**STUDENT 2469MD**

**SDLC**

**Software Development Life Cycle (SDLC)**

The software development life cycle (SDLC) is a model, used in software development, to describe the stages and tasks involved in each step of a project to write and deploy software.



A software development life cycle is the long-term view of software as a product, from initial planning through maintenance and eventual retirement of the completed application. Organizations strive to quickly and efficiently produce high-quality software through intricate planning, commonly referred to as SDLC *phases* or *steps*.

**Standard SDLC phases**

The number of phases in an SDLC can vary depending on the business and its software product goals, typically falling between five and seven steps. At the most basic level, software is designed, developed, tested and then released.

There are following phases in every Software development life cycle model:

1. Requirement gathering and analysis
2. Design
3. Implementation or coding
4. Testing
5. Deployment
6. Maintenance

1) **Requirement gathering and analysis:** Business requirements are gathered in this phase. This phase is the main focus of the project managers and stake holders. Meetings with managers, stake holders and users are held to determine the requirements like; Who is going to use the system? How will they use the system? What data should be input into the system? What data should be output by the system? These are general questions that get answered during a requirement gathering phase. After requirement gathering these requirements are analyzed for their validity and the possibility of incorporating the requirements in the system to be development is also studied.

Finally, a Requirement Specification document is created which serves the purpose of guideline for the next phase of the model. The testing team follows the Software Testing Life Cycle and starts the Test Planning phase after the requirements analysis is completed.

2) Design: In this phase the system and software design are prepared from the requirement specifications which were studied in the first phase. System Design helps in specifying hardware and system requirements and helps in defining overall system architecture. The system design specifications serve as input for the next phase of the model.

In this phase the testers come up with the Test strategy, where they mention what to test, how to test.

3) Implementation / Coding: On receiving system design documents, the work is divided in modules/units and actual coding is started. Since, in this phase the code is produced so it is the main focus for the developer. This is the longest phase of the software development life cycle.

4) Testing: After the code is developed it is tested against the requirements to make sure that the product is solving the needs addressed and gathered during the requirements phase. During this phase all types of functional testing like unit testing, integration testing, system testing, acceptance testing is done as well as non-functional testing are also done.

5) Deployment: After successful testing the product is delivered / deployed to the customer for their use.

As soon as the product is given to the customers they will first do the beta testing. If any changes are required or if any bugs are caught, then they will report it to the engineering team. Once those changes are made or the bugs are fixed then the final deployment will happen.

6) Maintenance: Once when the customers start using the developed system then the actual problems come up and needs to be solved from time to time. This process where the care is taken for the developed product is known as maintenance.

