



EBOOK

Essential Salesforce DevOps Tactics in 2022

Three industry leaders discuss how to maximize returns from your DevOps pipeline.

INTRODUCTION

The world of Salesforce DevOps is constantly evolving. New trends, new tools, and new best practices are always emerging to help developers make the most of their efforts. And as we reach the end of another year, we wanted to take a moment to speak to other leaders in Salesforce DevOps to explore a few subjects that should be on everyone's radar in 2022.

We'll hear from CodeScan, providers of powerful static code analysis tools to provide constant visibility into the health of your code. ACCELQ offers AI powered codeless test automation that drives cross-functional ownership of quality and increased collaboration. And AutoRABIT offers a full DevSecOps platform for all of your Salesforce development needs.

In this ebook, DevOps experts provide valuable insight into how to accomplish the following goals in 2022:

1. Support Data Security and Successful Deployments with High Quality Code
2. Accelerate Salesforce Implementation with AI-Powered Test Automation
3. Synergize the Various Elements of DevSecOps with a Unified Automated Release Management Platform

Support Data Security and Successful Deployments with High Quality Code

The need for strong code is not a new concept. However, the tools and processes that can be used to improve your code quality have gotten better. But strong code is more than reliable functionality and cool features—it also impacts your data security measures.

Salesforce data security was a major focus over the course of 2021 as many high-profile companies experienced costly data breaches. These security lapses highlighted the need for a multi-tiered approach to keeping system data safe.

Strong code shores up potential vulnerabilities and supports other aspects of your data security strategy. Static code analysis is no longer an optional tool in your Salesforce DevSecOps toolbox. The constant oversight it provides supports successful deployments and reinforces your data security measures by ensuring there aren't gaps in functionality for bad actors to exploit.

We'll look deeper into these considerations for high quality code in 2022:

1. How Does Code Strength Impact Data Security?
2. Benefits of Static Code Analysis
3. How to Improve Salesforce Code Quality

How Does Code Strength Impact Data Security?

Code quality might seem separate from securing your Salesforce environment, but they are actually connected. Here are some ways your code can influence your data security success.

1. Coding Errors Enable Data Loss/Corruption

There is a chance data will be miscataloged, forwarded to the wrong location, or simply lost if your application has coding errors.

Proper coding structures ensure your application functions correctly and accurately processes the data included within it.

2. Bugs Create Vulnerabilities

Misfires within your Salesforce environment from bad code can create opportunities for bad actors to exploit weak points and gain access to your system.

Think of your Salesforce data security strategy as a boat in water. A proper strategy will find any holes before the boat gets near the ocean.

3. Improper Code Can Create Unwanted Changes

Bad code can interrupt existing functionality and result in unintentional changes to your live environment.

Testing is an essential aspect of ensuring high quality code. Checkpoints along the way are great for catching errors but finding issues in real time is ideal.

4. Main Types of Security Liabilities from Bad Code

SQL Injection: An attacker gains access to your system through cracks in your Salesforce data security created by coding errors. They then inject malicious code and interfere with queries made to the database.

Cross-Site Request Forgery: Personal information can be exposed when a bad actor forces a user who has already been verified in an application to perform unwanted actions.

Cross-Site Scripting: Malicious scripts are injected into a website. The users are targeted for their personal information instead of targeting the website itself.

Broken Authentication: A bad actor steals the authentication credentials of a verified user to gain access to the system. This allows them to pretend they are the user to access records and perform functions.

Security Misconfiguration: There are many options for configuring security settings within a Salesforce environment. Improperly configuring these settings creates an opportunity for hacks.

Benefits of Static Code Analysis

Now that we know how improved code quality assists your data security measures, let's look into other reasons implementing static code analysis is a necessity for optimizing your Salesforce DevSecOps efforts in 2022.

1. High Quality Code

Salesforce development releases are intended to perform a specific function. This is best achieved with flawless coding structures.

Continuous testing and health checks on your lines of code ensure that the update or application will function properly. Consistently high-quality releases position you as an industry leader in the minds of your customers.

2. Increased Visibility

Static code analysis tools provide dashboards and reports for you to perform high-level analysis on the overall health of your code.

Realtime information into the state of your code allows you to make any necessary changes to ensure the final product meets your standards and expectations.

3. Reduced Technical Debt

Many Salesforce development teams focus on quick deployments, knowing they'll have to go back and address these shortcuts in the future. The growing amount of future work is what's known as technical debt.

Static code analysis detects errors as they occur, drastically reducing the technical debt facing your team as you move through the release phase of the development cycle.

4. Higher Release Velocity

Automating code reviews frees up your team members to spend time on moving a project through the pipeline or even getting started on the next release.

Integrating static code analysis tools with a dedicated CI/CD platform further streamlines these processes and enables your team to produce fast and reliable releases.

5. Saved Money

Optimizing and streamlining your Salesforce DevOps pipeline not only helps position your company as a leader in the industry, but it also reduces costs throughout the process.

Static code analysis tools save your team members from wasting their time. And saved labor hours equate to saved money.

How to Improve Salesforce Code Quality

Static code analysis is an essential aspect of improving the quality of your code. However, it is not the only thing you can do to ensure your projects are stable and secure.

1. Utilize Automation Whenever Possible

Automating manual processes reduces errors and increases data security. Repetitive processes take up a large amount of team member time.

Automated processes assist in making a profitable DevOps pipeline, but they also contribute to a reduction in coding errors and more successful deployments.

Tools such as CI/CD support high quality code by introducing more levels of testing that might otherwise be performed manually. Any errors that would have made it through this stage could contribute to a failed deployment or a faulty product.

2. Give Your Developers Enough Time

Salesforce test automation will have a large impact on the success of the products developed by your team members. However, your team members are ultimately your most valuable resource when it comes to writing high quality code and creating stable and secure development projects.

Salesforce test automation frees your team members up to pay proper attention to their tasks and create the most stable projects possible.

3. Avoid Errors with Proper Planning

There are certain things your DevOps team can do to ensure their efforts are working toward the same goal. This starts in the planning stage before anybody starts putting together lines of code for the project. Make sure everyone on the team is clear on what they need to be working on and who they can go to if a question comes up.

Clear communication and a defined hierarchy are essential to avoiding common errors in the development process.

4. Attract Experienced Team Members

Better developers will simply produce better lines of code. The work atmosphere will have a large impact on attracting more talented applicants. Quality DevOps tooling makes the work much more pleasant to produce, which creates a positive work atmosphere.

People want to work in a place that supports their efforts. Enabling your developers with quality DevOps tools will attract more qualified team members.

How to Improve Salesforce Code Quality cont.

CodeScan is a static code analysis solution built specifically for Salesforce. Equipped with the most comprehensive insights for Salesforce languages and Metadata, CodeScan enables organizations to more quickly create high-quality features, evaluate and reduce outstanding technical debt, define and deliver to code standards, and meet security and compliance requirements.

Visit www.codescan.io to learn more.

Accelerate Salesforce Implementation with AI-Powered Test Automation

Of the many enterprise systems available in the market today, Salesforce has been a top contender in every category. Salesforce enables teams to close more deals and grow sales faster by offering an intelligent set of features and capabilities. Given these cutting-edge capabilities, a new study from IDC finds that the Salesforce economy will create 9.3 million jobs and generate \$1.6 trillion in business revenues by 2026!

Organizations across sectors are constantly vying for the option of implementing Salesforce to propel their business towards success. But when it comes to accelerating the implementation process, they often hit a roadblock with testing. If you want to accelerate Salesforce implementation, you cannot rely on traditional test automation tools. Instead, you need AI-powered test automation capabilities.

Let's explore –

1. The Challenges with Traditional Test Automation Tools
2. The Benefits of AI-powered Test Automation
3. Benefits of AI-Powered Test Automation in Salesforce Implementation

The Challenges with Traditional Test Automation Tools



Test automation tools like Selenium, although powerful, do not deliver the right test automation results for Salesforce applications. Despite offering capabilities to automate user interactions, they do not cover all testing requirements. Since applications built on Salesforce are highly complex, testing them using traditional test automation tools can be challenging.

- › Since Salesforce applications have front ends built with dynamic frames, traditional test automation tools cannot handle these frames effectively. Even if they can, they take a lot of time and effort to find the right attributes – especially with nested and hidden frames.
- › Traditional tools are poor at handling dynamic Salesforce elements with no fixed ID, name, class, or CSS attributes. Therefore, it becomes challenging if we hard-code the locator of the element.
- › Minor UI changes make the tests extremely flaky while delivering synchronization issues with element loading.
- › Because Salesforce applications tend to have different database-driven tables, with dynamically created rows managed by different tabs, simple actions like ticking a checkbox can be complicated and prone to errors.
- › Salesforce applications can have complexities like Shadow DOM and pop-up windows, which traditional tools do not support directly.
- › Challenges across test case maintenance are also many, and most tasks require extensive custom coding – that eventually impacts the quality and timeliness of continuous testing goals.

The Benefits of AI-Powered Test Automation



AI-powered test automation tools take a modern approach to manual testing and are built to complement core state-of-the-art automation capabilities. Such a consolidated approach to testing brings significant speed and efficiency to the Salesforce implementation process, setting the stage for continuous testing and robust Agile quality management on Salesforce.

Whether you want to improve the Salesforce user experience, upgrade to the latest version, or implement a new module, these tools can:

- › Automate business process validation across the Salesforce implementation.
- › Easily handle the complexity of lightning controls.
- › Cut down on manual testing efforts.
- › Effectively deal with dynamically generated and nested frames.

Benefits of AI-Powered Test Automation in Salesforce Implementation

Let's look at how AI-powered test automation can accelerate your Salesforce implementation:

Seamless Lifecycle Management: With next-gen test management capabilities for manual and automated tests, AI-powered test automation tools enable intelligent test planning and change management.

Codeless Automation: AI-powered test automation tools also deliver functional virtualization capabilities, allowing organizations to achieve in-sprint automation optimized for Salesforce-specific technology. Such capabilities enable testing teams to shift the testing process left and automate tests – without writing a single line of code.

Quick Turnaround: Lightning-fast Salesforce applications need lightning-fast testing. AI-powered test automation tools provide the easiest, fastest, and most sustainable way to automate Salesforce testing. Using these tools, one can automate all aspects of the quality lifecycle – from test design all the way to test planning and execution – and efficiently meet the scalability and security needs of your Salesforce applications.

End-to-End Validation: Since AI-powered tools offer seamless integration with UI and functional testing tools in a single unified flow, they allow for end-to-end validation of complex Salesforce applications – across Classic, Lightning, and Lightning web components. Native integration

with CI/CD tools like Jira, Jenkins further ensures automation for build verification.

Self-Healing: With AI and ML at the core, modern test automation tools deliver an array of Natural Language, no-code automation capabilities with self-healing features. Since tools learn and adapt with time, they can automatically update and adjust testing. Such capabilities allow teams to run reliable test executions, save time and effort of script maintenance, prevent flaky tests and minimize test failures.

Detailed Test Scenarios: AI-powered test automation tools also drive coverage and traceability for higher quality outcomes. By developing test scenarios with predictive analytics and building a visual blueprint of the Salesforce application, such tools help drive automation across the quality lifecycle with a deep business process focus.

Salesforce Release Alignment: Test automation tools built especially for Salesforce applications also ensure that the testing efforts align with the latest releases. Regardless of what Salesforce application or version an organization wants to implement, these AI-powered tools can ensure smooth implementation and upgrades in line with that version.

Benefits of AI-Powered Test Automation in Salesforce Implementation *cont.*

Salesforce applications, although very proficient, are not easy to implement and maintain. For Salesforce users, it's important but hard to stay ahead of the latest updates and releases and make the most of new and improved features. Unfortunately, most organizations looking to achieve these goals or accelerate their Salesforce implementation use traditional testing tools, not realizing that these tools are incapable of matching the complexity of Salesforce technologies.

Embrace an AI-powered test automation tool if you want to keep up with the latest changes, make quality enhancements to your Salesforce deployment, or improve your Salesforce implementation outcomes. Such a tool can help you achieve the maximum impact by delivering Salesforce-level features such as seamless lifecycle management, codeless automation, end-to-end validation, self-healing, detailed test scenarios, and release alignment. In addition, a modern and competent test automation tool can make your test cycles faster while lowering effort on test automation for your Salesforce applications.

ACCELQ offers a modern, AI-powered codeless test automation and test management platform that allows organizations to streamline and simplify the test automation process. As a next-gen tool with AI and ML at its core, the tool paves the way for in-sprint, codeless automation, and end-to-end validation - perfect for those looking to make their business DevOps and Agile ready.

Learn more at www.accelq.com.

Synergize the Various Elements of DevSecOps with a Unified Automated Release Management Platform

Proper automated release management involves the utilization of a series of powerful tools. These tools can be cobbled together from multiple places, but this is a shortsighted view. Finding and sourcing tools that are perfectly aligned with each other is the best way to optimize your Salesforce DevOps efforts. Salesforce DevOps is a beneficial but often misunderstood pursuit. There are a variety of options for tools, processes, and approaches to this widely used development methodology. Simply adopting the DevOps mindset isn't enough to see remarkable and immediate positive results.

Automated release management bolsters your Salesforce DevOps efforts through intentional practices and powerful tools.

Release management is the process of guiding a singular version of an update or application through the development process to deployment. Branching is often utilized to allow multiple developers to work on a singular project without impacting the main source of truth. These branches will need to be combined prior to deployment while avoiding overwrites and other potential errors.

We'll explore these essential aspects of how a unified automated release management platform assists your DevSecOps efforts in 2022:

1. Risks of Using Multiple Sources for Release Management Tools
2. Benefits of a Unified DevSecOps Platform
3. Selecting a Salesforce Automated Release Management Platform

Risks of Using Multiple Sources for Release Management Tools

There are a lot of options for businesses to address their release management needs. Piecemealing together a Salesforce release management suite might seem like a great way to get everything you need but utilizing multiple vendors for services can create problems.

Your team members can become confused when they have to hop from one service provider to the next, creating the potential for costly errors.

A major aspect of a streamlined release management strategy is to make it easier for your team members to perform their tasks. Hopping between various tools will become cumbersome. A complete automated release management platform that exists within one interface removes this potential for confusion. Your team won't need to learn how to interact with multiple interfaces, making the training process much shorter and easier.

A Salesforce DevOps pipeline involves the efforts of multiple teams. It's this constant attention from multiple angles that streamlines the process as a whole and helps your business see the fastest returns possible. However, this is only possible if all of these various teams remain on the same page and are able to openly communicate with each other.

A singular Salesforce release management suite keeps all efforts in the same location, making it easier for team members to get the information they need.

Increased release velocity, reliable data security measures, and stronger code are all possible when collaboration is encouraged and facilitated. Linking together various DevOps tools assists release management. This is particularly true when all these tools work together seamlessly.

Benefits of a Unified DevSecOps Platform

1. Reduction in Errors

Coding errors have the tendency to snowball and become more complicated and expensive when they aren't found until late in the process.

Automated release management consists of many safeguards to keep bugs and errors from going unnoticed until deployment.

2. Increased Release Velocity

It's difficult to push new projects forward when your team members are busy correcting mistakes and troubleshooting existing projects.

Interconnected automated release management tactics and tools won't be bogged down by correcting past mistakes, allowing you to push updates through to deployment much faster.

3. Higher ROI

A successful product that doesn't contribute to your company in any way can't be considered a success.

Intentional practices and unified automated tools decrease wasted team member time and contribute to higher overall returns for each project.

4. Standardized Release Strategy

Automated release management provides a set of tactics and tools which can be improved over time.

Standardizing your release strategy and using

complementary tools allows your team members to become comfortable with the process and know what to expect.

5. Enhanced Data Security

Simple user errors or even outages caused by natural disasters can trigger a massive loss in data.

A reliable backup and recovery tool is an essential aspect of a complete automated release management strategy and will be included in a complete DevSecOps platform.

6. Consistency

Pre-established quality assurance checkpoints will be able to bring everything together in a uniform fashion.

Automated release management on a singular platform supports consistent results because it offers the ability to accurately repeat methods and processes numerous times.

7. Audits and Reports Provide Useful Insights

Learning from every Salesforce DevOps project will continue to enhance your development pipeline, optimizing and streamlining your processes.

An automated release management platform provides benefits as soon as the tools and processes are integrated into your Salesforce DevOps pipeline.

Selecting a Salesforce Automated Release Management Platform

A unified automated release management platform will be one that provides these capabilities:

- › Version control management
- › Continuous integration and continuous delivery/deployment
- › Robust deployment capabilities
- › Powerful data migrations
- › Test automation
- › Comprehensive integration with Salesforce DX
- › Intelligent reports
- › Environment creation and management
- › Static code analysis

Your Salesforce DevSecOps efforts are likely to expand. You want to be sure your platform is able to handle all of your needs. AutoRABIT provides the only unified DevSecOps platform for Salesforce that enables you to perform these essential functions and see the greatest possible returns from your development efforts.

AutoRABIT is a Continuous Delivery suite for SaaS platforms. We automate and accelerate the entire application development and release process. This enables continuous integration and delivery by providing fast, simple, and secure end-to-end automation across all Salesforce implementations. We help enterprises achieve higher release velocity and faster time-to-market.

Visit us at www.autorabit.com to learn more.

CONCLUSION

2021 presented a wide variety of unique challenges for every industry. Those that were able to best adapt their operations were rewarded. 2022 is sure to continue this trend of a great need for flexibility. And the best way to accomplish this is with a streamlined DevSecOps pipeline.

Strong code, ample testing, and a unified DevSecOps approach will provide the foundation you need for secure and profitable development projects.

Data security and stable releases are a hallmark of successful development projects, and that's not going to change. The tools and processes used to create reliable releases, however, will continue to evolve. Staying on top of these changes and utilizing the newest capabilities will position your company as a leader in your industry.



**Ready to accelerate your
Salesforce Continuous Delivery
Journey with AutoRABIT?**

[CHECK US OUT](#)

