**DPPL**-xxxx

DESKRIPSI PERANCANGAN PERANGKAT LUNAK

TUBES BESAR

PEMBANGUNN SISTEM INFORMASI INVENTARIS UNIVERSITAS

untuk:

UNIVERSITAS

Dipersiapkan oleh:

1. Shinta Surya S.U (1301164410)
2. Komang Martha Suteja (1301164119)
3. Devy Anugrah.R (1301164604)
4. Fadillah Rizky.R (1301164493)

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* | | 8 |
| 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

# Pendahuluan

## Tujuan Penulisan Dokumen

Dokumen ini berisi Deskripsi Perancangan Perangkat Lunak (DPPL) untuk Universitas Telkom. Tujuan dari penulisan dokumen ini adalah untuk memberikan penjelasan yang detail mengenai perangkat lunak yang akan dibangun. Penggunaan dari dokumen ini adalah pengembangan sebuah Sistem Inventaris Universitas untuk memudahkan pengguna (user) untuk mengelola sistem inventaris yang ada di universitas. Dokumen ini akan digunakan sebagai bahan acuan untuk pengembangan perangkat lunak selanjutnya dan sebagai bahan evaluasi untuk mewujudkan sistem inventaris ini agar terus berkembang dengan baik.

## Lingkup Masalah

Sistem Inventaris Universitas ini adalah sebuah perangkat lunak (software) dimana software ini dapat menunjang serta mengelola system inventaris agar lebih baik. Dengan adanya system inventaris ini pencatatan data inventaris serta proses peminjaman lebih mudah. Sistem ini akan dijalnakan dengan basis web, hal yang menunjangnya tentunya adalah server agar sisstem dapat berjalan dengan baik. Adapun ruang lingkup masalah yang akan dibahas:

1. Peminjaman akan dilakukan jika peminjam sudah memiliki akun dan inventaris yang dipinjam tidak sedang dalam masa pinjam.
2. Inventaris universitas sepenuhnya dikelola oleh customer service
3. Admin bertugas sebagai pengelola data pegawai
4. Karyawan akan melakukan update kondisi barang .
5. Logistik akan menerima laporan inventaris.

## Definisi dan Istilah

1. Admin : orang yang membantu kinerja sistem untuk user.
2. SKPL : Spesifikasi Kebutuhan Perangkat Lunak, atau dalam bahasa inggris sering juga disebut sebagai software requirements specification (SRS) dan merupakan spesifikasi perngkat lunak yang akan dikembangkan.
3. Use Case : merupakan sebuah teknik yang digunakan dalam pengembangan sebuah software atau sistem informasi untuk menangkap kebutuhan fungsional dari sistem yang bersangkutan
4. Diagram komponen : Diagram komponen atau component diagram dibuat untuk menunjukkan organisasi dan ketergantungan diantara kumpulan komponen dalam sebuah sistem. Diagram komponen fokus pada komponen sistem yang dibutuhkan dan ada didalam sistem.

## Aturan Penamaan dan Penomoran

Penulisan dokumen DPPL 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 1 berikut ini :

|  |  |
| --- | --- |
| **Kebutuhan** | **Aturan Penomoran** |
| Kebutuhan Fungsionalitas | SKPL-A-Asset Universitas-XXX : Menunjukkan kebutuhan fungsional ke-XXX |
| Perancangan Detil Kelas | 3.2.X <Nama Kelas> : Menunjukkan nama kelas ke-X |
| Perancangan Antar Muka | 3.6.X <Nama Tampilan> : Menunjukkan nama kelas ke-X |

Table 1 Aturan Penamaan dan Penomoran

## Referensi

http://kosemisme.blogspot.co.id/2015/04/component-diagram.html

## Ikhtisar Dokumen

Dokumen ini berisikan tentang deskripsi segala rancangan yang akan digunakan bagi programmer untuk membangun system inventaris. Selain itu, dokumen ini mendeskripsikan tentang rancangan perangkat lunak ini secara arsitektural dan deskriptif yang secara rinci. Yang terdiri dari :

1. BAB 1 : Pendahuluan

Pendahuluan berisi tentang tujuan penulisan dokumen, lingkup masalah yang menjadi Batasan dalam suatu pembuatan perangkat lunak (software), serta deskripsi umum dokumen yang berisi tentang rincian dari struktur dokumen

1. BAB 2 : Deskripsi Rancangan Global

Deskripsi ini berisi tengan hasil analisis dari kebutuhan perangkat lunak (software) yang akan diimplementasikan di lingkungan pengguna secara global.

1. BAB 4 : Matriks Kerunutan

Matriks Kerunutan adalah hasil mapping kelas terhadap use case.

# Deskripsi Perancangan Global

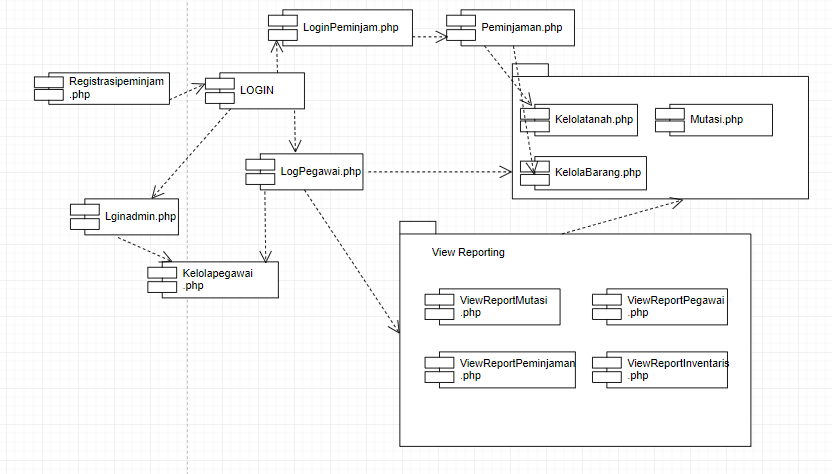
## 2.1 Rancangan Lingkungan Implementasi

Berikut daftar rancangan implementasi yang akan digunakan pada system inventaris universitas :

|  |  |
| --- | --- |
| ***Sistem Rancangan*** | ***Spesifikasi*** |
| *Sistem Operasi User* | *Windows* |
| *DBMS* | *MySql* |
| *Development Tools* | *Sublime* |
| *Bahasa Pemrograman* | *css, php, html* |

Table 2 Rancangan Lingkungan Implementasi

## 2.2 Deskripsi Arsitektural

Arsitektur dari system akan digambarkan dengan diagram komponen berikut : 

## 2.3 Deskripsi Komponen

| **No** | **Nama Komponen** | **Keterangan** |
| --- | --- | --- |
| 1. | LogPegawai.php | Prosedur untuk menjalankan komponen yang akan dilakukan pegawai (cs,karyawan,logistic) |
| 2. | LogAdmin.php | Prosedur untuk menjalankan komponen yang akan dilakukan admin |
| 3. | LogPeminjam.php | Prosedur untuk menjalankan komponen yang akan dilakukan peminjam |
| 4. | ViewReportMutasi.php | Prosedur dimana Pegawai yang login akan melihat laporan data mutasi |
| 5. | ViewReportInventaris.php | Prosedur dimana Pegawai yang login akan melihat laporan data inventaris |
| 6. | ViewReportPeminjaman.php | Prosedur dimana Pegawai yang login akan melihat laporan data peminjaman |
| 7. | ViewReportPegawai.php | Prosedur dimana Pegawai yang login akan melihat laporan data pegawai |
| 8 | Registrasipeminjam.php | Prosedur yang harus dijalankan peminjam sebelum login |
| 9. | Peminjaman.php | Prosedur yang dijalankan peminjam untuk melakukan peminjaman inventaris |
| 10. | KelolaPegawai.php | Prosedur berisi CRUD untuk mengelola data pegawai |
| 11. | KelolaBarang.php | Prosedur berisi CRUD untuk mengelola data barang |
| 12. | KelolaTanah.php | Prosedur berisi CRUD untuk mengelola data tanah |
| 13. | KelolaMutasi.php | Prosedur berisi CRUD untuk mengelola data mutasi |

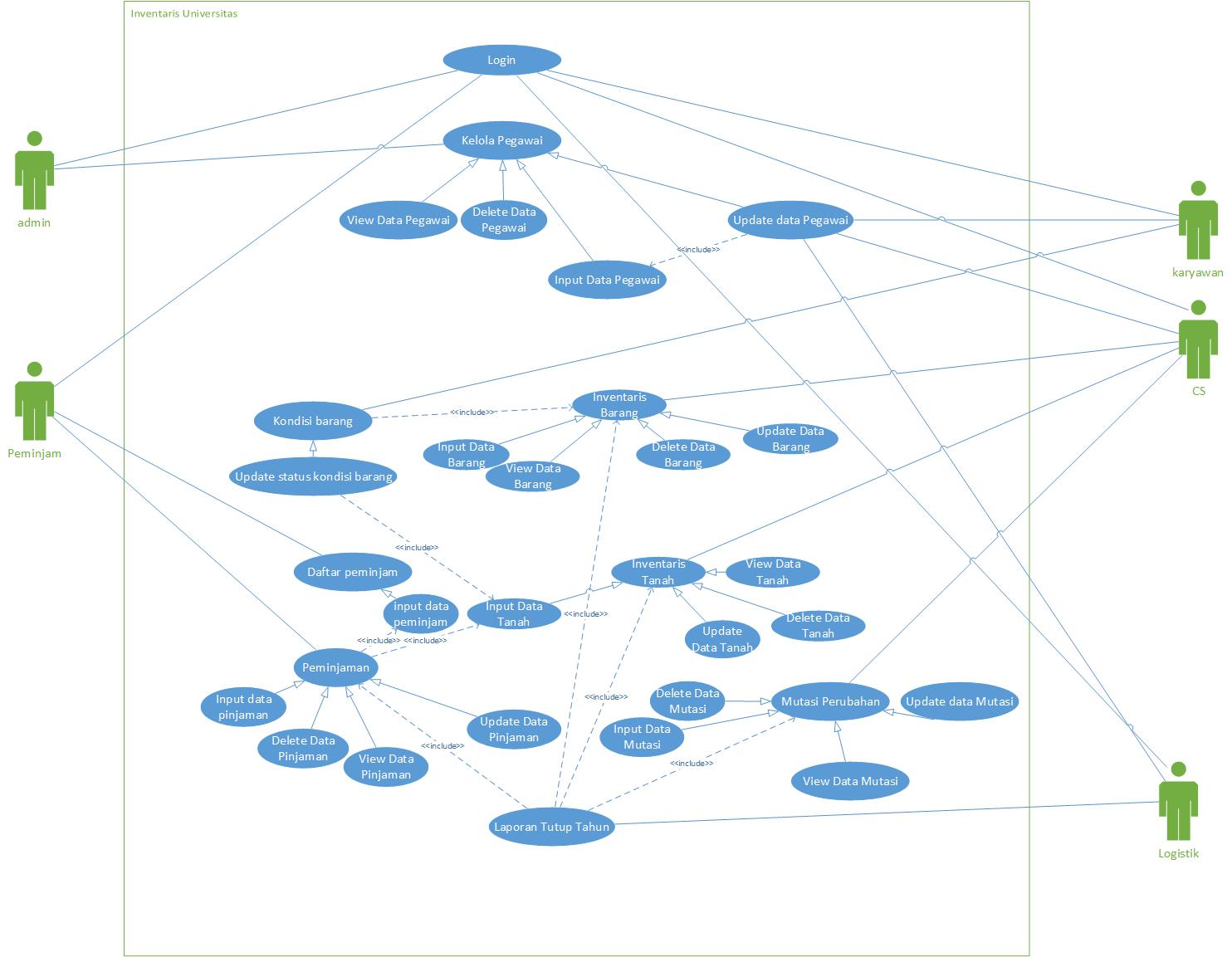
Table 3 Deskripsi Komponen

# Perancangan Rinci

## Realisasi Use Case

### Use Case <nama use case 1>

Jika use case ini akan direalisasikan dalam bentuk aplikasi berbasis web, maka subbab yang terkait dengan perancangan elemen aplikasi berbasis web harus diisi.



#### Use Case Scenario

*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:*

1. Nama Use Case : Input Data Pegawai

No Use Case : 2.1

Aktor : Admin

Pre-Condition : Admin belum login dan data pegawai belum diinputkan

Post-Condition : Admin telah login dan data pegawai tersimpan dalam database Pegawai

Deskripsi : Input pegawai digunakan untuk menginput data pegawai yang nantinya akan di simpan di database pegawai.

|  |  |
| --- | --- |
| Admin | Sistem |
| 1. Login |  |
| 1. Masuk menu kelola pegawai | 1. Tampilkan menu kelola pegawai |
| 1. Pilih menu input data pegawai |  |
|  | 1. Tampilkan form data pegawai |
| 1. Inputkan data pegawai |  |
|  | 1. Simpan data pegawai pada database pegawai |

1. Nama Use Case : Hapus Data Pegawai

No Use Case : 2.2

Aktor : Admin

Pre-Condition : Admin belum login dan data pegawai belum terhapus

Post-Condition : Admin telah login dan data pegawai terhapus

Deskripsi :Delete pegawai di gunakan untuk menghapus data pegawai yang diakses dari database pegawai dan pada kondisi akhir data akan disimpan di database pegawai.

|  |  |
| --- | --- |
| Admin | Sistem |
| 1. Login |  |
| 1. Masuk menu kelola pegawai | 1. Tampilkan menu kelola pegawai |
| 1. Pilih menu hapus data pegawai |  |
|  | 1. Tampilkan data pegawai |
| 1. Hapus data pegawai |  |
|  | 1. Data pegawai terhapus dari database pegawai |

1. Nama Use Case : View Data Pegawai

No Use Case : 2.3

Aktor : Admin,CS,Karyawan,Logistik

Pre-Condition : Admin belum login dan data pegawai belum ditampilkan

Post-Condition : Aktor telah login dan data pegawai ditampilkan

Deskripsi : View pegawai digunkan untuk menampilkan data pegawai yang diakses dari database pegawai, yang nantinya akan ditampilkan pada sisi actor

|  |  |
| --- | --- |
| Admin | Sistem |
| 1. Login |  |
| 1. Masuk menu kelola pegawai | 1. Tampilkan menu kelola pegawai |
| 1. Pilih menu view data pegawai |  |
|  | 1. Ambil data data pegawai untuk ditampilkan |
|  | 1. Data pegawai ditampilkan |
| 1. Login |  |

1. Nama Use Case : Update Data Pegawai

No Use Case : 2.4

Aktor : Admin, CS,Karyawan, Logistik

Pre-Condition : Aktor belum login dan data pegawai belum diupdate

Post-Condition : Aktor telah login dan data pegawai berhasil diupdate

Deskripsi : Update pegawai dilakukan oleh cs, karyawan, logistik (pegawai). Pegawai melakukan update username password dan akan disimpan kembali pada database pegawai.

|  |  |
| --- | --- |
| Admin | Sistem |
| 1. Login |  |
| 1. Masuk menu kelola pegawai | 1. Tampilkan menu kelola pegawai |
| 1. Pilih menu update data pegawai |  |
|  | 1. Tampilkan form data pegawai |
| 1. Update data pegawai |  |
| 1. Login |  |

1. Nama Use case : Input Data Barang

No Use Case : 3.1

Aktor : CS

Pre-Condition : CS belum login dan data barang belum diinputkan

Post-Condition : CS telah login dan data barang telah diinputkan

Deskripsi : Insert data barang, fungsi ini berguna untuk memasukan data barang yang nantinya disimpan ke dalam DataBase Barang.

|  |  |
| --- | --- |
| CS | Sistem |
| 1. Login |  |
| 1. Masuk menu kelola barang | 1. Tampilkan menu kelola barang |
| 1. Pilih menu input data barang |  |
|  | 1. Tampilkan form data barang |
| 1. Input data barang |  |
|  | 1. Data barang tersimpan pada database barang |

1. Nama Use case : Delete Data Barang

No Use Case : 3.2

Aktor : CS

Pre-Condition : CS belum login dan data barang belum terhapus

Post-Condition : CS telah login dan data barang telah terhapus

Deskripsi : Delete data barang, fungsi ini berguna untuk menghapus data barang yang tidak diperlukan dari data base barang

|  |  |
| --- | --- |
| CS | Sistem |
| 1. Login |  |
| 1. Masuk menu kelola barang | 1. Tampilkan menu kelola barang |
| 1. Pilih menu delete data barang |  |
|  | 1. Tampilkan data barang |
| 1. Hapus data barang |  |
|  | 1. Data barang terhapus |

1. Nama Use case : View Data Barang

No Use Case : 3.3

Aktor : CS

Pre-Condition : CS belum login dan data barang belum ditampilkan

Post-Condition : CS telah login dan Data barang ditampilkan

Deskripsi : View data barang digunakan untuk menampilkan data barang yang diakses dari database barang yang nantinya akan ditampilkan pada sisi actor.

|  |  |
| --- | --- |
| CS | Sistem |
| 1. Login |  |
| 1. Masuk menu kelola barang | 1. Tampilkan menu kelola barang |
| 1. Pilih menu view data barang |  |
|  | 1. Ambil data data barang untuk ditampilkan |
|  | 1. Data barang ditampilkan |
| 1. Login |  |

1. Nama Use case : Update Data Barang

No Use Case : 3.4

Aktor : CS

Pre-Condition : CS belum login dan data barang belum diupdate

Post-Condition : Data barang berhasil diupdate

Deskripsi : Update data barang, fungsi ini berguna untuk memperbarui data inventaris barang

|  |  |
| --- | --- |
| Admin | Sistem |
| 1. Login |  |
| 1. Masuk menu kelola barang | 1. Tampilkan menu kelola barang |
| 1. Pilih menu update data barang |  |
|  | 1. Tampilkan form data barang |
| 1. Update data barang |  |
|  | 1. Data pegawai terupdate dan tersimpan pada database barang |

1. Nama Use case : Input Data Tanah

No Use Case : 4.1

Aktor : CS

Pre-Condition : CS belum login dan data tanah belum diinputkan

Post-Condition : CS telah login dan data tanah telah diinputkan

Deskripsi : Insert data tanah, fungsi ini berguna untuk memasukkan data tanah yang nantinya disimpan ke dalam DataBase Tanah.

|  |  |
| --- | --- |
| Admin | Sistem |
| 1. Login |  |
| 1. Masuk menu kelola tanah | 1. Tampilkan menu kelola tanah |
| 1. Pilih menu input data tanah |  |
|  | 1. Tampilkan form data tanah |
| 1. Input data tanah |  |
|  | 1. Data tanah tersimpan pada database tanah |

1. Nama Use case : Delete Data Tanah

No Use Case : 4.2

Aktor : CS

Pre-Condition : CS belum login dan data tanah belum terhapus

Post-Condition : CS telah login dan data tanah telah terhapus

Deskripsi : Delete data tanah, fungsi ini berguna untuk menghapus data tanah yang tidak diperlukan dari data base tanah

|  |  |
| --- | --- |
| CS | Sistem |
| 1. Login |  |
| 1. Masuk menu kelola tanah | 1. Tampilkan menu kelola tanah |
| 1. Pilih menu delete data tanah |  |
|  | 1. Tampilkan data tanah |
| 1. Hapus data tanah |  |
|  | 1. Data tanah terhapus |

1. Nama Use case : View Data Tanah

No Use Case : 4.3

Aktor : CS

Pre-Condition : CS belum login dan data tanah belum ditampilkan

Post-Condition : CS telah login dan Data tanah ditampilkan

Deskripsi : View data tanah digunakan untuk menampilkan data tanah yang diakses dari database tanah yang nantinya akan ditampilkan pada sisi aktor.

|  |  |
| --- | --- |
| CS | Sistem |
| 1. Login |  |
| 1. Masuk menu kelola tanah | 1. Tampilkan menu kelola tanah |
| 1. Pilih menu view data tanah |  |
|  | 1. Ambil data data tanah untuk ditampilkan |
|  | 1. Data tanah ditampilkan |

1. Nama Use case : Update Data Tanah

No Use Case : 4.4

Aktor : CS

Pre-Condition : CS belum login dan data tanah belum diupdate

Post-Condition : Data tanah berhasil diupdate

Deskripsi : Update data tanah, fungsi ini berguna untuk memperbarui data inventaris tanah

|  |  |
| --- | --- |
| CS | Sistem |
| 1. Login |  |
| 1. Masuk menu kelola tanah | 1. Tampilkan menu kelola tanah |
| 1. Pilih menu update data tanah |  |
|  | 1. Tampilkan form data tanah |
| 1. Update data tanah |  |
|  | 1. Data pegawai terupdate dan tersimpan pada database tanah |

1. Nama Use case : Input Daftar peminjam

No use case : 5.1

Aktor : Peminjam

Pre-condition : Peminjam belum mendaftar

Post-condition : Peminjam telah terdaftar

Description : Input daftar peminjam ini digunakan sebagai syarat sebelum peminjam melakukan peminjaman, maka harus mendaftarkan dirinya terlebih dahulu. Yang selanjutnya dapat digunkan untuk login dan meminjam inventaris universitas

|  |  |
| --- | --- |
| Peminjam | Sistem |
| * + - 1. Masuk menu daftar peminjam | * + - 1. Tampilkan form daftar peminjam |
| * + - 1. Inputkan data peminjam | * + - 1. Data peminjam tersimpan dalam database peminjam |

1. Nama use case:Input data peminjaman

No Use case : 6.1

Aktor : Peminjam

Pre-Condition : Data pinjaman belum diinputkan

Post-Condition : Data pinjaman tersimpan pada database pinjaman

Description : Insert data barang, fungsi ini berguna untuk memasukan data pinjaman yang nantinya disimpan ke dalam DataBase Pinjaman.

|  |  |
| --- | --- |
| Peminjam | Sistem |
| 1. Login |  |
| 1. Masuk menu Peminjaman | 1. Tampilkan menu peminjaman |
| 1. Pilih menu input data pinjaman |  |
|  | 1. Tampilkan form data pinjaman |
| 1. Input data pinjaman |  |
|  | 1. Data peminjaman tersimpan pada databse pinjaman |

1. Nama use case : Delete data peminjaman

No Use case : 6.2

Aktor : Peminjam

Pre-Condition : Data pinjaman belum terhapus

Post-Condition : Data pinjaman berhasil dihapus

Description : Delete data pinjaman, fungsi ini berguna untuk menghapus data pinjaman yang tidak diperlukan dari database pinjaman

|  |  |
| --- | --- |
| Peminjam | Sistem |
| 1. Login |  |
| 1. Masuk menu peminjaman | 1. Tampilkan menu peminjaman |
| 1. Pilih menu delete data pinjaman |  |
|  | 1. Tampilkan data pinjaman |
| 1. Hapus data pinjaman |  |
|  | 1. Data pinjaman terhapus |

1. Nama use case : View data peminjaman

No Use case : 6.3

Aktor : Peminjam

Pre-Condition : Data pinjaman belum ditampilkan

Post-Condition : Data pinjaman ditampilkan

Description : View data pinjaman digunakan untuk menampilkan data pinjman yang diakses dari database pinjaman yang nantinya akan ditampilkan pada sisi aktor.

|  |  |
| --- | --- |
| Peminjam | Sistem |
| 1. Login |  |
| 1. Masuk menu peminjaman | 1. Tampilkan menu peminjaman |
| 1. Pilih menu view data peminjamn |  |
|  | 1. Ambil data data pinjaman untuk ditampilkan |
|  | 1. Data pinjaman ditampilkan |

1. Nama use case: Update data peminjaman

No Use case : 6.4

Aktor : Peminjam

Pre-Condition : Data pinjaman belum diupdate

Post-Condition : Data pinjaman berhasil diupdate

Description : Update data pinjaman, fungsi ini berguna untuk memperbarui data peminjaman.

|  |  |
| --- | --- |
| Peminjam | Sistem |
| 1. Login |  |
| 1. Masuk menu peminjaman | 1. Tampilkan menu peminjaman |
| 1. Pilih menu update data peminjaman |  |
|  | 1. Tampilkan form data pinjman |
| 1. Update data pinjaman |  |

1. Nama use case: Edit Kondisi Inventaris

No Use case : 7.1

Aktor : Karyawan

Pre-Condition : Karyawan belum login dan status barang belum berubah

Post-Condition : Karyawan telah login dan status kondisi barang berubah

Description : Update kondisi barang, fungsi ini berguna untuk memperbarui data kondisi barang.

|  |  |
| --- | --- |
| Karyawan | Sistem |
| 1. Login |  |
| 1. Masuk menu kondisi barang | 1. Tampilkan menu kondisi barang |
| 1. Pilih menu update kondisi inventaris |  |
|  | 1. Tampilkan form data barang atau tanah |
| 1. Update kondisi inventaris |  |
|  | 1. Data kondisi terupdate dan tersimpan pada database sesuai yang diupdate. |

1. Nama use case:Input data mutasi

No Use case : 8.1

Aktor : CS

Pre-Condition : CS belum login dan data mutasi belum diinputkan

Post-Condition : CS telah login dan data mutasi diinputkan

Description : Insert data mutasi, fungsi ini berguna untuk memasukan data mutasi inventaris yang

nantinya disimpan ke dalam DataBase Mutasi

|  |  |
| --- | --- |
| CS | Sistem |
| 1. Login |  |
| 1. Masuk menu mutasi | 1. Tampilkan menu mutasi |
| 1. Pilih menu input data mutasi |  |
|  | 1. Ambil data inventaris yang akan dimutasi dengan mengakses database barang atau tanah dan database fakultas atau universitas |
|  | 1. Tampilkan form data mutasi |
| 1. Input data mutasi | 1. Data mutasi tersimpan pada databse mutasi |

1. Nama use case:Delete data mutasi

No Use case : 8.2

Aktor : CS

Pre-Condition : CS belum login dan data mutasi belum terhapus

Post-Condition : CS telah login dan data mutasi berhasil dihapus

Description : Delete data mutasi, fungsi ini berguna untuk menghapus data mutasi yang tidak diperlukan dari database mutasi

|  |  |
| --- | --- |
| CS | Sistem |
| 1. Login |  |
| 1. Masuk menu mutasi | 1. Tampilkan menu mutasi |
| 1. Pilih menu haspus data mutasi |  |
|  | 1. Tampilkan form data mutasi |
| 1. Hapus data mutasi |  |
|  | 1. Data mutasi terhapus dari database mutasi |

1. Nama use case:View data mutasi

No Use case : 8.3

Aktor : CS

Pre-Condition : CS belum login dan data mutasi belum ditampilkan

Post-Condition : CS telah login dan data mutasi ditampilkan

Description : View data mutasi digunakan untuk menampilkan data mutasi yang diakses dari database mutasi yang nantinya akan ditampilkan pada sisi aktor.

|  |  |
| --- | --- |
| CS | Sistem |
| 1. Login |  |
| 1. Masuk menu mutasi | 1. Tampilkan menu mutasi |
| 1. Pilih menu view data mutasi |  |
|  | 1. Ambil data mutasi untuk ditampilkan |
|  | 1. Data mutasi ditampilkan |

1. Nama use case: Edit data mutasi

No Use case : 8.4

Aktor : CS

Pre-Condition : CS belum login dan data mutasi belum berubah

Post-Condition : CS telah login dan data mutasi berubah

Description : Edit data mutasi, fungsi ini berguna untuk memperbarui data mutasi.

|  |  |
| --- | --- |
| Admin | Sistem |
| 1. Login |  |
| 1. Masuk menu mutasi | 1. Tampilkan menu mutasi |
| 1. Pilih menu edit data mutasi |  |
|  | 1. Ambil data inventaris yang akan dimutasi dengan mengakses database barang atau tanah dan database fakultas atau universitas |
|  | 1. Tampilkan form data mutasi |
| 1. Edit data mutasi | 1. Data mutasi tersimpan pada databse mutasi |

1. Nama use case : Laporan mutasi tutup tahun

No Use case : 9.1

Aktor : Logistik

Pre Condition : Logistik belum login dan data report mutasi belum terambil

Post-Condition : Logistik telah login dan data report mutasi terambil

Description : Laporan mutasi tutup tahun digunakan untuk mendapat report dari sistem mutasi

|  |  |
| --- | --- |
| Logistik | Sistem |
| * + - 1. Login |  |
| * + - 1. Masuk menu tutup tahun | * + - 1. Tampilkan menu tutup tahun |
| * + - 1. Pilih menu laporan mutasi | * + - 1. Ambil data mutasi |
|  | 1. Report data mutasi |

1. Nama use case : Laporan barang tutup tahun

No Use case : 9.2

Aktor : Logistik

Pre Condition : Logistik belum login dan data report barang belum terambil

Post-Condition : Logistik telah login dan data report barang terambil

Description : Laporan barang tutup tahun digunakan untuk mendapat report dari sisteminventaris barang

|  |  |
| --- | --- |
| Logistik | Sistem |
| * + - 1. Login |  |
| * + - 1. Masuk menu tutup tahun | * + - 1. Tampilkan menu tutup tahun |
| * + - 1. Pilih menu laporan inventaris barang | * + - 1. Ambil data barang |
|  | 1. Report data barang |

1. Nama Use case : Laporan tanah tutup tahun

No Use case : 9.3

Aktor : Logistik

Pre Condition : Logistik belum login dan data report tanah belum terambil

Post-Condition : Logistik telah login dan data report tanah terambil

Description : Laporan tanah tutup tahun digunakan untuk mendapat report dari sistem inventaris tanah

|  |  |
| --- | --- |
| Logistik | Sistem |
| * + - 1. Login |  |
| * + - 1. Masuk menu tutup tahun | * + - 1. Tampilkan menu tutup tahun |
| * + - 1. Pilih menu laporan inventaris tanah | * + - 1. Ambil data tanah |
|  | 1. Report data tanah |

1. Nama use case : Laporan peminjaman tutup tahun

No Use case : 9.4

Aktor : Logistik

Pre Condition : Logistik belum login dan data report peminjaman belum terambil

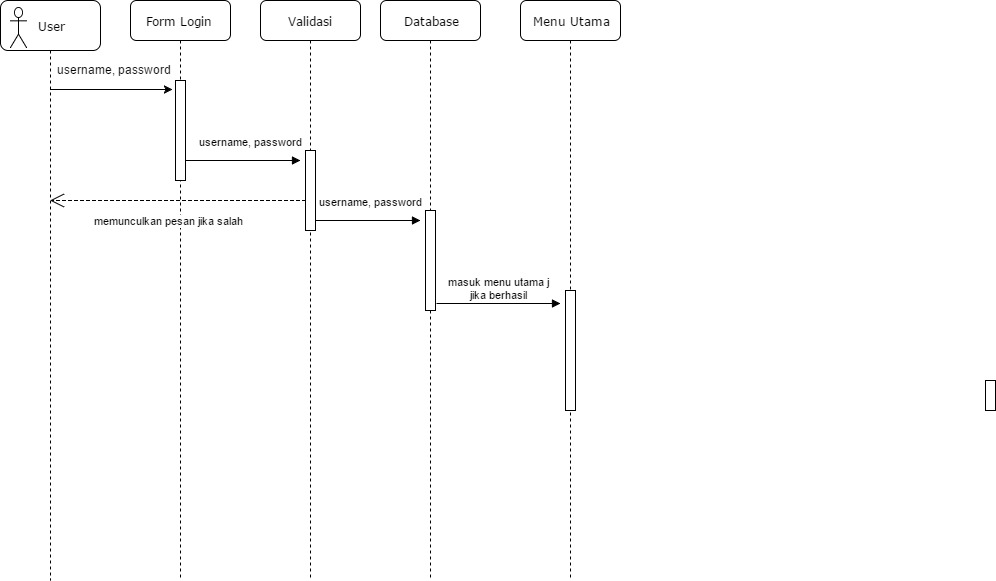
Post-Condition : Logistik telah login dan data report peminjaman terambil

Description : Laporan peminjaman tutup tahun digunakan untuk mendapat report dari sistem peminjaman

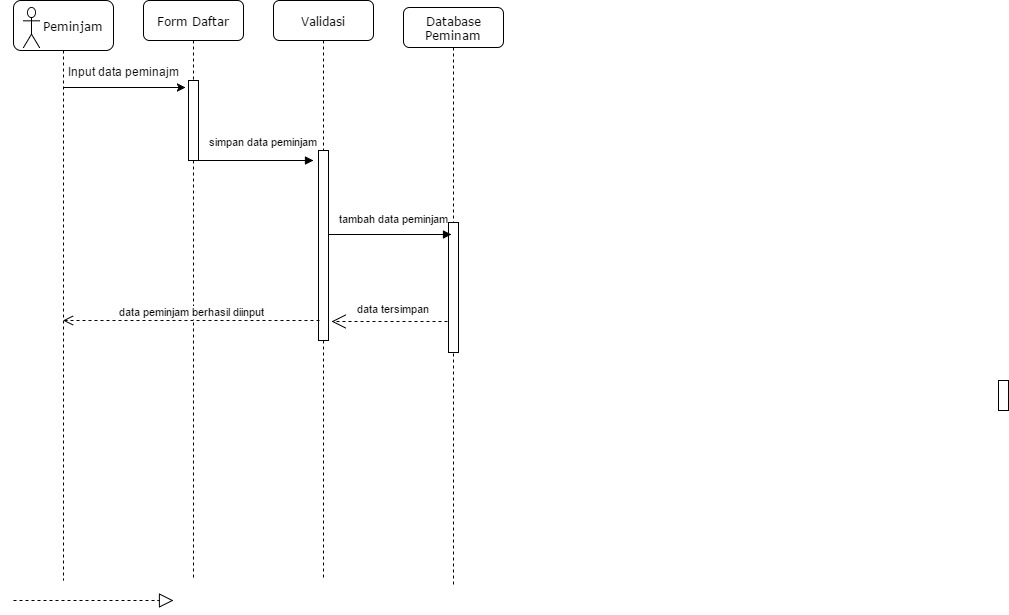
|  |  |
| --- | --- |
| Logistik | Sistem |
| * + - 1. Login |  |
| * + - 1. Masuk menu tutup tahun | * + - 1. Tampilkan menu tutup tahun |
| * + - 1. Pilih menu laporan peminjaman | * + - 1. Ambil data peminjaman |
|  | * + - 1. Report data peminjaman |

#### Sequence Diagram

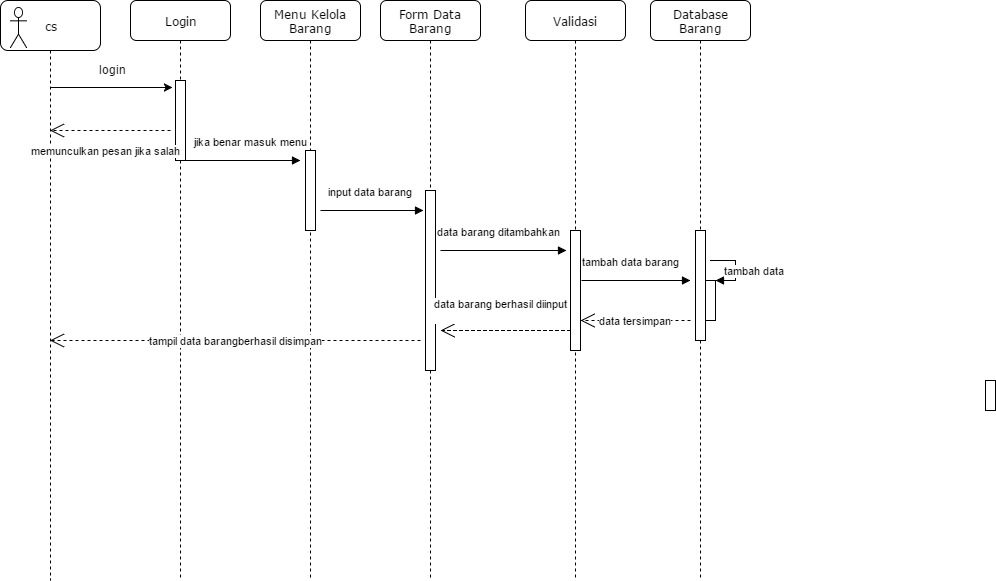
##### Login



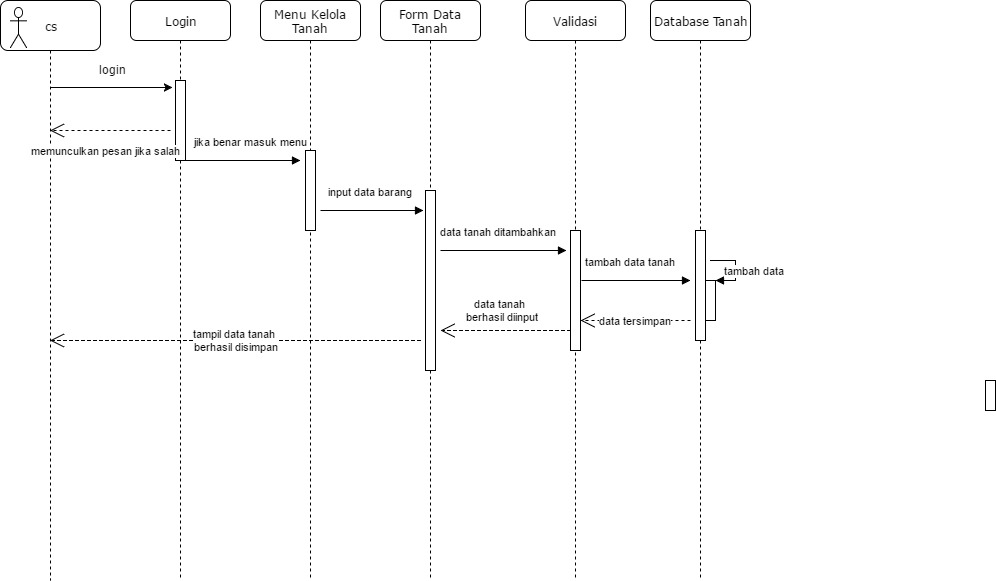
##### Daftar Peminjam



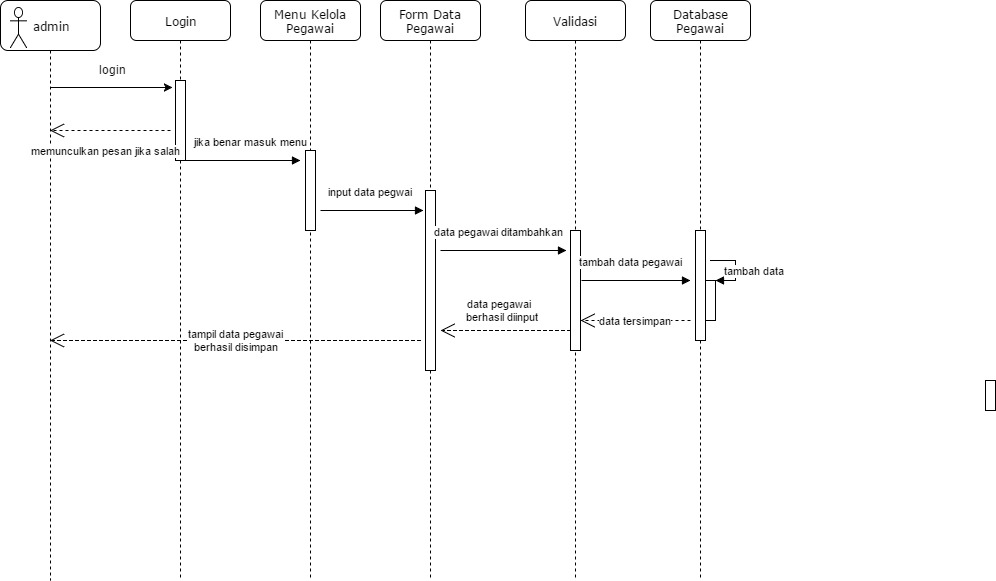
##### Input Kelola Barang



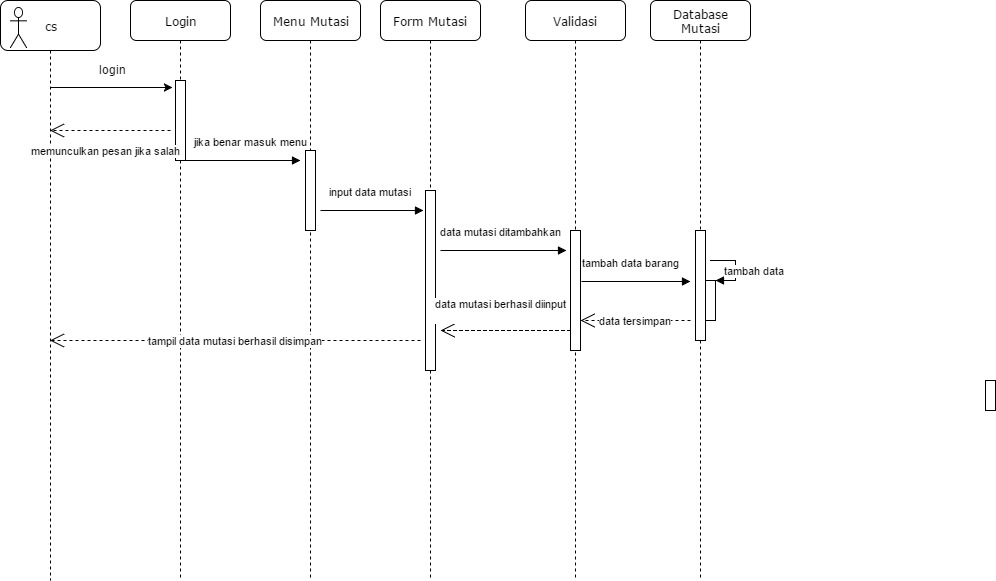
##### Input Kelola Tanah



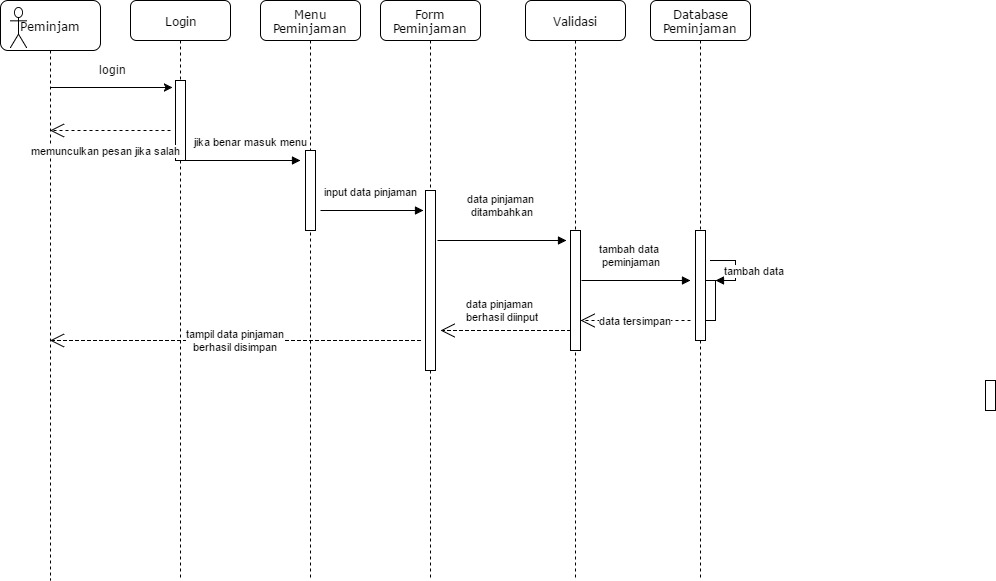
##### Input Kelola Pegawai



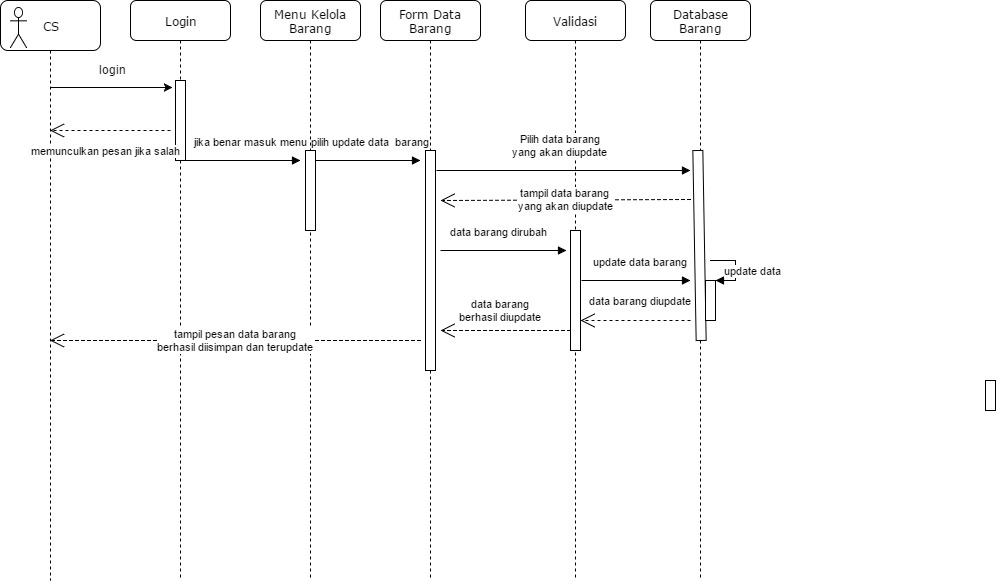
##### Input Mutasi



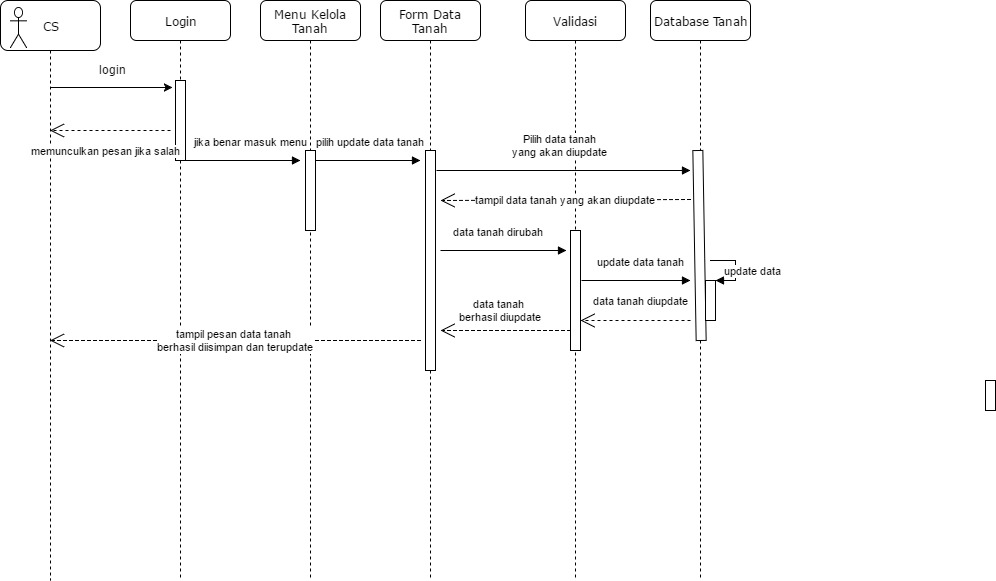
##### Input Pinjaman



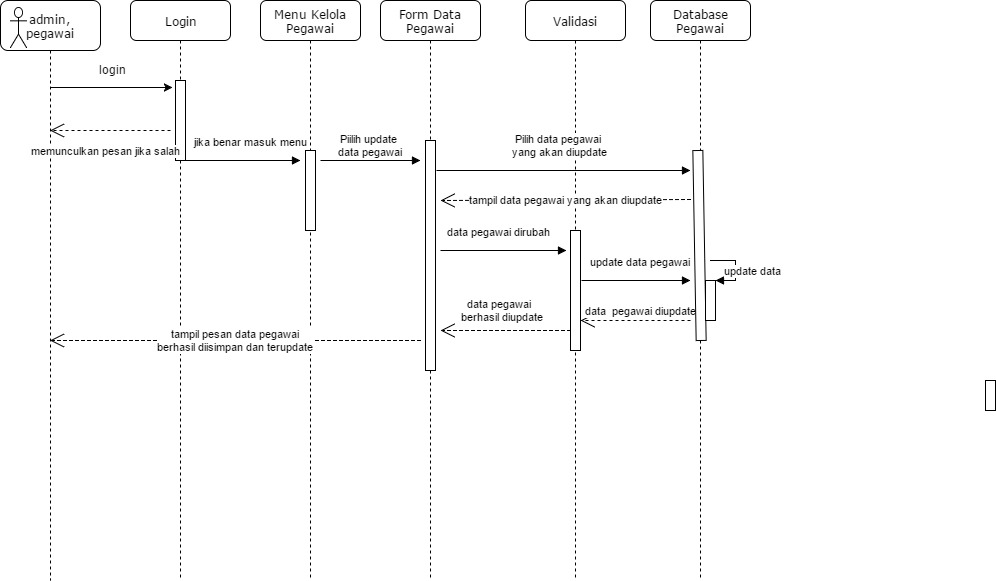
##### Update Kelola Barang



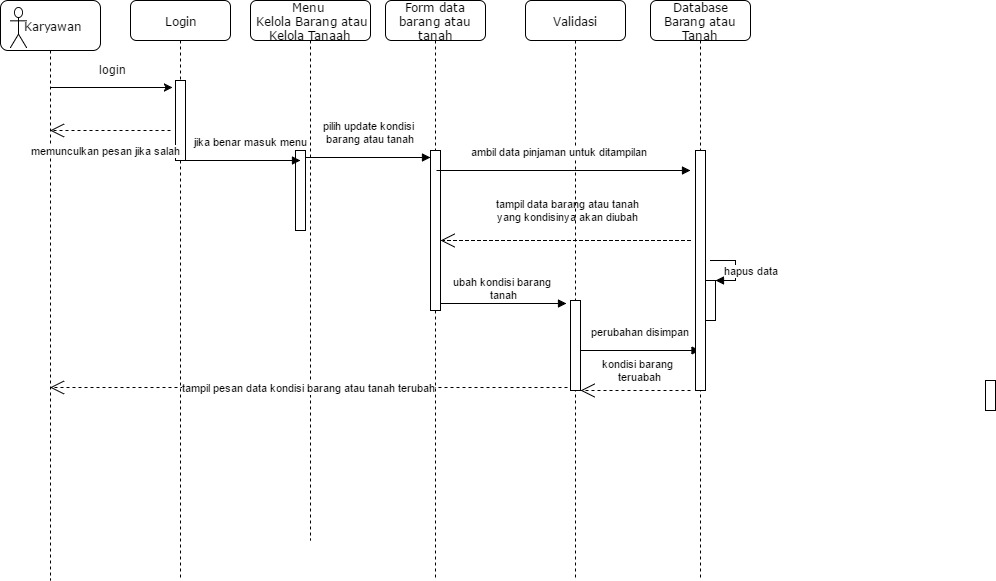
##### Update Kelola Tanah



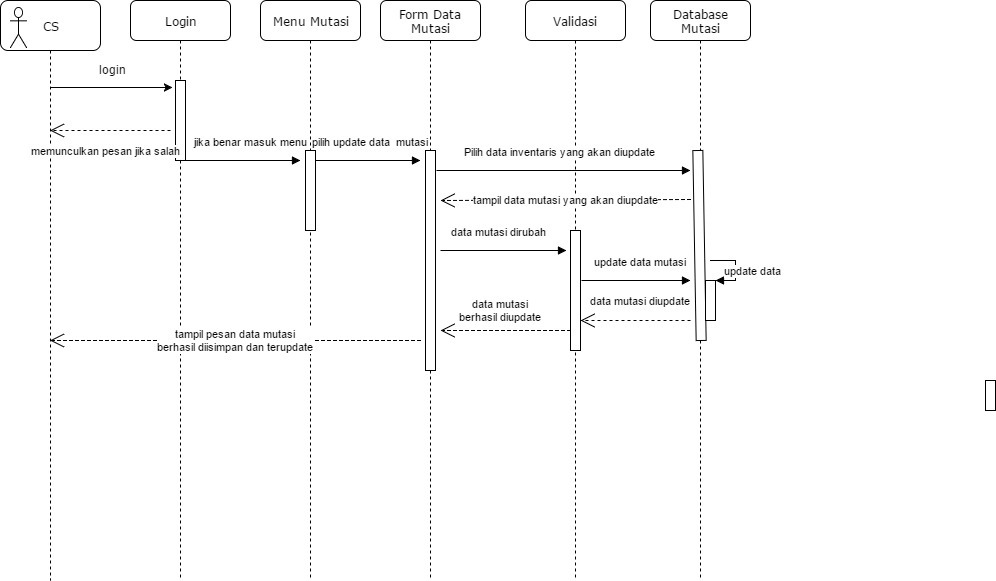
##### Update Kelola Pegawai



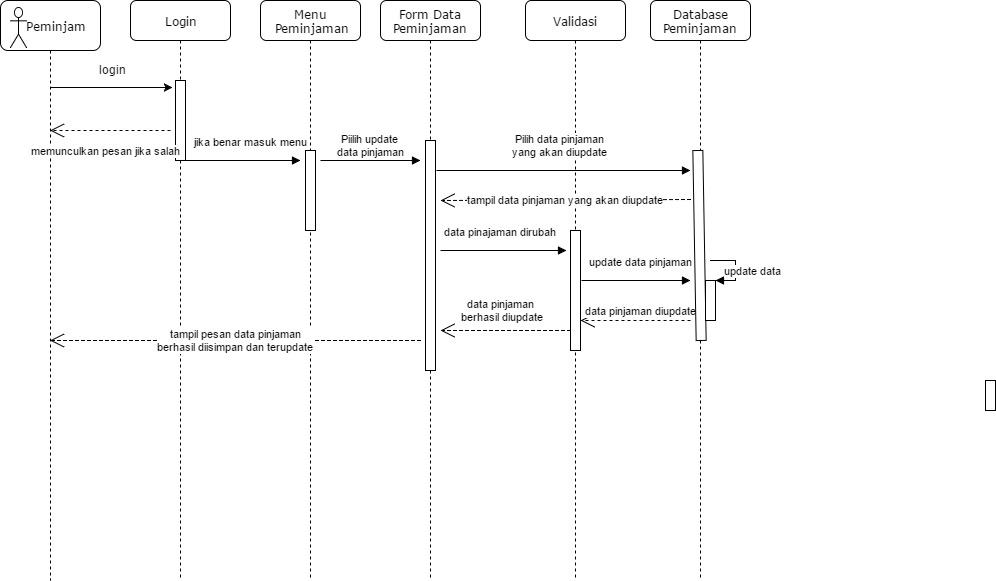
##### Update Kondisi Inventaris



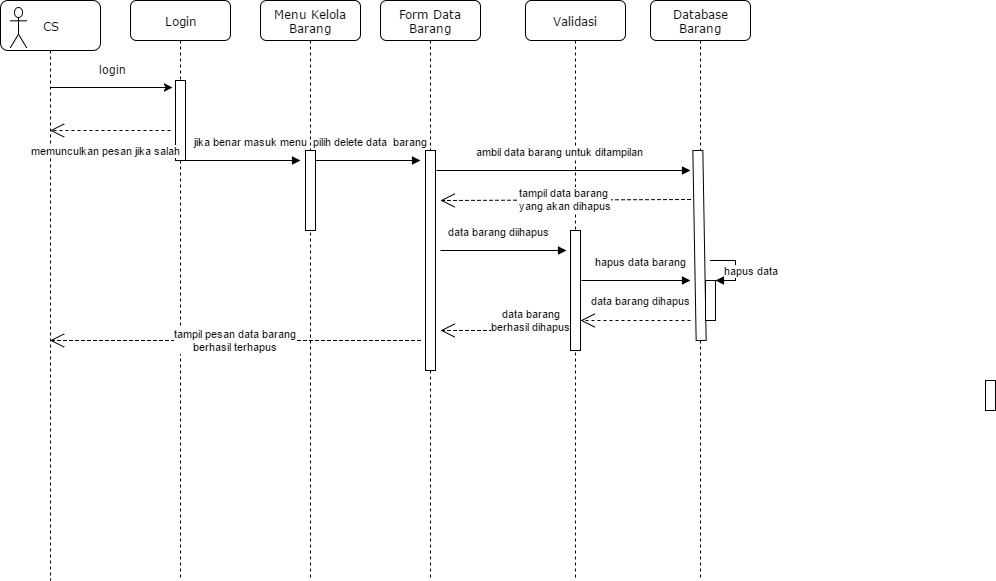
##### Update Mutasi



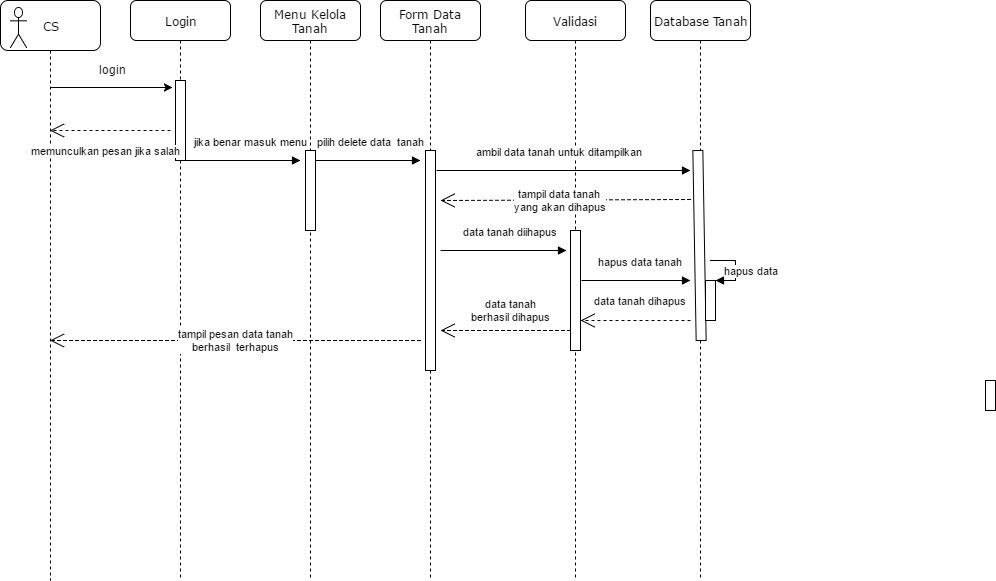
##### Update Pinjaman



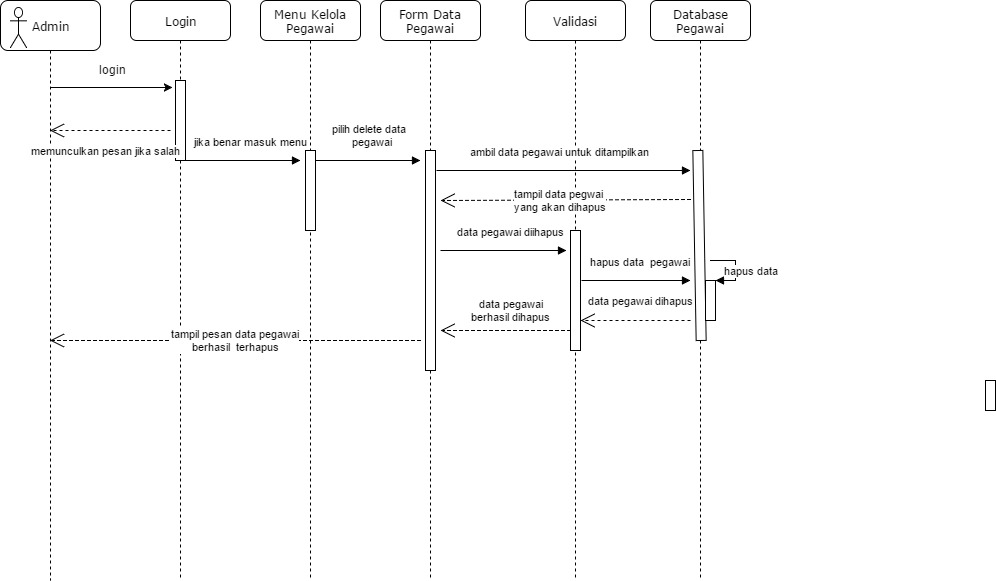
##### Delete Kelola Barang



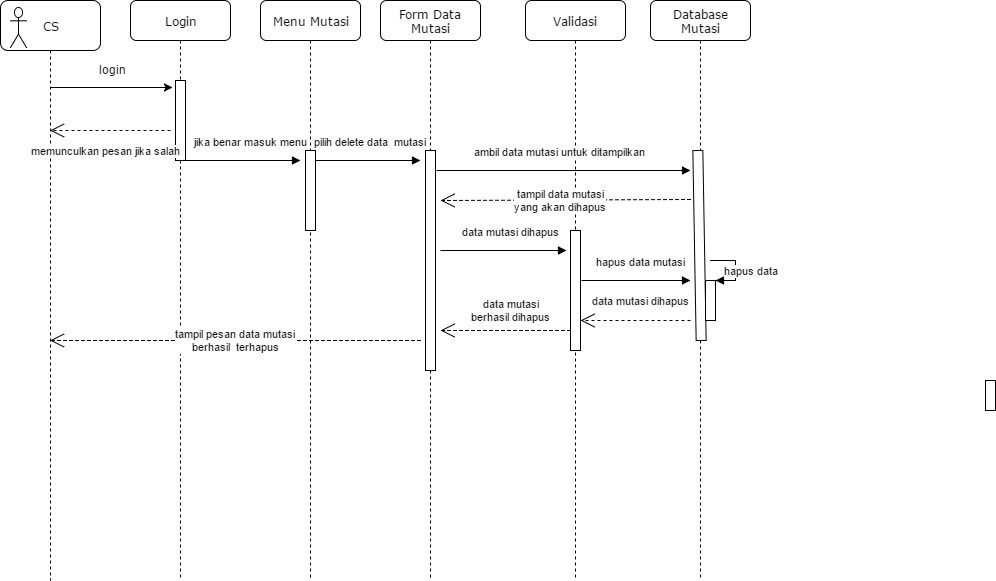
##### Delete Kelola Tanah



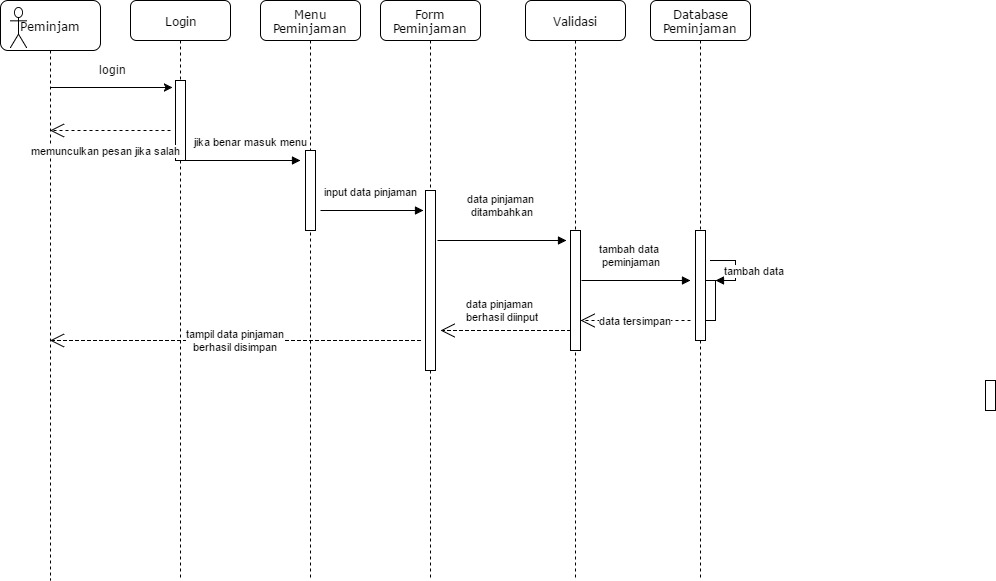
##### Delete Kelola Pegawai



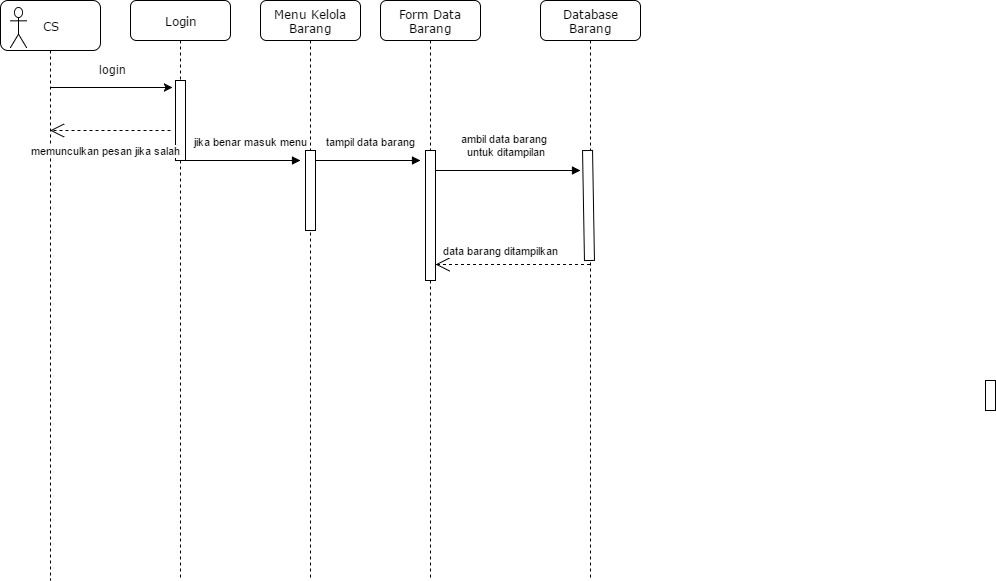
##### Delete Mutasi



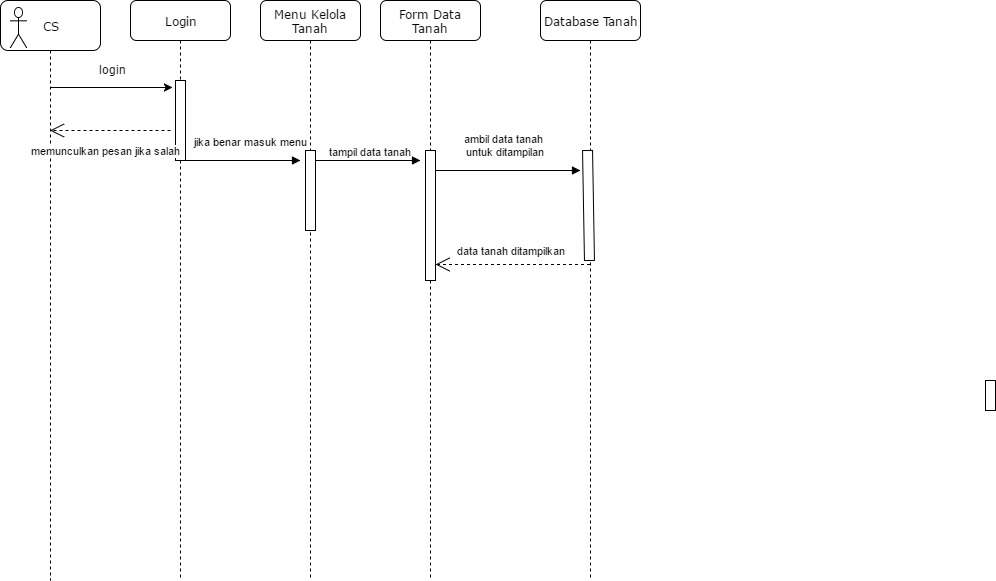
##### Delete Pinjaman



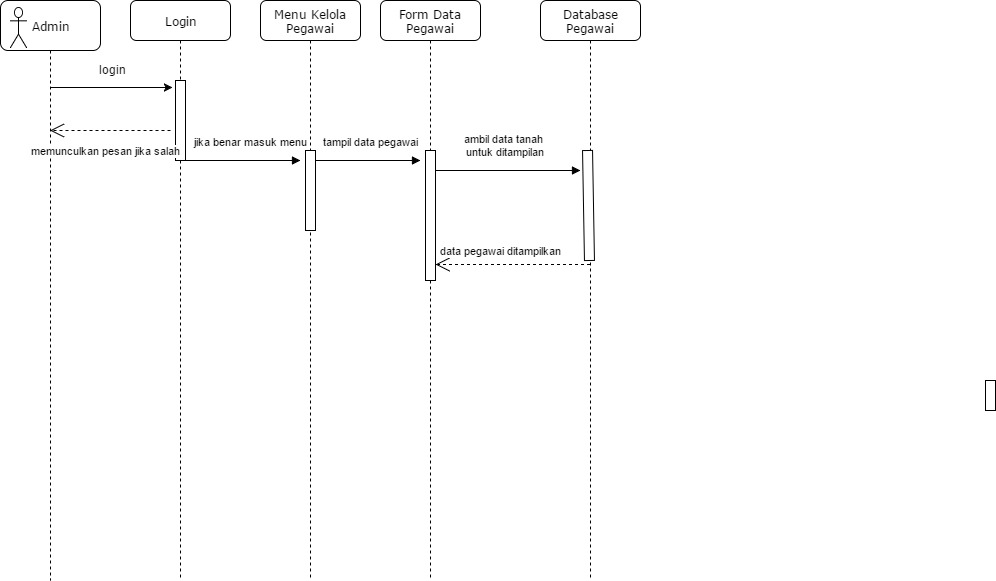
##### View Kelola Barang



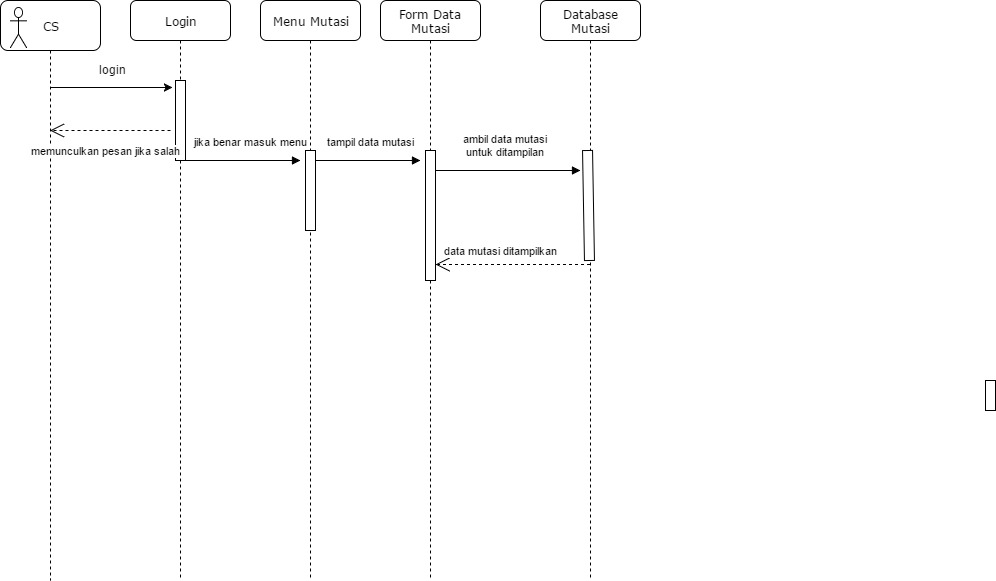
##### View Kelola Tanah



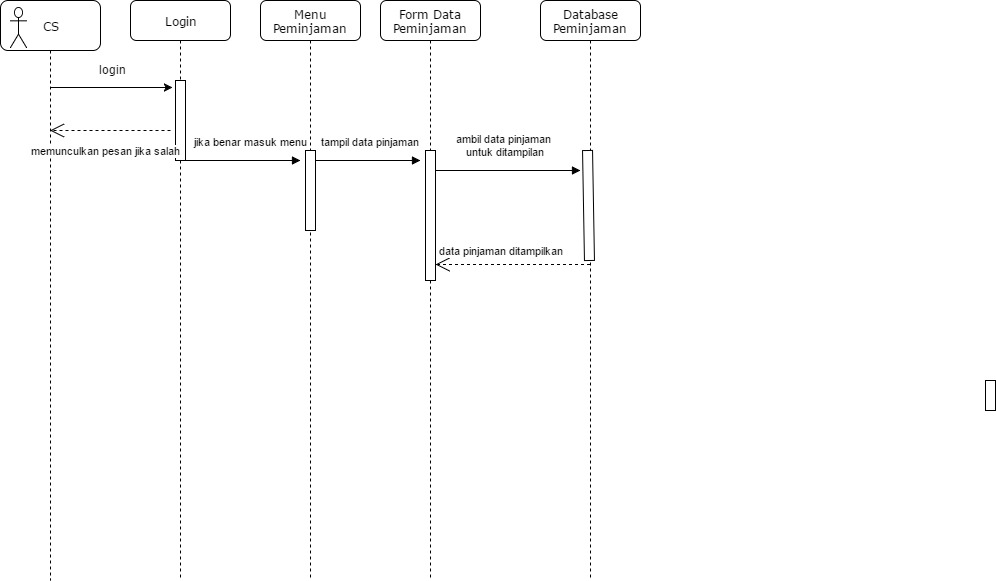
##### View Kelola Pegawai



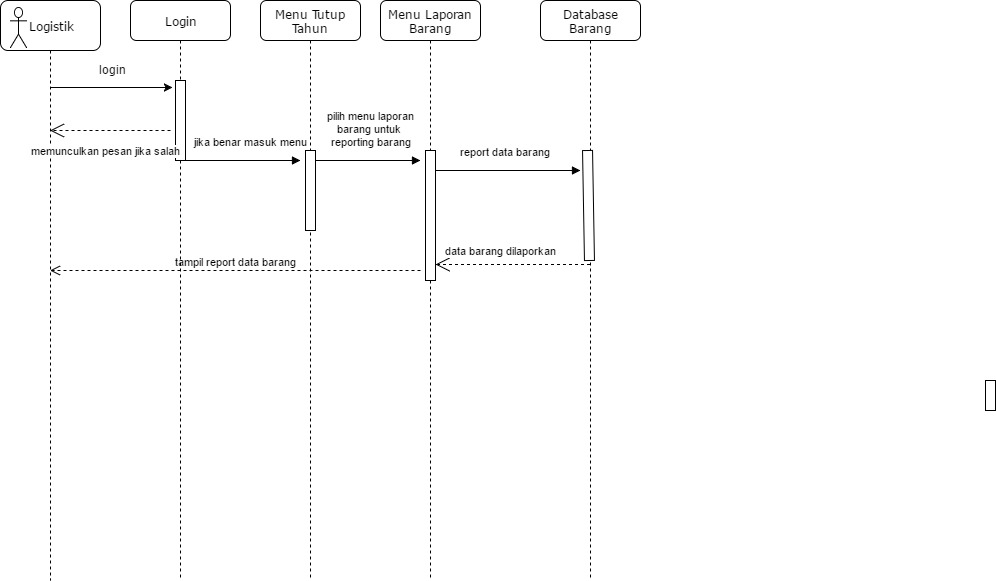
##### View Mutasi



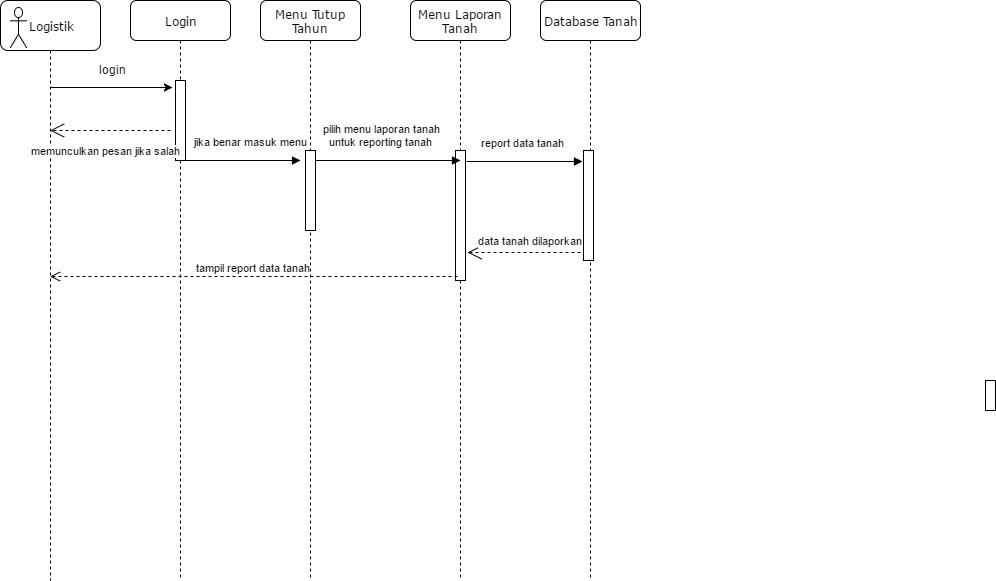
##### View Pinjaman



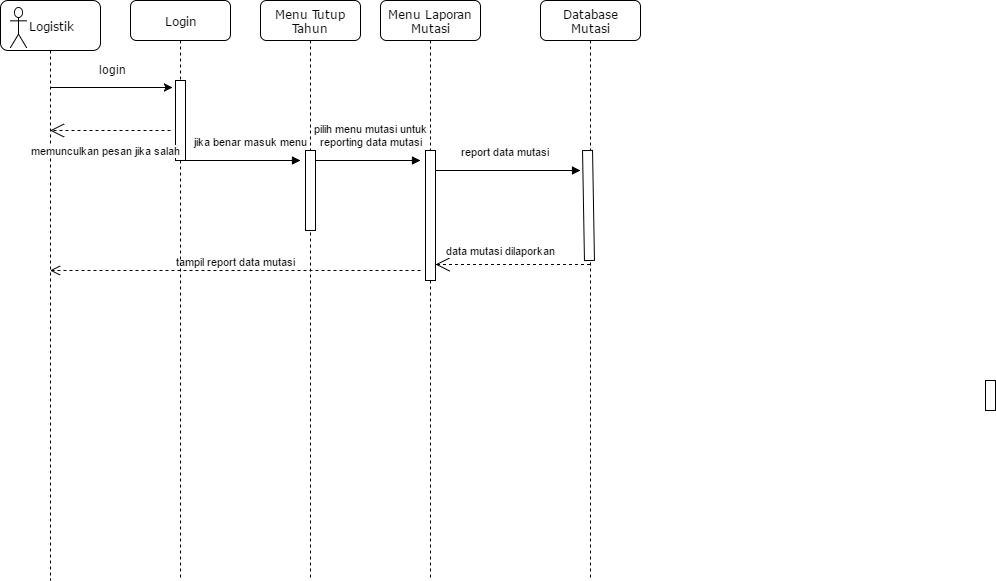
##### Laporan Barang Tutup Tahun



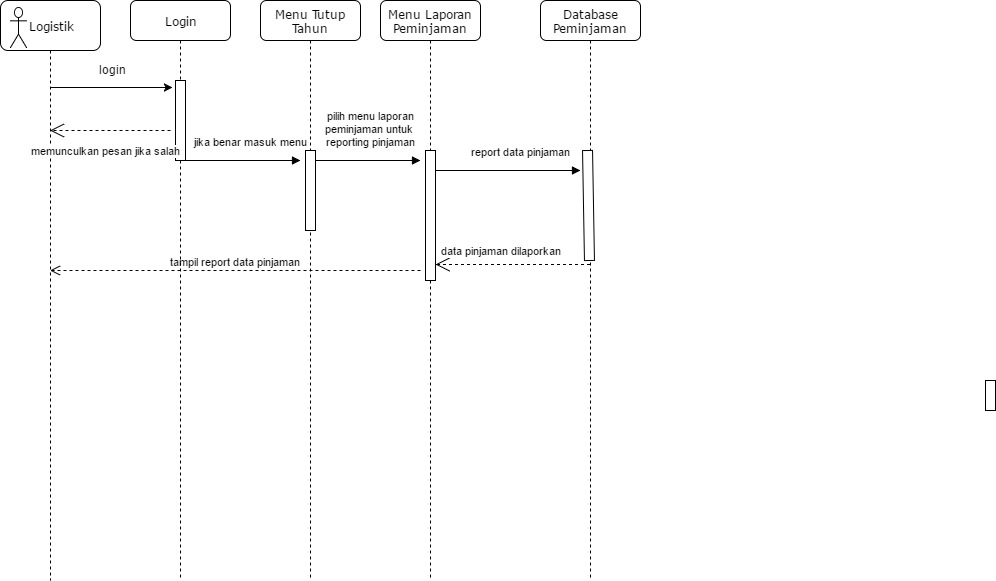
##### Laporan Tanah Tutup Tahun



##### Laporan Mutasi Tutup Tahun



##### Laporan Pinjaman Tutup Tahun



#### Diagram Kelas

*Buatlah diagram kelas untuk use case tersebut.*

## Perancangan Detil Kelas

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

|  |  |  |
| --- | --- | --- |
| *No* | *Nama Kelas Perancangan* | *Nama Kelas Analisis Terkait* |
|  |  |  |
|  |  |  |
|  |  |  |

*Untuk setiap kelas:*

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

### Admin

|  |  |  |
| --- | --- | --- |
| ***Nama Operasi*** | ***Visibility***  ***(private, public)*** | ***Keterangan*** |
| Getter( ) | *Public* | Untuk mengambil nilai di suatu variabel |
| Setter( ) | *Public* | Untuk mengisi nilai di suatu variabel |
| ***Nama Atribut*** | ***Visibility***  ***(private, public)*** | ***Tipe*** |
| *Id\_Admin* | *Private* | *String* |
| *Nama* | *Private* | *String* |
| *Alamat* | *Private* | *String* |
| *Tgl\_lahir* | *Private* | *Date* |

### Login

|  |  |  |
| --- | --- | --- |
| ***Nama Operasi*** | ***Visibility***  ***(private, public)*** | ***Keterangan*** |
| *InputUsernamePass(username:String, password: String)* | *public* | *Login Method* |
| ***Nama Atribut*** | ***Visibility***  ***(private, public)*** | ***Tipe*** |
| *Username* | *Private* | *String* |
| *Password* | *Private* | *String* |

### Pegawai

|  |  |  |
| --- | --- | --- |
| ***Nama Operasi*** | ***Visibility***  ***(private, public)*** | ***Keterangan*** |
| Getter( ) | *Public* | Untuk mengambil nilai di suatu variabel |
| Setter( ) | *Public* | Untuk mengisi nilai di suatu variabel |
| ***Nama Atribut*** | ***Visibility***  ***(private, public)*** | ***Tipe*** |
| *Id\_Pegawai* | *Private* | *String* |

### Peminjam

|  |  |  |
| --- | --- | --- |
| ***Nama Operasi*** | ***Visibility***  ***(private, public)*** | ***Keterangan*** |
| Getter( ) | *Public* | Untuk mengambil nilai di suatu variabel |
| Setter( ) | *Public* | Untuk mengisi nilai di suatu variabel |
| ***Nama Atribut*** | ***Visibility***  ***(private, public)*** | ***Tipe*** |
| *Id\_peminjam* | *Private* | *String* |
| *Nama* | *Private* | *String* |

### Logistik

|  |  |  |
| --- | --- | --- |
| ***Nama Operasi*** | ***Visibility***  ***(private, public)*** | ***Keterangan*** |
| Getter( ) | *Public* | Untuk mengambil nilai di suatu variabel |
| Setter( ) | *Public* | Untuk mengisi nilai di suatu variabel |
| ***Nama Atribut*** | ***Visibility***  ***(private, public)*** | ***Tipe*** |
| *Nama* | *Private* | *String* |
| *Alamat* | *Private* | *String* |
| *Tgl\_lahir* | *Private* | *Date* |

### CS

|  |  |  |
| --- | --- | --- |
| ***Nama Operasi*** | ***Visibility***  ***(private, public)*** | ***Keterangan*** |
| Getter( ) | *Public* | Untuk mengambil nilai di suatu variabel |
| Setter( ) | *Public* | Untuk mengisi nilai di suatu variabel |
| ***Nama Atribut*** | ***Visibility***  ***(private, public)*** | ***Tipe*** |
| *Nama* | *Private* | *String* |
| *Alamat* | *Private* | *String* |
| *Tgl\_lahir* | *Private* | *Date* |

### Karyawan

|  |  |  |
| --- | --- | --- |
| ***Nama Operasi*** | ***Visibility***  ***(private, public)*** | ***Keterangan*** |
| Getter( ) | *Public* | Untuk mengambil nilai di suatu variabel |
| Setter( ) | *Public* | Untuk mengisi nilai di suatu variabel |
| ***Nama Atribut*** | ***Visibility***  ***(private, public)*** | ***Tipe*** |
| *Nama* | *Private* | *String* |
| *Alamat* | *Private* | *String* |
| *Tgl\_lahir* | *Private* | *Date* |

### Tanah

|  |  |  |
| --- | --- | --- |
| ***Nama Operasi*** | ***Visibility***  ***(private, public)*** | ***Keterangan*** |
| Getter( ) | *Public* | Untuk mengambil nilai di suatu variabel |
| Setter( ) | *Public* | Untuk mengisi nilai di suatu variabel |
| input( ) | *Public* | Method input tanah |
| Update() | *Public* | Method update tanah |
| delete | *Public* | Method hapus tanah |
| ***Nama Atribut*** | ***Visibility***  ***(private, public)*** | ***Tipe*** |
| *Id\_tanah* | *Private* | *String* |
| *namaTanah* | *Private* | *String* |
| *jumlahTanah* | *Private* | *integer* |

### Barang

|  |  |  |
| --- | --- | --- |
| ***Nama Operasi*** | ***Visibility***  ***(private, public)*** | ***Keterangan*** |
| Getter( ) | *Public* | Untuk mengambil nilai di suatu variabel |
| Setter( ) | *Public* | Untuk mengisi nilai di suatu variabel |
| input( ) | *Public* | Method input barang |
| Update() | *Public* | Method update barang |
| delete | *Public* | Method hapus barang |
| ***Nama Atribut*** | ***Visibility***  ***(private, public)*** | ***Tipe*** |
| *Id\_barang* | *Private* | *String* |
| *namaBarang* | *Private* | *String* |
| *jumlahBarang* | *Private* | *integer* |

### Peminjaman

|  |  |  |
| --- | --- | --- |
| ***Nama Operasi*** | ***Visibility***  ***(private, public)*** | ***Keterangan*** |
| Getter( ) | *Public* | Untuk mengambil nilai di suatu variabel |
| Setter( ) | *Public* | Untuk mengisi nilai di suatu variabel |
| input( ) | *Public* | Method input peminjaman |
| Update() | *Public* | Method update peminjaman |
| Delete() | *Public* | Method hapus peminjaman |
| ***Nama Atribut*** | ***Visibility***  ***(private, public)*** | ***Tipe*** |
| *Id\_pinjaman* | *Private* | *String* |
| *Tglpinjaman* | *Private* | *Date* |
| *Tglkembali* | *Private* | *Date* |

### Kondisi Inventaris

|  |  |  |
| --- | --- | --- |
| ***Nama Operasi*** | ***Visibility***  ***(private, public)*** | ***Keterangan*** |
| Getter( ) | *Public* | Untuk mengambil nilai di suatu variabel |
| Setter( ) | *Public* | Untuk mengisi nilai di suatu variabel |
| ***Nama Atribut*** | ***Visibility***  ***(private, public)*** | ***Tipe*** |
| *Nama\_inventaris* | *Private* | *String* |
| *Status* | *Private* | *boolean* |

### Mutasi

|  |  |  |
| --- | --- | --- |
| ***Nama Operasi*** | ***Visibility***  ***(private, public)*** | ***Keterangan*** |
| Getter( ) | *Public* | Untuk mengambil nilai di suatu variabel |
| Setter( ) | *Public* | Untuk mengisi nilai di suatu variabel |
| input( ) | *Public* | Method input peminjaman |
| Update() | *Public* | Method update peminjaman |
| Delete() | *Public* | Method hapus peminjaman |
| Mutasi(id\_tanah,idfakultas,idUniversitas) | *Public* | Method untuk mengambil id\_Tanah, idfakultas, idUniversitas |
| Mutasi(id\_barang,idfakultas,idUniversitas) | *Public* | Method untuk mengambil id\_barang, idfakultas, idUniversitas |
| ***Nama Atribut*** | ***Visibility***  ***(private, public)*** | ***Tipe*** |
| *IdMutasi* | *Private* | *String* |
| *Mutasidari* | *Private* | *String* |
| *Mutasike* | *Private* | *String* |

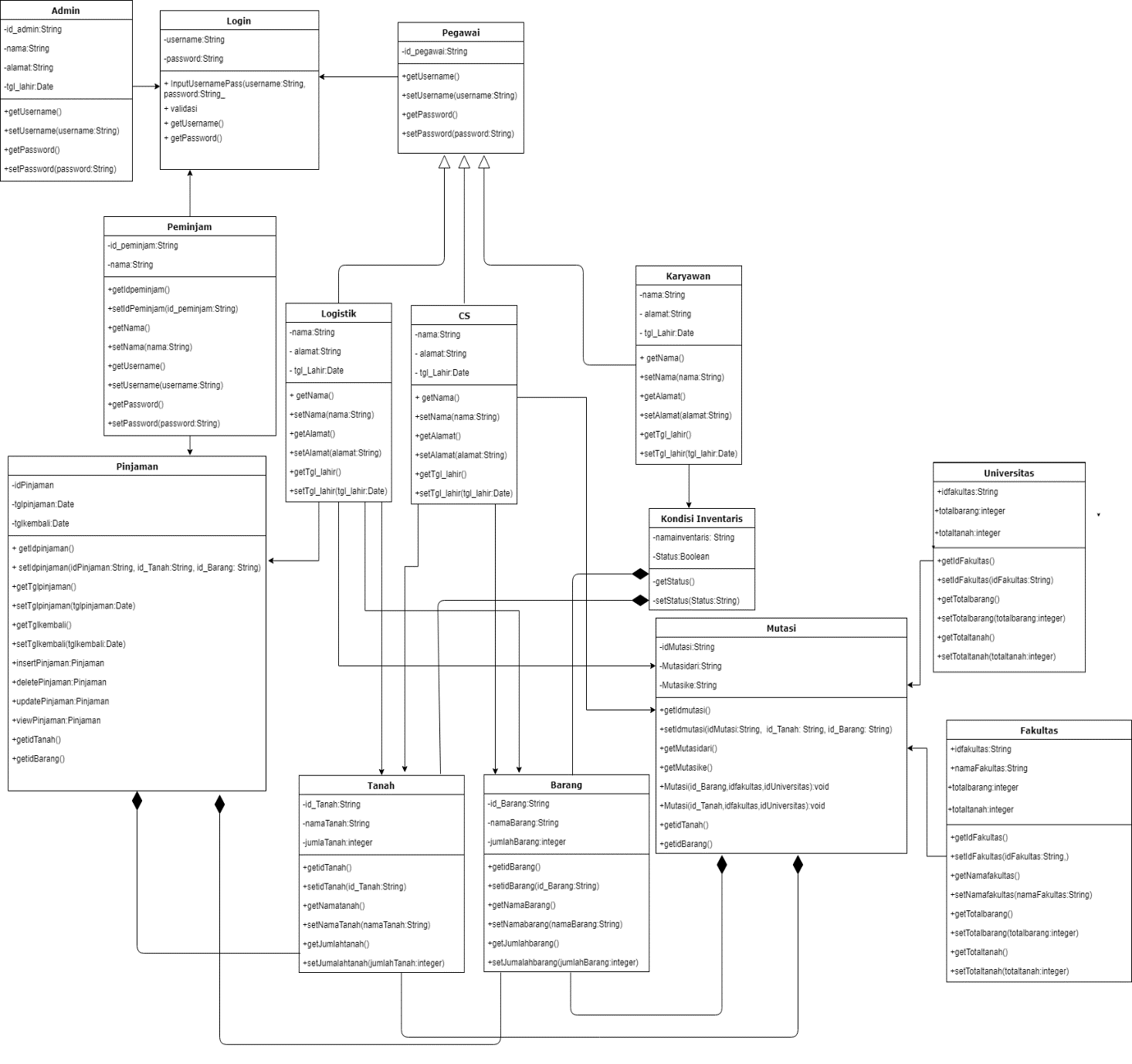
### Fakultas

|  |  |  |
| --- | --- | --- |
| ***Nama Operasi*** | ***Visibility***  ***(private, public)*** | ***Keterangan*** |
| Getter( ) | *Public* | Untuk mengambil nilai di suatu variabel |
| Setter( ) | *Public* | Untuk mengisi nilai di suatu variabel |
| ***Nama Atribut*** | ***Visibility***  ***(private, public)*** | ***Tipe*** |
| *Idfakultas* | *Private* | *String* |
| *Totaltanah* | *Private* | *Integer* |
| *Totalbarang* | *Private* | *Integer* |

### Universitas

|  |  |  |
| --- | --- | --- |
| ***Nama Operasi*** | ***Visibility***  ***(private, public)*** | ***Keterangan*** |
| Getter( ) | *Public* | Untuk mengambil nilai di suatu variabel |
| Setter( ) | *Public* | Untuk mengisi nilai di suatu variabel |
| ***Nama Atribut*** | ***Visibility***  ***(private, public)*** | ***Tipe*** |
| *Idfakultas* | *Private* | *String* |
| *Totaltanah* | *Private* | *Integer* |
| *Totalbarang* | *Private* | *Integer* |

## Diagram Kelas Keseluruhan

**

## Algoritma/Query

## 3.4.1

Nama Kelas : Login

Nama Operasi : Login

Algoritma :

Begin

If(username != data.username || password!= data.password then

Output(“Login Failed”);

Else

Login\_Berhasil

Show\_dashboard

End

## 3.4.2

Nama Kelas : Barang

Nama Operasi : Input data barang

Algoritma :

Begin

Total\_Barang\_yg\_dimasukkan = input

If(list.barang == NULL) then

Barang.ID <- ID\_Barang

Barang.nama<-NamaBarang

Barang.jumlah<-jumlahBarang

Else

For(i=0;i<input-1;i++){

Createlist(barang)

Barang.ID <- ID\_Barang

Barang.nama<-NamaBarang

Barang.jumlah<-jumlahBarang

Output(“Input”, input, “data Berhasil”)

End

## 3.4.3

Nama Kelas : Tanah

Nama Operasi : Input data tanah

Algoritma :

Begin

Total\_Tanah\_yg\_dimasukkan = input

If(list.tanah == NULL) then

Tanah.ID <- ID\_Tanah

Tanah.nama<-NamaTanah

Tanah.jumlah<-jumlahTanah

Else

For(i=0;i<input-1;i++){

Createlist(Tanah)

Tanah.ID <- ID\_Tanah

Tanah.nama<-NamaTanah

Tanah.jumlah<-jumlahTanah

Output(“Input”, input, “data Berhasil”)

End

## 3.4.4

Nama Kelas : Barang

Nama Operasi : View data barang

Algoritma :

If(list\_Barang!=null)then

Output(“List Data Barang”)

Show\_DataBarang

Else

Output(List barang Masih Kosong)

End

## 3.4.5

Nama Kelas : Tanah

Nama Operasi : View data tanah

Algoritma :

If(list\_Tanah!=null)then

Output(“List Data Tanah”)

Show\_DataTanah

Else

Output(List tanah Masih Kosong)

End

## 3.4.6

Nama Kelas : Mutasi

Nama Operasi : Mutasi

Algoritma :

begin

Masukan = user\_input\_ID

If(Click == barang)then

For(i=0,i<input,i++)do

If(barang.id == masukan) then

If(click == edit data) then

Edit\_data\_Barang

Else if(click == hapus\_data)then

Hapus\_data\_Barang

Output(“ID yg dimasukkan tidak ketemu pada data Barang”)

Else if(click == tanah) then

For(i=0,i<input,i++)do

If(tanah.id == masukan) then

If(click == edit data) then

Edit\_data\_tanah

Else if(click == hapus\_data)then

Hapus\_data\_tanah

Output(“ID yg dimasukkan tidak ketemu pada data tanah”)

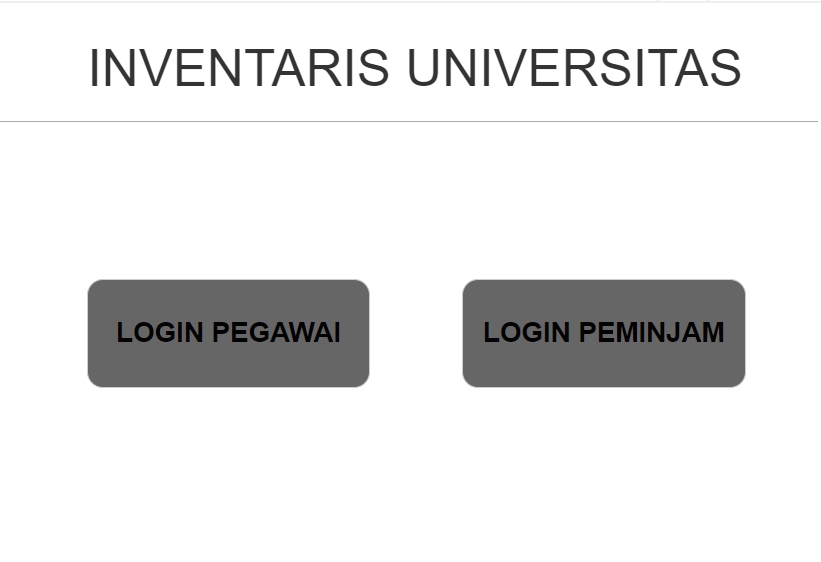
end

## Diagram Statechart

*Bagian ini hanya diisi jika ada kelas yang memiliki atribut dengan value yang dinamis (berubah-ubah sesuai state atau kondisi). Jika tidak ada kedinamisan value dari atribut kelas tersebut, diagram ini tidak perlu dibuat. Perubahan status kelas tersebut harus digambarkan dalam bentuk diagram statechart. Boleh dibuat subbab per kelas.*

## Perancangan Antarmuka

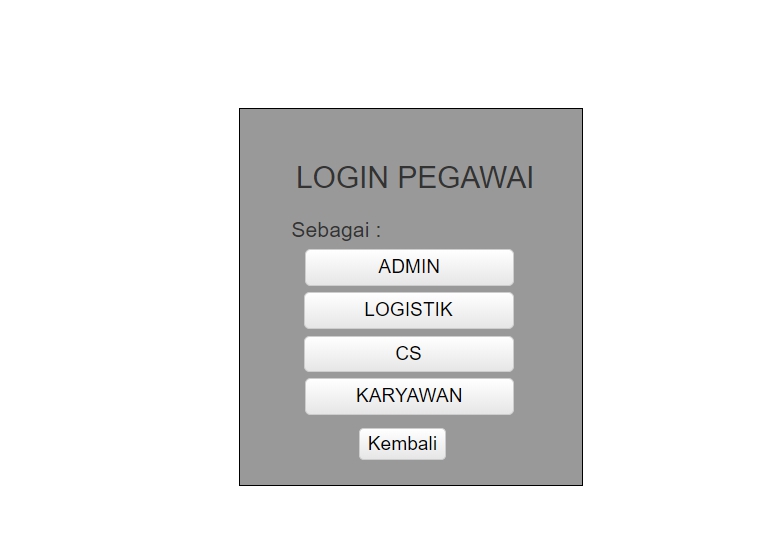
## 3.6.1



*Antarmuka : {Tampilan awal}*

| **Id\_Objek** | **Jenis** | **Nama** | **Keterangan** |
| --- | --- | --- | --- |
| *btnLoginPegawai* | *Button* | *Login Pegawai* | Button untuk login pegawai |
| *btnLoginPeminjam* | *Button* | *Login Peminjam* | Button untuk login peminjam |

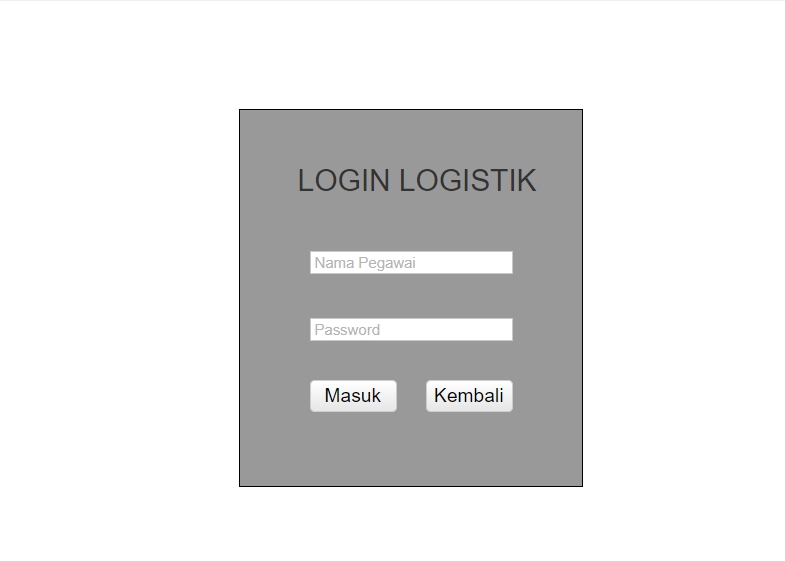
## 3.6.2



*Antarmuka : {Tampilan Login Pegawai}*

| **Id\_Objek** | **Jenis** | **Nama** | **Keterangan** |
| --- | --- | --- | --- |
| *btnLoginAdmin* | *Button* | *Login Pegawai* | Button untuk login pegawai |
| *btnLoginLogistik* | *Button* | *Login Peminjam* | Button untuk login peminjam |
| *btnLoginCS* | *Button* | *Login CS* | Button untuk login cs |
| *btnLoginKaryawan* | *Button* | *Login Kayawan* | Button untuk login kayawan |
| *btnBack* | *Button* | *Kembali* | Kembali ke tampilan awal |

## 3.6.3



*Antarmuka : {Tampilan Login Logistik}*

| **Id\_Objek** | **Jenis** | **Nama** | **Keterangan** |
| --- | --- | --- | --- |
| *Inputusername* | *Textfield* | *Nama Pegawai* | Textfield untuk input nama pegawai |
| *InputPassword* | *Textfield* | *Password* | Textfield untuk input password |
| *btnMasuk* | *Button* | *Masuk* | Button untuk masuk ke halaman logistik |
| *btnBack* | *Button* | *Kembali* | Kembali ke tampilan awal |

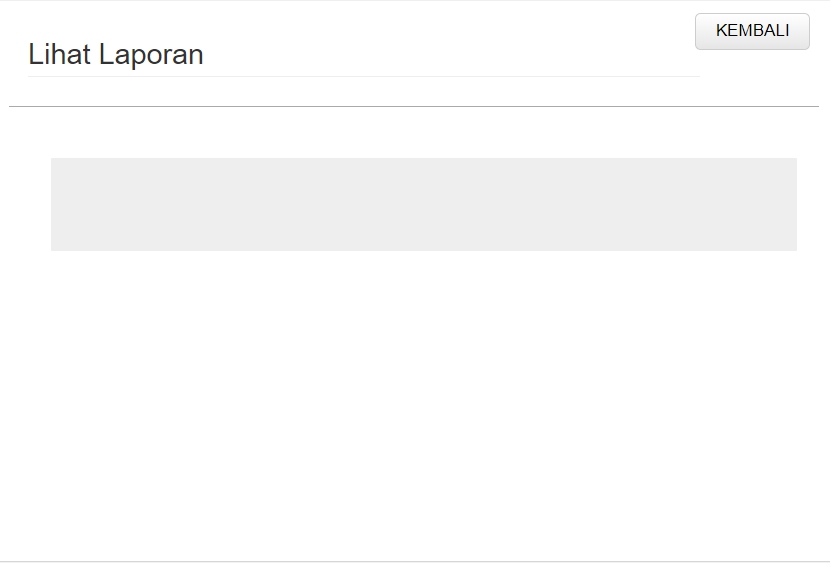
## 3.6.4



*Antarmuka : {Tampilan Login Logistik}*

| **Id\_Objek** | **Jenis** | **Nama** | **Keterangan** |
| --- | --- | --- | --- |
| *Btnlaporan* | *Button* | *LAPORAN* | Button untuk masuk ke menu laporan |
| *Btnlogout* | *Button* | *LOG OUT* | Button untuk keluar dari web |

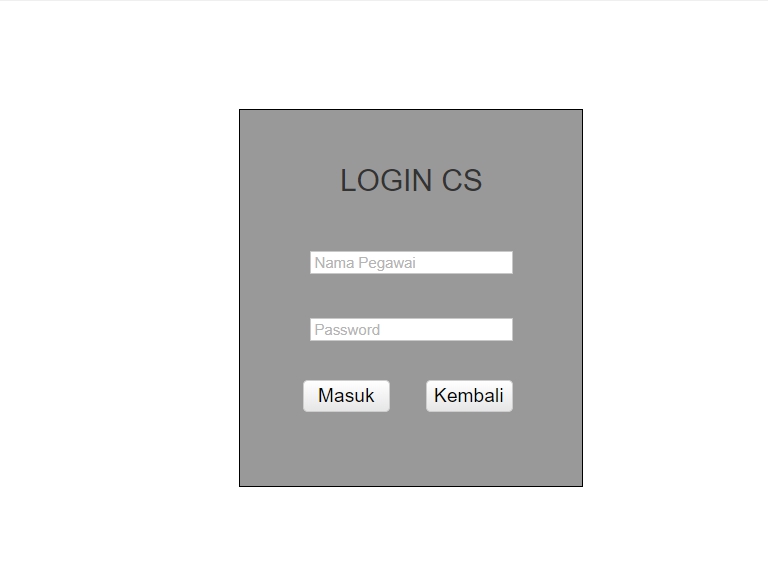
## 3.6.5



*Antarmuka : {Tampilan Login Logistik}*

| **Id\_Objek** | **Jenis** | **Nama** | **Keterangan** |
| --- | --- | --- | --- |
| *Btnlaporan* | *Button* | *LAPORAN* | Button untuk masuk ke menu laporan |
| *BtnKembali* | *Button* | *KEMBALI* | Button untuk kembali ke halaman sebelumnyya |

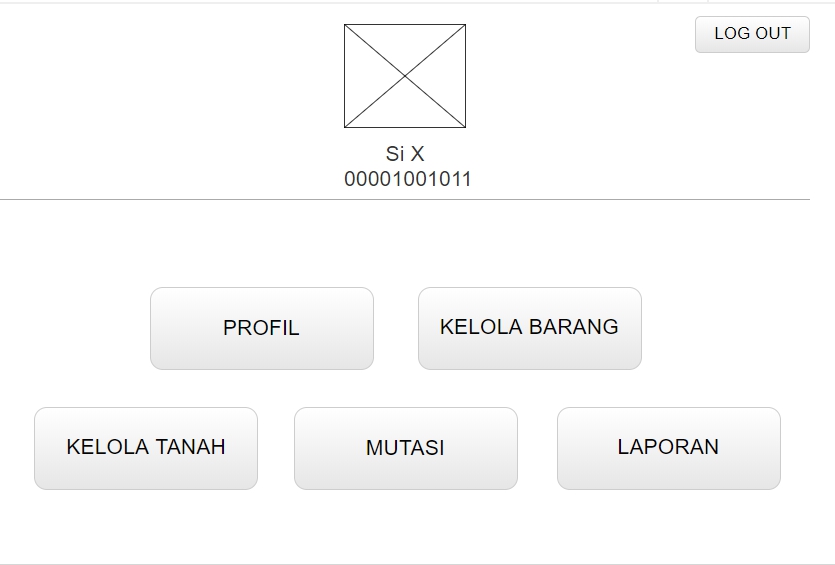
## 3.6.6



*Antarmuka : {Tampilan Login CS}*

| **Id\_Objek** | **Jenis** | **Nama** | **Keterangan** |
| --- | --- | --- | --- |
| *Inputusername* | *Textfield* | *Nama Pegawai* | Textfield untuk input nama pegawai |
| *InputPassword* | *Textfield* | *Password* | Textfield untuk input password |
| *btnMasuk* | *Button* | *Masuk* | Button untuk masuk ke halaman CS |
| *btnBack* | *Button* | *Kembali* | Kembali ke tampilan awal |
| *Logincs* | *Text* | *LOGIN CS* | Untuk menampilka tulisan ‘LOGIN CS’ |

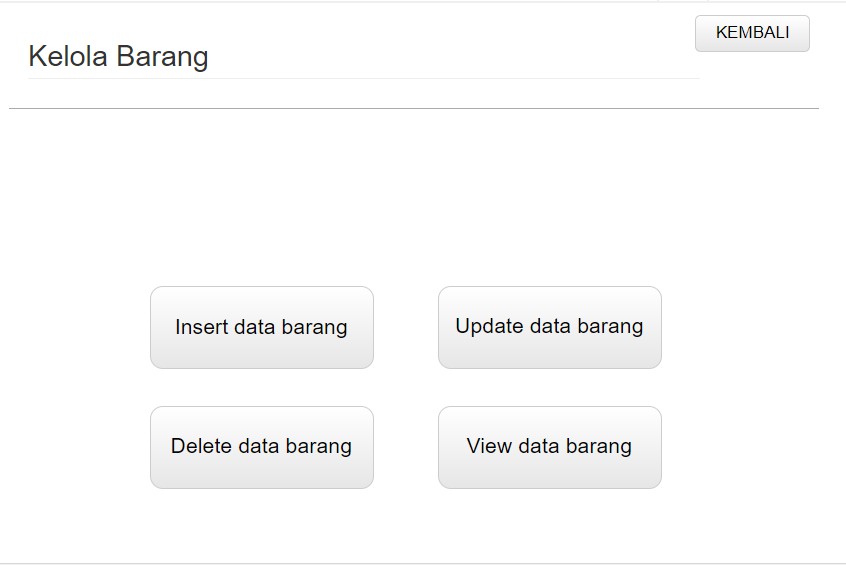
## 3.6.7



*Antarmuka : {Tampilan Login Logistik}*

| **Id\_Objek** | **Jenis** | **Nama** | **Keterangan** |
| --- | --- | --- | --- |
| *BtnProfil* | *Button* | *PROFIL* | Button untuk masuk ke profil cs |
| *Btnkelolabarang* | *Button* | *KELOLA BARANG* | Button untuk masuk ke kelola barang |
| *Btnkelolatanah* | *Button* | *KELOLA TANAH* | Button untuk masuk ke kelola tanah |
| *Btnkelomutasi* | *Button* | *MUTASI* | Button untuk masuk ke kelola MUTASI |
| *Btnlaporan* | *Button* | *KELOLA TANAH* | Button untuk masuk ke kelola tanah |

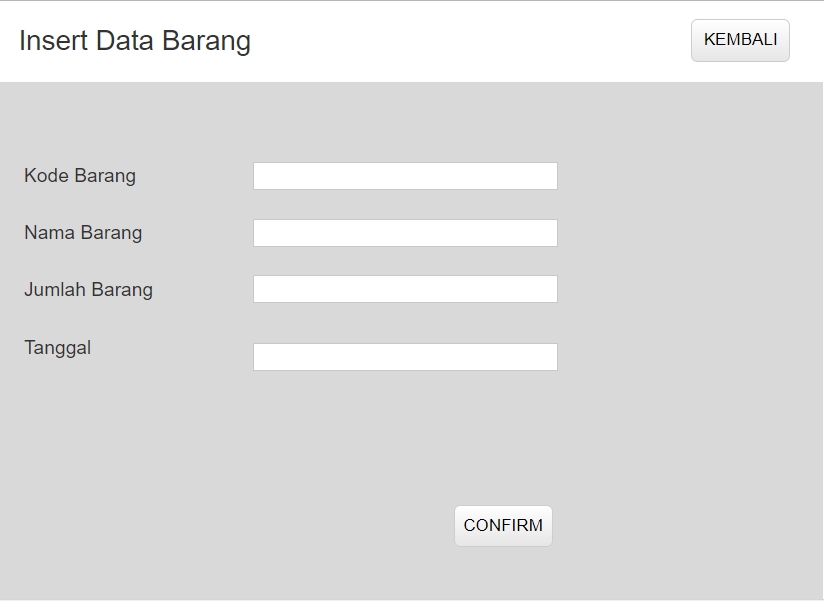
## 3.6.8



*Antarmuka : {Tampilan Kelola Barang}*

| **Id\_Objek** | **Jenis** | **Nama** | **Keterangan** |
| --- | --- | --- | --- |
| *BtnInsertDataBarang* | *Button* | *Insert data barang* | Button untuk menginputkan data barang |
| *BtnDeleteDataBarang* | *Button* | *Delete data barang* | Button untuk menghapus data barang |
| *BtnUpdateDataBarang* | *Button* | *Update data barang* | Button untuk mengupdate data barang |
| *BtnViewDataBarang* | *Button* | *View data barang* | Button untuk menampilkan data barang |

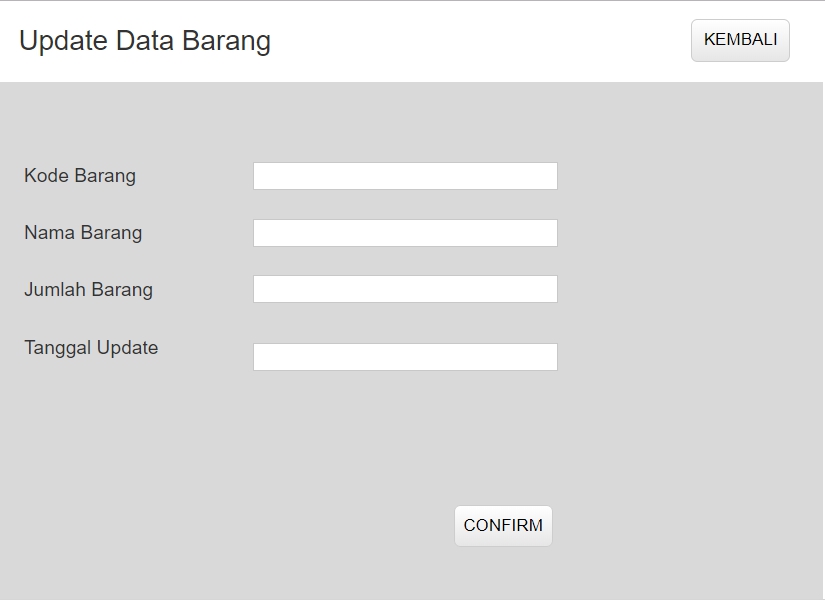
## 3.6.9



*Antarmuka : {Tampilan Insert Data Barang}*

| **Id\_Objek** | **Jenis** | **Nama** | **Keterangan** |
| --- | --- | --- | --- |
| *InputKodeBarang* | *Textfield* | *Kode Barang* | Textfield untuk input kode barang |
| *InputNamaBarang* | *Textfield* | *Nama Barang* | Textfield untuk input nama barang |
| *InputJumlahBarang* | *Textfield* | *Jumlah Barang* | Textfield untuk input jumlah barang |
| *InputTanggal* | *Textfield* | *Tanggal* | Textfield untuk input tanggal |
| *btnConfirm* | *Button* | *Confirm* | Mengkonfirmasi data barang yang telah di insert |
| *btnBack* | *Button* | *Kembali* | Kembali ke tampilan kelola barang |

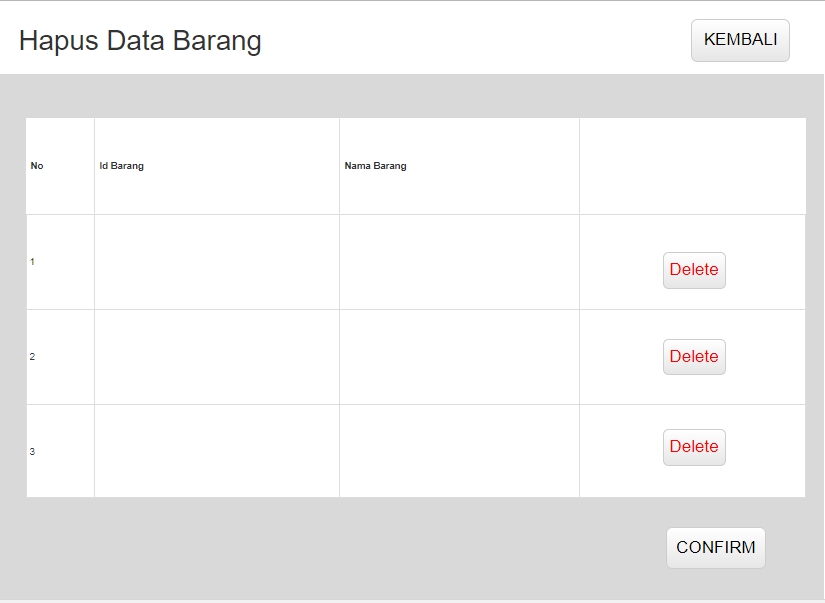
## 3.6.10



*Antarmuka : {Tampilan Update Data Barang}*

| **Id\_Objek** | **Jenis** | **Nama** | **Keterangan** |
| --- | --- | --- | --- |
| *UpdateKodeBarang* | *Textfield* | *Kode Barang* | Textfield untuk mengupdate kode barang |
| *UpdateNamaBarang* | *Textfield* | *Nama Barang* | Textfield untuk mengupdate nama barang |
| *UpdateJumlahBarang* | *Textfield* | *Jumlah Barang* | Textfield untuk mengupdate jumlah barang |
| *UpdateTanggal* | *Textfield* | *Tanggal* | Textfield untuk mengupdate tanggal |
| *btnConfirm* | *Button* |  | Mengkonfirmasi data barang yang telah di update |
| *btnBack* | *Button* | *Kembali* | Kembali ke tampilan kelola barang |

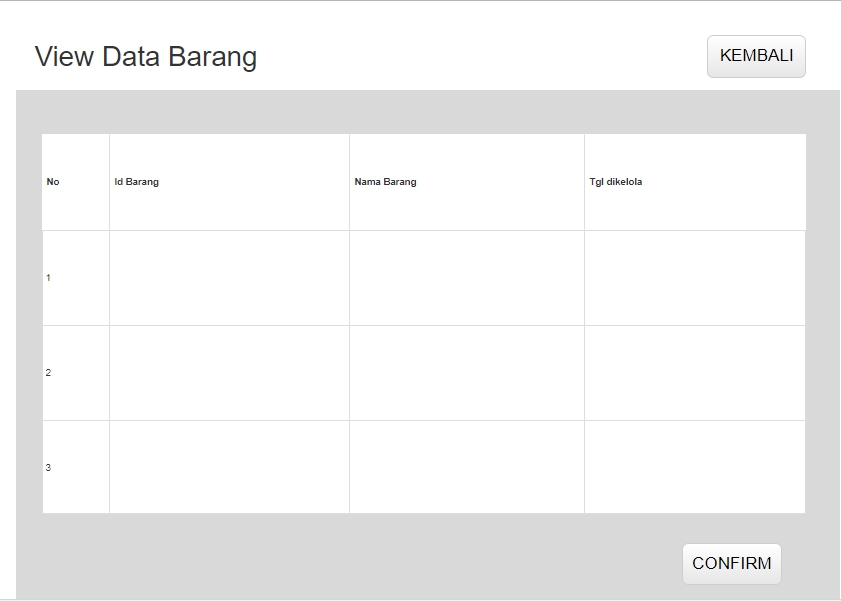
## 3.6.11



*Antarmuka : {Tampilan view data barang}*

| **Id\_Objek** | **Jenis** | **Nama** | **Keterangan** |
| --- | --- | --- | --- |
| *btnDelete* | *Button* | *Delete* | Untuk menghapus data yang ingin dihapus |
| *btnConfirm* | *Button* | *Confirm* | Mengkonfirmasi data barang yang telah di hapus |
| *btnBack* | *Button* | *Kembali* | Kembali ke tampilan kelola barang |

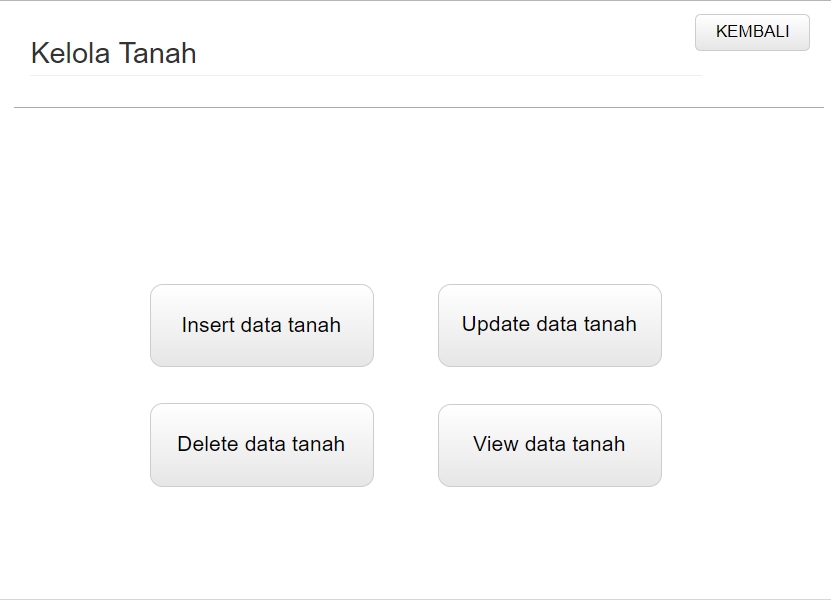
## 3.6.12



*Antarmuka : {Tampilan View Data Barang}*

| **Id\_Objek** | **Jenis** | **Nama** | **Keterangan** |
| --- | --- | --- | --- |
| *btnConfirm* | *Button* | *Confirm* | Mengkonfirmasi data barang yang di tampilkan |
| *btnBack* | *Button* | *Kembali* | Kembali ke tampilan kelola barang |

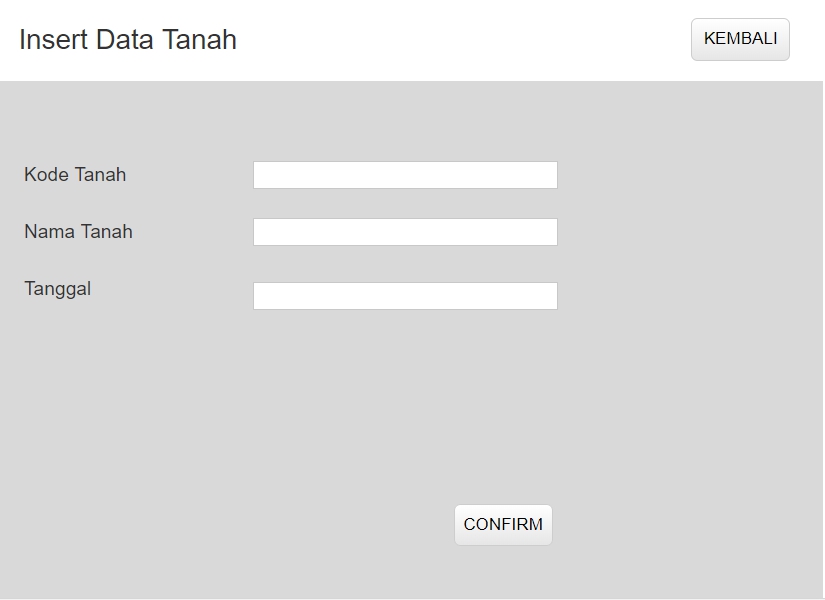
## 3.6.13



*Antarmuka : {Tampilan Kelola Tanah}*

| **Id\_Objek** | **Jenis** | **Nama** | **Keterangan** |
| --- | --- | --- | --- |
| *BtnInsertDataTanah* | *Button* | *Insert data tanah* | Button untuk menginputkan data tanah |
| *BtnDeleteDataTanah* | *Button* | *Delete data tanah* | Button untuk menghapus data tanah |
| *BtnUpdateDataTanah* | *Button* | *Update data tanah* | Button untuk mengupdate data tanah |
| *BtnViewDataTanah* | *Button* | *View data tanah* | Button untuk menampilkan data tanah |
| *btnBack* | *Button* | *Kembali* | Kembali ke tampilan sebelumnya |

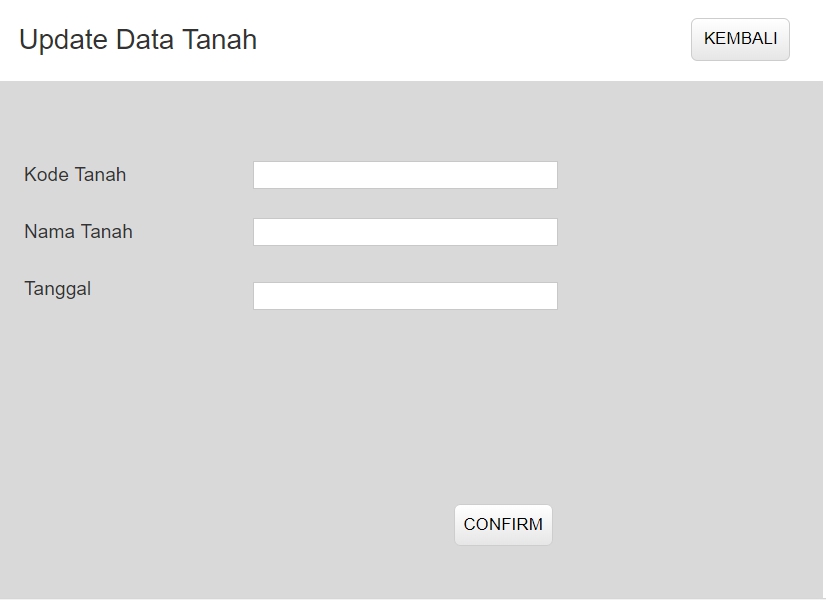
## 3.6.14



*Antarmuka : {Tampilan Insert Data Tanah}*

| **Id\_Objek** | **Jenis** | **Nama** | **Keterangan** |
| --- | --- | --- | --- |
| *InputKodeTanah* | *Textfield* | *Kode Tanah* | Textfield untuk input kode tanah |
| *InputNamaTanah* | *Textfield* | *Nama Tanah* | Textfield untuk input nama tanah |
| *InputTanggal* | *Textfield* | *Tanggal* | Textfield untuk input tanggal |
| *btnConfirm* | *Button* | *Cofirm* | Mengkonfirmasi data tanah yang telah di insert |
| *btnBack* | *Button* | *Kembali* | Kembali ke tampilan kelola tanah |

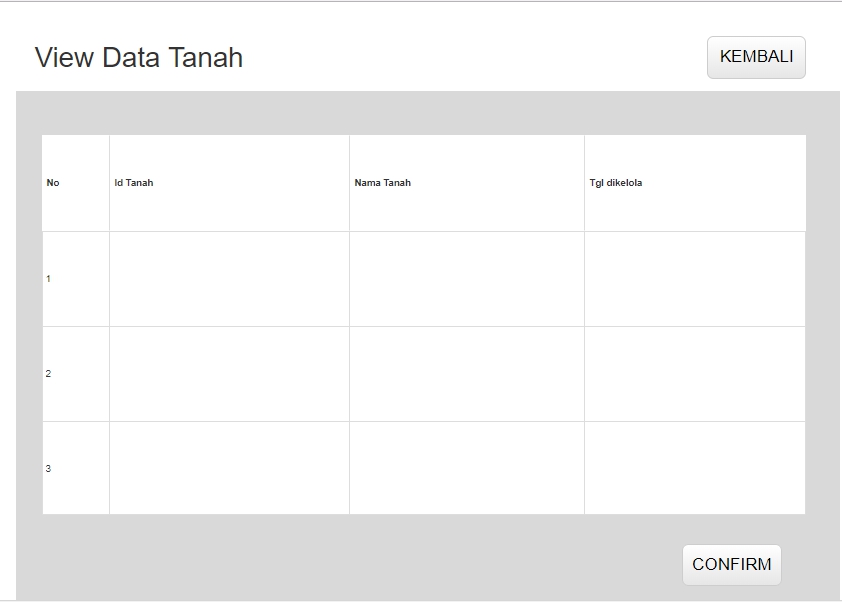
## 3.6.15



*Antarmuka : {Tampilan Update Data Tanah}*

| **Id\_Objek** | **Jenis** | **Nama** | **Keterangan** |
| --- | --- | --- | --- |
| *UpdateKodeTanah* | *Textfield* | *Kode Barang* | Textfield untuk mengupdate kode tanah |
| *UpdateNamaTanah* | *Textfield* | *Nama Barang* | Textfield untuk mengupdate nama tanah |
| *UpdateTanggal* | *Textfield* | *Tanggal* | Textfield untuk mengupdate tanggal |
| *btnConfirm* | *Button* | *Confirm* | Mengkonfirmasi data tanah yang telah di update |
| *btnBack* | *Button* | *Kembali* | Kembali ke tampilan kelola tanah |

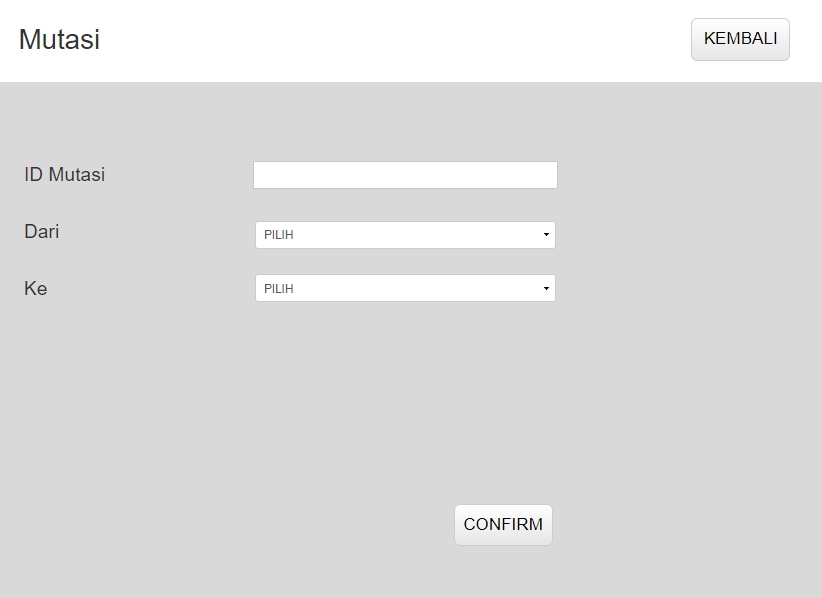
## 3.6.16



*Antarmuka : {Tampilan View Data Tanah}*

| **Id\_Objek** | **Jenis** | **Nama** | **Keterangan** |
| --- | --- | --- | --- |
| *btnConfirm* | *Button* | *Confirm* | Mengkonfirmasi data tanah yang telah di tampilkan |
| *btnBack* | *Button* | *Kembali* | Kembali ke tampilan kelola tanah |

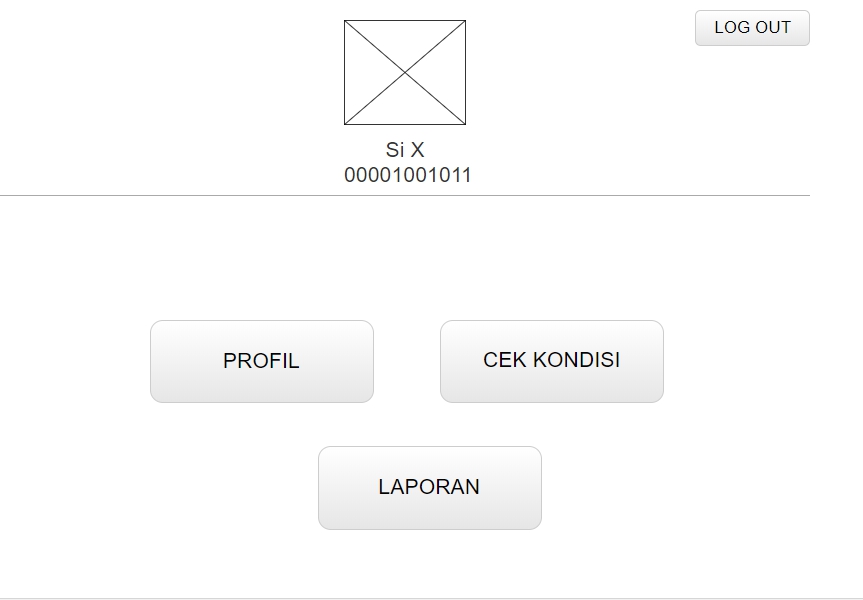
## 3.6.17



*Antarmuka : {Tampilan Mutasi}*

| **Id\_Objek** | **Jenis** | **Nama** | **Keterangan** |
| --- | --- | --- | --- |
| *idMutasi* | *Textfield* | *ID Mutasi* | Textfield untuk id mutasi |
| *dari* | *Dropdown* | *Dari* | Dropdown untuk meutasi inventaris |
| *ke* | *dropdown* | *Ke* | Dropdown untuk meutasi inventaris |
| *btnConfirm* | *Button* | *Confirm* | Mengkonfirmasi barang atu tanah yang akan di mutasi |
| *btnBack* | *Button* | *Kembali* | Kembali ke tampilan kelola barang |

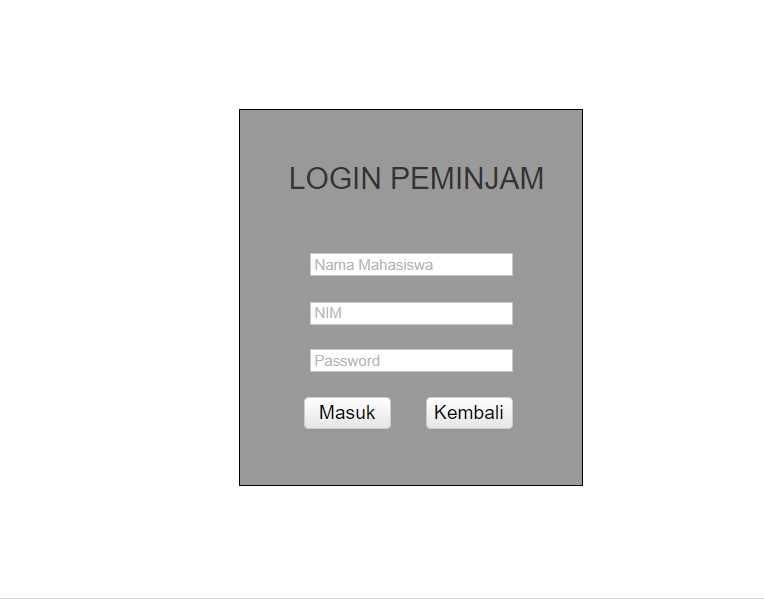
## 3.6.18



*Antarmuka : {Tampilan menu karyawan}*

| **Id\_Objek** | **Jenis** | **Nama** | **Keterangan** |
| --- | --- | --- | --- |
| *BtnProfil* | *Button* | *PROFIL* | Button untuk masuk ke profil ---- |
| *BtnCekKondisi* | *Button* | *CEK KONDISI* | Button untuk cek kondisi |
| *BtnLaporan* | *Button* | *LAPORAN* | Button untuk laporan |

## 3.6.19



*Antarmuka : {Tampilan Login Peminjam}*

| **Id\_Objek** | **Jenis** | **Nama** | **Keterangan** |
| --- | --- | --- | --- |
| *InputNama* | *Textfield* | *Nama Mahasiswa* | Textfield untuk input nama mahasiswa |
| *InputNIM* | *Textfield* | *NIM* | Textfield untuk input NIM mahasiswa |
| *btnMasuk* | *Button* | *Masuk* | Button untuk masuk ke halaman pemijam |
| *btnBack* | *Button* | *Kembali* | Kembali ke tampilan login peminjam / sebelumnya |

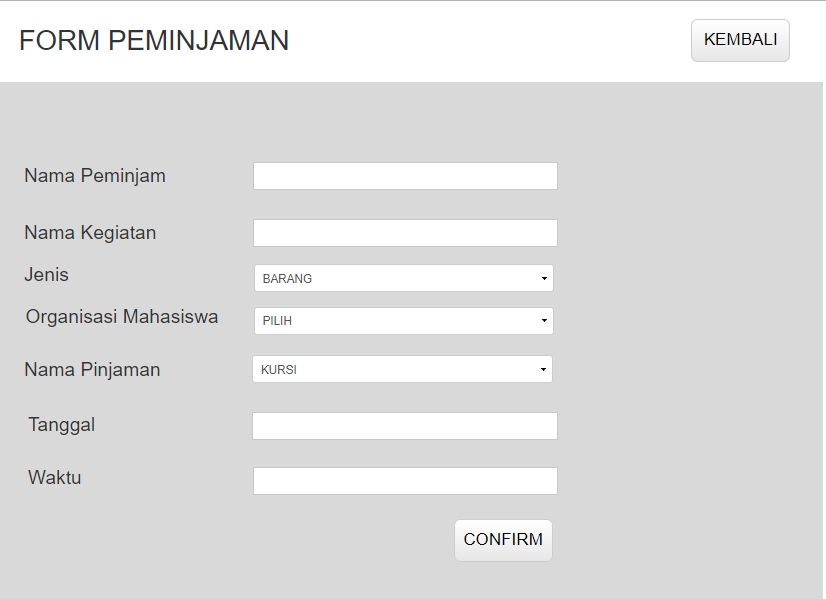
## 3.6.20



*Antarmuka : {Tampilan Login Peminjam}*

| **Id\_Objek** | **Jenis** | **Nama** | **Keterangan** |
| --- | --- | --- | --- |
| *BtnProfil* | *Button* | *PROFIL* | Button untuk masuk ke profil peminjam |
| *BtnFormPeminjaman* | *Button* | *FORM PEMINJAMAN* | Button untuk masuk ke form peminjaman |
| *BtnJadwalPeminjaman* | *Button* | *JADWAL PEMINJAMAN* | Button untuk masuk ke jadwal peminjaman |

## 3.6.21



*Antarmuka : {Tampilan Form Peminjaman}*

| **Id\_Objek** | **Jenis** | **Nama** | **Keterangan** |
| --- | --- | --- | --- |
| *namaPeminjam* | *Textfield* | *Nama Peminjam* | Textfield untuk input nama mahasiswa |
| *namaKegiatan* | *Textfield* | *Nama Kegiatan* | Textfield untuk input NIM mahasiswa |
| *jenis* |  | *Jenis* | Jenis barang barang yang akan di pinjam |
| *organisasiMahasiswa* |  | *Organisasi Mahasiswa* | Memilih organisasi mahasiswa sebagai tanda bahwa organisasi ini akan meminjam |
| *namaPeminjam* |  | *Nama Peminjam* | Mengisi barang yang akan di pinjam |
| *tanggal* | *Textfield* | *Tanggal* | Textfield untuk input tanggal peminjaman |
| *waktu* | *Textfield* | *Waktu* | Textfield untuk input waktu peminjaman |
| *btnConfirm* | *Button* | *Confirm* | Button untuk masuk ke halaman pemijam |
| *btnBack* | *Button* | *Kembali* | Kembali ke tampilan login peminjam / sebelumnya |

## 3.6.22



*Antarmuka : {Tampilan Jadwal Peminjaman }*

| **Id\_Objek** | **Jenis** | **Nama** | **Keterangan** |
| --- | --- | --- | --- |
| *BtnBarang* | *Button* | *BARANG* | Button untuk menampilkan data peminjaman barang |
| *BtnTanah* | *Button* | *TANAH* | Button untuk menampilkan data peminjaman tanah |
| *Table* | *Table* | *-* | Isinya menampilkan data jadwal peminjaman |
| *btnBack* | *Button* | *Kembali* | Kembali ke tampilan login peminjam |

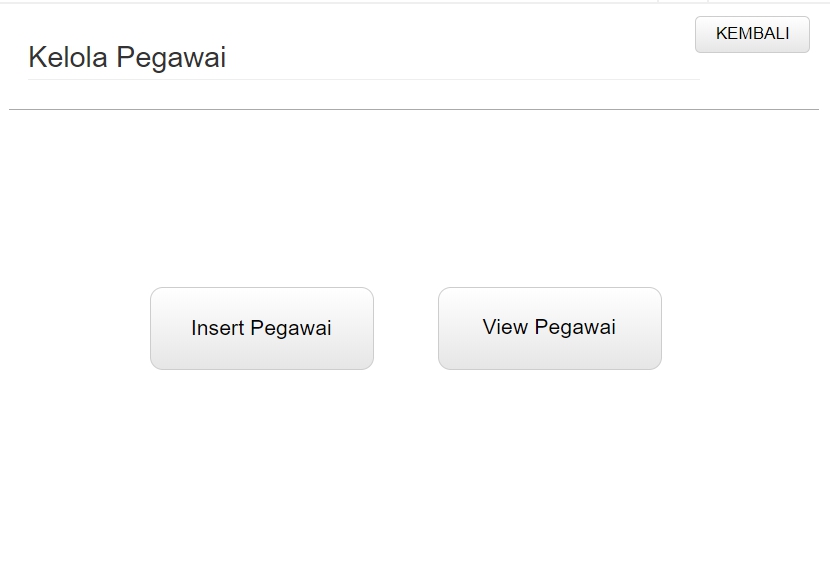
## 3.6.23



*Antarmuka : {Tampilan Login Kelola Pegawai}*

| **Id\_Objek** | **Jenis** | **Nama** | **Keterangan** |
| --- | --- | --- | --- |
| *BtnKelolaPegawai* | *Button* | *KELOLA PEGAWAI* | Button untuk masuk ke kelola pegawai |

## 3.6.24



*Antarmuka : {Tampilan Keloal Pegawai}*

| **Id\_Objek** | **Jenis** | **Nama** | **Keterangan** |
| --- | --- | --- | --- |
| *BtnInsertPegawai* | *Button* | *Insert Pegawai* | Button untuk mauk ke insert pegawai |
| *BtnViewPegawai* | *Button* | *View Pegawai* | Button untuk masuk ke view pegawai |
| *btnBack* | *Button* | *Kembali* | Kembali ke tampilan kelola pegawai |

## 3.6.25



*Antarmuka : {Tampilan Insert Pegawai}*

| **Id\_Objek** | **Jenis** | **Nama** | **Keterangan** |
| --- | --- | --- | --- |
| *InputNamaPegawai* | *Textfield* | *Nama Pegawai* | Textfield untuk input nama pegawai |
| *InputPassword* | *Textfield* | *Password* | Textfield untuk input password |
| *jabatan* | *dropdown* | *Jabatan* | Dropdown untuk isi jabatan pegawai |
| *btnConfirm* | *Button* | *Cofirm* | Mengkonfirmasi data tanah yang telah di insert |
| *btnBack* | *Button* | *Kembali* | Kembali ke tampilan kelola pegawai |

## Perancangan Representasi Persistensi Kelas

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

# Matriks Kerunutan

|  |  |
| --- | --- |
| **Kelas** | **Use Case Terkait** |
| Admin | Kelola pegawai |
| cs | Kelola barang, kelola tanah, mutasi perubahan |
| Logistic | Laporan tutup tahun |
| Karyawan | Kondisi barang |
| Peminjam | Pinjaman |

# 