



Kuliah

***Rekayasa Perangkat Lunak  
(Software Engineering)***

Bagian 2

***Software Engineering: A Practitioner's Approach, 6/e***

**Chapter 23**

**Pemodelan Analisis**



# Model berbasis Scenario

“[Use-cases] adalah bantuan untuk mendefinisikan apa yang ada pada sistem (aktor) dan apa yang harus dilakukan sistem (use-cases).” Ivar Jacobson

- (1) Apa yang harus ditulis?
- (2) Berapa banyak kita harus menulisnya?
- (3) Sedetail apa gambaran kita ?
- (4) Bagaimana kita mengatur deskripsi?



# Use-Cases

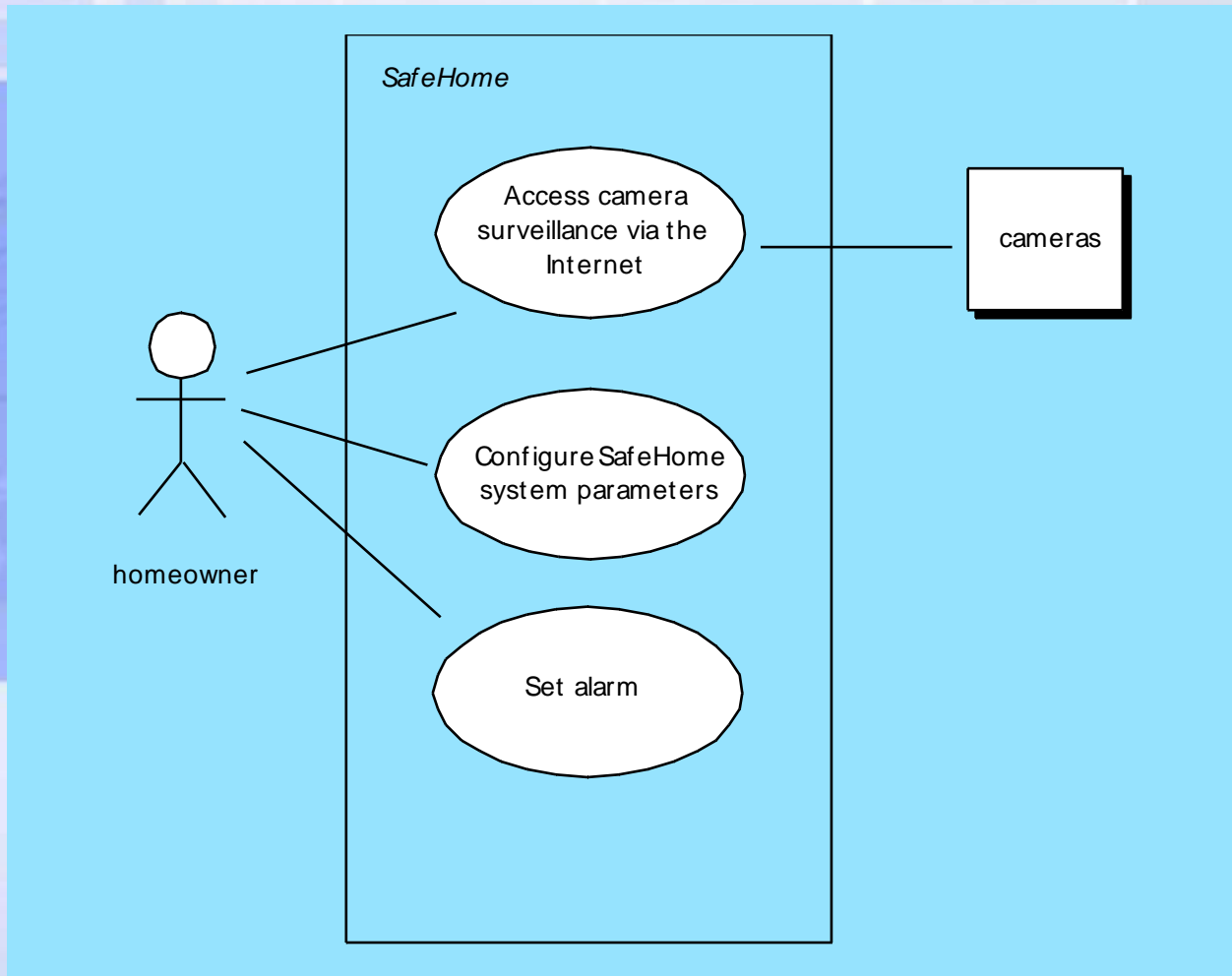
- Sebuah skenario yang menggambarkan rangkaian kegunaan pada sistem
- *actors* mewakili peran orang atau piranti yang dimainkan ketika sistem berfungsi
- *users* dapat berperan sebagai lebih dari satu peran dalam sebuah skenario yang ditentukan

# Mengembangkan Use-Case



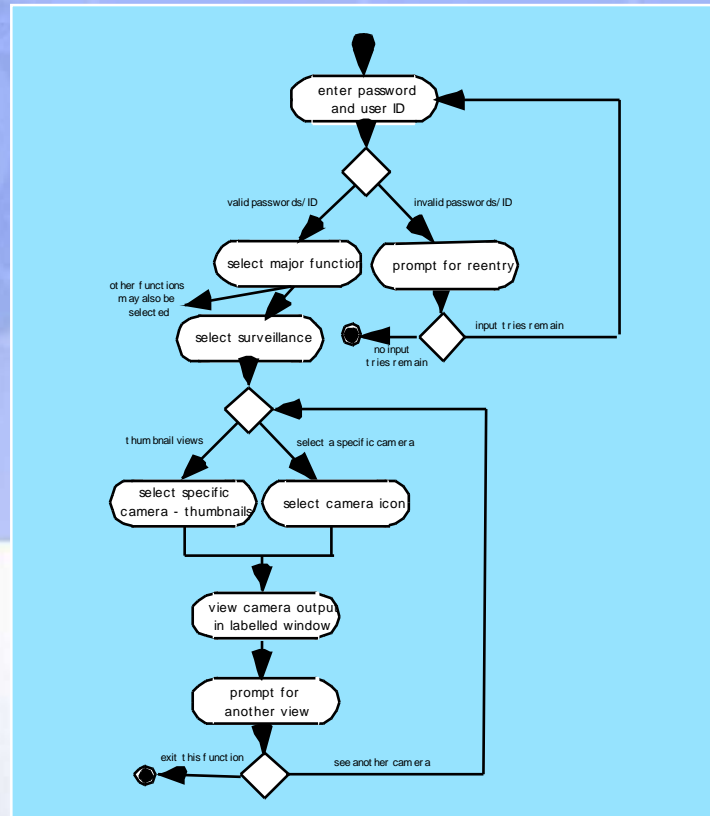
- Apa tugas atau fungsi utama yang harus dilakukan aktor ?
- Sistem Informasi seperti apa yang diperlukan, dihasilkan atau diubah oleh aktor ?
- Apakah aktor harus menginformasikan sistem tentang perubahan dalam lingkungan eksternal?
- Informasi apa yang diharapkan aktor dari sistem?
- Apakah aktor menginginkan diberitahu tentang perubahan yang tidak terduga?

# Use-Case Diagram



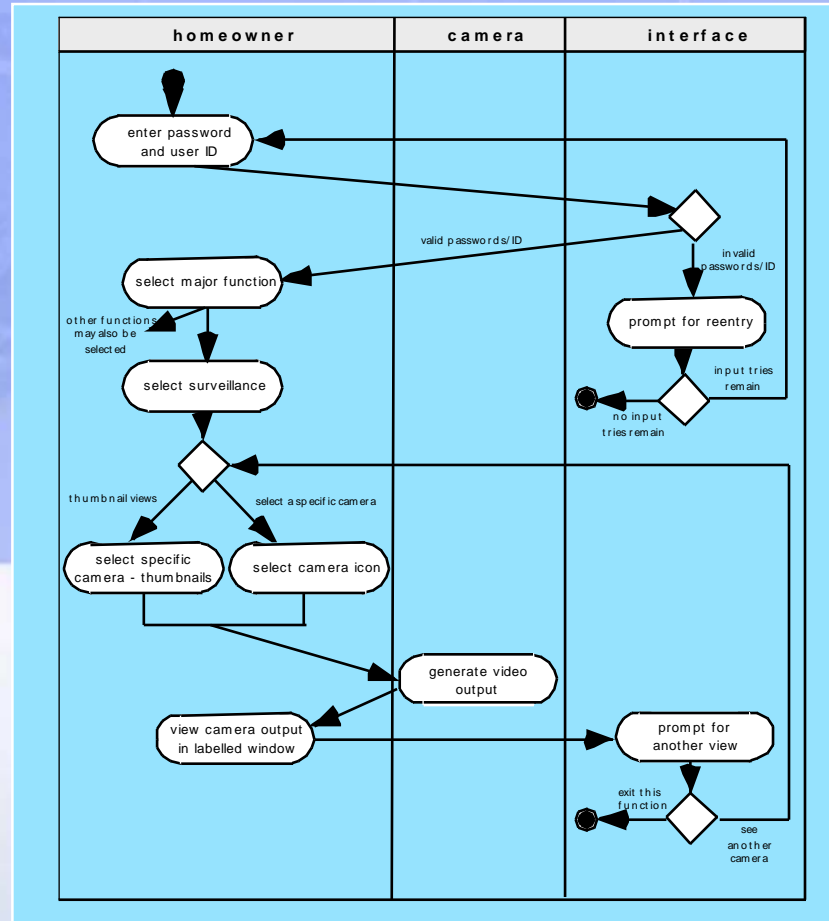
# Activity Diagram

Melengkapi use-case dengan menyediakan representasi diagram dari aliran prosedural.



# Swimlane Diagrams

Memungkinkan untuk menampilkan aliran aktivitas yang digambarkan oleh use-case, dan di saat yang sama mengindikasikan aktor yang mana, atau class analisis yang mempunyai tanggungjawab terhadap tindakan yang digambarkan oleh kotak aktivitas





# ***Pemodelan berorientasi aliran***

Menampilkan bagaimana objek data ditransformasi ketika mereka bergerak di dalam sistem

Sebuah **data flow diagram (DFD)** merupakan bentuk diagram yang digunakan

Walaupun dianggap pendekatan kuno, pemodelan berorientasi aliran menyediakan pandangan unik terhadap suatu sistem. Dia tetap layak digunakan untuk mendukung analisis elemen model lainnya.