Course : COMP6100/Software Engineering

Effective Period : Desember 2017

Understanding Requirements

Session 08

People
Innovation
Excellence

Acknowledgement

People
Innovation
Excellence

Learning Objectives

LO 2 :   Explain the software engineering practices and business environment

Contents

- **Requirement Engineering**
- **Building the Analysis Model**
- **Analysis Patterns**
- **Negotiating Requirements**
- **Validating Requirements**

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**

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.

Inception

- **Identify stakeholders**
  - **"who else do you think I should talk to?"**
- **Recognize multiple points of view**
- **Work toward collaboration**
- **The first questions**
  - **Who is behind the request for this work?**
  - **Who will use the solution?**
  - **What will be the economic benefit of a successful solution**
 - **Is there another source for the solution that you need?**

Requirement Engineering

Eliciting Requirements

- **meetings are conducted and attended by both software engineers and customers**
- **rules for preparation and participation are established**
- **an agenda is suggested**
- **a "facilitator" (can be a customer, a developer, or an outsider) controls the meeting**
- **a "definition mechanism" (can be work sheets, flip charts, or wall stickers or an electronic bulletin board, chat room or virtual forum) is used**
- **the goal is**
  - **to identify the problem**
  - **propose elements of the solution**
  - **negotiate different approaches, and**
  - **specify a preliminary set of solution requirements**

Requirement Engineering

Eliciting Requirements
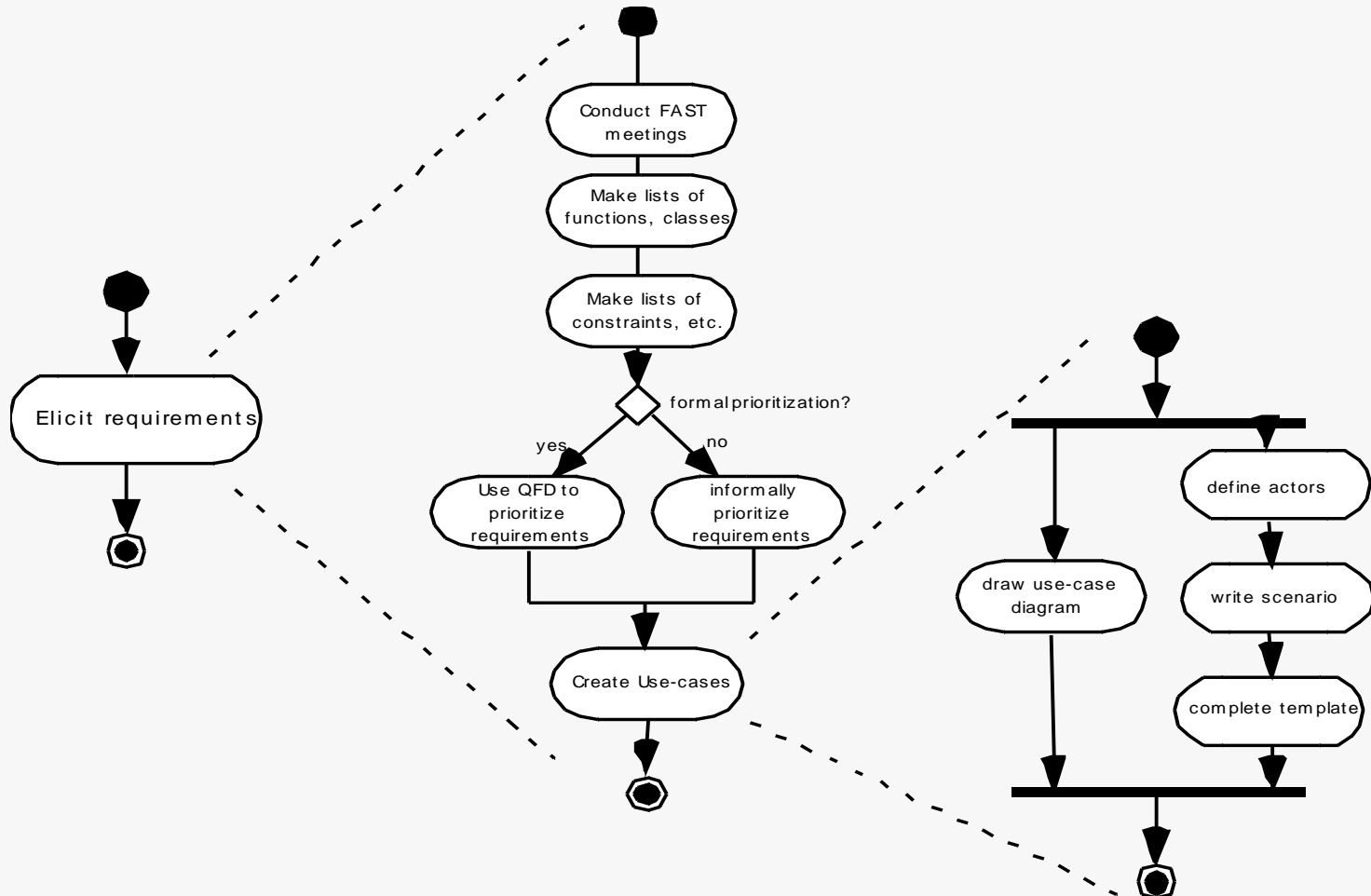
Quality Function Deployment

- **Function deployment determines the "value" (as perceived by the customer) of each function required of the system**

- **Information deployment identifies data objects and events**

- **Task deployment examines the behavior of the system**

- **Value analysis determines the relative priority of requirements**

Requirement Engineering

Elicitation Work Products

- a statement of need and feasibility.
- a bounded statement of scope for the system or product.
- a list of customers, users, and other stakeholders who participated in requirements elicitation
- a description of the system's technical environment.
- a list of requirements (preferably organized by function) and the domain constraints that apply to each.
- a set of usage scenarios that provide insight into the use of the system or product under different operating conditions.
- any prototypes developed to better define 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**
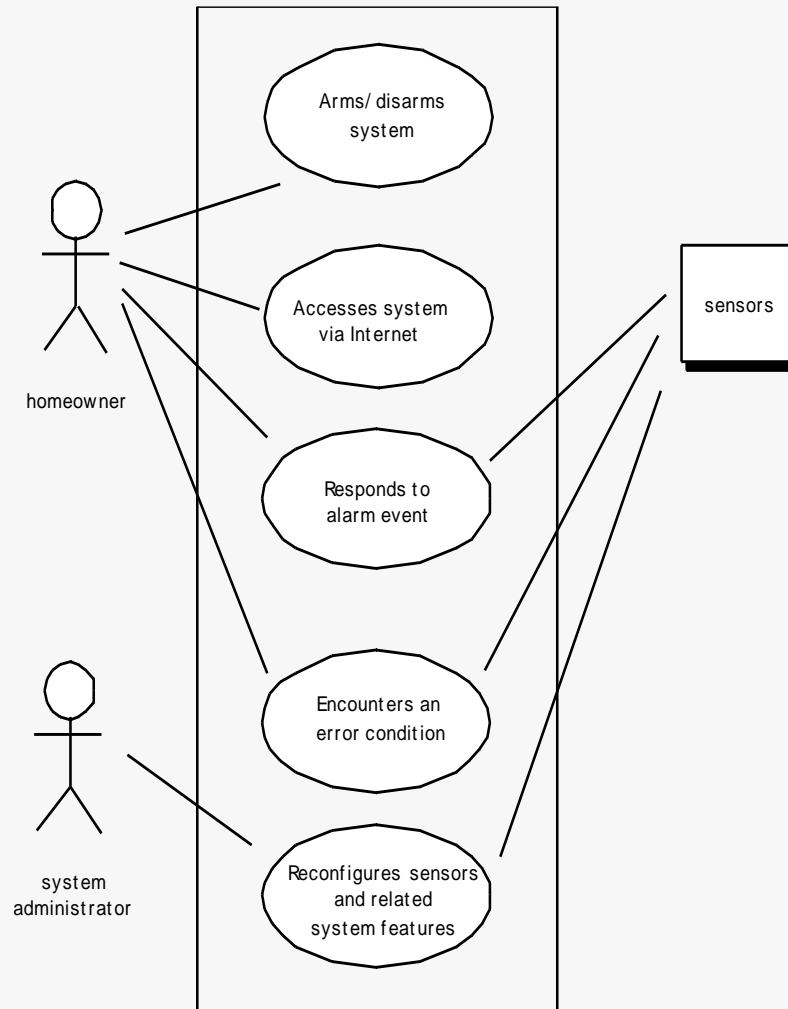- **Flow-oriented elements**
  - **Data flow diagram**

Building the Analysis Model

Use-Cases

- **A collection of user scenarios that describe the thread of usage of a system**
- **Each scenario is described from the point-of-view of an "actor"—a person or device that interacts with the software in some way**
- **Each scenario answers the following questions:**
  - **Who is the primary actor, the secondary actor (s)?**
  - **What are the actor's goals?**
  - **What preconditions should exist before the story begins?**
  - **What main tasks or functions are performed by the actor?**
  - **What extensions might be considered as the story is described?**
  - **What variations in the actor's interaction are possible?**
  - **What system information will the actor acquire, produce, or change?**
  - **Will the actor have to inform the system about changes in the external environment?**
  - **What information does the actor desire from the system?**
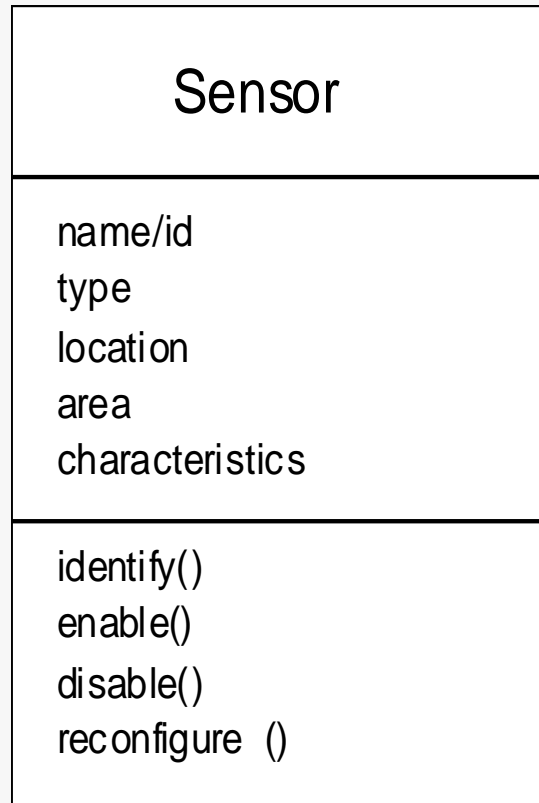 - **Does the actor wish to be informed about unexpected**

Use-Case Diagram

Building the Analysis Model

Class Diagram
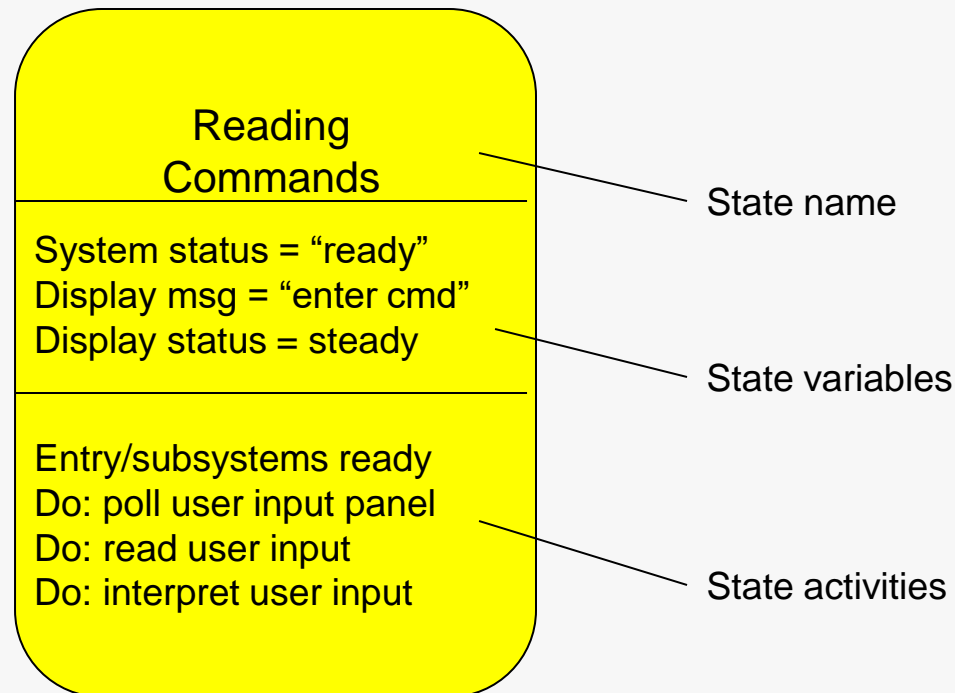
From the *SafeHome* system …

Sensor
name/id type location area characteristics
identify() enable() disable() reconfigure ()

People
Innovation
Excellence

State Diagram

Analysis Patterns

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

Negotiating Requirements

- **Identify the key stakeholders**
  - **These are the people who will be involved in the negotiation**
- **Determine each of the stakeholders "win conditions"**
  - **Win conditions are not always obvious**
- **Negotiate**
  - **Work toward a set of requirements that lead to "win-win"**

Validating Requirements

- **Is each requirement achievable in the technical environment that will house the system or product?**
- **Is each requirement testable, once implemented?**
- **Does the requirements model properly reflect the information, function and behavior of the system to be built.**
- **Has the requirements model been "partitioned" in a way that exposes progressively more detailed information about the system.**
- **Have requirements patterns been used to simplify the requirements model. Have all patterns been properly validated? Are all patterns consistent with customer requirements?**

People
Innovation
Excellence

Exercises

1. **Discuss some of the problems that occur when requirements must be elicited from three or four different customers**

2. **Develop a complete use case for one of the following**

   a. **Making a withdrawal at an ATM**

   b. **Using your charge card for a meal at a restaurant**

   c. **Buying a stock using an online brokerage account**

   d. **Searching for books (on a specific topic) using an on-line bookstore**

3. **Describe what an analysis pattern is in your own**

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**

- Collaborative Requirements Management, http://www.youtube.com/watch?v=tEXizjE05LA

People
Innovation
Excellence

Q & A