



# SWEN90016

## Software Processes and Management

### TUTORIAL 8

Software Quality Management

Semester 2, 2023





# Aim of the Tutorial

## What we intend to achieve at the end of today's tutorial?

Quality Management – Recap Quizzes

Quality Management in Agile

Activity 1 & 2 – Quality Management in Agile

Activity 3 – Quality Management using Formal SDLC

The learning outcomes from this tutorial may be a subject of discussion during the Q&A session for your Assignment 2 PMP presentation in Week 12.

Research Survey Reminder

🌐 When poll is active, respond at **pollev.com/rajesh**

📱 Text **RAJESH** to **+61 480 025 509** once to join

# Which of the following is true with respect to Quality Management

Relative cost of bug fix higher in Coding Phase than in Production

Cost of bug fix is the same irrespective of the project phase

✓ Relative cost of bug fix in Production very high than the same in Coding Phase

Relative cost of bug fix the same in Requirements and Design Phase

None of the above

Powered by  **Poll Everywhere**

Start the presentation to see live content. For screen share software, share the entire screen. Get help at [pollev.com/app](https://pollev.com/app)

🌐 When poll is active, respond at **pollev.com/rajesh**

📱 Text **RAJESH** to **+61 480 025 509** once to join

# Which of the following are Process Standards in Quality Management?



Design Review, Design Validation, Project Plan Approval, Change Control Process

Design Review, Design Validation, Requirements Document, Test Recording

Design Review Template, Coding Standards, Change Control Template

Requirements Specification, Design Process, Testing Process, Release Template

None of the above

Powered by  **Poll Everywhere**

Start the presentation to see live content. For screen share software, share the entire screen. Get help at [pollev.com/app](https://pollev.com/app)

🌐 When poll is active, respond at **pollev.com/rajesh**

📱 Text **RAJESH** to **+61 480 025 509** once to join

# The Capability Maturity Model (CMM)

Developed by Software Engineering Institute  
(SEI) at Carnegie Mellon University

Describes an approach for S/W companies to  
move from ad-hoc to mature process(es)

Organizations are characterized being at Level 1  
to Level 5, based on the processes they follow

☒ All of the above

☐ None of the above

Powered by  **Poll Everywhere**

Start the presentation to see live content. For screen share software, share the entire screen. Get help at [pollev.com/app](https://pollev.com/app)

# QA / Testing in Agile

**User Story:** describes the requirement

**Acceptance Criteria:** provides the definition of when User Story is 'Done' from a end user perspective

- Every new feature is tested during the sprint.
- Testers & developers work closely together. Testing is done by the whole team.
- Every sprint has its own User Acceptance testing phase (Sprint Review).
- A small piece of working software is delivered to the client at the end of the sprint.
- Client does User Acceptance test.



Image by [Freepik](#)

Agile Testing Methodology - Methods, Principles, and Advantages

# QA / Testing in Agile

**User Story:** As a Customer I want to be able to split my payments so that I can pay using multiple debit cards.

## Test Scenarios:

- User selects 'split payment' on the payment page
- User chooses different types of payment options
- User specifies the amount they want split to each option
- System automatically calculates if the total amount paid by the split payments is the correct total payment

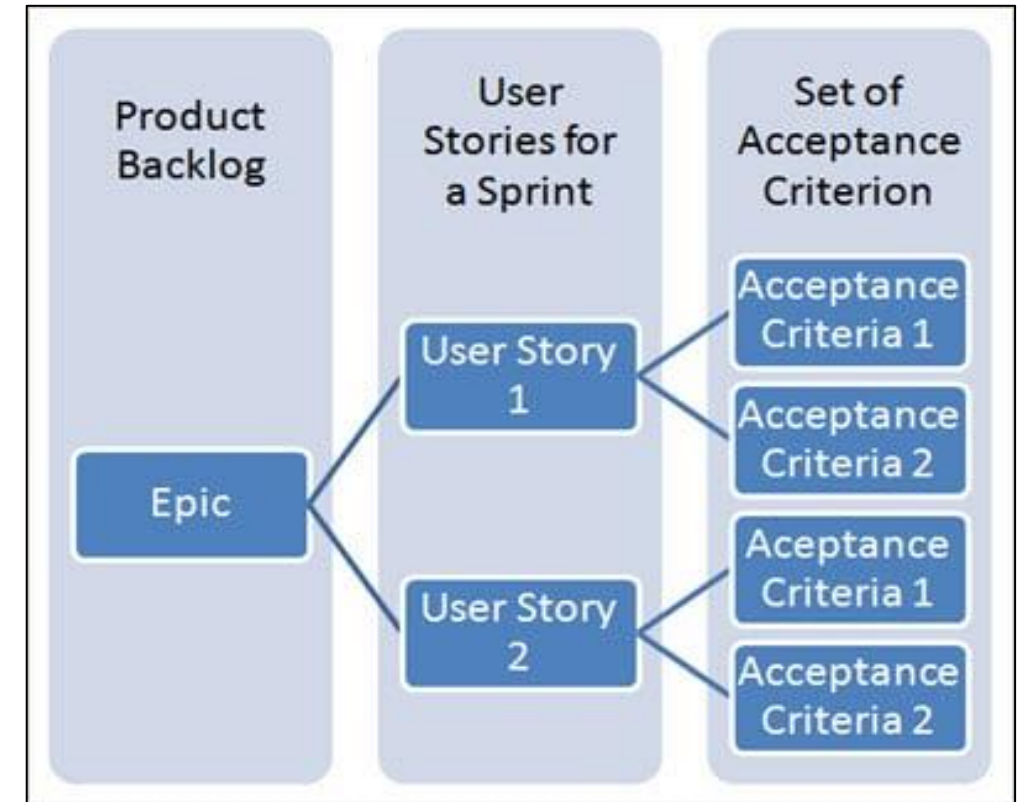
## Acceptance Criteria:

Each scenario can have multiple Acceptance Criteria



Image by [Freepik](#)

## User Story Acceptance Criteria



# QA / Testing in Agile – BDD Syntax

**SWEN90009:** Acceptance Criteria use the Behaviour Driven Development (BDD) Syntax

- **Given** a User wants to pay, **When** they click the 'split payment' button on the payment page, **then** multiple payment card options are displayed.
- **Given** a User wants to split a payment, **When** they select multiple payment card options, **then** the selected cards allow an amount to be specified.
- **Given** a User wants to confirm a split payment, **When** they specify the amount for each card, **then** the payments page verify if total payment is correct amount
- **Given** a User wants to receipt a split payment, **When** the payments page processes each payment card successfully, **then** the user knows payment is done



Image by [Freepik](#)

Writing BDD test cases in Agile Development



# QA / Testing in Agile

## Examples:

- **Given** a User wants to pay, **When** they click the 'split payment' button on the payment page, **then** multiple payment card options are displayed.
- **Given** a User wants to split a payment, **When** they select multiple payment card options, **then** the selected cards allow an amount to be specified.
- **Given** a User wants to confirm a split payment, **When** they specify the amount for each card, **then** the payments page verify if total payment is correct amount
- **Given** a User wants to receipt a split payment, **When** the payments page processes each payment card successfully, **then** the user knows payment is done

# QA in Agile – Multiple Acceptance Scenario

**User story:** As a user, I want to be able to recover the password to my account, so that I will be able to access my account in case I forgot the password.

**Scenario:** Forgot password

**Given:** The user has navigated to the login page

**When:** The user selected *forgot password* option

**And:** Entered a valid email to receive a link for password recovery

**Then:** The system sent the link to the entered email

**Given:** The user received the link via the email

**When:** The user navigated through the link received in the email

**Then:** The system enables the user to set a new password

User Story Acceptance Criteria



Image by [Freepik](#)



# Team Activity – 1

## User story:

As a temporal done surgeon, I want to be able to save a 3-D Visible Ear Simulator model, so that I will be able to retrieve and continue the simulation of the model at a later time.

**Scenario:** Save a 3-D Visible Ear Simulator Model to the database

**Formulate a suitable Acceptance Criteria for this User Story**



Image by Freepik



## Team Activity – 2

### User story:

As a temporal done surgeon, I want the system to meet the strict performance criteria for the saved 3-D model to load in less than 10 seconds, so that I can retrieve and continue with the simulation of the model.

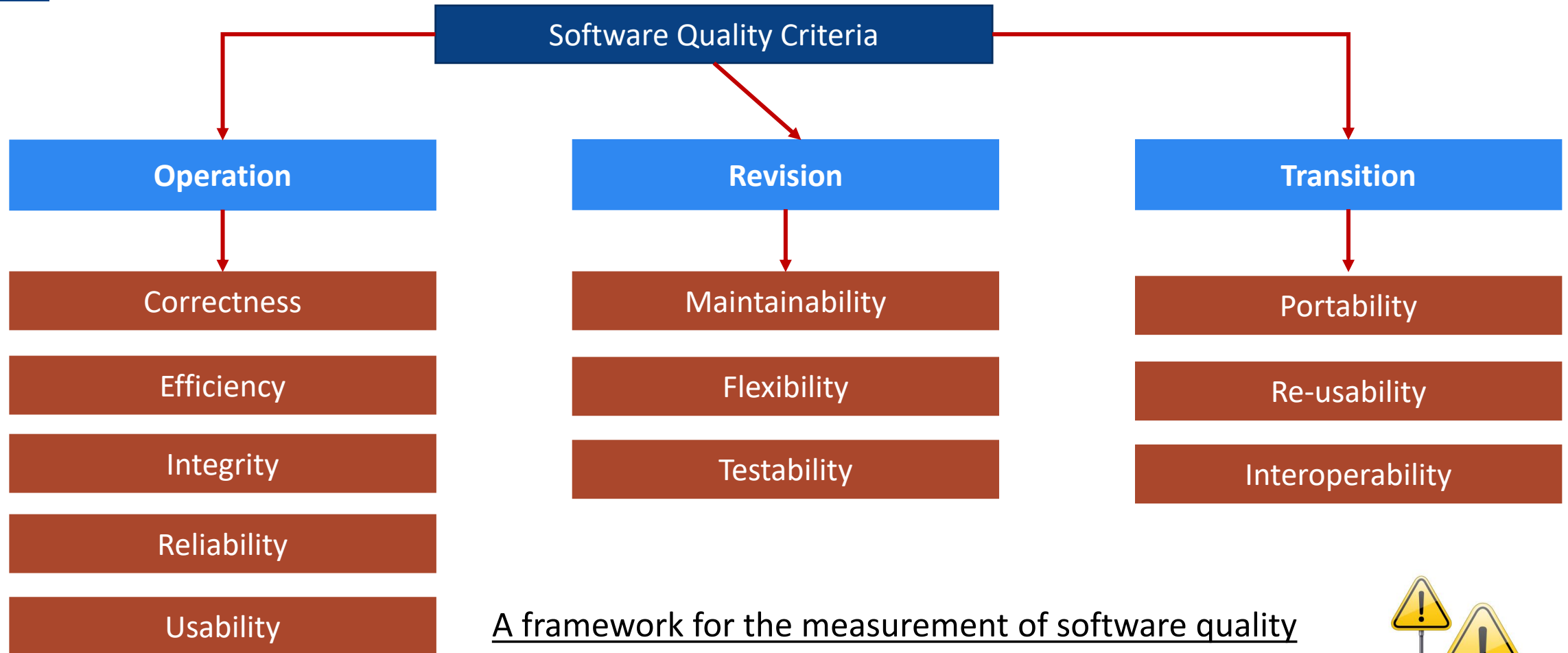
**Scenario:** Restore a saved 3-D Visible Ear Simulator Model from the database within 10 seconds

**Formulate a suitable Acceptance Criteria for this User Story**



Image by [Freepik](#)

# Quality Assurance: Formal SDLC Approach



A framework for the measurement of software quality

McCall's Quality Model



Image by [Freepik](#)

# Quality Plan – Checklist 1

Quality Attributes	Definition According to McCall's Quality Model
Correctness	The extent to which a program satisfies its specifications and fulfils the user's mission objectives
Reliability	The extent to which a program can be expected to perform its intended function with required precision.
Efficiency	The amount of computing resources and code required by a program to perform a given function.
Integrity	The extent to which access to software or data by unauthorized persons can be controlled.
Usability	The effort required to learn, operate, prepare input, and interpret output of a program.
Maintainability	The effort required to locate and fix an error in an operational program.
Testability	The effort required to test a program to ensure that it performs its intended function.
Flexibility	The effort required to modify an operational program.
Portability	The effort required to transfer a program from hardware and/or software environment to another.
Reusability	The extent to which a program (or parts thereof) can be reused in other applications.
Interoperability	The effort require to couple one system with another.

# Quality Plan – Checklist 2

## Checklist for Software Requirements specification artefact

### Organization and Completeness

- Are all internal cross references to other requirements, correct?
- Are all requirements written at a consistent and appropriate level of detail?
- Do the requirements provide an adequate basis for design?
- Is the implementation priority of each requirement included?
- Are all external hardware, software, and communication interfaces defined?
- Have algorithms intrinsic to the functional requirements been defined?
- Does the specification include all the known customer or system needs?
- Is the expected behaviour documented for all anticipated error conditions?

### Correctness

- Do any requirements conflict with or duplicate other requirements?
- Is each requirement written in clear, concise, unambiguous language?
- Is each requirement verifiable by testing, demonstration, review, or analysis?
- Is each requirement in scope for the project?
- Is each requirement free from content and grammatical errors?



# Quality Plan – Checklist 2 (CONTD..)

## Checklist for Software Requirements specification artefact

### Correctness

- Is any necessary information missing from a requirement? If so, is it identified as “to be decided”?
- Can all the requirements be implemented within known constraints?
- Are any specified error messages unique and meaningful?

### Quality Attributes

- Are all performance objectives properly specified?
- Are all security and safety considerations properly specified?
- Are all pertinent quality attribute goals explicitly documented and quantified, with the acceptable trade-offs specified?

**AND SO ON.....**





# Team Activity – 2

## Task 1:

Which of the items in Quality Checklist 1 and Quality Checklist 2 would be applicable for your PMP / I.T. Product being developed in your group assignment? Identify at-least 5 items from Checklist 1 and 5 items from Checklist 2

## Task 2:

Document your arguments for each of the importance of each of the items that you have identified in Task 1 as an indicator of a Formal SDLC Quality Checklist for the PMP, as well the I.T. Product

## Task 3:

Present at-least 3 custom quality indicators for your PMP submission in Week 11. Argue why are these custom quality indicators important for the PMP.



Image by [Freepik](#)



# Survey Reminder.....

Thank you!





THE UNIVERSITY OF  
MELBOURNE

# Thank you

**SWEN90016 – Software Processes and  
Management**  
**Semester 2, 2023**

---

Faculty of Engineering and Information Technology  
School of Computing and Information Systems