**SDLC BLOG POST**

1. **What is the software development life cycle?**

* SDLC or the Software Development Life Cycle is a process that produces software with the highest quality and lowest cost in the shortest time possible.
* SDLC provides a well-structured flow of phases that help an organization quickly produce high-quality software which is well-tested and ready for production use.
* Building models, material to use etc.
* Customer requirement analysis.

## **How was the SDLC created?**

In the 1950s and 1960s, computer science progressed rapidly. This swift evolution sparked the beginnings of a production framework that eventually grew into the SDLC we know today.

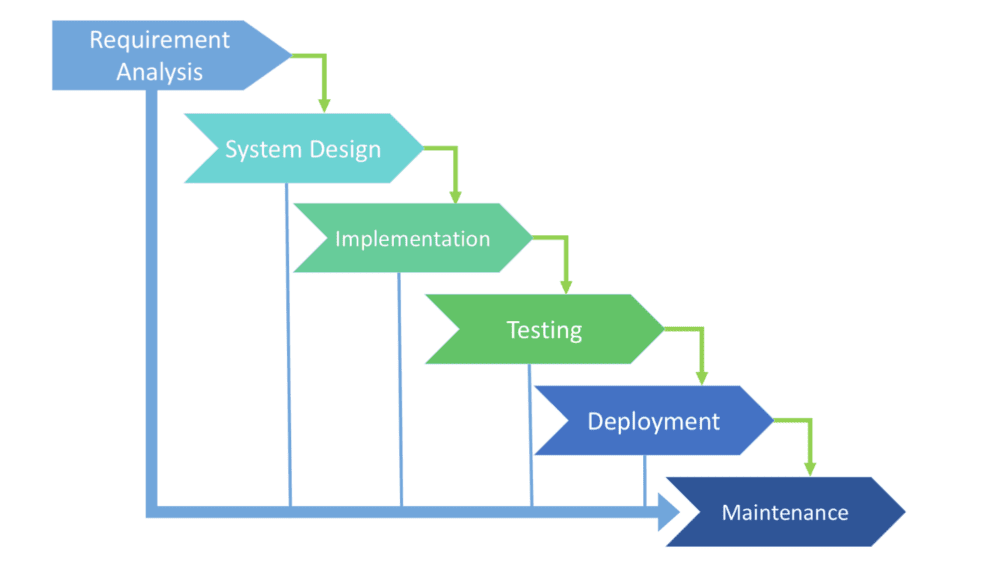
1. **Why is the SDLC important?**

* It provides a standardized framework that defines activities and deliverables
* In project planning, estimating, and scheduling
* It makes project tracking and control easier
* It increases visibility on all aspects of the life cycle to all stakeholders involved in the development process
* It increases the speed of development
* It improves client relations
* It decreases project risks
* It decreases project management expenses and the overall cost of production.

1. **How does the SDLC work?**

In detail, the SDLC methodology focuses on the following phases of software development:

* Requirement gathering
* Design
* Software development
* Testing
* Deployment
* Maintenace



1. **Requirement Gathering and Analysis -**

In this stage of the SDLC, the team determines the cost and resources required for implementing the analyzed requirements. It also details the risks involved and provides sub-plans for softening those risks.

1. **Design -**

This phase of the SDLC starts by turning the software specifications into a design plan called the Design Specification. This phase designs all the requirements given by the customer and code is written as per the design in next phase.

**3. Development-**

Expert software developers start the actual software development process. The required components and functionalities of software are created in this very stage.

**4. Testing-**

The testing stage assesses the software for errors and document bugs if there are any.

**5. Deployment-**

This stage is crucial for running the developed software assessed by all the stakeholders. This is to ensure the complete level of customer satisfaction.

**6. Maintenance-**

Once the software passes through all the stages without any issues, a maintenance process is followed wherein it will be maintained and upgraded

from time to time to adapt to changes. Almost every software development Indian company follows all the six steps, leading to the reputation that the country enjoys in the software market today.

## **4. What are the SDLC models/methodologies?**

1. **Waterfall model:** The waterfall model is a classical model used in SDLC to create a system with a linear & sequential approach. It is termed a waterfall because the model develops systematically from one phase to another in a downward fashion.

**Advantage:**

* Easy to understand.
* Resource management is easy.
* Suitable for small projects.
* The process is well documented.

**Disadvantage:**

* Clear & complete requirements are needed to login.
* Testing starts only after coding is done.
* Inefficient use of resources.

**2. V-model:** The V-model is a type of SDLC model where the process executes in a sequential manner in a V-shape. It is also known as the verification & validation model.

**Advantage:**

* The process is not too complicated.
* Project management is easy.
* Suitable for a small project.

**Disadvantage:**

* Not applicable for projects with changing requirements.
* Rigid process with no overlap.

**3. Iterative Model:** This is a particular implementation of an s/w development life cycle that focuses on an initial, simplified, implementation, which then progressively gains more complexity and a border feature set until the final system is complete.

* Start with sub-set of requirements.
* Each iteration handles a sub-set of requirements.
* Deployment happens & end.

**Advantage:**

* Complete requirements are not required to start.
* A small portion is developed in each iteration, which is easy to develop & test.
* Iteration is easy to manage.

**Disadvantage:**

* It is difficult to architect a system with incomplete requirements.
* Each iteration is rigid like waterfall model.

**4. Spiral Model:** The spiral model is a combination of waterfall & iterative development processes with an emphasizing on more risk analysis.

* Iterative Process
* Risk driven model
* Complex (compared with waterfall & V- model)
* Prototypes, benchmarks & simulation.

**Advantage:**

* Risk management is built into the model
* Best suitable for large projects
* Ideal for product-based companies.

**Disadvantage:**

* Complicated Process .
* Requirements constantly change
* Expensive

**5 . Agile Model:** SDLC model is a combination of iterative & incremental process models with a focus on process adaptability & customer satisfaction rapid delivery of working software products. Agile methods break the product into small incr**emental builds.**

**Advantage:**

* High customer satisfaction because of the continuous delivery of working software.
* People interaction are emphasized.
* Constant review of actions.
* Changes are welcomed.

**Disadvantage:**

* Architectural challenges.
* Efforts are difficult to access.
* Costs are high.
* Potential for scope creep.

**Benefits of SDLC:**

* Having overarching control over software development process
* Improving resource management and cost-effectiveness
* Gives teams a clear action plan
* Improves cooperation between participants