Test Management

- **The Goal of Test Planning:**

1. The software test plan is the primary means by which software testers communicate to the product development team what they intend to do.

2. The test plan is simply a by-product of the detailed planning process that's undertaken to create it. It's the planning process that matters, not the resulting document.

- A **Test Plan** is a detailed document that describes the test strategy, objectives, schedule, estimation, deliverables, and resources required to perform testing for a software product.

- Test Plan helps us determine the effort needed to validate the quality of the application under test.

- The test plan serves as a blueprint to conduct software testing activities as a defined process, which is minutely monitored and controlled by the test manager.

How to write a Test Plan

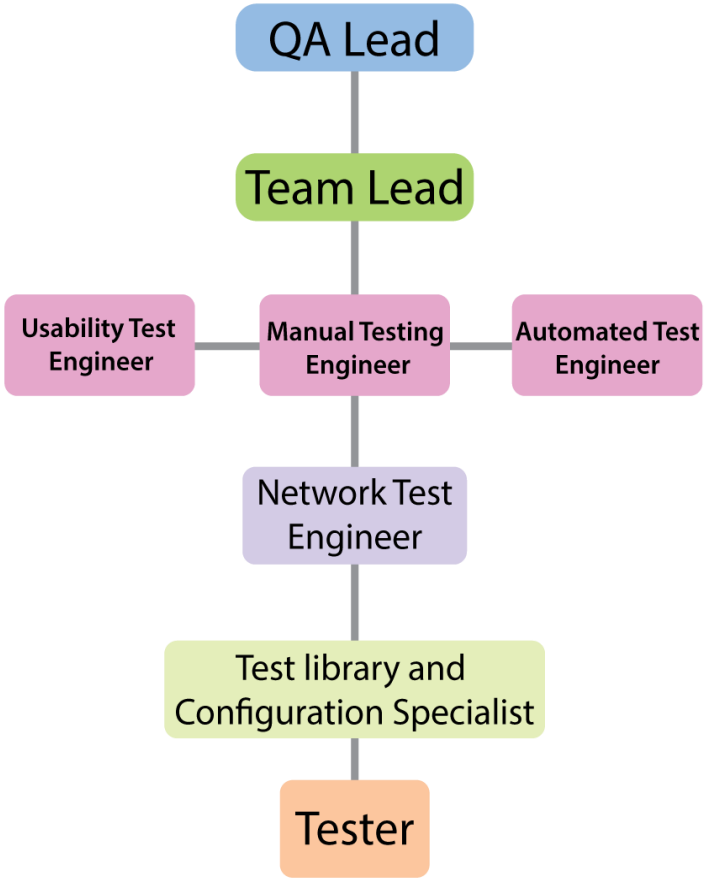
- Follow the seven steps below to create a test plan as per IEEE 829

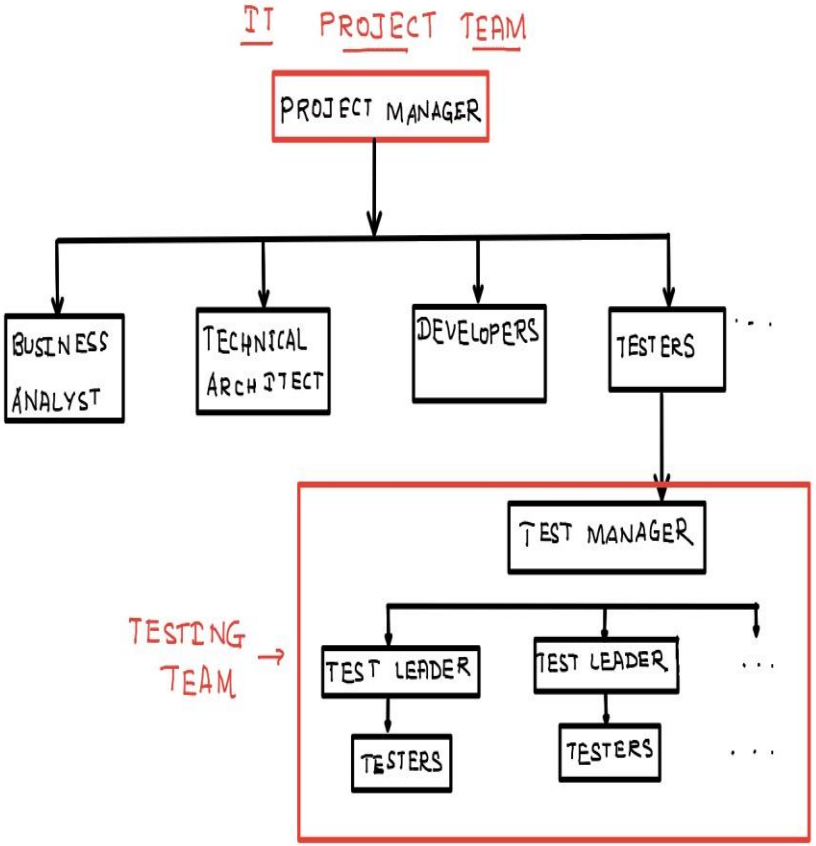
1. Analyze the product
2. Design the Test Strategy
3. Define the Test Objectives
4. Define Test Criteria
5. Resource Planning
6. Plan Test Environment
7. Schedule & Estimation
8. Determine Test Deliverables

Test Plan

1. Test plan identifier
2. Introduction
3. Test items
4. Features to be tested
5. Features not to be tested
6. Approach
7. Item pass/fail criteria
8. Test deliverables
9. Testing tasks
10. Environmental needs
11. Responsibilities
12. Staffing and training needs
13. Schedule
14. Risks and Mitigation
15. Approvals

Peoples


QA Lead
Team Lead
Usability Test Engineer
Manual Testing Engineer
Automated Test Engineer
Network Test Engineer
Test library and Configuration Specialist
Tester


IT PROJECT TEAM
PROJECT MANAGER
BUSINESS ANALYST
TECHNICAL ARCHITECT
DEVELOPERS
TESTERS
TEST MANAGER
TESTING TEAM
TEST LEADER
TEST LEADER
TESTERS
TESTERS

- A **TEST CASE** is a set of actions executed to verify a particular feature or functionality of your software application.

- A Test Case contains test steps, test data, precondition, postcondition developed for specific test scenario to verify any requirement.

Typical Test Case Parameters:

- Test Case ID
- Test Scenario
- Test Case Description
- Test Steps
- Prerequisite
- Test Data
- Expected Result
- Test Parameters
- Actual Result
- Environment Information
- Comments

Test Exit Criteria

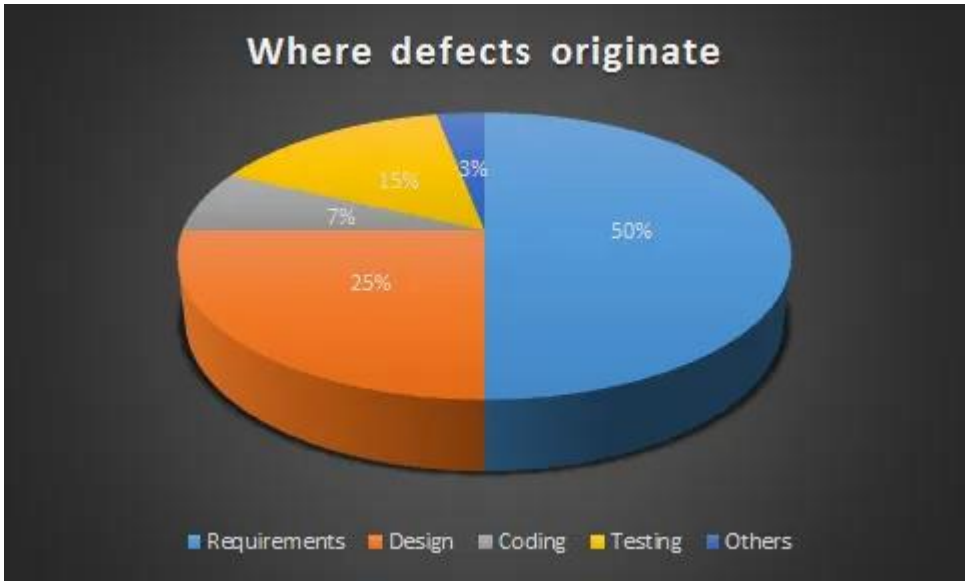
- Exit criterion is used to determine whether a given test activity has been completed or NOT. Exit criteria can be defined for all of the test activities right from planning, specification and execution.
- Exit criterion should be part of test plan and decided in the planning stage.

- Examples of Exit Criteria:

1. Verify if All tests planned have been run.
2. Verify if the level of requirement coverage has been met.
3. Verify if there are NO Critical or high severity defects that are left outstanding.
4. Verify if all high risk areas are completely tested.
5. Verify if software development activities are completed within the projected cost.
6. Verify if software development activities are completed within the projected timelines.

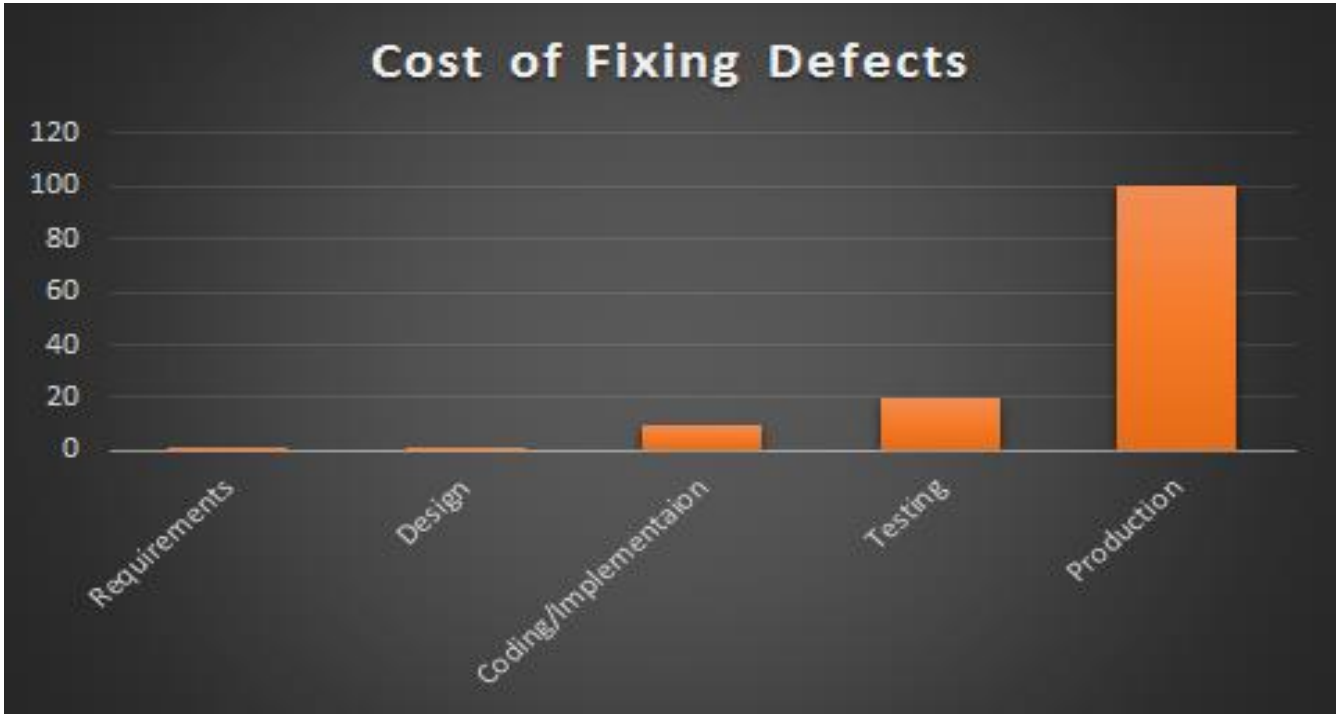
Economic impact of software testing.

- There is a definite economic impact of software testing. One economic impact is from the cost of defects and the another is the way we perform testing. Cost of defect is a very real and very tangible cost.



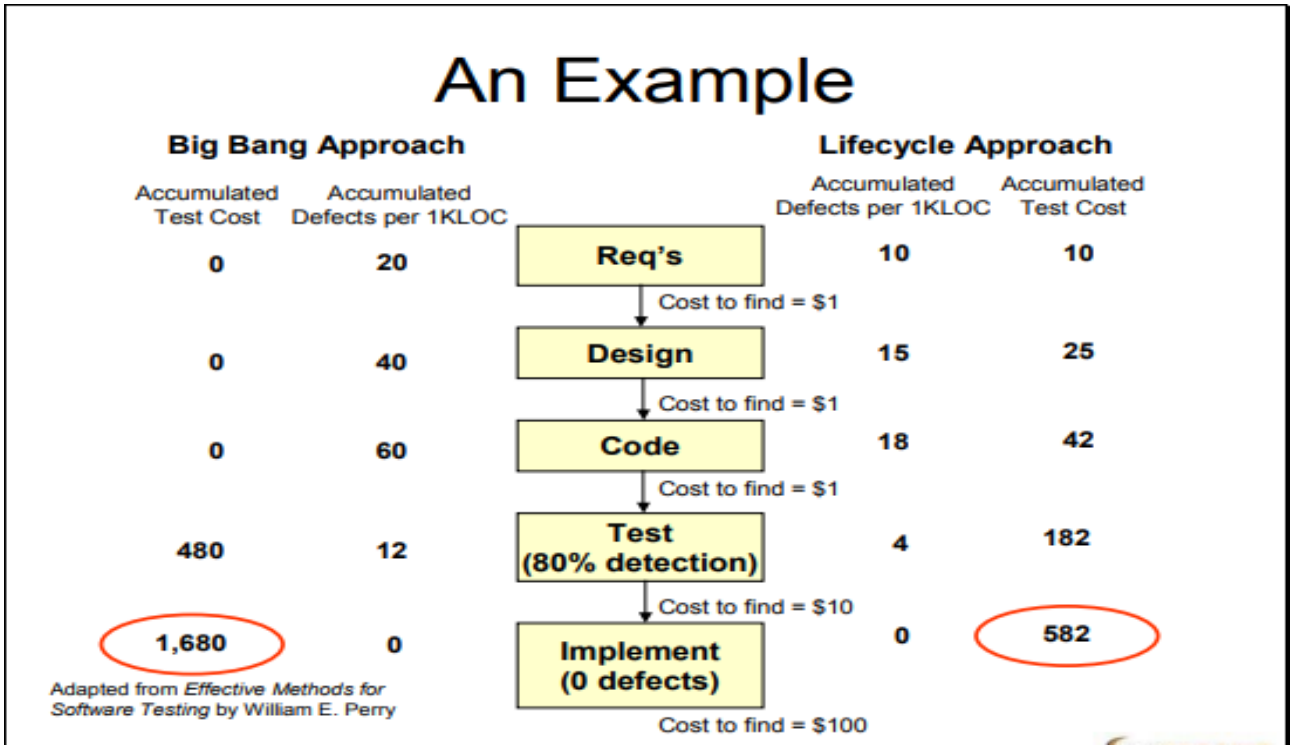
Relative cost of fixing defects

- One of the known facts about software defects is that the longer they go undetected, the more expensive they are to fix. Although research differs on the exact ratios, the general rule is 1:10:100.



- Most defects are created in the early stages of a project
- Most defects are found in the later stages of a project
- It costs 10 to 100 times as much to fix a defect in the later phases of a project.

- Two contrasting views of testing – the lifecycle approach where testing is performed throughout the project, and the big bang approach where testing is performed only at the end of the project. The final cost of the big bang approach is much greater than the lifecycle approach.



Test Approach

- A test approach is the test strategy implementation of a project, defines how testing would be carried out. Test approach has two techniques:

1. **Proactive -** An approach in which the test design process is initiated as early as possible in order to find and fix the defects before the build is created.
2. **Reactive -** An approach in which the testing is not started until after design and coding are completed.

Risk based testing

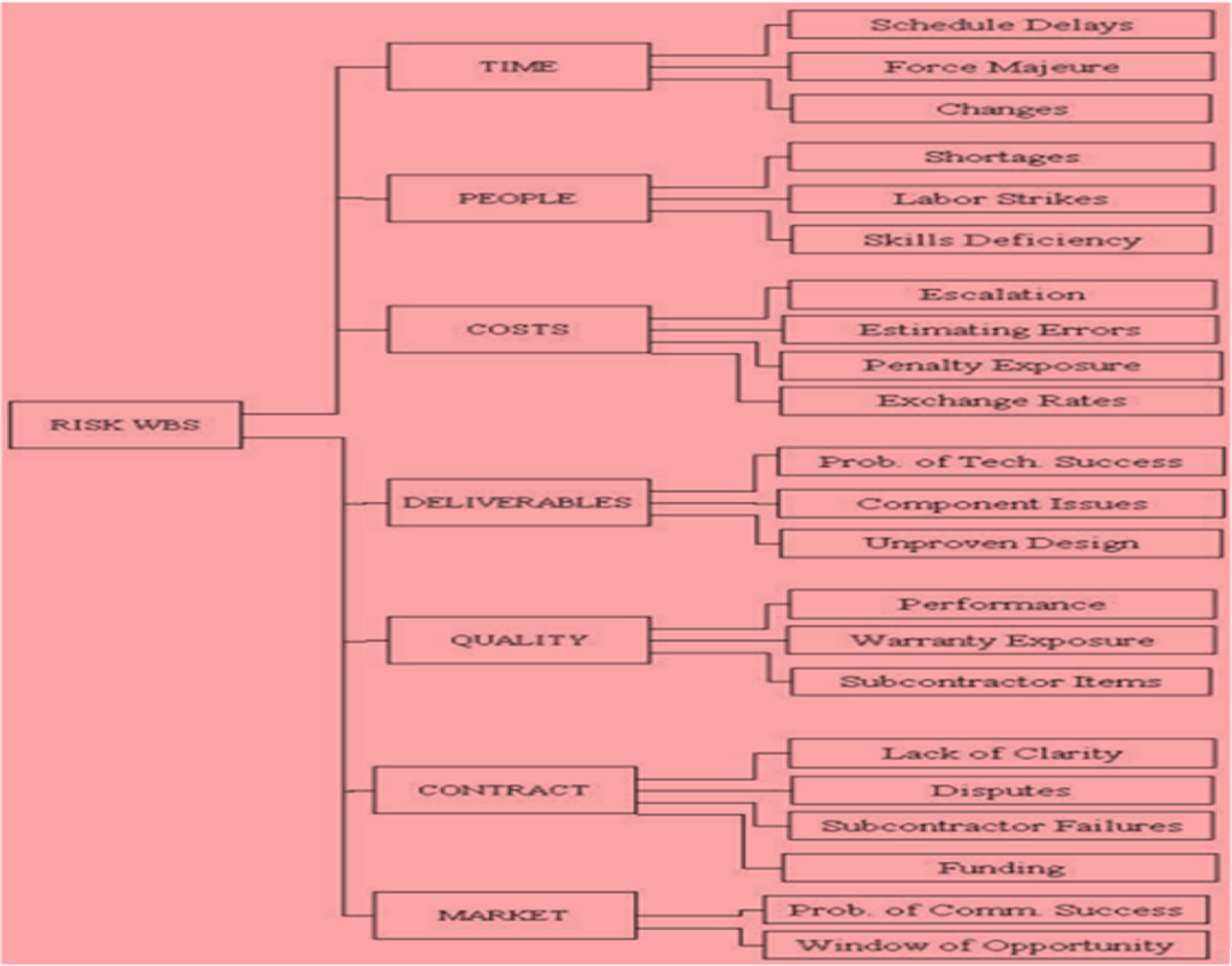
- Risk is the probability of occurrence of an undesirable outcome.
- Risk based testing is basically a testing done for the project based on risks.
- Risk based testing uses risk to prioritize and emphasize the appropriate tests during test execution

1. Risk-based testing starts early in the project, identifying risks to system quality and using that knowledge of risk to guide testing planning, specification, preparation and execution.

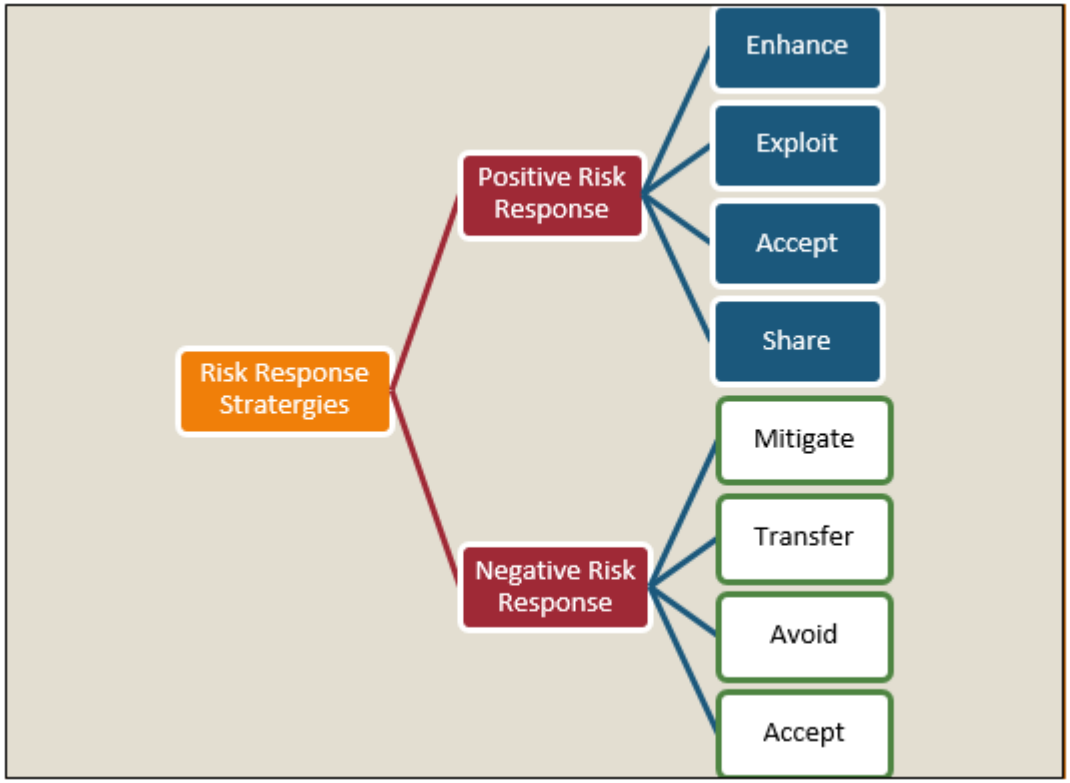
2. Risk-based testing involves both mitigation, contingency

3. Risk-based testing also involves measuring how well we are doing at finding and removing defects in critical areas.

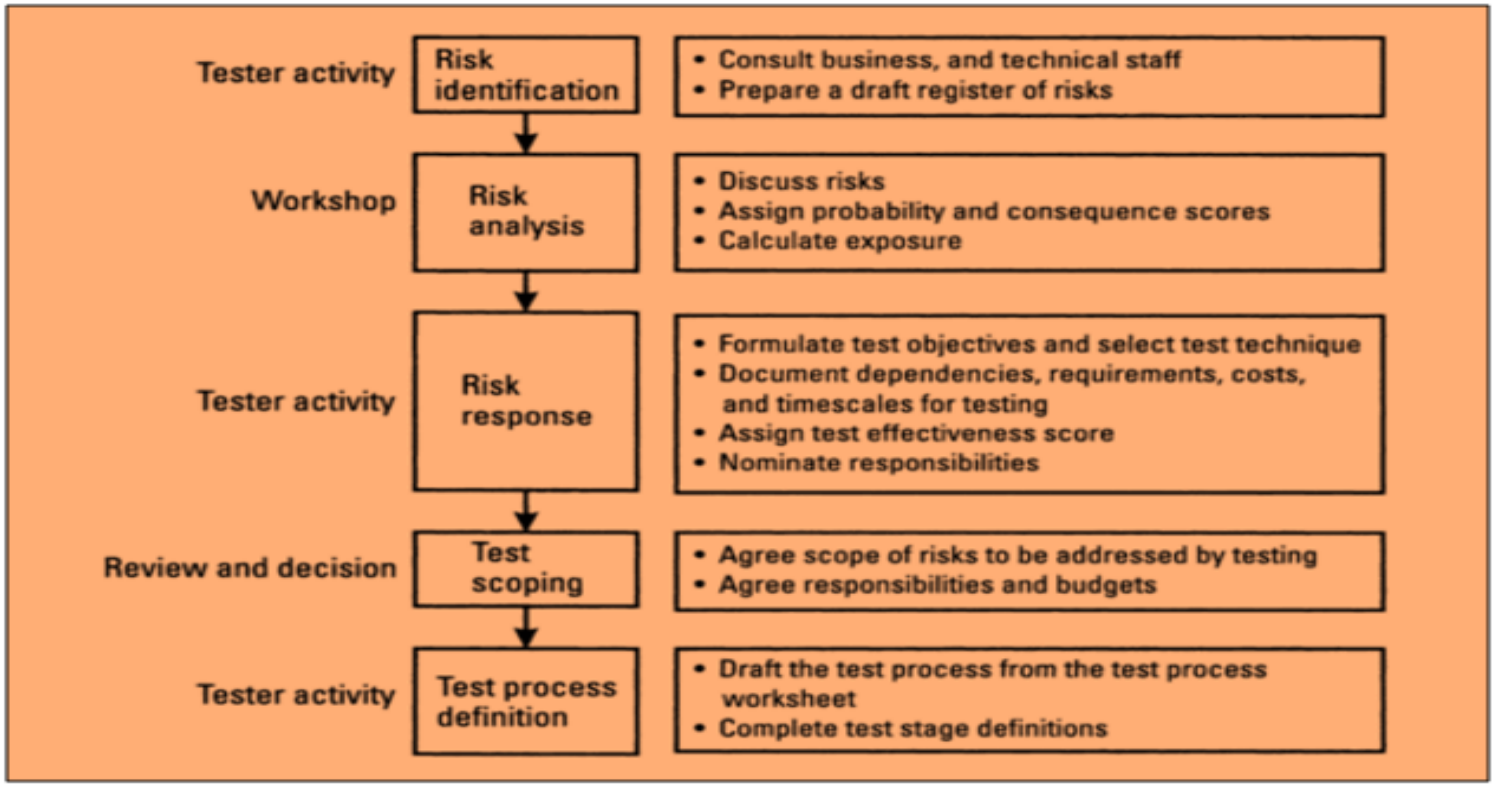
4. Risk-based testing can also involve using risk analysis to identify proactive opportunities to remove or prevent defects through non-testing activities and to help us select which test activities to perform.


RISK WBS
TIME
Schedule Delays
Force Majeure
Changes
PEOPLE
Shortages
Labor Strikes
Skills Deficiency
COSTS
Escalation
Estimating Errors
Penalty Exposure
Exchange Rates
DELIVERABLES
Prob. of Tech. Success
Component Issues
Unproven Design
QUALITY
Performance
Warranty Exposure
Subcontractor Items
CONTRACT
Lack of Clarity
Disputes
Subcontractor Failures
Funding
MARKET
Prob. of Comm. Success
Window of Opportunity

- **Risk mitigation** is a risk response method used to lessen the adverse impacts of possible threats. This can be done by eliminating the risks or reducing them to an acceptable level.



- **Risk Contingency**: Contingency can be described as a possibility of an uncertain event, but the impact is unknown or unpredictable
- A contingency plan is also known as the action plan/back up plans for the worst case scenarios



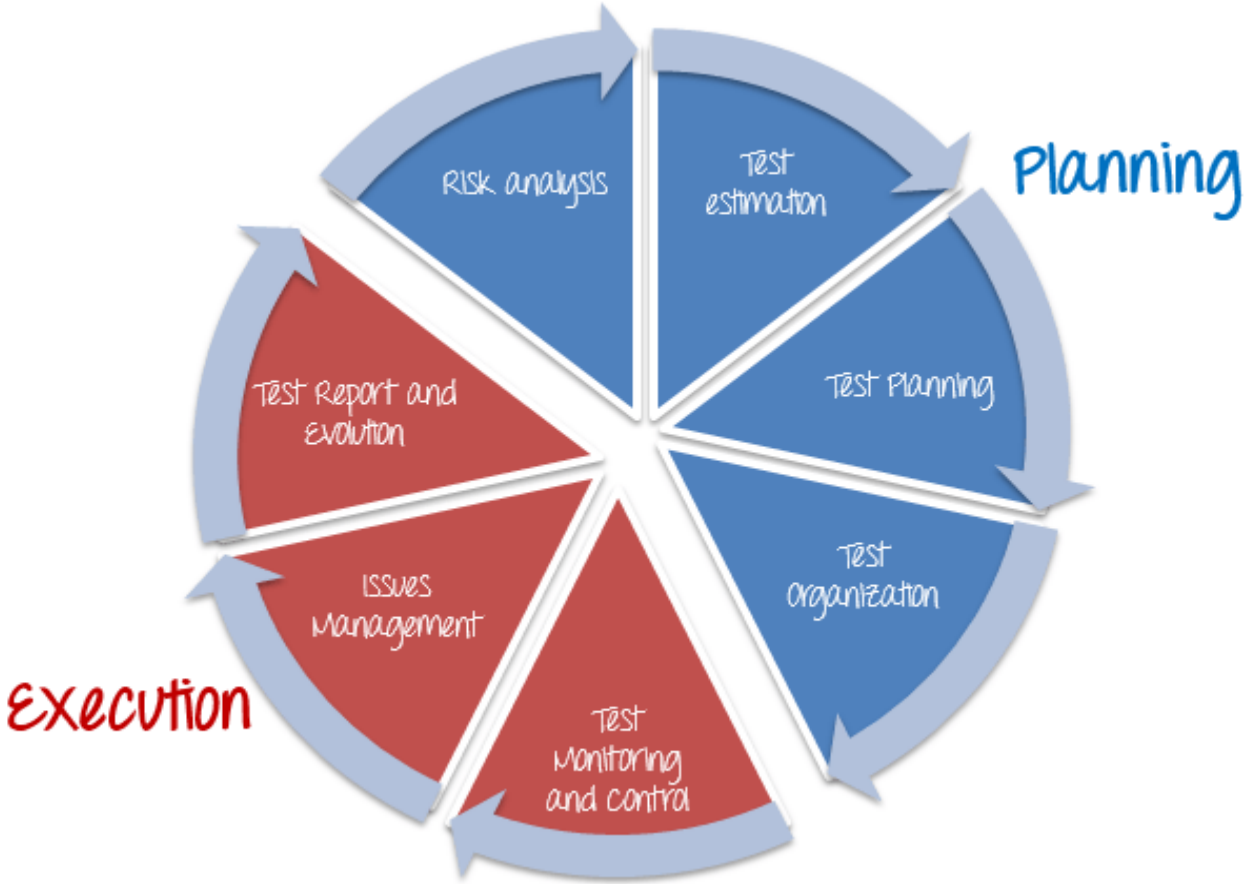
Test Management

- **Test Management** is a process of managing the testing activities in order to ensure high quality and high-end testing of the software application.

- The method consists of organizing, controlling, ensuring traceability and visibility of the testing process in order to deliver the high quality software application.

- It ensures that the software testing process runs as expected.

- **Test Management Phases**



Incident Management

- Incidents can be defined in simple words as an event encountered during testing that requires review.
- **Error:** An action performed by human that result in unexpected system behavior.

- **Defect:** This is a term usually used by testers. When tester finds a mistake or problem then it is referred to as Defect.