|  |  |  |  |
| --- | --- | --- | --- |
| Course No. | Course Title | No. of Credits | Credit Hours |
| **SE 605** | **Software Testing and Quality Assurance** | **Theory: 2** | **14 × (2×1) = 28** |
|  |  | **Lab: 1** | **14 × (1×2) = 28** |

**Introduction of the Course:**

Software testing is particularly important in Software Development Life Cycle (SDLC) since it ensures customer’s reliability and satisfaction towards the application. Due to its importance, testing is often considered as a process which is parallel to every software development activities. The testing process can be broadly planned into two activities – Verification and Validation (V&V). A systematic realization of these activities can enhance software quality to a great extent. The aim of this course is to illustrate these aspects of software testing as a methodical process to produce quality software. It is expected that students, at the end of this course, will have a clear understanding of the foundations, methodologies and tools in the area of software testing.

**Specific Objective:**

* To learn fundamental concepts in software testing, including software testing objectives, process, criteria, strategies, and methods.
* To discuss various software testing issues and solutions in software unit test; integration, regression, and system testing.
* To learn how to planning a test project, design test cases and data, conduct testing operations, manage software problems and defects, generate a testing report.
* To gain software testing experience by applying software testing knowledge and methods to practice-oriented software testing projects. To gain the techniques and skills on how to use modern software testing tools to support software testing projects.
* To understand software test automation problems and solutions.
* To learn how to write software testing documents, and communicate with engineers in various forms.

**Course Contents: Theory:**

The Psychology and Economics of Software Testing, Software Testing Life Cycle (STLC), Software Testing Terminology and Methodology, V&V Model, Dynamic Black Box Testing – Boundary Value Analysis, Equivalence Partitioning, State Transition based Testing, Decision Table based Testing, Cause-Effect Graphing based Testing and Error Guessing, Dynamic White Box Testing – Basis Path Testing, Data Flow Testing and Mutation Testing, Inspections, Walkthroughs, Technical Reviews, Unit Testing, Integration Testing, Function Testing, System Testing, Acceptance Testing, Regression Testing, Test Management – Test Organization, Test Plan, Test Design and Specifications, Software Metrics, Software Quality, Quality Control and Quality Assurance, Quality Management and Project Management, Software Quality Metrics, Testing Internet Applications - Security and Performance Testing, Debugging, Test Driven Development (TDD), Behavior Driven Development (BDD).

**Lab:**

Application of different testing tools – Bugzilla, JUnit, Selenium, Apache JMeter, Sprajax, Sqlninja, Cucumber or anything similar, and a Project. For the project, the students will be divided into small groups having at most 3 members and a class project will be given to them for preparing a system test case. They must validate the requirements and create Mock UIs during the preparation of test cases. Besides, each of the students will relate their learnings on unit, regression, performance and security testing, debugging, behavior driven development via different tools.

**Unit-wise Course Curriculum: Theory:**

| **Unit** | **Title of the Topics** | **Learning Outcome** | **Number of Theory Credit Hours** |
| --- | --- | --- | --- |
|  | The Psychology and Economics of Software Testing, Software Testing Life Cycle (STLC) | Student will know about Software Testing paradigm | 2 |
|  | Software Testing Terminology and Methodology, V&V Model | Student will learn the basics of Software Testing | 2 |
|  | Dynamic Black Box Testing – Boundary Value Analysis, Equivalence Partitioning, State Transition based Testing, Decision Table based Testing, Cause-Effect Graphing based Testing and Error Guessing | Student knows and can do black box testing | 6 |
|  | Dynamic White Box Testing – Basis Path Testing, Data Flow Testing and Mutation Testing, Inspections, Walkthroughs, Technical Reviews | Student will know how to do white box testing | 6 |
|  | Unit Testing, Integration Testing, Function Testing, System Testing, Acceptance Testing, Regression Testing | Student will learn different categories of testing and know which one to apply when | 6 |
|  | Test Management – Test Organization, Test Plan, Test Design and Specifications | Student will learn to manage test data and metadata | 1 |
|  | Software Metrics, Software Quality, Quality Control and Quality Assurance, Quality Management and Project Management, Software Quality Metrics | Software quality is understood by students | 2 |
|  | Testing Internet Applications - Security and Performance Testing, Debugging | Security Testing is learnt by students, as well as they learn to debug applications | 2 |
|  | Test Driven Development (TDD), Behavior Driven Development (BDD). | Student will learn TDD and BDD testing | 1 |
| **Total Number of Theory Credit Hours:** | | | **28** |

**Unit-wise Course Curriculum: Lab:**

| **Unit** | **Title of the Topics** | **Learning Outcome** | **Number of Lab Credit Hours** |
| --- | --- | --- | --- |
|  | Review of Software Testing Concept, Teach about bugs in software and reporting bugs in ***Bugzilla*** | Student will learn reporting bugs in Bugzilla | 3 |
|  | Learn Unit and Integration Test. Write Unit and Integration Tests in ***JUnit*** | Student will learn to write tests in Junit | 3 |
|  | Learn Performance, Stress and Load Test. Run these tests using ***Apache JMeter*** | Student will learn to run tests using JMeter | 3 |
|  | Learn SQL Injection Security Exploit. Run ***Sqlninja/Sqlmap*** to find SQL vulnerabilities | Student will learn about Sql injection and tools to avoid those | 3 |
|  | Learm about BDD. Learn ***Cucumber*** to do BDD testing | Student will learn BDD and Cucumber tool | 4 |
|  | Learn about automatic web testing. Learn ***Selenium*** tool for such kind of testing | Student will learn automated testing | 6 |
|  | Learn Web application security. Learn ***Burpsuite*** for graphical representation of such testing outcome. | Student will learn web application security and Burpsuite | 6 |
| **Total Number of Lab Credit Hours:** | | | **28** |

**Instructional Strategies:**

* The medium of instruction is English
* Lecture materials: recommended books, ppt files, and documents
* Discuss experimental results to learn analyzing techniques (using lab results)

**Assessment:**

Continuous Evaluation – 33%

(Attendance, Midterm Examination and Assignments/Reports)

Lab – 33%

(Application of different testing tools and Project)

Final – 34%

Total 100

**Prerequisite:** Structured Programming (CSE 101), Discrete Mathematics (CSE 102), Sociology (GE 105), Data Structure and Algorithms (CSE 201), Theory of Computing (SE 205), Object Oriented Concepts (SE 206, 306), Software Requirements Specification and Analysis (SE 406), Database Management Systems (CSE 404, 504), Web Technology (CSE 502), Design Patterns (SE 506).

**References:**

1. Naresh Chauhan, Software Testing: Principles and Practices, 1st or higher Edition, Oxford University Press.
2. Glenford J. Myers, Corey Sandler, and Tom Badgett. The Art of Software Testing, 3rd or higher Edition, John Wiley & Sons.
3. Lisa Crispin and Janet Gregory. Agile Testing: A Practical Guide for Testers and Agile Teams, 1st or higher Edition, Pearson Education.