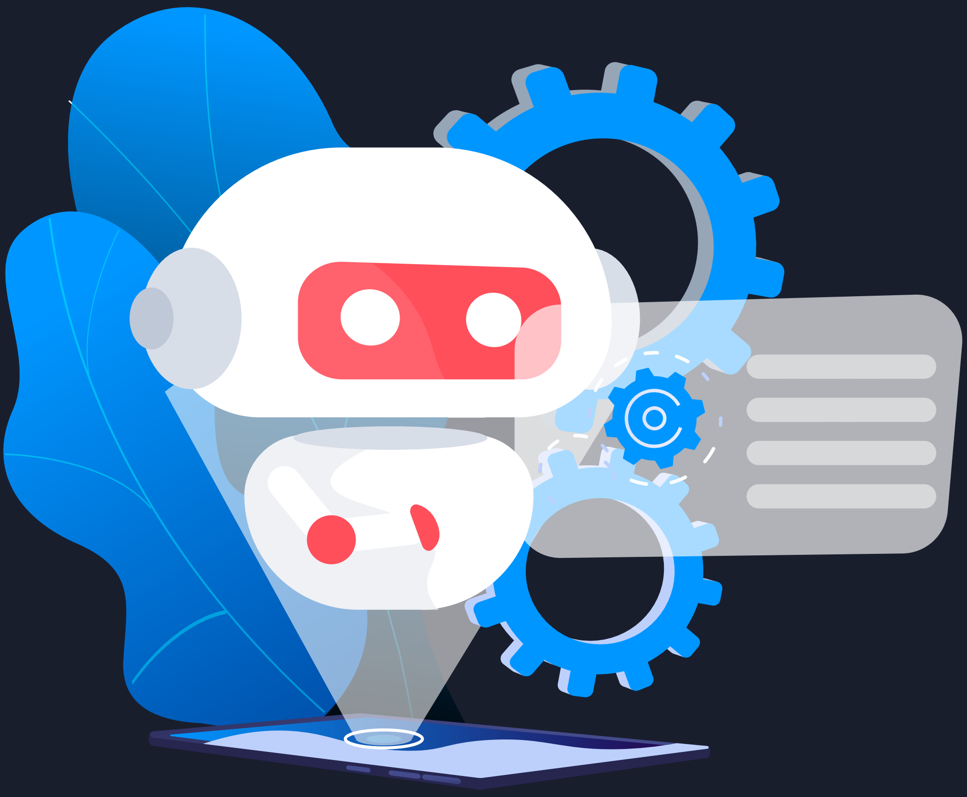
QUALIZEAL

How is AI changing the Dynamics of Software Testing?

Preface



Make testing human-centric while meeting the core business requirements

Advancements in artificial intelligence and the availability of quality datasets have created an opportunity to transform the old and redundant testing practices and overcome the challenges of speed, cost, and quality. It has also helped the software development stakeholders to create innovative products in less time and effort. Integrating AI in software testing enhances the quality of the end product while maintaining company protocols. The software industry has much to benefit from leveraging artificial intelligence, and this is the right time to start implementing it in the testing process.

86% of respondents believe that AI is a "must-have" when selecting new testing solutions, products, or tools ~ (World Quality Report, 2020-2021)

Rapid digitization and innovation are at the top of every CIO's priority. Higher productivity, superior and scalable digital products, and cost-effective development are in demand from all stakeholders. Companies are implementing new technologies to improve these aspects and do more, but not everyone has the budget or schedule to revamp their decades-old testing culture. Constraints like these don't have many positive outcomes. We need something that can be adapted to every company's IT infrastructure and customized based on the tester's requirements. So how do we achieve innovation at a faster pace?

Artificial intelligence (AI) has become an integral part of software testing. Organizations are adopting it in their testing practices and testing its potential to gain more resilience, agility, speed, and business growth. Artificial intelligence is the key to enabling autonomous and zero-touch quality engineering.

AI doesn't only improve the results of software testing, and it does way more than that. Some of the key benefits that you will find with the implementation of artificial intelligence are:

- Eliminating redundant tasks with the help of automation
- Seamless maintenance of continuously changing applications
- Self-healing upon detecting foreign or unwanted activities
- Reducing testing scope through smart impact analysis

Today's software testing challenges

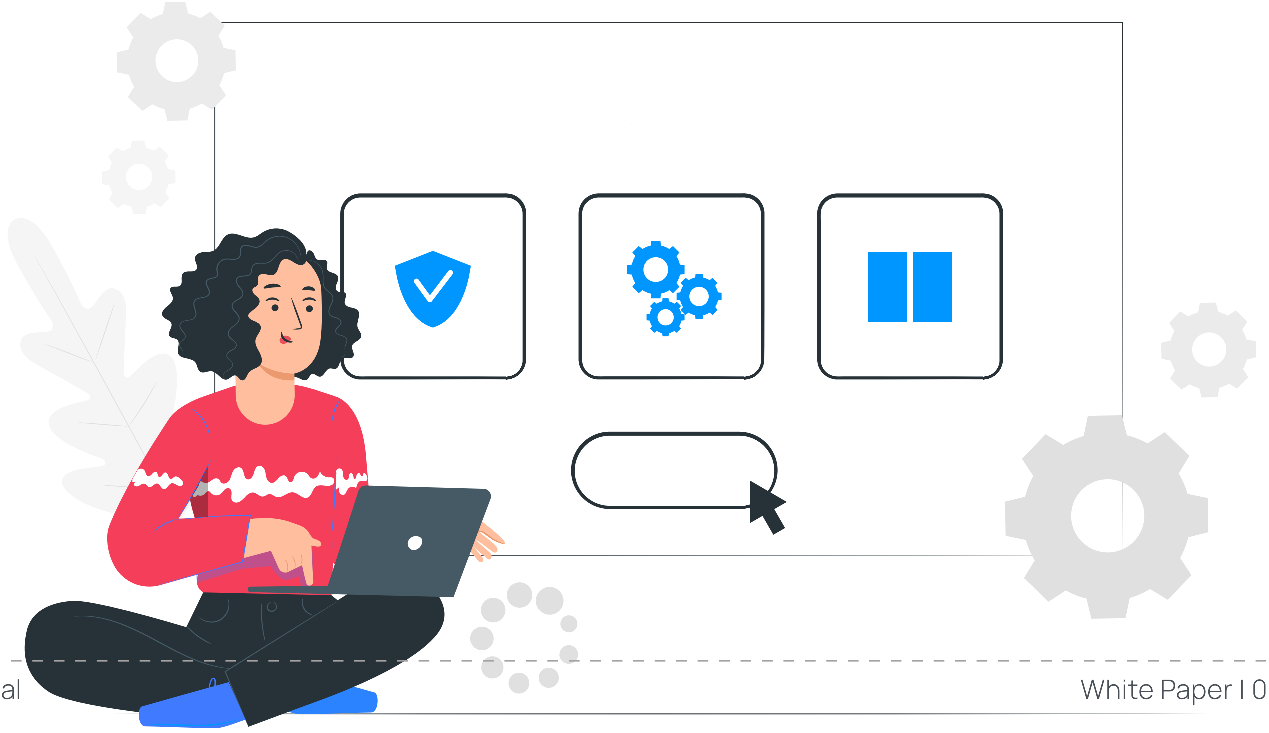
It is a well-known fact that organizations have always considered testing a major bottleneck in the software development process. A significant reason for these issues can be blamed on the waterfall testing method. The last two decades have seen continuous testing evolving and being adapted more frequently, but manual testing doesn't catch all the issues. It is also expensive, making it difficult for organizations to use it often.

In a survey conducted by Forrester, 77% of respondents said that AI-infused testing has resulted in smarter automation.

The role of AI is way more than just reducing the risk of failure or meeting the KPIs. It improves the morale of testing and development teams, allowing a better work-life balance and making the workplace more desirable among job seekers.

Compliance and regulations play another crucial role in the popularity of AI. AI can learn and detect compliance issues during the initial state of software testing. Since this is continuous testing, it doesn't need regular manual tests and revisions to make the process faster. It saves time and money, making the end product more accurate and reliable.

As per Omdia Report 2021, most organizations will adopt autonomous testing in 2022, and almost 90% of all surveyed companies said they will have fully deployed autonomous testing by 2024.



AI-Enabled Software Testing – Industrywide Use Cases

Banking and Finance

BFSI sector can leverage AI to determine the functionality, usability, and performance of digital wallets and end-to-end testing of mobile apps.

Banking implementing or upgrading their banking systems can use AI to conduct integrated testing of complex touchpoints.

Retail

AI simplifies and speed-ups retail app releases. With AI-integrated software testing, the applications can release new features, roll out updates, and dodge security incidents like DDoS attacks. Retail brands leverage AI to test partnerships across all digital and backend applications for functional testing.

Healthcare

The healthcare sector significantly benefits from AI-enabled software testing. It elevates the level of privacy and secures confidential data. Through AI, software testers can continuously monitor various machines and detect malfunctions in advance. Healthcare entities also prefer AI to make sure that their medical devices adhere to guidelines by OWASP & SANS Top 25.

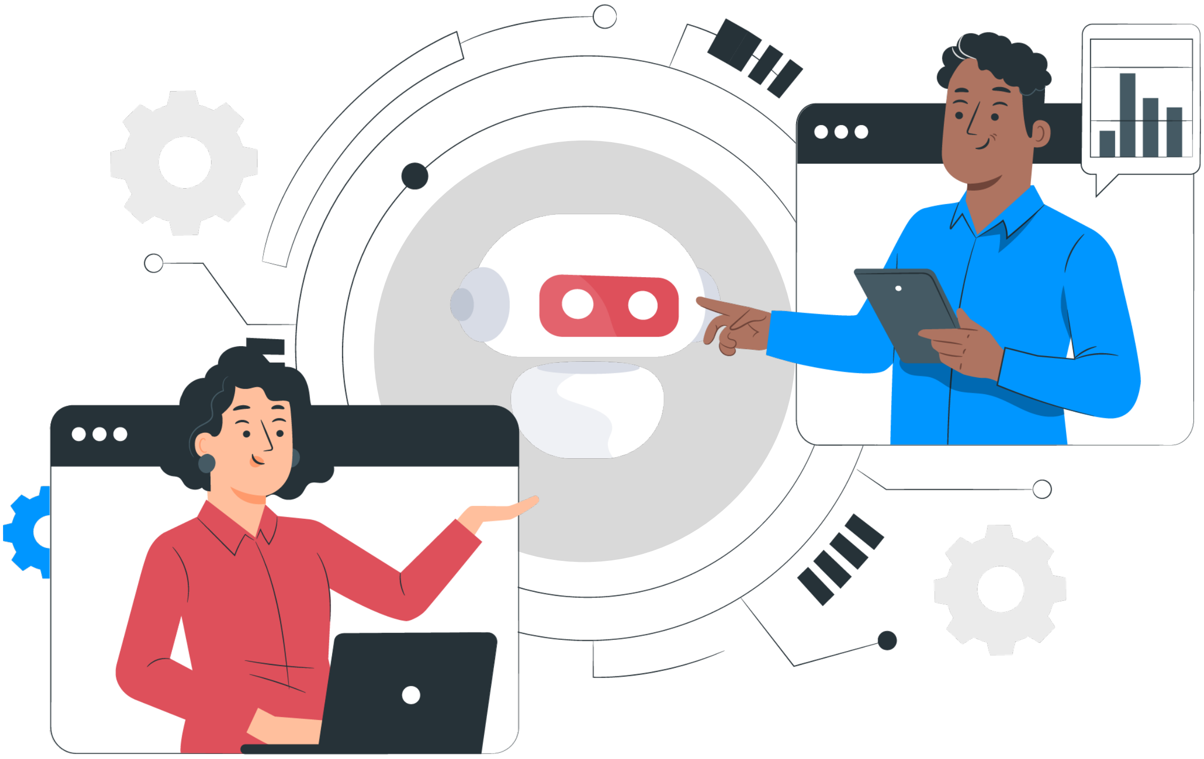
Aviation

AI can reduce the complexity of aviation software by creating a simplified framework. It can quickly improve the efficiency of in-flight connectivity and identify configuration issues. AI-enabled automation tools can also design tests to check these issues.

Travel

AI-enabled software testing makes ticketing, booking, and itinerary management systems more straightforward and user-friendly. These tests also make the application highly resistant to cyberattacks and protect confidential customer data.

Benefits of leveraging AI in software testing



Improved defect detection (48%), reduction in test maintenance costs (42%), and improved test coverage (41%) are some of the top benefits that AI/ML brings to software testing ~ Gartner

AI algorithms are capable of mimicking human intelligence. If we add ML to the mix, it removes human interference and automatically learns various tasks. This significant advantage enables AI and ML to develop algorithms to access and learn from that data. These features prove beneficial when used in software testing.

Here are some significant benefits of AI in Software Testing:

Seamless shift left testing

AI chooses the right tools based on the algorithms making the results highly accurate.

Precognitive monitoring

ML algorithms flag coding errors and eliminate issues like application overload and server-resource consumption.

Benefits of leveraging AI in software testing

Precognitive monitoring

ML algorithms flag coding errors and eliminate issues like application overload and server-resource consumption.

Visual validation

Visual validation ensures that visual elements are engaging, accessible, and functioning as they are supposed to.

Test suite and test script maintenance

AI tools maintain and extract appropriate test scripts and categorize different test suites according to their usage and needs.

Better test coverage

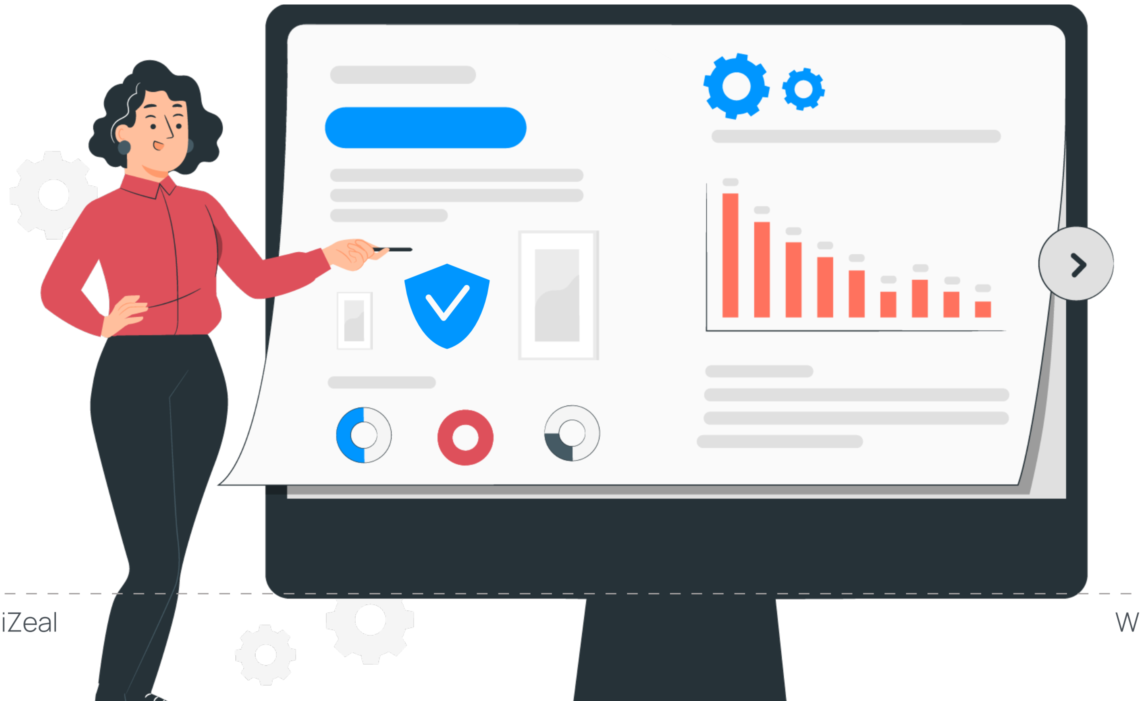
AI helps ensure that the program works as expected by covering a more significant portion of the application design.

Cost optimization

AI can help quality engineers to shift from proprietary to open-source tools, reducing the expenses of manual testing.

Advanced security tests

Testers use AI to close security gaps and vulnerability issues, making the application secure for users.



4 Key AI-driven testing approaches

Differential testing

Differences between application versions over each build are compared. It is ideal for finding semantic or logic bugs that do not result in crashes or assertion failures. Additionally, differential testing makes automated test generators more valuable because it abstracts away the interpretation and management high test volumes by focusing on the changes between test suites.

Visual testing

The AI algorithm tests the look and feels of the application by leveraging image-based learning and screen comparisons. In non-AI-based visual testing, the result depends on the developer's and the team's opinion and has no guarantee of meeting the market demands. AI tools learn about user requirements and ensure that the application they are building appears to the end-user as it was initially intended.

Declarative testing

The AI decides how to perform a test and specifies the intent in a natural or domain-specific language. The approach makes the test robust, making it hard to introduce bugs and easy to identify existing ones. AI makes declarative tests easy to optimize as an algorithm controls implementation.

Self-healing automation

The elements in the framework are automatically adjusted when the UI is changed or edited. In non-AI self-healing automation, the testers have to check and report the changes made while AI updates and shares reports without human involvement. In Non-AI self-healing automation, someone has to manually update the script or object repository and rerun the script, which is not the case with AI-enabled self-healing automation.

Impact of Machine Learning on Software Testing

Artificial Intelligence (AI) and Machine Learning (ML) transform software testing by removing manual testing from the process. With AI/ML, intelligent machines can achieve continuous delivery with high quality and speed.

This makes the impact of AI/ML very high in certain functions while it does very little for others. Here is an analysis of how Machine Learning can benefit various parameters from the software testing lifecycle

Software Test Lifecycle	ML Impact	Reason(s)
Requirement Analysis	Medium/ High	Requirement analysis is performed to understand the expectations and problems an application intends to solve. It has traditionally relied on human involvement. But many organizations are leveraging AI and ML to broaden this scope and identify user stories and requirements problems. Businesses can find more precise feedback and send better user stories and requirements to the development team.
Test Case Development	Medium/ High	ML only works when similar test cases are already present in test case development. Even during such scenarios, a human approach is needed for customization. MLs role is growing here as some tools are capable of creating test cases and even customizing it.

Setup and Configuration	Medium/ High	Most organizations prefer manual processes for setup and configuration because of the varying requirements. But with the introduction of tools that leverage ML and AI for these tasks, testers can seamlessly. Minor changes to the setup don't cause the test suite to fail, as ML tools can easily modify the configuration based on new requirements.
Functional Test Automation	Medium	Machine learning reduces false positives through algorithm advancements, but human involvement is still necessary for customized requirements.
Non-Functional Test Automation	High	The human approach does very little for non-functional tests. Machine learning makes analyzing aspects like user load, experience, and security more accessible.
Failure Analysis & Debugging	High	ML proves highly effective in failure analysis and debugging, reducing the timelines and achieving highly accurate results.
Audit and Reporting	High	Visual Quality (image recognition) and performance (classification) thresholds can be defined and analyzed in real-time.

AI is the Future – Enhance Your Software Testing Capabilities with Automated Solutions

The proliferation of AI and ML has fundamentally transformed software testing in profound ways. We will observe the positive impacts of upcoming technologies like internet-of-things, self-driving cars, and even robotics. It is the right time to start consulting with independent quality engineering partners to gain the best of AI and shift left testing for your benefit.

QualiZeal has integrated AI-based shift left testing for various quality engineering implementation projects. We have helped our clients leverage quality engineering to market faster by incorporating algorithmic frameworks into core business practices. We have also helped our clients effectively generate and optimize test cases, prioritize testing and automation, enhance UI testing, and reduce delivery timelines.

With QualiZeal as your quality engineering implementation partner, you'll start building test assets faster and easier with more efficiency and accuracy.