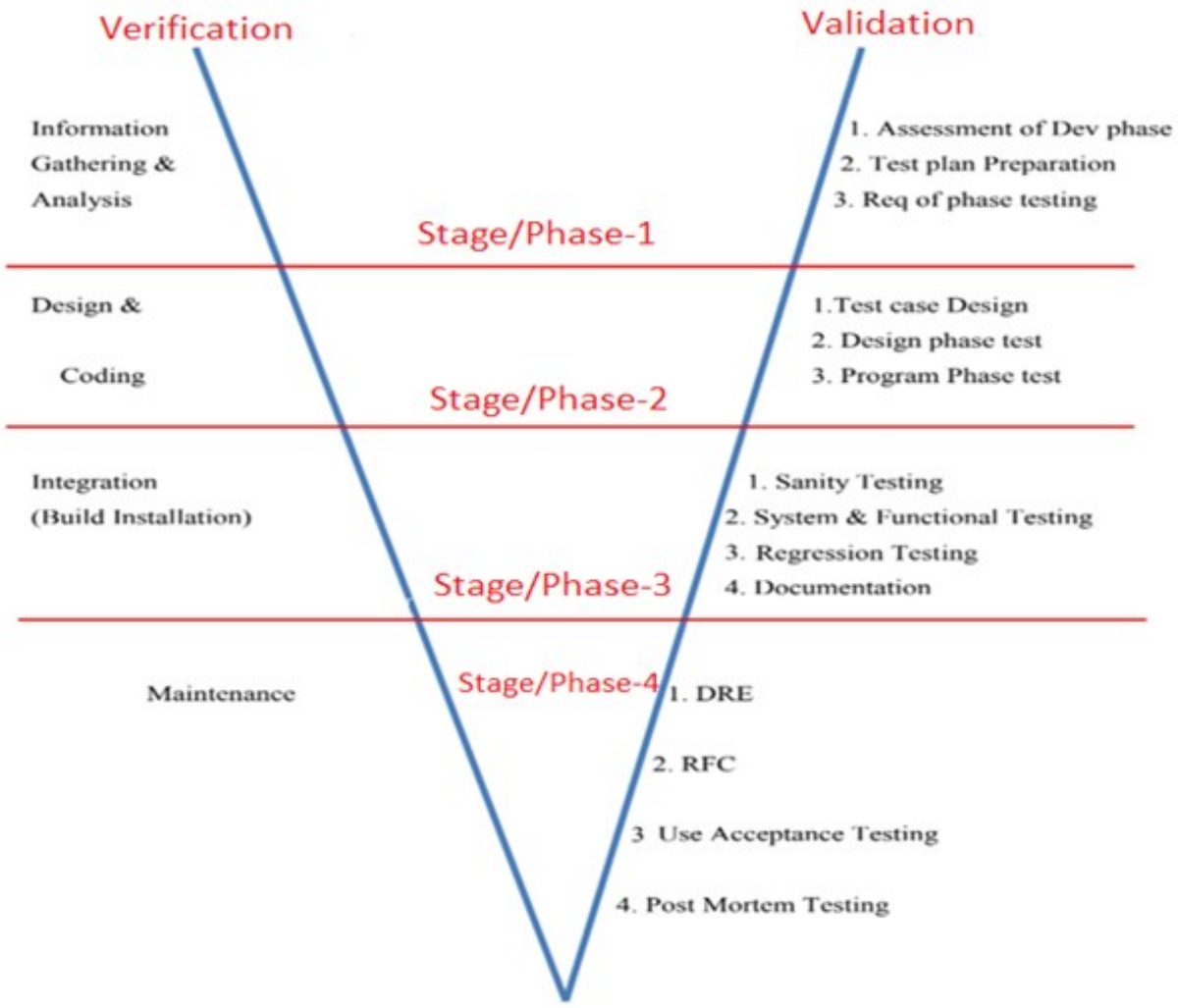
V Model

- **V model is a Extension of the Waterfall Model**
- V Model is a **combination** of the Verification and Validation
- In V model **Verification and Validation is running parallelly**
- In V Model **Developement Stage** are **mapped** with the **Testing Stage.**
- We can accept **CR(Change Request) at any stage/phase.**
- EX- when **1st** stage got **completed** and developement of **2nd stage** is **going on**, and client comes with **new requirement**, then we can **revert back** to the 1st stgae. But for that **Client have to pay extra amount for that.**
- V model is used in **Big Organization** and the Expected delivery time of the product is around **3 Months**



Stage/Phase-1

Assessment of Dev Phase: (PM / Leads)

- In this step **PM or TL** are involved.
- They decide the **Methodology** and **Stratergy** for the Testing.
- Methodology: (Manual, Automation, API, Database, Perfomanance)
- Stratergy: (Java, Python, Ruby, etc.. Postman, RestAssured, MySQL, SQL servermanager, Junit, Nunit.)
- Also They are take care of the TRM**(Test Responsiblity Matrix**)
- TRM- **mapping of work and resources.**

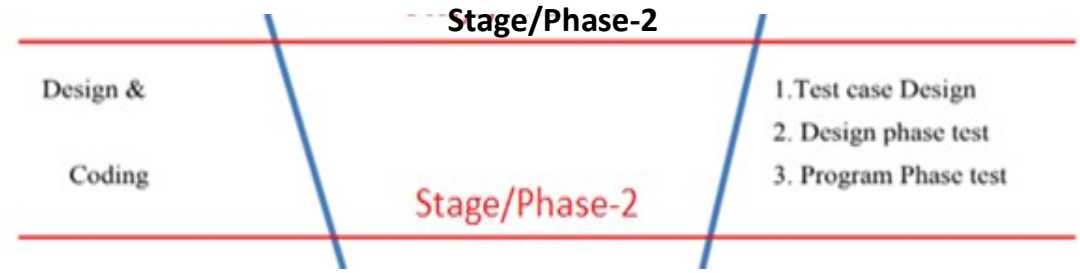
Test Plan Preparation: (PM/QA-Lead)

- In this step **PM or QA-Lead** are involved.
- In **Test Plan Preparation, PM/QA Lead** is responsible for **implementataion** of **TRM**
- They prepare test team, and distribute the work to all team member.
- EX- QA team size- 6

Work	Resources
Automation	Employee1, Employee2.
Manual	Employee1, Employee4, Employee5,.
DB	Employee4, Employee5.
API	Employee6.

Requirement Phase Testing: (QA Lead)

- In this step QA Lead is involved.
- He/She decided how much test cycles are required.
- Also they consider the time required for each cycle.



Test Case Design: (QA)

- In this stage QAs are involved, and they prepared Test Cases, and findout scenarios according to requirements.
- While designing test case we consider **positive and negative scenarios**
- These Test Cases will be executed while performing the testing in the further process, ex- BBT.
- It can be maintain with the help of Excel sheet or special tools like Testlink.

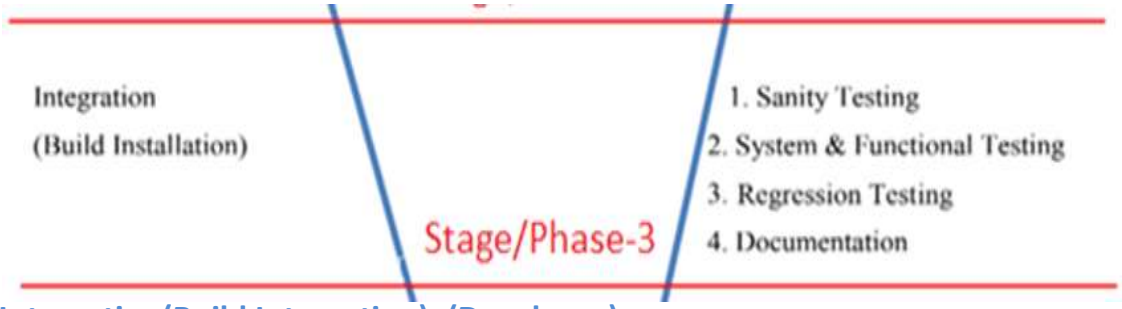
Design Phase Testing: (Developer)

- In Design phase Testing (Architect//UI **Developers**) are involved. Because they have **exact code knowledge** about it.
- This testing make sure that all the **build design is correct or not wrto the Requirements.**

Program Phase Testing: (Developer)

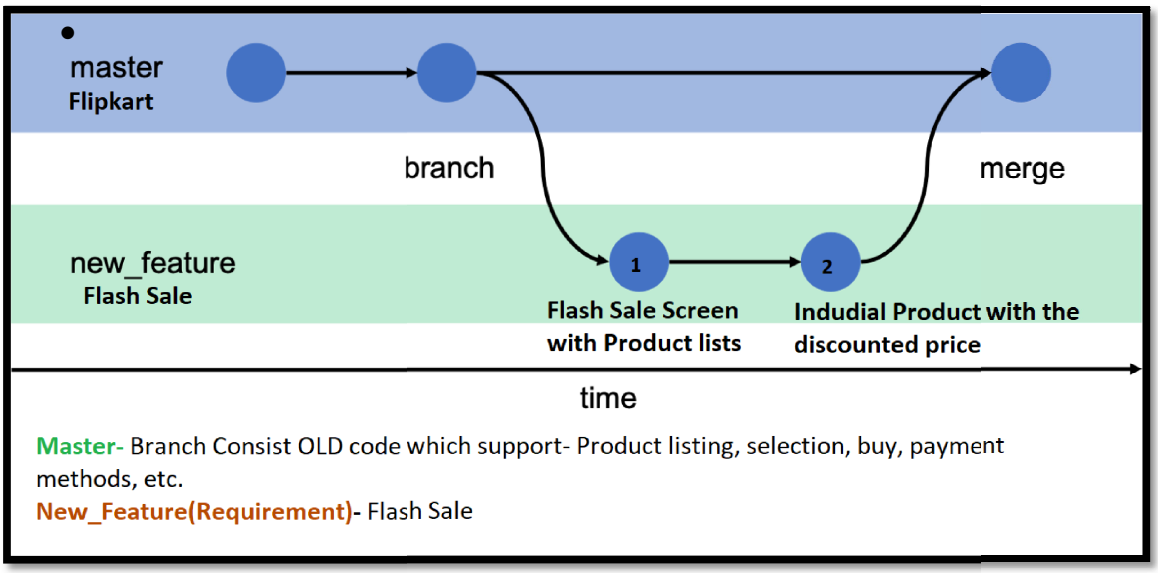
- In Program Phase Testing (FE/BE **Developers**) are involved, As they **aware about the code knowledge.**
- This tesing is also known **as Unit Level or Code Level Testing.**
- In this testing they make sure that all the **writen code is working as expected or not.**
- Developer only check **the positive scenarios** while performing the program phase testing.

Stage/Phase-3



Integration(Build Integration): (Developer)

- Build integration is done by the **developer**/developers
- They merge/integrate the new code with the existing code of the application.
- Once the WBT(Unit level testing) is done, the developer integrated the 2 module(2 units) and after sucessfully integration they performed the Integration testing on the build



Master- Branch Consist OLD code which support- Product listing, selection, buy, payment methods, etc.
New_Feature(Requirement)- Flash Sale

Sanity Testing: (QA)

- Testers/we are the responsible for the sanity testing
- In the sanity testing we check the **core functionality** of the system.
- In this testing if we found any **critical issue/Blockers** then only we communicate with the **Dev**.
- Output of the sanity testing is to ensure the **build is stable or not.**
- Without modules integration we can't perform the sanity testing.

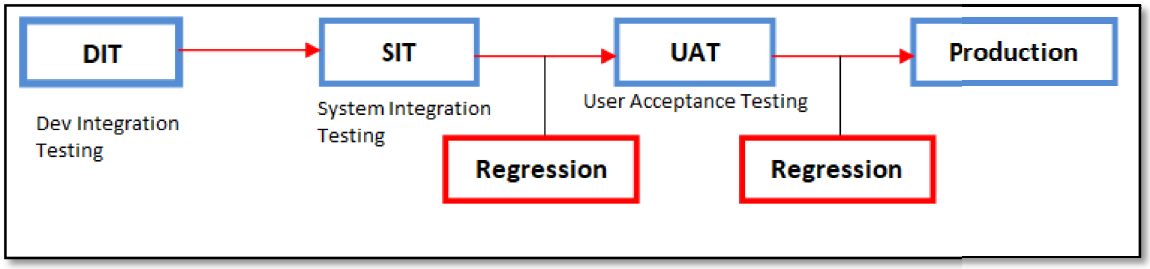
System and Functional Testing/**SIT**: (QA)

- In SIT we execute the Test Cases as per the steps writen in it.
- Here we perform Functional and Non-Functional Testing
- It is also known as the BBT.
- For each fail test case we logged the issue.
- In SIT, from small issue to the critical issue, all are been logged and communicated with the dev.

Regression Testing: (QA)

- This testing is done by the QA
- Regression testing is the **subset** of the Sanity testing.
- In this **after Integration and BBT** we ensure the core functionality is working fine or not.
- In this tesing we **check positive scenarios** and if time permits then we will check negative scenario as well.
- **Agenda** of regression testing is to check the **impact of newely added module with the existing module.**

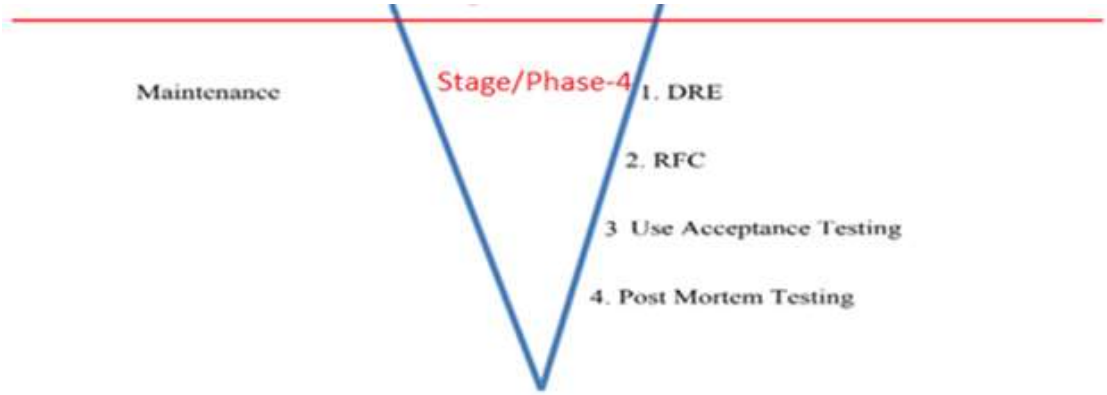
Testing Environments:



Documentation: (QA)

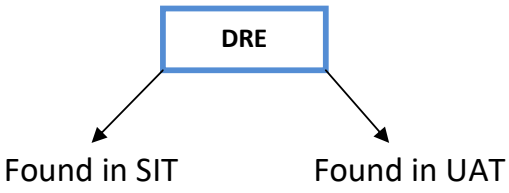
- Test documents created by the individual tester
- With the help of this document, we can easily understand that who did the testing on the particular module and what was the status of test case execution(Pass/Fail)
- After this individual documentation QA Lead create another document which is called as Test Summary Report, which contains all the test case execution records.

Stage/Phase-4



DRE: Defect Removal Effeciency-

- It is a process which which calculates at which **level/standard** tester did the testing while the BBT, Regression and UAT.
- EX- Login Functionality-
  Testing with Positive Scenario: **Vaild** UserName+**Vaild** Password.
  Negative Scenario: **Invaild** UserName+**Invaild** Password.
  Scenario Skip(Not consider while testing):
  - **Valid** UserName+**Invalid** Password.
  - **Invalid** UserName+**Valid** Password.



RFC: Request For Change-

- If any new **Change Request during** the developement process or **before Product Release** , Then we can easily accept it, and we highlight that CR point in the BRS document
- As per the requirement changes client have to pay the extra amount amout.

UAT: User Acceptance Testing-

- After removing all the open/logged defects in SIT, and product /application move to UAT enviroment we perform the User Acceptance testing.
- In UAT we check the **End to End functional flow** wrto BRS document.
- By performing the UAT **we ensure** that product/application is **defect free** and ready to for the production.

Post Mosterm Testing-

- If any unexpected issue is found before the product release at that time we sit with the dev and find the root cause and fix that issue at the same time.

Verification	Validation
1. Verification process includes checking of **documents, design, code and program**	1. Validation process includes testing and **validation of the actual product.**
2. Verification **does not** involve **code execution**	2. Validation involves **code execution.**
3. Verification uses methods like **reviews, walkthroughs, inspections**	3. Validation uses methods like black box testing, white box testing and non-functional testing.
4. Verification checks whether the **software confirms a specification/requirements**	4. Validation checks whether the **software meets the requirements and expectations.**
5. Verification process targets on software **architecture, design, database**, etc	5. Validation process targets the **actual build of the software.**
6. Verification = Static Testing	**6. Validation = Dynamic Testing**