**Perancangan Sistem Informasi**

**Pengelolaan Data Karya Ilmiah/Skripsi Mahasiswa**

**Menggunakan Metode Unifed Modelling Language(UML)**

**TUGAS PRAKTIKUM**

Disusun Oleh:

|  |  |
| --- | --- |
| **Hendro** | **3311811009** |
| **Hendra Febrian** | **3311811023** |
| **Nadya Alisa Sesiq Millenia Kasim** | **3311811039** |



Disusun untuk memenuhi salah satu

Syarat kelulusan matakuliah IF312

Rekayasa Perangkat Lunak II

**PROGRAM STUDI TEKNIK INFORMATIKA**

**JURUSAN TEKNIK INFORMATIKA**

**POLITEKNIK NEGERI BATAM**

**BATAM 2019**

# HALAMAN PENGESAHAN

**Sistem Pengelolaan Data Karya Ilmiah / Skripsi Mahasiswa**

**Menggunakan Metode Unified Modelling Language(UML)**

**Disusun oleh:**

|  |  |
| --- | --- |
| **Hendro** | **3311811009** |
| **Hendra Febrian** | **3311811023** |
| **Nadya Alisa Sesiq Millenia Kasim** | **3311811039** |

Batam, 29 Oktober 2019

Disetujui dan disahkan oleh:

Dosen Pengajar,

Rina Yulius

NIK/NIP:\_\_\_\_\_\_\_\_\_

# HALAMAN PERNYATAAN

Dengan ini, saya:

NIM : 3311811009

Nama : Hendro

adalah mahasiswa Program Studi Teknik Informatika Politeknik Negeri Batam menyatakan bahwa Tugas Praktikum dengan judul:

**Perancangan Sistem Informasi Pengelolaan Data Karya Ilmiah/Skripsi Mahasiswa Menggunakan Metode Unified Modelling Language(UML)**

disusun dengan:

1. tidak melakukan plagiat terhadap naskah karya oranglain
2. tidak melakukan pemalsuandata
3. tidak menggunakan karya orang lain tanpa menyebut sumber asli atau tanpa ijin pemilik

Jika kemudian terbukti terjadi pelanggaran terhadap pernyataan di atas, maka saya bersedia menerima sanksi apapun termasuk pencabutan gelar akademik.

Lembar pernyataan ini juga memberikan hak kepada Politeknik Negeri Batam untuk mempergunakan, mendistribusikan ataupun memproduksi ulang seluruh hasil Tugas Praktikum ini.

Batam, 29 Oktober 2019

**Hendro**

**NIM: 3311811009**

# HALAMAN PERNYATAAN

Dengan ini, saya:

NIM : 3311811023

Nama : Hendra Febrian

adalah mahasiswa Program Studi Teknik Informatika Politeknik Negeri Batam menyatakan bahwa Tugas Praktikum dengan judul:

**Perancangan Sistem Informasi Pengelolaan Data Karya Ilmiah/Skripsi Mahasiswa Menggunakan Metode Unified Modelling Language(UML)**

disusun dengan:

1. tidak melakukan plagiat terhadap naskah karya oranglain
2. tidak melakukan pemalsuandata
3. tidak menggunakan karya orang lain tanpa menyebut sumber asli atau tanpa ijin pemilik

Jika kemudian terbukti terjadi pelanggaran terhadap pernyataan di atas, maka saya bersedia menerima sanksi apapun termasuk pencabutan gelar akademik.

Lembar pernyataan ini juga memberikan hak kepada Politeknik Negeri Batam untuk mempergunakan, mendistribusikan ataupun memproduksi ulang seluruh hasil Tugas Praktikum ini.

Batam, 29 Oktober 2019

**Hendra Febrian**

**NIM: 3311811023**

# HALAMAN PERNYATAAN

Dengan ini, saya:

NIM : 3311811039

Nama : Nadya Alisa Sesiq Millenia Kasim

adalah mahasiswa Program Studi Teknik Informatika Politeknik Negeri Batam menyatakan bahwa Tugas Praktikum dengan judul:

**Sistem order Barang menggunakan**

**Metode Unified Modelling Language(UML)**

disusun dengan:

1. tidak melakukan plagiat terhadap naskah karya oranglain
2. tidak melakukan pemalsuan data
3. tidak menggunakan karya orang lain tanpa menyebut sumber asli atau tanpa ijin pemilik

Jika kemudian terbukti terjadi pelanggaran terhadap pernyataan di atas, maka saya bersedia menerima sanksi apapun termasuk pencabutan gelar akademik.

Lembar pernyataan ini juga memberikan hak kepada Politeknik Negeri Batam untuk mempergunakan, mendistribusikan ataupun memproduksi ulang seluruh hasil Tugas Praktikum ini.

Batam, 29 Oktober 2019

**Nadya Alisa Sesiq Millenia Kasim**

**NIM: 3311811039**

# Abstract

Pengelolaan data karya ilmiah yang dibuat oleh mahasiswa tak lagi dilakukan secara manual, hal ini dikarenakan sering terjadinya kekeliruan dalam memasukkan data sehingga mengakibatkan ketidak efisien dalam pengelolaan data, dan terkadang juga membuat sulit untuk mencari data berdasarkan tema karya ilmiah yang dibuat oleh mahasiswa sebelumnya. Di era digital seperti sekarang, sudah memiliki sistem pengelolaan data yang mudah, cepat, dan efisien sehingga dapat mempermudah pekerjaan pengelolaan data karya ilmiah tersebut, serta Mahasiswa pun kini tak lagi repot untuk mencari bahan referensi karya ilmiah/skripsi yang dibuat dari mahasiswa angkatan sebelumnya. Dalam melakukan praktek ini, penulis menggunakan bahasa pemodelan UML(unified Modelling Language)

Key : Elektronik, UML(Unified Modelling Language)

# Daftar Isi

[HALAMAN PENGESAHAN ii](#_Toc24471481)

[HALAMAN PERNYATAAN iii](#_Toc24471482)

[HALAMAN PERNYATAAN iv](#_Toc24471483)

[HALAMAN PERNYATAAN v](#_Toc24471484)

[Abstract vi](#_Toc24471485)

[Daftar Isi vii](#_Toc24471486)

[BAB II 1](#_Toc24471495)

[LANDASAN TEORI 1](#_Toc24471496)

[2.1 Pengertian UML 1](#_Toc24471497)

[2.2 Jenis-jenis diagram UML 1](#_Toc24471498)

[2.2.1 Use case 1](#_Toc24471499)

[2.2.2 Activity Diagram 2](#_Toc24471500)

[2.2.3 Sequence diagram 2](#_Toc24471501)

[BAB III 3](#_Toc24471502)

[ANALISIS DAN PERANCANGAN 3](#_Toc24471503)

[3.1 Gambaran Umum Perangkat Lunak 3](#_Toc24471504)

[3.2 Kebutuhan Fungsional 3](#_Toc24471505)

[3.3 Kebutuhan NonFungsional 3](#_Toc24471506)

[3.3.1 Internal Requirements 3](#_Toc24471507)

[3.3.2 External Requirements 4](#_Toc24471508)

[3.4 Use case 4](#_Toc24471509)

[3.5 Skenario use case pencarian data ilmiah oleh mahasiswa 5](#_Toc24471510)

[3.5.1 Skenario usecase penampilan grafik 6](#_Toc24471511)

[3.5.2 skenario usecase penampilan data nongrafik 6](#_Toc24471512)  
3.5.3 Skenario usecase memanipulasi atau karya ilmiah 6  
3.5.4 **Skenario usecase menampilkan data dosen pembimbing. 7**3.5.5 Skernario usecase menampilkan history 8  
3.5.6 Skenario uscase mencetak laporan 8

[3.6 Activity diagram 9](#_Toc24471513)

[3.7 Sequence diagram 17](#_Toc24471514)

[3.7.1 UML Sequence Mencari Data Karya Ilmiah/Skripsi 17](#_Toc24471515)

[3.7.2 Sequence menampilkan grafik data karya ilmiah/skripsi mahasiswa 18](#_Toc24471516)

3.7.3 Sequence memanipulasi data karya Ilmiah 18

3.7.4 Sequence memanipulasi data karya Ilmiah 19

3.7.5 Sequence menampilkan histori 20

3.7.6 Sequence mencetak laporan 21

3.7.7 Sequence menampilkan non grafik 22

[3.8 Manajemen Kualitas Perangkat Lunak 23](#_Toc24471514)

3.8.1 Safety & Security 23  
3.8.2 Reliability 23  
3.8.3 Resilience & Robustness 23  
3.8.4 Understandability 23  
3.8.5 Adaptability 23

[3.9 Manajemen Resiko 23](#_Toc24471514)  
[3.10 Perencanaan Proyek 25](#_Toc24471514)

[3.10.1 Perencanaan Jadwal (schedule) 25](#_Toc24471514)  
3.10.2 Biaya Proyek 26

# BAB II

# LANDASAN TEORI

## Pengertian UML

UML adalah bahasa untuk menspesifikasi, memvisualisasi, membangun dan mendokumentasikan artifacts (bagian dari informasi yang digunakan untuk dihasilkan oleh proses pembuatan perangkat lunak, artifact tersebut dapat berupa model, deskripsi atau perangkat lunak) dari sistem perangkat lunak,seperti pada pemodelan bisnis dan sistem non perangkat lunak lainnya. UML memungkinkan developer melakukan permodelan secara visual, yaitu penekanan pada penggambaran, bukan didominasi oleh narasi. Permodelan visual membantu untuk menangkap struktur dan kelaukan dari objek, mempermudah penggambaran interaksi antara elemen dalam system, dan mempertahankan konsistensi antara disain dan implementasi dalam pemrograman.

Namun karena UML hanya merupakan bahasa untuk pemodelan maka UML bukanlah rujukan bagaimana melakukan analisis dan disain berorientasi objek secara baik, sudah terdapat beberapa metodologi yang bisa diikuti, seperti Metode Booch, Metode Coad dan Yourdan, Metode Jacobson, Metode Rumbaugh, Metode Wirfs-Brock, atau mengikuti metode pengembangan system Unified Process. (Hermawan, 2005).

## Jenis-jenis diagram UML

### Use case

**Use Case** adalah pemodelan untuk menggambarkan kelakuan (behavior) sistem yang akan dibuat. Diagram use case mendeskripsikan sebuah interaksi antara satu atau lebih aktor dengan sistem yang akan dibuat. Dengan pengertian yang cepat, diagram use case digunakan untuk mengetahui fungsi apa saja yang ada di dalam sebuah sistem dan siapa saja yang berhak menggunakan fungsi-fungsi tersebut.Hal yang perlu diingat mengenai diagram use case adalah diagram use case bukan menggambarkan tampilan antarmuka (user interface), arsitektur dari sistem, kebutuhan nonfungsional, dan tujuan performansi.

Activity Diagram

*Activity Diagram*merupakan alur kerja(*workflow*) atau kegiatan (aktivitas) dari sebuah sistem atau menu yang ada pada perangkat lunak. *Activity Diagram*juga digunakan untuk mendefinisikan urutan atau pengelompokan tampilan dari sistem / *user interface*dimana setiap aktivitas dianggap memiliki sebuah rancangan antar muka tampilan serta rancang menu yang ditampilkan pada perangkat lunak.

### Sequence diagram

Sequence Diagram adalah suatu diagram yang menggambarkan interaksi objek dan mengindikasikan (memberi petunjuk atau tanda) komunikasi diantara objek-objek tersebut . Sequence diagram digunakan untuk menggambarkan perilaku pada sebuah skenario dan mendeskripsikan bagaimana entitas dam sistem berinteraksi , termasuk pesan yang digunakan saat interaksi . Semua pesan dideskripsikan dalam urutan pada eksekusi . Sequence diagram berhubungan erat dengan Use Case Diagram , dimana 1 Use Case akan menjadi 1 Sequence Diagram.

# BAB III

# ANALISIS DAN PERANCANGAN

## Gambaran Umum Perangkat Lunak

## Kebutuhan Fungsional

Kebutuhan fungsional adalah kebutuhan-kebutuhan yang memiliki keterkaitan langsung dengan sistem. Kebutuhan fungsional dari aplikasi ini meliputi :

F001 Dosen Prodi berhak menghapus dan mengelola data karya ilmiah/skripsi

mahasiswa.

F002 Mahasiswa dapat memperoleh informasi lengkap mengenai referensi judul

karya ilmiah / skripsi dari mahasiswa angkatan sebelumnya.

F003 Dosen Prodi dapat melihat jumlah tema, judul, dan jumlah karya ilmiah yang

telah dibuat oleh mahasiswa dalam bentuk grafik batang dan terdapat

laporan dalam bentuk non grafik atau tulisan.

F004 Mahasiswa dapat mencari data Dosen Pembimbing.

F005 Dosen Prodi dapat mencetak langsung karya ilmiah / skripsi berupa tulisan

yang ditentukan berdasarkan tahun akademik.

F006 Dosen Prodi dapat mencetak laporan karya ilmiah/skripsi.

## Kebutuhan NonFungsional

Kebutuhan non-fungional adalah kebutuhan yang tidak secara langsung terkait dengan fitur tertentu di dalam sistem, yaitu :

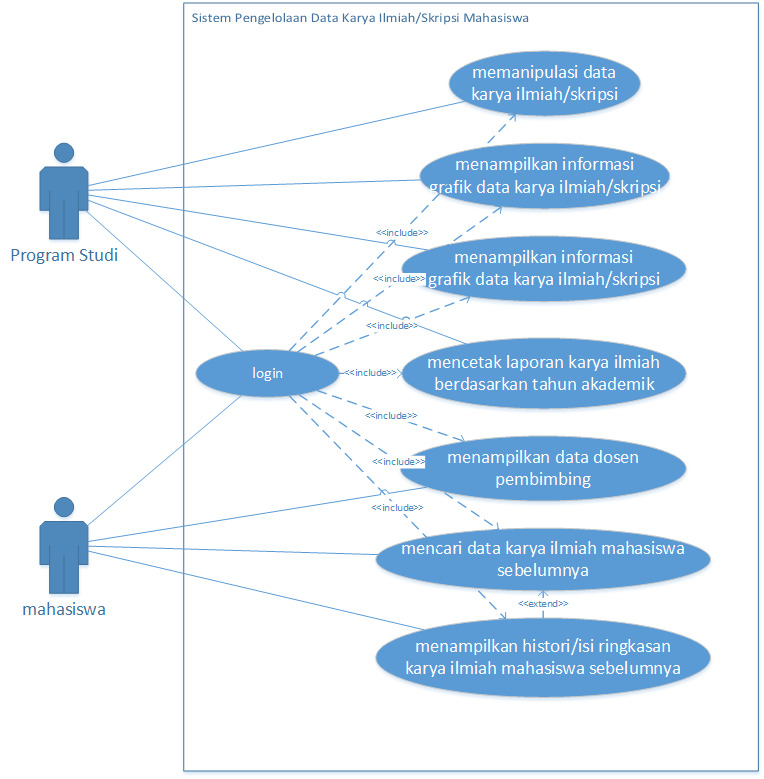
### Internal Requirements

* NF001 aplikasi pengelolaan data karya ilmiah/skripsi mahasiswa ini dapat diakses melalui web atau mobile.
* NF002 untuk pengguna mobile minimal resolusi layar 320x240 pixel.

### External Requirements

* NF001 sistem aplikasi dapat mengamankan data-data privasi pengguna.

## Use case



**Gambar 3.4** use case diagram sistem aplikasi pengelolaan data karya ilmiah

Pada gambar diatas merupakan use case diagram dari sistem pengelolaan data karya ilmiah/skripsi mahasiswa. Use case diatas memiliki dua faktor yaitu Dosen Prodi dan Mahasiswa. Dosen prodi dapat menginput data karya ilmiah/skripsi mahasiswa,dan dapat menampilkan informasi judul karya ilmiah/skripsi mahasiswa angkatan sebelumnya, kemudian Mahasiswa dapat melakukan pencarian data karya ilmiah/skripsi mahasiswa sebelumnya, lalu memilih judul karya ilmiah/skripsi yang dibutuhkannya sebagai referensi, dan juga dapat melihat data dosen pembimbing yang ditampilkan oleh dosen prodi di dalam sistem aplikasi. Kemudian Mahasiswa dapat juga mencetak ringkasan karya ilmiah yang ia pilih untuk dijadikan referensi.

## Skenario use case

### Skenario usecase pencarian data karya ilmiah oleh Mahasiswa.

|  |  |
| --- | --- |
| Aksi aktor | Reaksi sistem |
| 1. Mahasiswa melakukan login |  |
|  | 2. system memeriksa validasi data login |
|  | 3. Jika data login tidak valid, maka Mahasiswa gagal login, dan tidak bisa masuk ke halaman utama. |
|  | 4. Jika data login valid, maka Mahasiswa berhasil login dan akan masuk ke halaman utama. |
| 5. Mahasiswa memilih menu cari data karya ilmiah |  |
|  | 6. Sistem menampilkan form filter |
| 7. Mahasiswa input data filter |  |
|  | 8. Menampilkan data karya ilmiah hasil filter |

### skenario usecase menampilkan grafik

|  |  |
| --- | --- |
| Aksi aktor | Reaksi sistem |
| 1. Dosen Prodi melakukan login |  |
|  | 1. system memeriksa validasi data login |
|  | 3. Jika data login tidak valid, maka program studi gagal login, dan tidak bisa masuk ke halaman utama. |
|  | 4. Jika data login valid, maka program studi berhasil login dan akan masuk ke halaman utama |
| 5. Dosen Prodi memilih menu informasi grafik karya ilmiah. |  |
|  | 6. Sistem menampilkan grafik |

**3.5.3 Skenario usecase menampilkan data non grafik**

|  |  |
| --- | --- |
| Aksi aktor | Reaksi sistem |
| 1. Dosen Prodi melakukan login |  |
|  | 2. system memeriksa validasi data login |
|  | 3. Jika data login tidak valid, maka program studi gagal login, dan tidak bisa masuk ke halaman utama. |
|  | 4. Jika data login valid, maka program studi berhasil login dan akan masuk ke halaman utama |
| 5. Dosen Prodi memilih menu informasi karya ilmiah. | . |
|  | 6. Sistem menampilkan informasi karya ilmiah. |

**3.5.4 Skenario usecase memanipulasi atau karya ilmiah**

|  |  |
| --- | --- |
| Aksi aktor | Reaksi sistem |
| 1. Dosen Prodi melakukan login |  |
|  | 2. system memeriksa validasi data login |
|  | 3. Jika data login tidak valid, maka program studi gagal login, dan tidak bisa masuk ke halaman utama. |
|  | 4. Jika data login valid, maka program studi berhasil login dan akan masuk ke halaman utama. |
| 5. Dosen Prodi memilih menu memanipulasi data |  |
|  | 6. Sistem menampilkan form. |
| 7. Dosen Prodi memasukkan data |  |
|  | 8. Sistem menyimpan data yang dimasukkan. |
|  | 9. Sistem menampilkan data yang sudah dimanipulasi |

**3.5.5 Skenario usecase menampilkan data dosen pembimbing.**

|  |  |
| --- | --- |
| Aksi aktor | Reaksi sistem |
| 1. Mahasiswa melakukan login |  |
|  | 2. system memeriksa validasi data login |
|  | 3. Jika data login tidak valid, maka Mahasiswa gagal login, dan tidak bisa masuk ke halaman utama. |
|  | 4. Jika data login valid, maka Mahasiswa berhasil login dan akan masuk ke halaman utama. |
| 5. Mahasiswa memilih menu dosen pembimbing. |  |
|  | 6. Sistem menampilkan data dosen pembimbing. |

**3.5.6. Skenario usecase menampilkan histori**

|  |  |
| --- | --- |
| Aksi aktor | Reaksi sistem |
| 1. Mahasiswa melakukan login |  |
|  | 2. system memeriksa validasi data login |
|  | 3. Jika data login tidak valid, maka Mahasiswa gagal login, dan tidak bisa masuk ke halaman utama. |
|  | 4. Jika data login valid, maka Mahasiswa berhasil login dan akan masuk ke halaman utama. |
| 5. Mahasiswa memilih menu cari data karya ilmiah |  |
|  | 6. Sistem menampilkan form filter |
| 7. Mahasiswa input data filter |  |
|  | 8. Menampilkan data karya ilmiah hasil filter |
| 9. Memilih menampilkan histori/ringkasan karya ilmiah. |  |
|  | 9. Sistem menampilkan histori/ringkasan karya ilmiah. |

**3.5.7 Skenario usecase mencetak laporan**

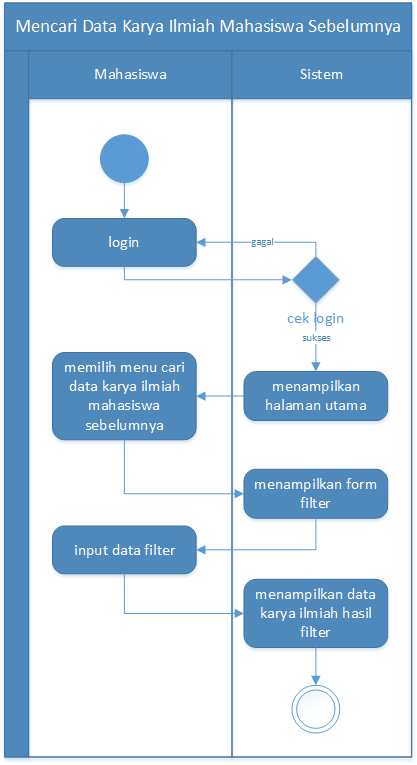
|  |  |
| --- | --- |
| Aksi aktor | Reaksi sistem |
| 1. Program studi melakukan login |  |
|  | 2. system memeriksa validasi data login |
|  | 3. Jika data login tidak valid, maka program studi gagal login, dan tidak bisa masuk ke halaman utama. |
|  | 4. Jika data login valid, maka program studi berhasil login dan akan masuk ke halaman utama. |
| 5. Program studi memilih menu cari cetak laporan karya ilmiah |  |
|  | 6. Sistem menampilkan form filter tahun akademik. |
| 7. Program studi input tahun akademik. |  |
|  | 8. Menampilkan data karya ilmiah berdasarkan tahun akademik yang diinput. |
| 9. Program studi memilih menu "cetak" |  |
|  | 9. Sistem memproses cetak karya ilmiah. |

## Activity diagram

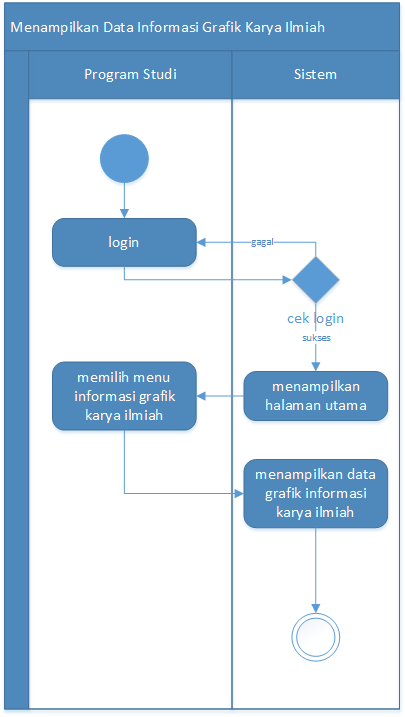
Activity diagram merupakan representasi grafis dari seluruh tahapan alur kerja. Berikut

activity diagram :

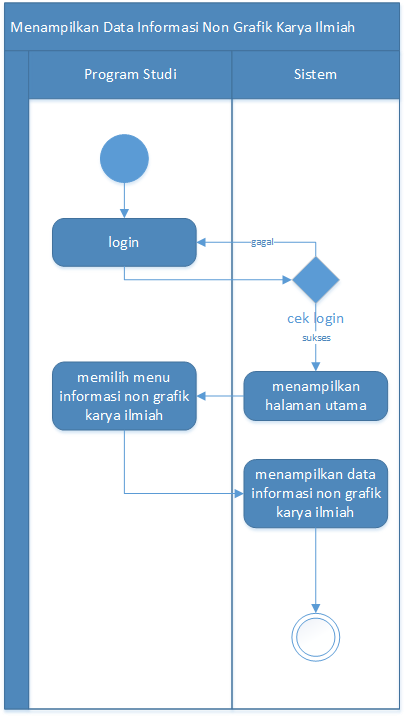
* Mencari data karya ilmiah sebelumnya.



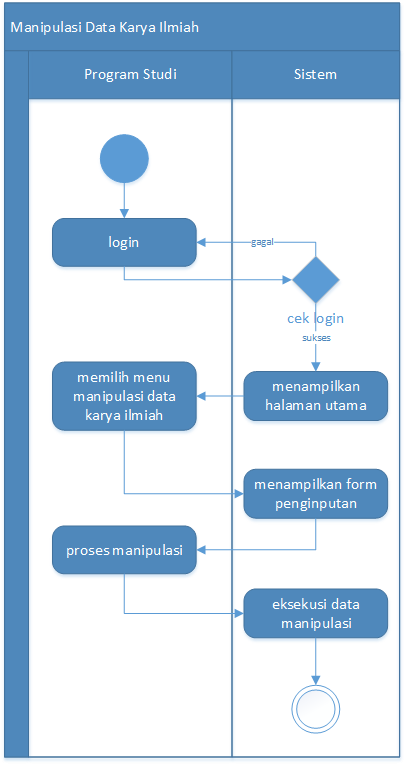
* Menampilkan grafik



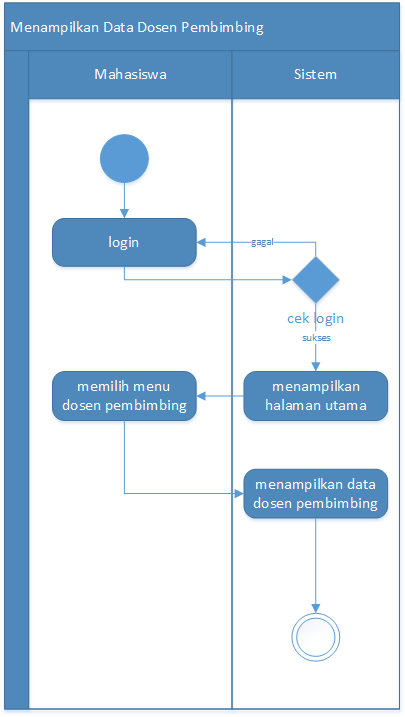
* Menampilkan nongrafik



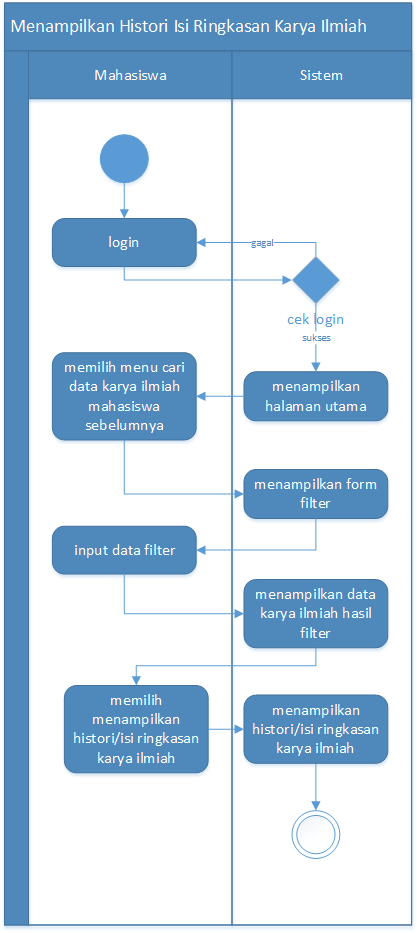
* Memanipulasi data karya ilmiah



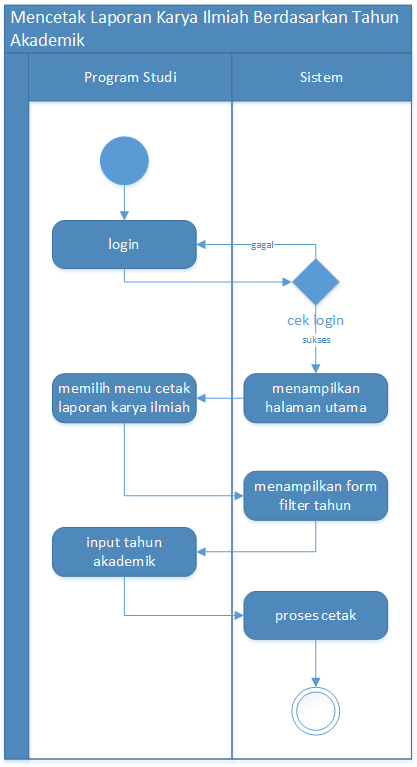
* Menampilkan data dosen pembimbing



* Menampilkan histori



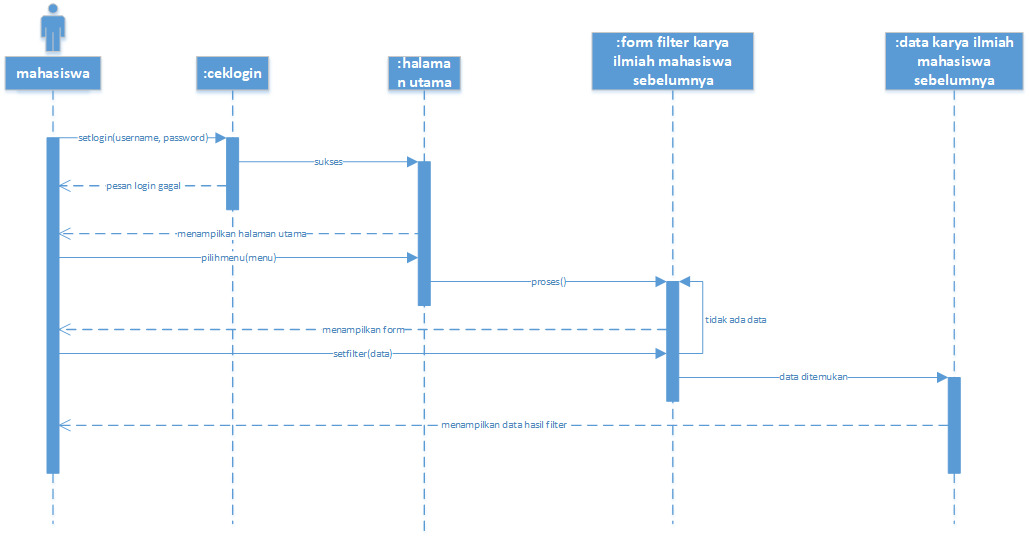
* Mencetak laporan



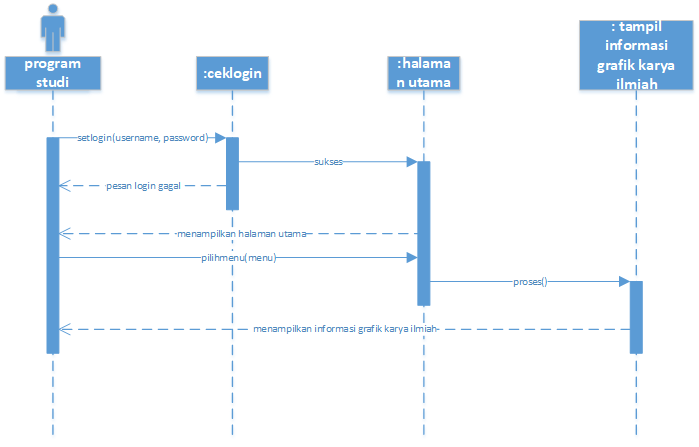
## Sequence diagram

Sequence diagram merupakan suatu diagram yang memperlihatkan atau menampilkan interaksi-interaksi antar objek didalam suatu sisem yang disusun pada sebuah urutan atau rangkaian waktu. Berikut sequence diagram:

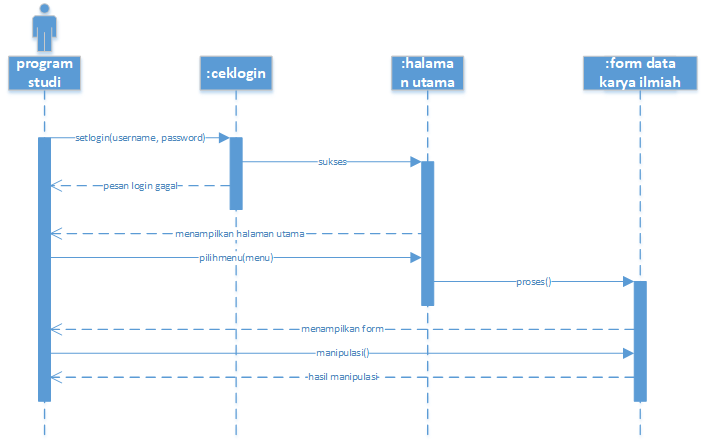
### UML Sequence Mencari Data Karya Ilmiah/Skripsi



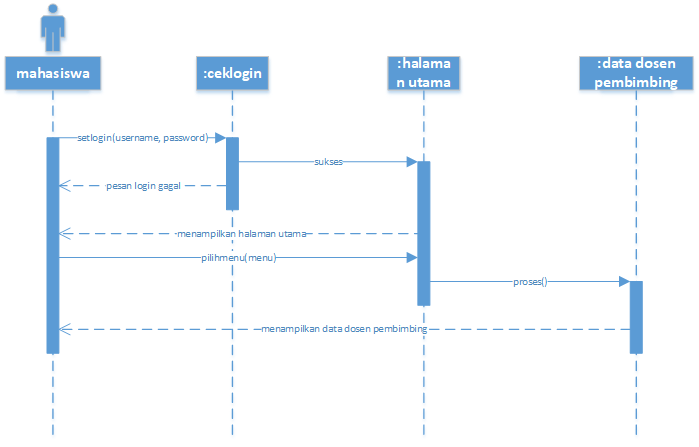
### Sequence menampilkan grafik data karya ilmiah/skripsi mahasiswa



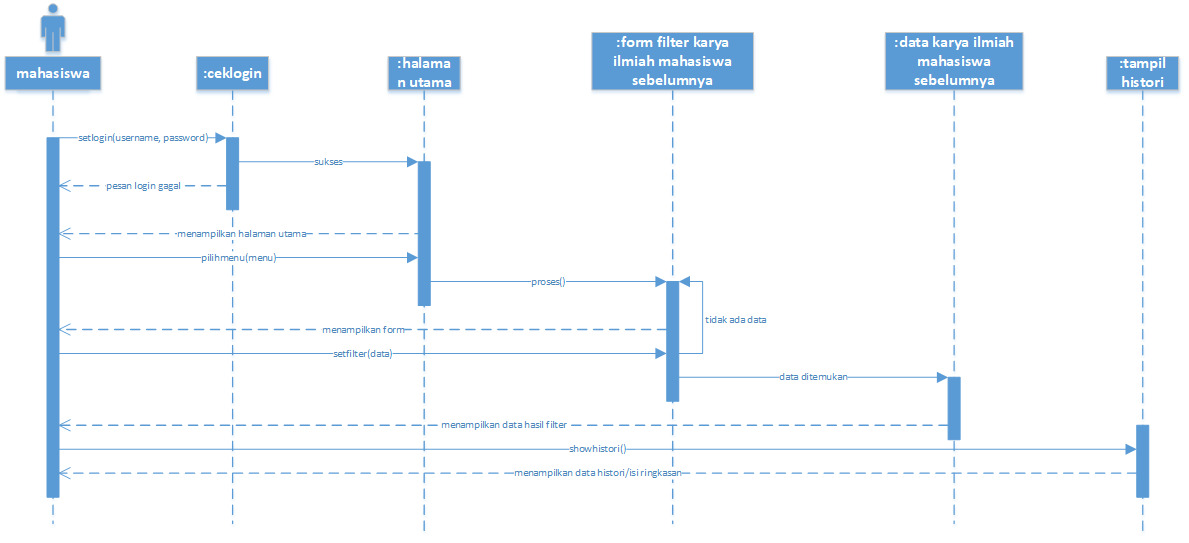
**3.7.3 Sequence memanipulasi data karya Ilmiah**



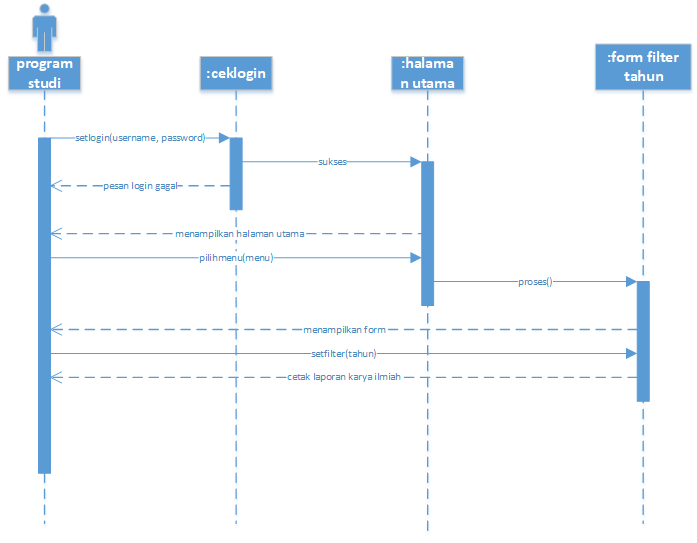
**3. 7.4. Sequence mencari data dosen pembimbing**



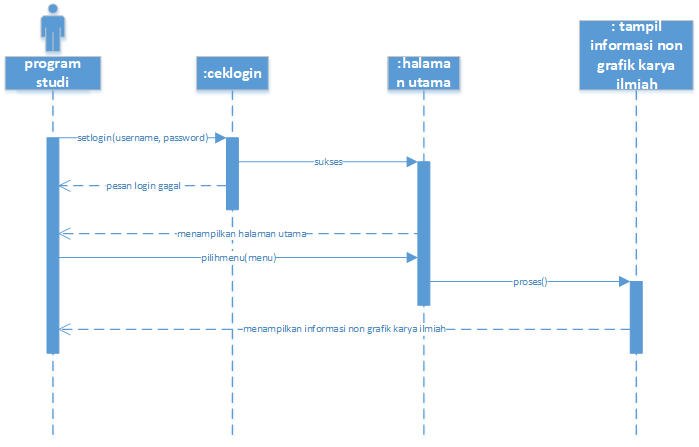
**3.7.5 Sequence menampilkan histori**



**3.7.6. Sequence mencetak laporan**



**3.7.7 Sequence menampilkan non grafik**



**3.8 Manajemen Kualitas Perangkat Lunak**

adalah aktifitas pelindung yang diaplikasikan pada seluruh proses software.

* **3.8.1 Safety & Security**

Aplikasi ini dilengkapi dengan sistem login.

* **3.8.2 Reliability**

Aplikasi ini dapat mempermudah Dosen Prodi memanipulasi data

karya ilmiah mahasiswa.

* **3.8.3 Resilience & Robustness**

Aplikasi ini memiliki validasi untuk mencegah error yang disebabkan

oleh user.

* **3.8.4 Understandability**

Aplikasi ini memiliki panduan (petunjuk teknis) dalam

menggunakannya serta dilengkapi dengan bahasa yang mudah

dimengerti oleh user.

* **3.8.5 Adaptability**

Aplikasi ini tidak membebani ruang memori aplikasi lainnya pada saat

di install.

**3.9 Manajemen Resiko**

Pengukuran atau penilaian resiko serta pengembangan strategi

pengelolaannya. Cara penanganan pada aplikasi ini yaitu, menggunakan

metodologi Spiral Boehm. Model ini menggunakan fitur yang ada pada model

waterfall dan prototype. Setiap tahapan model ini selalu dilakukan risk

analisys dan verivikasi atau testing.

* Model ini memiliki empat aktivitas yaitu :

1. **Determine objectives** : penentuan tujuan, alternative dan batasan dalam proses.
2. **Risk Analysis** : analisa alternatif terhadap resiko yang mungkin terjadi.

* Project Risk : Hal-hal yang mempengaruhi tahap proyek, contohnya kekurangan sumber daya.
* Technical Risk : Hal-hal yang mempengaruhi tahap actual, contohnya personil tidak terlatih ditahap tersebut.
* Bussiness Risk : Hal-hal yang mempengaruhi keinginan perusahaan untuk membuat software, contohnya software ternyata tidak dibutuhkan lagi.

1. **Engineering/develop** : pengembangan produk.
2. **Plant next phase** : penentuan rencana-rencana untuk tahap selanjutnya.

* Prioritas resiko dapat dikategorikan sebagai berikut :

1. **Catastrophic (luar biasa), contoh :** penurunan kualitas yang luar biasa, biaya yang tidak terkontrol.
2. **Critical (kritis), contoh :** tidak tepat waktu, biaya diluar perkiraan.
3. **Marginal (ringan), contoh :** penjadwalan yang terlambat
4. **Negligible (tidak berarti), contoh :**  penggunaan waktu proyek yang tidak optimal.

|  |  |  |
| --- | --- | --- |
| Resiko | Kejadian | Teknik Mengurangi Resiko |
| kegagalan personil | Kesalahan dalam bidang coding | memperkejakan staf yang handal |
| Tidak menguasai pada bidangnya | membangun tim |
| terjadi ketidakkompakan tim | mengadakan pelatihan dan peningkatan karir |
| jadwal yang rancu | membuat jadwal lebih awal bagi personil utama |
| Estimasi Biaya | biaya yang berlebihan | membuat beberapa estimasi |
| kekurangan biaya | desain untuk biaya |
| kesalahan perhitungan | standarisasi metode |
| mengembangkan fungsi software yang salah | kesalah pemrograman | eveluasi ditingkatkan |
| kesalahan desain | survey pengguna |
| analisa kebutuhan yang tidak sesuai | buat prototype |

**3.10. Perencanaan Proyek**

* **3.10.1 Perencanaan Jadwal (schedule)**

|  |  |  |  |
| --- | --- | --- | --- |
| **DAFTAR AKTIVITAS YANG HARUS DILAKUKAN** | | | **Durasi(Hari)** |
| **1** | **Analisis Sistem** | Melakukan wawancara kebutuhan ke pihak client | 2 |
| Membuat diagram | 3 |
| Membuat desain database | 7 |
| Dokumentasi kebutuhan dan desain sistem | 2 |
| **2** | **Desain Aplikasi** | Membuat Desain Aplikasi | 10 |
| Dokumentasi Desain Aplikasi | 2 |
| **3** | **Implementasi** | Membuat Program Aplikasi | 20 |
| Dokumentasi Program Aplikasi | 2 |
| **4** | **Testing** | Membuat List Testing Program & melakukan testing | 5 |
| Dokumentasi Testing Program & Catatan Perbaikan | 2 |
| **5** | **Instalasi** | Instalasi Program Aplikasi | 2 |
| Membuat Dokumentasi User Guide | 2 |
| **6** | **Maintenance** | Pemeliharaan Aplikasi | 2 |

* **3.10.2 Biaya Proyek**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Keterangan** | **Orang** | **hari** | **Harga/hari** | **Jumlah** | |
| **1. Implementasi** | | | | | |
| 1.1 Biaya Personil |  |  |  |  | |
| - Project Manager | 1 Orang | 30 | 400.000 | Rp 12.000.000,00 | |
| - System Analyst | 1 Orang | 12 | 350.000 | Rp 4.200.000,00 | |
| - Programmer | 1 Orang | 22 | 350.000 | Rp 7.700.000,00 | |
| - Desainer | 1 orang | 12 | 350.000 | Rp 4.200.000,00 | |
| **Sub Jumlah** |  |  |  | **Rp 28.100.000,00** | |
| **2. Biaya Penunjang** | | | | | |
| 2.1 Transportasi | 4 Orang | 30 | 15.000 | Rp 1.800.000,00 | |
| **Sub Jumlah** |  |  |  | **Rp 1.800.000,00** | |
| **3. Hardware & Sofware Tambahan** | | | | | |
| 3.1 1 Set PC Server |  |  |  | | Rp 8.500.000,00 |
| 3.2 Web Hosting Premium |  |  |  | | Rp 200.000,00 |
| **Sub Jumlah** |  |  |  | | **Rp 8.700.000,00** |
| **Total** | | | | | Rp 38.600.000,00 |
| **Biaya lain-lain** | | | | | Rp 400.000,00 |
| **Total Keseluruhan** | | | | | **Rp 39.000.000,00** |