A MINI PROJECT REPORT

ON

“PASSWORD PATTERN RECOGNITON USING KEYSTROKE DYNAMICS ”

Submitted to

SAVITRIBAI PHULE PUNE UNIVERSITY

in completion of
SKILL DEVELOPMENT LABORATORY
(T.E Computer Engineering)

BY

Saumya Purwar	Exam No : 305145
Rituja Pardhi	Exam No : 305149
Ishana Shinde.	Exam No : 305160

Department of Computer Engineering
Sinhgad College of Engineering, Pune-41
Accredited by NAAC with grade ‘A’

YEAR 2018-19

CERTIFICATE

Sinhgad Technical Education Society,
Department of Computer Engineering
Sinhgad College of Engineering, Pune-41
Accredited by NAAC with grade 'A'

"PASSWORD PATTERN RECOGNITION USING KEYSTROKE DYNAMICS"

Submitted to

SAVITRIBAI PHULE PUNE UNIVERSITY

in completion of

SKILL DEVELOPMENT LABORATORY
(T.E Computer Engineering)

BY

Saumya Purwar	Exam No : 305145
Rituja Pardhi	Exam No : 305149
Ishana Shinde.	Exam No : 305160

Prof. S. H. Sheikh
Internal Guide
Department of Computer Engineering
Engineering

Prof. M. P. Wankhade
Head of Dept.
Department of Computer

Dr. S.D. Lokhande
Principal
SCOE, Pune

ACKNOWLEDGEMENT

We are profoundly grateful to Prof. S. H. Sheikh for his expert guidance and continuous encouragement throughout to see that this project rights its target since its commencement to its completion. We would like to express deepest appreciation towards Dr. S. D. Lokhande, Principal, Sinhgad College of Engineering, Pune and Prof. M. P. Wankhade, Head of Department of Computer Engineering. At last we must express our sincere heartfelt gratitude to all the staff members of Computer Engineering Department who helped us directly or indirectly during this course of work.

Saumya Purwar, Rituja Pardhi, Ishana Shinde.

Contents

List of Figures

ABBREVIATIONS

UML	Unified Modeling Language
DFD	Data Flow Diagram

Abstract

We present a novel approach to improving the security of passwords. In our approach, the legitimate user's typing patterns (e.g., durations of keystrokes, and latencies between keystrokes) are combined with the user's password to generate a hardened password that is convincingly more secure than conventional passwords against both online and offline attackers. In addition, our scheme automatically adapts to gradual changes in a user's typing patterns while maintaining the same hardened password across multiple logins, for use in file encryption or other applications requiring a long term secret key. Using empirical data and a prototype implementation of our scheme, we give evidence that our approach is viable in practice, in terms of ease of use, improved security, and performance.

Chapter 1

INTRODUCTION

1.1 Background and Basics

Pattern Recognition is a mature but exciting and fast developing field, which under - pins developments in cognate fields such as computer vision, image processing, text and document analysis and neural networks. It is closely akin to machine learning, and also finds applications in fast emerging areas such as biometrics, bioinformatics, multimedia data analysis and most recently data science. A number of application systems require secure and reliable information to confirm and determine the identity of users.

The reason for such systems is to guarantee that the information and the system is accessed only by a legitimate user of that system. Such system applications are found in secure buildings, computing devices, ATMs and other similar environments. Due to the increasing number of security threats in these environments, public and private sectors are looking for more robust systems, such as those that utilize Biometric technology to narrow the gap of security vulnerability. Biometrics refers to the automated use. of physiological or behavioural characteristics to determine or verify identity.

1.2 Problem Definition

To build an Password Pattern Recognition System that will grant the user authentication to log in, will allow the users to check for intruders and will ensure the security of the User Account.

1.2.1 Scope Statement

The scope of the system includes developing a password pattern recognition application that can detect the user not only based on the password but also based on the typing biometrics by the typing patterns. The features of the application will empower the security of the system and make it more efficient for its users. The password pattern recognition system has two main fundamental features:

1. The Sign-Up Feature: This feature is for a new User where the user can enter his details which will be stored .

2. The Login Feature: This feature will check the username of the User and will then grant access for the login . Once the user is successfully logged in he/she can check for intruders who tried to login through their account.

3. The Check Login Feature: When the user logs in the check login will grant access to the authenticate user and will capture an image of the intruders.

Chapter 2

PROJECT PLANNING AND MANAGEMENT

2.1 Introduction

This chapter covers the project planning and management details. It also covers System Requirement specifications. SRS is considered as the base for the effort estimations and project scheduling.

2.2 System Requirement Specification

This section gives the detailed description of all types of requirements to be satisfied by the System to be developed.

2.2.1 Software Requirements

- Anaconda Navigator
- OpenCV
- Tkinter
- Python 3 or more
- Time
- Matplotlib
- Pandas
- Operating System (Windows)
- Microsoft Excel

2.2.2 Hardware Requirements

- Processor : Pentium 4 or more
- RAM : Minimum 1 GB
- HARD DISK : Minimum 30 GB
- Basic Input and Output devices : Minimum 30 GB
- Webcam

2.3 Project Process Model

To build an Password Pattern Recognition System that will grant the user authentication to log in, will allow the users to check for intruders and will ensure the security of the User Account. To develop this project, an evolutionary process model for software development is adopted. The incremental model combines elements of the linear sequential model (applied repetitively) with the iterative philosophy of prototyping. The incremental model applies linear sequences in a staggered fashion as calendar time progresses. Each linear sequence produces a deliverable increment of the software.

When an incremental model is used, the first increment is often a core product. That is, basic requirements are addressed, but many supplementary features remain undelivered. The core product is used by the customer. As a result of use and/or evaluation, a plan is developed for the next increment. The plan addresses the modification of the core product to better meet the needs of the customer and the delivery of additional features and functionality. This process is repeated following the delivery of each increment, until the complete product is produced. The incremental process model, like prototyping and other evolutionary approaches, is iterative in nature. The incremental model focuses on the delivery of an operational product with each increment. Early increments are stripped down versions of the product, but they do provide capability that serves the user and also provide a platform for evaluation by the user. Incremental development is particularly useful when staff is unavailable for a complete implementation by the business deadline that has been established for the project.

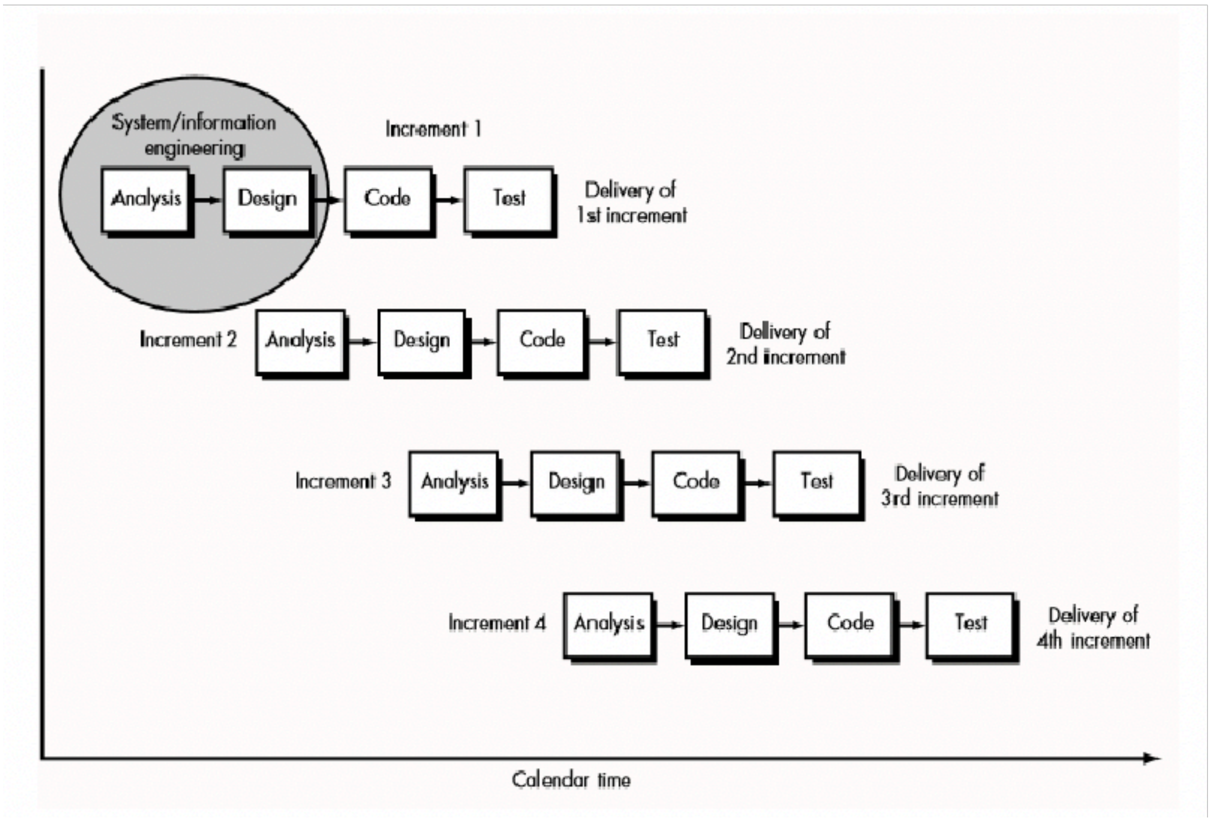


Figure 2.3: Incremental Process Model

Early increments can be implemented with fewer people. If the core product is well received, then additional staff can be added to implement the next increment. In addition, increments can be planned to manage technical risks. This project developed using the incremental model would deliver a password pattern recognition model with all the basic functionalities as a first increment. Then the linear sequences would be applied for the development of each add-on feature.

Chapter 3

ANALYSIS AND DESIGN

3.1. UML Diagrams

3.1.1. Use Case Diagram

Use case diagrams display the relationship among actors and use case. A use case diagram is a set of scenarios that describe the interaction among the user and the system.

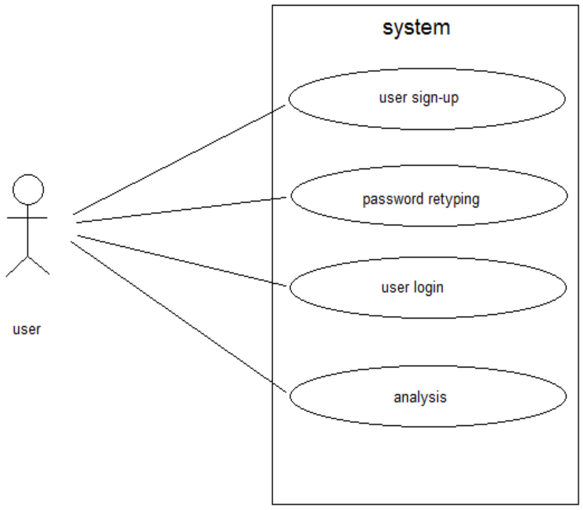


Figure 3.1.1: Use Case Diagram

3.1.2. Class Diagram

Use case diagrams display Class diagrams are backbone of the object-oriented model. Class diagram shows static design view of the system. These diagrams are built with structural things like Classes, Interfaces and collaboration relationships between them.

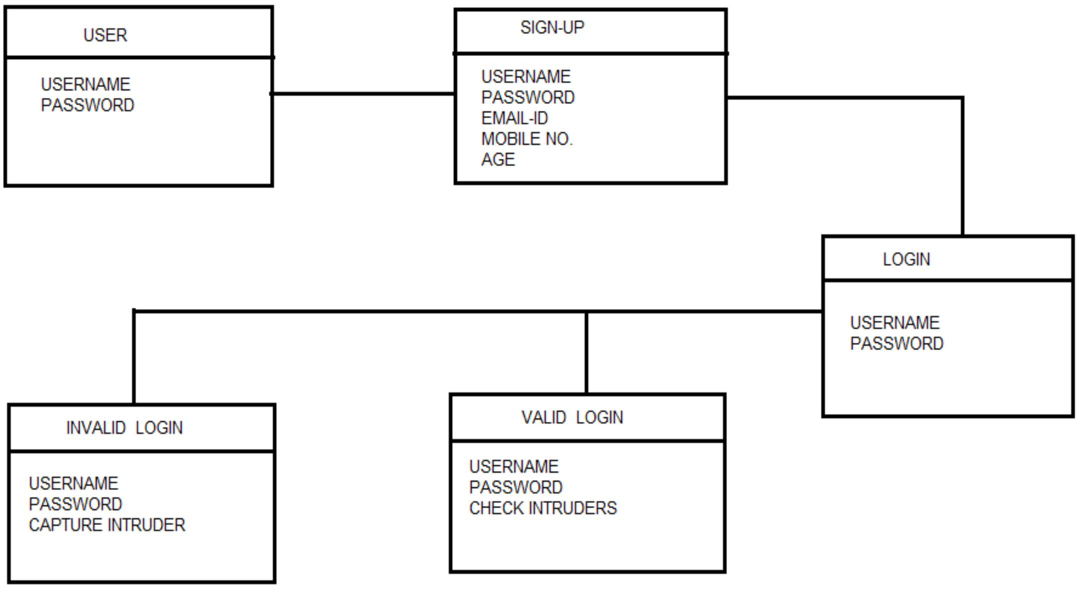


Figure 3.1.2: Class Diagram

3.1.3. Data Flow Diagram

Use case A data flow diagram (DFD) is a graphical representation of the "flow" of data through an information system, modelling its process aspects. A DFD is often used as a preliminary step to create an overview of the system without going into great detail, which can later be elaborated. DFD's can also be used for the visualization of data processing (structured design).

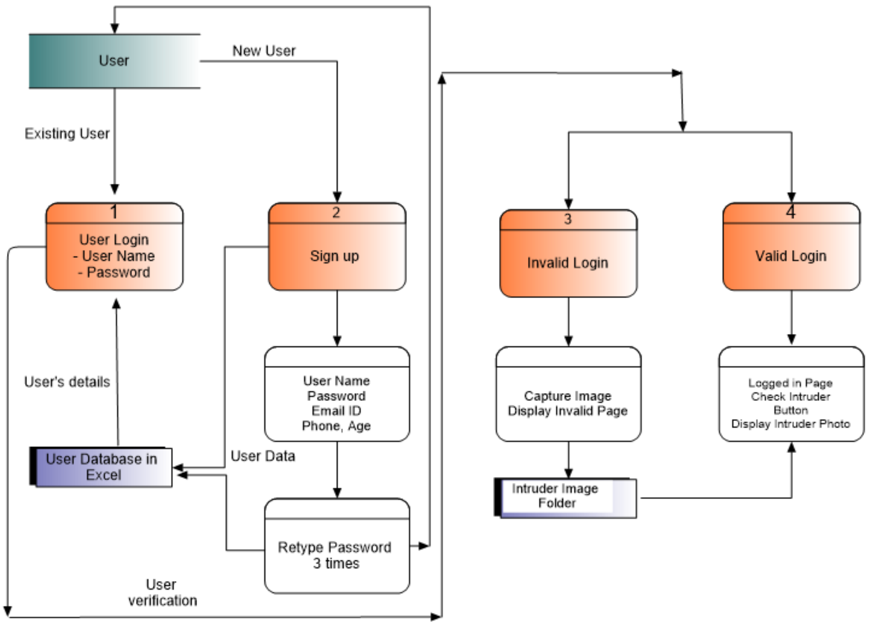


Figure 3.1.3: Data Flow Diagram

3.1.4. Deployment Diagram

A deployment diagram in the UML models the physical deployment of artifacts on nodes. To describe a web site, for example, a deployment diagram would show what hardware components ("nodes") exist (e.g., a web server, an application server, and a database server), what software components run on each node, and how the different pieces are connected.

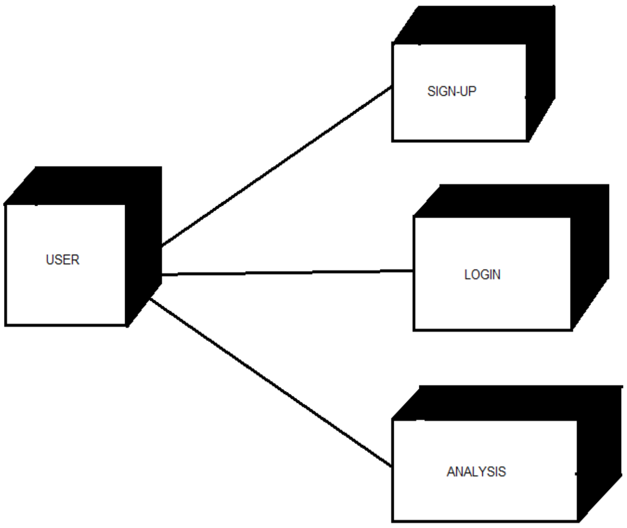


Figure 3.1.4: Deployment Diagram

Chapter 4

IMPLEMENTATION AND CODING

4.1 Methodology

Biometric is the most secure and convenient authentication tool. Biometrics measure is individual's physical or behavioural characteristics to recognize or authenticate their identity. Keystroke dynamics is a behavioural biometric based on the assumption that different people type in a unique manner. Keystroke dynamics have advantage as compare to another biometric methods that another biometric may evenly change due to slight accident and other environment factors but keystroke have based on behaviour of typing keys. The important terms used in keystroke is that there we required software and keyboard is needed for input. Using keyboard input system we check the human behaviour to type there password, what shortcuts typing methods, special keys, characters used by user. There all things are different in every human and this behaviour used to achieve a particular result in authentication. Performance of keystroke dynamics is depend on what type of keyboard is used by user.

KEYSTROKE DYNAMICS FEATURES :

A. Pressing time (the time in which the key is held down).

B. Releasing time (the time in which the key is released).

C. Latency (the time between two consecutive keys 2).

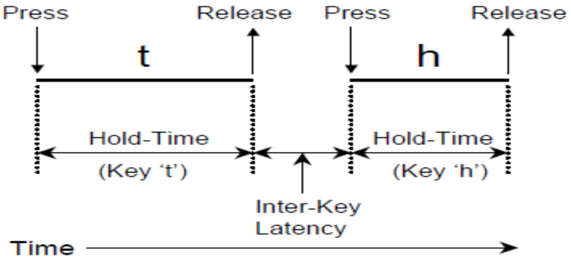


Fig 4.1 Timing Diagram

4.2 Algorithm and Flowchart

1. The home page displays three buttons for Login, Sign-up, and Analysis.
2. The new user has to first sign up.
    2.1. The user has to fill the following information such as Username, Password, Email-ID, Mobile Number and Age .
    2.2. After the details are saved, the user is asked to enter the password three times.
    2.3. The User is then directed to the Login page.
3. The Existing User can directly select the Login Option.
    3.1. Once the authenticate User has logged in he/she will be directed to the logged in page where it will ask if he wants to check for intruders.
    3.2. If an intruder tries to log in then his/her image will be captured through the web-cam and then he will be directed to the invalid login page.
    3.3. The exit button will redirect to the home page.
4. The Analysis option will display the graph of maximum average speed with respect to the age of the user.
    4.1. A new window will open and display the graph.
    4.2. The back button will redirect the User back to the home window.
5. Exit.

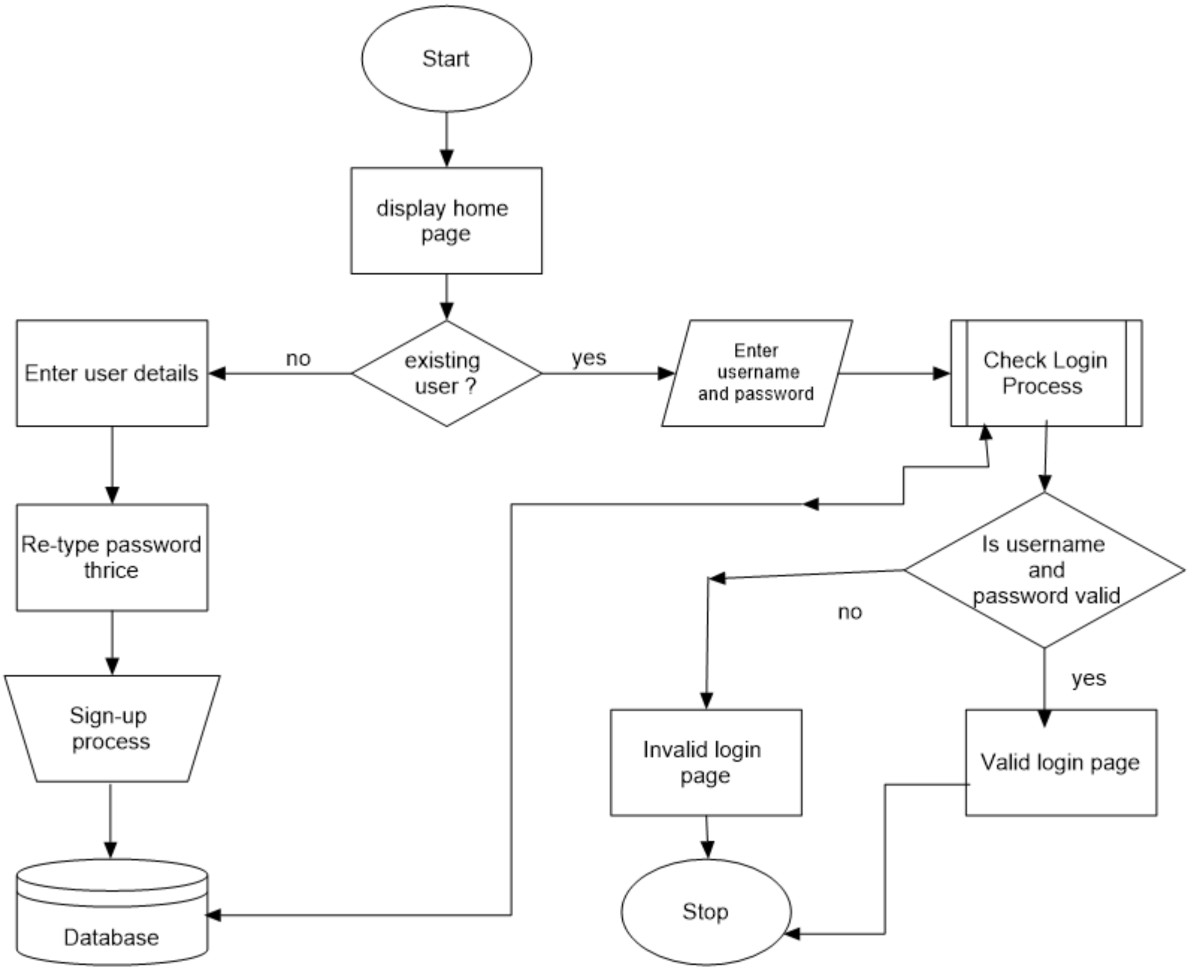


Figure 4.2: Flowchart

4.3 Screenshots

The working of important/major functionalities of the application is shown in the below figures.

Fig 4.1 shows the homepage of the system.

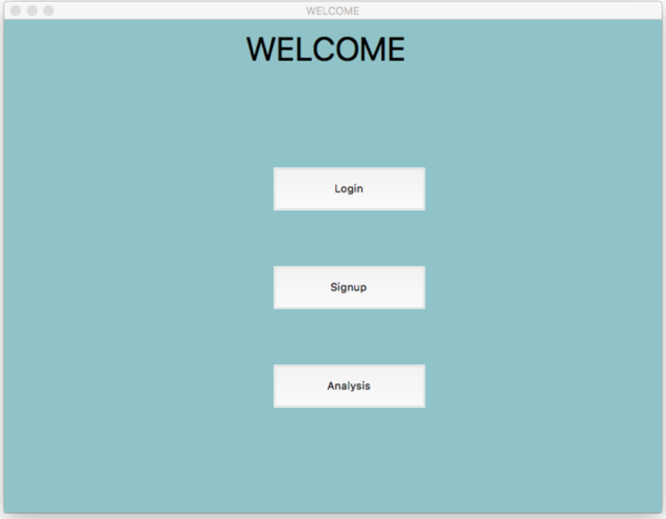
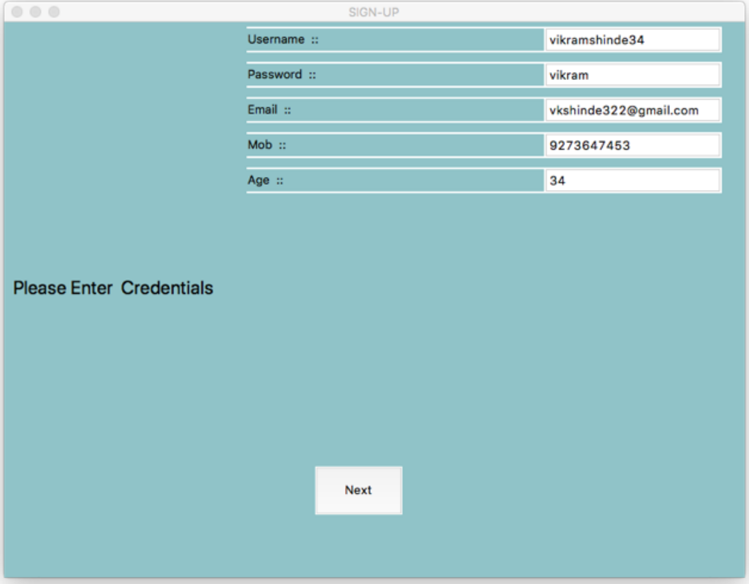


Figure 4.3.1: Home page



Username :: vikramshinde34

Password :: vikram

Email :: vkshinde322@gmail.com

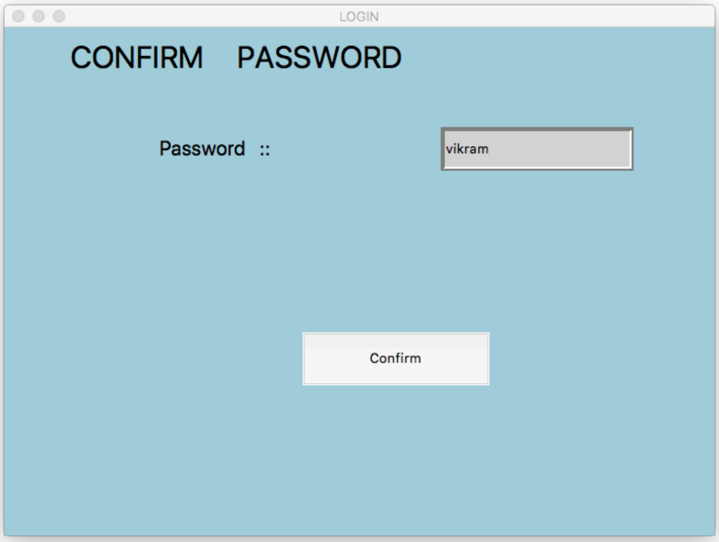
Mob :: 9273647453

Age :: 34

Please Enter Credentials

Figure 4.3.2: User Signup

Fig 4.2 shows the sign – up page of the system where the user will enter his/her details.

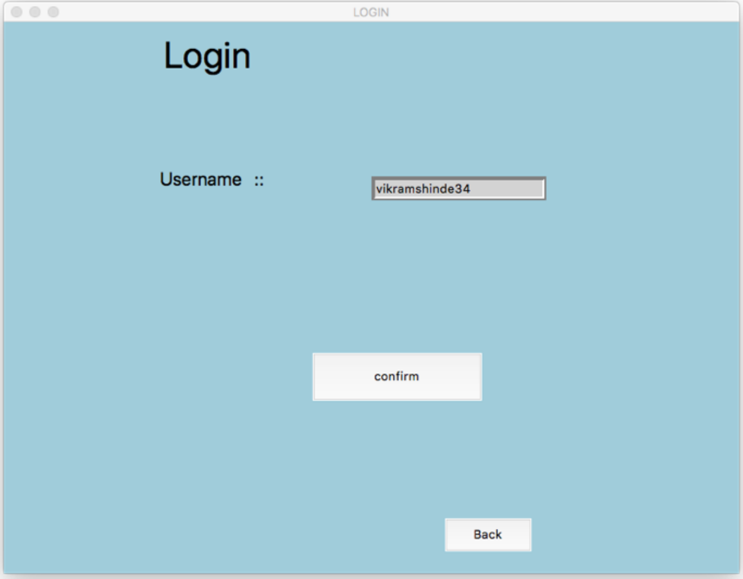


CONFIRM PASSWORD

Password :: vikram

Figure 4.3.3: Password Confirmation

Fig 4.3 shows the window where the user has to retype the password for confirmation.



Login

Username :: vikramshinde34

Figure 4.3.4: User Login Page

Fig 4.4 and Fig 4.5 shows the working screenshot of the signed up user to login.

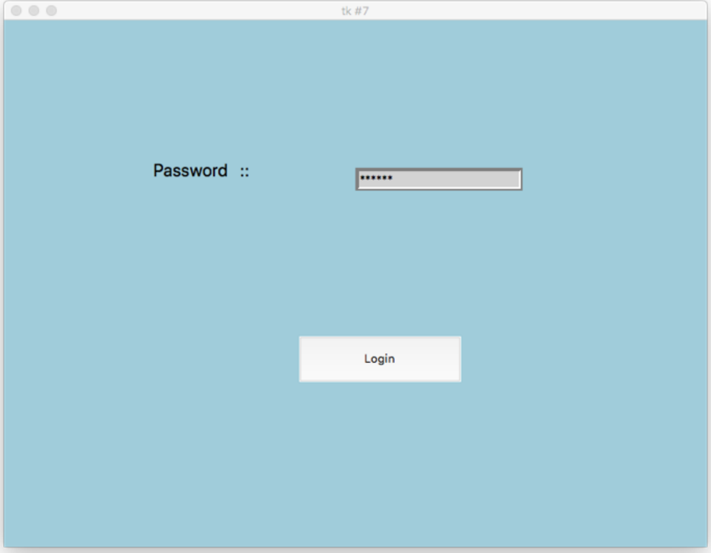


Figure 4.3.5: User Password Login Page

Fig 4.6 shows the screenshot of the page to which the User is redirected to once he/she is successfully logged in.

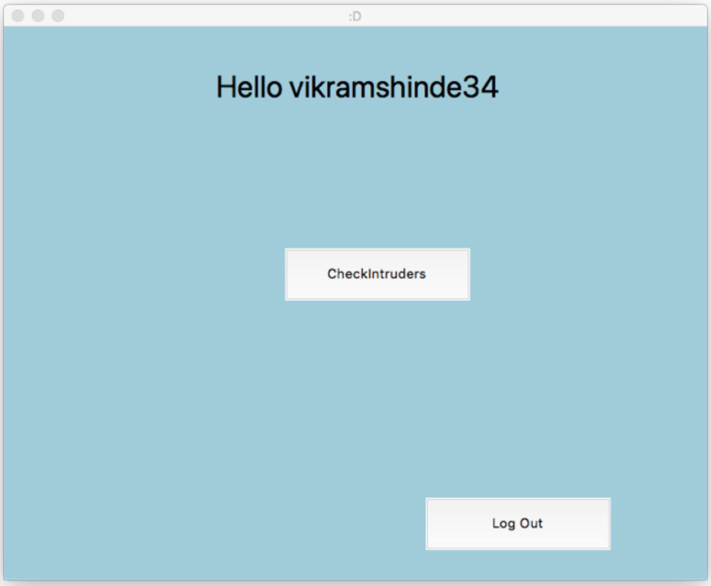
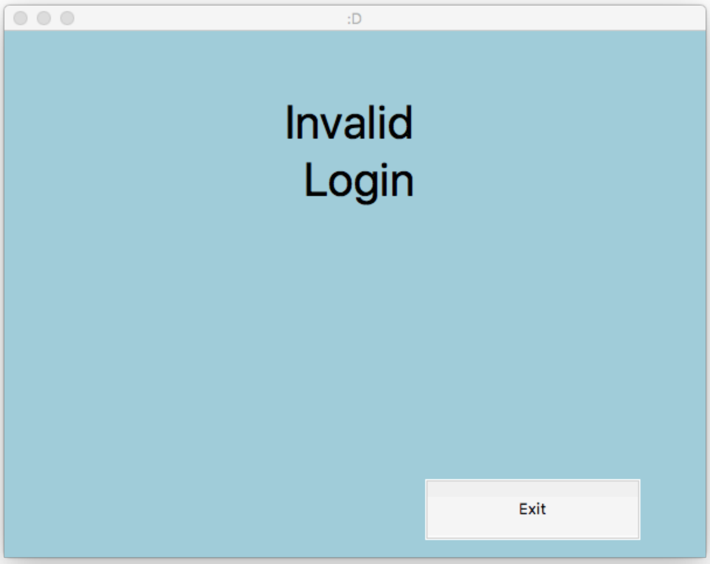


Figure 4.3.6: Valid Login Window



:D

Invalid Login

Figure 4.3.7: Invalid Login

Fig 4.7 shows the screenshot of the page to which the User is redirected to once he/she is successfully logged in.

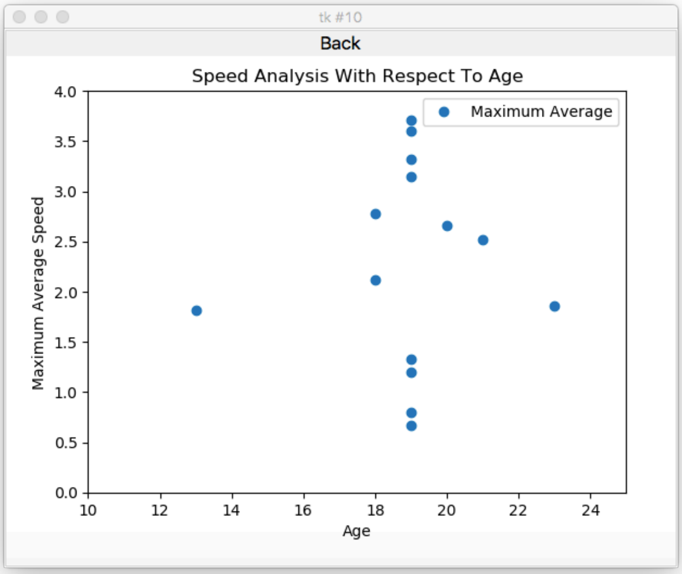


Figure 4.3.8: Analysis

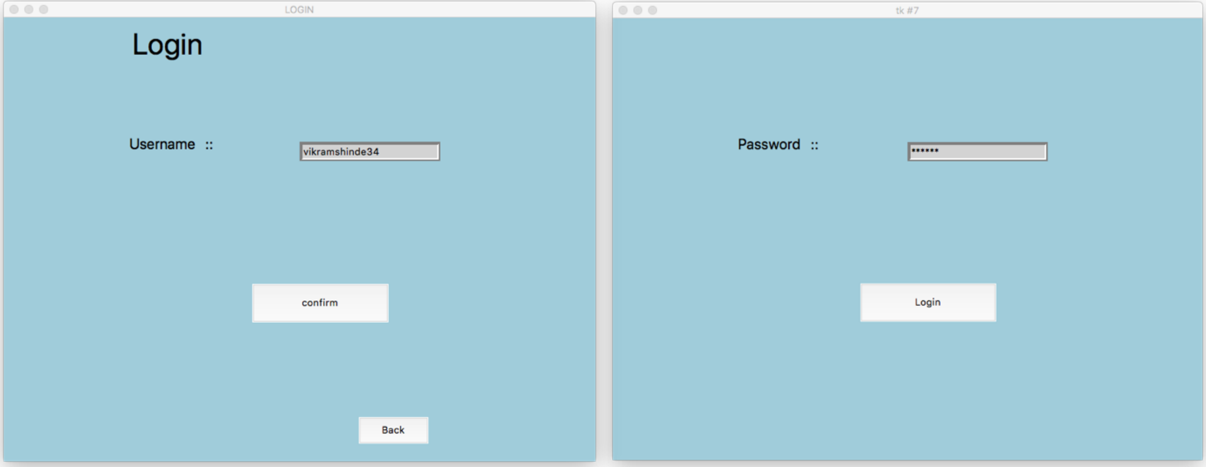
Fig 4.8 shows the graph which is plotted after the analysis of the data collected during user sign-up. The above graph shows the relation between the age of the user and their maximum average speed.

Chapter 5

Results and Discussions

5.1 Main GUI Snapshots

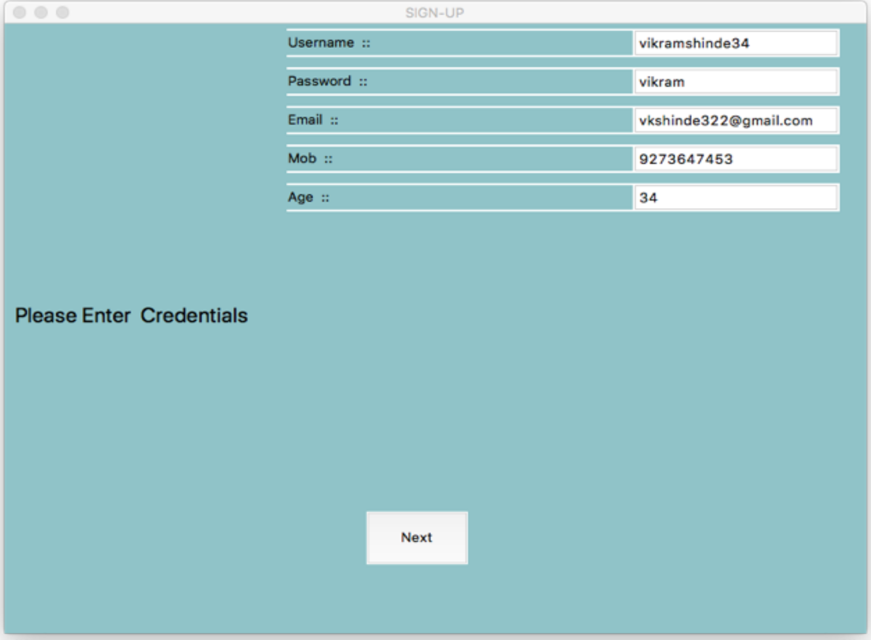
The application provides authentication to users by comparing the typing speed of the user. The fig.5.1 shows the login page and user interface.



Username :: vikramshinde34

Password :: ••••••

Figure 5.1.1: The login page and user interface



Username :: vikramshinde34

Password :: vikram

Email :: vkshinde322@gmail.com

Mob :: 9273647453

Age :: 34

Please Enter Credentials

Figure 5.1.2: User details and signup interface

Fig 5.1.2 shows the user details and the sign-up interface that is displayed to the user. The sign-up interface window also provides a next button which takes the user to the re-type password window.

5.2 Discussions

This Password Pattern Recognition system is a keystroke dynamic application. It allows the user access only when the typing speed is matched. The system runs on Anaconda Console. All the features incorporated work perfectly fine on the deployment devices with no system crashes. The storage is reliable and dynamic. It ensures that the basic functionality of the application is served. Reliability is top priority here.

Chapter 6

Conclusion

Keystroke Dynamics is a two factor security biometric security, hence, for a successful login, firstly password should be known and secondly, typing rhythm should match. In human behaviour security system of any keypad requires to make a programming. In another method of biometrics we require hardware but human behaviour method we generate a secure key to protect our password. This key is generating according to human behaviour for e.g. when user give password he use his typing speed to fill the password. The implementation of keystroke dynamics on desktop is cost effective and compatible as integration of external hardware is not required. The conclusion of this project is based on comparing the stored data of a user with the login input for authentication.

Chapter 7

Future Work

We believe that the future of the keystroke dynamics is no more on desktop application, but in the mobile and internet worlds, because mobile phones are more popular than computers and its use is very democratized. They are more powerful every year (in terms of calculation and memory) and embeds interesting sensors (pressure information with tactile phones). Nowadays, more applications are available in a web browser. These applications use the classical couple of login and password to verify the identity of a user Integrating them a keystroke dynamics verification would harden the authentication process. In order to spread the keystroke modality, it is necessary to solve various problems related to:

- The cross devices problem: We daily use several computers which can have different keyboards on timing resolution. These variability must not have an impact on the recognition performances. Users tend to change often their mobile phone. In an online authentication scheme, it could be useful to not re-enroll the user on its new mobile phone.
- The aging of the biometric data: Keystroke dynamics, is subject to a lot of intra class variability. One of the main reasons is related to the problem of template aging: performances degrade with time because user (or impostors) type differently with time.

References

- ***Araujo, L., Sucupira, L.H.R., J., Lizarraga, M., Ling, L. & Yabu-Uti, J. (2005). User authentication through typing biometrics features, IEEE Transactions on Signal Processing 53(2 Part 2).***
- ***Bergadano, F., Gunetti, D. & Picardi, C. (2002). User authentication through keystroke dynamics, ACM Transactions on Information and System Security (TISSEC) 5(4).***
- ***Janakiraman, R. & Sim, T. (2007). Keystroke dynamics in a general setting, Lecture notes in computer science.***
- ***Kang, P. & Cho, S. (2009). A hybrid novelty score and its use in keystroke dynamics-based. user authentication, Pattern Recognition.***
- ***https://appliedmachinelearning.blog/2017/07/26/user- verification-based-on-keystroke-dynamics-python-code/***