

CS010 605 SOFTWARE ENGINEERING

Teaching scheme

Credits: 4

3 hours lecture and 1 hour tutorial per week

Objectives:-

To familiarize the steps in designing a Computer Software System following the conventions in Engineering Design.

To introduce the fundamentals of Structured and Object Oriented Designs and Design Tools.

Module I (12 Hrs)

The Evolving role of Software – Software – The changing Nature of Software – Legacy software ,Introduction to CASE tools, A generic view of process– A layered Technology – A Process Framework – The Capability Maturity Model Integration (CMMI) – Process Assessment – Personal and Team Process Models. Product and Process. Process Models – The Waterfall Model – Incremental Process Models – Incremental Model – The RAD Model – Evolutionary Process Models – Prototyping – The Spiral Model – The Concurrent Development Model – Specialized Process Models – the Unified Process.

Module - II (12 Hrs)

Management: Functions - Project planning - Software productivity - Productivity metrics - Cost estimation - COCOMO & COCOMO II - Project control - Work breakdown structures, Gantt charts, PERT charts - Dealing with deviations - Team organization - centralized, de-centralized, mixed - An assessment of organizations - Risk management – Configuration Management. Introduction to project management and planning CASE tools.

Module - III (12 Hrs)

Requirements Engineering : Requirements Engineering tasks – Initiating the requirements Engineering Process-Eliciting Requirements – Developing Use cases – Building the Analysis Models – Elements of the Analysis Model – Analysis pattern – Negotiating Requirements – Validating Requirements. SRS Document.

Module - IV (12 Hrs)

Design activity & its objectives – Function Oriented and Object Oriented Design-Modularization techniques - module structure and its representation, interface and information hiding, categories, specific techniques to accommodate change, stepwise refinement, top-down and bottom-up design - Handling anomalies. Case Study with UML and CASE Tool support.

Module - V (12 Hrs)

Implementation Techniques - Programming principles and guidelines – Structured Programming. Software Testing Fundamentals-Test Case Design-White-Box Testing-Basis Path Testing-Control Structure Testing- Black-Box Testing- Various levels of Testing : Modules to System. Case study : Test case design and Testlog preparation

References

1. Roger S.Pressman, Software Engineering: A Practitioner's Approach, McGraw Hill International edition, Seventh edition.
2. Ian Sommerville, Software Engineering, 8th Edition, Pearson Education, 2008 (UNIT V)
3. Stephan Schach, Software Engineering, Tata McGraw Hill, 2007
4. Pfleeger and Lawrence Software Engineering: Theory and Practice, Pearson Education, second edition, 2001