

# ***Software Development Life Cycle (SDLC) Overview***

## **1. Requirements Gathering:**

Importance: Defines what the software should accomplish from the user's perspective.

Interconnects: Provides a foundation for subsequent phases, guiding design and implementation.

## **2. Design:**

Importance: Translates requirements into a blueprint for software construction.

Interconnects: Feeds into implementation, ensuring the software aligns with desired functionality.

## **3. Implementation:**

Importance: The actual coding and development of the software based on the design specifications.

Interconnects: Directly influenced by design, aims to bring the design to life in the form of functioning software.

## **4. Testing:**

Importance: Verifies that the software meets requirements and functions correctly.

Interconnects: Identifies issues or discrepancies between expected and actual behavior, providing feedback to refine implementation.

## **5. Deployment:**

Importance: The process of releasing the software for users after thorough testing.

Interconnects: Culmination of the SDLC, ensuring that the developed software is delivered and accessible to users.

Note: The SDLC phases are iterative and may require revisiting previous stages based on feedback and changes.

This infographic provides a concise overview of the SDLC phases, emphasizing their importance and interconnectedness in the software development process.