**Software Development Life Cycle (SDLC):**

* It’s a process used by software development teams to design, develop, test and deploy high-quality software. The typical phases include planning, requirement, analysis, design, implementation, testing, deployment and maintenance.

**There are two main SDLC Models**

* **Waterfall Model (Old Model)**

The Waterfall Model is a linear and sequential software development approach where each phase must be completed fully before the next phase can begin. It follows a structured process with distinct phases, such as requirements gathering and analysis, design, implementation and unit testing, and so on. Unlike other models, the phases in the Waterfall Model do not overlap, making it idealistic but less flexible

1. Requirement: In every Model business first need “Requirement”. They will take the complete requirement first.
2. Analysis: Here Technology component will decide. Means which technology should use for the Development.
3. Design: Entire project will have divided into modules. They will design all the module at a time.
   1. EX: Assume banking application.
      1. Saving Account Section
      2. Amount Transfer Section
      3. Profile Section
      4. Credit/Debit Card Section
4. Coding: Here developer will get the data flow diagram in design sheet from design team. And they will start the coding.
5. Testing: Once code will have completed testing team will test the application using script. And make sure application is behaving as per client requirement.
6. Deliver: Once testing is successful will deliver the application to client.

* **Agile Model (New Model)**

The Agile Model is an incremental and iterative process of software development that prioritizes working software over comprehensive documentation.

a. In Agile model business will took the requirement one by one in module basis. So all team will work parallel.

**Disadvantage of Waterfall Model**

* In the Waterfall Model, each stage must be completed before moving to the next. This sequential nature creates dependencies between teams. For instance, the testing team has to wait for the coding team to finish their work before they can begin testing.
* Consequently, this interdependence can lead to delays in project delivery, especially if any stage encounters unexpected issues or changes.
* The Waterfall Model follows a fixed plan with predefined requirements and outcomes. Once a stage is completed, it’s challenging to revisit and modify it.
* If changes are needed (due to design flaws, integrations, or evolving project needs), the process becomes cumbersome. A new design must be created, reviewed, and approved before implementation can proceed.
* Since changes are difficult to accommodate once a stage is completed, any necessary adjustments can be expensive and time-consuming.

**So if any Application Development in taking 2Y of time in Waterfall Model it can be reduce upto 1Y in Agile model.**

After introduced Agile Model also the expected time to deliver the application is more so after researches IT industries found that this delay is coming **Due to Gap between development team and operations team.**

* Development Team: Group of Developer are knowing as Development Team.
* Operation Team:
  + Installation of server hardware and OS.
  + Configuring of servers, networks, storage etc.
  + Monitoring of servers & Respond to outage.
  + Maintenance like patching.

Here DevOps method is introduced to fill the gap and reduce the time of SDLC.

**DevOps and its importance:**

* DevOps is a methodology that promotes collaboration between development and operations team. This allows deploying code to production faster and in an automate way. It helps to enable rapid deployment of product.
* DevOps is a methodology that can be implemented using tool.

Git is a DVCS that help to manage the source code and version of the code.

GIT & Linux is created by Linux Torvalds.

**Application Server:**

**Dependency File:**

Python - requirement.txt

Java - server.js

Node -

**Build Tool:**

java code -- maven

dotnet code --MS build

**Webserver:**

* NGINX and HTTPd (like Apache HTTP Server) are both web server software that support the HTTP protocol. They handle incoming HTTP requests from clients, such as web browsers, and serve web content, such as web pages, images, and other resources, in response to those requests. Both NGINX and HTTPd are widely used to host and deliver websites and web applications on the internet.