



**BITS Pilani**  
Pilani | Dubai | Goa | Hyderabad

**TDD**

**Sonika Rathi**

Assistant Professor  
BITS Pilani

# Agenda

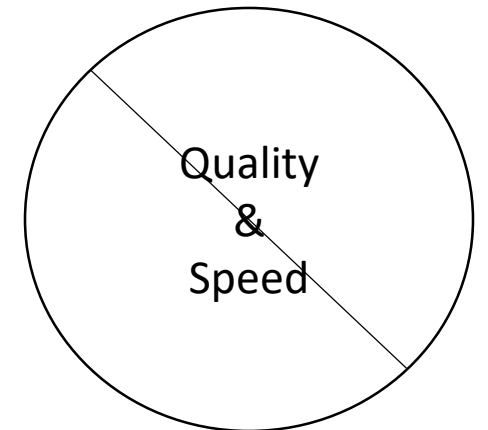
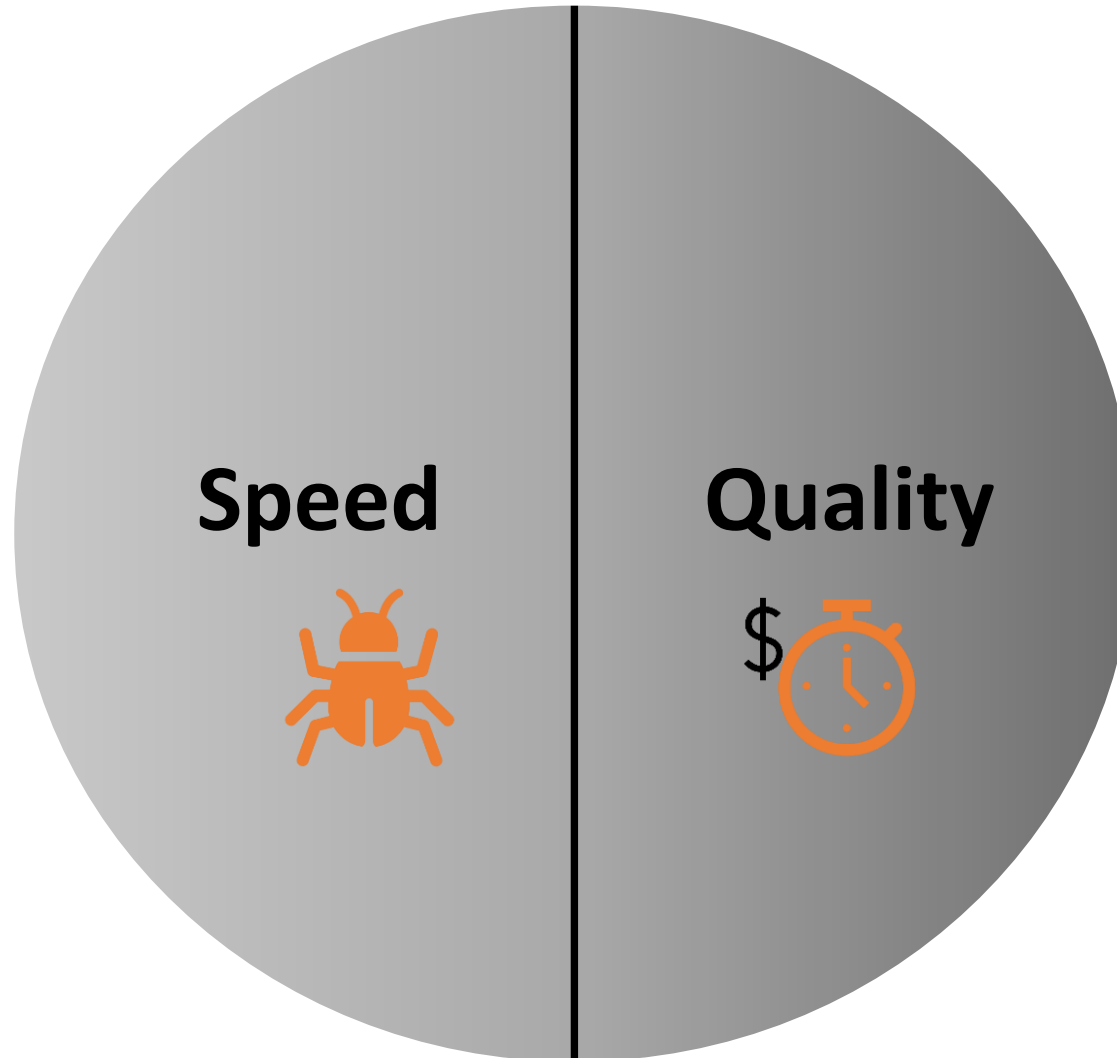
## TDD

- What is TDD & what it is not
- TDD Cycle
- TDD Cycle example
- What TDD brings



# Clear the basics

## Speed and Quality



# TDD

Claims are True???



**Defects**



**Productivity**



**Quality**



SILVER BULLET?

# What TDD is?

What it is not

## Definitions



**Automated Testing**



**Unit testing**

Test-last  
Test-first



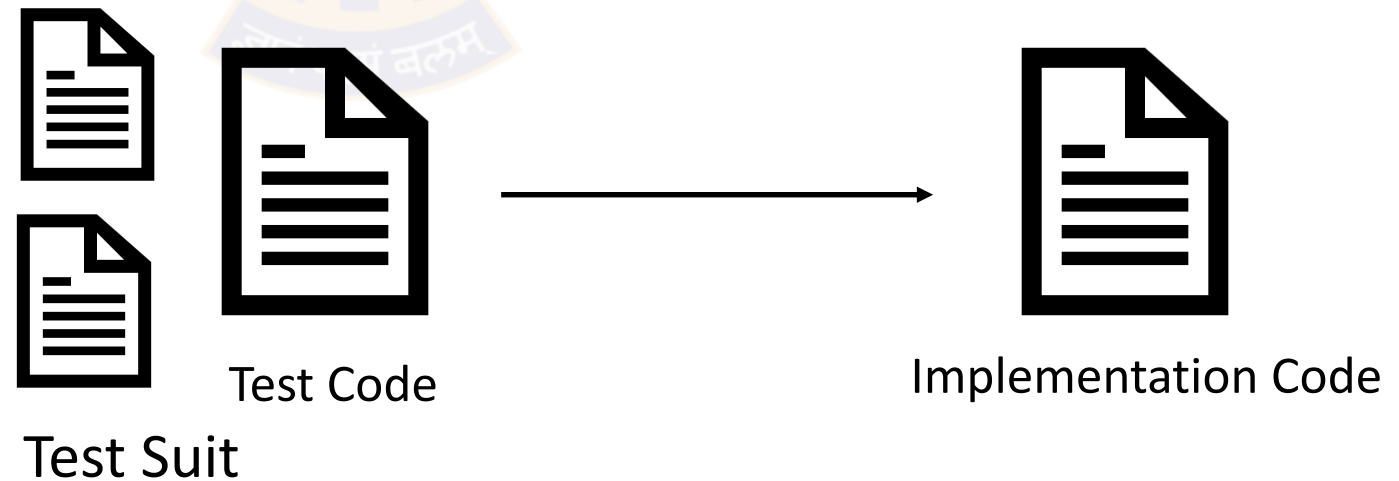
**Code Coverage**

# Lets Start

## Automated testing



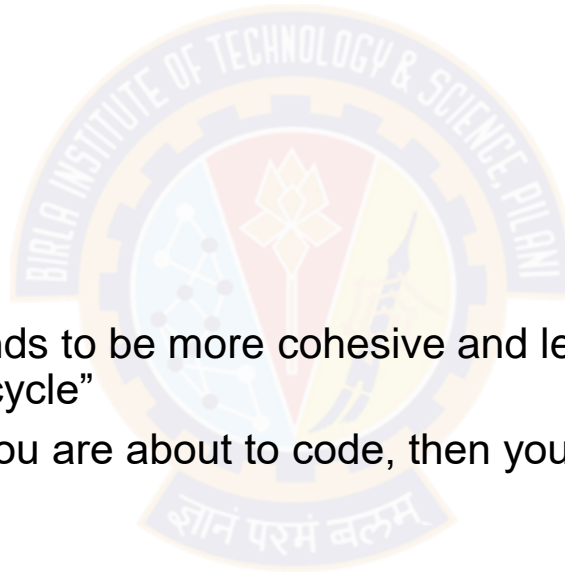
☒ **Automated Testing**



# TDD

## Define TDD

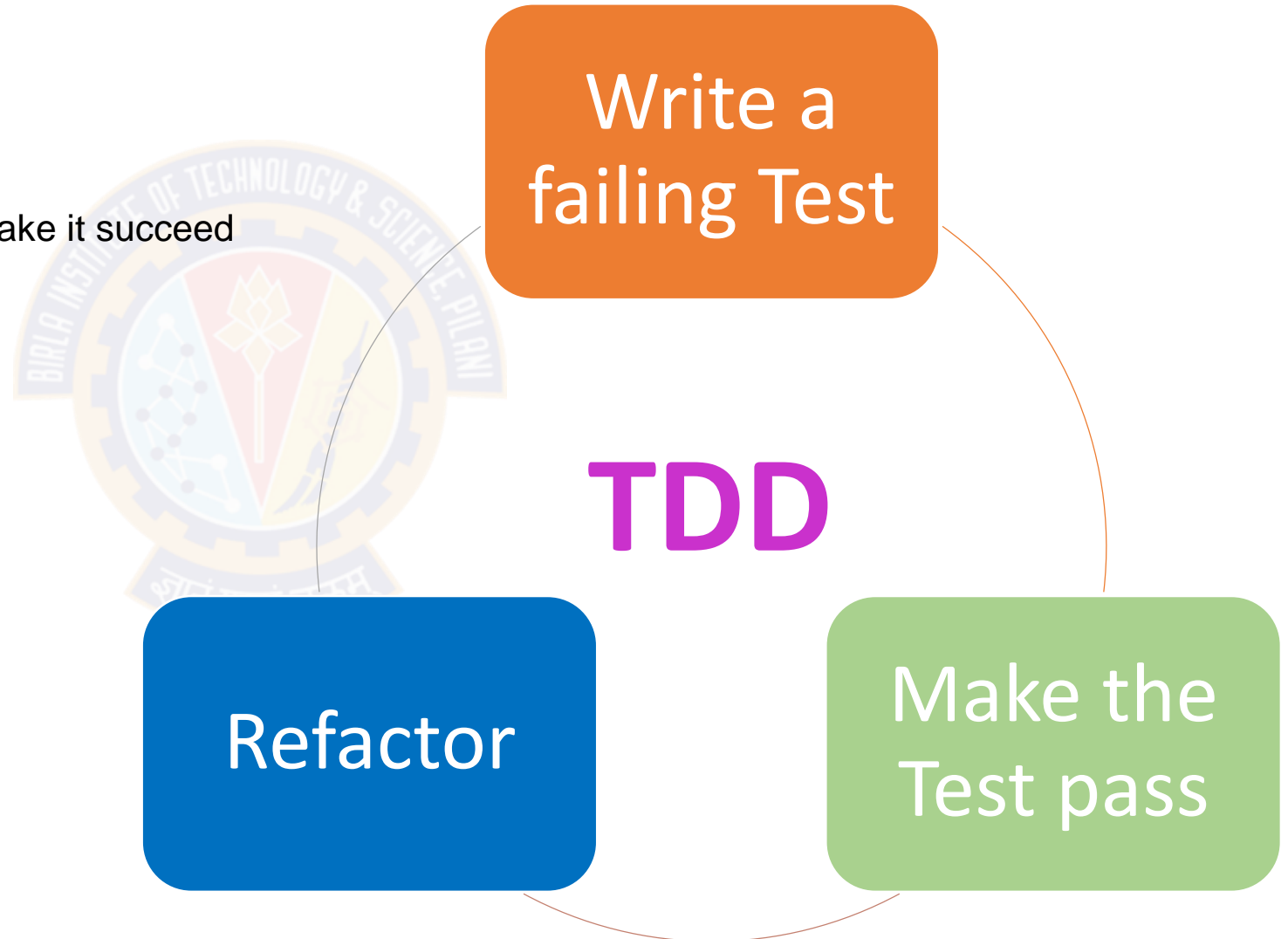
- Many Names
  - Test driven development
  - Test drive design
  - Emergent design
  - Test first development
- TDD Quotes
  - Kent Beck said “Test-first code tends to be more cohesive and less coupled than code in which testing isn’t a part of the intimate coding cycle”
  - “If you can’t write a test for what you are about to code, then you shouldn’t even be thinking about coding”



# TDD

## TDD cycle

- RED
  - Write a new TEST which fails
- GREEN
  - Write simplest possible code to make it succeed
- REFACTOR
  - Refactor the code





# TDD Cycle

## Red, Green and Refactor

- Example (payment gateway task)

1. Choose a small task
2. Write a failing test (**RED** Test)
3. Write simplest code to make the test pass (**GREEN** Test)
4. **REFACTOR**
5. **Repeat**

## Implementation Code

```
payment_gateway = function () {  
    credit_card ();  
    netbanking ();  
    upi_payment ();  
}
```

→ `all_upi_payment ();`

Run all  
test

## Tests

.....  
.....  
.....



```
new_test= function () {  
    payment_gateway_includes_upi_option ();  
}
```

# How TDD is different

## What so special about TDD

- Isn't unit testing is same thing?
- TDD is the process / method
- Order : TDD drives the code
- TDD is a skill of thinking about the problem you are trying to solve before actually trying to solve it



### Unit testing

Test-last / Test-after  
Test-first



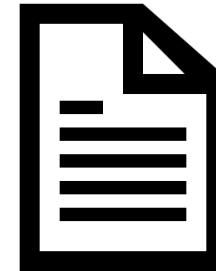
Test Code



Implementation Code



Test Code



Implementation Code

# TDD & Code Coverage

## What is Code coverage

$$\frac{800 \text{ lines executed by tests}}{1000 \text{ lines of production code}} = 80\% \text{ code coverage}$$

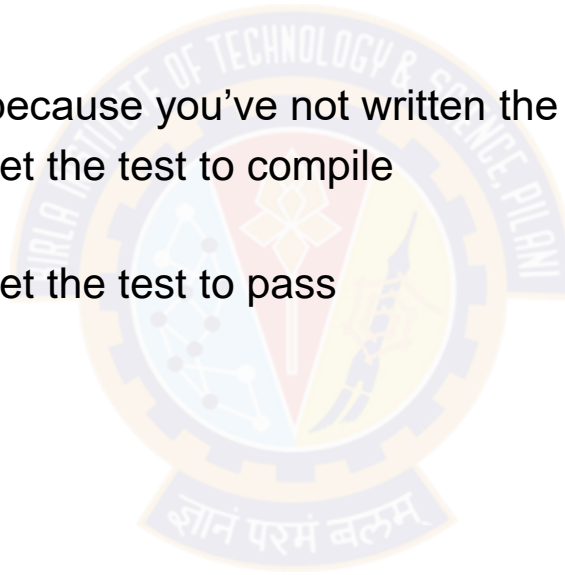


**Code Coverage**

# TDD

## TDD Overview

- TDD Steps:
  1. Write a single test
  2. Compile it. It shouldn't compile because you've not written the implementation code
  3. Implement just enough code to get the test to compile
  4. Run the test and see it fail
  5. Implement just enough code to get the test to pass
  6. Run the test and see it pass
  7. Refactor for clarity
  8. Repeat



# TDD

## TDD Brings

- TDD can lead to more modularized, flexible, and extensible code
- Clean code
- Better code documentation
- More productive
- Good design



### TDD increases quality

Code and test quality

### TDD impacts productivity

Decreases in short-term  
Increases in long-term



# Thank You!

In our next session: Need of DevOps