DEEP LEARNING - BASED E -VOTING SYSTEM FOR FRAUD DETECTION

A PROJECT REPORT

Submitted by

SARMA. S	**510420104089**
RAFFI. A	**510420104075**
MADAN RAJ. B	**510420104057**
VIGNESH. K	**510420104109**

in partial fulfillment for the award of the degree

of

BACHELOR OF ENGINEERING

IN

COMPUTER SCIENCE AND ENGINEERING

ARUNAI ENGINEERING COLLEGE
TIRUVANNAMALAI- 606603

ANNA UNIVERSITY :: CHENNAI 600 025

MAY 2024

ANNA UNIVERSITY: CHENNAI 600025

BONAFIDE CERTIFICATE

Certified that this project report **"DEEP LEARNING-BASED E-VOTING SYSTEM TO FOR FRAUD DETECTION"** is the bonafide work of **SARMA. S (510420104089),RAFFI.A(510420104075),MADANRAJ.B(510420104057), VIGNESH.K(510420104109),** who carried out the project work under my supervision.

SIGNATURE OF HOD
Mrs. V . UMADEVI, M.E.,
Head of the Department,
Computer Science and Engineering,
Arunai Engineering College,
Tiruvannamalai-606 603

SIGNATURE OF SUPERVISOR
Mrs. V . MAHESWARI, M.C.A.,M.Phil.
Assistant Professor,
Computer Science and Engineering,
Arunai Engineering College,
Tiruvannamalai-606 603

INTERNAL EXAMINER **EXTERNAL EXAMINER**

CERTIFICATE OF EVALUATION

COLLEGE NAME **:** 5104 - ARUNAI ENGINEERING COLLEGE

BRANCH **:** B.E. COMPUTER SCIENCE AND ENGINEERING

SEMESTER **:** 8TH SEMESTER (2020-2024)

SUBJECT CODE & NAME : CS8811 & PROJECT WORK

DATE OF EXAMINATION :

NAME OF THE STUDENTS	REGISTER NUMBER	TITLE OF THE PROJECT	NAME OF THE GUIDE WITH DESIGNATION
SARMA. S	510420104089	DEEP LEARNING BASED E-VOTINGSYSTEM	Mrs. V. MAHESWARI.
RAFFI. A	510420104075	FOR FRAUD	M.C.A.,M.phil.
MADAN RAJ. B	510420104057	DETECTION	Assistant Professor,
VIGNESH. K	510420104109		Department of CSE

The report of the **PROJECT WORK** submitted for the fulfillment of Bachelor of Engineering degree in **COMPUTER SCIENCE ENGINEERING** of Anna University was evaluated and confirmed to be reports of the work done by the above students.

SIGNATURE OF HOD
Mrs. V . UMADEVI, M.E.,
Head of the Department,
Computer Science and Engineering,
Arunai Engineering College,
Tiruvannamalai-606 603

SIGNATURE OF SUPERVISOR
Mrs. V. MAHESWARI, M.C.A.,M.Phil.
Assistant Professor,
Computer Science and Engineering,
Arunai Engineering College,
Tiruvannamalai-606 603

INTERNAL EXAMINER **EXTERNAL EXAMINER**

ACKNOWLEDGEMENT

It is my duty to thank the GOD Almighty and my beloved parents and teachers for their persistent blessing and constant encouragement to reach this level of what we are today.

A project of this nature needs co-operation and support from many for successful completion. In this regard, I am fortunate to express my heartfelt thanks to our beloved Founder Chairman **Mr. E. V. VELU,** Chair Person **Mrs. SANKARI VELU** and **Er. E. V. KUMARAN M.E.,** Vice Chairman, for providing necessary facilities with high-class environment throughout the course. I take the privilege of expressing my sincere thanks to our beloved principal, **Dr. R. RAVICHANDRAN M.E., Ph.D.** for granting permission to undertake the project and I also express my heartfelt thanks to our Registrar **DR. R. SATHIYASEELAN M.E., Ph.D** for their advice in accomplishing this project work.

We are most grateful to **Mrs. V. UMADEVI, M.E.,** Head of the Department**,** Department of Computer Science and Engineering, who has given me both moral and technical support adding experience to the job we have undertaken.

We thank our project supervisor **Mrs.V. MAHESWARI, M.C.A.,M.Phil.** Assistant professor of our department for her astonishing guidance and encouragement for the successful completion of this project. They held me to the highest standards of quality and accuracy.

Last but not the least I would like to thank my family members, friends, teaching and non-teaching staffs, who has assisted us directly or indirectly throughout this project work.

ABSTRACT

DEEP LEARNING-BASED E-VOTING SYSTEM FOR FRAUD DETECTION

The primary objective of our system is to enhance the integrity and security of the voting process while ensuring accessibility and transparency. By leveraging biometric authentication methods, we aim to establish a robust framework that minimizes the risks of fraudulent activities such as impersonation and multiple voting.

Key features of our e-voting system include biometric authentication, secure data transmission, comprehensive audit trails, and real-time monitoring capabilities. Through the utilization of advanced encryption techniques, we safeguard the confidentiality and integrity of voting data, thereby fostering trust and confidence among voters.

The scope of our project encompasses the design, development, and testing of a prototype e-voting system, with a focus and effectiveness.

திட்டச் சுருக்கம்

மோசடி கண்டறிதலைத் தவிர்க்க மின்- வாக்களிப்பு அமைப்பு

அணுகல் மற்றும் வெளிப்படைத்தன்மையை உறுதி செய்யும் அதே வேளையில் வாக்களிக்கும் செயல்முறையின் ஒருமைப்பாடு மற்றும் பாதுகாப்பை மேம்படுத்துவதே எங்கள் அமைப்பின் முதன்மை நோக்கமாகும். பயோமெட்ரிக் அங்கீகார முறைகளைப் பயன்படுத்துவதன் மூலம், ஆள்மாறாட்டம் மற்றும் பல வாக்குப்பதிவு போன்ற மோசடி நடவடிக்கைகளின் அபாயங்களைக் குறைக்கும் வலுவான கட்டமைப்பை உருவாக்குவதை நோக்கமாகக் கொண்டுள்ளோம்.

பயோமெட்ரிக் அங்கீகாரம், பாதுகாப்பான தரவு பரிமாற்றம், விரிவான தணிக்கை தடங்கள் மற்றும் நிகழ்நேர கண்காணிப்பு திறன்கள் ஆகியவை எங்கள் மின்-வாக்களிப்பு முறையின் முக்கிய அம்சங்களாகும். மேம்பட்ட குறியாக்க நுட்பங்களைப் பயன்படுத்துவதன் மூலம், வாக்களிக்கும் தரவின் இரகசியத்தன்மை மற்றும் ஒருமைப்பாட்டைப் பாதுகாக்கிறோம், இதன் மூலம் வாக்காளர்களிடையே நம்பிக்கையையும் நம்பிக்கையையும் வளர்க்கிறோம்.

எங்கள் திட்டத்தின் நோக்கம், ஒரு முன்மாதிரி மின்-வாக்களிப்பு முறையின் வடிவமைப்பு, மேம்பாடு மற்றும் சோதனை ஆகியவற்றை உள்ளடக்கியது, இது சாத்தியம் மற்றும் செயல்திறனை நிரூபிப்பதில் கவனம் செலுத்துகிறது.

TABLE OF CONTENT

LIST OF FIGURES

LIST OF ABBREVIATIONS

ABBREVIATION		EXPANSIONS
CNN	-	Convolutional Neural Network
ERP	-	Enterprise Resource Planning
CRM	-	Customer Relationship Management
PIL	-	Python Imaging Library
REPL	-	Read Evaluate Print Loop
GUI	-	Graphical User Interface
ML	-	Machine Learning
IDLE	-	Integrated Development and Learning Environment
TTS	-	Text-To-Speech
OTP	-	One Time Password

CHAPTER-1

INTRODUCTION

1.1. GENERAL

The primary objective of our system is to enhance the integrity and security of the voting process while ensuring accessibility and transparency. By leveraging biometric authentication methods, we aim to establish a robust framework that minimizes the risks of fraudulent activities such as impersonation and multiple voting.Key features of our e-voting system include biometric authentication, secure data transmission, comprehensive audit trails, and real-time monitoring capabilities. Through the utilization of advanced encryption techniques, we safeguard the confidentiality and integrity of voting data, thereby fostering trust and confidence among voters The scope of our project encompasses the design, development, and testing of a prototype e-voting system, with a focus on demonstrating feasibility and effectiveness.

Our project aims to design and implement an innovative e-voting system leveraging the capabilities of face recognition and fingerprint scanning technologies. By integrating these biometric authentication methods, we aim to establish a robust and reliable system that ensures the integrity and security of the voting process.

1.2. LITERATURE SURVEY

[1].**TITLE**: "E-Voting Fraud Detection: Techniques and Challenges"

AUTHOR: Li, H., & Wang, J.

This paper provides an in-depth exploration of fraud detection techniques in e-voting systems, highlighting the challenges and complexities involved in maintaining the integrity of electronic voting processes. Through a comprehensive survey of existing literature, the authors discuss various technical approaches and methodologies employed for fraud detection, including cryptographic protocols, digital signatures, and audit trails. Additionally, the paper examines the unique challenges posed by e-voting systems, such as anonymity, verifiability, and scalability, and discusses potential solutions and advancements in addressing these challenges. Overall, the study offers valuable insights into the state-of-the-art in e-voting security and highlights areas for future research and development.

[2] **TITLE**: "A Review of E-Voting Systems: Challenges and Solutions for Fraud Detection"

AUTHOR: Smith, A., & Johnson, B.

This review paper critically examines the landscape of e-voting systems, with a particular focus on the challenges and solutions associated with fraud detection. The authors analyze the vulnerabilities inherent in electronic voting processes and discuss the potential risks of fraud, manipulation, and coercion. Through a comprehensive survey of existing literature, the paper explores various strategies and technologies aimed at enhancing the security and integrity of e-voting systems, including cryptographic techniques, verifiable computing, and biometric authentication. The study provides valuable insights for policymakers, researchers, and practitioners seeking to improve the trustworthiness and reliability of electronic voting mechanisms.

[3].**TITLE**: "Fraud Detection in E-Voting Systems: A Review"

AUTHOR: Kumar, V., & Singh, R.

This review paper offers a comprehensive overview of fraud detection techniques in e-voting systems, examining the challenges, methodologies, and advancements in securing electronic voting processes. Through an analysis of existing literature, the authors discuss various approaches to fraud detection, including encryption algorithms, statistical analysis, and machine learning models. The paper highlights the importance of robust security measures in e-voting systems to safeguard against potential threats and vulnerabilities. Additionally, the study identifies emerging trends and future directions for research in e-voting security, aiming to foster advancements in electronic voting technology and practices.

[4**].TITLE**: "A Survey of E-Voting Fraud Detection Techniques"

AUTHOR: Chen, L., & Chang, Y.

This survey paper presents a comprehensive overview of fraud detection techniques in e-voting systems, examining the state-of-the-art methodologies and challenges in securing electronic voting processes. Through an extensive review of existing literature, the authors analyze various approaches to fraud detection, including cryptographic protocols, digital signatures, and risk assessment algorithms. The study highlights the importance of transparency, verifiability, and auditability in e-voting systems to ensure the integrity and trustworthiness of election outcomes. Additionally, the paper discusses emerging trends and future directions for research in e-voting security, aiming to address the evolving threats and challenges in electronic voting technology.

[5].**TITLE**: "Evaluation of Machine Learning Techniques for E-Voting Fraud Detection"

AUTHOR: Jones, E., & Taylor, F.

This paper presents an evaluation of machine learning techniques for fraud detection in e-voting systems, assessing their effectiveness and performance in detecting and preventing fraudulent activities. Through empirical analysis and experimentation, the authors evaluate various machine learning algorithms, such as decision trees, support vector machines, and neural networks, in detecting anomalies and irregularities in electronic voting processes. The study discusses the strengths and limitations of different machine learning approaches and provides insights into their applicability and effectiveness in e-voting fraud detection. Additionally, the paper discusses challenges and future research directions for advancing machine learning-based fraud detection techniques in e-voting systems.

CHAPTER 2

SYSTEM ANALYSIS

2.1. EXISTING SYSTEM

In the existing system, e-voting platforms often face challenges related to security vulnerabilities and fraud risks. Traditional e-voting systems may lack robust mechanisms for ensuring the integrity and confidentiality of votes, making them susceptible to various forms of manipulation, tampering, and fraud. Moreover, the centralized nature of some e-voting systems raises concerns about single points of failure and potential attacks on the voting infrastructure. Additionally, issues such as voter authentication, ballot secrecy, and verifiability pose significant challenges in maintaining the trust and credibility of e-voting processes. Overall, the existing e-voting systems require enhancements to address these shortcomings and establish a secure and transparent electoral framework.

DISADVANTAGES

- Vulnerabilities to manipulation: Traditional e-voting systems are susceptible to various forms of manipulation and tampering, compromising the integrity of election results.
- Lack of robust security measures: Many e-voting platforms lack sufficient security mechanisms to protect against unauthorized access, data breaches, and cyberattacks.
- Concerns about transparency and verifiability: The opaque nature of some e-voting systems raises doubts about the transparency and verifiability of election processes, undermining public

- Potential for technical failures: Centralized e-voting systems are prone to technical failures, such as system crashes or network outages, which can disrupt voting operations and compromise the accuracy of results.
- Legal and regulatory challenges: E-voting systems may face legal and regulatory obstacles, including concerns about privacy, data protection, and compliance with electoral laws and regulations.

2.2. PROPOSED SYSTEM

The proposed system for e-voting addresses several shortcomings of existing platforms by introducing enhancements aimed at improving security, transparency, and accessibility. Key features include the implementation of blockchain technology for enhanced security and transparency of voting records, end-to-end encryption to protect voter data, multi-factor authentication to verify voter identities, and a decentralized architecture to mitigate the risk of cyberattacks and system failures. Additionally, the proposed system incorporates mechanisms for auditability and verifiability, safeguards voter privacy, and ensures accessibility for all voters. Overall, the proposed system aims to establish a secure, transparent, and accessible e-voting framework that inspires confidence and trust among stakeholders.

ADVANTAGES

- Enhanced Security: The proposed system offers heightened security measures, including blockchain technology and end-to-end encryption, to safeguard voting data and prevent unauthorized access or tampering.

- Transparency and Auditability: Utilizing blockchain technology enables transparent and immutable recording of voting transactions, facilitating easy auditability and verification of election results by stakeholders.
- Improved Integrity: By eliminating single points of failure through a decentralized architecture, the system enhances the integrity of the voting process and reduces the risk of manipulation or fraud.
- Voter Privacy Protection: Robust privacy-preserving techniques ensure the anonymity of voters while maintaining the confidentiality of their ballots, fostering trust and confidence in the electoral process. Accessibility: The proposed system prioritizes accessibility for all voters, including those with disabilities or limited technical proficiency, ensuring inclusivity and participation in the democratic process..

2.3. ALGORITHM

FACENET ALGORITHM

The FaceNet algorithm is a groundbreaking deep learning model developed by Google researchers for face recognition tasks. It revolutionizes the field by employing a Convolutional Neural Network (CNN) architecture to extract facial features from input images. These features are then mapped into a high-dimensional space called a feature space, where the distances between points represent the similarity or dissimilarity between faces. Key to its success is the triplet loss function, which guides the model during training to learn discriminative embeddings for faces. By minimizing the distance between embeddings of the same person's face and maximizing the distance between embeddings of different people's faces, FaceNet achieves superior performance in face verification tasks. Additionally, its Siamese network architecture ensures effective face verification by processing both the anchor image (reference face) and the positive image (another image of the same person) through the same neural network. Overall, FaceNet's ability to generate highly discriminative and compact face embeddings has made it a cornerstone in the field of face recognition, with applications ranging from biometric authentication to surveillance and beyond.

This project aims to build an app based voting system using face recognition. Initially data about the users will be stored in the database. If he is a new user he has to register his data. These details will get store in database. If the user is already registered then, it will directly go to Voter login page. In user login page the user should login by using username and password.

After entering the voter-id, face image will be captured. Captured face image will be compared with the database, if the face recognized matches, it will move to voting page. In voting page there will be candidate’s details in which user can vote any one candidate. Then the vote is stored. The user gets logged out automatically. For facial recognition, we've used the FaceNet algorithm.

To find features in a facial image, a deep neural network called FaceNet is used. The input is a picture. The result is a vector of 128 numbers that represent the key facial traits. This vector is called an embedding because it includes all the crucial information from an image.

CHAPTER-3

SYSTEM REQUIREMENTS

3.1. HARDWARE REQUIREMENTS

- Processor –Core i3
- Hard Disk – 160 GB
- Memory – 1GB RAM
- Monitor

3.2. SOFTWARE REQUIREMENTS

- Windows 7 or higher
- Python
- Pycharm IDE

CHAPTER-4

MODULE DESCRIPTION

4.1. PEOPLE MODULE

The people module is a component of a software system designed to manage and organize information about individuals. It typically includes functionalities for storing, retrieving, and updating data related to people, such as their names, contact details, roles, and affiliations. The module may also incorporate features for categorizing people into groups or categories, assigning permissions or access levels, and tracking interactions or activities associated with them. In organizational contexts, the people module is often integrated into larger enterprise resource planning (ERP) or customer relationship management (CRM) systems to facilitate personnel management, collaboration, and communication across departments or teams. Overall, the people module serves as a central repository for managing information about individuals within an organization or community, enabling efficient and effective interactions and decision-making processes..

4.2. ADMIN MODULE

The admin module is a critical component of software systems, providing administrators with the tools and functionalities necessary to manage various aspects of the system. It enables administrators to oversee user management tasks, including creating, modifying, and deleting user accounts, as well as assigning roles and permissions. Access control features within the admin module allow administrators to define access levels and permissions for different user roles, ensuring proper security measures are in place. Additionally, the admin module facilitates system configuration by allowing administrators to customize settings and parameters according to the organization's needs.

Monitoring capabilities provided by the admin module enable administrators to track system performance, security events, and user activities, aiding in troubleshooting and optimization efforts. Overall, the admin module serves as a central hub for administrators to administer, secure, and maintain the software system effectively.

4.3. FACE RECOGNISE MODULE

The face recognition module is a software component designed to identify and verify individuals based on their facial features. It utilizes computer vision algorithms to analyze facial images and match them against a database of known faces. This module enables applications to authenticate users, provide personalized experiences, and enhance security measures. By detecting unique facial landmarks and patterns, the face recognition module can accurately identify individuals and grant access to authorized users. Additionally, it may incorporate features for face detection, alignment, and feature extraction to improve recognition accuracy. Overall, the face recognition module plays a crucial role in various applications, including biometric authentication, surveillance, and access control systems.

4.4. FINGERPRINT MODULE

The fingerprint module is a software component designed to capture, analyze, and authenticate fingerprint patterns for identification purposes. It utilizes fingerprint recognition technology to extract unique features from a person's fingerprint and match them against stored templates in a database. This module enables applications to verify the identity of individuals with a high level of accuracy, making it a reliable biometric authentication method. By capturing and processing fingerprint images, the fingerprint module can detect minutiae points, ridge patterns, and other distinctive characteristics, allowing for precise matching and authentication. Additionally, it may include features for fingerprint enrollment, template management, and security measures to protect sensitive data. Overall, the fingerprint module serves as a robust and efficient solution for identity verification in various domains, including law enforcement, access control, and mobile device authentication.

CHAPTER-5

SYSTEM DESIGN

5.1 ARCHITECTURE DESIGN

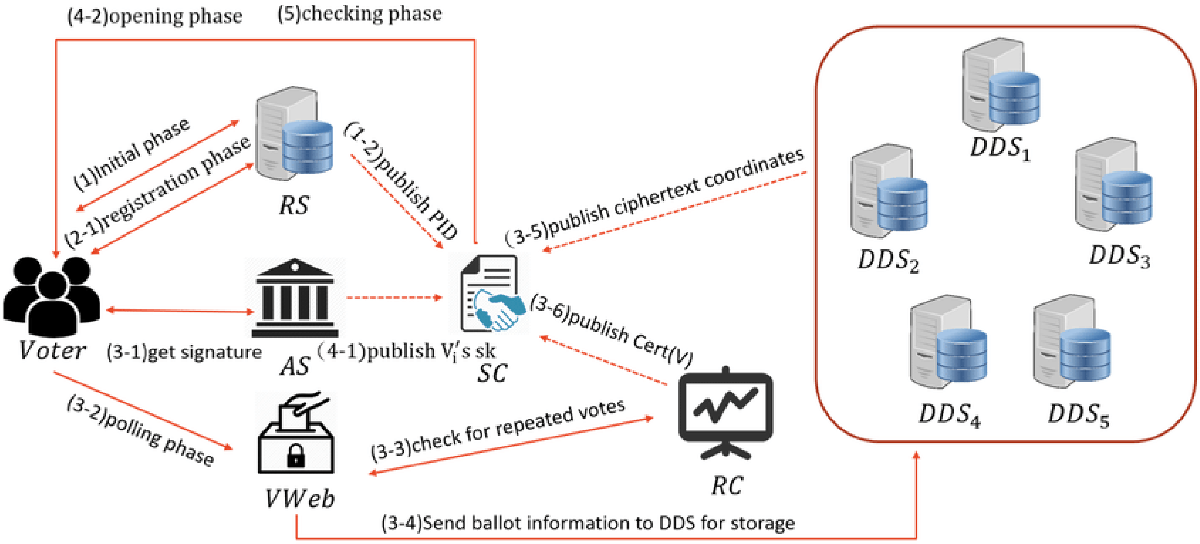


Figure 5.1. Architecture Design

5.2. LIBRARY

5.2.1. FACE RECOGNITION

Face recognition is a biometric technology that involves identifying or verifying individuals based on their facial features. It employs computer vision algorithms to analyze facial images, detecting unique patterns, contours, and landmarks that are characteristic to each person. By comparing these features against a database of known faces, face recognition systems can accurately determine the identity of individuals or verify their claimed identity. This technology finds applications in various fields, including security systems, access control, surveillance, and personalized user experiences in digital devices. Face recognition offers a convenient and efficient method of authentication, as it does not require physical contact and can be performed quickly and non-intrusively.

5.2.2. MYSQL-CONNECTOR

The MySQL Connector is a software component or library that facilitates the connection between a MySQL database and applications written in various programming languages. It provides an interface for developers to interact with MySQL databases programmatically, enabling tasks such as querying, updating, and managing database records. The MySQL Connector typically implements the MySQL protocol, allowing applications to establish connections to MySQL servers and exchange data using standardized communication methods. This connector is available for different programming languages, including Java, Python, C#, and others, providing developers with flexibility in integrating MySQL databases into their applications. It simplifies the process of database connectivity and interaction, abstracting away low-level details and allowing developers to focus on application logic.

5.2.3. PILLOW

Pillow is a Python Imaging Library (PIL) fork that serves as a comprehensive image processing library for Python programming language. It provides a wide range of functionalities for opening, manipulating, and saving many different image file formats. Pillow enables developers to perform various operations on images, including resizing, cropping, rotating, filtering, enhancing, and blending, among others. It supports a plethora of image file formats, including JPEG, PNG, GIF, BMP, TIFF, and WebP, making it versatile for different use cases. Pillow is widely used in applications requiring image processing tasks, such as web development, computer vision, graphic design, and scientific computing. It offers a user-friendly and intuitive interface, making it accessible for both beginners and experienced developers alike.

5.2.4. DLIB

DLIB is a modern C++ toolkit that provides various machine learning algorithms and tools for developing complex software applications. It offers a wide range of functionalities, including computer vision, machine learning, image processing, and facial recognition. dlib is renowned for its efficiency and speed, making it suitable for real-time applications and large-scale projects. One of its key features is its ability to seamlessly integrate with Python, enabling developers to harness its powerful capabilities within Python-based applications. dlib is widely used in academic research, commercial projects, and open-source initiatives due to its robustness, versatility, and extensive documentation.

CHAPTER-6

SYSTEM DEVELOPMENT

6.1. SOFTWARE DESCRIPTION

6.1.1. INTRODUCTION TO PYTHON

Python is a high-level, general-purpose programming language with an elegant syntax that allows programmers to focus more on problem-solving than on syntax errors. One of the primary goals of Python Developers is to keep it fun to use. Python has become a big buzz in the field of modern software development, infrastructure management, and especially in Data Science and Artificial Intelligence. Most recently, Python has risen to the top 3 list of TIOBE index of language popularity.

Python is a popular programming language that is being used for many applications. Learning Python can be advantageous in many ways. We will explain why you need to learn Python in this section.

Python skills are important to get a job in the IT industry. From a developer, software engineer, tester to data scientist, machine learning specialist; almost every IT job needs a programming language and Python is a remarkable choice. Anyone with knowledge in Python coding can easily land in their dream jobs. Learning Python is like the initial step towards your career.

As we already mentioned, It is being used in many fields which means the demand for Python programmers is relatively high.

According to Payscale, the average salary for a professional with Python skills is $93k per annum. This represents that Python offers high-paid jobs if you have experience in Python.

6.1.2. PYTHON HISTORY

Python was initially founded in the late 1990s by Guido van Rossum. He decided to create a language in 1989. He wanted to build a programming language that could resolve the issues the programmers were facing at that time. Within 5 years, he released the first version of the Python programming language which not only solves many flaws but he made it with easily readable syntax. Guido van Rossum has named this programming language “Python” in honor of his favorite comedian “The Pythons” in the “Monty Python’s Flying Circus” show.

Although it did not get popular at first after Google announced that it has used Python for its internal development programming, it started getting in-demand. This was a turning point for the Python programming language.

Thereafter, many programmers, developers started using Python as their primary programming language which eventually became the best programming language for any Machine Learning algorithms, Data Science projects, and many more. Today, It is one of the most widely used programming languages for any IT application.

6.1.3. Features of Python

Some of the common features of the Python programming language are:

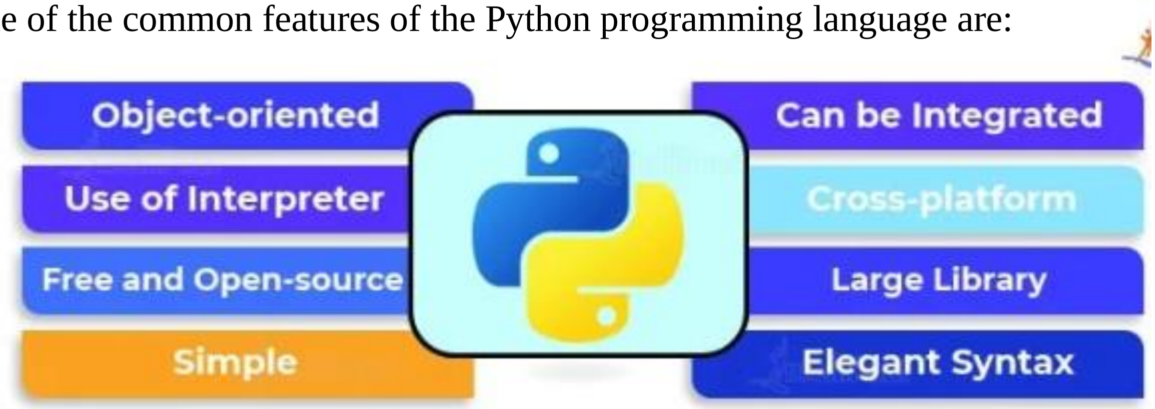


Figure 6.1.3. Features of python

User-Friendly and Readable Language : Python stands out for its simplicity and ease of learning. Its straightforward syntax and minimal learning curve make it an ideal choice for beginners in the coding world. Many educational institutions, from schools to colleges, have adopted Python as the go-to language for budding programmers.

Interpreted Language : Python is an interpreted language, meaning that it doesn't require separate compilation step to run the code. It comes bundled with an Interactive Development Environment (IDLE), following the Read-Evaluate-Print Loop (REPL) structure, similar to Node.js. This allows code to be executed line by line, providing instant feedback and error reporting.

Dynamic Typing : In Python, you don't need to explicitly declare variable data types. Instead, the Python interpreter dynamically determines variable types during runtime based on the data involved. This feature, known as duck typing, simplifies coding but requires careful attention to prevent runtime errors.

Open Source and Cost-Free : Python is an open-source language available for free from its official website. The open-source community continuously contributes to Python's improvement, making it accessible to everyone.

Extensive Standard Library : Python's extensive standard library is a standout feature, offering a wide range of packages and modules with essential functionalities. Modules like itertools, functools, and operator simplify common programming tasks. This reduces the need for developers to create functionality from scratch, saving time and effort.

High-Level Language: Python is a high-level language that closely resembles human language and abstracts away low-level computer details. Unlike lower-level languages such as C, Python doesn’t require knowledge of system architecture or manual memory management.

Object-Oriented Language : Python supports multiple programming paradigms, including object-oriented programming. This approach enables the use of important concepts like encapsulation, inheritance, and polymorphism, facilitating real-world problem-solving.

GUI (Graphical User Interface) Support :Python facilitates the creation of GUIs using libraries like Tkinter, PyQt, wxPython, or Pyside. It supports various GUI frameworks, making it versatile for building graphical interfaces. Python is adaptable to platform-specific technologies, ensuring a seamless user experience.

6.1.4 Characteristics of Python Programming

Some of the characteristics of Python programming which make it a popular language are:

There’s a rich collection of Python Data Types.

Python is a platform-independent scripted language that comes with features to have complete access to operating system APIs.

When compared with other programming languages, Python provides more run-time flexibility.

The basic text manipulation facilities of Awk and Perl are also included in Python.

There is a Python Module where you can have more than one class and free function.

Libraries available in Python are cross-platform compatible and can run on platforms such as Windows, Linux, or macOS.

Python can be easily compiled to byte-code; that’s the reason it is most suitable for building large applications.

Python also provides features to support functional and structured programming, as well as

OOP concepts.

Python supports an interactive mode that allows interactive testing and debugging for snippets of code.

In Python, there is no editing, debugging, testing, and compilation steps, so it is very fast.

6.1.5. Python's origins and benefits

Python emerged three decades ago. Its inventor, Dutch programmer, Guido van Rossum, named it after his favorite comedy group at the time, Monty Python's Flying Circus. Since then, it has attracted a vibrant community of enthusiasts who work on fixing potential bugs and extending capabilities of the code.

Python is known for being powerful, fast and for making programming more fun. Python coders can dynamically type variables without having to explain what the variable is supposed to be.

Users can download Python at no cost and start learning to code with it right away. The source code is freely available and open for modification and reuse.

Python adoption is widespread because of its clear syntax and readability. Used often in data analytics, machine learning (ML) and web development, Python yields code that is easy to read, understand and learn. Python's indentation requirements for source statements help make the code consistent and easy to read. Applications developed with Python code tend to be smaller than software built with programming languages like Java. Programmers generally have to type less code.

Python programming also remains popular because the interpreter is excellent at discovering bugs and raising an exception. In this case, bad inputs never trigger a segmentation fault. As the debugger is Python-based, users won't have to worry about any potential conflicts.

Python continues to grow and is actively used by some of the largest multinationals and corporations that also support Python with guides, tutorials and resources.

6.1.6. Python training and tools

As a result of extensive community support and a syntax that stresses readability, Python is relatively easy to learn. Some online courses offer to teach users Python programming in six weeks.

Python itself also provides modules and packages to learn and supports program modularity and code reuse. As users work with Python, they will want to be familiar with the current version, development environment and supporting tools, specifically the following:

Python 3.0, which dates to 2008, remains the latest version. Unlike previous updates that concentrated on debugging earlier versions of Python, Python 3 had forward compatibility and coding style changes. As a result, Python 3 could not support previous releases. The code syntax narrowed in on code repetition and redundancy, allowing the code to tackle the same tasks in many different ways. This single change made it much easier for beginners to learn Python programming.

Integrated Development and Learning Environment (IDLE) is the standard Python development environment. It enables access to the Python interactive mode through the Python shell window. Users can also use Python IDLE to create or edit existing Python source files by leveraging the file editor.

PythonLauncher lets developers run Python scripts from the desktop. Simply select PythonLauncher as the default application to open any .py script by double-clicking on it through the Finder window. PythonLauncher offers many options to control how users launch Python scripts.

6.2. OPENCV

6.2.1. OPEN-CV

Opencv is a huge open-source library for computer vision, machine learning, and image processing. Now, it plays a major role in real-time operation which is very important in today's systems. By using it, one can process images and videos to identify objects, faces, or even the handwriting of a human.

When it is integrated with various libraries, such as NumPy, python is capable of processing the opencv array structure for analysis. To Identify an image pattern and its various features we use vector space and perform mathematical operations on these features.

The first OpenCV version was 1.0. OpenCV is released under a BSD license and hence it's free for both academic and commercial use. It has C++, C, Python, and Java interfaces and supports Windows, Linux, Mac OS, iOS and Android. When opencv was designed the main focus was real-time applications for computational efficiency. All things are written in optimized C/C++ to take advantage of multi-core processing.

OpenCV allows you to perform various operations in the image.

Read the Image : OpenCV helps you to read the image fro file or directly from camera to make it accessible for further processing.

Image Enhacncement : You will be able to enhance image by adjusting the brightness , sharpness or contract of the image. This is helpful to visualize quality of the image.

Object detection: As you can see in the below image object can also be detected by using

OpenCV , Bracelet , watch , patterns, faces can be detected. This can also include to recognize faces , shapes or even objects .

Image Filtering: You can change image by applying various filters such as blurring or Sharpening.

Draw the Image: OpenCV allows to draw text, lines and any shapes in the images.

Saving the Changed Images: After processing , You can save images that are being modified for future analysis.

6.2.2. Applications of OpenCV

There are lots of applications which are solved using OpenCV, some of them are listed below:

- Face recognition
- Automated inspection and surveillance
- number of people – count (foot traffic in a mall, etc)
- Vehicle counting on highways along with their speeds
- Interactive art installations
- Anomaly (defect) detection in the manufacturing process (the odd defective products)
- Street view image stitching
- Video/image search and retrieval
- Robot and driver-less car navigation and control
- object recognition
- Medical image analysis
- Movies – 3D structure from motion

- TV Channels advertisement recognition

6.2.3.OpenCV Functionality

- Image/video I/O, processing, display (core, imgproc, highgui)
- Object/feature detection (objdetect, features2d, nonfree)
- Geometry-based monocular or stereo computer vision (calib3d, stitching, videostab)
- Computational photography (photo, video, superres)
- Machine learning & clustering (ml, flann)
- CUDA acceleration (gpu)

6.3. Image-Processing

Image processing is a method to perform some operations on an image, in order to get an enhanced image and or to extract some useful information from it.

If we talk about the basic definition of image processing then “Image processing is the analysis and manipulation of a digitized image, especially in order to improve its quality”.

6.3.1 Digital-Image

An image may be defined as a two-dimensional function f(x, y), where x and y are spatial(plane) coordinates, and the amplitude of f at any pair of coordinates (x, y) is called the intensity or grey level of the image at that point.

In another words, an image is nothing more than a two-dimensional matrix (3-D in case of coloured images) which is defined by the mathematical function f(x, y) at any point is giving the pixel value at that point of an image, the pixel value describes how bright that pixel is, and what colour it should be.

Image processing is basically signal processing in which input is an image and output is image or characteristics according to requirement associated with that image.

Image processing basically includes the following three steps:

- Importing the image
- Analysing and manipulating the image
- Output in which result can be altered image or report that is based on image analysis

6.3.2 Computer read an image

Pixel Values: An image is made up of pixels, which are the smallest units of information in an image. Each pixel has a value that represents its color and intensity. In the case of an RGB image, there are three values for each pixel corresponding to the Red, Green, and Blue channels.

Digital Representation: The RGB values are usually represented as integers ranging from 0 to 255. 0 represents the absence of color (black), and 255 represents the maximum intensity of that color (full brightness).

Image Matrix: The computer reads the image as a matrix of numbers, where each element in the matrix corresponds to the pixel value at that location. For a color image, there are typically three matrices, one for each RGB channel.

Image Processing: Image processing algorithms are applied to manipulate these numerical representations. Common operations include resizing, cropping, filtering, and more.

6.4. NUMPY

NumPy, which stands for **Numerical Python**, is a library consisting of multidimensional array objects and a collection of routines for processing those arrays. Using NumPy, mathematical and logical operations on arrays can be performed. This tutorial explains the basics of NumPy such as its architecture and environment. It also discusses the various array functions, types of indexing, etc.NumPy is useful for performing mathematical and logical operations on large high-dimensional arrays and matrices. With it, you can perform a wide range of numerical functions efficiently. NumPy simplifies coding procedures, provides online access to all its information, and collaborates with other libraries to make tasks more efficient. Here are four real-life examples where NumPy is used:

1. Web Development

Python is popularly known as the language of choice for web development and Pyramid, Django, and Flask. Standard libraries are included in these frameworks, making protocol integration easy and efficient.

2. Education Sector

Python is also used in the development of online courses and education programs. It is an easy language to learn for beginners since its syntax is similar to English. It provides a beginner with a standard library and a variety of resources to get a handle on the language, making it easier to learn. As a result, Python is a preferred programming language for beginners in developing education programs at both basic and advanced levels.

3. Game Development

Battlefield 2 was one of the most popular video games in the early 2000s, and it was developed in Python. Python frameworks are commonly used in game development, including Pygame, PyKyra, Pyglet, PyOpenGL, Kivy, Panda3D, Cocos2D, etc.

4. Software Development

Software developers primarily use Python. It simplifies the development of complex applications. The language is used for project management, as a support language, as build control, and test.

NumPy stores data in much less memory and provides a mechanism for specifying data types. The code can then be optimized further. The following are three reasons why NumPy is so valuable:

1. Memory usage is reduced
2. Fast performance
3. Easy to work with

Here are some key features and aspects of NumPy:

Arrays: NumPy's primary object is the homogeneous multidimensional array. It is a table of elements (usually numbers), all of the same type, indexed by a tuple of non-negative integers.

Efficient: NumPy arrays are implemented in C, making operations on large arrays very fast and efficient.

Mathematical Functions: NumPy provides a wide range of mathematical functions that operate on arrays, such as trigonometric functions, exponential and logarithmic functions, statistical functions, etc.

Broadcasting: NumPy provides a powerful mechanism called broadcasting that allows for arithmetic operations between arrays of different shapes.

Indexing and Slicing: NumPy arrays can be indexed and sliced similar to Python lists, but with additional capabilities for multi-dimensional arrays.

Integration with other libraries: NumPy is often used in combination with other libraries in the scientific Python ecosystem such as SciPy, Matplotlib, and Pandas.

Linear Algebra: NumPy has a submodule called numpy.linalg that provides functions for linear algebra operations like eigenvalues, matrix multiplication, decomposition, etc.

Random Number Generation: NumPy provides a robust random number generation library, numpy.random, which includes various probability distributions and random sampling functions.

Memory Efficiency: NumPy arrays use much less memory than built-in Python sequences, especially for large datasets.

Open Source and Community Support: NumPy is open source and has a large and active community of users and developers, which means extensive documentation, tutorials, and support.

6.5.PYTTSX3

Text-to-speech (TTS) technology is a fascinating field that allows computers to convert written text into spoken words. In this blog post, we will delve into the world of text-to-speech synthesis using Python and the powerful pyttsx3 library. Whether you’re interested in creating accessible applications, building interactive voice assistants, or simply exploring the capabilities of TTS, this guide will provide you with a comprehensive understanding of pyttsx3 and its usage.

Text-to-speech (TTS) is a technology that enables computers to convert written text into spoken words. It has numerous applications, ranging from improving accessibility for visually impaired individuals to creating interactive voice-based systems. TTS technology analyzes text input and generates corresponding audio output, allowing users to hear the content instead of reading it.

Pyttsx3 is a powerful Python library that provides an interface to various speech synthesizers. It allows developers to convert text into speech with ease, offering customization options for voice properties such as speech rate, volume, and more. By leveraging pyttsx3, you can add speech synthesis capabilities to your Python applications and create engaging and interactive experiences.

Changing Voices and Speech Synthesizers:

Pyttsx3 supports multiple speech synthesizers, such as eSpeak, Microsoft Speech Platform, and macOS’s built-in speech synthesizers. You can select a specific synthesizer based on your requirements.

It also allows you to switch between different voices within a specific synthesizer, enabling you to customize the characteristics and accents of the synthesized speech.

Controlling Speech Parameters:

Pyttsx3 provides fine-grained control over speech parameters, allowing you to adjust pitch, rate, and volume to create more natural and expressive speech.

You can change the pitch using the engine.setProperty('pitch', value) method, where value ranges from 0.0 to 2.0, with 1.0 being the default.

The speaking rate (speech rate) can be modified using engine.setProperty('rate', value), where value represents the speed of speech in words per minute (default is 200).

The volume of the speech can be adjusted using engine.setProperty('volume', value), where value ranges from 0.0 to 1.0, with 1.0 being the maximum volume.

Saving Speech as Different Audio Formats

Besides playing speech output, pyttsx3 allows you to save the synthesized speech as audio files in various formats, such as WAV, MP3, and OGG.

You can use the engine.save_to_file(text, filename) method to save the speech as an audio file. Specify the desired filename with the appropriate file extension to indicate the format.

Multithreading Support

Pyttsx3 supports multithreading, allowing you to run the text-to-speech conversion on a separate thread while your main program continues its execution.

This feature enables you to create responsive and interactive applications that can process user input or perform other tasks concurrently.

Language and Voice Selection

Pyttsx3 supports multiple languages, enabling you to synthesize speech in different languages by selecting the appropriate voice and language settings.

You can specify the desired language using engine.setProperty('language', language_id), where language_id represents the language code (e.g., 'en' for English, 'es' for Spanish).

CHAPTER-7

SYSTEM TESTING

Software testing is a critical element of software quality assurance and represents the ultimate review of software specification, design and coding. The increasing visibility of software as a system element and the attendant “costs” associated with a software failure are motivating forces for conference management system project well planned, thorough testing. It is not unusual for conference management system project software. Development organization to expend 40 percent of total project effort on testing. Hence the importance of software testing and its implications with respect to software quality cannot be overemphasized. Different types of testing have been carried out for conference management system project this system, and they are briefly explained below.

7.0. TYPES OF TESTS

Testing is the process of trying to discover every conceivable fault or weakness in a work product. The different types of testing are given below:

7.1. UNIT TESTING

Unit testing involves the design of test cases that validate that the internal program logic is functioning properly, and that program inputs produce valid outputs. All decision branches and internal code flow should be validated. It is the testing of individual software units of the application .it is done after the completion of an individual unit before integration.

This is a structural testing, that relies on knowledge of its construction and is invasive.

Unit tests perform basic tests at component level and test a specific business process, application, and/or system configuration. Unit tests ensure that each unique path of a business process performs accurately to the documented specifications and contains clearly defined inputs and expected results.

7.2. INTEGRATION TESTING

Integration tests are designed to test integrated software components to determine if they actually run as one program. Testing is event driven and is more concerned with the basic outcome of screens or fields. Integration tests demonstrate that although the components were individually satisfaction, as shown by successfully unit testing, the combination of components is correct and consistent. Integration testing is specifically aimed at exposing the problems that arise from the combination of components.

7.3. VALIDATION TESTING

Tests were performed to find conformity with the requirements. Plans and procedures were designed to ensure that all functional requirements are satisfied. The software was alpha-tested there are two goals in preparing test plans. Firstly, a properly detailed test plan demonstrates that the program specifications are understood completely. Secondly, the test plan is used during program testing to prove the correctness of the program.

CHAPTER-8

SOURCE CODE AND OUTPUT SCREENSHOT

E-Voting_page.py

```
from tkinter import *
import tkinter as tk
from tkinter import PhotoImage
from tkinter import messagebox
import mysql.connector
import cv2
import face_recognition
import PIL.Image
import PIL.ImageTk
from time import sleep
root = tk.Tk()
root.title("Face Recognition Voting System")
root.geometry("925x500+300+200")
root.config(bg='#fff')
win_logo = PhotoImage(file="Image/poll.png")
root.iconphoto(win_logo, win_logo)
root.resizable(False, False)
login_icon = PhotoImage(file='Image/login.png')
passshow_icon = PhotoImage(file='Image/eye.png')
passhide_icon = PhotoImage(file='Image/hide.png')
db = mysql.connector.connect(host='localhost',
                             port=3306,
                             database='vote_details',
                             user='root',
                             password=")
cursor = db.cursor()
password_mode = True
login_photo = tk.Label(root, image=login_icon, bg='white')
login_photo.place(x=30, y=30)
login_frame = tk.Frame(root, width=350, height=420, bg='white')
signup_frame = tk.Frame(root, width=350, height=420, bg='white')
def change_signin():
    def voting_poll(name_of_person, aadhar_id):
            def update_Vote():
                already_vote = 0
                if vote.get() == 1:
                    BJP_Vote_1 = messagebox.askyesno("Verification", "Are you ready to poll
your vote for BJP?")
                    if BJP_Vote_1:
                        BJP_Vote_2 = messagebox.askyesno("Second Verification",
                                                         "Are you damn sure to submit your vote for BJP?")
                        if BJP_Vote_2:
                            with open("Election_Database.txt", "r+") as f:
                                vote_value = f.readlines()
                                total_vote = len(vote_value)
                                for i in range(total_vote):
                                    if vote_value[i] == f"{name_of_person}\n":
                                        already_vote = 1
                            with open("Election_Database.txt", "a+") as f:
                                if already_vote == 0:
                                    print(f"Successfully, {name_of_person} voted for BJP")
                                    f.write(f'{name_of_person}\n')
                                    f.write('voted for BJP\n')
                                    messagebox.showinfo("Success",
                                                        f"{name_of_person} Successfully voted for BJP")
                                    print("Voting Polled Successfully ...")
                                else:
                                    messagebox.showwarning("Warning", f"{name_of_person} already
voted ...")
                                    print(
                                        f"{name_of_person} already polled vote in the mentioned
candidates ...")
                elif vote.get() == 2:
                    Congress_Vote_1 = messagebox.askyesno("Verification", "Are you ready to
poll your vote for Congress?")
                    if Congress_Vote_1:
                        Congress_Vote_2 = messagebox.askyesno("Second Verification",
                                                              "Are you damn sure to submit your vote for
Congress?")
                        if Congress_Vote_2:
                            with open("Election_Database.txt", "r+") as f:
                                vote_value = f.readlines()
                                total_vote = len(vote_value)
                                for i in range(total_vote):
                                    if vote_value[i] == f"{name_of_person}\n":
                                        already_vote = 1
                            with open("Election_Database.txt", "a+") as f:
                                if already_vote == 0:
                                    print(f"Successfully, {name_of_person} voted for Congress")
                                    f.write(f'{name_of_person}\n')
                                    f.write('voted for Congress\n')
                                    messagebox.showinfo("Success",
                                                        f"{name_of_person} Successfully voted for Congress")
                                    print("Voting Polled Successfully ...")
                                else:
                                    messagebox.showwarning("Warning", f"{name_of_person} already
voted ...")
                                    print(
                                        f"{name_of_person} already polled vote in the mentioned
candidates ...")
                elif vote.get() == 3:
                    ADMK_Vote_1 = messagebox.askyesno("Verification", "Are you ready to
poll your vote for ADMK?")
                    if ADMK_Vote_1:
                        ADMK_Vote_2 = messagebox.askyesno("Second Verification",
                                                          "Are you damn sure to submit your vote for
ADMK?")
                        if ADMK_Vote_2:
                            with open("Election_Database.txt", "r+") as f:
                                vote_value = f.readlines()
                                total_vote = len(vote_value)
                                for i in range(total_vote):
                                    if vote_value[i] == f"{name_of_person}\n":
                                        already_vote = 1
                            with open("Election_Database.txt", "a+") as f:
                                if already_vote == 0:
                                    print(f"Successfully, {name_of_person} voted for ADMK")
                                    f.write(f'{name_of_person}\n')
                                    f.write('voted for ADMK\n')
                                    messagebox.showinfo("Success",
                                                        f"{name_of_person} Successfully voted for ADMK")
                                    print("Voting Polled Successfully ...")
                                else:
                                    messagebox.showwarning("Warning", f"{name_of_person} already
voted ...")
                                    print(
                                        f"{name_of_person} already polled vote in the mentioned
candidates ...")
                elif vote.get() == 4:
                    DMK_Vote_1 = messagebox.askyesno("Verification", "Are you ready to poll
your vote for DMK?")
                    if DMK_Vote_1:
                        DMK_Vote_2 = messagebox.askyesno("Second Verification",
                                                         "Are you damn sure to submit your vote for DMK?")
                        if DMK_Vote_2:
                            with open("Election_Database.txt", "r+") as f:
                                vote_value = f.readlines()
                                print(vote_value)
                                total_vote = len(vote_value)
                                for i in range(total_vote):
                                    if vote_value[i] == f"{name_of_person}\n":
                                        already = 1
                            with open("Election_Database.txt", "a+") as f:
                                print(already)
                                if already == 0:
                                    f.write(f'{name_of_person}\n')
                                    f.write('voted for DMK\n')
                                    print("Voting Polled Successfully ...")
                                else:
                                    messagebox.showwarning("Warning", f"{name_of_person} already
voted")
                                    print(
                                        f"{name_of_person} already polled vote in the mentioned
candidates ...")
                else:
                    messagebox.showerror("Empty Poll",
                                         f"{name_of_person}, You are not selected any candidate from
the list")
                    print("Some Problem Occurs ...")
            vote_window = Tk()
            vote_window.title("Face Recognition Voting System")
            win_logo = PhotoImage(file="Image/poll.png")
            vote_window.iconphoto(win_logo, win_logo)
            vote_window.geometry("720x480")
            vote_window.config(bg="gray")
            vote_window.resizable(False, False)
            bg_img = PIL.Image.open('Image/back.png').resize((720, 480))
            bg_img = PIL.ImageTk.PhotoImage(bg_img)
            background_img = Label(vote_window, image=bg_img)
            background_img.pack()
            project_title = Label(vote_window, text="FACE RECOGNITION VOTING
SYSTEM",
                                  font=('poppins', 18, 'bold'))
            project_title.config(bg='#d5d9db')
            project_title.place(x=50, y=15)
            login_details = Label(vote_window, text=f"Welcome,
{name_of_person}\n({aadhar_id})",
                                  font=('poppins', 8, 'bold'))
            login_details.config(bg='#d5d9db')
            login_details.place(x=550, y=10)
            question_lable = Label(vote_window,
                                   text="Who do you want to vote in below mentioned parties ...")
            question_lable.config(bg='#d5d9db', font=('poppins', 13))
            question_lable.place(x=30, y=100)
def change_signup():
    login_frame.place_forget()
    signup_frame.place(x=480, y=50)
    login_head = tk.Label(signup_frame, text='Voter Register Page', fg='#57a1f8',
bg='white',
                          font=('Microsoft YaHei UI Light', 23, 'bold'))
    login_head.place(x=30, y=5)
    def signup():
        username = register_user_entry.get()
        password = register_pass_entry.get()
        aadhaar_id = register_staffid_entry.get()
        phone_no = register_phone_entry.get()
        if username == 'Voter Name' or password == 'Password' or aadhaar_id == 'Aadhaar
ID' or phone_no == 'Phone Number':
            messagebox.showwarning("Warning", "Please enter missing elements to proceed
voter register !!!")
        elif len(aadhaar_id) < 12 or len(aadhaar_id) > 12:
            messagebox.showerror("Aadhaar Error", "Please provide correct Aadhar ID ...")
        elif len(phone_no) < 10 or len(phone_no) > 10:
            messagebox.showerror("Phone Number Error", "Please provide Phone number
correctly ...")
        else:
            cursor.execute("insert into voter_content values(%s,%s,%s,%s)", (username,
aadhaar_id, phone_no, password))
            db.commit()
            messagebox.showinfo("Success", "Registered Successfully!!")
    def user_on_enter(e):
        register_user_entry.delete(0, 'end')
    def user_on_leave(e):
        register_username = register_user_entry.get()
        if register_username == '':
            register_user_entry.insert(0, 'Voter Name')
    register_user_entry = tk.Entry(signup_frame, width=25, fg='black', border=0,
bg='white',
                          font=('Microsoft YaHei UI Light', 11))
    register_user_entry.place(x=30, y=80)
    register_user_entry.insert(0, 'Voter Name')
    register_user_entry.bind('<FocusIn>', user_on_enter)
    register_user_entry.bind('<FocusOut>', user_on_leave)
    tk.Frame(signup_frame, width=295, height=2, bg='black').place(x=25, y=110)
    def staffid_on_enter(e):
        register_staffid_entry.delete(0, 'end')
    def staffid_on_leave(e):
        register_staffid = register_staffid_entry.get()
        if register_staffid == '':
            register_staffid_entry.insert(0, 'Aadhaar ID')
    register_staffid_entry = tk.Entry(signup_frame, width=25, fg='black', border=0,
bg='white',
                        font=('Microsoft YaHei UI Light', 11))
    register_staffid_entry.place(x=30, y=140)
    register_staffid_entry.insert(0, 'Aadhaar ID')
    register_staffid_entry.bind('<FocusIn>', staffid_on_enter)
    register_staffid_entry.bind('<FocusOut>', staffid_on_leave)
    tk.Frame(signup_frame, width=295, height=2, bg='black').place(x=25, y=170)
    def phone_on_enter(e):
        register_phone_entry.delete(0, 'end')
    def phone_on_leave(e):
        register_phone = register_phone_entry.get()
        if register_phone == '':
            register_phone_entry.insert(0, 'Phone Number')
    register_phone_entry = tk.Entry(signup_frame, width=25, fg='black', border=0,
bg='white',
                        font=('Microsoft YaHei UI Light', 11))
    register_phone_entry.place(x=30, y=200)
    register_phone_entry.insert(0, 'Phone Number')
    register_phone_entry.bind('<FocusIn>', phone_on_enter)
    register_phone_entry.bind('<FocusOut>', phone_on_leave)
    tk.Frame(signup_frame, width=295, height=2, bg='black').place(x=25, y=230)
    def pass_on_enter(e):
        register_pass_entry.delete(0, 'end')
        register_pass_entry.config(show='*')
        register_pass_show.config(image=passhide_icon)
    def pass_on_leave(e):
        register_pass_entry.config(show='')
        register_pass_entry.insert(0, 'Password')
        register_pass_show.config(fg='#fff', image='')
    def passshow():
        global password_mode
        if password_mode:
            register_pass_show.config(image=passshow_icon)
            register_pass_entry.config(show='')
            password_mode = False
        else:
            register_pass_show.config(image=passhide_icon)
            register_pass_entry.config(show='*')
            password_mode = True
    register_pass_entry = tk.Entry(signup_frame, width=25, fg='black', border=0,
bg='white',
                                   font=('Microsoft YaHei UI Light', 11))
    register_pass_entry.place(x=30, y=260)
    register_pass_entry.insert(0, 'Password')
    register_pass_entry.bind('<FocusIn>', pass_on_enter)
    register_pass_entry.bind('<FocusOut>', pass_on_leave)
    register_pass_show = tk.Button(signup_frame, image='', border=0, bg='#fff',
activebackground='#fff',
                                   command=passshow)
    register_pass_show.place(x=280, y=250, width=50, height=50)
    tk.Frame(signup_frame, width=295, height=2, bg='black').place(x=25, y=290)
    signup_btn = tk.Button(signup_frame, width=37, pady=3, text='Sign up', bg='#57a1f8',
fg='white', border=0,
                          command=signup,
                          font=('Microsoft YaHei UI Light', 10))
    signup_btn.place(x=25, y=325)
    signup_label = tk.Label(signup_frame, text="Already have an account?", fg='black',
bg='white',
                           font=('Microsoft YaHei UI Light', 9))
    signup_label.place(x=65, y=375)
    signup_btn = tk.Button(signup_frame, width=6, text='Sign In', border=0, bg='white',
cursor='hand2', fg='#57a1f8',
                          activebackground='white', activeforeground='#57a1f8',
command=change_signin)
    signup_btn.place(x=215, y=375)
change_signin()
root.mainloop()
```

Fingerprint_module.py

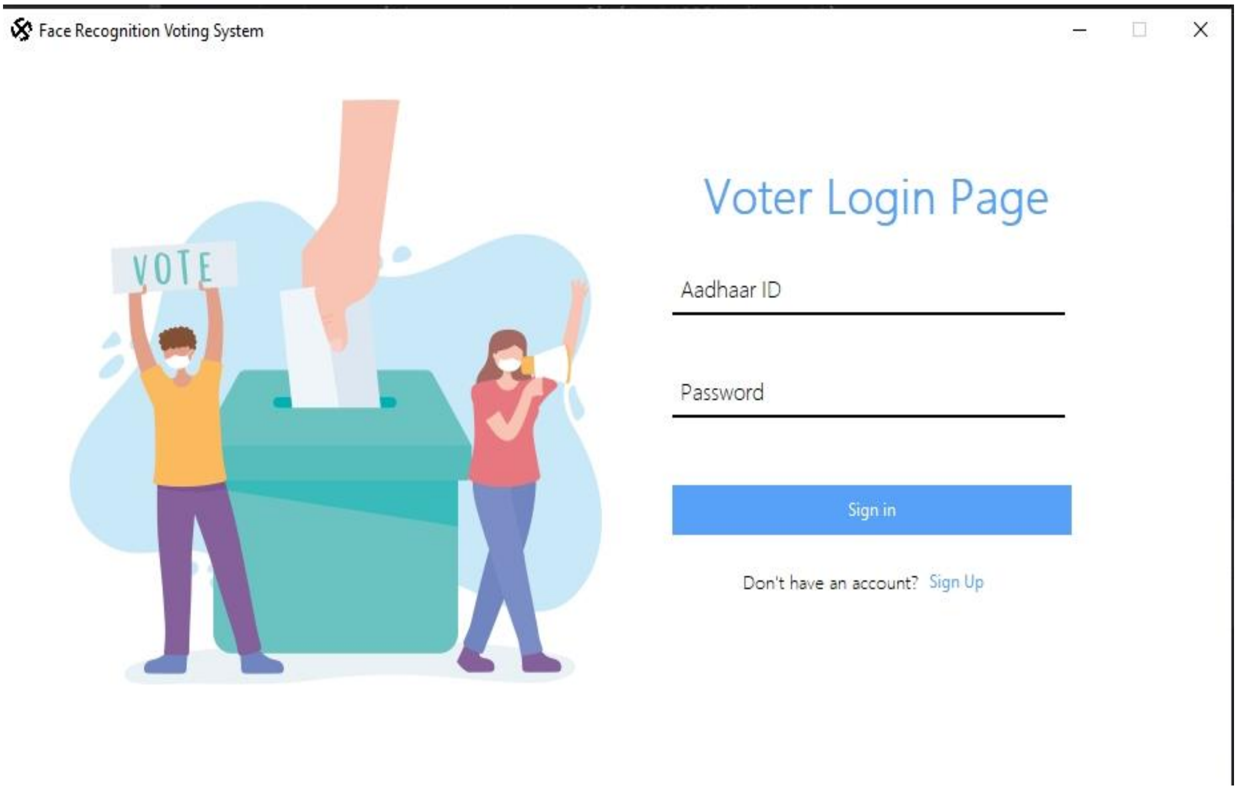
```
import os
import cv2
import numpy as np
def isIdentical():
    best_score = []
    filename = None
    image = None
    kp1, kp2, mp = None, None, None
    dp1, kp2 = None, None

os.system('E:/Fingerprint/Mantra/MFS100/Driver/MFS100Test/MANTRA.MFS100.Test.
exe')
    for i in range(1, len(os.listdir('Authorised'))+1):
        score = 0
        finger_print_one =
cv2.imread("Mantra/MFS100/Driver/MFS100Test/FingerData/FingerImage.bmp")
        finger_print_two = cv2.imread(f"Authorised/fingerprint_{i}.bmp")
        resized_1 = cv2.resize(finger_print_one, None, fx=4, fy=4)
        resized_2 = cv2.resize(finger_print_two, None, fx=4, fy=4)
        sift = cv2.SIFT_create()
        kp1, dp1 = sift.detectAndCompute(resized_1, None)
        kp2, dp2 = sift.detectAndCompute(resized_2, None)
        FLANN_INDEX_KDTREE = 0
        index_params = dict(algorithm=FLANN_INDEX_KDTREE, trees=5)
        search_params = dict(checks=50)
        flann = cv2.FlannBasedMatcher(index_params, search_params)
        matches = flann.knnMatch(np.asarray(dp1, np.float32), np.asarray(dp2, np.float32),
k=2)
        match_points = []
        for p, q in matches:
            if p.distance < 0.5 * q.distance:
                match_points.append(p)
        keypoints = 0
        if len(kp1) < len(kp2):
            keypoints = len(kp1)
        else:
            keypoints = len(kp2)
        if len(match_points) / keypoints * 100 > score:
            score = len(match_points) / keypoints * 100
            image = resized_2
            kp1, kp2, mp = kp1, kp2, match_points
            best_score.append(round(score))
            print(best_score)
        else:
            best_score.append(0)
            print(best_score)
    print("SCORE: " + str(max(best_score)))
isIdentical()

 import os
import cv2 as cv
count = 1
while True:

os.system('E:/Fingerprint/Mantra/MFS100/Driver/MFS100Test/MANTRA.MFS100.Test.
exe')
    finger =
cv.imread('E:/Fingerprint/Mantra/MFS100/Driver/MFS100Test/FingerData/FingerImage.b
mp')
    cv.imwrite('Authorised/fingerprint_{}.bmp'.format(count), finger)
    cv.imshow('Scanned Fingerprint', finger)
    count += 1
    if cv.waitKey(0) == ord(' '):
        break
print(f"Number of Fingerprints Saved: {len(os.listdir('Authorised'))}")
[4:04 PM, 5/5/2024] Màdháñ: from tkinter import *
import tkinter as tk
from tkinter import PhotoImage
from tkinter import messagebox
import PIL.Image
import PIL.ImageTk
admin_window = Tk()
admin_window.title("Face Recognition Voting System")
admin_window.geometry("720x480")
admin_window.config(bg='white')
admin_window.resizable(False, False)
admin_image = PhotoImage(file='Image/admin.png')
def Total_Poll():
    BJP_Poll = 0
    Congress_Poll = 0
    ADMK_Poll = 0
    DMK_Poll = 0
    with open("Election_database.txt", 'r+') as f:
        poll_list = f.readlines()
        for i in poll_list:
            if "voted for BJP" in i:
                BJP_Poll += 1
            if "voted for Congress" in i:
                Congress_Poll += 1
            if "voted for ADMK" in i:
                ADMK_Poll += 1
            if "voted for DMK" in i:
                DMK_Poll += 1
    BJP_count.config(text=BJP_Poll)
    Congress_count.config(text=Congress_Poll)
    ADMK_count.config(text=ADMK_Poll)
    DMK_count.config(text=DMK_Poll)
admin_label = Label(admin_window, text="Admin Module - Total Poll Vote Count",
                                    font=('poppins', 20, 'bold'), bg='white', fg="#de5f3c")
admin_label.place(x=108, y=30)
admin_logo = Label(admin_window, image=admin_image, bg='white')
admin_logo.place(x=10, y=80)
BJP_lable = Label(admin_window, text="BJP", font=('poppins', 18, 'bold'), bg='white',
fg="#ad444c")
BJP_lable.place(x=420, y=110)
arrow_1 = Label(admin_window, text="=>", font=('poppins', 18, 'bold'), bg='white',
fg="#ad444c")
arrow_1.place(x=560, y=110)
BJP_count = Label(admin_window, text='0', font=('poppins', 18, 'bold'), bg='white',
fg="#ad444c")
BJP_count.place(x=640, y=108)
Congress_lable = Label(admin_window, text="Congress", font=('poppins', 18, 'bold'),
bg='white', fg="#ad444c")
Congress_lable.place(x=420, y=160)
arrow_2 = Label(admin_window, text="=>", font=('poppins', 18, 'bold'), bg='white',
fg="#ad444c")
arrow_2.place(x=560, y=160)
Congress_count = Label(admin_window, text='0', font=('poppins', 18, 'bold'), bg='white',
fg="#ad444c")
Congress_count.place(x=640, y=158)
ADMK_lable = Label(admin_window, text="ADMK", font=('poppins', 18, 'bold'),
bg='white', fg="#ad444c")
ADMK_lable.place(x=420, y=210)
arrow_3 = Label(admin_window, text="=>", font=('poppins', 18, 'bold'), bg='white',
fg="#ad444c")
arrow_3.place(x=560, y=210)
ADMK_count = Label(admin_window, text='0', font=('poppins', 18, 'bold'), bg='white',
fg="#ad444c")
ADMK_count.place(x=640, y=208)
DMK_lable = Label(admin_window, text="DMK", font=('poppins', 18, 'bold'), bg='white',
fg="#ad444c")
DMK_lable.place(x=420, y=260)
arrow_4 = Label(admin_window, text="=>", font=('poppins', 18, 'bold'), bg='white',
fg="#ad444c")
arrow_4.place(x=560, y=260)
DMK_count = Label(admin_window, text='0', font=('poppins', 18, 'bold'), bg='white',
fg="#ad444c")
DMK_count.place(x=640, y=258)
refresh_count = Button(admin_window, text="Refresh Poll", font=('poppins', 16, 'bold'),
command=Total_Poll)
refresh_count.config(width=18, height=1, bg='black', fg='white',
activebackground="black", activeforeground='white')
refresh_count.place(x=420, y=320)
admin_window.mainloop()
```

Admin_module.py

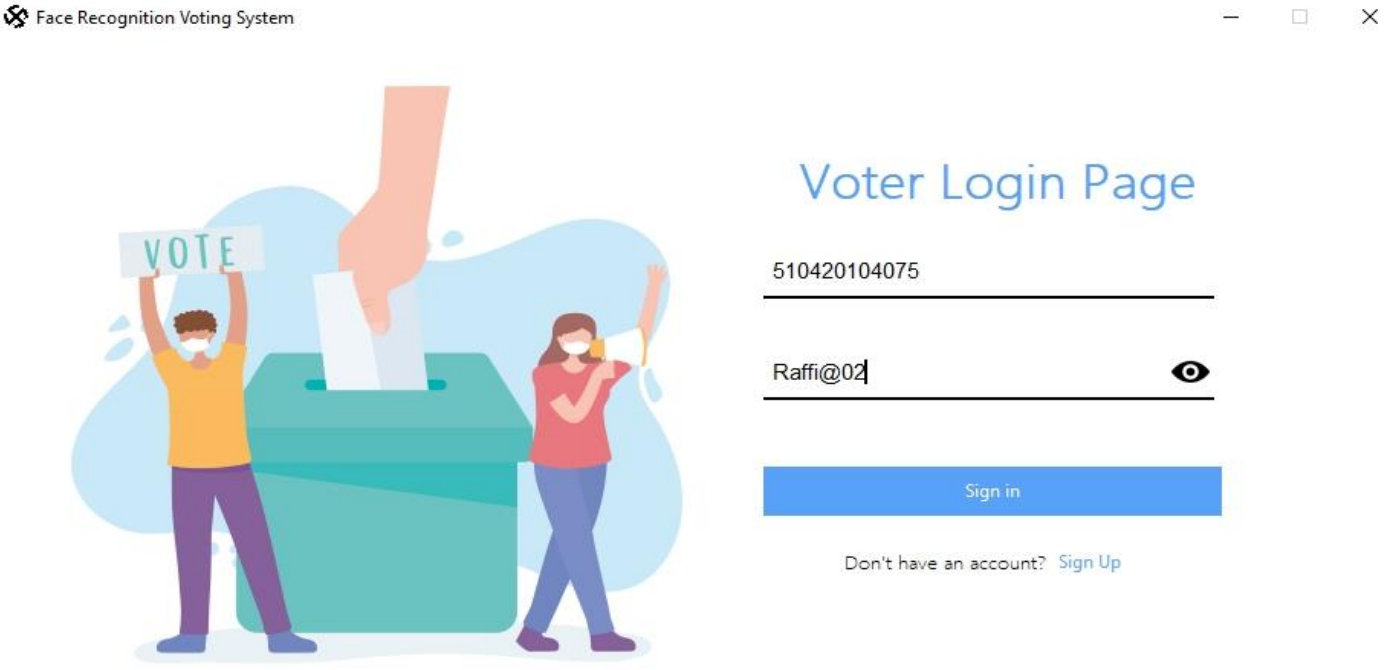
```
from tkinter import *
import tkinter as tk
from tkinter import PhotoImage
from tkinter import messagebox
import PIL.Image
import PIL.ImageTk
admin_window = Tk()
admin_window.title("Face Recognition Voting System")
admin_window.geometry("720x480")
admin_window.config(bg='white')
admin_window.resizable(False, False)
admin_image = PhotoImage(file='Image/admin.png')
```

```
def Total_Poll():
    BJP_Poll = 0
    Congress_Poll = 0
    ADMK_Poll = 0
    DMK_Poll = 0
    with open("Election_database.txt", 'r+') as f:
        poll_list = f.readlines()
        for i in poll_list:
            if "voted for BJP" in i:
                BJP_Poll += 1
            if "voted for Congress" in i:
                Congress_Poll += 1
            if "voted for ADMK" in i:
                ADMK_Poll += 1
            if "voted for DMK" in i:
                DMK_Poll += 1
    BJP_count.config(text=BJP_Poll)
    Congress_count.config(text=Congress_Poll)
    ADMK_count.config(text=ADMK_Poll)
    DMK_count.config(text=DMK_Poll)
admin_label = Label(admin_window, text="Admin Module - Total Poll Vote Count",
                                    font=('poppins', 20, 'bold'), bg='white', fg="#de5f3c")
admin_label.place(x=108, y=30)
admin_logo = Label(admin_window, image=admin_image, bg='white')
admin_logo.place(x=10, y=80)
BJP_lable = Label(admin_window, text="BJP", font=('poppins', 18, 'bold'), bg='white',
fg="#ad444c")
BJP_lable.place(x=420, y=110)
arrow_1 = Label(admin_window, text="=>", font=('poppins', 18, 'bold'), bg='white',
fg="#ad444c")
arrow_1.place(x=560, y=110)
BJP_count = Label(admin_window, text='0', font=('poppins', 18, 'bold'), bg='white',
fg="#ad444c")
BJP_count.place(x=640, y=108)
Congress_lable = Label(admin_window, text="Congress", font=('poppins', 18, 'bold'),
bg='white', fg="#ad444c")
Congress_lable.place(x=420, y=160)
arrow_2 = Label(admin_window, text="=>", font=('poppins', 18, 'bold'), bg='white',
fg="#ad444c")
arrow_2.place(x=560, y=160)
Congress_count = Label(admin_window, text='0', font=('poppins', 18, 'bold'), bg='white', fg="#ad444c")
Congress_count.place(x=640, y=158)
ADMK_lable = Label(admin_window, text="ADMK", font=('poppins', 18, 'bold'), bg='white', fg="#ad444c")
ADMK_lable.place(x=420, y=210)
arrow_3 = Label(admin_window, text="=>", font=('poppins', 18, 'bold'), bg='white', fg="#ad444c")
arrow_3.place(x=560, y=210)
ADMK_count = Label(admin_window, text='0', font=('poppins', 18, 'bold'), bg='white', fg="#ad444c")
ADMK_count.place(x=640, y=208)
DMK_lable = Label(admin_window, text="DMK", font=('poppins', 18, 'bold'), bg='white', fg="#ad444c")
DMK_lable.place(x=420, y=260)
arrow_4 = Label(admin_window, text="=>", font=('poppins', 18, 'bold'), bg='white', fg="#ad444c")
arrow_4.place(x=560, y=260)
DMK_count = Label(admin_window, text='0', font=('poppins', 18, 'bold'), bg='white', fg="#ad444c")
DMK_count.place(x=640, y=258)
refresh_count = Button(admin_window, text="Refresh Poll", font=('poppins', 16, 'bold'), command=Total_Poll)
refresh_count.config(width=18, height=1, bg='black', fg='white', activebackground="black", activeforeground='white')
refresh_count.place(x=420, y=320)
admin_window.mainloop()
```

OUTPUT SCREENSHOT



Face Recognition Voting System
VOTE
Voter Login Page
Aadhaar ID
Password
Sign in
Don't have an account? Sign Up

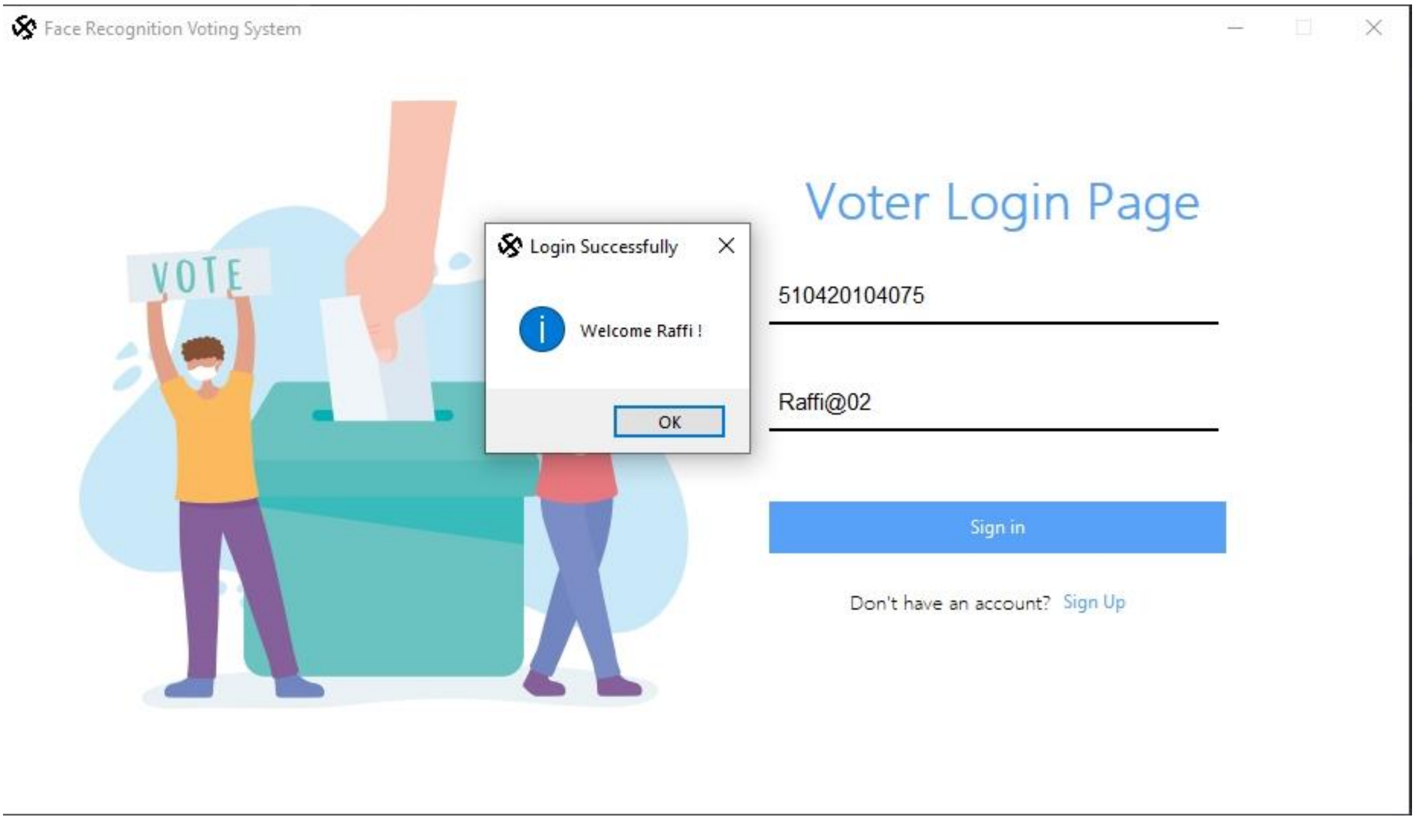


Face Recognition Voting System
VOTE
Voter Login Page

510420104075

Raffi@02

Sign in
Don't have an account? Sign Up



Face Recognition Voting System

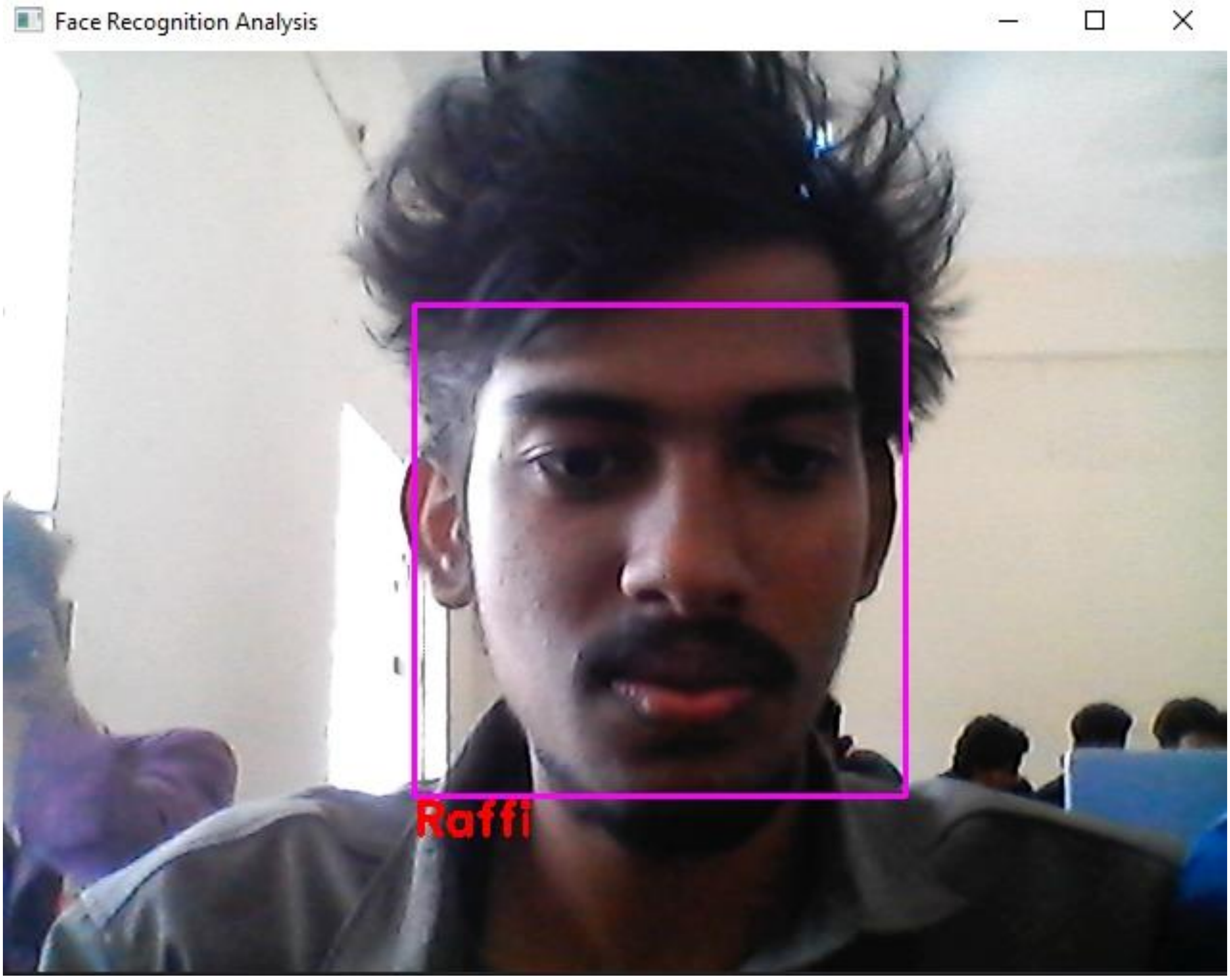
Voter Login Page

510420104075

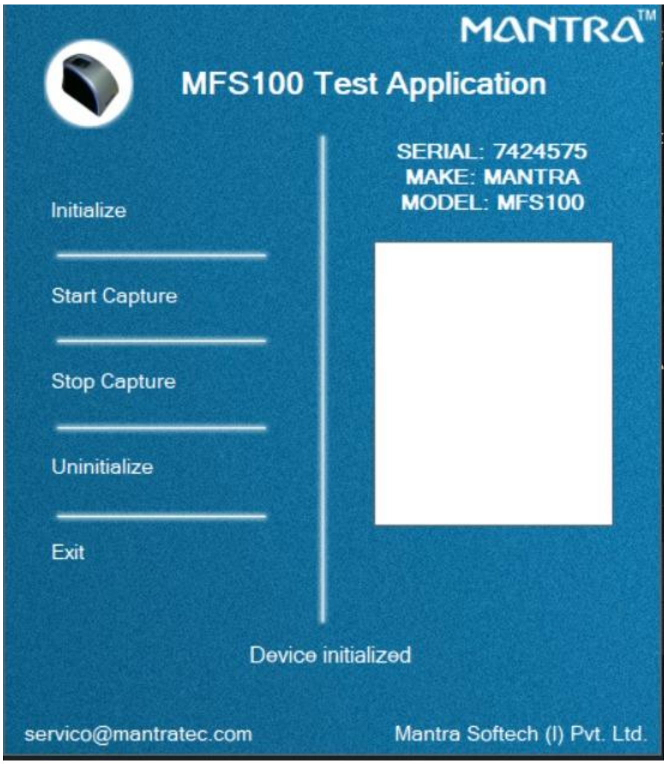
Raffi@02

Login Successfully

Welcome Raffi !



Face Recognition Analysis

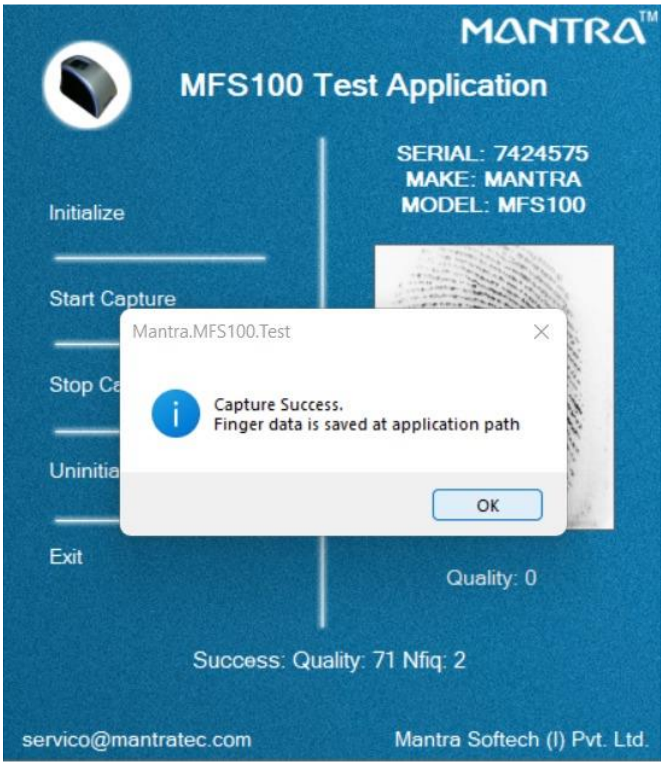


SERIAL: 7424575
MAKE: MANTRA
MODEL: MFS100

Device initialized

servico@mantratec.com

Mantra Softech (I) Pvt. Ltd.



SERIAL: 7424575
MAKE: MANTRA
MODEL: MFS100

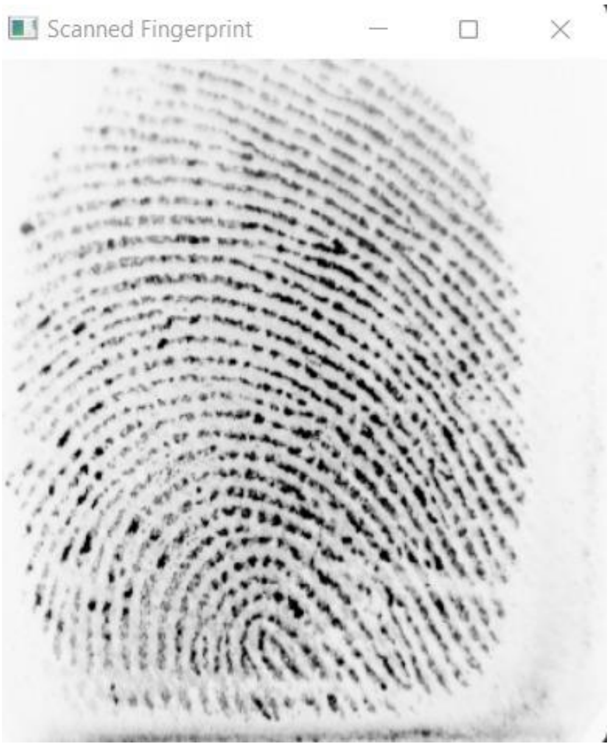
Capture Success.
Finger data is saved at application path

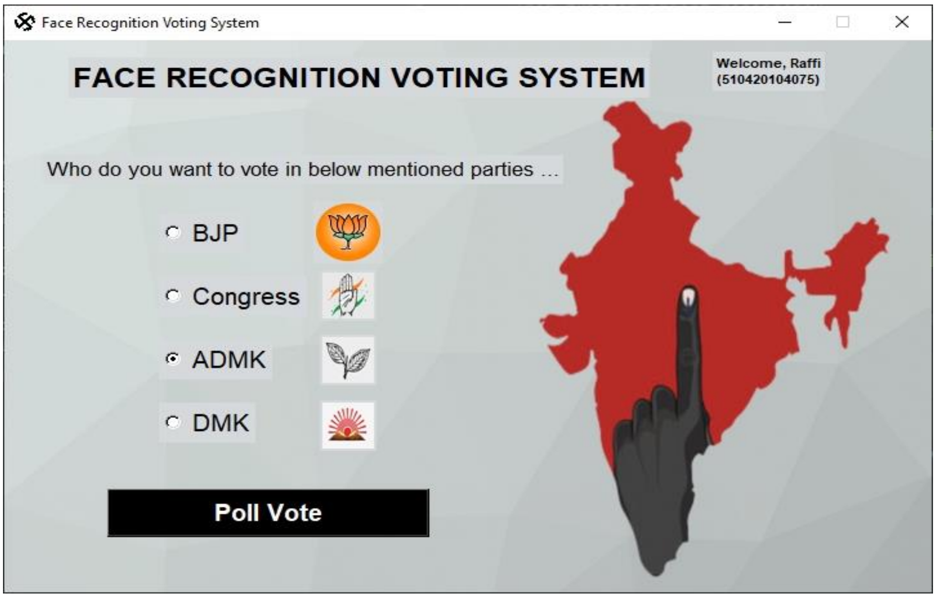
Quality: 0

Success: Quality: 71 Nfiq: 2

servico@mantratec.com

Mantra Softech (I) Pvt. Ltd.





Face Recognition Voting System

FACE RECOGNITION VOTING SYSTEM

Welcome, Raffi
(510420104075)

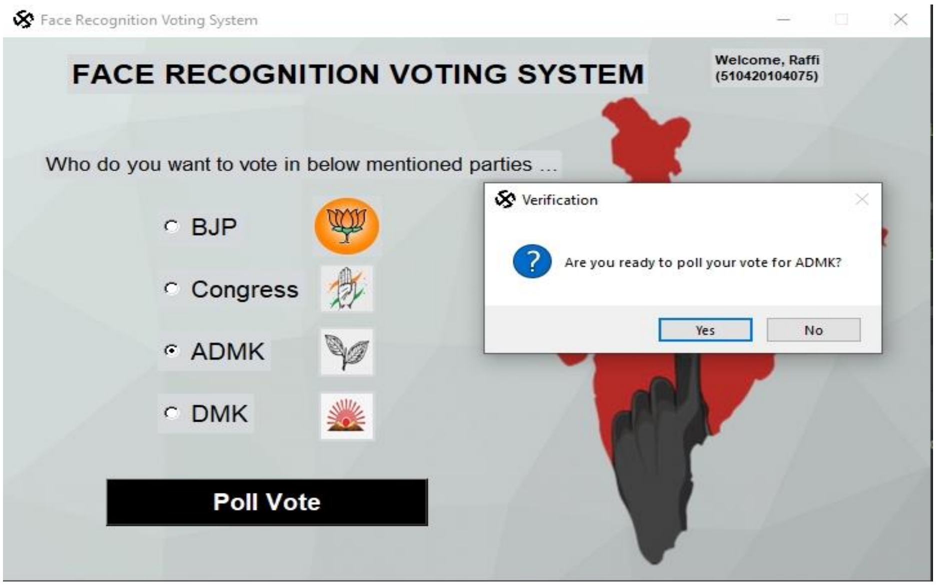
Who do you want to vote in below mentioned parties ...

○ BJP

○ Congress

● ADMK

○ DMK



Face Recognition Voting System

FACE RECOGNITION VOTING SYSTEM

Welcome, Raffi
(510420104075)

Who do you want to vote in below mentioned parties ...

○ BJP

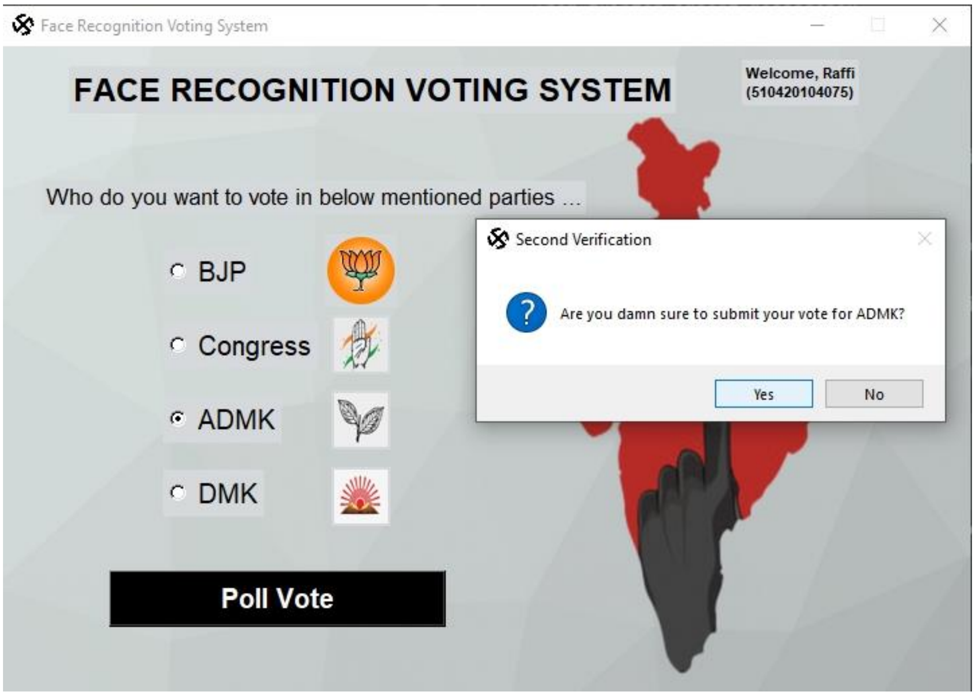
○ Congress

● ADMK

○ DMK

Verification

Are you ready to poll your vote for ADMK?



Face Recognition Voting System

FACE RECOGNITION VOTING SYSTEM

Welcome, Raffi
(510420104075)

Who do you want to vote in below mentioned parties ...

○ BJP

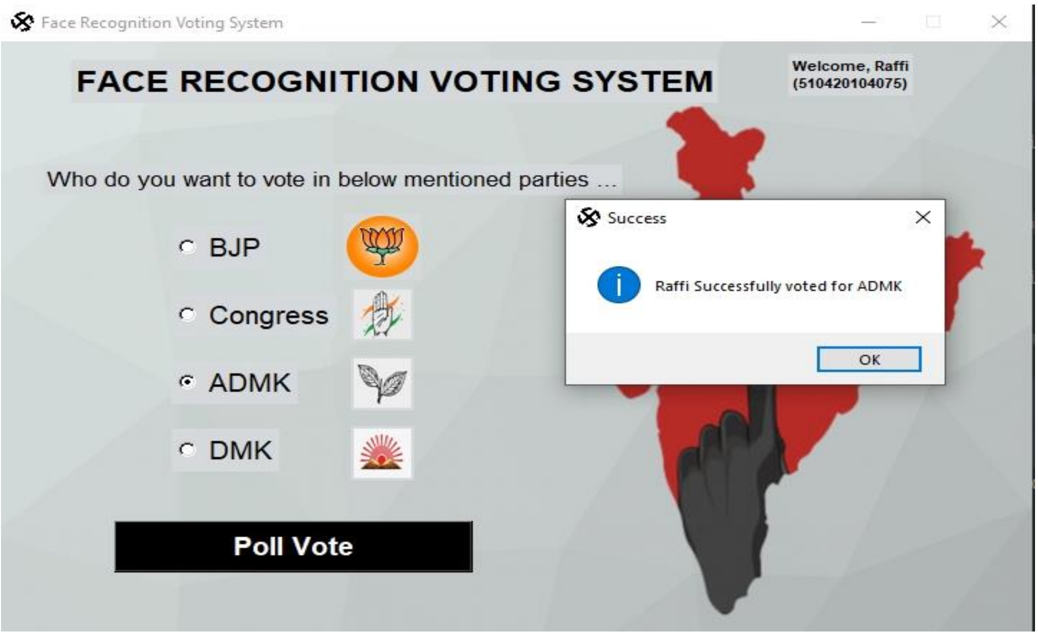
○ Congress

● ADMK

○ DMK

Second Verification

Are you damn sure to submit your vote for ADMK?



Face Recognition Voting System

FACE RECOGNITION VOTING SYSTEM

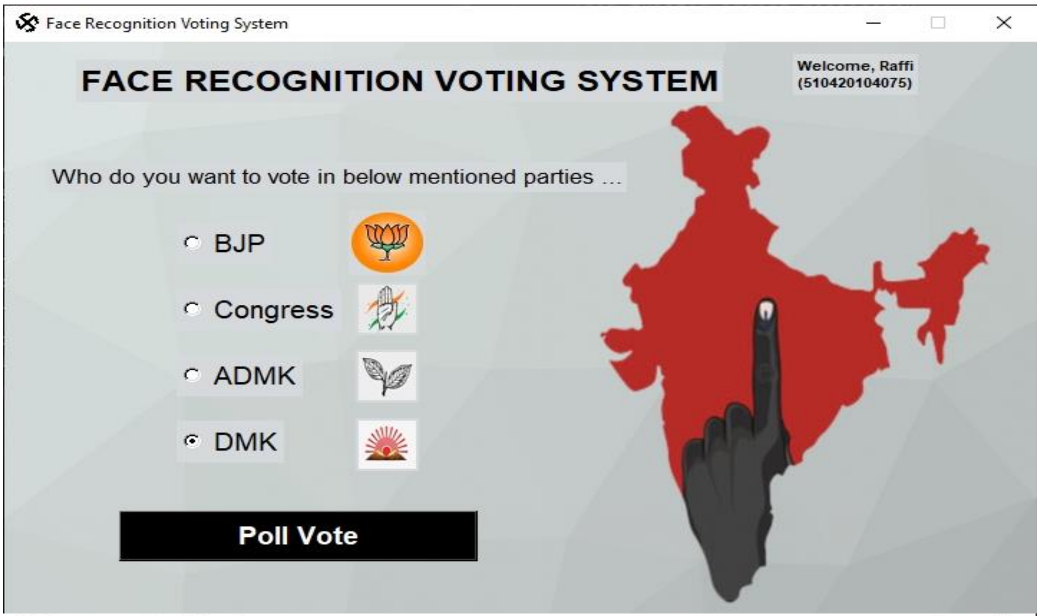
Welcome, Raffi
(510420104075)

Who do you want to vote in below mentioned parties ...

○ BJP
○ Congress
● ADMK
○ DMK

Success

Raffi Successfully voted for ADMK



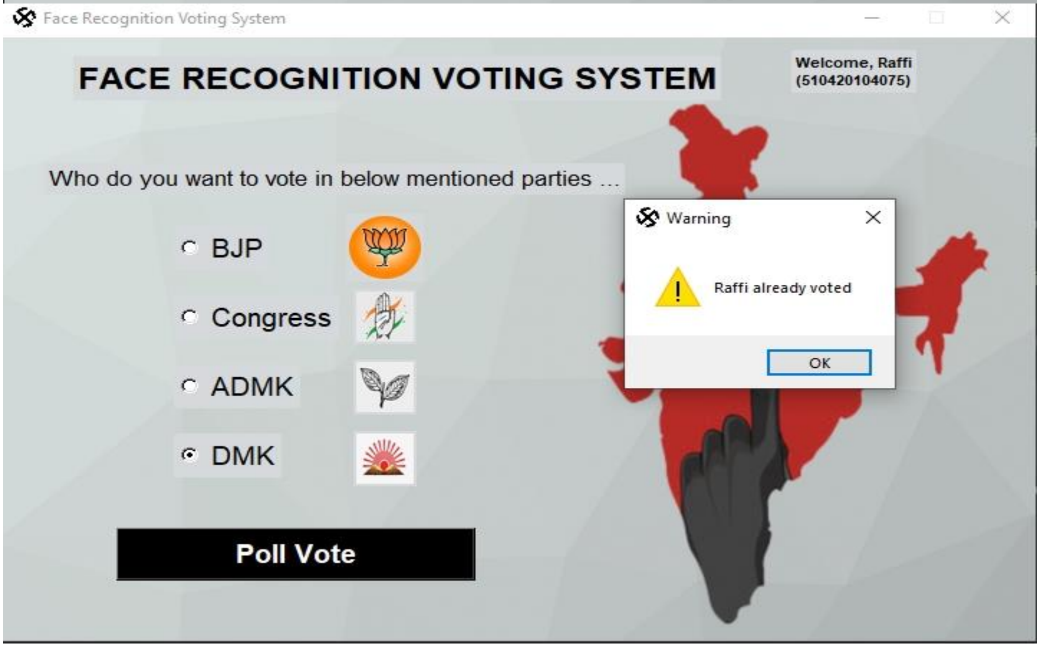
Face Recognition Voting System

FACE RECOGNITION VOTING SYSTEM

Welcome, Raffi
(510420104075)

Who do you want to vote in below mentioned parties ...

○ BJP
○ Congress
○ ADMK
● DMK



Face Recognition Voting System

FACE RECOGNITION VOTING SYSTEM

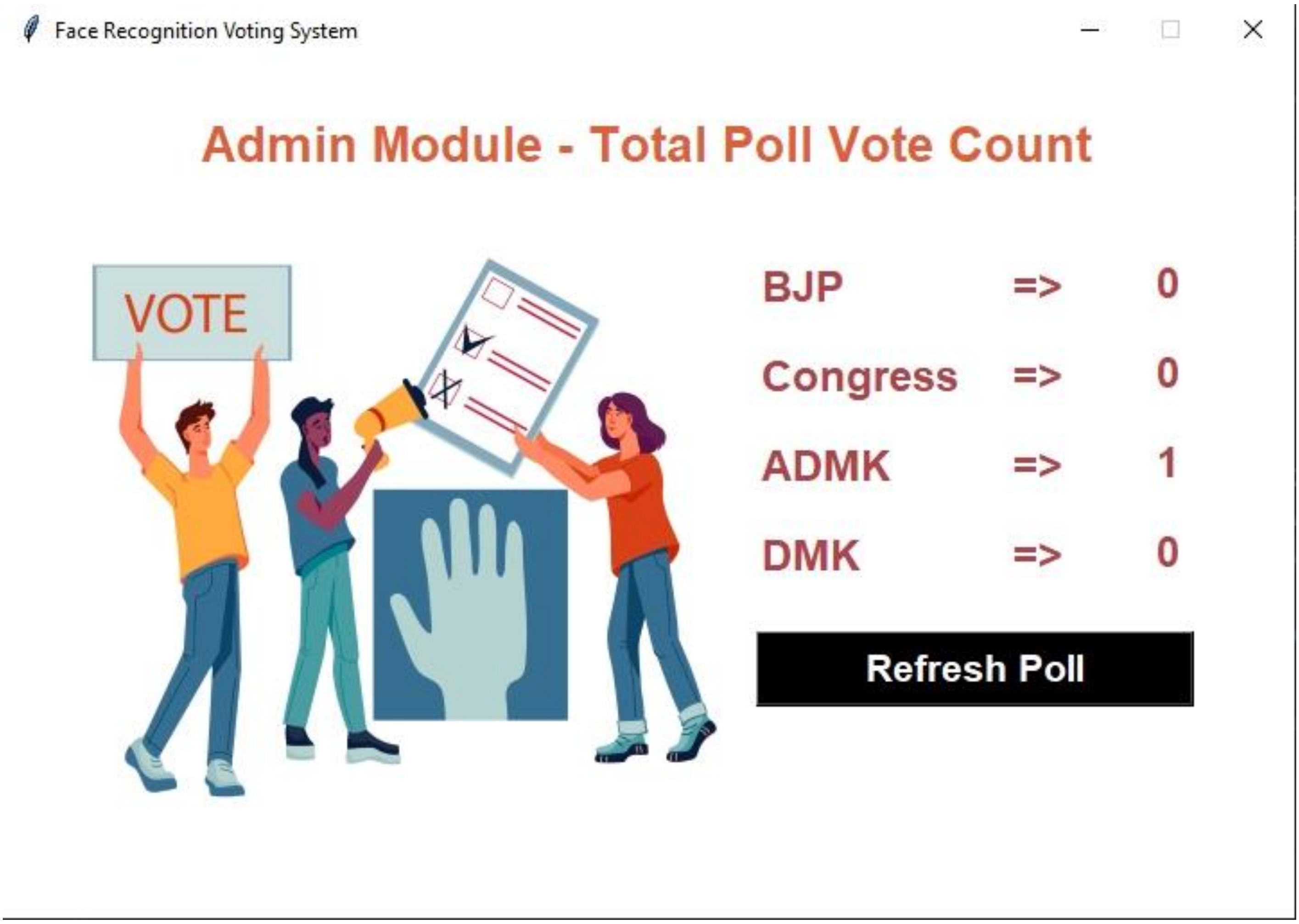
Welcome, Raffi
(510420104075)

Who do you want to vote in below mentioned parties ...

○ BJP
○ Congress
○ ADMK
● DMK

Warning

Raffi already voted



Face Recognition Voting System

Admin Module - Total Poll Vote Count

BJP => 0

Congress => 0

ADMK => 1

DMK => 0

CHAPTER-9

CONCLUSION & FUTURE WORK

9.1. CONCLUSION

A Secure E-voting system uses face recognition and fingerprint methods provide additional security has been discussed in this paper. As per our study and discussion,Eigen faces algorithm is superlative for face recognition because it provides high accuracy than other algorithm discussed in paper.

9.2. FUTURE WORK

The implemented system focuses on avoiding fake and duplicate voting by providing additional security. In future system can be built on android platform for online voting using same techniques. In this project, we have considered only face recognition and fingerprint. In future, it can be possible to use OTP method for voting.

CHAPTER-10

REFERENCES

1. Benaloh, J., & Tuinstra, D. (1994). Receipt-free secret-ballot elections. In Advances in Cryptology—EUROCRYPT'94 (pp. 393-403). Springer Berlin Heidelberg.

2. Chaum, D. (1981). Untraceable electronic mail, return addresses, and digital pseudonyms. Communications of the ACM, 24(2), 84-90.

3. Fu, K., Blum, A., Hopper, N., & Langford, J. (2008). A novel scheme for preventing replay attacks. In Proceedings of the 2008 ACM symposium on Information, computer and communications security (pp. 23-34).

4. Simons, B. (2002). Secure electronic voting. Information Security Technical Report, 7(4), 39-46.

5. Ryan, P. Y., Schneider, S., & Teague, V. (2009). The pretty good democracy. In International Conference on Financial Cryptography and Data Security (pp. 122-136). Springer, Berlin, Heidelberg.

6. Smith, W. R. (2005). Electronic voting system with verifiable audit trail. U.S. Patent No. 6,874,855.

7. Rivest, R. L. (2006). On the notion of “software independence” in voting systems. In Proceedings of the USENIX/Accurate Electronic Voting Technology Workshop (Vol. 7).

8. Sako, K., & Kilian, J. (1995). Receipt-free mix-type voting scheme—a practical solution to the implementation of a voting booth. In Advances in Cryptology—CRYPTO'94 (pp. 393-403). Springer Berlin Heidelberg.

9. Karlof, C., & Wagner, D. (2002). Secure audit logging for computer forensic analysis. In Proceedings of the 2003 ACM SIGSAC conference on Computer & communications security (pp. 171-180).

10. Bernhard, M., Pereira, O., & Schneider, S. (2012). In Pretty-Bad-Proxy: Cache-Based Side-Channel Attacks on the Web. In Proceedings of the 2012 ACM Conference on Computer and Communications Security (pp. 779-790).