Software Development Life Cycle or Software Development Process

Software Development Life Cycle is a systematic approach to develop software. It is a Process followed by Software Developers and Software Testing is an integral part of Software Development, so it is also important for Software Testers…

Software Development Life Cycle (SDLC) is a process used by the software industry to design, develop and test software. The SDLC aims to produce a high-quality software that meets or exceeds customer expectations, reaches completion within times and cost estimates.

**Phases of Software Development Life Cycle,**

These phases may vary form one organization to another, but purpose is almost all same, that is “Develop and Maintain Quality Software”,

1) Requirement Gathering  
2) Analysis  
3) Design  
4) Coding / Development  
5) Testing  
6) Deployment & Maintenance

Note: It is General Software Development Life Cycle, we have various SDLC Models in the IT Industry, Waterfall Model, V Model, Spiral Model and Agile Development Models etc…, Software Development process varies from one SDLC Model to another.

1) Requirement Gathering

Requirement Gathering is the most important phase in software development life cycle, Business Analyst collects the requirements from the Customer/Client After series of meeting with customer and releases Business Requirement Specification and provides the same to Development Team.

Note: Document name may vary from one Organization to another; some examples are Customer Requirement Specification (CRS), Business Requirement Document (BRD) etc…

Suppose Our Planned Software is not intended for a single customer and the software product for multiple customers then Business Analyst or Business Team collects Requirements from the Market and also evaluate other similar products in the Market

Key Role in this phase is Business Analyst and Outcome of the phase is “Business Requirement Specification”

The gathered requirements are of two types

* Functional Requirements(Entering username for login)
* Non-Functional Requirements(software should withstand 500 simultaneous customers without server going down or performance degrade)

2) Analysis

Once the Requirement Gathering is done the next step is to define and document the product requirements and get them approved by the customer. This is done through SRS (Software Requirement Specification) document. SRS consists of all the product requirements to be designed and developed during the project life cycle.

Key people involved in this phase are Project Manager, Business Analyst and senior members of the Team.   The outcome of this phase is Software Requirement Specification.

3) Design

Non functional requirements needs to be architected multiple high end web servers and database are need and hardware like RAM processor and software tools are decided on these requirements.

Module leads helps architects in designing the software architecture for the above nonfunctional requirements

In Design phase Senior Developers and Architects, they give the architecture of the software product to be developed. It has two steps one is HLD (High Level Design) or Global Design and another is LLD (Low Level Design) or Detailed Design,

The system is then analyzed in functionality perspective for example multiple pages must be required for the website each page would have related pages grouped together like page for showing bank balance, page for showing fixed deposits balance, page for showing user profile, page for showing bill payments. Here pages are different but for related functionality these can be called as modules. Modules may interact with each other for example home page might show summary from each module and contains links to various pages

High Level Design (HLD) is the overall system design, covers the system architecture and database design.  It describes the relation between various modules and functions/interfaces of the system.

Each module may contain smaller units or functionality for example a method can represent a unit or functionality the programmers may refer to HLD to come up with such units and their algorithms, flowcharts, data flow diagrams, class diagrams

Etc this is called Low Level Design (LLD) or the detailed system design, covers how each and every feature in the product should work and how every component should work.

The module lead and architect lead will review HLD, LLD and approve it or track it for further changes

The outcome of this phase is High Level Document and Low Level Document which works as an input to the next phase coding.

#### 4) Coding / Development

Developers (seniors, juniors, fresher) involved in this phase, this is the phase where we start building the software and start writing the code for the product.

The outcome of this phase is Source Code Document (SCD) and the developed product.

#### 5) Testing

Once the software is complete then it is deployed in the testing environment. The testing team starts testing (either test the software manually or using automated test tools depends on process defined in STLC)

Testing is done to verify that the entire application works according to the customer requirement.

During this phase, testing team may find defects which they communicate to developers; the development team fixes the defect and send back to Testing for a re-test. This process continues until the software is Stable, and working according to the business needs of that system.

#### 6) Deployment & Maintenance

After successful testing, the product is delivered (deployed to the customer for their use), Deployment is done by the Deployment/Implementation engineers and Once when the customers start using the developed system then the actual problems will come up and needs to be solved from time to time.

Fixing the issues found by the customer comes in the maintenance phase. 100% testing is not possible – because, the way testers test the product is different from the way customers use the product. Maintenance should be done as per SLA (Service Level Agreement)