

What is Risk Based Testing in Software testing?

Today's article is going to be **a complete guide to learn Risk Based Testing in Software testing**. Before explaining Risk based testing, it is necessary to know what mean by Risk in software testing. A Risk is a problem or situation that has not happened yet and it may never happen in future as well. It is basically a possible problem.

In this article we understand what is risk based testing, reasons & situations to implement risk based testing, and benefits of risk based testing.

Risk based testing is type of software testing that the features and functions to be tested based of priority, importance and potential failures. First we identify the risk to the project, we analyze the risk associated with the potential cost of the projects.

Let come to the point why and what all reasons and scenarios to implement in the risk based testing. Many projects have different constraints like time, resources, quality requirement in terms of organization standards. The Risk based testing works really well in this regards. While implementing new projects there are high risk factors involved like New technology, Lack of knowledge, lack of experience etc.

Major objective of Risk Based Testing:

- To identify when and how to use Risk Based Testing.
- To understand advantages and disadvantages of Risk Based Testing.
- To understand steps of implementing of Risk Based Testing on appropriate application.
- To make risk-free project using best practices in risk management to achieve a project outcome that balances risks with quality, features, budget and schedule.

Overview

Risk Based Testing (RBT) is a testing process with unique features. It is basically for those project and application that is based on risk. Using risk, Risk based testing prioritize and emphasize the suitable tests at the time of test execution. In other word, Risk is the chance of event of an unwanted outcome. This unwanted outcome is also related with an impact. Some time it is difficult to test all functionality of the application or it is might not possible. Use Risk based testing in that case; it tests the functionality which has the highest impact and probability of failure.

It's better to start risk based testing with product risk analysis. There are numerous methods used for this are,

- Clear understanding of software requirements specification, design documents and other documents.

- Brainstorming with the project stakeholders.

Risk-based testing is the process to understand testing efforts in a way that reduces the remaining level of product risk when the system is developed,

- Risk-based testing applied to the project at very initial level, identifies risks of the project that expose the quality of the project, this knowledge guides to testing planning, specification, preparation and execution.
- Risk-based testing includes both mitigation (testing to give chances to decrease the likelihood of faults, specially high-impact faults) and contingency (testing to know work-around to create the defects that do get past us less painful).
- Risk-based testing also includes measurement process that recognizes how well we are working at finding and removing faults in key areas.
- Risk-based testing also uses risk analysis to recognize proactive chances to take out or avoid defects through non-testing activities and to help us select which test activities to perform.



Major processes to execute the Risk-based testing are described below:

- Process 1 – Describe all requirements in terms of Risk involved in the project
- Process 2 – In terms of risk assessment, prioritize the requirements
- Process 3 – Plan and define tests according to requirement prioritization
- Process 4 – Execute test according to prioritization and acceptance criteria.

Process 1 – Projects associates various risks that are identified by the stakeholders of the project. The stake holders are basically a mixture of business and technical team. Stake holders involve various people from various departments for example, the client, customers, business experts, technical experts, project manager, project Leader, users, developers and infrastructure representative.

Process 2 – Once all the possible risks and their impacts are analyzed, the project manager has to get the requirements prioritized. Priority of the requirements should be agreed upon and should be updated in functional requirement document; the same should also be conveyed to the development and the test team.

Process 3 – After getting the Requirement with the priority tagged, we can start the test activities with keeping the priority of the requirements in mind.

Process 4 – If any of the identified risk realizes by the time of “Test execution Schedule”, then there is quite good chances of schedule slippage from development side. In this case, the final deadline given to the customer can’t be changed. In this situation, again, Test manager has to apply the Pareto principle and finalizes the scope of testing in the reduced timeline that will ensure least risk and highest quality.

Advantages/Benefits of Risk-based testing:

- Improved quality – All of the critical functions of the application are tested. Real time clear understanding of project risk.
- Give more focus on risks of the business project instead of the functionality of the information system.
- Provides a negotiating instrument to client and test manager similar when existing means are limited.
- Associate the product risk to the requirement identifies gaps.
- During testing, test reporting always takes place in a language (risks) that all stake-holder understands.
- Testing always concentrate on the most important matters first with optimal test delivery, in case of – limited time, money and qualified resources. With the time and resources we have, we just can able to complete 100% testing, so we need to determine a better way to accelerate our testing effort with still managing the risk of the application under test. Efforts are not wasted on non-critical or low risk functions.

- Improve customer satisfaction – Due to customer involvement and good reporting and progress tracking.

Some Helpful Test Techniques to proceed Risk-based testing

- Path Flow testing
- Some amount of exploratory / Experience based testing
- Boundary Value analysis
- Equivalence partitioning
- Decision tables

Source: <http://www.softwaretestingclass.com/what-is-risk-based-testing-in-software-testing/>