembalian(): void
 - o Buku (Book)

4. Buku (Book)

- Atribut:
 - o idRecord: string
 - o tglPinjam: integer

- o tglKembali: integer

- Behaviour:

- o addBuku(): void

- o deleteBuku(): void

- o editBuku(): void

5. Record

- Atribut:

- o id_buku: string

- o judul: string

- o genre: string

- o penerbit: string

- o halaman: integer

- Behaviour:

- o recordPeminjaman(): void

- o recordPengembalian(): void

- o historyPeminjaman(): void

TABEL KELAS : Pengguna

ID Kelas	Nama Kelas Perancangan	Atribute (visibility)	Method / Operation
C001	Pengguna	<i>role: string</i>	<i>chooseRole()</i>

TABEL KELAS : Anggota (subclass dari Pengguna)

ID Kelas	Nama Kelas Perancangan	Atribute (visibility)	Method / Operation

<i>C002</i>	<i>Anggota (subclass dari Pengguna)</i>	<i>idAnggota: string</i>	<i>login(idAnggota, password)</i>
		<i>password: string</i>	<i>register(idAnggota, password)</i>
		<i>umur: integer</i>	<i>recordPeminjaman()</i>
		<i>rgender: string</i>	<i>recordPengembalian()</i>
			<i>historyPeminjaman()</i>
			<i>addAcc()</i>
			<i>editAcc()</i>
			<i>deleteAcc()</i>

TABEL KELAS : Pustakawan (subclass dari Pengguna)

ID Kelas	Nama Kelas Perancangan	Atribute (visibility)	Method / Operation
<i>C003</i>	<i>Pustakawan (subclass dari Pengguna)</i>	<i>NIP: string</i>	<i>login()</i>
			<i>recordPeminjaman()</i>
			<i>recordPengembalian()</i>

TABEL KELAS : Buku

ID Kelas	Nama Kelas Perancangan	Atribute (visibility)	Method / Operation
<i>C004</i>	<i>Buku</i>	<i>idBuku: string</i>	<i>addBuku()</i>
		<i>judul: string</i>	<i>editBuku()</i>
		<i>genre: string</i>	<i>deleteBuku()</i>
		<i>penerbit: string</i>	<i>recordPeminjaman()</i>
		<i>tahun: integer</i>	<i>recordPengembalian()</i>
		<i>halaman: integer</i>	<i>historyPeminjaman()</i>

TABEL KELAS : Record

ID Kelas	Nama Kelas Perancangan	Atribute (visibility)	Method / Operation
<i>C005</i>	<i>Record</i>	<i>idRecord: string</i>	<i>recordPeminjaman()</i>
		<i>tglPinjam: integer</i>	<i>recordPengembalian()</i>
		<i>tglKembali: integer</i>	<i>historyPeminjaman()</i>
			<i>hitungDenda()</i>

3.4 Perancangan Algoritma dan/atau Query

Algoritma #1

Contoh:

Nama Kelas : UserAuthentication

Nama Operasi : login

Algoritma :

1. Mulai
2. Input: username dan password dari pengguna
3. Jika (username dan password tidak kosong) maka:
 - a. Lakukan sanitasi input untuk mencegah SQL Injection
 - b. Verifikasi keberadaan username dalam database
 - i. Jika username tidak ditemukan, tampilkan pesan kesalahan "Username tidak ditemukan"
 - ii. Lanjutkan jika username ditemukan
 - c. Bandingkan password yang dimasukkan dengan password dalam database (yang sudah di-hash)
 - i. Jika password cocok, cek apakah akun sudah diverifikasi dan tidak diblokir
 - Jika akun belum diverifikasi atau diblokir, tampilkan pesan kesalahan yang sesuai
 - Jika akun diverifikasi dan tidak diblokir, lanjutkan ke langkah berikutnya
 - ii. Jika password tidak cocok, tampilkan pesan kesalahan "Password salah"
 - d. Generate session ID untuk pengguna
 - e. Set session ID dan status login pengguna pada sistem
 - f. Redirect pengguna ke dashboard
4. Jika (username atau password kosong) maka:
 - a. Tampilkan pesan kesalahan "Username/password tidak boleh kosong"
5. Selesai

Algoritma #2

Contoh:

Nama Kelas : UserRegistration

Nama Operasi : registerNewUser

Algoritma :

1. Mulai
2. Input: data registrasi (seperti nama, alamat email, password, konfirmasi password, dan informasi tambahan lainnya)
3. Jika (semua input data wajib diisi) maka:
 - a. Lakukan validasi pada setiap input
 - i. Periksa format alamat email
 - ii. Pastikan password dan konfirmasi password cocok
 - iii. Cek kekuatan password (minimal panjang, mengandung angka dan simbol, dll.)
 - iv. Validasi data tambahan sesuai aturan yang berlaku (misal: format nomor telepon)
 - b. Jika validasi gagal, tampilkan pesan kesalahan yang relevan

- c. Jika validasi berhasil, lanjutkan ke langkah berikutnya
- 4. Periksa apakah alamat email sudah terdaftar dalam database
 - a. Jika email sudah terdaftar, tampilkan pesan "Alamat email sudah digunakan"
 - b. Jika email belum terdaftar, lanjutkan ke langkah berikutnya
- 5. Hash password menggunakan algoritma hash yang aman
- 6. Buat record baru dalam database dengan data yang telah divalidasi dan password yang sudah di-hash
- 7. Kirim email konfirmasi ke alamat email yang didaftarkan dengan link aktivasi akun
- 8. Tampilkan pesan bahwa registrasi berhasil dan instruksi untuk memeriksa email guna melakukan aktivasi akun
- 9. Selesai

Algoritma #3

Contoh:

Nama Kelas : BookLoanRequest

Nama Operasi : createLoanRequest

Algoritma :

- 1. Mulai
- 2. Input: ID buku yang ingin dipinjam, ID pengguna yang meminjam
- 3. Jika (pengguna sudah login dan memiliki hak peminjaman) maka:
 - a. Cari buku dalam database menggunakan ID buku
 - b. Jika (buku tidak ditemukan) maka:
 - i. Tampilkan pesan kesalahan "Buku tidak ditemukan"
 - ii. Hentikan proses
 - c. Jika (buku ditemukan) maka:
 - i. Periksa status buku, apakah tersedia untuk dipinjam
 - ii. Jika (buku tidak tersedia) maka:
 - Tampilkan pesan "Buku sedang dipinjam" atau "Buku tidak tersedia"
 - Tawarkan untuk menambahkan pengguna ke daftar tunggu jika ada
 - Hentikan proses
 - iii. Jika (buku tersedia) maka:
 - Buat record peminjaman baru dengan status 'pending'
 - Tentukan tanggal peminjaman dan perkiraan tanggal pengembalian
 - Tambahkan record peminjaman ke dalam database
 - Kirim notifikasi kepada pengguna bahwa request peminjaman berhasil dibuat
 - Opsi: Kirim notifikasi ke pustakawan untuk verifikasi dan persetujuan request
 - d. Jika (pengguna telah mencapai batas maksimal peminjaman) maka:
 - i. Tampilkan pesan kesalahan "Anda telah mencapai batas peminjaman"
- 4. Jika (pengguna belum login atau tidak memiliki hak peminjaman) maka:
 - a. Tampilkan pesan kesalahan "Silakan login untuk melanjutkan" atau "Anda tidak memiliki hak untuk meminjam buku"
- 5. Selesai

Algoritma #4

Contoh:

Nama Kelas : LoanStatusChecker

Nama Operasi : checkLoanStatus

Algoritma :

1. Mulai
2. Input: ID pengguna
3. Jika (pengguna sudah login) maka:
 - a. Mengakses database untuk mengambil semua record peminjaman yang terkait dengan ID pengguna
 - b. Jika (tidak ada record peminjaman) maka:
 - i. Tampilkan pesan "Tidak ada riwayat peminjaman"
 - c. Jika (ada record peminjaman) maka:
 - i. Untuk setiap record peminjaman, lakukan:
 - Ambil detail buku (seperti judul, penulis)
 - Periksa tanggal peminjaman dan tanggal pengembalian
 - Tentukan status peminjaman (sedang dipinjam, terlambat, dll.)
 - Jika buku terlambat dikembalikan, hitung denda berdasarkan kebijakan perpustakaan
 - ii. Tampilkan daftar peminjaman dan status masing-masing kepada pengguna
 - d. Tampilkan opsi untuk memperpanjang peminjaman atau mengajukan pengembalian jika memungkinkan
4. Jika (pengguna belum login) maka:
 - a. Tampilkan pesan kesalahan "Silakan login untuk melihat status peminjaman"
5. Selesai

Algoritma #5

Contoh:

Nama Kelas : BookReturnProcess

Nama Operasi : processBookReturn

Algoritma :

1. Mulai
2. Input: ID buku yang dikembalikan, ID pengguna yang mengembalikan
3. Jika (pengguna sudah login) maka:
 - a. Cari record peminjaman dalam database menggunakan ID buku dan ID pengguna
 - b. Jika (record peminjaman tidak ditemukan) maka:
 - i. Tampilkan pesan kesalahan "Record peminjaman tidak ditemukan"
 - c. Jika (record peminjaman ditemukan) maka:
 - i. Tentukan tanggal pengembalian aktual
 - ii. Bandingkan tanggal pengembalian aktual dengan tanggal pengembalian yang dijadwalkan
 - iii. Jika (tanggal pengembalian aktual melebihi tanggal yang dijadwalkan) maka:
 - Hitung denda berdasarkan kebijakan perpustakaan
 - Tambahkan denda ke record pengguna
 - Tampilkan pesan kepada pengguna tentang denda yang diakibatkan oleh keterlambatan
 - iv. Jika (tidak terlambat) maka:
 - Tampilkan pesan "Terima kasih telah mengembalikan buku tepat waktu"

- v. Update status buku dalam database menjadi 'tersedia'
- vi. Update record peminjaman buku untuk menandai bahwa buku telah dikembalikan
- d. Jika (pengguna telah mencapai atau melebihi batas peminjaman) maka:
 - i. Update record peminjaman untuk mengurangi jumlah buku yang dipinjam oleh pengguna
- 4. Jika (pengguna belum login) maka:
 - a. Tampilkan pesan kesalahan "Silakan login untuk mengembalikan buku"
- 5. Selesai

Algoritma #6

Contoh:

Nama Kelas : BorrowingHistoryViewer

Nama Operasi : displayBorrowingHistory

Algoritma :

- 1. Mulai
- 2. Input: ID pengguna
- 3. Jika (pengguna sudah login) maka:
 - a. Mengakses database untuk mengambil semua record peminjaman yang terkait dengan ID pengguna
 - b. Jika (tidak ada record peminjaman) maka:
 - i. Tampilkan pesan "Tidak ada history peminjaman"
 - c. Jika (ada record peminjaman) maka:
 - i. Untuk setiap record peminjaman, lakukan:
 - Ambil detail buku (seperti judul, penulis)
 - Periksa tanggal peminjaman dan tanggal pengembalian
 - Periksa status peminjaman (sedang dipinjam, sudah dikembalikan, denda jika ada)
 - Tambahkan informasi ini ke dalam list history
 - ii. Tampilkan list history peminjaman kepada pengguna
- 4. Jika (pengguna belum login) maka:
 - a. Tampilkan pesan kesalahan "Silakan login untuk melihat history peminjaman"
- 5. Selesai

Algoritma #7

Contoh:

Nama Kelas : LoanApprovalProcessor

Nama Operasi : processLoanApproval

Algoritma :

- 1. Mulai
- 2. Input: ID peminjaman, keputusan (setuju/tolak), ID pustakawan
- 3. Jika (pustakawan sudah login dan memiliki hak untuk menyetujui peminjaman) maka:

- a. Cari record peminjaman dalam database menggunakan ID peminjaman
- b. Jika (record peminjaman tidak ditemukan) maka:
 - i. Tampilkan pesan kesalahan "Record peminjaman tidak ditemukan"
- c. Jika (record peminjaman ditemukan) maka:
 - i. Periksa status permintaan peminjaman, pastikan masih 'pending'
 - ii. Jika (permintaan sudah diproses sebelumnya) maka:
 - Tampilkan pesan "Permintaan sudah diproses"
 - Hentikan proses
 - iii. Jika (keputusan adalah 'setuju') maka:
 - Update record peminjaman dengan status 'disetujui'
 - Tentukan tanggal peminjaman dan perkiraan tanggal pengembalian
 - Kirim notifikasi ke pengguna bahwa peminjaman disetujui
 - iv. Jika (keputusan adalah 'tolak') maka:
 - Update record peminjaman dengan status 'ditolak'
 - Kirim notifikasi ke pengguna bahwa peminjaman ditolak beserta alasannya jika ada
4. Jika (pustakawan belum login atau tidak memiliki hak akses) maka:
 - a. Tampilkan pesan kesalahan "Anda tidak memiliki hak untuk melakukan aksi ini"
5. Selesai

Algoritma #8

Contoh:

Nama Kelas : LoanStatusUpdater

Nama Operasi : updateLoanStatus

Algoritma :

1. Mulai
2. Input: ID peminjaman, status baru, ID pustakawan
3. Jika (pustakawan sudah login dan memiliki hak untuk mengubah status) maka:
 - a. Cari record peminjaman dalam database menggunakan ID peminjaman
 - b. Jika (record peminjaman tidak ditemukan) maka:
 - i. Tampilkan pesan kesalahan "Record peminjaman tidak ditemukan"
 - c. Jika (record peminjaman ditemukan) maka:
 - i. Verifikasi status baru apakah valid (misalnya 'sedang dipinjam', 'dikembalikan', 'hilang')
 - ii. Jika (status baru tidak valid) maka:
 - Tampilkan pesan kesalahan "Status peminjaman tidak valid"
 - Hentikan proses
 - iii. Jika (status baru valid) maka:
 - Update record peminjaman dengan status baru
 - Jika status adalah 'dikembalikan', perbarui juga tanggal pengembalian aktual
 - Jika ada denda terkait dengan status baru (misalnya terlambat atau hilang), hitung dan tambahkan denda ke record pengguna
 - Kirim notifikasi ke pengguna terkait perubahan status dan detailnya
4. Jika (pustakawan belum login atau tidak memiliki hak akses) maka:
 - a. Tampilkan pesan kesalahan "Anda tidak memiliki hak untuk melakukan aksi ini"
5. Selesai

Algoritma #9

Contoh:

Nama Kelas : BookReturnApproval

Nama Operasi : approveReturn

Algoritma :

1. Mulai
2. Input: ID peminjaman, ID pustakawan
3. Jika (pustakawan sudah login dan memiliki hak untuk menyetujui pengembalian) maka:
 - a. Cari record peminjaman dalam database menggunakan ID peminjaman
 - b. Jika (record peminjaman tidak ditemukan) maka:
 - i. Tampilkan pesan kesalahan "Record peminjaman tidak ditemukan"
 - ii. Hentikan proses
 - c. Jika (record peminjaman ditemukan) maka:
 - i. Verifikasi apakah buku sudah dikembalikan fisik oleh pengguna
 - ii. Jika (buku belum dikembalikan) maka:
 - Tampilkan pesan kesalahan "Buku belum dikembalikan fisik"
 - Hentikan proses
 - iii. Jika (buku sudah dikembalikan) maka:
 - Periksa kondisi buku saat pengembalian
 - Jika (kondisi buku rusak atau hilang) maka:
 - Hitung denda atau biaya penggantian sesuai kebijakan perpustakaan
 - Update record pengguna dengan denda atau biaya penggantian
 - Update record peminjaman dengan status 'dikembalikan'
 - Update inventaris untuk menandai buku sebagai 'tersedia'
 - Kirim notifikasi ke pengguna bahwa pengembalian telah disetujui dan mencatat detailnya (termasuk denda jika ada)
 4. Jika (pustakawan belum login atau tidak memiliki hak akses) maka:
 - a. Tampilkan pesan kesalahan "Anda tidak memiliki hak untuk melakukan aksi ini"
 5. Selesai

Algoritma #10

Contoh:

Nama Kelas : BookSearch

Nama Operasi : searchBooks

Algoritma :

1. Mulai
2. Input: Kata kunci pencarian
3. Jika (kata kunci pencarian tidak kosong) maka:
 - a. Lakukan sanitasi input untuk mencegah serangan seperti SQL Injection
 - b. Gunakan input tersebut untuk melakukan pencarian pada database buku dengan kriteria:
 - i. Judul buku mengandung kata kunci
 - ii. Penulis buku mengandung kata kunci
 - iii. ISBN cocok dengan kata kunci

- iv. Deskripsi buku mengandung kata kunci
- v. Kategori atau genre mengandung kata kunci
- c. Jika (pencarian menghasilkan hasil) maka:
 - i. Kumpulkan hasil pencarian dalam sebuah list
 - ii. Untuk setiap buku dalam list, ambil detail buku yang relevan (judul, penulis, status ketersediaan, dll.)
 - iii. Tampilkan list hasil pencarian kepada pengguna
- d. Jika (pencarian tidak menghasilkan hasil) maka:
 - i. Tampilkan pesan "Tidak ada hasil yang ditemukan, coba kata kunci lain"
- 4. Jika (kata kunci pencarian kosong) maka:
 - a. Tampilkan pesan kesalahan "Masukkan kata kunci untuk mencari buku"
- 5. Selesai

Algoritma #11

Contoh:

Nama Kelas : VisitorReportGenerator

Nama Operasi : generateVisitorReport

Algoritma :

- 1. Mulai
- 2. Input: Rentang tanggal untuk laporan (tanggal awal dan akhir)
- 3. Jika (rentang tanggal valid) maka:
 - a. Mengakses database untuk mengambil data pengunjung antara tanggal awal dan akhir
 - b. Jika (tidak ada data pengunjung dalam rentang tersebut) maka:
 - i. Tampilkan pesan "Tidak ada data pengunjung untuk periode ini"
 - c. Jika (ada data pengunjung) maka:
 - i. Hitung jumlah total pengunjung
 - ii. Identifikasi pola kunjungan, seperti waktu paling sibuk atau hari dengan pengunjung terbanyak
 - iii. Kumpulkan data demografis pengunjung jika tersedia (usia, jenis kelamin, dll.)
 - iv. Buat ringkasan data dalam bentuk yang dapat dibaca, seperti tabel atau grafik
 - v. Tampilkan laporan kepada pengguna atau berikan opsi untuk mengunduh laporan dalam format PDF atau Excel
- 4. Jika (rentang tanggal tidak valid) maka:
 - a. Tampilkan pesan kesalahan "Rentang tanggal tidak valid"
- 5. Selesai

Query #1

*Query: SELECT * FROM users WHERE username = ? AND password = ?*

Keterangan: Query ini digunakan untuk mencari pengguna dalam database dengan username dan password yang dimasukkan. Tanda tanya (?) menandakan placeholder untuk nilai yang akan di-bind dalam aplikasi untuk mencegah SQL Injection.

Query #2

Query: SELECT email FROM users WHERE email = ?

Keterangan: Query ini digunakan untuk mengecek apakah alamat email yang dimasukkan sudah terdaftar dalam database.

Query #3

Query: SELECT status, due_date FROM books WHERE book_id = ?

Keterangan: Query ini digunakan untuk mengecek status dan tanggal jatuh tempo peminjaman buku berdasarkan ID buku yang diminta.

Query #4

*Query: SELECT * FROM loans WHERE user_id = ?*

Keterangan: Query ini digunakan untuk mengambil semua record peminjaman yang terkait dengan ID pengguna tertentu.

Query #5

Query: UPDATE loans SET status = 'returned', return_date = CURRENT_DATE WHERE loan_id = ? AND user_id = ?

Keterangan: Query ini digunakan untuk memperbarui status peminjaman buku sebagai 'dikembalikan' dan menandai tanggal pengembalian.

Query #6

*Query: SELECT * FROM loans WHERE user_id = ? ORDER BY loan_date DESC*

Keterangan: Query ini digunakan untuk mengambil history peminjaman buku oleh pengguna, diurutkan dari yang terbaru.

Query #7

Query: UPDATE loans SET status = ? WHERE loan_id = ?

Keterangan: Query ini digunakan untuk mengupdate status permintaan peminjaman buku berdasarkan keputusan pustakawan.

Query #8

Query: UPDATE loans SET status = ? WHERE loan_id = ?

Keterangan: Query ini digunakan untuk memperbarui status peminjaman buku dalam database.

Query #9

Query: UPDATE loans SET status = 'returned', return_date = CURRENT_DATE, fines = ? WHERE loan_id = ?

Keterangan: Query ini digunakan untuk menyetujui pengembalian buku dan memperbarui database dengan tanggal pengembalian dan denda jika ada.

Query #10

*Query: SELECT * FROM books WHERE title LIKE ? OR author LIKE ? OR isbn LIKE ?*

Keterangan: Query ini digunakan untuk mencari buku yang judul, penulis, atau ISBN-nya sesuai dengan kata kunci yang dimasukkan oleh pengguna.

Query #11

Query: SELECT COUNT(), visit_date FROM visits WHERE visit_date BETWEEN ? AND ? GROUP BY visit_date*

Keterangan: Query ini digunakan untuk menghitung jumlah pengunjung perpustakaan dalam rentang tanggal tertentu dan mengelompokkannya berdasarkan tanggal kunjungan.

4 Matriks Kerunutan (Requirement Traceability Matrix)

<i>Kode FR</i>	<i>Nama Functional Requirement</i>	<i>Nama Use Case</i>
<i>FR1</i>	<i>User Registration</i>	<i>Registrasi Anggota</i>
<i>FR2</i>	<i>User Login</i>	<i>Login</i>
<i>FR3</i>	<i>Book Search</i>	<i>Mencari buku di perpustakaan</i>
<i>FR4</i>	<i>Loan Request Creation</i>	<i>Membuat request peminjaman buku</i>

Kode FR	Nama Functional Requirement	Nama Use Case
<i>FR1</i>	<i>User Registration</i>	<i>Registrasi Anggota</i>
<i>FR5</i>	<i>Loan Status Checking</i>	<i>Melihat status peminjaman buku</i>
<i>FR6</i>	<i>Book Return Processing</i>	<i>Mengembalikan buku</i>
<i>FR7</i>	<i>Borrowing History Viewing</i>	<i>Melihat history peminjaman buku</i>
<i>FR8</i>	<i>Loan Approval Processing</i>	<i>Menyetujui/Menolak peminjaman buku</i>
<i>FR9</i>	<i>Loan Status Updating</i>	<i>Update status peminjaman buku</i>
<i>FR10</i>	<i>Return Approval Processing</i>	<i>Menyetujui pengembalian buku</i>
<i>FR11</i>	<i>Visitor Report Generation</i>	<i>Melihat laporan pengunjung</i>