DEVELOPMENT OF USER AUTHENTICATION SYSTEM USING QR CODE AND FACE DETECTION

Minor project report submitted
in partial fulfillment of the requirement for award of the degree of

Bachelor of Technology
in
Computer Science & Engineering

By

YASH AGGARWAL	(20UECS1038)	**(VTU12418)**
MADHVENDRA SINGH	(20UECS0564)	**(VTU12421)**
ASHISH SONKARIA	(20UECS0072)	**(VTU17717)**

Under the guidance of
Dr. C.M. CHIDAMBARANATHAN, M.Tech, Ph.D.,
ASSISTANT PROFESSOR

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING
SCHOOL OF COMPUTING

VEL TECH RANGARAJAN DR. SAGUNTHALA R&D INSTITUTE OF SCIENCE & TECHNOLOGY

(Deemed to be University Estd u/s 3 of UGC Act, 1956)
Accredited by NAAC with A++ Grade
CHENNAI 600 062, TAMILNADU, INDIA

May, 2023

DEVELOPMENT OF USER AUTHENTICATION SYSTEM USING QR CODE AND FACE DETECTION

Minor project report submitted
in partial fulfillment of the requirement for award of the degree of

Bachelor of Technology
in
Computer Science & Engineering

By

YASH AGGARWAL	(20UECS1038)	**(VTU12418)**
MADHVENDRA SINGH	(20UECS0564)	**(VTU 12421)**
ASHISH SONKARIA	(20UECS0072)	**(VTU 17717)**

Under the guidance of
Dr. C.M. CHIDAMBARANATHAN,M.Tech, Ph.D.,
ASSISTANT PROFESSOR

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING
SCHOOL OF COMPUTING

VEL TECH RANGARAJAN DR. SAGUNTHALA R&D INSTITUTE OF SCIENCE & TECHNOLOGY

(Deemed to be University Estd u/s 3 of UGC Act, 1956)
Accredited by NAAC with A++ Grade
CHENNAI 600 062, TAMILNADU, INDIA

May, 2023

CERTIFICATE

It is certified that the work contained in the project report titled ”DEVELOPMENT OF USER AUTHENTICATION SYSTEM USING QR CODE AND FACE DETECTION” by YASH AGGARWAL (20UECS1038),MADHVENDRA SINGH (20UECS0564),ASHISH SONKARIA (20UECS0072)” has been carried out under my supervision and that this work has not been submitted elsewhere for a degree.

Signature of Supervisor
Dr. C.M. Chidambaranathan
Assistant Professor
Computer Science & Engineering
School of Computing
Vel Tech Rangarajan Dr. Sagunthala R&D
Institute of Science & Technology
May, 2023

Signature of Head of the Department
Dr. Muralidhar. M.S
Associate Professor
Computer Science & Engineering
School of Computing
Vel Tech Rangarajan Dr. Sagunthala R&D
Institute of Science & Technology
May, 2023

Signature of the Dean
Dr. V. Srinivasa Rao
Professor & Dean
Computer Science & Engineering
School of Computing
Vel Tech Rangarajan Dr. Sagunthala R&D
Institute of Science & Technology
May, 2023

DECLARATION

We declare that this written submission represents our ideas in our own words and where others' ideas or words have been included, we have adequately cited and referenced the original sources. We also declare that we have adhered to all principles of academic honesty and integrity and have not misrepresented or fabricated or falsified any idea/data/fact/source in our submission. We understand that any violation of the above will be cause for disciplinary action by the Institute and can also evoke penal action from the sources which have thus not been properly cited or from whom proper permission has not been taken when needed.

YASH AGGARWAL

Date: / /

MADHVENDRA SINGH

Date: / /

ASHISH SONKARIA

Date: / /

APPROVAL SHEET

This project report entitled ”DEVELOPMENT OF USER AUTHENTICATION SYSTEM USING QR CODE AND FACE DETECTION” by YASH AGGARWAL (20UECS1038), MADHVENDRA SINGH (20UECS0564), ASHISH SONKARIA (20UECS0072) is approved for the degree of B.Tech in Computer Science & Engineering.

Examiners **Supervisor**

Dr. C.M. Chidambaranathan, M.Tech.,Ph.D

Date: / /

Place:

ACKNOWLEDGEMENT

We express our deepest gratitude to our respected **Founder Chancellor and President Col. Prof. Dr. R. RANGARAJAN B.E. (EEE), B.E. (MECH), M.S (AUTO),D.Sc., Foundress President Dr. R. SAGUNTHALA RANGARAJAN M.B.B.S.** Chairperson Managing Trustee and Vice President.

We are very much grateful to our beloved **Vice Chancellor Prof. S. SALIVAHANAN,** for providing us with an environment to complete our project successfully.

We record indebtedness to our **Professor & Dean, Department of Computer Science & Engineering, School of Computing, Dr. V. SRINIVASA RAO, M.Tech., Ph.D.,** for immense care and encouragement towards us throughout the course of this project.

We are thankful to our **Head, Department of Computer Science & Engineering,Dr.M.S. MURALI DHAR, M.E., Ph.D.,** for providing immense support in all our endeavors.

We also take this opportunity to express a deep sense of gratitude to our Internal Supervisor **Dr. C.M. CHIDAMBARANATHAN,M.Tech, Ph.D,** for his cordial support, valuable information and guidance, he helped us in completing this project through various stages.

A special thanks to our **Project Coordinators Mr. V. ASHOK KUMAR, M.Tech., Ms. C. SHYAMALA KUMARI, M.E.,** for their valuable guidance and support throughout the course of the project.

We thank our department faculty, supporting staff and friends for their help and guidance to complete this project.

YASH AGGARWAL (20UECS1038)
MADHVENDRA SINGH (20UECS0564)
ASHISH SONKARIA (20UECS0072)

ABSTRACT

User authentication is a way used to verify the identity of users accessing an application or system.It involves different techniques, such as password-based authentication, two-factor authentication, and biometric authentication, and requires various components, such as user management, password storage, and authentication mechanisms. In this the readers will know about various authentication techniques and their strengths and weaknesses, along with best practices for securing user authentication. It is important to continuously evaluate and improve user authentication systems to keep up with evolving security threats and protect sensitive user data.In recent years, the development of secure and efficient user authentication systems has become increasingly important due to the rise in cyber threats and data breaches.
The user authentication process starts with the user accessing the web application and entering their credentials and registering on the application.Once the user has entered their credentials, a unique QR(Quick Response) code is generated on the web application. The user then opens the mobile application and scans the QR code with camera. After the QR code is scanned, the web application captures the user's face and compares it to the previously stored facial features to authenticate the user. This project proposes a new user authentication system that combines QR code and face detection technologies to enhance security and usability. First, the use of QR codes makes the authentication process quick and easy for users. Second, system's facial recognition technology provides an additional layer of security, making it difficult for unauthorized users to access the system. Finally, the system's web and mobile applications can be easily integrated into existing IT infrastructures, making it an attractive option for organizations looking to improve their user authentication processes.

Keywords: Biometric-authentication, Password storage,QR code,Two-factor authentication,User authentication,User management,

LIST OF FIGURES

LIST OF ACRONYMS AND ABBREVIATIONS

LIST OF ACRONYMS AND ABBREVIATIONS

AES	Advanced Encryption Standard
AI	Artificial Intelligence
DFD	Data-flow diagram
ID	Identity Document
ISMS	Information Security Management System
ML	Machine Learning
OS	Operating System
PIN	Personal Identification Number
UML	Unified Modeling Language
QR	Quick Response

TABLE OF CONTENTS

Chapter 1

INTRODUCTION

1.1 Introduction

User authentication is the process of verifying the identity of a user who wants to access an application or system.The goal of a user authentication system is to ensure that only authorized users are granted access while preventing unauthorized access to sensitive data and resources.

Different authentication techniques, such as password-based authentication , two-factor authentication, and biometric authentication, can be used in a user authentication system.The aim of this project is to develop a user authentication system using QR code and face detection. Authentication is a crucial aspect of security, and it ensures that only authorized users can access a system or data. Traditional methods of authentication, such as passwords and PINs, can be vulnerable to hacking and phishing attacks. To address this issue, biometric authentication techniques, such as face recognition, have been increasingly used in recent years. However, face recognition systems can still face challenges, such as changes in lighting conditions and facial expressions.

A user authentication system should be robust, reliable, and easy to use to ensure the security and integrity of modern applications and systems.User authentication systems can help prevent various types of attacks, such as brute-force attacks, password guessing attacks, and credential stuffing attacks.User authentication systems should be periodically audited and tested for vulnerabilities to ensure that they remain secure and effective.The proposed system has several potential benefits, including increased security, convenience, and accessibility. By combining QR codes and face detection, users can authenticate themselves quickly and securely, without the need for passwords or PINs. The system can be implemented in various applications, such as online banking, e-commerce, and social media platforms, to enhance user security and privacy.

1.2 Aim of the project

The aim of a user authentication system is to verify the identity of users accessing an application or system, ensuring that only authorized users are granted access while preventing unauthorized access to sensitive data and resources.

1.3 Project Domain

The project have two domains: image processing and web development.
Image processing : Image processing is a technique used to analyze, manipulate, and enhance digital images using various algorithms and techniques. It involves acquiring digital images using cameras, scanners, or other imaging devices, and then applying a set of mathematical operations to extract useful information or improve the visual quality of the image.
Image processing tasks can range from simple operations such as adjusting brightness and contrast to more complex tasks such as object recognition and pattern recognition. Some of the common tasks in image processing include image enhancement, image restoration, image compression, image segmentation, object recognition, and pattern recognition.
Programming languages such as Python, is used to implement image processing algorithms. There are also many libraries and frameworks available, such as OpenCV, scikit-image, and Flask, to facilitate image processing tasks. Overall, image processing has become an indispensable tool in various fields and continues to evolve with the advancement of technology.
Web Development : Web development is the process of creating websites or web applications using various technologies such as HTML, CSS, and JavaScript. It involves designing, building, and maintaining websites that can be accessed over the internet or on a local network.
Web development can be divided into two main categories: front-end development and back-end development. Front-end development involves creating the user interface of a website, which includes designing the layout, implementing visual elements, and adding interactive features. Back-end development involves building the server-side of a website, which includes creating and managing databases, server-side programming, and handling user requests.

1.4 Scope of the Project

The scope might be viewed as implementing an authentication system involving several steps, each of which is critical to building a secure and effective authentication system. The first step is to define the authentication requirements, which involves identifying the types of users who will be accessing the system, the types of data or resources that need to be protected, and the level of security required. This step will guide the design and implementation of the authentication system.

The next step is to develop the authentication system architecture. This involves selecting the appropriate authentication methods and technologies to use, such as passwords, biometrics, two-factor authentication, or single sign-on. The architecture should also consider factors such as data storage, access control, and user management.

Once the architecture has been designed, the authentication system can be implemented. This involves developing the code that will handle user authentication, creating user accounts and passwords, and integrating the authentication system with other parts of the application. The implementation should follow industry best practices and security guidelines to ensure that the system is robust and secure.

Testing the authentication system is crucial to ensure that it is working correctly and securely. This step involves testing various scenarios, such as invalid login attempts, password resets, and session timeouts. The testing should be comprehensive and include both functional and security testing to identify any vulnerabilities or weaknesses in the system.

After testing is complete, the authentication system can be deployed to production. This step involves configuring the system for the appropriate environment, such as a web server or a cloud-based platform. The deployment should follow industry best practices and security guidelines to ensure that the system is secure and performs optimally.

Maintaining the authentication system is crucial to ensure that it continues to work correctly and remain secure. This step involves regularly updating the system with security patches, monitoring user activity for suspicious behavior, and reviewing the system's logs for potential issues. The maintenance should be proactive and include ongoing monitoring and testing to identify and address any security issues or vulnerabilities.

Chapter 2

LITERATURE REVIEW

[1] M. Monwar Hossain,et.al made “A Survey of User Authentication Systems”, which provides an overview of different types of user authentication systems and compares their strengths and weaknesses. A qualitative survey of user authentication systems being used in today’s environment is presented here and a comparative study of various authentication mechanisms used in the world of Information security by various researchers is shown.

[2] Asad Masood Khattak,et.al implemented an “AI based Login System using Facial Recognition”, The system analyzes the application of Face detection systems to authenticate and login users It presents the prototype system implemented with the usage of a Flask server, requesting face recognition services from Amazon’s Rekognition. With rapid growth in the application of AI, Access Control Systems are walking in a new technology lane. Powered by deep learning technologies or cognitive analytics, login pages can implement more secure, efficient, and easy to use authentication systems.

[3]Saad Alshahrani,et.al made “A Review of Biometric Authentication Technologies: Applications, Challenges, and Research Directions”, which developed a User authentication in computer system is done based on certain security measures like passwords, keys, id cards, pin etc. however, the misuse and theft of these security measures are also increasing due to day by day advancement of technology. This paper provides a comprehensive review of biometric authentication technologies and covers recent advances in biometric authentication mechanisms, such as behavioral biometrics, multimodal biometrics, and deep learning-based biometrics.

[4]Sanjay Kumar, Syed Akbar Abbas Jafri,et.al implemented an “A New User Identity Based Authentication, Using Security and Distributed for Cloud Computing”, which implemented a new advanced security architecture for user identification which includes two factor authentication, AES based file encryption and decryption of data uploaded on cloud is presented.

[5]Ben Wycliff Mugalu, Rodrick Calvin Wamala,et.al implemented a ”Face Recog-

nition as a Method of Authentication in a Web-Based System", The system analyzes Biometric authentication, Face authentication, Face detection, face recognition using AI, data collection. This paper includes a comparison of combinations of detection and classification algorithms with FaceNet for face recognition.

[6]Paul C. van Oorschot, et.al made "User Authentication—Passwords, Biometrics and Alternatives", The system analyzes the user authentication—humans being authenticated by a computer system. Chapter 4 addresses machine-to-machine authentication and related cryptographic protocols. The main topics of focus here in are passwords, hardware-based tokens, and biometric authentication. It also discuss password managers, CAPTCHAs, graphical passwords, and background on entropy relevant to the security of user-chosen passwords.

[7]S. Khokad and V. Kala,et.al made "A study of SLIDE Algorithm: Revolutionary AI Algorithm that Speeds Up Deep Learning on CPUs," which predicts Deep learning as the emerging technology, having endless applications and heavy investment from industries. Deep learning constructs a model from many simple computational layers stacked up, resulting in a deep network.

[8]Munir Hussain, Amjad Mehmood, Shafiullah Khan,et.al implemented the "Authentication Techniques and Methodologies used in Wireless Body Area Networks", The system analyzes the ways for improving the authentication process in WBANs, an overview of WBAN and their characteristics, various authentication types and classification of authentication schemes has been done.

[9] Muhammad Sajjad, Salman Khan, Tanveer Hussain, Arun Kumar Sangaih,et.al made a "CNN-based anti-spoofing two-tier multi-factor authentication system" which implemented the two-fold scheme: Tier I integrates fingerprint, palm vein print and face recognition to match with the corresponding databases, and Tier II uses fingerprint, palm vein print and face anti-spoofing convolutional neural networks (CNN) based models to detect spoofing.

[10] Verena Zimmermann,et.al made "The password is dead, long live the password – A laboratory study on user perceptions of authentication schemes" which investigated objective features and subjective user perceptions of twelve different authentication schemes in a controlled laboratory setup. The results indicated that the password is the most preferred authentication scheme despite its downsides such as its high cognitive load for users.

Chapter 3

PROJECT DESCRIPTION

3.1 Existing System

Security solutions that authenticate and manage user access to systems have never been more important. Increasingly organizations are relying on cloud applications like Microsoft 365, Google Workspace, and many more for critical business functions. Securing access to these systems with user authentication tools is key to protecting against data loss and security breaches.User authentication tools verify that only authorized users can access cloud applications and company accounts. These systems are designed to ensure that only the right people can access the right business systems, and they offer a range of features which help to enhance basic username and password account security.
Textual Passwords should be easy to remember at the same time hard to guess. But if a textual password is hard to guess then it is very difficult to remember also. One of the major issues with multi-factor authentication is that it's an obstacle for people who want to login to their accounts as quickly and smoothly as possible. This could cause customers to abandon the process during onboarding or leave the platform in frustration.Although biometrics are extremely hard to spoof, it is not impossible. And once compromised, the data cannot be reset. This is a significant drawback as passwords can be reset and changed.

3.2 Proposed System

The proposed system is a user authentication system in which will use Face authentication and QR code authorization for logging in. It will enhance the security of the system. The aim of this user authentication system is to verify the identity of users accessing an application or system, ensuring that only authorized users are granted access while preventing unauthorized access to sensitive data and resources.It is be robust, reliable, and easy to use to ensure the security and integrity of modern appli-

cations and systems.
This User authentication system can help prevent various types of attacks, such as brute-force attacks, password guessing attacks, and credential stuffing attacks. It will ensure that only authorized users can access the data as face authentication is used so no one else can access the data plus QR code authentication is also required which is unique for each user and if any other QR code is used the user will not be able to login. And if the face is not verified in some seconds it will throw time out exception.

3.3 Feasibility Study

3.3.1 Economic Feasibility

The economic feasibility of a User authentication system using QR code and face detection depends on various factors, including the costs of development, deployment, and maintenance, as well as the potential benefits and cost savings it can bring to an organization. A thorough cost-benefit analysis, considering the organization's specific requirements, risks, and budget, is essential in determining the economic viability of implementing such a system.
Development costs may include hiring data scientists and acquiring datasets, while deployment costs may involve hardware, software, and integration requirements. Maintenance costs may include updates and ongoing investments in data collection and model updates. Benefits and cost savings may arise from improved detection accuracy. ROI analysis can help assess economic feasibility, and organizations should consider cost-benefit trade-offs in the context of their specific security needs and budget constraints.

3.3.2 Technical Feasibility

The Development of user authentication system using QR code and face detection offer promising capabilities, their technical feasibility depends on factors such as data availability, feature extraction, model selection and tuning, scalability and efficiency, model updates. Proper planning, expertise, and resources are required to address these technical challenges and develop an effective and reliable Development of user authentication system using QR code and face detection.

3.3.3 Social Feasibility

The Development of user authentication system using QR code and face detection offer significant potential in improving security, their social feasibility needs to be carefully evaluated, taking into account privacy, ethical, usability, scalability, and societal considerations. A well-designed system that addresses these concerns can contribute to enhanced security while minimizing potential negative impacts on society. The social feasibility of Development of user authentication system using QR code and face detection encompasses several important considerations. Ethical concerns such as bias, fairness, and transparency should be addressed to prevent discrimination and promote accountability.

3.4 System Specification

3.4.1 Hardware Specification

• Processor: Intel Core i5 or higher, AMD RYZEN 5 or higher.
• RAM: 8GB or higher.
• GPU: GTX1060(2GB VRAM minimum) or higher.
• Web Camera: 720p at 30 fps HD camera or higher.

3.4.2 Software Specification

• OS: A compatible os to run all the framework and libraries, suggested- Windows,Linux and macOS.
• Programming Language: Python (v3.8 or higher)
• Neccessary Libraries: Scikit-learn, TensorFlow, Keras, XGBoost should be preinstalled before executing the code.

3.4.3 Standards and Policies

Standard Used:ISO/IEC 27001:2013
Information Security Management System
Information Security Management System (ISMS) standard that provides a systematic approach to managing sensitive company information so that it remains secure. Compliance with this standard can help ensure the system is secure and protected against unauthorized access.

Standard Used:ISO/IEC 27002:2013

Code

Code of practice for information security controls that provides a framework for designing and implementing security controls. Compliance with this standard can help ensure that security controls are effectively implemented in the system. **Standard Used:ISO/IEC 30107-3:2019**

Biometric Presentation Attack Detection

Biometric Presentation Attack Detection (PAD) standard that provides guidance on detecting and preventing biometric presentation attacks. Compliance with this standard can help ensure that the face detection component of the system is robust and resistant to spoofing attacks. .

Standard Used: ISO/IEC 19794-5:2011 Biometric data interchange formats

Information technology - Biometric data interchange formats - Part 5: Face image data standard that provides guidelines for the interchange of face image data. Compliance with this standard can help ensure that the system can exchange biometric data with other systems that comply with the standard.

Chapter 4

METHODOLOGY

4.1 General Architecture

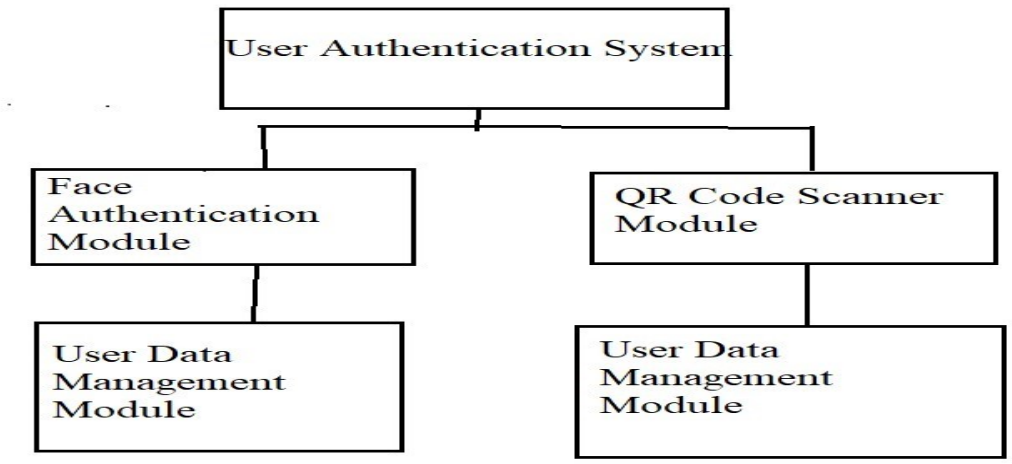


Figure 4.1: **Architecture Diagram**

Figure 4.1 depicts the research design for user authorization Using The Development of user authentication system using QR code and face detectio is typically a quantitative research design that involves using machine learning algorithms to classify malware samples based on their behavior and characteristics.The Development of user authentication system using QR code and face detection uses two-step verification first is Face authentication module and the second step is QR code authorization.This research design does not typically involve a case study, survey, or experiment, but rather involves a more data-driven approach using statistical and machine learning techniques.

4.2 Design Phase

4.2.1 Data Flow Diagram

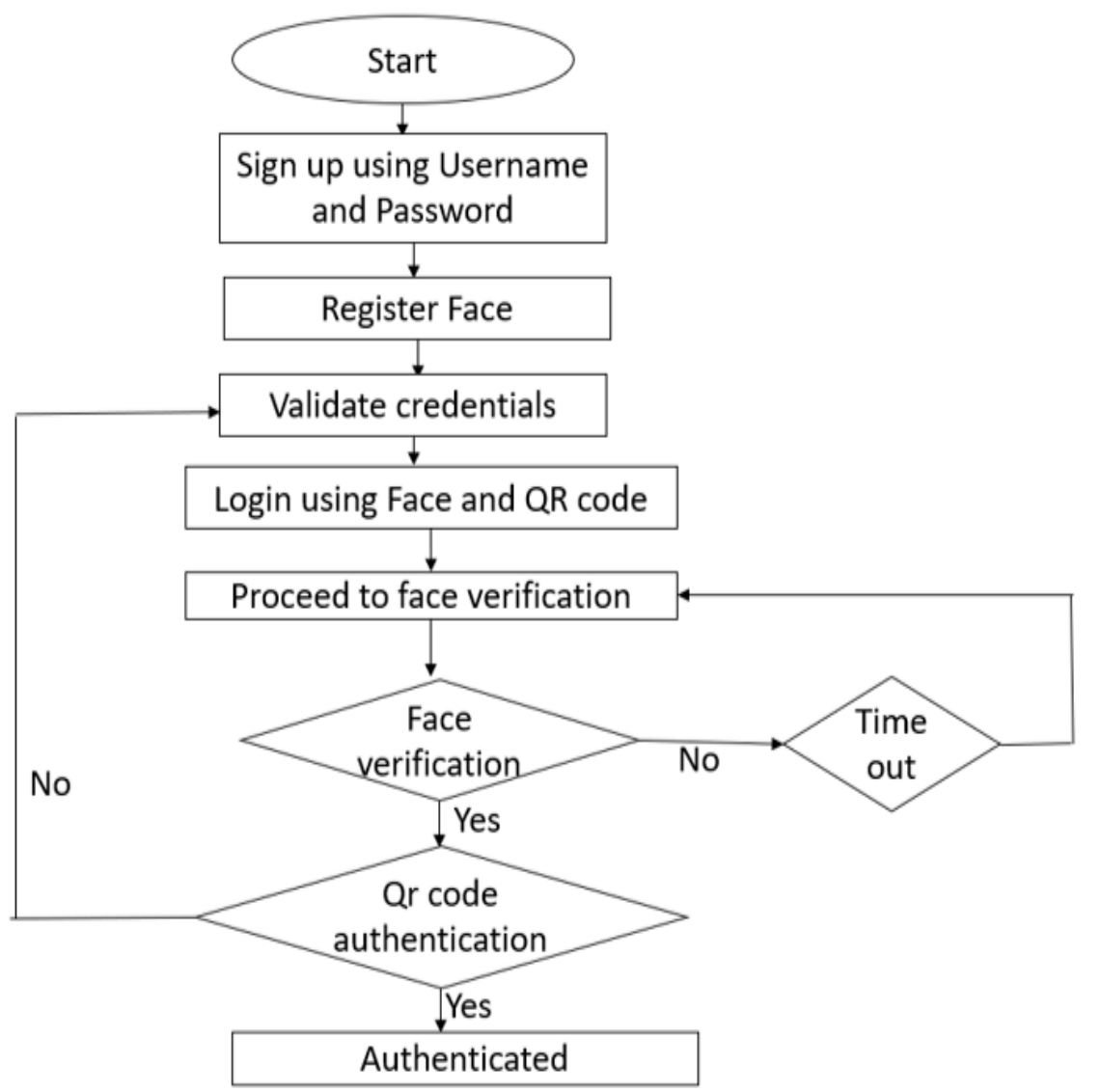


Figure 4.2: **Data Flow Diagram**

Figure 4.2 depicts a Data Flow Diagram (DFD) for the Development of User Authentication system using face detection and QR code authentication. The process starts with sign up using username and password, In the next step the user will register face then the credentials will be validated. Then the user can login by going through face detection and then through QR code authentication. If the face is not detected then it will time out and if the QR code is not authenticated the process will be repeated.

4.2.2 Use Case Diagram

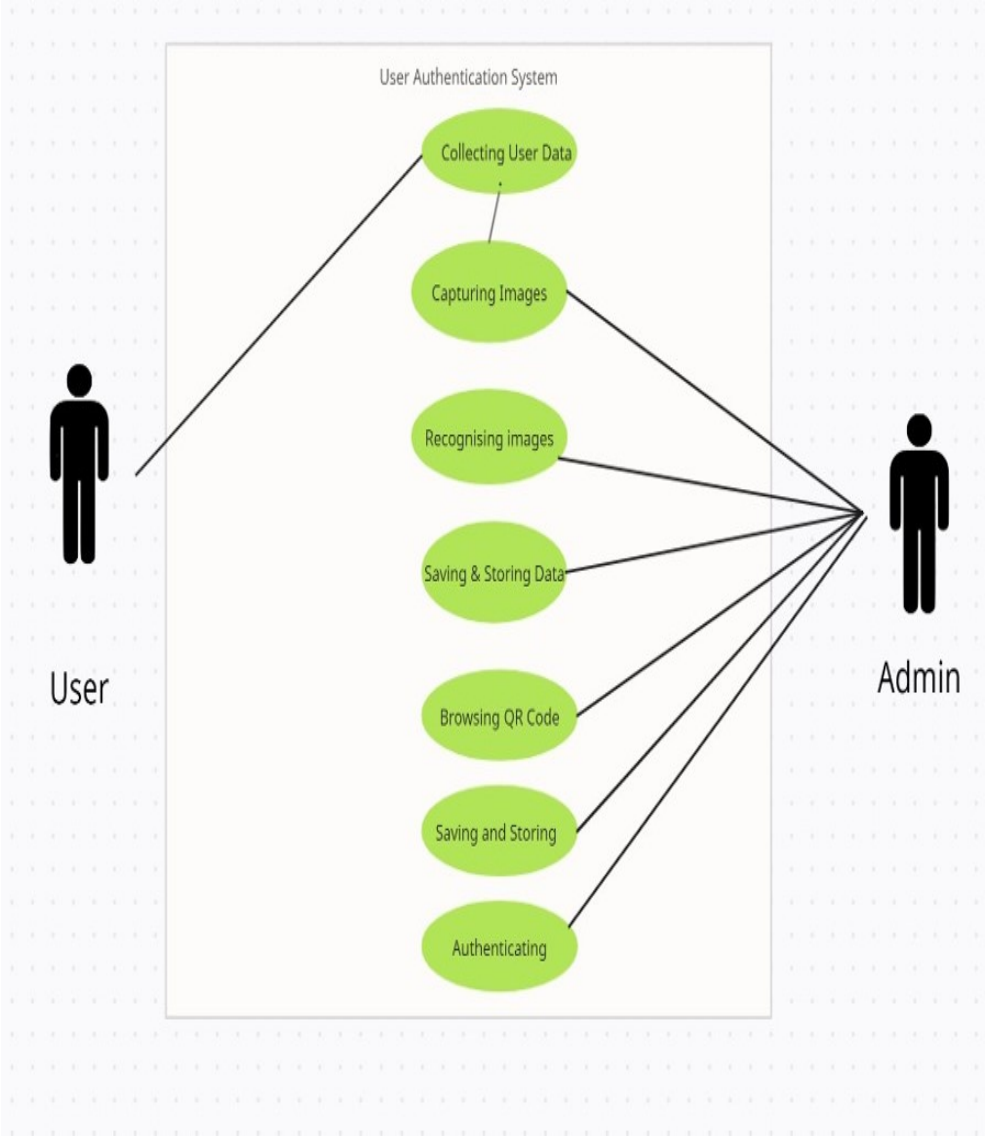


Figure 4.3: **Use Case Diagram**

Figure 4.3 depicts a use case diagram is a visual representation of the interactions and relationships within the Development of User Authentication system using face detection and QR code authentication. It includes two roles user and admin. Initially the user data will be collected from the user by capturing images of the face so that it can be stored in the database for the further face detection. Then the admin will recognise images, then the data will be saved and stored. then in the next step QR code will be browsed then it will be saved and stored in the database. In the last step the user information will be authenticated.

4.2.3 Sequence Diagram

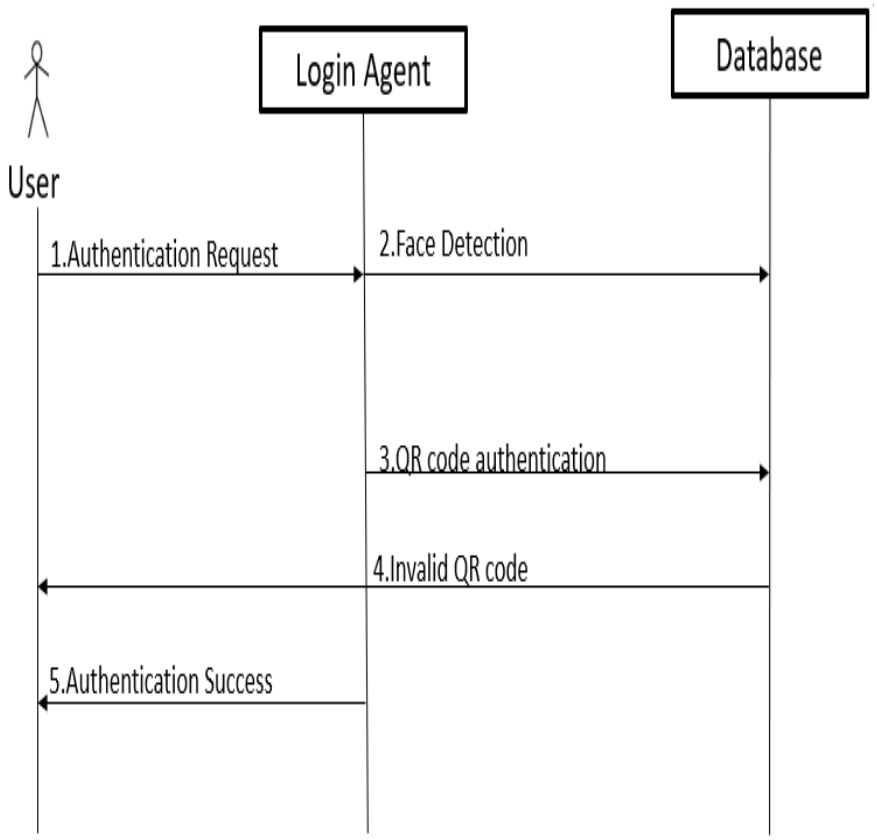


Figure 4.4: **Sequence Diagram**

Figure 4.4 depicts a sequence diagram is a Unified Modeling Language (UML) diagram that illustrates how a group of objects interact and operate with each other sequentially. The sequence diagram consists of a group of objects that are represented by lifelines and the messages that they exchange over time during the interaction.First the user will send an authentication request then it will proceed to face detection after that it will proceed to QR code authentication if the QR code is invalid the user will not be allowed to login and if the QR code is verified the user will be allowed to login and finally the the user will be authenticated.

4.2.4 Activity Diagram

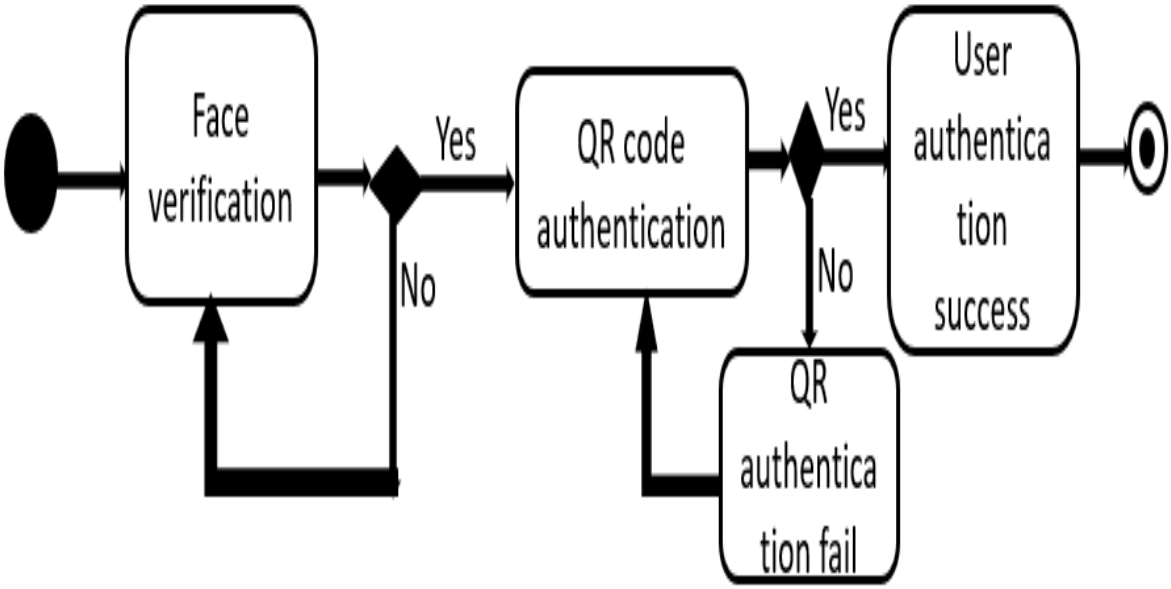


Figure 4.5: **Activity Diagram**

Figure 4.5 depicts a UML, the activity diagram for the Development of User Authentication system using face detection and QR code authentication. The process starts with sign up using username and password, In the next step the user will register face then the credentials will be validated.Then the user can login by going through face detection If the face is not detected then it will time out and this step is repeated .The face will be verified by the system,followed by this the user need to show the QR code for authentication.If it is failed the user authentication will not be processed,the user need to show QR code again and after scanning again the user authentication is done.

4.3 Algorithm & Pseudo Code

4.3.1 Algorithm

Step1: Start the authentication process

Step2: Capture an image of the user's face using a camera.

Step3: Use a face detection algorithm to detect the face in the captured image.

Step4: If the face is not detected, prompt the user to try again.

Step5: If the face is detected, match it with the stored facial features of the user.

Step6: If the face matches the stored facial features, generate a unique QR code for the user.

Step7: Display the QR code on the screen.

Step8: Instruct the user to scan the QR code using a QR code scanner app on their smartphone.

Step9: Wait for the user to scan the QR code.

Step10: When the QR code is successfully scanned, verify it against the stored QR code for the user.

Step11: If the QR code matches, authenticate the user and grant access.

Step12: If the QR code does not match, prompt the user to try again or contact the system administrator for assistance.

Step13: End the authentication process.

4.3.2 Pseudo Code

1.Initialize face recognition system

InitializeFRSystem()

2. Capture image from camera

capturedimage = CaptureImage()

3. Preprocess captured image for face detection

preprocessedimage = PreprocessImage(capturedimage)

4. Detect face in the preprocessed image

facelocation = DetectFace(preprocessedimage)

if facelocation != None:

facefeatures = ExtractFeatures(preprocessedimage, facelocation)

5. Compare extracted face features with registered faces

match = CompareFeatures(facefeatures, registeredfaces)

6. If a match is found, grant access

if match == True:

GrantAccess()

7. If no match is found, deny access

else:

DenyAccess()

8. If no face is detected, deny access

else:

DenyAccess()

7.. Generate QR code

qrcode = generate qrcode(userid)

8.Display QR code on screen for user to scan

display qrcode(qrcode)

9. Wait for user to scan QR code

userscanned = waitforscan()

10. Check if scanned user ID matches the one generated

if userscanned == userid:

11. User is authenticated

authenticateuser()

else:

12. User ID does not match

displayerrormessage(”Authentication failed”)

4.4 Module Description

4.4.1 Module 1: Setting up enviornment

Setting up an environment involves creating a development environment that provides all the tools and resources necessary to develop, test, and deploy an application. The environment should be set up in a way that is consistent with the production environment to ensure that the application works correctly and optimally when deployed.

4.4.2 Module 2: Importing necesarry libraries

Importing necessary libraries is a critical step in many programming languages, especially in languages such as Python, which rely heavily on libraries and modules to provide additional functionality. Libraries contain pre-written code that can be imported into a program to save time and effort in coding common functions or tasks.

4.4.3 Module 3: Execution Testing

Executing and testing the program and web page. It includes face detection and QR code scanning. Giving input's and generating the required output.

4.5 Steps to execute/run/implement the project

4.5.1 Step 1: Setting up enviornment

Setting-up necessary environment such as choosing the IDE(Integrated Development Environment), installing necessary libraries which include Flask, OpenCv, numpy, pandas, Scikit-learn, pyzbar etc. Creating necessary folder's and files such as python file, html file, Attendance folder for storing user's login information, templates folder's etc.

4.5.2 Step 2: Importing Libraries

Importing necessary libraries and modules in the python file and writing the required code. On the other side, creating a Sign Up and Login web page using html, css and javascript.

4.5.3 Step 3: Execution

Executing and testing the program and web page.It includes face detection and barcode/QR code scanning,giving input and generating the required output.The process starts with sign up using username and password,In the next step the user will register face then the credentials will be validated. Now the user can login by going through face detection and then through QR code authentication.If the face is not detected,it will time out. If the QR code and face matches with the registered data, authenticate the user and grant access. End the authentication process.

Chapter 5

IMPLEMENTATION AND TESTING

5.1 Input and Output

5.1.1 Input Design

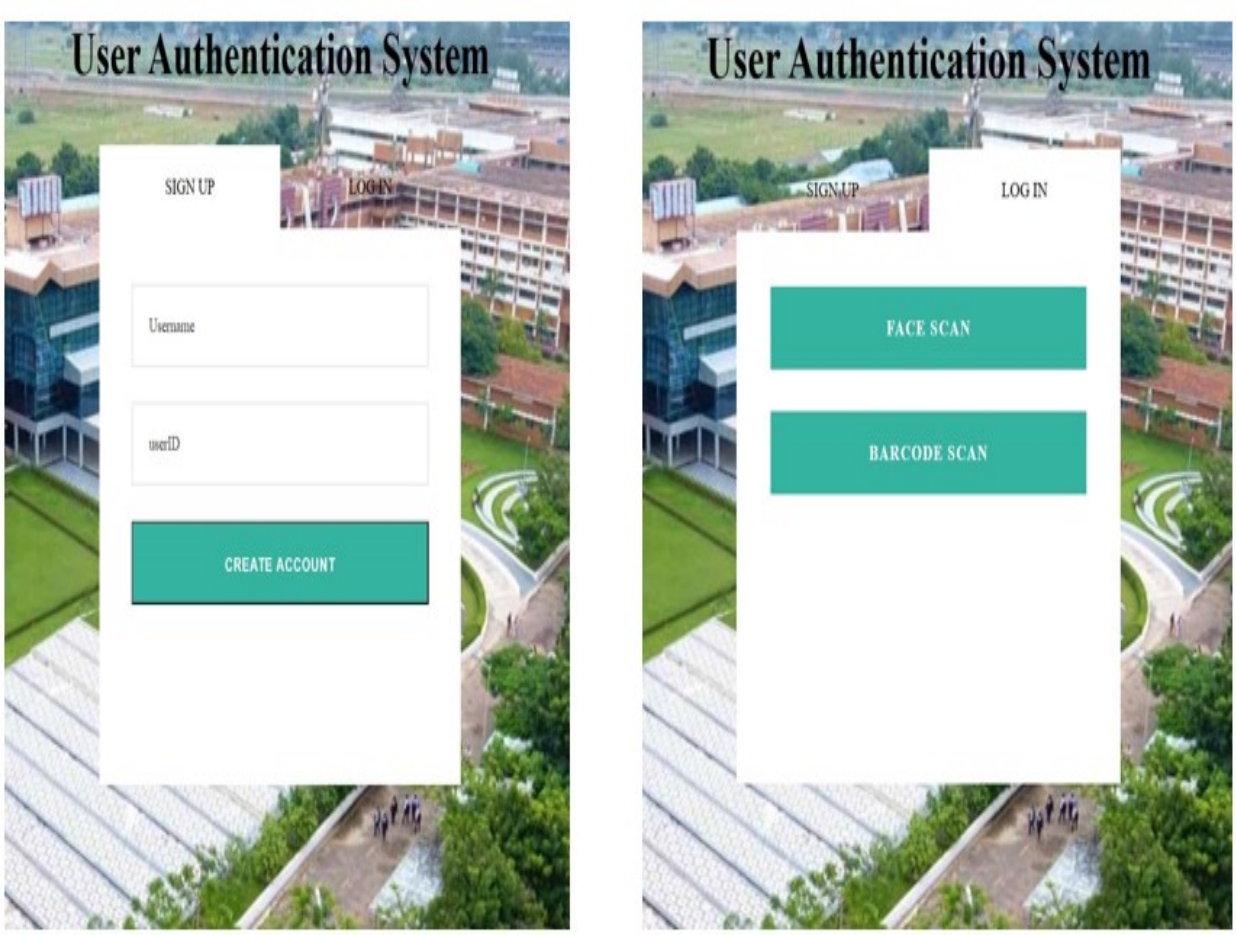


Figure 5.1: **Input Design**

Figure 5.1 depicts the interface of the User Authentication system, first image shows the login page and second image shows sign in page where have to register the face and QR code.Design inputs are the physical and performance characteristics of a device that are used as a basis for device design.

5.1.2 Output Design

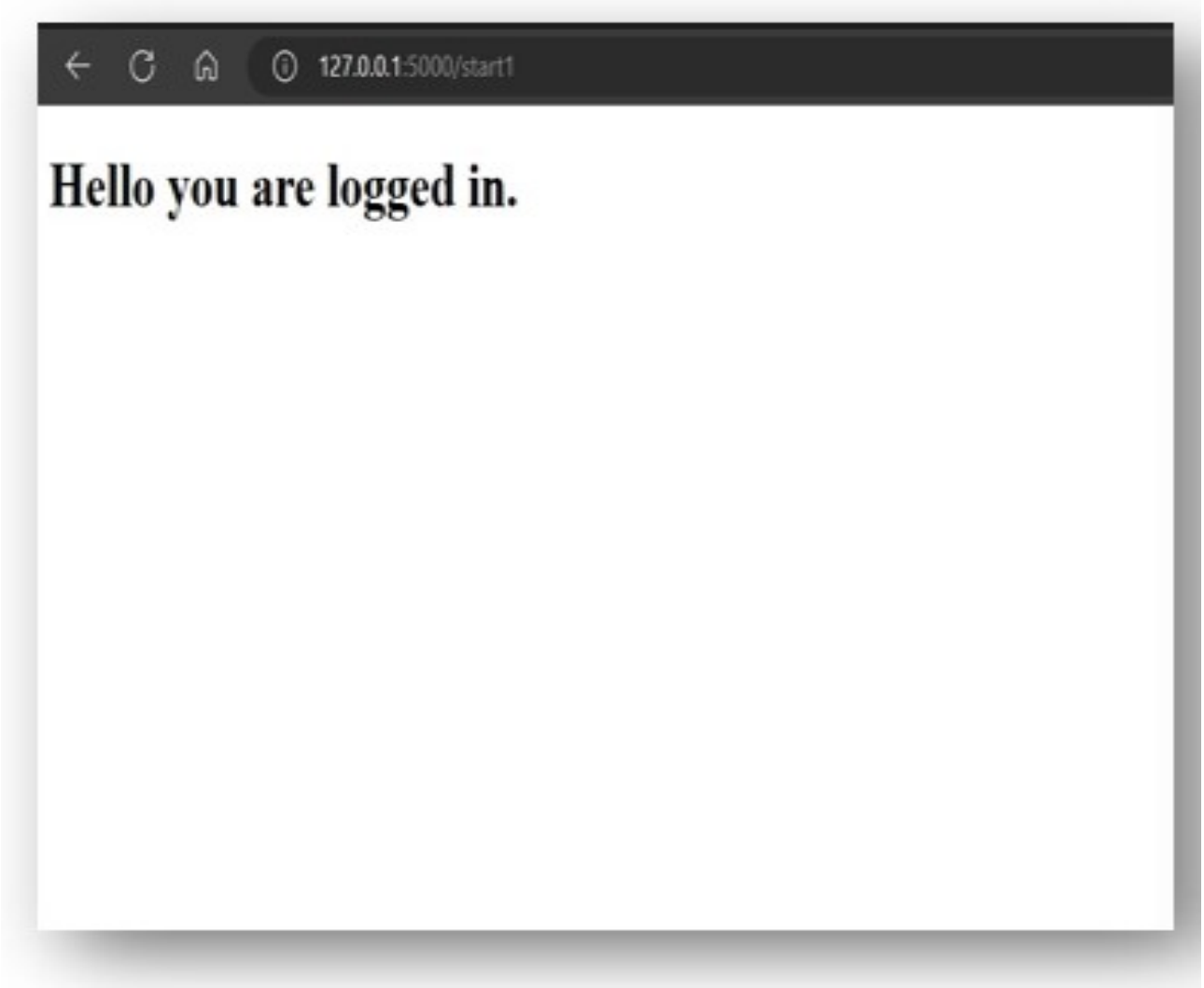


Figure 5.2: **Output Design**

Figure 5.2 depicts Output page which will be displayed after you have logged in on to the user interface used for taking input.Here two method are used for the input(Face Scan or BarCode Scan.).After the input is taken the code will process the input data and crosscheck with the data stored in the database during the signup process.The bar code will be processed by using OpenCv library.

5.2 Testing

5.3 Types of Testing

5.3.1 Unit testing

Input

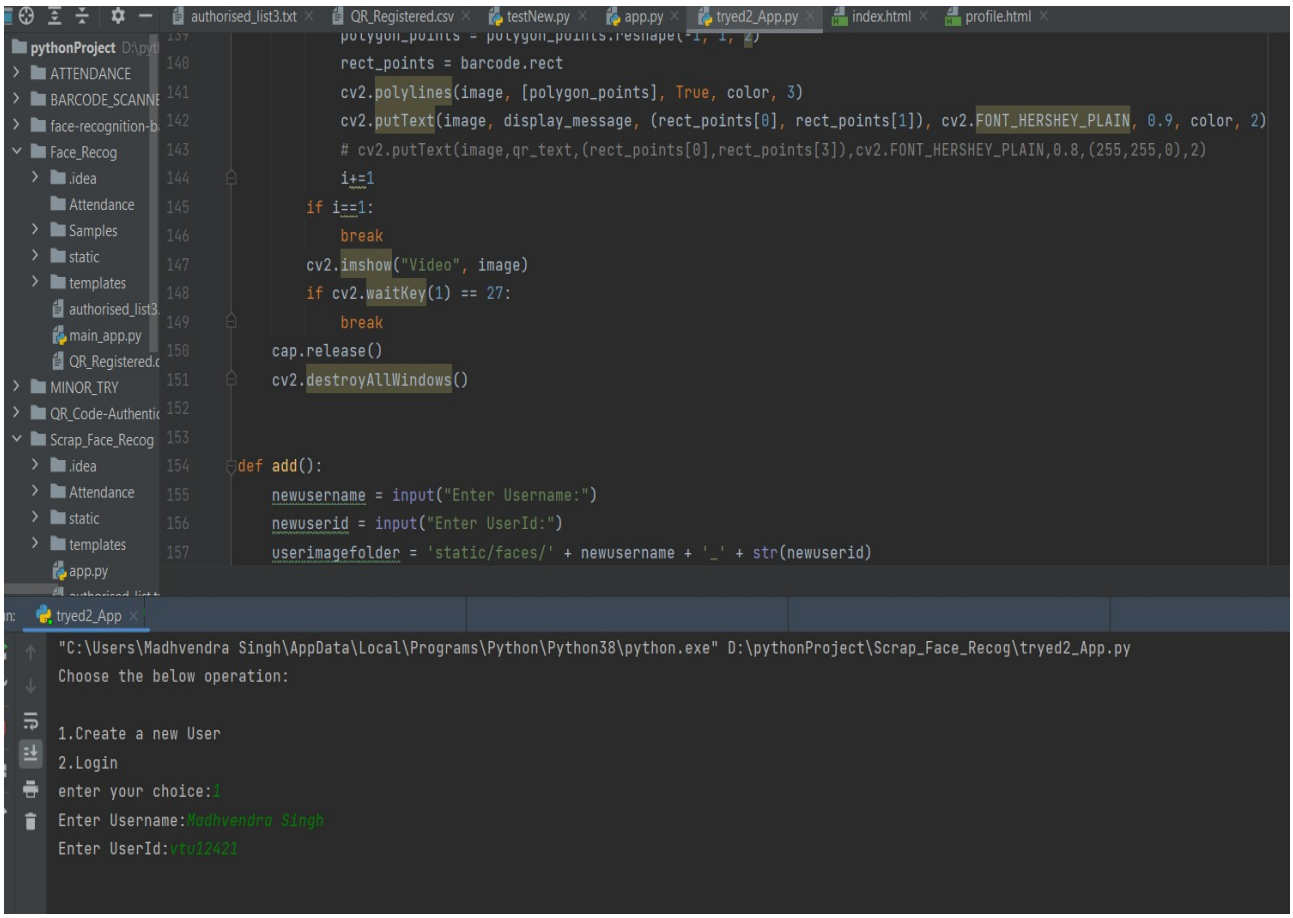


Figure 5.3: **Unit Testing Input**

Figure 5.3 depicts def add() function to get the Username , UserId and image which will be updated in the database.Here ,the OpenCv library of the python is imported. Unit testing involves the testing of each unit or an individual component of the software application.It is the first level of functional testing. The aim behind unit testing is to validate unit components with its performance. A unit is a single testable part of a software system and tested during the development phase of the application software. The purpose of unit testing is to test the correctness of isolated code. A unit component is an individual function or code of the application. White box testing approach used for unit testing and usually done by the developers.

Test result

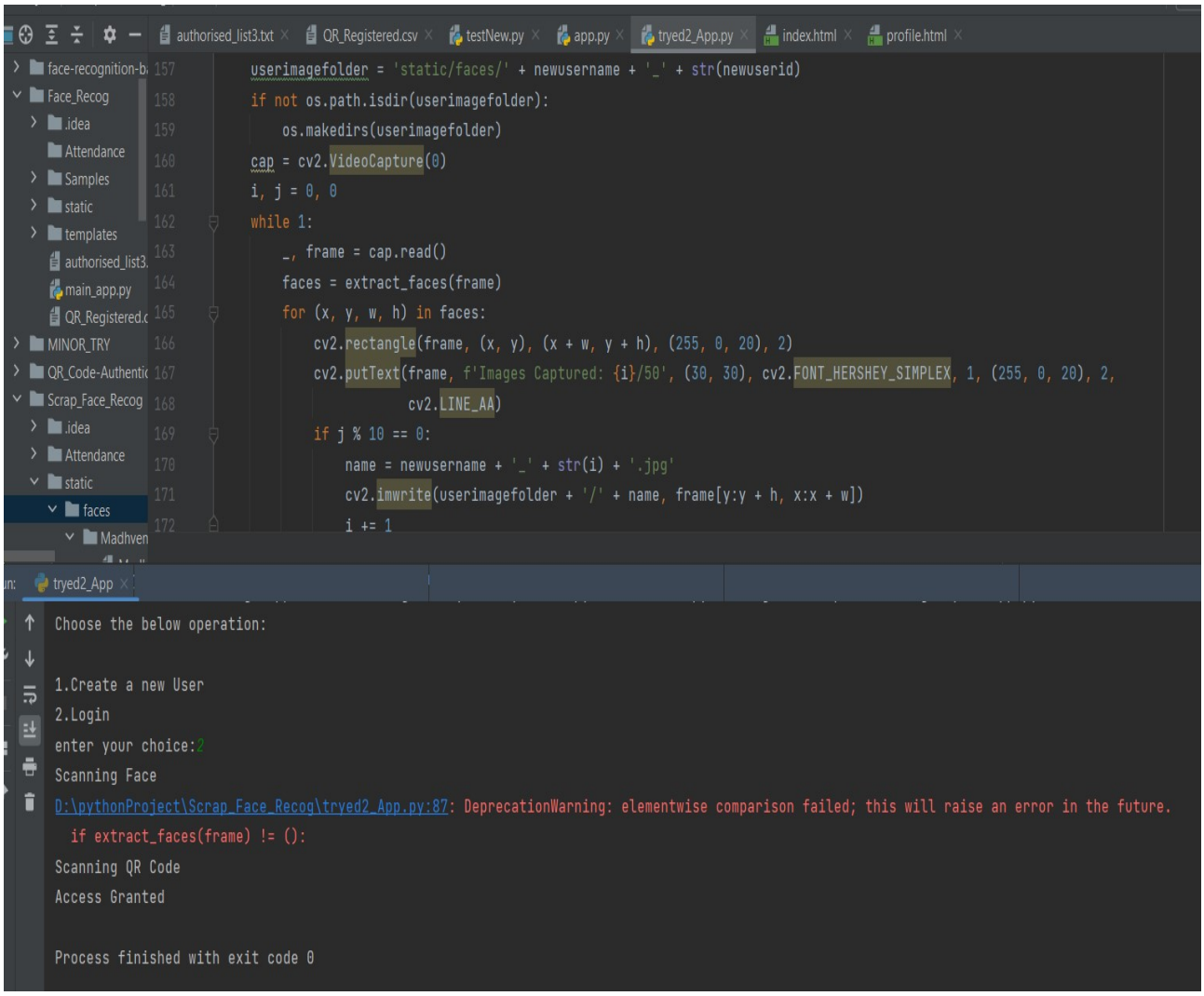


Figure 5.4: **Unit Testing Result**

Figure 5.4 depicts the unit testing result in which each unit is separately tested if there is any error in the units used so that the errors can be eliminated as soon as possible and the project functions properly.

The purpose of unit testing is to test the correctness of isolated code. A unit component is an individual function or code of the application. White box testing approach used for unit testing and usually done by the developers.

5.3.2 Integration testing

Input

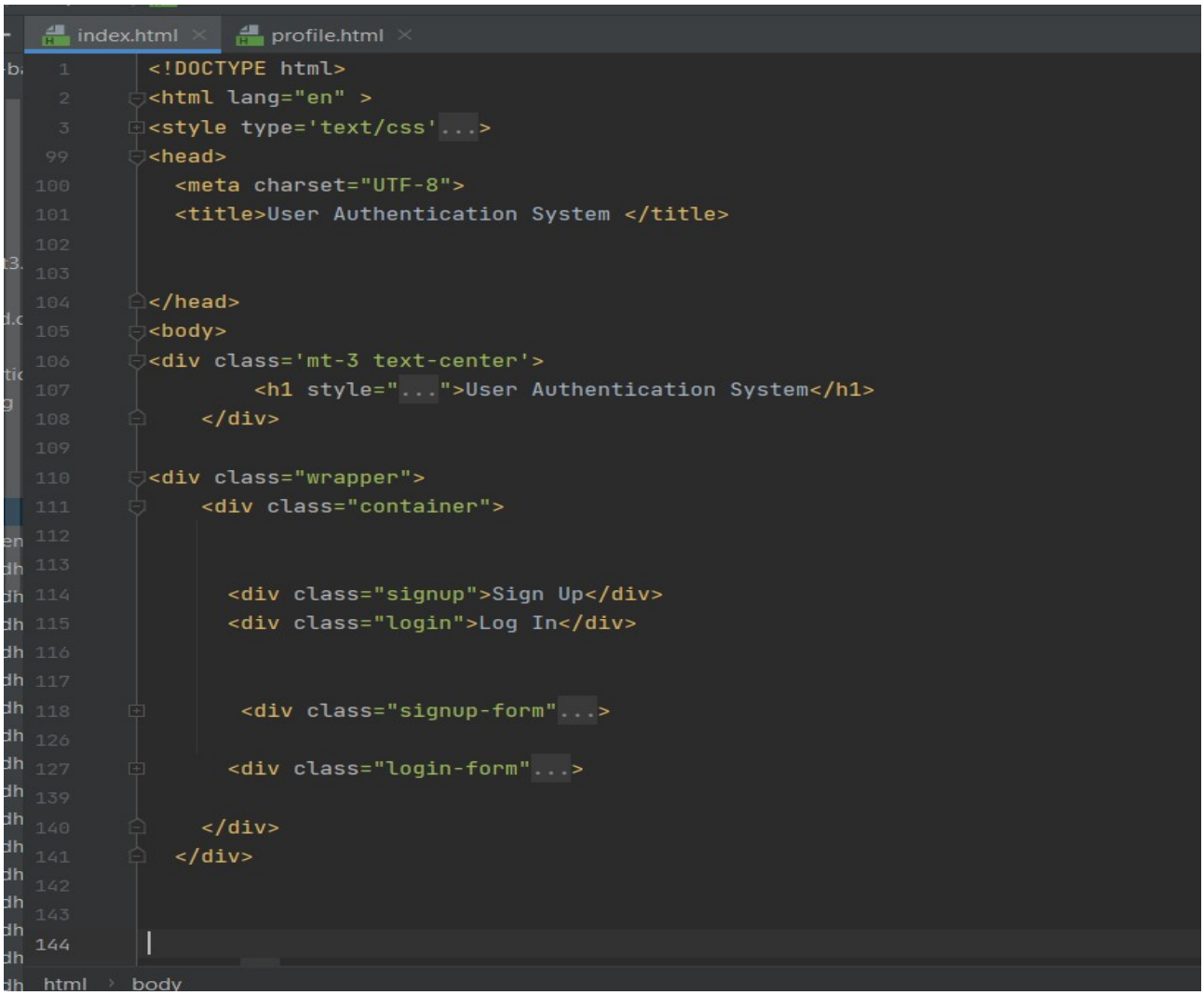


Figure 5.5: **Integration Testing**

Figure 5.5 depicts that inn integration the code was tested for any bugs or compilation errors and then the errors were rectified.Here ,HTML is used for intergration testing. Integration testing is the second level of the software testing process comes after unit testing. In this testing, units or individual components of the software are tested in a group. The focus of the integration testing level is to expose defects at the time of interaction between integrated components or units.

Test result

5.3.3 System testing

Input

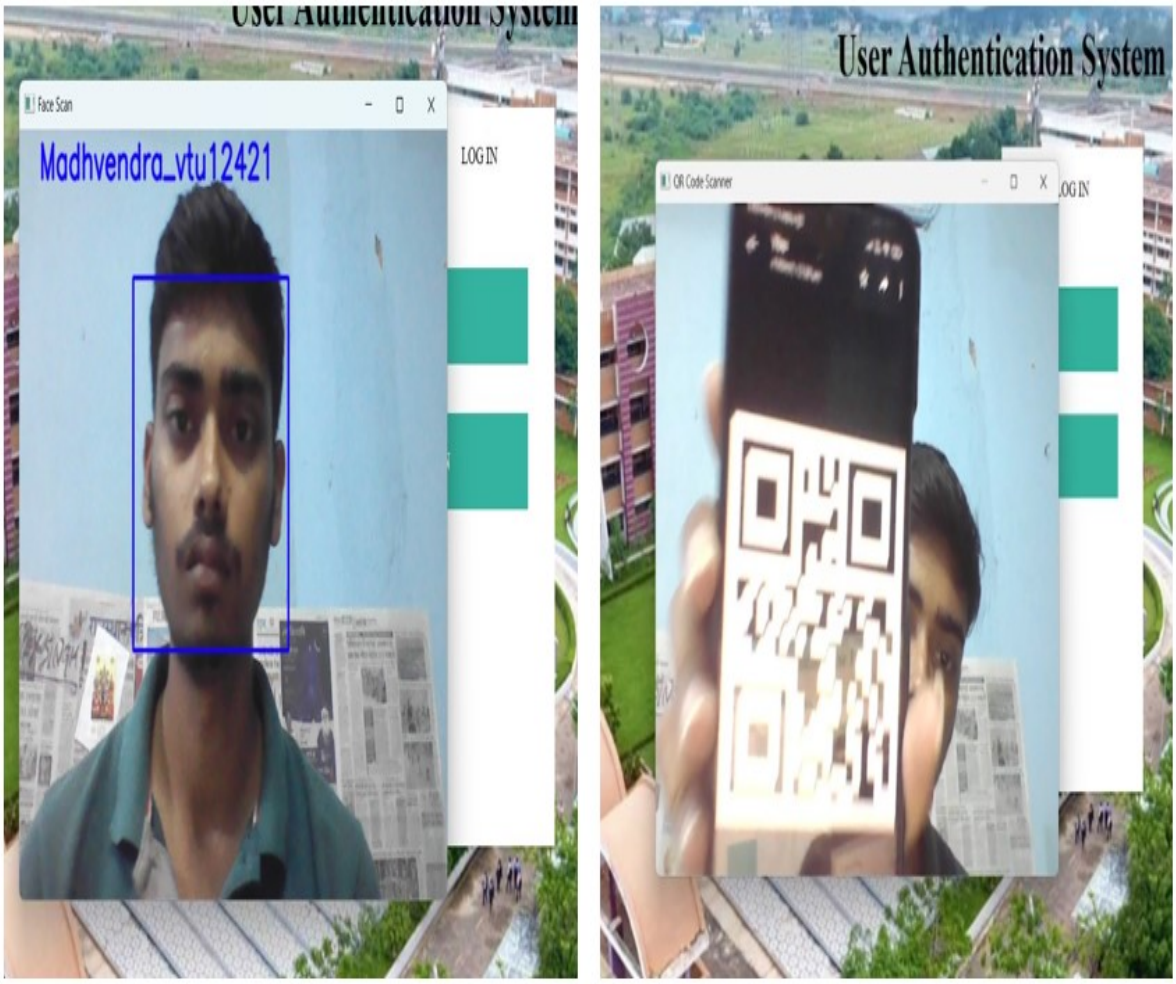


Figure 5.6: **System Testing Input**

Figure 5.6 depicts the input for system testing in which the face and the QR (Quick Response) code are given as input respectively.A QR code is a type of barcode that can be read easily by a digital device and which stores information as a series of pixels in a square-shaped grid. QR codes are frequently used to track information about products in a supply chain and often used in marketing and advertising campaigns.

Test Result

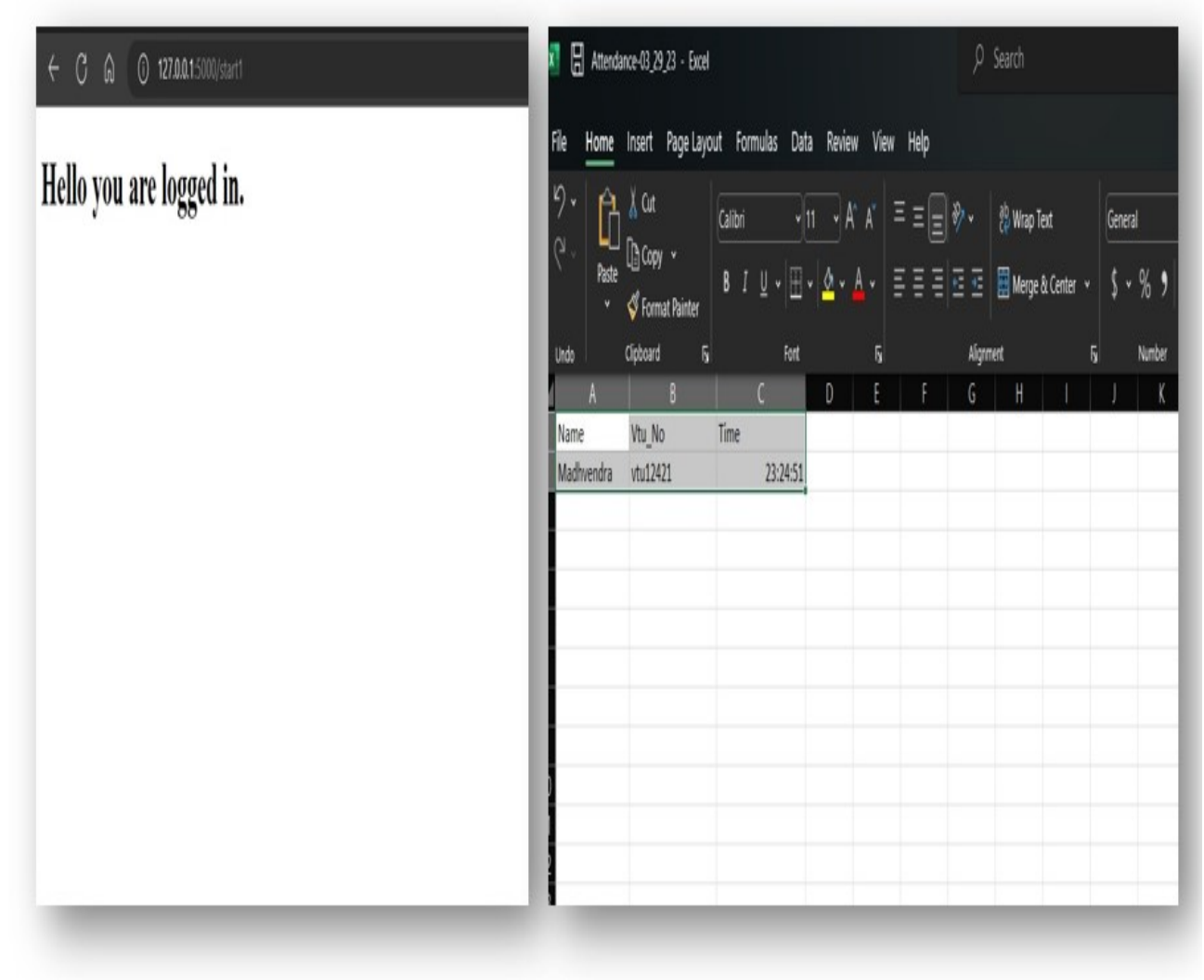


Figure 5.7: **System Testing Result**

Figure 5.7 shows the result of system testing which shows the web page which will be displayed after logging in and it also displays the excel sheet which will keep the user login record.

System Testing includes testing of a fully integrated software system. Generally, a computer system is made with the integration of software (any software is only a single element of a computer system). The software is developed in units and then interfaced with other software and hardware to create a complete computer system

5.3.4 Test Result

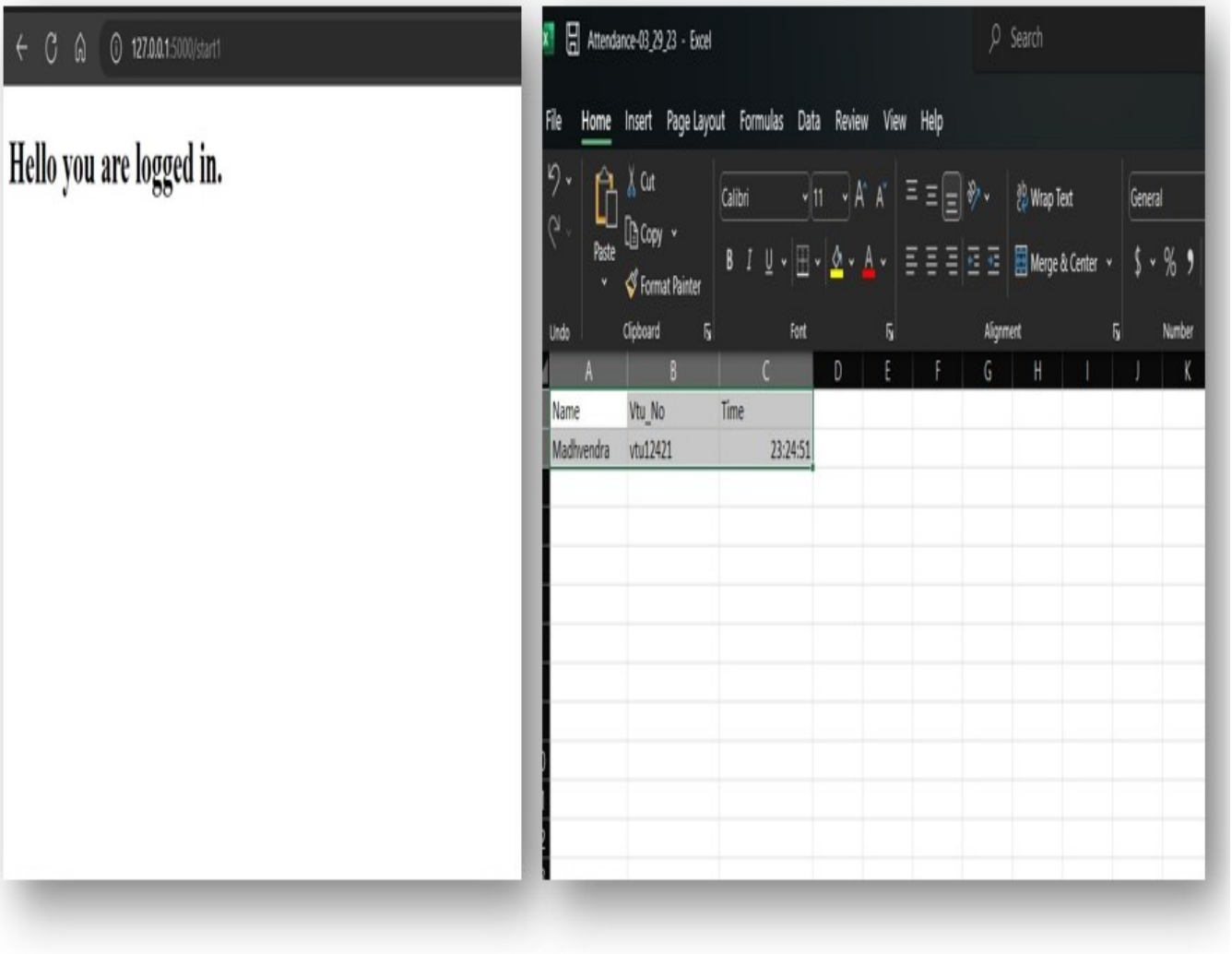


Figure 5.8: **Test Result**

Figure 5.8 shows the test result which shows the web page which will be displayed after logging in and it also displays the excel sheet which will keep the user login record.It shows the message displayed on the web page after user entered the interface.The excel file stores the name of the user ,id number and time of user login.The results evaluates and verifies that a software product or application does what it is supposed to do.The excel sheet will maintain the record of log in and log out of each user which will be helpful at the time of any discrepancy. It will maintain 3 columns name, vtuno, time of login.

Chapter 6

RESULTS AND DISCUSSIONS

6.1 Efficiency of the Proposed System

User authentication systems based on QR code and face recognition will gain popularity as a means to enhance security and usability. These systems combine two-factor authentication methods, QR code and facial recognition, to provide a secure and efficient authentication process. The QR code provides a unique identifier for the user and initiates the authentication process, while the facial recognition technology authenticates the user's identity. The use of facial recognition technology provides a higher level of security compared to traditional authentication methods like passwords or PINs, as it is more difficult to replicate a person's facial features than to guess a password or PIN. Moreover, this technology can be used to prevent identity theft, and it is more convenient for users as they do not need to remember passwords or carry physical tokens. However, these systems must be evaluated based on several factors like accuracy, speed, usability, and security to ensure their effectiveness. Therefore, the design and implementation of user authentication systems based on QR code and face recognition require careful consideration of these factors to ensure their efficiency and effectiveness in providing secure and user-friendly authentication.

6.2 Comparison of Existing and Proposed System

Existing system:(Password based authentication system)
Various existing technologies used in user authentication systems. One of the most commonly used technologies is passwords. Passwords have been used for a long time to authenticate users in various systems, such as email accounts, social media accounts, and online banking. Passwords work by allowing users to create a unique combination of characters, which only they know, and use it to access their accounts. gives less accurate output that is less when compared to proposed system.

Proposed system:(QR code Facial Recognition based system)

User authentication system when compared to the password based authentication,we found nowadays that with the improvement in hacking techniques we need to go for a new method for login which could be biometric or QR based methods.QR code and facial recognition based system is a secure and efficient authentication method that provides an added layer of security to protect user identities. The combination of these two-factor authentication methods ensures that only authorized users can gain access to the system, making it a reliable and user-friendly option for organizations and individuals alike.

6.3 Sample Code

```

main.py

import cv2
import os
from flask import Flask, request, render_template
from datetime import date
from datetime import datetime
import numpy as np
from sklearn.neighbors import KNeighborsClassifier
import pandas as pd
import joblib
from pyzbar.pyzbar import decode

app = Flask(__name__)

datetoday = date.today().strftime("%m_%d_%y")
datetoday2 = date.today().strftime("%d-%B-%Y")



def totalreg():
    return len(os.listdir('static/faces'))


def extract_faces(img):
    gray = cv2.cvtColor(img, cv2.COLOR_BGR2GRAY)
    face_points = face_detector.detectMultiScale(gray, 1.3, 5)
    return face_points

def identify_face(facearray):
    model = joblib.load('static/face_recognition_model.pkl')
    return model.predict(facearray)
```

```
34
35
36  def train_model():
37      faces = []
38      labels = []
39      userlist = os.listdir('static/faces')
40      for user in userlist:
41          for imgname in os.listdir(f'static/faces/{user}'):
42              img = cv2.imread(f'static/faces/{user}/{imgname}')
43              resized_face = cv2.resize(img, (50, 50))
44              faces.append(resized_face.ravel())
45              labels.append(user)
46      faces = np.array(faces)
47      knn = KNeighborsClassifier(n_neighbors=5)
48      knn.fit(faces, labels)
49      joblib.dump(knn, 'static/face_recognition_model.pkl')
50
51
52  def extract_attendance():
53      df = pd.read_csv(f'Attendance/Attendance-{datetoday}.csv')
54      names = df['Name']
55      rolls = df['Vtu_No']
56      times = df['Time']
57      l = len(df)
58      return names, rolls, times, l
59
60
61  def add_attendance(name):
62      username = name.split('_')[0]
63      userid = name.split('_')[1]
64      current_time = datetime.now().strftime("%H:%M:%S")
65
66      df = pd.read_csv(f'Attendance/Attendance-{datetoday}.csv')
67
68  def start():
69      with open('authorised_list3.txt', 'r+') as f:
70          f.truncate(0)
71      cap = cv2.VideoCapture(0)
72      ret = True
73      i=0
74      while ret:
75              ret, frame = cap.read()
76              if extract_faces(frame) != ():
77                  (x, y, w, h) = extract_faces(frame)[0]
78                  cv2.rectangle(frame, (x, y), (x + w, y + h), (255, 0, 20), 2)
79                  face = cv2.resize(frame[y:y + h, x:x + w], (50, 50))
80                  identified_person = identify_face(face.reshape(1, -1))[0]
81                  add_attendance(identified_person)
82                  cv2.putText(frame, f'{identified_person}', (30, 30), cv2.FONT_HERSHEY_SIMPLEX, 1,
                        (255, 0, 20), 2,cv2.LINE_AA)
```

```
                i+=1
            if i==10:
                break
            cv2.imshow('Face Scan', frame)
            if cv2.waitKey(1) == 27:
                break
    cap.release()
    cv2.destroyAllWindows()
    return render_template('index.html')

def barScanner():
    video = cv2.VideoCapture(0)
    video.set(3, 640)
    video.set(4, 740)

    with open('authorised_list3.txt', 'r') as file:
        authorised_list = file.read().strip()
        # print(authorised_list)
    i=0
    while True:
        success, image = video.read()
        for barcode in decode(image):
            qr_text = barcode.data.decode('utf-8')
            qr_text = str(qr_text).lower()
            if qr_text not in authorised_list:
                color = (0, 0, 255)
                display_message = "Denied Access"
                print("Access Denied")


            else:
                color = (0, 255, 0)
                display_message = "Access Granted"
                print("Access Granted")

                with open('QR_Registered.csv', 'r+') as f:
                    myDataList = f.readlines()
                    nameList = []
                    for line in myDataList:
                        entry = line.split(',')
                        nameList.append(entry[0])
                    if qr_text not in nameList:
                        now = datetime.now()
                        dtString = now.strftime('%H:%M:%S')
                        dtString1 = date.today()
                        f.writelines(f'\n{qr_text},{dtString1},{dtString}')
                return render_template('profile.html')
            polygon_points = np.array([barcode.polygon], np.int32)
            polygon_points = polygon_points.reshape(-1, 1, 2)
            rect_points = barcode.rect
```

```
            cv2.polylines(image, [polygon_points], True, color, 3)
            cv2.putText(image, display_message, (rect_points[0], rect_points[1]), cv2.
                FONT_HERSHEY_PLAIN, 0.9, color, 2)
            i+=1
        if i==1:
            break
        cv2.imshow("QR Code Scanner", image)
        if cv2.waitKey(1) == 27:
            break
    video.release()
    cv2.destroyAllWindows()
    return render_template('index.html')


def add():
    newusername = request.form['newusername']
    newuserid = request.form['newuserid']
    userimagefolder = 'static/faces/' + newusername + '_' + str(newuserid)
    cap = cv2.VideoCapture(0)
    i, j = 0, 0
    while 1:
        _, frame = cap.read()
        faces = extract_faces(frame)
        for (x, y, w, h) in faces:
            cv2.rectangle(frame, (x, y), (x + w, y + h), (255, 0, 20), 2)
            cv2.putText(frame, f'Images Captured: {i}/50', (30, 30), cv2.FONT_HERSHEY_SIMPLEX, 1,
                (255, 0, 20), 2,
                        cv2.LINE_AA)
            if j % 10 == 0:
                name = newusername + '_' + str(i) + '.jpg'
                cv2.imwrite(userimagefolder + '/' + name, frame[y:y + h, x:x + w])
                i += 1
            j += 1
        if j == 500:
            break
        cv2.imshow('Adding new User', frame)
        if cv2.waitKey(1) == 27:
            break
    cap.release()
    cv2.destroyAllWindows()
    print('Training Model')
    train_model()

    return render_template('index.html')



if __name__ == '__main__':
    app.run(debug=True)
```

```
181
182 ********************************************************************************************************************
183 index.html
184
185 <!DOCTYPE html>
186 <html lang="en" >
187 <style type='text/css'>
188     *{
189   margin: 0;
190   padding: 0;
191 }
192
193 body{
194    background-image: url('https://media-exp1.licdn.com/dms/image/C511BAQHgKtftzO34CA/company-
          background_10000/0/1582269725600?e=2159024400&v=beta&t=
          G9T29odiZGewlXhj2qnHHJjDrpkuI9Kd4lqYuqSpmvA');
195    background-size: cover;
196
197   font-family: roboto;
198   user-select: none;
199 }
200
201 .container{
202   width: 450px;
203   margin: 30px auto;
204 }
205
206 .signup,
207 .login{
208   width: 50%;
209   background: #fff;
210   float: left;
211   height: 60px;
212   line-height: 60px;
213   text-align: center;
214   cursor: pointer;
215   text-transform: uppercase;
216 }
217
218 .signup-form,
219 .login-form{
220   background: #fff;
221   padding: 40px;
222   clear: both;
223   width: 100%;
224   box-sizing: border-box;
225   height: 400px;
226 }
227 .input{
```

```
228     width: 100%;
229     padding: 20px;
230     box-sizing: border-box;
231     margin-bottom: 25px;
232     border: 2px solid #e9eaea;
233     color: #3e3e40;
234     font-size: 14px;
235     outline: none;
236     transform: all 0.5s ease;
237 }
238 
239 .input:focus{
240     border: 2px solid #34b3a0;
241 }
242 .btn{
243     width: 100%;
244     background: #34b3a0;
245     height: 60px;
246     text-align: center;
247     line-height: 60px;
248     text-transform: uppercase;
249     color: #fff;
250     font-weight: bold;
251     letter-spacing: 1px;
252     cursor: pointer;
253     margin-bottom: 30px;
254 }
255 
256 
257 
258 
259 span a{
260     text-decoration: none;
261     color: #000;
262 }
263 
264 ::-webkit-input-placeholder { /* Chrome/Opera/Safari */
265     color: #3e3e40;
266     font-family: roboto;
267 }
268 ::-moz-placeholder { /* Firefox 19+ */
269     color: #3e3e40;
270     font-family: roboto;
271 }
272 :-ms-input-placeholder { /* IE 10+ */
273     color: #3e3e40;
274     font-family: roboto;
275 }
276 :-moz-placeholder { /* Firefox 18- */
277     color: #3e3e40;
```

```
  font-family: roboto;
}


</style>
<head>
  <meta charset="UTF-8">
  <title>User Authentication System </title>


</head>
<body>
<div class='mt-3 text-center'>
        <h1 style="width: auto;margin: auto;text-align: center;color: black;padding: 11px;font-size:
            44px;">User Authentication System</h1>
    </div>

<div class="wrapper">
    <div class="container">


      <div class="signup">Sign Up</div>
      <div class="login">Log In</div>


       <div class="signup-form">
           <form action='/add' method="POST" >
                <input type="text" id="newusername" name='newusername' placeholder="Username" style=
                    "width: 100%;padding: 20px;box-sizing: border-box;margin-bottom: 25px;border: 2
                    px solid #e9eaea;color: #3e3e40;font-size: 14px;outline: none;transform: all 0.5
                    s ease;" required><br>
                <input type="text" id="newusereid" name='newuserid' placeholder="userID" style="
                    width: 100%;padding: 20px;box-sizing: border-box;margin-bottom: 25px;border: 2px
                     solid #e9eaea;color: #3e3e40;font-size: 14px;outline: none;transform: all 0.5s
                    ease;" required><br>
                <button style="width: 100%;background: #34b3a0;height: 60px;text-align: center;
                    line-height: 60px;text-transform: uppercase;color: #fff;font-weight: bold;letter
                    -spacing: 1px;cursor: pointer;margin-bottom: 30px;" type='submit' class='btn'>
                    Create Account
                </button>
           </form>
       </div>

      <div class="login-form">
           <a style="text-decoration: none;max-width: 300px;" href="/start">
          <div class="btn">Face Scan</div>
           </a>
          <a style="text-decoration: none;max-width: 300px;" href="/start1">
              <div class="btn" >BarCode Scan</div>

```

```
            </a>


        </div>

        </div>
    </div>


</body>
</html>

*****************************************************************************************************
profile.html

<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <title></title>
</head>
<body>
<h1>Hello you are logged in.</h1>

</body>
</html>
```

Output

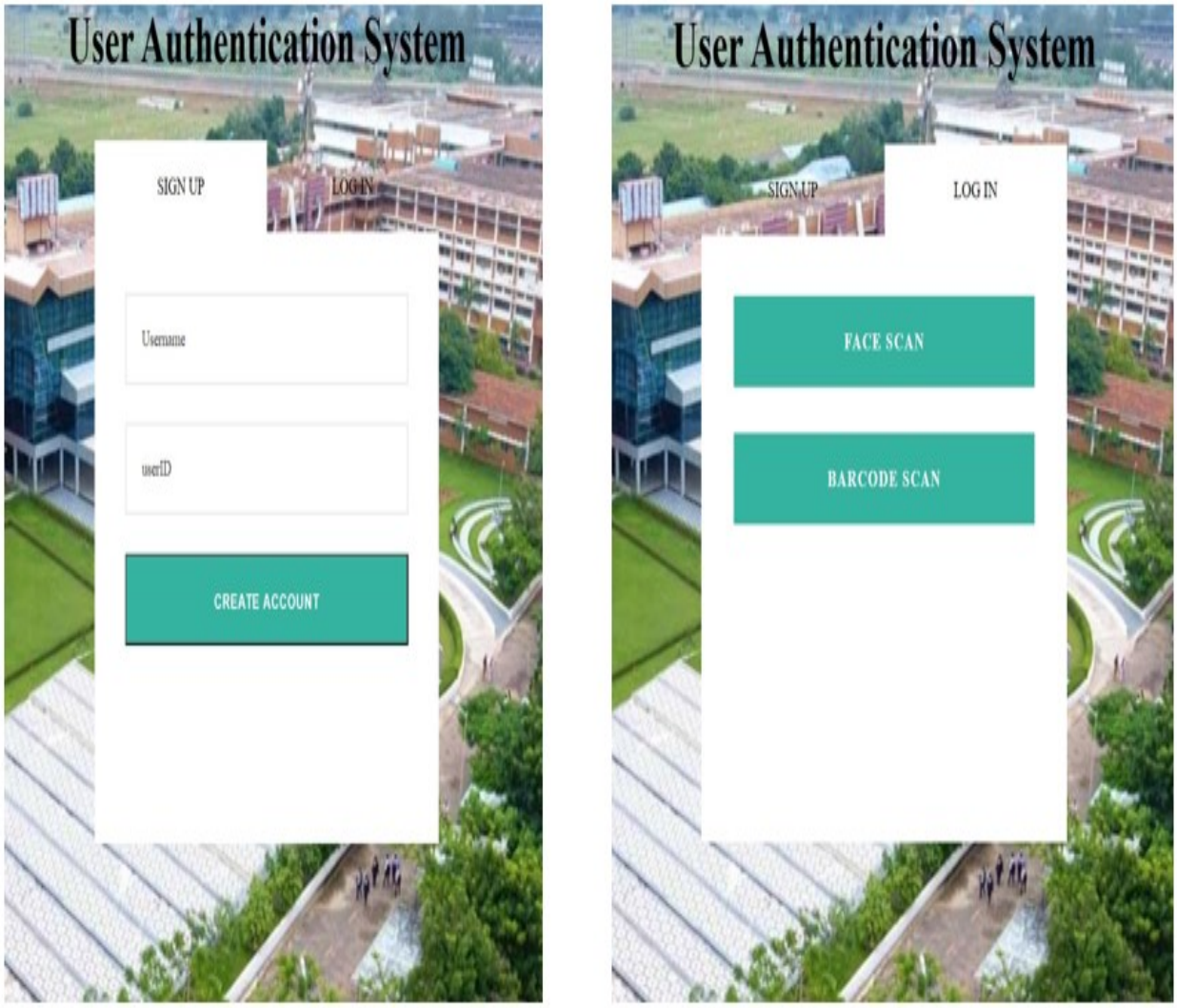


Figure 6.1: **Input Design**

Figure 6.1 shows the interface of the User Authentication system, first image shows the login page and second image shows sign in page where we have to register the face and QR code. The first image is the sign up page interface where the user will have to set the username and password which will be stored in the database. The second image shows the interface page where the user will have to do face scan and the QR code authentication.

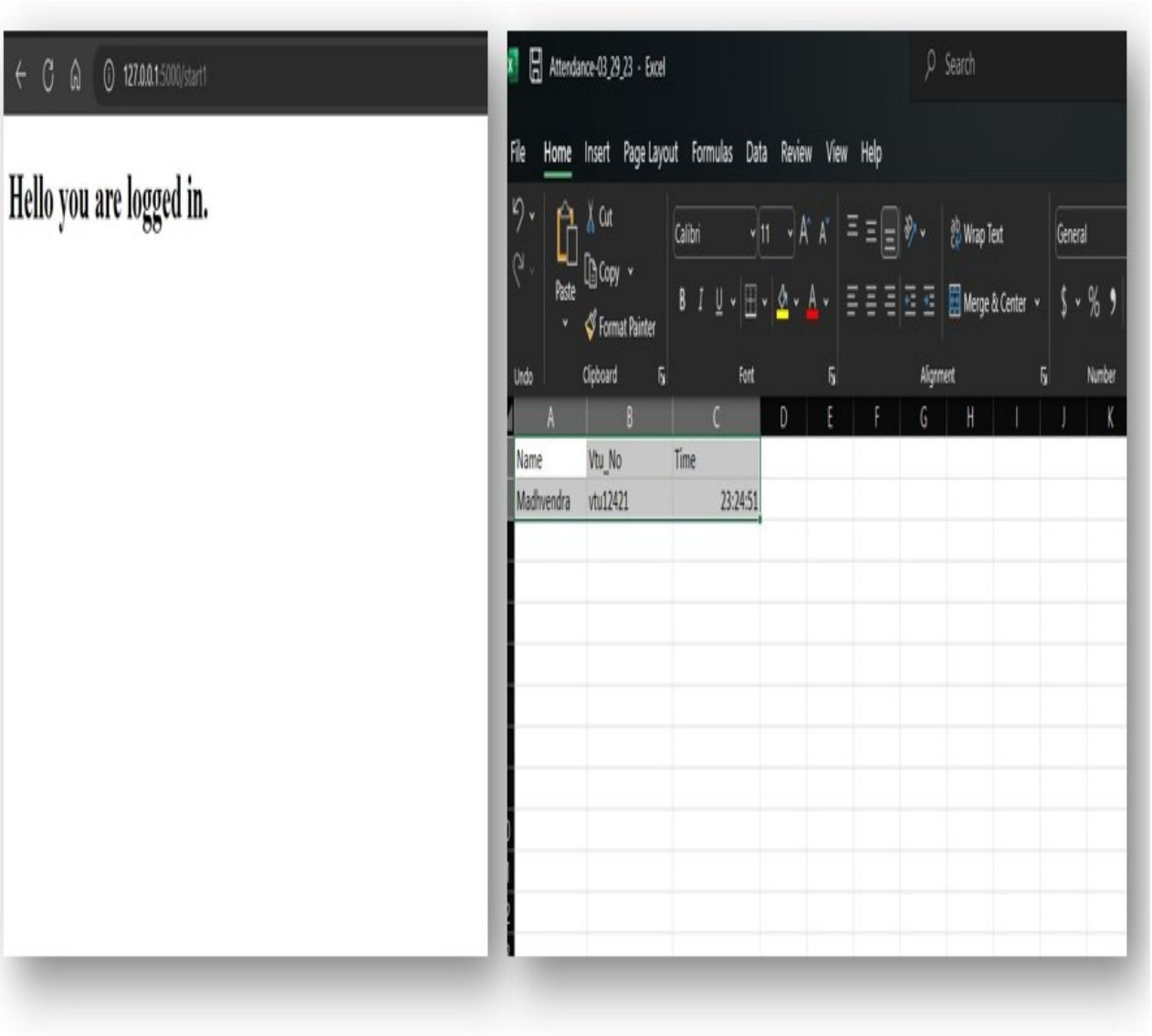


Figure 6.2: **Result System Testing**

Figure 6.2 shows the test result which shows the page which will be displayed after logging in and it also displays the excel sheet which will keep the user login record.t shows the message displayed on the web page after user entered the interface. The excel sheet will maintain the record of log in and log out of each user which will be helpful at the time of any discrepancy. It will maintain 3 columns name, vtuno, time of login.

Chapter 7

CONCLUSION AND FUTURE ENHANCEMENTS

7.1 Conclusion

Development of a user authentication system using face detection and QR code authentication is a powerful tool that can enhance user login system and improve security in a variety of settings. By leveraging the latest advancements in computer vision and machine learning, this system authenticate user through face recognition and QR code verification. The system has the potential to improve efficiency and accuracy, reduce errors, and enhance security in various industries, including education, e-commerce, banking and public spaces.The advantages of this system include increased security, faster authentication process, and ease of use. It eliminates the need for passwords or other forms of authentication that can be easily stolen or hacked. The system also reduces the risk of fraud and identity theft.

It includes face detection and barcode/QR code scaning Giving input's and generating the required output. The process starts with sign up using username and password, In the next step the user will register face then the credentials will be validated. Then the user can login by going through face detection and then through QR code authentication. If the face is not detected then it will time out. If the QR code matches, authenticate the user and grant access. If the QR code does not match, prompt the user to try again or contact the system administrator for assistance. End the authentication process. 17

The User authentication system using face detection and QR code authentication project is a secure and reliable way to authenticate users. The system combines two methods of authentication - face detection and QR code authentication to verify the identity of the user. The system can be implemented in various applications such

as online banking, e-commerce, and other secure platforms that require user authentication. Overall, this system provides an efficient and secure way to authenticate users and protect sensitive information. Furthermore, as technology continues to evolve, this system has the potential to become even more advanced, with improved algorithms, higher accuracy rates, and enhanced features. However, it is essential to consider the ethical and privacy implications of using facial recognition technology and ensure that appropriate measures are taken to protect individuals' rights.

7.2 Future Enhancements

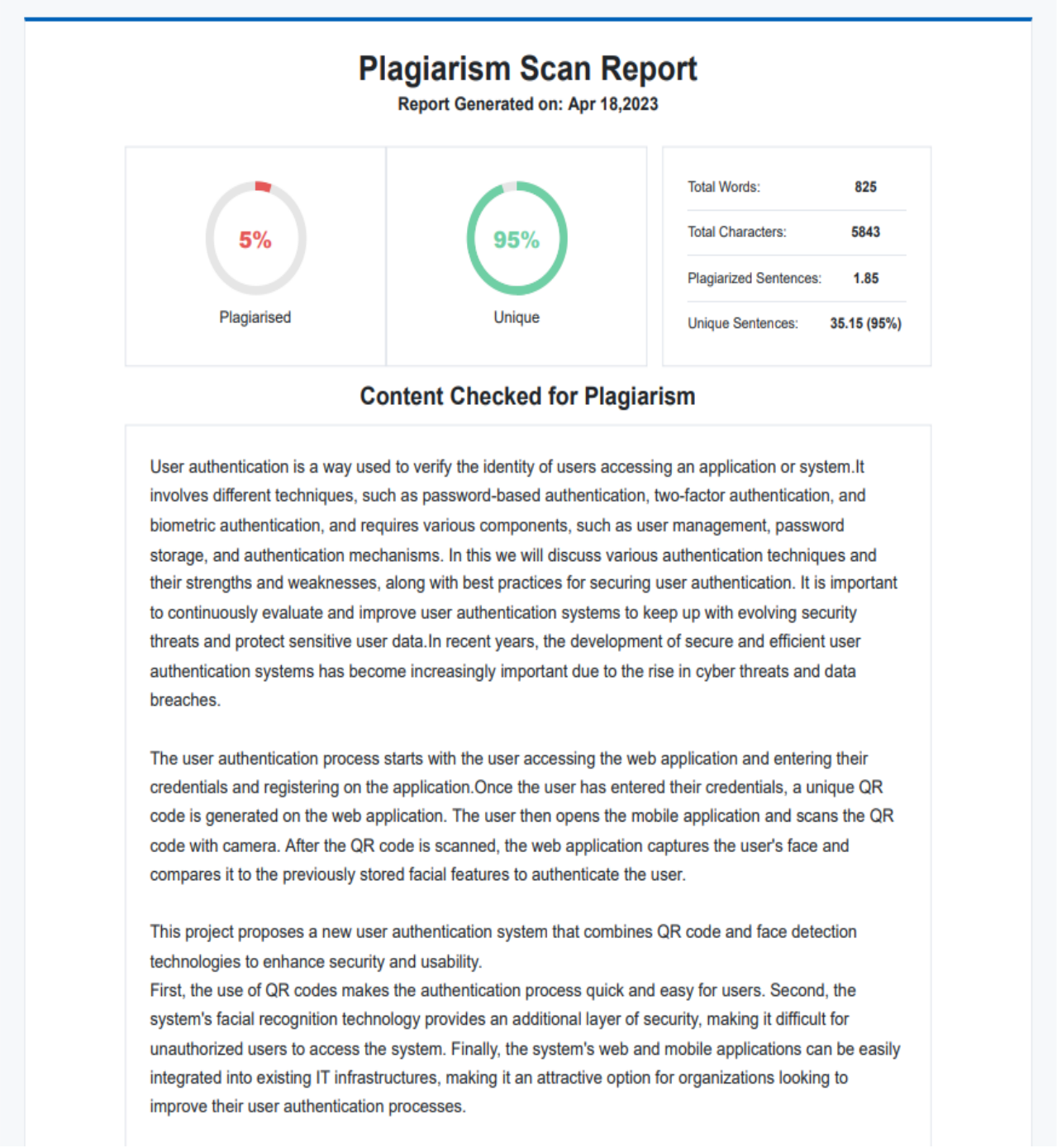
There are several potential future enhancements that could be made to the user authentication system developed using QR code and face detection technology. One potential area for improvement is in the accuracy and reliability of the face detection algorithm. As computer vision technology continues to advance, there may be new and more effective algorithms that could be used to improve the accuracy of the system. Additionally, the system could be further refined to detect and reject attempts at spoofing or falsifying the facial recognition process.

Multi-factor authentication ,in addition to using QR codes and face detection, other factors such as fingerprint scanning or voice recognition could be added to further strengthen the authentication process. It will be enhanced to provide real-time monitoring of user authentication attempts to detect potential security breaches. It can be integrated with a mobile app, allowing users to easily authenticate themselves using their smartphones.It will be used to store user authentication data in the cloud, providing greater accessibility and flexibility.The machine learning algorithms can be used to continuously improve its accuracy and performance over time.

Overall, the development of the user authentication system using QR code and face detection is an exciting advancement in the field of computer security. With ongoing improvements and enhancements, this technology has the potential to become a widespread and effective method for verifying the identities of users across a wide range of applications and industries.

Chapter 8

PLAGIARISM REPORT



Plagiarism Scan Report

Report Generated on: Apr 18,2023

5%
Plagiarised

95%
Unique

Total Words:	825
Total Characters:	5843
Plagiarized Sentences:	1.85
Unique Sentences:	35.15 (95%)

Content Checked for Plagiarism

User authentication is a way used to verify the identity of users accessing an application or system.It involves different techniques, such as password-based authentication, two-factor authentication, and biometric authentication, and requires various components, such as user management, password storage, and authentication mechanisms. In this we will discuss various authentication techniques and their strengths and weaknesses, along with best practices for securing user authentication. It is important to continuously evaluate and improve user authentication systems to keep up with evolving security threats and protect sensitive user data.In recent years, the development of secure and efficient user authentication systems has become increasingly important due to the rise in cyber threats and data breaches.

The user authentication process starts with the user accessing the web application and entering their credentials and registering on the application.Once the user has entered their credentials, a unique QR code is generated on the web application. The user then opens the mobile application and scans the QR code with camera. After the QR code is scanned, the web application captures the user's face and compares it to the previously stored facial features to authenticate the user.

This project proposes a new user authentication system that combines QR code and face detection technologies to enhance security and usability.
First, the use of QR codes makes the authentication process quick and easy for users. Second, the system's facial recognition technology provides an additional layer of security, making it difficult for unauthorized users to access the system. Finally, the system's web and mobile applications can be easily integrated into existing IT infrastructures, making it an attractive option for organizations looking to improve their user authentication processes.

Figure 8.1: **Plagiarism report**

Chapter 9

SOURCE CODE & POSTER PRESENTATION

9.1 Source Code

```
main.py

import cv2
import os
from flask import Flask, request, render_template
from datetime import date
from datetime import datetime
import numpy as np
from sklearn.neighbors import KNeighborsClassifier
import pandas as pd
import joblib
from pyzbar.pyzbar import decode

app = Flask(__name__)

datetoday = date.today().strftime("%m_%d_%y")
datetoday2 = date.today().strftime("%d-%B-%Y")



def totalreg():
    return len(os.listdir('static/faces'))


def extract_faces(img):
    gray = cv2.cvtColor(img, cv2.COLOR_BGR2GRAY)
    face_points = face_detector.detectMultiScale(gray, 1.3, 5)
    return face_points

def identify_face(facearray):
    model = joblib.load('static/face_recognition_model.pkl')
    return model.predict(facearray)


```

```
def train_model():
    faces = []
    labels = []
    userlist = os.listdir('static/faces')
    for user in userlist:
        for imgname in os.listdir(f'static/faces/{user}'):
            img = cv2.imread(f'static/faces/{user}/{imgname}')
            resized_face = cv2.resize(img, (50, 50))
            faces.append(resized_face.ravel())
            labels.append(user)
    faces = np.array(faces)
    knn = KNeighborsClassifier(n_neighbors=5)
    knn.fit(faces, labels)
    joblib.dump(knn, 'static/face_recognition_model.pkl')


def extract_attendance():
    df = pd.read_csv(f'Attendance/Attendance-{datetoday}.csv')
    names = df['Name']
    rolls = df['Vtu_No']
    times = df['Time']
    l = len(df)
    return names, rolls, times, l


def add_attendance(name):
    username = name.split('_')[0]
    userid = name.split('_')[1]
    current_time = datetime.now().strftime("%H:%M:%S")

    df = pd.read_csv(f'Attendance/Attendance-{datetoday}.csv')

def start():
    with open('authorised_list3.txt', 'r+') as f:
        f.truncate(0)
    cap = cv2.VideoCapture(0)
    ret = True
    i=0
    while ret:
            ret, frame = cap.read()
            if extract_faces(frame) != ():
                (x, y, w, h) = extract_faces(frame)[0]
                cv2.rectangle(frame, (x, y), (x + w, y + h), (255, 0, 20), 2)
                face = cv2.resize(frame[y:y + h, x:x + w], (50, 50))
                identified_person = identify_face(face.reshape(1, -1))[0]
                add_attendance(identified_person)
                cv2.putText(frame, f'{identified_person}', (30, 30), cv2.FONT_HERSHEY_SIMPLEX, 1,
                    (255, 0, 20), 2,cv2.LINE_AA)
                i+=1
            if i==10:
```

```
                break
            cv2.imshow('Face Scan', frame)
            if cv2.waitKey(1) == 27:
                break
    cap.release()
    cv2.destroyAllWindows()
    return render_template('index.html')

def barScanner():
    video = cv2.VideoCapture(0)
    video.set(3, 640)
    video.set(4, 740)

    with open('authorised_list3.txt', 'r') as file:
        authorised_list = file.read().strip()
        # print(authorised_list)
    i=0
    while True:
        success, image = video.read()
        for barcode in decode(image):
            qr_text = barcode.data.decode('utf-8')
            qr_text = str(qr_text).lower()
            if qr_text not in authorised_list:
                color = (0, 0, 255)
                display_message = "Denied Access"
                print("Access Denied")


            else:
                color = (0, 255, 0)
                display_message = "Access Granted"
                print("Access Granted")

                with open('QR_Registered.csv', 'r+') as f:
                    myDataList = f.readlines()
                    nameList = []
                    for line in myDataList:
                        entry = line.split(',')
                        nameList.append(entry[0])
                    if qr_text not in nameList:
                        now = datetime.now()
                        dtString = now.strftime('%H:%M:%S')
                        dtString1 = date.today()
                        f.writelines(f'\n{qr_text},{dtString1},{dtString}')
                return render_template('profile.html')
            polygon_points = np.array([barcode.polygon], np.int32)
            polygon_points = polygon_points.reshape(-1, 1, 2)
            rect_points = barcode.rect
            cv2.polylines(image, [polygon_points], True, color, 3)
```

```
134             cv2.putText(image, display_message, (rect_points[0], rect_points[1]), cv2.
                    FONT_HERSHEY_PLAIN, 0.9, color, 2)
135             i+=1
136         if i==1:
137             break
138         cv2.imshow("QR Code Scanner", image)
139         if cv2.waitKey(1) == 27:
140             break
141     video.release()
142     cv2.destroyAllWindows()
143     return render_template('index.html')
144
145
146 def add():
147     newusername = request.form['newusername']
148     newuserid = request.form['newuserid']
149     userimagefolder = 'static/faces/' + newusername + '_' + str(newuserid)
150     cap = cv2.VideoCapture(0)
151     i, j = 0, 0
152     while 1:
153         _, frame = cap.read()
154         faces = extract_faces(frame)
155         for (x, y, w, h) in faces:
156             cv2.rectangle(frame, (x, y), (x + w, y + h), (255, 0, 20), 2)
157             cv2.putText(frame, f'Images Captured: {i}/50', (30, 30), cv2.FONT_HERSHEY_SIMPLEX, 1,
                    (255, 0, 20), 2,
158                         cv2.LINE_AA)
159             if j % 10 == 0:
160                 name = newusername + '_' + str(i) + '.jpg'
161                 cv2.imwrite(userimagefolder + '/' + name, frame[y:y + h, x:x + w])
162                 i += 1
163             j += 1
164         if j == 500:
165             break
166         cv2.imshow('Adding new User', frame)
167         if cv2.waitKey(1) == 27:
168             break
169     cap.release()
170     cv2.destroyAllWindows()
171     print('Training Model')
172     train_model()
173
174     return render_template('index.html')
175
176
177
178
179 if __name__ == '__main__':
180     app.run(debug=True)
181
```

```
182 ********************************************************************************************************

183 index.html

184

185 <!DOCTYPE html>
186 <html lang="en" >
187 <style type='text/css'>
188     *{
189   margin: 0;
190   padding: 0;
191 }

192

193 body{
194    background-image: url('https://media-exp1.licdn.com/dms/image/C511BAQHgKtftzO34CA/company-
          background_10000/0/1582269725600?e=2159024400&v=beta&t=
          G9T29odiZGewlXhj2qnHHJjDrpkuI9Kd4lqYuqSpmvA');
195    background-size: cover;

196

197   font-family: roboto;
198   user-select: none;
199 }

200

201 .container{
202   width: 450px;
203   margin: 30px auto;
204 }

205

206 .signup,
207 .login{
208   width: 50%;
209   background: #fff;
210   float: left;
211   height: 60px;
212   line-height: 60px;
213   text-align: center;
214   cursor: pointer;
215   text-transform: uppercase;
216 }

217

218 .signup-form,
219 .login-form{
220   background: #fff;
221   padding: 40px;
222   clear: both;
223   width: 100%;
224   box-sizing: border-box;
225   height: 400px;
226 }
227 .input{
228   width: 100%;
```

```
    padding: 20px;
    box-sizing: border-box;
    margin-bottom: 25px;
    border: 2px solid #e9eaea;
    color: #3e3e40;
    font-size: 14px;
    outline: none;
    transform: all 0.5s ease;
}

.input:focus{
    border: 2px solid #34b3a0;
}
.btn{
    width: 100%;
    background: #34b3a0;
    height: 60px;
    text-align: center;
    line-height: 60px;
    text-transform: uppercase;
    color: #fff;
    font-weight: bold;
    letter-spacing: 1px;
    cursor: pointer;
    margin-bottom: 30px;
}




span a{
    text-decoration: none;
    color: #000;
}

::-webkit-input-placeholder { /* Chrome/Opera/Safari */
    color: #3e3e40;
    font-family: roboto;
}
::-moz-placeholder { /* Firefox 19+ */
    color: #3e3e40;
    font-family: roboto;
}
:-ms-input-placeholder { /* IE 10+ */
    color: #3e3e40;
    font-family: roboto;
}
:-moz-placeholder { /* Firefox 18- */
    color: #3e3e40;
    font-family: roboto;
```

```
}


</style>
<head>
  <meta charset="UTF-8">
  <title>User Authentication System </title>


</head>
<body>
<div class='mt-3 text-center'>
        <h1 style="width: auto;margin: auto;text-align: center;color: black;padding: 11px;font-size:
             44px;">User Authentication System</h1>
    </div>

<div class="wrapper">
    <div class="container">


      <div class="signup">Sign Up</div>
      <div class="login">Log In</div>


       <div class="signup-form">
           <form action='/add' method="POST" >
                <input type="text" id="newusername" name='newusername' placeholder="Username" style=
                    "width: 100%;padding: 20px;box-sizing: border-box;margin-bottom: 25px;border: 2
                    px solid #e9eaea;color: #3e3e40;font-size: 14px;outline: none;transform: all 0.5
                    s ease;" required><br>
                <input type="text" id="newusereid" name='newuserid' placeholder="userID" style="
                    width: 100%;padding: 20px;box-sizing: border-box;margin-bottom: 25px;border: 2px
                     solid #e9eaea;color: #3e3e40;font-size: 14px;outline: none;transform: all 0.5s
                    ease;" required><br>
                <button style="width: 100%;background: #34b3a0;height: 60px;text-align: center;
                    line-height: 60px;text-transform: uppercase;color: #fff;font-weight: bold;letter
                    -spacing: 1px;cursor: pointer;margin-bottom: 30px;" type='submit' class='btn'>
                    Create Account
                </button>
           </form>
       </div>

      <div class="login-form">
           <a style="text-decoration: none;max-width: 300px;" href="/start">
          <div class="btn">Face Scan</div>
           </a>
          <a style="text-decoration: none;max-width: 300px;" href="/start1">
              <div class="btn" >BarCode Scan</div>

          </a>
```

```
319
320
321
322         </div>
323
324         </div>
325       </div>
326
327
328
329 </body>
330 </html>
331
332 ********************************************************************************************************
333 profile.html
334
335 <!DOCTYPE html>
336 <html lang="en">
337 <head>
338     <meta charset="UTF-8">
339     <title></title>
340 </head>
341 <body>
342 <h1>Hello you are logged in.</h1>
343
344 </body>
345 </html>
```

USER AUTHENTICATION SYSTEM

Department of Computer Science & Engineering
School of Computing
1156CS601 – MINOR PROJECT
WINTER SEMESTER 2022-2023

ABSTRACT

User authentication system is a way used to verify the identity of user accessing an application or system. It involves different techniques such as password based authentication, two factor authentication, and biometric authentication and requires various components, such as user management, password storage, and authentication mechanisms. A user authentication system provides a way for users to securely authenticate themselves to a computer system or network, ensuring that only authorized users can access sensitive data or perform critical operations. In addition, user authentication systems help organizations comply with regulatory requirements and protect against cyber attacks, such as identity theft, phishing, and credential stuffing.

It is important to continuously evaluate and improve user authentication systems to keep up with evolving security threats and protect sensitive user data.

TEAM MEMBER DETAILS

<Student 1. VTU12418/Yash Aggarawal>
<Student2.VTU12421/Madhvendra Singh>
<Student 3 .VTU 17717/Ashish Sonkaria>
<Student 1. 83840 51074>
<Student 2. 6375116283 >
<Student 3. 6280362085>
<Student 1. vtu12418@veltech.edu.in>
<Student 2. vtu12421@veltech.edu.in>
<Student 3.vtu17717@veltech.edu.in>

INTRODUCTION

User authentication is the process of verifying the identity of a user who wants to access an application or system. The goal of a user authentication system is to ensure that only authorized users are granted access while preventing unauthorized access to sensitive data and resources.Different authentication techniques, such as password-based authentication,two-factor authentication and biometric authentication, can be used in a user authentication system. The aim of this project is to develop a user authentication system using QR code and face detection. Authentication is a crucial aspect of security, and it ensures that only authorized users can access a system or data. Traditional methods of authentication, such as passwords and PINs, can be vulnerable to hacking and phishing attacks. To address this issue, biometric authentication techniques, such as face recognition, have been increasingly used in recent years. However, face recognition systems can still face challenges, such as changes in lighting conditions and facial expressions.

A user authentication system should be robust, reliable, and easy to use to ensure the security and integrity of modern applications and systems. User authentication systems can help prevent various types of attacks, such as brute-force attacks, password guessing attacks, and credential stuffing attacks. User authentication systems should be periodically audited and tested for vulnerabilities to ensure that they remain secure and effective. The proposed system has several potential benefits, including increased security, convenience, and accessibility. By combining QR codes and face detection , users can authenticate themselves quickly and securely, without the need for passwords or PINs. The system can be implemented in various applications, such as online banking, e-commerce, and social media platforms, to enhance user security and privacy.

METHODOLOGIES

There are several authentication methods to choose from, including passwords, biometric authentication (such as fingerprint or facial recognition), two-factor authentication, and multi-factor authentication.

In this project we are using two-factor authentication, i.e, detecting and recognizing the user through webcam and then authenticating the user by its QR or Bar Code of his/her ID card.

The whole process is divided into two parts ,i.e:

MODULE 1: Detection and Recognition

MODULE 2:Authentication through QR Code.

MODULE 3: Execution Testing

RESULTS

In the proposed system , we have three module . First one is for the detection and recognition .The second is for the authentication through QR code. The concerned user image was uploaded from the database and scanned image was compared with the extracted features of face in module 1 .The third one is for final testing and execution of the system.

In the module 2 , the scanned QR code was decoded and access was granted if the detail matched in the csv file otherwise it was denied .All the data was stored in the database for future references.

In the module 3, there will be execution of the all the modules and testing is done by receiving the output.

Therefore, the user authentication is complete.

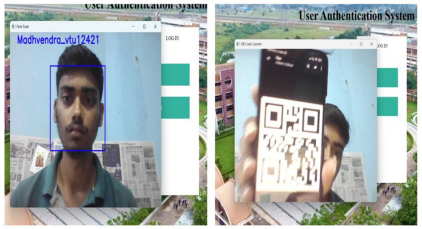


Figure 2. Face Scan &QR Code Scan

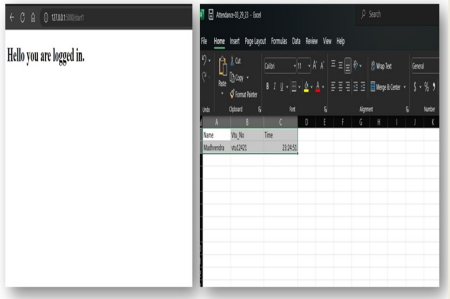


Figure 3. Result

STANDARDS AND POLICIES

To ensure that the model is safe, reliable, and effective in authentication of user . It requires a careful attention on data privacy, ethical considerations, accuracy and reliability, model documentation, regulatory compliance, model deployment, and ongoing evaluation and improvement by the developers.

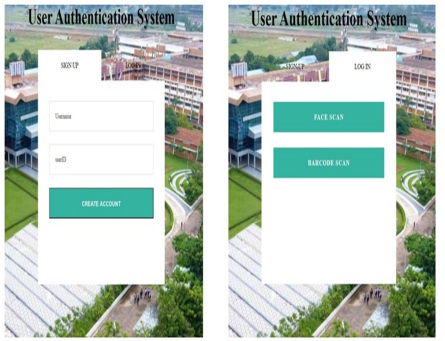


Figure 1. Signup & Login

CONCLUSIONS

In conclusion, a user authentication system using machine learning is a powerful tool that can enhance attendance tracking and improve security in a variety of settings. Furthermore, as technology continues to evolve, this system has the potential to become even more advanced, with improved algorithms, higher accuracy rates, and enhanced features. This system can automate attendance marking, provide real-time face identification and verification of individuals,verification through QR and deliver a more personalized experience for users.

ACKNOWLEDGEMENT

1. Dr. C.M. Chidambaranathan / Assistant Professor

2. 99407 68671

3. cmchidambaranatham@veltech.edu.in

References

[1] M. Monwar Hossain, “A Survey of User Authentication Systems”, (2019) volume 10, pp:513-519-Oriental Journey of Computer Science and Technology.

[2] Asad Masood Khattak, “AI based Login System using Facial Recognition”, (2021) volume 3- IEEE.

[3] Saad Alshahrani, “A Review of Biometric Authentication Technologies: Applications, Challenges, and Research Directions”, (2021) volume 113, pp:19-25- IEEE International Conference on Computational Intelligence and Computing Research.

[4] Sanjay Kumar, Syed Akbar Abbas Jafri, “A New User Identity Based Authentication, Using Security and Distributed for Cloud Computing”, (2020) volume 24, pp:42-48- IOP Conference Series:Materials Science and Engineering.

[5] Ben Wycliff Mugalu, Rodrick Calvin Wamala, ”Face Recognition as a Method of Authentication in a Web-Based System”, volume 21 pp:115-122- , National Conference on Communications.

[6] Paul C. van Oorschot, ”User Authentication—Passwords, Biometrics and Alternatives”, (2021) pp:55-90- Information Security and Cryptography.

[7] S. Khokad and V. Kala, ”A study of SLIDE Algorithm: Revolutionary AI Algorithm that Speeds Up Deep Learning on CPUs,” (2020$,pp.188-191-IEEE.$

[8] Munir Hussain, Amjad Mehmood, Shafiullah Khan, “Authentication Techniques and Methodologies used in Wireless Body Area Networks” (2019) volume 101- Journal of Systems Architecture.

[9] Muhammad Sajjad, Salman Khan, Tanveer Hussain, Arun Kumar Sangaih, “CNN-based anti-spoofing two-tier multi-factor authentication system” (2019) volume 126, pp:123-131- Pattern Recognition Letter.

[10] Verena Zimmermann “The password is dead, long live the password – A laboratory study on user perceptions of authentication schemes” (2020) volume 133, pp:26-44- International Journal of Human-Computer Studies.

[11] Angelica caro, “Authentication schemes and methods: A systematic literature review” (2018) volume 94, pp:30-37- Information and Software Technology.

[12] Zhengning Wang, Fanwei Zeng, Bing Zeng “Oriented attention ensemble for accurate facial expression recognition” (2021), volume 112- Pattern Recognition.

[13] Ben Wycliff Mugalu,Rodrick Calvin Wamala,Jonathan Serugunda,Andrew Katumba,”Face Recognition as a Method of Authentication in a Web-Based System”(2021)- National Conference on Communications.

[14] Rishabh Kaushik and Partha Pratim Ray”A Survey of Authentication Techniques for Distributed Systems”,(IEEE Access, 2021)- IEEE.

[15] Mohammad Moinul Islam,et al”Face Recognition Systems: A Review”(Journal of Ambient Intelligence and Humanized Computing, 2021)

[16] Krishna Dharavath,F.A.Talukdar,R.H.Laskar”Study on Biometric Authentication Systems,Challenges and Future Trends”,(2022)- IEEE

[17] Haiyu Chen, Yanzhe Zhu, and Chuan Qin,”A Novel User Authentication Scheme based on QR Code and Face Recognition”,(2021)