DOCUMENTAȚIA PROIECTULUI

MediCare

STUDENȚI

Ivan Bogdan

Hrihoreanu Radu

Prezentarea proiectului

Aplicația MediCare este o aplicație în domeniul medical creeată pentru a ușura acțiunile lungi și repetitive precum așteptatul cu orele la codă pentru a face o programare, prezentarea la cabinet pentru a afla informații precum orarul și numele asistentei cu care trebuie sa iei legătura pentru putea într-un final să fi văzut de un doctor.

Aceste informații pot fi ușor informatizate, în unele locuri sunt, în altele nu, ideea principală fiind că informația este răspândită prin jur și nu știi unde să cauți. În contextul actual al pandemiei, grupurile mari de oameni ce se strâng din cauza dezorganizării acestor instituții medicale crește riscul exponențial de îmbolnăvire. O bună organizare a programărilor la doctor, o situație detaliată a resurselor medicale din spital poate reduce foarte mult riscul infectării si oferă siguranță și încredere oamenilor. În acest scop a luat naștere MediCare.

MediCare își propune rezolvarea tuturor problemelor menționate anterior, având ca și scop principal buna organizare a programărilor la medic dar și gestionarea lor. Astfel, nefiind nevoie de contact fizic înainte de consultul propriu-zis MediCare reduce șansele de infectare exponențial. Aplicația permite pacienților, atât și medicilor de a stabili într-un mod cât mai eficient următoare întâlnire, sau de informare a părții opuse de o schimbare neprevăzuta ce duce la anularea programării.

La finalul unei programări, dacă este nevoie, doctorul poate prescrie o rețetă pacientului, foarte ușor, folosind aplicația pentru a-i trimite medicamentele necesare si în ce cantitate trebuiesc cumpărate.

Pe lângă faptul că oferă posibilitatea programărilor online, în această aplicație se poate gestiona si stocul de medicamente și materiale indispensabile din spital cu posibilitatea de a reface stocul fără a fi nevoie de un inventar fizic efectuat de cadrele medicale.

Tehnologii folosite

După cum am menționat anterior, MediCare este o aplicație mobilă, deocamdată disponibilă doar pentru utilizatorii de telefoane cu sistem de operare Android.

Pentru dezvoltarea acestei aplicatii am folosit mediul de dezvoltare Android Studio bazată pe software-ul IntelliJ IDEA de la JetBrains.

Ca și limbaje de programare am folosit Java, unul dintre cele mai cunoscute și puternice limbaje de pe piață, dar și limbajul Kotlin. Limbajul Kotlin a apărut pentru prima oară în anul 2011, dezvoltat de cei de la JetBrains și a fost făcut pentru a interpola în totalitate cu limbajul Java.

De ce am ales să folosim și limbajul Kotlin? Deoarece, datorită sintaxei, scrii mai puțin cod decât în alte medii de programare, sintaxa este concisă, mai exact proiectată să poată profita de capabilitățile puternice ale noilor compilatoare. Unul dintre cele mai importante motive pentru care am ales să folosim și Kotlin este error reduction. Compilatorul este capabil de a identifica potențialele erori datorita sistemului build-in de null safety.

Pentru manevrarea datelor, dar și pentru funcționalitatea de login si register am ales să lucram cu Firebase deoarece este un tool ușor de folosit și este integrat in IDE-ul folosit in dezvoltarea aplicatiei. Pentru autentificare am folosit produsul de Authentication (FirebaseAuth) pentru a putea gestiona utilizatorii intr-un mod simplu și sigur. Ca și metodă de autentificare am ales varianta email-parola.

Pentru distribuirea și stocarea datelor am folosit produsul de Cloud Storage (FirebaseFirestore). Cloud Storage-ul este construit astfel încât să putem stoca repede si ușor conținut în baza de date precum fotografii sau videoclipuri. SDK-ul Firebase pentru stocarea in cloud se integrează foarte bine cu Firebase Authhentication, astfel oferind control de acces simplu si intuitiv.

Pe scurt, am ales Firebase deoarece ne ajuta sa devoltam aplicații de înaltă calitate, cu un număr mare de utilizatori.

**Partea de backend**

Autentificarea Firebase oferă servicii de backend, SDK-uri ușor de utilizat și biblioteci UI gata făcute pentru a autentifica utilizatorii in aplicatie. Acceptă autentificarea folosind parole, numere de telefon dar si prin alți furnizori populari de identitate, precum Google, Facebook și Twitter și multe altele.

Partea de backend a aplicației noastre este reprezentată de Firebase. Mai întâi avem partea de autentificare (FirebaseAuth) la care am optat pentru varianta de logare folosind emailul si parola. Datorită Firebase-ului nu a trebuit să ne facem griji despre construirea unui backend pentru a susține procesul de autentificare a utilizatorului.

Cum funcționează? Pentru a vă conecta un utilizator primiți mai întâi datele de autentificare de la utilizator. Aceste acreditări pot fi adresa de e-mail a utilizatorului și parola . Apoi, transmiteți aceste acreditări la SDK-ul de autentificare Firebase. Serviciile Firebase vor verifica acele credențiale și vor returna un răspuns clientului.

Ca și serviciu de stocare avem Cloud Storage din Firebase, un servicu de stocare obiecte puternic, simplu și rentabil. Noi î-l folosim pentru a stoca utilizatorii, precum și programările acestora dar și detalii despre medicamentele disponibile.

Cum funcționează? Cloud Storage stochează fișierele într-un container Google Cloud Storage, făcându-le accesibile atât prin Firebase, cât și prin Google Cloud. Acest lucru vă permite flexibilitatea de a încărca și descărca fișiere de la clienți mobili prin intermediul SDK-urilor Firebase și de a face procesări din partea serverului, cum ar fi filtrarea imaginilor sau transcodarea video folosind platforma Google Cloud.

SDK-urile Firebase pentru Cloud Storage se integrează perfect cu autentificarea Firebase pentru a identifica utilizatorii și oferă un limbaj declarativ de securitate care ne permite să setăm controale de acces pe fișiere individuale sau grupuri de fișiere, astfel încât să putem face fișiere cât de publice sau private dorim.

**Arhitectura**

Arhitectura proiectului nostru este structurată astfel:

* 5 activități

*LoginActivity –* este activitatea cu care se lansează proiectul (asta dacă nu ești deja autentificat). In ea preluăm datele din câmpurile email si parola iar cu FirebaseAuth incerc sa ne autentificam. Daca autentificare nu s-a efectuat cu succes atunci notificăm utilizatorul de cauza erorii, în caz contrar reținem userul curent conectat la aplicație in MainActivity după care o deschidem.

*MainActivity –* contine toate butoanele ce definesc funționalitățile aplicației. Tot aici setăm mesajul de bun venit si în functie de rolul utilizatorului arătăm sau nu anumite opțiuni. De asemenea in acestă activitate avem si metoda care se ocupa cu logout-ul.

*RegisterActivity* – in această activitate userul își creeaza contul. Sunt mai multe campuri ce trbeuiesc completate unele avand validare altele nu. Daca înregistrarea reușește atunci userul este redirecționat la pagina de login pentru a se putea loga, dacă nu userul este informat de motivul din spatele eșuării înregistrării.

*AppointmentActivity –* oferă posibilitatea de a adăuga, vizualiza si anula programări. Această activitate incarcă dinamic fragmente, default incarcă fragmentul de meniu, apoi se adaugă in backstack fragmentul de manage appoinment sau add appointment.

*PrescriptionActivity –* activitatea aceasta se ocupa cu gestionaerea rețetelor și în funcție de alegerea utilizatorului incarcă în mod dinamic fragmente pentru a servi acest scop.

* 7 fragmente

*AppointmentFragment -*  acesta contine butonele corespunzătoare acțiunilor pe care le poate efectua userul asupra porgramărilor

*AddAppointmentFragment* – acesta contine elementele de view necesare creeri unui appointment (doctorul, data, ora)

*DatePickerFragmet* – ne folosește pentru alegerea datei unei programări

*ManageAppointentFragment –* operatiile de CRUD pe appointment

*PrescriptionFragment -*  acesta contine butonele corespunzătoare acțiunilor pe care le poate efectua userul asupra rețetelor

*AddPrescriptionFragment* – acesta contine elementele de view necesare creeri unei rețete

*PrescriptionAppointentFragment –* operatiile de CRUD pe appointment

Actiunile si fragmentele comunica prin intermediul campurilor statice din MainActivity, putand fi modificate de oriunde din aplicatie.

* 9 modele

Am creeat un model generic (AbstractUser) ce contine datele comune (nume, email, parola, rol) intre cele 5 tipuri de utilizatori (asistenta, pacient, doctor, admin, manager spital). Pe langa acestea mai avem modele si pentru Medicament, Programare si Reteta.

Pe langa aceste 9 modele mai avem si 3 enumuri pentru rolurile aplicatiei, statusul programarii si tipul de doctor.

**Utilizare Aplicatie**

Aplicatia are 5 tipuri de utilizatori, fiecare cu drepturi diferite. Rolurile sunt: Pacient, Doctor, Asistenta, Manager Spital si Admin.

1. *Pacient*

Un cont cu rolul de pacient este creeat de fiecare data cand cineva se inregistreaza in aplicatia noastra. Ca si pacient poti seta programari dar si sa le modifici sau sa le stergi in cazul in care apare un incident neplacut. De asemenea poti vedea si prescriptiile date de medic.

1. *Doctor*

Un doctor poate scrie reteta unui pacient dupa consultatie. Ii prescrie ce medicamente trebuie sa ia si la ce ora din zi.

1. *Asistenta*

Asistenta este responsabila de necesarul de medicamente al spitalului.

1. *Manager Spital*

Managerul de spital are acces la toti utilizatorii aplicatiei. El poate creea de asemenea conturi cu rol de doctor sau asistenta insa doar atat.

1. *Admin*

Adminul poate gestiona toti userii aplicatiei. Doar acesta poate creea un user ce are rolul diferit de pacient. El poate adauga, sterge sau modifica orice cont al aplicatiei.

Datorita numarului de roluri destul de divers aceasta aplicatie poate fi folosita pentru gestiunea programarilor si medicamentelor din spital, de asemnea si a scrierii de retete.

**CONCLUZII**

In concluzie, MediCare este o aplicatie medicala, de centralizare si gestionare a diferitor informatii din domeniul medical. Cu ajutorul acesteia putem micsora interactiunea inter-umana ceea ce este si recomandat in aceasta periaoda de criza.

Cu o arhitectura bogata, aceasta dispune de multiple functionalitati metionate in pasii anteriori.

Fiind o aplicatie de mobil este disponibila publicului larg pentru o mai buna organizare a timpului in ceea ce priveste sanatatea noastra si celor din jur.

Este o aplicatie moderna, cu un design simplist lucru ce va ajuta buna manevrare a acestei aplicatii oferind o experienta placuta utilizatorilor.

Aplicatia dispune de un numar ridicat de roluri ceea ce ofera o vastă arie de functionalitati permitand atat personalului spitalului cat si pacientilor sa se simta mai confortabil cand vine vorba de domeniul medical, sa nu i-l mai percepa ca pe ceva de temut.

Speram ca aplicatia noastra va fi de folos in aceste timpuri de criza si va fluidiza coridoarele aglomerate ale spitalelor.