Associate Course Syllabus - **Software Architecture**

Course Aims and Objectives

As the capstone for the associate software architect curriculum this course provides the final elements necessary to the knowledge and practice of the profession. During the course you will explore the software engineering and application development practices that distinguish you as a software architect. The program builds on the IT environment skills and design skills learned in the foundation associate courses.

After this course, you will be able to:

- Rise to design challenges through use of tools and methodologies.
- Reduce business risks associated with building technical solutions.
- Address stakeholder requirements and satisfy both functional and quality requirements.
- Refer to international standards to help structure your architect team and projects.

Format and Procedures

Each day the course instructor will guide students through, discussions and recommended best practice for delivery of Software architecture and value. Student's progress through detailed definitions and ontology, instructor and student led discussions, Offline and online workshops and industry case studies based on the fictitious Tinkleman company scenarios.

IMPORTANT: All courses now include the certification exams. Full membership is required to sit any certification exam. If you are not already a FULL MEMBER, please click here to register ($125).

To maintain your certification CEU's must be maintained. Click here for more information on CEU requirements.

Prerequisites

Minimum: Knowledge of the IT Body of Knowledge (ITABoK) - Downloadable materials included in membership (separate registration is required). Business Technology Strategy pre-work materials included.

Recommended: CITA-Foundation Certification

Associate Course Syllabus - **Software Architecture**

Module 1 – Software Architecture Fundamentals

Lesson 1 – Roles and Teams
- Workshop 1.1 – Creation of the Software Architecture Job Description

Lesson 2 – Viewpoint Considerations of the Software Architect & Terminology
- Workshop 1.2 – Views and Viewpoints Application

Lesson 3 – Software Architecture Principles

Lesson 4 – System Structures
- Workshop 1.3 – Patterns and Anti-Patterns Research Assignment

Module 2 - Software Construction

Lesson 1 – Application Development and Visualization
- Workshop 2.1 – Requirements Validation

Lesson 2 – Programming Patterns
- Workshop 2.2 – Selecting the Right Development Methodology

Lesson 3 – Software Construction

Lesson 4 – Technology Platforms

Module 3 – Components, Frameworks and Tools

Lesson 1 – Client Programming and User Experience (UX)
- Workshop 3.1 – Selection of Infrastructure Components

Lesson 2 – Client, Server and Storage Technologies
- Workshop 3.2 - Modeling

Lesson 3 - Workflow
- Workshop 3.3 – Creation of a Generic Architecture

Lesson 4 – Database Programming
- Workshop 3.4 – Database Design and Considerations

Module 4 – Service Network

Lesson 1 – Asynchronous and Synchronous Distributed Computing

Lesson 2 – SOA
- Workshop 4.1 – Applying SOA Principles

Lesson 3 - Messaging, XML and B2B

Lesson 4 – Application and Service Management
- Workshop 4.2 – Connecting the Sum of the Parts

Module 5 – Architectural Process, Methods and Artifacts

Lesson 1 – Modeling

Lesson 2- Applying Design Patterns
- Workshop 5.1 – Capture and Trace of Software Architecture

Lesson 3 – Code Quality Analysis

Lesson 4 – Design Patterns Selection and Application

Module 6 – Architecture Throughout the Lifecycle

Lesson 1 – Software Architecture Governance

Lesson 2 – Working with other Architects
- Workshop 6.1 – Application Service Transition

Lesson 3 – SDLC - What it means to the Software Architect
- Workshop 6.2 – Formal Release Policy

Lesson 4 – Professional Growth and Mentoring