**SKPL-xx**

SPESIFIKASI KEBUTUHAN PERANGKAT LUNAK

Reparasi Penyewaan Mobil Online

untuk:

<Nama User>

Dipersiapkan oleh:

Erik 1127050050

Ihsan Jatnika F 1127050070

JURUSAN TEKNIK INFORMATIKA

FAKULTAS SAINS DAN TEKNOLOGI

UNIVERSITAS ISLAM NEGERI SUNAN GUNUNG DJATI

BANDUNG

2014/2015

DAFTAR PERUBAHAN

|  |  |
| --- | --- |
| Revisi | Deskripsi |
| A |  |
| B |  |
| C |  |
| D |  |
| E |  |
| F |  |
| G |  |

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

Daftar Halaman Perubahan

|  |  |  |  |
| --- | --- | --- | --- |
| Halaman | Revisi | Halaman | Revisi |
|  |  |  |  |

Daftar Isi

1. Pendahuluan 5

1.1 Tujuan Penulisan Dokumen 5

1.2 Lingkup Masalah 5

1.3 Definisi, Istilah dan Singkatan 5

1.4 Aturan Penomoran 5

1.5 Referensi 5

1.6 Deskripsi umum Dokumen (Ikhtisar) 5

2 Deskripsi Umum Perangkat Lunak 6

2.1 Deskripsi Umum Sistem 6

2.2 Karakteristik Pengguna 6

2.3 Batasan 6

2.4 Lingkungan Operasi 6

3 Deskripsi Kebutuhan 7

3.1 Kebutuhan Antarmuka Eksternal 7

3.1.1 Antarmuka pemakai 7

3.1.2 Antarmuka Perangkat Keras 7

3.1.3 Antarmuka Perangkat Lunak 7

3.1.4 Antarmuka Komunikasi 7

3.2 Kebutuhan Fungsional 7

3.3 Model Use Case 7

3.3.1 Diagram Use Case 7

3.3.2 Definisi Actor 7

3.3.3 Definisi Use Case 8

3.3.4 Skenario Use Case 8

3.4 Diagram Kelas 8

3.5 Diagram Kelakuan 8

3.6 Kebutuhan Non Fungsional 9

3.7 Batasan Perancangan 9

3.8 Kerunutan (traceability) 9

3.8.1 Kebutuhan Fungsional vs Use Case 9

3.8.2 Use Case vs Kelas Terkait 10

3.9 Ringkasan Kebutuhan 10

3.9.1 Kebutuhan Fungsional 10

3.9.2 Kebutuhan Non Fungsional 10

Setelah Daftar Isi Boleh ada Daftar Tabel dan Daftar Gambar

# 1. Pendahuluan

## Tujuan Penulisan Dokumen

Dokumen ini dibuat untuk user (pengguna), menjelaskan atau menggambarkan bagaimana sistem ini bekerja, sehingga pengguna tidak merasa bingung ketika menggunakan aplikasi ini.

## Lingkup Masalah

Aplikasi Reparasi penyewaan mobil ini kami buat untuk membantu perusahaan yang bergerak pada bidang penyewaan kendaraan, sehingga diharapakan dengan dibuatnya aplikasi ini lebih memudahkan dan bisa membantu kelancaran perusahaan yang bersangkutan untuk lebih menarik perhatian masyarakat dengan menawarkan kemudahan dalam proses penyewaan mobil tanpa harus datang ke tempat perusahaan tersebut berdiri. Sistem ini sendiri hanya terbatas pada proses pemesanan kendaraan saja untuk mengantisipasi terjadinya kekosongan mobil yang terjadi akibat mobil yang disediakan oleh perusahaan sudah disewa. Aplikasi ini bisa menampilkan mobil yang bisa disewa yang menandakan bahwa kendaraan tersebut belum ada yang menyewa.

## Definisi, Istilah dan Singkatan

Semua definisi dan singkatan yang digunakan dalam dokumen ini dan penjelasannya

## Aturan Penomoran

Aturan penomoran:

Setiap use case memiliki penomoran dengan format : UC-00xx;

## Referensi

Foenadian dan Samuel Prakoso.2008.Pedoman Praktis Pengembangan Aplikasi Web Database Menggunakan JAVA Server Page. ANDI: Yogyakarta

## Deskripsi umum Dokumen (Ikhtisar)

Tuliskan sistematika pembahasan dokumen SKPL ini.

# Deskripsi Umum Perangkat Lunak

## Deskripsi Umum Sistem

Tuliskan overview P/L, dalam bentuk gambar dan narasi yang dapat memberikan gambaran tentang aplikasi dan konteksnya, yaitu hubungannya dengan dunia luar (gambar yang mirip dengan diagram konteks, tetapi dengan notasi yang lebih mudah dimengerti orang awam).

## Karakteristik Pengguna

| **Kategori Pengguna** | **Tugas** | **Hak Akses ke aplikasi** |
| --- | --- | --- |
| Penyewa | Mengisi data penyewaan | Tidak penuh |
| Admin | Mengisi dan mengatur data pada system | Penuh |
| Petugas (operator) | Memvalidasi data penyewaan | Tidak penuh |

## Batasan

Batasan (jika ada), ketergantungan SW terhadap SW/HW/sistem lain (misalnya modul Konsolidasi baru dapat dijalankan ketika rekapitulasidata akuntansi dari Aplikasi AKUNT sudah dijalankan dan datanya dinyatakan OK oleh petugas

Batasan yang harus dipakai. Misalnya :

* harus memakai file data dari Sistem lain (sebutkan),
* harus memakai format data yang sama dengan sistem lain
* harus berfungsi multi platform (di Windows dan linux)

## Lingkungan Operasi

Operating system, DBMS, ...

*Aplikasi Client server ini akan berfungsi dengan spesifikasi :*

*Server : ???*

*Client : ????*

*OS :*

*DBMS :*

# Deskripsi Kebutuhan

## Kebutuhan Antarmuka Eksternal

Hanya diisi jika P/L memerlukan fasilitas khusus .

### Antarmuka pemakai

* Keyboard

Keyboard ini dipakai untuk menuliskan user name, password, serta data-data yang user ingin inputkan ke system.

* Mouse

Mouse digunakan oleh user untuk memilih menu yang ada pada system.

* Modem
* Modem digunakan untuk menghubungkan user ke internet. Hal ini dikarenakan aplikasi yang dibuat berbasis web.
* Monitor

Monitor berguna untuk menampilkan tampilan dari aplikasi.

### Antarmuka Perangkat Lunak

Hanya diisi jika PL memakai interface (berupa PL), misalnya API Windows.

### Antarmuka Komunikasi

Hanya diisi jika PL beroperasi di jaringan dan membutuhkan alat komunikasi khusus, misalnya RS232.

## Kebutuhan Fungsional

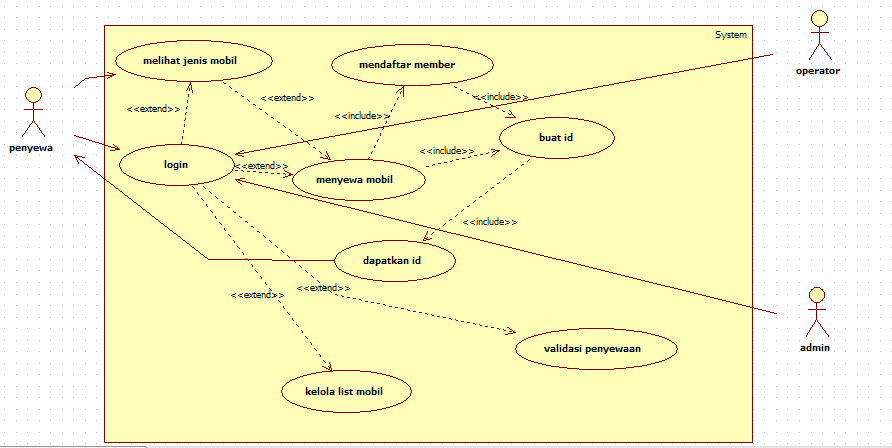
Diawali dengan membuat daftar kebutuhan fungsional P/L, lengkap dengan ID dan penjelasan jika perlu. Bisa dibuat dalam bentuk tabel.

| **ID** | **Kebutuhan** | **Penjelasan** |
| --- | --- | --- |
| FR-001 |  |  |
| FR-002 |  |  |
| FR-003 |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

Pada subbab berikutnya, buatlah diagram konteks dan DFD level berikutnya.

## Model Use Case

### Diagram Use Case



### Definisi Actor

Bagian ini diisi dengan daftar actor dan deskripsi role untuk actor tersebut. Deskripsi role harus menjelaskan wewenang pada role tersebut dalam perangkat lunak. Bisa dibuat dalam bentuk tabel berikut:

|  |  |  |
| --- | --- | --- |
| *No* | *Actor* | *Deskripsi* |
| *1* | *Admin* | *Actor dengan role ini mempunyai wewenang untuk melakukan registrasi pegawai baru dan memperbarui data mobil yang bisa disewa jika ada tambahan mobil dari pihak ke tiga* |
| *2* | *Operator* | *Operator atau disini seagai petugas di tempat perusahaan berdiri memiliki wewenang untuk memvalidasi data penyewaan yang sebelumnya diberikan kepada penyewa ketika proses pemesanan kendaraan telah dilakukan.* |
| *3* | *Penyewa (pelanggan)* | *Aktor ini hanya memiliki wewenng untuk memillih mobil itupun jika sudah menjadi member. Mengisi biodata.* |

### Definisi Use Case

Bagian ini diisi dengan daftar use case dan deskripsi singkat mengenai use case tersebut. Bisa dibuat dalam bentuk tabel berikut:

|  |  |  |
| --- | --- | --- |
| *No* | *Use Case* | *Deskripsi* |
| *UC-001* | *Memilih jenis mobil* | *Sistem menampilkan daftar mobil yang bisa di pilih oleh penyew.* |
| *UC-002* | *Login* | *Sistem meminta memasukan username beserta password* |
| *UC-003* | *Menyewa mobil* | *Sistem menerima data penyewaan mulai dari jenis mobil yang dipesan,tanggal mobil akan di ambil, jumlah harga yang harus dibayar.* |
| *UC-004* | *Mendaftar member* | *Sistem menerima pendaftaran member baru* |
| *UC-005* | *Buat id* | *Sistem otomatis memberikan id, baik itu id member dan id penyewaan.* |
| *UC-006* | *Dapatkan id* | *Sistem mengambil id untuk user* |
| *UC-007* | *Kelola list mobil* | *Sistem memberikan hak kepada admin untuk mengupdate (mengubah) list mobil yang dapat disewa* |
| *UC-008* | *Validasi penyewaan* | *Sistem bisa mengecek penyewaan dengan cara petugas menginputkan id penyewaan dan id member.* |

### Skenario Use Case

*Nama Use Case: memilih jenis mobil*

*Skenario:*

|  |  |
| --- | --- |
| ***Aksi Actor*** | ***Reaksi Sistem*** |
| *Skenario Normal* |  |
| *1. Memilih menu Daftar mobil* |  |
|  | *2. Menampilkan daftar produk dari basisdata ke*  *layar* |
| *3. Menekan tombol navigasi (next, prev)* |  |
|  | *4. Me-refresh tampilan daftar produk* |
| *Skenario Alternatif* |  |
| *1. Memilih menu Daftar mobil* |  |
|  | *2. Menampilkan pesan ‘Tidak ada mobil’* |

*Nama use case : Login*

*Use case scenario*

|  |  |
| --- | --- |
| ***Aksi Actor*** | ***Reaksi Sistem*** |
| *Skenario Normal* |  |
| *1. Memilih menu login* |  |
|  | *2. Menampilkan form login* |
| *3. memasukan username serta password* |  |
|  | *4. menampilkan pesan login sukses dan tampilkan menu bedasarkan user yang masuk.* |
| *Skenario Alternatif* |  |
| *1. Memilih menu login* |  |
|  | *2. Menampilkan form login* |
| *3. memasukan username serta password* |  |
|  | *4. menampilkan pesan login gagal.* |

*Nama use case : Menyewa mobil*

*Use case scenario*

|  |  |
| --- | --- |
| ***Aksi Actor*** | ***Reaksi Sistem*** |
| ***Skenario Normal*** |  |
| *Pre kondisi: penyewa sudah login* |  |
|  | 1. *Menampilkan form penyewaan* |
| 1. *Mengisi form penyewaan* |  |
| 1. *Menekan tombol sewa* |  |
|  | 1. *Menyimpan data penyewaan ke database.* |
| ***Skenario Alternatif*** | |
| *pre kondisi : penyewa sudah login* | |
| 1. *Membatalkan penyewaan.* 2. *Mobil yang di pilih sudah disewa (tidak tersedia)* | |

*Nama use case : mendaftar member*

*Use case skenario*

|  |  |
| --- | --- |
| ***Aksi Actor*** | ***Reaksi Sistem*** |
| ***Skenario Normal*** |  |
| *Pre kondisi: penyewa memasuki halaman utama web* |  |
| 1. *Klik buat menu daftar member* |  |
|  | 1. *Menampilkan form pendaftaran* |
| 1. *Mengisi field dari form yang sudah disediakan system* |  |
|  | 1. *Menyimpan data member ke database.* |
| ***Skenario Alternatif*** | |
| *pre kondisi : penyewa memasuki halaman utama web* | |
| *Membatalkan pendaftaran menjadi member.* | |

*Nama use case : buat id*

*Use case Skenario*

|  |  |
| --- | --- |
| ***Aksi Actor*** | ***Reaksi Sistem*** |
| ***Skenario Normal*** |  |
| *Pre kondisi: penyewa sudah mengisi form pendaftarn* |  |
|  | 1. *System mebuat id secara otomatis* |
| ***Skenario Alternatif*** | |
| *pre kondisi : penyewa sudah mengisi form pendaftaran* | |
| *Id sudah tedaftar* | |

*Nama use case : dapatkan id*

*Use case skenario*

|  |  |
| --- | --- |
| ***Aksi Actor*** | ***Reaksi Sistem*** |
| ***Skenario Normal*** |  |
| *Pre kondisi: penyewa sudah mengisi data diri dan telah menyimpan data pendaftaran* | |
|  | 1. *Menampilkan id member pelanggan* |

*Nama use case : kelola list mobil*

*Use case skenario*

|  |  |
| --- | --- |
| ***Aksi Actor*** | ***Reaksi Sistem*** |
| ***Skenario Normal*** |  |
| *Pre kondisi: admin telah login* |  |
| 1. *Pilih menu kelola* |  |
|  | 1. *Menampilkan list mobil* |
| 1. *Memilih data mobil yang akan diubah* |  |
| 1. *Memasukan data yang baru* |  |
|  | 1. *Menyimpan data member ke database.* |

*Nama use case :validasi penyewaan*

*Use case skenario*

|  |  |
| --- | --- |
| ***Aksi Actor*** | ***Reaksi Sistem*** |
| ***Skenario Normal*** |  |
| *Pre kondisi: operator telah login* |  |
| 1. *Pilih menu validasi* |  |
|  | 1. *sistem menampilkan form validasi* |
| 1. *memasukan id penyewaan* |  |
| 1. *menekan tombol validasi* |  |
|  | 1. *Menampilkan data penyewaan.* |
| ***Skenario alternatif*** | |
| *Pre kondisi: operator telah login* | |
| *Validasi gagal* | |

## Diagram Kelas

Bagian ini diisi dengan daftar seluruh kelas kelas analisis dalam tabel berikut:

|  |  |  |
| --- | --- | --- |
| *No* | *Nama Kelas* | *Jenis* |
|  |  |  |
|  |  |  |
|  |  |  |

Untuk setiap kelas analisis, lakukan (dengan melengkapi subbab-subbab berikutnya):

* identifikasi tanggung-jawab (responsibility)
* identifikasi atribut

|  |  |  |
| --- | --- | --- |
| *Nama Kelas* | *Daftar Tanggung-Jawab* | *Daftar Atribut* |
| *Kelas xxxx* | *1.*  *2.*  *3.* | *1.*  *2.*  *3.* |
|  |  |  |
|  |  |  |

## Diagram Kelakuan

*Jika diperlukan, buatlah diagram kelakuan sesuai kebutuhan. Bisa berupa diagram kelakuan sistem maupun diagram kelakuan sebuah kelas.*

## Kebutuhan Non Fungsional

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.*

## Batasan Perancangan

Sebutkan batasan perancangan jika ada. Contoh : harus memakai library yang ada, harus memakai sepotong kode yang sudah pernah dikembangkan, harus memperhatikan hal-hal tertentu

## Kerunutan (traceability)

Diisi dengan tabel yang berisi traceability dari hasil analisis. Gunanya untuk menilai apakah hasil analisis “runut” dan lojik. Untuik sementara, baru didefinisikan Data-store versus E-R.

### Kebutuhan Fungsional vs Use Case

Mapping kebutuhan fungsional dengan use case terkait

|  |  |
| --- | --- |
| **ID Kebutuhan Fungsional** | **ID Use Case Terkait** |
|  |  |

### Use Case vs Kelas Terkait

Mapping use case dengan kelas-kelas terkait

|  |  |
| --- | --- |
| **ID Use Case** | **Kelas Terkait** |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |

## Ringkasan Kebutuhan

Bab ini berisi ringkasan semua kebutuhan. Kebutuhan ini mencerminkan semua hal yang harus dipenuhi, dan nantinya akan menjadi arahan untuk tahapan testing, karena pada dasarnya, semua kebutuhan harus dapat ditest supaya dapat dibuktikan dipenuhi. Dibagi menjadi dua bagian: fungsional dan non fungsional.

### Kebutuhan Fungsional

| **ID** | **Deskripsi** |
| --- | --- |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |

### Kebutuhan Non Fungsional

| **ID** | **Deskripsi** |
| --- | --- |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |