Course : COMP6100/Software Engineering

Effective Period : Desember 2017

Requirement Modeling

Session 09 & 10

People
Innovation
Excellence

Acknowledgement

These slides have been adapted from Pressman, R.S. (2015). *Software Engineering : A Practioner's Approach. 8th ed*. McGraw-Hill Companies.Inc, Americas, New York. ISBN : 978 1 259 253157. Chapter 9, 10, 11 and 12

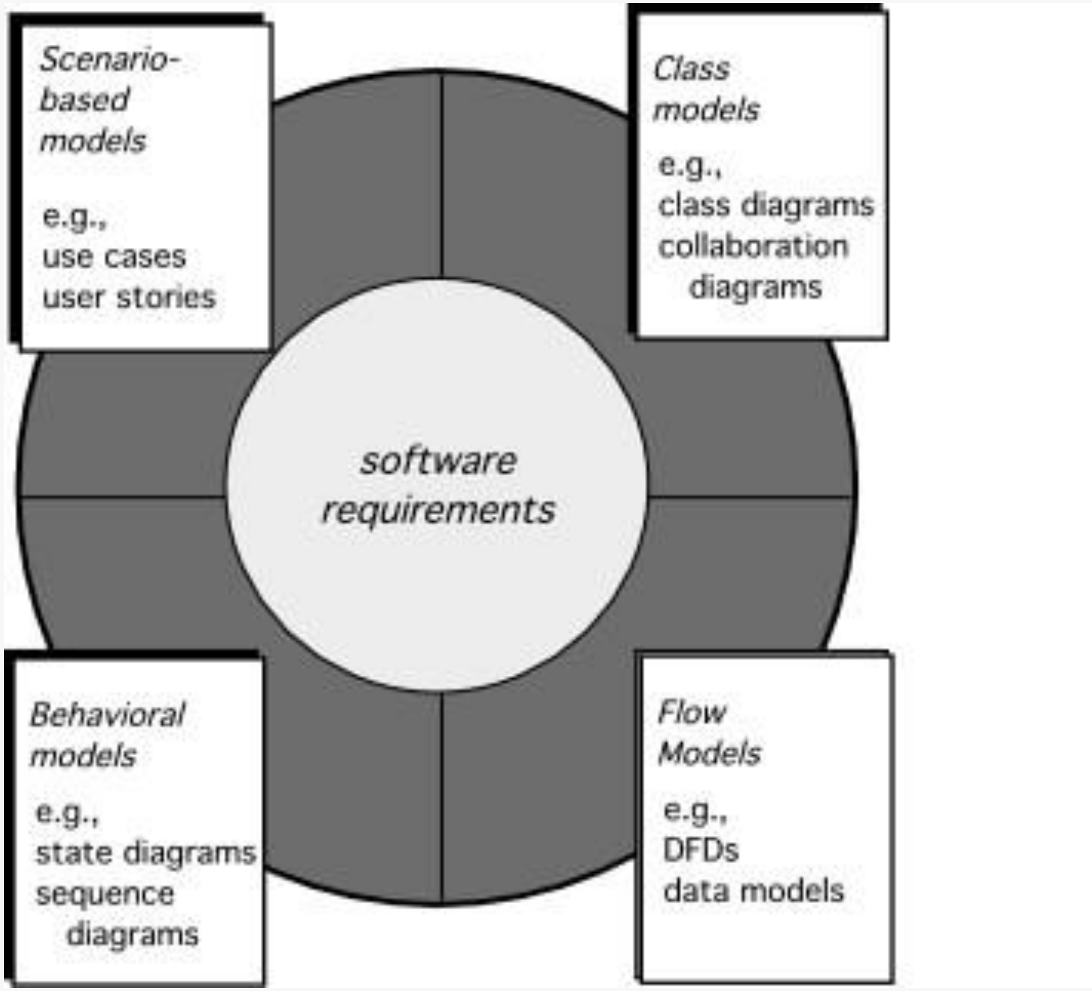
Learning Objectives

LO 2 : Explain the software engineering practices and business environment

Contents

- Requirement Analysis
- Scenario-Based Modeling
- Class-Based Modeling
- Creating Behavioral Modeling
- Requirements Modeling for Web and Mobile Apps

Requirement Analysis



software requirements

Scenario-based models

e.g.,
use cases
user stories

Class models

e.g.,
class diagrams
collaboration diagrams

Behavioral models

e.g.,
state diagrams
sequence diagrams

Flow Models

e.g.,
DFDs
data models

People
Innovation
Excellence

BINUS UNIVERSITY

Scenario-Based Modeling

"[Use-cases] are simply an aid to defining what exists outside the system (actors) and what should be performed by the system (use-cases)." Ivar Jacobson

(1) What should we write about?

(2) How much should we write about it?

(3) How detailed should we make our description?

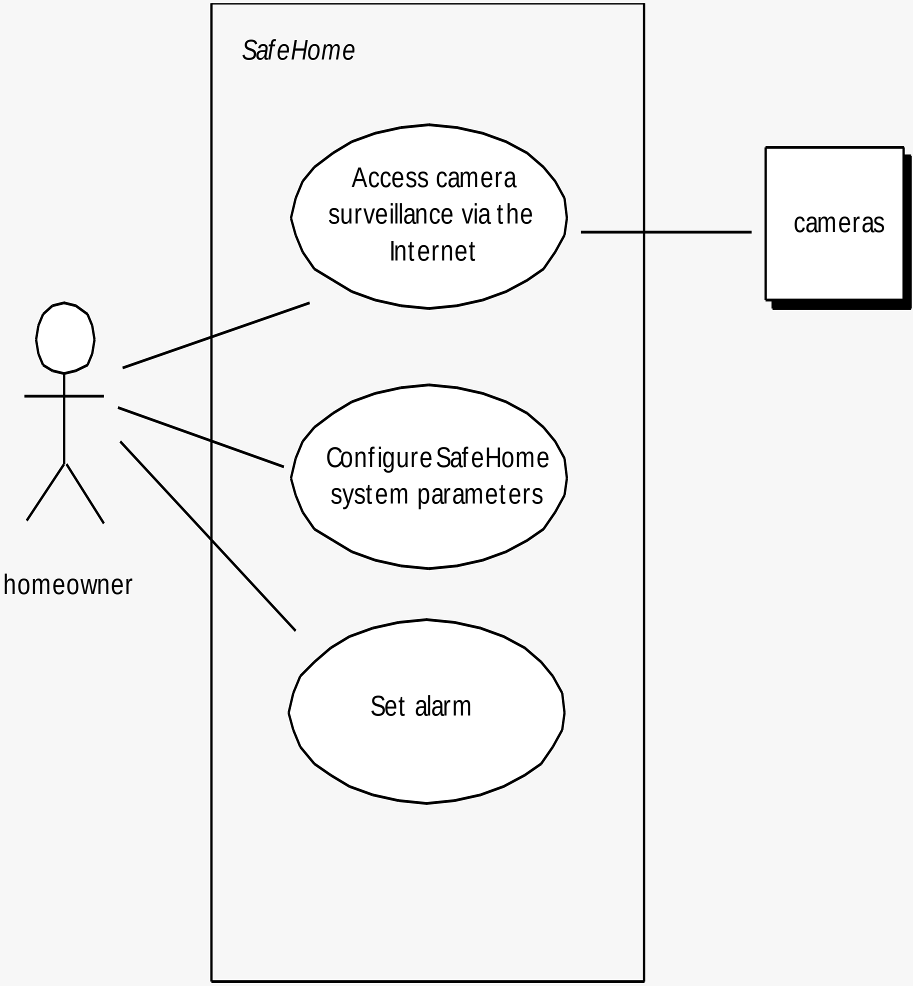
(4) How should we organize the description?

People
Innovation
Excellence

Scenario-Based Modeling

Use-Cases

- a scenario that describes a “thread of usage” for a system
- *actors* represent roles people or devices play as the system functions
- *users* can play a number of different roles for a given scenario

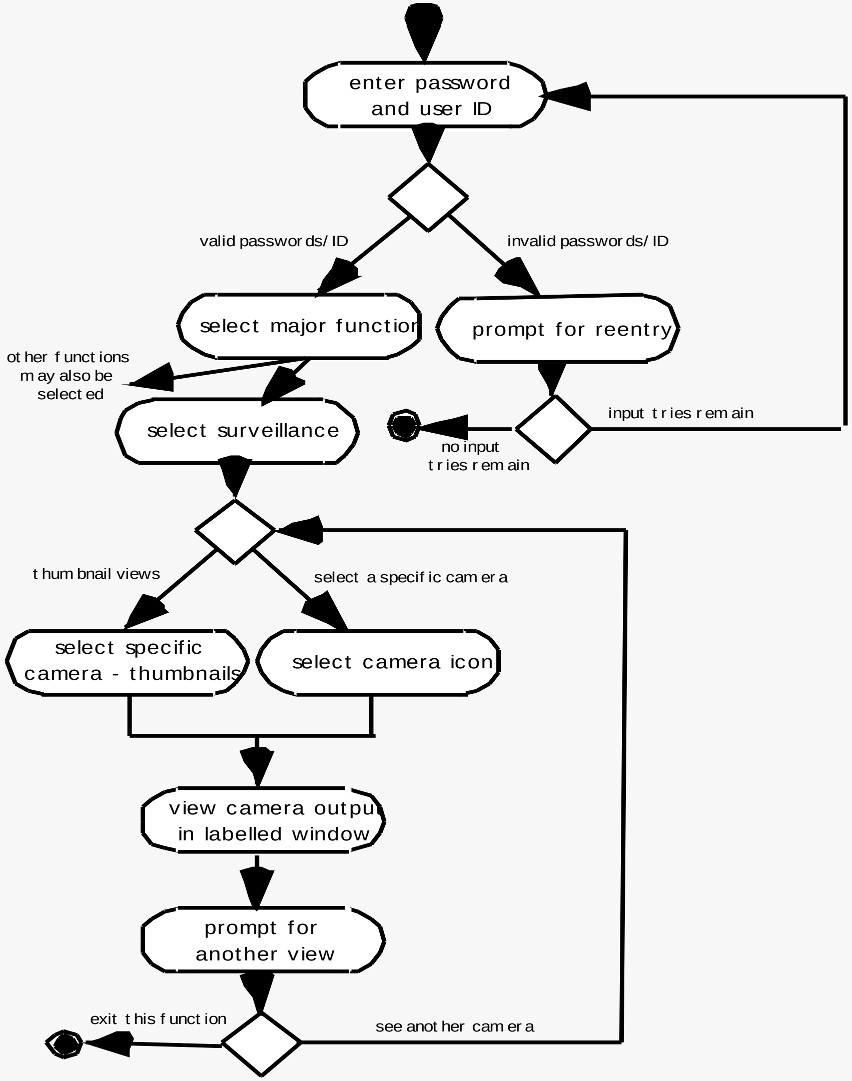

SafeHome
Access camera surveillance via the Internet
Configure SafeHome system parameters
Set alarm
homeowner
cameras

People
Innovation
Excellence

Scenario-Based Modeling

Activity Diagram

Supplements the use case by providing a graphical representation of the flow of interaction within a specific scenario

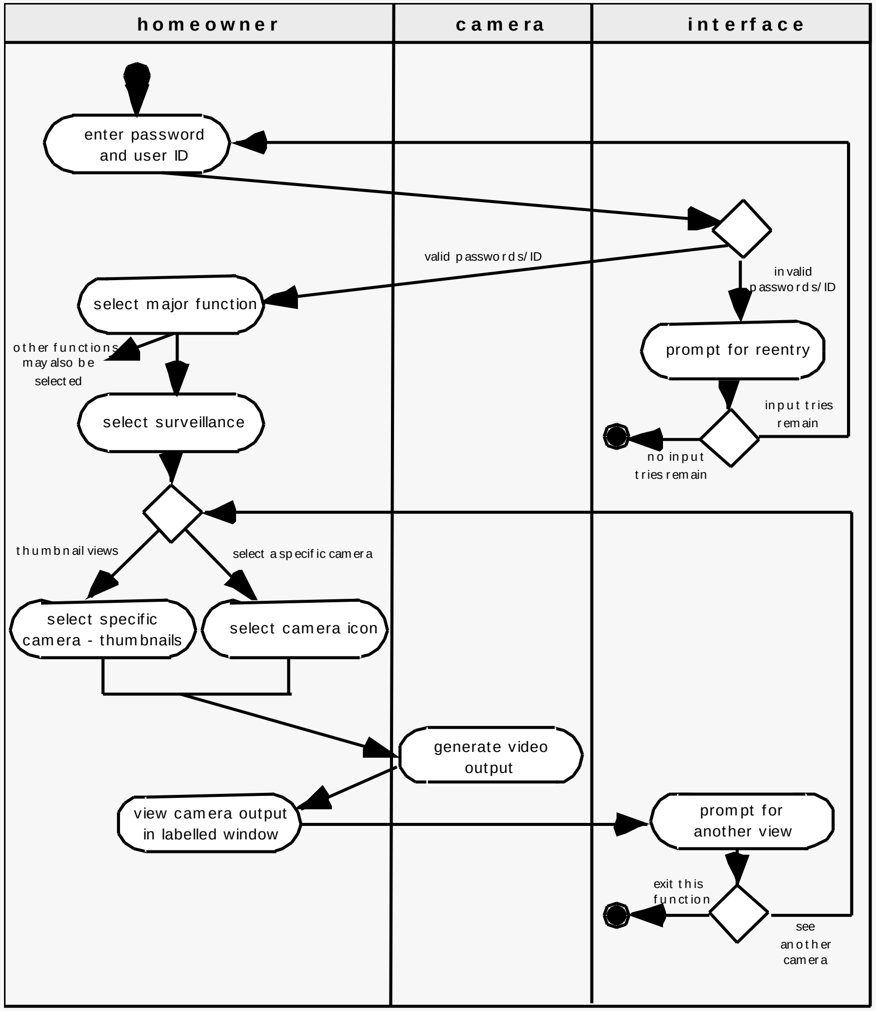


People
Innovation
Excellence

Scenario-Based Modeling

Swimlane Diagrams

Allows the modeler to represent the flow of activities described by the use-case and at the same time indicate which actor (if there are multiple actors involved in a specific use-case) or analysis class has responsibility for the action described by an activity rectangle



People
Innovation
Excellence

Class-Based Modeling

- Class-based modeling represents:
  - objects that the system will manipulate
  - operations (also called methods or services) that will be applied to the objects to effect the manipulation
  - relationships (some hierarchical) between the objects
  - collaborations that occur between the classes that are defined.
- The elements of a class-based model include classes and objects, attributes, operations, CRC models, collaboration diagrams and packages.

Class-Based Modeling

Identifying Analysis Classes

- Examining the usage scenarios developed as part of the requirements model and perform a "grammatical parse" [Abb83]
  - Classes are determined by underlining each noun or noun phrase and entering it into a simple table.
  - Synonyms should be noted.
  - If the class (noun) is required to implement a solution, then it is part of the solution space; otherwise, if a class is necessary only to describe a solution, it is part of the problem space.

- *But what should we look for once all of the nouns have been isolated?*

Class-Based Modeling

Manifestations of Analysis Classes

Analysis classes manifest themselves in one of the following ways:

- ***External entities:*** (e.g., other systems, devices, people) that produce or consume information
- ***Things***: (e.g, reports, displays, letters, signals) that are part of the information domain for the problem
- ***Occurrences or events:*** (e.g., a property transfer or the completion of a series of robot movements) that occur within the context of system operation
- ***Roles:*** (e.g., manager, engineer, salesperson) played by people who interact with the system
- ***Organizational units:*** (e.g., division, group, team) that are relevant to an application
- ***Places:*** (e.g., manufacturing floor or loading dock) that establish the context of the problem and the overall function
- ***Structures:*** *(*e.g., sensors, four-wheeled vehicles, or computers) that define a class of objects or related classes of objects

Class-Based Modeling

Defining Attributes

- *Attributes* describe a class that has been selected for inclusion in the analysis model.
  - build two different classes for professional baseball players
    - **For Playing Statistics software:** name, position, batting average, fielding percentage, years played, and games played might be relevant
    - **For Pension Fund software:** average salary, credit toward full vesting, pension plan options chosen, mailing address, and the like.

Class-Based Modeling

Defining Operations

- Do a grammatical parse of a processing narrative and look at the verbs
- Operations can be divided into four broad categories:
  - (1) operations that manipulate data in some way (e.g., adding, deleting, reformatting, selecting)
  - (2) operations that perform a computation
  - (3) operations that inquire about the state of an object, and
  - (4) operations that monitor an object for the occurrence of a controlling event.

Class-Based Modeling

CRC Models

- ***Class-responsibility-collaborator (CRC) modeling*** [Wir90] provides a simple means for identifying and organizing the classes that are relevant to system or product requirements. Ambler [Amb95] describes CRC modeling in the following way:
  - A CRC model is really a collection of standard index cards that represent classes. The cards are divided into three sections. Along the top of the card you write the name of the class. In the body of the card you list the class responsibilities on the left and the collaborators on the right.

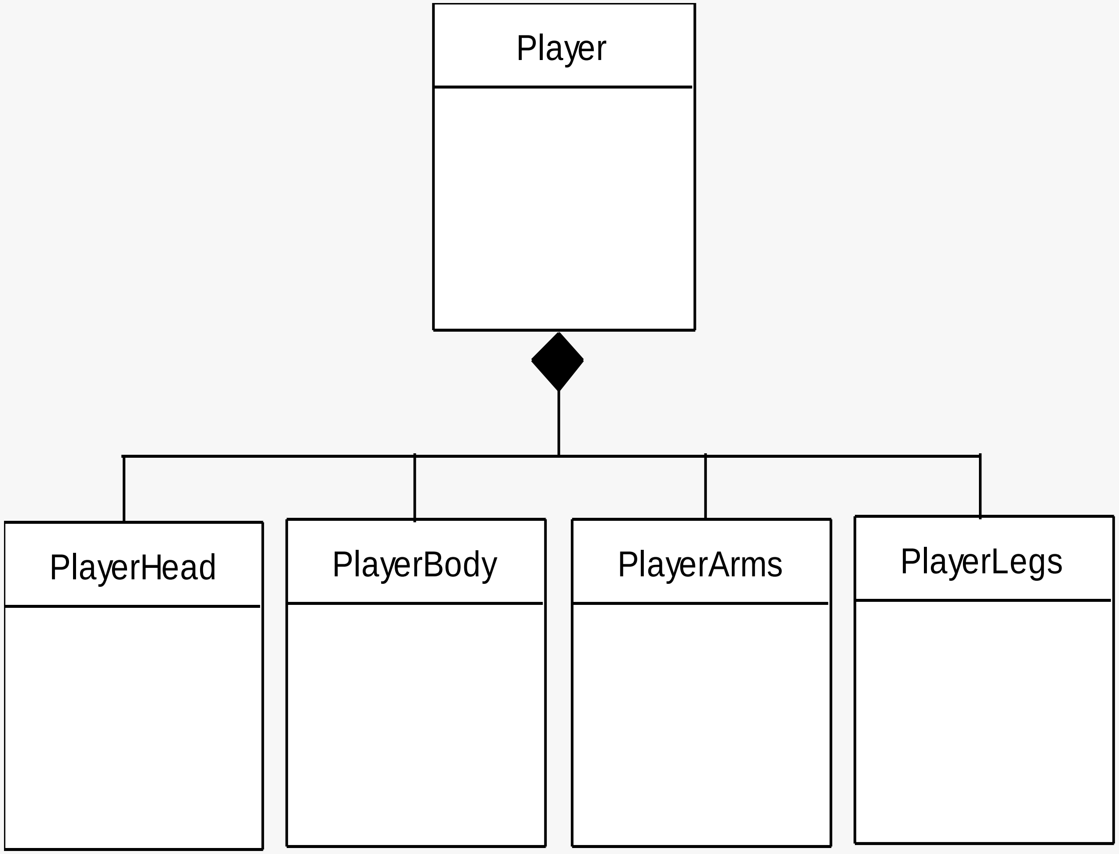
Class-Based Modeling

CRC Modeling

Class: FloorPlan	
Description:	
Responsibility:	**Collaborator:**
defines floor plan name/type	
manages floor plan positioning	
scales floor plan for display	
scales floor plan for display	
incorporates walls, doors and windows	Wall
shows position of video cameras	Camera

Class-Based Modeling

Composite Aggregate Class



Class-Based Modeling

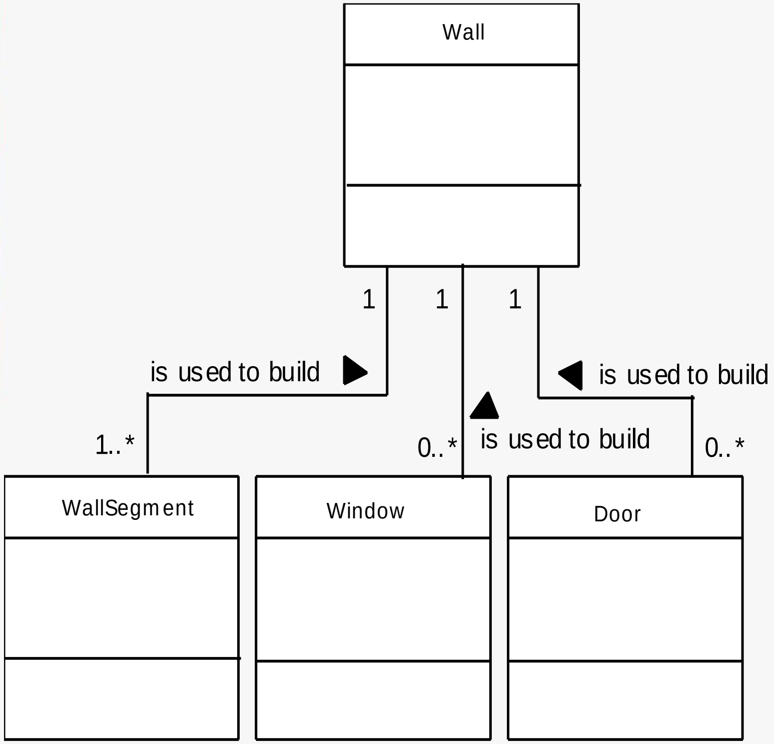
Associations and Dependencies

- Two analysis classes are often related to one another in some fashion
  - In UML these relationships are called *associations*
  - Associations can be refined by indicating *multiplicity* (the term *cardinality* is used in data modeling
- In many instances, a client-server relationship exists between two analysis classes.
  - In such cases, a client-class depends on the server-class in some way and a *dependency relationship* is established

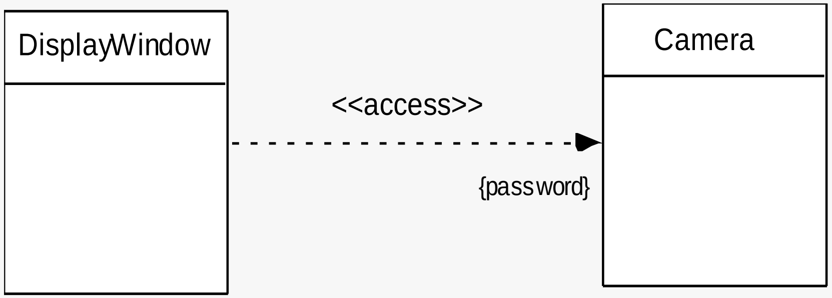
People
Innovation
Excellence

Class-Based Modeling

Multiplicity



Dependencies



People
Innovation
Excellence

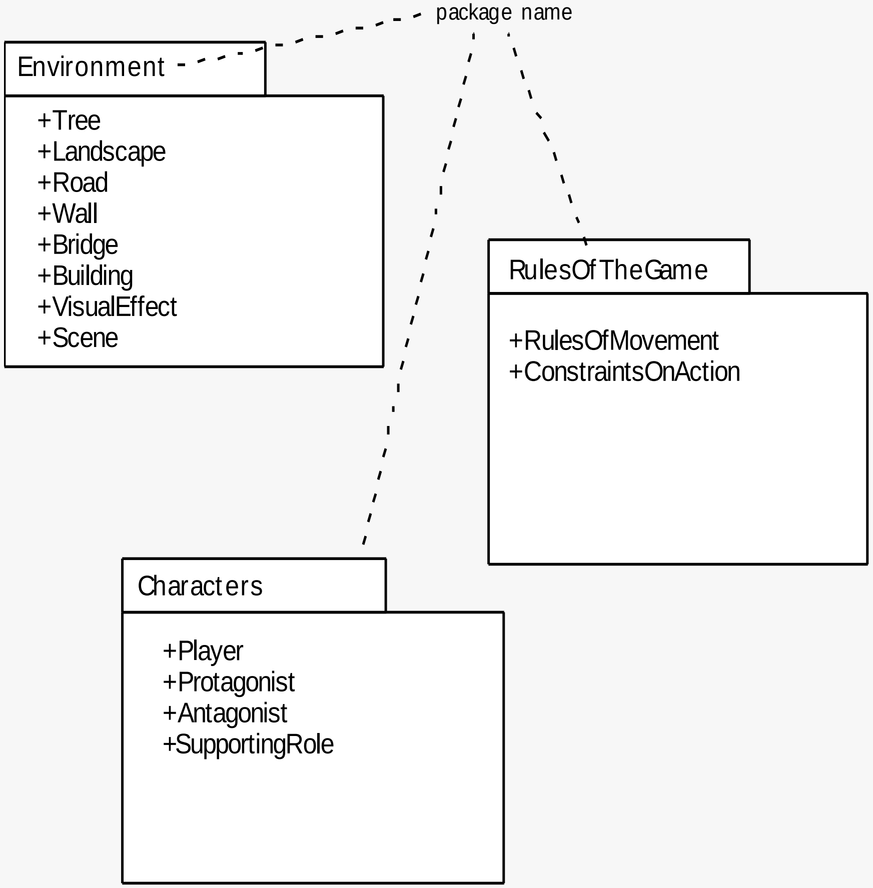
Class-Based Modeling

Analysis Packages

- Various elements of the analysis model (e.g., use-cases, analysis classes) are categorized in a manner that packages them as a grouping
- The plus sign preceding the analysis class name in each package indicates that the classes have public visibility and are therefore accessible from other packages.
- Other symbols can precede an element within a package. A minus sign indicates that an element is hidden from all other packages and a # symbol indicates that an element is accessible only to packages contained within a given package.

Class-Based Modeling

Analysis Packages



package name

Environment

+Tree
+Landscape
+Road
+Wall
+Bridge
+Building
+VisualEffect
+Scene

RulesOfTheGame

+RulesOfMovement
+ConstraintsOnAction

Characters

+Player
+Protagonist
+Antagonist
+SupportingRole

People
Innovation
Excellence

Creating A Behavioral Modeling

- The behavioral model indicates how software will respond to external events or stimuli. To create the model, the analyst must perform the following steps:
    - Evaluate all use-cases to fully understand the sequence of interaction within the system.
    - Identify events that drive the interaction sequence and understand how these events relate to specific objects.
    - Create a sequence for each use-case.
    - Build a state diagram for the system.
    - Review the behavioral model to verify accuracy and consistency

People
Innovation
Excellence

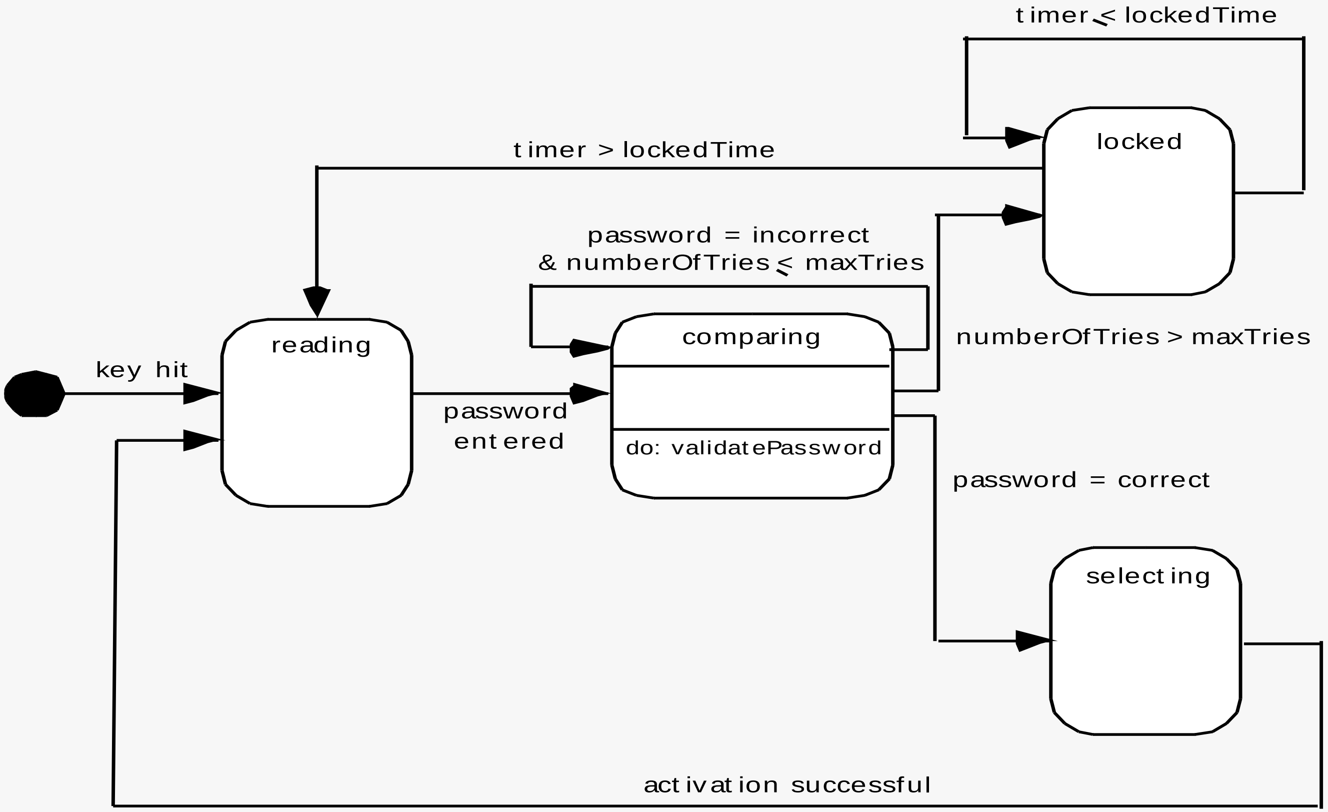
Creating A Behavioral Modeling

State Representations

- In the context of behavioral modeling, two different characterizations of states must be considered:
  - the state of each class as the system performs its function and
  - the state of the system as observed from the outside as the system performs its function
- The state of a class takes on both passive and active characteristics [CHA93].
  - A *passive state* is simply the current status of all of an object's attributes.
  - The *active state* of an object indicates the current status of the object as it undergoes a continuing transformation or processing.

Creating A Behavioral Modeling

State Diagram for the ControlPanel Class



People
Innovation
Excellence

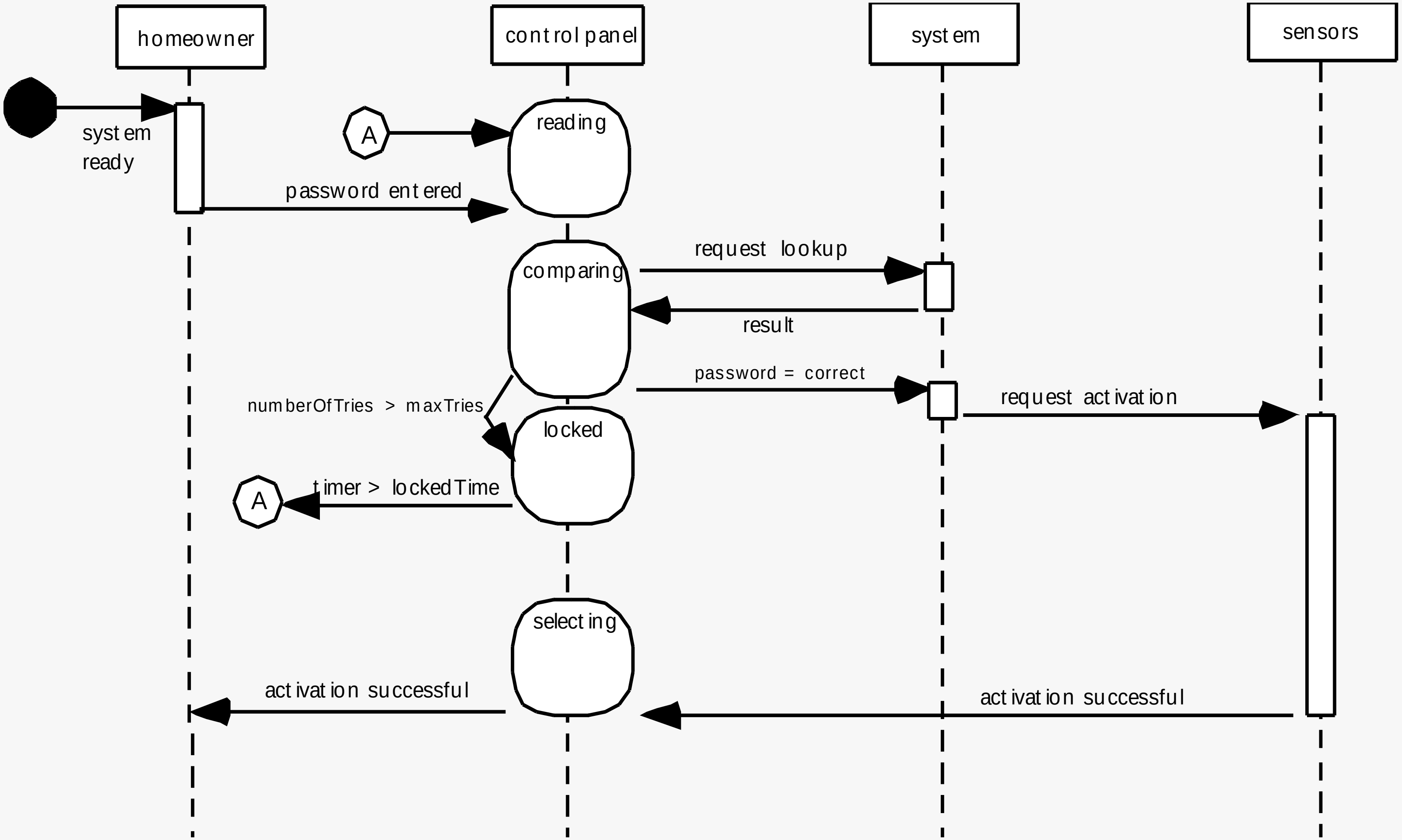
Creating A Behavioral Modeling

The States of a System

- state—a set of observable circum-stances that characterizes the behavior of a system at a given time
- state transition—the movement from one state to another
- event—an occurrence that causes the system to exhibit some predictable form of behavior
- action—process that occurs as a consequence of making a transition

Creating A Behavioral Modeling

Sequence Diagram



People
Innovation
Excellence

When Do We Perform Analysis?

- In some WebE situations, analysis and design merge. However, **an explicit analysis activity occurs when …**
  - the WebApp to be built is large and/or complex
  - the number of stakeholders is large
  - the number of Web engineers and other contributors is large
  - the goals and objectives (determined during formulation) for the WebApp will effect the business’ bottom line
  - the success of the WebApp will have a strong bearing on the success of the business

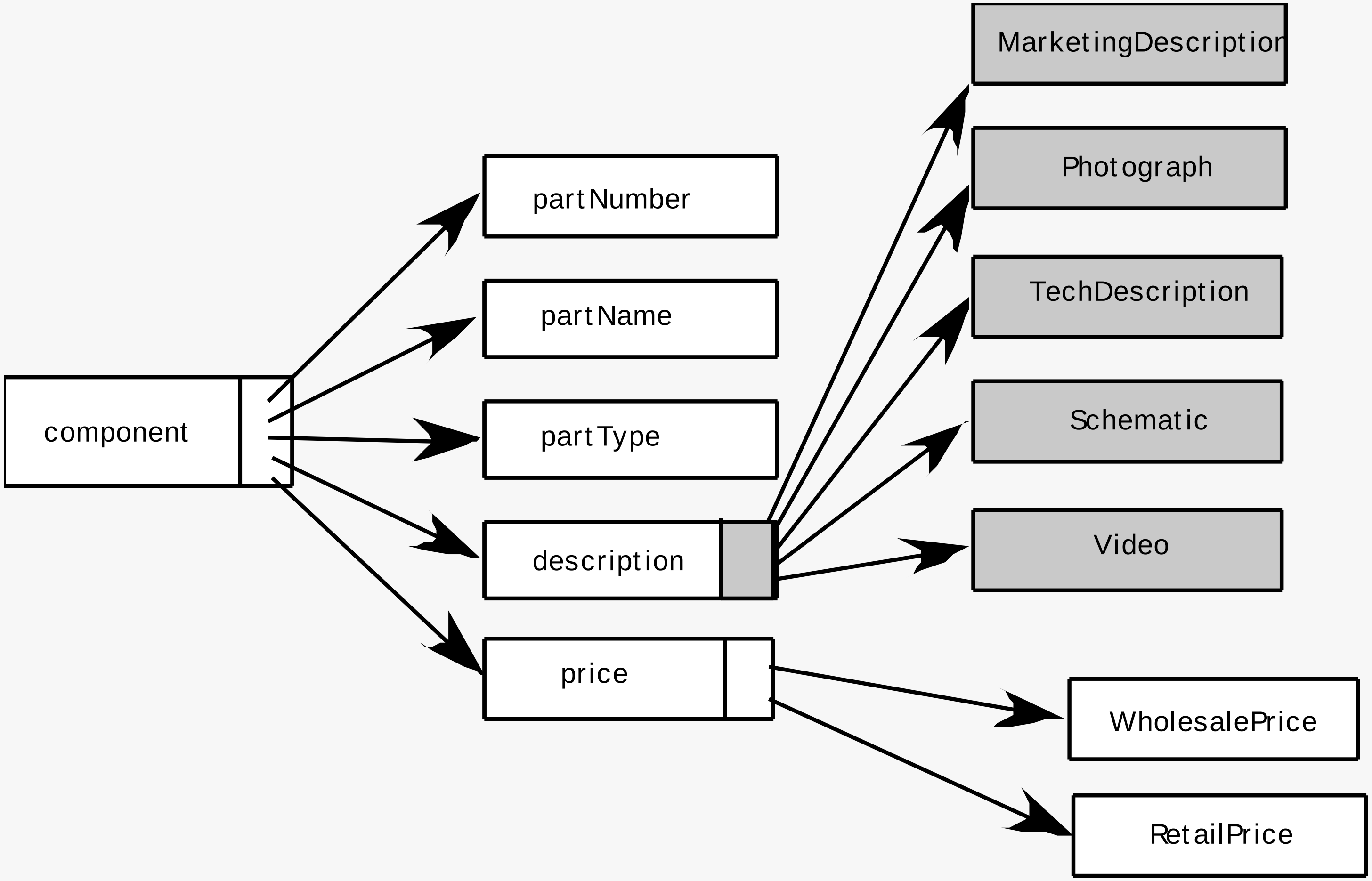
Requirements Modeling for Web and Mobile Apps

The Content Model

- **Content objects** are extracted from use-cases
  - examine the scenario description for direct and indirect references to content
- **Attributes of** each content object are identified
- The **relationships** among content objects and/or the hierarchy of content maintained by a WebApp
  - Relationships—entity-relationship diagram or UML
  - Hierarchy—data tree or UML

Requirements Modeling for Web and Mobile Apps

Data Tree



People
Innovation
Excellence

The Interaction Model

- Composed of four elements:
  - use-cases
  - sequence diagrams
  - state diagrams
  - a user interface prototype

Requirements Modeling for Web and Mobile Apps

The Functional Model

- The functional model addresses two processing elements of the WebApp
  - user observable functionality that is delivered by the WebApp to end-users
  - the operations contained within analysis classes that implement behaviors associated with the class.
- An activity diagram can be used to represent processing flow

The Configuration Model

- Server-side
  - Server hardware and operating system environment must be specified
  - Interoperability considerations on the server-side must be considered
  - Appropriate interfaces, communication protocols and related collaborative information must be specified
- Client-side
  - Browser configuration issues must be identified
  - Testing requirements should be defined

Exercises

- You have been asked to build one of the following systems:
  a. a network-based course registration system for your university
  b. a web-based order-processing system for a computer store
  c. a simple invoicing system for a small business.
  d. an internet based cookbook that is built into an electric range or microwave,

  Select the system that is of interest to you and develop an entity-relationship diagram that describes data objects, relationships, and attributes

Exercises

1. The depaftment of public works for a large city has decided to develop a web-based pothole tracking and repair system (PHTRS). A. description follows:

   Citizens can log onto a website and report the location and severity of potholes. As potholes are reported they are logged within a "public works department repair system" and are assigned an identifying .number, stored by street address, size (on a scale of 1 to 1 0), location-(middle, curb, etc.), district (determined from street address), and repair priority (determined from the size of the pothole). work order data are associated with each pothole. and include pothole location and size, repair crew identifying number, number of people on crew, equipment assigned, hours applied to repair, hole status (work in progress, repaired, temporary repair, not repaired), amount of filler material used and

   cost of repair computed (from hours applied, number of people, material and equipment

   used). FInally, a damage file is created to hold information about reported damage due

   to the pothoie and incllude citizen's name, address, phone number, type of damage, and

   dollar amount of damage. PHTRS is an online system; all queries are to be made interactively

   a. Draw a UML use case diagram for the PHTRS system. You',ll have to make a number of assumptions about the manner in which a user interacts with this system.

   b. Develop a class model for the PHTRS system

People
Innovation
Excellence

Exercises

2. What is the fundamental difference between the structured analysis and object-oriented strategies for requirements analysis?
3. There are two different types of "states" that behavioral models can represent. What are they?
4. Suggest three requirements pattems for a modern mobile phone and write a brief description of each. Could these patterns be used for other devices. Provide an example.

References

- Pressman, R.S. (2015). ***Software Engineering : A Practioner's Approach. 8th ed***. McGraw-Hill Companies.Inc, Americas, New York. ISBN : 978 1 259 253157
- **UML 2.0 Tutorial, http://www.youtube.com/watch?v=OkC7HKtiZC0**
- Introduction to Model-View-View-Model (MVVM) Pattern, http://www.youtube.com/watch?v=pY60lLotZCg

People
Innovation
Excellence

BINUS
UNIVERSITY

Q & A

People
Innovation
Excellence