



Learning Objectives of CP-CCT

“Knowledge with experience is power; certification is just a by-product”

What is CP-CCT?

CP-CCT stands for “Certified Professional - Cucumber driven Continuous Testing” certification prepared and honored by “Agile Testing Alliance” & “University Teknologi Malaysia (UTM)”.

You should have basic knowledge of Java. CP-MAT OR CP-BAT is the prerequisite for this continuous testing level certification called CP-CCT . Alternatively you should have Certified ScrumMaster® (CSM) or PMI Agile Certified Practitioner (PMI-ACP)® or Certified professional Devops Foundation (CP-DOF) or similar basic Agile / DevOps training. In absence of this you should be having at least 1 year of proven relevant experience in Agile/DevOps or Testing.

The course is applicable for all roles and not just "testers". Knowledge, experience & certification is consciously designed to focus on "Practical Continuous Testing using Cucumber and BDD" and is not limited to any specific role like "agile testers" or "DevOps testers"



How is it useful?

CP-CCT helps you in learning the collaborative methodology of BDD using Cucumber as tool. The course covers creating executable specifications on API and GUI layer. BDD can only achieve its full potential when it is combined with DevOps culture of Continuous Delivery. Cucumber with Jenkins helps in establishing the engineering practices required to enable Continuous Testing which is foundation for Continuous Delivery. All the concepts are driven using the case studies and actual tasks that are typically involved in any live agile projects. The course introduces you to the tools and languages such as Cucumber, Gherkin, Java, JUnit, Maven, Cobertura, Jenkins and Selenium.

Am I Eligible?

You should have basic knowledge of Java. CP-MAT OR CP-BAT is the prerequisite for this continuous testing level certification called CP-CCT. Alternatively you should have Certified ScrumMaster® (CSM) or PMI Agile Certified Practitioner (PMI-ACP)® or Certified professional Devops Foundation (CP-DOF) or similar basic Agile / DevOps training. In absence of this you should be having at least 1 year of proven relevant experience in Agile/DevOps or Testing.

Duration?

CP-CCT is designed specifically for corporate and working professionals alike. If you are a corporate you can opt for either 5 and a half days course or 3 full days course followed by an examination. If you are a working professional opt for 3 full days course followed by an examination.

How do I enroll myself?

You can enroll yourself here.



Learning Objectives of CP-CCT:

1. Introduction

- 1.1. ATA Introduction
- 1.2. Course Introduction
- 1.3. Introduction to DevOps
- 1.4. DevOps Lifecycle
- 1.5. What is Continuous Testing?

2. Lab preparation

- 2.1. Eclipse setup
- 2.2. Java project
 - Maven project structure
 - Maven POM.xml
 - Maven dependencies
 - Junit Test runner
 - API and Web Level feature automation

3. Ideation

- 3.1. MVP, MMF and Epic Hypothesis
 - User Stories and Features
 - 3 C's of User Stories
 - Acceptance Criteria and DoD
 - Challenges in Test Automation
- 3.2. Behavior and Requirements
 - Behavior of the System
 - Test Case specification
 - User Stories and Scenarios



4. Collaboration

4.1. BDD cycle and Three Amigos

- Define
- Collaborate
- Illustrate
- Build
- Automate
- Evaluate

4.2. Attributes of a Good feature document

4.3. Group Exercise

- Create features for the given Requirements and User Stories

5. BDD using Cucumber

5.1. Cucumber components

- Behavior in Feature document
- Gherkin language and keywords
- Step definition
- Human Language Support
- Cucumber for Java - Cucumber JVM
- Testing Possibilities
- Getting BDD wrong

5.2. Installation and configuration for Cucumber

- Eclipse IDE setup
- Creating Maven project
- Maven dependencies
- Manual setup (non-maven) - required libraries

5.3. Illustrate and Build Features

- Preparing Feature files having test scenarios
- Writing a step definition
- Writing Runner Class
- Given, When, Then, And, But annotations and usage in features class
- Passing parameters in Step Functions
- Using regular expression in steps

5.4. Agile Practical - Sprint 1

Cucumber in action for API layer testing



Features and Steps definition for Requirements
Running the Requirements as Test
Retrospective

5.5. Test-first concept in Cucumber

- JUnit Test generation in Cucumber
- Pretty, Tags and Glue
- Cucumber HTML Reports
- Using Assertions to report failure
- Running simple feature/Step scenario
- Building a simple test case
- Running all/limited tests

5.6. Continuous Testing using Jenkins

- Settings for local Jenkins sever
- Maven Jobs
- Creating BDD builds in Jenkins
- Jenkins Dashboard and Reporting
- Code coverage using Cobertura

5.7. Agile Practical - Sprint 2

Jenkins in action
BDD using Jenkins jobs for the given User Stories
Retrospective

6. Selenium and GUI Testing

6.1. Selenium basics

- Simple WebDriver API
- Object locators
- Assertions and frequently used commands

6.2. Eclipse

- Configuring Eclipse for Selenium
- Updating the steps code for GUI level using Selenium
- Driving the API and GUI level tests from Eclipse
- Trigger the jobs and continuously integrate for the builds

6.3. Agile Practical - Sprint 3

- Importing Selenium Tests
- BDD using Selenium for the given User Stories
- Scheduling and integrating the builds



- Retrospective

7. Testing in all phases of DevOps

- 7.1. Continuous Deployment vs. Continuous Delivery
- 7.2. Release on Demand
- 7.3. Dark Launches and Canary Releases
- 7.4. A/B Testing

8. Practical Agile Testing

Case Study

Practice all BDD and Continuous Testing fundamentals throughout the course on a case study on a sample project.