**MAKERERE  UNIVERSITY**

**A SECURE LAYER FOR A DESKTOP WEATHER DATA REPOSITORY**

By

BSE 20-04

WEB APPLICATION

DEPARTMENT OF NETWORKS

SCHOOL OF COMPUTING AND INFORMATICS TECHNOLOGY

A Project Report Submitted to the School of Computing and Informatics Technology

for the Study Leading to a Project in Partial Fulfillment of the

Requirements for the Award of the Degree of Bachelor of

Science in Software Engineering of Makerere University.

Supervisor

Dr. Mary Nsabagwa

Department of Networks

School of Computing and Informatics Technology, Makerere University

[mnsabagwa@cis.ac.ug](mailto:mnsabagwa@cis.ac.ug) , +256-701-124388

NOVEMBER, 2020

**Declaration**

We, group BSE 20-04, hereby declare that the work presented is original and has never been submitted for an award to any university or institution of higher learning

|  |  |  |  |
| --- | --- | --- | --- |
| # | Names | Registration Number | Student Number |
| 1 | ASINGWIRE DALLINGTON | 16/U/127 | 216000708 |
| 2 | ISABIRYE TAIBU | 16/U/20495 | 216022278 |
| 3 | KUNYA NATHERN | 16/U/6372/EVE | 216006158 |
| 4 | KAGANZI TIMOTHY | 16/U/5134/EVE | 216007287 |

**Approval**

This project report titled secure layer for a desktop weather data repository has been submitted for examination with my approval as the supervisor of group BSE 20-04.

Dr. Mary Nsabagwa

Department of networks

School of Computing and Informatics Technology;

College of Computing and Information Sciences,

Makerere University

Signature: ................................................... Date: ............................

Supervisor

**Dedication**

We dedicate this work to the WIMEA-ICT Project as an appreciation for the great support they have rendered to us through the stages of the project, it would have not been possible without them. We also dedicate this work to our loved ones who have rendered to us unconditional support during the course of our program till this stage.

**Acknowledgements**

First, we would like to thank God the Almighty for blessing us with life, good health and the continuous financial provision throughout all the four years of this course till this stage.

We would like to thank our parents, guardians and friends for providing us with financial and moral support throughout the course of this program and project.

We also acknowledge with great appreciation our supervisor Dr. Mary Nsabagwa who guided us throughout the conception, realization, design and completion of this project. She has been very supportive and present whenever we needed her.

Finally, we thank our fellow students for always being supportive, updating us about the relevant class information and contributing to our newly learnt skills during the development of this project.

**Abstract**

Availability and the accuracy of weather information is one of the factors that affect the accuracy of weather predictions. Agriculture being the backbone of the economy, these predictions vital and influence agricultural decisions by farmers. This information is also vital for other sectors such as aviation.

The security of this information therefore is not only important but must be ensured to make sure accurate and timely predictions are made. For that reason, this project implemented a secure layer for the desktop weather data repository (WDR) to enhance the security of weather data stored in the system. The WDR a system which shall be used by the Uganda National Meteorological Authority (UNMA) to collect, manage and ensure good use of weather data from all the weather stations in the country.

The core features implemented in the project intend to protect the weather data through data encryption, limiting the access to the data and ensuring the principles of least privilege by implementing user roles, Tracking user typing patterns to ensure only authentic users of the system can get access to the data, Ensuring only authorized system access by implementing two factor authentication and limiting chances of data loss by synching the data from the local database to the online central server.

Table of contents

[1. Introduction 1](#_Toc41480806)

[1.1. Purpose 1](#_Toc41480807)

[1.2. Scope 1](#_Toc41480808)

[1.3. Document Overview 2](#_Toc41480809)

[1.4. Reference Material 2](#_Toc41480810)

[2. System Overview 3](#_Toc41480811)

[3. System Architecture 5](#_Toc41480812)

[3.1 Architectural design 5](#_Toc41480813)

[4.2 Decomposition Description 7](#_Toc41480814)

[4.2.1 Log user actions component 8](#_Toc41480815)

[4.2.2 Data encryption component 8](#_Toc41480816)

[4.2.3 Create user levels and roles. 8](#_Toc41480817)

[4. Data Design 9](#_Toc41480818)

[4.1 Data Description 9](#_Toc41480819)

[4.2 Data Dictionary 11](#_Toc41480820)

[5. Component Design 15](#_Toc41480821)

[5.1 Log user actions component 15](#_Toc41480822)

[5.2 Data encryption component 15](#_Toc41480823)

[5.3 user level data access 15](#_Toc41480824)

[5.4 Tracking the typing patterns of the user 16](#_Toc41480825)

[6. Human Interface Design 17](#_Toc41480826)

[6.1 Overview of User Interface 17](#_Toc41480827)

[6.2 Screen Images 17](#_Toc41480828)

[6.3 Screen Objects and Actions 18](#_Toc41480829)

[7. Requirements Matrix 20](#_Toc41480830)

**List of figures**

[Figure 3.1 1 client server architecture for a security layer of the WDR system. 5](#_Toc58254275)

[Figure 3.2. 1 Levels of Security 7](#_Toc58254279)

[Table 1 Description of the different levels of security. 7](#_Toc58254280)

[Table 2 List of entities and attributes 9](#_Toc58254281)

[Figure 4.1. 1 Entity Relation Diagram 11](#_Toc58254282)

[Table 3: User Entity for inserting a new system user 11](#_Toc58254283)

[Table 4: Role Entity for the system roles 12](#_Toc58254284)

[Table 5: User logs Activity for storing user logs 12](#_Toc58254285)

[Table 6: Observation slip activity for storing weather data 13](#_Toc58254286)

[Table 7 Typying Pattern table 14](#_Toc58254287)

[Table 8 Login Attempts table 14](#_Toc58254288)

[Figure 6.2. 1 Search user logs 17](#_Toc58254289)

[Figure 6.2. 2 User logs interface 18](#_Toc58254290)

[Figure 6.2. 3 Add New User Role Interface 18](#_Toc58254291)

[Table 9: Screen objects and their actions 18](#_Toc58254292)

[Table 10 A requirements traceability Matrix 20](#_Toc58254293)

**List of Tables**

[Table 1 Description of the different levels of security. 7](#_Toc58254298)

[Table 2 List of entities and attributes 9](#_Toc58254299)

[Table 3: User Entity for inserting a new system user 11](#_Toc58254300)

[Table 4: Role Entity for the system roles 12](#_Toc58254301)

[Table 5: User logs Activity for storing user logs 12](#_Toc58254302)

[Table 6: Observation slip activity for storing weather data 13](#_Toc58254303)

[Table 7 Typying Pattern table 14](#_Toc58254304)

[Table 8 Login Attempts table 14](#_Toc58254305)

[Table 9: Screen objects and their actions 18](#_Toc58254306)

[Table 10 A requirements traceability Matrix 20](#_Toc58254307)

# Introduction

## Purpose

This Software Design Document (SDD) specifies the architecture and system design of a secure layer for Desktop-Weather Data Repository (WDR)[1] system. The WDR is a system used to collect and store weather data from both manual and automatic weather stations. The system shall enhance the security of weather data stored in the desktop version of WDR, a system which shall be used by the Uganda National Meteorological Authority (UNMA)[2]. The intended audience includes; system developers, system testers, project managers, system maintainers and UNMA staff.

## Scope

The system shall provide a secure layer for the desktop version of the WDR. The project will cover security features including logging user activities, weather data encryption, access control through user roles and levels among other security measures.

The objectives of the project are as follows.

1. To design a weather data encryption algorithm in order to secure data stored in the database.
2. To design keystroke monitoring algorithm which shall verify the user identity.
3. To implement user role-based access to the system data in order to enforce the principle of least privilege.
4. To design a user activity logging component for auditing and accountability for user actions
5. To design a two-factor authentication mechanism to make it harder for identity theft.

The system shall offer the following benefits to UNMA.

1. UNMA will use the system to track transactions performed on their data in order to establish the source of errors as a result of data input and modification.
2. The data shall be protected from malicious people since it will be encrypted and any authorized person who gains access to it will not be able to make sense out of it.
3. The system will allow only authorized access to data and so this will protect data from being accessed by untrusted users.
4. The system shall allow only authorized users to access data, which will protect data against any untrusted modifications.
5. The system will log and keep track of user activities to ensure data integrity. In case of any changes to the data, the log will keep the Who? What? When? And Why?

## Document Overview

The document has seven sections. The first section is an introduction, which contains the purpose of the document, scope, reference material, definitions and acronyms. The second section gives system overview, which describes the system being proposed. Section three is the system architecture which has the architectural design and decomposition description of the system. Section four is the data design, which has data description and dictionary. Section five is component design which has an overview of the components of the proposed system. Section six gives the human interface design which contains an overview of the human interfaces, screen images, screen objects and actions of the system intended to be developed. Section seven is the requirements matrix which is used to trace system requirements.

## Reference Material

[1] WIMEA-ICT, “No Title,” 2020. [Online]. Available: http://wimea.mak.ac.ug/wdr/. [Accessed: 10-Mar-2020].

[2] UNMA, “Uganda National Meteorological Authority (UNMA, 2016 and 2017). Climate data recorded in 2016 and 2017,” 2017.

# System Overview

As UNMA (Uganda National Meteorological Authority)[2] moves to a desktop version of the weather data repository, security of this data is increasingly becoming of concern. Being in a distributed environment, this data is prone to improper manipulations. This data may further be illegally distributed, hence misleading the public through wrong predictions. Because of the importance of weather data, there is need for mechanisms to secure this data from the time it is collected to the time it is used for forecasting and prediction. This set up comes with its challenges. For instance, unlimited system access, denial of service by the authorized roles, and insider attacks among others.

We are proposing a secure layer which will improve security of weather data accessed and manipulated via the desktop version of the WDR. The security of the system will be implemented in three levels.

Level 1 will provide user access control. That is, through user authentication, role-based access control and session timeouts. Keystroke dynamics which will involve the use of an Artificial Intelligent model to keep track of the user typing patterns. The Keystroke dynamics algorithm will retrieve the user’s typing speed and pattern and compare them as the user is typing to login. Through role-based access control, users will be assigned roles and through those roles, perform specific tasks on data.

Level 2, also known as data protection will provide data security through data encryption and data validation checks. Advanced Encryption Standard (AES) data encryption will be used to secure the data in forms which are not easily understood. The weather data will be encrypted as it is stored into the database and decrypted when the user is granted access after authentication. The user provides the key for the data to be decrypted.

Level 3, also known as activity tracking will be achieved through activity logging and suspicious behaviour reporting. With activity logging, the different activities that will be performed by a user will be recorded in logs. The logs shall enable all responsible persons to view a summarised list of data access and manipulation activities. The activities that will be logged include: - log on attempts both successful and unsuccessful, account changes that is account creation and deletion with the person who carried out the activity is to be logged, the various tasks performed by authenticated users, updating of data and inserting of new data.

The number of successful and failed application authentication attempts per user when logging into the system. Each activity that will be logged by the system will have username and full names of the user who carried out the activity, details of the activity and Internet Protocol address of the machine on which the activity was carried out.

# System Architecture

## Architectural design

The system will follow client-server architecture where by the client is represented by the user interface.



Figure 3.1 client server architecture for a security layer of the WDR system.

Each user, through a client WDR desktop application sends requests to its instance of local database server. The server handles those requests and sends a response back to the client. When a client is connected to internet, the data on the local database server is directly synced with the data on the remote main database server. Otherwise, the client has to wait until it gains Internet connection to sync with the remote database.

The system shall consist of the following modules: -

* User access control module:

This module is used to authenticate users so that only authorized users can access the application.

* Data protection module:

Data protection module validates, encrypts and decrypts weather data.

* Activity logging module:

This module logs every activity done by any application user to both a secured file and database.

The above modules are combined to provide a secure layer for the desktop weather data repository application.

## Decomposition Description



Figure 3.2. 1 Levels of Security

Table 1 Description of the different levels of security.

|  |  |
| --- | --- |
| Arrow Number | Description |
| 1 | After user authentication, the system will perform data validation checks on the data before it is recorded. The weather data that is recorded has to be consistent thus the need for validation checks. |
| 2 | After data validation checks, user activities are logged. This will help to track of the different actions performed by the user to help in audit tracking of the system. |
| 3 | There is encryption of the data that is to be stored in the database. This helps to provide more security of the data stored in the database, thus limiting the number of people who can access the data. |
| 4 | After authentication, there is logging of the activity of authentication that is if the authentication was successful or not, the number of times the authentication failed. |
| 5 | On authentication, if the user fails to login after five times then it is the recorded as suspicious behavior. |
| 6 | If any suspicious behavior is detected, it is then logged. |

### Log user actions component

Logging user activities involves keeping track and storing the state and outcome of all activities performed by the user. When the user performs an action on system data, a new log record is created and inserted into the database. This record includes the action performed plus the data variables changed.

### Data encryption component

Data encryption will involve translation of data into another form, or code, so that only people with access to a secret key (the decryption key) can read it. When a user enters data into a form and clicks on the submit button. The encryption component encrypts the data and inserts the encrypted data into the database.

### Create user levels and roles.

The system has different users each with rights to access given data according to their responsibility. When the user clicks add new user, the system loads a form for the user to enter data among which the user select the role of the new user being created. The system will create a user record in the database and notify the user creating the new user.

# Data Design

## Data Description

Since the weather data repository system is an information-based system, the users of the system are expected to create, manipulate, update and delete data. This section therefore gives a description of the data structures of the system.

Table 2 List of entities and attributes

|  |  |  |
| --- | --- | --- |
| Entity | Description | Attribute(s) |
| Station | Station is a place where weather data readings are taken and recorded. | * station\_id (Primary Key) * StationName * StationNumber * Location * stationCategory * StationType |
| User | User is any person who is involved in any activity related to the desktop weather data repository application. | * User\_id (Primary Key) * Station\_id (Foreign Key) * Region * FirstName * Surname * UserName * UserEmail * Role\_name (Foreign Key) * Station\_id (Foreign Key) |
| UserRole | Users in the system will have different roles for example data manager, weather analyst, weather forecaster among others. | * id (Primary Key) * role\_name * PublicKey * PrivateKey * createdAt * updatedAt |
| Observationslip | Observationslip is a form that holds weather data reading such as temperature and rainfall readings among others. | * Id (Primary Key) * Date * Time * Rainfall * Weather\_Direction * Weather\_Speed * Approved * UserID (Foreign Key) * Station\_id (Foreign Key) |
| Userlog | Userlog is any activity that is tracked and recorded by the system | * id (Primary Key) * Date * Time * Action * Details * old\_value * new\_value * field * UserID (Foreign Key) |
| TypyingPattern | This stores the typing characteristics of the user in terms of speed | * Id (Primary Key) * AverateSpeed * LastSpeedRecorded * UserId (Foreign Key) |
| LoginAttempt | This stores the failed Login attempts of the user | * Id (Primary key ) * FailedAttempts * UserId |

Described here is the database structure composed of the Station, UserRole, User, Userlog and observationSlip tables. Fields indicated PK are primary keys for each table and fields marked FK are foreign keys.

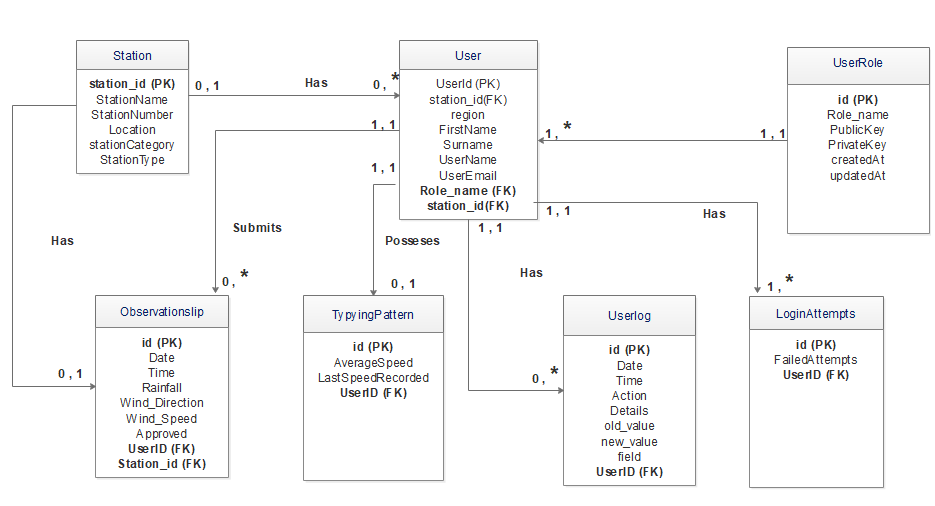


Figure 4.1. 1 Entity Relation Diagram

## 4.2 Data Dictionary

Table : User Entity for inserting a new system user

|  |  |  |  |
| --- | --- | --- | --- |
| Field Name | Data Type | Field Size for Display | Description |
| Id | integer | 11 | Unique identifier for each User |
| name | varchar | 25 | Users name |
| email | varchar | 30 | Users email address |
| password | varchar | 255 | Password for the user |
| createdAt | datetime |  | Date the user is created |
| updatedAt | datetime |  | Date the user is updated |

Table : Role Entity for the system roles

|  |  |  |  |
| --- | --- | --- | --- |
| Field Name | Data Type | Field Size for Display | Description |
| id | integer | 11 | Unique identifier for each user role |
| name | varchar | 25 | Users role |
| createdAt | datetime |  | Date the user role is created |
| updatedAt | datetime |  | Date the user role is updated |

Table : User logs Activity for storing user logs

|  |  |  |  |
| --- | --- | --- | --- |
| Field Name | Data Type | Field Size for Display | Description |
| date | datetime |  | Date for the user log |
| id | integer | 11 | Unique identifier for the user log |
| data\_id | integer | 20 | Unique identifier for each data item |
| userid | integer | 11 | Unique identifier for each user |
| action | varchar | 50 | Action performed by the user e.g. login |
| details | Varchar | text | Details about the action performed |
| Ip address | Varchar | 25 | Ip address for each user |
| status | Enum (‘00’,’11’,’10’,’01’) |  | User log status |

Table : Observation slip activity for storing weather data

|  |  |  |  |
| --- | --- | --- | --- |
| Field Name | Data Type | Field Size for display | Description |
| id | integer | 11 | Unique identifier for each record |
| Date | datetime |  | Submission Date for the record entered |
| Time | varchar | 11 | Time in zulu…used by meteorological stations |
| Rainfall | varchar | 225 | Amount of rainfall collected from the rain gauge |
| Wind\_Direction | varchar | 255 | The direction of wind |
| Wind\_Speed | varchar | 255 | The speed of wind |
| Approved | varchar | 5 | Status of the observation slip (approved or not approved by the intended person) |
| TotalAmountOfAllClouds | int | 11 | Total amount of all clouds observerd at a particular time period |

Table Typying Pattern table

|  |  |  |  |
| --- | --- | --- | --- |
| Field Name | Data Type | Field Size for Display | Description |
| id | integer | 11 | Unique identifier for each record |
| Average speed | double | 6 | Average typing speed of the user |
| LastSpeedRecordered | double | 6 | Last typing speed of the user |
| UserId | integer | 11 | Unique identifier for each user |

Table Login Attempts table

|  |  |  |  |
| --- | --- | --- | --- |
| Field Name | Data Type | Field Size for Display | Description |
| id | integer | 11 | Unique identifier for each record |
| FailedAttempts | integer | 11 | Total number of failed login attempts |
| UserId | integer | 11 | Unique identifier for each user |

# Component Design

This section provides the description of the components that constitute the architecture of the security layer of the weather data repository system. Sequence diagrams are used to describe the flow of control and data for each component.

## Log user actions component

Pseudo code

*If user is logged in and performs an action on data*

*Get user’s session identity*

*Get the modified data variables*

*Insert a record of modified variables and the user into a secured file and database*

*Print “Action performed successfully”*

*Else*

*Redirect user to login interface.*

## Data encryption component

Pseudo code

*While there is a new action performed on data*

*Read plain text x*

*Apply the advanced encryption standard algorithm to x to get xi.*

*Save xi into database*

*Else*

*Redirect user to login interface*

## user level data access

Pseudo code

*If user is logged in*

*If user role is x*

*Display specific dashboard (sub-menu) for user role x*

*Else*

*Redirect user to login interface.*

## Tracking the typing patterns of the user

*If user is logged inn and performs and action on data*

*Types in a text field.*

*System monitors the typing speed of the user.*

*The system retrieves the previous typing patterns*

*The system compares the current and retrieved typing patterns.*

*If**the current pattern matches the retrieved pattern.*

*All the user to continue the operation.*

***Else***

*System raises a warning to the manager data and mark the user’s data as suspected*

***Else***

*Redirect user to login interface.*

# Human Interface Design

## Overview of User Interface

The system will be a role-based system where the overall administrator is the Data manager. Data manager will be responsible for adding new system users and their roles as shown in figure

Every action performed by the user will be recorded and stored in the database. The user logs will be generated by the Data manager when needed.

The actions performed by the user will include login, logout, adding new observation slip (new weather data record), editing an observation slip and generating reports.

When logging into the system, the typing speed and pattern of a given user will be recorded and stored into the database. This pattern will later be used to compare the speed of the user will logging into the system.

If the user pattern does not match the one stored in the database, the user will be denied access to the system. This is to avoid malicious logins and system access by un authorized parties.

## Screen Images

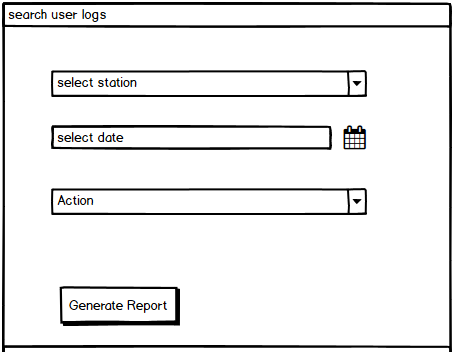


Figure 6.2. Search user logs

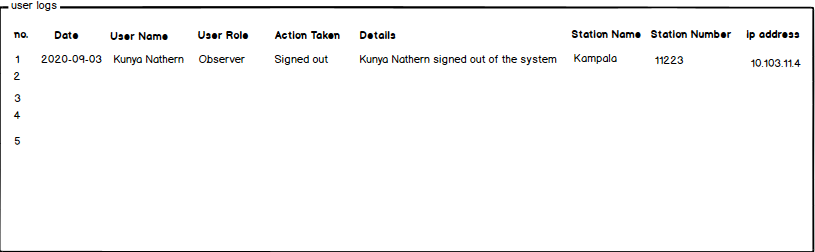


Figure 6.2. User logs interface

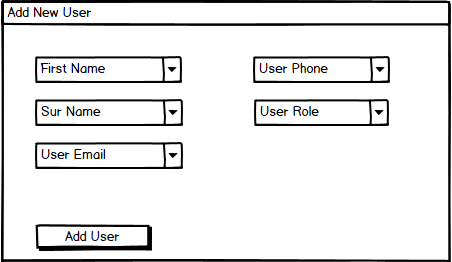


Figure 6.2. Add New User Role Interface

## Screen Objects and Actions

Table : Screen objects and their actions

|  |  |  |
| --- | --- | --- |
| **Screen Object** | **Action** | **What happens on performing the action** |
| User name | User name of the person performing the action | Display user name of the person who performed the action |
| Action Taken | action performed by the user. For example, add/ edit action, login/logout etc. | Display the action that was performed by the user |
| date | Date when the action was performed | Display the date when the action was taken |
| Station Name | Name of the station where the user is assigned. | Display the station name from which the action was performed |
| User role | The role played by the user in the system. For example, Observer. | Display the user role that performed the action |
| Details | What exactly the user did in the system. For example, added new observations lip, signed out | Display the actual action performed by the user |
| User email | Enter user email where the user logins will be sent for the user role being created | User email entered, verified  And stored into the database |
| User phone | Enter the phone number for the user being created | User phone number stored |
| Ip address | The internet protocol address of the user performing the action within the system | Display the ip address from which the action was performed |
| Station Number | The number of the station from which the user performed the action | Display the station number from which the action was performed |

# Requirements Matrix

Table A requirements traceability Matrix

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Business Use Case | Functional Requirement | | Priority | Test Case | |
| Requirement Id | Description | Id | Description |
| Logging of user activity | Req-1 | Authenticate all users before they access the system | High | TC01 | Verify if user is authenticated as they access the system |
| TC02 | Verify if user with wrong login details is not granted access |
| Req-2 | Logging of all user activities | High | TC03 | Verify if all user activities are logged |
| Encryption of data | Req-1 | Allow user enter raw data | Medium | TC04 | Verify if user provides correct input data |
| Req-2 | Encrypt data entered | High | TC05 | Verify that the system is able to encrypt data provided by user |
| Req-3 | Save encrypted data to the database | High | TC06 | Verify that the encrypted data is saved appropriately |
| Create user levels and roles | Req-1 | Allow data manager register new system users | High | TC07 | Verify if the user is able to register new users |
| Tracking of user typing patterns | Req-1 | Save user typing patterns | Medium | TC08 | Verify if the system is able to save the different user typing patterns |
| Req-2 | Retrieve and compare the typing pattern | High | TC09 | Verify if the system is able to identify a given user based on their typing pattern |
| Req-3 | Raise a warning in case of an unknown user | Medium | TC10 | Verify if the system is able to provide an alert in case of any intrusion |