CHAPTER 1

ABSTRACT

Artificial Intelligence is a branch of computer science dealing with simulation of intelligent behaviour in computers.**Machine learning** (**ML**) is the scientific study of algorithms and statistical models that computer systems use to perform a specific task without using explicit instructions, relying on patterns and inference instead. It is seen as a subset of artificial intelligence. Machine learning algorithms build a mathematical model based on sample data, known as "training data", in order to make predictions or decisions without being explicitly programmed to perform the task.

A **facial recognition system** is a technology capable of identifying or verifying a person from a digital image or a video frame from a video source. There are multiple ways of doing it. We are using Adaboost learning algorithm, Haar Cascades use the Adaboost learning algorithm which selects a small number of important features from a large set to give an efficient result of classifiers.

A Database is an organized collection of data, generally stored and accessed electronically from a computer system. Here we use Google firebase to store the data and retrieve it whenever required.

A QR code is a type of barcode which would store the data for the objects, which would retrieved when a camera reads it sends the image to the qr code reader.

REQIREMENT SPECIFICATION

HARDWARE REQUIREMENTS:

•	System	: 2.2 GHz or faster Processor
•	Hard Disk	: 10GB
•	RAM	: 4GB
•	Input Devices	: Keyboard, Mouse

SOFTWARE REQUIREMENTS:

•	Operating System	: Windows 10 or ubuntu14 above or raspberry pi
•	Software	: openCV functions, Google Firebase, QR reader, Bootstrap

CHAPTER 2:

INRODUCTION

1. OBJECTIVES:

There 6 are objectives to achieve in this project:

- To give image of a face as input and recognize it
- To read the data of the items using QR code
- To store and retrieve both those data in Google Firebase
- Once shopping is done, it should retrieve the data of items and generate the bill
- It should able to transfer the money from persons account seamless
- If the person is not in database it should allow him/her to pay in cash

2. METHODOLOGY

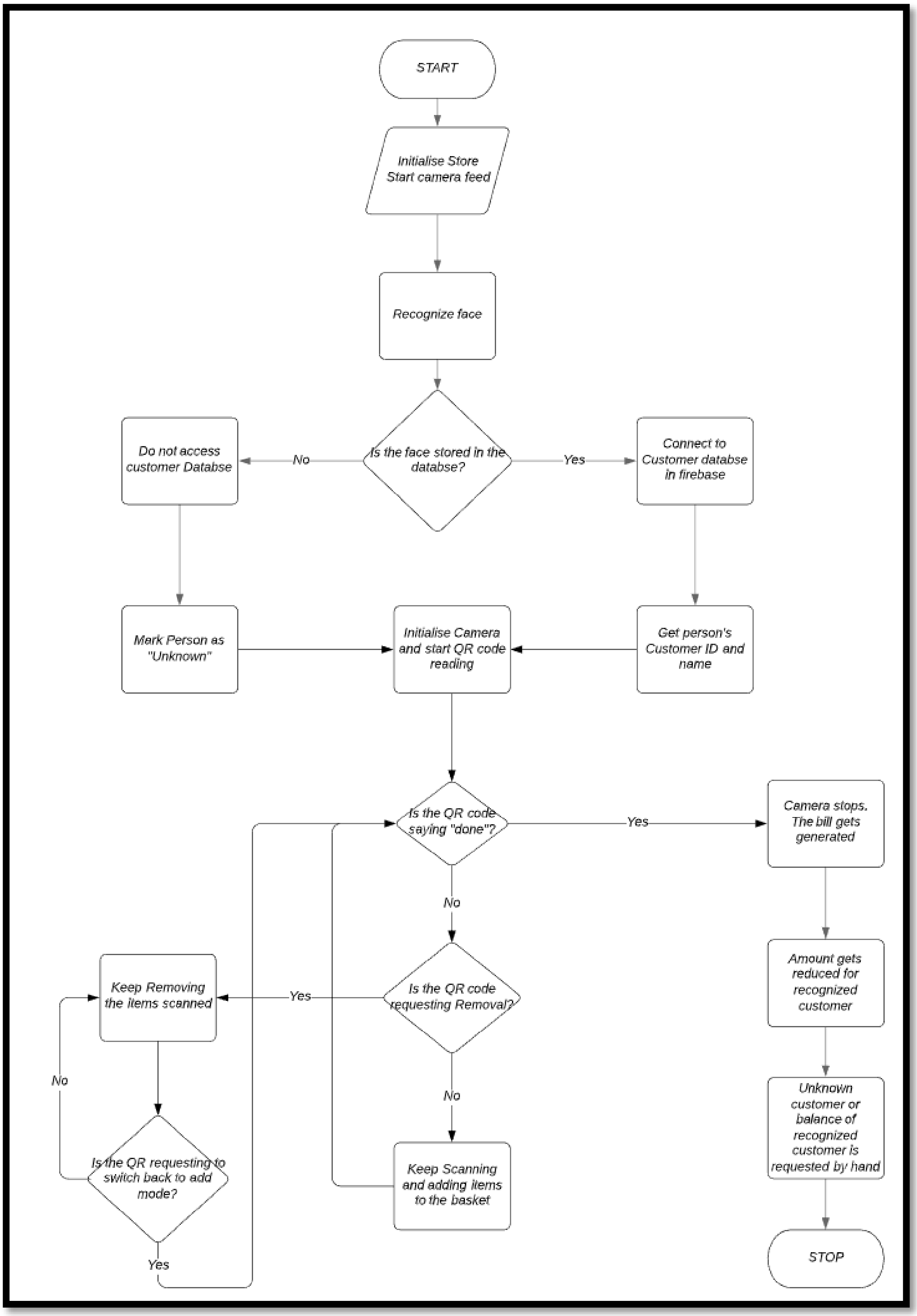
EXISTING METHODOLOGY

Automated Retail Store using AI Face Recognition is a novel method to solve the problem of efficiency. There are few stores that exists across the world and none in India that uses face recognition in the retail stores. There are quite a few researches that exists but most of them are not implemented. The ones which are implemented have more motives than to better the system of payments or to improve the productivity, they use it for loyalty programs and to do a personal marketing which would target the customer and show the customer what he/she likes. This can become counter-productive if the customer feels this is risking their privacy. Some retail stores use the data such as how frequent is the customer, on what products does the customer spends their time. Stores also looks into peoples' expression and try to gauge the user experience but finding the human expression has not been that successful and more researches are going on this field. In India they are using little to no facial recognition technology. They are using barcode to recognize the product using a IR device. Instead of face, they are using customer’s phone number to keep track of their purchases and send their promotional content to it. The payment system is regular old, card or cash system. Only some stores support QR code scan UPI system .In the perspective to a real life viral outbreak we would like to mention that having a contactless store is one of the best preventive and hygienic measures that can be taken.

PROPOSED METHODOLOGY

The proposed system is a faster, secure payment system which is very reliable and it works in all possible circumstance that is possible. Our system is very user friendly and our system respects data privacy of the user. We do not mine data to get a pattern and we do not sell the user data to third party, we care for their data. Our system will revolutionize the retail sector in India. There is no other company which uses this technology. We decided to use Facial Recognition using the Haar Cascade classifier which uses the Adaboost algorithm which is very accurate and fast to recognize a face and identify the person. We believe that the average time for billing will be shortened by at least 50% or more. Also we have switched barcodes to QR codes which can not only store more information but also scan really fast and can be read from any orientation (omnidirectional).We are reading the QR code using PyZbar (A python QR code Reading Library of functions) which reads it in less than a second and each individual item is given a unique QR code so you cannot bill the same item twice by mistake.

1. STEPS INVOLVED


START
Initialise Store
Start camera feed
Recognize face
Is the face stored in the databse?
No
Yes
Do not access customer Databse
Connect to Customer databse in firebase
Merk Person as "Unknown"
Initialise Camera and start QR code reading
Get person's Customer ID and name
Is the QR code saying "done"?
Yes
Camera stops. The bill gets generated
No
Amount gets reduced for recognized customer
Keep Removing the items scanned
Yes
Is the QR code requesting Removal?
No
No
Unknown customer or balance of recognized customer is requested by hand
Is the QR requesting to switch back to add mode?
Keep Scanning and adding items to the basket
Yes
STOP

1. Inputting the user face:

The store takes the user’s facial pictures when the user registers. The store then associates the user id to a customer-id. The images are stored in a folder called “/database” to later run the opencv train function on. The face is recognized using the Haar Cascade Algorithm.

Haar Cascade:

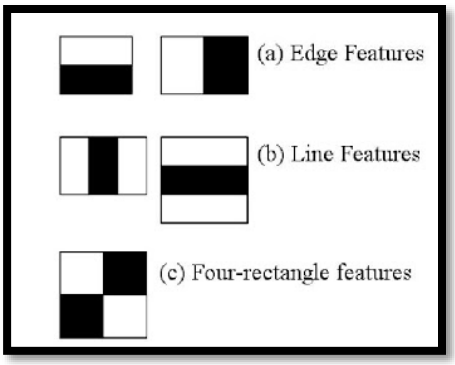
Haar Cascade is a machine learning object detection algorithm used to identify objects in an image or video and based on the concept of features proposed by Paul Viola and Michael Jones in their paper "Rapid Object Detection using a Boosted Cascade of Simple Features" in 2001.
It is a machine learning based approach where a cascade function is trained from a lot of positive and negative images. It is then used to detect objects in other images.

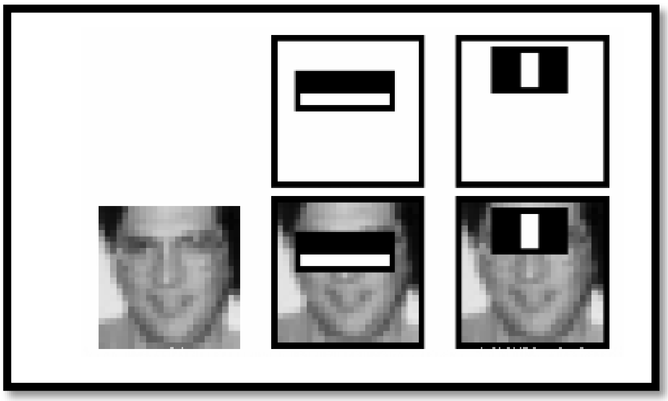
The algorithm has four stages:

- Haar Feature Selection
- Creating Integral Images
- Adaboost Training
- Cascading Classifiers

It is well known for being able to detect faces and body parts in an image, but can be trained to identify almost any object.

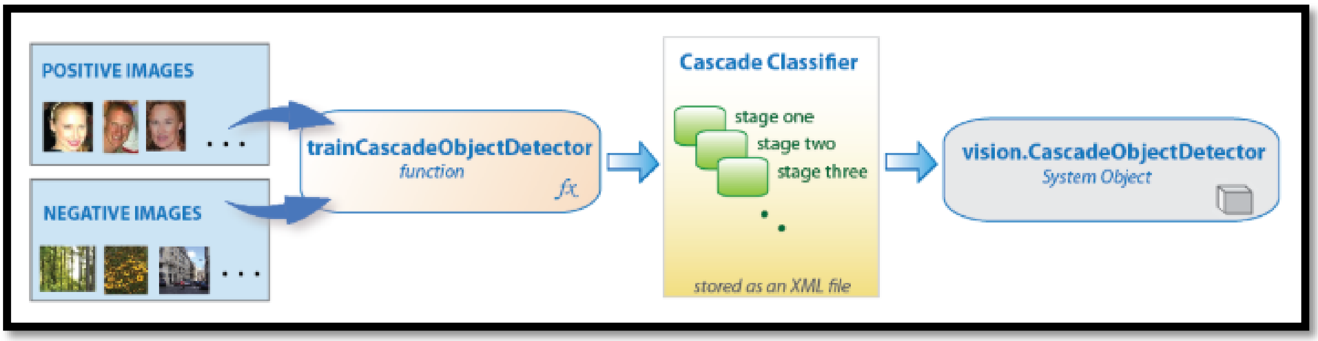
In this case we need to use it for Facial Recognition. Initially, the algorithm needs a lot of positive images of faces and negative images without faces to train the classifier. Then we need to extract features from it.





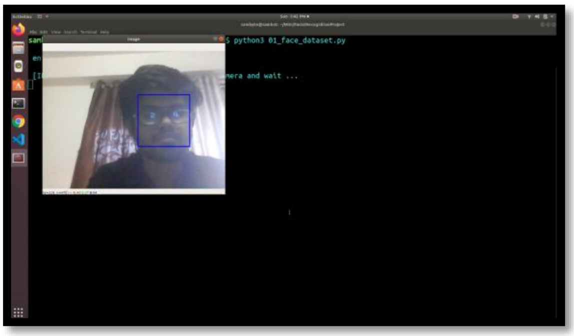
The Haar cascade emphasises on a perfect balance of both speed and accuracy. Sure it is not the most accurate nor the most efficient one for facial recognition, but it sure does give a pretty accurate result at a reasonable time.

The cascade Classifier uses the Adaboost algorithm and it works a something like this:

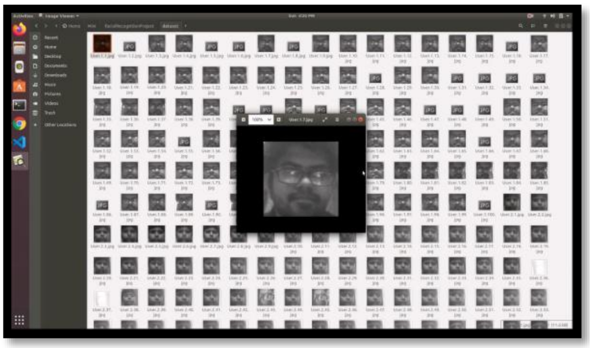


The cascade classifier consists of a collection of stages, where each stage is an ensemble of weak learners. The weak learners are simple classifiers called decision stumps. Each stage is trained using a technique called boosting. Boosting provides the ability to train a highly accurate classifier by taking a weighted average of the decisions made by the weak learners.

Snapshot from code:



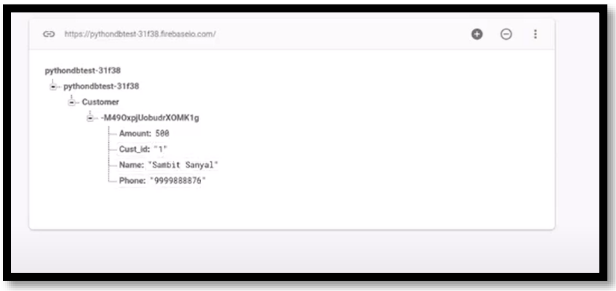
```
sam                                $ python3 01_face_dataset.py
 en
 [I                                mera and wait ...
```



1. Creating an account linked to the Face-Id:

As soon the face is registered the firebase console creates a customer registered with the customer-ID. The process is instant.

Snapshot from code:



```
https://pythondbtest-31f38.firebaseio.com/

pythondbtest-31f38
  pythondbtest-31f38
    Customer
      -M490xpjUobudrXOMK1g
        Amount: 500
        Cust_id: "1"
        Name: "Sambit Sanyal"
        Phone: "9999888876"
```

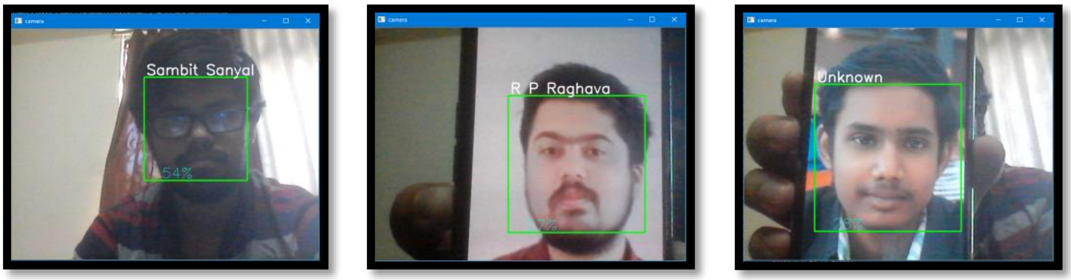
2. Recognizing the face:

The store initialises for every new costumer and first tries to recognize the face and get the face-id. The face-id is associated with the customer-id.

The facial recognition process is broken into 2 parts.

The first part is finding the face using Haar Cascade Classifier. But the second part is comparing the face with the existing dataset (registered Customers). That’s not all. If the system finds out who the face resembles most, it takes gets the customer-id from the firebase in no time. However if it recognizes that the face does not belong in the dataset, it then classifies the person as unknown.

Snapshot from code:



3. Recognize the QR codes:

After the facial recognition the store moves on to scanning and registration of items. The items are all QR coded to ease the use of access because unlike barcodes, QR codes read the same in all orientations. (Except of course lateral inversions).

We then run the QR code scanning snippet to read and register QR codes accordingly. Reading QR codes using the PyZBar module in python, we get results instantly. PyZBar cab also read barcodes. But we preferred QR because we wanted the entire system to get an upgrade.
PyZBar also uses OpenCV to read QR codes.

QR codes and PyZBar:

A QR code object returned by ZBar has three fields

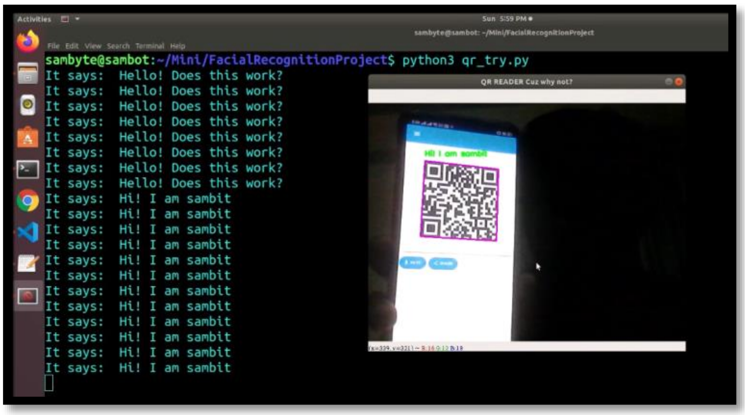
Type: If the symbol detected by ZBar is a QR code, the type is QR-Code. If it is barcode, the type is one of the several kinds of barcodes ZBar is able to read. In our example, we have used a barcode of type CODE-128

Data: This is the data embedded inside the QR code. This data is usually alphanumeric, but other types (numeric, byte/binary etc.) are also valid.

Location: This is a collection of points that locate the code. For QR codes, it is a list of four points corresponding to the four corners of the QR code quad. For barcodes, location is a collection of points that mark the start and end of word boundaries in the barcode. The location points are plotted for a few different kinds of symbols below.

ZBar location points plotted using red dots. For QR codes, it is a vector of 4 corners of the symbol. For barcodes, it is a collection of points that form lines along word boundaries.

Snapshot from code:


Activities
Sun 5:59 PM
sambyte@sambot: ~/Mini/FacialRecognitionProject
File Edit View Search Terminal Help
QR READER Cuz why not?
Hi! I am sambit

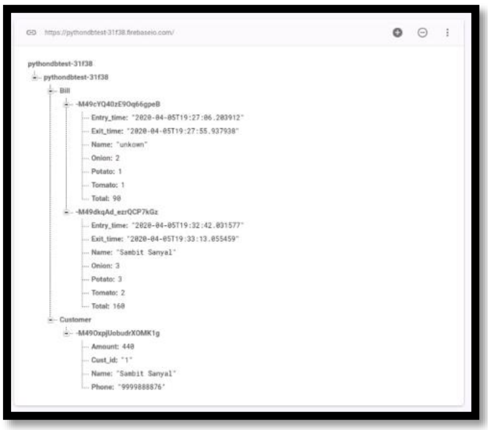
```
sambyte@sambot:~/Mini/FacialRecognitionProject$ python3 qr_try.py
It says:  Hello! Does this work?
It says:  Hello! Does this work?
It says:  Hello! Does this work?
It says:  Hello! Does this work?
It says:  Hello! Does this work?
It says:  Hello! Does this work?
It says:  Hello! Does this work?
It says:  Hello! Does this work?
It says:  Hi! I am sambit
It says:  Hi! I am sambit
It says:  Hi! I am sambit
It says:  Hi! I am sambit
It says:  Hi! I am sambit
It says:  Hi! I am sambit
It says:  Hi! I am sambit
It says:  Hi! I am sambit
It says:  Hi! I am sambit
It says:  Hi! I am sambit
It says:  Hi! I am sambit
It says:  Hi! I am sambit
```

4. Update to the Database:

The final step is to generate a bill and then reduce the total from the firebase database. This is fairly simple as we already have the details with us. All we need to do is upload the bill details to the database and subtract the amount from the customer (For a recognized existing customer).

We also need to upload the bill for customers who aren’t registered. We use the nametag unknown and generate the bill. That concludes the steps of how the program runs.

Snapshot from code:



```
https://pythondbtest-31f38.firebaseio.com/

pythondbtest-31f38
 - pythondbtest-31f38
    - Bill
       - -M49cYQ40zE9Oq66gpeB
          Entry_time: "2020-04-05T19:27:06.203912"
          Exit_time: "2020-04-05T19:27:55.937938"
          Name: "unkown"
          Onion: 2
          Potato: 1
          Tomato: 1
          Total: 90
       - -M49dkqAd_ezrQCP7kGz
          Entry_time: "2020-04-05T19:32:42.031577"
          Exit_time: "2020-04-05T19:33:13.055459"
          Name: "Sambit Sanyal"
          Onion: 3
          Potato: 3
          Tomato: 2
          Total: 160
    - Customer
       - -M49OxpjUobudrXOMK1g
          Amount: 440
          Cust_id: "1"
          Name: "Sambit Sanyal"
          Phone: "9999888876"
```

CHAPTER 3:

LITERATURE SURVEY

Amazon Go: The first thought that comes to any person living in the year of 21st century is the similarities of this store and Amazon Go. For the sake of formalities, let me explain the similarities and dissimilarities between our store and Amazon Go. First being the approach. Amazon Go is not based only on facial recognition. They have upgraded their system to entire body recognition. Pros of this system is that the buyer need not look at the face of camera always while tossing stuff into his/her bag. But the most reported problem is that buyers of similar body types are often mistaken for one another. Sure, both our store and Amazon Go scan individual items but Amazon Go does not use QR codes. They have their own system of recognition. Amazon Go asks for a QR code to be scanned using their phone as you enter the store but we instead ask for a Facial Recognition to be done. Amazon Go has over hundreds of cameras (lot of investment) all over the store to see your every move and recognize when you have picked up an object. Meaning if you pick up something to give someone else, you will be the one paying for it. Amazon go believes in walk away and pay later whereas our store does the payment real-time. Amazon Go also tackled the problem of returning items by using weight sensors. This way they can tell if someone has returned the item. Whereas we use a system of converting into remove mode so that the customer can remove the items he/she doesn’t want using the same QR codes. Amazon Go does not provide the alternate method in case of an unregistered user. Instead everyone is supposed to have the Amazon Go app and use Amazon Pay.

Facial Recognition in Retail stores: Although we thought Amazon Go as phenomenal, it wasn’t the first someone thought of using Machine Learning in a retail store. In fact so many retail stores have used it for their data mining needs. Using Facial Recognition, they have traced users to see which person prefers what and what they look for in the store. This is well and good considering some retailers do sorts of Customer Loyalty programs to appreciate their buyers. But some retails misuse this power to steal data and give it to data collecting companies. But however it hasn’t been implemented for payment yet.

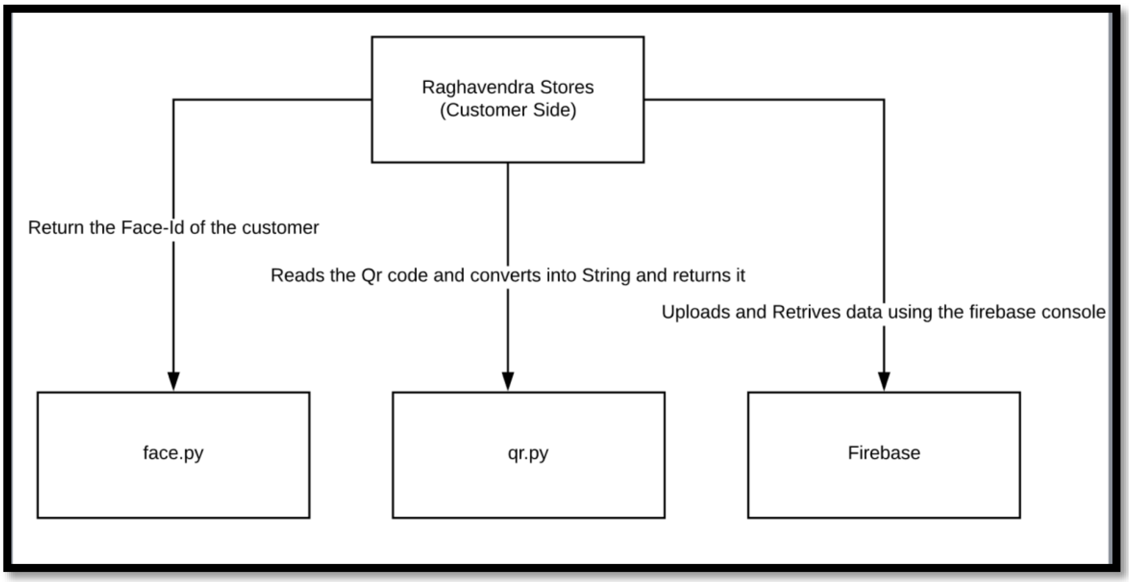
Sources:

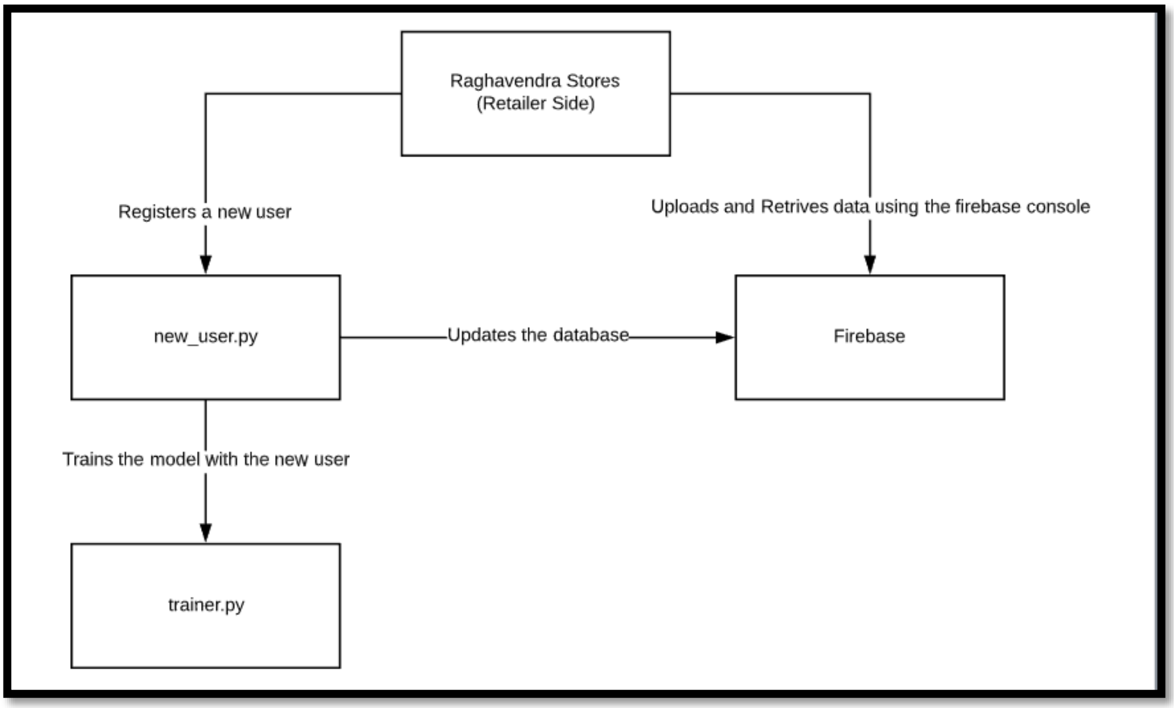
- https://www.justaskgemalto.com/en/amazon-go-checkout-free-store-work/
- https://www.abacademies.org/articles/a-review-on-facial-recognition-algorithms-&-their-application-in-retail-stores-1528-2678-22-3-166.pdf

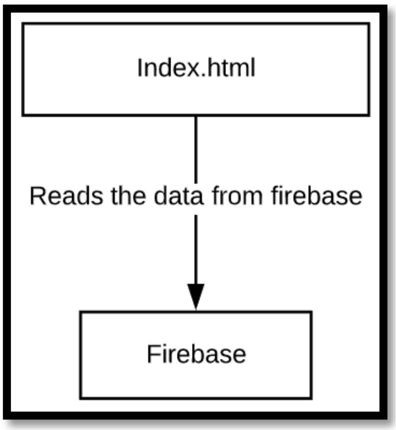
CHAPTER 4:

DESIGN

Site-Map:


Raghavendra Stores
(Customer Side)
Return the Face-Id of the customer
Reads the Qr code and converts into String and returns it
Uploads and Retrives data using the firebase console
face.py
qr.py
Firebase





We have used a system of maintaining multiple files and importing them in order to ensure a better file-function correspondence.

Here’s the list of files and what they do:

1. ***Final.py:***

This program file is for the customer side. It imports the Face.py and Qr.py. It is the only code that has to be called for the execution of the store. It calls the face.py to get the user face-id and then the Qr.py for the QR code decoding. It generates the bill and also updates it into the firebase. All modifications of items, like adding and deletion is handled purely by the Final.py file.

2. ***Face.py:***

This program file is not going to work explicitly. It only gets called by Final.py for the face-id. It imports OpenCV to use Haar Cascade and trainer.yml to detect and recognize the face. It also accesses the database to get a hold of the face-id of the person and send it over to final.py

3. ***Qr.py:***

This program file is not going to work explicitly. It only works on the implicit call of Final.py. This program uses the camera to read and convert the QR codes and send it back to final.py. It imports the OpenCV and PyZBar. It works in a loop until final.py relieves it of duty.

4. ***Admin.py:***

The one file to run to access all the admin functions on the go. Want to add new user? Admin.py! Delete? Admin.py! Check Balance? Perhaps add money too? ADMIN.py! For registering new users it calls new_user.py but the rest all functions are done by it itself. It has firebase access so it can get, alter and post new data on its own.

5. ***New_user.py:***

This program file has the importance of both adding a new user and also re-training the model again for every alteration. It can’t be run explicitly. It is called by admin.py implicitly.

6. ***Trainer.py:***

The trainer of the model. This file is surely not something in the front lines but it is one of the most important files as it retrains the model every time a new user is added. It used the OpenCV to train the model to find both trained faces and stranger faces in general.

7. ***Index.html:***

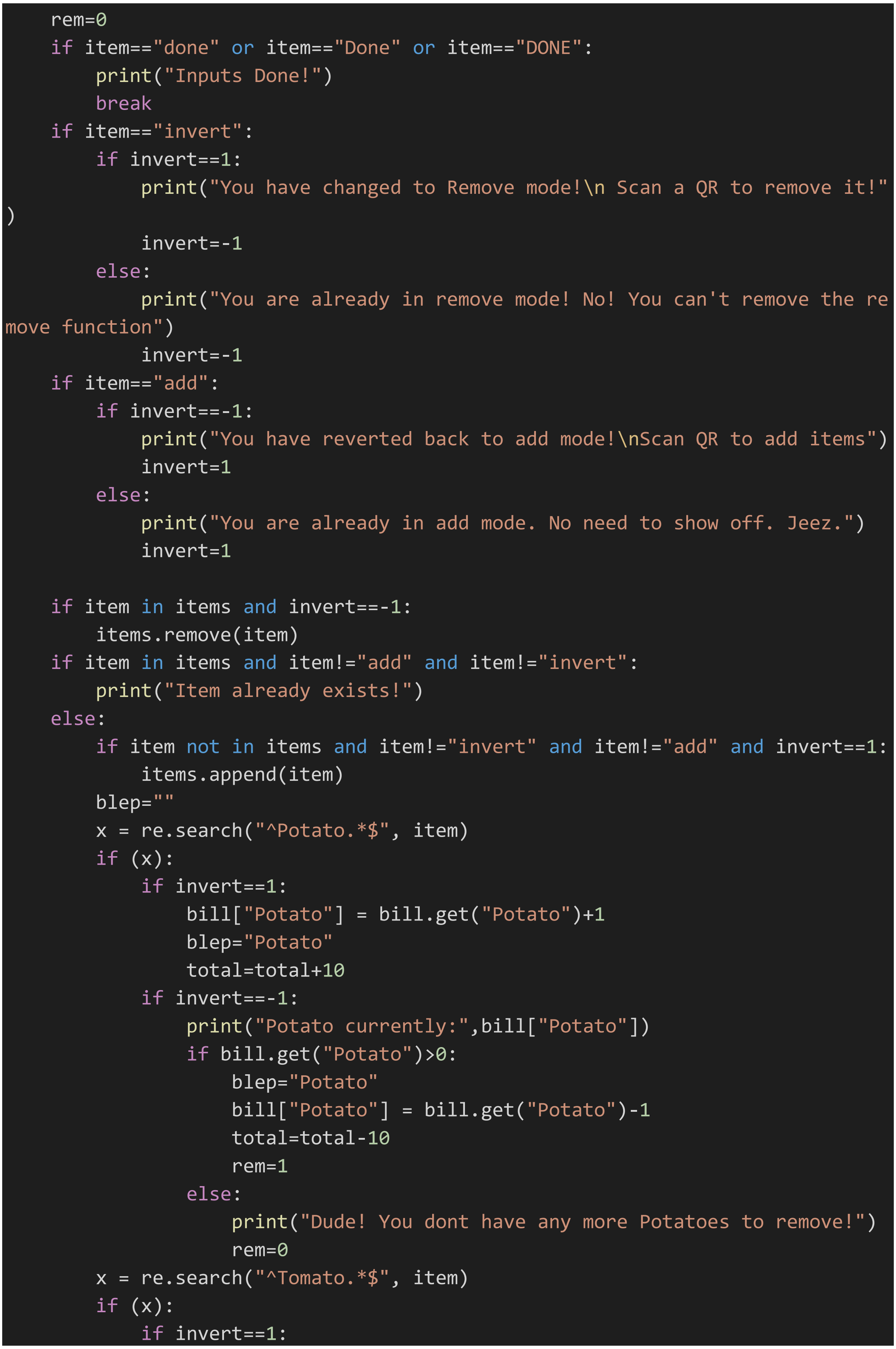
What good is an application without portability? That’s why we have made a webpage that is hosted on firebase and it can access the Firebase Real-time database and show it. This is a Retailer side website that shall be showing the bills and the status of customers.

CHAPTER 5:

CODING

Final.py:

```
import face
import qr
import re
import datetime
from firebase import firebase
uk=0
print("\n---------------------------------------------------------------------
\n     Welcome to Raghavendra Stores\n----------------------------------------
-----------------------------\n")
print("")
print("Currently we have these for sale!")
print("Potatoes at Rs 10")
print("Tomatoes at Rs 20")
print("Onions at Rs 30")
print("")
print("Please show us your beautiful face:")
name=face.main()
if name=='unknown' or name=="unknown":
    uk=1
if uk==0:
    print("Hey ! I know you ! You are :", name)
else:
    print("Sorry ! I guess we have never met before stranger!")
entry = datetime.datetime.now()
print("You entered the shop at ",entry)
item=""
total=0
items=[]
bill = {
    "Name":name,
    "Entry_time": entry,
  "Potato": 0,
  "Tomato": 0,
  "Onion": 0,
  "Total": 0,
  "Exit_time" :entry,
}
invert=1
while True:
    print("Scan QR:")
    item=qr.main()
```

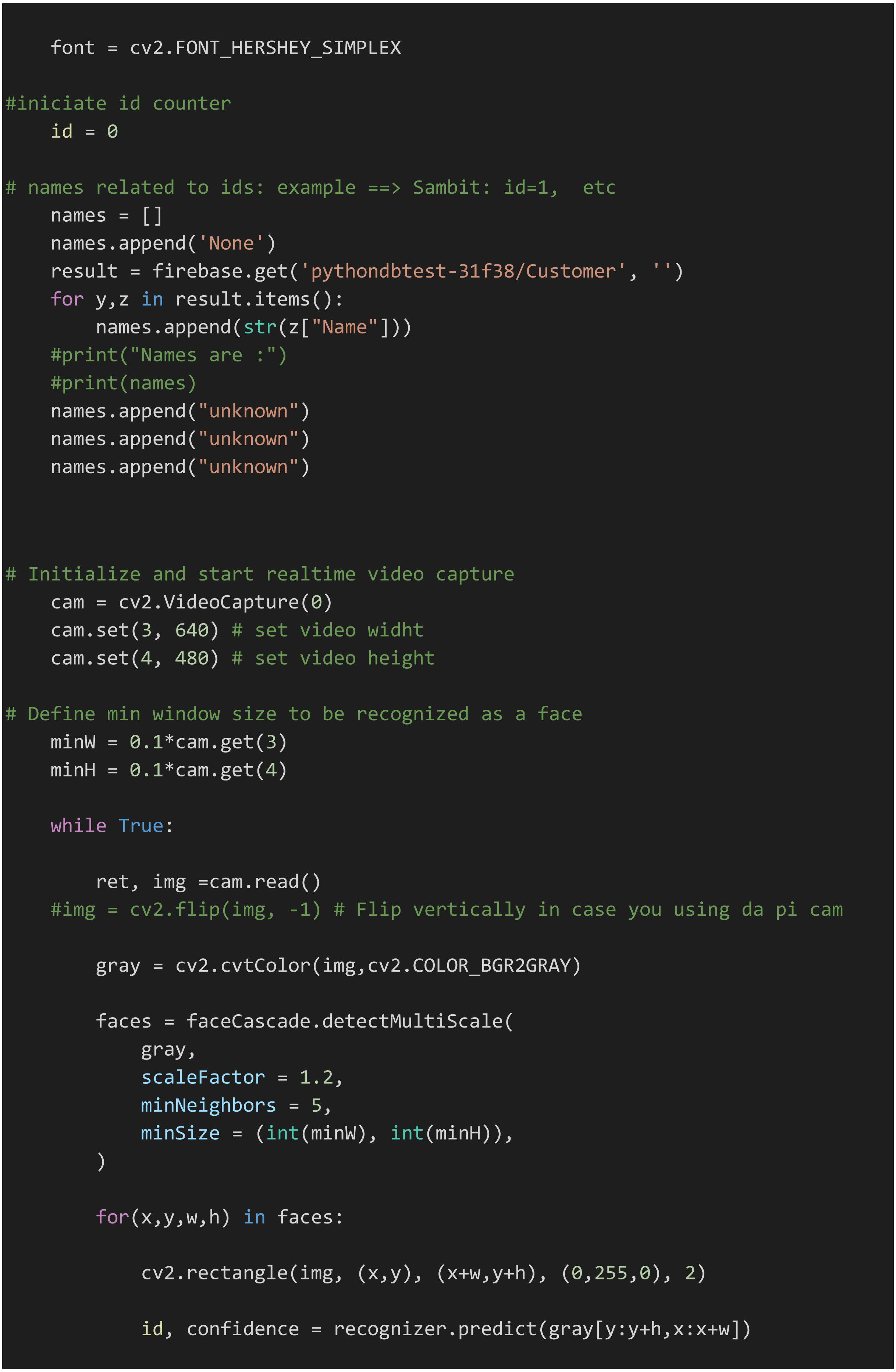


```
    rem=0
    if item=="done" or item=="Done" or item=="DONE":
        print("Inputs Done!")
        break
    if item=="invert":
        if invert==1:
            print("You have changed to Remove mode!\n Scan a QR to remove it!"
)
            invert=-1
        else:
            print("You are already in remove mode! No! You can't remove the re
move function")
            invert=-1
    if item=="add":
        if invert==-1:
            print("You have reverted back to add mode!\nScan QR to add items")
            invert=1
        else:
            print("You are already in add mode. No need to show off. Jeez.")
            invert=1

    if item in items and invert==-1:
        items.remove(item)
    if item in items and item!="add" and item!="invert":
        print("Item already exists!")
    else:
        if item not in items and item!="invert" and item!="add" and invert==1:
            items.append(item)
        blep=""
        x = re.search("^Potato.*$", item)
        if (x):
            if invert==1:
                bill["Potato"] = bill.get("Potato")+1
                blep="Potato"
                total=total+10
            if invert==-1:
                print("Potato currently:",bill["Potato"])
                if bill.get("Potato")>0:
                    blep="Potato"
                    bill["Potato"] = bill.get("Potato")-1
                    total=total-10
                    rem=1
                else:
                    print("Dude! You dont have any more Potatoes to remove!")
                    rem=0
        x = re.search("^Tomato.*$", item)
        if (x):
            if invert==1:
                bill["Tomato"] = bill.get("Tomato")+1
                blep="Tomato"
                total=total+20
            if invert==-1:
                print("Tomato currently:",bill["Tomato"])
                if bill.get("Tomato")>0:
                    blep="Tomato"
                    bill["Tomato"] = bill.get("Tomato")-1
                    total=total-20
                    rem=1
                else:
                    print("Dude! You dont have any more Tomatoes to remove!")
                    rem=0
        x = re.search("^Onion.*$", item)
        if (x):
            if invert==1:

                bill["Onion"] = bill.get("Onion")+1
                blep="Onion"
                total=total+30
            else:
                print("Onion currently:",bill["Onion"])
                if bill.get("Onion")>0:
                    blep="Onion"
                    bill["Onion"] = bill.get("Onion")-1
                    total=total-30
                    rem=1
                else:
                    print("Dude! You dont have any more Onions to remove!")
                    rem=0
        if item != "add" and item != "invert":
            if invert==1:
                print("New item:",blep)
            if invert==-1 and rem==1:
                print("Removed item:",blep)
            if invert==-1 and rem==0:
                print("Cant remove what wasnt added")
exsit = datetime.datetime.now()
print("You exited the shop at ",exsit)
bill["Exit_time"] = exsit
print("")
print("")
print("Your bill will be generated. Please wait!")
print("Hope to see you again",name,"! :)")
print("")
print("")
print("         Raghavendra Stores PVT LTD")
```

```
bill["Total"] = total
for y,z in bill.items():
    print(y,"  -  ",z)

firebase = firebase.FirebaseApplication('https://pythondbtest-31f38.firebaseio.com/', None)
result = firebase.post('pythondbtest-31f38/Bill/',bill)
print(result)
if uk==1:
    print("Stranger! Kindly Pay the amounnt of Rs.",total, "at the counter!")
    exit()
result = firebase.get('pythondbtest-31f38/Customer', '')
for y,z in result.items():
    if z["Name"]==name:
        print("User found!\n\n---------------------------")
        for i,j in z.items():
            print(i,j)
        uniq=y
        damt=z["Amount"]
        amt=damt-total
        if(amt<0):
            amt=-amt
            print("Oh no! looks like you do not have that much money! Please pay the remaining Rs.",amt," in cash to our cashier")
            amt=0
        else:
            print("Amount Decuced from your account : Rs.",total)
        firebase.put('pythondbtest-31f38/Customer/'+str(uniq),'Amount',amt)
        print('Record Updated')
        print("Current Amount : Rs.",amt)
```

Face.py:

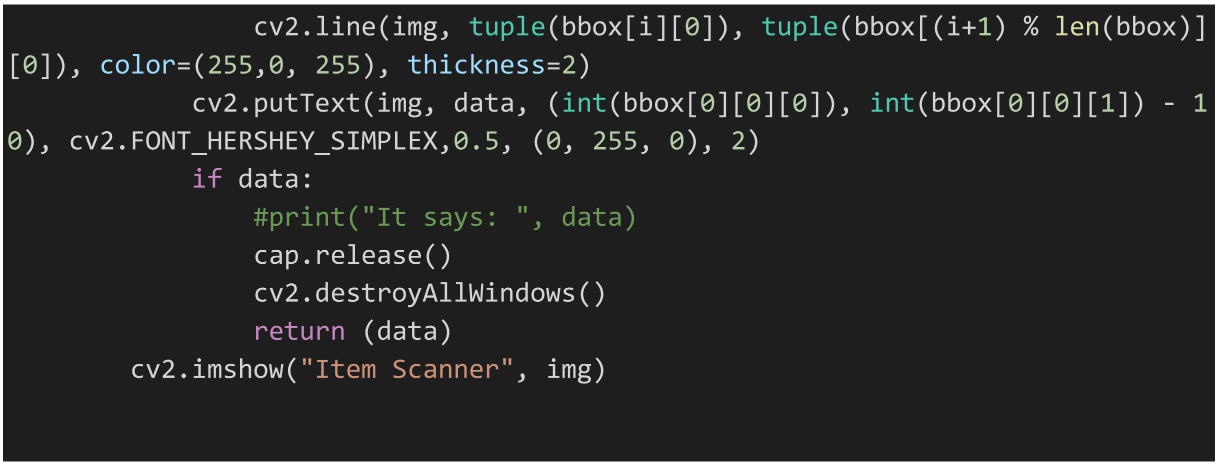
```
import cv2
import numpy as np
import os
from firebase import firebase
def main():
    from firebase import firebase
    firebase = firebase.FirebaseApplication('https://pythondbtest-31f38.firebaseio.com/', None)

    recognizer = cv2.face.LBPHFaceRecognizer_create()
    recognizer.read('trainer/trainer.yml')
    cascadePath = "haarcascade_frontalface_default.xml"
    faceCascade = cv2.CascadeClassifier(cascadePath);
```



```
    font = cv2.FONT_HERSHEY_SIMPLEX

#iniciate id counter
    id = 0

# names related to ids: example ==> Sambit: id=1,  etc
    names = []
    names.append('None')
    result = firebase.get('pythondbtest-31f38/Customer', '')
    for y,z in result.items():
        names.append(str(z["Name"]))
    #print("Names are :")
    #print(names)
    names.append("unknown")
    names.append("unknown")
    names.append("unknown")



# Initialize and start realtime video capture
    cam = cv2.VideoCapture(0)
    cam.set(3, 640) # set video widht
    cam.set(4, 480) # set video height

# Define min window size to be recognized as a face
    minW = 0.1*cam.get(3)
    minH = 0.1*cam.get(4)

    while True:

        ret, img =cam.read()
    #img = cv2.flip(img, -1) # Flip vertically in case you using da pi cam

        gray = cv2.cvtColor(img,cv2.COLOR_BGR2GRAY)

        faces = faceCascade.detectMultiScale(
            gray,
            scaleFactor = 1.2,
            minNeighbors = 5,
            minSize = (int(minW), int(minH)),
        )

        for(x,y,w,h) in faces:

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

            id, confidence = recognizer.predict(gray[y:y+h,x:x+w])
        # Check if confidence is less than 100 ==> "0" is perfect match
            if (confidence < 100):
                id = names[id]
                if ((100-confidence)>50):
                    #print(str(id)) remove the hash to debug
                    cam.release()
                    cv2.destroyAllWindows()
                    return(str(id))
                confidence = "  {0}%".format(round(100 - confidence))


            else:
                id = 'unknown'
                cam.release()
                cv2.destroyAllWindows()
                return(str(id))
                confidence = "  {0}%".format(round(100 - confidence))



            cv2.imshow('Please wait, while we admire your face',img)



# Do a bit of cleanup
    print("\n [INFO] Exiting Program and cleanup stuff")
    cam.release()
    cv2.destroyAllWindows()
```

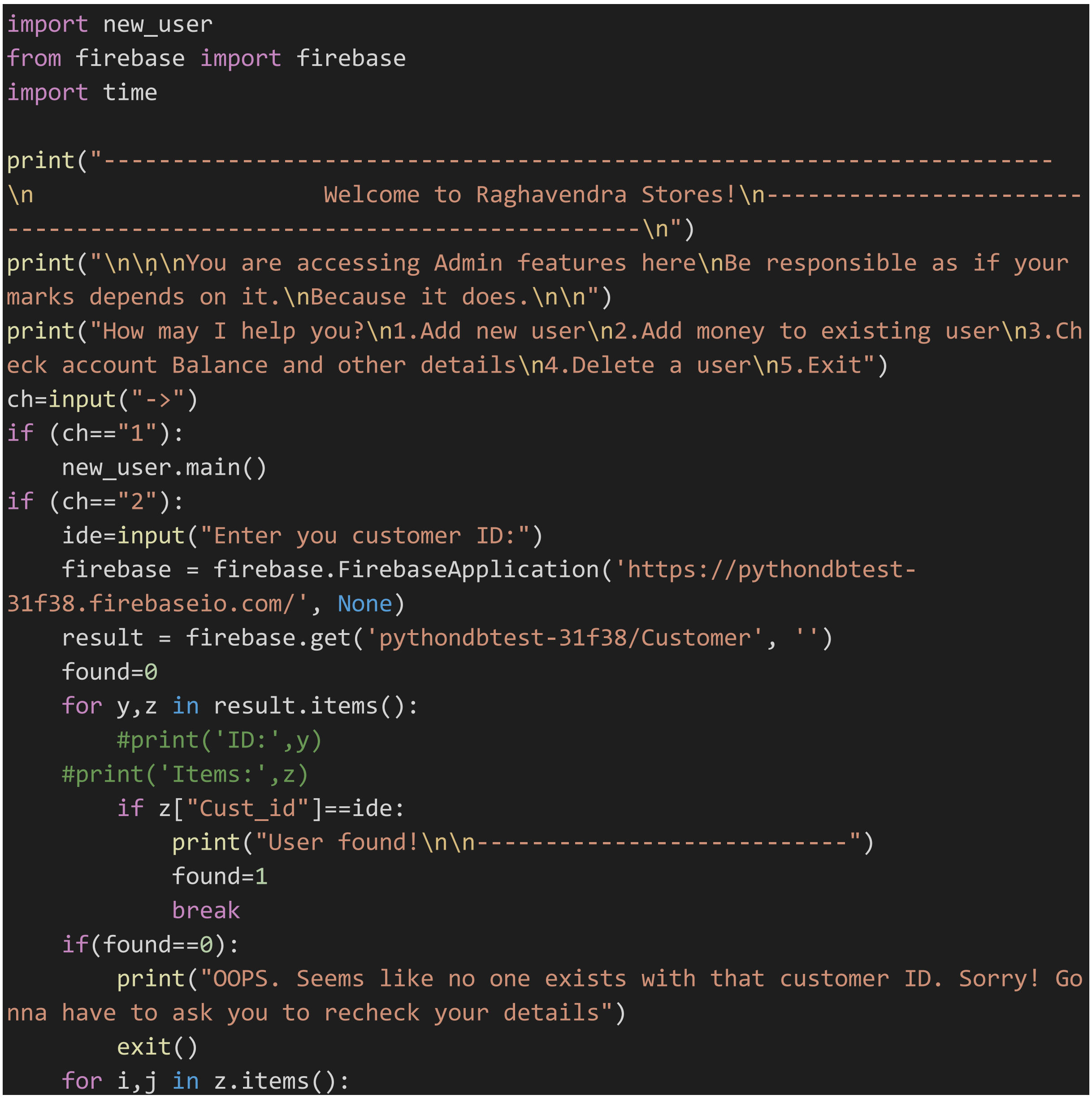
QR.py:

```
import cv2
def main():
    cap = cv2.VideoCapture(0)

    detector = cv2.QRCodeDetector()

    while True:
        _, img = cap.read()
        data, bbox, _ = detector.detectAndDecode(img)

        if(bbox is not None):
            for i in range(len(bbox)):
```

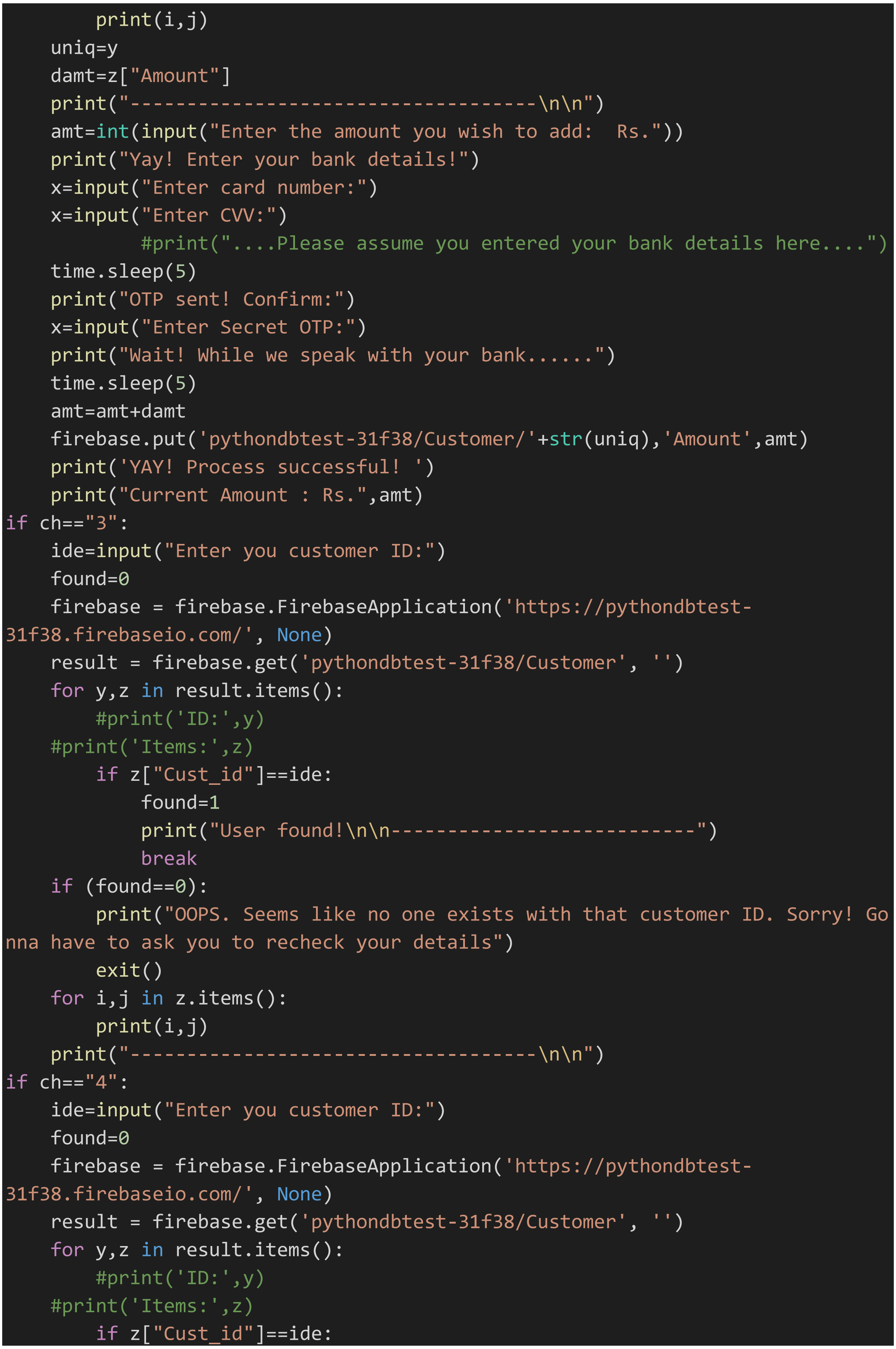


```
                cv2.line(img, tuple(bbox[i][0]), tuple(bbox[(i+1) % len(bbox)]
[0]), color=(255,0, 255), thickness=2)
            cv2.putText(img, data, (int(bbox[0][0][0]), int(bbox[0][0][1]) - 1
0), cv2.FONT_HERSHEY_SIMPLEX,0.5, (0, 255, 0), 2)
            if data:
                #print("It says: ", data)
                cap.release()
                cv2.destroyAllWindows()
                return (data)
        cv2.imshow("Item Scanner", img)
```

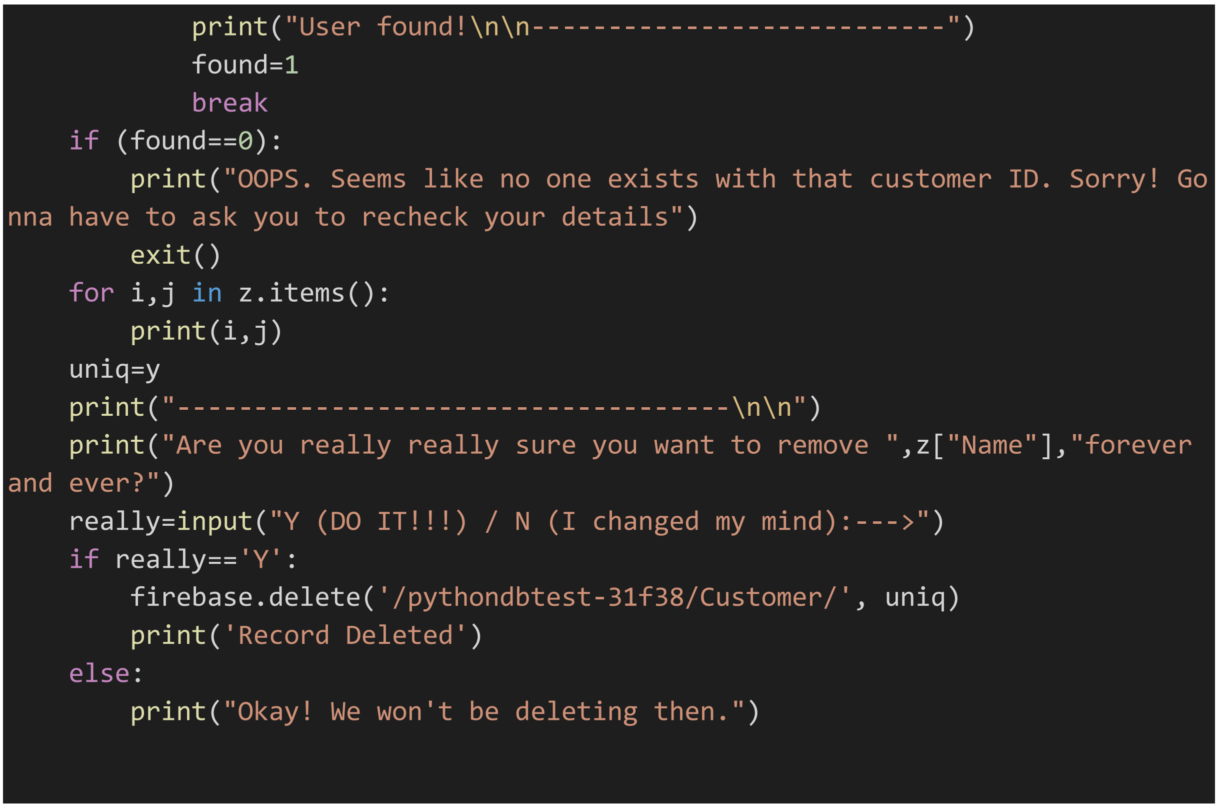
Admin.py:



```
import new_user
from firebase import firebase
import time

print("---------------------------------------------------------------------
\n                     Welcome to Raghavendra Stores!\n-----------------------
----------------------------------------------\n")
print("\n\ņ\nYou are accessing Admin features here\nBe responsible as if your
marks depends on it.\nBecause it does.\n\n")
print("How may I help you?\n1.Add new user\n2.Add money to existing user\n3.Ch
eck account Balance and other details\n4.Delete a user\n5.Exit")
ch=input("->")
if (ch=="1"):
    new_user.main()
if (ch=="2"):
    ide=input("Enter you customer ID:")
    firebase = firebase.FirebaseApplication('https://pythondbtest-
31f38.firebaseio.com/', None)
    result = firebase.get('pythondbtest-31f38/Customer', '')
    found=0
    for y,z in result.items():
        #print('ID:',y)
    #print('Items:',z)
        if z["Cust_id"]==ide:
            print("User found!\n\n---------------------------")
            found=1
            break
    if(found==0):
        print("OOPS. Seems like no one exists with that customer ID. Sorry! Go
nna have to ask you to recheck your details")
        exit()
    for i,j in z.items():
```

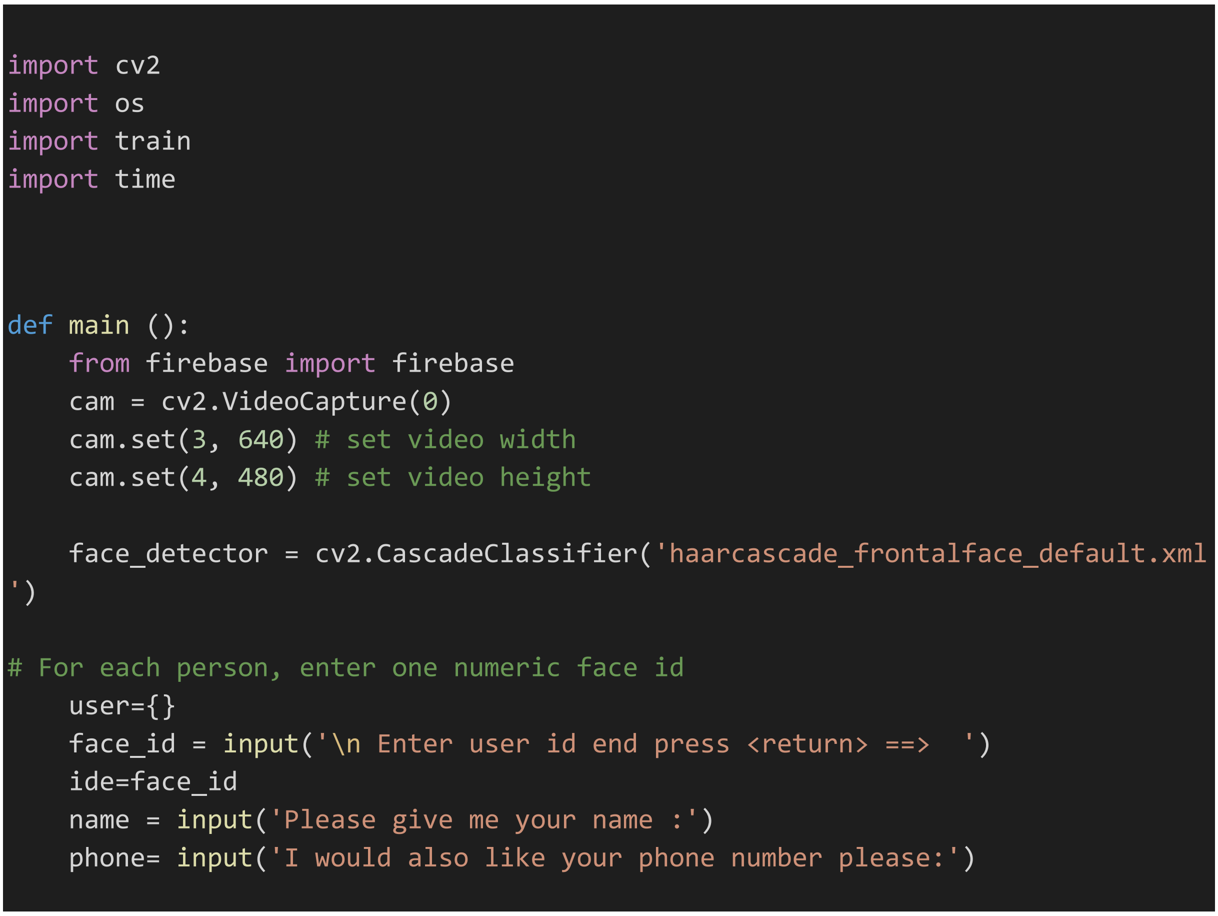


```
        print(i,j)
    uniq=y
    damt=z["Amount"]
    print("------------------------------------\n\n")
    amt=int(input("Enter the amount you wish to add:  Rs."))
    print("Yay! Enter your bank details!")
    x=input("Enter card number:")
    x=input("Enter CVV:")
            #print("....Please assume you entered your bank details here....")
    time.sleep(5)
    print("OTP sent! Confirm:")
    x=input("Enter Secret OTP:")
    print("Wait! While we speak with your bank......")
    time.sleep(5)
    amt=amt+damt
    firebase.put('pythondbtest-31f38/Customer/'+str(uniq),'Amount',amt)
    print('YAY! Process successful! ')
    print("Current Amount : Rs.",amt)
if ch=="3":
    ide=input("Enter you customer ID:")
    found=0
    firebase = firebase.FirebaseApplication('https://pythondbtest-
31f38.firebaseio.com/', None)
    result = firebase.get('pythondbtest-31f38/Customer', '')
    for y,z in result.items():
        #print('ID:',y)
    #print('Items:',z)
        if z["Cust_id"]==ide:
            found=1
            print("User found!\n\n---------------------------")
            break
    if (found==0):
        print("OOPS. Seems like no one exists with that customer ID. Sorry! Go
nna have to ask you to recheck your details")
        exit()
    for i,j in z.items():
        print(i,j)
    print("------------------------------------\n\n")
if ch=="4":
    ide=input("Enter you customer ID:")
    found=0
    firebase = firebase.FirebaseApplication('https://pythondbtest-
31f38.firebaseio.com/', None)
    result = firebase.get('pythondbtest-31f38/Customer', '')
    for y,z in result.items():
        #print('ID:',y)
    #print('Items:',z)
        if z["Cust_id"]==ide:
```

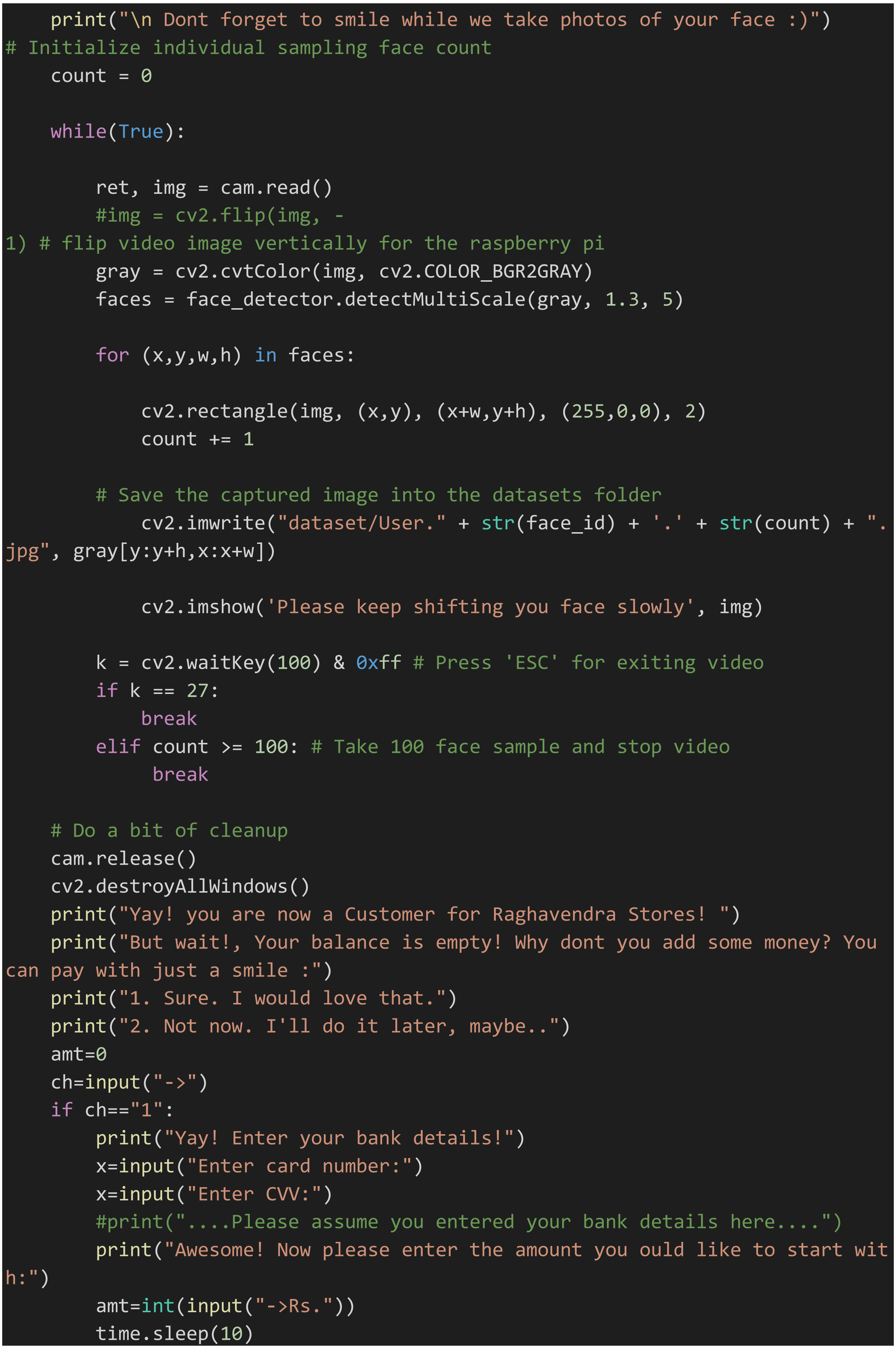


```
            print("User found!\n\n---------------------------")
            found=1
            break
    if (found==0):
        print("OOPS. Seems like no one exists with that customer ID. Sorry! Go
nna have to ask you to recheck your details")
        exit()
    for i,j in z.items():
        print(i,j)
    uniq=y
    print("------------------------------------\n\n")
    print("Are you really really sure you want to remove ",z["Name"],"forever
and ever?")
    really=input("Y (DO IT!!!) / N (I changed my mind):--->")
    if really=='Y':
        firebase.delete('/pythondbtest-31f38/Customer/', uniq)
        print('Record Deleted')
    else:
        print("Okay! We won't be deleting then.")
```

New_user.py:



```
import cv2
import os
import train
import time



def main ():
    from firebase import firebase
    cam = cv2.VideoCapture(0)
    cam.set(3, 640) # set video width
    cam.set(4, 480) # set video height

    face_detector = cv2.CascadeClassifier('haarcascade_frontalface_default.xml
')

# For each person, enter one numeric face id
    user={}
    face_id = input('\n Enter user id end press <return> ==>  ')
    ide=face_id
    name = input('Please give me your name :')
    phone= input('I would also like your phone number please:')
```



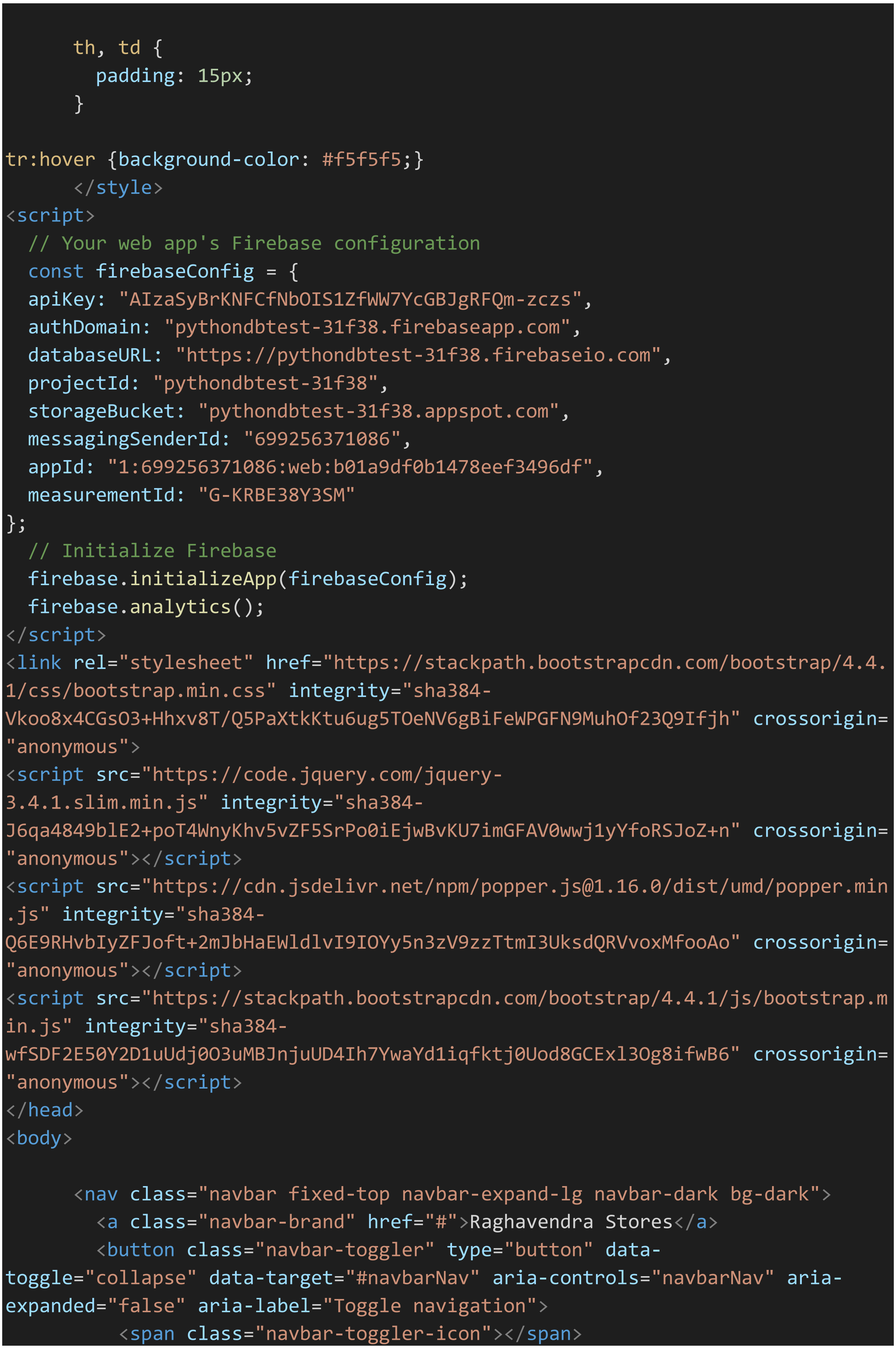
```
    print("\n Dont forget to smile while we take photos of your face :)")
# Initialize individual sampling face count
    count = 0

    while(True):

        ret, img = cam.read()
        #img = cv2.flip(img, -
1) # flip video image vertically for the raspberry pi
        gray = cv2.cvtColor(img, cv2.COLOR_BGR2GRAY)
        faces = face_detector.detectMultiScale(gray, 1.3, 5)

        for (x,y,w,h) in faces:

            cv2.rectangle(img, (x,y), (x+w,y+h), (255,0,0), 2)
            count += 1

        # Save the captured image into the datasets folder
            cv2.imwrite("dataset/User." + str(face_id) + '.' + str(count) + ".
jpg", gray[y:y+h,x:x+w])

            cv2.imshow('Please keep shifting you face slowly', img)

        k = cv2.waitKey(100) & 0xff # Press 'ESC' for exiting video
        if k == 27:
            break
        elif count >= 100: # Take 100 face sample and stop video
             break

    # Do a bit of cleanup
    cam.release()
    cv2.destroyAllWindows()
    print("Yay! you are now a Customer for Raghavendra Stores! ")
    print("But wait!, Your balance is empty! Why dont you add some money? You
can pay with just a smile :")
    print("1. Sure. I would love that.")
    print("2. Not now. I'll do it later, maybe..")
    amt=0
    ch=input("->")
    if ch=="1":
        print("Yay! Enter your bank details!")
        x=input("Enter card number:")
        x=input("Enter CVV:")
        #print("....Please assume you entered your bank details here....")
        print("Awesome! Now please enter the amount you ould like to start wit
h:")
        amt=int(input("->Rs."))
        time.sleep(10)
        print("OTP sent! Confirm:")
        x=input("Enter Secret OTP:")
        print("Wait! While we speak with your bank......")
        time.sleep(10)
        print("Payment is sucessfull! You now have Rs.", amt, "in your account
! Congratutations! >v<")
    print("Okay! Thank you for your time and patience ! You can now use our st
ore as a registered user!")
    user["Name"]=name
    user["Phone"]=phone
    user["Amount"]=amt
    user["Cust_id"]=ide
    firebase = firebase.FirebaseApplication('https://pythondbtest-
31f38.firebaseio.com/', None)
    result = firebase.post('pythondbtest-31f38/Customer/',user)
    print(result)
    train.main()
    print("\nAlways remember! Your customer id is :",ide)
```

Trainer.py:

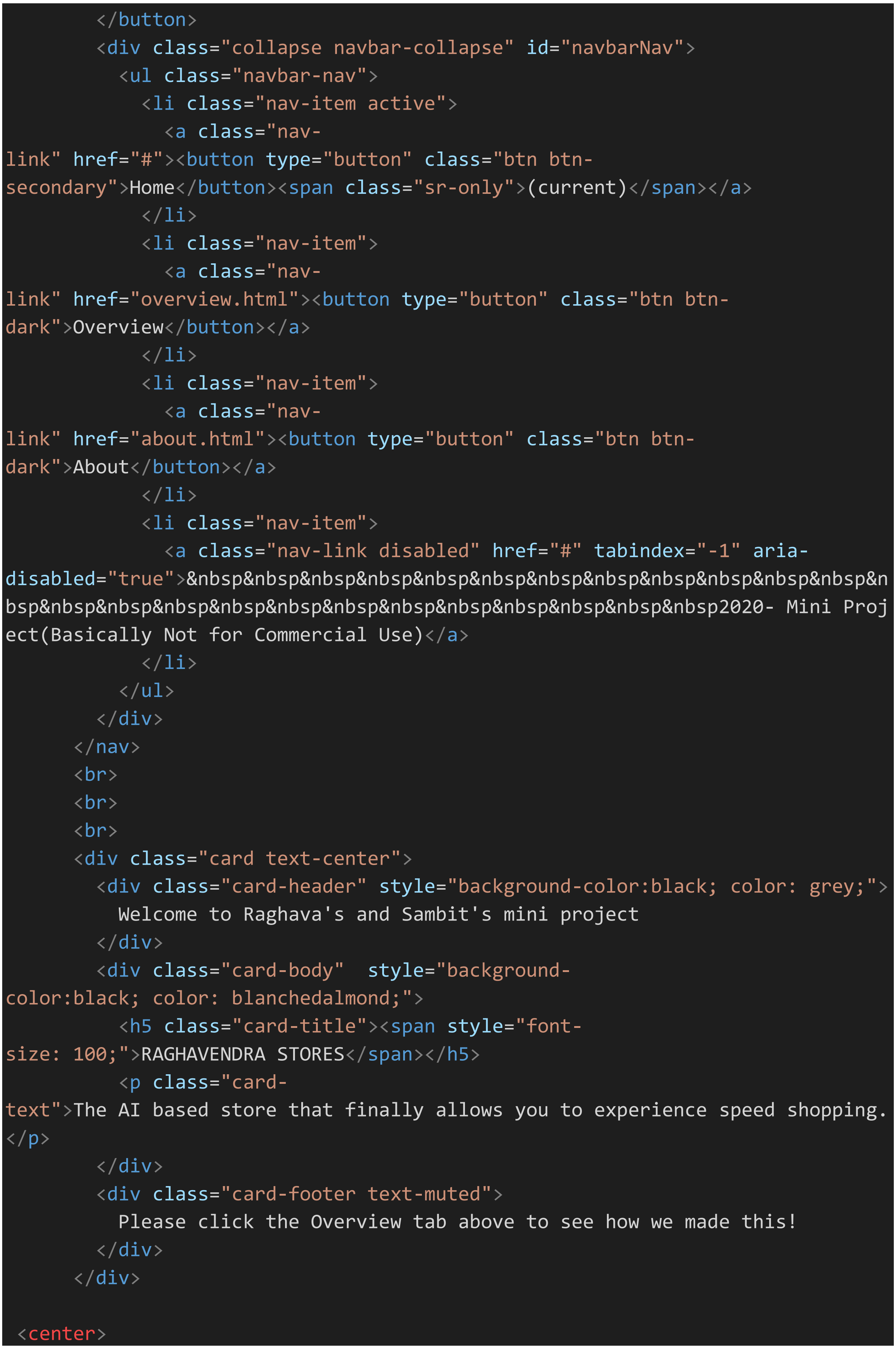
```
import cv2
import numpy as np
from PIL import Image
import os

path = 'dataset'
recognizer = cv2.face.LBPHFaceRecognizer_create()
detector = cv2.CascadeClassifier("haarcascade_frontalface_default.xml");

# function to get the images and label data
def getImagesAndLabels(path):

    imagePaths = [os.path.join(path,f) for f in os.listdir(path)]
    faceSamples=[]
    ids = []

    for imagePath in imagePaths:

        PIL_img = Image.open(imagePath).convert('L') # convert it to grayscale
        img_numpy = np.array(PIL_img,'uint8')

        id = int(os.path.split(imagePath)[-1].split(".")[1])
        faces = detector.detectMultiScale(img_numpy)
        for (x,y,w,h) in faces:
            faceSamples.append(img_numpy[y:y+h,x:x+w])
            ids.append(id)

    return faceSamples,ids
def main():
    os.remove("trainer/trainer.yml")
    print ("\nWe are Training faces. It will take a few seconds. Wait ...")
    faces,ids = getImagesAndLabels(path)
    recognizer.train(faces, np.array(ids))

# Save the model into trainer/trainer.yml
    recognizer.write('trainer/trainer.yml') # recognizer.save() worked on Mac,
 but not on Pi

# Print the numer of faces trained and end program
    print("\n [INFO] {0} faces trained. Exiting Program".format(len(np.unique(
ids))))
```

Index.html:

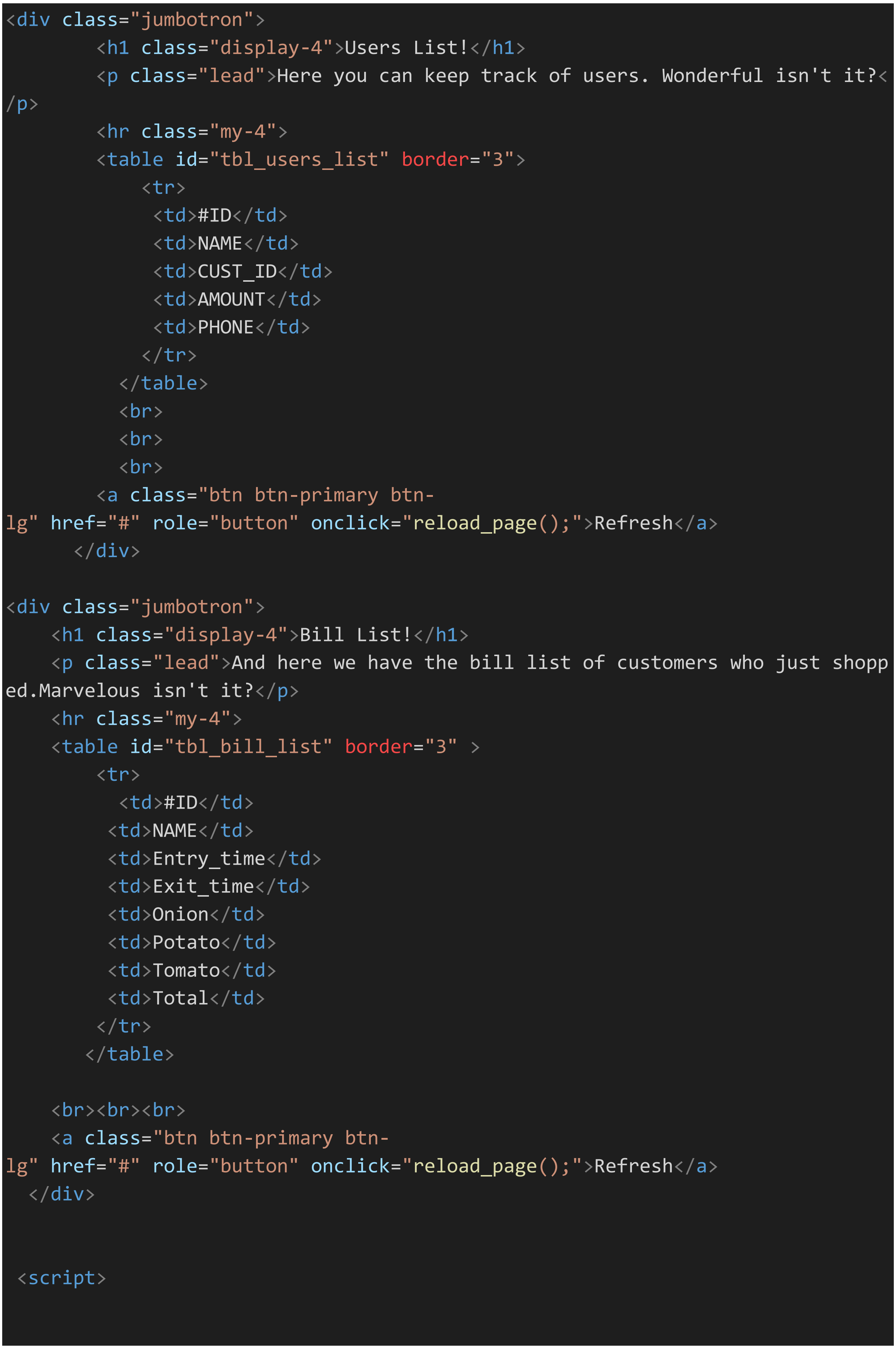
```
<html>
<head>
 <title>Raghavendra Stores Pvt. Ltd.</title>

<!-- The core Firebase JS SDK is always required and must be listed first -->
<script src="https://www.gstatic.com/firebasejs/7.13.2/firebase-
app.js"></script>


<!-- TODO: Add SDKs for Firebase products that you want to use
     https://firebase.google.com/docs/web/setup#available-libraries -->
<script src="https://www.gstatic.com/firebasejs/7.13.2/firebase-
analytics.js"></script>
<script src="https://www.gstatic.com/firebasejs/7.13.2/firebase-
database.js"></script>

<style>
    table, td, th {
        border: 1px solid #ddd;
        text-align: left;
      }

      table {
        border-collapse: collapse;
        width: 100%;
      }
```



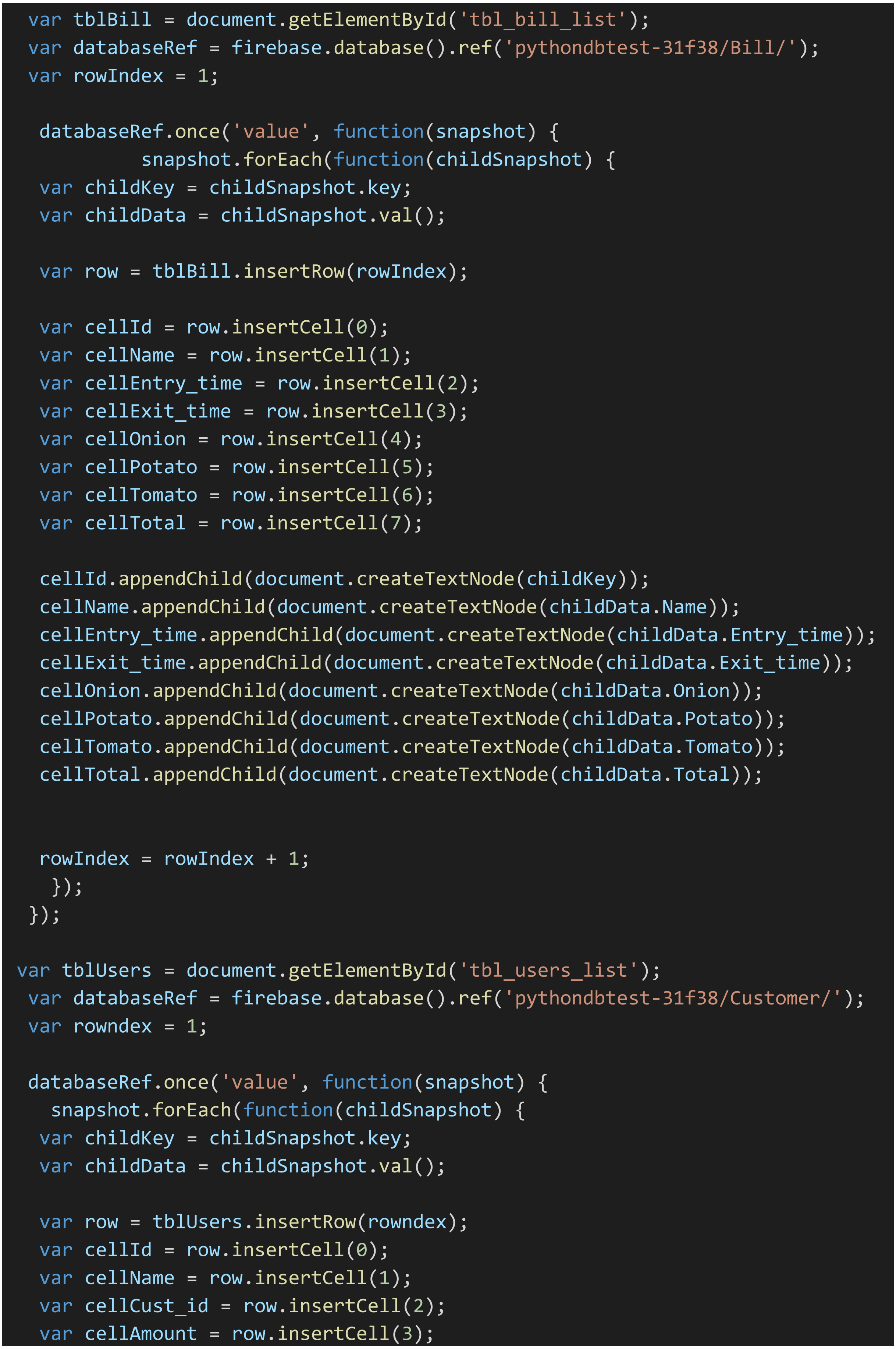
```
      th, td {
        padding: 15px;
      }

tr:hover {background-color: #f5f5f5;}
      </style>
<script>
  // Your web app's Firebase configuration
  const firebaseConfig = {
  apiKey: "AIzaSyBrKNFCfNbOIS1ZfWW7YcGBJgRFQm-zczs",
  authDomain: "pythondbtest-31f38.firebaseapp.com",
  databaseURL: "https://pythondbtest-31f38.firebaseio.com",
  projectId: "pythondbtest-31f38",
  storageBucket: "pythondbtest-31f38.appspot.com",
  messagingSenderId: "699256371086",
  appId: "1:699256371086:web:b01a9df0b1478eef3496df",
  measurementId: "G-KRBE38Y3SM"
};
  // Initialize Firebase
  firebase.initializeApp(firebaseConfig);
  firebase.analytics();
</script>
<link rel="stylesheet" href="https://stackpath.bootstrapcdn.com/bootstrap/4.4.
1/css/bootstrap.min.css" integrity="sha384-
Vkoo8x4CGsO3+Hhxv8T/Q5PaXtkKtu6ug5TOeNV6gBiFeWPGFN9MuhOf23Q9Ifjh" crossorigin=
"anonymous">
<script src="https://code.jquery.com/jquery-
3.4.1.slim.min.js" integrity="sha384-
J6qa4849blE2+poT4WnyKhv5vZF5SrPo0iEjwBvKU7imGFAV0wwj1yYfoRSJoZ+n" crossorigin=
"anonymous"></script>
<script src="https://cdn.jsdelivr.net/npm/popper.js@1.16.0/dist/umd/popper.min
.js" integrity="sha384-
Q6E9RHvbIyZFJoft+2mJbHaEWldlvI9IOYy5n3zV9zzTtmI3UksdQRVvoxMfooAo" crossorigin=
"anonymous"></script>
<script src="https://stackpath.bootstrapcdn.com/bootstrap/4.4.1/js/bootstrap.m
in.js" integrity="sha384-
wfSDF2E50Y2D1uUdj0O3uMBJnjuUD4Ih7YwaYd1iqfktj0Uod8GCExl3Og8ifwB6" crossorigin=
"anonymous"></script>
</head>
<body>

      <nav class="navbar fixed-top navbar-expand-lg navbar-dark bg-dark">
        <a class="navbar-brand" href="#">Raghavendra Stores</a>
        <button class="navbar-toggler" type="button" data-
toggle="collapse" data-target="#navbarNav" aria-controls="navbarNav" aria-
expanded="false" aria-label="Toggle navigation">
          <span class="navbar-toggler-icon"></span>
```



```
        </button>
        <div class="collapse navbar-collapse" id="navbarNav">
          <ul class="navbar-nav">
            <li class="nav-item active">
              <a class="nav-
link" href="#"><button type="button" class="btn btn-
secondary">Home</button><span class="sr-only">(current)</span></a>
            </li>
            <li class="nav-item">
              <a class="nav-
link" href="overview.html"><button type="button" class="btn btn-
dark">Overview</button></a>
            </li>
            <li class="nav-item">
              <a class="nav-
link" href="about.html"><button type="button" class="btn btn-
dark">About</button></a>
            </li>
            <li class="nav-item">
              <a class="nav-link disabled" href="#" tabindex="-1" aria-
disabled="true">&nbsp&nbsp&nbsp&nbsp&nbsp&nbsp&nbsp&nbsp&nbsp&nbsp&nbsp&nbsp&n
bsp&nbsp&nbsp&nbsp&nbsp&nbsp&nbsp&nbsp&nbsp&nbsp&nbsp&nbsp&nbsp2020- Mini Proj
ect(Basically Not for Commercial Use)</a>
            </li>
          </ul>
        </div>
      </nav>
      <br>
      <br>
      <br>
      <div class="card text-center">
        <div class="card-header" style="background-color:black; color: grey;">
          Welcome to Raghava's and Sambit's mini project
        </div>
        <div class="card-body"  style="background-
color:black; color: blanchedalmond;">
          <h5 class="card-title"><span style="font-
size: 100;">RAGHAVENDRA STORES</span></h5>
          <p class="card-
text">The AI based store that finally allows you to experience speed shopping.
</p>
        </div>
        <div class="card-footer text-muted">
          Please click the Overview tab above to see how we made this!
        </div>
      </div>

 <center>
```



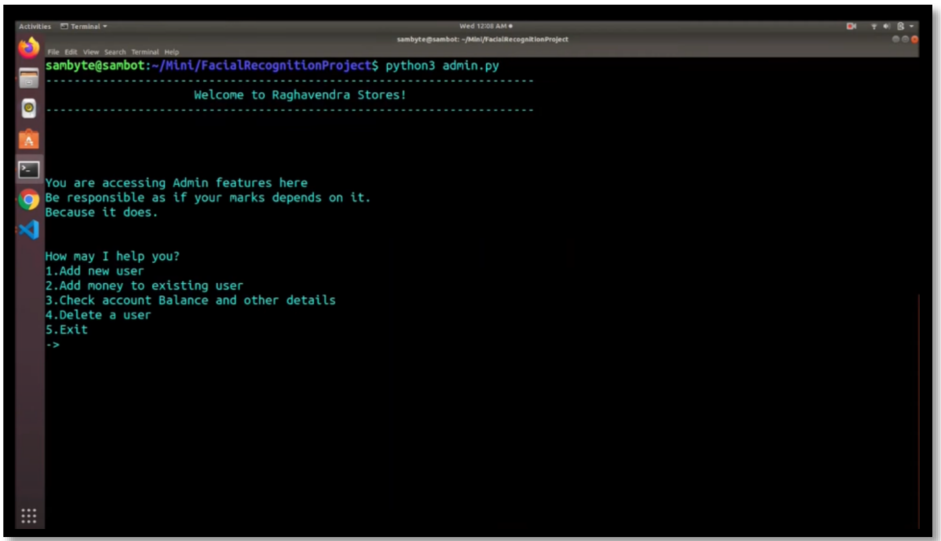
```
<div class="jumbotron">
        <h1 class="display-4">Users List!</h1>
        <p class="lead">Here you can keep track of users. Wonderful isn't it?<
/p>
        <hr class="my-4">
        <table id="tbl_users_list" border="3">
            <tr>
             <td>#ID</td>
             <td>NAME</td>
             <td>CUST_ID</td>
             <td>AMOUNT</td>
             <td>PHONE</td>
            </tr>
          </table>
          <br>
          <br>
          <br>
        <a class="btn btn-primary btn-
lg" href="#" role="button" onclick="reload_page();">Refresh</a>
      </div>

<div class="jumbotron">
    <h1 class="display-4">Bill List!</h1>
    <p class="lead">And here we have the bill list of customers who just shopp
ed.Marvelous isn't it?</p>
    <hr class="my-4">
    <table id="tbl_bill_list" border="3" >
        <tr>
          <td>#ID</td>
         <td>NAME</td>
         <td>Entry_time</td>
         <td>Exit_time</td>
         <td>Onion</td>
         <td>Potato</td>
         <td>Tomato</td>
         <td>Total</td>
        </tr>
       </table>

    <br><br><br>
    <a class="btn btn-primary btn-
lg" href="#" role="button" onclick="reload_page();">Refresh</a>
  </div>


 <script>
```



```
 var tblBill = document.getElementById('tbl_bill_list');
 var databaseRef = firebase.database().ref('pythondbtest-31f38/Bill/');
 var rowIndex = 1;

  databaseRef.once('value', function(snapshot) {
           snapshot.forEach(function(childSnapshot) {
  var childKey = childSnapshot.key;
  var childData = childSnapshot.val();

  var row = tblBill.insertRow(rowIndex);

  var cellId = row.insertCell(0);
  var cellName = row.insertCell(1);
  var cellEntry_time = row.insertCell(2);
  var cellExit_time = row.insertCell(3);
  var cellOnion = row.insertCell(4);
  var cellPotato = row.insertCell(5);
  var cellTomato = row.insertCell(6);
  var cellTotal = row.insertCell(7);

  cellId.appendChild(document.createTextNode(childKey));
  cellName.appendChild(document.createTextNode(childData.Name));
  cellEntry_time.appendChild(document.createTextNode(childData.Entry_time));
  cellExit_time.appendChild(document.createTextNode(childData.Exit_time));
  cellOnion.appendChild(document.createTextNode(childData.Onion));
  cellPotato.appendChild(document.createTextNode(childData.Potato));
  cellTomato.appendChild(document.createTextNode(childData.Tomato));
  cellTotal.appendChild(document.createTextNode(childData.Total));


  rowIndex = rowIndex + 1;
   });
 });

var tblUsers = document.getElementById('tbl_users_list');
 var databaseRef = firebase.database().ref('pythondbtest-31f38/Customer/');
 var rowndex = 1;

 databaseRef.once('value', function(snapshot) {
   snapshot.forEach(function(childSnapshot) {
  var childKey = childSnapshot.key;
  var childData = childSnapshot.val();

  var row = tblUsers.insertRow(rowndex);
  var cellId = row.insertCell(0);
  var cellName = row.insertCell(1);
  var cellCust_id = row.insertCell(2);
  var cellAmount = row.insertCell(3);
```

```
   var cellPhone = row.insertCell(4);

   cellId.appendChild(document.createTextNode(childKey));
   cellName.appendChild(document.createTextNode(childData.Name));
   cellCust_id.appendChild(document.createTextNode(childData.Cust_id));
   cellAmount.appendChild(document.createTextNode(childData.Amount));
   cellPhone.appendChild(document.createTextNode(childData.Phone));


   rowndex = rowndex + 1;
    });
  });

  function reload_page(){
   window.location.reload();
  }
 </script>
</center>
<div class="alert alert-warning alert-dismissible fade show" role="alert">
    <strong>Hey buddy!</strong> I Just wanted to let you know that Raghava and
 Sambit have put in a lot of effort for this. Kindly refrain from copying Sour
ce codes. <i>Oh Click the 'x' thingie to close this. Thanks</i>
    <button type="button" class="close" data-dismiss="alert" aria-
label="Close">
      <span aria-hidden="true">&times;</span>
    </button>
  </div>
</body>
</html>
```

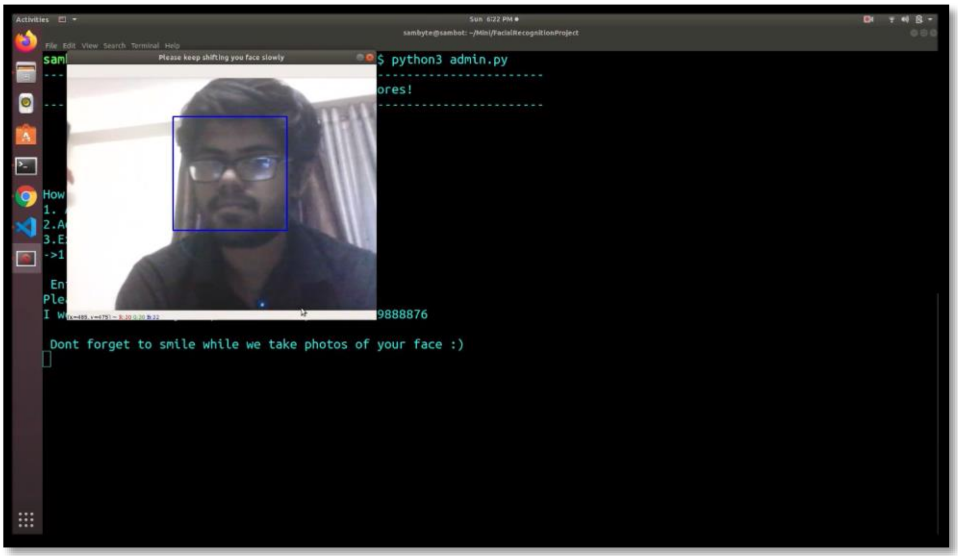
2. Snapshots:



Activities
Terminal
Wed 12:08 AM
sambyte@sambot: ~/Mini/FacialRecognitionProject
File Edit View Search Terminal Help

```
sambyte@sambot:~/Mini/FacialRecognitionProject$ python3 admin.py
--------------------------------------------------------------------
                     Welcome to Raghavendra Stores!
--------------------------------------------------------------------



You are accessing Admin features here
Be responsible as if your marks depends on it.
Because it does.


How may I help you?
1.Add new user
2.Add money to existing user
3.Check account Balance and other details
4.Delete a user
5.Exit
->
```

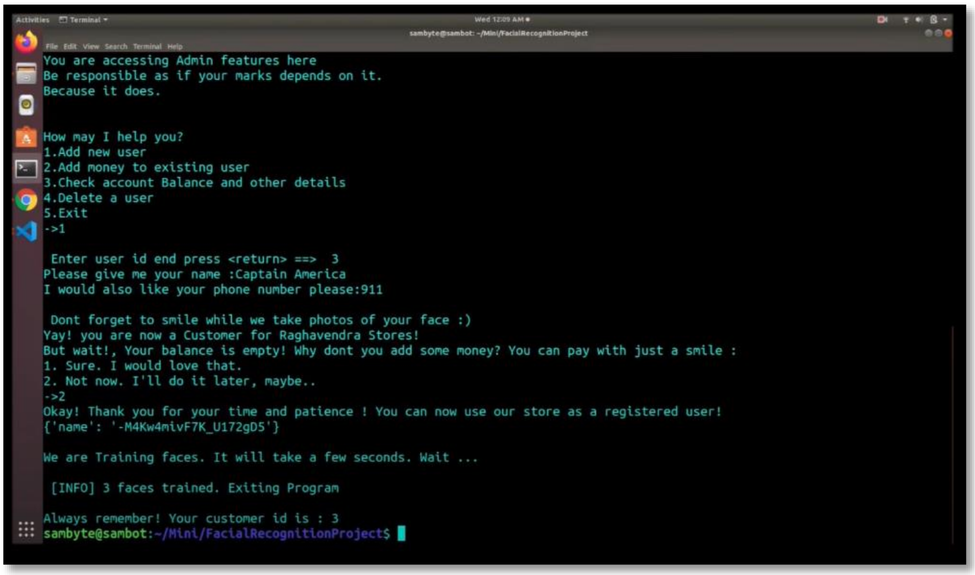
Admin page Menu



Activities
Sun 6:22 PM
sambyte@sambot: ~/Mini/FacialRecognitionProject
File Edit View Search Terminal Help
Please keep shifting you face slowly

```
sam                                            $ python3 admin.py
---                                            -----------------------
                                               ores!
---                                            -----------------------




How
1.
2.A
3.E
->1

 En
Ple
I w                                            9888876

 Dont forget to smile while we take photos of your face :)
```

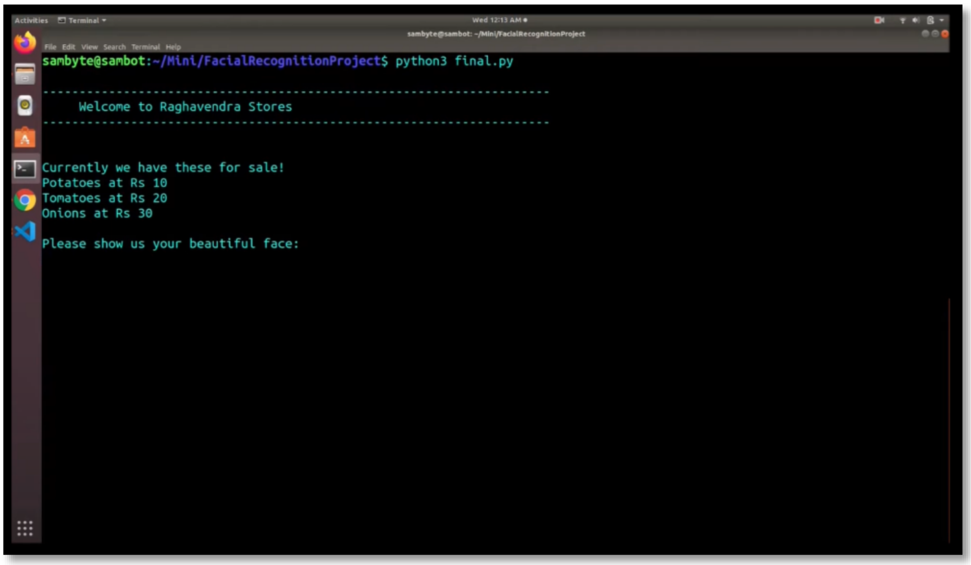
Admin Function → Gathering dataset



Activities
Terminal
Wed 12:09 AM
sambyte@sambot: ~/Mini/FacialRecognitionProject
File Edit View Search Terminal Help

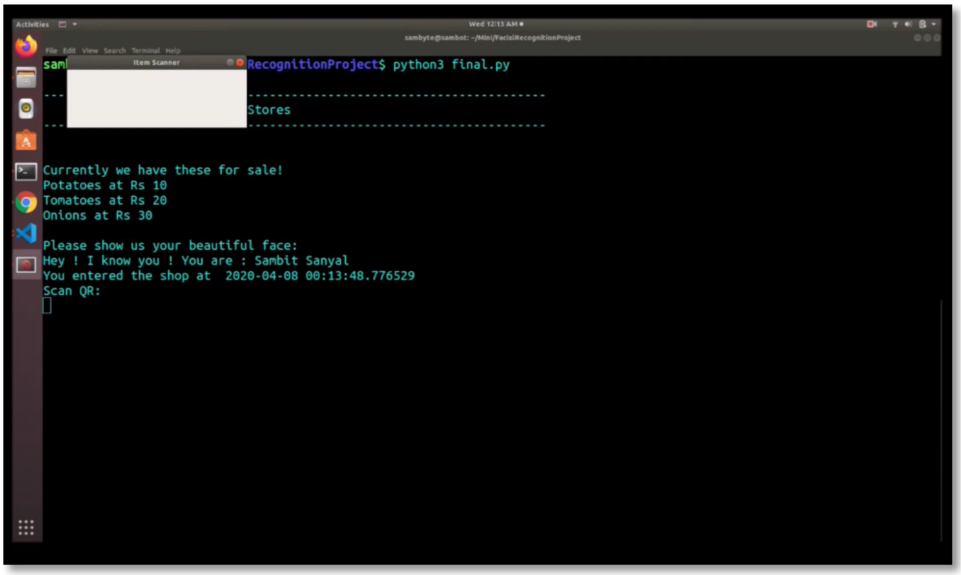
```
You are accessing Admin features here
Be responsible as if your marks depends on it.
Because it does.


How may I help you?
1.Add new user
2.Add money to existing user
3.Check account Balance and other details
4.Delete a user
5.Exit
->1

 Enter user id end press <return> ==>  3
Please give me your name :Captain America
I would also like your phone number please:911

 Dont forget to smile while we take photos of your face :)
Yay! you are now a Customer for Raghavendra Stores!
But wait!, Your balance is empty! Why dont you add some money? You can pay with just a smile :
1. Sure. I would love that.
2. Not now. I'll do it later, maybe..
->2
Okay! Thank you for your time and patience ! You can now use our store as a registered user!
{'name': '-M4Kw4mivF7K_U172gD5'}

We are Training faces. It will take a few seconds. Wait ...

 [INFO] 3 faces trained. Exiting Program

Always remember! Your customer id is : 3
sambyte@sambot:~/Mini/FacialRecognitionProject$ 
```

Admin Function → Add new user



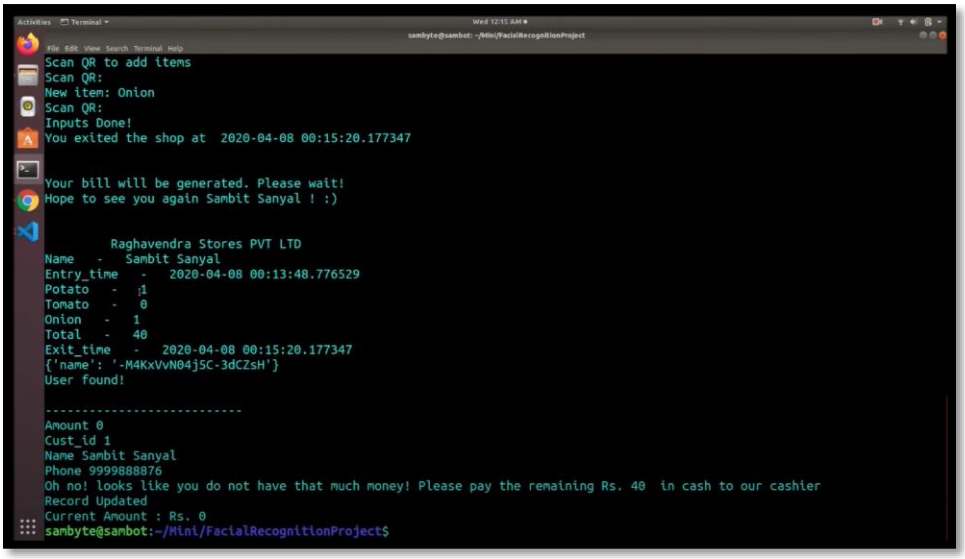
```
sambyte@sambot:~/Mini/FacialRecognitionProject$ python3 final.py

---------------------------------------------------------------------
     Welcome to Raghavendra Stores
---------------------------------------------------------------------


Currently we have these for sale!
Potatoes at Rs 10
Tomatoes at Rs 20
Onions at Rs 30

Please show us your beautiful face:
```

Raghavendra Store → Face-Recognition



```
sam                         RecognitionProject$ python3 final.py

---                         ------------------------------------------
                            Stores
---                         ------------------------------------------


Currently we have these for sale!
Potatoes at Rs 10
Tomatoes at Rs 20
Onions at Rs 30

Please show us your beautiful face:
Hey ! I know you ! You are : Sambit Sanyal
You entered the shop at  2020-04-08 00:13:48.776529
Scan QR:
```

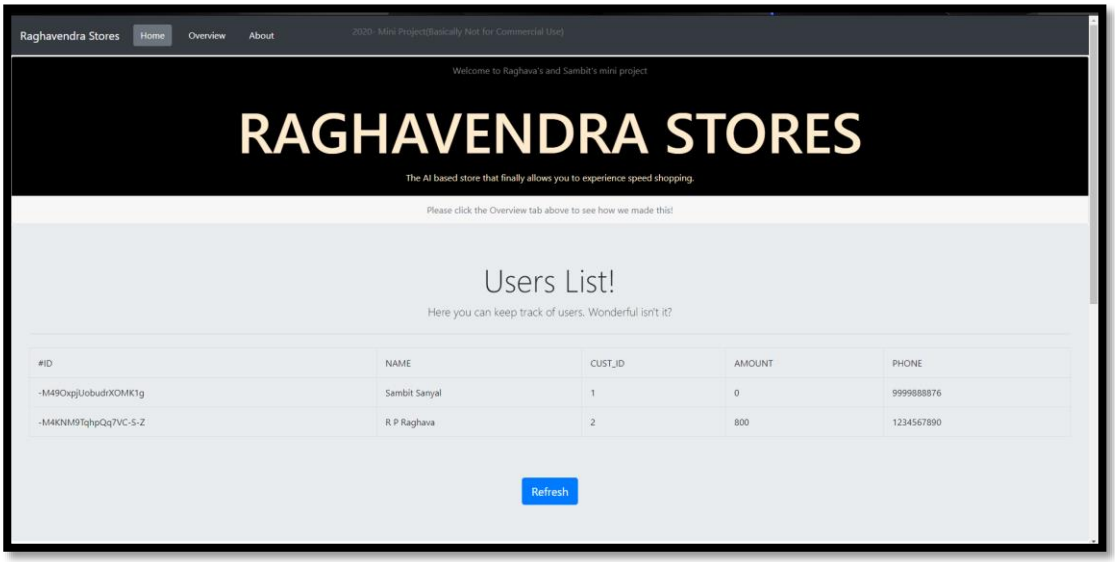
Raghavendra Store → QR code Inputting



Activities
Terminal
Wed 12:15 AM
sambyte@sambot: ~/Mini/FacialRecognitionProject
File Edit View Search Terminal Help

```
Scan QR to add items
Scan QR:
New item: Onion
Scan QR:
Inputs Done!
You exited the shop at  2020-04-08 00:15:20.177347


Your bill will be generated. Please wait!
Hope to see you again Sambit Sanyal ! :)


         Raghavendra Stores PVT LTD
Name   -   Sambit Sanyal
Entry_time   -    2020-04-08 00:13:48.776529
Potato   -   1
Tomato   -   0
Onion   -   1
Total   -   40
Exit_time   -   2020-04-08 00:15:20.177347
{'name': '-M4KxVvN04j5C-3dCZsH'}
User found!

---------------------------
Amount 0
Cust_id 1
Name Sambit Sanyal
Phone 9999888876
Oh no! looks like you do not have that much money! Please pay the remaining Rs. 40  in cash to our cashier
Record Updated
Current Amount : Rs. 0
sambyte@sambot:~/Mini/FacialRecognitionProject$
```

Raghavendra Store → Billing and Payment



Welcome to Raghava's and Sambit's mini project

RAGHAVENDRA STORES

The AI based store that finally allows you to experience speed shopping.

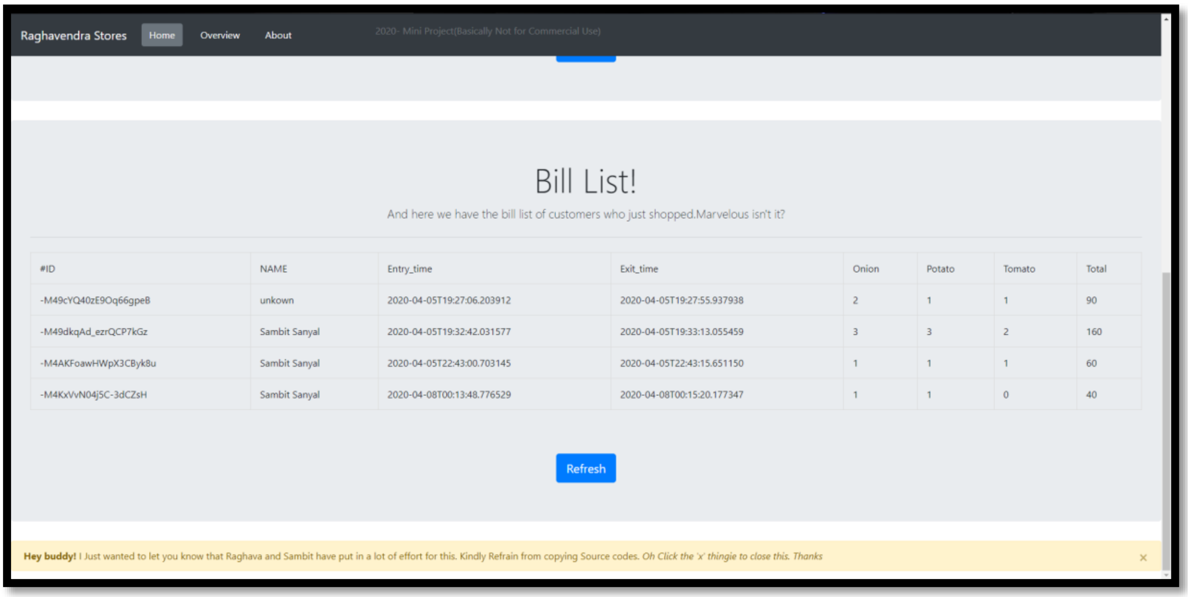
Please click the Overview tab above to see how we made this!

Users List!

Here you can keep track of users. Wonderful isn't it?

#ID	NAME	CUST_ID	AMOUNT	PHONE
-M49OxpjUobudrXOMK1g	Sambit Sanyal	1	0	9999888876
-M4KNM9TqhpQq7VC-S-Z	R P Raghava	2	800	1234567890

Raghavendra Store → Website for Database (Customers)



Bill List!

And here we have the bill list of customers who just shopped.Marvelous isn't it?

#ID	NAME	Entry_time	Exit_time	Onion	Potato	Tomato	Total
-M49cYQ40zE9Oq66gpeB	unkown	2020-04-05T19:27:06.203912	2020-04-05T19:27:55.937938	2	1	1	90
-M49dkqAd_ezrQCP7kGz	Sambit Sanyal	2020-04-05T19:32:42.031577	2020-04-05T19:33:13.055459	3	3	2	160
-M4AKFoawHWpX3CByk8u	Sambit Sanyal	2020-04-05T22:43:00.703145	2020-04-05T22:43:15.651150	1	1	1	60
-M4KxVvN04j5C-3dCZsH	Sambit Sanyal	2020-04-08T00:13:48.776529	2020-04-08T00:15:20.177347	1	1	0	40

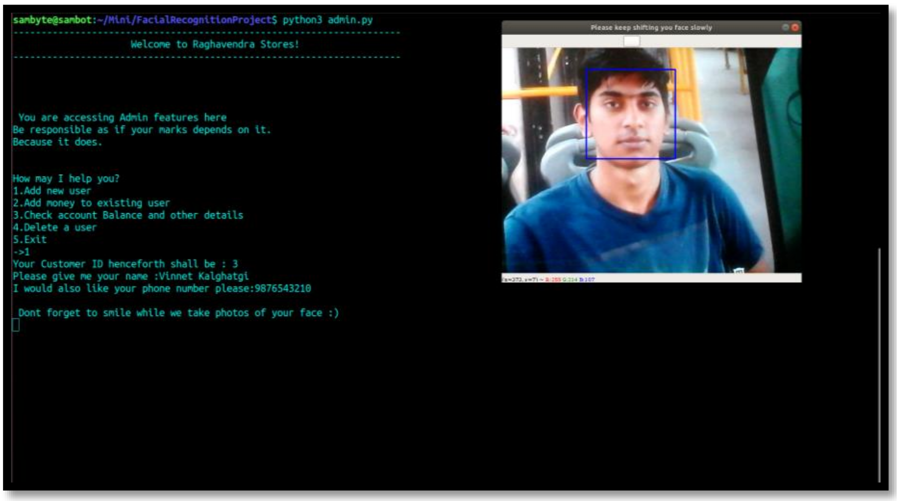
Hey buddy! I Just wanted to let you know that Raghava and Sambit have put in a lot of effort for this. Kindly Refrain from copying Source codes. *Oh Click the 'x' thingie to close this. Thanks*

Raghavendra Store → Website for Database (Bills)

3. Testing

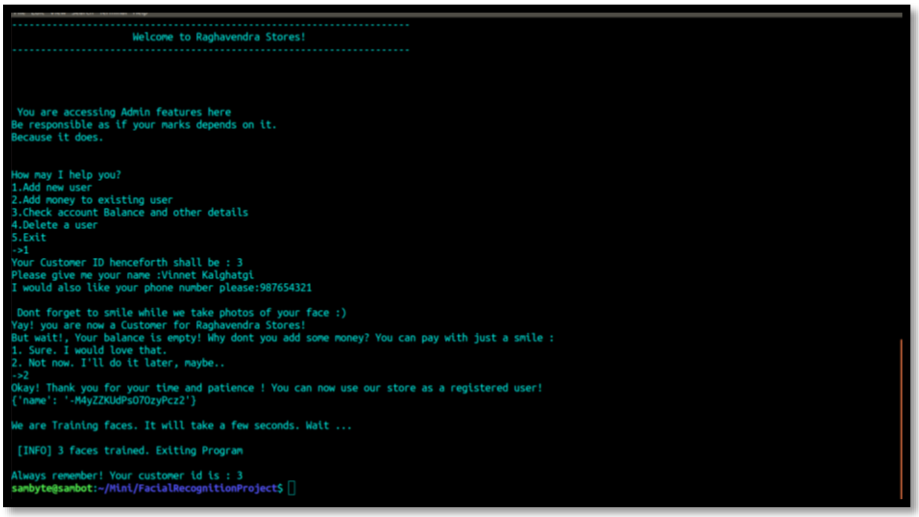
1. Trying to add a new user!

First we gather the face data



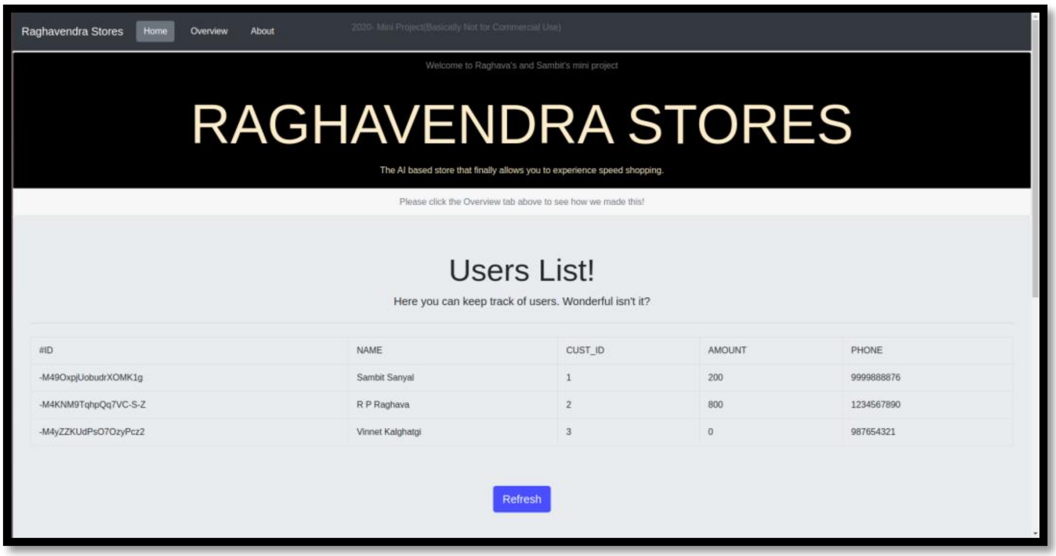
```
sambyte@sambot:~/Mini/FacialRecognitionProject$ python3 admin.py
--------------------------------------------------------------------
                    Welcome to Raghavendra Stores!
--------------------------------------------------------------------



 You are accessing Admin features here
Be responsible as if your marks depends on it.
Because it does.


How may I help you?
1.Add new user
2.Add money to existing user
3.Check account Balance and other details
4.Delete a user
5.Exit
->1
Your Customer ID henceforth shall be : 3
Please give me your name :Vinnet Kalghatgi
I would also like your phone number please:9876543210

 Dont forget to smile while we take photos of your face :)
```

Then the store registers the user



```
--------------------------------------------------------------------
                    Welcome to Raghavendra Stores!
--------------------------------------------------------------------



 You are accessing Admin features here
Be responsible as if your marks depends on it.
Because it does.


How may I help you?
1.Add new user
2.Add money to existing user
3.Check account Balance and other details
4.Delete a user
5.Exit
->1
Your Customer ID henceforth shall be : 3
Please give me your name :Vinnet Kalghatgi
I would also like your phone number please:987654321

 Dont forget to smile while we take photos of your face :)
Yay! you are now a Customer for Raghavendra Stores!
But wait!, Your balance is empty! Why dont you add some money? You can pay with just a smile :
1. Sure. I would love that.
2. Not now. I'll do it later, maybe..
->2
Okay! Thank you for your time and patience ! You can now use our store as a registered user!
{'name': '-M4yZZKUdPsO7OzyPcz2'}

We are Training faces. It will take a few seconds. Wait ...

 [INFO] 3 faces trained. Exiting Program

Always remember! Your customer id is : 3
sambyte@sambot:~/Mini/FacialRecognitionProject$
```

Then we see the results in the website that a new customer is added


Raghavendra Stores
Home
Overview
About
2020- Mini Project(Basically Not for Commercial Use)
Welcome to Raghava's and Sambit's mini project
RAGHAVENDRA STORES
The AI based store that finally allows you to experience speed shopping.
Please click the Overview tab above to see how we made this!
Refresh

Users List!

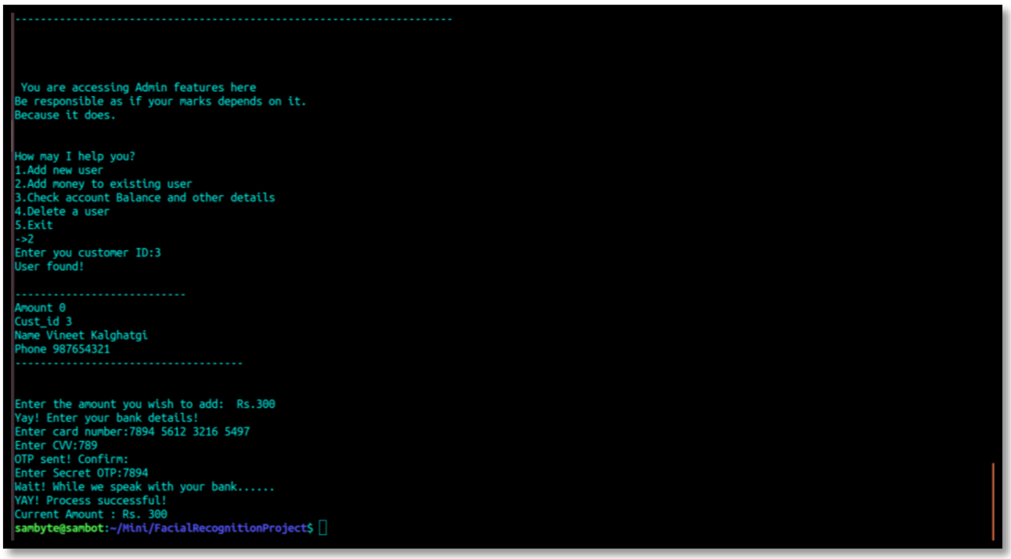
Here you can keep track of users. Wonderful isn't it?

#ID	NAME	CUST_ID	AMOUNT	PHONE
-M49OxpjUobudrXOMK1g	Sambit Sanyal	1	200	9999888876
-M4KNM9TqhpQq7VC-S-Z	R P Raghava	2	800	1234567890
-M4yZZKUdPsO7OzyPcz2	Vinnet Kalghatgi	3	0	987654321

Trying to add a new user: SUCCESS!

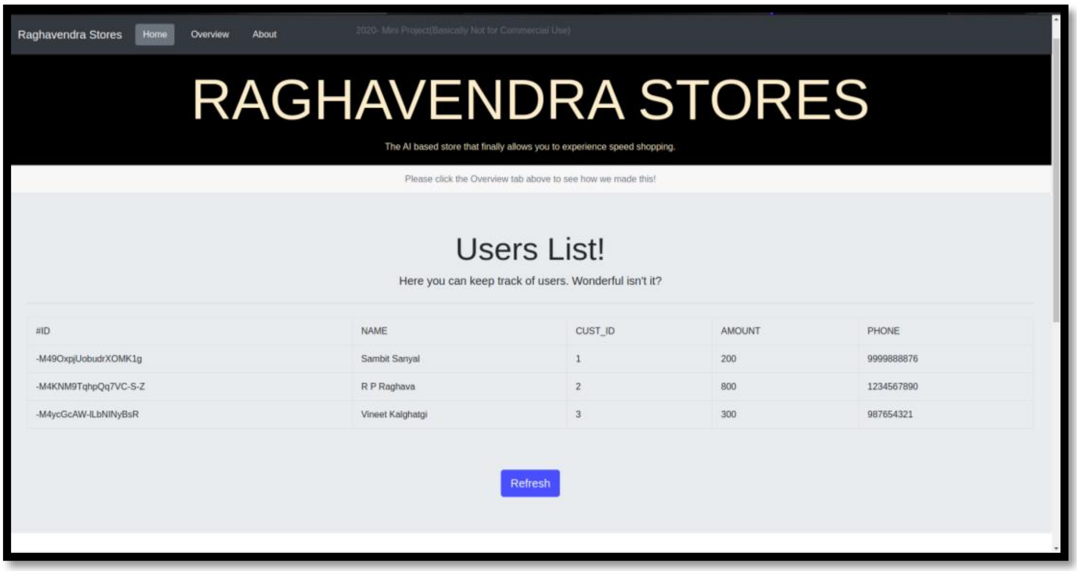
2. Trying to add money to existing user!

First we enter select "Add Money" and the customer ID



```
---------------------------------------------------------------------



 You are accessing Admin features here
Be responsible as if your marks depends on it.
Because it does.


How may I help you?
1.Add new user
2.Add money to existing user
3.Check account Balance and other details
4.Delete a user
5.Exit
->2
Enter you customer ID:3
User found!

---------------------------
Amount 0
Cust_id 3
Name Vineet Kalghatgi
Phone 987654321
------------------------------------



Enter the amount you wish to add:  Rs.300
Yay! Enter your bank details!
Enter card number:7894 5612 3216 5497
Enter CVV:789
OTP sent! Confirm:
Enter Secret OTP:7894
Wait! While we speak with your bank......
YAY! Process successful!
Current Amount : Rs. 300
sambyte@sambot:~/Mini/FacialRecognitionProject$
```

Then we see if money was added in the website!



RAGHAVENDRA STORES

The AI based store that finally allows you to experience speed shopping.

Please click the Overview tab above to see how we made this!

Users List!

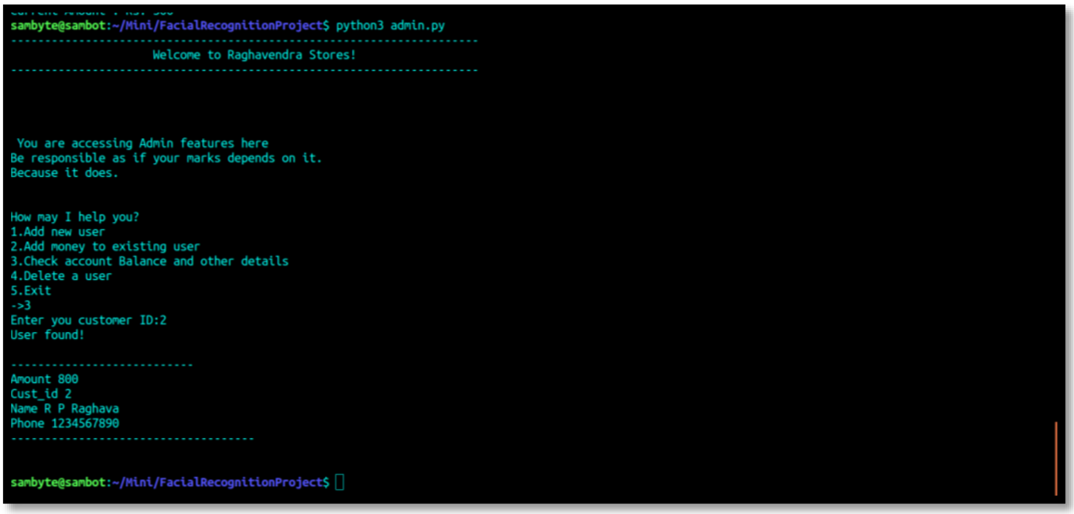
Here you can keep track of users. Wonderful isn't it?

#ID	NAME	CUST_ID	AMOUNT	PHONE
-M49OxpjUobudrXOMK1g	Sambit Sanyal	1	200	9999888876
-M4KNM9TqhpQq7VC-S-Z	R P Raghava	2	800	1234567890
-M4ycGcAW-ILbNINyBsR	Vineet Kalghatgi	3	300	987654321

Trying to add money to an existing user: SUCCESS!

3. Trying to check balance of existing user!

Run Admin Functions→Select 3→Enter Customer-ID

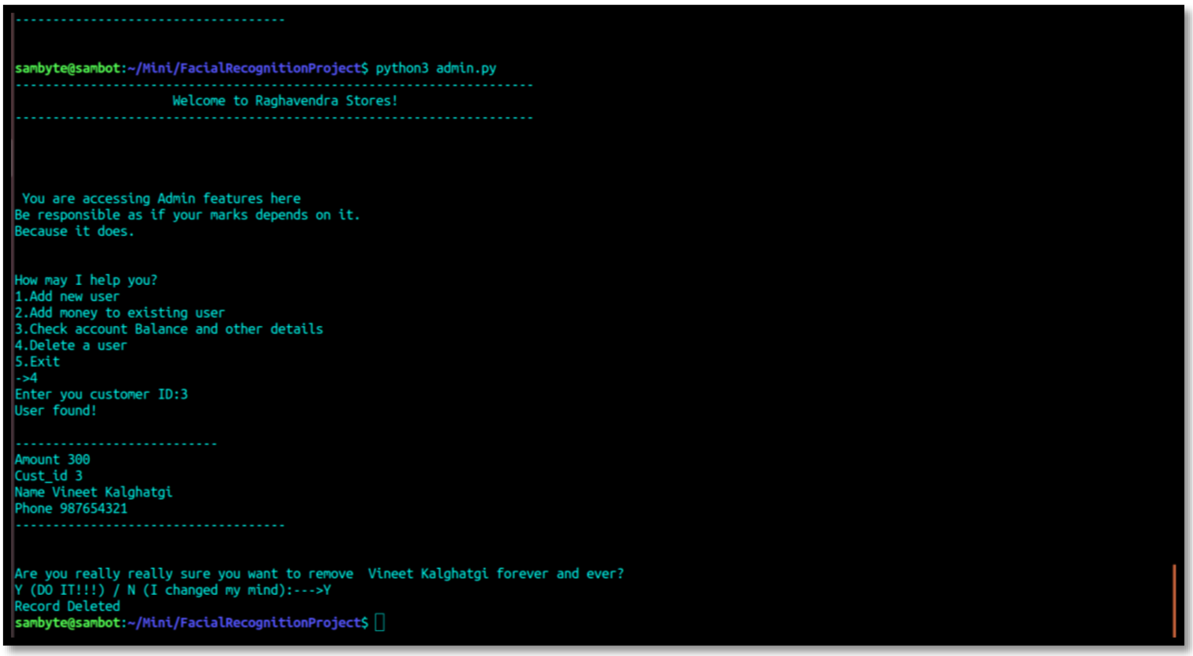


```
Current Amount : Rs. 300
sambyte@sambot:~/Mini/FacialRecognitionProject$ python3 admin.py
----------------------------------------------------------------------
                     Welcome to Raghavendra Stores!
----------------------------------------------------------------------



 You are accessing Admin features here
Be responsible as if your marks depends on it.
Because it does.


How may I help you?
1.Add new user
2.Add money to existing user
3.Check account Balance and other details
4.Delete a user
5.Exit
->3
Enter you customer ID:2
User found!

---------------------------
Amount 800
Cust_id 2
Name R P Raghava
Phone 1234567890
------------------------------------


sambyte@sambot:~/Mini/FacialRecognitionProject$ 
```

Trying to check money of an existing user: SUCCESS!

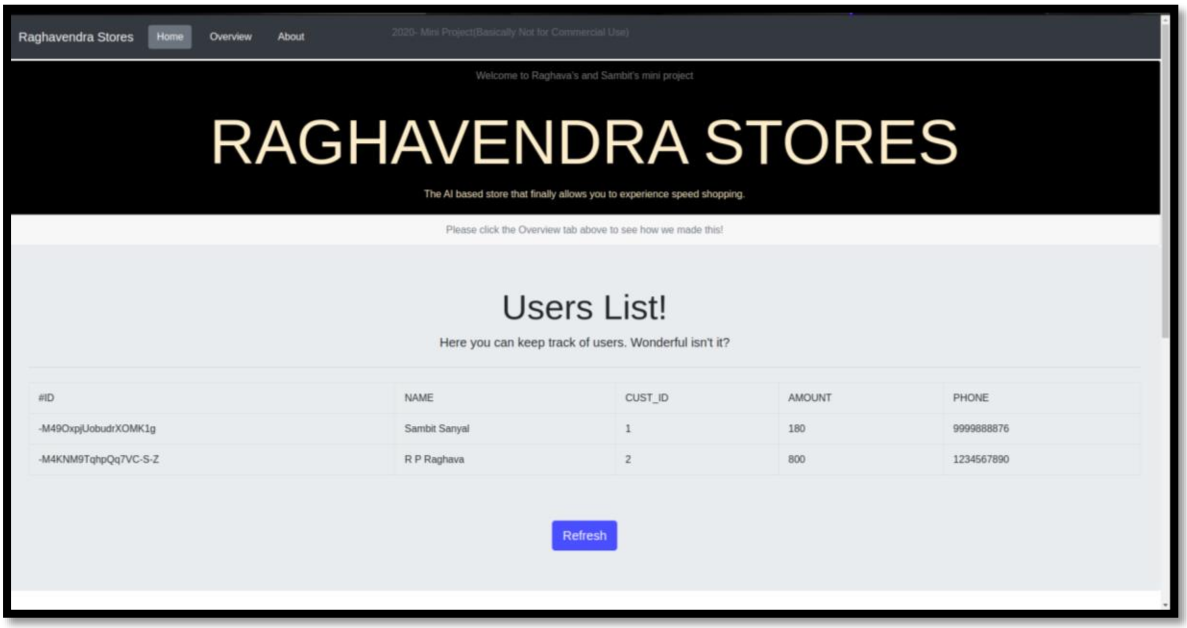
4. Trying to delete an user!

Run Admin Functions→Select 4→Enter Customer-ID→Confirm



```
------------------------------------

sambyte@sambot:~/Mini/FacialRecognitionProject$ python3 admin.py
----------------------------------------------------------------------
                     Welcome to Raghavendra Stores!
----------------------------------------------------------------------



 You are accessing Admin features here
Be responsible as if your marks depends on it.
Because it does.


How may I help you?
1.Add new user
2.Add money to existing user
3.Check account Balance and other details
4.Delete a user
5.Exit
->4
Enter you customer ID:3
User found!

---------------------------
Amount 300
Cust_id 3
Name Vineet Kalghatgi
Phone 987654321
------------------------------------


Are you really really sure you want to remove  Vineet Kalghatgi forever and ever?
Y (DO IT!!!) / N (I changed my mind):--->Y
Record Deleted
sambyte@sambot:~/Mini/FacialRecognitionProject$
```

Confirm with the Website:



Welcome to Raghava's and Sambit's mini project

RAGHAVENDRA STORES

The AI based store that finally allows you to experience speed shopping.

Please click the Overview tab above to see how we made this!

Users List!

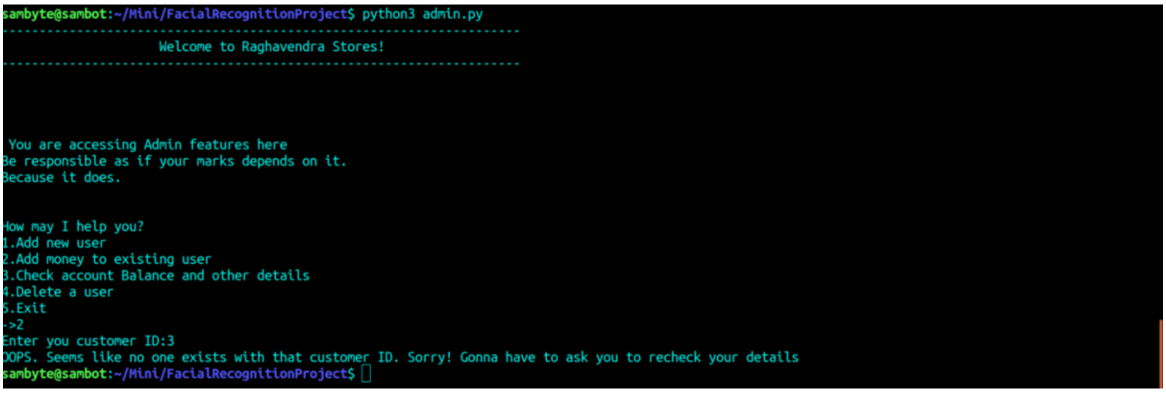
Here you can keep track of users. Wonderful isn't it?

#ID	NAME	CUST_ID	AMOUNT	PHONE
-M49OxpjUobudrXOMK1g	Sambit Sanyal	1	180	9999888876
-M4KNM9TqhpQq7VC-S-Z	R P Raghava	2	800	1234567890

Trying to delete an existing user: SUCCESS!

5. Trying to add money to deleted user!

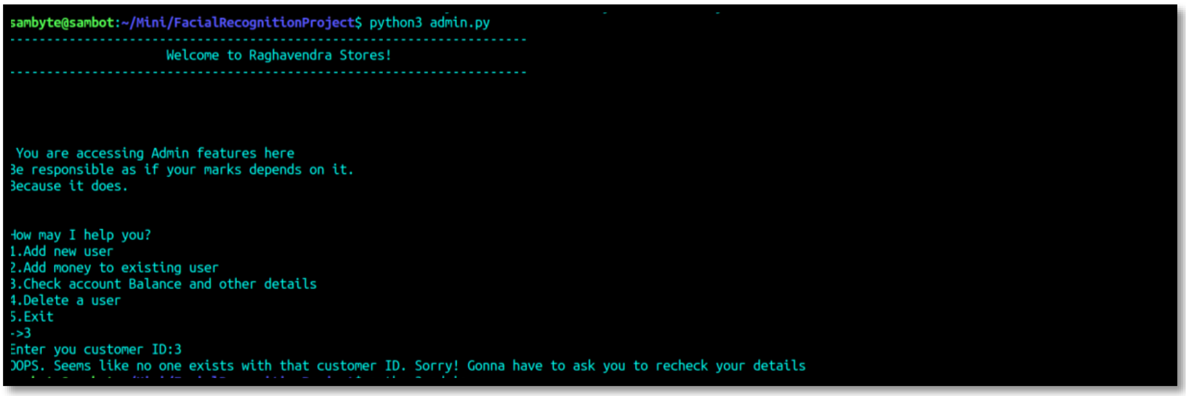
We try to add money to the deleted customer



```
sambyte@sambot:~/Mini/FacialRecognitionProject$ python3 admin.py
--------------------------------------------------------------------
                    Welcome to Raghavendra Stores!
--------------------------------------------------------------------



 You are accessing Admin features here
Be responsible as if your marks depends on it.
Because it does.


How may I help you?
1.Add new user
2.Add money to existing user
3.Check account Balance and other details
4.Delete a user
5.Exit
->2
Enter you customer ID:3
OOPS. Seems like no one exists with that customer ID. Sorry! Gonna have to ask you to recheck your details
sambyte@sambot:~/Mini/FacialRecognitionProject$ 
```

Trying to add money to a deleted user: EXCEPTION HANDLED!

6. Trying to check balance of a deleted user!

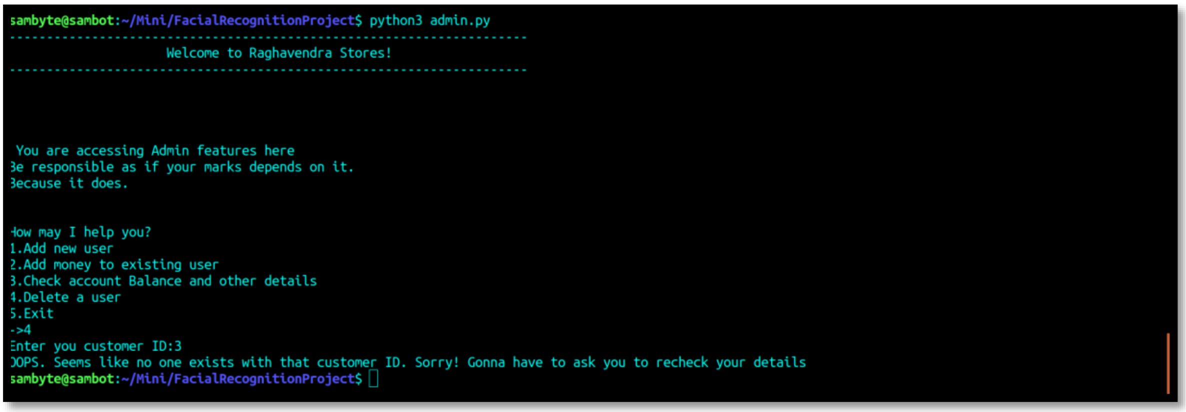


```
sambyte@sambot:~/Mini/FacialRecognitionProject$ python3 admin.py
--------------------------------------------------------------------
                    Welcome to Raghavendra Stores!
--------------------------------------------------------------------



 You are accessing Admin features here
3e responsible as if your marks depends on it.
3ecause it does.


How may I help you?
1.Add new user
2.Add money to existing user
3.Check account Balance and other details
4.Delete a user
5.Exit
->3
Enter you customer ID:3
OOPS. Seems like no one exists with that customer ID. Sorry! Gonna have to ask you to recheck your details
```

Trying to check balance of a deleted user: EXCEPTION HANDLED!

7. Trying to delete an already deleted user!

We try to delete the already deleted user

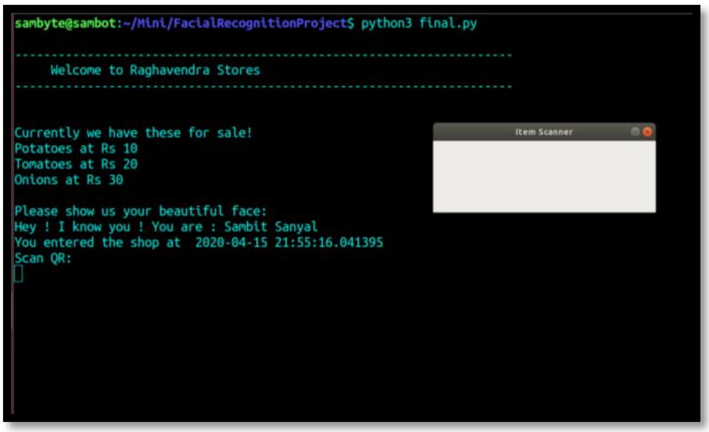


```
sambyte@sambot:~/Mini/FacialRecognitionProject$ python3 admin.py
--------------------------------------------------------------------
                    Welcome to Raghavendra Stores!
--------------------------------------------------------------------



 You are accessing Admin features here
3e responsible as if your marks depends on it.
3ecause it does.


How may I help you?
1.Add new user
2.Add money to existing user
3.Check account Balance and other details
4.Delete a user
5.Exit
->4
Enter you customer ID:3
OOPS. Seems like no one exists with that customer ID. Sorry! Gonna have to ask you to recheck your details
sambyte@sambot:~/Mini/FacialRecognitionProject$ 
```

Trying to delete an already deleted user: EXCEPTION HANDLED!

8. Trying to get the store to recognize a registered face!

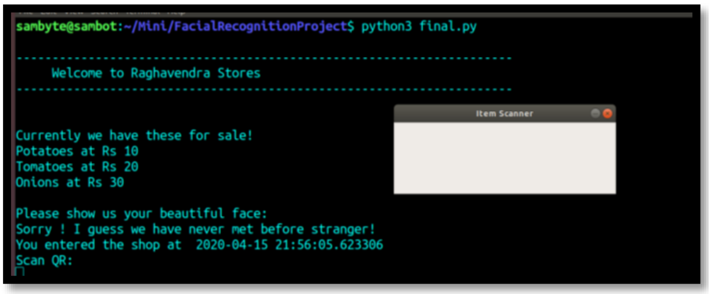
We activate the store and get it to recognize the registered face:



```
sambyte@sambot:~/Mini/FacialRecognitionProject$ python3 final.py

----------------------------------------------------------------------
     Welcome to Raghavendra Stores
----------------------------------------------------------------------

Currently we have these for sale!
Potatoes at Rs 10
Tomatoes at Rs 20
Onions at Rs 30

Please show us your beautiful face:
Hey ! I know you ! You are : Sambit Sanyal
You entered the shop at  2020-04-15 21:55:16.041395
Scan QR:
```

Trying to recognize a registered user: SUCCESS!

9. Trying to get store to recognize an unregistered face!

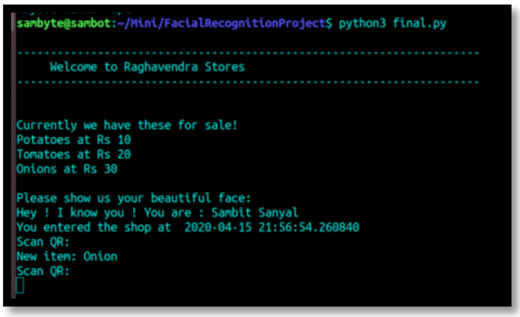


```
sambyte@sambot:~/Mini/FacialRecognitionProject$ python3 final.py

----------------------------------------------------------------------
     Welcome to Raghavendra Stores
----------------------------------------------------------------------

Currently we have these for sale!
Potatoes at Rs 10
Tomatoes at Rs 20
Onions at Rs 30

Please show us your beautiful face:
Sorry ! I guess we have never met before stranger!
You entered the shop at  2020-04-15 21:56:05.623306
Scan QR:
```

Trying to recognize an unregistered user: SUCCESS!

10.Trying to get store to accept QR codes!

Attempt to add an Onion

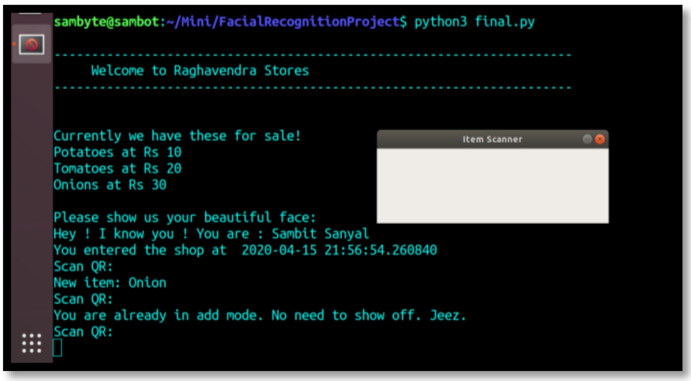


```
sambyte@sambot:~/Mini/FacialRecognitionProject$ python3 final.py

----------------------------------------------------------------------
     Welcome to Raghavendra Stores
----------------------------------------------------------------------

Currently we have these for sale!
Potatoes at Rs 10
Tomatoes at Rs 20
Onions at Rs 30

Please show us your beautiful face:
Hey ! I know you ! You are : Sambit Sanyal
You entered the shop at  2020-04-15 21:56:54.260840
Scan QR:
New item: Onion
Scan QR:
```

Trying to add an item to the cart: SUCCESS!

11.Trying to Change to add mode while in Add mode!

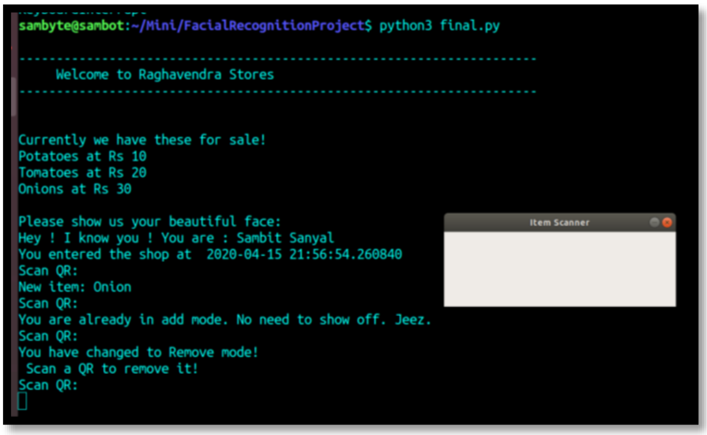
We show the “add” QR code while already in add mode



```
sambyte@sambot:~/Mini/FacialRecognitionProject$ python3 final.py

--------------------------------------------------------------------
     Welcome to Raghavendra Stores
--------------------------------------------------------------------

Currently we have these for sale!
Potatoes at Rs 10
Tomatoes at Rs 20
Onions at Rs 30

Please show us your beautiful face:
Hey ! I know you ! You are : Sambit Sanyal
You entered the shop at  2020-04-15 21:56:54.260840
Scan QR:
New item: Onion
Scan QR:
You are already in add mode. No need to show off. Jeez.
Scan QR:
```

Trying to change into add mode while already in add mode: EXCEPTION HANDLED!

12.Trying to change to remove mode!

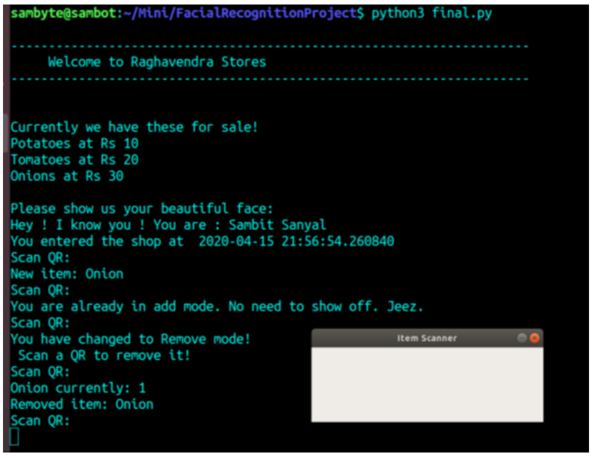


```
sambyte@sambot:~/Mini/FacialRecognitionProject$ python3 final.py

--------------------------------------------------------------------
     Welcome to Raghavendra Stores
--------------------------------------------------------------------

Currently we have these for sale!
Potatoes at Rs 10
Tomatoes at Rs 20
Onions at Rs 30

Please show us your beautiful face:
Hey ! I know you ! You are : Sambit Sanyal
You entered the shop at  2020-04-15 21:56:54.260840
Scan QR:
New item: Onion
Scan QR:
You are already in add mode. No need to show off. Jeez.
Scan QR:
You have changed to Remove mode!
 Scan a QR to remove it!
Scan QR:
```

Trying to change into remove mode while in add mode: SUCCESS!

13.Trying to remove an added item

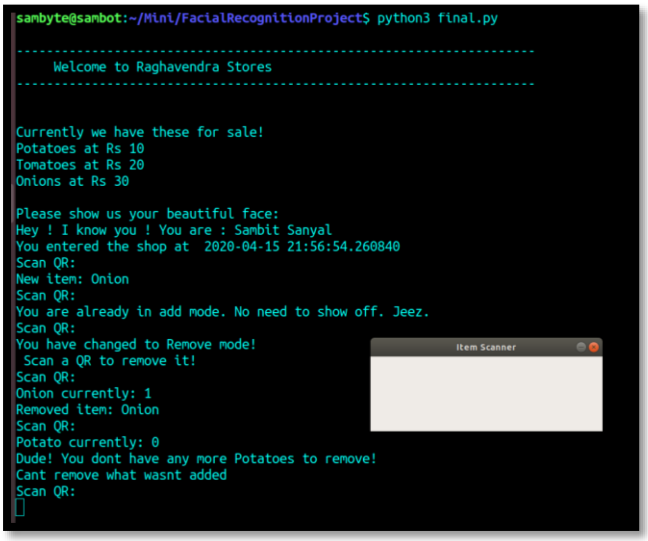
First we have to activate the remove mode! And then try to remove the item using the same QR code we scanned it with.



```
sambyte@sambot:~/Mini/FacialRecognitionProject$ python3 final.py

----------------------------------------------------------------------
     Welcome to Raghavendra Stores
----------------------------------------------------------------------


Currently we have these for sale!
Potatoes at Rs 10
Tomatoes at Rs 20
Onions at Rs 30

Please show us your beautiful face:
Hey ! I know you ! You are : Sambit Sanyal
You entered the shop at  2020-04-15 21:56:54.260840
Scan QR:
New item: Onion
Scan QR:
You are already in add mode. No need to show off. Jeez.
Scan QR:
You have changed to Remove mode!
 Scan a QR to remove it!
Scan QR:
Onion currently: 1
Removed item: Onion
Scan QR:
```

Trying to remove an item in Remove mode: SUCCESS!

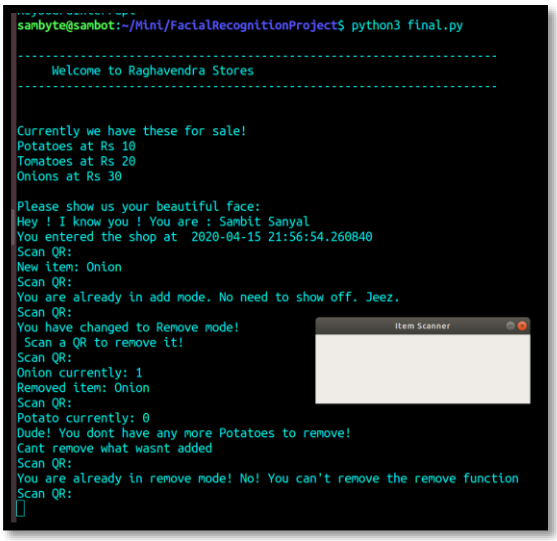
14.Trying to remove something that isn’t in the basket:



```
sambyte@sambot:~/Mini/FacialRecognitionProject$ python3 final.py

----------------------------------------------------------------------
     Welcome to Raghavendra Stores
----------------------------------------------------------------------


Currently we have these for sale!
Potatoes at Rs 10
Tomatoes at Rs 20
Onions at Rs 30

Please show us your beautiful face:
Hey ! I know you ! You are : Sambit Sanyal
You entered the shop at  2020-04-15 21:56:54.260840
Scan QR:
New item: Onion
Scan QR:
You are already in add mode. No need to show off. Jeez.
Scan QR:
You have changed to Remove mode!
 Scan a QR to remove it!
Scan QR:
Onion currently: 1
Removed item: Onion
Scan QR:
Potato currently: 0
Dude! You dont have any more Potatoes to remove!
Cant remove what wasnt added
Scan QR:
```

Trying to remove an item that isn’t in the basket: EXCEPTION HANDLED!

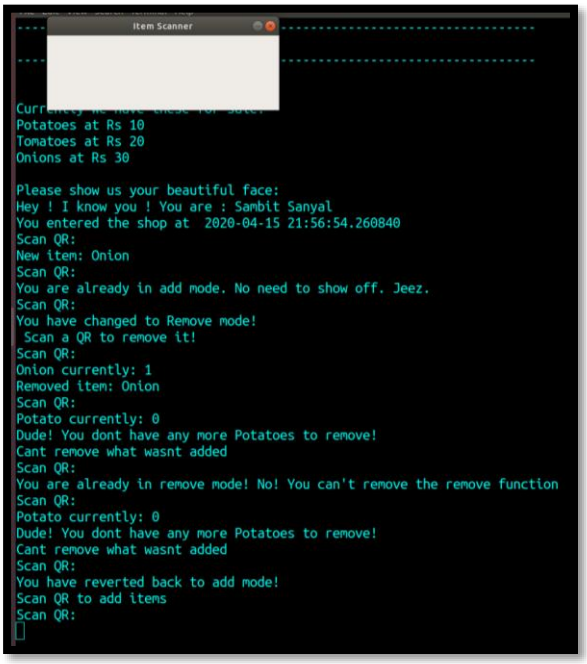
15. Trying to change into remove mode while already being in remove mode



```
sambyte@sambot:~/Mini/FacialRecognitionProject$ python3 final.py

-----------------------------------------------------------------
     Welcome to Raghavendra Stores
-----------------------------------------------------------------


Currently we have these for sale!
Potatoes at Rs 10
Tomatoes at Rs 20
Onions at Rs 30

Please show us your beautiful face:
Hey ! I know you ! You are : Sambit Sanyal
You entered the shop at  2020-04-15 21:56:54.260840
Scan QR:
New item: Onion
Scan QR:
You are already in add mode. No need to show off. Jeez.
Scan QR:
You have changed to Remove mode!
 Scan a QR to remove it!
Scan QR:
Onion currently: 1
Removed item: Onion
Scan QR:
Potato currently: 0
Dude! You dont have any more Potatoes to remove!
Cant remove what wasnt added
Scan QR:
You are already in remove mode! No! You can't remove the remove function
Scan QR:
```

Trying to change into remove mode while already in remove mode: EXCEPTION HANDLED!

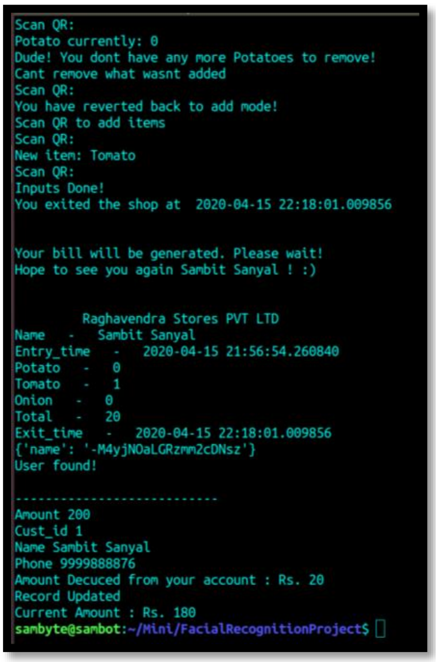
16.Trying to revert back to add mode once in remove mode



```
----                                  ----------------------------------

----                                  ----------------------------------


Currently we have these for sale!
Potatoes at Rs 10
Tomatoes at Rs 20
Onions at Rs 30

Please show us your beautiful face:
Hey ! I know you ! You are : Sambit Sanyal
You entered the shop at  2020-04-15 21:56:54.260840
Scan QR:
New item: Onion
Scan QR:
You are already in add mode. No need to show off. Jeez.
Scan QR:
You have changed to Remove mode!
 Scan a QR to remove it!
Scan QR:
Onion currently: 1
Removed item: Onion
Scan QR:
Potato currently: 0
Dude! You dont have any more Potatoes to remove!
Cant remove what wasnt added
Scan QR:
You are already in remove mode! No! You can't remove the remove function
Scan QR:
Potato currently: 0
Dude! You dont have any more Potatoes to remove!
Cant remove what wasnt added
Scan QR:
You have reverted back to add mode!
Scan QR to add items
Scan QR:
```

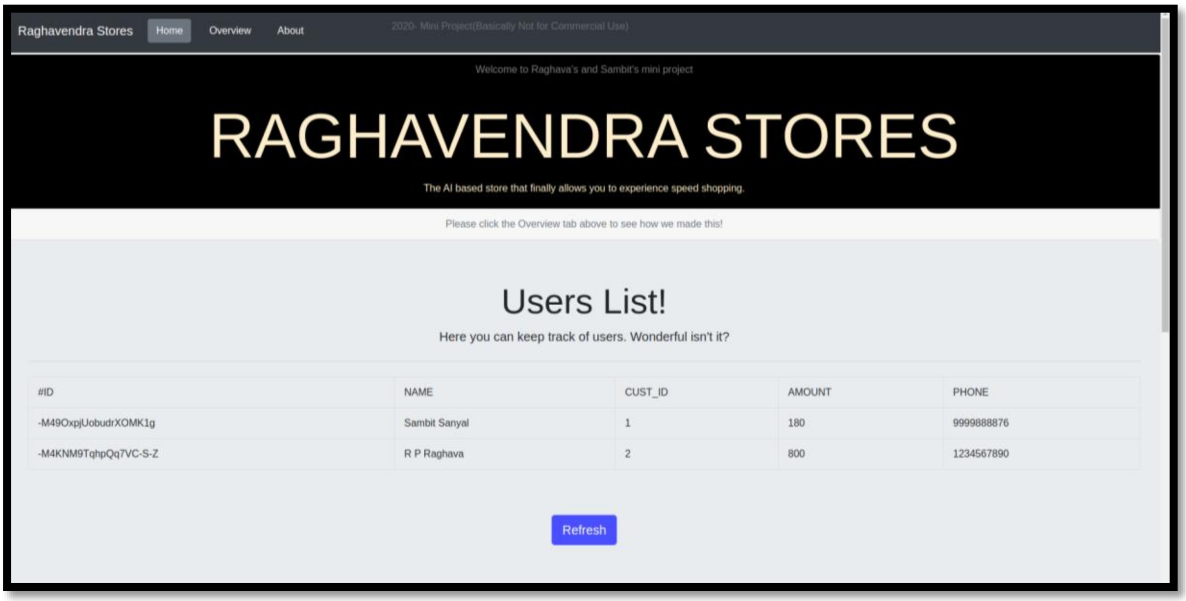
Trying change to add mode from remove mode: SUCCESS!

17.Trying to generate a bill for a registered customer!



```
Scan QR:
Potato currently: 0
Dude! You dont have any more Potatoes to remove!
Cant remove what wasnt added
Scan QR:
You have reverted back to add mode!
Scan QR to add items
Scan QR:
New item: Tomato
Scan QR:
Inputs Done!
You exited the shop at  2020-04-15 22:18:01.009856

Your bill will be generated. Please wait!
Hope to see you again Sambit Sanyal ! :)

         Raghavendra Stores PVT LTD
Name   -   Sambit Sanyal
Entry_time   -   2020-04-15 21:56:54.260840
Potato   -   0
Tomato   -   1
Onion   -   0
Total   -   20
Exit_time   -   2020-04-15 22:18:01.009856
{'name': '-M4yjNOaLGRzmm2cDNsz'}
User found!

---------------------------
Amount 200
Cust_id 1
Name Sambit Sanyal
Phone 9999888876
Amount Decuced from your account : Rs. 20
Record Updated
Current Amount : Rs. 180
sambyte@sambot:~/Mini/FacialRecognitionProject$
```

We then check the customer details to see reduced balance



Raghavendra Stores
Home
Overview
About
2020- Mini Project(Basically Not for Commercial Use)
Welcome to Raghava's and Sambit's mini project

RAGHAVENDRA STORES

The AI based store that finally allows you to experience speed shopping.

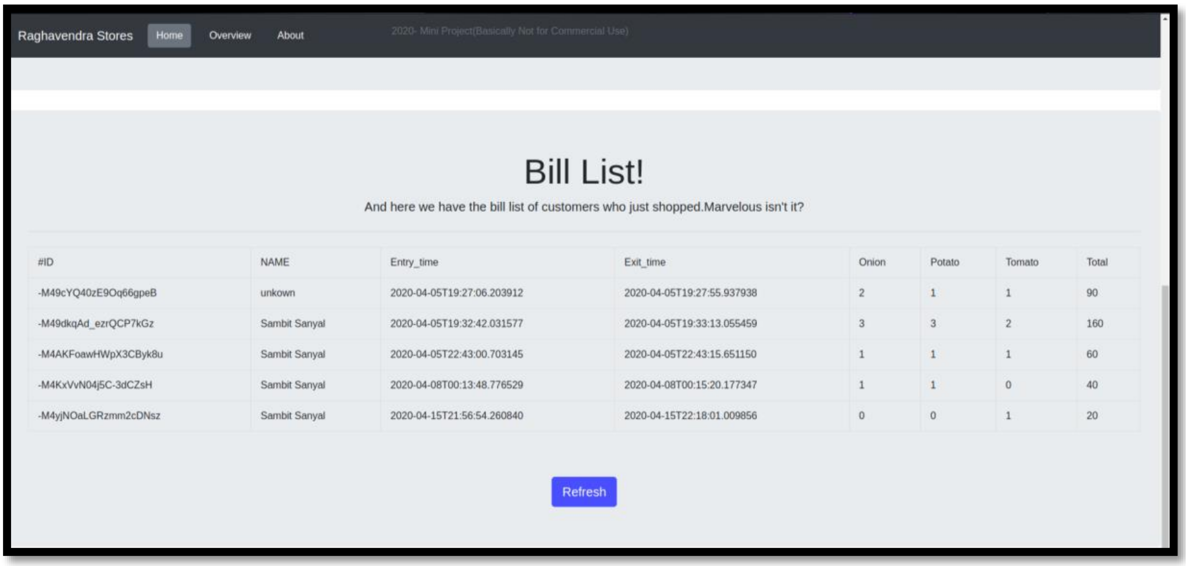
Please click the Overview tab above to see how we made this!

Users List!

Here you can keep track of users. Wonderful isn't it?

#ID	NAME	CUST_ID	AMOUNT	PHONE
-M49OxpjUobudrXOMK1g	Sambit Sanyal	1	180	9999888876
-M4KNM9TqhpQq7VC-S-Z	R P Raghava	2	800	1234567890

We then see the generated bill.



Bill List!

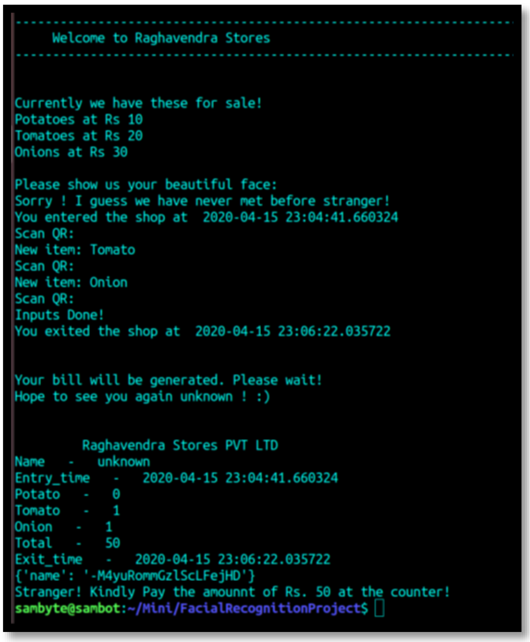
And here we have the bill list of customers who just shopped.Marvelous isn't it?

#ID	NAME	Entry_time	Exit_time	Onion	Potato	Tomato	Total
-M49cYQ40zE9Oq66gpeB	unkown	2020-04-05T19:27:06.203912	2020-04-05T19:27:55.937938	2	1	1	90
-M49dkqAd_ezrQCP7kGz	Sambit Sanyal	2020-04-05T19:32:42.031577	2020-04-05T19:33:13.055459	3	3	2	160
-M4AKFoawHWpX3CByk8u	Sambit Sanyal	2020-04-05T22:43:00.703145	2020-04-05T22:43:15.651150	1	1	1	60
-M4KxVvN04j5C-3dCZsH	Sambit Sanyal	2020-04-08T00:13:48.776529	2020-04-08T00:15:20.177347	1	1	0	40
-M4yjNOaLGRzmm2cDNsz	Sambit Sanyal	2020-04-15T21:56:54.260840	2020-04-15T22:18:01.009856	0	0	1	20

Trying to generate bill for registered customer: SUCCESS!

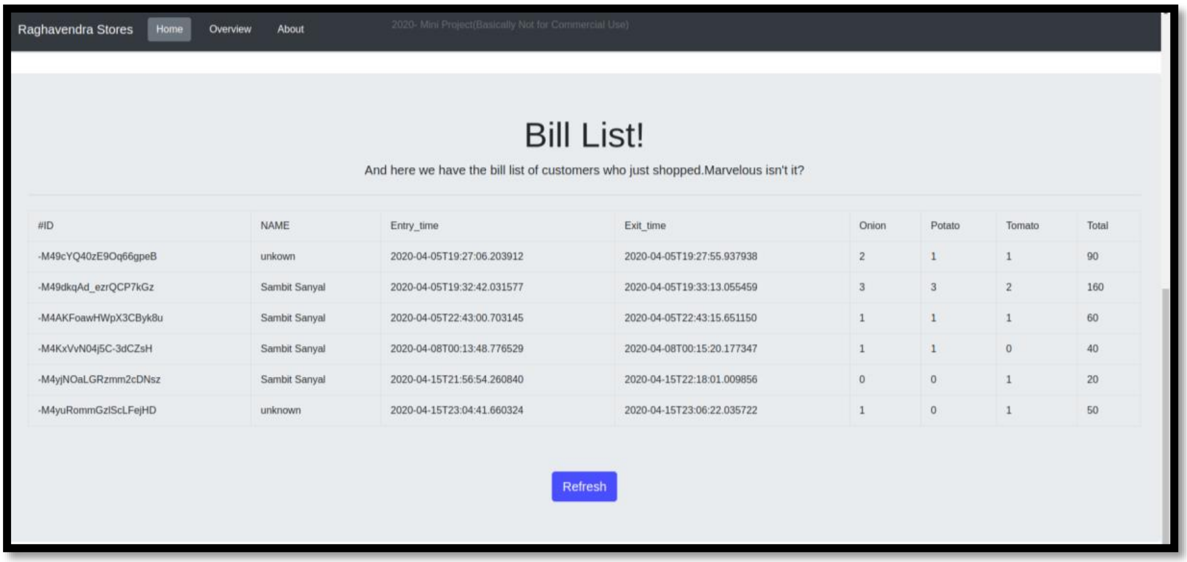
18.<u>Trying to generate bill for unregistered customer!</u>

We see that the customer is requested to manually pay the amount



```
-----------------------------------------------------------------
     Welcome to Raghavendra Stores
-----------------------------------------------------------------


Currently we have these for sale!
Potatoes at Rs 10
Tomatoes at Rs 20
Onions at Rs 30

Please show us your beautiful face:
Sorry ! I guess we have never met before stranger!
You entered the shop at  2020-04-15 23:04:41.660324
Scan QR:
New item: Tomato
Scan QR:
New item: Onion
Scan QR:
Inputs Done!
You exited the shop at  2020-04-15 23:06:22.035722


Your bill will be generated. Please wait!
Hope to see you again unknown ! :)


         Raghavendra Stores PVT LTD
Name    -   unknown
Entry_time    -   2020-04-15 23:04:41.660324
Potato    -   0
Tomato    -   1
Onion    -   1
Total    -   50
Exit_time    -   2020-04-15 23:06:22.035722
{'name': '-M4yuRommGzlScLFejHD'}
Stranger! Kindly Pay the amounnt of Rs. 50 at the counter!
sambyte@sambot:~/Mini/FacialRecognitionProject$
```

The bill is automatically generated as “Unknown” for customer name



Bill List!

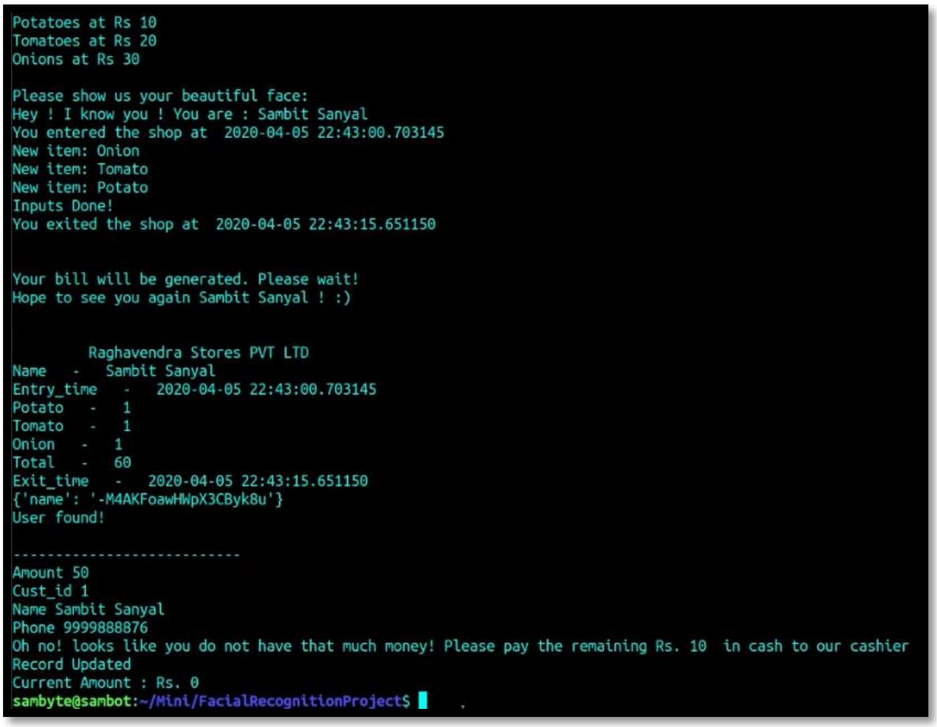
And here we have the bill list of customers who just shopped.Marvelous isn't it?

#ID	NAME	Entry_time	Exit_time	Onion	Potato	Tomato	Total
-M49cYQ40zE9Oq66gpeB	unkown	2020-04-05T19:27:06.203912	2020-04-05T19:27:55.937938	2	1	1	90
-M49dkqAd_ezrQCP7kGz	Sambit Sanyal	2020-04-05T19:32:42.031577	2020-04-05T19:33:13.055459	3	3	2	160
-M4AKFoawHWpX3CByk8u	Sambit Sanyal	2020-04-05T22:43:00.703145	2020-04-05T22:43:15.651150	1	1	1	60
-M4KxVvN04j5C-3dCZsH	Sambit Sanyal	2020-04-08T00:13:48.776529	2020-04-08T00:15:20.177347	1	1	0	40
-M4yjNOaLGRzmm2cDNsz	Sambit Sanyal	2020-04-15T21:56:54.260840	2020-04-15T22:18:01.009856	0	0	1	20
-M4yuRommGzlScLFejHD	unknown	2020-04-15T23:04:41.660324	2020-04-15T23:06:22.035722	1	0	1	50

Trying to generate bill for unregistered customer: SUCCESS!

19.Trying to generate bill for registered but short funded customer!

The customer who is unable to pay the amount completely through his/her account has to pay the balance amount in cash



```
Potatoes at Rs 10
Tomatoes at Rs 20
Onions at Rs 30

Please show us your beautiful face:
Hey ! I know you ! You are : Sambit Sanyal
You entered the shop at  2020-04-05 22:43:00.703145
New item: Onion
New item: Tomato
New item: Potato
Inputs Done!
You exited the shop at  2020-04-05 22:43:15.651150


Your bill will be generated. Please wait!
Hope to see you again Sambit Sanyal ! :)


         Raghavendra Stores PVT LTD
Name   -   Sambit Sanyal
Entry_time   -   2020-04-05 22:43:00.703145
Potato   -   1
Tomato   -   1
Onion   -   1
Total   -   60
Exit_time   -   2020-04-05 22:43:15.651150
{'name': '-M4AKFoawHWpX3CByk8u'}
User found!

---------------------------
Amount 50
Cust_id 1
Name Sambit Sanyal
Phone 9999888876
Oh no! looks like you do not have that much money! Please pay the remaining Rs. 10  in cash to our cashier
Record Updated
Current Amount : Rs. 0
sambyte@sambot:~/Mini/FacialRecognitionProject$ 
```

Trying to generate bill for registered customer but who is short on funds: SUCCESS!

CONCLUSION

Automated Retail Store using Face Recognition is a faster, secure system. It is a novel solution to improve efficiency and productivity in the retail sector. This system will help brick and mortar stores to give competition e-commerce. This system will give secure payment system which is better than the present day payments in terms of security. Automated Retail Store also considers the edge cases and works well in all conditions, such as when the person is not registered or has insufficient balance.

It also give exact billing details using the QR codes and best part is it takes way less time than any retail store that is present today. This system is very simple and intuitive to use and anyone can use this robust system.

BIBLIOGRAPHY

- Paul Viola, Michael Jones “Rapid Object Detection using a Boosted Cascade of Simple Features”
- Face Detection using Haar Cascades by Alexander Mordvintsev & Abid K
- https://firebase.google.com/docs
- https://docs.python.org/3/tutorial/
- “QR Code Analysis” by Sangeeta Singh
- “Facial Recognition using OpenCV” by Shervin EMAMI1, Valentin Petruț SUCIU