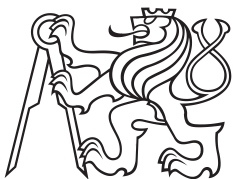


Bakalářská práce



České
vysoké
učení technické
v Praze

F3

Fakulta elektrotechnická
Katedra počítačové grafiky a interakce

Mobilní aplikace pro doporučování dárků

Jan Sládek

Softwarové technologie a management

Květen 2015

Vedoucí práce: Ing. David Sedláček Ph.D.

Poděkování / Prohlášení

Rád bych poděkoval panu Ing. Davidu Sedláčkovi, Ph.D. za vedení této bakalářské práce, cenné rady a zpětnou vazbu v průběhu tvorby. Dále mé rodině, přátelům za podporu a trpělivost, které mi projevili, Bc. Ondřeji Baslerovi za grafický návrh, Bc. Romanu Kozákovi a svému bratrovi, Bc. Tomáši Sládkovi za vývoj webového rozhraní pro komunikaci s databází. V neposlední řadě pak všem lidem, kteří aplikaci testovali a vyjádřili se k ní a panu RNDr. Petru Olšákovi za vytvoření texové šablony CTUStyle.

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 dne 20. 5. 2015

.....

Abstrakt / Abstract

Tato bakalářská práce si klade za cíl přispět k řešení problému výběru vhodného dárku, kterému jsou několikrát ročně vystaveny miliony lidí po celém světě. Mobilní aplikace, která je v rámci této bakalářské práce vyvíjena, narozdíl od většiny konkurenčních aplikací provádí podrobnou analýzu adresáta dárku. Díky tomu může poskytovat doporučení dárků, které jsou skutečně na míru a budou vyhovovat potřebám a přáním adresáta. Mobilní aplikace je založena na platformě Android a poskytuje celou řadu doplňujících funkcí, kterými se snaží maximálně zpříjemnit proces výběru a nákupu dárků.

Klíčová slova: bakalářská práce, Java, Android, doporučování dárků, nápady na dárky

This bachelor project intends to participate in solving the problem of choosing a suitable present, which millions of people have to cope with several times a year. The mobile application developed as a part of this project performs a sophisticated analysis of the gift's recipient, unlike the many competitors in the market. This allows people to get really suitable gift recommendations that match the recipient's needs and desires. The mobile application is based on the Android platform and provides a variety of supplementary functions to make gift selection and buying as pleasant as possible.

Keywords: bachelor thesis, Java, Android, gift recommendation, gift ideas

Title translation: Mobile application for gift recommendation

Obsah /

1 Úvod	1	6 Testování	48
2 Popis problému a cíle	2	6.1 Výběr participantů.....	48
3 Rešerše existujících řešení	4	6.2 Výsledky.....	51
3.1 El Gifto - Gift Ideas Guru.....	4	6.2.1 Přínos testování - nové	
3.2 Gift Generator	5	funkce a změny v bu-	
3.3 Gift Wizard	7	doucích verzích aplikace .	51
3.4 Gift Ideas India - allMemoirs....	8	7 Závěr	52
3.5 Gift Idea Generator	8	7.1 Splnění zadaných úkolů.....	52
4 Analýza aplikace	10	7.2 Další postup.....	53
4.1 Cílová skupina	10	Literatura	54
4.2 Funkční požadavky	10	A Zadání práce	57
4.3 Hlavní funkce	11	B Scénář prototypového testování	
4.4 Doplnkové funkce	13	58
4.4.1 Zjištěná rizika a navr-		C Obsah přiloženého CD	60
hovaná řešení.....	14	D Uživatelská příručka	61
4.5 Kreditový systém	15	E Zkratky a symboly	69
4.6 Obecné požadavky	15	E.1 Zkratky.....	69
4.7 Lokalizace	15		
4.8 Případy užití	16		
4.8.1 Model případů užití.....	16		
4.8.2 Scénáře případů užití	18		
4.9 Marketingový model aplikace .	22		
4.10 Datový model	23		
4.10.1 Lokální SQLite data-			
báze	23		
4.10.2 MySQL databáze na			
serveru.....	26		
4.11 Uživatelské rozhraní	27		
4.11.1 Grafický návrh	27		
4.11.2 Obrazovky.....	28		
5 Implementace	37		
5.1 Použité technologie	37		
5.1.1 Android	37		
5.1.2 Použité externí knihov-			
ny.....	37		
5.1.3 REST.....	38		
5.1.4 JSON	40		
5.1.5 SQLite.....	42		
5.2 Serverová strana	43		
5.3 Google Analytics.....	45		
5.4 Použité algoritmy	45		
5.4.1 Výpočet relevance dár-			
ku.....	45		
5.5 Vývoj aplikace	46		
5.6 Hardwarové a softwarové po-			
žadavky.....	46		

Tabulky /

4.1. Obecné údaje o adresátovi a dárku	11
4.2. Příklady zájmů adresáta	12
5.1. Hardwarové požadavky aplikace	47
5.2. Softwarové požadavky aplikace	47

Kapitola 1

Úvod

V civilizované společnosti existuje celá řada zvyklostí, během kterých si lidé dávají dary. Ať už se jedná o Vánoce, narozeniny, jmeniny či svátek svatého Valentýna, lidé musejí znovu a znovu přemýšlet, co by dotyčného potěšilo. Tráví tak hodiny přemýšlením a procházením obchodních center, zbytečně se stresují a nakonec dárek stejně často skončí v internetových aukcích nebo odložený ve skříni. Čas strávený vybíráním tak přijde navíc a zbude jen hořkost a zkažená nálada. Cílem této práce je vytvořit mobilní aplikaci, která uživateli maximálně zjednoduší proces obdarovávání svých blízkých. Navrhne mu na základě získaných znalostí o adresátovi nejvhodnější dárky a až se bude blížit daná příležitost, aplikace sama uživateli připomene jejich koupi. Pokud si uživatel bude chtít být opravdu jistý, že se darovaný předmět setká s pozitivní odezvou, může se prostřednictvím aplikace anonymně zeptat obdarovaného, co by si přál. Pokud naopak chce sdělit svá přání ostatním, může si vyplnit osobní seznam přání, který si ostatní uživatelé poté zobrazí. Aplikace byla implementována pro mobilní operační systém Android. Hlavním důvodem pro výběr této platformy byla skutečnost, že aplikace pro ni lze psát v Javě, se kterou jsem se důkladně seznámil během studia. Dalším důvodem pak byla nízká pořizovací cena zařízení pro testování a vývojářské licence. Toto téma jsem si zvolil především proto, že může lidem pomoci s řešením jejich problémů a přispět svou troškou k příjemnějšímu životu v 21. století. Navíc reflektuje současný trend, kdy se velká část myšlenkových pochodů přesouvá z lidského mozku do útrob mobilních telefonů, chytrých hodinek či webových služeb. Na základě prvotních reakcí mých přátel věřím, že výsledná aplikace si najde své uživatele a naplní jejich očekávání.

Kapitola 2

Popis problému a cíle

V této kapitole si podrobněji vysvětlíme problém, o jehož řešení vyvíjená aplikace usiluje. Dnešní společnost má tendenci směřovat k uspěchanému, konzumnímu způsobu života. Především v metropolích lidé nechtějí ztratit ani vteřinu navíc hledáním či přemýšlením. Hodnotné informace tak nabývají vyšší hodnoty než kdykoli dříve a chytré telefony, nerozluční společníci většiny z nás, jsou ideálními kandidáty pro jejich poskytování. Všeobecné zrychlení životního stylu překvapivě nijak neublížilo tradicím spojeným s nakupováním dárků. Možná právě konzumní způsob života přemístil těžiště oslav z duchovní roviny směrem k materialismu, který hmotné dary ideálně reprezentují.

Každý člověk, který někdy nakupoval dary pro své blízké, musel věnovat určitý čas přemýšlení nad jeho správným výběrem. Který dárek by obdarovaného nejvíce potěšil, nejlépe by vyhovoval jeho potřebám a zkrátka – udělal mu největší radost? Takové dilema zatěžuje každoročně myšlení milionů lidí slavících vánoční svátky, narozeniny, jmeniny a další. Obchodní centra jim v rozhodování příliš nepomáhají, předimenzovaný sortiment a regály plné zboží vedou spíše k rozhodovací paralýze [1] a zahlcení informacemi, než k efektivnímu a cílenému výběru dárku. Tyto faktory jen zvyšují míru stresu a psychické zátěže, které nijak neprospívají klidné sváteční atmosféře. Vyvíjená aplikace si klade za cíl dodávat lidem inspiraci při výběru dárků pro své blízké. K tomuto účelu bude sloužit vyhledávání dárků na základě podrobné specifikace adresáta prostřednictvím jeho zájmů, věku, pohlaví a dalších informací, které jsou podrobněji popsány v kapitole 4.1. Aplikace také usiluje o eliminaci situací, kdy obdarovaný není s dárkem spokojen. Takovým nepříjemnostem lze předcházet vyplněním osobního seznamu přání v aplikaci. Tento seznam si ostatní uživatelé mohou zobrazit a rezervovat si z něho dárek, který chtějí sami danému člověku zakoupit, aby se předešlo duplicitám. Druhým způsobem, jak předcházet nevhodným dárkům, je zaslání anonymního dotazu osobě, kterou chce uživatel obdarovat.

Cíl práce spočívá ve vytvoření mobilní aplikace pro Android komunikující pomocí REST API 5.1.3 s internetovou databází dárků. Aplikace bude z databáze nejen čerpat, ale také do ní vkládat data na základě uživatelského vstupu. V rámci semestrálního projektu bylo vytvořeno uživatelské rozhraní aplikace a byla zprovozněna komunikace s internetovou databází pomocí dotazů. V rámci bakalářské práce pak byly implementovány všechny hlavní prvky funkcionality aplikace a byla zprovozněna komunikace mezi aplikací a webovým serverem prostřednictvím REST API.

Důvodem volby platformy Android byla především skutečnost, že od svého vydání v roce 2009 [2] se rozšířila téměř do všech věkových a kulturních skupin. Díky tomu jsou mobilní aplikace pro Android dobrou příležitostí, jak rychle oslovit velmi široké publikum. Relativně jednoduchý vývoj a nízké poplatky za licence či testovací zařízení však také způsobily, že aplikací vzniká nepřehledné množství (v červenci 2014 jich v internetovém obchodě s aplikacemi Google Play existovalo více než 1 300 000 [3]). Proto není jednoduché najít pomyslnou mezeru na trhu a prosadit v takové konkurenci vlastní aplikaci. V nadcházející kapitole si však ukážeme, že konkurenční prostředí v oblasti mobilních aplikací pro doporučování dárků trpí řadou problémů, se kterými si vývojáři

nedokázali poradit. Aplikace mají zpravidla omezenou funkcionalitu a uživatelé jsou často nespokojeni s kvalitou doporučení. Pokud se nám tedy v rámci této práce podaří vyvinout fungující řešení, které reflektuje požadavky uživatelů, věříme, že v existující konkurenci obstojí.

Kapitola 3

Rešerše existujících řešení

V této kapitole se analyzují existující aplikace sloužící k doporučení dárků. Existuje jich relativně mnoho, nicméně většina z nich kvůli nevalné kvalitě není příliš konkurenceschopná. Některé z nich kvalitativně obstojí, avšak jejich zaměření se odlišuje od této aplikace. Bylo tedy vybráno 5 nejvýznamnějších konkurentů na platformě Android. U každé aplikace byly zhodnoceny její klady (ty mohou sloužit jako inspirace) a zápory (ty poukážou na problémy, kterým by bylo vhodné se v aplikaci vyhnout).

Vybrané aplikace pro Android:

- El Gifto - Gift Ideas Guru
- Gift Generator
- Gift Wizard
- Gift Ideas India – allMemoirs
- Gift Idea Generator

3.1 El Gifto - Gift Ideas Guru

Aplikace El Gifto slouží k doporučení dárků na základě poměrně přesné specifikace osoby, kterou chce uživatel obdarovat. Aplikace umožňuje zadat například věk, pohlaví, charakter a další atributy adresáta. Zadávání vlastností probíhá v rozbalovacím menu, které působí zajímavě a poskytuje i přehled o již vyplněných údajích.

Problém však nacházím v běžném, uživatelském dojmu z používání aplikace. Po spuštění El Gifto požaduje přihlášení pomocí e-mailu a hesla, jehož zahrnutí jsem zvažoval i v této bakalářské práci. Obával jsem se však, že uživatele přihlašovací proces odradí. El Gifto s touto situací počítá, a tak umožňuje přihlášení přeskočit. Poté ale přichází vyhledávání přátel. Aplikace nabízí výběr z nedávných přátel, kontaktů telefonu či sociálních sítí Twitter a Google+. Například u kontaktů z Google+ ale zřejmě aplikace získává pouze e-mail a nevyužívá například jméno či pohlaví, které toto API poskytuje. Je tedy nutné zadat všechny údaje pro kontakty ručně, a ve výsledku import z Google+ uživateli příliš nepomůže. Navíc zřejmě aplikace nijak nepáruje uživatele prostřednictvím sociálních sítí, a tak spolu nemohou komunikovat například vzájemným sdílením seznamů přátel. Potenciál sociálního faktoru zde tedy není plně využit.

Dalším problémem, na který jsem při testování aplikace narazil, byla nepřehledná správa kontaktů. Po zadání jména, příjmení a atributů kontaktu jsem stiskl tlačítko **Zobrazit dárky**, které mi zobrazilo seznam doporučení. Po návratu na obrazovku pro výběr kontaktů mi však chyběl nějaký seznam osob, ze kterého bych mohl vybírat pro příští vyhledávání. V menu je sice možnost zobrazit si nedávné kontakty, ale ani po několika vyhledáváních a opětovném zadání informací o kontaktu se daný kontakt neuložil a aplikace hlásila, že nejsou k dispozici žádné nedávné kontakty. Očekával bych, že kontakty se budou ukládat automaticky již při zadání atributů, nebo alespoň bude v uživatelském rozhraní viditelné tlačítko pro jejich uložení.

Abych ale nebyl pouze kritický, v prezentaci aplikace v obchodě Google Play obsahuje celou řadu velmi pozitivních komentářů, a tak zřejmě existuje poměrně mnoho uživatelů, kteří jsou s aplikací El Gifto spokojeni.

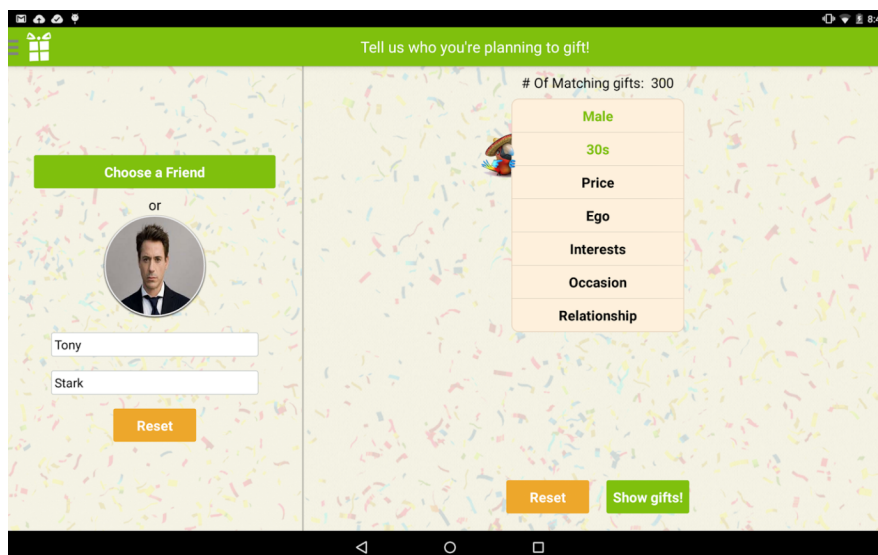
Pro lepší představu lze navštívit stránku aplikace na Google Play ¹⁾.

Klady

- Aplikace nabízí širokou škálu atributů, podle kterých lze upřesnit adresáta dárku
- Aplikace má zajímavý název a výrazné logo, díky kterým vynikne mezi svými konkurenty na Google Play
- Aplikace odkazuje uživatele přímo do internetového obchodu, kde může doporučený dárek zakoupit. Podobný systém by bylo možné v budoucnu zavést také do této aplikace.
- Aplikace obsahuje vtipné hlášky v anglickém jazyce

Zápory

- Uživatelské rozhraní není intuitivní – při zadávání údajů o kontaktu například najdeme vpravo tlačítko **Další**, ale vlevo již namísto očekávaného **Zpět** je pouze tlačítko **Reset**, které mi nepřipadalo v dané situaci užitečné
- Dotazy na databázi trvají poměrně dlouho
- Aplikace zřejmě doporučuje pouze dárky z jednoho internetového obchodu, a tak je počet doporučení omezený
- Aplikace není odladěná (při testování přestala pracovat)
- Aplikace nezískává data od uživatelů, a její databáze tak zřejmě není aktualizována
- Aplikace vyžaduje neustálé připojení k internetu. Její použití tedy může být problematické v místech, kde je připojení nestabilní



Obrázek 3.1. Snímek obrazovky z aplikace El Gifto

3.2 Gift Generator

Aplikace Gift Generator slouží k doporučování dárků na základě jediného atributu – vztahu k dané osobě (například otec, babička, přítel). Uživatelskému rozhraní dominuje

¹⁾ <https://play.google.com/store/apps/details?id=com.ibkr.elgifto>

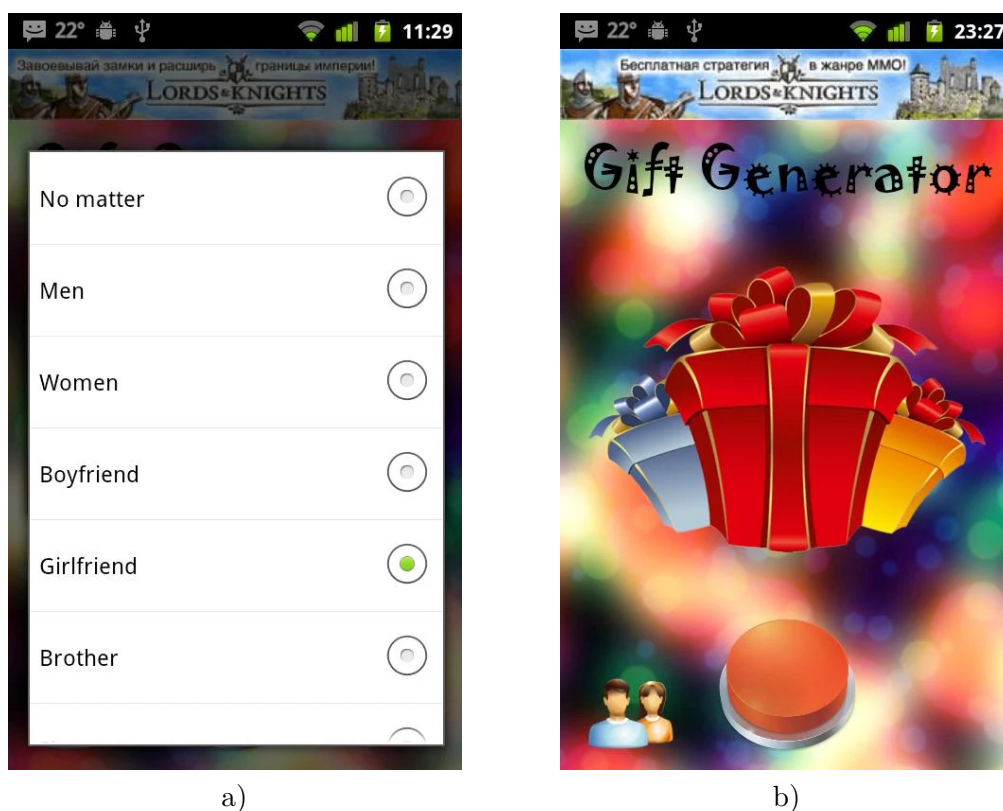
velké tlačítko ve spodní části obrazovky, které slouží ke spuštění vyhledávání dárku. V horní části obrazovky je pak název aplikace, pod kterým najdete samotné doporučení na dárek doplněné ilustrací. Pro lepší představu lze navštívit stránku aplikace na Google Play ¹⁾.

Klady

- Při vyhledávání doporučení je zde zajímavá animace, která uživateli zpříjemní čekání
- Ovládání formou velkého tlačítka působí intuitivně a potvrzuje tak primární funkci aplikace

Zápory

- Aplikace zřejmě disponuje jen omezeným seznamem doporučení (po chvíli testování se doporučení začala opakovat)
- Aplikace nekontroluje, zda je uživatel připojen k internetu. Po vypnutí internetového připojení telefonu tak aplikace působila, že vyhledává doporučení, ale uživatel by na ně čekal marně
- Aplikace není na Google Play průběžně aktualizována, převažují u ní negativní komentáře a její popis je velmi strohý
- Aplikace nezískává data od uživatelů, a její databáze tak zřejmě není aktualizována



Obrázek 3.2. Snímky obrazovky z aplikace Gift Generator

¹⁾ <https://play.google.com/store/apps/details?id=com.mihaelisaev.giftsgenerator>

3.3 Gift Wizard

Aplikace Gift Wizard slouží k doporučování dárků na základě relativně přesné specifikace adresáta. Lze zadat například věk, zájmy, pohlaví či osobnost. Uživatelské rozhraní představuje sekvence dialogů se zaškrťovacími poli, ve kterých lze upřesnit adresáta a příležitost, ke které chceme dárek zakoupit. Pro lepší představu lze navštívit stránku aplikace na Google Play ¹⁾.

Klady

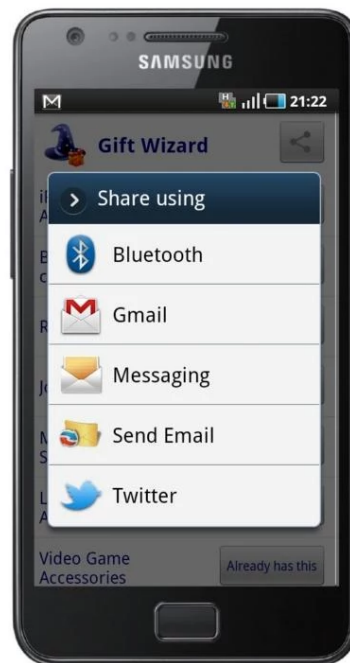
- Aplikace nabízí celou řadu atributů pro specifikaci adresáta dárku
- Aplikace zřejmě obsahuje také seznamy dárků (vzhledem ke zpoplatnění nebyla testována)
- Aplikace pravděpodobně umožňuje sdílení seznamů dárku na Twitteru. Sdílení informací na sociálních sítích by do této aplikace mohlo být doplněno později

Zápory

- Aplikace je zpoplatněná, aniž by nabízela alespoň omezenou Lite verzi
- Aplikace dává podle komentářů uživatelů vždy podobná doporučení, nehlédě na zadané atributy adresáta
- Uživatelské rozhraní aplikace působí zastarale a má poměrně strohý design
- Aplikace není na Google Play aktualizována



a)



b)

Obrázek 3.3. Snímky obrazovky z aplikace Gift Wizard

¹⁾ <https://play.google.com/store/apps/details?id=com.sockmonkeysolutions.giftwizard>

3.4 Gift Ideas India - allMemoirs

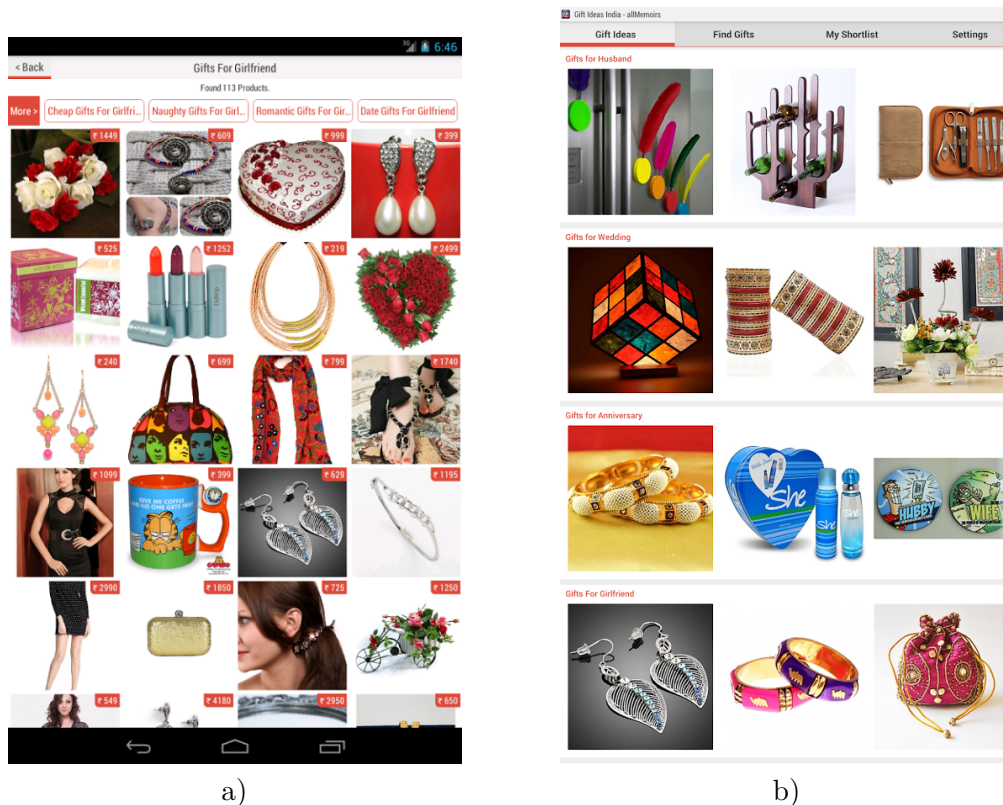
Aplikace Gift Ideas India slouží k doporučování dárků na základě vztahu k dané osobě a typu dárku, který chce uživatel zakoupit. Uživatelské rozhraní obsahuje 4 záložky a mřížku s obrázky, které představují doporučení na dárky. Pro lepší představu lze navštívit stránku aplikace na Google Play ¹⁾.

Klady

- Obrázky jako doporučení na dárky působí poutavě
- V aplikaci je integrováno přihlášení na Facebook, díky kterému se aplikace může šířit

Zápory

- Nelze přesně specifikovat adresáta dárku, a tak jsou doporučení poměrně obecná
- Uživatelské rozhraní aplikace je orientováno převážně na indický trh (například ceny jsou uvedeny pouze v rupiích)
- Aplikace nezískává data od uživatelů, a její databáze tak zřejmě není aktualizována



Obrázek 3.4. Snímky obrazovky z aplikace Gift generator

3.5 Gift Idea Generator

Aplikace Gift Idea Generator slouží k doporučování dárků na základě zodpovězení série otázek, na které lze odpovědět vždy dvěma způsoby. Takto lze dosáhnout mnoha scénářů, kdy každý je zakončen unikátním doporučením dárku. Uživatelské rozhraní

¹⁾ <https://play.google.com/store/apps/details?id=com.allmemoirs.gifts>

představuje jednoduché menu, ze kterého lze přejít do série otázek. Otázka dominuje prostřední části obrazovky a dvojice tlačítek pro odpovědi se nachází na dolním okraji. Pro lepší představu lze navštívit stránku aplikace na Google Play ¹⁾.

Klady

- Aplikace umožňuje specifikaci adresáta na základě konkrétních otázek
- Aplikace má intuitivní, poutavé uživatelské rozhraní. Po přečtení doporučení se aplikace vrátí do menu, díky čemuž se uživatel neztratí. Tento trend by bylo vhodné zachovat i v této aplikaci
- Doporučení na dárky jsou vtipná a zajímavá

Zápory

- Aplikace není zpětně kompatibilní – podporuje jen Android 4.0 a vyšší
- Pokud je doporučení na dárek delší než vyhrazené okno, nelze si ho přečíst celé
- Doporučení se opakují kvůli omezenému počtu větvení scénářů. Po vyzkoušení všech scénářů výrazně klesá uživatelská hodnota aplikace
- Aplikace nezískává data od uživatelů a její databáze tak zřejmě není aktualizována



Obrázek 3.5. Snímky obrazovky z aplikace Gift Idea Generator

¹⁾ <https://play.google.com/store/apps/details?id=com.GiftAndroid>

Kapitola 4

Analýza aplikace

V předchozí kapitole již bylo popsáno, jaká konkurence na trhu existuje a jaké jsou její silné a slabé stránky. Nyní je tedy čas začít se samotným návrhem mobilní aplikace. Nejprve bude charakterizována cílová skupina, a poté budou rozebrány jednotlivé aspekty aplikace. Primárním cílem bude vyvarovat se chyb, kterých se dopustili autoři jiných aplikací, a naopak maximalizovat přidanou hodnotu pro uživatele, kterou aplikace správnou funkcionalitou může poskytnout. Tato kapitola je pro výslednou aplikaci zcela zásadní, neboť zde jsou definovány funkce, kterými si bude získávat přízeň samotných uživatelů.

4.1 Cílová skupina

Při vymýšlení nových mobilních aplikací vždy nastává velké dilema, zda je lepší zaměřit se na užší cílovou skupinu a poskytnout jí vysokou přidanou hodnotu, anebo naopak rozšířit záběr na co nejvyšší počet potenciálních uživatelů. V prvním případě můžeme implicitně počítat s vyšší tzv. hodnotou uživatelů [4], protože jim přinášíme produkt určený speciálně pro ně a vhodně plní jejich potřeby. V této bakalářské práci ale půjdeme druhou cestou, neboť téma aplikace pro doporučování dáreků má potenciál pokrýt prakticky všechny věkové i zájmové kategorie. Jediným omezením tak zůstává skutečnost, že uživatel aplikace musí disponovat zařízením s operačním systémem Android. Vzhledem k zhruba 80% podílu Androidu na světovém poli mobilních operačních systémů (viz obrázek 4.1) nám stále zbývá více než dostatečná množina potenciálních uživatelů. Aby se přesto uživatelé necítili touto podmínkou omezení, implementovali jsme funkci anonymních e-mailových dotazů na dárek, díky které mohou stávající uživatelé zapojit také osoby nevlastnící telefon či tablet s Androidem. V budoucnosti však nevyklučujeme také implementaci verze pro operační systém iOS, pokud se nápad realizovaný v Androidu osvědčí.

Period	Android	iOS	Windows Phone	BlackBerry OS	Others
Q4 2014	76.6%	19.7%	2.8%	0.4%	0.5%
Q4 2013	78.2%	17.5%	3.0%	0.6%	0.8%
Q4 2012	70.4%	20.9%	2.6%	3.2%	2.9%
Q4 2011	52.8%	23.0%	1.5%	8.1%	14.6%

Obrázek 4.1. Graf podílů mobilních operačních systémů na světovém trhu mezi lety 2009 až 2014 [5]. Zdroj: IDC

4.2 Funkční požadavky

- Zadání obecných údajů o adresátovi dárku 4.1 a jeho zájmů 4.2

- Zadání dárku, který uživatel daroval dané osobě minulý rok (což zajistí organický růst databáze dáreků)
- Při zadávání loňského dárku bude k dispozici našeptávač, díky kterému uživatel nemusí pokaždé psát celý název dárku
- Doporučení dáreků na míru specifikovanému adresátovi
- Přihlašování pomocí sociálních sítí Google+ a Facebook, díky kterému se uživatel nemusí složitě registrovat.
- Importování přátel ze sociálních sítí Google+ a Facebook
- Ukládání dříve zadaných adresátů do persistentního úložiště aplikace
- Odesílání anonymních e-mailů lidem, kterým by uživatel rád koupil dárek, ale neví, co by si přáli
- Osobní seznamy přání uživatelů – uživatel si může nastavit dárky, které by si přál dostat. Ostatní uživatelé si mohou některý z těchto dáreků rezervovat - po dobu rezervace se dárek ostatním uživatelům zobrazuje, ale nemohou si jej sami rezervovat.
- Kreditový systém - uživatel bude pomoci získávat kredity, za které poté může například získávat nápady či využívat jiné funkce aplikace. Podrobněji se o kreditovém systému zmíníme v kapitole 4.5

4.3 Hlavní funkce

Nyní, když již známe základní funkční požadavky aplikace, můžeme se zamyslet nad tím, jakou přidanou hodnotu budou uživatelé poskytovat. Nejprve se zaměříme na primární funkcionalitu aplikace, tedy samotné doporučování dáreků.

Získávání doporučení na dárky

Jedná se o velmi důležitou funkci aplikace, která rozhoduje o jejím úspěchu na trhu. Podrobně jsme proto analyzovali všechna úskalí a nedostatky, které by mohly ohrozit jak kvalitu poskytovaných doporučení, tak uživatelskou přívětivost systému.

Údaje o adresátovi	Údaje o dárku
Pohlaví adresáta	Cenové rozpětí dárku
Věk adresáta	Příležitost, pro kterou je dárek určen
Vztah k adresátovi	

Tabulka 4.1. Obecné údaje o adresátovi a dárku

Vyplnění údajů o adresátovi a dárku

Uživatel, který chce získat doporučení na dárek, nejprve vyplní obecné údaje o adresátovi a dárku (viz. tabulka 4.1). Dále může zadat libovolný počet zájmů adresáta dárku. Množina všech zájmů představuje nejtypičtější činnosti, kterými se lidé v civilizované společnosti zabývají. Jedná se například o sport, módu, kulturu, knihy a seriály či péči o zvířata, ale také o specifičtější zájmy, jako jsou konkrétní typy sportů, odvětví módy, žánry knih či druhy zvířat. Tyto aktivity může uživatel filtrovat a vybírat prostřednictvím vyhledávacího pole.

Navržení vlastního zájmu

V případě, že uživatel v aplikaci postrádá určitý zájem, může jej autorům navrhnout pomocí zvláštního dialogu. Pokud jej autoři shledají jako vhodné rozšíření současné

Zájem
Auta
Cestování
Elektronika
Hraní her
Hudba
Kempování
Knihy a filmy
Kultura
Móda
Péče o zvířata
Posilování
Rybaření
Sporty
Vaření
Zahrada
a další...

Tabulka 4.2. Příklady zájmů adresáta

škály, vloží je do databáze zájmů, kterou si aplikace automaticky stahuje. Díky tomu mohou sami uživatelé přispět ke kvalitnějšímu obsahu aplikace a stanou se tak součástí komunity.

Kontrola kreditů a zadání minulého dárku

Po vyplnění alespoň jednoho atributu adresáta může uživatel vyhledat samotná doporučení na dárek. Po kliknutí na tlačítko pro vyhledání systém zkontroluje počet kreditů uživatele. V případě, že uživatel nemá dostatečný počet kreditů či placenou verzi aplikace, bude vyzván k zadání dárku, který dané osobě dal při minulé příležitosti. Pokud této osobě dosud žádný dárek nedal, může získat kredity vyplněním cen dárků, které jich dosud nemají přiřazen dostatečný počet. Při zadávání minulého dárku bude mít uživatel k dispozici našeptávač, který nabídne již existující dárky v databázi na základě již napsaného textu. Systém uloží minulý dárek do centrální databáze s údaji, kterými uživatel předtím specifikoval adresáta. Tyto údaje budeme nadále označovat jako tagy, kdy každý dárek v databázi může mít neomezený počet tagů s určitými váhami. Při každém vložení již existujícího dárku s tagy se přidají případné nové tagy s váhou 1 a stávajícím tagům bude navýšena váha o 1 bod. Tímto mechanismem se zajistí organický růst databáze dárků, aniž by do ní museli tvůrci aplikace zasahovat. Za vložení minulého dárku uživatel obdrží určitý počet kreditů, které může využít pro další vyhledávání.

Vyhledání a zobrazení doporučení na dárky

Pokud je vyplněn alespoň jeden atribut a uživatel má dostatek kreditů, zobrazí se prvních deset doporučení na dárky s nejvyšší relevancí. Relevance dárků se vypočítá jako podíl váženého součtu všech tagů, které se vyskytují jak u existujícího dárku v databázi, tak mezi tagy zadanými při specifikaci adresáta, a váženého součtu tagů přiřazených k dárku, který má nejvyšší vážený součet tagů v celé internetové databázi. Nalezení nápadů na dárky probíhá na serveru. Algoritmus výpočtu relevance dárku je podrobněji popsán v kapitole 5.4.1. Uživateli bude poté zobrazena relevance každého dárku v seznamu. Pokud se relevance budou uživateli zdát příliš nízké, může zopakovat vyhledávání například s větším počtem vyplněných zájmů.

4.4 Doplnkové funkce

Preferovaní a ostatní přátelé

Uživatel si může v aplikaci vytvářet přátele ručně zadáním jména, nebo je importovat ze sociální sítě Google+ či Facebook. Přátele budeme nadále v této bakalářské práci označovat také jako **kontakty**. Noví přátelé se automaticky vloží do skupiny **Ostatní přátelé**. Pokud chce uživatel pro tyto přátele např. vyhledávat nápady či seznamy přání, musí si je nejprve přidat do skupiny **Preferovaní přátelé**. Tímto mechanismem, k jehož konceptu mě přivedl vedoucí této bakalářské práce, pan Ing. David Sedláček, Ph.D., lze vybrat ze seznamu importovaných přátel pouze tu množinu, která je pro uživatele zajímavá a k níž má určitý bližší vztah. Vzhledem k dnešnímu trendu, kdy uživatelé sociálních sítí mají řádově stovky až tisíce přátel, je toto třídění v aplikaci potřebné. Aplikace dále umožňuje skupinu **Ostatní přátelé** filtrovat zadáním textu do vyhledávacího pole, čímž lze urychlit třídící proces.

Seznam přání uživatele

Osobní seznamy přání představují rozhraní, pomocí kterého může uživatel sdílet svým blízkým, jaké dárky by rád dostal. Pro použití této funkcionality se uživatel musí nejprve přihlásit pomocí sociální sítě Google+ nebo Facebook, aby jej aplikace mohla identifikovat a spárovat s ostatními uživateli. Poté již může přidávat dárky do svého seznamu přání. Ostatní uživatelé si tyto dárky mohou zobrazit a rezervovat či přidat do svého nákupního seznamu.

Seznam přání jiného člověka

Uživatel si může prohlížet seznamy přání lidí, které má nastaveny jako preferované přátele a kteří byli importováni ze sociální sítě. Uživatel si může rezervovat dárky z jejich seznamů přání a přidat si je do svého nákupního seznamu. Rezervaci může později zrušit, pokud si nákup daného dárku rozmyslí. Poté se uvolní dárek k rezervaci pro ostatní uživatele.

Přihlášení prostřednictvím sociální sítě Google+ a Facebook

Přihlášení do sociální sítě představuje důležitou funkcionalitu aplikace, která umožňuje vzájemně spárovat její uživatele. Bez přihlášení není možné používat funkci seznamů přání a anonymních dotazů, což může být pro některé uživatele nepříjemné zjištění. Na druhou stranu, proces přihlášení do sociálních sítí je výrazně jednodušší a méně obtěžující než případné vytváření proprietárního účtu v aplikaci, potvrzení e-mailem a vytváření hesla. Přihlášení na Facebook vyžaduje pouze zadání přihlašovacích údajů do dialogu, přihlášení na Google+ pak dokonce jen potvrzení požadovaných oprávnění aplikace a výběr aktivního Google+ účtu v telefonu. V případě, že uživatel nemá vytvořen účet v žádné z těchto sociálních sítí, pak nemůže zmíněné funkce využívat a musí si účet případně vytvořit. Předpokládám ale, že naprostá většina cílové skupiny účet na Facebooku či Google+ již vlastní.

Zaslání anonymního dotazu na dárek

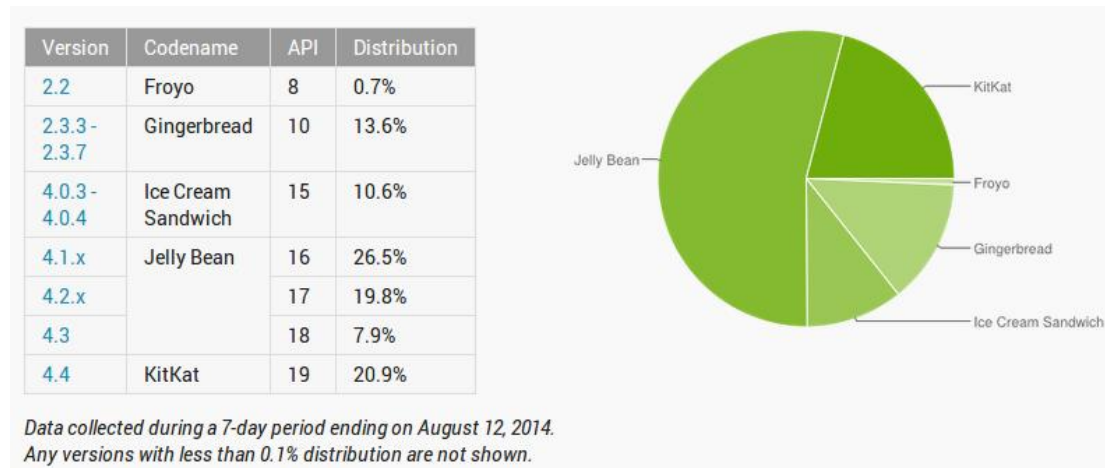
Aplikace umožňuje zapojit také osoby, které ji nemají ve svém zařízení nainstalováno, a to formou zaslání anonymního e-mailu. Tuto akci začíná uživatel aplikace (dále jen **tazatel**), který si ze svých kontaktů vybere osobu, jíž chce koupit dárek (dále jen **adresát**). Pokud adresáta ještě v kontaktech nemá, může jej vytvořit. Poté musí zadat

jeho platný e-mail, pokud nebyl získán ze sociální sítě, a alespoň tři návrhy dáreků, které by se adresátovi mohly líbit. Aplikace poté zašle ze svého poštovního serveru anonymní e-mail na zadanou adresu s dotazem, do jaké míry by si adresát přál dostat jednotlivé dárky navržené tazatelem. Adresát může ke každému z nich pomocí posuvníku nastavit prioritu, s jakou si daný dárek přeje. Tím se zajistí, aby sám adresát nevěděl, který dárek přesně dostane, ale všechny možnosti pro něj budou přijatelné. Dále může k navrženým dárkům doplnit textový komentář, který se poté zobrazí tazateli. Odesílání anonymních dotazů je chráněno proti posílání velkého množství zpráv kontrolou, zda uživatel nemá dosud nezodpovězený dotaz u daného kontaktu. V budoucích verzích aplikace bude tato ochrana pravděpodobně podpořena také zpoplatněním funkce anonymního dotazu za kredity.

Nákupní seznam Uživatel může získané nápady na dárky jedním kliknutím přidávat do svého nákupního seznamu spolu s osobou, pro níž je dárek určen, a časem přidání. Stejným způsobem pak dárky může z nákupního seznamu odstraňovat. Do nákupního seznamu lze obdobně přidávat také dárky ze seznamu přání jiného uživatele či z odpovědí na anonymní dotazy, které obdržel. Sekce nákupního seznamu je rozdělena na aktuální nákupní seznam a historii nákupů. V historii nákupů se nacházejí dárky, které uživatel označil jakou koupené. Pokud se uživatel například spletl a označí nesprávný dárek jako koupený, může dárek z historie jedním kliknutím vrátit zpět do aktuálního seznamu.

■ 4.4.1 Zjištěná rizika a navrhovaná řešení

- Doporučení mohou být příliš obecná, nebudou vystihovat charakteristické rysy adresáta dárku
 - **Řešení:** Uživatel může podrobně specifikovat adresáta pomocí věku, pohlaví, vztahu k němu a široké škály zájmů
- Doporučení se ze začátku mohou opakovat, pokud jich nebude dostatečný počet
 - **Řešení:** Manuální přidání dostatečného počtu doporučení před vydáním.
- Doporučení nebudou reflektovat nejnovější trendy, budou zkrátka zastarávat
 - **Řešení:** Získávání doporučení přímo od uživatelů
- Tvůrci aplikace budou muset pravidelně přidávat doporučení do databáze, aby udrželi jejich kvalitu
 - **Řešení:** Získávání doporučení přímo od uživatelů
- Pro některé funkce je nutné identifikovat uživatele. Proces přihlašování do sociální sítě může některé uživatele odradit, nebo ani nevlastní účet.
 - **Řešení:** Přihlášení bude vyžadováno pouze pro funkce, které jej vyžadují. Nebude však obalovat celou aplikaci. Pokud nevlastní účet v sociální síti, stále mohou používat funkci získání nápadu na dárek. Musí si ale vytvořit své kontakty ručně.
- Co když uživatel dané osobě nedal dosud žádný dárek, nebo si nepamatuje, jaký to byl? Jak poté může získat kredity pro vyhledávání?
 - **Řešení:** V současné verzi aplikace zatím řešení nebylo implementováno. Do budoucna se plánuje možnost doplňovat ceny k neoceněným dárkům či jinak přispívat k rozvoji aplikace za kredity.



Obrázek 4.2. Podíly verzí systému Android (Převzato z [6])

4.5 Kreditový systém

Ve funkčních požadavcích 4.2 jsme nastínili existenci kreditového systému v aplikaci. Každý uživatel, který si aplikaci nainstaluje, získá určitý obnos kreditů, za které může například vyhledávat nápady na dárky. Pokud by jej napadlo smazat si lokální data aplikace, což systém Android umožňuje, získá sice kredity nové, ale ztratí také všechny koníčky a další vyplněné informace o přátelích. Proto nepředpokládám, že by tímto způsobem uživatelé aplikaci zneužívali. Primárním cílem kreditového systému je omezit množství dotazů na databázi za účelem získání nápadů a současně motivovat uživatele, aby se podíleli na rozvoji aplikace. Kromě zadávání minulého dárku, kterým se rozšiřuje databáze nápadů, se do budoucích verzí aplikace plánuje přidat také možnost zadávání cen neoceněných dáreků či překládání názvů dáreků a koníčků. O problematice překládání se podrobněji zmíníme v sekci 4.7. Za tyto prospěšné činnosti bude uživatel odměněn určitým množstvím kreditů.

4.6 Obecné požadavky

- Aplikace bude podporovat Android verze 2.3.3 a vyšší. Počet zařízení využívajících tuto verzi Androidu sice představuje v době vývoje aplikace méně než 6,5% trhu (viz obrázek 4.2) a klesá, ale díky knihovnám zpětné kompatibility není problém zachovat podporu i pro starší zařízení.
- Aplikace bude napsána v jazyce Java.
- Pro komunikaci s MySQL databází na internetu bude využito REST rozhraní, jehož autorem je Bc. Roman Kozák a Bc. Tomáš Sládek. Právě REST API se jeví jako jediné rozumné řešení pro komunikaci Android aplikace s online MySQL databází.
- Získávání nápadů na dárky vyžaduje připojení k internetu (databáze nápadů nebude uživateli poskytnuta lokálně).
- Aplikace bude lokalizována do češtiny a angličtiny, v případě rozšíření také do dalších jazyků.

4.7 Lokalizace

Jak jsme zmínili v obecných požadavcích 4.6, aplikace bude lokalizována do češtiny a angličtiny. Vzhledem k charakteru databáze koníčků a dáreků, kdy uživatelé budou

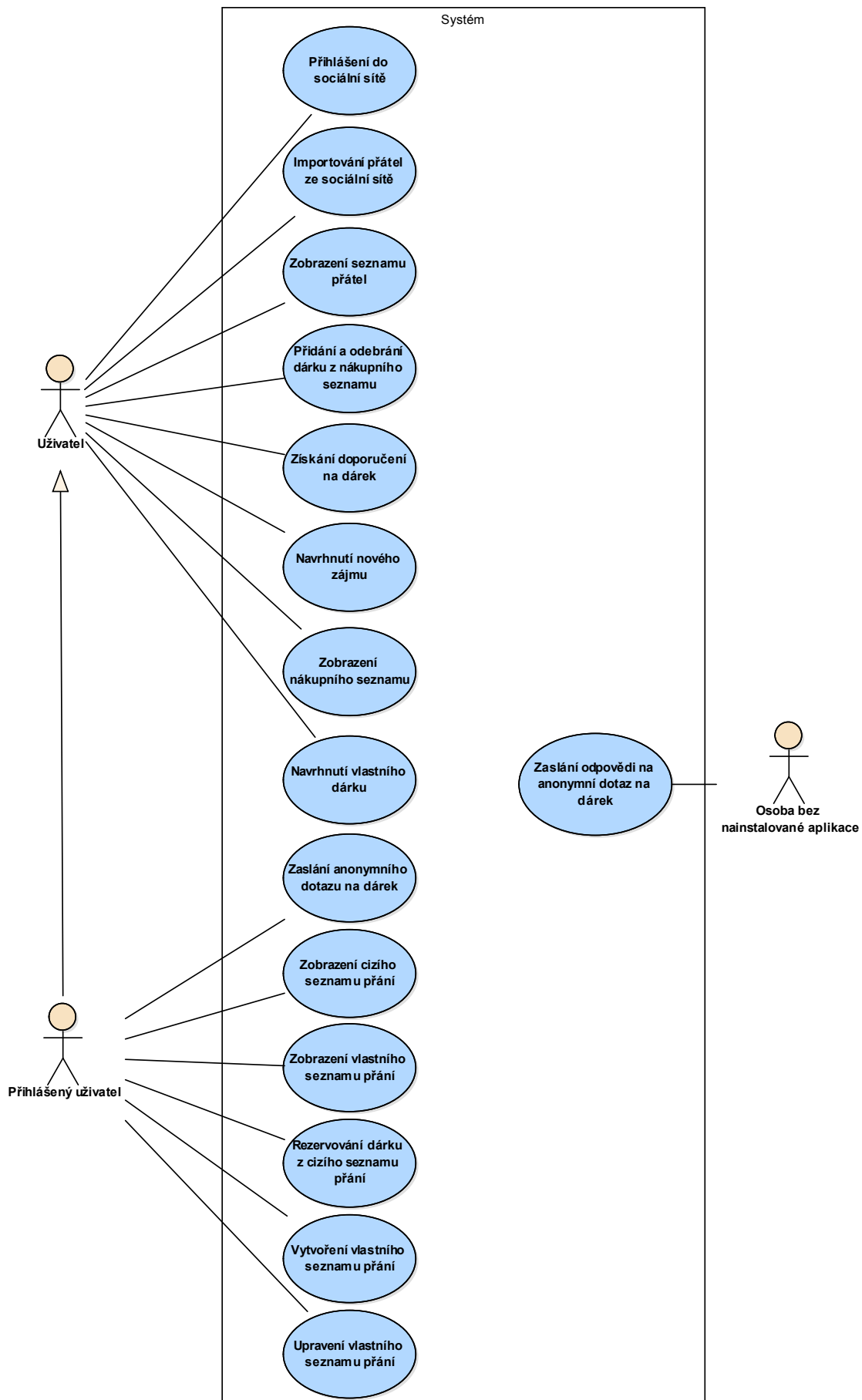
přidávat záznamy ve svém mateřském jazyce, však nastává problém s lokalizací těchto záznamů do ostatních jazyků. V současné verzi aplikace bude tedy lokalizováno pouze uživatelské rozhraní aplikace. Do budoucna se však plánuje, že uživatelé mohou sami překládat názvy dáreků a koníčků do jiných jazyků a tím také získat kredity potřebné pro vyhledávání.

■ 4.8 Případy užití

Již jsme si stanovili, jaké funkce bude aplikace poskytovat a které technické prostředky k tomu bude využívat. Na základě těchto poznatků nyní sestavíme případy užití, abychom mohli určit hranice vyvíjeného systému.

■ 4.8.1 Model případů užití

Hlavní aktéři, kteří budou v případech užití vystupovat, jsou (nepřihlášený) Uživatel, Přihlášený uživatel a Osoba bez nainstalované aplikace. Právě interakce osob bez nainstalované aplikace formou webového formuláře pokrývá požadavek v zadání na zapojení uživatelů nevlastnících mobilní telefon nebo tablet.



Obrázek 4.3. Model případů užití aplikace

4.8.2 Scénáře případů užití

Scénář případu užití	
Název případu užití	Vytvoření uživatelského účtu
Identifikační číslo případu užití	UC01
Primární aktéři	Uživatel
Sekundární aktéři	-
Vstupní podmínky	Dostupné internetové připojení
Výstupní podmínky	Na serveru je vytvořen nový uživatelský účet
Hlavní scénář	1. Uživatel zadá svou e-mailovou adresu a heslo a potvrdí
	2. Systém ověří platnost zadaných údajů IF Údaje jsou platné THEN Systém vytvoří nový uživatelský účet ELSE Alternativní scénář 1
	3. Systém přihlásí uživatele
Alternativní scénář 1	1. Systém zobrazí informaci o chybě

Obrázek 4.4. Scénář případu užití vytvoření uživatelského účtu

Scénář případu užití	
Název případu užití	Vyhledání doporučení na dárky
Identifikační číslo případu užití	UC02
Primární aktéři	Uživatel
Sekundární aktéři	-
Vstupní podmínky	Dostupné internetové připojení
Výstupní podmínky	Uživateli se zobrazí vhodná doporučení na dárky
Hlavní scénář	2. Uživatel zadá obecné údaje a zájmy adresáta a potvrdí
	Systém ověří, zda jsou zadány všechny povinné údaje (pohlaví, vztah k adresátovi, cena dárku, příležitost, alespoň 2 hlavní a 3 vedlejší zájmy) IF Všechny povinné údaje jsou zadány THEN Systém zobrazí dialog s očekávanou přesností vyhledávání ELSE Alternativní scénář 1
	3. Uživatel potvrdí, že chce spustit vyhledávání
	4. Systém ověří, zda má uživatel dostatek kreditů pro vyhledání doporučení IF Má dostatek kreditů THEN Systém zobrazí prvních 10 doporučení na dárky ELSE Alternativní scénář 2
Alternativní scénář 1	1. Systém zobrazí informaci o chybějících údajích
Alternativní scénář 2	1. Systém zobrazí dialog pro vyplnění minulého dárku
	2. Uživatel vyplní dárek, který dal adresátovi minulý rok a potvrdí
	3. Systém zobrazí prvních 10 doporučení na dárky

Obrázek 4.5. Scénář případu užití získání doporučení na dárek

Scénář případu užití	
Název případu užití	Rezervace dárku ze seznamu přání
Identifikační číslo případu užití	UC03
Primární aktéři	Uživatel
Sekundární aktéři	-
Vstupní podmínky	Dostupné internetové připojení
Výstupní podmínky	Na serveru je uložena informace o rezervaci dárku
Hlavní scénář	1. Uživatel vybere kontakt, jehož seznam přání chce zobrazit
	2. Systém ověří, zda daný kontakt má v systému aktivní účet IF Kontakt má aktivní účet THEN Systém zobrazí seznam přání kontaktu ELSE Alternativní scénář 1
	3. Uživatel vybere dárek, který chce rezervovat
	4. Systém odešle informaci o rezervaci na server
Alternativní scénář 1	1. Systém zobrazí dialog s dotazem, zda má požádat kontakt o vytvoření účtu v aplikaci
	2. IF Uživatel zvolí, že chce kontakt požádat THEN Systém zašle kontaktu výzvu k vytvoření účtu

Obrázek 4.6. Scénář případu užití rezervace dárku ze seznamu přání kontaktu

Scénář případu užití	
Název případu užití	Přidání dárku do vlastního seznamu přání
Identifikační číslo případu užití	UC04
Primární aktéři	Uživatel
Sekundární aktéři	-
Vstupní podmínky	Dostupné internetové připojení
Výstupní podmínky	Na serveru je uložen seznam přání s novým dárkem
Hlavní scénář	1. Uživatel zadá název dárku do vyhledávacího pole a potvrdí
	2. Systém ověří, zda uživatel zadal alespoň dva znaky IF Uživatel zadal alespoň dva znaky. THEN Systém vytvoří dárek v databázi (pokud ještě neexistuje), přidá jej do seznamu přání, zašle tyto informace na server a zobrazí výsledek uživateli. ELSE Alternativní scénář 1
Alternativní scénář 1	1. Systém zobrazí výzvu k zadání platného názvu dárku

Obrázek 4.7. Scénář případu užití přidání dárku do vlastního seznamu přání

Scénář případu užití	
Název případu užití	Zaslání anonymního dotazu na dárek
Identifikační číslo případu užití	UC05
Primární aktéři	Uživatel
Sekundární aktéři	-
Vstupní podmínky	Dostupné internetové připojení
Výstupní podmínky	Adresátovi je odeslán anonymní e-mail s dotazem, jaký dárek si přeje, a s návrhy zadanými tazatelem
Hlavní scénář	1. Uživatel vybere kontakt, kterému chce zaslat anonymní dotaz
	2. Systém zobrazí obrazovku pro zadání navrhovaných dárků
	3. Uživatel zadá dárky, které by se kontaktu mohly líbit a potvrdí
	4. Systém odešle aktuální adresu kontaktu anonymní e-mail s dotazem, jaký dárek si přeje, a s návrhy zadanými uživatelem a zobrazí výsledek

Obrázek 4.8. Scénář případu užití zaslání anonymního dotazu

4.9 Marketingový model aplikace

Základní verze aplikace bude k dispozici zdarma, protože uživatelé Androidu obecně nejsou příliš nakloněni placeným aplikacím. Například v kontrastu s platformou iOS, kde uživatelé s placeným obsahem zpravidla nemají problém, je tato nevole uživatelů Androidu opravdu výrazná. Možná je to způsobeno obecně nižší kvalitou aplikací pro Android. Kvalitativní rozdíl oproti iOS zřejmě souvisí také s nižší cenou testovacích zařízení i licencí, která poskytuje prostor pro více vývojářů rozličnějších kvalit.

Přesto však existuje určitá skupina uživatelů, tzv. high-value users [4], kteří jsou ochotni zaplatit za kvalitnější obsah aplikace či prémiové funkce. Na tyto uživatele se při plánování monetizace aplikace musíme primárně zaměřit, identifikovat je a přizpůsobit obsah jejich potřebám. Pro tyto účely budeme využívat zdarma dostupný nástroj Google Analytics, který podrobněji rozebereme v kapitole 5.3.

Prémiové funkce placené verze

- Lze vyhledávat doporučení na dárky bez platby kreditů
- Lze vyhledávat doporučení na dárky bez zadání minulého dárku
- Další výhody zřejmě přibudou v následujících verzích aplikace - je možné, že například zaslání anonymních e-mailů bude zpoplatněno kredity, a v takovém případě by prémiový uživatel byl od této povinnosti osvobozen.

4.10 Datový model

Aplikace komunikuje s databází typu MySQL na serveru, která obsahuje dárky, tagy a asociační tabulku s váhami, která tyto entity propojuje a poskytuje doporučení na dárky. Dále jsou v ní uložena data seznamů přání a anonymních dotazů. Samotná aplikace pak disponuje lokální SQLite databází, do které jsou ukládána všechna data stažená ze serveru a navíc přátelé daného uživatele s přiřazenými tagy a položky nákupního seznamu. O technologii SQLite se podrobněji zmíníme v kapitole 5.1.5.

4.10.1 Lokální SQLite databáze

V této sekci budou popsány jednotlivé tabulky lokální SQLite databáze v mobilním zařízení, se kterými bude mobilní aplikace přímo pracovat.

Kontakt (contact)

Tabulka obsahující atributy jednotlivých kontaktů, resp. přátel, které si uživatel v průběhu používání aplikace vytvoří či importuje ze sociálních sítí. Každý kontakt bude mít povinně své unikátní *contactId*, a jméno. Dále může mít volitelně přiřazeno *userId* ze sociální sítě, pohlaví, vztah uživatele k němu, přibližný věk, e-mailovou adresu, den a měsíc narození a den a měsíc výročí. Ke kontaktu mohou být také přiřazeny položky nákupního seznamu, položky seznamu přání, odpovědi na anonymní dotazy a tagy.

Tag (tag)

Tabulka obsahující tagy přiřazené jednotlivým kontaktům. K budování tabulky dochází automaticky při specifikaci adresáta před vyhledáváním nápadů na dárky. Mezi tagy se nacházejí například také příležitosti pro nákup dárků, které se využívají při vyhledávání doporučení na dárek a při zadávání dne a měsíce narození či výročí u kontaktu. Tato tabulka nemá v lokální databázi žádný vztah k tabulce dárků, neboť jak jsme již zmínili v kapitole týkající se obecných požadavků 4.6, nechceme z bezpečnostních i praktických důvodů (vzhledem k jejich budoucí velikosti) poskytovat data sloužící pro doporučování dárků uživateli. Standardně sice nejsou data aplikací v Androidu uživateli přístupná, ale při nahrání neoficiálního obrazu operačního systému je možné k nim proniknout.

Kontakt - Tag (contact_tag)

Asociační tabulka obsahující dvojice ID kontaktů a tagů. Jedná o tagy, které byly danému kontaktu přiřazeny při vyplňování jeho zájmů 4.2 či osobních údajů.

Přání - Kontakt (wishlist_item_contact)

Asociační tabulka obsahující dvojice ID kontaktů a dárků. Slouží k ukládání přání, která si uživatel vložil do svého seznamu přání. V praxi jsou její záznamy staženy vždy ze serveru, aby byla zachována její konzistence.

Dárek (gift)

Lokální obraz tabulky dárků na serveru. Využívá se například pro ukládání dárků do nákupních seznamů pro případ nakupování bez připojení k internetu či našeptávání při vyplňování navrhovaných dárků u anonymních dotazů.

Položka nákupního seznamu (shopping_list_item)

Tabulka pro ukládání dárků, které si uživatel vložil do svého nákupního košíku. Obsahuje sloupec `bought_timestamp`, který určuje, kdy byl dárek zakoupen, a na základě toho jej v uživatelském rozhraní případně umísťuje do historie. Pokud je tento sloupec u záznamu roven nule, znamená to, že dárek dosud nebyl zakoupen.

Navrhovaný dárek (suggestion)

Tabulka pro ukládání navrhovaných dárků, které uživatel vybral před posláním anonymního dotazu. Slouží především pro zvýšení pohodlí uživatele, aby například po vypnutí a zapnutí aplikace neztratil předvyplněná data.

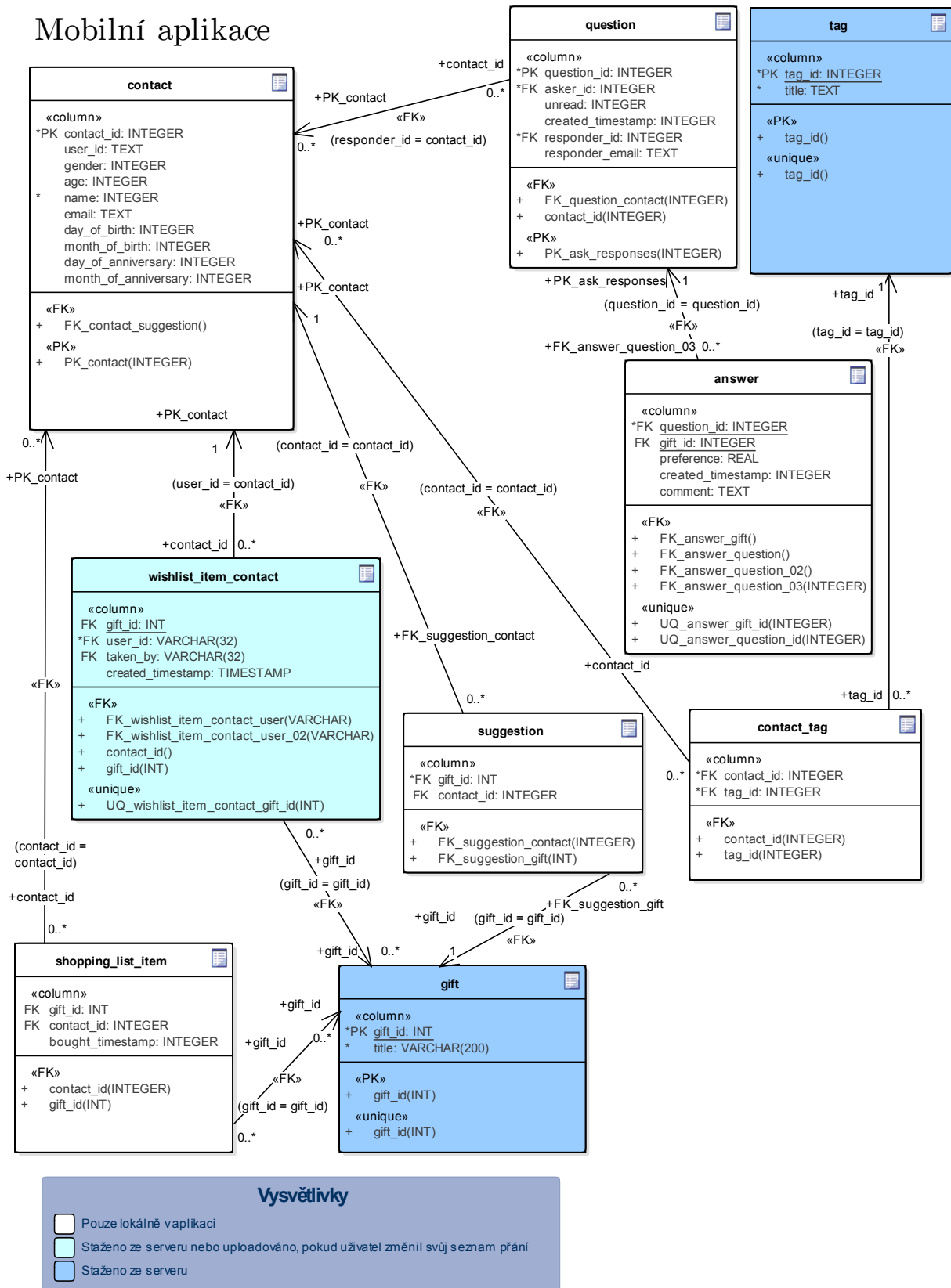
Anonymní dotaz - Odpověď (answer)

Tabulka obsahující odpovědi kontaktů na anonymní dotazy, které uživatel zaslal. Obsah tabulky vychází z odpovídajících záznamů v databázi na serveru. Každý záznam v této tabulce obsahuje ID jednoho dárku, který tazatel navrhl adresátovi při odesílání anonymního dotazu, jeho preferenci, čas vytvoření a případný textový komentář. Tabulka má přímou vazbu na tabulku otázek prostřednictvím ID otázky, ke které odpověď náleží.

Anonymní dotaz - Otázka (question)

Tabulka obsahující anonymní dotazy, které byly uživatelem položeny. Tabulka obsahuje také příznak, zda je zodpovězená otázka dosud nepřečtena, aby uživatel mohl být upozorněn na nově zodpovězené dotazy ihned po spuštění aplikace.

Mobilní aplikace



Obrázek 4.9. Datový model - Mobilní aplikace

4.10.2 MySQL databáze na serveru

Nyní, když byl vysvětlen význam jednotlivých tabulek v lokální databázi, zbývá nastínit strukturu databáze webového serveru. Většina tabulek se až na drobné odlišnosti v datových typech sloupců shoduje s tabulkami v lokální databázi. Bude proto zmíněno pouze několik hlavních odlišností:

Atribut *userId* namísto *contactId*

Databáze na serveru již nepracuje s atributem *contactId*, který byl v lokální databázi v podstatě výhradně kvůli možnosti ručně vytvářet přátele, kteří nejsou připojeni k sociální síti. Každý kontakt importovaný ze sociální sítě totiž již má nastaven atribut *userId*, na základě kterého je dále identifikován. V databázi na serveru tak *userId* představuje primární klíč uživatele, který je použit v tabulkách *wishlist_item_contact* a *question* obdobně, jako byl použit *contactId* v lokální databázi.

Z tohoto důvodu není možné používat funkcionality seznamů přání a anonymních dotazů pro ručně vytvořené přátele, kteří nemají přiřazeno *userId*. Obě tyto funkcionality jsou totiž založeny na úzké komunikaci se serverem, který zmíněný atribut vyžaduje.

Toto omezení by mohlo způsobit problémy při snaze položit anonymní dotaz člověku, který nepoužívá sociální síť, ale vlastní e-mailovou adresu. Proto byl v tabulce *question* zaveden sloupec *responder_email*, který slouží jako identifikátor adresáta dotazu v případě, že tento nemá nastaveno *userId*.

Nová tabulka *gift_tag_weight*

Jedná se o důležitou tabulku aplikace, která obsahuje trojice skládající se z ID dárku, ID tagu a váhy této asociace. Na základě záznamů v této tabulce vyhledává server nápady na dárky, jak je podrobněji popsáno v kapitole 5.4.1.

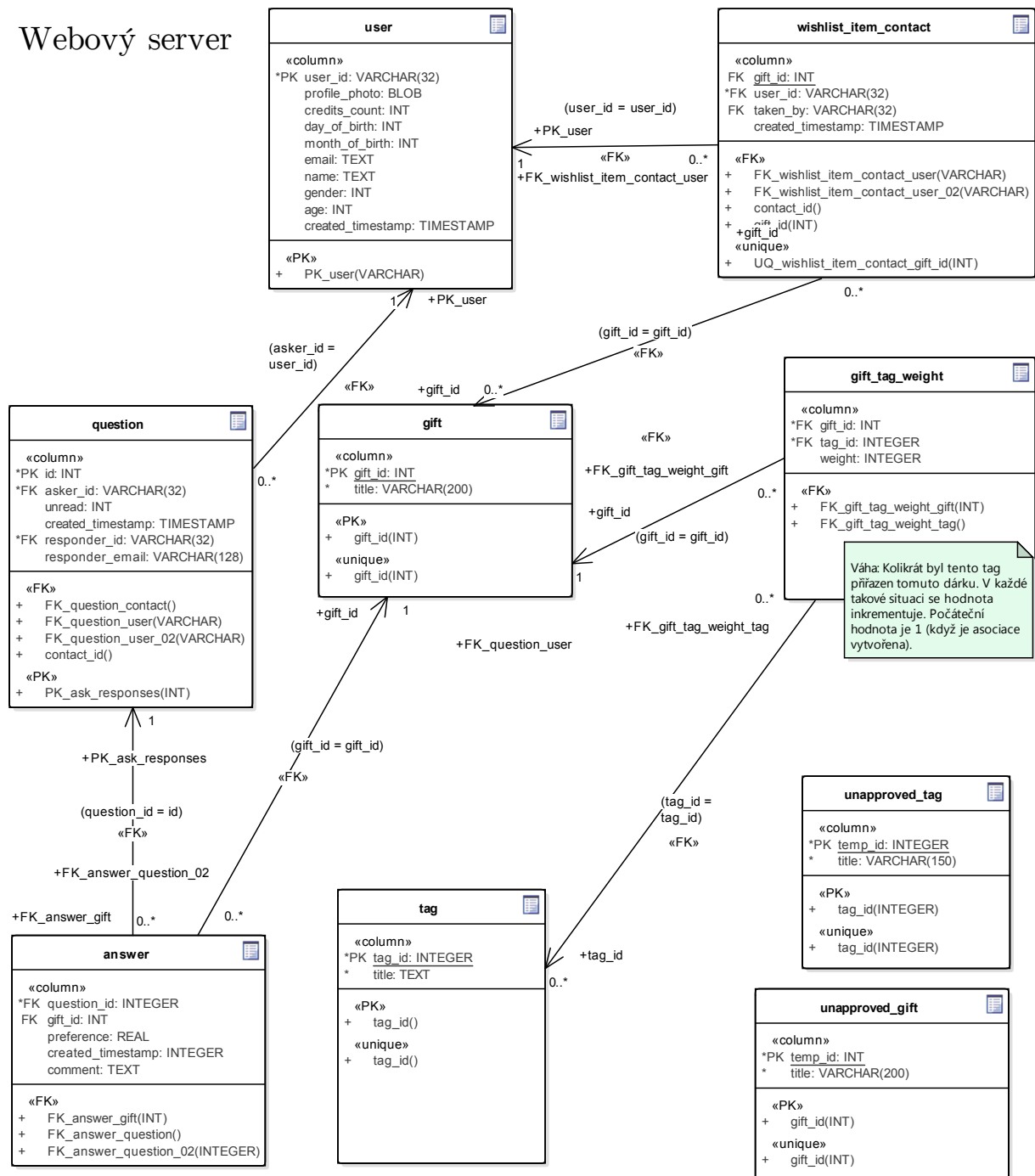
Nové tabulky *unapproved_tag* a *unapproved_gift*

Pozorný čtenář si ve schématu 4.10 mohl všimnout tabulek *unapproved_tag* a *unapproved_gift*, které nemají žádnou vazbu na ostatní tabulky. První jmenovaná slouží k ukládání tagů, které uživatel navrhl při výběru zájmů adresáta, pokud pocítoval, že mu některý zájem v seznamu chybí a stiskl tlačítko pro navržení tagu.

Druhá tabulka se naplňuje automaticky v případě, že uživatel při výběru navrhovaných dáreků u anonymního dotazu či při vytváření vlastního seznamu přání zadal jméno dárku, který se ještě v tabulce dáreků nevyskytuje.

Hlavním důvodem pro vznik těchto tabulek byla skutečnost, že se nelze spoléhat na kvalitu uživatelského vstupu natolik, abychom jej mohli ihned propagovat všem uživatelům. Mohlo by se stát, že někteří uživatelé budou do textových polí úmyslně psát vulgární či jinak nevhodné výrazy, a tak je potřeba navržené dárky a tagy nejdříve zkontrolovat. Pokud jsou v pořádku, není problém přenést je do tabulek *gift* a *tag*, které mají stejnou strukturu.

Webový server



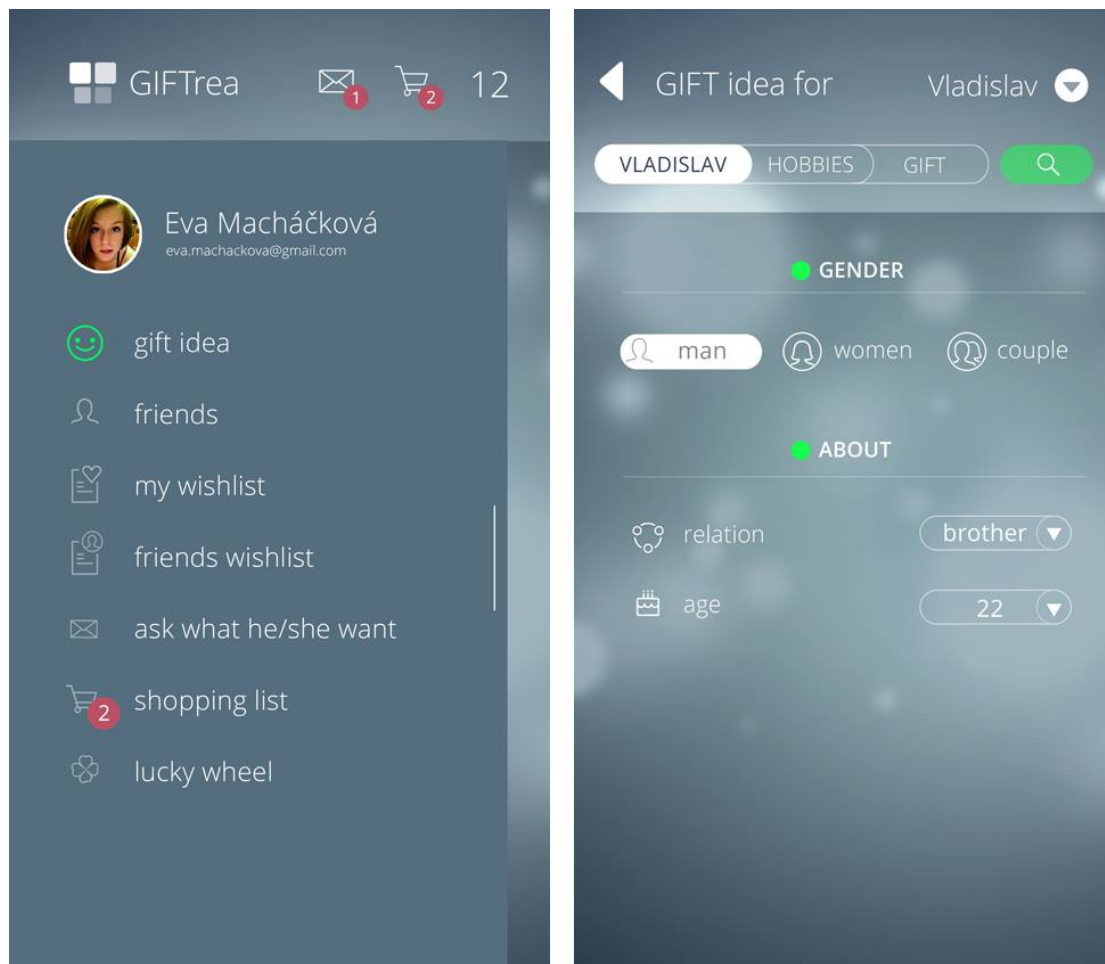
Obrázek 4.10. Datový model - Webový server

4.11 Uživatelské rozhraní

4.11.1 Grafický návrh

Autorem grafického vzhledu aplikace je Bc. Ondřej Basler. Výsledný vzhled aplikace však odpovídá grafickému návrhu pouze zčásti. Pan Basler byl v době před odevzdáním zaneprázdněn vlastními studijními povinnostmi, a tak nestihl navrhnout všechny obrazovky aplikace. Navíc řada komponent byla značně nestandardní, a tak by jejich

implementace zabrala příliš dlouhou dobu. Některé grafické prvky tak jsou dílem mé improvizace. Výsledný grafický vzhled, jehož ukázkou lze vidět na obrázku 4.11 bude implementován v příštích verzích aplikace.



Obrázek 4.11. Ukázka grafického návrhu od Bc. Ondřeje Baslera

4.11.2 Obrazovky

Aplikace z hlediska navigace odpovídá současným standardům používaným na platformě Android. Jako rozcestník v aplikaci slouží tzv. „navigation drawer“, tedy výsuvný panel s klikatelnými položkami, který lze aktivovat z libovolného místa v aplikaci tažením prstu (kromě hlouběji zanořených obrazovek) [7]. Každá položka panelu směřuje na určitou obrazovku, což umožňuje rychlou a přehlednou navigaci napříč celou aplikací. Navigation drawer je na obrázku níže označen jako „Navigační panel“.

Uživatelské rozhraní je v této kapitole graficky zachyceno v anglické verzi aplikace, zatímco dárky v databázi jsou převážně v českém jazyce. Tento problém bude v příštích verzích aplikace vyřešen způsobem, který byl již vysvětlen v kapitole 4.7.

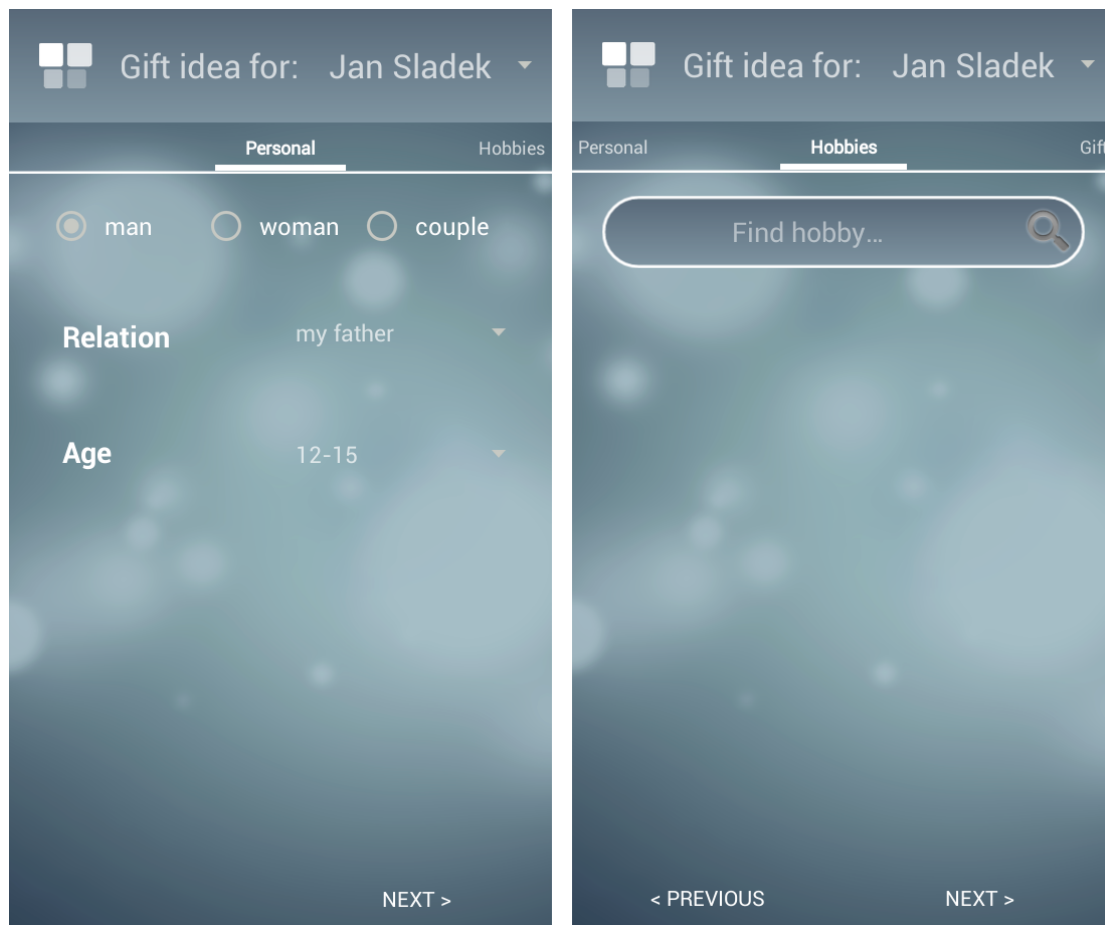
Nyní si popíšeme jednotlivé obrazovky aplikace.

O adresátovi

Tato obrazovka slouží k zadání obecných údajů o adresátovi. Jedná se o přibližný věk, vztah k dané osobě a pohlaví.

Zájmy

Na této obrazovce uživatel zadává zájmy adresáta. Má k dispozici vyhledávací pole, pomocí kterého může vyhledávat napříč všemi koníčky, které jsou momentálně v databázi aplikace. Ukázku posledních dvou zmíněných obrazovek lze vidět na obrázku 4.12.



Obrázek 4.12. Obrazovky pro zadání obecných údajů o adresátovi (vlevo) a jeho zájmů (vpravo)

O dárku

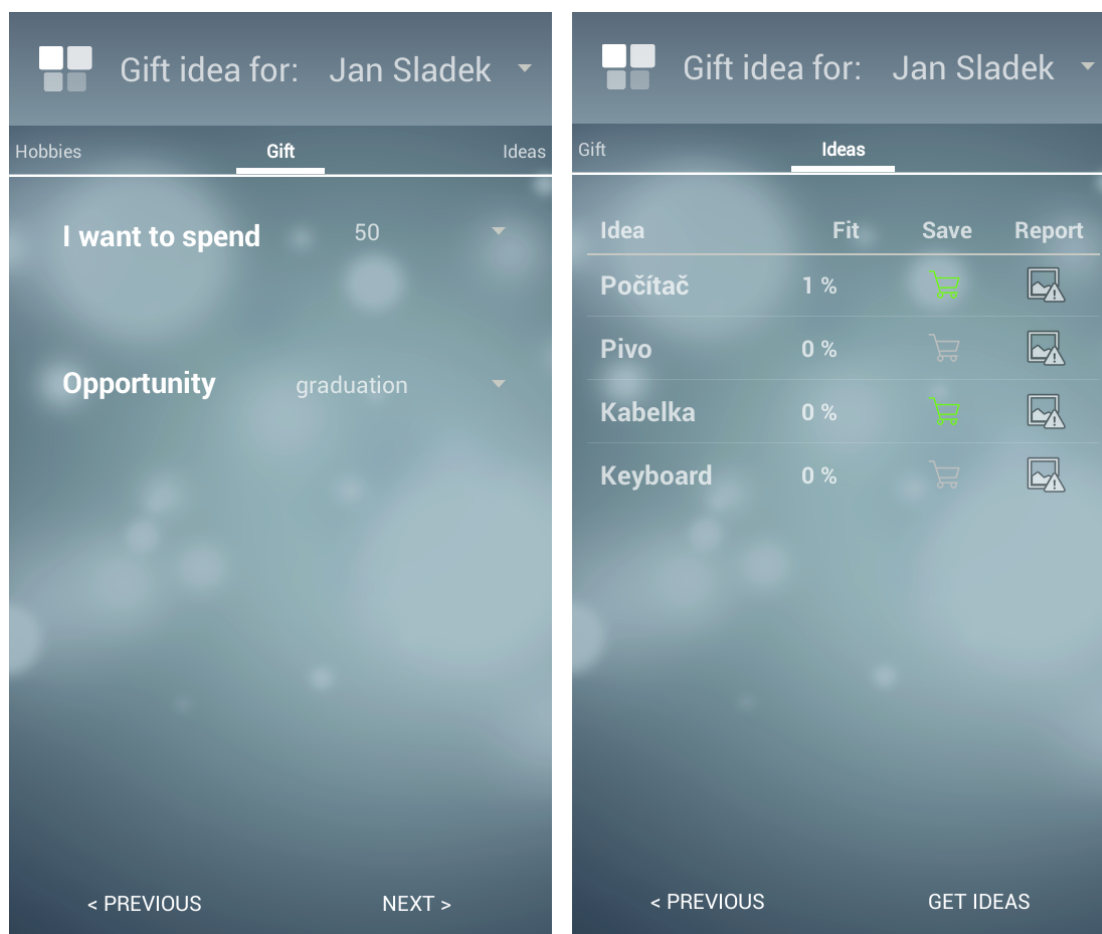
Tato obrazovka slouží k zadání cenového rozpětí dárku a příležitosti, ke které má být věnován.

Nápady

Obrazovka obsahující samotná doporučení na dárky. Každou položku seznamu lze přidat do nákupního seznamu, nebo případně nahlásit jako nevhodnou. Uživatel může vyhledání nápadů opakovat kliknutím na příslušné tlačítko. Ukázku posledních dvou zmíněných obrazovek lze vidět na obrázku 4.13.

Vlastní seznam přání

Na této obrazovce může uživatel spravovat svůj seznam přání. Pokud se dosud prostřednictvím aplikace nepřihlásil do sociální sítě, bude prostřednictvím dialogu



Obrázek 4.13. Obrazovky pro zadání údajů o dárků (vlevo) a pro zobrazení vyhledaných nápadů (vpravo)

vyzván, aby tak učinil. Pokud se nepřihlásí, nebude mu umožněno vytvářet vlastní seznam přání, protože by nebyla možná jeho pozdější identifikace. Pokud je přihlášen, může zde přidávat či odebírat dárky, které si aktuálně přeje. Ukázkou této obrazovky lze vidět na obrázku 4.14.

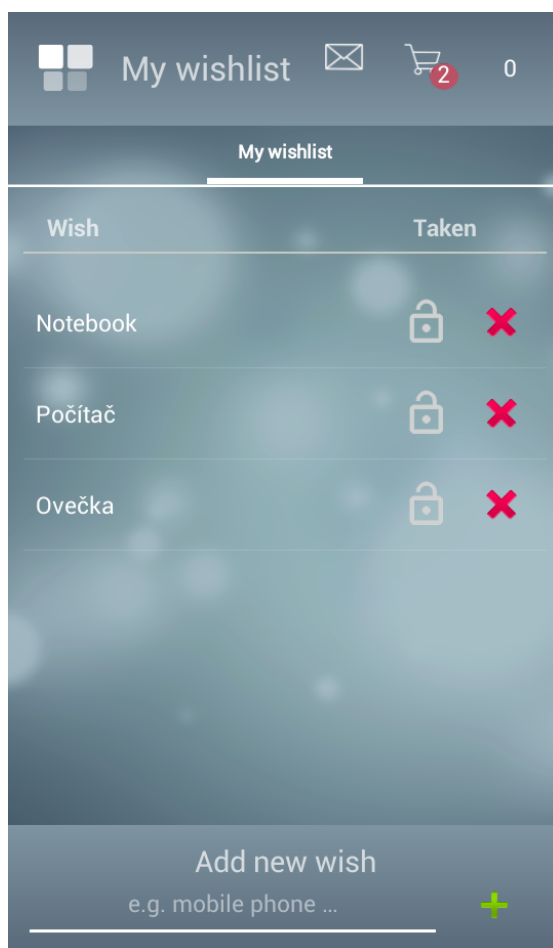
Seznam přání kontaktu

Na této obrazovce si uživatel může zobrazit seznamy přání svých preferovaných přátel. V seznamu si poté může rezervovat dosud neobsazené dárky a přidat si je do svého nákupního seznamu. Stejně tak může jedním kliknutím zrušit rezervaci či dárky odebrat z nákupního seznamu. Ukázkou této obrazovky lze vidět na obrázku 4.15.

Anonymní dotaz na dárek - Zaslání dotazu

Na této obrazovce může uživatel zaslat anonymní e-mailový dotaz svým preferovaným přátelům. Předtím musí vybrat přítele, kterému chce zprávu poslat, zadat jeho e-mailovou adresu, pokud ji aplikace nemá staženu ze sociální sítě a pokud ji tomuto kontaktu již dříve nepřiradil, a poté navrhnout alespoň 3 návrhy dárků pro adresáta.

Anonymní dotaz na dárek - Odpovědi



Obrázek 4.14. Vlastní seznam přání

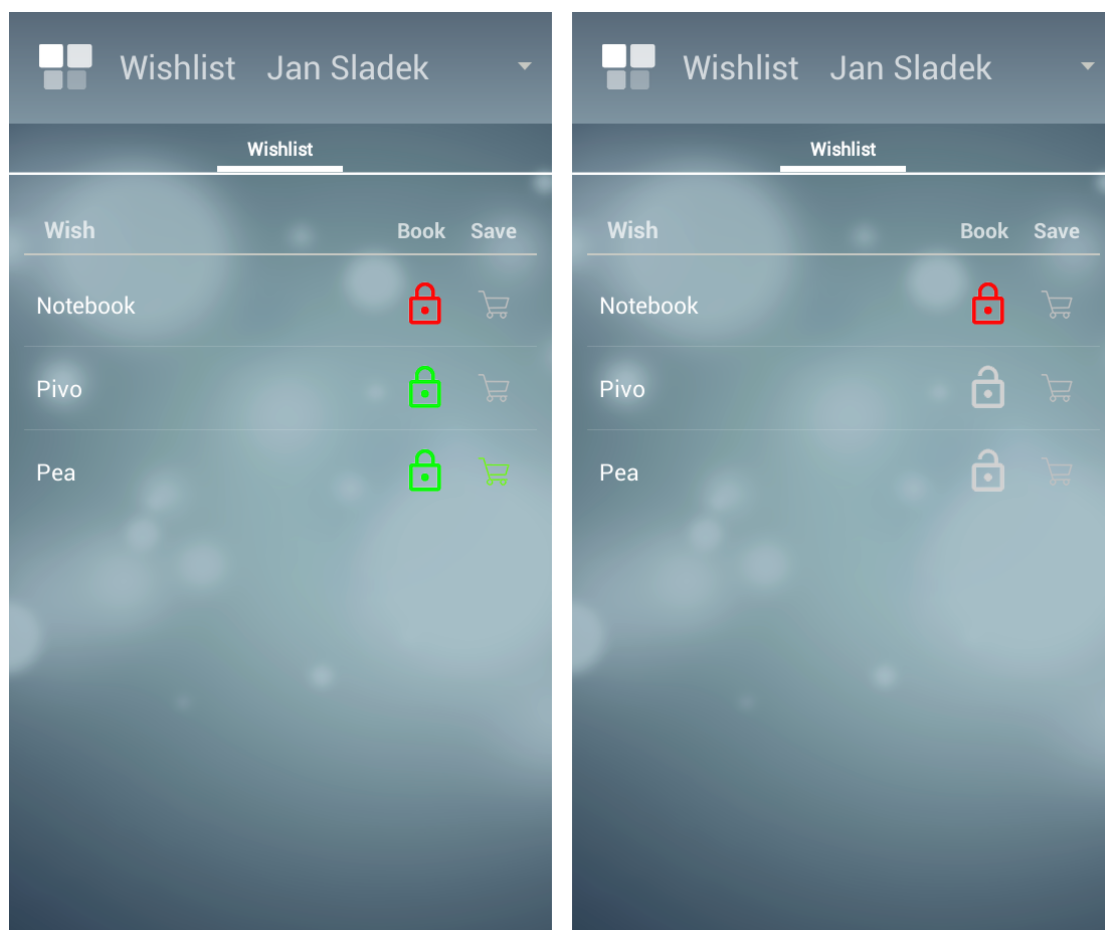
Na této obrazovce najde uživatel odpovědi na anonymní dotazy, které předtím zaslal aktuálně vybranému příteli. Počet nepřechtených odpovědí se zobrazuje také v horní liště aplikace. Ukázku posledních dvou zmíněných obrazovek lze vidět na obrázku 4.16.

Nákupní seznam - aktuální

Obrazovka obsahující seznam dárců, které si uživatel vybral při vyhledávání doporučení, procházení seznamů přání jiných uživatelů či odpovědí na anonymní dotazy. V seznamu je u každého dárku uvedeno jméno adresáta, kterému má být zakoupen a možnost označit jej jako koupený.

Nákupní seznam - historie

Obrazovka obsahující záznamy o dříve koupených dárcích. Umožňuje uživateli udržovat si přehled o tomu, komu již koupil jaké dárky a zamezit tak duplicitám. U každé položky je uveden adresát a datum zakoupení. Uživatel zde také může přesunovat dárky z historie zpět do aktuálního seznamu, pokud například dárek označil jako koupený omylem. Ukázku posledních dvou zmíněných obrazovek lze vidět na obrázku 4.17.



Obrázek 4.15. Seznam přání kontaktu s ukázkou rezervování dárků a přidávání dárků do nákupního seznamu

Preferovaní přátelé

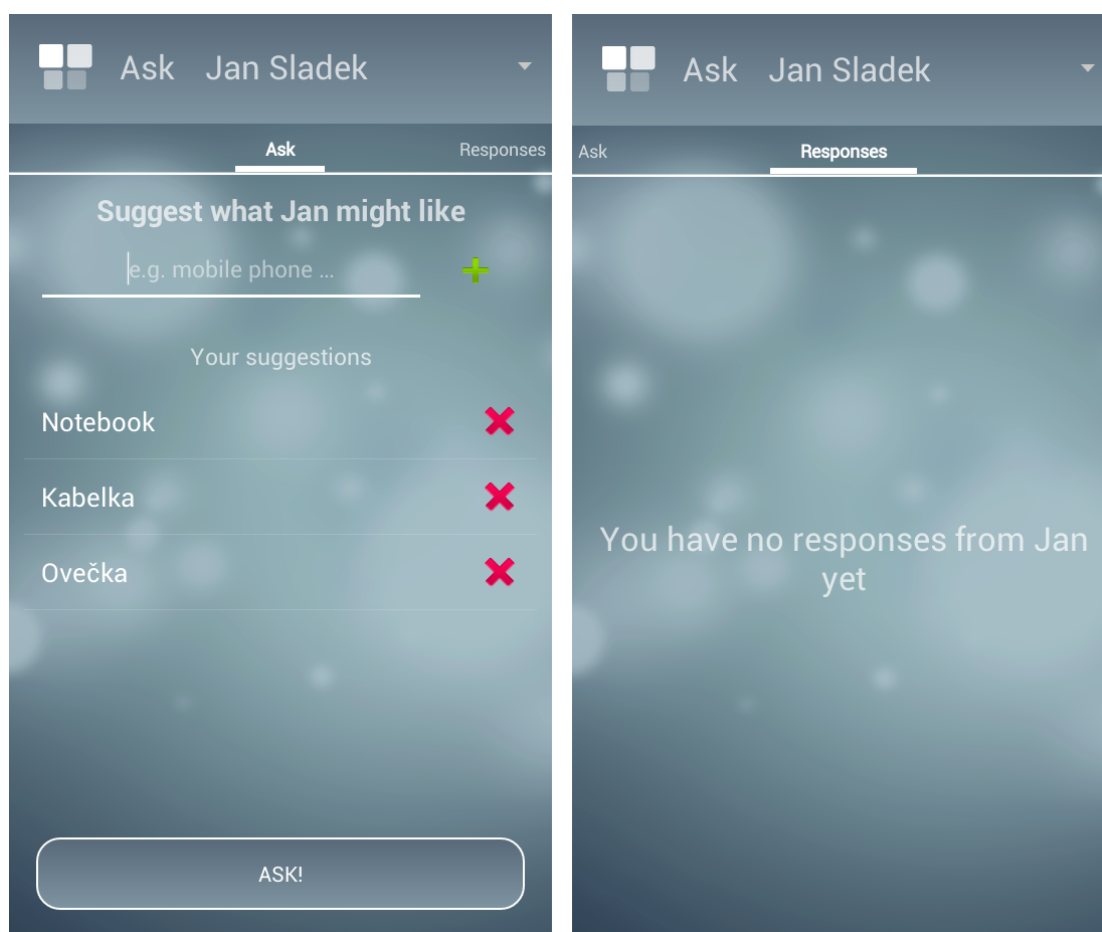
Na této obrazovce se nachází seznam přátel, které uživatel označil jako preferované. Přímo z této obrazovky lze přejít na vyhledávání doporučení na dárek pro vybraný kontakt, na jeho seznam přání či na zaslání anonymního dotazu kontaktu. Dále je možné kontakty přesunout do Ostatních.

Ostatní přátelé

Na této obrazovce se nachází seznam přátel, které si uživatel v průběhu používání aplikace vytvořil či importoval a dosud je neoznačil jako preferované. Na této obrazovce je možné smazat přátele či je přesunout do Preferovaných. Při mazání přátel je uživatel vyzván dialogem k potvrzení, že chce operaci opravdu provést. Po smazání kontaktu ztratí také jakékoli atributy, které mu dosud přiřadil. Ukázkou posledních dvou zmíněných obrazovek lze vidět na obrázku 4.18.

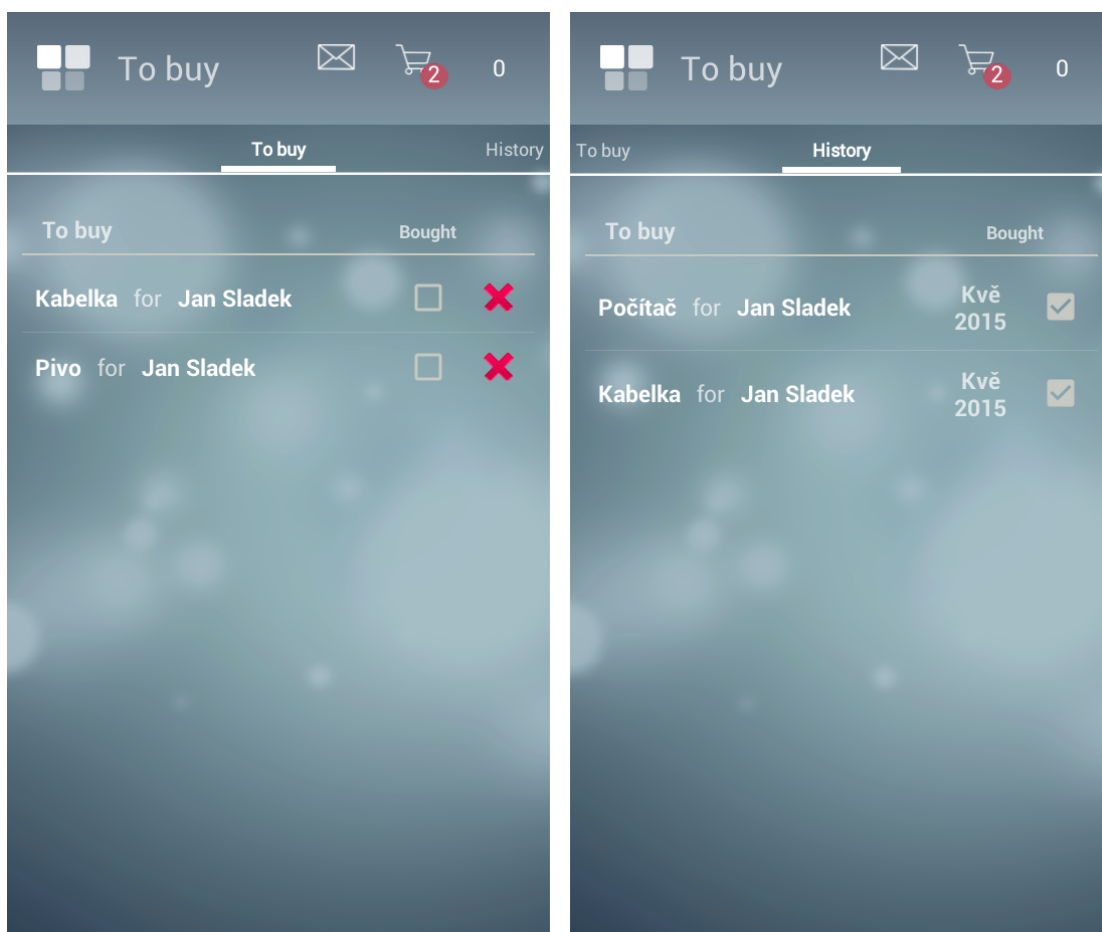
Dialogy

Většina mobilních aplikací používá dialogy pro situace, kdy je potřeba okamžitá interakce s uživatelem. Jedná se například o potvrzení určité akce či zobrazení výstrahy při vzniku problému. Aplikace vyvíjená v rámci této bakalářské práce v tomto ohledu není výjimkou a lze v ní najít řadu potvrzovacích a varovných dialogů, ale také dialogy

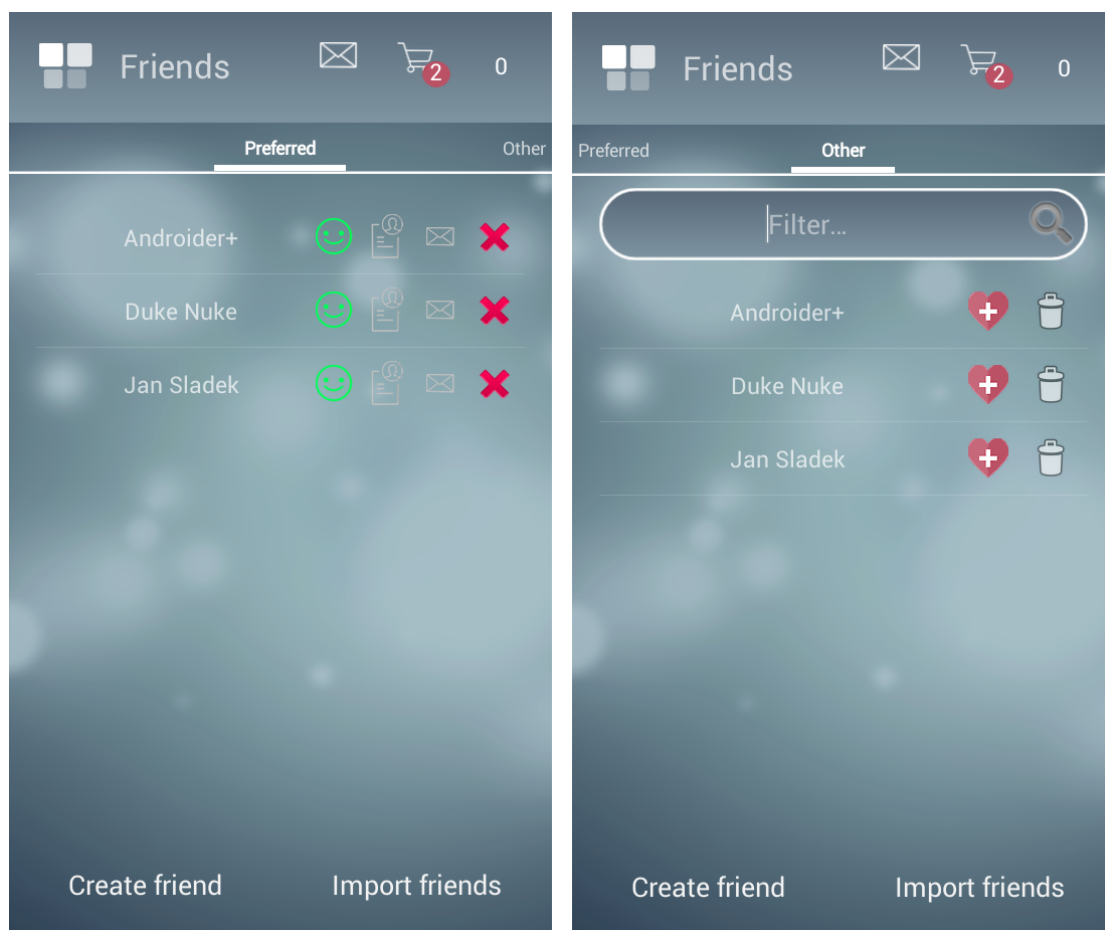


Obrázek 4.16. Zaslání anonymního dotazu (vlevo) a zobrazení odpovědí (vpravo)

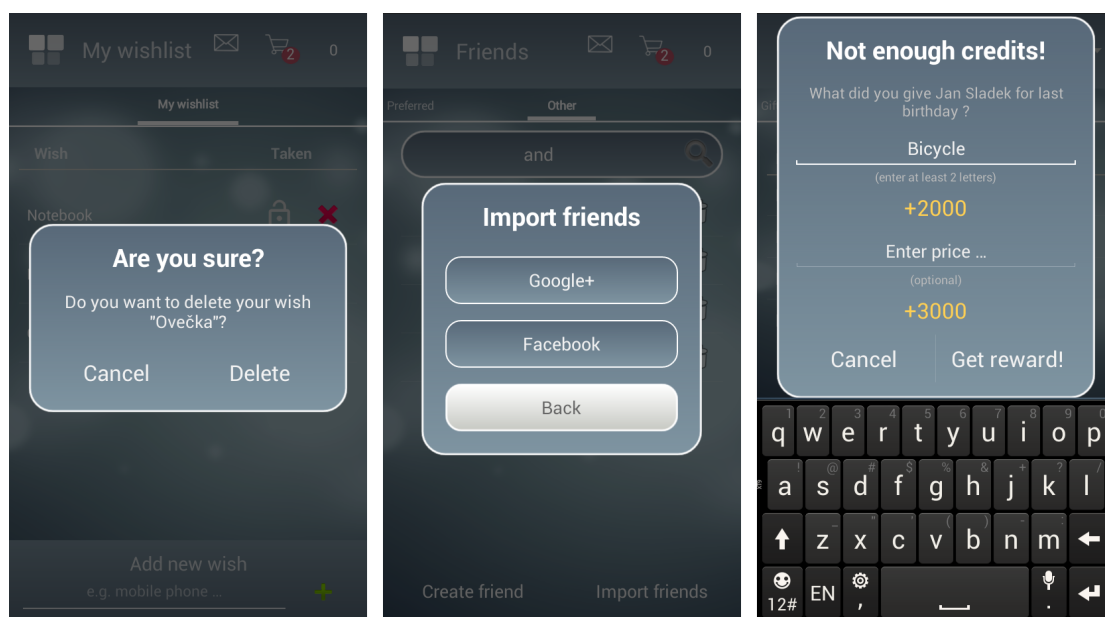
pro zadání dříve věnovaného dárku či importování přátel. Jejich grafickou podobu si lze prohlédnout na obrázku 4.19.



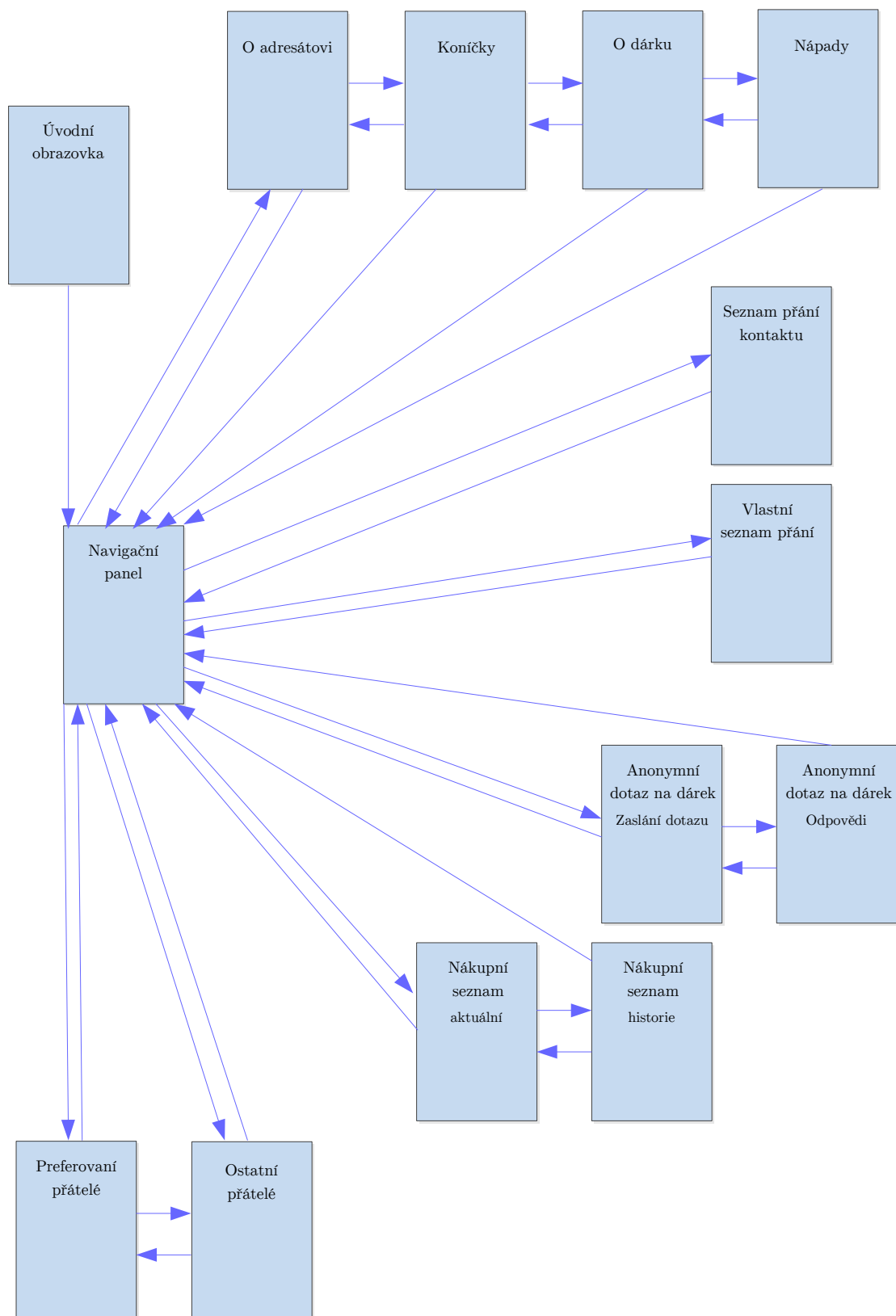
Obrázek 4.17. Aktuální nákupní seznam (vlevo) a historie (vpravo)



Obrázek 4.18. Preferovaní přátelé (vlevo) a ostatní přátelé (vpravo)



Obrázek 4.19. Dialogy: (zleva) Potvrzovací dialog při odstranění přání, dialog pro importování přátel a dialog pro zadání dříve věnovaného dárku



Obrázek 4.20. Mapa obrazovek aplikace

Kapitola 5

Implementace

Tato kapitola se zaměřuje na technická specifika platformy Android, použité technologie a vlastní vývoj aplikace.

5.1 Použité technologie

Při implementaci aplikace byla využita především standardní, osvědčená řešení, která díky široké uživatelské základně a rozsáhlé dokumentaci umožňují bezproblémový vývoj a podporu při řešení případných problémů.

5.1.1 Android

Pro vývoj aplikace byl použit operační systém Android, jehož základem je linuxové jádro. Systém je distribuován na základě open source licence Apache 2.0, díky čemuž jsou její zdrojové kódy zdarma k dispozici [8]. U zrodu operačního systému stála mladá firma Android Inc., kterou založili Andy Rubin, Rich Miner, Nick Sears a Chris White. V roce 2005 byl Android Inc. odkoupen softwarovým gigantem Google, který začal na vývoji platformy intenzivně pracovat. První verze systému Android byla vydána v listopadu 2007. O necelý rok později pak bylo vydáno první zařízení se systémem Android, které neslo označení T-Mobile G1.

První verze SDK trpěly řadou nedostatků, které byly v průběhu vývoje odstraňovány. Jak je u firmy Google zvykem, nové verze Androidu byly vydávány poměrně často a hlavním zdrojem zpětné vazby byli samotní uživatelé. Například mezi vydáním verze 1.6 (API 4, Donut) a 2.0 (API 5, Eclair) uplynul pouhý jeden měsíc [9]. Díky tak častému vydávání nových verzí dnes Android představuje uživatelsky přívětivý, relativně stabilní systém, který poskytuje velmi dobrý základ pro tvorbu kreativních aplikací. V této aplikaci budeme využívat především standardní grafické prvky Androidu, neboť pokusy o extravaganci v uživatelském rozhraní jen zřídka překonají mnohaletou práci profesionálů a spotřebují často obrovské množství času. Použitím vestavěných komponent je navíc zajištěn konzistentní vzhled obrazovek napříč zařízeními s různými velikostmi a rozlišeními displeje.

5.1.2 Použité externí knihovny

Výchozí sada grafických prvků při implementaci ne vždy postačovala, a tak bylo využito také několika knihovných projektů pocházejících ze samotné vývojářské komunity Androidu. Autoři své knihovny často poskytují k dispozici jako open-source, čímž také umožňují dalším vývojářům, aby je dále rozvíjeli a udržovali aktuální vzhledem k nejnovějším trendům.

ViewPagerIndicator

Až do verze Android API 21 byly nedílnou součástí většiny aplikací tzv. taby [10], tedy záložky v horní části obrazovky, mezi kterými lze přepínat buďto kliknutím, nebo

přetažením prstu do strany. Tyto záložky umožnily rychlou navigaci mezi větším počtem obrazovek a vývojáři si na jejich používání rychle zvykli. Po vydání API 21 však přišla nepříjemná změna v podobě označení celé knihovny ActionBar, obsahující také záložky, jako `deprecated` [11], tedy kód nedoporučený k dalšímu používání, jehož zachování a funkčnost v budoucnu nejsou garantovány. Návrh uživatelského rozhraní aplikace vyvíjené v rámci této bakalářské práce však s použitím záložek počítal, a tak bylo nutné najít přijatelnou náhradu. Knihoven simulujících funkci záložek existuje celá řada, ale nakonec byl zvolen `ViewPagerIndicator`¹⁾. Jedná se o jednu z nejpoužívanějších knihoven tohoto typu, která evokuje důvěryhodnost nejen počtem vývojářů, kteří k projektu přispívají, ale také jménem hlavního autora, kterým je Jake Wharton. Právě Wharton patří mezi významné osobnosti vývojářské komunity Androidu. Vytvořil například celosvětově používanou knihovnu `ButterKnife`²⁾, umožňující výraznou úsporu a zpřehlednění zdrojového kódu při interakci grafických prvků jednotlivých obrazovek s výkonnou logikou aplikace.

EventBus

Dříve či později každý Android vývojář narazí na nutnost vykonávat některé výpočetně náročné operace mimo hlavní vlákno, které vykresluje uživatelské rozhraní. V této bakalářské práci jsem zpočátku prováděl většinu operací na hlavním vlákně, zejména pro účely přehlednosti kódu v počátcích vývoje. Především databázové dotazy však brzy začaly způsobovat nepříjemné zpomalení uživatelského rozhraní a čekání při přechodech mezi obrazovkami by mohlo budoucí uživatele obtěžovat.

Pro realizaci vícevláknového zpracování úloh poskytuje Android vývojářsky přítivětivou knihovnu `AsyncTask`, o které lze získat podrobnější informace v [12]. Při jejím použití stačí pouze definovat úkoly prováděné mimo hlavní vlákno a ve zpětném volání pak zpracovat výsledek. O vytvoření nového vlákna a jeho životní cyklus se postará samotný systém. Problém však může nastat například v případě, kdy během vykonávání `AsyncTasku` uživatel otočí obrazovku zařízení [13]. V takové situaci dojde ke zrušení současné tzv. aktivity³⁾ a vytvoření nové aktivity s požadovanou konfigurací. Více o aktivitách a jejich životním cyklu se lze dočíst v [14]. Zrušená aktivita nemůže být uvolněna z paměti, protože obsahuje referenci na běžící `AsyncTask`, pokud je tento implementován jako vnitřní třída [15]. Pokud takto zůstane běžet několik dlouhých asynchronních operací, můžou neuvolnitelné aktivity výrazně zpomalit aplikaci a v krajním případě vyžadovat její násilné ukončení. Z tohoto důvodu byl vyvinut koncept `EventBus`, který slouží k doručování událostí napříč aplikací. Při použití `EventBusu` vyčleníme `AsyncTask` do samostatné třídy a jeho výsledek rozešleme všem odběratelům událostí, kteří jsou momentálně registrováni. Pokud rušenou aktivitu vyjmeme z množiny odběratelů, nebude již na ni existovat reference a systém může příslušné prostředky uvolnit. Další výhodou `EventBusu` je pak možnost rozesílat události globálně, aniž by bylo nutné řešit doručování výsledků do správné aktivity.

5.1.3 REST

Pro přenos dat mezi mobilní aplikací a webovým serverem byla použita REST architektura [16], která patřila mezi základní požadavky na tuto bakalářskou práci. Mezi

¹⁾ <https://github.com/JakeWharton/ViewPagerIndicator>

²⁾ <http://jakewharton.github.io/butterknife>

³⁾ Aktivita v Androidu představuje objekt, který uživateli umožňuje dosáhnout určitého konkrétního cíle. Většina aktivit poskytuje uživateli možnost interakce formou okna s grafickými prvky. Zjednodušeně řečeno, každá obrazovka mobilní aplikace má svou aktivitu.

hlavní rysy RESTful systémů patří především bezstavovost a komunikace pomocí požadavků v protokolu HTTP [17]. HTTP, neboli **H**yper**T**ext **T**ransfer **P**rotocol, představuje skupinu standardů sloužících k předávání informací v prostředí Internetu. Mezi hlavní principy HTTP patří používání portů 80, 8008 či 8080 a standardizované stavové kódy posílané v odpovědích, jako je například 200, 404 či 500. Pro komunikaci se využívají metody požadavků GET, POST, PUT, DELETE a HEAD, kdy v praxi se setkáme především s prvními dvěma zmíněnými. REST architektura v tomto ohledu není výjimkou a právě GET a POST požadavky představují stěžejní způsob výměny informací v systémech, které ji implementují.

Každá ze zmíněných metod má však z hlediska REST API svá specifika a doporučené využití. GET požadavek neobsahuje žádné tělo a jeho parametry jsou předávány přímo v URI. URI představuje řetězec znaků, které jednoznačně identifikují určitý zdroj na serveru, GET požadavky tedy slouží výhradně k získání informací z určitého umístění na serveru. POST požadavky pak umožňují posílat parametry ve svém těle, například v notaci JSON, o které se zmíníme podrobněji v kapitole 5.1.4. Díky tomu je možné poslat velké množství dat v různých formátech. Důležitým faktem je, že u POST požadavku narozdíl od PUT požadavku nevíme, kam na serveru budou posílaná data na serveru umístěna. Tuto informaci nám zpravidla sdělí server v odpovědi. Podrobnější srovnání PUT a POST požadavků lze najít v [18]. PUT požadavky pak slouží k modifikaci či vložení jednoho konkrétního prvku, jehož identifikátor uvádíme v URI, jako je tomu u GET požadavku, a jehož umístění na serveru tedy předem známe. Narozdíl od GET může PUT obsahovat tělo, kde předáváme nové informace o upravovaném objektu. A nakonec požadavky DELETE zajišťují smazání konkrétního prvku, který identifikujeme pomocí parametrů v URI. DELETE, stejně jako GET, nemá žádné tělo, ve kterém bychom předali další informace, neboť pro operaci smazání postačuje pouze identifikátor příslušného objektu.

Požadavky webového API této bakalářské práce využívají REST metody dle výše uvedených pravidel. Po většinu času na tvorbě API požadavků pracoval pan Bc. Roman Kozák, ale později byl již příliš zaneprázdněn svými studijními povinnostmi. Požádal jsem tedy o pomoc svého bratra, Bc. Tomáše Sládka, který implementaci požadavků dokončil. Pro realizaci jednotlivých REST metod byl využit PHP framework Slim¹).

Pro volání požadavků REST API ze strany mobilní aplikace byla v této bakalářské práci využita třída `DefaultHttpClient` z knihovny `Apache HttpComponents`, která je v Androidu standardně integrována. Každá REST metoda v této knihovně má svou speciální třídu, tedy `HttpGet`, `HttpPost`, `HttpPut` a `HttpDelete` [19]. Tyto třídy bohužel nemají žádného jednotného předka s vhodným rozhraním, což mi způsobilo problémy ve snaze generalizovat a abstrahovat proces komunikace s API ve zdrojovém kódu. Většina veřejných metod se totiž v těchto čtyřech třídách opakuje, ale přitom není možné je volat jednotně na instanci typu předka. Nakonec se mi ale podařilo dosáhnout přijatelného výsledku a omezit tak výskyt duplicitního kódu i v této části implementace aplikace.

Dalším důležitým rysem REST architektury je čitelnost URI jednotlivých zdrojů, ke kterým klient přistupuje. URI se skládá z následujících částí [20]:

- Schéma (scheme) - http či https, podle použitého protokolu
- Znaky `://`
- Autorita (authority) - doména, na které je umístěn webový server, v případě této bakalářské práce `giftrea.com`

¹) <http://www.slimframework.com/>

- Cesta (path) - posloupnost identifikátorů umístění souborů na serveru oddělených znakem lomítka. Jako celek představují jednoznačnou cestu k určitému souboru.
- Znak `?`
- Dotaz (query) - posloupnost dvojic *klíč = hodnota* oddělených apostrofem, na základě kterých může klient sdělit serveru upřesňující informace k dotazovaným datům
- (volitelně) Znak `#`
- (volitelně) Fragment - řetězec znaků, který odkazuje na určitou konkrétní část celku, který byl identifikován předchozími částmi URI

Příkladem URI z aplikace vyvíjené v rámci této bakalářské práce, které splňuje výše uvedenou strukturu, je:

```
http://giftrea.com/api/ask?asker_id=1&responder_id=2
```

Zde je důležité zdůraznit, že URI by nemělo končit nadbytečným znakem lomítka. Většina webových serverů nebude tento znak při vyhodnocování výsledné cesty zohledňovat, ale každý znak URI může nést určitou hodnotu, která se projeví ve výsledné lokalizaci zdroje. Proto je vhodné všechny znaky, které nijak nepřispívají k nalezení zdroje, z výsledného URI odstranit.

Webové API této aplikace se při vytváření URI drží výše zmíněných pravidel. Jedinou výjimkou jsou GET požadavky, které se dotazují na určitou množinu objektů prostřednictvím posloupnosti jejich ID. Pokud bychom v takovém případě uvedli všechna ID jako posloupnosti dvojic klíč = hodnota, pak by se klíč ve všech prvcích posloupnosti opakoval, což by vedlo k vysoké míře redundance v URI. Proto zde oddělujeme jednotlivé hodnoty čárkou a jejich klíč lze odvodit z cesty.

5.1.4 JSON

Jako přenosový formát byl vybrán JSON, který ve srovnání s alternativním XML působí jednodušeji a navíc jsem se s ním již minulosti setkal. Standard JSON představuje podmnožinu programovacího jazyka JavaScript [21]. JSON je velmi úsporný formát pro výměnu dat, jehož výhodou je především snadná čitelnost nejen pro člověka, ale i počítač při strojovém zpracování. Dalším důvodem jeho výběru byly také vestavěné funkce pro jeho čtení a úpravy v jazyce PHP i Java. Příjemnou zprávou je také nativní podpora formátu JSON ve většině prohlížečů, které jej automaticky formátují do přehledné a uspořádané podoby.

Notace JSON je založena na šesti základních datových typech [22]:

- Objekt - JSONObject
- Pole - JSONArray
- Textový řetězec - JSONString
- Číselná hodnota - JSONNumber
- Logická hodnota (true nebo false) - JSONBoolean
- Null - JSONNull

Objekty v JSON se ohraničují složenými závorkami, pole pak hranatými závorkami. Textové řetězce se uzavírají do uvozovek, nikoli však apostrofů, ačkoli jiné platformy toto nahrazení často tolerují, byť často s určitou změnou v metodice vyhodnocování výrazu mezi nimi. Číselné hodnoty pak zapíšeme zcela standardně, bez uvozovek. JSONBoolean se zapisuje také bez uvozovek a může obsahovat výhradně hodnotu *true* nebo *false*. JSONNull se zapíše jako hodnota *null*, opět bez uvozovek. Více informací o datových typech a syntaxi notace JSON lze najít v [21].

V této aplikaci využíváme zmíněné formáty standardním způsobem s výjimkou logických hodnot. Výchozí logické hodnoty notace totiž nahrazujeme číselnou hodnotou *0* pro *false* a *1* pro *true*, čímž dosahujeme určitého snížení objemu posílaných dat, a tedy zátěže webového serveru.

Objekty a pole lze vkládat vzájemně do sebe, zatímco ostatní datové typy mohou být pouze součástí objektu nebo pole. V následujícím úryvku kódu vidíme JSONObject, který obsahuje nejprve dva atributy *asker_id* a *responder_id* typu JSONString, dále atribut *created_timestamp* typu JSONNumber a nakonec atribut *gift_id* typu JSONArray, který dále obsahuje prvky typu JSONNumber.

```
{
  "asker_id" : "451324565791",
  "responder_id" : "134787978654",
  "created_timestamp" : 1431344361713,
  "gift_id" : [1, 2, 3]
}
```

Abychom zasadili uvedený úryvek kódu do kontextu této bakalářské práce, jedná se o data, která se posílají z aplikace na server při odesílání anonymního dotazu, o kterém jsme se již zmínili v kapitole 4.4. Atribut *asker_id* zde představuje identifikátor uživatele, který se ptá, zatímco *responder_id* identifikuje adresáta dotazu. *Created_timestamp* pak označuje přesný čas odeslání dotazu v milisekundách. Pole *gift_id* obsahuje identifikátory dárků, které odesílatel navrhl adresátovi a žádá o sdělení preference každého z nich.

Datové typy z jiných programovacích jazyků, které JSON nepodporuje, přenášíme zpravidla jako textový řetězec s předem definovaným formátem, aby jej druhá strana mohla správně rozpoznat. Při zpracování JSON dat je velmi důležité, aby při zjištění jakékoli chyby formátování bylo zpracovávání ihned ukončeno a došlo k ohlášení problému, například formou výpisu do konzole. Pokud by totiž aplikace pokračovala ve čtení, došlo by velmi pravděpodobně k získání a uložení neplatných dat. Právě striktní kontrola správného formátování je určitou daní za jednoduchost notace, ale na druhou stranu, ani objektové jazyky neposkytují programátorovi prostor pro chyby a nepřesnosti v kódu. Díky minimalismu notace jsou v této bakalářské práci využity v podstatě všechny zmíněné datové typy, které jsou v JSON dostupné.

Již naznačenou možnost vkládání objektů do pole v notaci JSON ukazují následující úryvek kódu, který obsahuje pole dvou objektů s atributy shodných typů.

```
[
  {
    "gift_id" : 2,
    "weight_sum" : 47,
    "relevance" : 1
  },
  {
    "gift_id" : 2,
    "weight_sum" : 47,
    "relevance" : 1
  }
]
```

Zajímavostí je, jak výrazně kontrastní je práce s kompaktní JSON notací ve srovnání s objektově orientovaným, silně typovým jazykem Java¹), který byl použit pro implementaci mobilní aplikace. Zpracovávání dat ve formátu JSON se tak ve vývojovém prostředí operačního systému Android stává určitou izolovanou zónou, ve které je nutné velmi pečlivě dbát na kvalitu a správnost kódu, neboť kontrola funkcionality zpravidla spočívá až v následné reálné komunikaci se serverem.

5.1.5 SQLite

V předchozí kapitole byl rozebrán způsob komunikace mezi mobilní aplikací a serverem a použitý formát pro přenos dat. Nyní se zaměříme na samotné ukládání dat do úložiště v mobilním zařízení. Pro tyto účely byl využit databázový engine SQLite, který je standardně instalován v zařízeních s operačním systémem Android. SQLite představuje transakční databázový engine využívající jazyk SQL. Na rozdíl od většiny databázových systémů neběží SQLite jako samostatný proces na serveru, se kterým by aplikace a služby komunikovaly prostřednictvím Internetu. Naopak, proces pro přístup k databázi je zde spuštěn přímo v zařízení a nevyžaduje síťovou komunikaci. Díky tomuto řešení není potřeba zajišťovat a konfigurovat server, postačuje pouze oprávnění k zápisu na datové úložiště zařízení. Nevýhodou pak mohou být bezpečnostní rizika, kdy chyba v aplikaci může způsobit ztrátu dat [24]. Android navíc standardně umožňuje smazat uživatelská data aplikací, a tak se nelze na jejich zachování během používání aplikace spoléhat. Naopak oblastí, kde SQLite vyniká, je vysoká paměťová efektivita, díky které může aplikace v systému Android pracovat s databází jako celkem a není nutné vyčleňovat pouze některé aktivní části kvůli úspoře paměťových nároků. Především u levnějších zařízení s nižší kapacitou operační paměti je toto kritérium velmi důležité. Samotný databázový engine SQLite navíc zabírá pouhých 256 kB na paměťovém médiu, a tak není problém jej aplikovat ani v zařízeních s omezenou kapacitou úložiště [25].

Výhodou SQLite engine je také vysoká spolehlivost zajištěná pečlivým testováním každé vydané verze. Jeho transakce jsou vždy atomické, konzistentní, izolované a trvalé (ACID). Právě princip ACID představuje jeden ze základních konceptů databázové teorie. **Atomicita** reflektuje pravidlo „všechno nebo nic“, tedy že každá úprava databáze musí proběhnout buď celá, nebo je anulována. Tento princip je zásadní především pro situace, kdy dojde během zápisu k selhání zařízení či programu a v databázi by zůstala neúplná data. **Konzistence** pak zajišťuje, aby do databáze byla zapsána pouze platná data. Pokud by se program pokusil o zápis dat, která například porušují integritní omezení, bude celá transakce zrušena. Připomeňme, že transakce představuje sekvenci operací, které jsou prováděny na databázi. V případě selhání některé operace je celá transakce zrušena, a výsledky již provedených operací se anulují [26]. **Izolace** databáze znamená, že v jednu chvíli může probíhat pouze jedna transakce. Tím se předejde situacím, kdy například jeden uživatel odešle platební příkaz, zatímco jiný uživatel s přístupem ke stejnému bankovnímu účtu provede další příkaz dříve, než se transakce prvního uživatele dokončí. Tím by došlo k nekonzistenci dat a vzniku záporného zůstatku na účtu. A konečně **trvalost** databáze zajišťuje, že všechny transakce v databázi musí být ukládány pro případ jejich replikace například po selhání úložiště [27].

¹) Podrobnější informace o objektově orientovaném programování a typovosti jazyka Java lze najít v [23] a v [12]

5.2 Serverová strana

Jak bylo již uvedeno výše, serverovou stranu aplikace v podobě REST API implementovali Bc. Roman Kozák a později Bc. Tomáš Sládek. Já, autor této bakalářské práce, jsem toto rozhraní navrhoval, především pak databázi serverové strany a požadavky, prostřednictvím kterých aplikace se serverem komunikuje. Všechny tyto požadavky včetně použitých metod, parametrů, formátu odpovědí a textového popisu funkcionality lze najít v 5.1 a 5.2. U požadavků, které nevracejí žádná data, se úspěšnost provedení indikuje přítomností JSON objektu `{"success": true}` v odpovědi.

REST API – 1. část

Dárky - Gifts

metoda	url	parametry	popis	Odpověď (JSON)
GET	http://giftrea.com/api/gifts		získá všechny záznamy	{gift_id, title}[]
GET	http://giftrea.com/api/gifts/from/2		získá záznamy s id > 2	{gift_id, title}[]
GET	http://giftrea.com/api/gifts/2		získá záznam s id=2	{gift_id, title}
POST	http://giftrea.com/api/gifts	title	navrhne nový dárek	
PUT	http://giftrea.com/api/gifts/2	title	updatuje záznam s id=2	
DELETE	http://giftrea.com/api/gifts/2		smaže záznam s id=2	

Nápady - Ideas

GET	http://giftrea.com/api/ideas/x,y,z		získá návrhy dáreků, které vyhovují tagům s id x, y, z atd.	{gift_id, weight_sum, relevance}[]
POST	http://giftrea.com/api/ideas	gift_id, tag_id[]	vloží nový návrh dárku/zvětší váhu existujícího o 1	

Seznam přání - Wishlist

GET	http://giftrea.com/api/wishlist/1		získá dárky z wishlistu uživatele s id=1	{gift_id, taken_by, created_timestamp}[]
POST	http://giftrea.com/api/wishlist	user_id, gift_id[]	vloží/aktualizuje wishlist uživatele user_id s dárky gift_id[]. Atributy taken_by u dáreků na serveru zůstanou nezměněny.	{gift_id, taken_by, created_timestamp}[]
PUT	http://giftrea.com/api/wishlist/1	user_id, taken_by	změní ID uživatele (taken_by), který si zabral dárek s id=1 ve wishlistu uživatele s user_id.	

Obrázek 5.1. REST API - Dotazy, které posílá aplikace serveru - 1. část

REST API – 2. část

Nahlášení nevhodného dárku - Report

metoda	url	parametry	popis	Odpověď (JSON)
POST	http://giftrea.com/api/report	message	pošle zprávu o problému na e-mail, který je interně nastavený na serveru	

Tagy - Tags

GET	http://giftrea.com/api/tags		vrátí všechny tagy	{tag_id, title}[]
POST	http://giftrea.com/api/tags	title	navrhne nový tag	

Uživatel - User

POST	http://giftrea.com/api/user	user_id, profile_photo, credits_count, day_of_birth, month_of_birthday, email, name, gender, age	vytvoří v DB na serveru uživatele se zaslánými atributy (table_user)	
------	-----------------------------	--	--	--

Anonymní dotaz - Ask

GET	http://giftrea.com/api/ask/asker_id=1&responder_id=2 <i>NEBO pokud není responder_id</i> http://giftrea.com/api/ask/asker_id=1&responder_email=example%40gmail.com		získá questions, které zodpověděl uživatel s id=2 uživateli s id=1	{question_id, unread, created_timestamp, preference:[], comment:[], gift_id:[]}[]
POST	http://giftrea.com/api/ask	asker_id, responder_id, gift_id[], email	pošle e-mail s textem „Jaký z těchto dárků bys nejraději dostal?“ a odkazem o řádek níž na zadanou e-mailovou adresu.	
GET	http://giftrea.com/api/ask/form/<hash> (md5(asker_id, responder_id, gift_id[]))		tento odkaz se pošle e-mailem dotázané osobě (viz skript výše). Po kliknutí se otevře formulář s výběrem dárků	Odpověď je webový formulář ve formátu HTML

Obrázek 5.2. REST API - Dotazy, které posílá aplikace serveru - 2. část

5.3 Google Analytics

Google Analytics představují velmi užitečnou sadu nástrojů pro sledování celé řady kvantitativních měřítek webových stránek a mobilních aplikací. Tyto nástroje jsou navíc k dispozici zcela zdarma, a tak nestojí žádná překážka v jejich využití po vydání mobilní aplikace, která je v rámci této bakalářské práce vyvíjena.

Sada Google Analytics umožňuje sledovat například statistické údaje o uživateli týkající se demografie, geografie, pohlaví, věku a zájmů uživatelů či nejrůznějších aspektů jejich chování v aplikaci.

Demografické údaje

Díky Google Analytics můžeme zjišťovat, jaké věkové skupiny aplikaci nejvíce využívají a jaké pohlaví či zájmy mezi uživateli převažují. Tyto údaje využijeme k průběžnému upřesňování cílové skupiny uživatelů, na kterou se poté zaměříme při propagaci aplikace.

Geografické údaje

Dalším sledovatelným jevem jsou geografické údaje o uživateli. Jedná se především o jazykové skupiny a mapu znázorňující rozmístění uživatelů v jednotlivých zemích světa. Díky těmto údajům můžeme identifikovat oblasti, ve kterých je aplikace nejpobulárnější, a následně na ně zacílit větší část propagace.

Chování uživatelů

Neméně důležitý nástroj pak představuje možnost sledování a měření toho, jak se uživatelé v aplikaci chovají, tedy jaké obrazovky navštěvují nejčastěji, jak dlouho se na nich zdržují a zda z některých obrazovek například rychle odcházejí. Poslední zmíněný případ by mohl indikovat například chybu v aplikaci či nespokojenost se zobrazeným obsahem. V takovém případě bychom obrazovku podrobněji analyzovali a zjistili, jaké potenciální problémy mohou nastat. Významným měřítkem je také podíl uživatelů, kteří se do aplikace opakovaně vrací. Právě v této skupině můžeme v budoucnu hledat již zmíněné uživatele s vysokou hodnotou [4], kteří jsou důležití pro úspěšnou monetizaci aplikace.

5.4 Použité algoritmy

V této sekci si vysvětlíme, jaké algoritmy byly použity k zajištění základní funkcionality aplikace. Jedná se především o výpočet relevance dárku, který představuje vedle marketingového modelu základní kámen úspěchu celého projektu.

5.4.1 Výpočet relevance dárku

Při vyhledávání doporučení na dárky chceme uživateli prezentovat pouze ta nejvhodnější doporučení, která nejlépe reflektují zadané zájmy adresáta. Navrhne proto cenovou funkci, která na základě množiny zadaných tagů a množiny dáreků v databázi vypočítá relevanci jednoho dárku jako reálné číslo.

Operace prováděné na serveru

Každý tag v databázi bude mít své unikátní identifikační číslo, dále jen ID. Vstupem do algoritmu pro výpočet relevance dárku bude množina T obsahující ID tagů zadaných uživatelem:

$$T = \{tag_id_1, tag_id_2, \dots, tag_id_n\}$$

kde n je počet zadaných tagů.

Množina tagů T poté vstupuje do dotazu na databázi, který vybere z asociační tabulky dárků a tagů všechna ID a celkové váhy dárků, které mají vazbu s některým tagem v množině T . Dále je seřadí sestupně podle celkových vah. Tento dotaz můžeme v jazyce SQL popsat následovně:

```
SELECT id_gift, SUM(weight) AS weight_sum FROM table_gift_tag_weight
WHERE tag_id IN (t1, t2, ..., tn) GROUP BY id_gift ORDER BY weight_sum
DESC
```

Tento dotaz vrátí množinu uspořádaných dvojic ID dárků a součtů vah všech přiřazených dárků. Váhu každého dárku nyní vydělíme součtem vah dárku, který má tento součet nejvyšší v celé databázi, čímž získáme jeho relativní relevanci.

$$M = \{[id_gift, relevance]\}$$

Získaná pole dárků a součtů vah k nim náležejících poté odešleme ze serveru do aplikace jako odpověď na zaslaný požadavek. V notaci JSON příklad takové odpovědi můžeme zapsat následovně:

```
"gift_id" : [1, 5, 12],
"relevance" : [1, 0.7, 0.5]
```

Operace prováděné u klienta

V samotné aplikaci zpracujeme data obdržaná v odpovědi a uložíme je do vhodné datové struktury. Využijeme k tomu třídy `JSONObject` a `JSONArray`, které jsou součástí standardních knihoven jazyka Java. Poté výsledná data uložíme do lokální SQLite databáze a zobrazíme je v uživatelském rozhraní aplikace.

5.5 Vývoj aplikace

Cílem této bakalářské práce bylo dodat aplikaci pro mobilní zařízení s operačním systémem Android, která plní stanovené funkční požadavky. Samotný vývoj začal již v říjnu roku 2014, nicméně od té doby byl opakovaně přepracován její koncept a klíčová funkcionalita. Spolu s Bc. Ondřejem Baslerem, hlavním grafikem projektu, jsme si uvědomili, že původní návrh byl příliš omezující a nedokázal by reflektovat všechny potřeby uživatelů. Například nový koncept tagů při hledání doporučení na dárky vyžadoval i výrazné změny uživatelského rozhraní.

Primární funkcí aplikace je komunikace s internetovou databází, prostřednictvím které získává doporučení na dárky ze serveru a naopak posílá na server informace o dříve věnovaných dárkách od uživatelů. Dále byly implementovány doplňující funkce, kterou jsou popsány ve funkčních požadavcích 4.2.

5.6 Hardwarové a softwarové požadavky

Aplikace podporuje Android od verze 2.3.3 (API 9, Gingerbread), která prostřednictvím knihoven zpětné kompatibility dokáže zajistit veškeré potřebné funkce pro aplikaci. Aplikace vzhledem ke svému zaměření a pojetí nevyžaduje žádné speciální funkce dostupné pouze pro novější verze Androidu.

Z hlediska hardwarových požadavků jsem usiloval o šetrné využívání prostředků, kterými mobilní telefony a tablety disponují. Snažil jsme se implementovat potřebné algoritmy efektivně a operace, které mohou způsobit zpomalení telefonu, neprovádět na hlavním vlákně. Nicméně samotný běh operačního systému Android vyžaduje určité minimální hardwarové vybavení, které je nutné pro svižný chod systému i většiny aplikací. Tyto hodnoty jsem odhadl na základě vlastních zkušeností a jsou uvedeny v tabulce 5.1. Jediný softwarový požadavek pak lze najít v tabulce 5.2.

Požadavek	Hodnota
Frekvence procesoru	1 GHz
Operační paměť	256 MB

Tabulka 5.1. Hardwarové požadavky

Požadavek	Hodnota
Verze systému android	2.3.3 a vyšší

Tabulka 5.2. Softwarové požadavky

Kapitola 6

Testování

Nyní, když byla vysvětlena základní struktura a funkcionalita aplikace v rámci analýzy a byla provedena její praktická implementace, zbývá ověřit, zda hotové řešení splňuje požadavky uživatelů. Využijeme k tomu metodu prototypového testování s uživateli, k jejíž realizaci poslouží beta verze vyvíjené aplikace.

6.1 Výběr participantů

Mobilní aplikace vyvíjená v rámci této bakalářské práce byla otestována s 5 participanty. První participant testoval dne 14. 5. 2015, zbylí 4 participantů pak dne 15. 5. 2015. Testování proběhlo v přirozených podmínkách, ve kterých bude aplikace využívána, tedy ve veřejných společenských zařízeních a v městských exteriérech. Pro testování byl využit prototyp aplikace, který už se velmi blížil finální verzi a disponoval již všemi plánovanými funkcemi.

Byli vybráni participanté, kteří mají zkušenosti s operačním systémem Android, neboť právě jeho uživatelé představují primární cílovou skupinu vyvíjené aplikace. Jednalo se o studenty středních a vysokých škol. Během testování participanté plnili úkoly zadané v testovacím scénáři B.

První participant

- Pohlaví: muž
- Věk: 22
- Vzdělání: středoškolské (student VŠ)
- Vlastní zařízení s Androidem: Ano

První participant si v při testování vedl poměrně dobře. Vzhled aplikace hodnotil kladně. Ikonku pro rychlý vstup do nepřečtených odpovědí na anonymní dotazy identifikoval správně, nákupní košík však považoval za proklik do internetového obchodu, kde lze zakoupit dárky. Možnost přechodu přímo do internetových obchodů za účelem nákupu dárků bude zřejmě do aplikace začleněna v budoucnu. Nerad bych však dopustil, aby aplikace působila komerčně, jak je tomu u většiny konkurentů zmíněných v rešerši existujících řešení 3.

O čísle udávajícím počet kreditů se participant domníval, že se jedná o celkovou cenu položek v nákupním košíku. Proto v budoucích verzích aplikace zvážím umístění symbolu mince u počtu kreditů, ačkoli se tento nenachází v grafickém návrhu, aby byl význam čísla jasnější.

Dále participant klikal na jména přátel na obrazovce *Preferovaní přátelé* s očekáváním, že se provede určitá akce. V budoucnu tak zřejmě bude vhodné spustit po kliknutí na jméno přítele například vyhledávání nápadů.

Druhý participant

- Pohlaví: žena

- Věk: 18
- Vzdělání: základní (studentka SŠ)
- Vlastní zařízení s Androidem: Ano

Druhou participantku příliš neoslovil vzhled aplikace, úvodní obrazovka jí připadala nezajímavá. Tento problém bude odstraněn v budoucí verzi aplikace, ve které se na úvodní stránce zobrazí seznam novinek v aplikaci (například nově přidaná přání přátel v seznamech či doporučené dárky). Participantka se, podobně jako její předchůdce, domnívala, že nákupní košík bude odkazovat přímo do internetového obchodu. Důležitým zjištěním bylo, že se bála přihlásit se svým Facebook či Google+ účtem z důvodu možné nevyžádané aktivity aplikace.

Při specifikaci adresáta dárku participantka navrhovala přidat do seznamů vztahů také bývalého partnera či partnerku, což by mohlo být zajímavé rozšíření. Při přidávání koníčků participantce vadilo, že musí po každém přidání znovu klikat na vyhledávací pole. Při prohlížení nápadů pak participantka klikala na názvy nápadů a očekávala, že se dostane přímo do internetového obchodu.

Participantka také poukázala na problém, kdy si uživatel rezervuje dárek ze seznamu přání jiného člověka, ale nakonec mu ho k dané příležitosti nekoupí. Aplikace by měla v takovém případě umožnit potrestání nezodpovědného uživatele, například omezením počtu rezervací v budoucnu. Tento nápad otevírá i diskuzi pro zavedení systému karmy v aplikaci - každý uživatel by si slušným chováním zvyšoval svou reputaci, což by mu v aplikaci otevíralo nové funkce. Stejně tak nízká karma by funkce blokovala. Podobný systém úspěšně provozuje například populární web StackOverflow ¹⁾. Otázkou však je, zda by takový přístup spíše neodrazoval nové uživatele s minimálními právy.

Posledním důležitým poznatkem z testování byla skutečnost, že participantka byla zmatena systémem preferovaných a ostatních přátel 4.4. V současné době mne ale nenapadá vhodnější způsob třídění.

Třetí participant

- Pohlaví: muž
- Věk: 22
- Vzdělání: středoškolské (student VŠ)
- Vlastní zařízení s Androidem: Ano

Třetí participant si při plnění úkolů vedl velmi dobře. Orientoval se v aplikaci velice rychle přesto, že ji nikdy předtím nepoužil. Při testování ocenil, že prvky uživatelského rozhraní jsou standardní a jejich ovládání si tak lze rychle osvojit. Participant podobně jako jeho kolegové hledal nápady na dárky i seznamy přání v záložce , *Přatelé*, ale uvědomoval si, že tyto sekce lze otevřít i ze samostatných záložek v menu.

U seznamu nápadů participant také očekával, že kliknutím na název dárku se dostane do internetového obchodu. Stejně jako jeho předchůdci ho nenapadlo, že v seznamu nápadů se zobrazují dříve vyhledané nápady a pro jejich obnovení je potřeba stisknout tlačítko. Co se týče přihlášení do sociálních sítí, participant by se obával přihlásit se ke svému účtu na Facebooku, zatímco Google+ by mu nevadilo, protože tam nemá citlivá osobní data.

Zajímavostí bylo, že participant jako jediný myslel, že ikonka zeleného zámku v seznamu přání znamená dárek dostupný k rezervování, zatímco šedý je již rezervován

¹⁾ <https://stackoverflow.com>

jiným uživatelem, a tedy nedostupný. Jejich funkcionalita je však přesně opačná, jak se lze dočíst v uživatelské příručce D. Vzhledem k ojedinelosti tohoto opačného výkladu však zřejmě barvy zámků budou zachovány.

Participant dále postrádal možnost dodatečně slučovat ručně vytvořené přátele s kontakty importovanými ze sociálních sítí. S touto funkcí se v dalších verzích počítá. Nakonec mne potěšilo zjištění, že participant by se nebál vložit do aplikace e-mail svých přátel. Uklidnilo ho, že v aplikaci nejsou žádná loga firem ani jiné známky komerce.

Čtvrtý participant

- Pohlaví: žena
- Věk: 22
- Vzdělání: středoškolské (studentka VŠ)
- Vlastní zařízení s Androidem: Ano

Participantka se v aplikaci orientovala poměrně dobře. Potěšilo mne, že údajně zatím podobnou aplikaci neviděla a považuje nápad za originální a praktický. Při vyhledávání nápadů participantku, stejně jako většinu ostatních, nenapadlo, že zobrazovaný seznam nápadů není aktuální, a pro jeho obnovení musí stisknout tlačítko.

Na obrazovce *Přátelé* se participantka domnívala, že ikona srdce (která přidává přátele do preferovaných) slouží pro vytvoření blízkých přátel či partnera. Participantka by se nebála přihlásit se svým Google+ nebo Facebook účtem ani zadat e-mail svých přátel při odesílání anonymního dotazu. Se samotným posláním dotazu participantka neměla žádný problém.

Pozitivní také bylo zjištění, že participantku by zajímalo, co si přejí její přátelé ve skutečnosti, a tak by aplikaci zřejmě využila i v budoucnu. Za nejužitečnější funkcionalitu považovala participantka seznamy přání.

Pátý participant

- Pohlaví: muž
- Věk: 26
- Vzdělání: vysokoškolské technického směru
- Vlastní zařízení s Androidem: Ano

Participant neměl s orientací v aplikaci problémy. Odhadl správně, že ikona obálky má souvislost s posíláním zpráv, ale nebyl si jistý, co přesně znamená v kontextu aplikace. Po vysvětlení funkce anonymních dotazů participant očekával, že po kliknutí se například otevře dialog obsahující samostatné prokliky na otázky a odpovědi. Tento postřeh mne přivedl k myšlence zobrazit po kliknutí na obálku dialog, kde bude seznam nejnovějších odpovědí, které budou odkazovat do příslušné sekce v aplikaci. Podobný dialog nabízí například sociální síť Facebook ¹⁾.

Dále by bylo možné z horní lišty zobrazovat dialog s naposledy přidávanými přáními přátel či další aktuality. Právě tyto sociální funkce participant ve výsledku hodnotili nejlépe, a tak by bylo vhodné se na ně při dalším vývoji aplikace zaměřit. Participant uvedl, že by se nebál přihlásit prostřednictvím aplikace na Google+, účet na Facebooku nevlastní.

Dalším důležitým postřehem pak byla skutečnost, že participant měl problémy se zmáčknutím poměrně malých tlačítek v sekci *Preferovaní přátelé*. Jejich šedá barva

¹⁾ <https://www.facebook.com>

pak jen podpořila dojem, že jsou neaktivní či nefunkční. V budoucích verzích tedy bude potřeba tlačítka zvětšit a pravděpodobně také změnit jejich barvu. Důležitost této změny podtrhuje také fakt, že v podstatě všichni participanté tyto prokliky využívali jako primární způsob navigace v aplikaci a záložky v menu zůstaly spíše na okraji. Stejně tak do odpovědí se participant dostával spíše pomocí ikony obálky než ze záložky v menu.

Participant si také nevyšiml rozbalovacího menu pro výběr přítele vpravo nahoře a myslel si, že volba přítele je možná pouze ze záložky *Přátelé*. Bude tedy potřeba tento prvek uživatelského rozhraní zvýraznit.

6.2 Výsledky

Obecně lze říci, že testování s uživateli proběhlo úspěšně a participanté neměli větší problémy s orientací v aplikaci, ačkoli ji nikdy předtím nepoužili. Přispělo k tomu hlavně využití standardních komponent Androidu a návrh uživatelského rozhraní v souladu s aplikacemi, které jsou dnes mezi uživateli nejpobulárnější.

Dále bylo zajímavé poslechnout si názory budoucích uživatelů na vyvíjenou aplikaci a potěšilo mne, že je její funkcionalita zaujala. Přesto však bylo při testování zjištěno několik problémů, které se často opakovaly i u více participantů. Právě tyto nedostatky bude důležité v blízké době vyřešit a odstranit, aby zbytečně neodradily první uživatele aplikace od jejího používání. Seznam změn vyplývajících ze zjištěných problémů je uveden v sekci 6.2.1.

6.2.1 Přínos testování - nové funkce a změny v budoucích verzích aplikace

- Proklik do internetového obchodu u každého dárku v seznamu nápadů, seznamu přání kontaktu a seznamu odpovědí na anonymní dotazy
- Přidat výchozí akci po kliknutí na jméno v seznamu přátel
- Možnost potrestat uživatele, který si rezervuje něčí přání a poté mu jej k dané příležitosti nekoupí
- Automatické otevírání rozbalovacího seznamu koníčků po přidání každého koníčku. Dále také možnost přidání koníčku po stisknutí tlačítka *Enter* v telefonu.
- Smazat z aplikace dříve vyhledané nápady po změně atributů kontaktů, aby uživatel věděl, že pro vyhledání nových musí stisknout příslušné tlačítko
- Možnost dodatečně slučovat ručně vytvořené přátele s jejich skutečnými účty na sociálních sítích
- Po kliknutí na ikonu obálky v horní liště zobrazit dialog se seznamem nejnovějších odpovědí na anonymní dotazy, který bude odkazovat do příslušné sekce. Dále přidat do horní lišty obdobnou ikonu a dialog pro zobrazení nejnovějších přání přátel.
- Zvětšit tlačítka odkazující do jednotlivých sekcí na obrazovce *Preferovaní přátelé*. Případně také změnit jejich barvu, aby šedá neevokovala dojem, že jsou neaktivní.
- Umožnit mazání položek z nákupního seznamu i ze záložky *Historie*
- Zvýraznit rozbalovací seznam pro výběr aktuálního přítele vpravo nahoře
- Obecně bude vhodné klást důraz na zkratky a rychlé způsoby navigace z horní lišty a seznamu přátel. Uživatelé nechtějí ztrácet čas a hledají centralizovaná řešení.

Kapitola 7

Závěr

V rámci této bakalářské práce byla provedena analýza, návrh, implementace a testování mobilní aplikace pro doporučování dárců na platformě Android. Dle výsledků testování 6 můžeme usoudit, že vytvořená aplikace poskytuje uživatelsky přívětivé grafické rozhraní, ve kterém se i nový uživatel dokáže poměrně rychle zorientovat. Všechny primární funkce aplikace byly úspěšně implementovány a nyní již záleží na samotných uživateli, do jaké míry ocení a využijí poskytnuté řešení. Povědomí o aplikaci bude rozšiřováno například propagací na sociálních sítích. Samotní uživatelé určitě přinesou během používání mnoho cenných námětů a podnětů k vylepšení některých částí aplikace či k opravě nedostatků a chyb. Právě komunikaci s uživateli v době po vydání aplikace považují za velmi důležitou, protože existence loajální komunity hraje klíčovou roli v dlouhodobém úspěchu aplikace.

7.1 Splnění zadaných úkolů

Na základě zadání práce A bylo potřeba splnit následující úkoly:

- Navrhnout a implementovat mobilní aplikaci pro doporučování dárců
- Aplikace bude sbírat data o dříve věnovaných dárcích od uživatelů
- Aplikace bude poskytovat doporučení na základě specifikace adresáta
- Navrhnout free a placený model aplikace
- Lokalizovat aplikaci do češtiny a angličtiny
- Zapojit také osoby nevlastnící mobilní telefon nebo tablet

Všechny zadané úkoly byly splněny. Problém nastal pouze u lokalizace do češtiny a angličtiny, která je vzhledem k charakteru aplikace komplikovaná. Uživatelské rozhraní bylo přeloženo do obou jazyků, ale názvy dárců a tagů, které mohou uživatelé sami přidávat, nejsou předem známé, a tak je nelze snadno přeložit. Proto byl navržen lokalizační model 4.7, na základě kterého mohou uživatelé překládat názvy dárců a tagů, za což budou odměněni kredity. Tento model v současné verzi aplikace není implementován, ale patří mezi prioritní cíle v budoucích verzích.

Vzhledem k rozsahu funkcionality vyvíjené aplikace a příchodu nových nápadů na její rozšíření v průběhu tvorby byly některé cíle, které nebyly součástí zadání a nejsou kritické pro její vydání, odloženy, a jejich realizace bude provedena již mimo rámec této bakalářské práce. Například placený model aplikace byl navržen a popsán, jak bylo zadáno, ale nebyl implementován v samotné aplikaci. Vzhledem k tomu, že placené modely aplikací často selhávají a nenaplňují očekávání vývojářů, rád bych jeho implementaci věnoval delší dobu, abych dosáhl skutečně kvalitního výsledku.

Dalším budoucím cílem zůstává implementace alternativních způsobů získání kreditů. V kapitole 4.3 byla zmíněna možnost vyplnit za kredity cenu dárců, které ji dosud nemají přiřazenu. Díky tomu by uživatel mohl získat kredity pro vyhledávání i v případě, že si například nepamatuje, jaký dárek adresátovi dal při minulé příležitosti, nebo když mu žádný dárek dosud nedal. Tato funkcionalita bude implementována v některé z příštích verzí aplikace.

7.2 Další postup

Budoucí vývoj aplikace velmi záleží na způsobu přijetí uživateli, zpětné vazbě a obecně úspěchu celého projektu. V případě, že by popularita aplikace mezi uživateli nesplnila očekávání a dočkala se spíše vlažného přijetí, přesunul bych těžiště své práce na jiné projekty, jejichž realizaci mám v plánu. Naopak, pokud aplikace zaznamená úspěšný vstup na trh a dokáže si vybudovat širší uživatelskou základnu, v což doufám, přicházelo by v úvahu další rozšíření funkcionality.

Velmi žádoucí rozšíření v případě rozrůstající se komunity by představovalo rozsáhlejší využití sociálních sítí. Možnosti sdílení a vzájemného předávání informací přímo z aplikace by zajistilo další organický růst uživatelské základny.

V případě, že se osvědčí placený model aplikace a projekt bude generovat příjem, který nebude zcela zanedbatelný, rád bych jej reinvestoval zpět do aplikace formou reklamy na Google Adwords a Facebooku.

Další možné rozšíření představuje webová verze aplikace, která by umožnila zapojení každého uživatele Internetu. Výhodou by byla možnost využití již existující databáze a REST API.

Nesmím zapomenout ani na možnou implementaci verze aplikace pro operační systém iOS. Jednalo by se o další poměrně rozsáhlý projekt, ale díky již hotové analýze a databázovým skriptům by se práce omezila v podstatě jen na samotnou implementaci.

Obecně vzato tedy lze říci, že pokud aplikace bude úspěšná, existuje celá řada možných cest, kterými by bylo možné se v rámci projektu dále ubírat. Především budoucí situace a okolnosti rozhodnou, která z nich se ukáže jako ta nejvhodnější.



Literatura

- [1] Petr Ludwig *Konec prokrastinace*. Jan Melvil Publishing, Příbram, 2013.
- [2] Android. EngineersGarage, 2014.
<http://www.engineersgarage.com/articles/what-is-android-introduction/>.
- [3] Number of available applications in the Google Play Store., 2014.
<http://www.statista.com/statistics/266210/number-of-available-applications-in-the-google-p>
- [4] Identify and target high-value users, 2014.
<https://support.google.com/analytics/answer/2819950?hl=en#highvalue>.
- [5] Smartphone OS Market Share, Q4 2014, 2015.
<http://www.idc.com/prodserv/smartphone-os-market-share.jsp>.
- [6] Android Developers - Dashboards, 2015.
https://developer.android.com/about/dashboards/index.html?utm_source=suzunone.
- [7] Android Developers - Navigation Drawer, 2015.
<https://developer.android.com/design/patterns/navigation-drawer.html>.
- [8] Android open source project license.
<https://source.android.com/source/licenses.html>.
- [9] Satya Komatineni and Dave MacLean. *Pro Android 4*. Apress, New York City, 2012.
- [10] ActionBar.Tab, 2015
<http://developer.android.com/reference/android/app/ActionBar.Tab.html>.
- [11] Deprecated, 2015
<http://docs.oracle.com/javase/7/docs/api/java/lang/Deprecated.html>.
- [12] Zigurd Mednieks, Laird Dornin, G. Blake Meike, Masumi Nakamura. *Programming Android*. O'Reilly Media, Inc., Sebastopol, 2011
- [13] Why AsyncTask is bad and you should feel bad, 2015.
<http://simonvt.net/2014/04/17/async-task-is-bad-and-you-should-feel-bad/>.
- [14] Activity, 2015.
<http://developer.android.com/reference/android/app/Activity.html>.
- [15] Nested classes, 2015.
<https://docs.oracle.com/javase/tutorial/java/java00/nested.html>.
- [16] Shu-Wai Chow. *Programujeme Mashup aplikace pro Web 2.0 v PHP*. Computer Press, a.s., Brno, 2008.
- [17] HTTP, 2015.
<http://www.computerhope.com/jargon/h/http.htm>.
- [18] PUT vs POST, 2015.
<http://restcookbook.com/HTTP%20Methods/put-vs-post/>.

-
- [19] Mark L. Murphy *Android 2*. Computer Press, a.s., Brno, 2011.
- [20] Mark Massé. REST API Design Rulebook. O'Reilly Media, Inc., Sebastopol, 2012
- [21] Introducing JSON, 2015.
<http://json.org/>.
- [22] JSON : jednotný formát pro výměnu dat, 2008.
<http://www.zdrojak.cz/clanky/json-jednotny-format-pro-vymenu-dat/>.
- [23] Steven Holzner. *Java*. Paraglyph Press, Phoenix, 2001.
- [24] About SQLite, 2015.
<http://www.sqlite.org/about.html>.
- [25] Wallace Jackson. Learn Android App Development. Apress, New York City, 2013.
- [26] Transaction, 2015.
<http://searchcio.techtarget.com/definition/transaction>.
- [27] The ACID Model, 2015.
<http://databases.about.com/od/specificproducts/a/acid.html>.
- [28] Application Programming Interface, 2015.
<http://www.pcmag.com/encyclopedia/term/37856/api>.
- [29] PHP: Hypertext Preprocessor, 2015.
<http://php.net/>.
- [30] Definition of iOS, 2015.
<http://www.pcmag.com/encyclopedia/term/45346/ios>.
- [31] What is GPS?, 2014.
<http://www8.garmin.com/aboutGPS/>.
- [32] What is MySQL?, 2015.
http://www.w3schools.com/php/php_mysql_intro.asp.
- [33] ID, 2015.
<http://www.thefreedictionary.com/ID>.
- [34] Software Development Kit, 2005.
<http://whatis.techtarget.com/definition/software-developers-kit-SDK>.
- [35] What is XML?, 2005.
<http://www.xml.com/pub/a/98/10/guide0.html?page=2#AEN58>.
- [36] Use Cases, 2015.
<http://www.usability.gov/how-to-and-tools/methods/use-cases.html>.

Příloha A

Zadání práce

České vysoké učení technické v Praze
Fakulta elektrotechnická

katedra počítačové grafiky a interakce

ZADÁNÍ BAKALÁŘSKÉ PRÁCE

Student: **Jan Sládek**

Studijní program: Softwarové technologie a management
Obor: Web a multimedia

Název tématu: **Mobilní aplikace poskytující uživatelům doporučení na dárky**

Pokyny pro vypracování:

Navrhněte a implementujte mobilní aplikaci pro doporučování dárků. Aplikace bude sbírat data o již věnovaných dárkách od aktivních uživatelů a vhodným způsobem bude navrhopvat nové dárky dle zadaných kritérií (např. věk, pohlaví, druh dárku, cena). Navrhněte vhodný free a placený model aplikace. Aplikace bude lokalizována minimálně do češtiny a angličtiny. Navrhněte způsob zapojení uživatelů nevlastnicích mobilní telefon nebo tablet.

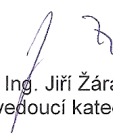
Aplikaci otestujte s cílovou skupinou uživatelů.

Seznam odborné literatury:

Satya Komatineni, Dave MacLean, Pro Android 4, Apress, 2012.

Vedoucí: Ing. David Sedláček, Ph.D.

Platnost zadání: do konce letního semestru 2015/2016


prof. Ing. Jiří Žára, CSc.
vedoucí katedry




prof. Ing. Pavel Ripka, CSc.
děkan

V Praze dne 25. 3. 2015

Obrázek A.1. Zadání bakalářské práce

Příloha B

Scénář prototypového testování

Scénář prototypového testování aplikace GIFTrea

Úvodní obrazovka a menu

1. Jak se Ti aplikace na první pohled líbí? Které grafické prvky Ti připadají hezké a které méně?
2. Co myslíš, že znamenají ikonky a číslo vpravo nahoře?
Otevři menu.
3. Co myslíš, že můžeš udělat v jednotlivých záložkách?

Nápady na dárky

Úkol: Vyhledej nápady na dárky pro uživatele Jan Sladek

1. Jak na Tebe působilo prohlášení „Nebudeme na Facebooku ani Google+ nic publikovat bez Vašeho povolení“? Báł by ses i přesto se svým účtem přihlásit?
2. Připadal Ti postup vyhledání nápadů srozumitelný?

Přání přátel

Úkol: Importuj si přátele z Google+ (Přihlas se účtem dforev@gmail.com). Zjisti, jaké dárky si přeje uživatel Jan Sladek

1. Bylo Ti zřejmé, kterou položku v menu otevřít?
2. Připadal Ti postup vyhledání přání srozumitelný?
3. Zajímalo by Tě, co si Tví kamarádi přejí i ve skutečnosti?
4. Co myslíš, že dělají jednotlivé ikonky u jeho přání?

Úkol: Vyber si některý z dárků, které si Jan Sládek přeje, a rezervuj si ho, aby ostatní věděli, že už mu ho někdo kupuje a nedostal ho dvakrát.

Úkol: Bojíš se, že mu tento dárek zapomeněš koupit, a rád by sis tuto informaci nějak uložil. Napadá Tě, jak to udělat?

Úkol: Rozmyslel sis to a dárek mu kupovat nebudeš. Uvolni rezervaci pro ostatní a odstraň dárek z nákupního seznamu.

1. Obtěžovalo by Tě přihlašování ke svému Google účtu, nebo bys byl naopak rád, že by se Ti automaticky importovali přátelé a jejich přání?

Vlastní seznam přání

2. Máš 10 vteřin na rozmyšlení. Co by sis přál/a k příštím Vánocům?

Úkol: Přidej si to do svého seznamu přání, aby tvoji blízcí věděli, co Ti mají koupit.

3. Bylo Ti zřejmé, kterou položku v menu otevřít?
4. Připadal Ti postup přidání dárku do vlastního seznamu přání srozumitelný?

Obrázek B.2. Scénář prototypového testování s cílovou skupinou uživatelů

Anonymní dotaz

Úkol: Představ si, že máš jiného kamaráda s e-mailovou adresou -baz-@email.cz. Ten bohužel aplikaci nezná, a tak nemá vyplněný seznam přání. Zeptej se ho anonymně, co si přeje. Tušíš, že má rád notebooky, auta a počítače.

1. Bylo Ti zřejmé, kterou položku v menu otevřít?
2. Připadal Ti postup přidání návrhů a poslání dotazu srozumitelný?
3. Kdybys zadával e-mail skutečného kamaráda, napadlo tě, že tato informace může být zneužita? Obával by ses do aplikace jeho e-mail vkládat navzdory ujištění aplikace, že nebude posílat nevyžádané zprávy?
4. Kde bys hledal kamarádovu odpověď na tvůj dotaz? Myslíš, že se o ní dozvíš i na jiném místě?

Opakované vyhledávání

Úkol: Představ si, že je o rok později. Vzpomeň si na příbuzného z prvního úkolu. Vyhledej pro něj doporučení na dárek k narozeninám, aniž bys musel znovu zadávat všechny jeho koníčky.

Úkol: Vrať se zpět, a vyhledej nápady několikrát, dokud se nezobrazí dialog.

1. Věděl sis rady se zobrazeným dialogem? Obtěžovalo Tě zadání minulého dárku, nebo jsi rád, že tím přispěješ k rozvoji aplikace? Napadlo by tě vložit do pole nějaký nesmyslný či vulgární text?

Obecné

1. Napadá tě ještě nějaký podnět, který bys chtěl k aplikaci říci?
2. Která funkcionality tě zaujala nejvíce?
3. Co by se dalo na aplikaci zlepšit?

Poděkování a rozloučení

Děkuji Ti za účast v tomto testování. Jako pozornost ode mě obdržíš zdarma neomezenou verzi aplikace po jejím vydání.

Obrázek B.3. Scénář prototypového testování s cílovou skupinou uživatelů - pokračování

Příloha C

Obsah přiloženého CD

Na přiloženém CD se nacházejí soubory se zdrojovými kódy dokumentace i implementační části této bakalářské práce. Ve složce *Bakalářská práce* se nacházejí podsložky *Dokumentace* a *Implementace*.

Ve složce *Dokumentace* pak najdete složku *PDF*, obsahující dokumentaci této bakalářské práce ve formátu PDF. Ve složce *Zdrojové soubory Tex* pak jsou zdrojové soubory, ze kterých lze výsledné PDF vygenerovat.

Ve složce *Implementace* se nachází podsložka *Instalační soubor apk*, která obsahuje instalační soubor vytvořené aplikace pro zařízení s operačním systémem Android. Ve složce *Zdrojové kódy* pak najdete projekt vytvořený programem Android Studio, který obsahuje všechny zdrojové kódy vyvinuté mobilní aplikace a také zdrojové kódy REST rozhraní serveru, se kterým aplikace komunikuje. Nachází se zde také soubor *readme.txt*, ve kterém jsou uvedeni autoři zdrojových kódů.

- **Bakalářská práce**
 - **Dokumentace**
 - PDF
 - Zdrojové soubory Tex
 - **Implementace**
 - Instalační soubor apk
 - Zdrojové kódy

Obrázek C.4. Obsah přiloženého CD

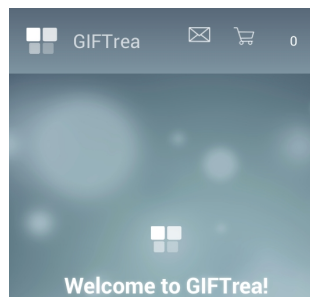
Příloha D

Uživatelská příručka

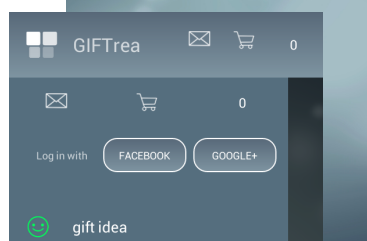
Uživatelská příručka aplikace GIFTrea

V této uživatelské příručce se dozvíte, jak používat základní funkce mobilní aplikace vyvíjené v rámci této bakalářské práce.

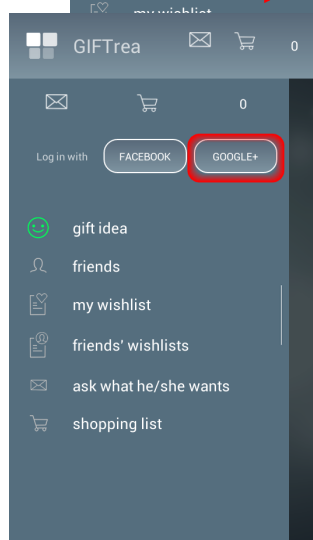
1) Nainstalujte a spusťte aplikaci



2) Zatáhněte prstem od levého okraje obrazovky doprava pro otevření menu

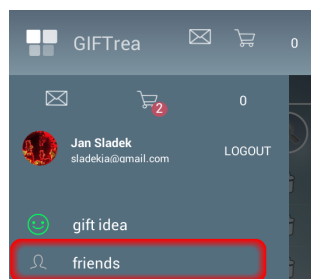


3) Přihlaste se pomocí některé sociální sítě, například Google+

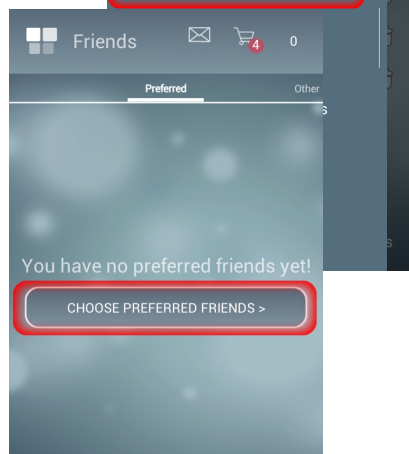


Obrázek D.5. Uživatelská příručka - část 1

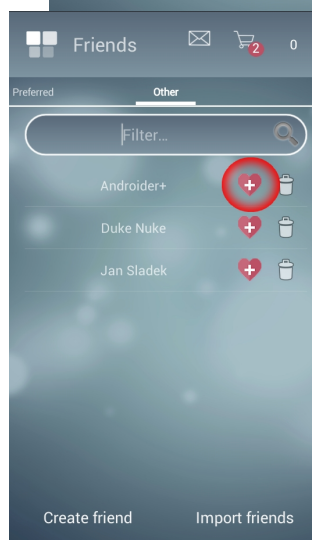
4) Klikněte na tlačítko „friends“ pro otevření sekce pro správu přátel



5) Klikněte na tlačítko „Choose preferred friends“. Alternativně můžete kliknout na nápis „Other“ v horní liště či se přesunout doprava přetažením prstu. Tyto formy navigace můžete použít ve většině obrazovek aplikace.



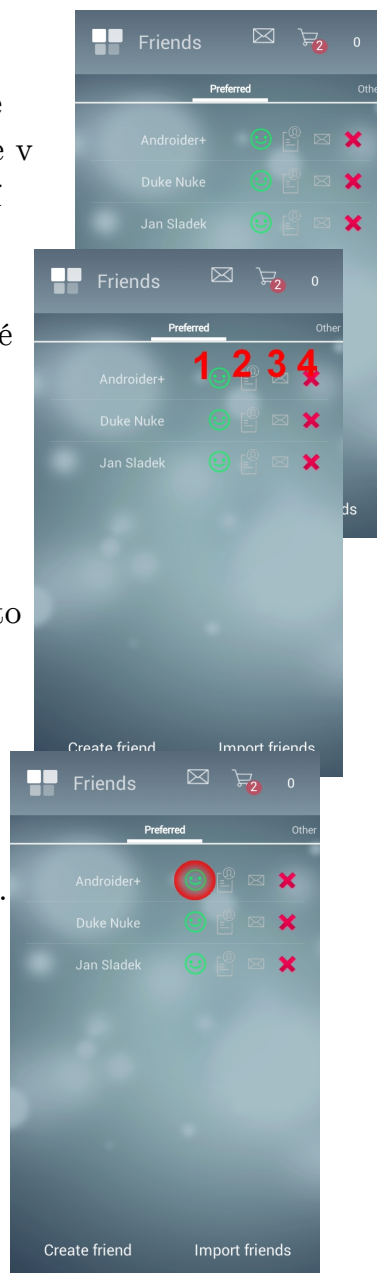
6) Klikněte na ikonu srdce u přítele, abyste jej přidali mezi preferované a mohli jej v aplikaci využívat.



Obrázek D.6. Uživatelská příručka - část 2

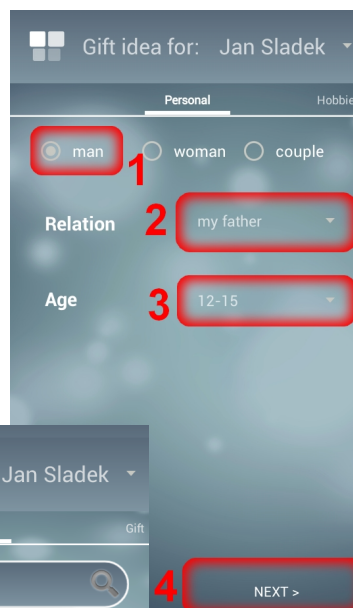
7) Po úspěšném přidání preferovaných přátel se můžete přesvědčit přetažením prstu, že v záložce „Preferred“ se nacházejí vámi vybraní přátelé.

8) Nyní můžete pro preferované přátele vyhledávat nápady na dárky (ikona 1), prohlížet si jejich seznamy přání (ikona 2), posílat jim anonymní dotazy (ikona 3) či je vrátit zpět do Ostatních přátel (ikona 4). Tyto ikony slouží jako zkratka k primárním funkcím aplikace. Můžete se k nim dostat také z vysouvacího menu. Nyní klikněte na tlačítko pro vyhledání nápadu na dárek (1).

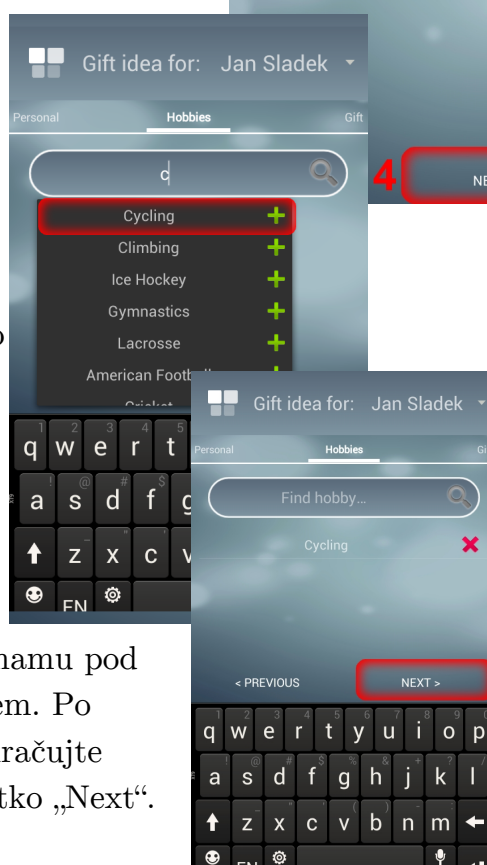


Obrázek D.7. Uživatelská příručka - část 3

9) Nyní se nacházíte na obrazovce pro zadání obecných údajů adresáta dárku. Vyplňte jeho pohlaví (1), svůj vztah k němu (2), jeho přibližný věk (3) a klikněte na tlačítko „Next“ vpravo dole (4) pro pokračování do další sekce.



10) Nyní můžete vyplnit zájmy adresáta dárku. Můžete vyhledávat v množině zájmů vepsáním textu do vyhledávacího pole. Kliknutím na zvolený zájem jej přiřadíte adresátovi. Vybrané zájmy poté uvidíte v seznamu pod vyhledávacím polem. Po přidání zájmů pokračujte kliknutím na tlačítko „Next“.



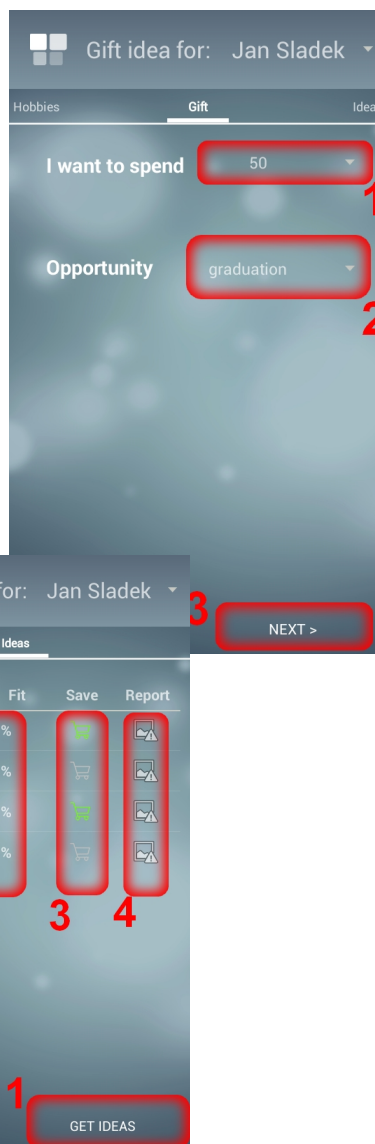
Obrázek D.8. Uživatelská příručka - část 4

11) Nyní můžete nastavit, kolik peněz chcete za dárek utratit (1) a k jaké příležitosti jej budete kupovat (2). Kliknutím na tlačítko „Next“ (3) již přejdete k samotnému vyhledávání nápadů.

12) Po kliknutí na (1) se již zobrazí nápady doporučené na základě atributů, které jste pro adresáta dárku vybrali. V prvním sloupečku (2) je relevance dárku,

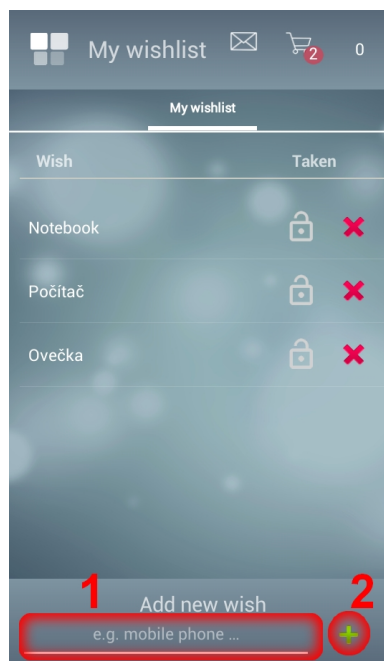
tedy míra shody se zadanými údaji. Ve druhém (3) pak najdete tlačítko pro přidání či odebrání dárku z nákupního seznamu. Poslední (4) tlačítko pak slouží k nahlášení

nevhodného dárku, Přidejte si některý z nápadů do svého nákupního seznamu kliknutím na tlačítko (3).

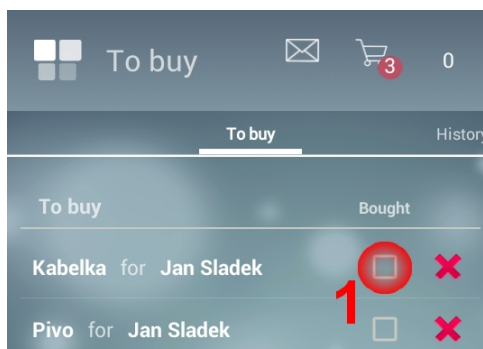


Obrázek D.9. Uživatelská příručka - část 5

13) Nyní, když jsme si již vyzkoušeli základní navigaci v menu i uvnitř sekce, otevřete vysouvací menu a klikněte na záložku „my wishlist“ pro otevření vlastního seznamu přání. Zde můžete přidávat dárky do svého seznamu přání vepsáním jejich názvu do pole (1) a následným potvrzením kliknutím na tlačítko (2). Tato přání si mohou zobrazit uživatelé ve vašich kruzích na Google+, kteří si nainstalují aplikaci a přihlásí se v ní do zmíněné sociální sítě.

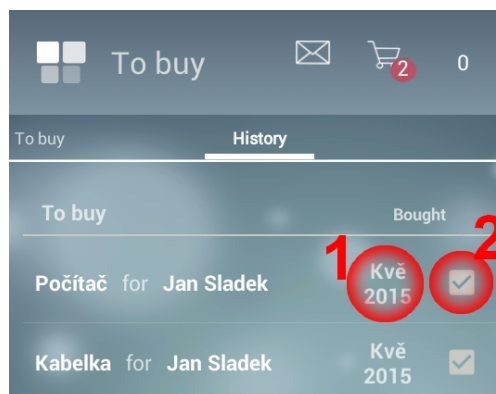




14) Přejděte v menu do záložky „shopping list“, kde najdete dárky, které jste předtím přidali do nákupního seznamu. Kliknutím na (1) je můžete označit jako koupené. Takové dárky přejdou do záložky „History“ vpravo.



Obrázek D.10. Uživatelská příručka - část 6

15) Přetažením prstu se můžete přesvědčit, že se v záložce „History“ nacházejí koupené dárky. Sloupec (1) informuje, kdy byl dárek zakoupen a zaškrtnutí pole (2) umožňuje dárek označit zpět jako dosud nekoupený například v případě omylu.



16) Nyní přejděte do záložky „friends' wishlist” v menu. Pokud některý z vašich přátel má vyplněn seznam přání, uvidíte je zde. Ikona zeleného zámečku  znamená, že dárek je vámi rezervován, a ostatní uživatelé si jej nemohou rezervovat. Červený zámeček  pak znamená, že dárek je rezervován, a tedy pro ostatní zablokován.

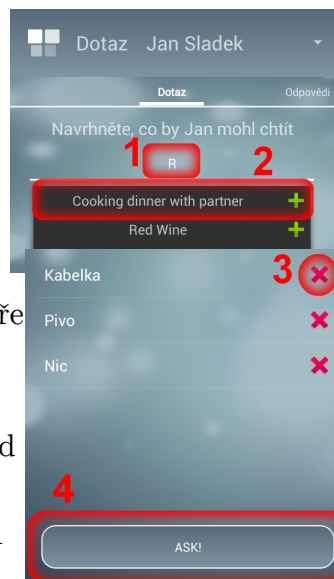
Ikona nákupního vozíku pak přidá či odebere dárek (pokud není zablokován jiným uživatelem) z vašeho nákupního seznamu.



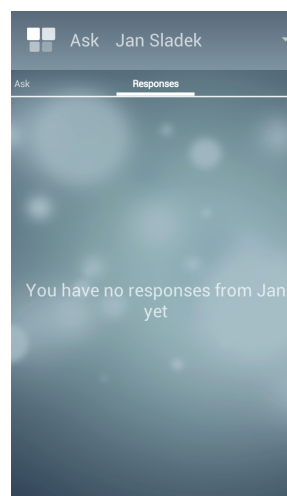
Obrázek D.11. Uživatelská příručka - část 7

17) Jako poslední si ukážeme sekci zaslání anonymního dotazu.

Klikněte v menu na záložku „ask what he/she wants“. Do textového pole (1) vepište název dárku, který by se adresátovi mohl líbit. Poté jej vyberte v našeptávači, který se otevře (2). V současné době nelze vkládat vlastní dárky, protože tyto by nejprve musely být schváleny. Pokud se například spletete, můžete navrhovaný dárek odebrat tlačítkem s křížkem (3). Přidejte alespoň 3 návrhy. Poté můžete kliknout na tlačítko „ASK!“ (4). Aplikace si v takovém případě buď vyžádá e-mailovou adresu cílové osoby, nebo přímo odešle anonymní dotaz, pokud již tuto adresu zná.



18) Přetažením prstu se pak můžete přemístit do záložky „Responses“ vpravo, kde později najdete odpověď na zaslaný anonymní dotaz v podobě hodnocení, do jaké míry si adresát přeje jednotlivé vámi navrhované dárky. Tím jsme si ukázali všechny primární funkce aplikace.



Obrázek D.12. Uživatelská příručka - část 8

Příloha E

Zkratky a symboly

E.1 Zkratky

Zde si vysvětlíme význam jednotlivých zkratk, které byly v práci použity.

- API Application Programming Interface, sada nástrojů a postupů pro tvorbu softwarových aplikací [28].
- PHP Hypertext PreProcessor, skriptovací jazyk využívaný především k vývoji webových aplikací [29].
- REST Representational State Transfer je styl softwarové architektury, který využívá bezstavové HTTP požadavky pro výměnu informací v rámci Internetu [16].
- iOS iDevice Operating System, operační systém používaný na zařízeních Apple iPhone, iPad a iPod Touch. Počítače Mac firmy Apple využívají operační systém Mac OS [30].
- GPS Global Positioning System, navigační systém 24 satelitů umístěných na zemskou orbitu Ministerstvem obrany Spojených států amerických. Původně bylo určeno pro vojenské účely, ale v 80. letech 20. století bylo uvolněno i pro civilní použití [31].
- MySQL Databázový systém využívající strukturovaný dotazovací jazyk SQL. Na rozdíl od SQLite se jedná o proces běžící na serveru. Jedná se o otevřený standard spravovaný společností Oracle. Slabika „My“ v názvu překvapivě nepředstavuje přivlastňovací zájmeno, nýbrž jméno dcery jednoho ze spoluzakladatelů [32].
- ID Forma identifikace určité entity. V této práci představuje zpravidla unikátní celé číslo, sloužící pro rozlišení záznamů v databázi [33].
- SDK Software Development Kit, sada programů sloužících k vývoji softwarových aplikací. Zpravidla zahrnuje vizuální nástroj pro tvorbu uživatelského rozhraní, editor kódu, kompilátor a linker pro vytvoření výsledného spustitelného souboru [34].
- XML Extensible Markup Language, značkovací jazyk sloužící k ukládání informací ve strukturované podobě. Pro každý prvek definuje jeho obsah, funkci a titulek [35].
- UC Use Case neboli případ užití popisuje určitou akci, kterou uživatel provádí na systému a vytváří tak měřitelnou hodnotu. Cílem tvorby případů užití je především nalezení hranic systému [36].
- URI URI neboli Uniform Resource Identifier slouží k jednoznačnému adresování prostředků a zdrojů, které daný systém využívá [20].