**SKPL**-xxxx

SPESIFIKASI KEBUTUHAN PERANGKAT LUNAK

<Nama Proyek>

untuk:

<nama pelanggan>

Dipersiapkan oleh:

<Nama Pelaksana Proyek>

Program Studi S1 Teknik Informatika – Fakultas Informatika

Universitas Telkom

Jalan Telekomunikasi Terusan Buah Batu, Bandung

Indonesia

|  |  |  |  |
| --- | --- | --- | --- |
| Program Studi S1 Teknik Informatika  -  Fakultas Informatika | Nomor Dokumen | | Halaman |
| *SKPL-xxx* | | 8 |
| Revisi | *<nomor revisi>* | *Tgl: <isi tanggal>* |

# Daftar Perubahan

|  |  |
| --- | --- |
| Revisi | Deskripsi |
| A | Bab 1  Bab 2  Flowmap |
| B | Context Diagram + DFD (3.2.1)  ERD (3.3)  Perbaikan.....  ..... |
| C | Deskripsi proses (3.2.2)  Deskripsi Data (3.3)  Kebutuhan antarmuka eksternal (3.1)  Kebutuhan Non Fungsional (3.4)  Perbaikan.....  ...... |
| D | Atribut Kualitas Perangkat Lunak (3.5)  Batasan Perancangan (3.6)  Matriks keterunutan (4)  Melengkapi Lampiran ()  Perbaikan.....  ........ |
| 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 |
| 18 | B  B  B  C  D |  |  |

# Daftar Isi

[Daftar Perubahan 1](#_Toc473622253)

[Daftar Halaman Perubahan 2](#_Toc473622254)

[Daftar Isi 3](#_Toc473622255)

[1. Pendahuluan 4](#_Toc473622256)

[1.1 Tujuan Penulisan Dokumen 4](#_Toc473622257)

[1.2 Konvensi Dokumen 4](#_Toc473622258)

[1.3 Cakupan Produk 4](#_Toc473622259)

[1.4 Refernsi 4](#_Toc473622260)

[2. Overall Description 4](#_Toc473622261)

[2.1 Perspektif Produk 4](#_Toc473622262)

[2.2 Fungsi Produk 5](#_Toc473622263)

[2.3 Kelas dan Karakteristik Pengguna 5](#_Toc473622264)

[2.4 Lingkungan Operasi 5](#_Toc473622265)

[2.5 Batasan Perancangan dan Implementasi 5](#_Toc473622266)

[2.6 Dokumentasi Pengguna 5](#_Toc473622267)

[2.7 Asumsi dan Dependensi 5](#_Toc473622268)

[3. Requirements Antarmuka Eksternal 6](#_Toc473622269)

[3.1 Antarmuka Pengguna 6](#_Toc473622270)

[3.2 Antarmuka Perangkat Keras 6](#_Toc473622271)

[3.3 Antarmuka Perangkat Lunak 6](#_Toc473622272)

[3.4 Antarmuka Komunikasi 6](#_Toc473622273)

[4. Model Domain 6](#_Toc473622274)

[5. Fitur Sistem (Use Cases) 7](#_Toc473622275)

[5.1 Use Case 1 7](#_Toc473622276)

[5.1.1 Nama Use Case: 7](#_Toc473622277)

[5.1.2 Tujuan: 7](#_Toc473622278)

[5.1.3 Input: 7](#_Toc473622279)

[5.1.4 Output: 7](#_Toc473622280)

[5.1.5 Skenario Utama: 7](#_Toc473622281)

[5.1.6 Prakondisi: 7](#_Toc473622282)

[5.1.7 Langkah-langkah: 7](#_Toc473622283)

[5.1.8 Pascakondisi 7](#_Toc473622284)

[5.1.9 Skenario eksepsional 1 7](#_Toc473622285)

[5.1.10 Contoh 7](#_Toc473622286)

[5.2 Use Case 2 (dst.) 7](#_Toc473622287)

[6. Requirements Nonfungsional Lainnya 7](#_Toc473622288)

[6.1 Requirements Performa 7](#_Toc473622289)

[6.2 Requirements Keselamatan 8](#_Toc473622290)

[6.3 Requirements Keamanan 8](#_Toc473622291)

[6.4 Atribut Kualitas Perangkat Lunak 8](#_Toc473622292)

[7. Requirements Lain 8](#_Toc473622293)

# Pendahuluan

## Tujuan Penulisan Dokumen

Dokumen Spesifikasi Kebutuhan Perangkat Lunak (SKPL) merupakan dokumen spesifikasi kebutuhan perangkat lunak yang akan dikembangkan. Dokumen ini digunakan oleh pengembang perangkat lunak sebagai acuan teknis pengembangan perangkat lunak pada tahap selanjutnya.

## Konvensi Dokumen

<Jelaskan setiap standar atau konvensi pengetikan yang digunakan saat menulis SKPL ini, seperti font atau highlight yang memiliki makna khusus. Misalnya, nyatakan apakah prioritas untuk requirments tingkat yang lebih tinggi diasumsikan untuk diwariskan dengan membuat requirements yang lebih rinci, atau apakah setiap pernyataan requirements memiliki prioritas tersendiri.>

## Cakupan Produk

<Berikan penjelasan singkat dari perangkat lunak yang dispesifikasikan dan tujuannya, termasuk manfaat yang relevan, tujuan, dan sasaran. Kaitkan perangkat lunak yang dibuat dengan tujuan perusahaan atau strategi bisnis. Jika visi dan lingkup dokumen terpisah tersedia, rukuk dokumen itu bukan duplikasi isinya di SKPL ini.>

## Refernsi

<Buat daftar dokumen lain atau alamat Web yang diacu SKPL ini. Termasuk panduan gaya antarmuka pengguna, kontrak, standar, sistem spesifikasi requirements, dokumen use case, atau dokumen visi dan ruang lingkup. Berikan informasi yang cukup sehingga pembaca bisa mengakses salinan setiap referensi, termasuk judul, penulis, nomor versi, tanggal, dan sumber atau lokasi.>

# Overall Description

## Perspektif Produk

<Jelaskan konteks dan asal produk yang ditentukan dalam SKPL ini. Misalnya, sebutkan apakah produk ini merupakan produk yang disengaja ada, pengganti sistem tertentu yang sudah ada, atau produk baru, atau self-contained product. 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. Sebuah diagram sederhana yang menunjukkan komponen utama dari sistem secara keseluruhan, interkoneksi subsistem, dan antarmuka eksternal (API) dapat membantu menjelaskan bagian ini. .>

## Fungsi Produk

<Rangkum fungsi utama produk yang harus diimplementasikan atau yang harus diberitahukan kepada pengguna. Rincian dari fungsi utama ini akan dituangkan lebih lanjut pada Bagian 4, sehingga hanya ringkasan tingkat tinggi yang dibutuhkan di bagian ini. Atur agar fungsi tersebut dapat dimengerti setiap pembaca dari SKPL ini. Suatu gambar yang menjelaskan pengelompkan fungsi utama dari requirements yang terkait dan bagaimana fungsi tersebut saling terhubung, seperti diagram kelas, seringkali efektif menjelaskan hal ini.>

## Kelas dan Karakteristik Pengguna

<Identifikasi kelas-kelas pengguna yang Anda antisipasi akan menggunakan produk ini. Kelas pengguna dapat dibedakan berdasarkan frekuensi penggunaan, bagian dari fungsi produk yang digunakan, keahlian teknis, tingkat keamanan atau hak istimewa, tingkat pendidikan, atau pengalaman. Gambarkan karakteristik terkait masing-masing kelas pengguna. Requirements tertentu mungkin hanya berhubungan dengan kelas pengguna tertentu. Bedakan kelas pengguna yang paling penting untuk produk ini denga orang-orang yang lebih kurang berkepentingan.>

## 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.>

## Batasan Perancangan dan Implementasi

<Jelaskan setiap item atau masalah yang akan membatasi fungsi yang tersedia untuk para developer. Hal yang mungkin termasuk dalam bagian ini antara lain: kebijakan atau peraturan perusahaan; keterbatasan perangkat keras (requirements waktu, requirements memori); antarmuka untuk aplikasi lain; teknologi tertentu, alat, dan database yang akan digunakan; operasi paralel; requirements bahasa; protokol komunikasi; pertimbangan keamanan; konvensi desain atau standar pemrograman (misalnya, jika organisasi pengguna akan bertanggung jawab untuk melakukan maintenance untuk perangkat lunak yang akan dikirimkan).>

## Dokumentasi Pengguna

<Buatkan daftar komponen dokumentasi pengguna (seperti manual pengguna, bantuan on-line, dan tutorial) yang akan disampaikan bersama dengan perangkat lunak. Identifikasi format atau standar pengiriman dokumentasi yang dikenali pengguna.>

## Asumsi dan Dependensi

<Buatkan daftar faktor-faktor yang diasumsikan (sebagai lawan dari fakta yang telah diketahui) yang dapat mempengaruhi requirements dalam SKPL ini. Hal ini dapat mencakup pihak ketiga atau komponen komersial yang Anda berencana untuk gunakan, isu seputar development atau lingkungan operasi, atau kendala yang akan dihadapi. Proyek ini bisa terpengaruh jika asumsi ini tidak benar, tidak disebarluaskan, atau berubah. Juga identifikasi dependensi yang dimiliki proyek pada faktor-faktor eksternal, seperti komponen perangkat lunak yang Anda berniat untuk gunakan kembali dari proyek lain, kecuali jika komponen tersebut sudah didokumentasikan di tempat lain (misalnya, dalam dokumen visi dan ruang lingkup atau rencana proyek).>

# Requirements Antarmuka Eksternal

## Antarmuka Pengguna

<Gambarkan karakteristik lojik dari setiap antarmuka antara produk perangkat lunak dan pengguna. Hal ini bisa termasuk gambar sampel layar, setiap standar GUI atau panduan gaya produk yang akan diikuti, kendala tata letak layar, tombol standar dan fungsi (misalnya, fungsi bantuan) yang akan muncul pada setiap layar, keyboard, standar tampilan pesan kesalahan, dan sebagainya. Definisikan komponen perangkat lunak yang diperlukan suatu antarmuka. Rincian dari desain antarmuka pengguna harus didokumentasikan dalam spesifikasi antarmuka pengguna yang terpisah.>

## Antarmuka Perangkat Keras

<Gambarkan karakteristik lojik dan fisik dari setiap antarmuka antara produk perangkat lunak dan komponen perangkat keras dari sistem. Hal ini bisa mencakup jenis perangkat yang didukung, sifat data dan kontrol interaksi antara perangkat lunak dan perangkat keras, dan protokol komunikasi yang akan digunakan.>

## Antarmuka Perangkat Lunak

<Gambarkan hubungan antara produk ini dengan komponen perangkat lunak khusus lainnya (nama dan versi), termasuk database, sistem operasi, alat, library, dan komponen komersial terintegrasi. Identifikasi item data atau pesan yang masuk ke sistem dan akan keluar dan jelaskan tujuan masing-masing. Gambarkan layanan yang dibutuhkan dan sifat komunikasi. Acu dokumen yang menggambarkan aplikasi rinci dari protokol API. Identifikasi data yang akan dibagikan diantara komponen perangkat lunak. Jika mekanisme berbagi data harus dilaksanakan dengan cara tertentu (misalnya, penggunaan area data global dalam sistem operasi multitasking), tentukan sebagai batasan implementasi.>

## 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.>

# Model Domain

*<Opsional. >*

# Fitur Sistem (Use Cases)

<Template ini menggambarkan cara mengatur requirements fungsional untuk produk berdasar fitur sistem, layanan utama yang disediakan oleh produk. Anda dapat memilih untuk mengatur bagian ini dengan use case, mode operasi, kelas pengguna, kelas objek, hirarki fungsional, atau kombinasi dari itu semua, yang dapat membuat artian yang paling logis untuk produk Anda. Anda harus menggunakan use-case diagram>

## Use Case 1

### Nama Use Case:

### Tujuan:

### Input:

### Output:

### Skenario Utama:

### Prakondisi:

### Langkah-langkah:

#### Langkah 1:

#### .

#### Langkah n:

### Pascakondisi

### Skenario eksepsional 1

### Contoh

## Use Case 2 (dst.)

# Requirements Nonfungsional Lainnya

## Requirements Performa

<Jika ada requirements performa/kinerja produk dalam berbagai keadaan, nyatakan requirements tersebut di sini dan jelaskan alasannya, untuk membantu para developer memahami maksud dan membuat pilihan desain yang cocok. Tentukan timng relationship untuk sistem real time. Buatlah requirements sespesifik mungkin. Anda mungkin perlu menyatakan requirements kinerja untuk requirements atau fitur fungsional individu.>

## Requirements Keselamatan

<Tentukan persyaratan yang berkaitan dengan kehilangan yang mungkin terjadi, kerusakan, atau kerugian yang mungkin berasal dari penggunaan produk. Definisikan perlindungan atau tindakan yang harus diambil, serta tindakan yang harus dicegah. Dengan mengacu pada setiap kebijakan atau peraturan negara tentang keamanan yang mempengaruhi desain produk atau penggunaan eksternal. Definisikan sertifikasi keselamatan yang harus dipenuhi.>

## Requirements Keamanan

<Tentukan requirements mengenai masalah keamanan atau privasi seputar penggunaan produk atau perlindungan data yang digunakan atau dibuat oleh produk. Definisikan persyaratan otentikasi identitas pengguna. Acu pada setiap kebijakan atau peraturan yang berhubungan dengan masalah keamanan yang mempengaruhi produk eksternal. Definisikan sertifikasi keamanan atau privasi yang harus dipenuhi.>

## Atribut Kualitas Perangkat Lunak

<Tentukan setiap karakteristik kualitas tambahan untuk produk yang akan menjadi penting baik pelanggan atau pengembang. Beberapa hal yang perlu dipertimbangkan adalah: adaptasi, ketersediaan, ketepatan, fleksibilitas, interoperabilitas, maintainability, portabilitas, kehandalan, usabilitas, ketahanan, testability, dan usability. Menulis ini untuk lebih spesifik, kuantitatif, dan dapat diverifikasi bila memungkinkan. Setidaknya, klarifikasi preferensi relatif untuk setiap atribut, seperti kemudahan penggunaan lebih mudah belajar.>

# Requirements Lain

<Definisikan requirments lain yang tidak tercakup di SKPL ini. Hal-hal yang mungkin termasuk requirements database, requirements internasionalisasi, requirements hukum, tujuan penggunaan kembali untuk proyek, dan sebagainya. Menambahkan bagian baru yang berkaitan dengan proyek.>

Lampiran A: Daftar Kata-Kata Sukar

<Tentukan semua requirements yang diperlukan untuk menafsirkan SKPL ini dengan benar, termasuk akronim dan singkatan. Anda mungkin ingin membuat daftar yang terpisah yang mencakup beberapa proyek atau seluruh organisasi, dan hanya mencakup istilah khusus untuk satu proyek di setiap SKPL.>

Lampiran B: Analysis Models

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