SOFTWARE DESIGN DOCUMENT (SDD)

PENGARSIPAN SKRIPSI

Kelompok 1

Kelas DSE - B

Anggota Kelompok :

Adhani Mulianti (3411201085)

Rayi Syifa Adriana F (3411201089)

Adisti Dwi Susanti (3411201090)

Alfin Gerliandeva (3411201094)

…

# **TABLE OF CONTENTS**

[**TABLE OF CONTENTS** i](#_Toc122183325)

[**TABLE OF PICTURES** ii](#_Toc122183326)

[**BAB I Introduction** 1](#_Toc122183327)

[**1.1 Purpose** 1](#_Toc122183328)

[**1.2 Scope** 2](#_Toc122183329)

[**1.3 Overview** 6](#_Toc122183330)

[**1.4 Reference** 6](#_Toc122183331)

[**1.5 Definitions and Acronyms** 7](#_Toc122183332)

[**BAB II System Overview** 9](#_Toc122183333)

[**2.1 Definition** 9](#_Toc122183334)

[**2.2 Function** 9](#_Toc122183335)

[**2.3 Feature** 10](#_Toc122183336)

[**BAB III Application Design** 11](#_Toc122183337)

[**3.1 Use Case Diagram** 11](#_Toc122183338)

[**3.2 Use Case Scenario** 11](#_Toc122183339)

[**3.3 Class Diagram** 17](#_Toc122183340)

[**3.4 Sequence Diagram** 19](#_Toc122183341)

[**3.7 Deployment Diagram** 23](#_Toc122183342)

[**BAB IV Data Design** 25](#_Toc122183343)

[**4.1 Logical & Physical Design** 25](#_Toc122183344)

[**BAB V User Interface Design** 28](#_Toc122183345)

[**5.1 Graphical User Interface** 28](#_Toc122183346)

# **TABLE OF PICTURES**

[Gambar 1 Usecase Diagram Aplikasi Pengarsipan Skripsi 11](#_Toc122183295)

[Gambar 2 Sequence Diagram Login Petugas TU 19](#_Toc122183296)

[Gambar 3 Sequence Diagram Data Mahasiswa 20](#_Toc122183297)

[Gambar 4 Sequence Diagram Data Dosen 21](#_Toc122183298)

[Gambar 5 Sequence Diagram Data Kategori Skripsi 22](#_Toc122183299)

[Gambar 6 Sequence Diagram Data Skripsi 22](#_Toc122183300)

[Gambar 7 Sequence Diagram Pencarian Skripsi 23](#_Toc122183301)

[Gambar 8 Deployment Diagram Aplikasi Pengarsipan Skripsi 24](#_Toc122183302)

# **BAB I** **Introduction**

## **1.1 Purpose**

**Latar Belakang**

Menurut IEEE SDD atau Software Design Document adalah sebuah desain dokumentasi dari perancangan perangkat lunak yang menyediakan informasi - informasi, dari mulai planning, analisis, desain interface, implementasi dan juga decision making. Lalu Software Design Document juga dapat disebut sebagai dokumen informasi yang memuat segala hal tentang cetak biru atau blueprint dari sebuah model sistem yang akan dirancang atau dikembangkan.

SDD atau Software Design Document harus dapat diselaraskan dengan setiap tugas - tugas, yaitu diantaranya komunikasi antar tugas, struktur data dan juga menjelaskan sistem model basis datanya. Kemudian pada dokumen SDD ini perlu memuat informasi yang dapat menjelaskan jenis aplikasi atau perangkat lunak yang akan dibangun dan juga pendekatan yang digunakan dalam pengembangan dan pengelolaannya.

Arsip adalah kumpulan dari berbagai informasi yang sangat penting untuk menunjang proses administrasi dan pengelolaan di sebuah lembaga atau pun instansi pendidikan. Pengarsipan data menjadi tidak kalah penting, karena diperlukannya proses pengarsipan data yang mudah, cepat akurat dan juga efektif.

Aplikasi Pengarsipan Skripsi sangatlah penting bagi jurusan informatika dalam proses penyajian dan pengolahan data. Aplikasi ini untuk membantu dan memudahkan mahasiswa untuk mencari arsip skripsi dengan cara yang lebih efisien.

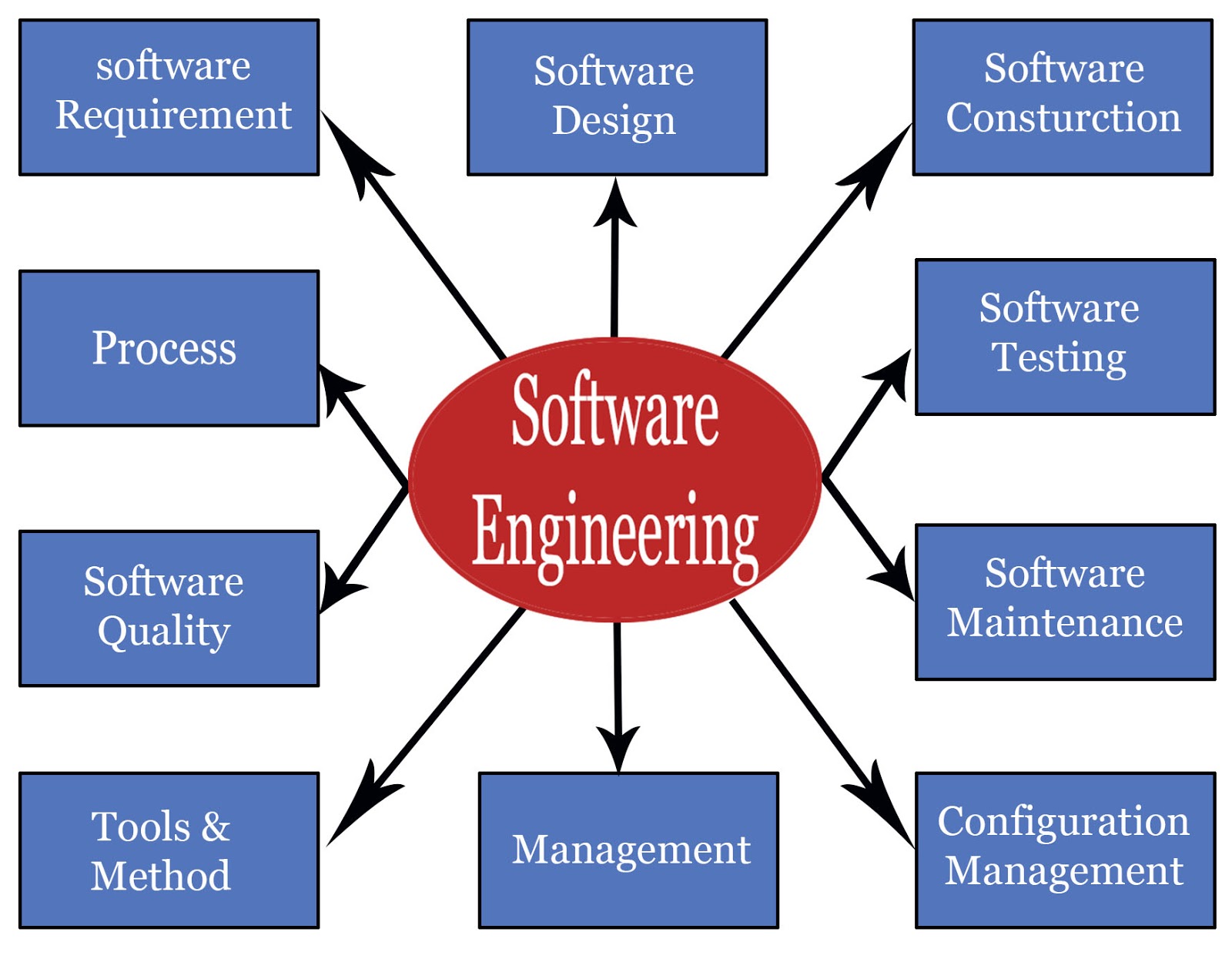
**Tujuan**

Untuk tujuan dibuatnya dokumen Software Design Document (SDD) ini adalah untuk membuat sebuah aplikasi yang dibuat khusus untuk staff tata usaha jurusan informatika unjani agar mempermudah pekerjaannya dalam proses pengarsipan data skripsi mahasiswa, dan menambah pengetahuan staff mengenai penerapan sistem informasi, serta menyediakan informasi yang cepat, tepat dan akurat pada data atau laporan yang sedang dibutuhkan.

## **1.2 Scope**

**Ruang Lingkup**

Ruang Lingkup perangkat lunak adalah meliputi penjelasan mengenai kebutuhan, desain, konstruksi, testing, pemeliharaan, manajemen konfigurasi, rekayasa alat - alat dan metode - metode, kualitas dan proses rekayasa dari sebuah perangkat lunak. Untuk penjelasan masing - masing dari ruang lingkup tersebut akan dijelaskan di bawah ini:



1. Software Requirements

Yaitu berhubungan dengan spesifikasi detail dari analisis kebutuhan (fungsional maupun non-fungsional) dan persyaratan sebuah perangkat lunak.

1. Software Design

Yaitu wilayah yang mencakup bagaimana proses dalam arsitektur perangkat lunak, komponen perangkat lunak, interface atau tampilan GUI, dan juga segala hal yang berkaitan dengan karakteristik dari ciri perangkat lunak.

1. Software Construction

Yaitu yang berhubungan dengan rincian dari detail pembangunan perangkat lunak, contohnya seperti algoritma, coding, testing, pencarian bugs atau kesalahan program dan lain sebagainya.

1. Software Testing

Yaitu yang berhubungan dengan pengujian perangkat lunak, apakah sudah sesuai dengan hasil dari analisis kebutuhan, kebutuhan user, dan planning awal dari perancangan suatu perangkat lunak tersebut.

1. Software Maintenance

Yaitu wilayah yang mencakup tentang usaha - usaha atau upaya dalam perawatan dan atau pemeliharaan perangkat lunak, diantaranya memperbaiki bugs, beradaptasi dengan teknologi baru, dan melakukan penyempurnaan aplikasi atau perangkat lunaknya.

1. Software Configuration

Yaitu yang berhubungan dengan upaya dalam melakukan perubahan dari konfigurasi atau pengaturan tentang perangkat lunak untuk memenuhi kebutuhan dari hasil analisis kebutuhan perangkat lunaknya.

1. Software Engineering Management

Yaitu hal - hal yang berkaitan tentang manajemen dan juga measurement dalam perancangan perangkat lunak dan juga termasuk dalam perencanaan dari proyek pembangunan dan atau pengembangan perangkat lunak.

1. Software Engineering Tools and Methods

Yaitu wilayah yang mencakup tentang segala informasi dari kajian atau teori dasar tentang alat bantu dan juga metode dari perancangan perangkat lunaknya.

1. Software Engineering Process

Yaitu yang berhubungan dengan pengertian atau definisi, kemudian juga implementasi, measurement, manajemen, dan perubahan, perbaikan dari sebuah proses panjang pengembangan perangkat lunak.

1. Software Quality

Yaitu hal - hal yang berkaitan tentang pada kualitas sebuah perangkat lunak yang akan dibangun atau dikembangkan.

**Tujuan**

Untuk tujuan dari lingkup perancangan perangkat lunak, yaitu bagaimana memaksimalkan hasil kinerja yang bagus dan baik dan tepat sasaran terhadap pengguna(robust), tetapi dengan biaya sumber daya yang rendah atau cukup efisien. Kemudian untuk penjelasan singkat masing - masing tujuan dari perancangan perangkat lunak yaitu sebagai berikut ini:

1. Mendapatkan hasil dari perangkat lunak dengan biaya pemeliharan atau perawatan dengan serendah mungkin (efisien).
2. Mendapatkan hasil dari perangkat lunak yang dapat bekerja di berbagai perangkat, sistem operasi dan juga multiplatform( interoperability).
3. Memperoleh biaya produksi dalam pengembangan perangkat lunak yang rendah (efisien).
4. Memperoleh hasil dari perangkat lunak dengan kinerja yang bagus dan baik serta andal dan tepat waktu(reliability).

**Manfaat**

Untuk manfaat dari perancangan perangkat lunak sendiri yaitu ada dari segi efisiensi yang baik dan juga problem solving dengan menggunakan program atau aplikasi perangkat lunak. Kemudian juga pada perangkat lunak yang berkualitas tinggi, maka akan selalu dapat dikembangkan programnya menjadi lebih baik dan atau sebagai pembelajaran untuk generasi selanjutnya. Kemudian untuk penjelasan singkat tentang masing - masing manfaat dari perancangan perangkat lunak akan dijelaskan di bawah ini:

1. Heavy Demand

Dengan perkembangan perangkat lunak hingga saat ini, maka dengan banyak pengembang - pengembang independent ataupun perusahaan dan juga tenaga programmer, desainer, software analyst dan project manager. Maka permintaan untuk tawaran pekerjaan pada bidang ini semakin tinggi dan juga dibutuhkan.

1. Flexible Schedules

Karena software engineer dapat bekerja dimana saja, maka jadwal dalam pengerjaan suatu proyek pengembangan lunak pun menjadi fleksibel dan dapat menyesuaikan dengan jadwal para tenaga kerja software engineer.

1. Higher Quality Programs

Dengan semakin berkembangnya perangkat lunak, maka hal - hal yang dapat dipelajari oleh software engineer semakin banyak pula, sehingga dalam merancang perangkat lunak dapat ditingkatkan ke sistem yang lebih berkualitas lagi dengan semakin banyak pengalaman dan juga dapat menghindari kesalahan - kesalahan umum sehingga dapat lebih baik dari segi kualitas program atau perangkat lunak.

1. Better Job Prospects

Karena ilmu atau bidang software engineer semakin dibutuhkan dan juga diminati sering dengan banyak perusahaan start-up, independent dan lain sebagainya. Maka permintaan tentang tenaga kerja dari software engineer khususnya akan semakin banyak dan merupakan salah satu hal dari prospek kerja yang lebih baik. Dan semua itu diharapkan dengan hasil dari perancangan perangkat lunak yang semakin baik dan mengurangi kegagalan dari sebuah proyek pengembangan perangkat lunaknya.

## **1.3 Overview**

Dokumen Software Design Document ini memuat sebuah informasi mengenai garis besar dari perancangan sistem pengarsipan skripsi ini, kemudian juga terdapat penjelasan mengenai application design, mulai dari use case, use case scenario, class, sequence, activity, state, dan deployment diagram. Kemudian juga ada penjelasan tentang data design, terbagi 2 yaitu logical design dan physical design. Yang terakhir yaitu ada kebutuhan mengenai antarmuka pengguna, diantaranya yaitu mulai dari user interface, hardware interface, software interface dan atau communication interface.

## **1.4 Reference**

|  |  |
| --- | --- |
| [1] | Admin, "Library," Meriam Webster, 5 December 2011. [Online]. Available: https://www.merriam-webster.com/dictionary/library. [Accessed 3 July 2022]. |
| [2] | D. Thakur, "SDD," Computers Note, 30 November 2013. [Online]. Available: https://ecomputernotes.com/software-engineering/softwaredesigndocumentation. [Accessed 3 July 2022]. |
| [3] | Admin, "UNS," UNS, 26 August 2016. [Online]. Available: https://eprints.uns.ac.id/8167/1/218580811201104171.pdf. [Accessed 3 July 2022]. |
| [4] | m. "Advantages and Disadvantages of Software Engineering," myAyan, 28 September 2021. [Online]. Available: https://www.myayan.com/advantages-and-disadvantages-of-software-engineering. [Accessed 3 July 2022]. |
| [5] | Admin, "Informasi Perpustakaan," Pustaka UNJANI, 4 February 2017. [Online]. Available: http://pustaka.unjani.ac.id/index.php?p=libinfo. [Accessed 3 July 2022]. |
| [6] | Admin, "UML Diagrams," UML Diagrams Organization, 20 June 2021. [Online]. Available: https://www.uml-diagrams.org/. [Accessed 3 July 2022]. |
| [7] | Admin, "User Interface Design," Intercation Design Foundation, 2 March 2020. [Online]. Available: https://www.interaction-design.org/literature/topics/ui-design. [Accessed 3 July 2022]. |
| [8] | Admin, "interface," PCMAG, 4 April 2019. [Online]. Available: https://www.pcmag.com/encyclopedia/term/interface. [Accessed 3 July 2022]. |

## **1.5 Definitions and Acronyms**

**Definisi :**

1. Software Design Document adalah sebuah dokumen yang memuat spesifikasi teknis dari analisis dan desain perangkat lunak yang menjelaskan desain perancangan perangkat lunak dan keseluruhan arsitektur rancangan sistem perangkat lunak.
2. Unified Modeling Language adalah bahasa yang digunakan pada arsitek dan pengembangan perangkat lunak, yang dipakai untuk menjelaskan, menggambarkan, mendokumentasikan, merancang dalam struktur dan artefak sebuah perangkat lunak.
3. Rekayasa Perangkat Lunak adalah sebuah disiplin ilmu atau bidang profesi yang mendalami dalam pembuatan, perancangan, desain, pemeliharaan, analisis dan manajemen organisasi pembangunan perangkat lunak serta manajemen kualitasnya.
4. Use Case Diagram adalah salah satu behaviour diagram yang menjelaskan tentang hubungan serangkaian use case yang dapat dilakukan bersama dalam sistem, kemudian terhubung terhadap aktor atau user.
5. Class Diagram adalah salah satu struktur UML dimana menjelaskan dalam struktur sistem / aplikasi yang dirancang pada level kelas dan atau antarmuka, dan class diagram lebih dekat dengan pengkodean atau coding.
6. Sequence Diagram adalah salah satu interaction diagram yang menjelaskan sequence atau urutan dari messages dan pertukaran antar messages pada beberapa lifelines.
7. Activity Diagram adalah salah satu behavior diagram, dimana menunjukkan aliran aktivitas dan juga aliran kontrol dalam urutan aktivitas hingga menuju ke final statenya.
8. State Diagram adalah adalah salah satu behavior diagram, dimana menjelaskan aliran state/kondisi dimana dalam urutan state atau perubahan kondisi pada sistem perancangannya.
9. Deployment Diagram adalah salah satu structure diagram yang menampilkan arsitektur sistem antara hardware dan juga software dan dihubungkan oleh sebuah relasi.
10. Entity Relationship Diagram adalah sebuah model yang digunakan untuk merepresentasikan object, people, konsep dengan menggunakan entitas, relasi dan juga atribut dalam perancangan perangkat lunak.
11. User Interface adalah proses mendesain untuk membuat atau mengembangkan sebuah antarmuka pada sebuah perangkat lunak, aplikasi atau program.

**Akronim :**

1. PL = Perangkat Lunak
2. UML = Unified Modeling Language
3. SDD = Software Design Document
4. RPL = Rekayasa Perangkat Lunak
5. UC = Use Case Diagram
6. CD = Class Diagram
7. SD = Sequence Diagram
8. DEP = Deployment Diagram
9. ERD = Entity Relationship Diagram

# **BAB II** **System Overview**

## **2.1 Definition**

Perangkat lunak atau peranti lunak adalah istilah khusus untuk data yang diformat, dan disimpan secara digital, termasuk program komputer, dokumentasinya, dan berbagai informasi yang bisa dibaca, dan ditulis oleh komputer. Arsip merupakan berbagai bentuk dokumen bersejarah dalam format atau fasilitas fisik tempat mereka disimpan. Arsip ini berisi berbagai informasi yang terakumulasi selama masa hidup sebuah organisasi atau individu, kemudian disimpan untuk menunjukkan ke orang lain fungsi organisasi atau individu tersebut.

## **2.2 Function**

Fungsi dari perangkat lunak adalah untuk mengatur hardware/perangkat keras yang ada pada komputer. Dengan begitu, komputer yang digunakan dapat bekerja dengan baik sesuai pekerjaan yang dibutuhkan. Software/ perangkat lunak juga dapat digunakan sebagai penghubung antara beberapa software yang lain dengan hardware komputer. Kemudian ada fungsi dari perancangan aplikasi pengarsipan skripsi berbasis java adalah sebagai berikut ini:

1. Aplikasi pengarsipan skripsi ini dapat memudahkan dalam pencarian informasi data skripsi mahasiswa.
2. Aplikasi pengarsipan skripsi ini dapat memudahkan proses pengolahan, penyajian data skripsi mahasiswa.
3. Aplikasi pengarsipan skripsi ini dapat membuat pengarsipan data skripsi menjadi terstruktur.

## 

## **2.3 Feature**

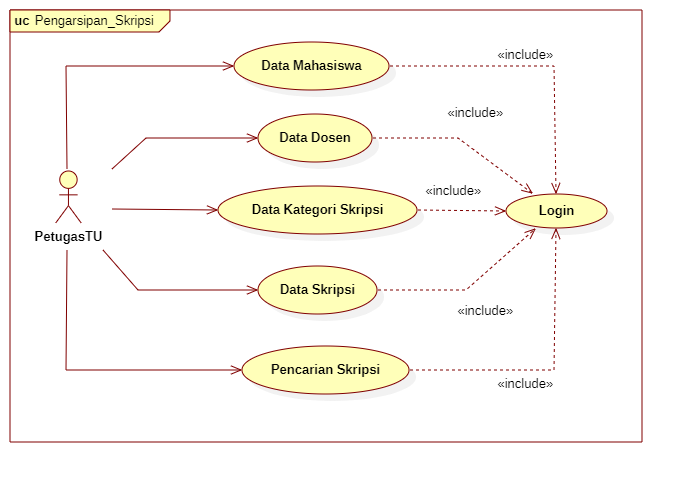
Adapun fitur - fitur yang dikembangkan pada aplikasi pengarsipan skripsi di antaranya adalah sebagai berikut ini:

1. Main Menu Pengarsipan Skripsi sistem aplikasi pengarsipan skripsi ini memiliki tampilan data mahasiswa, data dosen, data skripsi, data kategori skripsi, dan data pencarian skripsi.
2. Menu Form Login Petugas TU untuk masuk ke aplikasi pengarsipan skripsi.
3. Menu Form Data Mahasiswa untuk menyimpan, menghapus dan mengubah data mahasiswa.
4. Menu Form Data Dosen untuk menyimpan, menghapus dan mengubah data dosen.
5. Menu Form Data Skripsi untuk menyimpan, menghapus dan mengubah data skripsi.
6. Menu Form Kategori Skripsi untuk menyimpan, menghapus dan mengubah data kategori skripsi.
7. Menu Pencarian Skripsi untuk mencari skripsi berdasarkan id skripsi, judul skripsi, kategori studi, nama mahasiswa dan nama dosen.

# **BAB III** **Application Design**

## **3.1 Use Case Diagram**

Use Case Diagram dipakai untuk menyediakan atau menyajikan sebuah interaksi antara use case dan juga aktor. Kemudian Use Case diagram ini memiliki fungsi untuk menjelaskan bagaimana detail dari interaksi use case serta aktor yang dihubungkan oleh relasi. Lalu untuk penggambaran atau ilustrasi dari use case diagram perancangan aplikasi pengarsipan skripsi ini dapat dilihat pada gambar di bawah ini:



Gambar 1 Usecase Diagram Aplikasi Pengarsipan Skripsi

## **3.2 Use Case Scenario**

Berikut adalah penjelasan dari skenario masing - masing use case diagram aplikasi pengarsipan skripsi:

**USE CASE : LOGIN**

Tabel 1 Flow of Event Usecase Login

|  |  |  |
| --- | --- | --- |
| *Use case Name* | Login | |
| *Brief Description* | Use case ini mengatur proses login user | |
| *Primary Actor* | PetugasTU | |
| *Secondary Actor* | - | |
| *Pre-Condition* | - | |
| *Post-Condition* | User masuk ke dalam sistem | |
| *Included Use case* | - | |
| *Basic Flow of Events* | *Actor’s Action* | *Sistem’s Response* |
| 1 .User memasukkan username dan password kemudian mengklik tombol login. | 1. Sistem mengecek username dan password apakah sudah benar dengan cara mengambil data sesuai username dan password yang dimasukkan user dari database yang tersimpan pada database. 2. Jika username dan password benar, sistem menampilkan form main menu. |
| *Alternate Flow of*  *Events* | 3a. Jika username dan password salah, maka sistem akan  menampilkan informasi bahwa username dan password salah. | |
| *Extension Points* | - | |

**USE CASE : DATA MAHASISWA**

Tabel 2 Flow of Event Usecase Data Mahasiswa

|  |  |  |
| --- | --- | --- |
| *Use case Name* | Data Mahasiswa | |
| *Brief Description* | Use case ini mengatur proses tambah data, edit data, hapus data pada tabel t\_data\_mahasiswa. | |
| *Primary Actor* | PetugasTU | |
| *Secondary Actor* | - | |
| *Pre-Condition* | PetugasTU telah login ke dalam aplikasi. | |
| *Post-Condition* | Data Mahasiswa tersimpan atau terubah dan atau terhapus dalam aplikasi. | |
| *Included Use case* | - | |
| *Basic Flow of Events* | *Actor’s Action* | *Sistem’s Response* |
| 1 .PetugasTU memilih  menu data mahasiswa  3. PetugasTU  melakukan tambah, edit, hapus data dalam menu data mahasiswa. | 2 . Sistem menampilkan form menu data mahasiswa.  4. Sistem berhasil menyimpan atau mengubah dan atau menghapus data mahasiswa pada database.  5. Setelah sistem berhasil menyimpan atau mengubah dan atau menghapus data mahasiswa, maka data akan tersimpan atau terubah dan atau terhapus pada tabel t\_data\_mahasiswa. |
| *Alternate Flow of*  *Events* | 5a. Jika sistem tidak berhasil menyimpan atau mengubah dan atau menghapus data mahasiswa, maka sistem akan menampilkan pesan pop-up pesan kesalahan input. | |
| *Extension Points* | - | |

**USE CASE : DATA DOSEN**

Tabel 3 Flow of Event Usecase Data Dosen

|  |  |  |
| --- | --- | --- |
| *Use case Name* | Data Dosen | |
| *Brief Description* | Use case ini mengatur proses tambah data, edit data, hapus data pada tabel t\_data\_dosen. | |
| *Primary Actor* | PetugasTU | |
| *Secondary Actor* | - | |
| *Pre-Condition* | PetugasTU telah login ke dalam aplikasi. | |
| *Post-Condition* | Data Dosen tersimpan atau terubah dan atau terhapus dalam aplikasi. | |
| *Included Use case* | - | |
| *Basic Flow of Events* | *Actor’s Action* | *Sistem’s Response* |
| 1 .PetugasTU memilih  menu data dosen.  3. PetugasTU  melakukan tambah, edit, hapus data dalam menu data dosen. | 2 . Sistem menampilkan form menu data dosen.  4. Sistem berhasil menyimpan atau mengubah dan atau menghapus data dosen pada database.  5. Setelah sistem berhasil menyimpan atau mengubah dan atau menghapus data dosen, maka data akan tersimpan atau terubah dan atau terhapus pada tabel t\_data\_dosen. |
| *Alternate Flow of*  *Events* | 5a. Jika sistem tidak berhasil menyimpan atau mengubah dan atau menghapus data dosen, maka sistem akan menampilkan pesan pop-up pesan kesalahan input. | |
| *Extension Points* | - | |

**USE CASE : DATA KATEGORI SKRIPSI**

Tabel 4 Flow of Event Usecase Data Kategori Skripsi

|  |  |  |
| --- | --- | --- |
| *Use case Name* | Data Kategori Skripsi | |
| *Brief Description* | Use case ini mengatur proses tambah data, edit data, hapus data pada tabel t\_kategori\_skripsi. | |
| *Primary Actor* | PetugasTU | |
| *Secondary Actor* | - | |
| *Pre-Condition* | PetugasTU telah login ke dalam aplikasi. | |
| *Post-Condition* | Data Kategori Skripsi tersimpan atau terubah dan atau terhapus dalam aplikasi. | |
| *Included Use case* | - | |
| *Basic Flow of Events* | *Actor’s Action* | *Sistem’s Response* |
| 1 .PetugasTU memilih  menu data kategori skripsi.  3. PetugasTU  melakukan tambah, edit, hapus data dalam menu data kategori skripsi. | 2 . Sistem menampilkan form menu data kategori skripsi.  4. Sistem berhasil menyimpan atau mengubah dan atau menghapus data kategori skripsi pada database.  5. Setelah sistem berhasil menyimpan atau mengubah dan atau menghapus data kategori skripsi, maka data akan tersimpan atau terubah dan atau terhapus pada tabel t\_kategori\_skripsi. |
| *Alternate Flow of*  *Events* | 5a. Jika sistem tidak berhasil menyimpan atau mengubah dan atau menghapus data kategori skripsi, maka sistem akan menampilkan pesan pop-up pesan kesalahan input. | |
| *Extension Points* | - | |

**USE CASE : DATA SKRIPSI**

Tabel 5 Flow of Event Usecase Data Skripsi

|  |  |  |
| --- | --- | --- |
| *Use case Name* | Data Skripsi | |
| *Brief Description* | Use case ini mengatur proses tambah data, edit data, hapus data pada tabel t\_data\_skripsi. | |
| *Primary Actor* | PetugasTU | |
| *Secondary Actor* | - | |
| *Pre-Condition* | PetugasTU telah login ke dalam aplikasi. | |
| *Post-Condition* | Data Skripsi tersimpan atau terubah dan atau terhapus dalam aplikasi. | |
| *Included Use case* | - | |
| *Basic Flow of Events* | *Actor’s Action* | *Sistem’s Response* |
| 1 .PetugasTU memilih  menu data skripsi.  3. PetugasTU  melakukan tambah, edit, hapus data dalam menu data skripsi. | 2 . Sistem menampilkan form menu data skripsi.  4. Sistem berhasil menyimpan atau mengubah dan atau menghapus data skripsi pada database.  5. Setelah sistem berhasil menyimpan atau mengubah dan atau menghapus data skripsi, maka data akan tersimpan atau terubah dan atau terhapus pada tabel t\_data\_skripsi. |
| *Alternate Flow of*  *Events* | 5a. Jika sistem tidak berhasil menyimpan atau mengubah dan atau menghapus data skripsi, maka sistem akan menampilkan pesan pop-up pesan kesalahan input. | |
| *Extension Points* | - | |

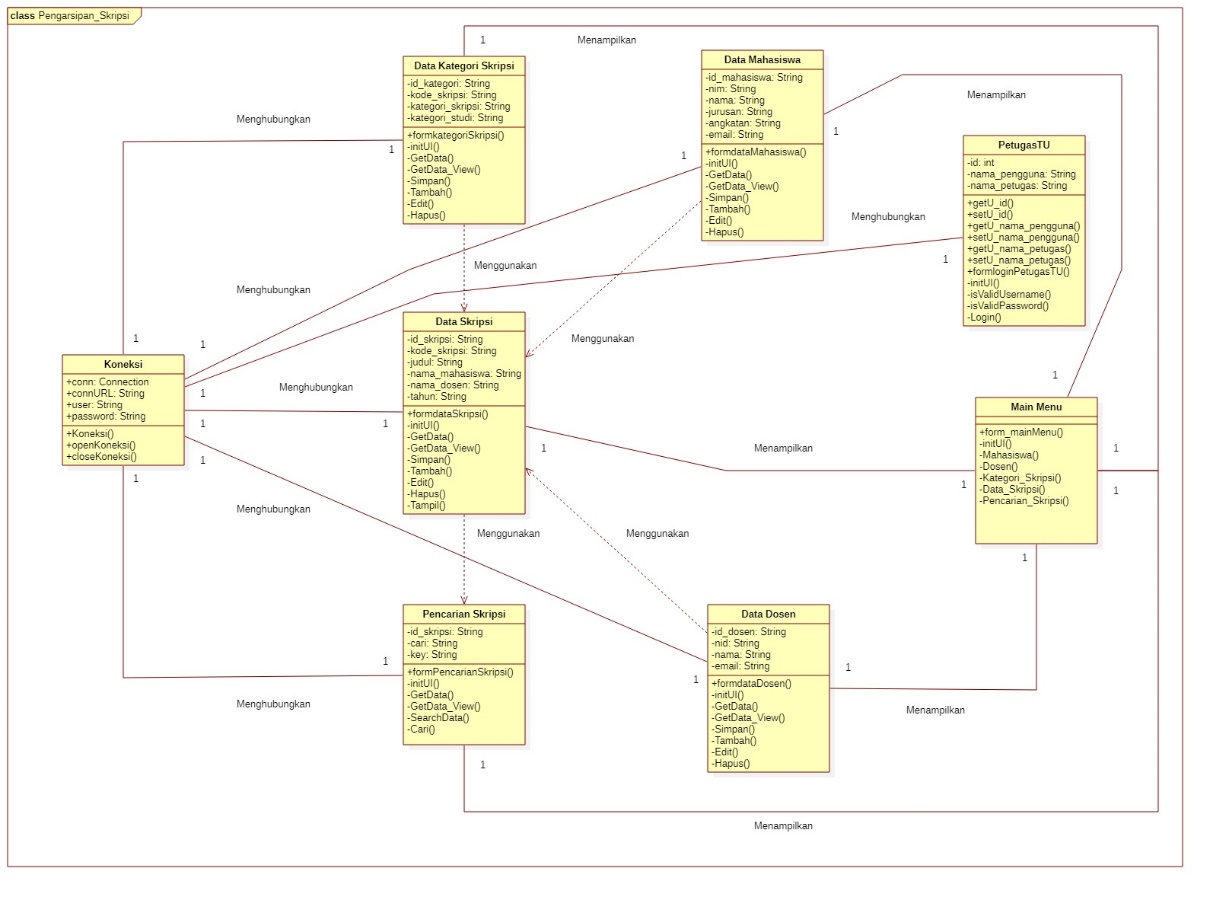
**USE CASE : PENCARIAN SKRIPSI**

Tabel 6 Flow of Event Usecase Pencarian Skripsi

|  |  |  |
| --- | --- | --- |
| *Use case Name* | Pencarian Skripsi | |
| *Brief Description* | Use case ini mengatur proses pencarian data – data skripsi melalui id\_skripsi, judul, kategori\_studi, nama mahasiswa, nama dosen. | |
| *Primary Actor* | PetugasTU | |
| *Secondary Actor* | - | |
| *Pre-Condition* | PetugasTU telah login ke dalam aplikasi. | |
| *Post-Condition* | Hasilnya yaitu menemukan data skripsi berdasarkan id\_skripsi, judul, kategori\_studi, nama mahasiswa, nama dosen. | |
| *Included Use case* | - | |
| *Basic Flow of Events* | *Actor’s Action* | *Sistem’s Response* |
| 1 .PetugasTU memilih  menu pencarian skripsi.  3. PetugasTU  melakukan pencarian data berdasarkan id\_skripsi, judul, kategori\_studi, nama mahasiswa, nama dosen. | 2 . Sistem menampilkan menu pencarian skripsi.  4. Sistem berhasil menemukan data – data skripsi yang telah dicari berdasarkan id\_skripsi, judul, kategori\_studi, nama mahasiswa, nama dosen. |
| *Alternate Flow of*  *Events* | - | |
| *Extension Points* | - | |

## **3.3 Class Diagram**

Class diagram digunakan untuk mengarahkan sebuah interaksi antar kelas - kelas di dalam sistem perancangan aplikasi pengarsipan skripsi Kemudian Class sendiri itu ada 3 bagian utama, yang pertama ialah Name Class, kedua yaitu ada Attribute dan yang ketiga yaitu ada Behavior atau Method. Selanjutnya suatu Class diagram juga mewakili setiap objek - objek pada diagram yang mewakili paradigma dari konsep OOP itu sendiri. Untuk gambar Class diagram yang dibuat dapat dilihat pada gambar di bawah ini:



Fungsi dari masing – masing class yang ada pada Aplikasi Pengarsipan Skripsi:

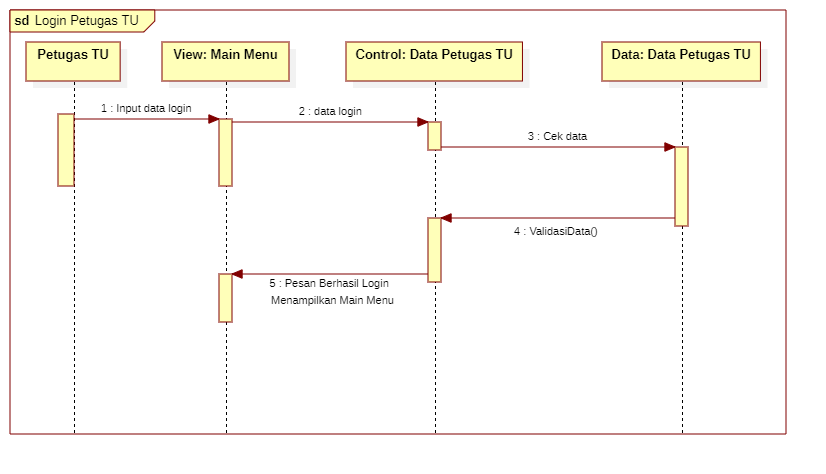
1. Koneksi: Class Koneksi yaitu memiliki fungsi untuk menghubungkan semua form menu dengan database pengarsipan skripsi.
2. PetugasTU: Class PetugasTU yaitu memiliki fungsi untuk mengambil dan mengatur id, nama pengguna dan nama petugas TU. Kemudian juga yaitu class formloginPetugasTU digunakan untuk melakukan login ke Aplikasi Pengarsipan Skripsi.
3. Main Menu: Class Main Menu memiliki fungsi yaitu sebagai menu utama, menampilkan 5 button utama, terdiri dari Data Mahasiswa, Data Dosen, Data Kategori Skripsi, Data Skripsi dan Pencarian Skripsi pada Aplikasi Pengarsipan Skripsi.
4. Data Mahasiswa: Class Data Mahasiswa memiliki fungsi yaitu untuk melakukan operasi CRUD sederhana di tabel t\_data\_mahasiswa pada Aplikasi Pengarsipan Skripsi.
5. Data Dosen: Class Data Dosen memiliki fungsi yaitu untuk melakukan operasi CRUD sederhana di tabel t\_data\_dosen pada Aplikasi Pengarsipan Skripsi.
6. Data Kategori Skripsi: Class Data Kategori Skripsi memiliki fungsi yaitu untuk melakukan operasi CRUD sederhana di tabel t\_kategori\_skripsi pada Aplikasi Pengarsipan Skripsi.
7. Data Skripsi: Class Data Skripsi memiliki fungsi yaitu untuk melakukan operasi CRUD sederhana di tabel t\_data\_skripsi pada Aplikasi Pengarsipan Skripsi.
8. Pencarian Skripsi: Class Pencarian Skripsi memiliki fungsi yaitu untuk melakukan search atau pencarian data skripsi pada tabel gabungan dari t\_data\_skripsi, t\_kategori\_skripsi, t\_data\_mahasiswa dan juga t\_data\_dosen pada Aplikasi Pengarsipan Skripsi.

## **3.4 Sequence Diagram**

Sequence diagram adalah merupakan salah satu dari behaviour diagram dimana diagram ini menampilkan dan menggambarkan sebuah kegiatan atau proses secara sequential dari objek - objek setiap use case, dan menjelaskan secara detail mengenai waktu proses dari objek dan message yang dikirim dan diterima antar objek.

* Sequence Diagram : Login Petugas TU

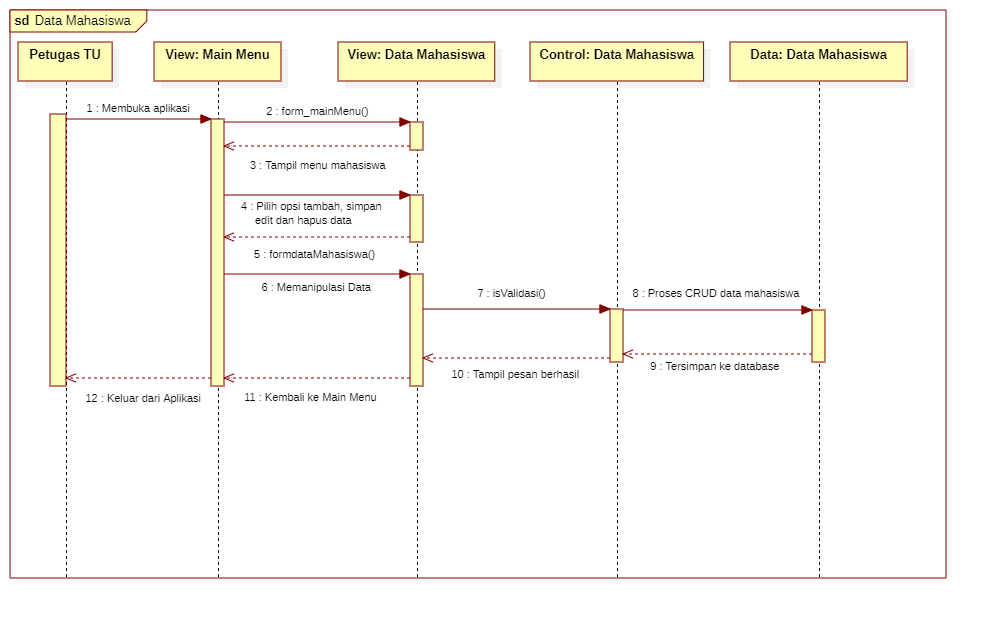
User Petugas TU melakukan proses login, proses login dilakukan dengan cara memasukkan input data login (username dan juga password). Lalu sistem basis data akan mengecek data login tersebut, apakah sudah sesuai dan juga melakukan validasi data, jika berhasil akan menampilkan berhasil login, lalu menuju ke menu utama aplikasi, sedangkan jika salah maka akan menampilkan pesan pop-up salah memasukkan input data login(username dan juga password).



Gambar 2 Sequence Diagram Login Petugas TU

* Sequence Diagram : Data Mahasiswa

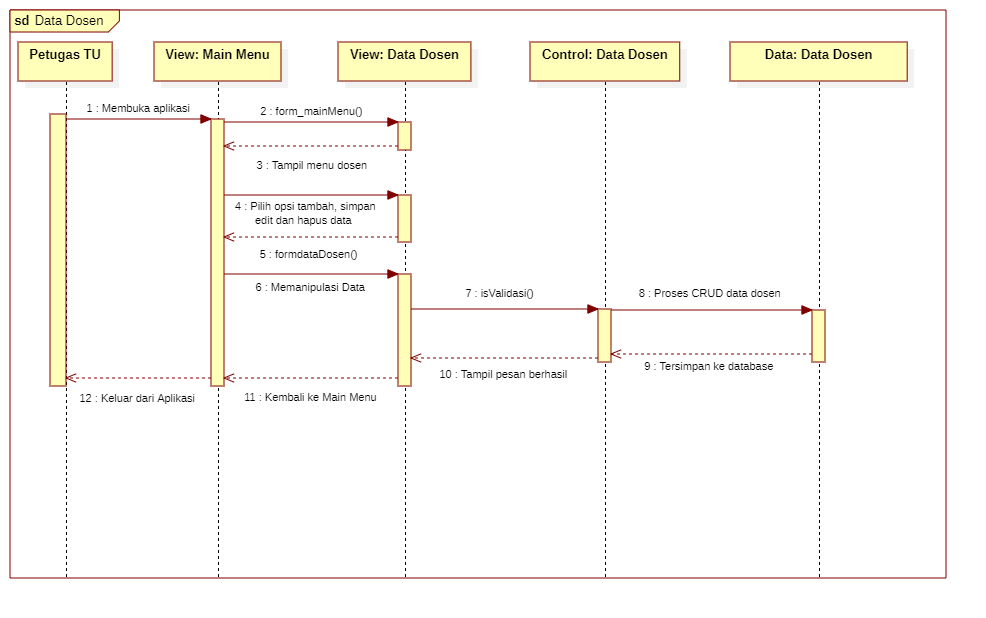
User Petugas TU setelah melakukan login, akan menampilkan Menu Utama, lalu pilih Menu data mahasiswa, kemudian lakukan operasi CRUD atau memanipulasi data pada tabel t\_data\_mahasiswa, maka setelah dilakukan akan dicek validasi-nya apakah telah sesuai atau belum. Setelah itu maka data akan berhasil tersimpan ke database dan menampilkan pesan berhasil menyimpan atau mengubah dan atau menghapus data.



Gambar 3 Sequence Diagram Data Mahasiswa

* Sequence Diagram : Data Dosen

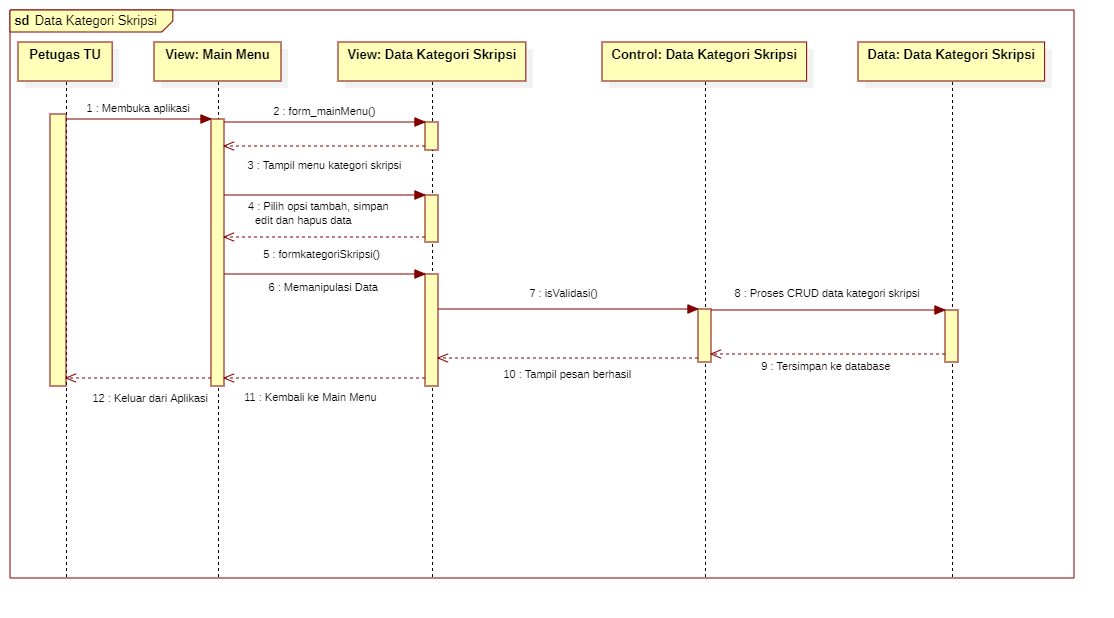
User Petugas TU setelah melakukan login, akan menampilkan Menu Utama, lalu pilih Menu data dosen, kemudian lakukan operasi CRUD atau memanipulasi data pada tabel t\_data\_dosen, maka setelah dilakukan akan dicek validasi-nya apakah telah sesuai atau belum. Setelah itu maka data akan berhasil tersimpan ke database dan menampilkan pesan berhasil menyimpan atau mengubah dan atau menghapus data.



Gambar 4 Sequence Diagram Data Dosen

* Sequence Diagram : Data Kategori Skripsi

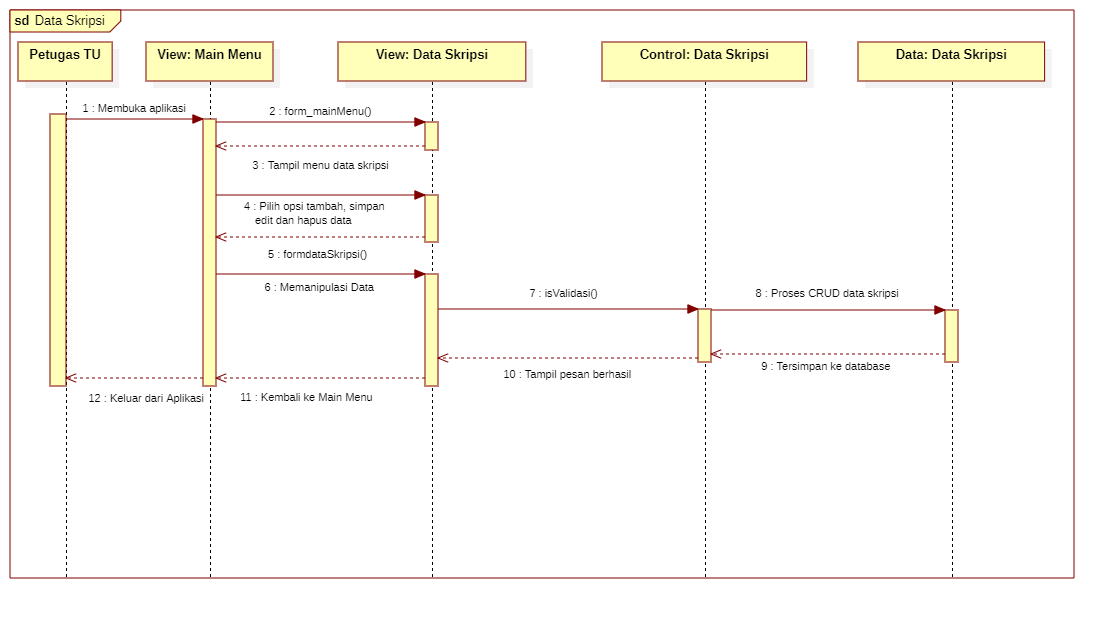
User Petugas TU setelah melakukan login, akan menampilkan Menu Utama, lalu pilih Menu data kategori skripsi, kemudian lakukan operasi CRUD atau memanipulasi data pada tabel t\_kategori\_skripsi, maka setelah dilakukan akan dicek validasi-nya apakah telah sesuai atau belum. Setelah itu maka data akan berhasil tersimpan ke database dan menampilkan pesan berhasil menyimpan atau mengubah dan atau menghapus data.



Gambar 5 Sequence Diagram Data Kategori Skripsi

* Sequence Diagram : Data Skripsi

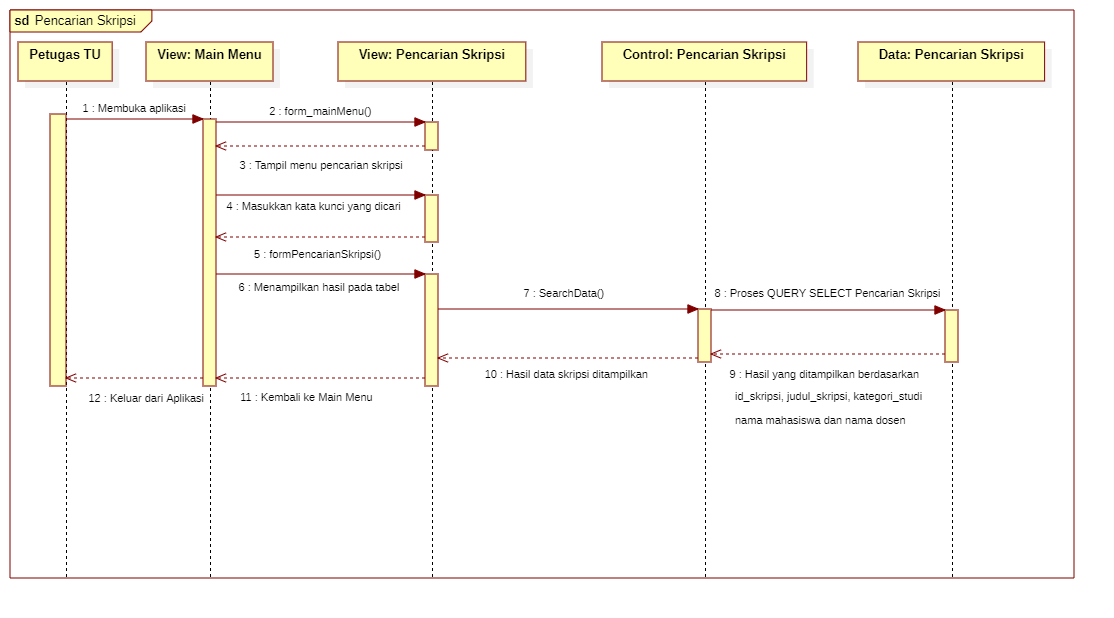
User Petugas TU setelah melakukan login, akan menampilkan Menu Utama, lalu pilih Menu data skripsi, kemudian lakukan operasi CRUD atau memanipulasi data pada tabel t\_data\_skripsi, maka setelah dilakukan akan dicek validasi-nya apakah telah sesuai atau belum. Setelah itu maka data akan berhasil tersimpan ke database dan menampilkan pesan berhasil menyimpan atau mengubah dan atau menghapus data.



Gambar 6 Sequence Diagram Data Skripsi

* Sequence Diagram : Pencarian Skripsi

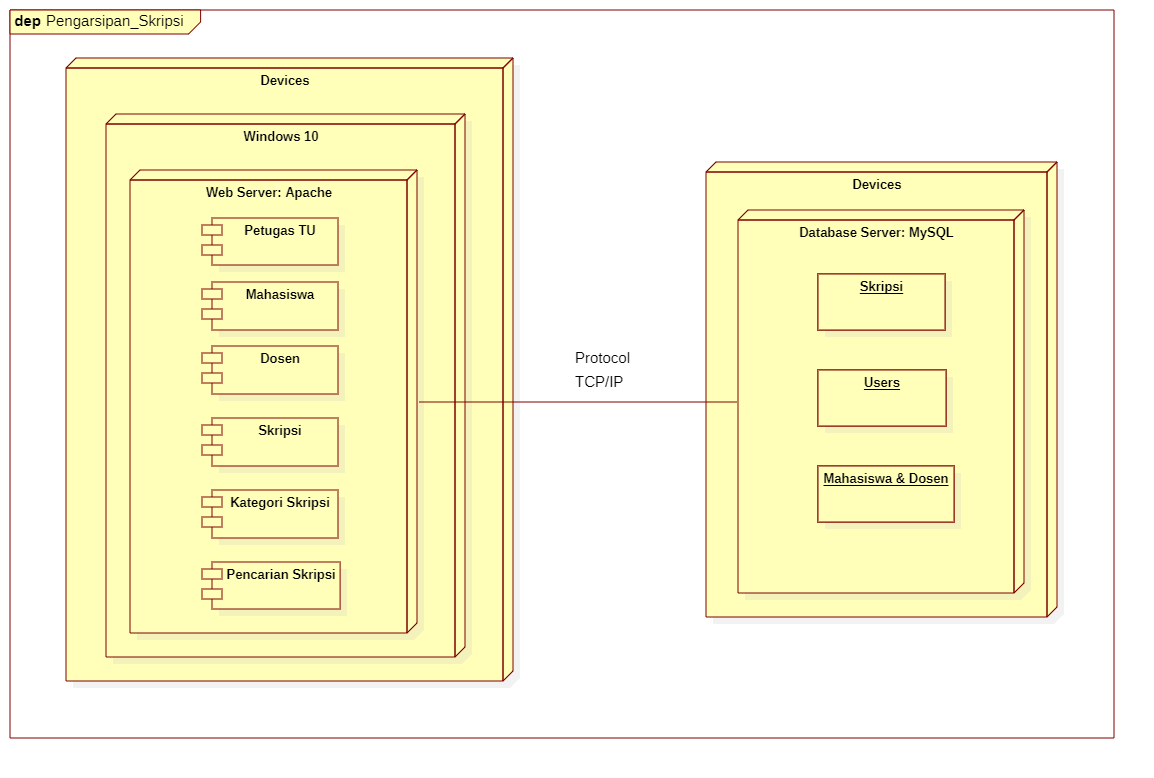
User Petugas TU setelah melakukan login, akan menampilkan Menu Utama, lalu pilih Menu data pencarian skripsi, kemudian masukkan kata kunci yang akan dicari pada textfieldnya. Kemudian pada Control: Pencarian Skripsi, yaitu metode SearchData() berfungsi untuk mencari dan atau menampilkan dari kata yang dimasukkan, yaitu menggunakan QUERY SELECT dan LIKE. Maka hasilnya pada tabel akan keluar atau muncul berdasarkan kata kunci yang dimasukkan, bisa id\_skripsi, judul\_skripsi, kategori\_studi, nama\_mahasiswa, dan juga data\_dosen



Gambar 7 Sequence Diagram Pencarian Skripsi

## **3.7 Deployment Diagram**

Deployment diagram adalah diagram struktur yang menunjukan bagaimana arsitektur sistem secara fisik dari suatu sistem dan cara mengimplementasikannya di perangkat lunak. Deployment diagram menampilkan node – node yang di dalamnya terdapat komponen dan juga artifak. Untuk deployment diagram pada Aplikasi Pengarsipan Skripsi dapat di lihat pada gambar di bawah ini :



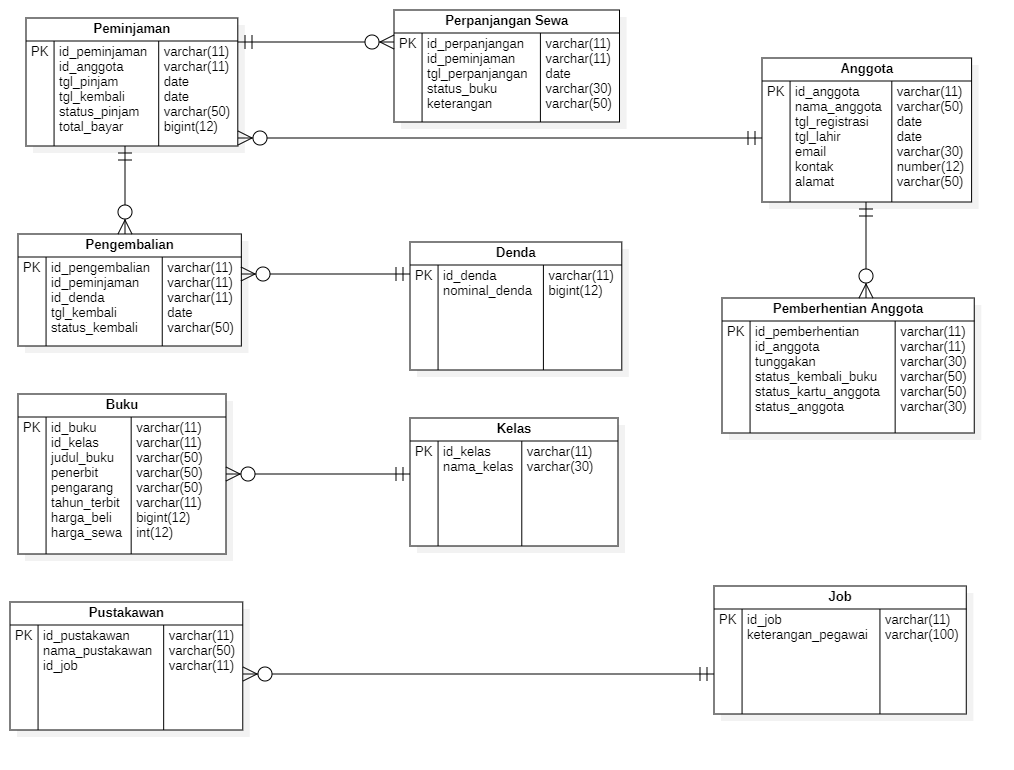
Gambar 8 Deployment Diagram Aplikasi Pengarsipan Skripsi

# **BAB IV** **Data Design**

## **4.1 Logical & Physical Design**

Entity Relationship Diagram(ERD) yaitu sebuah diagram yang menjelaskan mengenai relasi antar sebuah entitas atau tabel. Kemudian untuk ERD pada perancangan aplikasi pengarsipan skripsi ini dapat dilihat pada gambar di bawah ini:

**Skema Relasi**



* Struktur Tabel

1. Tabel t\_petugas\_tu

Primary Key : id

Fungsi : Untuk menyimpan data user dari PetugasTU.

Tabel 7 Struktur Tabel PetugasTU

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No. | Nama Kolom | Tipe Data | Ukuran | Keterangan |
| 1. | id | integer | 11 | id\_petugas |
| 2. | nama\_petugas | varchar | 30 | nama petugasTU |
| 3. | nama\_pengguna | varchar | 30 | nama pengguna petugasTU |
| 4. | kata\_sandi | varchar | 30 | kata sandi akun |

1. Tabel t\_data\_mahasiswa

Primary Key : id

Fungsi : Untuk menyimpan data mahasiswa oleh PetugasTU.

Tabel 8 Struktur Tabel Data Mahasiswa

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No. | Nama Kolom | Tipe Data | Ukuran | Keterangan |
| 1. | id | integer | 11 | id\_mahasiswa |
| 2. | nim | varchar | 10 | nim mahasiswa |
| 3. | nama | varchar | 30 | nama mahasiswa |
| 4. | jurusan | varchar | 30 | jurusan mahasiswa |
| 5. | angkatan | varchar | 4 | angkatan mahasiswa |
| 6. | email | varchar | 30 | email mahasiswa |

1. Tabel t\_data\_dosen

Primary Key : id

Fungsi : Untuk menyimpan data dosen oleh PetugasTU.

Tabel 9 Struktur Tabel Data Dosen

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No. | Nama Kolom | Tipe Data | Ukuran | Keterangan |
| 1. | id | integer | 11 | id\_dosen |
| 2. | nid | varchar | 10 | nid dosen |
| 3. | nama | varchar | 30 | nama dosen |
| 4. | email | varchar | 30 | email dosen |

1. Tabel t\_kategori\_skripsi

Primary Key : id

Fungsi : Untuk menyimpan data kategori skripsi oleh PetugasTU.

Tabel 10 Struktur Tabel Data Kategori Skripsi

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No. | Nama Kolom | Tipe Data | Ukuran | Keterangan |
| 1. | id | integer | 11 | id\_kategori |
| 2. | kode\_skripsi | varchar | 11 | kode skripsi |
| 3. | kategori skripsi | varchar | 30 | kategori skripsi |
| 4. | kategori studi | varchar | 30 | kategori studi |

1. Tabel t\_data \_skripsi

Primary Key : id

Fungsi : Untuk menyimpan data skripsi oleh PetugasTU.

Tabel 11 Struktur Tabel Data Skripsi

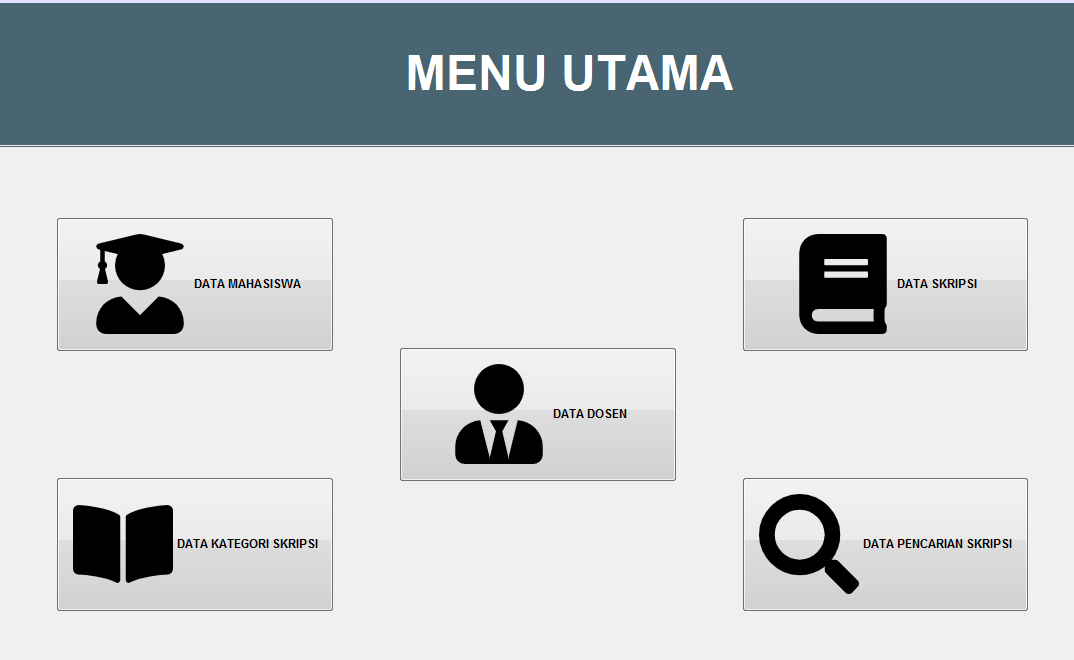
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No. | Nama Kolom | Tipe Data | Ukuran | Keterangan |
| 1. | id | integer | 11 | id\_skripsi |
| 2. | kode\_skripsi | varchar | 11 | kode skripsi |
| 3. | judul | varchar | 100 | judul skripsi |
| 4. | nim | varchar | 10 | nim mahasiswa |
| 5. | nid | varchar | 10 | nid dosen |
| 6. | tahun | varchar | 10 | tahun dibuat |

# **BAB V** **User Interface Design**

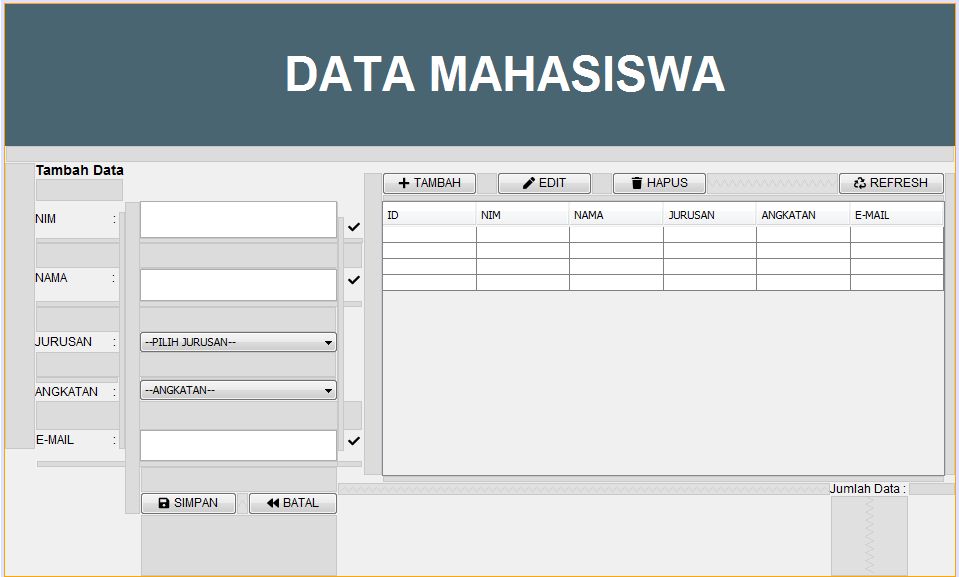
## **5.1 Graphical User Interface**

**Tampilan User Interface Aplikasi Pengarsipan Skripsi**

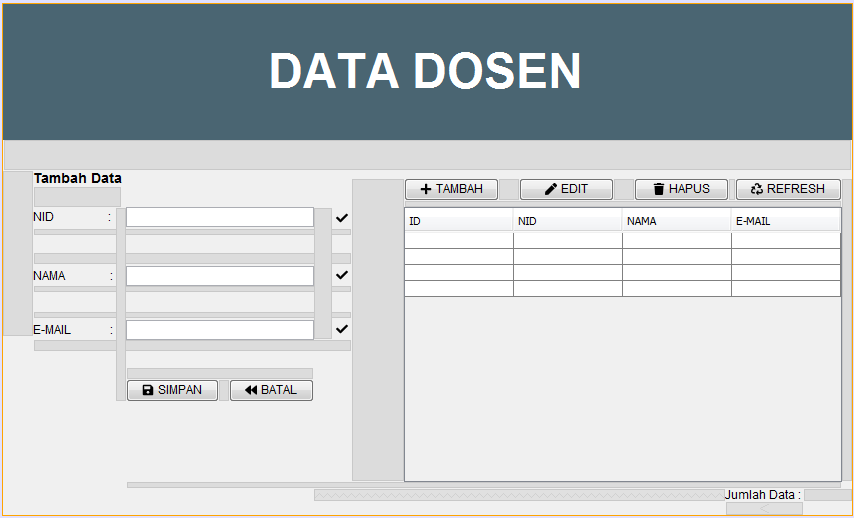
1. **Tampilan Menu Utama**

****

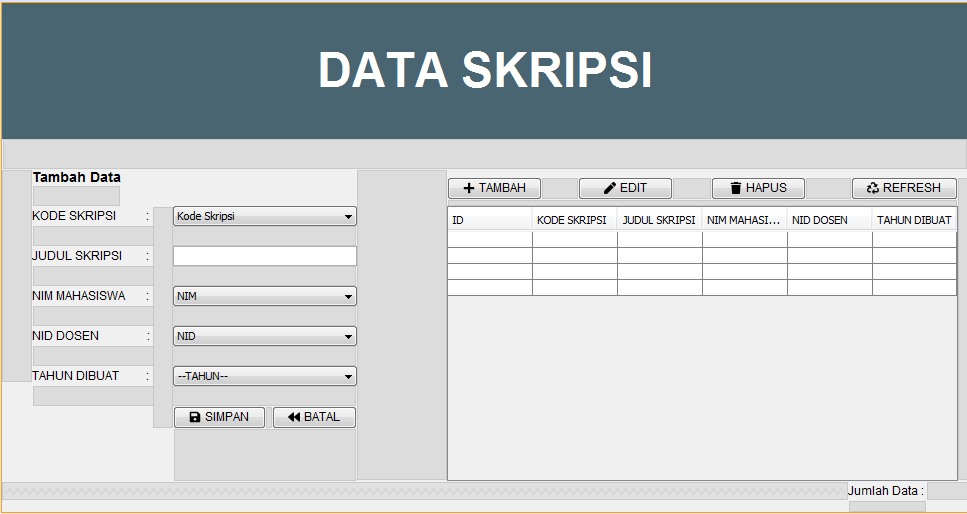
1. **Tampilan Data Mahasiswa**

****

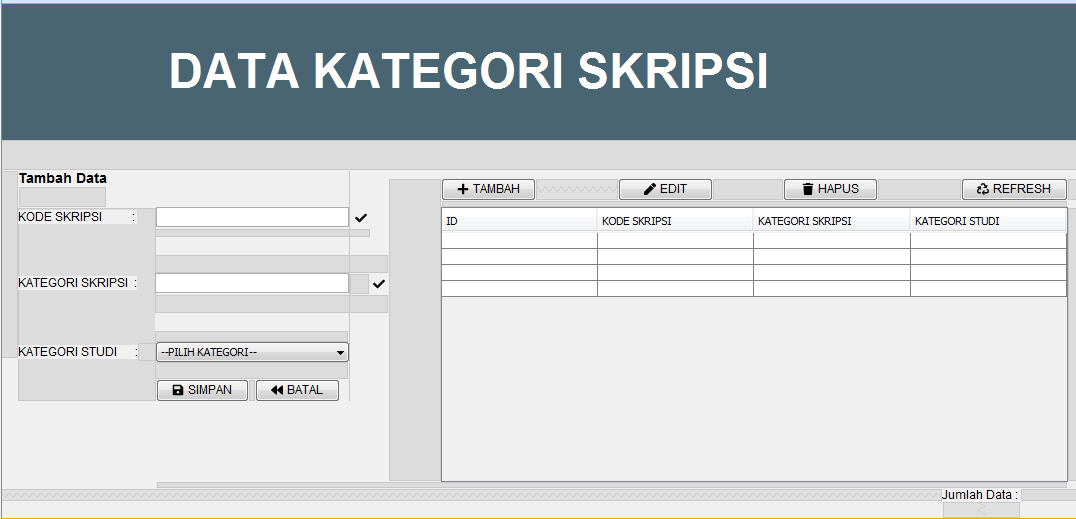
1. **Tampilan Data Dosen**

****

1. **Tampilan Data Skripsi**

****

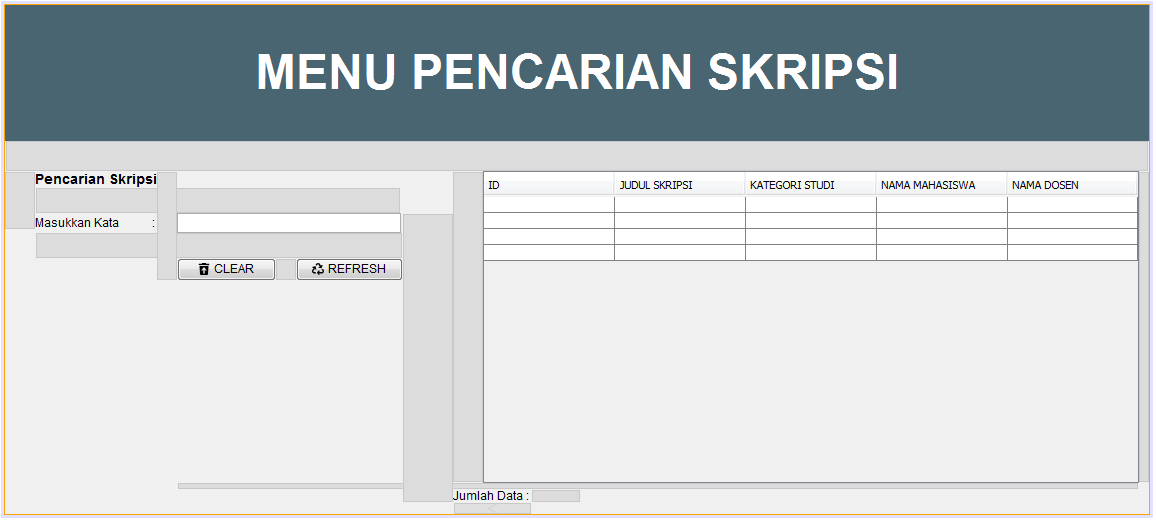
1. **Tampilan Kategori Skripsi**

****

1. **Tampilan Login Petugas TU**

****

1. **Tampilan Pencarian Skripsi**

****