



FC-MD

Contents

- 1 Software Inspections
- 2 Software Reading Techniques
- 3 PBR Concepts
- 4 Requirements Defects
- 5 The „tester“ scenario
- 6 A tester scenario example
- 7 The „user“ scenario
- 8 A user scenario example

Info-Slide

Subtitle:
Speaker: Forrest Shull
Reviewer:
Notes:
Filename:
Version: 1.0

Slide 1



FC-MD

The “user” scenario

Contents

- 1 Software Inspections
- 2 Software Reading Techniques
- 3 PBR Concepts
- 4 Requirements Defects
- 5 The „tester“ scenario
- 6 A tester scenario example
- 7 The „user“ scenario
- 8 A user scenario example

- User needs to make sure requirements adequately capture the functionality needed in the system
- One strategy for handling functionality is **use cases**:
 - Use cases represent the users’ view of the system (not an implementation view).
 - A use case describes a particular set of system functionality by modeling the “dialogue” a user undertakes with the system.
 - A use case is based on a **descriptive scenario** of how the user interacts with the system. It identifies events that can occur and describes the system response.
 - A use case is a complete and meaningful flow of events, taken from the perspective of a particular user of the system.
 - Taken together, all use cases constitute all possible ways of using the system.

Slide 2



FC-MD

The "user" scenario

Contents

- 1 Software Inspections
- 2 Software Reading Techniques
- 3 PBR Concepts
- 4 Requirements Defects
- 5 The „tester“ scenario
- 6 A tester scenario example
- 7 The „user“ scenario
- 8 A user scenario example

- Use cases provide a view of the system that is very focused on **functionality**.
 - They provide a very understandable format for **communication**, especially between system designers and the customers of a system. They can help validate whether the system being described will work the way the customers expect it will.
 - They feed into **later stages of the lifecycle** by helping to identify objects, develop test plans, and develop documentation.
 - They are widely accepted by the OO community as a way to **supplement object modeling** during system analysis.

Slide 3

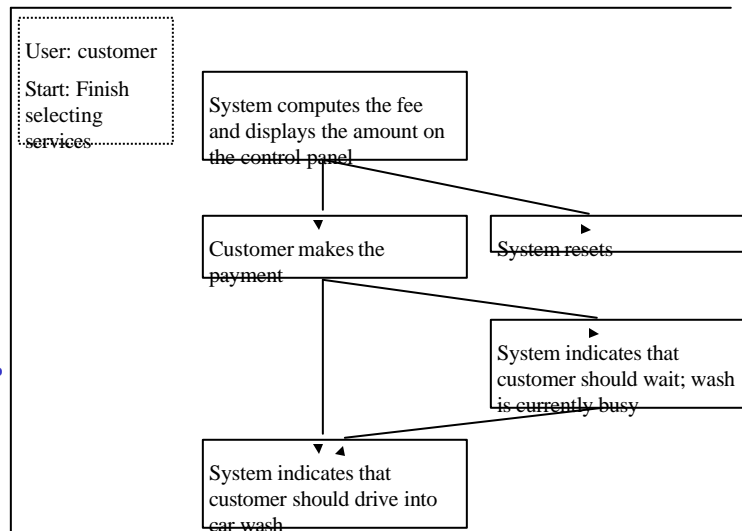


FC-MD

The "user" scenario

Contents

- 1 Software Inspections
- 2 Software Reading Techniques
- 3 PBR Concepts
- 4 Requirements Defects
- 5 The „tester“ scenario
- 6 A tester scenario example
- 7 The „user“ scenario
- 8 A user scenario example



Slide 4



FC-MD

The "user" scenario

Contents

- 1 Software Inspections
- 2 Software Reading Techniques
- 3 PBR Concepts
- 4 Requirements Defects
- 5 The „tester“ scenario
- 6 A tester scenario example
- 7 The „user“ scenario
- 8 A user scenario example

- **Use Cases May Help in Finding Requirements Defects**
- Use cases provide a different representation of the same functionality described in English-language requirements.
- Assumptions for defect detection:
 - If the English-language requirements are well-specified, we should be able to convert them into use cases relatively easily.
 - If there are hidden problems with the English-language requirements, they may prevent us from creating a clear, complete use case translation. These problems would probably also cause difficulties in constructing other artifacts based on the requirements, such as a design or code.
- The process of constructing use cases can be used as an aid in identifying defects.

Slide 5



FC-MD

The "user" scenario

Contents

- 1 Software Inspections
- 2 Software Reading Techniques
- 3 PBR Concepts
- 4 Requirements Defects
- 5 The „tester“ scenario
- 6 A tester scenario example
- 7 The „user“ scenario
- 8 A user scenario example

- First major component: participants
 - **entities** in the environment (i.e. **external** to the system)
 - that **interact with the system** to achieve some functionality.
 - (often known as *actors*).
- Participants can be: human users, other systems, external organizations, external devices, etc., which interact with the system.
- Some participants initiate events; others interact with the system as a result of other events.
- When users are involved: a participant is a role (i.e. a characteristic way of interacting with the system) NOT a specific person.

Slide 6



FC-MD

The "user" scenario

Contents

- 1 Software Inspections
- 2 Software Reading Techniques
- 3 PBR Concepts
- 4 Requirements Defects
- 5 The „tester“ scenario
- 6 A tester scenario example
- 7 The „user“ scenario
- 8 A user scenario example

- Second major component: Functionality
- More than anything else, use cases are concerned with describing system behavior.
- Problem: Very hard to specify an exact definition for the intuitive way in which we usually describe system functionality.
- Lot of variation in how formally use case concepts are applied to describing a system: very loosely in some situations, very rigorously in others.
- We ask for functionality to be described at 2 levels:
 - product functions (What features does the product have? What are the specific activities it can do?)
 - use cases/user scenarios (What can customers use the system for?)

Slide 7



FC-MD

The "user" scenario

Contents

- 1 Software Inspections
- 2 Software Reading Techniques
- 3 PBR Concepts
- 4 Requirements Defects
- 5 The „tester“ scenario
- 6 A tester scenario example
- 7 The „user“ scenario
- 8 A user scenario example

- **Step 1**
 - **General instructions:** Identify the participants in the requirements.
 - **Specific instructions:** (guidelines for recognizing participants)
 - **Questions:**
 - Q1.1 Are multiple terms used to describe the same participant in the requirements?
 - Q1.2 Is the description of how the system interacts with a participant inconsistent with the description of the participant? Are the requirements unclear or inconsistent about this interaction?
 - Q1.3 Have necessary participants been omitted? That is, does the system need to interact with another system, a piece of hardware, or a type of user that is not described?
 - Q1.4 Is an external system or a class of "users" described in the requirements which does not actually interact with the system?

Slide 8



FC-MD

The "user" scenario

Contents

- 1 Software Inspections
- 2 Software Reading Techniques
- 3 PBR Concepts
- 4 Requirements Defects
- 5 The „tester“ scenario
- 6 A tester scenario example
- 7 The „user“ scenario
- 8 A user scenario example

• Step 2.

- **General goal:** Describe system functionality.
- **Specific goal:** (Procedure for identifying specific system product functions and understanding how they contribute to use cases)
- **Questions:**
 - Q2.1 Are the start conditions for each use case specified at an appropriate level of detail?
 - Q2.2 Are the class(es) of users who use the functionality described, and are these classes correct and appropriate?
 - Q2.3 Is there any system functionality that should be included in a use case but is described in insufficient detail or omitted from the requirements?
 - Q2.4 Has the system been described sufficiently so that you understand what activities are required for the user to achieve the goal of a use case? Does the description allow more than one interpretation as to how the system achieves this goal?
 - Q2.5 Do the requirements omit use cases that you feel are necessary, according to your domain knowledge or the general description?

Slide 11

Form A - Participants and use cases in new requirements

Reviewer Name: _____		Document Reviewed: _____	
Participants	Involvement in which use cases?	System Functionalities	
1. <u>Credit card reader</u>	<input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/>	F1. <u>Update inventory</u>	_____
2. <u>Credit card system</u>	<input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/>	F2. <u>Bill at purchase</u>	_____
3. <u>Gas pump</u>	<input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/>	F3. <u>Bill on monthly bill</u>	_____
4. <u>Gas pump interface</u>	<input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/>	F4. <u>Accept credit card payment</u>	_____
5. <u>Cashier interface</u>	<input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/>	F5. <u>Accept cash payment</u>	_____
6. <u>Customer</u>	<input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/>	F6. <u>Receive monthly bill</u>	_____
7. <u>Cashier</u>	<input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/>	F7. _____	_____
8. _____	<input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/>	F8. _____	_____
9. _____	<input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/>	F9. _____	_____
10. _____	<input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/>	F10. _____	_____
11. _____	<input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/>	F11. _____	_____
12. _____	<input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/>	F12. _____	_____
13. _____	<input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/>	F13. _____	_____
14. _____	<input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/>	F14. _____	_____

Form B - Use Case

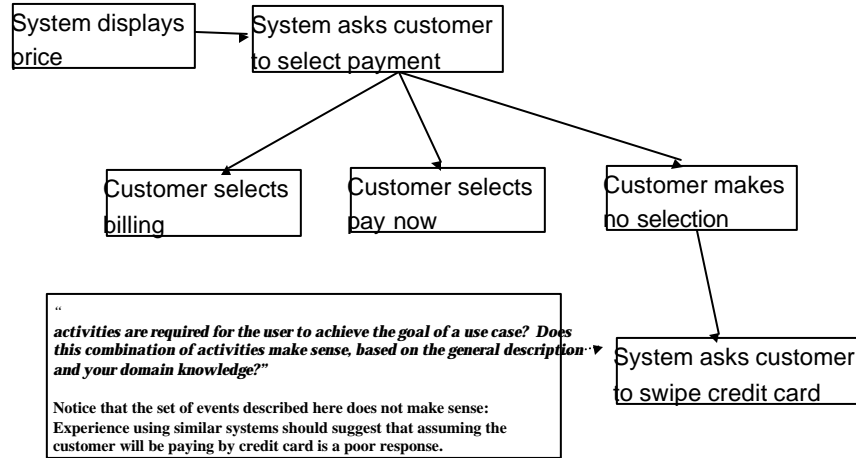
Reviewer Name: _____ Document Reviewed: _____

Use Case Name: Pay for gas Use Case Number: 1

User: Customer Involves which functionalities?

2	3	4	5	
---	---	---	---	--

Start: Replace nozzle



FC-MD

The “user” scenario

Contents

- 1 Software Inspections
- 2 Software Reading Techniques
- 3 PBR Concepts
- 4 Requirements Defects
- 5 The „tester“ scenario
- 6 A tester scenario example
- 7 The „user“ scenario
- 8 A user scenario example

• Step 3.

- **General goal:** Cross-index participants and functionality.
- **Specific goal:** (Procedure for helping identify which participants are involved in which system functionalities)
- **Questions:**
 - Q3.1 Is it clear from the requirements which participants are involved in which use cases?
 - Q3.2 Based on the general requirements and your knowledge of the domain, has each participant been connected to all of the relevant use cases?
 - Q3.3 Are participants involved in use cases that are incompatible with the participant's description?
 - Q3.4 Have necessary participants been omitted (e.g., are there use cases which require information that cannot be obtained from any source described in the requirements)?

Form A - Participants and use cases in new requirements

Reviewer Name: _____

Document Reviewed: _____

Participants		Involvement in which use cases?	System Functionalities	
1. Credit card reader	_____	1	F1. Update inventory	_____
2. Credit card system	_____	1	F2. Bill at purchase	_____
3. Gas pump	_____		F3. Bill on monthly bill	_____
4. Gas pump interface	_____	1	F4. Accept credit card payment	_____
5. Cashier interface	_____	1	F5. Accept cash payment	_____
6. Customer	_____	1	F6. Receive monthly bill	_____
7. Cashier	_____	1	F7. _____	_____
8. _____	_____		F8. _____	_____
9. _____	_____		F9. _____	_____
10. _____	_____		F10. _____	_____
11. _____	_____		F11. _____	_____
12. _____	_____		F12. _____	_____
13. _____	_____		F13. _____	_____
14. _____	_____		F14. _____	_____