

Quality engineering

Nash
Tech.



Case study:

Robot Framework reduces testing time and effort for leading insurance broker

Industry: Insurance

Markerstudy Group, one of the largest insurance brokers in the UK, needed to improve the efficiency of its regression testing cycles. These cycles were long and required manual effort, impacting its ability to deliver software updates swiftly. NashTech developed an automated testing framework using Robot Framework, reducing testing time and effort by 90%. This allowed non-technical staff to manage and execute automated tests, freeing up the team for higher-value tasks.

- 90% reduction in time and effort for testing
- Non-technical staff able to manage and execute automated tests



The collaboration between our onshore and offshore teams has not only improved the stats, figures and efficiency, but it has also changed the perception within the business.

Broker Systems Consultant
Markerstudy Group

Adopt a quality-first approach when designing, building and deploying secure, user-friendly software across mobile applications, cloud and edge/IoT platforms

Quality engineering is essential to designing and building secure and user-friendly software. NashTech quality engineering services integrate quality through each stage of the software development lifecycle (SDLC), accelerating time to market and improving resilience across digital systems.

Quality is not an afterthought. Our experts carry out automated testing early in the project lifecycle, with seamless integration into CI/CD pipelines to reduce rework and enable fast delivery. We ensure applications are accessible, secure and compatible with your channels.

We also make it easier to achieve quality by integrating it from the outset.

Our quality engineering solutions cover functional testing, performance engineering and end-to-end security testing across web, mobile, desktop and embedded platforms. We offer full-stack validation across unit, integration, system and user acceptance levels. We also support modern architectures including cloud-native, hybrid and legacy systems.

Solve your software quality challenges



Limited automation

It can be harder to scale and deliver measurable value when automation is not used effectively, or where it is implemented poorly.



Slow feedback in CI/CD pipelines

Testing needs to be integrated properly otherwise CI/CD pipelines could become slower with failed tests, unstable environments and long execution cycles.



Fragmented testing processes

Many organisations fail to standardise testing across different teams such as development, quality assurance (QA) and DevOps, which can lead to delays and quality gaps.



Security testing

Security testing is often isolated or overlooked, so you might miss risks or fall short of regulatory requirements.



Expensive rework

Late-stage testing makes it harder to identify design issues and defects. When caught later in development, these are expensive and can take longer to fix.

Quality engineering services for each stage of the software development lifecycle (SDLC)



CI/CD test optimisation

We optimise the testing process with prioritisation, flakiness control and parallel execution. This ensures tests are completed faster and your software is ready for release earlier.



Unified quality orchestration framework

Adopt a standardised quality framework so your teams can follow the same processes no matter the tools they are using. This helps to improve visibility and governance throughout the SDLC.



Security and compliance test packs

Our experts build security validations in line with OWASP, GDPR and PCI-DSS into functional and API tests to catch any compliance risks early on.



AI-powered requirement analysis

We use AI to analyse user stories early in the design/development stages to test gaps, improve requirement clarity and reduce rework.



Automation-as-a-product delivery model

We treat automation as a product with versioning, metrics and ROI tracking, so automation grows with your applications and is health tested.

Adopt a strategic approach to quality engineering

Quality is not limited to being a testing function. It is also crucial to delivering scalable, reliable and accessible software. At NashTech, we approach quality strategically to reduce compliance risks and enable faster releases tailored for multiple channels.

Fast delivery

We embed quality early on, so we can spot any defects quickly – and fix them – helping you launch more reliable applications.

Data-driven quality

Your teams can make data-driven quality decisions across all stages of production using centralised orchestration and real-time dashboards.

Enhance the user experience

Be confident your applications are easy to use and reliable. We test across cloud, mobile and edge environments.

Higher ROI on automation

Deliver consistent value and efficiency across software releases with our scalable frameworks and automation health metrics.

Reduced risk

Our experts will continually monitor your systems within QA cycles to ensure they are secure and identify risks before they develop into issues.

Our quality philosophy ensures your software shines

We provide strategic planning, robust test automation and seamless CI/CD integration. Our technical expertise is anchored in C#, Java, Scala and Node.js. but we can learn and master additional technologies rapidly to best align with your specific project needs.

- **Harness the power of AI and automation.** We possess deep expertise in test automation and framework development.
- **Increase operational efficiency and ROI.** Intelligent building blocks, test reuse and AI enable you to reduce maintenance costs and ensure a positive return on investment.
- **Balance innovation and risk mitigation.** By following best practices, we lower the risk of introducing technical debt that needs to be addressed later.
- **Improve quality continuously.** We measure and continuously improve in terms of the overall quality accreditations we hold and the quality management system (QMS) we deploy to assure a flawless delivery.

Contact us

Nash
Tech.