

# **Business Analysis Workshop 2012 (BA2012)**

**Modality: Virtual Classroom**

**Duration: 5 Days**

## **Course Information**

The Business analysis comprises understanding and uttering requirements, which constantly has been the feeblest link in projects. For systems, up to 67 percent of conservation and 40 percent of development is misused rework and creep mainly attributable to incompetently defined business necessities. Too frequently projects continue based on somewhat other than what the business actually needs; and development practices usually focus mainly on the setup for representing necessities of the product to be produced without sufficiently establishing the content it must provide. Format of course is imperative too; and of all the aspects that can influence requirements, lack of the clarity is the most deceptive.

In turn, Testability—the capability to demonstrate that a necessity has or has not been met—is maybe the single most effective pointer of clarity. If one cannot describe how to test that a requirement has been encountered, then it's unlikely the developer will be capable to encounter the requirement appropriately; and nevertheless there's no test to indorse the requirement was met. Using the powerful Problem Pyramid and supplementary procedures, this cooperative workshop first gives members practice determining and detailing an authentic case's REAL business necessities content what a product need do to produce value for shareholders. Next, participants inspect factors influencing clarity and techniques to overcome testability concerns. Then participants practice describing and writing necessities of a product/system/software how to content the REAL business requirements and testing that requirements definitely have been encountered. In conclusion, the course defines methods for handling the requirements and the business examination itself. This course will aid you to get ready for the certification of Certified Associate Business Analyst: (CABA)

The average amount of salary of Business Analyst earn is **\$70,170** per year.

## **Course Objectives:**

After finishing this leadership and management training course, students will be capable to:

- Understand the role, significance, and concerns of business analysis and models affecting defining business, requirements
- Understand the differences amid the REAL user's (business) necessities and the product's (design) necessities
- How to do a business analysis, collect data, spot the important things, and understand them meaningfully
- What differentiates additional actual requirements discoverers from fewer effective one?
- Consuming the Problem Pyramid tool to obviously define difficulties, value, reasons, and real requirements
- Establish setups for analyzing, documenting, and collaborating business requirements.
- Writing the 'good' perfect and testable product/system/software necessities provisions, and

use cases

- Describing ways to check those requirements have been fulfilled
- Procedures and programmed tools to handle requirements changes and the business inquiry itself

## **Audience:**

The Business Analysis Workshop 2012 is perfect for those working in roles such as:

- Business Analysts
- Systems and business executives
- Project front-runners
- Programmer analysts
- Quality/testing specialists
- Accountants

## **Prerequisite:**

Students must have previous knowledge regarding Basics of Business Analysis.

## **Course Outline:**

### **BUSINESS ANALYSIS (BA) ROLE AND IMPORTANCE**

- How requirements produce value
- Sources and economics of system errors
- Survey on improving requirements quality
- Business analysis role inconsistent attention
- Project management model misses key BA
- Business analyst vs. project manager roles
- System development model BA conflicts
- BABOK® v2 model Knowledge Area issues
- Business vs. product/system requirements
- Common erroneous requirements beliefs
- A different, better model
- Integrating business and analysis
- Requirements process overview
- Exercise: Review BA's requirements

### **DISCOVERING REAL BUSINESS REQUIREMENTS**

- Discovering, not gathering, requirements
- Quantifying value
- REAL vs. presumed processes
- Requirements are the should be process
- Horizontal processes and vertical silos
- The we don't have time fallacy
- Problem Pyramid tool to get on track

- Exercise: Your Problem Pyramid
- Exercise: Applying Evaluation Guidelines
- Aligning strategy, management, operations
- Technology requirements vs. design
- Management/supervisor vs. worker views
- Who should define requirements
- Exercise: Review BA's recommendation
- Exercise: Identifying the REAL problem

## **GATHERING AND ANALYZING DATA**

- Why bother gathering data
- Surveys and questionnaires
- Literature search
- Reviewing documents
- Observing operations
- Participating and learning to do the work
- Joint Application Development (JAD) limits
- Prototyping and proofs of concept
- Interviewing and planning key to success
- Controlling with suitable questions
- Exercise: Plan and conduct interviews
- Exercise: Review and evaluate interviews
- Exercise: Identifying what else to know

## **FORMATS/MODELS TO AID UNDERSTANDING**

- Four ways to add meaning to data
- Business rules, structured English
- Cause-and-effect graphing
- Decision trees and decision tables
- Entity-Relationship diagrams, data models
- Flow charts and data flow diagrams
- Organization, responsibility, RACI charts
- Performance, volume, frequency statistics
- Sample forms, reports, screens, menus
- Exercise: Apply data analysis techniques
- Natural and non-natural segments
- Process maps
- Exercise: Confirming problem, causes

## **FORMATS FOR DOCUMENTING/ COMMUNICATING**

- Commonalities of requirements formats
- Itemized deliverables vs. narratives, diagrams
- IEEE standard for software requirements
- Pigeon-holing strengths and weaknesses
- Use cases, advantages and warnings

- 7 guidelines for documenting requirements
- Hierarchical business deliverable whats
- Exercise: Write top-level requirements
- Exercise: Review top-level requirements
- Conceptual system design solutions
- Requirements negotiation
- Requirements vs. implementation scope
- Iterating to avoid analysis paralysis
- Key to making reliable cost/time estimates
- Driving down to more detail
- Exercise: Making subjective objective

## **DEFINING PRODUCT/SOFTWARE REQUIREMENTS**

- Causes of poor software requirements
- Specifying the invented the product/system
- Conceptual design guide to product features
- Exercise: Product feature requirements
- Writing good requirements
- Exercise: Reviewing feature requirements
- Characteristics of a good SRS
- Correct, complete, feasible/necessary
- Exercise: Review for correct and complete
- Exercise: Review for consistency
- Ambiguity, warning about total elimination
- Exercise: Inherently ambiguous terms
- Exercise: Logical ambiguities
- Verifiable (testable), issues
- Making the untestable testable
- Exercise: Write test cases
- Writing good requirements

## **WRITING USE CASES AND SPECIFICATIONS**

- Suggested use case creation steps
- Actors and system boundary
- Use case diagram
- Use case textual contents
- Exercise: Narrative use case format
- Exercise: One-column use cases
- Exercise: Happy, alternate paths
- Use case strengths, weaknesses
- Exercise: Supplementary specifications
- Functionality matrix
- Structural testing concepts, flowgraphing
- Exercise: Flowgraph two-column use case
- Testing each use case path/scenario, warning
- Exercise: Other use case test conditions

## PERFORMING REQUIREMENTS-BASED TESTS

- Business analyst's testing roles
- What is requirements-based testing, issues
- Gurus equating with favored test technique
- Black-box (functional) risk-based testing
- Overemphasis on testability/clarity
- Superficial over-reliance on use cases
- Proactive Testing reviews catch more
- Exercise: Identify missing stakeholders
- Proactive Testing planning, design too
- Risk elements and testing
- Dynamic, passive and active static testing
- V-model levels of testing, objectives
- Proactive Testing Life Cycle model
- Agile user story acceptance tests vs. UAT
- Proactive user acceptance criteria
- Functionality the user must demonstrate
- How much, how often user must test
- Determining system quality
- Who should carry out acceptance tests
- How acceptance tests should be performed
- Added benefit, revealing requirements errors
- IEEE Standard for Test Documentation
- Overcoming controversial interpretations
- Testing structure's advantages
- Enabling manageability, reuse, selectivity
- Test plans, designs, cases, procedures
- Proactive Testing risk analysis
- Exercise: Identify project-level risks
- Letting testing drive development
- Exercise: Identify detailed test plan risks
- Exercise: Identify test design risks
- Checklists and guidelines identify more
- Decision trees and tables for business rules
- Exercise: Create a decision table
- Checklist of common test concerns
- Equivalence classes, boundary testing
- Test script and matrix formats

## MANAGING THE BA AND THE REQUIREMENTS

- Supporting, controlling, tracing changes
- Automated requirements management tools
- Traceability matrix, forward and backward
- What is a test case, relevance to tracing
- Business analysis is a project to be managed
- Project manager's triangle, triple constraints

- Project success starts with results
- Work breakdown structure identifies tasks
- Estimate target and reference considerations
- Weighted average, variance to control risk
- Intrinsic vs. extrinsic schedule factors
- Critical path, network (dependency) diagram
- Monitoring milestones and checkpoints
- 3 ways to determine percentage completion
- Earned value measure, depict graphically
- Project manager's job, key to advancement
- Measuring the "proof of the pudding"