

UNIVERSITA' DEL PIEMONTE AMEDEO AVOGADO

Corso di Laurea in Informatica

MagicMirror

Relatore

Prof. Marco Guazzone

Relazione della prova

finale di:

Riccardo Berto

Matricola: 20003275

Anno Accademico 2016/2017

Abstract

L'obiettivo dello stage era lo sviluppo di moduli in diversi linguaggi che permettesse l'interazione tra uomo-macchina da parte di un dispositivo, e di un'interfaccia che permettesse di configurare quest'ultimo da una pagina web.

Tra i moduli sviluppati vi sono il riconoscimento di presenza umana o di uno specifico volto (tramite l'utilizzo di una telecamera) e il riconoscimento vocale, ovvero l'impartizione di comandi specifici tramite voce per mezzo di un microfono.

Nello sviluppo di questi moduli si sono incontrate moderne tipologie di software come OpenCV e Google SpeechToText che sono state implementate con diverse tipologie di linguaggi.

La pagina web sviluppata aveva il compito di modificare la configurazione dei moduli (anche quelli creati da terzi) senza dover andare a modificare il file in locale rendendo pi facile aggiornare le features del dispositivo.

Contents

Abstract	I
Introduzione	III
1 Tecnologie Implicate	IV
1.1 Electron	IV

Introduzione

Durante

prova

prova

prova

prova

Chapter 1

Tecnologie Implicate

Durante lo sviluppo del progetto sono state adottate diverse tecnologie recenti e non nel campo della creazione di moduli per un'applicazione (Electron [3], OpenCV [1], GoogleSpeechRecognition [2]) e di web server (NodeJS [11], Model-View-Controller Architecture [4], Express [5], MySQL [6], Mustache [7]), utilizzando diversi linguaggi di programmazione (JavaScript [8], Python [9]) e tecnologie per visionare e condividere codici (GitLab [10]).

1.1 Electron

Electron è un Framework open source rilasciato per la prima volta nel 2013, ma la prima versione stabile è uscita di recente. È disponibile sui sistemi operativi Window, MacOS e Linux ed è scritto in C++ e Javascript. Il framework permette la creazione di interfacce grafiche (GUI) per applicazioni cross platform, utilizzando tecnologie già esistenti per lo sviluppo del backend e del frontend (Javascript, NodeJS, V8). Un'applicazione Electron ha bisogno di 3 componenti principali:

- Il package.json, un file JSON, che deve contenere almeno il nome dell'applicazione, la versione dell'applicazione creata, la descrizione di quest'ultima e il nome del file principale dell'applicazione (necessaria per l'avvio)

```
{
  "name": "magicmirror",
  "version": "2.1.1",
  "description": "The open source modular smart mirror platform.",
  "main": "js/electron.js"
}
```

- fdfsaf

- la flessibilità e l'ef

Bibliography

- [1] OpenCV official website, <http://opencv.org/>
- [2] Google Speech to Text API documentation, <https://cloud.google.com/speech/>
- [3] Electron official website, <https://electron.atom.io/>
- [4] MVC Wikipedia, <https://en.wikipedia.org/wiki/Model%E2%80%93view%E2%80%93controller>
- [5] Express official website, <http://expressjs.com/it/>
- [6] MySQL official website, <https://www.mysql.com/it/>
- [7] Mustache Git Site, <https://mustache.github.io/>
- [8] JavaScript, <https://www.javascript.com/>
- [9] Python website, <https://www.python.it/>
- [10] GitLab website, <https://about.gitlab.com/>
- [11] Nodejs website, <https://nodejs.org/it/>