**DPPL**-xxxx

DESKRIPSI PERANCANGAN PERANGKAT LUNAK

Sistem Informasi Perpustakaan

untuk:

<nama pelanggan>

Dipersiapkan oleh:

1. Irsyad Fauzan Nurdin (1301164227)

2. Mutiara Ramadhani Wijaya (1301164191)

3. Pungki Nurhudha (1301160447)

4. Wisnu Adi Nurcahyo (1301160479)

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 |
| *DPPL-xxx* | | 16 |
| 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  TGL | - | A | B | C | D | E | F | G |
| Ditulis oleh |  |  |  |  |  |  |  |  |
| Diperiksa oleh |  |  |  |  |  |  |  |  |
| Disetujui oleh |  |  |  |  |  |  |  |  |

Daftar Halaman Perubahan

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

# Daftar Isi

[Daftar Isi 4](#_Toc473623651)

[1. Pendahuluan 5](#_Toc473623652)

[1.1 Tujuan Penulisan Dokumen 5](#_Toc473623653)

[1.2 Lingkup Masalah 5](#_Toc473623654)

[1.3 Definisi dan Istilah 5](#_Toc473623655)

[1.4 Aturan Penamaan dan Penomoran 5](#_Toc473623656)

[1.5 Referensi 5](#_Toc473623657)

[1.6 Ikhtisar Dokumen 5](#_Toc473623658)

[2 Deskripsi Perancangan Global 6](#_Toc473623659)

[2.1 Rancangan Lingkungan Implementasi 6](#_Toc473623660)

[2.2 Deskripsi Arsitektural 6](#_Toc473623661)

[2.3 Deskripsi Komponen 6](#_Toc473623662)

[3 Perancangan Rinci 7](#_Toc473623663)

[3.1 Realisasi Use Case 7](#_Toc473623664)

[3.1.1 Use Case <nama use case 1> 7](#_Toc473623665)

[3.2 Perancangan Detil Kelas 7](#_Toc473623666)

[3.2.1 Kelas <nama kelas> 7](#_Toc473623667)

[3.2.2 Kelas <nama kelas> 8](#_Toc473623668)

[3.3 Diagram Kelas Keseluruhan 8](#_Toc473623669)

[3.4 Algoritma/Query 8](#_Toc473623670)

[3.5 Diagram Statechart 8](#_Toc473623671)

[3.6 Perancangan Antarmuka 8](#_Toc473623672)

[3.7 Perancangan Representasi Persistensi Kelas 9](#_Toc473623673)

[4 Matriks Kerunutan 9](#_Toc473623674)

Setelah Daftar Isi Boleh ada Daftar Tabel dan Daftar Gambar

# 1. Pendahuluan

## Tujuan Penulisan Dokumen

Dokumen Deskripsi Perancangan Perangkat Lunak (DPPL) merupakan dokumen spesifikasi kebutuhan perangkat lunak yang akan dikembangkan.Tujuan dokumen ini digunakan oleh pengembang perangkat lunak sebagai acuan teknis pengembangan perangkat lunak pada tahap selanjutnya.

Dengan adanya penulisan dokumen Deskripsi Perancangan Perangkat Lunak (DPPL) ini megharapkan pengembang perangkat lunak pada tahap selanjutnya akan lebih terkonsep dan menarik pada saat pembangunannya.

## Lingkup Masalah

Sistem informasi perpustakaan ini merupakan perangkat lunak berupa sistem database perpustakaan. Sistem ini digunakan untuk penginputan dan penyimpanan data yang akan tersimpan lebih sistematis serta mempermudah pengunjung mencari dokumen atau data yang dicari hingga proses peminjaman buku.Kemudian sistem ini juga memudahkan dalam mengakses informasi persediaan buku yang diterima pengunjung perpustakaan dengan lebih cepat dan efektif yang dapat melakukan registrasi secara online. Sistem juga mengelola data anggota perpustakaan, dan informasi yang berhubungan dengan perpustakaan.

## Definisi dan Istilah

Berikut ini adalah beberapa definisi, singkatan, dan akronim yang terdapat di dalam dokumen ini:

### DPPL

Deskripsi Perangcangan Perancangan Perangkat Lunak (DPPL) merupakan mendeskripsikan dan menjabarkan secara tereperinci mengenai perancangan perangkat lunak yang dibuat.

### SPPL

SKPL adalah Spesifikasi Kebutuhan Perangkat Lunak, atau dalam bahasa Inggris-nya

sering juga disebut sebagai Software Requirements Spesification (SRS), dan merupakan spesifikasi dari perangkat lunak yang akan dikembangkan.

## Aturan Penamaan dan Penomoran

Penulisan dokumen SKPL ini menggunakan berbagai macam aturan penamaan dan penomoran yang berbeda-beda untuk beberapa bagian tertentu. Aturan penamaan dan penomoran yang digunakan berdasarkan hal/bagian tersebut adalah seperti yang tercantum pada Tabel T01 berikut ini.

**Tabel T01 Aturan Penamaan dan Penomoran**

| **Hal/Bagian** | **Aturan Penamaan dan Penomoran** |
| --- | --- |
| Bab | Tiap bab diberi nomor sesuai dengan urutannya dalam dokumen. Bila satu bab dibagi menjadi beberapa sub bab maka sub bab diberi nomor urut sesuai dengan urutannya pada bab tersebut. Antara nomor bab dan sub bab dipisahkan dengan tanda titik. |
| Tabel | Tiap tabel yang ada dinamai dengan TXX dengan XX adalah nomor urut tabel dalam dokumen. |
| Diagram | Tiap diagram yang ada dinamai dengan DXX dengan XX adalah nomor urut diagram dalam dokumen |

## Referensi

Referensi yang digunakan dalam pembuatan dokumen DPPL ini adalah sebagai berikut:

1. Dokumen Spesifikasi Kebutuhan Pearangkat Lunak (SKPL) Sistem Informasi Perpustakaan.
2. Modul Analisis dan Perancangan Perangkat Lunak Fakultas Informatika Telkom University.

## Ikhtisar Dokumen

Dokumen ini secara garis besar terdiri dari tiga bab dengan perincian sebagai berikut:

1. Bab 1 Pendahuluan, merupakan pengantar dokumen SKPL yang berisi tujuan penulisan dokumen, lingkup masalah pengembangan perangkat lunak, juga memuat definisi, akronim dan istilah yang digunakan serta deskripsi umum dokumen yang merupakan ikhtisar dokumen SKPL.
2. Bab 2 Deskripsi Global Perangkat Lunak, mendefinisikan rancangan lingkungan implementasi, deskripsi arsitektural, dan deskripsi komponen.

3. Bab 3 Deskripsi Rinci Kebutuhan, mendeskripsikan kebutuhan khusus bagi sistem informasi perpustakaan, yang meliputi kebutuhan realisasi use case , perancangan detil kelas, algoritma/query, perancangan antaramuka.

# Deskripsi Perancangan Global

## Rancangan Lingkungan Implementasi

Sistem ini diimplementasikan dalam lingkungan sebagai berikut :

a. Sistem informasi : Windows, Linux.

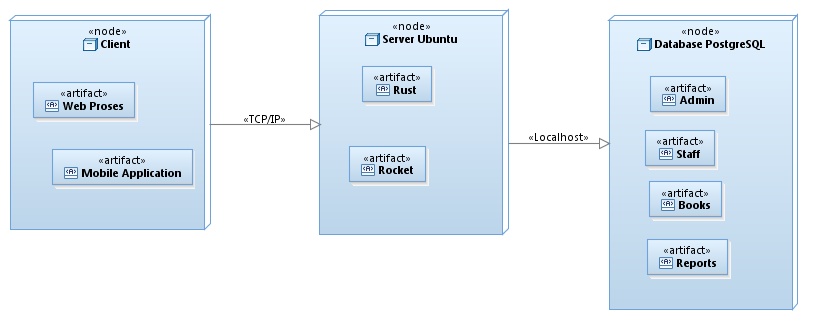
b. Bahasa Pemrograman : HTML, CSS, JavaScript, Rust JS.

c. Development Tools : Atom, VS Code.

## Deskripsi Arsitektural

Berikut merupakan komponen diagram yang digunakan untuk pengembangan sistem informasi perpustakaan:

**Diagram D01 komponen diagram**



## Deskripsi Komponen

Diisi dengan daftar modul. Daftar modul bisa dalam bentuk tabel berikut:

**Table T02 Deskripsi Komponen**

| **No** | **Nama Komponen** | **Keterangan** |
| --- | --- | --- |
| 1 | Login | Berfungsi sebagai pengalih halaman apabila login berhasil sesuai dengan tipe akun yang ada yaitu admin atau staff. |
| 2 | Books | Berfungsi sebagai modul yang berperan untuk mengatur buku informasi buku termasuk pengadaan buku. |
| 3 | Members | Berfungsi sebagai modul yang berperan untuk mengatur keanggotaan pengunjung. |
| 4 | Reports | Berfungsi sebagai pencatat aktivitas yang terjadi di dalam aplikasi ini. |

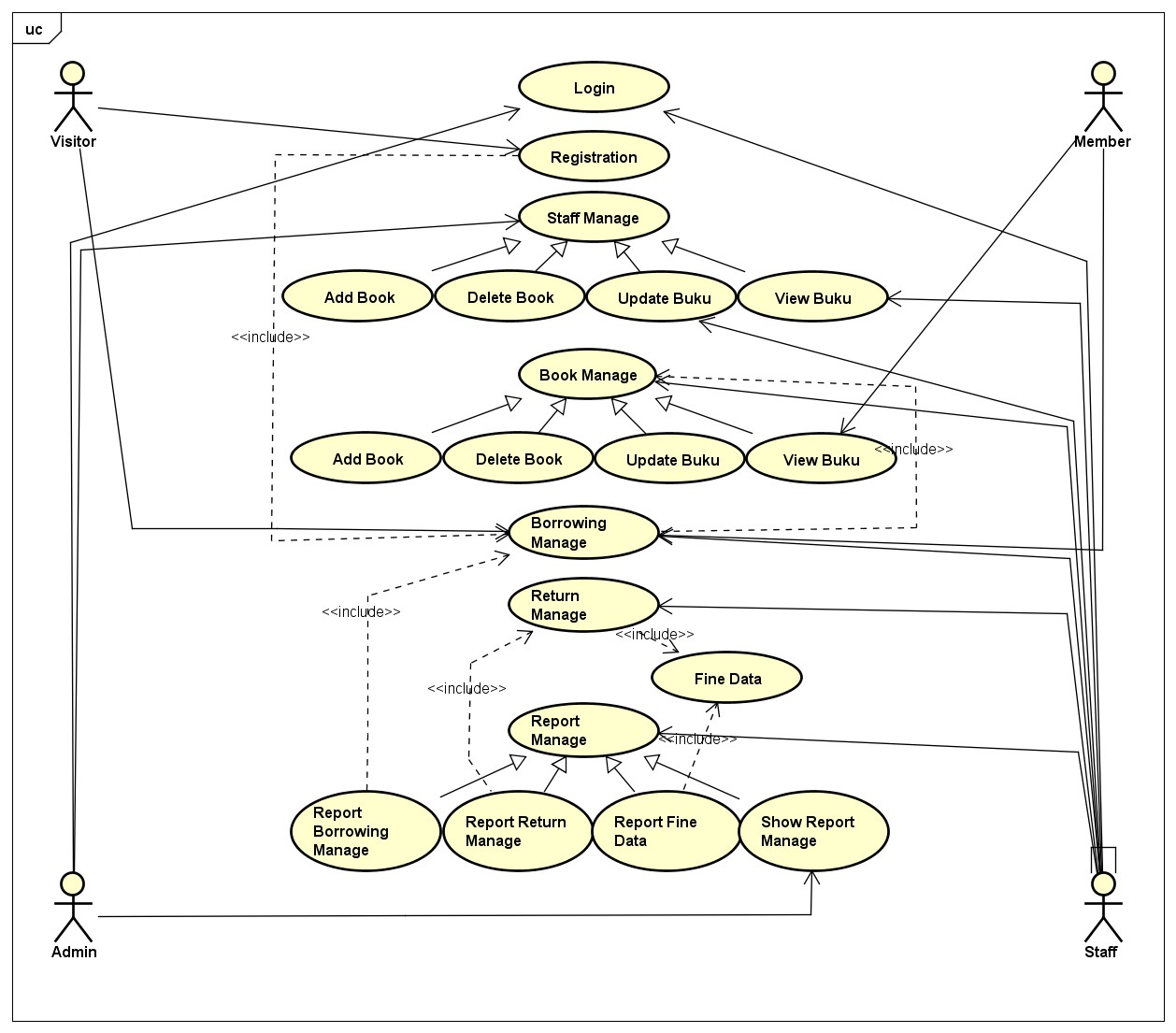
# Perancangan Rinci

## Realisasi Use Case

### Use Case Sistem Informasi Perpustakaan

Berikut merupakan rancangan usecase yang dibuat dalam pengembangan system informasi pengelolaan pendataan keuangan universitas:

**Diagram D02 Use Case Diagram**



#### Identifikasi Kelas

*Identifikasi kelas yang terkait dengan use case tersebut.Kelas di tahap perancangan dapat berbeda dengan dengan kelas di tahap analisis. Dapat menggunakan tabel di bawah:*

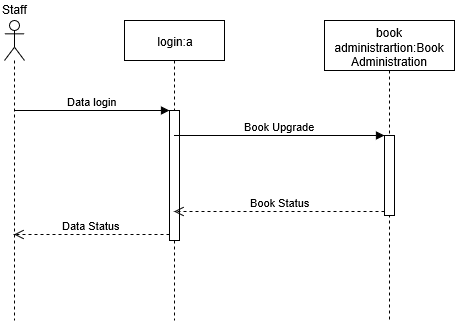
**Tabel T03 Identifikasi Kelas**

|  |  |  |
| --- | --- | --- |
| No | Nama Kelas Perancangan | Nama Kelas Analisis Terkait |
|  | Admin | * Login * Staff Manage |
|  | Staff | * Member * Login * Book Manage * Borrowing Manage * Return Manage * Report Manage |
|  | Member | * Borrowing Manage |
|  | Visitor | * Registration |

#### Sequence Diagram

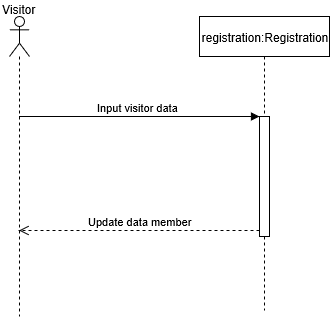
Login :

**Diagram D03 Sequence Diagram Login**



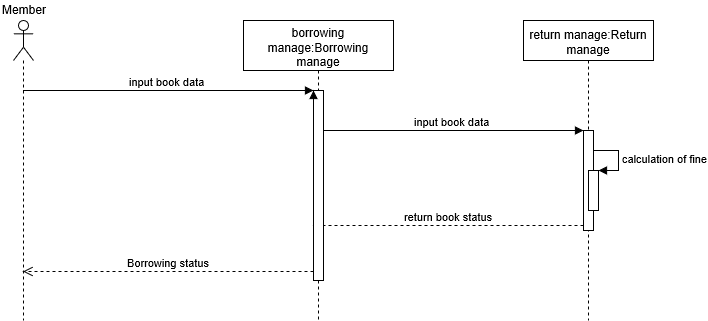
Registration:

**Diagram D04 Sequence Diagram Registration**



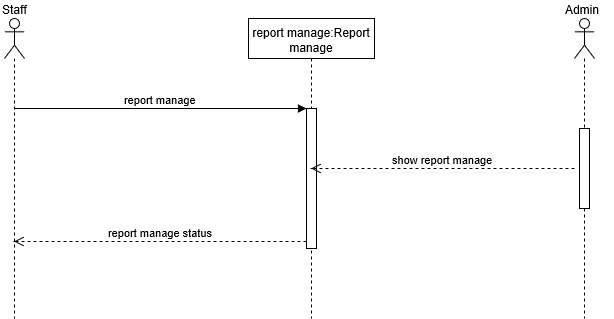
Borrowing and return manage :

**Diagram D05 Sequence Diagram Borrowing and Return Manage**

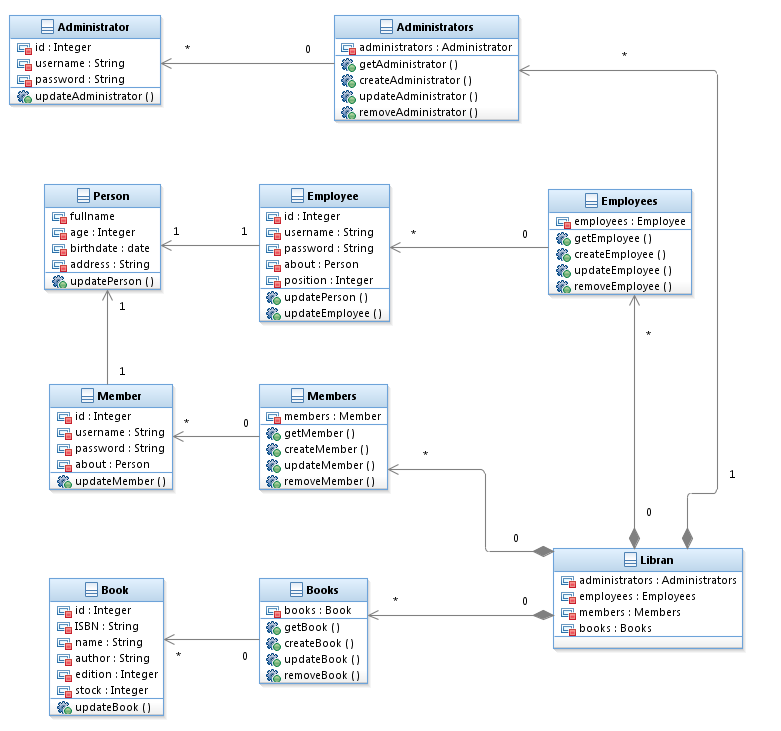


Report manage :

**Diagram D06 Sequence Diagram Report Manage**



#### Diagram Kelas

**

**Diagram D07 Diagram Kelas**

## Perancangan Detil Kelas

|  |  |  |
| --- | --- | --- |
| No | Nama Kelas Perancangan | Nama Kelas Analisis Terkait |
| 1 | Book | Books |
| 2 | Member | Members |
| 3 | Employee | Employees |

*Untuk setiap kelas:*

* *identifikasi operasi (mengacu pada tanggung-jawab kelas), termasuk visibility-nya*
* *identifikasi atribut, termasuk visibility-nya*

### Kelas <nama kelas>

Bagian ini diisi dengan daftar operasi dan atribut untuk setiap kelas.

*Nama Kelas : ……..*

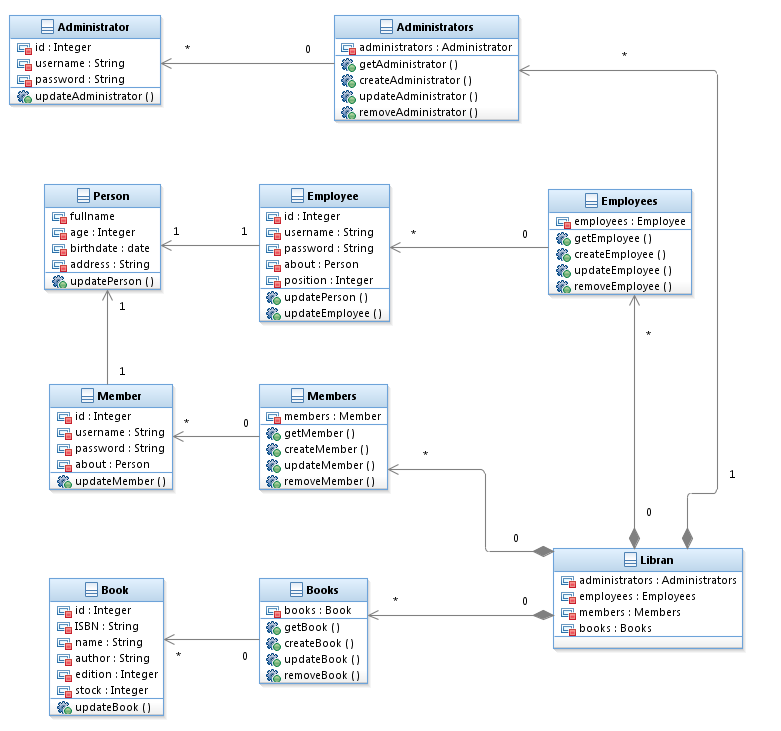
|  |  |  |
| --- | --- | --- |
| ***Nama Operasi*** | ***Visibility***  ***(private, public)*** | ***Keterangan*** |
| *Diisi dengan signature operasi* |  |  |
|  |  |  |
|  |  |  |
| ***Nama Atribut*** | ***Visibility***  ***(private, public)*** | ***Tipe*** |
| *Diisi dengan nama atribut* |  | *Tuliskan tipenya sesuai dengan yang dikenal pada bahasa pemrograman yang digunakan* |
|  |  |  |
|  |  |  |

### Kelas <nama kelas>

## Diagram Kelas Keseluruhan

*Bagian ini diisi dengan diagram kelas keseluruhan.*

**Diagram D08 diagram kelas keseluruhan**

**

## Algoritma/Query

*Bagian ini hanya diisi untuk kerangka algoritma untuk proses-proses yang dianggap cukup penting. Implementasi skeleton code juga sudah dapat dilakukan untuk kelas-kelas yang terdefinisi pada bahasa pemrograman tertentu. Boleh dibuat subbab per kelas.*

Nama Kelas: Book

Nama Operasi: updateBook

Algoritma:

let action

if action is lending then

book.stock - 1

report.add "lend book"

member.add book.isbn and duration

else if action is returns then

book.stock + 1

report add "return book"

if duration greater than limit then report.add "book fines"

member.remove book.isbn and duration

else if action is adds then

book = new book

report.add "book addition"

else

book = delete book

report.add "book removal"

*No Query: Q-1*

*Query:*

*UPDATE books*

*SET stock = current\_stock - 1*

*WHERE isbn = current\_isbn;*

*Keterangan: Mengubah stock buku sekarang.*

*No Query: Q-2*

*Query:*

*INSERT INTO reports VALUES (lend\_book);*

*Keterangan: Menambahkan laporan peminjaman buku.*

*No Query: Q-3*

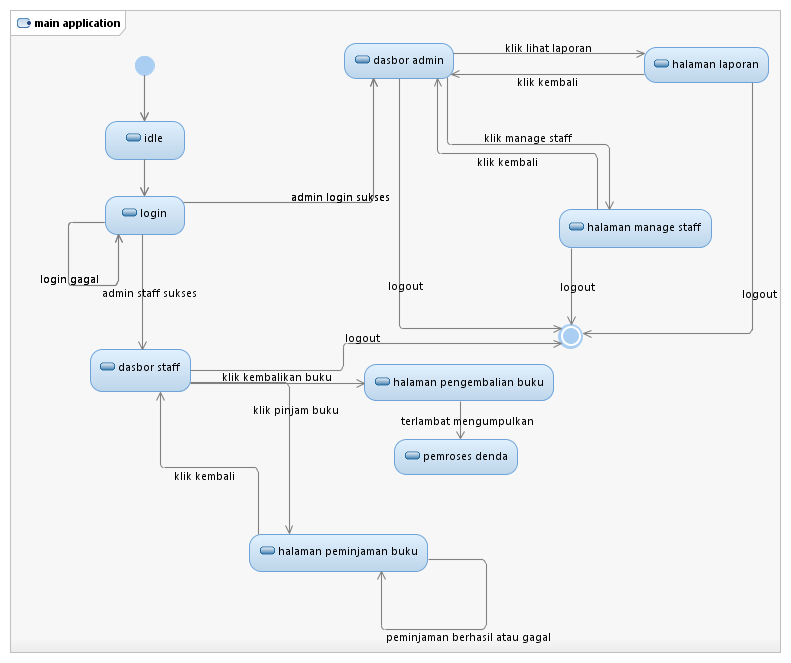
*Query:*

*INSERT INTO member\_books VALUES (member\_usercode, book\_isbn, duration);*

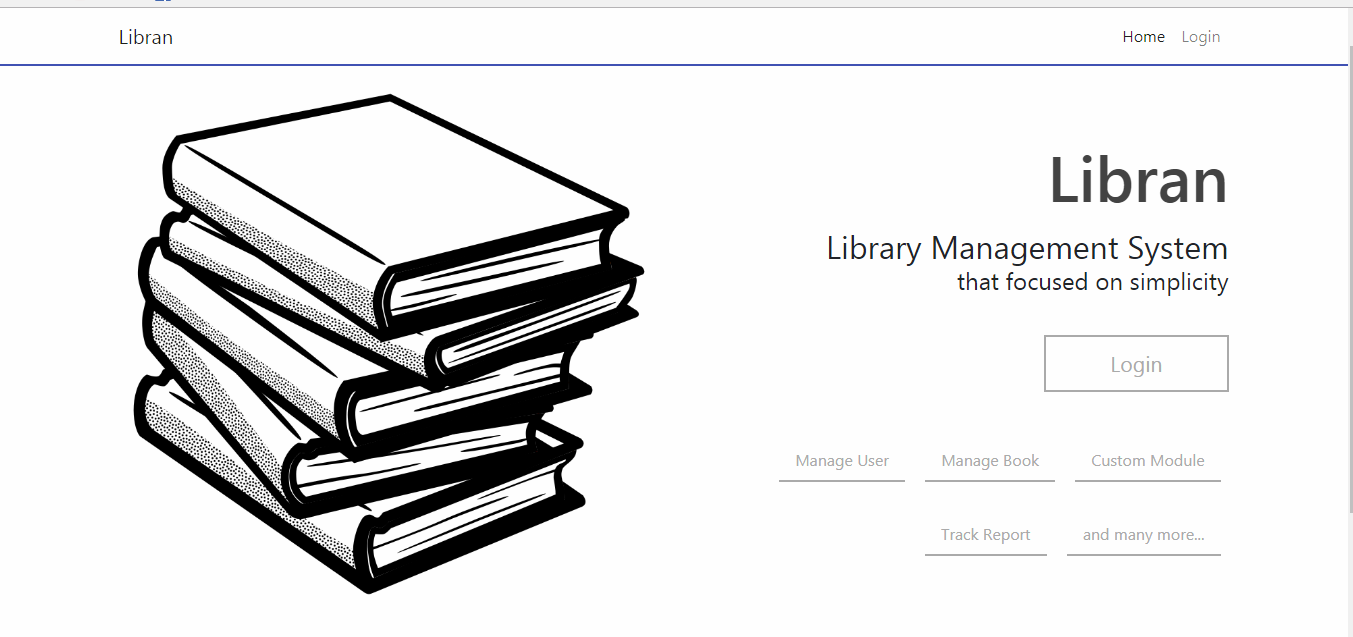
*Keterangan: Mencatat data peminjaman buku.*

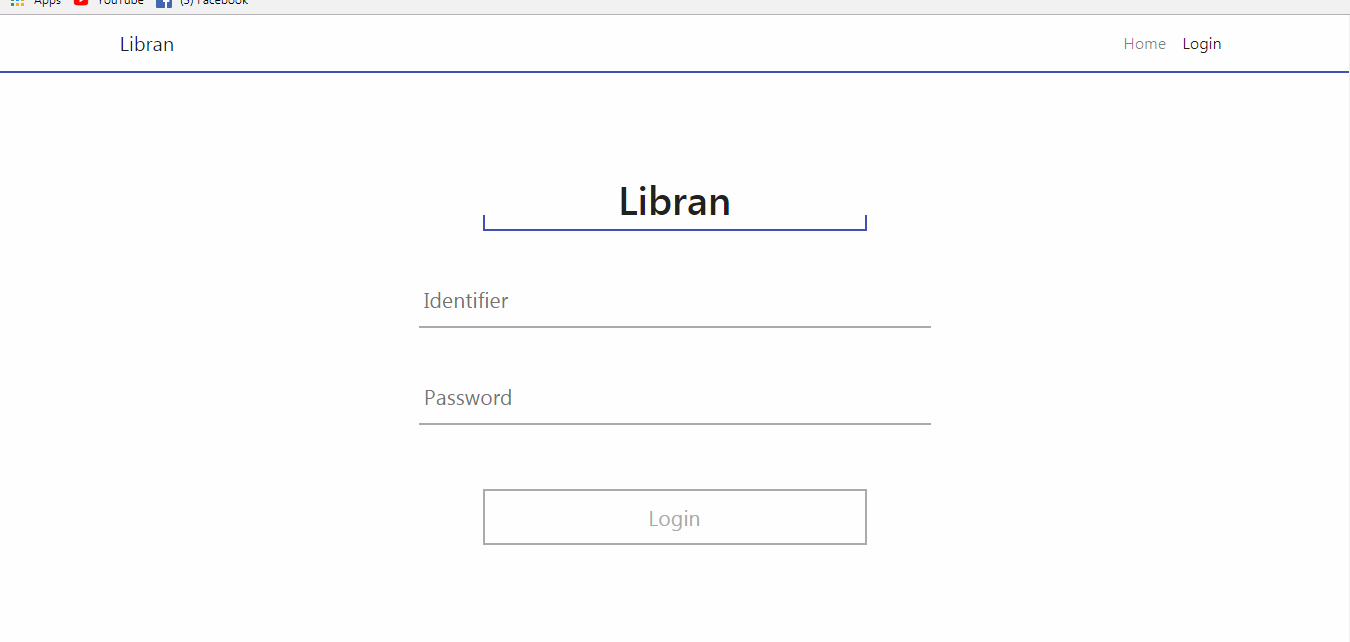
## Diagram Statechart

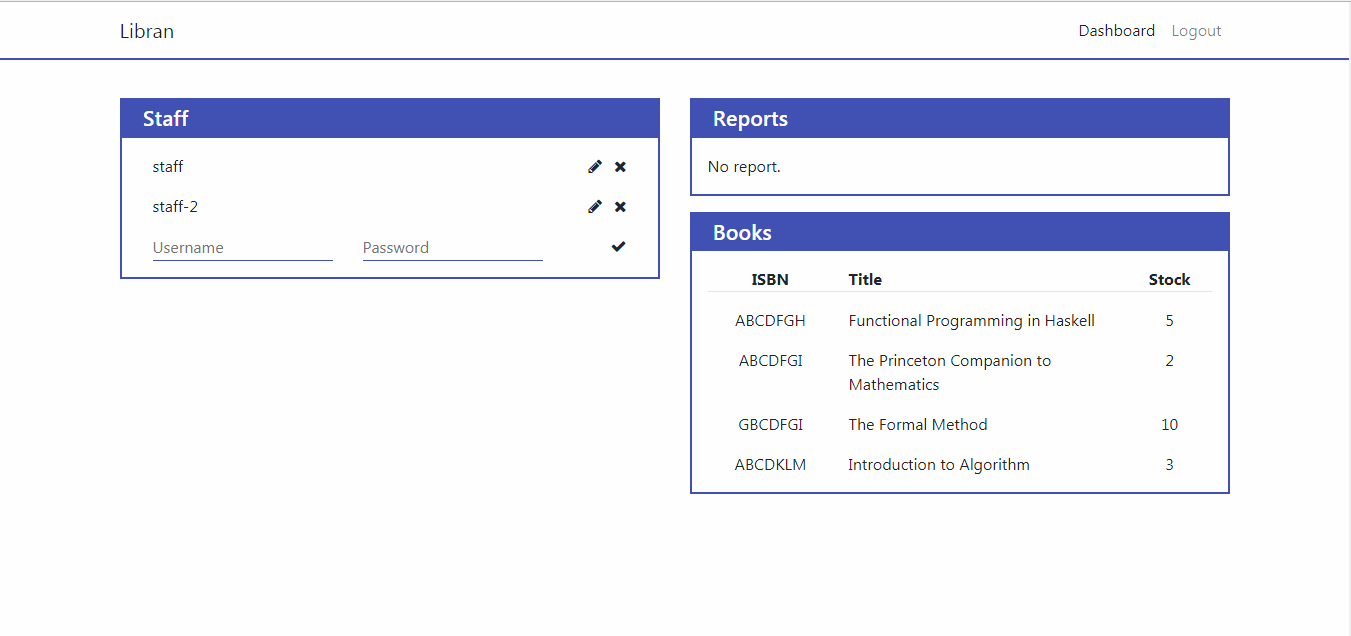
**Diagram D09 diagram *Statechart***

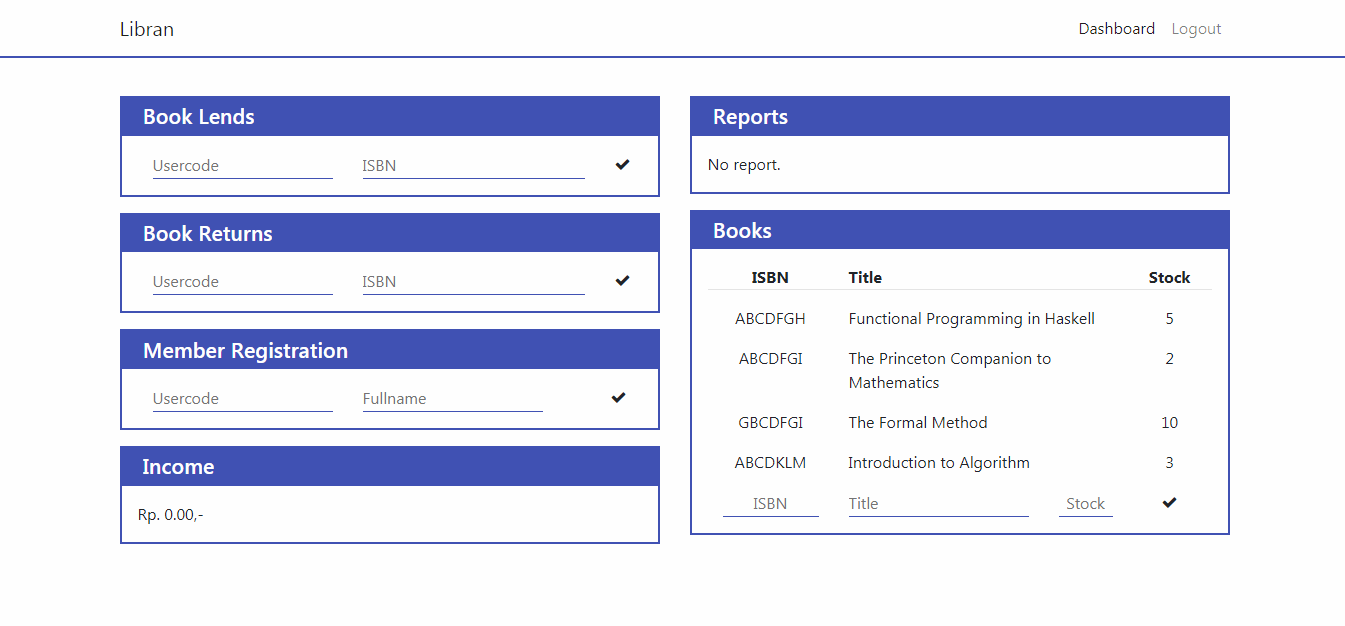


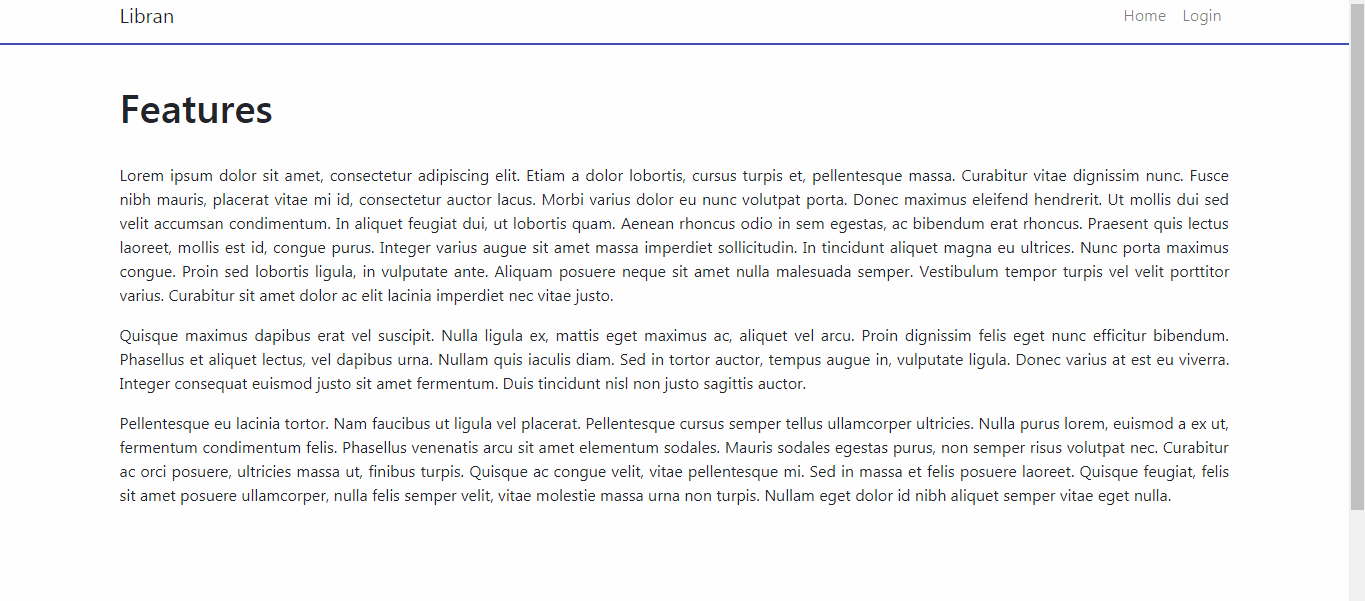
## Perancangan Antarmuka





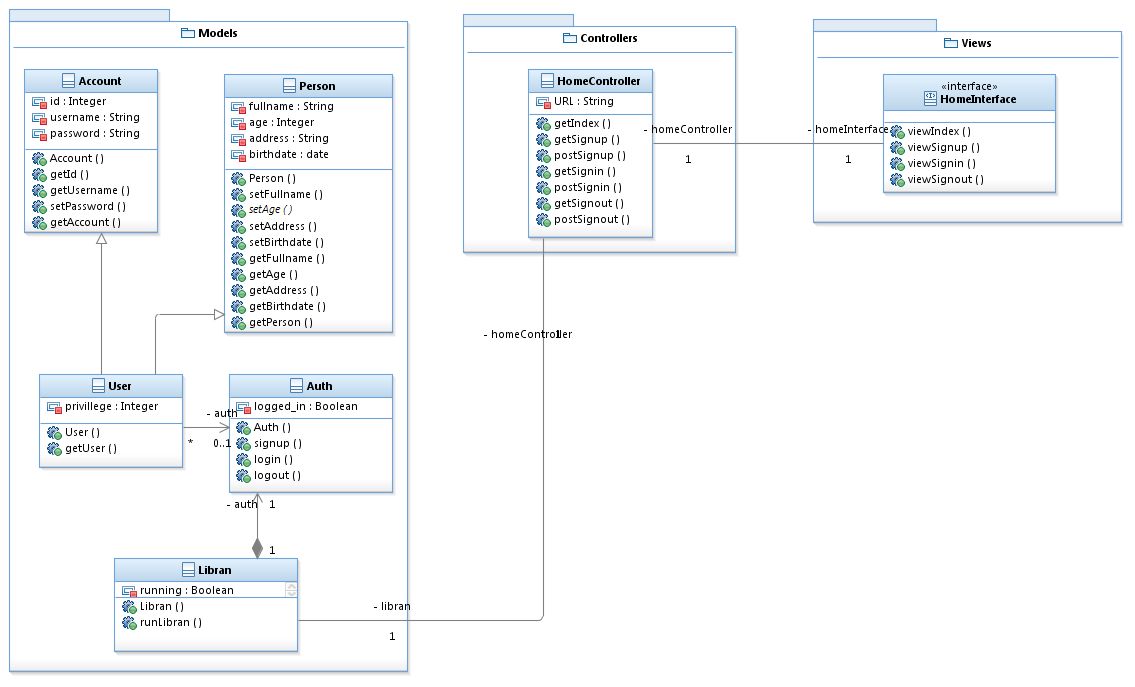


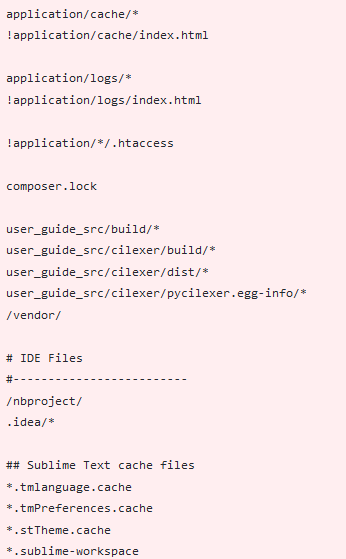


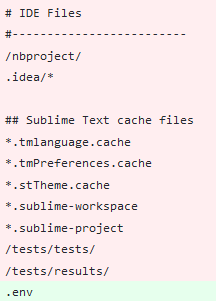


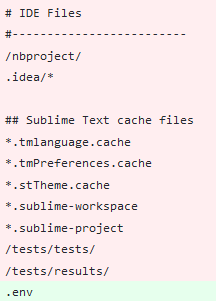
| **Id\_Objek** | **Jenis** | **Nama** | **Keterangan** |
| --- | --- | --- | --- |
|  |  | *Diisi dengan string yg tampil pd layer* | Diisi dengan penjelasan reaksi sistem, misalnya membuka layar apa, link kemana. Jika menyangkut suatu kode yang cukup rumit, acu algoritma yang telah diuraikan di atas. |
| *Button1* | Button | OK | Jika diklik, akan mengaktifkan Proses AlgoXXX. |
| *RTF1* | *RTF Box* |  | Isi Teks yang disimpan pada File xxx |

Jika objek dikaitkan ke File lain (misalnya file gambar, file teks), berikan nama file terkait dan deskripsi ringkas dalam kolom keterangan











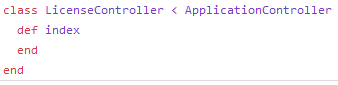
Pengontrol admin



Pengontrol Aplikasi



Pengontrol developer



Pengontrol lisensi



Pengontrol staff

## Perancangan Representasi Persistensi Kelas

*Bagian ini diisi dengan rancangan skema basisdata dan traceability-nya terhadap kelas entity.*

# Matriks Kerunutan

Mapping use case dengan kelas-kelas terkait

|  |  |
| --- | --- |
| **Kelas** | **Use Case Terkait** |
| Staff Manage | * Add book * Delete book * Update buku * View book |
| Book Manage | * Add book * Delete book * Update buku * View book |
| Return Manage | * Fine data * Report return manage |
| Report Manage | * Report borrowing manage * Report return manage * Report fine data * Show report manage |
|  |  |

# 