

# Business Analysis (PBA & CBAP)

**BA420: Requirements Analysis Using Use Cases and User Acceptance Testing (UAT) for BAs** \$2,795.00

• 4 Days

# **Upcoming Dates**

# **Course Description**

Use Cases are an industry best practice for defining, documenting, and analyzing functional requirements. A use case approach is a usercentered approach for developing a solution to your business needs. Unfortunately, methods for developing use cases vary substantially across the industry. *What information belongs in a use case? How can a use case be utilized to capture all the desired functionality? Are use cases applicable to all projects? Is there a template for writing use cases that is considered best practice?* If you've ever been involved in a project involving use cases, you've probably encountered some confusing answers to these questions. Learn practical answers to these questions from one of the pioneers in Object Oriented Analysis. We have been developing and sharing best practices in use cases since the early 1990s.

This course provides a strong foundation in the mechanics of use case diagramming and writing textual descriptions of use cases. In this highly interactive workshop, you will learn how to enhance and refine your use case skills, how to involve your stakeholders in the use case process, and how to develop use cases that provide valuable information to the designers and testers. Use cases that meet the needs of designers can be too technical and too detailed for other stakeholders. Use cases that satisfy business users are usually not very helpful to designers and testers. How do you satisfy these two disparate interests? And how do you handle the details like business rules, data validations and user interface specifications?

All too often the software products delivered to the business do not meet their expectations, especially with regard to the quality of the product. An effective User Acceptance Testing (UAT) process addresses this issue by confirming the functionality and performance of the product prior to its release.

This workshop also looks at the business issues which drive the need for a fully functional UAT process and describes the components of such a process. It is designed to help the Business Analyst (BA) to develop an understanding of their role, the process, and the deliverables associated with UAT.

# **Course Outline**

# **Module 1: Introduction**

- Course objectives
- Understanding of business analysis
- Knowledge areas of BABOK® Guide

## Module 2: Overview of Use Cases

- Describe the purpose and value of a use case approach
- Use case terminology
- Introducing a use case approach

### Module 3: Needs Assessment

- Purpose of Needs Assessment
- Needs Assessment and use cases
- Defining and understanding the strategic goals and objectives
- Understanding the business process
- Identifying actors
- Identifying business information
- Analyzing locations and operations
- Drawing the business use case diagram
- Building the business architecture
- Defining solution options

## Module 4: Defining System Scope

- Product scope/project scope
- System actors versus business actors
- Analyzing the business process model
- Analyzing business use cases
- Identifying candidate use cases
- Identifying scenarios
- Diagramming use cases
- Draw a use case diagram
- Create a use case catalog

## Module 5: Evaluating, Prioritizing, and Packaging Use Cases

- Iterative nature of this work
- Evaluating use cases
- Use case priority
- Use case risk
- Use case complexity
- Use case dependencies
- Evaluate and Prioritize use cases
- Dividing work up between releases
- Packaging
- A process for how to perform packaging

## Module 6: Writing the main success scenario

- Use case descriptions
- Primary and secondary goals
- Assumptions
- Pre-conditions
- Triggers
- Post-conditions
- Scenario example

- Main success scenario
- Conditional execution
- Use cases and requirements
- Best practices for writing a use case description

## Module 7: Writing the other scenarios

- Scenarios and flows
- Alternate scenarios and flows
- Alternate vs. basic flow
- Guidelines for alternate flows
- Exception flows
- Failed post conditions
- Write alternate and exception flows

## Module 8: Process Modeling to describe use case flows

- UML® Activity Diagram Notation
- Sequencing activities
- Developing an activity diagram
- Facilitated sessions
- Draw a UML® Activity Diagram

## Module 9 Using Advanced Diagramming Techniques

- Commonality
- Dependency Relationships
- <<Includes>>
- <<Extends>>
- Identify <<include>> and <<extend>> relationships
- Comparing the different relationships

# Module 10: Developing a Requirements Specification

- Requirements and use cases
- Detailed requirements
- Common approaches to specifications
- Non-functional requirements
- Write non-functional requirements
- User interface requirements
- UI data descriptions
- Business rules
- Decision tables and inference rules
- How to document simple calculations
- Reporting requirements
- Data requirements
- Data accessibility requirements
- Traceability

### Module 11: Course Summary

- Wrap-up
- Finding more information

### Module 12: Software Testing - the Basics

- ° What is software testing and why is it important?
- ° Typical problems that we encounter with software
- $^\circ\,$  The Cost of Quality (CoQ) and the cost of finding defects too late
- ° The "V" Testing Model
- <sup>o</sup> The four stages of software testing (Unit, Integration, System, User Acceptance Testing (UAT)
- Testing best practices
- *Exercise:* "How would <u>you</u> test it?" A new technology is presented to the participants and they are asked to brainstorm what should be tested to ensure it is working correctly. (15 minutes)

### Module 13: Understanding the Tester's Terminology

- ° What is UAT and what is the role of the Business Analyst?
- ° Characteristics of a good UAT tester
- ° The tasks of UAT testing
- ° The UAT documents
- ° Common terms & definitions
  - 1. The six testing types for UAT (i.e., smoke, functional, intersystem, usability, regression, stress/load testing)
  - 2. The three testing techniques for UAT (i.e., control flow, parallel, and positive testing)
- ° Testing visibility (White Box, Gray Box, and Black Box testing)
- Exercise: UAT Jeopardy A fun and fast-paced game of Jeopardy used as a review of terms, concepts, and roles learned in this lesson.
  (15 minutes)

# Module 14: The UAT Planning Process

- $^{\circ}\,$  The importance of UAT planning
- ° What is a UAT Test Plan?
- $^\circ~$  The six steps for creating a UAT Test Plan
- ° General testing tips (see Testing Guidelines and Techniques document)
- *Exercise:* "Create a UAT Test Plan" The participants will identify UAT test scenarios from a Business Use Case and identify test objectives and high-level test data for each scenario

### Module 15; UAT Test Coverage

- What is UAT test coverage?
- ° Using a Requirements Traceability Matrix
- $^\circ~$  Set the testing scope what to test and what not to test
- ° The UAT Test Coverage Matrix
- $^{\circ}\,$  Verifying Non-functional requirements (i.e., performance, security, usability, availability, etc.)

### Module 16: Creating and Executing the UAT Test Cases

What are the goals of UAT testing?

- What is a UAT Test Case?
- How do UAT Test Plans and Test Cases relate?
- The four steps for creating a UAT Test Case
- *Exercise*: "Write a UAT Test Case" The participants will be asked to write a high-level Test Case from their Business Use Case and define test data for a portion of the steps
- Preparing, running, and documenting the UAT Tests
- General testing tips and techniques

## Module 17: Verifying the Test Results

- ° Documenting UAT test results
- ° What is a defect?
- ° How to log a defect?
- ° The "bug" lifecycle
- ° Writing a good problem description
- ° Taking screen snapshots
- ° 10 tips to avoid writing bad defect reports
- ° Exercise: "Log a defect" The participants will write a concise and complete statement to explain a defect
- ° Signing-off on UAT

## Module 18: Testing Vendor-Supplied Applications

- ° Challenges of testing vendor-supplied applications
- $^{\circ}~$  Challenges to the business
- ° Eight steps for testing vendor-supplied applications
- ° Handout: An Insurance industry case study for testing vendor product

# Audience

This course has been expressly designed for the Business Analyst, Business System Analyst, System Analyst, and/or Requirements Engineer. The Designer, Developer and Tester who are actively involved in utilizing use cases may also benefit from this workshop.

This course is designed for the Business Analyst professional who is involved with testing the functionality of technology projects.

# Prerequisites

Knowledge of requirements process and requirements elicitation techniques.

# What You Will Learn

- Employ use cases to elicit requirements, at a business, system or subsystem level
- Employ use cases to document the scope of a project
- Write use cases in a clear and unambiguous way.
- Model use cases with workflow diagrams
- Plan and divide up the project work based on your use cases
- Manage a use case as it evolves over time and goes through many changes
- Develop an understanding about basic concepts associated with User Acceptance Testing
- See how UAT applies to the Software Development Lifecycle (SDLC)
- Recognize benefits of improved quality of deployed software using User Acceptance Testing
- Identify the key roles, activities and deliverables which make up User Acceptance Testing
- Use a Business Use Case to define scenarios for testing
- Create a UAT test plan and write UAT test cases with associated test data
- Understand the process for testing functional and non-functional requirements

• Identify the challenges of testing vendor-supplied applications