Savitribai Phule Pune University
Fourth Year of Computer Engineering (2019 Course)
Elective V
410245 (D): Software Testing and Quality Assurance

Teaching Scheme: TH: 03 Hours/Week	Credit 03	Examination Scheme: In-Sem (Paper): 30 Marks End-Sem (Paper): 70 Marks

Prerequisite Courses: Software Engineering (210253), Software Project Management(310245(D))

Companion Course: **Lab Practice IV**

Course Objectives:

- Introduce basic concepts of software testing.
- Understand the best way to increase the effectiveness, test coverage, and execution speed in software testing.
- Understand white box, block box, object oriented, web based and cloud testing.
- Understand the importance of software quality and assurance software systems development.
- Know in details automation testing and tools used for automation testing.
- To learn and understand the combination of practices and tools that are designed to help QA professionals test more efficiently.

Course Outcomes:

On completion of the course, student will be able to–

CO1: Describe fundamental concepts in software testing such as manual testing, automation testing and software quality assurance.
CO2: Design and Develop project test plan, design test cases, test data, and conduct test operations.
CO3: Apply recent automation tool for various software testing for testing software.
CO4: Apply different approaches of quality management, assurance, and quality standard to software system.
CO5: Apply and analyze effectiveness Software Quality Tools.
CO6: Apply tools necessary for efficient testing framework.

Course Contents

Unit I	Introduction to Software Testing	07 Hours

Introduction: historical perspective, Definition, Core Components, Customers suppliers and process, Objectives of Testing, Testing and Debugging, Need of Testing, Quality Assurance and Testing, Why Software has Errors, Defects and Failures and its Causes and Effects, Total Quality Management(TQM), Quality practices of TQM, Quality Management through- Statistical process Control, Cultural Changes, Continual Improvement cycle, Benchmarking and metrics, Problem Solving Techniques and Software Tools. Software Quality, Constraints of Software product Quality assessment, Quality and Productivity Relationship, Requirements of Product, Software Development Process, Types of Products, Software Development Lifecycle Models, Software Quality Management, Processes related to Software Quality, Quality Management System's Structure, Pillars of Quality Management System, Important aspects of quality management.

#Exemplar/Case Studies	1. Offshore delivery model for an Airline Company. 2. SAP test automation CoE for Financial Service Provider.

***Mapping of Course Outcomes for Unit I**	CO1

Unit II	**Test Planning and Quality Management**	**07 Hours**

Test Planning –Artifacts & Strategy, Test Organization –Test Manager & Tester Role, Test plan purpose & contents, Test Strategy and Approach, Test cases & Test Data, Test Entry-Exit criteria, Test Execution Schedule, Use case Testing, Scenario Testing, Test Monitoring & Control- Test Metrics –Test Case Productivity, Test case Coverage, Defect Acceptance & Rejection, Test Efficiency, Efforts and Schedule Variance, Test Efforts biasing Factors, Test Report & configuration Management, Quality Assurance Process, Documentation Risk & Issues. Software Quality, Quality Management Importance, Quality Best practices.

#Exemplar/CaseStudies	1. Online Recommendation System 2. Quality Engineering services for Medical Devices company | CaseStudy (cigniti.com)
***Mapping of Course Outcomes for Unit II**	CO2

Unit III	**Test Case Design Techniques**	**07 Hours**

Software Testing Methodologies: White Box Testing, Black Box Testing, Grey Box Testing. Test Case Design Techniques: Static Techniques: Informal Reviews, Walkthroughs, Technical Reviews, Inspection. Dynamic Techniques: Structural Techniques: Statement Coverage Testing, Branch Coverage Testing, Path Coverage Testing, Conditional Coverage Testing, Loop Coverage Testing Black Box Techniques: Boundary Value Analysis, Equivalence Class Partition, State Transition Technique, Cause Effective Graph, Decision Table, Use Case Testing, Experienced Based Techniques: Error guessing, Exploratory testing

Levels of Testing: Functional Testing: Unit Testing, Integration Testing, System Testing, User Acceptance Testing, Sanity/Smoke Testing, Regression Test, Retest. Non-Functional Testing: Performance Testing, Memory Test, Scalability Testing, Compatibility Testing, Security Testing, Cookies Testing, Session Testing, Recovery Testing, Installation Testing, Adhoc Testing, Risk Based Testing, I18N Testing, L1ON Testing, Compliance Testing.

Link:https://www.besanttechnologies.com/training-courses/software-testing-training/manual-testing-training-institute-in-chennai

#Exemplar/Case Studies	1. Case Study: Manual Testing (Online Marketing SoftwarePlatform) Link: https://www.360logica.com/blog/case-study-manual-testing-online-marketing-software-platform/ 2. Case Study: Decision Table Testing (transferring money online to an account which is already added and approved.)
***Mapping of Course Outcomes for Unit III**	**CO3**

Unit IV	**Software Quality Assurance and Quality Control**	**07 Hours**

Software Quality Assurance: Introduction, Constraints of Software Product Quality Assessment, Quality and Productivity Relationship, Requirements of a Product, Characteristics of Software,

Software Development Process, Types of Products, Schemes of Criticality Definitions, Software Quality Management, Why Software Has Defects? Processes Related to Software Quality, Quality Management System Structure, Pillars of Quality Management System, Important Aspects ofQualityManagement.

Software Quality Control: Software quality models, Quality measurement and metrics, Quality plan, implementation and documentation, Quality tools including CASE tools, Quality control and reliability of quality process, Quality management system models, Complexity metrics and Customer Satisfaction, International quality standards – ISO, CMM

#Exemplar/Case Studies	1. Case Study #1 – Android Application Acceptance Test Suite 2. Case Study #2 – API Acceptance Test Suite Link for above case studies - Software Quality Assurance Case Studies - Beta Breakers
***Mapping of Course Outcomes for Unit IV**	**CO4**

Unit V	**Automation Testing Tools / Performance Testing Tools**	**07 Hours**

Automation Testing: What is automation testing, Automated Testing Process, Automation Frameworks, Benefits of automation testing, how to choose automation testing tools. Selenium Automation Tools: Selenium's Tool Suite- Selenium IDE, Selenium RC, Selenium Web driver, Selenium Grid. Automation Tools: SoapUI, Robotic Process Automation (RPA), Tosca, Appium.

Performance Testing : What is Performance Testing what is use of it? Tools used for performance testing - Apache Jmeter.

#Exemplar/Case Studies	1. Case Study: Cucumber open-source automation testingframework. 2. Case Study: (PDF) Automated Software Testing—A Case Study(researchgate.net)
***Mapping of Course Outcomes for Unit V**	**CO5**

Unit VI	**Testing Framework**	**07 Hours**

Testing Framework: Software Quality, Software Quality Dilemma, Achieving Software Quality, Software Quality Assurance Elements of SQA, SQA Tasks, Goals and Metrics, Formal Approaches to SQA, Statistical Software Quality Assurance, Six Sigma for Software Engineering, ISO 9000 Quality Standards, SQA Plan, Total Quality Management, Product Quality Metrics, In process Quality Metrics, Software maintenance, Ishikawa's 7 basic tools, Flow Chart, Checklists, Pareto diagrams, Histogram, Run Charts, Scatter diagrams, Control chart, Cause Effect diagram. Defect Removal Effectiveness and Process.

#Exemplar/Case Studies	1. Case study: Software Quality In AcademicCurriculum. 2. Case study: Evaluation of an Automated Testing Framework: ACase Study (scielo.sa.cr)
***Mapping of Course Outcomes for Unit VI**	**CO6**

Learning Resources

Text Books:

1. M G Limaye, "Software Testing Principles, Techniques and Tools", Tata McGraw Hill, ISBN:9780070139909 0070139903
2. Srinivasan Desikan, Gopal Swamy Ramesh, "Software Testing Principles and Practices", Pearson,ISBN-10: 817758121X

Reference Books:

1. Naresh Chauhan, "Software Testing Principles and Practices", OXFORD, ISBN-10: 0198061846. ISBN-13: 9780198061847
2. Stephen Kan, "Metrics and Models in Software Quality Engineering", Pearson, ISBN-10: 0133988082; ISBN-13: 978-0133988086

e-Books :

1. M G Limaye, "Software Testing Principles, Techniques and Tools" https://books.google.co.in/books?id=zUm8My7SiakC&printsec=frontcover&source=gbs_ge_summary_r&cad=0#v=onepage&q&f=false
2. Srinivasan Desikan, Gopalswamy Ramesh, "Software Testing Principles and Practices" https://kupdf.net/queue/software-testing-principles-and-practices-by-srinivasan_5b0ae8eae2b6f51f7d862d26_pdf?queue_id=-1&x=1656562364&z=MTE1LjI0Mi4yNDIuNzA=
3. Naresh Chauhan, "Software Testing Principles and Practice" . https://pdfcoffee.com/download/se-4-pdf-free.html

MOOC Courses Links:

- https://nptel.ac.in/courses/106105150
- NPTEL : NOC: Software Testing (2017) (Computer Science and Engineering) (digimat.in)

@The CO-PO Mapping Matrix

CO\PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	3	1	1	2	2	-	-	1	2	1	2	1
CO2	1	3	3	2	1	-	-	1	2	1	2	-
CO3	1	-	1	2	3	-	-	-	2	1	1	-
CO4	1	1	2	3	1	1	1	2	2	2	2	-
CO5	1	2	1	2	3	1	-	-	1	1	2	-
CO6	1	2	3	2	3	1	-	-	2	1	1	-