

General Concepts

[blocked URL](#)

The terms listed below originate in qTest Manager.

Objects: Objects can be edited, deleted, copied or cut on the system tree to manage your test project in a flexible manner.

Release: A Release represents an internal or external distribution of software once a development loop within a product's lifecycle has been completed. In qTest, a start and end date of a Release must be defined.

Build: The Build information includes data on the relevant testing plan, tests conducted, test results, and Defect records. You can define Build as a Milestone. You might have a Build for your UAT environment and then another Build for a Beta release.

Module: Users can divide applications into modules, which are hierarchical groupings based on the product's structure and functionality. Requirements and Test Cases can be assigned to specific module levels. Often, it is helpful to organize Requirements and Test Cases into logical groupings, according to application functionality. Modules are commonly shared between the Requirement tree and the Test Case tree. Creating or editing modules in the Requirement tree will affect the Test Design tree and vice versa.

Requirement: Requirements describe in detail what needs to be achieved in order for the software developed to meet its objectives.

Test Cycle: The testing of a Release can be broken into a number of shorter Cycles and are assigned to an overall release, or you can assign them to a Build within a release. A Test Cycle is a container that shows a high-level summary of its underlying Test Suites and Test Runs, including the execution results of these tests and any Defects found. Within a given release, you may need to execute many types of tests such as new features and bug fixes. For this reason, you may have more than one Test Cycle within the Release or even multi-level Test Cycles, which will be beneficial for organizing and reporting. A possible example of multi-level Test Cycles would be for new features that need to be tested on a browser version and a mobile version of your software.

Test Suite: Test Suites are comprised of a set of the same type of Test Cases, such as functional tests, non-functional tests, sanity tests, or possibly even regression tests. Test Suites are focused on addressing specific test objectives. For example, tests in a Test Suite may be grouped to validate specific components of your software or to test given levels of integration. A Test Suite can be regarded as the lowest level container to organize test runs.

Test Cycles and Test Suites are simply containers for your Test Cases. How you use and organize these features are determined by your organization.

Test Case: A Test Case describes in detail what needs to be tested to meet quality objectives. A Test Case details steps, including actions and scenarios, and expected results.

Test Run: A Test Run refers to an uninterrupted period of time spent executing a Test Case. For each Test Run, the tester is provided with the instructions and resources required to perform the test. In addition, the Test Run may specify a choice of possible results and provide testers with the ability to attach supporting documents and comments.

Defect: A flaw that causes a component or system to fail to behave or function as expected. If encountered during execution, a Defect may cause a failure of the component or system.

System User Roles

- **Site Admin:** The first user who registers a site on the Tricentis Web page will be granted full privileges on site settings page.
- **Project Admin:** A user who is normally assigned to a project created by the Site Admin and who has all permissions enabled on their assigned project.
- **Project User:** Any other users who are invited to the registered site by the Site Admin and added to a specific project by the Project Admin. The Project Admin can specify a Project User's permissions and access rights for within a specific project.

qTest Menu: qTest's major menus provide different capabilities that correspond to each step in the product testing life cycle. qTest includes the following System tabs: Test Plan, Requirements, Test Design, Test Execution, Defects, and Reports.

Tree panel: A hierarchical display of objects, for example, with the root page at the top and underlying objects at the bottom. The tree panel of each menu displays different objects:

- **Test Plan menu:** Release tree
- **Requirements menu:** Requirement tree
- **Test Design menu:** Test Case tree
- **Test Execution menu:** Test Run tree

Test Data Source: A test database that is used for a specific test suite. All test runs for a given test suite must be based on the data extracted from the associated test data source.