**1 . What is white box testing and list the types of white box testing ?**

Ans : testing based on an analysis of this internal structure of the component or system.

Structure based testing :

Technique is also know as ‘white box’ or ‘ glass-box’ testing technique because hare the testers requirement knowledge of how the software is implement how is work.

Coverage = number of coverage item exercised +100

Total number of coverage items

Type of coverage.

Statement coverage

Decision coverage

**Condition covarage**

- State ment \ segment coverage

` The statement coverage is also know as line covarage or segment coverage.

` The statement coverage covers only true conditions.

` the statement coverage can be calculated as show below.

- statment/sagment coverage :

**Adavntage**.

` if verifies what the written code is expected to do and not to do.

` if measures the quality of code written.

`if cheek the flow of different oaths in the program and it also ensure that whether those parts here tested or not.

**Disadvantage .**

` it cannot test test the false conditions.

` if dose not report that when there loop reaches its termination condition.

` if does not understand the logical oppressors.

\* derision/branch covarage:-

`Desing covarage also known as branch covarage or all edges covarage.

` it covarage both the true and false condition unlike the statement covarage.

`aim is to demo instate that all decisions have been run at least once.

- advantage:-

`to validate that all the branches in the code are reached .

` to ensure that no branches total to and abomination of the programs operation.

**Disltange** :-   
 `this netric ignores branches within boolean expressions which occur due to short clout operation.

Note:

Branch covarage testing > statement covarage testing

\* condition covarage :-

`This it closely related to decision covarage but has better sensitivity to the control flow.

` however full condition coverage does not grantee full decision covarage.

`condition covarage reports the true or false outcome of each condition.

` condition covarage measures the condition independently of each other.

1. **. what is black box testing ?what are the different black box testing techniques?**

Ans : introduction.

|  |  |
| --- | --- |
| Black box testing functional to the of the component. | Testing either functional without internal structure or system. |
| The testers have of how the system structure inside the box testing to is in what the software how it does it. | The knowledge or component its box in black tester is connecting software does not it. |
|  |  |

**Advantage** . - suiited and efficient for large code segment.

- code access not requirement.

**Disadvantage**:

- limited covarage since only a selected number of test selected number of test scenarios are actually

Perform.

- inefficient testing due to the fact that the tester only has limited knowledge about an application.

-bland covarage since the tester connect target specific code segment or error prone area.

**\* Techniques of black box testing.**

- equivalence partitioning

- boundary value analysis

- decision tables

- state transition box testing

Aim is to treat group of inputs as equivalent and to select one representative input to test level of testing.

Says that by testing trust one value we have tested the partition c+y radically a mid -point value is used it assume that.

- Of one value finds a bug the others probably will to .

-If on doesn't find a bug,the others probably will too.

\* Boundary value analysis is a methodology for designing test cases that concentrates software. \*testing effect on cases near the limits of valid ranges.

\*Boundary value analysis is a methodology which refines equivalence partitioning.

\*the trick is to concentrate software testing efforts at the extreme and of the equivalen classes.

**3). what I error,defect,bug and failure ?**

1. Ans : a mistake in coding is called error, found by tester is called defect, defect accepted by viper
2. Meet team then it is called bug, build does not meet the requirements then it is failure.
3. Error - a human action that produce an incontinent result .
4. Defect - a flaw in a component or system that can cause the component or system to fail to perform its required function.
5. Failure - deviation of the compost or system from is excepted delivery service or result.

**4).what is the purpose of exit criteria ?**

Ans : how to we know when to stop testing

- run out of time.

- run out of bug get.

- the business tells you .

- went live last might.

- boss save stop.

- all defects have been exited.

- out exit criteria have been met.

\* thoroughness measures, such as coverage requiem covarage .

\* estimate of defect density or reliability measure ( how many defect open by category.

\* cost residual risks, such as defect not fixed or luck of covarage In covarage in creative areas .

- schedules cash at those baste on time to market.

**5).what is integration testing ?**  
ans : integration testing :- testing performed to expose defects in the interference and in interactions between integrated component or system - there are a level of integration testing.

- compment integration testing . - system intragration testing.

1. **. what is component system testing ?**

Ans : component testing : testing performed to expose defect in the interferes and interaction between integrated component.the following testing technique are propitiate for integration testing.

-Component testing is a animal software item that can be tested in isolation. It means “a unit is the smallest testable part of software”.

-component testing is the testing of individual software component.

-component testing is a leal of the software testing process where individual unit/component of the software/system are tested.the purpose is or validate that each unit of the software perfom ad designed.

-component tester are typically written and run by software developer to ensure that the code meets its design and behave as intended with debugging tool.

-component test find problems early in the development cycle.

-component testing is performed by using the white box testing method.

**7).what is functional system testing ?**

Ans: black box testing techniques against the interfacing requirement for the component under test.

-functinoal system testing is requirement that specifies that specifies a function that a system or system component must perform.

-a requirement may exist as a test document and or/ a model.

-there is two types of test approach.

requirement based functional testing.

Process based testing.

**8) -functinoal system testing functinoal testing.**

A.Accuracy: provision of right or agreed results of effects.

B.Interoperability: ability to interact with specified system.

C. Comliance : adhere to applicable standards, conventions,regulations or laws.

D.suitability : presence and appropriateness of functions for specified tasks.

. What in non-functional testing ?

Ans : where appose sprite, for performance or reliability testing of the component interfaces for example.

-non-functional testing the attributes of a component or system that do not relate to functionality,e.g. reliablity,efficiency,interoperability,maintainability and portablity.

-may be performed at all test levels(not just non-functional system testing.

-measuring the characteristics of the system/software that can be quantified on a varying.scale-e.g. performance test scaling.

-non-funcctional testing includes,but is not limited to,performance testing, load testing,tress testing,usability testing,maintainability testing,reliablity testing and portability testing.

**9). what is component testing ?**

Ans : unit tests ,it affords several benefits. Unit testing is perforce by using the white box testing method.

Component testing is a minimal software item that can be tested in isolation. It mens “a unit is the smallest testable part of software.

- component testing is the testing of individual software component

-component testing is a level of the software testing process where individual unit/ component of the software/system are tested. The purpose is to validate that each unit of the software perform as designed.

-component tests are typically written and run by software developer to ensure that the code meets its depressing and behave as intended with debugging tool

-component test find problems early in the development

-component testing is performed by using the white box testing method.

**10 )Mention what big bang testing is ?**

Ans : in big bang integration testing all components or modules is integrated resistances, after which everything is tested as a whole.

\*advantage : convenient for small system .

\*disadvantage : fault localization is difficult.

Since all module are tested at one, high risk critical modules are not isolated and tested on priority, peripheral module which deal with user interrogation are also not isolated and tester on priority.

**11).difference between smoke and sanity ?**

Ans : \*smoke testing :

-check the criteria functionality

-it is done in initial stage.

-it check the stability.

-part of acceptance testing.

-general health cheek up.

-done by setter and developer.

-it cheek the system and to and.

\* sanity testing :

-cheek the new functionality.

-it is done after 30 build

- it cheek the sanity rationality.

-part of regression testing.

-advance health check up.

-done by tester.

-it check only particular function of entire system.

1. **. difference between QA v/s QC v/s tester.**

|  |  |  |
| --- | --- | --- |
| Quality assure activities which ensure the implantation of processes conductress & standards in contact to variation of development software & intended requirement. | Quality control activities which ensure that verification of devolved software with respect to document (cor no-tin some cases requirement ). | Activities which ensure the indication or bugs / error/ defect in the software. |
| Focuses on possess & procedure rather then condensing actual testing on the system. | Excuse on actual testing executing software with intended bug defect | Focus on actual testing |
| Process on end activities | Product oriented activities | Product on noted activities |

1. **What is stress testing ?**

Ans : Stress testing is a form of deliberately intense or thorough testing, used to determine the stability of a given system, critical infrastructure or entity. It involves testing beyond normal operational capacity, often to a breaking point, in order to observe the results. A heart stress test starts by making your heart pump harder and faster. For many people, this includes walking on a treadmill or riding a stationary bicycle. That's why the test is often called an exercise stress test.

1. **Explain the difference between functional testing & non-functional testing?**

**\* Funcional testing** : functinoal testing sperform using the functinoal personification proved by the client & verifies the system against the functional requirement.

Functinoal testing is executed first.manual testing or automation tools camber used fer functional testing. Bussines requirement are the input functional testing.functinonal testing are the input function testing.easy to do manual testing.

Type of functinoal testing are…

- unit testing

-smoke testing

-sanity testing testing.

-integration testing

-white box testing

-user acceptanle testing

**Non-functional testing** : testing cheeks performance liberality scalability and other non-functional aspects of the software system.non functional testing should be performed after functional testing using tools will be effective for this testing.

Permance parametes like speed,scalability are input to non-functional testing. Non-functional action testing describes how good the products working though to do manual.

Type of non-functional testing are…

-perfoman testing

-load testing

-volume testing

-stress testing

-security testing

-installation testing

-pentratim testing

-compablity testing

-migration testing.

1. **What Is exploratory testing ?**

- though the current trend in testing is to push for automation, exploratory testing is a new way of thinking.automation has its limits.

-is not random testing but it is haddock testing with purpose of find bugs

-is cognitively(thinking) structured as compared to procedural structure of scripted testing.this structure comes from chanter,time boxing e etc.

-is highly teachable and manageable.

-is not technique but it is approach.what actions you perform next is governed by what you are doing currently.

1. **What is traceability matrix ?**

- To protect against changes you should be able to trace back from back from every system component to the original requirement that caused its presence.

- a softtware process should help you keeping the virtual table up-ti-date.

1. **What is ad-hock testing ?**

-ad-hock testing is an informal testing type with an aim to break the system.

-it does not follow any test design techniques to create test cases.

-the fact is does not create test cases altogether.

-this testing is primarily performed if the knowledge of testers in the system under yest is very sigh.

-main aim of this testing is to find by random checking.

-ad-hoc tesring can be achieved with the testing technique called error guessing.

-error guessing can be done by the people having enough experience on the system to “guess” the most likely source of errors.

1. **What is exploratory testing ?**

- though the current trend in testing is to push for automation,exploratory testing is a new way of thinking.automation has its limits.

-is not random testing But it is ad-hock testing with purpose of find bugs

-is strucutred and rigorous.

-is cognitively (thinking) structured as compared to procedural structure of scripted testing.this structure comes from charter,time boxing etc.

-is highly teachable and manageable.

-is not a technique but it is an approach.what actions you next is governed by what you are doing currently.

1. **What is boundary value testing?**

- Boundary testing is the process of testing between extreme ends or boundar between partitions of the input values.

-So these extreme ends like Start- End, Lower- Upper, Maximum-Minimum, Just Inside-Just Outside values are called boundary values and the testing is called “boundary testing”.

-Boundary Value Testing is one of the popular software testing mechanism, where testing of data is done based on boundary values or between two opposite ends where the ends may be like from start to end, or lower to upper or from maximum to minimum.

1. **What is equivalence partitioning testing ?**

-Equivalence Partitioning Method is also known as Equivalence class partitioning (ECP).

-It is a software testing technique or black-box testing that divides input domain into classes of data, and with the help of these classes of data, test cases can be derived.

-It is abbreviated as ECP. It is a software testing technique that divides the input test data of the application under test into each partition at least once of equivalent data from which test cases can be derived.

-An advantage of this approach is it reduces the time required for performing testing of a software due to less number of test cases.

1. **What determines the level of risk ?**

-As Risk is determined by a combination of Probability and Severity, the main area of the Matrix reveals the Risk Levels.

-The levels are Low, Medium, High, and Extremely High. To have a low level of risk, we must have a somewhat limited probability and level of severity.

1. **What iS GUI testing ?**

-graphical user interface (GUI) testing is the process testing the system under test,GUI of the screens menu ,dialog boxes and window etc. -cheekyou can excuse the intended functionality of the application using the GUI.

-cheek all the GUI elements,for size,possimon,with length and acceptance of charters or number for, inductance,you must be able to provide input to the input fields.

-check error message are displayed correctly.

-cheek for clear demarcation of different section on screen.

-check that positioning of GUI of elements for different screen desolation.

- It even tests beyond the normal operating point and analyses how the system works under extreme conditions

1. **When should “regression testing” be performed?**

-Regression testing is necessary after any feature (or application) enhancement, bug fix, or configuration changes.

-For example, when developers add a new widget to an application. As more regressions are found in software products, companies are moving towards test automation to perform regression tests.

-Regression testing is a black box testing techniques. It is used to authenticate a code change in the software does not impact the existing functionality of the product.

-Regression testing is a type of [software testing](https://www.javatpoint.com/software-testing-tutorial). Test cases are re-executed to check the previous functionality of the application is working fine, and the new changes have not produced any bugs.

1. **Difference between verification and validation?**

- **Verification :**is the process of checking that a software achieves its goal without any bugs. It is the process to ensure whether the pro duct that is developed is right or not. It verifies whether the developed product fulfills the requirements that we have. Verification is static testing.

-**Validation:**is the process of checking whether the software product is up to the mark or in other words product has high level requirements. It is the process of checking the validation of product i.e. it checks what we are developing is the right product. it is validation of actual and expected product. Validation is the dynamic testing.

| **Verification** | **Validation** |
| --- | --- |
| It includes checking documents, design, codes and programs. | It includes testing and validating the actual product. |
| Verification is the static testing. | Validation is the dynamic testing. |
| It does not include the execution of the code. | It includes the execution of the code. |
| Methods used in verification are reviews, walkthroughs, inspections and desk-checking. | Methods used in validation are Black Box Testing, White Box Testing and non-functional testing. |
| It checks whether the software conforms to specifications or not. | It checks whether the software meets the requirements and expectations of a customer or not. |
| It can find the bugs in the early stage of the development. | It can only find the bugs that could not be found by the verification process. |
| The goal of verification is application and software architecture and specification. | The goal of validation is an actual product. |
| Quality assurance team does verification. | Validation is executed on software code with the help of testing team. |
| It comes before validation. | It comes after verification. |
| It consists of checking of documents/files and is performed by human. | It consists of execution of program and is performed by computer. |
| Verification refers to the set of activities that ensure software correctly implements the specific function. | Validation refers to the set of activities that ensure that the software that has been built is traceable to customer requirements. |
| After a valid and complete specification the verification starts. | Validation begins as soon as project starts. |
| Verification is for prevention of errors. | Validation is for detection of errors. |
| Verification is also termed as white box testing or static testing as work product goes through reviews. | Validation can be termed as black box testing or dynamic testing as work product is executed. |
| Verification finds about 50 to 60% of the defects. | Validation finds about 20 to 30% of the defects. |
| Verification is based on the opinion of reviewer and may change from person to person. | Validation is based on the fact and is often stable. |
| Verification is about process, standard and guideline. | Validation is about the product. |

1. **Explain types of performance testing ?**

-Stress Testing. This test pushes an application beyond normal load conditions to determine which components fail first. …

1.Capacity Testing.

2.Load Testing.

3.Volume Testing.

4.Stress Testing.

5.Soak Testing.

1. **Capacity Testing** ; Capacity Testing ensures that the application and environment can smoothly handle the maximum number of users or transactions according to the performance requirements defined in your Service-Level Agreement (SLA).
2. **Load Testing** ;Load testing generally refers to the practice of modeling the expected usage of a software program by simulating multiple users accessing the program concurrently. As such, this testing is most relevant for multi-user systems; often one built using a client/server model, such as web servers
3. **Volume Testing** ;  is a type of Software Testing, where the software is subjected to a huge volume of data. It is also referred to as flood testing. Volume testing is done to analyze the system performance by increasing the volume of data in the database.
4. **Stress testing** ; is a form of deliberately intense or thorough testing, used to determine the stability of a given system, critical infrastructure or entity. It involves testing beyond normal operational capacity, often to a breaking point, in order to observe the results.
5. **Soak testing;** is a type of performance evaluation that gauges how an application handles a growing number of users or increasingly taxing tasks over an extended period of time. Soak testing is both a type of nonfunctional test and a form of stress testing.
6. **Difference between priority & severity ?**

- Priority is a term that defines how fast we need to fix a defect. Severity is basically a parameter that denotes the total impact of a given defect on any software. Priority is basically a parameter that decides the order in which we should fix the defects. Severity relates to the standards of quality.

-Priority is the order in which the developer should resolve a defect whereas Severity is the degree of impact that a defect has on the operation of the product. Priority is categorized into three types: low, medium and high whereas Severity is categorized into five types: critical, major, moderate, minor and cosmetic.

|  |  |  |
| --- | --- | --- |
| Parameters | Severity in Testing | Priority in Testing |
| Definition | Severity is a term that denotes how severely a defect can affect the functionality of the software. | Priority is a term that defines how fast we need to fix a defect. |
| Parameter | Severity is basically a parameter that denotes the total impact of a given defect on any software. | Priority is basically a parameter that decides the order in which we should fix the defects. |
| Relation | Severity relates to the standards of quality. | Priority relates to the scheduling of defects to resolve them in software. |
| Value | The value of severity is objective. | The value of priority is subjective. |
| Change of Value | The value of Severity changes continually from time to time. | The value of Priority changes from time to time. |
| Who Decides the Defect | The testing engineer basically decides a defect’s severity level. | The product manager basically decides a defect’s priority level. |
| Types | There are 5 types of Severities: Cosmetic, Minor, Moderate, Major, and Critical. | There are 3 types of Priorities: High, Medium, and Low. |

1. **.what is the diffrence between the STLC(software testing lifecycle) and SDLC(software development life cycle**)**?**

Software devlopment cycle a seris of steos of phases that provides the modal of devolpment.

It is life cycle mavaoement for the place of software or application.

**[ Phase of SDLC : ]**

1. costomer need.
2. Requerment from stake holder,client,coustomer,eco,etc.
3. Improvement in current software .

Example **: 1) costomer requerment**

1 login.lohin with google,fecebook.

2 add/remove,to from cart.

3 system should tate less responetime.

4 wobsite must responsive.

1. **Planning analysis : what we want ?**
2. Datals on computer programing langaues envirments ,machiners,packages,application,archietuter,distributer archivture layering,memory size,platform, engineeing detalits are established.
3. Risk of the project.
4. cost of the project.
5. Time of the project.
6. **Design : how can we get what we want ?**

1 desing archivecture document.

2 implementation plan.

3 critical priority anaysis.

4 perfomanced anaysis.

5 pest plan.

1. **Implementatin … creat what we want ?**
2. In implementation phase,the team sulid the components either from scath or by composition.
3. Implementation-code
4. Critical removal
5. **Testing,did we get what we want ?**
6. We test the bulid to check for dwfect.
7. We reports the defect and get it fixed.
8. We retest bulid until it fulfills costomer requerment.
9. **Mantalnnce :**
10. Corrective malntenace : I dentiyng and reparing defect.
11. Adaptive maintance : adapating the exiting sdution to the new platform.
12. Perfecttive malntences : implememting the new requerment .

**[ Phase of STLC : ]**

1. Requerment anaylis
2. Test planning
3. Test case development
4. Test envrment setup
5. Test excution
6. Test sycle closure.

What is entry and exit criterial in STLC ?

**Entry critiaria** : entry critaria gives the pereusite that must be completed befor testing can begin.

**Exit critiaria** : exit criteria defines the items that must be completed befor testing can be conudd.

In an ideal world, you will not enter the next stage antill the exit criteria for the previos stage is melpractically this is not always possible.

Requrment anaysis :

1. Requirment phase testing : also known as requirment anaylis in which test team studies the requirment from a testing point of view to idenfy testble requements and the af team interat with valres stak no idiers to undatand requiement in detall.

Requrment feasibity for the testing project is also done in this stage

1. Activites in requrment plush testing .

- idenfy type of tests to be perform

- gather detalls about testing periouies and focous

- prepar matrix (RTM)

- idenfiy test enviorment detalls where testing id supposed to be carried out

- automation feasiblity anaysis if reduired

1. Devieverbles of requrment phase testing.

- RTM

Automation feasiblity report ( if application )

Example :

-login, with gogle and FB

-add/remove item to for cart

-login should test respcnce fiwe second

-weblsite must be resposive

**2 ) Test plaaning .**

-A phase in which a senarios management delermines the test plan staterg alon with efforts and cost estimates for the project.

-moreover ,the respornce ,test enviroment ,test limited and the testing schedule are also determined.

\* activites in requrment phase testing :

Preparation of test plan/staratery document for various types if testing .

-test efforts estimation.

-resource planning and detarming reles & responsibites.

-tranning requirment.

\* Deliverbe of requrment phase testing :

-Test plan/ statey documnt

- efforts esitimation document

1. **Test case devolpment :**

The test case development pase involes the cration,verification,and rework of test case and test srips after test plan is ready.

-fainlly he test data is idia fied then crated and reviewdd & then reworked based on the precconditions.

-then the QA team starts the development procces of test cases for individual units.

\*Activitis in requrment phase testing.

- creat test case, utmotinal sripts.

-review & baseline test cases & scripts.

-ereate test data [ if test rnvoment is avalible ].

1. **Deversbles of roo phase testing - test case/ scripts : test data**

-test fnvirment setup:

-test envirment setup decids the software & hardware conditions under which a work sroducts is tested.

-it is one critical aspect of the testing process & cand be done in parallel with the testing case devolpen phase.

-test team may not be involed in this activity if the devolopment team provides the test envirment the test team is requirment to do areadiness check smoke testing of the given envorment.

\*activites requrment phase testing :

Understand the redurd achivthin envirment set-up and pre pare hardware & software requrment list for the test envirment.

-set up envirment & test data.

\*delivarabe of requrment phase testing.

-enviroment ready with data set-up

-smoke test result.

1. **Test execution :**

-test execution phase is carried out by the testers in which testing of the software bulids is done based on test plans & test cases prepared.

-test execution phase is carried out by the testers in witchtesting of the software build is done based on test plans and test cases prepared.

-the process consistof testscript execution,test script manctenace & bug preporting.

-if bugs are reported in is revelted back to developnent team for correction & restesting will be preformed.

1. **Test cycle clousure.**

- test cycle clourse : phase is complation of test exectun which involes several activities like test completion reporting collection of testcompletion matrics & test result.

-testing team members meet,dissuse&anaylis testing artiacts to idenify strartinge that was have to be implemented in future talking lessons from currant test cycle.

-the idia is to remove orocess bottal enecks for future test cycle.

1. **What is the difference between test scenarios,test cases & test script ?**

**-What is a Test Case**

A Test Case is a document that contains the sequence of actions that validate if the application is working as per the business requirements. Test cases cover all software product features, from clicking a button and accepting input in a text box to checking for storage and a seamless user interface. A complete test case document has to include every possible combination of data and functionalities to be tested on an application before the final product is ready for roll-out.

Test cases give a clear picture of the business requirements of the application. Test case documenting is essential so the testers can recall the feature tested after a long gap. It helps in achieving uninterrupted software testing.

[Test Case Prioritization](https://www.browserstack.com/guide/test-case-prioritization" \o "Test Case Prioritization: A Detailed Guide" \t "https://www.browserstack.com/guide/_blank) is a vital step in writing test cases. Running all the test cases in a test suite requires a lot of time and effort. As the number of features increases, testing the entire suite for every build is practically impossible. Test case prioritization helps overcome these challenges.

**Test Script**

Test Scripts are step-by-step instructions on how to test a test case. They are detailed and contain individual steps that test for each and every functionality. Test scripts are programs that execute tests on the software product/application. The tester has to write and run the test scripts to validate if the application’s outcome meets the business requirements.

Test cases serve as a backbone for writing test scripts. A single test case can have multiple test scripts for different testing environments. While most test scripts identify with Automation testing, they can also be a part of manual testing. The testers write automated scripts that can generate data to test every functionality. Usually, the test scripts run automatically and check every test case with various possible input data. Python, Ruby, Perl, Java, VB Script etc are a few of the scripting languages used to write test scripts.

Once written, a test script can execute many test instances that test a single application function. Since the test scripts are more detailed and extensive than test cases, writing test scripts in a reusable format is an excellent practice to make the tester’s life easier.

**Differences between Test Case and Test Scenario**

A test Case is a set of actions executed to verify particular features or functionality, whereas a Test Scenario is any functionality that can be tested.

Test Case is mostly derived from test scenarios, while Test Scenarios are derived from test artifacts like BRS and SRS.

Test Case helps in exhaustive testing of an application, whereas Test Scenario helps in an agile way of testing the end-to-end functionality.

Test Cases are focused on what to test and how to test, while Test Scenario is more focused on what to test.

Test Cases are low-level actions, whereas Test Scenarios are high-level actions.

Test Case requires more resources and time for test execution, while Test Scenario requires fewer resources and time for test execution.

Test Case includes test steps, data, and expected results for testing, whereas Test Scenario includes an end-to-end functionality to be tested.

1. **Explain what test plan is ? what is the information that should be covred.**

-A Test Plan is a detailed document that catalogs the test strategies, objectives, schedule, estimations, deadlines, and resources required to complete that project.

-A test plan is a detailed document which describes software testing areas and activities. It outlines the test strategy, objectives, test schedule, required resources (human resources, software, and hardware), test estimation and test deliverables. The test plan is a base of every software's testing.

**-The following information should be included in your cover letter.**

-Information about you. Begin your cover letter with your contact information. ...

-Date.

-Contact Person's Name, Title, Employer, and Address.

-Salutation.

-Opening Paragraph.

-Middle Paragraph.

-Second Middle Paragraph.

-Contact Information and Closing.

1. **What is bug life cycle ?**

A bug life cycle in software testing is a set of statuses designed to coordinate defect management. A bug status helps keep all the members of the development team posted on the progress.

Defect/Bug Life Cycle Explained

-Tester finds the defect.

-Status assigned to defect- New.

-A defect is forwarded to Project Manager for analyze.

-Project Manager decides whether a defect is valid.

-Here the defect is not valid- a status is given “Rejected.”

-so, project manager assigns a status rejected

1. **What is priority** ?

something that is more important than other things and that needs to be done or dealt with first. Reorganizing the sales force will be a top priority for the new president.

-Something that is more important enything eles

-Gols determine priorities

-example :- goal : get into college

Priority : good grades,extracurricular activities

Destination : goal as

Route : priority.

1. **What is severity ?**

One can define Severity as the extent to which any given defect can affect/ impact a particular software. Severity is basically a parameter that denotes the impact of any defect and its implication on a software's functionality. In other words, Severity defines the overall impact that any defect can have on a system.

it defines the impact that a given defect has on the system.

1. **Bug categories are …..**

Different Types of Software Bugs

-Functional Bugs. Functional bugs are associated with the functionality of a specific software component. ...

-Logical Bugs. ...

-Workflow Bugs. ...

-Unit Level Bugs. ...

-System-Level Integration Bugs. ...

Out of Bound Bugs. …

A bug is an unexpected problem with software or hardware. Typical problems are often the result of external interference with the program's performance that was not anticipated by the developer. Minor bugs can cause small problems like frozen screens or unexplained error messages that do not significantly affect usage.

1. **advantage of bugzila ?**

The advantages of Bugzilla are -

Product quality has improved as a result.

Testing and development teams are able to communicate more effectively through it.

Adaptability is one of its strongest characteristics.

Pros of Bugzilla

-Open source, free bug tracking tool.

-Automatic Duplicate Bug Detection.

-Search option with advanced features.

-File/Modify Bugs By Email.

-Move Bugs Between Installs.

-Multiple Authentication Methods (LDAP, Apache server).

-Time Tracking.

**\*The advantages of Bugzilla are\* -**

Product quality has improved as a result.

Testing and development teams are able to communicate more effectively through it.

Adaptability is one of its strongest characteristics.

1. **Difference between priority & severity …..**

Priority is a term that defines how fast we need to fix a defect. Severity is basically a parameter that denotes the total impact of a given defect on any software. Priority is basically a parameter that decides the order in which we should fix the defects. Severity relates to the standards of quality.

**For example**: If an application or web page crashes when a remote link is clicked, in this case clicking the remote link by an user is rare but the impact of application crashing is severe. So the severity is high but priority is low.

1. **What are the different methodologies in aglie development model ?**

The below agile methodologies list comprises famous types of agile methodology that one can opt from:

1) Kanban. ...

2) Scrum. ...

3) Extreme Programming (XP) ...

4) Crystal. ...

5) Dynamic Systems Development Method (DSDM) ...

6) Feature-Driven Development (FDD)

- In agile methodology, everything is out there and transparent. The clients and decision-makers are actively involved in the initiation, planning, review, and testing. Whereas in the traditional approach, the project manager holds the reins of the project, thus others don't get to make the major decisions.

1. **Explain the diference between authorization & authentication in web testing .what are the common problmes faced in web testing?**

- Authentication is used to verify that users really are who they represent themselves to be. Once this has been confirmed, authorization is then used to grant the user permission to access different levels of information and perform specific functions, depending on the rules established for different types of users.

- Authentication is used to verify that users really are who they represent themselves to be. Once this has been confirmed, authorization is then used to grant the user permission to access different levels of information and perform specific functions, depending on the rules established for different types of users.

|  |  |
| --- | --- |
| Authentication | Authorization |
| Authentication verifies who the user is. | Authorization determines what resources a user can access. |
| Authentication works through [passwords](https://www.sailpoint.com/products/password-management/), one-time pins, biometric information, and other information provided or entered by the user. | Authorization works through settings that are implemented and maintained by the organization. |
| Authentication is the first step of a good identity and access management process. | Authorization always takes place after authentication. |
| Authentication is visible to and partially changeable by the user. | Authorization isn’t visible to or changeable by the user. |
| Example: By verifying their identity, employees can gain access to a human resources (HR) application that includes their personal pay information, vacation time, and 401K data. | Example: Once their level of access is authorized, employees and HR managers can access different levels of data based on the permissions set by the organization |

**Common Authentication Methods.**

- Oftentimes, these types of information are combined using [multiple layers of authentication](https://www.sailpoint.com/identity-library/authentication-methods-used-for-network-security/). For example, a user may be asked to provide a username and password to complete an online purchase. Once that’s confirmed, a one-time pin may be sent to the user’s mobile phone as a second layer of security. Combining multiple authentication methods with consistent [authentication protocols](https://www.sailpoint.com/identity-library/identity-management-protocols/), organizations can ensure security as well as compatibility between systems.

**A Strong Authentication & Authorization Strategy Is Essential**

- A sound security strategy requires protecting one’s resources with both authentication and authorization. With a strong authentication and authorization strategy in place, organizations can consistently verify who every user is and what they have access to do—preventing unauthorized activity that poses a serious threat. By ensuring all users properly identify themselves and access only the resources they need, organizations can maximize productivity, while bolstering their security at a time when data breaches are robbing businesses of their revenue and their reputation.

**In this article, we shall discuss 16 challenges often faced by testers while testing a web application.**

- Cross Browser Compatibility. ...

- Responsiveness. ...

- Cross-Device Compatibility. ...

- Integration Testing. ...

- Security. ...

- Performance Testing. ...

- Application Getting Slow. ...

- Usability Testing.

1. **What is beta testing ?**

-Beta testing is an opportunity for real users to use a product in a production environment to uncover any bugsor issues before a general release.

Beta Testing is performed by real users of the software application in a real environment. Beta testing is one of the types of User Acceptance Testing. A Beta version of the software, whose feedback is needed, is released to a limited number of end-users of the product to obtain feedback on the product quality.

1. **What is load testing ?**

-examines how the system behaves during normal and high loads and determines if a system, piece of software, or computing device can handle high loads given a high demand of end-users.

-When your software development project is nearing completion, there’s one test that’s essential to understanding its readiness for deployment: load testing. This type of [performance testing](https://www.microfocus.com/en-us/what-is/performance-testing) allows you to determine how your web application will behave during normal and peak load conditions, as well as its breaking point (should it occur below the peak load condition). At its core, load testing is used to confirm that your web application meets your intended performance goals or objectives, which are frequently identified in a service level agreement.

1. **When to used usablity testing ?**

-Here are the four major times to do usability testing.

Usability testing ideas. Once you've got an idea, conduct usability testing before putting any design resources to work. ...

Usability testing prototypes. ...

Usability testing before launch. ...

Usability testing after launc

\* Providing a good User Experience (UX) that guides users through the site and turns that traffic into conversions and/or sales is crucial for getting the most out of your web presence. During the design and development of a website, we work with the site’s users to find out everything we can about how they use the website, what issues or pain points they have, and what features they would like to see added to the site. The insights gained from these testing sessions is invaluable for determining how to meet users’ needs in a way that will convert traffic into sales.