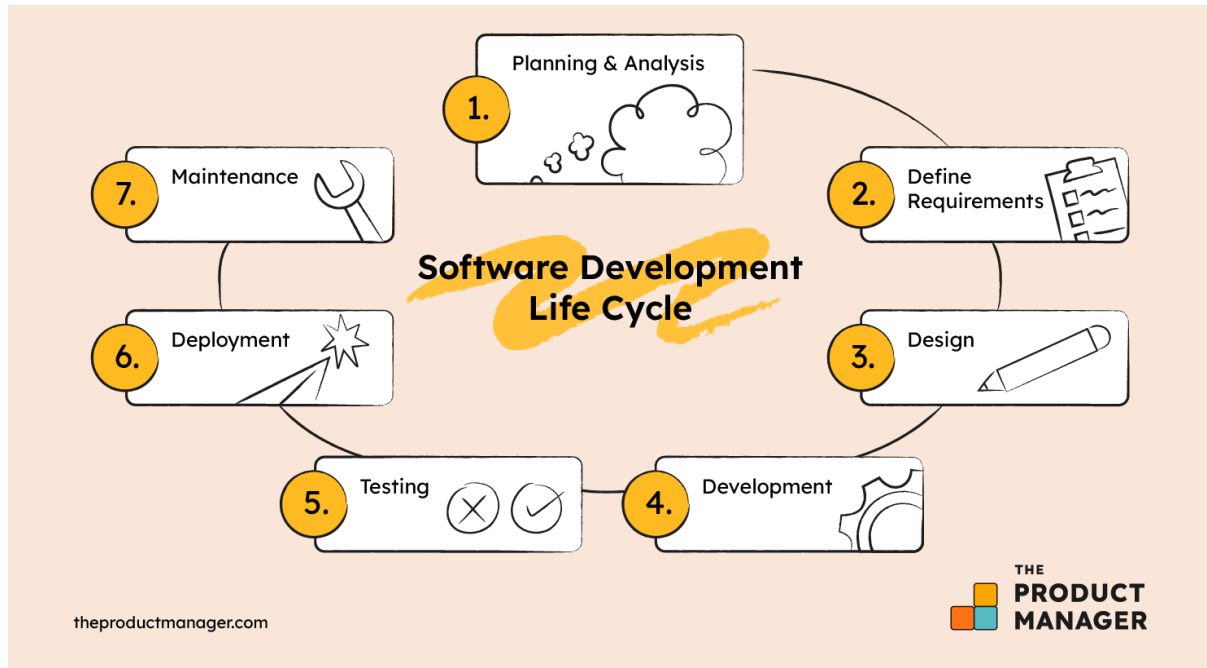


**Assignment 1: SDLC Overview** - Create a one-page infographic that outlines the SDLC phases (Requirements, Design, Implementation, Testing, Deployment), highlighting the importance of each phase and how they interconnect.

## SDLC Overview Infographic

**Title: Software Development Life Cycle (SDLC) Phases**



### Introduction:

- Brief overview of SDLC and its importance in software development.

### Phase 1: Requirements

**Objective:** Gather and document project requirements.

### Importance:

- Establishes project scope and objectives.
- Guides design and development efforts.
- Minimizes misunderstandings between stakeholders.

**Interconnection:** Requirements feed into the design phase to define system architecture and functionality.

---

## Phase 2: Design

**Objective:** Create a blueprint for the software system.

**Importance:**

- Translates requirements into technical specifications.
- Defines system architecture, components, and interfaces.
- Guides developers in implementation.

**Interconnection:** Design decisions influence implementation and testing phases, ensuring alignment with project goals.

---

## Phase 3: Implementation

**Objective:** Write and integrate code according to design specifications.

**Importance:**

- Turns design into functioning software.
- Requires coding standards and best practices.
- Involves collaboration among developers.

**Interconnection:** Implementation phase follows design specifications and precedes testing to verify functionality.

---

## Phase 4: Testing

**Objective:** Verify and validate software against requirements.

**Importance:**

- Identifies defects and ensures software quality.
- Includes unit testing, integration testing, and system testing.
- Validates functionality and performance.

**Interconnection:** Testing phase validates that implemented software meets specified requirements before deployment.

---

## **Phase 5: Deployment**

**Objective:** Release software for users to access and use.

**Importance:**

- Ensures smooth transition to production.
- Includes installation, configuration, and user training.
- Monitors performance and resolves post-deployment issues.

**Interconnection:** Deployment follows successful testing and involves stakeholders for user acceptance and feedback.

---

**Conclusion:**

- Summary of SDLC phases' roles in delivering high-quality software.
- Emphasizes the iterative and interconnected nature of SDLC phases for successful project outcomes.

**References:**

- Include sources or further reading materials on SDLC phases and best practices.
- 

## **Design Elements:**

- Use icons or illustrations to represent each SDLC phase (e.g., document for requirements, blueprint for design, code snippet for implementation, magnifying glass for testing, and rocket for deployment).
- Employ a consistent colour scheme and layout for clarity and visual appeal.
- Include arrows or connectors to visually show the flow and interconnection between SDLC phases.

## **Visual Style:**

- opt for a clean and structured layout with sections for each phase.
- Ensure readability with concise text and bullet points.
- Incorporate whitespace to enhance visual clarity and focus.

## **Tools:**

- Design software like Adobe Illustrator, Canva, or PowerPoint can be used to create the infographic.
- Focus on simplicity and effectiveness in conveying information about SDLC phases and their importance in software development.

By following these guidelines, you can create a concise and informative infographic that effectively outlines the SDLC phases, their significance, and how they interconnect to ensure successful software development projects.