**Systemy Telemedyczne**

Autorzy:

Patrycja Haraburda

Paulina Gralicka

12. Aplikacja mobilna – e-karty gorączkowe dla oddziału szpitalnego z czytaniem znaczników NFC w celu identyfikacji pacjentów

1. **Opis projektu**

W wyniku zrealizowania projektu została wykonana aplikacja mobilna „e-karty gorączkowe” dla oddziału szpitalnego z czytaniem znaczników NFC. Jest to aplikacja mobilna dla systemu Android napisana w języku Kotlin wykorzystująca bazę danych Firebase Firestore.

* 1. **Wymagania systemowe**
* Minimalna wersja Android 6.0
* Telefon z NFC

1. **Instrukcja użytkownika systemu**

Aplikacja ta posiada dwa profile użytkowników, profil pielęgniarki oraz lekarza. Nowy użytkownik rejestrując się, podaje swoje dane osobowe, e-mail, hasło i wybiera profil użytkownika – lekarz, pielęgniarka. Po zarejestrowaniu aplikacja przechodzi do ekranu logowania.

Po zalogowaniu się do profilu lekarza jest możliwe wykonanie dwóch akcji. Pierwszą z nich jest dodanie nowego pacjenta. W tym celu należy wpisać dane pacjenta, a następnie zeskanować pacjenta przy użyciu znacznika NFC. Dzięki temu nowy pacjent zostanie przypisany do konkretnego znacznika. Drugą możliwością jest wejście w historię pomiarów, która otwiera ekran z listą wszystkich zapisanych pacjentów wraz z pomiarami ich temperatury.

Po zalogowaniu się do profilu pielęgniarki również jest możliwe wykonanie dwóch czynności. Pierwszą z nich jest dodanie pomiaru temperatury i przypisanie jej do konkretnego pacjenta. Po dodaniu zostanie on zapisany w bazie danych. Drugą z możliwości jest zobaczenie historii pomiarów konkretnego pacjenta, poprzez sczytanie jego znacznika NFC.

Z każdego z profili użytkowników lekarza oraz pielęgniarki można się z łatwością wylogować. Opcja wylogowania znajduje się w głównym panelu zarówno pielęgniarki jak i lekarza. Powoduje to powrót do opcji logowania, a także rejestracji.

1. **Opis instalacji**

W celu zainstalowania aplikacji „karty e-gorączkowe” na telefonie należy wykonać kilka prostych kroków. W pierwszej kolejności należy pobrać repozytorium, które jest dostępne na https://gitlab-stud.elka.pw.edu.pl.

Kolejnym krokiem jest pobranie środowiska Android Studio z oficjalnej strony developer.android.com. Po zainstalowaniu aplikacji, należy wybrać File  New  Import Project i wybrać zapisany wcześniej projekt i uruchomić go.

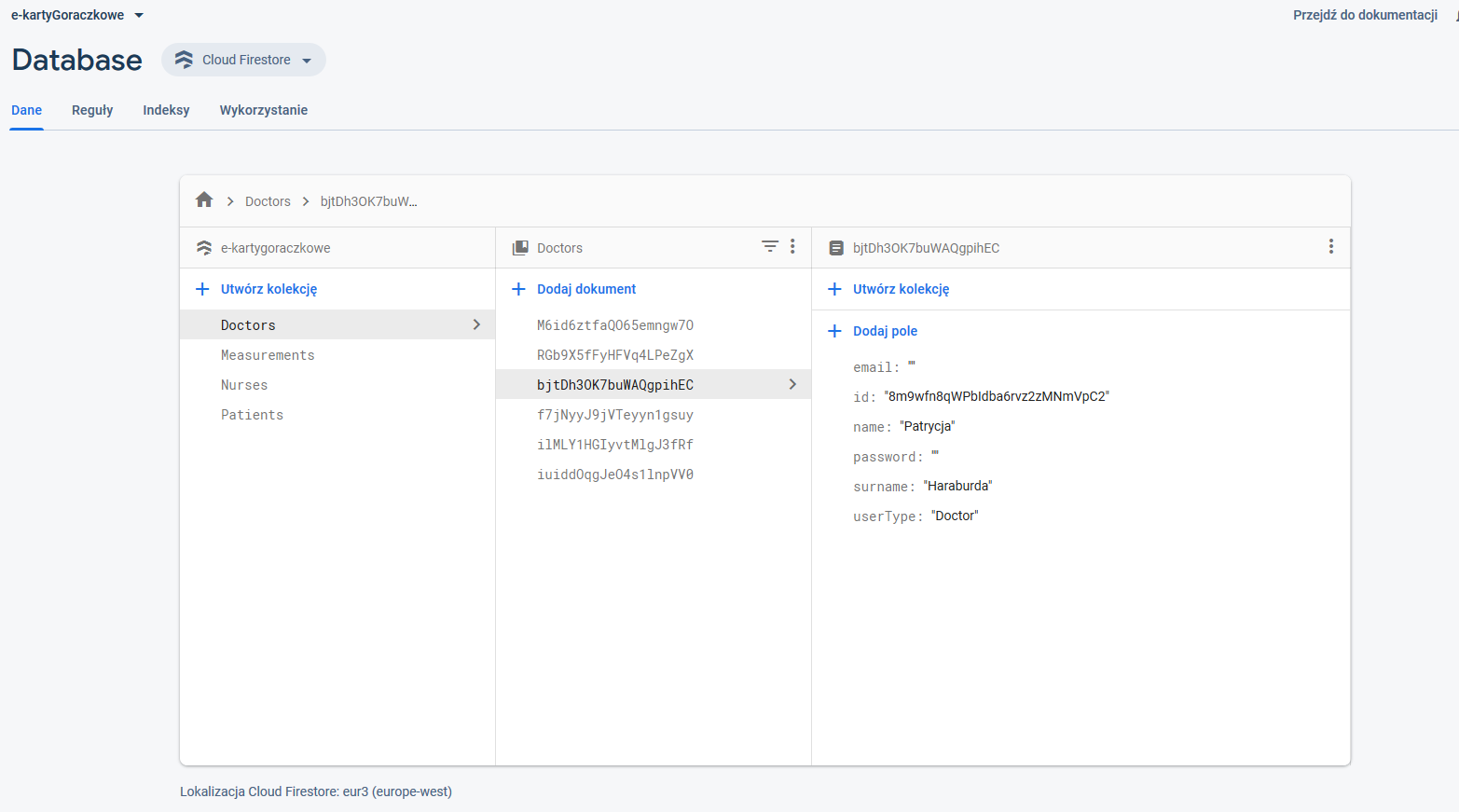
Przed ostatecznym zainstalowaniem aplikacji na telefonie należy włączyć w nim opcje deweloperskie. W tym celu należy włączyć ustawienia telefonu i wyszukać pozycję „Informacje o telefonie” a następnie na liście opcji „Informacje o oprogramowaniu”. W celu włączenia dostępu do opcji programistycznych w telefonie należy siedem razy nacisnąć pozycję „Numer wersji”. Kiedy będą już aktywne, na ekranie powinna pojawić się stosowna informacja. W kolejnym kroku należy wrócić do ustawień telefonu i otworzyć nową pozycję „Opcje programistyczne”. Następnie należy wyszukać na liście „Debugowanie USB” i włączyć.

W ostatnim kroku należy podłączyć telefon do komputera i upewnić się, że program widzi podłączony telefon w dostępnych urządzeniach. Następnie należy uruchomić program, a aplikacja automatycznie zainstaluje się na telefonie.

1. **Opis bazy danych**

W projekcie została wykorzystana baza danych Firebase Firestore, która jest nierelacyjną bazą danych, uaktualniającą dane w czasie rzeczywistym, łatwo skalowalną i jest dostępna za darmo w wersji podstawowej.

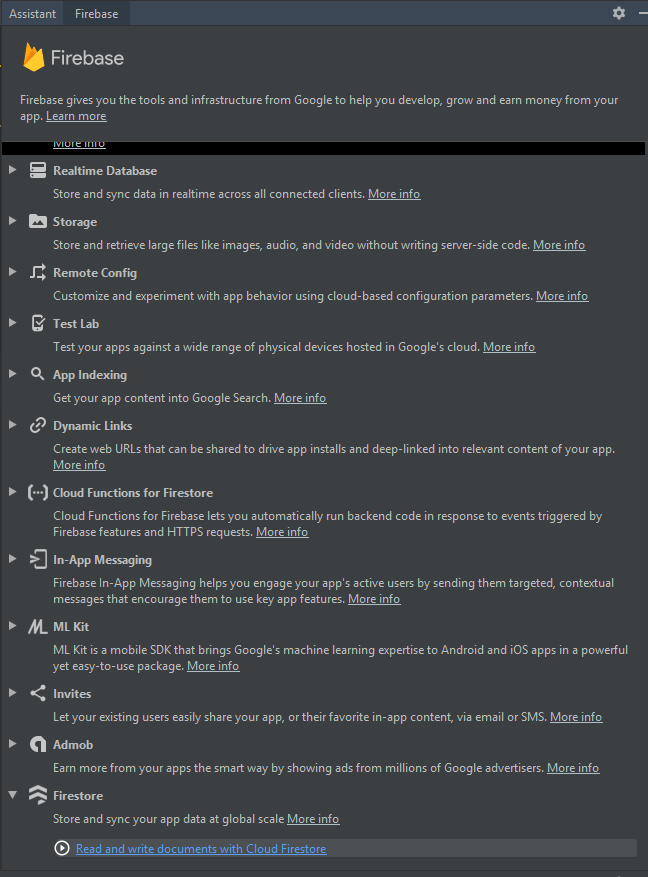
Jej struktura składa się z kolekcji, w których przechowuje się dokumenty. Dokumenty mogą zawierać takie pola, jak: String, numer, boolean, map, array, null, timestamp, geopoint, reference. Dokument może również przechowywać kolekcję dokumentów.



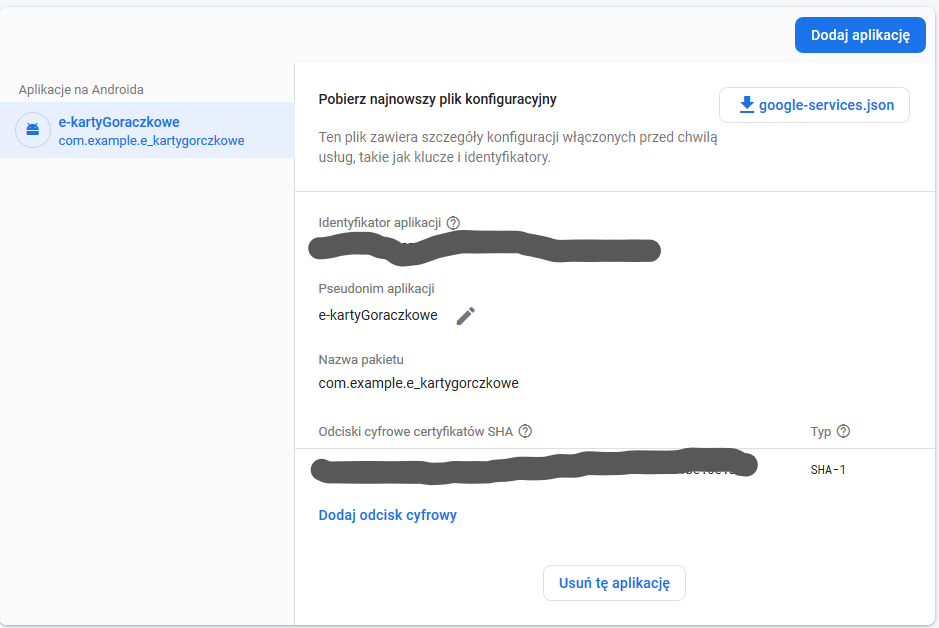
Z poziomu Firebase console określa się również reguły dostępu do bazy danych. Domyślnie, przez pierwsze 30 dni, wszyscy użytkownicy mają prawa read, write.

Możliwy jest też podgląd do podstawowej analityki, takiej jak ilość odczytów, zapisów, usunięć i aktywnych połączeń.

Jedną z podstawowych zalet korzystania z usług Firebasa, jest to, że są one w pełni zintegrowane z systemem Android. Do tego stopnia, że korzystając z Android studio, mamy dostęp do tutoriali, które krok po kroku tłumaczą, jak z nich korzystać.



Połączenie pomiędzy bazą danych a aplikacją ustanawia się poprzez dodanie w katalogu app pliku „gogle-services.json”. Jest to plik, który jest generowany po stworzeniu projektu na Firebasie , a w nim aplikacji na konkretną platformę.



Plik ten posiada takie pola, jak:

* Firebase\_url
* Project\_id
* Oauth\_client
* Api\_key
* Services

Korzystanie z tak skonfigurowanej aplikacji jest bardzo proste.

Pierwszym krokiem jest uzyskanie instancji bazy danych.

private val dbInstance: FirebaseFirestore = FirebaseFirestore.getInstance()

a następnie kolekcji.

dbInstance.collection("Doctors")

Dodawanie dokumentów odbywa się poprzez funkcję

.add(user)

Gdzie user jest obiektem, przechowującym wyłącznie pola.

Natomiast pobieranie obiektów odbywa się poprzez funkcję

.get()

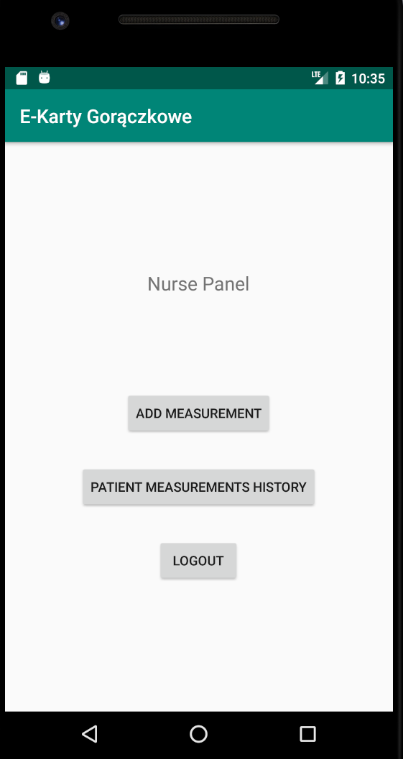
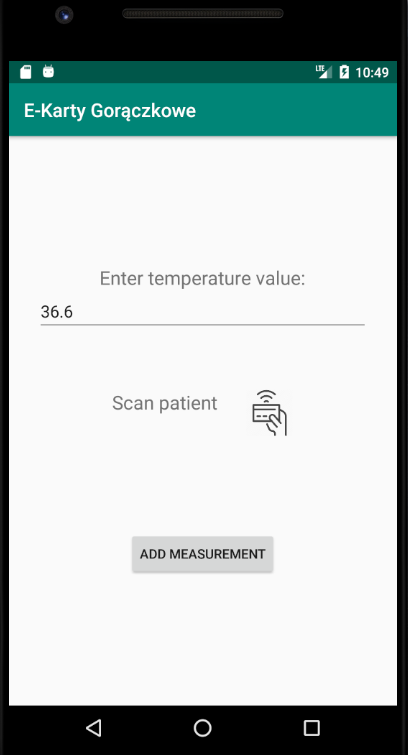
która pobiera dokumenty kolekcji. Konieczne jest przeparsowanie dokumentu do konkretnego obiektu.

.addOnSuccessListener **{** documents **->** if (documents.*isEmpty*) {  
 emitter.onComplete()  
 } else {  
 emitter.onSuccess(documents.*first*().toObject(User::class.*java*))  
 }  
**}**

Możliwe jest również dodawanie warunków.

.whereEqualTo("id", id)

1. **Przykładowe ekrany aplikacji**



ADD MEASURMENT

1. **Podział zadań w zespole**

Koncepcja – Patrycja Haraburda, Paulina Gralicka

Implementacja – Patrycja Haraburda

Dokumentacja – Patrycja Haraburda , Paulina Gralicka