Web based online attendance system using Keystroke Dynamics

(Project Report)

Project Supervisor:

Ms. Mahwish pervaiz

Submitted by

Sadaf Amanat Ali
01-134121-075
&
Shahzaib younis
01-134121-084

Department of Computer Science
BAHRIA UNIVERSITY, ISLAMABAD

CERTIFICATE

We accept the work contained in this report as a confirmation to the required standard for the partial fulfillment of the degree of BS (CS)

_____	_____
Head of Department	Supervisor
_____	_____
Internal Examiner	External Examiner

Table of Contents

List of Figures

DEDICATION

This work is dedicated to my parents and teachers. Without their support I would not have been able to complete this project.

ACKNOWLEDGEMENTS

First of all I would like to thank Al-Mighty Allah, the most Beneficial and the most Merciful. He gave me the strength to complete this major milestone of my degree program. I am also thankful to my teachers and parents who supported me both technically and morally at every stage of this project. I would also like to express my gratitude to my project supervisor Ms.Mahwish Pervaiz. Her technical guidance and support helped me complete this project and achieve its objectives.

ABSTRACT

This work presents a Web based application that authenticates the user based on his key stroke pattern. The application is capable of authenticating multiple users whom key patterns are stored in database. This application mainly relies on keystroke dynamics techniques to authenticate a legitimate user. A collection of key patterns of every legitimate user are stored in database to match the key pattern of every legitimate user. During evaluation phase the stored key patterns are matched to the users entered key pattern to authenticate the user. Finally if the key pattern matches the user is allowed to access the system or otherwise the user is not allowed to access the system.

Chapter # 1

Introduction

1.1 Overview

Researchers are keen on utilizing this keystroke dynamic data, which is typically tossed, to check or even attempt to focus the individual's character who is creating those keystrokes. This leads to a biometric recognition of the person based on his keystrokes. This authentication is based on the user speed and is strictly based on how the user types while typing.

1.2 Background

The behavior of bio metric technology regarding Keystroke Character varies from user to user as the way user use keyboard or keypad is different from other users. Every user has his own characteristic like typing speed, delay time between two keys, key pressure etc. The keystroke pattern of end user are stored in order to match the authenticated user and impostor. Natural size accessible by almost every computer keyboard might be registered to view Dwell moment (the moment an integral pressed) along with Flight moment (the moment concerning "key up" as well as the following "key down"). The registered keystroke timing information is subsequently refined by using a special sensory criteria, which in turn decides any main design intended for long term comparison.

This project is aimed at developing a Web based application to recognize the legitimate user based on his/her key patterns. This project relies on matching key patterns of the user stored in the database and then authenticating the user by applying keystroke analysis on these stored key patterns.

1.3 Problem Description

Mrs. D. Shanmugapriya classified [1] the user authentication into three categories:

- Knowledge-base (pin code or passwords)
- Object or token-base (voiceprint or gait traditional keys of the doors)
- Biometric-base

In knowledge-based and object-based methodologies, passwords and tokens can be forgotten, lost or stolen. There are also usability limitations related with them. To overcome with these problems biometric technology defined to verification and authentication of valid user based on physiological or behavioral characteristics. These categories are shown below using the figure with reference.

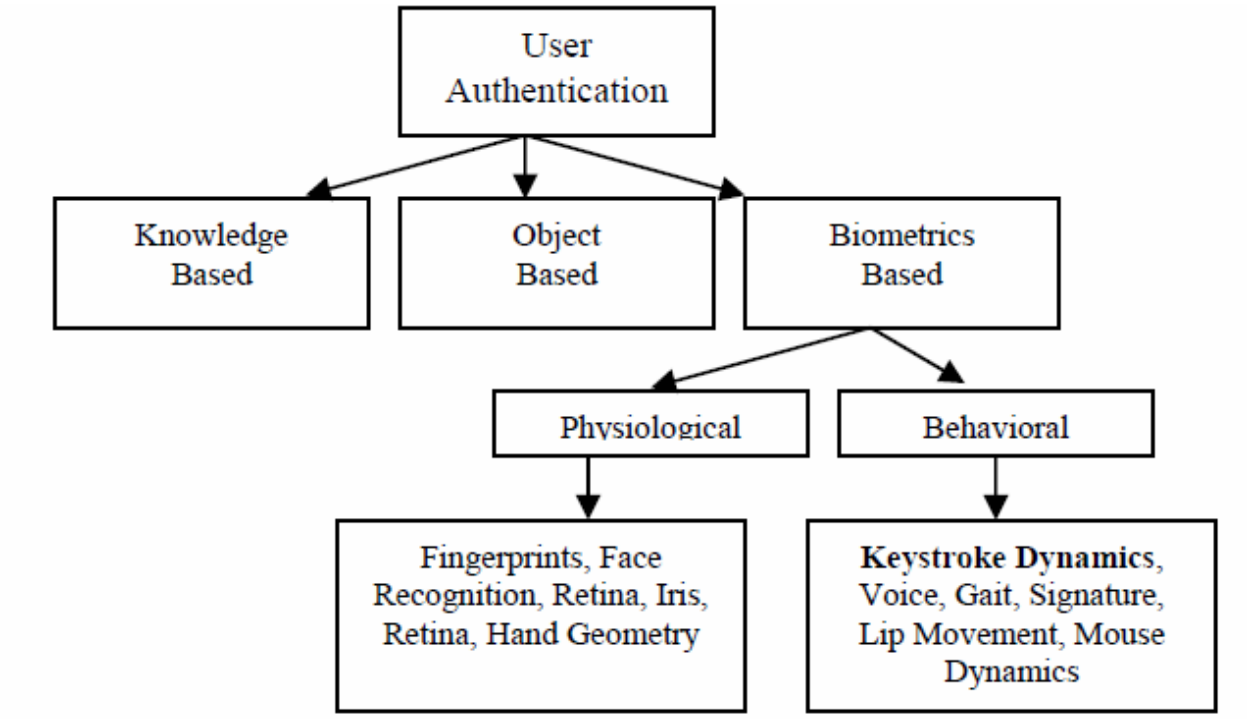


Figure 1.1 Biometrics

1.4 Keystroke Analysis Approach

There are many approaches that are used to analyze the keystroke efficiency but mainly these two are used. The below table shows the working of different researchers and what classifications they used in order to get the result. The below figure show the complete data with reference mentioned.

- Statistical approach
- Neural network approach

Neural networks

(Sajjad Haider) [2] In machine learning and psychological science, simulated neural systems are groups of models motivated by natural neural systems (the focal sensory systems of creatures, specifically the cerebrum) and it creates a set of neurons on the bases of the data collected from input and matches the results by the set that is created using the inputs.

Statistical approach

Sajjad Haider proposed in statistical approach [2] we compared the typing characteristics of the legitimate user with the typing characteristics of the test set of the same user or hacker. The results should be under a certain threshold or else the user is distinguish as a hacker.

Study	Classification Technique		Users	FAR (%)	FRR (%)
Joyce & Gupta (1990) [16]	Static	Statistical	33	0.25	16.36
Leggett et al. (1991) [18]	Dynamic	Statistical	36	12.8	11.1
Brown & Rogers (1993) [6]	Static	Neural Network	25	0	12.0
Bleha & Obaidat (1993) [27]	Static	Neural Network	24	8	9
Napier et al (1995) [23]	Dynamic	Statistical	24	3.8 (Combined)	
Obaidat & Sadoun (1997) [19]	Static	Statistical	15	0.7	1.9
		Neural Network		0	0
Monrose& Rubin (1999) [22]	Static	Statistical	63	7.9 (Combined)	
Cho et al. (2000) [7]	Static	Neural Network	21	0	1
Ord & Furnell (2000) [25]	Static	Neural Network	14	9.9	30
Bergadano et al. (2002) [5]	Static	Statistical	154	0.01	4
Guven & Sogukpinar(2003) [13]	Static	Statistical	12	1	10.7
Sogukpinar & Yalcin(2004) [28]	Static	Statistical	40	0.6	60
Dowland & Furnell (2004) [9]	Dynamic	Neural Network	35	4.9	0
Yu & Cho (2004) [10]	Static	Neural Network	21	0	3.69
Gunetti & Picardi (2005) [12]	Static	Neural Network	205	0.005	5
Clarke & Furnell (2007) [8]	Static	Neural Network	32	5 (Equal Error Rate)	
Lee and Cho (2007) [14]	Static	Neural Network	21	0.43 (Average Integrated Errors)	
Pin shen The et al (2008) [27]	Static	Statistical	50	6.36 (Equal Error Rate)	

Figure 2 1.2 Different Approaches

1.5Project Objectives

The main objectives of under taking this project include the following.

Academic Objectives

To apply the theoretical concepts acquired during the course work to authenticate the user and enhance the skill set by developing skill in the tools, technologies and concepts involved in the project.

Product Objectives

To develop an efficient web based application that will authenticate the genuine or impostor user to prevent illegal use of the system.

1.6 Scope

This system is developed for authorization of a genuine user and impostor of attendance system and will make system less vulnerable. This is done by calculating the dwell timeand flight time to increase the accuracy. Dwell time is the time that is spent while pushing a button on keyboard. Flight time is the time between the two key pressed.All the work is done in asp.net and data will be stored in backend database (SQL server).

Chapter # 2

Literature Review

This chapter presents an overview of well-known applications which authenticate the user with the help of matching the key pattern stored in database. The following sections discuss about keystroke dynamics.

2.1 Existing systems of keystroke:

Nowadays, organizations spent a lot of money to secure their data from illegal break-ins. Most of the operating systems use the simple approach to assigning the unique user name and password but it is not more efficient approach because if an impostor gets any information (user name or password) about the user then it will become a threat for the security of information resources. So to overcome these problems we use some approaches that are discussed over here.

2.2 Keystroke Approach:

- Neural networks
- Statistical approach

Statistical Approach:

(Sajjad Haider) [2] In statistical approach we compare the typing characteristics of the legitimate user with the typing characteristics of the test set of the same user or hacker. The results should be under a certain threshold or else the user is distinguish as a hacker.

Neural network:

Sajjad Haider[2] neural networks predict the outcome of the new trial on the basis of historical data.
Neural network has three layers:

- Input layer
- Hidden layer
- Output layer

There are six neurons in the input layer, four in the hidden layer and one in the output layer. The below figure shows the different layers connected with each other with reference mentioned

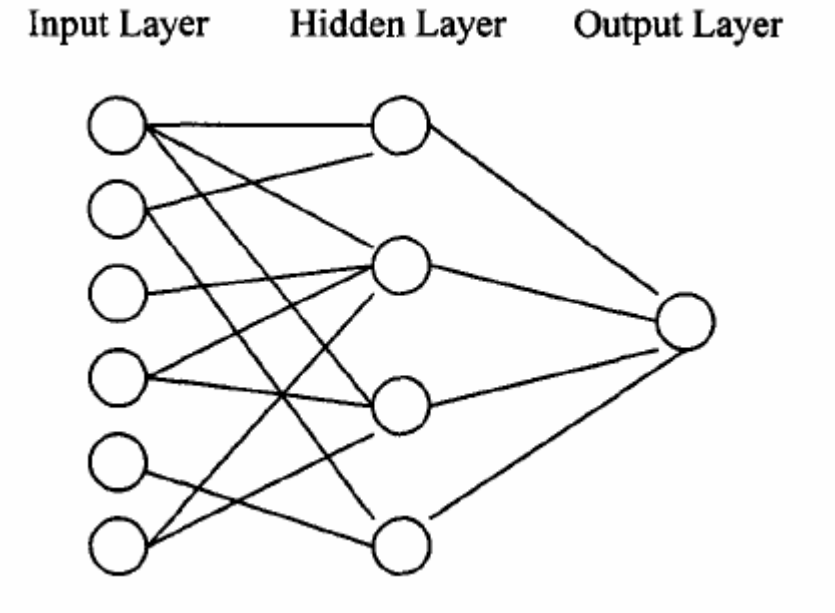


Figure 3 Neural Network Architecture in the System

2.3 KeyTrac:

KeyTrac must focus the special writing conduct of every user before it can dependably recognize user. Writing style refers to the way the user composes or types. This procedure utilizes the KeyTrac JavaScript and is totally straightforward for the user. This implies that the procedure can be incorporated in the enrollment handle that is at present set up for your application with no trouble. Recorded information does not contain any analyzable, secret data, for example, passwords or comparable data.

On the off chance that a client who is as of now in the framework needs to sign in, the user's writing conduct should first be resolved with the end goal it should be contrasted with the conduct put away in the KeyTrac framework. This procedure can likewise be coordinated in the current login structure with full straightforwardness for the client because of the JavaScript.
Feature of keyTrac.(KeyTrac, n.d.)

- Recognition by typing behavior.
- Account takeover prevention
- Access data remains secure

Chapter # 3

Requirements Specifications

3.1 Purpose of Document

Purpose of this document is to present a detailed description of web based online attendance system using keystroke dynamics. It is explaining the purpose and features of the application, the interfaces of the system, what the application will do, the constraints under which it must operate. This document is intended for both the stakeholders and the developers of the application and will be proposed to the evaluation team for its approval.

3.2 Application Overview

The intended application is used to authenticate user on keystroke patterns. This application will be designed to secure the system from unauthorized access it is making the system secure because of authentication process. The initial scope is to make the system more secure than it previously is. The main challenge is to deny the access of any impostor to the system.

3.3 System Environment

The overall system environment of the proposed application is illustrated in below figure. Application comprises two main functions first it is checking the user pattern with the stored patterns and the on these matches it is allowing or denying access to the user. The below mention diagram show the system environment.

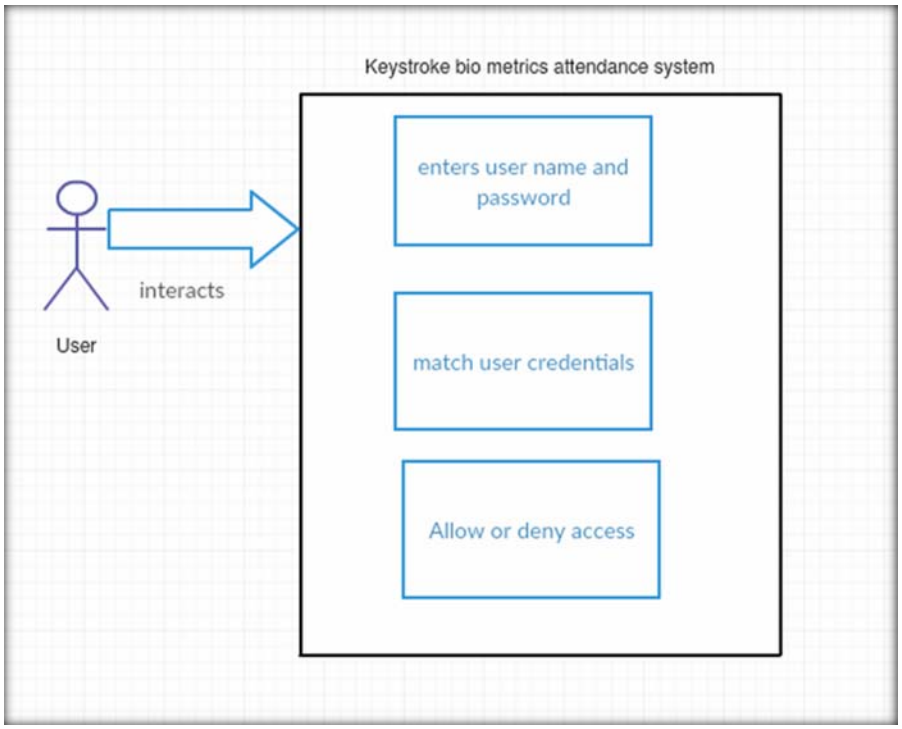


Figure 4 System Environment

3.4 Business Context

The proposed application is serving as a proof of concept which could later be extended into different other systems.

3.5 General Description

3.5.1 Product Functions

The user will interact with a graphical user interface which will allow user to enter the username and password. The system will then recognize key patterns. Once the user is identifies, the user will be allowed to access the system.

3.5.2 Similar System Information

Similar existing systems [1-3] have been discussed in detail in Chapter 2. Most of these systems work key patterns to recognize the user. In our case, we are also following the same way.

3.5.3 User Characteristics

The application does not require any special characteristics for the users. The users are expected to be familiar with the keyboard on which they are typing. The application will provide a simple and easy to use interface which would not require any specialized knowledge or expertise.

3.5.4 User Problem Statement

Authentication process includes login and password. If one have access to the password he can easily access all the details related to that account no matter what if he is the right person or not. Physical attacks can easily be done using camera recording the sequence when the password is typed .Delay time between two or more than two keys is different for different persons. One can enter his own password on relatively different speed than any other person. Same is the case with name or login etc.

We will add this factor in authentication process to make it less vulnerable and to eliminate the threat of unanimous access. The main focus will be on to achieve the maximum accuracy to achieve our goal.

3.5.5 General Constraints

Catering the large scale identification would be a challenging task. All users may not be recognized by the system (if the corresponding training data is missing). The initial version of the application will mainly focus on small number of users to check its working.

3.6. Functional Requirements

This section elaborates all the functional requirements that were gathered while keeping all the stakeholders in focus. Functional requirements tells about the system and its components. Functional requirements are supported by non-functional requirements.

3.6.1 Communication Channel

a. **Description**

The application will allow user to access the system using his/her key patterns.

b. **Technical issues**

The system should be equipped with a keyboard to allow access to the user.

c. **Cost**

Only development costs are involved, the application does not have any other cost.

d. **Risks**

Change in key patterns or change in keyboard.

3.6.2 Keyboard selection

a. **Description**

The keyboard should be of a specific brand.

b. **Criticality**

The specific keyboard is must to attain the goal.

c. **Technical issues**

The keyboard if damaged must be replaced.

d.**Cost**

The system owner has to pay the cost.

3.7 Interface Requirements

This section describes how the software interfaces with other software products or users for input or output

3.7.1 User Interfaces

This section describes how this product interfaces with the user.

3.7.1.1 GUI

The GUI is very simple, user will have to enter his credentials and the system will authenticate the user

3.7.1.2 Tools Used

The interface has been developed using .net framework.

3.7.2 Hardware Interfaces

The only hardware interface is keyboard which interacts with the application.

3.8 Performance Requirements

The keyboard should match with the minimum specification requirements that are required by the application. The keyboard type should also be compatible with the application.

3.9 Non-Functional Requirements

In addition to the attributes, other particular non-functional attributes required by the system are listed in the following.

3.9.1 Security

There is security issues involve but the application is capable of handling them.

3.9.2 Reliability

The application should be reliable should provide acceptably good results. From the view point of execution, the application should not crash in exceptional scenarios.

3.9.3 Maintainability

The maintenance of the application will be carried out by the developer, if required.

3.9.4 Portability

The application is portable as it is based on the .net platform which itself is portable.

3.9.5 Extensibility

The application is not extensible by the user but by the developer. The developer will release fresh versions of the application including enhanced functionality when required.

3.10 Operational Methodology

This project is to be implemented by using the .net framework and sql server database

3.10.1 .NET framework:

.NET framework is well known framework. The application will use .net framework to make the user interface and to interact with the database to store the key patterns of the user.

3.10.2 SQL Server Database:

Sql server database is used in the application to store the key patterns of the user so that they can be used for authentication later.

The output of recognition engine may not always be 100% correct,

3.11 Operational Scenarios

The planned version of the application would be capable of authenticating the users. The user can only be a faculty member. The user will enter his credentials. The application will authenticate the user based on his key patterns.

3.12 Preliminary Budget

The budget is only required in the development phase which will comprise

- Acquirement of keyboard.
- Acquirement of system.

Chapter # 4

System Design

This chapter documents and tracks the necessary information required to effectively define the architecture and system design. The intended audience of this chapter includes the project manager, project team and development team. Some portions of this document such as the user interface (UI) may occasionally be shared with the client/user and other stakeholders whose input/approval for the UI is needed.

4.1 Architecture Design

The application is intended to run on the any device connected to a network. Consequently, the basic interaction of the application with the hardware devices includes the following.

- Keyboard for entering the credentials.
- Database to store the key patterns.
- for executing the algorithm.

The interaction between different components within the application includes .net framework and sql server database.

4.2 Logically Significant Use-case

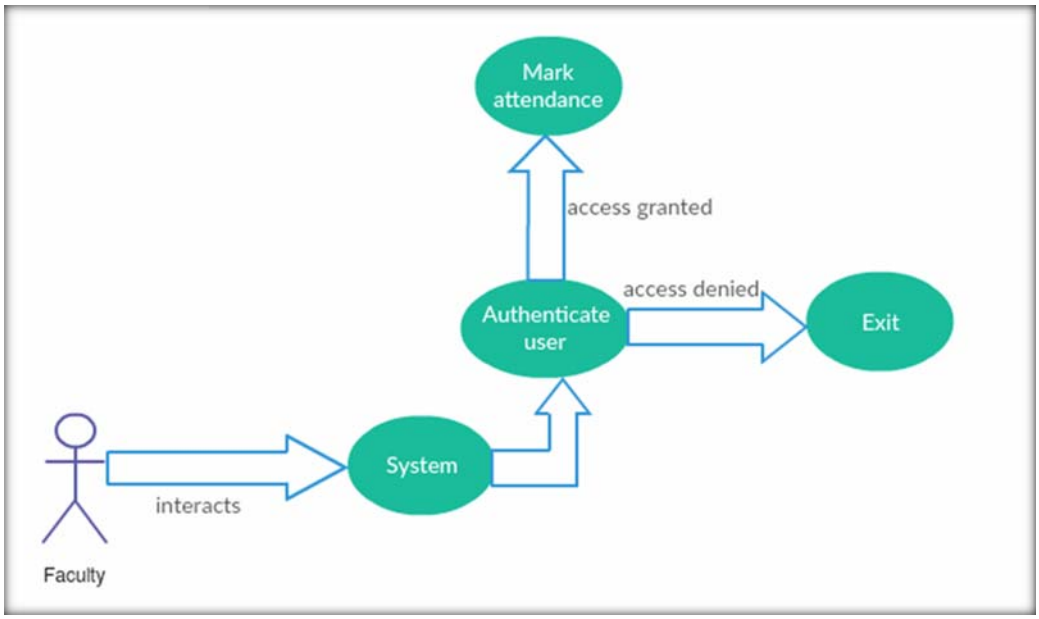


Figure 5 A typical session with the application

4.3 Use-Cases

Table: Use case Name

Use Case Id	Primary Actor	Use Cases
Uc1	User	Sign up
Uc2	User	Login
Uc3	User	Authenticate user
Uc4	User	Mark Attendance

ID:	Uc1
Title:	Sign up
Description:	The user sign ups
Primary Actor:	User
Preconditions:	User should signup
Post conditions:	This use case will tell whether the user has signed up
Main Success Scenario:	1.user open the application 2.enter username 3.enter password 4.enter other details 5.click on sign up
Extensions:	If the username or id is incorrect user will be asked to rewrite the username and

	password
Frequency of Use:	This use case has primary role in authenticating the user
Status:	Success
Owner:	Sadaf Amanat Ali
Priority:	P1-primary

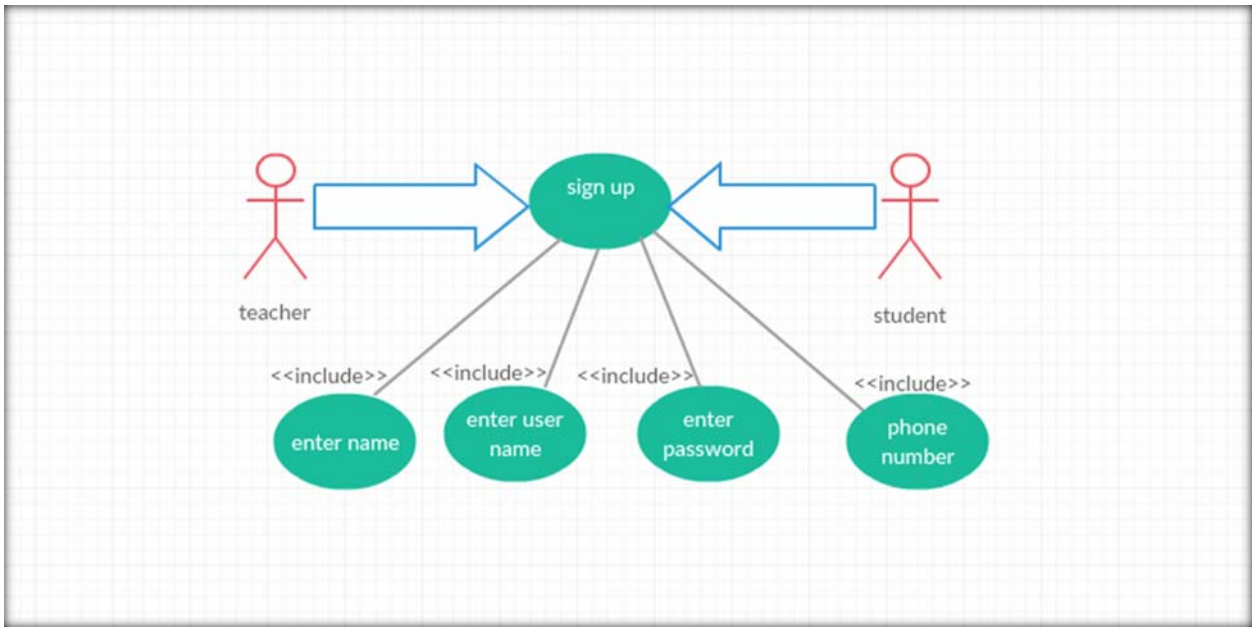


Figure 6 sign up use case

ID:	Uc2
Title:	Login
Description:	The user login to the into the system
Primary Actor:	User
Preconditions:	User should not login if impostor
Post conditions:	This use case will tell whether the application has allowed an impostor or not

Main Success Scenario:	1.user open the application 2.enter username 3.enter password 4.click on login
Extensions:	If the username or id is incorrect user will be asked to rewrite the username and password
Frequency of Use:	This use case has primary role in stopping the impostor
Status:	Success
Owner:	Shahzaib younis
Priority:	P1-primary

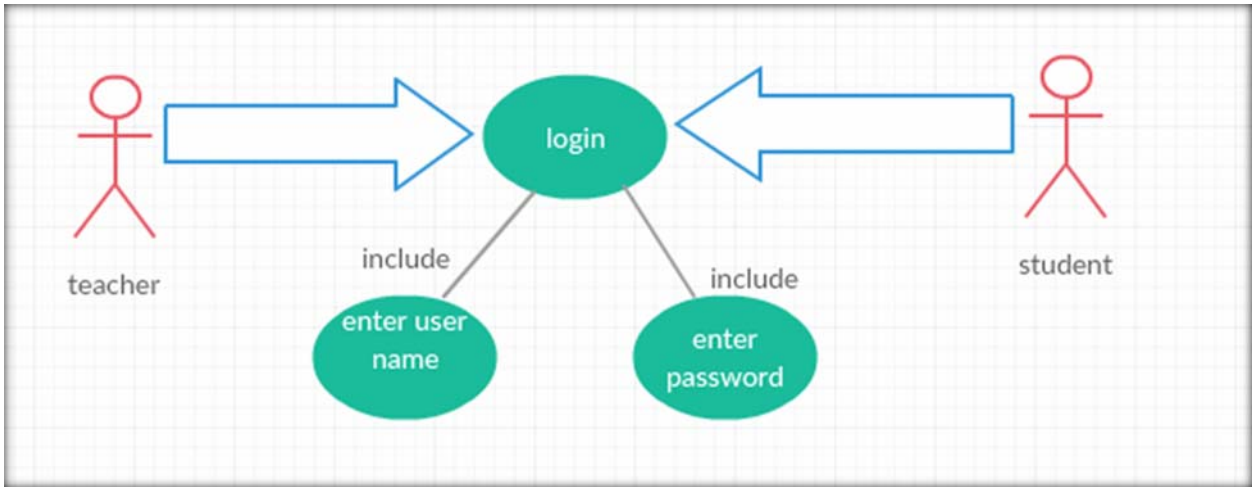


Figure 7 login use case

ID:	Uc3
Title:	Authenticate user
Description:	The application checks the authentication
Primary Actor:	User
Preconditions:	User should be authenticated
Post conditions:	This use case will tell whether the authentication is done or not.
Main Success Scenario:	1.login to the system 2.enter username and password 3.click on login
Extensions:	If there is no user authenticate will not be possible
Frequency of Use:	This use case has High level role in the application to view attendance
Status:	Success
Owner:	Shahzaib younis
Priority:	P2-medium

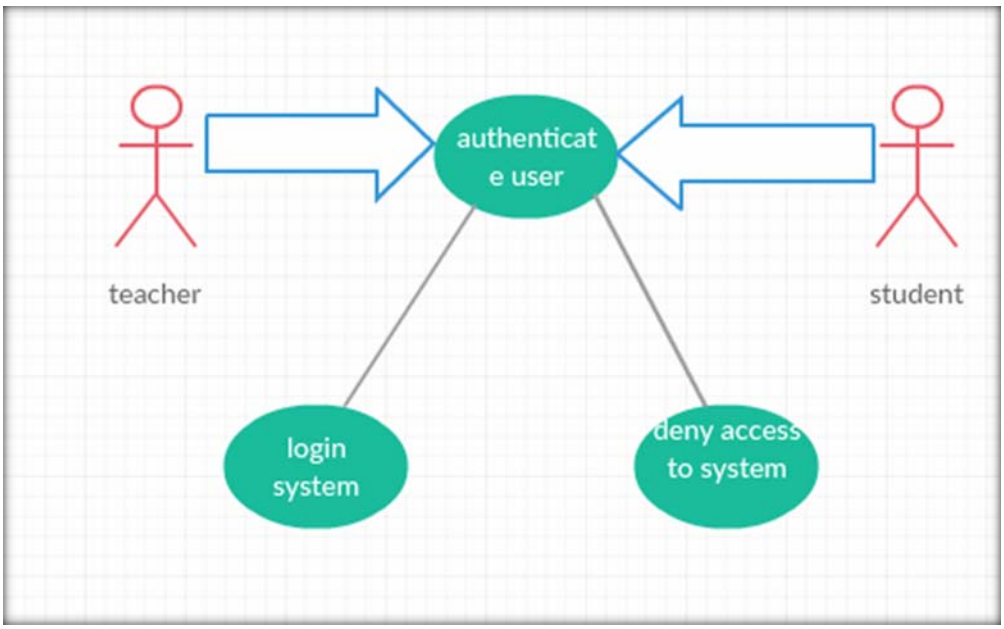


Figure 8 authenticate use case

ID:	Uc4
Title:	Mark attendance
Description:	The user should be able to mark attendance
Primary Actor:	User
Preconditions:	User should be able to mark the attendance
Post conditions:	This use case will tell whether the attendance can be marked
Main Success Scenario:	1.login to the system 2.click on attendance 3.mark the attendance
Extensions:	If there is holiday attendance will not be marked
Frequency of Use:	This use case has primary role in marking the attendance
Status:	Success
Owner:	Sadaf Amanat Ali
Priority:	P1-primary

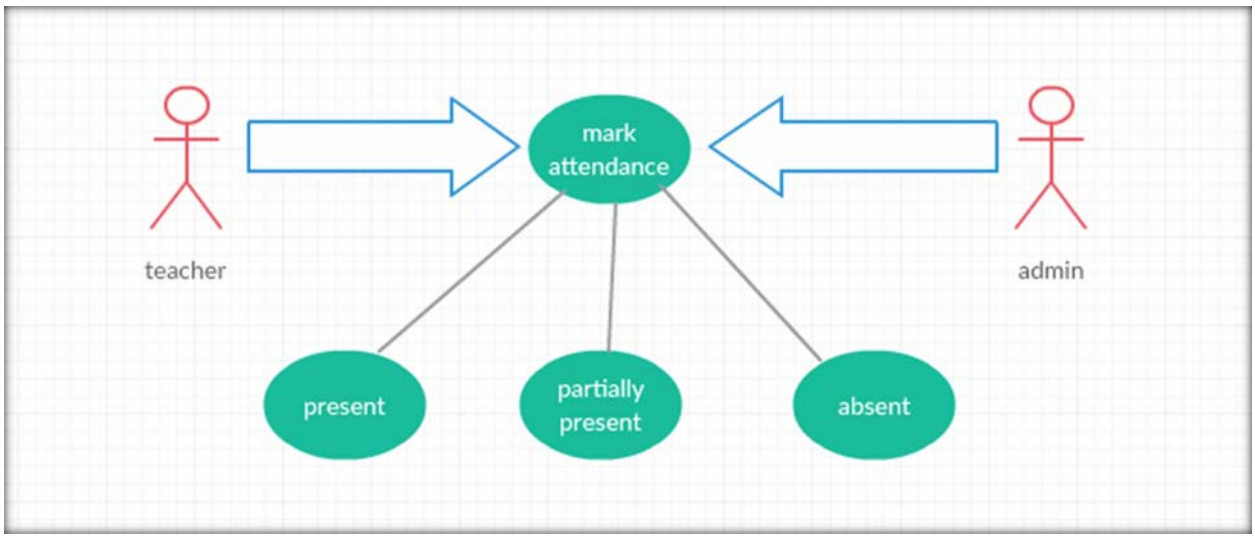


Figure 9 mark attendance use case

4.3.5 State Diagram

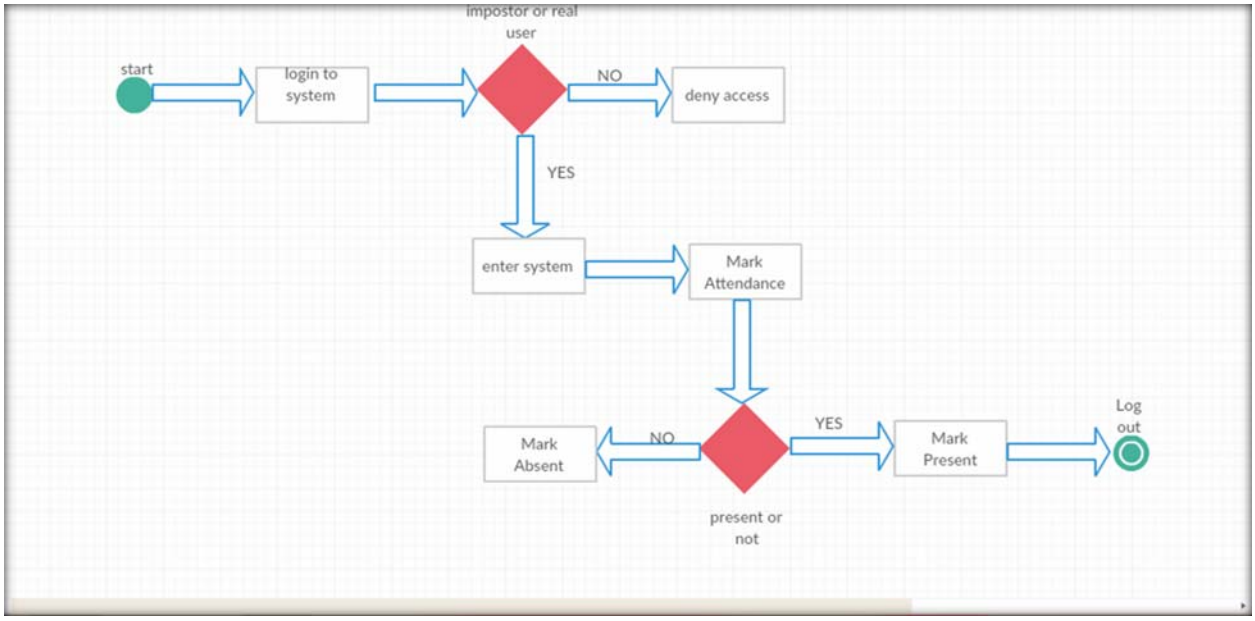


Figure 10 State Diagram

4.3.6 Data Flow Diagrams

Level 0

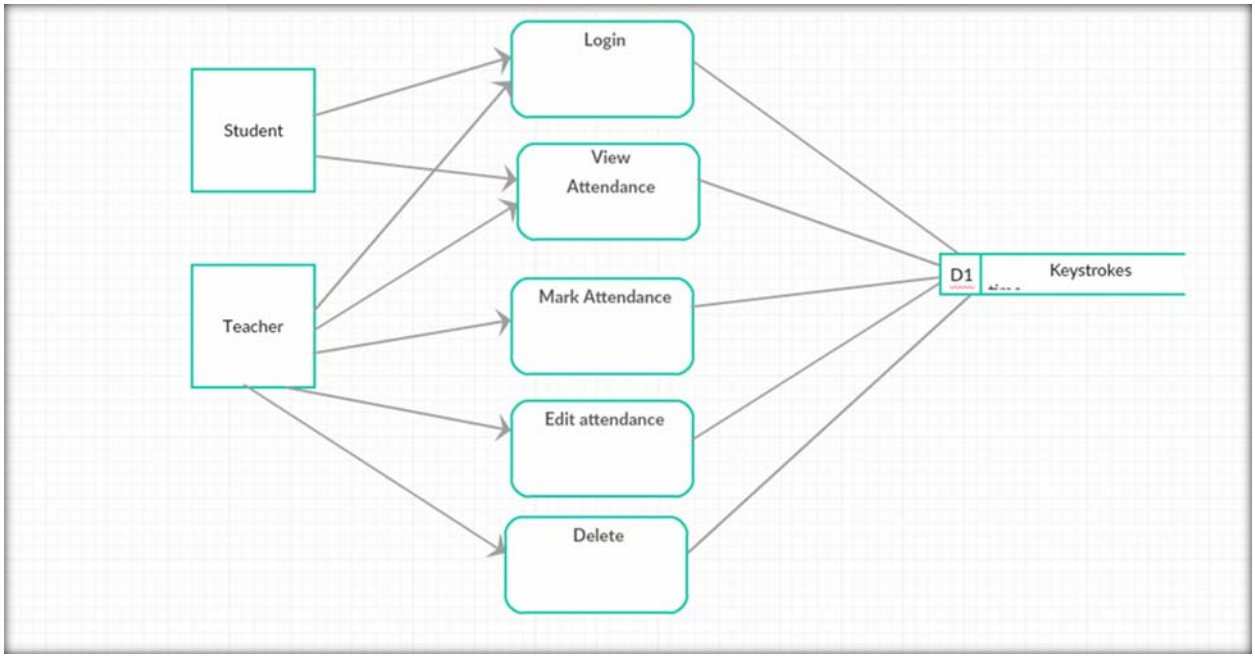


Figure 11 Data Flow Diagrams

Level 1 Login

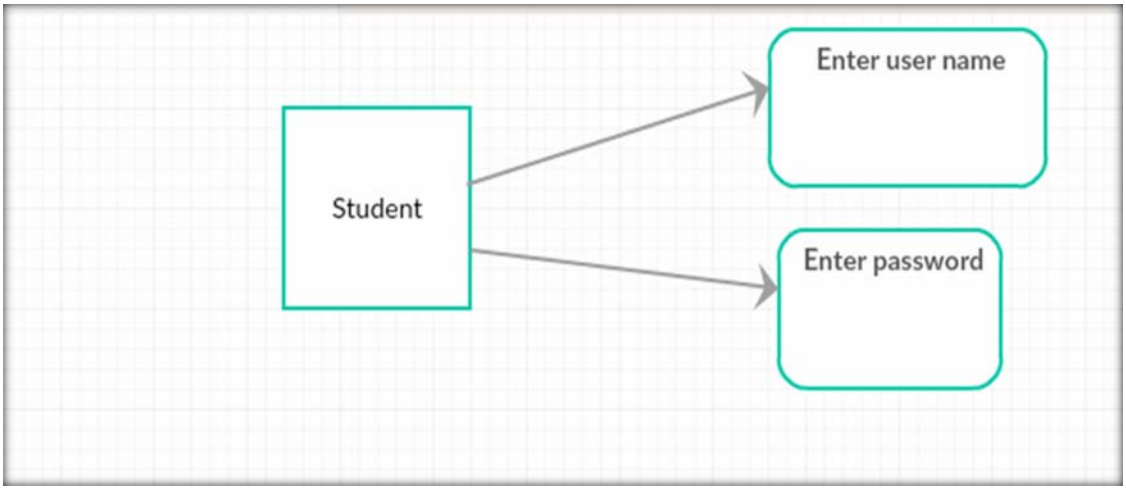


Figure 12 Data Flow Diagram for login

Level 1 Attendance

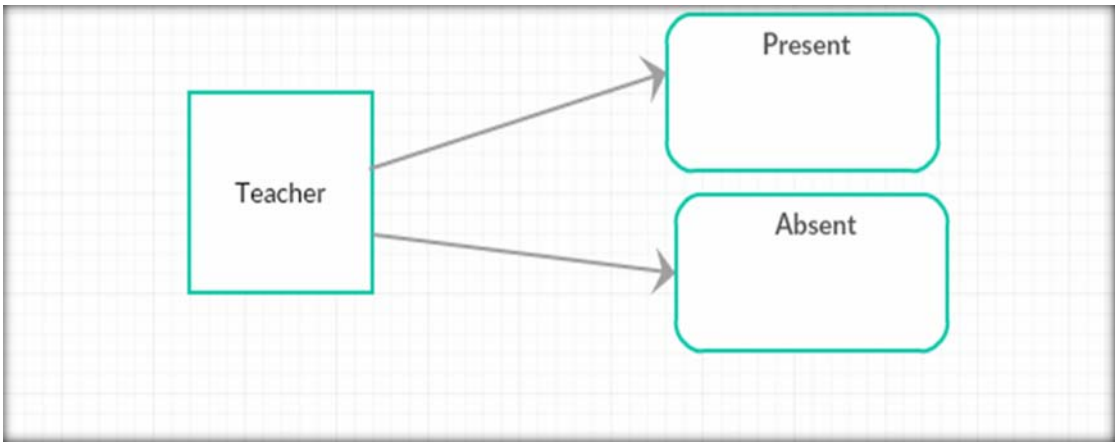


Figure 13 Data Flow Diagram for attendance

Level 1 edit attendance

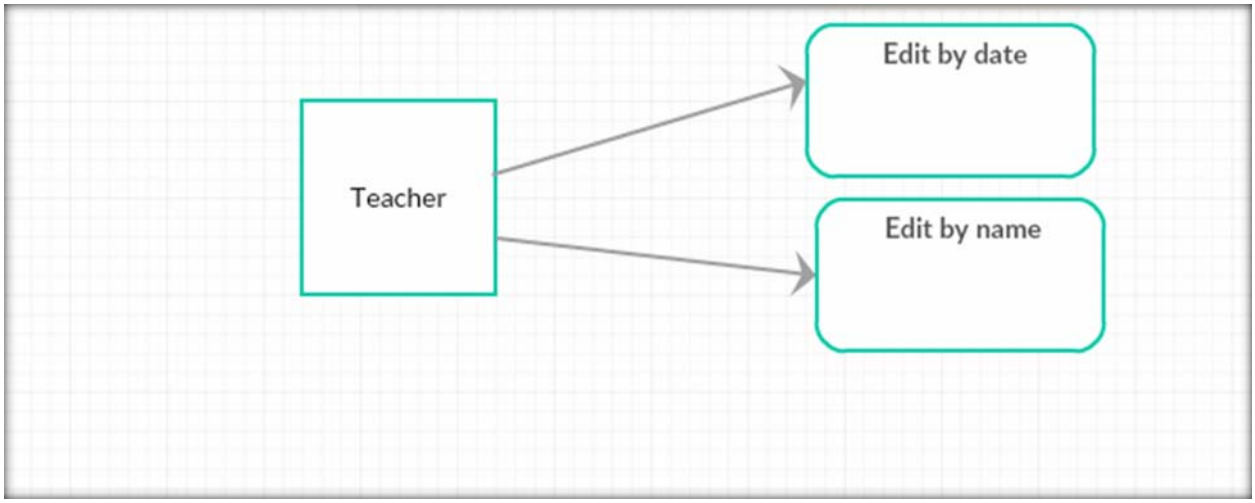


Figure 14 Data Flow Diagram for edit attendance

Level 1 Delete Attendance

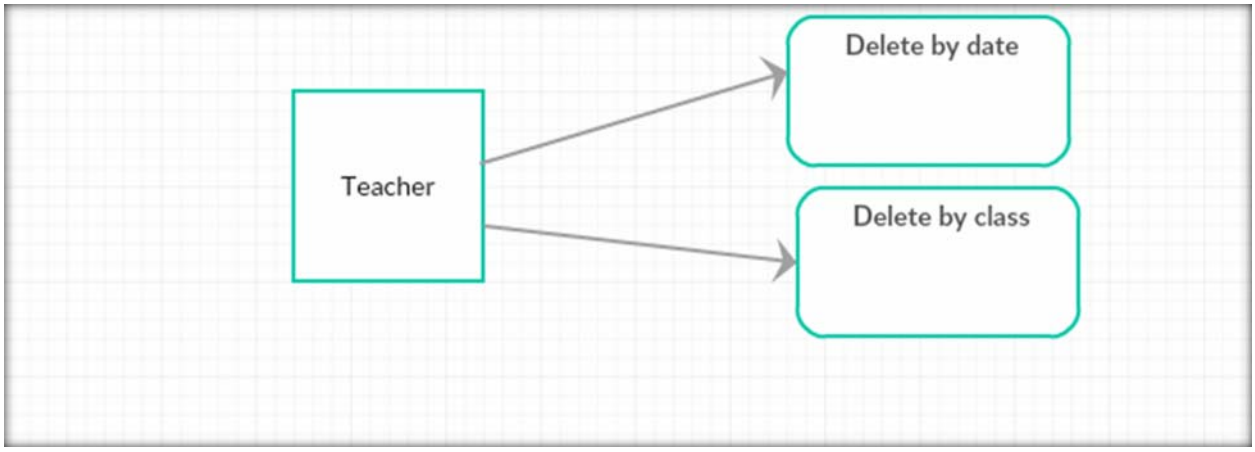


Figure 15 Data Flow Diagram for Delete attendance

4.3.7 Sequence diagram

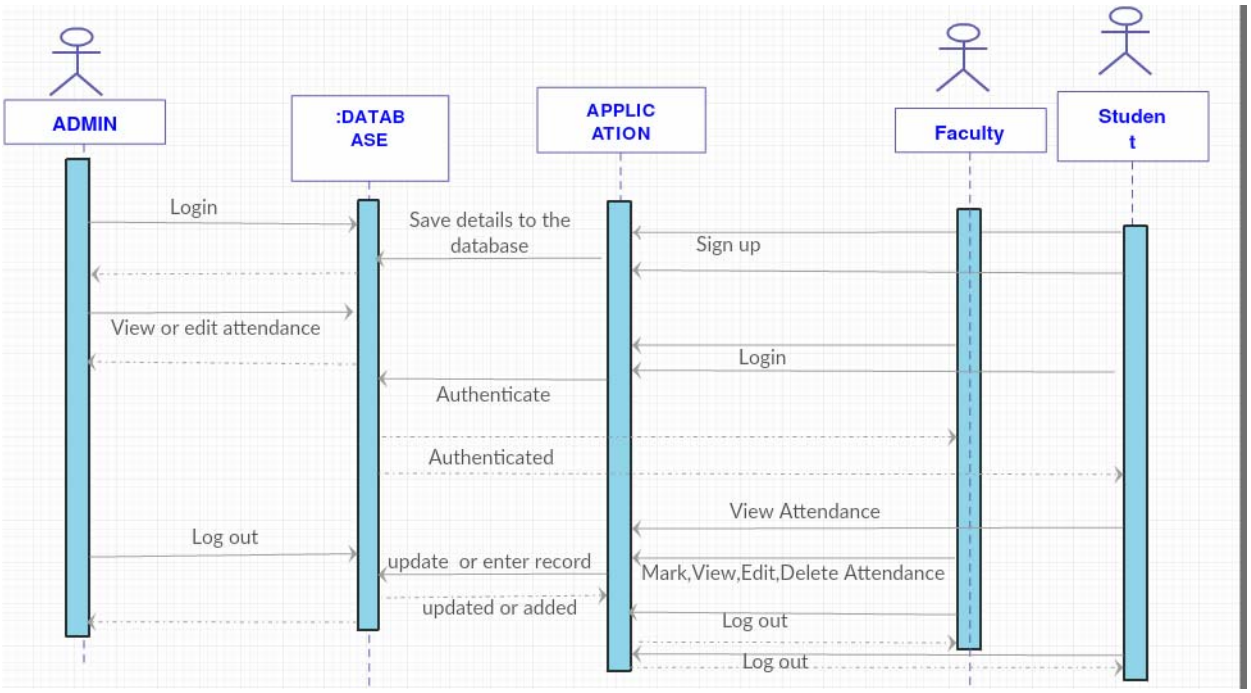


Figure 16 Sequence diagram

4.3.8 ERD

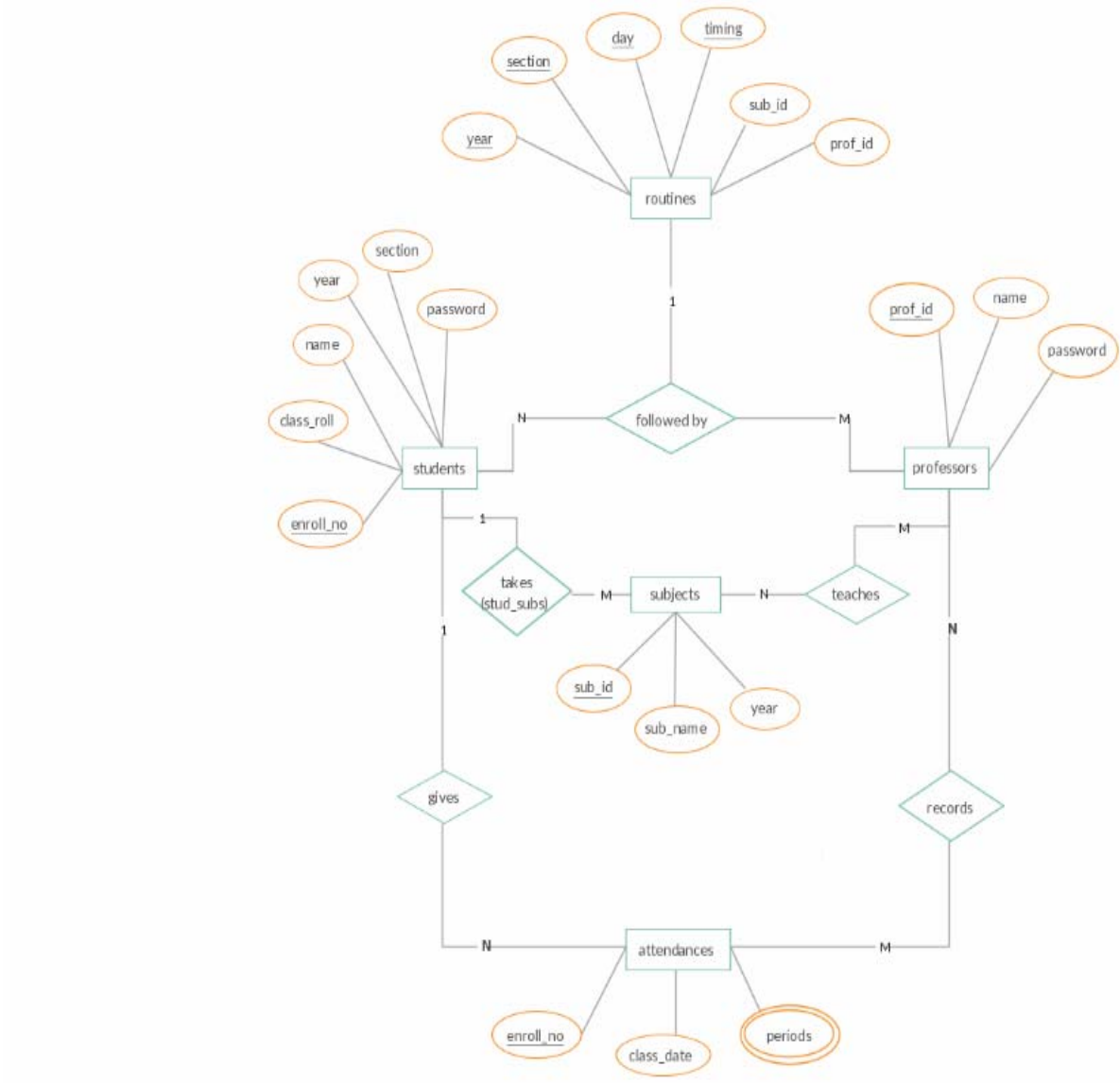


Figure 17 E-R Diagram

4.4 Application Program Interfaces

User Interface Design

The main screen of the application is illustrated in Figure 4.13.

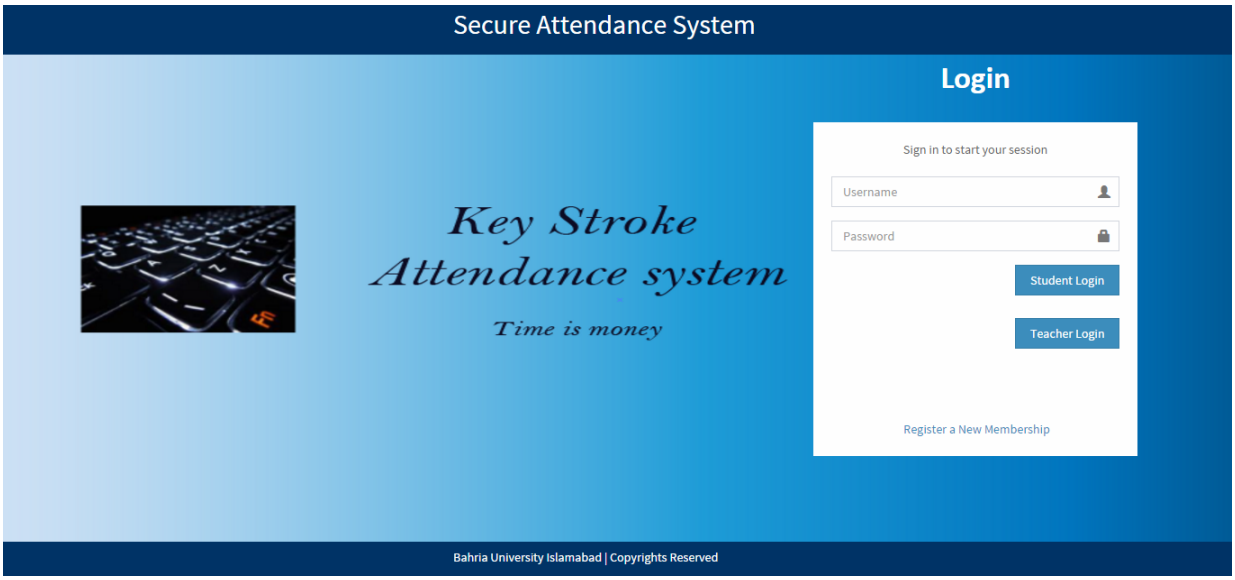


Figure 18 Main screen of application

Figure 19 register page

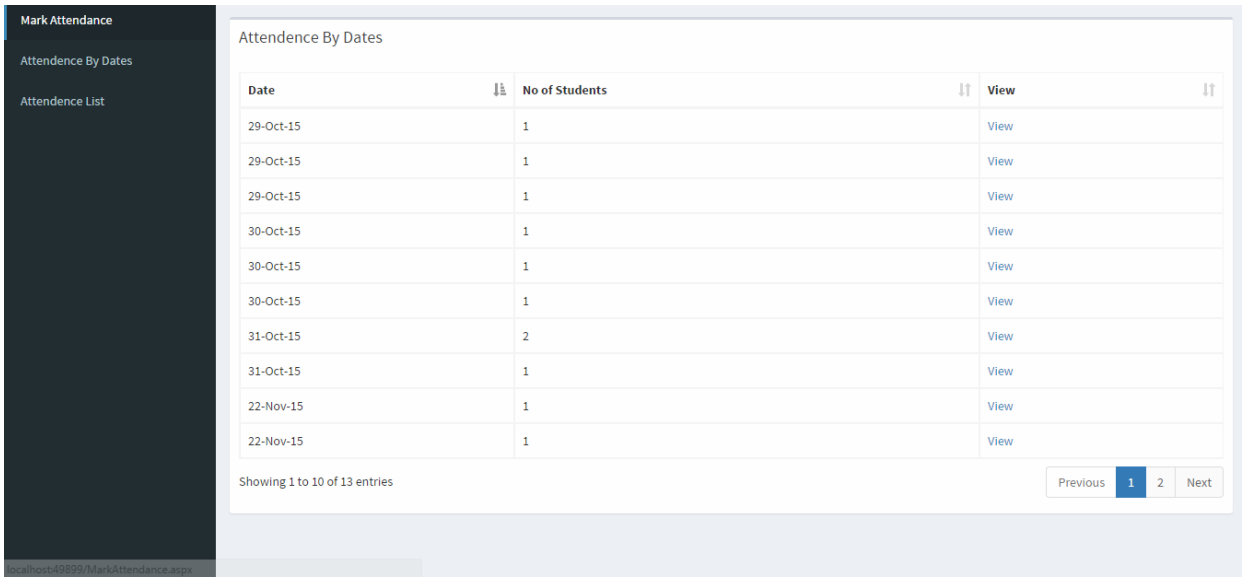


Figure 20 Attendance page

Chapter # 5

System Implementation

The chapter presents in detail the algorithmic details of the employed methodology, the tools used and the implementation details of the system.

5.1 Tools and Technology

The tools that were required for Web base online attendance system using keystroke dynamics application are as follow.

5.1.1 Visual studio

Visual studio is a commonly used IDE i.e. integrated development environment. It is open source software which is used to create different applications. Visual studio supports many programming languages including Web. Developers also have the flexibility to add required plugins in their implementations.

5.1.2 SQL Server

SQL Server is a relational database management system that is designed for enterprise systems.

5.2 Languages

The languages that are used in this project are as follow.

Java script

C sharp

HTML

SQL

JQuery

5.3 Methodology and Algorithmic Development

This section presents the algorithmic details of the different steps involved in our implementation.

The basic Idea of this algorithm is to get the time between each key down and key up while typing either a username or a password. So, at the time of user registration:

- First step is to get the time between each key down and key up for each character user types

- Second is to get the time difference between these two events, which will eventually be in milliseconds
- Third is to store the time differences in an array to make a string with comma separated values so it can be saved in the database

All the above mentioned steps are performed by our application at the time of user registration and is done using JavaScript and JQuery as we cannot afford the page to refresh as asp.net do post backs on each character type if we have set the method to be. So as the user is typing, everything is being monitored by predefined functions and they will calculate the time against each character.

As soon as the user hit save button, it saves up the complete user information along with that string on calculated times using the code behind function bound to save button. At this stage, the

Application will post back and page will refresh as we have to save the data in the Sql database using ado.net queries.

After a user is registered, the application takes the user to login page which almost works identical just there is one change. It also matches the username and password in the database to get the exact user and then matches its current time intervals with the saved ones.

As soon as the correct user is detected, its time intervals will be taken out of the string with comma separated values for time and will be matched against the time interval of currently typed username and password.

There is a range setter created as a function which can be adjusted as per the client needs. What it does is that we can set the range that will be allowed by the application to allow the time intervals not match exactly but it will allow them to be inside that range of actual value.

For example, if user had type a letter “A” at the time of registration and its time interval was 100 milliseconds. At the time of login if user has typed in letter “A” in 90 milliseconds or 110 milliseconds, then application will allow that entry to be correct as if the range setter is set to 30% and 30% of 100 means 30 millisecond so it allows the user to enter letter “A” at a speed from 100 – 30 = 70 milliseconds to 100 + 30 = 130 milliseconds but will not allow if “A” is typed at 135 milliseconds.

This range setter can be changed from 0% to any value but in our research and development of this application, we found out that from 30 – 50% it gives better results and do not allow different users to login with same credentials. The below mentioned diagram shows the workflow of program.

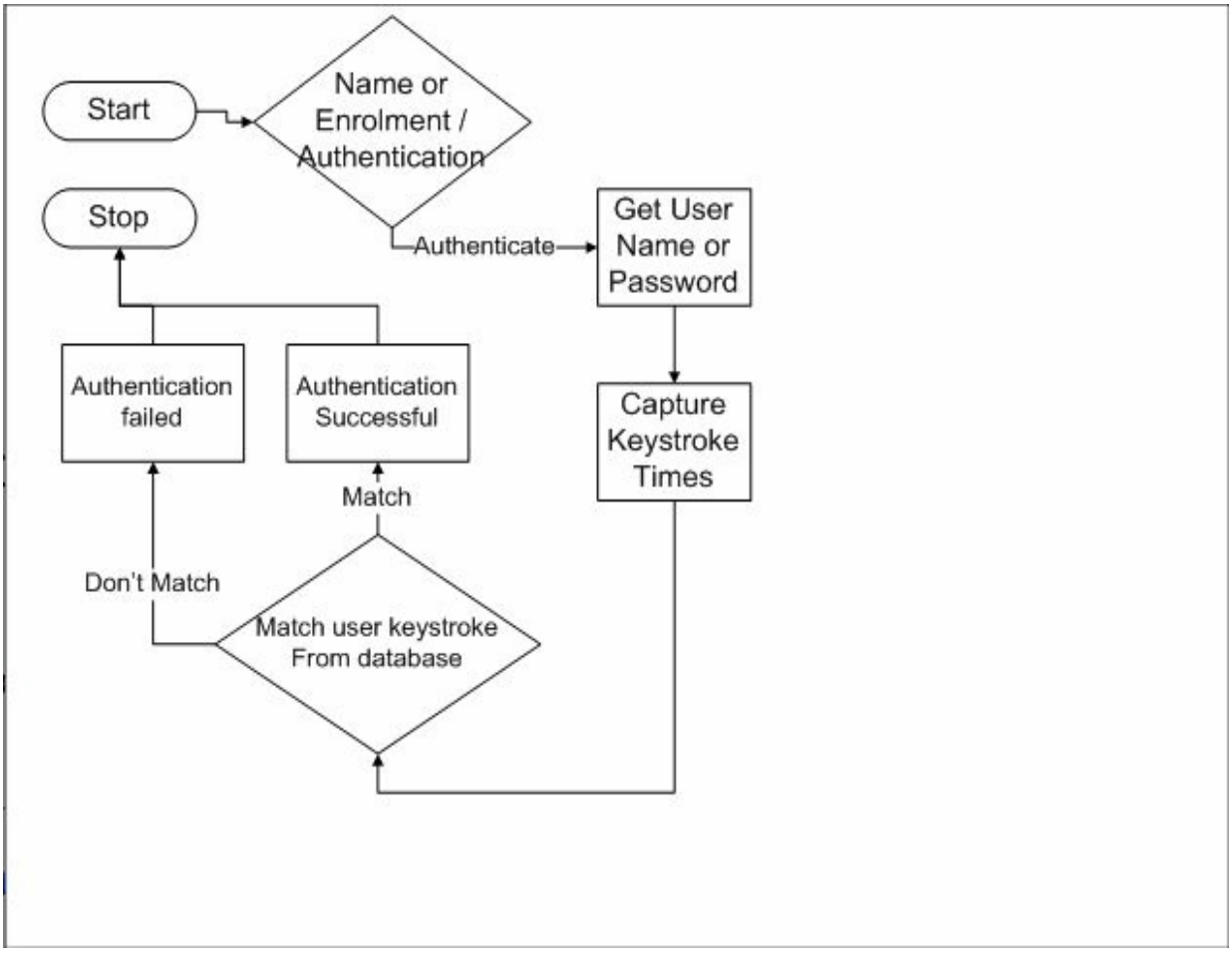


Figure 21 Working of program

Chapter # 6

System Testing

&

Evaluation

6.1 Introduction

Software testing process ensures that each module of the product is working and providing results according to the requirements. Each of the individual modules of the application is tested individually and complete application is also tested. The main objective of testing is to check whether the developed software meets the required quality standards or not. Testing is also aimed at determining whether the application is providing the desired result.

6.2 Usability Testing

Usability testing is aimed at measuring the ease with which the system can be used. The usability testing is carried out by choosing a sample of representative users and providing them the opportunity to use the application. Later, the feedback of the users can be recorded to identify the usability issues and resolve them.

6.3 Software Performance Testing

Software performance testing is the process of checking the performance, efficiency and reliability of the product. In general, the performance of the developed application is effective, reliable and efficient.

6.4 Compatibility Testing

Compatibility testing is the process in which the product is checked across different platforms which it supports. Since our application is web based it runs successfully on different browsers.

6.5 Load Testing http://en.wikipedia.org/wiki/Exception_handling

Load Testing is used to test the product under a specific expected load. As user can enter only user name and password, the application run smoothly no matter how many objects are there in the image. The application was tested by entering maximum characters of username and password.

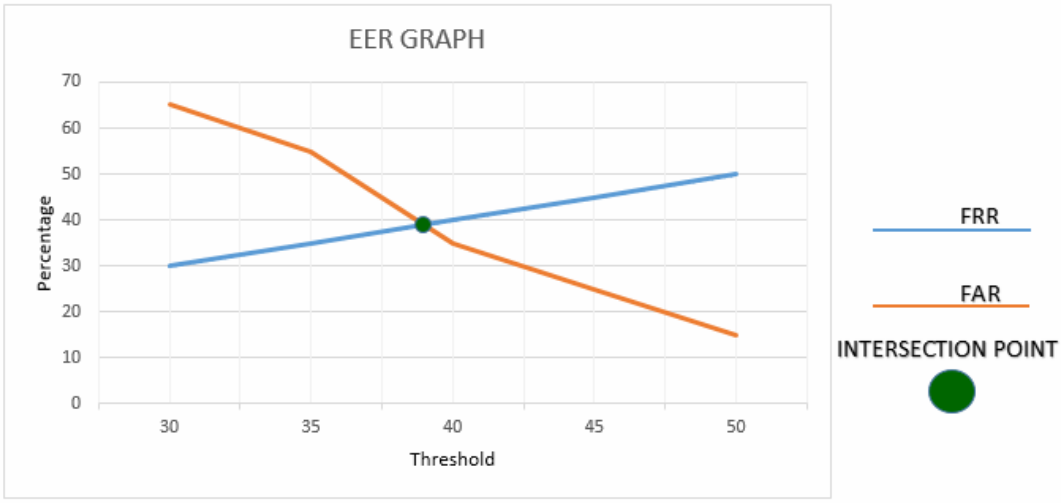
6.6 Installation Testing

Installation Testing is process of installing the application/system on the platform for which it has been developed. This application was tested on different browsers. The application debugged and executed successfully.

6.7 EER Graph

There were number of tests is done and the result of the tests are shown in the graph.

Threshold	FAR	FRR
30	9	11
35	7	13
40	4	16
45	6	14
50	3	17



6.8 Test Cases

Test Case # 1: Successful build

Test Case ID	TC#1
Unit of Test	Test to verify application is built successfully
Steps to be Executed	Open the visual studio and build the application.
Expected Result	Application should build.
Actual Result	Application build successfully.
Status	Success

Test Case # 2: browser check

Test Case ID	TC#2
Unit of Test	Test to check if the application is running successfully on the browser.
Steps to be Executed	Click on the debug button in visual studio
Expected Result	Application should run on browser successfully.
Actual Result	Application runs successfully.
Status	Success

Test Case # 3: Data storage

Test Case ID	TC#03
Unit of Test	Test to verify application is storing key patterns.
Steps to be Executed	Open database after registration and check the key patterns time.
Expected Result	Typing speed timing should be saved in database
Actual Result	Data stored successfully.
Status	Success

Test Case # 4: User recognition

Test Case ID	TC#04
Unit of Test	Test to verify application recognize the user
Steps to be Executed	Load the application and enter user name and password
Expected Result	The system must be able to recognize the user.
Actual Result	Application recognize the user.
Status	Success

Test Case # 5: Impostor recognition

Test Case ID	TC # 05
Unit of Test	Application is stopping the impostor
Steps to be Executed	Enter username and password of other user
Expected Result	Application should not allow the user access
Actual Result	Application doesn't allowed the access.
Status	Success

Test Case # 6: Attendance

Test Case ID	TC#6
Unit of Test	Attendance is marked successfully.
Steps to be Executed	After entering the system. Mark the attendance.
Expected Result	Attendance should be marked successfully.
Actual Result	Attendance marked successfully.
Status	Success

Test Case # 7: Uninstallation of application

Test Case ID	TC#7
Unit of Test	Test to verify if the application uninstalls from the System.
Steps to be Executed	Go to the programs and click on uninstall.
Expected Result	Application must be uninstalled.
Actual Result	Application is uninstalled successfully.
Status	Success

6.9Conclusion

This chapter presented the different test cases implemented to verify the working of different modules of the system. We also presented quantitative results of the evaluations carried out to assess the accuracy of the recognition system. Considering the challenges in unconstrained handwriting recognition, a correct recognition rate of N% is indeed promising. The recognition rates can be improved further by enhancing the size of training data to cater of different writing styles.

Chapter # 7

Conclusion

&

Perspectives

7.1 Conclusion

This project was aimed at developing a web application that will authenticate the real and impostor. The methodology relies on storing the key patterns of the users. These stored patterns are then matched to the user entered data after the user tries to login to the system. If the stored patterns matches the current patterns the user is allowed access to the system. After the user has validated/corrected the recognized expression, the system evaluates the patterns to compute its result.

7.2 Perspectives

The present version of the application realizes promising recognition by authenticating the real and impostor on the basis of key patterns. The recognition rates can be increased further by increasing the size of training data to cater more variations in the writing styles of different writers. Later on we can do more work on it to increase its efficiency more to get better results. ”

References

Mrs. D. Shanmugapriya, D. G. (2009). *A Survey of Biometric keystroke Dynamics:*, 1-5.

Sajjad Haider, A. A. (n.d.). *A Multi-Technique Approach for User Identification through K:eystroke Dynamics*, 1-6.

KeyTrac. (n.d.). Retrieved from KeyTrac: www.keytrac.net

”