PROGRESS ON FILE STORAGE USING FACIAL PATTERN-BASED ENCRYPTION

John Pocasangre Jacob Acosta

Andrew Ibrahim Carlos Sandoval

Database tables progression

Established our database schema.

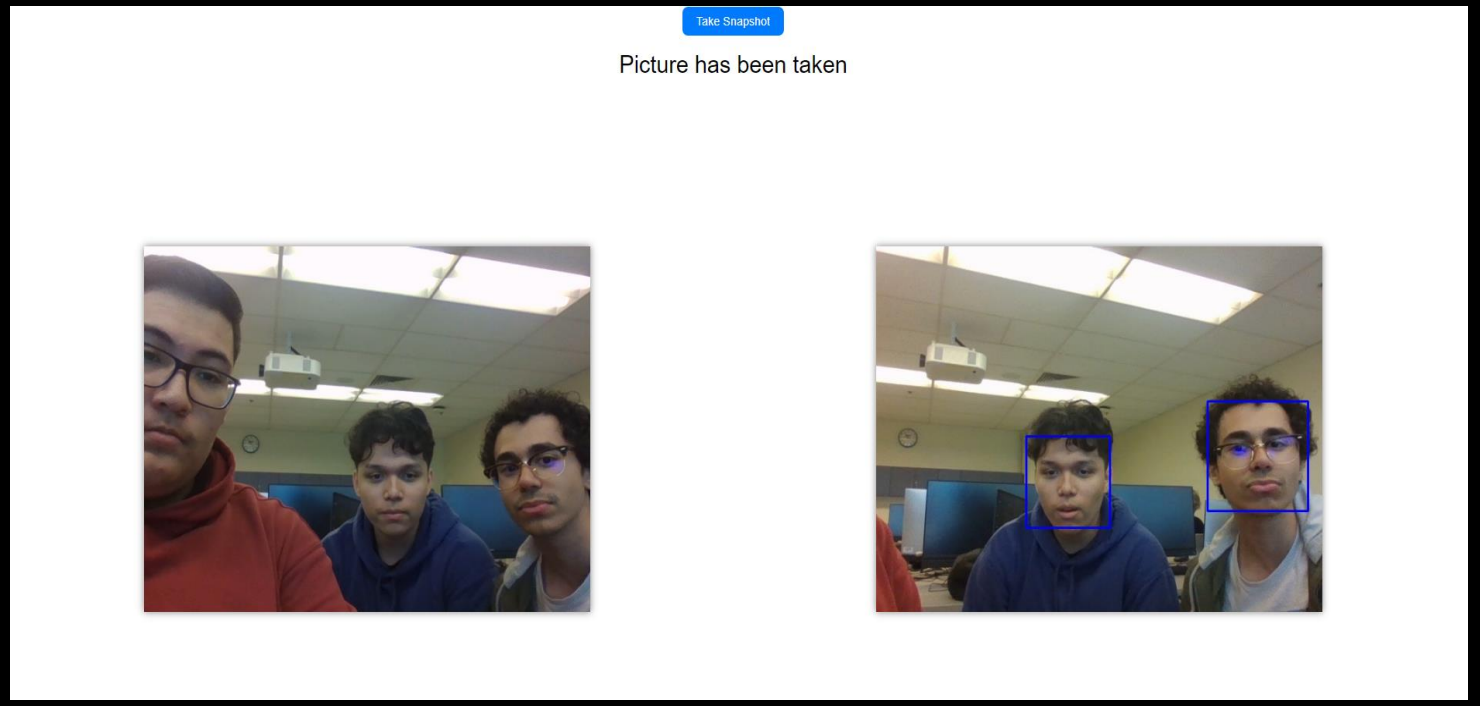
Will start with inserting mock data into the database and testing it.

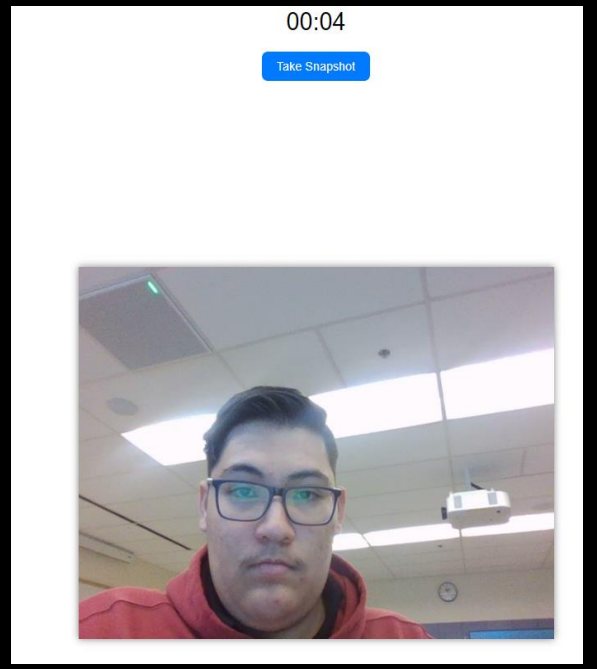
```
facial=> \dt
                List of relations
 Schema |         Name          | Type  | Owner
--------+-----------------------+-------+--------
 public | faceauthentication    | table | facial
 public | files                 | table | facial
 public | folder                | table | facial
 public | userinfo              | table | facial
(4 rows)

facial=>
```

Progress on analyzing Images from video stream to analyze if a Face is present

The camera stream that will be shown starts once camera access has been given, which then starts a 5 second count-down timer.

Once the timer reaches 0, a snapshot is taken of the current frame from the stream, and it is then analyzed by our python script. If a face is in the frame of said snapshot, a blue box is place on the face of the user, signaling that a face was detected/identified.





Python script for facial landmarks

- We have 2 different python scripts in use for our porject. This python script was designed to scan the face and return key values that we can use to apply to the encryption
- It gathers 128 plot points on the face, and it returns variables in an array that we can use as a base
- Our plan is to merge our two python scripts into one to manage both functionalities of detecting if a face is present, and then to analyze the face that was detected to collect the data.