OVERALL DESCRIPTION ABOUT TDD, BDD, & FDD

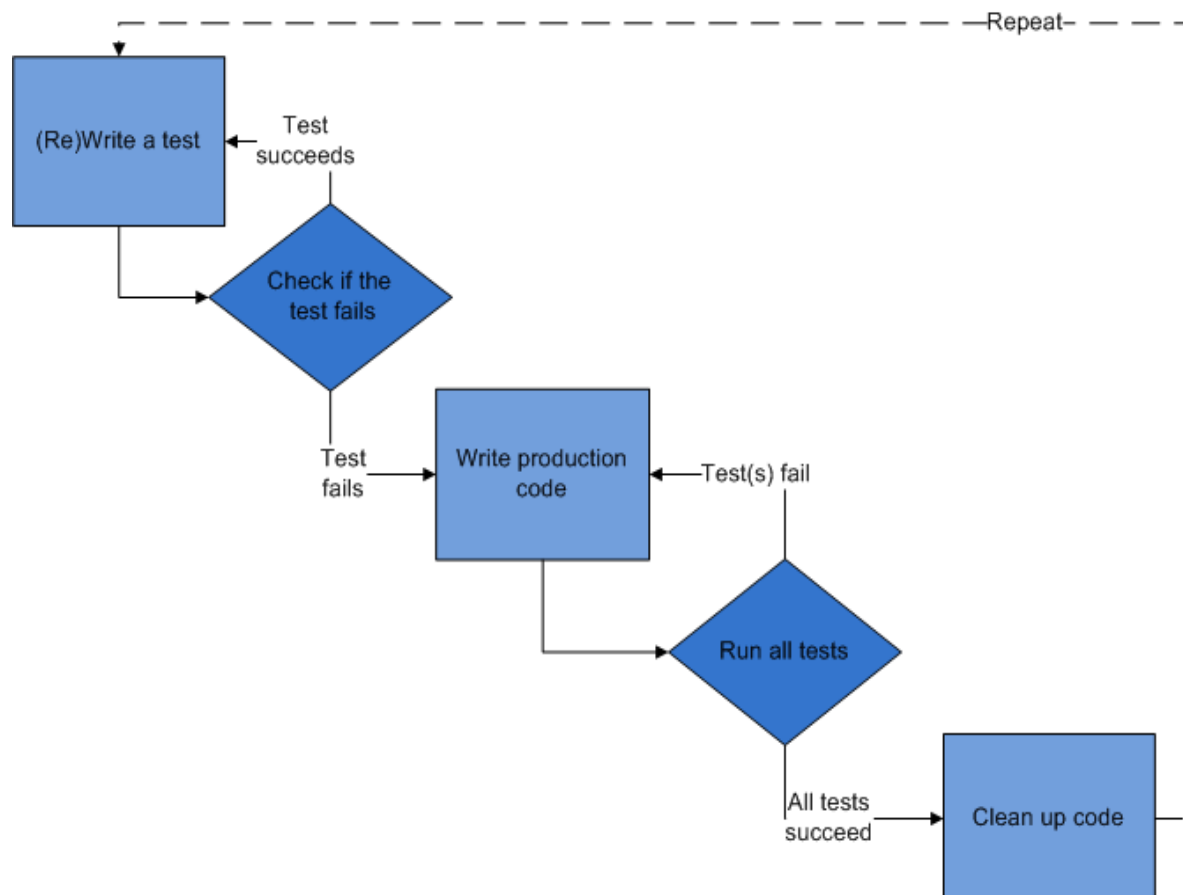**Approach:-** Write tests before writing code.

Benefits:-

Early bug detection.

Improved code quality.

Refactoring confidence.

Disadvantages:-

Can be time-consuming.

Requires mindset shift.

Suitability:-

Agile projects.

Small to medium-sized teams.

Projects with evolving requirements.

Approach:- Define behaviour through examples.

Benefits:-

Focuses on user needs.

Encourages collaboration.

Living documentation.

Uses cucumber framework(which is in "Gherkin" language).

Disadvantages:-
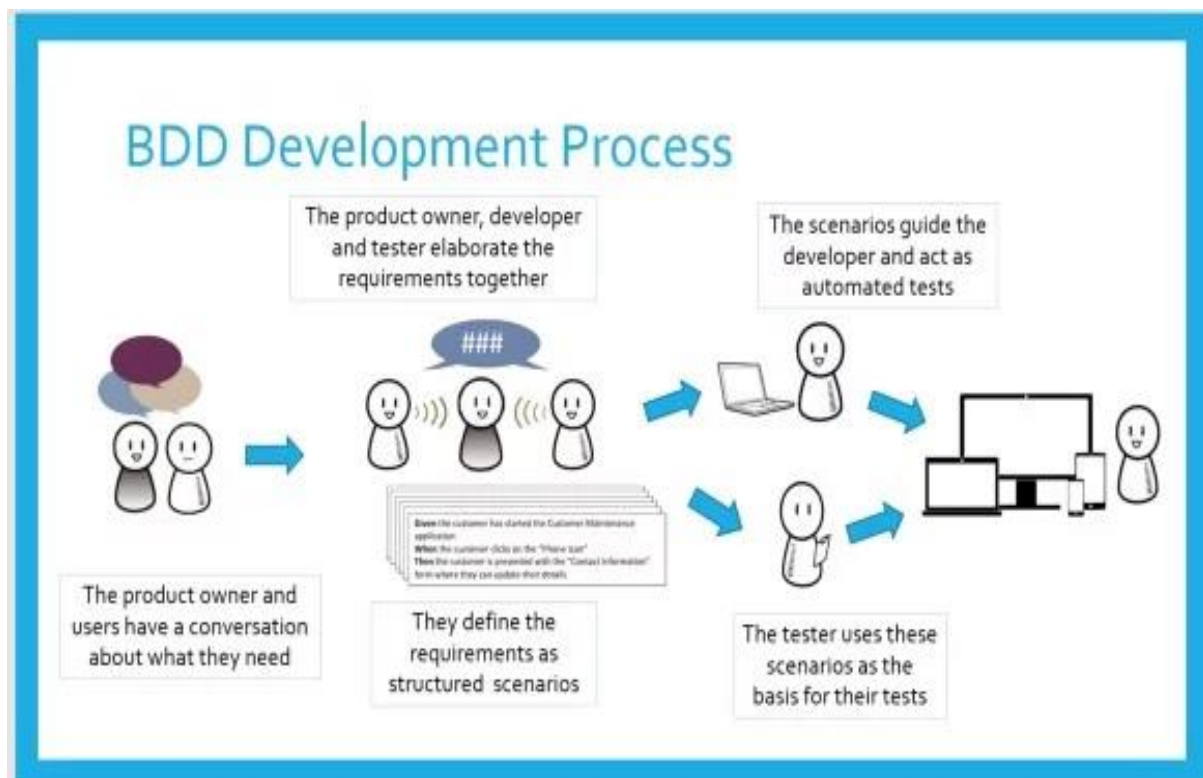
Learning curve for non-technical stakeholders.

Initial setup overhead.

Suitability:-

Customer-centric projects.

Cross-functional teams.

Projects with complex business logic.

Approach:- Breaks development into feature sets.

Benefits:-

Emphasizes progress tracking.

Clear feature ownership.

Structured development process.

Disadvantages:-

Requires upfront planning.

Less adaptable to changing requirements.

Suitability:-

Large-scale projects.

Well-defined requirements.

Experienced development teams.