

**A MINI PROJECT REPORT**  
*ON*  
**SMART VOTING SYSTEM**  
*Submitted in partial fulfilment of the requirement*  
*For the award of the degree of*  
**BACHELOR OF TECHNOLOGY**  
*IN*  
**COMPUTER SCIENCE AND ENGINEERING**  
*By*

<b>G. Bhargava</b>	<b>19P61A0567</b>
<b>G. Rohit</b>	<b>19P61A0572</b>
<b>G. Pranay</b>	<b>19P61A0577</b>

*Under the esteemed guidance of*

**Mrs. P. Subhadra**

*Associate Professor*

*Dept. of CSE*



Aushapur(V), Ghatkesar(M), Hyderabad, Medchal – Dist, Telangana – 501 301.

**December – 2022**



Aushapur(V), Ghatkesar(M), Hyderabad, Medchal – Dist, Telangana – 501 301.

**DEPARTMENT  
OF  
COMPUTER SCIENCE & ENGINEERING**

**CERTIFICATE**

*This is to certify that the mini project titled “SMART VOTING SYSTEM” submitted by G. Bhargava (19P61A0567), G. Rohit (19P61A0572), G. Pranay (19P61A0577) in B.tech IV-I semester Computer Science & Engineering is a record of the bonafide work carried out by them.*

*The results embodied in this report have not been submitted to any other University for the award of any degree.*

**INTERNAL GUIDE**

**Mrs. P. Subhadra**  
(Associate Professor,CSE)

**HEAD OF THE DEPARTMENT**

**Dr. G. Sreeram**

**EXTERNAL EXAMINER**

## **DECLARATION**

We, **G. Bhargava, G. Rohit, G. Pranay**, bearing hall ticket numbers **(19P61A0567, 19P61A0572, 19P61A0577)** hereby declare that the mini project report entitled “**SMART VOTING SYSTEM**” under the guidance of **Mrs. P. Subhadra**, Associate Professor, Department of Computer Science and Engineering, **Vignana Bharathi Institute of Technology, Hyderabad**, have submitted to Jawaharlal Nehru Technological University Hyderabad, Kukatpally, in partial fulfilment 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 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.

**G. Bhargava(19P61A0567)**

**G. Rohit (19P61A0572)**

**G. Pranay(19P61A0577)**

## **ACKNOWLEDGEMENT**

We are extremely thankful to our beloved Chairman, **Dr. N. Goutham Rao** and Secretary, **Dr. G. Manohar Reddy** who took keen interest to provide us the infrastructural facilities for carrying out the project work.

Self-confidence, hard work, commitment, and planning are essential to carry out any task. Possessing these qualities is sheer waste, if an opportunity does not exist. So, we whole- heartedly thank **Dr. P. V. S. Srinivas, Principal**, and **Dr. G. Sreeram**, Head of the Department, Computer Science and Engineering for their encouragement and support and guidance in carrying out the project.

We would like to express our indebtedness to the project coordinators, **Dr. K. Kalaivani**, Associate Professor, Department of CSE, **Mr. G. Srikanth Reddy**, Associate Professor, Department of CSE, **Mrs. P. Subhadra**, Associate Professor, Department of CSE, for their valuable guidance during the course of project work.

We thank our Project Guide, **Mrs. P. Subhadra**, Associate Professor, Department of CSE, for providing us with an excellent project and guiding us in completing our mini project successfully.

We would like to express our sincere thanks to all the staff of Computer Science and Engineering, VBIT, for their kind cooperation and timely help during the course of our project. Finally, we would like to thank our parents and friends who have always stood by us whenever we were in need of them.

## **ABSTRACT**

Digital voting is an easy and efficient way to conduct elections and avoid any errors, misleading's etc. We can start the process by a simple login to a portal that is accessible during the period of voting. A voter can look over the candidates participating in the election which will be provided in the portal. Every citizen can cast their vote by providing their details which will be verified through the database originated. By wrapping up all the verification steps a user will be accepted to cast a vote once only.

The latest technological developments can help us to overcome the problems faced during the elections, reducing the burden of the election committee. Biometric verification, Facial recognition, Retina scanner and One time password generators can help us in validating the user information. Hence protecting the protocol of rigging during the elections. A simple digital booth can be set up at rural areas where the people are unaware of the digital process. A user can save an amount of time by digital voting instead of standing in a long queue at the pooling booths. The election committee can assess the results quickly with the help of data management system and compute the results in a much easier way. Although the lack of knowledge regarding the voting process and data security, hacking might be a concern for this system it might become an easier way of practice.

# CONTENTS

CHAPTER	PAGE NO.
LIST OF FIGURES	
<b>1. Introduction</b>	
1.1. Introduction to system	1
1.2. Problem Detection	1
1.3. Objectives	1
1.4. Aim of the project	1
<b>2. Literature Survey</b>	
2.1. Existing system	3
2.2. Proposed system	3
2.3. Scope of the project	4
<b>3. Analysis</b>	
3.1. Technical Feasibility	5
3.2. Operational Feasibility	5
3.3. Economical Feasibility	5
<b>4. Hardware and Software Requirements</b>	
4.1. Software requirement	6
4.2. Hardware requirement	6
<b>5. System Design</b>	
5.1. UML Diagrams	9
<b>6. Results and Performance evaluation</b>	14
<b>7. Conclusion and Future work</b>	32
<b>8. References</b>	33

## **LIST OF FIGURES**

<b>S.NO.</b>	<b>FIGURE NAME</b>	<b>PAGE NO.</b>
1.	Use case diagram	10
2.	Class diagram	11
3.	Activity diagram	12
4.	Sequence diagram	13
5.	Home page	29
6.	Login page	30
7.	Voting page	30
8.	Dataset page	31
9.	Administration page	31

# CHAPTER – 1



# **1. INTRODUCTION**

## **1.1. INTRODUCTION TO THE SYSTEM**

Smart voting system is an easy and efficient way to conduct elections and avoid any errors, misleading's etc. The smart voting system is a procedure of conducting elections in a semi-online mode. The voters can cast their vote by verifying the voter card details presented to them. An online validation can be done using two step authentication process, which includes password generation to the user id and face recognition for confirming the person identity.

## **1.2. PROBLEM STATEMENT**

The existing system of voting consumes a lot of time for all the procedures. It also includes a lot of man power in order to conduct the elections fairly. We also need to tackle the long waiting process of counting votes. A lot f paper work and human resources can be rested by using several other methods.

## **1.3. OBJECTIVE**

The objective of this project is to conduct an election in an easy manner. The online execution of the voting system enables the voters to cast their vote simply by verifying their identity by sitting at home. It also avoids any rigging process.

## **1.4. AIM OF THE PROJECT**

The main aim of this project is to conduct an election in an easy manner. The online execution of the voting system enables the voters to cast their vote simply by verifying their identity by sitting at home. It also avoids any rigging process. It can also help in easy counting system for quick announcement of result.

# CHAPTER –2

## 2. LITERATURE SURVEY

**Patil Rahul et.al** suggest a system for E-Voting using face detection and finger print recognition which uses various algorithms and techniques such as Eigen face, AES, and besian classifier for providing highly reliable and secure system of voting. In this system the voting is done by identifying the face image and finger prints of the electorate who is going to cast their votes. The electorate information are verified by capturing the face image through web camera along with finger print and the electorate is allow to cast their vote after checking the details with database information. The votes are stored and later the result is declared.

**Ratnaprabha et.al** build an E-/voting system using “Auduino Software” to capture the face and finger print of an individual and match the finger print and face image to the details in the “Aadhar” which are already stored in database using Lab view and compare the details which are already stored in PC and the casted vote is done but not the accurate result is dispatched. This system just avoids duplication of votes but it is not the robust one.

**Priyataneja et.al** anticipated an online voting system by face detection. In this system various algorithms such as Gabor filter technique, Eigen face technique, Gold ratio technique and Line Edge Map (LEM) techniques to detect the face of an individual without any noise. This system is embedded with mobile phones and the electorate is permit to cast their votes through their mobile phone at their mobility and convenience. This system obtains information from the database which stores details of the particular voter through Aadhar database. Due to usage of more number of techniques leads to complexity of the system and also the voter should aware of their availability of network they use because the voting is done through their mobile phone these are the main issues in this system. This system uses front camera of the mobile to capture the face of the electorate so the phones without front camera is not suitable for this approach.

**Neha Gandhi** designs a secure online voting system using biometrics and stenography to give biometric security as well as password security to electorate account. In this model a secret key is used along with voter image as key image to produce a new image to perform electorate authentication and to minimize risk factors such as hacking. It also provides protection against fraudulent behaviours. The main drawback is that process of encoding the image along with key is a long term process.

**Niranjan malwade et.al** proposed smart voting system with face recognition. In this system the electorate use android applications which are downloaded from internet and the authentication will be done using system configuration and face recognition. The main drawback or issues in this system is that it can be used only in android mobiles i.e., this system is applicable only for android platform.

## **2.1. EXISTING SYSTEM**

The existing system is manual, with users and candidate information maintained in books. Users must queue for extended periods of time to vote. Unwanted and incorrect votes are cast. Vote counting is done manually, which takes a long time and results in erroneous counts. It is quite difficult to save historical data.

In the existing system, physical attendance is required at the time of election polling or vote counting. If elections are conducted in existing system model in the pandemic time, then there is sure spread of disease like COVID, which happened in the recent elections in India. It is difficult to maintain important information in books. More manual hours are needed for counting of votes. It is tedious to manage historical data which needs much space to keep all the information regarding the voters and the candidates.

## **2.2. PROPOSED SYSTEM**

The Smart Voting System is a software application which avoids more manual hours that need to spend in record keeping and calculating votes. Through this the users and the candidates are registered online. Their information is stored in the database the admin can easily access the details of the voters and the candidates.

This application keeps the data in a centralized way which is available to all the users simultaneously. It is very easy to manage historical data in database. They can easily use the tool that decreases manual hours spending for normal things and hence increases the performance. The voters are allowed to vote online they can even vote by sitting at home. Every User allowed to vote only once so there is no chance of duplicated votes. The proposed system does not require any physical presence during vote polling or counting. So it is very easy to conduct elections even during the pandemic situations without any spread of disease or human live losses. The proposed system has good authentication so only authorized person can able to vote and also cannot vote multiple types.

### **2.3. SCOPE OF THE PROJECT**

The online portal is easily accessed through the internet by anyone across the globe. So, it is easy to get into the system and cast their vote with simple steps to be followed. The results can also be displayed through the portal in a very quick manner using the database queries.

# CHAPTER – 3

### **3. ANALYSIS**

The major step in analysis is to verify the feasibility of the proposed system. “All projects are feasible given unlimited resources and infinite time.” But, both resources and time are scarce. Project should confirm to be time effective and should be optimal in their consumption of resources. This plays a constant role in approval of any project.

Three key considerations involved in the feasibility analysis are

- ◆ Technical Feasibility
- ◆ Operational Feasibility
- ◆ Economical Feasibility

#### **3.1. Technical Feasibility:**

This study is carried out to check the technical feasibility, that is, the technical requirements of the system. Any system developed must not have a high demand on the available technical resources. This will lead to high demands on the available technical resources. This will lead to high demands being placed on the client. The developed system must have a modest requirement, as only minimal or null changes are required for implementing this system.

#### **3.2. Operational Feasibility:**

The aspect of study is to check the level of acceptance of the system by the user. This includes the process of training the user to use the system efficiently. The user must not feel threatened by the system, instead must accept it as a necessity. The level of acceptance by the users solely depends on the methods that are employed to educate the user about the system and to make him familiar with it. His level of confidence must be raised so that he is also able to make some constructive criticism, which is welcomed, as he is the final user of the system.

#### **3.3. Economical Feasibility:**

This study is carried out to check the economic impact that the system will have on the organization. The amount of fund that the company can pour into the research and development of the system is limited. The expenditures must be justified. Thus the developed system as well within the budget and this was achieved because most of the technologies used are freely available. Only the customized products had to be purchased.

# CHAPTER – 4



## **4. HARDWARE AND SOFTWARE REQUIREMENTS**

### **4.1. Hardware Requirements**

- **RAM** : 2 GB or above
- **Hard Disk** : 40 GB or above
- **Key Board** : Standard Windows Keyboard
- **Mouse** : Two or Three Button Mouse
- **Monitor** : SVGA 21"

### **4.2. Software Requirements**

- **Operating system** : Windows 7 Ultimate or above
- **Coding Language** : Python
- **Front-End** : Python
- **Back-End** : Django

# CHAPTER– 5

## **5. SYSTEM DESIGN**

System design is the transition from a user-oriented document to programmers or database personnel. The design is a solution, specifying how to approach to the creation of a new system. This is composed of several steps. It provides the understanding and procedural details necessary for implementing the system recommended in the feasibility study. Designing goes through logical and physical stages of development. Logical design reviews the present physical system, prepare input and output specification, details of implementation plan and prepare a logical design walkthrough.

The database tables are designed by analyzing functions involved in the system and format of the fields is also designed. The fields in the database tables should define their role in the system. The unnecessary fields should be avoided because it affects the storage areas of the system. Then, in the input and output screen design, the design should be made user friendly. The menu should be precise and compact.

### **SOFTWARE DESIGN**

In designing the software, the following principles are followed:

- Modularity and partitioning: software is designed in such a way that; each system should consist of hierarchy of modules and serve to partition into separate function.
- Coupling: modules should have little dependency on the other modules of a system.
- Cohesion: modules should carry out the operations in a single processing function.
- Shared use: avoid duplication by allowing a single module which is called by other, that needs the function it provides.

## **INPUT DESIGN**

Considering the requirements, procedures are adopted to collect the necessary input data in most efficiently designed format. The input design has to be done keeping in view that, the interaction of the user with the system should be in the most effective and simplified way. Also, the necessary measures are taken for the following

- Controlling the amount of input
- Avoid unauthorized access to the users
- Eliminating the extra steps
- Keeping the process simple
- At this stage the input forms and screens are designed.

## **OUTPUT DESIGN**

All the screens of the system are designed with a view to provide the user with easy operations in a simpler and efficient way, with minimum key strokes possible. Important information is emphasized on the screen. Almost every screen is provided with no error and important messages and option selection facilitates. Emphasis is given for faster processing and speedy transactions between the screens. Each screen assigned to make it as much user friendly as possible by using interactive procedures. In other words, we can say that the user can operate the system without much help from the operating manual.

## 5.1. UML DIAGRAMS

### Unified Modelling Language

The Unified Modelling Language (UML) is a standard language for specifying, visualizing, constructing, and documenting the artefacts of software systems, as well as for business modelling and other non-software systems. The UML represents a collection of best engineering practices that have proven to be successful in the modelling of large and complex systems. The UML is a very important part of developing object-oriented software and the software development process. UML mostly uses graphical notations to express the design of software projects. Using the UML helps the project teams to communicate, explore potential designs and validate the architectural design of the software.

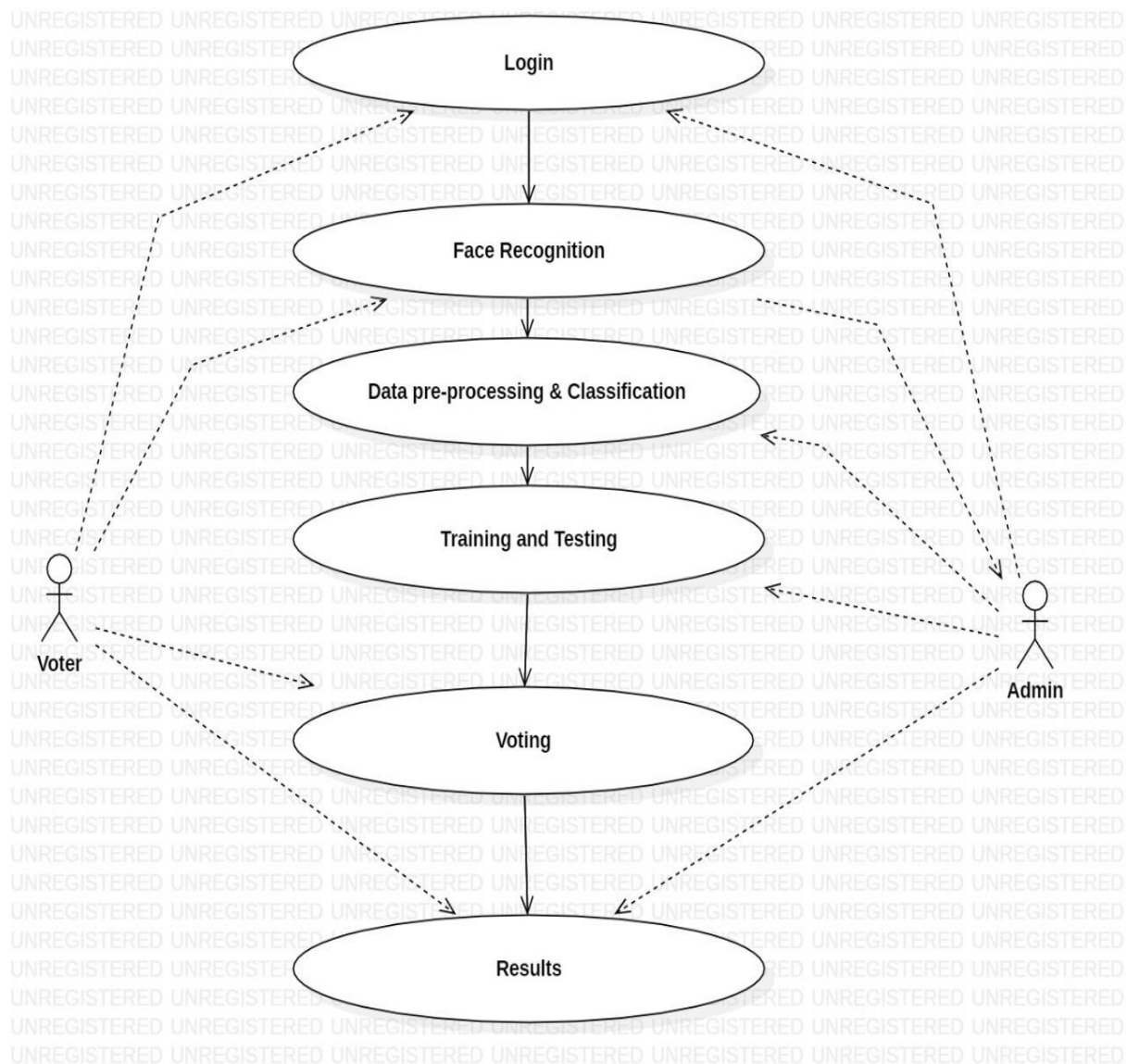
The Unified Modelling Language (UML) is a standard language for writing software blue prints. The UML is a language for

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

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.

## USE CASE DIAGRAM

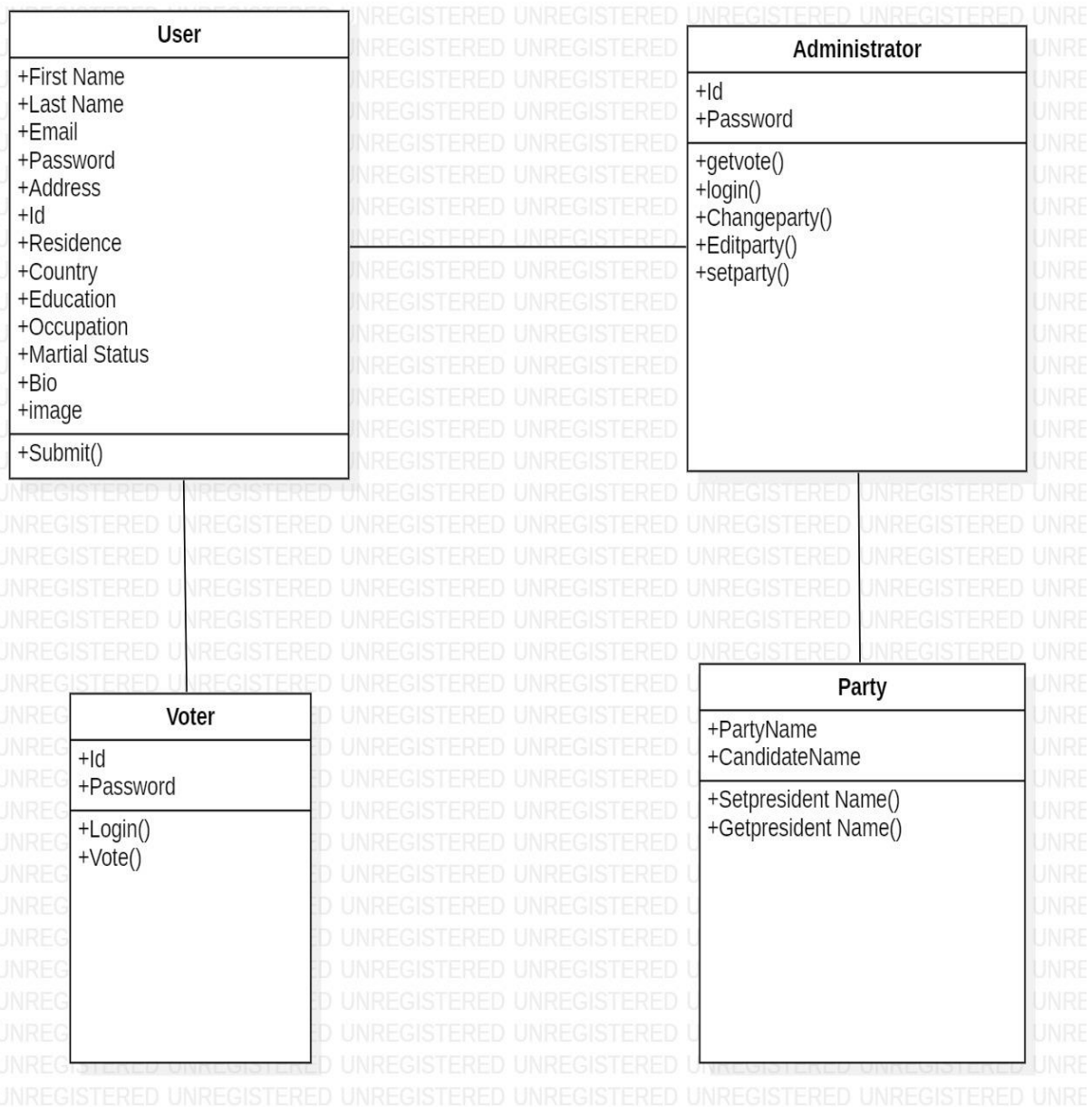
- Use case Diagram consists of use case and actors.
- The main purpose is to show the interaction between the use cases and the actor.
- It intends to represent the system requirements from user's perspective.
- The use cases are the functions that are to be performed in the module.



*Fig:5.1 Use Case Diagram for Smart Voting System*

## CLASS DIAGRAM

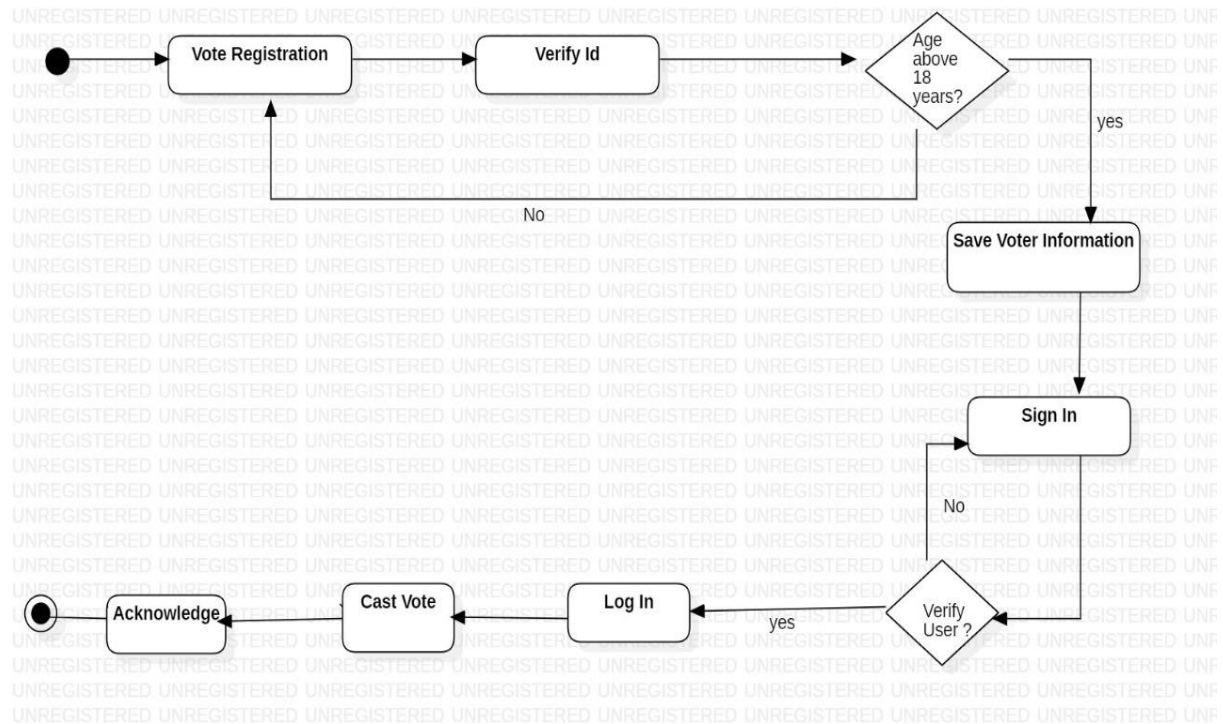
- It contains the classes involved and shows the connections between the various classes.
- Class diagram includes classes, which further has a class label or name, attributes of the class and the operations of functions performed by the class.



**Fig:5.2** Class Diagram for Smart Voting System

## ACTIVITY DIAGRAM

- It shows the flow of the various activities that are undergone from the beginning till the end.
- It consists of the activities that are held and carried out throughout the session from starting till the ending stage.

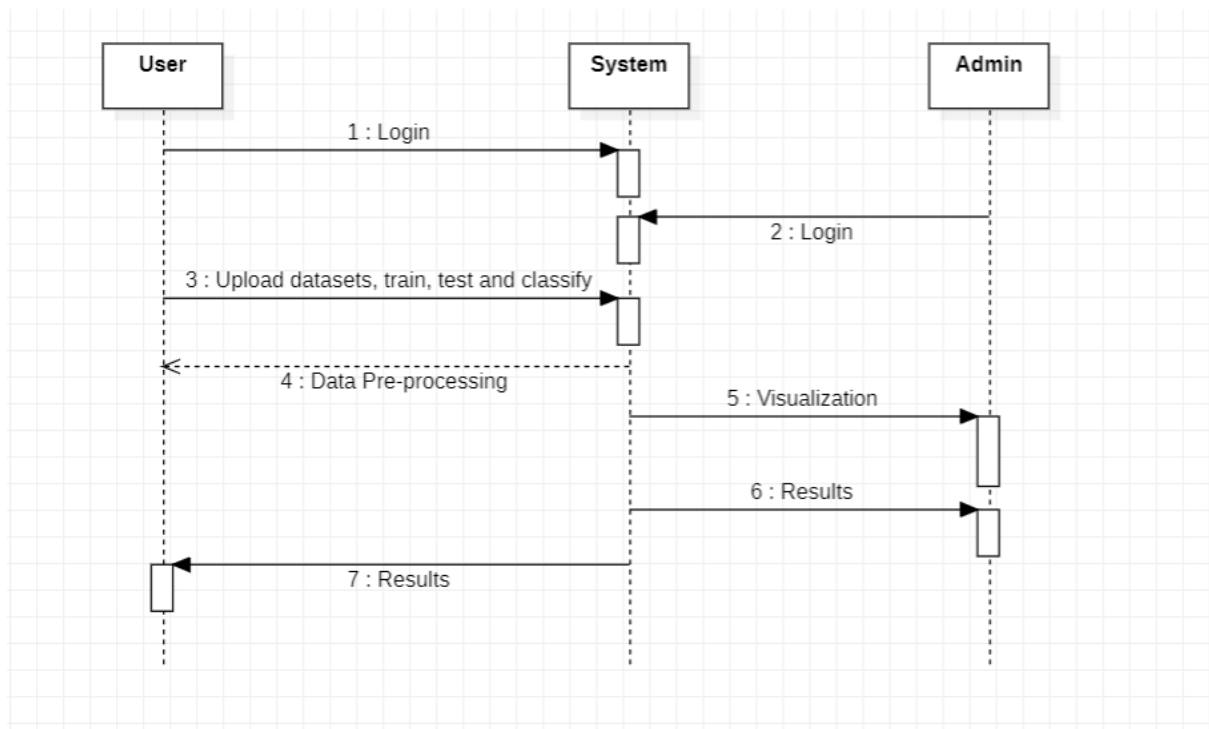


*Fig:5.3 Activity Diagram for Smart Voting System*



## SEQUENCE DIAGRAM

- It shows the sequence of the steps that are carried out throughout the process of execution.
- It involves lifelines or life time of a process that shows the duration for which the process is alive while the steps are taking place in the sequential manner.
- Sequence diagram specifies the order in which the various steps are executed.



**Fig:5.4** Sequence Diagram for Smart Voting System

# CHAPTER–6

## 6. RESULTS AND PERFORMANCE EVALUATION

### Views.py:

```
from django.shortcuts import render, get_object_or_404, redirect
from .models import User, UserProfile, Position, Candidate
from django.http import HttpResponseRedirect
from django.contrib.auth import logout, login, authenticate
from django.contrib.auth.decorators import login_required
import matplotlib
matplotlib.use('Agg')
from matplotlib import pyplot as plt
from io import BytesIO
import base64
import cv2
import numpy as np
import logging
from sklearn.model_selection import train_test_split
from . import dataset_fetch as df
from . import cascade as casc
from PIL import Image
from django.conf import settings
from time import time
from sklearn.decomposition import PCA
from sklearn.model_selection import GridSearchCV
from sklearn.svm import SVC
from sklearn.metrics import classification_report
from sklearn.metrics import confusion_matrix
import pickle
```

```

def index(request):
    return render(request, 'Vote/home.html')

def user_login(request):
    if request.method == 'POST':
        username = request.POST['username']
        password = request.POST['password']

        user = authenticate(username=username, password=password)
        if user is not None:
            if user.is_active:
                login(request, user)
                return HttpResponseRedirect('/Vote/detect/')
            else:
                return HttpResponseRedirect("Account disabled")
        else:
            print("Invalid credentials: {0}, {1}".format(username, password))
            return HttpResponseRedirect('/Vote/invalid/')
    else:
        return render(request, 'Vote/login.html', {})

@login_required(redirect_field_name='/Vote/login/')
def user_logout(request):
    logout(request)
    return HttpResponseRedirect('/Vote/home/')

def home(request):
    context = {}
    return render(request, 'Vote/home.html', context)

```

```

def home_hindi(request):
    context = {}
    return render(request, 'Vote/home_hindi.html', context)

def vote(request):
    context = {}
    try:
        pos = Position.objects.all()
        user = User.objects.get(username=request.user.username)
        profile = UserProfile.objects.get(user=user)
        context['candidates'] = []
        if profile.voted:
            return HttpResponseRedirect('/Vote/voted/')
        else:
            for c in pos:
                can = []
                candidate = Candidate.objects.filter(candidate=c)
                for i in range(0, c.no_of_candidates):
                    can.append([candidate[i], c.position])
                context['candidates'].append(can)
    except:
        return HttpResponseRedirect("/Vote/login/")
    if request.method == 'POST':
        pos = Position.objects.all()
        for c in pos:
            s = 'candidate' + c.position
            selected_candidate = Candidate.objects.get(pk=request.POST[s])
            selected_candidate.votes += 1
            selected_candidate.save()
            profile.voted = True
            profile.save()
        return render(request, 'Vote/casted.html', context)

```

```

else:

    print("No Post")

return render(request, 'Vote/vote.html', context)

def results(request):
    context = { }
    pos = Position.objects.all()
    context['candidates'] = []
    for c in pos:
        fig = plt.figure()
        vot = [] # No of votes for graph
        can = []
        cand = [] # Name of the candidates for graph labelling
        candidate = Candidate.objects.filter(candidate=c)
        for i in range(0, c.no_of_candidates):
            can.append([candidate[i]])
            cand.append(candidate[i].name)
            vot.append(candidate[i].votes)

        plt.pie(vot, labels=cand, autopct='% 1.1f%%', shadow=True, startangle=140)
        plt.axis('equal')
        name =
"C:\\Users\\grohi\\PycharmProjects\\FSVS\\venv\\SmartVoting\\Vote\\static\\Vote\\" +
c.position + '.png'
        fig.savefig(name)
        can[0].append(c)
        can[0].append(name)
        plt.close(fig)
        context['candidates'].append(can)

return render(request, 'Vote/results.html', context)

```

```

def about(request):
    context = {}
    return render(request, 'Vote/about.html', context)


def voted(request):
    context = {}
    return render(request, 'Vote/voted.html', context)


def invalid(request):
    context = {}
    return render(request, 'Vote/invalid.html', context)


def casted(request):
    context = {}
    return render(request, 'Vote/casted.html', context)


def create_dataset(request):
    #print request.POST
    userId = request.POST['userId']
    print (cv2.__version__)
    # Detect face
    #Creating a cascade image classifier
    faceDetect = cv2.CascadeClassifier(r'C:\Users\grohi\PycharmProjects\FSVS\venv\SmartVoting\ml\haar_cascade_frontalface_default.xml')
    #camture images from the webcam and process and detect the face
    # takes video capture id, for webcam most of the time its 0.
    cam = cv2.VideoCapture(0)

    # Our identifier
    # We will put the id here and we will store the id with a face, so that later we can identify
    whose face it is

```

```

id = userId
# Our dataset naming counter
sampleNum = 0
# Capturing the faces one by one and detect the faces and showing it on the window
while(True):
    # Capturing the image
    #cam.read will return the status variable and the captured colored image
    ret, img = cam.read()
    #the returned img is a colored image but for the classifier to work we need a greyscale
image
    #to convert
    gray = cv2.cvtColor(img, cv2.COLOR_BGR2GRAY)
    #To store the faces
    #This will detect all the images in the current frame, and it will return the coordinates
of the faces
    #Takes in image and some other parameter for accurate result
    faces = faceDetect.detectMultiScale(gray, 1.3, 5)
    #In above 'faces' variable there can be multiple faces so we have to get each and every
face and draw a rectangle around it.
    for(x,y,w,h) in faces:
        # Whenever the program captures the face, we will write that is a folder
        # Before capturing the face, we need to tell the script whose face it is
        # For that we will need an identifier, here we call it id
        # So now we captured a face, we need to write it in a file
        sampleNum = sampleNum+1
        # Saving the image dataset, but only the face part, cropping the rest

cv2.imwrite(r'C:\Users\grohi\PycharmProjects\FSVS\venv\SmartVoting\ml\dataset\'+'str(
id)+'.'+str(sampleNum)+'.jpg', gray[y:y+h,x:x+w])
        # @params the initial point of the rectangle will be x,y and
        # @params end point will be x+width and y+height
        # @params along with color of the rectangle
        # @params thickness of the rectangle

```



```

cv2.rectangle(img,(x,y),(x+w,y+h), (0,255,0), 2)

# Before continuing to the next loop, I want to give it a little pause
# waitKey of 100 millisecond
cv2.waitKey(250)

#Showing the image in another window
#Creates a window with window name "Face" and with the image img
cv2.imshow("Face",img)
#Before closing it we need to give a wait command, otherwise the open cv wont work
# @params with the millisecond of delay 1
cv2.waitKey(1)
#To get out of the loop
if(sampleNum>35):
    break
#releasing the cam
cam.release()
# destroying all the windows
cv2.destroyAllWindows()

return redirect('/Vote/face_index')

def trainer(request):
    """
    In trainer.py we have to get all the samples from the dataset folder,
    for the trainer to recognize which id number is for which face.

    for that we need to extract all the relative path
    i.e. dataset/user.1.1.jpg, dataset/user.1.2.jpg, dataset/user.1.3.jpg
    for this python has a library called os

    :param request:
    :return:
    """

```

```

import os

from PIL import Image

#Creating a recognizer to train
recognizer = cv2.face.LBPHFaceRecognizer_create()

#Path of the samples
path = r'C:\Users\grohi\PycharmProjects\FSVS\venv\SmartVoting\ml\dataset'

# To get all the images, we need corresponding id
def getImagesWithID(path):
    # create a list for the path for all the images that is available in the folder
    # from the path(dataset folder) this is listing all the directories and it is fetching the
    # directories from each and every pictures
    # And putting them in 'f' and join method is appending the f(file name) to the path
    # with the '/'
    imagePath = [os.path.join(path,f) for f in os.listdir(path)] #concatenate the path with
    # the image name
    #print imagePath

    # Now, we loop all the images and store that userid and the face with different image
    # list
    faces = []
    Ids = []

    for imagePath in imagePath:
        # First we have to open the image then we have to convert it into numpy array
        faceImg = Image.open(imagePath).convert('L') #convert it to grayscale
        # converting the PIL image to numpy array
        # @params takes image and conversion format
        faceNp = np.array(faceImg, 'uint8')

        # Now we need to get the user id, which we can get from the name of the picture
        # for this we have to split the path() i.e dataset/user.1.7.jpg with path splitter and then
        # get the second part only i.e. user.1.7.jpg
        # Then we split the second part with . splitter

```

```

        # Initially in string format so have to convert into int format
        ID = int(os.path.split(imagePath)[-1].split('.')[1]) # -1 so that it will count from
backwards and split the second index of the '.' Hence id

        # Images
        faces.append(faceNp)

        # Label
        Ids.append(ID)

        #print ID
        cv2.imshow("training", faceNp)
        cv2.waitKey(10)

    return np.array(Ids), np.array(faces)

# Fetching ids and faces
ids, faces = getImagesWithID(path)

#Training the recognizer
# For that we need face samples and corresponding labels
recognizer.train(faces, ids)

# Save the recognizer state so that we can access it later

recognizer.save(r'C:\Users\grohi\PycharmProjects\FSVS\venv\SmartVoting\ml\recognize
r\trainingData.yml')
cv2.destroyAllWindows()

return redirect('/Vote/home')

def detect(request):
    faceDetect =
    cv2.CascadeClassifier(r'C:\Users\grohi\PycharmProjects\FSVS\venv\SmartVoting\ml\haa
rcascade_frontalface_default.xml')

```

```

cam = cv2.VideoCapture(0)
# creating recognizer
rec = cv2.face.LBPHFaceRecognizer_create();
# loading the training data

rec.read(r'C:\Users\grohi\PycharmProjects\FSVS\venv\SmartVoting\ml\recognizer\trainin
gData.yml')
getId = 0
font = cv2.FONT_HERSHEY_SIMPLEX
userId = 0
while(True):
    ret, img = cam.read()
    gray = cv2.cvtColor(img, cv2.COLOR_BGR2GRAY)
    faces = faceDetect.detectMultiScale(gray, 1.3, 5)
    for(x,y,w,h) in faces:
        cv2.rectangle(img,(x,y),(x+w,y+h), (0,255,0), 2)

        getId,conf = rec.predict(gray[y:y+h, x:x+w]) #This will predict the id of the face

        #print conf;
        if conf<35:
            userId = getId
            cv2.putText(img, "Detected",(x,y+h), font, 2, (0,255,0),2)
        else:
            cv2.putText(img, "Unknown",(x,y+h), font, 2, (0,0,255),2)

        # Printing that number below the face
        # @Prms cam image, id, location,font style, color, stroke

cv2.imshow("Face",img)
if(cv2.waitKey(1) == ord('q')):
    break
elif(userId != 0):

```

```

cv2.waitKey(1000)
cam.release()
cv2.destroyAllWindows()
return HttpResponseRedirect('/records/details/'+str(userId))

```

```

cam.release()
cv2.destroyAllWindows()
return redirect('/Vote/home')

```

```

def face_index(request):
    return render(request, 'face_index.html')

```

## **Home.html**

```

<!doctype html>
<html>
<head>
    { % load static % }
<meta charset="utf-8">
<meta name="viewport" content="width=device-width, initial-scale=1;">
<title>Smart Voting System</title>
<link
                                                    rel="stylesheet"
href="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/css/bootstrap.min.css">

<!-- jQuery library -->
<script src="https://ajax.googleapis.com/ajax/libs/jquery/3.3.1/jquery.min.js"></script>

<!-- Latest compiled JavaScript -->
<script
src="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/js/bootstrap.min.js"></script>
<link rel="stylesheet" type="text/css" href="{ % static 'Vote/home.css' % }">
<style>

```

```

footer .glyphicon {
    font-size: 20px;
    margin-bottom: 20px;
    color: #f4511e;
}

.navbar-brand{
    float: none;
</style>
</head>
<body>
    <nav class="navbar navbar-default navbar-fixed-top">
        <div class="container-fluid">
            <div class=" navbar-header">
                
                <a class="navbar-brand" href="#">Smart Voting System</a>
            </div>
            <ul class="nav navbar-nav">
                <li class="active"><a href="#">Home</a></li>
                <li><a href="{% url 'Vote:face_index' %}">Add Facedata</a></li>
                <li><a href="{% url 'Vote:vote' %}">Vote</a></li>
                <li><a href="{% url 'Vote:results' %}">Results</a> </li>
                <li><a href="/admin">Admin</a></li>
                <li><a href="{% url 'Vote:about' %}">About</a></li>

            </ul>

            {% if user.is_authenticated %}
            <ul class="nav navbar-nav navbar-right">
                <li><a href="{% url 'Vote:logout' %}"><span class="glyphicon glyphicon-log-
in"></span> Log Out</a></li>
            </ul>
            {% else %}

```

```

        <ul class="nav navbar-nav navbar-right">
        <li><a href="{ % url 'Vote:login' % }"><span class="glyphicon glyphicon-log-
in"></span> Log In</a></li>
        </ul>
        { % endif % }
    </div>
</nav>
<div class="page-header bg-6">
    <h1 align="center" ><strong>Smart Voting System </strong></h1>
    <h3 align="center">A fast and efficient way to vote sitting at home.</h3>
</div>
<!--row column-->
<div class="row">
    <div class="col-lg-4" class="slideanim"></div>
    <div class="col-lg-8">
        <h4 align="justify">
            <strong>Smart voting System</strong> uses Facial Recognition along side traditionl
passwords which provides an additional security.

            This Web App is designed to serve the needs of organisation and institutions.
        </br>
        </br>
        <strong>Smart Voting System </strong>doesn't limit the number of polls you can
create. Unlike the vast majority of voting websites you can continue using the free service
as often as you'd like.

            It's a quick and easy way to organise elections for an organisation or an institution
and let your colleagues or CR's cast their vote.

        </br></br> <strong>Contact the Admin </strong> to get your Login Credentials.

            Once you get the Login Credentials you can add your Face Data and you are ready to
vote.
        </h4>
    </div>

```

```

</div>

<div class="row bg-5" >
  <div class="col-lg-8" class="slideanim">
    <h2> <strong>What makes this web app so special?</strong></h2>
    <h4 align="justify">
      <strong> Smart Voting System</strong> has a very minimalistic, user-friendly
design.

    </br>There's no hassle, no fuss, and no lengthy steps asking you to fill out
information. In fact, you don't even have to sign up for an account just get your login
credentials from the admin of your organisation. This voting website doesn't need your
mobile number, doesn't save your email address for sending newsletters and would never
sell your contact information to third party advertisers!!

    </h4>
  </div>

  <div class="col-lg-4"></div>

</div>

<div class="row">
  <div class="col-lg-12" align="center">
    <h2>Voting Just a <strong>Click</strong> Away</h2>
    <br>
    <a href="{ % url 'Vote:vote' % }"><button type="button"
class="btn1">Vote</button><br><br></a>
  </div>
</div>

<script>
$(document).ready(function(){
  $('[data-toggle="popover"]').popover();
});
</script>

```



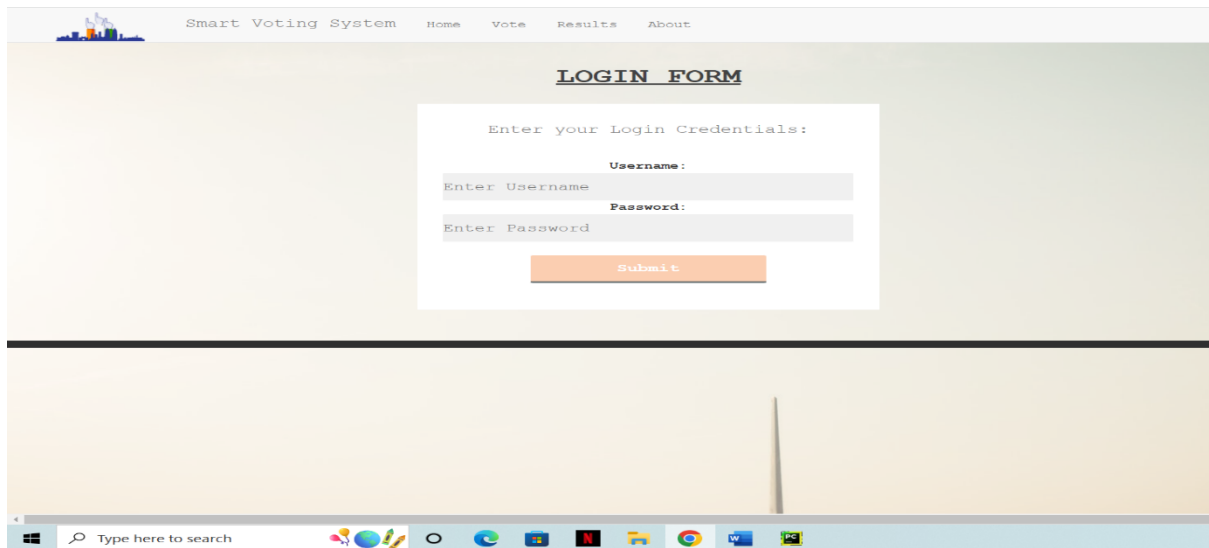
```

        <div class="container-fluid bg-4">
<h2 class="text-center">Get In Touch</h2>
</br>
<div class="row row2">
    <div class="col-sm-5">
        <p><strong>Contact Us</strong> and we'll get back to you within 24 hours.</p>
        <p><span class="glyphicon glyphicon-map-marker"></span>    Vignana Bharathi
Institute Of Technology, Hyderabad </p>
        <p><span class="glyphicon glyphicon-phone"></span>    +91-8125668249    ->
G.Pranay</p>
        <p><span class="glyphicon glyphicon-envelope"></span>
gurrampranay143@gmail.com -> G.Pranay</p>
        <p><span class="glyphicon glyphicon-phone"></span>    +91-8125615728    ->
G.Bhargava</p>
        <p><span class="glyphicon glyphicon-envelope"></span>
bhargava.naidu01@gmail.com -> G.Bhargava </p>
        <p><span class="glyphicon glyphicon-phone"></span>    +91-9121004186    ->
G.Rohit</p>
        <p><span class="glyphicon glyphicon-envelope"></span> grohit0303@gmail.com -
> G.Rohit</p>

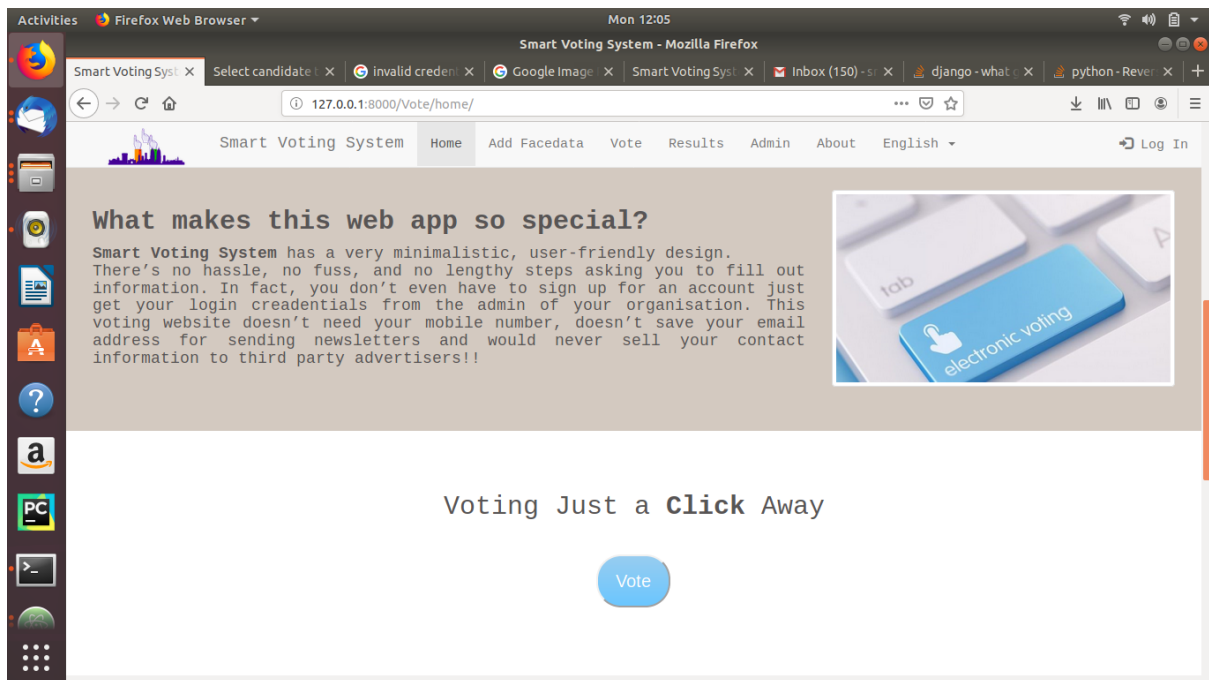
    </div>
<div class="col-sm-7">
    <div class="row row2">
        <div class="col-sm-6 form-group">
            <input class="form-control" id="name" name="name" placeholder="Name"
type="text" required>
        </div>
        <div class="col-sm-6 form-group">
            <input class="form-control" id="email" name="email" placeholder="Email"
type="email" required>
        </div>
    </div>

```

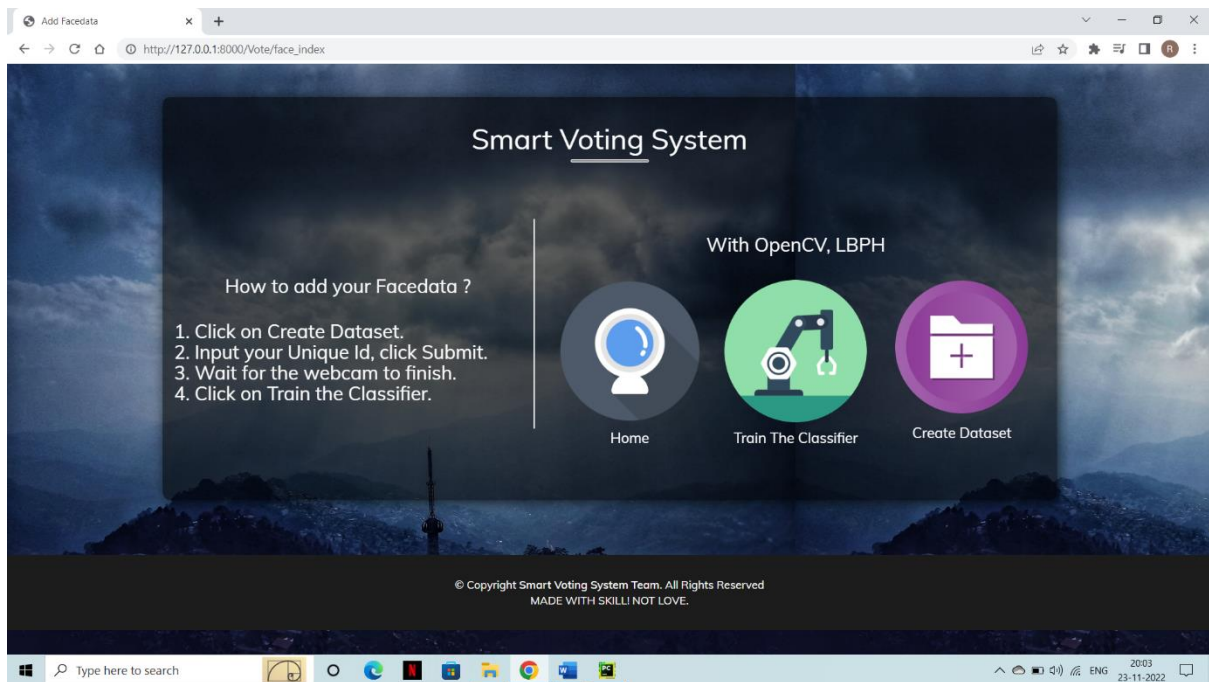




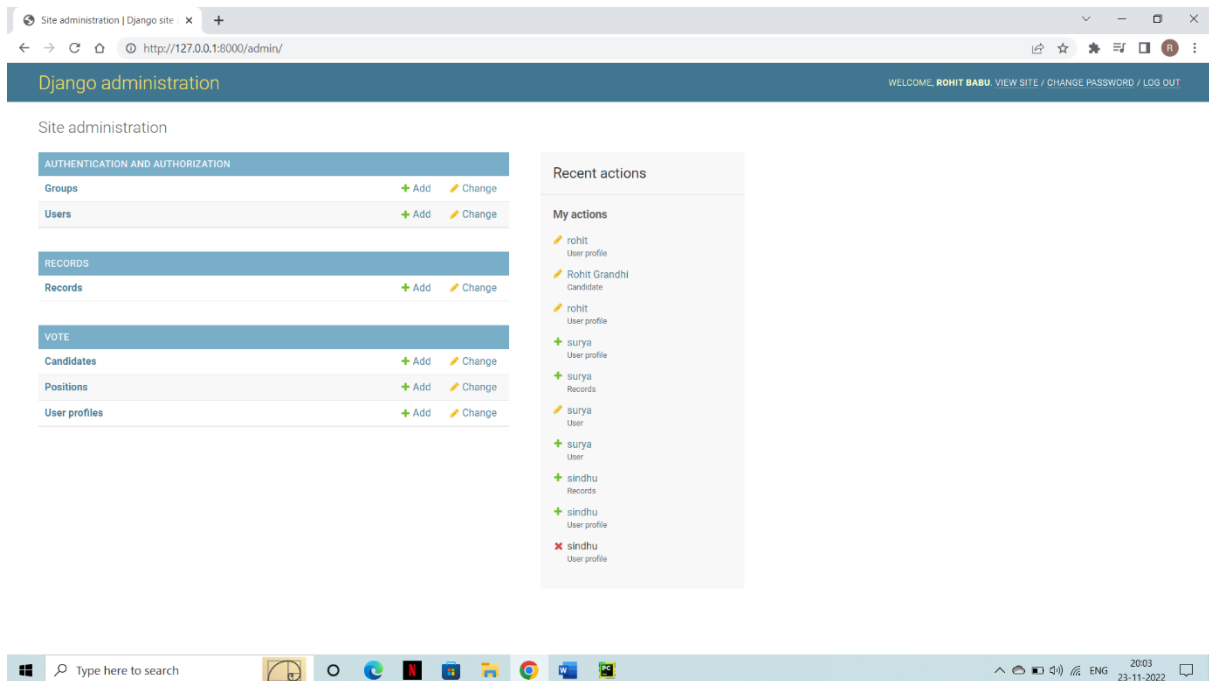
*Fig:6.2 Login page*



*Fig:6.3 Voting page*



**Fig:6.3** Dataset page



**Fig:6.4** Administration Page

# CHAPTER–7

## **Conclusion and Future Scope**

This online voting system will maintain voter information, allowing voters to login and exercise their voting rights. The system will include all voting system capabilities. It offers facilities for preserving voters' votes for each party and counts the overall number of votes cast for each party. The admin maintains a DATABASE in which all the names of voters with full information are saved. This user, who is eligible to registers his/her information in the database, and when he/she wants to vote, he/she must login with his/her id and password, and may vote for any party only once. Voting details are saved in a database, and the results are shown using a computation. The proportion of votes cast rises as a result of the online voting mechanism. It reduces the cost and duration of the voting process. It is incredibly simple to use and takes much less time. It is really simple to debug.

### **Future Scope:**

The “VOTING SOFTWARE” provides better information for the users of this system easily they can vote from anywhere without facing any difficulty. The proposed system has good authentication so only authorized person can able to vote and also cannot vote multiple types. Vote Counting can be made very quickly and results will be displayed in few minutes. An easy conduction of elections with very less man power comparatively to the existing system can developed. People can cast their vote from any where at any time.

# CHAPTER–8

## References

1. Fugatashlesha, Gaikward Shital, Gangurde jyoti, swat aishwarya, "Secured Online Voting System Using Face Recognition", International Journal of Advanced Research In Science Management And Technology, Vol.2, Issue.6, Jun 2016.
2. Patil rahul.H, Tarte babita.B, Wadekar sapana.S, Zurunge Bakti.S, Prof .Phursule Rajesh, "A Secure E-Voting System Using Face Recognition And Dactylogram", International Engineering Research Journal (IERJ), Vol. 2, Issue .2, pp. 758-762, 2016, ISSN 2395-1621.
3. Ratnaprabha.S, TriniXavier.X, Deepika.V, Iswarya.C,"A Survey On E-Voting System Using Audino Software" International Journal Of Advanced Research In Electrical, Electronics And Instrumental Engineering, Vol.5, Issue.2, Feb 2016.
4. Priya taneja, Rajender kumar trivaedi, Rajani Sharma,"A Proposed Method Of Online Voting By Face Detection", International Journal Of Advanced Technology in Engineering and Science, Vol.3, Issue. 1, March 2015.
5. Neha Gandhi "Study on security of Online Voting System Using Biometrics and Steganography" IJCSC, Vol.5, Mar-Sep 2014, pp.29-32.
6. Niranjanmalwade, Mahesh Taware, Akshay kamble & Aniruddha Kamble ,"Smart Voting System with Face Recognition", Vol.2, Issue.2, feb 2014.
7. Ankita Ladda, Rasika Gunjal, Ratnavidya Kanawade & Sanjay Agarwal 17 March 2013 "Automated Secure Attendance System" ICCSE.
8. Arati Kothari and Sampatkumari Maruti Bandagar . 2011. "Performance & Evaluation of Face Recognition Algorithm" World Journal of Science & Technology.
9. Ahonen T, Hadid A, Pietikainen M, "Face recognition with local binary patterns". In Proceeding of European conference on computer vision (ECCV2004), LNCS 3021, pp 469-481, 2004
10. Heseltine, T., Pears, N., Austin, J.: Evaluation of image pre-processing techniques for eigenface based face recognition. In Proc. of the Second International Conference on Image and Graphics, SPIE vol. 4875, (2002) 677-685.