

# Spesifikasi Kebutuhan Perangkat Lunak (SKPL)

Gambaran Tugas Besar Rekayasa Perangkat Lunak

Universitas Komputer Indonesia

2017



**Dosen Pembina**

Rauf Fauzan S.Kom., M.Kom

---

## 1. Format Dokumentasi

Untuk dokumentasi, terdapat beberapa hal yang harus ada mencakup kebutuhan aplikasi di atas. Berikut susunan Bab dan subbab untuk dokumen tugas besar ini :

### DAFTAR ISI

#### BAB I PENDAHULUAN

Pendahuluan dari SKPL harus memberikan gambaran umum dari seluruh dokumen SKPL (bukan sistem perangkat lunak yang hendak dibangun). Pendahuluan SKPL harus berisi bagian-bagian berikut:

##### 1.1 Kegunaan

Bagian ini harus menunjukkan tujuan pembuatan SKPL secara umum. Uraikan pula pengguna dari dokumen SKPL ini dan dengan tujuan apa para pengguna tersebut menggunakan SKPL ini.

##### 1.2 Ruang Lingkup

Bagian ini harus:

- Mengidentifikasi produk perangkat lunak yang dispesifikasi pada dokumen ini berdasarkan nama. Contoh, "MySoft Professional versi 2.3 for Windows".
- Menjelaskan apa yang akan dilakukan dan tidak dilakukan (bila perlu) oleh perangkat lunak yang dispesifikasikan pada dokumen ini.
- Menjelaskan penerapan perangkat lunak yang dispesifikasi pada dokumen ini beserta manfaat, tujuan dan sasaran dari pembuatan perangkat lunak tersebut.
- Merujuk pada identifikasi spesifikasi yang ada di dokumen-dokumen pendahulu SKPL ini (misalnya kontrak atau spesifikasi sistem) dan apa yang diutarakan pada bagian ini (serta bagian-bagian lainnya) harus konsisten dengan dokumendokumen tersebut.

##### 1.3 Definisi, Akronim dan Singkatan

Harus memberikan penjelasan terhadap semua definisi, akronim dan singkat yang digunakan agar dapat menginterpretasikan SKPL dengan benar dan satu arti. Informasi ini dapat dibuat pada lampiran atau dokumen terpisah. Pada kasus ini, bagian ini diisi dengan rujukan ke lampiran atau dokumen yang dimaksud.

---

1.3.1 Definisi

1.3.2 Akronim

1.3.3 Singkatan

#### **1.4 Referensi**

Bagian ini harus memberikan:

- Daftar lengkap dari dokumen (baik itu berupa buku, panduan, atau spesifikasi/deskripsi lain) yang dirujuk pada dokumen SKPL ini
- Identifikasi dari setiap dokumen berdasarkan judul, nomor dokumen (bila ada), tanggal dan organisasi penerbit
- Bila perlu, sebutkan sumber-sumber atau organisasi yang dapat memberikan referensi yang dituliskan tersebut.
- Cara penyusunan referensi, sesuaikan dengan standar IEEE.

#### **1.4 Tinjauan bagian dokumen berikutnya**

Bagian ini adalah ikhtisar dari dokumen SKPL. Tuliskan sistematika pembahasan dokumen SKPL ini.

---

## BAB II DESKRIPSI UMUM

Bagian ini merupakan penjelasan tentang perangkat lunak secara umum. Dijelaskan melalui perspektif perangkat lunak relatif terhadap konteksnya, fungsi dasar perangkat lunak, karakteristik pengguna yang diarah, batasan-batasan yang mempengaruhi perangkat lunak secara umum, serta asumsi dasar yang digunakan dan kebergantungan perangkat lunak pada fenomena lain di luar perangkat lunak.

### 2.1 Prespektif Produk

Bagian ini menjelaskan posisi perangkat lunak relatif terhadap konteks sistem lain yang melingkupinya. Jika produk tidak bergantung pada sistem atau produk lain, maka harus juga dinyatakan di sini. Jika SKPL mendefinisikan perangkat lunak sebagai sebuah komponen dari suatu sistem yang lebih besar yang melingkupinya, maka bagian ini harus menghubungkan kebutuhan dari sistem yang lebih besar ini dengan fungsionalitas dari perangkat lunak yang dispesifikasikan dan harus mengidentifikasi bagaimana antarmuka antara keduanya.

Untuk mempermudah, sebuah diagram blok dapat digunakan untuk menjelaskan disertai dengan narasinya. Diagram blok sebaiknya dapat menunjukkan :

- komponen-komponen utama dari sistem yang lebih besar yang melingkupi perangkat lunak yang dispesifikasikan
- interkoneksi antara perangkat lunak yang dispesifikasikan dengan komponen/sistem lain yang melingkupinya
- antarmuka eksternal dari perangkat lunak yang dispesifikasikan tersebut

### 2.2 Fungsi Produk

Bagian ini mengutarakan fungsi-fungsi dasar sistem yang utama. Pada akhirnya fungsi-fungsi dasar sistem ini berimpitan dengan *bubble* pada *level 1* DFD. Namun sebagai panduan kasar, yang disebut sebagai fungsi dasar sistem adalah jawaban atas masalah yang hendak diselesaikan melalui pembuatan perangkat lunak ini. Kadangkadang kesimpulan dari fungsi yang diperlukan untuk bagian ini dapat diambil secara langsung dari bagian spesifikasi yang lebih tinggi (jika ada) yang akan mengalokasikan fungsi khusus dari produk perangkat lunak. Perhatikan bahwa untuk alasan kejelasan:

---

1. Fungsi harus diorganisasikan dengan suatu cara sehingga daftar fungsi dapat dimengerti oleh pengguna atau orang lain yang membaca dokumen pertama kali
2. Metode tekstual dan grafik dapat digunakan untuk menunjukkan fungsi yang berbeda dan keterhubungannya. Diagram ini tidak ditujukan untuk menunjukkan perancangan produk tetapi juga menunjukkan hubungan logik antar fungsi. Sebagai contoh SKPL untuk perangkat lunak apotek, bagian ini digunakan untuk menjelaskan secara umum tentang pengelolaan obat, penerimaan resep, pendaftaran pemasok tanpa menyebutkan kebutuhan rinci dari masing-masing fungsi tersebut.

### 2.3 Karakteristik User

Karakteristik pengguna menggambarkan siapa saja pengguna dari perangkat lunak yang dispesifikasikan dan apa saja haknya terhadap perangkat lunak tersebut. Pengguna penting disebutkan karena pada akhirnya perangkat lunak yang dibangun harus mampu menjawab tantangan kebutuhan dari pengguna yang spesifik pula. Pengungkapan karakteristik pengguna dapat dilakukan dengan menyatakannya pada sebuah tabel dengan kolom-kolom: Pengguna, Tanggung Jawab (tanggung jawabnya relatif yang berkaitan dengan perangkat lunak ini), Hak Akses (hak akses ini dihubungkan pula ke fungsi dasar sistem yang tertulis pada bagian Fungsi Produk), Tingkat Pendidikan, Tingkat Keterampilan (yang dibutuhkan), Pengalaman (yang dibutuhkan), Jenis Pelatihan (yaitu pelatihan yang dibutuhkan agar pengguna ini dapat melakukan tanggung jawabnya, sifatnya opsional hanya diisi jika dibutuhkan).

#### 1.1 Batasan – Batasan

Bagian SPKL ini berisi deskripsi umum dari item lain yang akan membatasi pilihan atau keputusan pada spesifikasi. Hal-hal tersebut antara lain:

- a. Kebijakan umum organisasi/lingkungan
- b. Keterbatasan karena perangkat keras, contohnya kebutuhan *signal timing*
- c. Standar antarmuka ke aplikasi atau sistem lain
- d. Tuntutan pengoperasian secara paralel atau *multi platform*

#### 1.2 Asumsi dan Ketergantungan

Bagian ini mengungkapkan setiap faktor yang mempengaruhi kebutuhan yang dinyatakan pada SKPL. Faktor-faktor ini bukan merupakan pembatasan atas keputusan

---

yang diambil untuk perancangan perangkat lunak, melainkan hal-hal di luar cakupan perangkat lunak yang dispesifikasikan, yang bila diubah dapat berakibat pada atau mengubah kebutuhan yang tertulis di SKPL. Sebagai contoh asumsi bahwa suatu sistem operasi akan tersedia pada suatu platform perangkat keras dari produk perangkat lunak. Jika sistem operasi tidak ada maka SKPL harus diubah karena hal tersebut.

Di bagian ini dapat pula diungkapkan prioritas pengembangan dari sejumlah fungsi dasar sistem yang telah diuraikan sebelumnya. Identifikasikan pula kebutuhan yang ditunda pengembangannya sampai versi-versi lanjut.

## **BAB III DESKRIPSI KEBUTUHAN RINCI**

### **3.1 Metode Pendekatan Dan Pengembangan Sistem**

Jelaskan tentang metode pendekatan dan pengembangan sistem yang akan digunakan. Beserta alat bantu perancangan yang digunakan.

### **3.2 Kebutuhan Fungsional**

Kebutuhan fungsional harus mendefinisikan aksi dasar yang harus diambil oleh perangkat lunak untuk menerima dan memproses masukan dan menghasilkan keluaran. Dapat dilakukan juga pembagian kebutuhan fungsional menjadi sub fungsional atau sub proses. Hal ini tidak berarti bahwa rancangan perangkat lunak akan dibagi dengan cara yang sama.

#### **3.2.1 Deskripsi Kebutuhan Fungsional**

#### **3.2.2 Diagram Konteks**

Contoh Jika Menggunakan Metode Pendekatan Terstruktur

#### **3.2.3 DFD**

##### **3.2.3.1 DFD Lev 1**

Contoh Jika Menggunakan Metode Pendekatan Terstruktur

##### **3.2.3.2 DFD Lev 2 Proses 1**

Contoh Jika Menggunakan Metode Pendekatan Terstruktur

##### **3.2.3.3 DFD Lev 2 Proses 2**

Contoh Jika Menggunakan Metode Pendekatan Terstruktur

##### **3.2.3.4 DFD Lev 2 Proses n**

Contoh Jika Menggunakan Metode Pendekatan Terstruktur

#### **3.2.4 Kamus Data**

### **3.3 Kebutuhan Antarmuka Eksternal**

#### **3.3.1 Antarmuka Pemakai**

#### **3.3.2 Antarmuka Perangkat Keras**

---

### **3.3.3 Antarmuka Perangkat Lunak**

### **3.3.4 Antarmuka Infrastruktur dan Jaringan**

## **3.4 Kebutuhan Performansi**

Bagian ini harus menspesifikasikan baik kebutuhan numerik statik/dinamik yang terletak pada interaksi perangkat lunak atau pada interaksi manusia dengan perangkat lunak secara keseluruhan. Kebutuhan numerik statis mungkin melibatkan:

1. Jumlah terminal yang didukung
2. Jumlah pengguna simultan yang didukung
3. Jumlah dan tipe informasi yang ditangani

Kebutuhan numerik statik sering diidentifikasi pada bagian terpisah yang disebut kapasitas. Kebutuhan numerik dinamik mungkin dapat melibatkan, sebagai contoh, jumlah transaksi dan tugas dan jumlah data yang akan diproses selama jangka waktu tertentu, baik kondisi normal atau kondisi beban puncak.

Semua kebutuhan ini harus dinyatakan dalam istilah yang dapat diukur. Contohnya, kalimat "95 % transaksi harus diproses dalam 1 detik", akan lebih baik daripada kalimat "operator mungkin tidak harus menunggu transaksi akan selesai."

## **3.5 Kendala Perancangan**

## **3.6 Atribut Sistem Perangkat Lunak**

Ada sejumlah atribut kualitas perangkat lunak yang dapat ditampilkan sebagai kebutuhan. Atribut yang diinginkan harus dispesifikasikan sedemikian sehingga hasilnya dapat diverifikasi. Uraian minimum pada bagian ini berisi sebuah tabel dengan kolom: Kriteria Kualitas, Tuntutan Kualitas. Butir kualitas yang dapat dipertimbangkan antara lain: keandalan (*reliability*), ketersediaan (*availability*), keamanan (*security*), keremawatan (*maintainability*), kepemindahan (*portability*). Bila diperlukan uraian khusus, dapat dilakukan dengan menguraikannya menjadi sub-bab tersendiri.

### **3.6.1 Keandalan**

Bagian ini berisi spesifikasi factor-faktor yang diperlukan untuk mencapai keandalan sistem pada saat diserahkan.

---

### **3.6.2 Ketersediaan.**

Bagian ini berisi spesifikasi factor-faktor yang diperlukan untuk menjamin tingkat ketersediaan seluruh sistem saat sistem beroperasi, seperti *checkpoint*, *recovery* dan *restart*.

### **3.6.3 Keamanan**

Bagian ini berisi faktor untuk memproteksi perangkat lunak dari akses, penggunaan, pengubahan, penghancuran atau pengungkapan (*disclosure*) yang tidak disengaja atau yang merusak. Kebutuhan yang spesifik termasuk hal-hal berikut:

1. Penggunaan teknik kriptografi
2. Penyimpanan data log/history
3. Pemberian suatu fungsi ke modul-modul yang berbeda
4. Pembatasan komunikasi terhadap suatu area tertentu dalam program
5. Pemeriksaan integritas data untuk peubah-peubah kritis

### **3.6.4 Keremawatan (*Maintainability*)**

Bagian ini menentukan atribut perangkat lunak yang berhubungan dengan kemudahan perawatan dari perangkat lunak tersebut. Atribut tersebut dapat berupa kebutuhan akan tingkat modularitas, antarmuka, kompleksitas, dan lain-lain.

Penulisan atribut

keremawatan tidak dilakukan hanya atas dasar pemikiran atas praktik perancangan yang baik saja, tetapi harus didasari pada tuntutan kondisi sistem.

### **3.6.5 Kepemindahan (*Portability*)**

Atribut dari perangkat lunak yang berhubungan dengan kemudahan pemindahan perangkat lunak ke mesin dan/atau sistem operasi lain. Atribut ini berbentuk antara lain:

1. Persentase komponen yang berisi kode yang bergantung pada *host*
  2. Persentase kode yang bergantung pada *host*
  3. Penggunaan bahasa yang kepemindahannya terbukti
  4. Penggunaan suatu kompilator tertentu atau subset bahasa tertentu
  5. Penggunaan suatu sistem operasi tertentu
-

## **LAMPIRAN**

- 1. Perancangan Basisdata (Normalisasi, ERD & Tabel Relasi)**
- 2. Uraikan Tugas Dari masing-masing Kelompok Secara Detail.**