



Cloud Based Functional Testing Service Definition

GCloud15

Introduction

We deliver functional testing across IT systems and software applications - web, mobile and desktop.

The Challenges of Functionally Testing Applications

Test Design

Risk based testing underpins everything we do. Understanding how to achieve the right level of risk reduction alongside meeting target budget, resourcing and timeline expectations goes a long way to delivering what is needed to ensure maximum functional correctness and quality for any application.

Automated Testing

By automating testing, you can reduce both time and costs. Eliminate repetitive and time-consuming tests by automating them. However, in order for automation to be effective it needs to be implemented correctly. We often hear “Oh, we tried automating, it didn’t work” and that’s because a lot of thought and effort needs to be put into assessing what is ripe for automation, planning, tooling and scripting.

Automating testing also improves testing accuracy as tests can be run regularly and more frequently. Imagine freeing up your test team to get on with the rest of their job. Imagine using automation to help you shift left by implementing continuous integration to your agile or dev ops method. Automating testing helps it move further up the development pyramid.

Test Early, Test Often

Automating testing ensures the test team can keep up with sprint cycles. Automation increases test coverage, reduces test cycles times and improves time to market thereby reducing both time and cost. Testing Performance provides expert-level knowledge of Automation Testing.

We can assess suitability for automation, define requirements, select and advise with both open source and commercial tools, build a strategy, build the automation framework, script and execution, as well as provide training to any in-house test team.

Not all testing can be automated, but an automated test approach means that repetitive and essential testing can be undertaken in an efficient and cost-effective manner.

Tests into Pipelines

Being able to test on demand multiple tests executed via a pipeline can become that essential link between development and deployment, ensuring a good level of test case coverage, finding problems and issues quickly and getting feedback to the right people at the right time.

Open Source Tooling

By using where ever possible open source tools for functional test automation, costs can be controlled. By targeting the right tools used in the best way dependent on the technology under test and the test resources available, the maximum benefit can be achieved.

Data Migration

Testing in specialist areas such as data migration using bespoke test harness can quickly begin to provide benefit, especially when the data migration has to run multiple times. A test harness to support the differences in application data based on the comparison between legacy and any new replacement application can improve accuracy and allow business analysts and developers to focus on those areas that genuinely need to be resolved.

Key Benefits

- Risk-based testing ensures the right balance between quality, budget, resourcing, and delivery timelines.
- Automation reduces testing time and cost by eliminating repetitive and manual test execution.
- Carefully planned automation increases accuracy, reliability, and long-term return on investment.
- Early and frequent testing enables teams to keep pace with agile sprint and release cycles.
- Increased test coverage improves confidence in functional correctness and overall application quality.
- Pipeline-based test execution provides rapid feedback between development and deployment activities.
- Shift-left testing identifies defects earlier, reducing remediation cost and delivery risk.
- Open-source automation tools minimise licensing costs while maintaining flexibility and technical capability.
- Expert tool selection ensures automation solutions are aligned to technology stack and team skills.
- Data migration testing improves accuracy through repeatable validation of legacy and replacement systems.
- Bespoke test harnesses reduce manual effort and highlight genuine data discrepancies quickly.
- Automation frees test teams to focus on exploratory, risk-based, and high-value testing activities.

Return on Investment

- Reduced testing effort and rework lowers delivery costs while improving overall application quality.
- Early defect detection significantly reduces expensive late-stage fixes and production incidents.
- Automation accelerates release cycles, enabling faster time to market and earlier business value.
- Reusable automated tests provide ongoing value across multiple releases with minimal incremental cost.

- Risk-based testing focuses effort where it matters most, maximising quality improvement per pound spent.
- Pipeline execution reduces manual overhead and shortens feedback loops for development teams.
- Open-source tooling minimises licensing spend while maintaining scalable and effective test coverage.

Approach

The following actions are available when using the functional testing service.

Manual Functional Testing Actions

- Review and analysis of functional requirements, user stories, and acceptance criteria
- Creation and maintenance of manual test cases and test scenarios
- Exploratory testing to identify usability, workflow, and edge-case defects.
- Functional regression testing across releases and changes
- User interface validation for web, mobile, desktop, and cloud applications
- Integration testing across internal and third-party systems
- End-to-end business process validation
- Data validation and rule-based functional checks
- Defect logging, triage, and re-testing of fixes.
- Support for user acceptance testing (UAT), including test execution and defect management.

Automated Functional Testing Actions

- Automation suitability assessment and candidate selection
- Automation strategy and framework design
- Development of automated functional test scripts
- Automated regression test suite creation and maintenance
- Execution of automated tests across browsers, devices, and environments
- Integration of automated tests into CI/CD pipelines
- On-demand and scheduled automated test execution
- Automated test result reporting and analysis
- Maintenance and optimisation of test scripts following application changes
- Knowledge transfer and training for in-house teams on automated testing solutions

Training

Training is available and will take approximately 3 days.

Support

Support is available from 08:00 to 18:00 on working days in England. 1st

Line Support

1st line support will be available between the hours of 09:00 – 17:00 via the following channels;

- Web
- Phone
- email

2nd line support which will operate between the hours of 09:00 – 17:00.

Definition of SLA's (Will provide list)

- 0 – Critical – Within 1 Hour – Call back
- 1 – Severe – Within 2 Hours – Call back
- 2- High – 24 hours
- 3 – Medium – 48 hours
- 4 – Cosmetic – 72 hours