

Tomanano

Egyetemi projekt

Vezető tanár: Szántó Zoltán

Diákok: Toducz Endre, Majla Nándor, Nagy Márton Hunor, Kedves Norbert

Kezdés ideje: 2020.10.25
Befejezés dátuma: 2020.12.10

Tartalomjegyzék

1. Bevezető

2. Célok

3. Software követelmények

- Felhasználói követelmények
- Rendszerkövetelmények
 - 1. Funckcionális
 - 2. Nem funkcionális

4. Tervezés

Architektúra

Alkalmazás áttekintése

5. Modulok leírása

Adatbázis

Android

6. Kivitelezés

Verziókövetés

Tervezési elvek

7. Használati útmutató

8. Összegzés

1. Bevezető

Manapság az emberek egyre ritkábban találkoznak egymással, mivel létre jöttek alternatív lehetőségek a kapcsolattartásra. A személyes találkozást felváltották az online világban történő találkozások, ismerkedések. Az emberek egyre zárkózottabbak. Hiába van több száz ismerősük a közösségi oldalakon, mégis magányosak. Az online térben való rekedés pedig egyre csak elszigeteli az embereket egymástól úgy, hogy mindenki a közösségi oldalakról való visszajelzés alapján azt hiszi, rengeteg barátja van, de az offline világban ez nem így van. Egyre nehezebben fejezik ki magukat az emberek a valós beszélgetésekben, ez frusztrálja őket és már átszoktak a chatelésre, ami persze azt eredményezi, hogy még inkább kevesebb időt töltenek egymással.

Mi megpróbálkoztunk egy olyan applikáció fejlesztésével Android eszközre, amely elősegíti a közös interakció létrejöttét. Az applikáció arra bátorít, hogy egyre több emberrel készíts közös szelfit. A szelfi készítés közben az ember interakcióba lép a környezetével. Ez a találkozás akár tovább is alakulhat, fejlődhet. Legvégül pedig barátsággá alakulhat.

Az applikáció ezt burkoltan éri el, mivel egy versenyre épül. A célja, hogy minél több személlyel készíts fényképet, megelőzve a társaidat, akikkel együtt kezdtetek neki a versenynek.

2. Célok

A mi célunk egy olyan alkalmazás létrehozása volt, ami valamilyen szinten képes összehozni az embereket, új kapcsolatokat létrehozni. Egy versenyre épülő alkalmazást terveztünk meg, ahol az a cél, hogy minél több emberrel kell fényképet készíteni. A végén, akinek a legtöbb emberrel sikerült képet csinálni, az nyer. Szerettünk volna egy felhasználóbarát felületet létrehozni, ami ellátja a kitűzött feladatait is, így az alkalmazás elég minimalista és visszafogott designt kapott.

3. Software követelmények

A. Felhasználói követelmények

A felhasználók szempontjából igyekeztünk minél felhasználóbarátabb alkalmazást létrehozni, ami magába foglalja, hogy könnyen kezelhető legyen még a kezdő felhasználók számára is. Emellett átlátható és vonzó designja legyen.

B. Rendszerkövetelmények

-Funkcionális : Az alkalmazás futásához mindenképp szükség van egy androidos mobil eszközre és internet kapcsolatra, valamint kamerára.

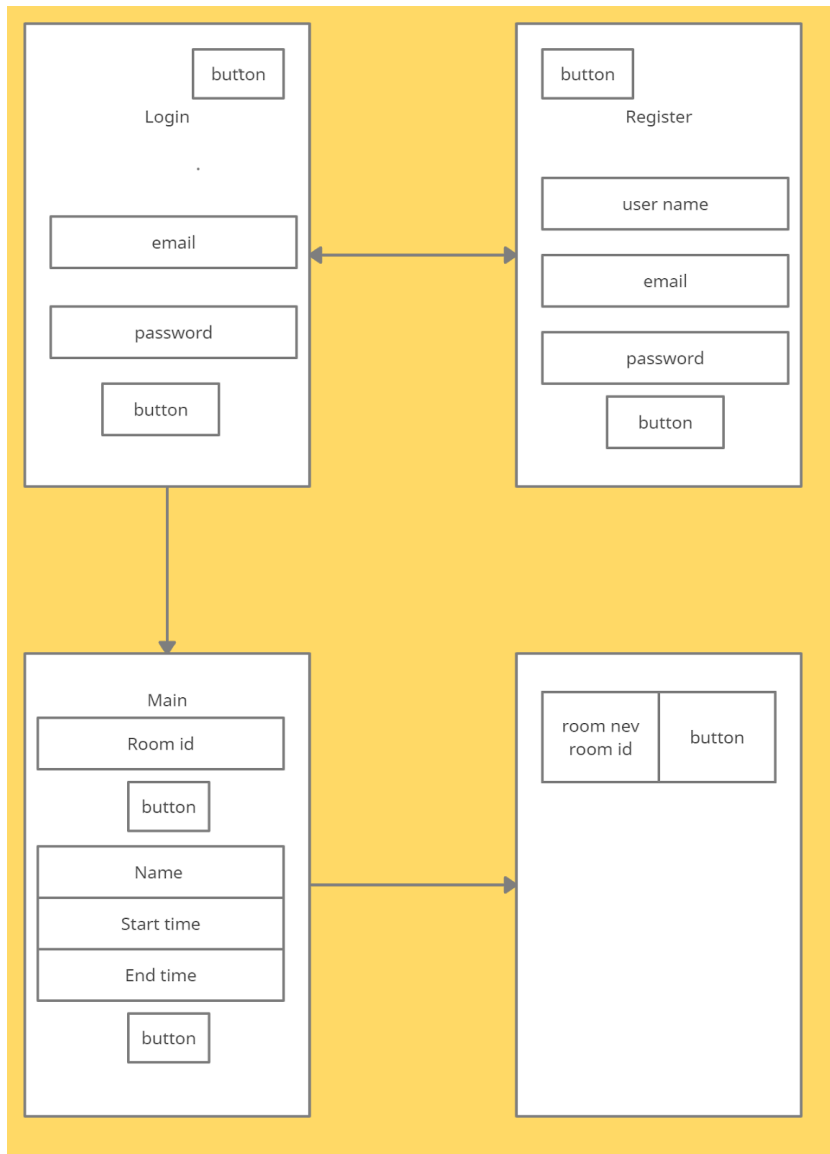
-Nem funkcionális : Az alkalmazás futtatásához nem funkcionális nagy teljesítményű hardware ,ujjlenyomat olvasó ,gps.

4. Tervezés

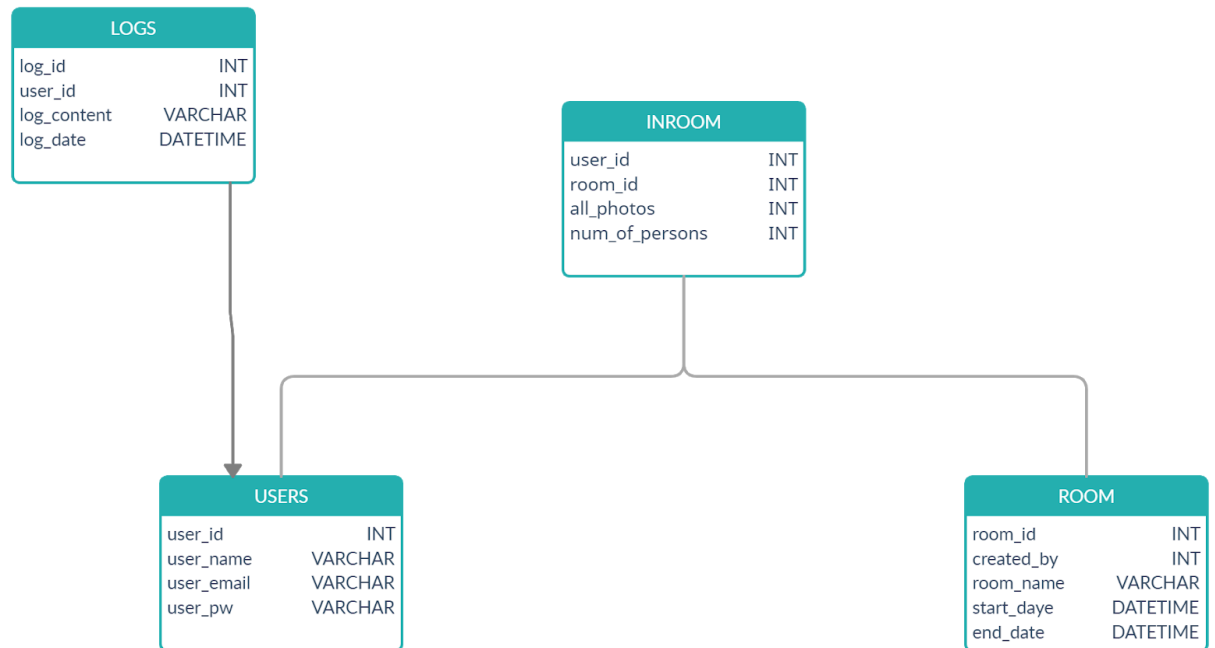
A. Architektúra

A projekt nagy része egy android applikációból épül fel, ami közvetlenül kommunikál egy API-val. Az API az adatokat egy MySQL adatbázisban tárolja, ahonnan szükség szerűen lekéri őket és beszúrja az új rekordokat.

B. Alkalmazás áttekintése



5. Modulok leírása



1. Adatbázis

Az alábbi MySQL adatbázis felhasználókat és szobákat (room) tárol. Minden felhasználó (avagy user) létrehozhat szobákat, amelybe a többi user csatlakozhat. Minden szoba egy versenyként fogható fel, ahol a be csatlakozott felhasználók fényképeket készítenek különböző emberekkel. A fényképeken található emberek számából felállítunk egy ranglistát, amelyet a verseny végén közlünk a versenyzőkkel.

Az adatbázis szerkezete a következő:

A users táblában a felhasználók személyes adatait találjuk. Ez azt jelenti, hogy eltárolásra kerül a regisztráláskor megadott neve, email-címe és a jelszava. A jelszavak titkosítva, SHA512-es kódolást használva vannak elmentve. Minden user kap egy egyedi azonosítót, valamint egy email-címmel csak egyszer lehet regisztrálni.

A room tábla a szobákkal kapcsolatos adatokat tartalmazza. Vagyis eltárolásra kerül a szoba neve, a szobát létrehozó felhasználó, valamint az érvényességi ideje, amely két dátum: az első az indulás dátuma, a második pedig a bezárulás dátuma. A szoba érvényessége idejének lejártá után a szoba inaktívvá válik, ami azt jelenti, hogy utána már nem csatlakozhat be senki oda.

Az inroom táblában megtaláljuk, hogy melyik user melyik szobába csatlakozott, továbbá az összes elkészített fényképek számát és a fényképeken található emberek számát, amelyet a verseny folyamán sikerült készítenie.

A fontosabb eseményeket egy logs nevű táblában naplózzuk.

The image displays two wireframe designs for a login/register interface, both featuring a blue header with the text "To_Ma_Na_No".

Left Design:

- Header: "To_Ma_Na_No"
- Buttons: "Login" (blue) and "REGISTER" (purple)
- Input Fields: "Email" (with user icon) and "Password" (with lock icon)
- Bottom Button: "LOGIN" (purple)

Right Design:

- Header: "To_Ma_Na_No"
- Buttons: "LOGIN" (blue) and "Register" (blue)
- Input Fields: "Name" (with user icon), "email" (with user icon), and "Password" (with lock icon)
- Bottom Button: "REGISTER" (purple)

2.Android

UI:

Az android UI az aplikacio vizualis felületétért felelős rész. Fontos hogy ez a rész eleget tegyen mind a felhasználói követelményeknek mind a rendszer követelményeknek. Egy nem intuitív UI nehezzé teszi az aplikáció felhasználását, ezért hihetetlenül fontos ennek sok figyelmet adni. Ha belegondolunk, az UI az, amivel a felhasználó interakcióba lép, az az amit először meglát, ezért fontos jól megtervezni, és jól megvalósítani. A különböző gombok és egyéb Android komponensek jól átlátható pozícióban kell, hogy legyenek, illetve a céljuk és használatuk egyértelmű kell legyen.

Retrofit:

Persze az aplikáció nem működhet egy webes kapcsolat nélkül.

Igen fontos ezt a kapcsolatot egy gyors, mégis biztonságos library-vel megvalósítani.

Csakis erre a célra volt kitalálva a Retrofit.

Szerintem egy hatékony és könnyen használható kódkészletről van szó,

Ami tökéletes a mi célunkra.

Feltéve egy szerver létezését,

A Retrofit foglalkozik a kapcsolat létrehozásával,

S a web request-ek küldésével.

Zéró kérdés, ez az ideális választás.

Sikeresen megírtuk a Retrofit-et felhasználó kódot.

Zavartalanul működött.

Ami lehetővé tette az adatok küldését a mobil és az API között.

Remekül tudtuk ezt használni.

Képfeldolgozás megoldása:

A képfeldolgozást Firebase ML Kit segítségével van megvalósítva.

A Camera osztály felhívja a Camera Api, amely segítségével az androidos eszköz a beépített kamerával készített képet elmenti a háttértárra. Ezt az elkészített képet elküldjük a firebase szerverre, ahol megtörténik az arcfelismerés és elemzés. Egy képen az összes arcot felismeri. Majd minden arcról részletes elemzést küld vissza. A mi applikációnk jelen esetben, csak az arcok számát kéri le ebből az adathalmazból. ű

6. Kivitelezés

-Verziókövetés

A verzió-követésre egy Git repository-t használtunk, amit GitHub-on tároltunk.

-Tervezési elvek:

Közösen megterveztük az adatbázist, utána felosztottuk magunk között a feladatokat az alábbi módon:

Kedves Norbert: az adatbázis megvalósítása, az API megírása, ami adatokat küld és fogad az android applikációtól

Nagy Márton Hunor: az android applikáció UI részének megtervezése és megvalósítása

Majla Nandor: az android applikáció megírása, ami kommunikál az API-val

Toducz Endre: az arcfelismerés megvalósítása, amely egy képen található összes személy arcát felismeri és annak számát visszatéríti

According to all known laws of aviation, there is no way a bee should be able to fly. Its wings are too small to get its fat little body off the ground. The bee, of course, flies anyway because bees don't care what humans think is impossible.

7. Használati útmutató

Lépések:

1. Belépés az applikációba
2. Regisztrálás és/vagy bejelentkezés
3. Csatlakozás egy szobához, vagy új szoba létrehozása
4. Képek feltöltése egy szobába
5. Fénykép elkészítése, majd annak jóváhagyása

8. Összegzés

A projekt megírása számunkra nagyon hasznos volt, rengeteg mindent tanultunk, megtanultunk még inkább csapatban dolgozni, megismertük a github nyújtotta lehetőségeket.