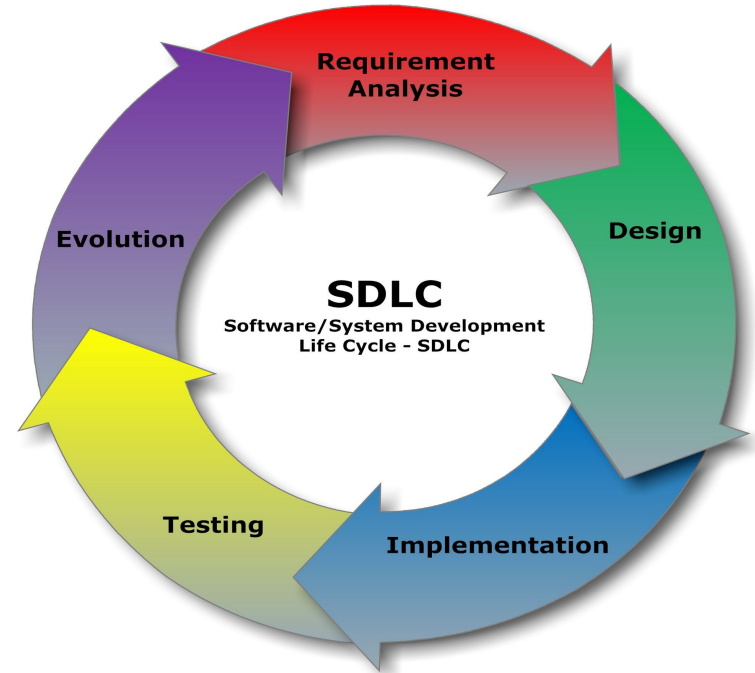# Software Development Life Cycle

What is SDLC?

The **Software Development Life Cycle (SDLC)** is a structured process for developing software, guiding projects through phases like planning, design, coding, testing, and maintenance. It ensures software is built efficiently and meets user requirements.

Types of SDLC

1. ### Waterfall Model

   A linear and sequential approach to software development where each phase must be completed before moving to the next, making it rigid and less flexible for changes.
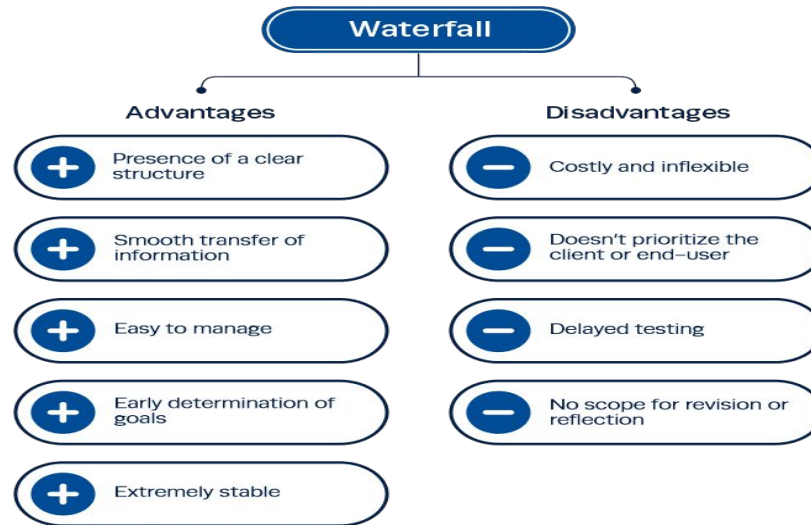
2. Iterative Model

A cyclical process where the software is developed in repeated iterations, allowing for incremental improvements and adjustments based on ongoing feedback.

Advantage and Disadvantages of Incremental Model

This slide represents the advantages, and disadvantages of the incremental model, including easy error detection, accessibility to test and troubleshoot, adaptability, ease to manage risk, and so on.

Advantages	Disadvantages
o Errors are Easily Identifiable	o Good Planning is Required
o A Lot Easier to Test and Troubleshoot	o Entire Cost is Rather Significant
o More Adaptable	o Module Interfaces Must be well Defined
o Risk is Easy to Manage Because it was Organized at the Iteration	-----------------------------------
o Necessary Functionality is Sent to the Client Early	-----------------------------------
o Add text here	o Add text here

3. Spiral Model

A risk-driven software development model that combines iterative development with systematic risk analysis, repeating the development process in spirals to refine the system progressively.

Spiral model

Advantages of Spiral model:

- High amount of risk analysis hence, avoidance of Risk is enhanced.
- Good for large and mission-critical projects.
- Strong approval and documentation control.
- Additional Functionality can be added at a later date.
- Software is produced early in the **software life cycle**.

Disadvantages of Spiral model:

- Can be a costly model to use.
- Risk analysis requires highly specific expertise.
- Project's success is highly dependent on the risk analysis phase.
- Doesn't work well for smaller projects.

4. V– Model

A linear software development model where each development phase is paired with a corresponding testing phase, emphasizing verification and validation at every stage of the process.

Advantages & Disadvantages of V-Model

Advantages & Disadvantages of V-Model

👍 Advantages of V-Model	👎 Disadvantages of V-Model
This is a highly disciplined model and Phases are completed one at a time.	High risk and uncertainty.
V-Model is used for small projects where project requirements are clear.	It is not good for complex and object-oriented projects.
Simple and easy to understand and use.	It is not suitable for projects where requirements are not clear and contains a high risk of changing.
This model focuses on verification and validation activities early in the life cycle thereby enhancing the probability of building an error-free and good quality product.	This model does not support iteration of phases.
It enables project managers to track progress accurately.	It does not easily handle concurrent events.

ITERATIVE APPROACH

The Iterative model is often considered the best among the four because it allows for flexibility and ongoing improvements by developing the software in small, manageable cycles. Unlike the Waterfall and V-Model, it adapts well to changes, and it's less complex and costly than the Spiral model.