Course 301: Approaches to Software Development (4 days)

Course Description…

There are as many approaches to software development as there are software developers. In this day and age, IT departments have to react quickly to the changing business environment creating new software, updating existing software and providing fixes to erroneous software. This course addresses the various methods, tactics, and procedures that will generate quality software from the structured approach to the newer agile methods.

Learning Objectives…

- Understand the various approaches that may be implemented to creating quality software.
- Examine the alternatives to developing software to determine which one to use under what circumstances.
- Review the overall software engineering discipline to gauge where software development fits into the picture.
- Show what a software development process is and how it can benefit the organization.

Who should attend…

Practitioners of software development: engineers, quality assurance professionals, programmers, designers, project managers, testers, architects, data base administrators, and any one else concerned with providing a quality software solution to a business problem.

Prerequisites…

No specific prerequisites are needed.

Course Outline…

Introduction and Overview
Course Objectives

Unit 1: Introduction

The Problem of Software
Systems and Software Engineering
Software Development Process
- What is a process?
- SDLC
- Benefits of a process
- Process improvement
- Process standards
- Policy and procedure

Modeling
Communication

Unit 2: Phased Approach

Exercise: What Should Be Done?
Why a Phased Approach
- The role of risk
- The Quality Gate

The Phases
- Phase definitions
- Order of the phases
- Transitions between the phases

Who Are the Players?
- Users and stakeholders
- Business Analysts and intermediaries
- Designers
- Programmers
- Testers
- Quality Assurance and Software Configuration Management
- Project management
- Upper level management

Linking the Phases
- Traceability
- The Traceability Matrix

The Role of Documentation
- Phase communication

In Practice
- What must be done
- What are good practices
- What will help
- What should be avoided

Keys to Success

Unit 3: Determining the Problem

Statement of Work
Problem Statement
- How to define the real problem
- Stating the vision

Exercise: What Is the Real Problem?
Business Case Overview
- Scope of work
- Justification
- Functional goals

Confirmation: How Do We Know It's the Right Problem
Workshop: Establishing the business case
Keys to Success

Unit 4: Defining the Problem

Requirements Process
What Are Requirements?
- Forms of requirements

Who Does Requirements?
- Requirements information gathering process
- Requirements definition process

Workshop: Deriving Requirements
Requirements Analysis Process
Requirements Documents
Validation
- Do we have all the requirements?
- Are the requirements valid?

Workshop: Validating requirements
Keys to Success

Unit 5: Designing the Solution

What Is Design?
The System Design Process
Architecture
- What is software architecture?
- Examples

Design Process
- Design levels
- Flowdown and allocation
- Creating the design
- Exercise: Modeling an User Interface
- Design methods
- Documenting the design
- Verification

Workshop: Building a model
Keys to Success

Unit 6: Developing the Product

What Is Development?
The Software Builder
Build Process

- Build methods
- Build techniques
- Coding language
- Tools

Validation

- Do the units do what they are supposed to?
- Do they work correctly?

The Debugging Process
Keys to Success

Unit 7: Demonstrating and Debugging the Software

Testing

- What is testing?
- Who does testing?

Test Cases
Exercise: Building Test Cases
Testing Process

- Integration testing
- System testing
- Acceptance testing

Defect Tracking and Quality Measurement
Validation

- Does the design work?
- Does the system work?
- Can the users use it?

Workshop: How would you test this?
Keys to Success

Unit 8: Delivering the Solution

Deployment Process

- What is deployment or delivery?
- Who does deployment or delivery?

Exercise: What Could Go Wrong?

Maintaining Operability

- What is maintenance?
- Who does maintenance?
- Maintenance process

Keys to Success

Unit 9: Approaches to Software Development

Evolution of Software Engineering Practices

The Discipline of the Software Engineering Approach

- Structure
- Method
- Documentation

Structured Approaches

- Linear
  - Waterfall
  - U or V Model
  - Information engineering
  - When to use
  - Risks involved
- Incremental
  - Staged delivery
  - Incremental life cycle
  - When to use
  - Risks involved
  - Exercise: Choosing increments

Evolutionary

- Top down Structured (Yourdon-DeMarco)
- Spiral- and risk-based (Boehm)
- Rapid application development
  - Prototyping
  - Iterative development
  - DSDM
- Exercise: Planned iterations
- When to use
- Risks involved

Object-Oriented Approaches

- Unified Process
- RUP
- When to use
- Risks involved

Agile Methods

- Extreme Programming
- Scrum
- Feature-Driven Development
- Test Driven Development
- Exercise: Preparing a User Story

Workshop: Planning the approach

Unit 10: Managing the Process

Estimation

Exercise: Estimating the Process

Quality Factors

Configuration Management

Overall Validation of the Process
- What have we done?
- Have we done it well?
- Was it worth doing?

Keys to Success

Unit 11: Automating the Software Development Process

CASE Tools
- Cradle-to-grave coverage
- Best of breed coverage

Benefits of Using Tools
Risks Involved with Using Tools
Representative Software Engineering Tools
- Requirements tools
- Design tools
- Testing tools
- Other software engineering tools

Keys to Success

Unit 12: Final Thoughts and Keys to Success

Where to go for more information
Industry Keys to Success summary

Please contact your ROI representative to discuss course tailoring!!!