

Arcfelismerés alapú beléptetőrendszer

csapatnév¹

Suszter Ágnes Eleonóra, Robotka Adrián

Felkészítő tanár: Csató Endre

Földes Ferenc Gimnázium, 3525 Miskolc, Kelemen Didák utca 5

1. Bevezetés

Pályamunkák egy arcfelismerésen alapuló beléptető rendszer megvalósítását tárgyalja. Az általunk megoldani kívánt problémát a Microsoft Imagine Learn to Win Contest nevű országos versenyen kaptuk.

Elsődleges célunk, hogy bizonyos rendezvények (pl. szakmai konferenciák) vendégeinek beléptetését egyszerűbbé és gyorsabbá tegyük. A beléptetés előfeltétele a regisztráció, amit egy mobil alkalmazás biztosít. Ehhez néhány személyes adat megadása és több fotó szükséges a regisztrálandó személyről. Az adatokat egy web API tárolja, ami hozzáférést is nyújt a külső arcfelismerő és azonosító szolgáltatáshoz. A beléptetéseket egy kliens alkalmazás végzi a helyszínen, amely egy kamera élőképet használja adatforrásként a felismeréshez. Megoldásunk készítése során igyekeztünk minél jobban lefedni azokat a problémákat, amik felmerülhetnek egy ilyen környezetben.

2. Probléma megoldásának menete

2.1. Fejlesztés menete

Amikor neki láttunk megoldani a problémát, akkor még nem volt elképzelésünk rendszerünk várható működéséről, így proof of concept² módszerrel igyekeztünk kialakítani egy működési modellt.

Lépésről lépésre tudtuk meg mi is a járható út. Ilyen volt az élő videófolyam feldolgozása és az abból történő arcfelismerés, továbbá az ezen rendszer köré épített eseménykezelő rendszer, ami a megfelelő visszajelzéseket és az alkalmazást irányítja. Újabb és újabb fejlesztési ágakat nyitottunk Git verziókezelő rendszerünkben. Volt hogy 1-2 vakvágánynak bizonyult, de ez a módszer több sikeres funkcionalitással gazdagította a végeredményt.

Fejlesztés közben igyekeztünk minél több esetvizsgálatot tenni, hogy „bomba biztos” legyen a munkánk. Így került bele a több képernyős mód is. Egy képernyő a beléptető asszisztensnek, egy másik a vendégnek. Erre azért volt szükség, hogy csak releváns információkat jelenítsünk meg a feleknek.

¹ A csapatnevünk tényleg a következő: csapatnév

² Kísérleti projekt; ötletek kipróbálása, hogy működőképesekek-e

A programjaink további fejlesztéséhez agilis módszertant választottunk. Ezzel folytatni kívántuk az előbbieken említett elképzeléseinket, ugyanakkor produktívabbá tettük a fejlesztést.

Mivel objektum-orientált nyelveket használunk, így megfelelő módon tudtuk implementálni az architektúrális, szerkezeti és létrehozási tervezési mintákat, amelyek segítségünkre voltak a megvalósításban.

2.2. Fejlesztés közben felmerült problémák

Az arcfelismerés és azonosítás alapkövei munkánknak. Fontos volt a minél megbízhatóbb azonosítás biztosítása. Ehhez a Microsoft Cognitive Services Face API-t választottuk. Ez egy pénzdíjas, de kiváló szolgáltatás. Mi az ingyenesen igénybe vehető API lekérdezések keretei közé szorultunk a továbbfejlesztés alatt.

Az első tesztelések alkalmával szembesültünk a webkameránk szűk alkalmazhatóságával, ezért megkerestük a Canon európai terjesztését, hogy részt vehessünk a Canon developer programjában. Mivel sikeres elbírálást kaptunk így lehetőségünk nyílt a csapat tulajdonában lévő 700D típusú DSLR fényképezőt webkameraként használni, de ehhez szükség volt többletkódolásra és a Cannon EOS SDK-ra.

3. Elért eredmények

A Java nyelvet és az Android platformot választottuk mobilalkalmazásunkhoz. Ezzel a többnyelvű alkalmazással lehet jelentkezni az eseményre. A tájékoztató képernyő után a regisztrációs képernyővel (lásd: 1. ábra) találkozik felhasználó, ha rendelkezik internet kapcsolattal. Az alkalmazás álló és fekvő orientációra is fel van készítve.

1. ábra: Regisztrációs felület

Figyeltünk a felület felhasználó barát kialakítására, így a jól megszokott vezérlő elemeket használtuk. Ahol tudtunk ott a még kényelmesebb formáját valósítottuk meg az adatbevitelnek, például a dátumválasztáshoz dedikált ablak jelenik meg. Adatrögzítés előtt ellenőrizzük, hogy az adatok megfelelnek-e a formai követelményeknek. Ennek része az egyedi ímélcím megszorításnak való megfelelés vizsgálata is. Az alkalmazásban 3 db képet kell feltölteni a személyről (ez az API-n beállítható), amelyeken nem lehet több arc. Ezt ellenőrizzük is. A fotók kiválasztására a kamera és a képgaléria áll rendelkezésre. Az alkalmazás előző bemutatása óta lecseréltük a kezdő és sikeres regisztráció képernyőt, valamint segítséggel láttuk el a képfeltöltést.

API URL-ek		
HTTP metódus	URL path része	Leírás
POST	/detect/	Arc(ok) felismerése
POST	/identify/	Arc(ok) azonosítása
POST	/participant/	Résztevő létrehozása
GET	/participant/{participant_id}	Adatok lekérdezése
POST	/participant/{participant_id}/image	Kép feltöltése
GET	/participant/checkEmail/{email}	Email ellenőrzése

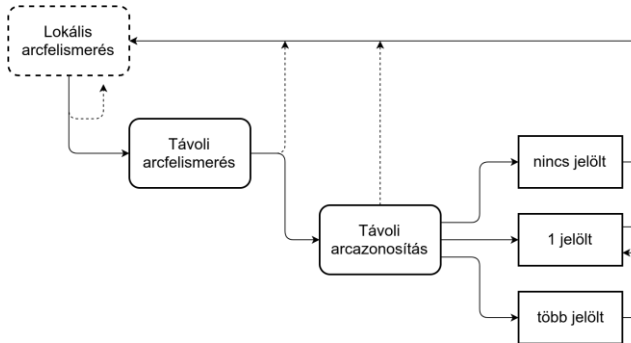
1. táblázat: API URL-ek

Az adatok tárolásához PHP nyelven készítettünk RESTFUL API-t³. Az API funkcionalitásáról áttekintést nyújt az 1. táblázat. Ehhez Laravel keretrendszert használtunk és ORM⁴ rendszeren keresztül MySQL adatbázis kezelővel üzemeltetjük. Ügyeltünk a biztonságra: kötelező az API kulcs továbbítása, figyelmet fordítottunk az XSS, sql injection, overflow jelenségek kiküszöbölésére. A keretrendszernek köszönhetően támogatja a gyorsítótárazást és könnyen skálázható az API. Szabadon választható az adatbázis kezelő rendszer és a helyi fájlkezelésen felül felhő alapú tárolást is be lehet állítani. Mivel az arcfelismerést egy külső API valósítja meg, tervezési, üzemeltetési és biztonsági szempontok alapján úgy gondoltuk, hogy az alkalmazás csak a mi API-nkon keresztül érheti el az arcfelismerés szolgáltatást. Így a mi API-nk adja hozzá az azonosítási kulcsot a lekérdezéshez. Az előző bemutatás óta weboldalként is funkcionál az API. Bejelentkezhetnek

³ REST: REpresentational State Transfer; kihasználjuk a get, post, put, delete HTTP metódusok lehetőségeit.

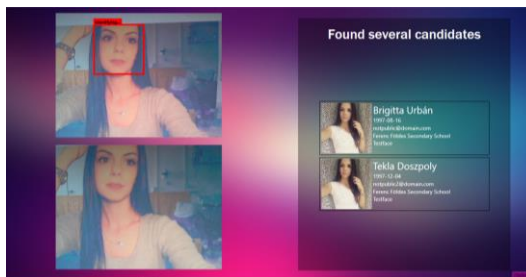
⁴ ORM: Object-releational mapping; az adatbázishoz objektum szinten férünk hozzá absztrakciókon keresztül

a szervezők és kezelhetik a regisztráltakat valamint a beléptető személyzetet, akiknek be kell lépniük ezen túl a beléptető kapuknál az alkalmazásba. A résztvevőket osztályokba csoportosíthatják (pl: VIP, normál, stb.) Ezen túl élő belépési képernyő is elérhető az oldalon.



2. ábra: Felismerési folyamat főbb esemény forrásai

C# nyelven írt kliens alkalmazásunk inicializációs ablakkal indul. Ebben ki kell választani a kamerát, be kell állítani, hogy működjön-e a lokális arcfelismerő, ami előfeltétele a távoli arcfelismerésnek, és hogy megjelenjen-e a látogatói ablak. A 2. ábrán az arcfelismerési folyamat főbb állomásai láthatóak. Itt az események vízésés modell szerint érhetőek el. A kezelhetőség javítása érdekében ezt a ciklust újraindulása előtt különböző várakoztatásokkal láttuk el. Elképzelésünk szerint egy pontos egyezés után kevesebb ideig kell a visszajelzést a képernyőn tartani, kisebb ideig tartva fel ezzel a következő azonosítást, mint amikor egy ikerpárból ki kell választani, hogy pontosan melyik lépett be (lásd: 3. ábra). Az előző bemutatás óta bejelentkezési képernyővel bővült az alkalmazás. Megjeleníti a résztvevők csoport tagságát, jelentést küld amikor több lehetséges személy közül választották ki a beléptetettet. Néhol a belső működésen is javításokat eszközöltünk.



3. ábra: Több azonosítási találat