

ANALISA & PERANCANGAN SISTEM INFORMASI

Sufajar Butsianto, M.Kom

MODUL 5

UML

**Unified Modelling
Language**



Tools :

Star UML, Rational Rose dll

TOOLS



1. Mahasiswa mengetahui tool untuk perancangan sistem informasi menggunakan Unified Modelling Language (UML)
2. Mahasiswa mengetahui teknik dalam membuat UML.
3. Mahasiswa mampu memahami dan membuat Use Case.

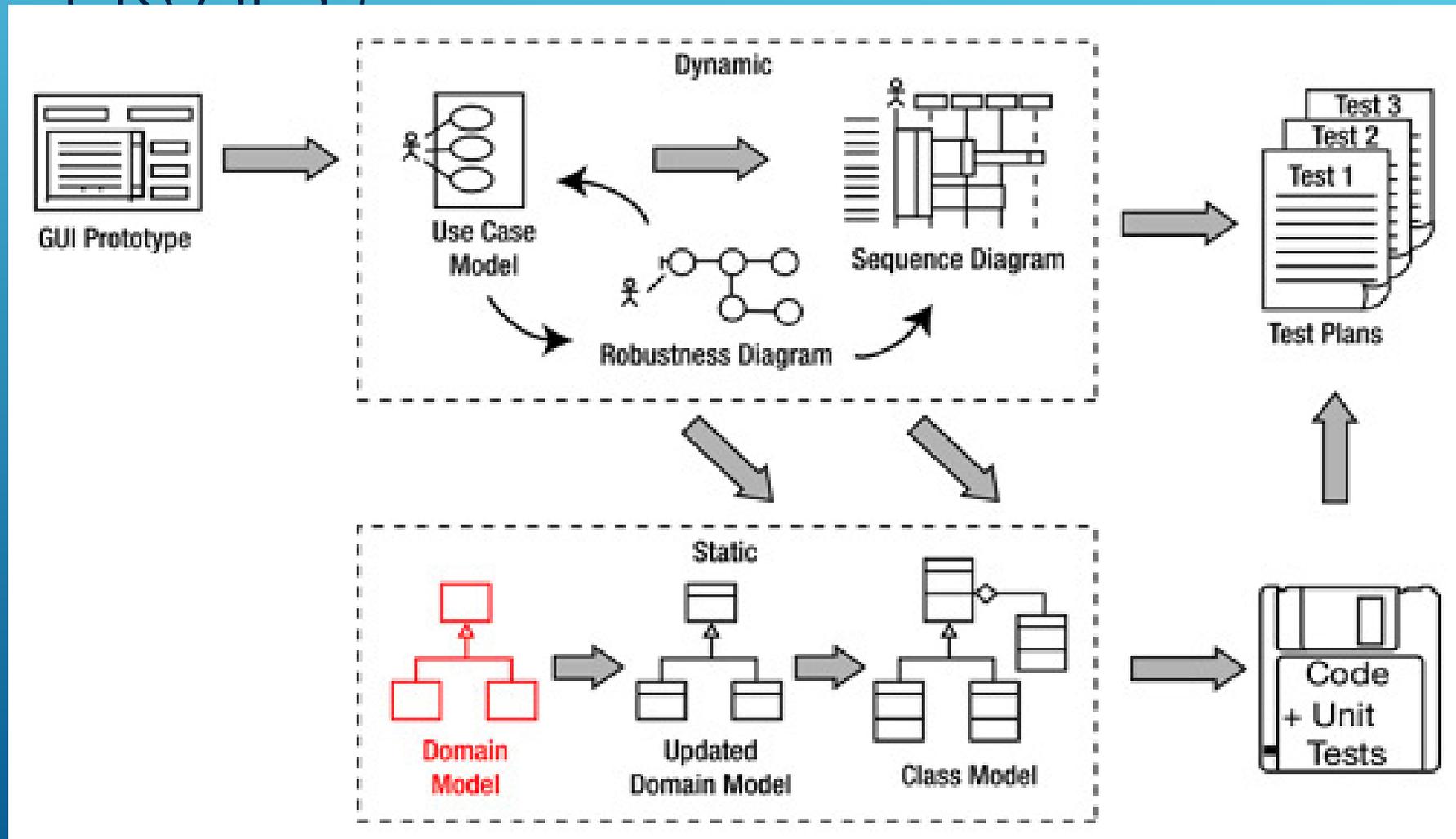
PENCAPAIAN
PEMBELAJARAN



PENGERTIAN UML

Unified Modelling Language (UML) adalah salah satu standar bahasa yang banyak digunakan di dunia industri untuk mendefinisikan kebutuhan (requirement), membuat analisis dan desain, serta menggambarkan arsitektur dalam pemrograman berorientasi objek (Rosa-Salahuddin, 2011:113).

GAMBARAN PROSES ANALISA & PERANCANGAN SISTEM (ICONIX PROSES)



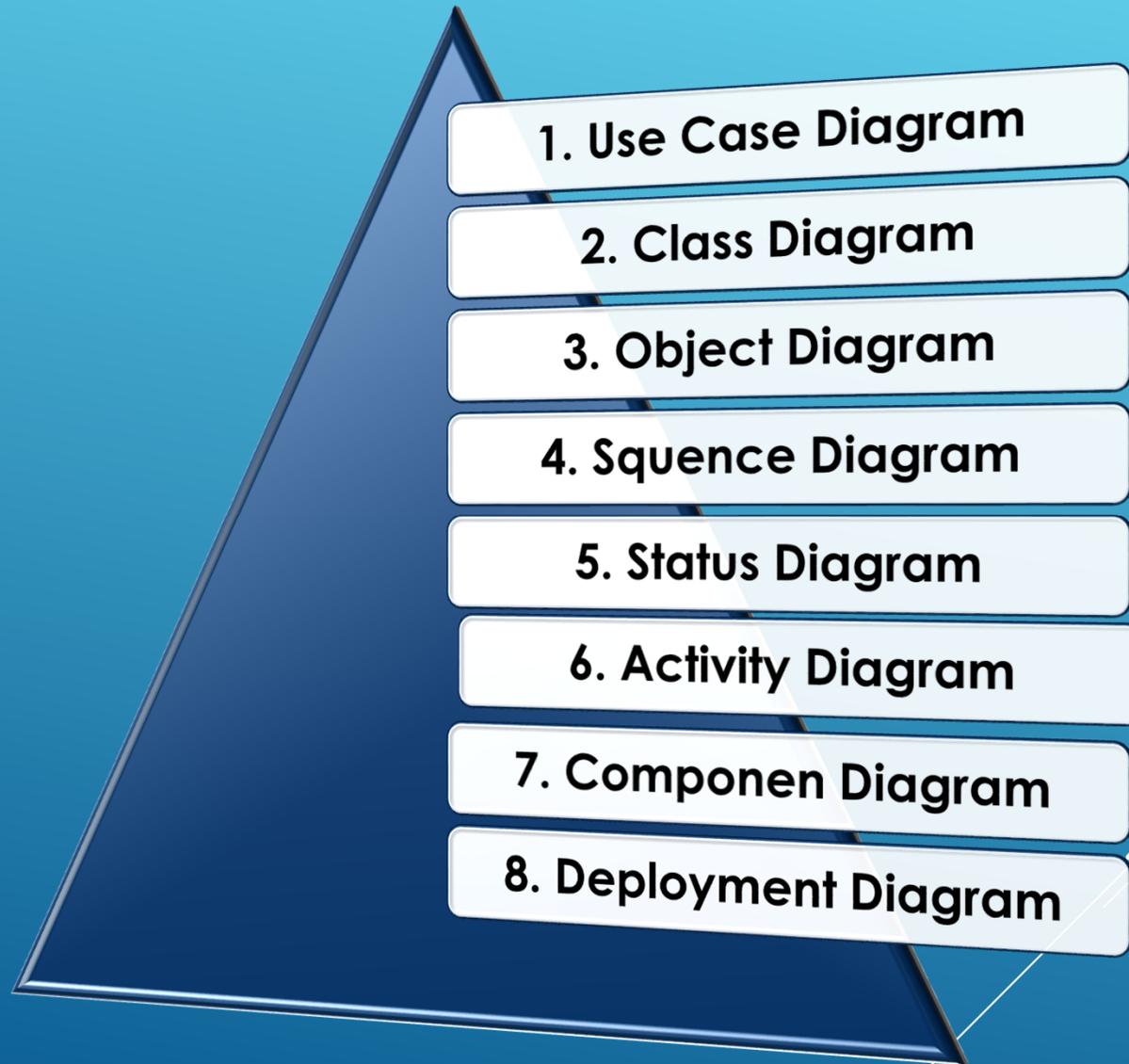
PROSES ANALISA & PERANCANGAN SI

Didalam proses analisa & perancangan sistem pada buku **Case Driven Object Modeling With UML: Theory and Practice** ialah sbb

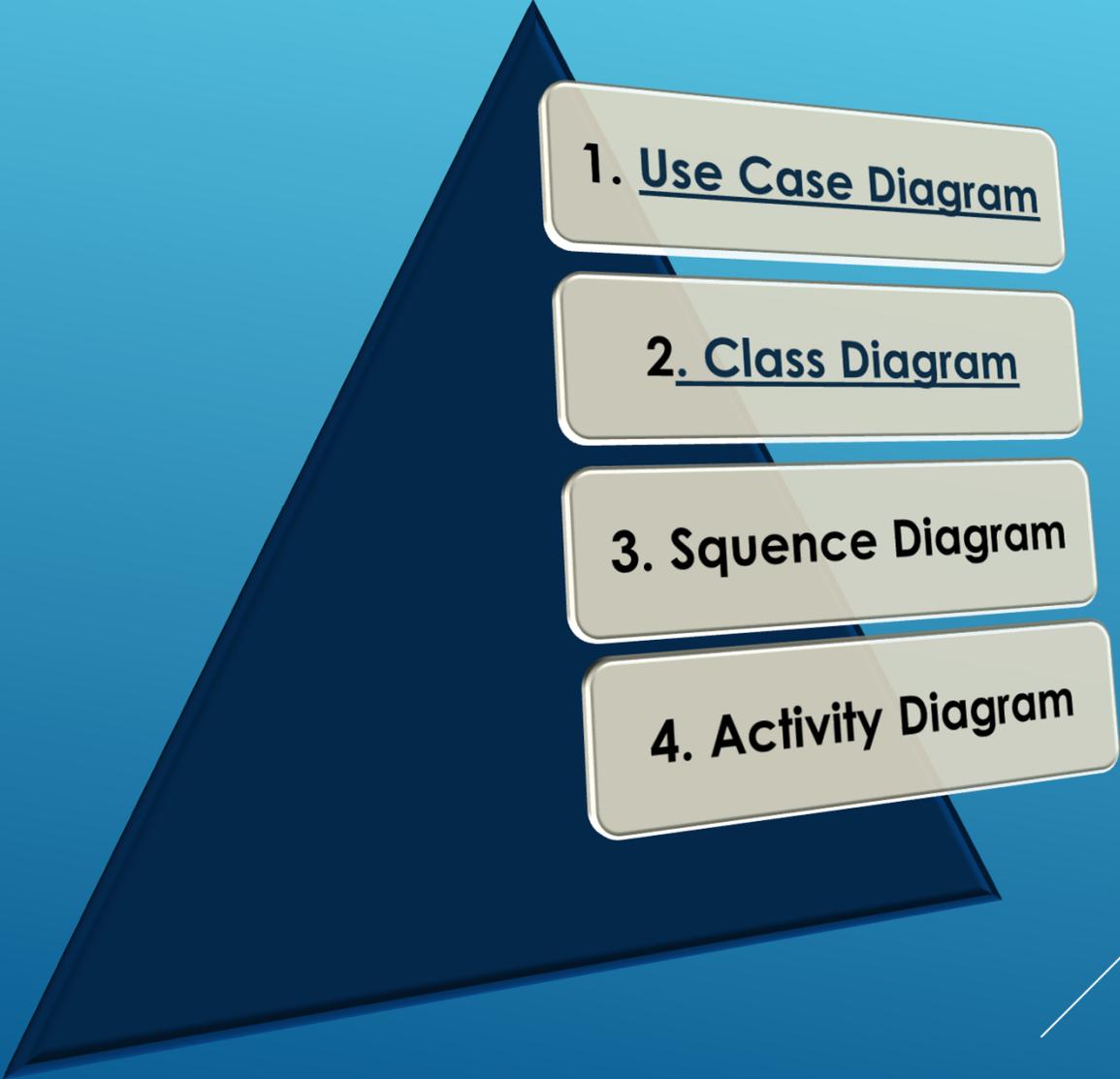
:

1. Membuat Functional Requirements
 2. Membuat Domain Model sederhana
 3. Membuat Use Case
 4. Requirements Review
 5. Melakukan Rebusness Analysis
 6. Preliminary Design Review
 7. Menentukan Technical Architecture
 8. Membuat Sequencen Diagram
 9. Critical Design Review
 10. Coding
- 

LANGKAH-LANGKAH MEMBUAT UML



PEMBAHASAN UML



1. Use Case Diagram

2. Class Diagram

3. Sequence Diagram

4. Activity Diagram





Use Case

PENGERTIAN USE CASE

- **Use case** class digunakan untuk memodelkan dan menyatakan unit fungsi/layanan yang disediakan oleh sistem (or bagian sistem: subsistem atau class) ke pemakai.
- **Use case** dapat dilingkupi dengan batasan sistem yang diberi label nama sistem.
- **Use case** adalah sesuatu yang menyediakan hasil yang dapat diukur ke pemakai atau sistem eksternal.

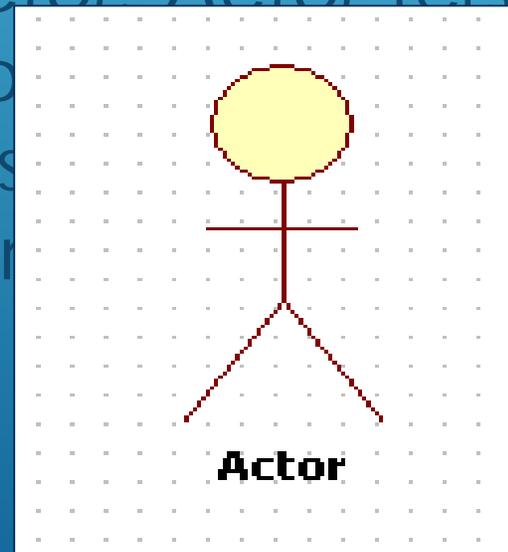
KARAKTERISTIK USE CASE

- ❑ **Use Case** adalah interaksi atau dialog antara sistem dan actor, termasuk pertukaran pesan dan tindakan yang dilakukan oleh sistem.
- ❑ **Use Case** diprakarsai oleh actor dan mungkin melibatkan peran actor lain.
- ❑ **Use Case** harus menyediakan nilai minimal kepada satu actor.
- ❑ **Use Case** bisa memiliki perluasan yang mendefinisikan tindakan khusus dalam interaksi atau use case lain mungkin disisipkan.
- ❑ **Use Case** class memiliki objek use case yang disebut skenario. Skenario menyatakan urutan pesan dan tindakan tunggal.

KOMPONEN PEMBENTUK USE CASE

1. Actor

Pada dasarnya *actor* bukanlah bagian dari *use case diagram*, namun untuk dapat terciptanya suatu *use case diagram* diperlukan beberapa *actor*. *Actor* tersebut mempresentasikan seseorang atau suatu (seperti perangkat, sistem yang



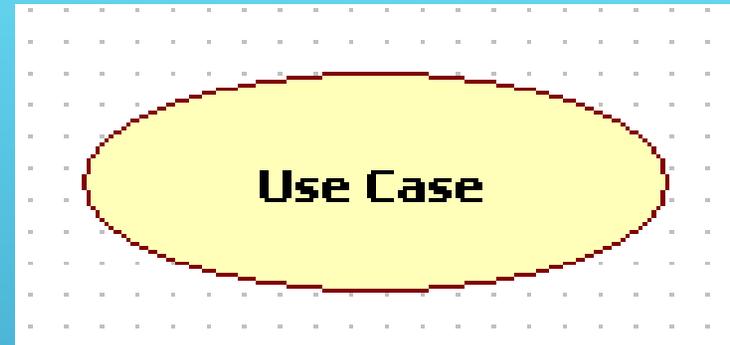
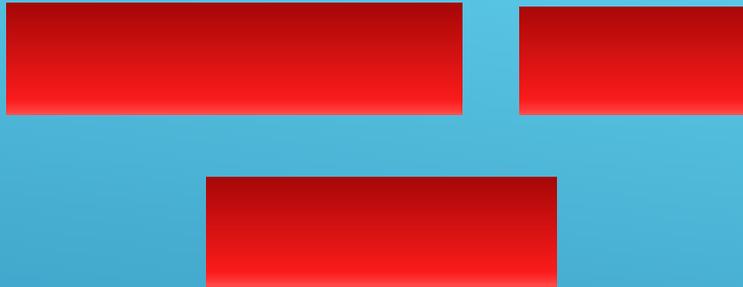
KOMPONEN PEMBENTUK USE CASE

2. Use Case

Use Case adalah gambaran fungsionalitas dari suatu sistem, sehingga *customer* atau pengguna sistem paham dan mengerti mengenai kegunaan sistem yang akan dibangun.

Cara menentukan Use Case dalam suatu sistem:

- a. Pola perilaku perangkat lunak aplikasi.
- b. Gambaran tugas dari sebuah *actor*.
- c. Sistem atau "benda" yang memberikan sesuatu yang bernilai kepada *actor*.
- d. Apa yang dikerjakan oleh suatu perangkat lunak (*bukan bagaimana cara mengerjakannya).



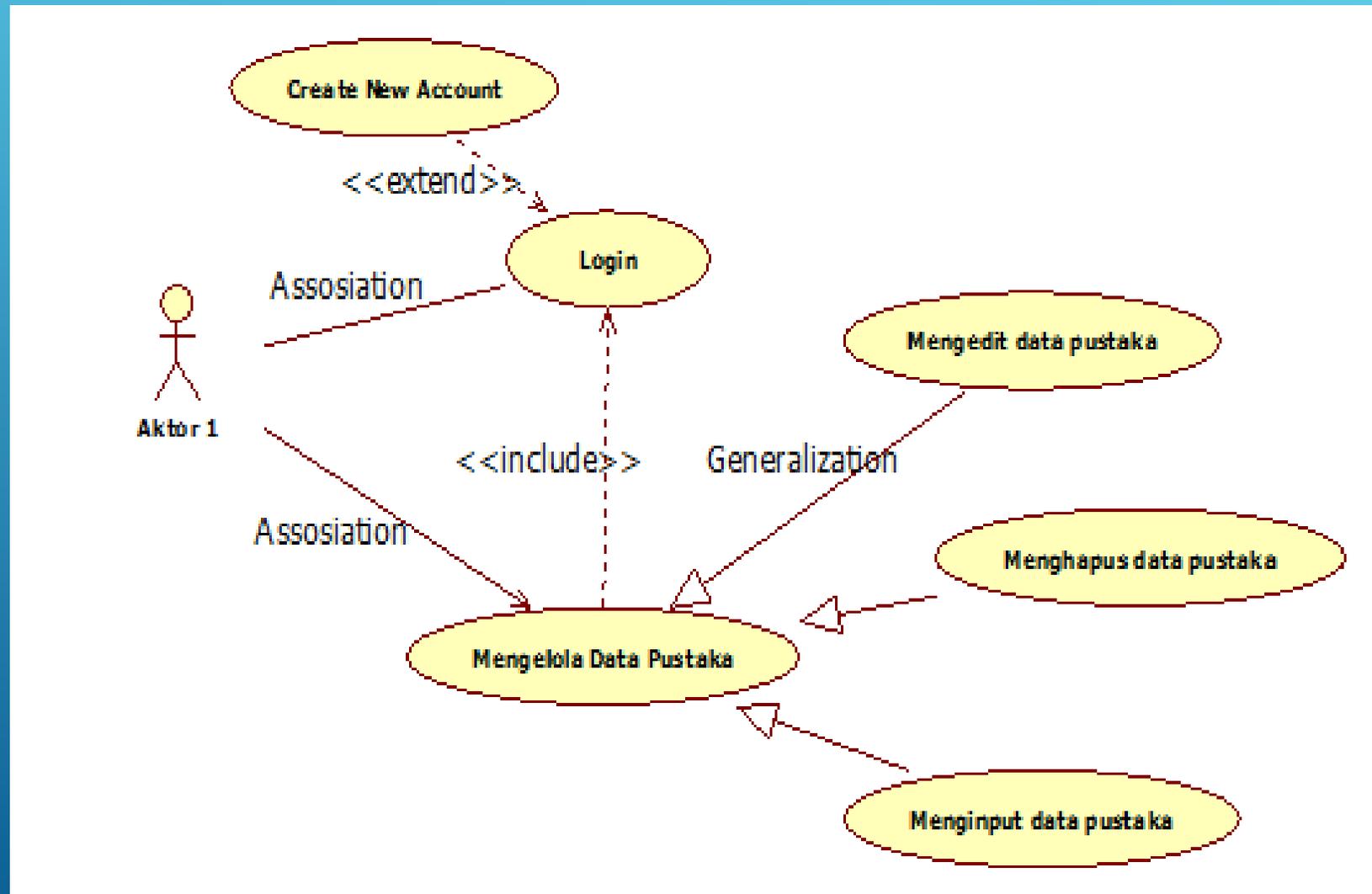
Relasi pada Use Case

1. *Association*, menghubungkan link antar element.
2. *Generalization*, disebut juga *inheritance* (pewarisan), sebuah elemen dapat merupakan spesialisasi dari elemen lainnya.
3. *Dependency*, sebuah element bergantung dalam beberapa cara ke element lainnya.
4. *Aggregation*, bentuk assosiation dimana sebuah elemen berisi elemen lainnya.

Tipe Relasi/*stereotype* yang mungkin terjadi pada Use Case Diagram :

1. <<include>> , yaitu kelakuan yang harus terpenuhi agar sebuah *event* dapat terjadi, dimana pada kondisi ini sebuah *use case* adalah bagian dari *use case* lainnya.
2. <<extends>>, kelakuan yang hanya berjalan di bawah kondisi tertentu seperti menggerakkan alarm.
3. <<communicates>>, mungkin ditambahkan untuk asosiasi yang menunjukkan asosiasinya adalah *communicates association* . Ini merupakan pilihan selama asosiasi hanya tipe *relationship* yang dibolehkan antara *actor* dan *use case*.

CONTOH USE CASE DIAGRAM

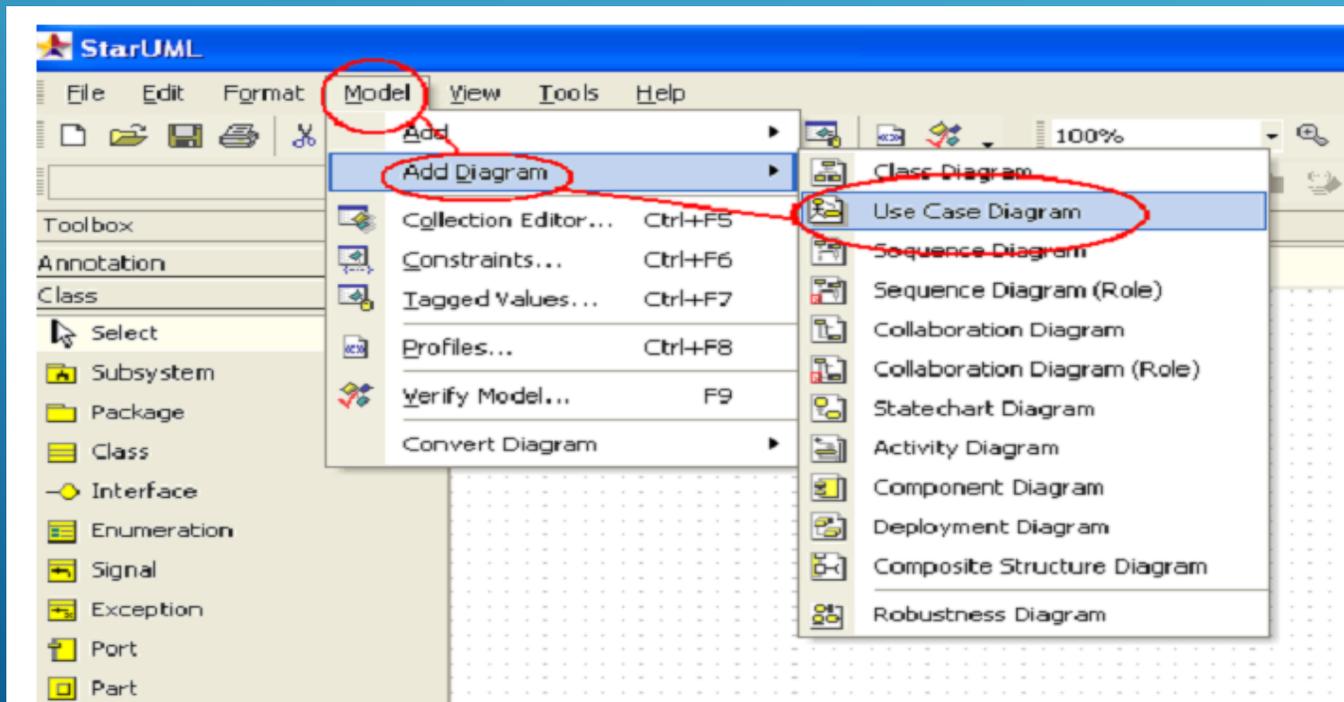


CONTOH MEMBUAT USE CASE DIAGRAM

Langkah-langkah membuat Use Case Diagram :

1.

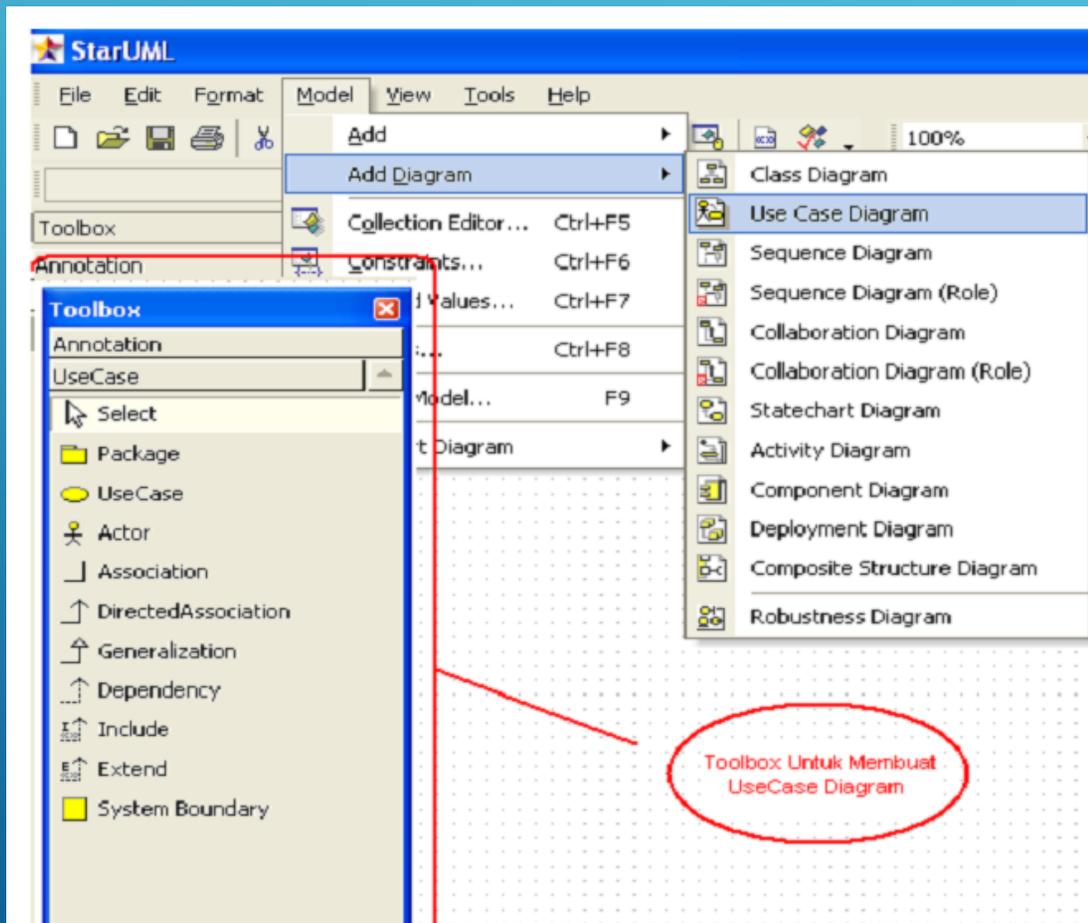
2.



... pada
... gram

CONTOH MEMBUAT USE CASE DIAGRAM

3.

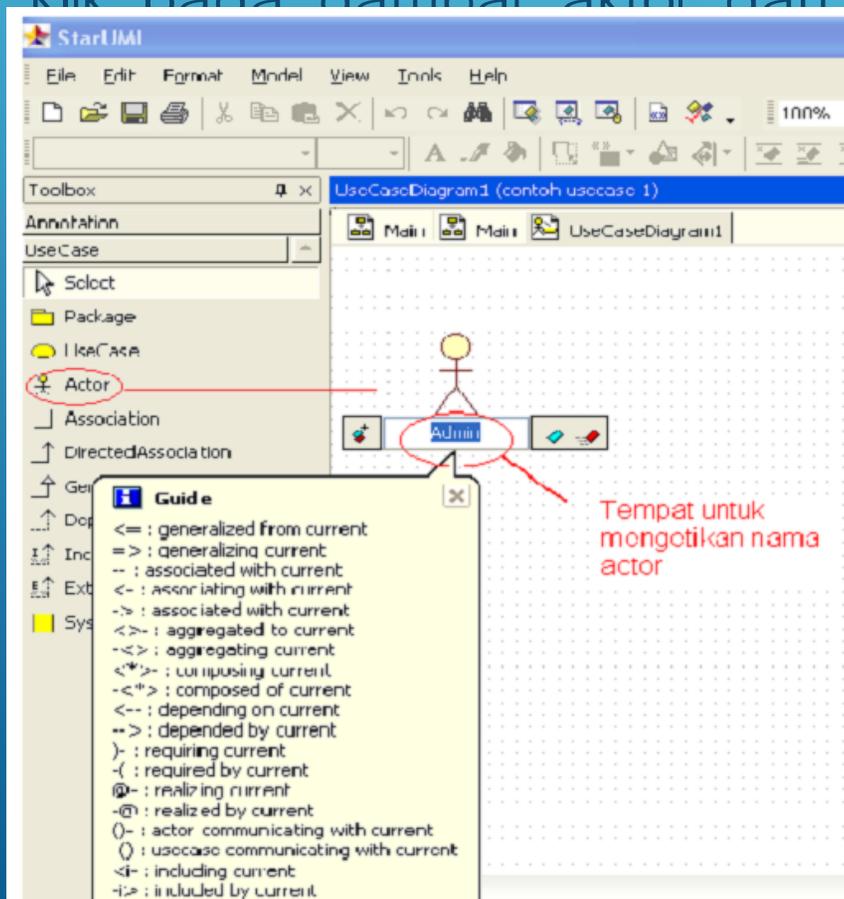


kiri akan berubah,

CONTOH MEMBUAT USE CASE DIAGRAM

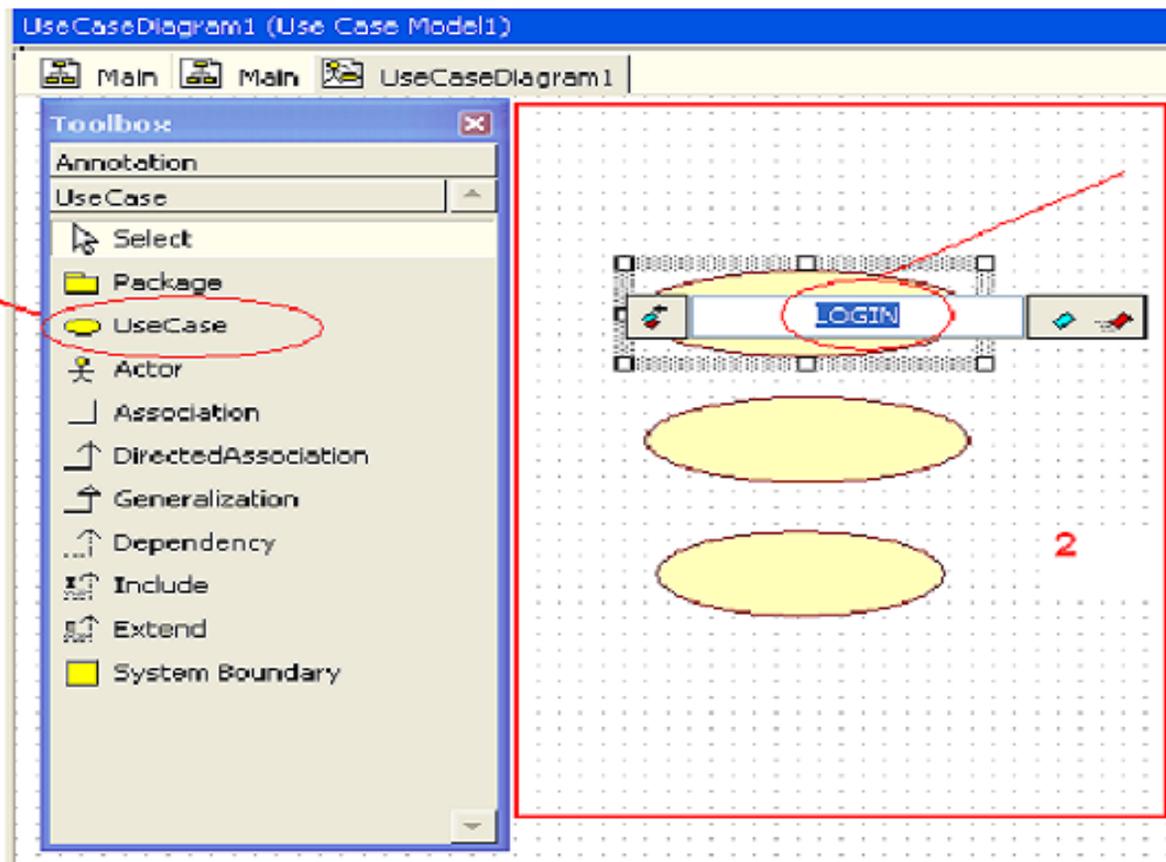
4.

Klik pada gambar aktor dan taruh kursor pada samping gambar orang yang disebut nama actor, seperti gambar



CONTOH MEMBUAT USE CASE DIAGRAM

1
Klik gambar use case



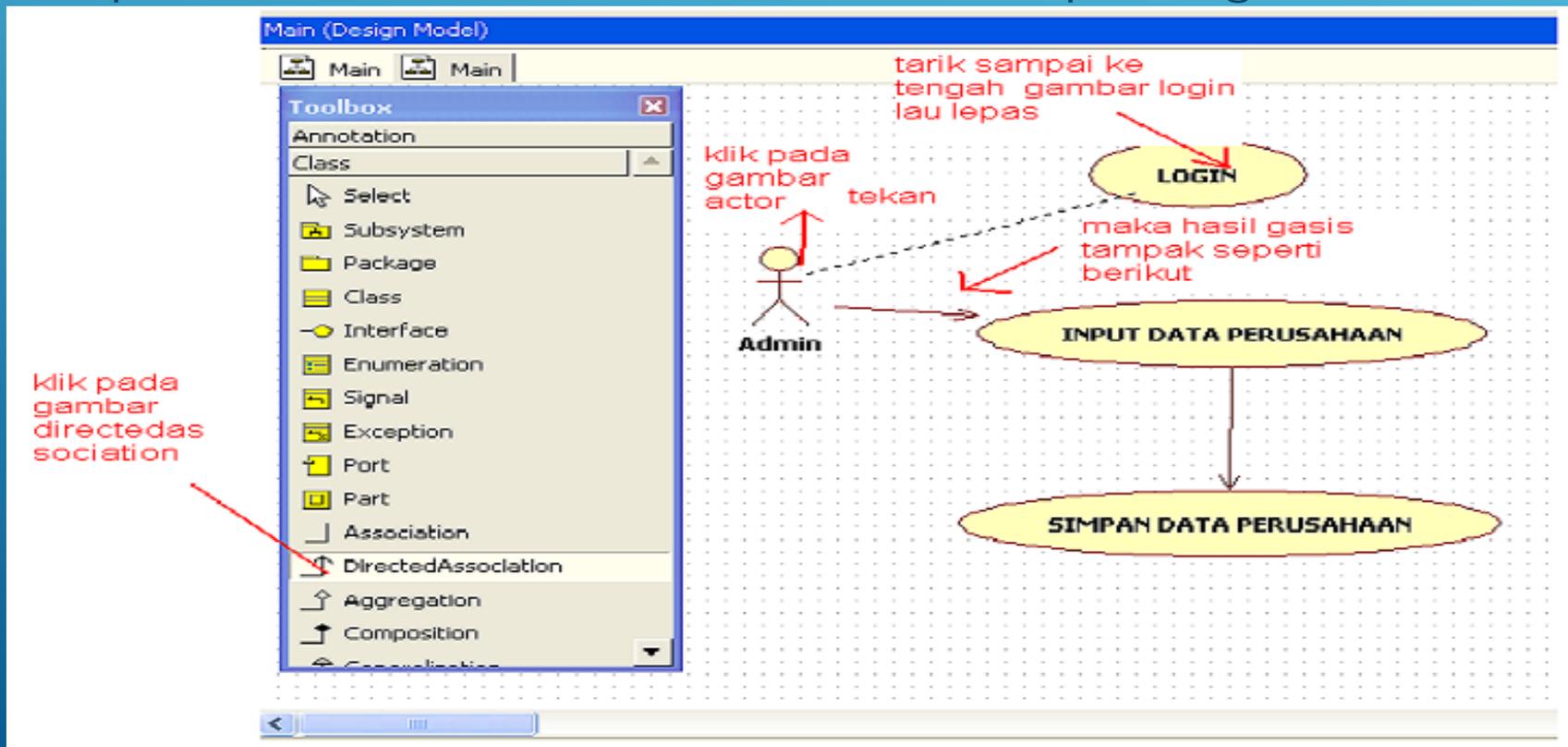
3
Tempat untuk mengetikkan nama usecase

2

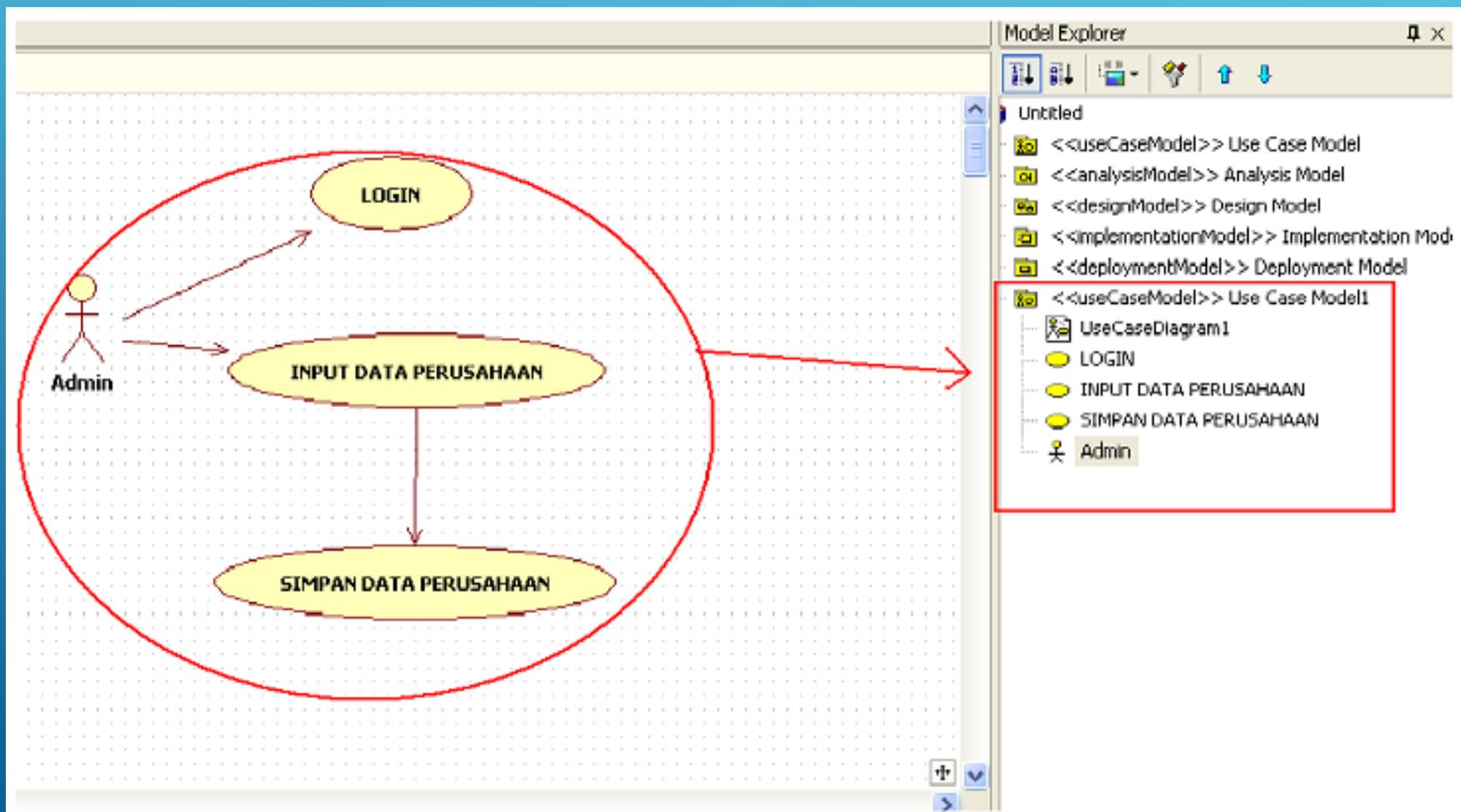
Lembar kerja

CONTOH MEMBUAT USE CASE DIAGRAM

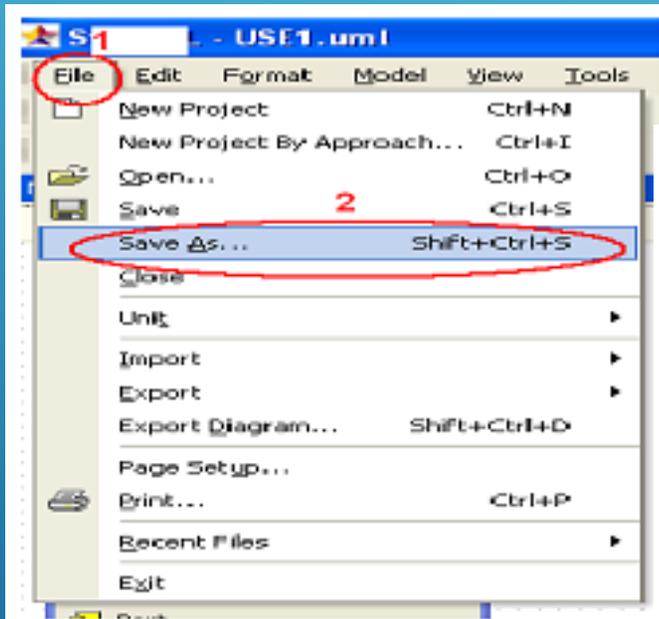
- Untuk membuat garis hubung antara actor dan use case pilih directed association tekan kursor pada gambar actor



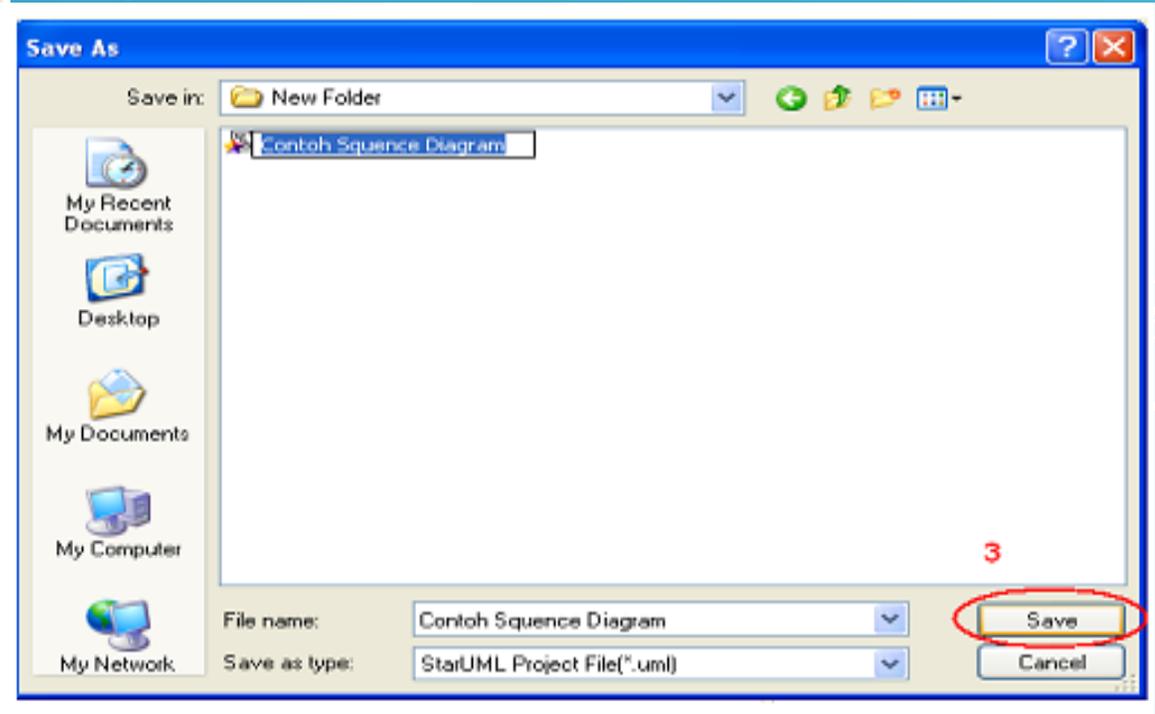
CONTOH MEMBUAT USE CASE DIAGRAM



CONTOH MEMBUAT USE CASE DIAGRAM

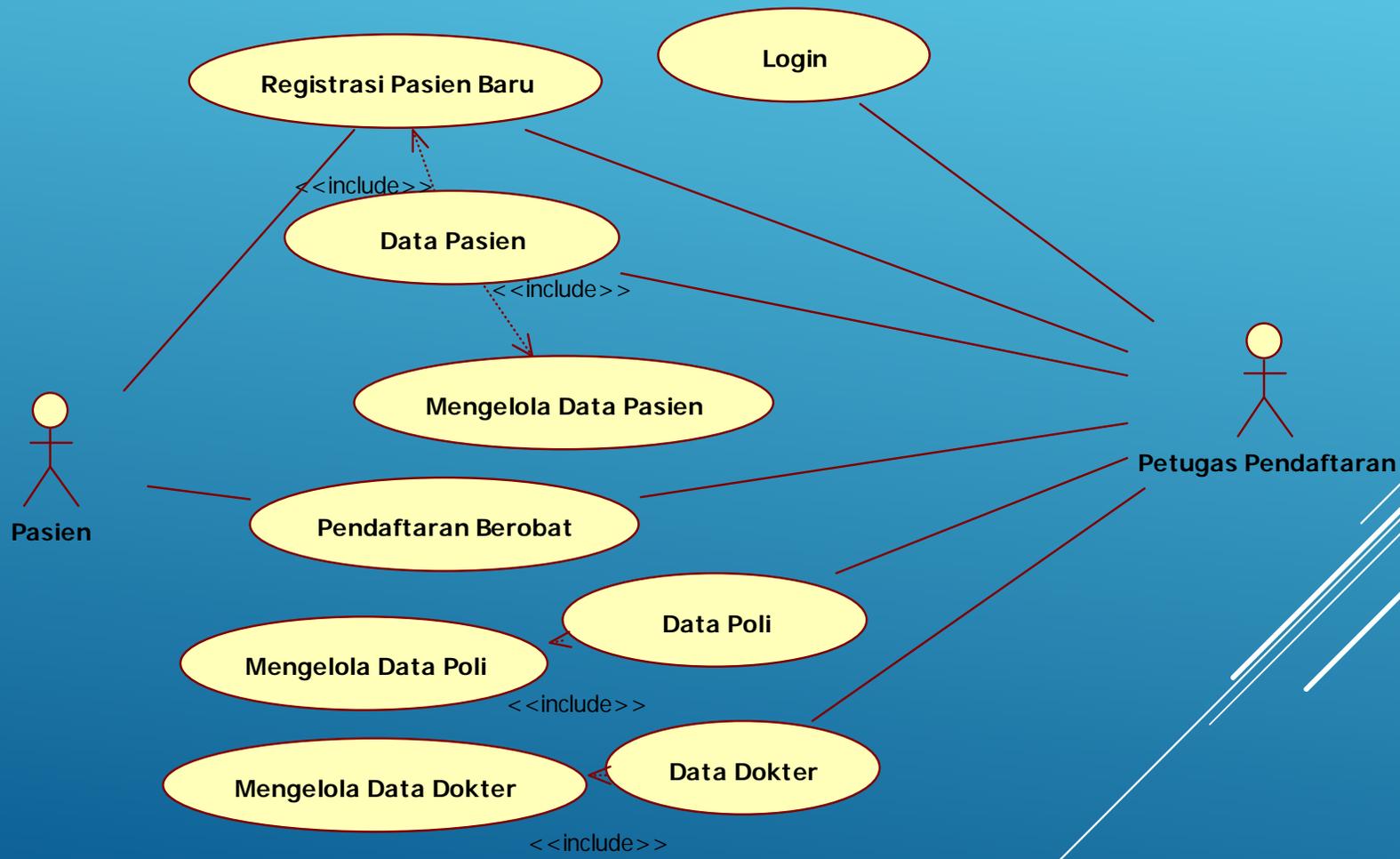


dengan cara pilih file -> save as, seperti



CONTOH USE CASE DIAGRAM

Studi Kasus Pendaftaran Pasien Berobat Pada Poliklinik





Modul Star

UML