MERLOT FACIAL RECOGNITION SUBSYSTEM

COS 301 PHASE 3 – GROUP 11

MEET THE TEAM

EMMA COETZER, HERBERT MAGAYA, KOKETSO MOLAWA, JUSTIN ROSS

EMMA COETZER

- Responsible for creating the databank to store the client's photographs used for the facial recognition. This was implemented as a JSON file and methods for interacting with the databank has also been supplied. Also responsible for the landing page and the PowerPoint presentation.

HERBERT MAGAYA

- Responsible for communicating with the rest of Merlot teams, most specifically with the authentication team. Also implemented the UPDATE function responsible for updating and inserting into the databank. Also set up hosting.

KOKETSO MOLAWA

- Responsible for integrating the facial recognition library provided by node.js, into the overall subsystem. This involves writing methods to also loop through the entire JSON databank for do the facial recognition and return the ClientID.

JUSTIN ROSS

- Responsible for writing the GET function responsible for interfacing with the databank module as well as providing the Authentication Subsystem with and interface to access the Facial Recognition. Also responsible for creating the LOGGING subsystem within the facial recognition subsystem.

WHAT WE DID

AN OVERVIEW OF THE TECHNOLOGIES THAT WE USED AND WHY WE USED THEM

NODE.JS

- After a discussion with the Authentication team, it was decided that Node.js would be the best technology to use in order to not only integrate all the subsystems together, but to also provide the necessary server and service required.

JSON

- JavaScript Object Notation has been used to store both the databank of client photographs as well as the logging file.
- JSON was used due to its universal implementation and because it fits in well with the subsystem that we have built.
- The databank JSON file is accessed by the databank module, which is then accessed by the node.js interface created by Justin.

UNIT TESTING

FOR THE MAIN NODE.JS GET FUNCTION

- The testing of the GetMethod is currently through the use of a node.js program that calls all the possible post queries using the request package. It sends a valid JSON file to the GetMethod and expects a valid JSON file in return. It is purely a test of how the method decides which function to perform depending on the JSON object sent.

GITHUB, WAFFLE.IO AND LANDING PAGE

LINKS TO THE TECHNOLOGIES THAT WE USED

- Project Management tool: https://waffle.io/CoolKayTwo/ATM-Facial-Recognition-System
- Landing Page: https://cs.up.ac.za/teams/pages/site_view/36
- GitHub Respositiory: https://github.com/CoolKayTwo/ATM-Facial-Recognition-System