

Lesson Plan

Name of Faculty : Ms. Varsha, Assistant Professor of CSE
Discipline : Computer Science and Engineering
Semester : VIIIth (Even)
Subject : Software Testing and Quality Assurance- PEC-CSE412-T
Lesson Plan Duration : 15 weeks (from January/ February-2025 to June/July-2025)
Work Load (Lecture/Practical) per week (in hours): (3-L) hours

| Week | Theory | | Topic Covered Date and Remarks | | |
|------------------|----------------------------------|--|--------------------------------|-----|---------------------|
| | Lecture- Day | Topic (Including Assignment/Test) | Date | HOD | Director- Principal |
| 1 st | 1 | Introduction to Basic of Software Testing & Terminology | | | |
| | 2 | Software Development & Software Testing Life Cycle- Role and Activities | | | |
| | 3 | Necessity and Objectives of testing, Quality Concepts | | | |
| 2 nd | 4 | Quality Control | | | |
| | 5 | McCall's factor model | | | |
| | 6 | Different Software Development Model, Object-oriented Testing | | | |
| 3 rd | 7 | Web testing | | | |
| | 8 | GUI testing | | | |
| | 9 | Elements of Software quality assurance, Quality Assurance Activities | | | |
| 4 th | 10 | Statistical Quality Assurance; Software Reliability | | | |
| | 11 | SQA Plan, Quality Standards: -IEEE, CMM, ANSI | | | |
| | 12 | Testing Concepts, Issues and Techniques, Levels of Testing | | | |
| 5 th | 13 | Verification and Validation Model , System testing, Installation Testing, | | | |
| | 14 | Techniques of Verification: -Peer Review, Walkthrough, Inspection, FTR, Usability Testing | | | |
| | 15 | Unit testing, Integration testing, Function Testing | | | |
| 6 th | 16 | Regression testing | | | |
| | 17 | Performance testing: -Load Testing, Stress Testing | | | |
| | 18 | Security testing, Volume testing; Acceptance testing: -Alpha testing, Beta testing, Gamma testing. | | | |
| 7 th | 1st Minor Test | | | | |
| 8 th | 19 | Black Box Testing Methods: Equivalence partitioning, Boundary-value analysis | | | |
| | 20 | Error guessing, graph-based testing methods, Decision Table Testing | | | |
| | 21 | White Box Testing Methods: Statement coverage, | | | |
| 9 th | 22 | Decision coverage | | | |
| | 23 | Condition coverage, Path testing, Data flow testing | | | |
| | 24 | Test Planning & Documentation: Development plan and quality plan objectives | | | |
| 10 th | 25 | Testing Strategy, Test Management | | | |
| | 26 | Testing Strategy, Test Management | | | |
| | 27 | Strategic Management | | | |
| 11 th | 28 | Operational Test Management, Managing the Test Team | | | |
| | 29 | Testing Tools, Features of test tool | | | |
| | 30 | Test Plans, Test Cases, Test Data, Risk Analysis. | | | |
| 12 th | 31 | Guidelines for selecting a tool | | | |
| | 32 | Tools and skills of tester | | | |
| | 33 | Static testing tools, Dynamic testing tools | | | |
| 13 th | 34 | Advantages and disadvantages of using tools | | | |
| | 35 | Revision and Problem Solving | | | |
| | 36 | Introduction to open-source testing tool | | | |
| 14 th | 2nd Minor Test | | | | |
| 15 th | 37 | Revision and Problem Solving | | | |
| | 38 | Revision and Quiz | | | |
| | 39 | Revision and Problem Solving | | | |

Software Testing and Quality Assurance- PEC-CSE412-T

General Course Information

| | |
|---|---|
| Course Code: PEC-CSE412-T/ PEC-IT412-T Course Credits: 3 Type: Professional Elective Contact Hours: 3 hours/week Mode: Lectures (L) Examination Duration: 3 hours | Course Assessment Methods (internal: 30; external: 70) Two minor examinations (20 marks), Class Performance measured through percentage of lectures attended (4 marks), assignments (6 marks), and the end- semester examination (70 marks). For the end semester examination, nine questions are to be set by the examiner. A candidate is required to attempt 5 questions in all. All questions carry equal marks. Question number 1 will be compulsory and based on the entire syllabus. It will contain seven parts of 2 marks each. Question numbers 2 to 9 will be given by setting two questions from each of the four units of the syllabus. A candidate is required to attempt the remaining four questions by selecting one question from each of the four units. |
|---|---|

Pre-requisites: Software Engineering.

About the Course:

This course introduces students to software testing process and describes the quality assurance process and its role in software development. During the course students learn about the testing methods and tools, creating good test cases to improve the quality of software.

Course Outcomes: By the end of the course students will be able to:

- CO1. **Recall** the process of software testing life cycle and quality assurance. (LOTS: Level 1: Remember)
- CO2. **Demonstrate** reusability testing on software applications. (LOTS: Level 2: Understand)
- CO3. **Apply** software testing tools for predicting the behavior of software applications. (LOTS: Level 3: Apply)
- CO4. **Identify** the test cases for software applications. (HOTS: Level 4: Analyse)
- CO5. **Plan** test cases and quality management activities. (HOTS: Level 6: Create)
- CO6. **Predict** software quality based on quality parameters and quality models. (HOTS: Level 6: Create)

Course Content

Unit I

Introduction to Basic of software testing & Terminology, Software Development & Software Testing Life Cycle- role and activities, Necessity and Objectives of testing; Quality Concepts, Quality Control, McCall's factor model; Different Software Development Model; Object-oriented testing, Web testing, GUI testing; Elements of Software quality assurance; Quality Assurance Activities, Statistical Quality Assurance; Software Reliability, SQA plan, Quality Standards: -IEEE, CMM, ANSI.

Unit II

Testing Concepts, Issues and Techniques, Levels of Testing, Verification and Validation Model; Techniques of Verification: -Peer Review, Walkthrough, Inspection, FTR; Unit testing, Integration testing, Function Testing; System testing, Installation Testing, Usability Testing, Regression testing; Performance testing: -Load Testing, Stress Testing, Security testing, Volume testing; Acceptance testing: -Alpha testing, Beta testing, Gamma testing.

Unit III

Black Box Testing Methods: Equivalence partitioning, Boundary-value analysis, Error guessing, graph-based testing methods, Decision Table Testing; White Box Testing Methods: Statement coverage, Decision coverage, Condition coverage, Path testing, Data flow testing.

Test Planning & Documentation: Development plan and quality plan objectives; Testing Strategy, Test Management, Strategic Management, Operational Test Management, Managing the Test Team, Test Plans, Test Cases, Test Data, Risk Analysis.

Unit IV

Testing Tools, Features of test tool; Guidelines for selecting a tool; Tools and skills of tester; Static testing tools, Dynamic testing tools, Advantages and disadvantages of using tools, Introduction to open-source testing tool.

Text and reference books:

1. M. G. Limaye, *Software Testing Principles, Techniques and Tools*, TMH, 2009.
2. Yogesh Singh, *Software Testing*, Cambridge University Press, 2016.
3. Ron Pattern, *Software Testing*, 2nd edition, Sams, 2005.
4. Roger S. Pressman, *Software Engineering- a Practitioners approach*, 8th edition, McGraw Hill, 2014
5. Jeff Tian, *Software Quality Engineering: Testing, Quality Assurance and Quantifiable Improvement*, Wiley, 2005.
6. Stephan H. Kan, *Metrics and Models in Software Quality Engineering*, 2nd edition, Addison-Wesley, 2009.
7. William E. Perry, *Effective Methods of Software Testing*, 2nd edition, Wiley, 2000.