



Un sistema di gestione dei parcheggi IOT che offre una serie di funzionalità innovative e real time per aiutare i clienti a trovare il parcheggio perfetto.

Parcheggi Automatizzati

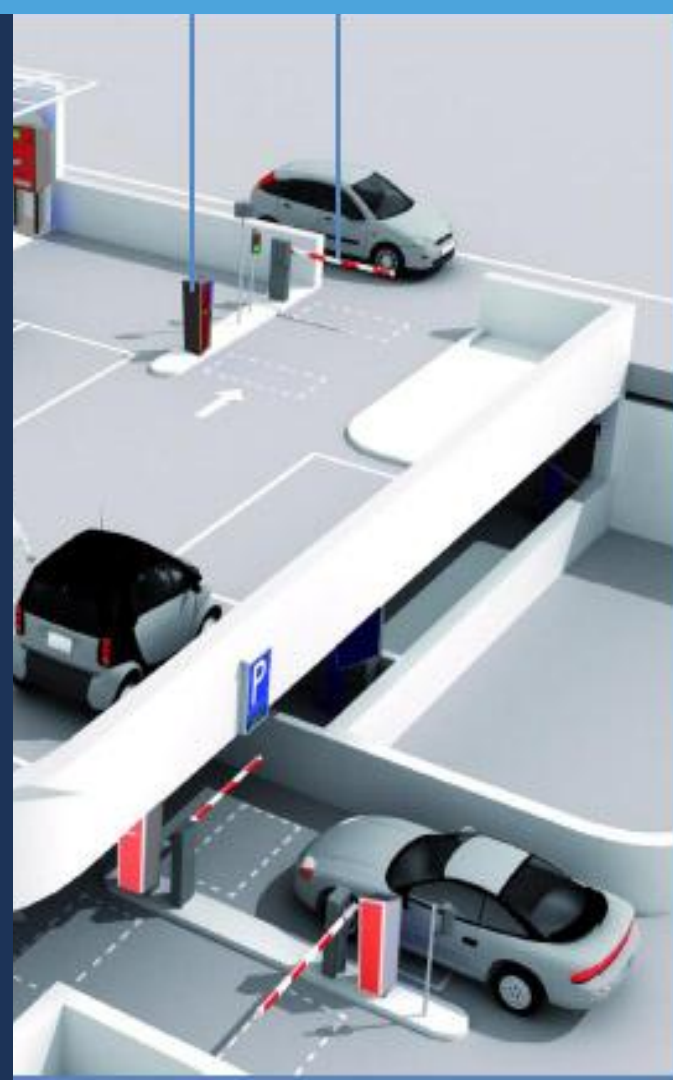
- Rende il parcheggio un'esperienza più comoda e piacevole.
- Abbatte i tempi di ingresso ed uscita attraverso sensoristica e rilevamento targhe
- Offre la possibilità di pagamento istantaneo tramite carta di credito associata
- Riduce il traffico e l'inquinamento velocizzando la ricerca del parcheggio



Caratteristiche principali

Parcheggio:

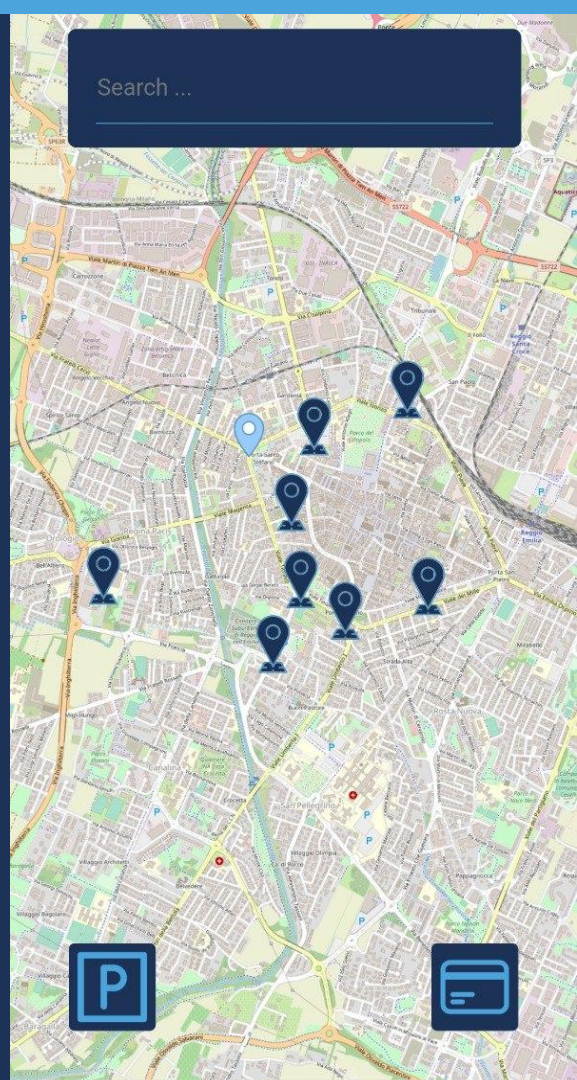
- Mostra il numero di posti disponibili
- Dispone di ingresso e uscita automatizzati grazie a una sbarra gestita tramite vari sensori
- Permette il pagamento automatico tramite carta di credito associata alla targa del veicolo oppure attraverso il metodo classico



Caratteristiche principali

Web App:

- Mostra una mappa interattiva dei parcheggi che aderiscono all'iniziativa
- Consiglia i migliori parcheggi in base alla destinazione inserita e relativi tempi di percorrenza
- Consente l'associazione targa-carta di credito



Prossimi Passi

- Prenotazione dei Parcheggi
- Algoritmi di previsione
- Integrazione con i principali sistemi di navigazione
- Gestione in grossi impianti
- Adesione all'iniziativa da parte di parcheggi privati
- Altri Metodi di pagamento (Telepass...)



Dettagli Architettonici

Architettura



Controllore

- Gestisce i 2 varchi di ingresso e uscita comunicando le informazioni
- Funge da Bridge connettendo i parcheggi con i loro digital twin
- Esegue il processing delle immagini



Varchi Ingresso - Uscita

- Rileva la posizione di un'automobile
- Permette il passaggio alzando la sbarra
- Mostra a display informazioni utili (pagamento, posti disponibili)



Hosting VM Azure

- Macchina virtuale sulla quale è attivo il Web Server che ospita l'applicazione
- Garantisce l'hosting della web app e le permette di essere raggiunta all'indirizzo:

<https://w2p.northeurope.cloudapp.azure.com/>

- Fornisce l'accesso alle API per la gestione dei parcheggi



Django Web App

- Applicazione web che permette all'utente di interrogare i parcheggi ed ottenere le informazioni utili alla selezione del parcheggio ottimo
- Algoritmo home made per consigliare i parcheggi
- Gestione Back-End tramite Python e SQLite
- Gestione Front-End tramite JS, CSS e HTML

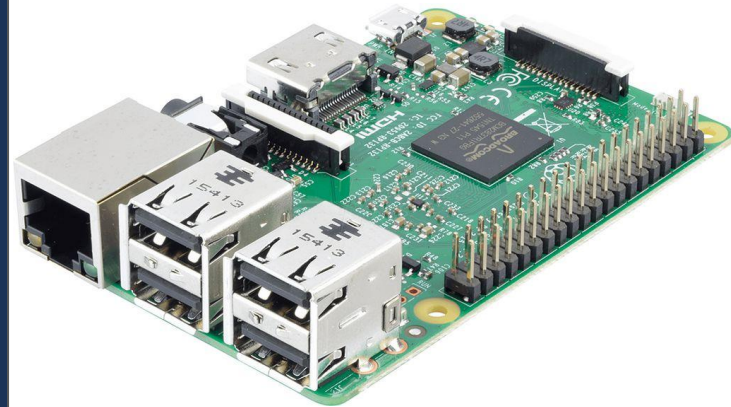


Dettagli Tecnici

Raspberry pi 3b+

Comunica con gli arduino di ingresso e uscita attraverso seriale ed esegue script python nei quali:

- Attende comunicazione da parte dei varchi riguardo la presenza di una vettura
- Acquisisce una foto tramite **Pi Camera** e ne effettua il processing
- Restituisce la targa letta
- Esegue chiamate API per l'aggiornamento del digital twin del parcheggio



Arduino

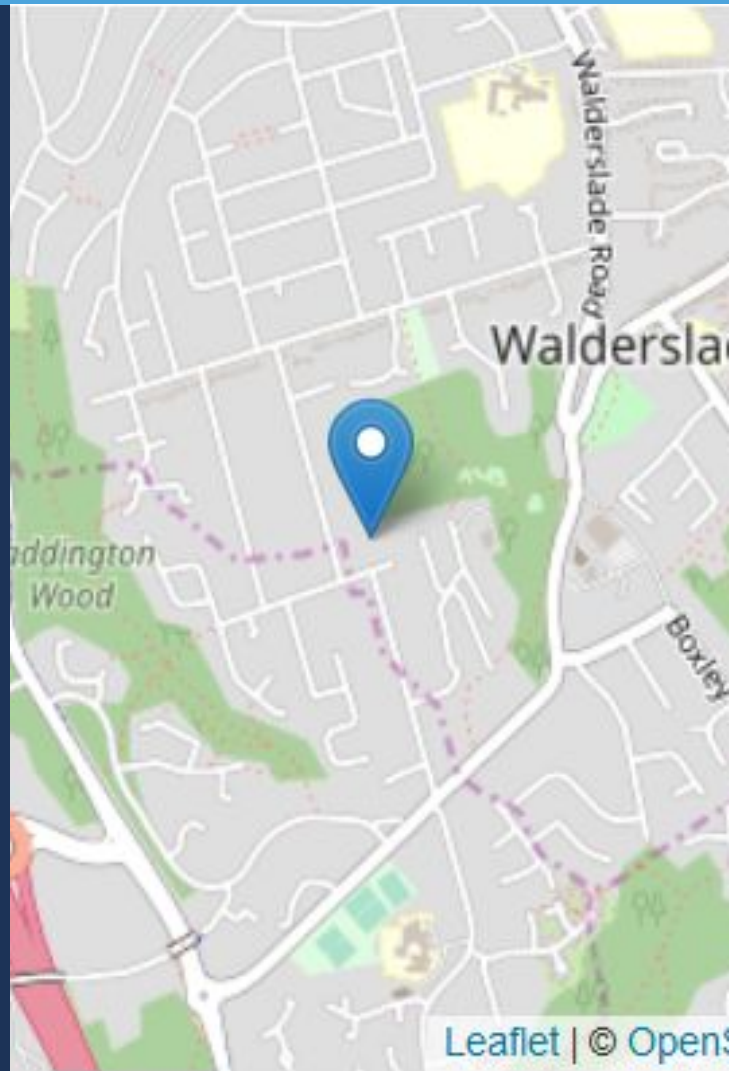
Attraverso uno script comunica via seriale con il Raspberry e gestisce i seguenti sensori:

- **Sensori ad Ultrasuoni:** rilevano la presenza di una macchina
- **Servomotori:** azionano la sbarra
- **Display a 7 segmenti :** mostra all'ingresso il numero di posti disponibili
- **Schermo LCD:** mostra il pagamento dovuto all'uscita in caso non ci sia un metodo di pagamento associato e saluta il cliente
- **Bottone:** simula pagamento con carta non associata



Componenti JS utilizzati

- **Leaflet**: libreria JavaScript per la creazione di mappe interattive su pagine web e integrazione con Open Street Map
- **Leaflet.AutoComplete**: plugin di autocompletamento e geocoding
- **OSRM HTTP**: API per ottenere distanze date le coordinate di 2 punti



Algoritmi utilizzati

- Algoritmo di selezione parcheggio:

script Python che date le coordinate di destinazione e posizione, restituisce tra i parcheggi a DB quello più vicino, quello gratuito più vicino e quello “ottimo”, valutato sulla base di occupazione e vicinanza

- API di comunicazione :

funzioni Python associate a determinati indirizzi web, che permettono lo scambio di informazioni tra il parcheggio fisico e il database centrale. Sono presenti API per la creazione di un posteggio, per la sua ‘chiusura’ e per il pagamento nel caso non si abbia una carta di credito associata.

```
ElementDesc elDesc;  
  
long sp_name = item->Attribute  
long spritename = item->Attr  
  
boost::lexical_cast<float  
boost::lexical_cast<float  
set = boost::lexical_cast<  
layer = 50; // default  
->Attribute( "layer" ) !=  
  
layer = boost::lexical_cast<
```

Riconoscimento targhe

Script Python che attraverso la libreria OpenCV acquisisce l'immagine ed esegue trasformazioni e manipolazioni su di essa.

Applicazione di Optical Character Recognition fornita dal framework Tesseract e dalla libreria PyTesseract per estrarre i caratteri della targa.

Per migliorare l'accuracy del Character Recognition abbiamo affiancato al nostro algoritmo una chiamata ad un'API gratuita che garantisce un servizio di riconoscimento targhe, anche con immagini in situazioni di scarsa luminosità o imperfetto orientamento.

Adottando questa soluzione siamo in grado di garantire robustezza ed affidabilità al sistema.



Try it out!



Valentini Andrea
Bertolani Stefano
Alessandro Artoni