

## MODUL 9

### USE CASE DIAGRAM



#### CAPAIAN PEMBELAJARAN

---

Mahasiswa dapat merancang use case diagram sesuai kebutuhan sistem dan menggambarannya dengan menggunakan tools desain.



#### KEBUTUHAN ALAT/BAHAN/SOFTWARE

---

Rational rose



#### DASAR TEORI

---

*Use Case* menjelaskan urutan kegiatan yang dilakukan *actor* dan sistem untuk mencapai suatu tujuan tertentu. Walaupun menjelaskan kegiatan namun *use case* hanya menjelaskan apa yang dilakukan oleh *actor* dan sistem, bukan bagaimana *actor* dan sistem melakukan kegiatan tersebut. Di dalam *use case* terdapat teks untuk menjelaskan urutan kegiatan yang disebut *Use Case Spesification*, yang terdiri atas:

a. **Nama *Use Case***

Penamaan *use case* sebaiknya diawali dengan kata kerja untuk menunjukkan suatu aktivitas. Contoh : Menerima Pendaftaran Mahasiswa Baru

b. **Deskripsi singkat (*Brief Description*)**

Menjelaskan secara singkat dalam 1 atau 2 kalimat mengenai tujuan *use case*.  
Contoh : - *Use Case* ini untuk mencatat data mahasiswa baru.  
- *Use case* ini dimulai saat mahasiswa datang untuk mendaftarkan diri.

c. **Aliran Normal (*Basic Flow*)**

Aliran normal menjelaskan interaksi antara *actor* dan sistem dalam keadaan normal, yaitu segala sesuatu berjalan dengan lancar, tiada halangan dalam mencapai tujuan dari *use case* ini.

Contoh :

- Petugas mengisi data mahasiswa baru.
- Setelah selesai mengisi semua data yang diperlukan, petugas mengkonfirmasi untuk menyimpan data tersebut.
- Sistem menuliskan data mahasiswa ke dalam *database*.
- Sistem memunculkan pesan bahwa proses pendaftaran sukses dilakukan.

**d. Aliran Alternatif (*Alternate Flow*)**

*Alternate flow* menjelaskan apa yang akan dilakukan bila suatu halangan atau hambatan terjadi sewaktu *use case* berlangsung, terutama berhubungan dengan *error* yang mungkin terjadi.

Contoh :

- Bila proses penulisan ke database gagal : sistem memunculkan pesan bahwa proses pendaftaran gagal dilakukan,
- Bila ada data yang diperlukan belum terisi : sistem meminta petugas untuk mengecek isian kembali

**e. *Special Requirement***

*Special requirement* berisi kebutuhan lain yang belum tercakup dalam aliran normal dan alternatif. Biasanya secara tegas dibedakan bahwa *basic flow* dan *alternate flow* menangani kebutuhan fungsional dari *use case*, sementara *special requirement* yang tidak berhubungan dengan kebutuhan fungsional.

Misalnya : kecepatan transaksi maksimum berapa cepat dan berapa lama, serta kapasitas akses yaitu jumlah *user* yang akan mengakses dalam waktu bersamaan.

**f. *Pre-Condition* :**

Menjelaskan persyaratan yang harus dipenuhi sebelum *use-case* bisa dimulai.

**g. *Post-Condition* :**

Menjelaskan kondisi yang berubah atau terjadi saat *use case* selesai dieksekusi.



## PRAKTIK

---

Kasus berikut akan digunakan untuk pertemuan saat ini dan berikutnya.

Sebelum menyelesaikan studinya, setiap mahasiswa diwajibkan untuk menyelesaikan Tugas Akhir (TA). Pada hakekatnya Tugas Akhir merupakan kegiatan akademik yang dirancang untuk melatih kemandirian dan tanggung jawab ilmiah mahasiswa sebagai calon ilmuwan, mulai dari pemilihan topik dan penyusunan rencana penelitian, pelaksanaan penelitian, evaluasi hasil penelitian, hingga penulisan laporan tugas akhir.

Pada Kampus Sumber Ilmu, Tugas Akhir ditulis dalam bentuk Laporan Tugas Akhir, yang akan dievaluasi oleh tim penguji yang dibentuk oleh pengelola program studi.

Ada tiga tahap yang harus dilakukan oleh mahasiswa tugas akhir, yaitu ujian proposal, ujian awal, dan ujian akhir. Untuk setiap tahap tersebut mahasiswa melakukan pembimbingan secara berkelanjutan kepada dosen pembimbing. Apabila pembimbing menyatakan telah menyetujui tugas akhir, maka pembimbing akan mengisi formulir check list format penulisan dan form check list isi naskah. Dokumen tugas akhir dan checklist yang sudah ditandatangani mahasiswa dan pembimbing diserahkan kepada administrator. Administrator akan menyusun jadwal ujian serta menentukan dosen penguji. Setelah jadwal ujian dan dosen penguji ditentukan maka mahasiswa menjalankan ujian sesuai jadwal. Pada saat ujian dosen penguji memberikan penilaian sesuai dengan borang evaluasi ujian Tugas Akhir. Hasil penilaian akan diserahkan kepada pihak administrator untuk di dokumentasikan.

Pada sistem yang dijelaskan sebelumnya, semua pekerjaan terpusat pada Administrator. Hal tersebut mengakibatkan beban kerja administrator lebih tinggi dan pekerjaan dokumentasi akan menjadi lebih lama. Dengan melihat permasalahan tersebut maka dikembangkan sistem informasi penilaian tugas akhir.

Berikut adalah ketentuan Tugas akhir.

a. Syarat ujian Proposal

- Naskah proposal, penulisan tabel, gambar dan daftar pustaka sudah memenuhi panduan penulisan proposal.

- Pengumpulan naskah paling lambat seminggu sebelum tanggal ujian, untuk menentukan dosen penguji.
- Bilamana tidak memenuhi persyaratan tersebut diatas, ujian dilaksanakan pada bulan berikutnya.
- Keterangan :
  - Kualitas proposal dievaluasi secara kuantitatif
  - Kemungkinan hasil evaluasi : diterima, perbaikan tanpa sidang lagi, perbaikan dengan sidang.

b. Syarat ujian awal

- Naskah tugas akhir sudah memenuhi check list format
- Naskah tugas akhir sudah memenuhi check list content /isi naskah
- Mahasiswa menandatangani pernyataan sudah memenuhi check list format dan isi, dan diketahui oleh dosen pembimbing
- Pengumpulan naskah paling lambat seminggu sebelum tanggal ujian, untuk menentukan dosen penguji
- Ujian dijadwalkan setiap bulan, pada minggu pertama
- Bilamana tidak memenuhi persyaratan tersebut diatas, ujian dilaksanakan pada bulan berikutnya.

c. Syarat ujian Akhir

- Naskah sudah memenuhi check list format
- Naskah tesis sudah memenuhi check list content/isi naskah
- Mahasiswa menandatangani pernyataan sudah memenuhi check list format dan isi, dan diketahui oleh dosen pembimbing
- Menyerahkan naskah publikasi sesuai dengan format jurnal yang dipilih
- Memenuhi semua usulan pebaikan yang disampaikan oleh penguji pada saat ujian awal
- Pengumpulan naskah paling lambat seminggu sebelum tanggal ujian, untuk menentukan dosen penguji
- Bilamana tidak memenuhi persyaratan tersebut diatas, ujian dilaksanakan pada bulan berikutnya.

Keterangan:

- Ujian akhir dievaluasi secara kuantitatif, dari sisi kualitas penulisan, kualitas kemampuan dan penyajian, serta kualitas penugasan dan penalaran.
- Kemungkinan hasil evaluasi adalah A, B, C, D, dengan persyaratan penyelesaian tugas akhir tidak lebih dari 1 tahun.
- Penyelesaian tugas akhir yang >1 tahun tidak dimungkinkan mendapatkan nilai A
- D: tidak lulus, C: operasional mengulang
- Form check list format dan check list isi tugas akhir ada pada administrasi prodi.

### **Pelaksanaan ujian awal**

Waktu ujian masimum 120 menit yang dibagi dengan alokasi waktu sebagai berikut,

1. Mahasiswa presentasi maksimum 20 menit
2. Tanya jawab untuk setiap penguji maksimum 30 menit
3. Rekapitulasi hasil dan penentuan hasil.

Ketentuan ujian awal dan ujian akhir :

#### 1. Ketua penguji :

- Memimpin pelaksanaan ujian dan menguji
- Mengatur perwaktuan pelaksanaan ujian agar ujian tepat 120 menit
- Menguji moderator ujian, mengalihkan waktu tanya jawab pada penguji lain, bila melebihi batas maksimal 30 menit
- Merangkap hasil ujian

#### 2. Pembimbing

- Mencatat saran/komentar penguji pada buku notulen dan ujian

#### 3. Penguji

- Melakukan tanya jawab, selama maksimum 30 menit
- Memberi nilai angka pada form evaluasi ujian

### **Pelaksanaan ujian awal/akhir**

1. Dilaksanakan pada hari Jumat dan Sabtu setiap minggu pertama setiap bulam
2. Jumlah penguji sebanyak 3 orang dosen penguji, paling sedikit 1 orang sesuai bidang dan 1 orang dosen pembimbing .
3. Ketua penguji ditentukan oleh ka prodi
4. Pembimbing berperan sebagai notulis
5. Jika pembimbing tidak hadir maka ujian tetap berjalan. Pembimbing akan diganti dengan dosen lain dengan bobot penilaian 1 bukan 2

6. Bilamana dosen penguji berhalangan supaya memberitahui ke prodi paling lambat 2 hari setelah penetapan jadwal.
7. Peserta ujian harus menyerahkan naskah ujian dan naskah publikasi yang sudah memenuhi syarat paling lambat 1 minggu setelah pelaksanaan ujian.
8. Pengumuman pendjadwalan ujian paling lambat 5 hari sebelum pelaksanaan ujian.

## Rancangan sistem

### 1. Requirement List

Berdasarkan permasalahan yang telah diuraikan, berikut adalah requirement list yang dapat didefinisikan.

No	Requirement	Use case
1	Administrator dapat mengentry data dosen	Entry Data Dosen
2	Administrator dapat mengentry data mahasiswa	Entry Data Mahasiswa
3	Administrator dapat mengentry data tugas akhir mahasiswa berupa proposal, naskah ujian awal, dan naskah ujian akhir	Entry TA
4	Administrator dapat mengentry borang penilaian ujian proposal, ujian awal, dan ujian akhir	Entry Borang
5	Administrator dapat mengentry rincian borang penilaian ujian proposal, ujian awal, dan ujian akhir	Entry Detail Borang
6	Administrator dapat menambah jadwal ujian dengan mengecek check list terlebih dahulu untuk memastikan bahwa naskah ujian sudah disetujui oleh pembimbing. Penambahan jadwal sekaligus menentukan dosen penguji, waktu dan tempat ujian	Entry Jadwal
7	Penguji dan administrator dapat melihat jadwal ujian	Lihat Jadwal
8	Pembimbing dan penguji bisa memberikan penilaian dan revisi terhadap hasil ujian tugas akhir mahasiswa dengan mengisi borang evaluasi	Penilaian
9	Penguji dan administrator dapat melihat nilai	Lihat Nilai

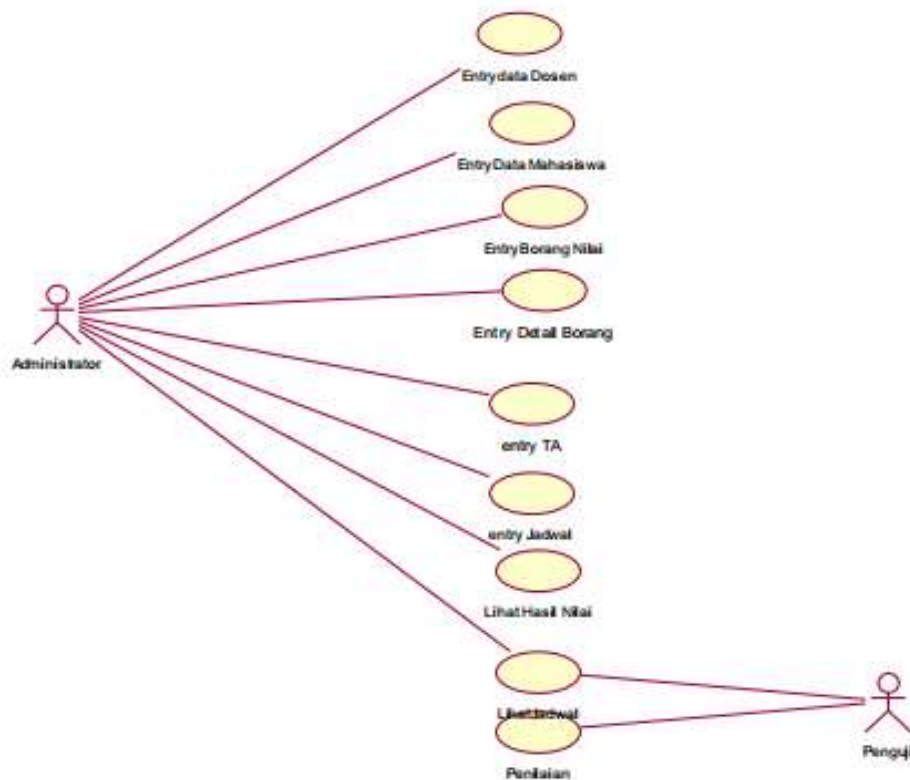
## 2. Komponen User (Actor)

Actor menggambarkan orang, system atau external entitas / stakeholder yang menyediakan atau menerima informasi dari sistem. Dalam sistem yang dibuat komponen user (actor), meliputi :

- a. Administrator : yaitu petugas atau staf bagian administrasi yang bertugas memasukkan data mahasiswa, dosen, TA dan mengelola tugas akhir mahasiswa seperti penjadwalan ujian, pengumuman hasil ujian
- b. Penguji : yaitu dosen yang ditunjuk untuk menguji TA mahasiswa termasuk juga dosen pembimbing. Penguji bisa melihat jadwal dari mahasiswa yang akan diuji dan bisa melakukan penilaian dengan mengisi borang penilaian.

## 3. Use Case Diagram

Dari actor dan requirement yang telah didefinisikan, dapat digambarkan use case diagram yang dirancang seperti yang ada pada Gambar 9.1



Gambar 9.1 Use case diagram system ujian tugas akhir

## 4. Deskripsi Use case diagram

Berikut beberapa deskripsi dari use case yang ada pada Gambar 9.1.

- i. Entry Data Dosen

Nama use case : entry data dosen

Actor : administrator

Tujuan : use case ini bertujuan untuk menggambarkan proses masukan data dosen yang merupakan pembimbing dan penguji tugas akhir mahasiswa

Deskripsi : proses ini adalah sebuah kegiatan untuk mendaftarkan dosen baru ataupun mengubah data dosen yang sudah ada

Precondition : 1. Administrator sudah harus memiliki ID agar bisa masuk di form data dosen  
2. Administrator sudah harus berada di menu memasukkan data dosen

Post-condition : data dosen baru sudah tersimpan

#### Skenario

Actor Action	System Respon
1. Menekan tombol Entry Data Dosen 3. Mengisi data dosen 5. Menekan tombol simpan	2. Menampilkan form entry Data Dosen 4. Menampilkan isian data dosen 6. Menyimpan data dosen dan menampilkan data dosen yang telah serta mengkosongkan isian data dosen untuk data entry data dosen lag
<p>Alur alternative No 6</p> <p>Jika data dosen yang dimasukkan tidak valid, maka akan menampilkan pesan “tidak sukses disimpan”</p>	

#### ii. Penilaian

Nama use case : Penilaian

Actor : Penguji

Tujuan : use case ini bertujuan untuk menampilkan form untuk melakukan penilaian terhadap proposal, ujian awal dan ujian akhir



Deskripsi : proses ini adalah sebuah kegiatan untuk memberikan penilaian terhadap hasil ujian mahasiswa

Precondition : 1. Penguji sudah harus memiliki ID agar bisa masuk di form penilaian  
2. Penguji sudah harus berada di menu penilaian

Post-condition : Data penilaian mahasiswa tersimpan

#### Skenario

Actor Action	System Respon
1. Menekan tombol penilaian	2. Menampilkan form penilaian dengan nim mahasiswa yang akan dinilai
3. Memilih nim yang akan dinilai	4. menampilkan informasi mengenai tugas akhir mahasiswa
5. Memilih form penilaian	6. menampilkan borang penilaian
7. Mengisi borang penilaian	8. menerima nilai yang diberikan
9. Menekan tombol simpan setelah seluruh borang penilaian sudah diisi	10. Menyimpan hasil penilaian
<p>Alur alternative No 10</p> <p>Jika data nilai yang dimasukkan tidak valid, maka akan menampilkan pesan "tidak sukses disimpan"</p>	



#### LATIHAN

---

1. Bentuk kelompok proyek anda sebanyak 3 orang
2. Tentukan kasus yang akan diselesaikan oleh kelompok anda
3. Diskusikan kasus anda dengan dosen pengampu
4. Rancang requirement dan use case diagram proyek anda seperti yang ada pada bagian praktik



#### TUGAS

---

Diberikan oleh dosen pengampu



## REFERENSI

---

Buku-buku pemodelan berorientasi objek

## MODUL 10

### CLASS DIAGRAM



#### CAPAIAN PEMBELAJARAN

---

Mahasiswa dapat mendefinisikan kebutuhan class diagram dari system yang akan dibangun



#### KEBUTUHAN ALAT/BAHAN/SOFTWARE

---

Rational Rose



#### DASAR TEORI

---

*Class diagram* menunjukkan hubungan antar *class* dalam sistem yang sedang dibangun dan bagaimana diagram tersebut saling berkolaborasi untuk mencapai suatu tujuan. *Class diagram* umumnya tersusun dari elemen *class*, *interface*, *dependency*, *generalization* dan *association*.

Relasi ***dependency*** menunjukkan bagaimana ketergantungan terjadi antar *class* yang ada. Relasi ***generalization*** menunjukkan bagaimana suatu *class* menjadi *superclass* dari *class* lainnya dan *class* yang lain tersebut menjadi *subclass* dari *class* tersebut.

Relasi ***association*** menggambarkan navigasi antar ***class***, berapa banyak obyek lain bisa berhubungan dengan satu obyek (***multiplicity*** antar ***class***), dan apakah suatu *class* menjadi bagian dari *class* lainnya (***aggregation***).

Relasi ***agregasi*** adalah bentuk khusus dari asosiasi yang menggambarkan seluruh bagian suatu obyek merupakan bagian dari obyek yang lain.

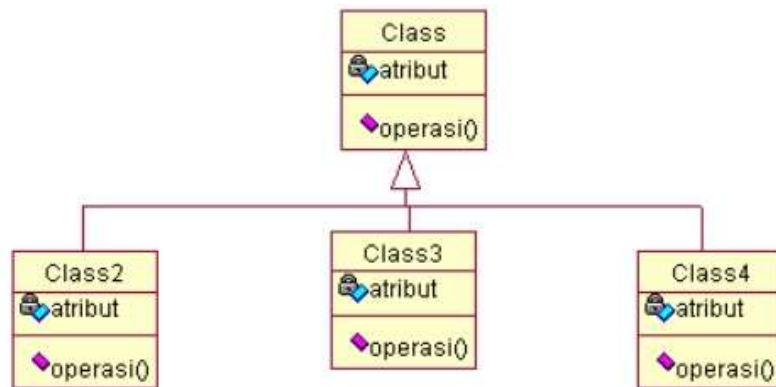
***Class diagram*** juga digunakan untuk menggambarkan desain statis dari sistem yang sedang dibangun.

Bentuk dari *Class diagram* yang mempunyai hubungan asosiasi ditunjukkan pada gambar 10.1



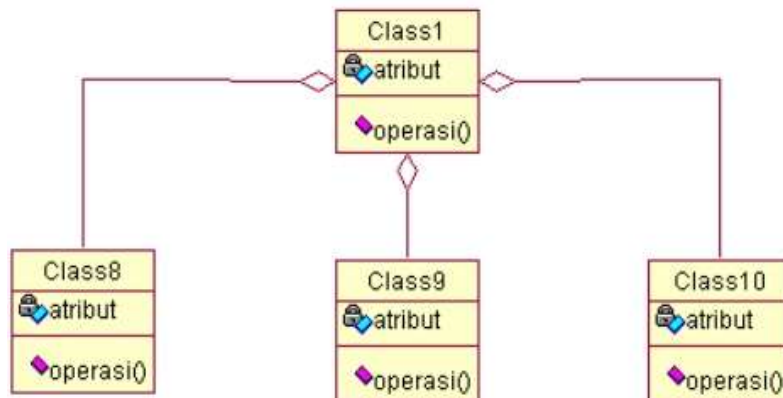
Gambar 10.1 Relasi Asosiasi antar Class

Bentuk dari *Class diagram* yang mempunyai hubungan *generalisasi* ditunjukkan pada gambar 10.2.



Gambar 10.2 Relasi Generalisasi Class Induk ke Sub Class Sub Class

Sedangkan *Class diagram* yang mempunyai hubungan agregasi ditunjukkan pada gambar 10.3.



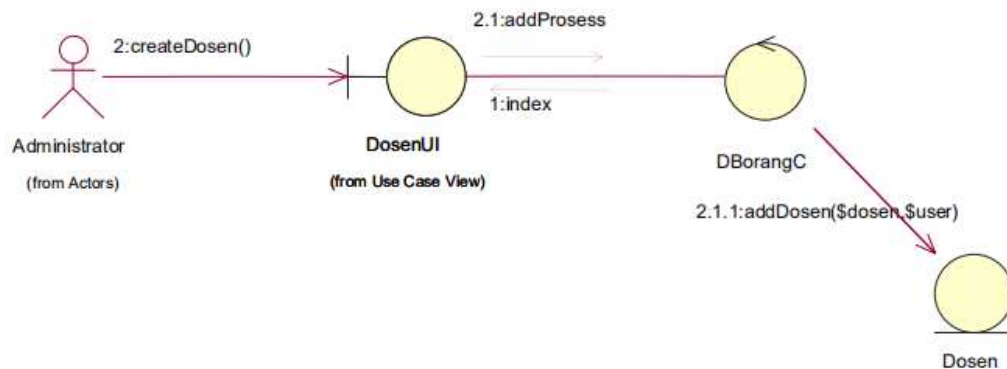
Gambar 10.3 Relasi Agregasi antar Class



## PRAKTIK

Dari kasus yang ada di modul 9. Dapat digambarkan communication diagram dan class diagram untuk use case entry data dosen dan penilaian sebagai berikut.

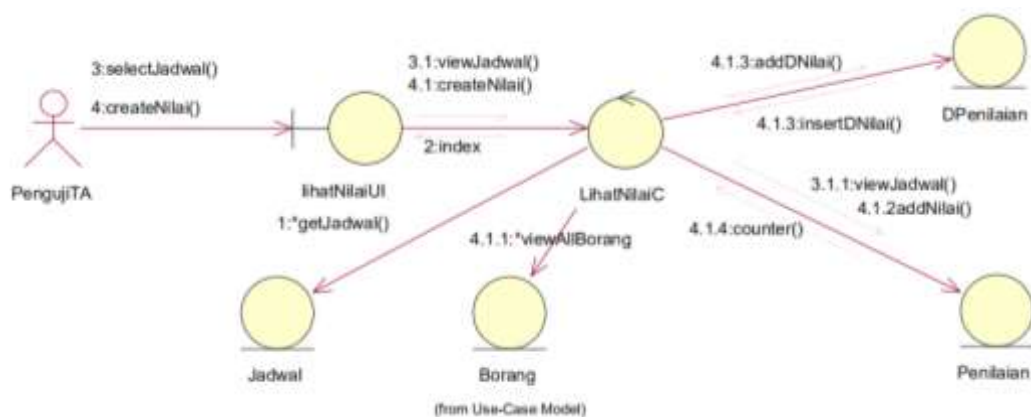
### Analisis communication diagram untuk entry data dosen



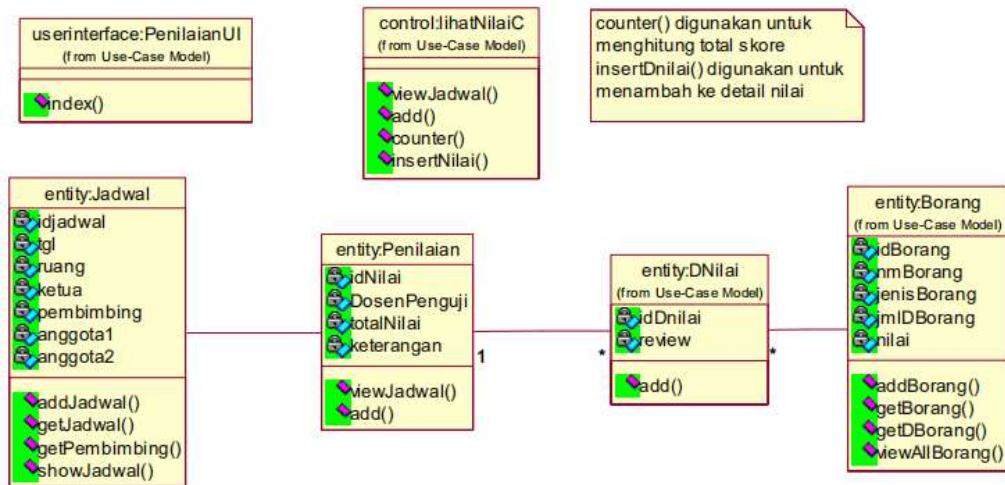
### Analisis class diagram untuk entry data dosen



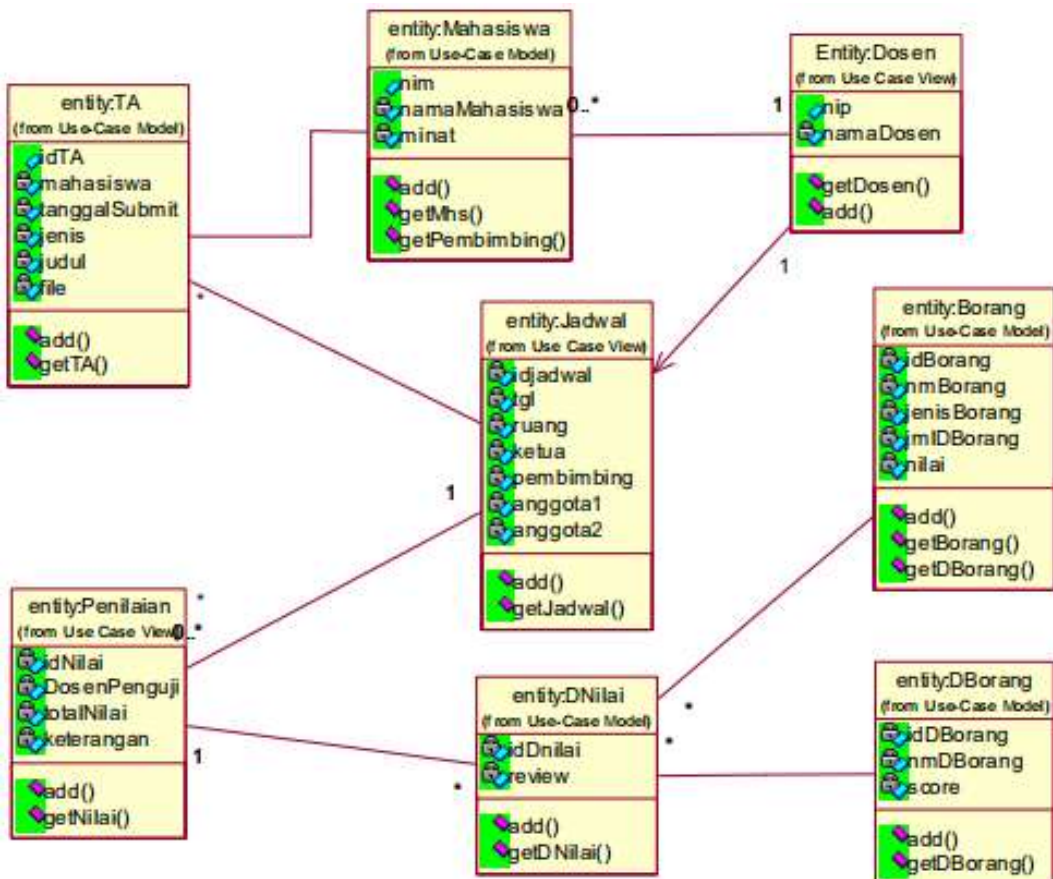
### Analisis communication diagram untuk penilaian



## Analisis Class diagram untuk penilaian



## Class Diagram gabungan untuk Stereotype Entity



LATIHAN

Gambarkan communication diagram dan class diagram dari kasus proyek kelompok anda masing-masing.



### **TUGAS**

---

Diberikan oleh masing-masing pengampu



### **REFERENSI**

---

Buku-buku pemodelan berorientasi objek

## MODUL 11

### SEQUENCE DIAGRAM



#### CAPAIAN PEMBELAJARAN

---

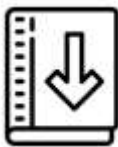
Mahasiswa dapat membuat Sequence Diagram untuk studi kasus tertentu



#### KEBUTUHAN ALAT/BAHAN/SOFTWARE

---

Rational rose



#### DASAR TEORI

---

*Sequence Diagram* menjelaskan secara detail tentang urutan proses yang dilakukan dalam sistem untuk mencapai tujuan dari use case, interaksi yang terjadi antar class, operasi apa saja yang terlibat, urutan antar operasi, dan informasi yang diperlukan oleh masing-masing operasi.

Pembuatan *sequence diagram* merupakan aktivitas yang paling kritical dari proses desain, karena artifak inilah yang menjadi pedoman dalam proses pemrograman nantinya dan berisi aliran kontrol dari program. Oleh karena itu, berharga untuk meluangkan waktu lebih lama di pembuatan *sequence diagram* ini untuk menghasilkan *sequence diagram* yang terdesain dengan baik.

*Sequence diagram* biasanya tersusun dari elemen obyek, interaction dan message. Interaction menghubungkan 2 obyek dengan pesannya. Diagram ini menjelaskan aspek dinamis dari sistem yang sedang dibangun.

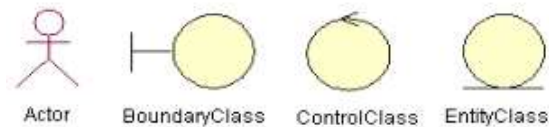
Di dalam *Sequence Diagram*, terdapat pelaku (*actor*), *boundary class*, *control class*, dan *entity class*.

- *Boundary Class* adalah kelas yang memodelkan interaksi antara satu atau lebih actor dengan sistem. *Boundary* memodelkan bagian dari system yang bergantung pada pihak lain disekitarnya dan merupakan pembatas sistem dengan dunia luar.
- *Control Class* digunakan untuk memodelkan “perilaku mengatur”, khusus untuk satu atau beberapa *use-case* saja.



- *Entity Class* memodelkan informasi yang harus disimpan oleh sistem. *Entity Class* memperlihatkan struktur data dari suatu sistem.

Adapun notasi dari *actor*, *boundary class*, *control class* dan *entity class* adalah seperti tampak pada gambar 11.1.



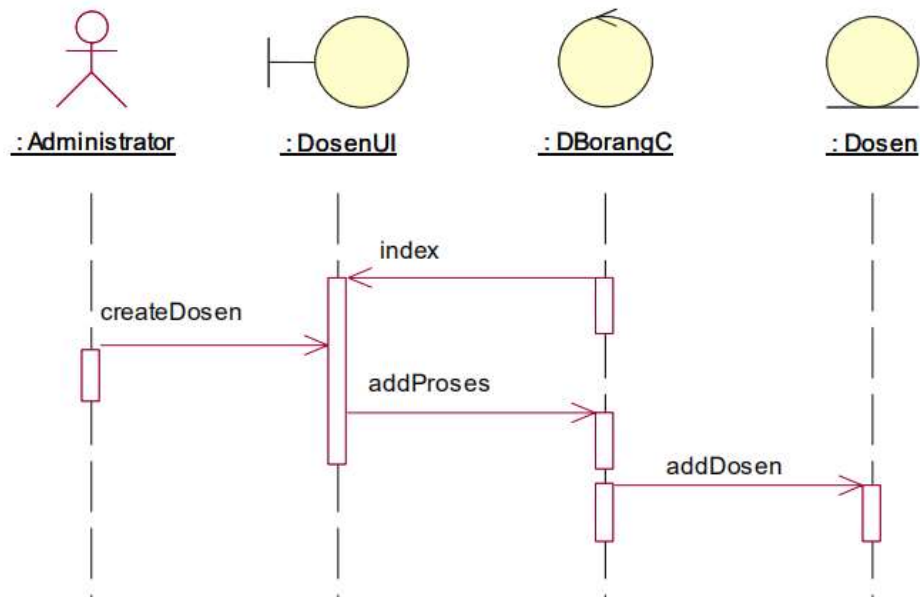
Gambar 11.1 Notasi/symbol dalam Sequence Diagram



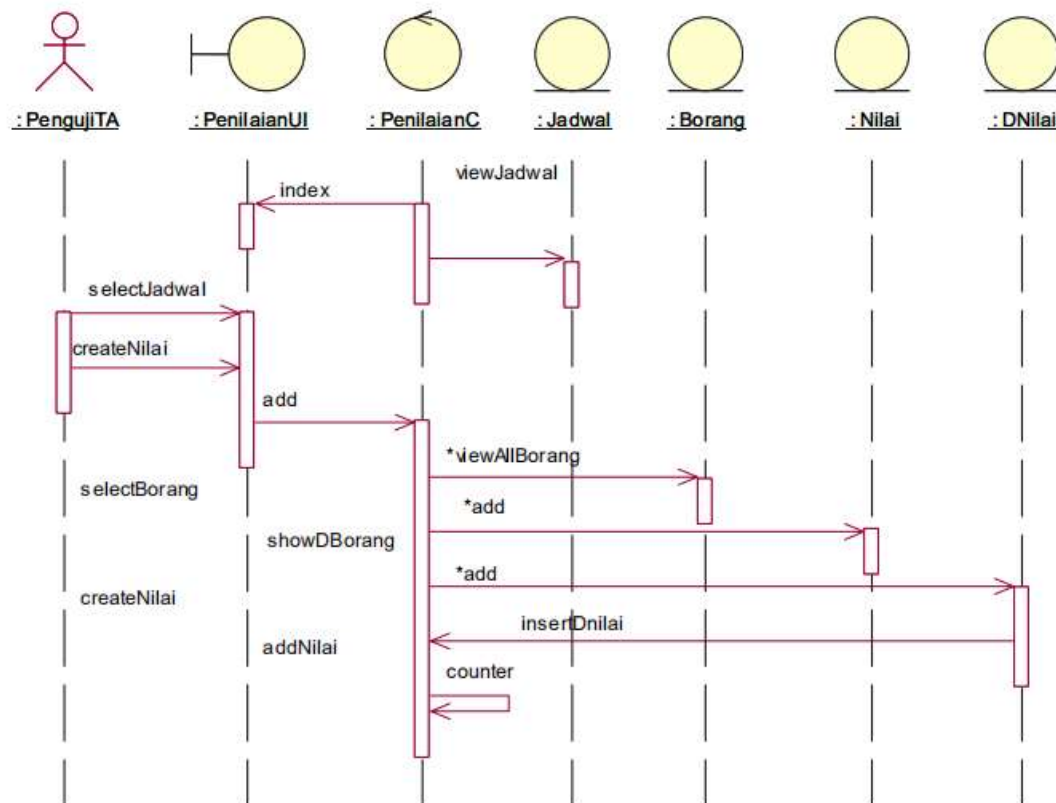
### PRAKTIK

Dari kasus yang ada di pertemuan ke-9, untuk use case entry data dosen dan penilaian dapat digambarkan sequence diagramnya sebagai berikut.

#### Sequence diagram untuk use case entry data dosen



## Sequence diagram untuk penilaian



### LATIHAN

Gambarkan sequence diagram dari kasus kelompok anda



### TUGAS

Diberikan oleh dosen pengampu



### REFERENSI

Buku-buku pemodelan berorientasi objek