# 20IT32E2 - SOFTWARE ARCHITECTURE

|  |  |  |  |
| --- | --- | --- | --- |
| Course Category: | Professional Elective | Credits: | 3 |
| Course Type: | Theory | Lecture-Tutorial-Practical: | 3-0-0 |
| Prerequisite: | Need to know the fundamentals of Software engineering | Sessional Evaluation:Univ. Exam Evaluation:Total Marks: | 4060100 |
| Objectives: | * Understand basic software architecture requirements, views and patterns etc.
* Evaluate software architecture and quality attributes
* Select and explore appropriate key architectural structures, tactics and methods
* Defining guidelines for documenting software Architecture
 |

|  |  |
| --- | --- |
| Course Outcomes | Upon successful completion of the course, the students will be able to: |
| CO1 | Understand the basics of software architectural requirements, views, patterns and influences on business and technical issues |
| CO2 | Analyze the quality attributes and to apply the same to prepare the documentation based on the suitability of attributes |
| CO3 | Specify the key structures, tactics and patterns to design and specify the architecture. |
| CO4 | Study different methods of agility and understand the business goals using other methods including various design strategies |
| CO5 | Prepare a document for a given architecture using views |
| CO6 | Identify the factors that influence the management and governance |
| Course Content | UNIT-I**Introduction to Software Architecture:** Definitions**,** Architectural Structures and Views, Patterns, What Makes a “Good” Architecture.**Importance of Software Architecture:** System’s Quality Attributes, change and prediction, communication enhancement, Design Decisions, Constraints, Influences, Evolutionary Prototyping, Improving Cost and Schedule Estimates, Transferable, Reusable Model, Independently Developed Components, Vocabulary of Design Alternatives and Training.UNIT-II**Context of Software Architecture:** Technical Context, Project Life-Cycle, Business, Professional**,** Stakeholders and influences.**Quality Attributes:** Understanding quality attributes, Availability, Interoperability, Modifiability, Performance and Security.UNIT-III**Architectural Tactics and Patterns:** Architectural Patterns, Overview, Relationships between Tactics and Patterns.**Quality Attribute Modelling and Analysis:** Modelling Architectures, Attribute Analysis and Checklists, Experiments, Simulations, and Prototypes, Different Stages of the Life.UNIT-IV**Architecture in Agile Projects: Overview,** Agility and Architecture Methods, examples, Guidelines for the Agile.**Architecture and Requirements:** Gathering ASRs from Requirements Documents, Stakeholders, Understanding the Business Goals, Utility Tree and Methods.**Designing an Architecture:** Design Strategy, The Attribute-Driven Design Method and supporting steps.UNIT-V**Documenting Software Architectures:** Uses and Audiences for Architecture Documentation, Notations, and Views, Choosing and Combining Views, Building the Documentation Package, Documenting Behavior, Architecture Documentation and Quality Attributes.**Architecture, Implementation, and Testing:** Architecture and Implementation, Architecture and Testing.**Architecture Reconstruction and Conformance:** Architecture Reconstruction Process, View Extraction, Database Construction, View Fusion, Finding Violations and Guidelines.UNIT-VI**Architecture Evaluation:** Evaluation Factors, Architecture Tradeoff Analysis Method and Evaluation.**Management and Governance:** Planning, Organizing, Implementing, Measuring and Governance.**Architecture in the cloud:** Basic cloud definitions, Service models and deployment options, Economic justification, Base mechanisms, Sample Technologies, Architecture in a cloud environment. |
| Text Books &ReferenceBooks | **TEXT BOOKS:**1. Len Bass, Paul Clements, Rick Kazman “Software Architecture in Practice”, Third Edition, Addison Wesley Publishers, 2013.

**REFERENCE BOOKS:**1. Mary Show, David Garlan, “S/W Arch. Perspective: on an Emerging Discipline”, 1996, PHI.
2. Jeff Garland, Richard Anthony, “Large-Scale Software Architecture A Practical Guide using UML”, John Wiley and Sons Ltd, 2003.
3. Oliver Vogel, Ingo Arnold, ArifChughtai, TimoKehrer “Software Architecture A Comprehensive Framework and Guide for Practitioners”, Springer, 2009.
4. Ian Gorton, “Essential Software Architecture”, Second Edition, Springer Publishers, 2011.
 |
| E-Resources | 1. <https://nptel.ac.in/courses>
2. <https://freevideolectures.com/university/iitm>
 |

**CO-PO Mapping:** 3-High Mapping, 2-Moderate Mapping, 1-Low Mapping, - -Not Mapping

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|   | **PO1** | **PO2** | **PO3** | **PO4** | **PO5** | **PO6** | **PO7** | **PO8** | **PO9** | **PO10** | **PO11** | **PO12** |
| **CO1** | 3 | - | - | - | - | - | 1 | 3 | - | - | - | 3 |
| **CO2** | 2 | 2 | - | 2 | - | 3 | - | - | - | - | - | 3 |
| **CO3** | - | 2 | 3 | 3 | 3 | - | - | - | - | - | - | 2 |
| **CO4** | - | - | - | 3 | 3 | 2 | - | - | - | 2 | - | 2 |
| **CO5** | - | 3 | - | - | - | 3 | 2 | - | 3 | 2 | - | 2 |
| **CO6** | - | - | - | 3 | 3 | - | - | - | - | 3 | 3 | 3 |