



**LAPORAN SKRIPSI**

**APLIKASI REKOMENDASI KESAMAAN JUDUL  
SKRIPSI PADA SEMINAR PROPOSAL  
MENGUNAKAN ALGORITMA  
*RATCLIFF/OBERSHELP*  
BERBASIS ANDROID**

**SETYO ADI SASONO**

**NIM. 201953073**

**DOSEN PEMBIMBING**

**Nanik Susanti, S.Kom., M.Kom  
Syafiul Muzid, ST., M.Cs., MTA**

**PROGRAM STUDI SISTEM INFORMASI  
FAKULTAS TEKNIK  
UNIVERSITAS MURIA KUDUS**

**2023**

## **HALAMAN PERSETUJUAN**

### **APLIKASI REKOMENDASI KESAMAAN JUDUL SKRIPSI PADA SEMINAR PROPOSAL MENGGUNAKAN ALGORITMA *RATCLIFF/OBERSHELP* BERBASIS ANDROID**

**SETYO ADI SASONO**

**NIM. 201953073**

Kudus, 07 Agustus 2023

Menyetujui,

Pembimbing Utama,

Pembimbing Pendamping,

Nanik Susanti, S.Kom., M.Kom

Syafiul Muzid, ST., M.Cs., MTA

NIDN. 0608088201

NIDN. 0623068301

Mengetahui

Koordinator Skripsi

Yudie Irawan, S.Kom M.Kom., MTA, MOS

NIDN. 0004047501

## **HALAMAN PENGESAHAN**

### **APLIKASI REKOMENDASI KESAMAAN JUDUL SKRIPSI PADA SEMINAR PROPOSAL MENGGUNAKAN ALGORITMA *RATCLIFF/OBERSHELP* BERBASIS ANDROID**

**SETYO ADI SASONO**

**NIM. 201953073**

Kudus, 07 Agustus 2023

Menyetujui,

Pembimbing Utama,

Pembimbing Pendamping,

Nanik Susanti, S.Kom., M.Kom  
NIDN. 0608088201

Syafiul Muzid, ST., M.Cs., MTA  
NIDN. 0623068301

Mengetahui

Plt.Ketua Program Studi Sistem  
Informasi

Koordinator Skripsi

Nanik Susanti, S.Kom., M.Kom  
NIDN. 0608088201

Yudie Irawan, S.Kom M.Kom., MTA, MOS  
NIDN. 0004047501

## **PERNYATAAN KEASLIAN**

Saya yang bertanda tangan dibawah ini :

Nama : Setyo Adi Sasono  
NIM : 201953073  
Tempat & Tanggal Lahir : Grobogan, 05 September 1999  
Judul Skripsi/Tugas Akhir : Aplikasi Rekomendasi Kesamaan Judul Skripsi  
Pada Seminar Proposal Menggunakan Algoritma  
*Ratcliff/Obershelp* Berbasis Android

Menyatakan dengan sebenarnya bahwa penulisan Skripsi/Tugas Akhir ini berdasarkan hasil penelitian, pemikiran dan pemaparan asli dari saya sendiri, baik untuk naskah laporan maupun kegiatan lain yang tercantum sebagai bagian dari Skripsi ini. Seluruh ide, pendapat, atau materi dari sumber lain telah dikutip dalam Skripsi dengan cara penulisan referensi yang sesuai.

Demikian pernyataan ini saya buat dengan sesungguhnya dan apabila di kemudian hari terdapat penyimpangan dan ketidakbenaran dalam pernyataan ini, maka saya bersedia menerima sanksi akademik berupa pencabutan gelar dan sanksi lain sesuai dengan peraturan yang berlaku di Universitas Muria Kudus.

Demikian pernyataan ini saya buat dalam keadaan sadar tanpa paksaan dari pihak manapun.

Kudus, Desember 2023

Yang memberi pernyataan,

Setyo Adi Sasono

NIM. 20195073

**APLIKASI REKOMENDASI KESAMAAN JUDUL SKRIPSI PADA  
SEMINAR PROPOSAL MENGGUNAKAN ALGORITMA  
RATCLIFF/OBERSHELP BERBASIS ANDROID**

Nama Mahasiswa : Setyo Adi Sasono

NIM : 201953073

Pembimbing :

1. Nanik Susanti, S.Kom., M.Kom
2. Syafiul Muzid, ST., M.Cs., MTA

## **RINGKASAN**

Penulisan skripsi merupakan tahap penting dalam menyelesaikan program sarjana, terutama pada Program Studi Sistem Informasi di Universitas Muria Kudus. Namun, seringkali terjadi kesalahan atau ketidaksesuaian antara judul skripsi yang diajukan dengan penelitian yang sudah ada sebelumnya, menyebabkan risiko duplikasi penelitian dan menghambat kemajuan mahasiswa. Dengan menggunakan algoritma *Ratcliff/Obershelp*, membandingkan judul skripsi dengan database judul skripsi yang telah ada sebelumnya.

Aplikasi ini dikembangkan dengan menggunakan metode UML (Unified Modelling Language) dan diterapkan dalam bentuk aplikasi Android dengan memanfaatkan bahasa pemrograman Kotlin dan menggunakan database Firebase. Dalam pengembangan aplikasi, pendekatan metode SDLC dengan *prototype* diterapkan dengan melakukan analisis kebutuhan melalui observasi, wawancara, dan studi pustaka. Hasil dan manfaat dari aplikasi rekomendasi kesamaan judul skripsi pada seminar proposal menggunakan algoritma *Ratcliff/Obershelp* berbasis android ini adalah sebagai sarana untuk mahasiswa dapat segera mengetahui apakah judul skripsi yang diajukan memiliki kemiripan dengan penelitian yang telah ada sebelumnya.

**Kata Kunci :** *Seminar Proposal, Rekomendasi, Judul Skripsi, Algoritma Ratcliff/Obershelp*

# **APPLICATION OF RECOMMENDATIONS FOR THE SIMILARITY OF THESIS TITLE IN PROPOSAL SEMINARS USING THE RATCLIFF/OBERSHELP ALGORITHM ANDROID BASED**

*Student Name* : Setyo Adi Sasono

*Student Identity Number* : 201953073

*Supervisor* :

1. Nanik Susanti, Skom., M.Kom
2. Syafiul Muzid, ST., M.Cs., MTA

## **ABSTRAC**

*Writing a thesis is an important stage in completing an undergraduate program, especially in the Information Systems Study Program at Muria Kudus University. However, errors or discrepancies often occur between the title of the proposed thesis and previously existing research, causing the risk of duplication of research and hampering student progress. Using the RatcliffOobershhelp algorithm, compare the thesis title with a previously existing database of thesis titles.*

*This application was developed using the UML (Unified Modeling Language) method and implemented in the form of an Android application using the Kotlin programming language and using the Firebase database. In application development, the SDLC method approach with prototypes is applied by conducting needs analysis through observation, interviews, and literature study. The results and benefits of the thesis title similarity recommendation application in proposal seminars using the Android-based RatcliffOobershhelp algorithm are as a means for students to immediately find out whether the proposed thesis title is similar to previous research.*

*Keywords: Proposal Seminar, Recommendations, Thesis Title, Algorithm Ratcliff/Oobershhelp*

## KATA PENGANTAR

Puji syukur kepada Allah SWT yang telah melimpahkan rahmat dan karunia-Nya, sehingga penulis dapat menyelesaikan skripsi yang berjudul “Aplikasi Rekomendasi Kesamaan Judul Skripsi Pada Seminar Proposal Menggunakan Algoritma *Ratcliff/Obershelp* Berbasis Android”.

Skripsi ini disusun sebagai salah satu syarat untuk menyelesaikan Program Studi Sistem Informasi di Fakultas Teknik Universitas Muria Kudus.

Dalam penyusunan skripsi ini, penulis telah banyak mendapatkan bantuan dan bimbingan dari berbagai pihak. Oleh karena itu, penulis ingin menyampaikan terima kasih kepada:

1. Bapak Prof. Dr. Ir. Darsono, M.Si selaku Rektor Universitas Muria Kudus.
2. Bapak Mohammad Dahlan, S.T, M.T selaku Dekan Fakultas Teknik Universitas Muria Kudus.
3. Nanik Susanti, Skom., M.Kom selaku Kepala Program Studi Sistem Informasi Fakultas Teknik Universitas Muria Kudus.
4. Ibu Nanik Susanti, S.Kom., M.Kom dan Bapak Syafiul Muzid, ST., M.Cs., MTA selaku pembimbing yang telah memberikan bimbingan dan arahan sehingga terselesaikannya penulisan laporan skripsi ini.
5. Bapak Yudie Irawan, M.Kom., MTA., MOS selaku koordinator Skripsi.
6. Nanik Susanti, Skom., M.Kom selaku dosen wali penulis.
7. Bapak dan Ibu dosen Sistem Informasi yang senantiasa memberi ilmu kepada penulis.
8. Ibu Sri Dwi Winarti, Bapak Sujono selaku orang tua penulis, yang selalu memberikan semangat, kasih sayang, doa, sehingga dapat memberi penulis semangat untuk dapat menyelesaikan skripsi.
9. Orang terdekat, yang selalu memberikan dukungan serta semangat satu sama lain.
10. Seluruh teman-teman penulis yang telah memberikan dukungan dan motivasi selama penulis menempuh pendidikan di Fakultas Teknik Universitas Muria Kudus.

Penulis berharap semoga langkah selanjutnya dalam mencari pekerjaan diridhoi oleh Allah SWT. Akhirnya sebagai penutup penulis berharap semoga Laporan Skripsi ini dapat memberikan manfaat bagi pembaca untuk mengembangkan ilmu pengetahuan, khususnya dibidang Ilmu Pengetahuan Teknologi dan Sistem Informasi. Aamiin.

Kudus, 25 Desember 2023

Penulis,

Setyo Adi Sasono



## DAFTAR ISI

HALAMAN PERSETUJUAN.....	ii
HALAMAN PENGESAHAN.....	iii
PERNYATAAN KEASLIAN.....	iv
RINGKASAN.....	v
ABSTRAC.....	vi
KATA PENGANTAR.....	vii
DAFTAR ISI.....	ix
DAFTAR GAMBAR.....	xi
BAB 1 PENDAHULUAN.....	1
1.1. Latar Belakang.....	1
1.2 Perumusan Masalah.....	2
1.3 Batasan Masalah.....	2
1.4 Tujuan.....	3
1.5 Manfaat.....	3
1.6 Metode Penelitian.....	4
1.6.1 Metode Pengumpulan Data.....	4
1.6.2 Metode Pengembangan Sistem.....	5
1.6.3 Metode Perancangan Sistem.....	6
1.7 Kerangka Pemikiran.....	8
BAB II TINJAUAN PUSTAKA.....	9
2.1 Penelitian Terkait.....	9
2.2 Tabel Perbandingan Penelitian Terkait.....	11
2.3 Landasan Teori.....	12
2.3.1 Pengertian Sistem.....	12
2.3.2 Pengertian Informasi.....	12
2.3.3 Pengertian Sistem Informasi.....	12
2.3.4 Pengertian Skripsi.....	12
2.3.5 Pengertian Algoritma <i>Ratcliff/Obershelp</i> .....	13
2.3.6 FOD (Flow Of Document).....	15
2.3.7 UML (Unified Modelling Language).....	16
BAB III METODOLOGI PENELITIAN.....	21
3.1 Objek Penelitian.....	21
3.1.1 Sekilas Tentang Program Studi Sistem Informasi.....	21
3.1.2 Visi dan Misi.....	21
3.1.3 Denah Lokasi.....	22
3.1.4 Struktur Organisasi.....	23

3.1.5	Deskripsi Tugas (Job Description) .....	23
3.2	Analisa Sistem Lama .....	25
3.2.1	Prosedur Pendaftaran Seminar Proposal Skripsi .....	25
3.2.2	Prosedur Pelaksanaan Seminar Proposal Skripsi .....	26

## DAFTAR GAMBAR

Gambar 1.1 Siklus Prototype .....	5
Gambar 1.2 Alur Kerangka Pemikiran.....	8
Gambar 2.1 Sample String .....	13
Gambar 2.2 Achor String .....	14
Gambar 2.3 String yang sama pertama .....	14
Gambar 2.4 String yang sama kedua.....	14
Gambar 3.1 Denah Lokasi Program Studi Sistem Informasi .....	22
Gambar 3.2 Struktur Organisasi Program Studi Sistem Informasi .....	23
Gambar 3.3 Aliran Dokumen Pendaftaran Seminar Proposal.....	26
Gambar 3.4 Aliran Dokumen Pelaksanaan Sidang Proposal Skripsi .....	27

## DAFTAR TABEL

Tabel 2.1 Perbandingan Penelitian .....	11
Tabel 2.2 Simbol-simbol diagram alir dokumen (Flow of Document) .....	15
Tabel 2.3 Simbol Bussiness Use Case Diagram.....	16
Tabel 2.4 Notasi Use Case .....	17
Tabel 2.5 Notasi Class Diagram.....	18
Tabel 2.6 Notasi Sequence Diagram .....	19
Tabel 2.7 Notasi Activity Diagram .....	19
Tabel 2.8 Notasi Statechart Diagram .....	20

# **BAB 1**

## **PENDAHULUAN**

### **1.1. Latar Belakang**

Penulisan skripsi merupakan tahap akhir yang penting dalam menyelesaikan program sarjana di berbagai disiplin ilmu. Skripsi merupakan bentuk penelitian yang dilakukan oleh mahasiswa dengan tujuan menggali pengetahuan baru, mengembangkan keterampilan analisis, dan memberikan kontribusi pada bidang studi yang dipilih.

Dalam program studi Sistem Informasi di Universitas Muria Kudus, penulisan skripsi juga menjadi tahap krusial dalam menyelesaikan program sarjana. Salah satu tahapan dalam penyusunan skripsi adalah seminar proposal, yaitu kegiatan presentasi dan diskusi mengenai rencana penelitian atau pengembangan yang akan dilakukan oleh mahasiswa. Seminar proposal bertujuan untuk mendapatkan masukan, saran, dan kritik dari dosen penguji, dosen pembimbing. Seminar proposal juga dapat membantu mahasiswa untuk mengetahui apakah judul skripsi yang dipilih sudah tepat, relevan, dan tidak tumpang tindih dengan judul skripsi lainnya.

Namun, dalam praktiknya, sering terjadi kesulitan dalam menentukan judul skripsi yang sesuai dengan minat, kemampuan, dan ketersediaan data mahasiswa. Selain itu, sering terjadi kemiripan atau kesamaan judul skripsi antara mahasiswa satu dengan yang lainnya. Hal ini dapat menyebabkan permasalahan seperti plagiat, duplikasi, atau kurangnya orisinalitas dalam penelitian atau pengembangan. Oleh karena itu, diperlukan sebuah aplikasi yang dapat memberikan rekomendasi judul skripsi yang sesuai dengan minat dan kemampuan mahasiswa, serta memiliki tingkat kesamaan yang rendah dengan judul skripsi lainnya.

Algoritma *Ratcliff/Obershelp* adalah algoritma yang digunakan untuk membandingkan kemiripan antara dua string atau teks berdasarkan pada perbandingan urutan karakter dalam kedua string. Dalam aplikasi pendeteksi kemiripan judul skripsi, algoritma ini membantu mahasiswa untuk mengecek apakah judul skripsi yang diajukan memiliki kemiripan dengan judul yang sudah ada sebelumnya. Dengan demikian, algoritma *Ratcliff/Obershelp* membantu

mencegah duplikasi penelitian dan mendorong variasi topik penelitian yang lebih luas.

Berdasarkan permasalahan yang diuraikan di atas, diperlukan sebuah aplikasi rekomendasi kesamaan judul skripsi menggunakan algoritma *Ratcliff/Obershelp* berbasis android. Aplikasi ini akan membantu mahasiswa dalam mengajukan judul skripsi yang unik dan menghindari duplikasi penelitian sebelumnya. Dengan menggunakan algoritma *Ratcliff/Obershelp*, aplikasi ini akan membandingkan judul skripsi yang diajukan oleh mahasiswa dengan database judul skripsi yang sudah ada sebelumnya, sehingga mahasiswa dapat segera mengetahui apakah judul tersebut memiliki kemiripan yang signifikan. Hal ini akan mendorong variasi topik penelitian yang lebih luas dan mengurangi risiko duplikasi penelitian di bidang Sistem Informasi.

## **1.2 Perumusan Masalah**

Berdasarkan penjelasan latar belakang di atas, permasalahan yang perlu dipecahkan meliputi merancang dan mengembangkan aplikasi, mengimplementasikan algoritma *Ratcliff/Obershelp* untuk merekomendasikan kesamaan judul skripsi, menyusun dan mengelola database judul skripsi sebelumnya, mengintegrasikan metode dengan aplikasi, serta memastikan akurasi dan efektivitas metode dalam mengidentifikasi rekomendasi kesamaan judul skripsi yang signifikan.

## **1.3 Batasan Masalah**

Untuk memastikan bahwa rumusan masalah tidak menyimpang dari tujuan awal, dibutuhkan batasan masalah. Batasan masalah ini akan membantu menjaga agar fokus penelitian tetap pada topik yang telah ditentukan.

1. Aplikasi pendeteksi kemiripan seminar proposal menggunakan algoritma *Ratcliff/Obershelp* berbasis android akan dikembangkan khusus untuk Program Studi Sistem Informasi di Universitas Muria Kudus.
2. Fokus aplikasi ini adalah merekomendasikan kesamaan judul skripsi pada seminar proposal, di mana mahasiswa mengajukan judul penelitian yang akan mereka teliti dan tulis.

3. Sistem ini dibangun sebagai aplikasi Android yang dibuat dengan menggunakan bahasa pemrograman Kotlin dan database Firebase.
4. Algoritma *Ratcliff/Obershelp* akan digunakan untuk membandingkan kesamaan judul skripsi yang diajukan oleh mahasiswa dengan database judul skripsi yang sudah ada sebelumnya.
5. Aplikasi ini akan memberikan hasil rekomendasi kesamaan judul skripsi dalam bentuk persentase atau nilai kemiripan yang dapat membantu mahasiswa mengevaluasi judulnya.

#### **1.4 Tujuan**

Membantu mahasiswa dalam menghindari duplikasi penelitian yang tidak disengaja. Dengan adanya aplikasi ini, mahasiswa dapat memperoleh informasi segera tentang kesamaan judul skripsi yang diajukan dengan penelitian yang sudah ada sebelumnya. Hal ini diharapkan dapat mendorong variasi topik penelitian yang lebih luas dan mengurangi risiko terulangnya penelitian yang serupa di bidang Sistem Informasi.

#### **1.5 Manfaat**

1. Bagi Penulis
  - a. Melakukan penerapan pengetahuan yang diperoleh, baik dari pengalaman dalam maupun di luar lingkungan perkuliahan.
  - b. Memperbandingkan teori yang dipelajari dalam perkuliahan dengan pengalaman nyata di lapangan.
2. Bagi Akademis
  - a. Menilai sejauh mana mahasiswa memahami materi yang diajarkan.
  - b. Menilai kemampuan mahasiswa dalam menerapkan ilmu yang dipelajarinya, baik secara teoritis maupun praktis, sebagai penilaian akhir.
  - c. Tujuannya adalah untuk memperluas dan memperdalam pemahaman tentang sistem informasi di Program Studi Sistem Informasi Universitas Muria Kudus.

## **1.6 Metode Penelitian**

### **1.6.1 Metode Pengumpulan Data**

Untuk memastikan keakuratan, relevansi, valid, dan keandalan data dalam penelitian ini, penulis telah melaksanakan dan memiliki metode pengumpulan data sebagai berikut:

#### **1. Sumber Data Primer**

Sumber data primer merupakan sumber data yang diperoleh langsung dari pengamatan atau mencatat informasi tentang objek penelitian yang ada. Ada beberapa contoh dari data primer, seperti:

##### **a. Wawancara**

Untuk penelitian ini, pendekatan yang diterapkan adalah dengan melakukan proses wawancara secara langsung dengan pihak-pihak yang terlibat. Data yang diperlukan adalah informasi terkait proses pendaftaran Skripsi yang sedang berlangsung di Program Studi Sistem Informasi Universitas Muria Kudus.

##### **b. Observasi**

Untuk mendapatkan pemahaman yang lebih mendalam, penulis juga melakukan observasi langsung di Program Studi Sistem Informasi Universitas Muria Kudus. Observasi ini bertujuan untuk melihat dan memperhatikan secara langsung proses pelaksanaan pendaftaran Skripsi yang sedang berlangsung di sana.

#### **2. Sumber Data Sekunder**

Sumber data sekunder merupakan sumber data yang diperoleh oleh penulis secara tidak langsung. Bisa dari buku, literatur ataupun dari penelitian yang terdahulu. Beberapa contoh dari sumber data sekunder meliputi:

##### **a. Studi Kepustakaan**

Studi kepustakaan adalah metode penelitian yang melibatkan pengumpulan, evaluasi, dan sintesis literatur yang relevan dengan topik penelitian untuk memahami penelitian sebelumnya.

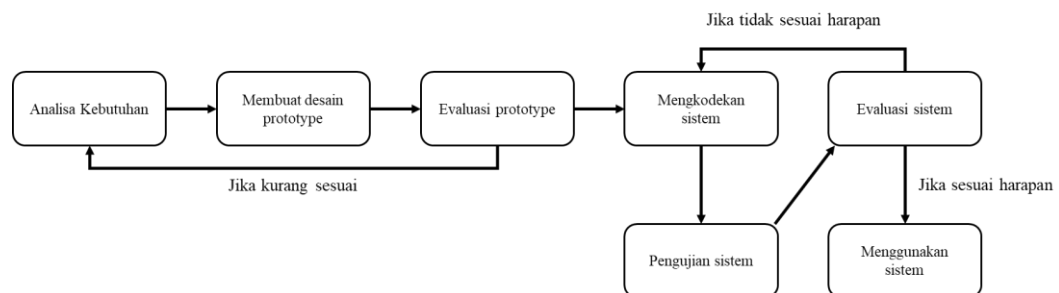


b. Studi Dokumentasi

Studi dokumentasi adalah metode penelitian yang melibatkan pengumpulan dan analisis data dari berbagai dokumen tertulis, seperti laporan, catatan, surat, dan buku, untuk mendapatkan pemahaman mendalam tentang suatu topik atau masalah berdasarkan informasi tertulis yang ada.

### 1.6.2 Metode Pengembangan Sistem

Metode pengembangan sistem yang akan diterapkan dalam penelitian ini adalah model SDLC (*System Development Life Cycle*) dengan *Prototype*. Model SDLC adalah pendekatan sistematis untuk mengembangkan dan memelihara sistem informasi. Model ini terdiri dari serangkaian langkah yang dijalankan secara berurutan untuk memastikan pengembangan sistem yang terstruktur dan terencana. Metode *Prototype* adalah salah satu pendekatan dalam model SDLC yang bertujuan untuk menciptakan *prototype* sistem yang dapat dievaluasi dan dikembangkan lebih lanjut berdasarkan umpan balik dari pengguna.



**Gambar 1.1 Siklus *Prototype***

Berikut adalah langkah-langkah umum dalam metode Prototype dalam model SDLC:

a. Analisa Kebutuhan

Pada tahap ini, pengembang aplikasi mengidentifikasi serta menganalisis seluruh persyaratan sistem yang akan dikembangkan. Tujuannya adalah untuk memahami secara mendalam apa yang diinginkan oleh pengguna atau pemangku kepentingan.

b. Membuat *Prototype*

Dalam tahap ini, pengembang perangkat lunak membuat prototipe atau perancangan sementara yang fokus pada penyajian kepada pengguna. Prototipe ini biasanya mencakup antarmuka pengguna, input yang dapat dimasukkan, dan format output yang dihasilkan.

c. Evaluasi *Prototype*

Prototipe yang telah dibuat dievaluasi untuk mengetahui sejauh mana prototipe tersebut sesuai dengan harapan pengguna atau pemangku kepentingan. Evaluasi ini dapat melibatkan penggunaan prototipe oleh pengguna yang kemudian memberikan umpan balik.

d. Mengkodekan Sistem

Jika prototipe disetujui, pengembang perangkat lunak melanjutkan dengan mengkodekan sistem menggunakan bahasa pemrograman yang sesuai. Pada tahap ini, semua komponen perangkat lunak dibangun dan diimplementasikan.

e. Pengujian Sistem

Setelah perangkat lunak dikodekan, tahap pengujian dilakukan agar memeriksa perangkat lunak berjalan dengan lancar dan memenuhi semua persyaratan yang telah ditetapkan.

f. Evaluasi Sistem

Setelah pengujian selesai, evaluasi perangkat lunak dilakukan. Jika memenuhi persyaratan, tahap selanjutnya dilanjutkan. Jika tidak, pengembang perlu memperbaiki dan mengulangi tahap mengkodekan sistem serta pengujian sistem.

g. Menggunakan Sistem

Setelah perangkat lunak melewati evaluasi dan diterima, perangkat lunak sudah siap dipergunakan. Pengguna dapat memanfaatkan perangkat lunak sesuai dengan tujuan dan kebutuhan.

### **1.6.3 Metode Perancangan Sistem**

Dalam pengembangan perangkat lunak berbasis pemrograman berorientasi objek, terdapat teknik pemodelan standar yang sering digunakan. Salah satu teknik tersebut adalah *Unified Modeling Language* (UML). UML merupakan sebuah bahasa visual yang dipakai untuk memodelkan dan mengkomunikasikan

sistem dengan menggunakan diagram dan teks pendukung (Sukamto & Shalahuddin, 2018). Ada beberapa jenis diagram *Unified Modelling Language* (UML), antara lain:

a. *Use Case Diagram*

*Use Case Diagram* menggambarkan hubungan dan interaksi antara sistem dengan aktor-aktor yang terlibat dalam sistem. Ini membantu dalam memahami kebutuhan fungsional sistem dan mengidentifikasi skenario penggunaan.

b. *Class Diagram*

*Class Diagram* digunakan untuk mendefinisikan karakteristik, metode, dan hubungan yang membentuk struktur kelas sistem. Ini mendukung pemahaman desain objek-orientasi dan pengembangan kode.

c. *Sequence Diagram*

*Sequence Diagram* menampilkan diagram interaksi antar objek dalam sistem yang telah disusun pada sebuah urutan atau rangkaian waktu. Ini berguna untuk memodelkan alur logika dan interaksi objek dalam skenario tertentu.

d. *Activity Diagram*

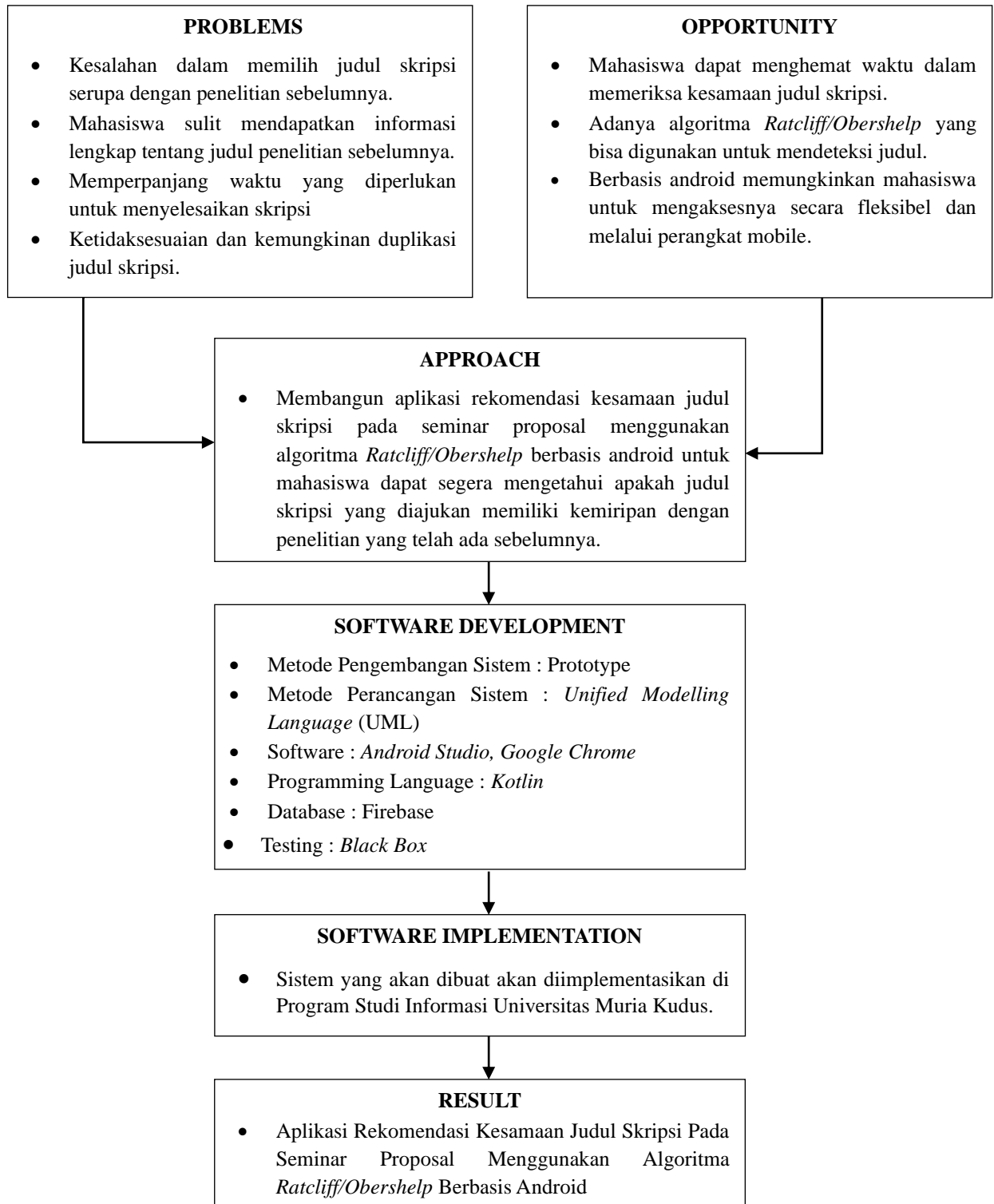
*Activity Diagram* digunakan untuk mendefinisikan bisnis internal sistem atau alur kerja. Ini membantu dalam memodelkan langkah-langkah yang dilakukan dalam suatu aktivitas atau algoritma.

e. *Statechart Diagram*

*Statechart Diagram* menggambarkan transisi keadaan suatu objek dalam sistem. Ini berguna untuk memodelkan bagaimana objek merespons peristiwa dan mengubah keadaannya dari satu keadaan ke keadaan lain.

## 1.7 Kerangka Pemikiran

Berikut adalah garis besar kerangka pemikiran yang penulis gunakan dalam proses pengembangan sistem ini:



**Gambar 1.2 Alur Kerangka Pemikiran**

## **BAB II**

### **TINJAUAN PUSTAKA**

#### **2.1 Penelitian Terkait**

Putra dkk. (2019) dalam jurnal penelitiannya menjelaskan tentang Pendeteksi Kesamaan Dokumen Pada Sistem Informasi Pendaftaran Proposal Skripsi Dengan Pendekatan Algoritma Rabin-Karp. Permasalahan yang dihadapi dalam hal plagiarisme pada konten digital tindakan menyalin dan menempelkan dokumen asli tanpa memberikan pengakuan yang layak kepada sumbernya. Sehingga penulis menggunakan suatu metode yang dapat menganalisis plagiarisme secara teknis. Dalam mengidentifikasi plagiarisme dalam teks, algoritma pencarian string Rabin-Karp dipilih karena efektivitas nya. Akan tetapi kekurangan dalam penelitian ini yaitu tidak adanya notifikasi aplikasi dan konsultasi dosen melalui aplikasi berbasis android.

Alamsyah dan Rasyidan, (2019) dalam jurnal menjelaskan tentang Deteksi Plagiarisme Tingkat Kemiripan Judul Skripsi Pada Fakultas Teknologi Informasi Menggunakan Algoritma Winnowing. Permasalahan yang di hadapi saat mengirimkan judul, beberapa mahasiswa berulang kali mengirimkan judul skripsi karena tidak menyadari bahwa judul yang mereka ajukan sudah ada. Hal ini dikarenakan dalam menentukan apakah judul diterima atau ditolaknya masih berlangsung secara manual yang dengan cara memeriksa satu per satu yang tersimpan dalam file Microsoft Excel. Sehingga penulis mencari tahu sejauh mana teks serupa satu sama lain, penulis menggunakan algoritma Winnowing yang dapat secara otomatis mendeteksi kemiripan teks antara judul yang diajukan dengan judul-judul yang sudah ada. Akan tetapi kekurangan dalam penelitian ini yaitu tidak adanya notifikasi aplikasi dan konsultasi dosen melalui aplikasi berbasis android.

Latifah dan Muzid (2021) dalam jurnal menjelaskan tentang Sistem Informasi Manajemen Pendaftaran Skripsi Online Dengan Fitur Deteksi Plagiarisme. Permasalahan yang dihadapi dengan meningkatnya jumlah mahasiswa yang mengajukan judul skripsi, risiko terjadinya plagiarisme judul juga meningkat. Hal ini dapat menyebabkan terjadinya kesamaan judul antara

skripsi yang diajukan oleh mahasiswa yang berbeda. Sehingga penulis untuk mengatasi masalah plagiarisme judul dan konten skripsi, telah dikembangkan sistem informasi untuk manajemen skripsi yang memiliki fitur deteksi plagiarisme terintegrasi. Akan tetapi kekurangan dalam penelitian ini yaitu tidak adanya notifikasi aplikasi dan konsultasi dosen melalui aplikasi berbasis android.

Thamrin dan Andriani, (2021) dalam jurnal menjelaskan tentang Perancangan Sistem Informasi Pendaftaran Dan Pengelolaan Data Skripsi Mahasiswa Berbasis Web. Permasalahan yang dihadapi untuk mengajukan judul skripsi, mahasiswa harus melakukan pengajuan judul skripsi secara langsung ke kampus. Pengumuman mengenai judul yang diajukan masih dilakukan melalui papan pengumuman di kampus STIMIK Sepuluh Nopember. Sehingga penulis membuat sebuah situs web yang mana bisa dipergunakan mahasiswa untuk pengajuan judul skripsi, dan bisa digunakan untuk memeriksa daftar judul skripsi yang sudah dipilih sebelumnya. Akan tetapi kekurangan dalam penelitian ini yaitu tidak adanya notifikasi serta fitur Persentase Plagiarisme serta tidak adanya notifikasi aplikasi dan konsultasi dosen melalui aplikasi berbasis android.

Sembiring dkk. (2022) dalam jurnal menjelaskan tentang Penerapan Data Mining Menggunakan Algoritma K-Means Untuk Menentukan Judul Skripsi Dan Jurnal Penelitian (Studi FTIK UNPRI). Permasalahan yang tengah dihadapi adalah kesulitan yang dihadapi oleh mahasiswa dalam menentukan judul skripsi. Mahasiswa seringkali bingung dan tidak mengerti bagaimana mencari ide judul yang sesuai dengan bidang ilmu mereka. Untuk mengatasi masalah ini, penulis merancang sebuah sistem yang menggunakan Data Mining dengan Clustering K-Means. Sistem ini bertujuan agar mempermudah mahasiswa untuk menemukan ide judul skripsi. dan jurnal penelitian dengan menyediakan kriteria pendukung. Akan tetapi kekurangan dalam penelitian ini yaitu tidak adanya notifikasi aplikasi dan konsultasi dosen melalui aplikasi berbasis android.

## 2.2 Tabel Perbandingan Penelitian Terkait

**Tabel 2.1 Perbandingan Penelitian**

No.	Judul	Berbasis	Algoritma	Persentase Plagiarisme	Konsultasi Dosen	Application Notification
1	Pendeteksi Kesamaan Dokumen Pada Sistem Informasi Pendaftaran Proposal Skripsi Dengan Pendekatan Algoritma Rabin-Karp. (Putra dkk, 2019)	Web	Rabin Karp	✓	—	—
2	Deteksi Plagiarisme Tingkat Kemiripan Judul Skripsi Pada Fakultas Teknologi Informasi Menggunakan Algoritma Winnowing. (Alamsyah & Rasyidan, 2019)	Web	Winnowing	✓	—	—
3	Sistem Informasi Manajemen Pendaftaran Skripsi Online Dengan Fitur Deteksi Plagiarisme. (Latifah & Muzid, 2021)	Web	—	✓	—	—
4	Perancangan Sistem Informasi Pendaftaran Dan Pengelolaan Data Skripsi Mahasiswa Berbasis Web. (Thamrin & Andriani, 2021)	Web	—	—	—	—
5	Penerapan Data Mining Menggunakan Algoritma K-Means Untuk Menentukan Judul Skripsi Dan Jurnal Penelitian (Studi FTIK	Web	K-Means	✓	—	—

UNPRI). (Sembiring dkk. 2022)						
6.	Aplikasi Rekomendasi Kesamaan Judul Skripsi Pada Seminar Proposal Android & Web Ratcliff / Obershelp			✓	✓	✓
Menggunakan Algoritma <i>Ratcliff/Obershelp</i> Berbasis Android (Sasono, 2023)						

## 2.3 Landasan Teori

### 2.3.1 Pengertian Sistem

Sistem adalah suatu kesatuan yang terdiri dari bagian-bagian yang saling berhubungan dan saling bergantung untuk mencapai tujuan bersama. Sistem memiliki karakteristik seperti komponen, batasan, lingkungan, input, output, proses, umpan balik dan tujuan. (Sutabri, 2012).

### 2.3.2 Pengertian Informasi

Informasi adalah hasil olahan data yang memiliki makna bagi penerimanya dan dapat digunakan untuk membuat keputusan yang lebih baik. Data adalah fakta atau bagian dari fakta yang memiliki arti dan dihubungkan dengan kenyataan. Data dapat berupa simbol-simbol, gambar-gambar, angka-angka, huruf-huruf, yang menunjukkan suatu ide, objek, kondisi, atau situasi lain-lain. (Sutabri, 2012).

### 2.3.3 Pengertian Sistem Informasi

Sistem informasi adalah sistem yang digunakan oleh organisasi untuk mengelola data dan informasi, mulai dari transaksi harian hingga kegiatan strategis. Sistem informasi ini dapat digunakan untuk mendukung pengambilan keputusan, meningkatkan efisiensi, dan memberikan informasi kepada pihak luar. (Sutabri, 2012).

### 2.3.4 Pengertian Skripsi

Menurut Maryaeni (2009), skripsi adalah karya tulis ilmiah berupa paparan tulisan hasil penelitian sarjana S1 yang membahas suatu permasalahan/ fenomena dalam bidang ilmu tertentu dengan menggunakan kaidah-kaidah yang berlaku.



### 2.3.5 Pengertian Algoritma Ratcliff/Obershelp

Algoritma *Ratcliff/Obershelp* adalah algoritma yang digunakan untuk menghitung kemiripan dua string. Algoritma ini pertama kali diusulkan oleh Dennis Ratcliff dan Tom Obershelp pada tahun 1988. Algoritma *Ratcliff/Obershelp* bekerja dengan cara menghitung panjang substring terpanjang yang sama antara dua string. Panjang substring ini kemudian digunakan untuk menghitung nilai kemiripan antara kedua string. Nilai kemiripan ini dapat dinyatakan dalam persentase.

Konsep pencocokan dari algoritma *Ratcliff/Obershelp* yaitu, pertama menemukan sub string terpanjang yang memiliki kesamaan dari string  $S_1$  dan  $S_2$  yang di sebut *anchor*. Nilai dari  $K_m$  bertambah berdasarkan panjang dari *anchor*. Kemudian bagian yang tersisa dari string sebelah kiri dan kanan dari *anchor* harus diperiksa sebagai string-string yang baru (dengan kata lain mengulangi step 1). Proses tersebut di ulangi sampai semua karakter dari string  $S_1$  dan  $S_2$  di analisa. Algoritma *Ratcliff/Obershelp* dinyatakan dengan rumus:

$$D_{ro} = \frac{2 * K_m}{|S_1| + |S_2|}$$

$K_m$  = Jumlah karakter yang sama

$|S_1|$  = Panjang dari string 1

$|S_2|$  = Panjang dari string 2

Mencari kesamaan kedua string SISTEMINFORMASI dan SYSTEMINFORMATION.

	[1]	[2]	[3]	[4]	[5]	[6]	[7]	[8]	[9]	[10]	[11]	[12]	[13]	[14]	[15]	[16]	[17]
S1	S	I	S	T	E	M	I	N	F	O	R	M	A	S	I		
S2	S	Y	S	T	E	M	I	N	F	O	R	M	A	T	I	O	N

**Gambar 2.1 Sample String**

- Panjang dari string  $S_1$ :  $|S_1| = 15$   
Panjang dari string  $S_2$ :  $|S_2| = 17$
- Substring yang terpanjang yang dimiliki oleh kedua string adalah STEMINFORMA. Maka STEMINFORMA merupakan sebuah *anchor*, dan  $K_m = |\text{STEMINFORMA}| = 11$

	[1]	[2]	[3]	[4]	[5]	[6]	[7]	[8]	[9]	[10]	[11]	[12]	[13]	[14]	[15]	[16]	[17]
S1	S	I	S	T	E	M	I	N	F	O	R	M	A	S	I		
S2	S	Y	S	T	E	M	I	N	F	O	R	M	A	T	I	O	N

**Gambar 2.2 Achor String**

- c) Di sebelah kiri dari *anchor* tersisa kumpulan huruf SI dan SY. Sub string yang terpanjang dan memiliki kesamaan dari kumpulan huruf tersebut adalah S. Maka,  $K_m = 11 + |S| = 12$

	[1]	[2]	[3]	[4]	[5]	[6]	[7]	[8]	[9]	[10]	[11]	[12]	[13]	[14]	[15]	[16]	[17]
S1	S	I	S	T	E	M	I	N	F	O	R	M	A	S	I		
S2	S	Y	S	T	E	M	I	N	F	O	R	M	A	T	I	O	N

**Gambar 2.3 String yang sama pertama**

- d) S merupakan substring yang berada di awal dari kedua string  $S_1$  dan  $S_2$ , disebelah kiri substring tersebut sudah tidak terdapat huruf lagi. Pada sebelah kanan sudah tidak ada huruf lagi.
- e) Maka  $K_m$  tetap sama dan kita lanjutkan pada karakter sebelah kanan dari *anchor*. Disebelah kanan dari *anchor* terdapat karakter SI dan TION. String yang memiliki kesamaan dari kumpulan huruf tersebut adalah I.
- Maka,  $K_m = 12 + |I| = 13$

	[1]	[2]	[3]	[4]	[5]	[6]	[7]	[8]	[9]	[10]	[11]	[12]	[13]	[14]	[15]	[16]	[17]
S1	S	I	S	T	E	M	I	N	F	O	R	M	A	S	I		
S2	S	Y	S	T	E	M	I	N	F	O	R	M	A	T	I	O	N

**Gambar 2.4 String yang sama kedua**

Penilaian *Ratcliff/Obershelp* untuk string SISTEMINFORMASI dan SYSTEMINFORMATION adalah:

$$D_{ro} \frac{2 * K_m}{|S_1| + |S_2|} = \frac{2 * 13}{15 + 17} = \frac{26}{32} = 0,8125$$




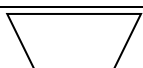

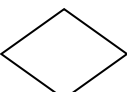
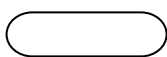
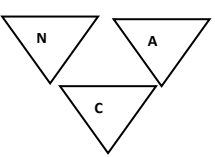
Jadi, dari kedua string SISTEMINFORMASI dan SYSTEMINFORMATION memiliki nilai kesamaan 0,8125 yang bisa dikatakan sama.

### 2.3.6 FOD (Flow Of Document)

*Flow Of Document* (FOD) adalah adalah bagan alir yang menunjukkan arus dari laporan dan formulir termasuk tembusan-tembusannya. FOD merupakan suatu model yang dapat dihubungkan satu sama lain dengan arus data, baik secara manual dan komputerisasi. (Jogiyanto, 2005).

Berikut adalah simbol-simbol yang digunakan dalam menggambar suatu *Flow of Document* (FOD) sebagai berikut:

**Tabel 2.2 Simbol-simbol diagram alir dokumen (Flow of Document)**

Simbol	Nama	Keterangan
	Simbol arus flow	Untuk menyatakan jalannya arus suatu proses.
	Simbol on page connector	Untuk menyatakan sambungan dari satu proses ke proses lainnya dalam halaman atau lembar yang sama.
	Simbol <i>offline connector</i>	Menunjukkan sambungan dari satu proses ke proses lainnya yang terdapat pada halaman atau lembar yang berbeda.
	Simbol manual	Menunjukkan suatu proses yang manual atau tidak dilakukan oleh komputer.
	Simbol <i>Process</i>	Simbol yang menunjukkan pengolahan yang dilakukan oleh komputer
	Simbol keputusan	Simbol yang menunjukkan pemilihan proses berdasarkan kondisi tertentu.
	Simbol Terminal	Menyatakan permulaan atau akhir suatu program.
	Simbol <i>offline storage</i>	Untuk menunjukkan data tersimpan ke suatu media. C (diarsipurut tanggal) A(diarsipurut huruf/ <i>alphabetic</i> ) N (diarsipurut angka/ <i>numerical</i> )

(Bersambung)



Simbol Dokumen

Untuk mencetak laporan.

Sumber: Jogiyanto (2005)

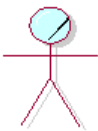
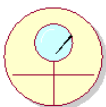
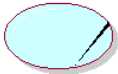

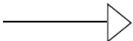
### 2.3.7 UML (Unified Modelling Language)

UML (*Unified Modeling Language*) adalah Salah satu standar bahasa industri yang sering digunakan untuk menentukan persyaratan, melakukan analisis dan desain, dan memberikan deskripsi arsitektur dalam pemrograman berorientasi objek (Rosa dan Shalahuddin, 2018). Bagian - bagian dari UML, yaitu:

#### 1. Business Use Case Diagram

*Business Use Case* adalah model yang digunakan untuk menggambarkan proses bisnis organisasi (Sholiq, 2006). Dengan kata lain *use case* bisnis menggambarkan tentang aktivitas bisnis yang dilakukan organisasi. Simbol-simbol yang digunakan dalam *Business Use Case* Diagram dapat dilihat pada table berikut:

**Tabel 2.3 Simbol Bussiness Use Case Diagram**





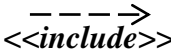
Simbol	Nama	Keterangan
	Aktor Bisnis	Simbol yang menunjukkan seseorang yang ada di luar organisasi yang berinteraksi dengan kegiatan bisnis organisasi.
	Pekerja Bisnis	Simbol yang menunjukkan peranan di dalam organisasi.
	<i>Use Case</i> Bisnis	Model yang menggambarkan proses bisnis organisasi.
	Relasi Asosiasi	Menunjukkan relasi antara aktor bisnis atau pekerja bisnis dengan <i>use case</i> bisnis.
	Relasi <i>Generalization</i>	Digunakan ketika ada dua atau lebih aktor bisnis, pekerja bisnis, atau <i>use case</i> bisnis yang sangat serupa.

Sumber: Sholiq (2006)

## 2. Use Case Diagram

*Use Case* atau diagram *use case* merupakan pemodelan untuk kelakuan (*behaviour*) sistem informasi yang akan dibuat. *Use Case* mendeskripsikan sebuah interaksi antara satu atau lebih aktor dengan sistem informasi yang akan dibuat (Sukamto dan Shalahuddin, 2018). Simbol-simbol yang digunakan dalam *Use Case Diagram* dapat dilihat pada table berikut:

**Tabel 2.4 Notasi Use Case**

Simbol	Nama	Keterangan
	<i>Use case</i>	Nama <i>Use Case</i>
	<i>Actor</i>	Orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat diluar sistem informasi.
	Asosiasi	Komunikasi antara <i>actor</i> dan <i>use case</i> .
	<i>Generalization</i>	Hubungan generalisasi dan spesialisasi (umum-khusus) antara dua buah <i>use case</i> dimana fungsi yang satu adalah fungsi yang lebih umum dari lainnya.
	<i>include</i>	Relasi <i>use case</i> tambahan ke sebuah <i>use case</i> dimana <i>use case</i> yang ditambahkan memerlukan <i>use case</i> ini untuk menjalankan fungsinya.

Sumber: Sukamto dan Shalahuddin (2018)


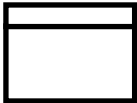


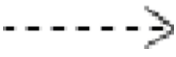

## 3. Class Diagram

*Class Diagram* atau Diagram kelas menggambarkan struktur sistem dari segi pendefinisian kelas - kelas yang akan dibuat untuk membangun sistem. Kelas mempunyai sebuah atribut dan metode atau operasi.

- Atribut adalah variabel-variabel yang dimiliki oleh kelas
- Operasi merupakan fungsi - fungsi yang dimiliki oleh kelas

Simbol-simbol yang digunakan dalam *Class Diagram* dapat dilihat pada table berikut:

**Tabel 2.5 Notasi Class Diagram**

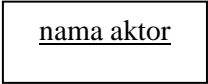

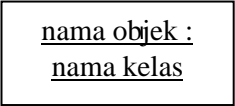

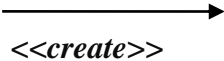


Simbol	Nama	Keterangan
	<i>Generalization</i>	Hubungan dimana objek anak ( <i>descendant</i> ) berbagi perilaku dan struktur data dari objek yang ada di atasnya objek induk ( <i>ancestor</i> ).
	<i>Class</i>	Himpunan dari objek - objek yang berbagi atribut serta operasi yang sama.
	<i>Collaboration</i>	Deskripsi dari urutan aksi-aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur bagi suatu actor.
	<i>Realization</i>	Operasi yang benar - benar dilakukan oleh suatu objek.
	<i>Dependency</i>	Hubungan dimana perubahan yang terjadi pada suatu element mandiri ( <i>independent</i> ) akan mempengaruhi elemen yang bergantung padanya elemen yang tidak mandiri.
	<i>Association</i>	Apa yang menghubungkan antara objek suatu dengan objek yang lain.

Sumber: Sukamto dan Salahuddin (2018)

#### 4. Sequence Diagram

*Sequence diagram* menggambarkan perilaku objek dalam *use case* dengan menggambarkan masa hidup objek dan pesan yang dikirim dan diterima antara objek. Oleh karena itu, untuk menjelaskan diagram sekuen, maka perlu mengetahui objek yang terlibat dalam *use case* dan metode yang termasuk dalam kelas yang dipakai dalam objek tersebut. Pembuatan *sequence diagram* juga diperlukan untuk melihat skenario *use case* (Sukamto dan shalahuddin 2015). Simbol-simbol yang digunakan dalam *Sequence Diagram* dapat dilihat pada table berikut:

**Tabel 2.6 Notasi Sequence Diagram**


Simbol	Nama	Deskripsi
	Aktor	Orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat di luar sistem informasi yang akan dibuat itu sendiri.
	Lifeline	Menyatakan kehidupan suatu objek.
	Objek	Menyatakan objek yang berinteraksi pesan.
	Waktu Aktif	Menyatakan objek dalam keadaan aktif dan berinteraksi pesan.
	Pesan tipe <i>create</i>	Menyatakan suatu objek membuat objek lain, arah panah mengarah pada objek yang dibuat
	Pesan tipe <i>send</i>	Menyatakan bahwa suatu objek mengirimkan data atau masukan.
	Pesan tipe <i>return</i>	Menyatakan bahwa suatu objek yang telah menjalankan suatu operasi atau metode.

Sumber: Sukamto dan Shalahuddin (2018)

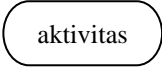
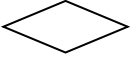


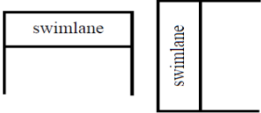
## 5. Activity Diagram

Diagram aktivitas atau *Activity Diagram* menggambarkan alur kerja atau aktivitas sistem atau proses bisnis atau menu yang terdapat dalam perangkat lunak. Penting untuk dicatat bahwa diagram aktivitas mewakili aktivitas sistem bukan apa yang dilakukan aktor, yaitu aktivitas yang dapat dilakukan pada sistem (Sukamto dan Salahuddin 2018). Simbol - simbol yang digunakan dalam *Activity Diagram* dapat dilihat pada table berikut:

**Tabel 2.7 Notasi Activity Diagram**

Simbol	Nama Simbol	Deskripsi
	Status awal	Status awal aktivitas sistem, diagram aktivitas memiliki

(Bersambung)




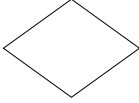

		sebuah status awal.
	Aktivitas	Aktivitas yang dilakukan sistem, aktivitas biasanya diawali dengan kata kerja.
	Percabangan / <i>decision</i>	Asosiasi percabangan dimana jika ada pilihan aktivitas lebih dari satu.
	Penggabungan / <i>join</i>	Asosiasi penggabungan dimana lebih dari satu aktivitas digabungkan menjadi satu.
	Status akhir	Status akhir yang dilakukan sistem, sebuah diagram aktivitas memiliki sebuah status akhir.
	Swimlane	Memisahkan organisasi bisnis yang bertanggung jawab terhadap aktivitas terjadi.

Sumber: Sukamto dan Shalahuddin (2018)

## 6. Statechart Diagram

*Statechart Diagram* digunakan untuk menggambarkan perubahan status atau transisi dari sebuah mesin atau sistem. Simbol - simbol yang digunakan dalam *Statechart Diagram* dapat dilihat pada table berikut:

**Tabel 2.8 Notasi Statechart Diagram**

Simbol	Nama	Keterangan
	<i>Start (Initial State)</i>	<i>Start</i> atau <i>initial state</i> adalah <i>state</i> atau keadaan awal pada saat sistem mulai hidup
	<i>End (Final State)</i>	<i>End</i> atau <i>final state</i> adalah <i>state</i> atau keadaan akhir dari daur hidup suatu sistem
	<i>Event</i>	<i>Event</i> adalah kegiatan yang menyebabkan berubah nya status mesin
	<i>Decision</i>	Digunakan untuk menggambarkan suatu keputusan / tindakan yang harus diambil pada kondisi tertentu
	<i>State</i>	<i>State</i> atau status adalah keadaan sistem pada waktu tertentu. <i>State</i> dapat berubah jika ada <i>event</i> tertentu yang memicu perubahan tersebut

Sumber: Sukamto dan Shalahuddin (2011)



## **BAB III**

### **METODOLOGI PENELITIAN**

#### **3.1 Objek Penelitian**

##### **3.1.1 Sekilas Tentang Program Studi Sistem Informasi**

Program Studi Sistem Informasi Fakultas Teknik Universitas Muria Kudus (FT UMK) didirikan berdasarkan kebutuhan tenaga kerja di bidang teknologi informasi di wilayah Jawa Tengah dan sekitarnya pada tahun 2002 melalui Surat Keputusan Direktorat Jenderal Pendidikan Perguruan Tinggi (Dirjen DIKTI) Nomor: 1324/D/T/2002. Pengelolaannya secara langsung berada di bawah Fakultas Teknik. Pada tahun 2004, berhasil memperoleh perpanjangan izin penyelenggaraan berdasarkan Surat Keputusan Dirjen DIKTI Nomor: 4563/D/T/2004.

Pada tahun 2006, Program Studi Sistem Informasi FT UMK meraih akreditasi tingkat C berdasarkan Surat Badan Akreditasi Nasional Perguruan Tinggi Nomor: 019/BAN-PT/Ak-XSI/XII/2006. Oleh karena itu, pada tahun 2022, Program Studi Sistem Informasi FT UMK sudah berusia dua puluh tahun dengan slogan “Dua Puluh Tak Mengenal Rapuh” pada usia dua puluh tahun ini diharapkan Program Studi Sistem Informasi Semakin Maju, Semakin Ampuh Tanpa Mengenal Rapuh Walau Berkucuran Peluh.

Dalam menjalankan proses pengajaran Program Studi Sistem Informasi, digunakan pendekatan Pembelajaran Berpusat pada Mahasiswa di mana mahasiswa didorong untuk menjadi lebih proaktif dalam mencapai tujuan pembelajaran. Pendekatan ini diterapkan melalui berbagai kegiatan seperti diskusi, tugas kelompok, dan penyelenggaraan sidang mahasiswa. Proses pembelajaran berlangsung selama setidaknya 14 pertemuan setiap semester, termasuk Ujian Tengah Semester dan Ujian Akhir Semester.

##### **3.1.2 Visi dan Misi**

###### **1. Visi**

Menjadi Program Studi yang Unggul di bidang Sistem Informasi,  
Berbasis Kearifan Lokal serta Berdaya Saing Global

###### **2. Misi**

- a. Menyelenggarakan dan mengembangkan pendidikan tinggi yang unggul dibidang sistem informasi bermuatan softskill dengan menerapkan kurikulum perguruan tinggi berbasis KKNI, kearifan lokal serta berdaya saing global.
- b. Menyelenggarakan dan mengembangkan penelitian yang lebih inovatif di bidang sistem informasi berbasis kearifan lokal untuk masyarakat dan.
- c. Menyelenggarakan pengabdian kepada masyarakat di bidang sistem informasi berbasis kearifan lokal dengan melibatkan partisipasi aktif civitas akademik dan masyarakat.
- d. Membangun dan mengembangkan kerjasama di bidang sistem informasi dengan institusi di dalam dan di luar negeri untuk menunjang penguatan program studi Unggul yang berbasis kearifan lokal dan berdaya saing global.
- e. Menyelenggarakan tata kelola program studi yang kredibel, transparan, akuntabel, bertanggungjawab dan berkeadilan.

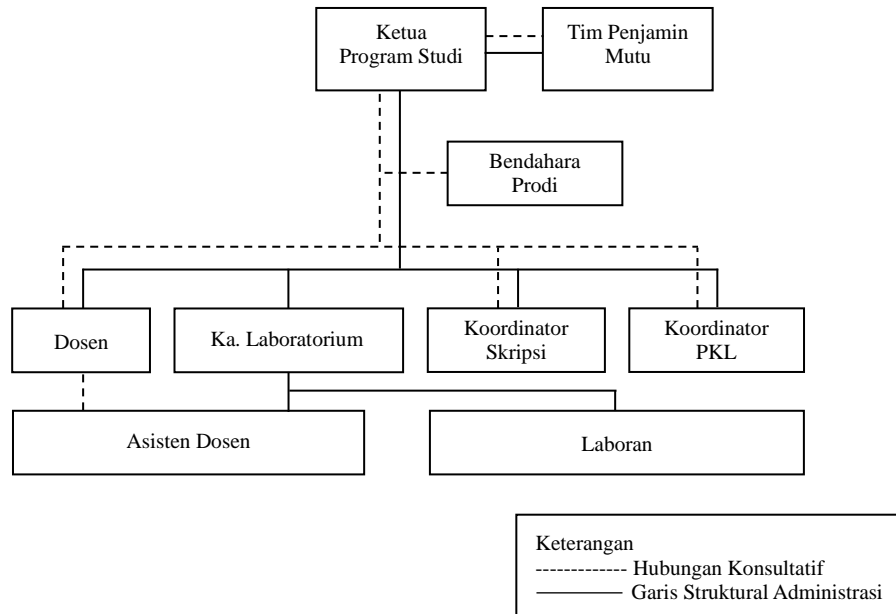
### 3.1.3 Denah Lokasi



**Gambar 3.1 Denah Lokasi Program Studi Sistem Informasi  
Universitas Muria Kudus**

### 3.1.4 Struktur Organisasi

Berikut adalah Gambar 3.2, struktur organisasi di Program Studi Sistem Informasi Universitas Muria Kudus:



**Gambar 3.2 Struktur Organisasi Program Studi Sistem Informasi Universitas Muria Kudus**

### 3.1.5 Deskripsi Tugas (Job Description)

1. Ketua Program Studi
  - a. Merencanakan dan mengkoordinasikan program akademik.
  - b. Mengelola dan mengawasi kebijakan dan prosedur di dalam program studi.
  - c. Memimpin rapat-rapat internal program studi.
  - d. Berkomunikasi dengan dosen dan mahasiswa untuk meningkatkan kualitas program.
2. Tim Penjaminan Mutu
  - a. Memantau dan mengevaluasi mutu program studi.
  - b. Mengembangkan dan mengevaluasi prosedur penjaminan mutu.
  - c. Menanggapi umpan balik dari mahasiswa dan dosen.
  - d. Melakukan audit internal dan eksternal untuk menilai kualitas program studi.

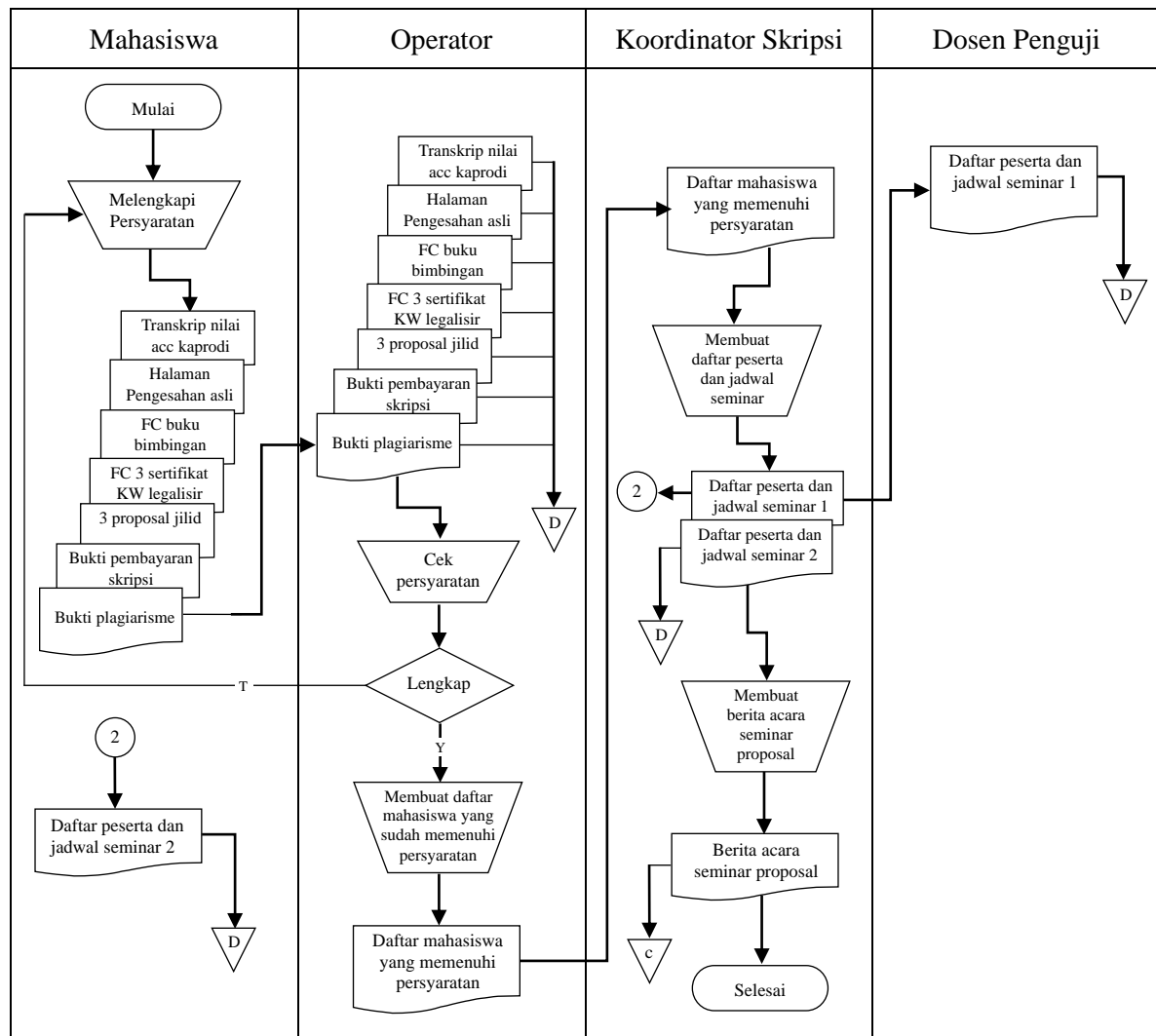
3. Bendahara Program Studi
  - a. Mengelola anggaran program studi.
  - b. Merekam dan melaporkan transaksi keuangan.
  - c. Mengelola pembayaran dan pengeluaran program studi.
4. Koordinator Praktek Kerja Lapangan
  - a. Mengkoordinasikan penempatan mahasiswa untuk praktek kerja lapangan.
  - b. Memastikan kesesuaian antara kegiatan praktek dan kurikulum.
  - c. Menilai laporan praktek kerja mahasiswa.
5. Koordinator Skripsi
  - a. Membimbing mahasiswa dalam penyusunan skripsi.
  - b. Mengorganisir dan memantau proses pengembangan skripsi.
  - c. Menilai dan memberikan umpan balik pada skripsi mahasiswa.
6. Kepala Laboratorium
  - a. Mengelola operasional laboratorium.
  - b. Memastikan keamanan dan ketersediaan peralatan laboratorium.
  - c. Mengelola staf laboratorium dan memberikan pelatihan jika diperlukan.
7. Dosen
  - a. Mengajar mata pelajaran sesuai dengan bidang keahliannya.
  - b. Melakukan penelitian dan publikasi ilmiah.
  - c. Memberikan bimbingan akademis kepada mahasiswa.
  - d. Berpartisipasi dalam pengembangan kurikulum.
8. Asisten Dosen
  - a. Membantu dosen dalam kegiatan pengajaran.
  - b. Memberikan dukungan administratif kepada dosen.
  - c. Mungkin terlibat dalam membimbing mahasiswa.
9. Laboran
  - a. Menangani perawatan dan pengelolaan peralatan laboratorium.
  - b. Membantu dalam persiapan dan pelaksanaan praktikum.
  - c. Memberikan dukungan teknis kepada dosen dan mahasiswa dalam penggunaan fasilitas laboratorium.

## **3.2 Analisa Sistem Lama**

### **3.2.1 Prosedur Pendaftaran Seminar Proposal Skripsi**

- a. Mahasiswa mengajukan dokumen persyaratan seminar proposal skripsi kepada Operator, termasuk tiga eksemplar proposal skripsi yang sudah dijilid, transkrip nilai yang sudah ditandatangani oleh Ketua Program Studi Sistem Informasi, salinan buku bimbingan, fotokopi sertifikat tiga Keterampilan Wajib, halaman pengesahan asli, bukti tidak plagiat, dan tanda bukti pembayaran skripsi.
- b. Operator akan memeriksa kelengkapan persyaratan tersebut. Apabila ternyata masih ada yang kurang, persyaratan akan dikembalikan kepada mahasiswa untuk melengkapinya.
- c. Apabila semua persyaratan sudah terpenuhi dan pendaftaran seminar proposal skripsi telah ditutup, Operator akan menyusun daftar mahasiswa yang telah melengkapinya dan kemudian menyerahkan daftar tersebut kepada Koordinator Skripsi.
- d. Koordinator Skripsi menyusun daftar peserta serta menetapkan jadwal seminar proposal skripsi.
- e. Jadwal seminar yang sudah disusun akan dicetak dalam tiga salinan. Salinan pertama akan diarsip oleh Koordinator Skripsi, salinan kedua akan diberikan kepada dosen penguji yang bersangkutan, dan salinan ketiga akan diberikan kepada mahasiswa.
- f. Koordinator Skripsi membuat berita acara seminar proposal.

Berikut adalah Gambar 3.3 aliran dokumen prosedur pendaftaran seminar proposal skripsi.



**Gambar 3.3 Aliran Dokumen Pendaftaran Seminar Proposal**

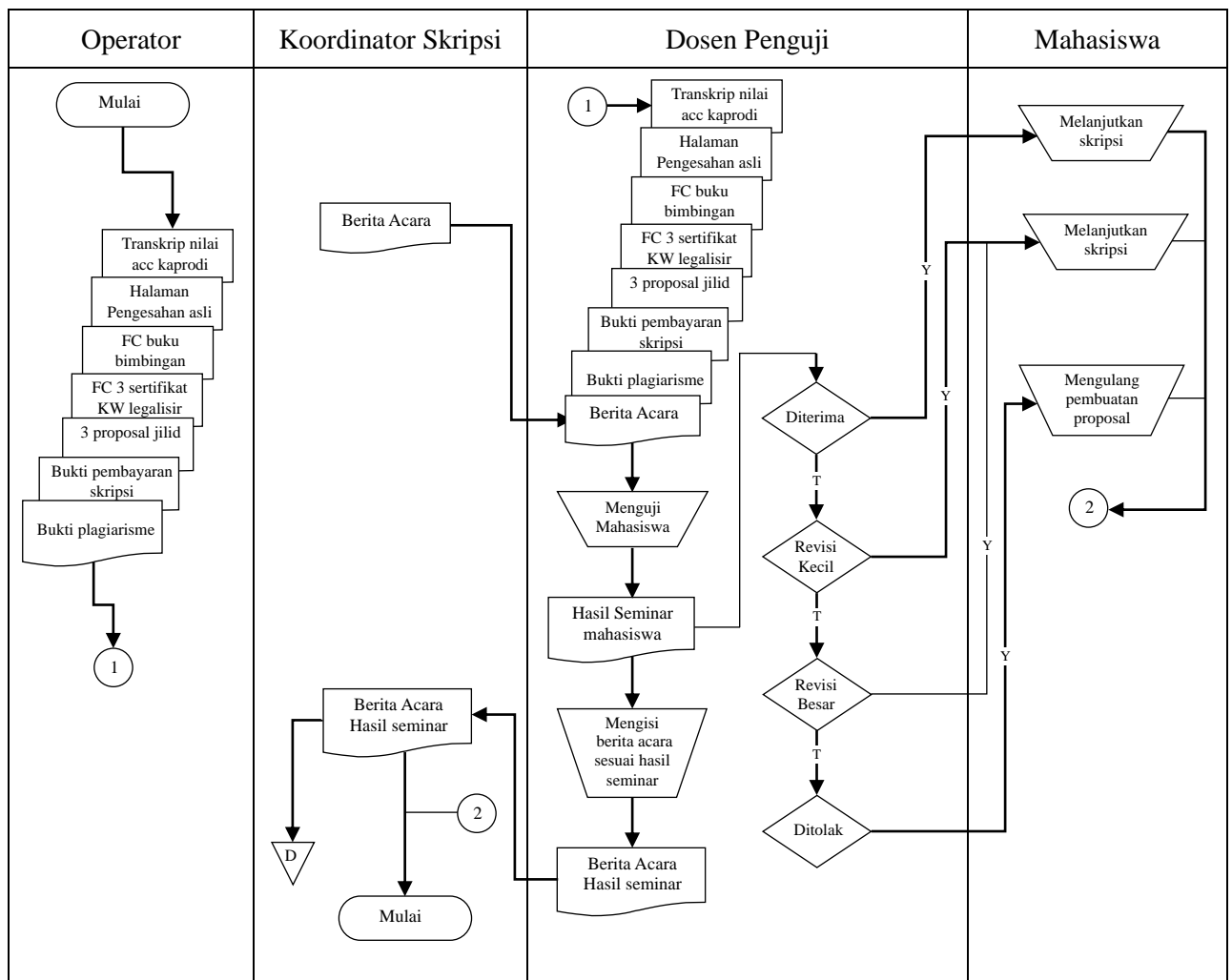
### 3.2.2 Prosedur Pelaksanaan Seminar Proposal Skripsi

- Koordinator Skripsi membuat berita acara seminar proposal.
- Koordinator mengirimkan berita acara sidang proposal kepada dosen yang akan menguji.
- Sidang proposal dilaksanakan oleh mahasiswa.
- Setelah sidang selesai, dosen yang bertugas menilai kelayakan sidang proposal akan memilih: diterima langsung, revisi kecil, revisi besar, atau ditolak dan masukan di Berita Acara.
- Jika isian diterima langsung, maka mahasiswa dapat melanjutkan pengerjaan skripsi tanpa bantuan. Jika isian revisi kecil, maka

mahasiswa dapat melanjutkan pengerjaan skripsi dengan bimbingan. Jika isian revisi besar, maka mahasiswa harus mengikuti seminar ulang. Jika isian ditolak, maka mahasiswa harus memulai kembali proposal dari awal.

- f. Setelah sidang proposal selesai, dosen penguji akan mengirimkan Berita Acara kepada Koordinator Skripsi.

Berikut adalah gambar 3.4 aliran dokumen dalam prosedur pelaksanaan seminar proposal skripsi.



**Gambar 3.4 Aliran Dokumen Pelaksanaan Sidang Proposal Skripsi**

### 3.3 Analisa dan Rancangan Sistem Baru

Penulisan skripsi di program studi Sistem Informasi di Universitas Muria Kudus merupakan tahap krusial dalam menyelesaikan program sarjana, dengan seminar proposal sebagai salah satu tahapan penting. Namun, seringkali mahasiswa mengalami kesulitan dalam menentukan judul skripsi yang sesuai dengan minat dan kemampuan mereka, serta mencegah kemiripan judul dengan penelitian sebelumnya. Oleh karena itu, diperlukan sebuah aplikasi yang menggunakan algoritma *Ratcliff/Obershelp* untuk membandingkan judul skripsi yang diajukan dengan database judul skripsi yang sudah ada. Aplikasi ini akan membantu mahasiswa mendapatkan rekomendasi judul yang unik dan menghindari duplikasi penelitian. Dengan demikian, sistem ini diharapkan dapat mendorong variasi topik penelitian yang lebih luas dan mengurangi risiko duplikasi penelitian di bidang Sistem Informasi.

#### 3.3.1 Analisa Kebutuhan

##### 1. Kebutuhan Perangkat Keras (Hardware)

Perangkat keras yang dibutuhkan untuk mengimplementasikan sistem ini minimal harus memiliki spesifikasi sebagai berikut :

- a. Monitor berwarna dan mampu beresolusi 1024 x 800 pixel, bisa dengan berbagai bentuk dan merk.
- b. CPU (*Central Processing Unit*), terdiri dari *prosesor*, *motherboard*, *harddisk*, RAM, VGA, dll.
- c. Jaringan WiFi.
- d. Perangkat pendukung lain seperti *printer*, *mouse*, *keyboard*, *scanner*, dll.

##### 2. Kebutuhan Perangkat Lunak (Software)

Perangkat lunak yang digunakan untuk mengembangkan sistem informasi ini adalah sebagai berikut :

- a. Sistem operasi, *windows* atau *linux*.
- b. Android Studio digunakan untuk menulis code program.
- c. *Web server* (Apache) dan database MYSQL dimana kedua software tersebut sudah tersedia didalam paket *Xampp*.
- d. Web browser untuk menjalankan *web* seperti *Mozilla Firefox*, *Google Chrome*, *Opera*, dll.



### 3. Kebutuhan Pengguna (*Brainware*)

Pengguna yang terlibat dalam sistem yang akan dibuat yaitu:

- a. Koordinator Skripsi,
- b. Operator,
- c. Dosen,
- d. Mahasiswa.

#### 3.3.2 Rancangan Sistem Baru

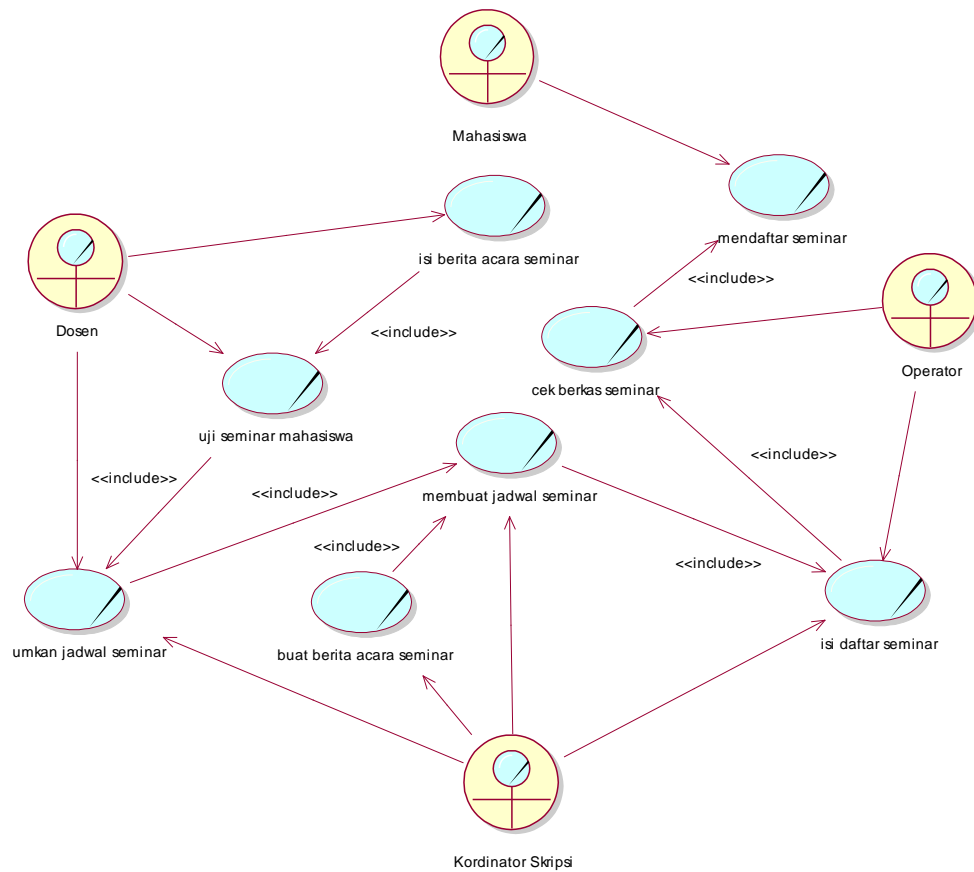
##### 1. Use Case Diagram

Analisa aktor sistem yang akan menjalankan sistem tersebut adalah:

- a. Koordinator Skripsi, aktor yang dapat melakukan pembagian penjadwalan, dan hasil seminar.
- b. Operator, aktor yang menangani pendaftaran seminar berupa pengecekan berkas pendaftaran seminar.
- c. Dosen, aktor yang menyetujui pengajuan judul oleh mahasiswa serta melakukan penilaian seminar.
- d. Mahasiswa, aktor yang dapat melakukan pendaftaran seminar proposal dengan mengunggah berkas dan dapat mengecek kesamaan judul skripsi.

##### 2. *Bussiness Use Case Diagram*

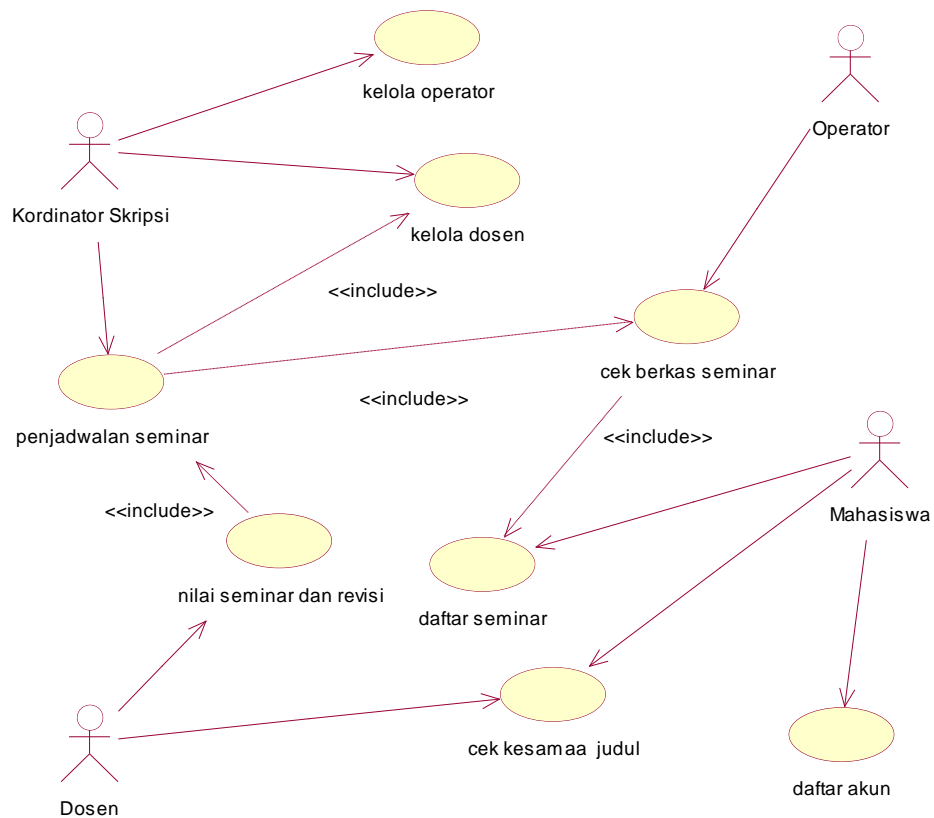
Diagram *Use Case* Bisnis menguraikan tentang relasi antara para aktor dalam proses bisnis yang terjadi baik di dalam maupun di luar suatu organisasi. Ilustrasi dari proses bisnis tersebut dapat diperlihatkan pada gambar 3.5.



**Gambar 3.5 Bussiness Use Case Diagram**

### 3. System Use Case Diagram

Diagram Use Case mengilustrasikan interaksi atau keterkaitan antara satu atau lebih aktor dengan sistem informasi yang sedang direncanakan untuk dibangun di dalam organisasi. Representasi dari use case sistem ini bisa dilihat pada gambar 3.6.



**Gambar 3.6 System Use Case Diagram**

#### 4. Skenario Use Case (Flow of Event)

Skenario use case memberikan penjelasan mengenai tindakan yang akan diambil oleh sistem. Rincian perilaku yang terkandung dalam *use case* diuraikan dalam alur kejadian. Alur kejadian bertujuan untuk mendokumentasikan aktivitas logis yang terjadi di dalam *use case*. Berdasarkan *system use case* yang telah disebutkan, narasi skenario yang dapat dibuat adalah sebagai berikut.

a. Deskripsi *Use Case* Kelola Data Operator

**Tabel 3.1 . Deskripsi Use Case Kelola Data Operator**

<p><b>Aktor Utama :</b> Koordinator Skripsi</p> <p><b>Nama Use Case :</b> kelola data operator</p> <p><b>Tujuan Use Case :</b> mengelola data operator meliputi tambah, hapus, dan ubah</p> <p><b>Kondisi Awal :</b> sistem menampilkan tampilan daftar data operator</p> <p><b>Kondisi Akhir :</b> tampil data operator</p>	<p><b>Alur <i>Optimistic</i> :</b></p> <ol style="list-style-type: none"> <li>1. Koordinator Skripsi melakukan login,</li> <li>2. Memilih menu data operator,</li> <li>3. Mahasiswa melakukan pendaftaran</li> <li>4. Koordinator Skripsi dapat melakukan penambahan, pengubahan, dan penghapusan data.</li> </ol> <p><b>Alur <i>Pesimistic</i></b></p> <p>1.a. Login tidak berhasil</p> <p><b>Solusi</b></p> <p>1.a. Login kembali dengan memasukkan username dan password dengan benar.</p>
--	---

b. Deskripsi *Use Case* Kelola Data Dosen

**Tabel 3.2 Deskripsi Use Case Kelola Data Dosen**

<p><b>Aktor Utama :</b> Koordinator Skripsi</p> <p><b>Nama Use Case :</b> kelola data dosen</p> <p><b>Tujuan Use Case :</b> mengelola data Dosen meliputi tambah, hapus, dan ubah</p> <p><b>Kondisi Awal :</b> sistem menampilkan tampilan daftar data dosen</p> <p><b>Kondisi Akhir :</b> tampil data dosen</p>	<p><b>Alur <i>Optimistic</i> :</b></p> <ol style="list-style-type: none"> <li>1. Koordinator Skripsi melakukan login,</li> <li>2. Memilih menu data dosen,</li> <li>3. Mahasiswa melakukan pendaftaran</li> <li>4. Koordinator Skripsi dapat melakukan penambahan, pengubahan, dan penghapusan data.</li> </ol> <p><b>Alur <i>Pesimistic</i></b></p> <p>1.a. Login tidak berhasil</p> <p><b>Solusi</b></p> <p>1.a. Login kembali dengan memasukkan username dan password dengan benar.</p>
--	--

c. Deskripsi Use Case Daftar Akun

**Tabel 3.3 Deskripsi Use Case Daftar Akun**

<p><b>Aktor Utama :</b> Mahasiswa</p> <p><b>Nama Use Case :</b> daftar akun</p> <p><b>Tujuan Use Case :</b> mendaftar akun mahasiswa</p> <p><b>Kondisi Awal :</b> sistem menampilkan tampilan pendaftaran akun</p> <p><b>Kondisi Akhir :</b> tampilan menu login</p>	<p><b>Alur Optimistic :</b></p> <ol style="list-style-type: none"> <li>5. Mahasiswa melakukan login,</li> <li>6. Memilih menu data pendaftaran,</li> <li>7. Mahasiswa melakukan pendaftaran</li> </ol> <p><b>Alur Pesimistic</b></p> <p>1.a. Pendaftaran Gagal</p> <p><b>Solusi</b></p> <p>1.a. Melakukan pendaftaran kembali dengan mengisi data yang benar</p>
--	--

d. Deskripsi Use Case Daftar Seminar

**Tabel 3.4 Deskripsi Use Case Daftar Seminar**

<p><b>Aktor Utama :</b> Mahasiswa</p> <p><b>Nama Use Case :</b> daftar seminar</p> <p><b>Tujuan Use Case :</b> mendaftar seminar melalui sistem</p> <p><b>Kondisi Awal :</b> sistem menampilkan tampilan pendaftaran seminar berupa form unggah berkas</p> <p><b>Kondisi Akhir :</b> tampilan pendaftaran berhasil</p>	<p><b>Alur Optimistic :</b></p> <ol style="list-style-type: none"> <li>1. Mahasiswa melakukan login,</li> <li>2. Memilih menu daftar seminar,</li> <li>3. Mahasiswa melakukan imputan yang ditentukan berupa berkas</li> <li>4. Mahasiswa tekan tombol daftar</li> </ol> <p><b>Alur Pesimistic</b></p> <p>1.a. Login tidak berhasil</p> <p>4.a. Tidak tersimpan</p> <p><b>Solusi</b></p> <p>1.a. Login kembali dengan memasukkan username dan password dengan benar</p> <p>4.a. Mengisi kembali data yang diperlukan sesuai format klik enter</p>
--	---

e. Deskripsi Use Case Cek Berkas Pendaftaran Seminar

**Tabel 3.5 Deskripsi Use Case Cek Berkas Pendaftaran Seminar**

<p><b>Aktor Utama :</b> Operator</p> <p><b>Nama Use Case :</b> Cek Berkas Pendaftaran</p> <p><b>Tujuan Use Case :</b> memeriksa kelengkapan atau syarat pendaftaran seminar yang diajukan mahasiswa</p> <p><b>Kondisi Awal :</b> tampilan daftar mahasiswa yang mendaftar seminar</p> <p><b>Kondisi Akhir :</b> tampilan data mahasiswa sudah dikonfirmasi</p>	<p><b>Alur <i>Optimistic</i> :</b></p> <ol style="list-style-type: none"> <li>1. Operator melakukan login,</li> <li>2. Operator memilih mahasiswa yang akan diperiksa dan dikonfirmasi,</li> <li>3. Mahasiswa melakukan imputan yang ditentukan berupa berkas</li> </ol> <p><b>Alur <i>Pesimistic</i></b></p> <p>1.a. Login tidak berhasil</p> <p><b>Solusi</b></p> <p>1.a. Login kembali dengan memasukkan username dan password dengan benar</p>
--	--

f. Deskripsi Use Case Penjadwalan Seminar

**Tabel 3.6 Deskripsi Use Case Penjadwalan Seminar**

<p><b>Aktor Utama :</b> Koordinator atau Dosen</p> <p><b>Nama Use Case :</b> penjadwalan seminar</p> <p><b>Tujuan Use Case :</b> membuat jadwal dan peserta seminar</p> <p><b>Kondisi Awal :</b> sistem menampilkan menu penjadwalan seminar</p> <p><b>Kondisi Akhir :</b> tampilan jadwal seminar</p>	<p><b>Alur <i>Optimistic</i> :</b></p> <ol style="list-style-type: none"> <li>1. Koordinator skripsi melakukan login,</li> <li>2. Memilih dan mengisi menu penjadwalan seminar,</li> <li>3. Koordinator skripsi tekan tombol simpan</li> <li>4. Dosen melakukan login</li> <li>5. Memilih dan mengecek menu penilaian seminar,</li> </ol> <p><b>Alur <i>Pesimistic</i></b></p> <p>1.a. Login tidak berhasil</p> <p>6.a. Tidak tersimpan</p> <p>8.a. Jadwal belum tampil</p>
--	---

(Bersambung)

Tabel 3.6 Lanjutan

	<p><b>Solusi</b></p> <p>1.a. Login kembali dengan memasukkan username dan password dengan benar.</p> <p>6.a. mengisi kembali data yang diperlukan sesuai format tekan daftar.</p> <p>7.a. Menunggu Koordinator melakukan penjadwalan.</p>
--	---

g. Deskripsi *Use Case* Cek Kesamaan Judul

**Tabel 3.7 Deskripsi Use Case Cek Kesamaan Judul**

<p><b>Aktor Utama :</b> Dosen dan Mahasiswa</p> <p><b>Nama Use Case :</b> cek kesamaan judul</p> <p><b>Tujuan Use Case :</b> mengetahui seberapa mirip dengan judul skripsi penelitian sebelumnya</p> <p><b>Kondisi Awal :</b> tampilan daftar mahasiswa yang mendaftar seminar</p> <p><b>Kondisi Akhir :</b> tampilan hasil kesamaan</p>	<p><b>Alur <i>Optimistic</i> :</b></p> <ol style="list-style-type: none"> <li>1. Mahasiswa melakukan login,</li> <li>2. Memilih menu kesamaan judul skripsi,</li> <li>3. Mahasiswa mengisi inputan berupa judul skripsi.</li> <li>4. Mahasiswa klik cek</li> </ol> <p><b>Alur <i>Pesimistic</i></b></p> <p>1.a. Login tidak berhasil</p> <p><b>Solusi</b></p> <p>1.a. Login kembali dengan memasukkan username dan password dengan benar.</p>
---	---

h. Deskripsi *Use Case* Nilai Seminar dan Revisi

**Tabel 3.8 Deskripsi Use Case Nilai Seminar dan Revisi**

<p><b>Aktor Utama :</b> Dosen</p> <p><b>Nama Use Case :</b> nilai seminar dan revisi</p> <p><b>Tujuan Use Case :</b> memberi penilaian seminar dan revisi</p>	<p><b>Alur <i>Optimistic</i> :</b></p> <ol style="list-style-type: none"> <li>1. Dosen melakukan login,</li> <li>2. Memilih menu penilaian seminar,</li> <li>3. Dosen mengisi inputan berupa hasil dan revisi seminar.</li> </ol>
---	---

(Bersambung)

Tabel 3.8 Lanjutan

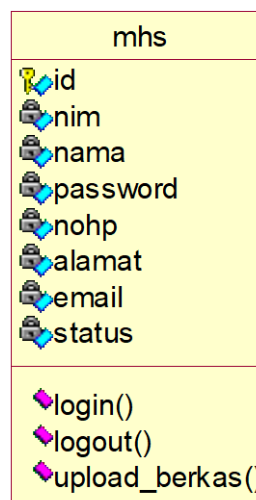
<p><b>Kondisi Awal :</b> tampilan daftar mahasiswa yang mendaftar seminar</p> <p><b>Kondisi Akhir :</b> tampilan hasil seminar</p>	<p>4. Dosen klik simpan</p> <p><b>Alur <i>Pesimistic</i></b></p> <p>1.a. Login tidak berhasil</p> <p>4.a. Tidak Tersimpan</p> <p><b>Solusi</b></p> <p>1.a. Login kembali dengan memasukkan username dan password dengan benar.</p> <p>4.a. mengisi kembali data yang diperlukan sesuai format kemudian tekan daftar.</p>
--	--

## 5. Class Diagram

Diagram kelas merupakan representasi visual yang menggambarkan berbagai kelas dalam suatu sistem atau perangkat lunak yang sedang dalam proses pengembangan. Diagram ini menunjukkan struktur dari sistem atau perangkat lunak tersebut beserta hubungan antar kelas yang terlibat. Berikut adalah kelas-kelas dari sistem yang akan dibuat.

### a. Kelas Mhs

Kelas mhs dirancang untuk menyimpan atribut - atribut dari mahasiswa . Gambaran dari kelas mahasiswa ini dapat dilihat pada gambar nomor 3.7.

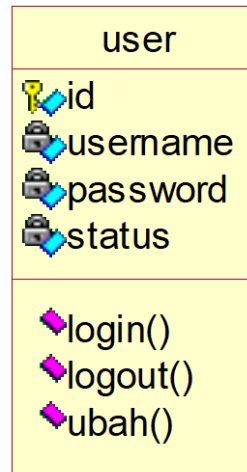


Gambar 3.7 Kelas Mhs



b. Kelas *User*

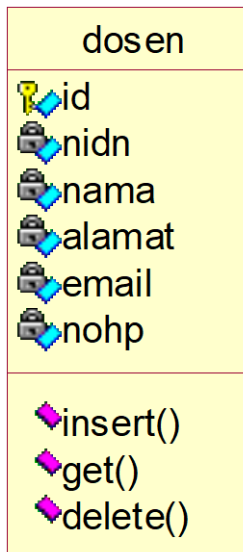
Kelas *user* dirancang untuk menyimpan atribut - atribut dari dosen. Gambaran dari kelas *user* ini dapat dilihat pada gambar nomor 3.8.



**Gambar 3.8 Kelas *User***

c. Kelas Dosen

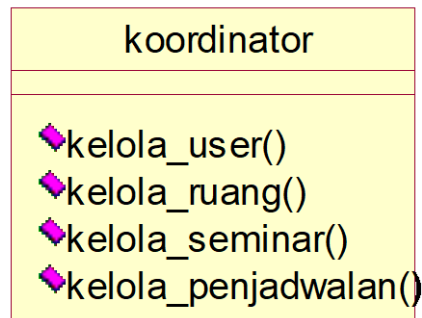
Kelas dosen dirancang untuk menyimpan atribut - atribut dari dosen. Kelas dosen memiliki kelas turunan yang meliputi kelas koordinator dan kelas operator. Gambaran dari kelas dosen ini dapat dilihat pada gambar nomor 3.9.



**Gambar 3.9 Kelas Dosen**

d. Kelas Koordinator

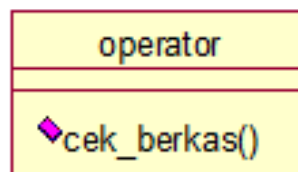
Kelas koordinator adalah kelas turunan dari kelas dosen. Gambaran dari kelas koordinator ini dapat dilihat pada gambar nomor 3.10.



**Gambar 3.10 Kelas Koordinator**

e. Kelas Operator

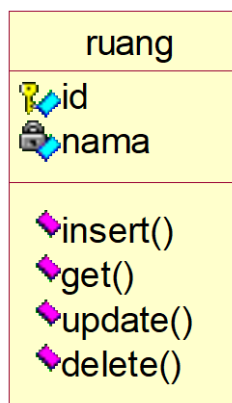
Kelas operator adalah kelas turunan dari kelas dosen. Gambaran dari kelas operator ini dapat dilihat pada gambar nomor 3.11.



**Gambar 3.11 Kelas Operator**

f. Kelas Ruang

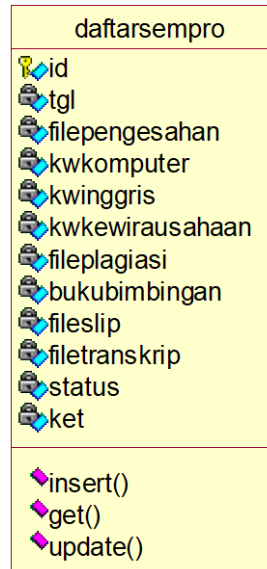
Kelas user dirancang untuk menyimpan atribut - atribut dari dosen. Gambaran dari kelas user ini dapat dilihat pada gambar nomor 3.8.



**Gambar 3.12 Kelas Ruang**

g. Kelas Daftar Sempro

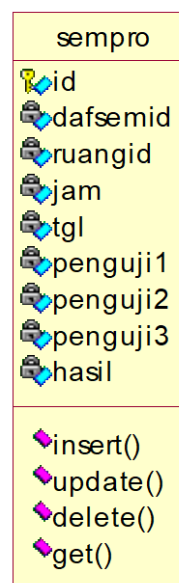
Kelas `daftarsempro` digunakan untuk yang mendeskripsikan berbagai atribut dan metode yang berkaitan dengan pendaftaran seminar proposal. Informasi lebih lanjut mengenai kelas ini dapat ditemukan pada Gambar 3.13.



**Gambar 3.13 Kelas Daftar Sempro**

h. Kelas Sempro

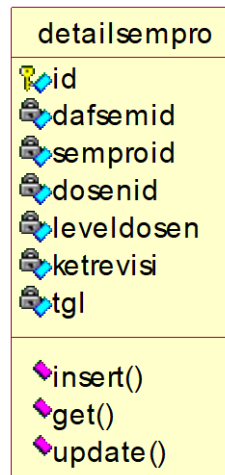
Kelas `sempro` digunakan untuk menggambarkan atribut dan method yang ada pada seminar proposal. Kelas `sempro` dapat dilihat pada gambar 3.14.



**Gambar 3.14 Kelas Sempro**

i. Kelas Detail Sempro

Kelas detailsempro digunakan untuk menggambarkan atribut dan method yang ada pada hasil seminar proposal. Kelas detailsempro dapat dilihat pada gambar 3.15



**Gambar 3.15 Kelas Detail Sempro**

Dari beberapa kelas yang telah didefinisikan di atas, diagram yang dapat dibuat adalah sebagai gambar 3.17 berikut:



