

HUMAN COMPUTER INTERACTION

INTERMEDIA

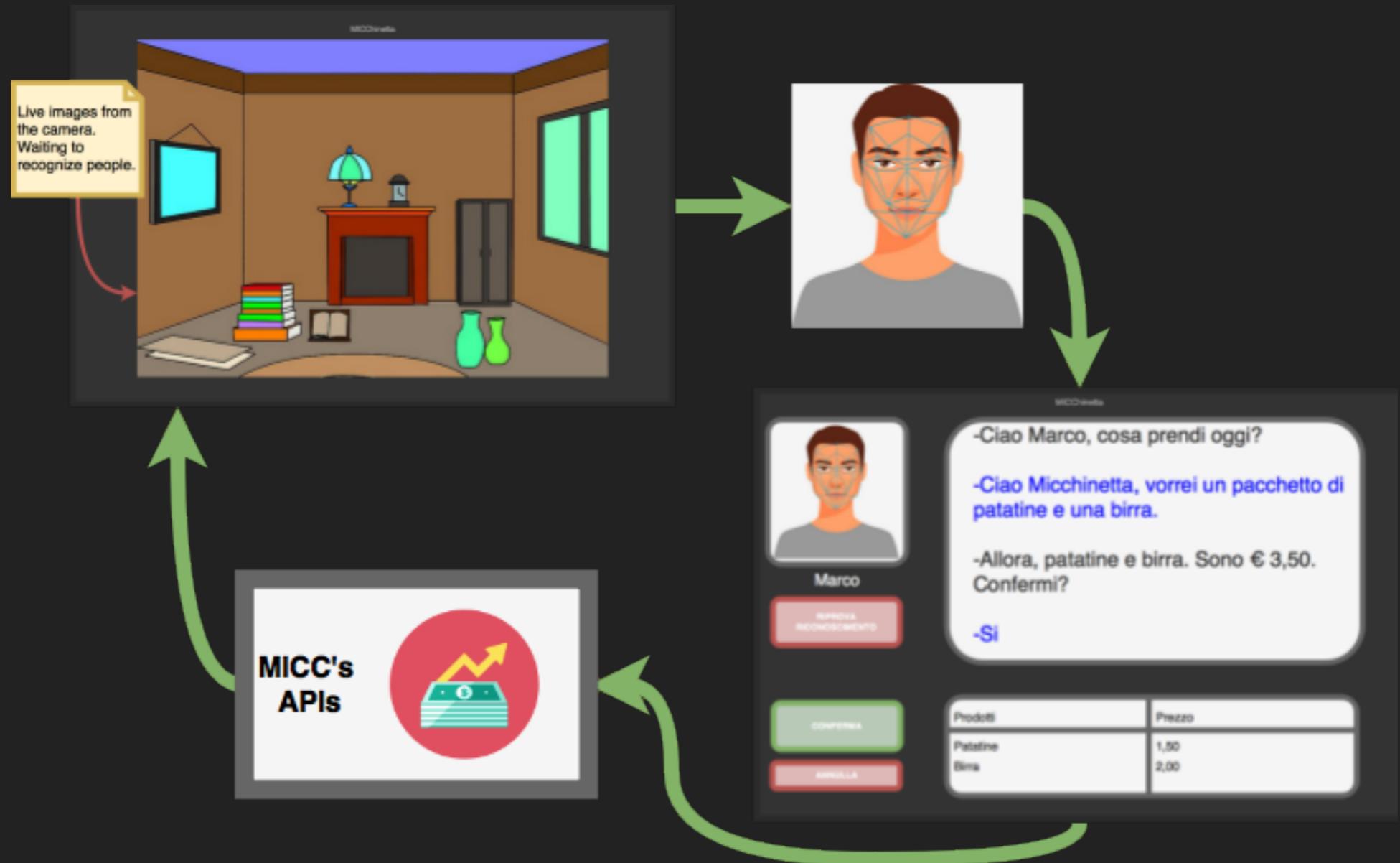
PROJECT ASSIGNMENT

ASSIGNMENT

- ▶ Deploy an application interface using both face and speech recognition to achieve a transaction between the user and a vending machine.
- ▶ With a webcam, the system should recognize known people. After the recognition, the application must start a conversation-like dialogue with the user:
 - ▶ **Micchinetta:** Good morning Andy, what do you need?
 - ▶ **Andy:** I would like three beers and a box of chips, thank you.
 - ▶ **Micchinetta:** Ok, it's three beers and a box of chips for € 3,80. Ok?
 - ▶ **Andy:** Ok!

GOALS

- ▶ Recognize people from a set of known faces
- ▶ Sustain a simple conversation with the user to understand his needs
- ▶ Use existing web API, provided by MICC, for transactions and accountability



TECHNOLOGIES

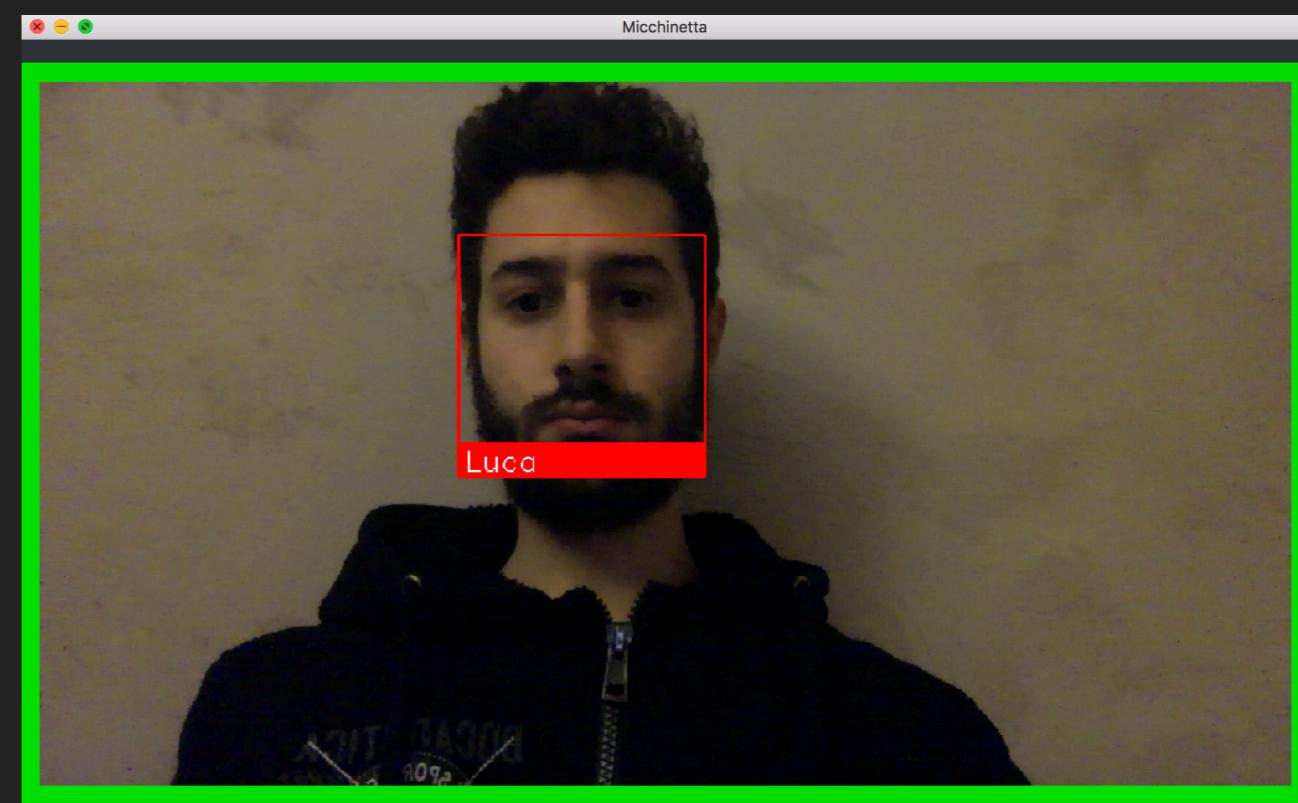
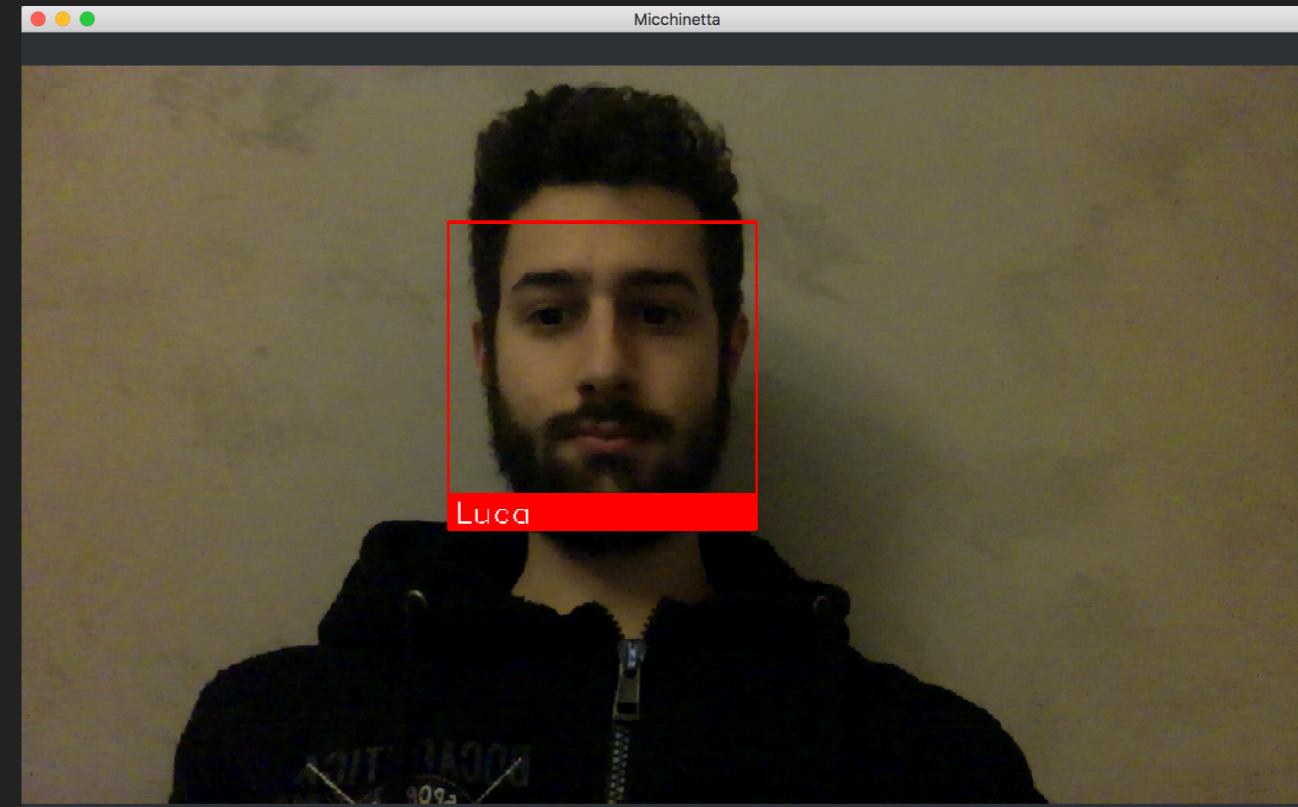
- ▶ The interface is developed with the PyQt Framework using Python.
- ▶ Face recognition makes use of a [Python package](#) and OpenCV for some elaborations and tasks.
- ▶ Speech recognition uses a [library](#) based on Google Cloud APIs.
- ▶ The app, to be able to speak, will use this other [Python package](#).
- ▶ The Identities along with the face descriptors are maintained in an SQLite database

The system must be connected to the internet to be able to use Google APIs and internal MICC APIs for accountability. Moreover, the software, to run a simple conversation, will need some capabilities in natural language processing. Many NLP software packages exist, but not many support Italian language along the English language. [TreeTagger](#) is the software chosen.

All of these software components have a significant computational cost and will require a decent machine to run on.

INTERFACE AND DESCRIPTION

- ▶ To provide a visual feedback to the user, the system implements a simple interface.
- ▶ The user will find the application in **recognition state**, with the interface showing a live view of the scene. Approaching the device, the user will appear on the screen and the recognition will start.
- ▶ If recognized (matched with one of the known faces) the system gives a visual feedback of the successful recognition
- ▶ If multiple faces are present int the scene, the closest (biggest) one is used for identity recognition.



INTERFACE AND DESCRIPTION

- ▶ The application will then change state, entering the **conversation state** represented in the next slide.
- ▶ Reference image of the user from the database will be showed and the conversation shall begin.
- ▶ If the identity matched is wrong, the user can tell the system during the conversation and the system will apologize and retry the recognition.
- ▶ While speaking, the conversation and the “bill” will be displayed and updated. If the user agrees to the final bill, the machine will commit the transaction and will change its state to idle.

Conversation State

Micchinetta

Ciao Luca cosa ti serve?

una coca-cola e un acqua

Quindi vuoi 1 coca-cola 1 acqua al prezzo di 2.6 € ? Dì ok per addebitare, o continua a modificare la richiesta

un arachidi

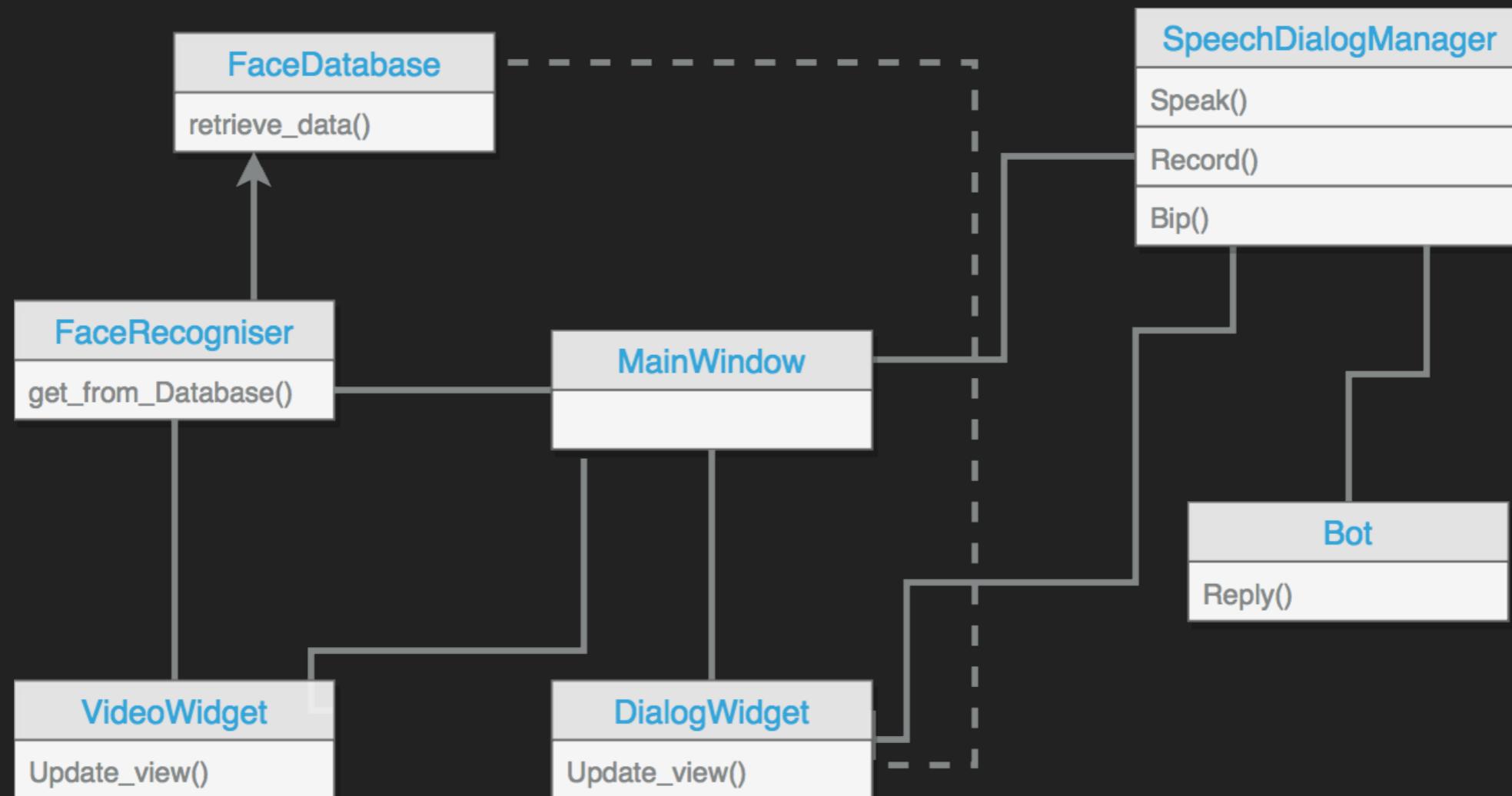
Quindi vuoi 1 arachidi 1 coca-cola 1 acqua al prezzo di 4.6 € ? Dì ok per addebitare, o continua a modificare la richiesta

Luca

Visual and Audio feedback

	prodotto	prezzo
1	arachidi X1	2.0
2	coca-cola X1	1.6
3	acqua X1	1.0

MODEL VIEW CONTROLLER

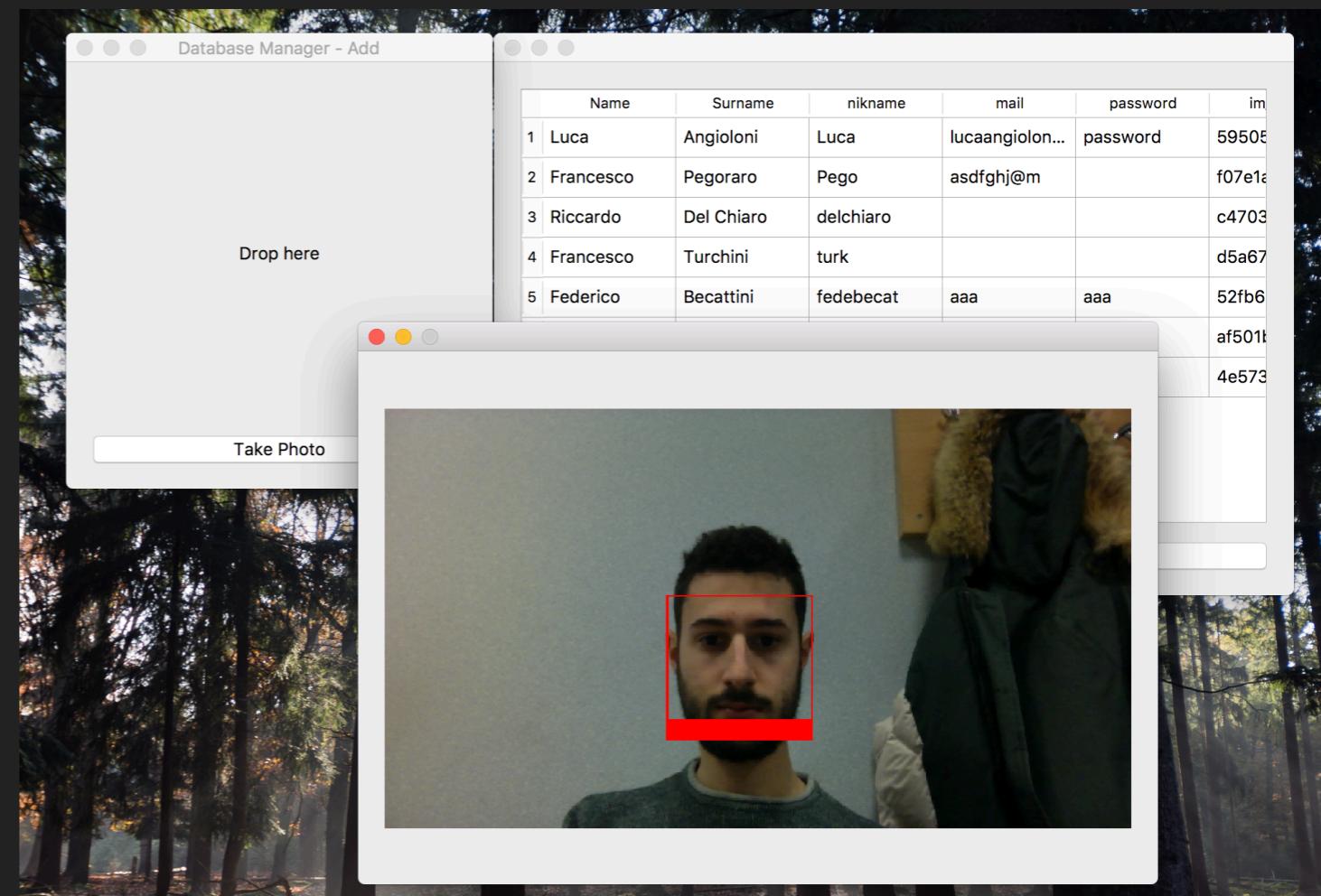


► Separation of concerns:

- controllers: **VideoWidget**, **DialogWidget**
- View: Qt Widgets
- Model: **FaceDatabase** and **Bot**

DATABASE MANAGER APP

- ▶ Along the main app, an utility app has been developed to help this system admins to manage the Faces database.
- ▶ The script DatabaseManager.py starts this graphical application that allows to add, remove and modify users identities (and the credentials for the Transactions APIs).

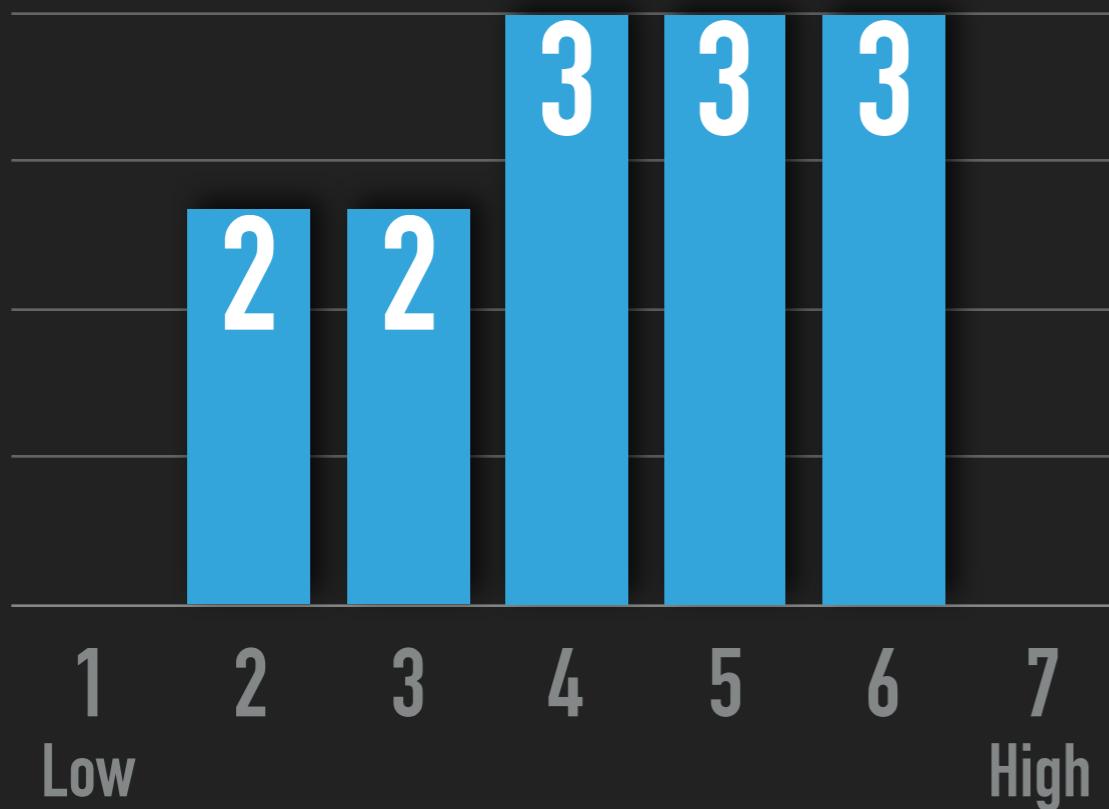


USABILITY TESTS

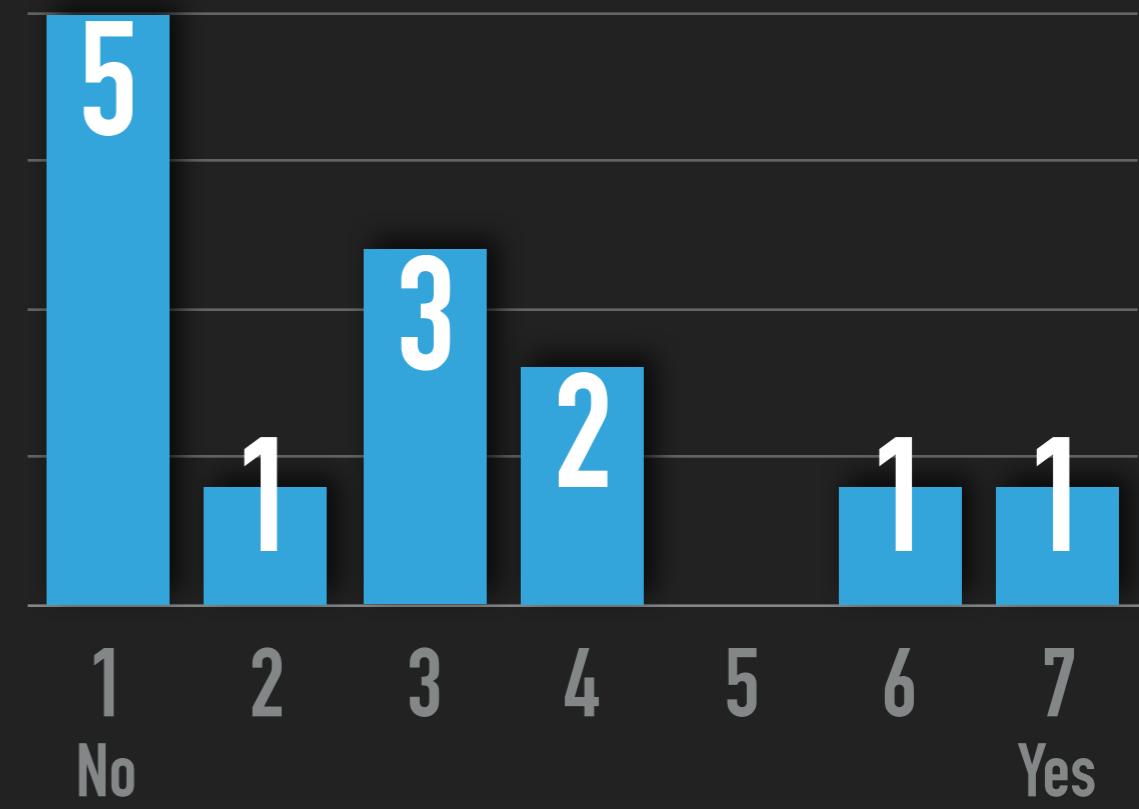
► Tasks:

- ▶ *To be recognized*: Present yourself in front of the machine and let it recognize you.
- ▶ *Wrong recognition*: Simulate a wrong recognition and explain it to the system.
- ▶ *Ask for a product*: You are hungry or thirsty, so you ask for a product.
- ▶ *Ask for more products*: You are very hungry or thirsty, so you ask for more than one product at once.
- ▶ *Remove a product*: You changed your mind, remove a single product.
- ▶ *Remove more products*: You changed your mind, remove many products.
- ▶ *Confirm the purchase*: You are finished, agree to the transaction.

I FOUND THE REPLY TIME TOO HIGH.

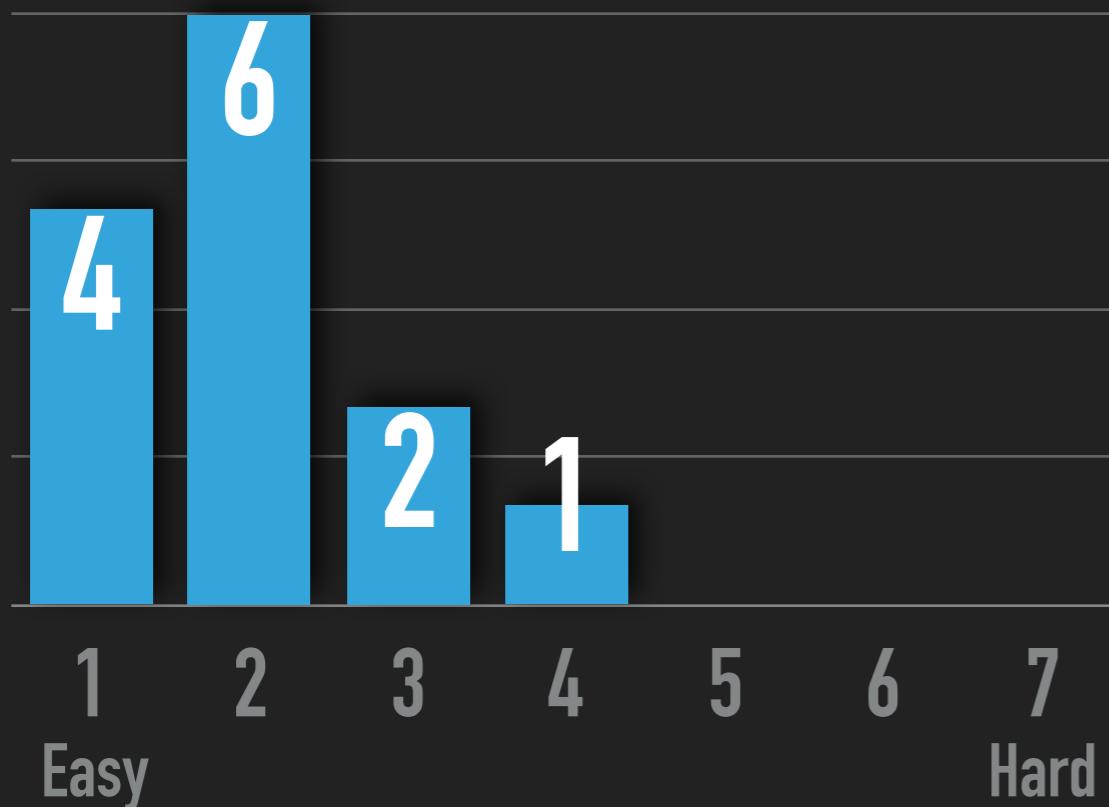


I PREFER USING THE MOBILE APPLICATION
INSTEAD OF THIS SYSTEM.

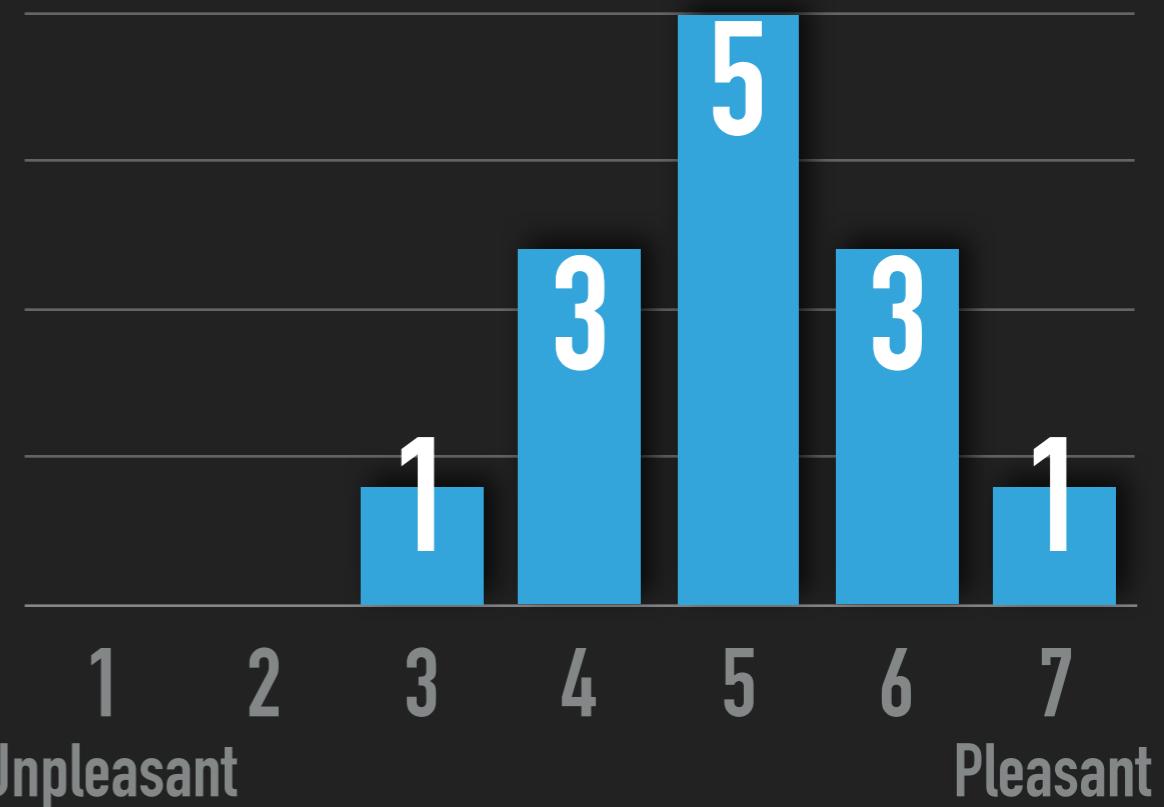


- ▶ User's opinions vary when asked if the latency between MICChinetta's replies are too high. This may be linked to the fact that the reply time varies with respect to Internet connection.
- ▶ The majority of users prefers to use the VUI instead of the mobile App.

IT IS HARD TO USE IT.



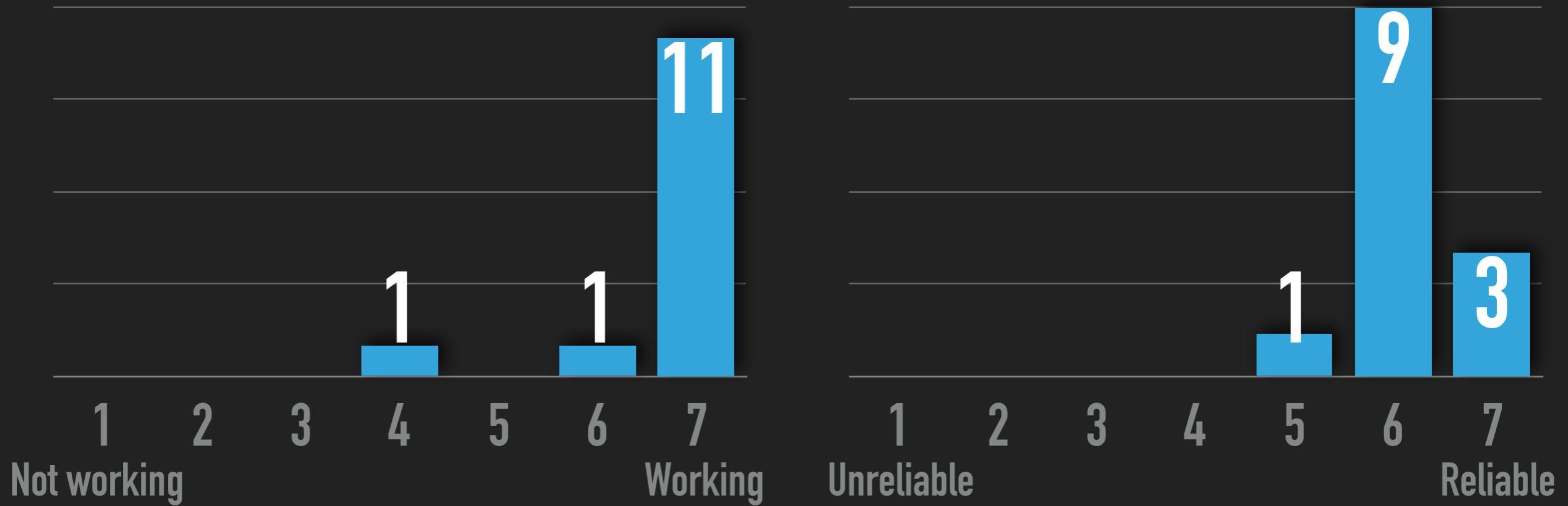
THE GRAPHIC INTERFACE IS PLEASANT.



- ▶ Most of the users found the system to be self explanatory, this is indeed crucial using a never-seen application.
- ▶ Users agree about the pleasantness of the Graphical Interface

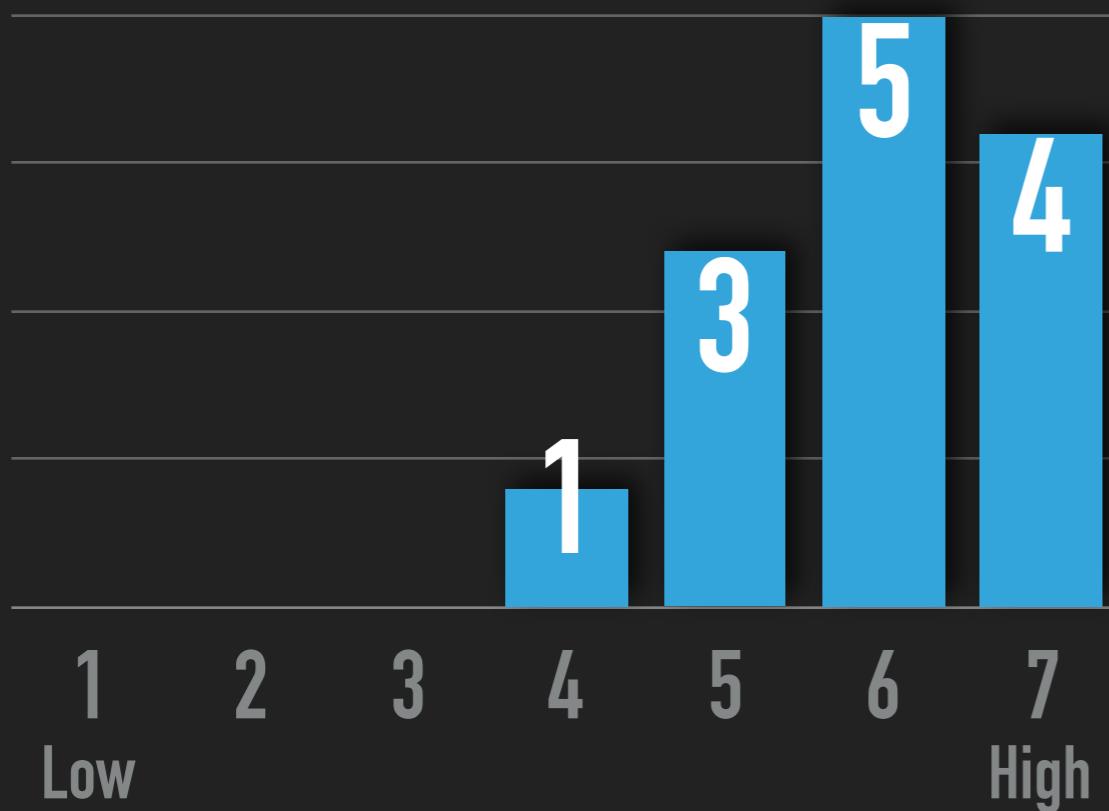
THE FACIAL RECOGNITION WORKS CORRECTLY.

I CAN TRUST THE SYSTEM.

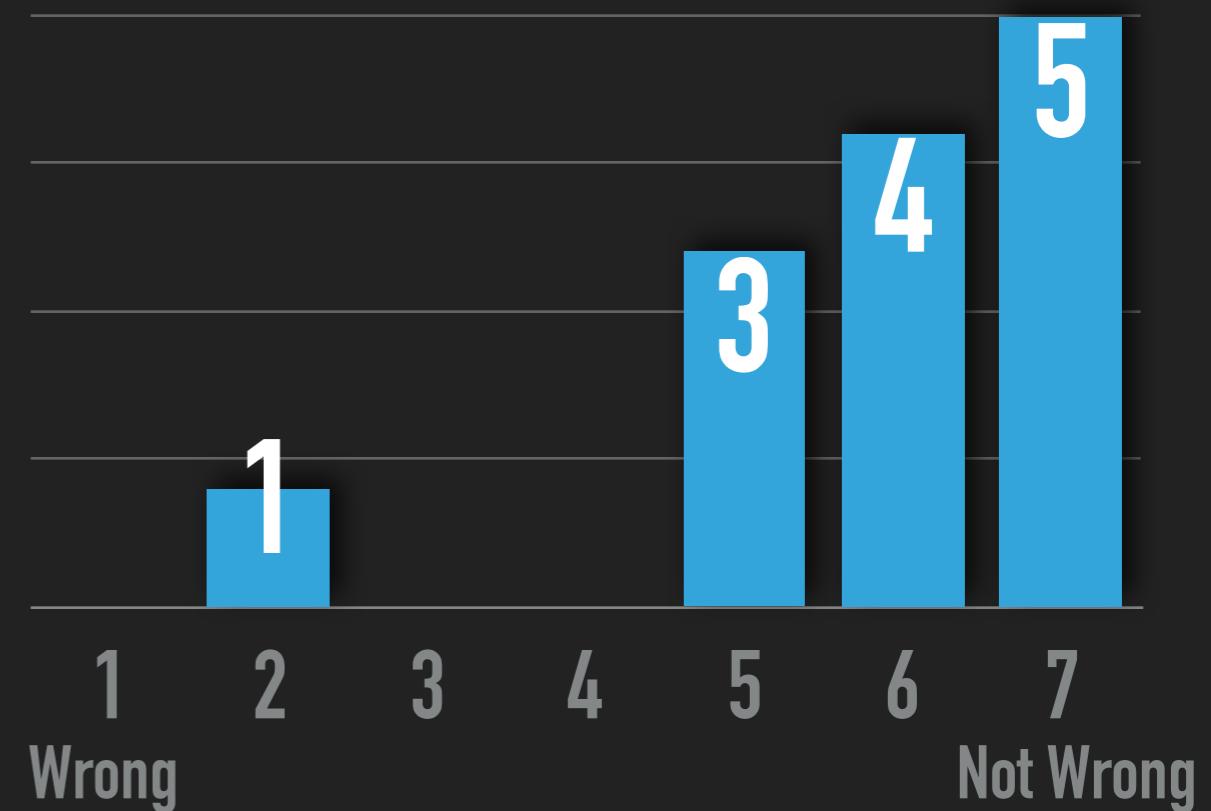


- ▶ Face recognition was definitely working.
- ▶ This is a factor in the trust placed by the user, reported in the relative chart.

MICCHINETTA CAN UNDERSTAND MY REQUEST.



MICCHINETTA'S REPLYS ARE WRONG.



- ▶ These plots are a good demonstration of the conversation skills of the system; it can understand what users ask and answers accordingly.

CONCLUSIONS

- ▶ VUIs are growing fast in popularity for the most various scenarios, like car driving, mobile phones and many more.
- ▶ We discovered the possibilities for augmenting human-computer interactions provided by a prototype like MICChinetta, showing pros and cons of this new approach.
- ▶ Our final study confirmed that MICChinetta, but more in general VUIs, can do almost everything a GUI can do, not reducing tasks times, but giving the users different possibilities and experiences without using arms or hands.
- ▶ We believe that pushing ourselves in this direction and improving these technologies, they will efficiently substitute standard interfaces where less and less interaction and effort from humans is required