**Service based company:**

* Here customer will give the requirement, a person from the company called as BA (Business Analyst) will perform the requirement collection from the customer
* Company will develop the software and test the software and gives the software to respective customer who has given the requirement
* Here the software company don’t have any rights to keep source code with them and the software company don’t have any right to sell the same software to different customer.
* The company will get the projects from the customers.

**Product based company:**

* Initially the customer will not be present . A person called as PA (Product analyst) will perform market research and gather the requirement.
* Here the company will develop the software and test the software and sell the software to multiple customers. Later the customer while using the software if they want any changes, then they will give requirement to the company, then the company will develop the changes and test the changes gives the software to the customer by charging some amount .
* The company will have all the rights to keep source code with them .
* The company will develop the product based on the client requirement.

**Key Process Areas(KPA):**

* Each of the KPA defines the basic requirements that should be met by a software process in order to satisfy the KPA achieve that level of maturity.
* Conceptually, key process area form the basis for management control of the software project and establish a context in which technical methods are applied , work products like models, documents, data, reports, etc. are produced milestones are established, quality is ensured and change is properly managed.

**Primary Process:**

Once the client or users asked to develop some project they will be giving the details in that they will mention what are the requirements for them ?

The process of starting the execution is nothing but the primary process.

**Conceptualization and Initiation:**

Once we get the project definition then we have to think how we can do this project and what are the application requirements for the project

Once we have decided that these are the software requirements we are going to use By using these concepts how best we can design our application and give to the client.

**Requirement Gathering:**

It is nothing but collecting the requirement from the customer, we call it as Customer requirement specification(CRS).

**Functional Specification:**

It is the detailed document which explains how each and every features or functionality should works.

**Use case modelling:**

It is a graphical notation to communicate more clearly than natural languages and codes.

Represents an action that accomplishes some sort of task within the system.

**Development and Implementation:**

* **Analysis:** This is the stage where the company will undergo several considerations in order to develop the software which the customer has given the requirements, so the company will cross check whether we have sufficient resources or not?, whether we have sufficient lab setups or not?, and they will also cross check if we invest on this project do we get the profits, and they will decide which good model to follow to develop the software.
* **Design:** Once after the analysis stage is completed hen the company will go for design phase , it is done by the Architectures wherein they will look into the requirements and start designing the software generally the software will have 2 types of designs

1. High level design(HLD)
2. Low level design(LLD)

* **Coding:** Once after the design phase is completed the company will go for coding wherein the developers will look into both requirements and design and start developing the software by writing the code.
* **Testing:** After completing coding the developers will give the software to the test engineers, the test engineers will take up the software and start testing the functionality of the software by execution the test cases.

**Deployment and operation:**

Once after the software is tested then it will be deployed into the production server which is present in the customer place, wherein a person from the company will move to the customer place and setup the environment and deploy the software into production server so that the customer can use the software and run their business.

**Maintenance:**

Once the software installation is completed and customer while using the software if they find any defect or problem in this case they will communicate to the company,

in software company they will freely service or they will give free support for fixing the defect for some time period based on the agreement between the company and the customer. This agreement period is called as maintenance period.

**Support Process**

**Documentation**

This will be done after the completion of the product building. This includes requirements , define , design , building , testing, deployment will undergo to documentation process.

Considering the documents in conceptualization and initiation it will be having details like what are the different types of softwares we are discussing and what type of software we have finialised to use in the project these are some of the concepts mentioned in the documents Likewise for every stage we will maintain the document so that it will become easy for the followup of the process and there will be a documentation support.

**Configuration Management**

* Configuration management determines clearly about the items that make up the software or system. These items include source code, test scripts, third-party software, hardware, data and both development and test documentation.
* Configuration management is also about making sure that these items are managed carefully, thoroughly and attentively during the entire project and product life cycle.
* Configuration management has a number of important implications for testing. Like configuration management allows the testers to manage their Test ware and test results using the same configuration management mechanisms.
* Configuration management also supports the build process, which is important for delivery of a test release into the test environment. Simply sending Zip archives by e-mail will not be sufficient, because there are too many opportunities for such archives to become polluted with undesirable contents or to harbor left-over previous versions of items. Especially in later phases of testing, it is critical to have a solid, reliable way of delivering test items that work and are the proper version.
* Last but not least, configuration management allows us to keep the record of what is being tested to the underlying files and components that make it up. This is very important

**Quality Assurance:**

Good developers will self-check for bugs and errors while coding, but even the best developer can miss an issue This happens most often when dealing with complex lines of code.

* To the developer, their coding's intention makes perfect sense and works for them, but the user does not have the same amount of knowledge or experience with the program. This may result in a program that doesn't perform the way the developer intended. This is why quality assurance becomes a necessary part of the development process.
* The role of quality assurance in software development is to detect and report issues in a program. This helps prevent issues, bugs, or failures when the software is released to the public QA happens before a final product is created and focuses on prevention. Some people confuse quality assurance with quality control, but they are actually two different things. While both quality assurance and quality control focus on catching any bugs or issues in a program, quality assurance focuses on completed sections of the larger project, and quality control focuses on the entire completed project.

**Verification:**

In the verification, the documents will be verified and verifying the document is to check whether it is according to the customer requirement.

Here we will check the correctness and completeness of the document. It is done before the software is developed.

**Validation:**

In the validation, the software will be tested and by testing the functionality of the software we will check whether the software is working as expected or not.

It is done once after the software is developed.

**Joint Review:**

A joint review requires that the dossier be submitted to all participating regulatory authorities simultaneously. This is a formal process in which timelines and work allocation are negotiated in advance.

**Audit:**

Audit means an independent examination of a software product or processes to assess compliance with specifications, standards, contractual agreements, or other criteria

**Organizational Process:**

Includes activities that establish the business goals of the organization and develop process, product and resource assets which, when used will help to achieve business goals. Managerial processes, resource and infrastructure processes are all in organizational process category.

**Project Management:**

Project management is the application of processes, methods, skills, knowledge and experience to achieve specific project objectives according to the project acceptance criteria within agreed parameters. Project management has final deliverables that are constrained to a finite timescale and budget.

**Process Improvement:**

Software Process Improvement (SPI) methodology is defined as a sequence of tasks, tools, and techniques to plan and implement improvement activities to achieve specific goals such as increasing development speed, achieving higher product quality or reducing costs.

**Infrastructure:**

Infrastructure are the components required to operate and manage enterprise IT environments. ... These components include hardware, software, networking components, an operating system (OS), and data storage, all of which are used to deliver IT services and solutions.

**Training/Skill Development:**

Skill Development Training. Skill development is a process of identifying your skill gap and ensuring you develop these skills. Your skills determine your ability to execute plans and achieve your goals. Skill development and training is an integral part of an employee's life

**Software Development Life Cycle(SDLC)**

**SDLC MODEL:** SDLC is a step by procedure or standard procedure to develop a new software.

When a person or a company want to start with new project they should follow SDLC.

**WHY SDLC?**

>We will never get to know how many engineers are required to work on the project.

>We will never get to know how much cost should be invested to work on project.

>There might not be proper supporting requirement documents.

>Chances are there we might delay in releasing in software to the customer.

**Stages of SDLC:**

1.Requirement collection

2.Feasibily study

3.Design

4.Coding

5.Testing

6.Installation

7.Maintanence

**Types of SDLC:**

1.Waterfall model

2.Spiral model

3.V and V model

4.Prototype model

5.Hybrid model

6.Derived model

7.Agile model

**Example for Prototype model:**

**Project:** **Facebook application**

**Requirements:**

Facebook is the customer who wants to extend their business into sales and they want it to be added as a feature in Facebook application, but they don’t have particular business requirements because they are new to the domain, so that they have approached a software company, here the software company takes up the project and follows the Prototype model as technology.

**Why Prototype Model?:**

Here we chosen Prototype model because customer is not aware of the business and don’t have particular business requirement, so software company can’t assume the customers requirements, so that the company will follow prototype model, so that they can prepare the dummy model and showcase the model to the customer. Customer might get some ideas while reviewing the prototype and they might give some inputs to the company, so that it will be easy for company to develop a software according to the customers requirements.

**Advantages of Prototype Model:**

* This model is flexible in design.
* It is easy to detect errors.
* We can find missing functionality easily.
* There is scope of refinement, it means new requirements can be easily accommodated.
* It can be reused by the developer for more complicated projects in the future.
* It ensures a greater level of customer satisfaction and comfort.
* It is ideal for online system.
* It helps developers and users both understand the system better.
* Integration requirements are very well understood and deployment channels are decided at a very early stage.
* It can actively involve users in the development phase.

**AGILE MODEL**

**What is agile model:**

Agile model is an iterative and incremental approach where requirements keeps on changing as a company we will be flexible enough to take up the requirement changes and develop the changes test the changes and give the quality piece of software to the customer within short span of time is called as Agile Model.

In agile there will be good communication between customer, BA, Developer, Test engineer wherein if the customer has made any change then we can directly contact the customer and understand the changes and develop the change and test the changes and release the quality software to the customer within short span of time

Main goal of agile is customer satisfaction through quick delivery of working piece of software.

**Teams involved in agile model:-**

Project team, customer, product-owners, stakeholders.

**Project team**:

These team responsible for design and product delivered time

**Customers:**

* They judge the features to be completed based on priority
* They will select the product and accept the product using user acceptance

testing.

**Product owners:**

* Act as mediator between customer and company.
* He is responsible for collect the requirement or input from the end-

user, customer, project team and stake holders.

* He is responsible for preparing the product blockage.

**Difference between waterfall model and Agile model**

|  |  |
| --- | --- |
| Waterfall | Agile |
| It is a linear and sequential | It is a incremental and iterative |
| It follows independent approach | It follows a whole team approach |
| BA, developers, testers are involved | Domain experts, developers, testers are involved |
| Delivering the product once | Delivering the product in every stage |

**Agile Testing:**

Testing the software by following the principles of agile is called as Agile testing.

**Types of Agile Model:**

* Scrum model
* Extreme Programing (XP)
* Feature driven development/ Test driven development.
* Crystal clear
* Lean and Kanban
* Dynamic system development method(DSDM)
* Adaptive software development method(ASDM)

**Scrum model/Scrum process:**

Scrum model is a standard procedure or step by step procedure or a framework which helps team to work together and develop a new software.

**Scrum Meetings/Scrum ceremonies:**

**Sprint planning:**

It is an meeting conducted by scrum master on the first day of every sprint.

Wherein in this meeting Developer, Test engineer, scrum master, BA, sometimes PM will be present.

BA will explain the complete requirement to the entire team, wherein BA will explain how each and every features should works and also BA will explain the customer business work flows to the entire team. Scrum master will prioritize the stories from the customer business point of view and scrum master will assign the task to each and every individual Engineers based on their available hours.

**Scrum Meeting:**

It is a meeting conducted by the scrum master on daily basis, this meeting is strictly bounded for 15 minutes. In this meeting BA, Developer, Test engineer, Scrum master, PM (sometimes) will be present

In this we will be discussing below mentioned 3 points

* What did you do yesterday?
* What will you do today?
* Are they any obstacles or Impendent.

**Sprint Retrospective meeting:**

This meeting is conducted by the scrum master on the last day of the sprint . . In this meeting BA, Developer, Test engineer, Scrum master, PM (sometimes) will be present

In this we will be discussing below mentioned 3 points.

* What went well? (Achievements)
* What didn’t went well? (Mistakes)
* What are the action plans?

**Release Retrospective Meeting:**

This meeting is conducted by the scrum master on the last day of the Release . . In this meeting BA, Developer, Test engineer, Scrum master, PM (sometimes) will be present.

In this we will be discussing below mentioned 3 points.

* What went well? (Achievements)
* What didn’t went well? (Mistakes)
* What are the action plans?

**Bug Triaging/ Defect triaging Meeting:**

This meeting is conducted by the Test engineer/BA/scrum master a week before the Release or few days before the sprint . In this meeting BA, Developer, Test engineer, Scrum master will be present.

In this meeting we will collect list of all the pending and open defects which are not fixed by the developer in current and previous release/sprint. As a team we will be re-prioritizing the defect from the customer business point of view and decide which defect should be fixed in the current release/ sprint.

**Pending defects:**

These are the defects which are not fixed by the developer in the previous release.

**Open defects:**

These are the defects which are not fixed by the developers in the current release.

**Product Backlog Grooming/Meeting/Sprint Backlog Grooming:**

This meeting is conducted by the Scrum master a week before the Release or few days before the sprint . In this meeting BA, Developer, Test engineer, Scrum master will be present.

Here entire team will gather the list of pending stories which are not implemented in the current release/sprint. As a team we will be Re-Prioritize the stories from the customer business point of view and we will decide which story should be developed in the current release/sprint and which stories should be moved to the upcoming release/sprint.

**Agile model**

Design

Develop Sprint 1

Test

Design

Develop Sprint 2

Test

**Advantages of Agile :**

* Customer can change the requirement at any stage of the development
* There will be good communication between customer ,BA, Developer, Test engineer.
* Releases will be short.
* It is very simple model to follow.
* Teams will be self Organized.

**Drawbacks of Agile:**

* Less Scope for design and documentation.
* If the customer is unaware about the requirements changes chances are there we might mess-up the project.
* Experience resources are required.

**Fundamentals of software testing:**

**Why Testing is Necessary:**

Human Testing is essential because we all make mistakes. Some of those mistakes are not important, but some are expensive or could be life-threatening. We have to test everything that we produce because thing can go wrong, human can make at any time. Error can cause a mistake or failure at any stage of SDLC. Testing will be helpful.

* To identify defects
* To reduce the mistakes in the components or system.
* Increase the overall quality of the system.
* The testing is important since it discovers defects/bugs before the delivery to the client, which guarantees the quality of the software.
* It makes the software more reliable and easier to use.
* Thoroughly tested software ensures reliable and high-performance software operations.

**Basic concepts of Testing**

**Software Testing**

* It is a process of identify or catching the defects in the software for all possible conditions is called Software testing.

**Why software Testing?**

* Software testing is a method of determining whether the actual software products meets the expected requirements and ensuring that the software products is free of defects.
* When a software development project is going on, you need to know that errors may appear in any phase of the life cycle.

**Importance of Software Testing:**

* Software testing is important because if there are any bugs or errors in the software they can be identified early and fixed before the software products is delivered.
* A properly tested software products ensure dependability, security, and high performance, which leads to time savings, cost effectiveness, and customer satisfaction.
* Ensures that the software developed is according to the customer requirements or not.
* Ensures that the software is error free .
* Ensures that the software works as expected by the customer in all possible ways such as stability, response time, capacity.

**Quality Assurance:**

Quality assurance is the way of preventing mistakes and defects in the manufactured products and avoid problems when delivering products or services to customers.

* Purpose of QA is to provide confidence to the customers by constant delivery of products according to specification.
* It includes procedures and tools applied by the software professionals to ensure that software meet the specified standard for its intended use and performance.

**Bug Free Software:**

* Bug it nothing but the defects which are accepted by the developers, due to which produces an incorrect or unexpected results.
* Bug free software provides proper and accurate visibility of the software project.
* Bug free software means it will deliver the products without any defects.

**Goal of Software testing :**

* The goal of software testing is to find errors, gaps, or missing requirements in comparison to the actual requirements.
* To deliver the product effectively to customers without any errors.

**Testing v/s Quality:**

**Testing:** It represents the operations -levels activity of actually checking the software for error and bugs.

**Usability Testing:** Testing the user-friendliness of an application.

**Compatibility Testing:** Testing the functionality of an application in different configurations or platform.

**Functional Testing:** Testing each and every component of an application rigorously or thoroughly according to the customer requirement specifications.

**Performance testing:** Testing the stability and response time of an application by applying the load.

**Quality:**

Quality is the degree/state to which the product has to meet in order to satisfy the requirements of the customer.

**Defect Prevention:**

* Defects prevention is a process to identify the root cause of the defects and establish the process to avoid defects.
* By focusing on the Defects prevention, we can work on improving the quality of the software.

**How much testing is enough?**

* **The level of risk**

Technical

Safety

Business Risks

* Project Constraints

Time

Budget

**Types of Testing:-**

1.Functional Testing

2.Non Functional Testing

**Functional Testing**

Testing each and every parameter, functional follow of the application that is complete functionality of the application is called Functional testing.

**Need of Functional testing**

* Functional testing ensures that software is working properly as per the customer requirement specification.

**Non functional testing**

Testing the stability, appearance, capacity, performance and security of the application and it also defines how the software operates .

**Need of Non functional testing**

* Non Functional testing ensures that software is performing as expected by the customer.

**Performance testing**:

* Testing the stability and response time of an application by applying load is called as performance testing.
* Performance testing is used to measure processing speed, response time, resource consumption, throughput, efficiency.

There are 3 types of performance testing

1. **Load test**:

* Testing the stability and response time of an application by applying load up to and including limits.
* Load tests refers to the load size and related values.

1. **Stress test**:

* Testing the stability and response time of an application by applying load which is more than designed number of users.
* It determines the applications ability to handle large amounts of data.

**Testing techniques:**

* **Blackbox testing**:
* Verifying the functionality of an application against requirement specification is called as black box testing.
* It is done by test engineers.
* It ignores the internal mechanism of a system and focuses only on the output generated.

**Advantages :**

Programming knowledge is not required.

Test engineers and developers are independent each other.

Tastings are done from users point of view.

**Disadvantages :**

Testing only the small number of inputs so that time taken will be more.

It is hard to design test cases without knowing about requirement specifications.

Unnecessary repetition of test inputs.

Many programs paths are remain untested.

* **White box testing(Structural testing)**
* Testing each and every line of the code is called as white box testing.
* It is done by developers.
* Here they test internal logic of the code.
* By using test cases they determines appropriate output.
* Here code is visible and readable.

**Advantages:**

* It helps in optimizing the code.
* Very easy to find out which type of input will help for testing

**Disadvantages**:

* Skilled tester is needed so it increase the cost.
* It is hard to look into every bit of code so which leads to failure of the application
* **Positive testing**: Testing the software by providing valid data.
* **Negative testing**: Testing the software by providing invalid data.
* **Static testing/Dry run testing:**
* These testing is used by developers
* It is done before compilation.
* It involves code review, Inspection and walkthrough.
* Manually read the code and syntax checking to find the errors are methods of static testing.
* **Dynamic testing:**
* It involves giving input values to the software and checking whether the output is as expected.
* It takes place after compilation.
* It helps to find the defects.