The software development life cycle (SDLC) is a framework defining tasks performed at each step in the software development process.

SDLC consists of following activities: -

Requirements Gathering: A Software Requirement Specification or SRS is a document which records expected behaviour of the system or software which needs to be developed.

Design: Software design is the blueprint of the system, which once completed can be provided to developers for code development. Based on the components in design, they are translated into software modules/functions/libraries, etc… and these pieces together form a software system.

Coding: During this phase, the blueprint of the software is turned to reality by developing the source code of the entire application. Time taken to complete the development depends on the size of the application and number of programmers involved.

Testing: Once the application development is completed, it is tested for various issues like functionality, performance, and so on. This is to ensure that the application is performing as expected. If there are any issues, these issues are fixed before/after going to production depending on the nature of issue and the urgency to go live for the application.

Deployment: Once the application is ready to go live, it is deployed on a production server in this phase. If it is developed for a client, the deployment happens in a client premise or datacentre where there client wants to get the application installed.

There are several software development models followed by various organizations:

Waterfall Model: This model involves finishing each phase completely before commencing the next one. When each phase is completed successfully, it is reviewed to see if the project is on track and whether it is feasible to continue.

V-Shaped Model: This model focuses on the execution of processes in a sequential manner, similar to the waterfall model but with more importance placed on testing. Testing procedures are written even before the commencement of writing code. A system plan is generated before starting the development phase.

Incremental Model: This life cycle model involves multiple development cycles. The cycles are divided up into smaller iterations. These iterations can be easily managed and go through a set of phases including requirements, design, implementation and testing. A working version of the software is produced during the first iteration, so working software is created early in the development process.

What is S-SDLC?

S-SDLC stresses on incorporating security into the Software Development Life Cycle.

Each phase of the Sample SDLC is mapped with security activities, as demonstrated in the figure and as explained below:

Requirements Gathering:-

- Security Requirements
- Setting up Phase Gates
- Risk Assessment

Design:-

- Identify Design Requirements from security perspective
- Architecture & Design Reviews
- Threat Modelling

Coding:-

- Coding Best Practices
- Perform Static Analysis

Testing:-

- Vulnerability Assessment
- Fuzzing

Deployment:-

- Server Configuration Review
- Network Configuration Review