RESEARCH

**When and why BDD was defined**

**The most important aspects of BDD**

BDD makes a priority to the behavioural aspect of the system, instead of the implementation aspect of the system, just like TDD. It also gives you a clear understanding of what the system should do from the point of view of the customer, and the developer. It gives the developer quite a better understanding of what the system is meant to do.

**Why is useful and when could be applied.**

Behavior Driven Development is really useful when this is combined with many other good practices, it is a good way of a shared understanding.

**Which type of teams could implement BDD**

Teams who

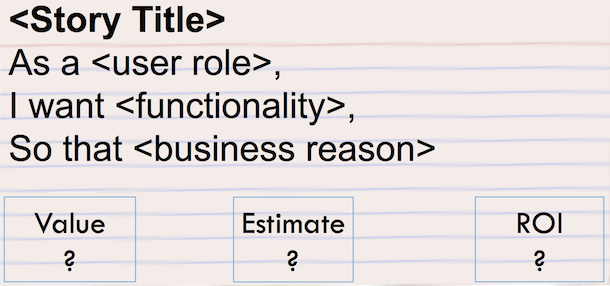
* Rely on a good communication between business and development and have a commong language
* Need the business needs directly tied to the code that is written.
* Want to maintain and understand code easier.

**Structure of a feature – Give an example**

Feature: withdrawal cash

**Structure of a user story – Give an example**

The structure of a User Story is something like this image



For example

**As an** *Account Holder*

**I want** *to withdraw cash from an ATM*

**So that** *I can get money when the bank is closed*

**Structure of a scenario – Give an example**

There are three steps which are:

Given: These are the necessary steps to put the system in the state you want to test.

When: User interaction that triggers the functionality we want to test.

Then: In this step we can see the changes in the system and see if they are the desirable ones.

Let’s keep on the same example mentioned before.

Scenario 1: Account has sufficient funds

***Given*** the account balance is Bs. 80

And the card is valid

And the machine contains enough money

***When*** the Account Holder requests Bs. 20

***Then*** the ATM should dispense Bs. 20

And the account balance should be Bs. 60

And the card should be returned

**Differences between BDD and BDT**

Some differences are that BDD gets its focus on behavior and specification, so linear and points of view who prefer something like waterfall minds get kinda confused, BDD is more easily understood as a design practice and not as a testing practice.

TDD focus most about testing where tests comes after implementation, sometimes leading to misunderstanding and wrong behavior.