

SEQUENCE DIAGRAM

Pendahuluan

- ▶ *Sequence diagram* menggambarkan interaksi antar objek di dalam dan di sekitar sistem (termasuk pengguna, display/form) berupa *message* yang digambarkan terhadap waktu.
- ▶ *Sequence diagram* terdiri atas dimensi vertikal (waktu) dan dimensi horizontal (objek-objek yang terkait).
- ▶ *Sequence diagram* biasa digunakan untuk menggambarkan skenario atau rangkaian langkah-langkah yang dilakukan sebagai respons dari sebuah *event* untuk menghasilkan *output* tertentu. Diawali dari apa yang men-*trigger* aktivitas tersebut, proses dan perubahan apa saja yang terjadi secara internal dan *output* apa yang dihasilkan.
- ▶ Diagram ini secara khusus berasosiasi dengan use case diagram dan memperlihatkan tahap demi tahap apa yang seharusnya terjadi untuk menghasilkan sesuatu di dalam use case

Tujuan

- ▶ Digunakan untuk memperlihatkan interaksi antar obyek dalam perintah yang berurut.
- ▶ Tujuan utama adalah mendefinisikan urutan kejadian yang dapat menghasilkan output yang diinginkan
- ▶ Mirip dengan activity diagram
 - ▶ Menggambarkan alur kejadian sebuah aktivitas
 - ▶ Lebih detail dalam menggambarkan aliran data, termasuk data atau behaviour yang dikirimkan atau diterima
 - ▶ Namun kurang mampu menjelaskan detail dari sebuah algoritma (loop, branching)

Komponen

- Actor
- Interface (Boundary)
- Proses pembacaan (Control)
- Nama table (Entity)

Simbol

a. An Actor



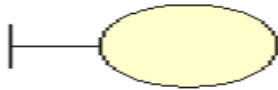
Menggambarkan orang yang sedang berinteraksi dengan sistem

b. Entity Class



Menggambarkan hubungan kegiatan yang akan dilakukan

c. Boundary Class



Menggambaran sebuah penggambaran dari form

d. Control Class



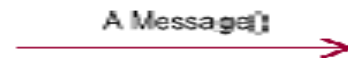
Menggambarkan penghubung antara boundary dengan tabel

e. A focus Of Control & A life line



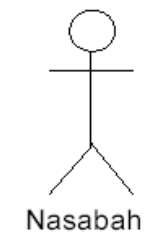
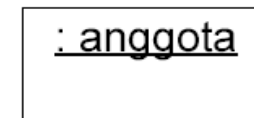
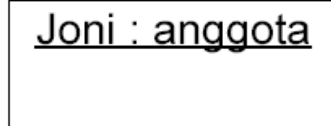
Menggambarkan tempat mulai dan berakhirnya sebuah message

f. A message



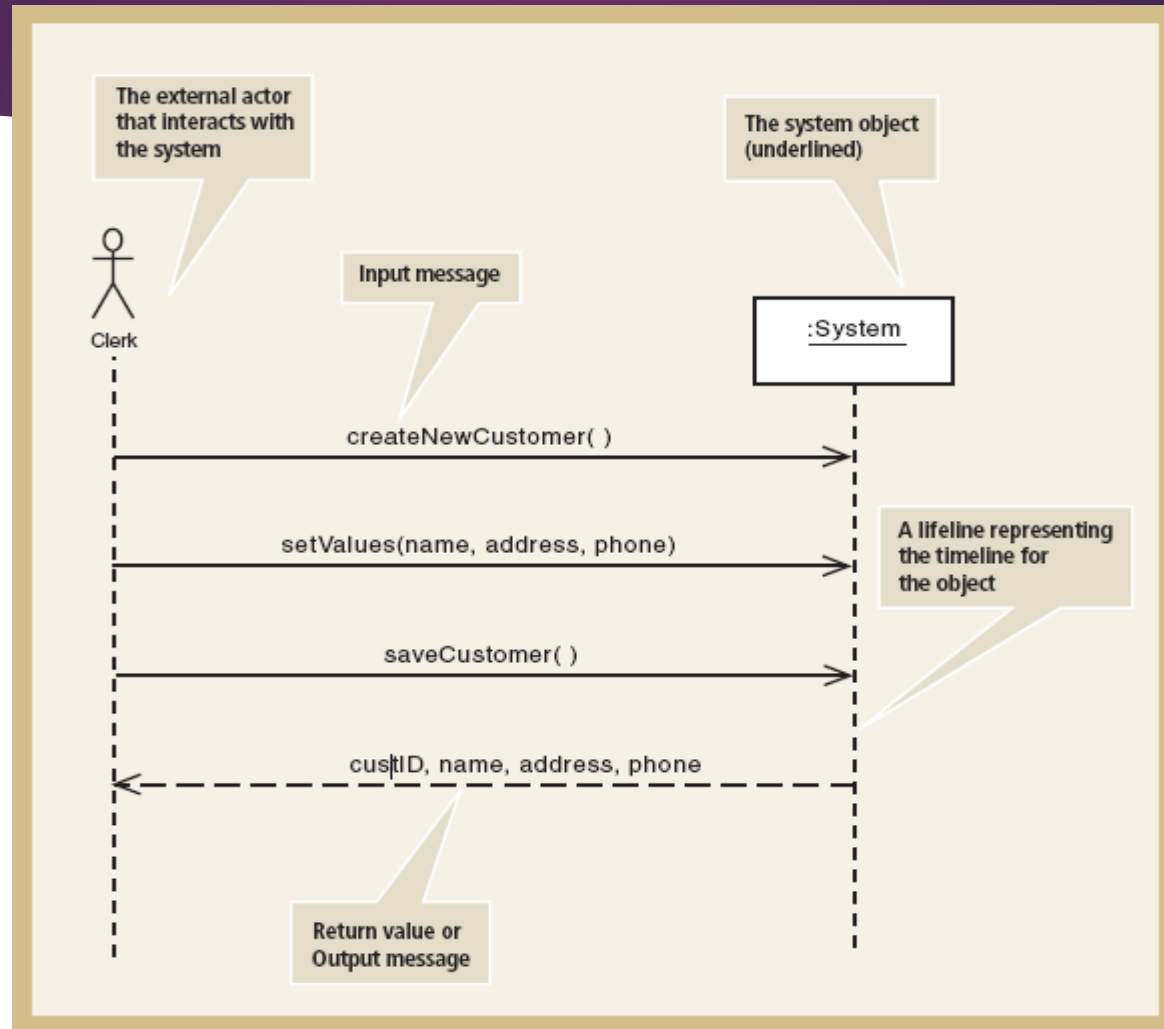
Menggambarkan Pengiriman Pesan

- ▶ Partisipan : obyek atau entitas yang bertindak dalam sequence diagram
- ▶ Message : komunikasi antar obyek partisipan
- ▶ Terdapat 2 tipe garis yaitu vertikal dan horisontal
 - ▶ Vertikal : waktu → maju berdasarkan waktu
 - ▶ Horisontal : obyek mana yang beraksi
- ▶ Nama Object / Class
 - ▶ name bersifat optional
 - ▶ boxes berupa object diberikan tanda garis bawah
 - ▶ object yang tidak bernama disebut anonymous objects
 - ▶ boxes berupa actor dapat juga digambar dengan stick figure

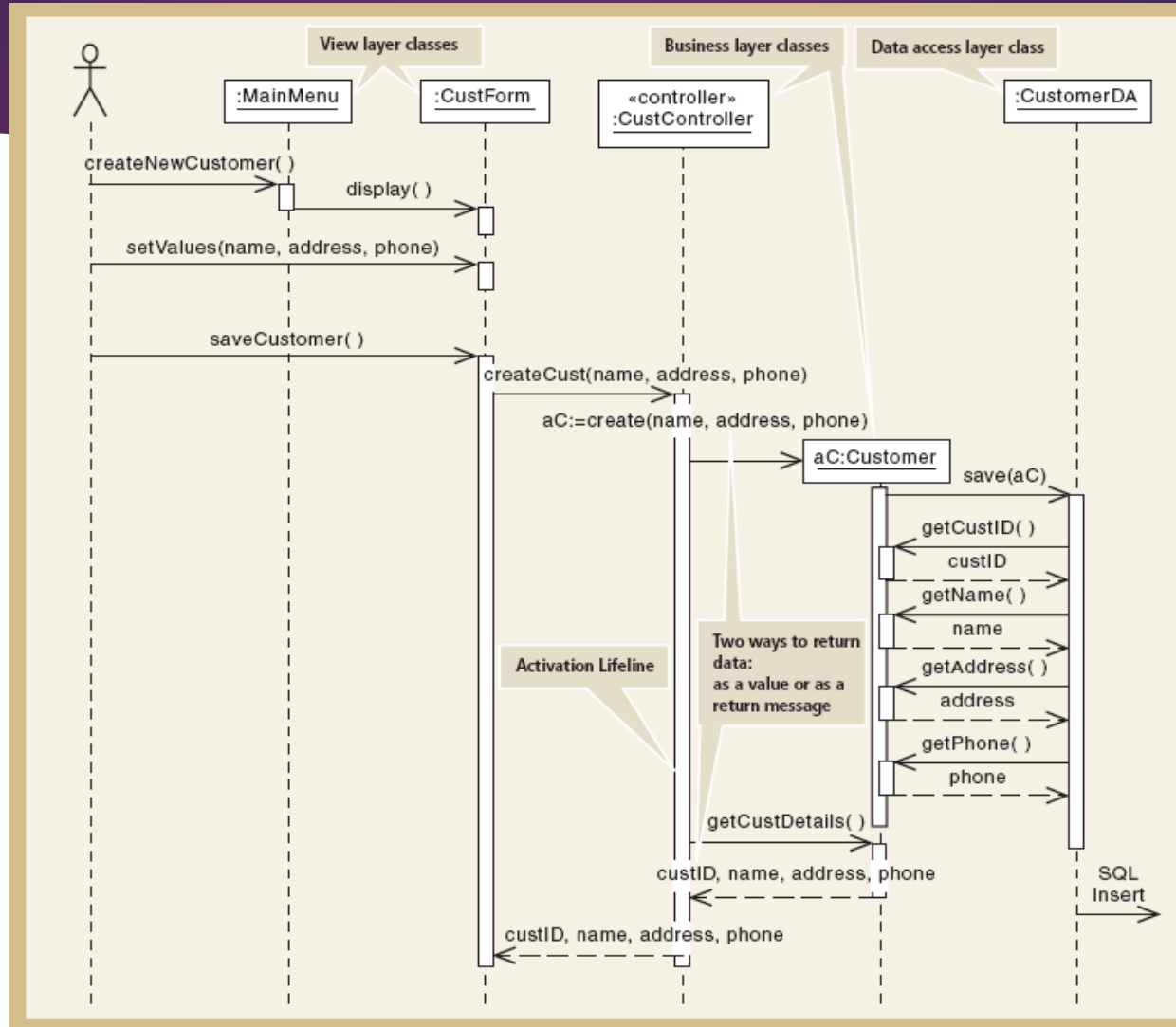


Daftar Istilah

- ▶ Participant
 - ✓ Obyek yang terkait dengan sebuah urutan proses
- ▶ Lifeline
 - ✓ Menggambarkan daur hidup sebuah obyek
- ▶ Activation
 - ✓ Suatu titik waktu dimana sebuah obyek mulai berpartisipasi didalam sebuah sequence
 - ✓ Ditandai dengan sebuah bar
- ▶ Time
 - ✓ Elemen penting dalam sequence diagram
 - ✓ Konteksnya adalah **urutan, bukan durasi**
- ▶ Return
 - ✓ Suatu hasil kembalian sebuah operasi
 - ✓ Operasi mengembalikan hasil, tetapi boleh tidak ditulis

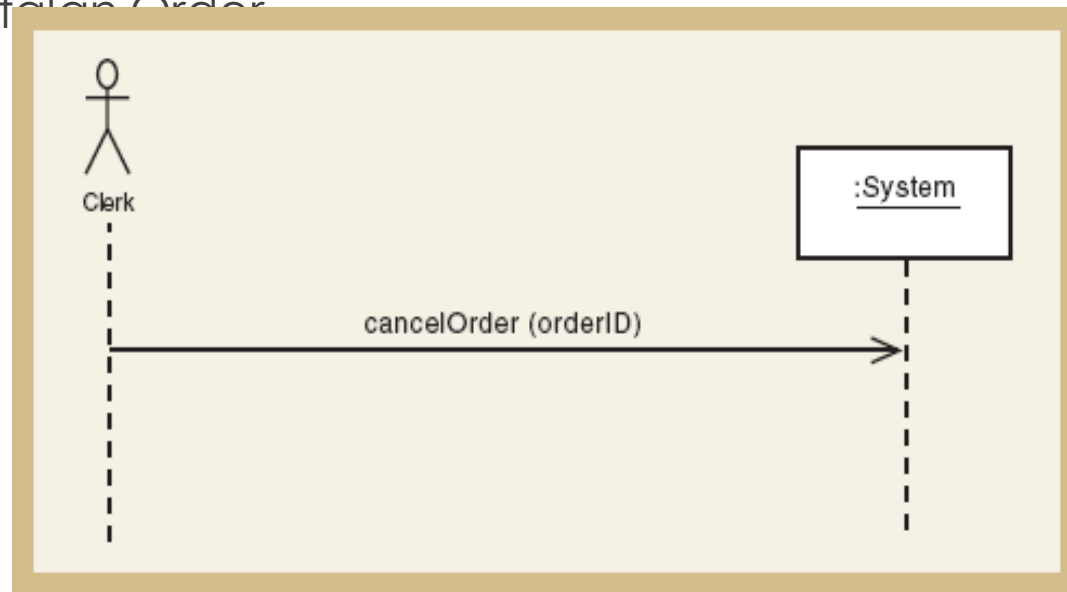


Detail SD

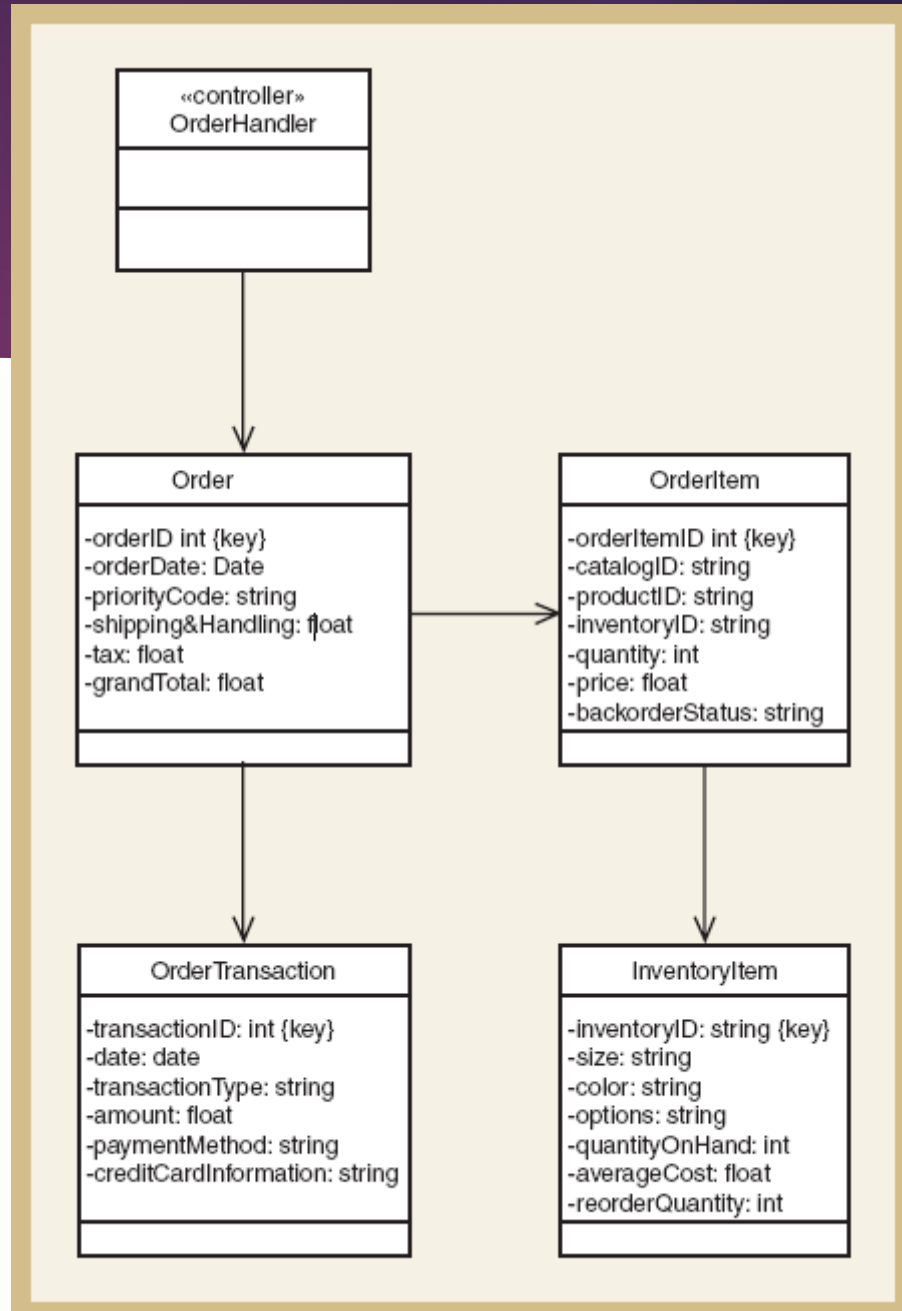


Hubungan Class diagram dan Sequence diagram

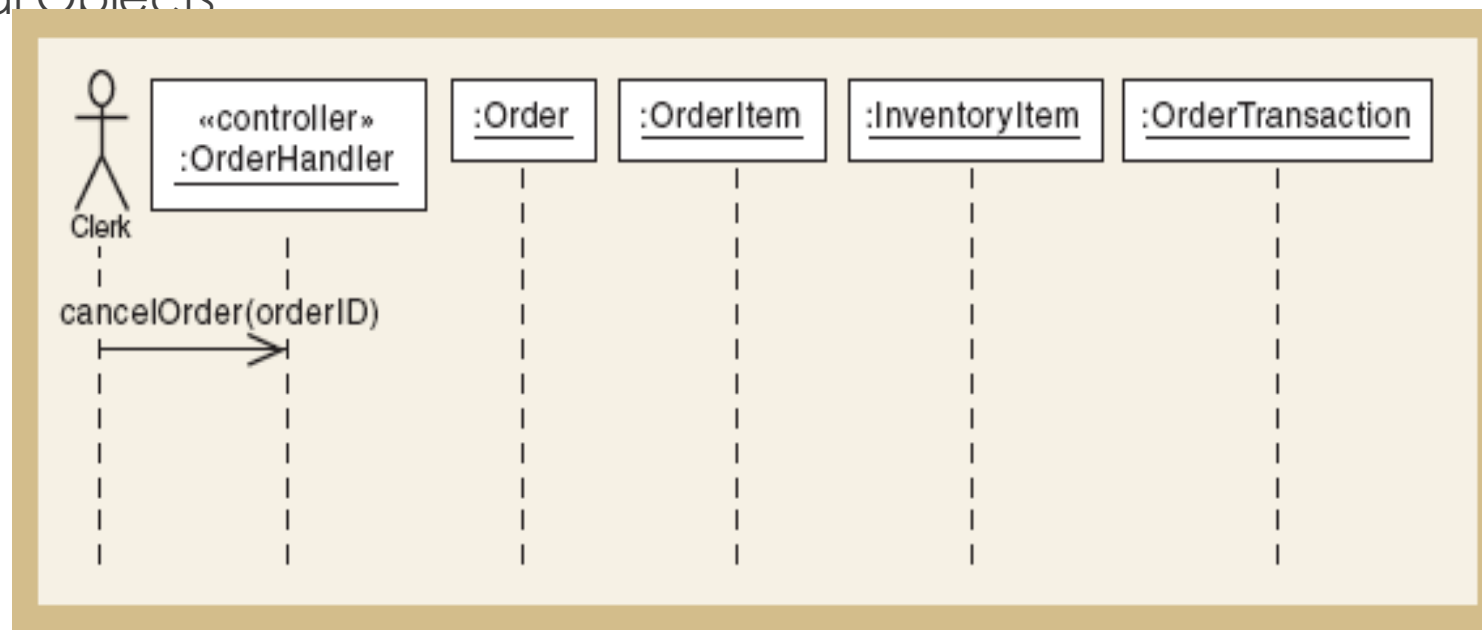
- ▶ Contoh : Pembatalan Order



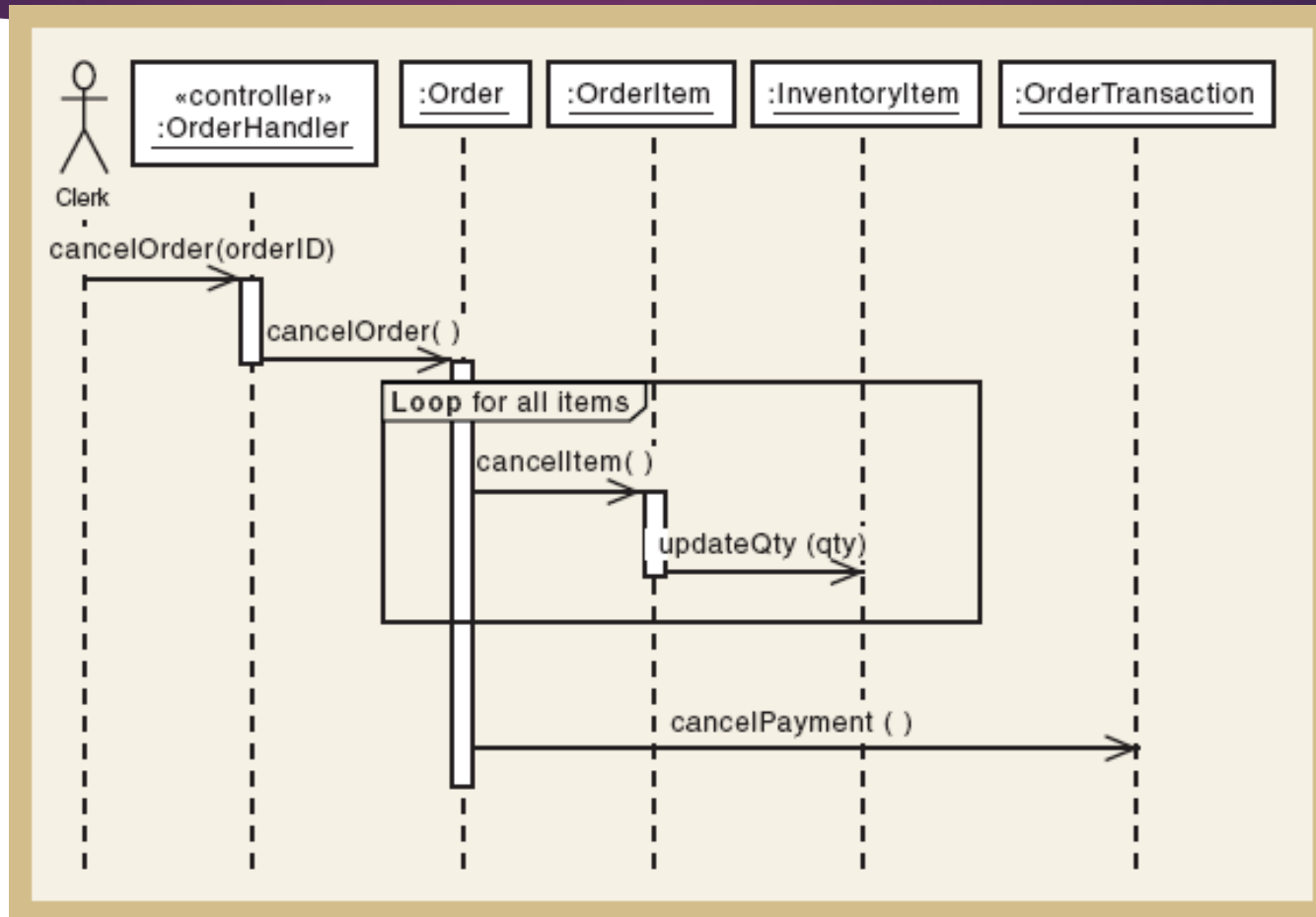
► Class Diagram



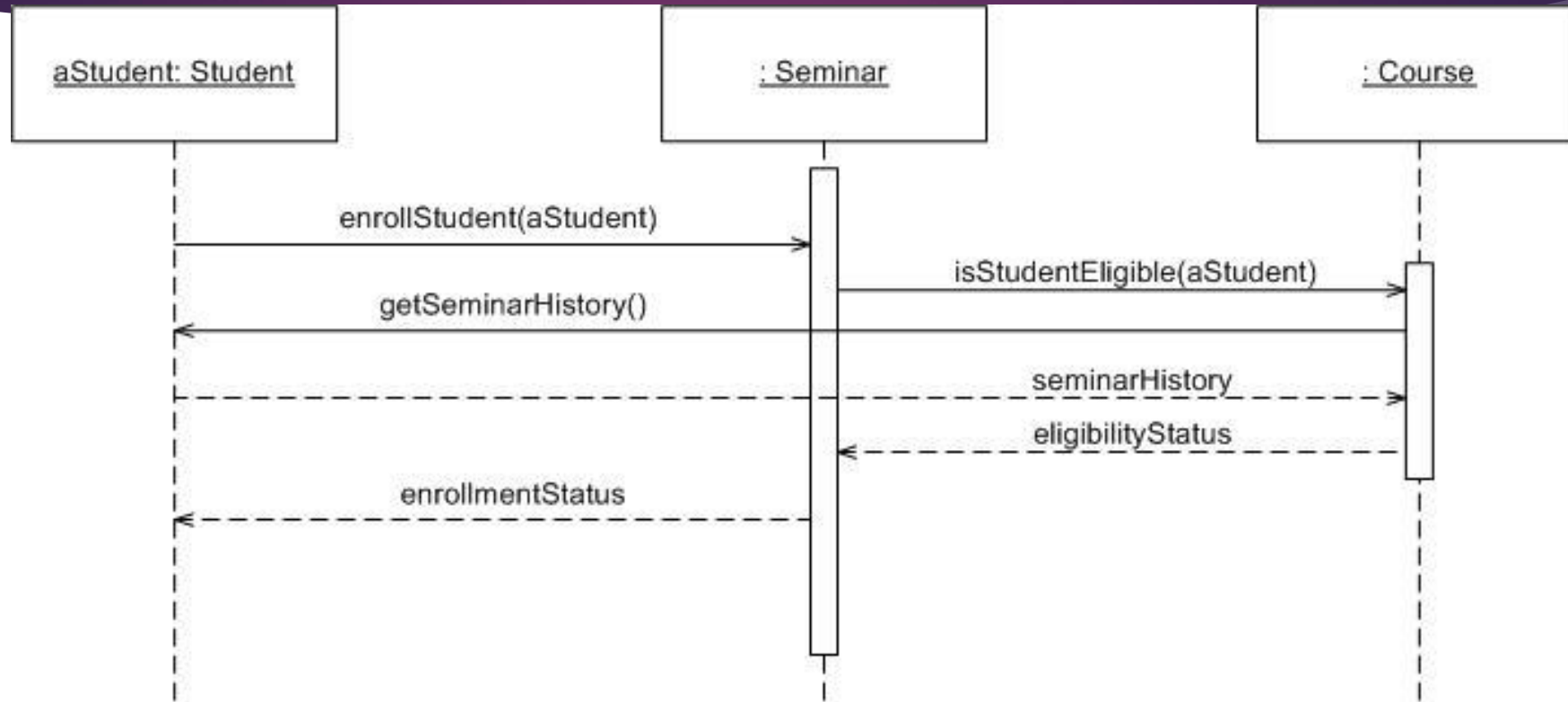
► Potential Objects



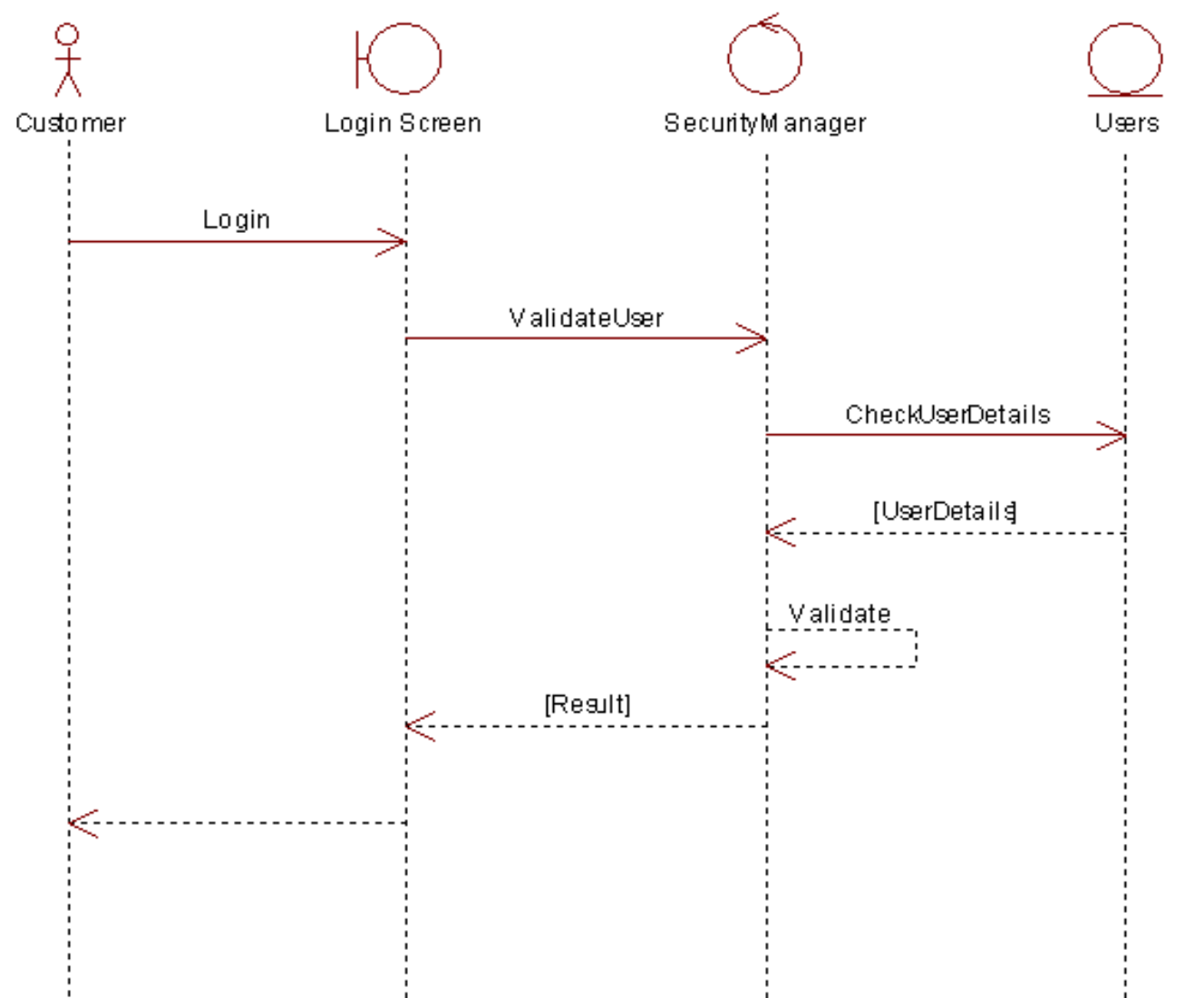
► Finally



Contoh 1

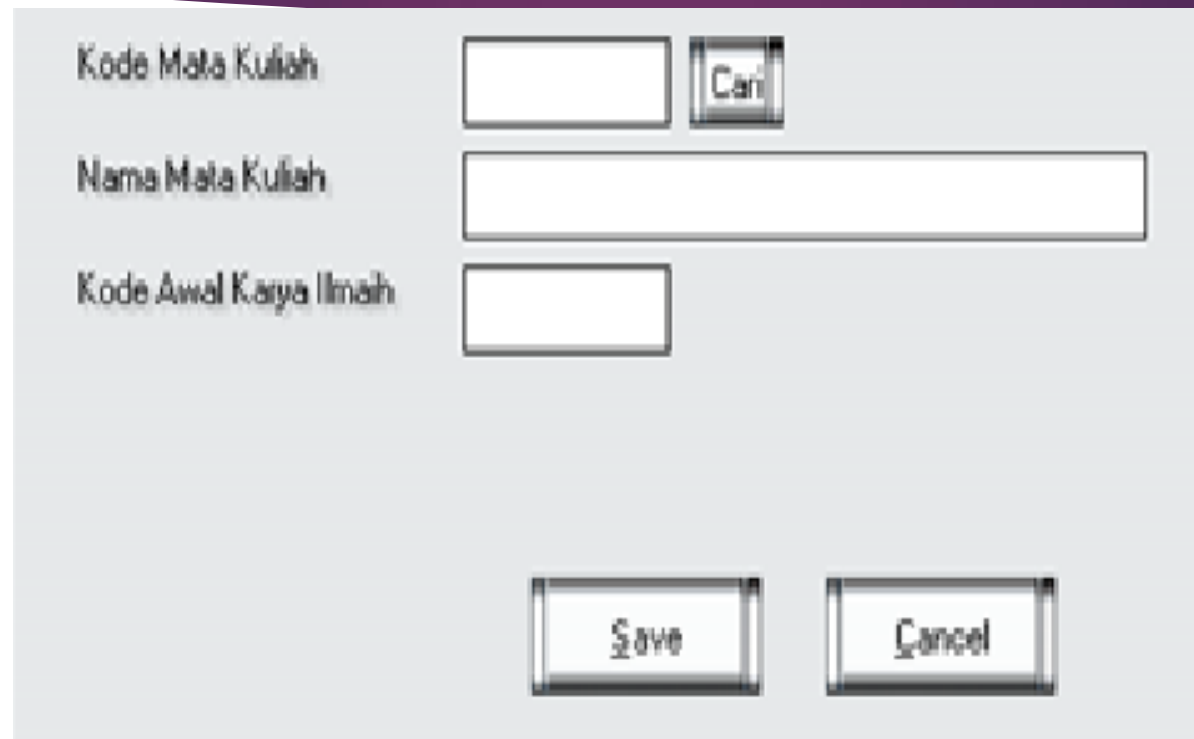


Contoh 2



Contoh 3

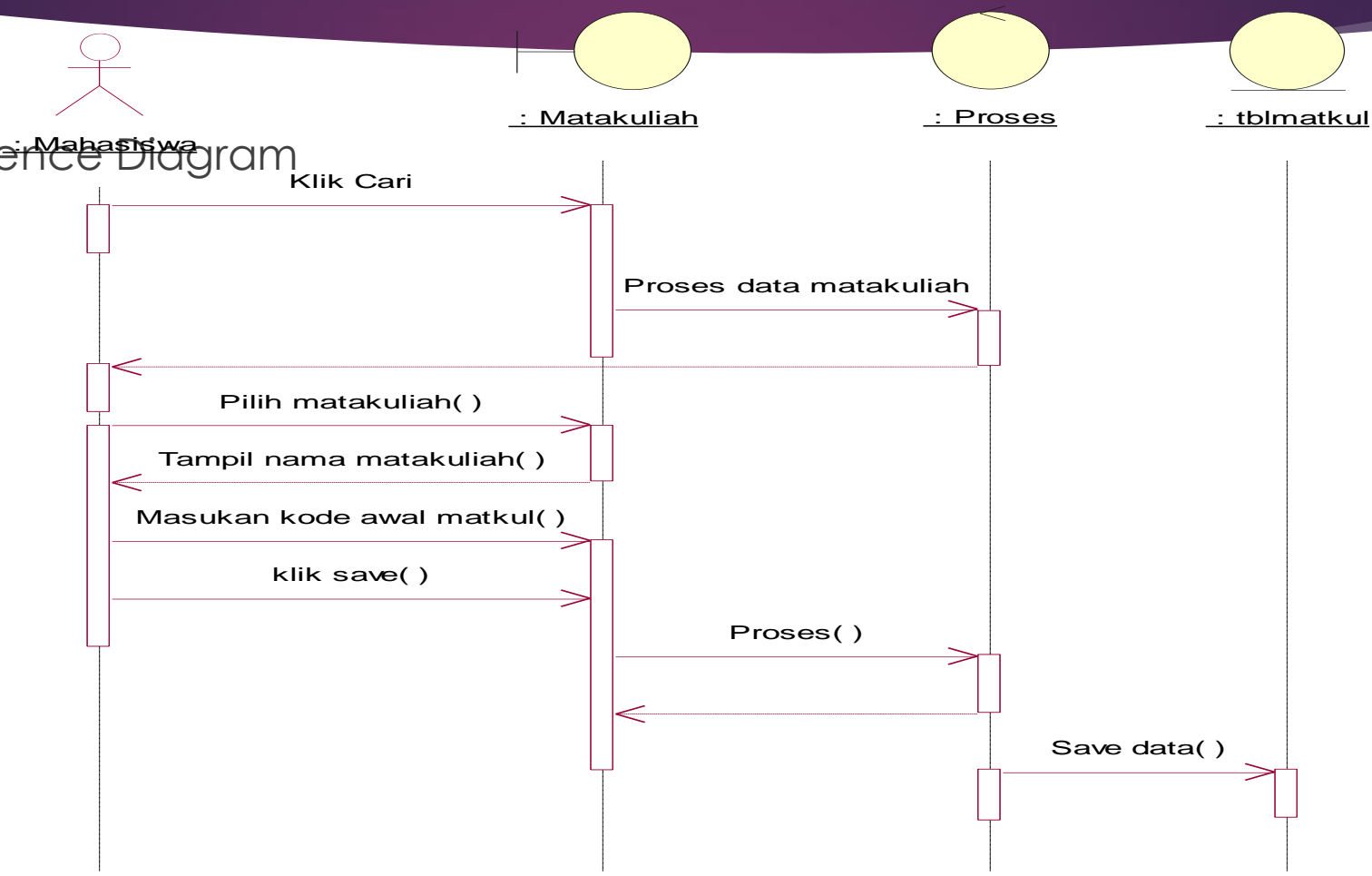
► Form

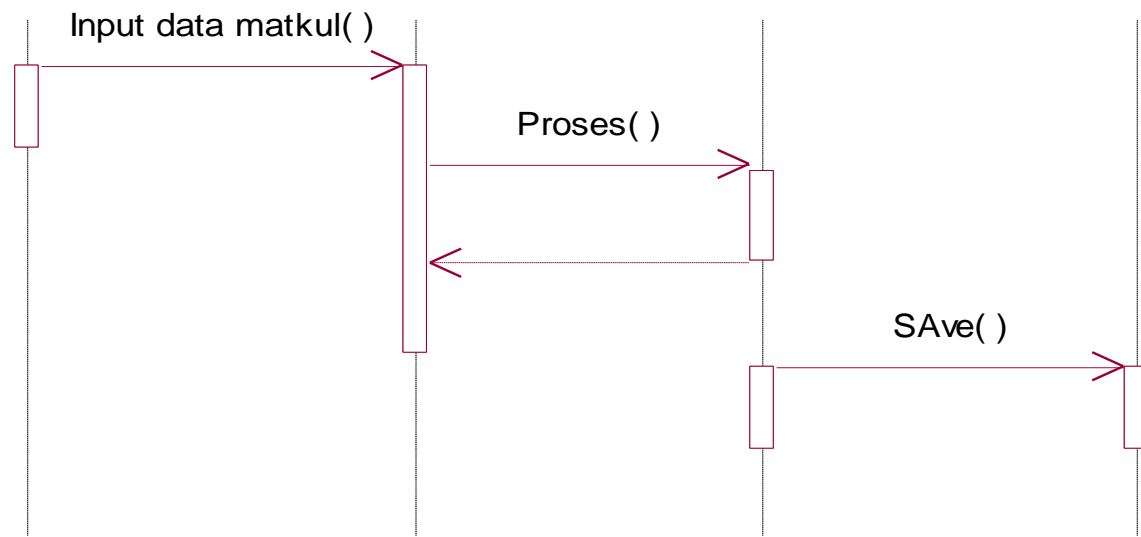
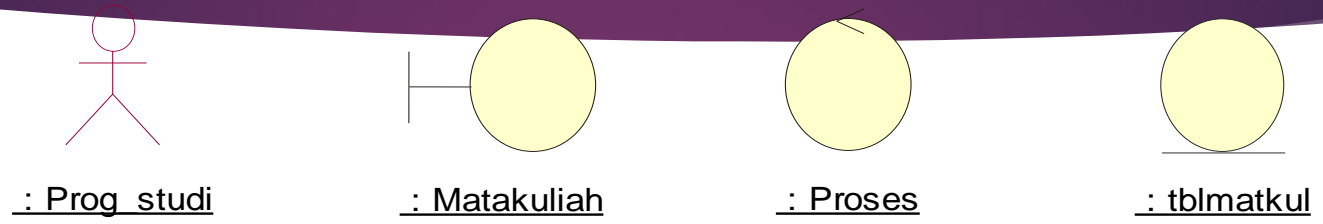


The image shows a web form with the following elements:

- Kode Mata Kuliah:** A text input field followed by a "Cari" button.
- Nama Mata Kuliah:** A wide text input field.
- Kode Awal Karya Ilmiah:** A text input field.
- Buttons:** "Save" and "Cancel" buttons located at the bottom of the form.

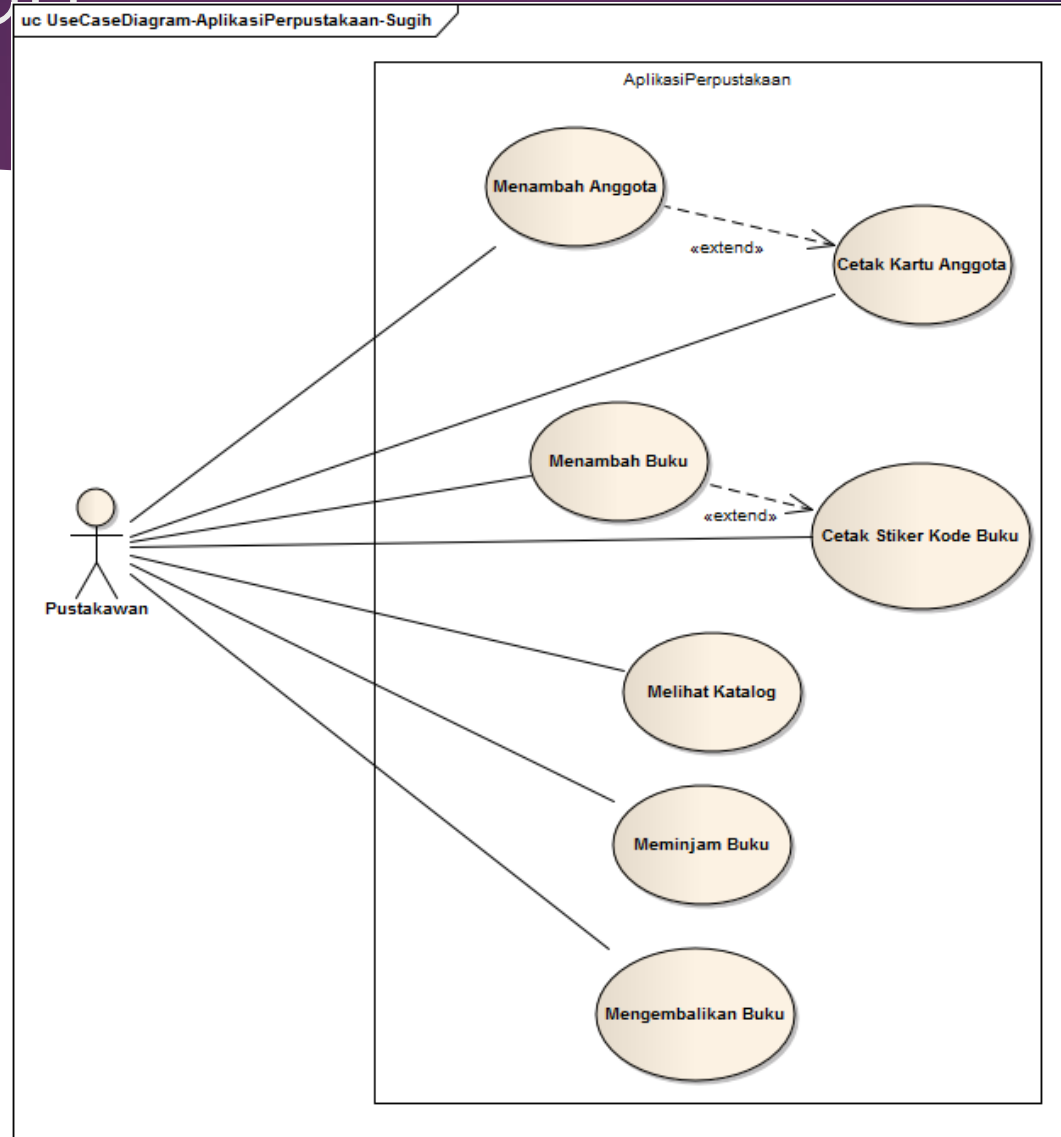
► Sequence Diagram






Perpustakaan

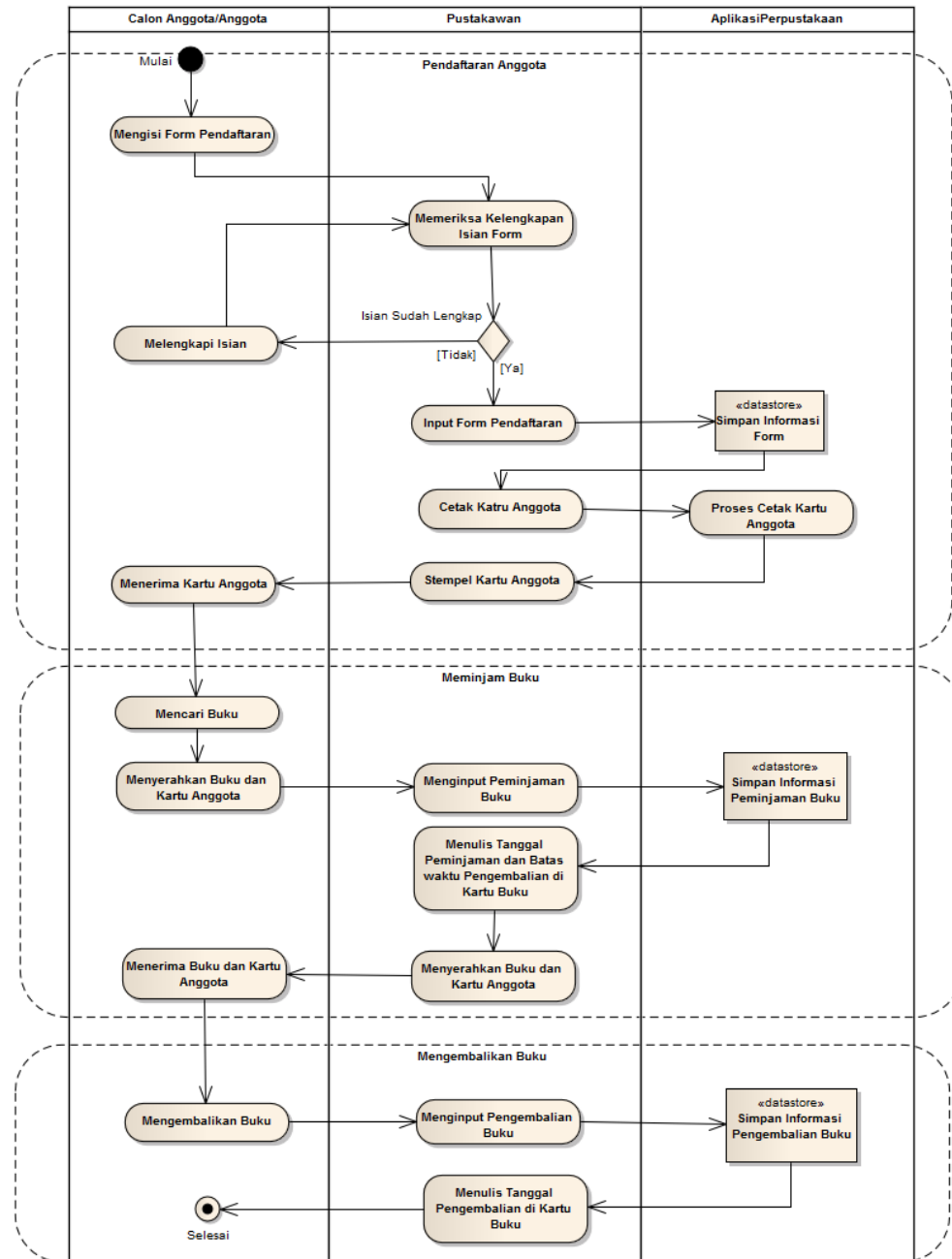
uc UseCaseDiagram-AplikasiPerpustakaan-Sugih




- 
- ▶ Gambar di atas menggambarkan Aplikasi mempunyai satu Actor/user yaitu Pustakawan dan 7 use case. Hal ini menjelaskan bahwa dalam aplikasi, pustakawan bisa Menambah Anggota, Mencetak Kartu Anggota, Menambah Buku, Mencetak Stiker Kode Buku, Melihat Katalog, Meminjam Buku, dan Mengembalikan Buku.
 - ▶ Mungkin ada kebingungan, mengapa yang meminjam dan mengembalikan buku adalah Pustakawan, bukan anggota perpustakaan.
 - ▶ Kalau kita lihat Business Process atau Activity Diagram , terlihat bahwa yang berinteraksi langsung dengan aplikasi adalah Pustakawan, bukan anggota. Anggota meminjam

Activity Diagram

act ActivityDiagram-AplikasiPerpustakaan-Sugih

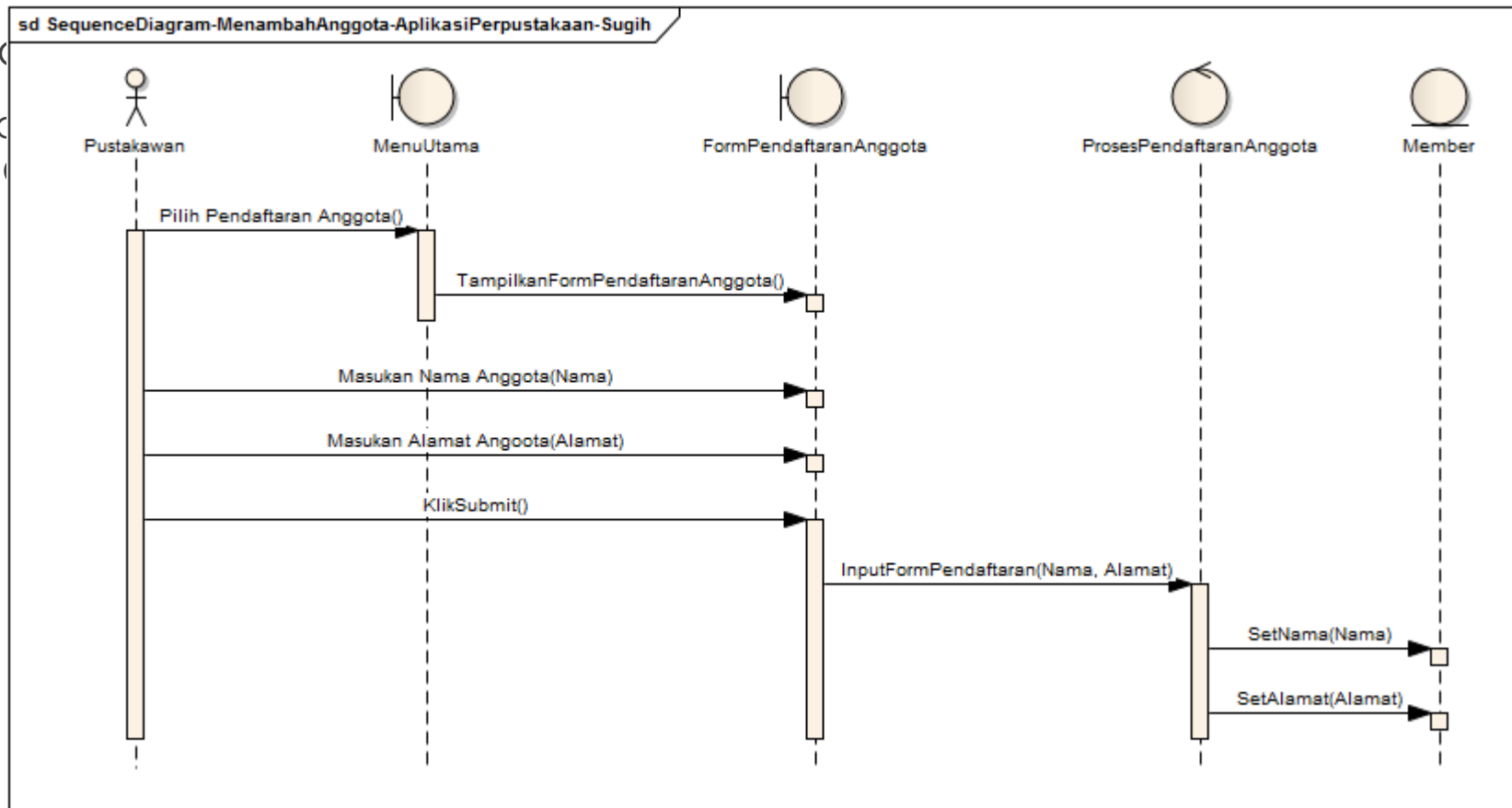


- 
- ▶ Diagram di atas menggambarkan 3 Activity utama di dalam perpustakaan, yaitu:
 - ▶ Menambah anggota/member perpustakaan.
 - ▶ Anggota meminjam buku.
 - ▶ Anggota mengembalikan buku.
 - ▶ Walaupun mungkin masih banyak activity-activity lain yang terkait dengan perpustakaan tetapi bukan merupakan business process yang utama dari perpustakaan.

Sequence Diagram

- ▶ Sequence Diagram menggambarkan urutan interaksi antar objek ketika satu Use Case dieksekusi/dilakukan, sehingga Sequence Diagram (sebaiknya) sebanyak Use Case nya.
- ▶ Dalam membuat Sequence Diagram ini memakai metode programming MVC (Model-View-Controller) atau dalam istilah lain Model=Entity, View=Boundary, dan Controller=Control.
- ▶ View/Boundary adalah class yang berinteraksi langsung dengan Actor. Controller/Control adalah class interaksi perantara antara View/Boundary dan Model/Entity. sedang Model/Entity adalah class yang menyimpan data.

- ▶ Men...
- ▶ Ada c...
- class



► Men

