Assignment-1

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.

Introduction

The Software Development Life Cycle (SDLC) is a structured process that ensures the quality and correctness of software built. It consists of several phases: Requirements, Design, Implementation, Testing, and Deployment. Each phase is crucial and interdependent, forming a coherent flow from conception to delivery.

Phases of SDLC

1.**Requirements**

- **Objective:** Gather and analyze the needs and expectations of the stakeholders.
- **Importance:**
- Defines project scope and purpose.
- Serves as a foundation for all subsequent phases.

Key Activities:

- Requirement gathering (interviews, surveys, etc.).
- Requirement analysis and validation.
- Documentation (SRS - Software Requirement Specification).

2.Design

- **Objective:** Translate requirements into a blueprint for the system.

Importance:

- Provides a clear roadmap for developers.
- Ensures all functional and non-functional requirements are addressed.

Key Activities:

- System design (architecture, data flow diagrams, etc.).
- Detailed design (modules, interfaces, etc.).

- Design documentation.

3.Implementation

Objective: Convert the design into executable code.

Importance:

- Realizes the project’s functionality.
- Ensures code adheres to design specifications.

Key Activities:

- Coding.
- Code review and optimization.
- Version control.

4.Testing

Objective: Verify that the software meets requirements and is bug-free.

Importance:

- Ensures quality and reliability of the software.
- Identifies defects before deployment.

Key Activities:

- Unit testing, integration testing, system testing.
- User acceptance testing (UAT).
- Test documentation and reporting.

5.Deployment

Objective: Release the software to production for use.

Importance:

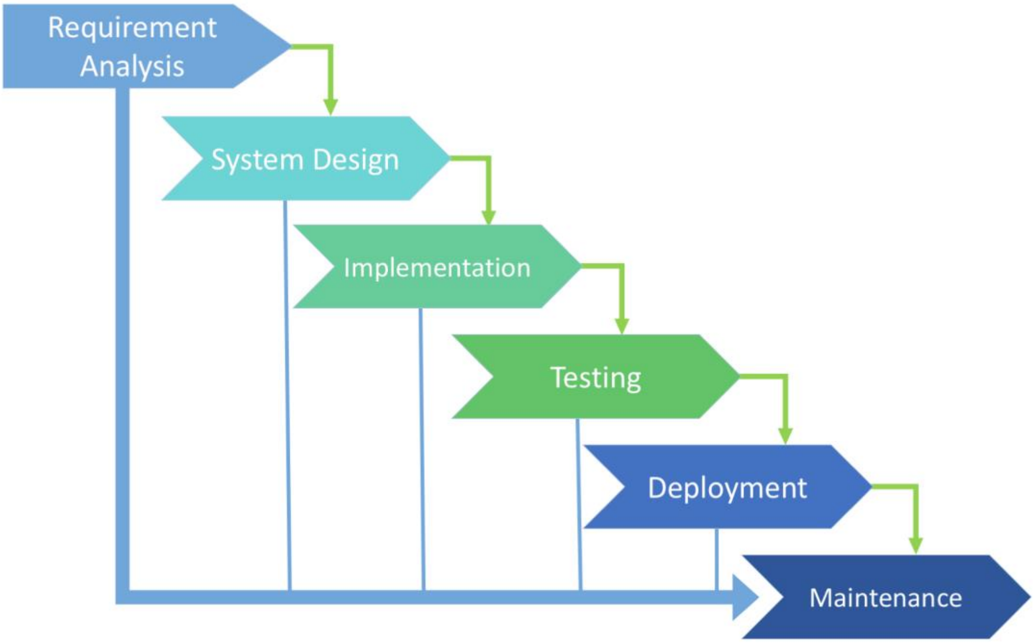
- Delivers the final product to users.
- Involves critical steps to ensure smooth operation.

Key Activities:

- Deployment planning and preparation.
- Environment setup and configuration.
- Monitoring and support post-deployment.

Interconnections and Flow

- **Requirements ↔ Design:** Clear requirements ensure a precise design. Any ambiguity in requirements can lead to design flaws.
- **Design ↔ Implementation:** A well-thought-out design simplifies implementation. Developers use design documents as a reference.
- **Implementation ↔ Testing:** Testing relies on implementation. The code written is rigorously tested to ensure it meets the design and requirements.
- **Testing ↔ Deployment:** Successful testing ensures a smooth deployment. Any issues found during testing must be resolved before deployment.
- **Deployment ↔ Requirements:** Feedback from deployment can lead to new requirements, initiating a new SDLC cycle for enhancements or new features.


Requirement Analysis
System Design
Implementation
Testing
Deployment
Maintenance