*Software Requirements Specification*

for

Aplikasi Peminjaman dan Manajemen Stok Buku Perpustakaan Global

Version 1.0 approved

Prepared by

1122140079 – Miftahudin Aldi Saputra

1122140068 – Dwi Bayu Nugraha

1122140029 – Wahyu Nayoga

Fakultas Teknik Informatika/Software Engineering

26 Mei 2025

Table of Contents

1. Pendahuluan 1

1.1 Tujuan Penulisan Dokumen 1

1.2 Audien yang Dituju dan Pembaca yang Disarankan 1

1.3 Batasan Produk 1

1.4 Definisi dan Istilah 1

1.5 Refrensi 1

2. Deskripsi Keseluruhan 2

2.1 Deskripsi Produk 2

2.2 Fungsi Produk 2

2.3 Penggolongan Karakterik Pengguna 2

2.4 Lingkungan Operasi 2

2.5 Batasan Desain dan Implementasi 2

2.6 Dokumentasi Pengguna 3

3. Kebutuhan Antarmuka Eksternal 4

3.1 User Interfaces 4

3.2 Hardware Interface 4

3.3 Software Interface 4

3.4 Communication Interface 4

4. Functional Requirement 5

4.1 Use Case Diagram 5

4.2 Nama Use Case 1 5

4.3 Nama Use Case 2 5

4.4 Class Diagram 6

5. Non Functional Requirements 7

Revision History

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Date** | **Reason For Changes** | **Version** |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

# Pendahuluan

## Tujuan Penulisan Dokumen

Dokumen ini bertujuan untuk mendefinisikan kebutuhan fungsional dan non-fungsional dari aplikasi "Aplikasi Peminjaman dan Manajemen Stok Buku Perpustakaan Global", yang akan digunakan untuk membantu proses peminjaman buku, pengembalian, pembayaran denda, dan manajemen stok buku oleh petugas perpustakaan. Tujuan utamanya adalah memberikan panduan yang jelas bagi tim pengembang, tester, dan pemangku kepentingan (stakeholders) dalam merancang, membangun, dan memelihara aplikasi sesuai dengan kebutuhan bisnis.

## Audien yang Dituju dan Pembaca yang Disarankan

Dokumen ini ditujukan untuk:

* Tim pengembang aplikasi (Developer)
* Staf perpustakaan

## Batasan Produk

Aplikasi ini dibangun untuk platform mobile (Flutter), menggunakan Firebase sebagai backend (Authentication, Firestore). Aplikasi ini akan berjalan secara online dan tidak mendukung mode offline.

## Definisi dan Istilah

1. **SRS**: Software Requirements Specification
2. **Firebase**: Platform backend dari Google
3. **Flutter**: Framework UI untuk pengembangan aplikasi lintas platform
4. **Denda**: Biaya yang dikenakan untuk keterlambatan pengembalian buku

## Refrensi

 [Flutter Documentation](https://flutter.dev)

 [Firebase Documentation](https://firebase.google.com/docs)

# Deskripsi Keseluruhan

## Deskripsi Produk

Aplikasi Mobile PerpusGlo adalah sistem digital berbasis Flutter dengan Firebase sebagai backendnya yang dirancang untuk mendukung proses peminjaman dan manajemen stok buku di perpustakaan global. Aplikasi ini dirancang sebagai solusi untuk memungkinkan pengguna untuk meminjam buku perpustakaan secara digital, mengelola buku yang dipinjam, memperpanjang masa pinjam, membayar denda keterlambatan, mengakses katalog buku secara online dan staf perpustakaan untuk mengelola buku di perpustakaan.

## Fungsi Produk

1. Registrasi dan login pengguna (Email global)
2. Melihat dan mencari buku
3. Melakukan peminjaman buku
4. Melakukan pengembalian/perpanjangan
5. Melihat riwayat transaksi
6. Melakukan pembayaran denda (via QRIS/manual/ TF rek)
7. Notifikasi pengingat jatuh tempo
8. Manajemen stok buku (Admin)

## Penggolongan Karakterik Pengguna

<Identifikasi berbagai golongan pengguna yang terkait dengan produk yang dikembangkan>

Tabel 1 Karakteristik Pengguna

| **Kategori Pengguna** | **Tugas** | **Hak Akses ke aplikasi** | **Kemampuan yang harus dimiliki** |
| --- | --- | --- | --- |
| Mahasiswa | Meminjam dan mengembalikan buku | Akses peminjaman, histori, pembayaran | Dasar penggunaan aplikasi |
| Admin | Mengelola buku dan pengguna | Insert, Update, Delete Data | Pengetahuan IT dasar |
| Petugas Perpustakaan | Verifikasi manual jika dibutuhkan | Validasi transaksi | Penggunaan aplikasi perangkat |
|  |  |  |  |

## Lingkungan Operasi

Aplikasi akan beroperasi dalam lingkungan sebagai berikut:

1. Platform Aplikasi:

a. Android OS (versi minimal 8.0/Oreo)

b. iOS (versi minimal iOS 13)

2. Framework Frontend: Flutter (latest stable release)

3. Backend & Database: Firebase Firestore, Firebase Authentication

4. Perangkat Keras:

a. Smartphone minimal RAM 4 GB dan penyimpanan 32 GB

b. Koneksi internet diperlukan untuk sinkronisasi data secara real-time

5. Tools Pengembangan:

a. Flutter SDK

b. Firebase Console

c. VSCode / Android Studio

## Batasan Desain dan Implementasi

Berikut batasan teknis dan kebijakan dalam pengembangan:

1. Teknologi: Harus menggunakan Flutter dan Firebase.

2. Integrasi Pembayaran: Tidak tersedia pada saat ini, akan dimasukkan di fase pengembangan berikutnya.

3. Offline Mode: Tidak didukung – aplikasi memerlukan koneksi internet aktif.

4. Multi User Support: Akan disediakan hanya untuk dua level akses (Admin dan Staff perpus).

5. Keamanan: Sistem login terbatas pada Firebase Email & Password Auth (Authentikasi).

6. Visual Branding: Belum tersedia, maka UI/UX akan bersifat minimal dan fungsional dulu.

## Dokumentasi Pengguna

1. Tutorial Onboarding (in-app): Penjelasan ringkas pertama kali membuka aplikasi.

2. FAQ Online: Daftar pertanyaan dan jawaban umum (jika diperlukan).

3. Video Tutorial (opsional): Untuk edukasi pemilik usaha dalam penggunaan aplikasi

# Kebutuhan Antarmuka Eksternal

## User Interfaces

User interface berada di figma : [Link UI](https://www.figma.com/design/jX8rLpd1cknpKaZ38NE15M/Perpus?node-id=0-1&t=Db4nRtiLtoNTIkch-1)

## Hardware Interface

1. Android minimum API 23+
2. Kamera untuk scan QR (opsional)
3. Koneksi internet aktif

## Software Interface

1. Flutter SDK ≥ 3.3
2. Firebase (Auth, Firestore, Messaging)

## Communication Interface

<Describe the requirements associated with any communications functions required by this product, including e-mail, web browser, network server communications protocols, electronic forms, and so on. Define any pertinent message formatting. Identify any communication standards that will be used, such as FTP or HTTP. Specify any communication security or encryption issues, data transfer rates, and synchronization mechanisms.>

# Functional Requirement

| **ID** | **Kebutuhan Fungsional** | **Penjelasan** |
| --- | --- | --- |
| F01 | Autentikasi Pengguna | Login/register via email/password |
| F02 | Pencarian Buku | Pencarian berdasarkan judul/penulis/kategori |
| F03 | Peminjaman Buku | Tambah ke keranjang lalu konfirmasi pinjam |
| F04 | Pengembalian/Perpanjangan | Pengajuan pengembalian atau perpanjangan buku |
| F05 | Pembayaran Denda | Upload bukti pembayaran denda |
| F06 | Notifikasi | Push notif untuk jatuh tempo/denda |
| F07 | Manajemen Buku (admin) | Tambah/edit/hapus buku |
| F08 | Riwayat Transaksi | Melihat semua aktivitas pinjam & denda |

## Use Case Diagram

<*Gambarkan use case diagramnya dari functional requirement yang didapatkan*>

## Nama Use Case 1

4.1.1 Deskripsi Use Case

<desripsikan / jabarkan mengenai use case ini >

4.1.2 Stimulus and Respon

<menyediakan daftar aksi yang dilakukan oleh user dan respon dari sistem.>

|  |  |
| --- | --- |
| Action by user | Response from system |
| 1 |  |
|  | 2 |
| 3 |  |
|  | 4 .. |

4.1.4 *Activity Diagram*

## Nama Use Case 2

<Sama seperti di atas, dan seterusnya sesuai jumlah use case yang didapatkan>

## Class Diagram

<*identifikasi kelas yang terkait dan hubungannya pada sistem yang dikembangkan*>

# Non Functional Requirements

<*Uraikan dengan ringkas kebutuhan non fungsional dalam tabel sebagai berikut. Isilah Kolom Kebutuhan dengan kalimat yang jelas dan kelak dapat ditest untuk dipenuhi. ID adalah nomor kebutuhan yang harus ditelusuri pada saat test. Tuliskan N/A bila Not Applicable>*

| **ID** | **Parameter** | **Kebutuhan** |
| --- | --- | --- |
|  | Availability |  |
|  | Reliability |  |
|  | Ergonomy |  |
|  | Portability |  |
|  | Memory |  |
|  | Response time |  |
|  | Safety | N/A |
|  | Security |  |
|  |  |  |
|  | Others 1: Bahasa komunikasi | Misalnya : semua tanya jawab harus dalam bahasa Indonesia |
|  |  | Setiap layar harus mengandung logo PT Pos Indonesia |
|  |  |  |

Catatan :

*Availability : ketersediaan aplikasi, misalnya harus terus menerus beroperasi 7 hari perminggu, 24 jam per haritanpa gagal*

*Reliability : keandalan, misalnya tidak pernah boleh gagal(atau kegagalan yang ditolerir adalah …%) sehingga harus dipikirkan fault tolerant architecture. Biasanya hanya perlu untuk Critical Application yang jika gagal akan berakibat fatal.*

*Ergonomy : kenyamanan pakai bagi pengguna*

*Portability : kemudahan untuk dibawa dan dioperasikan ke mesin/sistem operasi/platform yang lain*

*Memory : jika perhitungan kapasitas memori internal kritis (misalnya untuk SW yang harus dijadikan CHIPS dan ukurannya harus kecil*

*Response time : Batasan waktu yang harus dipenuhi. Sangat penting untuk aplikasi Real Time. Contoh: “Aaplikasi harus mampu menampilkan hasil dalam 4 detik”, atau “ATM harus menarik kembali kartu yang tidak diambil dalam waktu 3 menit”*

*Safety: yang menyangkut keselamatan manusia, misalnya untuk SW yang dipakai pada sistem kontrol di pabrik*

*Security : aspek keamanan yang harus dipenuhi*