**Lesson Plan**

**Name of the Faculty - Manisha**

**Discipline - B.tech. Computer Science and Engg.**

**Semester - 5th Sem.**

**Subject - Software Engineering(PEC-CSE-311 G)**

**Lesson Plan Duration - 15Weeks**

|  |  |  |
| --- | --- | --- |
| Week | Theory | |
| Lecture Day | Topic (including assignment/Test) |
| 1 | 1. | Sofware process,products,Emergence of software engineering |
| 2. | Evolving Role of software, Life cycle models, Software Application and characteristics,Crisis |
| 3. | Spm concepts, software process metrices and Projectplaning,project and emperical estimation techniques,COCOMO-Heursitic estimation method. |
| 2 | 4. | Staffing level estimation ,team structure, risk analysis and mangment,project scheduling and tracking |
| 5. | Requirement analysis and specification,Requirement engineering, System modeling and principle modeling. partitioning Software, prototyping: |
| 6. | Prototyping methods and tools,Specification principles, Representation, the software requirements specification and reviews Analysis Modeling, Data Modeling, Functional modeling and information flow, Data flow diagrams |
| 3 | 7. | Behavioral Modeling,The mechanics of structured analysis,Creating entity/ relationship diagram |
| 8. | Data flow model, control flow model, the control and process specification; The data dictionary,Other classical analysis methods |
| 9. | System Design,Design concepts and principles, the design process, Design and software quality |
| 4 | 10. | Design principles, Design concepts,Abstraction, refinement, modularity, software architecture, control hierarchy |
| 11. | Structural partitioning, data structure, software procedure, information hiding; Effective modular design,Functional independence |
| 12. | Cohesion, Coupling, Design Heuristics for effective modularity,The design model,Design documentation |
| 5 | 13. | Architectural Design, Software architecture, Data Design: Data modeling, data structures |
| 14. | Databases and the data warehouse, Analyzing alternative Architectural Designs |
| 15. | Architectural complexity,Mapping requirements into a software architecture, Transform flow |
| 6 | 16. | Transaction flow,Transform mapping: Refining the architectural design |
| 17. | Testing and maintenance,Software Testing Techniques, |
| 18. | Software testing fundamentals,objectives, principles |
| 7 | 19. | Testability,Test case design, white box testing |
|  | 20. | Assignment, Control structure testing, basis path testing |
|  | 21. | Black box testing |
| 8 | 22. | Testing for specialized environments ,architectures and applications |
|  | 23. | Software Testing Strategies,Verification and validation |
|  | 24. | Unit testing, Integration testing, Validation testing |
| 9 | 25. | Alpha and beta testing, System testing |
|  | 26. | Revision , stress testing |
|  | 27. | Recovery testing, security testing, |
| 10 | 28. | Performance testing, The art of debugging |
|  | 29. | The debugging process debugging approaches. |
|  | 30. | Software re-engineering , reverse engineering |
| 11 | 31. | Assignment |
|  | 32. | Restructuring, forward engineering |
|  | 33. | Software Reliability and Quality Assurance |
| 12 | 34. | Quality concepts, Software quality assurance |
|  | 35. | SQA activities, Software reviews |
|  | 36. | Cost impact of software defects, defect amplification and removal |
| 13 | 37 | formal technical reviews, The review meeting |
|  | 38. | Review reporting and record keeping, review guidelines |
|  | 39. | Formal approaches to SQA,Statistical software quality assurance |
| 14 | 40. | Software reliability, Measures of reliability and availability |
| 41 | The ISO 9000 Quality standards: The ISO approach to quality assurance systems |
| 42 | The ISO 9001 standard |
| 15 | 43 | Software Configuration Management, Computer Aided software Engineering |
| 44 | CASE, building blocks, integrated case environments and architecture, repository. |
| 45 | Revision |