

## **Course 329: Rapid Application Development: Prototyping for Results (4 days)**

### **Course Description...**

In these times of agile business and rapid development of software, to support the high-speed changes in the business, methods are needed to quickly, effectively and accurately create software that responds to the changing business environment. This course presents the various methods of rapid development that can be applied to generate requirements, validate a solution, or even create a new operational system.

### **Learning Objectives...**

- Define the concept of rapid development
- Examine the issues of quality in a rapid development environment
- Understand the various methods of prototyping requirements and software
- Develop processes for effectively using prototypes
- Compare the various methods for delivering software at higher velocity

### **Prerequisites...**

An understanding of the software development life cycle and experience developing software.

### **Who should attend...**

- ◆ Programmers
- ◆ IT project managers
- ◆ Project leads
- ◆ Technical leads
- ◆ System designers
- ◆ System architects
- ◆ Anyone interested in quality software development at higher velocity

### **The Approach...**

The first two days will introduce the concepts of rapid development and prototyping. The last days will introduce the methods for applying those concepts. There will be a student workshop for each of the methods so that the students gain hands-on experience in each prototyping technique.

**See next page for a detailed course outline...**



## Course Outline...

### Unit 1: Introduction

#### Software Development Life Cycles

- Mandatory elements
- Transitions
- Traditional Life Cycle Models

#### Process-oriented development

- CMM
- Other process models
- Downsides of process models

#### Basis for Rapid Development

- Timeboxing
- Iterative development
- Incremental delivery

#### Principles of agile methods

- Adaptation rather than prediction
- People-oriented rather than process-oriented

### Unit 2: Rapid Development

#### What is Rapid Development

- Mental models
- Kinds of schedule-oriented practices
- Real-world tradeoffs
- General strategy for attaining rapid development

#### The Quality Issue

- Relationship between quality and velocity
- What quality costs

#### The Management Issue

- Dealing with business management
- Dealing with IT management

#### A Rapid Development Strategy

- Avoiding classic mistakes
- Applying software fundamentals
- Avoiding the traps of rapid development

### Unit 3: Rapid Development Considerations

#### Risk Management

- Risk identification
- Risk analysis
- Risk prioritization
- Risk control

#### Documentation

- The purpose of documentation
- Effective use of artifacts



## **Change Management**

- What is change management
- Rapid change management
- Agile change management

## **Alternative life cycle approaches**

- Matching lifecycles with projects' schedule needs
- Linear models
- Staged delivery
- Design to schedule
- Design to tools
- Spiral model
- Agile approaches

## **Unit 4: Customer-Oriented Development**

- The rise of agile programming methods
- The agile manifesto
  - Individuals and interactions
  - Customer collaboration
  - Working software
  - Response to change
- Forms of communication
- The people factor
  - Involving the user
  - Involving the Business Analyst
  - The agile development team
- User participation
- Problem statement and vision (metaphor)
- Time boxing
- Requirements stacks or user story cards
- Refactoring
- Testing

## **Unit 5: Iterative Development**

### **What is iterative development**

- Iterating development cycles
- Benefits of iterative development
- User involvement

### **Forms of Iterative Development**

- Rational Unified Process
- Evolutionary prototyping
- Incremental delivery



## Unit 6: Evolutionary Prototyping

### What is prototyping?

- Static prototyping
- Dynamic prototyping
- Throw-away prototyping
- Evolutionary prototyping
- Where prototyping is used

### Prototyping sessions

- Structure
- Method
- Documentation
- Phases
- Change control

### Prototyping to define requirements

- Enough requirements to get started
- Form of prototyping
- Prototyping Tools
- Structured approach – data, cosmetics, function

### Prototyping to verify and validate requirements

- Confirmation strategy
- Format for presentation
- Identifying issues
- Handling change requests
- Revalidation

### User interface and Usability Prototyping

- Basic Elements
- Human Centered Design
- Principles of User interface
- Approaches and techniques

## Unit 7: Rapid Development methods

### Agile

- Extreme Programming
- Story telling

### Prototyping

- Storyboarding
- Use Case prototyping
- Scenarios
- CRC Cards
- Dynamic prototyping

## Unit 8: The Bottom Line

- The Culture of Rapid Development
- Rapid Development Best Practices
- Where to go for more information

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