easy-talk

A device to facilitate communication of children with difficulties, powered by Artificial Intelligence

View the Project on GitHub marconicivitavecchia/easy-talk

Easy Talk - Hello Frog

Repository Easy Talk - Hello Frog, a project born inside the program "Ambizione Italia" sponsored by Microsoft and powered by Fondazione Mondo Digitale.

Brief

We propose a technology to facilitate learning of social skills and educational activities for both verbal and non-verbal language in order to help people with communication difficulties. Our proposed technology is composed by a portable device, with a small color screen, a microphone, a speaker and two cameras, that act as a facilitator in the communication between a person with difficulties and other people. The device "listen to" the conversation between these people and figure out their emotional state. If their emotional state diverges, probably due to a misunderstanding of emotional states, the device enters into action trying to help the conversation and resolve the misunderstanding.

Design

Please see the design README.md

Code

Please see the code README.md

Press Review [ITA]

- Video interview
- Post on the Marconi website

Fondazione Mondo Digitale:

- Meeting with Satya Nadella
- Intelligenza inclusiva

Microsoft Italia:

- MS Innovation Summit

Newspapers and Online Magazines:

- Il Corriere della Sera, May 30, 2019
- Il Sole 24 Ore, May 31, 2019
- Punto a Capo online, June 2, 2019

---

easy-talk

A device to facilitate communication of children with difficulties, powered by Artificial Intelligence

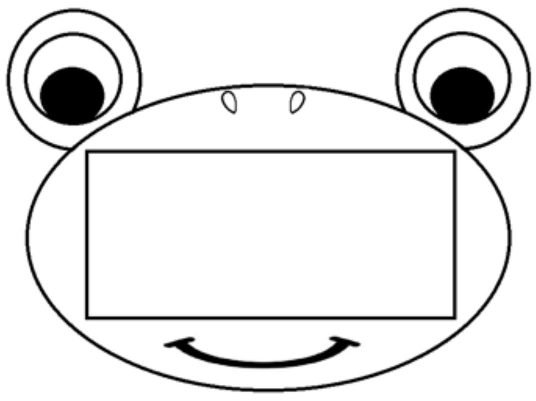
View the Project on GitHub marconicivitavecchia/easy-talk

Prototype

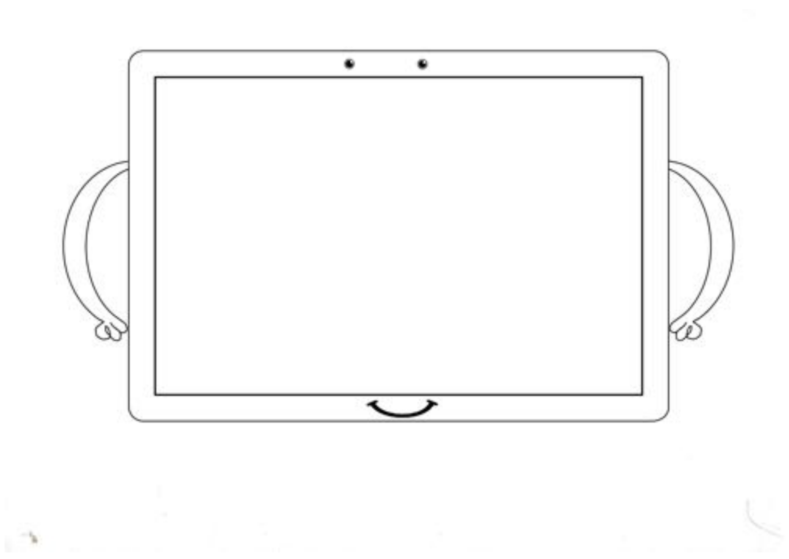
Rationale

We propose two shapes to target different audience. Both shapes of the device have been designed for young users. The goal is to help the user to perceive a possible misunderstanding through a “friendly-funny looking” device.

FROG: it addresses the younger age target, that are children aged between 4 to 8 years old. The choice of this subject is due to the fact that his funny looking and it may be accepted easily between males and females.



IDK: is aimed at the “elder” age range of the target audience, that the children aged between 8 to 11 years old. Its shape has been designed to be more anonymous in order to avoid embarrassment in the subject with communication difficulties despite having a playful apperance; besides the character’s “arms” act like a handle which allow a better portability even when the device is not in use. (this device will be prototyped in the future)

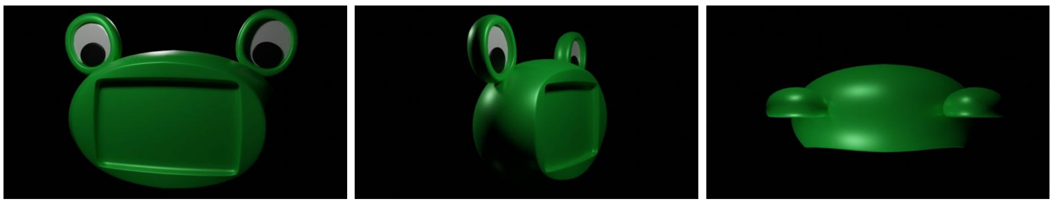


We choose to start prototyping the first device. The second device may be developed in the future.

3D model

Preliminary design in Blender

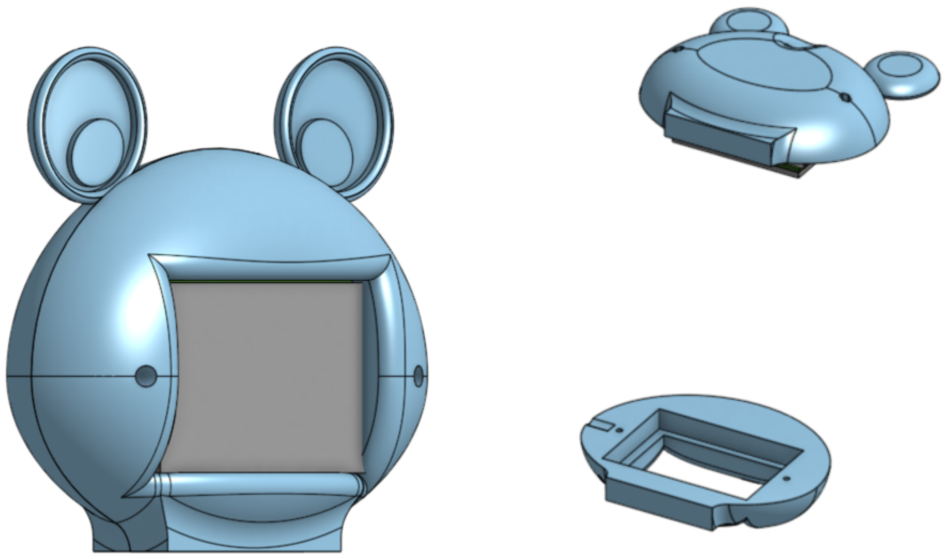
We started with a rapid prototyping in Blender.



Then we add a bit of fancy rendering to give a better idea.

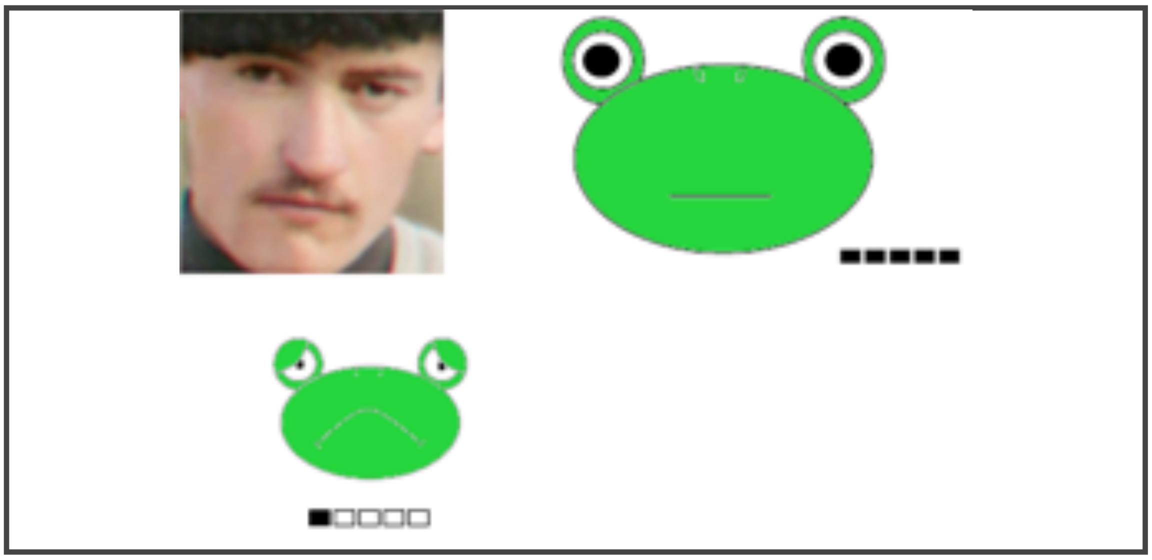
3D printing with OnShape

We moved to OnShape with this document to prepare the file for 3D printing.



User Interface

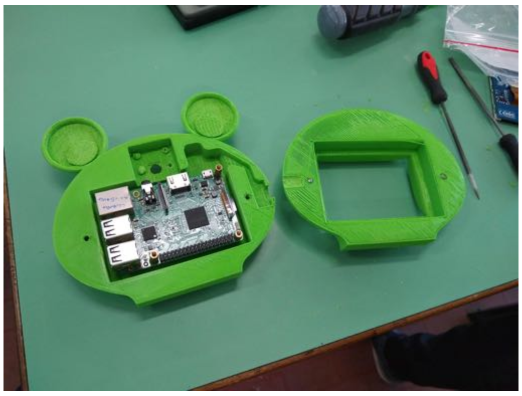
Here a proposal for the user interface.

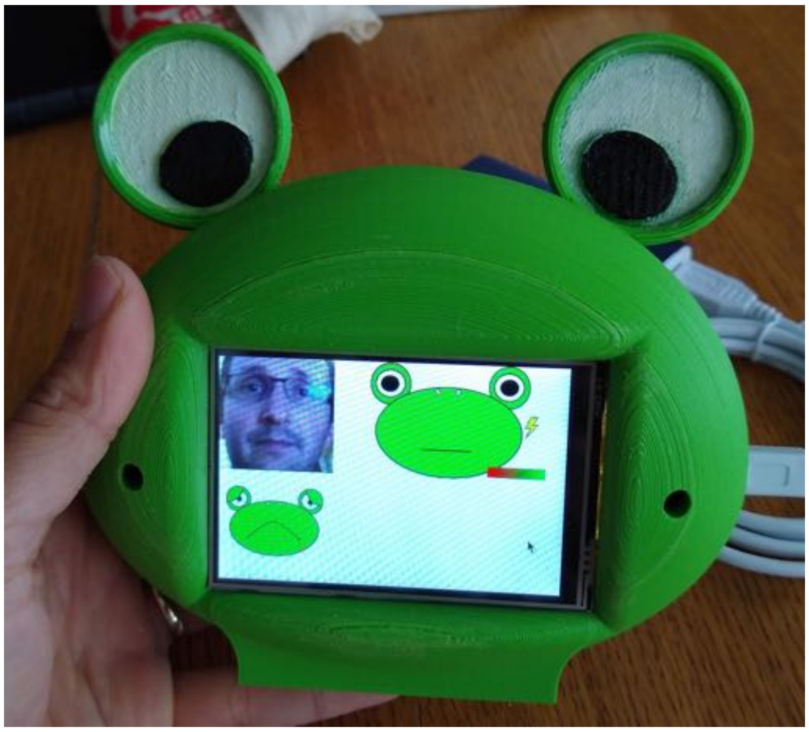


The idea is to show the most likely expression on the top, close to the crop of the snapshot from the camera. In the lower part, there are other faces in order of emotion probability, as received from Azure Cognitive Services.

Printed model

It works!





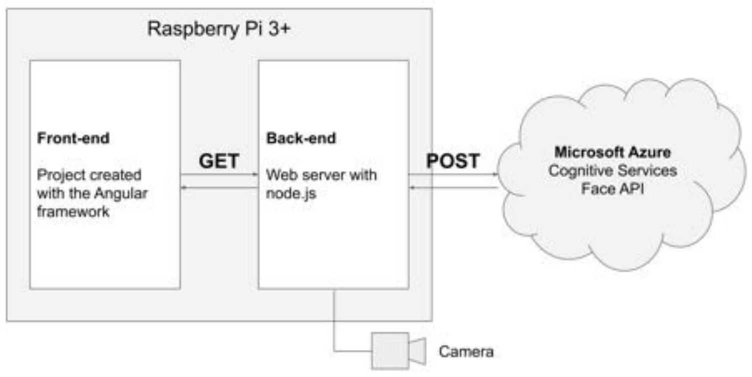
easy-talk

A device to facilitate communication of children with difficulties, powered by Artificial Intelligence

View the Project on GitHub marconicivitavecchia/easy-talk

Architecture

We used a client-server approach for this project, to easily test the code, to share tasks among the team and to support future improvements.


Raspberry Pi 3+
Front-end
Project created with the Angular framework
GET
Back-end
Web server with node.js
POST
Microsoft Azure Cognitive Services Face API
Camera

Back-end

We used Node.js as language for the back-end because JavaScript is a language already studied by the students of the forth year and for similiarity with the front-end.

We used:

- express for the web server
- pi-camera-connect to read pictures from the Raspberry camera

Front-end

We used Angular to develop the front-end. This is beyond the common knowledge of students of the forth year, but since there are several parts of the UI that are repeated, we thought that the effort of studying and using a new tool based on the concept of “components” paid off.

Microsoft Azure Cloud Services

We used the Microsoft Azure Cognitive Services to analyze and recognize the emotions in the picture.

We didn’t make a comparative benchmark with other similiar services, but we found the service well documented and with a good accuracy. Overall we are very satisfied with it.

Getting Started

Back-end

What you need:

- Rasberry PI, tested on 3 and 3+
- Camera, tested with original camera v2.1

First of all, connect to the Raspberry via SSH or keyboard and mouse and open a Terminal.

Install Node.js

To install node.js, go to the download page and select ARMv7:

```
# Last stable version at 2019-05-27
https://nodejs.org/dist/v10.15.3/node-v10.15.3-linux-armv7
tar -xzf node-v10.15.3-linux-armv7l.tar.xz
cd node-v10.15.3-linux-armv7l
sudo cp -R * /usr/local/
```

Then test that everything is OK:

```
node -v
npm -v
```

If this doesn’t work… please submit an issue!

Download this repository

Just download this repo:

```
git clone http://github.com/marconicivitavecchia/microsoft
```

Configure environment

Before running the server, we need to add Azure key that cannot be shared in GitHub.

Navigate to the server folder and create the .env file:

```
cd microsoft-ai/code/server
touch .env
```

Edit the file with nano, `vim` or whatever and add the following line, with an API Key:

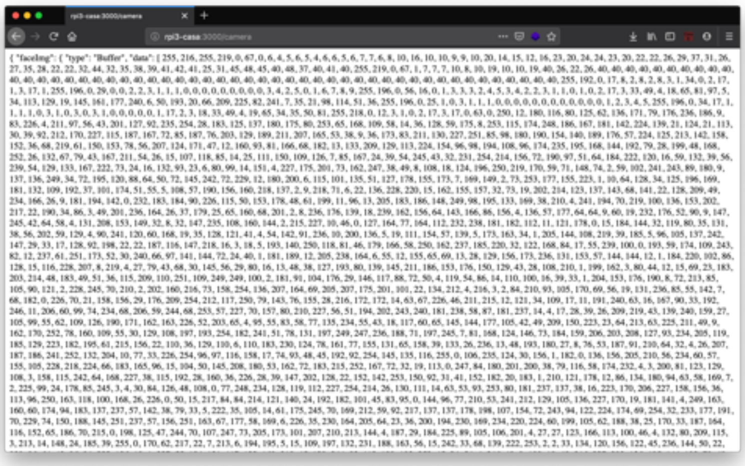
```
API_KEY='your-azure-api-key'
```

Run the server

Navigate into the server folder and run the server:

```
cd microsoft-ai/code/server
# Install dependencies
npm install
# Run the server
node main.js
```

Now with any browser, open a new tab and go to the url `/camera`:



```
{ "faceImg": { "type": "Buffer", "data": [ 255, 216, 255, 219, 0, 67, 0, 6, 4, 5, 6, 5, 4, 6, 6, 5, 6, 7, 7, 6, 8, 10, 16, 10, 10, 9, 9, 10, 20, 14, 15, 12, 16, 23, 20, 24, 24, 23, 20, 22, 22, 26, 29, 37, 31, 26,
27, 35, 28, 22, 22, 32, 44, 32, 35, 38, 39, 41, 42, 41, 25, 31, 45, 48, 45, 40, 48, 37, 40, 41, 40, 255, 219, 0, 67, 1, 7, 7, 7, 10, 8, 10, 19, 10, 10, 19, 40, 26, 22, 26, 40, 40, 40, 40, 40, 40, 40, 40, 40, 40,
40, 40, 40, 40, 40, 40, 40, 40, 40, 40, 40, 40, 40, 40, 40, 40, 40, 40, 40, 40, 40, 40, 40, 40, 40, 40, 40, 40, 40, 40, 40, 40, 40, 40, 40, 40, 40, 40, 40, 40, 40, 40, 40, 40, 40, 255, 192, 0, 17, 8, 2, 8, 2, 8, 3, 1, 34, 0, 2, 17,
1, 3, 17, 1, 255, 196, 0, 29, 0, 0, 2, 2, 3, 1, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 3, 4, 2, 5, 0, 1, 6, 7, 8, 9, 255, 196, 0, 56, 16, 0, 1, 3, 3, 3, 2, 4, 5, 3, 4, 2, 2, 3, 1, 1, 0, 1, 0, 2, 17, 3, 33, 49, 4, 18, 65, 81, 97, 5,
34, 113, 129, 19, 145, 161, 177, 240, 6, 50, 193, 20, 66, 209, 225, 82, 241, 7, 35, 21, 98, 114, 51, 36, 255, 196, 0, 25, 1, 0, 3, 1, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 2, 3, 4, 5, 255, 196, 0, 34, 17, 1,
1, 1, 1, 0, 3, 1, 0, 3, 0, 3, 1, 0, 0, 0, 0, 0, 1, 17, 2, 3, 18, 33, 49, 4, 19, 65, 34, 35, 50, 81, 255, 218, 0, 12, 3, 1, 0, 2, 17, 3, 17, 0, 63, 0, 250, 12, 180, 116, 80, 125, 62, 136, 171, 79, 176, 236, 186, 9,
83, 226, 4, 211, 97, 56, 43, 201, 127, 92, 235, 254, 28, 183, 125, 137, 180, 175, 80, 253, 65, 168, 109, 58, 14, 36, 128, 59, 175, 8, 253, 115, 174, 248, 186, 167, 181, 142, 224, 139, 21, 124, 21, 113,
30, 39, 92, 212, 170, 227, 115, 187, 167, 72, 85, 187, 76, 203, 129, 189, 211, 207, 165, 53, 38, 9, 36, 173, 83, 211, 130, 227, 251, 85, 98, 180, 190, 154, 140, 189, 176, 57, 224, 125, 213, 142, 158,
152, 36, 68, 219, 61, 150, 153, 78, 56, 207, 124, 171, 47, 12, 160, 93, 81, 166, 68, 182, 13, 133, 209, 129, 113, 224, 154, 96, 98, 194, 108, 96, 174, 235, 195, 168, 144, 192, 79, 28, 199, 48, 168,
252, 26, 132, 67, 79, 43, 167, 211, 54, 26, 15, 107, 118, 85, 14, 25, 111, 150, 109, 126, 7, 85, 167, 24, 39, 54, 245, 43, 32, 231, 254, 214, 156, 72, 190, 97, 51, 64, 184, 222, 120, 16, 59, 132, 39, 56,
239, 54, 129, 133, 167, 222, 73, 24, 16, 132, 93, 23, 6, 80, 99, 14, 151, 4, 227, 175, 201, 73, 162, 247, 38, 49, 8, 108, 18, 124, 196, 250, 219, 170, 59, 71, 148, 74, 2, 59, 102, 241, 243, 89, 180, 9,
137, 136, 249, 34, 72, 195, 120, 88, 64, 50, 72, 145, 242, 72, 229, 12, 180, 200, 6, 115, 101, 135, 51, 127, 178, 155, 173, 7, 169, 149, 2, 73, 253, 177, 155, 223, 1, 10, 64, 128, 34, 125, 196, 169,
181, 132, 109, 192, 37, 101, 174, 51, 55, 5, 108, 57, 190, 156, 160, 218, 137, 2, 9, 218, 71, 6, 22, 136, 228, 220, 15, 162, 155, 157, 32, 73, 19, 202, 214, 123, 137, 143, 68, 141, 22, 128, 209, 49,
234, 166, 26, 9, 181, 194, 142, 0, 232, 183, 184, 90, 226, 115, 50, 153, 178, 48, 61, 199, 11, 96, 13, 205, 183, 186, 148, 249, 98, 195, 133, 169, 38, 210, 4, 241, 194, 70, 219, 100, 136, 153, 202,
217, 22, 190, 34, 86, 3, 49, 201, 236, 164, 26, 37, 179, 25, 65, 160, 68, 201, 2, 8, 236, 176, 139, 18, 239, 162, 156, 64, 143, 166, 86, 156, 4, 136, 57, 177, 64, 64, 9, 60, 19, 232, 176, 52, 90, 9, 147,
245, 42, 64, 58, 4, 131, 208, 153, 149, 32, 8, 32, 147, 235, 108, 160, 144, 2, 215, 227, 10, 46, 0, 127, 164, 77, 164, 112, 232, 238, 181, 182, 112, 11, 121, 178, 0, 15, 184, 144, 32, 119, 80, 35, 131,
38, 56, 202, 59, 129, 4, 90, 241, 120, 60, 168, 19, 35, 128, 121, 41, 4, 54, 142, 91, 236, 10, 200, 136, 5, 19, 111, 154, 57, 139, 5, 173, 163, 34, 1, 205, 144, 108, 219, 39, 185, 5, 96, 105, 137, 242,
147, 29, 33, 17, 128, 92, 198, 22, 22, 187, 116, 147, 218, 16, 3, 18, 5, 193, 140, 250, 118, 81, 46, 179, 166, 58, 250, 162, 237, 185, 220, 32, 122, 168, 84, 17, 55, 239, 100, 0, 193, 59, 174, 109, 243,
82, 12, 237, 61, 251, 173, 52, 30, 240, 66, 97, 141, 144, 72, 24, 40, 1, 181, 189, 12, 205, 238, 164, 6, 55, 12, 155, 65, 69, 13, 28, 129, 156, 173, 236, 131, 153, 57, 144, 144, 12, 1, 184, 220, 102, 86,
128, 15, 116, 228, 207, 8, 219, 4, 27, 79, 43, 68, 30, 145, 56, 29, 80, 16, 13, 48, 38, 127, 193, 80, 139, 145, 211, 186, 153, 176, 150, 129, 43, 28, 108, 210, 1, 199, 162, 3, 80, 44, 12, 15, 69, 23, 183,
203, 214, 48, 183, 49, 51, 36, 115, 209, 110, 251, 109, 249, 249, 100, 2, 181, 91, 104, 176, 29, 146, 117, 88, 72, 50, 4, 119, 54, 86, 14, 110, 100, 16, 39, 33, 1, 204, 153, 176, 190, 8, 72, 213, 85,
105, 90, 121, 2, 228, 245, 70, 210, 2, 202, 160, 216, 73, 158, 254, 136, 207, 164, 69, 205, 207, 175, 201, 101, 22, 134, 212, 4, 216, 3, 2, 84, 210, 93, 105, 170, 69, 56, 19, 131, 236, 85, 55, 142, 7,
68, 182, 0, 226, 70, 21, 158, 156, 29, 176, 209, 254, 212, 117, 250, 79, 143, 76, 155, 28, 216, 172, 172, 14, 63, 67, 226, 46, 211, 215, 12, 121, 34, 109, 17, 11, 191, 240, 63, 16, 167, 90, 33, 192,
246, 11, 206, 60, 99, 74, 234, 68, 206, 59, 244, 68, 253, 57, 227, 70, 157, 80, 210, 227, 56, 51, 194, 202, 243, 240, 181, 238, 58, 87, 181, 237, 14, 4, 17, 28, 39, 26, 209, 219, 43, 139, 240, 159, 27,
105, 99, 55, 62, 109, 126, 190, 171, 162, 163, 226, 52, 203, 65, 4, 95, 55, 83, 58, 77, 135, 234, 55, 43, 18, 117, 60, 65, 145, 144, 177, 105, 42, 49, 209, 150, 223, 23, 64, 213, 63, 225, 211, 49, 9,
162, 170, 252, 78, 160, 109, 55, 30, 129, 108, 197, 193, 254, 182, 241, 51, 78, 131, 197, 249, 247, 236, 188, 71, 197, 245, 7, 81, 168, 124, 146, 73, 184, 159, 206, 203, 208, 127, 93, 234, 205, 119,
185, 129, 223, 182, 195, 61, 215, 156, 22, 110, 36, 129, 110, 6, 110, 183, 230, 124, 78, 161, 77, 155, 131, 65, 158, 39, 133, 26, 236, 13, 48, 193, 180, 27, 8, 76, 53, 187, 91, 210, 64, 32, 4, 26, 207,
187, 186, 241, 252, 132, 204, 10, 77, 33, 226, 254, 96, 97, 116, 158, 17, 74, 93, 48, 45, 192, 92, 254, 145, 135, 116, 255, 0, 106, 235, 124, 30, 156, 1, 182, 0, 136, 156, 205, 210, 56, 234, 60, 57,
155, 105, 228, 218, 224, 66, 183, 165, 96, 15, 104, 50, 145, 208, 180, 53, 162, 72, 183, 215, 252, 167, 72, 32, 19, 113, 0, 247, 84, 180, 201, 200, 38, 79, 116, 58, 174, 232, 4, 3, 200, 81, 123, 129,
108, 3, 158, 115, 242, 64, 168, 227, 38, 115, 192, 28, 160, 36, 226, 28, 39, 147, 202, 128, 22, 152, 142, 253, 150, 92, 31, 41, 152, 182, 20, 183, 1, 210, 121, 178, 12, 86, 134, 180, 94, 63, 58, 169, 7,
2, 225, 99, 24, 178, 85, 245, 3, 4, 30, 84, 126, 48, 108, 0, 77, 248, 234, 128, 119, 112, 227, 254, 214, 26, 130, 111, 14, 63, 53, 93, 253, 80, 181, 237, 137, 38, 16, 223, 170, 206, 227, 158, 156, 36,
113, 96, 250, 163, 118, 100, 168, 26, 226, 0, 50, 15, 217, 84, 84, 214, 121, 140, 24, 192, 182, 101, 45, 83, 95, 0, 144, 96, 77, 210, 53, 241, 212, 129, 105, 136, 227, 170, 19, 181, 141, 4, 249, 163,
160, 60, 174, 94, 183, 137, 237, 57, 142, 38, 79, 33, 5, 222, 35, 105, 14, 61, 175, 245, 70, 169, 212, 59, 92, 217, 137, 137, 178, 198, 107, 154, 72, 243, 94, 122, 224, 174, 69, 254, 32, 233, 177, 191,
70, 229, 74, 150, 188, 145, 251, 237, 57, 156, 251, 163, 67, 177, 58, 169, 6, 226, 35, 230, 164, 205, 64, 23, 36, 200, 194, 230, 169, 234, 220, 224, 60, 199, 105, 62, 188, 38, 25, 170, 33, 187, 164,
116, 152, 65, 186, 70, 215, 0, 198, 125, 47, 244, 70, 107, 247, 73, 205, 173, 101, 207, 210, 213, 144, 4, 187, 29, 184, 225, 89, 105, 106, 201, 4, 27, 27, 123, 166, 113, 100, 46, 4, 132, 80, 209, 115,
3, 213, 14, 148, 24, 185, 39, 255, 0, 170, 62, 217, 22, 7, 213, 6, 194, 195, 5, 15, 109, 197, 132, 231, 188, 163, 56, 15, 242, 33, 68, 139, 222, 253, 2, 2, 33, 134, 120, 156, 122, 45, 236, 144, 50, 22,
```

Front-end

Install Node.js

We need to install node and npm also in the front-end. If you are using the same Raspberry also for the front-end, we don't have to do anything now.

If you are using another machine for development, you have to install node also in this machine. On mac, we suggest to install node via `brew install` node. On other systems, please go to the official download page.

Install Angular

Install the Angular framework:

```
npm install -g @angular/cli
```

Download this repository

Again, if you are using the same Raspberry, you don't have to download the repository twice. Otherwise, in the development machine, use the same command as above:

```
git clone http://github.com/marconicivitavecchia/microsoft
```

Run the front-end

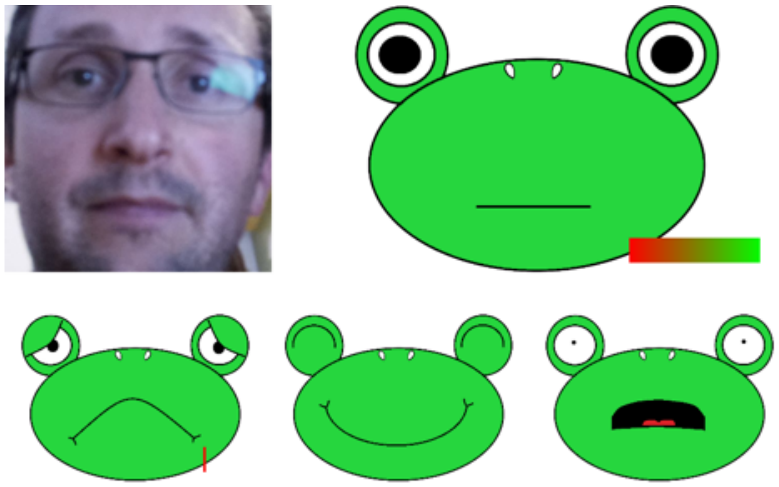
Go to the front-end folder and run it:

```
cd microsoft-ai/code/hello-frog
```

```
ng serve --open
```

Enjoy!

Screenshots



This project is maintained by marconicivitavecchia

Hosted on GitHub Pages — Theme by orderedlist