SOFTWARE ARCHITECTURE AND DESIGN PATTERNS

Course Code: 13IT2112 **L P C**
4 0 3

Pre requisites: Software Engineering.

Course Educational Objectives:
The main objective is to introduce the student to architecture of software and design Patterns. Upon completion of this course the student will Get an idea on envisioning architecture, creating an architecture, analyzing architecture.

1. Understand the creational and structural patterns.
2. Be capable of applying his knowledge to create an architecture for given application.
3. Be able to explain the role of analyzing architectures.
4. Be able to identify different structural patterns.

Course Outcomes:
At the end of the course the student will be able to

1. Understand the architecture, creating it and moving from one to any, different structural patterns.
2. Analyze the architecture and build the system from the components.
3. Design creational and structural patterns.
4. Learn about behavioral patterns.
5. Do a case study in utilizing architectural structures.

UNIT- I
Envisioning Architecture: The Architecture Business Cycle, What is Software Architecture, Architectural patterns, reference models, reference architectures, architectural structures and views.

Creating an Architecture: Quality Attributes, Achieving qualities, Architectural styles and patterns, designing the Architecture, Documenting software architectures, Reconstructing Software Architecture.

UNIT -II

Analyzing Architectures: Architecture Evaluation, Architecture design decision making, ATAM, CBAM.

Moving from one system to many: Software Product Lines, Building systems from off the shelf components, Software architecture in future.

UNIT-III

Patterns: Pattern Description, Organizing catalogs, role in solving design problems ,Selection and usage.

Creational and Structural patterns: Abstract factory, builder, factory method, prototype, singleton, adapter, bridge, composite, façade, flyweight, Proxy.

UNIT- IV

Behavioral patterns: Chain of responsibility, command, Interpreter, iterator, mediator, memento, observer, state, strategy, template method, visitor.

UNIT -V

Case Studies: A-7E – A case study in utilizing architectural structures, The World Wide Web - a case study in interoperability, Air Traffic Control – a case study in designing for high availability, Celsius Tech – a case study in product line development

Text Books:

1. Len Bass,Paul Clements&Rick Kazman, *Software Architecture in Practice*, 2nd Edition, Pearson Education, 2003.
2. Erich Gamma, *Design Patterns*, 1st Edition, Pearson Education,1995.

References:

1. Luke Hohmann , *Beyond Software architecture*, Addison wesley, 2003.
2. David M. Dikel, David Kane and James R. Wilson, *Software architecture*, 1st Edition, Prentice Hall,2001
3. F.Buschmann , *Pattern Oriented Software Architecture*, Wiley&Sons,1st Edition,2001

Web references :

http://en.wikibooks.org/wiki/Introduction_to_Software_Engineering/Architecture/Design_Patterns.