

GROUP 4

HUMAN "EMOTION TRACKING" MOBILE APP

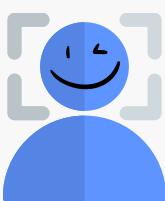
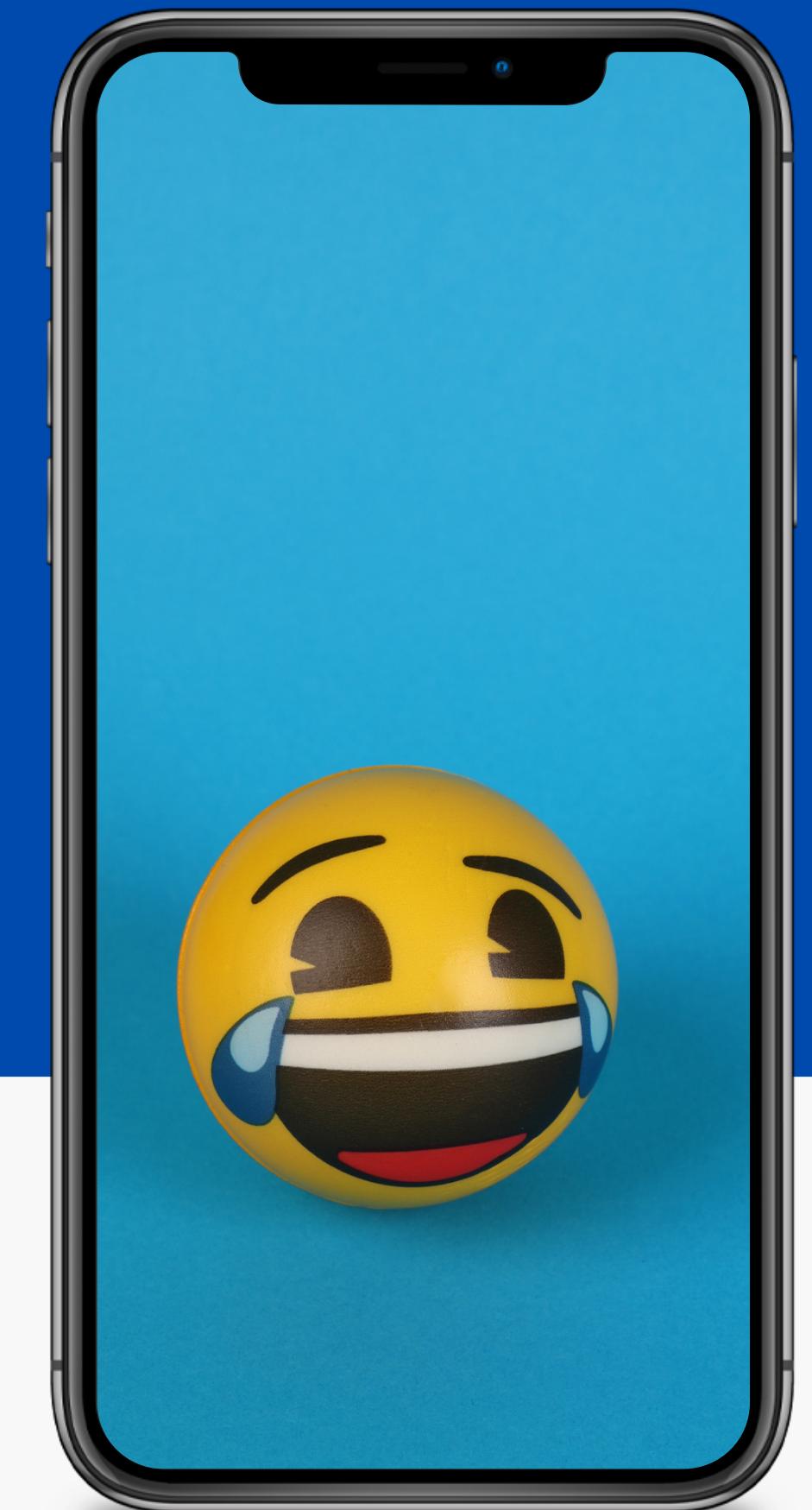
885II 22/23 - Mobile And Social
Sensing System

Fabio Buchignani, Tesfaye Yimam Mohammad,
Gabriele Marino, Matteo Razzai



TABLE OF CONTENTS

1. Introduction
2. Emotion Recognition
3. Clustering
4. Map and Firebase DB
5. Conclusion



INTRODUCTION

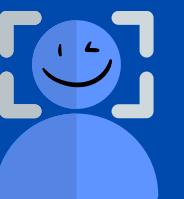


Why Emotion tracking?

Challenges

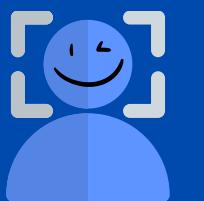
Goals of the project

Techs and Tools Used



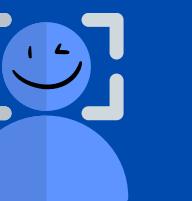
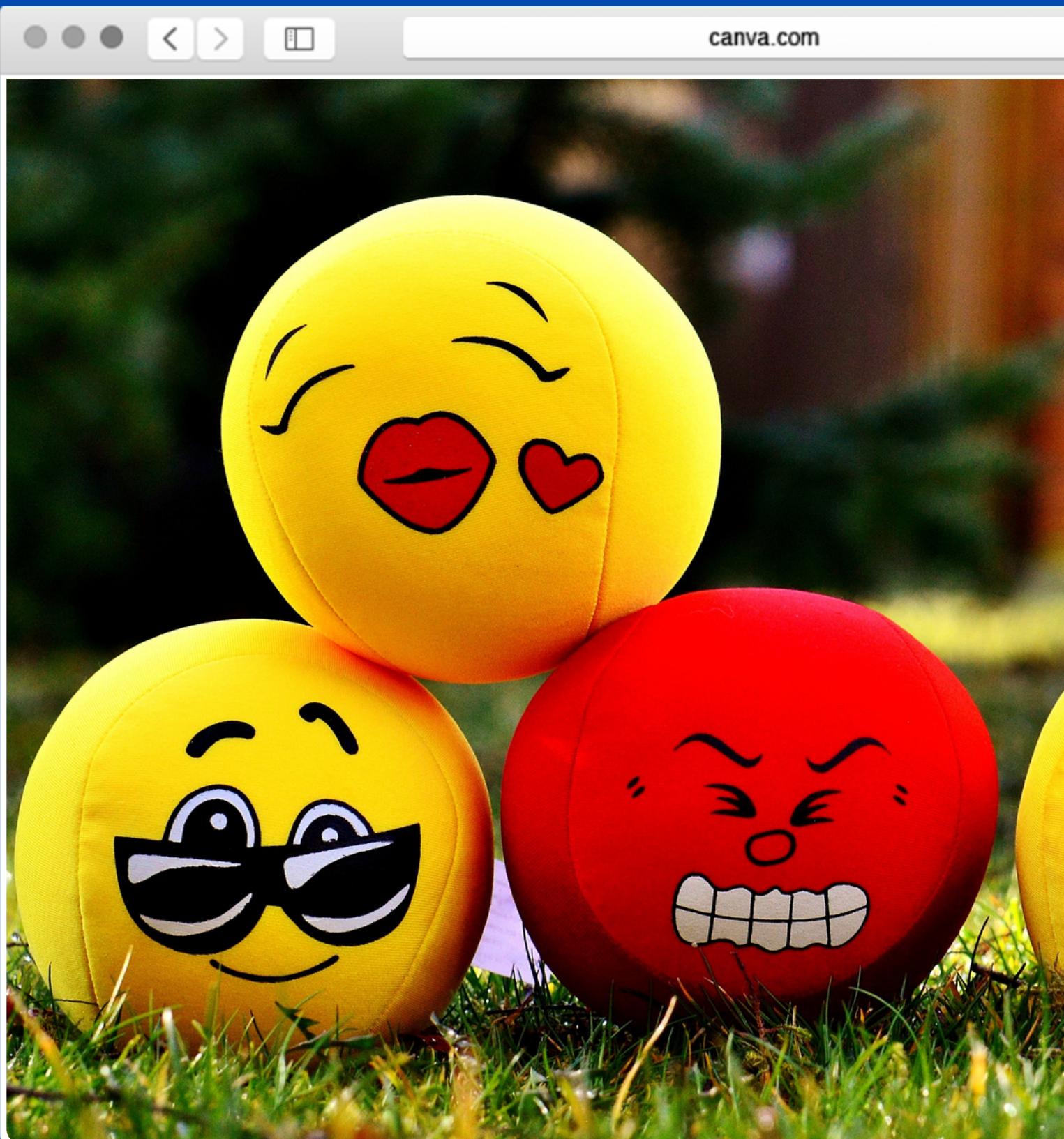
WHY EMOTION TRACKING?

- Emotions are one of the most important parts of our daily life, and we found it reasonable to work on this area.
- Facial Expressions play their role in a non-verbal communications and also helps to identify the current state of the person we are communicating with.
- Face being one of the exposed part of the human body allows for the use of different ML techniques, especially using the camera, to analyze and recognize the emotions of an individual.



CHALLENGES

- Develop a DL model for real time face detection to calculate happiness index.
- Accessing the location of the User
- Store the information in a safe place like remote FirebaseDatabase
- Do a clustering for a deeper understanding of user emotions and their spatial distribution.

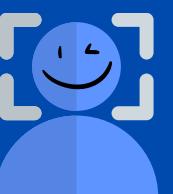


GOALS OF THE SYSTEM

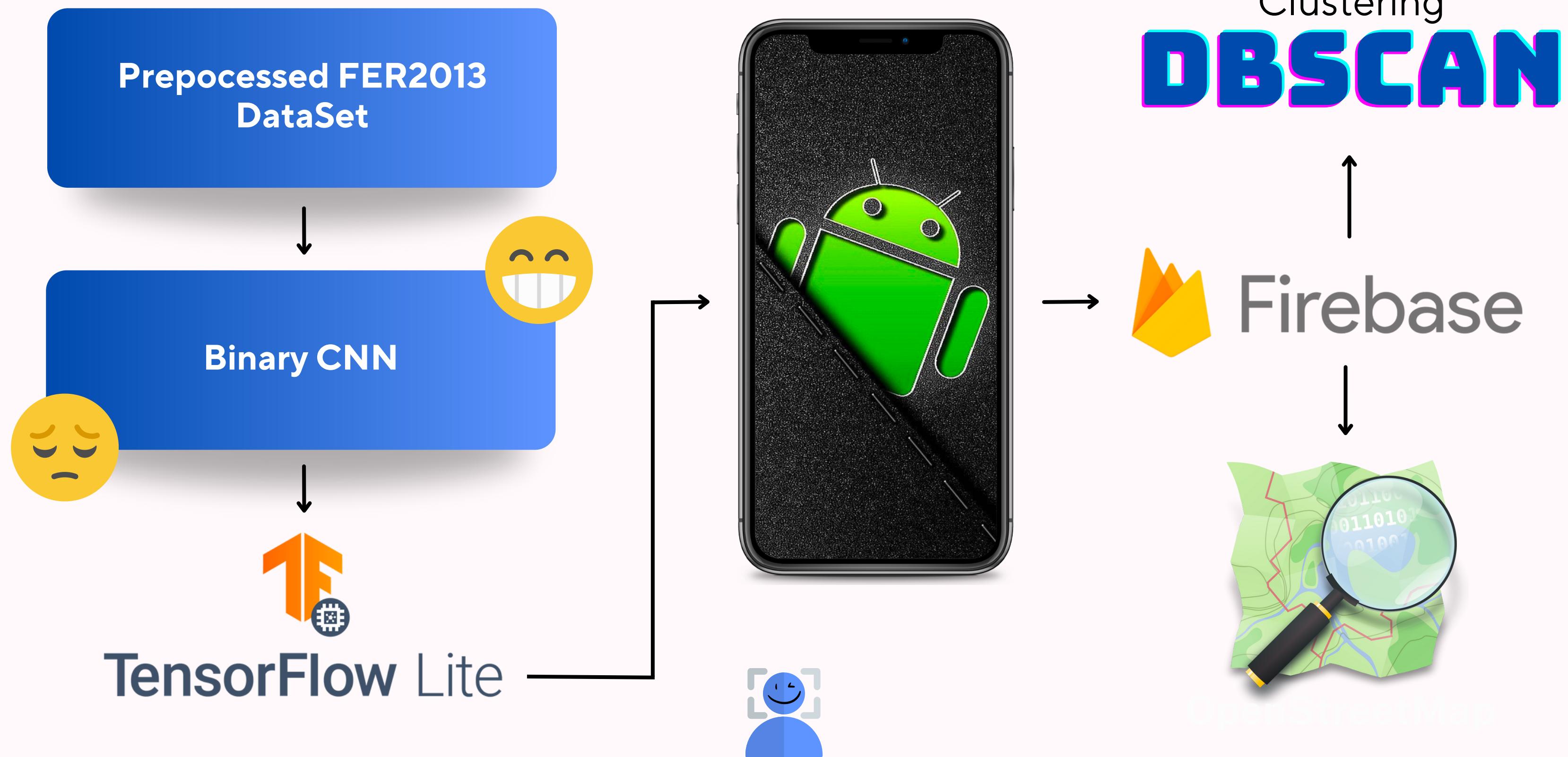
The goal of this project is to develop a solution that can collect the emotional status of the end user through the front camera and create a happiness map on the **openstreetmap**.

The sensor data, the location and the happiness level, is collected in real-time for calculating the emotions of the user through a smartphone app named as 'Emotoion Tracker'.

It requires the end user to give acces to their front camera, to switch-on the GPS to access their location where the application is being utilized.

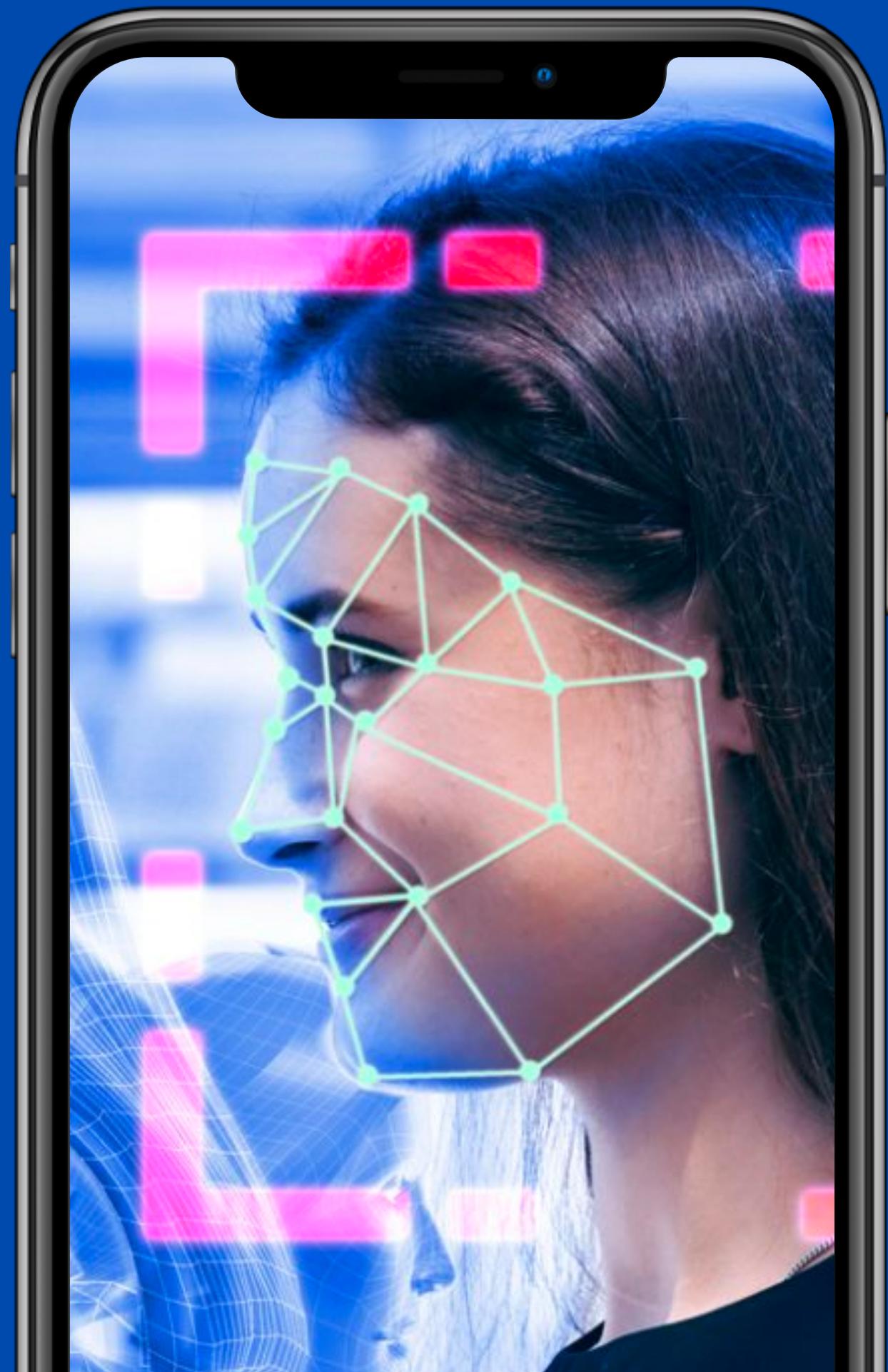
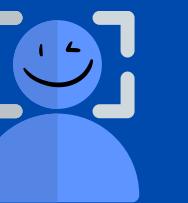


TECH AND TOOLS USED



EMOTION RECOGNITION

1. Handling data from the smartphone
2. The face detection module
3. The emotion recognizer module
4. Training details
5. Overall processing pipeline



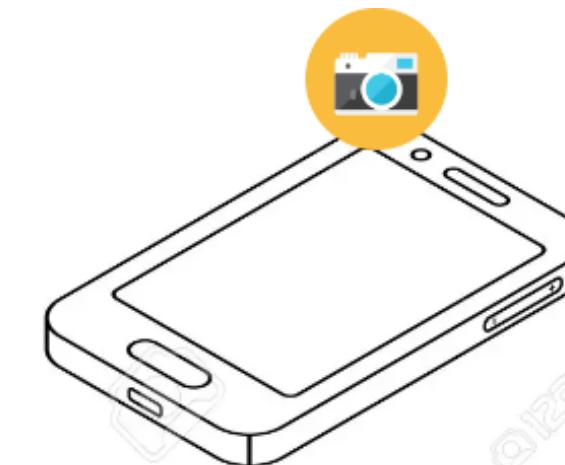
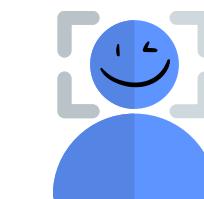
HANDLING DATA FROM THE SMARTPHONE

- **Goal:**

- Estimate the degree of happiness of the user in a specific location, using smartphone's sensors

- **Method:**

- Get video streaming from the smartphone, process frames to obtain an estimate of the user emotion over time using facial expressions
- Use GPS to get location information, periodically send sensed location and estimated happiness to the database



THE FACE DETECTION MODULE

The face detection module is implemented through the MLKIT face detection API, available in Android.

INPUT

Frame from the front camera

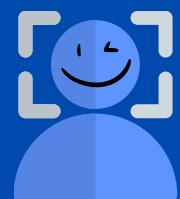
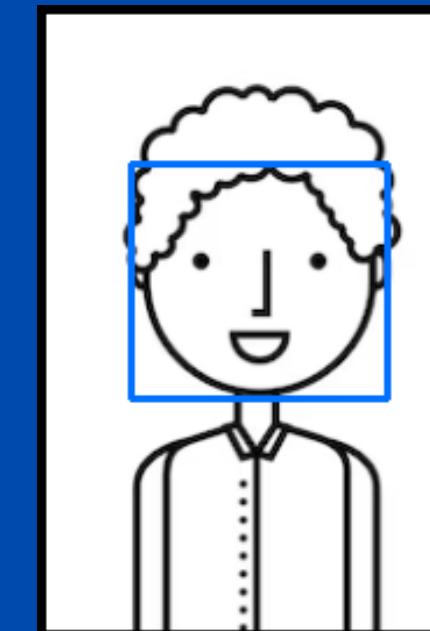


PROCESS

Detect faces in the frame, if any

OUTPUT

Bounding boxes information for the detected faces

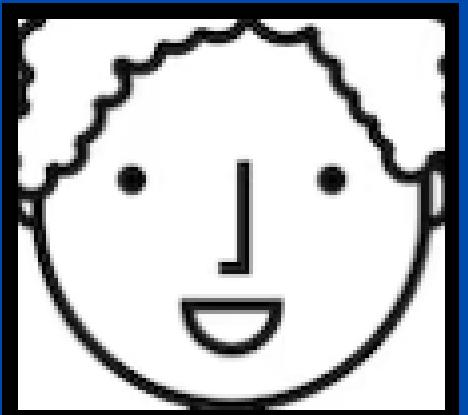


THE EMOTION RECOGNIZER MODULE

The emotion recognizer is a custom CNN developed in Tensorflow. The CNN has been trained using a modified version of the FER2013 dataset on a binary classification problem (happy/unhappy)

INPUT

Cropped frame
representing a face



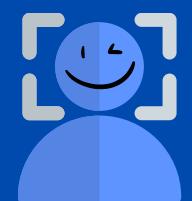
PROCESS

Evaluate happiness of the
represented face

OUTPUT

Probability that the
represented face is happy

0.9



THE EMOTION RECOGNIZER MODULE

(TRAINING DETAILS)

Emotion	Training set	Test set
Anger	3995	958
Disgust	436	111
Fear	4097	1024
Happiness	7215	1774
Neutral	4954	1233
Sad	4830	1247
Surprise	3171	831

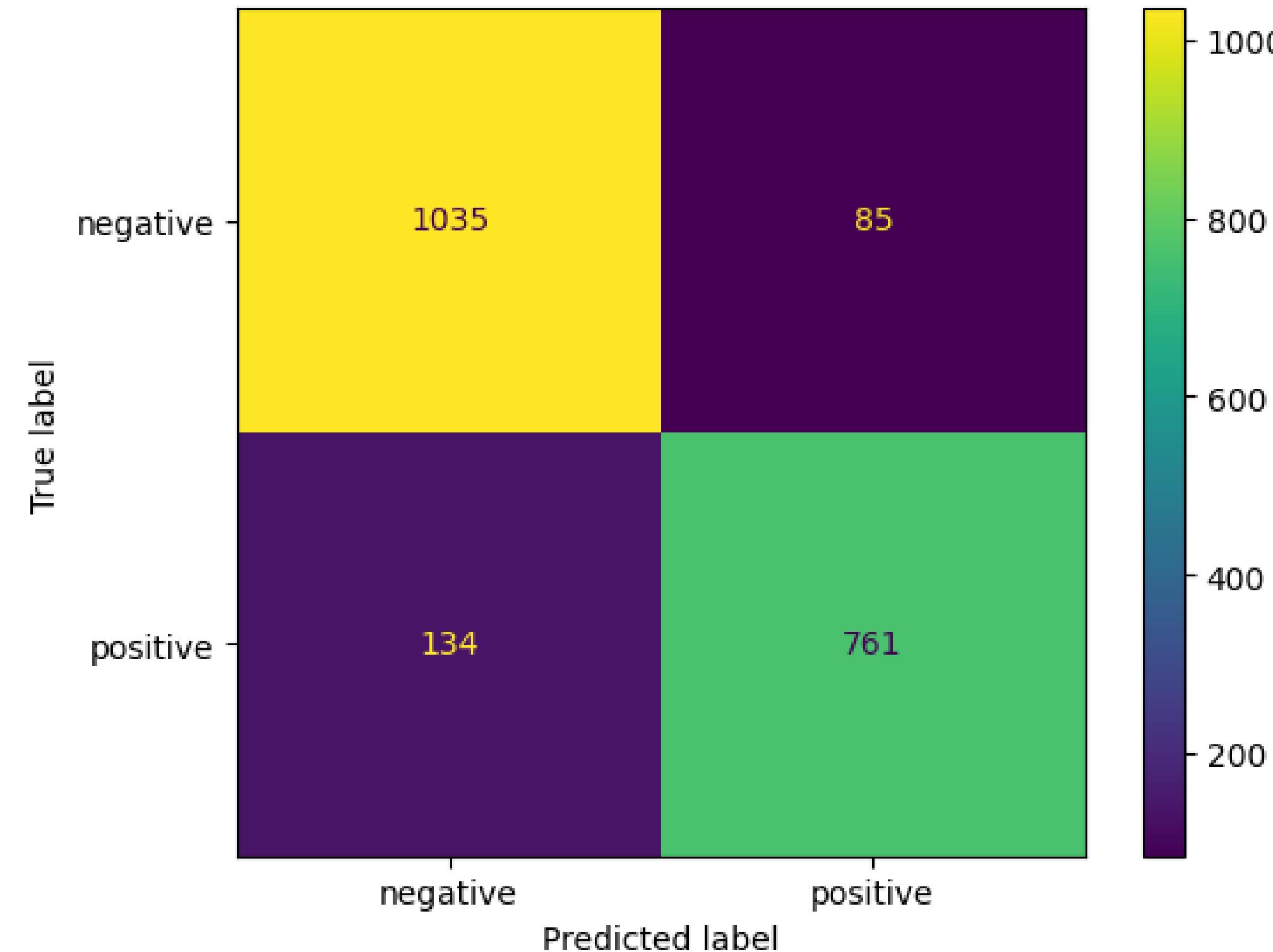


Emotion	Training set	Test set
Happy	7215	1774
Unhappy	8825	2205

	Training set	Validation set	Test set
Accuracy	91.63 %	89.26%	89.13 %
Loss	0.225	0.313	0.315



CONFUSION MATRIX ON TEST



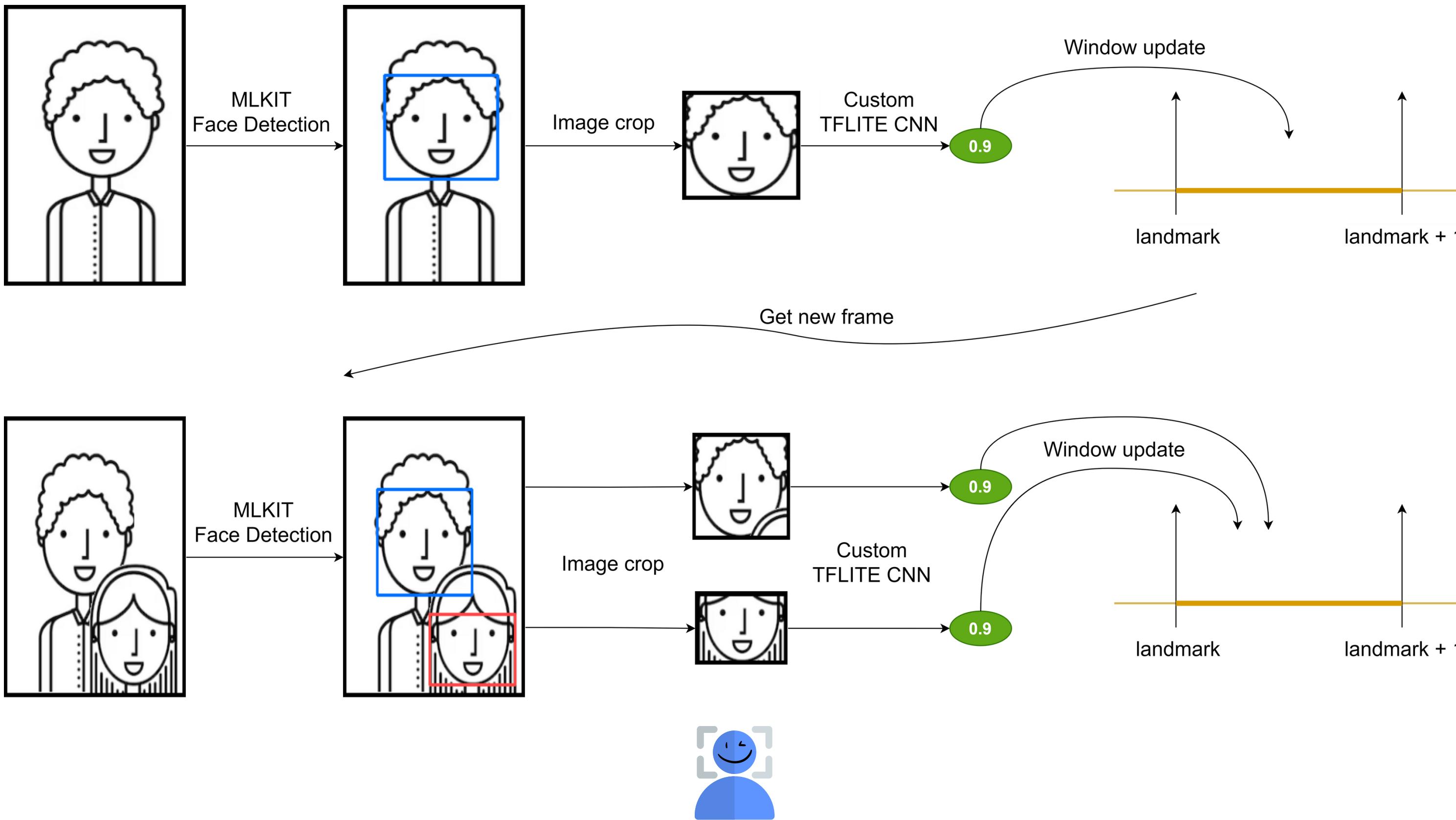
OVERALL PROCESSING PIPELINE

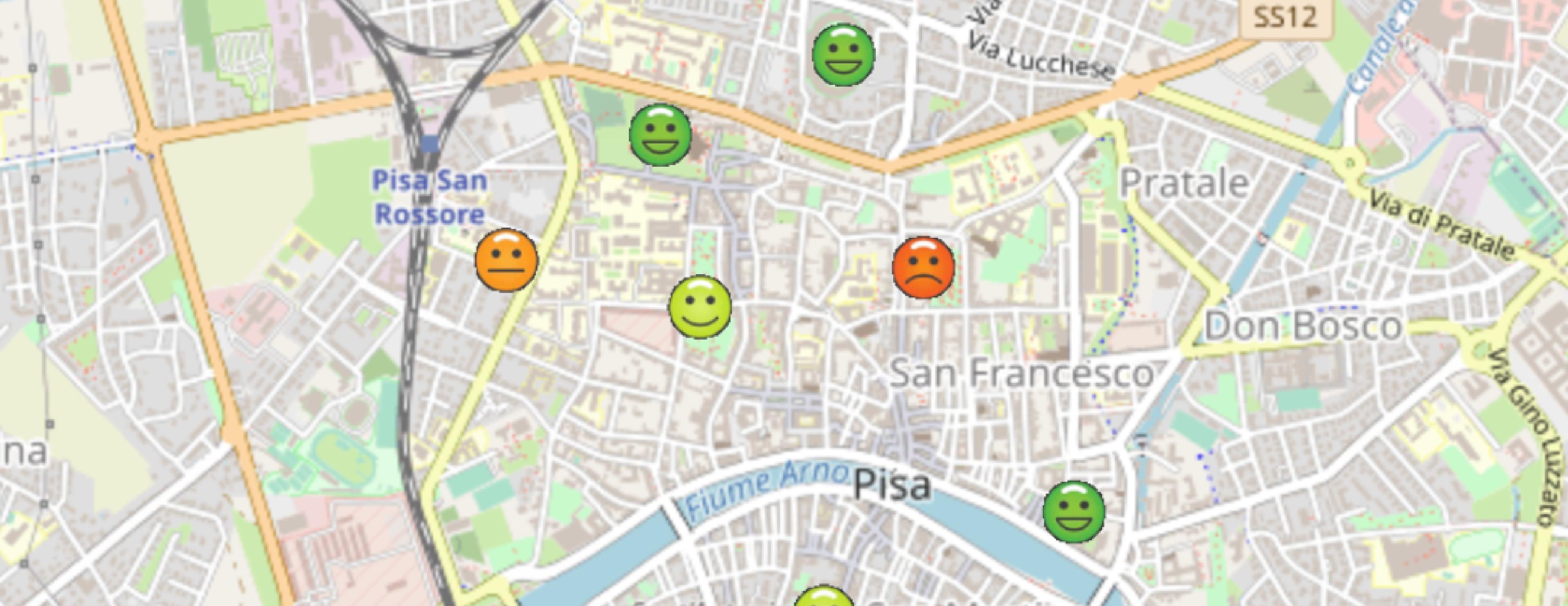
Landmark window model:

- When the app is being used (and if the user gave the necessary permissions), the processing of the video streaming starts.
- The predictions from the CNN are accumulated, along with the number of predictions.
- Each 10 seconds, the location of the smartphone is obtained and the average happiness in the last 10 seconds is computed (using the accumulators updated above). This information is sent to the DB.
- Accumulators are reset and a new window begins.



OVERALL PROCESSING PIPELINE





CLUSTERING



WHY CLUSTERING?

One of the fundamental components of our application is the clustering stage, aimed at identifying the happiest places within the city

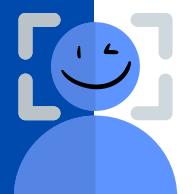
WHY DBSCAN?

Clusters emerge naturally from regions of high data density, allowing for the detection of clusters of varying sizes and shapes
Not need to specify the number of clusters to be found a priori

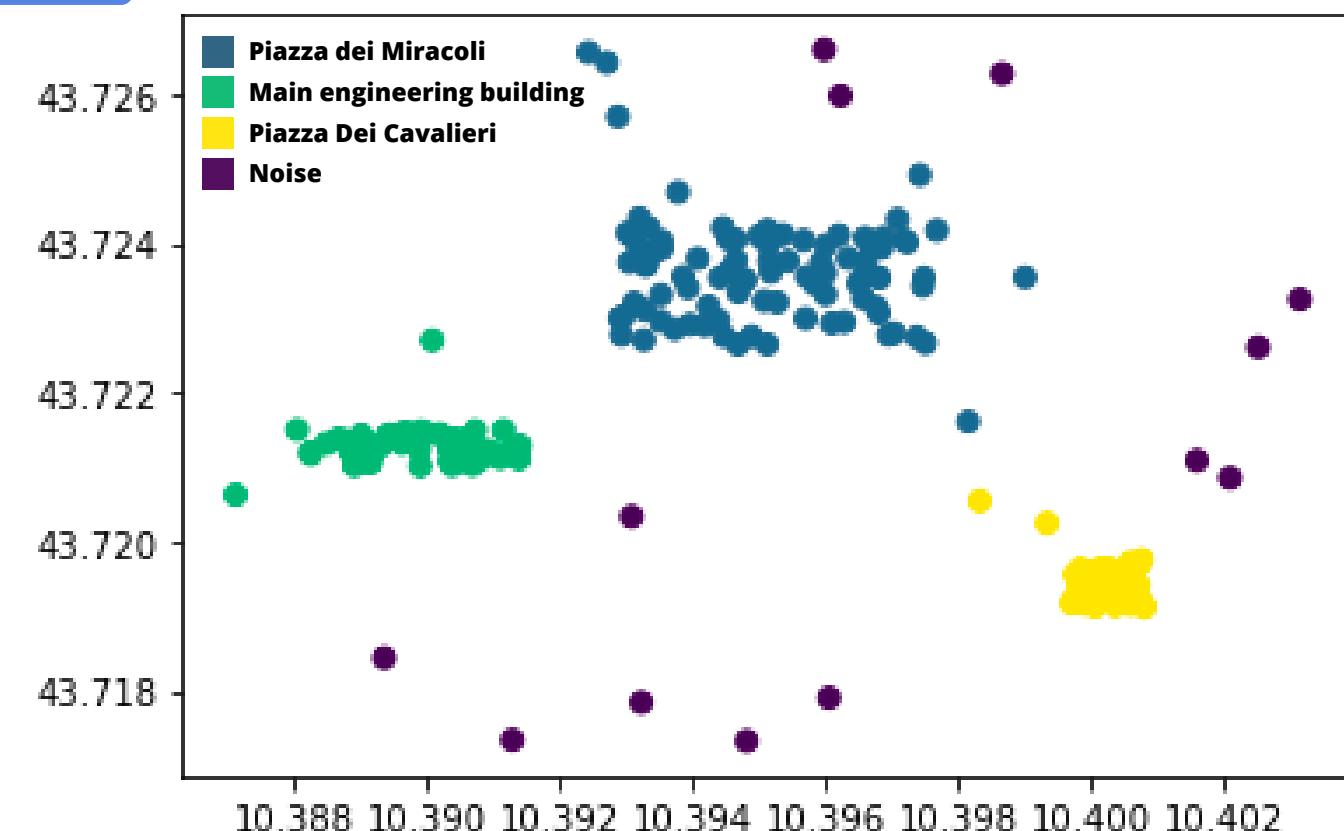
CLUSTERING

VALUATE BEST PARAMETERS

min_samples	eps	silhouette_coefficient
5.0	0.0015	0.636463
8.0	0.0015	0.632234
6.0	0.0015	0.632234
9.0	0.0015	0.632234
7.0	0.0015	0.632234
4.0	0.0015	0.631169
9.0	0.001	0.612066
6.0	0.001	0.612066
4.0	0.001	0.612066
7.0	0.001	0.612066



DBSCAN RESULTS

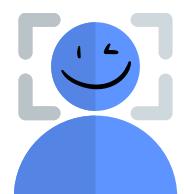




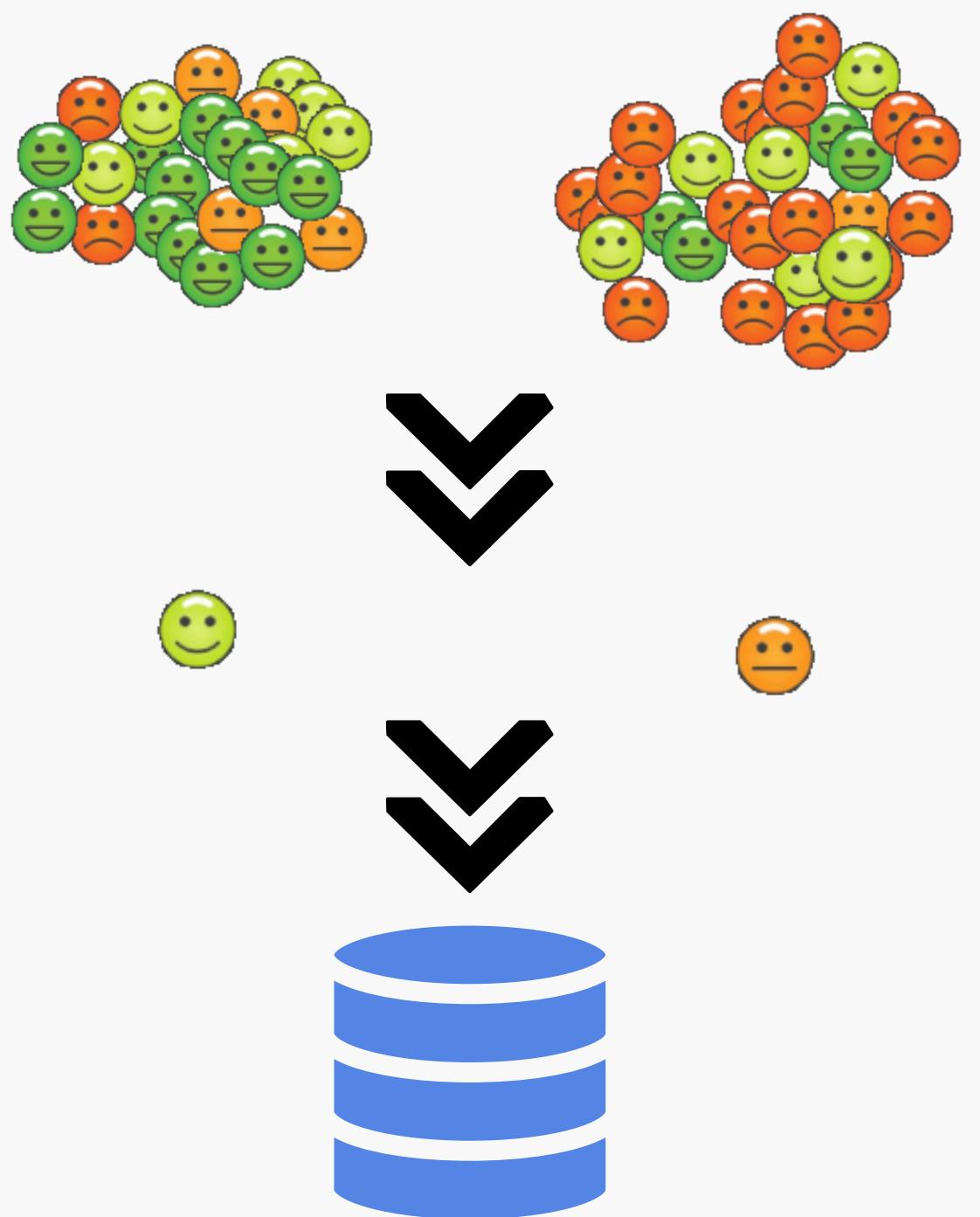
SMILE LIBRARY

import smile.clustering.DBSCAN

We use the SMILE library for
perform DBSCAN locally



ANDROID SERVICE

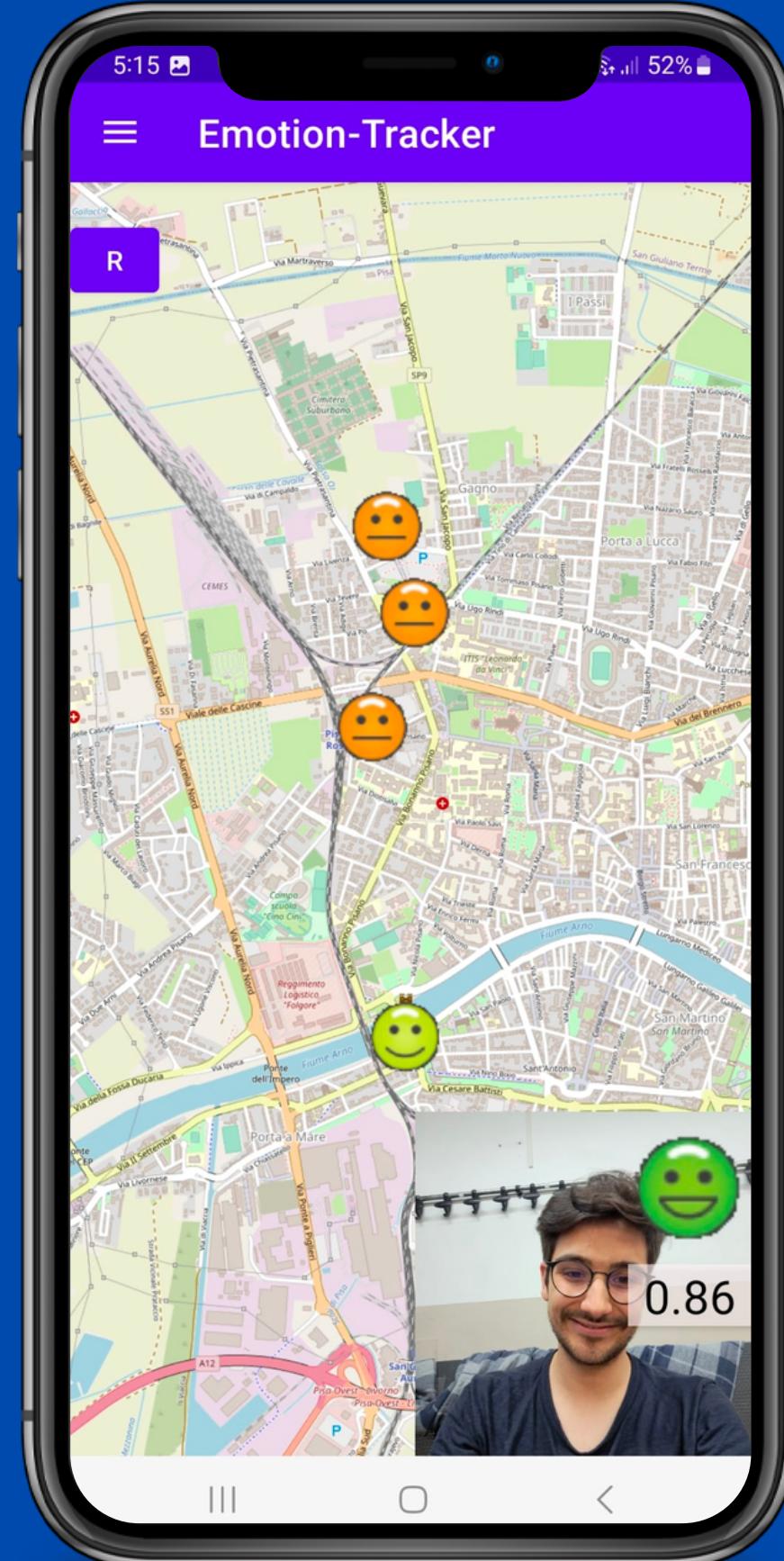


For the implementation of the cluster phase we leveraged a built-in Android service to perform the clustering process in the background.

Once the clusters are computed, they are summarized by a single representative point known as the centroid, that are the mean of all the points within the cluster.

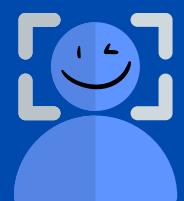
The service clears the collection of clusters for the current user and replaces it with the newly computed clusters.





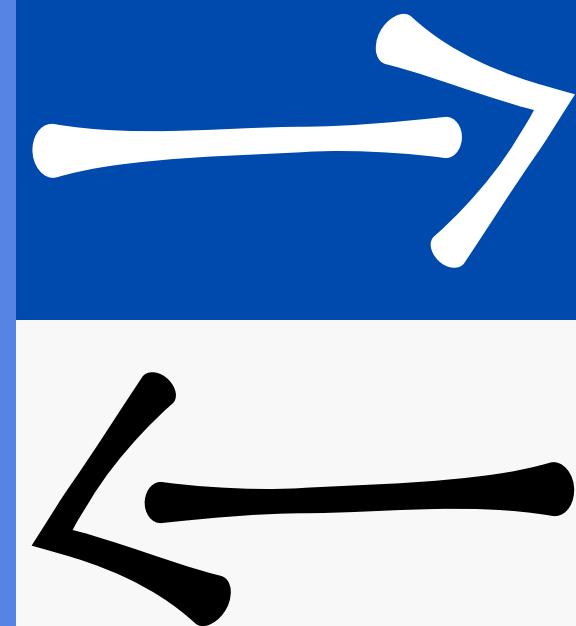
TEST

- We retrieve real world data, testing the app.
- We successfully uncovered clusters representing distinct areas within the city
- Even with a relatively small number of data points



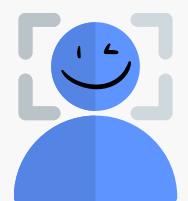
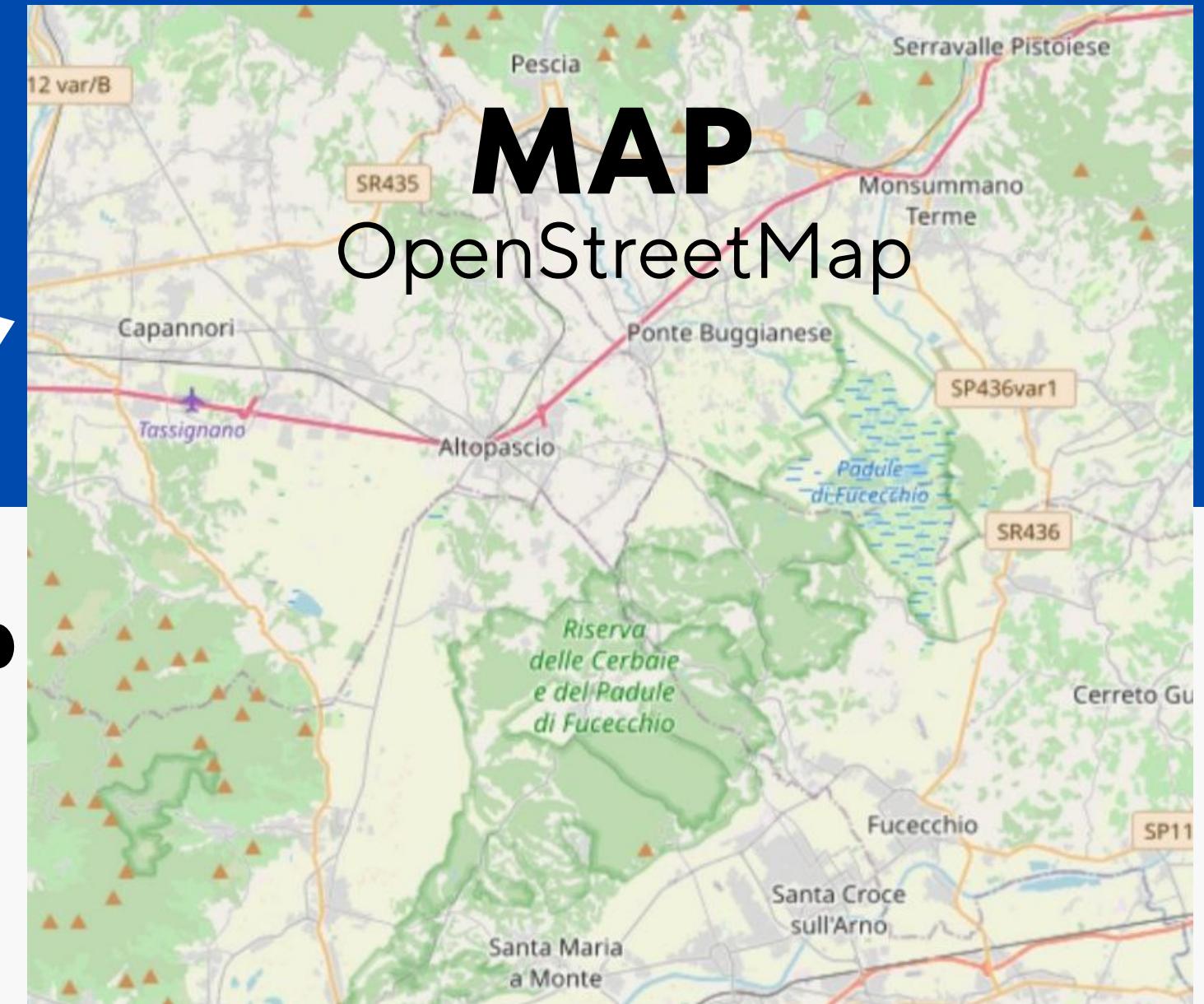
DATABASE

Firebase Realtime Database



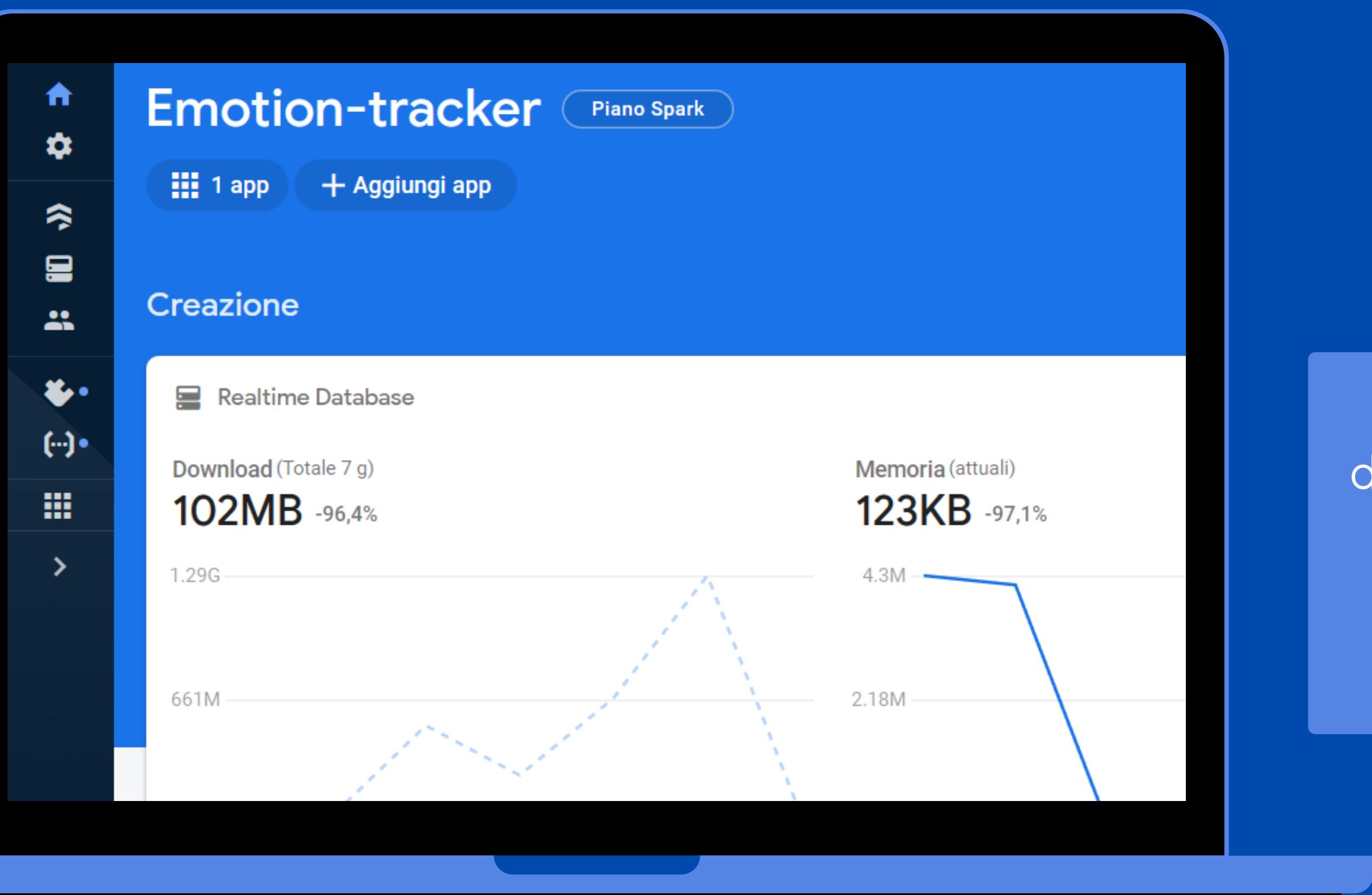
MAP

OpenStreetMap

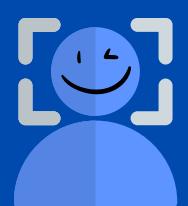


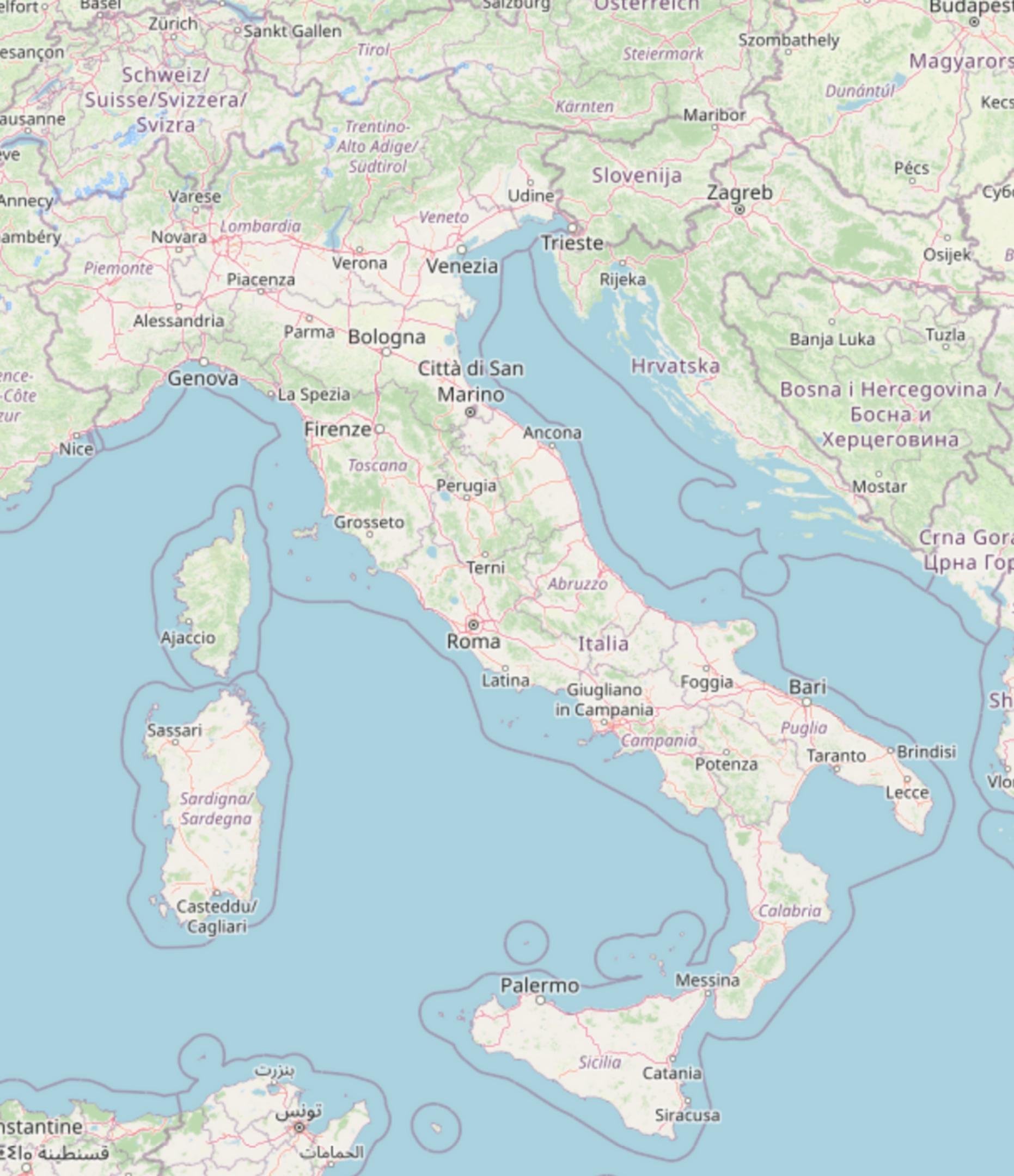
DATABASE

Firebase Realtime Database



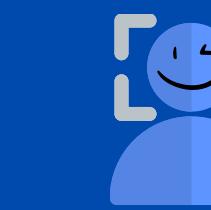
Firebase Realtime Database is a database hosted in the cloud. The data is stored as JSON and synchronized in real time with each connected client



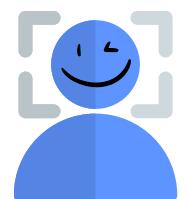
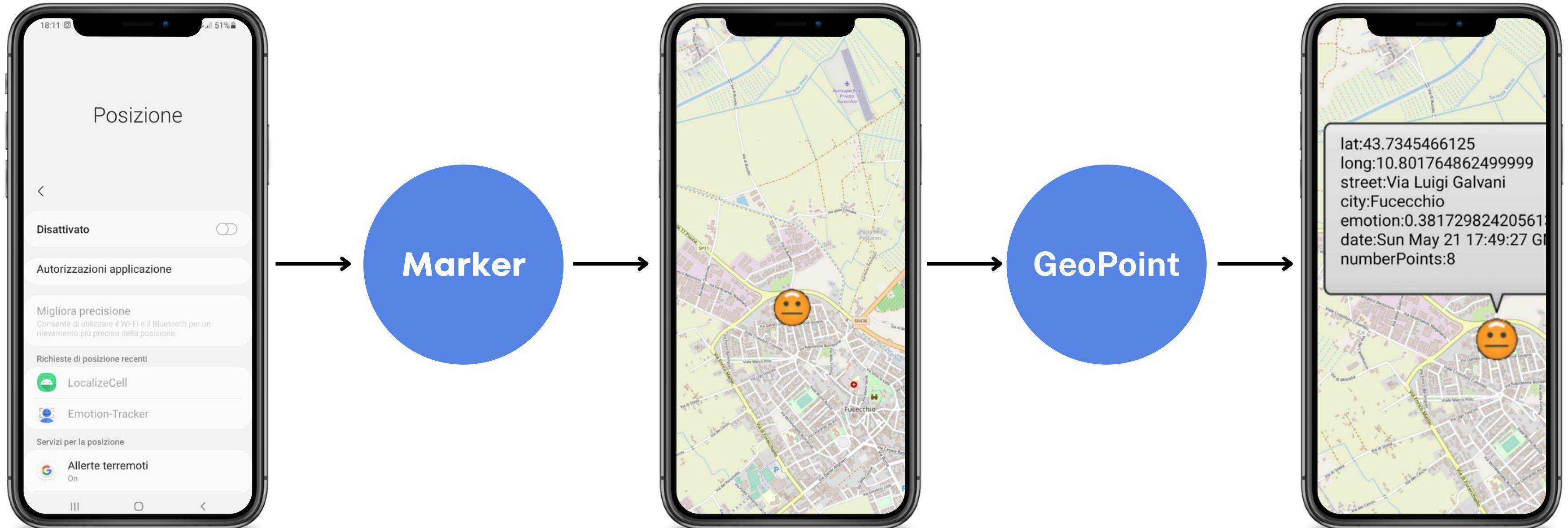


OPENSTREETMAP

OpenStreetMap is a free, editable map of the whole world that is being built by volunteers largely from scratch and released under an open-content license



SMARTPHONE'S GPS AND OSMDROID LIBRARY



FIREBASE COLLECTION

Users locations

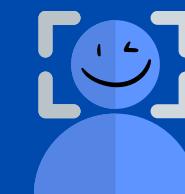
```
position_emotion
  -NVV-9I5G4UwLFVNhqu6
    city: "San Giuliano Terme"
    emotion: 0.09200962697013892
    latitude: 43.714107
    longitude: 10.4960106
    street: "Via G. Toniolo"
    timestamp: 1684164091335
    username: "tess"
```

Users

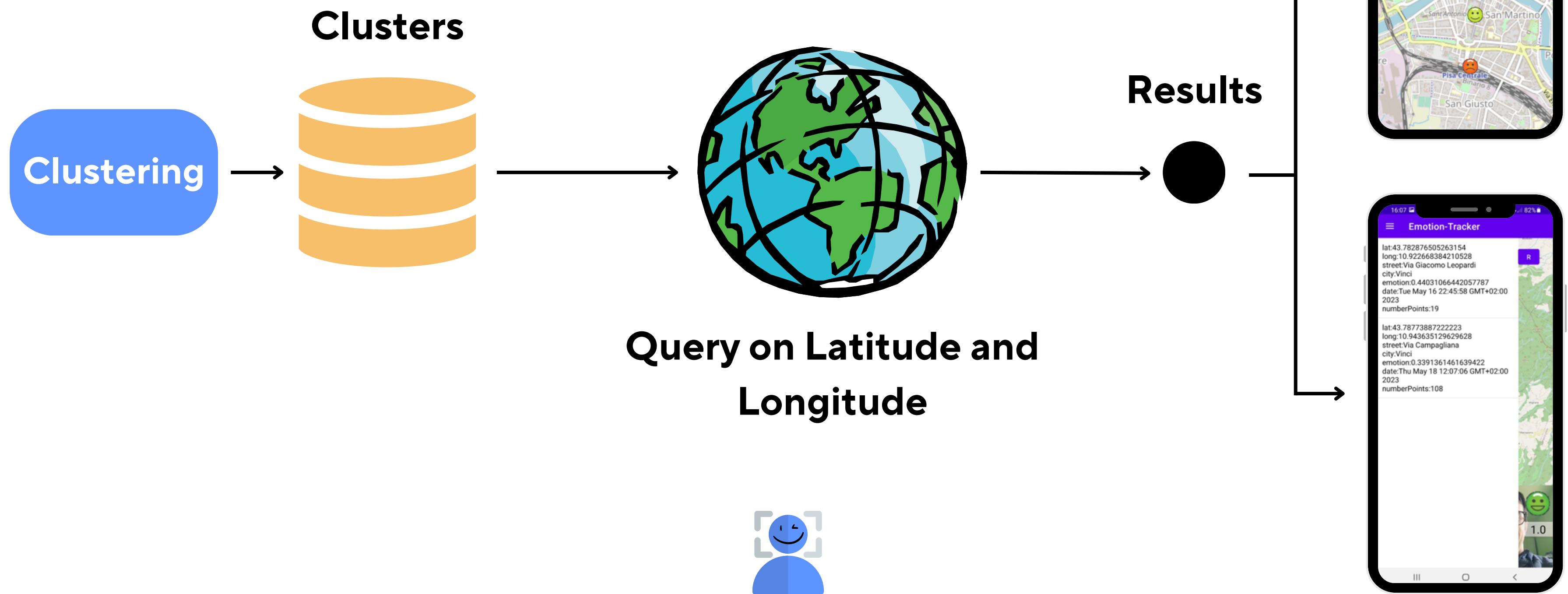
```
-NV0JkWkAD6lxpk2QdVW
  date_of_birth: "1999/10/10"
  password: "26ef33af3f99c756154f03b275c8ee4244ddd82cf0fb046dd84b7b9f0242e9fa"
  username: "fabio"
```

Clusters

```
clusters_matteo
  -NW-BizOweOKvRPNgn8W
    city: "San Giuliano Terme"
    emotion: 0.450482117240514
    latitude: 43.71410727631581
    longitude: 10.496014452631577
    numberOfPoints: 38
    street: "Via G. Toniolo"
    timestampDate: 1684483323683
```



MAP-DB INTERACTION



CONCLUSION

EMOTION RECOGNITION SYSTEM

In this work we presented a Binary CNN classifier that takes the pre-processed FER2013 dataset for creating a happiness map.

BINARY CLASSIFICATION PROBLEM

According to the CNN model we got a high accuracy on the test data showing that using a binary approach is a valuable option to analyze the happiness level.

FUTURE WORK

For the future we recommend using a multimodal approach to recognize facial emotions with the associated geographic locations.



QUESTION SESSION

If you have any questions, feel
free to ask

