

SKPL-XXXX

# SPESIFIKASI KEBUTUHAN PERANGKAT LUNAK

## LIBRARIES MANAGEMENT

untuk:

<nama pelanggan>

Dipersiapkan oleh:


Group A

Program Studi S1 Informatika

Fakultas Informatika

Universitas Telkom

2024

 <b>UNIVERSITAS</b> <b>Telkom</b>	<b>Program Studi</b> <b>S1 Informatika</b> <b>-</b> <b>Fakultas</b> <b>Informatika</b>	<b>SKPL - Nomor Dokumen</b>		<b>Halaman</b>
		<b>Revisi</b>	<i>&lt;nomor revisi&gt;</i>	<i>Tgl: &lt;isi tanggal&gt;</i>

# 1. Daftar Perubahan

Revisi	Deskripsi
<b>A</b>	
<b>B</b>	
<b>C</b>	
<b>D</b>	
<b>E</b>	
<b>F</b>	
<b>G</b>	

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

## 2. Daftar Halaman Perubahan

Halaman	Revisi	Halaman	Revisi

### 3. Daftar Isi

<b>Daftar Perubahan</b>	<b>1</b>
<b>Daftar Halaman Perubahan</b>	<b>2</b>
<b>Daftar Isi</b>	<b>3</b>
<b>1.</b>	<b>6</b>
1.1	6
1.2	<b>Kesalahan! Bookmark tidak ditentukan.</b>
1.3	6
1.4	6
<b>2.</b>	<b>7</b>
2.1	7
2.2	7
2.3	Profil dan Kelas Pengguna
2.4	8
2.5	8
2.6	9
<b>3.</b>	<b>10</b>
3.1	10
3.1.1	10
3.1.2	13
3.2	16
3.2.1	16
3.2.2	27
<b>4.</b>	<b>29</b>
4.1	29
4.2	31
4.3	32
4.4	33
4.5	Sistem Cerdas

# 1. Pendahuluan

## 1.1 Tujuan Penulisan Dokumen

Tujuan utama dari dibuatnya dokumen ini adalah sebagai gambaran kebutuhan Sistem Manajemen Perpustakaan. Dokumen ini memberikan deskripsi rinci tentang pembuatan Sistem Manajemen Perpustakaan, mulai dari manajemen koleksi buku, peminjaman dan pengembalian buku, serta informasi pengelola perpustakaan dan pengguna.

## 1.2 Ruang Lingkup / Cakupan Dokumen

Dalam penulisan dokumen ini kami menggunakan font Arial. Dengan spesifikasi untuk judul utamanya yakni: font menggunakan Arial, ukuran hurufnya 32, dan type hurufnya bold. Selanjutnya spesifikasi untuk sub judul yakni : font menggunakan Times New Roman, ukuran hurufnya 14 dan 18, dan type hurufnya bold. Dan terakhir untuk spesifikasi isinya sendiri yakni: font menggunakan Arial, dan ukuran hurufnya 11.

## 1.3 Definisi, Singkatan, dan Akronim

Sistem Manajemen Perpustakaan ini dirancang untuk meningkatkan efisiensi pengelolaan perpustakaan dengan mengutamakan pengelolaan koleksi buku, peminjaman dan pengembalian buku serta pendataan terkait informasi perpustakaan. Fitur utama sistem ini termasuk pendaftaran buku baru, pelacakan inventaris, pengendalian peminjaman, serta memberikan akses ke informasi buku dan riwayat peminjaman. Antarmuka pengguna yang mudah digunakan adalah prioritas utama.

## 1.4 Referensi

Website :

<https://www.coursehero.com/file/65330960/SRS-document-finaldocx/>

## 2. Deskripsi Global Perangkat Lunak

### 2.1 Statement of Objective Perangkat Lunak

Aplikasi libraries management ini ditujukan untuk pustakawan, dan pengunjung perpustakaan. Sistem yang digunakan bersifat mandiri. Sistem yang akan kami buat memberikan fungsi pencarian koleksi didasarkan pada berbagai kategori. Selain itu, pengunjung perpustakaan dapat meminta peminjaman dan pengembalian buku yang nantinya harus mengikuti beberapa kriteria tertentu.

### 2.2 Perspektif dan Goal Perangkat Lunak

Aplikasi libraries management menyediakan informasi secara real-time tentang buku-buku yang tersedia di perpustakaan dan informasi pengguna. Tujuan utama proyek ini adalah mengurangi pekerjaan manual. Perangkat lunak ini mampu mengelola peminjaman buku, pengembalian serta menghitung / mengelola denda. Pustakawan akan bertindak sebagai administrator untuk mengendalikan anggota dan mengelola buku-buku. Status peminjaman/pengembalian anggota disimpan dalam database perpustakaan. Detail anggota dapat diambil oleh pustakawan dari database kapanpun diperlukan.

### 2.3 Profil dan Kelas Pengguna

Terdapat 2 jenis pengguna untuk aplikasi libraries management ini, yakni:

1. Pustakawan
2. Pengunjung perpustakaan

Tabel berikut menggambarkan karakteristik umum pengguna.

Type of User	User Characteristic	User Technical Experience
Pustakawan	<ul style="list-style-type: none"><li>➤ Memiliki pemahaman yang baik terhadap system perpustakaan</li><li>➤ Bertanggung jawab terhadap system perpustakaan</li></ul>	<ul style="list-style-type: none"><li>➤ UI yang menggunakan sedikit input</li><li>➤ Mudah untuk dipelajari</li></ul>
Pengunjung Perpustakaan	<ul style="list-style-type: none"><li>➤ Rata – rata usia 18 – 40 tahun</li><li>➤ Tidak memerlukan pelatihan dalam penggunaan sistem</li></ul>	<ul style="list-style-type: none"><li>➤ GUI yang lebih mudah di pelajari</li><li>➤ Sistem memberikan pesan kesalahan yang tepat dan informatif saat ada kesalahan input dari pengguna</li></ul>

## 2.4 Lingkungan Operasi

Produk Perpustakaan yang akan dioperasikan dalam lingkungan Windows akan didasarkan pada aplikasi desktop. Aplikasi ini dirancang untuk berjalan pada berbagai sistem operasi Windows. Beberapa komponen kunci yang harus diperhatikan adalah:

- Sistem Operasi : Aplikasi ini akan kompatibel dengan Windows 7, 8, 10 dan 11.
- Browsing : Dalam konteks aplikasi desktop, tidak diperlukan kompatibilitas dengan browser. Pengguna hanya perlu menginstal aplikasi pada komputer mereka.
- Persyaratan Jaringan : Aplikasi ini akan memerlukan koneksi internet untuk sinkronisasi data perpustakaan dan pembaruan database.
- Konfigurasi perangkat keras : Hardware minimum yang diperlukan untuk menjalankan aplikasi ini adalah:
  - a. **Hard disk** : Minimum 40 GB
  - b. **Monitor** : Resolusi layar sekurang-kurangnya 1024 × 768 piksel
  - c. **Keyboard** : Standar 101 atau 104 tombol
  - d. **Mouse** : Untuk interaksi dengan antarmuka
  - e. **Printer** : Opsi untuk mencetak laporan dan tag peminjaman

Aplikasi perpustakaan ini akan dirancang untuk mengelola koleksi buku, catatan peminjaman, pengembalian, dan informasi anggota perpustakaan. Pengguna dapat mencari buku, meminjam, dan mengembalikan dengan mudah. Dengan dukungan untuk berbagai jenis buku dan fitur pencarian yang kuat, aplikasi ini akan membantu perpustakaan mengelola sumber daya mereka dengan efisien.

Selain itu, aplikasi perpustakaan ini akan memiliki kemampuan untuk menghasilkan laporan statistik dan menerbitkan kartu keanggotaan. Ini akan memungkinkan perpustakaan untuk memberikan layanan yang lebih baik kepada anggotanya.

## 2.5 Batasan Perangkat Lunak / Sistem

Berikut adalah daftar kendala desain dan implementasi yang akan membatasi pilihan yang tersedia bagi pengembang perangkat lunak Libraries Management:

- Kebijakan Perusahaan dan Regulasi: Perangkat lunak harus sesuai dengan kebijakan perusahaan dan regulasi yang berlaku, termasuk aspek perlindungan data dan keamanan.
- Keterbatasan Perangkat Keras: Terdapat batasan terkait waktu eksekusi dan penggunaan memori yang harus dipertimbangkan selama proses perancangan.
- Integrasi dengan Aplikasi Lain: Perangkat lunak harus diintegrasikan dengan aplikasi lain, yang akan membatasi pilihan antarmuka yang dapat digunakan.
- Teknologi, Alat, dan Database yang Spesifik: Penggunaan teknologi, alat pengembangan, dan database sudah ditetapkan sebelumnya.
- Operasi Paralel: Kemampuan operasi paralel harus dipertimbangkan untuk menampung penggunaan bersama.
- Protokol Komunikasi: Perangkat lunak harus mematuhi protokol komunikasi tertentu yang digunakan dalam lingkungan tersebut.



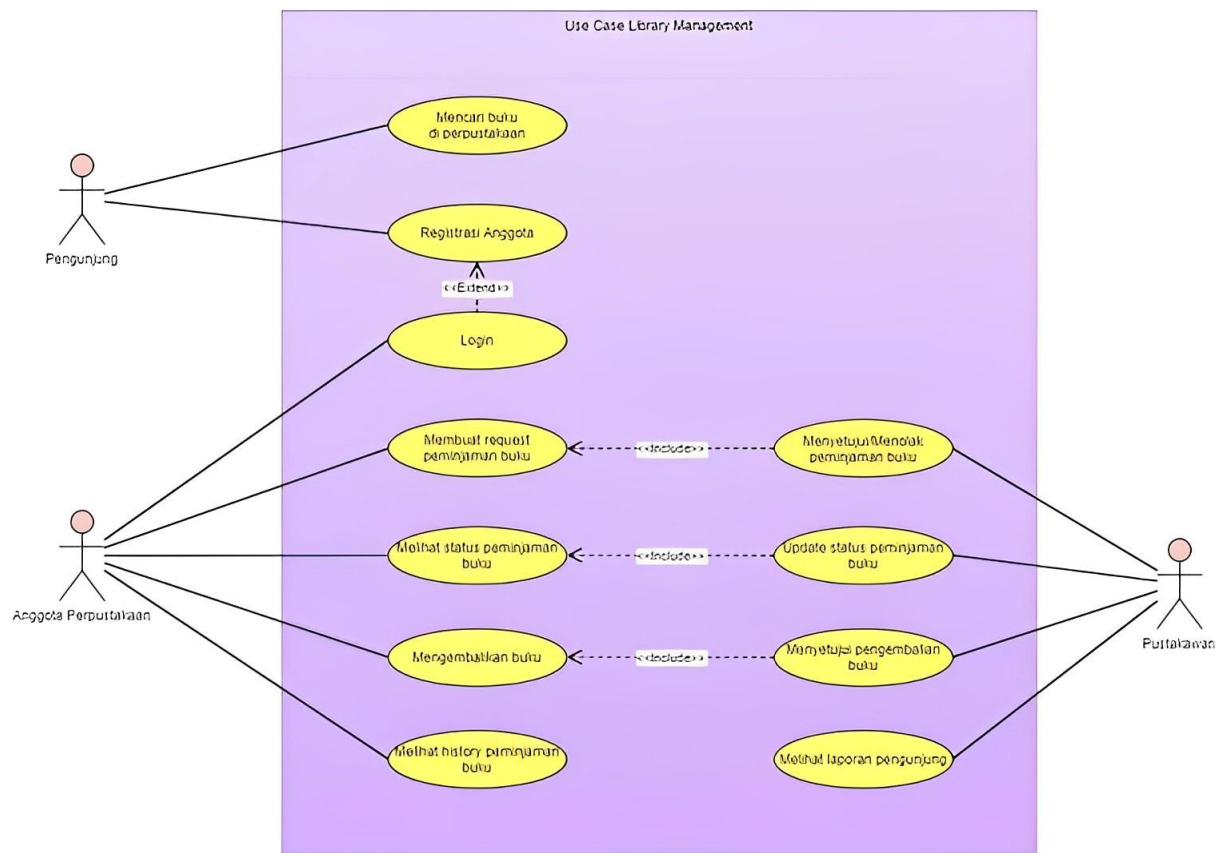
- Pertimbangan Keamanan: Keamanan data dan privasi pengguna harus dipertimbangkan, termasuk tindakan seperti enkripsi dan autentikasi.
- Konvensi Desain dan Standar Pemrograman: Perangkat lunak yang dikembangkan harus dibuat sesuai dengan aturan dan pedoman yang telah ditetapkan oleh organisasi atau industri tertentu.

## 2.6 Asumsi dan Dependensi

Berikut adalah daftar asumsi dan ketergantungan yang dapat mempengaruhi kebutuhan perangkat lunak jika ternyata tidak sesuai:

1. Pustakawan diharapkan memiliki pemahaman dasar tentang komputer dan kemampuan yang efisien dalam menggunakan perangkat lunak.
2. Diperlukan suatu metode untuk mentransfer semua data buku dan catatan pengguna perpustakaan dari sistem yang sudah ada ke dalam Libraries Management.
3. Ketersediaan perangkat keras dan perangkat lunak yang diperlukan.
4. Perpustakaan diharapkan memiliki akses internet yang stabil dan cepat untuk mendukung fitur seperti pencarian online dan penggunaan basis data eksternal.
5. Stabilitas pasokan listrik sangat penting untuk menjaga operasional sistem tanpa gangguan.
6. Perlindungan data yang memadai, termasuk backup data berkala dan tindakan keamanan, akan diterapkan untuk melindungi data sensitif.
7. Ketergantungan pada sistem manajemen database tertentu yang digunakan dalam perangkat lunak ini.
8. Ketergantungan ada pada ketersediaan perangkat pengembangan perangkat lunak, termasuk IDE dan perangkat pengujian yang diperlukan.
9. Ketergantungan ada pada ketersediaan dan keakuratan data yang akan dimasukkan ke dalam sistem. Jika data yang dibutuhkan tidak tersedia atau tidak akurat, hal ini dapat mempengaruhi pelaksanaan proyek.

### 3. Deskripsi Rinci Perangkat Lunak



#### 3.1 Deskripsi Kebutuhan

##### 3.1.1 Kebutuhan Fungsional

Skenario Kebutuhan Fungsional

- **Guest** : Guest merupakan pengunjung yang tidak memiliki Akun
- **User** : User merupakan Anggota Perpustakaan yang berhak dapat meminjam dan mengembalikan Buku
- **Admin** : Admin merupakan Pustakawan yang dapat mengelola seluruh system dan data Buku di Perpustakaan tersebut

No.	Kode Kebutuhan	Deskripsi	Nama Kebutuhan
1.	FR-XX	User dapat menginputkan data barang baru ke dalam sistem	Registrasi Barang
		Atau	

		As User Can Menginputkan data barang baru ke dalam sistem So Data barang baru akan disimpan di basis Data	
1	FR001	User dapat melakukan Register  As Guest to User <b>Can</b> aplikasi manajemen perpustakaan sudah terinstal dan Pengguna akan diminta menginput Akun anggota perpustakaan dengan username dan password sama dengan perulangan password yang diinput <b>So</b> User sudah bisa mendaftarkan akunnya dan sudah terdaftar	Guest do Register
2	FR002	User dapat mencari buku  As Guest <b>Can</b> mencari data buku di Perpustakaan <b>So</b> User sudah bisa melihat data buku di Perpustakaan	Guest do Mencari Buku
3	FR003	User dapat melakukan Login  As User <b>Can</b> aplikasi manajemen perpustakaan sudah terinstal dan Pengguna akan menginput Akun anggota perpustakaan dengan username (Nomor Anggota Perpustakaan) dan password yang valid  <b>So</b> User sudah bisa masuk dari halaman login ke halaman dashboard	Login
4	FR004	User dapat mencari buku  As User <b>Can</b> User telah berhasil login dan memasuki halaman Home dan menekan Button and Field Search <b>Then</b> lalu mencari data buku yang ingin dicari <b>So</b> User sudah bisa menemukan hasil pencarian sesuai dengan kriteria	Cari Buku

5	FR005	<p>User dapat meminjam Buku</p> <p><i>As</i> User  <i>Can</i> User telah bergabung dengan sistem aplikasi manajemen perpustakaan dan telah mengetahui informasi atau judul buku yang ingin dipinjam  <i>So</i> User sudah bisa meminjam Buku dengan baik</p>	Pinjam Buku
6	FR006	<p>User dapat melihat status peminjaman buku</p> <p><i>As</i> User  <i>Can</i> User melihat data Buku yang dipinjam  <i>So</i> User sudah bisa menemukan status peminjaman buku</p>	Status peminjaman Buku
7	FR007	<p>User dapat mengembalikan buku</p> <p><i>As</i> User  <i>Can</i> User melihat data Buku yang dipinjam dan melihat Button pengembalian buku  <i>Then</i> melihat Button pengembalian buku  <i>Do</i> menekan Button pengembalian buku  <i>So</i> User sudah dapat mengembalikan Buku yang akan di setujui oleh Admin</p>	Kembalikan Buku
8	FR008	<p>User dapat melihat History peminjaman buku</p> <p><i>As</i> User  <i>Can</i> User mau melihat History Peminjaman  <i>Then</i> User melihat button history  <i>Do</i> User menekan button history  <i>So</i> User sudah dapat melihat history buku yang dipinjam</p>	History Peminjaman Buku
9	FR009	<p>Admin dapat melakukan Login</p> <p><i>As</i> Admin  <i>Can</i> aplikasi manajemen perpustakaan sudah terinstal dan admin sudah memiliki akun yang sudah terdaftar di system database</p>	Login

		<b>So</b> Admin sudah bisa masuk dari halaman login ke halaman Dashboard	
10	FR010	Admin dapat menyetujui peminjaman Buku  <i>As</i> Admin <b>Can</b> Admin telah mengetahui informasi atau judul buku yang ingin dipinjam oleh User <b>So</b> User dapat menolak Peminjaman Buku	Menyetujui Peminjaman Buku
11	FR011	Admin dapat menolak peminjaman Buku  <i>As</i> Admin <b>Can</b> Admin telah mengetahui informasi atau judul buku yang ingin dipinjam oleh User <b>So</b> admin dapat menolak Peminjaman Buku	menolak Peminjaman Buku
12	FR012	Admin dapat mengupdate status peminjaman Buku  <i>As</i> Admin <b>Can</b> Admin telah mengetahui informasi atau judul buku yang ingin dipinjam oleh User <b>So</b> admin dapat mengupdate status Peminjaman Buku	Update Status Peminjaman buku
13	FR013	Admin dapat melihat Laporan Pengunjung  <i>As</i> Admin <b>Can</b> Admin ingin mengetahui Laporan Pengunjung <b>Then</b> Admin melihat Button Report untuk melihat Laporan Pengunjung <b>Do</b> Admin menekan Button Report untuk melihat Laporan Pengunjung <b>So</b> admin dapat melihat Laporan Pengunjung	Melihat Laporan Pengunjung

### 3.1.2 *Kebutuhan Non-Fungsional*

No.	Quality Criteria	Kode Kebutuhan	Deskripsi
1.	Usability	NFR-XX	Aspek sejauh mana aplikasi mudah dan efektif digunakan.

			Diukur berdasarkan kuesioner SUS
1	Usability	NFR001	<b>Login</b> – UI Harus dapat berhasil dalam melakukan Login saat Guest melakukan Login pada halaman Login
	Usability	NFR002	<b>Cari Buku</b> – UI harus dapat menunjukkan keberhasilan dalam mencari buku yang diinginkan dalam pengujian SUS
	Usability	NFR003	<b>Status Peminjaman Buku</b> – UI harus dapat menunjukkan keberhasilan dalam melihat status peminjaman buku dalam pengujian SUS
	Usability	NFR004	<b>Kembalikan Buku</b> – UI harus dapat menunjukkan dalam mengembalikan buku yang dilakukan dalam Pengujian SUS
	Usability	NFR005	<b>History Peminjaman Buku</b> – UI harus dapat menunjukkan dalam History peminjaman buku yang dilakukan dalam Pengujian SUS
	Usability	NFR006	<b>Register</b> – UI harus dapat menunjukkan dalam pendaftaran akun dilakukan dalam Pengujian SUS
	Functional	NFR007	<b>Register</b> – UI harus dapat menunjukkan dalam pendaftaran akun dilakukan memastikan dapat berhasil
	Functional	NFR008	<b>Login</b> – UI dan System Harus dapat berhasil dalam melakukan Login saat Guest melakukan Login pada halaman Login
	Security	NFR009	<b>Login</b> – System Harus dapat berhasil dalam melakukan otorisasi dan autentikasi login saat melakukan Login pada halaman Login
	Functional	NFR010	<b>Cari Buku</b> – System harus bisa menampilkan data buku yang diinginkan
	Performance	NFR011	<b>Cari Buku</b> – System harus bisa menampilkan data searching buku yang dalam kurun waktu 3 Detik
	Functional	NFR012	<b>Status Peminjaman Buku</b> – system dapat menunjukkan dalam

			keberhasilan melihat status peminjaman Buku
	Functional	NFR013	<b>Kembalikan Buku</b> – system dapat menunjukan dalam keberhasilan mengembalikan peminjaman Buku
	Reliability	NFR014	<b>Kembalikan Buku</b> – system tidak terdapat error dalam mengembalikan peminjaman Buku
	Functional	NFR015	<b>History Peminjaman Buku</b> - system dapat menunjukan keberhasilan mengembalikan peminjaman Buku dalam menampilkan Visual
	Maintability	NFR016	<b>History Peminjaman Buku</b> - system dapat menunjukan Riwayat keberhasilan data buku rapih sesuai filtering
	Usability	NFR017	<b>Menyetujui Peminjaman Buku</b> – UI Admin yang ditampilkan Admin harus berhasil dalam menyetujui Peminjaman Buku
	Usability	NFR018	<b>Menolak Peminjaman Buku</b> – UI admin yang ditampilkan Admin harus berhasil dalam menolak Peminjaman Buku
	Usability	NFR019	<b>Update status Peminjaman Buku</b> – UI admin yang ditampilkan Admin harus berhasil dalam mengupdate status peminjaman buku
	Usability	NFR020	<b>Melihat Laporan Pengunjung</b> – UI admin yang ditampilkan dalam melihat laporan pengunjung
	Functional	NFR021	<b>Menyetujui Peminjaman Buku</b> – System Admin dapat memvalidasi bisnis proses persetujuan peminjaman buku
	Functional	NFR022	<b>Menolak Peminjaman Buku</b> – System Admin dapat memvalidasi bisnis proses menolak peminjaman buku
	Functional	NFR023	<b>Update status Peminjaman Buku</b> – System Admin dapat memvalidasi bisnis proses update peminjaman buku pada system
	Functional	NFR024	<b>Melihat Laporan pengunjung</b> – System Admin dapat

			memvalidasi dalam Melihat Laporan Pengunjung
	Reliability	NFR025	<b>Menyetujui Peminjaman Buku</b> – System Admin tidak terdapat error dalam persetujuan peminjaman buku
	Reliability	NFR026	<b>Update status Peminjaman Buku</b> – System Admin tidak terdapat error update peminjaman buku pada system

## 3.2 Pemodelan Analisis

### 3.2.1 Usecase Diagram

<Gambar UseCase Diagram dari sistem yang akan dikembangkan. Diagram Usecase wajib memiliki minimal 1 relasi INCLUDE dan minimal 1 relasi EXTEND>

#### 3.2.1.1 Usecase Scenario #1 “REGISTER ANGGOTA”

Nama Use Case	Register Anggota	
Deskripsi	Fungsi ini memungkinkan individu untuk mendaftar sebagai anggota perpustakaan melalui sistem online, memungkinkan mereka untuk mengakses layanan dan sumber daya perpustakaan.	
Pre-Kondisi	Pengguna belum terdaftar sebagai anggota perpustakaan.	
Post-Kondisi	Pengguna berhasil terdaftar sebagai anggota perpustakaan dan dapat mengakses layanan yang tersedia.	
Skenario Utama		
	Aktor	Sistem
	3. Memilih opsi 'Daftar sebagai Anggota' pada website perpustakaan atau sistem online	
		4. Menampilkan form pendaftaran online untuk diisi
	5. Mengisi form dengan informasi yang diperlukan seperti nama lengkap, alamat, email, dan informasi lain yang relevan	



		4. Memproses informasi yang dimasukkan dan melakukan verifikasi data
	5. Mengunggah dokumen identifikasi atau dokumen lain yang diperlukan jika diminta	
		6. Mengkonfirmasi pendaftaran dan mengirim email verifikasi atau notifikasi ke pengguna  7. Menciptakan akun anggota dalam sistem dan memberikan akses ke layanan perpustakaan
Skenario Eksepsional (Alternative flow)		
	Aktor	Sistem
	1. ...	
		2. ...

### 3.2.1.2 Usecase Scenario #2 “LOGIN”

Nama Usecase	Login Sistem	
Deskripsi	Fungsi ini digunakan oleh pengguna untuk mengakses sistem dengan memasukkan kredensial yang valid.	
Pre-Kondisi	Pengguna berada di halaman login.	
Post-Kondisi	Pengguna berhasil masuk ke dashboard utama aplikasi manajemen perpustakaan.	
Skenario Utama		
	Aktor	Sistem
	1. Memasukkan nama pengguna	
		2. Meminta sandi
	3. Memasukkan sandi	

		4. Memvalidasi kredensial
		5. Menampilkan dashboard jika kredensial benar
		6. Menampilkan pesan error jika kredensial salah
Skenario Eksepsional (Alternative flow)		
	Aktor	Sistem
	1.	
		2.

### 3.2.1.3 Usecase Scenario #3 “REQUEST PEMINJAMAN BUKU”

Nama Usecase	Request Peminjaman Buku	
Deskripsi	Fungsi ini digunakan oleh pengguna untuk mengajukan peminjaman buku melalui sistem perpustakaan berbasis web.	
Pre-Kondisi	Pengguna telah berhasil login ke dalam sistem.	
Post-Kondisi	Permintaan peminjaman buku telah tercatat di sistem dan menunggu persetujuan dari pustakawan.	
Skenario Utama		
	Aktor	Sistem
	1. Memilih opsi 'Peminjaman Buku' dari menu utama	
		2. Menampilkan daftar buku yang tersedia untuk dipinjam
	3. Mencari dan memilih buku yang diinginkan	
		4. Menampilkan detail buku dan formulir peminjaman
	5. Mengisi formulir peminjaman dengan informasi yang diperlukan (tanggal peminjaman, durasi, dll.)	

		6. Mengonfirmasi detail peminjaman dan mengajukan request  7. Menyimpan request peminjaman dan memberikan notifikasi berhasil kepada pengguna
Skenario Eksepsional (Alternative flow)		
	Aktor	Sistem
	1.	
		2.

#### 3.2.1.4 Usecase Scenario #4 “STATUS PEMINJAMAN BUKU”

Nama Usecase	Status Peminjaman Buku	
Deskripsi	Fungsi ini digunakan oleh pengguna untuk mengecek status terkini dari buku-buku yang sedang dipinjam atau dalam proses peminjaman.	
Pre-Kondisi	Pengguna telah berhasil login dan memiliki buku yang sedang dipinjam atau telah mengajukan peminjaman.	
Post-Kondisi	Pengguna mendapatkan informasi detail tentang status peminjaman bukunya.	
Skenario Utama		
	Aktor	Sistem
	1. Memilih opsi 'Status Peminjaman' dari menu utama atau dashboard	
		2. Menampilkan daftar semua peminjaman buku yang dilakukan oleh pengguna
	3. Memilih peminjaman tertentu untuk melihat detail lebih lanjut	
		7. Menampilkan detail status peminjaman seperti tanggal peminjaman,

		tanggal pengembalian, status persetujuan, dan denda jika ada
Skenario Eksepsional (Alternative flow)		
	Aktor	Sistem
	1.	
		2.

### 3.2.1.5 Usecase Scenario #5 “PENGEMBALIAN BUKU”

Nama Usecase	Pengembalian Buku	
Deskripsi	Fungsi ini digunakan oleh pengguna untuk melakukan proses pengembalian buku yang telah dipinjam melalui sistem perpustakaan berbasis web.	
Pre-Kondisi	Pengguna telah berhasil login dan memiliki buku yang sedang dipinjam.	
Post-Kondisi	Buku yang dipinjam telah dikembalikan ke perpustakaan dan sistem telah memperbarui status peminjaman.	
Skenario Utama		
	Aktor	Sistem
	1. Memilih opsi 'Pengembalian Buku' dari menu utama	
		2. Menampilkan daftar buku yang sedang dipinjam oleh pengguna
	3. Memilih buku yang ingin dikembalikan	
		4. Menampilkan form pengembalian buku dengan detail peminjaman
	5. Memverifikasi informasi pada form pengembalian dan mengkonfirmasi pengembalian	
		6. Memproses pengembalian dan mengupdate status buku

		sebagai 'dikembalikan'
		7. Menampilkan notifikasi bahwa buku telah berhasil dikembalikan dan menginformasikan jika ada denda yang harus dibayar
Skenario Eksepsional (Alternative flow)		
	Aktor	Sistem
	1.	
		2.

#### 7.1.1.1 Usecase Scenario #6 “HISTORY PEMINJAMAN BUKU”

Nama Usecase	History Peminjaman Buku	
Deskripsi	Fungsi ini memungkinkan pengguna untuk melihat riwayat lengkap dari semua buku yang telah dipinjam dari perpustakaan melalui sistem.	
Pre-Kondisi	Pengguna telah berhasil login ke dalam sistem.	
Post-Kondisi	Pengguna dapat meninjau semua transaksi peminjaman buku yang telah dilakukan.	
Skenario Utama		
	Aktor	Sistem
	1. Memilih opsi 'History Peminjaman' dari menu utama atau dashboard	
		2. Memilih opsi 'History Peminjaman' dari menu utama atau dashboard
	3. Jika diperlukan, pengguna dapat memilih transaksi tertentu untuk melihat detail lebih lanjut	
		4. Menampilkan detail lengkap dari transaksi peminjaman yang dipilih, termasuk riwayat denda dan catatan lainnya

Skenario Eksepsional (Alternative flow)		
	Aktor	Sistem
	1.	
		2.

### 3.2.1.7 Usecase Scenario #7 “MENYETUJUI ATAU MENOLAK PEMINJAMAN BUKU”

Nama Usecase	Menyetujui atau Menolak Peminjaman Buku	
Deskripsi	Fungsi ini digunakan oleh pustakawan untuk menyetujui atau menolak permintaan peminjaman buku yang diajukan oleh pengguna.	
Pre-Kondisi	Pustakawan telah berhasil login ke dalam sistem dan ada permintaan peminjaman buku yang tertunda.	
Post-Kondisi	Permintaan peminjaman buku telah diupdate dengan status disetujui atau ditolak.	
Skenario Utama		
	Aktor	Sistem
	1. Memilih opsi 'Review Peminjaman' dari dashboard pustakawan	
		2. Menampilkan daftar permintaan peminjaman yang menunggu persetujuan
	3. Memilih permintaan peminjaman tertentu untuk ditinjau	
		4. Menampilkan detail permintaan peminjaman termasuk informasi peminjam dan detail buku
	5. Memutuskan untuk menyetujui atau menolak permintaan berdasarkan ketersediaan buku dan kebijakan perpustakaan	
		6. Mengupdate sistem dengan keputusan

		pustakawan dan mengirim notifikasi ke pengguna terkait status permintaan
Skenario Eksepsional (Alternative flow)		
	Aktor	Sistem
	1.	
		2.

### 3.2.1.8 Usecase Scenario #8 “UPDATE STATUS PEMINJAMAN BUKU”

Nama Usecase	Update Status Peminjaman Buku	
Deskripsi	Fungsi 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.	
Pre-Kondisi	Pustakawan telah berhasil login ke dalam sistem.	
Post-Kondisi	Status peminjaman buku telah diperbarui dalam sistem sesuai dengan kondisi aktual.	
Skenario Utama		
	Aktor	Sistem
	1. Memilih opsi 'Update Status Peminjaman' dari dashboard pustakawan	
		2. Menampilkan daftar peminjaman buku yang aktif
	3. Memilih peminjaman buku tertentu untuk diperbarui	
		4. Menampilkan form untuk memperbarui status peminjaman buku
	5. Memasukkan pembaruan status, seperti perubahan tanggal pengembalian, kondisi buku, atau catatan lain yang relevan	

		6. Memverifikasi dan mengkonfirmasi perubahan yang dimasukkan  7. Mengupdate informasi dalam sistem dan memberikan konfirmasi sukses kepada pustakawan
Skenario Eksepsional (Alternative flow)		
	Aktor	Sistem
	1.	
		2.

### 3.2.1.9 Usecase Scenario #9 “MENYETUJUI PENGEMBALIAN BUKU”

Nama Usecase	Menyetujui Pengembalian Buku	
Deskripsi	Fungsi ini digunakan oleh pustakawan untuk memverifikasi dan menyetujui pengembalian buku yang telah dilakukan oleh pengguna.	
Pre-Kondisi	Pustakawan telah berhasil login ke dalam sistem dan buku yang dipinjam telah fisik dikembalikan ke perpustakaan.	
Post-Kondisi	Buku yang dikembalikan telah diperiksa, statusnya diperbarui di sistem, dan proses peminjaman ditutup.	
Skenario Utama		
	Aktor	Sistem
	1. Memilih opsi 'Verifikasi Pengembalian Buku' dari dashboard pustakawan	
		2. Menampilkan daftar buku yang dikembalikan dan menunggu verifikasi
	3. Memilih buku yang dikembalikan untuk diverifikasi	
		4. Menampilkan form verifikasi pengembalian dengan detail peminjaman dan kondisi buku



	5. Memeriksa buku secara fisik dan mengisi form verifikasi dengan status kondisi buku	
		6. Mengkonfirmasi kebenaran pengembalian dan kondisi buku  7. Mengupdate status peminjaman buku di sistem sebagai 'dikembalikan' dan mengakhiri peminjaman
Skenario Eksepsional (Alternative flow)		
	Aktor	Sistem
	1.	
		2.

### 3.2.1.10 Usecase Scenario #10 “MENCARI BUKU DI PERPUSTAKAAN”

Nama Usecase	Mencari Buku di Perpustakaan	
Deskripsi	Fungsi ini memungkinkan pengguna untuk mencari dan menemukan buku yang tersedia dalam koleksi perpustakaan menggunakan sistem online.	
Pre-Kondisi	Pengguna telah berhasil login ke dalam sistem atau menggunakan interface publik sistem pencarian.	
Post-Kondisi	Pengguna mendapatkan daftar buku yang sesuai dengan kriteria pencarian mereka.	
Skenario Utama		
	Aktor	Sistem
	1. Memasuki kata kunci pencarian dalam bar pencarian atau menggunakan fitur pencarian lanjutan	
		2. Menerima kata kunci dan melakukan pencarian dalam basis data perpustakaan

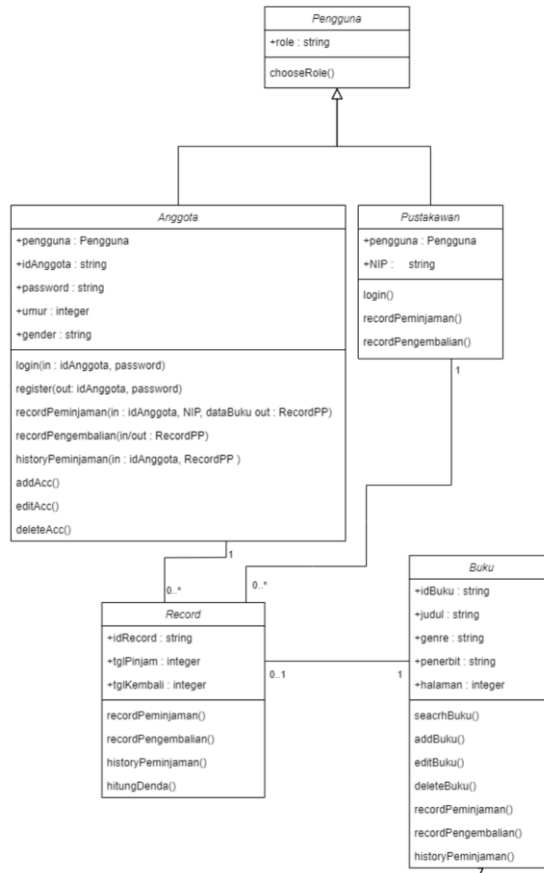
		3. Menampilkan daftar buku yang sesuai dengan kriteria pencarian, termasuk judul, penulis, dan status ketersediaan
	4. Jika diperlukan, memilih judul buku untuk mendapatkan informasi lebih detail	
		5. Menampilkan informasi detail buku seperti lokasi rak, ringkasan, dan informasi peminjaman
Skenario Eksepsional (Alternative flow)		
	Aktor	Sistem
	1.	
		2.

### 3.2.1.11 Usecase Scenario #11 “MELIHAT LAPORAN PENGUNJUNG”

Nama Usecase	Melihat Laporan Pengunjung	
Deskripsi	Fungsi ini digunakan oleh pustakawan untuk meninjau dan menganalisis data pengunjung yang telah masuk ke perpustakaan dalam periode tertentu.	
Pre-Kondisi	Pustakawan telah berhasil login ke dalam sistem.	
Post-Kondisi	Pustakawan mendapatkan laporan pengunjung yang dapat digunakan untuk analisis statistik atau keperluan manajemen perpustakaan.	
Skenario Utama		
	Aktor	Sistem
	1. Memilih opsi 'Laporan Pengunjung' dari dashboard pustakawan	
		2. Menampilkan pilihan untuk menentukan periode waktu laporan
	3. Memasukkan rentang tanggal dan kriteria lainnya untuk laporan	

		4. Menghasilkan dan menampilkan laporan berdasarkan data pengunjung pada periode yang dipilih
	5. Menganalisis data laporan yang disajikan dalam bentuk tabel atau grafik	
		6. Menyediakan opsi untuk mengunduh laporan dalam format yang diinginkan (PDF, Excel, dll.)
Skenario Eksepsional (Alternative flow)		
	Aktor	Sistem
	1.	
		2.

### 3.2.2 Class Diagram:



## 4 Kebutuhan Lain - Lain

### 4.1 Antarmuka Pengguna

Berikut beberapa kebutuhan UI (User Interface) dan UX (User Experience) yang berkaitan dengan kedua kelas pengguna, yaitu pustakawan dan pengunjung perpustakaan.

#### Untuk Pustakawan:

- a. **UI:**
  - Antarmuka yang fungsional dan efisien dengan akses cepat ke berbagai fungsi perpustakaan seperti katalog, peminjaman, pengembalian, dan laporan.
  - Desain antarmuka yang lebih teknis yang memungkinkan untuk tindakan administratif seperti pengelolaan data anggota, update inventaris buku, dan pengaturan sistem.
- b. **UX:**
  - Alur kerja yang terorganisir dengan baik yang sesuai dengan proses kerja sehari-hari pustakawan, meminimalkan jumlah klik untuk melakukan tugas umum.
  - Sistem harus memberikan umpan balik yang jelas dan tepat ketika pustakawan melakukan aksi, seperti konfirmasi peminjaman, pengembalian buku, atau saat terjadi error.
  - Dukungan untuk multitasking, memungkinkan pustakawan untuk dengan mudah beralih antara tugas tanpa kehilangan konteks kerja.

#### Untuk Pengunjung Perpustakaan:

- a. **UI:**
  - Antarmuka yang intuitif dan mudah digunakan, dengan navigasi yang sederhana sehingga pengunjung baru dengan rentang usia 18-40 tahun dapat dengan mudah memahami cara menggunakan sistem tanpa perlu pelatihan.
  - GUI (Graphical User Interface) harus menyertakan visualisasi yang menarik dan interaktif yang membantu dalam pencarian dan penjelajahan katalog buku.
- b. **UX:**
  - Pengalaman pengguna yang memuaskan dengan loading times yang cepat dan respon sistem yang konsisten, mengurangi frustrasi pengguna saat mencari informasi.
  - Pesan kesalahan yang informatif dan bantuan dalam aplikasi yang memandu pengguna melalui proses seperti pendaftaran anggota, pencarian buku, dan membuat permintaan peminjaman.
  - Fitur yang mendukung aksesibilitas, termasuk teks yang dapat dibaca, kontras yang cukup, dan navigasi yang dapat diakses melalui keyboard atau teknologi bantu lainnya.

Adapun penerapan UI dan UX yang disesuaikan dengan kebutuhan pustakawan dan pengunjung perpustakaan:

**Untuk Pustakawan:**

a. UI:

➤ **Dashboard yang Dapat Dikustomisasi:**

- Panel kontrol dengan widget yang dapat disesuaikan untuk memantau status inventaris buku, permintaan peminjaman terbaru, dan pemberitahuan penting.

➤ **Menu Navigasi Tingkat Lanjut:**

- Menu samping atau dropdown dengan kategori yang terorganisir berdasarkan fungsi administratif seperti manajemen pengguna, transaksi buku, laporan keuangan, dan statistik perpustakaan.

➤ **Tabel dan Formulir Interaktif:**

- Tabel dengan fungsi pencarian, penyortiran, dan penyaringan untuk mengelola daftar buku dan anggota, serta formulir yang dapat diisi dengan auto-complete untuk entri data yang efisien.

b. UX:

• **Workflow Otomatisasi:**

- Proses otomatis untuk tugas-tugas umum seperti peringatan untuk buku yang hampir jatuh tempo dan rekomendasi pembelian buku baru berdasarkan tren peminjaman.

• **Bantuan Kontekstual dan Tutorial:**

- Pop-up bantuan atau video tutorial yang tersedia untuk menjelaskan fitur baru atau mengatasi error yang umum terjadi.

• **Fitur Multi-Tugas:**

- Kemampuan untuk membuka tab atau jendela baru tanpa kehilangan progres pada tugas yang sedang dikerjakan, seperti membalas pertanyaan anggota sambil memproses peminjaman buku.

**Untuk Pengunjung Perpustakaan:**

a. UI:

➤ **Desain yang Intuitif dan Responsif:**

- Antarmuka yang responsif yang menyesuaikan tampilan berdasarkan perangkat yang digunakan, baik desktop, tablet, atau smartphone.

➤ **Pencarian Cerdas:**

- Kotak pencarian dengan fitur autocomplete yang menyarankan judul buku, penulis, atau genre saat pengguna mulai mengetik.

➤ **Kartu Informasi Buku:**

- Kartu-kartu buku yang menampilkan gambar sampul, ringkasan singkat, dan status ketersediaan buku dengan opsi untuk menempatkan buku pada daftar tunggu atau peminjaman.

b. UX:

➤ **Proses Pendaftaran yang Mudah:**

- Proses pendaftaran langkah demi langkah dengan instruksi yang jelas dan jumlah form yang minim untuk memudahkan pengguna baru bergabung.
- **Pesan Kesalahan yang Bermanfaat:**
  - Ketika pengguna melakukan kesalahan input, sistem menyediakan pesan kesalahan yang jelas dengan petunjuk tentang cara memperbaikinya.
- **Navigasi yang Efisien:**
  - Breadcrumbs, ikon yang mudah dikenali, dan tombol balik yang memungkinkan pengguna untuk dengan mudah menavigasi kembali ke halaman atau fungsi sebelumnya.

## 4.2 Antarmuka Perangkat Keras

Berikut ini kebutuhan sistem yang menjelaskan interaksi antara aplikasi dengan berbagai perangkat keras:

1. **Bar Code Reader:**
  - **Integrasi Driver:** Sistem harus mampu mengintegrasikan driver untuk berbagai model pembaca barcode agar perangkat dapat dikenali dan berfungsi dengan benar.
  - **Protokol Input:** Sistem harus menggunakan protokol input yang standar (misalnya keyboard emulation) agar data yang dibaca dari barcode langsung dimasukkan ke dalam form yang sesuai di aplikasi web tanpa perlu proses manual.
  - **Pemrosesan Data:** Harus ada prosedur untuk memvalidasi dan memproses data yang diterima dari barcode reader, misalnya untuk mengidentifikasi buku yang sedang dipinjam atau dikembalikan.
2. **Server:**
  - **Kapasitas dan Kebutuhan Hardware:** Spesifikasi server harus memadai untuk menangani beban kerja aplikasi, dengan mempertimbangkan jumlah pengguna simultan, kapasitas penyimpanan data, dan kebutuhan pemrosesan.
  - **Koneksi Jaringan:** Server harus memiliki koneksi jaringan yang stabil dan cepat untuk mendukung akses simultan dari banyak pengguna tanpa penurunan performa.
3. **Mesin Pembaca Fingerprint:**
  - **Integrasi API:** Aplikasi harus menyediakan API untuk integrasi dengan pembaca sidik jari, yang memungkinkan otentikasi pengguna.
  - **Keamanan Data:** Sistem harus mengimplementasikan standar keamanan yang tinggi untuk menyimpan dan mengelola data biometrik, termasuk enkripsi dan kepatuhan terhadap regulasi privasi data.
4. **Printer dan Perangkat Lainnya:**
  - **Driver Universal atau Web Printing:** Sistem harus kompatibel dengan driver printer universal atau mendukung teknologi web printing untuk memungkinkan pencetakan dari berbagai perangkat tanpa perlu instalasi driver spesifik.
  - **Format Output:** Harus mendukung generasi dokumen dalam format yang umum seperti PDF untuk memastikan kompatibilitas lintas platform dan perangkat.

5. **Perangkat Mobile (Tablets dan Smartphones):**

- **Responsive Design:** Antarmuka pengguna aplikasi web harus responsif dan dioptimalkan untuk penggunaan melalui perangkat mobile.
- **Touch Interface:** Aplikasi harus mendukung interaksi melalui touch interface, yang memungkinkan penggunaan fitur-fitur aplikasi melalui layar sentuh.

6. **Self-Service Kiosks:**

- **Integrasi dengan Hardware Kiosk:** Aplikasi harus bisa berintegrasi dengan kiosk yang memiliki hardware seperti scanner dokumen, printer, dan layar sentuh.
- **User Interface Kiosk:** Antarmuka untuk kiosk harus dirancang agar mudah digunakan oleh pengguna dengan petunjuk yang jelas dan navigasi yang sederhana.

## 4.3 Antarmuka Perangkat Lunak

Berikut adalah kebutuhan untuk memastikan sistem dapat berinteraksi dengan baik dengan sistem lain:

1. **API Terdefinisi dengan Baik (Application Programming Interface):**

- Harus ada API yang terdokumentasi dengan baik untuk memungkinkan sistem manajemen perpustakaan berkomunikasi dengan sistem lain seperti sistem tagihan, sistem pendaftaran pelanggan, atau sistem inventaris.
- API harus mendukung protokol komunikasi standar seperti REST atau SOAP untuk pertukaran data yang mudah dan aman.

2. **Format Data Standar:**

- Sistem harus dapat mengimpor dan mengekspor data dalam format yang standar dan luas digunakan, seperti XML, JSON, atau CSV, untuk memastikan kompatibilitas antar sistem.
- Skema data dan model harus diselaraskan antara sistem untuk memastikan integritas data saat transfer data.

3. **Otentikasi dan Otorisasi:**

- Mekanisme otentikasi yang aman seperti OAuth, token JWT, atau kredensial berbasis sertifikat harus diterapkan untuk mengontrol akses ke API.
- Harus ada kebijakan otorisasi yang mendefinisikan level akses yang berbeda untuk sistem yang berbeda, mencegah akses tidak sah ke data sensitif.

4. **Keselarasan Proses Bisnis:**

- Proses bisnis yang berjalan di sistem manajemen perpustakaan harus selaras dengan sistem lain untuk menghindari konflik dan redundansi.
- Workflow otomatis harus didefinisikan untuk proses yang membutuhkan interaksi antar sistem, seperti pembaharuan status tagihan setelah pendaftaran pelanggan.

5. **Middleware dan Integrasi Layanan:**

- Mungkin diperlukan middleware atau perangkat lunak integrasi untuk mengkoordinasikan komunikasi antara sistem yang tidak langsung kompatibel.



- Layanan integrasi seperti ESB (Enterprise Service Bus) atau iPaaS (Integration Platform as a Service) dapat digunakan untuk mengelola interaksi yang kompleks.
6. **Sinkronisasi dan Pembaruan Data:**
    - Mekanisme harus ada untuk memastikan bahwa data disinkronkan secara real-time atau melalui jadwal teratur untuk memastikan konsistensi.
    - Sistem harus dapat menangani pembaruan data yang bertentangan atau duplikasi dengan logika bisnis yang tepat.
  7. **Pemantauan dan Log Transaksi:**
    - Sistem harus menyediakan log transaksi yang rinci untuk semua interaksi antar sistem.
    - Pemantauan dan alat analitik harus digunakan untuk melacak kesehatan integrasi dan untuk mengidentifikasi serta menyelesaikan masalah yang muncul.
  8. **Manajemen Error dan Pemulihan:**
    - Harus ada strategi untuk penanganan kesalahan, seperti kembali mencoba transaksi yang gagal atau memasukkan item ke dalam antrian untuk diproses nanti.
    - Sistem harus mampu pulih dari kegagalan dan memastikan integritas data melalui transaksi yang atomic.

## 4.4 Antarmuka Komunikasi

Berikut adalah beberapa requirements yang harus diperhatikan:

1. **Email:**
  - **SMTP Server:** Sistem harus terintegrasi dengan server SMTP yang dapat diandalkan untuk mengirim notifikasi dan komunikasi lain via email kepada pengguna dan staf perpustakaan.
  - **Format Email:** Template email harus memiliki format standar HTML/CSS yang responsif untuk tampilan di berbagai klien email.
  - **Keamanan Email:** Penggunaan TLS/SSL harus diimplementasikan untuk memastikan keamanan dalam pengiriman email.
2. **Web Browser:**
  - **Kesesuaian Browser:** Antarmuka pengguna harus kompatibel dengan versi terbaru dari browser utama seperti Chrome, Firefox, Safari, dan Edge.
  - **Responsive Design:** Desain web harus responsif untuk menyesuaikan dengan berbagai ukuran layar dan resolusi perangkat.
3. **Protokol Komunikasi Server Jaringan:**
  - **HTTP/HTTPS:** Semua komunikasi antara klien dan server harus menggunakan protokol HTTP dengan SSL/TLS (HTTPS) untuk enkripsi dan keamanan data.
  - **WebSocket:** Untuk real-time communication, seperti chat atau notifikasi instan, WebSocket dapat digunakan.

#### 4. Formulir Elektronik:

- **Validasi Formulir:** Harus ada validasi client-side dan server-side untuk semua input form untuk mencegah kesalahan pengguna dan serangan injeksi.
- **CAPTCHA:** Untuk formulir yang terbuka bagi publik, seperti pendaftaran, CAPTCHA atau reCAPTCHA harus digunakan untuk mencegah spam.

#### 5. Standar Komunikasi:

- **FTP/SFTP:** Untuk transfer file yang aman, SFTP (FTP melalui SSH) harus digunakan daripada FTP yang tidak aman.
- **API RESTful:** API untuk komunikasi antar sistem harus mengikuti prinsip RESTful menggunakan protokol HTTP/HTTPS.

#### 6. Masalah Keamanan Komunikasi dan Enkripsi:

- **SSL/TLS:** Enkripsi SSL/TLS harus digunakan untuk semua komunikasi.
- **OAuth 2.0:** Untuk otentikasi API, standar OAuth 2.0 harus digunakan.

#### 7. Kecepatan Transfer Data:

- **Pemampatan Data:** Data harus dikompresi untuk transfer yang lebih cepat, terutama untuk data yang besar seperti gambar buku atau dokumen.
- **Content Delivery Network (CDN):** Menggunakan CDN untuk mendistribusikan konten statis dapat meningkatkan kecepatan loading.

#### 8. Mekanisme Sinkronisasi:

- **Queueing Service:** Menggunakan layanan antrian seperti RabbitMQ atau Amazon SQS untuk menangani proses yang memerlukan sinkronisasi data antar sistem.
- **Cron Jobs:** Tugas-tugas yang dijadwalkan (cron jobs) untuk sinkronisasi batch data antara sistem pada interval waktu yang ditentukan.

#### 9. Pengujian dan Kepatuhan:

- **Pengujian Beban dan Keamanan:** Rutin melakukan pengujian beban dan keamanan untuk memastikan sistem dapat menangani trafik tinggi dan melindungi dari kerentanan.
- **Kepatuhan Standar:** Memastikan sistem memenuhi standar kepatuhan seperti GDPR untuk perlindungan data pribadi.

### 4.5 Fitur sistem cerdas

Berikut adalah kebutuhan yang berkaitan dengan pemanfaatan fitur cerdas dalam sistem:

#### 1. Natural Language Processing (NLP):

- Fitur pencarian cerdas dengan NLP yang memungkinkan pengguna untuk melakukan query pencarian menggunakan bahasa alami.
- Chatbots yang dapat menjawab pertanyaan umum pengguna dan membantu navigasi sistem dengan penggunaan NLP.

#### 2. Prediksi Ketersediaan Buku:

- Menggunakan data historis untuk memprediksi ketersediaan buku dan waktu terbaik untuk peminjaman.
  - Pemberitahuan otomatis kepada pengguna saat buku yang mereka inginkan kembali tersedia.
3. **Pengenalan Suara:**
    - Fitur pengenalan suara untuk memudahkan pengguna melakukan pencarian atau perintah lain dalam aplikasi.
  4. **User Interface Adaptif:**
    - Antarmuka yang beradaptasi dengan pola penggunaan pengguna, menyesuaikan tampilan dan fungsi berdasarkan preferensi dan perilaku pengguna.
  5. **Feedback dan Learning Loop:**
    - Sistem harus mengumpulkan feedback dari pengguna dan belajar dari interaksi ini untuk terus meningkatkan rekomendasi dan layanan.
  6. **Otomatisasi Tugas Rutin:**
    - Otomatisasi proses pengingat pengembalian buku, penagihan denda, dan tugas administratif lainnya.
  7. **Keamanan dan Privasi Data:**
    - Implementasi keamanan yang kuat untuk melindungi data pengguna dan interaksi mereka dengan sistem, termasuk penggunaan enkripsi dan protokol keamanan yang tepat.