MCA418: Software Quality Assurance &Testing	
Teaching Scheme Lectures: 3 hrs/Week Tutorials: 1 hr/Week Credits: 4	Examination Scheme Class Test -12Marks Teachers Assessment - 6Marks Attendance – 12 Marks End Semester Exam – 70 marks

Prerequisite: - programming languages, software engineering.

Course Objectives:

The objectives of this course are

1. To study fundamental concepts in software testing, including software testing objectives, process, criteria, strategies, and methods.
2. To highlight the strategies for software testing and understand the various types of black box and white box testing methods.
3. To discuss various software testing issues and solutions in unit testing, integration, regression, and system testing
4. To identify the issues in testing management and understand test planning.
5. To gain the techniques and skills on how to use modern software testing tools to support software testing projects.

Detailed Syllabus:

UNIT I (6 Hours) Software Quality Assurance: Software crisis, Birth of software engineering, Why Software engineering, Criteria for the success of a software project, phases in SDLC, Software Quality Assurance, Quality Management Systems.
UNIT II (10 Hours) Software Testing Process: Verification and Validation, Cost of Quality, Why Testing is difficult, Levels of testing-Unit Testing, Module Testing, Integration and System Testing, Acceptance Testing, Testing Approaches: Top-down versus Bottom-up, Functional versus Structural testing, Mutation testing, Regression Testing, Types of Testing, Manual Testing and its Limitations.

Head
Department of Computer Applications
Faculty of Computer Applications
Invertis University, Bareilly (UP)

Dean-Academic
Faculty of Computer App
Invertis University, Bare

UNIT III (10 Hours)
Software Testing Tools: Need for Automated Testing Tools, Taxonomy of testing tools, Functional/Regression Testing Tools, Performance Testing tools, Testing Management Tools, Source Code Testing Tools, How to select a Testing Tool?

UNIT IV (12 Hours)
WinRunner: Overview, Testing an application using WinRunner, TestScript Language(TSL), GUI MAP file, Synchronization of Test cases, Data driven testing, Checking GUI objects.

UNIT V (12 Hours)
SQA Robot: overview, testing an application, Synchronization of Test procedures, creating checkpoints.TestDirector: overview, testing management process, managing the testing process using TestDirector.

UNIT VI (6 Hours)
Source Code Testing Utilities in Unix and Linux Environnement: GNU tools, Timings of programs, Profiler, Code optimization, Productivity tools, Portability Testing Tool, Configuration Management Tools, Coding Guidelines and Standards.

Text and Reference Books

1. "Effective Software Testing", Elfriede Dustin, Pearson Education, IV edition.
2. "Software Testing Concepts and Tools", N. R. Pusuluri, Dreamtech press, 2008.
3. "Automated Software Testing", Jeff Rashka, John Paul and E. Dustin, Pearson
4. Education, 2008.
5. "Effective Methods for Software Testing", W. E. Perry, Wiley-India, III edition.

Course Outcomes:

After completing the course, students will be able to:

1. Have an ability to apply software testing knowledge and engineering methods. Have an ability to design and conduct a software test process for a software testing project.
2. Have an ability to understand and identify various software testing problems, and solve these problems by designing and selecting software test models, criteria, strategies, and methods
3. Have an ability to design and conduct various types and levels of software testing for a software project.
4. Have basic understanding, knowledge of contemporary issues in software testing and test planning. Have an ability to use various communication methods and ethical skills to communicate with their teammates to conduct their practice-oriented software testing projects.
5. Have an ability to identify the needs of software test automation, and define and develop a test tool to support test automation.