

CAP437:SOFTWARE ENGINEERING PRACTICES

Course Outcomes: Through this course students should be able to

CO1 :: represent the given project in various phases of a lifecycle.

CO2 :: select appropriate process model depending on the user requirements.

CO3 :: apply the knowledge, techniques, and skills in the development of a software product.

CO4 :: differentiate various processes used in all the phases of the software development.

CO5 :: illustrate various life cycle activities like analysis, design, implementation, testing and maintenance.

Unit I

Introduction to software engineering : define software engineering, software process, software engineering practices

Software process models : classical software development lifecycle model, prototyping model, V model, software development life cycle (SDLC), incremental Model, introduction to agile method of software development

Unit II

Requirement engineering : requirement analysis, stakeholder analysis, software requirement specification document, characteristics of a good SRS, requirement engineering, requirement eliciting/gathering, negotiating requirement, validating requirement, functional and non-functional requirement

Unit III

Design : design process, design concepts, coupling, cohesion, data flow diagram (DFD), flow chart, architectural design, component based design, object oriented design, class based components, use case diagram, class diagram, activity diagram

Unit IV

User interface design : golden rules, interface design models, interface design process, interface design activities

Standards : good coding practices, coding standards, code reusability, documentation, documentation standards

Unit V

Software testing : test planning, test case template, test design, test case definition

Testing strategies : black box testing, white box testing, sanity testing, smoke testing

Testing levels : unit testing, integration testing, system testing, acceptance testing, regression testing

Bugs : bug/defect definition, bugs life cycle, bug tracking, bug tracking tool (bugzilla overview)

Unit VI

Software maintenance : software supportability, reengineering, business process reengineering, software reengineering, restructuring, economics of reengineering, software maintenance

Software process improvement : approaches to SPI, maturity models, SPI process

Product metrics : measure, metrics and indicators, measurement principles, function based metrics, metrics for specification quality

Text Books:

1. SOFTWARE ENGINEERING A PRACTITIONERS APPROACH by R.S. PRESSMAN, MCGRAW HILL EDUCATION

References:

1. FUNDAMENTALS OF SOFTWARE ENGINEERING by RAJIB MALL, PHI Learning
2. INTEGRATED APPROACH TO SOFTWARE ENGINEERING by PANKAJ JALOTE, NAROSA PUBLISHING HOUSE