What is Exploratory Testing? Techniques with Examples

Testing Tutorials

Testing Methodology

End to End Testing

Exploratory Testing

System Integration Testing

Mutation Testing

What is Exploratory Testing?

EXPLORATORY TESTING is a type of software testing where Test cases are not created in advance but testers check system on the fly. They may note down ideas about what to test before test execution. The focus of exploratory testing is more on testing as a "thinking" activity.
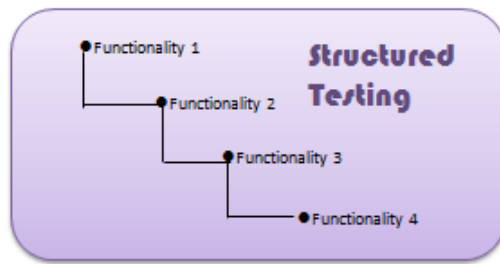
Exploratory Testing is widely used in Agile models and is all about discovery, investigation, and learning. It emphasizes personal freedom and responsibility of the individual tester.

Under scripted testing, you design test cases first and later proceed with test execution. On the contrary, exploratory testing is a simultaneous process of test design and test execution all done at the same time.

Scripted Test Execution is usually a non-thinking activity where testers execute the test steps and compare the actual results with expected results. Such test execution activity can be automated does not require many cognitive skills.

Structured Testing

- Functionality 1
- Functionality 2
- Functionality 3
- Functionality 4

Functionalities are checked in a structured manner

Exploratory Testing

- Functionality 1
- Functionality 4
- Functionality 2
- Functionality 3
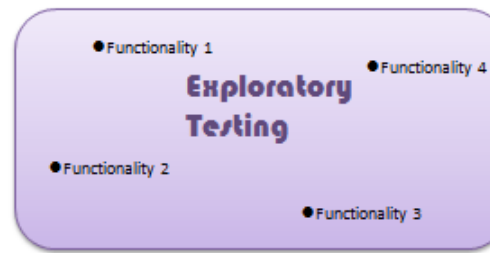
Functionalities are checked in a ad-hoc manner

Though the current trend in software testing is to push for automation, exploratory testing is a new way of thinking. **Automation has its limits**

It can't THINK!!

predict

suggest

Automation can't

refine

collaborate
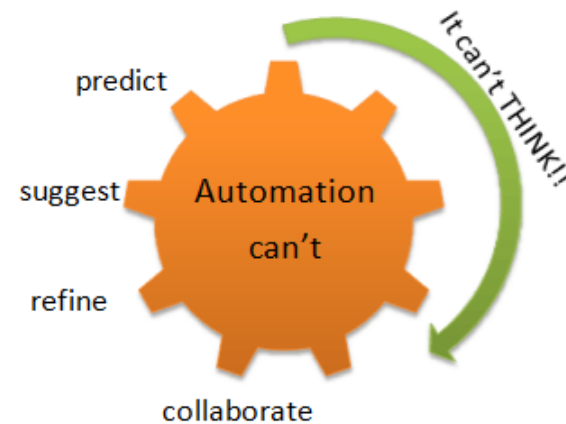
Differences between Scripted and Exploratory Testing

Scripted Testing	Exploratory Testing
Directed from requirements	Directed from requirements and exploring during testing
Determination of test cases well in advance	Determination of test cases during testing
Confirmation of testing with the	Investigation of system or application

requirements	
Emphasizes prediction and decision making	Emphasizes adaptability and learning
Involves confirmed testing	Involves Investigation
Is about Controlling tests	Is about Improvement of test design
Like making a speech - you read from a draft	Like making a conversation - it's spontaneous
The script is in control	The tester's mind is in control

Exploratory testing -

- Is not random testing but it is ad-hoc testing with a purpose of find bugs
- Is structured and rigorous
- Is cognitively (thinking) structured as compared to the procedural structure of scripted testing. This structure comes from Charter, time boxing etc.
- Is highly teachable and manageable
- It is not a technique but it is an approach. What actions you perform next is governed by what you are doing currently

How to do Exploratory Testing

Exploratory test preparation goes through following 5 stages detailed below and it is also called session based test management (SBTM Cycle):

1. **Create a Bug Taxonomy (classification)**
   - Categorize common types of faults found in the past projects
   - Analyze the root cause analysis of the problems or faults
   - Find the risks and develop ideas to test the application.
2. **Test Charter**
   - Test Charter should suggest

1. what to test
2. how it can be tested
3. What needs to be looked
   - Test ideas are the starting point of exploration testing
   - Test charter helps determine how the end user could use the system

3. **Time Box**
   - This method includes a pair of testers working together not less than 90 minutes
   - There should not be any interrupted time in those 90 minutes session
   - Timebox can be extended or reduced by 45 minutes
   - This session encourages testers to react on the response from the system and prepare for the correct outcome

4. **Review Results:**
   - Evaluation of the defects
   - Learning from the testing
   - Analysis of coverage areas

5. **Debriefing:**
   - Compilation of the output results
   - Compare the results with the charter
   - Check whether any additional testing is needed

For Example, during exploratory execution, the following needs to be done:

- The mission of testing should be very clear
- Keep notes on what needs to be tested, why it needs to be tested and the assessment of the product quality
- Tracking of questions and issues raised during exploratory testing
- Better to pair up the testers for effective testing
- The more we test, more likely to execute right test cases for the required scenarios

It is very important to take a document and monitor the following

- Test Coverage - Whether we have taken notes on the coverage of test cases and improve the quality of the software
- Risks - Which risks need to be covered and which are all important ones?
- Test Execution Log - Recordings on the test execution
- Issues / Queries - Take notes on the question and issues on the system

Smarter exploratory testing finds more errors in less time.

Pros and Cons of Exploratory Testing

Pro	Con
- This testing is useful when requirement documents are not available or partially available - It involves Investigation process which helps find more bugs than normal testing- - Uncover bugs which are normally ignored by other testing techniques - Helps to expand the imagination of testers by executing more and more test cases which finally improves productivity as well - This testing drill down to the smallest part of an application and covers all the requirements - This testing covers all the types of testing and it covers various scenarios and cases - Encourages creativity and intuition - Generation of new ideas during test execution	- This testing purely depends on the tester skills - Limited by domain knowledge of the tester - Not suitable for Long execution time

Challenges of Exploratory Testing:

There are many challenges of exploratory testing and those are explained below:

- Learning to use the application or software system is a challenge
- Replication of failure is difficult
- Determining whether tools need to be used can be challenging
- Determine the best test cases to execute can be difficult
- Reporting of the test results is a challenge as the report doesn't have planned scripts or cases to compare with the actual result or outcome
- Documentation of all events during execution is difficult to record
- Don't know when to stop the testing as exploratory testing has definite test cases to execute.

When use exploratory testing?

Exploratory testing can be used extensively when

- The testing team has experienced testers
- Early iteration is required
- There is a critical application
- New testers entered into the team

Conclusion:

In Software Engineering, Exploratory testing is performed to overcome the limitations of scripted testing. It helps in improving Test Case suite. It empathizes on learning and adaptability.

YOU MIGHT LIKE:

COURSE

Agile Testing Tutorial

Training Summary Testing using Agile Methodology is the buzzword in the industry as it yields quick...
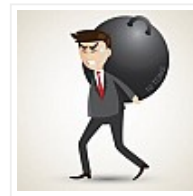
Read more »

SOFTWARE TESTING

What is SDET? Full Form, Meaning, Role and Responsibilities

What is an SDET? SDET is an IT professional who can work equally effectively in development and...

Read more »

SOFTWARE TESTING

What is Scalability Testing? Learn with Example

What is Scalability testing? SCALABILITY TESTING is a non functional testing, that measures...

Read more »

COURSE

TEST Management Tutorials: Complete Training Course

Course Summary Test Management is a series of planning, execution, monitoring and control...

Read more »

SOFTWARE TESTING

Sanity Testing Vs Smoke Testing: Introduction & Differences

Smoke and Sanity testing are the most misunderstood topics in Software Testing. There is an...

Read more »

SOFTWARE TESTING

20 Best Bug/Defect Tracking Tools In 2020

{loadposition top-ads-automation-testing-tools} You can put this in another way "Better is the bug...
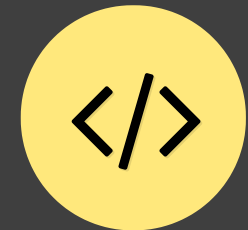
Read more »

Top Tutorials

TestProject

Testing

Hacking

SAP

Java

Python

Jmeter

Informatica

JIRA