

# **ParkM&G**

## **PMG**

### **Progetto di Progettazione, Algoritmi e Computabilità**

#### **Team**

Riccardo Maffeis, Alessio Giovanelli

#### **Idea**

Realizzazione di un sistema informatico per la gestione di un parcheggio multipiano a pagamento, con l'obiettivo di ottimizzare l'uso degli spazi, automatizzare i processi di accesso/uscita e semplificare la gestione economica del servizio.

Il sistema deve integrare una componente Desktop destinata agli operatori di cassa e una componente Web dedicata agli utenti, che potranno verificare la disponibilità dei posti e prenotare in anticipo.

Il sistema principale sarà utilizzato dal personale del parcheggio per:

- Controllare le barre di accesso e uscita;
- Monitorare e aggiornare in tempo reale la disponibilità dei posti;
- Gestire i blocchi fisici (manutenzione o posti riservati);
- Emettere e annullare i ticket di parcheggio;
- Calcolare automaticamente la tariffa in base al tempo di permanenza del veicolo;
- Gestire la cassa e i pagamenti, registrando ogni transazione.

Il software dovrà quindi fornire un'interfaccia intuitiva e sicura per l'operatore, consentendo la supervisione e l'intervento sui dispositivi automatici del parcheggio.

La componente Web del sistema è destinata agli utenti che desiderano:

- Consultare la disponibilità dei posti in tempo reale;
- Effettuare la prenotazione di un posto in una data e fascia oraria prefissata;
- Ricevere la conferma della prenotazione (voucher elettronico o QR code da presentare all'ingresso).

Il portale dovrà interfacciarsi con il sistema centrale per mantenere aggiornata la situazione complessiva del parcheggio, evitando sovrapposizioni o doppie prenotazioni.

Inoltre, il sistema prevede agevolazioni per fasce d'età / occupazione e la possibilità di sottoscrivere un abbonamento mensile e annuale.

## Algoritmi

Algoritmo di ottimizzazione basato su euristiche che assegna automaticamente il miglior posto disponibile in base a:

- Distanza dall'uscita;
- Tipo di utente (abbonato, disabile, VIP);
- Minimizzazione degli spostamenti interni;
- Distribuzione uniforme dei veicoli.

L'app web o un display può guidare l'utente verso il posto ottimale con un percorso calcolato tramite algoritmi.

Il sistema:

- Evita zone congestionate;
- Propone il percorso più rapido o più sicuro;
- Assegna dinamicamente il percorso in caso di manutenzione.

Un tool per gli operatori che simula:

- Scenari di affluenza
- Impatto delle manutenzioni
- Variazioni tariffarie
- Prognosi dell'occupazione

## Requisiti software

Utenti:

- Creazione delle credenziali personali;
- Selezione delle preferenze;
- Arrivare al posto assegnato tramite tracking;
- Consultare disponibilità posti;
- Effettuare prenotazione;
- Ricevere codice / QR code prenotazione.

Operatore:

- Accesso;
- Monitoraggio sbarre di accesso e uscita;
- Monitorare la disponibilità dei posti in tempo reale ;
- Gestire i blocchi fisici;
- Convalidare i ticket;
- Visualizzare panoramica della situazione generale;
- Gestire la cassa e i pagamenti.

Database:

- Salvataggio dati utenti e operatori.

## Tool chain

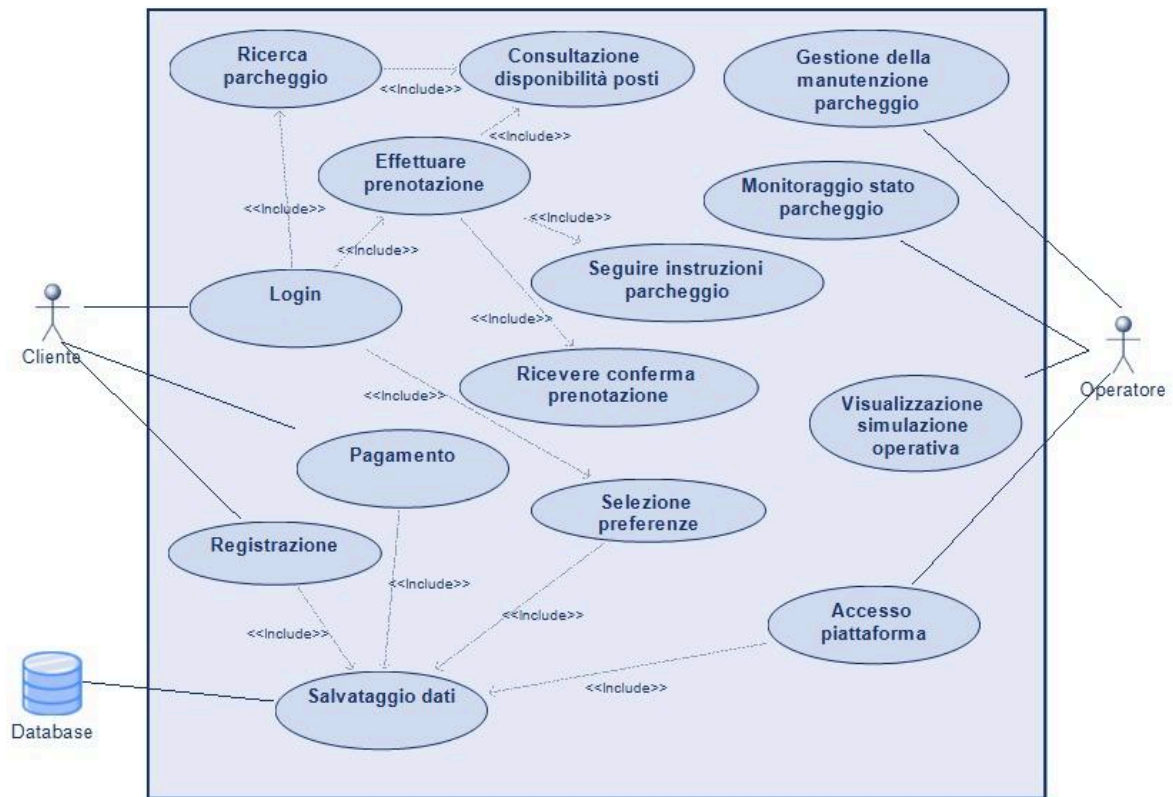
- Attività di design:
  - Modelio
  - WindowBuilder
- Sviluppo collaborativo: Github Desktop, Postman e JAutoDoc
- Analisi statica e dinamica del codice: Structure101, JUnit e Playwright (<https://playwright.dev/>) per testare su web
- Distribuzione: Github
- Versionamento: Github

## Agenda e organizzazione

- Linea temporale: GitHub Projects Roadmap
- Suddivisione compiti: GitHub Projects Backlog
  - Backlog: nuovi issues creati
  - Ready: issues pronti da sviluppare
  - In progress: issues che sono in sviluppo
  - In review: issues che devono essere testati e controllati
  - Done: issues terminati
- Riunioni giornaliere per organizzarsi sul lavoro da svolgere durante la giornata
- Riunioni settimanali per discutere su come sta procedendo lo sviluppo

## Modellazione

- Diagrammi UML
  - Casi d'uso



- Architettura

