

# Software Architecture [SA]

B.E. Sem. VIII [CMPN]

---

---

## EVALUATION SYSTEM

	Time	Marks
<b>Theory Exam</b>	3 Hrs.	100
<b>Oral Exam</b>	–	25
<b>Term Work</b>	–	25

## SYLLABUS

**Objectives of the course:** Software architecture is foundational to the development of large, practical software-intensive applications. Critically, this course focuses on supporting creation of real implemented systems. Hence the course details not only modeling techniques, but design, implementation, deployment, and system adaptation -- as well as a host of other topics -- putting the elements in context and comparing and contrasting them with one another. Rather than focusing on one method, notation, tool, or process, this new course widely surveys software architecture techniques, enabling us to choose the right tool for the job at hand.

**Pre-requisites:** Object Oriented Software Engineering

### Module

#### 1. Basic Concepts

- 1.1 Concepts of Software Architecture
- 1.2 Models.
- 1.3 Processes.
- 1.4 Stakeholders.

#### 2. Designing Architectures

- 2.1 The Design Process.
- 2.2 Architectural Conception.
- 2.3 Refined Experience in Action: Styles and Architectural Patterns.
- 2.4 Architectural Conception in Absence of Experience.

#### 3. Connectors

- 3.1 Connectors in Action: A Motivating Example.
- 3.2 Connector Foundations.
- 3.3 Connector Roles.
- 3.4 Connector Types and Their Variation Dimensions.
- 3.5 Example Connectors.

#### 4. Modeling

- 4.1 Modeling Concepts.
- 4.2 Ambiguity, Accuracy, and Precision.
- 4.3 Complex Modeling: Mixed Content and Multiple Views.
- 4.4 Evaluating Modeling Techniques.
- 4.5 Specific Modeling Techniques.

#### 5. Analysis

- 5.1 Analysis Goals.
- 5.2 Scope of Analysis.
- 5.3 Architectural Concern being Analyzed.

- 5.4 Level of Formality of Architectural Models.
- 5.5 Type of Analysis.
- 5.6 Analysis Techniques.

## **6. Implementation and Deployment**

- 6.1 Concepts.
- 6.2 Existing Frameworks.
- 6.3 Software Architecture and Deployment.
- 6.4 Software Architecture and Mobility.

## **7. Conventional Architectural styles**

- 7.1 Pipes and Filters
- 7.2 Event- based, Implicit Invocation
- 7.3 Layered systems
- 7.4 Repositories
- 7.5 Interpreters
- 7.6 Process control

## **8. Applied Architectures and Styles**

- 8.1 Distributed and Networked Architectures.
- 8.2 Architectures for Network-Based Applications.
- 8.3 Decentralized Architectures.
- 8.4 Service-Oriented Architectures and Web Services.

## **9. Designing for Non-Functional Properties**

- 9.1 Efficiency.
- 9.2 Complexity.
- 9.3 Scalability and Heterogeneity.
- 9.4 Adaptability.
- 9.5 Dependability.

## **10. Domain-Specific Software Engineering**

- 10.1 Domain-Specific Software Engineering in a Nutshell.
- 10.2 Domain-Specific Software Architecture.
- 10.3 DSSAs, Product Lines, and Architectural Styles.

## **References**

1. Software Architecture: Foundations, Theory, and Practice (*Richard N. Taylor, Nenad Medvidovic*) Eric Dashofy , ISBN: 978-0-470-16774-8
2. Software Architecture Perspectives on an Emerging Discipline (*M. Shaw*) Prentice-Hall.
3. Software Architecture in Practice (*Len Bass, Paul Clements, Rick Kazman*) Pearson
4. Pattern Oriented Software Architecture (*Frank Buchanan et al*) Wiley India.
5. The Art of Software Architecture (*Stephen T. Albin*)

