

# *Suplemen Materi*

## *Use Case Diagram*



**Dr. Sutedi, S.Kom., M.T.I**

# Use Case Modeling

- ❖ Salah satu tantangan tim pengembangan *system* adalah bagaimana memperoleh persyaratan *system* yang benar dan diperlukan oleh para *stakeholder* serta menetapkannya dalam sebuah cara yang dapat dipahami para *stakeholder* agar persyaratan-persyaratan itu dapat diverifikasi dan divalidasi.
- ❖ Proses pengembangan *system* yang didasarkan pada pemahaman akan kebutuhan *stakeholder* dan alasan mengapa *system* itu harus dikembangkan adalah sebuah konsep yang dikenal dengan istilah “*user-centered development*”.



# Use Case Modeling

- ❖ Fokusnya pada bagaimana *system* akan digunakan dan bukan pada bagaimana *system* dibangun.
- ❖ *Use Case modeling* adalah sebuah pendekatan yang memfasilitasi pengembangan berpusat pada pengguna.
- ❖ *Use Case Diagram* adalah pemodelan untuk menggambarkan *behavior system* yang akan dibuat.



# Use Case Modeling

- ❖ *Use case diagram* menggambarkan interaksi antara satu atau lebih *actor* dengan *system* yang akan dibuat.
- ❖ Secara sederhana, *use case diagram* digunakan untuk memahami fungsi apa saja yang ada di dalam sebuah *system* dan siapa saja yang dapat menggunakan fungsi-fungsi tersebut.
- ❖ *Use case diagram* tidak menjelaskan secara detail tentang penggunaan tiap *use case*, namun hanya memberi gambaran singkat hubungan antara *use case*, *actor*, dan *system*.



# Use Case Diagram

❖ Ada dua hal utama pada *use case diagram* yaitu: pendefinisian *actor* dan *use case*.



➤ **Use case** adalah **fungsionalitas yang disediakan system** sebagai unit-unit yang saling bertukar pesan antar unit atau *actor*. Biasanya **use case** diberikan penamaan dengan menggunakan **kata kerja** di awal frase nama *use case*.

➤ **Actor** adalah **orang, proses, atau system lain yang berinteraksi** dengan *system* informasi yang akan dibuat, jadi **meskipun simbol dari actor ialah gambar orang, tapi actor belum tentu merupakan orang**. Biasanya penamaan aktor dinamakan menggunakan **kata benda** di awal frase nama *actor*.

# Use Case

## ❖ Simbol dan aturan pemodelan *Use Case*:

- *Use case* merupakan urutan langkah-langkah yang secara tindakan saling terkait (skenario), baik terotomatisasi maupun secara manual, untuk tujuan melengkapi satu tugas bisnis tunggal.
- *Use case* menggambarkan fungsi-fungsi *system* dari sudut pandang pengguna eksternal.

## ➤ Simbol *use case*:



*Use Case*

# Use Case

## ❖ Simbol dan aturan pemodelan *Use Case*:

- Sebuah *use case* harus **mendesripsikan sebuah pekerjaan** dimana pekerjaan tersebut akan **memberikan nilai** yang bermanfaat bagi *actor* (Kurt Bittner, Ian Spence. 2002).
- **Untuk menemukan *use cases***, mulailah dari sudut pandang *actor*, dengan pertanyaan berikut.
  - Informasi apa sajakah yang akan didapatkan *actor* dari *system* ?
  - Apakah ada kejadian dari *system* yang perlu diberitahukan ke *actor* ?

# Use Case

## ❖ Simbol dan aturan pemodelan *Use Case*:

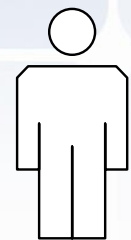
- Sedangkan **dari sudut pandang *system***, gunakan pertanyaan berikut.
  - Apakah ada informasi yang perlu disimpan / diambil dari *system* ?
  - Apakah ada informasi yang harus dimasukkan oleh *actor*?
- Pengertian *use case* yang disampaikan Kurt Bittner dan Ian Spence penting untuk diingat, karena dari hal tersebut akan menentukan agar sebuah *use case* tidak menjadi terlalu kecil. Karena *use case* yang terlalu kecil tidak akan memberikan nilai bagi *actor*.

# Actor

## ❖ Simbol dan aturan pemodelan *Use Case*:

➤ *Actor* adalah segala sesuatu yang perlu berinteraksi dengan *system* untuk pertukaran informasi.

➤ Symbol *actor*:



*actor*

*atau*



*actor*



# Actor

❖ Simbol dan aturan pemodelan *Use Case*:

➤ Terdapat empat macam tipe *actor*.

## 1. *Primary business actor*

- **Stakeholder** yang berupa **pelaku utama dalam bisnis** yang **paling mendapatkan keuntungan** dari pelaksanaan *use-case* dengan menerima nilai yang terukur.
- **Contoh:** dalam *system* penggajian, karyawan merupakan *primary business actor* yang menerima keuntungan berupa nilai terukur dari *system* setiap periode penggajian.

# Actor

❖ Simbol dan aturan pemodelan *Use Case*:

➤ Terdapat empat macam tipe *actor*.

## 2. *Primary system actor*

- **Stakeholder** yang merupakan **pelaku utama dalam system** yang **secara langsung berhadapan dengan system** untuk menginisiasi / memicu kegiatan / *system*.
- **Contoh:** staf keuangan dalam *system* penggajian.
- Dalam beberapa kasus, terdapat kejadian bisnis yang diinisiasi oleh kalender atau waktu (*temporal event*).

# Actor

❖ Simbol dan aturan pemodelan *Use Case*:

➤ Terdapat empat macam tipe *actor*.

### 3. *External server actor*

- **Stakeholder** di luar *system* yang melayani kebutuhan pengguna *use case*.
- **Contoh:** aplikasi layanan bank dalam *system* penggajian.

### 4. *External receiving actor*

- **Stakeholder** di luar *system* dan bukan pelaku utama, tapi menerima nilai yang terukur (*output*) dari *use case*.
- **Contoh:** pimpinan merupakan *external system* (bukan pelaku utama *system*) yang menerima nilai terukur berupa laporan terkait dengan pelaksanaan penggajian tersebut.

# Actor

## ❖ Simbol dan aturan pemodelan *Use Case*:

- *Actor* adalah segala hal di luar *system* yang akan menggunakan *system* tersebut untuk melakukan sesuatu (Kurt Bittner, Ian Spence. 2002).
- Cara termudah untuk menemukan *actor* adalah dengan bertanya "Siapa yang akan menggunakan *system* ?"
- Namun tidak semua *actor* adalah manusia, *actor* juga dapat berupa *system* lain (yang berada di luar *system* yang akan dibuat), ciri *system* sebagai *actor* adalah sebagai berikut:
  - Jika *system* yang akan dibuat / dimodelkan bergantung pada *system* lain untuk melakukan sesuatu, maka *system* lain itu adalah *actor*.
  - Jika *system* yang akan dibuat / dimodelkan meminta (*request*) informasi dari *system* lain, maka *system* lain itu adalah *actor*.

# Actor

## ❖ Simbol dan aturan pemodelan *Use Case*:

- Untuk kasus *system* lain yang bertindak sebagai *actor*, dapat di ilustrasikan sebagai berikut.
  - Misalkan: sebuah *system* akademik baru dapat menampilkan nilai mahasiswa apabila pembayaran mahasiswa sudah lunas, artinya *system* akademik memerlukan informasi dari *system* pembayaran. Oleh karena itu maka saat kita akan memodelkan *use case diagram* *System* akademik, kita akan memasukkan *system* pembayaran sebagai *actor*.

# Association

## ❖ Simbol dan aturan pemodelan *Use Case*:

➤ *Association*/asosiasi adalah komunikasi antara *actor* dan *use case* yang berpartisipasi pada *use case diagram* atau *use case* yang memiliki interaksi dengan *actor*. Asosiasi merupakan simbol yang digunakan untuk menghubungkan *link* antar *element*.

➤ Simbol *Association*: \_\_\_\_\_



# Extend

## ❖ Simbol dan aturan pemodelan *Use Case*:

➤ *Extend* merupakan relasi *use case* tambahan ke sebuah *use case* dimana *use case* inti dapat berdiri sendiri meski tanpa *use case* tambahan tersebut.

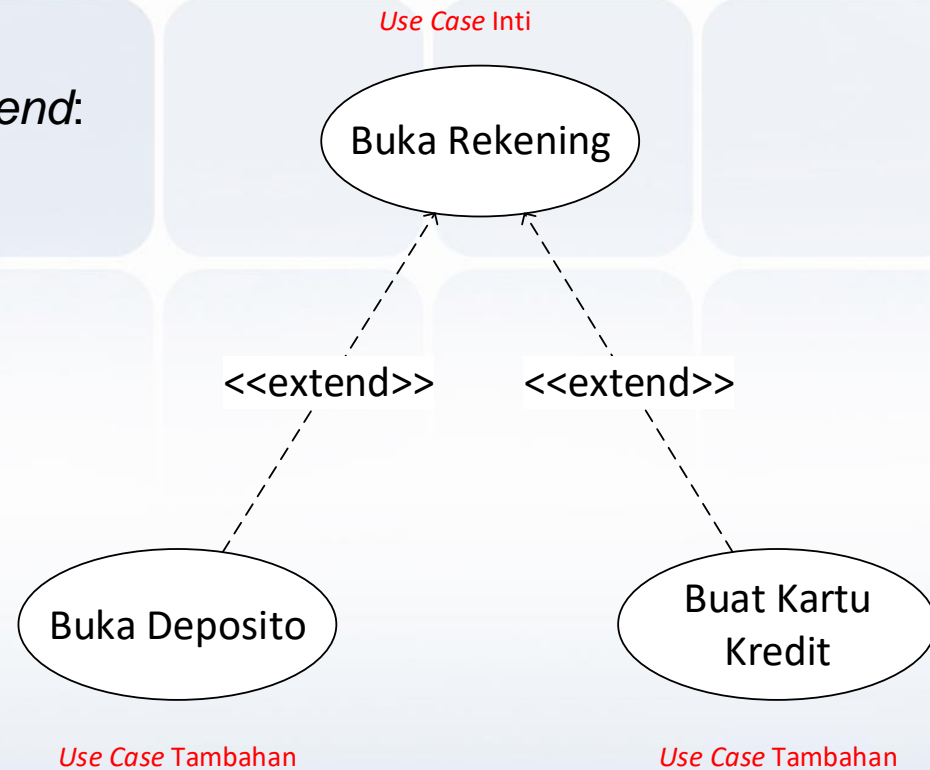
➤ Simbol *Extend* :                    ----<<extend>>--->

➤ Arah panah **menuju *use case* inti.**

# Extend

❖ Simbol dan aturan pemodelan *Use Case*:

➤ Contoh penggunaan *extend*:



# *Include*

## ❖ Simbol dan aturan pemodelan *Use Case*:

➤ *Include* merupakan relasi *use case* tambahan ke sebuah *use case* dimana *use case* inti membutuhkan *use case* tambahan untuk menjalankan fungsinya atau sebagai syarat dijalankan *use case* inti tersebut.

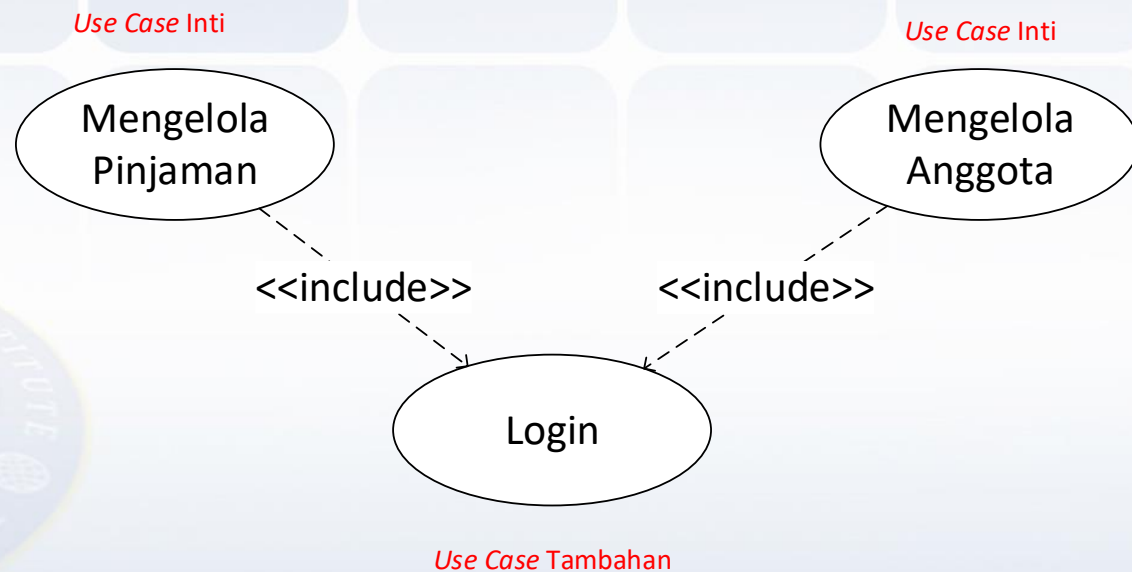
➤ Simbol *include*:      ---<<include>>---

➤ **Arah panah** *include* mengarah ke ***use case* tambahan** atau *use case* yang dipakai (dibutuhkan).

# Include

❖ Simbol dan aturan pemodelan *Use Case*:

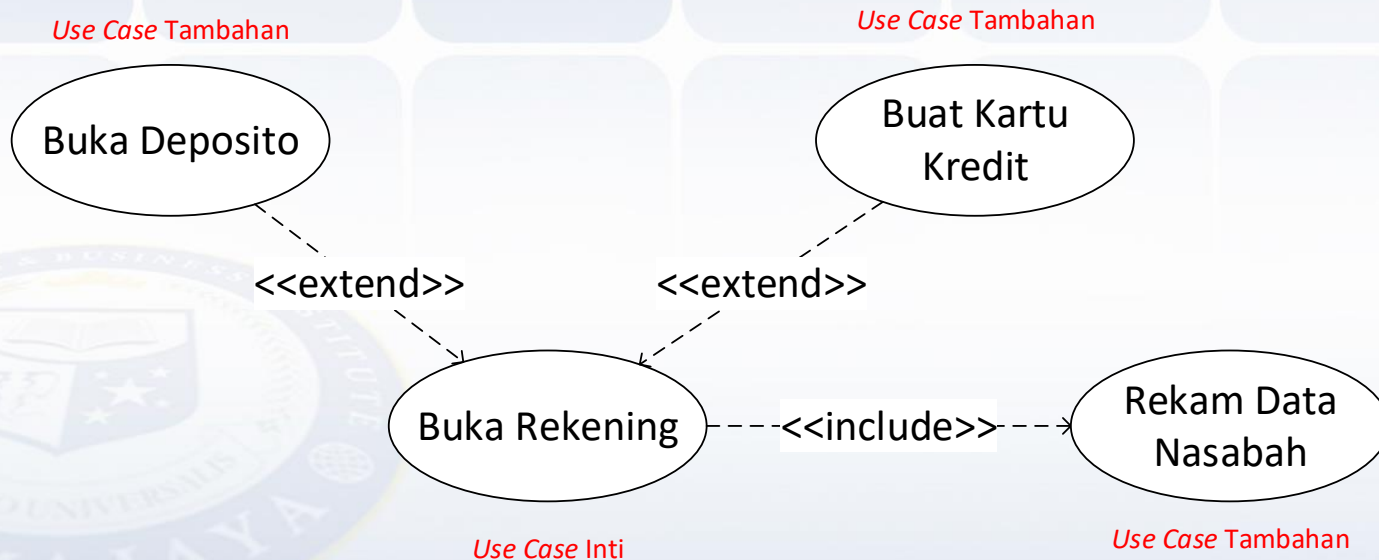
➤ Contoh penggunaan *include*:



# Extend dan Include


❖ Simbol dan aturan pemodelan *Use Case*:

➤ Contoh penggunaan *extend* dan *include*:



# Generalization

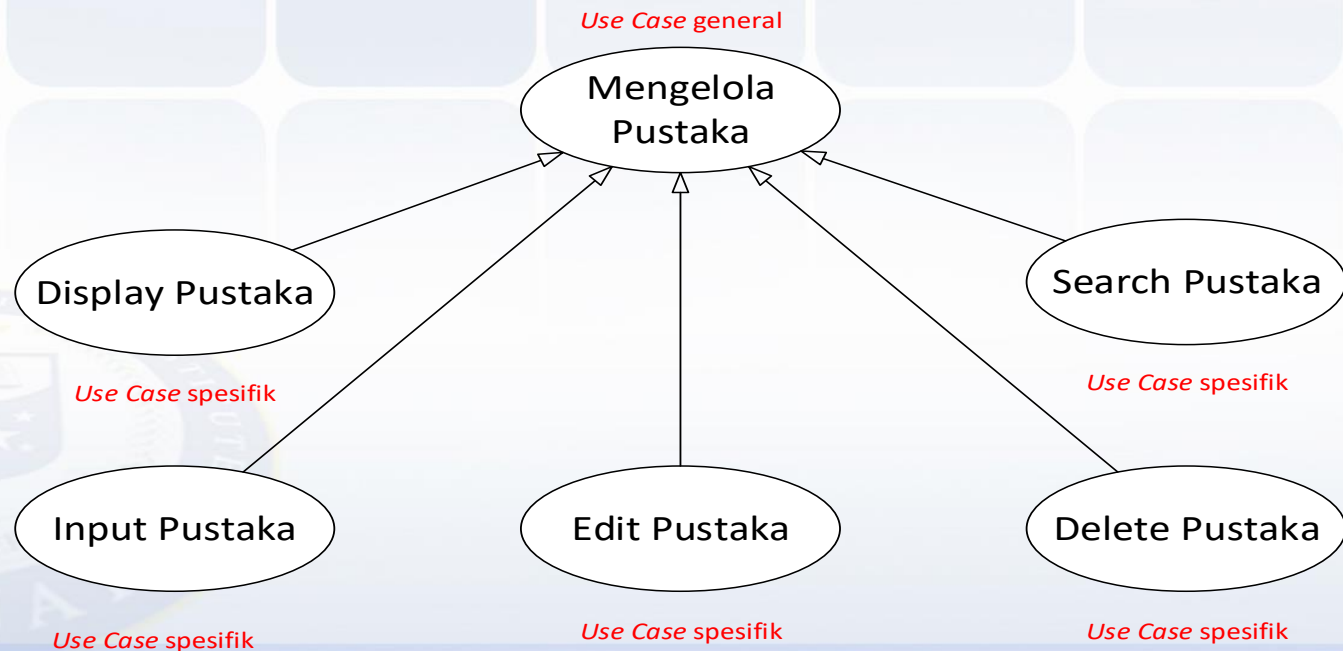
## ❖ Simbol dan aturan pemodelan *Use Case*:

- *Generalization* merupakan hubungan generalisasi dan spesialisasi (umum - khusus) antara dua buah *use case* dimana fungsi yang satu merupakan fungsi yang lebih umum dari lainnya.
- Simbol *generalization*: 
- **Arah panah *generalization* mengarah ke *use case* umumnya (umum).**

# Generalization

❖ Simbol dan aturan pemodelan *Use Case*:

➤ Contoh penggunaan *generalization*:



# Use Case Scenario

❖ Simbol dan aturan pemodelan *Use Case*:

- Setiap *use case diagram* harus dilengkapi dengan *scenario*, dimana *use case scenario* merupakan alur jalannya proses *use case* dari sisi *actor* dan *system*. Berikut adalah format dokumen *use case scenario*.

Nama Actor	Reaksi System
<i>Scenario Normal</i>	
<i>Scenario Alternatif</i>	

# Use Case Scenario

## ❖ Simbol dan aturan pemodelan *Use Case*:

- *Use Case Scenario* dibuat per *use case* terkecil, misal untuk generalisasi maka *scenario* yang dibuat adalah *use case* yang spesifik.
- **Scenario normal** adalah *scenario* bila **system berjalan normal** tanpa terjadi kesalahan atau *error*.
- Sedangkan **scenario alternatif** adalah *scenario* bila **system tidak berjalan normal** atau mengalami *error*.
- *Scenario* normal dan *scenario* alternatif **dapat berjumlah lebih dari satu**. Alur *scenario* inilah yang nantinya menjadi landasan pembuatan *sequence diagram*.

# Studi Kasus

❖ Nama Aplikasi: **System Informasi Manajemen Perpustakaan**

➤ **Deskripsi:**

- ✓ *System* informasi manajemen perpustakaan adalah sebuah *system* informasi untuk mengelola informasi yang diperlukan dalam sebuah perpustakaan yang meliputi pengelolaan pustaka, pengelolaan anggota, pengelolaan petugas dan pengelolaan peminjaman pustaka.
- ✓ Aturan perpustakaan yang harus dipenuhi pada *system* informasi manajemen perpustakaan yang akan dimodelkan adalah sebagai berikut:
  - Pustaka dapat memiliki lebih dari satu pengarang.
  - Anggota dapat meminjam lebih dari satu buku (pustaka) dalam satu waktu (waktu yang bersamaan).
  - Anggota dapat memiliki lebih dari satu nomor telepon.

# Studi Kasus

❖ Nama Aplikasi: **System Informasi Manajemen Perpustakaan**

➤ **Deskripsi:**

✓ Lanjutan Aturan perpustakaan:

- Anggota dapat mengembalikan pustaka yang dipinjam tidak dalam waktu yang bersamaan, meskipun pustaka-pustaka tersebut dipinjam pada waktu yang bersamaan.
- Pengunjung yang bukan anggota diperbolehkan mencari data pustaka yang ingin dibacanya.
- Pengunjung yang bukan anggota tidak diperbolehkan meminjam pustaka.
- Proses pendaftaran pustaka, anggota, dan peminjaman dilakukan oleh petugas perpustakaan.
- Anggota dan pengunjung dapat melakukan pencarian pustaka.
- Satu pustaka akan disimpan sebagai satu data dengan id yang unik.

# Studi Kasus

❖ Nama Aplikasi: **System Informasi Manajemen Perpustakaan**

➤ **Deskripsi:**

- ✓ *System Informasi Manajemen Perpustakaan* yang akan dimodelkan memiliki fungsi-fungsi sebagai berikut.
  - Validasi Petugas menggunakan login.
  - Mengelola data Pustaka melingkupi kegiatan berikut.
    - Memasukkan data pustaka baru
    - Mengubah data pustaka
    - Menghapus data pustaka
    - Mencari data pustaka
    - Melihat data pustaka
  - Mengelola data Peminjaman melingkupi kegiatan berikut.
    - Memasukkan data peminjaman
    - Mengubah data peminjaman
    - Mencari data peminjaman
    - Melihat data peminjaman

# Studi Kasus

❖ Nama Aplikasi: **System Informasi Manajemen Perpustakaan**

➤ **Deskripsi:**

✓ Lanjutan fungsi-fungsi yang dimiliki *system*.

- Mengelola data Anggota melingkupi kegiatan berikut.
  - Memasukkan data anggota baru
  - Mengubah data anggota
  - Menghapus data anggota
  - Mencari data anggota
  - Melihat data anggota
- Mengelola data petugas melingkupi kegiatan berikut.
  - Memasukkan data petugas baru
  - Mengubah data petugas
  - Menghapus data petugas
  - Mencari data petugas
  - Melihat data petugas

# Penyelesaian Studi Kasus

❖ Terdapat 4 tahapan yang harus dilalui.

1. **Pendefinisian Actor**, yaitu sebagai berikut.

No	Actor	Deskripsi
1	Petugas Perpustakaan	Petugas perpustakaan adalah orang yang bertugas dan memiliki hak akses untuk melakukan operasi pengelolaan data pustaka, anggota, dan proses peminjaman pustaka.
2	Anggota / Pengunjung Perpustakaan	Anggota adalah orang yang diperbolehkan meminjam pustaka sesuai dengan hak aksesnya, sedangkan pengunjung hanya memiliki hak akses melihat pustaka dan membaca di perpustakaan tanpa memiliki hak untuk meminjam pustaka.

# Penyelesaian Studi Kasus

❖ Terdapat 4 tahapan yang harus dilalui.

**2. Pendefinisian Use Case**, yaitu sebagai berikut.

No	Use Case	Deskripsi
1	<i>Login</i>	Merupakan proses untuk melakukan <i>login</i> petugas perpustakaan
2	Mengelola Pustaka	Mengelola pustaka merupakan proses pengelolaan data pustaka yang meliputi: memasukkan pustaka, melihat pustaka, mengubah pustaka, menghapus pustaka dan mencari pustaka.
3	Memasukkan Pustaka	Merupakan proses memasukkan data pustaka ke dalam basis data

# Penyelesaian Studi Kasus

❖ Terdapat 4 tahapan yang harus dilalui.

## 2. Pendefinisian Use Case (lanjutan).

4	Melihat Pustaka	Merupakan proses menampilkan data pustaka yang ada di dalam basis data
5	Mengubah Pustaka	Merupakan proses mengubah data pustaka yang ada di dalam basis data
6	Menghapus Pustaka	Merupakan proses menghapus data pustaka yang ada di dalam basis data
7	Mencari Pustaka	Merupakan proses mencari data pustaka yang ada di dalam basis data
8	Mengelola Anggota	Mengelola Anggota merupakan proses pengelolaan data anggota yang meliputi memasukkan anggota, melihat anggota, mengubah anggota, menghapus anggota dan mencari anggota.
9	Memasukkan Anggota	Merupakan proses memasukkan data anggota ke dalam basis data
10	Melihat Anggota	Merupakan proses menampilkan data anggota yang ada di dalam basis data

# Penyelesaian Studi Kasus

❖ Terdapat 4 tahapan yang harus dilalui.

## 2. Pendefinisian *Use Case* (lanjutan).

11	Mengubah Anggota	Merupakan proses mengubah data anggota yang ada di dalam basis data
12	Menghapus Anggota	Merupakan proses menghapus data anggota yang ada di dalam basis data
13	Mencari Anggota	Merupakan proses mencari data anggota yang ada di dalam basis data
14	Mengelola Peminjaman	Mengelola Peminjaman merupakan proses pengelolaan data peminjaman yang meliputi memasukkan peminjaman, melihat peminjaman, mengubah peminjaman, menghapus peminjaman dan mencari peminjaman.
15	Memasukkan Peminjaman	Merupakan proses memasukkan data peminjaman ketika ada anggota yang meminjam pustaka

# Penyelesaian Studi Kasus

❖ Terdapat 4 tahapan yang harus dilalui.

## 2. Pendefinisian Use Case (lanjutan).

16	Melihat Peminjaman	Merupakan proses menampilkan / melihat data peminjaman yang ada di dalam basis data
17	Mengubah Peminjaman	Merupakan proses mengubah data peminjaman yang dapat dilakukan untuk mengubah status peminjaman begitu pustaka dikembalikan
18	Menghapus Peminjaman	Merupakan proses menghapus data peminjaman jika ternyata peminjaman tidak jadi dilakukan atau data sudah terlalu banyak dan data sudah di <i>backup</i> terlebih dahulu
19	Mencari Peminjaman	Merupakan proses mencari data peminjaman yang ada di dalam basis data
20	Mengelola Petugas	Mengelola petugas merupakan proses pengelolaan data petugas yang meliputi memasukkan petugas, melihat petugas, mengubah petugas, menghapus petugas dan mencari petugas.

# Penyelesaian Studi Kasus

❖ Terdapat 4 tahapan yang harus dilalui.

## 2. Pendefinisian *Use Case* (lanjutan).

21	Memasukkan Petugas	Merupakan proses memasukkan data petugas ke dalam basis data
22	Melihat Petugas	Merupakan proses menampilkan data petugas yang ada di dalam basis data
23	Mengubah Petugas	Merupakan proses mengubah data petugas yang ada di dalam basis data
24	Menghapus Petugas	Merupakan proses menghapus data petugas yang ada di dalam basis data
25	Mencari Petugas	Merupakan proses mencari data petugas yang ada di basis data

# Penyelesaian Studi Kasus

❖ Terdapat 4 tahapan yang harus dilalui.

## 3. Pembuatan *Use Case Scenario*.

- ✓ Berikut adalah contoh sebagian hasil pendefinisian beberapa *use case scenaio* (tidak disertakan semua) dari masing-masing *use case* yang telah didefinisikan sebelumnya.

**Nama *Use Case* :** *Login*

***Scenario*:**

Aksi Actor	Reaksi System
<i>Scenario Normal</i>	
1. Memasukkan <i>username</i> dan <i>password</i>	
	2. Memeriksa valid tidaknya data masukan dengan memeriksa ke tabel petugas
	3. Masuk ke aplikasi pengelolaan data perpustakaan
<i>Scenario Alternatif</i>	
1. Memasukkan <i>username</i> dan <i>password</i>	
	2. Memeriksa valid tidaknya data masukan dengan memeriksa ke tabel petugas
	3. Menampilkan pesan <i>login</i> tidak valid
4. Memasukkan <i>username</i> dan <i>password</i> yang valid	
	5. Memeriksa valid tidaknya data masukan dengan memeriksa ke tabel petugas
	6. Masuk ke aplikasi pengelolaan data perpustakaan

# Penyelesaian Studi Kasus

❖ Terdapat 4 tahapan yang harus dilalui.

## 3. Pembuatan *Use Case Scenario* (lanjutan).

Nama *Use Case* : Memasukkan Pustaka

*Scenario*:

Aksi Actor	Reaksi System
<i>Scenario Normal</i>	
	1. Memeriksa status <i>login</i>
2. Memasukkan data pustaka sesuai kolom yang ada	
	3. Memeriksa valid tidaknya data masukan
	4. Menyimpan data pustaka ke basis data
	5. Menampilkan pesan sukses disimpan
<i>Scenario Alternatif</i>	
	1. Memeriksa status <i>login</i>
2. Memasukkan data pustaka sesuai kolom yang ada	
	3. Memeriksa valid tidaknya data masukan
	4. Mengeluarkan pesan bahwa data masukan tidak valid
5. Memperbaiki data masukan yang tidak valid	
	6. Memeriksa valid tidaknya data masukan
	7. Menyimpan data pustaka ke basis data
	8. Menampilkan pesan sukses disimpan

# Penyelesaian Studi Kasus

❖ Terdapat 4 tahapan yang harus dilalui.

## 3. Pembuatan *Use Case Scenario* (lanjutan).

Nama *Use Case* : MengubahPustaka

*Scenario*:

Aksi Actor	Reaksi System
<i>Scenario Normal</i>	
	1. Memeriksa status <i>login</i>
2. Memasukkan kata kunci dan kategori pencarian	
	3. Mencari data pustaka yang akan diubah
	4. Menampilkan data pustaka yang dicari (belum detail, misal hanya judulnya saja dan tampil dalam bentuk <i>list</i> )
5. Memilih data pustaka yang akan diubah	
	6. Menampilkan data pustaka (detail sebuah data pustaka) dari pustaka yang akan diubah
7. Mengubah data pustaka	
	8. Memeriksa valid tidaknya data masukan
	9. Menyimpan data yang telah diubah ke basis data
	10. Menampilkan pesan bahwa data sukses disimpan

# Penyelesaian Studi Kasus

❖ Terdapat 4 tahapan yang harus dilalui.

## 3. Pembuatan *Use Case Scenario* (lanjutan).

Nama *Use Case* : MengubahPustaka

*Scenario*:

<i>Scenario</i> Alternatif	
	1. Memeriksa status <i>login</i>
2. Memasukkan kata kunci dan kategori pencarian	
	3. Mencari data pustaka yang akan diubah
	4. Menampilkan data pustaka yang dicari (belum detail, misal hanya judulnya saja dan tampil dalam bentuk <i>list</i> )
5. Memilih data pustaka yang akan diubah	
	6. Menampilkan data pustaka (detail sebuah data pustaka) dari pustaka yang akan diubah
7. Mengubah data pustaka	
	8. Memeriksa valid tidaknya data masukan
	9. Menampilkan pesan bahwa data masukan tidak valid
10. Memperbaiki data masukan yang diubah dan tidak valid	
	11. Memeriksa valid tidaknya data masukan
	12. Menyimpan data yang telah diubah ke basis data
	13. Menampilkan pesan bahwa data sukses disimpan

# Penyelesaian Studi Kasus

❖ Terdapat 4 tahapan yang harus dilalui.

## 3. Pembuatan *Use Case Scenario* (lanjutan).

Nama *Use Case* : Menghapus Pustaka

*Scenario*:

Aksi Actor	Reaksi System
<i>Scenario Normal</i>	
	1. Memeriksa status <i>login</i>
2. Memasukkan kata kunci dan kategori pencarian	
	3. Mencari data pustaka yang akan dihapus
	4. Menampilkan data pustaka yang dicari (belum detail, misal hanya judulnya saja dan tampil dalam bentuk <i>list</i> )
5. Memilih data pustaka yang akan dihapus	
	6. Menampilkan pesan konfirmasi apakah data akan benar-benar dihapus
7. Mengklik pilihan setuju data dihapus	
	8. Menghapus data pustaka dari basis data
	9. Menampilkan pesan bahwa data sukses dihapus

# Penyelesaian Studi Kasus

❖ Terdapat 4 tahapan yang harus dilalui.

## 3. Pembuatan *Use Case Scenario* (lanjutan).

Nama *Use Case* : Menghapus Pustaka

*Scenario*:

<i>Scenario</i> Alternatif	
	1. Memeriksa status <i>login</i>
2. Memasukkan kata kunci dan kategori pencarian	
	3. Mencari data pustaka yang akan dihapus
	4. Menampilkan data pustaka yang dicari (belum detail, misal hanya judulnya saja dan tampil dalam bentuk <i>list</i> )
5. Memilih data pustaka yang akan dihapus	
	6. Menampilkan pesan konfirmasi apakah data akan benar-benar dihapus
7. Mengklik pilihan tidak setuju data dihapus	
	8. Kembali ke <i>form</i> pencarian pustaka

# Penyelesaian Studi Kasus

❖ Terdapat 4 tahapan yang harus dilalui.

## 3. Pembuatan *Use Case Scenario* (lanjutan).

Nama *Use Case* : Mencari Pustaka

*Scenario*:

Aksi Actor	Reaksi System
<i>Scenario Normal</i>	
1. Memasukkan kata kunci dan kategori pencarian	
	2. Mencari data pustaka yang dicari
	3. Menampilkan data pustaka yang dicari (belum detail, misal hanya judulnya saja dan tampil dalam bentuk list)
4. Memilih data pustaka yang akan dicari	
	5. Menampilkan data pustaka (detail sebuah data pustaka) dari pustaka yang dipilih

# Penyelesaian Studi Kasus

❖ Terdapat 4 tahapan yang harus dilalui.

## 3. Pembuatan *Use Case Scenario* (lanjutan).

Nama *Use Case* : Mencari Pustaka

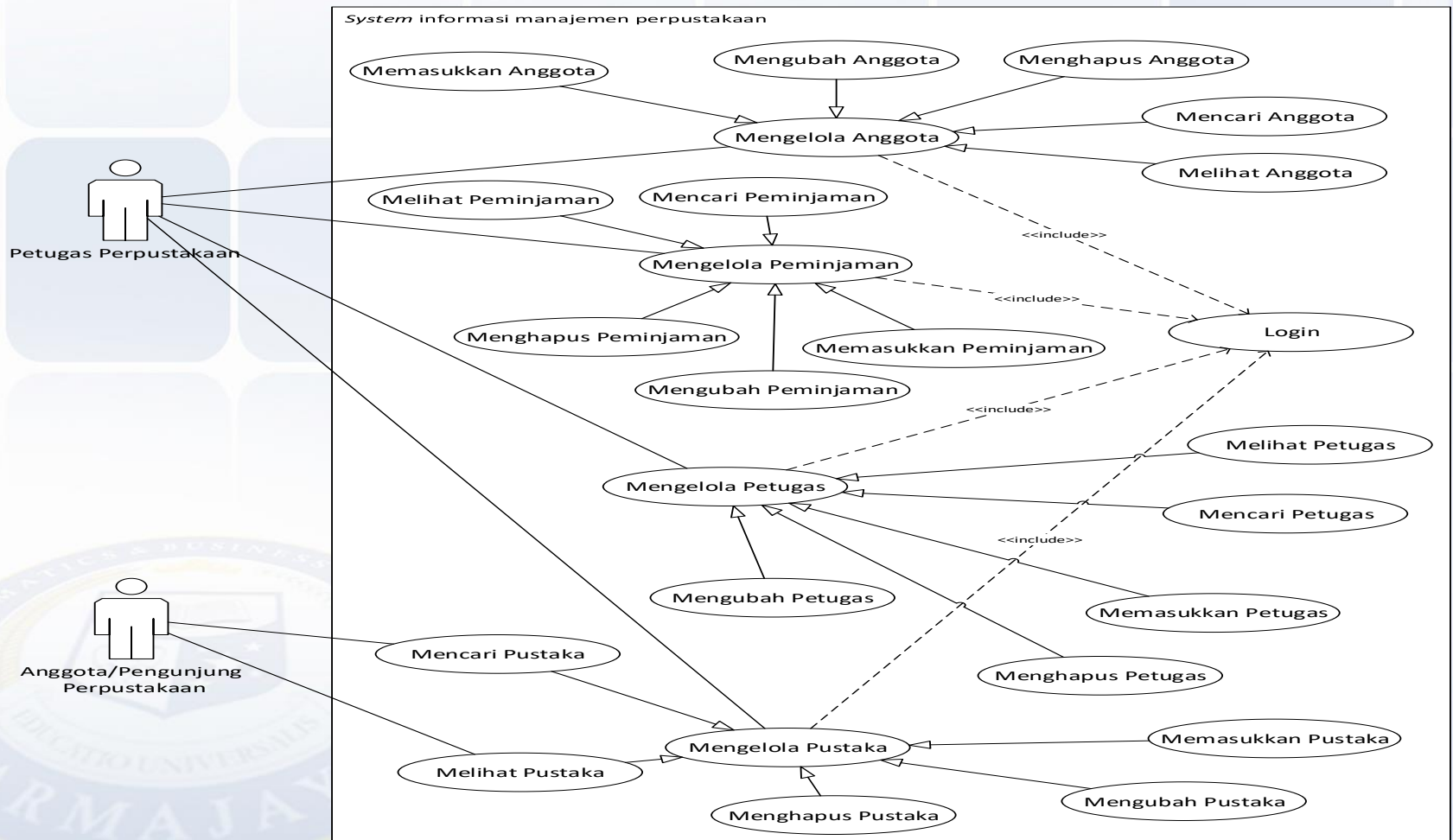
*Scenario*:

<i>Scenario</i> Alternatif	
1. Memasukkan kata kunci dan kategori pencarian	
	2. Mencari data pustaka yang dicari
	3. Menampilkan pesan data pustaka tidak ada
4. Memasukkan kata kunci dan kategori pencarian	
	5. Mencari data pustaka yang dicari
	6. Menampilkan data pustaka yang dicari (belum detail, misal hanya judulnya saja dan tampil dalam bentuk <i>list</i> )
7. Memilih data pustaka yang akan dicari	
	8. Menampilkan data pustaka (detail sebuah data pustaka) dari pustaka yang dipilih

# Penyelesaian Studi Kasus

❖ Terdapat 4 tahapan yang harus dilalui.

## 4. Menggambar *Use Case Diagram*.



THANK YOU

*Sampai jumpa di sesi berikutnya*

