LAMPIRAN A

DOKUMENTASI PERANGKAT LUNAK

DOKUMENTASI PEMBANGUNAN PERANGKAT LUNAK

*<terstruktur>*

<xxxxxx >

untuk:

<Jurusan IF>

Dipersiapkan oleh:

<Nama NPM>

Jurusan Teknik Informatika – Unikom

Jl. Dipatiukur No. 112-116, Bandung 40132

|  |  |  |
| --- | --- | --- |
| Jurusan Teknik Informatika Unikom | Nomor Dokumen | Halaman |
| *<KLS /1 >* | *1/50* |
| Revisi | *1.0* | *Tgl: <isi tanggal>* |

DAFTAR PERUBAHAN

|  |  |
| --- | --- |
| Revisi | Deskripsi |
| A | Perubahan DFD |
| B | Perubahan ER |
| C |  |
| D |  |
| E |  |
| F |  |
| G |  |

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| INDEXTGL | - | A | B | C | D | E | F | G |
| Ditulis oleh |  | Mhs | Mhs | Mhs |  |  |  |  |
| Diperiksa oleh |  |  |  |  |  |  |  |  |
| Disetujui oleh |  |  |  |  |  |  |  |  |

Daftar Halaman Perubahan

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

Daftar Isi

[1. Pendahuluan 7](#_Toc129421000)

[1.1 Tujuan Penulisan Dokumen 7](#_Toc129421001)

[1.2 Lingkup Masalah 7](#_Toc129421002)

[1.3 Definisi, Istilah dan Singkatan 7](#_Toc129421003)

[1.4 Aturan Penomoran 7](#_Toc129421004)

[1.5 Referensi 7](#_Toc129421005)

[1.6 Deskripsi umum Dokumen (Ikhtisar) 7](#_Toc129421006)

[2 Deskripsi Umum Perangkat Lunak 7](#_Toc129421007)

[2.1 Deskripsi Umum Sistem 7](#_Toc129421008)

[2.2 Fungsi Produk 7](#_Toc129421009)

[2.3 Karakteristik Pengguna 7](#_Toc129421010)

[2.4 Batasan 7](#_Toc129421011)

[2.5 Lingkungan Operasi 8](#_Toc129421012)

[3 Deskripsi Umum Kebutuhan 8](#_Toc129421013)

[3.1 Kebutuhan antarmuka eksternal 8](#_Toc129421014)

[3.1.1 Antarmuka pemakai 8](#_Toc129421015)

[3.1.2 Antarmuka perangkat keras 8](#_Toc129421016)

[3.1.3 Antarmuka perangkat lunak 8](#_Toc129421017)

[3.1.4 Antarmuka komunikasi 8](#_Toc129421018)

[3.2 Deskripsi Fungsional 8](#_Toc129421019)

[3.2.1 Context Diagram 8](#_Toc129421020)

[3.2.1.1 DFD Level 1 8](#_Toc129421021)

[3.3 Data Requirement 9](#_Toc129421022)

[3.3.1 E-R diagram 9](#_Toc129421023)

[3.4 Non Functional Requirement 9](#_Toc129421024)

[3.5 Batasan Perancangan 10](#_Toc129421025)

[3.6 Kerunutan (traceability) 10](#_Toc129421026)

[3.6.1 Data Store vs E-R 10](#_Toc129421027)

[3.7 Ringkasan Kebutuhan 10](#_Toc129421028)

[3.7.1 Functional Requirement Summary 10](#_Toc129421029)

[3.7.2 Non Functional Requirement Summary 10](#_Toc129421030)

[4 Deskripsi Perancangan Global 11](#_Toc129421031)

[4.1 Rancangan Lingkungan Implementasi 11](#_Toc129421032)

[4.2 Deskripsi Data 11](#_Toc129421033)

[4.2.1 Definisi Domain/Type 11](#_Toc129421034)

[4.2.2 Conceptual Data Model 11](#_Toc129421035)

[4.2.3 Physical Data Model 11](#_Toc129421036)

[4.2.4 Daftar Tabel Aplikasi 11](#_Toc129421037)

[4.3 Dekomposisi Fungsional Modul 12](#_Toc129421038)

[5 Deskripsi Perancangan Rinci 12](#_Toc129421039)

[5.1 Deskripsi Rinci Tabel 12](#_Toc129421040)

[5.1.1 Tabel <Nama..> 12](#_Toc129421041)

[5.1.2 Tabel <Nama> 13](#_Toc129421042)

[5.2 Deskripsi Fungsional secara Rinci 13](#_Toc129421043)

[5.2.1 Spesifikasi Fungsi/Proses <nama fungsi 1> 13](#_Toc129421044)

[5.2.2 Spesifikasi Fungsi/Proses <nama fungsi 2> 13](#_Toc129421045)

[5.3 Dekomposisi Fisik Modul 13](#_Toc129421046)

[5.4 Matriks Kerunutan 14](#_Toc129421047)

[6 Pengujian Perangkat Lunak 14](#_Toc129421048)

[6.1 Lingkungan Pengujian 14](#_Toc129421049)

[6.1.1 Perangkat Lunak Pengujian 14](#_Toc129421050)

[6.1.2 Perangkat Keras Pengujian 14](#_Toc129421051)

[6.2 Material Pengujian 14](#_Toc129421052)

[6.3 Sumber Daya Manusia 15](#_Toc129421053)

[6.4 Prosedur Umum Pengujian 15](#_Toc129421054)

[6.4.1 Pengenalan dan Latihan 15](#_Toc129421055)

[6.4.2 Persiapan Awal 15](#_Toc129421056)

[6.4.2.1 Persiapan Prosedural 15](#_Toc129421057)

[6.4.2.2 Persiapan Perangkat Keras 15](#_Toc129421058)

[6.4.2.3 Persiapan Perangkat Lunak 16](#_Toc129421059)

[6.4.3 Pelaksanaan 16](#_Toc129421060)

[6.4.4 Pelaporan Hasil 16](#_Toc129421061)

[6.5 Identifikasi dan Rencana Pengujian 16](#_Toc129421062)

[6.6 Deskripsi dan Hasil Uji 17](#_Toc129421063)

[6.6.1 <Identifikasi kelas pengujian XXXXX> 17](#_Toc129421064)

[6.6.1.1 <Identifikasi butir pengujian YYYYY> 17](#_Toc129421065)

[6.6.1.2 <Identifikasi butir pengujian YYYYY> 17](#_Toc129421066)

[6.7 Keterunutan Pengujian 19](#_Toc129421067)

[7 Spesifikasi Produk Perangkat Lunak 20](#_Toc129421068)

[7.1 Perangkat Lunak Siap Eksekusi 20](#_Toc129421069)

[7.2 Berkas Sumber 20](#_Toc129421070)

[7.3 Syarat Pemaketan 20](#_Toc129421071)

[7.4 Prosedur Konstruksi 20](#_Toc129421072)

[8 Panduan Instalasi 20](#_Toc129421073)

[8.1 Instalasi Program Siap Eksekusi 20](#_Toc129421074)

[8.2 Instalasi Kode Program Sumber 20](#_Toc129421075)

[9 Penutup 21](#_Toc129421076)

Setelah Daftar Isi Boleh ada Daftar Tabel dan Daftar Gambar

# 1. Pendahuluan

## Tujuan Penulisan Dokumen

Tuliskan dengan ringkas tujuan dokumen SKPL ini dibuat, dan digunakan oleh siapa.

## Lingkup Masalah

Tuliskan dengan ringkas nama aplikasi dan deskripsinya. Maksimal 1 paragraf

## Definisi, Istilah dan Singkatan

Semua definisi dan singkatan yang digunakan dalam dokumen ini dan penjelasannya

## Aturan Penomoran

Tuliskan jika anda memakai aturan penomoran

## Referensi

Dokumentasi PL yang dirujuk oleh dokumen ini.

Buku, Panduan, Dokumentasi lain yang dipakai dalam pengembangan PL ini.

## Deskripsi umum Dokumen (Ikhtisar)

Tuliskan sistematika pembahasan dokumen SKPL ini.

# Deskripsi Umum Perangkat Lunak

## Deskripsi Umum Sistem

Tuliskan System Overview, dalam bentuk gambar dan narasi yang dapat memberikan gambaran tentang aplikasi dan konteksnya (gambar yang mirip dengan Context diagram, tetapi dengan logo yang lebih gampang dimengerti awam).

## Fungsi Produk

Memuat fungsi-fungsi sistem yang utama dan diberikan langsung ke pengguna, kira-kira sama dengan Bubble level 1, tapi dengan kata-kata. Boleh juga disertai dengan diagram semacam yang telah dibuat dengan judul diagram keterkaitan antar modul

## Karakteristik Pengguna

Minimal sebuah tabel dengan Kolom : Pengguna, Pekerjaan, Hak Akses. Kolom Hak Akses dihubungkan dengan Fungsi utama yang muncul pada Fungsi Produk

| **Kategori Pengguna** | **Tugas** | **Hak Akses ke aplikasi** |
| --- | --- | --- |
| Petugas |  | Insert |
| Manager | Menerima laporan | Tidak ada |
|  |  |  |

## 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 Umum Kebutuhan

## Kebutuhan antarmuka eksternal

Hanya diisi jika Aplikasi memerlukan fasilitas khusus .

### Antarmuka pemakai

User interface untuk mengoperasikan Perangkat Lunak : keyboard, mouse, printer

### Antarmuka perangkat keras

TIdak ada

### Antarmuka perangkat lunak

Hanya diisi jika PL memakai interface (berupa PL), misalnya API Windows, Apache, Chrystal Report, ODBC

### Antarmuka komunikasi

Hanya diisi jika PL beroperasi di jaringan dan membutuhkan alat komunikasi khusus, misalnya RS23, TCP IP.

## Deskripsi Fungsional

Awali dengan Context diagram dan penjelasan berupa narasi

### Context Diagram

Buat dan ceritakan Context diagram

#### DFD Level 1

Chapter nya dapat dibuat dengan luwes Awali dengan Context diagram dan sedikit penjelasan berupa narasi jika perlu. Perhatikan kaidah perancangan :

* Pilih notasi sehingga proses yang didekomposisi atau tidak didekomposisi dapat dibaca dengan mudah
* Nama Bubble harus terdiri dari katakerja dan kata benda
* Nama yang dipakai untuk Bubble, data store, dataflow harus konsisten (identitas perlu)
* Setiap level harus konsisten aliran datanya dengan level sebelumnya
* Usahakan agar external entity pada setiap level konsisten peletakannya
* Banyaknya bubble yang disarankan pada setiap level tidak melebihi 7 bubble
* Dekomposisi berdasarkan kelompok data lebih disarankan (memudahkan aliran data ke storage yang sama)
* Nama Proses yang umum hanya untuk bubble yang masih akan didekomposisi
* Nama Proses spesifik (Add, Update, Delete,Calculate, Compare, Merge, ..) pada CASE tools harus disertai dengan Pspec yang jelas walaupun Pspec tidak diprint di dokumen ini
* Pada Proses yang sudah tidak didekomposisi, nama Proses dan nama Data harus sudah spesifik
* Aliran ke storage harus melalui proses, tidak boleh langsung dari ecternal entity
* Aliran data untuk Proses “Report ..” : harus ada aliran keluar. Akan ada aliran masuk jika perlu parameter untuk mengaktifkan report
* Aliran data yang tidak ada datastorenya harus diteliti, apakah memang tidak mencerminkan persisten entity (perlu disimpan dalam file/tabel) , yaitu kelak hanya akan menjadi “variabel” dalam program.

Dst sampai level terendah

#### DFD Level 2

DFD Level 2 Proses 2

Table Spesifikasi Proses

DFD Level 2 Proses 3

Table Spesifikasi Proses

## Data Requirement

Uraikan dengan ringkas, data apa saja yang harus dikelola oleh aplikasi, disarikan dari semua kata benda yang ada pada business process

### E-R diagram

Gambar E-R diagram yang benar-benar konseptual, dengan VISIO. Minimal ada nama Entity, Relasi dan Key (Skema relasi). Sudah dijelaskan apa bedanya E-R konseptual dengan Conceptual Data Model pada Case Tools, karena E-R diagram ini tidak mungkin digambar dengan Case Tools. Keterbatasan CASE Tools biasanya adalah:

* tidak mungkin mempunyai relasi dengan atribut non-key
* tidak mungkin mempunyai relasi bukan biner (terner, dan lebih tinggi)

sehingga akibatnya, relasi dijadikan “entity”. Kenapa E-R konseptual disarankan untuk digambar, adalah karena E-R ini sebenarnya lebih mencerminkan abstraksi perancang

## Non Functional Requirement

Uraikan dengan ringkas kebutuhan non fungsional dalam tabel sebagai berikut. Isilah Kolom Requirement dengan kalimat yang jelas dan kelak dapat ditest untuk dipenuhi. SRS-NO adalah nomor requirement yang harus ditelusuri pada saat test. Tuliskan N/A bila Not Applicable..

| **SRS-Id** | **Parameter** | **Requirement** |
| --- | --- | --- |
|  | Availability | 5 hari/ mgu, 8 jam/hari |
|  | Reliability | 5% error |
|  | Ergonomy |  |
|  | Portability | Windows Xp |
|  | Memory |  |
|  | Response time | 5dtk |
|  | 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 design 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.

### Data Store vs E-R

Mapping data store pada DFD dengan Entity - Relasi

|  |  |  |
| --- | --- | --- |
| **Data Store** | **Entity** | **Relasi** |
| DetailTransaksiKirim |  | √ |
| TransaksiKirim | √ |  |
| Barang | √ |  |
|  |  |  |
|  |  |  |

## Ringkasan Kebutuhan

Bab ini berisi ringkasan semua “Requirement item”. Requirement item ini mencerminkan semua hal yang harus dipenuhi, dan nantinya akan menjadi arahan untuk tahapan testing, karena pada dasarnya, semua requirement harus dapat ditest supaya dapat dibuktikan dipenuhi. Dibagi menjadi dua bagian: functional dan non functional

### Functional Requirement Summary

| **SRS-Id** | **Description** |
| --- | --- |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |

### Non Functional Requirement Summary

| **SRS-Id** | **Description** |
| --- | --- |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |

# Deskripsi Perancangan Global

## Rancangan Lingkungan Implementasi

Sebutkan Operating system, DBMS, development tools, filing system, bahasa pemrograman yang dipakai

## Deskripsi Data

Berisi deskripsi tabel-tabel basis data jika aplikasi berbasis data. Awali dengan daftar tabel basisdata dan deskripsi isinya. Untuk setiap tabel, harus mengandung Nama tabel, jenisnya, Volume, laju, primary key, constraint integrity dengan tabel lain( jika ada). Volume dan laju harus mimimal mengandung angka kira-kira.

Boleh berasal dari “dumb” dari database yang digunakan.

### Definisi Domain/Type

Sebutkan nama domain (type terdefinisi) yang anda rancang pada aplikasi ini dengan mengisi tabel sebagai berikut

| **Domain name** | **Power Designer Type (mis)** |
| --- | --- |
| Rupiah | NUM … |
|  |  |

### Conceptual Data Model

Gambar ini diambil dari Case Tools. . Hasilnya diprint di bagian ini

### Physical Data Model

Jika ada, Gambar ini adalah hasil generate diambil dari Case Tools. Gunanya supaya nantinya langsung diterjemahkan menjadi tabel atau bahkan mungkin dipakai untuk membangkitkan tabel secara otomatis.

### Daftar Tabel Aplikasi

.Awali dengan daftar tabel basisdata, primary key dan deskripsi isinya.

| **Nama Tabel** | **Primary key**  | **Data Store** | **E/R** | **Deskripsi isi** |
| --- | --- | --- | --- | --- |
| Menu | IdMenu | Menu | E |  |
| Barang | IdBArang | Produk | E |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

Untuk setiap tabel, buatlah deskripsi rincinya pada Sub-bab Deskripsi Rinci Tabel

## Dekomposisi Fungsional Modul

Berisi dekomposisi “lojik” dari modul. Minimal berisi tabulasi dengan kolom: Modul, Proses, Keterangan. Kolom keterangan hanya diisi jika proses tidak tergambarkan dalam DFD. Misalnya untuk proses-proses yang mewakili suatu library umum

**Tabel. XXX. Input-Proses-Output**

No. Fungsi diturunkan dari semua hal yang harus diprogram :

* Bubble DFD yang tidak didekomposisi lagi.
* Proses-proses yang “membungkus” proses lain (misalnya menjadi Menu)

Setelah IPO ini, setiap fungsi akan dibuat detailnya. Pada bagian berikutnya

Konvensi : setiap bubble harus menjadi fungsi saja. Jika “menghilang” dan tidak akan diprogram, maka akan ditulis di bagian keterangan (misalnya Menu yang akan diprogram dengan Editor VB dan tidak lagi ditulis kodenya, maka pada keterangan akan ditulis :”Ditangani oleh VB”

Disarankan agar penomoran fungsi dibuat hirarkhis, sehingga jika dijadikan menu akan sekaligus merupakan Function Tree

| **No.Fungsi** | **Fungsi/Proses****(proses daun)** | **Tabel/Data Input** | **Tabel /Data Output** | **Keterangan** |
| --- | --- | --- | --- | --- |
|  | Validasi Login | User | User |  |
|  | InsertdtUser | User | User |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

Untuk setiap nomor fungsi, buatlah spesifikasi rincinya pada Deskripsi rancangan Rinci

# Deskripsi Perancangan Rinci

## Deskripsi Rinci Tabel

Setiap tabel pada rancangan global, dirinci satu per satu

### Tabel TransaksiKirim

Identifikasi/Nama : Transaksi Kirim

Deskripsi Isi : ……..

Primary Key : IdTrans

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Id Field** | **Deskripsi** | **Tipe & length** | **Boleh****NULL** | **Default** | **Keterangan** |
| Id\_master |  | CHAR(8) | NO |  |  |
| Id\_ref |  |  |  |  | Refer ke t\_ref |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |

Catatan : kolom “Boleh NULL” berisi “NO” artinya tidak boleh kosong, berisi “YES” artinya boleh NULL

### Tabel <Nama>

Buat seperti di atas

## Deskripsi Fungsional secara Rinci

Setiap Fungsi pada rancangan global, dirinci satu per satu

### Spesifikasi Fungsi/Proses Validasi Login

Identifikasi/Nama : Form Login

Deskripsi Isi : ……..

Jenis : **Form Entry columnar/Tabular/Master-Detail**

 Report Columnar/tabular/Master-Detail

 Form berisi dialog/button saja

 Proses tanpa layar

Tabel input : User

Tabel output : User

Query : (diisi jika query cukup rumit)

Layar Utama :

*Gambarkan layar dan percabangan ke layar lain function key/pilihan yang dilakukan)*

*Jika layar mengandung filed dan label, gambarkanlah pada posisi nya, supaya siap dikoding. Jika ada zoning/frame, gambarkan pula an jelaskan pada spesifikasi Objek pada layar*

Objek

OK

Algoritma : (diisi jika algoritma cukup kompleks)

Layout Report : (diisi jika fungsi menghasilkan report)

### Spesifikasi Fungsi/Proses <nama fungsi 2>

Untuk setiap fungsi, buat detailnya seperti di atas

## Dekomposisi Fisik Modul

Berisi dekomposisi “fisik” dari modul. Minimal berisi tabulasi dengan kolom: Sub Aplikasi, Modul, Nama File, Input, Output. Sub Aplikasi biasanya dibuat per pengguna. Dibuat per modul

Berisi struktur direktori dan pengumpulan fungsi menjadi file. Minimal berisi tabulasi dengan kolom: Modul, Proses, Keterangan. Kolom keterangan hanya diisi jika proses tidak tergambarkan dalam DFD. Misalnya untuk proses-proses yang mewakili suatu library umum

| **Nama Direktori** | **Nama File** | **Nama Modul** | **Nama Fungsi** | **Keterangan** |
| --- | --- | --- | --- | --- |
| C: | Kelolatransaksi.vb | Kelola Transaksi |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

## Matriks Kerunutan

Pada bagian ini, diisi dengan tabel, yang memungkinkan pembaca untuk menelusuri keterkaitan perancangan terhadap spesifikasi kebutuhan.

Diisi dengan tabel yang berisi traceability dari SRS dengan nomor CSU

| **SRS-Id** | **No. Fungsi** | **Keterangan** |
| --- | --- | --- |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

# Pengujian Perangkat Lunak

## Lingkungan Pengujian

Bagian ini akan dibagi menjadi beberapa sub bab, untuk menjelaskan lingkungan yang dibutuhkan dalam pengujian perangkat lunak. Bagian ini juga menjelaskan rencana implementasi dan pengendalian sumber daya (perangkat lunak, perangkat keras dan dari sisi persiapan organisasi) yang akan melakukan pengujian kualifikasi formal.

### Perangkat Lunak Pengujian

Bagian ini berisi identifikasi dari nama, nomor dan versi (jika ada atau jika sudah ada), dari item perangkat lunak (misalnya sistem operasi, kompilator, perangkat komunikasi, paket aplikasi yang terkait, basisdata, file masukan, *code auditor*, *Tools* pengujian) yang diperlukan untuk melakukan pengujian. Sebutkan pula hak pemakaian atau lisensi dari tiap perangkat lunak pengujian yang digunakan.

Bagian ini juga akan menjelaskan guna dari setiap item, penjelasan media yang digunakan, dukungan peralatan (jika ada) dan masalah keamanan yang berhubungan dengan item perangkat lunak.

### Perangkat Keras Pengujian

Bagian ini berisi identifikasi dari nama, nomor dan versi (jika ada) dari perangkat keras yang dilibatkan dalam pengujian, peralatan khusus (misalnya *interface card* khusus), peralatan komunikasi (jaringan dan peralatannya), dan peralatan lain yang mungkin terlibat.

## Material Pengujian

Beberapa material tambahan yang mungkin dibutuhkan dapat diperjelas dibagian ini. Material ini misalnya manual perangkat lunak, listing program, media yang berisi perangkat lunak yang akan diuji, contoh tampilan keluaran, formulir terkait, atau instruksi-instruksi khusus. Material yang dituliskan di sini adalah material yang belum dituliskan di dokumen-dokumen lainnya.

## Sumber Daya Manusia

Bagian ini menjelaskan jumlah, tingkat keahlian, dan kriteria/prasyarat dari sumber daya manusia yang terlibat dalam pengujian, termasuk saat dibutuhkan (tipe pengujian).

## Prosedur Umum Pengujian

### Pengenalan dan Latihan

Bagian ini menjelaskan pengenalan dan latihan yang akan diberikan sebelum dan selama pengujian, bila ada. Informasi yang berhubungan dengan orang yang terlibat sudah dijelaskan di 2.4. Pelatihan ini termasuk instruksi penggunaan perangkat lunak bagi pengguna akhir atau operator, instruksi perawatan perangkat lunak dan instruksi pengendalian perangkat lunak berkelompok. Berikan pula jadwal atau waktu kapan dan seberapa lama pengenalan atau latihan ini dilakukan.

### Persiapan Awal

Bagian ini akan dibagi menjadi beberapa sub bab, untuk menjelaskan lingkungan yang dibutuhkan dalam pengujian perangkat lunak. Bagian ini juga menjelaskan rencana implementasi dan pengendalian sumber daya (perangkat lunak, perangkat keras dan dari sisi persiapan organisasi) yang akan melakukan pengujian. Bagian ini dapat dijelaskan secara terpisah untuk tiap kelas atau butir uji bila ada persiapan awal khusus yang perlu dilakukan untuk satu kelas atau satu butir uji. Bagian khusus ini dijelaskan pada deskripsi uji di bawah.

#### Persiapan Prosedural

Bagian ini menyatakan persiapan prosedural (manual) yang perlu dilakukan untuk melakukan pengujian. Contohnya: bila pengujian dilakukan di suatu lingkungan khusus, misalnya di ruang komputer, maka untuk melakukan pengujian ini perlu ada ijin masuk khusus, ijin penginstallan perangkat lunak yang akan diujikan, pencatatan log-book dan lain-lain.

#### Persiapan Perangkat Keras

Bagian ini akan menjelaskan prosedur yang perlu untuk menyiapkan perangkat keras untuk pengujian. Acuan dapat dibuat untuk menerbitkan petunjuk operasi dari setiap prosedur ini. Pada bagian ini misalnya akan menyatakan hal-hal berikut:

1. Perangkat keras yang akan digunakan, nama dan nomor jika ada
2. Setting dari switch (misalnya untuk printer)
3. Instruksi langkah-langkah untuk penyiapan perangkat keras hingga siap pakai.

Paragraf ini berisi identifikasi dari nama, nomor dan versi (jika ada) dari perangkat keras yang dilibatkan dalam pengujian, peralatan antarmuka (*interface*), peralatan komunikasi, peralatan pengujian waktu (jika diperlukan), dan peralatan lain yang mungkin terlibat.

**Contoh:**

|  |
| --- |
| *Perangkat keras yang perlu disiapkan antara lain:** 3 perangkat komputer yang masing-masing dilengkapi dengan:
* *1 harddisk dengan kapasitas minimum 5****00*** *MB*
* *1 color monitor VGA pada perangkat yang sama tempat harddisk berada*
* *32 MB RAM*
* *1 keyboard*
* *1 Floppy drive*
* 1 printer Laser Jet yang terhubung ke salah satu perangkat komputer
* 1 Network Hub
* 3 NIC , yang terpasang pada masing-masing komputer, dan kabel UTP yang terhubung ke masing-masing komputer dengan konfigurasi star dan terpusat di Network Hub
 |

Bila diperlukan suatu konfigurasi yang khusus, dapat dibuat dalam suatu gambar.

#### Persiapan Perangkat Lunak

Bagian ini akan menjleaskan prosedur atau tata cara yang diperlukan untuk menyiapkan item yang akan diuji, perangkat lunak yang terkait termasuk data untuk pengujian. Informasi yang mungkin perlu ada antara lain:

1. Perangkat lunak yang diuji (bisa dalam bentuk media penyimpanannya misalnya disket, cdrom, atau media lain)
2. Perangkat lunak yang digunakan untuk menguji (misalnya simulator, test driver, database)
3. Instruksi untuk mengaktifkan program, termasuk urutan langkah rincinya bila perlu
4. Instruksi untuk inisialisasi umum untuk suatu kasus uji.

### Pelaksanaan

Bagian ini menjelaskan strategi pelaksanaan pengujian itu sendiri. Contoh strategi ini adalah pembagian pengujian menjadi dua tahap: pengujian unit dan pengujian sistem. Contoh lain adalah: pengujian dilakukan pada lingkungan khusus yang dibangun untuk pengujian dan tidak dilakukan pada lingkungan operasional sesungguhnya.

### Pelaporan Hasil

Bagian ini menjelaskan pada siapa saja dokumen hasil pengujian akan diserahkan baik untuk diverifikasi maupun penyerahan akhir.

## Identifikasi dan Rencana Pengujian

Subbagian ini akan dibagi menjadi beberapa sub bagian untuk mengenali kondisi umum pengujian yang dilakukan serta kelas pengujian yang akan dilakukan.

Bagian ini menjelaskan lingkup keseluruhan dari perencanaan pengujian. Dari sejumlah requirement yang akan diuji yang dituliskan di SKPL, buatlah pengelompokkannya dan jadikan tabel pada bagian ini.

Contoh:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| *Kelas Uji* | *Butir Uji* | *Identifikasi* | *Tingkat Pengujian* | *Jenis Pengujian* | *Jadwal* |
| *SKPL* | *PDHUPL* |
| *Pengujian Antarmuka Pengguna* | *Pengujian Pewarnaan* | *SKPL\_Xsoft\_01* | *AU\_01* | *Pengujian Sistem* | *White Box* | *12/01/2000 – 15/01/2000* |
| *Penataletakan Window* | *SKPL\_Xsoft\_02* | *AU\_02* | *Pengujian Unit* | *Black Box* | *15/01/2000 – 17/01/2000* |
| *Pembangkitan kode* | *Pembangkitan Kode Pelanggan* | *SKPL\_Xsoft\_03* | *BK\_01* | *Pengujian Unit* | *Black Box* | *18/01/2000 – 19/01/2000* |
| *Kebenaran Data Pelanggan* |  | *BK\_02* | *Pengujian Unit* | *White Box* | *19/01/2000 – 20/01/2000* |

Pada contoh di atas daftar requirement yang akan diuji dituliskan di kolom Butir Uji.

## Deskripsi dan Hasil Uji

Bagian ini diuraikan untuk setiap butir uji yang ada. Setiap butir uji yang ada di sana dijelaskan rinciannya dalam tabel seperti di bawah ini. Hasil yang didapat baru diisi setelah pengujian dilakukan. Bila perlu ada penjelasan khusus untuk mendeskripsikan tiap kelas pengujian, maka tuliskan sebelum merincikan butir ujinya. Bila perlu ada penjelasan khusus untuk mendeskripsikan tiap butir uji, maka tuliskan sebelum tabel.

### <Identifikasi kelas pengujian XXXXX>

#### <Identifikasi butir pengujian YYYYY>

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Identifikasi  | Deskripsi | Prosedur Pengujian | Masukan | Keluaran yang Diharapkan | Kriteria Evaluasi Hasil | Hasil yang Didapat | Kesimpulan |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |

#### <Identifikasi butir pengujian YYYYY>

contoh:

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Identifikasi  | Deskripsi | Prosedur Pengujian | Masukan | Keluaran yang Diharapkan | Kriteria Evaluasi Hasil | Hasil yang Didapat | Kesimpulan |
| *BK\_02\_01* | Pengujian hasil pemasukan data pelanggan oleh operator | * Buka File data pelanggan
* Cari rekord dengan data modus pemasukan yang diinginkan
* Lihat tanggal lahir pelanggan
* Lihat kode pelanggan
* Bandingkan dengan rumus pembangkitan kode pelanggan
 | Kode modus pemasukan operator (01) | 01<tgl\_lahir>00101<tgl\_lahir>00201<tgl\_lahir>003dst | 01<tgl\_ lahir> <nomor terurut> | 01<tgl\_ lahir><no\_ loncat | ditolak |
| BK\_02\_02 | Pengujian hasil pemasukan data pelanggan oleh pelanggan secara on-line | Kode modus pemasukan on-line(02) | 02<tgl\_lahir>00102<tgl\_lahir>00202<tgl\_lahir>003dst | 02<tgl\_ lahir> <nomor terurut> | 02<tgl\_ lahir><no\_ terurut> | diterima |

Penjelasan setiap kolom yang ada di atas diuraikan dalam penjelasan di bawah ini:

* Identifikasi

Bagian ini mengidentifikasi suatu kasus uji dengan suatu kode yang unik. Bagian ini juga akan berisi guna dari pengujian dan penjelasan singkatnya. Subparagraf berikut ini akan memberikan deskripsi rinci dari setiap kasus uji.

* Deskripsi

Bagian ini menguraikan deskripsi singkat tentang kasus uji yang dipilih.

* Masukan

Bagian ini akan berisi penjelasan tentang masukan pengujian yang diperlukan untuk suatu kasus uji. Hal-hal berikut dapat dimasukkan, jika perlu:

1. Nama, guna dan deskripsi dari setiap masukan (termasuk akurasi dan jangkauan nilainya)
2. Sumber masukan pengujian dan metode yang digunakan untuk memiliki masukan pengujian
3. Apakah masukan untuk pengujian ini adalah masukan yang nyata atau hanya simulasi
4. Waktu atau urutan pemasukan
5. Perilaku dari data masukan yang akan dikendalikan, misalnya
6. Pengujian suatu item dengan jumlah tipe data dan nilai yang minimum atau secukupnya
7. Mencoba setiap item dengan suatu rangkaian nilai bertipe data yang benar dan pada setiap pengujian memeriksa efek overflow, underflow dan kondisi-kondisi jelek lainnya
8. Mencoba setiap item dengan tipe data yang tidak valid dan nilainya untuk setiap masukan data yang tidak pasti
9. Pengujian ulang jika mungkin
* Keluaran yang Diharapkan

Bagian ini menberisi penjelasan setiap hasil harapan uji untuk setiap kasus uji. Baik nilai hasil sementara atau hasil akhir dapat dinyatakan.

* Kriteria Evaluasi Hasil

Paragraf yang berikut ini akan mengidentifikasi kriteria yang digunakan untuk mengevaluasi nilai sementara dan nilai akhir untuk setiap kasus uji. Untuk setiap hasil uji, informasi yang dihasilkan antara lain *misalnya*:

1. Keakuratan atau jangkauan nilai keluaran yang masih dapat diterima
2. Jumlah kombinasi minimum atau alternatif kondisi masukan/keluaran yang akan menunjukkan hasil uji yang dapat diterima
3. Durasi maksimum/minmum pengujian, dalam satuan waktu atau jumlah kejadian
4. Jumlah maksimum interupt, *halt* atau penyebab lain yang membuat sistem berhenti
5. Batas yang masih dapat diijinkan untuk terjadinya kesalahan
6. Kondisi yang menyebabkan hasil uji tidak memuaskan, sehingga perlu dilakukan pengujian ulang
7. Kondisi dimana keluaran akan diinterpretasikan sebagai adanya kesalahan di data uji masukan, pada file data/basisdata, atau pada tata cara uji
8. Indikasi yang masih dapat diijinkan terhadap pengendali, status dan hasil dari pengujian dan kesiapan untuk kasus uji berikutnya (yang mungkin berupa keluaran tambahan)
* Prosedur Pengujian

Pada bagian ini akan dijelaskan prosedur uji untuk setiap kasus uji. Prosedur uji ini akan didefinisikan sebagai sekumpulan langkah yang terurut secara sekuensial. Untuk memudahkan dalam dokumentasinya, prosedur uji ini dapat dimasukkan sebagai lampiran pada paragraf ini.

Tingkat kerincian dari cocok adalah level dimana tingkat tersebut berguna untuk menentukan hasil yang diharapkan dan membandingkannya dengan hasil yang nyata. Hal-hal berikut sebaiknya diberikan dalam prosedur uji (jika mungkin):

1. Operasi pengujian operator dan peralatan yang dibutuhkan pada setiap langkah, termasuk, misalnya, perintah-perintah untuk

a) Inisialisasi kasus uji dan mencoba masukan pengujian

b) Periksa kondisi pengujian

1. Melakukan evaluasi tunggal terhadap hasil uji
2. Pencatatan Data

 e) Penghentian suatu kasus uji.

f) Jika diperlukan, instruksi pengambilan data

g) Modifikasi basisdata/file data

 h) Ulangi kasus uji jika tidak sukses

1. Lakukan alternatif lain yang dibutuhkan oleh kasus uji

j) Hentikan kasus uji

 2. Hasil harapan uji dan kriteria evaluasi untuk setiap langkah

 3. Jika kasus uji digunakan untuk beberapa kebutuhan (*requirement*), berikan identifikasi yang jelas

 4. Aksi yang harus dilakukan bila terjadi program berhenti atau ada kesalahan seperti

1. Pencatatan data kritis sebagai indikator untuk referensi
2. Sistem berhenti pada suatu waktu
3. Berhenti (*halt*) pada suatu perangkat lunak pendukung pengujian
4. Sekumpulan sistem dan operator yang mencata suatu hasil uji

 5. Prosedur yang digunakan untuk mereduksi dan menganalisa hasil uji untuk mendapatkan kemugkinan-kemungkinan berikut ini:

 a) Mengetahui apakah keluaran sudah dihasilkan

1. Identifikasi meida dan lokasi data yang dihasilkan oleh suatu kauus uji
2. Evaluasi keluaran sebagai dasar untuk melanjutkan urutan pengujian
3. Evaluasi keluaran pengujian terhadap keluaran

## Keterunutan Pengujian

Bagian ini akan berisi:

1. Keterunutan (*traceability*) dari setiap kasus uji ke kebutuhan sistem. Jika suatu kasus uji terdiri dari banyak kebutuhan, keterunutan harus dibuat dari setiap kumpulan prosedur uji hingga kebutuhan yang diuji
2. Keterunutan dari setiap kebutuhan sistem yang dicakup hingga ke kasus ujinya. Untuk pengujian sistem secara keseluruhan, keterunutan dari setiap kebutuhan sistem secara keseluruhan dari dokumen SKPL. Untuk pengujian sistem, keterunutan dari setiap kebutuhan dalam spesifikasi sistem/subsistem. Jika suatu kasus uji digunakan untuk menangani beberapa kebutuhan, maka isi dari keterunutan ini harus dapat menunjukkan suatu langkah pengujian yang khsusu untuk menangani setiap kebutuhan.

# Spesifikasi Produk Perangkat Lunak

## Perangkat Lunak Siap Eksekusi

Bagian ini menyebutkan rujukan pada berbagai lampiran atau media elektronik dan juga memuat daftar perangkat lunak yang siap dieksekusi, berkas *batch* (*batch files*), berkas perintah (*command files*), berkas data atau berkas-berkas lain yang diperlukan untuk instalasi perangkat lunak pada komputer target.

Bagian ini juga menguraikan atau memberi rujukan pada dokumen panduan/prosedur instalasi program eksekusi (penggunaan gambar akan sangat membantu).

## Berkas Sumber

Bagian ini menyebutkan rujukan pada berbagai lampiran atau media elektronik dan juga memuat daftar berkas sumber, berkas *batch* (*batch files*), berkas perintah (*command files*), berkas data atau berkas-berkas lain yang diperlukan untuk membangun ulang perangkat lunak yang siap dieksekusi.

## Syarat Pemaketan

Bagian ini menyatakan syarat yang mungkin diperlukan untuk memaketkan dan menandai keabsahan duplikat perangkat lunak.

## Prosedur Konstruksi

Bagian ini menguraikan prosedur atau rujukan pada dokumen yang menguraikan prosedur untuk membuat perangkat lunak yang siap dieksekusi dari berkas sumber yang ada serta bagaimana cara memuat perangkat lunak tersebut ke perangkat *firmware* atau media distribusinya (misalnya Compact Disc). Bagian ini juga menyatakan kompilator apa yang diperlukan, versinya, perangkat keras dan perangkat lunak yang diperlukan untuk mengkonstruksi ulang. Cara/prosedur kompilasi, *linking*, dan *building* perangkat lunak tersebut juga dinyatakan termasuk bila perlu, variasinya untuk berbagai situs, konfigurasi, versi.

# Panduan Instalasi

## Instalasi Program Siap Eksekusi

Bagian ini menjelaskan:

* Kebutuhan konfigurasi sistem yang minimum dan yang disarankan
* *Environment variable* yang harus disiapkan
* Data (basisdata) yang harus disiapkan
* Panduan instalasi program eksekusi (penggunaan gambar akan sangat membantu)

## Instalasi Kode Program Sumber

Bagian ini menjelaskan:

* Cara penyalinan program sumber (nama direktori khusus jika ada)
* Lingkungan pengembangan yang harus disiapkan (kompilator yang digunakan, environment variable yang harus ada)
* Data yang harus disiapkan, letak direktori.

# Penutup

Berisi catatan lain yang dirasakan perlu.