**Software Engineering**

**DBIT SEM3**

**Group work**

1. 093084-Joseph Otieno
2. 138261 Crystie Topaz
3. 138089 Odhiambo Oliver Omondi
4. 138200- Ardhie Acosta
5. 138560 Elvis Mwiringi
6. 138507 Dan Maina

**Requirements**

# Elicit and analyze requirements

**Questionnaires-** a list of questions about the project requirements. Typically, the questions are organized by feature and can force users to select from choices, rate something, or have open ended questions allowing free-form responses.

**Brainstorming -** to get as many ideas as possible from group of people. Generally used to identify possible solutions to problems and clarify details of opportunities.

# Specify Requirements for the system

**Functional Requirements**

The software should allow running of server-side apps on remote clients. Clients who are not on physical premises of the organization may log in remotely while connected to the internet to access the software to interact with server-side applications, especially for version maintenance for the development team members.

The software should allow easy logging in of registered employees. Clients may enter their details into a log in verification module through which the software compares data entered to the database to provide the client access to the software system.

The software should provide a means for end-to-end encryption for information and resource sharing, with the aid of public-private key pairing processes, where the destination terminal is the only one who can decrypt messages sent to prevent DDoS and man-in-the-middle attacks.

The software should control traffic before it reaches the private cloud network by setting up IP deny or allow lists allowing only legitimate traffic to reach servers. This may be accomplished by setting up a traffic regulation module which provides information about source and destination IP addresses, and the type of traffic.

**Non-functional Requirements**

**Cost**: The software development process should be of low cost by the principle of building a minimum viable product by developing the minimum set of features for performing the required tasks. Also, by being a custom software, its complexity should be limited further reducing costs.

**Scalability**: The software should automatically increase with proportion to the growing number of employees without having to overprovision unused slots that count as wastage. The software should tolerate an increase in Cloud Space by improving the Cloud Server Space to accommodate for the growth in network terminals to ensure network traffic is regulated.

**Round-the-clock service**: The software should operate with minimal chances of failure, running 24/7. This may be accomplished by redundancy, having two separate instances of the software running, tracking network performance such as current number of active users, network moderators who supervise shared information and resources among workstations and ensuring the continuous and efficient operation of the hit or deny list.

**Accessibility:** The software should provide Cloud network access for users not on the physical premises of the company. Users may still obtain information and resources from the network using verified devices with the aid of an internet connection.

**User Friendly:** The software should provide a neat and well laid-out interface for users to interact with the features of the software. Consistent templates used as guided by a pre-defined front-end wireframe may guide the development of the interface, so as to ensure user satisfaction when utilizing the software.

# Requirements met

Admin should be able to:

1. Register new employees
2. Register new valid IP addresses
3. Remove employees from the system
4. View the network browsing session data of employees
5. Log in
6. Log out

Employees should be able to:

1. Log in
2. Log out
3. See their current public IP address
4. Change their password
5. Access server-side applications

System should:

1. Read public IP address of employees trying to gain access.
2. Check the first two bytes of the address against the employee’s registered addresses. This takes note of IP addresses in the same location by the same ISP not varying by much.
3. Grant access to employees if credentials are valid and their IP address is in the authorized range.
4. Record the time an employee accessed the server along with the IP address used.

Encryption and decryption of messages could not be properly demonstrated on the test environment of one computer with a localhost server.