



TOP 11 CHALLENGES IN TEST AUTOMATION

& what to do about them



THE 11 MOST COMMON CHALLENGES IN AUTOMATION TESTING AND WHAT TO DO ABOUT THEM

In the world of automated testing, failed tests happen all the time. One small thing changes and suddenly you're drowning in red. And when you're testing continuously, small issues can easily grow into huge ones — which is why you need visibility.

When something breaks or doesn't work as well as it should, it's important to know why. You need the visibility and fast feedback to be truly considered "DevOps." You also need these insights so you can take corrective action.

At Perfecto, we've found that 80% of issues have a pattern. That means that teams can easily overcome the majority of their issues just by identifying the root cause.

And that's where smart analytics are key.

With Perfecto's [Smart Test Reporting and Analytics](#), we've discovered the top eleven most common test automation issues for teams. Keep reading to learn more about the most common testing challenges and how you can overcome them.



THE MOST COMMON ISSUES IN TEST AUTOMATION

Issues With Your Test Scripts and Frameworks

The most common sign your test automation is failing is if you encounter issues with test scripts or test frameworks. In fact, these account for 40% of all issues that DevOps teams face.

Why?

Test script and test framework issues stem from problems with skillsets, culture and processes, and an overall lack of communication between testers and developers.

1

Issues With Objects

Particularly problematic to DevOps teams are objects. With object identifiers, too often teams lack the knowledge to define the right object being used. This is especially true when dev designs pages featuring multiple objects with the same ID. Two similar objects on the same script are sure to cause issues in automation.

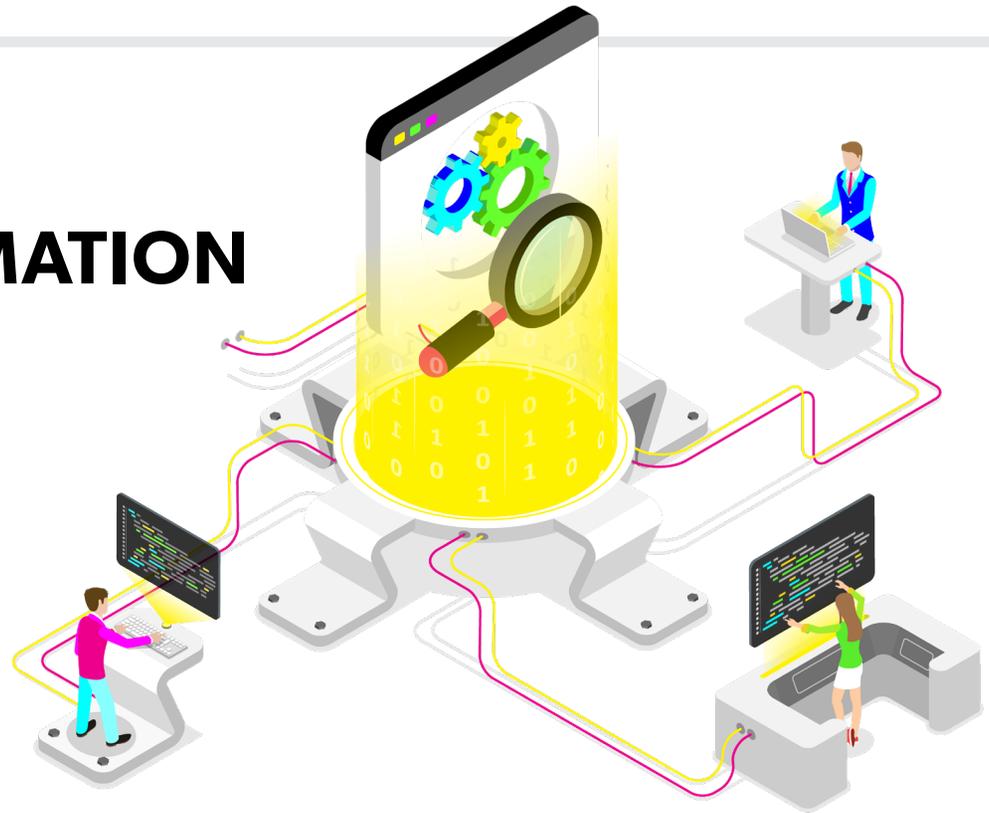
Teams can address this issue head on with the help of a page object model (POM). This design pattern ensures that if something changes, it all changes from one place. Some tools, such as [Perfecto Codeless](#), can help with object identification and script maintenance thanks to AI-powered technology.

2

Issues With Processes

Equally problematic are the culture and processes within a DevOps team. The lack of communication between testers and developers can result in an array of issues. Take for instance this common scenario: developers change the object IDs without telling test engineers.

Process and culture changes are needed to close the communication gap between testers and developers. DevOps teams need enact communication standards in order to get all team members on the same page. For instance, when objects change by a developer, that needs to always be communicated to test teams.



3 Issues With Test Code

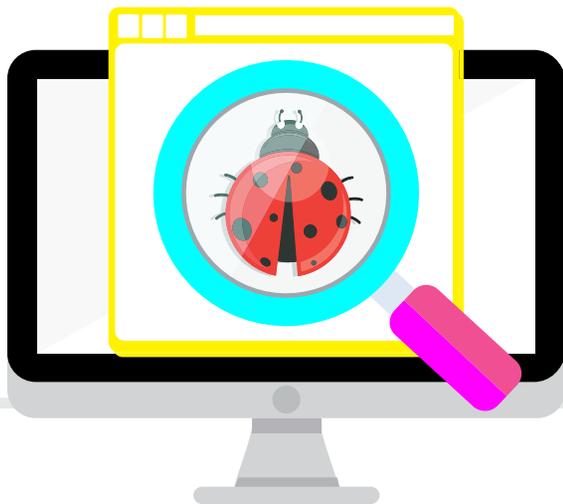
If teams lack the skillset or best practices related to coding, issues with test scripts can increase. For teams encountering this issue, reusability can help them maintain their code.

Teams can reuse scripts to mitigate issues. By treating test code as production code — which include code reviews, refactoring, and version control — code can be better maintained. Teams can also do periodic testing of their code, schedule debugging sessions, and identify issues with object identifiers ahead of critical regressions.

Issues With the Backend

Backend issues account for 30% of all automation issues teams encounter. These issues stem from network, availability, and data issues.

Teams need to ensure they are always working with the latest environment — whether it's test, staging, or production. They need to ensure that the data for the environment is up to date and will produce the expected outcomes.



4 Issues With The Network And Availability

Another common challenge in automation testing applies to the network. Testing teams need stable and reliable network connections. That's because when there's a network disconnection, teams are unable to access specific services, such as databases, third party services, the API a specific app needs, access to specific testing environment, and the VPN. Any disconnection to the network causes delays for DevOps teams and blocks testing.

Another automation testing challenge teams encounter is problems with availability. Teams today virtualize specific environments for testing purposes. In general, if the environment isn't up and running in a cloud and you can't access what you need, your entire testing process collapses.

DevOps teams experience issues with the network and availability due to the architecture of the testing environment. Within a testing environment, you don't have many safe controlling systems, such as backups. Many test architectures are built with a single point of failure. When something fails, everything fails. The entire pipeline is affected.

Teams can solve their network and availability problems with a good monitoring solution. Monitoring helps them know when there's a problem, and they can fix it as soon it happens. A third-party lab, like Perfecto, can help.

5 Issues With Data

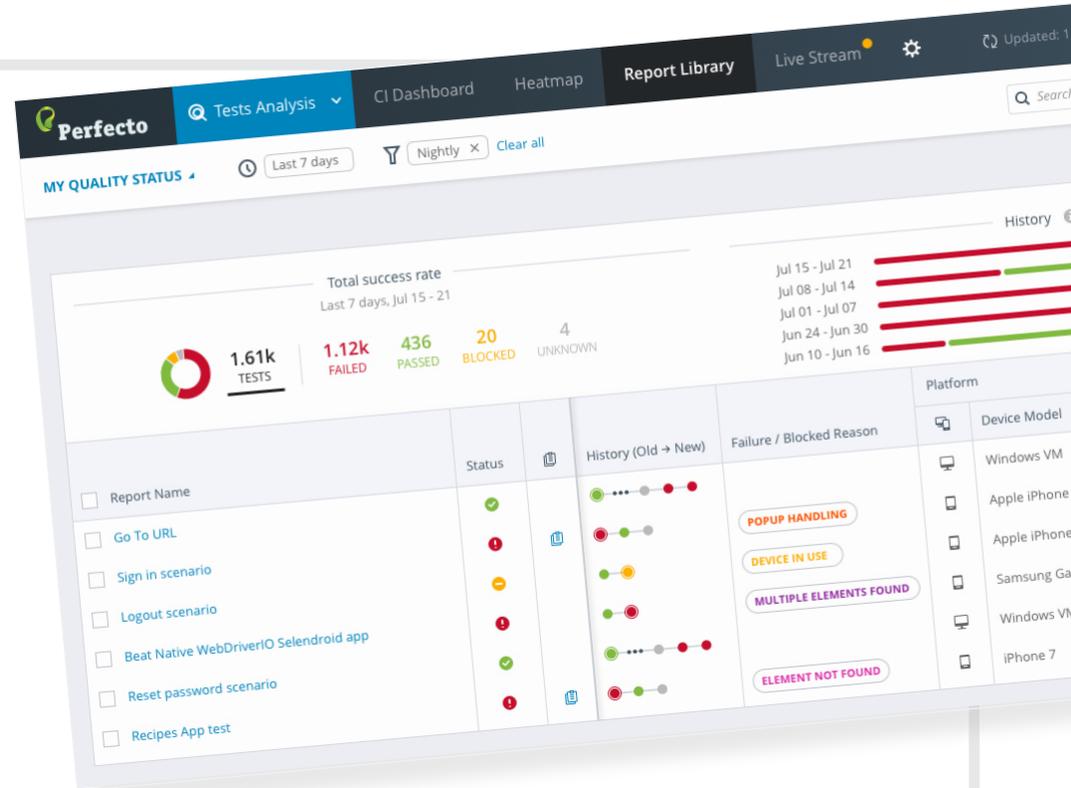
Another backend issue stems from data.

Too often, teams test against the wrong data. They use data that is outdated simply due to the fact that it takes them too long to gather and organize that data. If the data model wasn't updated correctly, outdated data can provide irrelevant or incorrect results.

Processes and communication can also keep teams from receiving correct and current data. Sometimes data is owned and generated from another team member, such as a data scientist. Disconnections can throw off data. And of course, communication gaps can affect the quality of the data as well.

Data accuracy and relevancy is needed for automated testing to be successful. DevOps teams need fast feedback in order to test off the latest data.

Through a sophisticated reporting and analytics solution, teams can achieve fast feedback. They can gain visibility into data lags. Data visualization can show them how it impacts the productivity of their team.

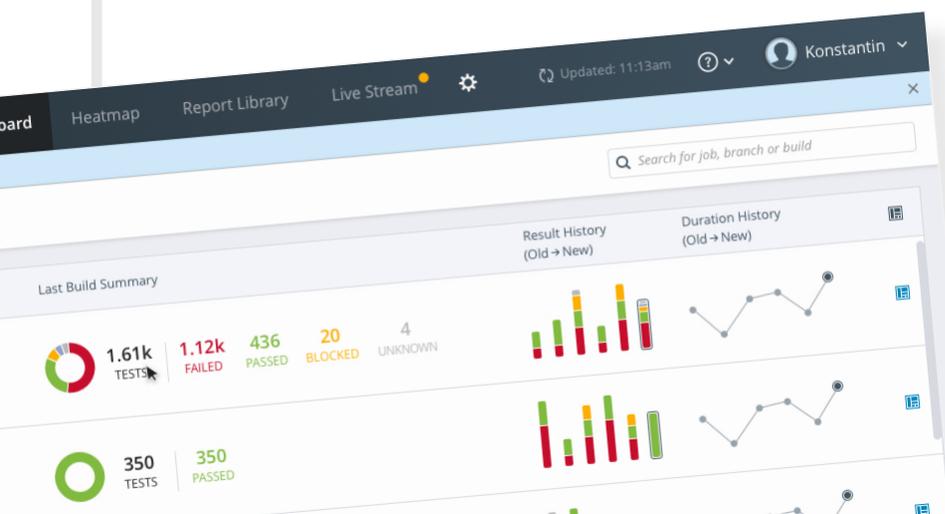


A tool such as Perfecto's Smart Reporting and Analytics can provide the visibility needed in all areas of the pipeline. It can provide a high-level visualization for executive teams. It can also dive deeper with detailed test reports and an automation dashboard. You'll get the visibility you need into the backend to find root causes of test automation failures.

Issues With Your Lab

Another common cause of test automation failure is from problems with the lab. These account for 15% of the issues teams encounter with test automation.

Problems with the lab stem from network, stability, and lock issues.



6 Issues With Networking

A main pain point with lab issues pertains to networking.

Within a lab, devices are used by customers. These devices are either connected to Wi-Fi, SIM card, or network services. But connectivity issues can wreak havoc on devices. When they disconnect from the internet, nothing works. Devices cannot connect with test frameworks, which disrupts all automated testing processes.

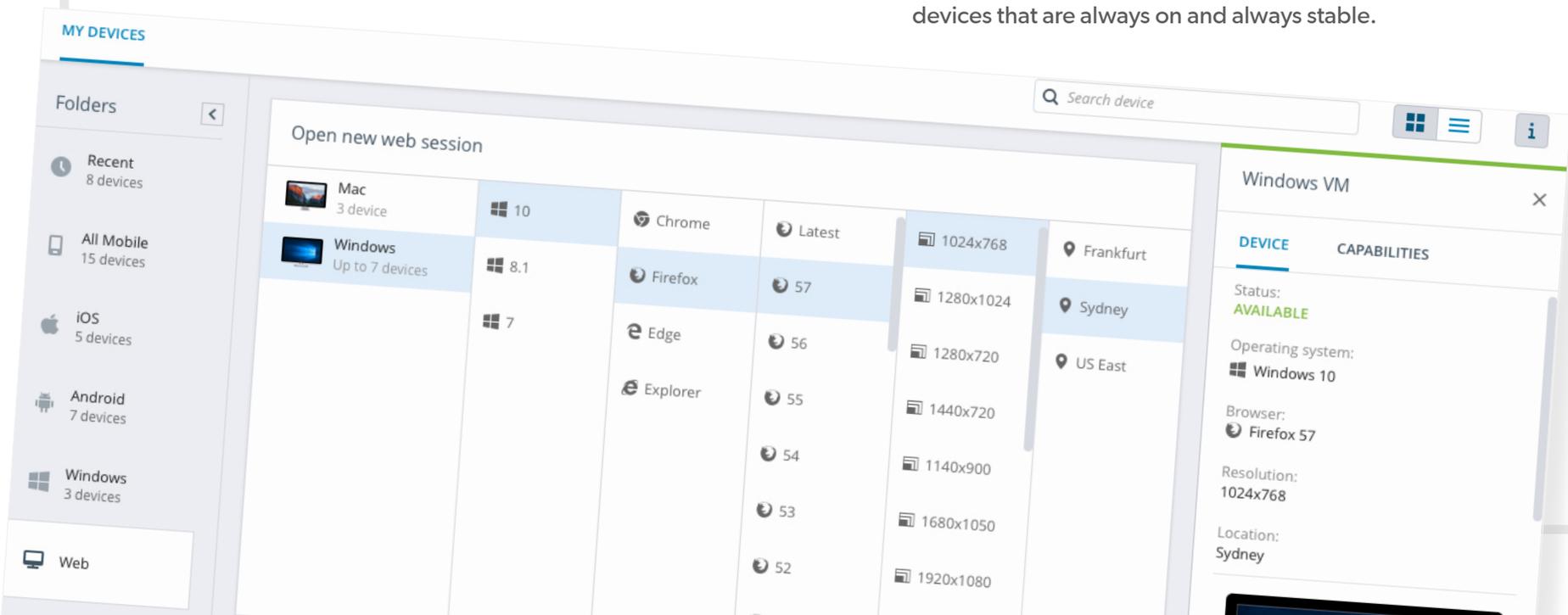
DevOps teams need dedicated support to specifically address networking issues. For instance, Perfecto is proud to have a Network Operations Center that monitors devices in the lab 24/7. If something gets disconnected, the center immediately reconnects it, so automation testing goes uninterrupted.

7 Issues With Stability

Another common issue with labs is the stability of the devices it houses. Mobile devices are unstable in nature. With the constant flow of new apps, new operating systems, updates, and performance issues, mobile devices have many elements in flux. If an app fails to install, if a mobile device has performance problems, these issues affect the stability of the device.

Why? Because labs are difficult to create and maintain. They require constant maintenance. For those who have their own lab, they need to continuously update it. They must constantly monitor the devices to ensure their stability and testability. With DIY labs, there's no one to help you with unstable devices.

Because of this, we recommend that you get help from the experts. Utilize a cloud-based lab with real devices to test, such as Perfecto's [Smart Lab](#). With 24/7 dedicated support, you'll be sure to test on devices that are always on and always stable.



8 Issues With Lock

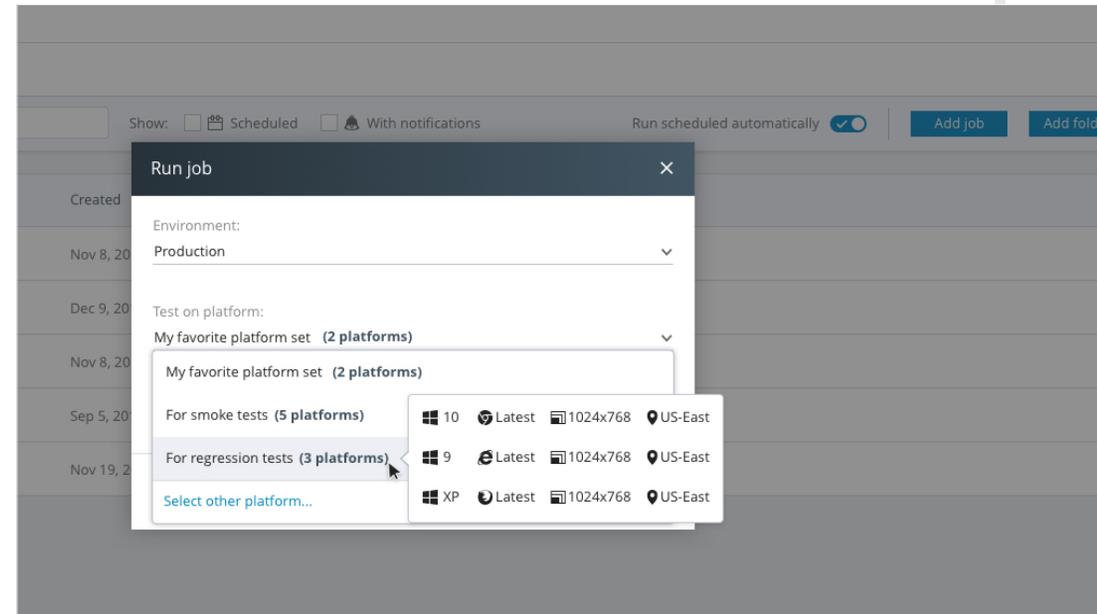
Issues with devices locking is another mobile testing challenge many teams face with automation. With security settings such as face ID, fingerprint authentication, and device passwords, being locked out of a mobile device is a very real possibility in automated testing. When devices are not being closed correctly in the lab, they can lock, preventing DevOps teams from automating their tests.

Issues with devices locking slows down DevOps and cuts into your automated testing.

Again, the solution here is a cloud-based lab with 24/7 monitoring that can prevent your mobile devices from locking up.

Issues With Your Executions

The fourth most common cause of test automation failure is from issues with test executions. Test execution issues stem from problems with licenses, in-use, and disconnection.



9 Issues With Licenses

This is a simple one — yet many teams encounter this problem when trying to automate their testing. Usually, DevOps teams do not have enough licenses or devices to cover their test execution needs. They do not have the right capacity to support their execution needs in order to cover a certain amount of tests in a certain amount of time.

To overcome such a problem, teams should leverage techniques such as sharding where executions are split across multiple devices. Or, they should perform a thorough sizing exercise to assess the required devices and platforms they need to accomplish their tests given the time constraints they have.

10 Issues With In-Use

Another issue within test execution is in-use — when one team member tries to execute something and another team member is already using it. In instances like this, team members interfere with each other. They don't use smart scheduling to prevent cases of in-use.

In this case, performing proper sizing of the test execution needs, the available resources, and the testing objectives can help prevent these conditions.

11 Issues With Disconnection

Another issue with test execution is disconnection. Too often, teams find the devices they need to test on are either disconnected or unavailable. Teams may try to test on a device that is offline.

In this case, a smart lab can help. It can proactively scan and identify network disconnections and perform a re-connect attempt. Another option is monitoring scripts. These poll the lab devices for connectivity. In case a call fails, it triggers a reconnect event.

The screenshot shows a dashboard titled "MY DEVICES" with a "COMING SOON" notification. A search bar is present with the text "Search device". On the left, there are folders: "Recent" (1 device), "All mobile" (73 devices), "Android" (32 devices), "iOS" (41 devices), and "Web". The main area displays "All mobile (73 devices)" and is sorted by "Availability". The devices are listed in a grid, each with a small image of the device, its name, model, OS version, and status "AVAILABLE". Each device card has an "OPEN" button at the bottom.

Device Name	Model	OS Version	Status
Apple iPhone-7 Plus	Apple iPhone-7 Plus	10.2.1	AVAILABLE
Apple iPhone-8	Apple iPhone-8	12.1	AVAILABLE
Apple iPhone-8	Apple iPhone-8	11.2.5	AVAILABLE
Apple iPhone-8	Apple iPhone-8	11.2.6	AVAILABLE
Apple iPhone-8 Plus	Apple iPhone-8 Plus	11.0	AVAILABLE
Apple iPhone-X	Apple iPhone-X	12.0.1	AVAILABLE
Apple iPhone-X	Apple iPhone-X	11.2.1	AVAILABLE
Apple iPhone-X	Apple iPhone-X	11.0.1	AVAILABLE
Apple iPhone-XS	Apple iPhone-XS	12.0	AVAILABLE
Apple iPhone-XS Max	Apple iPhone-XS Max	12.0	AVAILABLE
Google Pixel 3	Google Pixel 3	9	AVAILABLE
HTC 10	HTC 10	7.0	AVAILABLE
Samsung Galaxy Note 9	Samsung Galaxy Note 9	8.1.0	AVAILABLE
Samsung Galaxy S5	Samsung Galaxy S5	6.0.1	AVAILABLE
Samsung Galaxy S5 SM-G900A	Samsung Galaxy S5 SM-G900A	6.0.1	AVAILABLE
Samsung Galaxy S6	Samsung Galaxy S6	7.0	AVAILABLE
Samsung Galaxy S6	Samsung Galaxy S6	7.0	AVAILABLE
Samsung Galaxy S6	Samsung Galaxy S6	6.0.1	AVAILABLE
Samsung Galaxy S6	Samsung Galaxy S6	6.0.1	AVAILABLE
Samsung Galaxy S6	Samsung Galaxy S6	6.0.1	AVAILABLE

BOTTOM LINE

Most teams experience test automation failure to some extent. Know that you're not alone in facing these common problems.

If you're experiencing issues with your scripts, backend, lab, or orchestration, the key to overcoming them is through visibility. You need insight into where the issue is coming from and why it is happening, as visibility and insights are key to fully understanding an issue and resolving it.

Get the smart insights through AI, fast feedback, and the visibility you need. Be sure to check out Perfecto's Smart Reporting and Analytics to learn more.

SMART REPORTING & ANALYTICS

Related Content

[The Top 3 Roadblocks to Sustainable Continuous Testing](#)

[How to Calculate the Value of Analytics Solutions](#)

[Top 5 Challenges in Automation Testing and Analytics \(and How to Overcome Them\)](#)



ABOUT PERFECTO

Perfecto is a **Perforce** company. We enable exceptional digital experiences and help you strengthen every interaction with a quality-first approach for web and native apps through a cloud-based test environment called the Smart Testing Lab. The lab is comprised of real devices and real end-user conditions, giving you the truest test environment available.

More than 1,500 customers, including 50 percent of the Fortune 500 companies across banking, insurance, retail, telecommunications, and media rely on Perfecto to deliver optimal mobile app functionality and end-user experiences, ensuring their brand's reputation, establishing loyal customers, and continually attracting new users. For more information about Perfecto, visit www.perfecto.io, join our community follow us on Twitter at [@PerfectoMobile](https://twitter.com/PerfectoMobile).

TRY PERFECTO

