## . Ginnovadeas

#### CASE STUDY

Evolution of Test Automation Landscape for Enhanced Quality Assurance

#### **OVERVIEW**

The client, a leading software development company, faced challenges in ensuring consistent software quality and timely releases across their applications.

They had initially implemented a basic test automation framework using Selenium with C# and NUnit for conducting sanity and regression tests.

However, as the complexity of their applications grew, they recognized the need for a more comprehensive and robust automation solution to address the evolving testing requirements

#### OBJECTIVE

The primary goal of the project was to enhance the test automation landscape (TAL) by transitioning from a basic Selenium-NUnit framework to a more advanced enterprise framework.

This transition aimed to improve test coverage, data handling, database validations, and API automation.

Additionally, the client sought to establish a CI/CD pipeline to streamline testing and ensure faster feedback loops.

### AT A GLANCE

#### Challenges

- Limited Test
- Coverage Data
- Management
- Backend Validation
  Scalability Collaboration Gap

#### Benefits

- Enhanced Test Coverage
- Data Integrity
- Enabled Backend
- Validations Scalability and
- Efficiency CI/CD Pipeline



"Working with Innovadeas has been an absolute game-changer for our testing processes. Their dedication to delivering streamlined testing solutions has not only saved us time but has also ensured faster feedback loops, allowing us to iterate and improve our products more rapidly. Their expertise and attention to detail have been invaluable, and we look forward to continuing our partnership for future projects"

#### SOLUTION

The solution involved a phased approach to evolve the TAL and address the identified challenges.

#### **Framework Enhancement**

Framework Evaluation: The team conducted a thorough evaluation of available automation frameworks and decided to adopt an enterprise-level framework based on Behaviour-Driven Development (BDD) principles. Framework Setup: The new framework was set up, leveraging tools such as SpecFlow for BDD-style test scenarios and integration with Selenium for UI automation.

Step Definition Refactoring: The existing step definitions from the legacy framework were refactored and adapted to the new BDD framework's syntax.

API Automation Integration: The new framework was extended to incorporate API automation using libraries for enhanced backend testing.

#### **Data Management and Backend Validation**

Data Handling: A data management strategy was developed, allowing tests to generate, manipulate, and clean up test data effectively.

Database Validations: Custom database validation utilities were integrated into the framework to verify backend behaviour and data integrity.

#### **Pipeline Setup and Integration**

CI/CD Implementation: An automated CI/CD pipeline was established using Azure devops, ensuring continuous integration, test execution, and deployment.

Parallel Execution: To address scalability concerns, test suites were divided and executed in parallel, reducing test execution times.

#### **Training and Documentation**

Team Training: All team members were provided with comprehensive training on the new framework's features, including BDD principles, data handling, and backend validation.

Documentation: Detailed documentation was created to aid team members in understanding the new framework's architecture, best practices, and integration procedures.

#### CONCLUSION

The successful evolution of the Test Automation Landscape from a basic Selenium-NUnit framework to a comprehensive enterprise framework equipped with BDD principles, data handling, database validations, and API automation resulted in improved software quality and accelerated release cycles for the client. The enhanced TAL now serves as a robust foundation for the organization's quality assurance efforts, enabling efficient and thorough testing of their applications while promoting collaboration and innovation.

# **02 PHASE**



PHAS

