



## Foundational Courses

### TEAQ 1010 Introduction to Quality Assurance

**3 Credits / 90 Clock-Hours**

The Introduction to Quality Assurance course is designed to provide an overview of software quality assurance and its role in the software development process. The course will cover the fundamentals of manual QA, including test plan creation, test case design, and defect management. The course will introduce students to common tools used in software quality assurance, such as bug tracking systems, version control systems, and automated testing tools. By the end of the course, students have an understanding of software quality assurance principles and practices.

Objectives:

- Explain the role of QA in the software development process.
- Create effective test plans and test cases.
- Describe the different types of testing, including unit testing, integration testing, system testing, and acceptance testing.
- Collaborate using common software quality assurance tools, including bug tracking systems, version control systems, and automated testing tools.
- Summarize commonly used industry terms and practices.

### TEAQ 1011 Databases and SQL Queries

**1 Credit / 30 Clock-Hours**

The Databases and SQL Queries course aims to equip students with an understanding of database management systems and SQL (Structured Query Language). Students learn how to create, maintain and manipulate databases, as well as how to write SQL queries to extract, modify and analyze data. The course covers a range of topics including data modeling, database normalization, indexing, data integrity, and data security.

Objectives:

- Explain the concepts and principles of database management systems.
- Discuss the fundamentals of relational database design and normalization to build efficient and scalable databases.
- Execute SQL queries to extract and manipulate data from databases.
- Optimize database performance through indexing and other techniques.
- Ensure data integrity and security through the use of constraints and other measures.
- Troubleshoot and debug common issues in database management and SQL queries.

### TEAQ 1013 Automated Testing

**4 Credits / 120 Clock-Hours**

The Automated Testing course focuses on the development and implementation of automated testing strategies and tools for software applications. Students learn how to design, develop and execute automated tests for software applications, including unit tests, integration tests, and end-to-end tests. The course covers a range of topics, including test automation frameworks, scripting languages, continuous integration, and best practices in automated testing.

Objectives:

- Describe the importance of automated testing in software development and quality assurance.
- Explain different types of automated tests, including unit tests, integration tests, and end-to-end tests.
- Design and develop automated tests using industry-standard test automation frameworks.
- Set up and integrate automated tests into a continuous integration (CI) pipeline using industry standard tools.
- Write maintainable and scalable test automation code using best practices and design patterns.



**Utah System of Higher Education**  
Software Quality Assurance  
FY2026 / 10 Credits (300 Clock-Hours)

UTAH SYSTEM OF  
HIGHER EDUCATION

**TEAQ 1014 API Testing**

**2 Credits / 60 Clock-Hours**

The API Testing course focuses on testing the performance and functionality of software applications. Students learn how to evaluate the speed, scalability, stability, and reliability of software systems, as well as how to test their API endpoints. The course covers a range of topics including load testing, stress testing, spike testing, and endurance testing, as well as API testing tools and techniques.

**Objectives:**

- Articulate the importance of performance and API testing in software development and quality assurance.
- Analyze and interpret performance test results and identify performance bottlenecks in software applications.
- Explain API testing concepts, such as endpoints, request/response types, and authentication methods.
- Design and execute API tests using industry standard testing frameworks.
- Evaluate the quality of API endpoints, including status codes, and error handling.