**Sixteen Week Plan**

**Department of Computer Science**

**Faculty of Computing & Information Technology**

**Hafiz Hayat Campus, University of Gujrat**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Title** | | | | **Software Engineering** | | | |
| **Code** | | | | CS-321 | | | |
| **Credit hours** | | | | 3.0 | | | |
| **Prerequisite** | | | | Programming fundamental, Object Oriented Programming, Software Engineering Processes | | | |
| **Category** | | | | Core -Computer Science | | | |
| **Course Description** | | | | -- | | | |
| **Aims & Objectives** | | | | **Aim:**  Application of software engineering practices to the development of software in information system development domain where professionalism, quality, schedule, and cost are important in producing an information system.  **Objectives:**  To understand the importance and need of information system software engineering  To discuss different software development models, appropriate for the development and maintenance of software products  To introduce the basic project management concepts for the development of a high-quality product  To impart comprehensive knowledge regarding software development lifecycle  To demonstrate, with justification, an appropriate set of tools to support the development of a range of software projects  Hands on Training for CASE Tools and testing tools. | | | |
| **Learning Outcomes** | | | | Extract and analyze software requirements  Develop some basic level of software Architecture/Design  Apply standard coding practices  Apply different testing techniques  Describe debugging and its effectiveness.  Students will be able to select and apply appropriate Design Patterns  Understand quality, Testing issues | | | |
| **Text Book** | | | | Roger S. Pressman, “Software Engineer: A Practitioner’s Approach”, 6th/e, M c Graw-Hill, 2001, ISB: 0072496681  Bruce M., Roger P., Software Engineering: A Practitioner’s Approach, viii ed. McGraw-Hill Science, 2014, ISBN 978-0078022128 | | | |
| **Reference Books & Material** | | | | Internet | | | |
| **Grading Breakup and Policy** | | | | Assignment(s): 10% Quizzes: 5%  Project: 10% Midterm Examination: 25% Final Examination: 50% | | | |
|  | |  | |  | | | |
| Week# | Lecture # | | **TOPICS** | | **Source**  (**Book, Chapter No)** | Recommendationsfor LearningActivities (Mention Assignments, Test, Case Study, Projects, Lab Work or Reading Assignments) |  |
| **01** | **01** | | The Nature of Software, Unique Nature of WebApps, Software Engineering | | [TB2: Ch. 1] |  |  |
| **02** | | The Software Process, Software Engineering Practice, Software Myths. | | [TB2: Ch. 1] |  |  |
| **02** | **03** | | Introduction to Software Process Model  Water fall model  Incremental development  Reuse-oriented software engineering | | [TB1: Ch. 2] |  |  |
| **04** | | Process Activities,  Techniques to cope with change. | | [TB1: Ch. 2] | Assignment-1 |  |
| **03** | **05** | | Coping with change (cont.),  Spiral Model,  The Rational Unified Process | | [TB1: Ch. 2] |  |  |
| **06** | | Introduction to Agile Development,  Plan Driven Development VS Agile Development | | [TB1: Ch. 3] | Quiz 1 |  |
| **04** | **07** | | Extreme Programming, Testing in XP, Pair Programming, Agile Project Management, Scaling Agile Methods | | [TB1: Ch. 3] |  |  |
| **08** | | Project Management Concepts, Project Management Lifecycle | | Handouts | Project Proposal Announced |  |
| **05** | **09** | | Project Scheduling: GANTT chart, Critical Path Method  LAB Session: Microsoft Project (Project Scheduling and Tasks, Tasks Dependencies and Resources Allocation) | | Handouts | LAB session  Case Study |  |
| **10** | | Introduction to Requirement Engineering, Functional requirements and non-functional requirement,  Requirement Document | | [TB1: Ch. 4] |  |  |
| **06** | **11** | | Requirement Specification, Natural language processing, Structured Specification, Engineering Processes, Requirement Elicitation and analysis | | [TB1: Ch. 4] |  |  |
| **12** | | Requirements discovery, Interviewing, Scenarios, Use cases, Ethnography, Requirement Validation, Requirement Management, Requirement Management Planning, Requirement Change Management | | [TB1: Ch. 4] |  |  |
| **07** | **13** | | Introduction to System Modelling, Context models, Interaction models, Use case modeling, Sequence diagrams | | [TB1: Ch. 5] |  |  |
| **14** | | Intro to Structural Models, Class Diagrams, Generalization, Aggregation, Behavioral Models, Data-driven modeling, Event-driven modeling | | [TB1: Ch. 5] |  |  |
| **08** | **15** | | Model Driven Engineering, Model Driven Architecture, Executable UML | | [TB1: Ch. 5] | Assignment 2 |  |
| **16** | | Introduction to Architectural Design,  Architectural Design Decisions, Architectural Views, Architectural Patterns, Layered architecture,  Repository architecture,  Client–server architecture,  Pipe and filter architecture | | [TB1: Ch. 6] | Quiz 2 |  |
| **Mid Term Exam** | | | | | | |  |
| **09** | **17** | | Application architectures,  Transaction processing systems, Information systems, Language processing systems | | [TB1: Ch. 6] |  |  |
| **18** | | Revision for Mid Term | |  |  |  |
| **10** | **19** | | Introduction to Object Oriented Design using UML, System Context and Interactions, Architectural Design, Object Class Identification, Design Models, Interface Specifications | | [TB1: Ch. 7] |  |  |
| **20** | | Design patterns, Creational Design Patterns, Structural Design Patterns, Behavioral Design Patterns, Implementation issues, Reusability, Configuration Management | | [TB1: Ch. 7] |  |  |
| **11** | **21** | | Host Target Development,  Open Source Development, Open Source Licensing, | | [TB1: Ch. 7] |  |  |
| **22** | | Introduction to Software Testing, Development testing, Unit testing, Choosing unit test cases, Component testing, System testing | | [TB1: Ch. 8] |  |  |
| **12** | **23** | | Test-driven development, Release testing, Requirements-based testing, Scenario testing, Performance testing, User testing, | | [TB1: Ch. 8] |  |  |
| **24** | | Introduction to Project management, Risk Management, Risk Identification, Risk analysis, Risk planning, Risk monitoring | | [TB1: Ch. 22] | Assignment 3 |  |
| **13** | **25** | | Managing people, Motivating people, Teamwork, Selecting group members, Group organization, Group communications, | | [TB1: Ch. 22] | Quiz 3 |  |
| **26** | | Introduction to Software Pricing, Plan Driven Development, Project Plan, The Planning Process, Project scheduling, Schedule representation | | [TB1: Ch. 23] |  |  |
| **14** | **27** | | Agile planning, Estimation techniques, Algorithmic cost modeling, The COCOMO II model | | [TB1: Ch. 23] |  |  |
| **28** | | Introduction to Software Quality, Software standards, The ISO 9001 standards framework, Reviews and inspections, The review process, Program Inspections | | [TB1: Ch. 24] |  |  |
| **15** | **29** | | Software measurement and metrics, Product metrics, Software component analysis, Measurement ambiguity, | | [TB1: Ch. 24] | Assignment 4 |  |
| **30** | | Introduction to Configuration Management, Change Management, Version Management, System building, Release management, | | [TB1: Ch. 25] | Quiz 4 |  |
| **16** | **31** | | Project Presentation | |  |  |  |
| **32** | | Project Presentation | |  |  |  |
| **Final Exam** | | | | | | |  |