

## BAB 6

### KESIMPULAN DAN SARAN

#### 6.1 Kesimpulan

Setelah sistem CariEdu ini selesai diimplementasikan maka dapat ditarik beberapa kesimpulan, yaitu:

1. Perangkat Lunak CariEdu telah berhasil dibangun sebagai sistem layanan berbasis lokasi yang menampilkan lokasi dan informasi unit pendidikan dengan menggunakan teknologi *Google Maps API* pada perangkat *mobile* Android.
2. Dari hasil pengujian dapat diketahui bahwa fungsi-fungsi yang disediakan oleh perangkat lunak CariEdu berjalan dengan benar dan sesuai dengan yang diharapkan.

#### 6.2 Saran

Penulis ingin memberikan beberapa saran untuk pengembangan lebih lanjut perangkat lunak CariEdu ini:

1. Mengembangkan aplikasi agar dapat berjalan di semua platform perangkat *mobile*.
2. Menambahkan informasi terpusat yang disajikan pemerintah mengenai pendidikan sebagai *given information* untuk melengkapi informasi kolektif yang ada.

## DAFTAR PUSTAKA

- Cisco System, Inc. 2008. *Wi-Fi Location-Based Services 4.1 Design Guide*.
- Deitel, Harvey M., Deitel Paul J., Nieto, Tem R., Steinbuhler, Kate. 2002. *Wireless Internet & Mobile Business How to Program*. Prentice Hall.
- Developer Android. 2010. *Developing In Eclipse, With ADT*. <http://developer.android.com/guide/developing/eclipse-adt.html>.
- McAndrew, James. *Working with Google Maps, A brief introduction to the Google Maps API*. Diakses pada Juni 2011 di <http://www.scribd.com/doc/9277394/A-Brief-Introduction-to-the-Google-Maps-API>.
- Object Technology International, Inc. 2003. *Eclipse Platform Technical Overview*.
- PERATURAN PEMERINTAH REPUBLIK INDONESIA NOMOR 17 TAHUN 2010 TENTANG PENGELOLAAN DAN PENYELENGGARAAN PENDIDIKAN.
- Riyanto. 2010. *Sistem Informasi Geografis Berbasis Mobile*. Gava Media.
- Simmonds, Chris. 2010. *What else can you do with Android?*. 2net Limited.
- Stefan Steiniger, Moritz Neun, Alistair Edwardes. 2006. *Foundations of Location Based Services*. University of Zurich.
- UNDANG-UNDANG REPUBLIK INDONESIA NOMOR 20 TAHUN 2003 TENTANG SISTEM PENDIDIKAN NASIONAL.
- Wibowo, Argo. 2011. *Pembangunan Sistem Layanan Berbasis Lokasi Pencarian UMKM Terdekat Berbasis Android*. Universitas Atma Jaya Yogyakarta.

- Wicaksono, Antonius. 2008. *Pembangunan Sistem GPS Mobile Phone Tracker Menggunakan Google Maps API*. Universitas Atma Jaya Yogyakarta.
- Yudistira, I.B. Putu Sandhi. 2007. *PEMBANGUNAN APLIKASI LOCATION BASED SERVICE KOTA YOGYAKARTA PADA PERANGKAT MOBILE BERBASIS J2ME*. Universitas Atma Jaya Yogyakarta.



# SKPL

## SPEKIFIKASI KEBUTUHAN PERANGKAT LUNAK

CariEdu

Untuk:

Universitas Atma Jaya Yogyakarta

Dipersiapkan oleh:

Eduardo Aji Pradana / 070705215

Program Studi Teknik Informatika - Fakultas Teknologi Industri

Universitas Atma Jaya Yogyakarta

	Program Studi Teknik Informatika Fakultas Teknologi Industri	Nomor Dokumen		Halaman
		<b>SKPL-CariEdu</b>		35
		Revisi		

## DAFTAR PERUBAHAN

Revisi	Deskripsi
<b>A</b>	
<b>B</b>	
<b>C</b>	
<b>D</b>	
<b>E</b>	
<b>F</b>	

INDEX TGL	-	A	B	C	D	E	F	G
Ditulis oleh								
Diperik sa oleh								
Disetuj ui oleh								

### Daftar Halaman Perubahan

Halaman	Revisi	Halaman	Revisi

## Daftar Isi

1	Pendahuluan.....	6
1.1	Tujuan .....	6
1.2	Lingkup Masalah .....	6
1.3	Definisi, Akronim dan Singkatan .....	7
1.4	Referensi .....	8
1.5	Deskripsi umum (Overview) .....	8
2	Deskripsi Kebutuhan.....	9
2.1	Perspektif produk .....	9
2.2	Fungsi Produk .....	11
2.3	Karakteristik Pengguna .....	13
2.4	Batasan-batasan .....	13
2.5	Asumsi dan Ketergantungan .....	14
3	Kebutuhan khusus.....	15
3.1	Kebutuhan antarmuka eksternal .....	15
3.2	Kebutuhan fungsionalitas Perangkat Lunak ...	17
4	Spesifikasi Rinci Kebutuhan.....	18
5	Entity Relationship Diagram (ERD).....	35

## Daftar Gambar

Gambar 1	Arsitektur Perangkat Lunak CariEdu.....	10
Gambar 2	Use Case Diagram .....	17
Gambar 3	Entity Relationship Diagram .....	35



## 1 Pendahuluan

### 1.1 Tujuan

Dokumen Spesifikasi Kebutuhan Perangkat Lunak (SKPL) ini merupakan dokumen spesifikasi kebutuhan perangkat lunak CariEdu yang mendefinisikan kebutuhan perangkat lunak yang meliputi antarmuka eksternal, database, dan feature yang dimiliki, serta mendefinisikan fungsi perangkat lunak. SKPL CariEdu ini juga mendefinisikan batasan perancangan perangkat lunak. Dokumen ini juga dapat digunakan oleh pengembang perangkat lunak sebagai acuan teknis pengembangan perangkat lunak pada tahap selanjutnya.

### 1.2 Lingkup Masalah

Perangkat Lunak CariEdu mempunyai ruang lingkup permasalahan yang hanya akan dibatasi pada:

1. Aplikasi ini dibangun khusus untuk berjalan pada perangkat *mobile* bersistem operasi Android.
2. Peta yang digunakan adalah *Google Maps*.
3. Fitur-fitur yang disediakan pada Aplikasi ini yaitu:
  - a. Menampilkan informasi dan posisi mengenai sampel unit-unit pendidikan berada di Kota Yogyakarta.
  - b. Katalog dan pencarian unit pendidikan.
  - c. Rute dari satu lokasi ke unit pendidikan terpilih.

### 1.3 Definisi, Akronim dan Singkatan

Daftar definisi akronim dan singkatan:

Keyword/Phrase	Definisi
SKPL	SKPL (Spesifikasi Kebutuhan perangkat Lunak) Merupakan spesifikasi kebutuhan dari perangkat lunak yang akan dikembangkan.
SKPL-CariEdu-XXX	Kode yang merepresentasikan kebutuhan pada CariEdu dimana XXX merupakan nomor fungsi produk.
CariEdu	Aplikasi <i>mobile</i> untuk menampilkan lokasi dan informasi mengenai unit-unit pendidikan.
Database	Database atau basis data adalah kumpulan rekaman/data yang terintegrasi dalam wadah umum yang menyediakan data untuk satu atau lebih kebutuhan penggunaan.

Pengguna	Pengguna adalah administrator, curator dan anggota.
Administrator	Administrator adalah aktor yang berhak mengelola unit pendidikan dan pengguna dalam CariEdu.
Kurator	Kurator adalah aktor yang berhak menyetujui atau menolak permintaan.
Anggota	Anggota adalah aktor yang telah terdaftar dalam CariEdu.
Edu	Edu adalah istilah yang digunakan dalam aplikasi CariEdu yang sama artinya dengan unit pendidikan.

#### 1.4 Referensi

Referensi yang digunakan pada perangkat lunak tersebut adalah:

1. Eduardo Aji Pradana, *Spesifikasi Kebutuhan Perangkat Lunak Siwabima (Sistem Persewaan Mobil Mahasiswa)*.

#### 1.5 Deskripsi umum (Overview)

Secara umum dokumen SKPL ini terbagi atas 5 bagian utama. Bagian utama berisi penjelasan mengenai dokumen SKPL tersebut yang mencakup tujuan pembuatan SKPL, ruang lingkup masalah dalam pengembangan perangkat lunak tersebut, definisi, akronim, singkatan, referensi dan deskripsi umum tentang dokumen SKPL ini.

Bagian kedua berisi penjelasan umum tentang perangkat lunak CariEdu yang akan dikembangkan, mencakup perspektif produk yang akan dikembangkan, fungsi produk perangkat

lunak, karakteristik pengguna, batasan dalam penggunaan perangkat lunak dan asumsi yang dipakai dalam pengembangan perangkat lunak CariEdu tersebut.

Bagian ketiga berisi penjelasan secara lebih rinci tentang kebutuhan perangkat lunak CariEdu yang akan dikembangkan, menjelaskan kebutuhan antarmuka eksternal dan kebutuhan fungsionalitas perangkat lunak.

Bagian keempat berisi tentang spesifikasi rinci kebutuhan, yaitu bagian yang menjelaskan tentang fungsional produk berdasarkan use case yang ada. Spesifikasi ini meliputi kondisi awal, kondisi akhir, alur utama, alur sampingan/cadangan, dan alur kesalahan dari masing-masing fungsionalitas.

Bagian kelima berisi tentang entity relationship diagram, yaitu diagram dan notasi yang digunakan untuk merepresentasikan struktur data statis pada perangkat lunak.

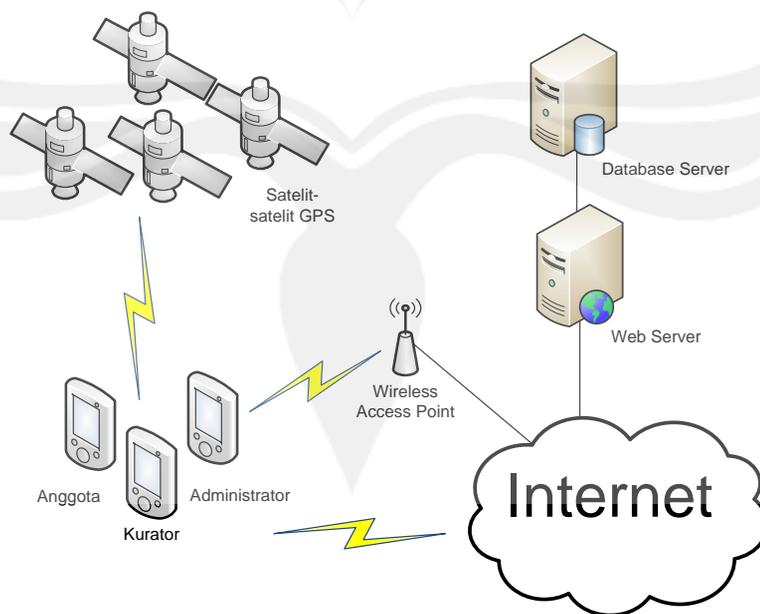
## **2 Deskripsi Kebutuhan**

### **2.1 Perspektif produk**

CariEdu merupakan suatu perangkat lunak untuk perangkat mobile yang dikembangkan untuk memberikan informasi mengenai unit-unit pendidikan serta menampilkan lokasinya dalam peta digital. Perangkat Lunak ini juga menangani pendaftaran anggota, menambahkan lokasi unit pendidikan, menyunting unit pendidikan, menampilkan rute ke unit pendidikan terpilih, serta menyediakan fungsi pencarian.

CariEdu berjalan pada platform Android, dan dibuat menggunakan bahasa pemrograman java. CariEdu dikembangkan dalam lingkungan pemrograman Eclipse. Pengguna akan berinteraksi dengan sistem melalui antarmuka GUI (Graphical User Interface).

Pengguna dalam CariEdu meliputi Administrator, Kurator dan Pengunjung. Setiap peran pengguna ini memiliki aksesibilitas dan peran yang berbeda-beda. Anggota dapat meletakkan *landmark* unit pendidikan dalam peta dan menambahkan informasi mengenai unit pendidikan itu serta menyuntingnya. Kurator dapat mempunyai kemampuan seorang Anggota ditambah fungsi untuk menyetujui permintaan peletakan unit pendidikan. Administrator mempunyai fungsi tambahan selain yang dimiliki Anggota, yaitu menjadikan peran anggota menjadi kurator dan sebaliknya serta menyetujui permintaan peletakan sebuah *landmark* unit pendidikan.



**Gambar 1** Arsitektur Perangkat lunak CariEdu

## 2.2 Fungsi Produk

Fungsi produk perangkat lunak CariEdu adalah sebagai berikut:

1. Fungsi *Login Administrator* (**SKPL-CariEdu-001**).

Merupakan fungsi yang digunakan oleh administrator untuk dapat masuk dalam sistem CariEdu.

2. Fungsi *Login Kurator* (**SKPL-CariEdu-002**).

Merupakan fungsi yang digunakan oleh kurator untuk dapat masuk dalam sistem CariEdu.

3. Fungsi *Daftar Anggota* (**SKPL-CariEdu-003**).

Merupakan fungsi yang digunakan untuk mendaftar dalam CariEdu bagi pengunjung untuk menjadi seorang anggota.

4. Fungsi *Sunting Edu* (**SKPL-CariEdu-004**).

Merupakan fungsi yang digunakan untuk update data Edu.

5. Fungsi *Tambah Edu* (**SKPL-CariEdu-005**).

Merupakan fungsi untuk menambah/mengajukan Edu baru.

6. Fungsi *Hapus Edu* (**SKPL-CariEdu-006**).

Merupakan fungsi untuk menghapus Edu.

7. Fungsi *Setujui Peletakan Edu* (**SKPL-CariEdu-007**).

Merupakan fungsi yang digunakan untuk menyetujui permintaan peletakan Edu.

8. Fungsi *Hapus Anggota* (**SKPL-CariEdu-007**).

Merupakan fungsi untuk menghapus Anggota.

Program Studi Teknik Informatika	SKPL – CariEdu	11/ 35
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

9. Fungsi *Ganti Peran Anggota*(**SKPL-CariEdu-009**) .

Merupakan fungsi untuk mengganti peran dari anggota biasa ke kurator dan sebaliknya.

10. Fungsi *Ganti Lokasi Tempat Tinggal* (**SKPL-CariEdu-010**) .

Merupakan fungsi untuk mengganti lokasi dari tempat tinggal pengguna.

11. Fungsi *Tampil Peta Edu*(**SKPL-CariEdu-011**) .

Merupakan fungsi yang digunakan untuk menampilkan lokasi unit pendidikan dalam peta digital Google Maps.

12. Fungsi *Tampil Rute Edu*(**SKPL-CariEdu-012**) .

Merupakan fungsi yang digunakan untuk menampilkan rute menuju lokasi unit pendidikan dari lokasi pengguna atau dari lokasi tempat tinggal pengguna.

13. Fungsi *Cari Edu*(**SKPL-CariEdu-013**)

Merupakan fungsi yang digunakan untuk mencari dan menyaring unit-unit pendidikan yang ada.

14. Fungsi *Tampil Informasi Edu*(**SKPL-CariEdu-014**)

Merupakan fungsi yang digunakan untuk menampilkan informasi tentang unit pendidikan yang terpilih.

15. Fungsi *Beri Komentar*(**SKPL-CariEdu-015**)

Merupakan fungsi yang digunakan untuk menambahkan komentar pada unit pendidikan yang terpilih.

16. Fungsi *Beri Nilai Edu*(**SKPL-CariEdu-016**)

Merupakan fungsi yang digunakan untuk menambahkan nilai pada unit pendidikan yang terpilih.

Program Studi Teknik Informatika	SKPL – CariEdu	12/ 35
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

17. Fungsi *Cari Location Provider* (**SKPL-CariEdu-017**)

Merupakan fungsi yang digunakan untuk mencari *location provider*.

### 2.3 Karakteristik Pengguna

Karakteristik dari pengguna perangkat lunak CariEdu adalah sebagai berikut:

1. Anggota familiar dengan perangkat mobile dengan sistem operasi android.
2. Kurator familiar dengan perangkat mobile dengan sistem operasi android.
3. Administrator familiar dengan perangkat mobile dengan sistem operasi android.

### 2.4 Batasan-batasan

Batasan-batasan dalam pengembangan perangkat lunak CariEdu adalah:

1. CariEdu tidak menangani registrasi dan urusan administratif terhadap unit-unit pendidikan yang ditampilkan dalam aplikasi.
2. Keterbatasan perangkat keras

Dapat diketahui kemudian setelah sistem ini berjalan (sesuai dengan kebutuhan).

## 2.5 Asumsi dan Ketergantungan

Asumsi dalam perangkat lunak CariEdu ini yaitu kursensi mata uang yang dipakai adalah rupiah.

Ketergantungan aplikasi CariEdu adalah dijalankan pada perangkat *mobile* bersistem operasi Android dan mempunyai modul GPS serta mendukung protokol http. Ketergantungan lain yaitu harus tersedianya akses internet.



### 3 Kebutuhan khusus

#### 3.1 Kebutuhan antarmuka eksternal

CariEdu dibuat sebagai aplikasi *mobile*. Kebutuhan antar muka eksternal pada perangkat lunak CariEdu meliputi kebutuhan antarmuka pemakai, antarmuka perangkat keras, antarmuka perangkat lunak, antarmuka komunikasi.

##### 3.1.1 Antarmuka pemakai

Pengguna berinteraksi dengan antarmuka yang ditampilkan dalam bentuk *form-form*.

##### 3.1.2 Antarmuka perangkat keras

Antarmuka perangkat keras yang digunakan dalam perangkat lunak CariEdu adalah perangkat mobile seperti handphone mendukung protokol http dan modul GPS.

##### 3.1.3 Antarmuka perangkat lunak

Perangkat lunak yang dibutuhkan untuk mengoperasikan perangkat lunak CariEdu adalah sebagai berikut :

1. Nama : Android OS  
Sumber : Android Software.  
Sebagai sistem operasi untuk perangkat *mobile*.
2. Nama : Mysql  
Sumber : Mysql  
Sebagai *Data Base Management System*.
3. Nama : Google Maps API  
Sumber : Google

Sebagai API dalam pembangunan aplikasi peta.

4. Nama : PHP

Sumber : PHP

Sebagai penghubung aplikasi dengan database.

5. Nama : Apache

Sumber : Apache Software Foundation

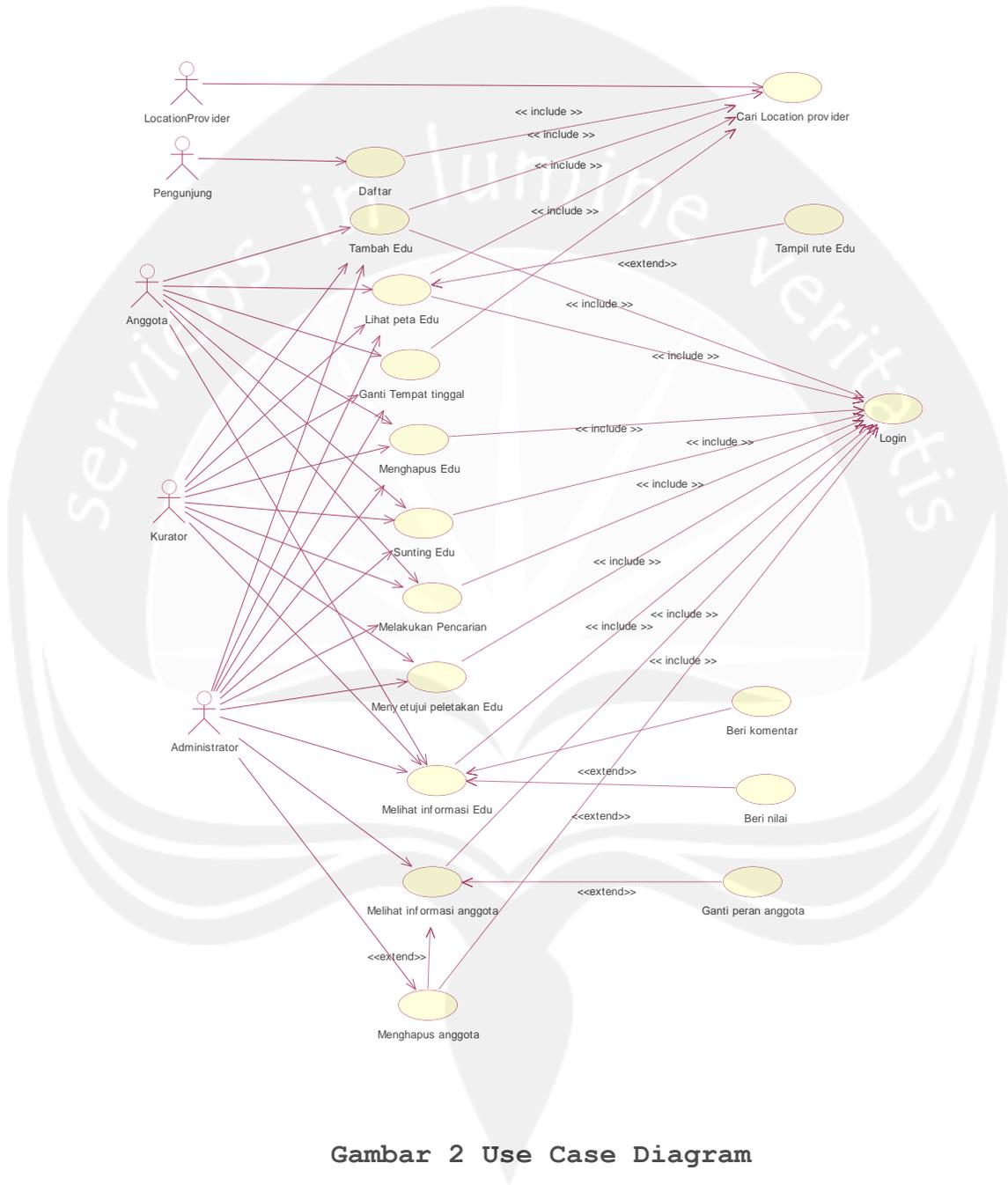
Sebagai Server.

#### **3.1.4 Antarmuka Komunikasi**

Antarmuka komunikasi perangkat lunak CariEdu menggunakan protocol TCP/IP.

### 3.2 Kebutuhan fungsionalitas Perangkat Lunak

#### 3.2.1 Use Case Diagram



Gambar 2 Use Case Diagram

## **4 Spesifikasi Rinci Kebutuhan**

### **4.1.1 Use case Spesification : Login Administrator**

#### **1. Brief Description**

Use Case ini digunakan oleh aktor untuk memperoleh akses ke sistem. Login didasarkan pada sebuah id unik yaitu username dari aktor dan password yang berupa rangkaian karakter.

#### **2. Primary Actor**

1. Administrator

#### **3. Supporting Actor**

none

#### **4. Basic Flow**

1. Use Case ini dimulai ketika aktor memilih untuk melakukan login
2. Sistem menampilkan antarmuka untuk login
3. Aktor memasukkan username dan password
4. Sistem memeriksa username dan password yang dimasukkan aktor  
E-1 Password atau username tidak sesuai
5. Sistem memberikan akses ke aktor
6. Use Case ini selesai

#### **5. Alternative Flow**

none

#### **6. Error Flow**

- E-1 Password atau username tidak sesuai
1. Sistem menampilkan peringatan bahwa username atau password tidak sesuai
  2. Kembali ke Basic Flow langkah ke 3

#### **7. PreConditions**

1. Aktor sudah mendaftar

## 8. PostConditions

1. Aktor memasuki sistem dan dapat menggunakan fungsi-fungsi pada sistem.

### 4.1.2 Use case Spesification : Login Kurator

#### 1. Brief Description

Use Case ini digunakan oleh aktor untuk memperoleh akses ke sistem. Login didasarkan pada sebuah id unik yaitu username dari aktor dan password yang berupa rangkaian karakter.

#### 2. Primary Actor

1. Kurator

#### 3. Supporting Actor

none

#### 4. Basic Flow

1. Use Case ini dimulai ketika aktor memilih untuk melakukan login
2. Sistem menampilkan antarmuka untuk login
3. Aktor memasukkan username dan password
4. Sistem memeriksa username dan password yang dimasukkan aktor  
E-1 Password atau username tidak sesuai
5. Sistem memberikan akses ke aktor
6. Use Case ini selesai

#### 5. Alternative Flow

none

#### 6. Error Flow

- E-1 Password atau username tidak sesuai
1. Sistem menampilkan peringatan bahwa username atau password tidak sesuai
  2. Kembali ke Basic Flow langkah ke 3

#### 7. PreConditions

1. Aktor sudah mendaftar

Program Studi Teknik Informatika	SKPL – CariEdu	19/ 35
----------------------------------	----------------	--------

## 8. PostConditions

1. Aktor memasuki sistem dan dapat menggunakan fungsi-fungsi pada sistem.

### 4.1.3 Use case Spesification : Login Anggota

#### 1. Brief Description

Use Case ini digunakan oleh aktor untuk memperoleh akses ke sistem. Login didasarkan pada sebuah id unik yaitu username dari aktor dan password yang berupa rangkaian karakter.

#### 2. Primary Actor

1. Anggota

#### 3. Supporting Actor

none

#### 4. Basic Flow

1. Use Case ini dimulai ketika aktor memilih untuk melakukan login
2. Sistem menampilkan antarmuka untuk login
3. Aktor memasukkan username dan password
4. Sistem memeriksa username dan password yang dimasukkan aktor
- E-1 Password atau username tidak sesuai
5. Sistem memberikan akses ke aktor
6. Use Case ini selesai

#### 5. Alternative Flow

none

#### 6. Error Flow

- E-1 Password atau username tidak sesuai
1. Sistem menampilkan peringatan bahwa username atau password tidak sesuai
  2. Kembali ke Basic Flow langkah ke 3

#### 7. PreConditions

1. Aktor sudah mendaftar

Program Studi Teknik Informatika	SKPL – CariEdu	20/ 35
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

## 8. PostConditions

1. Aktor memasuki sistem dan dapat menggunakan fungsi-fungsi pada sistem.

### 4.1.4 Use case Spesification : Tambah Edu

#### 1. Brief Description

Use Case ini digunakan oleh aktor untuk menambah *landmark* unit pendidikan yang dapat dilihat oleh semua anggota.

#### 2. Primary Actor

1. Administrator
2. Anggota
3. Kurator

#### 3. Supporting Actor

none

#### 4. Basic Flow

1. Use Case ini dimulai ketika aktor memilih untuk menambah *landmark* unit pendidikan.
2. Sistem menampilkan antarmuka untuk menambah *landmark* unit pendidikan.
3. Aktor memasukkan isian yang ada mengenai data unit pendidikan yang akan dibuat.
4. Aktor meminta sistem untuk menyimpan data *landmark* unit pendidikan yang telah dimasukkan.
5. Sistem menyimpan/menambahkan data *landmark* unit2 pendidikan ke database.
6. Use Case selesai.

#### 5. Alternative Flow

#### 6. Error Flow

#### 7. PreConditions

1. Use Case Login telah dilakukan
2. Aktor telah memasuki sistem

#### 8. PostConditions

1. Data *landmark* unit pendidikan di database telah terbaru

Program Studi Teknik Informatika	SKPL – CariEdu	21/ 35
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

#### **4.1.5 Use case Spesification : Sunting Edu**

##### **1. Brief Description**

Use Case ini digunakan oleh aktor untuk menyunting unit pendidikan yang telah dibuat aktor tersebut.

##### **2. Primary Actor**

1. Administrator
2. Anggota
3. Kurator

##### **3. Supporting Actor**

none

##### **4. Basic Flow**

1. Use Case ini dimulai ketika aktor memilih untuk menyunting unit pendidikan.
2. Sistem menampilkan antarmuka untuk menyunting unit pendidikan.
3. Aktor memasukkan data pada isian yang ada mengenai data unit pendidikan yang akan disunting.
4. Aktor meminta sistem untuk menyimpan data unit pendidikan yang telah dimasukkan.
5. Sistem membarui data unit pendidikan ke database.
6. Use Case selesai.

##### **5. Alternative Flow**

none

##### **6. Error Flow**

none

##### **7. PreConditions**

1. Use Case Login telah dilakukan
2. Aktor telah memasuki sistem

##### **8. PostConditions**

1. Data unit pendidikan di database telah terbaru

#### **4.1.6 Use case Spesification : Hapus Edu**

##### **1. Brief Description**

Use Case ini digunakan oleh aktor untuk menghapus unit pendidikan yang telah dibuat aktor tersebut.

##### **2. Primary Actor**

1. Administrator
2. Anggota
3. Kurator

##### **3. Supporting Actor**

none

##### **4. Basic Flow**

1. Use Case ini dimulai ketika aktor memilih untuk menghapus unit pendidikan.
2. Aktor menghapus unit pendidikan.
3. Sistem menghapus data unit pendidikan ke database.
4. Use Case selesai.

##### **5. Alternative Flow**

none

##### **6. Error Flow**

none

##### **7. PreConditions**

1. Use Case Login telah dilakukan
2. Aktor telah memasuki sistem

##### **8. PostConditions**

1. Data unit pendidikan di database telah terbaru

#### **4.1.7 Use case Spesification : Hapus Anggota**

##### **1. Brief Description**

Use Case ini digunakan oleh aktor untuk menghapus anggota yang ada.

##### **2. Primary Actor**

1. Administrator

##### **3. Supporting Actor**

none

Program Studi Teknik Informatika	SKPL – CariEdu	23/ 35
----------------------------------	----------------	--------

#### **4. Basic Flow**

1. Use Case ini dimulai ketika aktor memilih untuk menghapus anggota.
2. Sistem menampilkan daftar anggota.
3. Aktor memilih untuk menghapus anggota terpilih.
4. Sistem menghapus data anggota ke database
5. Use Case selesai

#### **5. Alternative Flow**

none

#### **6. Error Flow**

none

#### **7. PreConditions**

1. Use Case Login telah dilakukan
2. Aktor telah memasuki sistem

#### **8. PostConditions**

1. Data anggota di database telah terbaru

#### **4.1.8 Use case Spesification : Lihat Peta Edu**

##### **1. Brief Description**

Use Case ini memungkinkan aktor untuk melihat lokasi unit pendidikan pada peta.

##### **2. Primary Actor**

1. Administrator
2. Anggota
3. Kurator

##### **3. Supporting Actor**

none

##### **4. Basic Flow**

1. Use Case ini dimulai ketika aktor memilih untuk melihat posisi unit pendidikan pada peta
2. Sistem menampilkan daftar unit pendidikan yang ada
3. Aktor memilih salah satu unit pendidikan yang ada.

4. Sistem menampilkan lokasi unit pendidikan yang terpilih pada peta.

5. Use Case selesai

**5. Alternative Flow**

none

**6. Error Flow**

none

**7. PreConditions**

1. Use Case Login sudah dilakukan
2. Aktor telah memasuki sistem

**8. PostConditions**

1. Lokasi unit pendidikan pada peta ditampilkan

**4.1.9 Use case Spesification : Lihat Rute Edu**

**1. Brief Description**

Use Case ini memungkinkan aktor untuk melihat lokasi unit pendidikan pada peta.

**2. Primary Actor**

1. Administrator
2. Anggota
3. Kurator

**3. Supporting Actor**

none

**4. Basic Flow**

1. Use Case ini dimulai ketika aktor memilih untuk melihat rute menuju lokasi *landmark* unit pendidikan pada peta.
2. Sistem menampilkan daftar unit pendidikan yang ada.
3. Aktor memilih salah satu unit pendidikan yang ada.
4. Sistem menampilkan lokasi unit pendidikan yang terpilih pada peta.
5. Aktor memilih untuk melihat rute.

6. Sistem menampilkan rute menuju lokasi landmark unit pendidikan pada peta.

7. Use Case selesai

**5. Alternative Flow**

none

**6. Error Flow**

none

**7. PreConditions**

1. Use Case Login sudah dilakukan
2. Use Case Lihat Peta Edu sudah dilakukan
3. Aktor telah memasuki sistem

**8. PostConditions**

Rute menuju lokasi unit pendidikan pada peta ditampilkan

**4.1.10 Use case Spesification : Ganti Lokasi Tempat Tinggal**

**1. Brief Description**

Use Case ini memungkinkan aktor untuk melihat lokasi tempat tinggal serta mengubah/membarui lokasi tempat tinggal.

**2. Primary Actor**

1. Administrator
2. Anggota
3. Kurator

**3. Supporting Actor**

none

**4. Basic Flow**

1. Use Case ini dimulai ketika aktor memilih untuk mengganti lokasi tempat tinggal
2. Sistem menampilkan lokasi tempat tinggal dan posisi actor pada peta.

3. Aktor memilih untuk menyimpan lokasi tempat tinggal yang baru.
4. Sistem membarui lokasi tempat tinggal dalam basis data
5. Use Case selesai

**5. Alternative Flow**

none

**6. Error Flow**

none

**7. PreConditions**

1. Use Case Login sudah dilakukan
2. Aktor telah memasuki sistem

**8. PostConditions**

Lokasi tempat tinggal dalam basis data terbarui

**4.1.11 Use case Spesification : Lihat informasi Edu**

**1. Brief Description**

Use Case ini memungkinkan aktor untuk melihat informasi unit pendidikan.

**2. Primary Actor**

1. Administrator
2. Anggota
3. Kurator

**3. Supporting Actor**

none

**4. Basic Flow**

1. Use Case ini dimulai ketika aktor memilih untuk melihat informasi unit pendidikan.
2. Sistem menampilkan daftar unit pendidikan yang ada.
3. Aktor memilih salah satu unit pendidikan yang ada.
4. Sistem menampilkan informasi unit pendidikan yang terpilih.
5. Use Case selesai

#### **5. Alternative Flow**

none

#### **6. Error Flow**

None

#### **7. PreConditions**

1. Use Case Login sudah dilakukan
2. Aktor telah memasuki sistem

#### **8. PostConditions**

1. Informasi tentang unit pendidikan ditampilkan

### **4.1.12 Use case Spesification : Lihat informasi Anggota**

#### **1. Brief Description**

Use Case ini memungkinkan aktor untuk melihat informasi anggota.

#### **2. Primary Actor**

1. Administrator

#### **3. Supporting Actor**

none

#### **4. Basic Flow**

1. Use Case ini dimulai ketika aktor memilih untuk melihat informasi anggota.
2. Sistem menampilkan daftar anggota.
3. Aktor memilih salah satu anggota yang ada.
4. Sistem menampilkan informasi anggota yang terpilih.
5. Use Case selesai

#### **5. Alternative Flow**

none

#### **6. Error Flow**

none

#### **7. PreConditions**

1. Use Case Login sudah dilakukan
2. Aktor telah memasuki system

## 8. PostConditions

1. Informasi tentang anggota ditampilkan

### 4.1.13 Use case Spesification : Ganti peran anggota

#### 1. Brief Description

Use Case ini memungkinkan aktor untuk mengganti peran dari anggota menjadi kurator dan sebaliknya.

#### 2. Primary Actor

1. Administrator

#### 3. Supporting Actor

none

#### 4. Basic Flow

1. Use Case ini dimulai ketika aktor memilih untuk mengganti peran anggota.
2. Sistem menampilkan daftar anggota.
3. Aktor memilih salah satu anggota yang ada.
4. Sistem menampilkan informasi anggota yang terpilih.
5. Aktor memilih untuk mengganti peran anggota.
6. Sistem membarui data peran anggota yang ada dalam basis data.
7. Use Case selesai

#### 5. Alternative Flow

none

#### 6. Error Flow

none

#### 7. PreConditions

1. Use Case Login sudah dilakukan
2. Use Case Lihat Informasi Anggota sudah dilakukan
3. Aktor telah memasuki sistem

#### 8. PostConditions

Data peran dari anggota diperbarui

#### **4.1.14 Use case Spesification : Beri komentar**

##### **1. Brief Description**

Use Case ini memungkinkan aktor untuk menambahkan komentar pada unit pendidikan yang ada.

##### **2. Primary Actor**

1. Administrator
2. Anggota
3. Kurator

##### **3. Supporting Actor**

none

##### **4. Basic Flow**

1. Use Case ini dimulai ketika aktor memilih untuk menambahkan komentar.
2. Sistem menampilkan daftar unit pendidikan yang ada.
3. Aktor memilih salah satu unit pendidikan.
4. Sistem menampilkan antar muka untuk menambahkan komentar.
5. Aktor memasukkan komentar.
6. Aktor meminta sistem untuk menyimpan komentar baru.
7. Sistem menyimpan komentar ke dalam basis data.

##### **5. Alternative Flow**

##### **6. Error Flow**

##### **7. PreConditions**

1. Use Case Login sudah dilakukan
2. Use Case Lihat Informasi Edu sudah dilakukan
3. Aktor telah memasuki sistem

##### **8. PostConditions**

1. komentar terbaru

#### **4.1.15 Use case Spesification : Beri Nilai**

##### **1. Brief Description**

Use Case ini memungkinkan aktor untuk member nilai pada unit pendidikan yang ada.

##### **2. Primary Actor**

1. Administrator
2. Anggota
3. Kurator

##### **3. Supporting Actor**

none

##### **4. Basic Flow**

1. Use Case ini dimulai ketika aktor memilih untuk menambahkan nilai.
2. Sistem menampilkan daftar unit pendidikan yang ada.
3. Aktor memilih salah satu unit pendidikan.
4. Sistem menampilkan antar muka untuk menambahkan nilai.
5. Aktor memasukkan nilai.
6. Aktor meminta sistem untuk menyimpan nilai baru.
7. Sistem menyimpan/membarui nilai ke dalam basis data.

##### **5. Alternative Flow**

none

##### **6. PreConditions**

1. Use Case Login sudah dilakukan
2. Use Case Lihat Informasi Edu sudah dilakukan
3. Aktor telah memasuki sistem

##### **7. PostConditions**

1. nilai terbaru

#### **4.1.16 Use case Spesification : Melakukan pencarian**

##### **1. Brief Description**

Use Case ini memungkinkan actor untuk melakukan pencarian terhadap unit pendidikan yang ada.

##### **2. Primary Actor**

1. Anggota
2. Administrator
3. Kurator

##### **3. Supporting Actor**

none

##### **4. Basic Flow**

1. Use Case ini dimulai ketika aktor memilih untuk melakukan pencarian.
2. Sistem akan menampilkan antar muka untuk melakukan pencarian.
3. Aktor memasukkan informasi mengenai pencarian yang diinginkan.
4. Aktor meminta sistem untuk melakukan pencarian.
5. Sistem menampilkan daftar hasil pencarian.
6. Use Case selesai

##### **5. Alternative Flow**

none

##### **6. Error Flow**

none

##### **7. PreConditions**

1. Use Case Login sudah dilakukan
2. Aktor telah memasuki sistem

##### **8. PostConditions**

1. Hasil pencarian ditampilkan

**4.1.17 Use case Spesification** : Menyetujui peletakan unit pendidikan

**1. Brief Description**

Use Case ini memungkinkan aktor untuk menyetujui permintaan peletakan *landmark* unit pendidikan.

**2. Primary Actor**

1. Administrator
2. Kurator

**3. Supporting Actor**

none

**4. Basic Flow**

1. Use Case ini dimulai ketika Aktor memilih untuk menyetujui permintaan peletakan *landmark* unit pendidikan.
2. Sistem menampilkan daftar permintaan yang belum disetujui
3. Aktor memilih salah satu permintaan peletakan *landmark* unit pendidikan.
4. Sistem akan membarui data unit pendidikan dan menghapus permintaan dari daftar permintaan
5. Use Case selesai

**5. Alternative Flow**

none

**6. Error Flow**

none

**7. PreConditions**

1. Use Case Login sudah dilakukan
2. Aktor telah memasuki sistem

**8. PostConditions**

1. Data permintaan peletakan *landmark* unit pendidikan terbaru

#### **4.1.18 Use case Spesification : Mendaftar**

##### **1. Brief Description**

Use Case ini memungkinkan aktor untuk mendaftar menjadi anggota.

##### **2. Primary Actor**

1. Pengunjung

##### **3. Supporting Actor**

none

##### **4. Basic Flow**

1. Use Case ini dimulai ketika Aktor memilih untuk mendaftar.
2. Sistem menampilkan antar muka pendaftaran
3. Aktor mengisi informasi yang diperlukan untuk mendaftar.
4. Aktor meminta untuk menyimpan masukan Aktor
5. Sistem memeriksa masukan Aktor  
E-1 masukan data salah
6. Sistem menyimpan masukan Aktor
7. Use Case selesai

##### **5. Alternative Flow**

none

##### **6. Error Flow**

E-1 masukan data salah

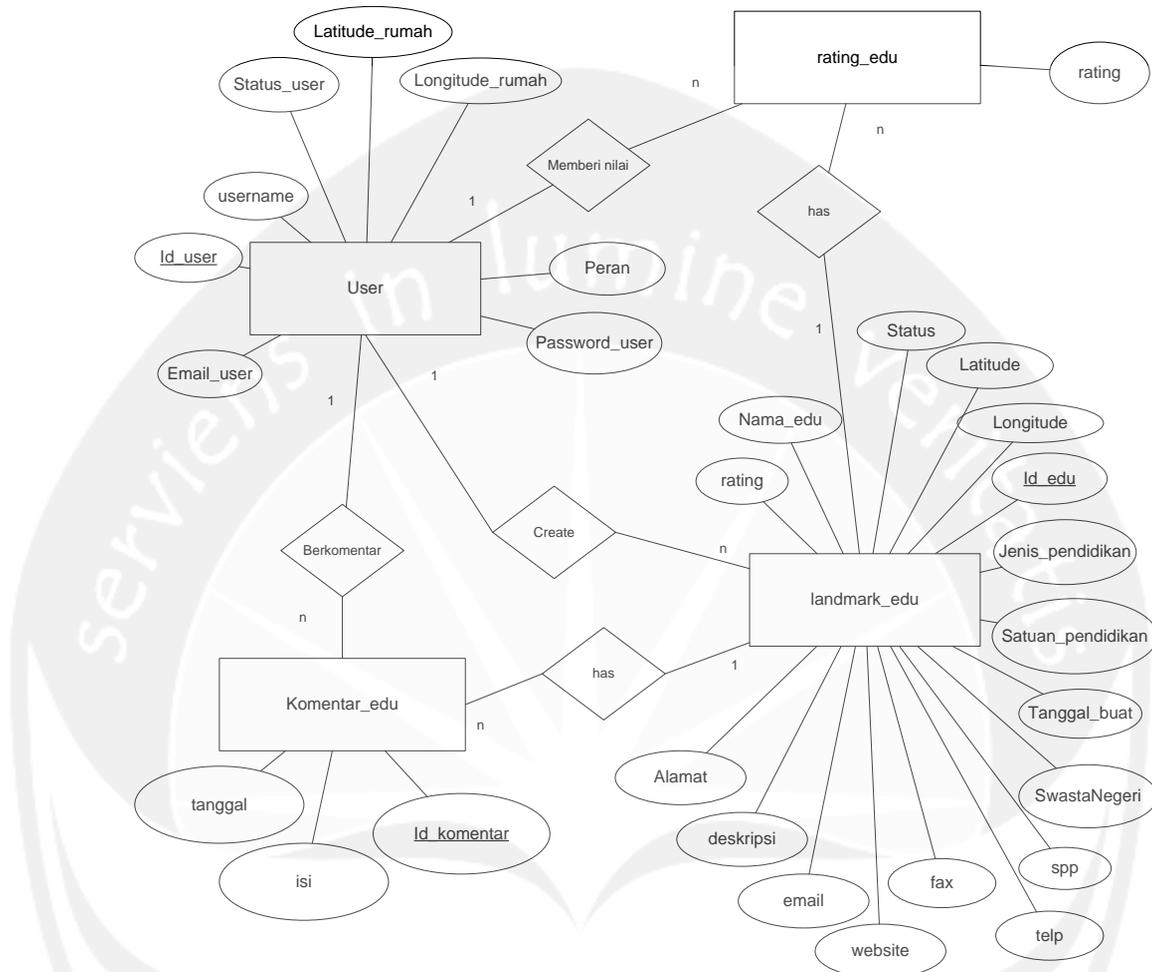
1. Sistem memberitahu Aktor bahwa masukan data salah
2. Flow kembali ke Basic Flow langkah ke-2

##### **7. PreConditions**

##### **8. PostConditions**

1. Data pendaftaran diproses

## 5 Entity Relationship Diagram (ERD)



Gambar 3 Entity Relation Diagram

# DPPL

## DESKRIPSI PERANCANGAN PERANGKAT LUNAK

**CariEdu**

Untuk:

**Universitas Atma Jaya Yogyakarta**

Dipersiapkan oleh:

**Eduardo Aji Pradana / 070705215**

**Program Studi Teknik Informatika - Fakultas Teknologi Industri**

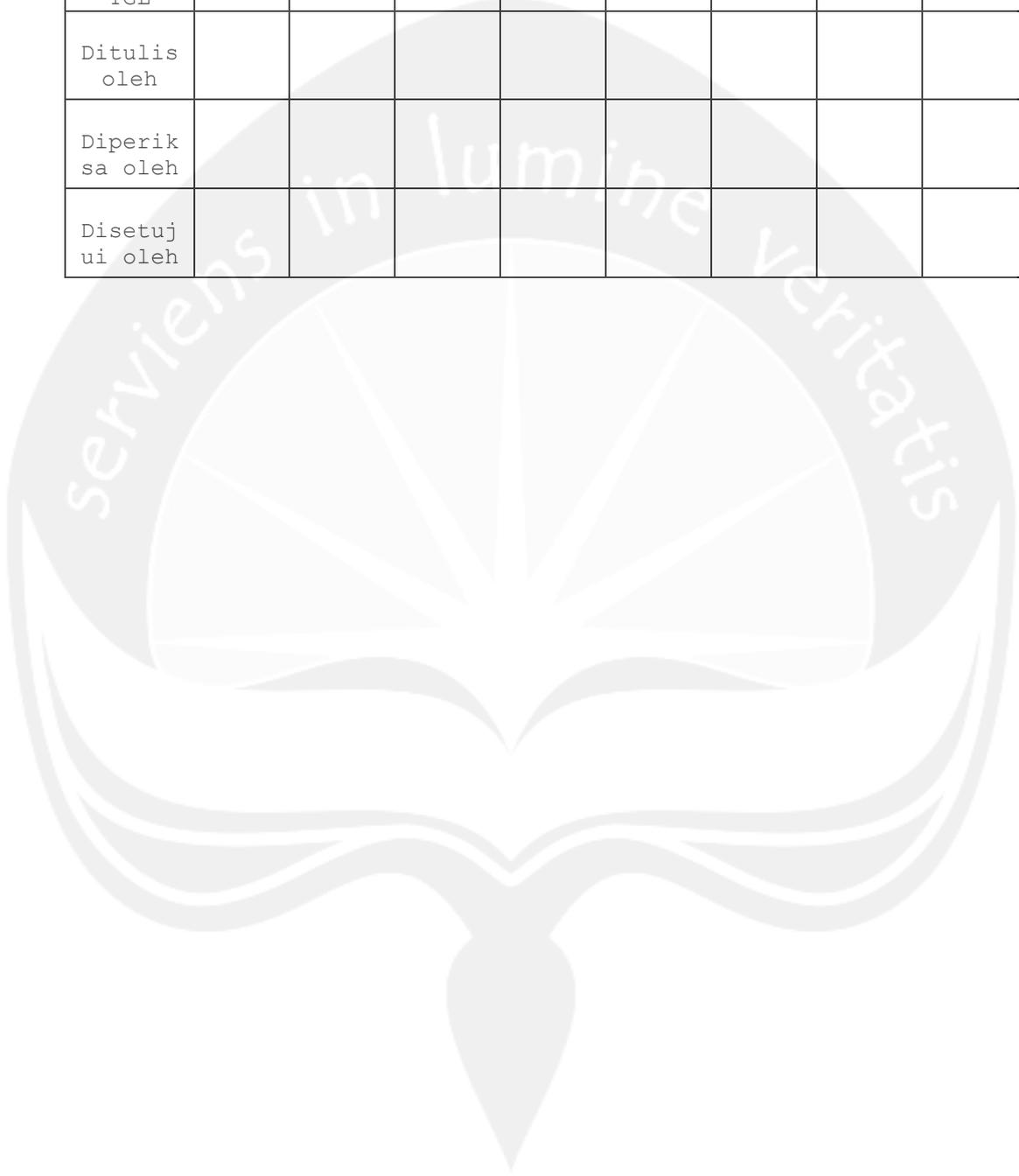
**Universitas Atma Jaya Yogyakarta**

	Program Studi Teknik Informatika Fakultas Teknologi Industri	Nomor Dokumen		Halaman
		<b><i>DPPL-CariEdu</i></b>		1/69
		Revisi		

## DAFTAR PERUBAHAN

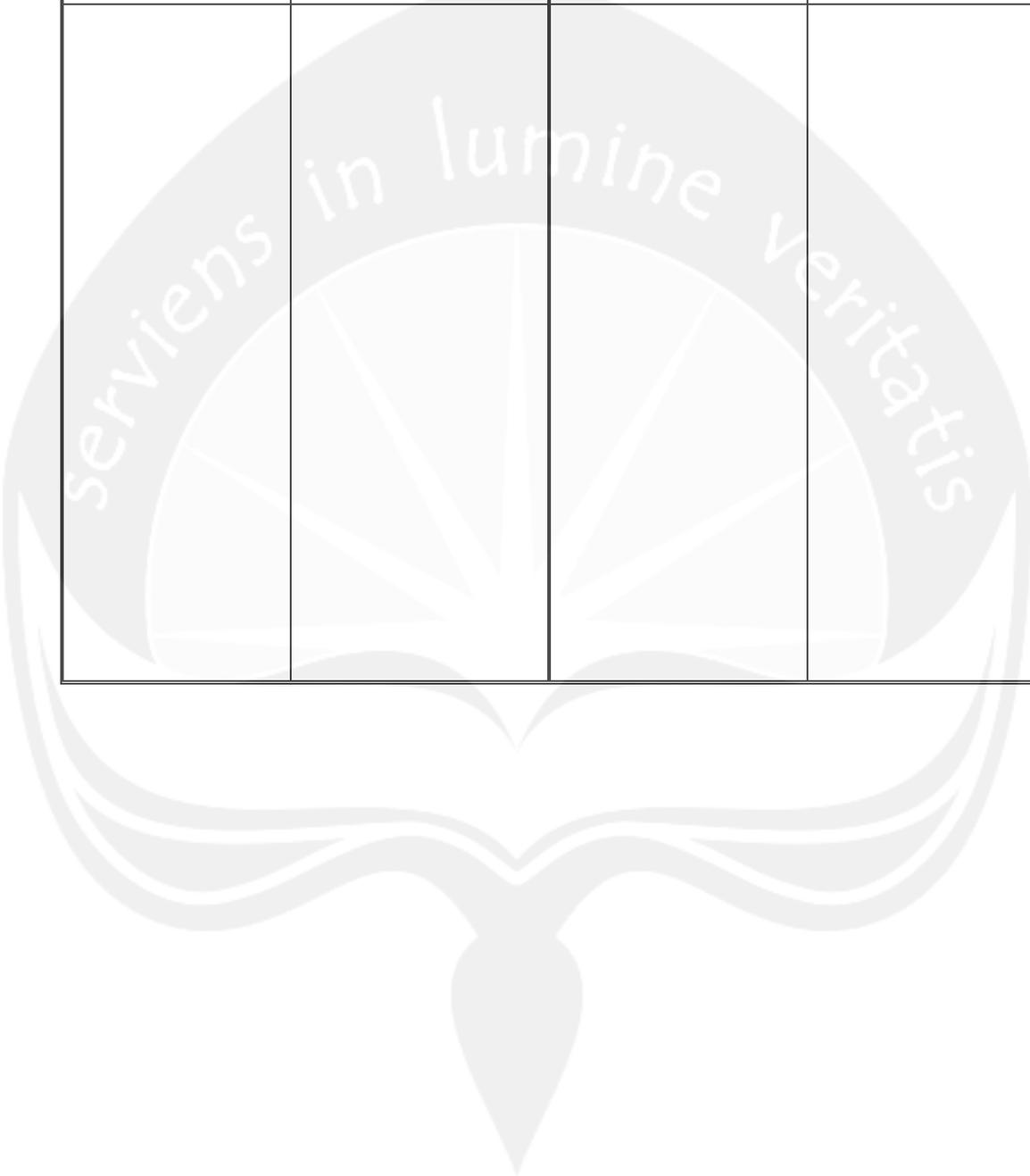
Revisi	Deskripsi
A	
B	
C	
D	
E	
F	

INDEX TGL	-	A	B	C	D	E	F	G
Ditulis oleh								
Diperik sa oleh								
Disetuj ui oleh								



### Daftar Halaman Perubahan

Halaman	Revisi	Halaman	Revisi



## Daftar Isi

1	Pendahuluan .....	9
1.1	Tujuan .....	9
1.2	Ruang Lingkup.....	9
1.3	Definisi dan Akronim.....	10
1.4	Referensi.....	11
2	Perancangan System.....	12
2.1	Perancangan Architecture.....	12
2.2	Perancangan Rinci .....	13
2.2.1.1	Login .....	13
2.2.1.1.1	Login Administrator.....	13
2.2.1.1.2	Login Anggota .....	14
2.2.1.2	Lihat Lokasi Tempat Tinggal .....	14
2.2.1.2.2	Daftar.....	15
2.2.1.3	Lihat detail permintaan.....	20
2.2.1.4	Setujui Permintaan.....	21
2.2.1.5	Hapus Edu.....	22
2.2.1.6	Hapus Anggota .....	23
2.2.1.7	Lihat Peta .....	24
2.2.1.8	Lihat List Permintaan.....	25
2.2.1.9	Beri rating.....	26
2.3	Class Diagram.....	27
2.4	Class Diagram Specific Descriptions.....	29
	Specific Design Class MainMenuUI .....	30
	Specific Design Class KatalogUI .....	30
	Specific Design Class ListEduUI .....	30
	Specific Design Class ListlayananUI.....	31
	Specific Design Class ListAnggotaUI.....	31
	Specific Design Class ListPermintaanUI.....	32
	Specific Design Class DetailEduUI.....	32
	Specific Design Class DetailAnggotaUI .....	33
	Specific Design Class DetailPermintaanUI.....	33
	Specific Design Class KomentarUI .....	34
	Specific Design Class ListSearchEduUI .....	34
	Specific Design Class PetaEduUI .....	35
	Specific Design Class SearchUI.....	35
	Specific Design Class RatingUI.....	35
	Specific Design Class TambahEduUI.....	36
	Specific Design Class SuntungEduUI .....	36

Specific Design Class EduControl.....	37
Specific Design Class UserControl.....	39
Specific Design Class LoginControl.....	40
Specific Design Class KomentarControl.....	41
Specific Design Class RatingControl.....	41
Specific Design Class DataEdu.....	42
Specific Design Class DataUser.....	45
Specific Design DataKomentar.....	47
Specific Design DataRating.....	47
3 Perancangan Data.....	48
3.1 Dekomposisi Data.....	48
3.1.1 Deskripsi Entitas Data Pengguna.....	48
3.1.2 Deskripsi Entitas Data Edu.....	48
3.1.3 Deskripsi Entitas Data Komentar.....	50
3.1.4 Deskripsi Entitas Data Rating.....	50
4 Deskripsi Perancangan Antarmuka.....	52
4.1 Log in Anggota.....	52
4.2 Log in Administrator.....	53
4.3 Log in Kurator.....	54
4.4 Menu Utama.....	55
4.5 Detail Anggota.....	56
4.6 Detail Permintaan.....	57
4.7 Detail Edu.....	58
4.8 Rating.....	59
4.9 Peta Edu.....	60
4.10 List Edu.....	61
4.11 List Layanan.....	62
4.12 List Anggota.....	63
4.13 List Hasil Pencarian.....	64
4.14 Pencarian.....	65
4.15 Katalog.....	66
4.16 Komentar.....	67
4.17 Tempat Tinggal.....	68
4.18 Daftar.....	69

## Daftar Gambar

Gambar 1 Rancangan Arsitektur CariEdu.....	12
Gambar 2.1 Sequence Diagram: Login Administrator.....	13
Gambar 2.2 Sequence Diagram: Login Anggota.....	14
Gambar 2.3 Sequence Diagram: Lihat lokasi tempat tinggal.....	14
Gambar 2.4 Sequence Diagram: Perbarui Tempat Tinggal..	15
Gambar 2.5 Sequence Diagram: Daftar.....	15
Gambar 2.6 Sequence Diagram : Lihat daftar edu.....	16
Gambar 2.7 Sequence Diagram: Lihat Detail Edu.....	16
Gambar 2.8 Sequence Diagram: Lihat Komentar.....	17
Gambar 2.9 Sequence Diagram: Kirim Komentar.....	17
Gambar 2.10 Sequence Diagram: Tambah Edu.....	18
Gambar 2.11 Sequence Diagram: Sunting Edu.....	18
Gambar 2.12 Sequence Diagram: Lihat Detail Anggota.....	19
Gambar 2.13 Sequence Diagram: Ganti peran anggota.....	20
Gambar 2.14 Sequence Diagram: lihat detail permintaan	20
Gambar 2.15 Sequence Diagram: Setujui permintaan.....	21
Gambar 2.16 Sequence Diagram: Hapus Edu.....	22
Gambar 2.17 Sequence Diagram : Hapus Anggota.....	23
Gambar 2.18 Sequence Diagram: Lihat Peta.....	24
Gambar 2.18 Sequence Diagram: Lihat List Permintaan...	25
Gambar 2.18 Sequence Diagram: Beri rating.....	26
Gambar 3.1 Gambar PDM.....	51
Gambar 4.1 Rancangan Antarmuka Login Anggota.....	52
Gambar 4.2 Rancangan Antarmuka Login Administrator.....	53
Gambar 4.3 Rancangan Antarmuka Login Kurator.....	54
Gambar 4.4 Rancangan Antarmuka Menu Utama.....	55
Gambar 4.5 Rancangan Antarmuka Detail Anggota.....	56
Gambar 4.6 Rancangan Antarmuka Detail Permintaan.....	57
Gambar 4.7 Rancangan Antarmuka Detail Edu.....	58
Gambar 4.8 Rancangan Antarmuka Rating.....	59
Gambar 4.9 Rancangan Antarmuka Peta Edu.....	60
Gambar 4.10 Rancangan Antarmuka List Edu.....	61
Gambar 4.11 Rancangan Antarmuka List layanan.....	62
Gambar 4.12 Rancangan Antarmuka List Anggota.....	63
Gambar 4.13 Rancangan Antarmuka List hasil Pencarian	
64	
Gambar 4.14 Rancangan Antarmuka Pencarian.....	65

Gambar 4.15 Rancangan Antarmuka Katalog.....	66
Gambar 4.16 Rancangan Antarmuka Komentar.....	67
Gambar 4.17 Rancangan Antarmuka Tempat Tinggal.....	68
Gambar 4.18 Rancangan Antarmuka Daftar Anggota.....	69



## **1 Pendahuluan**

### **1.1 Tujuan**

Dokumen Deskripsi Perancangan Perangkat Lunak (DPPL) bertujuan untuk mendefinisikan perancangan perangkat lunak yang akan dikembangkan. Dokumen DPPL tersebut digunakan oleh pengembang perangkat lunak sebagai acuan untuk implementasi pada tahap selanjutnya.

Dokumen ini menjelaskan tentang Tujuan, Ruang Lingkup, Definisi dan akronim, Referensi. Juga menjelaskan tentang Arsitektur, Sequence Diagram, Class description, Class diagram.

### **1.2 Ruang Lingkup**

Perangkat Lunak CariEdu dikembangkan dengan tujuan untuk:

1. Memberikan informasi dan posisi mengenai sampel unit-unit pendidikan di kota Yogyakarta.
2. Mencari dan menyaring unit pendidikan.
3. Memberikan rute dari satu lokasi ke unit pendidikan terpilih.
4. Menyediakan fasilitas untuk menambahkan informasi dan posisi unit pendidikan.

### 1.3 Definisi dan Akronim

Daftar definisi akronim dan singkatan:

Keyword/Phrase	Definisi
SKPL	SKPL (Spesifikasi Kebutuhan perangkat Lunak) Merupakan spesifikasi kebutuhan dari perangkat lunak yang akan dikembangkan.
CariEdu	Perangkat lunak yang menyediakan informasi dan lokasi dari unit pendidikan serta memungkinkan penggunaannya untuk menambahkan dan menyunting informasi dari unit pendidikan.
Database	Database atau basis data adalah kumpulan rekaman/data yang terintegrasi dalam wadah umum yang menyediakan data untuk satu atau lebih kebutuhan penggunaan.
Pengguna	Pengguna adalah administrator, Kurator dan anggota.
Kurator	Kurator adalah aktor yang berhak menyetujui atau menolak permintaan.
Administrator	Administrator adalah aktor yang berhak mengelola <i>landmark</i> dan pengguna dalam CariEdu.
Anggota	Anggota adalah aktor yang telah terdaftar dalam CariEdu.
Edu	Edu adalah istilah yang digunakan dalam perangkat lunak CariEdu yang sama artinya dengan unit pendidikan.

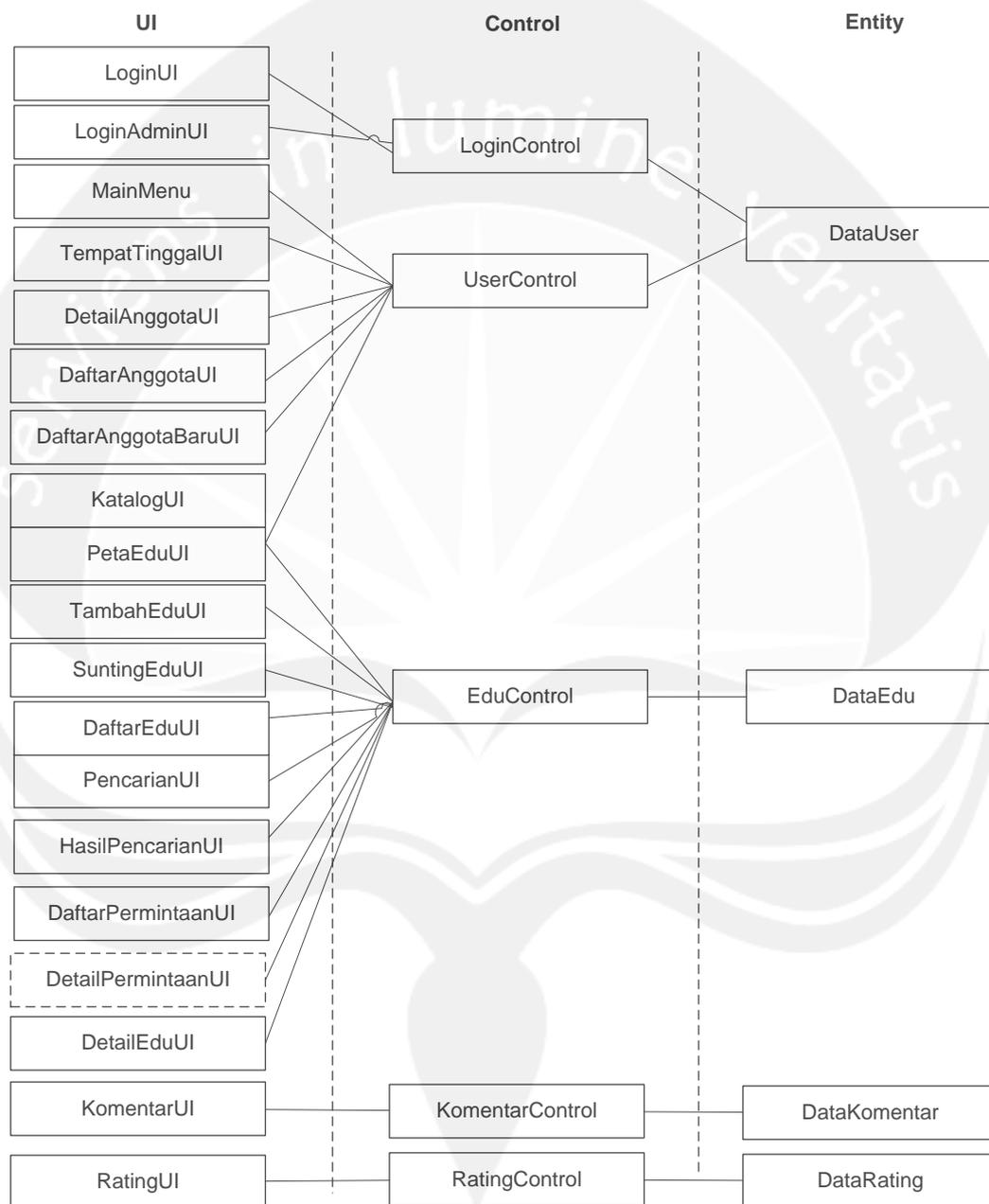
#### 1.4 Referensi

Referensi yang digunakan pada perangkat lunak tersebut adalah:

1. Bennet Simon, McRobb Steve, Farmer Ray, *Object-Oriented System Analysis and Design Using UML*, McGraw-Hill Companies, 2002.
2. Boggs Wendy, Boggs Michael, *Mastering UML with Rational Rose 2002*, SYBEX Inc, 2002.
3. Sapta Juli, Deskripsi Perancangan Perangkat Lunak SC3, Universitas Atma Jaya Yogyakarta, 2006.
4. Eduardo Aji Pradana, *Spesifikasi Kebutuhan Perangkat Lunak SIWABIMA*, Universitas Atma Jaya Yogyakarta, 2006.

## 2 Perancangan System

### 2.1 Perancangan Architecture



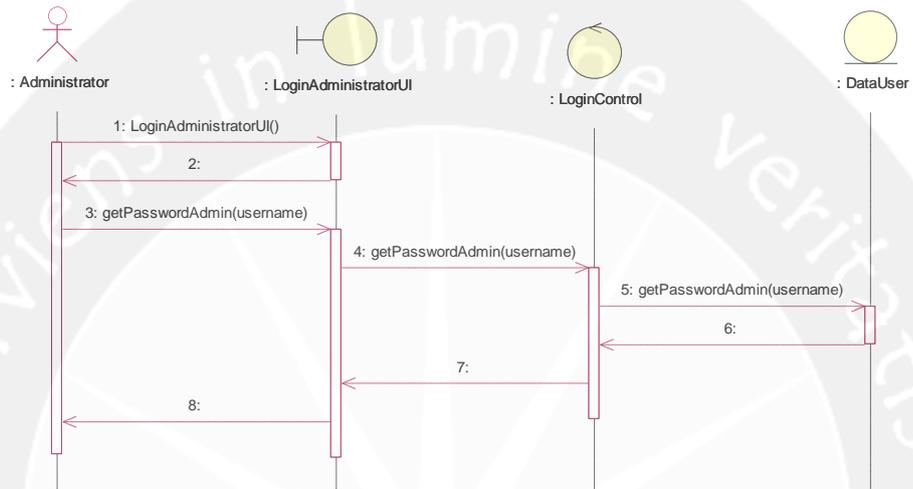
Gambar 1 Rancangan Arsitektur CariEdu

## 2.2 Perancangan Rinci

### 2.2.1 Sequence Diagram

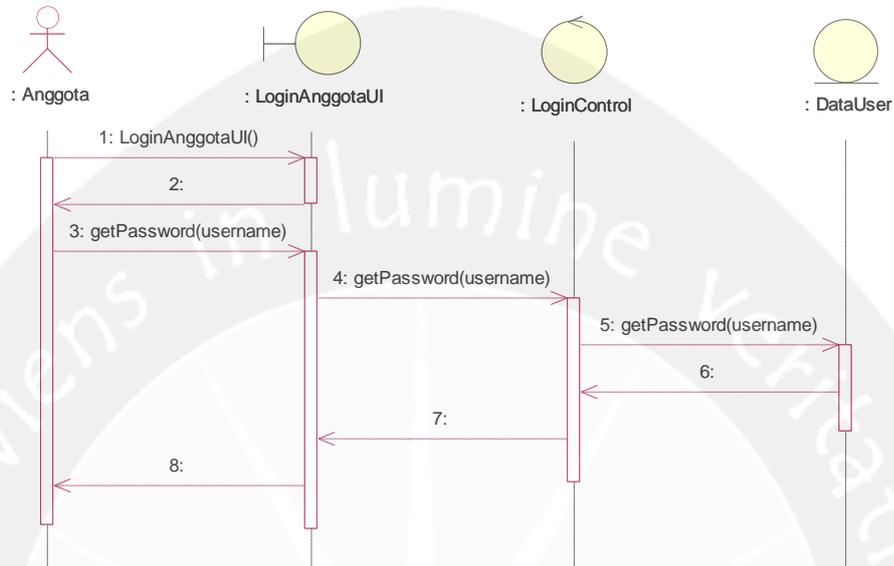
#### 2.2.1.1 Login

##### 2.2.1.1.1 Login Administrator



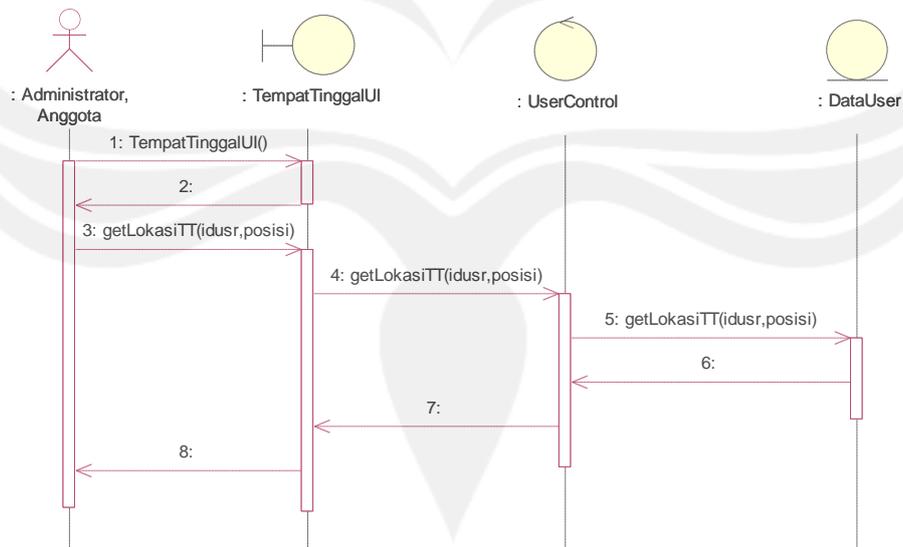
Gambar 2.1 Sequence Diagram: Login Administrator

### 2.2.1.1.2 Login Anggota



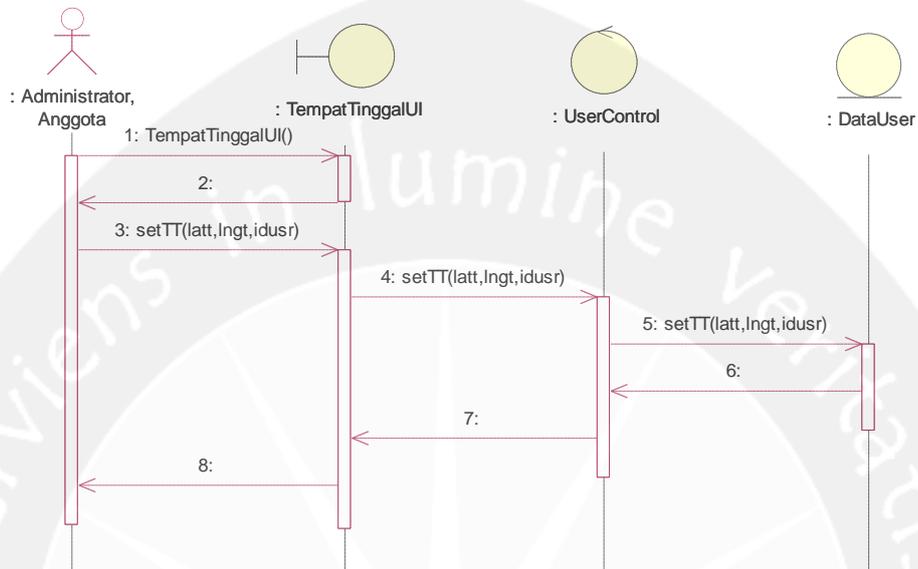
Gambar 2.2 Sequence Diagram: Login Anggota

### 2.2.1.2 Lihat Lokasi Tempat Tinggal



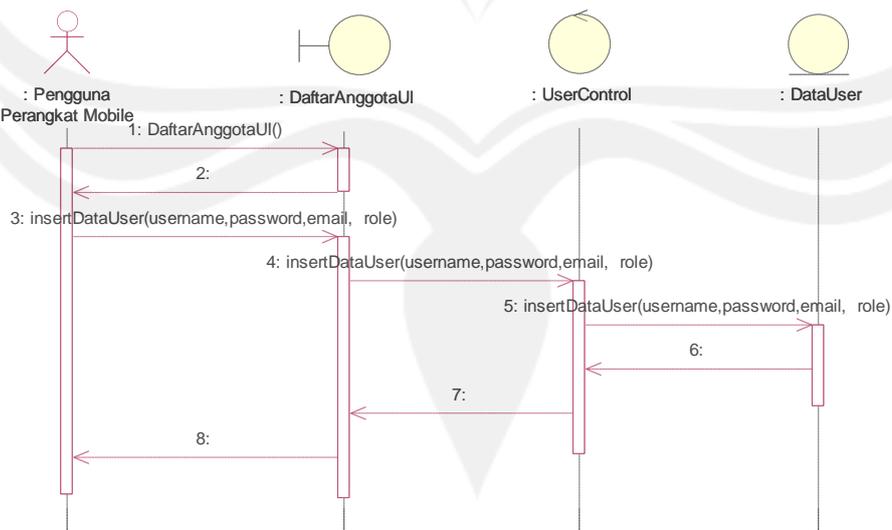
Gambar 2.3 Sequence Diagram: Lihat lokasi tempat tinggal

### 2.2.1.2.1 Perbarui Tempat Tinggal



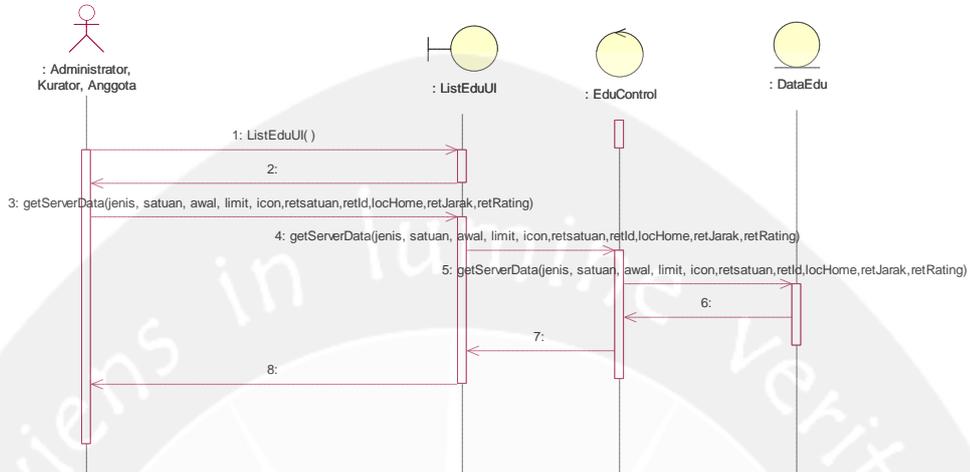
Gambar 2.4 Sequence Diagram: Perbarui Tempat Tinggal

### 2.2.1.2.2 Daftar



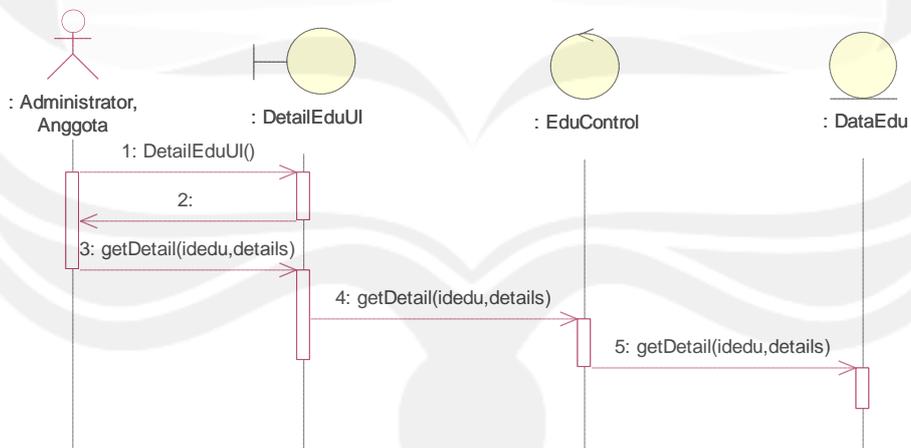
Gambar 2.5 Sequence Diagram: Daftar

### 2.2.1.2.3 Lihat List Edu



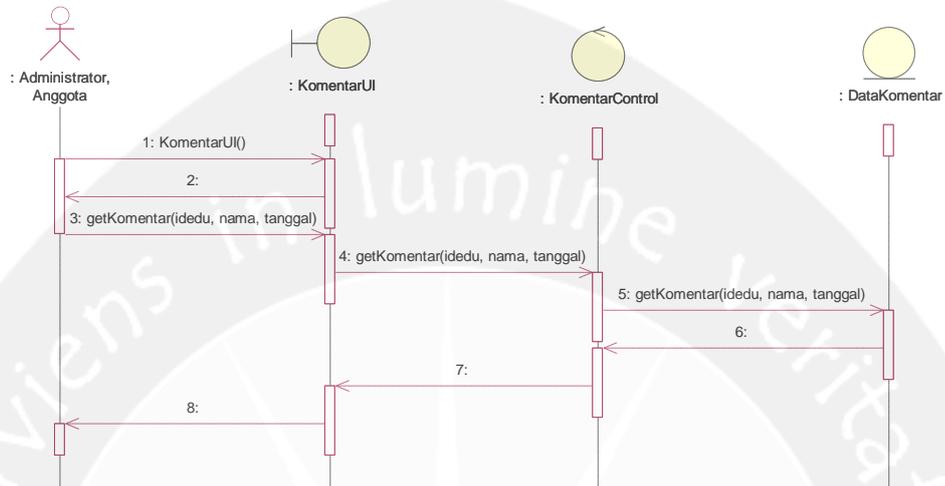
Gambar 2.6 Sequence Diagram : Lihat daftar edu

### 2.2.1.2.4 Lihat Detail Edu



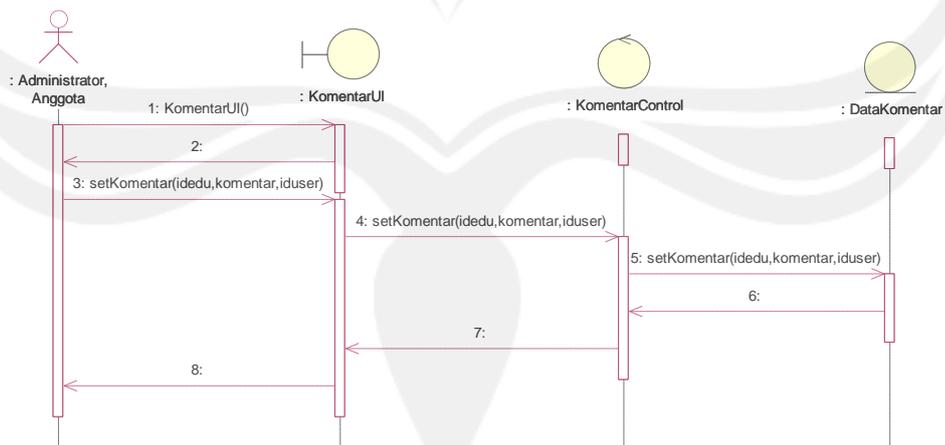
Gambar 2.7 Sequence Diagram: Lihat Detail Edu

### 2.2.1.2.5 Lihat Komentar



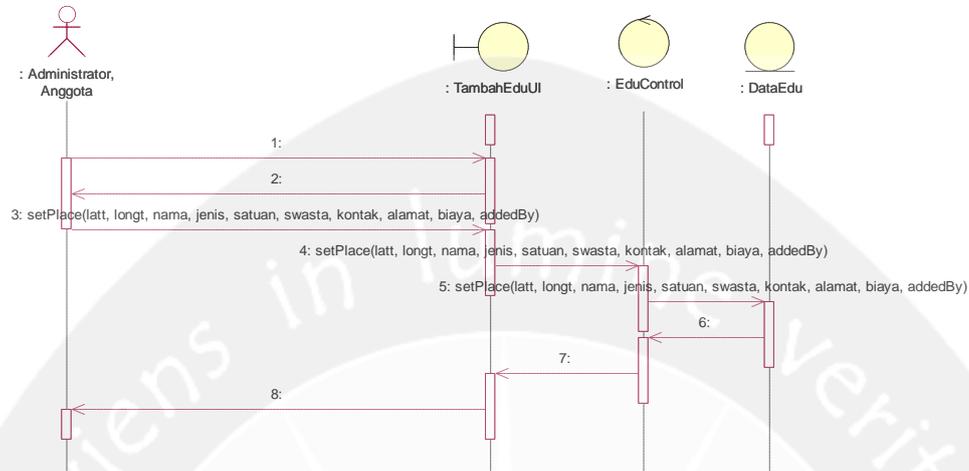
Gambar 2.8 Sequence Diagram: Lihat Komentar

### 2.2.1.2.6 Kirim Komentar



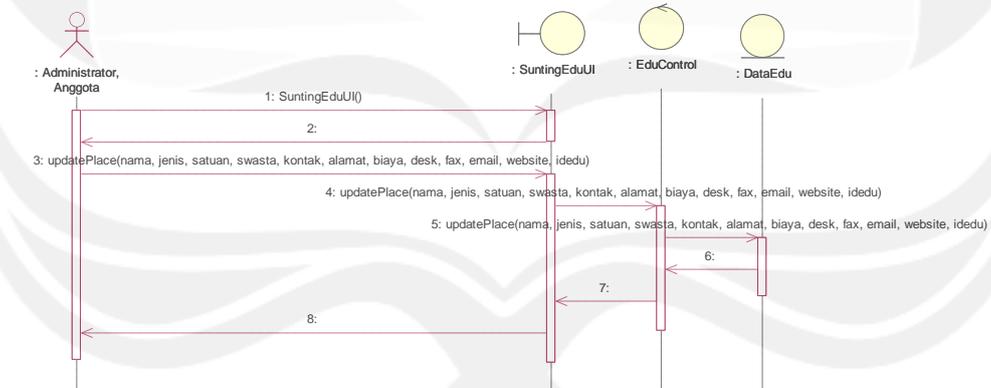
Gambar 2.9 Sequence Diagram: Kirim Komentar

### 2.2.1.2.7 Tambah Edu



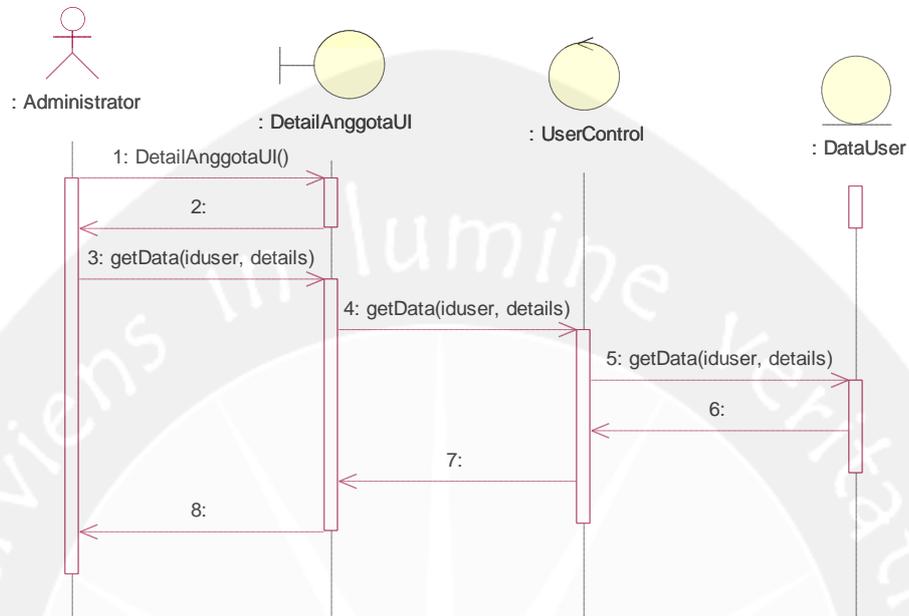
Gambar 2.10 Sequence Diagram: Tambah Edu

### 2.2.1.2.8 Sunting Edu



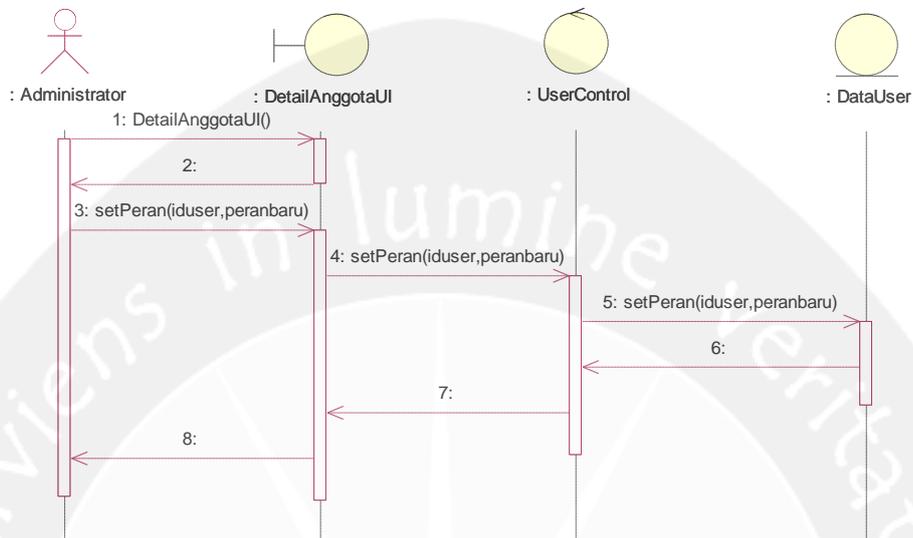
Gambar 2.11 Sequence Diagram: Sunting Edu

### 2.2.1.2.9 Lihat Detail Anggota



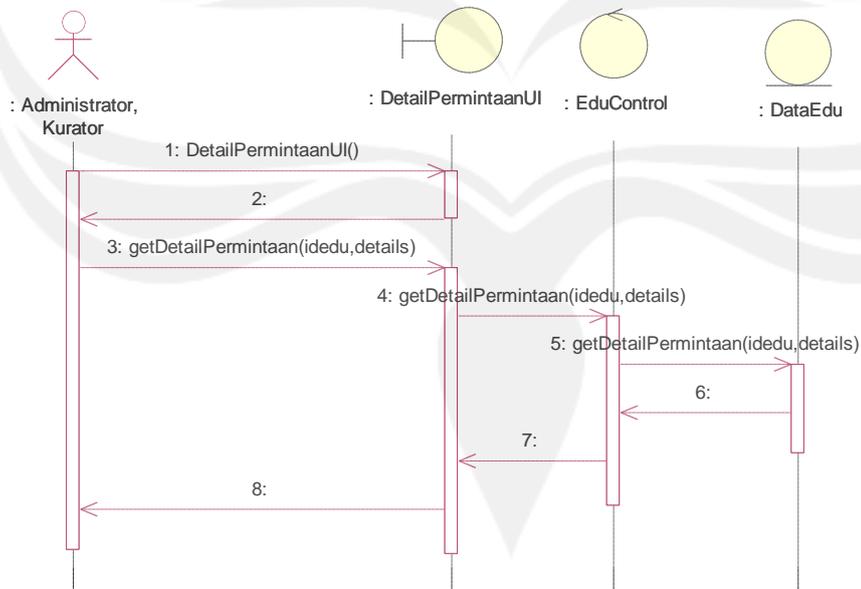
Gambar 2.12 Sequence Diagram: Lihat Detail Anggota

### 2.2.1.2.10 Ganti peran anggota



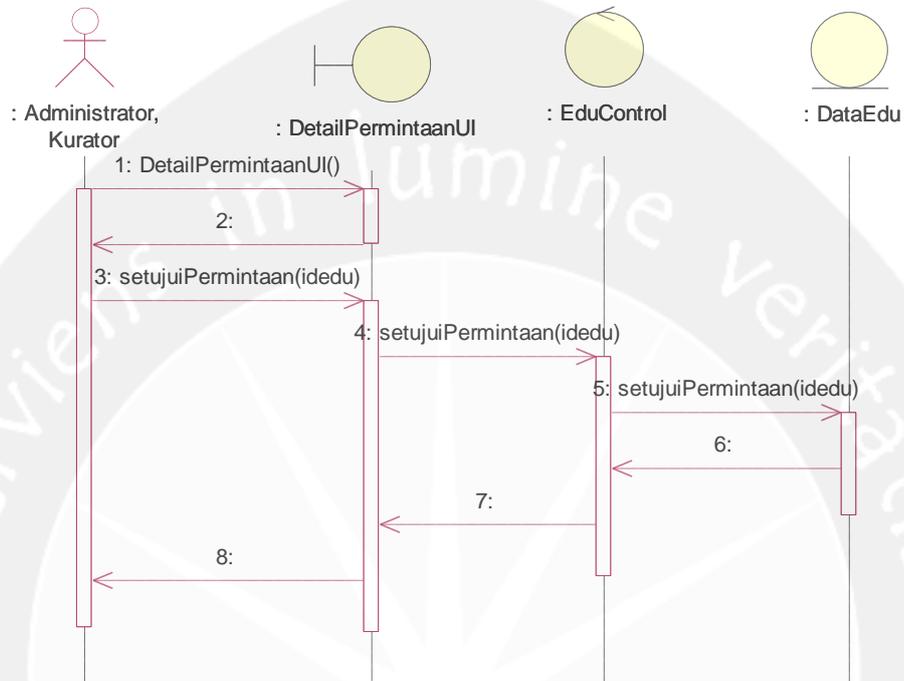
Gambar 2.13 Sequence Diagram: Ganti peran anggota

### 2.2.1.3 Lihat detail permintaan



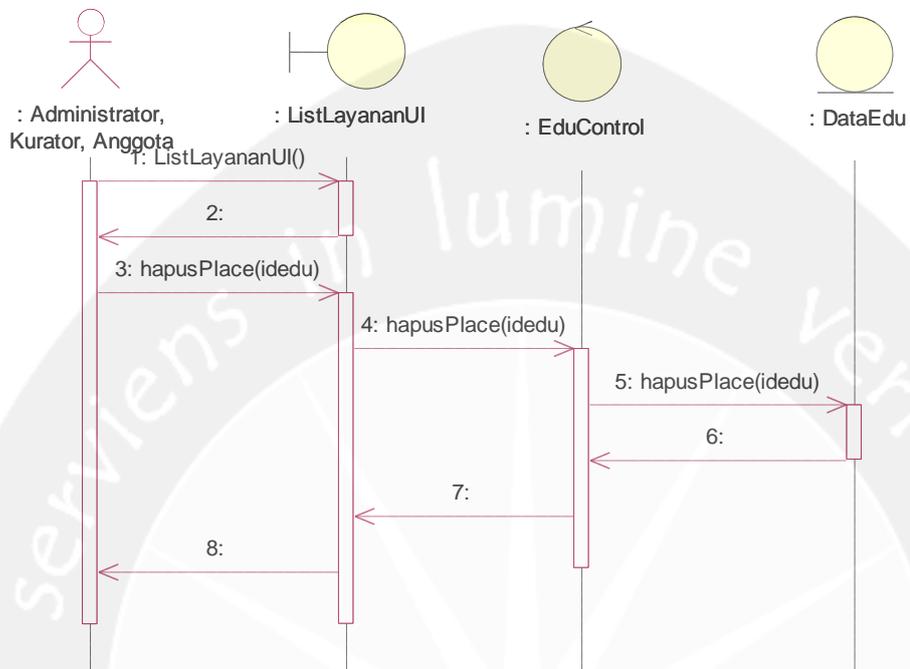
Gambar 2.14 Sequence Diagram: lihat detail permintaan

### 2.2.1.4 Setujui Permintaan



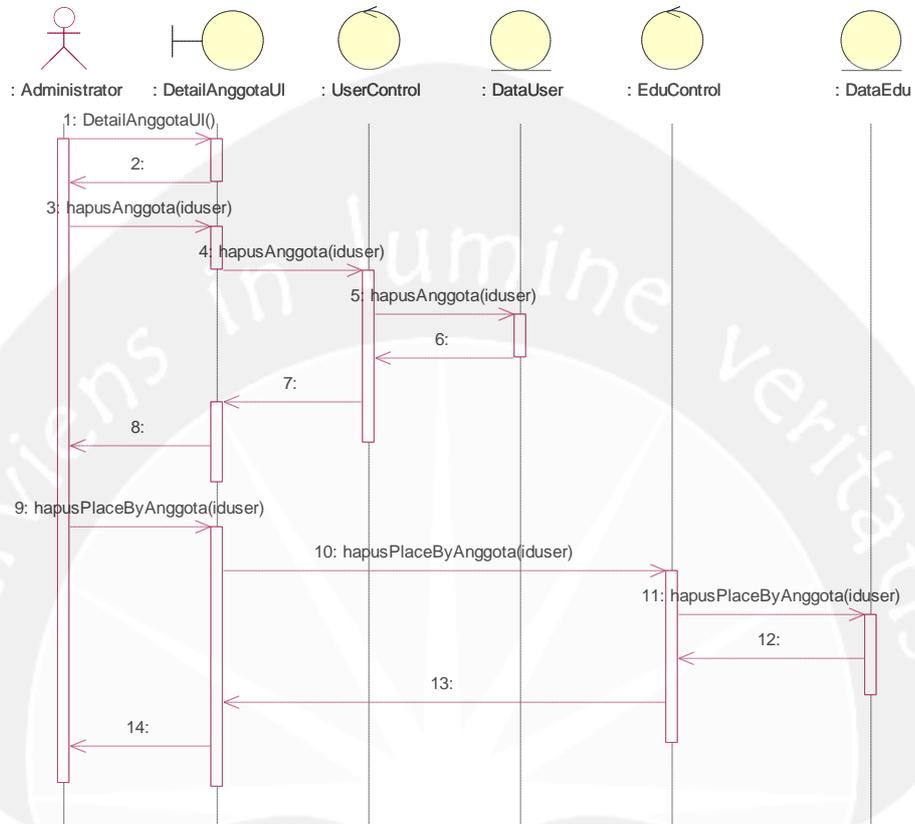
Gambar 2.15 Sequence Diagram: Setujui permintaan

### 2.2.1.5 Hapus Edu



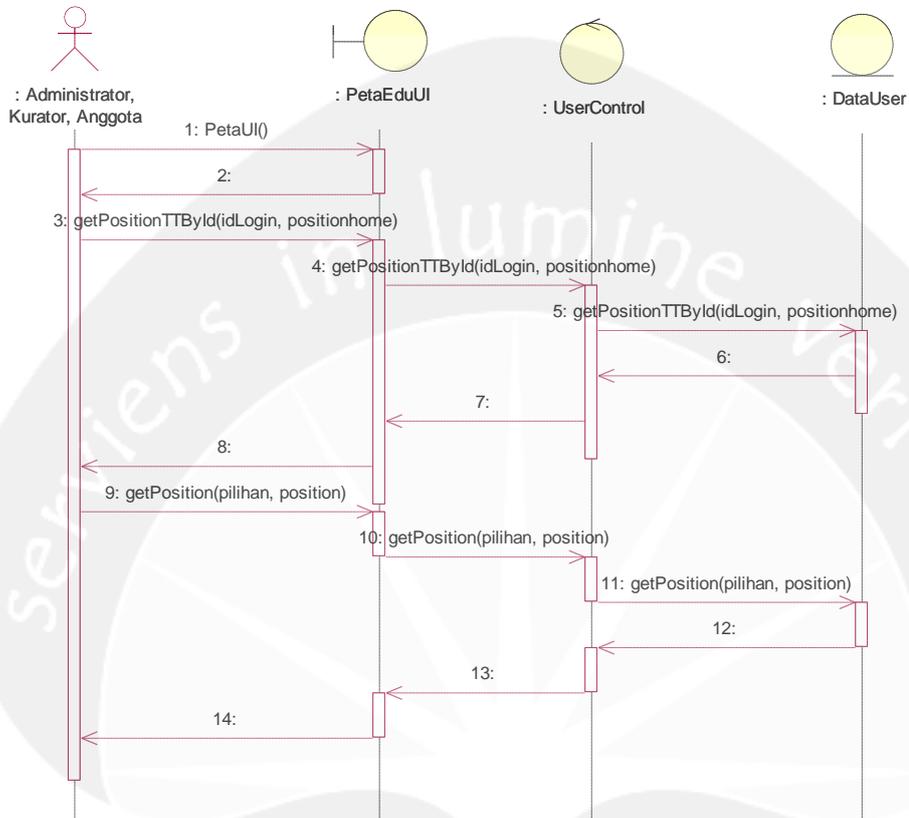
Gambar 2.16 Sequence Diagram: Hapus Edu

### 2.2.1.6 Hapus Anggota



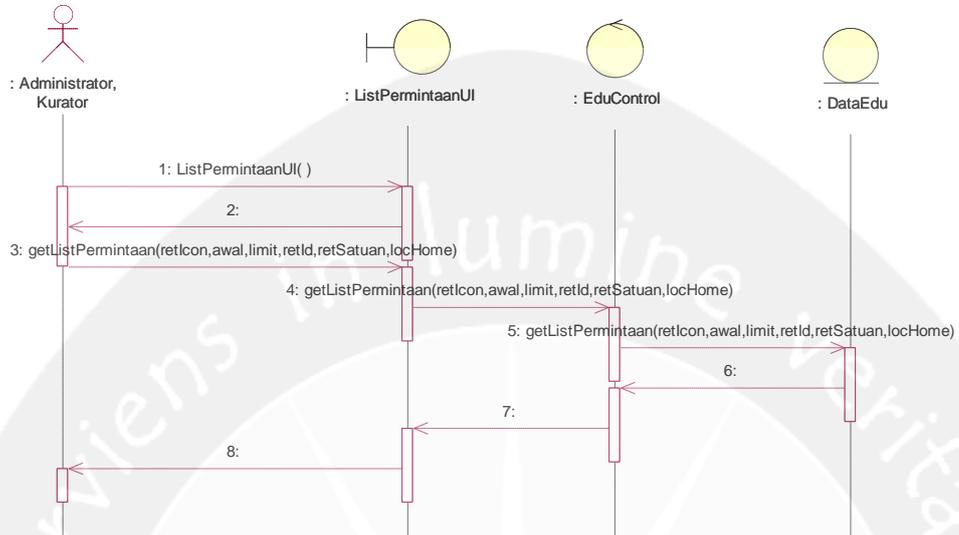
Gambar 2.17 Sequence Diagram : Hapus Anggota

### 2.2.1.7 Lihat Peta



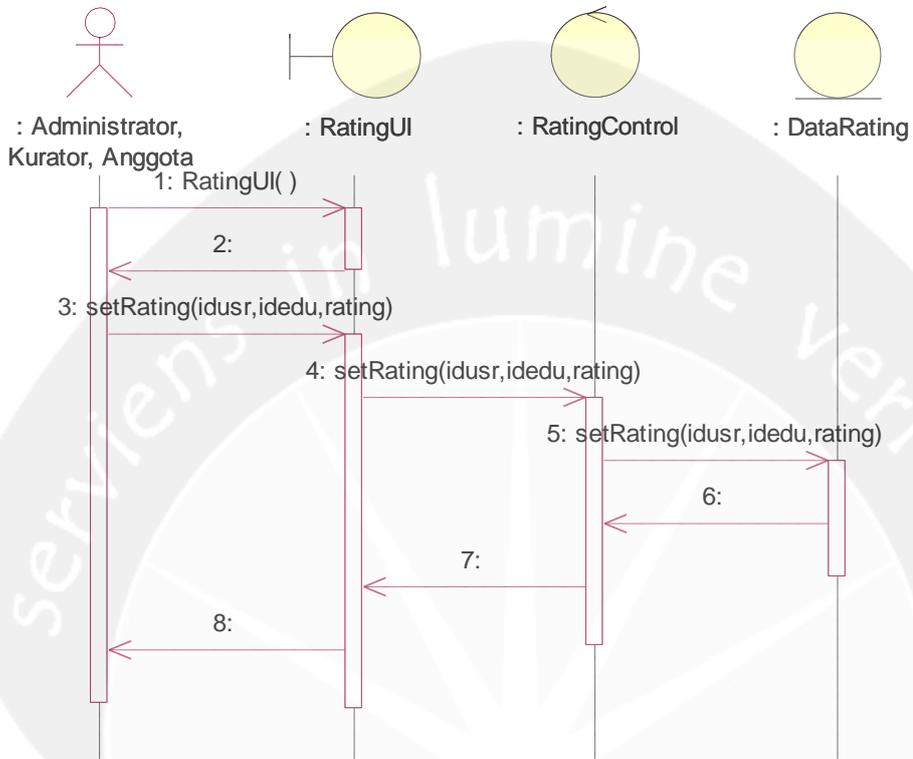
Gambar 2.18 Sequence Diagram: Lihat Peta

### 2.2.1.8 Lihat List Permintaan



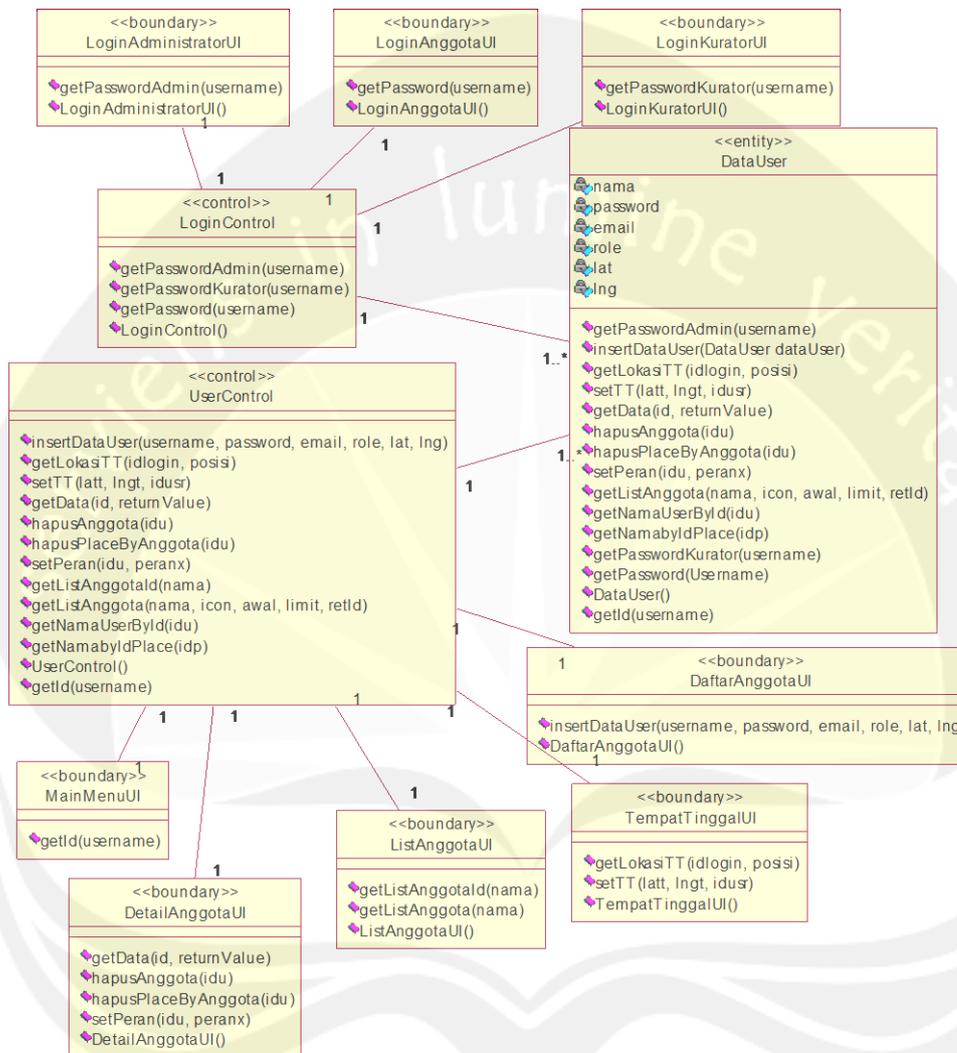
Gambar 2.18 Sequence Diagram: Lihat List Permintaan

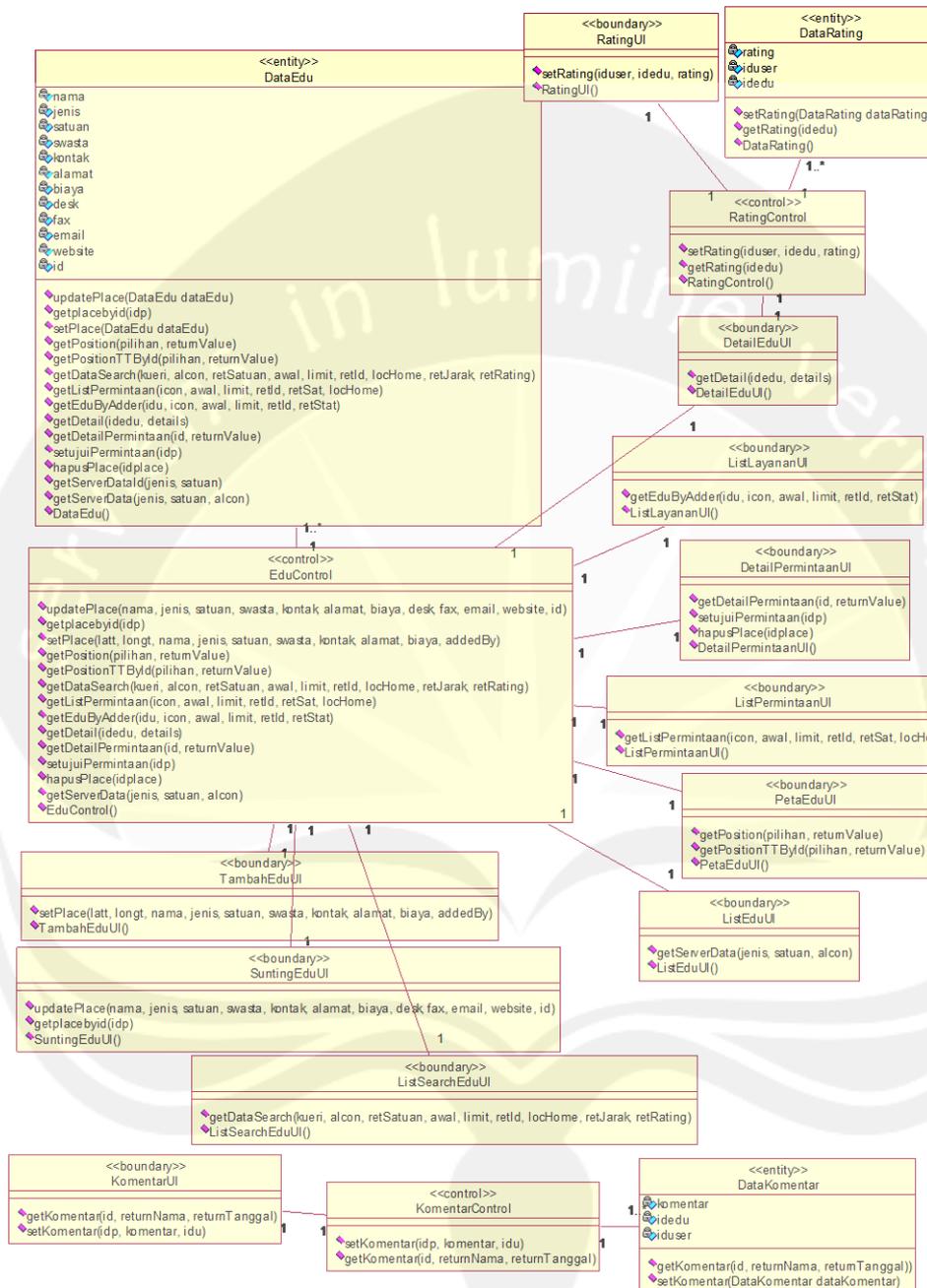
### 2.2.1.9 Beri rating



Gambar 2.18 Sequence Diagram: Beri rating

## 2.3 Class Diagram





Gambar 3 Class Diagram

## 2.4 Class Diagram Specific Descriptions

### Specific Design Class DaftarAnggotaUI

<b>DaftarAnggotaUI</b>	<b>&lt;&lt;boundary&gt;&gt;</b>
<pre>+DaftarAanggota() Default konstruktor, digunakan untuk inisialisasi semua attribute dari kelas ini.  + insertDataUser(String username, String password, String email, String role,String latt, String lngt) Operasi ini digunakan untuk mendaftarkan pengguna baru.</pre>	

### Specific Design Class LoginUI

<b>LoginUI</b>	<b>&lt;&lt;boundary&gt;&gt;</b>
<pre>+LoginUI() Default konstruktor, digunakan untuk inisialisasi semua attribute dari kelas ini.  + getPassword(String username):String Mendapatkan password dari nama pengguna.</pre>	

### Specific Design Class LoginAdminUI

<b>LoginAdminUI</b>	<b>&lt;&lt;boundary&gt;&gt;</b>
<pre>+LoginAdminUI() Default konstruktor, digunakan untuk inisialisasi semua attribute dari kelas ini.  + getPasswordAdmin(String username):String Mendapatkan password dari nama pengguna.</pre>	

### Specific Design Class LoginKuratorfUI

<b>LoginKuratorUI</b>	<b>&lt;&lt;boundary&gt;&gt;</b>
+LoginKuratorUI() Default konstruktor, digunakan untuk inisialisasi semua attribute dari kelas ini.	
+ getPasswordKurator(String username):String Mendapatkan password dari nama pengguna.	

### Specific Design Class MainMenuUI

<b>MainMenuUI</b>	<b>&lt;&lt;boundary&gt;&gt;</b>
+MainMenuUI() Default konstruktor, digunakan untuk inisialisasi semua attribute dari kelas ini.	
+getIddanStatus(String username):String Operasi ini digunakan untuk mengambil id berdasarkan nama pengguna	

### Specific Design Class KatalogUI

<b>KatalogUI</b>	<b>&lt;&lt;boundary&gt;&gt;</b>
+KatalogUI() Default konstruktor, digunakan untuk inisialisasi semua attribute dari kelas ini.	

### Specific Design Class ListEduUI

<b>ListEduUI</b>	<b>&lt;&lt;boundary&gt;&gt;</b>
+ListEduUI() Default konstruktor, digunakan untuk inisialisasi semua attribute dari kelas ini.	
+ getServerData(String jenis, String satuan, int awal, int	

```

limit,      ArrayList<Integer>      aIcon,      ArrayList<String>
returnSatuan,      ArrayList      retId,      Location      locHome,
ArrayList<Float>      aJarak,      ArrayList<String>
returnRating):ArrayList

```

Operasi ini digunakan untuk mengisi List berisi nama unit pendidikan beserta icon yang bersesuaian dengan satuan pendidikan.

### Specific Design Class ListlayananUI

<b>ListLayananUI</b>	<b>&lt;&lt;boundary&gt;&gt;</b>
<pre> +ListLayananUI() Default konstruktor, digunakan untuk inisialisasi semua attribute dari kelas ini. + getEduByAdder(String idu,ArrayList&lt;Integer&gt; aIcon,int awal,int limit,ArrayList retId,ArrayList retStat):ArrayList Operasi ini digunakan untuk mengambil nama unit pendidikan berdasarkan id pengguna, id Edu, icon, status dan batasan kembalian dari database. +hapusPlace(String idedu) Operasi ini digunakan untuk menghapus unit pendidikan berdasarkan id unit pendidikan </pre>	

### Specific Design Class ListAnggotaUI

<b>ListAnggotaUI</b>	<b>&lt;&lt;boundary&gt;&gt;</b>
<pre> +ListAnggotaUI() Default konstruktor, digunakan untuk inisialisasi semua attribute dari kelas ini. + getListAnggota(String nama,ArrayList&lt;Integer&gt; aIcon,int awal,int limit, ArrayList retId):ArrayList Operasi ini digunakan untuk mengambil nama pengguna (Anggota dan Kurator) dengan icon dan batasan kembalian dari </pre>	

database.

+hapusAnggota(String iduser)

Operasi ini digunakan untuk menghapus pengguna berdasarkan id pengguna

+hapusPlaceByAnggota(String iduser)

Operasi ini digunakan untuk menghapus semua unit pendidikan yang terasosiasi berdasarkan id pengguna

### Specific Design Class ListPermintaanUI

<b>ListPermintaanUI</b>	<b>&lt;&lt;boundary&gt;&gt;</b>
<p>+ListPermintaanUI()</p> <p>Default konstruktor, digunakan untuk inisialisasi semua attribute dari kelas ini.</p> <p>+getListPermintaan(ArrayList&lt;Integer&gt; aIcon, int awal, int limit, ArrayList retId, ArrayList retSat, Location locHome): ArrayList</p> <p>Operasi ini digunakan untuk mengambil nama permintaan penempatan unit pendidikan.</p> <p>+hapusPlace(String idedu)</p> <p>Operasi ini digunakan untuk menghapus unit pendidikan berdasarkan id unit pendidikan</p>	

### Specific Design Class DetailEduUI

<b>DetailEduUI</b>	<b>&lt;&lt;boundary&gt;&gt;</b>
<p>+DetailEduUI()</p> <p>Default konstruktor, digunakan untuk inisialisasi semua attribute dari kelas ini.</p> <p>+getDetail(String idEdu,ArrayList details): String</p> <p>Operasi ini digunakan untuk mengisi List berisi detail dari unit pendidikan berdasarkan id dari unit pendidikan.</p>	

### Specific Design Class DetailAnggotaUI

<b>DetailAnggotaUI</b>	<b>&lt;&lt;boundary&gt;&gt;</b>
<pre>+DetailAnggotaUI() Default konstruktor, digunakan untuk inisialisasi semua attribute dari kelas ini. +getData(String iduser,ArrayList details):String Operasi ini digunakan untuk mengisi List berisi detail dari pengguna berdasarkan id dari pengguna. +hapusAnggota(String iduser) Operasi ini digunakan untuk menghapus pengguna berdasarkan id pengguna. +hapusPlaceByAnggota(String iduser) Operasi ini digunakan untuk menghapus semua unit pendidikan yang terasosiasi berdasarkan id pengguna.</pre>	

### Specific Design Class DetailPermintaanUI

<b>DetailPermintaanUI</b>	<b>&lt;&lt;boundary&gt;&gt;</b>
<pre>+DetailPermintaanUI() Default konstruktor, digunakan untuk inisialisasi semua attribute dari kelas ini. +getDetailPermintaan(String idEdu,ArrayList details):String Operasi ini digunakan untuk mengisi List berisi detail dari permintaan penempatan unit pendidikan berdasarkan id dari unit pendidikan. +hapusPlace(String idedu) Operasi ini digunakan untuk menghapus unit pendidikan berdasarkan id unit pendidikan</pre>	

### Specific Design Class KomentarUI

<b>KomentarUI</b>	<b>&lt;&lt;boundary&gt;&gt;</b>
<pre>+KomentarUI() Default konstruktor, digunakan untuk inisialisasi semua attribute dari kelas ini. + setKomentar(String idedu,String komentar,string iduser) Operasi ini digunakan untuk mengirim komentar untuk unit pendidikan berdasarkan id unit pendidikan, isi komentar, dan id pengguna. + getKomentar(String idedu, ArrayList returnNama, ArrayList returnTanggal) :ArrayList Operasi ini digunakan untuk mengambil isi komentar, mengisi List dengan nama komentator,mengisi List dengan tanggal komentar berdasarkan id unit pendidikan.</pre>	

### Specific Design Class ListSearchEduUI

<b>ListSearchEduUI</b>	<b>&lt;&lt;boundary&gt;&gt;</b>
<pre>+ListSearchEdu() Default konstruktor, digunakan untuk inisialisasi semua attribute dari kelas ini. + getDataSearch(kueri, aIcon, retSatuan, awal, limit, retId, locHome, retJarak, retRating):ArrayList Operasi ini digunakan untuk mengisi List berisi nama unit pendidikan beserta icon yang bersesuaian dengan satuan pendidikan berdasarkan string kueri, jarak dari rumah dan nilai.</pre>	

### Specific Design Class PetaEduUI

<b>PetaEduUI</b>	<b>&lt;&lt;boundary&gt;&gt;</b>
<pre>+PetaEduUI() Default konstruktor, digunakan untuk inisialisasi semua attribute dari kelas ini. + getPositionTTById(String iduser,ArrayList positionhome): String Operasi ini digunakan untuk mengambil posisi dari tempat tinggal berdasarkan id pengguna. + getPosition(String idedu,ArrayList position):String Operasi ini digunakan untuk mengambil posisi dari unit pendidikan berdasarkan id unit pendidikan.</pre>	

### Specific Design Class SearchUI

<b>SearchUI</b>	<b>&lt;&lt;boundary&gt;&gt;</b>
<pre>+SearchUI() Default konstruktor, digunakan untuk inisialisasi semua attribute dari kelas ini. + buatQuery() Operasi ini digunakan untuk membangun string kueri untuk pencarian.</pre>	

### Specific Design Class RatingUI

<b>RatingUI</b>	<b>&lt;&lt;boundary&gt;&gt;</b>
<pre>+RatingUI() Default konstruktor, digunakan untuk inisialisasi semua attribute dari kelas ini. + setRating(String iduser,String idedu, Float Rating) Operasi ini digunakan untuk mengirimkan rating berdasarkan id user, id edu, dan nilainya.</pre>	

### Specific Design Class TambahEduUI

<b>TambahEduUI</b>	<b>&lt;&lt;boundary&gt;&gt;</b>
<pre>+KelolaKaryawan() Default konstruktor, digunakan untuk inisialisasi semua attribute dari kelas ini. + setPlace(String latt, String longt, String nama, String jenis, String satuan, String swasta, String kontak, String alamat, String biaya, String addedBy) Operasi ini digunakan untuk menambahkan detail dari unit pendidikan yang akan ditambahkan.</pre>	

### Specific Design Class SuntingEduUI

<b>SuntingEduUI</b>	<b>&lt;&lt;boundary&gt;&gt;</b>
<pre>+SuntingEdu() Default konstruktor, digunakan untuk inisialisasi semua attribute dari kelas ini. + updatePlace(String nama, String jenis, String satuan, String swasta, String kontak, String alamat, String biaya, String desk, String fax, String email, String website, String id); Operasi ini digunakan untuk menyimpan detail dari unit pendidikan berdasarkan id unit pendidikan. + getplacebyid(idp) :String Operasi ini digunakan untuk mengambil detail unit pendidikan berdasarkan id unit pendidikan.</pre>	

### Specific Design Class EduControl

EduControl	<<control>>
<pre> +EduControl() Default konstruktor, digunakan untuk inisialisasi semua attribute dari kelas ini. + updatePlace(String nama,String jenis,String satuan,String swasta,String kontak,String alamat,String biaya,String desk,String fax,String email,String website,String id) Operasi ini digunakan untuk menyimpan detail dari unit pendidikan berdasarkan id unit pendidikan. + getplacebyid(String idp) Operasi ini digunakan untuk mengambil detail unit pendidikan + setPlace(String latt,String longt,String nama,String jenis,String satuan,String swasta,String kontak,String alamat,String biaya,String addedBy) Operasi ini digunakan untuk menambahkan detail dari unit pendidikan yang akan ditambahkan. + getPosition(String pilihan, ArrayList&lt;Double&gt; returnValue) Operasi ini digunakan untuk mengambil posisi dari unit pendidikan berdasarkan id unit pendidikan. + getPositionTTById(String pilihan, ArrayList&lt;Double&gt; returnValue):String Operasi ini digunakan untuk mengambil posisi dari tempat tinggal berdasarkan id pengguna. + getDataSearch(String kueri, ArrayList&lt;Integer&gt; aIcon, ArrayList&lt;String&gt; returnSatuan, int awal, int limit, ArrayList retId, Location locHome, ArrayList&lt;Float&gt; aJarak,ArrayList&lt;String&gt; returnRating):ArrayList Operasi ini digunakan untuk mengisi List berisi nama unit pendidikan beserta icon yang bersesuaian dengan satuan pendidikan, limit, mengisi id, jarak dari rumah dan satuan pendidikan berdasarkan string kueri. </pre>	

```
+ getListPermintaan(ArrayList<Integer> aIcon, int awal, int limit, ArrayList retId, ArrayList retSat, Location locHome):ArrayList
```

Operasi ini digunakan untuk mengambil nama permintaan penempatan unit pendidikan, id, satuan, icon dan jarak dari rumah beserta batasan jumlah kembalian.

```
+ getEduByAdder(String idu, ArrayList<Integer> aIcon, int awal, int limit, ArrayList retId, ArrayList retStat):ArrayList
```

Operasi ini digunakan untuk mengambil nama unit pendidikan berdasarkan id pengguna, id Edu, icon, status dan batasan kembalian dari database.

```
+ getDetail(String idedu, ArrayList<String> details):String
```

Operasi ini digunakan untuk mengisi List berisi detail dari unit pendidikan berdasarkan id dari unit pendidikan.

```
+ getDetailPermintaan( String id, ArrayList<String> returnValue):String
```

Operasi ini digunakan untuk mengisi List berisi detail dari permintaan penempatan unit pendidikan berdasarkan id dari unit pendidikan.

```
+ setujuPermintaan(String idp)
```

Operasi ini digunakan untuk menyetujui permintaan penempatan Edu baru.

```
+ hapusPlace(String idplace)
```

Operasi ini digunakan untuk menghapus unit pendidikan berdasarkan id unit pendidikan

```
+ getServerData(String jenis, String satuan, int awal, int limit, ArrayList<Integer> aIcon, ArrayList<String> returnSatuan, ArrayList retId, Location locHome, ArrayList<Float> aJarak, ArrayList<String> returnRating):ArrayList
```

Operasi ini digunakan untuk mengisi List berisi nama unit pendidikan beserta icon yang bersesuaian dengan satuan pendidikan, jarak dari rumah, nilai, dan batasan kembalian dari database.

### Specific Design Class UserControl

UserControl	<<Control>>
<pre> +UserController() Default konstruktor, digunakan untuk inisialisasi semua attribute dari kelas ini. + getPassword(String username) Operasi ini digunakan untuk mendapatkan password berdasarkan nama pengguna. + getIdanStatus(String username, ArrayList status, ArrayList&lt;Double&gt;ahome) Operasi ini digunakan untuk mendapatkan id pengguna, status pengguna, dan lokasi rumah pengguna berdasarkan nama pengguna. + insertDataUser(String username, String password, String email, String role, String latt, String lngt) Operasi ini digunakan untuk mendaftarkan pengguna baru. + cekUsername(String username) Operasi ini digunakan untuk mengecek apakah nama yang akan didaftarkan sudah ada dalam table di database. + cekEmail(String email) Operasi ini digunakan untuk mengecek apakah format email sudah benar. + getNamaByIdp(String idp) Operasi ini digunakan untuk mendapatkan nama pengguna yang mendaftarkan sebuah unit pendidikan berdasarkan id unit pendidikan. + getNamaUserById(String idu) Operasi ini digunakan untuk mendapatkan nama pengguna berdasarkan id pengguna. + getLocationTT(String idlogin, ArrayList&lt;Double&gt; posisi) Operasi yang digunakan untuk mendapatkan posisi tempat tinggal dari pengguna berdasarkan id pengguna. </pre>	

```

+ setTT(String latt, String lngt, String idusr)
Operasi yang digunakan untuk menentukan posisi dari tempat tinggal pengguna.
+ getData(String id, ArrayList<String> returnValue)
Operasi yang digunakan untuk mendapatkan detail pengguna berdasarkan id pengguna.
+ hapusAnggota(String idu)
Operasi yang digunakan untuk menghapus pengguna berdasarkan id pengguna.
+ hapusPlaceByAnggota(String idu)
Operasi yang digunakan untuk menghapus semua unit pendidikan yang didaftarkan oleh pengguna berdasarkan id pengguna.
+ setPeran(String idu,String peranx)
Operasi yang digunakan untuk menentukan peran dari seorang pengguna.
+ getListAnggota(String nama)
Operasi yang digunakan untuk mendapatkan nama dari pengguna.

```

**Specific Design Class LoginControl**

<b>LoginControl</b>	<b>&lt;&lt;control&gt;&gt;</b>
<pre> +LoginControl() Default konstruktor, digunakan untuk inisialisasi semua attribute dari kelas ini. + getPassword(String username) Operasi yang digunakan untuk mendapatkan password dari anggota. + getPasswordAdmin(String username) Operasi yang digunakan untuk mendapatkan password dari administrator. + getPasswordKurator(String username) Operasi yang digunakan untuk mendapatkan password dari kurator. + cekEmail(String email) </pre>	

Operasi yang digunakan untuk mengecek apakah format penulisan email sudah benar.

### Specific Design Class KomentarControl

<b>KomentarControl</b>	<b>&lt;&lt;control&gt;&gt;</b>
<pre>+KomentarControl() Default konstruktor, digunakan untuk inisialisasi semua attribute dari kelas ini. + getKomentar(String id,ArrayList returnNama,ArrayList returnTanggal) Operasi ini digunakan untuk mendapatkan komentar berdasarkan id unit pendidikan + setKomentar(String idp,String komentar,String idu) Operasi yang digunakan untuk mengirimkan komentar ke unit pendidikan.</pre>	

### Specific Design Class RatingControl

<b>RatingControl</b>	<b>&lt;&lt;control&gt;&gt;</b>
<pre>+RatingControl() Default konstruktor, digunakan untuk inisialisasi semua attribute dari kelas ini. + setRating(String iduser,String idedu, Float Rating) Operasi ini digunakan untuk mengirimkan rating berdasarkan id user, id edu, dan nilainya. + getRating(String idedu) Operasi ini digunakan untuk mendapatkan rating dari sebuah unit pendidikan berdasarkan id unit pendidikan tersebut. +getRatingUser(String idedu,String iduser) Operasi ini digunakan untuk mendapatkan nilai/rating dari sebuah unit pendidikan oleh seorang pengguna. +setSummaryRating(String idedu, float rating) Operasi ini digunakan untuk menentukan nilai dari sebuah</pre>	

unit pendidikan sebagai nilai rata-rata yang diletakkan pada kolom rating pada table place agar dapat dengan cepat diperoleh.

### Specific Design Class DataEdu

DataEdu	<<entity>>
<p>-_nama: String Atribut ini untuk menyimpan data nama Edu.</p> <p>-_jenis: String Atribut ini untuk menyimpan data jenis Edu.</p> <p>-_satuan: String Atribut ini untuk menyimpan data satuan Edu.</p> <p>-_swasta: String Atribut ini untuk menyimpan data status swasta dari Edu.</p> <p>-_alamat: String Atribut ini untuk menyimpan data alamat Edu.</p> <p>-_biaya: String Atribut ini untuk menyimpan data biaya Edu.</p> <p>-_telp: String Atribut ini untuk menyimpan data telp Edu.</p> <p>-_fax: String Atribut ini untuk menyimpan data fax Edu.</p> <p>-_email: String Atribut ini untuk menyimpan data email Edu.</p> <p>-_web: String Atribut ini untuk menyimpan data website Edu</p> <p>-_id: String Atribut ini untuk menyimpan data id Edu</p> <p>-_rating: String Atribut ini untuk menyimpan data rating Edu.</p> <p>-_desk: String Atribut ini untuk menyimpan data deskripsi Edu.</p> <p>-_latt: String Atribut ini untuk menyimpan data koordinat latitude dari</p>	

tempat tinggal pengguna.

-\_lngt: String

Atribut ini untuk menyimpan data koordinat longitude dari tempat tinggal pengguna.

-\_addedby: String

Atribut ini untuk menyimpan data id user yang menambahkan Edu.

+DataEdu()

Default konstruktor, digunakan untuk inisialisasi semua attribute dari kelas ini.

+ updatePlace(DataEdu dataEdu)

Operasi ini digunakan untuk menyimpan detail dari unit pendidikan berdasarkan id unit pendidikan.

+ getplacebyid(String idp)

Operasi ini digunakan untuk mengambil detail unit pendidikan

+ setPlace(DataEdu dataEdu)

Operasi ini digunakan untuk menambahkan detail dari unit pendidikan yang akan ditambahkan.

+ getPosition(String pilihan, ArrayList<Double> returnValue)

Operasi ini digunakan untuk mengambil posisi dari unit pendidikan berdasarkan id unit pendidikan.

+ getPositionTTById(String pilihan, ArrayList<Double> returnValue)

Operasi ini digunakan untuk mengambil posisi dari tempat tinggal berdasarkan id pengguna.

+ getDataSearch(String nama, String jenis, String satuan, String swasta, String desk, String kuerix, ArrayList<Integer> aIcon, ArrayList<String> returnSatuan, int awal, int limit, ArrayList retId, Location locHome, ArrayList<Float> aJarak, ArrayList<String> returnRating)

Operasi ini digunakan untuk mengisi List berisi nama unit pendidikan beserta icon yang bersesuaian dengan satuan

pendidikan, jarak dari rumah, dan nilai berdasarkan string kueri serta dibatasi dengan nilai limit.

```
+ getListPermintaan(ArrayList<Integer> aIcon, int awal, int limit, ArrayList retId, ArrayList retSat, Location locHome)
```

Operasi ini digunakan untuk mengambil nama permintaan penempatan unit pendidikan dengan id, satuan pendidikan, dan jarak dari rumah.

```
+ getEduByAdder(String idu, ArrayList<Integer> aIcon, int awal, int limit, ArrayList retId, ArrayList retStat)
```

Operasi ini digunakan untuk mengambil nama unit pendidikan berdasarkan id pengguna beserta id, status, icon dan dibatasi oleh nilai limit.

```
+ getDetail(String idedu, ArrayList<String> details)
```

Operasi ini digunakan untuk mengisi List berisi detail dari unit pendidikan berdasarkan id dari unit pendidikan.

```
+ getDetailPermintaan(String id, ArrayList<String> returnValue)
```

Operasi ini digunakan untuk mengisi List berisi detail dari permintaan penempatan unit pendidikan berdasarkan id dari unit pendidikan.

```
+ setujuPermintaan(String idp)
```

Operasi ini digunakan untuk menyetujui permintaan penempatan Edu.

```
+ hapusPlace(String idplace)
```

Operasi ini digunakan untuk menghapus unit pendidikan berdasarkan id unit pendidikan

```
+ getServerData(String jenis, String satuan, int awal, int limit, ArrayList<Integer> aIcon, ArrayList<String> returnSatuan, ArrayList retId, Location locHome, ArrayList<Float> aJarak, ArrayList<String> returnRating)
```

Operasi ini digunakan untuk mengisi List berisi nama unit pendidikan beserta icon yang bersesuaian dengan satuan pendidikan beserta jarak dari rumah dan nilai/rating.

### Specific Design Class DataUser

DataUser	<<entity>>
<p>-_idUser:String Atribut ini untuk menyimpan data id user</p> <p>-_username:String Atribut ini untuk menyimpan data nama user</p> <p>-_password:String Atribut ini untuk menyimpan data password user</p> <p>-_roleUser:String Atribut ini untuk menyimpan data peran user</p> <p>-_latt:String Atribut ini untuk menyimpan data koordinat latitude rumah/tempat tinggal dari user</p> <p>-_lngt:String Atribut ini untuk menyimpan data koordinat longitudo rumah/tempat tinggal dari user</p>	
<p>+DataUser() Default konstruktor, digunakan untuk inisialisasi semua attribute dari kelas ini.</p> <p>+ getPassword (String username):String Operasi ini digunakan untuk mendapatkan password berdasarkan nama pengguna.</p> <p>+ getPasswordKurator(String username) :String Operasi ini digunakan untuk mendapatkan password kurator berdasarkan nama pengguna.</p> <p>+ getPasswordAdministrator(String username) :String Operasi ini digunakan untuk mendapatkan password administrator berdasarkan nama pengguna.</p> <p>+ getIdanStatus(String username) : String Operasi ini digunakan untuk mendapatkan id pengguna, status, dan lokasi rumah/tempat tinggal berdasarkan nama pengguna.</p> <p>+ insertDataUser(DataUser dataUser) Operasi ini digunakan untuk mendaftarkan pengguna baru.</p>	

```

+ cekNama(String username) : String
Operasi ini digunakan untuk mengecek apakah nama yang akan
didaftarkan sudah ada dalam table di database.
+ getNamaByIdp(String idp) : String
Operasi ini digunakan untuk mendapatkan nama pengguna yang
mendaftarkan sebuah unit pendidikan berdasarkan id unit
pendidikan.
+ getNamaUserById(String idu): String
Operasi ini digunakan untuk mendapatkan nama pengguna
berdasarkan id pengguna.
+ getLocationTT(String idlogin, ArrayList<Double> posisi)
:String
Operasi yang digunakan untuk mendapatkan posisi tempat
tinggal dari pengguna berdasarkan id pengguna.
+ setTT(String latt, String lngt, String idusr)
Operasi yang digunakan untuk menentukan posisi dari tempat
tinggal pengguna.
+ getData(String id, ArrayList<String> returnValue) :String
Operasi yang digunakan untuk mendapatkan detail pengguna
berdasarkan id pengguna.
+ hapusAnggota(String idu)
Operasi yang digunakan untuk menghapus pengguna berdasarkan
id pengguna.
+ hapusPlaceByAnggota(String idu)
Operasi yang digunakan untuk menghapus semua unit pendidikan
yang didaftarkan oleh pengguna berdasarkan id pengguna.
+ setPeran(String idu,String peranx)
Operasi yang digunakan untuk menentukan peran dari seorang
pengguna.
+ getListAnggota(String nama,ArrayList<Integer> aIcon,int
awal,int limit, ArrayList retId):ArrayList
Operasi yang digunakan untuk mendapatkan nama dari pengguna
beserta icon dan jumlah kembalian dengan nilai limit.

```

### Specific Design DataKomentar

DataKomentar	<<entity>>
<pre>-_komentar:String Atribut ini untuk menyimpan data isi komentar -_idUser:String Atribut ini untuk menyimpan data id user -_idedu:String Atribut ini untuk menyimpan data id Edu</pre>	
<pre>+DataKomentar() Default konstruktor, digunakan untuk inisialisasi semua attribute dari kelas ini. + getKomentar(String id,ArrayList returnNama,ArrayList returnTanggal) :ArrayList Operasi ini digunakan untuk mendapatkan komentar berdasarkan id unit pendidikan + setKomentar(DataKomentar dataKomentar) Operasi yang digunakan untuk mengirimkan komentar ke unit pendidikan.</pre>	

### Specific Design DataRating

DataRating	<<entity>>
<pre>-_rating:String Atribut ini untuk menyimpan data isi komentar -_idUser:String Atribut ini untuk menyimpan data id user -_idedu:String Atribut ini untuk menyimpan data id Edu</pre>	
<pre>+DataRating() Default konstruktor, digunakan untuk inisialisasi semua attribute dari kelas ini. + setRating(DataRating dataRating):String</pre>	

Operasi ini digunakan untuk mengirimkan rating berdasarkan id user, id edu, dan nilainya.

```
+ getRating(String idedu)
```

Operasi ini digunakan untuk mendapatkan rating dari sebuah unit pendidikan berdasarkan id unit pendidikan tersebut.

```
+getRatingUser(String idedu,String iduser):float
```

Operasi ini digunakan untuk mendapatkan rating dari seorang user dari sebuah unit pendidikan.

### 3 Perancangan Data

#### 3.1 Dekomposisi Data

##### 3.1.1 Deskripsi Entitas Data Pengguna

Nama	Tipe	Panjang	Keterangan
ID_PENGGUNA	integer	11	Id pengguna, <b>Primary key</b>
PERAN	varchar	15	Peran pengguna
EMAIL	varchar	50	Alamat email pengguna
LATITUDE	double	50	Latitude dari tempat tinggal pengguna
LONGITUDE	double	50	Latitude dari tempat tinggal pengguna
USERNAME	varchar	25	Username pengguna
PASSWORD	varchar	25	Password pengguna
ISAKTIF	integer	11	bernilai 1 jika telah diaktifkan dan 0 jika belum

##### 3.1.2 Deskripsi Entitas Data Edu

Nama	Tipe	Panjang	Keterangan
ID_EDU	integer	11	Id unit pendidikan, <b>Primary key</b>
NAMA_EDU	varchar	50	Nama unit pendidikan
ALAMAT_EDU	Varchar	50	Alamat unit pendidikan

LONGITUDE_EDU	Double	10	Longitude dari unit pendidikan
LATITUDE_EDU	Double	10	latitude dari unit pendidikan
TELP_EDU	Varchar	50	Telepon unit pendidikan
JENIS_PENDIDIKAN	Varchar	25	Jenis pendidikan dari unit pendidikan
SATUAN_PENDIDIKAN	Varchar	15	Satuan pendidikan dari unit pendidikan
SWASTA	Varchar	10	Status swasta atau negeri dari unit pendidikan
SPP	Int	11	Kisaran spp dari unit pendidikan
RATING	Double		Nilai rata-rata dari unit pendidikan
<b>ID_PENGGUNA</b>	<b>Integer</b>	<b>11</b>	<b>Id pengguna, Foreign Key</b>
TANGGAL_BUAT	Date		Tanggal unit pendidikan didaftarkan
STATUS	integer	11	Bernilai 1 jika unit pendidikan sudah disetujui, dan 0 jika belum
DESKRIPSI	Text		Deskripsi dari unit pendidikan
FAX	Varchar	50	Fax dari unit pendidikan
EMAIL	Varchar	50	Alamat email unit pendidikan
WEBSITE	varchar	50	Website unit pendidikan

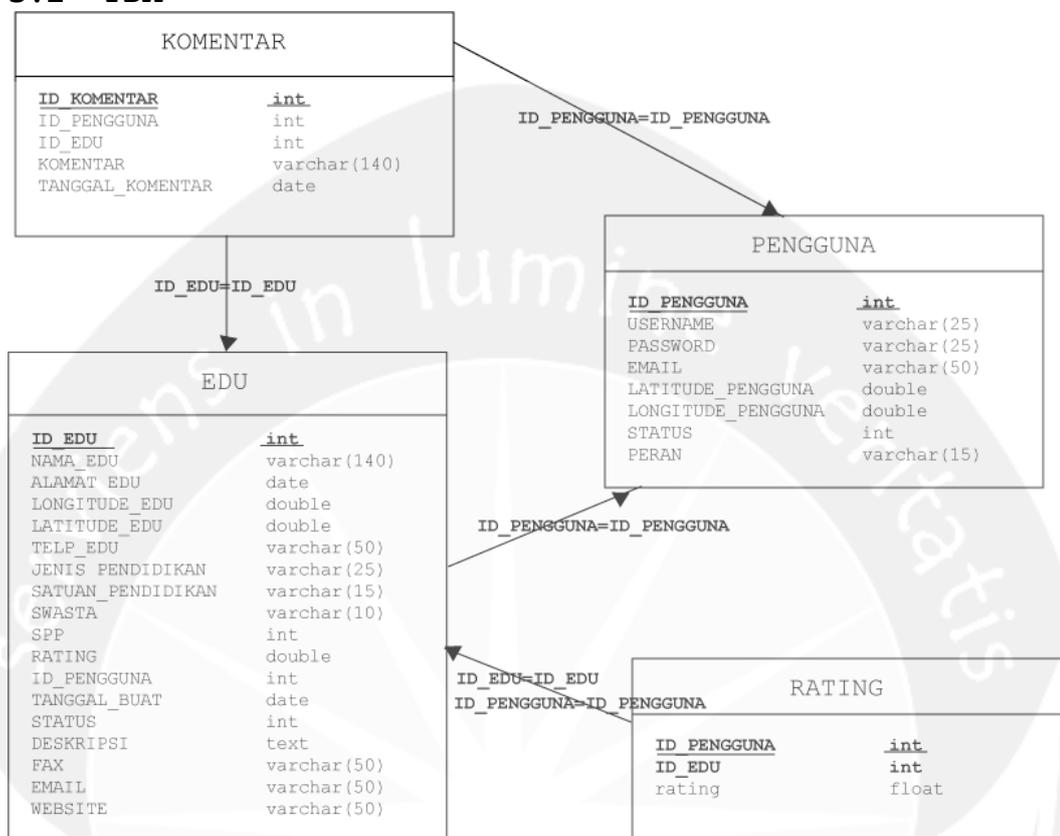
### 3.1.3 Deskripsi Entitas Data Komentar

Nama	Tipe	Panjang	Keterangan
ID_KOMENTAR	Integer	11	ID Transaksi, Primary key
ID_PENGGUNA	Integer	11	Id pengguna, Foreign Key
ID_EDU	Integer	11	Id unit pendidikan, Foreign Key
KOMENTAR	Varchar	140	Isi komentar
TANGGAL_KOMENTAR	datetime	-	Tanggal komentar dibuat

### 3.1.4 Deskripsi Entitas Data Rating

Nama	Tipe	Panjang	Keterangan
ID_PENGGUNA	Integer	11	ID pengguna, Primary key
ID_EDU	Integer	11	Id Edu, Primary Key
RATING	float		Nilai rating

### 3.2 PDM



Gambar 3.1 Gambar PDM

## 4 Deskripsi Perancangan Antarmuka

### 4.1 Log in Anggota



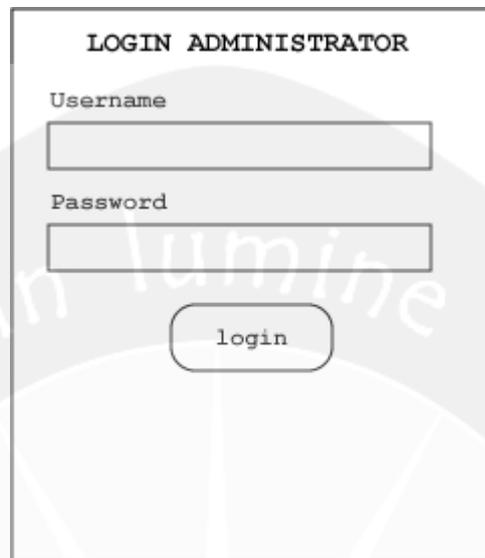
The image shows a login form with the following elements:

- Title: LOGIN
- Username label and text input field
- Password label and text input field
- login button
- daftar button

**Gambar 4.1 Rancangan Antarmuka Login Anggota**

Antarmuka pada gambar 4.1 ini digunakan untuk melakukan proses login ke dalam sistem CariEdu. Anggota harus memasukkan username dan password dengan benar pada textbox yang telah disediakan. Pada saat tombol login ditekan, sistem akan mengecek username dan password yang dimasukkan dengan data username dan password yang telah tersimpan di database. Jika data username dan password cocok maka pengguna akan masuk ke dalam sistem, sebaliknya akan diberikan pesan peringatan jika tidak cocok.

## 4.2 Log in Administrator

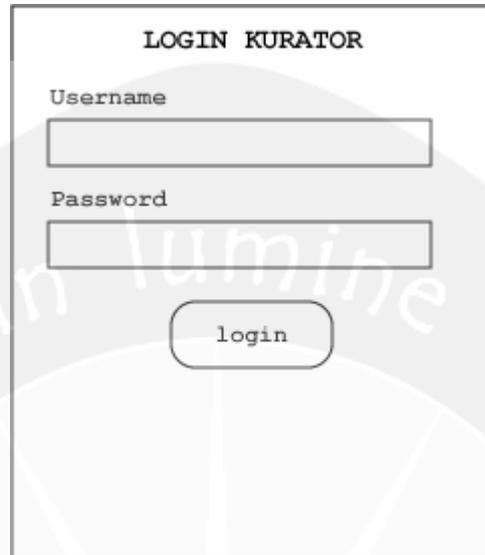


The image shows a login form for an administrator. The form is enclosed in a rectangular border and has the title "LOGIN ADMINISTRATOR" at the top center. Below the title, there are two text input fields. The first field is labeled "Username" and the second is labeled "Password". Below these fields is a rounded rectangular button with the text "login" inside it.

**Gambar 4.2 Rancangan Antarmuka Login Administrator**

Antarmuka pada gambar 4.2 ini digunakan untuk melakukan proses login ke dalam sistem CariEdu. Administrator harus memasukkan username dan password dengan benar pada textbox yang telah disediakan. Pada saat tombol login ditekan, sistem akan mengecek username dan password yang dimasukkan dengan data username dan password yang telah tersimpan di database. Jika data username dan password cocok maka pengguna akan masuk ke dalam sistem, sebaliknya akan diberikan pesan peringatan jika tidak cocok.

### 4.3 Log in Kurator



The image shows a login form titled "LOGIN KURATOR". It contains two text input fields: "Username" and "Password". Below these fields is a rounded rectangular button labeled "login". The form is set against a light gray background with a faint watermark of a university logo and the motto "serviens in lumine veritatis".

**Gambar 4.3 Rancangan Antarmuka Login Kurator**

Antarmuka pada gambar 4.3 ini digunakan untuk melakukan proses login ke dalam sistem CariEdu. Kurator harus memasukkan username dan password dengan benar pada textbox yang telah disediakan. Pada saat tombol login ditekan, sistem akan mengecek username dan password yang dimasukkan dengan data username dan password yang telah tersimpan di database. Jika data username dan password cocok maka pengguna akan masuk ke dalam sistem, sebaliknya akan diberikan pesan peringatan jika tidak cocok.

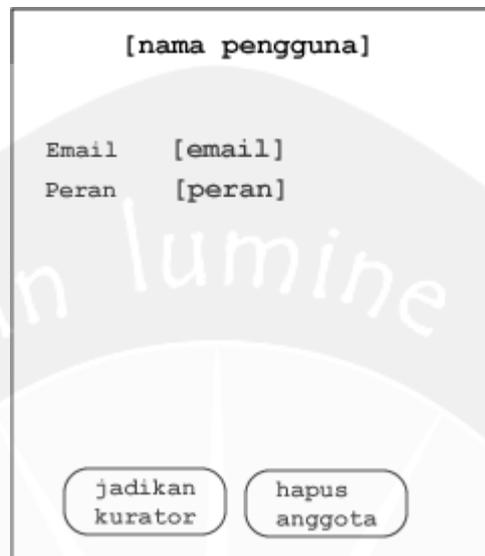
#### 4.4 Menu Utama



**Gambar 4.4 Rancangan Antarmuka Menu Utama**

Antarmuka pada gambar 4.4 ini adalah antar muka menu utama yang berisi tombol-tombol yang mengarah ke antarmuka lain. Selain itu antarmuka ini menampilkan nama pengguna yang masuk.

#### 4.5 Detail Anggota



[nama pengguna]

Email [email]  
Peran [peran]

jadikan kurator    hapus anggota

**Gambar 4.5 Rancangan Antarmuka Detail Anggota**

Antarmuka pada gambar 4.5 ini digunakan untuk menampilkan detail anggota yang ada dalam database dan melakukan pengelolaan anggota, termasuk didalamnya hapus anggota dan memberikan peran baru kepada anggota.

#### 4.6 Detail Permintaan

	[nama edu]
Satuan	[satuan_pendidikan]
Swa/neg	[swasta/negeri]
Alamat	[alamat]
SPP	[kisaran_spp]
Deskripsi	[deskripsi]
Telp	[telp]
Fax	[fax]
Email	[email]
Website	[website]
Ditambahkan oleh	[nama pengguna]
pada tanggal	[tanggal]
<input type="button" value="tolak"/> <input type="button" value="setujui"/> <input type="button" value="lihat&lt;br/&gt;peta"/>	

**Gambar 4.6 Rancangan Antarmuka Detail Permintaan**

Antarmuka pada gambar 4.6 ini digunakan untuk menampilkan detail permintaan penempatan unit pendidikan yang ada dalam database. Di dalamnya terdapat fungsi untuk menerima permintaan, atau menolak permintaan, serta melihat permintaan tersebut di dalam peta.

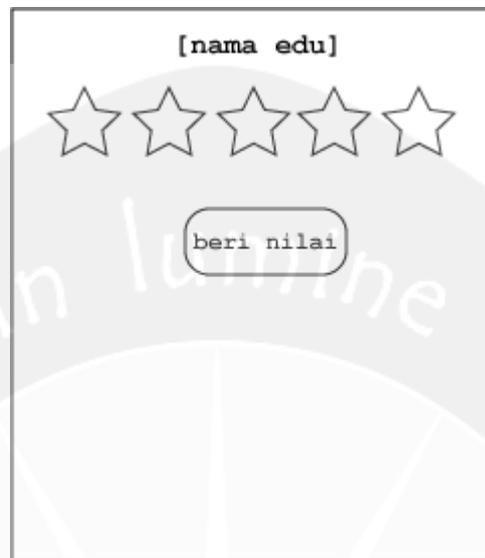
#### 4.7 Detail Edu

	[nama edu]	
	[rating]	
Satuan	[satuan_pendidikan]	
Swa/neg	[swasta/negeri]	
Alamat	[alamat]	
SPP	[kisaran_spp]	
Deskripsi	[deskripsi]	
Telp	[telp]	
Fax	[fax]	
Email	[email]	
Website	[website]	
Ditambahkan oleh [nama pengguna]		
pada tanggal [tanggal]		
<input type="button" value="rating"/>	<input type="button" value="komentar"/>	<input type="button" value="lihat&lt;br/&gt;peta"/>

**Gambar 4.7 Rancangan Antarmuka Detail Edu**

Antarmuka pada gambar 4.7 ini digunakan untuk menampilkan detail unit pendidikan yang ada dalam database. Di dalamnya terdapat fungsi untuk menambahkan komentar, memberikan rating dan melihat unit pendidikan tersebut di peta.

#### 4.8 Rating



**Gambar 4.8 Rancangan Antarmuka Rating**

Antarmuka pada gambar 4.8 ini digunakan untuk member nilai unit pendidikan. Nilai diberikan dengan ukuran bintang, dengan kisaran nilai 1 sampai 5.

#### 4.9 Peta Edu



**Gambar 4.9 Rancangan Antarmuka Peta Edu**

Antarmuka pada gambar 4.9 ini digunakan untuk menampilkan posisi unit pendidikan, posisi tempat tinggal, dan posisi pengguna. Juga dapat ditampilkan rute dari tempat tinggal atau posisi pengguna menuju posisi unit pendidikan.

#### 4.10 List Edu

[Icon Edu] [Nama Edu] [Jarak Edu] [Nilai]

**Gambar 4.10 Rancangan Antarmuka List Edu**

Antarmuka pada gambar 4.10 ini digunakan untuk menampilkan daftar nama unit pendidikan yang ada berdasarkan katalog yang dipilih. Daftar ini ditampilkan dengan icon satuan pendidikan yang sesuai. Terdapat pilihan fungsi untuk melihat detail unit pendidikan atau melihat unit tersebut di peta.

#### 4.11 List Layanan

[icon] [nama edu]

**Gambar 4.11 Rancangan Antarmuka List layanan**

Antarmuka pada gambar 4.11 ini digunakan untuk menampilkan daftar nama unit pendidikan yang didaftarkan oleh pengguna yang sedang login. Daftar ini ditampilkan dengan icon satuan pendidikan yang sesuai. Terdapat pilihan fungsi untuk melihat detail unit pendidikan atau menghapusnya.

#### 4.12 List Anggota



**Gambar 4.12 Rancangan Antarmuka List Anggota**

Antarmuka pada gambar 4.12 ini digunakan untuk menampilkan daftar nama anggota dan kurator, dan icon yang bersesuaian. Terdapat pilihan fungsi untuk melihat detail anggota atau menghapusnya.

#### 4.13 List Hasil Pencarian

[Icon Edu] [Nama Edu] [Jarak Edu] [Nilai]

**Gambar 4.13 Rancangan Antarmuka List hasil Pencarian**

Antarmuka pada gambar 4.13 ini digunakan untuk menampilkan daftar nama unit pendidikan yang ada berdasarkan katalog kueri pada saat pencarian. Daftar ini ditampilkan dengan icon satuan pendidikan yang sesuai. Terdapat pilihan fungsi untuk melihat detail unit pendidikan atau melihat unit tersebut di peta.

#### 4.14 Pencarian



The image shows a search form with the following elements:

- A checked checkbox labeled "mengandung nama:" followed by a text input field.
- A dropdown menu labeled "[jenis pendidikan]" with a right-pointing arrow.
- Two dropdown menus: "[satuan pendidikan]" and "[swasta /negeri]", both with right-pointing arrows.
- A section titled "kisaran spp:" containing two checked checkboxes: "minimal" and "maksimal", each followed by a text input field.
- A checked checkbox labeled "deskripsi mengandung:" followed by a text input field.
- A rounded rectangular button labeled "Cari" at the bottom.

**Gambar 4.14 Rancangan Antarmuka Pencarian**

Antarmuka pada gambar 4.14 ini digunakan untuk menampilkan form pencarian. Terdapat isian untuk mengisikan sebagian nama unit pendidikan yang akan dicari. Terdapat juga pilihan untuk memilih besar spp, status, dan satuan pendidikan serta kata kunci deskripsi untuk menyempitkan hasil pencarian.

#### 4.15 Katalog

Pendidikan awal

Pendidikan dasar

Pendidikan menengah

SMA

SMK

MA

SMA-LB

Pendidikan tinggi

**Gambar 4.15 Rancangan Antarmuka Katalog**

Antarmuka pada gambar 4.15 ini digunakan untuk menampilkan form katalog yang berisi pilihan jenis pendidikan dan satuan pendidikan. Jenis pendidikan berbentuk spoiler yang jika ditekan akan menampilkan satuan pendidikan yang bersangkutan.

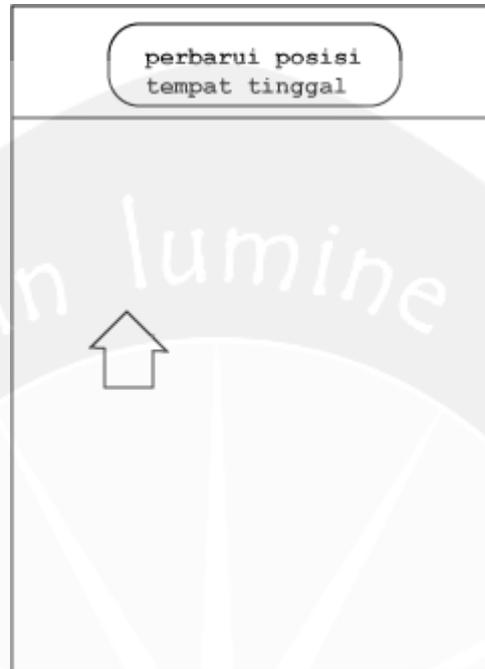
#### 4.16 Komentar

[komentator],[tanggal komentar] [isi komentar]
<input type="text"/>
<input type="button" value="beri komentar"/>

**Gambar 4.16 Rancangan Antarmuka Komentar**

Antarmuka pada gambar 4.16 ini digunakan untuk menampilkan komentar dari sebuah unit pendidikan. Form ini juga menyediakan sebuah isian untuk memberikan komentar. Setiap komentar disertai tanggal pemberian komentar dan nama komentator.

#### 4.17 Tempat Tinggal



**Gambar 4.17 Rancangan Antarmuka Tempat Tinggal**

Antarmuka pada gambar 4.17 ini digunakan untuk menampilkan posisi tempat tinggal di dalam peta dan diberikan fungsi untuk membarui posisi tempat tinggal.

#### 4.18 Daftar

The image shows a registration form titled "DAFTAR". It contains four text input fields: "Username", "Password", "Konfirmasi Password", and "email". Below the fields is a button labeled "daftar". The form is set against a background with a watermark of a university logo and the motto "serviens in lumine veritatis".

**Gambar 4.18 Rancangan Antarmuka Daftar Anggota**

Antarmuka pada gambar 4.18 ini digunakan untuk melakukan proses pendaftaran calon anggota ke dalam sistem CariEdu. Anggota harus memasukkan username, password, konfirmasi password dan email dengan benar pada textbox yang telah disediakan. Pada saat tombol login ditekan, sistem akan mengecek sudah ada tidaknya username dalam database serta benar tidaknya penulisan format email. Sistem juga memeriksa apakah password dan konfirmasi password adalah sama. Jika pengecekan sudah sesuai dengan syarat maka calon anggota tadi akan didaftarkan, jika tidak akan diberikan pesan peringatan jika tidak cocok.