## **What is the difference between a product based and a service company?**

## **Product Based:**

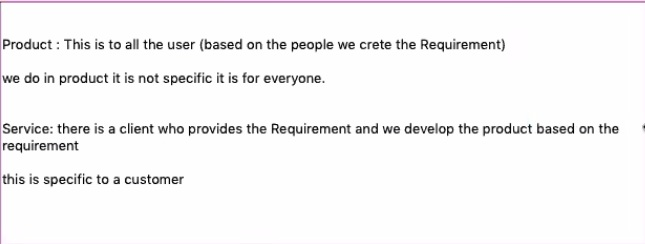
Product: end user, developed for the mass . This is all user based (based on user requirement)

This is not specific

## **Service based:**

service is made from a client and developed for a certain kind (Client provides the requirement and we develop a product that is for that client)

this is specific to the customer



Moolya is a product (Bugasura) and Service based organization

# **Who approaches a client first, client or moolya**

moolya and client can approach each other

## **when a client approaches to moolya**

The client will have requirements and have an open bid (cost, quality, tools, processing) the organizations will provide these features and whichever they feel would be a fit

## **If moolya approaches a client**

They will go through the HR has a business developers

They interact with the BA

### **Business Analyst**

Domain expert

explain what they do

These days, BA are trained technically

**BDD**

Behavior driven developement

They create and give it to developer and tester

Primarily they are BDD strong

It is a frame work

**What are the requirement they gather from the client?**

1. Estimation
2. Effort
3. Tools
4. process
5. People
6. overall cost estimation

**The estimation is given to the client**

One of two things happen

1. client agrees to the estimation
2. or they negotiate

This is aka **Request for Proposal Terms and Conditions (RFP)**

**SRS (Software/system Requirement Specification)**

**BRS(Business requirement specification)**

SRS and BRS provides the complete requirement

FRS: Functional Requirement Spec is given (like a mind map) so that they can start saving time

**Jar**

Desktop application

Java Archive File

# **War**

Web application, can be used in a mobile form

Website Archive File

# **UAT**

User Acceptance Testing

This done after the testing and deployment is done

They are continuous and will occur when it is used with clients

# **What do we do when we find a defect during testing?**

We will bring it up to Bugasura which is a defect management framework.

## **Bugasura**

This is a defect management framework created by moolya

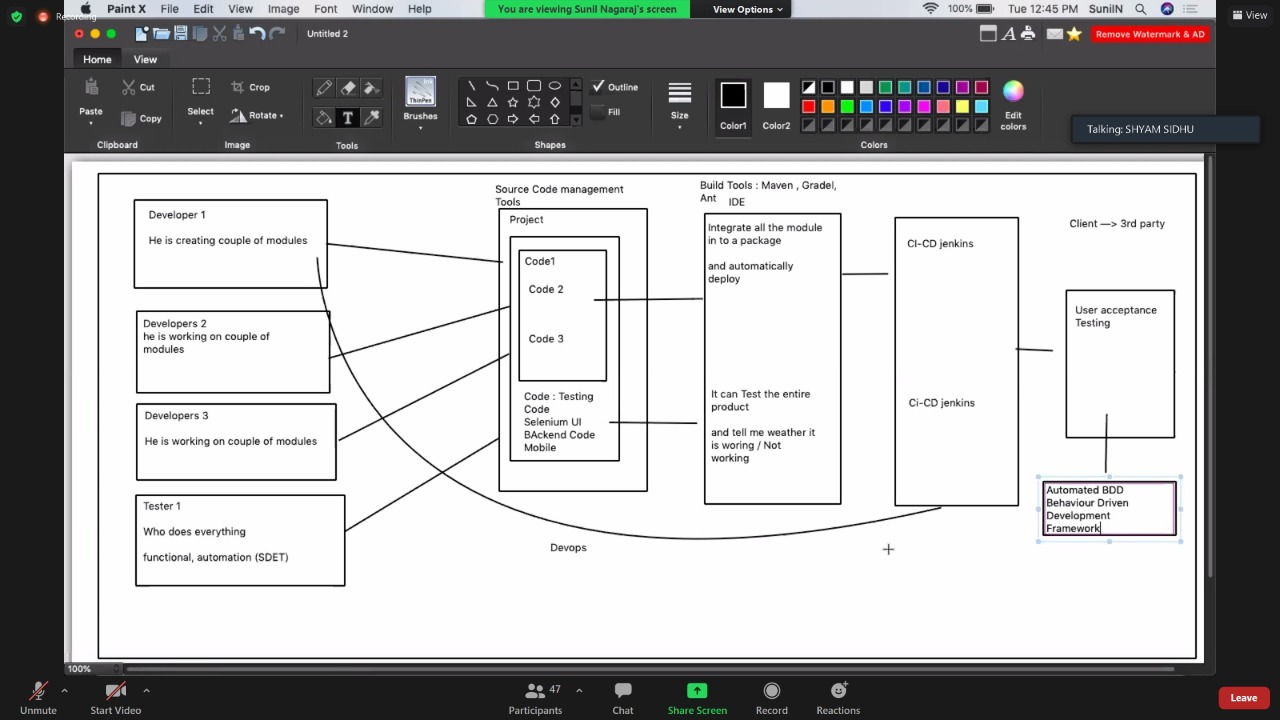
### **Other defect management tools:**

1. Jira (Agile): it is an issue tracking tool aka project management tools
2. ALM (Mircofocus): application life cycle managment
3. SCTM:
4. MTM: Microsoft test management
5. Rational IBM

## **Open source tool**

1. Java (IDE) eclipe open source
2. mobile —> appium (Open source: OS)
3. API —> Rest assured
4. post man (OS)
5. soapui (OS)
6. soapui pro (paid)
7. github jenkins (os)
8. maven os
9. BDD (cucumber)

Mantra of IT Industry



# **What is Mind Mapping (technology on the web)**

Make a mind map of what tests I want to do

Mind maps of what we want to do in each process

# **Mind map of testing (Requirement Tractability Matrix)**

**The document we make in testing AKA RTM Requirement Tractability Matrix AKA Software Testing Life Cycle:**

→ test strategy

→ test plan

→ test scenario (test idea)

→ test cases

→ test environment

→ test execution

→ test report

→ defect report

→ test summary report

### **1. What is Test Strategy**

* it is a one time document
* It is for the whole product
* It contains the context, objective and goals to achieve
* It also contains the environment for the product

### **2. What is test Plan**

* It’s a document created for all the levels of testing
* Levels of Testing
  + Unit testing: Developer (junit) jar, TDD (Test Driven Development) → Module/ unit/ piece of code and test the functionality
  + Integrating testing: Developer & tested do functional testing, integrate the module and test functionality to test the data flow between two modules
  + System testing: Testing team (End-End of the product to ensure functionality, performance, security) → To give the confidence to put the product into production, I have to make sure all the critical features are working
  + UAT: Client aka end-user does this (BDD), client ensures all the features are working, Retrospecting meeting to meet and talk about what could be done faster and easier

### **3. Test Scenario + Test Case = Test Design**

Test design is the activity of deriving and specifying test cases from test conditions to test software.

Test Scenario: on the basis of requirement, we take each module

Test Case: according to the Test Scenario we will make our test case design

### **4. Test Environment**

We run the product in different environment (like windows, linux, ubunto, etc)

But say that the client wants the test run in Ubuntu, so we will have the environment set up

It also include the wifi, internet used, hardware, software, OS

### **5. Test Execution**

All the test, we will do it in this phase

The test cases written to make the test scenario, we will check if the test passes in the environment

### **6. Test Report**

We will make a report on the details of the test

It is module

We will report the following

1. Number of tests
2. Number of test passed
3. Number of test Failed
4. Number of defects found
5. Number of defects raised

### **7. Defect Report**

If there are any defects, it will be recorded in this report

This is part of the test report, that is:

1. Number of test failed
2. number of defects found
3. number of defects raised

### **8. Summary Report aka Test Closure**

The above processes can be divided among the team, but the Summary Report contains all the teams report

This report mainly deals with if a product can be put into production or if the product needs to go back to the developers and the cycle begins from test execution

# **Two type of testing**

1. Static
2. Dynamic

## **1. Static testing (Validation)**

with documentation

Without any code excutation

No code required (no product)

It is necessary because it will create a strategy

### **Process of Static testing (documents produced in order)**

**Test Design (Test Scenario and test Case)**

test strategy → test plan → test scenario (test idea) → test cases → test environment

→ test execution → test report → defect report → test summary report

### **Example of test scenario**

Is a high level design that contains the test case

### **Example of test case**

It is the detailed test for the test scenario

1. login
2. language

like so

## **2. Dynamic (Verification)**

It is Validation

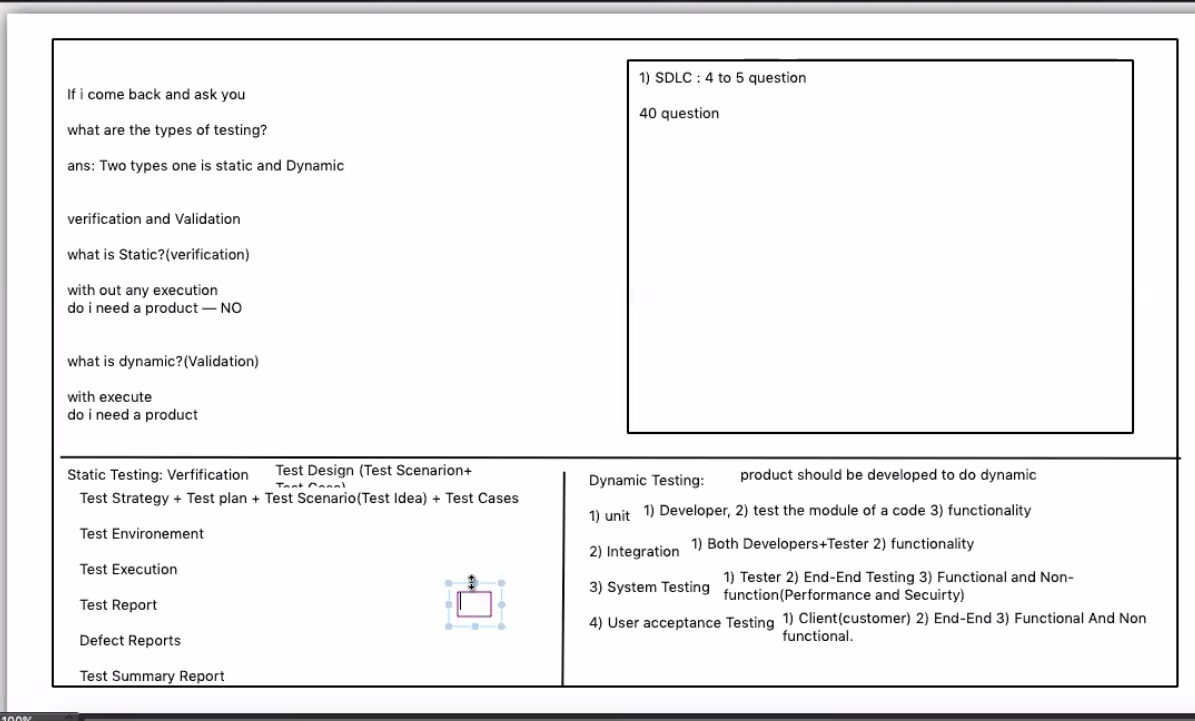
So that we can find bugs in the early stage, and we won’t spend time money on rectifying the bug

If we find the defect in the later stages, it will cost 4 times the money

Code required

### **Types of Dynamic Testing AKA levels of testing**

1. Unit: 1) Developer, 2) tests the module of the code, 3) only the functionality they test
2. Integration: 1) (developer and tester) , 2) only the functionality
3. System Testing: 1. tester, 2. End- end testing, 3. Functionality and non functionality (performance and security)
4. User acceptance testing: 1. client (customer), 2. End to end, 3 functionality and non functional



# **requirement phase**

In this, everyone will come together and understand what the product is

then designer will go to design the product

Everything is happening simultaneously

# **Where to find all the details of testing**

On coda, shortcuts

# **Bug/ defect**

It is a non-confirmance to requirement

If the product functionality does not work as expected in the requirement then we call it a bug

# **Error**

Syntax is not correct

When the developer is not able to compile the code

Like missing semicolon, missing indentation, spelling mistake, etc.

# **Failure**

After deploying the product, if the customer find anything going wrong, we call it as failure