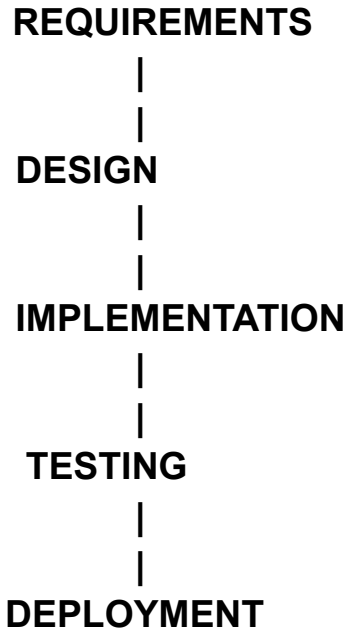
Assignment 1 :SDLC Overview - Create a one-page infographic that outlines the SDLC phases (Requirements, Design, Implementation, Testing, Deployment), highlighting the importance of each phase and how they interconnect.



SDLC:
The Software Development Life Cycle (SDLC)A Structured Approach to Building High-Quality Software.
The process of planning, designing, developing, testing, deploying, and maintaining software applications is called the Software Development Life Cycle (SDLC). It offers a structure that enables teams to collaborate productively and successfully to produce high-caliber software that satisfies user needs. Since the SDLC is an iterative process, teams may go back and review previous stages as they gain further project knowledge.

1. Requirements

Importance:

- **Foundation**: Clear, well-documented requirements set the foundation for the entire project.
- **Understanding**: Ensures all stakeholders have a shared understanding of the project scope and objectives.

Key Activities:

- Gathering requirements from stakeholders
- Analyzing and prioritizing requirements
- Documenting requirements (e.g., Requirement Specification Document)

Output:

- Requirement Specification Document

2. Design

Importance:

- **Blueprint**: Provides a detailed blueprint for the system architecture and design.
- **Plan**: Helps in planning the resources, time, and budget.

Key Activities:

- System design (High-Level Design)
- Detailed design (Low-Level Design)
- Creating wireframes and mockups

Output:

- Design Documents, Wireframes, Mockups

3. Implementation

Importance:

- **Build**: Actual coding and building of the system based on design specifications.
- **Integration**: Integration of various modules and functionalities.

Key Activities:

- Coding
- Unit Testing
- Integration of different modules

Output:

- Source Code, Unit Test Results

4. Testing

Importance:

- **Validation**: Ensures the system meets the requirements and works as expected.
- **Quality Assurance**: Identifies and fixes bugs before deployment.

Key Activities:

- Writing test cases
- Conducting various tests (Unit, Integration, System, User Acceptance)
- Logging and fixing bugs

Output:

- Test Plans, Test Cases, Bug Reports, Test Results

5. Deployment

Importance:

- **Go Live**: Making the system available for use in the production environment.
- **Transition**: Ensures a smooth transition from development to operation.

Key Activities:

- Preparing deployment plans
- Setting up the production environment
- Deploying the system

Output:

- Deployed Application, Deployment Documentation

Interconnections:

- **Requirements to Design**: Requirements guide the design of the system.
- **Design to Implementation**: Design documents serve as a blueprint for the coding phase.

- **Implementation to Testing**: The implemented code is tested to ensure it meets the requirements.
- **Testing to Deployment**: A thoroughly tested system is deployed to the production environment.
- **Feedback Loop**: Post-deployment feedback can lead to new requirements, starting the cycle again.

This SDLC infographic outlines the phases and emphasizes their importance and interconnections, providing a comprehensive overview of the software development process.