BINUS UNIVERSITY

People
Innovation
Excellence

Course : COMP6100/Software Engineering

Effective Period : Desember 2017

Requirement Engineering and Modeling

Session 05

BINUS UNIVERSITY

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 8,9,10 and 11

Learning Objectives

LO 2 : Explain the software engineering practices and business environment

People
Innovation
Excellence

BINUS UNIVERSITY

Contents

- Requirement Engineering
- Requirement Modeling : Scenario-Based Methods
- Requirement Modeling : Class-Based Methods
- Requirement Modeling : Behavior, Patterns, and Web/Mobile Apps

People
Innovation
Excellence

Requirement Engineering

- **Inception**—ask a set of questions that establish …
  - basic understanding of the problem
  - the people who want a solution
  - the nature of the solution that is desired, and
  - the effectiveness of preliminary communication and collaboration between the customer and the developer
- **Elicitation**—elicit requirements from all stakeholders
- **Elaboration**—create an analysis model that identifies data, function and behavioral requirements
- **Negotiation**—agree on a deliverable system that is realistic for developers and customers

Requirement Engineering

- **Specification**—can be any one (or more) of the following:
  - A written document
  - A set of models
  - A formal mathematical
  - A collection of user scenarios (use-cases)
  - A prototype
- **Validation**—a review mechanism that looks for
  - errors in content or interpretation
  - areas where clarification may be required
  - missing information
  - inconsistencies (a major problem when large products or systems are engineered)
  - conflicting or unrealistic (unachievable) requirements.

Building the Analysis Model

Elements of the analysis model

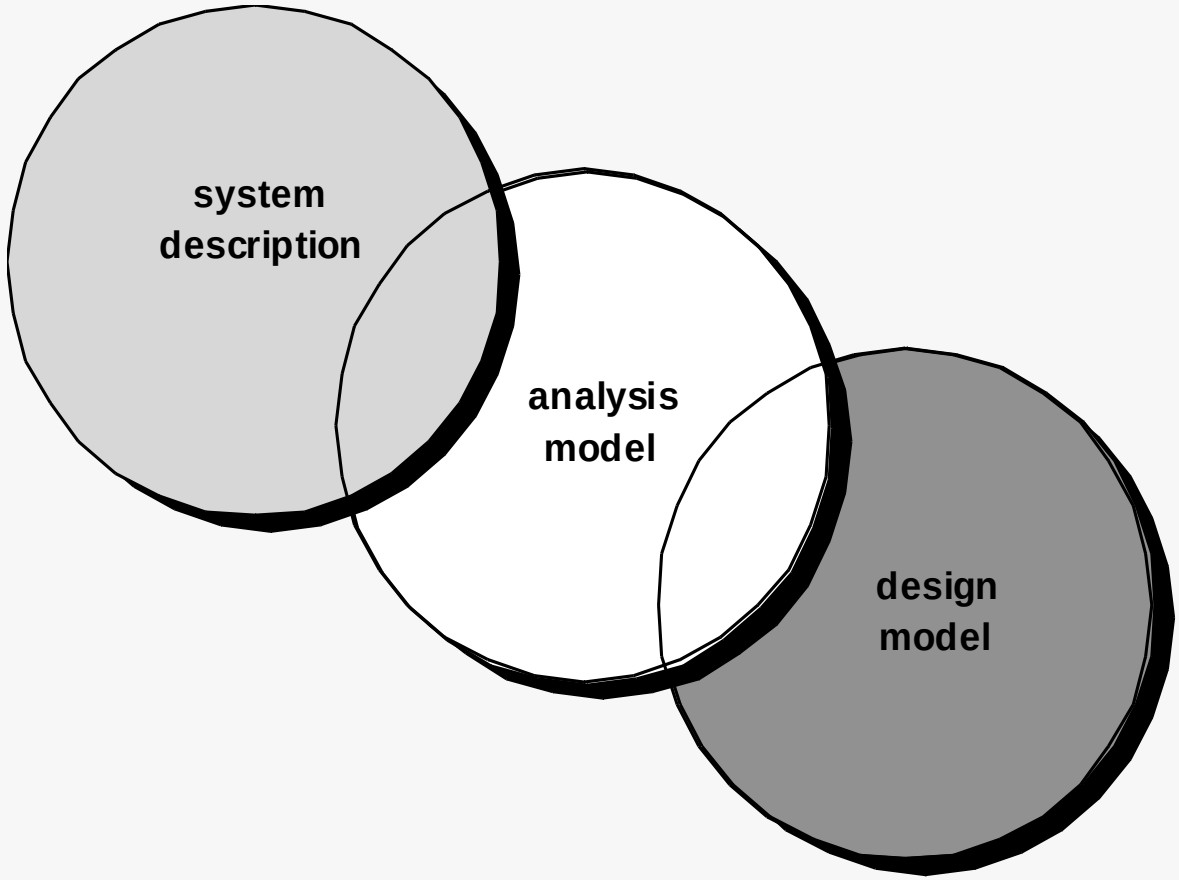
- **Scenario-based elements**
  - Functional—processing narratives for software functions
  - Use-case—descriptions of the interaction between an "actor" and the system
- **Class-based elements**
  - Implied by scenarios
- **Behavioral elements**
  - State diagram
- **Flow-oriented elements**
  - Data flow diagram

Requirement Modeling : Scenario-Based Methods

- Requirements analysis
  - specifies software's operational characteristics
  - indicates software's interface with other system elements
  - establishes constraints that software must meet
- Requirements analysis allows the software engineer (called an *analyst* or *modeler* in this role) to:
  - **elaborate** on basic requirements established during earlier requirement engineering tasks
  - **build models** that depict user scenarios, functional activities, problem classes and their relationships, system and class behavior, and the flow of data as it is transformed

Requirement Modeling : Scenario-Based Methods

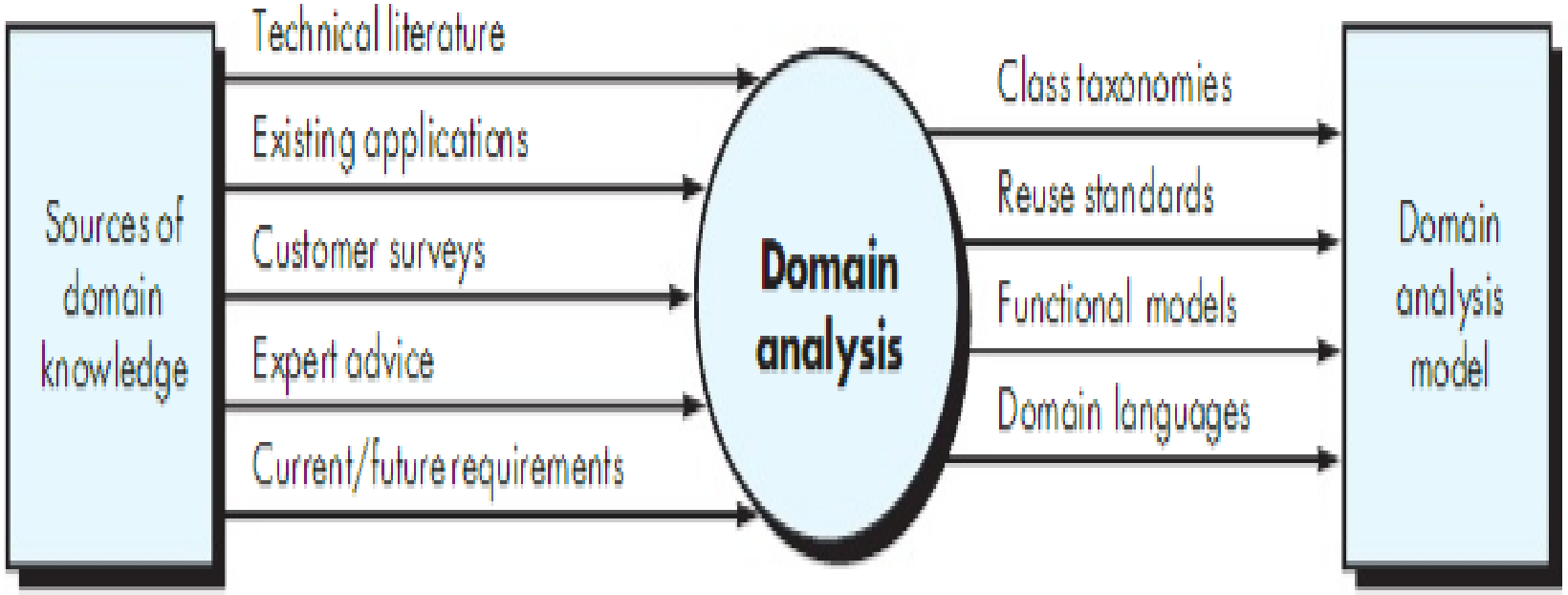
A Bridge



People
Innovation
Excellence

Requirement Modeling : Scenario-Based Methods

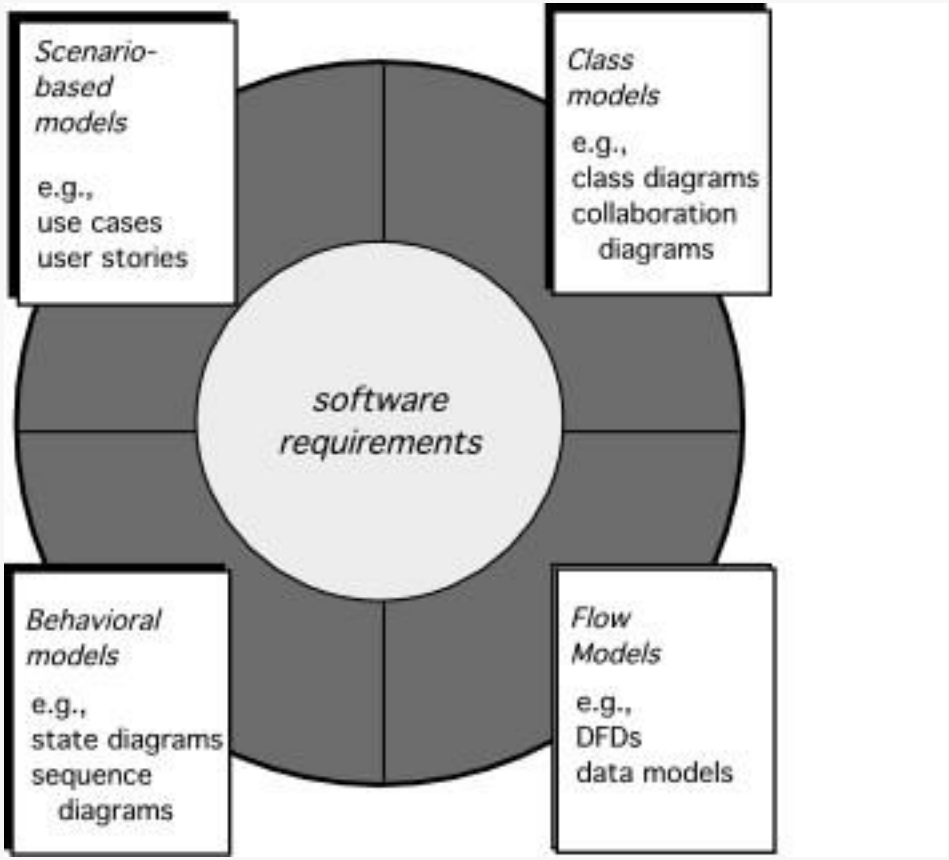
Domain analysis



People
Innovation
Excellence

Requirement Modeling : Scenario-Based Methods

Elements of Requirements Analysis



Requirement Modeling : Scenario-Based Methods

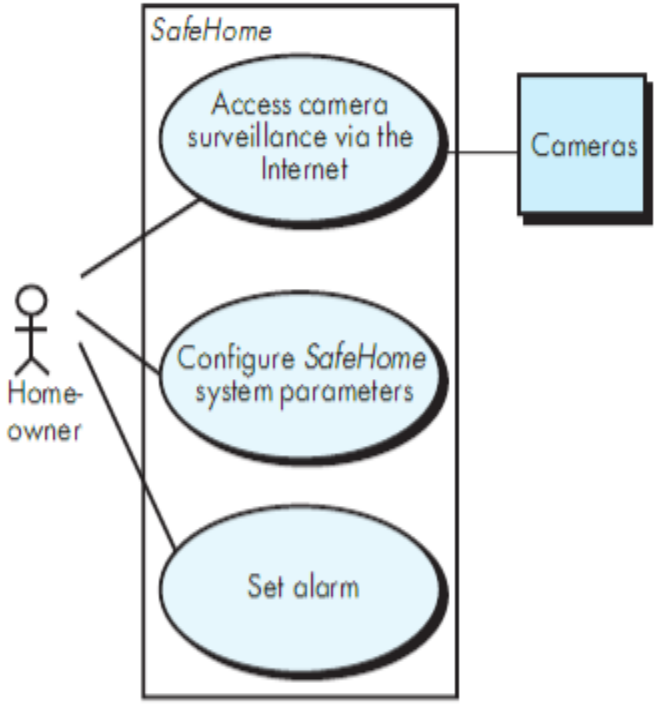
- Although the success of a computer-based system or product is measured in many ways, user satisfaction resides at the top of the list.
- If you understand how end users (and other actors) want to interact with a system, your software team will better able to properly characterize requirements and build meaningful analysis and design models
- Hence, requirement modeling with UML begins with the creation of scenarios in the form of use cases, activity diagrams, and swimlane diagrams

Requirement Modeling : Scenario-Based Methods

The steps of scenario-based modeling :

- Creating a preliminary use case
- Refining a preliminary use case
- Writing a formal use ca

Preliminary use case diagram for the SafeHome system

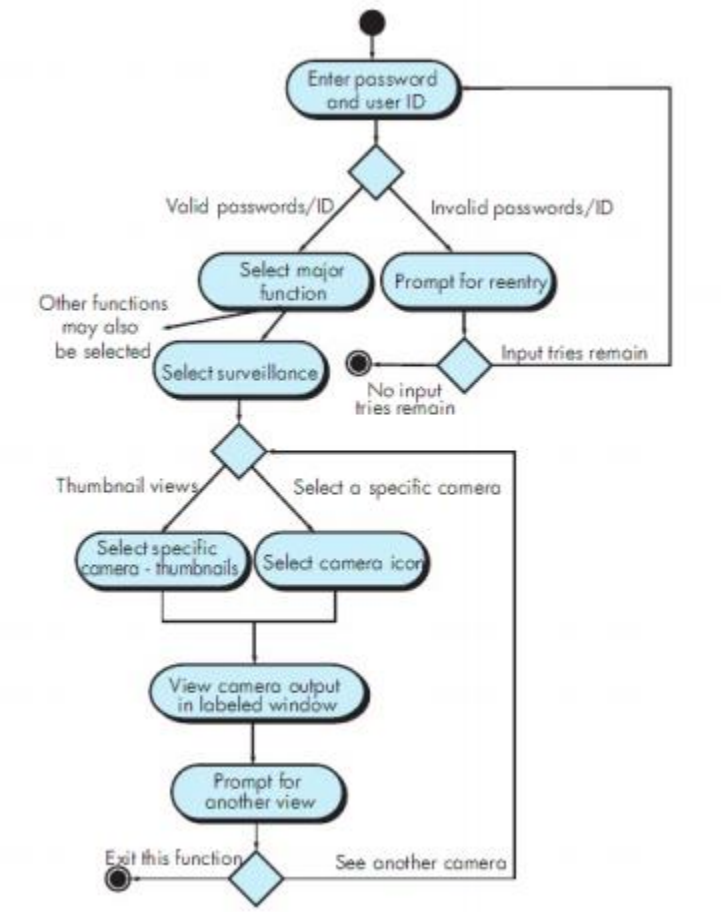


People
Innovation
Excellence

Requirement Modeling : Scenario-Based Methods

UML Model That Supplement the Use Case

- Developing an activity diagram

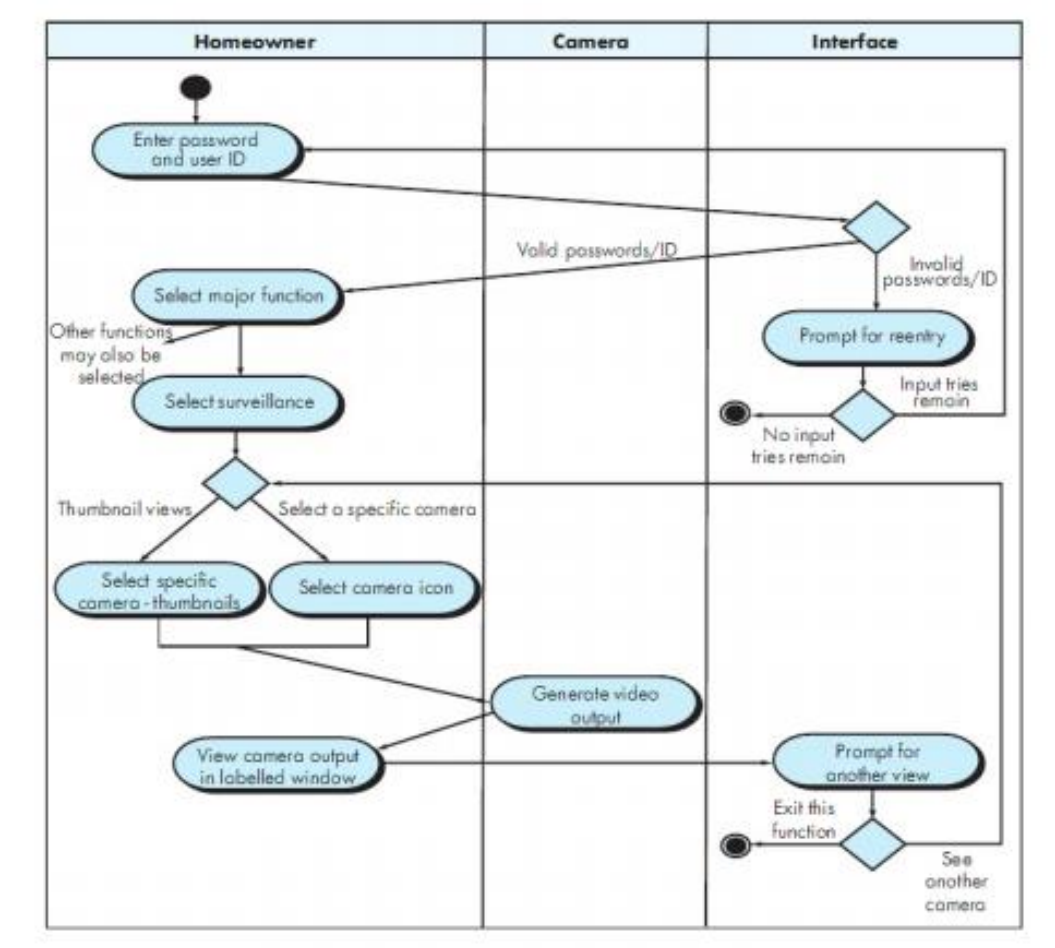


People
Innovation
Excellence

Requirement Modeling : Scenario-Based Methods

UML Model That Supplement the Use Case

- Swimlane Diagram



People
Innovation
Excellence

Requirement Modeling : Class-Based Methods

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

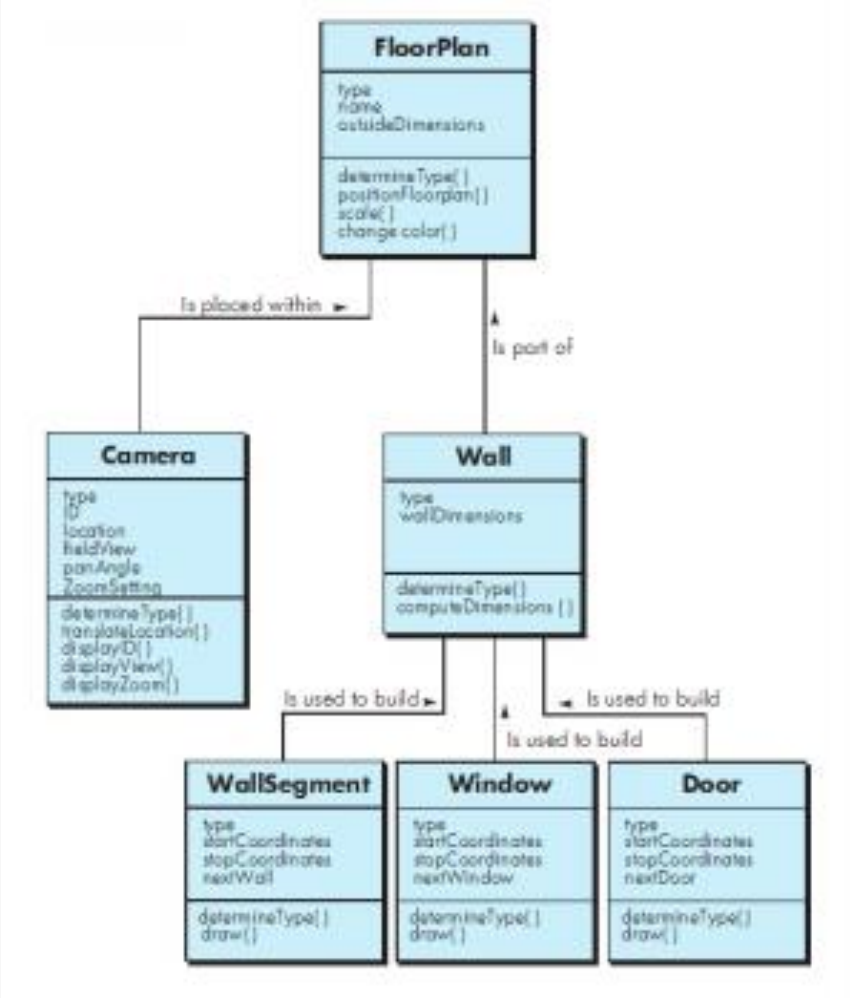
Requirement Modeling : Class-Based Methods

Identifying Analysis Classes

- **External entities** (e.g., other systems, devices, people) that produce or consume information to be used by computer-based system
- **Things** (e.g., reports, displays, letters, signals) that are part of the information domain for the problem
- **Occurances or events** (e.g., a property transfer or the completion of 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 relevantto an application
- **Places** (e.g., manufacturing floor or loading dock) that establish the context of the problem and the overall function of the system
- **Structures** (e.g., sensors, four-wheeled vehicles, or computers) that define a class of objects or related classes of objects

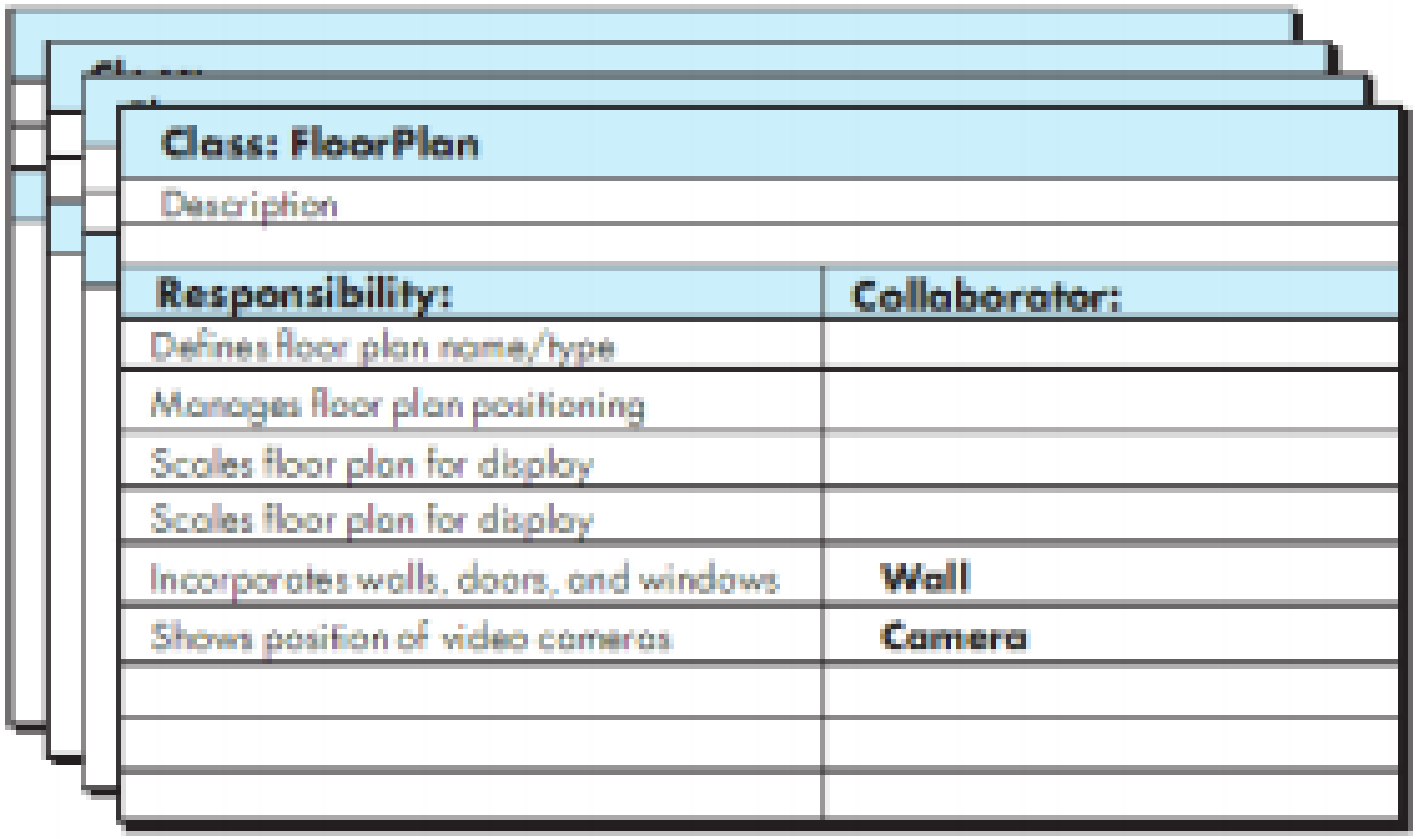
Requirement Modeling : Class-Based Methods

Class diagram FloorPlan



Requirement Modeling : Class-Based Methods

A CRC Model Index Card



People
Innovation
Excellence

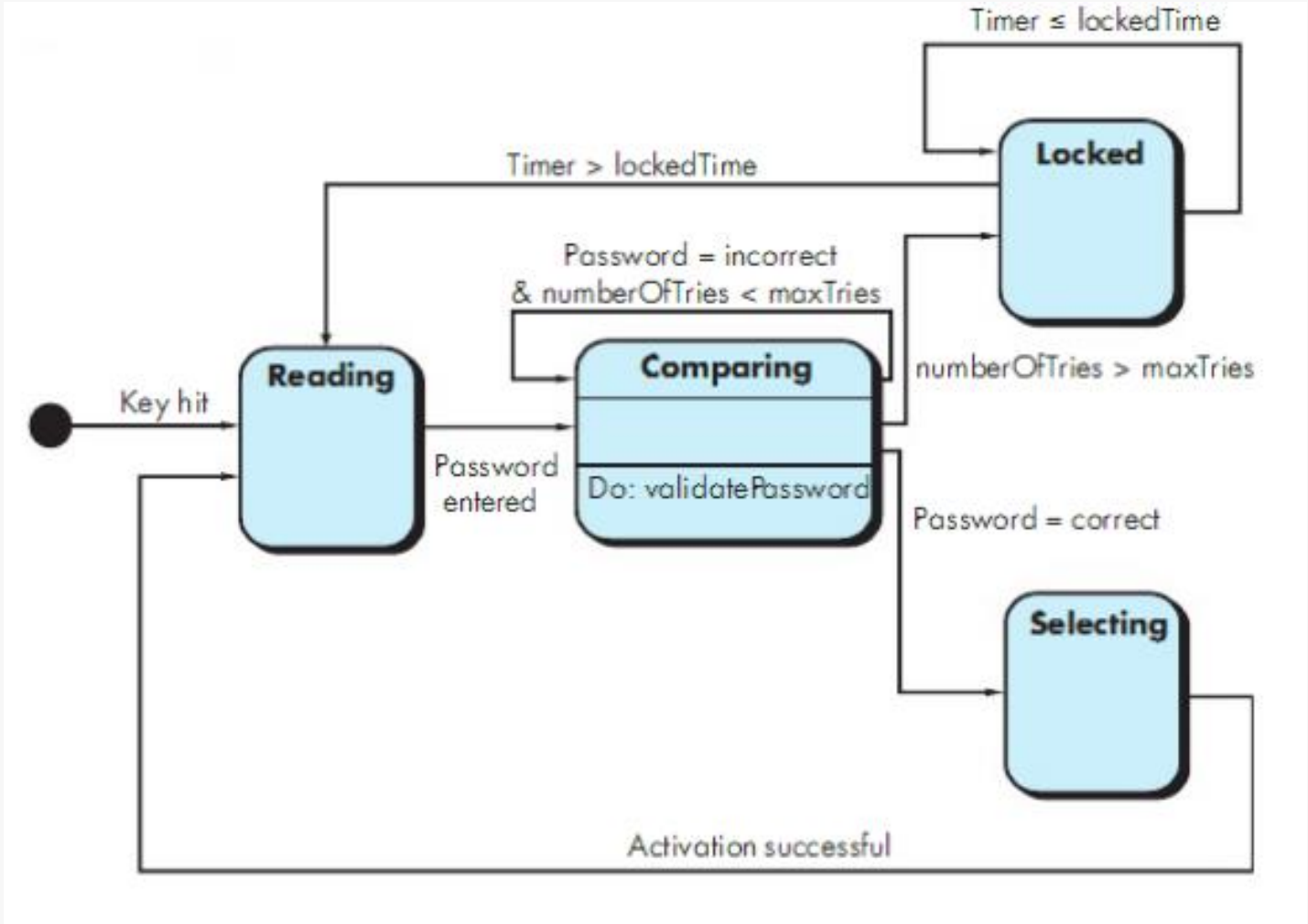
Requirement Modeling : Behavior, Patterns, and Web/Mobile Apps

Creating a Behvior Model

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

Requirement Modeling : Behavior, Patterns, and Web/Mobile Apps

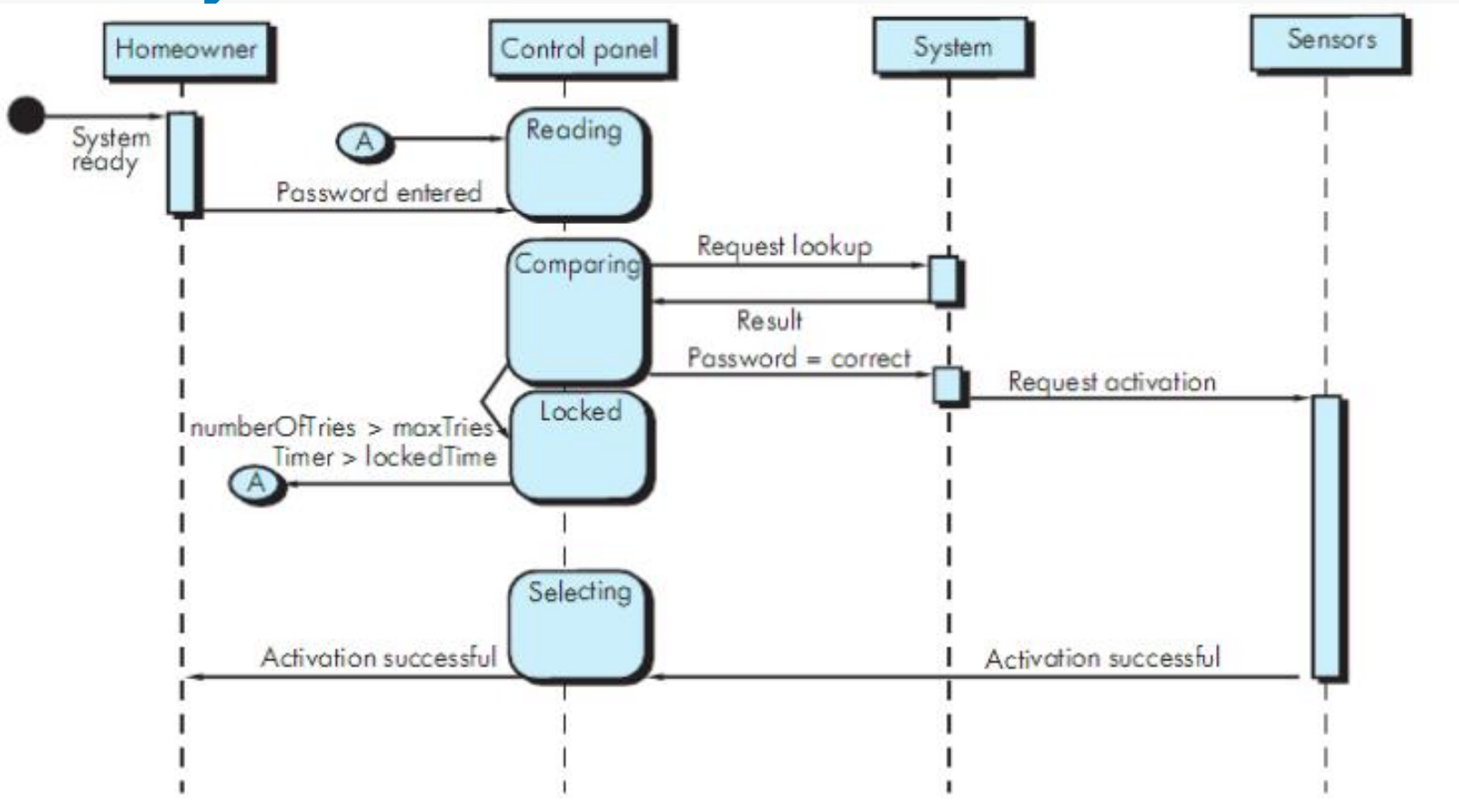
State diagram for the ControlPanel class



People
Innovation
Excellence

Requirement Modeling : Behavior, Patterns, and Web/Mobile Apps

Sequence diagram (partial) for the SafeHome security function



People
Innovation
Excellence

Patterns for Requirement Modeling

Pattern name: A descriptor that captures the essence of the pattern.

Intent: Describes what the pattern accomplishes or represents

Motivation: A scenario that illustrates how the pattern can be used to address the problem.

Forces and context: A description of external issues (forces) that can affect how the pattern is used and also the external issues that will be resolved when the pattern is applied.

Solution: A description of how the pattern is applied to solve the problem with an emphasis on structural and behavioral issues.

Consequences: Addresses what happens when the pattern is applied and what trade-offs exist during its application.

Design: Discusses how the analysis pattern can be achieved through the use of known design patterns.

Known uses: Examples of uses within actual systems.

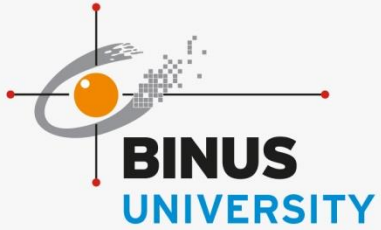
Related patterns: On e or more analysis patterns that are related to the named pattern because (1) it is commonly used with the named pattern; (2) it is structurally similar to the named pattern; (3) it is a variation of the named pattern.

Requirement Modeling for Web and Mobile Apps

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



Requirement Modeling for Web and Mobile Apps

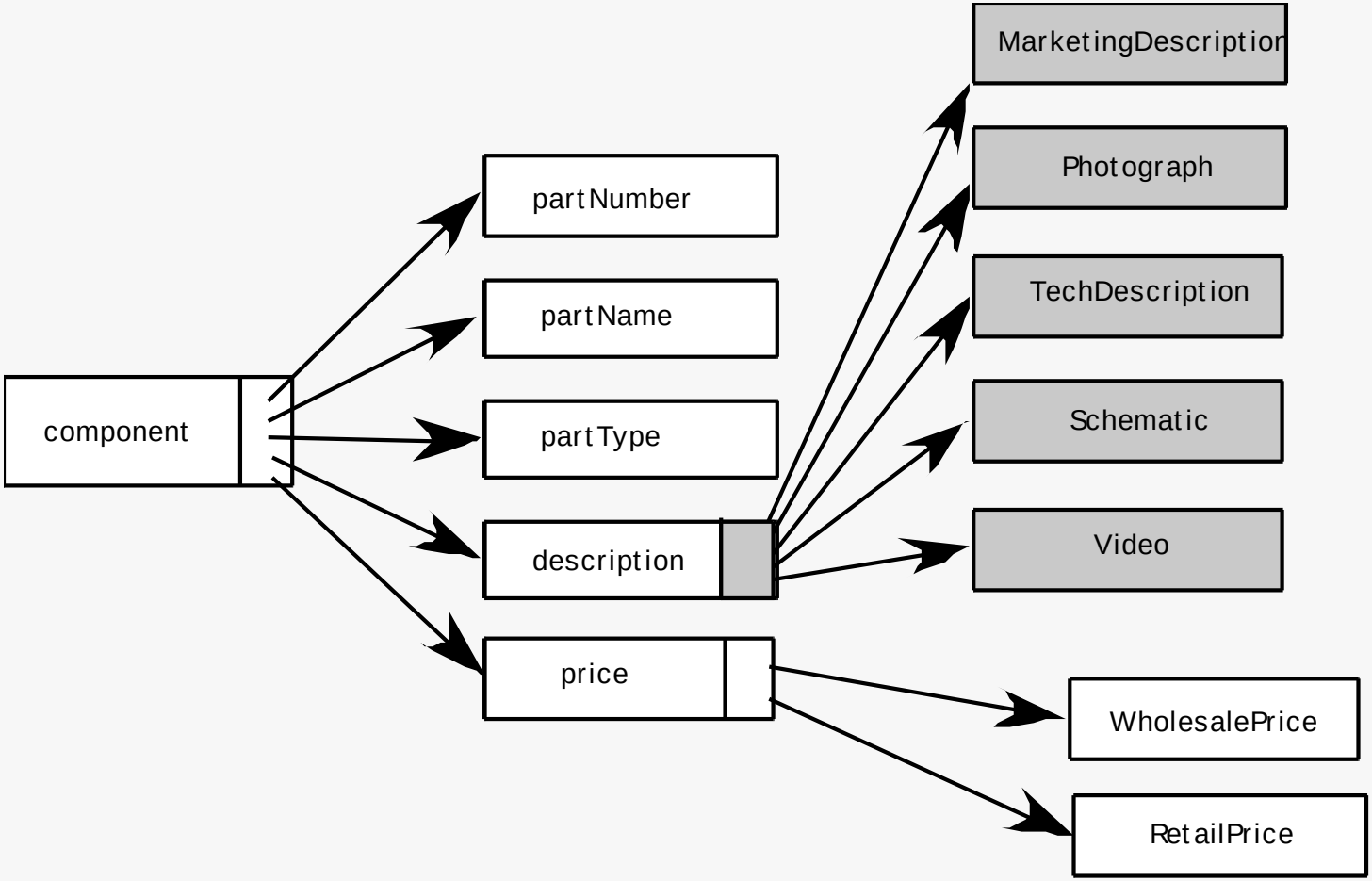
The Content Model

- 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

Requirement Modeling for Web and Mobile Apps

People
Innovation
Excellence

Data Tree



Requirement Modeling for Web and Mobile Apps

The Interaction Model

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

People
Innovation
Excellence

Requirement 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

Requirement Modeling for Web and Mobile Apps

People
Innovation
Excellence

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

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.
- **Requirements Engineering / Specification, http://www.youtube.com/watch?v=wEr6mwquPLY**
- **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

BINUS UNIVERSITY

Q & A

People
Innovation
Excellence