**Software Testing Assignment**

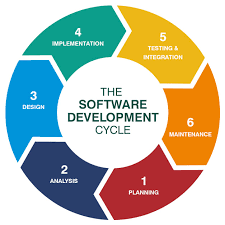
**Module-1(Fundamental)**

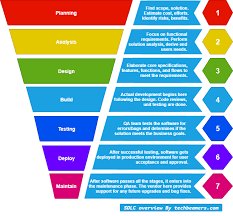
* **What is SDLC?**

SDLC stands for software development life cycle.

SDLC is a structure imposed on the development of a software product that defines the process for planning, implementation, testing, documentation, deployment and ongoing maintenance and support.

1. **Planning**
2. **Analysis**
3. **Design**
4. **Implementation/ Coding**
5. **Software Testing**
6. **Deployment**
7. **Maintenance**





**1.Planning**

* A high-level plan is being worked out with a business intent to take care the resources required for creating, modifying, or upgrading a service or the solution.
* The term resources refer to the HW/SW requirements, costs, time, benefits, and a few other essential items.
* QA (quality assurance) activity identification, project risk assessment happen in this phase.

**2.Analysis (Requirement Gathering)**

* The software development team works to carry on the project.
* The team holds discussions with various stakeholders for problem domain and tries to bring out as much info as possible on their requirements.
* The requirements are collected using a number of practices as given-

-studying the existing or obsolete system and S/W

-Conducting interviews of users and developers

-referring to the database

-collecting answers from the questionnaires.

* Requirements definitions usually consist of **natural language**, supplemented by **diagrams** and **tables**.

Types of Requirements: -

**Functional Requirements**: describe system **Services** or **functions.**

**Non-Functional Requirements:** describe **constraints** on the system or the development process.

* **Functional Requirement for WhatsApp: -** 
  + Login page (with mobile number)
  + User registration
  + Last seen
  + Send message/receive message
  + Scan QR code
  + Camera
  + Adding new contacts
  + Send attachments (photo, video, contact, document, location, audio, poll of question)
  + Message status
  + Broadcast message
  + End to end encryption
  + Voice and video call
  + User banning and reporting
  + Notification
* **Non-Functional Requirements for WhatsApp: -**
  + Privacy
  + Scalability
  + Robustness
  + Performance
  + Attractive UI
  + Security

**3.Design**

* in this 3rd phase, software design documents are prepared as per the requirements specification document.
* This helps define overall system architecture.
* 2 kind of design documents developed in this phase:

High-Level Design (HLD)

* Brief description and name of each module,
* An outline about the functionally of every Module.
* interface relationship and dependencies Between modules.
* Database tables identified along with their key elements.
* complete architecture Diagrams along with technology details.

Low-Level Design (LLD)

* Functional logic of the modules
* Database tables, which include type and size
* Complete detail of the interface
* Addresses all types of dependency issues
* Listing of error messages
* Complete input and outputs for every module

**4.Implementation/Coding**

* This stage has many names such as the Build, Development, Coding, or implementation phase.
* Developers start build the entire system by writing code using the chosen programming language.
* Tasks are divided into units or modules and assigned to the various developers.
* Longest phase of the Software Development Life Cycle process.
* Developer needs to follow certain predefined coding guidelines and to use programming tools like compiler, interpreters, debugger to generate and implement the code.

**5.Software Testing**

* QA(Quality Assurance)/ validation phase testing is defined as an activity to check whether the actual results match the expected results and to ensure that the software system is Defect free.
* The testing team starts testing the functionally of the entire system.
* QA and testing team may find some bugs/defects which they communicate to developers.
* The development team fixes the bug and send back to QA for a re-test.
* This process continues until the software is bug-free, stable, and working according to the business needs of that system.

**6.Deployment**

* Acceptance or Beta evaluation phase
* It includes a release specifically for market-facing group of people and gets it tested in real-time environment for their acceptance .it is a sort of **User Acceptance Testing(UAT)**
* Focuses on fixing some usability bugs or enhancements crucial for the market perspective or can also give a green status for delivering it to the target customers.

**7.Maintenance**

* After the software clears all the SDLC phases without any issues, then it goes into the maintenance stage.
* It allows the customers to request for upgrades and get the fixes/patches for problems internally or externally identified.
* Maintenance is the process of changing a system after it has been deployed.

-Corrective maintenance: identifying and repairing defects

-Adaptive maintenance: adapting the existing solution to the new platforms.

-Perfective Maintenance: implementing the new requirements.

* Following 3 activities occur: -

**Bug fixing**: bugs are reported because of some scenarios which are not tested at all

**Upgrade:** Upgrading the application to the newer versions of the Software

**Enhancement:** Adding some new features into the existing software.

* **What is software testing?**

Testing is the process of evaluating a system or it’s components with the intent to find that whether it satisfies the specified requirements or not.

This activity results in the actual, expected and difference between their results.

In simple words : testing is a executive a system in order to identify any gaps, errors or missing requirements in contrary to the actual desire or requirements

According to ANSI/IEEE 1059 standard, Testing can be defined as A process of analysing a software item to detect the differences between existing and required conditions (that is defect/errors/bugs) and to evaluate the features of the software item.

Software testing is a process of executing a program or application with the intent of finding the software bugs.

* **What is agile methodology?**
  + Agile SDLC model is a combination of iterative and incremental process model with focus on process adaptability and customer satisfaction by rapid delivery of working software product.
  + Agile method breaks the product into small incremental builds.
  + These builds are provided in iterations.
  + Each iteration typically lasts from about one to three weeks.
  + Every iteration involves cross function teams working. Simultaneously on various areas like planning, requirements analysis, design, coding, unit testing, and acceptance testing.
  + At the end of the iteration a working product is displayed to the customer and important stakeholders.
  + Agile model believes that every project needs to be handled differently and the existing methods need to be tailored to best suit the project requirements. In agile the tasks are divided to time boxes (small time frames) to deliver specific features for a release.
  + Iterative approach is taken and working software build is delivered after each iteration. Each build is incremental in terms of features; the final build holds all the features required by the customer.
  + Agile thought process had started early in the software development and started becoming popular with time due to its flexibility and adaptability.

**Pros: -**

* + - * Is a very realistic approach to software development
      * Promotes teamwork and cross training.
      * Functionality can be developed rapidly and demonstrated.
      * Resource requirements are minimum.
      * Suitable for fixed or changing requirements
      * Delivers early partial working solutions.
      * Good model for environments that change steadily.
      * Minimal rules, documentations easily employed.
      * Enables concurrent development and delivery within an overall planned context.
      * Little or no planning required.
      * Easy to manage.
      * Gives flexibility to developers.

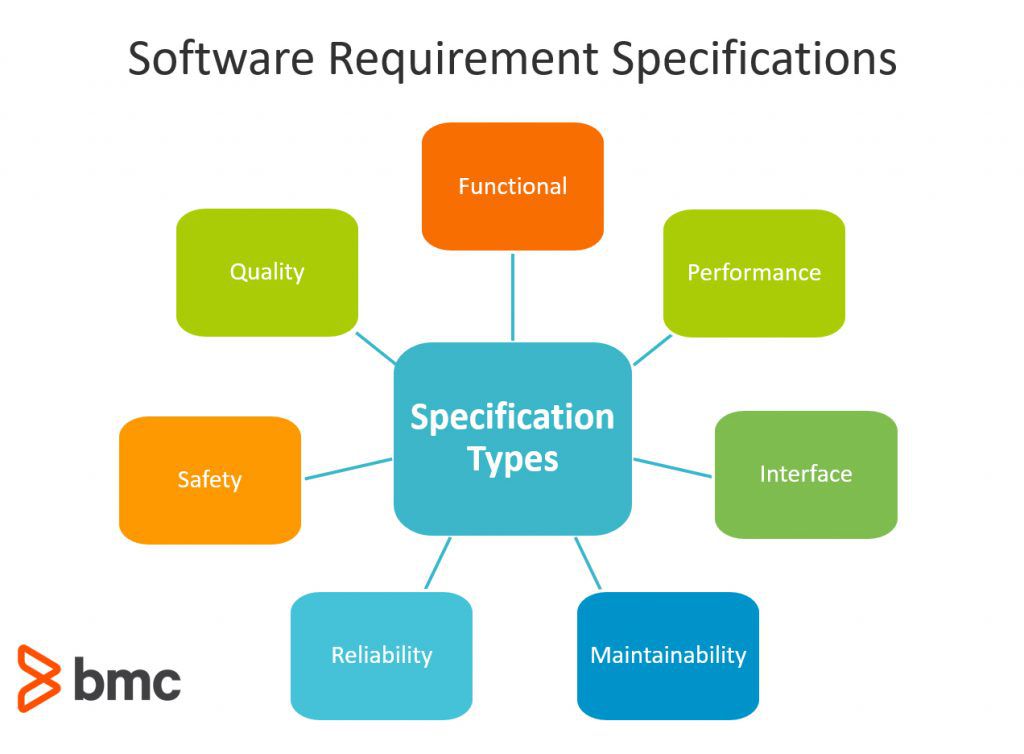
**Cons: -**

* + - * Not suitable for handling complex dependencies.
      * More risk of sustainability, maintainability, and extensibility.
      * An overall plan, an agile leader and agile PM practice is must without which it will not work.
      * Strict delivery management dictates the scope, functionality to be delivered, and adjustment to meet the deadlines.
      * Depends heavily on customer interaction, so if customer is not clear, team can be driven in the wrong direction.
      * There is very high individual dependency, since there is minimum documentation generated.
      * Transfer of technology to new team members may be quite challenging due to back documentation.

* **What is SRS**

A software requirements specification (SRS) is a complete description of the behaviour of the system to be developed.

It includes a set of use case(functional) and non-functional requirements that describe all of the interactions that the users will have with the software.



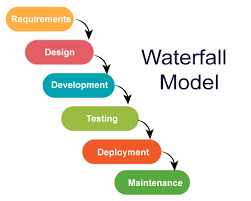
Types of Requirements in SRS

Customer Requirements

Functional Requirements

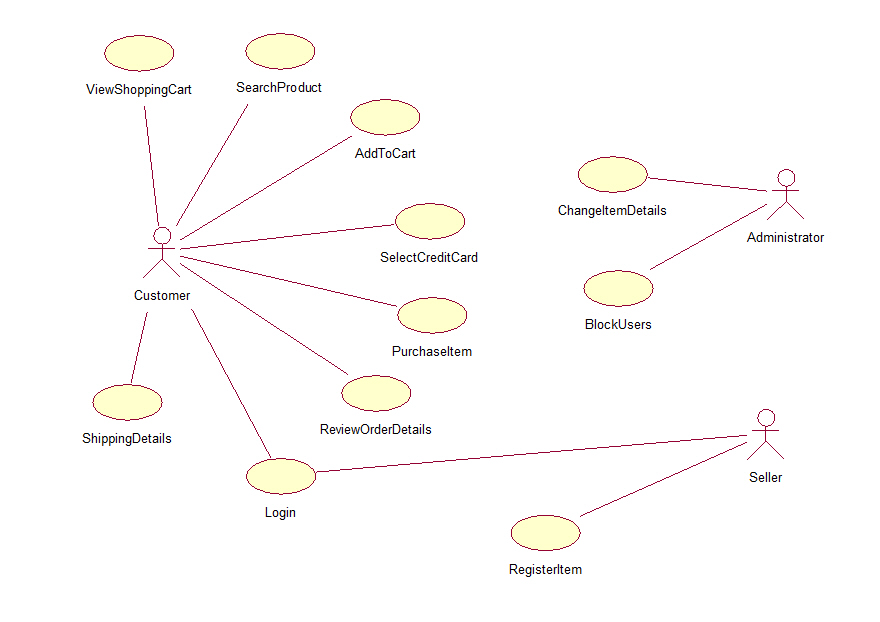
Non-Functional Requirements.

* **Write SDLC phases with basic introduction**

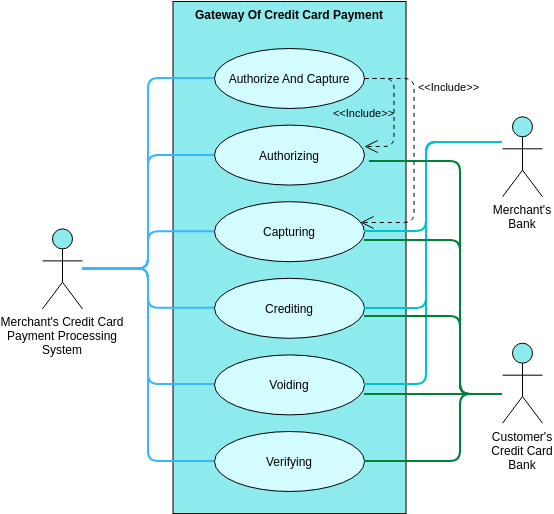


* **Requirement Gathering and analysis** − All possible requirements of the system to be developed are captured in this phase and documented in a requirement specification document.
* **System Design** − the requirement specifications from first phase are studied in this phase and the system design is prepared. This system design helps in specifying hardware and system requirements and helps in defining the overall system architecture.
* **Implementation** − with inputs from the system design, the system is first developed in small programs called units, which are integrated in the next phase. Each unit is developed and tested for its functionality, which is referred to as Unit Testing.
* **Integration and Testing** − All the units developed in the implementation phase are integrated into a system after testing of each unit. Post integration the entire system is tested for any faults and failures.
* **Deployment of system** − Once the functional and non-functional testing is done; the product is deployed in the customer environment or released into the market.
* **Maintenance** − There are some issues which come up in the client environment. To fix those issues, patches are released. Also to enhance the product some better versions are released. Maintenance is done to deliver these changes in the customer environment.

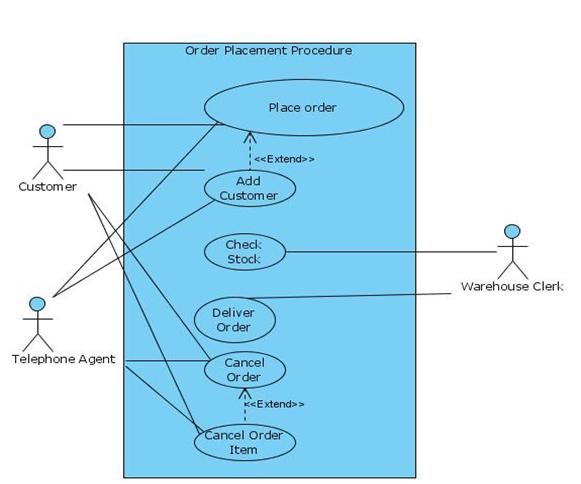
* **Draw Use case on Online book shopping**



* **Draw Use-case on online bill payment system (Paytm)**



* **Draw use case on Online shopping product using COD.**



* **Draw use case on Online shopping product using payment gateway.** 