**Use Case and USABILITY Testing:**

**Use Case:**

Use case is a pictorial representation of requirements. It explains how the end user interacts with the application. It gives all possible ways of how the end user uses the application.

Below is shown an example of how a use case looks like,

**User : USER PRIVILEGE**

**FULL ACCESS**

**ADMIN**

**PARTIAL ACCESS**

**PAID USER**

**NO ACCESS**

**FREE USER**

**FULL ACCESS :- ……….**

**……….**

**……….**

**………. 7 FEATURES**

**……….**

**……….**

**……….**

**PARTIAL ACCESS :- ……….**

**………. 4 FEATURES**

**……….**

**……….**

**Pre-condition : ………….**

**Action : ………….**

**Post-condition : ……………**

**A**

**Module**

The above figure shows a sample use case of one of the requirements in the CRS.

For the module A of the application, there are 7features.

Admin has access to all the 7 features.

For a paid user – access to 4 features

For a free user – no access to any of the features.

Ex – for admin

Precondition – admin must be created

Action – login as admin user

Post condition – 7 features must be there

Ex – for free user

Precondition – free user must be created

Action – login as free user

Post condition – no features

**Who writes use cases**

**CUSTOMER**

**Review Use Case**

**Use Case**

**CRS**

**Development Team**

**CUSTOMER**

**APPROVAL**

**TESTING TEAM DEVELOPMENT TEAM**

**Coding**

**Design**

**Test Case**

**Test Plan**

Customer gives the CRS for the application to be developed. The development team write the use case for the CRS and the use case is sent to the customer for review. If the customer approves it, then the approved use case is sent to the development team for design and coding. The approved use case is also sent to the testing team who start writing test plan and later on start writing test cases for the features of the application.

**USABILITY Testing:**

Testing the user friendliness of an application is called Usability testing

Given below are some of the parameters we look into for testing. In this most of them are not measurable,

* Speed
* Help
* Navigation should be simple
* Compatibility
* Look & feel
* Features
* Location of components

**Difference between Defect, Bug, Error and failure:**

The DEFECT/Bug is a human made mistake which intern leads to an error and an error leads to failure.

E.g.: Client explains the requirements to Business Analyst (BA) and BA misunderstands the requirements and gathers the requirements incorrectly, this is a human made mistake which has introduced a defect in the requirement document. This document is given to development team and they look at this defective requirement and develops code therefore the defect in the requirement automatically introduce an error in the code and after the entire coding is done, because there is an error in the code, we tell that there is a failure.