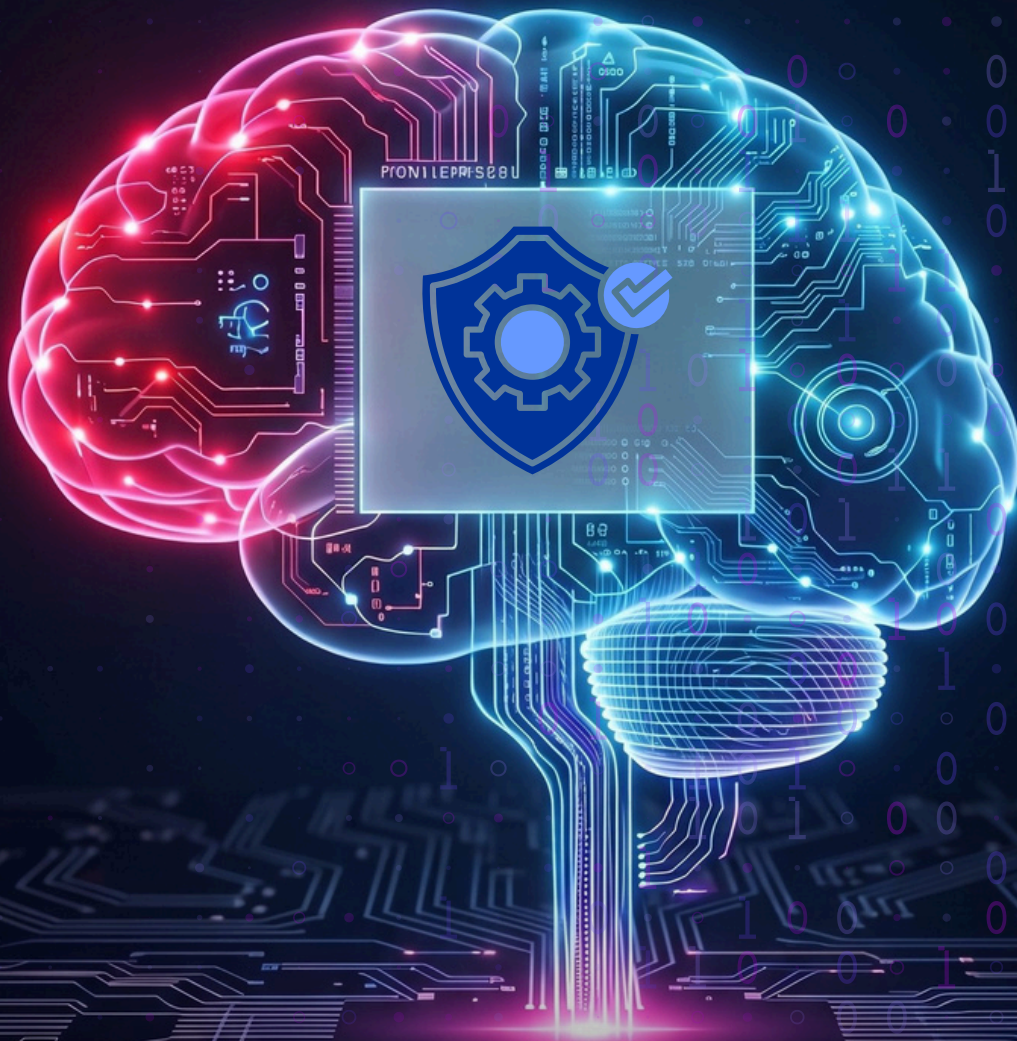


AI integrated Software QA



ABOUT THE PROGRAM

This course is designed for students with basic programming knowledge and prepares them for modern QA, SDET, and AI-QA roles.

It has 8 modules (containing several topics) and a capstone project.

Module-1

QA Fundamentals & Prompt Engineering

Module-2

Programming Fundamentals for QA

Module-3

Manual Testing + AI-Driven Exploratory Testing

Module-4

Web Test Automation with AI

Module-5

API Testing & AI-Assisted Validation

Module-6

AI in CI/CD, Cloud & DevOps for QA

Module-7

Performance & Security Testing with AI

Module-8

Testing AI & ML Systems and Test Optimization

Module 1: QA Fundamentals (2 weeks)

- Traditional: SDLC, STLC, Agile/Scrum, Jira, Writing Bug Reports, Test artifacts (Test Plan, Test cases, RTM)
- AI-Assisted Documentation: Using AI to review test plans, format bug reports and summarize sprint results.
- Prompt Engineering for QA: How to prompt ChatGPT/Claude to generate user stories, acceptance criteria, and comprehensive test cases from a requirement doc.

Module 2: Programming Fundamentals for QA (Python)

- Basic: Python basics (Variables, Loops, Functions), OOP (classes, objects, inheritance)
- Code Interpretation: Using LLM to explain legacy code.
- Code Generation: Learning to review and debug AI-generated code (students must learn to read code as well as write it), Refactoring test code with AI feedback

Module 3: Manual Testing + AI-Driven Exploratory Testing

- Test design techniques: Boundary Value Analysis, Equivalence Partitioning, Decision tables
- Exploratory testing strategies
- Risk-based testing
- AI-suggested edge cases
- Session-based exploratory testing using AI prompts
- AI-generated test charters

Module 4: Web Test Automation with AI & Prompt Engineering

- Basic: Selenium/Playwright architecture, Locators and waits, Details of the DOM structure, Page Object Model. (Unit testing)
- Auto-generating automation scripts, AI-assisted debugging
- Self-Healing Scripts: Introduction to tools like Healenium or Mabl that automatically fix broken locators when the UI changes.
- Visual Regression Testing: Using Appliflow (Visual AI) to detect UI layout issues that standard code misses.

Module 5: API Testing & AI-Assisted Validation

- Basic: REST API, HTTP methods (GET, POST, PUT, DELETE), Postman, Status codes, JSON parsing.
- Database and Intelligent validation: Database and SQL queries
- Automated API Test Gen: Using Gen-AI to generate edge-case payloads (e.g., asking AI to "generate 50 malicious JSON inputs for a login endpoint").
- Anomaly detection: AI-based anomaly detection in responses.

Module 6: AI in CI/CD, Cloud & DevOps for QA

- Basic: CI/CD pipelines (GitHub Actions / Jenkins), Version control (Git), Docker basics
- AI-driven pipeline optimization
- Predictive Test Selection: Using AI to determine which tests to run based on the code that changed (saving time vs. running the whole suite).
- Log Analysis: Using AI tools to parse server logs and identify the root cause of a crash automatically.

Module 7: Performance & Security Testing with AI

- Basic: Performance testing concepts, JMeter, Load, stress, spike testing, Basic security testing (OWASP Top 10)
- AI-based performance trend analysis
- Predictive bottleneck detection
- AI-assisted vulnerability scanning concepts

Module 8: Testing AI & ML Systems and Test Optimization

- Basic: Test case prioritization, Test coverage optimization, Defect prediction models, Flaky test detection, risk-based testing, Challenges in testing AI systems
- ML Model validation
- Bias and fairness testing
- Using historical test data
- Intro to predictive analytics
- E.g. Testing recommendation systems, Chatbot testing, Image/text model testing

Capstone Project

Students should complete a project that proves they can handle the new landscape.

- The "Self-Healing" Suite: Build a Playwright automation suite for a demo e-commerce site that incorporates a self-healing mechanism (using a library or API) so the test passes even if IDs change.
- The Chatbot Auditor: Build an automated test suite using Python that sends 100 confusing prompts to a custom chatbot and uses a secondary LLM (as a judge) to score the chatbot's responses for accuracy and safety.

Category	Traditional Tool	AI-Integrated Tool / Skill
Coding	VS Code	GitHub Copilot / Cursor
Web UI	Selenium / Cypress	Applitools / Testim / Mabl
API	Postman	Postman AI Assistant
Unit Test	PyTest	CodiumAI (AI test generation)
Model Eval	(None)	DeepEval / LangSmith
Management	Jira	Jira Product Discovery (AI)