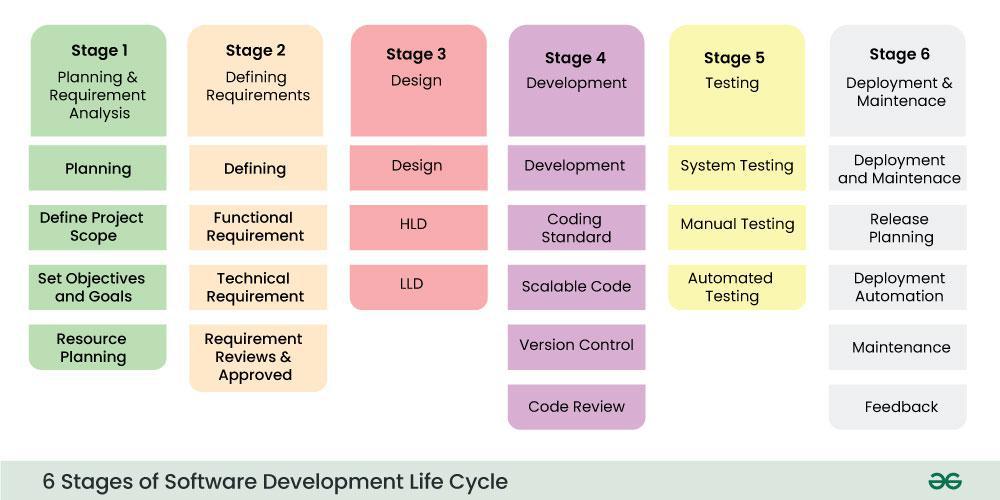
SDLC

The **Software Development Life Cycle (SDLC)** is a structured process used for developing software applications with a systematic approach. It consists of several phases, ensuring that software is built efficiently, with high quality, and within budget.



**Stages of the Software Development Life Cycle**

SDLC specifies the task(s) to be performed at various stages by a software engineer or developer. It ensures that the end product is able to meet the customer’s expectations and fits within the overall budget. Hence, it’s vital for a software developer to have prior knowledge of this software development process. SDLC is a collection of these six stages, and the stages of SDLC are as follows:



**Phases of SDLC**

1. **Planning**
   * Define project scope, objectives, and feasibility.
   * Identify risks, budget, and resource allocation.

A diagram of a project

AI-generated content may be incorrect.

1. **Requirement Analysis**
   * Gather and document functional and non-functional requirements.
   * Work with stakeholders to finalize system specifications.

A diagram of a technical requirements

AI-generated content may be incorrect.

1. **Design**
   * Create architectural designs, UI/UX, and database structures.
   * High-Level Design (HLD) and Low-Level Design (LLD).

A diagram of a low level design

AI-generated content may be incorrect.

1. **Implementation (Coding)**
   * Developers write the actual code based on design specifications.
   * Ensure best coding practices and version control.

A diagram of a code

AI-generated content may be incorrect.

1. **Testing**
   * Identify and fix bugs through unit, integration, system, and user acceptance testing.
   * Validate that software meets requirements.

A close-up of a manual testing

AI-generated content may be incorrect.

1. **Deployment**
   * Release software to production.
   * Deploy in a staged manner (e.g., alpha, beta, full release).

A diagram of a pink square with black text

AI-generated content may be incorrect.

1. **Maintenance & Support**
   * Provide updates, bug fixes, and enhancements.
   * Monitor system performance and ensure reliability.

**SDLC Models**

There are different SDLC methodologies depending on project requirements:

* **Waterfall Model** (Sequential, step-by-step approach)
* **Agile Model** (Iterative, flexible, with frequent releases)
* **V-Model** (Testing at each stage of development)
* **Spiral Model** (Risk-driven, iterative process)
* **Big Bang Model** (Minimal planning, suitable for small projects)
* **RAD (Rapid Application Development) Model** (Fast prototyping, quick delivery)

Waterfall

🡪fundamental model of the software development life cycle

🡪very simple model

🡪not in practice anymore, but it is the basis for all other SDLC model

🡪Because of its simple structure and easier to use and provides a tangible output

🡪once a phase seems to be completed, it cannot be changed,

🡪due to this less flexible nature, the waterfall model is not in practice anymore.