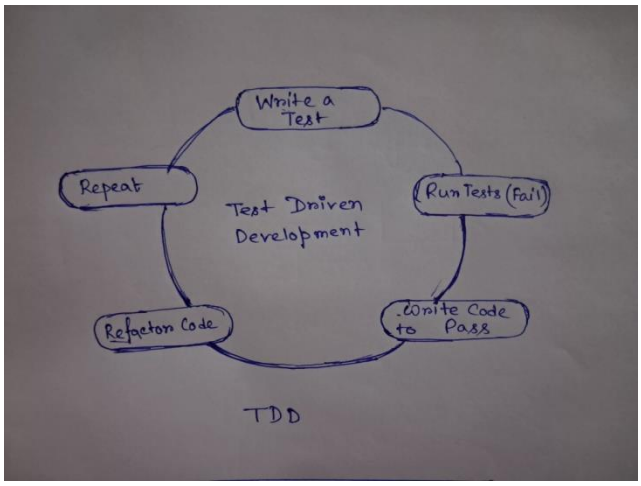
Debakshmi Pan

Assignment 1: <u>Create an infographic illustrating the Test-Driven Development (TDD) process. Highlight steps like writing tests before code, benefits such as bug reduction, and how it fosters software reliability.</u>

Test-Driven Development (TDD)



TDD Cycle Steps

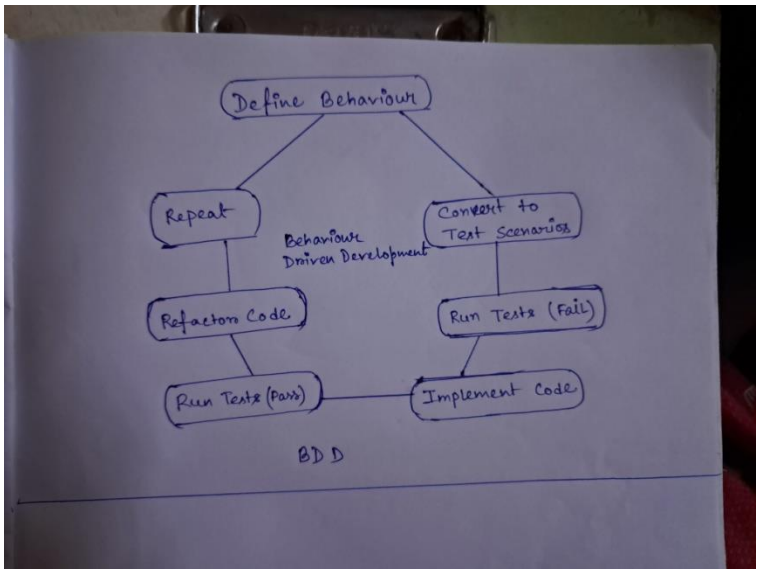
1. Write a Test - Start with a specific test for new functionality.
2. Run Tests (Fail) - Ensure the new test fails initially.
3. Write Code to Pass - Write just enough code to pass the test.
4. Run Tests (Pass) - Run all tests to ensure they pass.
5. Refactor Code - Clean up the code without changing functionality.
6. Repeat - Continue the cycle for each feature or fix.

Benefits of TDD

1. Bug Reduction - Early defect detection.
2. Better Design - Modular, decoupled code.
3. Documentation - Tests as living documentation.
4. Confidence in Changes - Safe refactoring and feature addition.
5. Continuous Integration - Supports automated testing.

Assignment 2: <u>Produce a comparative infographic of TDD, BDD, and FDD methodologies. Illustrate their unique approaches, benefits, and suitability for different software development contexts. Use visuals to enhance understanding.</u>

Behavior-Driven Development (BDD)



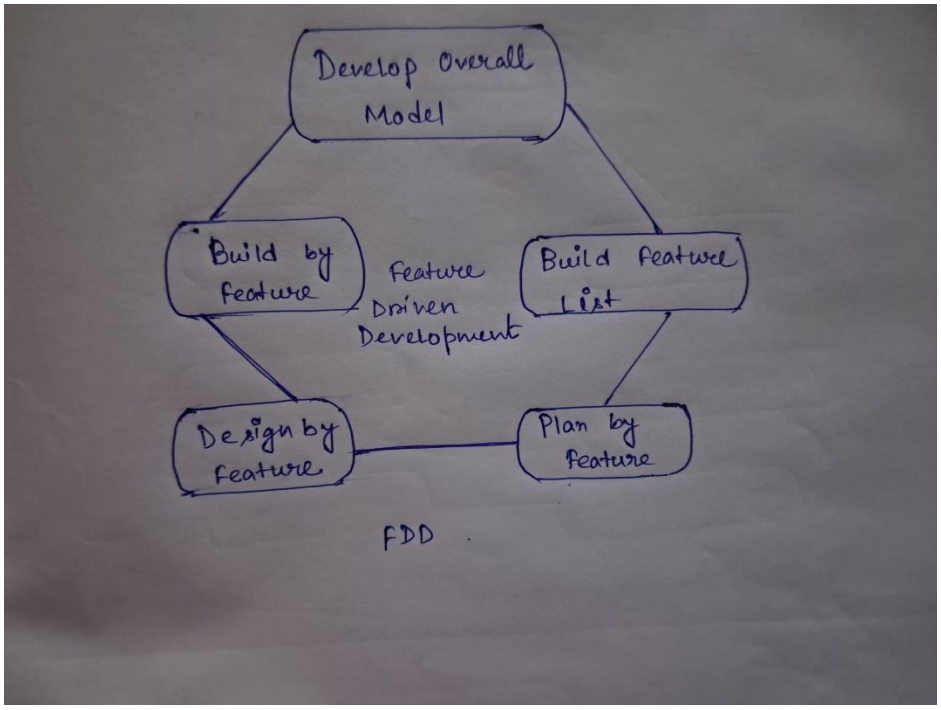
BDD Cycle Steps

1. Define Behavior - Write user stories in plain language.
2. Convert to Test Scenarios - Translate stories into executable tests.
3. Run Tests (Fail) - Ensure new scenarios fail initially.
4. Implement Code - Write code to fulfill the behavior.
5. Run Tests (Pass) - Execute tests to ensure they pass.
6. Refactor Code - Clean up the code while keeping functionality intact.
7. Repeat - Continue the cycle for each user story.

Benefits of BDD

1. Enhanced Collaboration - Improves communication between stakeholders.
2. Clear Requirements - Ensures understanding of requirements.
3. Reduced Misunderstandings - Aligns development with business goals.
4. Living Documentation - User stories and tests serve as documentation.
5. Improved Quality - Focuses on user behavior and expectations.

Feature-Driven Development (FDD)



FDD Process Steps

1. Develop Overall Model - Create a high-level domain model.
2. Build Feature List - Identify and prioritize features.
3. Plan by Feature - Assign features to iterations.
4. Design by Feature - Design feature-specific solutions.
5. Build by Feature - Implement features in small iterations.

Benefits of FDD

1. Scalability - Effective for large-scale projects.
2. Feature Focus - Prioritizes delivering customer value.
3. Predictable Outcomes - Clear progress tracking with features.
4. Improved Team Productivity - Structured and efficient workflow.
5. Regular Updates - Frequent delivery of working features.

Key Differences Between TDD, BDD, and FDD

Test-Driven Development (TDD)

- Focus: Writing tests before code to ensure functionality and drive design.

Behavior-Driven Development (BDD)

- Focus: Specifying behavior in plain language to enhance collaboration between developers and non-technical stakeholders.

Feature-Driven Development (FDD)

- Focus: Delivering software in iterative, feature-based cycles to manage large-scale projects efficiently.