

App PuliziaStrade Firenze

Progetto Human Computer Interaction

Pietro Zarri 7042627 - Luglio 2021



Introduzione

PuliziaStrade Firenze è un applicazione per smartphone che permette di conoscere rapidamente il giorno di pulizia strade in una via e allo stesso tempo funge da promemoria per avvisare in anticipo gli utenti ed evitare la rimozione del proprio vicolo.

L'idea è nata dopo un confronto con amici sulle problematiche legate al servizio di pulizia strade.

Lo strumento principale utilizzato per realizzare questo progetto è il framework **Flutter**.



Fasi del progetto

Le fasi che si sono susseguite durante la realizzazione del progetto sono:

1. **Needfindings**
2. **Personas**
3. **Scenari**
4. **Requisiti**
5. **Mockup**
6. **Implementazione**
7. **Usability Testing**
8. **Miglioramenti**



Needfindings

La prima fase del lavoro ha riguardato la ricerca e la scoperta dei requisiti e degli obiettivi da implementare all'interno del applicazione.

Sono state fatte delle **domande** a dei possibili utenti per ricavare informazioni con cui iniziare il progetto. Le domande principali poste sono state le seguenti:

- Quanti veicoli possiedi e quale il loro principale utilizzo ?
- È facile parcheggiare nella tua zona di residenza?
- Hai difficoltà a rintracciare i che indicano la data di pulizia strade?
- Hai mai avuto inconvenienti dovuti al servizio di pulizia strade?
- Ricordi il giorno di pulizia strade nelle strade in cui parcheggi abitualmente?
- Riesci sempre a ricordare dove hai parcheggiato la macchina?
- Potrebbe utile un applicazione per ricevere informazioni sulle date di pulizia strade e memorizzare i luoghi in cui si è parcheggiato il veicolo?



Needfindings

Dalle interviste è emerso che:

- Molte persone che risiedono a Firenze hanno difficoltà a **trovare** un **parcheggio** per il proprio veicolo, quindi cambiano spesso strada.
- Ad alcuni intervistati è capitato di subire una **rimozione** del proprio veicolo a causa della pulizia strada..
- Ad alcuni intervistati è risultato difficile o faticoso **rintracciare il cartello** che indica il giorno di pulizia strade.
- Ad alcuni intervistati è capitato di **non ricordare la via** in cui hanno parcheggiato il proprio veicolo.
- Al alcuni intervistati è capitato di **dimenticare il giorno di pulizia strade** della via in cui hanno parcheggiato il proprio veicolo.
- Molti intervistati sono stati favoriti alla realizzazione di **un'applicazione** per **risolvere le problematiche** appena descritte.



Personas

- **Studente** (20 - 25 anni): Risiede in città e possiede un'auto personale che al ritorno a casa parcheggia nelle vie circostanti la propria abitazione. Spesso impiega molto tempo a trovare un parcheggio libero.
- **Impiegato** (30 - 60 anni): Possiede due veicoli e parcheggia il primo nel suo garage privato mentre il secondo nei pressi della propria abitazione.
- **Libero professionista** (30 - 60 anni): Possiede un veicolo per la propria attività e che solitamente rimane parcheggiato nei pressi di essa.



Scenari

Una volta delineate delle **personas** possiamo definire alcuni **scenari**:

- **Studente** torna casa e parcheggia la macchina senza controllare il giorno di pulizia strade. Vorrebbe uno strumento per **conoscerlo in modo rapido**.
- **Studente** dimentica il giorno di pulizia strade. Vorrebbe aver ricevuto un **promemoria** per spostare la macchina.
- **Impiegato** vorrebbe **conoscere in anticipo** la data di pulizia strade per poter scegliere la strada in cui parcheggiare la macchina.
- **Libero professionista** gradirebbe ricevere una **notifica** nel giorno di pulizia strade della via della sua attività.



Requisiti

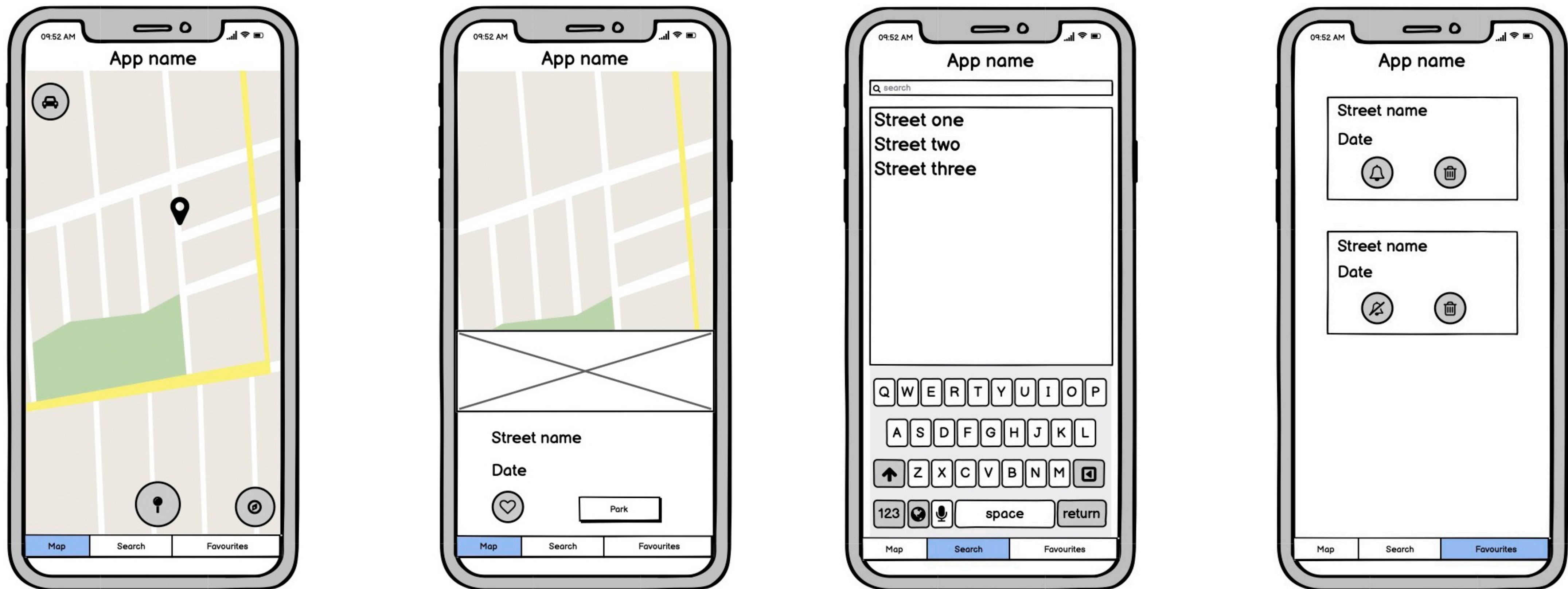
Sono stati definiti così i **requisiti** che l'applicazione deve avere:

- Ottenere informazioni sulla posizione tramite **geolocalizzazione**.
- Poter **salvare una via** per consultare rapidamente informazioni su di essa.
- Poter **salvare la posizione** in cui si è parcheggiato il proprio veicolo.
- **Ricevere una notifica** nel giorno di pulizia strade per la via in cui si è parcheggiato il veicolo.
- Consentire una **ricerca** rapida tra le vie di tutto il comune.
- Consentire all'utente di **decidere** per quali vie ricevere un notifica di avviso pulizia strada



Mockup

I mockup sono stati realizzati tramite **Balsamiq Mockup**.



Implementazione

Il framework principale utilizzato per la realizzazione dell'applicazione è **Flutter SDK**.

Flutter è un framework open-source creato da Google per la creazione di interfacce native per **iOS** e **Android**.

I dati sui giorni di pulizia strade sono ottenuti tramite il sito **OpenData del comune di Firenze**.

Tramite il servizio **Pythonanywhere** sono state create della API per ottenere in formato **JSON** informazioni su:

- **Ottenere elenco** di tutte le vie presenti nel dataset.
- **Conoscere i tratti** in cui è suddivisa ciascuna via.
- **Conoscere la data** della prossima pulizia strade rispetto ad una specifica via e tratto.



Implementazione

Per la persistenza dei dati sono stati utilizzati due plugin:

- **sqflite**: plugin SQLite per Flutter.
- **sharedpreferences**: plugin Flutter per lettura/scrittura di coppie key-value.

Le mappe sono state integrate all'interno dell'app tramite le API di **Google Maps**, per la localizzazione è utilizzato il package **Location**.

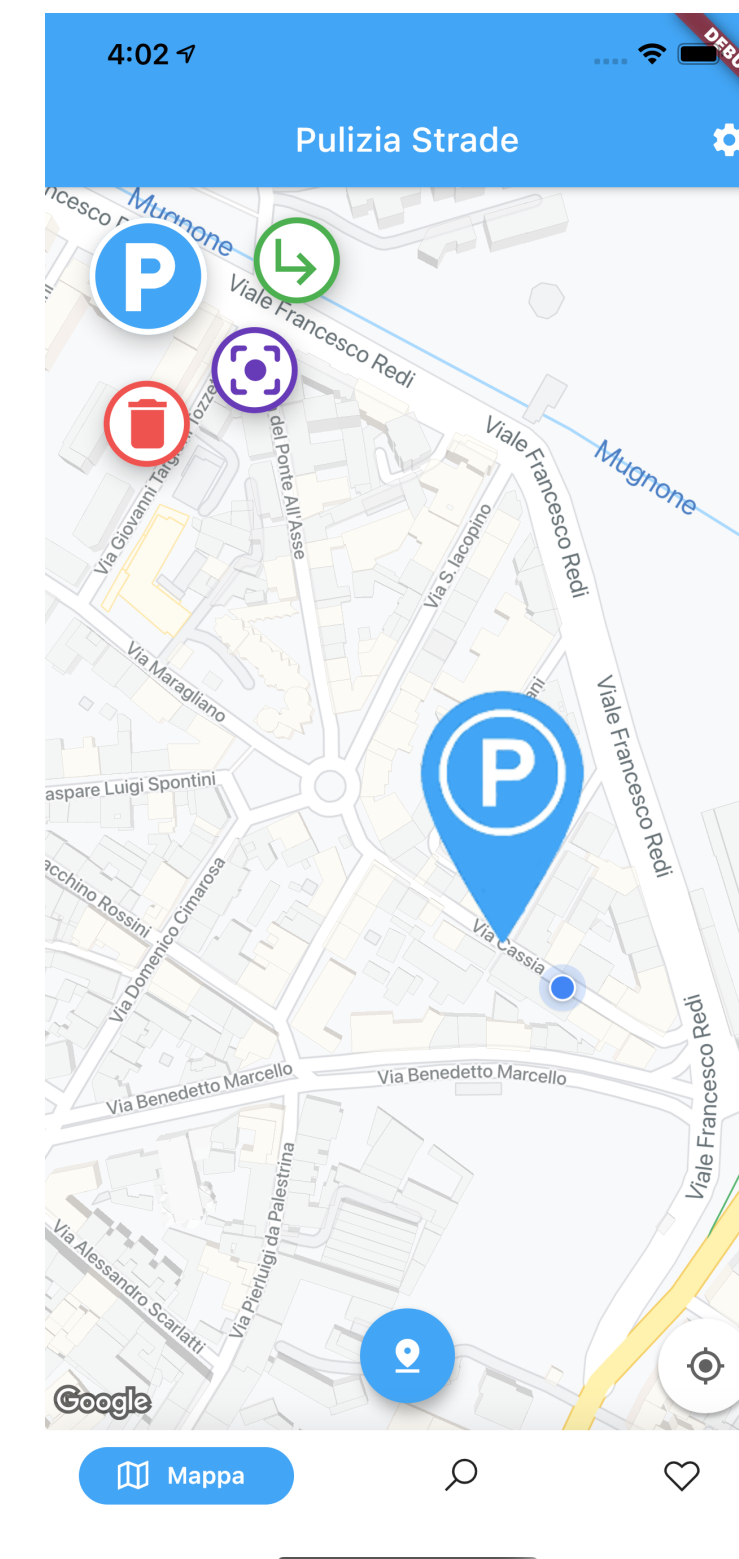
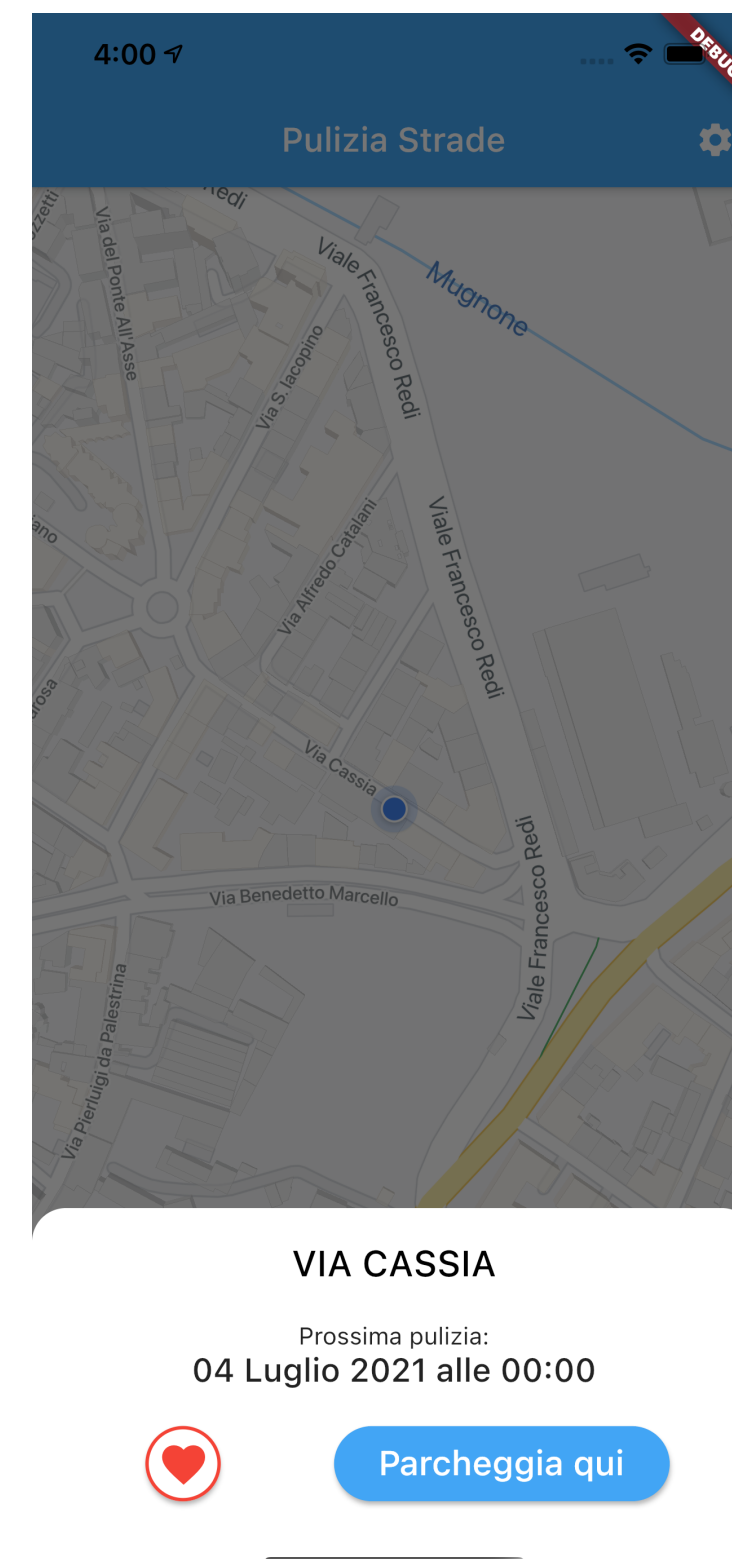
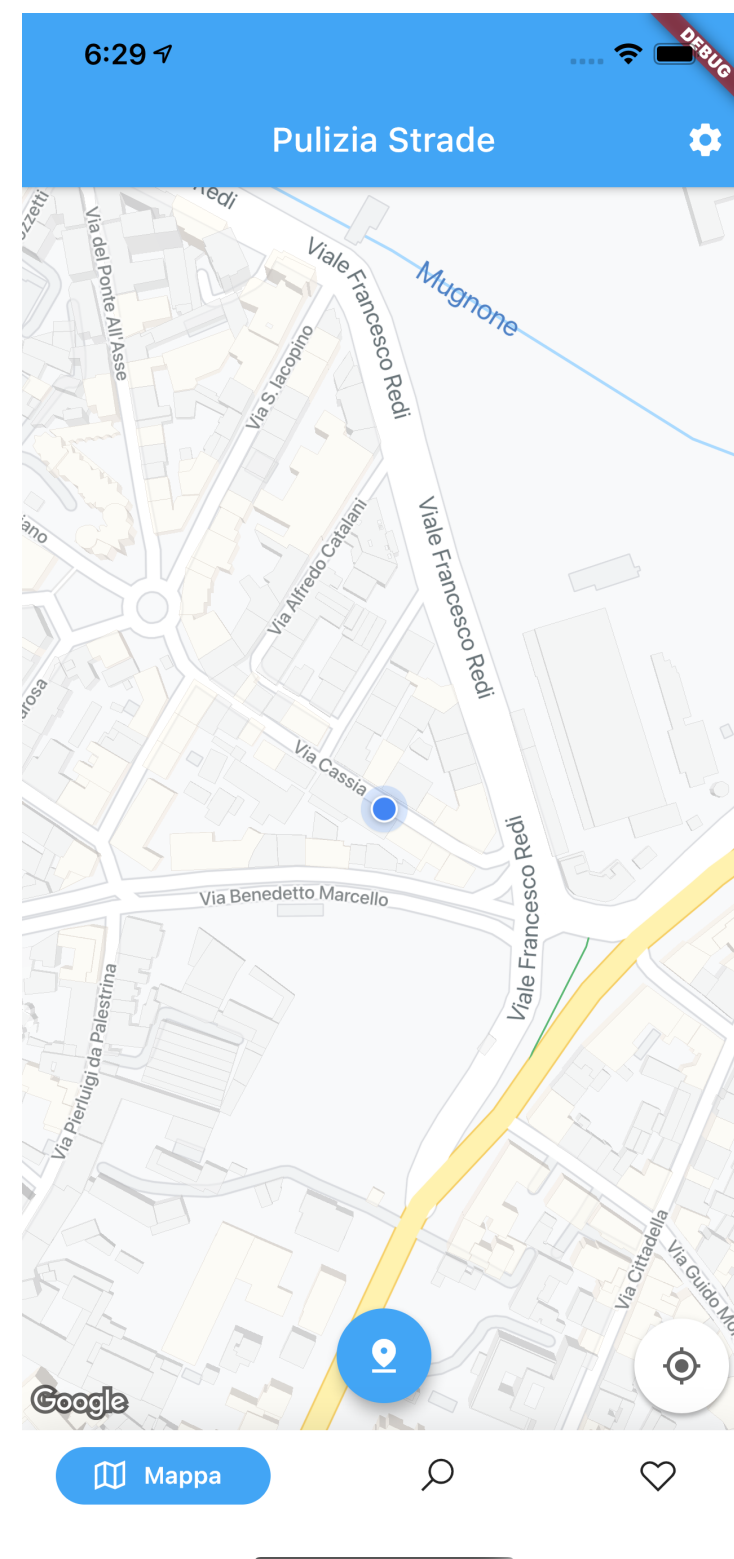
Il servizio utilizzato per implementare le notifiche di tipo push ai dispositivi è **Firebase Cloud Messaging**.

Altri package Flutter sono stati utilizzati per implementare alcuni elementi grafici all'interno dell'app (*search bar, pagina impostazioni,...*) .



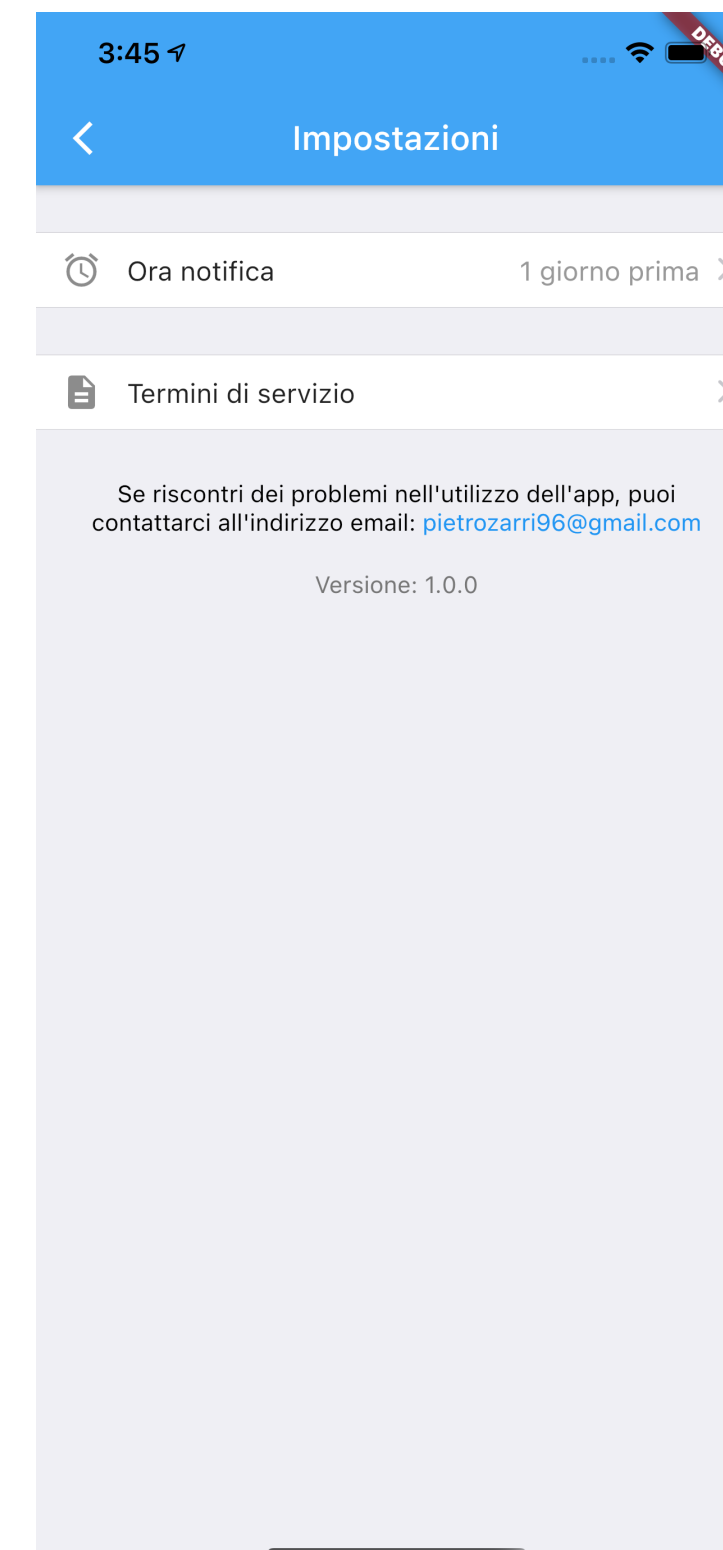
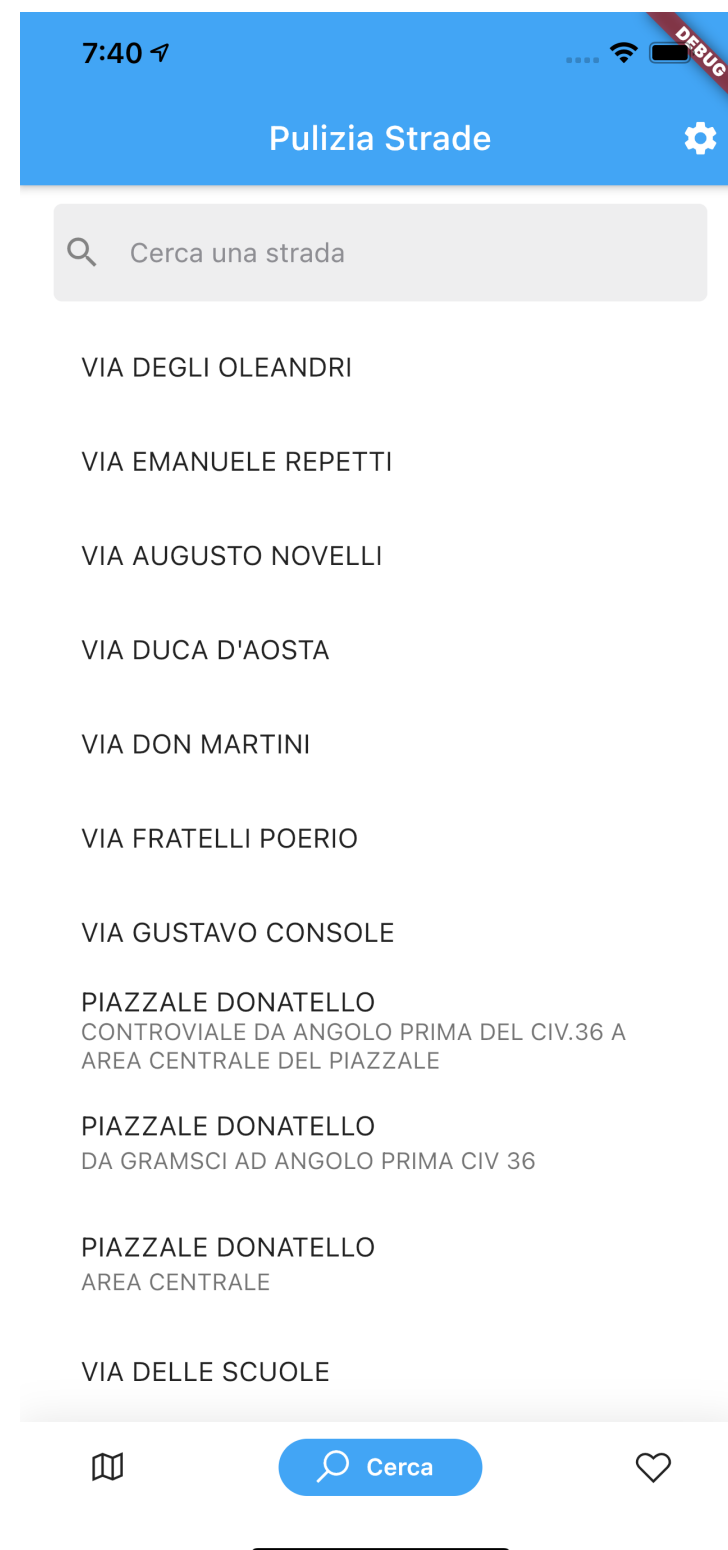
Implementazione

La pagina **Mappa** consente di ottenere informazioni sulla **posizione attuale** e di salvare una posizione come parcheggio.



Implementazione

Le altre tre pagine dell'app sono: **Cerca**, **Preferiti** e **Impostazioni**.



Implementazione

Tramite il servizio **Firestore Cloud Messaging** di Firebase sono state implementate notifiche push su iOS e Android.

Ogni via è considerata come un **topic** a cui un singolo dispositivo può effettuare una **subscription** per ricevere notifiche in modo continuo nei giorni di pulizia strade.



Usability testing

Utile per rilevare potenziali problemi e capire quanto il prototipo costruito risulti intuitivo e *user-friendly* ad un utente medio.

Sono stati proposti alcuni *task* di tipo *scenario* ad esempio:

- **Cerca la data** della prossima pulizia strade per la via in cui ti trovi adesso.
- **Parcheggia** il tuo veicolo nel luogo in cui ti trovi adesso.
- **Cerca la data** della prossima pulizia strade in via dei Ciompi.
- **Disattiva le notifiche** per via Cassia



Usability testing



Implementazione

Per la persistenza dei dati sono stati utilizzati due plugin:

- **sqflite**: plugin SQLite per Flutter.
- **sharedpreferences**: plugin Flutter per lettura/scrittura di coppie key-value.

Le mappe sono state integrate all'interno dell'app tramite le API di **Google Maps**, per la localizzazione è utilizzato il package **Location**.

Il servizio utilizzato per implementare le notifiche di tipo push ai dispositivi è **Firebase Cloud Messaging**.

Altri package Flutter sono stati utilizzati per implementare alcuni elementi grafici all'interno dell'app (*search bar, pagina impostazioni,...*) .



Implementazione

Per la persistenza dei dati sono stati utilizzati due plugin:

- **sqflite**: plugin SQLite per Flutter.
- **sharedpreferences**: plugin Flutter per lettura/scrittura di coppie key-value.

Le mappe sono state integrate all'interno dell'app tramite le API di **Google Maps**, per la localizzazione è utilizzato il package **Location**.

Il servizio utilizzato per implementare le notifiche di tipo push ai dispositivi è **Firebase Cloud Messaging**.

Altri package Flutter sono stati utilizzati per implementare alcuni elementi grafici all'interno dell'app (*search bar, pagina impostazioni,...*) .

