**Software Testing Course Outline**

**FAST-NU, Lahore**

|  |  |
| --- | --- |
| **Course Code** | CS497 |
| **Course Title** | Software Testing |
| **Credit Hours** | 3 |
| **Prerequisite** |  |
| **Grading Criteria** | Quizzes (10%), Assignments + Class Activities (25%), Mid Terms (25%), Final Exam (40%) |
| **Semester** | Spring 2022 |
| **Class and Exam Schedule** | See TimeTable Exam: See date sheet |
| **Course Instructor** | Lehmia Kiran [lehmia.kiran@nu.edu.pk](mailto:zeeshan.rana@nu.edu.pk) |
| **Instructor Office Hours** | **TBD** |
| **Course TA** | TBD |
| **Plagiarism Policy** | All the parties involved will be awarded negative or Zero in first instance. Repeat of the same offense will result in (F) grade. |
| **Textbook(s)** | Naik and Tripathy, Software Testing and Quality Assurance: Theory and Practice. Wiley 2008 |
| **Reference Material** | 1. Code Complete by Steve McConnel (2nd Edition) 2. A Practitioners Guide to Software Test Design by Lee Copeland 3. Software Testing: A Craftsman’s Approach by Paul C. Jorgensen 4. Anne MetteJonassen Hass, *Guide to Advanced Software Testing*, Artech House, 2008. |
| **Course Goals** | * Familiarize the students with the terms, software quality and software testing. * Introduce Software Quality Assurance Process and its steps to students * Explain complete process of testing to students * Familiarize the students with common methods used for testing * Familiarize the students different methods used for test case selection. * Familiarize students with software testing tools. |
| **Learning Outcomes** | After successful completion of the course, the students will be able to:   1. List different steps of a Software Quality Assurance Program. 2. Differentiate between black box and white box testing. 3. Design test cases for black box and white box testing. 4. Select appropriate number of test cases using an appropriate strategy. 5. Execute test cases using software testing tools. 6. Understand Software Testing Process |
| **Programming Assignments Done in the Course** | Yes |

Tentative Topics and Course Plan (might be slightly changed)

|  |  |  |
| --- | --- | --- |
| Week # | Lecture # | Topics Covered |
| 1 | 1 | **Course Introduction.**  **Software Lifecycle. Software development processes.** Where does testing phase fit in? |
| 2 | **Software Quality Landscape:** What is quality? Characteristics of quality. Introduction of Defect Detection Techniques. Relationship among the quality characteristics, Types of quality characteristics, Improving Quality, Effectiveness of Defect Detection Techniques. General Quality Principle. Significance of testing. Test case Design. |
| 2 | 3 | **Unit Testing:** Debugging. |
| 4 | **White box Testing:** Structural Testing, Basis Path Testing, Control Flow graph, Cyclomatic Number, Selection of minimum number of test cases, Test coverage (EclEmma, JUnit) |
| 3 | 5 | **White box Testing:**Structural Testing, Basis Path Testing, Control Flow graph, Cyclomatic Number, Selection of minimum number of test cases, Test coverage |
| 6 | **White box Testing:**Structural Testing, Basis Path Testing, Control Flow graph, Cyclomatic Number, Selection of minimum number of test cases, Test coverage |
| 4 | 7 | **White box Testing:** Data flow testing |
| 8 | **Black box testing:** Functional Testing, GUI Testing (SilkTest/Abbot) |
| 5 | 9 | **Black box testing:** Equivalence Class Partitioning |
| 10 | **Black box Testing:**  Boundary Value Analysis, Domain Analysis Testing |
| 6 | **Midterm 1** | |
| 7 | 11 | **Black box testing:** Decision Table based testing, State transition testing. |
| 12 | **Black box testing:** Pairwise Testing, Orthogonal Testing |
| 8 | 13 | **Black box Testing:**  Orthogonal Testing |
| 14 | **Black box Testing:**  Use Case based Testing |
| 9 | 15 | **Levels of Testing:**Unit Testing, debugging, diagnosis. Integration Testing. Big Bang, Top Down, Bottom UP, Call Graph based |
| 16 | **Levels of Testing:** Integration Testing. Integrating Component/Off-the-shelf components |
| 10 | 17 | **Levels of Testing:** System Testing, Performance Testing, Load and Stress Testing, Security Testing, Usability Testing |
| 18 | **Levels of Testing:** Regression Testing. Acceptance Testing. |
| 11 | 19 | **Testing Process. Test Documentation** |
| 20 | **Software Testing Tools: Automated Testing. Selenium.** |
| 12 | **Midterm 2** | |
| 13 | 21 | **Software Testing Tools:** |
| 22 | **Presentations** |
| 14 | 23 | **Presentations** |
| 24 | **Presentations** |
| 15 | 25 | **Presentations** |
| 26 | **Presentations** |
| 16 | **Final** | |