**Documentatie proiect**



**Disciplina:** Programare pe dispositive mobile

**Componenta echipei:** Crupa Gabriel

Hurloi Iulia

**Numele proiectului:** TextRecognitionApp

**Data Trimiterii:** 14/05/2020

**Destinatar:** Lector Dr. Liviu-Octavian Mafteiu-Scai

**Rezumat**

Proiectul consta intr-o aplicatie, realizata in Android Studio, destinata utilizarii pe dispositive mobile, care identifica textul in imaginile capturate sau selectate de catre utilizator. Aceasta implementeaza kitul de machine learning de la Google pentru text\_recognition, legatura cu acesta s-a realizat cu ajutorul site-ului Firebase.

**Scopul**

Utilitatea principala a aplicatiei este de a prelua textul dintr-o poza, afisandu-l apoi pe ecran. Utilizatorii potentiali sunt persoanele care cauta solutii mai rapide si mai simple pentru a extrage textul dintr-o imagine, un grup tinta ar putea fi persoanele in varsta sau persoanele care in general nu mai au o vedere buna si sunt pusi in dificultate in momentul in care trebuie sa citeasca texte scrise in fonturi foarte mici, astfel aplicatia le-ar usura acest aspect al vietii, textul fiind redat in fonturi mari in cadrul aplicatiei.

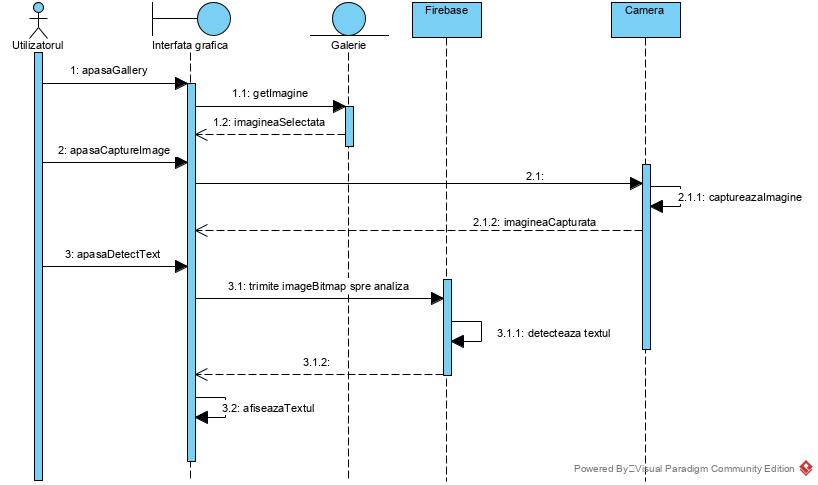
**Introducere**

Problema principala indentificata, care a dus la implementarea unei astfel de aplicatii, este faptul ca desi sunt deja pe piata mai multe varinante de aplicatii cu aceste functii, acestea nu presteaza la nivelul dorit de societatea actuala, care este tot mai grabita si doreste eficienta maxima.

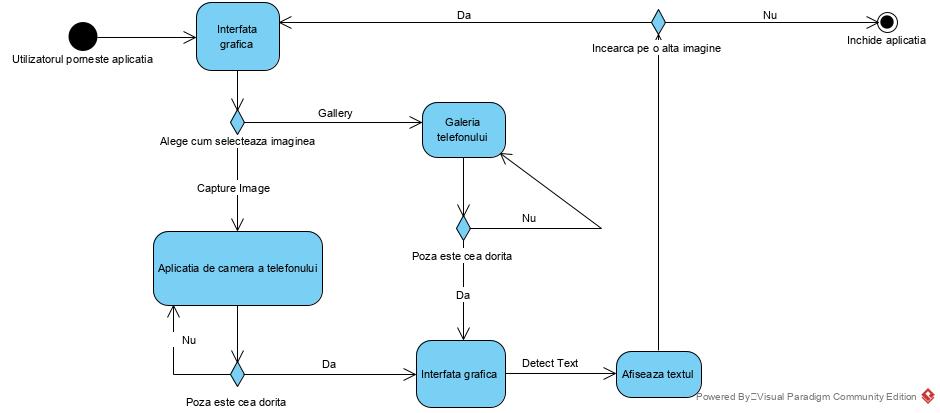
O astfel de aplicatie, care isi propune recunoasterea textului din imagini, este Text Scanner [OCR], valabila in GooglePlay. Aceasta este o aplicatie care transforma pozele in text, poate accesa galeria telefonului si sa selecteze de acolo imaginile sau le poate captura. Printre cel mai bine cotate aplicatii de acest tip, aceasta suporta pana la 50 de limbi diferite si de asemenea poate recunoaste scrisul de mana.

Text Scan Pro este de asemenea o aplicatie care are in linii mari aceleasi functionalitati ca si prima prezentata, insa, in plus, vine cu posibilitatea de a transforma textul identificat intr-un document pdf.

**Functionalitate**

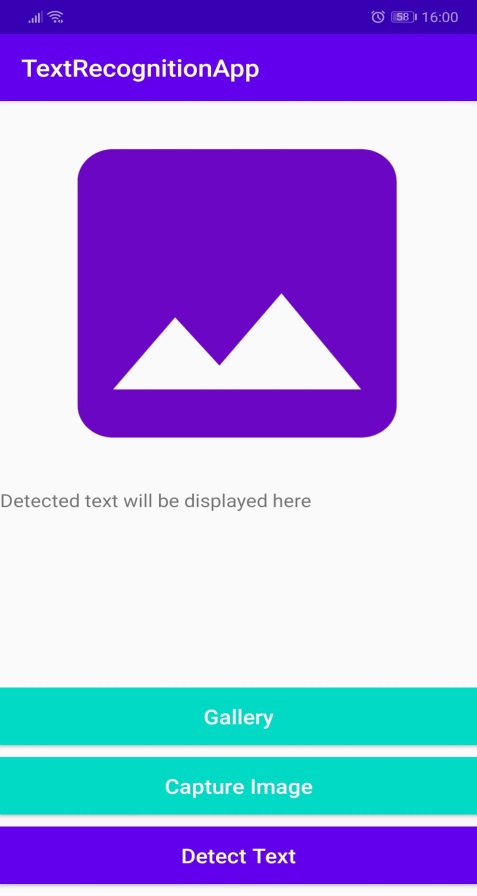


**Interfata Utilizator**



**Rularea aplicatiei**

Interfata utilizator este una simpla si usor de utilizat de orice categorie de persoane. In momentul in care deschidem aplicatia observam ca are 3 butoane,, denumite sugestiv pentru a facilita modul de interactiune a utilizatorului cu aplicatia, acestea sunt: Gallery, Capture Image si Detect Text.



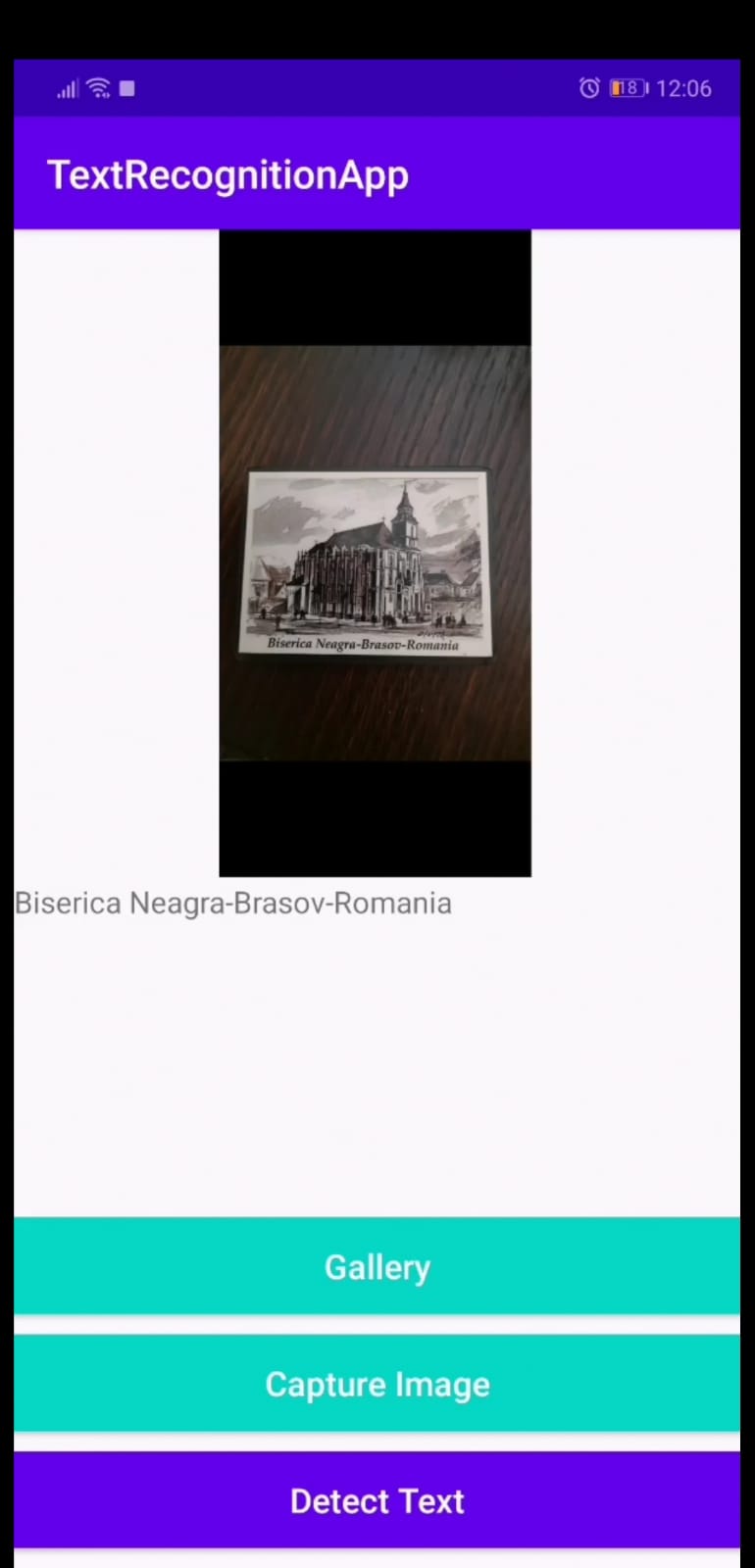
Primul buton face legatura dintre aplicatie si galeria personala a utilizatorului, de unde acesta poate alege o imagine ce contine textul pe care acesta doreste sa il extraga, la selectarea imaginii primeste doua optiuni, sa pastreze imaginea si sa o trimita spre analiza sau sa renunte la acea imagine si sa selecteze alta.

In cazul in care utilizatorul nu dispune de o imagine pre-capturata cu textul pe care doreste sa il extraga, acesta are la dispozitie optiunea de a accesa camera dispozitivului si a captura o imagine pe loc, prin al doilea buton, Capture Image.

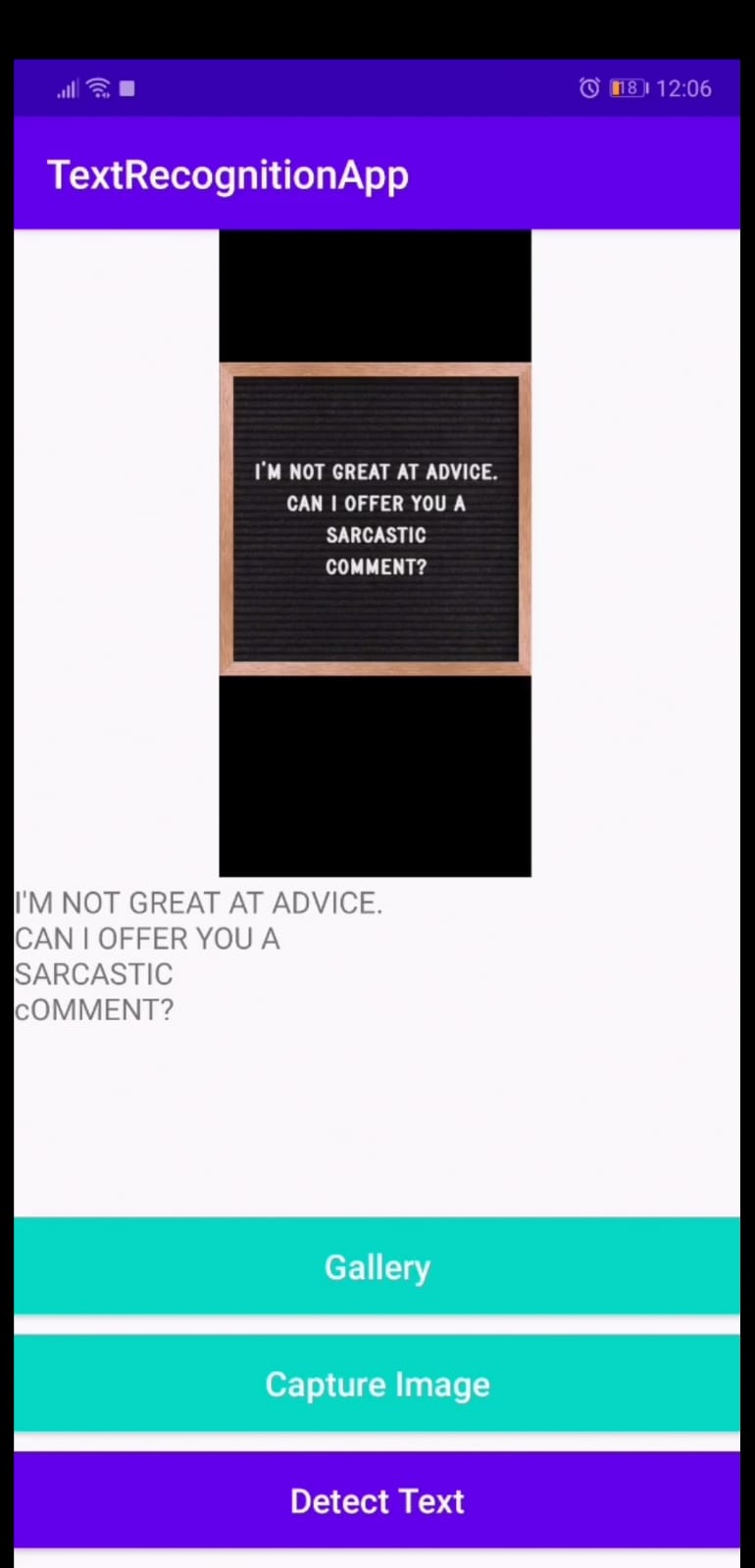
Butonul va accesa camera dispozitivului, iar dupa caturarea imaginii avem o fereastra cu optiunea de a pastra spre analiza imaginea sau a incerca inca odata capturarea.

Dupa ce alegerea imaginii, printr-o metoda din cele doua enumerate, a fost efectuata, utilizatorul va apasa ultimul buton, Detect Text, care va actiona functia de recunoastere a textului iar dupa ce acesta va fi extras va fi afisat in zona de text view situata imediat sub imaginea selectata, care ramane pe ecran pentru referinta.

Test cu poza facuta din camera foto:



Test cu poza din galerie:



**Structura programului**

Programul este scris in aplicatia Andoid Studio, in limbajul de programare Java.

Activitu\_main.xml si MainActivity.java sunt fisierele importante ale proiectului, primul contine codul din spatele interfetei grafice iar al doilea contine cele sase metode principale care conduc la functionarea codului.

Metoda *onCreate()* face legaturile dintre butoane si metodele ce trebuiesc chemate la pasarea acestora.

Metoda *pickImageFromGallery()* descrie procesul de selectare a unei imagini din galleria telefonului, o operatie usoara, se salveaza calea spre imagine intr-o variabila de tip intent.

Metoda *dispatchTakePictureIntent(),* acceseaza camera dispozitivului pentru a captura o imagine ce urmeaza sa fie analizata.

Cea mai importanta metoda (din punctul meu de vedere) este *onActivityResult(),* care trimite imaginea intr-un bitmap pentru a nu mai incarca memoria sa o salveze si astfel ea sa ajunga sa fie analizata.Aceasta functie proceseaza de asemenea calea primita de la functia pickImageFromGallery() si face legatura intre aceasta si imageBitmap.

Metoda *detectTextFromImage().* In aceasta metoda sunt folosite variabile si functii specific platformei Firebase, de unde a fost importat kitul de machine learning. Metoda doar proceseaza imagine si salveaza textul.

Metoda *displayTextFromImage(),* dupa cum ii spune si numele preia textile determinat mai sus si il afiseaza in zona de textview.

**Imbunatatiri si extensii**

Pentru viitor ne propunem sa aducem cateva imbunatatiri aplicatiei, precum:

-optiunea de salvare a textului extras intr-un fisier la alegerea utilizatorului (.pdf, .docx);

-imbunatatirea certitudinii cu care se recunoaste textul;

-capacitatea de a manipula imaginea si de a o rotii, astfel incat textul sa fie usor de extras, fara erori;

-capacitatea de a recunoaste caracterele speciale ale limbii romana;

**Concluzii**

In concluzie, TextRecognitionApp, este o aplicatie utilia deoarece usureaza intr-o anumita masura munca robotica pe care o foloseau oamenii inainte pentru a-si extrage textul dintr-o imagine.

**Referinte**

\*Android Take Photos with camera Link:

[https://developer.android.com/training/camera/photobasics#java](https://developer.android.com/training/camera/photobasics" \l "java)

\*Firebase Text Recognition Link:

<https://firebase.google.com/docs/ml-kit/android/recognize-text>

\*Firebase Dependencies Link:

<https://firebase.google.com/support/release-notes/android>

**Anexe**

Anexat se va gasi fisierul care contine MainActivity.java, alte clase nu au fost necesare pentru aceasta aplicatie.