

Bakalářská práce



České
vysoké
učení technické
v Praze

F3

Fakulta elektrotechnická
Katedra počítačů

Sociální pracovník v mobilu

Karel Zavadil

Školitel: Ing. Martin Komárek

Studijní program: Softwarové inženýrství a technologie

Květen 2020

Poděkování

Děkuji ČVUT za to, že je tak skvělou univerzitou. Děkuji všem mým učitelům za to, že se mnou měli trpělivost. Děkuji všem, kdo pomáhají bojovat proti epidemii COVID-19 a také těm, kdo v této době zachovávají chladnou hlavu.

Prohlášení

Prohlašuji, že jsem předloženou práci vypracoval samostatně a že jsem uvedl veškeré použité informační zdroje v souladu s Metodickým pokynem o dodržování etických principů při přípravě vysokoškolských závěrečných prací.

V Praze, 14. května 2020

Karel Zavadil

Abstrakt

Mobilní aplikace, která umožňuje snadno najít kontakt na pomoc v krizových sociálních situacích. Aplikace je určena pro osoby v sociální krizi, sociální pracovníky, státní úředníky, kteří pracují s osobami v sociální krizi, a kohokoli dalšího, kdo chce vyhledat pomoc pro druhou osobu.

Aplikace si klade za cíl řešit následující problémy:

- lidé nemají informace, které jim mají pomoci, nebo s nimi neumí pracovat
- zejména je těžké najít konkrétní kontakt (tel. číslo) pro konkrétní sociální situaci
- sociální služby o sobě nevědí a přehazují si klienta ze služby na službu

Mobilní aplikace nejprve nabídne uživateli řadu jednoduchých otázek. Na základě zvolených odpovědí poté aplikace vyhodnotí situaci klienta a nabídne návod na další postup a kontakty na relevantní pomoc. Formulace otázek a odpovědí vkládá správce pomocí webové aplikace.

Klíčová slova: sociální pracovník, sociální krize, softwarové inženýrství, 3-vrstvá architektura, mobilní aplikace

Školitel: Ing. Martin Komárek

Abstract

Mobile application that helps to find helpful contact in critical social situations. Application is aimed at people in social crisis, social workers, civil servants that work with people in social crisis and anyone else who wants to find help for another person.

Goal of the application is to solve following problems:

- people don't have information that could help them or they don't know how to work with it
- it is especially difficult to find specific contact (phone number) for specific social situation
- social services are not aware of each other and dispatch clients from one social service to another

First, mobile application will present user with series of simple questions. Application will then diagnose client situation based on chosen answers and provide a recommendation of next steps together with contact for relevant help. Wording of questions and answers is typed in by administrators through a web application.

Keywords: social worker, social crisis, software engineering, 3-tier architecture, mobile application

Title translation: Social Worker in Mobile Phone

Obsah

1 Úvod	1	4.2 Vývoj	29
1.1 Cíle projektu	1	4.3 Finance	29
1.2 Řešení ve zkratce	2	4.4 Marketing	30
2 Analýza systému	3	4.5 Problémy	30
2.1 Cílové skupiny	3	4.5.1 Scraping	30
2.1.1 Poskytovatelé sociálních služeb	3	4.5.2 Geolokace	31
2.1.2 Osoby v sociální krizi	3	4.5.3 Vkládání obsahu	31
2.1.3 Kohokoli dalšího, kdo chce vyhledat pomoc pro druhou osobu	3	4.6 Zpětná vazba	32
2.2 Business požadavky	4	5 Budoucnost projektu	33
2.3 Uživatelské role	5	6 Závěr	35
2.4 Funkční požadavky	5	A Seznam použitých zkratk	37
2.4.1 Anonym	5	B Literatura	39
2.4.2 Správce	5	C Externí přílohy	41
2.4.3 Superuser	5	C.1 PNG pro draw.io	41
2.5 Kvalitativní požadavky	6	C.2 Zdrojové kódy	41
2.6 Mobilní aplikace	6	D Zadání práce	43
2.6.1 Případy užití mobilní aplikace	6		
2.6.2 Navigace a návrh obrazovek	9		
2.7 Webová aplikace	9		
2.7.1 Případy užití webové aplikace	9		
2.7.2 Navigace	15		
2.7.3 Návrhy obrazovek	16		
2.8 Sběr kontaktů - scraper	18		
2.9 Datový model	19		
3 Návrh systému	23		
3.1 Architektura systému	23		
3.1.1 Databáze	23		
3.1.2 Mobilní aplikace	23		
3.1.3 Aplikační server	24		
3.1.4 Webový server	25		
3.1.5 Webová aplikace	25		
3.2 Testování	25		
3.2.1 Mobilní aplikace	25		
3.2.2 Aplikační server	25		
3.2.3 Webová aplikace	26		
3.3 Sestavení a nasazení	26		
3.3.1 Aplikační server	26		
3.3.2 Mobilní aplikace	26		
3.4 Bezpečnost	26		
3.4.1 Zabezpečení proti ztrátě dat	26		
3.4.2 Zabezpečení proti neoprávněné změně nebo odcizení dat	27		
4 Realizace	29		
4.1 Tým	29		

Obrázky

2.1 Případy užití mobilní aplikace. (Příloha 2)	6
2.2 Navigace a návrh obrazovek pro mobilní aplikaci. (Příloha 3)	10
2.3 Případy užití webové aplikace. (Příloha 10)	11
2.4 Navigace ve webové aplikaci. (Příloha 9)	15
2.5 Úvodní stránka. (Příloha 11) . . .	16
2.6 Přehled otázek. (Příloha 8)	17
2.7 Editace otázky. (Příloha 7)	18
2.8 Editace kontaktu. (Příloha 6) . .	19
2.9 Reprezentace struktury dotazníku. (Příloha 4)	20
2.10 Datový model. (Příloha 1)	21
3.1 Součásti systému. (Příloha 5) . . .	24
4.1 Rozdělení negativních zpětných vazeb podle typu problému.	32

Tabulky

4.1 Využití peněz z grantu (v Kč). . .	30
--	----

Kapitola 1

Úvod

Sociální pracovník v mobilu je mobilní aplikace, která umožňuje snadno najít kontakt na pomoc v krizových sociálních situacích. Jak je patrné již z názvu, aplikace se snaží přenést alespoň část práce sociálního pracovníka do mobilního telefonu.

Projekt se dostal do inkubačního akceleračního programu **Laboratoř Naddace Vodafone** a obdržel finanční podporu ve formě grantu.

1.1 Cíle projektu

Existuje množství organizací, ať už státních nebo nestátních, které se zaměřují na pomoc lidem v krizi, ale pro člověka, který se ocitne v tíživé sociální situaci, je často těžké tu správnou organizaci najít a kontaktovat.

Hlavním cílem projektu je zprostředkování toho správného kontaktu pro osobu v sociální krizi. Různých sociálních krizových situací je mnoho, je jich ale konečný počet a existují instituce, které se na jejich řešení zaměřují. Náš projekt stojí na předpokladu, že pokud se podaří správně rozpoznat situaci klienta, tak je možné na tomto základě doporučit klientovi další postup a poskytnout kontakt na vhodnou instituci nebo organizaci, která mu může pomoci. Toto se nazývá *základní sociální poradenství* a je definováno zákonem [38].

Aplikace si klade za cíl řešit následující problémy:

- lidé nemají informace, které jim mají pomoci, nebo s nimi neumí pracovat
- zejména je těžké najít konkrétní kontakt (tel. číslo) pro konkrétní sociální situaci
- sociální služby o sobě nevědí a přehazují si klienta ze služby na službu
- zákon ukládá státním zaměstnancům v sociální sféře povinnou znalost sociální problematiky, ta ale v praxi často není dostatečná

■ 1.2 Řešení ve zkratce

Aplikace se nejprve pokusí diagnostikovat sociální situaci klienta tím, že nabídne uživateli řadu jednoduchých otázek. Na základě zvolených odpovědí poté aplikace nabídne návod na další postup a kontakty na relevantní pomoc.

Kontaktní údaje bude aplikace sbírat a průběžně aktualizovat z předem definovaných veřejně dostupných zdrojů jako např. open data nebo webové stránky.

Formulace otázek a odpovědí do aplikace vkládají správci - odborníci na sociální problematiku. Cílem této práce je navrhnout a vytvořit nástroj, který umožní profesionálním sociálním pracovníkům snadno vkládat data potřebná pro diagnózu sociální situace a její řešení a následně tato data využít k pomoci sociálně potřebným. Cílem této práce není vytvoření samotných dat pro diagnózu a řešení konkrétních sociálních krizových situací.

Kapitola 2

Analýza systému

2.1 Cílové skupiny

Aplikace je určena pro následující cílové skupiny:

2.1.1 Poskytovatelé sociálních služeb

Formy poskytování sociálních služeb a druhy zařízení sociálních služeb jsou vymezeny Zákonem o sociálních službách [38]. Naší cílovou skupinou budou pracovníci v těchto zařízeních. Mezi ně se počítají například:

- terénní sociální pracovníci
- zaměstnanci neziskových organizací poskytujících sociální služby
- státní úředníci, kteří pracují s osobami v sociální krizi

Zákon o sociálních službách ukládá všem poskytovatelům sociálních služeb povinnost poskytovat základní sociální poradenství:

Základní sociální poradenství poskytuje osobám potřebné informace přispívající k řešení jejich nepříznivé sociální situace. Základní sociální poradenství je základní činností při poskytování všech druhů sociálních služeb; poskytovatelé sociálních služeb jsou vždy povinni tuto činnost zajistit. [38]

Aplikace tedy poskytovatelům sociálních služeb usnadňuje jejich zákonnou povinnost poskytovat základní sociální poradenství i v oblastech, které nejsou jejich hlavním předmětem činnosti a odbornosti.

2.1.2 Osoby v sociální krizi

Osoby v sociální krizi mohou aplikaci použít, aby vyřešili svojí tíšňovou situaci.

2.1.3 Kohokoli dalšího, kdo chce vyhledat pomoc pro druhou osobu

Kdokoli může aplikaci použít, aby pomohl svému příbuznému nebo jinému osobě.

■ 2.3 Uživatelské role

Systém musí rozeznávat následující uživatelské role:

- anonym
- správce
- superuser

Anonym je jakýkoliv nepřihlášený návštěvník stránek nebo uživatel mobilní aplikace.

Správce je přihlášený uživatel, jehož hlavním úkolem je spravovat data pro dotazník.

Superuser je přihlášený uživatel, má všechna oprávnění správce a navíc má právo přidávat nebo odebírat jiné správce.

■ 2.4 Funkční požadavky

■ 2.4.1 Anonym

FR 1. Systém musí uživateli nabídnout dotazníkové otázky tak, jak je vložil správce.

FR 2. Systém musí na konci dotazníku zobrazit kontakty na pomoc podle aktuální lokality.

FR 3. Systém musí uživateli umožnit začít dotazník znovu od začátku.

FR 4. Systém musí uživateli umožnit vrátit se k předchozí otázce.

FR 5. Systém musí zobrazovat kontakty podle aktuální polohy uživatele.

■ 2.4.2 Správce

FR 6. Systém musí umožnit, aby se správce přihlásil pomocí jména a hesla.

FR 7. Systém musí správci umožnit vkládat, upravovat a mazat dotazníkové otázky a odpovědi a definovat druh zobrazených kontaktních informací.

FR 8. Systém musí umožnit, aby správce mohl změnit své heslo.

FR 9. Systém musí správci poskytnout informace o kvalitě obsahu a struktuře dotazníkových otázek.

■ 2.4.3 Superuser

FR 10. Systém musí umožnit, aby superuser mohl přidávat, upravovat a odebírat správce systému.

2.5 Kvalitativní požadavky

QR 1. Systém musí zobrazit dotazníkové otázky v jejich poslední verzi tak, jak je vložil správce.

QR 2. Systém musí zobrazovat aktuální kontakty na pomoc (musí reflektovat změny dat na zdroji).

QR 3. Systém musí být zabezpečený proti změně nebo zcizení dat neoprávněnou osobou.

QR 4. Systém musí být zabezpečený proti ztrátě dat (všechna data musí být pravidelně zálohována).

QR 5. Mobilní aplikace musí fungovat na telefonech s operačním systémem Android.

QR 6. Mobilní aplikace musí být implementována způsobem, který později umožní snadný přenos na jiné operační systémy, zejména iOS (telefony od společnosti Apple).

2.6 Mobilní aplikace

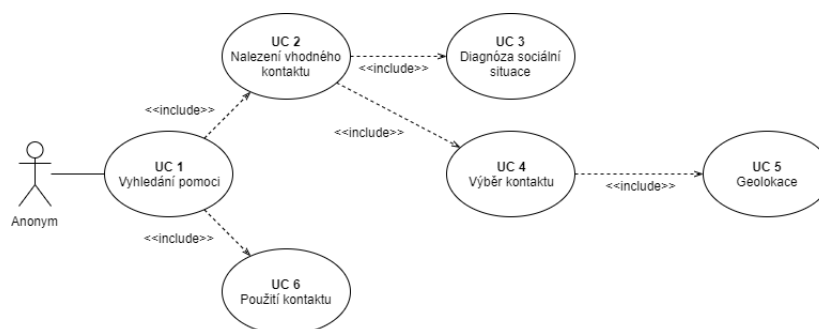
Mobilní aplikace je hlavní viditelnou součástí systému. Její úlohou je zobrazovat dotazníkové otázky a následně vhodné kontakty.

Aplikace si stahuje aktuální verzi otázek i kontaktů ze serveru.

Mobilní aplikace také sbírá statistické údaje o použití aplikace a zpětnou vazbu od uživatelů a odesílá je na server.

2.6.1 Případy užití mobilní aplikace

Diagram případů užití mobilní aplikace zachycuje obrázek 2.1.



Obrázek 2.1: Případy užití mobilní aplikace. (Příloha 2)

UC 1. Vyhledání pomoci

Role: anonym

1. Uživatel spustí aplikaci

2. UC 2 - Nalezení vhodného kontaktu

3. UC 6 - Použití kontaktu

UC 2. Nalezení vhodného kontaktu

Role: anonym

1. UC 3 - Diagnóza sociální situace

2. UC 4 - Výběr kontaktu

UC 3. Diagnóza sociální situace

Role: anonym

1. Systém zobrazí otázku

2. Uživatel zvolí vhodnou odpověď

3. IF [dotazník pokračuje] THEN GO TO UC 4 ELSE GO TO 1

UC 4. Výběr kontaktu

Role: anonym

Startovní podmínky: dokončen UC 3 (Diagnóza sociální situace)

1. Systém zobrazí text uzlu s kontaktem a data, která nevyžadují geolokaci (např. odkazy)

2. UC 5 - Geolokace

3. IF [známe polohu uživatele] THEN zobrazit kontaktní údaje podle lokace ELSE GO TO 2

UC 5. Geolokace

Role: anonym

1. Systém se pokusí zjistit polohu pomocí GPS

2. IF [nepodařilo se zjistit polohu] THEN GO TO 4

3. IF [uživatel zvolí "vybrat polohu ručně"] THEN GO TO 4 ELSE FINISH

4. Systém zobrazí výběr okresů

5. IF [uživatel zvolí "použít GPS"] THEN GO TO 1

6. Uživatel vybere svůj okres ze seznamu

UC 6. Použití kontaktu

Role: anonym

Startovní podmínky: dokončen UC 2 (Nalezení vhodného kontaktu)

1. Systém zobrazí data kontaktu
2. Uživatel zvolí, která data chce použít
3. CASE [typ kontaktních dat]
WHEN web THEN otevřít prohlížeč se zvolenou adresou
WHEN email THEN otevřít emailového klienta s připravenou zvolenou adresou
WHEN telefon THEN otevřít aplikaci pro telefonování a vytočit zvolené číslo

2.6.2 Navigace a návrh obrazovek

Návrh navigace v mobilní aplikaci a návrh obrazovek (tzv. *wireframes*) jsem se rozhodl spojit do jednoho dokumentu. To je možné vzhledem k malému rozměru displeje mobilního telefonu. Tento dokument zachycuje dvě spolu související věci a stále zůstává přehledný.

Návrh navigace mezi obrazovkami mobilní aplikace zachycuje obrázek 2.2. Návrh jsem vytvořil pomocí aplikace **draw.io** [9].

2.7 Webová aplikace

Webová aplikace na titulní stránce zobrazuje stručný popis fungování aplikace. Nabízí odkaz pro stažení aplikace a odkaz do Google Play (případně i Apple Store).

Webová aplikace umožňuje správcům přihlásit se a prohlížet si a upravovat dotazníkové otázky a ostatní data pro mobilní aplikaci. Aplikace také správcům nabídne automatickou kontrolu kvality dotazníku a ulehčí tak identifikaci možných chyb a opomenutí při zadávání.

Správci si pomocí webové aplikace také mohou prohlížet statistická data o použití mobilní aplikace.

2.7.1 Případy užití webové aplikace

Diagram případů užití webové aplikace zachycuje obrázek 2.3.

UC 7. Zobrazení otázky

Role: správce

1. Uživatel klikne na záložku *Otázky*
2. Systém zobrazí otázky ve formě stromu a náhled dotazníku (obrazovka *Přehled otázek*)

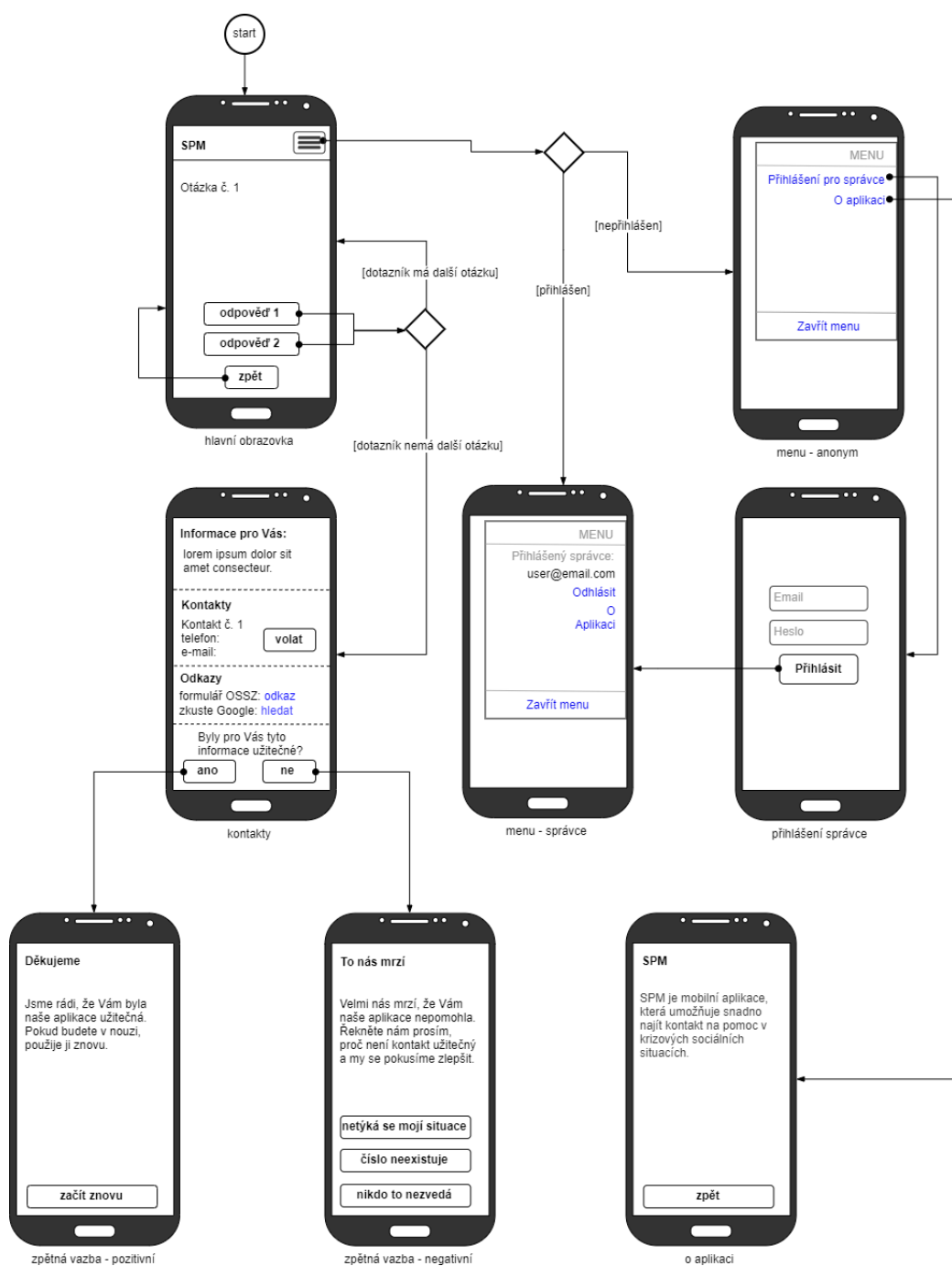
UC 8. Kontrola otázek

Role: správce

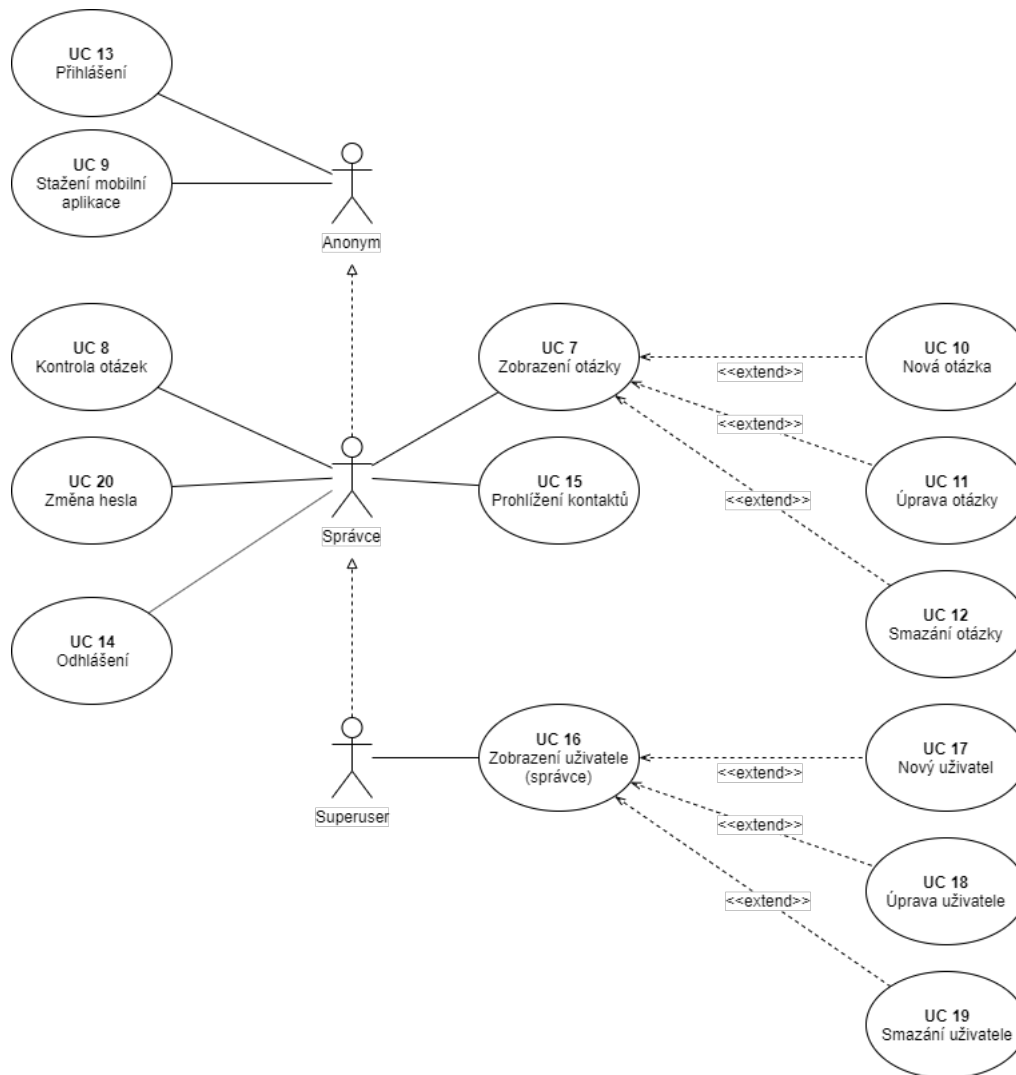
1. Uživatel klikne na záložku *Kontrola*
2. Systém zobrazí otázky, ve kterých byl nalezen problém ve formě seznamu s filtrem podle typu problému

UC 9. Stažení mobilní aplikace

Role: anonym



Obrázek 2.2: Navigace a návrh obrazovek pro mobilní aplikaci. (Příloha 3)



Obrázek 2.3: Případy užití webové aplikace. (Příloha 10)

1. System zobrazí úvodní stránku
2. Uživatel klikne na odkaz ke stažení
3. System předá prohlížeči soubor .apk (balíček pro Android)

UC 10. Nová otázka

Role: správce

Startovní podmínky: správce je přihlášen a je v editaci otázek

1. Uživatel klikne na tlačítko *Nová*
2. System zobrazí formulář *Editace otázky*

UC 14. Odhlášení

Role: správce

Startovní podmínky: správce je přihlášen

1. Uživatel klikne na tlačítko *Odhlásit*
2. Systém odhlásí uživatele

UC 15. Prohlížení kontaktů

Role: správce

Startovní podmínky: správce je přihlášen

1. Uživatel klikne na tlačítko *Odhlásit*
2. Systém odhlásí uživatele

UC 16. Zobrazení uživatele (správce)

Role: superuser

Startovní podmínky: superuser je přihlášen

1. Superuser klikne na položku menu *Administrace uživatel*
2. Systém zobrazí seznam uživatelů.

UC 17. Nový uživatel

Role: superuser

Startovní podmínky: superuser je přihlášen a je v administraci uživatelů

1. Superuser klikne na tlačítko *Nový*
2. Systém zobrazí formulář *Editace uživatele*
3. Superuser vloží email, jméno a ostatní údaje uživatele
4. Superuser klikne na tlačítko *Uložit*
5. Systém uloží uživatele

UC 18. Úprava uživatele

Role: superuser

Startovní podmínky: superuser je přihlášen a je v administraci uživatelů

1. Superuser vybere uživatele ze seznamu
2. Superuser klikne na tlačítko *Upravit*
3. Systém zobrazí formulář *Editace uživatele*

4. Superuser upraví údaje uživatele
5. Superuser klikne na tlačítko *Uložit*
6. Systém uloží uživatele

UC 19. Smazání uživatele

Role: superuser

Startovní podmínky: superuser je přihlášen a je v administraci uživatelů

1. Superuser vybere uživatele ze seznamu
2. Superuser klikne na tlačítko *Smazat*
3. Systém zobrazí otázku *Opravdu si přejete smazat tohoto uživatele?*
4. IF [superuser klikne na tlačítko *Ano*] THEN Systém smaže uživatele
ELSE FINISH

UC 20. Změna hesla

Role: správce

Startovní podmínky: správce je přihlášen

1. Uživatel klikne na položku menu *Profil*
2. Systém zobrazí formulář *Editace uživatele*
3. Uživatel klikne na tlačítko *Změnit heslo*
4. Systém zobrazí formulář *Změna hesla*
5. Uživatel vloží stávající heslo, nové heslo a potvrzení nového hesla
6. Uživatel klikne na tlačítko *Uložit*
7. Systém uloží nové heslo uživatele

2.7.3 Návrhy obrazovek

Pro webovou aplikaci jsem vytvořil návrhy obrazovek jednotlivě. K jejich vytvoření jsem použil nástroj **draw.io** [9].

Úvodní stránka

Úvodní stránka (wireframe 2.5) ve stručnosti představí aplikaci a nabídne odkaz na stažení. Také zobrazí kontakt na autory aplikace a odkaz pro vstup do správcovské části.



Obrázek 2.5: Úvodní stránka. (Příloha 11)

Přihlášení správce

Zde se mohou správci přihlásit a vstoupit do administrace.

Hlavní stránka administrace

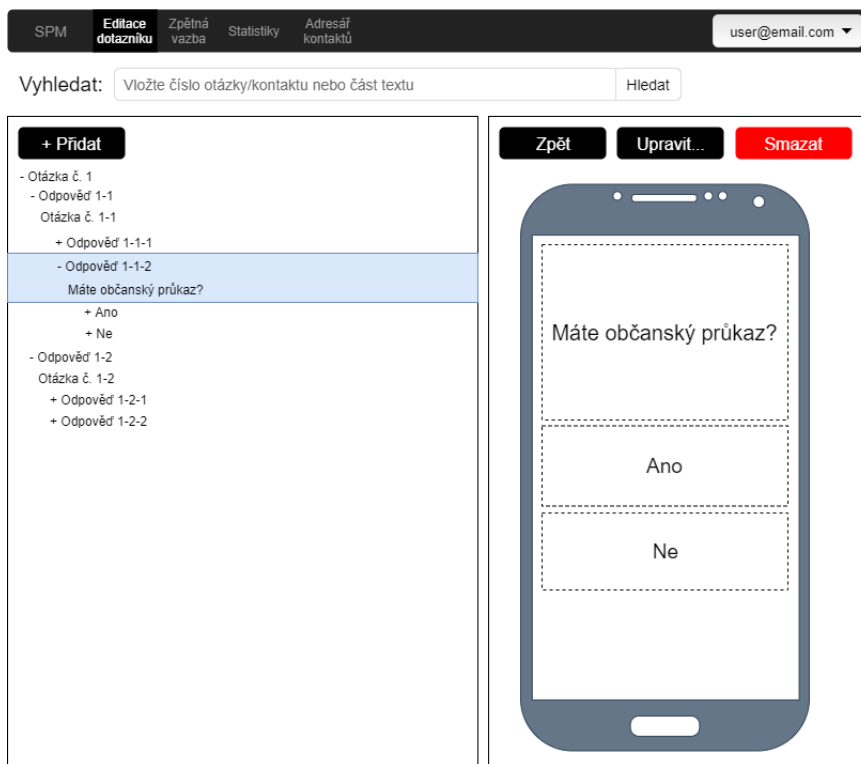
Rozcestník pro správce, odkud se mohou dostat do všech částí administrace.

Přehled otázek

Zde správci vkládají a upravují strukturu a znění dotazníkových otázek a kontaktů (wireframe 2.6).

Otázky jsou zobrazeny ve stromové struktuře, ve které lze vyhledávat podle čísla otázky nebo části textu (tzv. *fulltextové vyhledávání*).

K dispozici je rovněž náhled otázky tak, jak bude vypadat v mobilní aplikaci. V náhledu lze kliknout na odpověď a dostat se tak na další otázku, případně kontakt.



Obrázek 2.6: Přehled otázek. (Příloha 8)

Po kliknutí na tlačítko *Upravit...* vyskočí formulář pro editaci otázky (nebo kontaktu):

■ Editace otázky

V tomto formuláři lze nastavit, jaká otázka předchází a jaká odpověď vede k této otázce (2.7).

■ Editace kontaktu

Po zaškrtnutí políčka *Kontakt* se z otázky stane stránka s kontakty (wireframe 2.8). To znamená, že uživatel již nedostane další otázku, ale místo toho budou zobrazeny kontakty z katalogu a užitečné odkazy.

Správce musí ručně vybrat, která část katalogu je pro tuto stránku kontaktů relevantní. Může vybrat více částí z katalogu nebo nemusí vybrat žádnou.

Obrázek 2.7: Editace otázky. (Příloha 7)

2.8 Sběr kontaktů - scraper

Součástí systému bude tzv. *scraper* - program provádějící automatický sběr kontaktů z veřejných zdrojů na internetu (*Web scraping* [37]). Jeho úlohou bude pravidelně aktualizovat kontaktní informace, které jsou na konci dotazníku zobrazeny uživatelům. Toto je nutné pro naplnění kvalitativního požadavku **QR 2**.

Ideální by samozřejmě bylo získávat data kontaktů v nějaké strojově čitelné formě, například dnes populární **open data** [36], ale žádná vhodná k našemu účelu bohužel v ČR neexistují, a tak musíme data sbírat z webových stránek institucí, které data zveřejňují.

Sbírat budeme kontakty z následujících zdrojů:

- Registr sociálních služeb - azylové domy, sociální poradenství, ad. (<http://iregistr.mpsv.cz>)
- Úřady práce - všechny úřady práce v ČR (<https://www.uradprace.cz>)

Obrázek 2.8: Editace kontaktu. (Příloha 6)

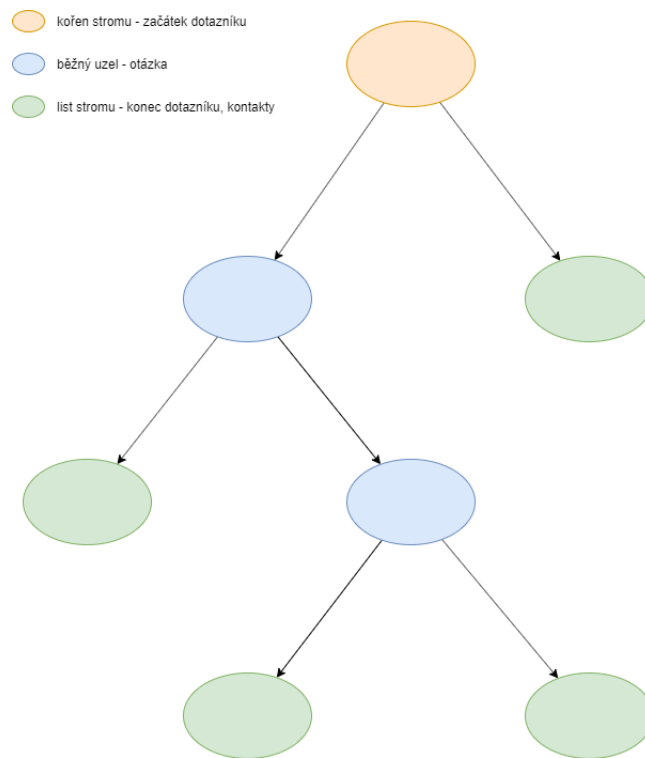
- Obecní úřady - pověřené obecní úřady, obce s rozšířenou působností (<https://mesta.obce.cz>)
- ČSSZ - Česká správa sociálního zabezpečení (<https://www.cssz.cz>)

Více o problémech se sběrem kontaktů pomocí scraperu v kapitole 4.5.1.

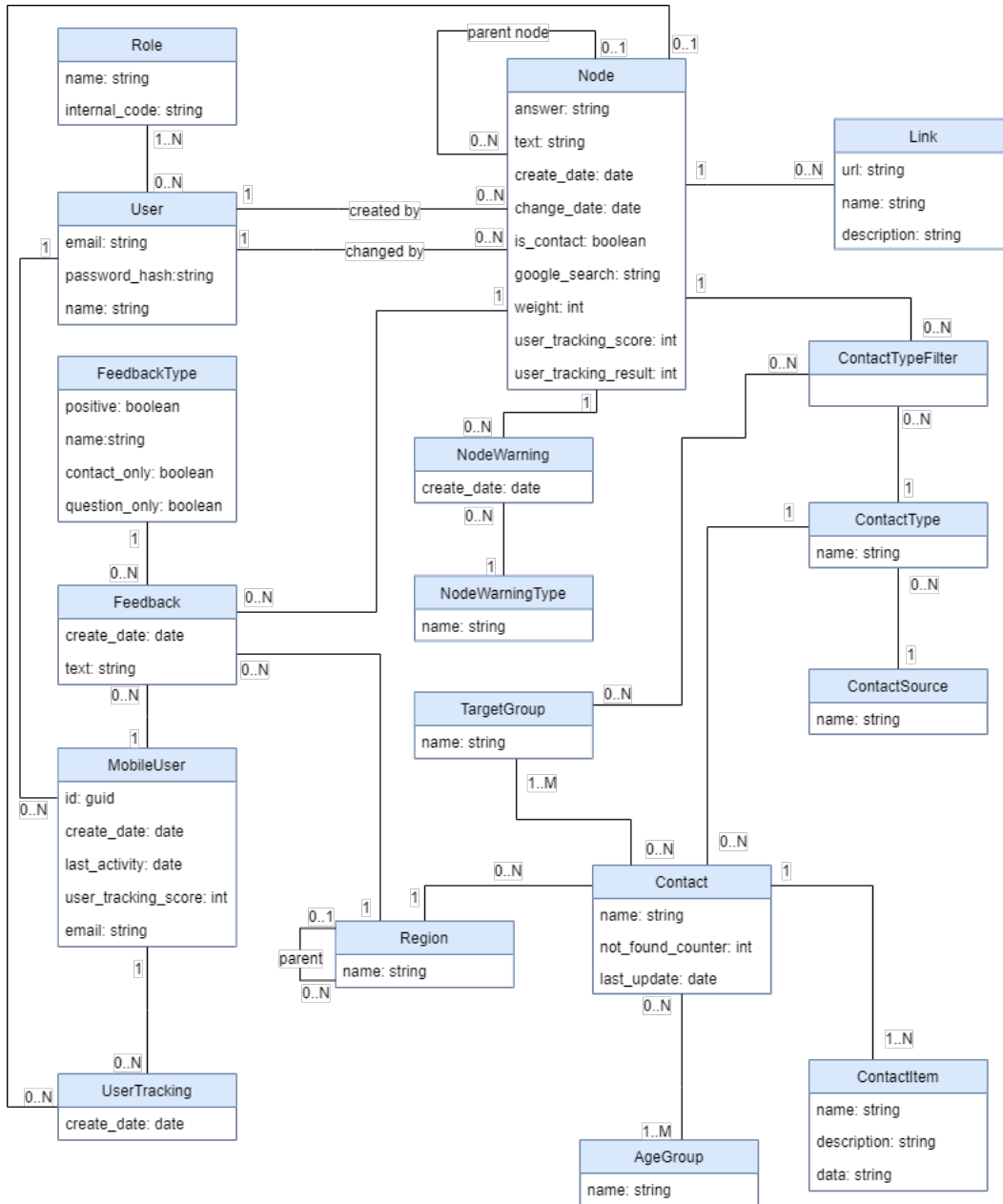
2.9 Datový model

Otázky a kontakty dotazníku jsou uspořádané do struktury stromu, kde uzel reprezentuje jednu otázku nebo stránku s kontakty a hrana reprezentuje odpověď vedoucí k další otázce. (obrázek 2.9).

Datový model je zachycen na diagramu 2.10



Obrázek 2.9: Reprezentace struktury dotazníku. (Příloha 4)



Obrázek 2.10: Datový model. (Příloha 1)

Kapitola 3

Návrh systému

3.1 Architektura systému

Pro implementaci jsem zvolil tzv. **3-Tier** architekturu, která je rozšířením **Client-Server** architektury [32].

Systém se skládá z těchto částí:

- databáze
- aplikační server
- mobilní aplikace
- webový server
- webová aplikace

Vztahy mezi jednotlivými částmi systému zachycuje diagram 3.1.

3.1.1 Databáze

Pro implementaci databázového uložení jsem použil **PostgreSQL** ve verzi 10.12 [16].

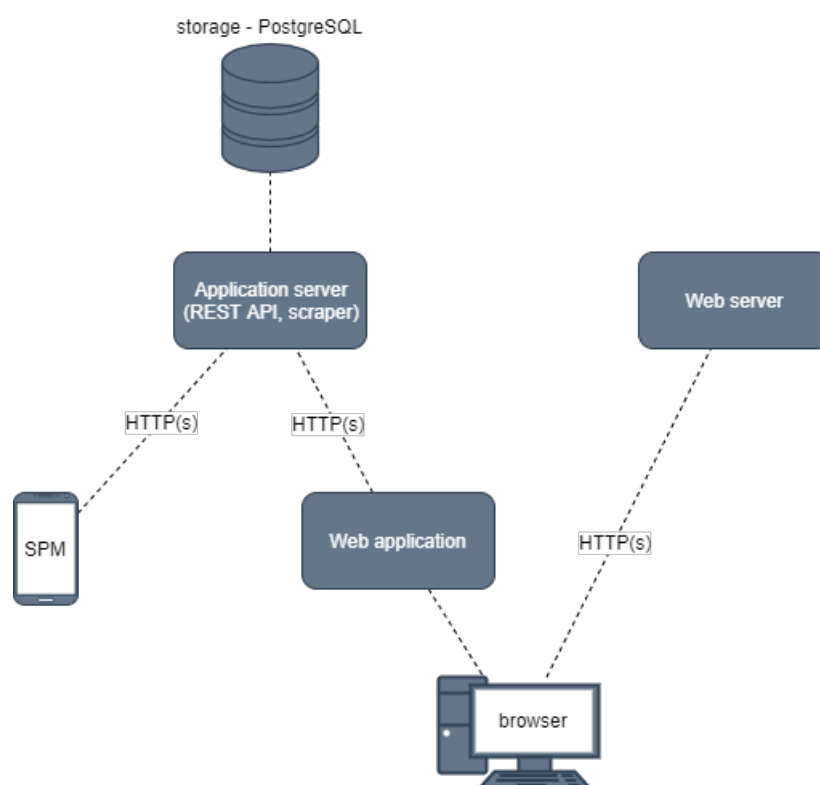
Datový model je zachycen na diagramu 2.10.

3.1.2 Mobilní aplikace

Aplikace pro chytré telefony je naprogramovaná v jazyce **TypeScript** verze 3.4 [25] s pomocí frameworku **React Native** ve verzi SDK 0.36 [21]. Tuto technologii jsem zvolil, protože je multiplatformní a umožní snadný přenos aplikace na jiné mobilní operační systémy, zejména iOS, což je výslovný požadavek.

Pro build používám platformu **Expo** [1].

Aplikace se distribuuje přes obchod **Google Play** [24] a pomocí webových stránek jako balíček **.apk**.



Obrázek 3.1: Součásti systému. (Příloha 5)

■ 3.1.3 Aplikační server

Aplikační server je naprogramovaný v jazyce **Java** verze 1.8 [27] s pomocí frameworku **Spring** [35].

Pro práci s databází jsem použil ORM framework **Hibernate** [17] implementující **Java Persistence API** [26].

Pro build používám nástroj **Maven** [11].

■ REST API

Server bude poskytovat REST API, které bude zdokumentované pomocí nástroje **Swagger** [30], které odpovídá **OpenAPI Specification** ve verzi 2.0 [29].

■ Scraper - sběrač kontaktů

Součástí serveru bude *scraper* pro sběr kontaktních informací. Protože data sbíráme z několika zdrojů a každý z nich je jiný, scraper musí implementovat pro každý z nich separátní logiku - extrahovat potřebná data a transformovat je do podoby vhodné pro uložení do databáze.

Pro sběr z veřejných webových stránek použijeme knihovnu **org.jsoup** [18]. Některé kontakty (např. na sociální kurátory) se budou importovat z Excelu

(formát `.xlsx`) a k tomu použijeme knihovnu **Apache POI** [12]. Import těchto dat bude implementován jako jeden ze zdrojů uvnitř scraperu.

Scraper se bude spouštět automaticky jednou týdně. Periodu spouštění lze snadno změnit v konfiguraci.

O problémech se sběrem kontaktů pomocí scraperu píší v kapitole 4.5.1.

■ 3.1.4 Webový server

Jedinou úlohou web serveru je doručit klientu webovou aplikaci zabalenou jako `.js` soubor. Skládá se z jediné HTML stránky, která uživateli načte webovou aplikaci.

Pro implementaci webového serveru použijeme framework **Thymeleaf** ve verzi 2.1 [33].

■ 3.1.5 Webová aplikace

Webová aplikace bude napsaná v jazyce **TypeScript** verze 3.4 [25] s pomocí frameworku **React** verze 16.9 [20]. Jazyk **TypeScript** jsem zvolil, protože stejný je použit pro vývoj mobilní aplikace. Framework **React** jsem zvolil, protože je příbuzný s frameworkem **React Native** použitým pro vývoj mobilní aplikace. Množství jazyků a technologií v rámci celého projektu je díky tomu menší a části kódu lze mezi mobilní a webovou aplikací sdílet.

Pro sestavení aplikace použijí nástroj **webpack** verze 4.19 [8].

■ 3.2 Testování

■ 3.2.1 Mobilní aplikace

Pro jednotkové testování (*unit tests* [32]) kódu mobilní aplikace použijí framework **jest** [19] ve verzi **26.0**.

Mobilní aplikace jako celek se bude před nasazením nové verze testovat ručně metodou tzv. *black box*) testování [32] pomocí definovaných testovacích scénářů, které pokryjí hlavní případy užití.

Z dlouhodobého hlediska by bylo mnohem lepší testovat mobilní aplikaci metodou *end-to-end*. Zatím na to nebyly časové možnosti, ale pokud bude projekt dále pokračovat, měly by *end-to-end* testy být implementovány.

■ 3.2.2 Aplikační server

Pro jednotkové testování (*unit tests* [32]) kódu jádra aplikačního serveru použijí framework **junit** [23] ve verzi 4.12.

Z dlouhodobého hlediska by bylo mnohem lepší testovat aplikační server ještě pomocí nástroje vhodného pro testování REST API, např. **SoapUI** [31]. Zatím na to nebyly časové možnosti, ale pokud bude projekt dále pokračovat, měly by testy REST API být implementovány.

■ 3.2.3 Webová aplikace

Pro jednotkové testování (*unit tests* [32]) kódu webové aplikace použijí framework **jest** [19] ve verzi 26.0.

Z dlouhodobého hlediska by bylo mnohem lepší testovat webovou aplikaci metodou *end-to-end*. Zatím na to nebyly časové možnosti, ale pokud bude projekt dále pokračovat, měly by *end-to-end* testy být implementovány.

■ 3.3 Sestavení a nasazení

Sestavení a nasazení se provádí pomocí připravených skriptů. Není tedy třeba dělat větší množství úkonů manuálně.

Z dlouhodobého hlediska by bylo mnohem lepší používat metody *continuous integration* a *continuous delivery* [7]. Tedy mít připravenou automatizovanou *pipeline*, která by zahrnovala sestavení, automatické testování a nasazení. Zatím na to však nebyly časové možnosti.

■ 3.3.1 Aplikační server

Aplikační server se příkazem **npm build** sestaví do podoby **.jar** balíčku, který v sobě obsahuje webový server **Tomcat 7** [35, 13] a pro nasazení jej stačí nahrát do prostředí serveru s **Java 1.8** [27] a spustit.

Druhou (nyní nevyužívanou) možností je sestavit aplikační server do podoby **.war** balíčku, který je určen k nahrání do prostředí s běžícím serverem **Tomcat** [13].

■ 3.3.2 Mobilní aplikace

Pro build se používá platforma **Expo** [1]. Mobilní aplikaci je nutné sestavit dvakrát - jednou pro distribuci přes obchod **Google Play** [24] a jednou pro distribuci na webových stránkách projektu.

Příkazem **expo build:android** se sestaví balíček **.apk**, který se posléze zkopíruje na webový server.

Příkazem **expo build:android -t app-bundle** se sestaví balíček, který se posléze nahraje do obchodu **Google Play** [24]. Použití **App bundle** namísto **.apk** vede ke zmenšení velikosti aplikace - Google Play jej dokáže optimalizovat.

■ 3.4 Bezpečnost

■ 3.4.1 Zabezpečení proti ztrátě dat

Databáze je pravidelně zálohována pomocí příkazu **pg_dump** [16]. Výsledný soubor je dále zálohován na jiný fyzický stroj, aby nedošlo ke ztrátě dat. Výsledný soubor je na serveru ponechán po dobu pěti dnů a poté smazán. Princip

zálohování je možné vidět ve skriptu **script/backup** ve složce zdrojových kódů.

Obnovu databáze je možné provést pomocí příkazu **pg_restore** [16]. Princip obnovení je možné vidět ve skriptu **script/restore** ve složce zdrojových kódů.

■ 3.4.2 Zabezpečení proti neoprávněné změně nebo odcizení dat

Aby nemohlo dojít k neoprávněnému přístupu k datům, je přístup do administrační části systému chráněn uživatelským jménem a heslem. Pro ověřování identity a autorizace jsem použil balíček **spring-boot-starter-security** [34] z frameworku **Spring** [34] [35]. Tento balíček využívá tzv. **JWT** (*JSON Web Token*) [6], což je otevřený průmyslový standard označený jako **RFC 7519**.

Veškerá komunikace mezi serverem a klientem probíhá šifrovaně pomocí protokolu **https**. Použitý protokol šifrování je **TLS 1.3** s veřejným RSA klíčem o délce 2048 bitů. Jako certifikační autoritu jsem využil **Let's Encrypt** [15], protože je zdarma. Pro získání certifikátu a nastavení serveru jsem využil nástroj **Certbot** [10], který automatizuje získání a pravidelnou obnovu certifikátů.

Kapitola 4

Realizace

4.1 Tým

Na realizaci projektu pracoval tým o čtyřech lidech. Já jsem byl v úloze vývojáře a kompletně jsem navrhnul a implementoval celý systém. Dále na projektu pracoval odborník v oblasti sociálních služeb Jakub Marek, který se dvěma dalšími pomocníky vymyslel a vložil obsah otázek a odpovědí.

4.2 Vývoj

Vývoj aplikace začal již v říjnu 2019, největší část byla naprogramována během prvních tří měsíců, ale vývoj probíhá s přestávkami až do současnosti. Programování aplikace jsem dělal kompletně sám v integrovaném prostředí (IDE) **IntelliJ IDEA** [22] (pro jazyk Java) a **Atom** [14] (pro Javascript, React, HTML, CSS a ostatní).

Zdrojové kódy jsem verzoval pomocí nástroje **GIT** [28] umístěném na vlastním serveru. Protože jsem na vývoji pracoval sám, používal jsem jednoduché centralizované *workflow*. Vytvářel jsem poctivě *release branches* pokaždé, ale jen vyjíměčně *feature branches* [5].

4.3 Finance

Projekt se dostal do inkubačního akceleračního programu **Laboratoř Nadace Vodafone** v rámci **Operačního programu Zaměstnanost** Evropské Unie a obdržel finanční podporu ve formě grantu. Mobilní i webová aplikace proto nesou loga těchto dvou organizací.

Práce na implementaci systému byly v rozsahu zhruba **250** hodin a jen část z nich byla za finanční odměnu. Velká část prací byla vykonána zdarma jako příspěvek ušlechtilé myšlenky projektu.

Využití peněz z grantu je zachyceno v tabulce 4.1.

Práce na vývoji software	100 000
Odborná práce na obsahu	80 000
CELKEM	180 000

Tabulka 4.1: Využití peněz z grantu (v Kč).

4.4 Marketing

Propagaci aplikace jsme zahájili prezentací na festivalu **Noc venku**, který se věnuje problematice lidí bez domova a mezi návštěvníky je řada sociálních pracovníků, kteří jsou naší cílovou skupinou.

Další naší marketingovou akcí bylo rozeslání emailů s informacemi o aplikaci a prosbou o její vyzkoušení a šíření. Jelikož součástí našeho systému je databáze kontaktů na neziskové organizace věnující se pomoci sociálně znevýhodněným, použili jsme kontakty z této databáze. Díky tomuto hromadnému emailu rozeslanému na zhruba **6 000** správně vybraných adres jsme získali několik stovek uživatelů během jediného týdne.

Součástí hromadného emailu byl také informační plakát cílený na osoby v sociální nouzi, o jehož vytisknutí a vylepení jsme zaměstnance oslovených neziskových organizací požádali. O počtu takto vylepených plakátů bohužel nemáme informace.

4.5 Problémy

Při realizaci projektu jsem narazil na několik problémů.

4.5.1 Scraping

Původním záměrem bylo ke sběru dat o kontaktech použít *open data*. Vhodný katalog kontaktů na poskytovatele sociálních služeb ovšem v ČR neexistuje [36], a tak samotná existence scraperu je vlastně již řešením problému.

Při implementaci scraperu, který je blíže popsán v kapitolách 2.8 a 3.1.3, se objevilo hned několik překážek.

Některé stránky, přestože zveřejněné informace jsou veřejné a dokonce musí takové být ze zákona [38], aktivně brání scrapingu tím, že detekují automatizovaný sběr dat a po vyčerpání určitého počtu načtení, přestanou zobrazovat požadované informace. Toto jsem bohužel zjistil až po implementaci aplikační logiky scraperu pro dotčený zdroj dat a jeho spuštění, takže bylo nutné najít jiný zdroj a implementovat novou logiku scrapingu.

V případě kontaktů na *CDZ* a sociální kurátory se nakonec vůbec nepodařilo nalézt veřejný zdroj dat. Data jsou nakonec importována ze souboru, který nám poskytli přátelé neziskové organizace.

Druhým problémem je, že každá změna v HTML kódu stránky, ze které scraper sbírá data, vede k rozbití funkce scraperu. Jedna ze stránek, kterou používáme pro sběr dat, prodělala krátce po spuštění projektu redesign, a tak bylo potřeba v tomto případě logiku scraperu naprogramovat znovu.

■ 4.5.2 Geolokace

Několik výzev se také vynořilo v souvislosti s geolokací.

GPS v mobilním telefonu vrací jen prosté souřadnice polohy, naše kontakty jsou ale uspořádány podle kraje a okresu a je tedy potřeba souřadnice převést na okres v ČR. Souřadnice je třeba nejprve převést na PSČ pomocí tzv. *reverzního geokódování* (*reverse geocoding*) [2]. Poté je třeba na základě získaného PSČ získat název okresu. K tomu jsem použil veřejné API České pošty zveřejněné přes platformu **ApiTalks** [3].

Dále při sběru dat o kontaktech některé zdroje dat neobsahují jasně uvedenou informaci o okresu a je třeba okres každému kontaktu přiřadit na základě názvu obce. K tomu jsem použil veřejné API katalogu **RÚIAN**, který zveřejňuje **ČÚZK** [39].

Zobrazování kontaktů na základě okresu, kde se uživatel nachází, bohužel není ideální. Někdy se nejbližší vhodná organizace nachází ve vedlejší obci, která ale může ležet v jiném okrese nebo dokonce kraji. V takovém případě bohužel naše aplikace zobrazí kontakty z aktuálního okresu, které ale nejsou těmi nejvhodnějšími.

Bylo by lepší řešit geolokaci nějak přímočařeji a jednodušeji - ignorovat kraje a okresy a namísto toho zobrazovat geograficky nejbližší kontakty s využitím nějakých mapových služeb. Na druhou stranu ale například u ČSSZ nebo Úřadů práce je nutné okresy zohledňovat, protože obsluhují vždy pouze obyvatele svého příslušného okresu.

Systém geolokace plánujeme v budoucnu znovu promyslet a předělat.

■ 4.5.3 Vkládání obsahu

Řada problémů se vynořila při vkládání obsahu. Ukázalo se, že je řada otázek stylistického rázu, jako například, zda při vykání psát zájmeno *Vy* s velkým písmenem, jaký formát používat pro psaní finančních částek, jak členit text do odstavců a podobně. Tyto drobnosti nemají samy o sobě zásadní důležitost, ale při vkládání obsahu docházelo k tomu, že každý ze správců obsahu zvolil poněkud jiný přístup a výsledek působil velice nekonzistentně. Byla velká chyba, že nebyl předem vytvořen stylistický manuál, který by výše zmíněné otázky zodpověděl a styl textů by sjednotil.

Dále se ukázalo, že odborníci na sociální práci nejsou nutně odborníky na český pravopis, a vynořila se potřeba automatické nebo manuální kontroly pravopisu, na kterou bohužel nezbýval čas. Bylo chybou, že členem týmu nebyl schopný češtinář, který by texty korigoval, případně že nebyla implementována automatická kontrola pravopisu.

Největší výzvou se ukázalo samotné formulování otázek a odpovědí. Přestože na obsahu pracovali odborníci na základní sociální poradenství, forma mobilní aplikace je až příliš vzdálena jejich běžnému způsobu vedení rozhovorů s klienty, a tak mnoho textů v aplikaci trpí nízkou srozumitelností, používáním příliš odborných výrazů, nebo špatnou strukturou.

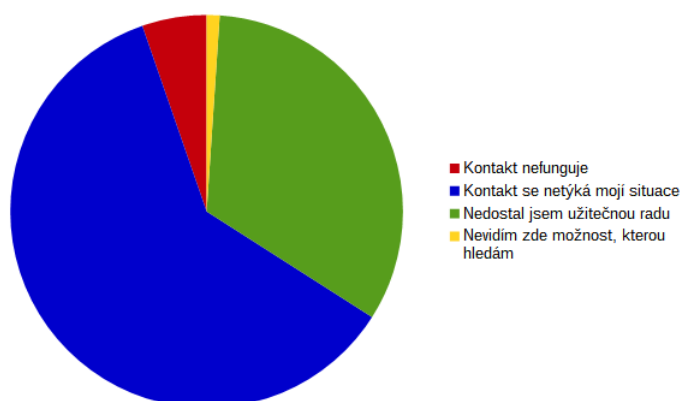
4.6 Zpětná vazba

Na naši aplikaci jsme od okamžiku spuštění dostávali od uživatelů zpětnou vazbu. Nejprve prostřednictvím emailu, později jsem narychlo improvizovaně přidal do mobilní aplikace formulář, který umožňuje uživatelům snadno odeslat zpětnou vazbu. Uživatelé mohou zvolit, zda byli s aplikací spokojeni, mohou zvolit důvod nespokojenosti a také mohou vložit libovolný text jako zprávu pro nás.

Odeslaná zpětná vazba dále obsahuje informaci o tom, kde se uživatel nachází geograficky a také kterou otázku v dotazníku právě prohlíží. Z těchto dat čerpáme informace o tom, co našim uživatelům nejvíc vadí, které části dotazníku jsou nejproblematictější a ve kterých okresech je nejnižší kvalita dat s kontakty.

Na naši aplikaci nám pomocí kontaktního formuláře poskytlo zpětnou vazbu zhruba 300 našich uživatelů. Dvě třetiny se vyjádřili pozitivně, zbylá jedna třetina uživatelů nás upozornila na nedostatky obsahu, zejména na menší srozumitelnost otázek nebo chybějící druh pomoci.

Obrázek 4.1 znázorňuje rozdělení negativních zpětných vazeb podle typu problému.



Obrázek 4.1: Rozdělení negativních zpětných vazeb podle typu problému.

Zpětnou vazbu průběžně vyhodnocujeme a vedle toho, že zpětnou vazbu používáme k identifikaci problémů, ze zpráv od uživatelů také čerpáme inspiraci pro další rozšíření aplikace.

Kapitola 5

Budoucnost projektu

Do budoucna máme s projektem řadu plánů technického i organizačního charakteru. Aplikace funguje dobře a zadání se podařilo naplnit, nicméně ze zpětné vazby od uživatelů a odborníků v sociální oblasti vzešla řada nápadů na další rozšíření. Chtěli bychom například přidat verzi pro telefony značky Apple, upravit aplikaci pro nevidomé, přidat offline režim, propojit vyhledávání kontaktů s mapami a navigací a další věci.

V obsahové části máme náměty na další oblasti sociálních služeb, které bychom mohli pokrýt. Například *vězni a propuštění*, *týrané osoby*, *náhradní rodinná péče*, *integrace cizinců* a další. Máme také nápady na celé nové služby, které bychom mohli nabídnout klientům i pracovníkům v sociálních službách jako například možnost procházet databází kontaktů přímo nebo přidat bazar pro nabídky a poptávky věcí nebo služeb.

Projekt zatím fungoval na soukromé bázi, chtěli bychom nyní založit právnickou osobu - neziskovou organizaci, která by celý projekt zastřešovala. Do budoucna bychom rádi spolupracovali s dalšími neziskovými organizacemi v oblasti sociálních služeb a to zejména při tvorbě a kontrole obsahu. Naší vizí je vytvořit svobodnou platformu, která bude plně pokrývat problematiku základního sociálního poradenství, bude propojovat svět informačních technologií se světem sociální práce, bude fungovat na demokratickém komunitním principu, nebude závislá na státu ani na jedné konkrétní organizaci, ale naopak bude vycházet z ideálů občanské společnosti a dobrých úmyslů.

Ukázalo se zkrátka, že nápad má velký potenciál a stávající podoba aplikace je jen zlomkem toho, čím by jednou mohla být.

Kapitola 6

Závěr

Cíle projektu se podařilo naplnit. Byl vytvořen a do ostrého provozu nasazen systém, který plně uspokojuje zadání - vytvořit mobilní aplikaci, která umožní snadno najít kontakt na pomoc v krizových sociálních situacích.

Po analýze problému spolu s odborníkem na sociální práci jsem sám navrhl a implementoval technické řešení v podobě systému, který se skládá z aplikačního serveru, mobilní aplikace a webové aplikace. Mobilní aplikace nabídne uživateli řadu jednoduchých otázek a na základě zvolených odpovědí poté nabídne návod na další postup a kontakty na relevantní pomoc. Formulace otázek a odpovědí do aplikace vkládají správci - odborníci na sociální problematiku - přes webovou aplikaci. Definují také, jaké instrukce a jaké typy kontaktních informací se mají po zodpovězení dotazníku zobrazit. Kontaktní údaje aplikace sbírá a průběžně aktualizuje z předem definovaných zdrojů.

Projekt se dostal do inkubačního akceleračního programu Laboratoř Nadace Vodafone a obdržel finanční podporu ve formě grantu. Mobilní i webová aplikace proto nesou loga těchto dvou organizací.

Vývoj aplikace začal již v říjnu 2019, největší část byla naprogramována během prvních tří měsíců, ale vývoj probíhá s přestávkami až do současnosti.

Během vývoje jsem narazil na několik problémů technického a organizačního charakteru, které se podařilo úspěšně překonat. Největší problémy se objevily v souvislosti se sběrem kontaktních údajů, protože neexistuje jeden dobře strojově zpracovatelný zdroj, který by agregoval všechny potřebné kontakty. Data jsou fragmentovaná a často nejsou vhodná pro automatické zpracování. Další problémy se objevily v souvislosti s geolokací, která je nutná pro zobrazování kontaktů na nejbližší vhodnou pomoc. Musel jsem se také zabývat problémy spojenými s vkládáním obsahu. Všechny zmíněné problémy se mi podařilo odstranit nebo minimalizovat jejich dopady.

Systém jsme v souladu s podmínkami grantu úspěšně spustili v prosinci 2019 a k dnešnímu dni (Květen 2020) máme přes 1 500 uživatelů, kteří mají aplikaci nainstalovanou ve svém telefonu. Každý den aplikaci v průměru použije 30 různých uživatelů, z toho asi 10 nových uživatelů si každý den naši aplikaci nainstaluje. Aplikaci si lze stáhnout v obchodě Google Play, nebo na našich stránkách www.socpracvmobilu.cz.

V dalším rozvoji projektu budeme dále pokračovat a systém vylepšovat po technické i obsahové stránce.

Příloha A

Seznam použitých zkratk

- **SPM**: Označuje celý projekt *Sociální pracovník v mobilu*.
- **JPA**: Java Persistence API [26].
- **CDZ**: Centra duševního zdraví. Síť zařízení pro péči o lidi s duševním onemocněním a jeden z pilířů reformy psychiatrické péče v ČR.
- **GPS**: Global Positioning System.
- **PSČ**: Poštovní směrovací číslo.
- **ČÚZK**: Český úřad zeměměřický a katastrální.
- **RÚIAN**: Registr územní identifikace, adres a nemovitostí.
- **ČSSZ**: Česká správa sociálního zabezpečení.
- **JWT**: JSON Web Token. Open standard RFC 7519.
- **TLS**: Transport Layer Protocol. Open standard RFC 8446.



Příloha B

Literatura

- [1] Inc. 650 Industries. *Expo Documentation*. 2020.
- [2] Inc. 650 Industries. *ExpoLocation Documentation*. 2020.
- [3] ApiTalks. *Data České pošty*. 2020.
- [4] Atlassian. *Atlassian Agile Coach*. Atlassian, 2020.
- [5] Atlassian. *Comparing Workflows*. Atlassian, 2020.
- [6] Auth0. *Introduction to JSON Web Tokens*. 2020.
- [7] Codeship. *Continuous Integration Essentials*. 2020.
- [8] Webpack Community. *Webpack Documentation*. 2020.
- [9] Diagrams.Net. *draw.io Documentation*. 2020.
- [10] Electronic Frontier Foundation. *Certbot Documentation*. 2020.
- [11] The Apache Software Foundation. *Maven Documentation*. 2019.
- [12] The Apache Software Foundation. *Apache POI - the Java API for Microsoft Documents*. 2020.
- [13] The Apache Software Foundation. *Apache Tomcat Documentation*. 2020.
- [14] Inc. GitHub. *Atom Documentation*. 2020.
- [15] Internet Security Research Group. *Let's Encrypt Documentation*. 2020.
- [16] The PostgreSQL Global Development Group. *PostgreSQL Documentation*. 2020.
- [17] Red Hat. *Hibernate Documentation*. 2020.
- [18] Jonathan Hedley. *org.jsoup Documentation*. 2020.
- [19] Facebook Inc. *Jest Documentation*. 2020.
- [20] Facebook Inc. *React Documentation*. 2020.

Příloha C

Externí přílohy

C.1 PNG pro draw.io

Tyto PNG soubory lze editovat v nástroji **draw.io** [9], protože obsahují navíc metadata pro tento nástroj.

EA 1. data-model.png: Datový model.

EA 2. mobile-use-cases.png: Případy užití mobilní aplikace.

EA 3. mobile-wireframes.png: Návrh obrazovek pro mobilní aplikaci.

EA 4. question-tree.png: Struktura uspořádní otázek v dotazníku.

EA 5. system-overview.png: Architektura systému a vztahy mezi jednotlivými částmi.

EA 6. web-edit-contact.png: Návrh obrazovky pro editaci kontaktu.

EA 7. web-edit-question.png: Návrh obrazovky pro editaci otázky.

EA 8. web-edit-questionnaire.png: Návrh obrazovky pro editaci dotazníku.

EA 9. web-navigation.png: Návrh navigace pro webovou aplikaci.

EA 10. web-use-cases.png: Případy užití webové aplikace.

EA 11. web-wireframe-1.png: Návrh úvodní stránky webové aplikace.

C.2 Zdrojové kódy

Veškeré zdrojové kódy včetně dokumentace a textu této bakalářské práce.

EA 12. spm-src.zip: Kompletní zdrojové kódy.

I. OSOBNÍ A STUDIJNÍ ÚDAJE

Příjmení: **Zavadil** Jméno: **Karel** Osobní číslo: **474378**
Fakulta/ústav: **Fakulta elektrotechnická**
Zadávací katedra/ústav: **Katedra počítačů**
Studijní program: **Softwarové inženýrství a technologie**

II. ÚDAJE K BAKALÁŘSKÉ PRÁCI

Název bakalářské práce:

Sociální pracovník v mobilu

Název bakalářské práce anglicky:

Social Worker in Mobile Phone

Pokyny pro vypracování:

Analýzujte, navrhňte, vytvořte, otestujte a nasadte systém umožňující pomocí mobilního zařízení najít kontakt na pomoc v krizových sociálních situacích.

Aplikace je určena pro osoby v sociální krizi, sociální pracovníky, státní úředníky pracující s osobami v sociální krizi, ...

Mobilní část aplikace musí uživatel:

- nabídnout dotazníkové otázky a umožnit volbu odpovědi,
- na konci dotazníku zobrazit relevantní kontakty na pomoc,
- zobrazovat kontakty na pomoc podle aktuální polohy uživatele (GPS).

Webová část aplikace musí umožnit správci vkládat, upravovat a mazat dotazníkové otázky.

Systém musí pravidelně sbírat a aktualizovat data kontaktů.

Seznam doporučené literatury:

[1] SOMMERVILLE, Ian. Software engineering. 9th ed. Boston: Pearson, c2011. ISBN 9780137053469.

Jméno a pracoviště vedoucí(ho) bakalářské práce:

Ing. Martin Komárek, kabinet výuky informatiky FEL

Jméno a pracoviště druhé(ho) vedoucí(ho) nebo konzultanta(ky) bakalářské práce:

Datum zadání bakalářské práce: **14.02.2020**

Termín odevzdání bakalářské práce: **22.05.2020**

Platnost zadání bakalářské práce: **30.09.2021**

Ing. Martin Komárek
podpis vedoucí(ho) práce

podpis vedoucí(ho) ústavu/katedry

prof. Mgr. Petr Páta, Ph.D.
podpis děkana(ky)

III. PŘEVZETÍ ZADÁNÍ

Student bere na vědomí, že je povinen vypracovat bakalářskou práci samostatně, bez cizí pomoci, s výjimkou poskytnutých konzultací. Seznam použité literatury, jiných pramenů a jmen konzultantů je třeba uvést v bakalářské práci.

Datum převzetí zadání

Podpis studenta