**Panaya** | **FORESIGHT**

# Salesforce Smart Testing Best Practices

## with Panaya Foresight

Building in Salesforce is easy, but keeping it from breaking is not. Keep your Salesforce org running properly from release to release by using better testing practices with Panaya Foresight.

One of the most appealing aspects of Salesforce is its out-of-the-box features that can be easily tailored to meet the specific needs of your business. But while it may be easy to update a new sales process or create a custom layer of code in Salesforce, the real challenge is making sure your customizations do not break over time as you continue to build and expand the complexity of your Salesforce org. Making new applications within Salesforce might be easy, but managing its downstream impacts and preventing bugs can become very challenging. To ensure business continuity, you need a robust testing plan in place so that you can continue to build, innovate and experiment without fear of failure. Smart testing in Salesforce is your insurance policy and your key to innovation and operational savings.

Are you testing too much? Are you testing too little? Are you covering all impacted business processes? Do you know what downstream impacts your new release changes might have? How do you address the multiple challenges and complexities of testing adoption to find the perfect balance between protecting yourself from risks in production and creating long and expensive test plans? If you're delivering a very small change, do you really need the entire test catalog or regression suite? What testing can be automated and what testing still needs to be done manually?

**If you've been asking yourself any of the questions above, you're in luck: Panaya has the answers.**
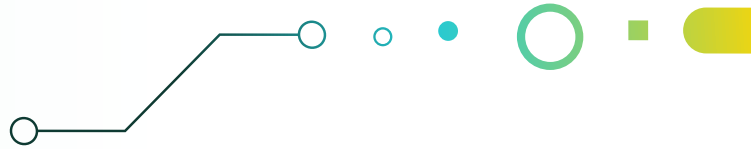
BOOK A DEMO

# THE CHALLENGES OF SALESFORCE TESTING

Highly customizable environments like Salesforce allow different technical methods to achieve the same business goals, while also allowing you to deliver quality on time without stopping your development schedule. However, the variety of methodologies in Salesforce can create the potential for significant breakage and discerning how to test to keep releases efficiently moving can often result in oversight and trade-offs between competing priorities. There can be a lot of uncertainty about change impact and test coverage, as trying to understand the potential ripple effects of the work you're about to release can feel next to impossible. There is a need in Salesforce testing for a solution that details and encompasses how to scope and test correctly to ensure there are no gaps in your testing plan and code coverage.

Salesforce testing requires a strong and precise understanding of business processes–it's not just checking for where bugs fire in code. Business users are essential to validate your changes, and understanding how business users act in their environments requires verification and testing before release. Because managing tests across multiple teams requires end-to-end visibility, you need to be able to define who is responsible for what in E2E testing and need full insight into where work begins and ends during the entirety of your test process.

BOOK A DEMO

Additionally, User Acceptance Testing (UAT) needs to be done by business end-users who still have to do their everyday job. While end-user testers are critical to ensuring that the new code behaves as the business would expect it to, testing is still overhead for them, and can be viewed as a tedious process they don't like to do. But you need UAT to understand that existing processes aren't disrupted by new builds.

BOOK A DEMO

# SALESFORCE TESTING TYPES

To identify what to test and where to locate efficiencies within your testing plan, it's important to understand the kinds of testing needed for effective Salesforce releases.

## UNIT TESTING

Unit Testing is a staple of software development in any language or platform and has to be done on each unit of code in a single component. In Salesforce development, Unit Testing is usually performed by Apex Developers for things like Visualforce or Lightning Components. Unit Tests are done at the coding level, checking to make sure it works.

## FUNCTIONAL TESTS

Functional tests are a step above unit tests and are used to validate the system against the functional requirements. Functional tests check the code against the system as a whole, ensuring that the system is operating in the way you would expect in the later use of the application. Functional tests are usually performed by a QA team

## USER ACCEPTANCE TESTING (UAT)

Often the most manual part of testing, UAT is performed by business end-users who can define the process and method that the new code is supposed to represent. UAT is done in the development stage in a sandbox and is testing functional requirements before deploying to production before wide release to other Salesforce end-users.

BOOK A DEMO

## REGRESSION TESTING

Regression testing is the most comprehensive system test, and many command servers and DevOps pipelines often require regression tests. A regression test is a standard test you can run to see if anything has inadvertently broken something–it's designed to check that a bug fix you just released did not create a new bug. A regression test ensures changes are not affecting any existing features and can be automated. Regression tests are usually handled by a QA team.

## SYSTEM TESTING/E2E TESTING

System and E2E testing are critical in orgs with many managed packages or APIs. If you want to check an external API, an automation in Marketing Cloud, or that a PDF generator is performing after a new release, you need to run a system test or an End-to-End test to test technical processes from start to finish. A few different kinds of people can perform E2E testing, but usually a Salesforce Expert, an external consultant or a system owner of an API-connected solution perform these tests.

BOOK A DEMO

# SALESFORCE TESTING BEST PRACTICES

Testing can feel like a chore, but with planning and the right tools, it doesn't have to feel impossible. Here are a few best practices that will make your release testing go much better.

## 1 START TESTING EARLY IN THE DEVELOPMENT LIFECYCLE

For Apex classes especially, plan your testing as you build. Build a test class alongside your code and ensure your class can pass at the 75% coverage mark. However, if you have the resources, it's also not a bad idea to have external partner or QA personnel write the test class for you–developers can sometimes only test for the "happy path", but having a third party review your work can provide insights into non-happy paths. As you work, deploy and test regularly throughout the build. Developers should unit test the code they have developed as they work at their requirements to validate their code is working as intended. While going through unit testing, focus your testing efforts on the highest areas of risk to maintain efficiency and identify priorities in testing. If you can, involve Test/QA functions in the project early and often.

## 2 PREPARE YOUR FUNCTIONAL TEST CASES FROM THE START

Within your projects, lay out who is responsible for what test, what testing you plan to involve and what manual steps need to be done (if there are manual steps). Create a detailed test plan including: Features and scenarios to be tested

- Identification of test cycles and acceptance criteria

- Roles and responsibilities: this is especially important with your UAT testers, as they may need to adjust their schedule to accommodate testing.

- Defect management strategy and how quickly you plan to address bugs or wonky code

- Test documentation strategy to support compliance, documentation and end-user training

BOOK A DEMO

## 3 NEVER COMPROMISE UAT TESTING

UAT is a critical connection between the business and the developers working on a Salesforce build. Developers can sometimes be blinded to the realities of other people's work, so focusing on integrity in the UAT process can help you put your end-users front and center.

- Ensure the functionality developed meets the business need and works efficiently for the business. Technically meeting requirements but making work harder on your end-users isn't really meeting requirements.

- Assemble an experienced team of key users for your UAT and define what you consider reliable feedback.

- Ensure full visibility of the UAT testing process for End to End

- Use your users' feedback for future releases and determine what new priorities may emerge.

- Leverage business users' knowledge for potential avenues of automation

## 4 USE TESTING TOOLS TO STREAMLINE YOUR PROCESS

With the growing complexity of Salesforce development, Salesforce developers know they need tools to do testing efficiently. While identifying and acquiring testing tools is often a management challenge instead of a technical challenge, automatic invocation of testing in your automation pipeline helps. Think about lean management principles when you do find your defect and leverage automation where possible. Make sure you have a small queue for defect management, and where needed, be able to swarm on a complex defect to keep developing progress moving along.

- Apply end-to-end thinking and manage your testing accordingly: think of requirements, test planning, execution, defect management and real-time monitoring in terms of a comprehensive picture.

- Automate test evidence and business process documentation where possible.

- Ensure defect closure with smart defect management workflows.

- Track and share accurate testing progress with relevant stakeholders.

BOOK A DEMO

# SMART TESTING IN ACTION
# WITH PANAYA FORESIGHT

For smart testing and managing business users, consider using a comprehensive testing and change management tool like Panaya Foresight. Panaya Foresight provides real-time Impact Analysis with the click of a button by monitoring changes in your Salesforce org's metadata. With a simple button click, you can see the entire ripple effect of any architectural and metadata change to give yourself X-ray visibility into your org. By spending two minutes of your time and authenticating through Panaya Foresight, we can pull all your interdependencies and pinpoint exact downstream impacts.

Include accelerated testing alongside your test automation management as a holistic solution. Accelerated testing is not always the same as test automation–UAT testing still has to be performed by a human. However, manual testing activities can be accelerated and streamlined by using Panaya to centralize your test criteria.

Actionable insights for achieving and maintaining a healthy org are driven by Panaya Foresight's Smart test management, which gives users a unified end-to-end testing platform that brings intelligence to test management.

**When you're shopping for testing solutions, consider if you're buying a first-generation or a second-generation.**

**It's important to look at newer technologies in your test tech suite and keep them in the cloud. Condensing tools for release management and testing should be a long term part of your tech stack management strategy to facilitate.**

BOOK A DEMO

# TRY PANAYA FOR THE SWEET SPOT OF RISK-BASED TESTING

Determining what to test to minimize breakage in production is a critical part of establishing your Salesforce testing strategy, but identifying what to test and how to create that plan can feel very overwhelming. The potential for missing critical steps in your testing plan is large and high risk, but addressing and establishing a comprehensive and efficient test plan doesn't have to be an uphill battle. By using the right tools that integrate multiple elements of your design and testing process, you can quickly identify what changes in your deployment have the highest risk of breakage and focus your test efforts around preventing them. With Panaya Foresight, you can also consolidate your test feedback and streamline your documentation creation. You can also centralize and consolidate your User Acceptance Testing responses and quickly reproduce bugs or unexpected behavior to address them.

Panaya Foresight can help you operate from a risk-based testing perspective to help you scope your test activities on the items that matter and have the largest potential for impact. With Panaya Foresight, you can dramatically simplify your testing and release management strategy to release faster with fewer defects.

**For more information on Panaya foresight Smart Testing Solution BOOK YOUR PERSONALIZED DEMO**

BOOK A DEMO

# ABOUT PANAYA

Panaya provides cloud-based application delivery and testing solutions that provide collaboration between Business and IT.

Enabling enterprise agility with faster release velocity and uncompromising quality, Panaya delivers an optimized user experience with end-to-end visibility of the application lifecycle.

Since 2006, 3,000 companies in 62 countries, including a third of the Fortune 500, have been turning Panaya to deliver quick quality change to enterprise applications.

BOOK A DEMO

O Panaya | FORESIGHT