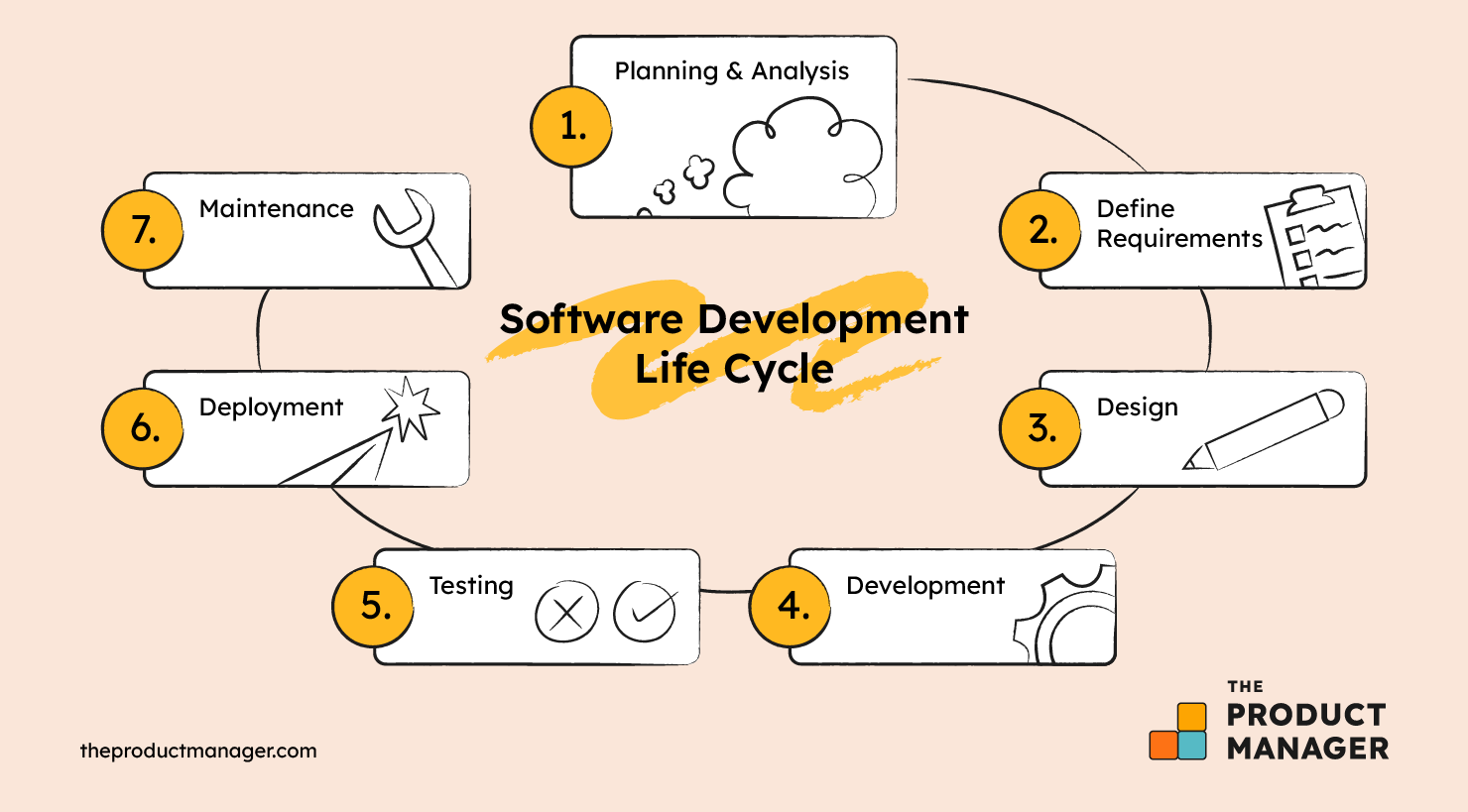
**Assignment 1: SDLC Overview - Create a one-page infographic that outlines the SDLC phases (Requirements, Design, Implementation, Testing, Deployment), highlighting the importance of each phase and how they interconnect.**

**SOFTWARE DEVELOPMENT LIFE CYCLE**



**SDLC Phases :**

**Phase 1: Requirements**

Importance: Defines the project scope, objectives, and constraints by gathering stakeholder needs.

Key Points: Identifies functional and non-functional requirements and documents them for the development team.

**Phase 2: Design**

Importance: Establishes the system architecture and detailed design, translating requirements into a blueprint for the system.

Key Points: Involves creating design specifications, user interface designs, and setting the foundation for implementation.

**Phase 3: Implementation**

Importance: The actual coding and development of the system take place in this phase, transforming design documents into a working software application.

Key Points: Includes source code development, module integration, and maintaining version control.

**Phase 4: Testing**

Importance: Ensures the system meets all specified requirements and identifies any defects before deployment.

Key Points: Comprises unit testing, system testing, and user acceptance testing to verify and validate the software.

**Phase 5: Deployment**

Importance: Involves making the software available for use by deploying it to the production environment.

Key Points: Encompasses deployment planning, data migration, and the actual release of the system to users.

**Connecting the Phases**

**Requirements to Design:**

* Clear and well-documented requirements guide the design process, ensuring that the system will address user needs effectively.
* The requirements provide a foundation for creating detailed system and architectural designs, aligning with the intended functionality.

**Design to Implementation:**

* The design specifications act as a blueprint for developers, detailing how each component should be built and integrated.
* By following the design document, developers ensure consistency and coherence in the coding process, translating the design into functional code.

**Implementation to Testing:**

* Once coding is completed, the software undergoes various levels of testing to verify that it meets the specified requirements and is free of defects.
* Testing identifies and allows for the correction of issues, ensuring that the implementation aligns with the intended design and functions properly.

**Testing to Deployment:**

* After successful testing, the software is validated as ready for release, ensuring that it performs reliably and meets quality standards.
* Deployment involves planning and executing the release of the software to the production environment, making it available for end-users.

**Feedback Loop:**

* Post-deployment feedback is collected to identify any issues or additional requirements, which can initiate a new cycle of the SDLC for continuous improvement.
* This feedback ensures that the software evolves to meet user needs and adapt to changing conditions, maintaining its relevance and effectiveness.