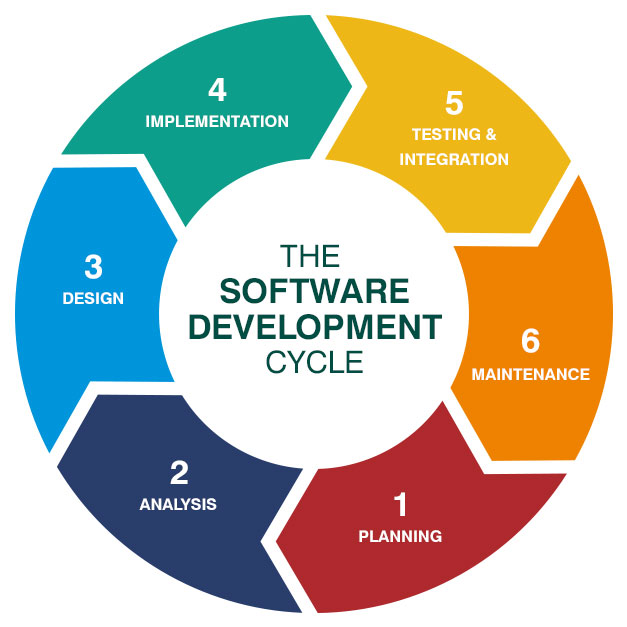
INTRODUCTION TO DEVOPS

The software development lifecycle (SDLC) encompasses all the stages involved in software development, from initial planning to deployment and maintenance. The typical stages of the SDLC include:



Planning: This phase involves gathering requirements, defining project scope, and creating a roadmap for the software project.

Analysis: In this phase, the gathered requirements are analyzed in detail to understand the needs of the users and the business. This often involves creating documentation such as functional specifications or user stories.

Design: During this stage, the system architecture and design are created based on the requirements and analysis. This includes designing the user interface, database structure, and overall system architecture.

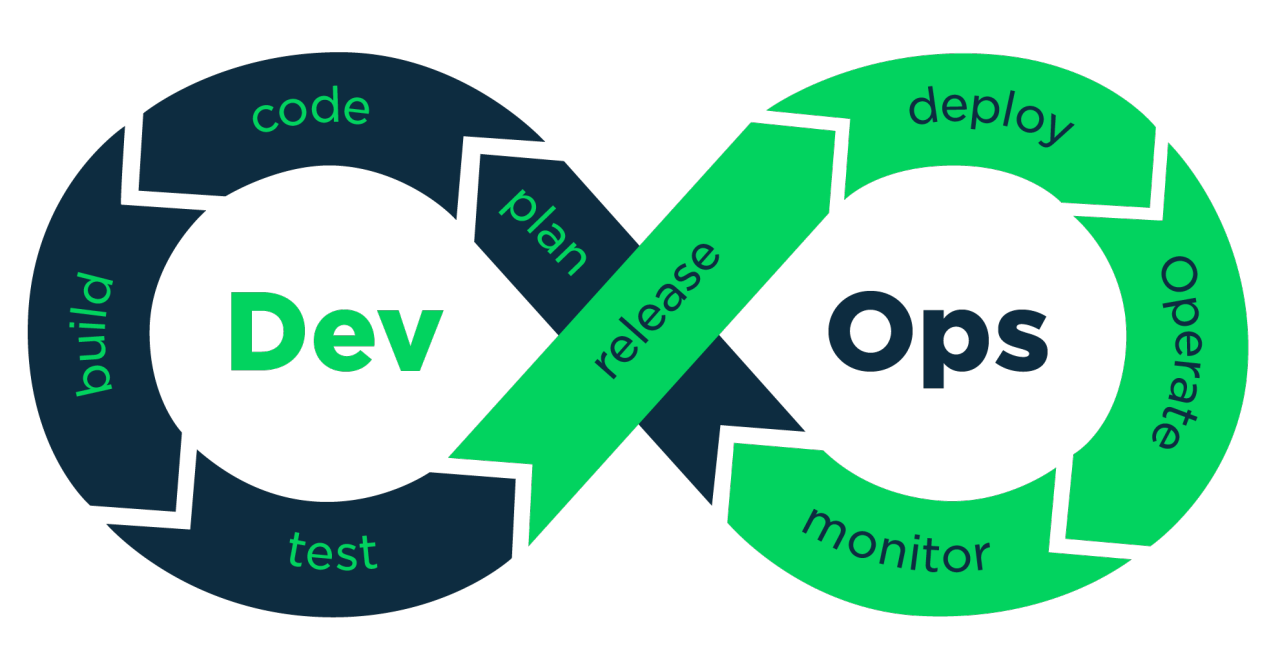
Implementation (Coding): In this phase, developers write the code based on the design specifications. This is where the actual development of the software takes place.

Testing: Once the code is written, it undergoes testing to identify and fix any defects or bugs. Testing can include unit testing, integration testing, system testing, and acceptance testing.

Deployment: After successful testing, the software is deployed to production or made available to users. This involves installing the software on servers, configuring it, and making it accessible to users.

Operation and Maintenance: Once the software is deployed, it enters the maintenance phase where it is monitored, updated, and maintained. This includes fixing bugs, adding new features, and addressing any issues that arise in production.

DevOps practices are integrated throughout the SDLC to streamline processes, improve collaboration, and automate tasks wherever possible. This includes continuous integration (CI), continuous delivery (CD), infrastructure as code (IaC), automated testing, and continuous monitoring. These practices help to accelerate development cycles, improve software quality, and enhance the overall efficiency of the SDLC.



DevOps (a portmanteau of “development” and “operations”) is the combination of practices and tools designed to increase an organization’s ability to deliver applications and services faster than traditional software development processes. This speed enables organizations to better serve their customers and compete more effectively in the market.

In simple terms, DevOps is about removing the barriers between traditionally siloed teams, development and operations. Under a DevOps model, development and operations teams work together across the entire software application life cycle, from development and test through deployment to operations.

Some of the challenges DevOps strives to solve include:

Siloed Development and Operations: Historically, development and operations teams have operated in isolation, leading to inefficiencies, misunderstandings, and delays in software delivery.

Manual Processes: Manual processes in software development and deployment are error-prone, time-consuming, and hinder agility.

Slow Deployment Cycles: Traditional software development methods often result in long release cycles, delaying the delivery of new features and updates to end-users.

Lack of Collaboration: Poor communication and collaboration between development, operations, and other stakeholders can lead to misunderstandings, conflicts, and delays in software delivery.

Scalability and Flexibility: Traditional infrastructure and deployment methods may struggle to scale and adapt to rapidly changing business requirements and user demands.

Quality Assurance: Ensuring the quality and reliability of software releases can be challenging without automated testing, continuous integration, and continuous deployment practices.

DevOps addresses these challenges by promoting automation, collaboration, and continuous improvement across the software development lifecycle, from planning and coding to testing, deployment, and monitoring.