**Software Testing Assignment**

**Module 2**

1. **What is Exploratory Testing?**

Ans. Exploratory testing” is the practice of allowing testers to enter a product and find bugs and errors without the help of a script or test cases. It encourages testers to think creatively and simulate real-world usage, mimicking the actions of end users. In the case of Global App Testing, testers are incentivized to find bugs which the QA team considers high or critical priority.

**2).What is traceability matrix?**

Ans. A Traceability Matrix is a document that maps or rather traces the relationship between two baseline documents. Here, one of the documents has the requirement specifications, whereas the other one has test cases. The Traceability Matrix is an essential document used during the [**software** **development lifecycle**](https://www.browserstack.com/guide/learn-software-development-process) of a product, and it ensures completeness and transparency of the underlying product. It is also called Cross Reference Matrix (CRM) or Requirement Traceability Matrix (RTM).

**3). What is Boundary value testing?**

Ans. 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. This testing process was introduced to select boundary values that came from the boundary based on the inputs at different ends of testing values. This black box testing strategy was introduced after equivalence class partitioning where the partition of classes takes place first followed by a partition at the boundaries.

**4).** **What is Equivalence partitioning testing?**

Ans. Equivalence partitioning is a black-box testing technique that allows testers to group input data into sets or classes, making it possible to reduce the number of test cases while still achieving comprehensive coverage. This technique is particularly useful when dealing with a large range of input values

**5).What is Integration testing?**

Ans. integration testing is a type of software testing where components of the software are gradually integrated and then tested as a unified group. Usually these components are already working well individually, but they may break when integrated with other components. With integration testing, testers want to find defects that surface due to code conflicts between software modules when they are integrated with each other.

**6. What determines the level of Risk?**

Ans. Risk should be evaluated at the Business Level, Technological Level, Project Level and Testing Level.

Risk also used to decide where to start and where more testing is needed

**7. What is Alpha Testing?**

Alpha testing is definitely performed and carried out at the developing organizations location with the involvement of developers.

* It is always performed by the developers at the software development site.
* Sometimes it is also performed by independent testing team.
* It is always performed in virtual Environment.
* It comes under the category of both White box testing and Black box testing

1. **. What is Beta Testing?**

* Beta testing is performed and carried out by users or you can say people at their own location and site using customer data
* It is always performed by the customer at their own site
* It is not performed by independent testing team
* Beta testing is always open to the market and public.
* It is performed in Real Time Environment
* It is only kind of Black box testing

**9). What is Component Testing?**

* Component testing- A minimal software item that can be tested in isolation. It means “A component is the smallest testable part of software.”
* Component testing – The testing of individual software components.
* Component testing is a level of software testing process where individual units/components of a software /system are tested. The purpose is to validate that each unit of the software performs as designed
* Component testing is the first level of testing and is performed to integration testing.
* Sometimes also known as unit testing, module testing or program testing

**10). What is Functional System Testing?**

* Functional system testing – A requirement that specifies a function that a system or system component must perform.

A requirement may exist as a text document or a model.

**11). What is Non-Functional Testing?**

* Non-Functional testing: Testing the attributes of a component or a system that do not relate to functionality, e.g. reliability, efficiency, usability, interoperability, maintainability and portability.
* Non-functional testing includes, but is not limited to, performance testing, load testing, stress testing, usability testing, maintainability testing, reliability testing and portability testing.
* It is the testing of “how” the system works. Non – Functional testing may be performed at all test levels.

**12).What Is GUI Testing?**

* Graphical User Interface (GUI) testing is the process of testing the system’s GUI of the system under test. GUI testing involves checking the screen with the controls like menus, buttons, icons and all type of bars - tool bar , menu bar , dialog boxes and windows etc.

**12). What is Adhoc testing?**

* **Ans.** Main aim of Adhoc testing is to find defects by random checking.
* Adhoc testing is informal testing type with an aim to break the system.
* It does not follow any test design techniques to create test case.
* In fact is does not create test case altogether.
* This testing is primarily performed if the knowledge of testers in the system under test is very high.
* Adhoc testing can be achieved with the testing technique called error guessing.

**13.) What is load Testing?**

* Load testing – It’s a performance testing to check system behavior under load Testing an application under heavy loads, such as testing of a website under a range of loads to determine at what point the system’s response time degrades or fails.
* Load testing is kind of performance testing which determines a system’s performance under real life load conditions. This testing helps determine how the application behave when multiple users access it simultaneously.

1. **What is Stress Testing?**

* Stress Testing – System is stressed beyond its specifications to check how and when it fails. Performed under heavy load like putting large number beyond storage capacity , complex database queries , continuous input to system or database load.
* Stress testing is done to make sure that the system would not crash under crunch situation.
* Stress testing is also known as endurance testing.

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

* White box testing: Testing based on an analysis of the internal structure of the component or system.
* Structure based testing technique is also known as “white box “or “ glass box “ testing technique because here the testers require knowledge of how the software is implemented , how it works.
* White box testing is also called glass testing or open box testing. In order to perform white box on an application, the tester needs to possess knowledge of the internal working of the code.

Types of coverage

* Statement coverage
* Decision coverage
* Condition coverage

**16). What is black box Testing? What are the different black box testing.?**

Ans.The technique of testing without having any knowledge of the interior workings of the application is Black box testing.

The tester is oblivious to the system architecture and does not have access to the source code. Typically, when performing a black box test, a tester interact with the system’s user interface by providing inputs and examining outputs without knowing how and where the inputs are worked.

Techniques of Black box testing

* Equivalence partitioning
* Boundary value analysis
* Decision Tables
* State transition testing
* Use case testing
* Other black box testing

**17). Mention what are the categories of defects?**

Ans. There are some categories which are as below

* Data quality/Database Defects
* Critical Functionality Defects
* Functionality Defects
* Security Defects
* User Interface Defects

In big bang integration testing all components and modules is integrated simultaneously, after which everything is tested as a whole.

Big bang testing has the advantages that everything is finished before integration testing starts.

The major disadvantage is that in general it is time consuming and difficult to trace the cause of failures because of this late integration.

**18).Mention what big bang testing is?**

* In big bang integration testing all components and modules is integrated simultaneously, after which everything is tested as a whole.
* Big bang testing has the advantages that everything is finished before integration testing starts.
* The major disadvantage is that in general it is time consuming and difficult to trace the cause of failures because of this late integration.

**19). What is the purpose of exist criteria?**

* Purpose of exit criteria is to define when we STOP testing either at the:
* End of all testing – i.e. product Go Live
* End of phase of testing (handover from system test to UAT)

**20). When should “Regression Testing “be performed?**

* Change in requirements and code Is modified according to the requirements
* New feature is added to the software
* Defect fixing
* Performance issue fix

**21). What is 7 key principles? Explain in detail?**

* Testing shows presence of Defects
* Exhaustive Testing is Impossible
* Early testing
* Defect Clustering
* The pesticide Paradox
* Testing is context dependent
* Absence of Errors Fallacy

**1. Testing shows presence of Defects:**

* Testing can show that defects are present, but cannot prove that there are no defects.
* Testing reduce the probability of undiscovered defects remaining in the software but. Even if no defects are found, it is not a proof of correctness.
* As we find more defects. The probability of undiscovered defects remaining in a system reduces

**2. Exhaustive Testing is Impossible:**

* Testing everything including all combinations of inputs and preconditions is not possible
* For example, in an application in one screen there are 15 input fields. Each having 5 possible values then to test all the valid combination you would need
* That is, we must Priorities our testing efforts using a Risk Based Approach

**3. Early testing:**

* Testing activities should start as early as possible in the software or system development life cycle. And should be focused on defined objectives
* Testing activities should start as early as possible in the development life cycle.

**4. Defect Clustering:**

* A small number of modules contain most of the defects discovered during pre-release testing. Or are responsible for the most operational failures
* Defects are not evenly spread in a system
* In other words, most defects found during testing are usually confined to small number of modules

**5. The pesticide Paradox:**

* If the same tests are repeated over and over again, eventually the same set of test case will no longer find any new defects
* To overcome this “pesticides paradox” the test cases need to be regularly reviewed and revised and new and different tests need to be written to exercise different parts of the software or system to potentially find more defects.

**6. Testing is context dependent:**

* Testing is basically context dependent.
* Testing is done differently context
* Different kinds of sites are tested differently

**7. Absence of Errors Fallacy:**

* If the system built is unusable and does not fulfill the user’s needs and expectations then finding and fixing defects does not help.
* Even after defects have been resolved it may still be unusable and/or does not fulfill the users need and expectations

**22). Difference between QA v/s QC v/s Tester**

Ans . QA (Quality Assurance)

* It is a subset of Software Test Life Cycle (STLC)
* Preventive activities
* Process oriented activities
* Focus on processes and procedures rather than conducting actual testing on the system

QC (Quality Control)

* QC can be considered as the subset of Quality Assurance
* It is a corrective process
* Product oriented activities
* Focuses on actual testing by executing software with intend to identify bug/defect through implementation of procedures and process

Tester

* Testing is the subset of Quality Control
* It is a preventive process
* Product oriented activities
* Focuses on actual testing

**23). Difference between smoke and sanity?**

**Ans . Smoke Testing**

Smoke testing is performed to ascertain that the critical functionalities of the program is working fine

* This testing is performed by the developers or testers
* Smoke testing is usually documented or scripted
* Smoke testing is a subset of Regression testing

**Sanity Testing**

* Sanity testing is done to check the new functionality / bugs have been fixed
* Sanity testing is usually performed by testers
* Sanity testing is usually not documented and is unscripted
* Sanity testing is a subset of Acceptance Testing

**24). Difference between verification and validation**

**Ans. Verification**

* The process of evaluating work products (not the actual final product) of a development phase to determine whether they meet the specified requirements for that phase
* To ensure that the product is being built according to the requirements and design specification. In other words, to ensure that work products meet their specified requirement
* Are we building the product right?

**Validation:**

* The process of evaluating software during or at the end of the development process. To determine whether it satisfies specified business requirements
* To ensure that the product actually meets the user’s needs and that the specification were correct in the first place in other word, to demonstrate that the product fulfills its intended use when placed in its intended environment Are we building the right product

**25). Explain types of performance testing.**

* Load testing
* Stress testing
* Endurance testing
* Spike testing
* Volume testing
* Scalability testing

**1. Load testing:**

It’s a performance testing to check system behavior under load testing an application under heavy loads such as testing of a web site under a range of loads to determine at what point the system’s response time degrades or fails

**2. Stress testing:**

Stress testing system is stressed beyond its specifications to check how and when it fails. Performed under heavy load like putting large number beyond storage capacity complex capacity queries, continuous input to system or database load.

**26**). **What are Error, Defects, Bug and Failure?**

* **“**A mistake in coding is called error, error found by tester is called defect, defect accepted by development team then it is called bug, build does not meet the requirements then it is failure”

**27). Difference between priority and severity**

* **Priority:**
* Priority is relative and business focused. Priority defines the order in which we should resolve a defect, should we fix it now or can it wait? This priority status is set by the tester to the developer mentioning the time frame to fix the defect. If high priority is mention then the developer has to fix it at the earliest. The priority status is set based on the customer requirements.
* **Severity:**
* Severity is absolute and customer focused. It is the extent to which the defect can affect the software In other words it defines the impact that given defect has on the system

**28). What is Bug Life Cycle?**

* **“A** computer bug is an error, flow, mistake, failure or fault in a computer program that prevents it from working correctly or produces an incorrect result. Bug arise from mistake and errors made by people, in either a program’s source code or its design.”
* When bug is discovered, it goes through several states and eventually reaches one of the terminal states where it becomes inactive and closed

**29). explain the difference between Functional testing and Non-functional testing**

* **Functional Testing :**
* Functional testing based on an analysis of the specification of the functionality of a component or system
* Functional testing is based on the Functions and features – may be applied at all test levels
* Functional testing verifies that each function of the software application operates in conformance with the requirement specification
* **Non – Functional testing:**
* Non – functional testing is the attributes of a component or system that do not relate to functionality e.g. reliability , efficiency , usability , interoperability , maintainability and portability
* Measuring the characteristics of the system/software that can be quantified on varying scale
* Non – functional testing includes, but is not limited to, performance testing, load testing, stress testing , usability testing , maintainability testing , reliability testing and portability testing.

**30). What is the difference between the STLC (Software Testing Life Cycle) and SDLC (Software Development Life Cycle)?**

* **Software Testing Life Cycle:**
* STLC is mainly related to software testing
* It focuses only on testing the software
* STLC involves only five phases or steps
* In STLC, less number of members are needed
* Goal of STLC is to complete successful testing of software
* **Software Development Life Cycle:**
* SDLC is mainly related to software development
* Beside development other phases like testing is also included
* SDLC involves total six phases or steps
* In SDLC, more number of developers are required for the whole process

**31). What is the difference between test scenario, test cases and test script?**

* **Test Scenario:**
* Is any functionality that can be tested
* Is derived from test artifacts like Business Requirements Specification (BRS) and Software Requirements Specification (SRS)
* Helps test the end to end functionality in Agile way
* Is more focused on what to test
* Test less time and fewer resources to create
* **Test Cases:**
* Is a set of actions executed to verify particular features or functionality
* Is mostly derived from test scenario
* Helps in exhaustive testing of an app
* Is focused on what to test and how to test
* Requires more resources and time
* **Test Script :**
* Is a set of instructions to test an app automatically
* Is mostly derived from test cases
* Helps to test specific things repeatedly
* Is focused on the expected result
* Requires less time for testing but more resources for scripts creating and updating

**32). What is priority?**

* Priority is Relative and Business focused. Priority defines the order in which we should resolve a defect. Should we fix it now or Can it wait? This priority status is set by the tester to the developer mentioning the same frame to fix the defect. If the high priority is mentioned then the developer has to fix it at earliest. The priority status to set based on the customer requirements.
* For example : If the company name is misspelled in the home page of the website then the priority is high and severity is high and severity is low to fix.

**33). What is severity?**

* Severity is absolute and customer focused. It is the extent to which the defect can affect the software. In other word it defines the impact that a given defect has on the system.
* For example : if an application of the 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 severe. So the severity is high but priority is low.

**34). Bugs categories are…**

* There are some categories
* Security
* Database
* Functionality (Critical/General)
* UI

**35). Advantages of Bugzilla**

* Open source, free bug tracking
* Automatic Duplicate bug detection
* Search option with advanced features
* File/Modify bugs by mail
* Move bugs between installs
* Multiple authentication methods
* Time tracking
* Automated bug reporting, has an API to interact with system

**36). What are the different Methodology in Agile Development Model ?**

**1. Scrum methodology:**

Scrum is a lightweight framework of Agile Project Management, it can be adopted to conduct iterative and all types of incremental projects.

Due to specific characteristics like simplicity, sustained productivity and strength for blending several underlying approaches adapted by other agile methods, Scrum has obtained popularity over the years.

2**. Kanban:**

Kanban is an eminently visual workflow management approach, that can be employed for visualizing and thoroughly maintaining the making of products, it focuses on continual delivery of the product , but is not making stress to the entire software development life cycle.

Similar to scrum, kanban is the process developed for supporting collaborative teamwork more effectively.

**37). Explain the difference between Authorization and Authentication in web testing. What are the common problems faced in web testing?**

|  |  |
| --- | --- |
| **Authentication** | **Authorization** |
| Authentication verifies who the user is. | Authorization determines what resources a user can access. |
| Authentication works through passwords, 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 an 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. |

**38). When to use Usability Testing?**

* Usability testing is a method of testing the functionality of a website, app or other digital product by observing real users as they attempt to complete tasks on it. The users are usually observed by researchers working for a business.
* Usability testing can and should be conducted on the current iteration of a product before beginning any new design work, after you have begun the strategy work around a brand new site or app.

**39). What is the procedure for GUI Testing?**

* Check all the GUI elements for size, position, width, length and acceptance of

Characters or numbers. For instance, you must be able to provide inputs to the input fields.

* Check you can execute the intended functionality of the application using the GUI.
* Check Error Messages are displayed correctly
* Check for Clear demarcation of different sections on screen
* Check Font used in application is readable
* Check the alignment of the text is proper
* Check the Color of the font and warning messages is aesthetically pleasing
* Check that the images have good clarity
* Check that the images are properly aligned
* Check the positioning of GUI elements for different screen resolution.

**Write a Scenario of Pen?**

Ans. Verify the type of pen, whether it is a ballpoint pen, ink pen, or gel pen.

1. Verify that the user is able to write clearly over different types of papers.
2. Check the weight of the pen. It should be as per the specifications. In case not mentioned in the specifications, the weight should not be too heavy to impact its smooth operation.

Verify if the pen is with a cap or without a cap.

Verify the color of the ink on the pen.

Check the odor of the pen’s ink on writing over a surface.

Verify the surfaces over which the pen is able to write smoothly apart from paper e.g. cardboard, rubber surface, etc.

Verify that the text written by the pen should have consistent ink flow without leaving any blob.

Check that the pen’s ink should not leak in case it is tilted upside down.

Verify if the pen’s ink should not leak at higher altitudes.

Verify if the text written by the pen is erasable or not.

Check the functioning of the pen by applying normal pressure during writing.

Verify the strength of the pen’s outer body. It should not be easily breakable.

Verify that text written by pen should not get faded before a certain time as mentioned in the specification.

Check if the text written by the pen is waterproof or not.

Verify that the user is able to write normally by tilting the pen at a certain angle instead of keeping it straight while writing.

Check the grip of the pen, and whether it provides adequate friction for the user to comfortably grip the pen.

Verify if the pen can support multiple refills or not.

In the case of an ink pen, verify that the user is able to refill the pen with all the supported ink types

10. For ink pens, verify that the mechanism to refill the pen is easy to operate.In the case of a ballpoint pen, verify the size of the tip.

11.Inthe case of a ball and gel pen, verify that the user can change the refill of the pen easily.

**Negative Test Scenarios for Pen:**

1. Verify the functioning of a pen at extreme temperatures – much higher and lower than room temperature.
2. Verify the functioning of a pen at extreme altitude.
3. Check the functioning of a pen at zero gravity.
4. Verify the functioning of the pen by applying extreme pressure.
5. Verify the effect of oil and other liquids on the text written with a pen.
6. Check if the user is able to write with a pen when used against gravity i.e. upside down.
7. Verify the functioning of a pen when a user tries to write on unsupported surfaces like glass, plastic, wood, etc.
8. Verify if the pen works normally or not when used after immersing in water or any other liquid for some period of time.

Write a Scenario of ATM?

1. Verify the type of ATM machine, if it has a touch screen, both keypad buttons only, or both.
2. Verify that on properly inserting a valid card different banking options appear on the screen.
3. Check that no option to continue and enter credentials is displayed to the user when the card is inserted incorrectly.
4. Verify that the touch of the ATM screen is smooth and operational.
5. Verify that the user is presented with the option to choose a language for further operations.
6. Check that the user is asked to enter a pin number before displaying any card/bank account detail.
7. Verify that there is a limited number of attempts up to which the user is allowed to enter the pin code.
8. Verify that if the total number of incorrect pin attempts gets surpassed then the user is not allowed to continue further. And operations like temporary blocking of the card, etc get initiated.
9. Check that the pin is displayed in masked form when entered.
10. Verify that the user is presented with different account type options like- saving, current, etc.
11. Verify that the user is allowed to get account details like available balance.
12. Check that the correct amount of money gets withdrawn as entered by the user for cash withdrawal.
13. Verify that the user is only allowed to enter the amount in multiple denominations as per the specifications.
14. Verify that the user is prompted to enter the amount again in case the amount entered is less than the minimum amount configured.
15. Check that the user cannot withdraw more amount than the total available balance and a proper message should be displayed.
16. Verify that the user is provided the option to get the transaction details in printed form.
17. Verify that the user’s session timeout is maintained.
18. Check that the user is not allowed to exceed one transaction limit amount.
19. Verify that the user is not allowed to exceed the one-day transaction limit amount.
20. Verify that the user is allowed to do only one transaction per pin request.
21. Check that in case the ATM machine runs out of money, a proper message is displayed to the user.
22. Verify that the applicable fee gets deducted along with the withdrawn amount in case the user exceeds the limit of the number of free transactions in a month.
23. Verify that the applicable fee gets deducted along with the withdrawn amount in case the user uses a card of a bank other than that of an ATM.
24. Check that the user is not allowed to proceed with the expired ATM card and that a proper error message gets displayed.
25. Verify that in case of sudden electricity loss before withdrawing cash, the transaction is marked as null and the amount is not withdrawn from the user’s account.

**Write a scenario of Microwave Owen?**

1. Verify that the dimensions of the oven are as per the specification provided.
2. Verify that the oven’s material is optimal for its use as an oven and as per the specification.
3. Verify that the oven heats the food at the desired temperature properly.
4. Verify that the oven heats food at the desired temperature within a specified time duration.
5. Verify the ovens functioning with the maximum attainable temperature.
6. Verify the ovens functioning with minimum attainable temperature.
7. Verify that the oven’s plate rotation speed is optimal and not too high to spill the food kept over it.
8. Verify that the oven’s door gets closed properly.
9. Verify that the oven’s door opens smoothly.

10. Verify the battery requirement of the microwave oven and check that it function’s smoothly at that power. Verify that the text written over the oven’s body is clearly readable.

11. Verify that the digital display is clearly visible and functions correctly.

12. Verify that the temperature regulator is smooth to operate.

13. Verify that the temperature regulator works correctly.

14. Check the maximum capacity of the oven and test its functioning with that volume of food.

15. Check the oven’s functionality with different kinds of food – solid, and liquid.

16. Check the oven’s functionality with different food at different temperatures.

17. Verify the oven’s functionality with different kinds of container material.

18. Verify that the power cord of the oven is long enough.

19. Verify that the usage instruction or user manuals have clear instructions.

**Write a scenario of Gmail Receiving Mail ?**

1. Verify that a newly received email is displayed as highlighted in the Inbox section.
2. Verify that a newly received email has correctly displayed sender email Id or name, mail subject and mail body(trimmed to a single line).
3. Verify that on clicking the newly received email, the user is navigated to email content.
4. Verify that the email contents are correctly displayed with the desired source formatting.
5. Verify that any attachments are attached to the email and are downloadable.
6. Verify that the attachments are scanned for viruses before download.
7. Verify that all the emails marked as read are not highlighted.
8. Verify that all the emails read as well as unread have a mail read time appended at the end on the email list displayed in the inbox section.
9. Verify that count of unread emails is displayed alongside ‘Inbox’ text in the left sidebar of Gmail.
   1. Verify that unread email count increases by one on receiving a new email.
   2. Verify that unread email count decreases by one on reading an email ( marking an email as read).
   3. Verify that email recipients in cc are visible to all users.

**Write a scenario of chair?**

1. Verify that the chair is stable enough to take an average human load.
2. Check the material used in making the chair-wood, plastic etc.
3. Check if the chair’s leg are level to the floor.
4. Check the usability of the chair as an office chair, normal household chair.
5. Check if there is back support in the chair.
6. Check if there is support for hands in the chair.
7. Verify the paint’s type and color.
8. Verify if the chair’s material is brittle or not.
9. Check if cushion is provided with chair or not.  
   1. Check the condition when washed with water or effect of water on chair.
   2. Verify that the dimension of chair is as per the specifications.
   3. Verify that the weight of the chair is as per the specifications.
   4. Check the height of the chair’s seat from floor.

**Write a Scenario of Wrist Watch**

1. Verify the type of watch – analog or digital.
2. In the case of an analog watch, check the correctness time displayed by the second, minute, and hour hand of the watch.
3. In the case of a digital watch, check the digital display for hours, minutes, and seconds is correctly displayed.
4. Verify the material of the watch and its strap.
5. Check if the shape of the dial is as per specification.
6. Verify the dimension of the watch is as per the specification.
7. Verify the weight of the watch.
8. Check if the watch is waterproof or not.
9. Verify that the numbers in the dial are clearly visible or not.

10.Check if the watch is having a date and day display or not.

11.Verify the color of the text displayed in the watch – time, day, date, and other information.

12.Verify that clock’s time can be corrected using the key in case of an analog clock and buttons in case of a digital clock.

13.Check if the second hand of the watch makes ticking sound or not.

14.Verify if the brand of the watch and check if its visible in the dial.

15.Check if the clock is having stopwatch, timers, and alarm functionality or not.

16.In the case of a digital watch, verify the format of the watch 12 hours or 24 hours.

17.Verify if the watch comes with any guarantee or warranty.

18.Verify if the dial has glass covering or plastic, check if the material is breakable or not.

19.Verify if the dial’s glass/plastic is resistant to minor scratches or not.

20. Check the battery requirement of the watch.

**Write a Scenario of Lift?**

1. Verify the dimensions of the lift.
2. Verify the type of door of the lift is as per the specification.
3. Verify the type of metal used in the lift interior and exterior.
4. Verify the capacity of the lift in terms of the total weight.
5. Verify the buttons in the lift to close and open the door and numbers as per the number of floors.
6. Verify that the lift moves to the particular floor as the button of the floor is clicked.
7. Verify that the lift stops when the up/down buttons on a particular floor are pressed.
8. Verify if there is an emergency button to contact officials in case of any mishap.
9. Verify the performance of the floor – the time taken to go to a floor.
10. Verify that in case of power failure, the lift doesn’t free-fall and gets halted on the particular floor.
11. Verify lifts working in case the button to open the door is pressed before reaching the destination floor.
12. Verify that in case the door is about to close and an object is placed between the doors if the doors sense the object and again open or not.
13. Verify the time duration for which the door remains open by default.
14. Verify if the lift interior is having proper air ventilation.
15. Verify lighting in the lift.
16. Verify that at no point the lift door should open while in motion.
17. Verify that in case of power loss, there should be a backup mechanism to safely get into a floor or a backup power supply.
18. Verify that in case the multiple floor number button is clicked, the lift should stop on each floor.
19. Verify that in case of capacity limit is reached users are prompted with a warning alert- audio/visual.
20. Verify that inside lift users are prompted with the current floor and direction information the lift is moving towards- audio/visual prompt.

Verify the type of door of the lift is as per the specification.

1. Verify the type of metal used in the lift interior and exterior.
2. Verify the capacity of the lift in terms of the total weight.
3. Verify the buttons in the lift to close and open the door and numbers as per the number of floors.
4. Verify that the lift moves to the particular floor as the button of the floor is clicked.
5. Verify that the lift stops when the up/down buttons on a particular floor are pressed.
6. Verify if there is an emergency button to contact officials in case of any mishap.
7. Verify the performance of the floor – the time taken to go to a floor.
8. Verify that in case of power failure, the lift doesn’t free-fall and gets halted on the particular floor.
9. Verify lifts working in case the button to open the door is pressed before reaching the destination floor.
10. Verify that in case the door is about to close and an object is placed between the doors if the doors sense the object and again open or not.
11. Verify the time duration for which the door remains open by default.
12. Verify if the lift interior is having proper air ventilation.
13. Verify lighting in the lift.
14. Verify that at no point the lift door should open while in motion.
15. Verify that in case of power loss, there should be a backup mechanism to safely get into a floor or a backup power supply.
16. Verify that in case the multiple floor number button is clicked, the lift should stop on each floor.
17. Verify that in case of capacity limit is reached users are prompted with a warning alert- audio/visual.
18. Verify that inside lift users are prompted with the current floor and direction information the lift is moving towards- audio/visual prompt.

**Write a scenario of Coffee vending Machine?**

1. Verify that the dimension of the coffee machine is as per the specification.
2. Verify that outer body, as well as inner part’s material, is as per the specification.
3. Verify that the machine’s body color as well brand is correctly visible and as per specification.
4. Verify the input mechanism for coffee ingredients-milk, water, coffee beans/powder, etc.
5. Verify that the quantity of hot water, milk, coffee powder per serving is correct.
6. Verify the power/voltage requirements of the machine.
7. Verify the effect of suddenly switching off the machine or cutting the power. The machine should stop in that situation and in power resumption, the remaining coffee should not get come out of the nozzle.
8. Verify that coffee should not leak when not in operation.
9. Verify the amount of coffee served in single-serving is as per specification.
10. Verify that the digital display displays correct information.
11. Check if the machine can be switched on and off using the power buttons.
12. Check for the indicator lights when the machine is switched on-off.
13. Verify that the functioning of all the buttons work properly when pressed.
14. Verify that each button has an image/text with it, indicating the task it performs.
15. Verify that complete quantity of coffee should get poured in a single operation, no residual coffee should be present in the nozzle.
16. Verify the mechanism to clean the system work correctly- foamer.
17. Verify that the coffee served has the same and correct temperature each time it is served by the machine.
18. Verify that system should display an error when it runs out of ingredients.
19. Verify that pressing the coffee button multiple times leads to multiple serving of coffee.
20. Verify that there is the passage for residual/extra coffee in the machine.
21. Verify that machine should work correctly in different climatic, moistures and temperature conditions.
22. Verify that machine should not make too much sound when in operation.
23. Performance test – Check the amount of time the machine takes to serve a single serving of coffee.
24. Check the performance of the machine when used continuously until the ingredients run out of the requirements.

1.[Negative Test](https://artoftesting.com/negative-testing) – Check the functioning of the coffee machine when two/multiple buttons are pressed simultaneously.

2.Negative Test – Check the functioning of coffee machine with a lesser or higher voltage than required.

3.Negative Test – Check the functioning of the coffee machine if the ingredient container’s capacity is exceeded.

1. Verify the type of watch – analog or digital.
2. In the case of an analog watch, check the correctness time displayed by the second, minute, and hour hand of the watch.
3. In the case of a digital watch, check the digital display for hours, minutes, and seconds is correctly displayed.
4. Verify the material of the watch and its strap.
5. Check if the shape of the dial is as per specification.
6. Verify the dimension of the watch is as per the specification.
7. Verify the weight of the watch.
8. Check if the watch is waterproof or not.
9. Verify that the numbers in the dial are clearly visible or not.
10. Check if the watch is having a date and day display or not.
11. Verify the color of the text displayed in the watch – time, day, date, and other information.
12. Verify that clock’s time can be corrected using the key in case of an analog clock and buttons in case of a digital clock.
13. Check if the second hand of the watch makes ticking sound or not.
14. Verify if the brand of the watch and check if its visible in the dial.
15. Check if the clock is having stopwatch, timers, and alarm functionality or not.
16. In the case of a digital watch, verify the format of the watch 12 hours or 24 hours.
17. Verify if the watch comes with any guarantee or warranty.
18. Verify if the dial has glass covering or plastic, check if the material is breakable or not.
19. Verify if the dial’s glass/plastic is resistant to minor scratches or not.
20. Check the battery requirement of the watch.

**2. Online shopping to buy product (flip kart)?**

|  |  |  |
| --- | --- | --- |
| Verify that the shopping cart icon is displayed as expected or not. |  | |
|  |  |  | |
| Verify that the shopping cart page UI should be as expected |  |  | |
| Verify that the user is able to add items into the cart or not |  |  | |
| Verify if the user increases the number of items then the cart should be reflected as per quantities |  |  | |
| Verify if the user decreases the number of items then the cart should be reflected  as per quantities |  |  | |
| Verify that the user is able to click on the cart icon button when the cart is empty |  |  | |
| Verify that when the user clicks on the cart button then the user should navigate to the shopping cart page |  |  | |
| Verify that the user is able to remove items from the cart |  |  | |
| Verify that the user is able to increase the item’s quantity |  |  | |
| Verify that the user is able to decrease item’s quantities |  |  | |
| Verify that the total price should be corrected |  |  | |
| Verify that shipping charges should be displayed as expected |  |  | |
| Verify that tax charges should be displayed as expected |  |  | |
| Verify that Proceed button should be working properly |  |  | |
| Verify if the user removes all items from the cart, the cart should be displayed as empty |  |  | |
| Verify that add maximum items into cart |  |  | |
| Verify if the user place an order successfully then the cart should be empty |  |  | |
| Verify that each items’ price on the shopping cart page |  |  | |
| Verify that all item’s total prices on the shopping cart page |  |  | |