

Semestrální práce KIV/NET

Xamarin aplikace pro hudebníky a hudební skupiny

Daniel Stuš
stusd@students.zcu.cz
18. 2. 2021

Souhlasím s vystavením této semestrální práce na stránkách katedry informatiky a výpočetní techniky a jejímu využití pro prezentaci pracoviště.

Daniel Stuš

Obsah

1	Zadání.....	3
1.1	Popis a cíl projektu	3
1.2	Požadavky.....	3
2	Analýza	4
3	Implementace.....	5
3.1	Rozdělení aplikace	5
3.1.1	Dělení součástí hlavního projektu.....	5
3.2	Použité technologie	6
3.2.1	Xamarin.Forms.....	6
3.2.2	SQLite	6
3.2.3	Firebase.....	8
3.2.4	Google Maps Platform.....	8
4	Uživatelská příručka	8
4.1	Předpoklady a instalace	8
4.2	Přihlášení.....	9
4.2.1	Zapomenuté heslo	9
4.3	Registrace	10
4.4	Hlavní navigace	10
4.5	Domů/Katalog	12
4.6	Profil.....	13
4.6.1	Dovednosti	14
4.7	Detail skupiny	15
4.8	Moje skupiny/Pozvánky	16
4.9	Objevujte	17
4.10	Nastavení	18
4.11	O aplikaci.....	19
4.12	Odhlásit.....	19
5	Závěr	20
6	Reference.....	21
A	Programátorský deník.....	0

1 Zadání

Pomocí technologií .NET vytvořte mobilní aplikaci na evidenci a správu a vyhledávání hudebníků a kapel.

Umožněte registraci jednotlivcům nebo celým kapelám, provázání těchto hudebníků a hudebních skupin a taktéž je v aplikaci rozumným způsobem zobrazte. Každý z hudebníků i z kapel by měl mít možnost přidat o sobě podrobné informace a zobrazit ostatním uživatelům v aplikaci, zda hledá kapelu, či v případě skupin, zda hledají muzikanta a na které pozici.

V Aplikaci umožněte zobrazit a vyhledávat tyto uživatele a skupiny v nejbližším okolí (např. podle GPS) a podle dalších kritérií.

1.1 Popis a cíl projektu

Cílem práce je naprogramovat funkční a robustní mobilní aplikaci, představující portál pro hudebníky a hudební skupiny. Tento portál by měl umožňovat registraci a správu jednotlivce(hudebníka), či celé hudební skupiny a poskytovat registrovaným uživatelům způsob, jak snáze vyhledat ty hudebníky a hudební seskupení ve svém okolí, které hledají nové spoluhráče. Tímto způsobem by pak měla aplikace napomoci růstu a zakládání nových úspěšných kapel.

1.2 Požadavky

Obecné požadavky:

- Alespoň část aplikace je vypracována v jazyce C#
- Aplikace používá principy objektového programování
- Aplikace spolupracuje s databází pomocí technologie ADO.NET
- Aplikace implementuje přehledné a jednoduché GUI, které je vhodně odděleno od prezenční vrstvy
- Aplikace disponuje úplnou funkcionalitou
- Robustní mobilní aplikace funkční na platformě OS Android
- Dokumentace
- [Možné rozšíření] Multiplatformní aplikace (iOS, UWP, PWA)

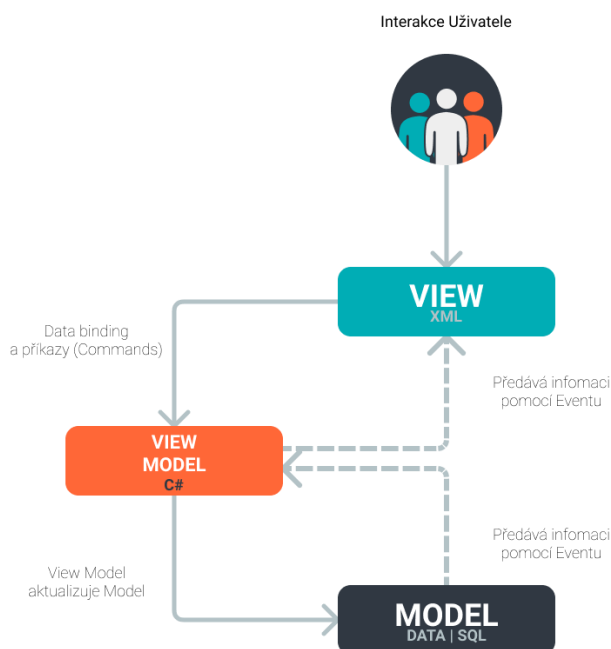
Požadavky na funkcionalitu:

- Registrace a evidence hudebníků/muzikantů
- Správa údajů a dovedností muzikantů
- Přiřazení muzikantů do hudebních skupin
- Registrace a evidence skupin
- Správa údajů hudebních skupin
- Zobrazení muzikantů a skupin v Aplikaci a jejich vyhledávání podle kritérií.
- Objevovat nové skupiny a muzikanty v okolí (Vyhledávání podle GPS lokace)
- Autentizace a Autorizace uživatele
- Persistentní ukládání těchto dat v databázi (SQLite)
- [Možné rozšíření] Mapa s vyobrazením skupin a hudebníků
- [Možné rozšíření] Komunikace přímo v aplikaci (Jednoduchý chatovací systém a oznámení)

2 Analýza

Mobilní aplikace je taková softwarová aplikace, která byla vytvořena speciálně pro chytré telefony, tablety, emulátory a další mobilní zařízení. Technologie .NET poskytuje možnost vývoje takových aplikací v jazyce C# pomocí otevřené platformy Xamarin. Xamarin aplikace lze vyvíjet pro jednotlivá zařízení s výchozími ovládacími prvky s operačními systémy Android, iOS, tvOS, watchOS, macOS a Windows. Nadstavbový framework Xamarin.Forms pak umožňuje vyvíjet jednu aplikaci zároveň pro Android, iOS i Windows a sdílet až 90 % kódu. Proto budu volit v této semestrální práci právě framework Xamarin.Forms.

Konstrukce aplikací Xamarin respektuje softwarový architektonický vzor Model-View-ViewModel (dále jen MVVM). V této architektuře Model představuje veškerou logiku a zároveň data s kterými pracuje. View je vrstva uživatelského rozhraní a ViewModel spojuje tyto dvě vrstvy, tedy připravuje data z modelové vrstvy pro zobrazení v GUI (grafickém uživatelském prostředí) a zároveň reaguje na všechny změny v GUI, které propisuje zpět do Modelu. Jedná se tedy o tzv. TwoWay Binding, kdy úprava dat uživatelem ve View provede změnu hodnot ve ViewModelu a změna hodnot ve ViewModelu na druhou stranu aktualizuje data ve View. Při implementaci budu tento architektonický vzor rovněž respektovat.



Obrázek 1 - Diagram MVVM Xamarin aplikace

Ačkoli technologie .NET umožňuje práci s velkým množstvím SQL či NoSQL databázových systémů, zůstává standardem s největší podporou systém SQLite, který rovněž využívají a doporučují samotní vývojáři platformy Xamarin. Pro účely tohoto projektu je SQLite zcela dostačující, budu tedy implementovat právě SQLite, a to nejprve lokálně pro testování a později hostovaný na vzdálené službě.

3 Implementace

3.1 Rozdělení aplikace

Aplikace a její rozdělení vychází z architektonického vzoru MVVM. Je proto rozdělena do více částí, z nichž každá plní zvláštní účel. V kořenovém adresáři se nachází tři podsložky, každá představující zvláštní projekt:

- **Project_Ensemble** – Hlavní projekt aplikace. Zde se nachází 90 % celkového zdrojového kódu a implementace funkcí, tříd, metod a grafického uživatelského rozhraní. Pomocí tohoto projektu jsou pak generovány výsledné aplikační balíčky pro všechny operační systémy. Pouze kód, který nelze sdílet napříč všemi systémy a platformami je pak implementován v některých z dalších podprojektů.
- **Project_Ensemble.Android** – Projekt obsahující kód specifický pro platformu android. Obsahuje pouze implementaci základní třídy (MainActivity), uchovává si vlastní reference na potřebné závislosti a balíčky a implementuje specifický kód pro Autentifikaci uživatele pomocí služby Firebase.
- **Project_Ensemble.iOS** – Obsahuje pouze základní implementaci vstupní třídy pro zařízení s operačními systémy iOS. Uchovává si závislosti a požadavky specifické pro platformu iOS.

3.1.1 Dělení součástí hlavního projektu

Zdrojové soubory hlavního projektu jsou rozvrženy tak, aby co nejlépe respektovali architektonický vzor MVVM. Popis a obsah jednotlivých složek:

- **bin** – Obsahuje soubory pro překlad a výslednou přeloženou aplikaci v režimu Debug a Release.
- **Constants** – Neměnné třídy s konstantami využívanými napříč celou aplikací. Obsahuje Soubor ColorConstants, který definuje barvy využívané v aplikaci a MaterialDesignIcons.cs - Soubor, který definuje kódy (code point) téměř všech ikon z písma material design icon od společnosti Google.
- **Helpers** – Pomocné třídy, které jsou využívány v aplikaci pro převody (Converter) a výpočty vzdálenosti.
- **Models** – Datové modely aplikace. Část **MVVM**. Definují datové struktury a třídy pro přenos dat, které jsou ukládány v databázi a s kterými pracují ViewModely aplikace.
- **obj** – Mezi-soubory překladače.
- **Resources** – Obsahuje datové zdroje využívané v aplikaci jako jsou písma a obrázky.
- **Services** – Služby aplikace, které slouží jako mezivrstva pro komunikaci s dalšími technologiemi jako např. Firebase Storage, Firebase Auth, API Google map, nebo databázové spojení.
- **Views** – Část **MVVM**. Všechny soubory definující grafické uživatelské rozhraní, prvky a rozložení jednotlivých stránek celé aplikace. Každá stránka je definována XAML souborem, udávající strukturu a vizuální elementy, se kterými uživatel interaguje; a tzv. Codebehind souborem, který umožňuje nastavovat těmto vizuálním prvkům vlastnosti, přímo je upravovat a měnit vizuální podobu stránky pomocí kódu.
- **ViewModels** – Část **MVVM**. Obsahuje všechny třídy představující Viewmodely, které slouží jako spojení modelů a uživatelského rozhraní, udržují stav aplikace a pomocí Bindingu propojují svoje atributy s ovládacími prvky v zobrazení. Provádí se v nich filtrování dat a jejich transformace pomocí příslušných metod.

Další důležité soubory v kořenovém adresáři

- **App.xaml a AppShell.xaml** – Úvodní zobrazení aplikace. Definuje menu, vizuální styly a registruje stránky pro přechod pomocí pseudo "url" adres.
- **Configuration.cs** - Soubor se základní konfigurací aplikace. Nastavuje například cestu a vlastnosti databáze.

3.2 Použité technologie

3.2.1 Xamarin.Forms

Xamarin je platforma spravovaná společností Microsoft určená pro vývoj nativních aplikací pro operační systémy Android, iOS, Windows a s komunitní podporou mnoha dalších. Výhodou platformy Xamarin je fakt, že oproti nativnímu vývoji lze mezi všemi platformami sdílet větší část kódu a zároveň však zůstává 100% přístup k nativním API každého z OS. Výstupem kompilovaného souboru platformy Xamarin je naprosto stejný jako při nativním vývoji. Xamarin tedy nemá negativní vliv na výkon výsledné aplikace.

Nevýhody Xamarinu vychází právě ze sdíleného kódu, neboť musí být tento kód přibalen každého balíčku pro výsledný OS a tak je jeho velikost větší než při nativním vývoji.

Xamarin.Forms je framework, nadstavbou Xamarinu, určený pro rychlou tvorbu GUI napříč všemi operačními systémy a umožňuje sdílet ještě větší součást kódu mezi všemi platformami než samotný Xamarin. Ve společné knihovně lze definovat komponenty, které jsou pak za běhu aplikace mapovány na nativní komponenty jednotlivých OS. Výsledkem je aplikace, která na všech podporovaných platformách vypadá (téměř) totožně.

Celá aplikace je realizována pomocí technologie Xamarin.Forms.

3.2.2 SQLite

SQLite (Hipp, 2021) je relační databázový systém obsažený v relativně malé knihovně napsané v programovacím jazyce C. Oproti běžným databázím založených na principu klient – server jakými jsou například MySQL nebo PostgreSQL se jedná pouze o jednoduchou knihovnu, kde jsou data jsou uloženy ve formě souboru na stejném zařízení, na kterém je nasazen software, který k ní přistupuje.

Základní myšlenkou SQLite je jednoduchost (odtud název lite) a její dostupnost. Projekt SQLite je šířen pod public domain licencí s otevřeným kódem.

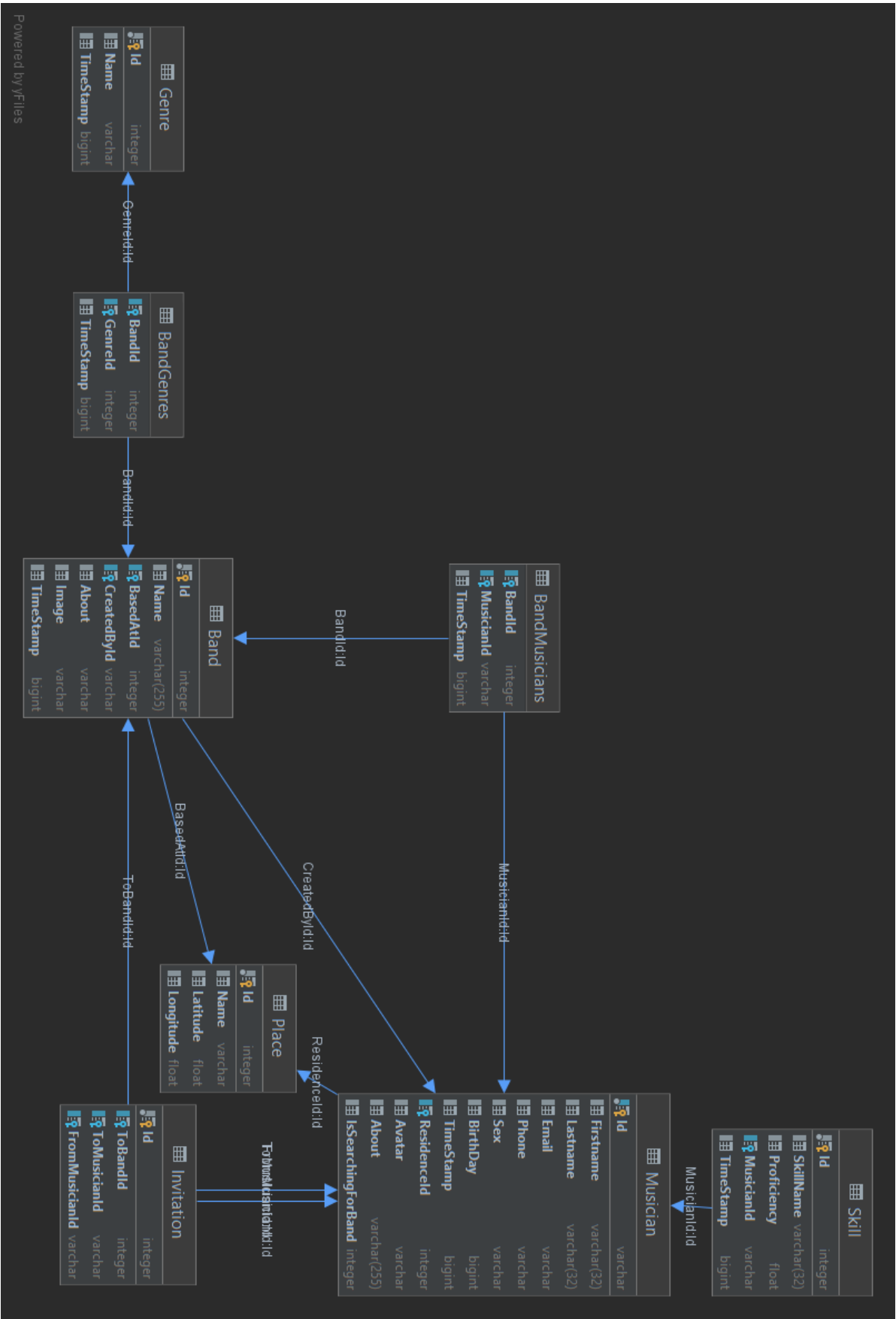
Výhody

- Jednoduchost
- Malá velikost
- Databáze v jednom souboru
- Nízká konfigurace a režie
- Paralelní přístup (čtení)

Nevýhody

- Nižší výkon oproti běžnému přístupu (zejména při zápisu)
- Každý zápis – vlastní transakce
- Nemožnost paralelního zápisu

V této aplikaci je SQLite databáze využíván pro ukládání veškerých textových dat uživatelů skupin a míst (Modelů ve složce Models). Pro finální nasazení aplikace je potřeba lokální SQLite databázi nahradit online REST API či GraphQL databází. SQLite databáze v aplikaci může zůstat například pro cachování dat.



Obrázek 2 - Diagram databáze

3.2.3 Firebase

Firebase je platforma pro vývoj mobilních a webových aplikací. Obsahuje řadu knihoven a funkcí pro mobilní (iOS, Android) a web (Javascript) platformy. Umožňuje pomocí těchto funkcí rychle a efektivně implementovat množství standardních nástrojů, které jsou pro nezbytné pro vývoj složitějších aplikací. Lze ji využít například pro Real-Time databázi, autorizační server, cloudové úložiště, push notifikace, statistiky apod. Všechny nástroje jsou dostupné zdarma s určitým omezením a pevně danými kvótami. Nabízí však spoustu tarifů, které se přizpůsobují potřebám každé z vyvíjených aplikací.

V projektu je tato platforma použita jako autorizační server a cloudové úložiště pro obrázky uživatelů a skupin.

3.2.3.1 *Firebase Authentication*

Firebase Authentication poskytuje back-end služby, sady SDK a knihovny uživatelského rozhraní pro ověřování uživatelů pomocí standardů O-Auth 2.0. Podporuje ověření pomocí hesel, telefonních čísel, populárních aplikací a sociálních sítí jako Google, Facebook, Twitter a další.

3.2.3.2 *Firebase Storage*

Firebase Storage je výkonná a jednoduchá služba pro ukládání dat. Umožňuje nahrávat a stahovat uživatelská data, stará se navazování a obnovování ukládání a stahování v závislosti na kvalitě sítě a poskytuje způsob jak přístup k souborům zabezpečit a provázat například s autorizačním serverem firebase.

3.2.4 Google Maps Platform

Google mapy (Společnost Google, 2021) jsou nejpoužívanější mapovou službou na světě s více než miliardou aktivních uživatelů v průběhu každého měsíce. Google Maps Platform je kolekce API, nad funkcionalitou Google map pro vývoj aplikací, webových stránek a pro zařízení využívající geologická data.

V této aplikaci je využíváno Google Places API (Společnost Google, 2021) a to pro automatickou predikci, vyhledávání a doplnění měst v České republice při výběru Bydliště či Sídla skupiny.

4 Uživatelská příručka

4.1 Předpoklady a instalace

Aplikace byla vyvíjena a testována na systému Android ve verzi 9.0. Aby mohlo být zajištěno správné a plynulé fungování aplikace je potřeba, aby zařízení, na kterém je aplikace instalována disponovalo verzí androidu 9.0 (API level 28) a vyšší.

Pro instalaci aplikace je potřeba aby zařízení mělo povolené instalování aplikace z neznámých/neověřených zdrojů. Toho lze docílit v nastavení mobilního zařízení. Postup se liší v závislosti na verzi operačního systému. Běžný postup:

1. Přejděte do **Nastavení > Zabezpečení**
2. Zaškrtněte možnost **“Neznámé zdroje”**
3. Potvrďte vyskakovací okno možností **OK**
4. Vyberte **“Důvěřovat”**

Poté lze aplikaci nainstalovat kliknutím na balíček APK v úložišti zařízení.

UPOZORNĚNÍ: Aplikace v aktuálním, testovacím stavu využívá pouze lokální databáze. Pro otestování je tedy potřeba veškeré akce provádět na jednom zařízení a registrovat se v tomto zařízení vícekrát. Data databáze se mezi jednotlivými zařízeními NESDÍLEJÍ!

4.2 Přihlášení

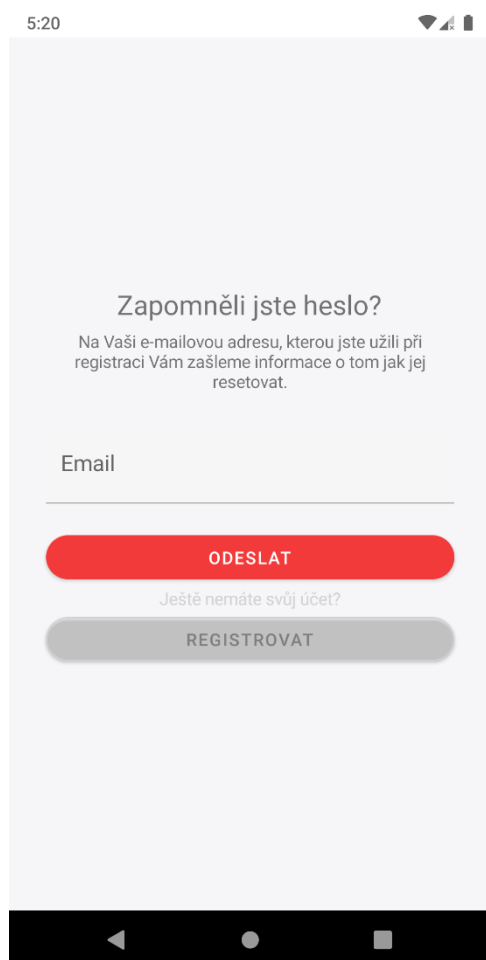
Úvodní obrazovka, se kterou se uživatel setká (pokud již není přihlášen) je obrazovka přihlášení. Uživatel rovněž bude vyzván k povolení aplikaci k přístupu k GPS souřadnicím zařízení.

Aplikace využívá GPS souřadnice uživatele k identifikaci jeho lokace a na její základě pak navrhuje uživateli výsledky v jeho okolí a vypočítává vzdálenosti mezi uživatelem a ostatními muzikanty a skupinami v databázi.

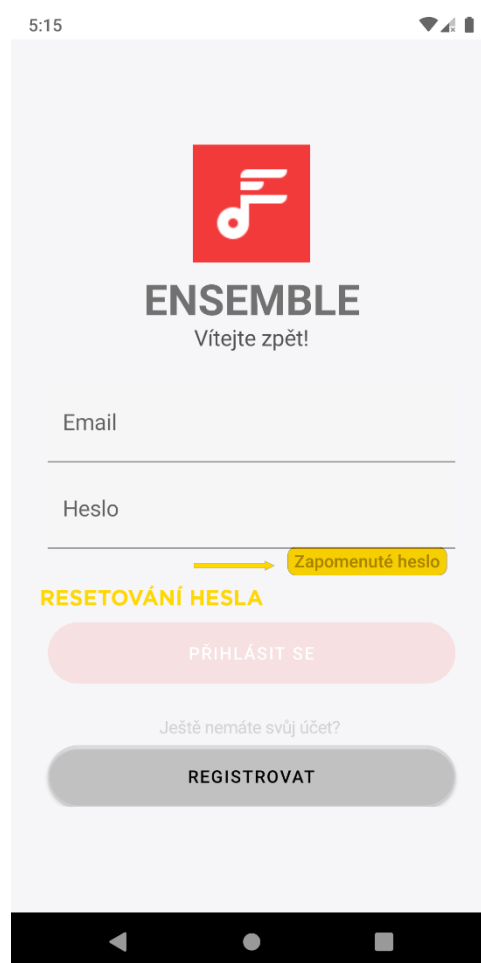
Pokud ještě uživatel nemá účet, musí se k přístupu do aplikace registrovat. To provede na stránce registrace po kliknutí na tlačítko **Registrovat**.

Pokud již uživatel má vytvořený účet v aplikaci, může se přihlásit vyplněním příslušných přihlašovacích údajů a stisknutím možnosti **Přihlásit se**.

Může se stát, že uživatel zapomene heslo, v takovém případě lze na obrazovce přihlášení pod vstupním polem s heslem kliknout na možnost **“Zapomenuté heslo”**. Zvolením této možnosti přenese aplikace uživatele na stránku, kde si může nechat zaslat na svou emailovou adresu zprávu s postupem k resetování přístupového hesla.



Obrázek 4 - Stránka zapomenutého hesla



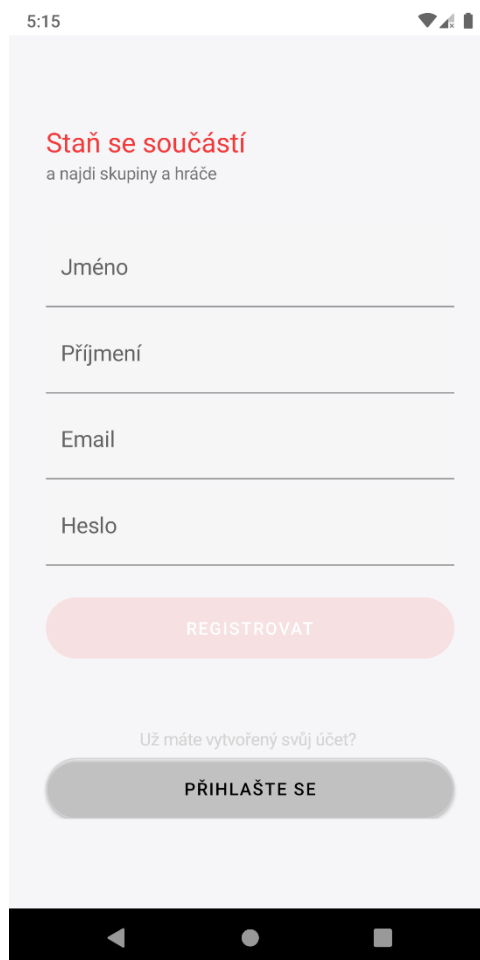
Obrázek 3 - Stránka přihlášení

4.2.1 Zapomenuté heslo

Na této stránce může uživatel požádat o resetování svého vstupního hesla. Vyplněním pole e-mail a kliknutím na možnost **Odeslat** je uživateli zaslán na jeho e-mailovou adresu, kterou si zvolil v průběhu registrace zpráva, pomocí které může změnit své heslo pro přihlášení do aplikace.

4.3 Registrace

Stránka registrace je klíčovou stránkou pro systém Ensemble, neboť neregistrovaný uživatel nemá do systému žádný přístup. Pro úspěšnou registraci je potřeba vyplnit příslušná pole registračního formuláře sestávající z *Jména*, *Příjmení*, *Registračního emailu* a *Hesla*, které chce uživatel použít pro přihlašování do systému. Po vyplnění příslušných polí se uživatel registruje kliknutím na tlačítko **Registrovat**. Pokud byla registrace úspěšná, je uživatel automaticky přenesen na hlavní stránku (seznam muzikantů a skupin) aplikace a má k ní nyní plný přístup. Přihlášenému uživateli je po spuštění aplikace rovnou zobrazena domovská stránka se seznamem muzikantů a skupin. Není tedy potřeba se přihlašovat do aplikace při každém spuštění.



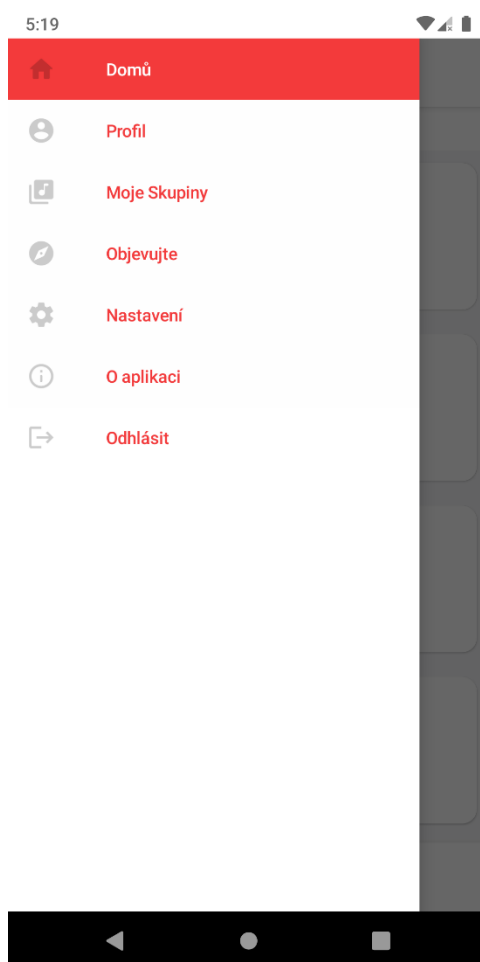
Obrázek 5 - Stránka registrace

4.4 Hlavní navigace

Hlavní navigace aplikace je dostupná po kliknutí na tlačítko menu v levém horním rohu. Tato navigace slouží pro orientaci v celém systému aplikace a se po kliknutí na jednu z položek umožňuje přesun na příslušné stránky:

- **Domů** – Hlavní stránka aplikace zobrazující seznam muzikantů a skupin registrovaných v databázi. Rovněž umožňuje tyto záznamy řadit pomocí jednoduchých filtrů a zobrazovat jejich detail po kliknutí.
- **Profil** – Profilová stránka uživatele. Na této stránce je možné si prohlédnout jaké informace a v jakém formátu zobrazuje uživatel ostatním uživatelům aplikace. Rovněž umožňuje základní informace editovat a upravovat seznam dovedností.

- **Moje skupiny** – Stránka zobrazuje všechny skupiny, které uživatel vytvořil. Z této stránky lze také vytvořit skupiny nové, spravovat jejich obsah a členy a zobrazit si aktivní obdržené pozvánky do skupin.
- **Objevujte** – Na této stránce je uživateli zobrazena náhodná posloupnost pěti muzikantů a skupin z databáze. Tato posloupnost se po každé aktualizaci mění. Rovněž tato stránka slouží pro vyhledávání skupin a uživatelů podle jejich jmen a názvů.
- **Nastavení** – Slouží pro základní nastavení aplikace. Aktuálně poskytuje pouze možnost kompletního smazání účtu. Dalším potenciálním nastavením, které je možné přidat je např. možnost změny tmavého/světlého tématu aplikace, změna přihlašovacího emailu a hesla apod.
- **O aplikaci** – Poskytuje základní informace o nainstalované aplikaci. Její název, cíl, autora, technologie a aktuální verzi.
- **Odhlásit** – Slouží k odhlášení uživatele z aplikace



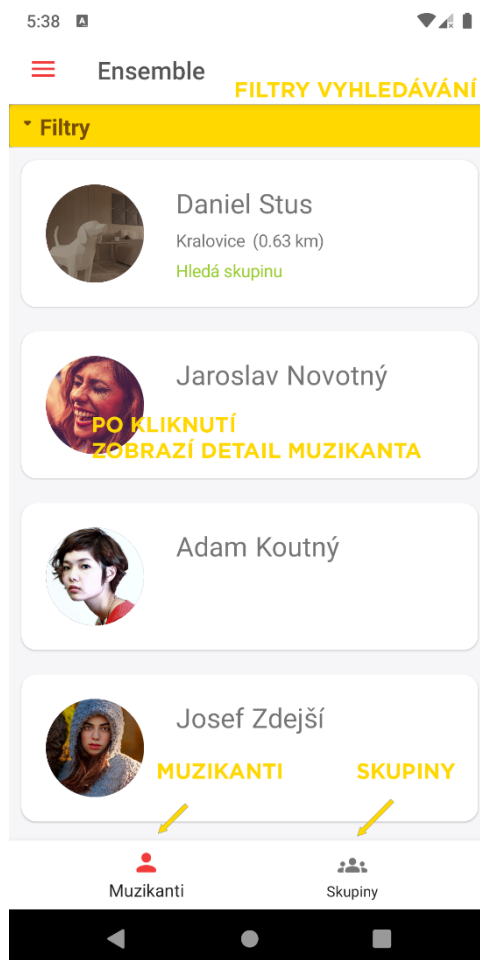
Obrázek 6 - Hlavní navigace

4.5 Domů/Katalog

Hlavní stránka aplikace, která se zobrazí přihlášenému uživateli. Z této stránky lze zobrazit seznam všech muzikantů či skupin, které jsou registrovány v systému. Rovněž jde tento seznam seřadit podle filtrů dostupných v horní části obrazovky. Dostupné filtry jsou: **Název, Vzdálenost a Datum přidání**.

Veškeré seznamy v aplikaci lze aktualizovat přetažením seznamu směrem nahoru.

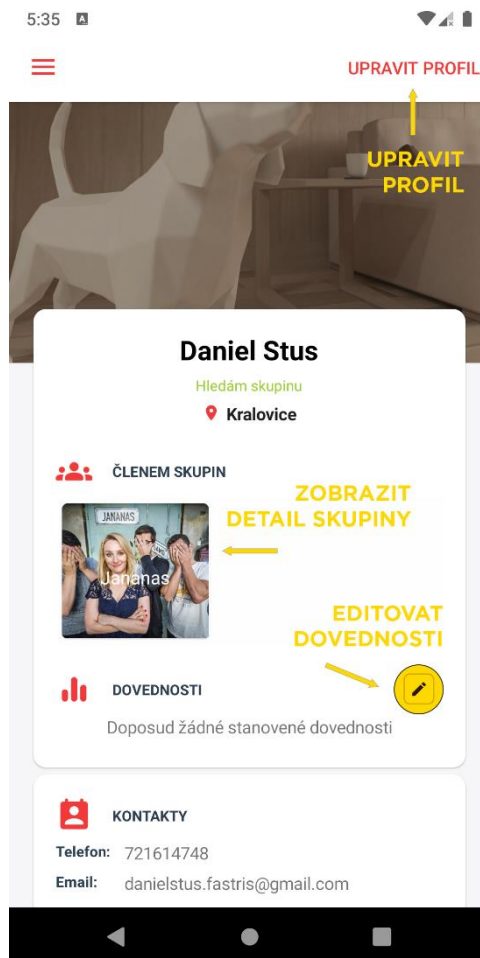
Kliknutím na muzikanta nebo skupinu zobrazíte jeho detail.



Obrázek 7 - Domovská stránka / Katalog

4.6 Profil

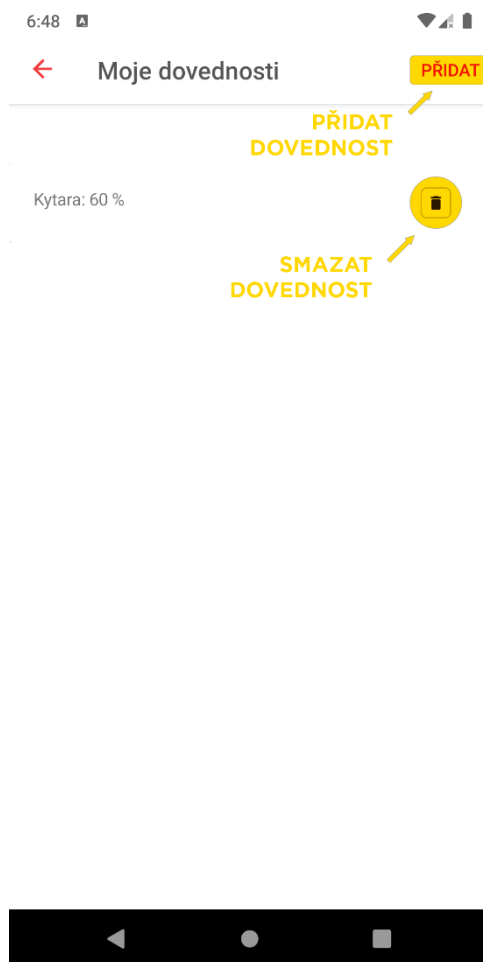
Profilová stránka umožňuje uživateli přehledně zobrazit informace, které sdílí v systému. Tak jak se zobrazuje stránka profilu se až na malé výjimky zobrazí profil uživatele i ostatním, kteří jej vyhledají v databázi. Na této stránce je uživateli umožněno svůj profil, fotografii a sdílené informace editovat pomocí tlačítka **Upravit profil** a editovat **Seznam dovedností**, kliknutím na tlačítko s tužkou vedle nápisu **DOVEDNOSTI**.



Obrázek 8 - Profil uživatele

4.6.1 Dovednosti

Správa dovedností, dostupná po kliknutí na tlačítko pro upravení dovedností z profilu uživatele, umožňuje přidávat/mazat a zobrazovat dovednosti uživatele. Každý uživatel může definovat až 4 vlastní dovednosti a lze je přidat pomocí tlačítka v pravé horní části obrazovky. Dovednost se skládá z názvu a procentuálního odhadu odbornosti. Uživatel své dovednosti může kdykoliv smazat kliknutím na tlačítko s ikonou koše vedle názvu příslušné dovednosti v seznamu.



Obrázek 9 - Správa dovedností

4.7 Detail skupiny

Stránka s detailem skupiny se zobrazí pokaždé, když uživatel klikne na skupinu v kterémkoliv seznamu v celé aplikaci. Na této stránce jsou zobrazeny podrobné informace o skupině, včetně seznamu jejích členů. Po kliknutí na jméno některého člena ze seznamu se zobrazí detail příslušného muzikanta.



Obrázek 10 - Detail skupiny

4.8 Moje skupiny/Pozvánky

Na stránce „Moje skupiny“ vidí uživatel seznam všech skupin, které doposud vytvořil. U každé vytvořené skupiny lze zobrazit dostupné akce přetažením její karty vpravo či vlevo. Potažení vpravo zobrazí tlačítko pro správu členů (**Členové**) skupiny. Přetažením vlevo pak tlačítka **Editovat** a **Smazat** skupinu. Vybráním jedním z tlačítek se uživatel dostane na příslušnou podstránku, kde může obsah a členy skupiny upravit nebo

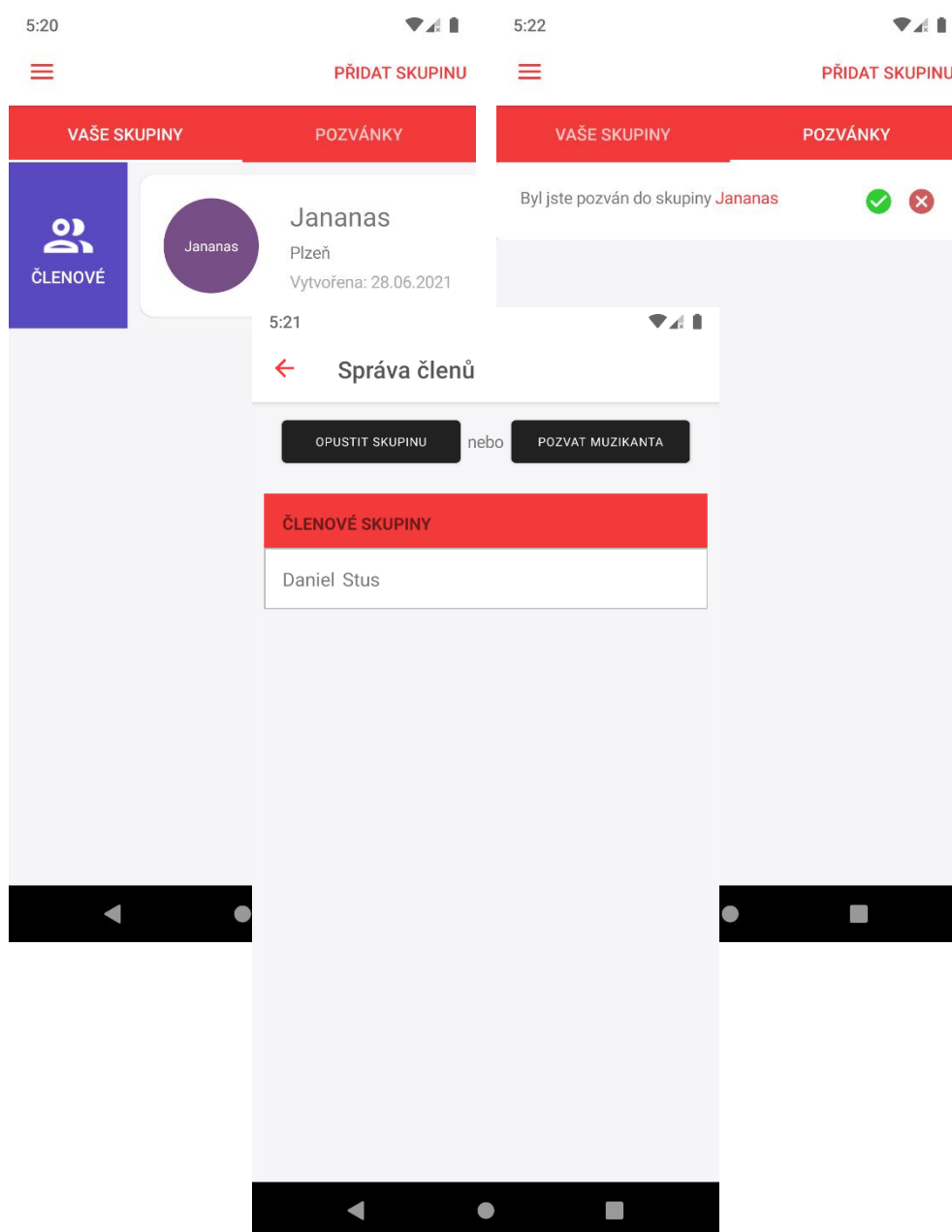
skupinu

zcela

smazat.

Stránka rovněž obsahuje záložku „Pozvánky“. Zde jsou uživatelům zobrazeny veškeré doposud nevyřešené aktivní pozvánky do skupin od jejich zakladatelů. Na každou z pozvánek může uživatel zareagovat přijutím (zelené tlačítko) či odmítnutím (červené tlačítko) pomocí tlačítek vedle popisu poznámky. Pokud uživatel pozvánku přijme, stává se ihned členem skupiny, do které byl pozván. Pokud ji zamítne, bude pozvánka odstraněna z databáze a uživatel může být v budoucnu znovu pozván.

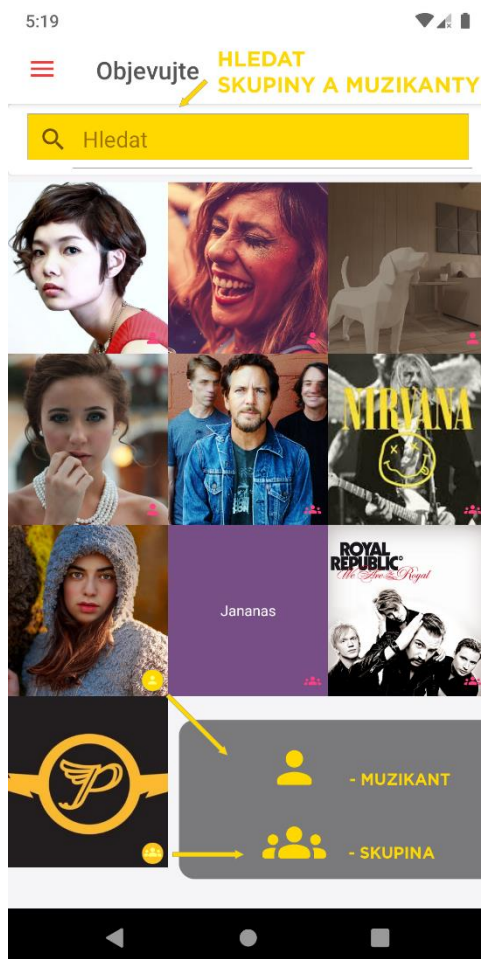
Spravovat členy skupiny může pouze její zakladatel, proto pokud chce uživatel skupinu, které je členem, avšak ne vlastníkem, opustit, musí kontaktovat jejího správce.



Obrázek 11 - Moje skupiny a pozvánky

4.9 Objevujte

Stránka objevujte zobrazuje při každé aktualizaci nový náhodný výběr pěti muzikantů a pěti skupin v databázi. Rozlišení skupiny a muzikanta je indikováno ikonou v pravé dolní části čtverce muzikanta, resp. skupiny. Tato stránka rovněž slouží pro vyhledávání muzikantů a skupin dle jejich jmen a názvů pomocí vyhledávacího pole v horní části obrazovky. Po kliknutí na skupinu či muzikanta se zobrazí jejich detail.

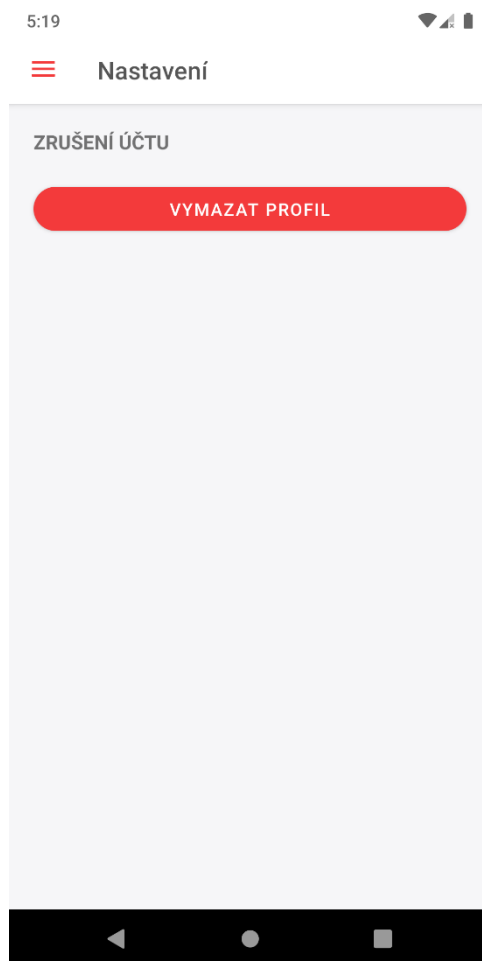


Obrázek 12 - Stránka objevujte

4.10 Nastavení

Stránka **Nastavení** slouží pro základní konfiguraci aplikace. Aktuálně poskytuje pouze možnost kompletního smazání účtu, kliknutím na tlačítko **Vymazat profil**. Po kliknutí na toto tlačítko je uživatel vyzván k zadání svého aktuálního hesla pro ověření. Pokud zadá správné heslo, je tímto smazán jeho účet z autorizačního serveru a všechny jeho záznamy skupin, pozvánek a míst z databáze. Uživatel se může kdykoliv znovu do systému registrovat.

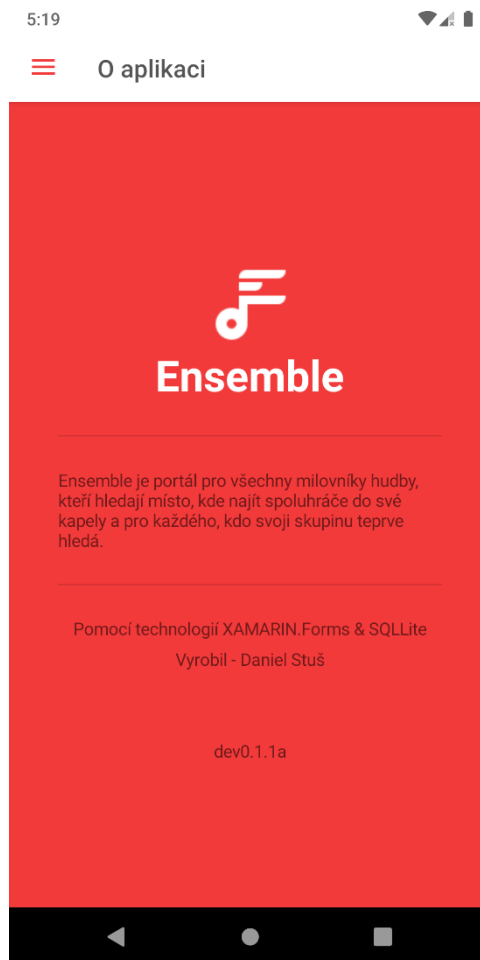
Dalším potenciálním nastavením, které je možné přidat je např. možnost změny tmavého/světlého tématu aplikace, změna přihlašovacího emailu a hesla apod.



Obrázek 13 - Stránka nastavení aplikace

4.11 O aplikaci

Stránka „O aplikaci“ umožňuje uživateli zobrazit základní informace o cíli aplikace, technologiích využitých při její implementaci, jméno autora a aktuální nainstalovanou verzi.



Obrázek 14 - Stránka o aplikaci

4.12 Odhlásit

Možnost **Odhlásit** v hlavní navigaci odhlásí aktuálního uživatele z aplikace a přenesení jej na stránku přihlášení. Odhlášenému uživateli je při spuštění aplikace zobrazena namísto domovské stránky se seznamy stránka přihlášení.

5 Závěr

Výsledná aplikace splňuje požadavky zadání a umožňuje správu uživatelů a skupin.

Při implementaci práce jsem narazil na nespočet problémů, které byli zapříčiněny jak nepříliš dobrou znalostí technologie Xamarin.Forms, tak faktem, že tato platforma má být v aktuálním roce 2021 nahrazena novým systémem .NET MAUI a proto se již netěší tak velké podpoře a na opravu funkcí je třeba čekat do další verze.

Vytvoření této aplikace mi pomohlo lépe porozumět problematice mobilních aplikací, operačních systémů Android a iOS a jejich způsobu zacházení s oprávněními k přístupu nativních API.

Aplikace pracuje pouze s lokální databází. Pro finální nasazení a využitelnost aplikace by bylo potřeba lokální SQLite databázi nahradit online REST API či GraphQL databází. SQLite databáze v aplikaci může zůstat například pro cachování dat.

V aktuálním stavu představuje aplikace ověření koncepce a stabilní základ, který je dále možné obohatit například o mapu zobrazující uživatele a skupiny v okolí, push notifikace nebo o přímý chatovací systém mezi uživateli.

Ačkoli bych s nově nabitými vědomostmi a znalosti bezesporu při re-implementaci přistupoval k samotné struktuře programu odlišně, jsem i přesto s konečným výsledkem spokojen.

6 Reference

Hipp, D. Richard. 2021. About. *SQLite*. [Online] 2021. [Citace: 28. řerven 2021.] <https://www.sqlite.org/about.html>.

Společnost Google. 2021. Google Places API. *Google Places API*. [Online] Google, 2021. <https://developers.google.com/maps/documentation/places/web-service/overview>.

Obrázek 1 - Diagram MVVM Xamarin aplikace.....	4
Obrázek 2 - Diagram databáze.....	7
Obrázek 3 - Stránka přihlášení	9
Obrázek 4 - Stránka zapomenutého hesla	9
Obrázek 5 - Stránka registrace.....	10
Obrázek 6 - Hlavní navigace.....	11
Obrázek 7 - Domovská stránka / Katalog	12
Obrázek 8 - Profil uživatele	13
Obrázek 9 - Správa dovedností.....	14
Obrázek 10 - Detail skupiny	15
Obrázek 11 - Moje skupiny a pozvánky	16
Obrázek 12 - Stránka objevujte	17
Obrázek 13 - Stránka nastavení aplikace	18
Obrázek 14 - Stránka o aplikaci.....	19

A Programátorský deník

Datum	Aktivita	Délka [hod]
12.04.	Studium Xamarin.Forms	5
13.04.	Založení projektu	0,5
	Úprava základní šablony	3
14.04.	Studium Xamarin.Forms	3
15.04.	Studium MongoDB v Xamarin	1
	Implementace MongoDB	3
16.04.	Tvorba základních stránek	3
17.04.	Přepřpracování projektu (Opuštění od MongoDB)	1
	Přepřpracování projektu (Přechod na MVVM)	3
18.04.	Studium SQLite v Xamarin	3
	Implementace SQLite	3
	Implementace Stránek (About, Catalog, ListBand, ListMusician, ProfilePage...)	4
19.04.	Dokončování Implementace a provázání s SQLite (CRUD)	5
	Github - Readme - Diagram	1,5
20.04.	Příprava Dokumentace	3
25.04.	Firebase autentizace	5
26.04.	Implementace žánrů a příslušných stránek	4
28.04.	Úprava vzhledu aplikace, QoL vylepšení	3
30.04.	Profilová stránka, Přepřpracování musician uid (souhlasí s firebase databází)	4
01-20.05.	Vyhldávání míst (Google maps api), Bottom sheet a změny vzhledu	25
25.05.	Filtiry	3
30.05.-20.06	Systém pozvánek, Firebase storage, Vyhledávání, Stránka objevování, Nové logo, úpravy vzhledu a rozložení, začišťnění kódu a kódová dokumentace	40
20.06.-28.06.	Dokončení textové dokumentace a uživatelské příručky	8
		134