19-11-2024(Tuesday)

K. Hema

Manual testing

The goal of a software tester is to find bugs, find them as early as possible, and make sure they get fixed.

**Software Testing** is an activity to detect and identify the defects in software which is a part of software development.

**Software Quality:** Software should be maintainable with bugs-free and within-budget. It should be delivered on time meeting all the requirements and expectations.

**Product vs Project:** If a software is built for a specific client according to requirement, then it is called Project. If a software is built for multiple clients according to market requirement, then it is called Product.

Error: A human action that produces problem in the system is called Error.

Defect: Deviation from the expected behaviour is called defect.

Failure: The deviation identified by the end user while using the application is called Failure.

Software has bugs normally due to mis-communications, Software complexity, programming errors, Changing requirements or due to lack of skilled testers.

Waterfall Model

Advantages:

* Quality of the product will be good.
* Since Requirement changes are not allowed, chances of finding bugs will be less.
* Initial investment is less since the testers are hired at the later stages.
* Preferred for small projects where requirements are feezed.

Disadvantages:

* Requirement changes are not allowed.
* If there is defect in Requirement that will be continued in later phases.
* Total investment is more because time taking for rework on defect is time consuming which leads to high investment.
* Testing will start only after coding.

Spiral Model

* Spiral Model is iterative model.
* Spiral Model overcome drawbacks of Waterfall model.
* We follow spiral model whenever there is dependency on the modules.
* In every cycle new software will be released to customer.
* Software will be released in multiple versions. So it is also called version control model.

Advantages of Spiral Model:

* Testing is done in every cycle, before going to the next cycle.
* Customer will get to use the software for every module.
* Requirement changes are allowed after every cycle before going to the next cycle.

Disadvantages of Spiral Model:

* Requirement changes are NOT allowed in between the cycle.
* Every cycle of spiral model looks like waterfall model.
* There is no testing in requirement & design phase.

Incremental Model: requirement is divided into multiple modules each module goes through SDLC phase

Using this model project will have a lengthy development schedule

Disadvantages: Need good planning, Total Cost is high.

WHY TESTING IS NECESSARY?

Developers assume that whatever they have developed is as per customer requirements and Will always work. Developers may have excellent skills of coding but integration issues can be present when different units do not work together, even though they work independently One must bring individual units together and make the final product, as some defects may be possible when the sources are developed by people sitting at different places. The primary role of software testing is not to demonstrate the correctness of software product, but to expose hidden defects so that they can be fixed. Testing is done to protect the common users from any failure of system during usage. Testing is a process of demonstrating that errors are not present in the product, this approach is used in acceptance testing where if the application meets acceptance criteria, then it must be accepted by the customer, Testing gives number of detects present which indirectly gives a measurement of software quality. More number of defects indicate bad software and bad processes of development, software bug occurs when

1. The software does not do something that the product specification says it should do.

2. The software does something that the product specification says it should not do.

3. The software does something that the product specification does not mention

**Causes of software bugs** are Specification, design, code and other

The bugs are caused for many reasons but main one is Specification. specifications are the largest bug producers because it may not be communicated well to the entire development team. Planning software is vitally important. If it is not done correctly, bugs will be created. The next largest source of bugs is the design if not designed well then bugs are occurred.

Types of Testing 20-11-2024

1.Manual testing: Manual testing includes testing a software manually, i.e., without using any automated tool or any script

2. Automation Testing: Automation testing, which is also known as Test Automation. The tester writes scripts and uses another software to test the product. This process involves automation of a manual process. Automation Testing is used to re-run the test scenarios that were performed manually, quickly, and repeatedly.

Static Testing Technique: In this phase analysis of a problem is done then early documentation will be prepared, which is verified and then design files will be created because most of the errors are found in design phase. It is not a formal process which is created by an author and he/she guide the participants according to their view and makes certain changes according to the feedbacks received from the participants.

Static technique has 3 types

Review: review is done before development with simple document and with complete Requirement review, design review, Test plan review. Review testing can be done by anyone either the manager, developer, or tester or even coworker etc.

Walk-thorough: It is informal which is done at any time or whenever required. Author of the document will explain it to their team.

Inspection: Most formal with 3- 8 people and with Proper meeting and schedule.

Dynamic Testing Techniques The process of evaluating a system or component based upon its behaviour during execution. A testing which is done after code development. The main purpose of dynamic testing is to test software behaviour with dynamic variables.

Levels Of Testing

1. Unit Testing: in unit testing individual component of software tested. The purpose of this testing is that each module is working properly It focuses on the smallest unit of software design (done by developer by using sample input and observing its sample output) E.g. In a program we are checking if loop, method, or function is working fine.

2. Integration Testing: in integration Testing individual units are combined and tested as group (developer). Main Purpose of Integration Testing is to check modules are communicating each other as DFD Data Flow Diagram.

3. System Testing: in this testing we can test whole application (complete / integrated software is tested) done by tester

4. Acceptance Testing: a level of software Testing in which software is tested for user acceptance UAT done at client location where software is used.

Alpha Testing: done by tester in company in presence of customer

Beta Testing: done by customer to check software is ok.

Testing Types

**Functional testing:** It is the process through which QAs determine if a piece of software is acting in accordance with pre-determined requirements. It uses black-box testing techniques, in which the tester has no knowledge of the internal system logic. Functional testing is only concerned with validating if a system works as intended. The goal of the functional testing is to ensure that software works according to specifications and user expectations.

1**. Non-Functional Testing:** Load testing, Reliability, The readiness of a system, Usability testing. A practical example would be checking how many people can simultaneously check out of a shopping basket

2. **Black Box Testing:** (without code) High level Black box testing is that kind of software testing you can do when you do not have the source code, just the executable code in hand The testing is done without the internal knowledge of the products.

3. **White box testing:** (with code) low level Monitoring internal structure, check internal logic done by developer.

4. **Smoke testing:** - It is first testing on newly released build. In this testing we need check the deployed software built is stable or not.

**Sanity Testing:** Testing the functionality once again

**Regression Testing** (Re- running if code changes): It is overall testing whenever new change is occurred. Every time a new module is added leads to changes in the program. This type of testing makes sure that the whole component works properly even after adding components to the complete program.

**Alpha Testing:** It is final testing in development. The main advantage is immediate solution is possible.

**Beta Testing:** It is 1st testing in client side. it is also called user acceptance testing UAT, but disadvantage is if defect is found there is no immediate solution.

**Installation Testing:** providing required resources at client location, it is type of testing in which test engineer check deployment process is successful as per user guideline

**Deployment document /user manual:** it is document prepared by project manager

**Usability Testing:** checking application for user friendliness

**Monkey Testing:** It is used for game testing, where it uses random input to check the application or system crash.

**Portability Testing:** Developed application Should support multiple environments.

**Forced error Testing:** To check valid error message.

**Exploratory Testing:** When test engineer does not have idea of functional testing then he will learn through exploring application.

**End to End Testing:** We can check all internal components for successful response Internal components like Client, Network, Server Database etc

**Security Testing:** Checking Security of application

**Reliability Testing:** The Developed application Should Support Longer Duration.

**Stability Audit:** it is independent evolution of software.

**Inspection:** it is formal evolution of software

**Concurrency Testing:** multiuser Testing

**Debugging:** executing program line by line for finding errors.

Types of System Testing:

**1. GUI Testing:** Testing GUI application, user interface testing-such as menu, check boxes, icon, images. Non-functional -we need to check size and position of element, image quality, spelling, alignment etc.

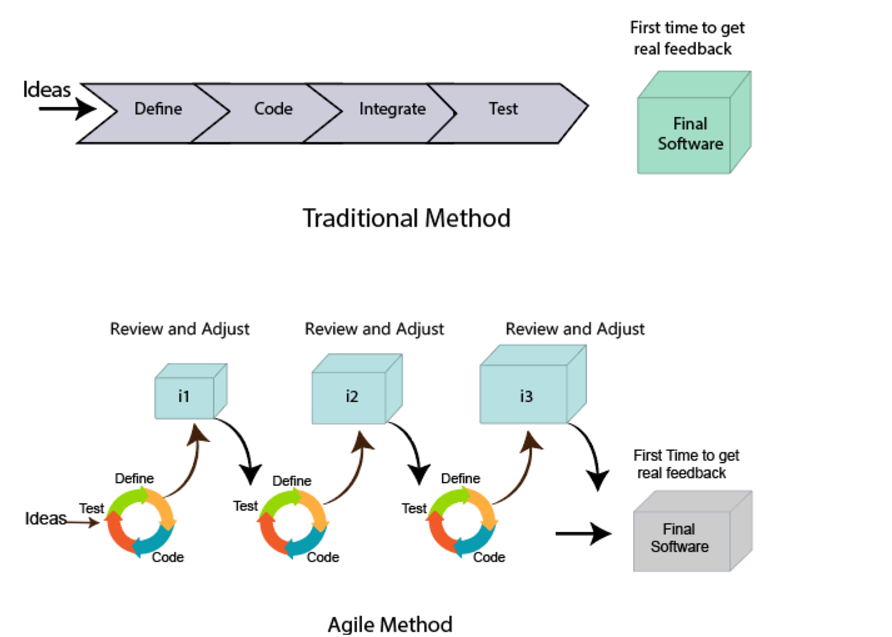
**2. Usability Testing:** check the easiness of application, by displaying messages if user confuse, and check if it is user-friendly application.

**Agile Methodology 21-11-2024**

What is Agile Methodology?

In English, Agile means ‘the ability to move quickly and easily’ and respond to change rapidly this is an important aspect of Agile software development.

Agile Software Development Methodology is a process of software development similar to other software development methodologies like waterfall models, V-models, iterative models, etc. Agile methodology follows the iterative as well as incremental approach that emphasizes the importance of delivering of working product very quickly.



An agile methodology is an iterative approach to software development. Each iteration of agile methodology takes a short time. The agile development process is aligned to deliver the changing business requirement. It distributes the software with faster and fewer changes.

What is a user requirement?

The user requirement defines the requirements of the user in terms of functionalities. There may be of two type of functionality.

* As a <User Role> I want <Functionality> so that <Business Value>
* In order to <Business value> as a <User Role> I want <Functionality>.

During software release planning, a rough estimate is given to a user requirement using relative scale points. During iteration planning, the requirement is broken down into tasks.

Relation between User requirement and Task

* User requirement talks about what is to be done. It defines the needs of users.
* Task talks about how it is to be done. It defines how functionality is implemented.
* User requirements are implemented by tasks. Every requirement is gathering as the task.
* User requirement is divided into different tasks when it is planned in current iteration.
* User tasks are estimated in hours based, generally it is between 2 to 12 hours.
* Requirements are validated using acceptance test.

**Cross-functional team**

Every agile team contains self-sufficient team with 5 to 9 team members. The average experience of each member ranges from 6 to 10 years. The agile team contains 3 to 4 developers, 1 tester, 1 technical lead, 1 scrum master and 1 product owner.

The Scrum master and Product owner are considered as a part of Team Interface, on the other hand remaining members are the part of Technical Interface.

Roles in Agile

There are two different roles in a Agile methodology. These are the Scrum Master and Product Owner.

**1. Scrum Master**

The Scrum Master is a team leader and facility provider who helps the team member to follow agile practices, so that the team member meets their commitments and customers requirements. The scrum master plays the following responsibilities:

* They enable the close co-operation between all the roles and functions.
* They remove all the blocks which occur.
* They safeguard the team from any disturbances.
* They work with the organization to track the progress and processes of the company.
* They ensure that Agile Inspect & Adapt processes are leveraged correctly which includes
  + Planned meetings
  + Daily stand-ups
  + Demo
  + Review
  + Retrospective meetings, and
  + Facilitate team meetings and decision-making process.

**2. Product Owner**

The Product Owner is one who runs the product from a business perspective. The Product Owner plays the following responsibilities:

* He defines the requirements and prioritizes their values.
* He sets the release date and contents.
* He takes an active role in iteration and releasing planning meetings.
* He ensures that the team is working on the most valued requirement.
* He represents the voice of the customer.
* He accepts the user stories that meet the definition of done and defined acceptance criteria.

In earlier times, iterative waterfall model was used to create software. But in today's time developers have to face many problems. The biggest problem is that in the middle of software development, the customer asks to make changes in the software. It takes a lot of time and money to make these changes.

So to overcome all these shortcomings, the agile model was proposed in the 1990s.

The agile model was created mainly to make changes in the middle of software development so that the software project can be completed quickly.

Advantages of Agile Model

• In this, two programmers work together due to which the code is error free and there are very few mistakes in it.

• In this the software project is completed in a very short time.

• In this the customer representative has an idea of each iteration so that he can easily change the requirement.

• This is a very realistic approach to software development.

• In this, focus is given on teamwork.

• There are very few rules in this and documentation is also negligible.

• There is no need for planning in this.

• It can be managed easily.

Disadvantages of Agile Model

• It cannot handle complex dependencies.

• Due to lack of formal documentation in this, there is confusion in development.

• It mostly depends on the customer representative, if the customer representative gives any wrong information then the software can become wrong.

• Only experienced programmers can take any decision in this. New programmers cannot take any decision.

• In the beginning of software development, it is not known how much effort and time will be required to create the software.

Why companies are shifting toward agile Software Development models?

Agile focuses on customer-centric approach that constantly take feedback from customer and make changes in the middle of software developer according to their needs and expectation. Agile Methodologies like scrum adopt iterative and incremental approach this accelerate the time-to-market for new features and product.

Conclusion

Choosing the right software development model is crucial for the success of a project. Each model offers unique advantages and is suitable for different types of projects. Agile methodologies, due to their flexibility and customer-centric approach, are increasingly popular in the industry. By adopting Agile, companies can accelerate time-to-market and better adapt to changing requirements, leading to higher customer satisfaction and successful project outcomes.

In earlier times, iterative waterfall model was used to create software. But in