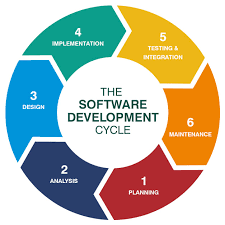
**Software Development Life Cycle (SDLC):** 

* Phase 1: Planning
* Phase 2: Requirements Analysis
* Phase 3: Design
* Phase 4: Coding
* Phase 5: Testing
* Phase 6: Deployment
* Phase 7: Maintenance

**Phase 1: Planning**

The initial stage of software development, **Planning**, involves **defining the software’s purpose and scope**, much like pinpointing our destination and plotting the best route.

The Planning phase fosters effective communication and collaboration within the team. By defining clear roles, responsibilities, and expectations, it lays a solid foundation for an efficient software development process.

**Phase 2: Requirements Analysis**

Phase 2 of the SDLC, **Requirements Analysis**, seeks to **identify and record the precise requirements of the final users**.

The success of the Requirements Analysis phase is pivotal for the entire project. Done right, it leads to a software solution that meets users’ needs and exceeds their expectations.

**Phase 3: Design**

The **Design** phase is all about **building the framework**. The development team is responsible for software engineering and outlines the software’s functionality and aesthetic. This ultimately results in the software product.

The Design phase is the link between the software’s purpose (Planning and Requirement phases) and its execution (coding phase). It’s an essential step in creating software that works efficiently and provides an excellent user experience.

**Phase 4: Coding**

The **Coding** phase in the Software Development Life Cycle (SDLC) is when engineers and developers get down to business and start **converting the software design into tangible code**.

At the end of this phase, a functional piece of software comes to life.

**Phase 5: Testing**

Consider the **Testing** phase of the SDLC as a **stringent quality inspection on a production line**. It is when vulnerabilities are uncovered. Software testing involves examination of the software for any bugs or glitches that might have slipped through during coding. The aim is to ensure flawless software operation before it reaches the end-users. And even identify opportunities for enhancement. The Testing phase is instrumental in ensuring the software’s robustness and reliability.

**Phase 6: Deployment**

After crafting a product with precision, it’s time to present it to the users by pushing to the production environment. The **Deployment** phase **involves rolling out the tested and fine-tuned software to its end-users**.

The Deployment phase doesn’t signal the end, but rather a notable milestone. It signifies the shift from a project phase to a product phase, where the software begins to fulfil its purpose.

**Phase 7: Maintenance**

In the Software Development Life Cycle, the **maintenance** phase is characterized by constant assistance and improvement, which **guarantees the software’s best possible functioning and longevity and ensures it meets customer expectations**.

Maintenance tasks encompass frequent software updates, implementing patches, and fixing bugs. User support is also a crucial component, offering help and guidance to users facing difficulties with the software.