



TEST AUTOMATION TO AUTONOMOUS ASSURANCE:

THE NEXT LEVEL SOFTWARE TESTING

EXECUTIVE SUMMARY

Autonomous Software Testing has taken over manual testing methods because:

- Manual software testing methods are prone to error due to human intervention.
- First-gen test automation has been limited to regression testing.
- Second-gen test automation has failed to optimize test scenarios and test execution.
- Risk-based and predictive analysis, 360-degree view of defects, and optimization of test cases have paved the way for autonomous testing.
- Autonomous testing incorporates AI/ML in software testing.
- Autonomous testing is the next step forward for continuous testing.
- Autonomous Assurance is a one-step solution for end-to-end test coverage, impact analysis, and impact-based testing.

EVOLUTION OF SOFTWARE TESTING

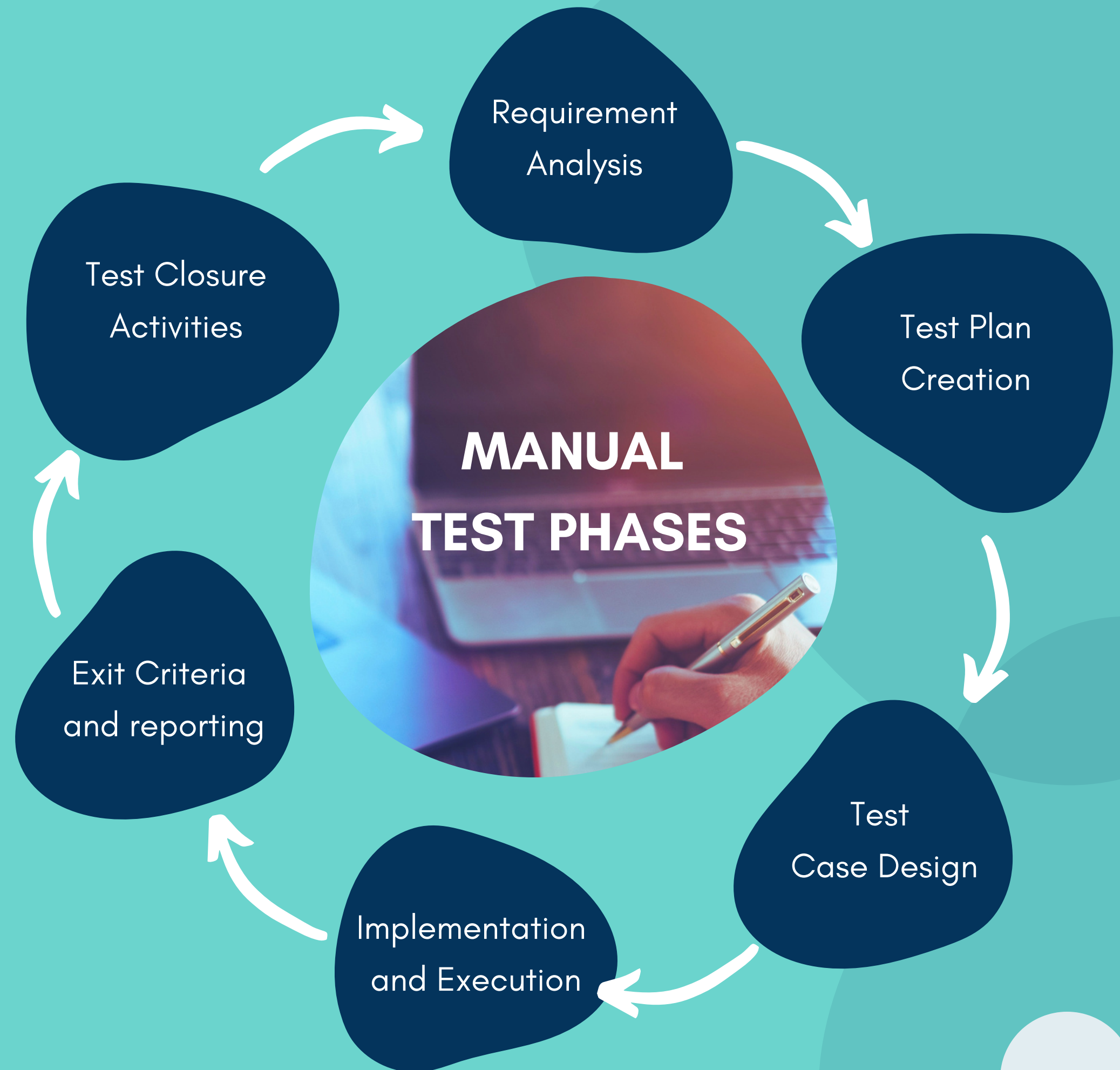
Testing plays a pivotal role in the overall software development process. Since its inception in the 1980s, software testing has seen a tremendous transformation over the decades. In the erstwhile model, software testing played a minor role, where the tester was fundamentally involved in finding bugs in codes. The capturing of bugs was mostly conducted manually.



From manual processes, testing has scaled toward automation. The evolution of the software testing industry is highly aligned to the advancements of newer technologies. These new technologies have helped enterprise Quality Assurance (QA) teams to automate a bulk of repetitive tasks.



Traditional methods of software testing involved manual creation of test cases, execution, and result analysis. This made testing highly dependent on human intervention. It reduced efficiency as the process was prone to human errors and was limited to tacit knowledge and expertise of the tester.

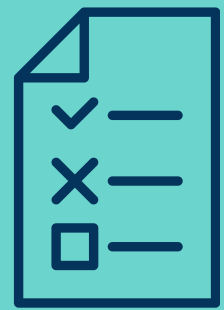


Over the years, the IT industry witnessed a surge in demand for online, customer-oriented software development with the proliferation of Internet services and disruptive technologies. This paved the way for comprehensive software testing methodologies.

Many enterprises saw this as an opportunity to switch to an iterative and incremental approach for testing. This marked the beginning of the test automation era in software testing. Many QA engineers started implementing automation in the phases of testing that were repetitive in nature.

With test automation, Development teams were able to conduct regression testing and sanity checking efficiently. As more and more repetitive tasks were automated, testers got the liberty to carry out other tasks with accuracy within a short period of time.

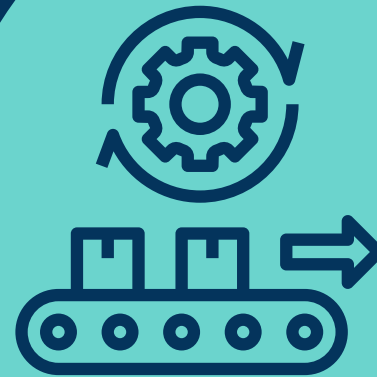
EVOLUTION OF TESTING



Manual
Testing



Bulk
Automation Tools



Robust Automation Tools +
Open Source Frameworks



Intelligent test
Automation

EVOLUTION OF SOFTWARE TESTING FROM MANUAL TO AUTOMATION

1980-1990

WATERFALL METHODOLOGY

Manual Testing

1990-2000

**Experimentation with difference
Development Approaches**

Bulk Automation tools

2000-2010
**AGILE APPROACHES,
FASTER RELEASE CYCLE**

More robust automation tools+
open source frameworks

2011-2018
**DEVOPS, CONTINUOUS
TESTING, CI/CD**

Intelligent test automation
with an integrated approach

CHALLENGES IN TEST AUTOMATION

The software testing space saw a paradigm shift in testing methodologies from manual to test automation with the advent of newer technologies and the need for faster and efficient software development.

Further, it was observed that script-based automation testing required high maintenance to remain at par with dynamic release cycles. Also, in most of enterprises, QA testers were well-versed with test execution automation, which was limited to regression testing and smoke testing. Script-based test automation is time-consuming, which grossly affected the testing of regression suite for each build.



Most enterprises lacked the knowledge and expertise required to achieve end-to-end automation in the planning, design, estimation, and maintenance phases. These are the crucial factors that paved the way for software testing solutions that would help testers augment their capabilities, prevent human errors in production, and optimize the overall test automation/management process.

Even after advancements in test automation, the average rate of adoption of test automation still remains less.



IN A 2018
FORRESTER SURVEY,

27% OF 3228 GLOBAL
DEVELOPERS SURVEYED SAID

THE NUMBER DECREASED
SIGNIFICANTLY FROM



THEIR ORGANIZATION RELEASED
SOFTWARE MONTHLY OR FASTER.



36% IN 2017



As per a *report by the Omdia group in 2020,
the top challenge was integrating test automation
into Development teams.

*Why Autonomous Assurance
is critical for DevOps teams-2021'

WHAT DRIVES THE NEED FOR AUTONOMOUS TESTING?



**INCREASED
DIGITIZATION IN
ALL DOMAINS**



**INCREASED
ADOPTION OF
AI/ML TECHNOLOGY**



**INCREASED
ADOPTION OF
DEVOPS PRACTICES**

01

Digitization certainly comes with benefits, such as increased productivity, but at the same time, it results in complexity of applications. Fintech applications can be the best examples, where banking services have moved to complex app-based applications. The challenge is to provide the best customer experience on these applications and, at the same time, perform well in the market. Software testing can be the game changer and will play a vital role in this fast-paced DevOps world. With traditional test automation approaches, it would be difficult for organizations to maintain the best possible customer experience and at the same time, deliver faster to the market.



02

With advances in utilizing AI/ML capabilities in software development, testing is reaching new heights every day. Software built on AI and ML techniques have revolutionized the business and IT world. Autonomous software testing leverages this technique to reduce human intervention in the development process and enable end-to-end testing and self-healing capabilities of the testing platforms.

Autonomous testing, powered by AI and ML, enables software testing platforms to have a unified 360-degree view of the test application. This testing methodology has helped testing tools develop contextual awareness, autonomously detect defects within the test application, and execute test scenarios with minimal or no human intervention.





Autonomous testing, with DevOps and the Shift Left approach, plays a pivotal role in enabling continuous testing right from the inception of software development. With autonomous testing, enterprises have achieved autonomous generation of test scripts and test scenarios and automated self-healing of automation scripts, change impact, and application defects. This has allowed enterprises to not only reduce manual labor, but also scale up their business by focusing on innovation and adapting to newer business and IT changes, efficiently.

WHAT IS AUTONOMOUS TESTING?

Autonomous testing is the ability to create tests autonomously without human intervention. However, the autonomous testing approach is not restricted to creating tests autonomously. It also brings out autonomous behavior in the complete testing process, that is, minimum human intervention in the test design as well as test execution phase. This includes taking autonomous decisions about which scenario to test under what circumstances, creating right test data to test as well as self-healing when any changes are made in the application under test.

Autonomous testing significantly reduces the task of testing applications by leveraging modern AI/ML techniques. It would not be wrong to say that autonomous testing is one step ahead than test automation.





DELIVER BETTER SOFTWARE FASTER WITH IGNIO AI.ASSURANCE

ignio AI.Assurance is an autonomous assurance product that enables enterprises to deliver better software faster, ensuring stronger brand value. Its distinguished features such as autonomous test suite generation, self-healing of test suites, intelligent test selection, and insightful dashboards help in achieving the perfect cost, quality, and time equilibrium in software development.

It enables Development teams to harness the power of AI/ML and adopt cutting-edge quality engineering practices for timely software releases that help enterprises deliver unique and compelling customer experiences and enhance their competitive differentiation.

ignio AI.Assurance addresses some of the major industry challenges, which include manual creation of test suites, lack of visibility of end-to-end application flow, maintenance of automation scripts, and identification of right tests based on time. It builds test scenarios, automation scripts, and generates application blueprint autonomously. It rectifies application blueprint, test scenarios, and test scripts autonomously based on application changes. With the intelligent test selection feature, ignio AI.Assurance selects test suites for execution based on time and quality parameters to perform fit-for-purpose testing.



IGNIO AI.ASSURANCE FEATURES

ignio AI.Assurance enhances productivity and agility of the software development process by autonomously generating test scenarios and test automation scripts for the software application undergoing testing. It ensures seamless testing and caters to quick testing of builds with a comprehensive test coverage.

AUTONOMOUS TEST SUITE GENERATION

01

- Autonomous generation of application blueprint
- Autonomous generation of test scenarios
- Autonomous generation of automation scripts

IGNIO AI.ASSURANCE FEATURES

ignio reduces human intervention in creating and maintaining test scenarios and automation scripts. It can detect application changes and self-heal the application model, test scenarios, and test automation scripts. It performs impact analysis to identify potential impact of any change on the application. Thus, improving business agility.

SELF-HEALING OF TEST SUITES

02

ignio
AI.Assurance:

- Understands changes in application functionality and corrects application blueprint
- Updates the test scenarios based on application blueprint changes
- Generates modified test scripts for the changed test scenarios

IGNIO AI.ASSURANCE FEATURES

ignio enables execution of test suites based on various test selection strategies – business critical, 100% coverage, change optimized, and custom scenarios. This helps establish a balance between test coverage and testing time, thereby **enhancing the quality of testing**.

INTELLIGENT TEST SELECTION

03

ignio
AI.Assurance offers:

- Change in impact-based testing for rapid regression tests
- Testing of business-critical scenarios
- Comprehensive 100% coverage testing for major releases
- Customized suites for validation of specific functional areas

IGNIO AI.ASSURANCE FEATURES

ignio AI.Assurance provides a single-pane view for key application quality and testing-related metrics and summary dashboards. These interactive dashboards provide valuable, real-time updates that enable businesses to **make intelligent decisions.**

INSIGHTFUL DASHBOARDS

04

ignio AI.Assurance assesses the following questions:

- What is the probability of failure?
- Is the test coverage adequate?
- What is the acceleration in time-to-market?
- What productivity gains did you get?

IGNIO AI.ASSURANCE FEATURES

ignio AI.Assurance helps users input validation checks or assertions by providing relevant information scanning through various sources of documents at the field and screen level. Testers need not remember every validation or acceptance rule from requirement documents or other sources. This also helps in early validation if code and requirement specifications are uploaded, as seeing the difference, the issue can be raised.

SMART DATA VALIDATION ASSIST

05

Validation or assertion of information is displayed as assist at screen and field level using the following:

- Requirement documents
- Code information
- User stories
- Behavior Driven Development documents

IGNIO AI.ASSURANCE FEATURES

ignio AI.Assurance ensures agility in the testing process by autonomously generating test data for every data field. The dynamic test data generated is realistic, versatile, and enables businesses to save a lot of time and reduce dependencies on end customers.

SMART DATA GENERATION

06

ignio AI.Assurance autonomously generates input data for every field on the screen using either of the following:

- Existing data in test environments or
- Excel input or
- Application-specific and random data based on field type

It attaches various data combination to test scenarios and builds necessary test cases for autonomous testing.

BENEFITS ACHIEVED THROUGH IGNIO AI.ASSURANCE

With a distinguished set of features, ignio AI.Assurance aims to achieve perfect equilibrium between cost, quality, and time for every software developed.

- Improves release cycle agility through self-healing tests, thereby, reducing time-to-market by **6X**
- Reduces software failure risks through intelligent constraint-based testing, thereby, reducing cost of quality by **60%**
- Increases throughput and reduces cost through autonomous assurance, thereby, improving efficiency by **70%**

CASE STUDY

LARGEST FASHION RETAILER

From my perspective I have been impressed with the ignio team – their presentations of the product and its benefits have been clear and professional.

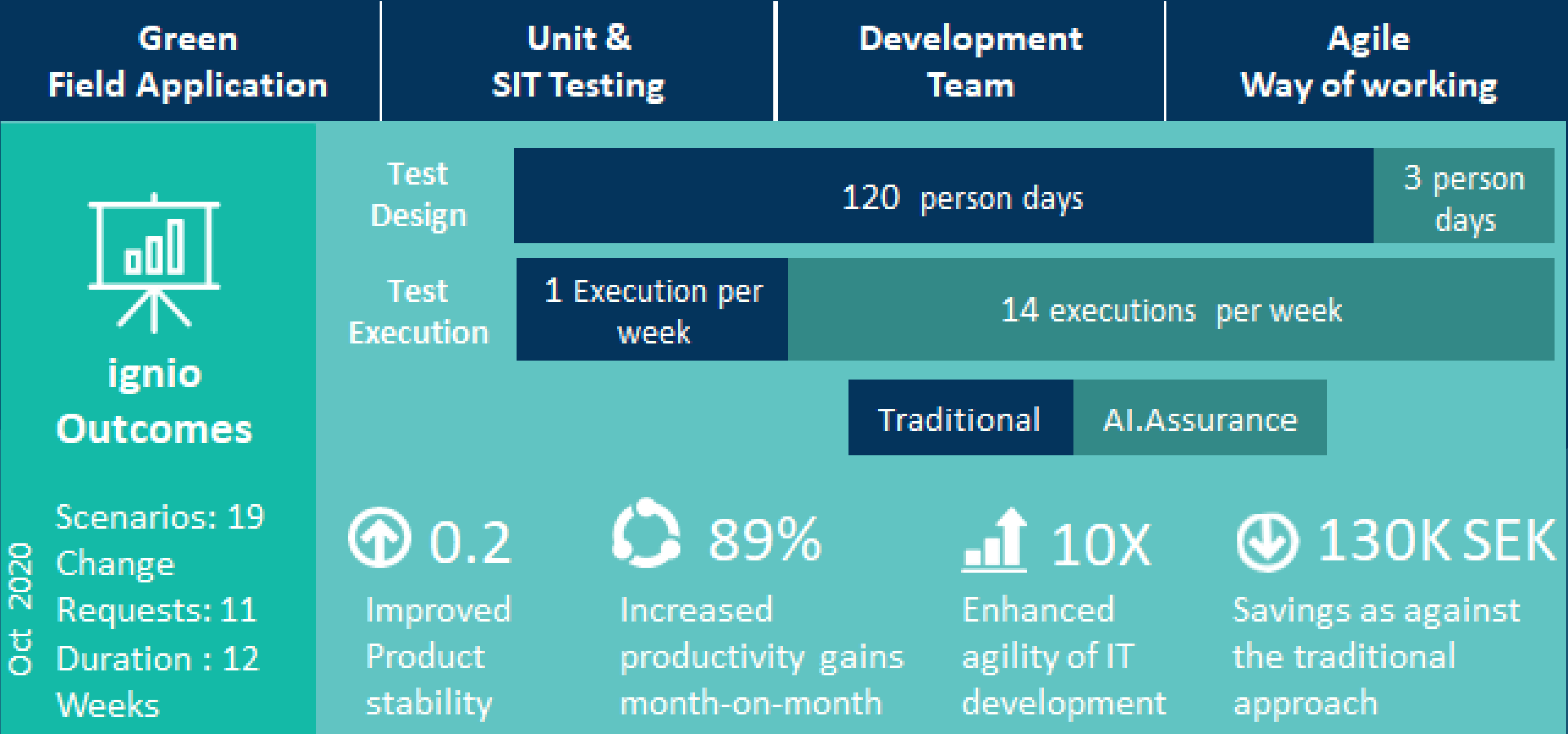
Furthermore the smoothness and simplicity with which the initial tests have been implemented has been very good .. this has not taken much effort from them and so that is a clear plus point.

– By Barrington Jack, Product owner

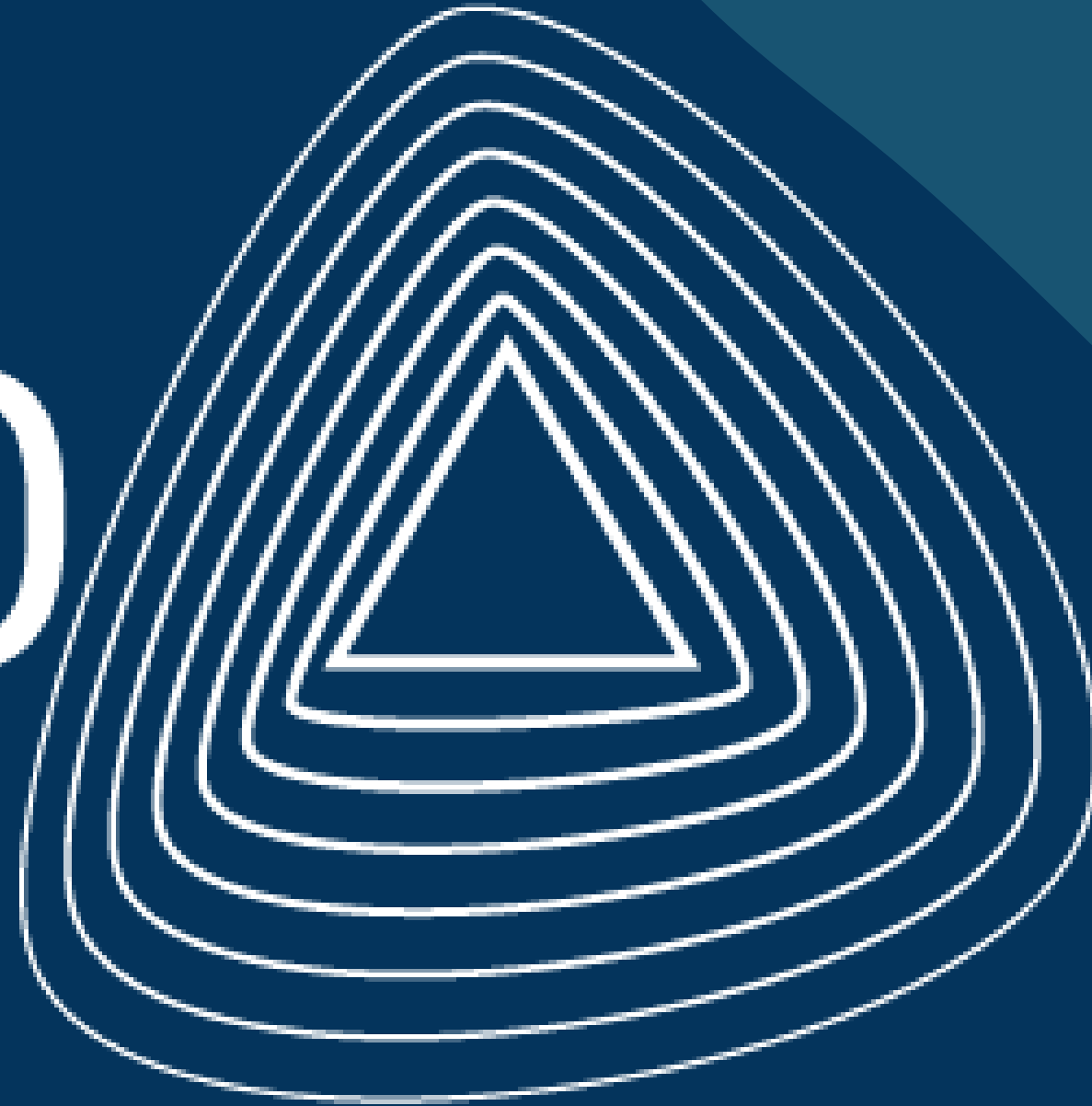


BUSINESS CONTEXT

Consumer behavior and expectations are rapidly changing with the advancements in technology, which influence their shopping habits. The client needed to embrace digital transformation to be relevant in the digital age. Customer had to build a product to restrict sale of specific type of goods in specific countries, which would comply with rules and regulations of that country.



ignio
AI.Assurance



For a demo, visit us at <https://digitate.com/ignio-ai-assurance/>