Types of testing strategies

Some of the **testing methodologies** that may be part of an **organization’s testing strategy** are:

* [**Analytical strategy**](http://tryqa.com/what-is-test-strategy-types-of-strategies-with-examples/#analytical_strategy)
* [**Model based strategy**](http://tryqa.com/what-is-test-strategy-types-of-strategies-with-examples/#model_based_strategy)
* [**Methodical strategy**](http://tryqa.com/what-is-test-strategy-types-of-strategies-with-examples/#methodical_strategy)
* [**Standards compliant or Process compliant strategy**](http://tryqa.com/what-is-test-strategy-types-of-strategies-with-examples/#standards_or_process_compliant_strategy)
* [**Reactive strategy**](http://tryqa.com/what-is-test-strategy-types-of-strategies-with-examples/#reactive_strategy)
* [**Consultative strategy**](http://tryqa.com/what-is-test-strategy-types-of-strategies-with-examples/#consultative_strategy)
* [**Regression averse strategy**](http://tryqa.com/what-is-test-strategy-types-of-strategies-with-examples/#regression_averse_strategy)

Analytical strategy

For example [**risk based testing**](http://tryqa.com/what-is-risk-based-testing-identifying-assessing-minimizing-managing-risks/) or [**requirements based testing**](http://tryqa.com/test-selection-techniques-requirement-model-based-checklists-reactive-testing/#requirement_based_testing). Here the testing team defines the [**testing conditions**](http://tryqa.com/what-is-test-condition-test-analysis-advantages-disadvantages-level-of-detail/) to be covered after analyzing the [**test basis**](http://tryqa.com/what-is-test-analysis-or-how-to-identify-the-test-conditions/), be it **risks** or **requirements**, etc.

So, in case of testing based on requirements, requirements are analyzed to derive the test conditions. Then [**tests are designed**](http://tryqa.com/what-is-test-design-when-to-create-it/)**,**[**implemented**](http://tryqa.com/what-is-test-implementation-advantages-disadvantage-of-early-implementation/)**and**[**executed**](http://tryqa.com/what-is-test-execution/) to meet those requirements.

Even the results are recorded with respect to requirements, like requirements tested and passed, those that were tested but failed and those requirements which are not fully tested, etc.

Model based strategy

In this technique, testing team chooses an **existing or expected situation** and **creates a model for it**, taking into account inputs, outputs, processes and possible behavior.

The models are also developed according to existing software, hardware, data speeds, infrastructure, etc.

Let us consider the scenario of mobile application testing. To carry out its [**performance testing**](http://tryqa.com/what-is-performance-testing-in-software/), models may be developed to emulate outgoing and incoming traffic on mobile network, number of active/inactive users, projected growth, etc.

Methodical strategy

Here test teams follow a **predefined quality standard** (like ISO25000), **checklists** or simply a **set of test conditions**. Standard checklists can exists for specific types of testing (like security), application domains.

For instance, in case of [**maintenance testing**](http://tryqa.com/what-is-maintenance-testing/), a checklist describing important functions, their attributes, etc. is sufficient to perform the tests

Standards or process compliant strategy

Medical systems following US Food and Drugs Administration (FDA) standards are good examples of this technique.

Here the testers follow the **processes or** **guidelines established by committee for standards** or panel of industry experts to identify test conditions, define test cases and put testing team in place.

In the case of a project following [**Scrum**](http://tryqa.com/what-are-agile-software-development-approaches-scrum-kanban-xp-explained/) Agile technique, testers will create its complete test strategy, starting from identifying test criteria, defining test cases, executing tests, report status etc. around each [**user stories**](http://tryqa.com/what-is-user-story-template-in-agile-software-development/).

Reactive strategy

Here tests are designed and implemented only after the real software is delivered. So **testing is based on defects found** in actual system.

Consider a scenario where [**exploratory testing**](http://tryqa.com/what-is-exploratory-testing-in-software-testing/) is being used. Test charters are developed based on the existing features and functionalities. These test charters are updated based on the results of the testing by testers. [**Exploratory testing can be applied to Agile**](http://tryqa.com/what-is-exploratory-testing-in-agile-methodology/) development projects as well.

Consultative strategy

As the name suggests, this testing technique **uses consultations from**[**key stakeholders**](http://tryqa.com/how-to-identify-stakeholders-for-testing/)**as input** to **decide the scope** of test conditions as in the case of user directed testing.

Let us consider a situation where the [**compatibility**](http://tryqa.com/what-is-compatibility-testing-in-software/) of any web based application with possible browsers is to be tested. Here the application owner would provide a list of browsers and their versions in order of priority.

They may also provide a list of connection types, operating systems, anti malware software, etc. against which they want the application to be tested.

The testers may then use different techniques like testing pair wise or [**equivalence partitioning**](http://tryqa.com/what-is-equivalence-partitioning-in-software-testing/) techniques depending upon priority of the items in the provided lists.

Regression averse strategy

Here testing strategies **focus on** [**reducing regression risks**](http://tryqa.com/how-to-manage-regression-risk-and-evolve-manual-automated-test-cases-in-agile-methodology/) for functional or non-functional product parts.

Continuing our previous example of web application, if the application needs to be [**tested for regression issues**](http://tryqa.com/what-is-regression-testing-in-software/), testing team can create test automation for both typical and exceptional use cases.

They can even use GUI based automation tools so that the tests can be run whenever the application is changed.

It is not necessary to use any one of the techniques listed above for any testing project. Depending on product and organization’s requirements, two or more techniques may be combined.

Details included in test strategy

The final **test strategy should include details** about these factors:

* Test levels
* Entry as well exit conditions for each test level
* Relationships between the test levels
* Procedure to integrate different test levels
* Techniques for testing
* Degree of independence of each test
* Compulsory as well as non-compulsory standards that must be adhered
* Testing environment
* Level of automation for testing
* Tools to be used in testing
* Confirmation and regression testing
* Re-usability of both software and testing work products
* Controlling testing
* Reporting on test results
* Metrics and measurements to be evaluated during testing
* Managing defects detected
* Managing test tools and infrastructure configuration
* Test team members roles and responsibilities

Test strategy selection

The **testing strategy selection may depend on these factors**:

* Is the strategy a short term or long term one?
* Organization type and size
* Project requirements – Safety and security related applications require rigorous strategy
* Product development model