Project Submission Report

Project Title: Automatic annotations and Face detection For Ticket Booking

Student Details:

Name: Vardaan Shukla

Roll Number: E23MCAG0021

Class/Section: MCA BATCH II

Email: E23MCAG0021@bennett.edu.in

Contact Number: 9868768833

Project Overview

The project discusses face detection and annotation from cameras in computer systems. This application harnesses the power of computer vision and OpenCV to create a streamlined process of Airport check-in. In an increasingly touchless and technology-driven world, the need for a seamless check-in process is required to eliminate the long queues at airports. This report addresses this need by enabling users to upload facial data on the website that can be used to speed up the process during arrival.

Introduction

In an era where time is paramount and efficiency reigns supreme, the airport experience often stands as a bottleneck to seamless travel. The increasing queues, and convoluted check-in process- represent the frustration that has long affected air travelers. To redefine the very essence of airport functionality, infusing efficiency and convenience into every facet of the traveler's journey. "Jet Set Go" aims to emerge as a catalyst of change amid the ongoing hustle and bustle of airports, a visionary project poised to untangle the complexities of airport procedures. This system, powered by the prowess of artificial intelligence, stands as the center of attention of the seamless travel experience. Bid farewell to the tedious queues and long waiting, by simply uploading facial data through an intuitive application or website, passengers become part of the streamlined process that transcends the limitations of traditional check-in protocol.

Literature Review

Our project is inspired by the Emirates virtual tunnel concept and their underdevelopment facial recognition technology, along with that we have mentioned other underdevelopment work that is similar to ours.

Delta Biometric Terminal Experience (Atlanta, Airport, USA)

Introduction of facial recognition for check-in, security, and boarding, eliminating the need for physical documents.

Singapore Changi Airport's Facial Recognition Trials (Singapore)

They have been testing facial recognition systems for various processes, including check-in and boarding**.**

Emirates Biometric Path (Dubai International, UAE)

They implemented a Biometric Path using facial recognition for seamless passenger' journeys, starting from check-in to boarding.

British Airways Biometric Boarding (Heathrow Airport, UK)

British Airways conducted trials of Biometric boarding gates at Heathrow Airport, utilizing facial recognition for boarding procedures.

JetBlue and U.S. Customs and Border Protection (Multiple Airports, USA)

JetBlue partnered with CBP to implement facial recognition for boarding, enhancing the boarding, process, and customs clearance for international flights.

Qatar Airways' Digital Travel Document (Hamad International Airport, Qatar)

Qatar Airways tested a digital identity solution using facial recognition for seamless travel experiences.

SITA Smart Path (Various Airports Worldwide)

SITA, a technology company, offers the "Smart Path" solution that integrates biometric technology for smoother airport processes.

KLM's biometric boarding at Schiphol Airport (Amsterdam, Netherlands)

KLM Royal Dutch Airlines introduced biometric boarding gates at Schiphol Airport for a faster and more convenient boarding experience.

Methodology

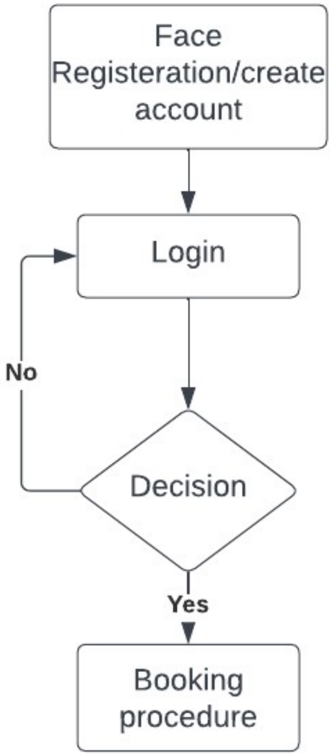
We have used **Python** language as our main pillar for this project build and this is still under development many steps are required to be polished for perfection for step one.

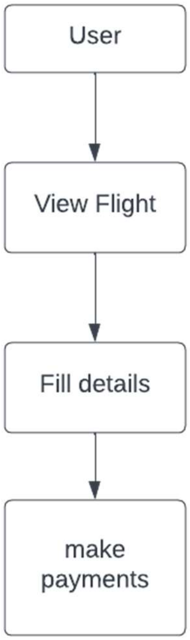
We have used the **OpenCV** library which is a real-time optimized computer-based library with the basic motive of using this is that it will help us to capture the faces of the user. After that, we **created an algorithm so that every single photo of the frame reaches its right place** in this, we use a **face recognition library** to identify the face after the identification phase it will create a **bounding box on the face area and give the coordinates of the face area**.

The next step is to **crop the face area** and store it in the proper place we have used **HaralikTextureExtraction** to extract features from the image. The last and final step is the detection, for now, we are not using YOLO or TensorFlow because of less time with the help of HaralikTextureExtraction we have matched the value that we have extracted and performed the detection process. It is also a reliable process for face detection.

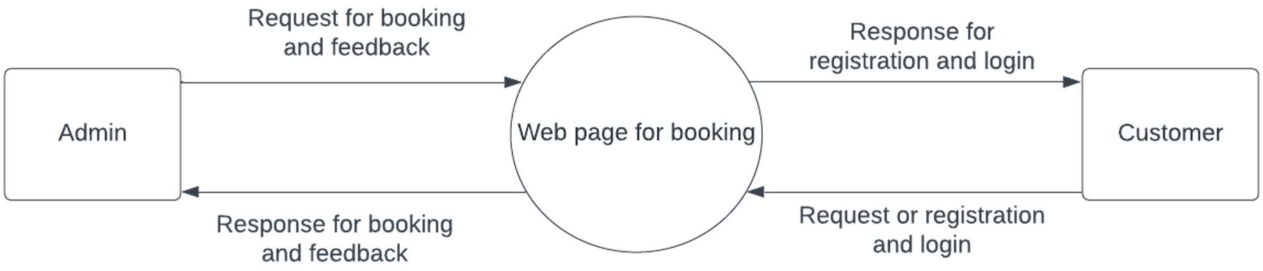
Design

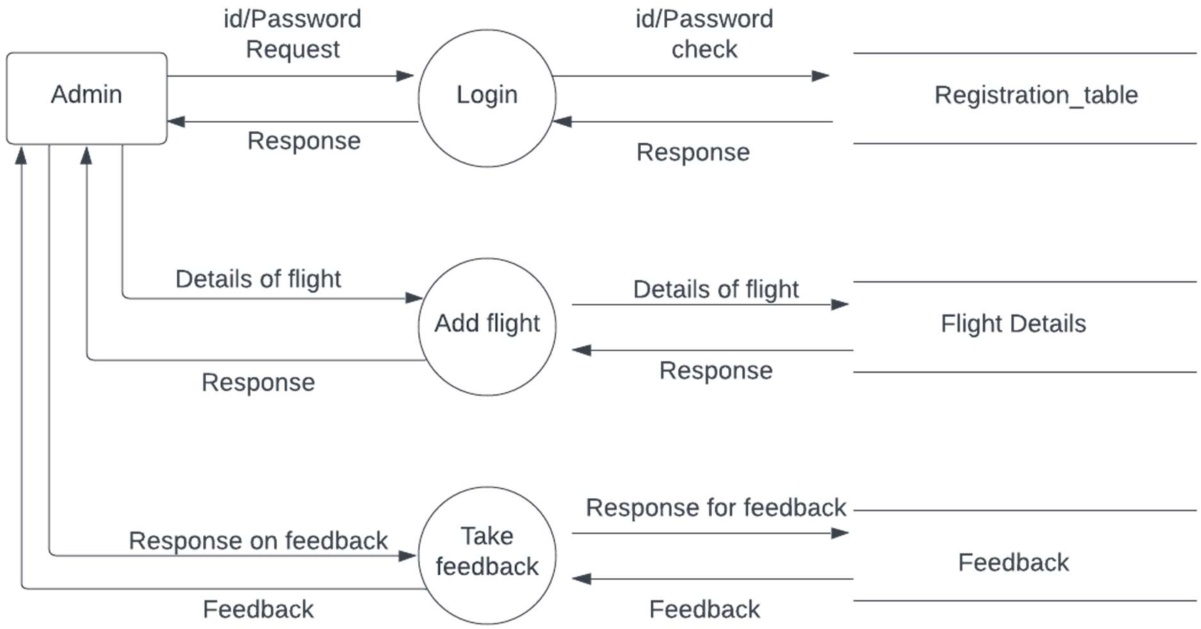
USER SIDE FLOW CHART

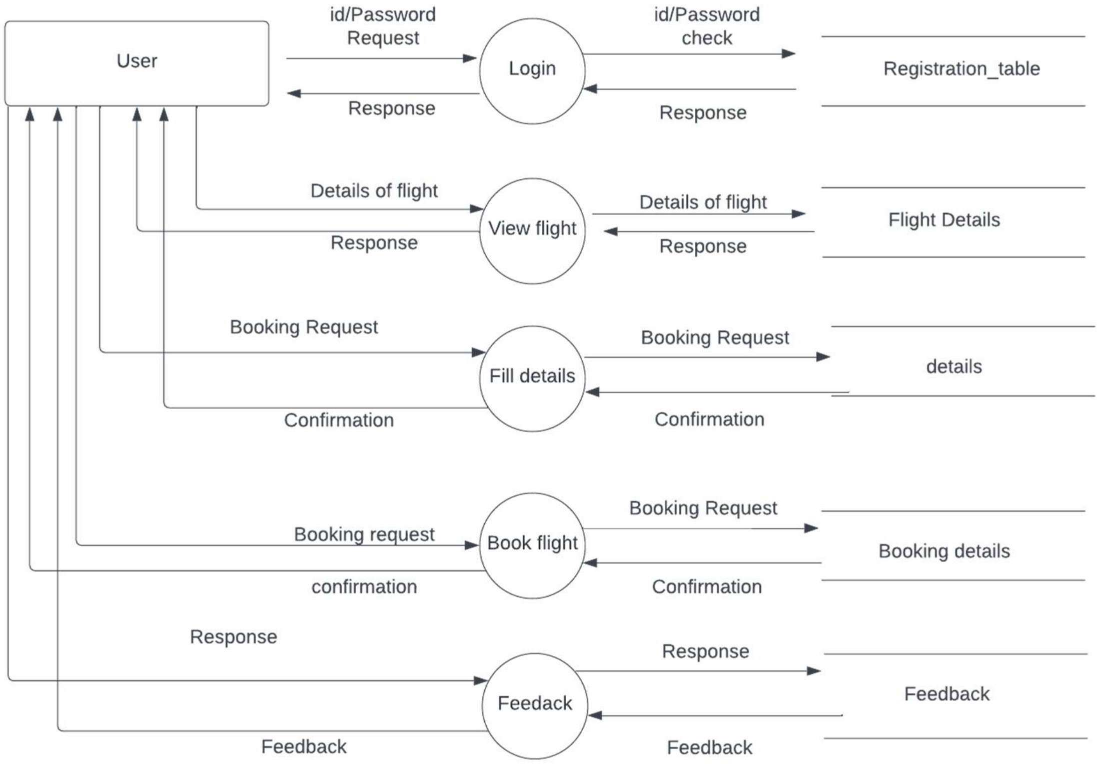




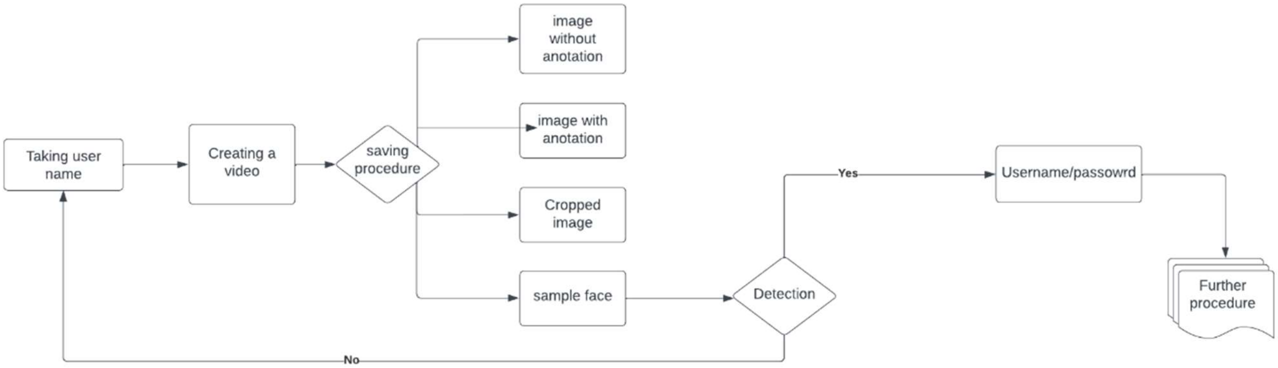
DATA FLOW DIAGRAM


Request for booking and feedback
Response for registration and login
Admin
Web page for booking
Customer
Response for booking and feedback
Request or registration and login

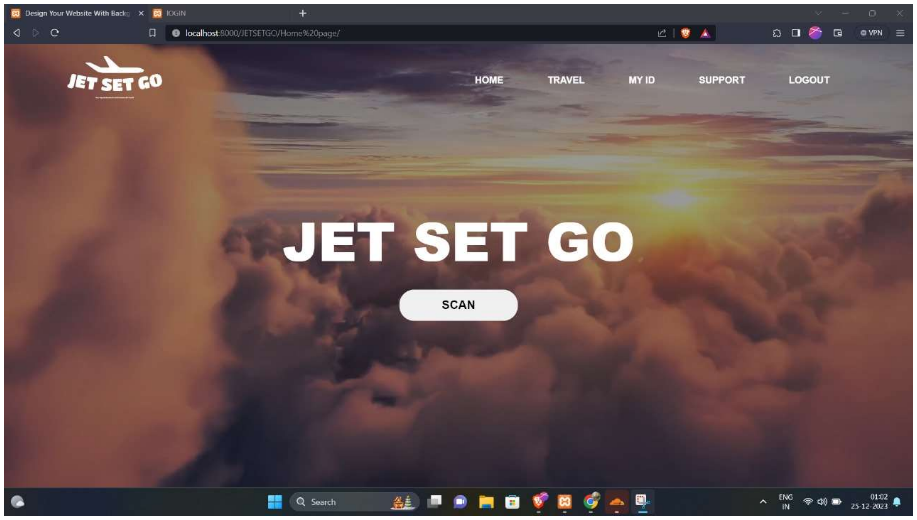

Admin
id/Password Request
Response
Login
id/Password check
Response
Registration_table
Details of flight
Response
Add flight
Details of flight
Response
Flight Details
Response on feedback
Feedback
Take feedback
Response for feedback
Feedback
Feedback

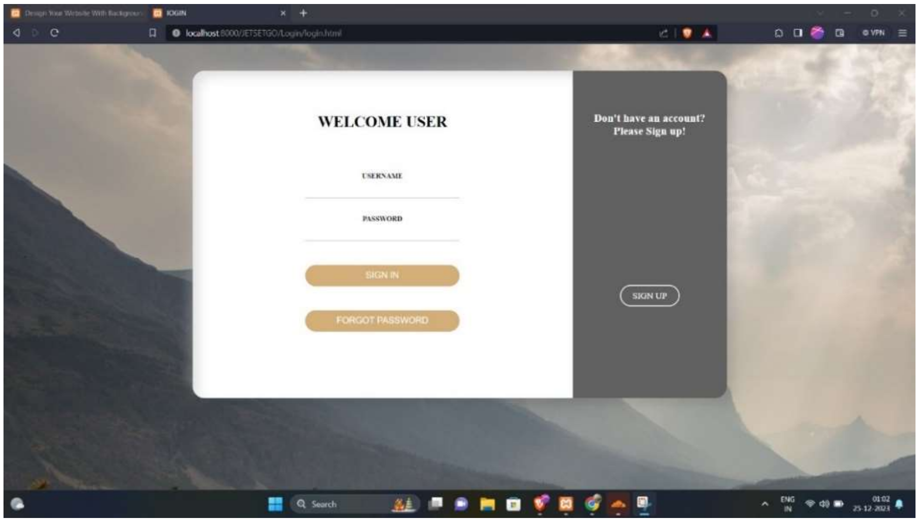

User
id/Password Request
Response
Login
id/Password check
Response
Registration_table
Details of flight
Response
View flight
Details of flight
Response
Flight Details
Booking Request
Confirmation
Fill details
Booking Request
Confirmation
details
Booking request
confirmation
Book flight
Booking Request
Confirmation
Booking details
Response
Feedback
Feedack
Response
Feedback
Feedback

Model workflow

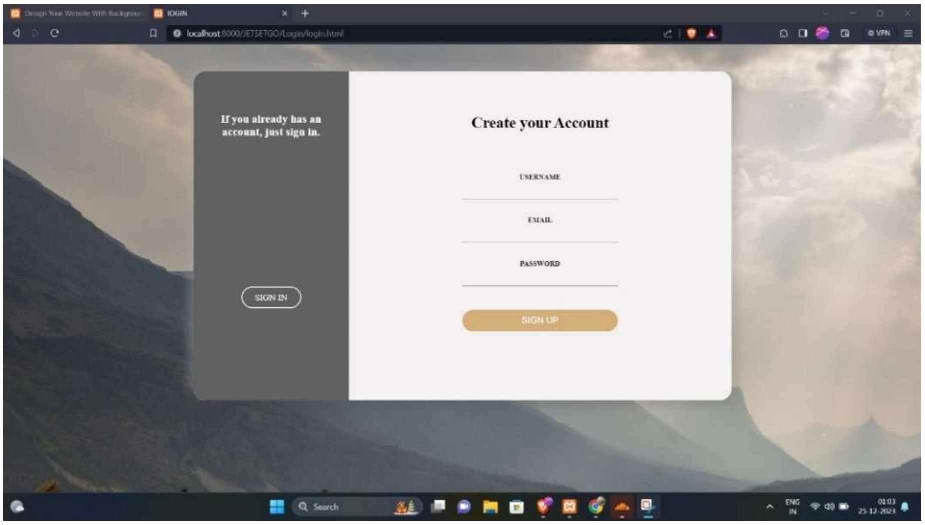


These will be the basic Concepts of UI for the Website





Don't have an account? Please Sign up!

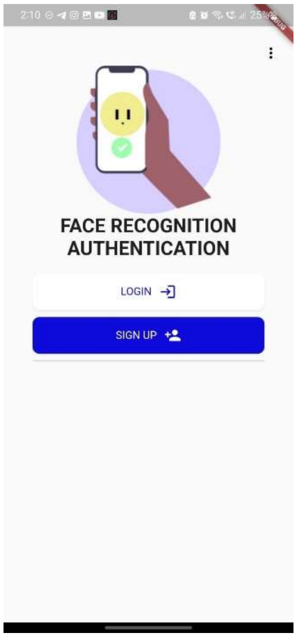


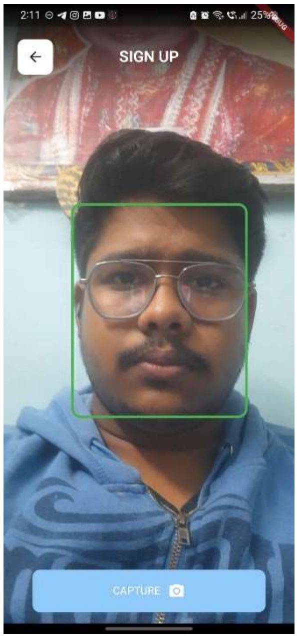
If you already has an account, just sign in.

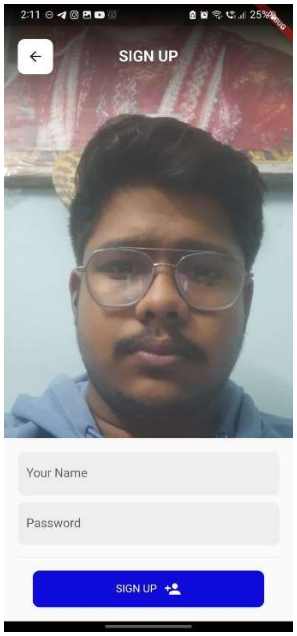
Create your Account

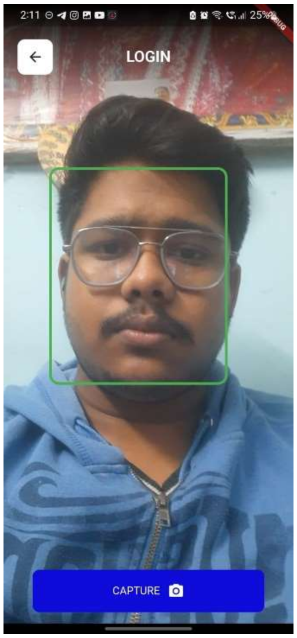
Basic Concept for Android Application

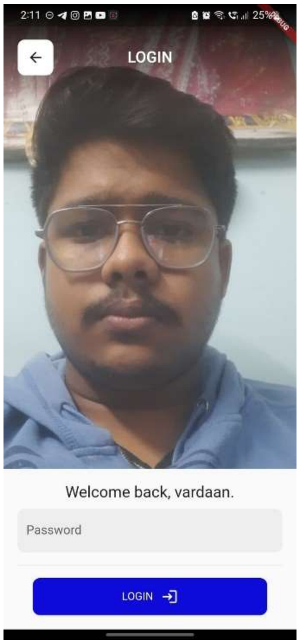
Homepage and Sign-up page:


FACE RECOGNITION AUTHENTICATION

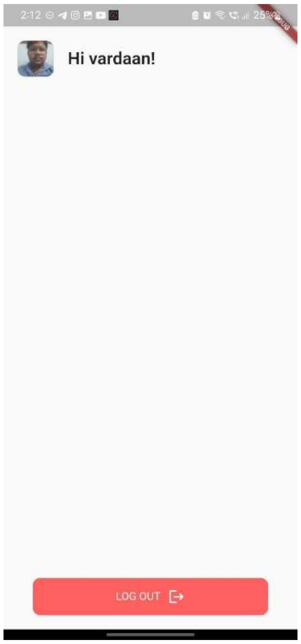






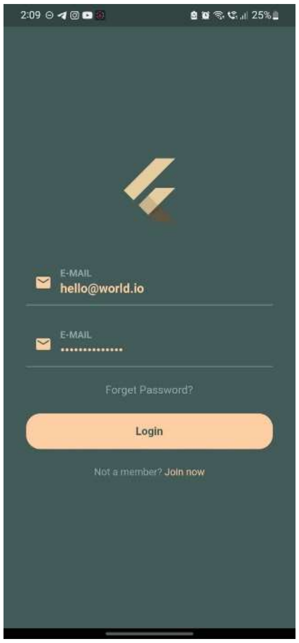


Welcome back, vardaan.



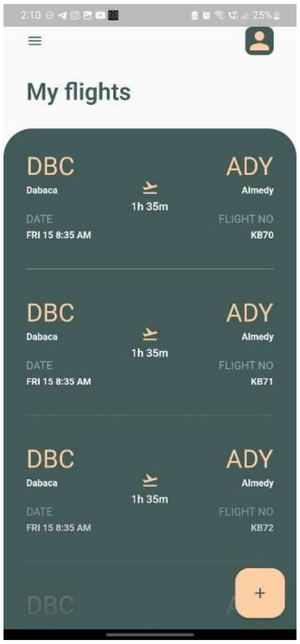
Hi vardaan!

Flight Booking page:



E-MAIL
hello@world.io

E-MAIL
••••••••••••••



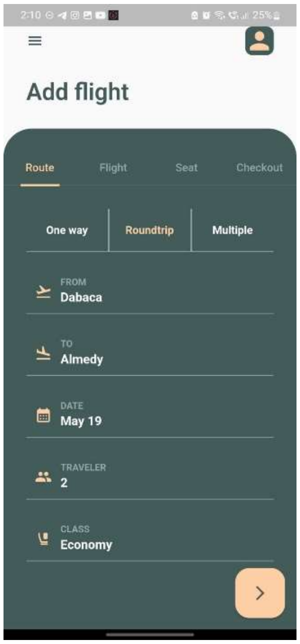
My flights

DBC Dabaca 1h 35m ADY Almedy
DATE FRI 15 8:35 AM FLIGHT NO KB70

DBC Dabaca 1h 35m ADY Almedy
DATE FRI 15 8:35 AM FLIGHT NO KB71

DBC Dabaca 1h 35m ADY Almedy
DATE FRI 15 8:35 AM FLIGHT NO KB72

DBC



Add flight

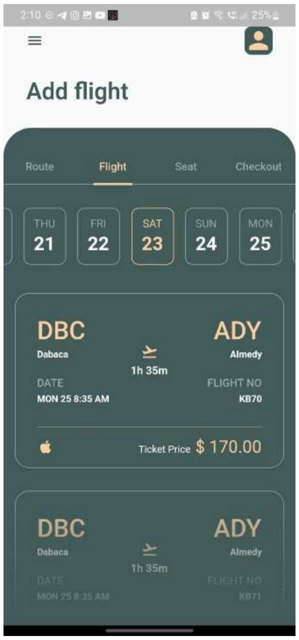
FROM
Dabaca

TO
Almedy

DATE
May 19

TRAVELER
2

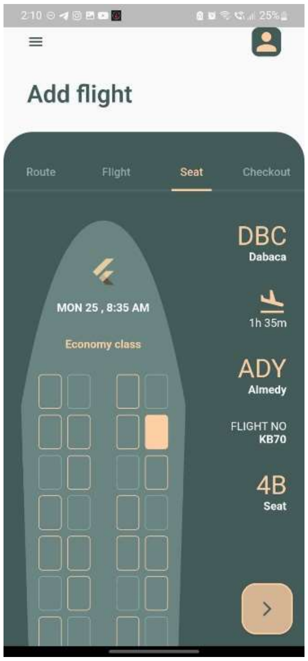
CLASS
Economy



Add flight

DBC Dabaca 1h 35m ADY Almedy
DATE MON 25 8:35 AM FLIGHT NO KB70
Ticket Price $ 170.00

DBC Dabaca 1h 35m ADY Almedy
DATE MON 25 8:35 AM FLIGHT NO KB71



DBC
Dabaca

MON 25 , 8:35 AM

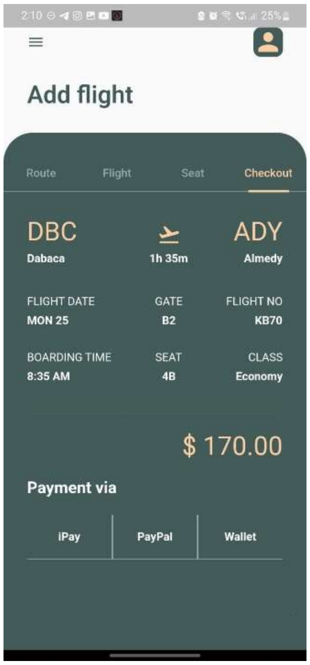
Economy class

1h 35m

ADY
Almedy

FLIGHT NO
KB70

4B
Seat



DBC
Dabaca

1h 35m

ADY
Almedy

FLIGHT DATE	GATE	FLIGHT NO
MON 25	B2	KB70
BOARDING TIME	SEAT	CLASS
8:35 AM	4B	Economy

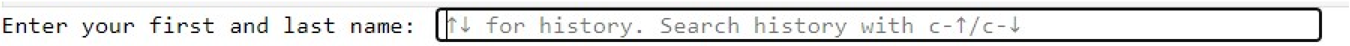
$ 170.00

Payment via

Implementation

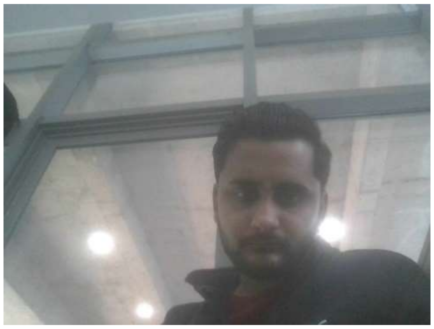
The IDE or software we used for this project is Jupyter Notebook it is an open-source software that supports multiple languages, but we use Python as our main language because it supports it well with AI.

The first step taking the name from the user to create a bounding box

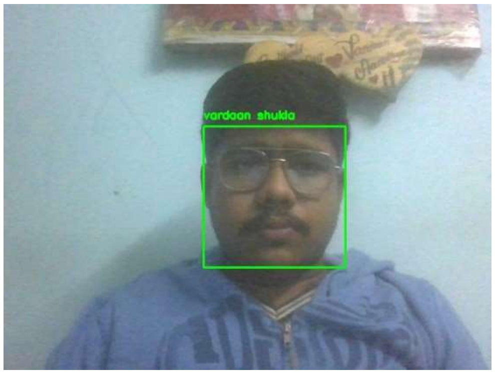


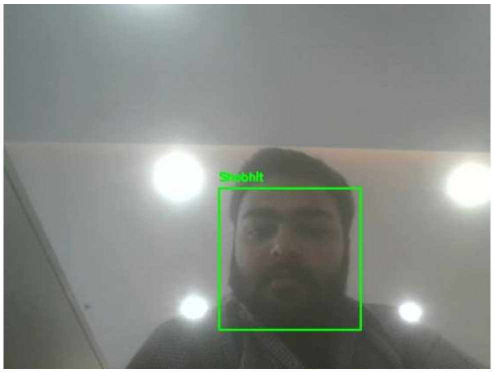
```
Enter your first and last name: ↑↓ for history. Search history with c-↑/c-↓
```

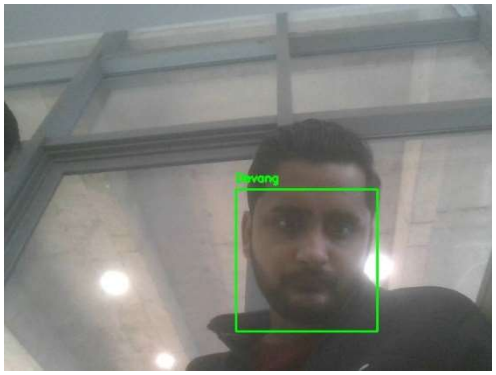
After that, we’ll capture the image of the user



The next step will be annotation

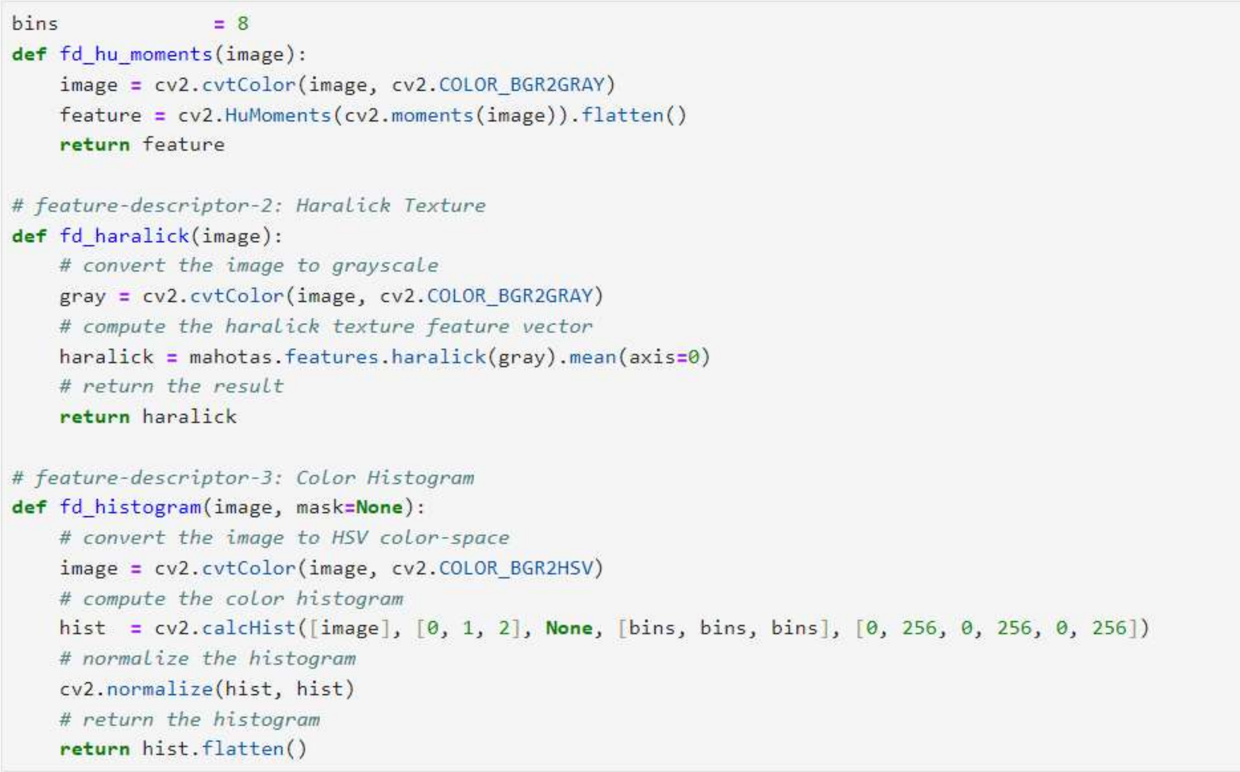






After that cropped images will be taken for feature extraction

Code for feature extraction

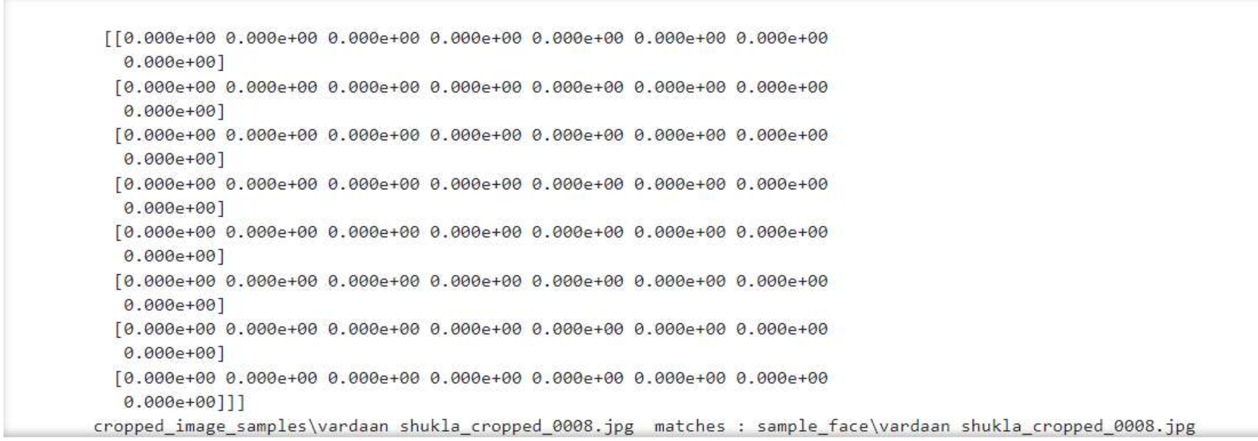


```
bins             = 8
def fd_hu_moments(image):
    image = cv2.cvtColor(image, cv2.COLOR_BGR2GRAY)
    feature = cv2.HuMoments(cv2.moments(image)).flatten()
    return feature

# feature-descriptor-2: Haralick Texture
def fd_haralick(image):
    # convert the image to grayscale
    gray = cv2.cvtColor(image, cv2.COLOR_BGR2GRAY)
    # compute the haralick texture feature vector
    haralick = mahotas.features.haralick(gray).mean(axis=0)
    # return the result
    return haralick

# feature-descriptor-3: Color Histogram
def fd_histogram(image, mask=None):
    # convert the image to HSV color-space
    image = cv2.cvtColor(image, cv2.COLOR_BGR2HSV)
    # compute the color histogram
    hist  = cv2.calcHist([image], [0, 1, 2], None, [bins, bins, bins], [0, 256, 0, 256, 0, 256])
    # normalize the histogram
    cv2.normalize(hist, hist)
    # return the histogram
    return hist.flatten()
```

Output for features extraction and face detection



```
   [[0.000e+00 0.000e+00 0.000e+00 0.000e+00 0.000e+00 0.000e+00 0.000e+00
     0.000e+00]
    [0.000e+00 0.000e+00 0.000e+00 0.000e+00 0.000e+00 0.000e+00 0.000e+00
     0.000e+00]
    [0.000e+00 0.000e+00 0.000e+00 0.000e+00 0.000e+00 0.000e+00 0.000e+00
     0.000e+00]
    [0.000e+00 0.000e+00 0.000e+00 0.000e+00 0.000e+00 0.000e+00 0.000e+00
     0.000e+00]
    [0.000e+00 0.000e+00 0.000e+00 0.000e+00 0.000e+00 0.000e+00 0.000e+00
     0.000e+00]
    [0.000e+00 0.000e+00 0.000e+00 0.000e+00 0.000e+00 0.000e+00 0.000e+00
     0.000e+00]
    [0.000e+00 0.000e+00 0.000e+00 0.000e+00 0.000e+00 0.000e+00 0.000e+00
     0.000e+00]
    [0.000e+00 0.000e+00 0.000e+00 0.000e+00 0.000e+00 0.000e+00 0.000e+00
     0.000e+00]]]
cropped_image_samples\vardaan shukla_cropped_0008.jpg  matches : sample_face\vardaan shukla_cropped_0008.jpg
```

Testing

Testing is the one type of process in which it checks that the system is working properly or not?”

Type of testing:

1) Unit testing:

In unit testing the analyst tests the programs making up a system.

2) System testing:

It contains details regarding security testing, performance testing, load testing, and compatibility testing information details in it.

3) Code testing: The code testing strategy examines the logic of the program.

4) Validation testing:

In validation testing first, we have to check whether all the requirements that were given to the Administration are fulfilled or not.

Results and Discussion

In the latest time we can see that if anybody wants to create a new dataset, he or she must do annotation on its own it is a very hectic process because a dataset includes a huge number of images, we have just tried to ease this process but for now it is just for face. All the process working, and output are presented above.

The project is still under development and further polishing of the code and other minute details is required to work efficiently. As of now the model is working with accepted accuracy and provides output as per the expectation. Further improvements are still undergoing for the project.

Conclusion

The project is in the early development phase as this is a long-term project and will require time for further development. The project is so far as per the expectations and once the project is 100% developed, we can approach different companies in the aviation industry and can pitch the project to them. As many companies are already developing the same kind of technology there is a high chance of our project going into public bid. Till that time, we will be polishing the project and will introduce further improvements as we go along.

References

1. HaralikTextureExtraction
https://www.orfeo-toolbox.org/CookBook/Applications/app_HaralickTextureExtraction.html
https://stackoverflow.com/

2. OpenCV
https://opencv.org/
https://github.com/opencv/opencv

3. Application concept
https://docs.flutter.dev/

4. Coding application
https://www.python.org/

5. Literature review reference links

https://www.nec.com/en/global/solutions/safety/resourcecenter/download/NEC_IDelight_infographic.pdf

https://www.forbes.com/sites/jenniferleighparker/2021/10/27/first-look-delta-tsa-launch-facial-recognition-at-atlanta-airport/?sh=1bf62dfd4dc2

https://www.changiairport.com/corporate/media-centre/changijourneys/the-changi-experience/fast-changi.html

https://www.softwaretestingnews.co.uk/singapore-changi-airport-tests-facial-recognition-systems/

https://www.emirates.com/in/english/experience/contactless-journey/

https://www.airport-technology.com/news/emirates-biometric-path-dubai-airport/

https://mediacentre.britishairways.com/pressrelease/details/14300#:~:text=Customers%20who%20sign%20up%20to,being%20kept%20safe%20and%20secure.

https://mediacentre.britishairways.com/pressrelease/details/8422

https://www.cbp.gov/travel/biometrics

https://www.passengerselfservice.com/2019/06/hamad-airport-to-use-facial-recognition-at-passenger-touch-points/

https://www.sita.aero/globalassets/docs/brochures/smart-path-product-brochure.pdf

https://news.klm.com/test-at-schiphol-boarding-quickly-and-easily-using-facial-recognition/

Acknowledgments

An acknowledgement should be an expression of genuine gratitude and not a set of platitudes. Here with we pen down the thank fullness. We own to all those who made this endeavour ours successful. We do not want to miss the opportunity to thanks all those who had helped us either directly or indirectly, through their constant guidance, which helped us, put in our best.

I extend my heartfelt gratitude to **Mr. Manoj K. Singh** for their invaluable guidance and support, which were instrumental in the success of this project, along with **Dr. Mohit Agarwal** who guided us for the successful working and implementation of the project.

We are thankful to Bennett University for providing us with an opportunity to work on their premises and for guidance & cooperation during the project.