App MIRPR

**System requirements**

Pentru vizualizarea modelului in Scene Viewer, dispozitivul cu android trebuie sa aiba suport pentru  *[ARCore](https://developers.google.com/ar/discover/supported-devices)* si *Android 7.0 Nougat (API Level 24)* sau mai recent.

**Interfata:**

In partea de sus a ecranului apar instructiunile de folosire (cum trebuie facute fotografiile).

La inceput este afisat “Take a left profile picture (-50°)”, prin care se sugereaza ca imaginea ce urmeaza a fi capturata sa fie una din profil stanga (la un unghi de -50° fata de planul frontal).

Dupa ce prima poza este capturata, utilizatorul va reveni in aplicatie si va vedea fotografia si mesajul “Take a frontal picture” prin care se sugereaza ca imaginea ce urmeaza a fi capturata sa fie una frontala

Dupa ce a doua poza este capturata, utilizatorul va reveni in aplicatie si va vedea fotografia si mesajul “Take a right profile picture (50°)”, prin care se sugereaza ca imaginea ce urmeaza a fi capturata sa fie una din profil dreapta (la un unghi de +50° fata de planul frontal).

Dupa ce fiecare imagine este capturata, aplicatia afiseaza un toast message prin care utilizatorul este informat cu privire la numarul de imagini pe care mai trebuie sa le captureze.

Dupa ce si cea de-a treia poza a fost capturata, acestea se trimit catre server si apare un loading indicator, alaturi de mesajul “The mesh is being generated…”.

Cand meshul se termina de generat, apare un toast message care confirma acest fapt, si sugereaza vizualizarea meshului in Scene Viewer.

Interfata cu utilizatorul dispune de 2 butoane:

**Take photo** - deschide camera si salveaza temporar imaginea capturata

**Show 3D** – afiseaza meshul generat in Scene Viewer

**Arhitectura si functionalitate**

Aplicatia Android **3DReconstruction** reprezinta *clientul* in cadrul acestui proiect.

Aplicatia este compusa dintr-o activitate cu un fragment principal. Butoanele, mesajele informative si fotografiile sunt parte a acestui fragment.

La actionarea butonului Take photo, se lanseaza un ActivityResultCaller, avand ca si parametru call contractul TakePicture, care deschide camera default a dispozitivului, iar dupa ce se captureaza o imagine, asteapta confirmarea ca este in regula, dupa care trimite inapoi in aplicatie un URI corespunzator locatiei temporare unde este salvata.

In continuare, imaginea este convertita intr-un byte array encoded in Base64, sub forma de String, care este salvat intr-un container.

Odata ce in container sunt stocate 3 imagini, acestea sunt trimise catre server, iar utilizatorul vede un loading indicator.

Clientul HTTP este de tipul OkHttpClient, iar requestul se face printr-un post, avand ca body un form cu cele 3 imagini.

Deoarece Google AR Core are nevoie de obiecte in format GLTF stocate pe un server, raspunsul serverului la requestul de mai sus este doar o confirmare ca meshul a fost generat cu succes, la primirea caruia loading indicatorul dispare, iar utilizatorului i se sugereaza sa vizualizeze obiectul generat prin apasarea butonului Show 3D.

La apasarea butonului Show 3D, aplicatia face un apel la Scene Viewer, caruia ii furnizeaza URL-ul unde se gaseste modelul generat pe server, dupa care deschide aplicatia AR Viewer, unde incarca meshul ce poate fi vizualizat.

Server

Pe partea de server sunt disponibile 3 endpoints

/process

Primeste prin POST cele 3 imagini sub forma de Base64 encoded byte array String, prin parametri image1, image2 si image3.

Imaginile sunt trimise spre prelucrare si se raspunde cu OK daca modelul 3D a fost generat cu success.

/model

Acest endpoint este apelat de catre Scene Viewer si returneaza un fisier GLTF generat pe baza meshului rezultat din procesarea imaginilor.

/display\_mesh.bin

Deoarece fisierul GLTF generat de noi produce separat un bin file, parte din GLTF, Scene Viewer va face un request spre server pentru obtinerea acestuia.