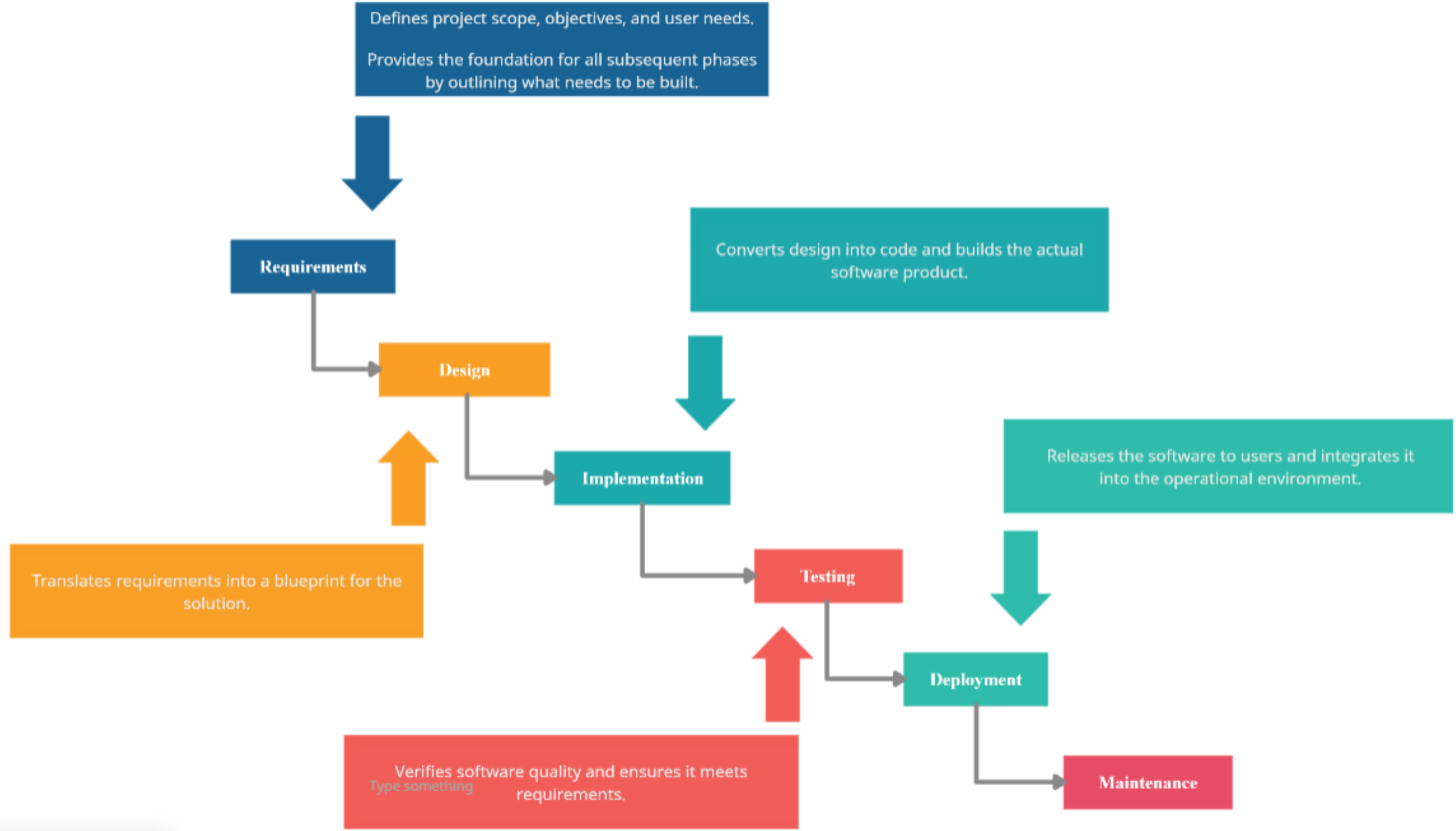

Defines project scope, objectives, and user needs.

Provides the foundation for all subsequent phases by outlining what needs to be built.

Translates requirements into a blueprint for the solution.

Converts design into code and builds the actual software product.

Verifies software quality and ensures it meets requirements.

Releases the software to users and integrates it into the operational environment.

# Assignment 2

## Case Study

## Implementation of SDLC Phases in an Online Banking System

**1. Requirement Gathering**:

The project team conducts surveys with bank customers to understand their banking habits, preferences.

Based on user feedback, the team defines requirements such as account management, fund transfers, bill payments, and security features like multi-factor authentication.

Thorough requirement gathering ensures that the online banking system meets user expectations and regulatory standards.

**2. Design:**

The team designs the system architecture, including client-server communication protocols, database.

UX/UI designers create design to visualize the user interface, navigation and accessibility.

Effective design ensures that the online banking system is scalable, secure, and user-friendly

3. Implementation:

Developers write code according to the design specifications, implementing features such as account creation, transaction processing, and encryption algorithms.

Modules are integrated, and unit, integration, and system tests are conducted to verify functionality and identify defects.

Successful implementation ensures that the online banking system functions as intended, providing reliable and secure banking services to customers.

**4. Testing:**

Testers verify that all system functions, such as account login, balance inquiries, and fund transfers, work correctly.

**5. Deployment:**

The system is deployed in phases, starting with a pilot rollout to a small group of users before full-scale deployment to all customers.

Training sessions are conducted to familiarize customers with the new online banking platform and its features.

Smooth deployment minimizes disruption to banking operations and ensures a seamless transition for customers, leading to high adoption rates and customer satisfaction.

**6. Maintenance:**

The development team continues to monitor the system post-deployment, addressing any issues reported by users and releasing updates to improve functionality and security.

# Assignment 3

## Waterfall Model:

Best suited for projects with stable and well-understood requirements, such as traditional banking systems where changes are infrequent.

**Advantages:** It's simple and easy to understand, making it suitable for small projects with well-defined requirements. Each phase has distinct deliverables, making progress easy to track.

**Disadvantages:** Lack of flexibility, as changes are difficult and costly to implement once a phase is completed. It's less suitable for projects with evolving or unclear requirements.

********************************************************************************

## Agile Methodology:

Ideal for projects with evolving or unclear requirements, such as developing innovative banking features or digital banking solutions.

**Advantages**: Emphasizes flexibility and collaboration, allowing for iterative development and frequent customer feedback. It accommodates changes in requirements and encourages continuous improvement.

**Disadvantages**: Requires active involvement and commitment from stakeholders. It may be challenging to manage in large, complex projects with extensive documentation requirements.

********************************************************************************

## Spiral Model

Suitable for projects with high levels of uncertainty or risk, such as developing new banking technologies or integrating complex systems.

**Advantages**: Integrates elements of both waterfall and iterative development, allowing for risk management and early prototyping. It accommodates changes while providing a structured approach to development.

**Disadvantages:** Requires extensive planning and risk analysis, which can increase project overhead. It may be complex to manage and require skilled project management expertise.

********************************************************************************

## V-Model:

Well-suited for projects with strict regulatory requirements or where quality and reliability are paramount, such as building critical banking infrastructure or financial systems.

**Advantages:** Emphasizes verification and validation throughout the development lifecycle, ensuring quality and compliance with requirements. It provides a systematic approach to testing and integration.

**Disadvantages:** It can be rigid and sequential, making it less adaptable to changes in requirements or project scope. It may not be suitable for projects with evolving or dynamic requirements.

******************************************************************************************