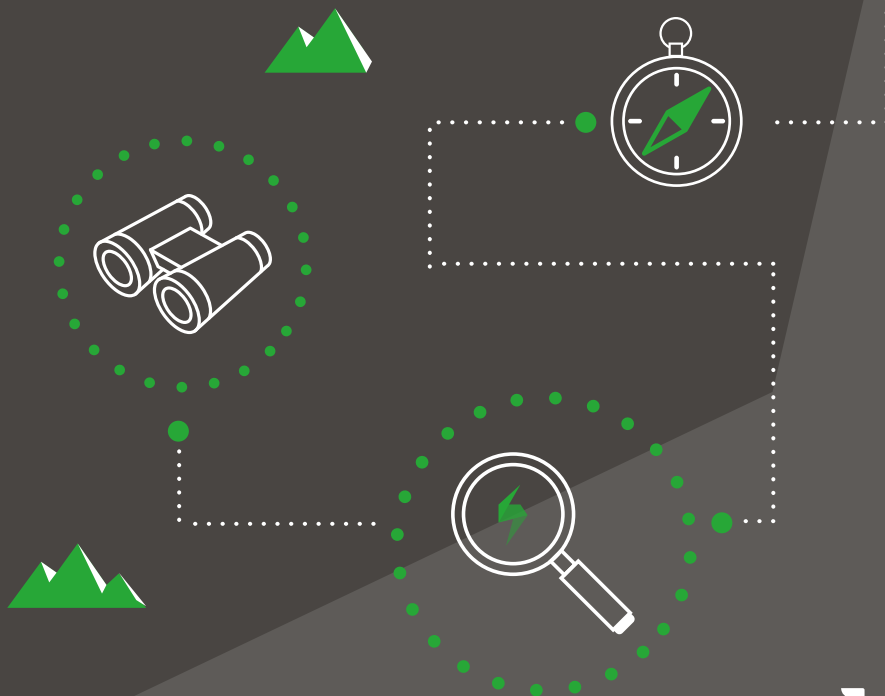




Uncover product risks with **Exploratory Testing**



Content

1. Introduction: Ensure thorough test coverage with exploratory testing

2. Scripted testing vs. exploratory testing

3. The 3 advantages of exploratory testing

Test your system from different angles to uncover unknown risks

Use the skills, expertise and knowledge of testers to intuitively test the system

Share insights with the whole team to uncover risks that you might not find alone

4. Introducing the Xray Exploratory App

5. How to effectively perform exploratory testing

6. Conclusion: Take your quality to new heights with exploratory testing

Introduction

Ensure thorough test coverage with exploratory testing

“Did you know that there are 100+ different types of product risks that can affect your software?”

Dan Ashby | Head of Quality Engineering at PhotoBox

With so many potential risks to your software, it's impossible that all of your scripted/pre-planned tests cover the entirety of these risks. These unknown risks can lead to unexpected software failures which **cost 40x more** to fix and repair later in the development cycle or in the hands of your customers.

One surefire way to ensure software meets the highest quality is to perform unscripted or exploratory testing in order to investigate the system for all potential threats.

Instead of only relying on executing a set of steps and validating them against predefined results, exploratory testing gives you freedom to decide what actions to perform during the session.

In this document, you'll learn the main differences between scripted and exploratory testing, 3 key advantages of exploratory testing and how to perform successful exploratory testing and uncover dangerous product risks.

“Exploratory testing is a style of software testing that encourages freedom to think and explore within the constraints of a specific goal. It aims to answer questions about risks and relies on documented and shareable observations to help teams make decisions.”

James Bach | Pioneer of Exploratory Testing and Creator of Rapid Software Testing

Scripted testing vs.
exploratory testing

2

Scripted testing vs. exploratory testing

A foolproof testing strategy combines several testing approaches to ensure software meets the highest quality. When you diversify your testing strategy and include exploratory testing, you'll test your product from all different angles and ensure maximum test coverage.

Keep in mind that exploratory testing follows an unstructured and unpredictable path and most likely produces different results each time. Each tester will perform different types of testing because of their unique experience, skill-set and knowledge of the system.

Don't mix up exploratory sessions with random or ad-hoc testing. Exploratory testing should have a scope as a means to provide focus and overview of the coverage (i.e. what you are targeting in that session).

Here are the main differences between scripted testing and exploratory testing:

Scripted Testing

Exploratory Testing

Predefined actions with corresponding expected results

Actions conditioned by how the system behaves and by the expertise and knowledge of the tester

Directed by the requirements

Directed by the insights discovered while testing

Predefined tests that will either pass or fail according to their result

Predefined goal/purpose but no predefined tests

Test cases are determined in advance

Test cases are determined during testing

Executed manually or automated using automation frameworks (i.e. Selenium/Webdriver/Cucumber)

Usually executed with a companion App, like Xray Exploratory App, that allows you to record sessions, take screenshots, notes, etc

Provides information about the expected (i.e. knowns)

Provides information about the unknowns

Structured, predictable and repeatable; if the test passes, it should produce an expected result

“Unstructured”, unpredictable, produces different insights every time

Focused on expected results

Focused on insights

As you can see, exploratory testing moves you from a passive role in testing to an active one. This allows you to use your skills, experience and knowledge of the system to test on-the-go and effectively understand the SUT (system under test).

3 advantages
of exploratory testing

3

3 advantages of exploratory testing

Exploratory testing helps test different scenarios using the unique expertise of testers and brings it back to the entire development team to uncover difficult to find errors and gaps in requirement coverage. When combined with an existing testing strategy, you get the most out of your testing efforts.



Test your system from different angles to uncover unknown risks

With exploratory testing you start with a purpose or “mission.” Your actions are conditioned by how the system behaves and by your expertise. You iteratively explore in order to learn more about the system. Results and insights build off of each other and ensure you test the system from all scenarios.

When you control the path of testing instead of simply executing what someone predefined for you, you’ll prevent gaps in test coverage which could result in major software failures and unaddressed product needs.

Use the skills, expertise and knowledge of testers to intuitively test the system

Nowadays, organizations believe that they should automate all of their testing. However, pre-scripted tests still leave important gaps in your testing.

Using the exploratory testing approach, you learn and iterate your testing as you test, growing your knowledge and skills along the way. So far, no machine can match the unique combination of skill, knowledge and experience that a real-life tester brings.

Automation can't



Predict
Suggest
Refine
Collaborate

Share insights with the whole team to uncover risks that you might not find alone

Share your insights from your exploratory testing sessions in order to add to the collective knowledge of everyone involved in the SDLC (software development life-cycle) and help make a better product, that comprehensively addresses the customer needs and does it without compromising product quality.

Gather rich evidence like videos, recordings, and notes and make it fully visible to your team. When you share your knowledge, you'll discover insights which you alone might not notice.

Introducing
the Xray
Exploratory App

4

Introducing the Xray Exploratory App

As you've seen, in order to reap the benefits of exploratory testing, it's important to share your insights and evidence with your team and use this collective knowledge to uncover common patterns, hidden errors and unknown code discrepancies.

However, with so much to document and record, it's likely you spend more time on documentation instead of focusing on the testing itself.

To help you effectively and efficiently perform exploratory testing, we built the Xray Exploratory App.

The [Xray Exploratory App](#) is an exploratory testing companion that assists you in gathering rich evidence during your exploratory sessions. Using the app you can record videos, make screenshots, do annotations and share your results with your team.

The app keeps you focused, eases the process of documentation and note-taking and lets you concentrate on the testing itself. You can run the app on your desktop and seamlessly integrate it with Xray and Jira to record the results within a test run.

You can [install the app](#) for free and start exploring **your systems.**

Read on to learn how to perform an Exploratory Testing session with the Xray Exploratory App.

How to effectively
perform exploratory
testing

5

How to effectively perform exploratory testing

To get started with exploratory testing, you want to define your objective. Test charters are a "framework" commonly used in exploratory testing to provide guidance on the exploratory testing session through a statement of objectives.

SBTM (Session Based Test Management) gives you a more structured approach to exploratory testing. SBTM identifies the purpose of the session and according to James Bach, the creator of this method, answers the questions **“what are we testing or what problems are we looking for?”**

Once you've identified your mission/purpose, you want to timebox your session and have the Xray Exploratory App ready.

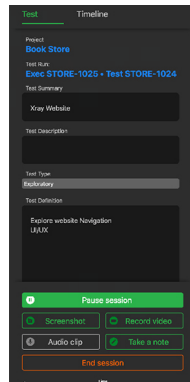
Follow these steps to perform an exploratory testing session with the Xray Exploratory App:

1. Create a generic/unstructured test in Xray
2. Link it to a story/requirement if you want to have coverage visibility
3. Plan/schedule that test as usual (e.g. add it to a Test Plan or to a Test Execution)

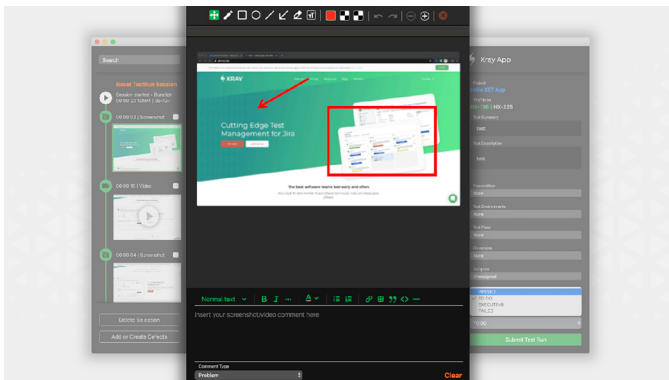
4. Execute with the Xray Exploratory Testing App (in the Test Execution issue screen, select "Execute with Exploratory App")

The Xray Exploratory App will launch and load the test data

5. Perform your exploratory testing session and use the Xray Exploratory App to assist you during the process
Create a new test session.
Take screenshots, record videos, add voicenotes, and take notes.



Use annotation tools. Draw and write on the screenshots, add comments to videos, and use wiki markup to edit notes.



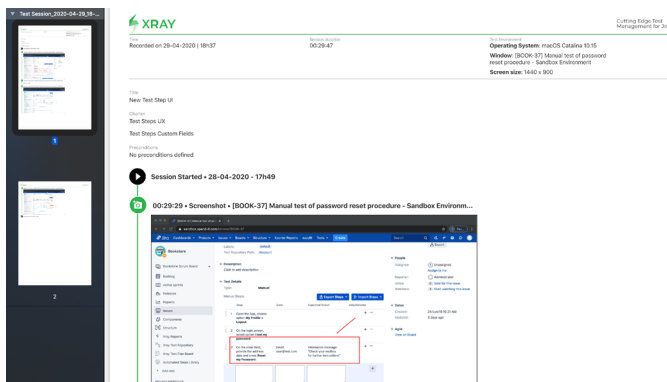
6. At the end of your session, report the status back in Xray. For example, if you consider if it was successful or not, you can assign it a status of "PASS" or "FAIL")

Create/edit issues, add/remove preconditions and select fix version.



All notes, evidence, and statuses are reported back to the "test run" entity in Xray.

7. When you're done - export your session to a PDF and save to your computer. You can edit the report names when exporting locally.



Conclusion

Take your quality to new heights with exploratory testing

Exploratory testing goes beyond scripted testing to fully uncover hidden bugs and product risks. With more than 100+ product risks that can affect your system, adding exploratory testing to your test strategy ensures your testing exceeds quality.

Exploratory testing keeps your testing relevant as you continuously learn about your system and discover new ways to test and explore. This continuous innovation and iteration keeps testing at the highest standards.

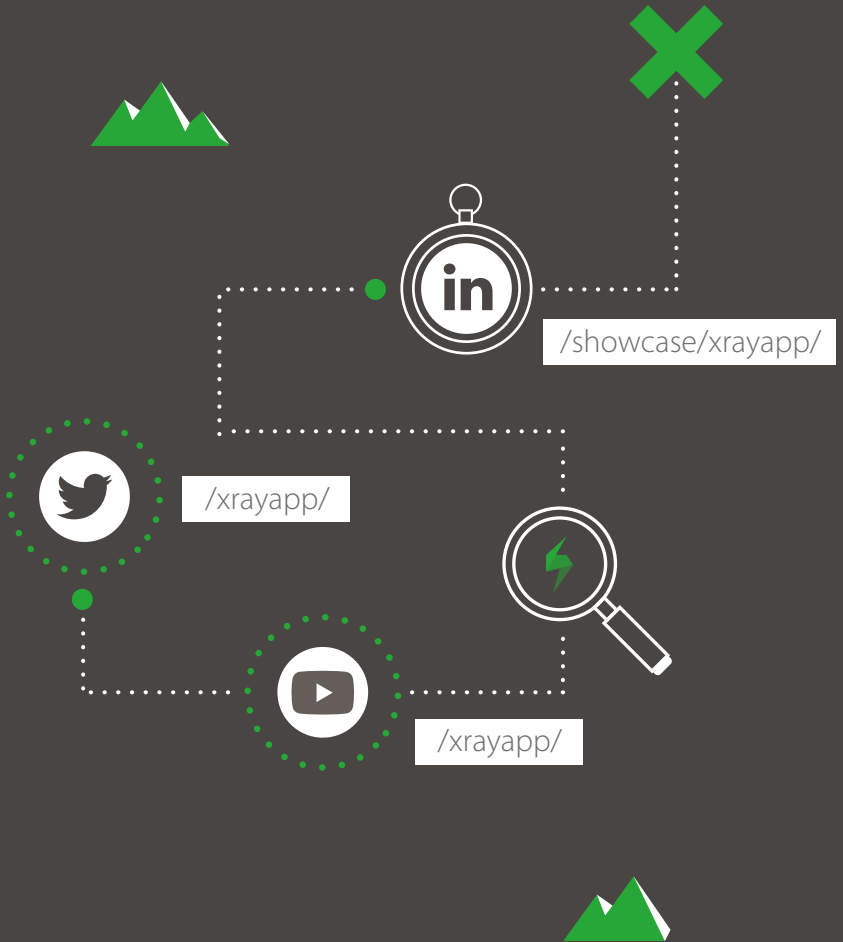
Even if exploratory testing feels like uncharted water, you can always experiment and see if it works for you. Investigate potential risks, explore the unknown and use your knowledge and skills as a tester to deeply understand and test your system.

Use the [Xray Exploratory App](#) to gather evidence and capture valuable insights, then bring it back to your team to collectively learn and uncover product risks.

Links



www.getxray.app





getxray.app