MANUAL TESTING

**1. Format for writing the test case….?**

**There is no any generic format to write the test case but we follow this**

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Test Scenario** | **Test .No** | **Test Case** | **Test Case Navigation** | **Test Data** | **Priority** | **Severity** | **Excepted Result** | **Actual Result** | **Status** |

**2. How many test cases you can execute in a day?**

**A sample answer is “In my previous project, we generally execute 30-40 simple test cases (like login functionality) per day,**

**10-20 medium test cases (like Assigning user roles) per day, and**

**5-10 complex test cases (complete purchase flow) per day.**

**3.** **What is the difference between Quality Assurance v/s Quality Control in Testing?**

**Quality Assurance: Quality Assurance involves in process-oriented activities. It ensures the prevention of defects in the process used to make Software Applications. So the defects don’t arise when the Software Application is being developed.**

**Quality Control: Quality Control involves in product-oriented activities. It executes the program or code to identify the defects in the Software Application.**

**4.** **What is Verification in software testing?**

**Verification is the process, to ensure that whether we are building the product right i.e., to verify the requirements which we have and to verify whether we are developing the product accordingly or not. Activities involved here are Inspections, Reviews, Walk-through, requirements Analysis, etc (QA Activity)**

**5. What is Validation in software testing?**

**Validation is the process, whether we are building the right product i.e., to validate the product which we have developed is right or not. Activities involved in this is Testing the software application (QC Activity)**

**6.** **What is White Box Testing?**

**White Box Testing is also called as Glass Box, Clear Box, and Structural Testing. It is based on applications internal code structure. In white-box testing, an internal perspective of the system, as well as programming skills, are used to design test cases. This testing usually was done at the unit level and by Programmer .**

**Various white-box testing techniques are:**

1. **Statement Coverage**
2. **Decision Coverage**
3. **Condition Coverage**
4. **Multiple Condition Coverage**

**7. What is Black Box Testing?**

**Black Box Testing is a software testing method in which testers evaluate the functionality of the software under test without looking at the internal code structure. This can be applied to every level of software testing such as Unit, Integration, System and Acceptance Testing.**

**8. What is Grey Box Testing?**

**Grey box is the combination of both White Box and Black Box Testing. The tester who works on this type of testing needs to have access to design documents. This helps to create better test cases in this process.**

**9.** **What is Test Strategy?**

**Test Strategy is a high-level document (static document) and usually developed by the project manager. It is a document that captures the approach on how we go about testing the product and achieve the goals. It is normally derived from the Business Requirement Specification (BRS).**

**10. What is Test Plan and contents available in a Test Plan?**

**Test plan document is a document which contains the plan for all the testing activities to be done to deliver a quality product. Test Plan document is derived from the Product Description, SRS It is usually prepared by the Test Lead or Test Manager.**

**11. What is Test Closure?**

**Test Closure is the note prepared before test team formally completes the testing process. This note contains the total no. of test cases, total no. of test cases executed, total no. of defects found, total no. of defects fixed, total no. of bugs not fixed, total no of bugs rejected etc.,**

**12. What are the most common components of a defect report?**

|  |  |
| --- | --- |
| **Bug ID** | **B1** |
| Module | Personal details And Attachments |
| Requirements | Personal details page requirement |
| Test Case Name | TC4 |
| Release | QA |
| Version | NA |
| Status | New |
| Reporter | Login in person |
| Date | 9th may 22 |
| Assign To | Developer |
| CC | TL/PM |
| Severity | S2 |
| Priority | P2 |
| Server | QA |
| Platform | Web |
| Build No. | NA |
| Test Data | 1. eg:- 123 2. eg:- @#$ 3. eg:- @#$+abc+123 |
| Attachment | Attach screenshot |
| Brief Description | First name should not accept Wrong data |
| Navigation Steps | 1.Enter URL on browser 2.Go to username 3. click on My info 4. click on First Name box |
| Expected Result | 1. First name should not accept only digit 2.First name should not accept only Special chacter 3. First name should not accept combintion Special Chacter + alphabet + digit |
| Actual Result | 1. First name accept only digit 2.First name accept Special chacter 3. First name is accept combintion Special Chacter + alphabet + digit |
| Additional Comments | IF ANY |

**13.** **What are the levels of testing?**

**In software testing, there are four testing levels.**

1. **Unit Testing or component level testing**
2. **Integration Testing**
3. **System Testing**
4. **Acceptance Testing**

**14. What is Unit Testing?**

**Unit Testing is also called Module Testing or Component Testing. It is done to check whether the individual unit or module of the source code is working properly. It is done by the developers in the developer’s environment.**

**15.** **What is Integration Testing?**

**Integration Testing is the process of testing the interface between the two same or different software units. Integration testing is done in three ways. Big Bang Approach, Top-Down Approach, Bottom-Up Approach.**

**16. What is System Testing?**

**Testing the fully integrated application to evaluate the system’s compliance with its specified requirements is called System Testing AKA End to End testing. Verifying the completed system to ensure that the application works as intended or not**

**17.** **What is Big Bang Approach?**

**Combining all the modules once and verifying the functionality after completion of individual module testing.**

**18.** **What is Top-Down Approach?**

**Testing takes place from top to bottom. High-level modules are tested first and then low-level modules and finally integrating the low-level modules to a high level to ensure the system is working as intended. Stubs are used as a temporary module if low-level modules are not ready for integration testing.**

**19. What is Bottom-Up Approach?**

**Testing takes place from bottom to up. Lowest level modules are tested first and then high-level modules and finally integrating the high-level modules to a low level to ensure the system is working as intended. Drivers are used as temporary high-level modules for integration testing.**

**20.** **What is Functional Testing?**

**In simple words, what the system actually does is functional testing. To verify that each function of the software application behaves as specified in the requirement document. It falls within the scope of black box testing**

**Testing types are**

**• Unit testing**

**• Smoke testing**

**• User Acceptance**

**• Integration Testing**

**• Regression testing**

**• Localization**

**• Globalization**

**• Interoperability**

**21. What is Non-Functional Testing?**

**In simple words, how well the system performs is non-functionality testing. Non-functional testing refers to various aspects of the software such as performance, load, stress, scalability, security, compatibility etc., Main focus is to improve the user experience on how fast the system responds to a request.**

**Testing types are**

**• Performance Testing**

**• Volume Testing**

**• Scalability**

**• Usability Testing**

**• Load Testing**

**• Stress Testing**

**• Compliance Testing**

**• Portability Testing**

**• Disaster Recover Testing**

**22. What is Acceptance Testing?**

**It is also known as pre-production testing. This is done by the end-users or the customer (client) end testers to validate the functionality of the application.** **Types of acceptance testing are Alpha, Beta & Gamma.**

**23. What is Alpha Testing?**

**Alpha testing is done by the in-house developers (who developed the software) and testers before we ship the software to the customers. Sometimes alpha testing is done by the client or outsourcing team with the presence of developers or testers.**

**24.** **What is Beta Testing?**

**Beta testing is done by a limited number of end-users before delivery. It is done after the Alpha Testing. Usually, it is done in the client’s place.**

**25.** **What is Smoke Testing?**

**Smoke Testing is done to make sure if the build we received from the development team is testable or not.**

**Basically we testing key features of applications.**

**26. What is Sanity Testing?**

**Sanity Testing is done during the release phase to check for the main functionalities of the application without going deeper. It is also called as a subset of Regression testing.**

**Release phase: build after any modification or after any bug fixing**

**27. What is Regression Testing?**

**It is a deeper testing perform after any kind of changes made in application,**

**Purpose: we conform or validate the whatever change we made does its affects other functionality of the application or not..? irrespective of their positive or negative affect**

**Usually, we do regression testing in the following cases:**

1. **New functionalities are added to the application**
2. **Requirement Change frequently (In organizations, we call it as CR)**
3. **Defect Fixing**
4. **Performance Issue Fix**
5. **Environment change (E.g., Updating the DB from MySQL to Oracle)**

**28. What is Adhoc Testing?**

**Ad-hoc testing is quite opposite to the formal testing. It is an informal testing type. In Adhoc testing, testers randomly test the application without following any documents and test design techniques**

**29. What is Cross-Browser Testing?**

**Cross Browser Testing is a type of non-functional test which helps us ensure that our website or web application works well in various browsers.**

**30. What is Bug Severity?**

**Bug/Defect severity can be defined as how much effect will be there on the system because of a particular defect, It can be Critical, Major or Minor.**

**31. What is Bug Priority?**

**Defect priority can be defined as how soon the defect should be fixed. It gives the order in which a defect should be resolved. Developers decide which defect they should take up next based on the priority. It can be High, Medium or Low. Most of the times the priority status is set based on the customer requirement**

**32. What are the principles of Software Testing?**

1. **Testing shows presence of defects: Although testing can discover software defects, it cannot ensure that the product is free of errors. Testing may reduce the quantity of glitches, but it will never be able to eliminate them all.**
2. **Exhaustive testing is impossible: Since it’s impossible to test the software exhaustively, we can only run a few select tests and we assume that the software will always produce correct output. More comprehensive testing would be too costly and time-consuming.**
3. **Early testing: It’s crucial to test software early on to find defects. In the early stages of SDLC, it’s easier and less expensive to identify defects. Software testing should begin in the first phase of software development, during requirement analysis.**
4. **Defect clustering: According to the Pareto Principle, 80% of software defects arise from 20% of modules. In other words, most project defects are found in only a few sections of code.**
5. **Pesticide Paradox: If you want to find any new bugs, you can’t just keep running the same test cases repeatedly. You need to add or update your existing test cases.**
6. **Testing is context depending: Depending on the software development context, the testing approach will differ. Based upon its type, different software requires variations in testing.**
7. **Absence of error fallacy: Not only does the software need to be bug-free 99% of the time, but it must meet customer requirements if it is ever going to be used.**

**33. Tell some examples of Bug Severity and Bug Priority?**

1. **High Priority & High Severity: Submit button is not working on a login page and customers are unable to login to the application**
2. **Low Priority & High Severity: Crash in some functionality which is going to deliver after couple of releases**
3. **High Priority & Low Severity: Spelling mistake of a company name on the homepage**
4. **Low Priority & Low Severity: FAQ page takes a long time to load**

**34. What is the back box testing techniques..?**

1. **Boundary Value Analysis**
2. **Equivalence Class Partition**
3. **Decision Table testing**
4. **State Transition**

#### ****What is Boundary Value Analysis?****

#### ****Boundary value analysis (BVA) is based on testing the boundary values of valid and invalid partitions.****

#### ****What is Equivalence Class Partition?****

#### ****Equivalence Partitioning is also known as Equivalence Class Partitioning. In equivalence partitioning, inputs to the software or system are divided into groups in such way that it covers invalid + in process +invalid****

#### ****Ex :- input data in range of 17 to 25…..so Equivalence Class Partition would be like .. (invalid) 16 + ( in process ) 18 to 24 among any value + (invalid) 26****

#### ****What is Decision Table testing?****

#### ****In the Decision table technique, we deal with combinations of inputs. To identify the test cases with a decision table, we consider conditions and actions. We take conditions as inputs and actions as outputs.****

#### ****What is State Transition?****

#### ****Using state transition testing, we pick test cases from an application where we need to test different system transitions. We can apply this when an application gives a different output for the same input, depending on what has happened in the earlier state****

#### ****35. What is SDLC?****

#### ****Software Development Life Cycle (SDLC) aims to produce a high-quality system that meets or exceeds customer expectations****

#### ****Following are the stages in the SDLC.****

#### ****1. Requirements & Analysis: Identify Customer or project Requirements, Business analysts(BA) gather requirements from their customers, target market, and industry experts & Do feasible Study over it. to make BRS (Business Requirement Specification) Document****

#### ****2. Project Planning: Based on the BRS Document, The project may be focused on building a new software product or improving a current one.****

#### ****During this initial development phase, team members work together to discuss and plan out:****

#### ****Intentions behind the project (Goal of the Project)****

#### ****Requirements of the project****

#### ****Anticipated issues (Predicted problems With Solutions)****

#### ****Opportunities****

#### ****Risks****

#### ****All of these elements are recorded in a Software Requirement Specification (SRS) document. .****

#### ****3. Design: This stage focuses on designing the product. It involves product architects and developers who will come up with concepts and present a design of the product. They may present more than one design approach, and these ideas are documented in a Design Document Specification (DDS).****

#### ****4. Coding & Implementation: on this stage product will build by Programmer based on DDS****

#### ****5. Testing: in this stage developer’s test their Software code, and ensure that whatever they developed its fulfill customers requirement or not ,if they fulfill then the software should enter a QA process to validate the product’s quality.****

#### ****6. Deployment: In this stage once the software has undergone testing and QA, it is delivered to the customer.****

#### ****7. Maintenance: Because a software product’s usage varies from customer to customer (each person has different needs), there may be unique issues that come up and need to be addressed. These customer issues are solved in this maintenance stage.****

#### ****36**. **What are the stages in the Software Testing Lifecycle?****

#### ****Following are the stages in the STLC.****

#### ****1. Requirement Analysis: includes brainstorming sessions, identifying blind spots or unclear areas in the requirements, and clarifying those doubts from seniors and prioritizing certain assessments****

#### ****Activities in Requirement Phase Testing****

#### ****Identify types of tests to be performed.****

#### ****Gather details about testing priorities and focus.****

#### ****Prepare Requirement Traceability Matrix (RTM).****

#### ****Identify test environment details where testing is supposed to be carried out.****

#### ****Automation feasibility analysis (if required).****

#### ****Deliverables of Requirement Phase Testing****

#### ****RTM****

#### ****Automation feasibility report. (if applicable)****

#### ****2. Test Planning:** **it is a phase in which a Senior QA manager determines****

#### ****Test Planning Activities****

#### ****Preparation of test plan/strategy document for various types of testing****

#### ****Test tool selection****

#### ****Test effort estimation****

#### ****Resource planning and determining roles and responsibilities.****

#### ****Training requirement****

#### ****Deliverables of Test Planning****

#### ****Test plan /strategy document.****

#### ****Effort estimation document.****

#### ****3. Test Design& Development:****

#### ****The Test Case Development Phase involves the creation, verification and rework of test cases & test scripts after the test plan is ready.****

#### ****Test Case Development Activities****

#### ****Create test cases, automation scripts (if applicable)****

#### ****Review and baseline test cases and scripts****

#### ****Create test data (If Test Environment is available)****

#### ****Deliverables of Test Case Development****

#### ****Test cases/scripts****

#### ****Test data****

#### ****4. Test Environment Setup:****

#### ****Decides the software and hardware conditions required for testing process and can be done in parallel with the Test Case Development Phase.****

#### ****Test Environment Setup Activities****

#### ****prepare hardware and software requirement list for the Test Environment.****

#### ****Setup test Environment and test data****

#### ****Perform smoke test on the build****

#### ****Deliverables of Test Environment Setup****

#### ****Environment ready with test data set up****

#### ****Smoke Test Results.****

#### ****5. Test Execution:****

#### ****Test Execution Phase is carried out by the testers in which testing of the software build is done based on test plans and test cases prepared. The process consists of test script execution, test script maintenance and bug reporting. If bugs are reported then it is reverted back to development team for correction and retesting will be performed.****

#### ****Test Execution Activities****

#### ****Execute tests as per plan****

#### ****Document test results, and log defects for failed cases****

#### ****Map defects to test cases in RTM****

#### ****Retest the Defect fixes****

#### ****Track the defects to closure****

#### ****Deliverables of Test Execution****

#### ****Completed RTM with the execution status****

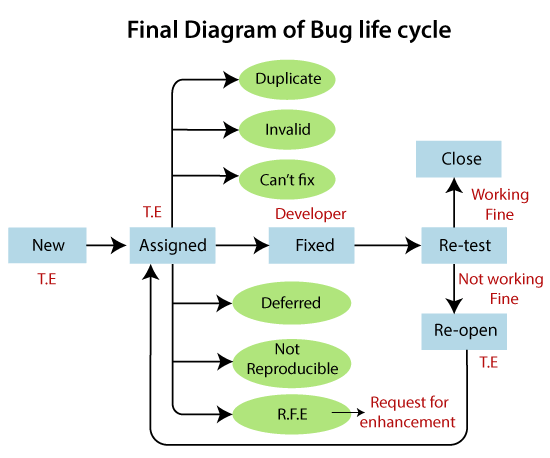
#### ****Test cases updated with results****

#### ****Defect reports****

#### ****6. Test Closure:****

#### ****It is phase is completion of test execution which involves several activities like test completion reporting, collection of test completion matrices and test results. Testing team members meet, discuss and analyze testing artifacts to identify strategies that have to be implemented in future, taking lessons from current test cycle. The idea is to remove process bottlenecks for future test cycles.****

#### 37.  ****What is Bug Life Cycle?****

****

**This is a time span between the defect is found and defect is closed. It includes below varies Status ,Which is given by Tester To developer or vice versa.**

**NEW (Tester):**

**As soon as the test engineer finds the bug, status is given as New, which indicates that a bug is just found.**

**ASSIGNED (Tester):**

**This new bug needs to be reported to the concerned Developer by changing the status as Assigned so that the responsible person should take care of the bug.**

**FIXED (Developer):**

**Then the Developer first go through the bug, which means that the Developers read all the navigation steps to decide whether it is a valid bug or not.**

**Based on this, if the bug is valid, the Developer starts reproducing the bug on the application, once the bug is successfully reproduced, the Developer will analyze the code and does the necessary changes, and change the status as Fixed.**

**RE-TEST (CLOSE (VERIFIED) + RE- OPEN) (Tester):**

**CLOSE (VERIFIED):**

**Once the code changes are done, and the bug is fixed, the test engineer re-test the bug, which means that the test engineer performs the same action once again, which is mentioned in the bug report, and changes the status Close, if the bug fixes properly, and functionally working according to the requirement.**

**OR**

**RE-OPEN:**

**if the bug still exists or not working properly as per the requirement, then the bug sends it back to the Developer once again.**

**This process is going on continuously until all the bugs are fixed and closed.**

**PENDING RETEST (Tester):**

**Once the defect is fixed the developer gives a particular code for retesting the code to the tester. Since the software testing remains pending from the testers end, the status assigned is "pending retest."(From tester side)**

**Following are the different status of the bug life cycle from developer side:**

**● Invalid/rejected**

**● Duplicate**

**● Postpone/deferred**

**● Can't fix**

**● Not reproducible**

**● RFE (Request for Enhancement)**

**REJECTED/INVALID (Developer):**

**If the developer feels the defect is not a genuine defect then it changes the defect to "rejected”.**

**DUPLICATE (Developer):**

**When the same bug has been reported multiple times by the different test engineers are known as a duplicate bug**

**DEFERRED / POSTPONED (Developer):**

**The deferred/postpone is a status in which the bugs are postponed to the future release due to time constraint**

**OR**

**If the present bug is not of a priority and if it is expected to get fixed in the next release, then status "Deferred"**

**CAN'T FIX (Developer):**

**When Developer accepting the bug and also able to reproduce, but can't do the necessary code changes due to some constraints.**

**NOT REPRODUCIBLE (Developer):**

**The Developer accepts the bug, but not able to Reproduce due to some reasons. These are the bug where the developer is not able to find it, after going through the navigation step given by the test engineer in the bug report.**

**RFE (REQUEST FOR ENHANCEMENT) (Developer):**

**These are the suggestions given by the test engineer towards the enhancement of the application in the form of a bug report. The RFE stands for Request for Enhancement**

**Not a bug: If it does not affect the functionality of the application then the status assigned to a bug is "Not a bug".**

**Whom to assign the bug**

**The bug can be assigned to the following:**

**● Developers**

**● Developers lead**

**● Test lead**

**Agile**

**----------------------------------------------------**

**1. What is Agile..?**

**Agile is a multi-iterative approach (in which the project is broken up into several smaller pieces) to project management and software development that helps teams deliver value to their customers faster and with fewer headaches**

**Each iteration which typically lasts from one to four weeks. The division of the entire project into smaller parts helps to minimize the project risk and to reduce the overall project delivery time.**

**Each iteration involves a team working through a full software development life cycle (SDLC) including planning, requirements analysis, design, coding, and testing before a working product is demonstrated to the client.**

**Advantages:**

* **Requirement changes are allowed at any stage of development.**
* **Release Will be very fast (Weakly).**
* **Customer no need to wait for long time bz We developed, Test and release piece of software to the customer with few numbers of features .**
* **Good Communication between ..customer +B.A + Developers + Tester**

**Dis-Advantages:**

* **Less focus on design and documentation**

**2. What are the principles of Agile Software Development?**

**1. Highest priority is to satisfy the customer through early and continuous delivery of business valuable software**

**2. Welcome changing requirements, even late in development**

**3. Deliver working software frequently**

**4. Business people and developers must work together with transparency on daily basis throughout the project**

**5. Build projects around motivated individuals**

**6. The best form of communication is to do face-to-face conversation**

**7. Working software is the primary measure of progress**

**8. Able to maintain a constant pace**

**9. Continuous attention to technical excellence**

**10. Simplicity – the art of maximizing the amount of work not done – is essential**

**11. Self-organizing teams**

**12. At regular intervals, the team reflects on how to become more effective, then tunes and adjusts its behavior accordingly**

**2. What is Scrum?**

**Scrum is a framework for project management that focused on teamwork, accountability and iterative progress toward a well-defined goal.**

**Scrum includes groups of people called as Scrum team. Normally contains 5-9 members.**

**1. Stakeholder:**

* **Customer/client**
* **He is a top most body in agile**

**2. Product Owner: Responsible for**

* **Gathering Requirements from Stakeholders**
* **He is team Member of Sprint planning**
* **Creating Product backlog and Sprint backlog**

**3. Scrum Master:**

* **he is responsible for maintain the process planning and utilize the resources to the project**
* **Clear any kind of obstacle/ issue arises during executing the sprint.**

**4. Development team:**

* **Responsible for full fill the customer requirement through code/ programming(developing the software)**
* **Follow the SDLC Methodology.**

**5. QA Team:**

* **Responsible for verify and validate Software /application against customer requirements and ensure that software is maximum bug or defect free**
* **Follow STLC And Defect life cycle**

**SCRUM TERMINOLOGY:**

**1. User Stories:**

**informal, natural language description of a feature of the software or product from the end-user perspective.(requirements).**

**2. Epic:**

**Collection of same kind of user-story.**

**3. Product Backlog:**

**It Contains total list of user-story of Application/software and prepaid by product owner.**

**4. Sprint Backlog:**

**It Contains Specific list of user-story of Application/software and prepaid by scrum master, product owner, and development team, Testing team should all collaborate to plan the sprint and make sure the tasks involved have realistic timeframes for development**

**Completion time of Each Sprint is 3 – 4 week depend on complicity and length of Sprint.**

**5. Sprint / Iteration:**

**It is Small part or feature of Application.**

**6. Sprint planning Meeting:**

* **It happed before the sprint backlog**
* **It is discussion among the Scrum team to decide what would be the Next Sprint from the product backlog**
* **Generally it select Top most High priority and severity first**

**7. Scrum Meeting:**

**It is basically 15 min meeting where Whole Agile Team is present along with product owner**

**And each team member is responsible to answer three Questions**

1. **What did you do yesterday..?**
2. **What are plan today’s task..?**
3. **Is there any obstacle (issue)..?**

**8. Sprint Retrospective Meeting:**

**It take after completion of each sprint where scrum team discus about the previous sprint**

**Like how it went ,any positive done, any negative done , any improvement, any suggestion for next sprint**

**9. Sprint Review Meeting:**

**It is an informal meeting held at the end of every Sprint, where the scrum team discus the feedback**

**3. What are the Scrum artifacts..?**

**User stories, Product Backlog, Sprint backlog, Burn down chart**

**4. What is the Burn down chart..?**

* **It is the graphical representation of Sprint work (task), it show how much amount of work is remaining**
* **Prepaid And maintain by Scrum master.**

**5. What are the Scrum or Agile ceremonies..?**

1. **Sprint planning**
2. **Daily Scrum meeting**
3. **Sprint review Meeting**
4. **Sprint retrospective meeting**

**6. Scrum board components..?**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Stories** | **To Do** | **In Process** | **Testing** | **Done** |

**7. When do we use Agile Scrum Methodology?**

1. **When the client is not so clear on requirements**
2. **When the client expects quick releases**
3. **When the client doesn’t give all the requirements at a time**

**8. What is the difference between Burn-up and Burn-down chart?**

**Burn Down Charts provides proof that the project is on track or not. Both the burn-up and burn-down charts are graphs used to track the progress of a project.**

**Burn-up charts represent how much work has been completed in a project whereas**

**Burn-down chart represents the remaining work left in a project.**

**AGILE ESTIMATION:**

**Agile estimation is the process for estimating the effort required to complete a prioritized task in the product backlog. This effort is usually measured with respect to how much time it will take to complete that task, Which is help us to do accurate sprint planning.**