Chapter 9

Regression Testing

Overview

- Regression Testing
- When to do regression testing
- Regression Testing Cycle
- Advantages and Disadvantages of Regression Testing
- Challenges of Regression Testing
- Types of regression testing
- Types of regression testing process
- Process of Conducting Regression Testing

Regression Testing

Def: Regression testing is a test to make sure that the product works fine with new functionality, bug fixes, or any change in the existing feature.

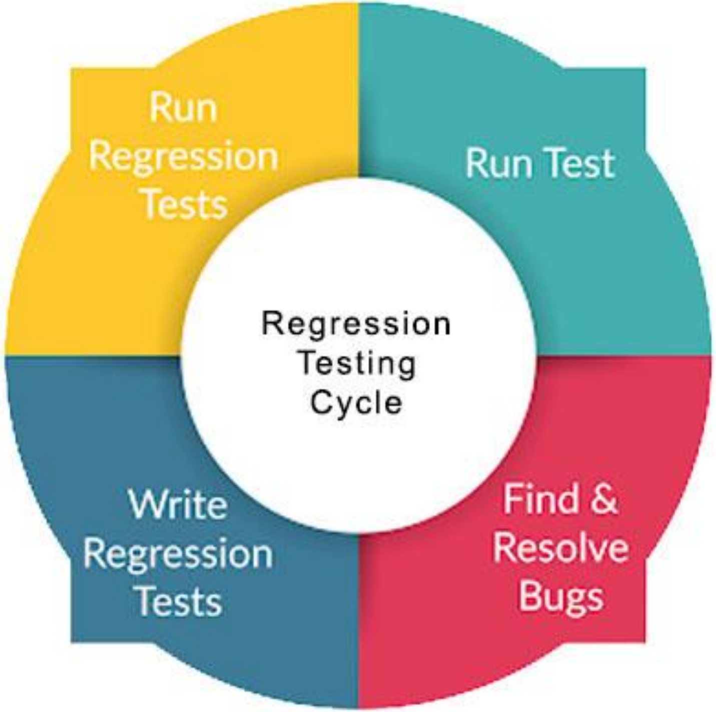
Regression testing is a black box testing techniques.

Regression testing can be performed on a new build when there is a significant change in the original functionality. It ensures that the code still works even when the changes are occurring. Regression means Re-test those parts of the application, which are unchanged.

When to do Regression Testing

- If new functionality added to the application
- If there is a Change Requirement.
- If any defect fixed
- If there is a performance issue fix
- If there is an environment change

Regression Testing Cycle

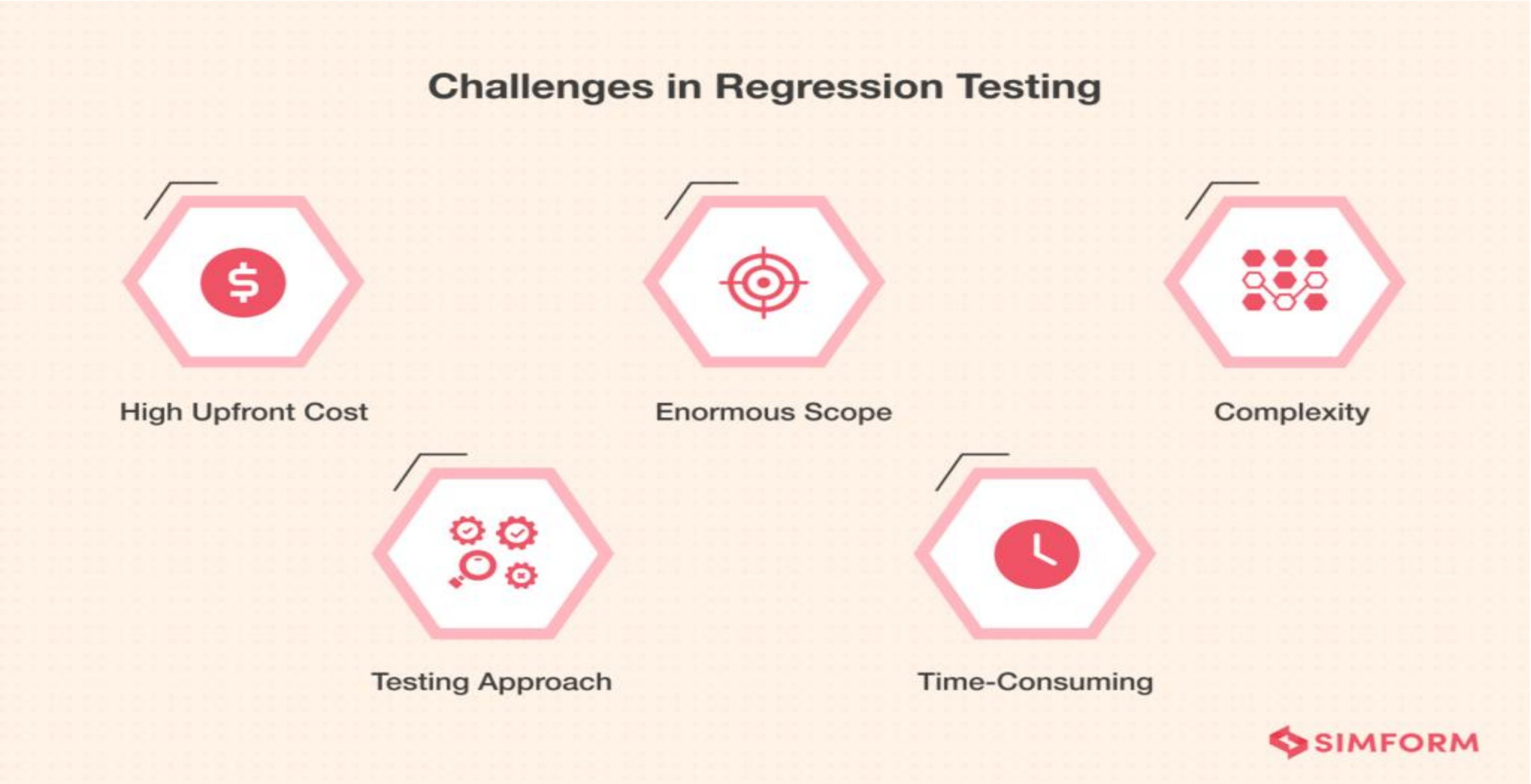


Advantages of Regression Testing

- Increases the chance of detecting bugs at early stages
- Helps identifying unwanted side effects and errors due modification
- Ensures no errors occurs again by retesting (selective or whole)

Disadvantages of Regression Testing

- Without automation, regression testing can be time-consuming.
- Need to perform for every small change of code.
- A highly repetitive process may affect the agile sprint.
- Requires you to create complex test cases



Challenges in Regression Testing

High Upfront Cost

Enormous Scope

Complexity

Testing Approach

Time-Consuming

SIMFORM

Challenges in Regression Testing

- **High Upfront Cost:** Regression testing is very time-consuming if done manually. That's why automation takes preference, and for that to happen, you require high-end infrastructure and tools, which can be highly expensive.
- **Testing Approach:** Selecting a proper testing approach is crucial to success as far as regression testing goes. You may have skilled resources and tools at your disposal, but you will struggle if you don't have a proper plan for when and how to approach or conduct regression testing.
- **Enormous Scope and Coverage:** The success of regression testing depends on the type of test cases suite you have built. Between every build and release, you need to update the regression suite created. Now, remembering each change would be hard for testers, and regression testing is impossible without that. That's the only way to cover all the elements. Thus, the enormous scope is a massive challenge.
- **Complexity:** As you move from the first build to the second and then so on, the number of test cases increases. That's why the regression test suite will become more and more complex. You will have to test the new cases as well as the old cases all the time.

Types of Regression Testing

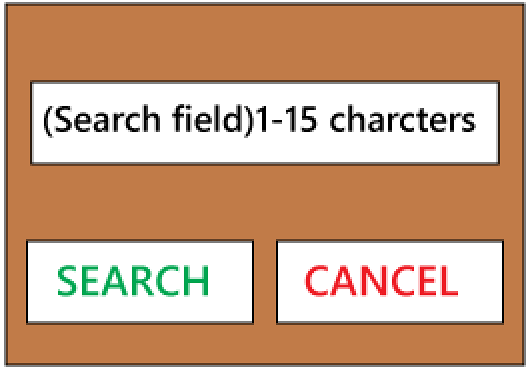
The different types of Regression Testing are as follows:

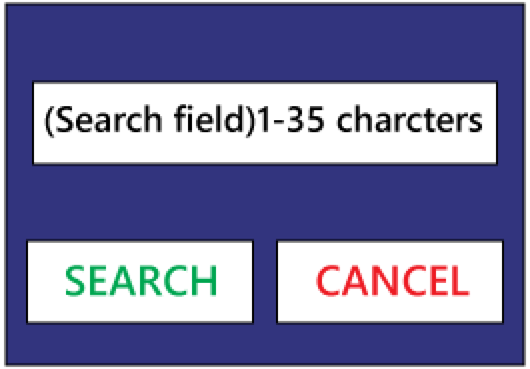
- Unit Regression Testing [URT]
- Regional Regression Testing[RRT]
- Full or Complete Regression Testing [FRT]

Unit Regression Testing

In this, we are going to test only the changed unit, not the impact area, because it may affect the components of the same module.

Example: Client does some modification in the requirement and also requests that the **Search button** can accept the **1-35 characters** instead of **1-15 characters**. The test engineer will test only the Search button to verify that it takes 1-35 characters.



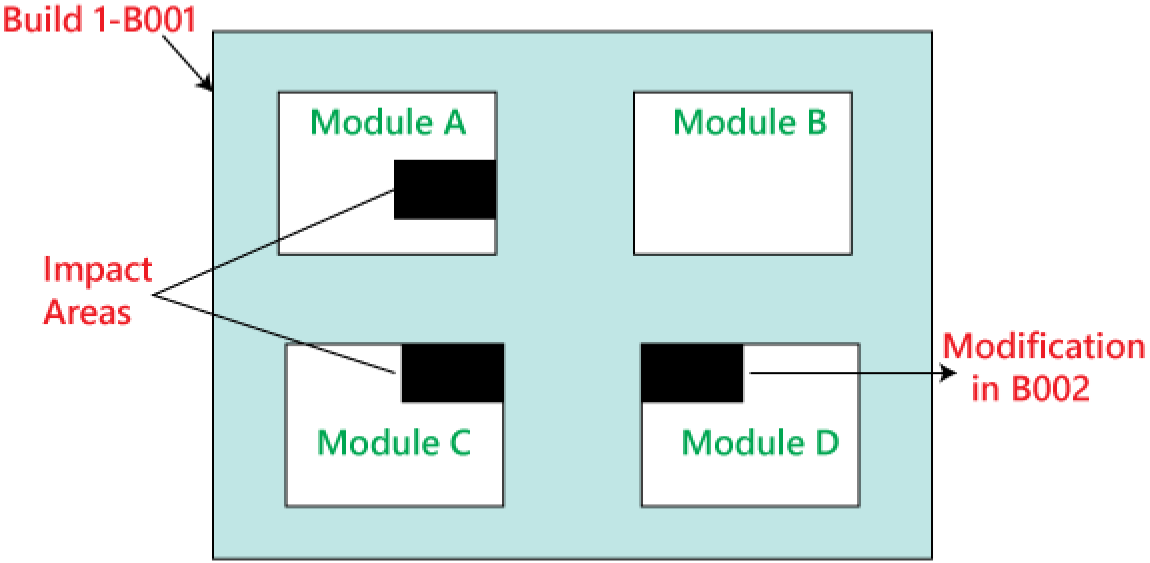


Regional Regression testing [RRT]

To test the modification along with the impact area or regions, are called the **Regional Regression testing**. Here, we are testing the impact area because if there are dependable modules, it will affect the other modules also.

Example:
The bug fixing in Module D has impacted some features in **Module A and C**.

Hence, the test engineer first tests the Module D where the bug has been fixed. Then checks the impact areas in **Module A and Module C**.



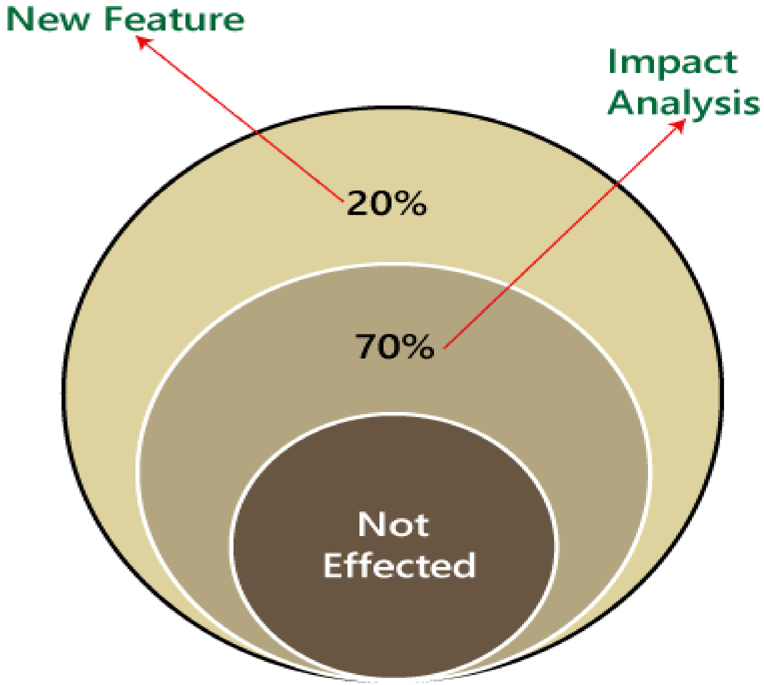
Disadvantages of IRT and RRT :

Following are some of the drawbacks of using Unit and Regional regression testing:

- We may miss some impact area.
- It is possible that we may identify the wrong impact area.

Full Regression testing [FRT]

Testing the **modified** **features** and **all the remaining (old) features** is called the **Full Regression testing**.



Full Regression testing [FRT]

When the application is given for the testing, the test engineer will test the first 10-14 cycle, and will do the **RRT**. Then for the 15th cycle, we do FRT. And again, for the next 10-15 cycle, we do **Regional regression testing**, and for the 31th cycle, we do the **full regression testing**, and we will continue like this.

The drawback of doing regression testing repeatedly:

- Productivity will decrease.
- It is a difficult job to do.
- There is no consistency in test execution.
- Test execution time is also increased.

Types of Regression Testing Process

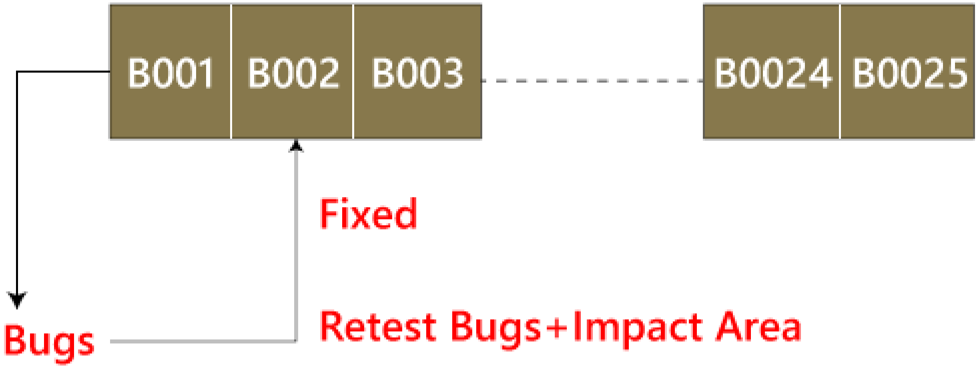
The regression testing process can be performed across the

- **Builds** and
- The **Releases**.

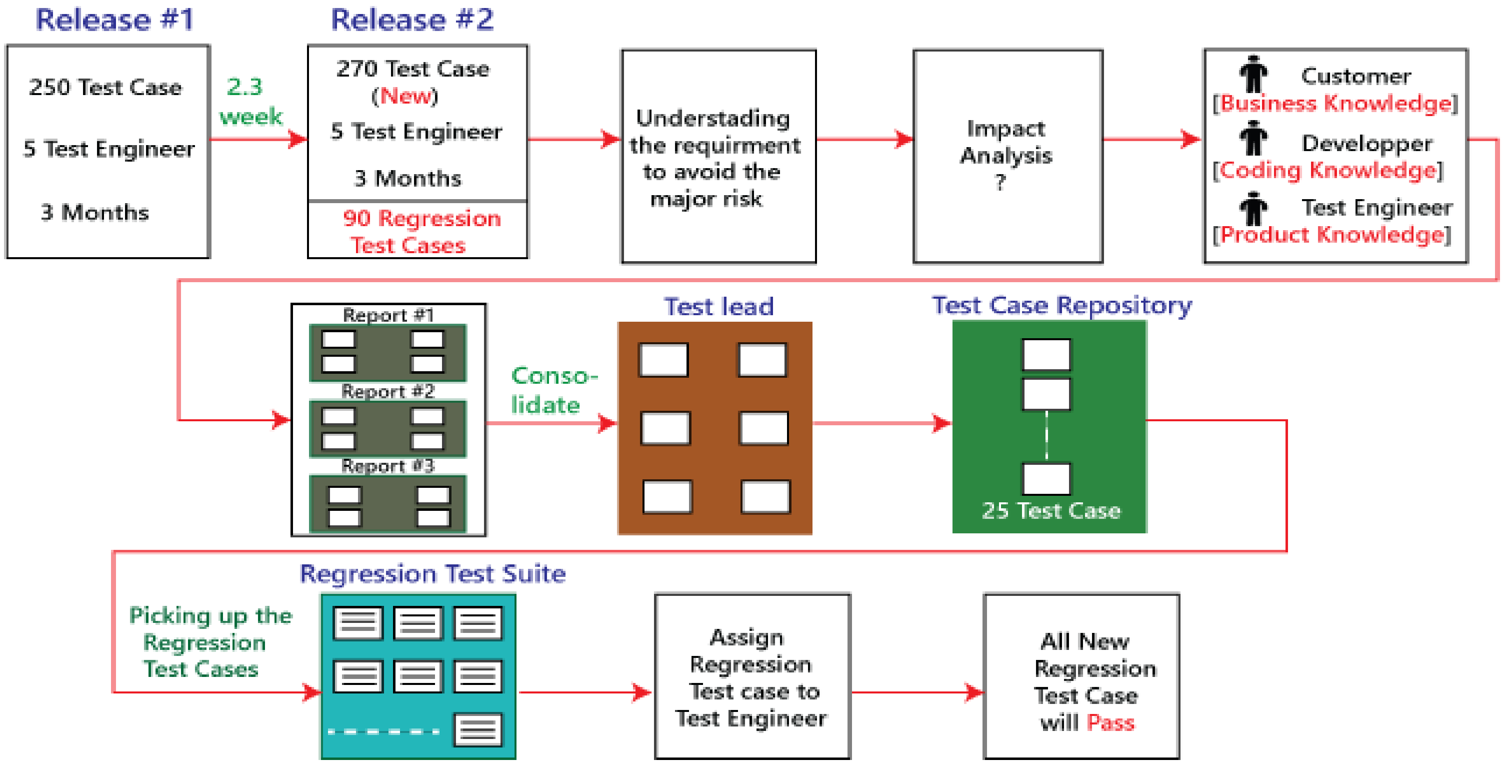
The main difference between Build and Release in Software Testing is that **Build is a version of the software the development team hands over to the testing team for testing purpose** while **Release is the software the testing team hands over to the customer**.

Regression testing across the builds

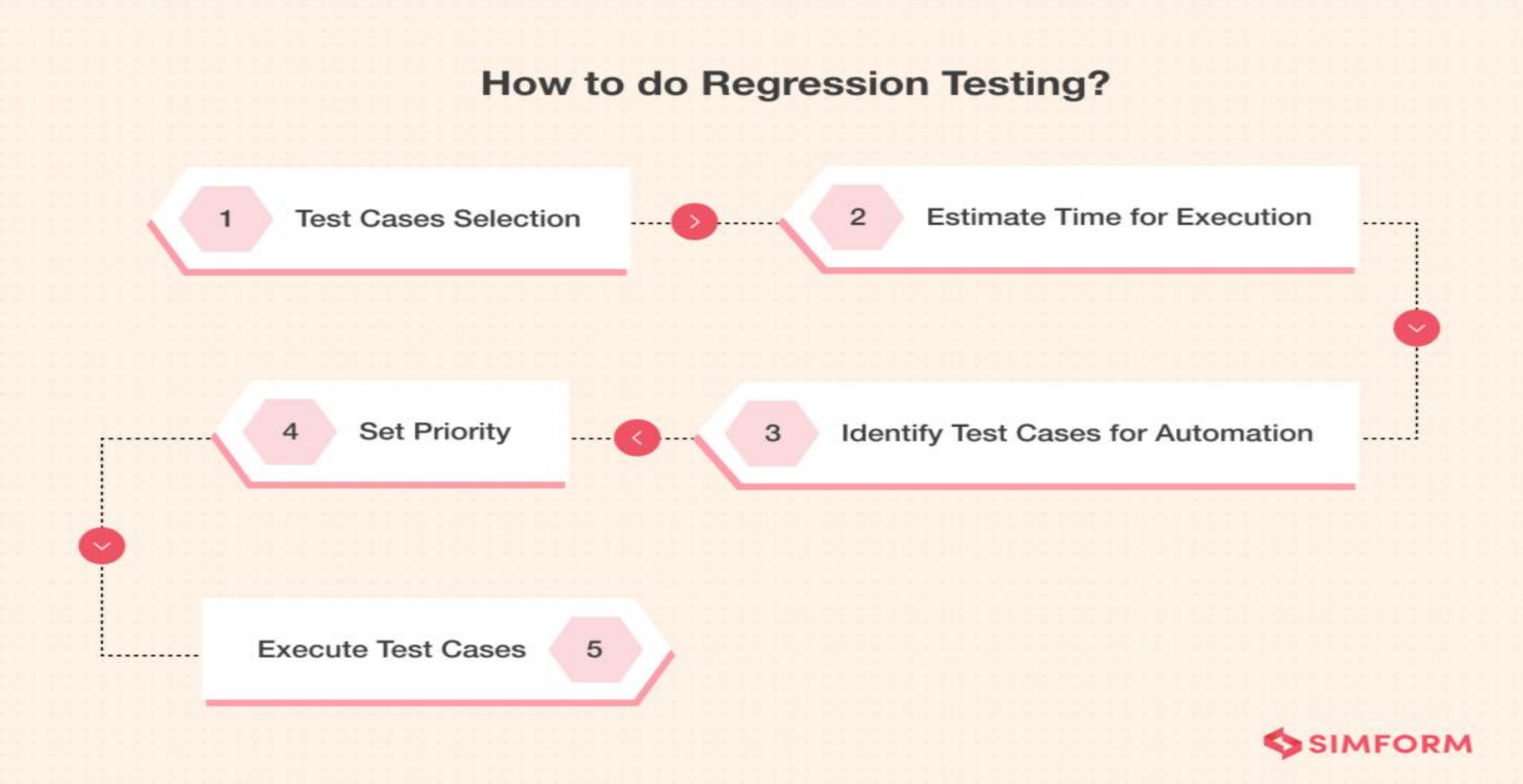
Whenever the bug fixed, we retest the Bug, and if there is any dependent module, we go for a Regression Testing.



Regression Test across Releases



Release #1
250 Test Case
5 Test Engineer
3 Months
2.3 week
Release #2
270 Test Case (New)
5 Test Engineer
3 Months
90 Regression Test Cases
Understading the requirment to avoid the major risk
Impact Analysis ?
Customer [Business Knowledge]
Developper [Coding Knowledge]
Test Engineer [Product Knowledge]
Report #1
Report #2
Report #3
Conso-lidate
Test lead
Test Case Repository
25 Test Case
Picking up the Regression Test Cases
Regression Test Suite
Assign Regression Test case to Test Engineer
All New Regression Test Case will Pass



How to do Regression Testing?

1 Test Cases Selection

2 Estimate Time for Execution

3 Identify Test Cases for Automation

4 Set Priority

5 Execute Test Cases

SIMFORM

Process of Conducting Regression Testing

- **Step 1: Regression Test Selection:**
  Selection of test cases will depend on the module where there is a change in the source code. Then divide these test cases into (i) Reusable Test Cases and (ii) Obsolete Test Cases. Only reusable ones will be used for future regression cycles.

  Here are the tricks to identify the reusable test cases:
  - highly error-prone
  - verifies the functionality of application
  - contain features that are visible to users
  - have a recent change in the source code
  - related to integration
  - related to boundary-value

- **Step 2: Determine the Time for Executing Test Cases**
  To determine the time, factors that affect the execution time are considered. These are test data creation, regression test planning by the QA team, review of all test cases, etc

Process of Conducting Regression Testing

- **Step 3: Identify the Test Cases that can be Automated:**
  QA team can decide the test cases that they can automate. Automated test cases are faster than manual testing and allows to re-use the same script repeatedly.
- **Step 4: Test Cases Prioritization**
  After gathering all the test cases and prioritize, those should be prioritized, i.e., high, medium, and low.
  The priority will depend on the product’s functionality and user involvement.

  Below is the waterfall structure of priorities:
  - Priority 0: The most critical test cases include all the core functionalities.
  - Priority 1: Essential features, but not a part of core functionality.
  - Priority 2: Test cases related to technical debt.

Examples of technical debt are Cut and pasted or duplicate code, Untested code. Unreadable code, Out of date documentation, Dead code (code that never runs), Spaghetti architecture.

Process of Conducting Regression Testing

- **Step 5: Executing Test Cases:**
  Finally all the test cases are executed one by one to see whether the product is working as it should or not. Both manual and automated testing can be done based on the requirement. For automated regression testing, utilizing tools like Selenium, QTP, Watir, etc., allows you to execute the test cases faster.