**SOFTWARE TESTING**

1. USER REQUIREMENTS

ABOUT:

* The **user requirement**(s) document (USD) or **user requirement**(s) specification is a document usually used in software engineering that specifies what the **user** expects the software to be able to do.
* Once the required information is completely gathered it is documented in a URD, which is meant to spell out exactly what the software must do and becomes part of the contractual agreement. A customer cannot demand features not in the URD, while the developer cannot claim the product is ready if it does not meet an item of the URD.
* The URD can be used as a guide to planning cost, timetables, milestones, etc….
* As users exhaustive needs may not be covered separately by business analysts through studying and analyzing the user requirements.

These types of requirements are generally gathered and documented using

* Use cases
* User scenario
* User stories

These requirements are documented in a user requirement document(USD) format by making use of narrative text and are usually signed off by the intended users.

**USER REQUIREMENTS EXAMPLES;**

**EXAMPLE:(WHATSAPP)**

**USER CASES:**

**User ->adds contacts from contact list on phone**

**User ->sends a message to an individual**

**User ->send a message to a group**

**User ->views message history**

USER STORIES:

* **As a new user, I want to register by creating a username and password so that the system can remember me and my data.**
* **As a registered user, I want to log in with my username and password so that the system can authenticate me and I can trust it.**
* **As a registered user, I want to be able to occasionally change my password so that I can keep it secure.**
* **As a registered user, I want to be able to request a new password so that I don’t permanently lose access to my data if I forget it.**

User story format

As a < whatsapp user >,

I want < to login the whatsapp page, to change the password, send request>,

So that < I don’t forget my password >

**ACCEPTANCE CRITERIA:**

*Acceptance criteria format*

**Correctness of implementation of the user story**

**Given;** the user navigates to the login page

**When;** the user selects <forgot password> option

**Then;** the system sends the link to the entered email

**2) Functional requirements**

**ABOUT:**

* A functional requirement defines a system or its component.
* How system should react to external environment.
* What response the system should give on certain input.
* What output should generated.
* How the system should behave in particular situations.

**TYPES:**

* Business rules
* Transaction corrections
* Adjustments & cancellation
* Administrative functions
* Authentication
* Authorization levels

**Example1:**

Some functional requirements of a university library system.

* The user should be able to search for a library item by specifying a key word.
* The library staff member should be able to issue a library item by scanning the bar codes of the library item and the student card.
* Students can reserve a library item on line.

Example2:

As an applicant, I must be able to conduct a resume on the website and edit and print it.

* Completing, editing, adding, and deleting different fields in the resume
* Downloading and printing the resume
* Displaying the resume’s completion progress

**3) Non functional requirements**

**ABOUT:**

* Non-functional requirements are the constraints or the requirements imposed on the system.
* Non functional requirements are used to judge the system as whole.
* Non functional requirements are more important than functional requirements because even we miss one or two functional requirements the system can work properly but we miss any of the non functional requirements then the system becomes completely useless.
* While developing software we have to more focus on non functional requirements.

**Types:**

* **Performance & scalability**
* **Portability & compatibility**
* **Reliability**
* **Security**
* **Localization**
* **Usability**

**Example1:**

Some non functional requirements on university library system.

* Security: Privacy of users their identity, information, all should be protected.
* Usability: all the users, applicants, can effectively use the software. It should be user friendly.
* Availability: the application should be available 24/7.
* Scalability: where college students increased or there are more books thus the application should scalable enough to accommodate those changes.
* Reliability: if the problem in the system what should be done. So this type the system should reliable enough to handle those problems.