

1. Pendahuluan

1.1 Latar Belakang

Manajemen keuangan pribadi adalah aspek penting dalam kehidupan sehari-hari. Banyak individu mengalami kesulitan dalam mengelola pengeluaran, pendapatan, dan tabungan mereka secara sistematis. Oleh karena itu, diperlukan **Sistem Pencatatan Keuangan Pribadi** yang membantu pengguna dalam memantau dan mengelola keuangan mereka dengan mudah.

Dalam proyek ini, mahasiswa akan merancang dan membuat **prototipe high-fidelity untuk sistem pencatatan keuangan pribadi** menggunakan **Figma dan JavaFX 2.x**. Sistem ini akan mencakup fitur pencatatan transaksi, kategori pengeluaran, serta visualisasi sederhana untuk membantu pengguna memahami pola keuangan mereka..

Pengelolaan keuangan pribadi yang buruk dapat menyebabkan masalah seperti hutang yang menumpuk, kesulitan mencapai tujuan finansial, dan stres akibat ketidakpastian keuangan. Oleh karena itu, kebutuhan akan alat yang praktis dan mudah digunakan untuk membantu individu mengelola keuangan mereka semakin meningkat.

Maka dari itu kami memutuskan untuk membuat **Sistem Pencatatan Keuangan Pribadi** agar mempermudah individu - individu dalam mengelola keuangan. Dengan adanya **Sistem Pencatatan Keuangan Pribadi** harapan kami keuangan keluar dan masuk dapat dikelola dengan baik agar tidak terjadi pemborosan.

1.1 Tujuan Proyek

Proyek ini bertujuan untuk:

- Mengembangkan **dokumen kebutuhan sistem** yang mencakup analisis stakeholder, use case diagram, activity diagram, dan lingkup sistem.
- Melakukan **perancangan sistem** menggunakan class diagram, Entity Relationship Diagram (ERD), dan arsitektur sistem.
- Membangun **prototype high-fidelity** menggunakan **Figma** serta mengimplementasikan UI dengan **JavaFX 2.x**, tanpa koneksi ke database.

1.2 Definisi dan Akronim

- **UI (User Interface)**: Antarmuka pengguna, yaitu tampilan visual dari suatu sistem atau aplikasi yang memungkinkan pengguna berinteraksi dengannya.
- **ERD (Entity Relationship Diagram)**: Diagram hubungan entitas, yaitu diagram yang digunakan untuk memodelkan struktur data suatu sistem, yang menunjukkan entitas-entitas dan hubungan antar entitas tersebut.

- **JavaFX 2.x:** Kerangka kerja (framework) perangkat lunak berbasis Java yang digunakan untuk membuat aplikasi desktop kaya dengan antarmuka pengguna yang interaktif.
- **High-fidelity prototype:** prototipe dengan tingkat ketelitian tinggi, yang menyerupai produk akhir dalam hal tampilan dan interaksi.

3. Kebutuhan Spesifik

3.1 Karakteristik Pengguna

- Pengguna yang sering untuk mengabaikan transaksi kecil maupun besar secara cash
- **Sistem tercatat:** Orang yang ingin melakukan pengeluaran dan pemasukan secara tercatat
- **Individu yang sibuk dan aktif:** Mereka yang memiliki banyak aktivitas dan kesulitan meluangkan waktu untuk mencatat keuangan secara manual.
- **Pelajar atau mahasiswa:** Mereka yang baru mulai belajar mengelola keuangan dan membutuhkan alat yang mudah digunakan.
- **Profesional muda:** Mereka yang mulai memiliki penghasilan tetap dan ingin mengatur keuangan mereka dengan lebih baik.

3.2 Kebutuhan Fungsional

1. Pengguna dapat menambahkan transaksi pengeluaran dan pemasukan
2. Pengguna dapat mengkategorisasi transaksi
3. Pengguna dapat melihat daftar transaksi dalam waktu tertentu
4. Pengguna dapat mengedit dan menghapus transaksi dalam rentang waktu 30 hari
5. Pengguna dapat melihat grafik ringkasan keuangan
6. Pengguna dapat menyaring berdasarkan transaksi
7. Pengguna dapat mengatur pengeluaran dalam jangka waktu yang ditentukan oleh user
8. Pengguna dapat login menggunakan no hp dan password key

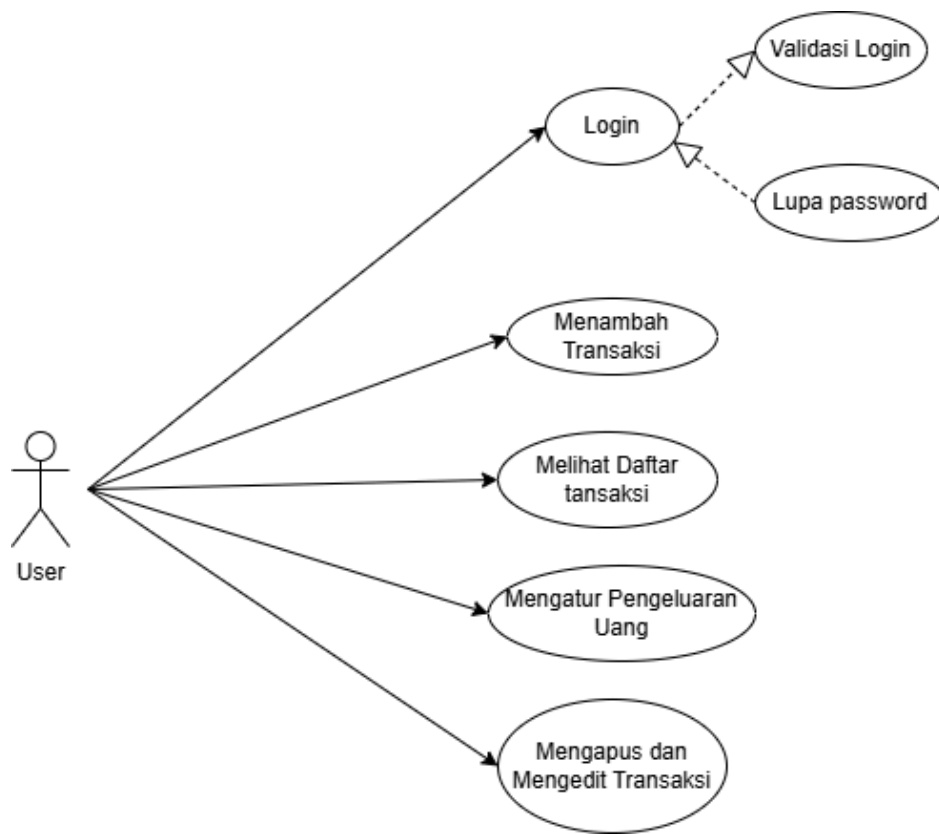
3.2 Kebutuhan Non-Fungsional

1. **Responsif:** UI harus mudah digunakan dan mendukung berbagai ukuran layar.
2. **Aksesibilitas:** Sistem harus dapat digunakan tanpa koneksi internet.
3. **Keamanan:** Data transaksi harus dijaga privasinya agar tidak dapat diakses oleh pihak lain.
4. Ukuran font dengan ukuran yang dapat dilihat
5. Manipulasi data hanya dilakukan oleh pengguna
6. Terdapat warna indikasi yang memberitahu bahwa pengeluaran sudah melebihi target
7. Dapat digunakan secara offline
8. Terdapat warna yang memberitahukan antara transaksi masuk dan keluar

4. Batasan Sistem

1. Satu akun hanya bisa digunakan satu orang
2. Tidak ada fitur administrasi orang lain
3. Aplikasi dikembangkan dengan platform desktop
4. Tidak ada mekanisme pendukung back-up
5. Setiap ingin mengakses aplikasi pengguna diharuskan untuk login

5. Diagram Use Case



5. Analisis Stakeholder

1) Identifikasi Stakeholder

Kategori Stakeholder	Stakeholder Spesifik	Peran/kepentingan
User Interface	1. Pengguna umum (individu)	Pengguna utama aplikasi untuk mencatat pengeluaran/pemasukan

		pribadi
	2. Mahasiswa / pekerja freelance	Target pasar potensial yang membutuhkan manajemen keuangan sederhana
Tim Pengembang	1. Developer (frontend/backend)	Membangun dan menguji aplikasi dan merancang antarmuka yang user-friendly
	2. Quality analyst	Memastikan aplikasi bebas bug dan sesuai kebutuhan pengguna
	3. Project manager	Mengelola timeline, anggaran dan koordinasi tim
Sponsor	1. Investor	Penyedia dana yang mengharapkan keuntungan dari keberhasilan aplikasi