**Severity & Priority in Testing: Introduction & Differences**

**What is Severity?**

Severity is defined as the degree of impact a defect has on the development or operation of a component application being tested.

Higher effect on the system functionality will lead to the assignment of higher severity to the bug. Quality Assurance engineer usually determines the severity level of defect

*Defect severity can be categorized into four classes*

* **Critical**: This defect indicates complete shut-down of the process, nothing can proceed further
* **Major**: It is a highly severe defect and collapse the system. However, certain parts of the system remain functional
* **Medium**: It cause some undesirable behavior, but the system is still functional
* **Low**: It won't cause any major break-down of the system

**What is Priority?**

Priority is defined as the order in which a defect should be fixed. Higher the priority the sooner the defect should be resolved.

Defects that leave the software system unusable are given higher priority over defects that cause a small functionality of the software to fail.

*Defect priority can be categorized into three class*

* **Low:**The defect is an irritant but repair can be done once the more serious defect have been fixed
* **Medium:**During the normal course of the development activities defect should be resolved. It can wait until a new version is created
* **High:**The defect must be resolved as soon as possible as it affects the system severely and cannot be used until it is fixed

**Tips for determining Severity of a Defect**

* **Decide the frequency of occurrence:** In some cases, if the occurrence of a minor-defect is frequent in the code, it can be more severe. So from user's perspective, it is more serious even though it is a minor defect.
* **Isolate the defect:** Isolating the defect can help to find out its severity of the impact.

|  |  |
| --- | --- |
| Priority | **Severity** |
| Defect Priority is defined the order in which the developer should resolve a defect | * Defect Severity is defined as the degree of impact that a defect has on the operation of the product |
| Priority is categorized into three types  Low  Medium  High | * Severity are categorized into five types   + Critical   + Major   + Moderate   + Minor   + Cosmetic |
| Priority is associated with scheduling | * Severity is associated with functionality or standards |
| Priority indicates how soon the bug should be fixed | * Severity indicates the seriousness of the defect on the product functionality |
| Priority of defects is decided in consultation with the manager/client | * QA engineer determines the severity level of the defect |
| Priority is driven by business value | * Severity is driven by functionality |
| Its value is subjective and can change over a period of time depending on the change in the project situation | * Its value is objective and less likely to change |
| High priority and low severity status indicates, defect have to be fixed on immediate bases but does not affect the application | * High severity and low priority status indicates defect have to be fixed but not on immediate bases |
| Priority status is based on the customer requirements | * Severity status is based on the technical aspect of the product |
| During UAT the development team fix defects based on priority | * During SIT, the development team will fix defects based on the severity and then priority |

**Difference between Priority and Severity**

**A software system can have a**

Let see an example of low severity and high priority and vice versa

* **A very low severity with a high priority:** A logo error for any shipment website, can be of low severity as it not going to affect the functionality of the website but can be of high priority as you don't want any further shipment to proceed with wrong logo.
* **A very high severity with a low priority:**Likewise, for flight operating website, defect in reservation functionality may be of high severity but can be a low priority as it can be scheduled to release in a next cycle.

**Defect Triage**

Defect triage is a process that tries to do the re-balancing of the process where test team faces the problem of limited availability of resources. So, when there are large number of the defect and limited testers to verify them, defect triage helps trying to get as many defects resolved based on defect parameters like severity and priority.

**Defect Triage Process:**

Most systems use priority as the main criteria to assess the defect. However a good triage process considers the severity as well.

The triage process includes following steps

* Reviewing all the defects including rejected defects by the team
* Initial assessment of the defects is based on its content and respective priority and severity settings
* Prioritizing the defect based on the inputs
* Assign the defect to correct release by product manager
* Re-directs the defect to the correct owner/team for further action

**Guidelines that every tester should consider before selecting severity**

Severity parameter is assessed by the tester whereas the priority parameter is assessed by the product manager or by the triage team. For prioritizing the defect, it is imperative for a tester to choose the right severity to avoid confusion with the development team.

* Understand the concept of priority and severity well
* Always assign the severity level based on the issue type as this will affect its priority
* Understand how a particular scenario or test case would affect the end-user
* Need to consider how much time it would take to fix the defect based on its complexity and time to verify defect

**Conclusion:**

* Assigning wrong severity to defect can delay the STLC process and can have some drastic implication on the overall performance of the team. So, the responsible person needs to be precise and accurate on its call for assigning defect.

**Example for High Priority and Low Severity Bug**

**Example 1)** Next example is about High Priority and Low Severity Bug. Here we take example of yahoo.com. Everyone remind the logo or yahoo.com. Suppose while updating yahoo.com by mistake they updated the wrong logo with spell missing like yaho.com here 'o ' is missing. It should be Yahoo.com. Now here**bug is High Priority and Low Severity Bug**Because  
  
**This bug is High Priority**- Yahoo.com is company logo and mistake in company logo need to resolve on high priority to keep brand.  
  
**This bug is Low Severity**- As its just spelling mistake then impact on  user is not much high.  
  
  
**Example 2)**  
Suppose there is ATM machine. Every one use ATM machine for daily transactions.  
When you visit ATM machine sometime you may see advertisements from bank on festival season. Suppose on 2nd Oct on occasions of Gandhi jayanti (Last date to do saving is 3rd oct) bank is giving 0.5% extra (basic 5 % + additional 0.5%= 5.5%) additional interest for senior citizens on savings. This advertisement need to flash on or before 3rd oct. Suppose you visited ATM on 4th Oct morning and still you are seeing this add and date displaying in advertisement is 3rd oct last date to do savings.

Here priority is high as advertisement is flashing on multiple ATMs after end of date to do savings. and Impact is not much as date displaying in  advertisement 3rd oct  
 means scheme is already over so severity is low . So need to stop this advertisement on priority.

**Example High Priority High Severity Bug**

As I have near about 5+ years experience in software testing i realized that setting correct priority and severity is necessary to overcome on bug. That's the reason many times people face this question" Please give the example of High priority and High severity Bug" . Reason behind it to find out person is able to set severity and priority. There are 4 combinations of Severity and priority.

1. **High Priority and High Severity**
2. **High Priority and Low Severity**
3. **Low Priority and Low Severity**
4. **Low Priority and High Severity**

**Example for High Priority and High Severity :-**

 Suppose there is ATM Machine hope everyone is aware of ATM ma machine which is used for banking transactions. If ATM machine has bug like when user withdraw money from same bank ATM for which he is holding bank account, He is getting charged by 20 rs per transaction. Which is invalid as bank policy says withdrawing money from owns bank ATM no charge will be applied.

So this**bug is** **high priority**  because Bank is charging 20 rs per transaction for own ATM which is opposite to business logic.

and **bug is high severity** this bug need to resolved immediately because thousands of user withdraw money per hour so it cost high.

**Example of low priority and high severity bug**

This is the best example for low priority and high severity bug. Lest's think there is Banking application who gives interest of rs 2 for every 1000 rs in account on the last day of year. Means on last day of year 31.12.YYYY the bank will deposit 2 rs interest for every 1000rs in account. Now bank found a bug that instead of 2 rs application giving interest of 4 rs for every 1000 rs in account. Means due to bug interest is going double.  
  
This bug is high severity - Due to bug interest is going double and bank may have thousands of accounts, So it will not be profitable for bank.  
  
This bug is Low priority  - Depositing interest is happen on last day of year so if its beginning of year like January then there is lot of time to solve this bug .

1. [**Example for High Priority and Low Severity Bug**](http://softwareandtesting.blogspot.in/2013/10/Example-High-Priority-Low-Severity-Bug-testing.html)
2. [**Example for Low Priority and Low Severity Bug**](http://softwareandtesting.blogspot.in/2013/10/Example-Low-Priority-Low-Severity-Bug-testing.html)
3. [**Example for High Priority and High Severity Bug**](http://softwareandtesting.blogspot.in/2013/10/Example-high-Priority-High-Severity-Bug-Testing.html)

# Test Strategy Vs Test Plan

### **What is Test Plan?**

A test plan is defined as a document which outlines the scope, objective, method and weight on a software testing task

### **What is Test Strategy?**

Test strategy is defined as a set of guiding principle that enlightens test design & regulates how testing needs to be done

Test Plan V/s Test Strategy is a prominent confusion among multiple levels of QA Aspirants

Below is the detailed guide to it

## Difference between Test Strategy and Test Plan

|  |  |
| --- | --- |
| **Test Plan** | **Test Strategy** |
| * A test plan for software project can be defined as a document that defines the scope, objective, approach and emphasis on a software testing effort | * Test strategy is a set of guidelines that explains test design and determines how testing needs to be done |
| * Components of Test plan include- Test plan id, features to be tested, test techniques, testing tasks, features pass or fail criteria, test deliverables, responsibilities, and schedule, etc. | * Components of Test strategy includes- objectives and scope, documentation formats, test processes, team reporting structure, client communication strategy, etc. |
| * Test plan is carried out by a testing manager or lead that describes how to test, when to test, who will test and what to test | * A test strategy is carried out by the project manager. It says what type of technique to follow and which module to test |
| * Test plan narrates about the specification | * Test strategy narrates about the general approaches |
| * Test plan can change | * Test strategy cannot be changed |
| * Test planning is done to determine possible issues and dependencies in order to identify the risks. | * It is a long-term plan of action. You can abstract information that is not project specific and put it into test approach |
| * A test plan exists individually | * In smaller project, test strategy is often found as a section of a test plan |
| * It is defined at project level | * It is set at organization level and can be used by multiple projects |