ONLINE VOTING SYSTEM USING FACIAL RECOGNITION

Minor Project submitted in partial fulfillment of the requirement for the award of the degree of

BACHELOR OF TECHNOLOGY

IN

COMPUTER SCIENCE AND ENGINEERING

Under the esteemed guidance of

DR. K. AMBEDKAR
ASSOCIATE PROFESSOR

By

VUDHANTHI NEERAJA 19R11A0548
VANKAYALAPATI VINITHA 19R11A0544
KAMUJU NOVA 19R11A0517

Department of Computer Science and Engineering
Accredited by NBA

Geethanjali College of Engineering and Technology
(UGC Autonomous)
(Affiliated to J.N.T.U.H, Approved by AICTE, New Delhi)]
Cheeryal (V), Keesara (M),
Medchal.Dist.-501 301.
November-2022

Geethanjali College of Engineering & Technology

(UGC Autonomous)

(Affiliated to JNTUH, Approved by
AICTE,New Delhi)Cheeryal (V),
Keesara(M), Medchal Dist.-501 301.

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
Accredited by NBA

CERTIFICATE

This is to certify that the B.Tech Minor Project report entitled **"ONLINE VOTING SYSTEM USING FACIAL RECOGNITION"** is a bonafide work done by **Vudhanthi Neeraja (19R11A0548), Vankayalapati Vinitha (19R11A0544), Kamuju Nova (19R11A0517)** in partial fulfillment of the requirement of the award for the degree of Bachelor of Technology in "**Computer Science and Engineering**" from Jawaharlal Nehru Technological University, Hyderabad during the year 2022-2023.

Internal Guide
Dr.K.Ambedkar
Associate Professor

HOD-CSE
Dr. Sree Lakshmi
Professor

External Examiner

Geethanjali College of Engineering & Technology

(UGC Autonomous)
(Affiliated to JNTUH Approved by
AICTE,New Delhi)Cheeryal (V),
Keesara(M), Medchal Dist.-501 301.

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

Accredited by NBA

DECLARATION BY THE CANDIDATE

We, **Vudhanthi Neeraja, Vankayalapati Vinitha, Kamuju Nova**, bearing Roll Nos. **19R11A0548 ,19R11A0544 ,19R11A0517** , hereby declare that the project report entitled **"ONLINE VOTING SYSTEM USING FACIAL RECOGNITION"** is done under the guidance of **Dr.K.Ambedkar, Associate Professor**, Department of Computer Science and Engineering, Geethanjali College of Engineering and Technology, is submitted in partial fulfillment of the requirements for the award of the degree of **Bachelor of Technology in Computer Science and Engineering**.

This is a record of bonafide work carried out by us in **Geethanjali College of Engineering and Technology** and the results embodied in this project have not been reproduced or copied from any source.The results embodied in this project report have not been submitted to any other University or Institute for the award of any other degree or diploma.

Vudhanthi Neeraja (19R11A0548)

Vankayalapati Vinitha (19R11A0544)

Kamuju Nova (19R11A0517)

ACKNOWLEDGEMENT

We are greatly indebted to the authorities of Geethanjali College of Engineering and Technology, Cheeryal, Medchal District, for providing us the necessary facilities to successfullycarry out this mini project work titled **"ONLINE VOTING SYSTEM USING FACIAL RECOGNITION".**

Firstly, we thank and express our solicit gratitude to our Head of the Department (HoD) **Dr A. SREE LAKSHMI**, Professor, CSE department, Geethanjali College of Engineering and Technology, for her invaluable help and support which us a lot in successfully completing our mini project.

Secondly, we express our gratitude to **Dr. K. Ambedkar,** Associate Professor, internal guide, Geethanjali College of Engineering and Technology, for his suggestion and encouragementwhich helped us in the successful completion of our mini project.

We would like to express our sincere gratitude to our Principal **Dr. S. UDAYA KUMAR** for providing the necessary infrastructure to complete our project.

We convey our gratitude to our Chairman, **Mr. G. RAVINDER REDDY**, for his invaluable support and encouragement for propelling our innovation forward.

Finally, we would like to express our heartfelt thanks to our parents who were very supportiveboth financially and mentally and for their encouragement to achieve our set goals.

ABSTRACT

Voter ID is an important document to have, as this allows us to voice our opinion and elect the candidates of our government. As important it is for people to possess a Voter ID, the proper verification for issuing the same and authenticating the user by providing them with an electoral card. These verification processes currently are done manually where in the possible voter fills in a form appropriately either online or offline and attaches the necessary documents along with it. These documents include address proof and identity proof etc. The attached documents are then verified and if any discrepancies are found the application is rejected or else if the details are proven to be satisfactory the Voter ID is issued. After the Voter ID is verified a few days before election day, the electoral are issued electoral cards/ slips which permits the voter to enter the booth allotted to them. These issued slips are verified by an election booth officer based on the details provided by the user as part of the application process. Along with these details, the user's identity is verified based on the photo on the Voter ID that is issued to them.

The major problem with the current voting verification system is the details that are provided can be counterfeited and can be gone unnoticed if proper verification steps are not taken. Another major problem of the present day is the votes being rigged. This problem arises due to the unethical means of confirmation regarding the voter's physical aspects.

To eliminate rigging, and automate the verification of the person's identity accurately and efficiently, we make use of facial recognition to verify and legitimize the voter. The facial recognition is done on the input image that is given by the voter as part of the verification process to check the lineaments of the voter. Even though there might be changes in the appearance, the facial alignments of the voter will always be constant. As the facial recognition pinpoints and measures facial features from input the voter will not be able to rig the vote even if their appearance is altered. The project matches the input image with the face images available in the data, and checks for the vote attribute that indicates whether a person has voted or not. If the voter has not voted, the system changes the status of the voter to "voted" and permit the voter to vote, else if the voter seems to have voted previously and is trying to rig their vote, the voter is not allowed to vote.

LIST OF FIGURES

TABLE OF CONTENTS

1. INTRODUCTION

1.1 ABOUT THE PROJECT

A voting system is a set of rules that determine how elections are conducted and how their results are determined. Electoral systems are used in politics to elect governments. These rules govern all aspects of the voting process: when elections occur, who is allowed to vote, who can stand as a candidate, how ballots are marked and cast, how the ballots are counted, how votes translate into the election outcome, limits on campaign spending, and other factors that can affect the result. Political voting systems are defined by constitutions and electoral laws, are typically conducted by election commissions, and can use multiple types of elections for different offices.

Election plays an important role in such a huge democratic country like India where the leader is elected by residents. Elections preserve a truthful state functioning, as they provide people the choice to select their personal government. So the election ought to be an unfastened and truthful process. Every citizen of a democratic country has a right of voting with his/her own choice. One of the fundamental issues in the conventional democratic framework is that it expends bunches of labor and resources. Also some humans can be worried about illegal publications of movement at some point of this manner of election or its preparation. There are some disadvantages of the conventional election voting process which is being used in our country such as machine stops working, chances of brutality, time consuming, resource consuming, spot arranged etc. Many people couldn't vote because the voter has to reach the poll booths to vote or some people like those who are living far away from their original birth place where they are allowed to vote. So to get rid of their drawbacks, a new System is introduced i.e. Online Voting System, which provides accuracy, security, flexibility, mobility etc. An online voting System in a web-based application to use in the election process.

With the advancements of machine learning and deep learning methods and techniques in recent years, especially the ones utilizing convolutional neural networks

(CNNs), Eigen faces, and Support Vector Machine (SVM), various research works proved that the application of such methods against the facial recognition and authentication is resulting in encouraging results. Considering these advancements and based on previous experience with the traditional voting system, we set out goals to adapt the facial recognition and Machine Learning algorithms to authenticate the voter and build a verification model and evaluated this predictive model by various analysis and interpretable representations of our model in order to gain useful insights.

The proposed system with facial recognition authentication to makes the voting process more secure and reduce the time taken in the voting process. By the use of this, the electorate can solidify their vote for his or her preferred candidate through the use of their system.

1.2 PROJECT SCOPE

The aim is to develop an application that seeks to use various stages of security authentication to enhance the election process for political party elections using the real case studies, in the end imparting an internet platform which permits all eligible electorate to workout their franchise from any region for the duration of the election period.

The scope of this project is:

- To create a secured online voting platform where authenticity of votes and voters are ensured with the use of mechanisms such as facial recognition and one – time password
- To enhance Voter’s identity due to the fact that facial lineaments can not be shared.
- To ease the trouble of queuing in the course of balloting duration in elections.

1.3 PROBLEM STATEMENT

Electronic voting systems are rapidly overlapping the traditional paper-based voting. In traditional voting there the basic methodology as applied to online voting systems would involve giving voters realistic voting tasks to accomplish using a variety of ballot design and allow the voter to cast a vote in a secure manner while maintaining the time, verification and also the security of the entire system.

2. SYSTEM ANALYSIS

2.1 EXISTING SYSTEM

At present there are two types of voting methods, they are:

- Ballot Voting
- EVM Voting

A. Ballot Voting:

A ballot is a device used to cast votes in an election, and may be a piece of paper used in secret voting. In this the voter is given a paper which consists of all the party symbols along with representative names in it. Here, people come to the polling booth, take the ballot paper and vote by putting a stamp on the desired party symbol. Finally, the ballot paper is folded and dropped into the ballot box. At last, the votes are counted by the Election commission officers.

B. EVM (Electronic Voting Machine) Voting:

An EVM is a device which is used for voting. This machine consists of party symbols along with the representative's name and a button at the end for each and every party name. The voters come near the EVM machine after completion of their verification at the early level before voting. After verification the voter goes near the EVM and casts their vote by pressing the button. The above procedures are not so accurate as there may be possibility for the false/fake voting. The ballot papers may be lost at the time of counting which may affect results of the particular area or people may miscount the number of votes which leads authority into wrong hands. EVM machines sometimes get corrupted and polling gets stopped temporarily and a lot of time is wasted or EVM may be tampered and the casted votes may be polled to a particular party only, even the vote is casted to different candidates or parties. This may lead authority into the wrong hands. They also lack security as one's vote can be casted by another voter or even a miscellaneous person. This factor is known as fake voting. Without proper authentication there is a possibility of fake voting. So,

the existing system is not efficient for voting. Even though there is very little false/fake voting, this minor setback can turn the results in the opposite direction.

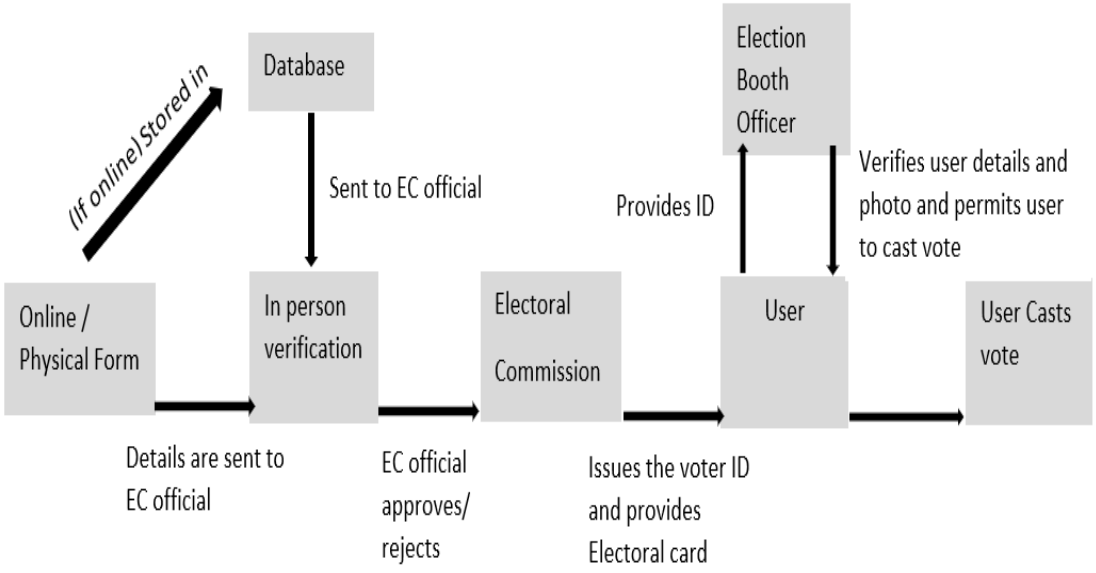

Fig 1. Existing System Architecture

2.1.1 Disadvantages of existing system

- The details and evidences can be counterfeited and illegitimate identities can be created.
- Fake identities are used to rig votes.
- Less Accuracy in verifying the user.
- Expensive and time consuming
- Too much paper work
- Errors during data entry
- Loss of registration forms

2.2 PROPOSED SYSTEM

To eliminate rigging, we put forth an automated approach that can verify the person's identity accurately and efficiently. This project revolves around the development of an online voting system that employs facial recognition registration each voter per election, which will allow voters to participate in the elections regardless of their physical location. In our project, we make use of facial recognition to verify and legitimize the voter. The facial recognition is done without the booth officer manually having to check the lineaments of the voter. Even though there might be changes in the appearance, the facial alignments of the voter will always be constant. As the facial recognition pinpoints and measures facial features from a image/ video input the voter will not be given a chance to rig even if their appearance is altered. The project inputs the voter's face to match it with the face image available in the data, and checks for the vote attribute that indicates whether a person has voted or not. If the voter has not voted, the status of the vote attribute in the database changes to "voted" and permit the voter to vote, else if the voter seems to have voted previously and is trying to rig their vote, he/she will be directed to a webpage indicating that they have already voted. If the voter is not registered, then the voter is directed to a webpage where he/she can register themselves to vote. Thus, the proposed system helps detect when votes are being rigged and help in avoiding the rigging of votes.

2.2.1 Advantages of Proposed System

- Better accuracy
- Time saving
- Cost efficient

2.2.2 Details

An online voting system is a software platform that allows voters to securely cast their votes. High-quality online voting systems balance ballot security, accessibility, and the overall requirements of an electorates voting event. At their core, online voting systems

protect the integrity of your vote by preventing voters from being able to vote multiple times. As a digital platform, they eliminate the need to gather in-person, cast votes using paper, or by any other means. Using an online voting tool will generate confidence in the results of your votes and elections, lower your voting-related costs, and streamline the election process for both you and your voters. In our project, we make use of facial recognition to verify and legitimize the voter. The facial recognition is done without the booth officer manually having to check the lineaments of the voter. Even though there might be changes in the appearance, the facial alignments of the voter will always be constant. As the facial recognition pinpoints and measures facial features from a image/ video input the voter will not be given a chance to rig even if their appearance is altered.

2.2.3 IMPACT ON ENVIRONMENT

- There is no impact on the environment.
- Doesn’t cause damage to environment.

2.2.4 ETHICS

This project follows the general software and hardware ethics. This system does not harm any individual in any way.

2.2.5 TYPE

This project is a machine learning based project. The aim of the project is to design/develop an easy online voting system which is feasible to use by the public.

2.3 MODULES DESCRIPTION

A module is a collection build settings that allow us to divide your project into discrete units of functionality. The project can have one or many modules, and one module may use another module as a dependency. We can independently build, test, and debug each module. The Project Implementation module focuses on the Implementation

of project with a focus on placemaking. The concepts of scale, timeline, resources and power are seen as key elements of implementation.

The modules for building the project are as follows:

1. Data Collection
2. Data Pre-processing
3. Feature Extraction
4. Build a facial recognition model
5. Accuracy

2.3.1 Data Collection

Data collection is defined as the procedure of collecting, measuring and analyzing accurate insights for research using standard validated techniques. A researcher can evaluate their hypothesis on the basis of collected data. In most cases, data collection is the primary and most important step for research, irrespective of the field of research. The approach of data collection is different for different fields of study, depending on the required information. For our project we collect data on different faces for training our facial recognition model .

2.3.2 Data Pre-processing

Data preprocessing in Machine Learning is a crucial step that helps enhance the quality of data to promote the extraction of meaningful insights from the data. Data preprocessing in Machine Learning refers to the technique of preparing (cleaning and organizing) the raw data to make it suitable for a building and training Machine Learning and deep learning models. In simple words, data preprocessing in Machine Learning and deep learning is a data mining technique that transforms raw data into an understandable and readable format.

2.3.3 Feature Extraction

Feature extraction refers to the process of transforming raw data into numerical features that can be processed while preserving the information in the original data set. It yields better results than applying machine learning directly to the raw data.

Feature extraction can be accomplished manually or automatically:

1. Manual feature extraction requires identifying and describing the features that are relevant for a given problem and implementing a way to extract those features. In many situations, having a good understanding of the background or domain can help make informed decisions as to which features could be useful. Over decades of research, engineers and scientists have developed feature extraction methods for images, signals, and text. An example of a simple feature is the mean of a window in a signal.

2. Automated feature extraction uses specialized algorithms to extract features automatically from signals or images without the need for human intervention. This technique can be very useful when you want to move quickly from raw data to developing machine learning algorithms. Wavelet scattering is an example of automated feature extraction. With the ascent of deep learning, feature extraction has been largely replaced by the first layers of deep networks – but mostly for image data. For signal and time-series applications, feature extraction remains the first challenge that requires significant expertise before one can build effective predictive models.

In our project we have made use of the automated feature extraction for to classify images based on an object-based approach by using a group of pixels with similar spectral, spatial, and/or texture attributes.

2.3.4 Build a Facial Recognition model

1. Build a database of pictures

You need to build a database of pictures using facial-recognition software. If you already have pictures, you can use them as well, but make sure they are taken from

different angles and distances. One way that artificial intelligence can be used to build a database of pictures for this type of software is by having an artificially intelligent computer program analyses all the pictures on social media. Then create a database with them categorized by which person's face was in the picture.

2. Train the software by inserting new pictures into the database

Building facial recognition software is not as easy as it sounds. That is why we need to train the software by inserting new pictures into the database. The system will be able to learn and identify images of faces and can then compare them with other images that we haven't trained it with. To build a good system, we need to start by taking a picture of a person's face. Then extracting the most important facial features like the eyes, nose, mouth, and eyebrows. We can extract various data points such as distance between specific points on the face or angles of the face etc., which are important for identifying different faces. Building this AI-powered system may seem complicated at first glance. But if you break down each step into smaller tasks it becomes much easier.
This will help to avoid errors during identification. The algorithm will learn from your pictures to identify others with similar features in future use cases.

3. Test your software to check its accuracy

Train the algorithm with the survey data to make it more accurate in recognizing faces.
The first thing to do is to create a training set of faces that are labelled with names. Then, train your algorithm on the data. The goal is to create an algorithm that can correctly identify people in an image or video frame by matching the face with the name associated with it.

4. Make changes to improve its accuracy

The main issue that needs to be addressed before the facial-recognition software can be used as an identification tool is its accuracy. There are several issues with the accuracy

of the facial-recognition software such as its inability to recognize faces of people who are not white, different skin tones and there is also an issue of false positives. So, there need to be some changes made in order to improve its accuracy.

In order to build facial recognition software that is accurate enough for use as an identification tool, it should have layers of data to compare with. In comparison with one-layer recognition where it only has one reference photo available for comparison. Two-layer recognition has two reference images available for comparison.

For facial-recognition software, accurate identification of a face is crucial. This accuracy ranges from 99.9% to 100%. The accuracy of the software depends upon the resolution, angle, and lighting conditions.

Improve resolution: It will be helpful if we can improve the resolution to make our product more accurate.

Improve angle: We should make sure that our product does not get distorted because of wrong angles and lighting conditions.

5. Use the system on images through other inputs:

In order to make sure that the accuracy of this software is high, we need to make sure that it can work in any lighting condition and from any angle. The best way to do it is by training the AI with some facial photographs from those angles and lighting conditions and then testing it on live events once again after this training. This is to evaluate how well your algorithm performs.

2.3.5 Accuracy

Accuracy is one metric for evaluating classification models. Informally, accuracy is the fraction of predictions our model got right. Formally, accuracy has the following definition:

Accuracy=Number of correctly recognized faces/Total number of faces recognized

For binary classification, accuracy can also be calculated in terms of positives and negatives as follows:

$$\text{Accuracy} = (TP + TN) / (TP + TN + FP + FN)$$

Where :

TP = True Positives (Actual image in dataset and recognized correct),
TN = True Negatives (Actual image not in dataset and recognized correct),
FP = False Positives (Actual image not in dataset and recognized correct) and
FN = False Negatives (Actual image in dataset and recognized not correct).
This is used while training and testing the facial recognition of the verification system.

2.4 SYSTEM CONFIGURATION

2.4.1 Software Requirements

- Operating system : Windows 7/8/8.1/10/11
- Coding Language : Python
- Coding platform : Pycharm
- Databases (for User information) : MySQL
- OpenCV

2.4.2 Hardware Requirements

- Processor : i3,i5,i7
- Hard Disk : 500GB.
- Ram : 2 GB, 8 GB.
- Keyboard : Standard keyboard
- CPU: Intel Core 2 Quad Processor

3. LITERATURE OVERVIEW

Literature survey is the most important step in software development process. Before developing the tool it is necessary to determine the time factor, economy and company strength. Once these things are satisfied, ten next steps are to determine which operating system and language can be used for developing the tool. Once the programmers start building the tool the programmers need lot of external support. This support can be obtained from senior programmers, from book or from websites. Before building the system the above consideration are taken into account for developing the proposed system.

3.1 A deep facial recognition system using computational intelligent algorithms

Authors: Diaa Salama AbdELminaam , Abdulrhman M. Almansori, Mohamed Taha, Elsayed Badr

The development of biometric applications, such as facial recognition (FR), has recently become important in smart cities. Many scientists and engineers around the world have focused on establishing increasingly robust and accurate algorithms and methods for these types of systems and their applications in everyday life. FR is developing technology with multiple real-time applications. The goal of this paper is to develop a complete FR system using transfer learning in fog computing and cloud computing. The developed system uses deep convolutional neural networks (DCNN) because of the dominant representation; there are some conditions including occlusions, expressions, illuminations, and pose, which can affect the deep FR performance. DCNN is used to extract relevant facial features. These features allow us to compare faces between them in an efficient way. The system can be trained to recognize a set of people and to learn via an online method, by integrating the new people it processes and improving its predictions on the ones it already has. The proposed recognition method was tested with

different three standard machine learning algorithms (Decision Tree (DT), K Nearest Neighbor(KNN), Support Vector Machine (SVM)). The proposed system has been evaluated using three datasets of face images (SDUMLA-HMT, 113, and CASIA) via performance metrics of accuracy, precision, sensitivity, specificity, and time. The experimental results show that the proposed method achieves superiority over other algorithms according to all parameters. The suggested algorithm results in higher accuracy (99.06%), higher precision (99.12%), higher recall (99.07%), and higher specificity (99.10%) than the comparison algorithms.

3.2 Face Recognition Using Eigenface Approach

Authors: Marijeta Slavković, Dubravka R. Jevtic

In this article, a face recognition system using the Principal Component Analysis (PCA) algorithm was implemented. The algorithm is based on an eigenfaces approach which represents a PCA method in which a small set of significant features are used to describe the variation between face images. Experimental results for different numbers of eigenfaces are shown to verify the viability of the proposed method.

This problem is solved by the method called Principal Component Analysis. PCA is a projection technique that finds a set of projection vectors designed such that the projected data retains the most information about the original data. The most representative vectors are eigenvectors corresponding to highest eigenvalues of the covariance matrix. This method reduces the dimensionality of data space by projecting data from M-dimensional space to P-dimensional space, where P<<M.

3.3 The implementation of eigenface algorithm for face recognition in attendance system

Authors: Vincentius Kurniawan; Arya Wicaksana; Maria Irmina Prasetiyowati

Technology advancement has brought in mobility and flexibility into the workplaces in contrast to the old days. Workers are demanded to perform their job at places other than

their office. The well-known long-established attendance systems that are widely used in workplaces are heavily depending on technologies such as the Radio Frequency Identification (RFID) and fingerprint. Both technologies have limitation especially when it comes to flexibility and mobility. Thus, this research proposes an attendance system that addresses the mentioned condition. The attendance system is built using Android and web technologies with geolocation extraction feature and biometric technology: the face recognition. The Eigenface algorithm is chosen for face recognition process in the system. In addition to that, Euclidean distance is used for calculate the distance between input image and the training image. There are variables in this research that may disturb the recognition process: lighting, distance between the face and the camera, and hardware specifications, which are not taken into consideration. Based on the implementation and testing process, the overall accuracy of the system is 86.67%.

3.4 Support Vector Machine Classification Algorithm and Its Application

Authors: Yongli Zhang

The support vector machine is a new type of machine learning methods based on statistical learning theory. Because of good promotion and a higher accuracy, support vector machine has become the research focus of the machine learning community. This paper introduces the basic theory of support vector machine, the basic idea of the classification and currently used support vector machine classification algorithm. Practical problems with which an algorithm, and proves the effectiveness of the algorithm, the final outlook of the prospects of support vector machines in classification applications. Finally the prospect of the prospect of support vector machines in classification applications.

3.5 Face Recognition as an Authentication Technique in Electronic Voting

Authors: Noha E. El-Sayad, Rabab Farouk Abdel-Kader, Mahmoud Ibraheem Marie

In this research a Face Detection and Recognition system (FDR) used as an Authentication technique in online voting, which one of electronic is voting types, is proposed. Web based voting allows the voter to vote from any place in state or out of state. The voter's image is captured and passed to a face detection algorithm (Eigenface or Gabor filter) which is used to detect his face from the image and save it as the first matching point. The voter's National identification card number is used to retrieve and return his saved photo from the database of the Supreme Council elections (SCE) which is passed to the same detection algorithm (Eigenface or Gabor filter) to detect face from it and save it as second matching point. The two matching points are used by a matching algorithm to check wither they are identical or not. If the results of the matching algorithm are two point match then checks wither this person has the right to vote or not. If he has right to vote then a voting form is presented to him. The result shows that the proposed algorithm capable of finding over 90% of the faces in database and allows their voter to vote in approximately 58 seconds.

3.6 Smart Voting Machine Based on Finger Prints and Face Recognition

Authors: Nadar Rajkani Paulraj, G.Rajagopalan, M.Rajesh, S.V.Kiruthika, I.Jasmine

Bogus (fake) voting is still major drawbacks in the Election. In AADHAR CARD the Government has all the data Base for us including finger print and Retina. Biometric Finger print devices and Web Camera are used in the Electronics Voting Machine for voter verification. We have designed a Smart Voting Machine where there is no need for the user to carry his ID which contains his required details. The person at the polling booth needs only to place his Finger in finger print scanner and capture the face identity

in web camera at the counter of the polling booth, thus allowing the acquisition of an on-spot Fingerprint and Face from the voter which serves as an identification. This data is passed on to the controlling unit for the verification. The controller fetches the data from the reader and compares this data with the already existing data stored during the registration of the voters. If the data matches with the pre-stored information of the registered fingerprint and Face, the person is allowed to cast his vote. If not, a warning message is displayed on LCD and warns through the voice by this way, the person is barred from polling his vote. The vote casting mechanism is carried out manually using the keyboard. LCD is used to display the related messages, warnings and ensuing results. The evaluation of the development quality measures and then their optimization are then proposed so that cognitive computing, notably by inclusion of some decision making, behavioral, social, etc., biases, in particular the so-called status quo and minimal change biases. We extend the model to include a more sophisticated analysis of variability of temporal evolution of some life quality indicators and a human perception of its goodness. We also mention how to reflect elements related to fairness. We show how the new elements of the regional development model proposed can change the best development scenarios derived.

4. SYSTEM DESIGN

4.1 SYSTEM ARCHITECTURE

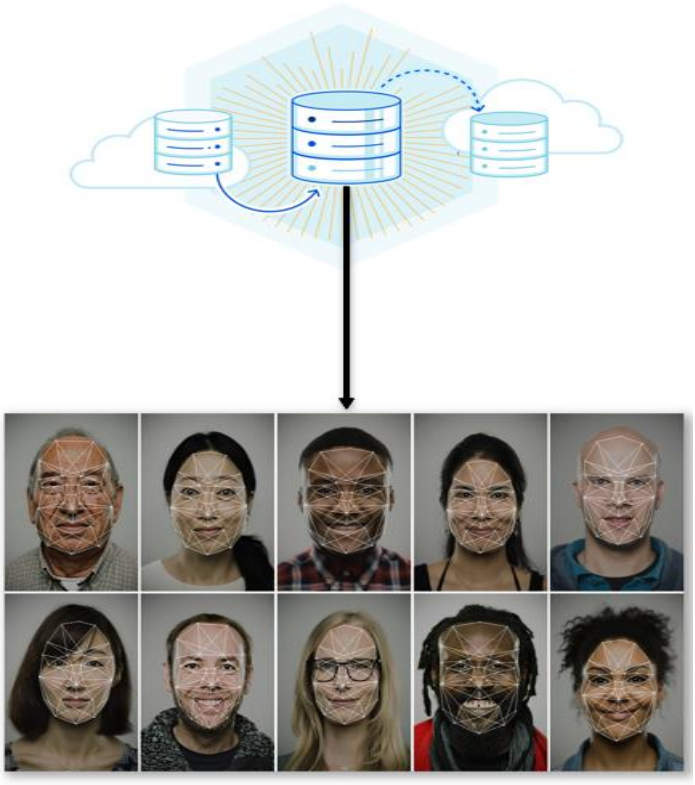

Fig 2.1 Instance of verification system using images from database to recognize the face which is given as input.

We are given with a dataset through which feature extraction is done using eigen faces and SVM followed by creation of sub-dataset, training data sets using different classifiers and using the models created after training to test the given faces. The process is further illustrated in the below architecture.

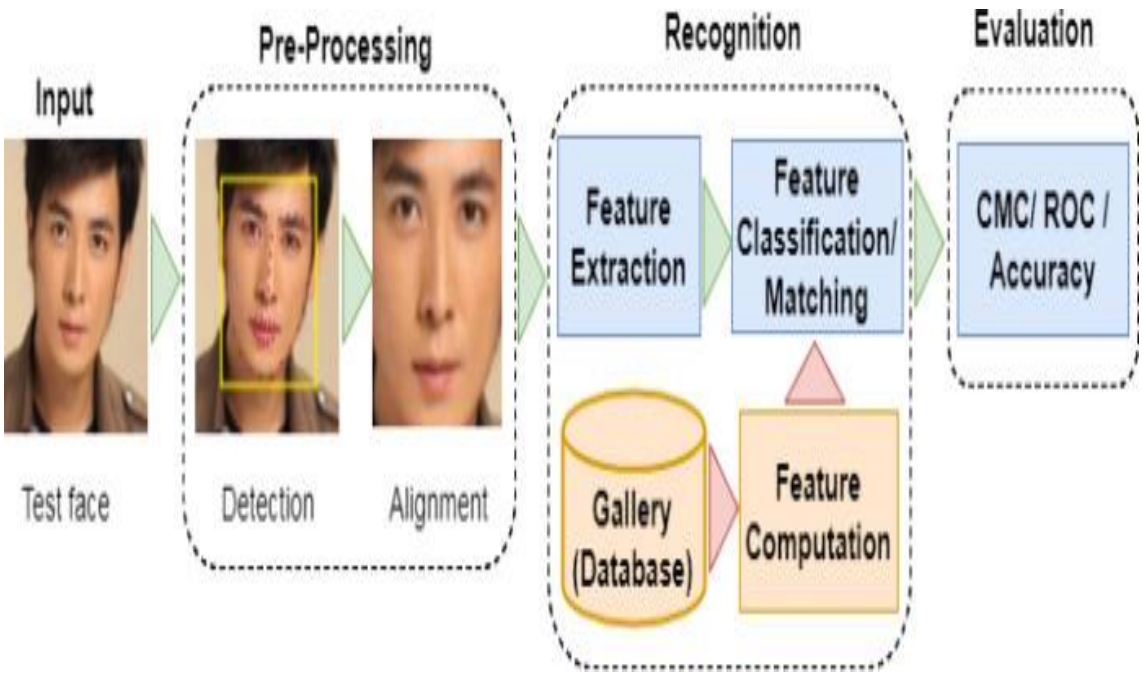

Fig 2.2 System architecture with pre processing and feature extraction

Preprocessing:

Usually, the purpose of using preprocessing steps in face detection system is to speed up the detection process and reducing false positives. The pre-processing is an improvement of the image data that suppresses unwilling distortions or enhances some image features important for further processing, although geometric transformations of images (e.g. rotation, scaling, translation) are classified among pre-processing methods. A preprocessing step should reject an acceptable amount of non-face windows. First proposed criterion is based on linear image transform (LIT) which ignores scanning a number of non-face windows. Second criterion utilizes regional minima (RM) to reject non-face windows. The last one uses a modified adaptive thresholding (ADT) technique to convert input image into a binary representation and perform an exclusion process on the latter form.

Facial Detection in pre-processing

Person detection is a variant of object detection used to detect a primary class “person” in images or video frames. Detecting people in video streams is an important task in modern facial recognition systems. The recent deep learning algorithms provide robust person detection results. Most modern person detector techniques are trained on frontal and asymmetric views.

Facial Alignment in pre-processing

Face alignment is a computer vision technology for identifying the geometric structure of human faces in digital images. Given the location and size of a face, it automatically determines the shape of the face components such as eyes and nose. A face alignment program typically operates by iteratively adjusting a deformable models, which encodes the prior knowledge of face shape or appearance, to take into account the low-level image evidences and find the face that is present in the image.

Feature Extraction

Facial feature extraction is the process of extracting face component features like eyes, nose, mouth, etc from human face image. Facial feature extraction is very much important for the initialization of processing techniques like face tracking, facial expression recognition or face recognition. Among all facial features, eye localization and detection is essential, from which locations of all other facial features are identified. However, the existing face recognition techniques failed to identify the exact person.

Feature Classification

Matching of sketch and photo is very important in police verification and intelligence as it is used to track the criminals or some person. Therefore, it should be accurate so that mismatching of the sketch with photo can be avoided. For this, feature based

matching method is introduced through this paper. Feature based matching includes a feature vector which contains features of the face image (either it is a sketch or a photo) in terms of histogram of oriented gradients (HoG) features and gray level co-occurrence matrix (GLCM) features. Computing the features, first, increases the chances of correct matches. It can be depicted from the results as well that, comparing with other state-of-art approaches, proposed method is more accurate in matching the sketches with photos of the same person.

Max Pooling

The input given passes through the facial recognition model that which defines the images with a particular input size and filters the images by converting the images from RGB to BRG. The max pooling layers that are in built in the eigen faces implementation of facial recognition collect the filter data after which the flatten layer converts the images from RGB to BRG. The following dense layers that come after the flatten layer defines the output layer and will predict 1 disease from the given 21 disease classes.

The presented architecture on a whole consists of pre-processing using detection and alignment, recognition using feature extraction and feature classification and a layer that checks for accuracy.

4.2 UML CONCEPTS

The Unified Modelling Language (UML) is a standard language for writing software blue prints. The UML is a language for

- Visualizing
- Specifying
- Constructing
- Documenting the artifacts of a software intensive system.

The UML is a language which provides vocabulary and the rules for combining words in that vocabulary for the purpose of communication. A modelling language is a language whose vocabulary and the rules focus on the conceptual and physical representation of a system. Modelling yields an understanding of a system.

4.2.1 Building Blocks of the UML:

The vocabulary of the UML encompasses three kinds of building blocks:

- Things
- Relationships
- Diagram

Things are the abstractions that are first class citizens in a model; Relationships tie these things together; Diagrams group interesting collections of things.

4.2.2 Things in the UML

- Structural things
- Behavioral things
- Grouping things
- Annotational things

4.2.3 Relationships in the UML

- Dependency:

In UML, a dependency relationship is a relationship in which one element, the client, uses or depends on another element, the supplier. You can use dependency relationships in class diagrams, component diagrams, deployment diagrams, and use-case diagrams to indicate that a change to the supplier might require a change to the client. You can also use a dependency relationship to represent precedence, where one model element must precede another. Typically, dependency relationships do not have names.

- Associations:

In UML models, an association is a relationship between two classifiers, such as classes or use cases, that describes the reasons for the relationship and the rules that govern the relationship. An association represents a structural relationship that connects two classifiers. Like attributes, associations record the properties of classifiers. For example, in relationships between classes, you can use associations to show the design decisions that you made about classes in your application that contain data, and to show which of those classes need to share data. You can use an association's navigability feature to show how an object of one class gains access to an object of another class or, in a reflexive association, to an object of the same class. The name of an association describes the nature of the relationship between two classifiers and should be a verb or phrase. In the diagram editor, an association appears as a solid line between two classifiers.

- Generalization:

In UML modeling, a generalization relationship is a relationship in which one model element (the child) is based on another model element (the parent). Generalization relationships are used in class, component, deployment, and use-case diagrams to indicate that the child receives all of the attributes, operations, and relationships that are defined in the parent.

To comply with UML semantics, the model elements in a generalization relationship must be the same type. For example, a generalization relationship can be used between actors or between use cases; however, it cannot be used between an actor and a use case. You can add generalization relationships to capture attributes, operations, and relationships in a parent model element and then reuse them in one or more child model elements. Because the child model elements in generalizations inherit the attributes, operations, and relationships of the parent, you must only define for the child the attributes, operations, or relationships that are distinct from the parent. The parent model element can have one or more children, and any child model element can have one or more parents. It is more common to have a single parent model element and multiple child model elements. Generalization relationships do not have names.

- Realization:

In UML modeling, a realization relationship is a relationship between two model elements, in which one model element (the client) realizes the behavior that the other model element (the supplier) specifies. Several clients can realize the behavior of a single supplier. You can use realization relationships in class diagrams and component diagrams. Typically, realization relationships do not have names. If you name a realization, the name is displayed beside to the realization connector in the diagram. As the following figure illustrates, a realization is displayed in the diagram editor as a dashed line with an unfilled arrowhead that points from the client (realizes the behavior) to the supplier (specifies the behavior).

4.3 UML DIAGRAMS

4.3.1 Use Case Diagram

A use case diagram is a graph of actors set of use cases enclosed by a system boundary, communication associations between actors and users and generalization among use cases. The use case model defines the outside(actors) and inside (use case) of the system's behavior. Use case diagram is quite simple in nature and depicts two types of elements; one representing the business roles and the other representing the business processes. The main purpose of a use case diagram is to show what system functions are performed for which actor. Roles of the actors in the system can be depicted.

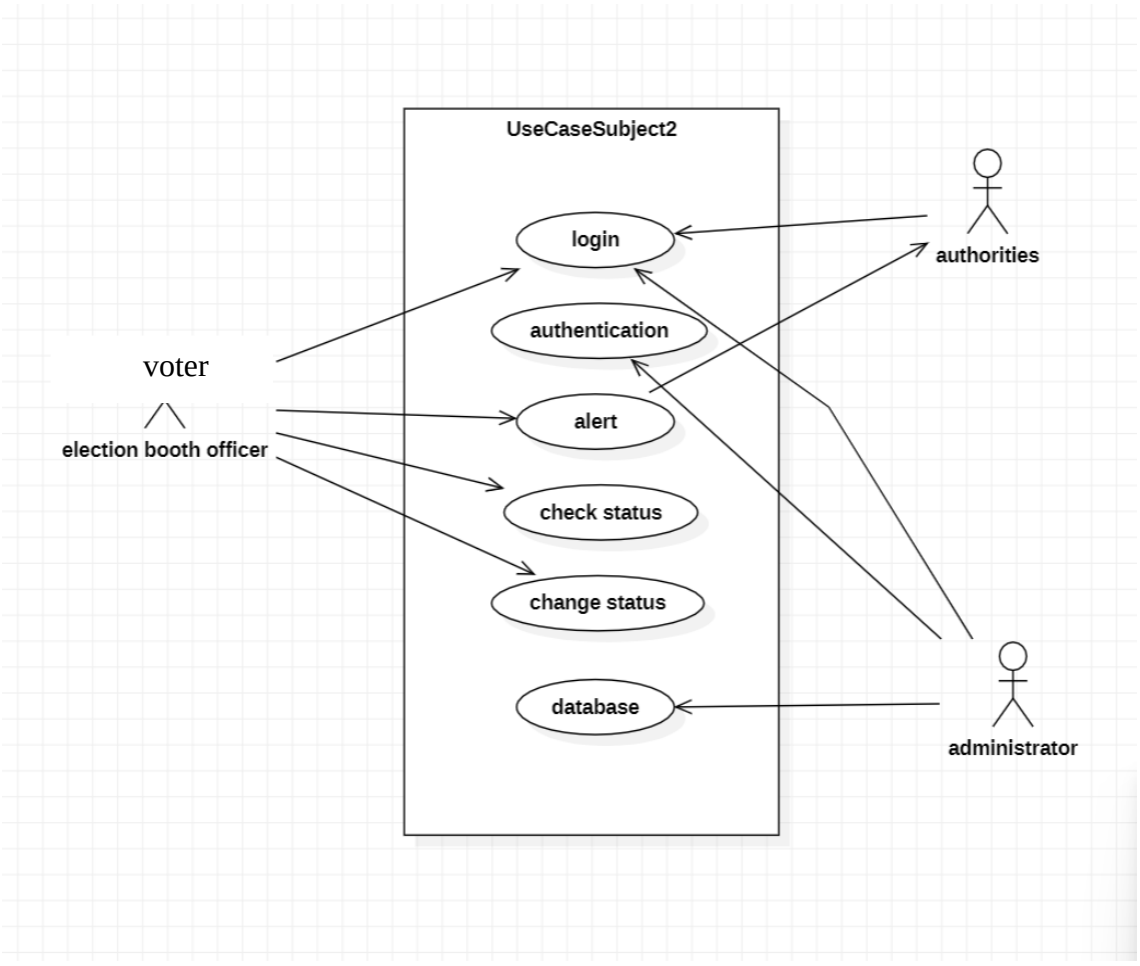

Fig 3.1 Use Case Diagram

4.3.2 Class Diagram

Class diagram is a static diagram. It represents the static view of an application. Class diagram is not only used for visualizing, describing, and documenting different aspects of a system but also for constructing executable code of the software application.

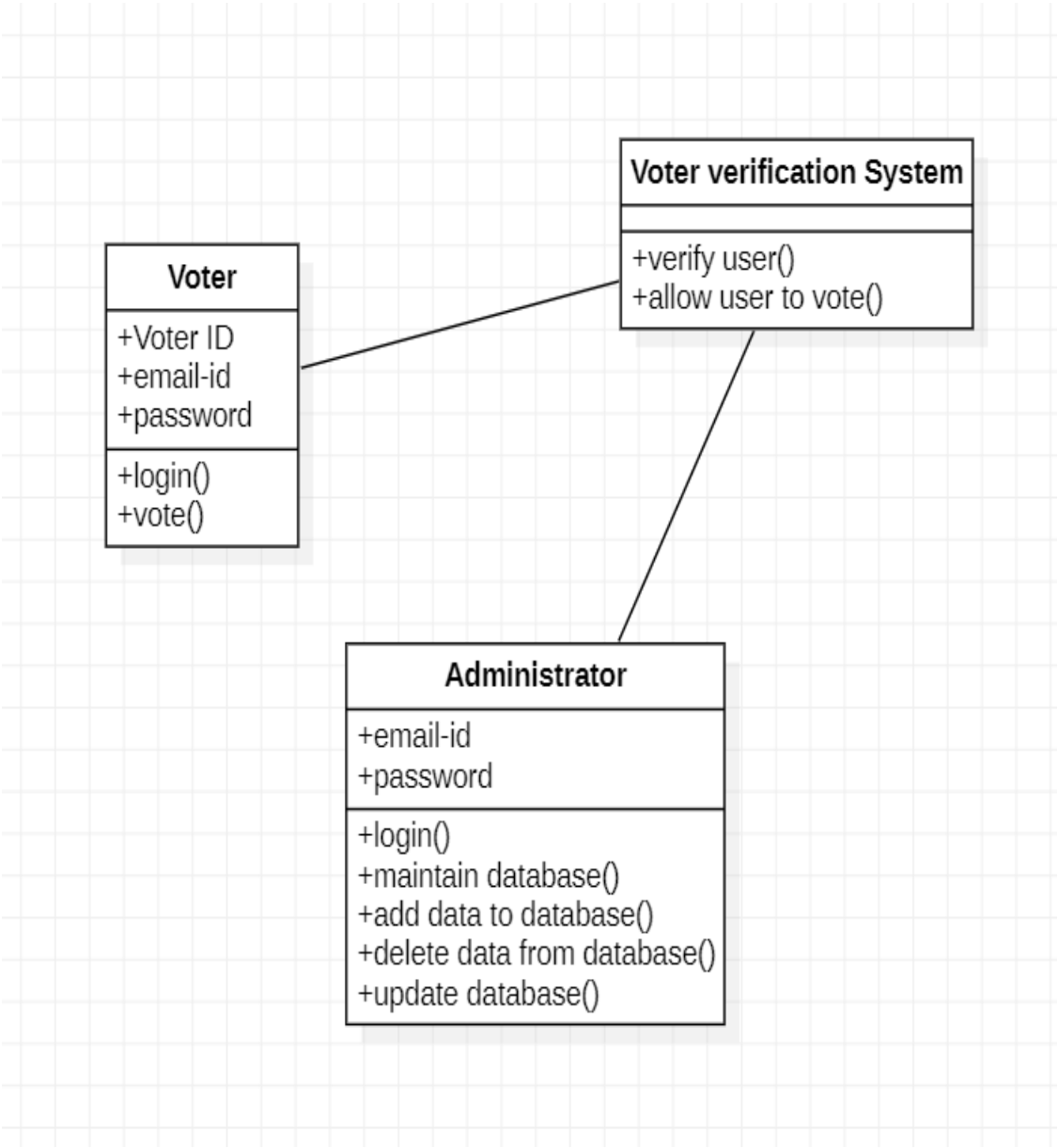

Fig 3.2 Class Diagram

4.3.3 Sequence Diagram

UML sequence diagram are used to represent the flow of messages, events and actions between the objects or components of a system. Time is represented in the vertical direction showing the sequence of interactions of the header elements, which are displayed horizontally at the top of the diagram.

Sequence Diagrams are used primarily to design, document and validate the architecture, interfaces, and logic of the system by describing the sequence of actions that need to be performed to complete a task or scenario. UML sequence diagrams are useful design tools because they provide a dynamic view of the system behavior which can be difficult to extract from static diagrams or specifications. It is a construct of a Message Sequence Chart.

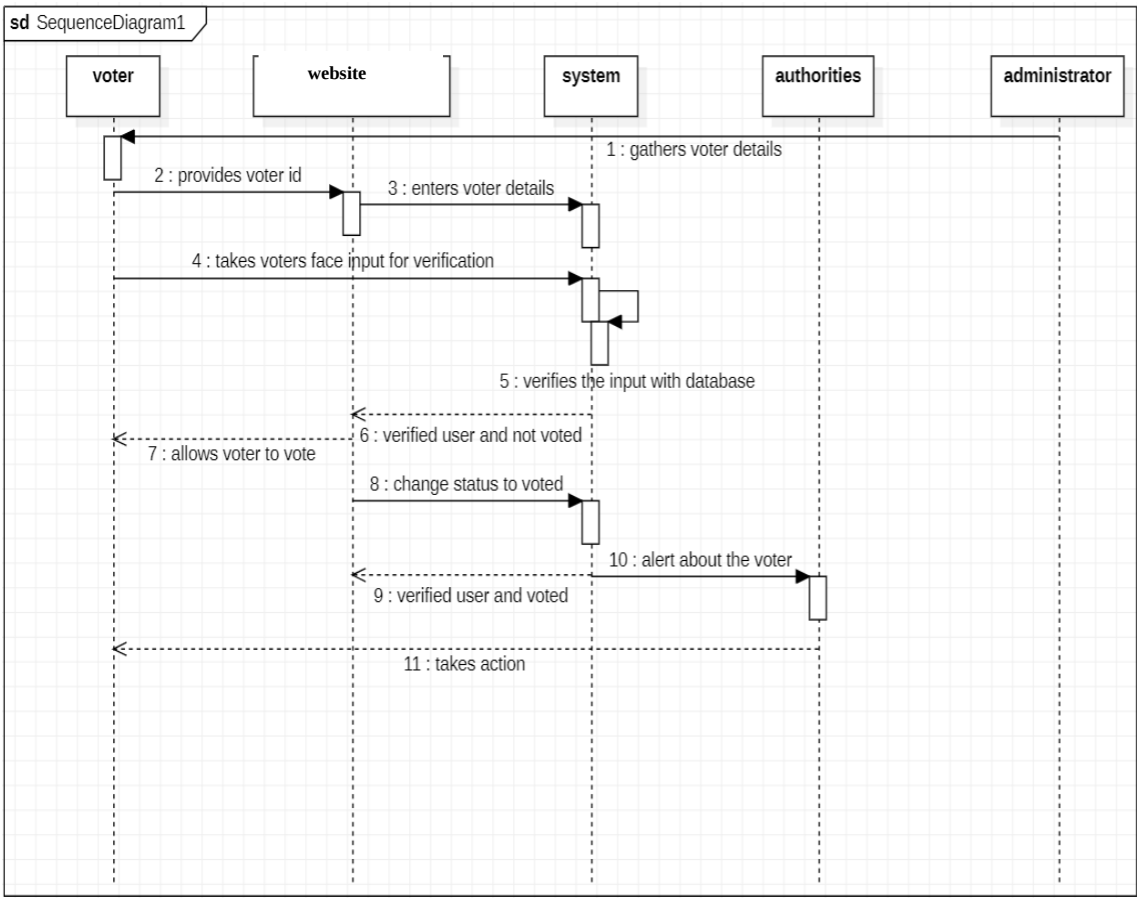

Fig 3.3 Sequence Diagram

4.3.4 Activity Diagram

We use Activity diagram to illustrate the flow of control in a system and refer to the steps involved in the execution of a use case. We model sequential and concurrent activities using activity diagram. So, we basically depict workflows visually using an activity diagram. An activity diagram focuses on condition of flow and the sequence in which it happens. We describe or depict what causes a particular event using an activity diagram.

An activity diagram portrays the control flow from a start point to a finish point showing the various decision paths that exist while the activity is being executed. We can depict both sequential processing and concurrent processing of activities using an activity diagram.

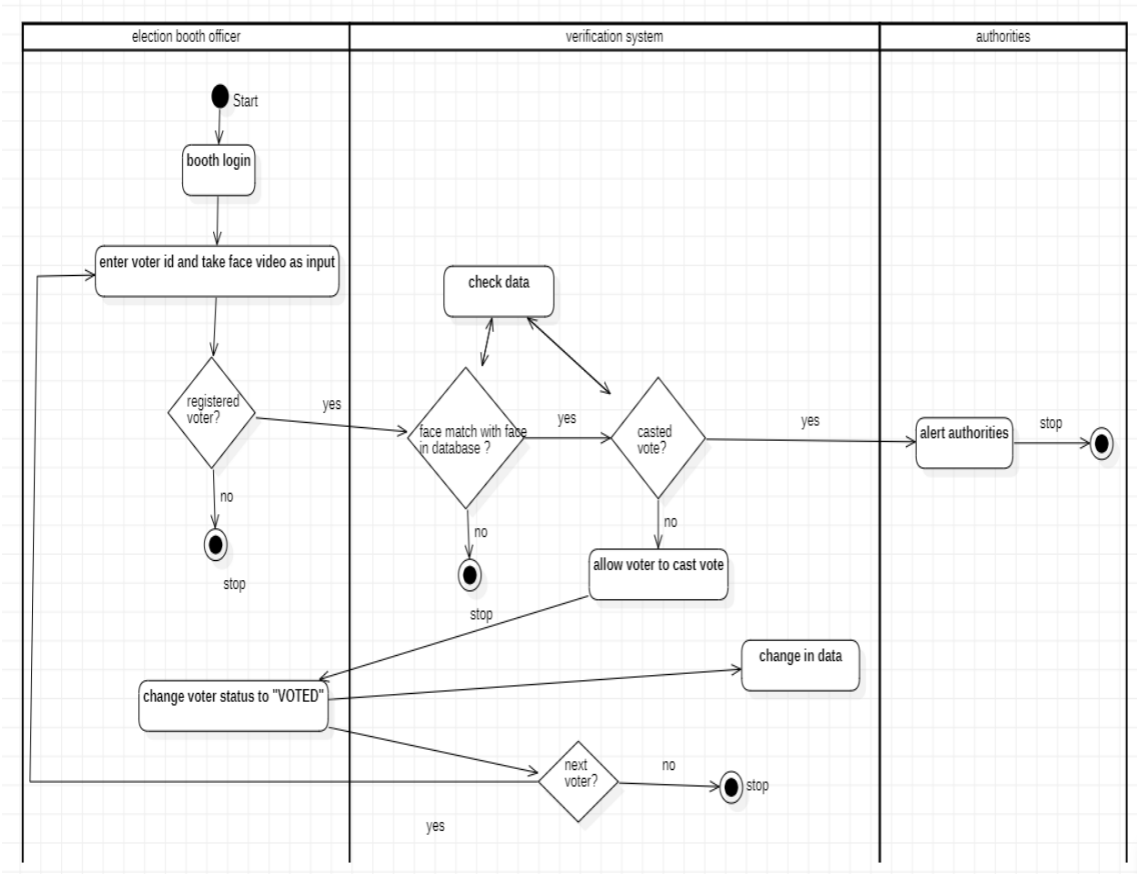

Fig 3.4 Activity Diagram

4.3.5 State Diagram

A state diagram is a type of diagram used in computer science and related fields to describe the behavior of systems. State diagrams require that the system described is composed of a finite number of states; sometimes, this is indeed the case, while at other times this is a reasonable abstraction.

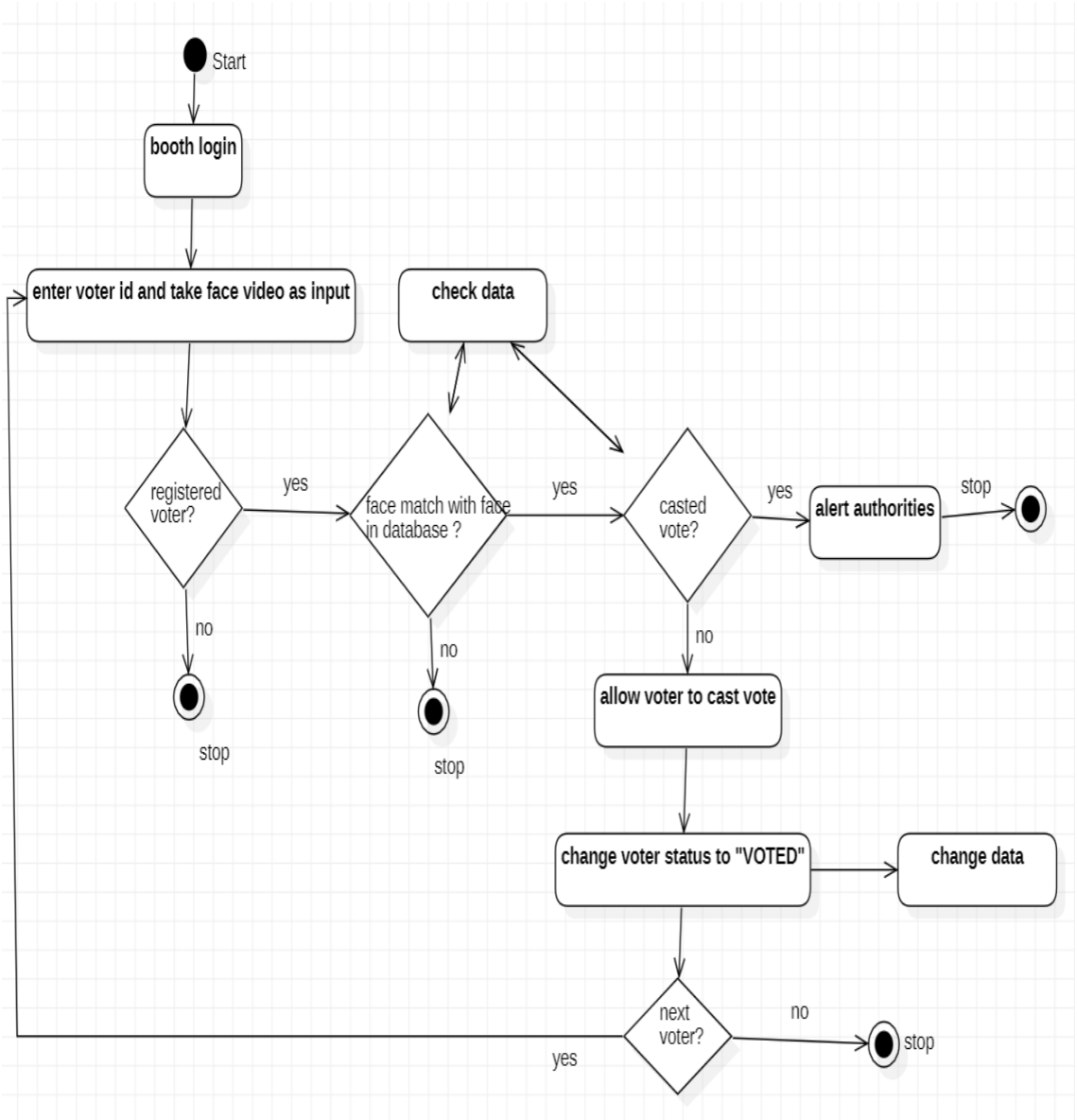

Fig 3.5 State Diagram

5 SAMPLE CODE

5.1 CODING

Facial recognition model (webapp.py)

```
import streamlit as st
import webbrowser
from PIL import Image
import cv2
import numpy as np
import face_recognition
import os
from datetime import datetime,date
import time
import mysql.connector
db = mysql.connector.connect(
    host="localhost",
    user='root',
    passwd='1236',
    database='testdatabase'
)
mycursor = db.cursor( buffered=True)
path = 'VotingImages'
```

```
images = []
classNames = []
myList = os.listdir(path)
print(myList)
for cl in myList:
    curImg = cv2.imread(f'{path}/{cl}')
    images.append(curImg)
    classNames.append(os.path.splitext(cl)[0])
print(classNames)
def findEncodings(images):
    encodeList = []
    for img in images:
        img = cv2.cvtColor(img, cv2.COLOR_BGR2RGB)
        encode = face_recognition.face_encodings(img)[0]
        encodeList.append(encode)
    return encodeList


def markVoted(name):
    query = """SELECT Status FROM VoteStatus WHERE Name=%s"""
    tuple1 = name
    mycursor.execute(query,(tuple1,))
    status = mycursor.fetchone()
```

```
    #print(status)
    if status[0] == 'notvoted':
        query = """UPDATE VoteStatus SET Status = 'voted' where Name=%s"""
        tuple1 = name
        mycursor.execute(query, (tuple1,))
        db.commit()
    return status[0]
def getVoteCount(name):
    query = """SELECT Count FROM VoteStatus WHERE Name=%s"""
    tuple1 = name
    mycursor.execute(query, (tuple1,))
    votes = mycursor.fetchone();
    #print(votes)
    if votes[0] == 0:
        query = """UPDATE VoteStatus SET Count = 1 where Name=%s"""
        tuple1 = name
        mycursor.execute(query, (tuple1,))
        db.commit()
    return votes[0]


encodeListKnown = findEncodings(images)
# print(len(encodeListKnown))
```

```
print('Encoding complete')
def detect_faces(image):
    name = 'not registered'
    list1 = []
    #image = cv2.resize(image, (0, 0), None, 0.25, 0.25)
    #image = cv2.cvtColor(image, cv2.COLOR_BGR2RGB)
    image = np.array(image.convert('RGB'))
    gray = cv2.cvtColor(image, cv2.COLOR_BGR2GRAY)
    imageFaceLoc = face_recognition.face_locations(image)
    encodesImage = face_recognition.face_encodings(image, imageFaceLoc)
    for encodeFace, faceLoc in zip(encodesImage, imageFaceLoc):
        matches = face_recognition.compare_faces(encodeListKnown,encodeFace)
        faceDis = face_recognition.face_distance(encodeListKnown,encodeFace)
        matchIndex = np.argmin(faceDis)
        if matches[matchIndex]:
            name = classNames[matchIndex]
    if name != 'not registered':
        status = markVoted(name)
        votes = getVoteCount(name)
        if status == 'notvoted' and votes == 0:
            return 0
        elif status=='voted' and votes == 1:
```

```
            return 1
    else:
        return 2
def main():
    """Face Recognition App"""
    html_temp = """
    <body style="background-color:red;">
    <div>
    <h2 style="color:white;text-align:center; background-color:teal
;padding:10px">IDENTITY VERIFICATION</h2>
    <h3 style="color:white;text-align:center;">UPLOAD IMAGE </h3>
    </div>
    </body>
    """
    st.markdown(html_temp, unsafe_allow_html=True)
    list = []
    image_file = st.file_uploader("", type=['jpg', 'png', 'jpeg'])
    if image_file is not None:
        image = Image.open(image_file)
        st.text("Uploaded Image")
        st.image(image)
    if st.button("Verify Identity"):
```

```
    num = detect_faces(image)
    if (num == 0):
        #st.button("VOTE")
        if st.button("VOTE"):
            url = 'num0.html'
            webbrowser.open_new_tab(url)
    elif(num==1):
        #st.button("ALREADY VOTED")
        if st.button("ALREADY VOTED"):
            url = 'num1.html'
            webbrowser.open_new_tab(url)
    elif(num==2):
        #st.button("NOT REGISTERED")
        if st.button("NOT REGISTERED"):
            url = 'num2.html'
            webbrowser.open_new_tab(url)

# if st.button("Verify Identity"):
#     num = detect_faces(image)
#     if(num==0):
#         url = 'num0.html'
#         webbrowser.open_new_tab(url)
```

```
    #     elif(num==1):
    #         url = 'num1.html'
    #         webbrowser.open_new_tab(url)
    #     elif(num==2):
    #         url = 'num2.html'
    #         webbrowser.open(url,new=0)
    # url = 'D:\Python Projects\VotingSystemProject\GFG.html'
    # if st.button('Open browser'):
    #     webbrowser.open_new_tab(url)
if __name__ == '__main__':
    main()
```

Main html page (main.html):

```
<!DOCTYPE html>
<html lang="en" >
<head>
  <meta charset="UTF-8">
  <title>Login Page</title>
  <link rel='stylesheet' href='https://fonts.googleapis.com/css?family=Rubik:400,700'><link rel="stylesheet" href="./style.css">
</head>
<body>
<!-- partial:index.partial.html -->
```

```
<div class="login-form">
  <form align="center">
    <h1 align="center">VOTER LOGIN</h1>
    <div class="content">
      <div class="input-field">
        <input id="voter" type="text" placeholder="Voter ID" autocomplete="nope">
      </div>
      <div class="input-field">
        <input id="email" type="email" placeholder="Email" autocomplete="nope">
      </div>
      <div class="input-field">
        <input id="pass" type="password" placeholder="Password" autocomplete="new-password">
      </div>
    </div>
    <div>
<!--      <button >Sign in</button>-->
<!--       <button>Register</button>-->
      <input type="button" onclick="javascript:validate()" value="LOGIN" class="action" />
<!--              <input type="button" onclick="location.href='https://google.com';" value="Go to Google" class="action"/>-->
    </div>
```

```
    <br>
    <div class="link1">
    <a href="https://eci.gov.in/voter/voter-registration/" target="_blank">NOT
REGISTERED? REGISTER HERE!</a>
    </div>
    <br>
  </form>
  <div>
    <img src="img.png" width="500" height="350" alt="VOTE IMG  ">
  </div>
</div>
<!-- partial -->
  <script type="text/javascript">
function validate()
{
    if(   document.getElementById("voter").value == "123456"
       &&                document.getElementById("email").value                ==
"vudhanthineeraja@gmail.com"
       && document.getElementById("pass").value == "1236")
    {
        alert( "validation succeeded" );
        location.href="http://localhost:8501/";
    }
```

```
    else
    {
        alert( "validation failed" );
        location.href="num2.html";
    }
}
</script>
</body>
</html>
```

Voter voting page (num0.html):

```
<!DOCTYPE html>
<html lang="en" >
<head>
  <meta charset="UTF-8">
  <title>REGISTERED</title>
  <linkrel='stylesheet'
href='https://fonts.googleapis.com/css?family=Rubik:400,700'><link
rel="stylesheet" href="./style.css"
</head>
<body>
<!-- partial:index.partial.html -->
<div class="login-form">
  <form align="center">
```

```
    <div>
   <img src="votehere.jpg" width="500" height="350" alt="VOTE IMG  ">
  </div>
    <br>
   <h2>PLEASE SELECT THE CANDIDATE: </h2>
    <br>
    <p> The candidate you select here, is the candidate you cast your vote for.</p>
    <br>
    <br>
    <input type="radio" id="C1" name="fav_language" value="HTML">
<label for="C1">K. Chandrashekar Rao,TRS  <img src="trs.png" width="150" height="100" alt="VOTE IMG  "></label><br><br>
    <input type="radio" id="C2" name="fav_language" value="HTML">
<label for="C2">Revanth Reddy, INC  <img src="inc.png" width="150" height="100" alt="VOTE IMG  "></label><br><br>
    <input type="radio" id="C3" name="fav_language" value="HTML">
<label for="C3">Bandi Sanjay Kumar,BJP  <img src="bjp.png" width="150" height="100" alt="VOTE IMG  "></label><br><br>
    <input type="radio" id="C4" name="fav_language" value="HTML">
<label for="C4">Bakkan Narasimhulu, TDP  <img src="tdp.png" width="150" height="100" alt="VOTE IMG  "></label><br><br>
    <input type="radio" id="C5" name="fav_language" value="HTML">
<label for="C5">Kunamneni Sambasiva Rao, CPI  <img src="cpi.png" width="150" height="100" alt="VOTE IMG  "></label><br><br>
```

```
    <input type="radio" id="C6" name="fav_language" value="HTML">
<label for="C6">M. Kodandaram, TJS  <img src="tjs.png" width="150" height="100" alt="VOTE IMG  "></label><br><br>
    <br><br>
        <input type="button" onclick="location.href='votingcomplete.html';" value="VOTE" class="action1" />
   <br><br><br>
   <div class="link1">
     <p> Not sure how to vote? Check the FAQ's page through "HELP".</p><br>
   <a style="font-size: 18px; text-decoration: none; color: #f3130b" href="https://eci.gov.in/faqs/" target="_blank">HELP!</a>
   </div>
   <br>
  </form>
</div>
</body>
</html>
```

Voter already voted page (num1.html):

```
<!DOCTYPE html>
<html lang="en" >
<head>
  <meta charset="UTF-8">
  <title>ALREADY VOTED</title>
```

```
 <link rel='stylesheet' href='https://fonts.googleapis.com/css?family=Rubik:400,700'><link rel="stylesheet" href="./style.css">

</head>

<body>

<!-- partial:index.partial.html -->

<div class="login-form">

 <form align="center">

    <div>

  <img src="restricted.png" width="500" height="350" alt="VOTE IMG  ">

 </div>

    <br>

  <h2>LOOKS LIKE YOU HAVE ALREADY <i> VOTED </i></h2> <br>

  <h3>PLEASE CONTACT US IF THIS IS A MISTAKE</h3>

  <div>

  </div>

  <br>

  <div class="link1">

  <a style="font-size: 30px; text-decoration: none; color: #f3130b" href="https://eci.gov.in/contact-us/contact-us/" target="_blank">CONTACT US!</a>

  </div>

  <br>
```

```
  </form>
</div>
</body>
</html>
```

Voter not registered page (num2.html):

```
<!DOCTYPE html>
<html lang="en" >
<head>
  <meta charset="UTF-8">
  <title>NOT REGISTERED</title>
  <link rel='stylesheet' href='https://fonts.googleapis.com/css?family=Rubik:400,700'><link rel="stylesheet" href="./style.css">
</head>
<body>
<!-- partial:index.partial.html -->
<div class="login-form">
  <form align="center">
     <div>
    <img src="register.jpg" width="500" height="350" alt="VOTE IMG  ">
  </div>
     <br>
    <h2>OOPS LOOKS LIKE YOU HAVE NOT REGISTERED TO VOTE</h2>
```

```
    <div>
    </div>
    <br>
    <div class="link1">
    <a style="font-size: 30px; text-decoration: none; color: #f3130b" href="https://eci.gov.in/voter/voter-registration/" target="_blank">REGISTER HERE!</a>
    </div>
    <br>
  </form>
</div></body></html>
```

6 TESTING

6.1 TESTING

The purpose of testing is to discover errors. Testing is the purpose of trying to discover every conceivable fault or weakness in a work product. It provides a way to check the functionality of components, sub-assembles, assemblies and/or a finished product. And it is the process of exercising software with the intent of ensuring that the software system meets its requirements and expectation and does not fail in an unacceptable manner. There are various types of test. Each test type addresses a specific testing requirement.

6.2 TYPES OF TESTS

6.2.1 Manual Testing

Manual testing is the process of testing software by hand to learn more about it, to find what is and isn't working. This usually includes verifying all the features specified in requirements documents, but often also includes the testers trying the software with the perspective of their end user's in mind. There are lots of sophisticated tools on the market to help with manual testing like Test Pad.

6.2.2 Automation Testing

Automation testing is the process of testing the software using an automation tool to find the defects. In this process, testers execute the test scripts and generate the test results automatically by using automation tools. Some of the famous automation testing tools for functional testing are QTP/UFT and Selenium.

6.2.3 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.

6.2.4 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.

6.2.5 Functional Testing

Functional tests provide systematic demonstrations that functions tested are available as specified by the business and technical requirements, system documentation, and user manuals. Functional testing is centered on the following items:

• Valid input: identified classes of valid input must be accepted.

• Invalid input: identified classes of invalid input must be rejected.

• Functions: identified functions must be exercised.

• Output: identified classes of applications must be exercised.

Systems/procedures: interfacing systems or procedures must be invoked.

Organization and preparation of functional tests is focused on requirements, key functions, or special test cases. In addition, systematic coverage pertaining to identify business process flows, data fields, re defined processes, successive processes must be considered for testing. Before functional testing is complete, additional tests are identified and the effective value of current tests is determined.

6.2.6 System Testing

System testing ensures that the entire integrated software system needs requirement. It tests a configuration to ensure known and predictable results. An example of system

testing is the configuration-oriented system integration test. System testing is based on process descriptions and flows, emphasizing pre driven process links and integration points.

6.2.7 White Box Testing

White box testing is a testing in which the software tester has knowledge of the inner workings, structure and language of the software, or at least its purpose. It is used to test areas that cannot be reached from a black box level.

6.2.8 Black Box Testing

Black box testing is testing the software without any knowledge of the inner workings, structure or language of the module being tested. Block box tests, as most other kinds of tests, must be written from a definitive source document, such as specification or requirements document, such as specification or requirements document. It is a testing in which the software under test is treated, as a block box, you cannot “see” into it. The test provides inputs and responds to outputs without considering how the software works.

6.3 TEST CASES

S.No	TestCase	ExpectedOutput	ActualOutput	Result
1	Input with a registered but not yet voted user	Allows to vote and website directs to voting page (num0.html)	Website directs to voting page (num0.html)	Success
2	Input with a registered but voted user	Does not allow user to vote and website directs to restricted voting page (num1.html)	Does not allow user to vote and website directs to restricted voting page (num1.html)	Success
3	Input with a non registered	Does not allow user to vote and website directs to register to vote page (num2.html)	Does not allow user to vote and website directs to register to vote page (num2.html)	Success

7 OUTPUT SCREENS

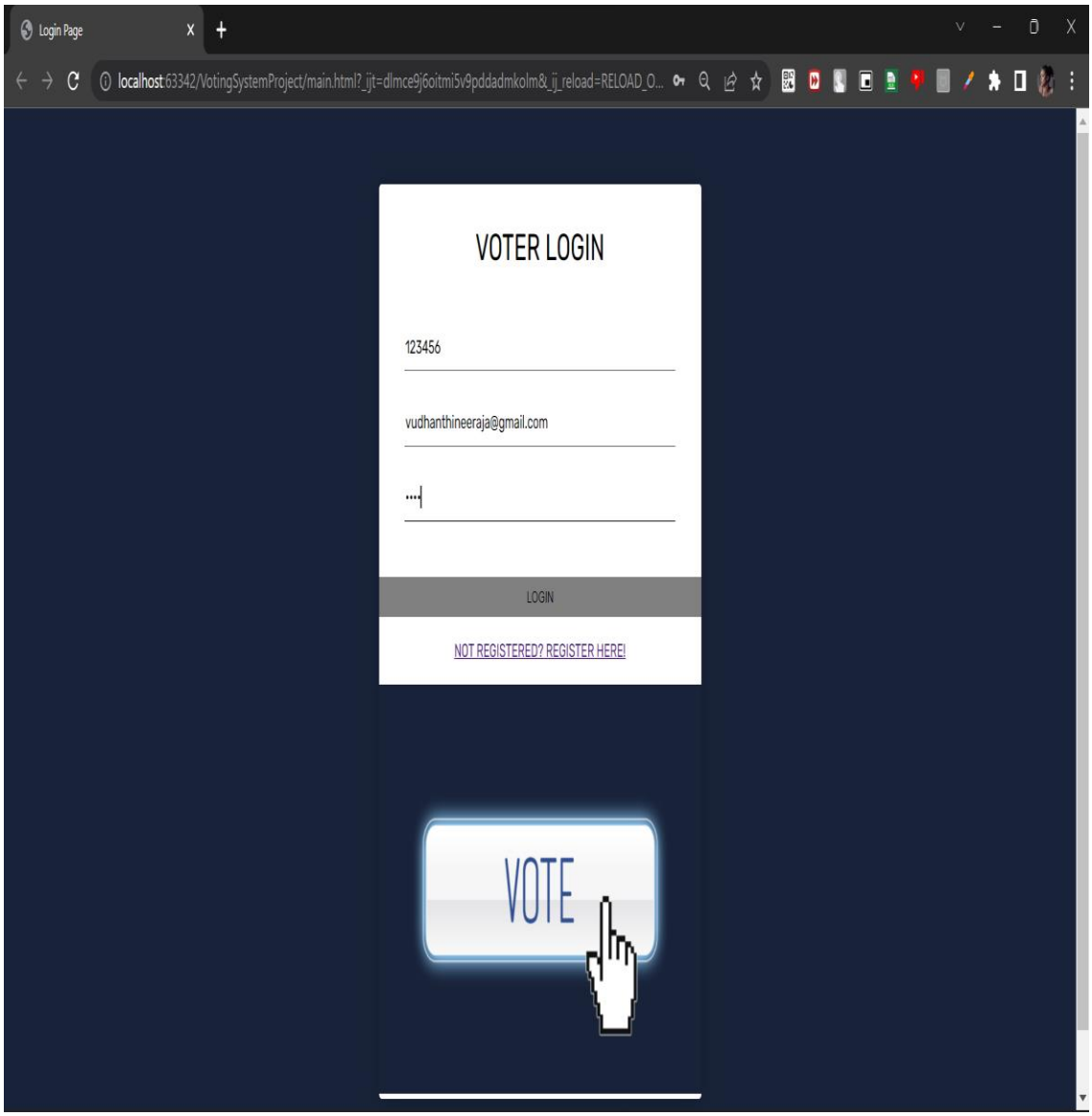

Fig 4.1 Login Screen

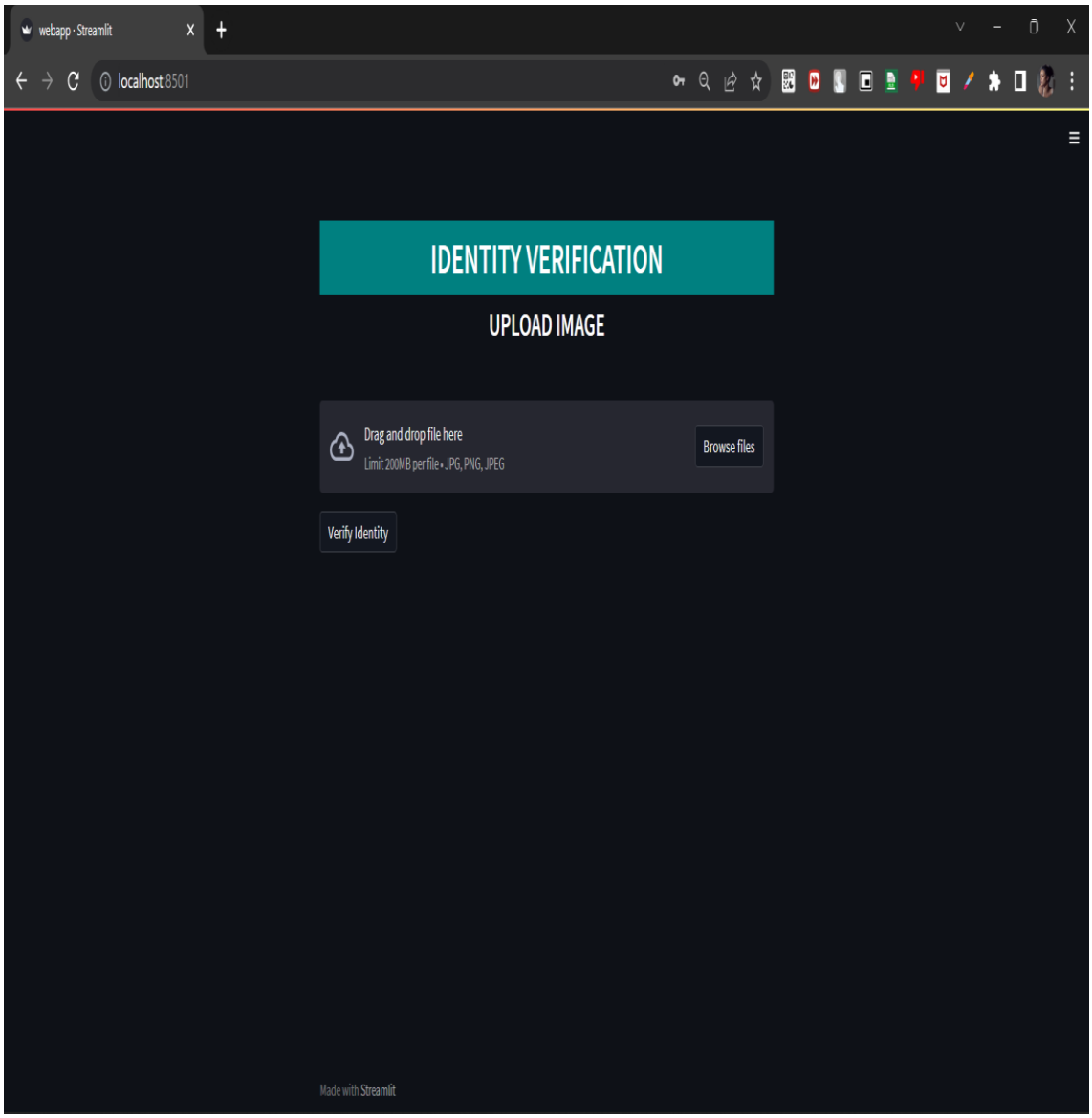

Fig 4.2 Identity Verification screen where image is uploaded

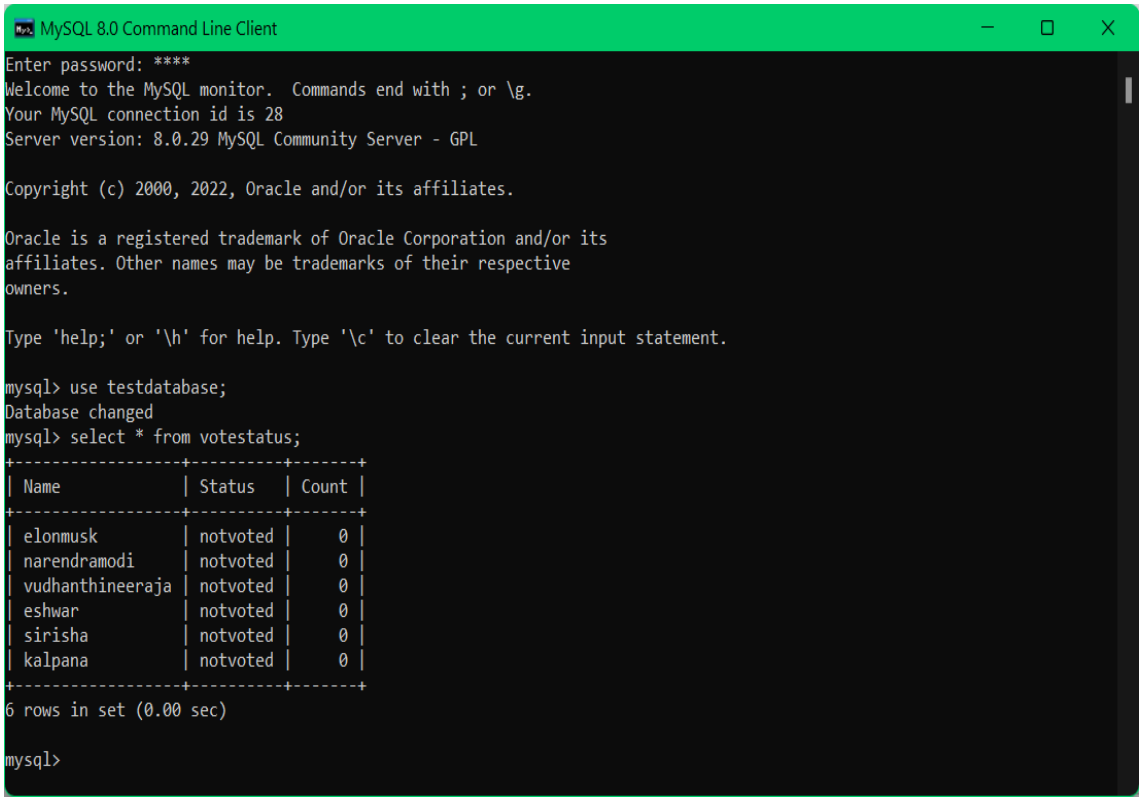

Fig 4.3 The database before the voter has voted

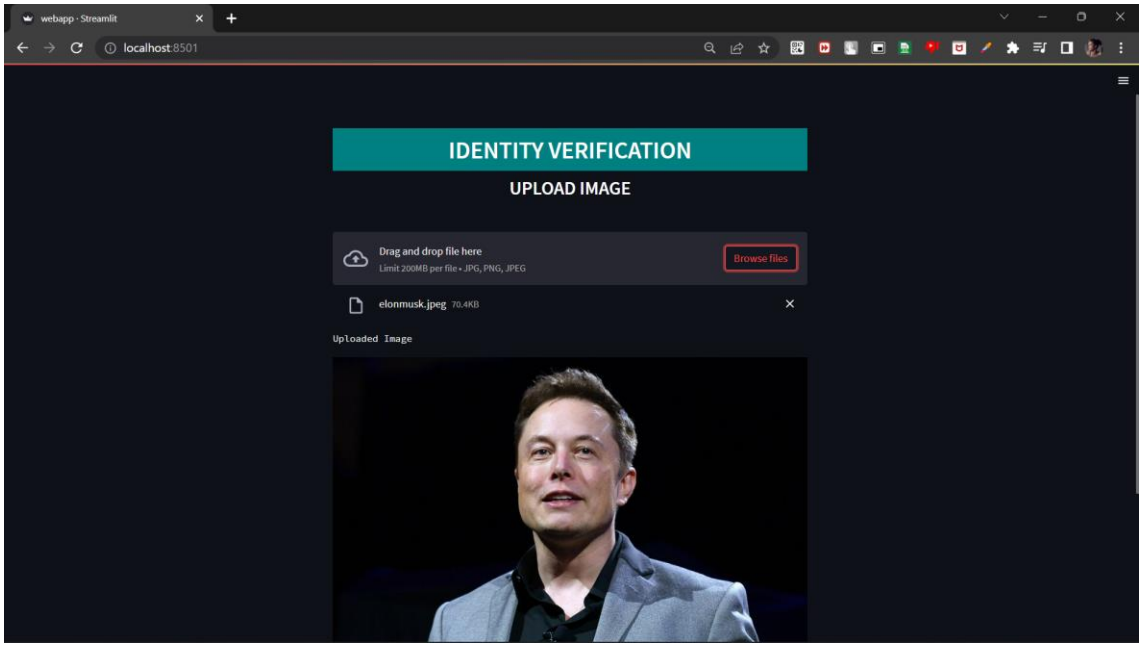

Fig 4.4 Identity Verification screen after image is uploaded

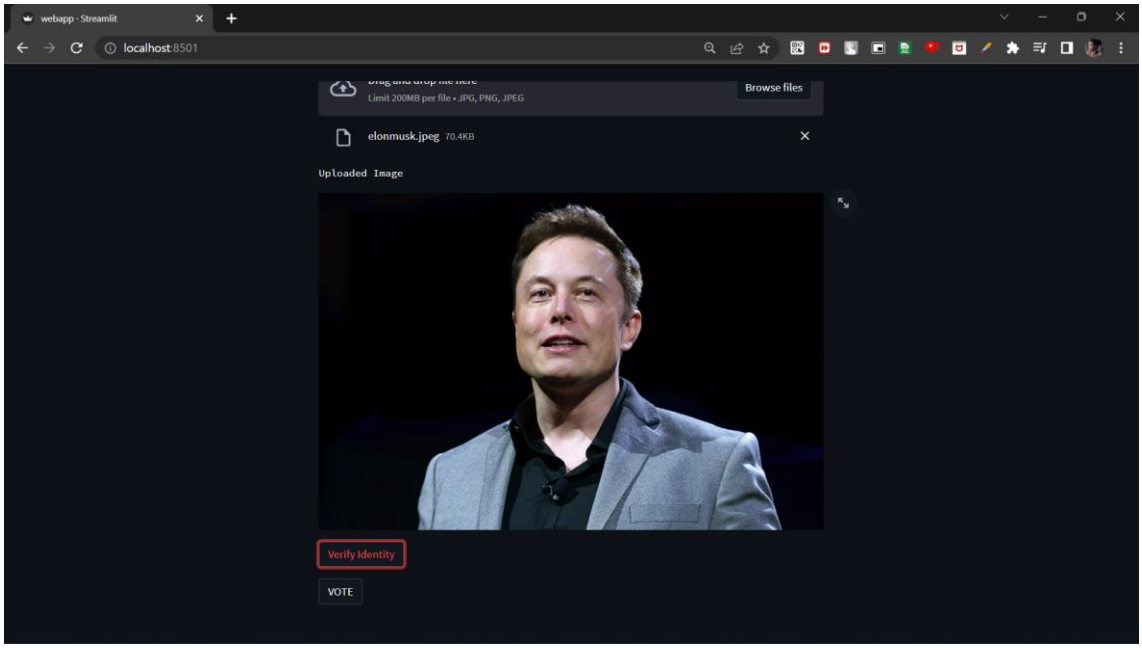

Fig 4.5 Vote button displayed after voter has been verified as registered.

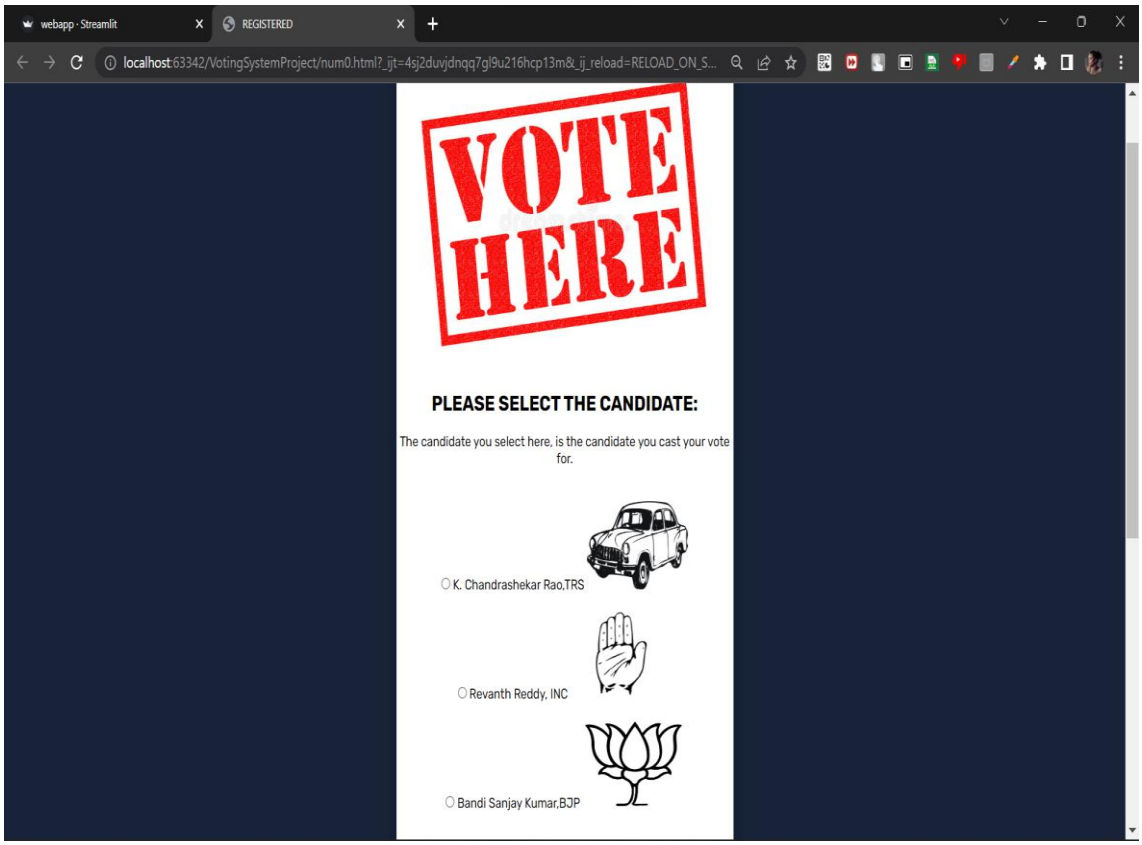

Fig 4.5.1 Voting Screen after the voter has been verified.

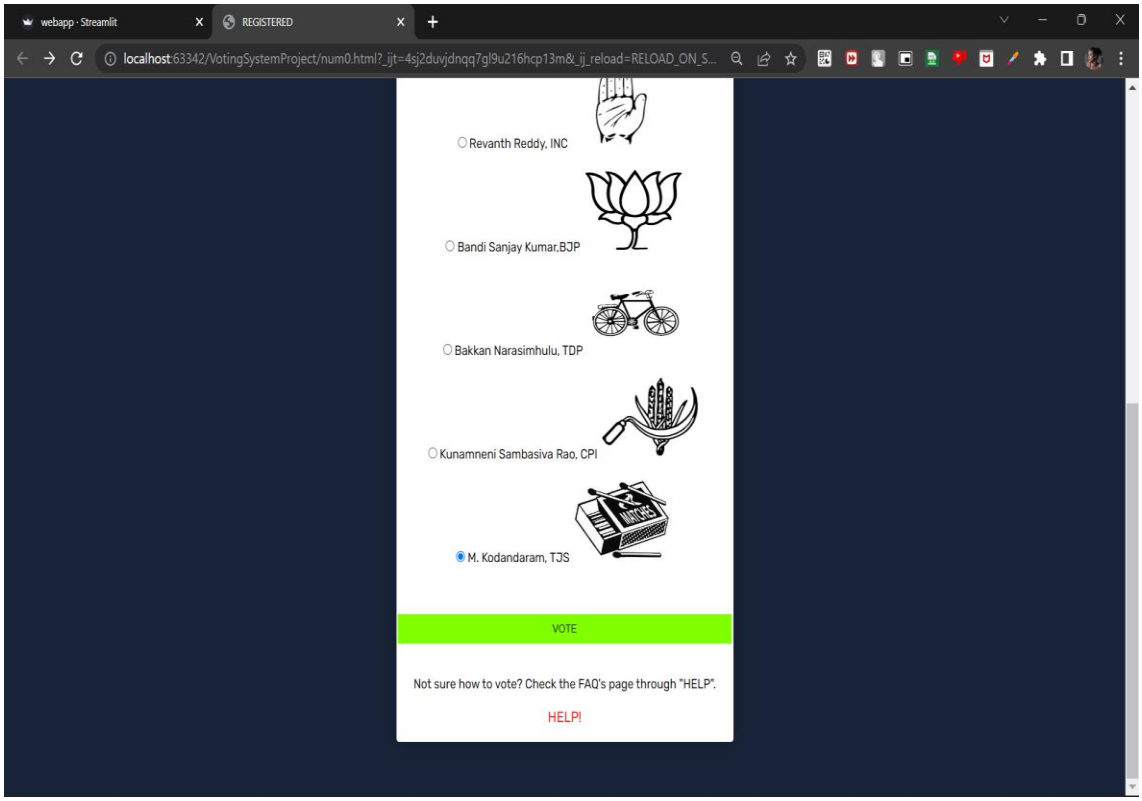

Fig 4.5.2 Voting Screen after the voter has been verified.

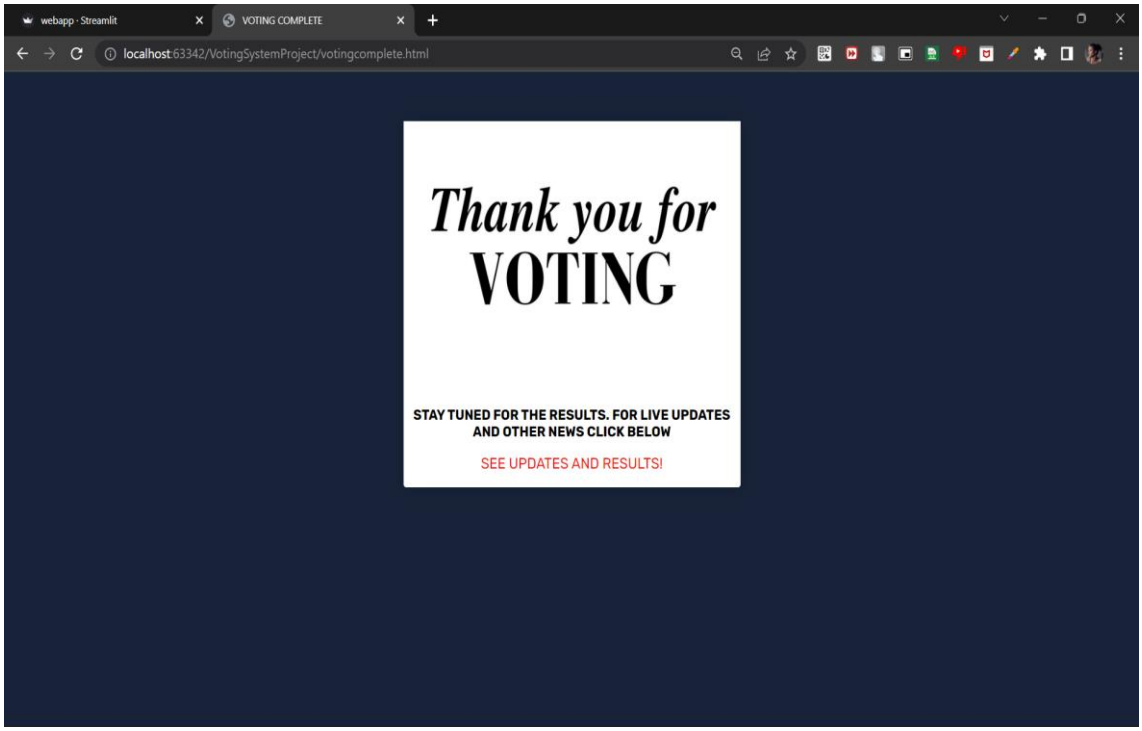

Fig 4.6 Output screen after the voter has casted their vote

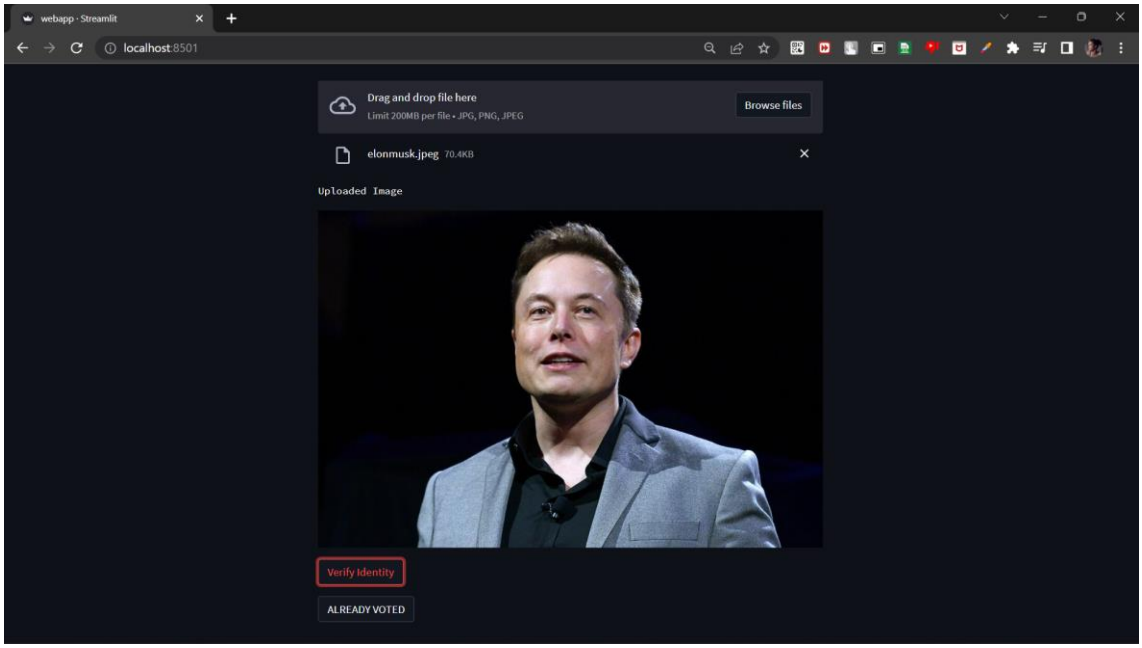

Fig 4.7 Already Vote button displayed after voter has been verified has already voted.

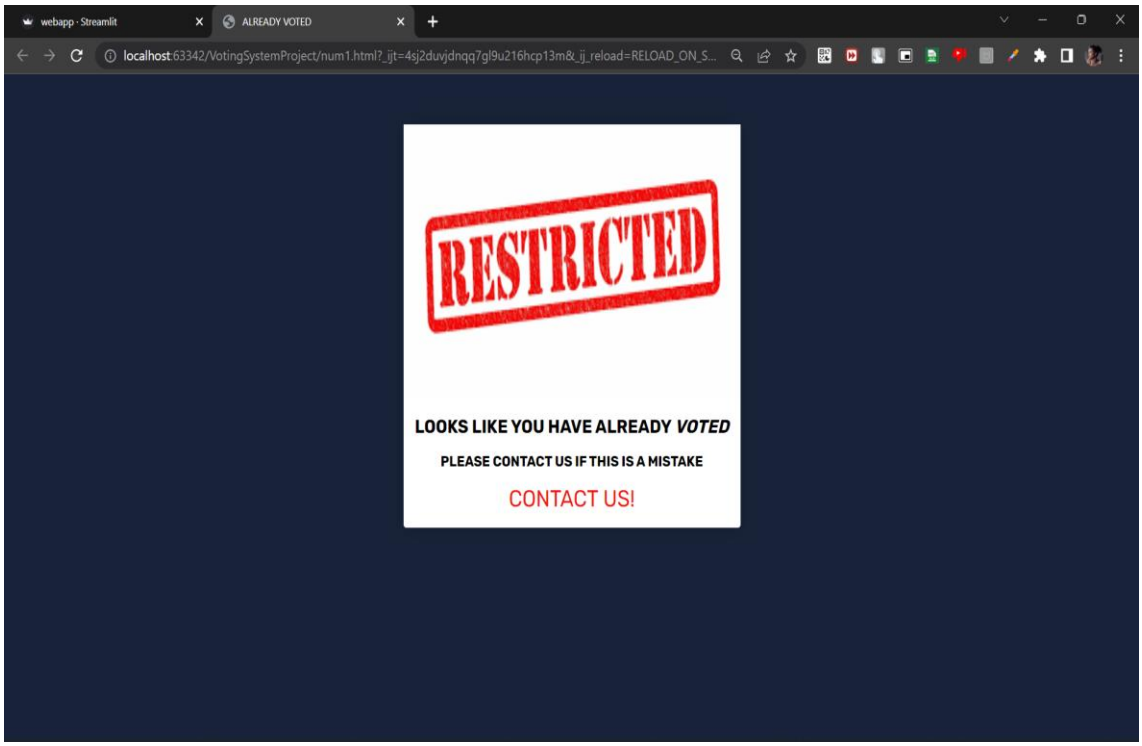

Fig 4.8 Restricted output screen to restrict voter to vote again.

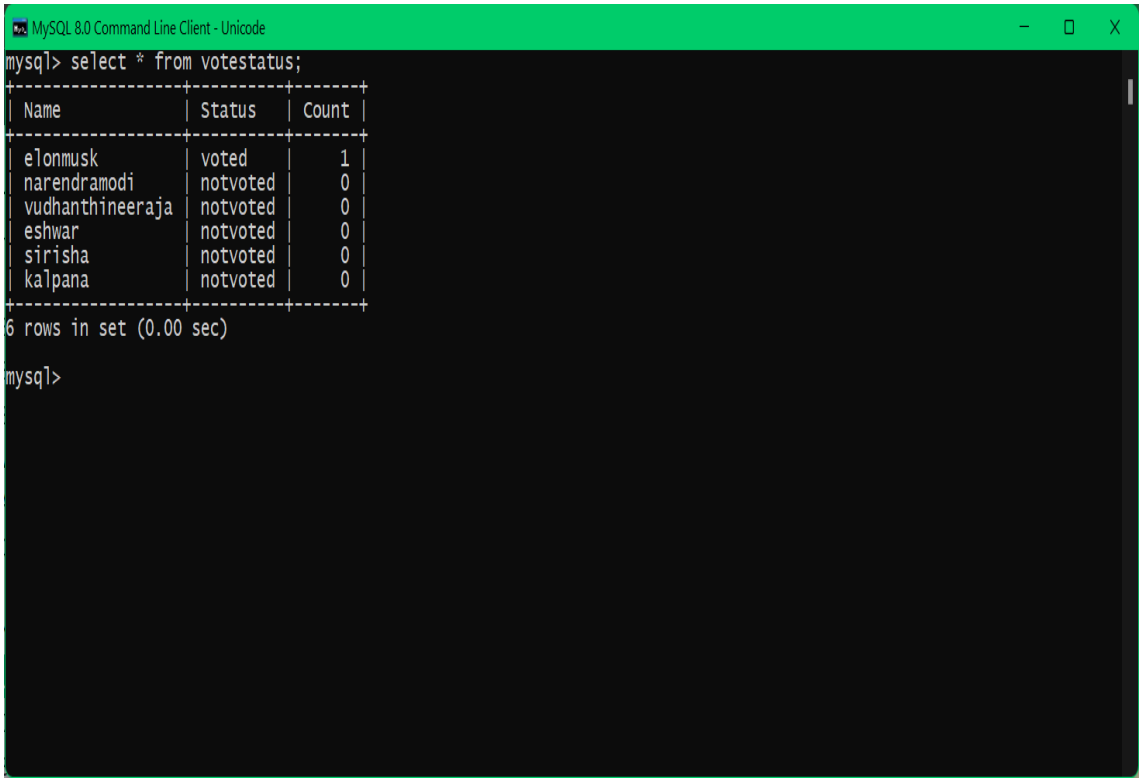

Fig 4.9 The database after the voter has voted

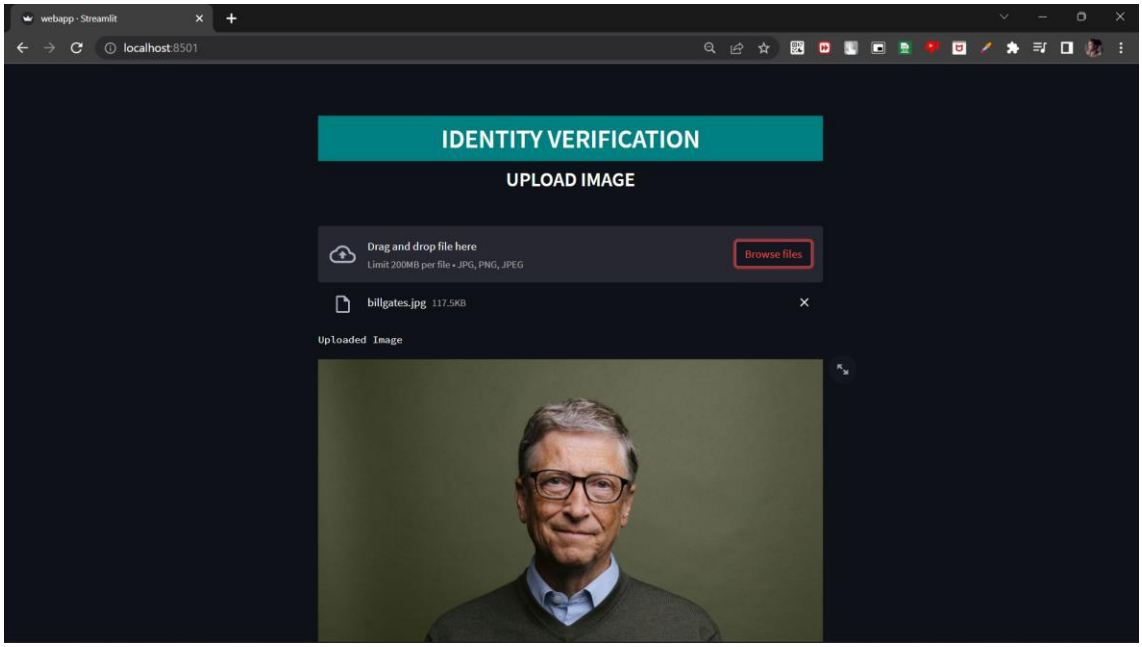

Fig 4.10 Identity Verification screen after non registered voter image is uploaded

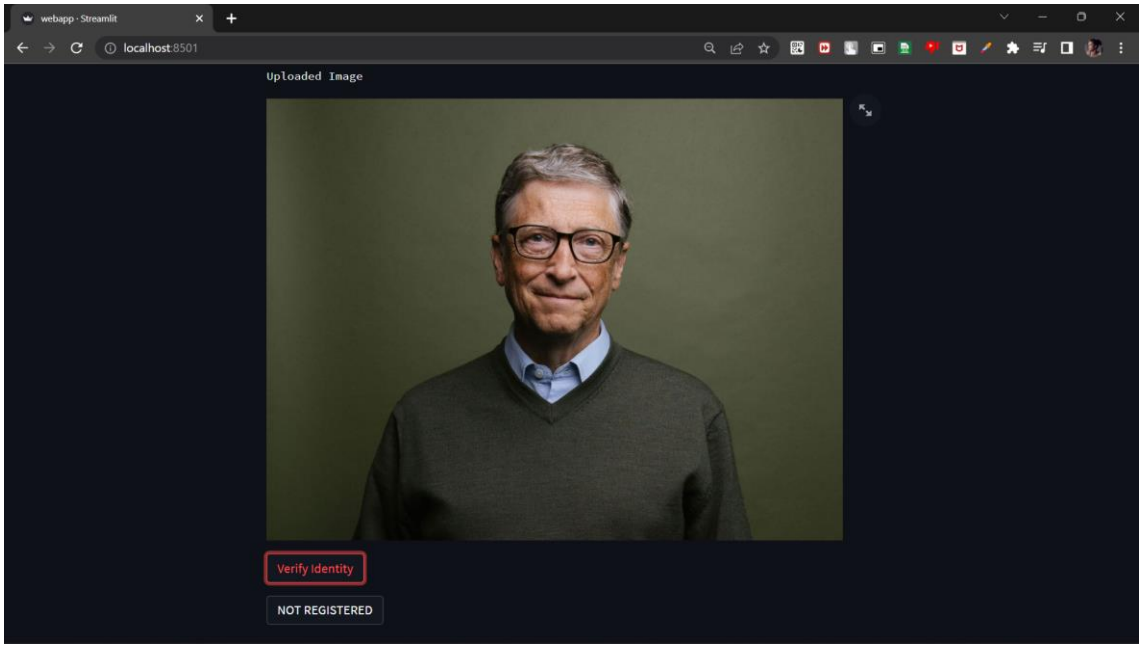

Fig 4.11 Not Registered button displayed after identity has been is not authentic.

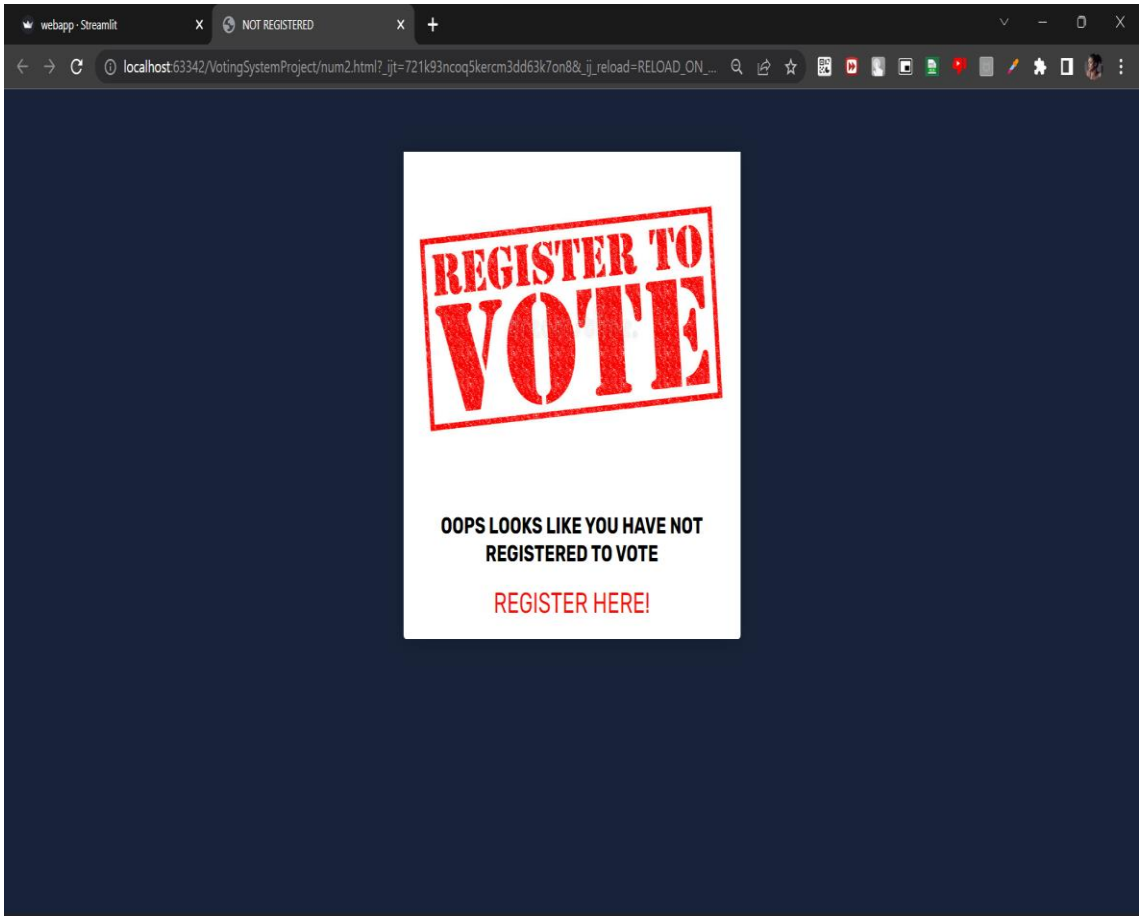

Fig 4.12 Output page to allow non registered voters to registers

8 CONCLUSION

8.1 CONCLUSION

Online Voting Systems have many advantages over the traditional voting system. Some of these advantages are less cost, faster generation results, easy accessibility, accuracy, and low risk of human and mechanical errors. It is very difficult to develop online voting system which can allow security and privacy on the high level. Future development focused to design a system which can be easy to use and will provide security and privacy of votes on acceptable level by proper authentication and processing section.. It is easy to use and it is less time consuming. It is very easy to debug.

This Online Voting system will manage the Voter's information by which voter can login and use his voting rights. The system will incorporate all features of Voting system. It provides the tools for maintaining voter's vote to every party and it count total no. of votes of every party. There is a database which is maintained by the ELECTION COMMISION OF INDIA in which all the names of voter with complete information is stored.

The main aspect behind OVS is that it enabled us to bring out the new ideas that were sustained within us for many for many days. This project offers the voters to cast easily through interne. Vote counting is also made easy by the OVS since it's just a matter of querying the database. Developing a good system is critical to the success of the system to prevent system failures and to gain wide acceptance as the best method available. A good OVS system requires ten characteristics which this system already has. These are: In analyzing, designing, implementing, and maintaining standards, we considered these characteristics as the foundation. These standards were made national. OVS will be an inexpensive, and less time consuming method once a system exhibiting national standards and the above mentioned characteristics is implemented. The results showed that the voter is allowed to vote with mean ROC AUC scores of 96.51 and 96.33%, respectively. Therefore, the use of machine learning-based automated high-accuracy technologies and other facial recognition algorithms may provide further ROC

AUC scores and increase the accuracy of detecting the face of the voters and accurately allowing them to cast their vote.

8.2 FUTURE ENHANCEMENTS

Despite different views on the integrity of the online and electronic voting systems, they have always been deemed secure with some fundamental security and anonymity principles. Numerous electronic systems have been proposed and implemented but some suspicion has been raised regarding the integrity of elections due to detected security vulnerabilities within these systems. Electronic voting and online voting systems, to be successful, requires a more transparent and secure approach, than is offered by current protocols. The future enhancements involves a protocol developed on blockchain technology. The underlying technology used in the voting system is a payment scheme, which offers anonymity of transactions, a trait not seen in security protocols used for online and electronic systems to date. The block chain protocol when integrated to the online voting system offers anonymity of voter transactions, while keeping the transactions private, and the election transparent and secure.

9 BIBLIOGRAPHY

[1]Himanshu Agarwal and G.N. Pandey “Online Voting System for India Based on AADHAAR ID” 2013 Eleventh International Conference on ICT and Knowledge Engineering

[2] Smita B. Khaimar, P. Sanyasi Naidu, Reena Kharat “Secure Authentication for Online Voting System”

[3]Shivendra Katiyar, Kullai Reddy Meka, Ferdous A. Barbhuiya, Sukumar Nandi “Online Voting System Powered By Biometric Security” 2011 Second International Conference on Emerging Applications of Information Technology.

[4]Anderson C. (2006). How to Rig a Democracy: A Timeline of Electronic Voting in the United States. The Independent. Retrieved November 28, 2006
from: http://www.indypendent.org/?p=608

[5]Bellis, M. (2007). The History of Voting Machines. Retrieved November 9, 2006 from: http://inventors.about.com/library/weekly/aa111300b.htm

[6]Cranor, L.F., & Cytron, R.K. (1996). Design and Implementation of a Security-Conscious Electronic Polling System. Washington University Computer Science Technical Report (WUCS). Retrieved October 9, 2006
from: http://www.acm.org/crossroads/ords2-4lvotinu.html

[7] Umang Shah, Trupt Shah, Marteen Kansagara, Saagar Daxini, Biometric Secured Voting Machine to Avoid Bogus Voting Based on AADHAR CARD , International Journal of Innovative Research in Computer and Communication Engineering , Vol. 3, Issue 3, March 2015

[8] Rudrappa B. Gujanatti, Shivaram N. Tolanur, Murughendra
S. Nemagoud, Shanta S. Reddy, Sangameshwar Neelagund,
A Finger Print based Voting System , International Journal of Engineering Research & Technology (IJERT),ISSN: 2278-0181. Vol. 4 Issue 05, May-2015

[9] M.Sudhakar, B.Divya Soundarya Sai, Biometric System Based Electronic Voting Machine Using Arm9
Microcontroller IOSR Journal of Electronics and Communication Engineering (IOSR-JECE) e-ISSN: 2278- 2834,p- ISSN: 2278-8735.Volume 10, Issue 1, Ver. II (Jan – Feb.

2015)

[10] Soumyajit Chakraborty, Siddhartha Mukherjee, Bhaswati Sadhukhan, Kazi Tanvi Yasmin, Biometric Voting System using Aadhar Card in India , International Journal of Innovative Research in Computer and Communication Engineering (An ISO 3297: 2007 Certified Organization) Vol. 4, Issue 4, April 2016 .

[11] P. R. Lorenzo, J. Nalepa, M. Kawulok, L. S. Ramos, and J. R. Pastor, "Particle swarm optimization for hyper-parameter selection in deep neural networks," in Proceedings of the genetic and evolutionary computation conference, 2017, pp. 481-

[12] Q. Yang, C. Ling, X. Chai, and R. Pan, "Test-cost sensitive classification on data with missing values," IEEE Transactions on Knowledge & Data Engineering, vol. 18, no. 5, pp. 626--638, 2006.

10.PLAGARISM REPORT