

POLITECNICO DI MILANO

Scuola di Ingegneria Industriale e dell'Informazione

Corso di Laurea Magistrale in
Computer Science and Engineering



Titolo della tesi

Relatore: Prof. Luciano BARESI

Tesi di Laurea di:

Andrea CALICI

Matr. 944717

Anno Accademico 2020 - 2021

Indice

Definizioni e Acronimi	13
1 Introduzione	3
1.1 Panoramica Generale	3
1.2 Stato dell'Arte	3
1.2.1 Carta d'Identità Salvavita	4
1.3 Obiettivo del Progetto	4
1.3.1 Goals	5
2 Panoramica del progetto	7
2.1 Architettura	7
2.2 Web Application	7
2.3 Mobile Application	7
2.3.1 Flutter	8
2.4 Database	8
3 Implementazione	9
3.1 Applicazione Mobile	9
3.1.1 Model	9
3.1.2 Screens	10
3.1.3 Services	10
3.1.4 Widgets	11
3.2 Lato Server	12
3.2.1 Autenticazione	12
3.2.2 Firestore Database	13
3.3 External Services and Libraries	13
4 Funzionalità	15
4.1 Panoramica dell'Applicazione	15
4.2 Descrizione delle Funzionalità	16
4.2.1 Registrazione	16
4.2.2 Login	16
4.2.3 Schermata del Volontario	16
4.2.4 Homepage	16
4.2.5 Numeri Utili	17
4.2.6 Scanner QR Code	18

5	Interfaccia Utente e Casi d'Uso	19
5.1	UI Design	19
5.1.1	Logo	19
5.1.2	Schermate Principali	19
5.1.3	Layout Web	19
5.2	Use Cases	19
5.2.1	Login	20
5.2.2	Logout	21
5.2.3	Homepage	22
5.2.4	Schermata del Volontario	26
5.2.5	Numeri Utili	29
5.2.6	QR Code Scanner	30
5.3	User-Flow Diagram	30
6	Discussions and Conclusion	31
7	Bibliografia	33

Elenco delle figure

3.1	The Model files	9
-----	---------------------------	---

Elenco delle tabelle

Listings

Ringraziamenti

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetur id, vulputate a, magna. Donec vehicula augue eu neque. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Mauris ut leo. Cras viverra metus rhoncus sem. Nulla et lectus vestibulum urna fringilla ultrices. Phasellus eu tellus sit amet tortor gravida placerat. Integer sapien est, iaculis in, pretium quis, viverra ac, nunc. Praesent eget sem vel leo ultrices bibendum. Aenean faucibus. Morbi dolor nulla, malesuada eu, pulvinar at, mollis ac, nulla. Curabitur auctor semper nulla. Donec varius orci eget risus. Duis nibh mi, congue eu, accumsan eleifend, sagittis quis, diam. Duis eget orci sit amet orci dignissim rutrum.

Desidero inoltre ringraziare esplicitamente:

Esplicito1 per vari motivi;

Esplicito2 per altri motivi;

Esplicito3 per puro piacere, senza particolari motivi.

Milano, Mese 2021

A. C.

Sommario

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetur id, vulputate a, magna. Donec vehicula augue eu neque. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Mauris ut leo. Cras viverra metus rhoncus sem. Nulla et lectus vestibulum urna fringilla ultrices. Phasellus eu tellus sit amet tortor gravida placerat. Integer sapien est, iaculis in, pretium quis, viverra ac, nunc. Praesent eget sem vel leo ultrices bibendum. Aenean faucibus. Morbi dolor nulla, malesuada eu, pulvinar at, mollis ac, nulla. Curabitur auctor semper nulla. Donec varius orci eget risus. Duis nibh mi, congue eu, accumsan eleifend, sagittis quis, diam. Duis eget orci sit amet orci dignissim rutrum.

Parole chiave: PoliMi, Tesi, LaTeX, Scribd

Abstract

Text of the abstract in english. . .

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetur id, vulputate a, magna. Donec vehicula augue eu neque. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Mauris ut leo. Cras viverra metus rhoncus sem. Nulla et lectus vestibulum urna fringilla ultrices. Phasellus eu tellus sit amet tortor gravida placerat. Integer sapien est, iaculis in, pretium quis, viverra ac, nunc. Praesent eget sem vel leo ultrices bibendum. Aenean faucibus. Morbi dolor nulla, malesuada eu, pulvinar at, mollis ac, nulla. Curabitur auctor semper nulla. Donec varius orci eget risus. Duis nibh mi, congue eu, accumsan eleifend, sagittis quis, diam. Duis eget orci sit amet orci dignissim rutrum.

Keywords: PoliMi, Master Thesis, LaTeX, Scribd

Definizioni e Acronimi

Definizioni

In questa sezione verrà presentata una lista di termini comuni e di utile comprensione che verranno utilizzati nel corso della trattazione:

- Utente: il generico utente finale dell'applicazione e comprensivo delle figure dei Cittadino e del Volontario;
- Cittadino: utente finale che utilizzando l'applicazione è in grado di ;
- Volontario: utente finale dell'applicazione che rappresenta la figura di riferimento per l'inserimento dei dati di natura pratica del Cittadino e che può avere accesso al sistema sia per la fase di inserimento di informazioni, sia per eseguire funzioni di utilizzo dei dati;
- Carta d'Identità Salvavita: documento contenente il codice QR salvavita e i dati sanitari di emergenza del cittadino;
- Fascicolo Sanitario Elettronico:
- Scheda Sanitaria Individuale:

Acronimi

In questa sezione verrà presentata una lista di acronimi comuni e di utile comprensione che verranno utilizzati nel corso della trattazione:

- CIS: Carta d'Identità Salvavita;
- MMG: Medico di Medicina Generale;
- ICE: In Caso di Emergenza;
- MVI: Medici Volontari Italiani;
- FSE: Fascicolo Sanitario Elettronico;
- SSI: Scheda Sanitaria Individuale;

Capitolo 1

Introduzione

1.1 Panoramica Generale

Lo scopo di questo documento è di descrivere il processo di design e di implementazione dell'applicazione mobile e web sviluppata per Android e per i più comuni web browsers attualmente disponibili attraverso la descrizione funzionale dell'intero sistema, l'analisi dettagliata dell'architettura, i componenti con le relative interazioni e i più frequenti casi d'uso.

Partendo dal lavoro svolto da Medici Volontari Italiani che ha sviluppato la Carta d'Identità Salvavita e il progetto Busta Rossa del Comune di Milano volto alla salvaguardia della salute del cittadino, io e Davide Laffi abbiamo sviluppato un sistema in grado di:

- salvare le informazioni dei cittadini inserite dal Medico di Medicina Generale o dal volontario incaricato;
- mostrare queste informazioni all'utente finale dell'applicazione sviluppata;
- utilizzare i vari dati salvati per generare documenti di utilità per il cittadino.

In particolare il sistema si compone di 3 parti fondamentali:

- l'applicazione web sviluppata da Davide Laffi e pensata per gli MMG e i volontari per la fase di inserimento e modifica dei dati sanitari e salvavita dei cittadini;
- l'applicazione mobile, accessibile anche dal web, sviluppata da me e pensata soprattutto per i cittadini che possono accedere ai propri dati e avere sempre a portata di mano i propri dati sanitari e salvavita;
- il lato server che fornisce funzioni di autenticazione e database e che permette di interfacciare le due componenti precedenti.

1.2 Stato dell'Arte

Lo stato attuale dell'arte presenta per il singolo cittadino una serie di entità differenziate in grado di gestire i suoi dati sanitari, con l'ovvia conseguenza di avere:

- informazioni duplicate;
- diversi sistemi proprietari che non si interfacciano tra di loro;
- dati inseriti manualmente e spesso soggetti ad errori o perdita degli stessi.

Il singolo cittadino, infatti:

- è dotato del Fascicolo Sanitario Elettronico accessibile attraverso il portale della regione di residenza che riporta la storia sanitaria dello stesso;
- presenta la sua storia clinica attuale nella Scheda Sanitaria Individuale compilata dal Medico di Medicina Generale;
- può utilizzare l'app ICE che permette di contattare i soccorsi sanitari in caso di emergenza;
- può richiedere la Carta d'Identità Salvavita.

Necessità fondamentale in questa situazione è cercare di integrare quanto più possibile questi diversi sistemi per realizzare un prodotto accessibile dall'utente finale e manutenibile dalle stesse entità senza la presenza di informazioni duplicate o contrastanti.

IMMAGINE SCHEMATICA

1.2.1 Carta d'Identità Salvavita

Medici Volontari Italiani ha sviluppato, con l'aiuto di Fondazione IBM, Società Gisette e il Comune di Milano, un sistema in grado di digitalizzare i dati sanitari salvavita in modo che siano utilizzabili da soccorritore in caso di necessità. Tramite l'inserimento delle informazioni del cittadino, il sistema genera la Carta d'Identità Salvavita, una tessera cartacea che presenta sulla facciata sinistra i dati anagrafici e i dati salvavita forniti dall'MMG e sulla parte destra il cosiddetto badge, ovvero un QR code riassuntivo di tali informazioni e i numeri di emergenza da contattare in caso di necessità.

1.3 Obiettivo del Progetto

L'obiettivo del progetto è quello di realizzare un sistema che unifichi il flusso di informazioni suddividendolo in due parti fondamentali:

- l'inserimento dei dati sanitari e salvavita del cittadino realizzato dal MMG e dal volontario;
- l'utilizzo di queste informazioni per generare la CIS e altri documenti di utilità per il cittadino per essere sempre in grado di fornirli al soccorritore in caso di necessità.

1.3.1 Goals

Gli obiettivi fondamentali dell'applicazione sono enunciati in seguito.

Generali

[G1] Permettere all'utente di eseguire il reset della password tramite l'invio via mail di un link;

[G1] Mostrare una lista di numeri di emergenza che possono essere contattati in caso di necessità;

[G1] Permettere all'utente di eseguire la scansione di un codice QR e mostrare i relativi dati;

[G1] Generare partendo dai dati del cittadino la il QR code, il Profilo Sanitario Sintetico, la Carta d'Identità Salvavita, il badge e il braccialetto.

Cittadino

[G1] Permettere al cittadino di diventare utente dell'applicazione a seguito dell'inserimento dei suoi dati da parte del MMG e del volontario;

[G2] Permettere all'utente di eseguire il login nel sistema attraverso l'utilizzo di valide credenziali (email e password);

[G3] Mostrare all'utente il QR code che riassume le informazioni salvavita con la possibilità di aprirlo come immagine, salvarlo nel dispositivo o stamparlo in formato .pdf;

[G4] Mostrare all'utente i documenti fondamentali generati dai suoi dati sanitari, quali il Profilo Sanitario Sintetico, la Carta d'Identità Salvavita, il badge e il braccialetto;

[G5] Mostrare lo storico Covid dell'utente con la possibilità di aprire i documenti relativi sul web e il QR code riassuntivo degli stessi con la possibilità di aprirlo come immagine, salvarlo nel dispositivo o stamparlo in formato .pdf;

[G6] Permettere all'utente di contattare l'MMG e il volontario incaricato dell'inserimento e della gestione dei dati tramite email o numero telefonico.

[G1] Permettere al cittadino di selezionare i dati precedenti a quelli attuali e i relativi documenti in base alla data.

Volontario

[G7] Permettere al volontario di eseguire il login nel sistema attraverso l'utilizzo di valide credenziali (email e password);

[G8] Permettere al volontario di ricercare tra i cittadini a lui assegnati e selezionarli;

[G1] Permettere al volontario di selezionare per il singolo cittadino i dati scelti in base alla data in cui sono stati inseriti;

[G9] Permettere al volontario di stampare la Carta d'Identità Salvavita, il badge e il braccialetto di un cittadino;

[G10] Permettere al volontario di stampare la Carta d'Identità Salvavita, il badge e il braccialetto a più cittadini nello stesso momento.

Capitolo 2

Panoramica del progetto

2.1 Architettura

L'intero sistema può essere suddiviso in 3 parti fondamentali:

- l'applicazione web, sviluppata da Davide Laffi, e realizzata con lo scopo di essere utilizzata dall'MMG o dal Volontario per la fase di inserimento dati sanitari e salvavita relativi al Cittadino;
- l'applicazione mobile, sviluppata da me, e realizzata con lo scopo di essere utilizzata dal Cittadino per avere traccia delle sue informazioni sanitarie e salvavita e di conseguenza accedere ai documenti generati e dal Volontario che ha la facoltà di ricercare uno o più utenti con lo scopo di fornire servizi di utilità ai cittadini, quali la stampa dei loro documenti;
- il lato server, realizzato utilizzando Google Firebase, e che fornisce il servizio di autenticazione per gli utenti e il servizio di database che consente il salvataggio delle informazioni relative ai cittadini.

Nelle sezioni seguenti verranno analizzate le 3 parti nel dettaglio, focalizzando l'attenzione sulle tecnologie utilizzate.

IMMAGINE SCHEMATICA

2.2 Web Application

L'applicazione web è stata sviluppata da Davide Laffi.

2.3 Mobile Application

L'applicazione mobile è stata realizzata utilizzando Flutter.

2.3.1 Flutter

Flutter è un UI software development kit open-source creato da Google e rilasciato nel 2017 e utilizzabile per sviluppare applicazioni mobile cross-platform. Il codice di Flutter è scritto in Dart, un linguaggio di programmazione client-optimized per applicazioni su multiple piattaforme, sviluppato anch'esso da Google e utilizzato per l'implementazione di applicazioni mobile, desktop, server e web. Tra le caratteristiche principali di Flutter è bene considerare:

- **Cross-platform:** si basa su Dart, linguaggio unico che permette lo sviluppo su Android, iOS e sul web;
- **Firebase:** si integra in maniera semplice e completa con Firebase;
- **Widgets:** si compone di widgets, componenti atomiche che rappresentano elementi della UI e che possono essere uniti nella creazione di schermate pulite e complesse allo stesso tempo;
- **Plugins and Packages:** grazie alla vasta community di supporto esistono moltissimi packages in grado di implementare i più svariati servizi.

2.4 Database

Il lato server del sistema è stato realizzato utilizzando **Google Firebase**, una piattaforma online che permette di salvare, sincronizzare e utilizzare i dati generati dalle applicazione web e mobile. Dalla moltitudine di funzionalità offerte dal servizio, per lo sviluppo del sistema sono state utilizzate in particolare due di esse:

- **Autenticazione:** abilita la creazione e la gestione degli utenti per il sistema. Tra i vari metodi di sign-in disponibili è stato scelto quello classico di iscrizione e login tramite email e password. Nel servizio di autenticazione è fornita anche la possibilità del reset della password;
- **Firestore Database:** permette la creazione di collezioni e documenti che andranno a popolare i dati che verranno utilizzati dal sistema.

Capitolo 3

Implementazione

3.1 Applicazione Mobile

L'applicazione mobile è stata sviluppata utilizzando Flutter 2.0. Le varie classi dell'applicazione sono state suddivise in packages in base alla tipologia per incrementare la manutenibilità, la chiarezza e la separazione delle stesse.

3.1.1 Model

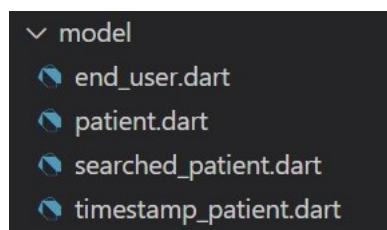


Figura 3.1: The Model files

Il package Model contiene i vari oggetti utilizzati all'interno dell'applicazione che spesso si presentano con la presenza dei vari campi, del costruttore e di metodi di utilità generale per la classe. In particolare:

- **end_user**: rappresenta l'utente finale dell'applicazione e permette di salvare le sue informazioni fondamentali ottenute da Firebase quali l'id, il codice fiscale, l'email e la tipologia di utente (cittadino o volontario);
- **patient**: rappresenta il cittadino e contiene le informazioni invariabili (codice fiscale, nome e cognome) e una mappa di **TimestampPatient** organizzata per data con la funzione di storico per i dati precedenti a quelli attuali;
- **searched_patient**: rappresenta la classe utilizzata per gestire la ricerca dei cittadini da parte del volontario;
- **timestamp_patient**: contiene tutte le informazioni sanitarie e salvavita del cittadino relative ad una certa data e metodi di utilità per accedere ai dati.

3.1.2 Screens

IMMAGINE SCHEMATICA

Il package Screens contiene le schermate principali dell'applicazione. Ogni file corrisponde ad una schermata i cui widgets sono spesso contenuti nel package Widgets. In particolare:

- **emergency_numbers**: rappresenta la schermata dei Numeri Utili e si suddivide in due tabs fondamentali. La prima tab contiene i contatti (email e numero di telefono) dell'MMG e del volontario ed è costruita utilizzando le **NumbersCard**. La seconda tab contiene i numeri di emergenza da contattare in caso di necessità ed è costruita usando le **EmergencyNumberTile**;
- **homepage**: rappresenta la schermata principale dell'applicazione per quanto riguarda il cittadino. Si suddivide in 3 tab fondamentali, ovvero
- **login**
- **qr_code_scanner**
- **volunteer**
- **wrapper**

3.1.3 Services

IMMAGINE SCHEMATICA

Il package Services contiene le classi responsabili dei principali servizi utilizzati nelle varie schermate e nella logica dell'applicazione. In particolare:

- **auth_service**: rappresenta la classe incaricata di gestire l'autenticazione degli utenti. Si compone dei seguenti metodi:
 - **getCurrentUser**: restituisce l'utente Firebase corrente;
 - **login**: esegue il login dell'utente specificato attraverso email e password e restituisce un booleano rappresentante il risultato dell'operazione;
 - **resetPassword**: permette di inviare all'utente il link per il reset della password alla mail specificata.
- **database_service**: è la classe che si occupa della gestione del database Firebase. Ha una variabile per ogni collezione (users, patients, volunteers) e i seguenti metodi:
 - **getUser**: restituisce il documento Firebase relativo all'utente corrente. Viene mappato nella classe **EndUser** attraverso il metodo **_userFromFirebase**;
 - **getPatient**: restituisce il documento Firebase relativo al cittadino corrente insieme ai dati sanitari e salvavita. Viene mappato nella classe **Patient** attraverso il metodo **_patientFromFirebase**;

- `getPatientsList`: restituisce la lista dei documenti Firebase relativi a tutti i cittadini di competenza del volontario specificato. Viene mappato nella lista di `Patient` attraverso il metodo `populatePatientsData`.
- `pdf_handler`: è la classe che si occupa della gestione dei file in formato .pdf all'interno dell'applicazione. In particolare per ogni documento viene specificato un metodo di creazione per il file. Successivamente per ogni file è presente un metodo che si occupi di:
 - aprire il file;
 - scaricare il file;
 - condividere il file;
 - stampare il file.
- `qr_code_handler`: è la classe che si occupa della gestione dei codici QR all'interno dell'applicazione. In particolare:
 - `generateQRCode`: genera il codice QR partendo da una stringa di dati;
 - `openQRCode`: apre il codice QR dopo averlo generato;
 - `saveQRCodeToGallery`: salva nella galleria del dispositivo il codice QR in formato immagine;

3.1.4 Widgets

IMMAGINE SCHEMATICA

Il package `Widgets` contiene i widgets utilizzati frequentemente nelle altre classi dell'applicazione con l'obiettivo di ridurre al minimo la duplicazione del codice. In particolare:

- `appbar.button`: rappresenta le icone cliccabili nell'appbar a cui è collegata una funzione;
- `covid.tile`: rappresenta le tiles presenti nello storico Covid, caratterizzate dall'evento Covid, dalla data in cui è avvenuto e dal link al documento correlato;
- `emergency_numbers.tile`: rappresenta le tiles presente nella tab Emergenza dei Numeri Utili, caratterizzate da un'icona, la descrizione del contatto e il numero telefonico;
- `form.text.field`:
- `function.button`:
- `function.card`:
- `function.icon`:

- **numbers_card**: rappresenta le cards presenti nella tab Contatti dei Numeri Utili, caratterizzate da un'icona, il nome, la professione (medico o volontario) e i contatti (email e numero di telefono);
- **processing_indicator**: rappresenta l'indicatore di caricamento presente nelle schermate dell'applicazione;
- **radio_tile**:
- **volunteer_card**: rappresenta le cards relative ai cittadini presenti nella Schermata del Volontario. Ognuna di esse si compone del nome del cittadino, il codice fiscale e i pulsanti per la stampa dei documenti relativi.

3.2 Lato Server

Il lato server dell'applicazione è interamente sviluppato attraverso **Google Firebase** che offre due funzionalità fondamentali per l'intero sistema:

- **Autenticazione**: gestisce il login degli utenti e la creazione degli accounts;
- **Firestore Database**: permette la creazione di collezioni e documenti con cui eseguire il salvataggio dei dati degli utenti e dei cittadini.

Il vantaggio fondamentale nell'utilizzo di Firebase consiste nella sua perfetta integrazione con Flutter in quanto sono entrambi servizi Google.

3.2.1 Autenticazione

Poichè l'intero sistema si basa su 3 profili utente principali (MMG, volontario e cittadino), risulta necessario un servizio di autenticazione efficiente in grado servire questo scopo. In particolare l'accesso avviene tramite email e password, considerato il metodo di autenticazione più comune e scelto per:

- consentire ai volontari incaricati di creare gli account dei cittadini;
- consentire alle persone meno pratiche o tecnologiche di ricordare solamente questi due parametri senza la necessità di utilizzare ulteriori fattori di autenticazione che andrebbero a complicare il processo;
- uniformare il processo di login basandosi sul concetto che gli account social non sono posseduti da tutti, ma l'utilizzo delle mail risulta pressoché essenziale anche per i servizi più generici.

In generale i metodi di Firebase utilizzati in fase di autenticazione sono essenzialmente i seguenti:

- **signInWithEmailAndPassword**: consente di eseguire il login di un utente attraverso email e password nel caso in cui le credenziali fossero valide;
- **sendPasswordResetEmail**: permette di inviare un link per il reset della password all'indirizzo email specificato.

3.2.2 Firestore Database

Cloud Firestore è un database cloud NoSQL che permette di salvare e utilizzare i dati lato client in maniera istantanea e rapida. I dati vengono sincronizzati in tempo reale e viene offerto supporto per il lavoro indipendentemente dalla latenza di rete. Le principali funzionalità offerte, reperibili nella documentazione ufficiale, sono:

- **flessibilità:** permette di organizzare i dati in collezioni e documenti;
- **interrogazione espressiva:** le query possono essere utilizzate per richiedere documenti specifici o intere collezioni, attraverso l'uso di filtri e criteri di ordinamento;
- **supporto offline:** i dati utilizzati dall'applicazione vengono memorizzati automaticamente nella cache in modo che il sistema possa funzionare anche in assenza di rete.

Il database del sistema è stato organizzato in 4 collezioni fondamentali:

- **users:** contiene un documento per ogni utente, caratterizzato dall'id utente assegnato in fase di creazione dell'account di Firebase. All'interno di ogni documento sono specificati id, codice fiscale, email e tipologia di account;
- **patients:** contiene un documento per ogni cittadino, caratterizzato dal relativo codice fiscale. Ogni documento è collegato a sua volta ad una sottocollezione che tiene in memoria tutti i dati relativi allo storico del relativo cittadino;
- **doctors:** contiene un documento per ogni MMG, caratterizzato dal relativo codice fiscale. Ogni documento contiene i dati anagrafici del relativo medico;
- **volunteers:** contiene un documento per ogni volontario, caratterizzato dal relativo codice fiscale. Ogni documento contiene i dati anagrafici del relativo volontario.

Di seguito un diagramma ER che mostra più nel dettaglio le interazioni tra le varie collezioni e i campi dei singoli documenti.

IMMAGINE SCHEMATICA

3.3 External Services and Libraries

Durante la fase di sviluppo dell'applicazione mobile è stato fatto largo utilizzo dei packages forniti dal sito `pub.dev`. Le motivazioni alla base di questa scelta sono dovute alla possibilità di sfruttare la presenza di funzionalità già esistenti per arricchire l'applicazione evitando il riutilizzo di codice e favorendo la pulizia dello stesso. Nell'elenco seguente verranno menzionati tutti i packages utilizzati con le relative funzionalità principali:

- `firebase_core`: per connettere l'applicazione ai vari servizi offerti da Firebase;
- `firebase_auth`: per utilizzare le funzionalità di autenticazione offerte da Firebase;
- `cloud_firestore`: per utilizzare le funzionalità del database cloud offerto da Firebase;
- `path_provider`: per ottenere percorsi comuni di cartelle nei dispositivi in cui viene eseguita l'applicazione;
- `pdf`: per generare i file .pdf utilizzati nell'applicazione e modificarli;
- `qr_flutter`: per generare i codici QR utilizzati per i dati salvavita e per lo storico Covid;
- `image_gallery_saver`: per salvare nella galleria del dispositivo i QR code;
- `fluttertoast`: per mostrare messaggi in rilievo a seguito di operazioni eseguite con successo;
- `open_file`: per aprire i file .pdf in maniera nativa nel sistema di appartenenza;
- `downloads_path_provider`: per ottenere il path relativo alla cartella di Download nei dispositivi mobili;
- `permission_handler`: per gestire i permessi all'interno dell'applicazione;
- `flutter_barcode_scanner`: per fornire la funzionalità di scansione dei codici QR;
- `url_launcher`: per aprire pagine internet, indirizzi email o numeri telefoni con l'applicazione di sistema correlata;
- `humanitarian_icons`: per avere icone aggiuntive da utilizzare nella schermata relativa ai contatti di emergenza;
- `printing`: per abilitare le funzionalità di condivisione e stampa dei vari documenti;
- `path`: per eseguire operazioni di manipolazione sui percorsi del sistema in uso;
- `universal_html`: per gestire in maniera cross-platform il package html;
- `material_floating_search_bar`: per generare una barra di ricerca consona al Material design;
- `flutter_login`: per generare una schermata di login pulita ed efficiente;
- `unicorndial`: per utilizzare un floating action button con molteplici funzionalità.

Capitolo 4

Funzionalità

4.1 Panoramica dell'Applicazione

L'applicazione è composta da una schermata principale e da un'insieme di pagine che forniscono diverse funzionalità. In particolare l'homepage è composta da 3 tabs il cui scopo è rispettivamente di:

1. mostrare il QR code contenente le informazioni salvavita del cittadino con la possibilità di aprirlo come immagine, salvarlo nel dispositivo o stamparlo;
2. mostrare i documenti originali generati dai dati sanitari e dalle informazioni salvavita del cittadino con la possibilità di aprirli, salvarli nel dispositivo o stamparli;
3. mostrare uno storico relativo agli ultimi eventi relativi all'argomento Covid del cittadino (come ad esempio i tamponi o le vaccinazioni effettuati) e un QR code che riassume tutti questi eventi.

Oltre alla schermata principale sono presenti ulteriori funzionalità che permettono all'utente di:

- contattare il volontario e l'MMG responsabili dell'inserimento dei dati tramite email o numero telefonico;
- avere una lista dei numeri utili di emergenza (quali forze dell'ordine, pronto soccorso, ...), con la possibilità di contattarli in caso di necessità;
- scannerizzare un altro QR code attraverso uno scanner integrato all'applicazione con la possibilità di mostrare i dati salvavita di un altro cittadino. Questa funzionalità è pensata per essere utilizzata dai soccorritori in caso di necessità.

Oltre alle schermate appena descritte è presente un'intera parte dell'applicazione riservata al volontario con lo scopo di abilitare la ricerca di uno o più cittadini con la facoltà di stampare per essi i documenti generati dai loro dati.

4.2 Descrizione delle Funzionalità

Nelle sezioni successive verranno analizzate nel dettaglio le funzionalità introdotte nel paragrafo precedente cercando di mettere in luce gli scopi e i modi d'utilizzo pensati.

4.2.1 Registrazione

La registrazione degli accounts avviene in due modi fondamentali in base alla tipologie di utente:

- **volontario:** la creazione dell'account del volontario (e anche quello dell'MMG) avviene per mano dell'amministratore di sistema a seguito di una richiesta. In questo modo è possibile assicurare solamente alle persone corrette il privilegio di esercitare la propria professione e avere l'accesso ai dati sensibili dei cittadini;
- **cittadino:** l'account del cittadino viene creato dal volontario a seguito dell'inserimento dei dati sanitari da parte dell'MMG. In questo modo il cittadino può avere accesso all'applicazione solo a seguito dell'inserimento dei suoi dati.

Le password degli accounts sono generate in maniera randomica e inoltrate agli utenti via mail. A seguito di ciò l'utente sarà libero di modificarla secondo le proprie preferenze.

4.2.2 Login

La schermata di Login permette agli utenti di accedere al sistema fornendo credenziali valide e fornite a seguito del processo di registrazione analizzato precedentemente. Gli utenti sono in grado di modificare la propria password cliccando su "Hai dimenticato la password?" che invierà all'utente una mail contenente un link per il reset della stessa. Sia il cittadino che il volontario condividono la stessa schermata di Login. Il sistema controlla le credenziali e reindirizza l'utente alla schermata corretta.

4.2.3 Schermata del Volontario

Il volontario ha accesso ad una singola schermata nell'applicazione mobile in quanto la maggior parte delle sue facoltà sono correlate all'utilizzo dell'applicazione web. Il volontario tramite un'apposita barra di ricerca è in grado di ricercare attraverso nome o codice fiscale gli utenti di sua competenza e selezionarli. A seguito della selezione è in grado di eseguire la stampa dei documenti fondamentali (CIS, badge e braccialetto) del singolo o del gruppo di utenti selezionati.

4.2.4 Homepage

L'Homepage è la schermata principale dell'applicazione ha cui ha accesso il cittadino. Questa schermata è suddivisa in 3 tabs principali che forniscono diverse funzionalità:

1. la prima tab mostra il QR code salvavita del cittadino contenente:

- Nome e cognome;
- Data di nascita;
- Gruppo sanguigno;
- Primo contatto ICE;
- Secondo contatto ICE;
- Lista delle eventuali patologie;
- Lista delle eventuali allergie;
- Informazioni aggiuntive.

e permette all'utente di aprirlo come immagine, salvarlo nella galleria del dispositivo o stamparlo in formato .pdf;

2. la seconda tab mostra i documenti principali generati dai dati sanitari e salvavita inseriti dall'MMG e dal volontario incaricato:

- Profilo Sanitario Sintetico: contiene tutti i dati sanitari e salvavita relativi al cittadino;
- Carta d'Identità Salvavita: mostra la CIS del cittadino contenente la foto, il QR code, i contatti ICE e le informazioni salvavita;
- Badge: mostra la foto, il QR code e i contatti ICE;
- Braccialetto: contiene i loghi delle società partner e il QR code.

3. la terza tab mostra gli eventi relativi alla storia Covid del cittadino, come ad esempio i tamponi effettuati o le vaccinazioni con la possibilità di aprire i documenti relativi inseriti dall'MMG o dal volontario. La schermata contiene anche un QR code riassuntivo dei vari eventi.

Tramite l'Homepage il cittadino è in grado di accedere alle altre funzionalità dell'applicazione (accessibili anche dalla schermata di Login) o di disconnettersi.

4.2.5 Numeri Utili

La schermata dei Numeri Utili è accessibile sia dalla schermata di Login, sia dall'Homepage e presenta due funzionalità principali:

- permette al cittadino loggato di contattare l'MMG e il volontario incaricato della gestione dei suoi dati tramite email o numero telefonico;
- provvede ad una lista di numeri di emergenza che possono essere contattati dall'utente in caso di necessità:
 - carabinieri;
 - polizia di stato;
 - vigili del fuoco;
 - guardia di finanza;
 - emergenza sanitaria.

4.2.6 Scanner QR Code

La schermata dello Scanner è accessibile sia dalla schermata di Login, sia dall'Homepage. Questa funzionalità è stata pensata per essere utilizzata dai soccorritori che, senza dover necessariamente fare l'accesso all'applicazione, possono utilizzare lo scanner QR per avere accesso immediato alle informazioni salvavita dei cittadini in difficoltà. L'applicazione infatti, a seguito della scansione, mostra i dati del cittadino in forma leggibile ed organizzata.

Capitolo 5

Interfaccia Utente e Casi d'Uso

5.1 UI Design

Il design dell'interfaccia utente è basato principalmente sulle linee guida del Material design, cercando di focalizzarsi simultaneamente su:

- semplicità d'uso: garantita dalle schermate pulite e prive di troppi elementi a schermo, cercando di rendere chiaro all'utente lo scopo di ogni singola componente;
- efficienza: garantita dalla presenza di singole operazioni atomiche che non bloccano eccessivamente l'utilizzo dell'applicazione all'utente.

Nei prossimi paragrafi verranno analizzate in dettaglio le scelte stilistiche e progettuali volte a soddisfare i requisiti appena definiti, focalizzandosi sulle motivazioni di tali scelte.

5.1.1 Logo

5.1.2 Schermate Principali

5.1.3 Layout Web

5.2 Use Cases

Nelle sezioni successive verranno analizzate i casi d'uso più frequenti, avendo cura di specificare:

- Scenario: il titolo riassuntivo del caso d'uso analizzato;
- Condizione d'ingresso: requisiti minimi per l'esecuzione del caso d'uso;
- Flusso degli eventi: azioni o avvenimenti che caratterizzano il caso d'uso;
- Condizione d'uscita: evento finale che termina il flusso degli eventi;
- Eccezioni: possibili avvenimenti che causano l'interruzione o privano l'esecuzione degli eventi.

5.2.1 Login

Scenario	Login utente
Condizione d'ingresso	L'utente ha scaricato l'applicazione o ha eseguito l'accesso attraverso il web browser L'utente ha già un account creato dal volontario incaricato
Flusso degli eventi	<ol style="list-style-type: none">1. L'utente inserisce la sua email nel campo "Email" della schermata2. L'utente inserisce la sua password nel campo "Password" della schermata3. L'utente clicca sul bottone di "Login"
Condizione d'uscita	L'applicazione reindirizza l'utente nella schermata corretta, in base alla tipologia di account (cittadino o volontario)
Eccezioni	<ul style="list-style-type: none">• L'utente non ha inserito credenziali valide• L'utente non ha inserito le credenziali in tutti i campi obbligatori• Il sistema non è in grado di completare la richiesta a causa di un errore interno

Scenario	Reset della password
Condizione d'ingresso	L'utente ha scaricato l'applicazione o ha eseguito l'accesso attraverso il web browser L'utente ha già un account creato dal volontario incaricato
Flusso degli eventi	<ol style="list-style-type: none"> 1. L'utente clicca sulla scritta "Password dimenticata?" 2. L'applicazione mostra il form per il reset della password 3. L'utente inserisce la sua email nel campo "Email" della schermata 4. L'utente clicca sul bottone di "Reset"
Condizione d'uscita	L'applicazione mostra un messaggio di successo
Eccezioni	<ul style="list-style-type: none"> • L'utente non ha inserito credenziali valide • L'utente non ha inserito l'email nel campo relativo • Il sistema non è in grado di completare la richiesta a causa di un errore interno

5.2.2 Logout

Scenario	Cambio schermata
Condizione d'ingresso	L'utente è nella schermata di Homepage o nella Schermata del Volontario
Flusso degli eventi	<ol style="list-style-type: none"> 1. L'utente clicca sull'icona relativa alla logout in alto a sinistra
Condizione d'uscita	L'applicazione esegue il logout dell'utente e lo reindirizza alla schermata di Login
Eccezioni	<ul style="list-style-type: none"> • Il sistema non è in grado di completare la richiesta a causa di un errore interno

5.2.3 Homepage

Scenario	Cambio schermata
Condizione d'ingresso	L'utente è nella schermata di Homepage
Flusso degli eventi	<ol style="list-style-type: none"> 1. L'utente clicca sull'icona relativa alla schermata scelta in alto a destra (Numeri Utili o Scanner QR)
Condizione d'uscita	L'applicazione reindirizza l'utente nella schermata scelta
Eccezioni	<ul style="list-style-type: none"> • Il sistema non è in grado di completare la richiesta a causa di un errore interno

Scenario	Scelta documenti
Condizione d'ingresso	L'utente è nella schermata di Homepage
Flusso degli eventi	<ol style="list-style-type: none"> 1. L'utente clicca sull'icona relativa al menu in alto a destra della schermata 2. L'applicazione mostra le date relative allo storico dell'utente 3. L'utente clicca sulla data scelta
Condizione d'uscita	L'applicazione aggiorna i dati del cittadino mostrando quelli relativi alla data scelta
Eccezioni	<ul style="list-style-type: none"> • Il cittadino non dispone di altre date tra cui scegliere oltre a quella attuale • Il sistema non è in grado di completare la richiesta a causa di un errore interno

Scenario	Codice QR
Condizione d'ingresso	L'utente è nella schermata di Homepage, nella tab Codice QR
Flusso degli eventi	<ol style="list-style-type: none">1. L'utente clicca sul bottone:<ul style="list-style-type: none">• "Apri"• "Salva"• "Stampa"2. L'applicazione esegue l'azione richiesta
Condizione d'uscita	L'applicazione: <ul style="list-style-type: none">• apre il codice QR come immagine• salva nel dispositivo il codice QR come immagine• apre la schermata di stampa per il codice QR in formato .pdf
Eccezioni	<ul style="list-style-type: none">• Il sistema non è in grado di completare la richiesta a causa di un errore interno

Scenario	Informazioni
Condizione d'ingresso	L'utente è nella schermata di Homepage, nella tab Informazioni
Flusso degli eventi	<ol style="list-style-type: none">1. L'utente sceglie il documento di interesse:<ul style="list-style-type: none">• Dati• Badge• CIS• Braccialetto2. L'utente clicca sull'icona relativa a:<ul style="list-style-type: none">• "Apri"• "Salva"• "Stampa"3. L'applicazione genera il documento ed esegue l'azione richiesta
Condizione d'uscita	L'applicazione: <ul style="list-style-type: none">• apre il documento in formato .pdf• salva nel dispositivo il documento in formato .pdf• apre la schermata di stampa per documento in formato .pdf
Eccezioni	<ul style="list-style-type: none">• Il sistema non è in grado di completare la richiesta a causa di un errore interno

Scenario	Covid19 - QR code
Condizione d'ingresso	L'utente è nella schermata di Homepage, nella tab Covid19
Flusso degli eventi	<ol style="list-style-type: none"> 1. L'utente clicca sul bottone: <ul style="list-style-type: none"> • "Apri" • "Salva" • "Stampa" 2. L'applicazione esegue l'azione richiesta
Condizione d'uscita	L'applicazione: <ul style="list-style-type: none"> • apre il codice QR come immagine • salva nel dispositivo il codice QR come immagine • apre la schermata di stampa per il codice QR in formato .pdf
Eccezioni	<ul style="list-style-type: none"> • Il sistema non è in grado di completare la richiesta a causa di un errore interno

Scenario	Covid19 - Storico
Condizione d'ingresso	L'utente è nella schermata di Homepage, nella tab Covid19
Flusso degli eventi	<ol style="list-style-type: none"> 1. L'utente sceglie dalla lista l'evento di interesse 2. L'utente clicca sul bottone "Apri" 3. L'applicazione esegue l'azione richiesta
Condizione d'uscita	L'applicazione apre il documento richiesto nella pagina web di provenienza
Eccezioni	<ul style="list-style-type: none"> • L'utente non ha alcun dato nello storico Covid • Il sistema non è in grado di completare la richiesta a causa di un errore interno

5.2.4 Schermata del Volontario

Scenario	Ricerca cittadino
Condizione d'ingresso	Il volontario ha scaricato l'applicazione o ha eseguito l'accesso attraverso il web browser Il volontario ha già un account
Flusso degli eventi	<ol style="list-style-type: none">1. Il volontario inserisce il nome (o il codice fiscale) del cittadino di sua competenza nella barra di ricerca che presenta la scritta "Cerca..."2. L'applicazione mostra i cittadini che soddisfano la ricerca del volontario3. Il volontario seleziona il cittadino (o i cittadini) desiderato cliccando il pulsante "Aggiungi"
Condizione d'uscita	L'applicazione aggiunge alla lista il cittadino selezionato e lo mostra nella schermata
Eccezioni	<ul style="list-style-type: none">• Il volontario ha inserito un nome (o un codice fiscale) inesistente• Il cittadino non fa parte delle competenze del volontario• Il sistema non è in grado di completare la richiesta a causa di un errore interno

Scenario	Rimozione cittadino
Condizione d'ingresso	<p>Il volontario ha scaricato l'applicazione o ha eseguito l'accesso attraverso il web browser</p> <p>Il volontario ha già un account</p> <p>Il volontario ha selezionato un cittadino attraverso la barra di ricerca</p>
Flusso degli eventi	<ol style="list-style-type: none"> 1. Il volontario seleziona il cittadino di interesse tra quelli presenti nella lista 2. Il volontario clicca sull'icona ad X nella scheda relativa al cittadino
Condizione d'uscita	L'applicazione rimuove dalla lista il cittadino selezionato
Eccezioni	<ul style="list-style-type: none"> • Il sistema non è in grado di completare la richiesta a causa di un errore interno

Scenario	Scelta documenti cittadino
Condizione d'ingresso	<p>Il volontario ha scaricato l'applicazione o ha eseguito l'accesso attraverso il web browser</p> <p>Il volontario ha già un account</p> <p>Il volontario ha selezionato un cittadino attraverso la barra di ricerca</p>
Flusso degli eventi	<ol style="list-style-type: none"> 1. Il volontario seleziona il cittadino di interesse tra quelli presenti nella lista 2. Il volontario clicca sull'icona rappresentante il menu nella scheda relativa al cittadino 3. Il volontario clicca sulla data di interesse
Condizione d'uscita	L'applicazione aggiorna i dati del cittadino mostrando quelli relativi alla data scelta
Eccezioni	<ul style="list-style-type: none"> • Il cittadino selezionato non dispone di altre date tra cui scegliere oltre a quella attuale • Il sistema non è in grado di completare la richiesta a causa di un errore interno

Scenario	Stampa singola
Condizione d'ingresso	<p>Il volontario ha scaricato l'applicazione o ha eseguito l'accesso attraverso il web browser</p> <p>Il volontario ha già un account</p> <p>Il volontario ha selezionato un cittadino attraverso la barra di ricerca</p>
Flusso degli eventi	<ol style="list-style-type: none"> 1. Il volontario seleziona il cittadino di interesse tra quelli presenti nella lista 2. Il volontario seleziona il documento di interesse tra Badge, CIS e Braccialetto 3. Il volontario clicca sul pulsante "Stampa"
Condizione d'uscita	L'applicazione mostra la schermata di stampa del documento selezionato in formato .pdf
Eccezioni	<ul style="list-style-type: none"> • Il sistema non è in grado di completare la richiesta a causa di un errore interno

Scenario	Stampa multipla
Condizione d'ingresso	<p>Il volontario ha scaricato l'applicazione o ha eseguito l'accesso attraverso il web browser</p> <p>Il volontario ha già un account</p> <p>Il volontario ha selezionato più cittadini attraverso la barra di ricerca</p>
Flusso degli eventi	<ol style="list-style-type: none"> 1. Il volontario clicca sul pulsante in basso a destra dello schermo che presenta l'icona di stampa 2. Il volontario seleziona il documento di interesse cliccando sull'icon corrispondente
Condizione d'uscita	L'applicazione mostra la schermata di stampa dei documenti selezionati in formato .pdf
Eccezioni	<ul style="list-style-type: none"> • Il sistema non è in grado di completare la richiesta a causa di un errore interno

5.2.5 Numeri Utili

Scenario	Contatti
Condizione d'ingresso	<p>L'utente ha scaricato l'applicazione o ha eseguito l'accesso attraverso il web browser</p> <p>L'utente ha già un account creato dal volontario incaricato</p> <p>L'utente è nella schermata Numeri Utili, nella tab Contatti</p>
Flusso degli eventi	<ol style="list-style-type: none"> 1. L'utente sceglie il contatto desiderato tra MMG e volontario 2. L'utente clicca sulla forma di comunicazione desiderata (numero telefonico o email)
Condizione d'uscita	L'applicazione apre l'applicazione del dispositivo in grado di gestire la richiesta
Eccezioni	<ul style="list-style-type: none"> • Il sistema non è in grado di completare la richiesta a causa di un errore interno

Scenario	Contatti
Condizione d'ingresso	<p>L'utente ha scaricato l'applicazione o ha eseguito l'accesso attraverso il web browser</p> <p>L'utente è nella schermata Numeri Utili, nella tab Emergenza</p>
Flusso degli eventi	<ol style="list-style-type: none"> 1. L'utente sceglie il contatto desiderato tra Carabinieri, Polizia di Stato, Vigili del Fuoco, Guardia di Finanza e Emergenza Sanitaria 2. L'utente clicca sul contatto desiderato
Condizione d'uscita	L'applicazione apre l'applicazione del dispositivo in grado di gestire la richiesta
Eccezioni	<ul style="list-style-type: none"> • Il sistema non è in grado di completare la richiesta a causa di un errore interno

5.2.6 QR Code Scanner

Scenario	Scansione QR
Condizione d'ingresso	L'utente ha scaricato l'applicazione L'utente è nella schermata QR Code Scanner
Flusso degli eventi	<ol style="list-style-type: none">1. L'utente clicca sul pulsante "Scan"2. L'applicazione attiva la camera del dispositivo in attesa della scansione3. L'utente esegue la scansione di un codice QR
Condizione d'uscita	L'applicazione mostra i dati relativi al codice QR scansionato
Eccezioni	<ul style="list-style-type: none">• L'utente cerca di eseguire la scansione di un codice QR che non soddisfa i requisiti dell'applicazione• Il sistema non è in grado di completare la richiesta a causa di un errore interno

5.3 User-Flow Diagram

Il grafico seguente mostra nel dettaglio tutte le possibili interazioni tra le schermate dell'applicazione attraverso un user flow diagram.

Capitolo 6

Discussions and Conclusion

Discussione sui risultati raggiunti e sulle possibili migliorie.

Capitolo 7

Bibliografia

Di seguito le fonti utilizzate per la stesura del documento e la realizzazione del progetto:

- Medici Volontari Italiani;
- Comune di Milano;