**Beet Seed :**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **What is being checked ?** | **When applicable ?** | **Restrictions** | **Peculiarities** |
| **Functional** | It verifies the functionality of a  Software system or application and focuses on ensuring that the system behavior matches with the specified requirement and meets the customer/ business needs. | It can be performed at any stage in the development cycle when there is a working code to test. | * Incomplete or unclear requirements * Insufficient test data * Complex or changing test environment * Limited time or resources | * It focuses on basic usability of the system. * It tests accessibility of the function * It focuses on testing the main function * It checks the error condition * It can be done manually |
| **Non- Functional** | It verifies non-functional requirements of the application and tests all the aspects that are not tested in functional testing. It tests the readiness of the systema as per non-functional parameters. | It should be performed at all test levels and done as early as possible. | * It is hard to perform non-functional tests manually. | * It should be measurable * Ensures that quality attributes are identified correctly |
| **Retesting** | It is an iterative testing process that focuses on validating whether specific code changes or bug fixes have successfully removed the identified issues. | * When a particular bug is fixed with a new build. * When a particular bug is rejected. * Before regression testing * When the client requests it | * At the time of testing, retesting can not be automated. * When retesting results are unsuccessful, it requires more time and effort to fix all the issues | * It is a repetition of testing which are done with the same test cases in a new build. * It can only be implemented when particular test cases are involved which are considered as failed tests |
| **Regression** | It is a testing technique that re-runs functional and non-functional tests to ensure that a software application works as intended after any code changes, updates, revisions, improvements or optimizations. | It is performed after a code update to ensure that the update introduced no new bugs.  When configuration changes, adding new patches, a whole new functionality or feature is added to the product, a new requirement is added to the existing product feature. | The time available for regression testing is limited within each sprint. | It maintains existing code functionality while evaluating new program code and checking for flaws after testing the application. |
| **Smoke Testing** | It verifies that important features are working and there are no flaws in the build that is under testing. It test all critical functionalities are working correctly or not. | It is done whenever new functionalities of software are developed and integrated with existing build that is deployed in QA environment | A limited number of test scenarios are provided, so there is a chance of crucial bugs being missed. | * Based on the requirements it can be executed either manually or automated * It can be easily applied to different test levels like : system, integration, acceptance testing |
| **Sanity Testing** | It quickly checks whether the basic functionality of a new software build is working correctly or not. | Whenever there is an improvement in the functionality of the software, it should be performed. | It is not possible to cover all test cases in test scenarios  Due to its limited scope it may not detect all potential bugs in the software | It is a subset of regression testing mainly focused on less important units of the application.  It can't be documented. |

Differences between Regression and Retesting

|  |  |
| --- | --- |
| **Regression Testing** | **Retesting** |
| It tests the general area of the software | It tests a specific feature of the software |
| It tests a software which was working, but now due to updates which might not be working | It test a software which was not working earlier, but now might be fixed and working |
| Ideal for automation | Not ideal for automation |
| Should always be a part of testing and performed each time whent there is change/update in software | It should be a part of the testing process only when a bug is detected in the code. |
| A lower priority than retesting is applied to regression testing since it checks for potential defects | A higher priority is applied since this testing focused on known defects |

**Beet Sprout ;**

I think checking non-functional requirements and perform the non-functional testing is a necessity as long as it is about verifying whether the software/application meets the quality attributes that can not be tested through functional testing. If a non-functional testing is not done before releasing the product, it may result in some critical issues like system response time slows down in case multiple users use it at the same time.

I think smoke testing is necessary to detect critical issues as early as possible in a software/application before performing detailed testing. Uncovering defects at early stages also helps with cost saving and prevent delays in product release.

**Mighty Beet :**

Test Suit link for Cat Photos App

<https://beetrootqa2024.testrail.io/index.php?/suites/view/68&group_by=cases:section_id&group_order=asc&display_deleted_cases=0>