**SKPL-01**

SPESIFIKASI KEBUTUHAN PERANGKAT LUNAK

Information System of Pharmacy

untuk:

Apotek

Dipersiapkan oleh:

Kelompok 1

IFIK-40-05

Zeyhan Aliyah 1301160803

Muhammad Yasin Fajari 1301164666

Wanda Yogei Wardani 1301164691

R. Muhammad Rifqi P. 1301164713

Abdan Hafidh A. 1301164727

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-01* | | 20 |
| 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](#_Toc527386552)

[Daftar Halaman Perubahan 2](#_Toc527386553)

[Daftar Isi 3](#_Toc527386554)

[1. Pendahuluan 4](#_Toc527386555)

[1.1 Tujuan Penulisan Dokumen 4](#_Toc527386556)

[1.2 Konvensi Dokumen 4](#_Toc527386557)

[1.3 Cakupan Produk 4](#_Toc527386558)

[1.4 Referensi 5](#_Toc527386559)

[2. Overall Description 5](#_Toc527386560)

[2.1 Perspektif Produk 5](#_Toc527386561)

[2.2 Fungsi Produk 6](#_Toc527386562)

[2.3 Kelas dan Karakteristik Pengguna 7](#_Toc527386563)

[2.4 Lingkungan Operasi 8](#_Toc527386564)

[2.5 Batasan Perancangan dan Implementasi 8](#_Toc527386565)

[2.6 Dokumentasi Pengguna 9](#_Toc527386566)

[2.7 Asumsi dan Dependensi 9](#_Toc527386567)

[3. Requirements Antarmuka Eksternal 10](#_Toc527386572)

[3.1 Antarmuka Pengguna 10](#_Toc527386573)

[3.2 Antarmuka Perangkat Keras 10](#_Toc527386574)

[3.3 Antarmuka Perangkat Lunak 10](#_Toc527386575)

[3.4 Antarmuka Komunikasi 11](#_Toc527386576)

[4. Fitur Sistem (Use Cases) 11](#_Toc527386577)

[4.1 Use Case 1 12](#_Toc527386578)

[4.2 Use Case 2 14](#_Toc527386588)

[4.3 Use Case 3 16](#_Toc527386598)

[4.4 Use Case 4 18](#_Toc527386608)

[4.5 Use Case 5 20](#_Toc527386618)

[4.6 Use Case 6 22](#_Toc527386628)

[4.7 Use Case 7 24](#_Toc527386638)

[5. *Requirements* Nonfungsional Lainnya 26](#_Toc527386648)

[5.1 *Requirements* Performa 26](#_Toc527386649)

[5.2 *Requirements* Keselamatan 26](#_Toc527386650)

[5.3 *Requirements* Keamanan 26](#_Toc527386651)

[5.4 Atribut Kualitas Perangkat Lunak 26](#_Toc527386652)

[5.4.1 Adaptasi 26](#_Toc527386653)

[5.4.2 Ketersediaan 27](#_Toc527386654)

[5.4.3 Ketepatan 27](#_Toc527386655)

[5.4.4 Keandalan 27](#_Toc527386656)

[5.4.5 Keamanan 27](#_Toc527386657)

[5.4.6 Kemerawatan 27](#_Toc527386658)

[5.4.7 Kepemindahan 28](#_Toc527386659)

[6. *Requirements* Lain 28](#_Toc527386660)

[6.1 *Memory* 28](#_Toc527386661)

[6.2 *Design* 28](#_Toc527386663)

[7. Lampiran A: Daftar Kata-Kata Sukar 28](#_Toc527386665)

# Pendahuluan

## Tujuan Penulisan Dokumen

Dokumen ini berisi Spesifikasi Kebutuhan Perangkat Lunak (SKPL) untuk pembuatan Sistem Informasi Apotek. Tujuan penulisan dokumen ini adalah untuk mendefinisikan objek-objek yang akan diimplementasikan dari perangkat lunak yang akan dibangun, baik berupa gambaran umum hingga penjelasan mendetail dan menyeluruh. Dokumen ini akan digunakan sebagai acuan teknis dan bahan evaluasi pada saat proses pengembangan perangkat lunak maupun pada tahap pengembahan selanjutnya.

## Konvensi Dokumen

* Font: Times New Roman
* Size: Ukuran font yang digunakan untuk menulis Spesifikasi Kebutuhan Perangkat Lunak ini bermacam, size 12 untuk setiap sub dari subbab yang ada, size 14 untuk setiap subbab, size 16 untuk judul dari SKPL ini, dan size 18 untuk setiap penulisan bab.
* Font style: **Bold** dan Italic. **Bold** digunakan pada penulisan judul, bab serta subbab yang ada. Sedangkan Italic digunakan pada penulisan Bahasa asing atau selain Bahasa Indonesia.
* Font Alignment: Agar SKPL ini terlihat rapih, kami menggunakan Justify untuk font alignment disetiap paragraf yang ada.
* Table digunakan untuk mendefinisikan tugas dari sebuah user yang terbagi menjadi Pemilik dengan Apoteker.
* Bullets dan Numbering yang digunakan untuk menyertakan sebuah list didalam subbab (jika memang ada atau diperlukan).
* Link yang dibutuhkan sebagai referensi.

## Cakupan Produk

Perangkat lunak ini membantu pengelolaan data dan barang yang diperjualbelikan di Apotek dimana mencakup beberapa aspek, seperti pengelolaan stok obat, pengelolaan data member, dan pengelolaan pemasokan. Perangkat lunak ini akan memudahkan apoteker dalam rekap data sehingga data-data apotek lebih mudah untuk diakses dan dicari karena datanya telah tersimpan secara rapih dalam *database*. Setiap obat yang di *input* akan dicatat baik tanggal produksi dan tanggal kadaluarsanya. Sehingga apabila suatu obat akan mendekati masa kadaluarsa, sistem akan memberikan pesan notifikasi yang akan memudahkan dalam proses pemusnahan obat. Hal ini dilakukan dengan tujuan untuk menghindari pemasaran obat-obatan yang telah jatuh masa aktifnya di apotek. Apabila proses pencatatan masih dilakukan secara manual, proses pemusnahan obat akan sangat memakan waktu karena obat yang sama dapat memiliki tanggal kadaluarsa yang berbeda-beda. Aktivitas apotek pun dapat lebih mudah diawasi. Pemilik apotek dapat melakukan proses pemeriksaan laporan penjualan dengan lebih teliti karena segala sesuatu telah direkam dan tersimpan dalam sistem. Sehingga apabila terdapat beberapa data yang tidak sesuai, dapat terdeteksi dengan mudah.

## Referensi

Dokumen ini merujuk pada hasil wawancara dan observasi pada pengguna tentang kebutuhan yang diperlukannya dan penulisan dokumen ini berdasarkan pada:

* <https://www.academia.edu/11355568/Contoh_SKPL_SPESIFIKASI_KEBUTUHAN_PERANGKAT_LUNAK_>
* <https://www.google.co.id/search?q=atribut+kualitas+perangkat+lunak+skpl&oq=atribut+kualitas+perangkat+lunak+skpl&aqs=chrome..69i57.6321j0j7&client=ms-android-samsung&sourceid=chrome-mobile&ie=UTF-8>

# Overall Description

## Perspektif Produk

Produk ini dijalankan oleh *administrator* (pemilik apotek) dan apoteker yang dihubungkan oleh *server* dan berjalan pada *platform* atau sistem operasi Microsoft Windows XP/Vista/7.Dengan adanya perangkat lunak ini, apotek akan lebih dimudahkan dalam proses mendata, merekap dan mengelola transaksi barang yang dijual. Lalu untuk pemilik dapat menggunakan perangkat lunak ini untuk mengelola data obat, data pemasokan, data member, data pegawai, dan melihat laporan pembelian yang bisa diakses dimana saja dan kapan saja. Dengan berbagai fitur didalamnya seperti daftar obat, lihat pemasokan, kelola pegawai dan laporan transaksi. Proses pembentukan *database* menjadi lebih efektif dan efisien.

## Fungsi Produk

Perangkat lunak ini dikembangkan dengan fungsi utama yaitu fungsi pengelolaan data obat. Fungsi tersebut antara lain:

* **Pendataan Obat**

Melakukan pendataan obat nama obat, kode obat, jenis, resep, hingga tanggal produksi dan tanggal kadaluarsa dari setiap obat yang diperjualbelikan serta dipasok ke dalam apotek. Fitur ini juga berfungsi untuk memberikan pesan notifikasi bahwa obat dengan kode tertentu akan segera kadaluarsa sebelum masa aktifnya habis, sehingga obat-obatan yang telah kadaluarsa tidak lagi diperjualbelikan atau dipasarkan di apotek.

* **Pengecekan Stok Obat**

Subfungsi ini memiliki kemampuan melakukan transaksi stok obat, dimana setiap obat yang terjual akan dilakukan *update* stok secara otomatis oleh sistem, sehingga memudahkan pada pengecekan stok obat apabila terdapat pembelian berikutnya. Juga berfungsi apabila apoteker akan melakukan pemasokan obat, sebagai data obat manakah yang telah habis stoknya sehingga harus dipasok ulang.

* **Rekapitulasi Data Obat**

Memiliki kemampuan untuk melakukan rekapitulasi dari hasil penjualan dan pembelian dalam apotek. Rekapitulasi ini meliputi nomor transaksi, tanggal pembelian obat, kode obat, nama dan jumlah obat yang terjual maupun dibeli dari pemasok. Fungsi dari fitur ini adalah sebagai data yang nantinya akan digunakan untuk menyusun laporan penjualan yang dikirimkan kepada pemilik apotek secara periodik (harian, bulanan dan tahunan), sehingga pemilik apotek dapat mengetahui aktivitas terakhir dari apotek.

* **Data *member***

Menampilkan data *member* apotek yang terdaftar dalam perangkat lunak ini, yang meliputi *id member*, nama, alamat, nomor telepon, dan riyawat pembelian.

* **Kelola Data Pemasok**

Mengelola data pemasok yang menyediakan obat, yang meliputi *id* pemasok, nama, nomor telepon, alamat dan jenis obat. Fitur ini digunakan untuk memudahkan apoteker dalam melakukan pemesanan obat, karena tidak semua obat dapat dipesan dari satu pemasok yang sama. Dalam fitur ini juga tercatat riwayat pemasokan obat-obatan yang dapat diakses oleh pemilik apotek.

## Kelas dan Karakteristik Pengguna

|  |  |  |  |
| --- | --- | --- | --- |
| Kategori Pengguna | Tugas | Hak akses ke aplikasi | Syarat kemampuan yang dimiliki |
| Pemilik | - Mengelola data obat, data pemasokan obat, dan data *member*  - Menetapkan dan mengubah harga obat  - Menghapus data obat, data pemasokan  - Melihat dan memeriksa laporan transaksi  - Melakukan pengecekan stok obat | - Proses lihat data obat, lihat data pemasokan, dan lihat data *member*  - Proses ubah dan update harga obat  - Proses hapus data obat, data pemasokan  - Proses lihat stok obat  - Proses lihat dan periksa laporan transaksi | - Mampu mengoperasikan komputer dan menggunakan aplikasi berbasis GUI serta aplikasi dasar seperti Microsoft excel.  -Mampu dan mengerti cara mengoperasikan aplikasi berbasis web, dan memahami sistem informasi. |

|  |  |  |  |
| --- | --- | --- | --- |
| Apoteker | - Mengelola data obat, data pemasokan obat, data *member*  - Meng-*input* data transaksi  - Mengecek data resep | - Proses lihat data obat, lihat data pemasokan, data *member*  - Proses tambah data resep  - Proses pencarian obat | - Mampu mengoperasikan komputer dan menggunakan aplikasi berbasis GUI, serta aplikasi dasar seperti Microsoft Excel, dan memahami administrasi suatu sistem informasi. |
| Gudang | - Merekap data obat  - Menerima Laporan  - Mendata stok Obat | - Update, delete, edit dan insert data obat  - cek kadaluarsa dan stok obat  - Persediaan obat, transaksi obat, tambah data obat dan hapus data obat | - Mampu mengoperasikan computer serta aplikasi ini, lalu bisa merekap di Microsoft Excel. Melakukan pengecekan laporan dari file PDF/Word, serta memahami pengoperasian terhadap MySQL |

## Lingkungan Operasi

Perangkat lunak ini akan beroperasi pada lingkungan yang dikhususkan yaitu lingkungan Apotek itu sendiri, yaitu Apotek Putra Farma. Tujuan pengoperasian perangkat lunak ini adalah sebagai pendukung dalam mengolah segala data yang ada pada Apotek tersebut, baik data obat, data pemasokan, data *member*, data pegawai, dan melihat laporan pembelian yang bisa diakses dimana dan kapan saja.

Aplikasi ini berjalan di atas *platform* perangkat keras berupa beberapa komputer yang saling terhubung satu sama lain. Perangkat lunak dapat dijalankan pada sistem operasi apapun. Adapun sistem operasi yang disarankan adalah Microsoft Windows XP/Vista/7. Sedangkan untuk pengolah *database* yang digunakan adalah MySQL.

## Batasan Perancangan dan Implementasi

Berikut batasan-batasan pada sistem:

* Sistem operasi pada client menggunakan sistem operasi Windows
* Sistem dikembangkan dengan menggunakan bahasa pemrograman Java.
* Database sistem dibangun menggunakan MySQL.
* Sistem beroperasi secara online maupun offline selama 24 jam, 7 hari dalam seminggu.
* User yang berhak mengakses sistem ini hanya administrator (pemilik apotek) dan apoteker, pembeli hanya dapat melakukan pembelian secara offline

Kebutuhan kinerja dari sistem memiliki respon kurang dari 30 detik untuk setiap aksi pergantian page. Sistem memiliki waktu respon kurang dari 2 menit untuk setiap proses load data. Sistem memiliki waktu respon kurang dari 1 menit untuk proses pencarian data.

Interface dari aplikasi ini pada sisi client berupa GUI yang dapet diakses sistem operasi Windows dengan menggunakan web server. Perangkat lunak ini menggunakan Bahasa Indonesia agar mudah dipahami oleh client.

## Dokumentasi Pengguna

* Pemilik Apotek

1. Membuat akun, melakukan pendaftaran dengan *username* dan *password*
2. *Login*, akun sudah dibuat, maka Pemilik dapat masuk ke halaman Pemilik
3. Jika sudah pemilik dapat melihat, meng-*input*, *edit* dan menghapus data obat, data pemasokan, data *member*, menentukan harga obat, data transaksi, dan pengecekan obat.

* Apoteker
  1. Membuat akun, melakukan pendaftaran dengan *username* dan *password*.
  2. *Login*, akun sudah dibuat, maka Apoteker dapat masuk ke halaman Apoteker.
  3. Jika sudah Apoteker dapat melihat dan mencari data obat, menambahkan resep, menambahkan dan mencari data *member*, serta menginput data transaksi.
* Gudang
  1. Membuat akun untuk bagian manager, melakukan pendaftaran dengan *username* dan *password*
  2. *Login*, jika terverifikasi maka akan masuk ke halaman Gudang
  3. Jika sudah masuk akan terdapat fitur merekap dan mendata stok obat, lalu juga ada notifikasi menerima laporan

## Asumsi dan Dependensi

Asumsi – asumsi pada Sistem Informasi Apotek ini antara lain:

### Sistem operasi yang harus tersedia pada suatu platform perangkat keras dapat berupa Microsoft Windows XP/Vista/7.

### Tersedia sarana web server untuk dapat mengakses aplikasi ini.

### Pengguna baik Pemilik Apotek maupun Apoteker harus memiliki pengetahuan dasar tentang cara mengoperasikan komputer

### Administrator akan memiliki hak yang memadai untuk mengakses database administrasi

# Requirements Antarmuka Eksternal

## Antarmuka Pengguna

Pengguna berinteraksi dengan perangkat lunak melalui antarmuka berbasis GUI yang dapat menampilkan menu beserta fungsi nya kepada pengguna melalui aplikasi. Perangkat lunak mendapat masukan berupa teks melalui *keyboard*, gambar melalui *scanner*, dan navigasi klik melalui *mouse*. Keluaran dari perangkat ini dapat dilihat dapat oleh pengguna menggunakan *monitor* atau dicetak menggunakan *printer*.

## Antarmuka Perangkat Keras

Aplikasi ini berjalan di atas perangkat keras berupa beberapa komputer dan terhubung satu sama lain melalui *web server*. File aplikasi atau perangkat lunak ini ditempatkan pada *web hosting* yang dioperasikan oleh *administrator,* yaitu pemilik apotek.

## Antarmuka Perangkat Lunak

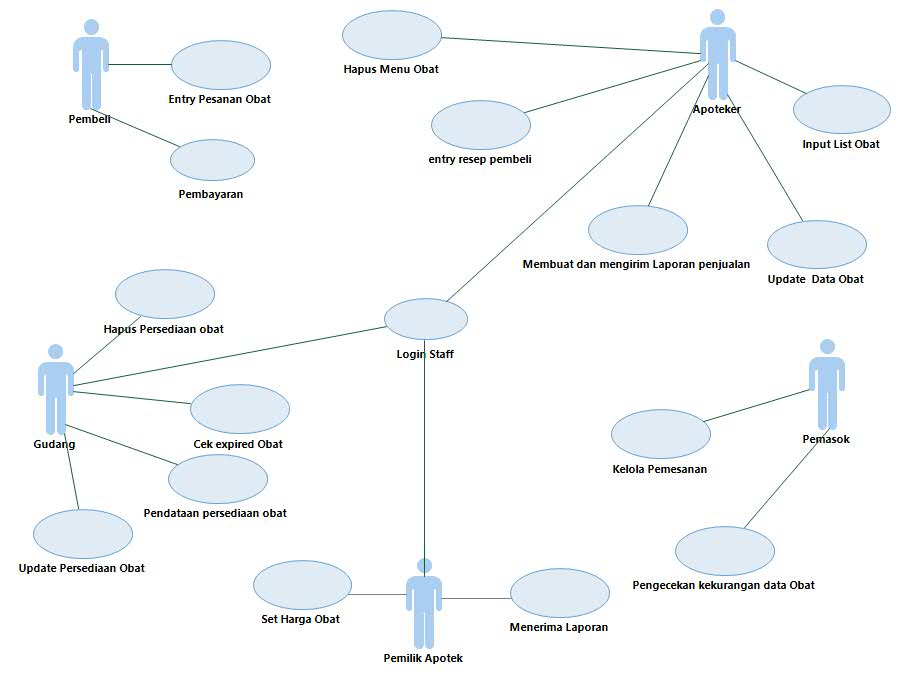
Perangkat lunak ini dibangun dengan *web server* Apache 2.4.17 dan *platform web server* XAMPP *for* Windows. Bahasa Pemrograman yang digunakan adalah Java. Pengolah *database* yang digunakan adalah MySQL karena dapat menyimpan data secara baik  dan memiliki sistem *back-up* yang baik.

Perangkat lunak dapat dijalankan pada sistem operasi apapun. Adapun sistem operasi yang disarankan adalah Microsoft Windows XP/Vista/7.

## Antarmuka Komunikasi

Antarmuka komunikasi yang dibutuhkan adalah sebuah komputer *server* dan beberapa komputer *client* yang terhubung secara *client-server* dalam lingkup jaringan Intranet atau Local Area Network berbasis protokol *Transmission Control Protocol/Internet Protocol* (TCP/IP).

# Fitur Sistem (Use Cases)



## Use Case 1

### Nama Use Case: Hapus Data Obat

### Tujuan: Apoteker menghapuskan data obat-obatan yang telah habis persediaannya di apotek sehingga tidak dapat dipesan kembali oleh pembeli

### Input: Data obat yang akan dihapus

### Output: Data obat terhapus

### Skenario Utama: Apoteker menghapus data obat, data obat terhapus dari sistem dan sistem tidak lagi menampilkan data obat tersebut pada menu obat

### Prakondisi: Apoteker ingin menghapus data obat yang telah habis persediaannya

### Langkah-langkah:

#### Langkah 1: Apoteker melakukan login.

#### Langkah 2: Sistem memeriksa valid atau tidaknya data login yang diinputkan.

#### Langkah 3: Sistem menampilkan halaman utama apabila proses login berhasil.

#### Langkah 4: Apoteker memilih menu obat

#### Langkah 5: Sistem menampilkan menu obat

#### Langkah 6: Apoteker memilih data obat yang akan dihapus

#### Langkah 7: Sistem menampilkan data obat yang dipilih

#### Langkah 8: Apoteker memilih opsi hapus data obat

#### Langkah 9: Sistem menampilkan pesan konfirmasi apakah data obat benar-benar akan dihapus

#### Langkah 10: Apoteker menekan pilihan setuju

#### Langkah 11: Sistem menghapus data obat dari basis data

#### Langkah 12: Sistem menampilkan pesan bahwa data berhasil dihapus dan data tidak ditampilkan kembali pada menu obat

### Pascakondisi: Apoteker telah menghapus data obat yang habis persediaannya

### Skenario eksepsional 1:

#### Langkah 1: Apoteker melakukan login.

#### Langkah 2: Sistem memeriksa valid atau tidaknya data login yang di input.

#### Langkah 3: Sistem menampilkan halaman utama apabila proses login berhasil.

#### Langkah 4: Apoteker memilih menu obat

#### Langkah 5: Sistem menampilkan menu obat

#### Langkah 3: Apoteker memilih data obat yang akan dihapus

#### Langkah 6: Sistem menampilkan data obat yang dipilih

#### Langkah 7: Apoteker memilih opsi hapus data obat

#### Langkah 8: Sistem menampilkan pesan konfirmasi apakah data obat benar-benar akan dihapus

#### Langkah 9: Apoteker menekan pilihan tidak setuju

#### Langkah 10: Sistem mengembalikan pada daftar menu obat

## Use Case 2

### Nama Use Case: Entry Resep Pembeli

### Tujuan: Apoteker memasukkan resep pembelian

### Input: Resep pembelian

### Output: Resep pembelian disimpan dalam daftar resep pembelian

### Skenario Utama: Apoteker meng-input resep pembelian, sistem menyimpan dan memasukkannya kedalam daftar resep pembelian

### Prakondisi: Apoteker ingin menginputkan resep pembelian sebagai rekap

### Langkah-langkah:

#### Langkah 1: Apoteker melakukan login.

#### Langkah 2: Sistem memeriksa valid atau tidaknya data login yang diinputkan.

#### Langkah 3: Sistem menampilkan halaman utama apabila proses login berhasil.

#### Langkah 4: Apoteker memilih menu pembelian

#### Langkah 5: Sistem menampilkan menu pembelian

#### Langkah 6: Apoteker memilih opsi tambah resep pembelian

#### Langkah 7: Sistem menampilkan form penambahan resep

#### Langkah 8: Apoteker mengisi secara lengkap form penambahan resep dan mengkonfirmasi data yang di input dengan menekan tombol simpan

#### Langkah 9: Sistem menampilkan pesan konfirmasi apakah data resep yang dimasukkan telah valid

#### Langkah 10: Apoteker menekan pilihan setuju

#### Langkah 11: Sistem menyimpan resep pembelian yang di input ke dalam list resep pembelian

### Pascakondisi: Apoteker telah memasukkan data resep pembelian sebagai rekap

### Skenario eksepsional 1:

#### Langkah 1: Apoteker melakukan login.

#### Langkah 2: Sistem memeriksa valid atau tidaknya data login yang di input.

#### Langkah 3: Sistem menampilkan halaman utama apabila proses login berhasil.

#### Langkah 4: Apoteker memilih menu pembelian

#### Langkah 5: Sistem menampilkan menu pembelian

#### Langkah 6: Apoteker memilih opsi tambah resep pembelian

#### Langkah 7: Sistem menampilkan form penambahan resep

#### Langkah 8: Apoteker mengisi secara lengkap form penambahan resep dan mengkonfirmasi data yang di input dengan menekan tombol simpan

#### Langkah 9: Sistem menampilkan pesan konfirmasi apakah data resep yang dimasukkan telah valid

#### Langkah 10: Apoteker menekan pilihan tidak setuju

#### Langkah 11: Sistem mengembalikan pada halaman form penambahan resep

## Use Case 3

### Nama Use Case: Membuat dan mengirim laporan penjualan

### Tujuan: Apoteker membuat dan mengirim laporan penjualan kepada pemilik apotek secara periodik

### Input: Rekap data penjualan

### Output: Laporan penjualan

### Skenario Utama: Apoteker membuat laporan dari data penjualan yang ada, lalu apoteker melaporkan atau mengirimkan laporan kepada pemilik apotek

### Prakondisi: Apoteker ingin membuat laporan penjualan untuk dikirimkan kepada pemilik apotek

### Langkah-langkah:

#### Langkah 1: Apoteker melakukan login.

#### Langkah 2: Sistem memeriksa valid atau tidaknya data login yang di input.

#### Langkah 3: Sistem menampilkan halaman utama apabila proses login berhasil.

#### Langkah 4: Apoteker memilih menu data penjualan

#### Langkah 5: Sistem menampilkan menu data penjualan

#### Langkah 6: Apoteker membuat laporan berdasarkan rekap data penjualan secara periodik

#### Langkah 7: Apoteker memilih opsi kirim untuk mengirim laporan kepada pemilik apotek

#### Langkah 8: Sistem menampilkan pesan konfirmasi apakah yakin ingin mengirim laporan

#### Langkah 9: Apoteker menekan tombol yakin

#### Langkah 10: Sistem mengirimkan laporan kepada pemilik apotek

### Pascakondisi: Apoteker telah membuat dan mengirimkan laporan penjualan kepada pemilik apotek

### Skenario eksepsional 1:

#### Langkah 1: Apoteker melakukan login.

#### Langkah 2: Sistem memeriksa valid atau tidaknya data login yang diinputkan.

#### Langkah 3: Sistem menampilkan halaman utama apabila proses login berhasil.

#### Langkah 4: Apoteker memilih menu data penjualan

#### Langkah 5: Sistem menampilkan menu data penjualan

#### Langkah 6: Apoteker membuat laporan berdasarkan rekap data penjualan secara periodik

#### Langkah 7: Apoteker memilih opsi kirim untuk mengirim laporan kepada pemilik apotek

#### Langkah 8: Sistem menampilkan pesan konfirmasi apakah yakin ingin mengirim laporan

#### Langkah 9: Apoteker menekan tombol tidak yakin

#### Langkah 10: Sistem mengantarkan apoteker ke halaman data penjualan

## Use Case 4

### Nama Use Case: Update data obat

### Tujuan: Apoteker meng-update data obat yang sudah ada

### Input: Informasi obat

### Output: Data obat yang sudah ter-update­

### Skenario Utama: Apoteker meng-update data obat yang sudah ada, sistem menyimpan data obat yang sudah ter-update­

### Prakondisi: Apoteker ingin meng­-update data obat yang sudah ada

### Langkah-langkah:

#### Langkah 1: Apoteker melakukan login

#### Langkah 2: Sistem memeriksa valid atau tidaknya data login yang di input

#### Langkah 3: Sistem menampilkan halaman utama apabila proses login berhasil

#### Langkah 4: Apoteker memilih menu obat

#### Langkah 5: Sistem menampilkan menu obat

#### Langkah 6: Apoteker memilih data obat yang akan di update

#### Langkah 7: Sistem menampilkan informasi obat yang dipilih

#### Langkah 8: Apoteker memilih opsi update data obat

#### Langkah 9: Apoteker melakukan perubahan pada informasi obat

#### Langkah 10: Sistem menampilkan pesan konfirmasi apakah data obat benar-benar akan di update

#### Langkah 11: Apoteker menekan pilihan setuju

#### Langkah 12: Sistem menyimpan update pada data obat dari basis data

#### Langkah 13: Sistem menampilkan pesan bahwa data berhasil di update

### Pascakondisi: Apoteker telah meng-update data obat

### Skenario eksepsional 1:

#### Langkah 1: Apoteker melakukan login

#### Langkah 2: Sistem memeriksa valid atau tidaknya data login yang di input

#### Langkah 3: Sistem menampilkan halaman utama apabila proses login berhasil

#### Langkah 4: Apoteker memilih menu obat

#### Langkah 5: Sistem menampilkan menu obat

#### Langkah 6: Apoteker memilih data obat yang akan di update

#### Langkah 7: Sistem menampilkan informasi obat yang dipilih

#### Langkah 8: Apoteker memilih opsi update data obat

#### Langkah 9: Apoteker melakukan perubahan pada informasi obat

#### Langkah 10: Sistem menampilkan pesan konfirmasi apakah data obat benar-benar akan di update

#### Langkah 11: Apoteker menekan pilihan tidak setuju

#### Langkah 12: Sistem mengantarkan apoteker ke halaman update obat

## Use Case 5

### Nama Use Case: Input list obat

### Tujuan: Apoteker menambahkan data obat baru

### Input: Informasi obat

### Output: Data obat baru

### Skenario Utama: Apoteker menambahkan data obat yang belum ada, sistem menyimpan data obat yang baru

### Prakondisi: Apoteker ingin menambahkan data obat baru

### Langkah-langkah:

#### Langkah 1: Apoteker melakukan login

#### Langkah 2: Sistem memeriksa valid atau tidaknya data login yang di input

#### Langkah 3: Sistem menampilkan halaman utama apabila proses login berhasil

#### Langkah 4: Apoteker memilih menu obat

#### Langkah 5: Sistem menampilkan menu obat

#### Langkah 6: Apoteker memilih opsi tambah obat

#### Langkah 7: Sistem menampilkan form tambah obat

#### Langkah 8: Apoteker memasukkan informasi obat

#### Langkah 9: Apoteker menekan tombol simpan saat sudah selesai meng-input data obat

#### Langkah 10: Sistem menampilkan pesan konfirmasi apakah data obat sudah benar

#### Langkah 11: Apoteker menekan pilihan setuju

#### Langkah 12: Sistem menyimpan data obat

#### Langkah 13: Sistem menampilkan data obat pada menu obat

### Pascakondisi: Apoteker telah menambahkan data obat baru

### Skenario eksepsional 1:

#### Langkah 1: Apoteker melakukan login

#### Langkah 2: Sistem memeriksa valid atau tidaknya data login yang di input

#### Langkah 3: Sistem menampilkan halaman utama apabila proses login berhasil.

#### Langkah 4: Apoteker memilih menu obat

#### Langkah 5: Sistem menampilkan menu obat

#### Langkah 6: Apoteker memilih opsi tambah obat

#### Langkah 7: Sistem menampilkan form tambah obat

#### Langkah 8: Apoteker memasukkan informasi obat

#### Langkah 9: Apoteker menekan tombol simpan saat sudah selesai meng-input data obat

#### Langkah 10: Sistem menampilkan pesan konfirmasi apakah data obat sudah benar

#### Langkah 11: Apoteker menekan pilihan tidak setuju

#### Langkah 12: Sistem mengantarkan apoteker ke halaman form tambah obat

## Use Case 6

### Nama Use Case: Set harga obat

### Tujuan: Pemilik apotek ingin mengubah data harga obat yang lama menjadi data harga obat yang baru

### Input: Harga obat yang baru

### Output: Data harga obat baru disimpan dan ditampilkan pada sistem

### Skenario Utama: Pemilik mengubah data harga obat lama menjadi data harga obat yang baru, sistem menyimpan perubahan dan menampilkan data harga obat yang terbaru

### Prakondisi: Pemilik apotek ingin melakukan set harga obat

### Langkah-langkah:

#### Langkah 1: Pemilik apotek melakukan login

#### Langkah 2: Sistem memeriksa valid atau tidaknya data login yang di input

#### Langkah 3: Sistem menampilkan halaman utama apabila proses login berhasil

#### Langkah 4: Pemilih apotek memilih menu obat

#### Langkah 5: Sistem menampilkan menu obat

#### Langkah 6: Pemilik apotek memilih obat yang akan diubah harganya

#### Langkah 7: Sistem menampilkan data obat yang dipilih

#### Langkah 8: Pemilik apotek memilih opsi set harga obat

#### Langkah 9: Sistem menampilkan form harga obat

#### Langkah 10: Pemilik apotek mengubah harga obat yang lama menjadi harga obat yang baru dan menekan tombol simpan

#### Langkah 11: Sistem menampilkan pesan konfirmasi apakah harga obat akan benar-benar di set menjadi harga yang baru

#### Langkah 12: Pemilik apotek menekan tombol setuju

#### Langkah 13: Sistem menyimpan data harga obat yang baru dalam basis data

### Pascakondisi: Pemilik apotek telah mengubah data harga obat yang lama menjadi data harga obat yang baru

### Skenario eksepsional 1:

#### Langkah 1: Pemilik apotek melakukan login

#### Langkah 2: Sistem memeriksa valid atau tidaknya data login yang di input

#### Langkah 3: Sistem menampilkan halaman utama apabila proses login berhasil

#### Langkah 4: Pemilih apotek memilih menu obat

#### Langkah 5: Sistem menampilkan menu obat

#### Langkah 6: Pemilik apotek memilih obat yang akan diubah harganya

#### Langkah 7: Sistem menampilkan data obat yang dipilih

#### Langkah 8: Pemilik apotek memilih opsi set harga obat

#### Langkah 9: Sistem menampilkan form harga obat

#### Langkah 10: Pemilik apotek mengubah harga obat yang lama menjadi harga obat yang baru dan menekan tombol simpan

#### Langkah 11: Sistem menampilkan pesan konfirmasi apakah harga obat akan benar-benar diset menjadi harga yang baru

#### Langkah 12: Pemilik apotek menekan tombol tidak setuju

#### Langkah 13: Sistem akan kembali pada halaman form harga obat

## Use Case 7

### Nama Use Case: Menerima laporan

### Tujuan: Pemilik apotek memeriksa laporan sebagai bukti penjualan yang telah dikirimkan oleh apoteker untuk mengetahui aktivitas dari apotek secara periodik dan sebagai pertimbangan dalam mengambil keputusan kedepannya

### Input: Laporan penjualan yang masuk

### Output: Laporan penjualan ditandai telah diperiksa

### Skenario Utama: Pemilik menerima dan memeriksa setiap laporan penjualan yang dikirimkan oleh apoteker, sistem kemudian menandai laporan mana saja yang telah selesai diperiksa oleh pemilik apotek

### Prakondisi: Pemilik apotek ingin memeriksa laporan penjualan yang telah dikirimkan oleh apoteker

### Langkah-langkah:

#### Langkah 1: Pemilik apotek melakukan login

#### Langkah 2: Sistem memeriksa valid atau tidaknya data login yang di input

#### Langkah 3: Sistem menampilkan halaman utama apabila proses login berhasil

#### Langkah 4: Pemilih apotek memilih menu laporan penjualan

#### Langkah 5: Sistem menampilkan menu laporan penjualan

#### Langkah 6: Pemilik apotek memilih laporan penjualan untuk diperiksa

#### Langkah 7: Sistem menampilkan laporan penjualan yang dipilih

#### Langkah 8: Pemilik apotek memeriksa laporan penjualan yang masuk

#### Langkah 9: Pemilik menyetujui laporan penjualan apabila dinilai kevalidannya

#### Langkah 10: Sistem menandai laporan yang telah diperiksa dan divalidasi

### Pascakondisi: Pemilik apotek telah memeriksa laporan penjualan yang dikirimkan oleh apoteker

### Skenario eksepsional 1:

#### Langkah 1: Pemilik apotek melakukan login

#### Langkah 2: Sistem memeriksa valid atau tidaknya data login yang di input

#### Langkah 3: Sistem menampilkan halaman utama apabila proses login berhasil

#### Langkah 4: Pemilih apotek memilih menu laporan penjualan

#### Langkah 5: Sistem menampilkan menu laporan penjualan

#### Langkah 6: Pemilik apotek memilih laporan penjualan untuk diperiksa

#### Langkah 7: Sistem menampilkan laporan penjualan yang dipilih

#### Langkah 8: Pemilik apotek memeriksa laporan penjualan yang masuk

#### Langkah 9: Pemilik tidak menyetujui laporan penjualan apabila dinilai terdapat data yang tidak valid

#### Langkah 10: Sistem akan memberi pesan notifikasi kepada apoteker bahwa laporan penjualan yang dikirimkan belum sesuai

#### Langkah 11: Sistem mengembalikan pada menu laporan penjualan

## Use Case 8

### Nama Use Case: Hapus Persediaan Obat

### Tujuan: Gudang melakukan penghapusan terhadap obat yang sudah kadaluarsa atau tidak layak dijualbelikan.

### Input: Data obat yang sudah kadaluarsa atau tidak layak pakai.

### Output: Data obat telah dihapus

### Skenario Utama: Gudang menerima laporan dari Apotek dan memeriksa data kadaluarsa obat. Jika obat sudah kadaluarsa atau tidak layak dijualbelikan, maka obat akan dihapuskan atau dimusnahkan

### Prakondisi: Gudang menerima laporan dari Apotek

### Langkah-langkah:

#### Langkah 1: Pemilik apotek melakukan login

#### Langkah 2: Sistem memeriksa valid atau tidaknya data login yang di input

#### Langkah 3: Sistem menampilkan halaman utama apabila proses login berhasil

#### Langkah 4: Pemilih apotek memilih menu laporan penjualan

#### Langkah 5: Sistem menampilkan menu laporan penjualan

#### Langkah 6: Pemilik apotek memilih laporan penjualan untuk diperiksa

#### Langkah 7: Sistem menampilkan laporan penjualan yang dipilih

#### Langkah 8: Pemilik apotek memeriksa laporan penjualan yang masuk

#### Langkah 9: Pemilik menyetujui laporan penjualan apabila dinilai kevalidannya

#### Langkah 10: Sistem menandai laporan yang telah diperiksa dan divalidasi

### Pascakondisi: Gudang menghapus data obat yang sudah kadaluarsa

### Skenario eksepsional 1:

#### Langkah 1: Pemilik apotek melakukan login

#### Langkah 2: Sistem memeriksa valid atau tidaknya data login yang di input

#### Langkah 3: Sistem menampilkan halaman utama apabila proses login berhasil

#### Langkah 4: Pemilih apotek memilih menu laporan penjualan

#### Langkah 5: Sistem menampilkan menu laporan penjualan

#### Langkah 6: Pemilik apotek memilih laporan penjualan untuk diperiksa

#### Langkah 7: Sistem menampilkan laporan penjualan yang dipilih

#### Langkah 8: Pemilik apotek memeriksa laporan penjualan yang masuk

#### Langkah 9: Pemilik tidak menyetujui laporan penjualan apabila dinilai terdapat data yang tidak valid

#### Langkah 10: Sistem akan memberi pesan notifikasi kepada apoteker bahwa laporan penjualan yang dikirimkan belum sesuai

#### Langkah 11: Sistem mengembalikan pada menu laporan penjualan

## Use Case 9

### Nama Use Case: Cek Expired Obat

### Tujuan: Gudang memeriksa tanggal kadaluarsa pada setiap obat

### Input: Menerima stok obat dari pemasok

### Output: Mengecek lalu memilah obat yang sudah dicek kadaluarsanya

### Skenario Utama: Gudang menerima stok obat dari pemasok dan memeriksa data kadaluarsa obat. Jika obat sudah kadaluarsa atau tidak layak dijualbelikan, maka obat akan dipilah.

### Prakondisi: Gudang menerima stok obat dari pemasok

### Langkah-langkah:

#### Langkah 1: Pemilik apotek melakukan login

#### Langkah 2: Sistem memeriksa valid atau tidaknya data login yang di input

#### Langkah 3: Sistem menampilkan halaman utama apabila proses login berhasil

#### Langkah 4: Pemilih apotek memilih menu laporan penjualan

#### Langkah 5: Sistem menampilkan menu laporan penjualan

#### Langkah 6: Pemilik apotek memilih laporan penjualan untuk diperiksa

#### Langkah 7: Sistem menampilkan laporan penjualan yang dipilih

#### Langkah 8: Pemilik apotek memeriksa laporan penjualan yang masuk

#### Langkah 9: Pemilik menyetujui laporan penjualan apabila dinilai kevalidannya

#### Langkah 10: Sistem menandai laporan yang telah diperiksa dan divalidasi

### Pascakondisi: Gudang memilah data obat yang sudah kadaluarsa

### Skenario eksepsional 1:

#### Langkah 1: Pemilik apotek melakukan login

#### Langkah 2: Sistem memeriksa valid atau tidaknya data login yang di input

#### Langkah 3: Sistem menampilkan halaman utama apabila proses login berhasil

#### Langkah 4: Pemilih apotek memilih menu laporan penjualan

#### Langkah 5: Sistem menampilkan menu laporan penjualan

#### Langkah 6: Pemilik apotek memilih laporan penjualan untuk diperiksa

#### Langkah 7: Sistem menampilkan laporan penjualan yang dipilih

#### Langkah 8: Pemilik apotek memeriksa laporan penjualan yang masuk

#### Langkah 9: Pemilik tidak menyetujui laporan penjualan apabila dinilai terdapat data yang tidak valid

#### Langkah 10: Sistem akan memberi pesan notifikasi kepada apoteker bahwa laporan penjualan yang dikirimkan belum sesuai

#### Langkah 11: Sistem mengembalikan pada menu laporan penjualan

## Use Case 10

### Nama Use Case: Pendataan persediaan obat

### Tujuan: Gudang mengirim stok obat ke Apotek

### Input: Menerima laporan dari Apotek

### Output: Mengirim stok obat yang baru

### Skenario Utama: Gudang menerima laporan dari Apotek dan mengirim stok obat yang baru.

### Prakondisi: Gudang menerima laporan dari Apotek

### Langkah-langkah:

#### Langkah 1: Pemilik apotek melakukan login

#### Langkah 2: Sistem memeriksa valid atau tidaknya data login yang di input

#### Langkah 3: Sistem menampilkan halaman utama apabila proses login berhasil

#### Langkah 4: Pemilih apotek memilih menu laporan penjualan

#### Langkah 5: Sistem menampilkan menu laporan penjualan

#### Langkah 6: Pemilik apotek memilih laporan penjualan untuk diperiksa

#### Langkah 7: Sistem menampilkan laporan penjualan yang dipilih

#### Langkah 8: Pemilik apotek memeriksa laporan penjualan yang masuk

#### Langkah 9: Pemilik menyetujui laporan penjualan apabila dinilai kevalidannya

#### Langkah 10: Sistem menandai laporan yang telah diperiksa dan divalidasi

### Pascakondisi: Gudang mengirim stok obat yang baru ke Apotek

### Skenario eksepsional 1:

#### Langkah 1: Pemilik apotek melakukan login

#### Langkah 2: Sistem memeriksa valid atau tidaknya data login yang di input

#### Langkah 3: Sistem menampilkan halaman utama apabila proses login berhasil

#### Langkah 4: Pemilih apotek memilih menu laporan penjualan

#### Langkah 5: Sistem menampilkan menu laporan penjualan

#### Langkah 6: Pemilik apotek memilih laporan penjualan untuk diperiksa

#### Langkah 7: Sistem menampilkan laporan penjualan yang dipilih

#### Langkah 8: Pemilik apotek memeriksa laporan penjualan yang masuk

#### Langkah 9: Pemilik tidak menyetujui laporan penjualan apabila dinilai terdapat data yang tidak valid

#### Langkah 10: Sistem akan memberi pesan notifikasi kepada apoteker bahwa laporan penjualan yang dikirimkan belum sesuai

*Langkah 11: Sistem mengembalikan pada menu laporan penjualan*

## Use Case 11

### Nama Use Case: Update persediaan obat

### Tujuan: Gudang memperbarui stok obat yang telah ada yang dikirimkan oleh pemasok

### Input: Menerima stok obat dari pemasok

### Output: Memperbarui stok obat yang telah ada

### Skenario Utama: Gudang mendapatkan stok obat baru dari pemasok, lalu meng-update data obat yang ada digudang.

### Prakondisi: Gudang menerima stok obat

### Langkah-langkah:

#### Langkah 1: Pemilik apotek melakukan login

#### Langkah 2: Sistem memeriksa valid atau tidaknya data login yang di input

#### Langkah 3: Sistem menampilkan halaman utama apabila proses login berhasil

#### Langkah 4: Pemilih apotek memilih menu laporan penjualan

#### Langkah 5: Sistem menampilkan menu laporan penjualan

#### Langkah 6: Pemilik apotek memilih laporan penjualan untuk diperiksa

#### Langkah 7: Sistem menampilkan laporan penjualan yang dipilih

#### Langkah 8: Pemilik apotek memeriksa laporan penjualan yang masuk

#### Langkah 9: Pemilik menyetujui laporan penjualan apabila dinilai kevalidannya

#### Langkah 10: Sistem menandai laporan yang telah diperiksa dan divalidasi

### Pascakondisi: Gudang memperbarui stok obat.

### Skenario eksepsional 1:

#### Langkah 1: Pemilik apotek melakukan login

#### Langkah 2: Sistem memeriksa valid atau tidaknya data login yang di input

#### Langkah 3: Sistem menampilkan halaman utama apabila proses login berhasil

#### Langkah 4: Pemilih apotek memilih menu laporan penjualan

#### Langkah 5: Sistem menampilkan menu laporan penjualan

#### Langkah 6: Pemilik apotek memilih laporan penjualan untuk diperiksa

#### Langkah 7: Sistem menampilkan laporan penjualan yang dipilih

#### Langkah 8: Pemilik apotek memeriksa laporan penjualan yang masuk

#### Langkah 9: Pemilik tidak menyetujui laporan penjualan apabila dinilai terdapat data yang tidak valid

#### Langkah 10: Sistem akan memberi pesan notifikasi kepada apoteker bahwa laporan penjualan yang dikirimkan belum sesuai

*Langkah 11: Sistem mengembalikan pada menu laporan penjualan*

# *Requirements* Nonfungsional Lainnya

## *Requirements* Performa

Perangkat lunak ini beroperasi selama jam kerja di apotek. Minimal 12 jam per hari tanpa henti. Karena perangkat ini berbasis GUI dan di gunakan oleh apoteker dan pemilik maka harus siap di gunakan kapan saja.

Kebutuhan kinerja dari sistem memiliki respon kurang dari 30 detik untuk setiap aksi pergantian page. Sistem memiliki waktu respon kurang dari 2 menit untuk setiap proses load data. Sistem memiliki waktu respon kurang dari 1 menit untuk proses pencarian data.

## *Requirements* Keselamatan

Aplikasi ini akan sering di update *software* secara *online* oleh pengembang/*developer*. Karena akan ada banyak data dan laporan yang terus bertambah. Dengan adanya *update, software* akan menghindari *thread* dan *spam*.

## *Requirements* Keamanan

Perangkat ini harus dibangun dengan pondasi *security* dan *firewall* yang kuat dan fleksibel. Karena pentingnya data – data yang di rekap di perangkat lunak tersebut. Sehingga data dan informasi akan aman jika terkena *virus*.

Proses keamaan pada data *password* *developer* menggunakan enkripsi vignere. Dengan menggunakan Teknik enkripsi, maka kerahasiaan data *developer* akan terjamin. Untuk *client* menggunakan enkripsi md5. Lalu untuk aplikasi nya sendiri menggunakan *antivirus* dan *firewall*.

## Atribut Kualitas Perangkat Lunak

### Adaptasi

Perangkat lunak ini memiliki kelebihan dapat beradaptasi lebih cepat dengan lingkungan penggunaannya. Karena perangkat lunak ini dirancang dengan sifat *user-friendly*, pengguna baik pemilik apotek maupun apoteker tidak membutuhkan waktu yang lama untuk menyesuaikan diri dengan perangkat lunak yang digunakan (kemudahan dalam belajar)

### Ketersediaan

Perangkat lunak ini dapat berjalan dan tersedia selama tidak mengalami kendala, seperti suplai tenaga listrik atau terkendalanya jaringan lokal serta *web server hosting* sistem.

### Ketepatan

Perangkat lunak ini dirancang dengan tujuan yang dapat mendukung fungsi operasi, menajemen, dan pengambilan keputusan dalam organisasi Apotek itu sendiri, baik keputusan-keputusan rutin maupun keputusan-keputusan yang strategis. Sebelumnya segala proses transaksi dalam Apotek dicatat atau direkap secara manual. Dengan adanya perangkat lunak ini, dapat dikatakan dapat berfungsi tepat dan memenuhi kebutuhan dari Apotek.

### Keandalan

Perangkat lunak ini dapat digunakan selama 1 x 24 jam, dengan dukungan sistem operasi Windows yang memiliki stabilitas yang tinggi.

### Keamanan

User baik pemilik apotek maupun apoteker harus melakukan *login* terlebih dahulu untuk dapat mengakses sistem dan didukung oleh keandalan dari sistem. Data *login* sendiri harus berupa *user id* yang unik dan disertai dengan *password*. Setiap pengguna memiliki hak akses yang berbeda sesuai dengan kategorinya. Site dilengkapi dengan enkripsi md5 yang meng-enkripsi setiap data yang dikirim melalui *web server*.

### Kemerawatan

Perangkat lunak ini dirancang secara *full* parameter dan dinamis. Isi *site*, serta informasi yang terdapat didalamnya dapat di *update* kapan saja sesuai dengan informasi transaksi penjualan dan pembelian. Pemilik apotek hanya perlu mengakses perangkat lunak sebagai *administrator* dan mengakses menu yang khusus disediakan untuk *administrator.*

### Kepemindahan

Perangkat lunak ini tidak bersifat *portable*, karena berjalan khusus untuk mendukung fungsi operasi dari Apotek Putra Farma saja.

# *Requirements* Lain

## *Memory*

## Dalam penggunaan, memori akan sangat penting karena semua data dan laporan akan disimpan didalam memori dan diteruskan pada database.

## *Design*

## Aplikasi ini menggunakan desain yang sederhana terutama dalma hal fitur-fitur penting, sehingga *client* akan lebih mudah menggunakan aplikasi ini.

# Lampiran A: Daftar Kata-Kata Sukar

1. SKPL: Spesifikasi Kebutuhan Perangkat Lunak
2. GUI: *Graphic User Interface*
3. *User-friendly­*: merupakan suatu kondisi dimana kita bisa menggunakan sesuatu dengan mudah dan nyaman.
4. *Interface*: merupakan mekanisme komunikasi antara pengguna (user) dengan sistem.
5. Enkripsi md5: fungsi hash kriptografik yang digunakan secara luas dengan *hash value* 128-bit.