

## **Overview:**

In previous modules, software architecture was presented as a macro design approach where requirements are used to create design elements that specify the main components and interfaces of software systems. This holistic black-box approach is essential for establishing the initial design and a framework of quality that guides the development process. However, architecture alone is not sufficient for constructing software in a systematic, disciplined, and quantifiable approach. During detailed design, the design process continues where software architecture leaves off for providing a white-box approach to design, where details left undefined and deferred to downstream designers are specified. These detailed designs fill the gaps in the design and provide a complete picture of how the system achieves its functional requirements within the quality framework established by the software architecture. Detailed design decisions can significantly shape the system's quality properties; therefore, they must be made within the boundaries of the software architecture. Upon completion of the detailed design activity, the software design is sufficiently complete to begin construction.

### Module Objectives:

- *Understand the role of detailed design within the software design phase.*
- *Become familiar with detailed design tasks during software design.*
- *Become familiar with the detailed design process.*
- *Understand fundamental concepts of object-oriented concepts and principles for component designs.*
- *Understand the role of documentation in software design and how to create a software design document.*

## **Session 1: Overview of Detailed Design (20 - 30 minutes)**

This session focuses on introducing the concepts of detailed design as tool for internal design of logical architectures. The difference between architecture vs. detailed design is presented as well as the process and tasks associated with detailed design. An important conceptual model for detailed designers is introduced; this reflects the tasks that need to be performed by detailed designers during different phases of a project (e.g., design, construction, or maintenance phase). Finally, the session presents the key tasks performed during detailed design, which include: understanding architecture and requirements, creating, evaluating, and documenting detailed designs, and managing synchronicity between design and implementation.

## **Session 2: Structural and Behavioral Design of Components (30 - 45 minutes)**

This session presents essential information for designing the internal structure and behavior of components identified during the software architecture activity. The discussion focuses on the object-oriented design strategy; therefore, important OO concepts, such as classes vs. objects, interfaces, types, subtypes, dynamic binding, and polymorphism are presented. In addition, mainstream design principles for achieving quality (in terms of reusability, maintainability, etc.) are covered, including the Open-Closed, Liskov, and Interface Segregation principles. Finally, the UML is employed to present concepts for modeling detailed behavioral aspects of software systems. This session provides the background material required to successfully model software using design patterns, the topic of the next module.

Slides – Chapter 5 – Session II

### **Homework #5**

- Read chapter 5 and answer review questions 1-17. Submit your answers as a word or PDF document.

### **Quiz #5**

*See Quiz #5.*