

CSE376:AUTOMATED TESTING

L:2 T:0 P:2 Credits:3

Course Outcomes: Through this course students should be able to

CO1 :: understand various kinds of Software Testing tools for software quality control.

CO2 :: compare manual and automated software testing techniques.

CO3 :: use bug tracking tool to report bugs.

CO4 :: use testing tools for unit testing, code coverage and website testing.

CO5 :: describe the basics of Jenkins, TestNG, JUnit 5 and Gherkin.

CO6 :: use Maven to create reports, checks and testing automation setups.

Unit I

Automated Testing and Test Tools : The Benefits of Automation and Tools, Test Tools, Software Test Automation, Random Testing, Realities of Using Test Tools and Automation

Unit II

Bug Reporting : Getting Your Bugs Fixed, Isolating and Reproducing Bugs, Not All Bugs Are Created Equal, A Bug's Life Cycle, Bug-Tracking Systems, Manual Bug Reporting and Tracking, Automated Bug Reporting and Tracking, Introduction to Bugzilla

Unit III

Test Administration : Goal of Test Planning, Test Planning Topics, Goals of Test Case Planning, Test Case Planning Overview, Test Case Organization and Tracking, Test case Design, Building Manual test cases, Building Automated test cases using Java Framework tools

Unit IV

Various Testing Frameworks Available in Java : Testing Tools Selection Criteria, Fundamentals of Jenkins, TestNg, Maven, JUnit 5, and Gherkin, Introduction to JUnit framework, Define a test in JUnit, JUnit naming conventions, JUnit test suites, Example JUnit test, JUnit code constructs, Eclipse support for JUnit, Installation of JUnit

Unit V

Structural testing using automated tool : Structural testing, Data and Code Coverage, Statement Coverage, Branch Coverage, Path Coverage, Other Coverages and Understanding Their Differences, Introduction to Code Coverage Tool Eclemma, Interpreting results of Eclemma, Cyclomatic Complexity

Unit VI

Test Automation and Management/Reporting Frameworks : Introduction to Maven, Benefits and features of Maven, Activities managed by Maven, Architecture of Maven, Installing/Configuring Maven, Creating Maven Project, Understanding POM.xml file, Adding Dependencies to POM.xml, Introduction to TestNG, TestNG Vs JUnit, Downloading and Installing TestNG, Creating Test Cases using TestNG Annotations, Creating Reports using TestNG, Understanding Annotations of TestNG, Need of TestNG in Selenium, Running the Test, Checking reports created by TestNG, Generating HTML Reports, Annotations used in TestNG, Validating Tests with Assertions, Creating multiple Tests, Prioritizing Tests

List of Practicals / Experiments:

List of Practicals

- Creating manual test cases
- Understanding bug reporting process
- Test case Design
- Installing JUnit
- Executing test cases using JUnit
- Automated test cases using Java Framework tools
- Installing Eclemma
- Executing test cases using Eclemma

- Installing maven
- Configuring maven with junit on eclipse
- Installing TestNG
- Configuring maven with TestNG on eclipse

Text Books: 1. SOFTWARE TESTING by RON PATTON, PEARSON

References: 1. FOUNDATIONS OF SOFTWARE TESTING by ADITYA P MATHUR, PEARSON
2. SOFTWARE TESTING PRINCIPLES AND PRACTICES by SRINIVASAN DESIKAN, GOPALASWAMY RAMESH, PEARSON