UTT

UNIVERSIDAD TECNOLÓGICA DE TIJUANA

GOBIERNO DE BAJA CALIFORNIA

Assignment:

Planning process of development of software.

BY:

Daniel Chavez Madrigal

GROUP:

9-B

SUBJECT:

Software development process management

PROFESSOR:

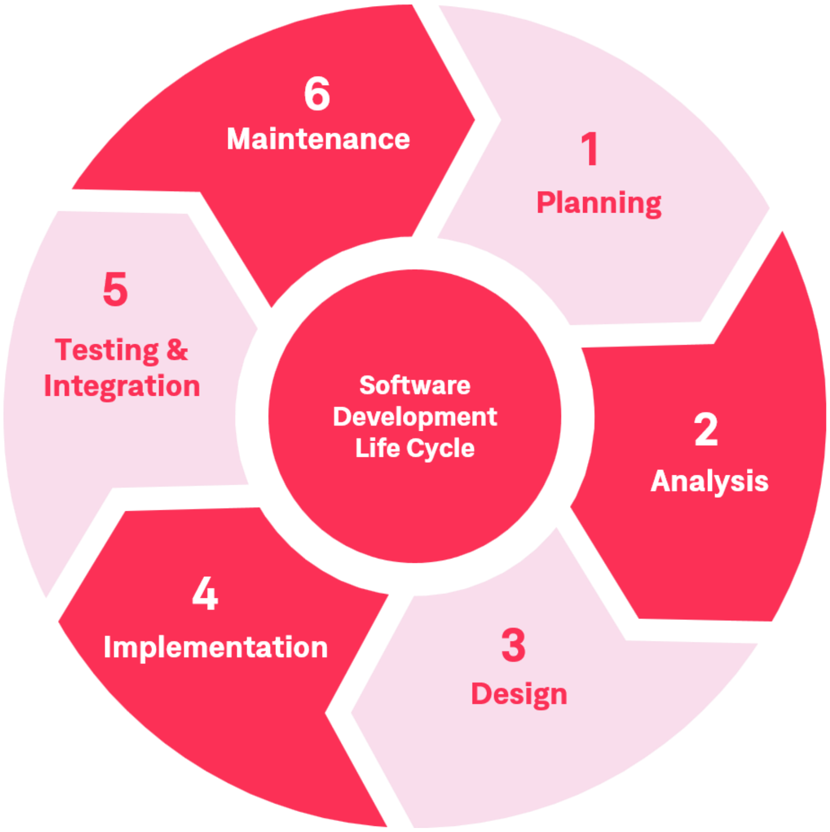
Ray Brunett Parra Galaviz

Tijuana, Baja California, 10 de enero del 2025

Introduction to the Software Development Process

The software development process, also known as the Software Development Life Cycle (SDLC), is a structured approach to creating software applications. It involves a series of steps that guide the development team from the initial concept to the final deployment and maintenance of the software. The primary goal of the SDLC is to produce high-quality software that meets or exceeds customer expectations, is completed within time and cost estimates, and is maintainable and scalable.

The SDLC typically includes the following phases: planning, analysis, design, implementation, testing and integration, and maintenance. Each phase has specific objectives and deliverables, and the process is iterative, allowing for continuous improvement and adaptation.



Detailed Explanation of Each Phase

1. Planning

- **Objective**: Define the project goals, scope, and deliverables. This phase involves understanding the requirements, setting timelines, and identifying the resources needed.
- **Stakeholder Involvement**: Engage with stakeholders to gather their input and ensure alignment with business objectives.
- **Risk Management**: Identify potential risks and develop strategies to mitigate them.
- **Project Roadmap**: Create a detailed project roadmap that outlines key milestones and deliverables.

2. Analysis

- **Requirement Gathering**: Collect detailed requirements from stakeholders. This includes functional and non-functional requirements.
- **Feasibility Study**: Assess the technical, financial, and operational feasibility of the project.
- **Documentation**: Create a Software Requirement Specification (SRS) document that outlines all the requirements.
- **Stakeholder Review**: Review the requirements with stakeholders to ensure accuracy and completeness.

3. Design

- **System Design**: Develop the architecture of the system, including data flow diagrams, entity-relationship diagrams, and system interfaces.
- **Prototyping**: Create prototypes to visualize the design and gather feedback from stakeholders.
- **Detailed Design**: Define the detailed specifications for each component of the system, including database design, user interfaces, and algorithms.
- **Design Review**: Conduct design reviews to ensure that the design meets the requirements and is feasible.

4. Implementation

- **Coding**: Write the actual code based on the design specifications. This phase involves using programming languages, frameworks, and tools.
- **Version Control**: Use version control systems like Git to manage code changes and collaborate with team members.
- **Unit Testing**: Perform unit tests to ensure that individual components work as expected.
- **Code Review**: Conduct code reviews to ensure code quality and adherence to standards.

5. Testing and Integration

- **Integration Testing**: Combine individual components and test them as a group to ensure they work together correctly.
- **System Testing**: Test the entire system to verify that it meets the specified requirements.
- **User Acceptance Testing (UAT)**: Conduct testing with end-users to ensure the system meets their needs and expectations.
- **Bug Fixing**: Identify and fix any issues or bugs found during testing.

6. Maintenance

- **Deployment**: Release the software to the production environment.
- **Monitoring**: Continuously monitor the system for performance and issues.
- **Updates and Enhancements**: Provide regular updates and enhancements based on user feedback and changing requirements.
- **Bug Fixes**: Address any issues or bugs that arise after deployment.
- **Support**: Offer ongoing support to users to ensure they can effectively use the software.

References:

Introduction to software development and applications. (n.d.). Coursera. https://www.coursera.org/learn/introduction-to-software-development-and-applications

Bansal, A. (2024, September 26). 7 Phases of the System Development Life Cycle Guide. *CloudDefense.AI*. https://www.clouddefense.ai/system-development-life-cycle/