

# Assignment-3

## (Modern development methodologies)

Prakash Manikanta Irrinki  
Prakashnaidu9494@gmail.com

### Assignment-1:

Create an info graphic 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.

### Test Driven Development [TDD] Process:

#### Introduction to Test - Driven Development (TDD)

- Definition of TDD
- Significance and benefits of TDD

#### The TDD Cycle

Step1: Write a failing test

Step2: Run the test (and see it fail)

Step3: Write the minimum code to pass the test Step 4: Run the test (and see it pass)

Step5: Refractor the code

Repeat the cycle for each new feature or functionality

#### Benefits of TDD

- Early bug detection and prevention
- Improved code quality and reliability
- Better code documentation through tests
- Modular and flexible code design
- Increased confidence in code changes and refactoring

#### How TDD Fosters Software Reliability

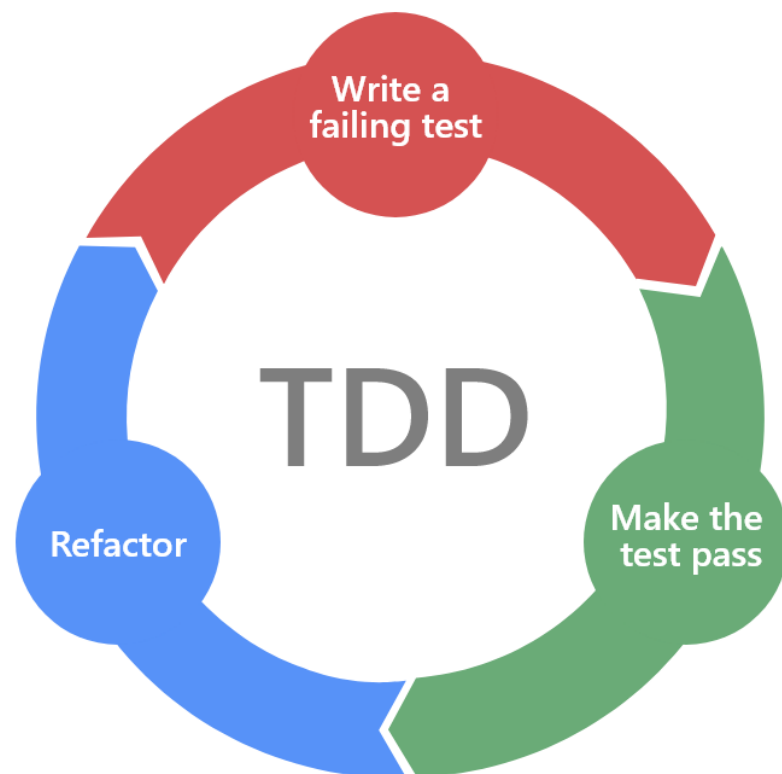
- Tests act as a safety net for the code base
- Regression testing with each code change
- Encourages modular and test able code design
- Facilitates continuous integration and delivery
- Enables refactoring and code maintenance with confidence

## Challenges and Best Practices

- Initial learning curve and mindset shift
- Writing good tests (FIRST principles)
- Test code organization and maintenance
- Balancing TDD with other development approaches

## Conclusion

- Summary of TDDs benefits and impact on software reliability
- Encouragement to adopt TDD practices.



## Assignment2:-

Produce a comparative info graphic of TDD, BDD, and FDD methodologies. Illustrate their unique approaches, benefits, and suitability for different software development contexts. Use visuals to enhance understanding.

### Test- Driven Development (TDD):

#### Approach:

- Developers write tests before writing production code.
- Focuses on writings small, incremental tests to drive the development process.

#### Benefits:

- Early Bug Detection: Catch bugs early in the development process.
- Improved Code Quality: Encourages clean, modular code design.
- Increased Confidence: Provides a safety net for refactoring and code changes.

#### Suitability:

- Ideal for projects with clear and well-defined requirements.
- Best suited for small to medium – sized projects with a focus on code reliability.

### Behavior-Driven Development (BDD):

#### Approach:

- Focuses on behavior and outcomes rather than implementation details.
- Uses natural language specifications (e.g., Given-When-Then) to define tests.

#### Benefits:

- Enhanced Collaboration: Promotes collaboration between developers, testers, and stakeholders.
- Improved Communication: Help ensure alignment between technical and non-technical team members.
- User-Centric: Tests are written from the perspective of end-users, ensuring that features meet their needs.

#### Suitability:

- Suitable for projects with complex business logic and evolving requirements.
- Best suited for teams that prioritize collaboration and communication.

### Feature-Driven Development (FDD):

#### Approach:

- Focuses on building features incrementally based on client priorities.
- Emphasizes short iterations and frequent client feedback.

**Benefits:**

- Incremental Delivery: Delivers tangible results to clients in short cycles.
- Client-Centric: Aligns development efforts with client priorities and business objectives.
- Scalable: Scales well for large, complex projects with multiple teams.

**Suitability:**

- Suitable for large-scale projects with evolving requirements and multiple stakeholders.
- Best suited for projects where client involvement and feedback are essential.

**Conclusion:**

Each methodology offers a unique approach to software development, catering to different project requirements and team dynamics. Whether it's the test-driven approach of TDD, the collaborative nature of BDD, or the feature-centric approach of FDD, choosing the right methodology depends on factors such as project size, complexity, and stakeholder involvement.