In software development, a test plan defines your testing team’s test strategy, goals, and scope, which ultimately work together to ensure that all your software components are tested sufficiently before a release.

Follow these six steps to create an efficient test plan:

1. Define the release scope :

Before any test activity occurs, it’s important to define the scope of testing for your release. This means defining the features or functions that need to be included in the release, considering any constraints and dependencies that can affect the release, and determining what type of release it is.

Are there new features being released in this version?

What are the risk areas?

For example, what information would you require if your organization has just launched a new e-commerce site and wants to test it before it launches?

1. Schedule timelines :

Specify release deadlines to help you decide your testing time and routine.

* Consult your project manager to understand the current release timeline.
* Look at past release times and schedules.
* Consider the timeframes for development.

1. Define test objectives :

A test objective is a reason or purpose for designing and executing a test.

Examples of general test objectives include:

* Identifying and reporting defects
* Testing new features
* A certain level of test coverage

1. Determine test deliverables :

Test deliverables are the products of testing that help track testing progress.

**Before testing :**

* + Test plan document
  + Test suite
  + **Test design and environment specifications**

**During testing :**

* + Test log
  + Defect report
  + Test data
  + Test summary report

**After testing :**

* + Test completion report
  + User acceptance test (UAT) report
  + Release notes

1. Design the test strategy :

Test strategy helps determine test cost, test effort, and which features will be in-scope (planned to be tested) versus out-of-scope (not planned to be tested).

* Test objectives
* Your project’s feature requirements
* The complexity of your product
* Your team’s experience levels
* Regulatory requirements
* Time and budget

1. Plan test environment and test data :

Planning a test environment guarantees precise and robust testing. The test environment includes hardware, software, and network configurations for software testing.

* + Determine your hardware and program requirements.
  + Install the required software.
  + Create the test data.