

HỌC VIỆN CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG
CƠ SỞ THÀNH PHỐ HỒ CHÍ MINH



MÔN HỌC: NHẬP MÔN CÔNG NGHỆ PHẦN MỀM
BÁO CÁO ĐỀ TÀI: CHƯƠNG TRÌNH CẢNH BÁO TỶ LỆ
LÂY NHIỄM COVID-19 GIỮA NGƯỜI VỚI NGƯỜI

Giảng viên	:	Nguyễn Thị Bích Nguyên
Sinh viên thực hiện	:	Đỗ Anh Quân.....N17DCAT055
		Hoàng Thị Mỹ Như.....N17DCCN052
		Bùi Tuấn AnhN17DCAT002

1. Hiện trạng

Dịch COVID 19 là một đại dịch bệnh truyền nhiễm bởi tác nhân virus SARS-CoV-2, đang diễn ra trên phạm vi toàn cầu. Chủng virus mới này chủ yếu lan qua các giọt bắn trong không khí khi một cá nhân bị nhiễm bệnh ho hoặc hắt xì hơi trong phạm vi khoảng 0,91 m đến 1,8 m. Với tình hình cấp bách đó ý tưởng nhận diện khuôn mặt để cảnh báo tỷ lệ lây nhiễm virus COVID -19 ra đời để giúp các cơ quan y tế có thể kịp thời ứng phó với sự lây nhiễm của dịch bệnh.

- Quy trình các công việc liên quan đến đề tài:

+ **Đăng kí người dùng:** người dung nhập thông tin và hình ảnh.

+ **Sử dụng AI để nhận dạng và tính toán tỷ lệ:** dùng công cụ có chức năng ghi hình để nhận dạng người dung và tính toán tỷ lệ

+ **Xác nhận người dung bị nhiễm bệnh và đưa ra các cảnh báo:** Người quản lý xác nhận người dung đã bị nhiễm bệnh và đưa thông tin lên máy tính để nhận được các kết quả cảnh báo với những người đã tiếp xúc với người bị nhiễm.

- Các mẫu biểu có liên quan:

+ Đăng kí người dùng.

+ Hiệu chỉnh số liệu.

- Các quy định ràng buộc có liên quan:

+ Người dung phải cung cấp hình ảnh liên quan đến bản thân.

- Các quy định công thức tính có liên quan:

+ Công thức tính tỷ lệ lây nhiễm bằng (khoảng cách ít nhất không lây bệnh / khoảng cách thực tế) * 100 – tỷ lệ kháng bệnh khi đeo khẩu trang.

+ Công thức tính tỷ lệ kháng bệnh khi đeo khẩu trang bằng (tỷ lệ kháng bệnh của khẩu trang khi người giao tiếp có mang khẩu trang) + (tỷ lệ kháng bệnh của khẩu trang khi người giao tiếp có mang khẩu trang) / 2 (**Ghi chú: ở đây chỉ tính giao tiếp 1 người với 1 người**)

2. Yêu cầu

2.1 Yêu cầu chức năng nghiệp vụ

Bảng yêu cầu chức năng nghiệp vụ

Bộ phận (người thực hiện): Quản lý

STT	Công việc	Loại công việc	Quy định/ Công thức liên quan	Biểu mẫu liên quan	Ghi chú
1	Quản lý người dùng	Lưu trữ	Việc thêm người dùng dựa vào tên và hình ảnh do người dùng cung cấp.	Đăng kí người dùng	
2	Xác nhận người bị nhiễm bệnh	Tính toán	Xác nhận người bị nhiễm bệnh để máy tính đưa ra các dự đoán Công thức: Công thức tính tỷ lệ lây nhiễm.	Xác nhận nhiễm bệnh	

Bộ phận (người thực hiện): Người dùng

STT	Công việc	Loại công việc	Quy định/ Công thức liên quan	Biểu mẫu liên quan	Ghi chú
1	Sử dụng	Kết xuất	Xuất hiện để hệ thống xác định phân tích tỷ lệ lây nhiễm.		

Đăng kí người dùng

Name

Files

Không tệp nào được chọn

Xác nhận nhiễm bệnh

Name

Meo

At time:

nn/mm/yyyy --:-- --



ADD

2.2 Yêu cầu chức năng hệ thống và yêu cầu chất lượng

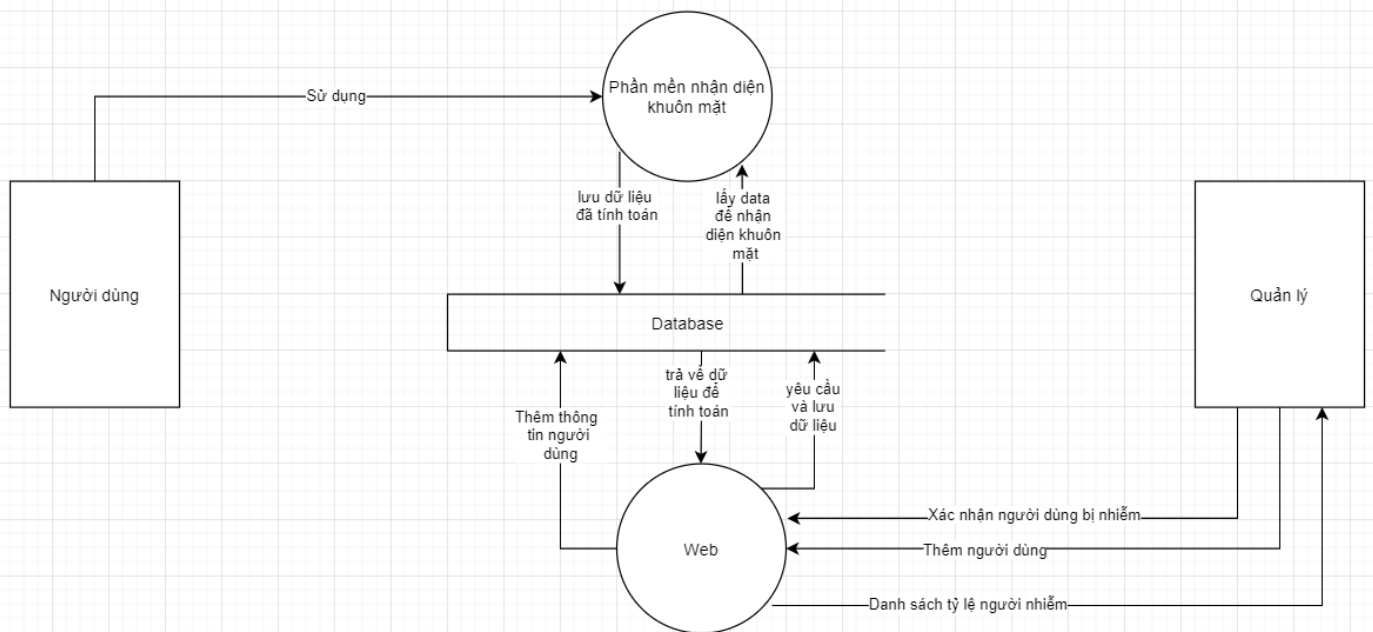
Bảng yêu cầu chức năng hệ thống:

STT	Nội dung	Mô tả chi tiết	Ghi chú
1	Nhận diện khuôn mặt của người dùng	Hệ thống phải nhận diện được khuôn mặt của mỗi người dùng, người đó có đeo khẩu trang không, khoảng cách giữa người với người, dựa vào các hình ảnh của người dùng đó đã cung cấp cho hệ thống.	
2	Tính toán tỷ lệ lây nhiễm giữa người với người	Hệ thống tính toán tỷ lệ lây nhiễm giữa người với người dựa vào số liệu được ghi nhận. Chia ra 2 phần: + Tính toán tỷ lệ người dùng khi chưa mắc bệnh (tích hợp trong nhận diện khuôn mặt). + Tính toán tỷ lệ người dùng khi mắc bệnh.	
3	Thêm người dùng	Thêm người dùng với hình ảnh và tên.	

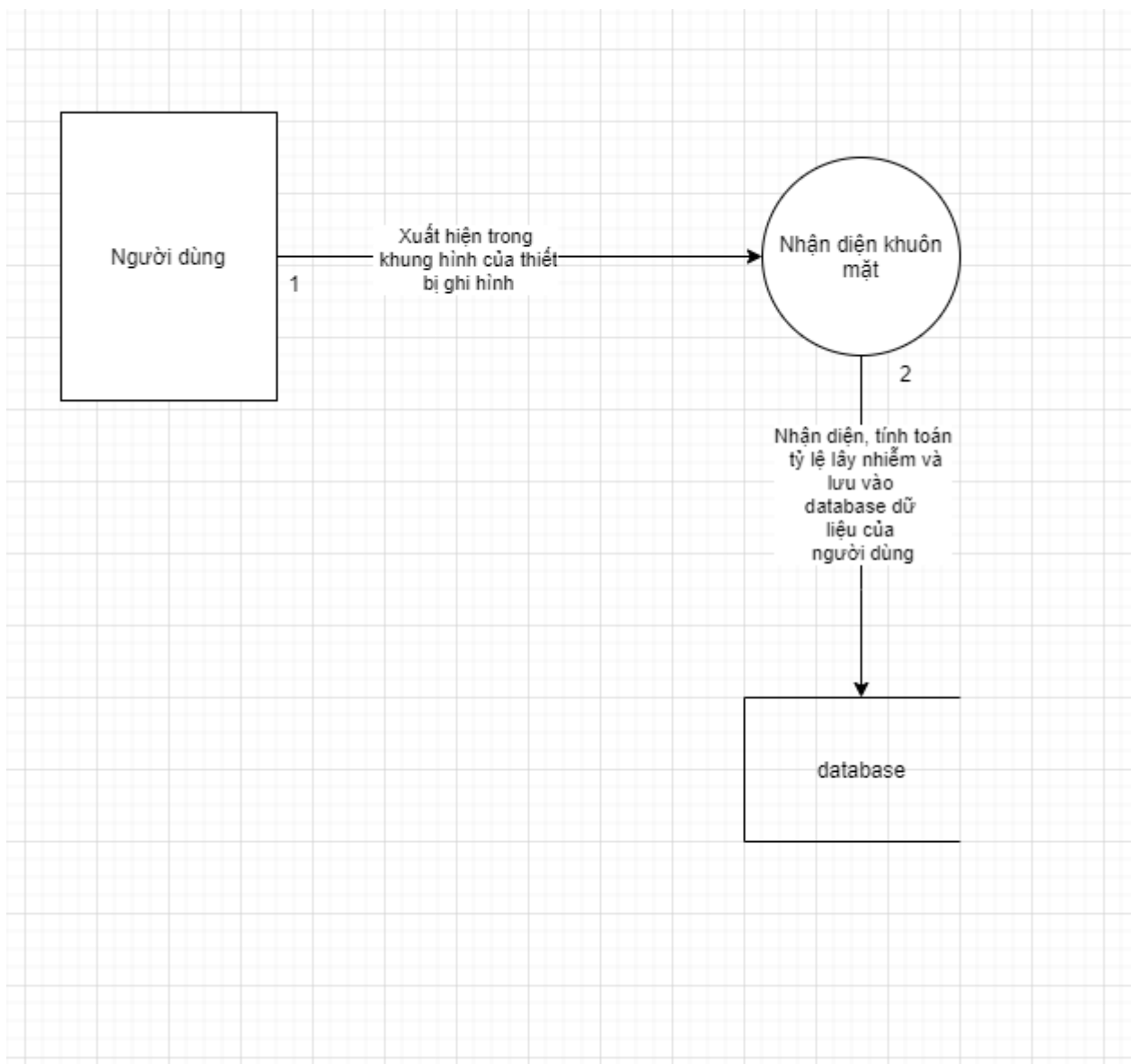
Bảng yêu cầu về chất lượng hệ thống:

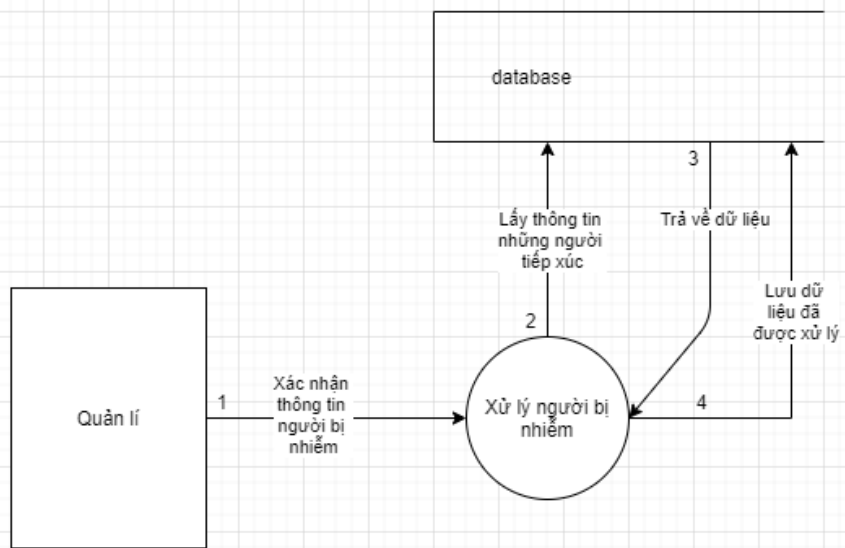
STT	Nội dung	Tiêu chuẩn	Mô tả chi tiết	Ghi chú
1	Nhận diện khuôn mặt của người dùng	Hiệu quả	Hệ thống phải nhận ra người dùng chính xác	
2	Tính toán tỷ lệ lây nhiễm giữa người với người	Hiệu quả	Hệ thống phải tính toán tỷ lệ xác xuất một cách chính xác	
3	Thêm người dùng	Tiện dụng	Thêm người dùng với hình ảnh và tên	

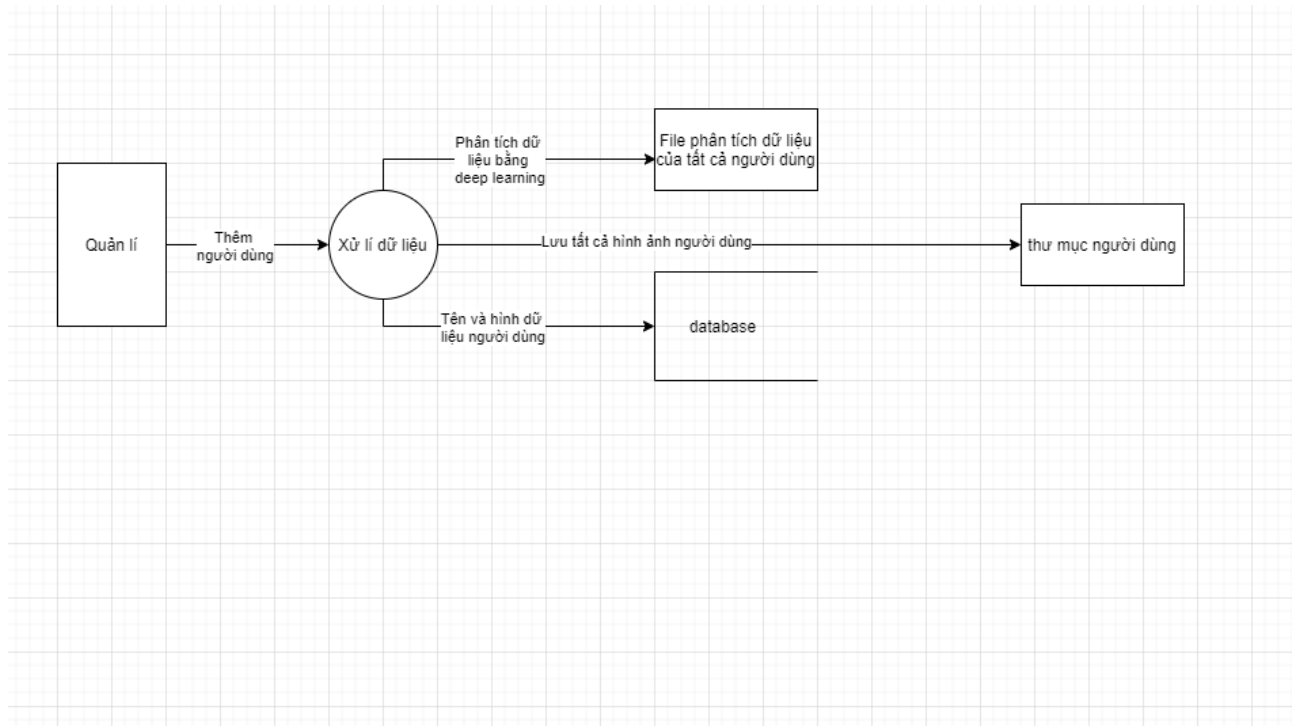
2.4 Mô hình luồng dữ liệu (DFD) chung cho toàn bộ hệ thống



2.5 Mô hình luồng dữ liệu (DFD) theo yêu cầu



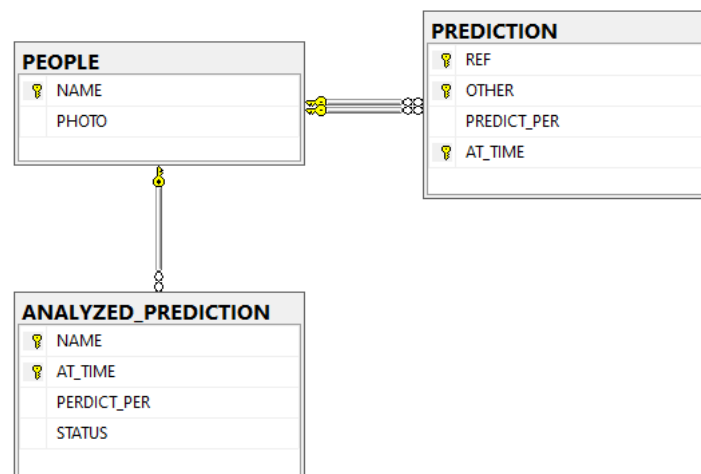




3. Thiết kế phần mềm

3.1 Thiết kế dữ liệu

Mô hình:



Danh sách các thành phần của sơ đồ

Stt	Tên	Loại	Ý nghĩa	Ghi chú
1	PEOPLE		Người dùng	
2	PREDICTION		Tỷ lệ lây nhiễm người với người	
3	ANALYZED_PREDICTION		Kết quả phân tích của hệ thống	

Danh sách các thuộc tính của từng thành phần

PEOPLE

Stt	Tên	Loại	Kiểu	Miền giá trị	Ý nghĩa
1	NAME		String	$x \geq 0$ $x \leq 250$	Khoá chính
2	PHOTO		String	$x \geq 0$ $x \leq 250$	Ảnh đại diện của người dùng

PREDICTION

Stt	Tên	Loại	Kiểu	Miền giá trị	Ý nghĩa
1	REF		String	$x \geq 0$ $x \leq 250$	Tên người tiếp xúc
2	OTHER		String	$x \geq 0$ $x \leq 250$	Tên người bị tiếp xúc
3	PREDICT_P R		Float	$x \geq 0$ $x \leq 100$	Tỷ lệ lây nhiễm khi tiếp xúc
4	AT_TIME		Date time		Thời gian tiếp xúc

-

ANALYZED_PREDICTION

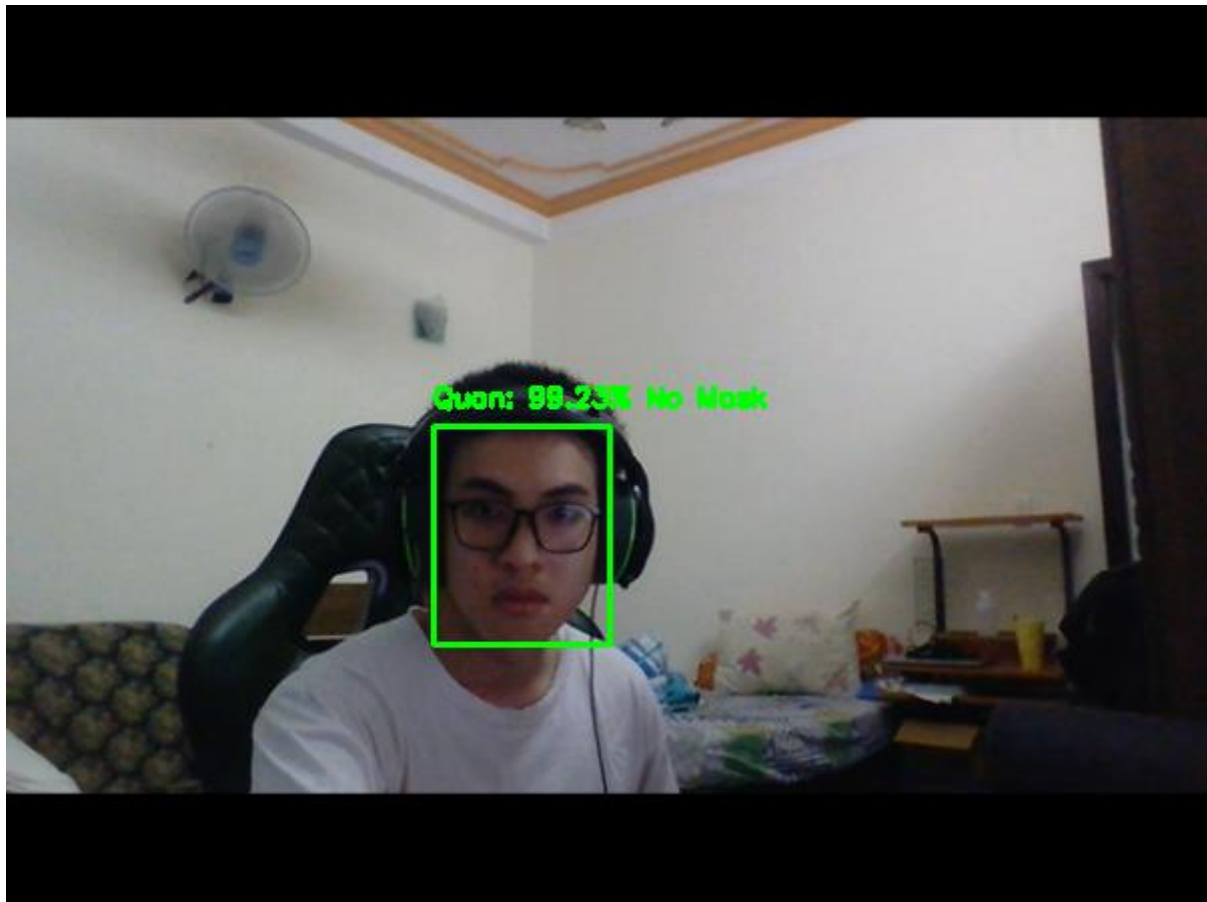
Stt	Tên	Loại	Kiểu	Miền giá trị	Ý nghĩa
1	NAME		String	$x \geq 0$ $x \leq 250$	Tên người dùng
2	AT_TIME		Date time		Thời điểm người dùng mắc bệnh
3	PERDICT_P R		Float	$x \geq 0$ $x \leq 100$	Tỷ lệ mắc bệnh
4	STATUS		Bool		Tình trạng sức khỏe của người dùng

3.2 Thiết kế giao diện

Stt	Mã số	Loại	Ý nghĩa	Ghi chú
1	MH_1	Màn hình hệ thống nhận diện người dùng	Hệ thống nhận diện và phân tích dữ liệu	
2	MH_2	Màn hình tra cứu tỷ lệ nhiễm bệnh / màn hình chính	Tra cứu tỷ lệ nhiễm bệnh của mỗi người dùng	
3	MH_3	Màn hình chọn người bị nhiễm	Ấn định người bị nhiễm cho phép hệ thống tính toán tỷ lệ nhiễm bệnh	
4	MH_4	Màn hình nhập dữ liệu người bị nhiễm bệnh	Nhập dữ liệu người bị nhiễm bệnh	
5	MH_5	Màn hình thêm người dùng	Thêm người dùng	

Mô tả chi tiết từng màn hình

MH_1







Stt	Biến cố	Ý nghĩa	Xử lý tương ứng	Mã số xử lý
1	Khung khuôn mặt người	Hệ thống nhận diện mặt người	Lấy thông tin từ khung hình và tìm trên đó một dãy các chuỗi số đồng dạng với chuỗi số mặt người	
2	Dòng chữ trên khung mặt người	Hệ thống nhận diện người dùng đã đăng kí trên hệ thống	Lấy thông tin từ khuôn mặt đã nhận dạng được và tìm định dạng khuôn mặt có trùng với khuôn mặt của người đã đăng kí với hệ thống	
3	Các số trên khung mặt người	Hệ thống phỏng đoán tỷ lệ chính xác của khuôn mặt người dùng	Khi người dung đã được nhận dạng và cho ra tỷ lệ phỏng đoán của hệ thống	

4	Chữ “No Mask” hay “Mask”	Hệ thống người dùng có đeo khẩu trang hay không	Hệ thống sẽ đưa ra phỏng đoán dựa trên khuôn mặt của người dùng khi người đó đeo hay không đeo khẩu trang	
---	--------------------------	---	---	--

MH_2

PREDICT COVID ADD USER ADD EFFECTED



#	NAME	PERCENTAGE	IMAGE	TIME	STATUS	ACTION
1	Meo	100%		5/29/2020 11:08:00 AM	Effected	Remove effected
2	Quan	100%		5/30/2020 11:12:00 AM	Effected	Remove effected
3	Quan	100%		5/31/2020 10:44:00 PM	Predicted	
4	Heo	100%		5/31/2020 10:44:00 PM	Predicted	

© 2020 - PREDICT COVID Application

Stt	Biến cố	Ý nghĩa	Xử lý tương ứng	Mã số xử lý
1	Nhập Remove Effected	Xác nhận người bị nhiễm đã hết bệnh	Hệ thống sẽ xóa dòng dữ liệu tương ứng	

MH_3

PREDICT COVID ADD USER ADD EFFECTED

#	NAME	IMAGE	ACTION
1	Cho		SET EFFECTED
2	Heo		SET EFFECTED


© 2020 - PREDICT COVID Application

Stt	Biến cố	Ý nghĩa	Xử lý tương ứng	Mã số xử lý
1	Nhấp chuột vào SET EFFECTED	Xác nhận người dùng bị nhiễm bệnh	Chuyển sang form set effected	

MH_4

PREDICT COVID ADD USER ADD EFFECTED

Name Cho

At time: 

© 2020 - PREDICT COVID Application

Stt	Biến cố	Ý nghĩa	Xử lý tương ứng	Mã số xử lý
1	Nhập thời gian bị nhiễm bệnh ở box At time	Thời gian người dùng bị nhiễm bệnh	Hệ thống sẽ phân tích dữ liệu từ database và đưa ra các phỏng đoán về người có tỷ lệ nhiễm bệnh của người bị tiếp xúc	

Name

Files Không có tệp... o được chọn

Stt	Biến cố	Ý nghĩa	Xử lý tương ứng	Mã số xử lý
1	Nhập tên người dùng ở box Name	Tên người dùng		
2	Nhấp vào box Files	Nhập nhiều file hình ảnh của người dùng		
3	Nhấp chuột vào nút Add	Tạo người dùng	Hệ thống sẽ tạo người dùng và phân tích lại file dữ liệu để nhận dạng	

3.3 Thiết kế xử lý

Danh sách các xử lý

Stt	Tên	Loại	Ý nghĩa	Ghi chú
1	Xử lý hình ảnh	Xử lý hệ thống	Nhận dạng khuôn mặt	
2	Tính toán tỷ lệ nhiễm bệnh	Xử lý nghiệp vụ	Tính toán tỷ lệ nhiễm bệnh	

Chi tiết từng xử lý

1. Xử lý hình ảnh

Bước 1: Lấy thông tin từ khung hình ghi nhận

```
frame = vs.read()
```

Bước 2: Tìm trong khung hình đó có khuôn mặt người không

```
# dimensions
frame = imutils.resize(frame, width=600)
(h, w) = frame.shape[:2]
frameBlob = cv2.dnn.blobFromImage(
    cv2.resize(frame, (300, 300)), 1.0, (300, 300),
    (104.0, 177.0, 123.0))
#, swapRB=False, crop=False
# apply OpenCV's deep learning-based face detector to localize
# faces in the input image
detector.setInput(frameBlob)
detections = detector.forward()
for i in range(0, detections.shape[2]):
    confidence = detections[0, 0, i, 2]
```

Bước 3: Nếu có thì xử lý trong vùng mặt người đó có đeo khẩu trang hay không

```
# mask recognize
# ensure the bounding boxes fall within the dimensions of
# the frame
(startXMask, startYMask) = (max(0, startX), max(0, startY))
(endXMask, endYMask) = (min(w - 1, endX), min(h - 1, endY))
# extract the face ROI, convert it from BGR to RGB channel
# ordering, resize it to 224x224, and preprocess it
# face = frame[startY:endY, startX:endX]
faceMask = cv2.cvtColor(faceMask, cv2.COLOR_BGR2RGB)
faceMask = cv2.resize(faceMask, (224, 224))
faceMask = img_to_array(faceMask)
faceMask = preprocess_input(faceMask)
faceMask = np.expand_dims(faceMask, axis=0)
(mask, withoutMask) = maskNet.predict(faceMask)[0]
if mask >= withoutMask:
    label = "Mask"
    color = (255, 0, 0)
else:
    label = "No Mask"
```

Bước 4: Xử lý nhận dạng danh tính của người đó có trong hệ thống hay không và sau đó lưu trong mảng để tiếp tục xét hết khuôn mặt trong khung hình đó

```
text = ""
# make sure proba is greater than 50%
if proba * 100 > 55 and not FaceClass.isFaceExistedInDetected(name, detectedFacesArr):
    detectedFacesArr.append((faceROI,label,name))
    color = (0,255,0)
    text = "{:}: {:.2f}%".format(name, proba * 100)

cv2.putText(frame,text + " " + label, (startX, startY - 10), cv2.FONT_HERSHEY_SIMPLEX, 0.45, color, 2)
cv2.rectangle(frame, (startX,startY), (endX,endY),
               color,2)
```

Bước 5: Xử lý khoảng cách từ người tới người

```
# distance from reference face to other faces
for i in range(0,len(avg)):
    currentDetectedFace = None
    # find reference face
    if FaceClass.findROIInDetectedFaces(faces[i], detectedFacesArr):
        currentDetectedFace = FaceClass.findDetectedFace(faces[i], detectedFacesArr)
    # to other faces
    for p in range(i+1,len(avg)):
        # draw line
        cv2.line(frame, (avg[i][0], avg[i][1]), (avg[p][0], avg[p][1]), (0,0,255), 1)
        (m1, m2) = midpoint(avg[i], avg[p])
        # distance
        D = dist.euclidean(avg[i], avg[p]) / avg[i][2]
        # draw text
        cv2.putText(frame, "{:.1f}".format(D), (int(m1), int(m2 - 10)),
                    cv2.FONT_HERSHEY_SIMPLEX, 0.55, (0,0,255), 1)
        # find other faces
        if FaceClass.findROIInDetectedFaces(faces[p], detectedFacesArr) and currentDetectedFace is not None:
            otherDetectedFace = FaceClass.findDetectedFace(faces[p], detectedFacesArr)
            # calculate percentage
            percentage = Analyzer.analyzePercentByContactRange(D, currentDetectedFace[1], otherDetectedFace[1])
            # if reference and other face already in analyzed face
            analyzedFacesArr = Analyzer.analyzePredicted(analyzedFacesArr,
                FaceClass.AnalyzedFaces(currentDetectedFace[2],
                otherDetectedFace[2],
                percentage,0))
    # print(analyzedFacesArr)
    # print(len(faces), " + ", len(avg) )
```

2. Tính toán tỷ lệ nhiễm bệnh

Bước 1: Xét phần trăm trung bình 2 người có đeo khẩu trang hay không

```
# check if mask on or off
if maskRef == "Mask": percentRef = initialParams["mask_lower_percent"]
else: percentRef = 0
if maskOther == "Mask": percentOther = initialParams["mask_lower_percent"]
else: percentOther = 0
```

Bước 2: Tính toán phần trăm khoảng cách từ người với người so với ngoài cách chuẩn để không bị lây bệnh

```

percentArgMask = (percentKey + percentOther) / 2
# percent by contact range
percentDis = (initialParams["contact_range"] / distance) * 100
# if percent of contact is higher than 100% then make it 100%
if percentDis > 100: percentDis = 100

```

Bước 3: Tính toán phần trăm tỷ lệ nhiễm bệnh bằng phần trăm khoảng cách – phần trăm kháng bệnh

```

if percentDis > 100: percentDis = 100
# final calculation
finalCalculate = percentDis - percentArgMask
if finalCalculate < 0: finalCalculate = 0
return finalCalculate

```

3. Cài đặt và thử nghiệm

Cài đặt

Stt	Chức năng	Mức độ hoàn thành	Ý nghĩa
1	Nhận diện khuôn mặt	Phát triển	Xử lý
2	Tính toán tỷ lệ nhiễm bệnh	Hoàn thành	Tính toán tỷ lệ nhiễm bệnh
3	Tạo người dùng	Hoàn thành	Tạo người dùng
4	Thêm người bị nhiễm bệnh	Hoàn thành	Thêm người bị nhiễm bệnh

4. Tổng kết

- **Các kết quả đã thực hiện:** Tạo ra được sản phẩm gần hoàn thiện, có đầy đủ các chức năng cốt lõi.
- **Ưu điểm:** Có hoạt động.
- **Khuyết điểm:** Chưa đầy đủ chức năng.
- **Hướng mở rộng tương lai:** Phát triển thành 1 phần mềm hoàn chỉnh mang tính thực tế cao.

Tài liệu tham khảo:

<https://www.pyimagesearch.com/2020/05/04/covid-19-face-mask-detector-with-opencv-keras-tensorflow-and-deep-learning/>

<https://handmap.github.io/measuring-size-and-distance-opencv/>

<https://www.pyimagesearch.com/2016/04/04/measuring-distance-between-objects-in-an-image-with-opencv/>

<https://www.worldometers.info/coronavirus/transmission/>

<https://www.pyimagesearch.com/2020/05/04/covid-19-face-mask-detector-with-opencv-keras-tensorflow-and-deep-learning/>

<https://www.pyimagesearch.com/2018/09/24/opencv-face-recognition/>

