UTAH SYSTEM OF HIGHER EDUCATION

Utah System of Higher Education
Software Quality Assurance
FY2027 / 12 Credits (360 Clock-Hours)

Foundational Courses

TEAQ 1000 Introduction to Quality Assurance — 2 Credits / 60 Clock-Hours

This course introduces students to the principles and practices of software quality assurance (QA) and its essential role in the software development lifecycle. Students will learn how to design and execute manual tests, document results, and manage defects using industry-standard methods and tools. The course emphasizes test planning, test case development, and the use of QA systems for tracking, version control, and automation. Upon completion, students will understand the core concepts, terminology, and workflows that underpin effective software quality assurance.

Objectives:
• Explain the purpose and role of quality assurance within the software development process.
• Create clear, organized test plans and test cases aligned with project requirements.
• Describe and differentiate among testing types, including unit, integration, system, and acceptance testing.
• Collaborate using industry tools such as bug tracking systems, version control platforms, and automated testing frameworks.
• Apply common QA terminology and best practices to evaluate and improve software quality.

TEAQ 1015 Backend Testing — 2 Credits / 60 Clock-Hours

This course introduces students to the principles and practices of backend quality assurance, focusing on the validation of databases, APIs, and system performance. Students learn to design and execute tests that ensure the reliability, accuracy, and scalability of backend systems supporting modern applications. Through hands-on practice, they gain experience with SQL-based data verification, API functionality, and performance evaluation using industry tools and standards.

Objectives:
• Explain core database concepts, including tables, relationships, and keys.
• Write SQL queries to retrieve, update, and validate data.
• Describe the role and structure of APIs within software systems.
• Perform API testing, including endpoint validation, request and response handling, and authentication verification.
• Apply tools and methodologies to assess API functionality, reliability, and security.
• Conduct basic performance tests to evaluate backend system responsiveness and stability.
• Analyze test results and identify potential issues affecting backend system performance or data integrity.

TEAQ 1018 Automated Testing Essentials — 3 Credits / 90 Clock-Hours

This course introduces the foundational technical skills required for software test automation and team collaboration. Students learn the fundamentals of JavaScript for testing web applications, including working with HTML structures to create selectors, and element locators. The course also covers version control using Git, focusing on managing code, tracking changes, and collaborating within development teams. By the end of the course, students will be able to apply these tools and techniques to support automated testing.

Objectives:
- Write and interpret basic JavaScript code for automated testing scenarios.
- Analyze HTML elements, tags, and attributes relevant to testing web interfaces.
- Identify and use selectors such as CSS and XPath to locate and validate web elements.
- Explain fundamental Git concepts, including repositories, commits, and branching.
- Understand and use version control for collaborative code management.
- Integrate JavaScript, selectors, and Git into automated testing and quality assurance processes.

TEAQ 1020 Automated Testing — 3 Credits / 90 Clock-Hours

This course provides students with the knowledge and skills to design, develop, and execute automated tests for software applications. Emphasis is placed on selecting and implementing automation frameworks that enhance efficiency, consistency, and accuracy in quality assurance. Students gain hands-on experience creating maintainable test scripts and interpreting test results to support continuous integration and delivery practices.

Objectives:
- Explain the role and value of automated testing in modern software development and quality assurance.
- Configure and implement a basic automated testing framework.
- Develop and execute automated test scripts to validate functionality and behavior.
- Apply best practices and design patterns to write scalable, maintainable test automation code.
- Use industry-standard automation tools and frameworks to support testing across diverse applications.

TEAQ 1050 Capstone Project — 2 Credits / 60 Clock-Hours

The Automated Testing Capstone Project provides students with a culminating experience to apply the concepts, tools, and techniques learned throughout the quality assurance program. Students design, implement, and present an automated testing solution that reflects real-world software development practices. The course emphasizes project planning, problem solving, version control, and use of automation frameworks. By completion, students will have produced a professional portfolio project demonstrating their technical competence and readiness for industry roles in software quality assurance.

Objectives:
- Define the goals, scope, and deliverables of an automated testing project.
- Select and configure an appropriate automation framework to support project requirements.
- Design, develop, and execute automated test cases that validate system functionality and behavior.
- Use version control tools to manage and document project code.
- Analyze and interpret test results to evaluate software quality and communicate findings effectively