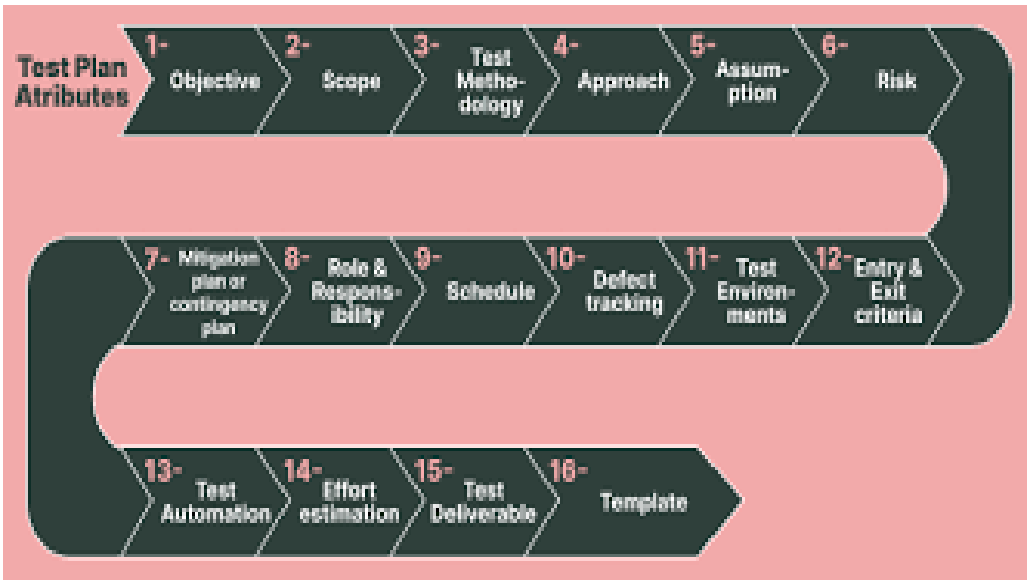
TASK-10 TEST PLAN

A Test plan is a detailed document which explains all testing areas and activities. It outlines test strategy, objectives, test schedule, required resources, test estimation and test deliverables. The test plan is prepared by the test lead, test manager, and test engineer.

TYPES OF TESTPLAN:

- **Major test plan**- Master Test Plan is a type of test plan that has multiple levels of testing. It includes a complete test strategy.
- **Phase test plan**- A phase test plan is a type of test plan that addresses any one phase of the testing strategy. For example, a list of tools, a list of test cases, etc.
- **Specific test plan**- Specific test plan designed for major types of testing like security testing, load testing, performance testing, etc. In other words, a specific test plan designed for non-functional testing.

TEST PLAN COMPONENTS OR ATRIBUTES:



OBJECTIVES:

It consists of information about modules, features, test data etc., which indicate the aim of the application means application behaiour ,goal etc.,

SCOPE:

It consists of information that must be tested with respective of an application. The scope is further divided into two parts:

- In scope
- Out scope

INSCOPE: These are the modules that need to be tested rigorously.

OUTSCOPE: These are the modules that need not to be tested rigorously.

FOR EXAMPLE:

Suppose we are testing a Gmail application we need to check the features like compose mail, sent items, inbox, drafts etc. and the features that should not be tested are like help and so on., the features that need to be tested are decided at the planning phase itself based on the time limit given.

TEST METHODOLOGY:

The test methodology consists of the information about what kind of testing is performed, like functional testing, integration testing, system testing etc., on an application. In this we will decide what type of testing we will perform on the various features based on the application requirement. And here, we should also define what kind oof testing we will use in the testing methodologies so that everyone likes the management, development team and the testing team can understand easily because the testing terms are not standard. For example, for a standalone application like Adobe Photoshop, we will perform the following testing types

Smoke testing >Functional testing>Integration testing>System testing>Ad hoc testing>Compatibility testing>Regression testing>Globalization testing>Accessibility testing>Usability testing>Reliability testing>Recovery testing>Installation or Uninstallation testing.

APPROACH:

This attribute is used to describe the flow of the application while performing testing and for future reference.

We can understand the flow of the application with the help of the below aspects:

- By writing the high-level scenarios.
- By writing the flow graph.

Writing high level scenarios:

For example, suppose we are testing the Gmail application:

Send a mail and check whether it is in sent mails or not and so on.

We are writing this to describe the approaches which must be taken for testing the product and only for critical features will we write the high-level scenarios.

Writing the flow graph

The flow graph is written because writing the high-level scenarios is a bit of a time taking process.

Benefit's are:

- Coverage is easy
- Merging is easy

The approach can be classified in two parts which are two Top to Bottom approach and Bottom to top approach.

ASSUMPTION:

It contains information about a problem or issue which maybe occurred during the testing process, and we are writing the test plans, the assured assumptions would be made like resources and technologies, etc.,

RISK:

Risks are the challenges we need to face to test the application in the current release. If the assumptions fail, then risks are involved. For example, the effect of an application release date becomes postponed.

MITIGATIO PLAN OR CONTINGENCY PLAN:

It is a backup plan which is prepared to overcome the risks or issues.

The assumptions, risk and mitigation or contingency plan are precise on the product itself. The various types of risks are as follows:

Customer perspective

Resource approach

Technical opinion

ROLE & RESPONSIBILITY:

It defines the complete task which needs to be performed by the entire testing team. When a large project comes, then the Test Manager is the person who writes the test plan. If there are 3-4 small projects, then the test manager will assign each project to each Test Lead. And then, the test lead writes the test plan for the project, which he/she is assigned.

SCHEDULE:

It is used to explain the timing to work, which needs to be done or this attribute covers when exactly each testing activity should start and end? And exact data is also mentioned for every testing activity for a particular date.

DEFECT TRACKING:

It is generally done with the help of tools because we cannot track the status of each bug manually. And we also comment about how we communicate the bugs which are identified during the testing process and send it back to the development team and how the development team and reply. Here we also mention the priority of the bugs such as high, low and medium.

TEST ENVIRONMENTS:

These are the environments where we will test the application, and here we have two types of environments, which are software and hardware configuration. The software configuration means the details about operating systems such as windows, Linux, UNIX, mac and various browsers like google chrome, Firefox, internet explorer etc.,

The hardware configuration means the information about different sizes of RAM, ROM, and the processors.

ENTRY AND EXIT CRITERIA:

It is a necessary condition which needs to be satisfied before starting and stopping the testing process.

Entry Criteria

The entry criteria contain the following conditions:

- White box testing should be finished.
- Understand and analyze the requirement and prepare the test documents or when the test documents are ready.
- Test data should be ready.
- Build or the application must be prepared
- Modules or features need to be assigned to the different test engineers.
- The necessary resources must be ready.

Exit Criteria

The exit criteria contain the following conditions:

- When all the test cases are executed.
- Most of the test cases must be passed.
- Depends on severity of the bugs which means that there must not be any blocker or major bug, whereas some minor bugs exist.

TEST AUTOMATION

In this, we will decide the following:

Which feature must be automated and not automated?

Which test automation tool are we going to use on which automation framework?

We automate the test case only after the first release.

Effort estimation

In this, we will plan the effort needed to be applied by every team member.

Test Deliverable

These are the documents which are the output from the testing team, which we handed over to the customer along with the product. It includes the following:

Test plan

Test Cases

Test Scripts

RTM (Requirement Traceability Matrix)

Defect Report

Test Execution Report

Graphs and metrics

Release Notes

Graphs and Metrics

Graph

In this, we will discuss the types of graphs we will send, and we will also provide a sample of each graph.

We have different kinds of graph

Bug distribution graph

Build wise graph

Bug trend analysis graph

Template

This part contains all the templates for the documents that will be used in the product, and all the test engineers will use only these templates in the project to maintain the consistency of the product. Here, we have different types of the template which are used during the entire testing process such as:

Test case template

Test case review template

RTM Template

Bug Report Template

Test execution Report

Test Plan Guidelines

1. Collapse your test plan.
2. Avoid overlapping and redundancy.
3. If you think that you do not need a section that is already mentioned above, then delete that section and proceed ahead.
4. Be specific. For example, when you specify a software system as the part of the test environment, then mention the software version instead of only name.
5. Avoid lengthy paragraphs.
6. Use lists and tables wherever possible.
7. Update plan when needed.
8. Do not use outdated and unused documents.

Test plan template:

1.0 overview

2.0 scope

2.1 in-scope

• features to be tested

• types of testing applicable

2.2 out-scope

• features not tested

• types of testing not applicable

3.0 approach

3.1 test design approach

3.2 test execution approach

3.3 defect management approach

4.0 resources

4.1 hardware resources

4.2 software resources

4.3 human resources

6.0 entry criteria and exit criteria

6.1 entry criteria

6.2 exit criteria

7.0 test deliverable

7.1 test scenarios

7.2 test cases

7.3 test date documents

7.4 defect reports

7.5 test summary reports