

SPESIFIKASI KEBUTUHAN PERANGKAT LUNAK

<Finance Management Application>

untuk:

<nama pelanggan>

Dipersiapkan oleh:

Putri Khairamulya Ramadhini - 1301220176

Marsha Sukma Ningsih - 1301223024

Luthfiah Nazla Hidayat - 1301223363

Raisa Lukman Hakim - 1301223301

Necha Agmeyla Tama - 1301223412


Muhammad Amer Hadafi Utomo - 1301223096

Program Studi S1 Informatika

Fakultas Informatika

Universitas Telkom

2022

 UNIVERSITAS Telkom	Program Studi S1 Informatika - Fakultas Informatika	SKPL - Nomor Dokumen	Halaman
---	---	----------------------	---------

Prodi S1 Informatika - Universitas Telkom	SKPL-xxx	Halaman 1 dari 35
Dokumen ini dan informasi yang ada di dalamnya adalah milik Prodi S1 Informatika-Universitas Telkom dan bersifat rahasia. Dilarang untuk mereproduksi dokumen ini tanpa diketahui oleh Program Studi S1 Informatika, Universitas Telkom		

	Revisi	<nomor revisi>	Tgl: <isi tanggal>
--	---------------	----------------	--------------------

Daftar Perubahan

Revisi	Deskripsi
A	
B	
C	
D	
E	
F	
G	

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

Daftar Halaman Perubahan

Halaman	Revisi	Halaman	Revisi

Daftar Isi

Daftar Perubahan	2
Daftar Halaman Perubahan	3
Daftar Isi	4
1. Pendahuluan	5
1.1 Tujuan Penulisan Dokumen	5
1.2 Ruang Lingkup / Cakupan Dokumen	5
1.3 Definisi, Singkatan, dan Akronim	6
1.4 Referensi	7
2. Deskripsi Global Perangkat Lunak	8
2.1 Statement of Objective Perangkat Lunak	8
2.2 Perspektif dan Goal Perangkat Lunak	8
2.3 Profil dan Kelas Pengguna	9
2.4 Lingkungan Operasi	11
2.5 Batasan Perangkat Lunak / Sistem	13
2.6 Asumsi dan Dependensi	14
3. Deskripsi Rinci Perangkat Lunak	16
3.1 Deskripsi Kebutuhan	16
3.1.1 Kebutuhan Fungsional	16
3.1.2 Kebutuhan Non-Fungsional	18
3.2 Pemodelan Analisis	19
3.2.1 Usecase Diagram	19
3.2.1.1 Usecase Scenario #1 "ACCOUNT SIGN IN"	19
3.2.1.2 Usecase Scenario #2 "ACCOUNT LOGIN"	20
3.2.1.3 Usecase Scenario #3 <RECORD PENDAPATAN>	20
3.2.1.4 Usecase Scenario #4 <PEMBUATAN & PEMANFAATAN RENCANA ANGGARAN KEUANGAN>	21
3.2.1.5 Usecase Scenario #5 <IDENTIFIKASI PENGELUARAN>	22
3.2.1.6 Usecase Scenario #6 <LAPORAN KEUANGAN>	22
3.2.1.7 Usecase Scenario #7 <PERINGATAN PENGGUNAAN>	23
3.2.2 Class Diagram:	23
4. Kebutuhan Lain - Lain	26
4.1 Antarmuka Pengguna	26
4.2 Antarmuka Perangkat Keras	28
4.3 Antarmuka Perangkat Lunak	29
4.4 Antarmuka Komunikasi	29
4.5 Fitur sistem cerdas	32

1. Pendahuluan

1.1 Tujuan Penulisan Dokumen

Dokumen Software Requirements Specification (SRS) ini mencakup spesifikasi perangkat lunak untuk Aplikasi Manajemen Keuangan Pribadi dengan Nomor Rilis 1.0. Perangkat lunak ini akan membantu mengatur dan melacak keuangan penggunanya, tujuannya adalah untuk membantu pengguna mengelola pengeluaran, menentukan anggaran dan mengidentifikasi pengeluaran penggunanya. Perangkat lunak pengatur keuangan ini digunakan untuk mencatat semua transaksi untuk mempermudah pengelolaan keuangan. Perangkat membantu pengguna untuk merencanakan keuangan jangka panjang dan potensi penghematan uang dan membantu membuat keputusan keuangan yang lebih baik. Produk ini mencakup fitur-fitur utama seperti Manajemen Akun Pengguna, Pencatatan Transaksi Keuangan, Perencanaan dan Pemantauan Anggaran, Pengidentifikasian Pengeluaran, Pelaporan Keuangan, dan Peringatan Anggaran. Aplikasi ini dirancang untuk beroperasi pada berbagai platform termasuk Windows, macOS, dan Android, memastikan ketersediaan kepada berbagai jenis pengguna. Dengan ketergantungan pada API dari bank dan lembaga keuangan eksternal, aplikasi ini menyediakan fungsionalitas yang luas untuk membantu individu, seperti mahasiswa, pemilik bisnis, dan keluarga, mengelola keuangan pribadi mereka dengan lebih efektif.

1.2 Ruang Lingkup / Cakupan Dokumen

Dalam dokumen Spesifikasi Persyaratan Perangkat Lunak (SKPL) ini, terdapat serangkaian konvensi penulisan yang diatur untuk memastikan keterbacaan, struktur yang jelas, dan pemahaman yang optimal. Beberapa standar atau konvensi pengetikan yang diterapkan dalam SKPL ini adalah:

1. Prioritas Kebutuhan:

Setiap kebutuhan pengguna dalam perangkat lunak dibagi menjadi tiga tingkatan: primer, sekunder, dan tersier. Hal ini memberikan pemahaman yang jelas mengenai urgensi dan pentingnya setiap kebutuhan.

2. Penomoran Kebutuhan:

Setiap kebutuhan diberikan nomor referensi untuk memudahkan identifikasi dan organisasi. Penomoran yang berbeda diberikan untuk kebutuhan utama dan kebutuhan rincian, memfasilitasi pemahaman struktur kebutuhan secara hierarkis.

3. Deskripsi Kebutuhan:

Setiap kebutuhan dilengkapi dengan deskripsi kepentingan yang jelas dan komprehensif. Hal ini membantu dalam pemahaman mengapa suatu kebutuhan diperlukan, serta tingkat keperluannya. Deskripsi yang rinci memastikan pemahaman yang mendalam tentang setiap elemen kebutuhan.

<i>Prodi SI Informatika - Universitas Telkom</i>	<i>SKPL-xxx</i>	<i>Halaman 5 dari 35</i>
<i>Dokumen ini dan informasi yang ada di dalamnya adalah milik Prodi SI Informatika-Universitas Telkom dan bersifat rahasia. Dilarang untuk mereproduksi dokumen ini tanpa diketahui oleh Program Studi SI Informatika, Universitas Telkom</i>		

4. Highlight atau Warna:

Tingkatan kebutuhan diberikan warna yang berbeda sesuai dengan prioritasnya. Pemberian warna juga diterapkan pada sisa saldo dan total pengeluaran untuk memberikan penekanan visual terhadap informasi yang krusial. Hal ini memudahkan pembaca dalam mengidentifikasi dan memahami urgensi setiap elemen.

5. Font:

Font yang digunakan dalam dokumen ini adalah Arial. Pemilihan font yang konsisten membantu dalam memberikan kejelasan dan keterbacaan dokumen.

6. Perincian Kebutuhan:

Setiap kebutuhan diperinci dengan detail yang mencakup aspek-aspek tertentu. Hal ini meningkatkan kejelasan dan menghindari interpretasi ganda terhadap kebutuhan tersebut.

7. Header dan Footer:

Setiap halaman dokumen mencakup header yang mencantumkan informasi penting seperti nama proyek, nomor revisi, dan nama penulis proyek. Footer mencantumkan tanggal revisi, memberikan konteks waktu terhadap perkembangan dokumen.

1.3 Definisi, Singkatan, dan Akronim

Perangkat lunak yang dispesifikasikan dalam dokumen SKPL ini adalah Aplikasi Manajemen Keuangan Pribadi. Tujuan utama dari perangkat lunak ini adalah membantu pengguna dalam mengelola keuangan pribadi mereka dengan lebih efektif. Dengan fokus pada pencatatan transaksi keuangan, perencanaan anggaran, dan identifikasi pola pengeluaran, aplikasi ini bertujuan untuk memberikan alat yang kuat bagi pengguna agar dapat membuat keputusan keuangan yang lebih baik.

Manfaat yang relevan dari perangkat lunak ini mencakup:

1. Pencatatan Transaksi:

Mempermudah pengguna dalam mencatat semua transaksi keuangan mereka, baik pendapatan maupun pengeluaran, untuk menciptakan pemahaman yang lebih baik tentang situasi keuangan mereka.

2. Perencanaan Anggaran:

Memungkinkan pengguna untuk menetapkan dan memantau anggaran keuangan, memberikan visibilitas terhadap alokasi dana dan membantu menghindari pengeluaran yang tidak terduga.

3. Identifikasi Pengeluaran:

Membantu pengguna dalam mengenali dan mengelompokkan jenis pengeluaran, memberikan wawasan yang lebih baik tentang pola pengeluaran dan memungkinkan untuk membuat perubahan yang sesuai.

<i>Prodi S1 Informatika - Universitas Telkom</i>	<i>SKPL-xxx</i>	<i>Halaman 6 dari 35</i>
<i>Dokumen ini dan informasi yang ada di dalamnya adalah milik Prodi S1 Informatika-Universitas Telkom dan bersifat rahasia. Dilarang untuk mereproduksi dokumen ini tanpa diketahui oleh Program Studi S1 Informatika, Universitas Telkom</i>		

Aplikasi ini juga sejalan dengan tujuan perusahaan atau strategi bisnis dengan menyediakan solusi yang dapat meningkatkan kesejahteraan finansial pengguna. Dengan memberdayakan individu untuk mengelola keuangan pribadi mereka, perusahaan dapat mencapai tujuan strategis seperti peningkatan kepuasan pelanggan, membangun reputasi sebagai penyedia solusi finansial yang andal, dan berkontribusi pada perkembangan keuangan yang berkelanjutan.

Sasaran aplikasi ini melibatkan berbagai kelompok pengguna, termasuk mahasiswa, pemilik bisnis, dan keluarga. Dengan menyediakan fitur-fitur yang relevan untuk setiap kelompok ini, perangkat lunak ini diharapkan dapat menjadi alat yang berguna dan bermanfaat dalam konteks keuangan pribadi mereka. Dengan memahami visi dan lingkup dokumen, pengguna dapat lebih baik mengimplementasikan dan memanfaatkan fungsionalitas yang disediakan oleh aplikasi ini sesuai dengan kebutuhan dan tujuan mereka.

1.4 Referensi

Hasan, S., Elpisah, E., Sabtohadhi, J., Nurwahidah, M., Abdullah, A., & Fachrurazi, F. (2022). Manajemen keuangan. Penerbit Widina.

Musthafa, H., & SE, M. (2017). Manajemen keuangan. Penerbit Andi.

<i>Prodi S1 Informatika - Universitas Telkom</i>	<i>SKPL-xxx</i>	<i>Halaman 7 dari 35</i>
<i>Dokumen ini dan informasi yang ada di dalamnya adalah milik Prodi S1 Informatika-Universitas Telkom dan bersifat rahasia. Dilarang untuk mereproduksi dokumen ini tanpa diketahui oleh Program Studi S1 Informatika, Universitas Telkom</i>		

2. Deskripsi Global Perangkat Lunak

2.1 Statement of Objective Perangkat Lunak

Perangkat lunak ini adalah produk mandiri yang bertujuan untuk membantu pengguna mengelola keuangan pribadi mereka. Ini tidak merupakan bagian dari produk lainnya atau pengganti sistem yang sudah ada.

Komponen utama dari perangkat lunak ini meliputi:

1. Manajemen Akun Pengguna: Fitur ini memungkinkan individu untuk membuat akun dan masuk ke dalam sistem untuk dapat menggunakan fitur-fitur dari perangkat lunak.
2. Pencatatan Transaksi Keuangan: Pengguna dapat mencatat semua transaksi keuangan mereka, seperti pendapatan dan pengeluaran, guna melacak semua aktivitas keuangan yang terjadi.
3. Perencanaan dan Pemantauan Anggaran: Perangkat lunak ini dilengkapi dengan alat-alat yang memungkinkan individu untuk membuat dan memantau anggaran keuangan mereka. Pengguna dapat menetapkan tujuan anggaran dan memonitor kemajuannya.
4. Pengidentifikasian Pengeluaran: Fitur ini membantu pengguna dalam mengelompokkan dan mengidentifikasi berbagai jenis pengeluaran, sehingga memudahkan untuk memahami pola pengeluaran.
5. Pelaporan Keuangan: Pengguna dapat menghasilkan laporan keuangan yang memberikan wawasan mendalam mengenai aktivitas keuangan mereka dan bagaimana kinerja anggaran mereka.
6. Peringatan Anggaran: Perangkat lunak ini memiliki kemampuan untuk memberi notifikasi kepada pengguna ketika mereka mendekati atau melebihi batas anggaran atau tujuan keuangan yang telah ditetapkan.

Perangkat lunak ini tidak memerlukan ketergantungan terhadap sistem atau antarmuka eksternal lainnya, karena dirancang sebagai produk mandiri untuk mengelola keuangan pribadi. Pengguna dapat berinteraksi dengan perangkat lunak melalui antarmuka pengguna yang disediakan, yang memungkinkan untuk melakukan berbagai tindakan seperti membuat akun, melakukan login, mencatat transaksi, merencanakan anggaran, mengidentifikasi pengeluaran, menghasilkan laporan keuangan, dan menerima notifikasi apabila mendekati atau melebihi anggaran yang telah ditetapkan.

<i>Prodi S1 Informatika - Universitas Telkom</i>	<i>SKPL-xxx</i>	<i>Halaman 8 dari 35</i>
<i>Dokumen ini dan informasi yang ada di dalamnya adalah milik Prodi S1 Informatika-Universitas Telkom dan bersifat rahasia. Dilarang untuk mereproduksi dokumen ini tanpa diketahui oleh Program Studi S1 Informatika, Universitas Telkom</i>		

2.2 Perspektif dan Goal Perangkat Lunak

Model Goal untuk Perangkat Lunak Manajemen Keuangan Pribadi:

1. Goal Utama:

- Deskripsi: Menciptakan platform yang memungkinkan pengguna mengelola keuangan pribadi mereka dengan efektif dan efisien.
- Korelasi Kebutuhan Fungsional: Pencatatan transaksi, perencanaan anggaran, identifikasi pengeluaran, pelaporan keuangan, dan peringatan anggaran.

2. Goal Sekunder:

- Deskripsi: Memberikan pengguna informasi yang jelas dan terperinci tentang situasi keuangan mereka, termasuk tren pengeluaran, sisa saldo, dan kemajuan terhadap anggaran yang ditetapkan.
- Korelasi Kebutuhan Fungsional: Tampilan laporan keuangan yang komprehensif dan real-time, pengelompokan pengeluaran, dan pemberian warna pada sisa saldo dan total pengeluaran.

3. Goal Penggunaan yang Mudah:

- Deskripsi: Menyediakan antarmuka pengguna yang ramah pengguna dan intuitif agar pengguna dapat dengan mudah mengakses dan menggunakan fitur-fitur aplikasi.
- Korelasi Kebutuhan Fungsional: Desain antarmuka pengguna yang sederhana dan navigasi yang mudah.

4. Goal Keamanan Data:

- Deskripsi: Menjamin keamanan data pengguna, terutama informasi keuangan yang sangat sensitif.
- Korelasi Kebutuhan Fungsional: Implementasi kebijakan keamanan data, enkripsi transmisi data, dan autentikasi pengguna.

5. Goal Relevansi Bisnis:

- Deskripsi: Menyediakan solusi yang relevan dengan kebutuhan dan tujuan bisnis, mendukung visi perusahaan dalam meningkatkan kesejahteraan finansial pengguna.
- Korelasi Kebutuhan Fungsional: Integrasi dengan API bank dan lembaga keuangan, pelacakan bisnis dan pengeluaran pribadi.

Pemodelan goal ini membantu memetakan tujuan utama dan sekunder dari perangkat lunak ke dalam kebutuhan fungsional yang akan diimplementasikan. Setiap goal dapat diartikan sebagai kebutuhan tingkat tinggi yang kemudian dijabarkan ke dalam spesifikasi detail, memastikan bahwa perangkat lunak benar-benar mencapai tujuan yang diinginkan oleh pengguna dan perusahaan.

<i>Prodi S1 Informatika - Universitas Telkom</i>	<i>SKPL-xxx</i>	<i>Halaman 9 dari 35</i>
<i>Dokumen ini dan informasi yang ada di dalamnya adalah milik Prodi S1 Informatika-Universitas Telkom dan bersifat rahasia. Dilarang untuk mereproduksi dokumen ini tanpa diketahui oleh Program Studi S1 Informatika, Universitas Telkom</i>		

2.3 Profil dan Kelas Pengguna

Dalam perangkat lunak Manajemen Keuangan Pribadi ini, terdapat beberapa kelas pengguna yang memiliki karakteristik, kebutuhan, dan peran yang berbeda. Setiap kelas pengguna memiliki profil dan peran unik, yang mempengaruhi kebutuhan fungsional dan persyaratan spesifik yang relevan. Berikut adalah beberapa kelas pengguna yang diantisipasi:

1. Mahasiswa:

Mahasiswa dengan pendapatan terbatas, mungkin belum memiliki pengalaman manajemen keuangan yang luas. Pengguna ini menggunakan aplikasi untuk mencatat dan mengelola pengeluaran sehari-hari, mengontrol uang kuliah, dan merencanakan pengeluaran selama masa kuliah.

Requirements:

- Desain antarmuka yang intuitif untuk memudahkan penggunaan.
- Fitur pelaporan keuangan yang sederhana dan mudah dimengerti.

2. Pemilik Bisnis:

Pengusaha atau pemilik bisnis kecil dengan kebutuhan manajemen keuangan bisnis dan pribadi. Menggunakan aplikasi untuk melacak pengeluaran bisnis, menghasilkan laporan keuangan bisnis, dan merencanakan anggaran pribadi.

Requirements:

- Integrasi dengan API untuk mendukung pelacakan transaksi bisnis.
- Fitur pemisahan antara keuangan bisnis dan pribadi.

3. Keluarga:

Pengguna dengan banyak sumber pendapatan, pengeluaran keluarga, dan kebutuhan pengelolaan keuangan bersama. Menggunakan aplikasi untuk mengelola anggaran keluarga, melacak pengeluaran bersama, dan menetapkan tujuan tabungan untuk kebutuhan keluarga.

Requirements:

- Fitur pembagian tugas atau alokasi anggaran berdasarkan peran masing-masing anggota keluarga.
- Kemampuan untuk melihat laporan keuangan keluarga secara keseluruhan.

4. Pengguna Teknis Tinggi:

Pengguna dengan tingkat keahlian teknis yang tinggi, mungkin tertarik dengan analisis data mendalam. Menggunakan aplikasi untuk mendapatkan wawasan mendalam melalui analisis data dan laporan keuangan yang rinci.

Requirements:

- Akses ke data mentah untuk analisis ekstensif.
- Fitur pengelompokan dan filterisasi data yang canggih.

5. Pengguna Berkemampuan Keuangan Rendah:

Pengguna yang mungkin tidak memiliki pengetahuan keuangan yang luas. Menggunakan aplikasi untuk mendapatkan panduan keuangan, saran pengelolaan keuangan, dan pelaporan yang mudah dimengerti.

Requirements:

Prodi S1 Informatika - Universitas Telkom	SKPL-xxx	Halaman 10 dari 35
Dokumen ini dan informasi yang ada di dalamnya adalah milik Prodi S1 Informatika-Universitas Telkom dan bersifat rahasia. Dilarang untuk mereproduksi dokumen ini tanpa diketahui oleh Program Studi S1 Informatika, Universitas Telkom		

- Fitur panduan atau tutorial interaktif.
- Notifikasi dan peringatan yang jelas untuk membantu menghindari kesalahan pengeluaran.

Melalui identifikasi kelas-kelas pengguna ini, perangkat lunak dapat disesuaikan untuk memenuhi kebutuhan beragam pengguna dengan memberikan fungsionalitas dan tampilan yang sesuai dengan profil dan peran masing-masing.

2.4 Lingkungan Operasi

< Gambarkan lingkungan di mana perangkat lunak ini akan beroperasi, termasuk platform perangkat keras, sistem operasi beserta versinya, dan setiap komponen perangkat lunak lain atau aplikasi lain yang berjalan bersamaan. Sebuah diagram sederhana yang menunjukkan komponen utama dari sistem secara keseluruhan, interkoneksi subsistem, dan antarmuka eksternal (API) dapat membantu menjelaskan bagian ini. Jika SKPL mendefinisikan komponen dari sistem yang lebih besar, hubungkan requirements sistem yang lebih besar dengan fungsi dari perangkat lunak ini dan identifikasi antarmuka (API) antara keduanya >

lingkungan operasional perangkat lunak Manajemen Keuangan Pribadi:

1. Perangkat Keras:

Kompatibel dengan berbagai perangkat keras, termasuk komputer desktop dan perangkat mobile (smartphone dan tablet).

2. Sistem Operasi:

- Dukungan untuk berbagai sistem operasi, termasuk Windows 10, macOS Big Sur, dan Android 9.0 ke atas.

3. Komponen Perangkat Lunak Utama:

- Aplikasi Manajemen Keuangan Pribadi sebagai inti sistem.
- Database untuk penyimpanan dan pengelolaan data pengguna.
- Antarmuka Pengguna (UI/UX) untuk interaksi pengguna yang intuitif.
- Modul Pelaporan Keuangan untuk menghasilkan laporan yang informatif.
- Modul Koneksi API untuk menghubungkan ke bank dan lembaga keuangan eksternal.

4. Antarmuka Pengguna:

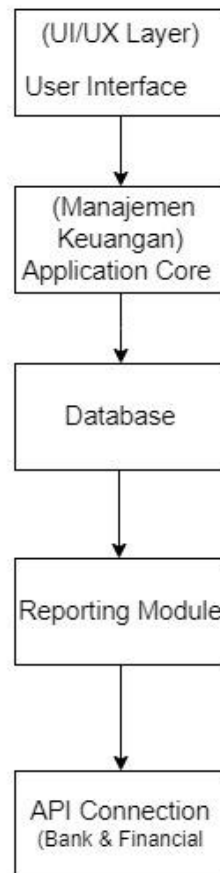
- Menyediakan akses ke berbagai fitur, termasuk pencatatan transaksi, perencanaan anggaran, laporan keuangan, dan pengaturan akun.

5. Koneksi API:

- Terhubung dengan API dari bank dan lembaga keuangan eksternal untuk mengambil informasi transaksi dan saldo.

Prodi S1 Informatika - Universitas Telkom	SKPL-xxx	Halaman 11 dari 35
Dokumen ini dan informasi yang ada di dalamnya adalah milik Prodi S1 Informatika-Universitas Telkom dan bersifat rahasia. Dilarang untuk mereproduksi dokumen ini tanpa diketahui oleh Program Studi S1 Informatika, Universitas Telkom		

Diagram Komponen Sistem:



Hubungan dengan Sistem Lebih Besar (Jika Ada):

- Aplikasi terhubung dengan sistem bank dan lembaga keuangan eksternal melalui API Connection.
- Data yang diambil dari bank dan lembaga keuangan diolah oleh Application Core untuk memberikan informasi keuangan yang terperinci.

Hubungan dengan Requirements Sistem Lebih Besar:

- Kompatibilitas dengan sistem operasi umum sesuai dengan persyaratan sistem yang lebih besar.
- Integrasi dengan API bank dan lembaga keuangan untuk mencapai konektivitas eksternal sesuai dengan persyaratan sistem yang lebih besar.

Dengan demikian, perangkat lunak ini dirancang untuk beroperasi di lingkungan yang beragam, terhubung dengan sumber daya eksternal melalui API, dan menyediakan antarmuka pengguna yang ramah pengguna untuk memenuhi kebutuhan berbagai pengguna.

2.5 Batasan Perangkat Lunak / Sistem

faktor-faktor yang dapat membatasi fungsi yang tersedia untuk para pengembang adalah sebagai berikut:

1. Kebijakan atau Peraturan Perusahaan:

- Kebijakan atau peraturan perusahaan terkait keamanan, privasi, dan standar pengembangan perangkat lunak akan membatasi opsi pengembangan. Para pengembang harus mematuhi aturan-aturan ini untuk memastikan kepatuhan dan keamanan aplikasi.

2. Keterbatasan Perangkat Keras:

- Meskipun dokumen tidak memberikan detail spesifik tentang keterbatasan perangkat keras, keterbatasan waktu dan memori bisa menjadi faktor yang mempengaruhi desain dan implementasi perangkat lunak.

3. Antarmuka untuk Aplikasi Lain:

- Jika perangkat lunak perlu berinteraksi dengan aplikasi atau sistem lain, diperlukan antarmuka yang sesuai dengan persyaratan eksternal. Kompatibilitas dengan API dari bank dan lembaga keuangan eksternal, seperti yang disebutkan dalam dokumen, akan menjadi pertimbangan.

4. Teknologi Tertentu, Alat, dan Database:

- Pilihan teknologi, alat pengembangan, dan database yang sudah ditentukan dalam dokumen dapat membatasi pilihan pengembangan. Penggunaan teknologi tertentu seperti Android untuk aplikasi mobile, serta keputusan penggunaan API untuk koneksi eksternal, dapat menjadi pembatas dalam pengembangan.

5. Requirements Bahasa:

- Meskipun dokumen tidak merinci bahasa pemrograman yang harus digunakan, keputusan terkait bahasa pemrograman akan mempengaruhi pilihan pengembangan dan implementasi.

6. Protokol Komunikasi:

- Interaksi dengan bank dan lembaga keuangan eksternal memerlukan pemahaman dan penggunaan protokol komunikasi yang sesuai. Pengembang harus memastikan kompatibilitas dan kepatuhan terhadap protokol tersebut.

7. Pertimbangan Keamanan:

- Dokumen mencakup kebijakan keamanan, dan oleh karena itu, pengembang harus memperhatikan aspek keamanan dalam semua tahap pengembangan. Pertimbangan ini dapat membatasi beberapa opsi implementasi untuk memastikan keamanan data dan informasi pengguna.

8. Konvensi Desain atau Standar Pemrograman:

- Dokumen merinci konvensi dan standar pemrograman yang harus diikuti. Para pengembang harus mematuhi standar ini untuk menjaga konsistensi dan kualitas kode.

<i>Prodi S1 Informatika - Universitas Telkom</i>	<i>SKPL-xxx</i>	<i>Halaman 13 dari 35</i>
<i>Dokumen ini dan informasi yang ada di dalamnya adalah milik Prodi S1 Informatika-Universitas Telkom dan bersifat rahasia. Dilarang untuk mereproduksi dokumen ini tanpa diketahui oleh Program Studi S1 Informatika, Universitas Telkom</i>		

9. Maintenance oleh Pengguna:

- Jika organisasi pengguna akan bertanggung jawab untuk pemeliharaan perangkat lunak, pengembang harus membangun aplikasi dengan pertimbangan pemeliharaan yang mudah dan memberikan dokumentasi yang jelas.

Dengan mempertimbangkan faktor-faktor ini, para pengembang dapat membuat keputusan yang tepat dalam pengembangan perangkat lunak sesuai dengan kebutuhan dan persyaratan yang ada.

2.6 Asumsi dan Dependensi

Berikut adalah daftar faktor-faktor yang diasumsikan yang dapat mempengaruhi requirements dalam SKPL Manajemen Keuangan Pribadi beserta identifikasi dependensi proyek pada faktor-faktor eksternal:

1. Ketersediaan API dari Bank dan Lembaga Keuangan:

- Asumsi: API dari bank dan lembaga keuangan eksternal akan selalu tersedia.
- Dependensi: Proyek sangat bergantung pada ketersediaan dan fungsionalitas API eksternal untuk mengakses data keuangan pengguna.

2. Stabilitas Koneksi Internet Pengguna:

- Asumsi: Pengguna memiliki koneksi internet yang stabil.
- Dependensi: Fungsionalitas aplikasi memerlukan ketersediaan internet, dan perubahan dalam kondisi koneksi pengguna dapat mempengaruhi pengalaman pengguna.

3. Kebijakan Privasi Pengguna:

- Asumsi: Pengguna akan memberikan izin untuk mengakses dan mencatat informasi keuangan pribadi mereka.
- Dependensi: Proyek bergantung pada kepatuhan pengguna terhadap kebijakan privasi, dan perubahan dalam kebijakan privasi dapat mempengaruhi penerimaan aplikasi oleh pengguna.

4. Ketersediaan Komponen Komersial:

- Asumsi: Komponen komersial yang direncanakan akan selalu tersedia dan sesuai dengan kebutuhan.
- Dependensi: Jika terjadi perubahan atau ketidaktersediaan komponen komersial, proyek mungkin perlu mencari solusi alternatif atau melakukan penyesuaian desain.

5. Kebijakan Keamanan dan Perlindungan Data:

- Asumsi: Kebijakan keamanan dan perlindungan data pihak ketiga sesuai dengan standar.
- Dependensi: Proyek bergantung pada kepatuhan kebijakan keamanan dan perlindungan data pihak ketiga, dan perubahan dalam kebijakan ini dapat mempengaruhi implementasi dan integrasi.

6. Penggunaan Teknologi Tertentu:

Prodi S1 Informatika - Universitas Telkom	SKPL-xxx	Halaman 14 dari 35
Dokumen ini dan informasi yang ada di dalamnya adalah milik Prodi S1 Informatika-Universitas Telkom dan bersifat rahasia. Dilarang untuk mereproduksi dokumen ini tanpa diketahui oleh Program Studi S1 Informatika, Universitas Telkom		

- Asumsi: Penggunaan teknologi tertentu, seperti Android untuk aplikasi mobile, akan tetap relevan.

- Dependensi: Proyek bergantung pada dukungan dan keberlanjutan teknologi yang dipilih, dan perubahan tren teknologi dapat mempengaruhi keberlanjutan aplikasi.

7. Ketersediaan Tenaga Pengembang:

- Asumsi: Tenaga pengembang yang diperlukan akan tersedia.

- Dependensi: Ketersediaan dan keahlian tenaga pengembang sangat mempengaruhi kemajuan proyek dan kualitas implementasi.

8. Ketersediaan Sumber Daya Server:

- Asumsi: Server yang digunakan untuk menyimpan data dan mendukung fungsionalitas aplikasi akan selalu tersedia.

- Dependensi: Ketersediaan server sangat penting untuk menjaga kinerja dan aksesibilitas aplikasi.

9. Ketersediaan dan Keandalan Hardware Pengguna:

- Asumsi: Perangkat keras pengguna memiliki ketersediaan dan keandalan yang memadai.

- Dependensi: Kondisi dan kinerja perangkat keras pengguna dapat memengaruhi pengalaman pengguna.

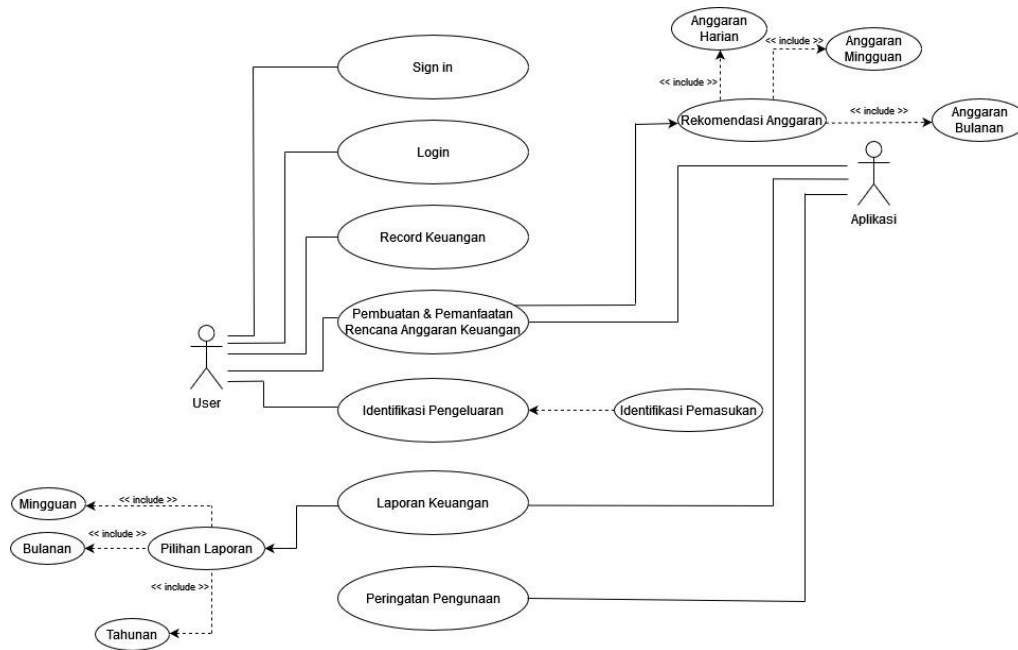
10. Tidak Ada Perubahan Signifikan dalam Kebijakan Perusahaan:

- Asumsi: Kebijakan perusahaan tidak akan mengalami perubahan signifikan.

- Dependensi: Perubahan dalam kebijakan perusahaan dapat mempengaruhi persyaratan dan integrasi aplikasi.

Menyadari dan mengelola asumsi-asumsi ini dengan baik selama siklus hidup proyek sangat penting untuk menghindari ketidakpastian dan mitigasi risiko yang mungkin muncul.

3. Deskripsi Rinci Perangkat Lunak



3.1 Deskripsi Kebutuhan

3.1.1 Kebutuhan Fungsional

No.	Kode Kebutuhan	Deskripsi	Nama Kebutuhan
1.	FR-XX	User dapat menginputkan data user untuk membuat akun Atau As User Can Menginputkan data user untuk membuat akun So Data akun user akan tersimpan di database	Sign In Akun

2.	FR-XX	User dapat menginputkan data user sesuai dengan akun yang telah dibuat Atau As User Can Menginputkan data user sesuai dengan akun yang telah dibuat So Sistem memvalidasi akun dan user dapat masuk ke aplikasi	Log In Akun
3.	FR-XX	User dapat mencatat pendapatan user Atau As User Can Menginputkan jumlah pendapatan So Jumlah pendapatan akan disimpan oleh sistem	Record Pendapatan
4.	FR-XX	User memilih jenis rencana anggaran keuangan yang ingin ditampilkan Atau As User Can Memilih jenis rencana anggaran keuangan So Sistem akan menampilkan jenis rencana anggaran sesuai pilihan user.	Pembuatan dan Pemanfaatan Rencana Anggaran Keuangan
5.	FR-XX	User menginputkan jumlah pengeluaran Atau As User Can Menginputkan jumlah pengeluaran So Jumlah pengeluaran akan disimpan oleh sistem	Identifikasi Pengeluaran
6.	FR-XX	User memilih jenis laporan keuangan yang ingin ditampilkan Atau	Laporan Keuangan

		As User Can Memilih jenis laporan keuangan So Sistem akan menampilkan jenis laporan keuangan sesuai pilihan user.	
7.	FR-XX	User menerima peringatan jika penggunaan finansial melebihi nominal rencana anggaran atau mendekati batas anggaran.	Peringatan Penggunaan

3.1.2 *Kebutuhan Non-Fungsional*

No.	Quality Criteria	Kode Kebutuhan	Deskripsi
1.	Security	NFR-SEC	Aspek ini terkait sejauh mana aplikasi dapat menerapkan enkripsi, mengontrol akses, serta membatasi akses tanpa izin.
2.	Performance	NFR-PERF	Aspek ini terkait secepat apa aplikasi dapat merespon otentikasi dan tampilan pada aplikasi.
3.	Usability	NFR-USAB	Aspek ini terkait sejauh mana aplikasi dapat dengan mudah digunakan sehingga ramah pada user.
4.	Reliability	NFR-REL	Aspek ini terkait sejauh mana aplikasi dapat menyimpan dan merepresentasikan data yang akurat.
5.	Availability	NFR-AVAIL	Aspek ini terkait sejauh mana aplikasi dapat memberi pengetahuan tepat waktu jika saldo sudah rendah atau melebihi rencana anggaran.
6.	Compatibility	NFR-COMP	Aspek ini terkait sejauh mana aplikasi dapat menyelaraskan data jika user menggunakan perangkat yang berbeda.
7.	Data Integrity	NFR-DATA	Aspek ini terkait sejauh mana aplikasi dapat memeriksa validasi data untuk menjaga keakuratan data.

3.2 Pemodelan Analisis

3.2.1 Usecase Diagram

3.2.1.1 Usecase Scenario #1 “ACCOUNT SIGN IN”

Nama Use Case	Sign In
Deskripsi	User sign in terlebih dahulu untuk membuat akun pada aplikasi
Aktor	1. User
Precondition	<ul style="list-style-type: none">- User belum memiliki akun.- User Memiliki akses internet
Postcondition	Aplikasi akan membuat akun untuk user.
Normal Flow of event	<ol style="list-style-type: none">1. User membuka aplikasi<ul style="list-style-type: none">- User memilih registrasi sign in- Aplikasi akan menampilkan halaman registrasi.2. User memasukkan username, password, email pengguna, tanggal lahir, status individu (pelajar, keluarga muda, dll).<ul style="list-style-type: none">- Sistem akan mengecek informasi yang diberikan oleh user.- Jika informasi yang diberikan oleh user itu valid maka sistem akan membuat profile user baru.3. User akan mendapatkan verifikasi melalui email yang telah diinputkan<ul style="list-style-type: none">- Email yang didapatkan user berisi link konfirmasi.4. User akan verifikasi melalui link<ul style="list-style-type: none">- Profile user akan teraktivasi
Alternative Flow of event	<ol style="list-style-type: none">1. Jika user memasukkan email yang telah memiliki akun, aplikasi akan menampilkan informasi<ul style="list-style-type: none">- Aplikasi akan menampilkan “email tersebut sudah registrasi akun”

	2. User akan submit ulang informasi 3. Hal ini akan berulang sampai user memberikan informasi data yang valid
--	--

3.2.1.2 Usecase Scenario #2 “ACCOUNT LOGIN”

Nama Use Case	Login
Deskripsi	User login untuk dapat mengakses aplikasi
Aktor	1. User
Precondition	User telah memiliki akun. User memiliki akses internet.
Postcondition	User berhasil masuk ke akunnya di aplikasi.
Normal Flow of event	1. User membuka aplikasi - User memilih log in - Aplikasi akan menampilkan halaman login. 2. User memasukkan username dan password 3. Aplikasi akan memverifikasi user agar user dapat menggunakan aplikasi. 4. User dapat mengakses fitur-fitur pada aplikasi
Alternative Flow of event	1. Jika user memasukkan username dan password yang salah, aplikasi akan menampilkan informasi - Aplikasi akan menampilkan “username atau password yang diinputkan salah” 2. User akan submit ulang informasi 3. Hal ini akan berulang sampai user memberikan informasi data yang valid

3.2.1.3 Usecase Scenario #3 <RECORD PENDAPATAN>

Nama Use Case	Record Pendapatan
Deskripsi	User dapat mencatat pendapatan mereka dalam aplikasi.
Aktor	1. User
Precondition	User sudah login ke akun nya.
Postcondition	Pendapatan keuangan tersimpan dalam aplikasi.
Normal Flow of event	<ol style="list-style-type: none">1. User masuk ke dalam aplikasi.2. User memilih menu "Record Pendapatan."3. User mengisi informasi tentang pendapatan (jumlah, sumber, tanggal, dsb.).4. Sistem menyimpan catatan pendapatan tersebut.
Alternative Flow of event	Jika user tidak memasukkan informasi yang lengkap, aplikasi akan menampilkan pesan kesalahan

3.2.1.4 Usecase Scenario #4 <PEMBUATAN & PEMANFAATAN RENCANA ANGGARAN KEUANGAN>

Nama Use Case	Pembuatan & Pemanfaatan Rencana Anggaran Keuangan
Deskripsi	Aplikasi akan membuat rencana anggaran keuangan dari jumlah pendapatan yang telah diinputkan
Aktor	<ol style="list-style-type: none">1. User2. Aplikasi
Precondition	User memilih jenis rencana anggaran keuangan
Postcondition	Aplikasi menampilkan jenis rencana anggaran keuangan sesuai pilihan user

Normal Flow of event	<ol style="list-style-type: none"> 1. User masuk ke dalam aplikasi. 2. User memilih menu "Rencana Anggaran Keuangan." 3. User memilih kategori anggaran 4. Aplikasi membuat rencana anggaran dengan menentukan anggaran untuk setiap kategori pengeluaran. 5. User dapat mengikuti rencana anggaran ini dalam manajemen keuangan mereka.
Alternative Flow of event	

3.2.1.5 Usecase Scenario #5 <IDENTIFIKASI PENGELUARAN>

Nama Use Case	Identifikasi Pengeluaran
Deskripsi	Aplikasi akan menyimpan jumlah pengeluaran yang telah diinputkan
Aktor	1. User
Precondition	User akan memasukkan setiap pengeluaran
Postcondition	Pengeluaran user berhasil dicatat dalam sistem.
Normal Flow of event	<ol style="list-style-type: none"> 1. User masuk ke dalam aplikasi. 2. User memilih menu "Identifikasi Pengeluaran." 3. User mencatat pengeluaran yang baru, termasuk jumlah, tujuan pengeluaran, dan tanggal pengeluaran. 4. Aplikasi menyimpan catatan pengeluaran tersebut.
Alternative Flow of event	Jika user tidak memasukkan informasi yang lengkap, aplikasi akan menampilkan pesan kesalahan

3.2.1.6 Usecase Scenario #6 <LAPORAN KEUANGAN>

Nama Use Case	Laporan Keuangan
Deskripsi	Aplikasi akan membuat laporan keuangan dari jumlah pendapatan dan pengeluaran yang telah diinputkan
Aktor	1. Aplikasi
Precondition	User memilih jenis laporan keuangan
Postcondition	Aplikasi menampilkan jenis laporan keuangan sesuai dengan pilihan user
Normal Flow of event	<ol style="list-style-type: none">1. User masuk ke dalam aplikasi.2. User memilih menu "Laporan Keuangan."3. Aplikasi menampilkan laporan keuangan yang berisi informasi tentang pendapatan dan pengeluaran user.
Alternative Flow of event	

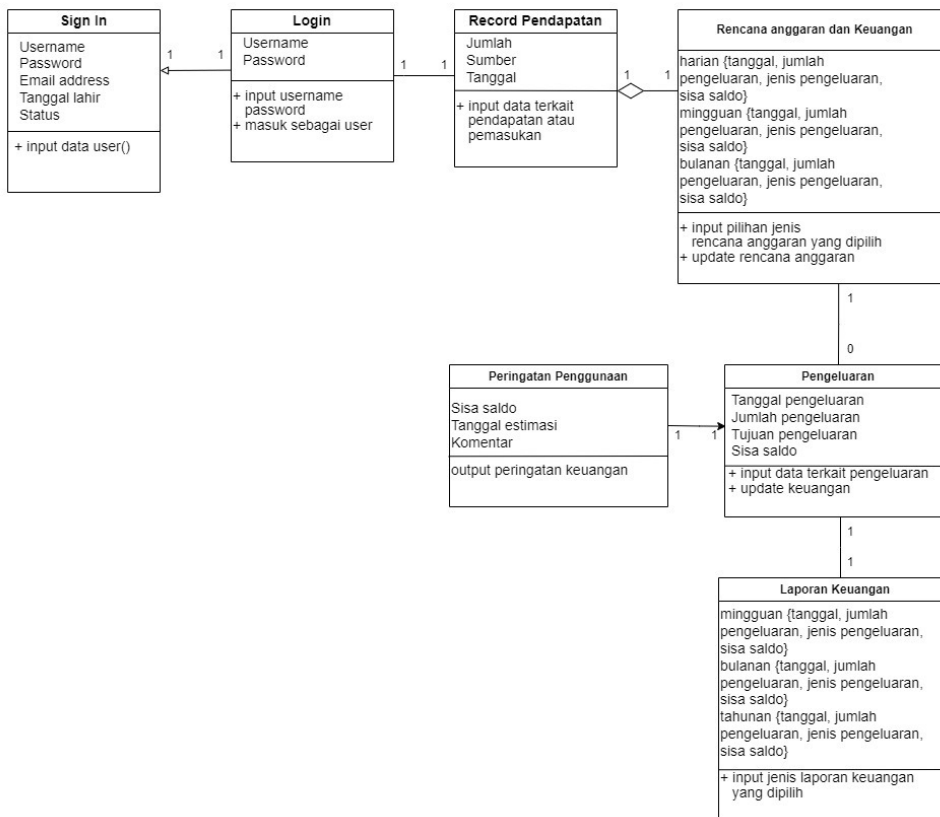
3.2.1.7 Usecase Scenario #7 <PERINGATAN PENGGUNAAN>

Nama Use Case	Peringatan Penggunaan
Deskripsi	Aplikasi memberikan peringatan kepada user tentang pengeluaran yang mendekati batas anggaran.
Aktor	1. Aplikasi
Precondition	User sudah membuat rencana anggaran dalam aplikasi.
Postcondition	User menerima peringatan jika pengeluaran mendekati atau melebihi batas anggaran yang telah ditetapkan.

Normal Flow of event	<ol style="list-style-type: none"> 1. Aplikasi secara rutin memeriksa pengeluaran user. 2. Jika pengeluaran mendekati atau melebihi batas anggaran yang telah ditetapkan, aplikasi mengirimkan peringatan kepada user.
Alternative Flow of event	

3.2.2 Class Diagram:

< Gambarkan kelas diagram model domain untuk sistem yang akan dikembangkan . Tuliskan lengkap keterhubungan antar class dengan atribut minimal dan multiplicitynya >



Berdasarkan Diagram UML:

A. Class 1: Sign In

Atribut:

- Username: String
- Password: String
- Email Address: String
- Tanggal Lahir: String
- Status: String

B. Class 2: Log In

Atribut:

- Username: String
- Password: String

C. Class 3: Record Pendapatan

Atribut:

- Jumlah: Integer
- Sumber: String
- Tanggal: String

D. Class 4: Rencana Anggaran dan Keuangan

Atribut:

- Pilihan Rencana Anggaran dan Keuangan
 1. Harian
 - Tanggal: String
 - Jumlah Pengeluaran: Integer
 - Jenis Pengeluaran: String
 - Sisa Saldo: Integer
 2. Mingguan
 - Tanggal: String
 - Jumlah Pengeluaran: Integer
 - Jenis Pengeluaran: String
 - Sisa Saldo: Integer
 3. Bulanan
 - Tanggal: String
 - Jumlah Pengeluaran: Integer
 - Jenis Pengeluaran: String
 - Sisa Saldo: Integer

E. Class 5: Pengeluaran

Atribut:

- Tanggal Pengeluaran: String
- Jumlah Pengeluaran: Integer
- Tujuan Pengeluaran: String
- Sisa Saldo: Integer

F. Class 6: Laporan Keuangan

Atribut:

1. Mingguan

- Tanggal: String
- Jumlah Pengeluaran: Integer
- Jenis Pengeluaran: String
- Sisa Saldo: Integer

2. Bulanan

- Tanggal: String
- Jumlah Pengeluaran: Integer
- Jenis Pengeluaran: String
- Sisa Saldo: Integer

3. Tahunan

- Tanggal: String
- Jumlah Pengeluaran: Integer
- Jenis Pengeluaran: String
- Sisa Saldo: Integer

G. Class 7: Peringatan Penggunaan

Atribut:

- Sisa Saldo: Integer
- Tanggal Estimasi: String
- Komentar: String

Keterhubungan antar class:

- Sign In dan Log In
Kedua class ini terhubung oleh atribut username dan password.
- Rencana Anggaran dan Keuangan dan Pengeluaran
keterhubungannya memiliki rencana pengeluaran berdasarkan rencana yang dipilih, pengeluaran merekam dan membandingkan pengeluaran dengan rencana yang sudah dibuat.
- Record Pendapatan dan Rencana Anggaran
Keterhubungannya merekam informasi pendapatan yang menjadi pertimbangan dalam rencana anggaran dan keuangan.
- Laporan Keuangan dan Rencana Anggaran dan Keuangan
Keterhubungannya menggunakan data dari rencana anggaran dan keuangan untuk membuat laporan tentang pengeluaran aktual dengan rencana yang dibuat
- Peringatan Penggunaan
Keterhubungannya menghubungkan info sisa saldo untuk memberikan peringatan terkait penggunaan dana.

4. Kebutuhan Lain - Lain

4.1 Antarmuka Pengguna

<Jelaskan kebutuhan yang terkait dengan kebutuhan UI dan UX sesuai dengan karakteristik Class Pengguna .>

1. Sign In dan Log In

- UI
Antarmuka untuk memastikan formulir input seperti `Username`, `Password`, dan `Email Address`, pada sign in. Serta memastikan `Username` dan `Password` pada log in, mudah ditemukan dan diisi oleh pengguna.
- UX
Pengalaman yang lancar saat akan masuk ke akun (Log In) dan saat membuat akun baru (Sign In). Notifikasi yang diberikan jelas ketika terjadi kesalahan dalam memasukkan informasi.

2. Record Pendapatan

- UI
Antarmuka untuk mencatat pendapatan memungkinkan pengguna untuk memasukkan informasi seperti `Jumlah Pendapatan`, `Sumber`, dan `Tanggal` dengan jelas dan mudah diidentifikasi.
- UX
Pengalaman pengguna yang mudah dalam mencatat pendapatan. Ada validasi input yang jelas.

3. Rencana Anggaran dan Keuangan:

- UI
Tampilan yang memudahkan pengguna untuk membuat rencana keuangan berdasarkan 3 opsi yaitu harian, mingguan, dan bulanan. Bagian input harus mampu menampung informasi seperti `Tanggal`, `Jumlah Pengeluaran`, `Jenis Pengeluaran`, dan `Sisa Saldo`.
- UX
Pengalaman pengguna yang efisien dalam merencanakan keuangan. Memungkinkan pengguna untuk memiliki pandangan menyeluruh atas rencana tersebut dengan menggunakan tabel informatif untuk menampilkan data.

4. Pengeluaran:

- UI
Antarmuka yang memungkinkan pengguna mencatat pengeluaran secara rinci, seperti `Tanggal Pengeluaran`, `Jumlah Pengeluaran`, `Tujuan Pengeluaran`, dan `Sisa Saldo`.
- UX
Proses pencatatan pengeluaran yang mudah dan cepat. Serta adanya validasi input yang jelas dan notifikasi terkait transaksi yang berhasil.

Prodi S1 Informatika - Universitas Telkom	SKPL-xxx	Halaman 27 dari 35
Dokumen ini dan informasi yang ada di dalamnya adalah milik Prodi S1 Informatika-Universitas Telkom dan bersifat rahasia. Dilarang untuk mereproduksi dokumen ini tanpa diketahui oleh Program Studi S1 Informatika, Universitas Telkom		

5. Laporan Keuangan:

- UI
Tampilan yang memungkinkan pengguna untuk melihat laporan keuangan secara jelas dengan 3 opsi, yaitu mingguan, bulanan, dan tahunan. Penggunaan tabel informatif untuk menampilkan data.
- UX
Pengalaman yang memudahkan pengguna memahami laporan keuangan. Kemampuan untuk menjelajahi data dengan mudah dan komprehensif.

6. Peringatan Penggunaan:

- UI
Tampilan notifikasi atau pemberitahuan yang menarik perhatian pengguna terkait status `Sisa Saldo` yang telah ditentukan.
- UX
Pengalaman yang membantu pengguna untuk memantau penggunaan dana dan memberikan peringatan jika pengeluaran seseorang lebih dari rencana yang telah dibuat.

4.2 Antarmuka Perangkat Keras

1. Koneksi ke Server:

- Deskripsi: Aplikasi memerlukan koneksi yang stabil dengan server untuk menyimpan dan mengambil data pengguna.
- Kebutuhan:
 - Komunikasi dengan server harus menggunakan protokol yang aman seperti HTTPS.
 - Aplikasi harus memiliki mekanisme koneksi ulang jika terjadi gangguan atau putus sambung.

2. Koneksi dengan Mesin Pencetak:

- Deskripsi: Aplikasi memerlukan kemampuan untuk berkomunikasi dengan mesin pencetak untuk mencetak rencana anggaran dan laporan keuangan.
- Kebutuhan:
 - Dukungan untuk protokol komunikasi yang kompatibel dengan mesin pencetak.
 - Pengaturan konfigurasi yang memungkinkan pengguna memilih jenis dan model mesin pencetak.

3. Antarmuka dengan Perangkat Penyimpanan Eksternal:

- Deskripsi: Sistem dapat berinteraksi dengan perangkat penyimpanan eksternal seperti USB drive atau kartu memori.
- Kebutuhan:
 - Dukungan untuk standar penyimpanan data eksternal seperti USB Mass Storage atau USB OTG.
 - Antarmuka yang memungkinkan pengguna mentransfer data (seperti menyimpan laporan keuangan bulanan) antara perangkat penyimpanan dan aplikasi.

<i>Prodi S1 Informatika - Universitas Telkom</i>	<i>SKPL-xxx</i>	<i>Halaman 28 dari 35</i>
<i>Dokumen ini dan informasi yang ada di dalamnya adalah milik Prodi S1 Informatika-Universitas Telkom dan bersifat rahasia. Dilarang untuk mereproduksi dokumen ini tanpa diketahui oleh Program Studi S1 Informatika, Universitas Telkom</i>		

4. Komunikasi dengan Perangkat Input Eksternal:

- Deskripsi: Sistem dapat berinteraksi dengan perangkat input eksternal seperti keyboard atau mouse.
- Kebutuhan:
 - Dukungan untuk standar perangkat input yang umum digunakan.
 - Antarmuka yang memungkinkan pengguna mengkonfigurasi atau memilih perangkat input yang diinginkan.

4.3 Antarmuka Perangkat Lunak

<Jelaskan kebutuhan yang menjelaskan bagaimana sistem akan berinteraksi dengan sistem lainnya, yang merupakan satu kesatuan. Misalnya sistem tagihan akan berinteraksi dengan sistem pendaftaran pelanggan.

1. Sinkronisasi Data Pelanggan
Data user yang diinput melalui sistem pendaftaran harus secara otomatis terintegrasi dan tersinkronisasi dengan sistem. Ini mencakup informasi seperti nama user, password, tanggal lahir, alamat email, dan status pendidikan/pekerjaan.
2. Pemberian Notifikasi atau Peringatan
Sistem dapat memberikan notifikasi kepada user jika pengeluaran melebihi rencana anggaran yang telah dibuat oleh sistem.
3. Manajemen Informasi Sisa Saldo dan Pengeluaran:
Data terkait sisa saldo dan informasi pengeluaran pelanggan harus disinkronkan dengan baik antara kedua sistem agar pengelolaan keuangan pelanggan dapat diakses dan dikelola secara konsisten.
4. Keselarasan Laporan Keuangan:
Data keuangan yang dihasilkan oleh sistem perlu dapat diselaraskan dan terintegrasi dengan laporan keuangan yang dihasilkan oleh sistem lain, seperti laporan keuangan.

Prodi S1 Informatika - Universitas Telkom	SKPL-xxx	Halaman 29 dari 35
Dokumen ini dan informasi yang ada di dalamnya adalah milik Prodi S1 Informatika-Universitas Telkom dan bersifat rahasia. Dilarang untuk mereproduksi dokumen ini tanpa diketahui oleh Program Studi S1 Informatika, Universitas Telkom		

4.4 Antarmuka Komunikasi

<Jelaskan requirements yang terkait dengan proses komunikasi yang dibutuhkan produk ini, termasuk e-mail, web browser, protokol komunikasi server jaringan, formulir elektronik, dan sebagainya. Definisikan format pesan yang tepat. Identifikasi standar komunikasi apapun yang akan digunakan, seperti FTP atau HTTP. Tentukan masalah keamanan komunikasi atau enkripsi, kecepatan transfer data, dan mekanisme sinkronisasi.>

Kebutuhan Proses Komunikasi

1. Email:

Requirements:

- Sistem akan mengirim email verifikasi saat pengguna baru mendaftar.
- Pengguna akan menerima notifikasi melalui email terkait aktivitas penting seperti perubahan kata sandi.

Format Pesan:

- Pesan email harus mengandung informasi yang jelas dan tautan verifikasi yang aman.

Masalah Keamanan:

- Pengiriman email harus dienkripsi menggunakan protokol TLS untuk melindungi data pengguna.
- Verifikasi email harus memastikan penggunaan tautan yang unik dan memiliki waktu terbatas.

2. Web Browser:

Requirements:

- Sistem dapat diakses melalui berbagai browser web populer (Chrome, Firefox, Safari, dan Edge).
- Antarmuka pengguna harus dioptimalkan untuk pengalaman pengguna yang konsisten di berbagai browser.

Masalah Keamanan:

- Sistem harus menerapkan standar keamanan web seperti HTTPS untuk melindungi data pengguna selama transmisi.

3. Protokol Komunikasi Server Jaringan:

Requirements:

- Sistem akan menggunakan protokol HTTPS untuk komunikasi dengan server jaringan.
- API endpoint harus diatur dengan baik dan dokumentasi harus tersedia untuk pengembang eksternal.

Masalah Keamanan:

Prodi S1 Informatika - Universitas Telkom	SKPL-xxx	Halaman 30 dari 35
Dokumen ini dan informasi yang ada di dalamnya adalah milik Prodi S1 Informatika-Universitas Telkom dan bersifat rahasia. Dilarang untuk mereproduksi dokumen ini tanpa diketahui oleh Program Studi S1 Informatika, Universitas Telkom		

- Seluruh komunikasi antara sistem dan server jaringan harus dienkripsi menggunakan protokol TLS.

4. Formulir Elektronik:

Requirements:

- Sistem harus menyediakan formulir elektronik untuk memasukkan data, seperti pendapatan, pengeluaran, dan informasi pengguna.
- Validasi formulir harus dilakukan di sisi klien dan sisi server untuk memastikan integritas data.

Format Pesan:

- Pesan formulir harus dirancang agar mudah dipahami dan diisi oleh pengguna.

5. Kecepatan Transfer Data:

Requirements:

- Waktu respon sistem terhadap permintaan pengguna harus kurang dari 3 detik.
- Kecepatan transfer data antara frontend dan backend harus dioptimalkan untuk mengurangi waktu pemuatan halaman.

Masalah Kecepatan:

- Penggunaan teknik caching dan kompresi data untuk meningkatkan kecepatan transfer.

6. Mekanisme Sinkronisasi:

Requirements:

- Data antara perangkat yang berbeda (misalnya, antara perangkat seluler dan komputer desktop) harus disinkronkan secara otomatis.
- Perubahan data oleh satu pengguna harus terlihat secara instan oleh pengguna lain yang terkait.

Masalah Sinkronisasi:

- Menggunakan teknologi sinkronisasi data seperti WebSockets untuk menghindari keterlambatan dalam pembaruan data.

Keamanan Komunikasi:

- Komunikasi antara frontend dan backend harus dienkripsi menggunakan protokol HTTPS.
- Penggunaan token keamanan atau metode otentikasi yang aman untuk memverifikasi pengguna saat mengakses atau memodifikasi data.
- Implementasi kontrol akses yang ketat untuk memastikan hanya pengguna yang berwenang yang dapat mengakses data tertentu.

Kebutuhan Antarmuka Komunikasi

1. Antarmuka dengan Sistem Pendaftaran Pelanggan:

Requirements:

Prodi S1 Informatika - Universitas Telkom	SKPL-xxx	Halaman 31 dari 35
Dokumen ini dan informasi yang ada di dalamnya adalah milik Prodi S1 Informatika-Universitas Telkom dan bersifat rahasia. Dilarang untuk mereproduksi dokumen ini tanpa diketahui oleh Program Studi S1 Informatika, Universitas Telkom		

- Sistem harus dapat mengintegrasikan data pendaftaran pelanggan dari sistem eksternal.
- Autentikasi dan otorisasi harus diimplementasikan untuk memastikan keamanan akses data.

3. Antarmuka dengan Sistem Eksternal Lainnya:

Requirements:

- Sistem harus menyediakan API yang dapat diakses oleh sistem eksternal untuk mendapatkan data tertentu.
- Dokumentasi API harus lengkap dan mudah dipahami.

Masalah Keamanan:

- Autentikasi dan otorisasi harus diimplementasikan untuk mencegah akses yang tidak sah ke data.

4. Integrasi dengan Layanan Email Eksternal:

Requirements:

- Sistem harus dapat berintegrasi dengan penyedia layanan email eksternal untuk mengirim notifikasi dan verifikasi email.

Format Pesan:

- Format pesan email harus sesuai dengan standar email dan mendukung HTML untuk pesan yang lebih terstruktur.

Masalah Keamanan:

- Penggunaan token atau metode keamanan lainnya saat berinteraksi dengan layanan email untuk memastikan integritas dan keamanan data.

4.5 Fitur sistem cerdas

Kebutuhan untuk Meningkatkan Kecerdasan Sistem

1. Pengenalan Pola dan Klasifikasi:

- Kebutuhan:
 - Sistem harus dapat mengenali pola dari data historis pengeluaran dan pendapatan pengguna.
 - Kemampuan untuk mengklasifikasikan pola-pola tersebut ke dalam kategori tertentu, seperti jenis pengeluaran atau sumber pendapatan.

2. Prediksi Pengeluaran dan Pendapatan:

- Kebutuhan:
 - Sistem harus memiliki kemampuan untuk memprediksi pengeluaran dan pendapatan berdasarkan data historis.

Prodi S1 Informatika - Universitas Telkom	SKPL-xxx	Halaman 32 dari 35
Dokumen ini dan informasi yang ada di dalamnya adalah milik Prodi S1 Informatika-Universitas Telkom dan bersifat rahasia. Dilarang untuk mereproduksi dokumen ini tanpa diketahui oleh Program Studi S1 Informatika, Universitas Telkom		

- Penggunaan metode kecerdasan buatan, seperti model regresi atau algoritma pembelajaran mesin, untuk memberikan perkiraan yang akurat.

3. Rekomendasi Keuangan Personal:

- Kebutuhan:
 - Sistem harus dapat memberikan rekomendasi cerdas terkait pengelolaan keuangan personal.
 - Analisis terhadap pola pengeluaran dan pendapatan untuk memberikan saran terkait peningkatan efisiensi atau pengelolaan risiko keuangan.

4. Optimasi Rencana Anggaran:

- Kebutuhan:
 - Sistem harus dapat mengoptimalkan rencana anggaran berdasarkan tujuan pengguna.
 - Menggunakan algoritma optimasi untuk mengatur alokasi dana yang paling efisien sesuai dengan preferensi dan prioritas pengguna.

5. Deteksi dan Peringatan Dini:

- Kebutuhan:
 - Sistem harus dapat mendeteksi anomali atau pola yang tidak biasa dalam pengeluaran.
 - Memberikan peringatan dini kepada pengguna jika terjadi perubahan signifikan dalam perilaku pengeluaran yang dapat mempengaruhi keuangan.

6. Analisis Sentimen Keuangan:

- Kebutuhan:
 - Kemampuan untuk menganalisis sentimen keuangan berdasarkan data pengeluaran dan pendapatan.
 - Menggunakan pemrosesan bahasa alami atau teknik analisis sentimen untuk memahami sikap dan perasaan pengguna terhadap keuangan mereka.

7. Pengoptimalan Laporan Keuangan:

- Kebutuhan:
 - Sistem harus dapat menghasilkan laporan keuangan yang terstruktur dan mudah dimengerti.
 - Pemanfaatan teknik visualisasi data untuk membuat laporan yang informatif dan mudah dinavigasi.

8. Adaptabilitas dan Pembelajaran:

- Kebutuhan:
 - Sistem harus dapat belajar dari perilaku dan keputusan pengguna.
 - Mampu beradaptasi dengan perubahan preferensi atau situasi keuangan pengguna seiring waktu.

<i>Prodi S1 Informatika - Universitas Telkom</i>	<i>SKPL-xxx</i>	<i>Halaman 33 dari 35</i>
<i>Dokumen ini dan informasi yang ada di dalamnya adalah milik Prodi S1 Informatika-Universitas Telkom dan bersifat rahasia. Dilarang untuk mereproduksi dokumen ini tanpa diketahui oleh Program Studi S1 Informatika, Universitas Telkom</i>		

9. Keamanan Data dan Privasi:

- Kebutuhan:
 - Memastikan bahwa penggunaan kecerdasan buatan tetap mematuhi standar keamanan data dan privasi.
 - Mengimplementasikan kontrol akses yang ketat untuk melindungi informasi keuangan pengguna.

Kebutuhan Teknis untuk Kecerdasan Buatan:

1. Model Pembelajaran Mesin:

- Kebutuhan:
 - Menyediakan infrastruktur untuk melatih dan menerapkan model pembelajaran mesin.
 - Memastikan penggunaan algoritma dan teknik pembelajaran mesin yang sesuai dengan kebutuhan aplikasi keuangan.

2. Infrastruktur Big Data:

- Kebutuhan:
 - Mampu menangani dan menganalisis volume besar data keuangan.
 - Penggunaan infrastruktur big data untuk memproses dan menyimpan data historis dengan efisien.

3. Integrasi API Layanan Keuangan Eksternal:

- Kebutuhan:
 - Integrasi dengan API layanan keuangan eksternal untuk mengakses data pasar, suku bunga, atau informasi keuangan terkini.
 - Memastikan keakuratan informasi yang diperoleh dari sumber eksternal.

4. Keamanan Algoritma Pembelajaran Mesin:

- Kebutuhan:
 - Memastikan keamanan algoritma pembelajaran mesin untuk mencegah serangan atau manipulasi data.
 - Penerapan teknik enkripsi atau deteksi anomali untuk melindungi integritas model.

5. Keterbacaan Model dan Penjelasan:

- Kebutuhan:
 - Menyediakan mekanisme untuk menjelaskan hasil prediksi atau rekomendasi yang dihasilkan oleh model.
 - Memastikan transparansi dalam pengambilan keputusan berbasis kecerdasan buatan.

6. Kemampuan Skalabilitas:

- Kebutuhan:

Prodi S1 Informatika - Universitas Telkom	SKPL-xxx	Halaman 34 dari 35
Dokumen ini dan informasi yang ada di dalamnya adalah milik Prodi S1 Informatika-Universitas Telkom dan bersifat rahasia. Dilarang untuk mereproduksi dokumen ini tanpa diketahui oleh Program Studi S1 Informatika, Universitas Telkom		

- Kemampuan sistem untuk berkembang seiring waktu dan pertumbuhan pengguna.
- Memastikan infrastruktur dapat diskalakan untuk menangani peningkatan jumlah data dan pengguna.

7. Perawatan Model:

- Kebutuhan:
 - Memastikan model pembelajaran mesin tetap relevan dan akurat seiring waktu.
 - Melibatkan pemantauan terus-menerus dan pembaruan model secara berkala.

8. Ketersediaan dan Ketahanan:

- Kebutuhan:
 - Memastikan ketersediaan tinggi sistem untuk mengakomodasi penggunaan berkelanjutan.
 - Menyediakan cadangan dan mekanisme pemulihan dalam hal kegagalan sistem.

Lampiran A: Daftar Kata-Kata Asing

1. API (Application Programming Interface)
2. UI/UX (User Interface/User Experience)
3. FR (Functional Requirement) dan NFR (Non-Functional Requirement)
4. OS (Operating System)
5. RAM (Random Access Memory)
6. API Connection
7. Use Case
8. Server
9. Maintenance
10. Usecase Scenario
11. Class Diagram

Lampiran B: Analysis Models

<Opsional. Masukkan model analisis yang berhubungan, seperti, state-transition diagrams, flow-map, atau entity-relationship diagrams (ERD).

Catatan : Flow-map dan ERD + Skema Relasi BD (untuk aplikasi SI) dan Flow chart (untuk aplikasi non SI, misal game

>

Prodi SI Informatika - Universitas Telkom	SKPL-xxx	Halaman 35 dari 35
Dokumen ini dan informasi yang ada di dalamnya adalah milik Prodi SI Informatika-Universitas Telkom dan bersifat rahasia. Dilarang untuk mereproduksi dokumen ini tanpa diketahui oleh Program Studi SI Informatika, Universitas Telkom		