

©2016 Capgemini. All rights reserved.

The information contained in this document is proprietary and confidential. For Capgemini only.

Course Goals and Non Goals



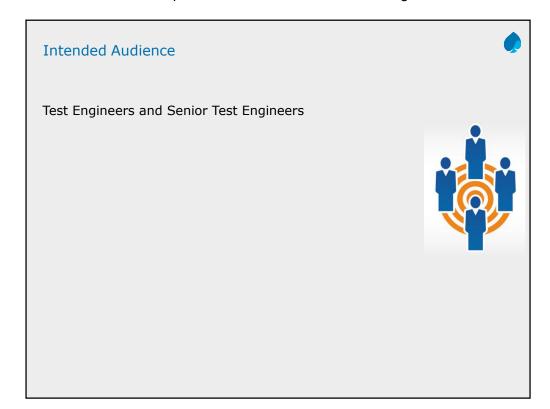
Course Goals

- At the end of this course participants gain an understanding of what is requirement and what is requirement engineering
- Differentiate between two types of requirements Functional Requirement and Non Functional requirements
- What is the importance of requirements in projects?
- What are the characteristics of good requirements?
- What is Requirement Itemization?
- How to use different templates to perform requirement itemization
- With the help of a case study?
- How to manage and successfully execute change in the requirement during Project life cycle?
- What is Requirement Traceability?

Course Non Goals

This course does not cover tools training to manage requirements

Pre-requisites Fundamental knowledge of Software Testing Different types of Software Testing techniques Different types of Software Testing



Day Wise Schedule



Day 1

- Lesson 1: Introduction to Requirements Engineering
- Lesson 2: Evolution and Types of Requirements
- Lesson 3: Requirements Itemization

Day 2

- Lesson 3: Requirements Itemization (Cont.)
- Lesson 4: Requirements Gathering Challenges & Techniques

Day 3

- Lesson 4: Requirements Gathering Challenges & Techniques (Cont.)
- Lesson 5: Requirement Management



Lesson 1: Introduction to Requirements Engineering

- 1.1 Requirements Engineering and Projects failure
- 1.2. Why do Projects fail? Current Survey
- 1.3 Requirements Contains Defects
- 1.4 Importance of Requirements Some statistics!
- 1.5 Why do we need good requirements?
- 1.6 What is a Requirement?
- 1.7 Requirement Definition
- 1.8 Why are Requirements important?
- 1.9 Requirements Engineering



Lesson 2: Evolution and Type of Requirements

- 1.1 Evolution of Requirements
- 1.2 Who provides the Requirements?
- 1.3 Types of Requirements
- 1.4 Functional Vs Non-Functional Requirements
- 1.5 Do not overlook the "Non-Functional Requirements"!
- 1.6 Non Functional Requirements: FURPS +
- 1.7 Other Non Functional Requirements: "+"
- 1.8 What is a good software requirement?



Lesson 3: Requirements Itemization

- 1.1 Requirements Itemization How?
- 1.2 Application Background
- 1.3 Assumptions
- 1.4 Explicit requirements
- 1.5 Implicit requirements
- 1.6 Interface Requirements
- 1.7 Requirements Analysis to Test Scenarios
- 1.8 Testable Items
- 1.9 Non Testable Items
- 1.10 Error Conditions
- 1.11 Application Invocation
- 1.12 Application Termination

Lesson 3: Requirements Itemization (Cont.)

- 1.13 File Handling
- 1.14 Requirement Prioritization
- 1.15 Case Study Equipment Tracking System
- 1.16 Template 1 Equipment tracking System
- 1.17 Template 2 Equipment tracking System



Lesson 4: Requirements Gathering - Challenges & Techniques

- 1.1 Requirements Gathering A Typical Illustration
- 1.2 Requirement Gathering Patterns
- 1.3 Challenges in Requirements
- 1.4 Challenges Clarity of requirements
- 1.5 Challenges Communication
- 1.6 Ambiguity From a Requirements Perspective Pitfalls of the English language
- 1.7 Ambiguity and Pitfalls
- 1.8 Ambiguity Checklist
- 1.9 Ambiguity Review
- 1.10 Requirement Gathering Skills Required
- 1.11 Tips to Requirement Gathering
- 1.12 Identification and Verification of Requirements



Lesson 5: Requirement Management

- 1.1 What is Requirements Management?
- 1.2 Why do requirements change?
- 1.3 Stable and Volatile Requirements
- 1.4 Requirements Classification
- 1.5 Baselining Requirements
- 1.6 Requirements Traceability
- 1.7 Types of Requirements Traceability
- 1.8 Requirement Traceability Matrix
- 1.9 Maintaining Requirement Traceability
- 1.10 Requirement Traceability Matrix Simple Example
- 1.11 Change Management
- 1.12 Change Request Management
- 1.13 Change Management Process
- 1.14 Define Requirement Creep
- 1.15 Why does Requirement Creep occur?
- 1.16 Measures to control Requirement Creep
- 1.17 Requirement Metrics

References Student material: Class Book (presentation slides with notes)

