Webomates

Webomates - Intelligent Test Automation Platform

How One Of The Biggest Telecom Companies Is Lowering Software Testing Costs with AI Test Automation

Webomates' AI Test Automation for Telecom - Whitepaper 2022

Webomates

Abstract - Telecom company and Webomates

An insight into applying intelligent test methodology to optimize operation and engineering cost for just in time software deployment and how advancements in Artificial Intelligence can now save 50% or more in software testing costs for just in time software deployment.

Working together on three production pilots the telecom company and Webomates have demonstrated software testing cost savings up to 69% [1] when compared to industry standards with the advent of AI Powered, Intelligent software testing automation.

Additionally, software testing time has been reduced by 80% [2] plus with fewer software bugs finding their way into production.

This month Webomates introduced to the said company another industry first to address code coverage with a target to reach 80% [2] at 50% the cost.

Save > 50% in software testing

Save more with AI

Testing time reduced by 80%

80% code coverage

Webomates

TABLE OF CONTENTS

Webomates

Introduction

The demand on software development teams to release software faster, at the highest level of quality has never been greater. The cost to scale today's industry standard automation systems simply cannot keep up with the need to test faster with better quality in a more cost-effective way.
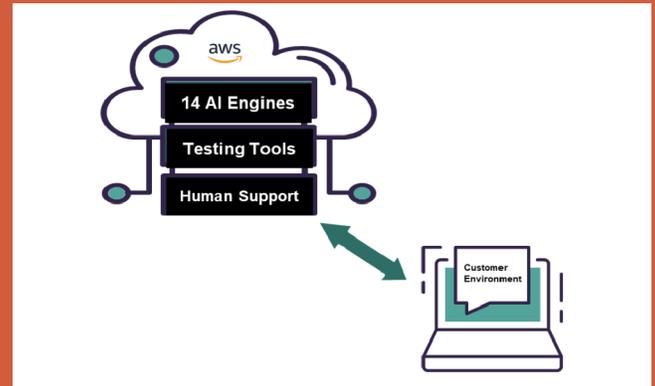

Figure 1: Test Environment

Two teams from the large telecom company are addressing and solving this issue with Webomates AI Powered Intelligent Test Automation.

Webomates is a state-of-the-art patented cloud-based AI software Testing as a Service platform that provides functional software testing for User Interfaces (including native), mobile applications, APIs, and set-top/gateways.

The platform creates test cases and scripts and executes them using multiple testing execution techniques like AI automation, and AI healing. Performance is guaranteed by way of twenty-four hour (full regression testing) and (eight-hour overnight testing) of software modules with guaranteed performance SLAs. At the end of the test execution, actionable triaged defects are logged and recorded for customer's software QA team to review and approve.

Webomates

Webomates Test Strategy

Below is a high level table highlighting the simplicity and time/cost savings with Webomates TaaS when compared to legacy automated software test systems.

How AI-Powered Intelligent Test Automation Saved Large Telecom Company's Time & Money At Every Step During Software Testing

	Action	Traditional Industry Standard Automation	Webomates Intelligent Test Automation
Create	Webomates TaaS reduces labor time/cost to generate test cases and test scripts from several months to weeks	Months	Weeks
	Software SME time and associated expense drops from months to hours	High person months	Hours of SME time
Execute	Modified test cases are automatically updated during execution reducing the time and labor cost to develop new test cases for new software features.	No	Yes with AI Healing
	Test Cloud infrastructure is included in Webomates TaaS	Purchase	Included
	Execution time with AI healing enabled to reduce software testing time and expense	Weeks	Hours
Analyze	Pass/Fail report has False Positives. False positives are one of the biggest contributors to failed software tests taking up valuable SME time to correct them.	Yes	No
	The time and cost to review software defects is shortened as Webomates provides a 20 second video of every software bug recorded	No Pass/Fail report	Defects with 20 second video

Table 1: Time and Cost Savings

Webomates creates a test strategy, test cases and test scripts. And the customer's SME reviews and approves. Execution of the test cases and test scripts occurs during a twenty-four hour (full regression) or eight-hour (module testing) overnight testing cycle with guaranteed performance SLAs. All that is required is for a customer's SME to review and approve the test cases and test scripts.

Webomates

When new features are introduced into the application Webomates' AI Healing detects and updates test cases and test scripts within the 8 or 24-hour SLA with no false positives.

A twenty second video is included for every defect to further reduce time and labor expense.
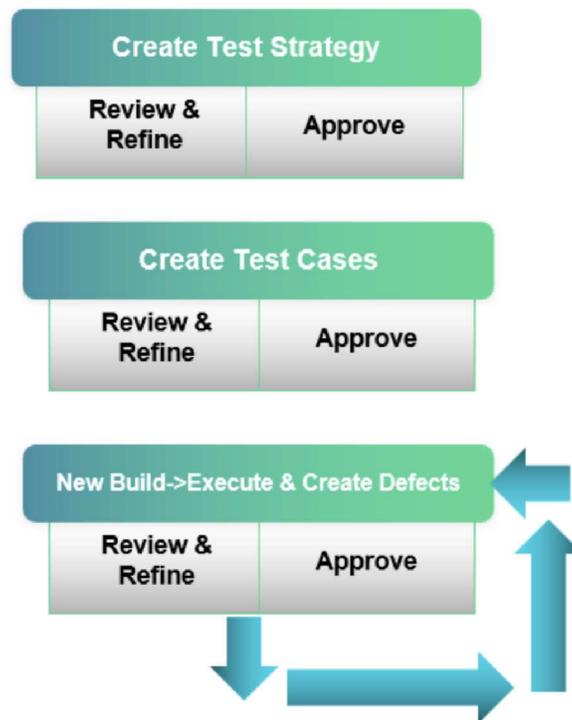

Figure 2: Test Process Flow

Webomates

Large Telecom Company and Webomates Success Stories

The first production pilot began in 2021 with large telecom company's application one team. Please refer to the table below highlighting some of the cost and time savings realized with AI Powered, Intelligent Test Automation:

Customer Results

Quantitative	Industry Standard	Webomates CQ	Value	
Test creation	100% / Test	31% / Test	69% savings	Direct Cost Saving
Test Case Execution	100% / Test	31% / Test	68% savings	
Test Automation Infrastructure	$2000-$3000 / month	None	100% savings	
Test Creation Duration	40 days	3 Days	92% Improved Velocity	Velocity Increase
Execution Duration	5 days (120 Hrs)	1 day (24 Hrs)	80% Improved Velocity	

Table 2: Application Tests and Savings

The telecom company's application one team achieved a 69% cost reduction and an increase in software testing velocity increase by over 80% when compared to today's industry standard automation systems.

Webomates

Large Telecom Company's Internal Application 2

Earlier this year this team began a twelve-month production pilot and the results have been outstanding.

Here's one example on how the they are reducing production defects by leveraging AI in their software testing.
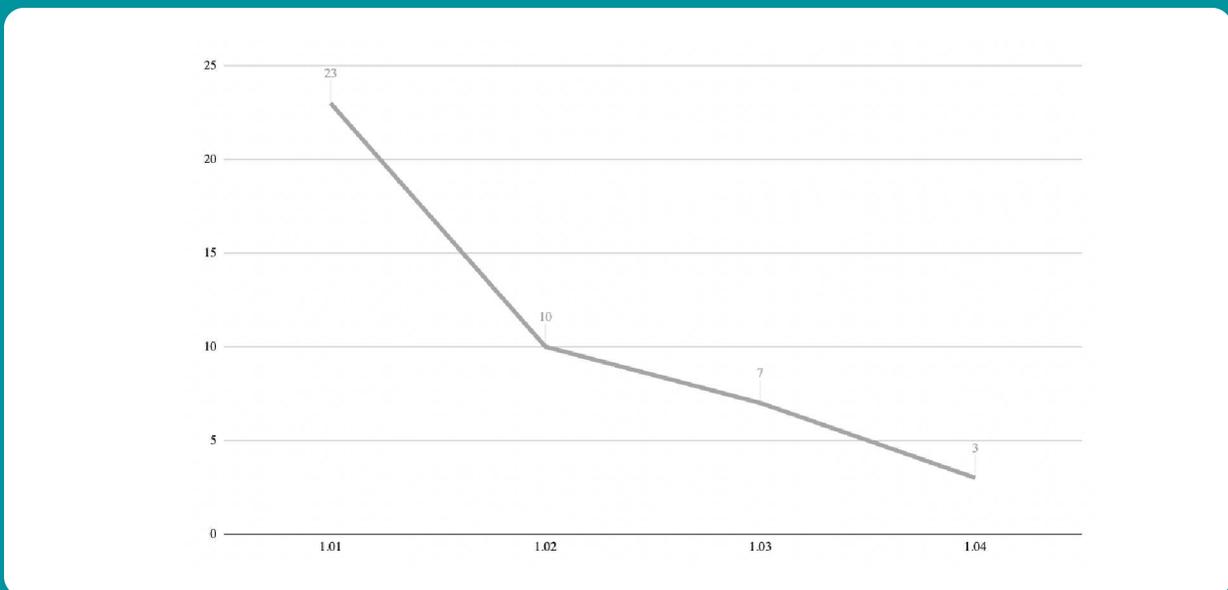

Figure 3: Defect leakage for application 2

This graph shows the defect leakage numbers since Webomates CQ has been applied. Prior to the introduction of Webomates CQ UAT (User Acceptance Testing) had discovered 25 defects. In the month after the Webomates CQ was deployed this number dropped to 7 defects. In the latest monthly bundle this has dropped to 4 defects a truly dramatic reduction in defects.
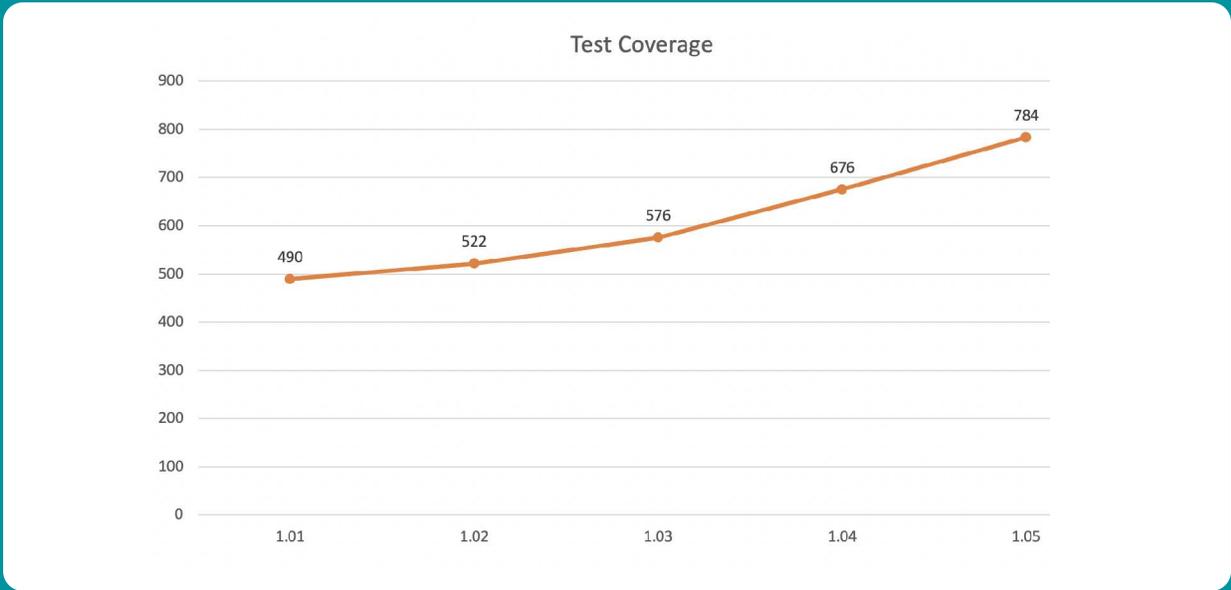
Webomates

Figure 4: Product test coverage

The above chart shows a 69% increase in the number of tests from 371 to 627 based on feature addition and targeted expansion.
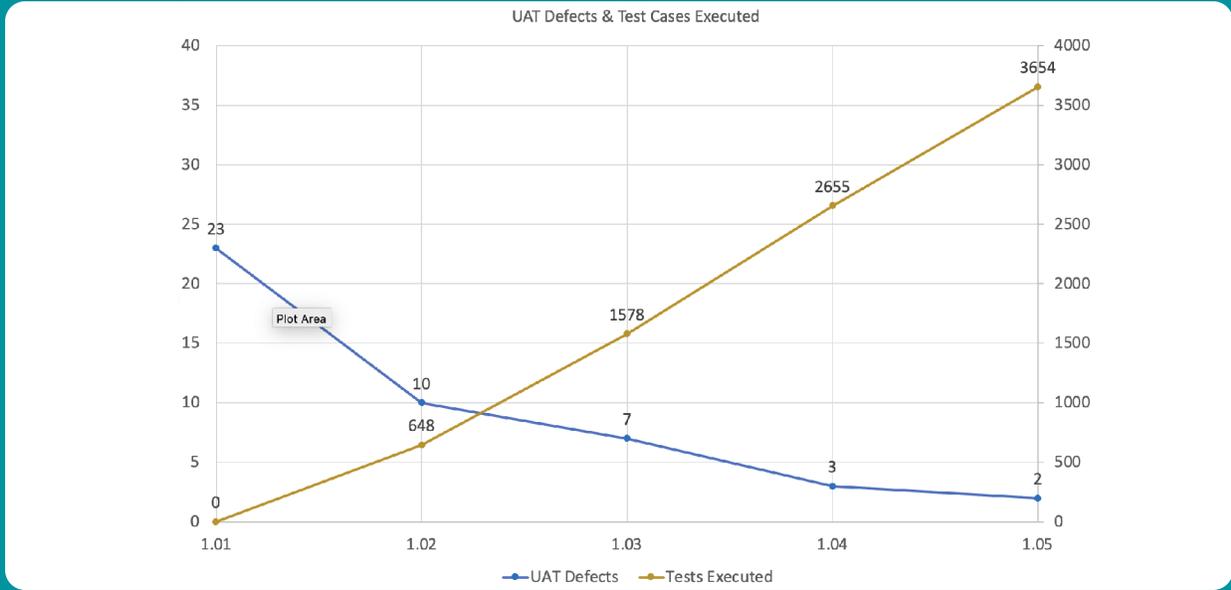

Figure 5: Test Executions

Over the same two-month period there was a dramatic increase in the frequency of usage of Webomates CQ measured by the Test Execution count. This is the number of times that a test case is executed per month.
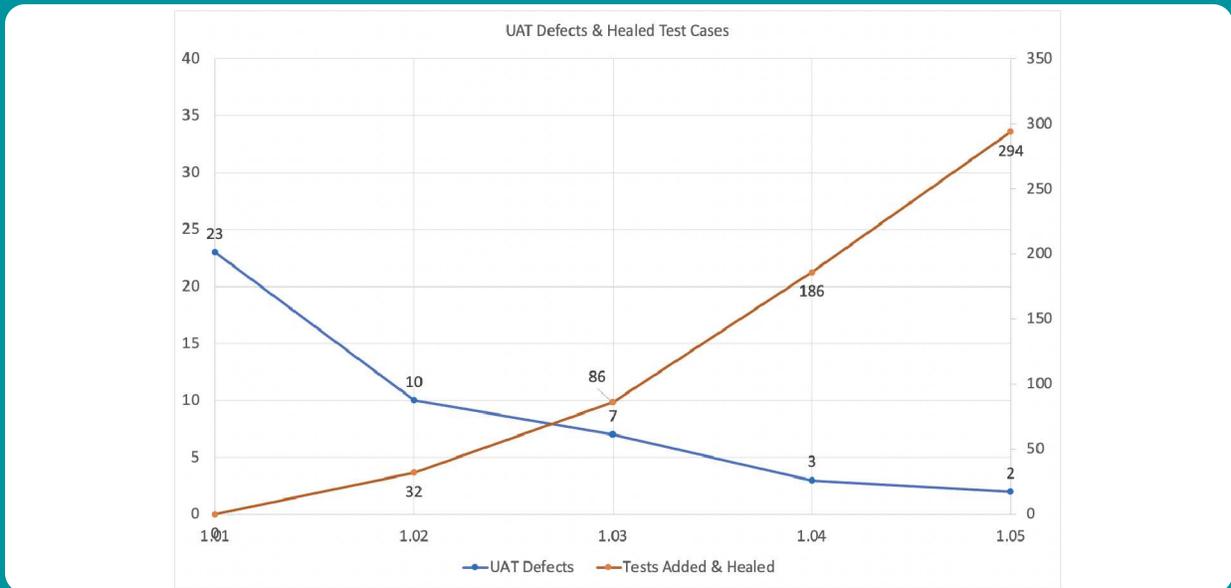
Webomates

Figure 6: Test Case Healing

The final metric that is shown above is the number of existing test cases and their associated automation scripts that were AI healed by Webomates CQ. AI healing is "fixing" existing test cases and test scripts for :
      (a) Feature modification and addition
      (b) Automation Script Fix Due to change in
         (i) Locators
         (ii) Timeout
         (iii) Test Data change

The large telecom company's implemented Software Reliability Engineering (SRE) in its network based on the work by their internal team. Furthermore, an unnamed executive and their team at this company (currently a technical adviser at Webomates) continued the work of implementing Software Readiness Estimation [5] from the statistical methodologies and defect prediction from SRE to all wireline and wireless network software.

A Web based platform was implemented to report significant defects thus improving the production roll-out of reliable software while reducing the cost impact to the large telecom company's operations (truck rolls, etc. and lost customer revenues). Webomates builds on this work offering greater capabilities regarding defect detection and correction by way of its patented AI software.

The graphs above show test coverage for this application, test cases added and automatically healed along with a reduction in UAT defects. The Webomates CQ platform is finding more software bugs when compared to industry standard offerings that builds on the groundbreaking work carried out by the customer's Software Reliability Engineering taking the next step towards bug free software deployment.

The ability to find software bugs earlier in the development process is also helping their development teams shift left by moving testing, quality, and performance testing earlier in the software development process.

Webomates

Large Telecom Company's Digital Transformation – Migration to Another Cloud Environment

A follow-on project for the application 1 team expanded the scope of the engagement. Rather than a single application the team subscribed to a "bucket" of test cases and test case execution executions.

The power of this model was that application 1 could pick any application, onboard in a week and start testing. The flexibility of the bucket concept is illustrated below. Here is the original "bucket" purchased by this company.

Business Scenario		Tests	Executions 10 months
Bucket Model	New Project with the Length of project around 12 weeks. Apply test cases as per project rather predefined	2200	6600

Table 3: Bucket Scenario

Webomates

However the key power in the bucket concept is that this is not a fixed plan. The team for application 1 has the flexibility to apply whatever number of test cases and test case executions to whatever project they need as long as it does not exceed the maximum test cases and test cases executions in the contract period.

This is very similar to and modelled after the cloud capacity models that AWS (Amazon Web Services) pioneered and now offered by every cloud provider. In these Cloud SaaS services what you plan to consume is what is purchased. How you decide to consume it is entirely up to the customer.

This team is currently in the process of migrating almost a hundred applications to another cloud environment.

Webomates

To generate these 670 test cases and test scripts would typically take months. It is now being carried out in weeks with the applications being validated on Azure cloud post migration enabling a far quicker and more predictable migration. Webomates rated capacity is 2,000 test cases and automation scripts in 4 weeks. This would typically take a test automation team 9 to 12 months

Webmates

New Webomates Initiative – Meet Code Coverage Targets at Half the Cost

A new initiative between the large telecom company and Webomates was to address code coverage. Code coverage is a metric that can help better understand how much of their source is tested. Our target is to reach 80% code coverage using functional tests.

Currently they utilizes only automated unit testing in order to achieve code coverage targets. Here the employees or contracted developers write unit tests that result in a code coverage report. This has three major disadvantages:

1.   Developers writing unit tests are not writing new features
2.   Developers are spending large amounts of time using unit tests to perform unit testing, integration testing and system testing
3.   In order meet code coverage targets, there is an exponential increase in effort for higher levels of code coverage numbers

Initial results are showing a 50% decrease in developer efforts in unit testing when combined with Webomates AI Automation. Webomates is working on an advanced service where this number can be advanced to 80% line code coverage which is customer's corporate target for all software.

Webomates

The figures below show the two different areas that code coverage is provided by Webomates CQ :

Front end testing – this portion of a typical software web application is implemented in Javascript. The code coverage is achieved by using Webomates CQ UI testing in conjunction with SonarQube
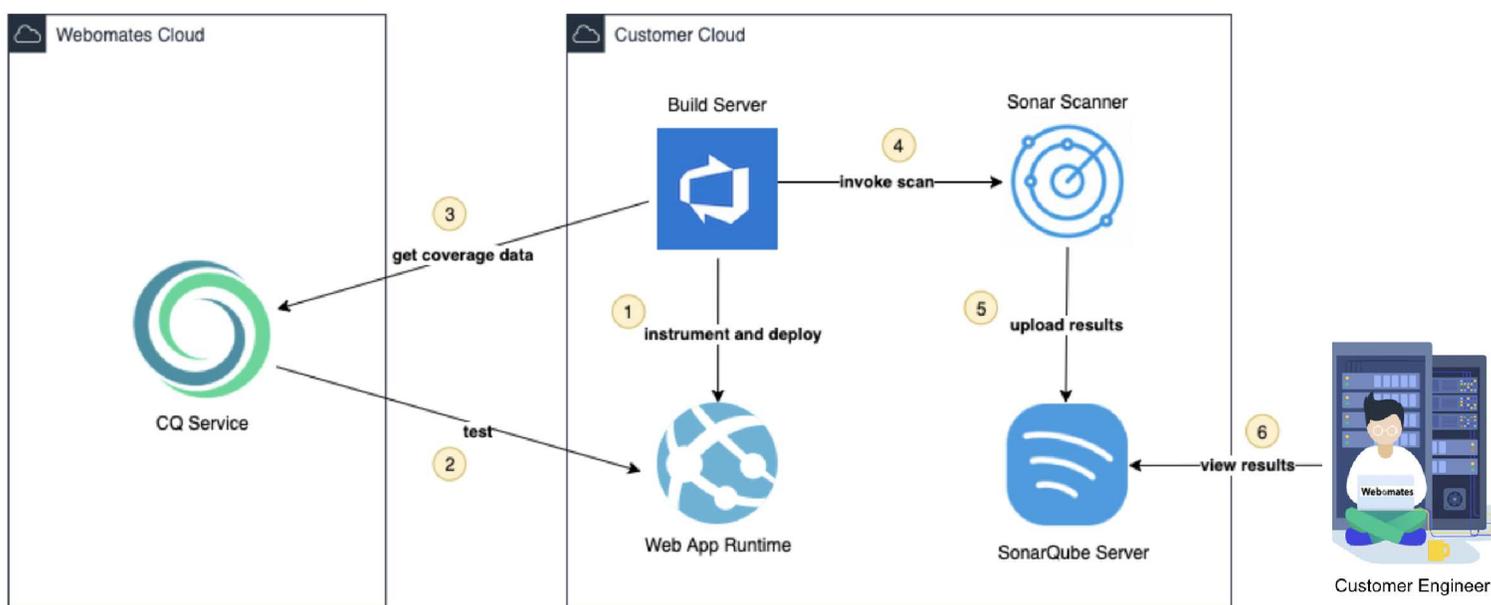

Figure 7: Webomates SonarQube Architecture

Backend (API) testing – the backend micro-services are typically implemented in Java. Another possibility is C# (a.Net environment). In either case Webomates CQ API testing + Webomates CQ UI testing are both used to generate code coverage reports for the back end services and fed into SonarQube.
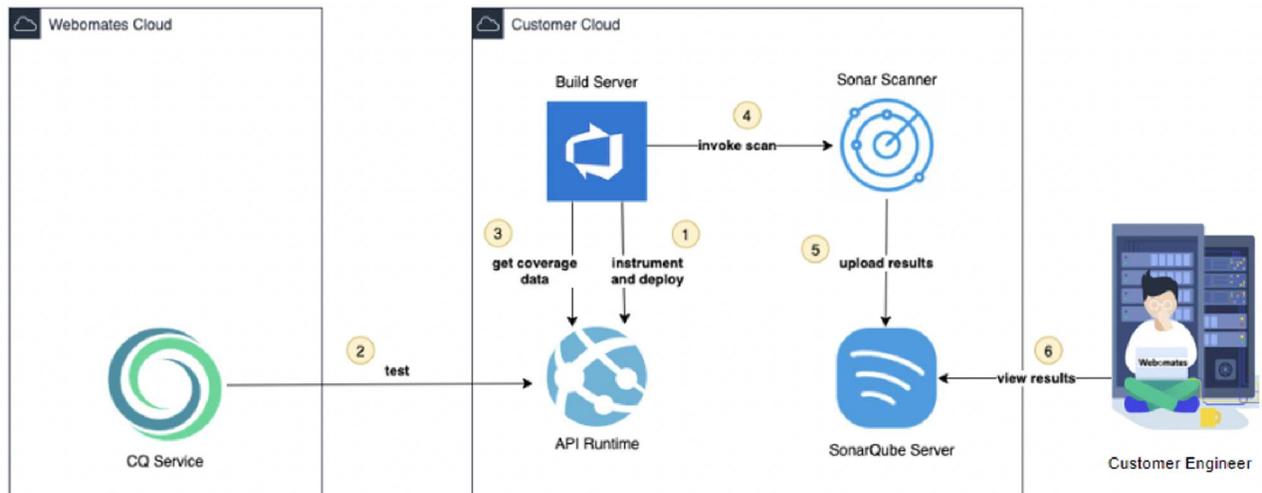
Webomates

Figure 8: Webomates SonarQube Architecture

Webomates offers code coverages as an optional add on to its AI automation. Get code coverage with no effort with Webomates

Webomates-Web Based Software Analysis Tools

Webomates Intelligent Test Automation systems is powered by 14 AI Engines residing in the cloud. Webomates has 6 patents granted with more underway. The AI technology accelerates test case/script creation, execution time, decreases software test execution time, eliminate reporting of False positives (a major limitation in legacy automation systems) and automatically updates test cases as new features are introduced to the software under test.
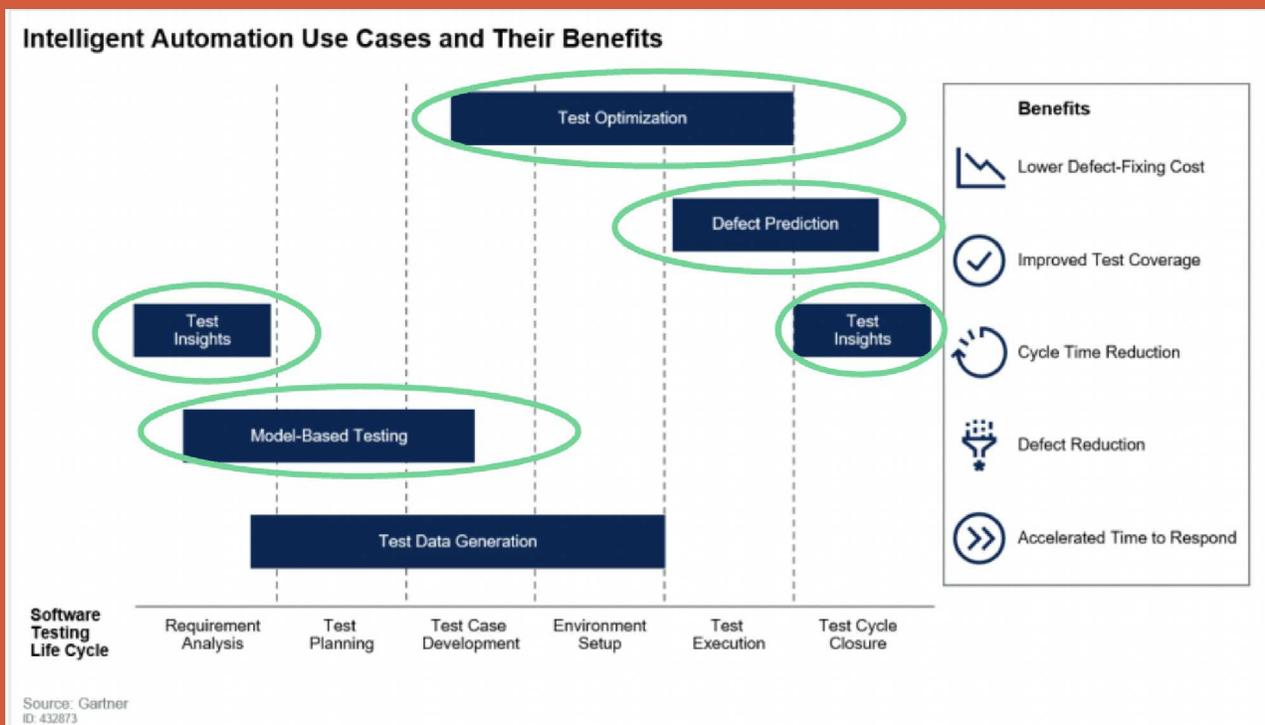

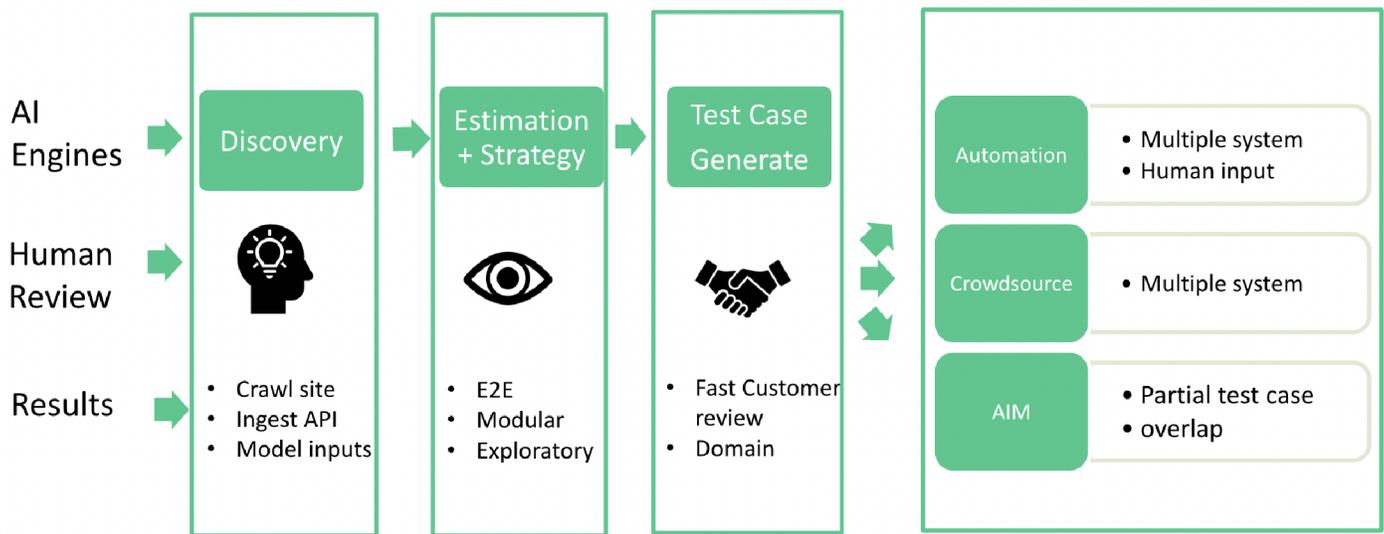Figure 9: Intelligent Test Automation use cases

Figure 10: Reduce Setup time from months to weeks

Figure 10 shows above the patented Webomates CQ Setup process uses a multi stage AI engine to generate test strategy, test cases and test scripts for both UI (User Interface) Testing and API (Application Programming Interface) Test. These are reviewed and refined by the Webomates Team and presented to the customer for ultimate approval at each stage.
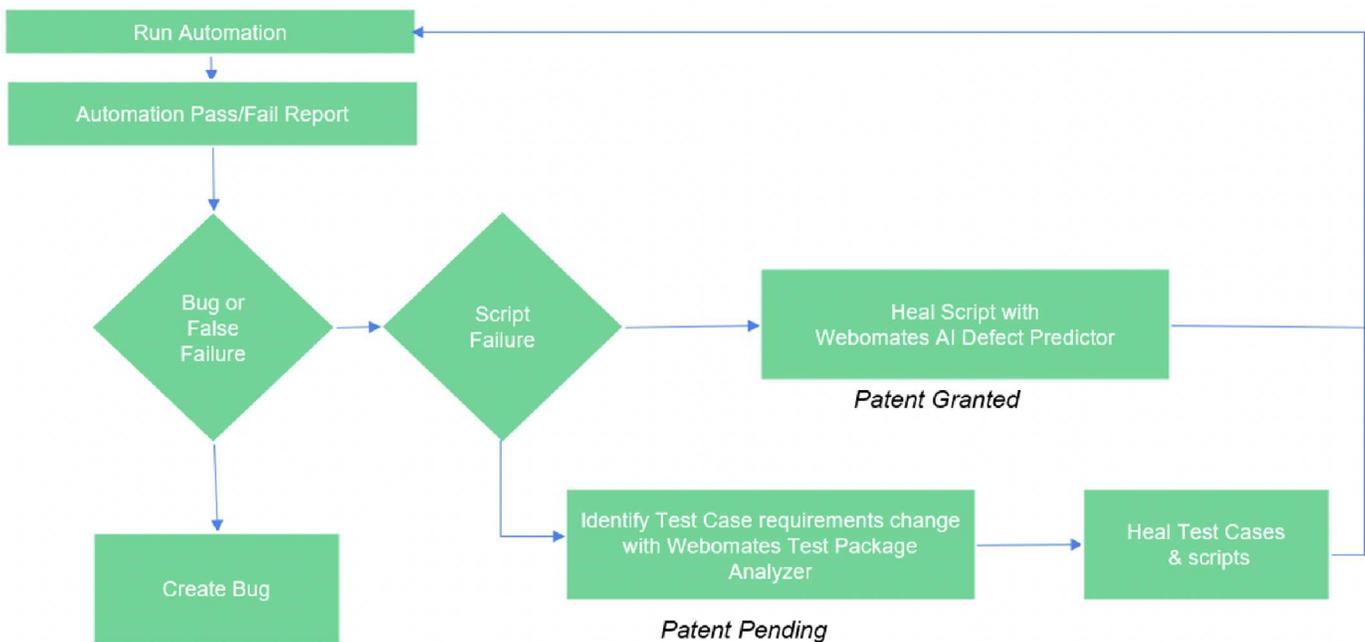

Figure 11: AI Test Automation

As the figure 11 shows above the patented AI Healing process allows Webomates CQ to achieve industry first SLA (Service Level Agreement). Webomates guarantees that with AI Healing all Test cases and test scripts will be healed in 24 hours. In addition, with a subset or a module of test cases and test scripts will be healed in 8 hours, typically used by customers to get healing overnight. In the AI Healing process Webomates CQ AI identifies the test cases and test scripts that need to be updated or added. It then generates or re-generates them. A member of the Webomates team reviews and refines them. Finally, after the execution is complete and at the customers leisure the modified or added test cases need to be approved or re-approved by the customer.
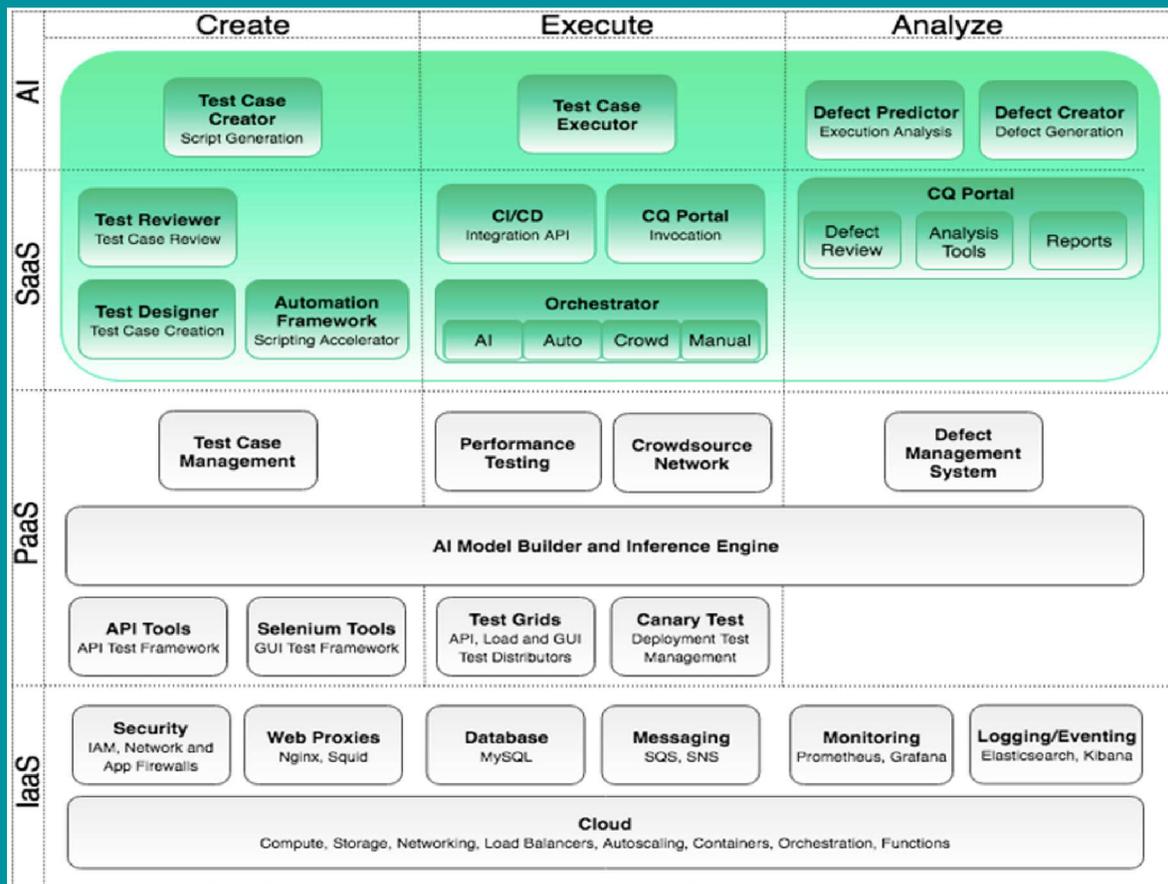
Webomates

The figure above shows the high level architecture diagram of Webomates CQ. The platform is built on a cloud services and deployed in the Webomates Cloud. There are over 40 micro services in the system. The overall areas of components are broken into:

1.   Create – This is the part of the system that generates the Test Strategy, Test Cases an dTest Scripts

2.   Execute – This section orchestrates the execution of the test cases and test scripts

3.   Analyze – This section analyzes the result of test case and script executions and carries out AI Healing. In addition a number of analytical reports such as shift left defect discovery are part of this subsystem

Web Portal Reporting Format

Webomates CQ portal provides access, initiation of software tests and test results for customer's software QA team.

The easy to understand and operate UI provides detailed reporting and analysis supporting multiple end users.

In addition, every defect identified by Webomates Software QA platform includes a twenty second video for software development, QA and Product Management teams to review, simplifying and reducing the time consumed during the review process.

Webomates

Summary

Over the course of 2021 and 2022 the large telecom customer and Webomates have successfully demonstrated the significant cost and time savings (50% or more cost savings along with an 80% reduction in software testing time) achieved with AI Powered Intelligent Test Automation when compared to today's legacy software test automation systems.

The new work being demonstrated to expand code coverage promises to lower time and effort by around 50%.

Other advantages to the customer are the positive impacts of fewer software bugs reaching production and the associated cost, accelerated time to market for new revenue generating features, applications and the refocusing of development resources away from software validation to creation of new applications and services.

Webomates

References

1. First Customer Production Pilot Cost Savings
2. First Customer Production Pilot software tet time velocity acceleration
3. Customer SRE requirements June 17, 2013
4. Patents list 6 patents :

- Classifying a test case executed on a software - US11068387B1
- Software defect creation - US11249890B2
- Test package analyzer - US20220206927A1
- Method and system for determining mapping of test case(s) to code snippets of computer program - US10175657B2
- Method and system for real-time identification of anomalous behavior in a software program - US10423520B2
- Method and system for multi-channel testing - US10545859B2
- Method and system for testing an application using multiple test case execution channels - US20200151089A1

Web◌mates