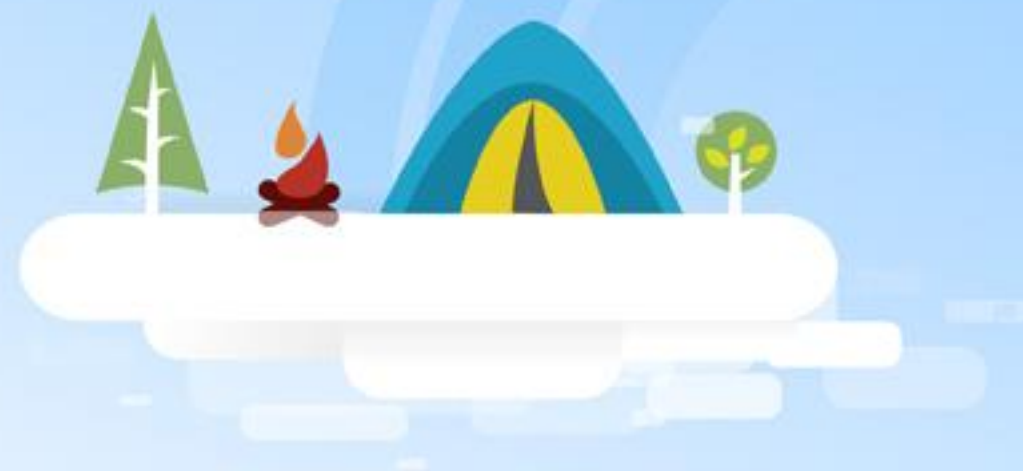


Test Driven Development

Ridi Ferdiana | ridi@acm.org

Version 1.0.0



Agenda

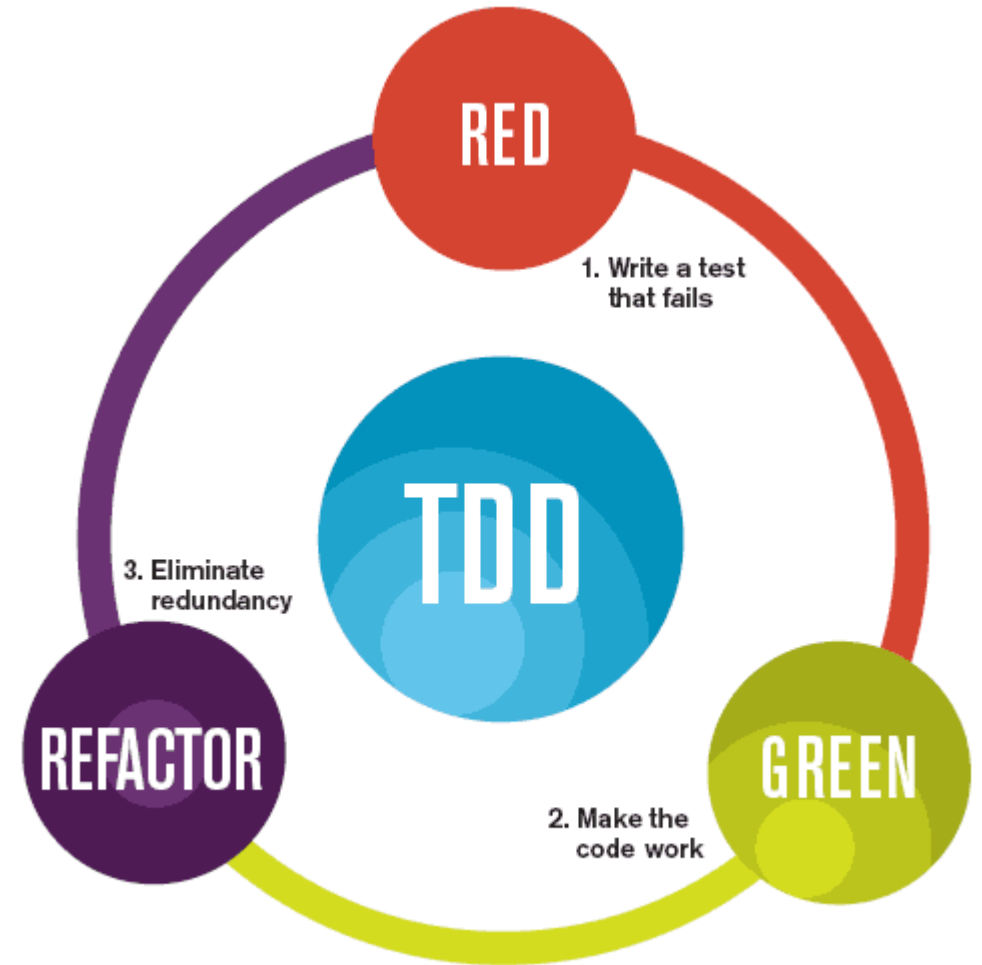
- What is Test Driven Development (TDD)?
- Test Driven vs Feature Driven
- TDD Benefits and Limitation
- TDD in Visual Studio

What is TDD



What is TDD

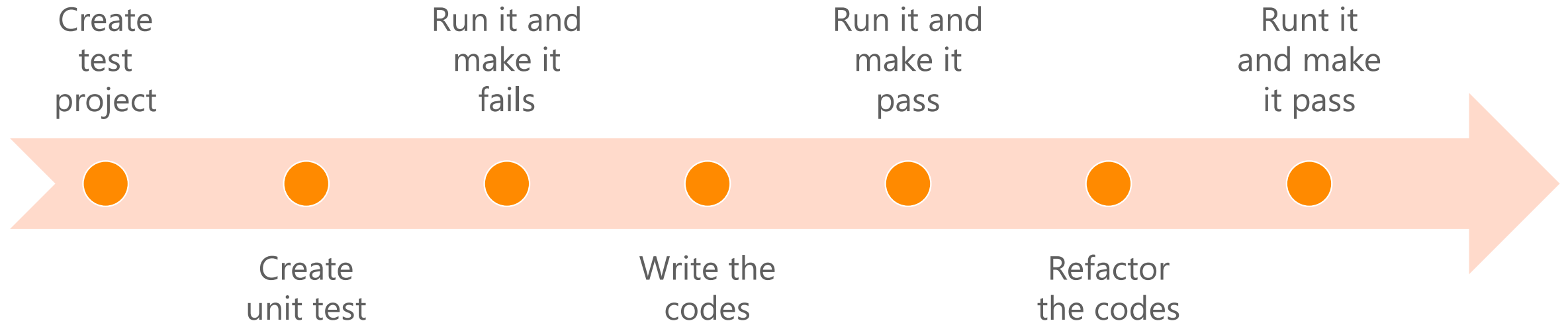
- Test First , Codes Later
- Method of developing software not just testing software
- Popular in eXtreme Programming Method.



The mantra of Test-Driven Development (TDD) is "red, green, refactor."

[This Photo](#) by Unknown Author is licensed under [CC BY-SA](#)

TDD Process



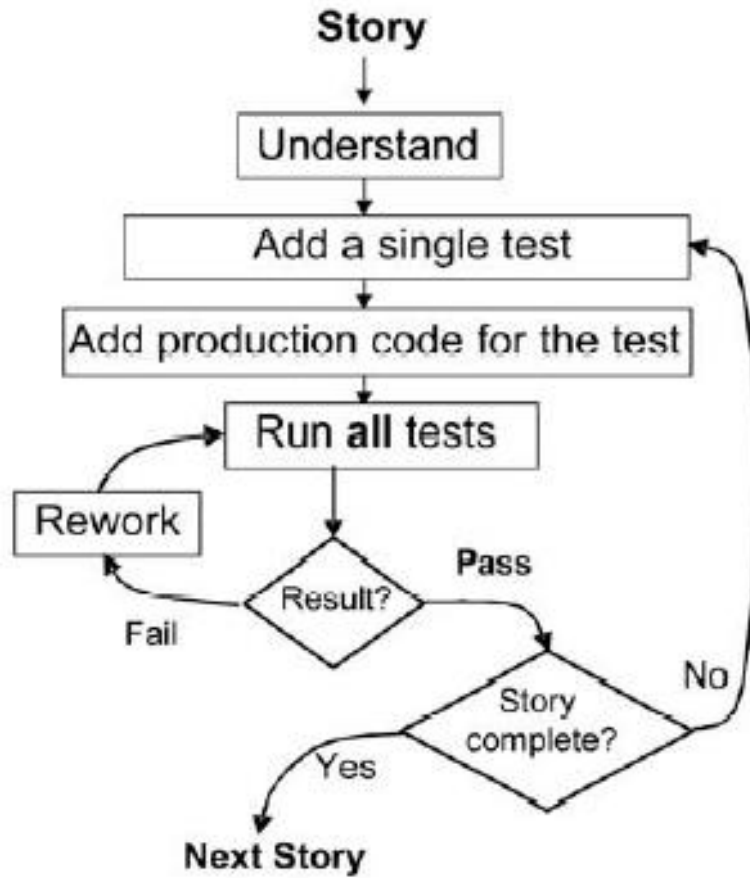
TDD vs FDD



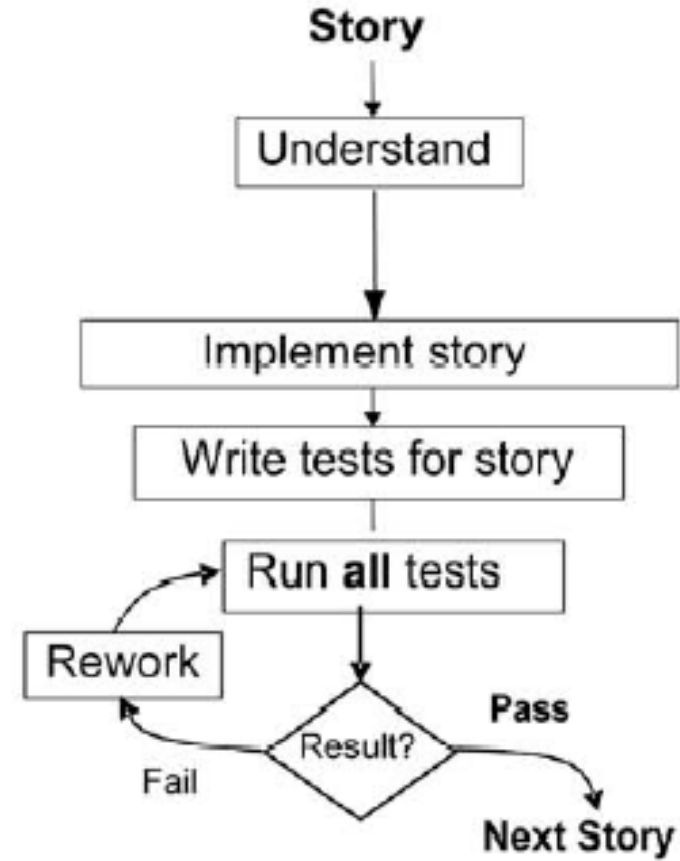
Test Driven vs Feature Driven

- Pick a piece of functionality
 - Write a test that expresses a small task that fails
 - Write production code until test passes
 - Run all tests
 - Rework code until all tests pass
 - Repeat [1]
- Pick a piece of functionality
 - Write production code that implements entire functionality
 - Write tests to validate all functionality
 - Run all tests
 - Rework code until all tests pass [1]

TDD vs FDD



[1]



[1]

TDD Benefits & Limitation



Benefits

- Instant and automated feedback
- Better development practices
 - Decompose the codes
 - Low level design support
 - Choosing the codes
- Quality Assurance
- Lower Rework

Limitation

- Counterproductive and hard to learn
- Difficult in Some Situations
 - GUIs, Relational Databases, Web Service
 - Requires mock objects
- TDD does not often include an upfront design
 - Focus is on implementation and less on the logical structure

TDD in Visual Studio



Demo

Creating Unit Test

Demo

Creating Live Unit Test

Demo

Creating JustMock project

Keypoints

- The TDD focus on
 - Quality
 - Productivity
- TDD can be effective if you consider
 - Goals of your software group
 - Kind of software being developed
 - The skill level and experience of your developers

References

- Justin Kadrmas & Dawn Jones. 2008. Test Driven Development