

**ČESKÉ VYSOKÉ UČENÍ TECHNICKÉ V PRAZE
FAKULTA DOPRAVNÍ**

Bc. Jan Soldát

MOBILNÍ APLIKACE PRO OBJEKTY HROMADNÉHO PARKOVÁNÍ

Diplomová práce

2018



K620..... Ústav dopravní telematiky

ZADÁNÍ DIPLOMOVÉ PRÁCE (PROJEKTU, UMĚLECKÉHO DÍLA, UMĚLECKÉHO VÝKONU)

Jméno a příjmení studenta (včetně titulů):

Bc. Jan Soldát

Kód studijního programu a studijní obor studenta:

N 3710 – IS – Inteligentní dopravní systémy

Název tématu (česky): **Mobilní aplikace pro objekty hromadného parkování**

Název tématu (anglicky): Mobile Application for Public Parking Objects

Zásady pro vypracování

Při zpracování diplomové práce se řiďte osnovou uvedenou v následujících bodech:

- Seznamte se s problematikou informačních systémů pro hromadné parkovací objekty, a dále mapovými nebo jinými geografickými informačními systémy v oblasti hledání objektů v okolí mobilního uživatele.
- Vypracujte příslušnou analýzu. Zaměřte se mj. na potřeby uživatelů, a dále na funkční, informační a technologickou stránku. Analyzujte také související aplikace, které jsou provozovány v současné době.
- Navrhněte aplikaci systému, který informuje řidiče během jízdy o možnostech parkování v hromadných parkovacích objektech v okolí. Aplikaci směřujte na konkrétní město nebo jeho část. Zaměřte se mj. na architekturu telematického systému, detailně rozpracujte jednotlivé funkce a přenášené informace a návrh způsobu realizace.
- Implementujte navrženou aplikaci v prostředí mobilních zařízení. Vytvořený návrh náležitě zdokumentujte.
- Proveďte testování vytvořené aplikace a analyzujte její spolehlivost.

- Rozsah grafických prací: blíže nespecifikován
- Rozsah průvodní zprávy: minimálně 55 stran textu (včetně obrázků, grafů a tabulek, které jsou součástí průvodní zprávy)
- Seznam odborné literatury: Příbyl, P.: Inteligentní dopravní systémy a dopravní telematika. Praha: Vydavatelství ČVUT, 2004. 184 s. ISBN 80-01-03122-5.
- Svítek, M: Telematika nad dopravními sítěmi. Vydavatelství ČVUT, Praha, 2004, 263 s., ISBN 80-01-03087-3.
- Příbyl, P. Svítek, M: Inteligentní dopravní systémy. Praha: BEN -technická literatura, 2001, 543 s. ISBN 80-7300-029-6.

Vedoucí diplomové práce: **Ing. Vladimír Faltus, Ph.D.**

Datum zadání diplomové práce: **3. července 2017**
(datum prvního zadání této práce, které musí být nejpozději 10 měsíců před datem prvního předpokládaného odevzdání této práce vyplývajícího ze standardní doby studia)

Datum odevzdání diplomové práce: **29. května 2018**
a) datum prvního předpokládaného odevzdání práce vyplývající ze standardní doby studia a z doporučeného časového plánu studia
b) v případě odkladu odevzdání práce následující datum odevzdání práce vyplývající z doporučeného časového plánu studia



doc. Ing. Pavel Hrubeš, Ph.D.
vedoucí
Ústavu dopravní telematiky



prof. Dr. Ing. Miroslav Svítek, dr. h. c.
děkan fakulty

Potvrzuji převzetí zadání diplomové práce.



Bc. Jan Soldát
jméno a podpis studenta

V Praze dne 3. července 2017

Poděkování

V první řadě bych tímto rád poděkoval vedoucímu mé diplomové práce, panu Ing. Bc. Vladimírovi Faltusovi, Ph.D. za perfektní vedení a přístup, obětování svého času pro konzultace a za velmi cenné rady a poznatky.

Dále bych chtěl poděkovat celé své rodině a přátelům za podporu, především rodičům Janě a Zdeňkovi za neutichající podporu, lásku, umožnění studia a možnosti pravidelně nabírat potravinové zásoby. Sestře Petře s rodinou a bratrovi Zdeňkovi za rady ve všech odvětvích života a v neposlední řadě přítelkyni Gabriele za trpělivost a porozumění.

Prohlášení

Nemám závažný důvod proti užívání tohoto školního díla ve smyslu § 60 Zákona č. 121/2000 Sb., o právu autorském, o právech souvisejících s právem autorským a o změně některých zákonů (autorský zákon).

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

V Sepekově dne 28. 5. 2018

.....
Jan Soldát

MOBILNÍ APLIKACE PRO OBJEKTY HROMADNÉHO PARKOVÁNÍ

Diplomová práce

květen 2018

Úplná bibliografická identifikace

Jméno a příjmení autora: Jan Soldát

Název bakalářské práce: Mobilní aplikace pro objekty hromadného parkování

Pracoviště: Ústav dopravní telematiky

Vedoucí bakalářské práce: Ing. Bc. Vladimír Faltus, Ph.D.

Rok obhajoby bakalářské práce: 2018

Počet stran: 75

Příloha: CD-ROM se zdrojovým kódem aplikace, interaktivním návrhem a instalačním souborem aplikace pro platformu Android

Abstrakt: Diplomová práce se zabývá problematikou návrhu a vytvoření hybridní mobilní aplikace pro objekty hromadného parkování. Zabývá se vyhledáním parkovišť, zobrazením míst, řazením a komunikací se serverem. Je popsána tvorba začínající analýzou, pokračující několika koly návrhu grafického uživatelského rozhraní, následnou implementací do prostřední hybridního frameworku Ionic a končící uživatelským testováním a analýzou spolehlivosti.

Abstract: This thesis deals with the problems of design and development of hybrid mobile application for public parking objects. It deals with searching for parking lots, displaying free spots, sorting and communicating with the server et cetera. It begins with analysis, then continues with a several steps of designing graphical user interface and followed by implementation to the actual environment for developing hybrid mobile application named Ionic framework. This thesis is terminated with user testing and analysis of reliability.

Klíčová slova: Aplikace, Parkování, Hromadné parkovací objekty, iOS, Android, Ionic Framework, HTML, CSS, Angular

Keywords: Application, Parking, Public Parking objects, iOS, Android, Ionic Framework, HTML, CSS, Angular

Obsah

Seznam použitých zkratk	6
Úvod	7
1 Analýza	8
1.1 Informační systémy u dopravy v klidu	8
1.1.1 Kategorie dopravy v klidu	8
1.1.2 Cíle	9
1.1.3 Typ informací	10
1.1.4 Prostředky předávání informací	11
1.1.5 Poskytovatelé dat a poskytovaná data	11
1.1.6 Architektura ITS systémů u dopravy v klidu	12
1.2 Mobilní platforma	13
1.2.1 Hlavní operační systémy	13
1.2.2 Přístupy vývoje mobilní aplikace	14
1.3 Geografický informační systém	15
1.3.1 GIS pro mobilní aplikace	15
1.4 Analýza mobilních aplikací	16
1.4.1 eParkomat	16
1.4.2 Parkování v Praze	19
1.4.3 Průvodce parkování v Praze	23
1.4.4 Parken in Mannheim	25
1.4.5 Zhodnocení aplikací	28
2 Návrh	29
2.1 Motivace	30
2.2 Funkční požadavky	30
2.3 Kvalitativní požadavky	30
2.4 Případy užití	31
2.4.1 UC1 – Vybrat zobrazení	31
2.4.2 UC2 – Vyhledat parkoviště	32
2.4.3 UC3 – Setřídít parkoviště	33
2.4.4 UC4 – Zobrazit detail	33
2.4.5 UC5 – Vybrat možnost navigace	34
2.4.6 UC6 – Aktualizovat data	35
2.5 Vztah funkčních požadavků a případů užití	36
2.6 Prvotní návrh uživatelského rozhraní	36
2.7 Druhý krok návrhu uživatelského rozhraní	39
3 Implementace	42
3.1 Výběr přístupu a nástrojů	42
3.1.1 Nástroje vývoje pro hybridní aplikaci	42

3.1.2	<i>Zhodnocení výběru přístupu</i>	43
3.2	Struktura projektu v prostředí Ionic	43
3.3	Návrh GUI v prostředí Ionic Framework	44
3.4	Použité pluginy a jejich použití	52
3.4.1	<i>Seznam důležitých pluginů použitých v navrhované aplikaci</i>	53
3.5	Logika aplikace	53
3.5.1	<i>Získání polohy uživatele</i>	54
3.5.2	<i>Práce se serverem</i>	55
3.5.3	<i>Kontrola připojení k Internetu</i>	57
3.5.4	<i>Aktualizace dat</i>	57
3.5.5	<i>Vyhledávání</i>	58
3.5.6	<i>Úprava barev na základě aktuálních informací</i>	59
3.5.7	<i>Filtrace a řazení dat</i>	59
3.5.8	<i>Uložení posledních navštívených míst</i>	60
3.5.9	<i>Navigace na parkoviště</i>	60
4	Testování a analýza spolehlivosti	61
4.1	Uživatelské testy	61
4.1.1	<i>Zadané úlohy</i>	61
4.1.2	<i>Vyhodnocení odpovědí od participantů</i>	62
4.2	Nedostupnost Internetu	63
4.3	Nedostupnost GPS	64
4.4	Výpadek serveru	64
4.5	Výpadek dodávky dat jednoho parkoviště nebo skupiny parkovišť	64
4.6	Zrušení parkoviště	65
4.7	Návrhy na vylepšení aplikace	65
5	Závěr	66
	Seznam použitých zdrojů	68
	Seznam obrázků	72
	Seznam tabulek	74
	Seznam příloh	75

Seznam použitých zkratek

API	Application Programming Interface – rozhraní pro programování aplikací
CSS	Cascading Style Sheets – kaskádové styly
GIS	Geografický Informační Systém
GPS	Global Positioning System – globální družicový navigační systém
GSM	Groupe Spécial Mobile – Global System for Mobile Communications – Globální Systém pro Mobilní komunikaci
GUI	Graphical User Interface – grafické uživatelské rozhraní
HPO	Hromadné parkovací objekty
HTML	Hypertext Markup Language – hypertextový značkovací jazyk
HW	Hardware
OHA	Open Handset Alliance – uskupení výrobců mobilních telefonů
P+R	Park & Ride
PDZ	Proměnné dopravní značení
PNG	Portable Network Graphics – přenosná síťová grafika
RSS	Rich Site Summary – formát určený pro čtení novinek
RZ	Registrační značka
SW	Software
UCD	User Centered Design – uživatelsky zaměřený design

Úvod

Dopravou v klidu se nazývají procesy parkování a odstavování vozidel mimo jízdní pruhy pozemní komunikace. [1] Velká města trpí nedostatečnou kapacitou parkovacích míst a vznikají tak hromadné parkovací objekty, které ani tak nestačí (viz Praha, naprosto nedostatečná kapacita P+R i parkovišť v centru, případně na sídlištích nebo v okolí zastávek metra nebo v okolí zón placených stání). Nicméně i tak existuje velké množství parkovišť a řidič potřebuje vybrat ten správný objekt, který se nachází co nejbližší cíle cesty a disponuje volnými místy.

Hlavním cílem této práce je vytvořit aplikaci sloužící uživateli k pohodlnému výběru objektu pro parkování svého automobilu. Dílčím cílem je tedy analyzovat informační systémy pro hromadné parkovací objekty, mobilní platformy vzhledem k jejich uživatelské základně a možnosti vývoje mobilní aplikace. Dále analyzovat současný stav podobných mobilních aplikací a na základě těchto kroků provést kompletní návrh požadované aplikace. Ten spočívá v určení uživatelských požadavků vytvářené aplikace, jejich transformování do případů užití a několika kroků návrhu uživatelského rozhraní. Začíná v analogové formě na papíře, pokračuje přes převedení do interaktivního návrhového nástroje a končí vytvořením návrhu ve vybraném vývojovém prostředí. Dále je zapotřebí jednotlivým prvkům „vdechnout život“ logikou a metodami, podle kterých pracují, a vytvořit tak aplikaci sloužící k usnadnění problému výběru parkovacího objektu. Na závěr je vše třeba řádně otestovat a provést analýzu spolehlivosti.

1 Analýza

První část mé diplomové práce je věnována teorii a analýze v oblasti informačních systémů dopravy v klidu, hromadných parkovacích objektů, geografických informačních systémů a mobilních platforem včetně vyprofilování a určení nástrojů potřebných k úspěšné implementaci teoretických znalostí do praktické roviny v podobě mobilní aplikace.

1.1 Informační systémy u dopravy v klidu

Bavíme-li se o jakémkoliv informačním systému, je nutné si říci, jaké jsou jeho funkce, jaké informace a jakými prostředky předává, kde je možné takovýto systém provozovat a proč je to v našem zájmu. Odpovědím na tyto otázky jsou věnovány následující řádky.

Vzhledem k neustálému nárůstu automobilizace je problematika dopravy v klidu velmi důležitá pro všechny obyvatele především větších měst. Průměrně totiž vozidlo 90 % času stojí a zbylých 10 % se poté pohybuje. Z tohoto důvodu je vhodné využívat informační systémy, udávající základní informace o parkovištích a usnadňující nalezení trasy k vhodnému parkovišti. Takovýto systém může pomoci ulevit dopravní vytíženosti komunikací tím, že řidič omezí pobyt na komunikaci, nutný k nalezení parkoviště či volného místa.

1.1.1 Kategorie dopravy v klidu

Dopravou v klidu se nazývají procesy parkování a odstavování vozidel mimo jízdní pruhy pozemní komunikace. K řešení této problematiky je nutno přistupovat centralizovaně v závislosti na urbanistickém charakteru územního celku. Problematika dopravy v klidu lze rozdělit na několik základních kategorií parkování: [1]

- Parkovací kapacity na veřejných komunikacích
- Hlídaná parkoviště
- Záchytná parkoviště typu P+R¹
- Hromadné parkovací kapacity v garážových objektech [1]

V této práci jsou řešena především parkování v hromadných parkovacích objektech a na záchytných parkovištích.

1) **Hromadné parkovací objekty** – Jedná se o takové stavební objekty, které jsou kryté a umožňují odstavení nebo parkování více než tří vozidel. [7] Objekty lze dělit podle několika kritérií. Nejzákladnějším rozdělení je možné učinit podle:

- stavebního řešení
 - jednopodlažní

¹ Park & Ride

- vícepodlažní
- vztahu k terénu
 - nadzemní
 - podzemní
 - kombinované

Hromadné parkovací objekty (dále jen HPO) hrají významnou roli v dopravní infrastruktuře města. Zejména s ohledem na stále vzrůstající tendenci počtu automobilových vozidel ve městech. Proto je přímo žádoucí odstavení či zaparkování vozidla v takovémto objektu. [1], [7]

- 2) **Záchytná parkoviště typu P+R** – Tyto objekty jsou umísťovány v blízkosti navazujících dopravních spojení veřejné dopravy pro veřejnou dopravou, jako jsou například železniční stanice, stanice metra, tramvajové zastávky apod. Jejich dopad z hlediska města, jako dopravního celku, je především ve snížení intenzity osobních automobilů a s tím korelující omezení negativních dopadů dopravy na životní prostředí. Z hlediska uživatele/řidiče je poté výhodou zejména v kratší cestovní době prostředkem MHD, nežli by tomu bylo prostřednictvím individuální dopravy. [1]

Záchytná parkoviště jsou stavebně řešena stejně, jako HPO, liší se pouze ve své funkci. Zatímco HPO jsou většinou cílové místo pro řidiče (co se týče jeho mobility po městě), tak v případě P+R pouze zaparkuje vozidlo a dále pokračuje městskou hromadnou dopravou.

1.1.2 Cíle

V zjednodušené podobě si lze funkce takového systému rozdělit do tří kategorií v závislosti na časovém průběhu jízdy:

- před cestou – uživatel má možnost si pomocí webových portálů nebo mobilních aplikací zjistit základní údaje o poloze parkoviště, typu parkoviště, ceně a dalších rozšiřujících údajích. Může si předem naplánovat trasu a provést rezervaci parkovacího místa.
- během jízdy k objektu – pro uživatele je důležitá informace o obsazenosti a navigace k takovémuto parkovišti, případně historický model, ukazující pravděpodobnost zaparkování v danou hodinu.
- během jízdy v objektu – hlavní funkcí je navedení uživatele uvnitř objektu na volné parkovací místo a s tím související informace o počtu volných míst v jednotlivých částech parkoviště pomocí PDZ.

Nejdůležitějšími funkcemi tedy jsou:

- **informování o obsazenosti** pomocí počtu volných míst, počtu obsazených míst z celkové kapacity nebo binární informací „volno“, „obsazeno“;
- **navigace** uvnitř a vně aplikace;
- **rezervace** parkovacího místa;
- **platba** za parkovací místo;
- **informování o dodatečných informacích**, jako je např. cena za parkování, celková kapacita parkoviště nebo objektu, technická či časová omezení apod.

1.1.3 Typ informací

Informace, které uživatel vozidla může dostat, lze rozdělit na statické a dynamické. Běžně se na parkovací objekt vztahují jak statické, tak dynamické informace.

- **Statické** informace jsou neměnného charakteru. Můžeme jimi například vyjádřit polohu parkovišť, jejich typ nebo, v případě navigování, pevně danou trasu k parkovacímu objektu či základní a v čase neměnné informace o objektu. Příkladem parkovacího objektu, který je uváděn čistě statickými informacemi, mohou být parkovací zóny.
- **Dynamické** informace jsou ty, které se v průběhu času mění a u kterých uživatele zajímá jejich aktuální stav. Mohou to být především údaje o volné kapacitě objektů, některé mimořádné situace nebo navigování vozidla po ideální trase, na základě aktuální dopravní situace. Objekty poskytující tyto informace jsou především hromadné parkovací objekty a záchytná parkoviště.

1.1.4 Prostředky předávání informací

Uživateli lze informace poskytnout pomocí proměnného či statického dopravního značení, mobilních aplikací nebo webového portálu. Přičemž funkce webu a aplikace mají potenciál přesáhnout základní funkce.

Tabulka 1: Klady a zápory prostředků předávání informací

	Statické dopravní značení	PDZ	Web	Aplikace
Klady	Jednoduchost	Možnost zareagování na aktuální stav	Reakce na okamžitý stav	Reakce na okamžitý stav
	Jasně určený význam	Všeobecná známost	Funkční přesah, jako je rezervace, platba	Možnost navigace ve vozidle
	Všeobecná známost	Jednoduchost	Není třeba instalace	Funkční přesah, jako je rezervace, platba
			Možnost přehledně zobrazit větší množství dat	Možnost přehledně zobrazit větší množství dat
Zápory	Nemožnost zareagování na aktuální stav	Nemožnost zobrazit větší množství údajů	Nutnost připojení k Internetu	Nutnost připojení k Internetu
	Nemožnost zobrazit větší množství údajů	Nutnost jejich instalace	Nutnost zobrazení v prohlížeči	Nutnost instalace
	Jednoznačně zaměřené			

1.1.5 Poskytovatelé dat a poskytovaná data

Poskytovatelem dat jsou v případě parkovacích domů jejich vlastníci. Někteří data poskytují veřejnosti, většina však ne. Například TSK Praha nabízí informace na svých webových stránkách pomocí seznamu, mapy, ale i pomocí RSS². TSK Praha je také jediná společnost, od které se mi podařilo získat data v podobném formátu. Ostatní společnosti většinou data o počtu volných míst veřejně neuvádějí. Lze ovšem předpokládat, že každá společnost bude mít vlastní výstupní soubor s daty. Dle sledování webových stránek TSK usuzují, že aktualizace parkovišť probíhá každých 7 minut.

² Rich Site Summary – formát určený pro čtení novinek

```

<item>
  <title>Chodov A</title>
  <description>
    <div id="tsk-parkings-mapbubble" class="parking-bubble parking5340171"><table><tr><td class="first
name">Název: </td><td>Chodov A <span class="isPr isPrFalse" >(není P+R)</span></td></tr> <tr><td
class="first total-places">Celkem míst:</td><td>1205</td></tr><tr><td class="first taken-
places">Obsazených míst:</td><td>97</td></tr><tr><td class="first free-places">Volných míst:</td>
<td>1108</td></tr><tr><td class="first last-updated">Aktualizováno:</td><td>21.05.2018
22:13:16</td></tr></table></div>
  </description>
  <pubDate>Mon, 21 May 2018 20:13:16 GMT</pubDate>
  <dc:date>2018-05-21T20:13:16Z</dc:date>
</item>

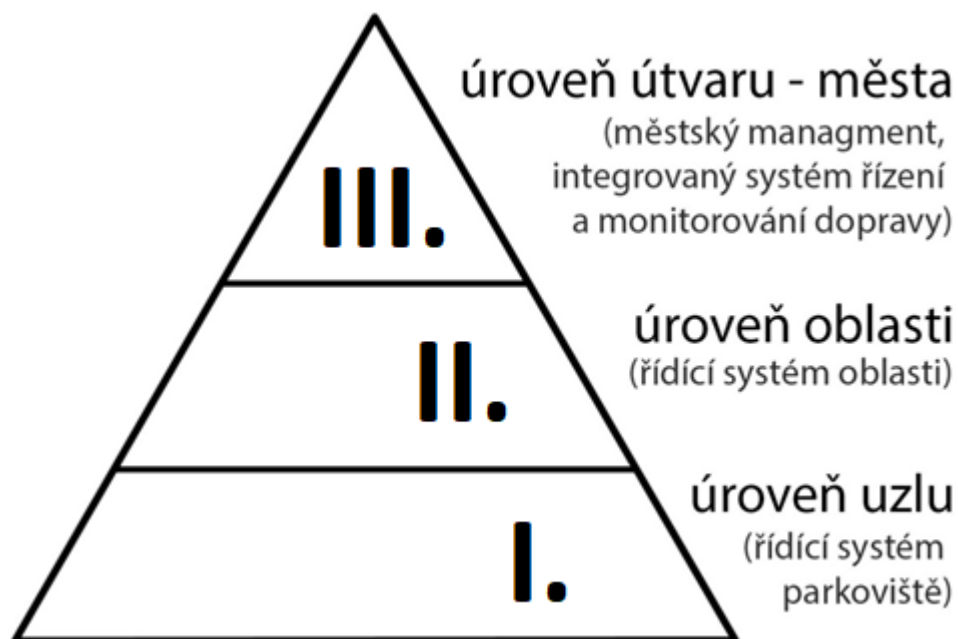
```

Obrázek 1: Ukázka poskytovaných dat od TSK Praha

Dle vzoru TSK Praha jsou data poskytovaná parkovišti:

- název parkoviště;
- celkový počet míst;
- počet obsazených/volných míst;
- datum a čas poslední aktualizace těchto informací.

1.1.6 Architektura ITS systémů u dopravy v klidu



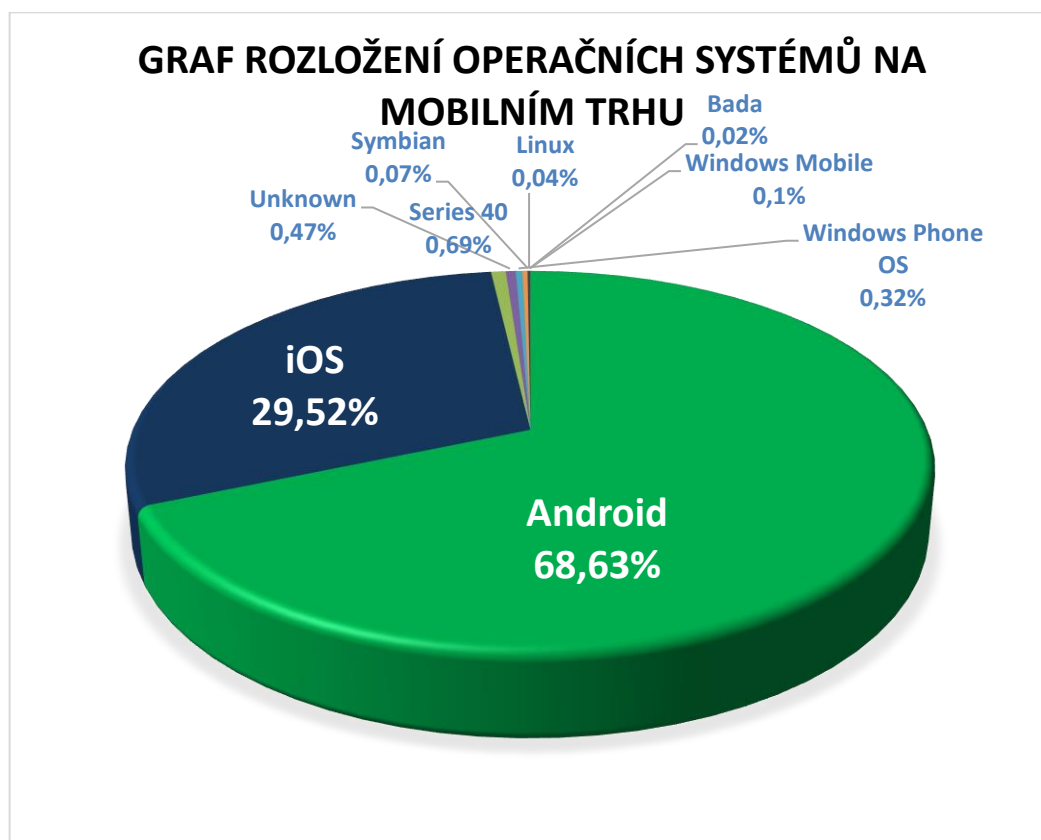
Obrázek 2: Architektura telematického parkovacího systému

Architektura telematických systémů dopravy v klidu je obvykle strukturovaná do tří úrovní.

- **úroveň uzlu** – zde jsou myšlena jednotlivá parkoviště, která shromažďují informace a přenášejí je hierarchicky vyšší oblasti
- **úroveň oblasti** – sdružuje jednotlivá parkoviště, v závislosti na jejich technologii či provozovateli. Shromažďuje data z jednotlivých parkovišť.
- **úroveň útvaru** – může sloužit jako dispečink či krizový management, zejména u větších měst. Může kontrolovat data jednotlivých oblastí.

1.2 Mobilní platforma

Mluvíme-li o mobilních platformách, respektive chytrých telefonech, je třeba si uvědomit nadvládu dvou mobilních platform Android a iOS.



Graf 1.: Graf rozložení mobilních operačních systémů k únoru 2018 [3]

Z grafu jasně vyplývá výše zmíněné tvrzení, proto vývoj aplikace (z pohledu nákladů na vývoj vzhledem k potenciálu využití) má smysl pouze na tyto dvě platformy. Ostatní platformy zabírají něco málo přes 1 %, proto není v jejich možnostech oslovit dostatečné množství uživatelů. Z toho plyne, že vývoj pro tyto platformy je téměř bezpředmětný.

1.2.1 Hlavní operační systémy

Mezi hlavní mobilní operační systémy lze tedy zařadit iOS a Android. Shrnutí těchto dvou operačních systémů se nachází na řádcích níže.

- **Android** – Platforma vlastněná společností Google. Za jeho vývojem stálo konsorcium zakladatelských členů OHA³. Rozšířeno na nejrůznějších značkách mobilních telefonů. Velmi modifikovatelný systém, jedná se o otevřený SW⁴.

³ Uskupení výrobců mobilních telefonů

⁴ Software

- **iOS** – Platforma vlastněná společností Apple. Pouze na zařízeních společnosti Apple. Uzavřený, proprietární SW.

Tabulka 2: Porovnání Androidu a iOS

	Android	iOS
Oficiální programovací jazyky	Java, C++, Kotlin	Swift, Objective-C
Primární vývojové prostředí	Android Studio	Xcode
Obchod	Google Play	App Store
Cena distribuce aplikace	Zdarma	99 \$ za rok
Procentuální zastoupení na trhu	≈ 69 %	≈ 30 %

1.2.2 Přístupy vývoje mobilní aplikace

Při vývoji mobilních aplikací je často využíváno rozdělení do tří základních přístupů vývoje:

- **Nativní** – aplikace vytvořené pro konkrétní platformu. K přenosu na jinou platformu je třeba aplikaci naprogramovat znovu. Nejvhodnější typ pro výkonově náročné aplikace. Přístup k nejaktuálnější verzi API⁵. Tímto přístupem je myšlena tvorba v aplikaci přímo ve vývojovém prostředí dodaném od společnosti Google (tj. Android Studio), případně Apple (tj. Xcode).
- **Webový** – tj. optimalizovaný mobilní web, nutnost přístupu k Internetu. Vhodné pro velmi nenáročné aplikace. Sice odpadá nutnost instalace aplikace, ovšem chybí nativní prvky aplikace a nemožnost přístupu k API. Jde vlastně o přizpůsobené www stránky, takže možnost přístupu k hardwaru je nižší.
- **Hybridní** – využívá principu „Write once, deploy anywhere“. Kód je napsán pouze jednou a potom přeložen na více platform. Toto ušetření času je ovšem vykoupeno nižším výkonem a možnou neaktuálností API. Jde o jakýsi kompromis mezi nativním a webovým přístupem, kdy aplikace je tvořena pomocí webových technologií, ale stále se chová jako aplikace. Tj. je třeba ji nainstalovat a může využívat hardware zařízení. Většinu práce lze simulovat a testovat uvnitř prohlížeče. [11], [12], [13]

⁵ Application Programming Interface

Tabulka 3: Zhodnocení kladů a záporů jednotlivých přístupů vývoje aplikací [11][12][13]

	nativní přístup	webový přístup	hybridní přístup
klady	Možnost využití všech nativních prvků	Jednoduchost a rychlost	Multiplatformovost
	Nejlepší možnost využití HW ⁶ zařízení	Cena	Možnost využití nativních prvků
	Využití pro platformu oficiálně podporovaných jazyků	Multiplatformovost	Cena
zápory	Cena vývoje	Nižší spolupráce s HW	Možná neaktuálnost API
	Pokud aplikaci chceme mít multiplatformní, pak náročnost vývoje a udržování	Nemožnost využívat nativní prvky	Vyšší HW nároky – nehodí se pro velmi náročné aplikace
	Zasáhne menší skupinu uživatelů nežli aplikace vyvíjené zároveň na více platformech	Nutnost být online a používat web v prohlížeči	

1.3 Geografický informační systém

Informační systém můžeme zjednodušeně chápat jako systém pro sběr, přenos, uložení, zpracování a distribuci informací. Geografický informační systém je oproti němu rozšířen o prostorovou složku dat. Definice od společnosti ESRI⁷ zní: „GIS je organizovaný soubor počítačového hardware, software a geografických údajů navržený pro efektivní získávání, ukládání, upravování, obhospodařování, analyzování a zobrazování všech forem geografických informací.“ [10]

1.3.1 GIS pro mobilní aplikace

Mají-li aplikace ve svém repertoáru mapové rozhraní, využívají nejčastěji API třetích stran. API je zkratka pro pojem **A**pplication **P**rogramming **I**nterface, což lze velmi zjednodušeně popsat jako sadu nástrojů a protokolů, které programátor může využít a nemusí vše programovat od úplného začátku. V případě mapových API dostává tedy programátor do ruky mapové podklady, funkce, knihovny atd., s kterými může nadále nějakým způsobem pracovat a využívat ve své aplikaci, tj. nemusí si mapové rozhraní sám vytvářet.

Existuje velké množství rozličných mapových API, jež lze při tvorbě aplikací využít. Níže jsou někteří zástupci přiblíženi.

⁶ hardware

⁷ Americká společnost vyvíjející software pro práci s GIS, ArcGIS

- **Google Maps** – jak již název prozrazuje, jedná se o produkty společnosti Google. Jde o světově nejvíce využívané mapové rozhraní. Jeho využití sahá od aplikací na Android, přes aplikace na iOS až po webové aplikace a multiplatformní aplikace. Nabízí dvě varianty. První je určena pro bezplatné a veřejně dostupné weby a aplikace, a je zdarma. Další, již placenou variantou, jsou aplikace a weby, jež mají omezený přístup, či chtějí sledovat a monitorovat pozici svých zařízení nebo pro komerční weby a aplikace s velkým počtem uživatelů. K dispozici je obsáhlá dokumentace a ukázky kódu. Společnost Google neustále přidává nové funkce a stará se o bezproblémový chod. [15]
- **Leaflet** – velmi rozšířená open source JavaScript knihovna, jejímž autorem je Vladimír Agafonkin. Její hlavní výhodou je široká uživatelská základna, malá velikost a široká možnost rozšíření. Lze použít jak na webových stránkách, tak v mobilních aplikacích. Poskytuje obsáhlou dokumentaci, která je ovšem oproti Google Maps přeci jen stručnější. [16]
- **Here Maps** – mapy od společnosti HERE Global B.V. Nabízejí pět cenově ohodnocených variant pro uživatele. Varianta zdarma je pro aplikace do 15 tisíc požadavků na server za měsíc. Též, mimo jiné, nabízejí API v JavaScriptu. Mají poměrně širokou dokumentaci včetně příkladů, jako je přidání bodů do mapy. [17]

Jak bylo uvedeno výše, existuje velké množství mapových rozhraní, zde jsou uvedeny pouze některé. Mezi další patří například API od společností Esri, OpenLayers, CartoDB, Microsoft atd.

1.4 Analýza mobilních aplikací

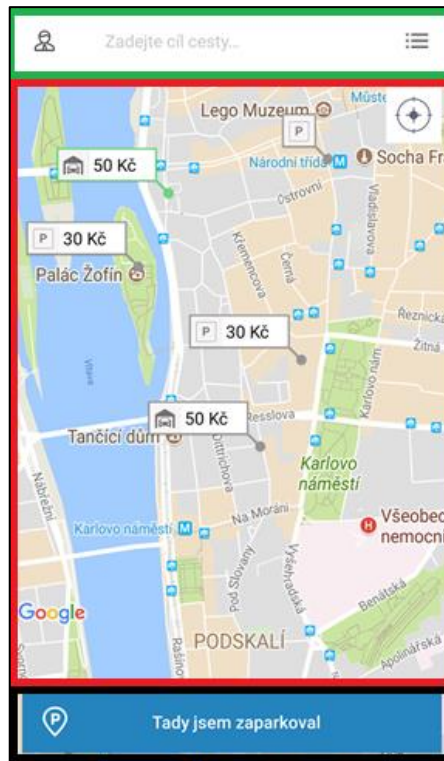
Následující řádky jsou věnovány analýze, vyhodnocení a čerpání podnětů ze současných mobilních aplikací, jež slouží k stejnému, či podobnému účelu, tj. nalezení parkovacího místa v hromadném parkovacím objektu.

1.4.1 eParkomat

Aplikace od společnosti City Smart Parking s.r.o. využívá propojení s „Chytrým autem“ od společnosti T-Mobile. Obsahuje informace jak o parkovacích objektech, tak o zónách. Umožňuje platbu v aplikaci za parkování na veřejných komunikacích v Praze a na komerčních parkovištích v ČR. [3]

Popis

Aplikace využívá upravené mapové rozhraní Google Maps. Hlavní obrazovka je rozdělena na tři základní oblasti viz Obr. 3.



Obrázek 3: Hlavní obrazovka aplikace eParkomat

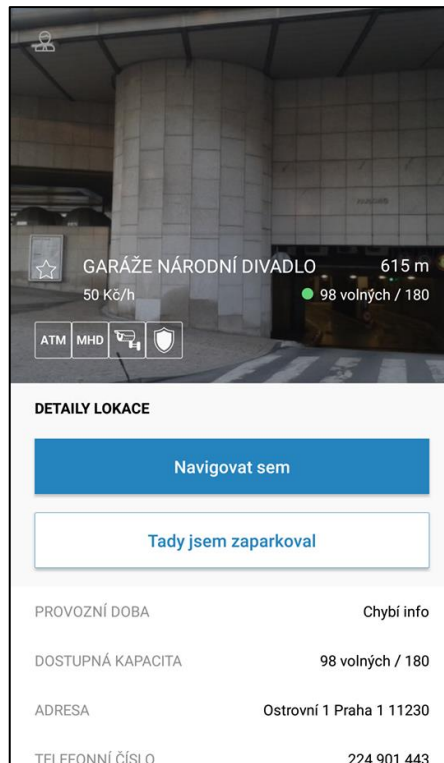
Zeleně zvýrazněná část obsahuje ikony pro přihlášení, vyhledávač pro zadání adresy, kde chce uživatel nalézt místo k zaparkování, a poslední ikona přesměruje uživatele do seznamu míst, se kterými má nějakou zkušenost.

Prostřední (červeně ohraničená) část je mapa. V té jsou umístěny obdélníkové ikony, jejichž obsah vyplňuje grafická informace o typu parkovacího místa (zda se jedná o objekt či nikoliv) a údaj o ceně za jednu hodinu parkování. Celá tato ikonka je ohraničena barvou, v závislosti na údaji o dostupnosti parkování viz Tabulka 4. V drtivé většině případů je ovšem informace o obsazenosti neznámá.

Tabulka 4: Význam barev v aplikaci eParkomat

barva	informace o obsazenosti
zelená	volno
oranžová	několik posledních míst
červená	obsazeno
šedá	neznámý údaj

Po rozkliknutí nějaké ikonky se uživateli zobrazí detail daného parkoviště, viz Obr. 4.



Obrázek 4: Detail parkoviště v aplikaci eParkomat

Detail obsahuje informace o vzdálenosti, obsazenosti a dostupné (volná místa) a celkové kapacitě, provozní době, ceně, adrese parkoviště, kontaktu a rozšiřující informace o parkovišti, jako je například blízkost zastávky MHD nebo informace, je-li parkoviště hlídané. V tomto detailu je možnost zvolit trasování k tomuto parkovišti nebo uvést potvrzení, že zde uživatel již zaparkoval.

Spodní (červeně ohraničená) část náleží velkému tlačítku „Tady jsem zaparkoval“. Po jeho stisknutí je zobrazena nová obrazovka s informací o délce parkování, adrese parkoviště a dá na uživateli k výběru možnosti „Navigovat do...“, „Zaplatit“ nebo „Už jsem pryč“, jež slouží k návratu na úvodní obrazovku a ukončení probíhající akce parkování.

Důležité funkce

- **Platba** – v aplikaci je možnost platby pomocí externí webové aplikace, tuto platbu však nelze provést všude.
- **Navigace k cíli** – při zadání adresy, na kterou chceme navigovat, aplikace nalezne nejbližší parkoviště, ovšem nenacentruje na něj pohled tak, aby uživatel rozhodl, zda se mu nenabízí lepší možnost.
- **Zaparkování na současných souřadnicích** – po zvolení této možnosti se uživateli zobrazí okno s informací, kde parkuje, jak dlouho zde stojí a informace o ceně. Dále má možnost zvolit „Navigovat k autu“, v takovém případě aplikace předá informaci o poloze

předpokládaného stojícího automobilu navigační aplikaci. Druhou možností je „Už jsem pryč“, v tom případě se údaj vymaže a uživatel se vrátí na úvodní obrazovku.

- **Seznam parkovišť** – na této obrazovce uživatel vidí seznam míst seřazených podle kritérií, jako je počet volných míst, cena a vzdálenost. V tomto případě, tj. seřazení aplikací podle volných míst, mi aplikace nabídla množství možností v řádech desítek, ovšem pouze u šesti byla informace o obsazenosti kompletní (počet volných míst / počet celkových míst). U ostatních byl údaj pouze o celkovém množství míst, nebo informace, že počet míst je neznámý.

Zhodnocení

Tabulka 5: Klady a zápory aplikace eParkomat

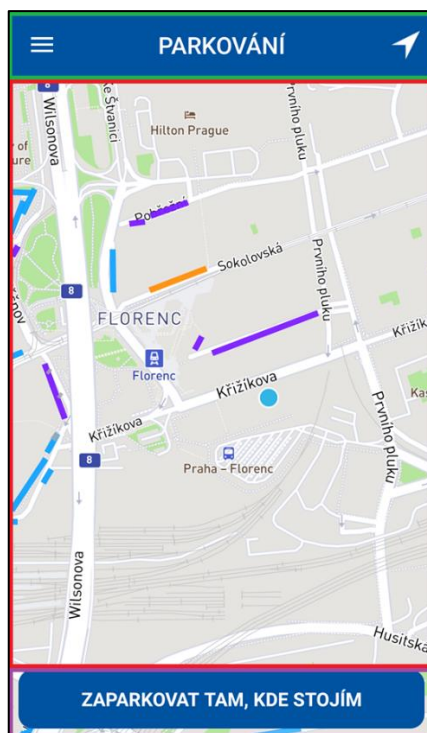
Klady	Zápory
Příjemné uživatelské prostředí	Občasná nepřehlednost jednotlivých prvků
Rychlá a moderní aplikace	Nevycentrování mapy k požadovanému cíli
Mnoho funkcí	Plné využití aplikace pouze s použitím „Chytrého automobilu“
Zobrazení v mapě i v tříděném seznamu	Nepříliš kompletních informací
	Neexistující odhad, zdali bude parkoviště, v době příjezdu volné
	Neexistující informace o aktuálnosti zobrazených dat o obsazenosti

1.4.2 Parkování v Praze

Tato aplikace je, jak již její název napovídá, určena pro usnadnění parkování v hlavním městě. V mapě jsou zobrazeny zóny celého města včetně příslušných parkovacích tarifů platných na daném místě. Zobrazeny jsou také informace o některých P+R parkovištích včetně jejich aktuální obsazenosti. [5]

Popis

Aplikace opět využívá mapového rozhraní a je rozdělena na tři funkční části, viz Obr. 5.



Obrázek 5: Úvodní obrazovka aplikace Parkování v Praze

Vrchní, zeleně označená část, obsahuje dvě tlačítka. Pravé (ve tvaru šipky) slouží k zaměření aktuální polohy zařízení. Tato funkce může pomoci uživatelům již pohybově statických vozidel k identifikaci příslušné zóny, ve které se nacházejí.

Levé tlačítko slouží k vyvolání postranního menu. Položka menu „Virtuální parkovací hodiny“ přeměruje uživatele v současné okně, bez opouštění současné aplikace, pomocí webového prohlížeče v aplikaci, na stránku, kde je možné provést platbu.



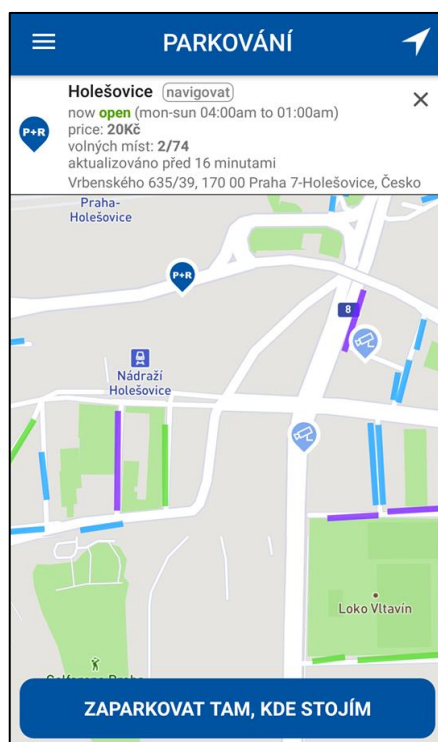
Obrázek 6: Ukázka platby v aplikaci Parkování v Praze

Naproti tomu položka „Parkovací hodiny“ slouží pouze k nastavení upozornění na vypršení časového limitu parkování, tj. uživatel zvolí dobu, po kterou platí jím zaplacené parkování a spustí časovač, který ho třikrát upozorní:

- 20 minut před expirací;
- 10 minut před expirací;
- v momentě expirace.

Položka „Registrační značka“ slouží k zápisu RZ v případě, že si ji uživatel nepamatuje a položka „Parkovací zóny“ je informační stránka s vysvětlením barev jednotlivých zón, případně je zde uvedeno, jakým způsobem lze v těchto zónách platit.

Střední, červeně označená část náleží mapovému rozhraní pracující na API⁸ Google maps. V mapě jsou barevně vyznačeny parkovací zóny a podle klíče v položce menu „Parkovací zóny“ jsou rozlišeny. Dále jsou zde pomocí odlišných ikoněk zobrazeny jednotlivé parkovací domy P+R, parkování zprostředkované službou „Mr. Parkit“ a místa, kde je v provozu kamerový systém, s jehož pomocí lze v aplikaci nahlédnout na fotografii, což se hodí zejména pro zjištění aktuální dopravní situace v případě, že by se uživatel rozhodl vydat po této komunikaci.



Obrázek 7: Detail v aplikaci Parkování v Praze

⁸ Application Programming Interface

Při rozkliknutí ikonky, např. pro P+R, se uživateli na stejné obrazovce zobrazí detail, viz Obr. 7. Ten obsahuje základní, shrnující informace, jako je:

- provozní doba;
- cena;
- poměr počtu volných míst k celkovému počtu míst;
- adresa;
- časový údaj, kdy byla informace naposledy aktualizována (podle výsledků pozorování se tak děje i v delších než půlhodinových intervalech, a to včetně aktualizace fotografií z kamerového systému).

Lehce přehlédnutelná je zde možnost „navigovat“. Aplikace se po její aktivaci uživatele zeptá, zdali chce navigovat v aplikaci nebo využije prostředí Google Maps.

V poslední části je osamocené, velké tlačítko „Zaparkovat tam, kde stojím“. Po jeho stisknutí přejde aplikace do režimu parkování, na mapě je zobrazena ikonka pozice auta a uživatel má možnost okamžitě zapnout časovač „Parkovací hodiny“, poté navigaci k současné pozici automobilu a poslední možností je „Ukončit parkování“, kdy aplikace přejde do výchozího režimu.

Důležité funkce

- **Zaparkovat na současných souřadnicích** – nejzákladnější funkce této aplikace. Adresa mobilního zařízení je uložena a do mapy a přibude ikonka s příslušnými souřadnicemi. Uživatel potom může využít funkci navigace k zaparkovanému vozidlu.
- **Navigace k cíli** – funguje zde pouze tak, že uživatel najde ikonku příslušného parkoviště na mapě a zvolí tlačítko „navigovat“.
- **Platba** – probíhá stejně jako u eParkomatu, přesměrováním na externí webovou aplikaci, rovněž lze tímto způsobem platit jen v určitých místech.
- **Parkovací hodiny** – časovač, jenž může upozornit na konec parkovací doby a zabránit tak případnému odtahování vozidla.

Zhodnocení

Tabulka 6: Klady a zápory aplikace Parkování v Praze

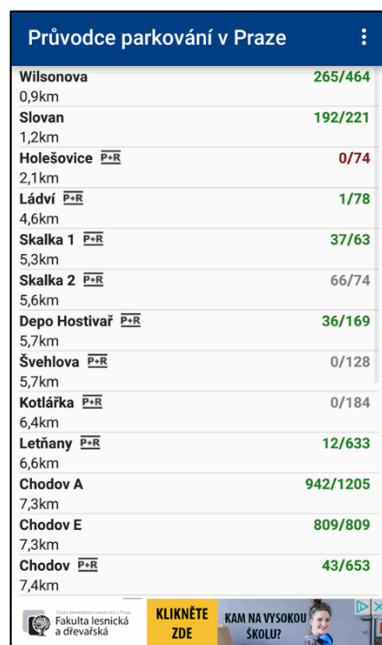
Klady	Zápory
Příjemné uživatelské rozhraní	Absence třídění podle určitých kritérií
Parkovací hodiny	Nelze zadat adresu cíle
Vysvětlení jednotlivých barev zón	Velká časová prodleva při aktualizaci dat
Možnost nahlédnutí do kamerového systému	Neexistující odhad, zdali bude parkoviště, v době příjezdu, volné
Informace o poslední aktualizaci dat	Na mapě není barevně ani jinak znázorněno obsazené parkoviště
Možnost navigace v aplikaci i mimo	Malé množství objektů hromadného parkování
	Malé množství kompletně vyplněných informací o objektech hromadného parkování
	Detail neobsahuje žádné dodatečné informace

1.4.3 Průvodce parkování v Praze

Na první pohled jednoduchá aplikace, sloužící k přehledu P+R parkovišť. Ze všech vybraných aplikací podporuje nejnižší verzi systému Android tj. 2.3 (lze spustit na systému Android 2.3 a vyšší). Poslední aktualizace proběhla 15. října 2016, tudíž na aplikaci se očividně již nepracuje.

Popis

Na první pohled jednoduchá aplikace, jejíž vizuálně nejvýraznější článek je ukryt v dolní části obrazovky, a to v podobě reklamy.



Průvodce parkování v Praze	
Wilsonova	265/464
0,9km	
Slovan	192/221
1,2km	
Holešovice P+R	0/74
2,1km	
Ládví P+R	1/78
4,6km	
Skalka 1 P+R	37/63
5,3km	
Skalka 2 P+R	66/74
5,6km	
Depo Hostivař P+R	36/169
5,7km	
Švehlova P+R	0/128
5,7km	
Kotlářka P+R	0/184
6,4km	
Letňany P+R	12/633
6,6km	
Chodov A	942/1205
7,3km	
Chodov E	809/809
7,3km	
Chodov P+R	43/653
7,4km	

Obrázek 8: Úvodní obrazovka aplikace Průvodce parkování v Praze

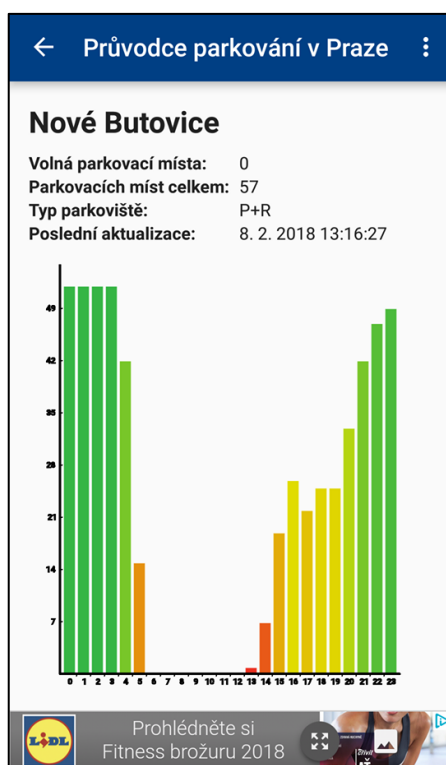
Hlavní obrazovka aplikace je tvořena dvěma funkčními prvky. Tlačítkem pro vyvolání menu a seznamem jednotlivých objektů hromadného parkování, seřazených podle vzdálenosti od uživatele. V seznamu je napsán název parkoviště, případně pomocí ikonky uvedeno, zda se jedná o typ P+R, a v pravé části je poté uveden údaj o volných parkovacích místech vzhledem k celkové kapacitě objektu, který je ještě barevně rozlišen, viz Tabulka 7.

Tabulka 7: Význam barev v aplikaci Průvodce parkování v Praze

barva	Počet dostupných míst [Ø]
Zelená	≥ 1
Červená	$= 0$
Šedá	Dlouho neaktualizovaný údaj

Po výběru některého z parkovišť se zobrazí detail vybraného parkoviště obsahující základní údaje:

- volná parkovací místa;
- celkové množství parkovacích míst;
- typ parkoviště – zde se zobrazuje pouze údaj, zdali se jedná či nejedná o P+R;
- časový údaj o poslední aktualizaci;
- časový diagram – zobrazující průměrný počet volných míst, rozdělených podle hodin, vzhledem ke dni v týdnu.



Obrázek 9: Detail parkoviště v aplikaci Průvodce parkování v Praze

Menu se liší vzhledem k tomu, je-li uživatel na:

- hlavní obrazovce – v tom případě má na výběr pouze položku „Obnovit“, která aktualizuje údaje o volných parkovacích místech;
- detailu záznamu – má dvě možnosti:
 - „Navigovat“ – uživatel je přesměrován do aplikace Google Maps (případně jiné navigační aplikace dle svého výběru), kde je zadána adresa parkoviště jako cíl a jako počátek je vzata aktuální poloha;
 - „Ukázat mapu“ – dojde k přesměrování, jako v prvním případě, ovšem není zadána trasa, pouze je na mapě zobrazena poloha vybraného objektu.

Důležité funkce

- **Seřazený seznam** – seznam parkovišť seřazen podle vzdálenosti, nemožnost jiného třídění.
- **Navigace k cíli** – probíhá pouze v externí aplikaci.
- **Histogram** – zobrazuje průměrnou hodinovou obsazenost v příslušný den týdne.
- **Poslední aktualizace** – možnost obnovit data uživatelem a informace o aktuálnosti dat.

Zhodnocení

Tabulka 8: Klady a zápory aplikace Průvodce parkování v Praze

Klady	Zápory
Jednoduchost	Velmi malé množství rozšiřujících informací
Přehlednost	Zobrazení informací jako volná/celková místa může být pro uživatele zavádějící
Informace o aktuálnosti dat	Některá data se neaktualizují
Histogram	Může působit nepřehledně
Řazení dle vzdálenosti	Nepříliš hezký a moderní interface
Možnost aktualizace dat	Neindikuje, že kapacita se blíží naplnění
	Absence informace o ceně
	Absence mapového rozhraní

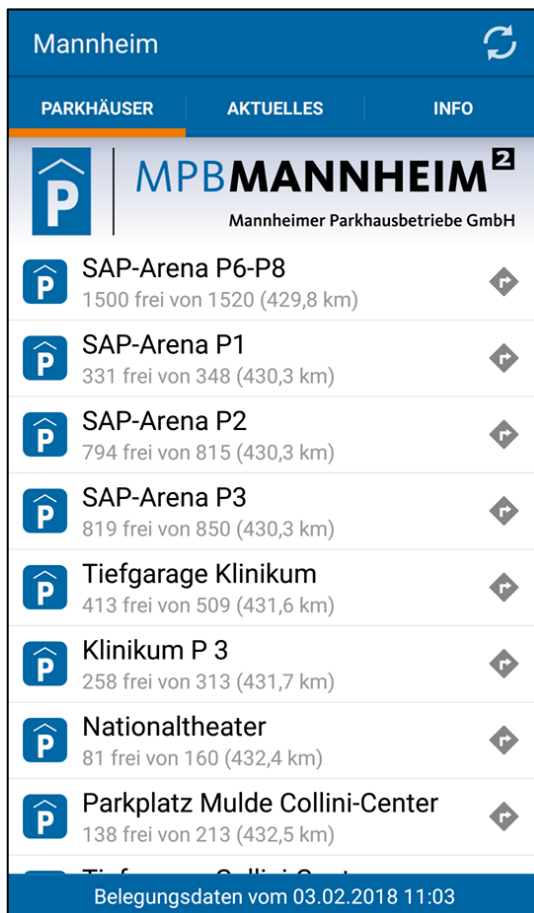
1.4.4 Parken in Mannheim

Aplikace určená k nalezení místa v německém městě Mannheim. Aplikace zobrazuje nejbližší parkoviště vzhledem k aktuální poloze a případně předá souřadnice navigačnímu SW. [14]

Popis

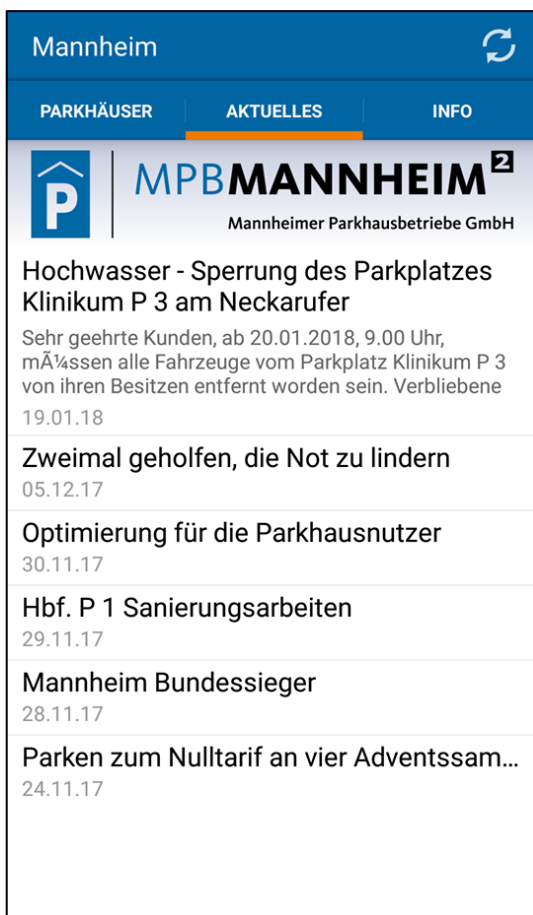
Aplikace se skládá z hlavní neměnné ovládací části, kde je tlačítko pro aktualizaci dat (které ovšem po stisknutí nereaguje) a menu pro přechod mezi jednotlivými obrazovkami, a dále z části samotného obsahu, která se mění v závislosti, na jaké obrazovce se uživatel nachází.

První položka menu je „PARKHÄUSER“. Zde je zobrazen seznam parkovišť včetně údaje o počtu volných míst z celku a vzdálenosti uživatele od daného objektu.



Obrázek 10: Hlavní obrazovka aplikace Parken in Mannheim

U každé položky je možnost ji rozkliknout, případně zvolit ikonku pro navigaci k cíli. V detailu parkoviště lze vidět upřesňující informace, včetně cenových tarifů, viz Obr. 11. Na záložce „AKTUELLES“ jsou zobrazeny aktuální zprávy, týkající se změn ohledně parkovišť (např. pro uživatele parkovacího objektu) a na poslední záložce „INFO“ jsou základní informace o aplikaci. Nutno ještě zmínit, že data aplikace jsou velmi dlouhou dobu neaktualizovaná a poslední aktualizace dat proběhla 3.2.2018 v 11:03. Tzn., že aplikace ztratila význam a uživatel nemá důvod takto neaktuální aplikaci používat.



Obrázek 11: Detail aplikace Parken in Mannheim

Důležité funkce

- **Seřazený seznam** – seznam parkovišť seřazen podle vzdálenosti, nemožnost jiného třídění.
- **Poslední aktualizace** – možnost obnovit data uživatelem a informace o aktuálnosti dat.
- **Navigace k cíli** – probíhá pouze v externí aplikaci.

Zhodnocení

Tabulka 9: Klady a zápory aplikace Parken in Mannheim

Klady	zápory
Jednoduchost	Velmi malé množství rozšiřujících informací
Přehlednost	Absence mapového rozhraní
Teoretická možnost informace o aktuálnosti dat	Data se neaktualizují
Možnost aktualizace dat	Vývoj na aplikaci byl podle všeho ukončen
Řazení dle vzdálenosti	Nepříliš hezký a moderní interface
	Neindikuje, že kapacita se blíží naplnění

1.4.5 Zhodnocení aplikací

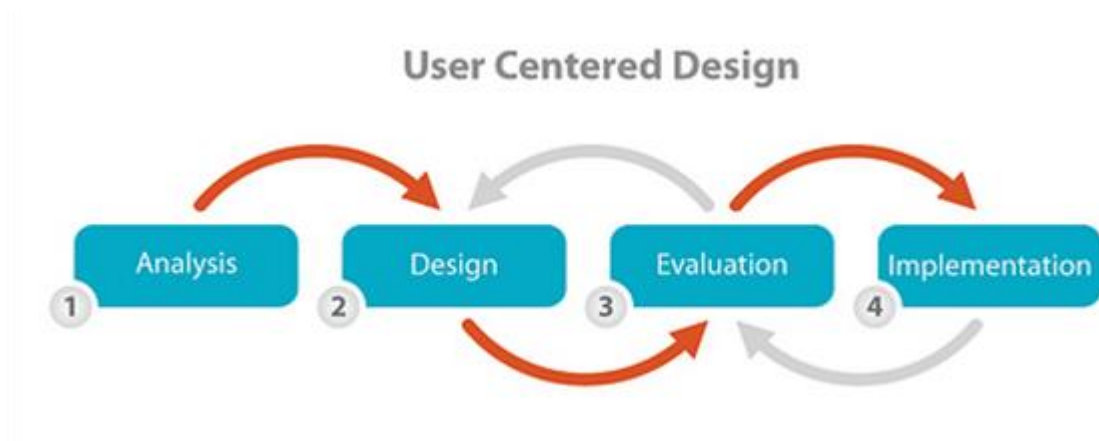
Na základě analýzy aplikací vzhledem k zadání a představě o požadovaném návrhu, jsou z analyzovaných aplikací vyextrahovány některé funkce a pomocí Tabulka 10 porovnány. Z porovnání mimo jiné vyplývá, že pouze jedna aplikace nabízí jak mapové, tak seznamové rozhraní a žádná aplikace neumožňuje vícekriteriální filtr. Historickými daty se zabývá pouze jedna ze zkoumaných aplikací, která je ovšem oproti ostatním aplikacím zase skoupá na ostatní obsah.

Tabulka 10: Zhodnocení aplikací

	eParkomat	Parkování v Praze	Průvodce parkováním v Praze	Parken in Mannheim
Seznamové rozhraní	+	-	+	+
Mapové rozhraní	+	+	-	-
Zajištění navigace k cíli	+	+	+	+
Seřazený seznam, dle vybraného kritéria	+	-	-	-
Vícekriteriální možnost filtrace	-	-	-	-
Historická data	-	-	+	-
Manuální aktualizace dat	-	-	+	+

2 Návrh

Při návrhu aplikace jsem vycházel z přístupu UCD (**U**ser **C**entered **D**esign), což je, jak z názvu vyplývá, uživatelsky orientovaný přístup, v kterém se klade důraz na uživatele a jejich potřeby. Proto byla prvním krokem již samotná analýza viz kap. 1.



Obrázek 12: Princip UCD [18]

Samotný postup návrhu (respektive tvorby celé diplomové práce) jsem prováděl následovně:

Tabulka 11: Postup návrhu a tvorby

Bod	Číslo kapitoly, kde je bod řešen
Definice uživatelských požadavků, na základě provedené analýzy	1 & 2
Definice případů užití včetně jejich specifikace a zhodnocení implementace všech uživatelských požadavků	2
Tvorba prvního grafického návrhu (na papíře)	2
Tvorba návrhu v interaktivním návrhovém programu, upraveného na základě zhodnocení předchozího kroku	2
Převedení návrhu do prostředí, ve kterém bude aplikace vznikat a úprava návrhu na základě zhodnocení předchozího kroku a specifik tohoto prostředí	3
Implementace funkcí a logiky aplikace	3
Vytvoření prototypu	3
Testování prototypu a analýza spolehlivosti	4
Zhodnocení prototypu	4

Tato kapitola se věnuje návrhu aplikace, avšak z globálního hlediska je de facto celá tato diplomová práce návrhem aplikace, který začíná analýzou a končí implementací a testováním. Ovšem v této kapitole jsou řešeny pouze body týkající se návrhu v užším slova smyslu, zbylé

body jsou řešeny v ostatních kapitolách této práce, jak uvádí Tab. 11, která má poukázat na následnost kroků a jejich rychlý přehled.

2.1 Motivace

Motivace vytvoření této aplikace je, že žádná z analyzovaných aplikací neobsahuje všechny funkce, které považuji za důležité zároveň, tj. mapové i seznamové rozhraní, možnost filtrace na základě více než jednoho parametru atd. Další motivací bylo samotné vypracování takovéto aplikace a možnost poměrně jednoduchého napojení na reálná data z parkovišť a samozřejmě pokus o ulehčení procesu hledání parkovacího místa co možná nejširší skupině uživatelů.

2.2 Funkční požadavky

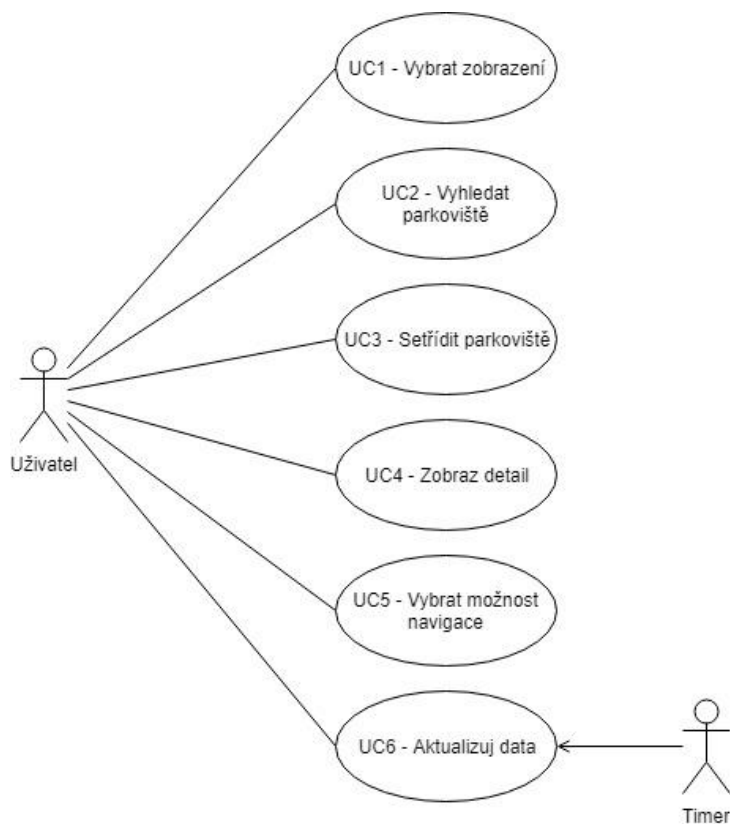
- FP01 – Musí obsahovat aktuální údaje o obsazenosti.
- FP02 – Měla by zde být možnost řazení podle zvolených kritérií.
- FP03 – Zobrazení parkovišť na mapě i v seznamu.
- FP04 – Zřetelné rozlišení dle obsazenosti.
- FP05 – Měla by zde být možnost zobrazení dodatečných informací o jednotlivých parkovištích.
- FP06 – Uživatel by měl mít možnost vyžádat si aktualizaci dat.
- FP07 – Aplikace by měla umožnit uživateli využít jeho oblíbenou navigační aplikaci k vedení k cíli nebo ho k cíli sama navigovat.
- FP08 – Uživatel by měl mít možnost vyhledávání parkoviště dle cíle trasy i aktuální polohy.

2.3 Kvalitativní požadavky

- KP01 – Aplikace musí pracovat rychle a zbytečně uživatele nezdržovat.
- KP02 – Aplikace by měla aktualizovat data každých 5 minut. Data starší než 15 minut by měla zřetelně označit.
- KP03 – Aplikace musí obsahovat dostatečné množství důležitých dat, díky kterým se uživatel rozhodne, kam pojede zaparkovat, jako je vzdálenost, cena, obsazenost a celkový počet míst.
- KP04 – Aplikace nesmí uživatele zahltit nepodstatnými informacemi.

2.4 Případy užití

Obrázek diagramu případů užití (v angličtině Use Case) ukazuje, jaké funkce systém obsahuje a kým jsou iniciovány. Specifikace poté rozvíjí jednotlivé případy užití.



Obrázek 13: Navrhovaný Use Case diagram aplikace

2.4.1 UC1 – Vybrat zobrazení

Krátký popis

Umožňuje vybrat zobrazení pomocí mapy nebo seznamu.

Aktéři

- Uživatel
- Systém

Podmínky pro spuštění

Aplikace musí být spuštěna.

Základní tok

1. Systém vygeneruje mapu, navigační ikonky pro volbu mezi seznamem a mapou a řádek pro vyhledávání.
2. Uživatel vybere seznamové rozhraní.

3. Systém vygeneruje příslušné zobrazení.

Alternativní tok

2.1. Bod 1 základního toku.

2.2. Uživatel nechá zvolenu mapu, tok pokračuje na bod 3 základního toku.

Podmínky pro dokončení

Korektně zobrazené zvolené rozhraní.

2.4.2 UC2 – Vyhledat parkoviště

Krátký popis

Umožňuje najít a vybrat vhodné parkovací místo.

Aktéři

- Uživatel
- Systém

Podmínky pro spuštění

Data již musejí být načtena v příslušném rozhraní.

Základní tok

1. Systém vygeneruje parkovací místa, včetně informací o jejich stavu, vzdálenosti a ceně, v blízkém okolí uživatele do mapy.
2. Uživatel vybere příslušné parkoviště z nabízených možností.
3. Systém nabídne možnost navigace do zvoleného parkoviště.

Alternativní tok 1

2.1. Systém vygeneruje parkovací místa, včetně informací o jejich stavu, vzdálenosti a ceně, v blízkém okolí uživatele pomocí seznamu. Tok pokračuje bodem 2 základního toku.

Alternativní tok 2

3.1. Bod 1 základního toku.

3.2. Uživatel vyhledá parkoviště pomocí vyhledávání cílové destinace své jízdy.

3.3. Systém vycentruje mapu na požadovanou cílovou destinaci. Tok pokračuje bodem 2 základního toku.

Alternativní tok 3

4.1. Bod 2.1 alternativního toku 1.

4.2. Bod 3.2 alternativního toku 2.

4.2. Systém zobrazí seznam na základě požadované cílové destinace. Tok pokračuje bodem 2 základního toku.

Podmínky pro dokončení

Vygenerování nabídky navigace do cílové destinace.

2.4.3 UC3 – Setřídít parkoviště

Krátký popis

Umožňuje setřídít parkoviště dle vybraného kritéria či na základě více kritérií parkoviště filtrovat.

Aktéři

- Uživatel
- Systém

Podmínky pro spuštění

Zobrazení vygenerovaných parkovišť v rozhraní seznamu.

Základní tok

1. Systém vygeneruje parkoviště pomocí seznamu a prvky pro výběr kritérií.
2. Uživatel vybere kritérium, podle kterého si přeje parkoviště zobrazit.
3. Systém setřídí a vypíše parkoviště.

Alternativní tok

2.1. Bod 1 základního toku.

2.2. Uživatel vybere více kritérií, podle kterých si přeje vypsát seznam.

2.3. Systém provede filtraci seznamu na základě splnění všech požadovaných parametrů. Tok pokračuje bodem 3 základního toku.

Podmínky pro dokončení

Seznam seřazený dle uživatelových potřeb.

2.4.4 UC4 – Zobrazit detail

Krátký popis

Umožňuje zobrazit detailní popis parkoviště.

Aktéři

- Uživatel
- Systém

Podmínky pro spuštění

Vygenerovaná parkoviště v příslušném zobrazení.

Základní tok

1. Systém vygeneruje parkoviště v příslušném zobrazení.
2. Uživatel zvolí parkoviště.
3. Systém nabídne možnost zobrazení detailu.
4. Uživatel zvolí zobrazení detailu
5. Systém zobrazí detail.

Podmínky pro dokončení

Vygenerovaný detail parkoviště

2.4.5 UC5 – Vybrat možnost navigace

Krátký popis

Nabízí možnost navigace vně i uvnitř aplikace.

Akteři

- Uživatel
- Systém
- Externí navigační aplikace

Podmínky pro spuštění

Bod 3 základního toku UC2.

Základní tok

1. Systém vygeneruje možnost výběru navigace.
2. Uživatel zvolí možnost navigace v aplikaci.
3. Systém naviguje uživatele do cíle.

Alternativní tok

- 2.1. Bod 1 základního toku.
- 2.2. Uživatel zvolí možnost navigace v externí aplikaci.
- 2.3. Systém předá souřadnice cílové destinace externí aplikaci.
- 2.4. Externí aplikace naviguje uživatele do cíle.

Podmínky pro dokončení

Uživatel je naveden do cíle své cesty.

2.4.6 UC6 – Aktualizovat data

Krátký popis

Aktualizuje informace o parkovištích.

Aktéři

- Uživatel
- Systém
- Timer (časovač, hodiny)

Podmínky pro spuštění

Vygenerovaná parkoviště v příslušném rozhraní

Základní tok

1. Systém načte informace ze serveru a zobrazí je.
2. Timer pošle každých 5 minut dotaz na server.
3. Systém načte informace ze serveru a zobrazí je.

Alternativní tok

2.1. Bod 1 základního toku.

2.2. Uživatel vyžádá aktualizaci dat. Tok pokračuje bodem 3 základního toku.

Podmínky pro dokončení

Zobrazení aktualizovaných dat.

2.5 Vztah funkčních požadavků a případů užití

Níže uvedená tabulka ukazuje, že všechny funkční požadavky jsou zahrnuty do návrhu systému. Navržený systém by tedy měl obsahovat vše dle svého zadání.

Tabulka 12: Vztah funkčních požadavků a případů užití

Požadavky	Případy užití					
	UC1	UC2	UC3	UC4	UC5	UC6
FP01		+				+
FP02			+			
FP03	+					
FP04		+	+			
FP05			+	+		
FP06						+
FP07					+	
FP08		+				

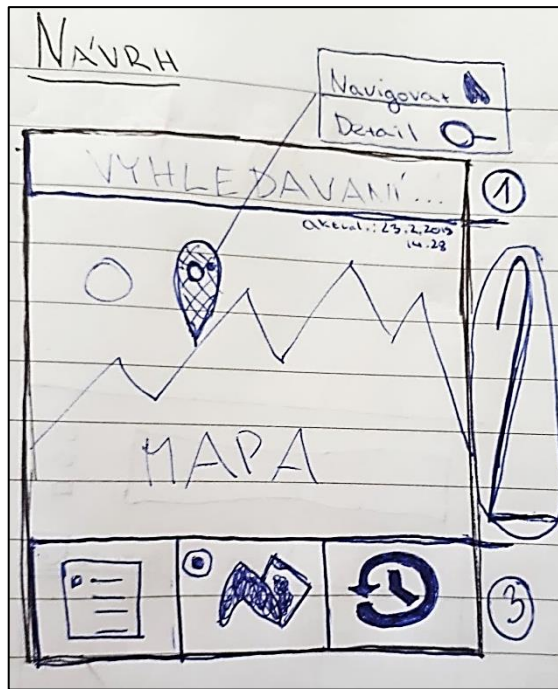
2.6 Prvotní návrh uživatelského rozhraní

První kolo návrhu jsem nejdříve provedl zákresem na papír. Tento krok sloužil především k ujasnění chápání systému a hrubému návrhu toho, jak by měla aplikace vypadat a v ní byly obsaženy všechny požadované funkce a k celkovému rozmyšlení nad strukturou aplikace. Bylo nutné si uvědomit jaké je minimální množství obrazovek, aby byly splněny všechny požadavky co možná nejefektivněji.

Po zvážení požadavků jsem určil, že aplikace bude k naplnění funkčních požadavků vyžadovat celkem tři hlavní odlišné obrazovky/stránky a jednu rozšiřující. Tyto tři hlavní obrazovky jsou:

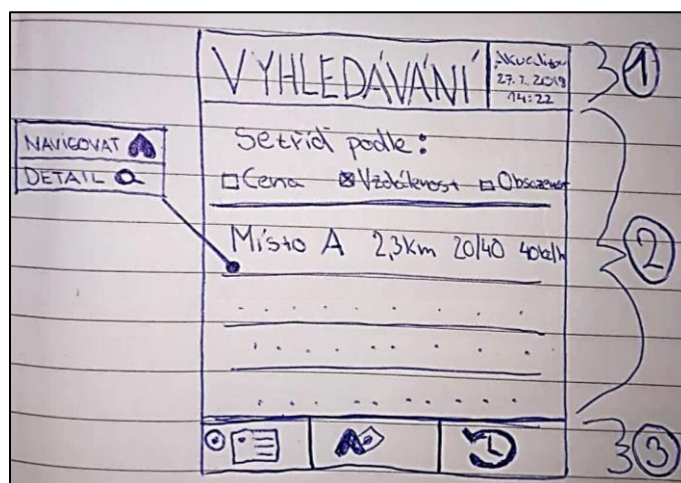
- Obrazovka se zobrazením parkovišť na mapě.
- Obrazovka se zobrazením parkovišť v seznamu.
- Obrazovka se zobrazením několika posledních navštívených míst ve formě seznamu.

Zbývající, rozšiřující obrazovkou je poté detail parkoviště.



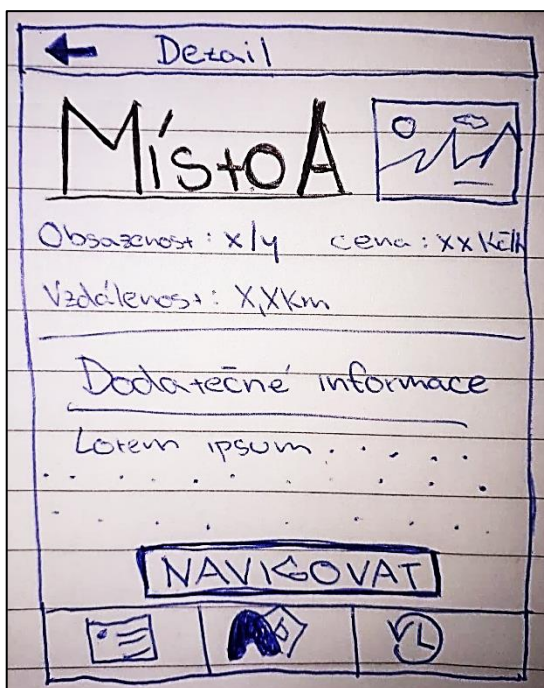
Obrázek 14: Prvotní návrh úvodní obrazovky

Úvodní obrazovka byla navržena následovně, viz Obr. 14. Aplikace je rozdělena do tří pomyslných částí: hlavičky, obsahu a zápatí. V hlavičce je umístěn vyhledávač, který, z důvodu své důležitosti, zaplňuje celou šířku okna. Co se týče okna s obsahem, v pravém horním rohu je umístěna informace o aktuálnosti dat ve formátu DD.MM.RRRR, HH:MM. Dále je zde umístěna mapa, která zaplňuje celou část obsahu. V mapě jsou umístěny jednotlivé značky na souřadnicích daných parkovišť. Po rozkliknutí značky by se mělo objevit okno s výběrem mezi navigací na dané parkoviště a zobrazením detailu daného parkoviště. Třetí částí je poté menu tvořeno ze seznamu, mapy a historie s tím, že záložka, na které se uživatel nachází, je nějakým způsobem vybrána a označena. Toto menu je na každé z oněch tří hlavních obrazovek.



Obrázek 15: Prvotní návrh obrazovky se seznamem

Další, neméně důležitá, je obrazovka se seznamem parkovišť. Ta je opět rozdělena do tří částí. Záhloví je tvořeno stejným polem pro vyhledávání, ale zároveň obsahuje, z důvodu ušetření místa v části obsahu a zpřehlednění zobrazovaných dat, i informaci o aktuálnosti dat v již zmíněném formátu (DD.MM.RRRR, HH:MM). Prostřední část byla navržena tak, že pod záhlavím je možnost řazení pomocí checkboxů⁹ na základě tří důležitých parametrů: cena, vzdálenost a obsazenost, tj. údajů pro uživatele nejdůležitějších. Pod těmito prvky se ukrývá seznam jednotlivých parkovišť. Ten je tvořen jednotlivými položkami a každá z těchto položek je tvořena názvem parkoviště, vzdáleností parkoviště od současné polohy uživatele, počtem obsazených míst vzhledem k celkovému počtu míst a cenou za jednu hodinu pobytu na parkovišti. Po zvolení příslušné položky se opět objeví okno s výběrem mezi navigací na dané parkoviště a zobrazení detailu daného parkoviště. Třetí spodní část je, jako v případě mapy, hlavní menu aplikace. Historie byla zamýšlena identicky s tím rozdílem, že by ukazovala posledních pět navštívených parkovišť a neobsahovala by řazení.



Obrázek 16: Prvotní návrh obrazovky detailu

Horní část obrazovky je tvořena pomocí zpětného tlačítka na předchozí obrazovku (detail, mapa či historie) a popisku stránky. Prostřední část je tvořena z informací o parkovišti. První třetinu zabírají nejdůležitější informace: název parkoviště, údaje o obsazenosti, ceně a vzdálenosti, dalším prvkem je poté obrázek daného parkoviště. Pod tímto segmentem je poté část na dodatečné informace o parkovišti a tlačítko „NAVIGOVAT“, které uživateli umožní navigaci na zvolené parkoviště.

⁹ zaškrťovací políčko

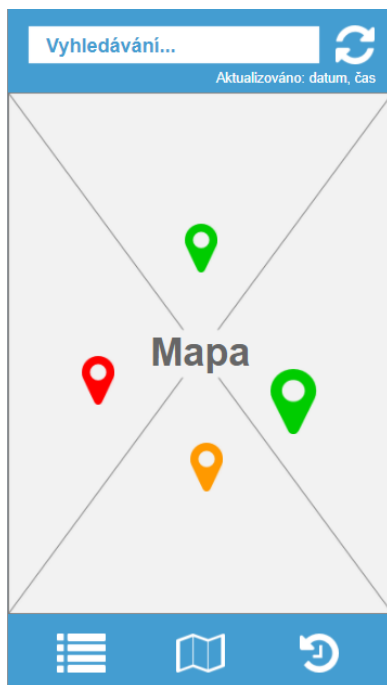
2.7 Druhý krok návrhu uživatelského rozhraní

Druhý krok návrhu grafického uživatelského rozhraní (GUI¹⁰) probíhal pomocí interaktivního návrhového softwaru (výstup z toho softwaru je součástí přílohy). Nejprve jsem zhodnotil prvky předchozího návrhu vzhledem k použití na mobilní aplikaci a jejich praktičnosti, srozumitelnosti a efektivitě zobrazení a znovu si připomněl uživatelské požadavky na systém, tak aby se docílilo, nebo alespoň co nejvíce přiblížilo, jejich plnému vyhovění. Dále jsem vytvořil i hlavní logo aplikace. To bude použito jako tzv. splash screen (úvodní obrazovka), tj. obrazovka, která se uživateli zobrazí po spuštění během doby načítání aplikace.



Obrázek 17: Hlavní logo a zároveň splash screen aplikace

Rozložení obrazovek a jejich počet zůstal oproti „papírové“ verzi z prvního kola návrhu stejný. Provedené změny se týkají jednotlivých prvků, zejména pak jejich vhodnějšího umístění.



Obrázek 18: Hlavní obrazovka aplikace v druhém kole návrhu

¹⁰ Graphical User Interface

Úvodní obrazovka je stránka s mapou. Tato obrazovka se skládá stále ze tří částí, kde zápatí zůstalo v celé aplikaci beze změn. První část, záhlaví, stále obsahuje pole na vyhledávání cíle uživatelské trasy. Ovšem nově sem byla přesunuta informace o aktuálnosti dat a tlačítko pro manuální aktualizaci dat, tak, aby bylo vyhověno FP06 z funkčních požadavků.

Druhá, prostřední část se skládá z mapového podkladu, který je v bodech zájmu, tedy v zeměpisných souřadnicích parkovišť, osazen barevně rozlišenými značkami. Jejich barva je určena na základě obsazenosti parkoviště podle následujícího klíče:

Tabulka 13: Barevné rozlišení značek

Barva	Poměr počtu obsazených míst vzhledem k celkovému počtu míst
zelená	< 0; 0,9)
oranžová	< 0,9; 1)
červená	{1}

Při zvolení značky je uživateli nabídnuto okno k zobrazení detailu či navigaci na toto parkoviště.

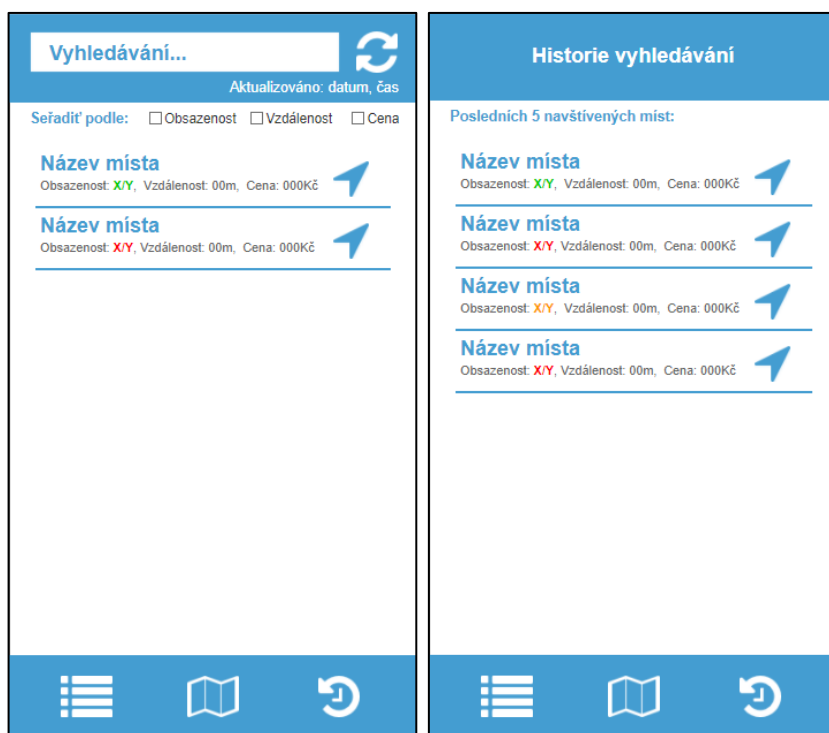


Obrázek 19: Detailní informace o aplikaci v druhém kole návrhu

V případě zobrazení **detailu** je uživatel na něj přesměrován, zde se může dočíst více rozšiřující informace. Obrazovka je řešena stejně, co se týče záhlaví a zápatí, pouze obsah se dočkal kosmetické obměny. Obrázek ilustrující příslušné parkoviště byl přesunut nalevo. Uprostřed se nachází velkým písmem název parkoviště a nejdůležitější informace jsou kvůli přehlednosti řazeny pod sebe. Obsazenost je barevně rozlišena na základě stejného klíče jako

značky na mapě viz Tabulka 13. Dále byla přidána informace o stáří údajů vztažených k danému parkovišti. V pravé části je poté ikonka sloužící k navigaci na parkoviště. Ta byla přesunuta z důvodu jejího většího zdůraznění a tak, aby k ní byl snadnější přístup.

Další stránkou je obrazovka se **seznamem** parkovišť. Ten se opět skládá z hlavičky, jako v obrazovka s mapou, která doznala stejných změn tak, aby byl její vzhled v aplikaci stálý, navigačního panelu v dolní části a samotného seznamu. Jednotlivé prvky seznamu jsou poté tvořeny názvem místa, základními údaji o obsazenosti (opět barevně rozlišeny viz Tabulka 13), vzdálenosti, ceně a tlačítkem pro spuštění navigace. Při stlačení položky mimo tlačítko navigace přejde uživatel na detail jednotlivého místa. Nad seznamem má uživatel ještě možnost seřadit parkoviště na základě obsazenosti, ceny nebo vzdálenosti, tyto prvky byly přesunuty na stejný řádek jako jejich štítek z důvodu úspory místa.



Obrázek 20: Seznam parkovišť a historie navštívených míst v druhém kole návrhu

Poslední obrazovkou je **historie**. Ta je tvořena seznamem s tím rozdílem, že tento seznam zobrazuje posledních pět navštívených destinací seřazených podle poslední návštěvy a neobsahuje možnost řazení podle jiných kritérií. Dále také neobsahuje možnost vyhledávání, není zde možnost seznam aktualizovat a není zobrazen čas poslední aktualizace.

3 Implementace

Tato kapitola je věnována výběru přístupu a nástrojů pro implementaci návrhu z předchozí kapitoly a samotné práci s vybranými nástroji, jako je vytvoření GUI a logiky systému.

3.1 Výběr přístupu a nástrojů

Vzhledem k výše uvedeným údajům v kap. 1.2.2 na str. 14, jsem se rozhodl pro vytvoření hybridní aplikace, protože na rozdíl od nativní aplikace využívá principu: „Write once, deploy anywhere“, dokáže tedy oslovit větší počet uživatelů, což dokazuje Graf 1 na str. 13 a to je v tomto případě vítaná volba, jak bylo již uvedeno a doporučeno v kap. 2.1. Oproti webové aplikaci dovoluje využívat nativní prvky a pro uživatele se stále tváří jako aplikace, nikoliv web.

3.1.1 Nástroje vývoje pro hybridní aplikaci

V současné době je velké množství nástrojů pro vývoj hybridní aplikace. Většina je založena na jazyku JavaScript a ke své činnosti většinou využívá webové technologie HTML, CSS, ale existuje i mnoho dalších nástrojů založených například na jazyku Java či C#. Liší se především v početnosti vývojářské komunity, množstvím pluginů¹¹ a již zmíněným jazykem. Níže uvedená Tabulka 14 uvádí některé z nejvýznamnějších nástrojů a kritéria, podle kterých byly vybírány. Tato kritéria jsou sice poněkud subjektivní, ale plně vystihují důvody výběru. Tím nejpodstatnějším byly jazyky, v kterých probíhá vývoj, dále zdali je lze používat zdarma, také jak je aktivní komunita okolo těchto nástrojů a zároveň skutečnost, zdali jsem se již s některým nástrojem setkal.

Tabulka 14: Porovnání nástrojů pro vývoj hybridních aplikací

Jméno nástroje	Jazyk	Možnost bezplatného vývoje	Dostupnost důležitých pluginů zdarma	Uživatelská podpora	Vývojářská podpora	Předchozí zkušenost
Adobe PhoneGap [25]	HTML5, CSS3, JavaScript	ANO	ANO	Široká	Široká	NE
Codename One [27]	Java	ANO	ANO	Méně rozšířené	Široká	NE
Ionic [29]	HTML5, CSS3, Angular	ANO	ANO	Velmi široká	Velmi široká	ANO
React Native [28]	JavaScript	ANO	ANO	Široká	Velmi široká	NE
Xamarin [26]	C#	NE	NE	Středně široká	Velmi široká	NE

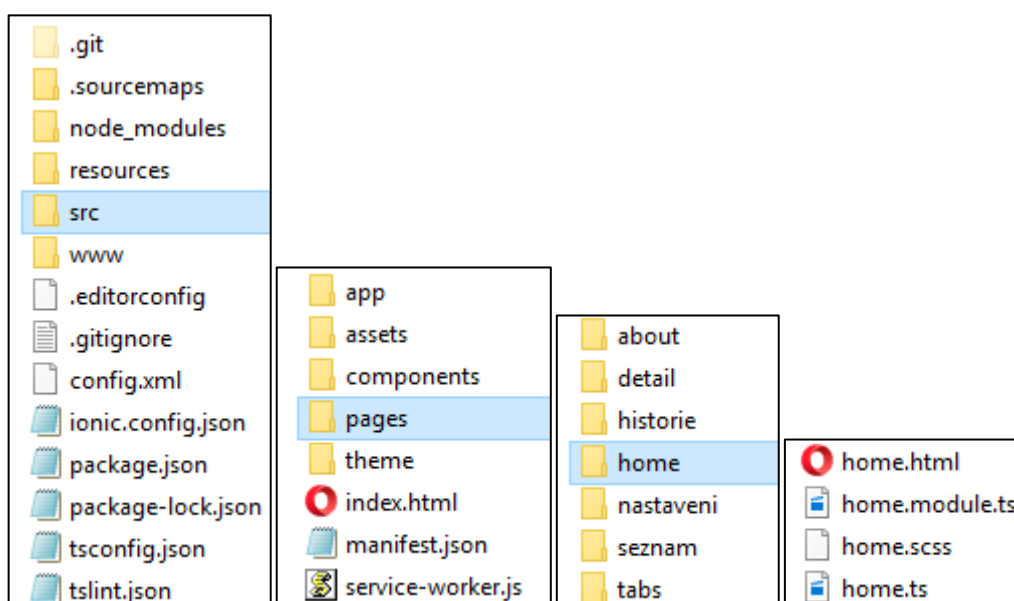
¹¹ Rozšiřující, samostatně nefungující modul

3.1.2 Zhodnocení výběru přístupu

Pro tvorbu aplikace jsem se rozhodl využít Ionic Framework (verze 3.20.0) [31], což je kompletně open-source SDK pro vývoj hybridních aplikací. Programování probíhá v JavaScript frameworku Angular a poté pomocí HTML5 a CSS. Obsahuje velké množství nativní prvků zařízení. Umožňuje provádět testování aplikace v prohlížeči, virtuálním zařízení nebo reálném zařízení. Má poměrně širokou vývojářskou komunitu, rozsáhlou a přehlednou dokumentaci.

3.2 Struktura projektu v prostředí Ionic

Struktura každé aplikace (vytvářené jako projekt a ukládané formou složky se soubory) je na první pohled velmi komplexní a složitá. Ovšem většina z vytvořených souborů jsou systémové soubory prostředí Ionic, které se vytvoří automaticky při startu nového projektu, případně jsou rozšiřovány přidanými pluginy. Je-li řeč o pluginech, ty se ukládají do složky „node_modules“. Z vývojářského hlediska je místo jejich uložení ovšem relativně nepodstatné a jak se k nim přistupuje, názorně zobrazuje kap. 3.4. Nejvíce důležitá je ovšem složka „src“, která v sobě ukrývá veškerý zdrojový kód tvořené aplikace. Zde okomentuji nejdůležitější z podadresářů zmiňované složky.



Obrázek 21: Průřez strukturou projektu

Soubor „index.html“ obsahuje základní informace o aplikaci nutné k vytvoření, kompilaci a spuštění aplikace, například znaková sada, název aplikace, odkaz na soubory potřebné k tvorbě a odkaz na složku „app“, jakožto hlavní adresář projektu.

Složka „app“ je vlastně kostra celé aplikace. Zde jsou například styly, které se poté využívají v celé aplikaci. Zde probíhá inicializace jednotlivých stránek i nastavení, která ze stránek

se použít jako první. Případně sem lze zahrnout metody, které mají běžet nad celou aplikací. Probíhá zde deklarace všech modulů použitých v aplikaci.

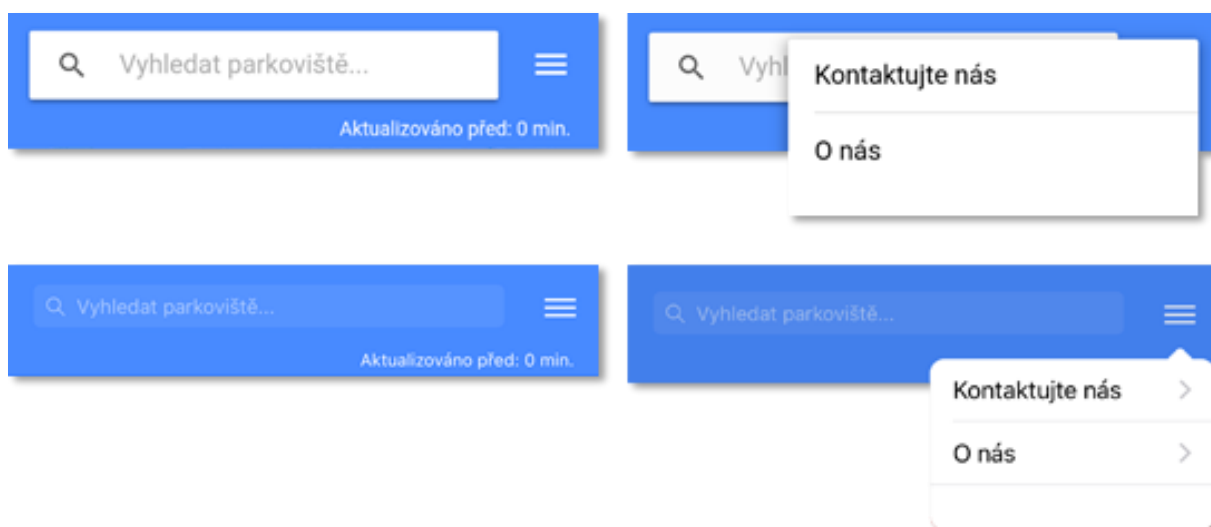
Ve složce „assets“ se nacházejí ikonky, obrázky a mohou zde být například i soubory s daty atd.

Jelikož se aplikace, po vzoru webu, skládá z jednotlivých stránek, je jim zde věnována složka „pages“. Uvnitř lze nalézt všechny soubory vztažené k dané stránce. Při zobrazení adresáře stránky „home“, se objeví přehled o tom, že každá stránka je tvořena čtyřmi soubory. Soubor s koncovkou „.html“ je klasický a notoricky známý soubor pro tvorbu webového obsahu. Koncovka „.scss“ reprezentuje styly, které zajišťují požadovaný vzhled. Do souborů s koncovkou „.ts“ se píše logika a metody dané stránky a soubor „module.ts“ slouží k například k importování a deklaraci pluginů.

Dalšími složkami, které se vyskytnou v této práci, jsou: „components“ a „providers“ – obě složky jsou tvořeny na principu, že jejich obsah lze využít na více stránkách. Do první složky umísťujeme celé komponenty obsahující soubory s koncovkou .html, .ts i .scss. Může se jednat například o kontextové menu využívané na více stránkách. Do složky „providers“ vkládáme pouze soubory s logikou (.ts). To může být například soubor zajišťující komunikaci se serverem. Je vhodné si uvědomit, že ve složce projektu lze vytvořit jakékoliv složky. Výše zmíněný příklad je ale všeobecně uznávaným postupem, který byl použit i v této práci.

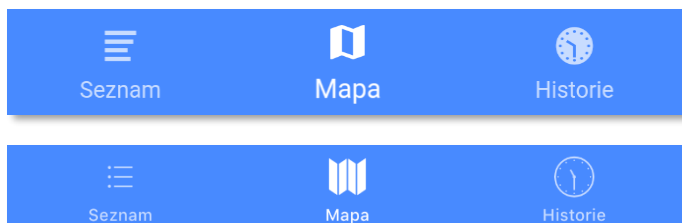
3.3 Návrh GUI v prostředí Ionic Framework

V tomto kroku návrhu je vytvořeno GUI přímo v pracovním prostředí Ionicu. Využívá se tedy především jazyků HTML a SCSS. Koncepte třech hlavních stran a jedné rozšiřující zůstala od počátečního návrhu stejná.



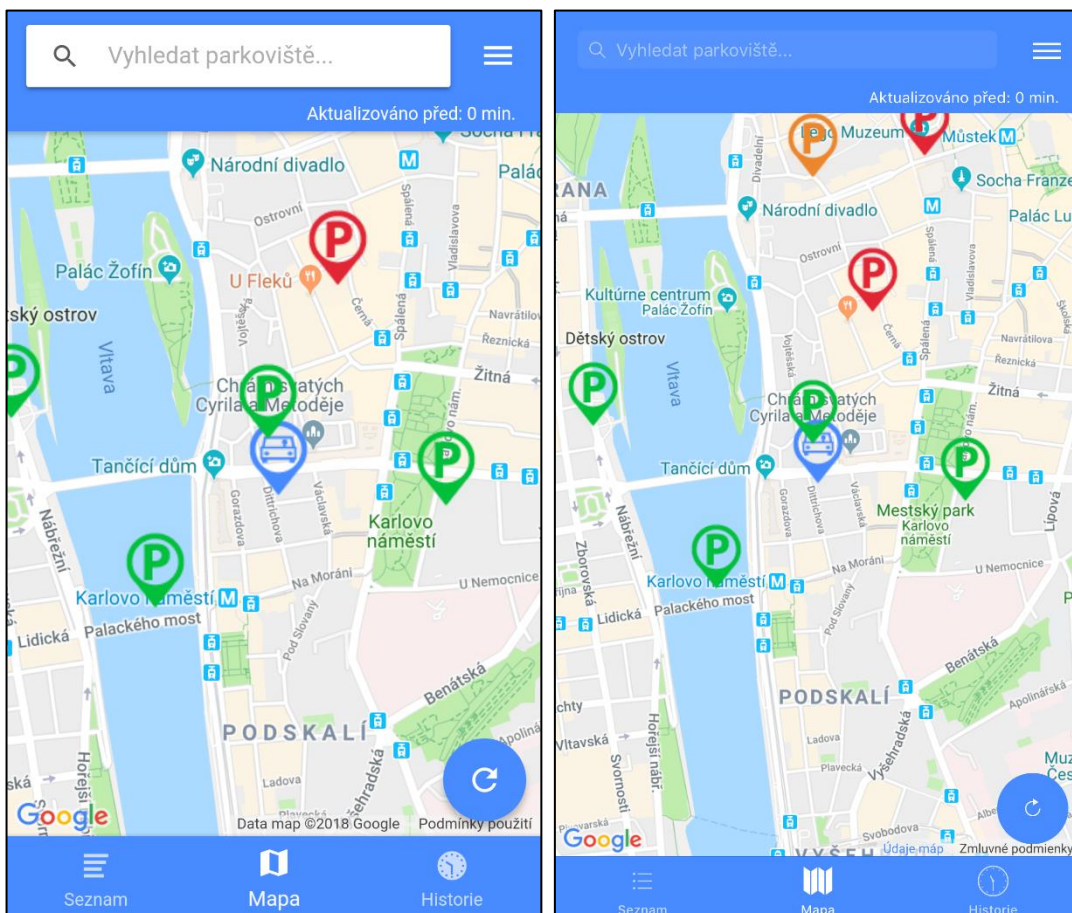
Obrázek 22: Záhloví aplikace bez a se zobrazeným menu (nahore Android, dole iOS)

V tomto kole ovšem byly přidány i obrazovky „Kontaktujte nás“ a „O nás“. Tyto obrazovky byly umístěny do kontextového menu aplikace, které se nachází v pravé horní části záhlaví, kde původně sídlila ikonka pro aktualizaci dat, a je ukryto pod symbolem vyjadřujícím menu. V záhlaví se také nachází vyhledávač a informace o aktuálnosti dat ve změněném formátu: „Aktualizováno před: X min.“. Záhlaví má na všech stránkách stejný základ. Pouze na některých stránkách nejsou některé prvky umístěny.



Obrázek 23: Navigační menu aplikace (nahore Android, dole iOS)

Naproti tomu zápatí je tvořeno navigačním menu aplikace, které se vyskytuje na všech obrazovkách krom stránek „Kontaktujte nás“ a „O nás“ a ve všech případech vypadá identicky. Vybraná položka je zobrazena lehkým zesvětlením a zvětšením.



Obrázek 24: Úvodní stránka aplikace (vlevo Android, vpravo iOS)

Úvodní obrazovka od minulé verze na první pohled nezaznamenala příliš mnoho změn. Avšak, kromě již zmiňovaného záhlaví, jsou jimi tlačítka na manuální aktualizaci a nová, již personalizovaná sada mapových značek a s těmi spojené informační okno. Tlačítka byla vytvořena jako tzv. plovoucí tlačítka a jeho umístění bylo zvoleno do pravého dolního rohu tak, aby bylo co možná nejintuitivněji přístupné palci.

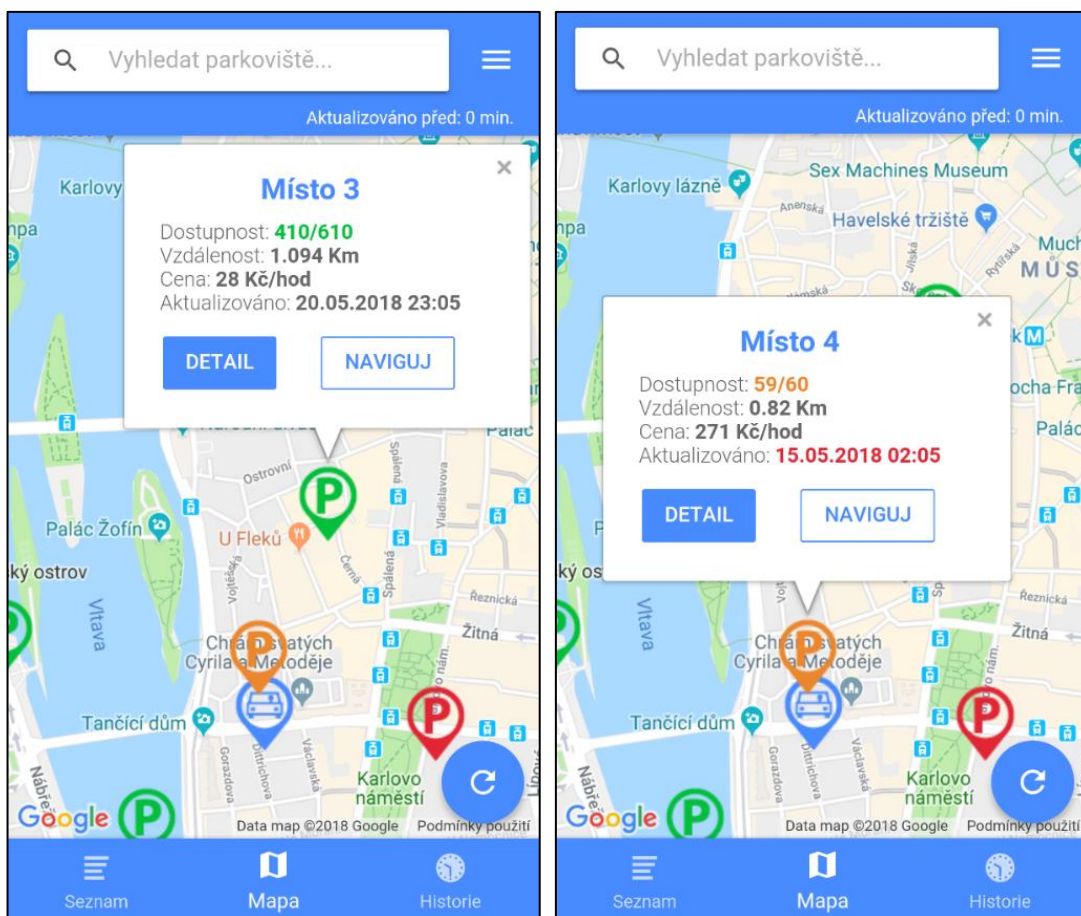
Značky příslušných parkovišť byly barevně rozlišeny dle klíče viz Tabulka 13, plus obsahují ještě značku pro aktuální polohu uživatele a šedivou ikonku značky v případě, že data u daného parkoviště budou starší než 15 minut. Ikonky pro jednotlivé značky jsem vytvořil v grafickém editoru pro vektorovou grafiku a jsou uloženy ve složce „src\assets\icon“ ve formátu PNG.



Obrázek 25: Přehled vytvořených mapových značek

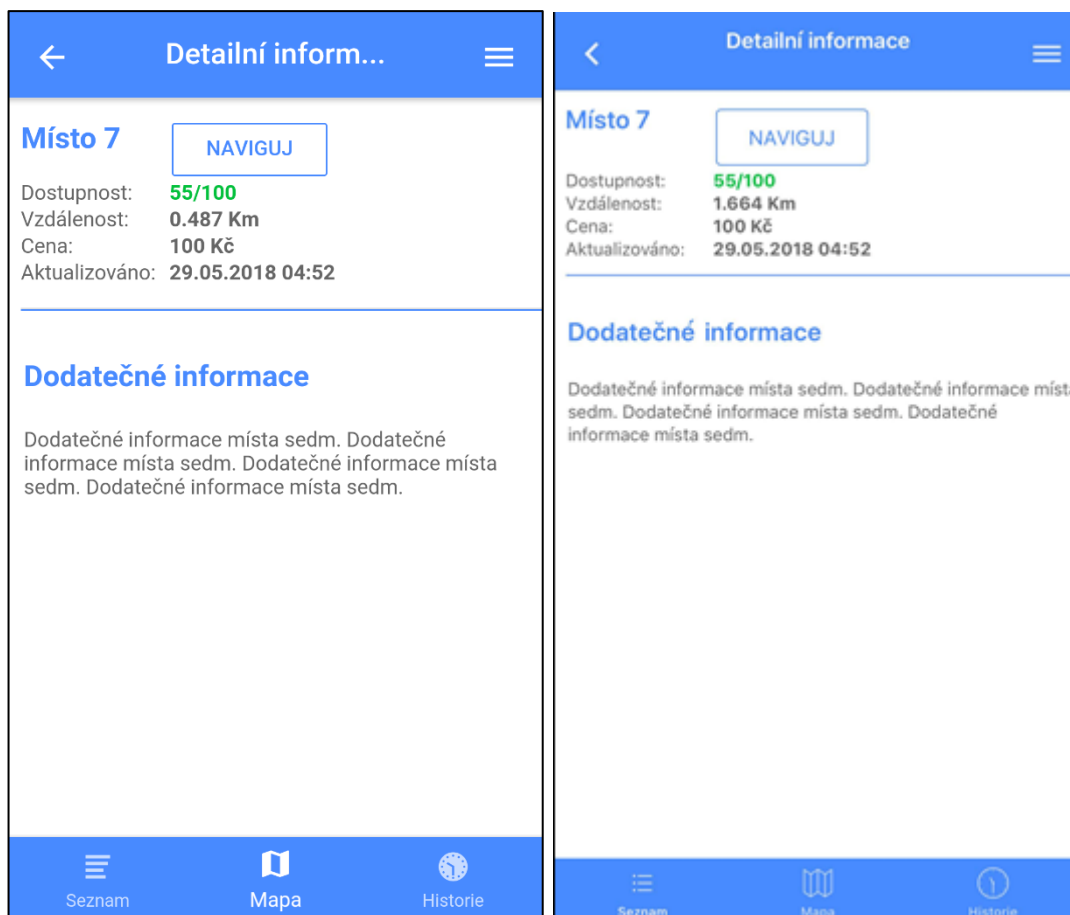
Po zvolení některé značky se uživateli zobrazí informační okno shrnující základní informace o parkovišti včetně možnosti zobrazení detailu a navigace na dané parkoviště. Informace zobrazené v tomto okně jsou:

- název parkoviště.
- dostupnost – vyjádřena jako poměr počtu obsazených míst k celkovému počtu míst a v minulé verzi byl tento údaj označen jako obsazenost.
- vzdálenost – v kilometrech vyjádřená vzdálenost parkoviště od současné polohy uživatele.
- cena – cena za hodinu parkování vozidla.
- aktualizováno – datum a čas poslední změny dat na daném parkovišti. Pokud jsou data starší 15 minut, je tato položka zbarvena červeně.



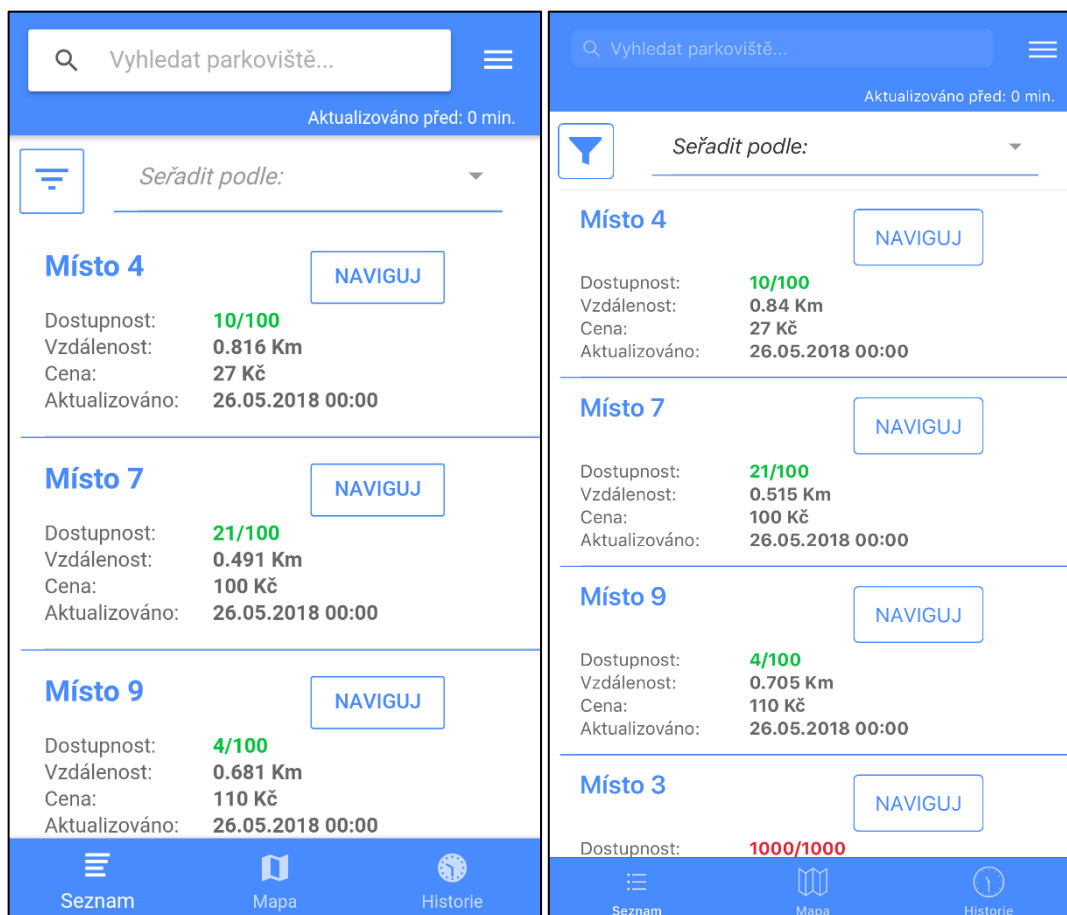
Obrázek 26: Zobrazení informačního okna v případě aktuálních i starých dat

Po kliknutí na tlačítko „Detail“ je uživatel přesměrován na příslušnou stránku s rozšiřujícími údaji. Tato stránka utrpěla pár kosmetických změn. Tou první je přidání kontextového menu do záhlaví stránky. Tou druhou je vynechání ilustračního obrázku parkoviště. Po analýze kladů a záporů toho, jestli ho zobrazovat či ne, jsem došel k závěru, že efektivnější je nezobrazovat, a to ze dvou důvodů. Zaprvé jeho výpovědní hodnota je poměrně malá, neboť uživatel by viděl pouze objekt či vjezd do něj, a za druhé při načítání ze serveru se ušetří nezanedbatelné množství dat, pokud obrázek zde nebude a tím spíše aplikace naplní svůj primární cíl, a to, aby pracovala svižně.



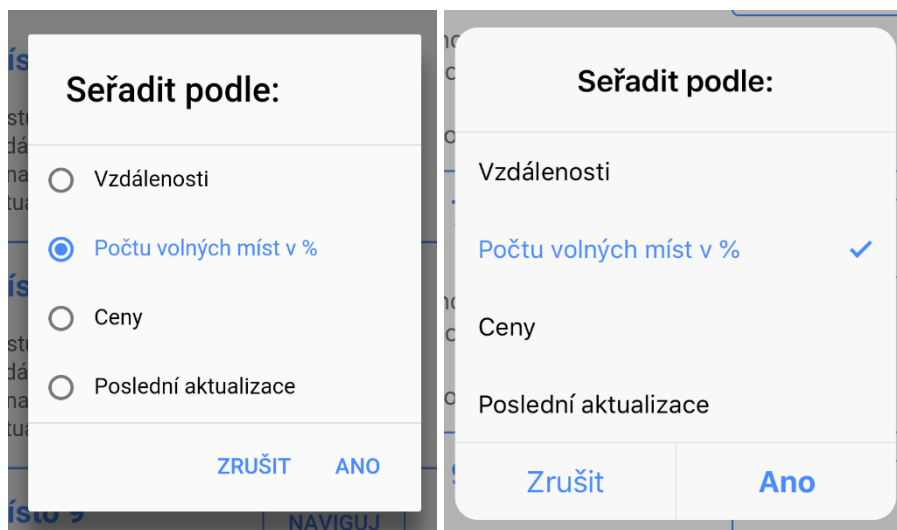
Obrázek 27: Obrazovka detailu parkoviště (vlevo Android, vpravo iOS)

Další změna se týká tlačítka pro navigaci. To bylo sjednoceno do stylu, jakým je zobrazováno i na všech ostatních stránkách. Tlačítko s nápisem „Naviguj“ místo pouhé ikonky bylo zvoleno z důvodu, že uživateli je hned jasné, co znamená. Pro někoho by mohla být pouhá ikonka zmatečná a nevěděl by si pod ní rychle vybavit, jaká je její funkce. Další informace zůstaly nezměněné, pouze pole „Aktualizováno“ se opět v případě stáří dat starších než 15 minut zobrazí červeně. Důvodem je přehledná a nepřehlédnutelná informace o potencionálních neaktuálních hodnotách.



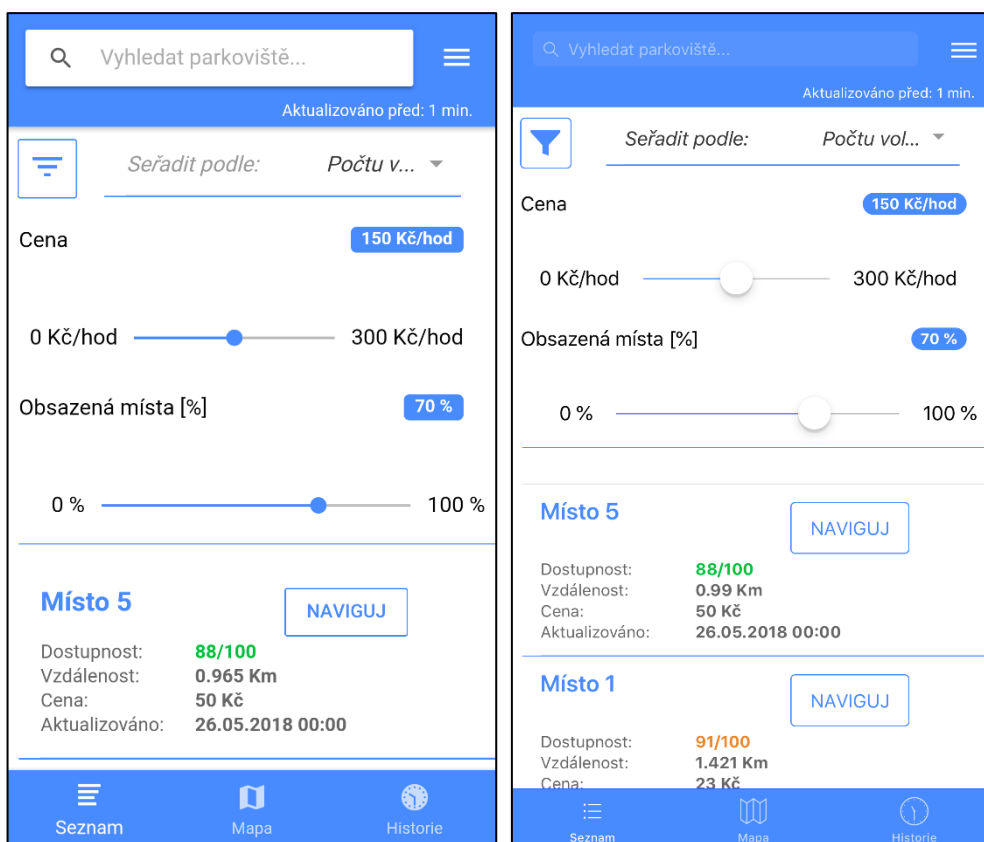
Obrázek 28: Základní vzhled obrazovky (vlevo Android, vpravo iOS)

Při zobrazení obrazovky seznamu lze pozorovat nejvíce změn. Záhloví je stejné jako v případě mapy, tj. přibýlo menu a změnil se formát času. V obsahu byla přidána nalevo od řazení ikonka pro filtraci dat. Po jejím stisknutí se zobrazí dva posuvníky, jeden pro cenu a druhý pro dostupnost, kde uživatel změnou hodnoty bude moci filtrovat zobrazovaná data a docílí se tak tedy multikriteriálního výběru. Modré ikonky s číslem značí aktuální hodnotu posuvníku. V případě, že se uživatel rozhodne data seřadit podle některého z parametrů, objeví se mu okno s možností volby tohoto parametru. Nově byla přidána možnost seřadit podle aktuálnosti dat.



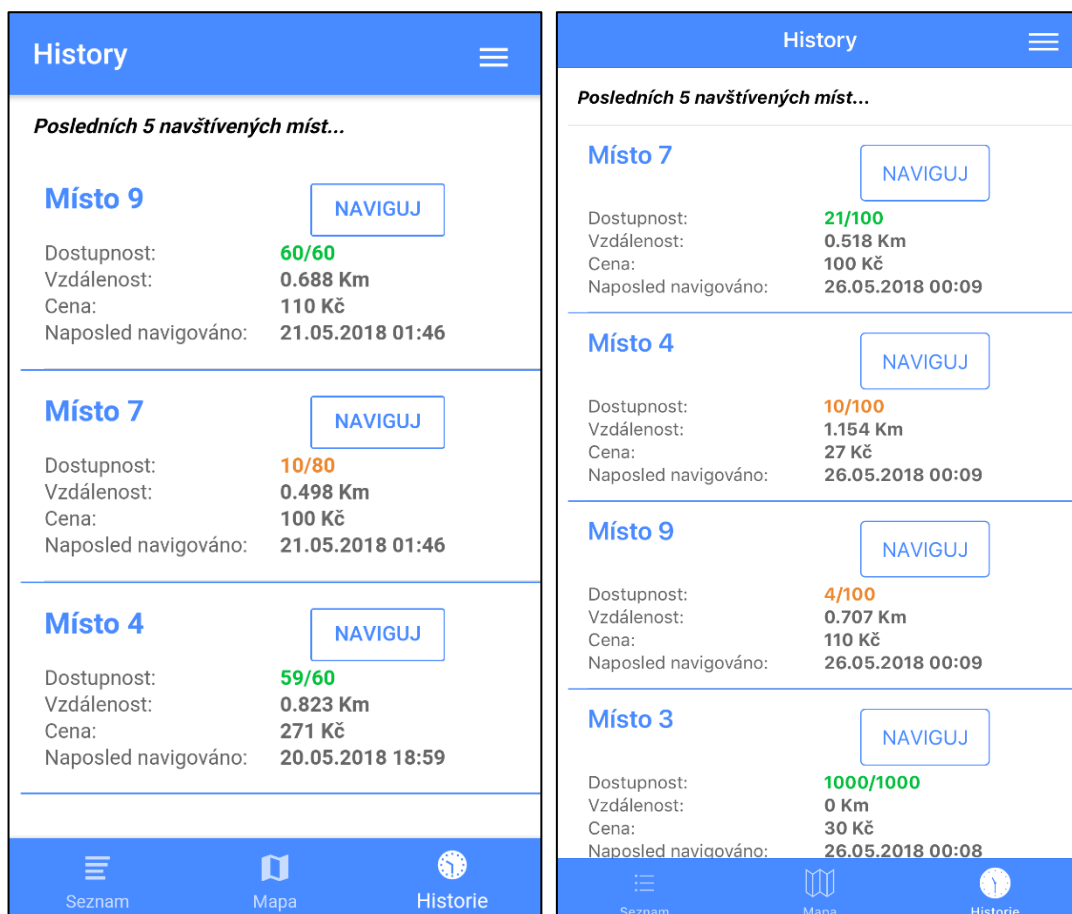
Obrázek 29: Ukázka okna pro seřazení podle (vlevo Android, vpravo iOS)

Samotný seznam se také oproti minulému kroku lehce změnil. Informace již nejsou zobrazovány na řádku vedle sebe, ale na řádcích pod sebou a opět byla přidána informace o aktuálnosti dat, s barevně zvýrazněnými starými daty, společně s vyměněným tlačítkem pro navigaci. Zobrazení informací tímto způsobem bylo vybráno z důvodu přehlednosti, jelikož všechna data se na jeden řádek nemohla vejít. I v seznamu je možnost manuální aktualizace dat. Využívá se zde pro mobilní platformu typickým posunutím obrazovky směrem dolů.



Obrázek 30: Ukázka seznamu s rozbaleným filtrem (vlevo Android, vpravo iOS)

Historie doznala podobných změn jako seznam, jediným rozdílem je absence řazení a filtrace, prvků, které nemají pro pět posledních navštívených míst valný význam. Pole aktualizace dat je nahrazeno polem poslední návštěvy. Pokud se uživatel chce dozvědět, zdali jsou data aktuální, zobrazí si detail parkoviště. Důvodem je, aby seznam historie neobsahoval až příliš velké množství informací.



Obrázek 31: Ukázka obrazovky historie (vpravo Android, vlevo iOS)

Obrazovka “Kontaktujte nás” slouží uživateli, pokud by měl nějaké připomínky k aplikaci. Skládá se ze dvou částí, záhlaví a obsahu. V záhlaví je umístěn pouze název a tlačítko pro navrácení se na předchozí stránku. V obsahu je uvedena verze aplikace a možnost kontaktu. Po zvolení položky kontaktování je uživatel přesměrován do emailu s předvyplněnými údaji o adrese cíle, předmětu a první větou emailu.



Obrázek 32: Obrazovky „o nás“ a „kontaktuje nás“

Obrazovka „O nás“ je věnována základní informacím o aplikaci. Uvádí, že vznikala jako součást diplomové práce, byla tvořena na fakultě dopravní ČVUT v Praze a jméno autora.

3.4 Použité pluginy a jejich použití

V této podkapitole je řešeno, jakým způsobem se pluginy importují do projektu a které pluginy byly použity včetně jejich krátkého popisu. Samotná práce s těmito pluginy je řešena v popisu důležitých metod této práce.

```
import { Geolocation } from '@ionic-native/geolocation';
:
:
:
providers: [
  StatusBar,
  SplashScreen,
  DetailPage,
  {provide: ErrorHandler, useClass: IonicErrorHandler}
  GoogleMapsProvider,
  LocationHelperProvider,
  Geolocation,
  LaunchPage,
```

Obrázek 33: Ukázka import pluginu Geolocation v souboru app.module.ts

Pro využití nativních prvků telefonu se v prostředí Ionic využívají pluginy. Slovem plugin se zde myslí knihovny zdrojového kódu třetích stran. Využívají se zejména pro přístup k nativním prvkům telefonu, k usnadnění a zrychlení práce. Jejich implementace probíhá tak,

že v příkazovém řádku uvnitř adresáře s aplikací je potřeba nejprve pomocí specifického příkazu nainstalovat požadovaný plugin, tím se vloží do složky „node_modules“ a je možné k němu přistupovat. To probíhá tak, že se v souboru „app.module.ts“ nejdříve požadovaný plugin inicializuje. Tím je zajištěna možnost jeho použití v projektu. Zabezpečení toho, aby mohl být použit na stránce ještě vyžaduje import pluginu do požadovaného souboru, který je identický (zeleně označená část Obr. 33), jako v „app.module.ts“.

3.4.1 Seznam důležitých pluginů použitých v navrhované aplikaci

Zde je uvedeno několik základních pluginů využívaných v aplikaci včetně jejich krátkého popisu. Na jejich využití bude upozorněno v kap. 3.5.

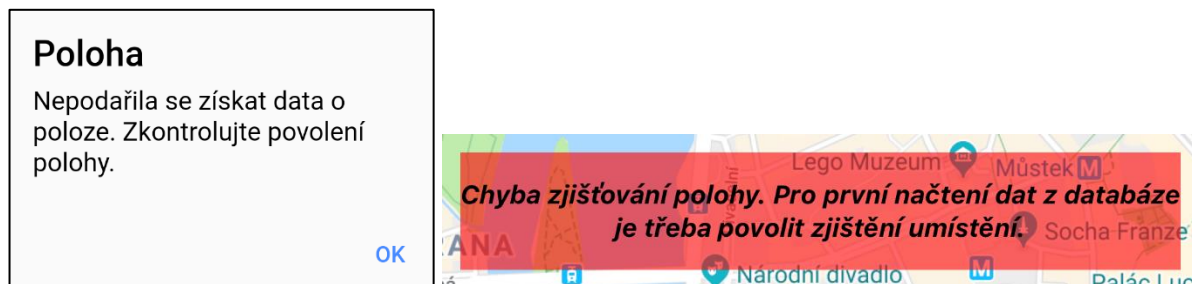
- **Angular Google Maps** – sada směrnic napsaných v jazycích CoffeeScript a Javascript, které integrují mapy Google do aplikací Angular. Tento plugin umožňuje používat Google maps API JDK pomocí jazyka Angular. Založeno na Google Maps Javascript API verzi 3. [19]
- **Diagnostic** – Slouží ke kontrole, zdali jsou hardwarové funkce zařízení pro aplikaci povoleny, případně jsou-li k dispozici. Umožňuje spravovat oprávnění přístupů za běhu aplikace. Zabývá se například kontrolou kamery, GPS, Wi-Fi atd. [20]
- **Email Composer** – zaručuje přístup ke standardnímu rozhraní, které spravuje úpravu a odesílání emailových zpráv. Též umožňuje přednastavit některé vstupy v této zprávě. [21]
- **Geolocation** – plugin pracující se získáváním informací o poloze zařízení. Dokáže využívat informace z GPS, ale i informace odvozené ze síťových signálů jako Wi-Fi, GSM atd. Neřeší dostupnost těchto zařízení. [22]
- **Launch Navigator** – zajišťuje otevření navigační aplikace, která je předem vyplněna údaji o startu a cíli. [23]
- **Network** – další diagnostický nástroj. Oproti nástroji „Diagnostic“ se soustřeďuje pouze na přístup k Internetu, ale má výhodu, že bere v potaz jak připojení pomocí Wi-Fi, tak připojení pomocí GSM. [24]

3.5 Logika aplikace

Jelikož se v aplikaci vyskytuje relativně mnoho funkcí a metod, budou probrány v této podkapitole ty nejzajímavější a pro fungování aplikace nejzásadnější z nich. Jsou zde popisovány spíše z hlediska funkčnosti, ovšem veškerý zdrojový kód je k nahlédnutí v příloze.

3.5.1 Získání polohy uživatele

Při žádosti aplikace o získání polohy je nejdříve zkontrolováno, zdali má uživatel povolené získávání polohy. K ověření se používá plugin Diagnostic, který zjistí, jestli jsou koordináty dostupné, a vrátí hodnotu ano/ne.



Obrázek 34: Ukázka případu při vypnuté GPS, alert a chybová hláška

Pokud poloha není povolena, je vytvořen a zobrazen tzv. alert (vyskakovací okno) viz Obr. 34. Uživatel se může nadále v aplikaci pohybovat, ovšem je zobrazena hláška o neaktuálnosti dat, která zmizí až v případě, že uživatel zapne polohu a dojde k aktualizaci.



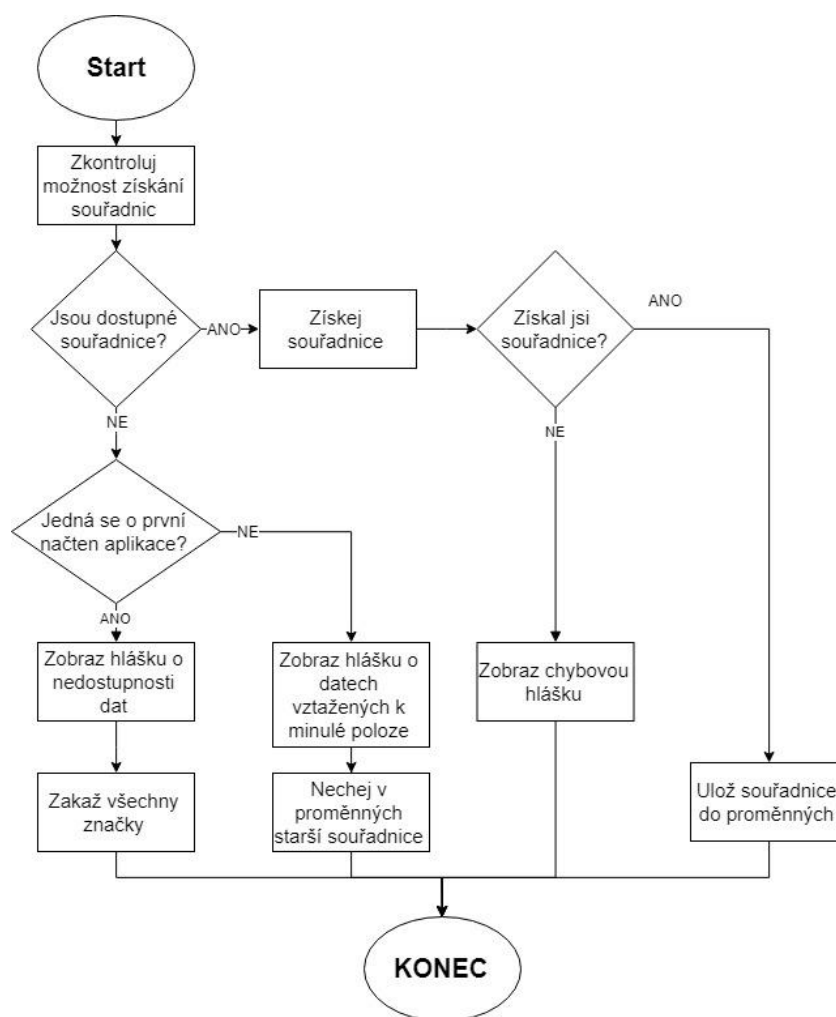
Obrázek 35: Druhý typ chybové hlášky

Tato hláška se liší, pokud poloha není zapnuta při startu aplikace a nebylo možné tedy zatím načíst data ze serveru (k prvotnímu načtení je zapotřebí zeměpisných souřadnic uživatelské polohy) nebo zdali uživatel již jednou data načel. Hláška se zobrazuje kvůli neaktuální hodnotě vzdálenosti cíle od polohy uživatele. Upozornění o zastaralosti této polohy je zobrazeno i v seznamu a historii. Tam je tvořeno textem „!NEAKTUÁLNÍ!“ v poli pro vzdálenost.



Obrázek 36: Ukázka staré polohy na obrazovce seznamu

Pokud je poloha zapnuta, dojde ke zjištění zeměpisných souřadnic za pomoci pluginu Geolocation. Tyto údaje jsou poté uloženy do proměnných, se kterými se dále pracuje, a tato současná poloha je následně v mapě označena modrou značkou s obrázkem vozidla a použita pro vycentrování mapy na tuto polohu. Tato poloha je předávána dalším navazujícím metodám. Vzhledem k důležitosti této funkce jsem níže přiložil stručný vývojový diagram.



Obrázek 37: Algoritmus určení polohy

3.5.2 Práce se serverem

Popis serveru – jelikož je aplikace zamýšlena pro práci v reálném čase, jsou všechna data shromažďována na serveru, odkud jsou aplikací stahována. Veškeré změny tedy stačí provést na serveru. Byl vybrán server Back4App.com, z důvodu nepříliš náročné integrace, rychlosti stahování dat ze serveru a také proto, že bezplatná verze plně uspokojuje požadavky na vývoj a testování funkčnosti této aplikace. V aplikaci budou data ohledně parkovišť simulovaná. V reálném provozu by bylo třeba data od různých společností průběžně stahovat a překládat do formátu datové tabulky. Každý parkovací dům zapojený do toho projektu, by obdržel unikátní id, díky kterému by bylo možné data na server zapisovat a upravovat pod tímto id. Ovšem komunikace aplikace se serverem je stejná u simulovaných i reálných dat. Data, s nimiž aplikace pracuje a server je má tedy uložené, vycházejí z kap. 1.1.5 a vzhledem k potřebám aplikace jsou dále o několik záznamů rozšířena. Všechna data jsou tedy uložena na serveru, lokálně jsou uloženy pouze záznamy o historii.

Zde je uvedena ukázka databáze ze serveru:

- `objectId` – unikátní id záznamu (parkovacího domu). Typ textový řetězec.
- `price` – cena za jednu hodinu parkování. Typ číslo.
- `locName` – název parkovacího místa. Typ textový řetězec.
- `updatedAt` – datum a čas poslední změny tohoto záznamu. Typ datum.
- `location` – zeměpisné souřadnice objektu. Typ geopoint (zeměpisná šířka, zeměpisná délka).
- `occupancy` – počet obsazených míst. Typ číslo.
- `description` – dodatečný popis dané položky. Typ textový řetězec.
- `totalSlots` – celkový počet míst. Typ číslo.
- `address` – název města, případně části města či ulice, kde se parkoviště nachází. Typ textový řetězec.

<code>objectId</code> String	<code>locationImage</code> File	<code>price</code> Number	<code>ACL</code> ACL	<code>locName</code> String
<input type="checkbox"/> gFISpTqE13		21	Public Read + Write	Místo 1
<code>updatedAt</code> Date	<code>location</code> GeoPoint	<code>updated</code> Date	<code>createdAt</code> Date	
21 May 2018 at 21:33:22 UTC	(50.08205, 14.4157...	30 Apr 2018 at 20:28:04 UTC	30 Apr 2018 at 20:...	
<code>occupancy</code> Number	<code>description</code> String	<code>totalSlots</code> Number	<code>address</code> String	
410	Místo 3, šli tři v...	610	Praha, Jih	

Obrázek 38: Ukázka struktury tabulky a jednoho zápisu v prostředí serveru

Načtení dat ze serveru – se řeší v souboru „data-service.ts“. Ovšem server je nejdříve potřeba inicializovat, což je vyřešeno pomocí funkce při spuštění aplikace v „app.component.ts“. Zde se přistupuje pomocí serverové API adresy, id aplikace a klíče pro přístup a komunikaci k serveru sloužící jako backend pro aplikaci, na kterém se data nacházejí v databázi. Základem je, že aplikace podává dotazy na server a ten jí odpovídá. Jeho odpověď je uložena do lokálního pole „Items“. S těmi se nadále v aplikaci pracuje. Při každém požadavku na aktualizaci dat je toto pole přepsáno novými daty.

```

//Najdi všechny objekty
return query.find()
  .then(function (results) {
    //Získej data z výsledků
    for (var i = 0; i < results.length; i++) {
      let temp: any = {};
      var object = results[i];
      temp.id = object.id;
      temp.locName = object.get("locName");
      temp.occupancy = object.get("occupancy");
      temp.totalSlots = object.get("totalSlots");
      temp.price = object.get("price");
      temp.description = object.get("description");
      temp.updated = object.updatedAt;
      temp.locationlatlng = object.get("location");
      temp.latitude = temp.locationlatlng.latitude;
      temp.longitude = temp.locationlatlng.longitude;
      temp.locationImage = object.get("locationImage").url();
      temp.distanceBwPoints = ref.distanceInKmBetweenEarthCoordinates
        (ref.mylat, ref.mylng, temp.locationlatlng.latitude, temp.locationlatlng.longitude, i);
      ref.Items.push(temp);
    }
    ref.timeUpdated = new Date();
  })
return ref.Items;

```

Obrázek 39: Ukázka dotazu na server a uložení dat do pole

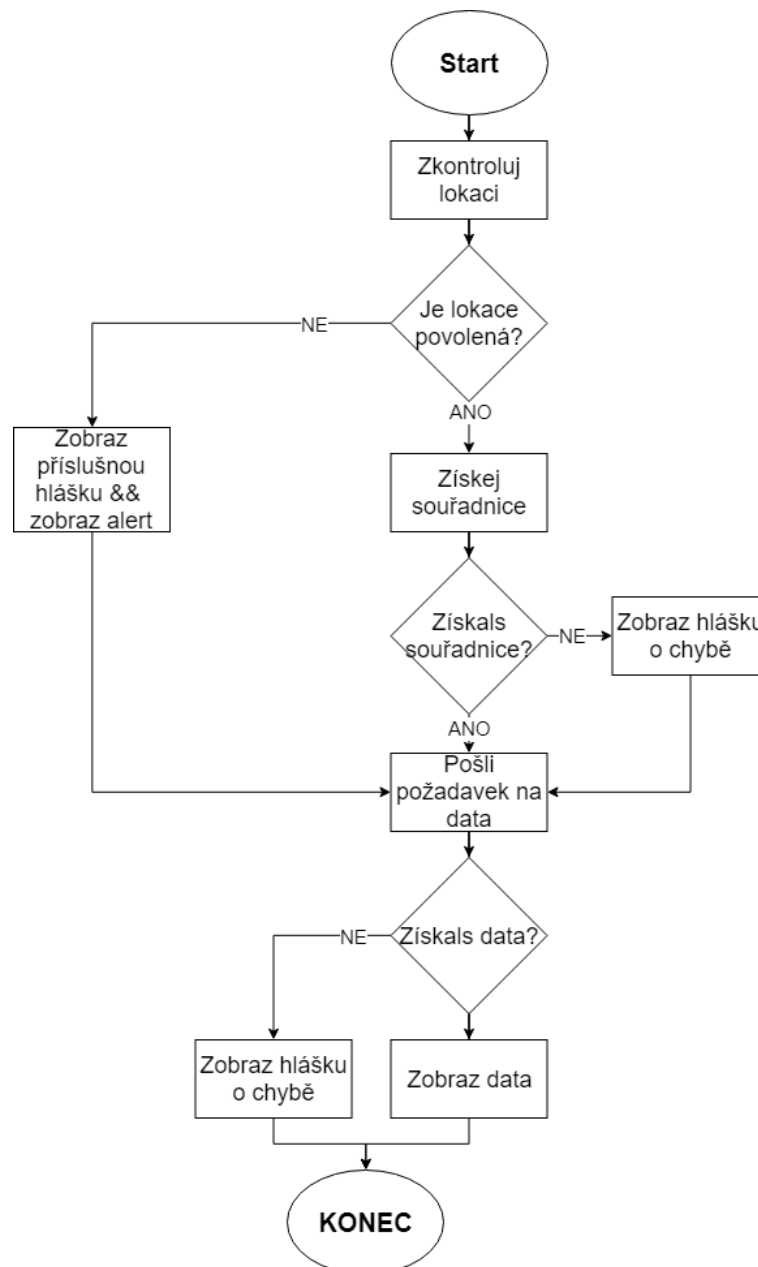
3.5.3 Kontrola připojení k Internetu

Tato funkce je volána v „app.component.ts“ a využívá pluginu Network. Běží tedy nad celým programem. V případě, že aplikace nemá přístup k Internetu, a to pomocí Wi-Fi ani GSM, je zobrazena chybová hláška informující o tomto faktu a aplikaci nelze používat, dokud není spojení obnoveno. Pro správný chod aplikace je připojení k Internetu naprosto zásadním požadavkem.

3.5.4 Aktualizace dat

Aktualizace dat je v aplikaci dvojího typu:

- manuální – je inicializována uživatelem. Probíhá stiskem plovoucího tlačítka na obrazovce s mapou nebo přetažením obrazovky směrem dolů u seznamu;
- periodická – probíhá každých 5 minut; v případě, že uživatel provede manuální aktualizaci, začne odpočet opět od nuly.



Obrázek 40: Algoritmus aktualizace dat

3.5.5 Vyhledávání

Vyhledávání probíhá na dvou obrazovkách: mapě a seznamu. Liší se ve způsobu vyhledávání. V případě vyhledávání v seznamu je tento seznam okamžitě filtrován a zobrazuje pouze odpovídající záznamy. V případě vyhledávání v mapě je při psaní uživateli zobrazován našeptávač, uživatel má možnost zvolit hledanou položku z našeptávače nebo ji dopsat, případně napsat dostatečně dlouhý text k jasné identifikaci. Při zvolení jedné z těchto možností se zobrazí mapa vycentrovaná na hledanou položku a ostatní značky, kromě značky aktuální polohy, jsou z mapy dočasně odstraněny tak, aby uživatel v případě velkého výskytu značek neměl pozornost odváděnou od cíle hledání. Při vymazání obsahu vyhledávače nebo aktualizaci dat je mapa vrácena do původního stavu.

Další možností vyhledávání je vyhledávat podle adresy, jelikož uživateli se zobrazují, z důvodu úspory dat a zrychlení aplikace, pouze záznamy v okolí 50 kilometrů. V případě, že ovšem plánuje cestu a jeho cíl je vzdálen více než 50 km, zadá do vyhledávače adresu, stačí například město. Na to je následně vycentrována mapa a jsou zobrazeny záznamy z okolí vyhledávaného místa. V případě budoucího rozvoje a rozšíření aplikace o velké množství parkovišť má tato funkce potenciál enormně ulehčit náročnosti zpracování dat, aplikaci zrychlit a nezahlcovat jí při načítání velkým množstvím dat.

3.5.6 Úprava barev na základě aktuálních informací

Jedná se o změny barvy prvků vzhledem k údajům o dostupnosti a aktuálnosti dat. Je-li řeč o značkách, je tato funkce dosažena jednoduchou funkcí „IF“ uvnitř HTML souboru. Výběr barvy probíhá podle několikrát zmíněného klíče dle Tab. 13 kap. 2.7, navíc je přidána šedivá značka v případě, že data daného parkoviště jsou starší než 15 minut. Na základě posouzení se poté na místě značky objeví příslušně zbarvená značka. Zabarvení textu probíhá podle stejných principů pouze v souboru typu TypeScript.

```
<agm-marker *ngFor="let m of Items;let i = index"  
[latitude]="m.latitude" [longitude]="m.longitude"  
[iconUrl]="((m.availability/m.totalSlots)==1)? markerCervena:  
((m.availability/m.totalSlots)>0.9)? markerZluta : markerZelena"  
(markerClick)="showInfoWindow(infoWindow, i)"  
>
```

Obrázek 41: Ukázka podmínky pro výběr barvy značky

Začervenání starých dat je dosaženo porovnáním současného času s časem, kdy byla provedena poslední aktualizace daného parkoviště. Tento čas je převeden na minuty a v případě, že je větší než 15 minut, vrací funkce červenou barvu, která je následně použita na zbarvení textu.

3.5.7 Filtrace a řazení dat

Jedná se o funkci objevující se na obrazovce seznamu. Princip **řazení dat** funguje tak, že v případě zaškrtnutí některého parametru ho nejdříve určí a potom, na tomto základě, data seřadí dle přednastaveného kritéria na Obr. 29 str. 50 dle:

- vzdálenosti – od nejbližšího po nejvzdálenější parkoviště;
- počtu volných míst v % – od plně obsazených po volná parkoviště;
- ceny – od nejlevnějšího po parkoviště s nejdražším parkováním;
- aktuálnosti – od nejaktuálnějších dat po nejstarší.

Nabídka **filtrace dat** se zobrazí po stisknutí tlačítka s ikonkou filtru (viz Obr. 30 str. 50). Pomocí dvou posuvníků může uživatel filtrovat hodnoty seznamu. Zobrazují se pouze data

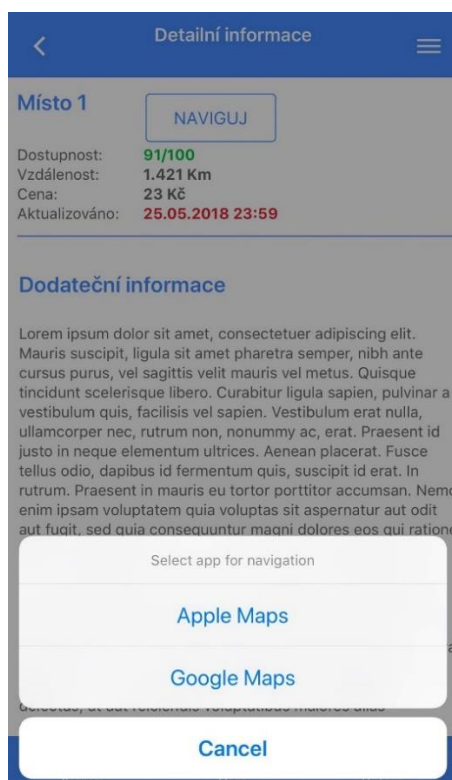
odpovídající oběma posuvníkům současně. Navíc takto vyfiltrovaná data má uživatel možnost následně ještě seřadit dle minulého bodu. V případě přechodu na jinou stránku dojde k obnovení filtru i seřazení dat na výchozí hodnotu. V případě aktualizace dat zůstává filtr i seřazení na uživatelem zadaných hodnotách.

3.5.8 Uložení posledních navštívených míst

Popis této funkce je, že při zvolení možnosti navigovat u některého z parkovišť je tato akce zaznamenána a parkoviště je zobrazeno v historii. V té funguje seznam na principu zásobníku. Každé nově navštívené parkoviště je přidáno navrch zásobníku. V případě, že zásobník přeteče, je od spodu oříznut tak, aby vždy obsahoval pouze pět položek. Každá položka se smí v zásobníku vyskytovat pouze jednou. Pokud je zvolena v historii se již vyskytující položka, která ovšem není na prvním místě zásobníku, dojde k jejímu přesunu na první místo.

3.5.9 Navigace na parkoviště

Navigace probíhá v externí aplikaci. Její spuštění zajišťuje plugin Launch Navigator, kterému jsou předány údaje o zeměpisných souřadnicích vybrané položky (parkoviště). Zároveň je touto akcí parkoviště uloženo do historie posledních navštívených míst.



Obrázek 42: Ukázka výběru požadované aplikace k navigaci

4 Testování a analýza spolehlivosti

Obsahem této kapitoly je testování na reálných uživateliích a analýza spolehlivosti v případě poruchových situací.

4.1 Uživatelské testy

Testy proběhly na sedmi uživateliích. Dva provedli testování na operačním systému iOS a pět na systému Android. Oba dva uživateli, kteří provedli testování na systému iOS, ho absolvovali na stejném zařízení, protože systémy iOS nepovolují volné šíření aplikace bez jeho stažení z Apple Store. Bylo nutné ho tedy spustit na zařízení přes program XCode, jako tzv. vývojář s využitím připojení pomocí USB. Dotazník byl vytvořen pomocí online nástroje Google Forms.

Parkuj Rychle - uživatelské testy

Formulář sloužící k uživatelským testům aplikace Parkuj Rychle, tvořené jako součást diplomové práce. Do odpovědí napište, jak Vám daná úloha přišla srozumitelná, jestli jste ji dokázal/a vypracovat a jestli se vyskytly nějaké potíže či nejasnosti.

* Required

Na jakém operačním systému byla aplikace zkoušena? *

Android

iOS

Úloha č. 1.: Na mapě najdi ve svém okolí volné parkoviště a zjisti aktuální hodnotu dostupnosti takového parkoviště. *

Your answer

Úloha 2.: V seznamu seřaď parkoviště dle ceny a zobraz si detail nejlevnějšího parkoviště. *

Your answer

Úloha 3.: Proveď aktualizaci dat v seznamu i v mapě. *

Obrázek 43: Ukázka formuláře pro uživatelské testování

4.1.1 Zadané úlohy

Zadané úlohy byly zvoleny tak, aby obsahovaly průchod všemi funkčními požadavky.

Úloha 1: Na mapě najdi ve svém okolí volné parkoviště a zjisti aktuální hodnotu dostupnosti takového parkoviště.

Úloha 2: V seznamu seřaď parkoviště dle ceny a zobraz si detail nejlevnějšího parkoviště.

Úloha 3: Proveď aktualizaci dat v seznamu i v mapě.

Úloha 4: Najdi parkoviště v městyse „Sepekov“ a spusť navigaci k němu.

Úloha 5: Kontaktuj vývojáře pomocí e-mailu.

Úloha 6: Vypni přístup k Internetu a umístění k poloze. Při vypnuté poloze a zapnutém přístupu k Internetu si aktualizuj data.

4.1.2 Vyhodnocení odpovědí od participantů

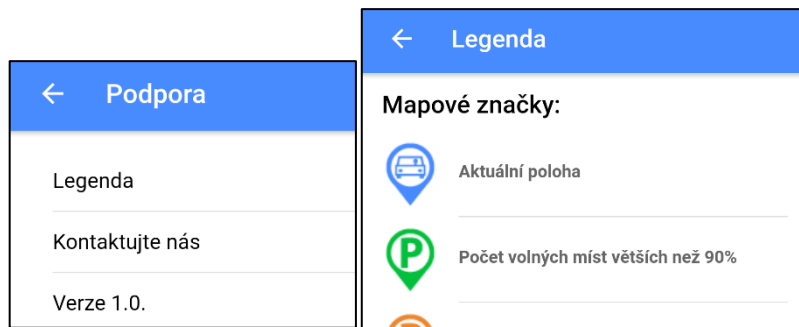
Všichni participanté zvládli všechny úlohy bez problému. Pouze si všimli několika drobných a relativně lehce opravitelných nedostatků a chyb. Verze aplikace byla po testovací fázi a zapracování oprav změněna z 0.9 na současnou verzi 1.0. Vyplněné dotazníky jsou součástí přílohy.

Tyto chyby jsou:

- 1) Byl zmíněn problém s názvem údaje „dostupnost“. Uživatelé si nebyli na první pohled jistí, zdali například hodnota 9/10 vyjadřuje 9 míst volných či obsazených.
- 2) Bylo upozorněno, že v detailu parkoviště jsou, oproti seznamu či mapě, prohozené barvy údaje dostupnosti.
- 3) Někteří uživatelé v obrazovce „seznam“ původně hledali stejné tlačítko pro aktualizaci jako v mapě, než se pokusili využít gesta pro potažení.
- 4) Uživatelé nahlásili, že by uvítali jakousi legendu, která by vysvětlila barvy jednotlivých značek.

Provedená řešení:

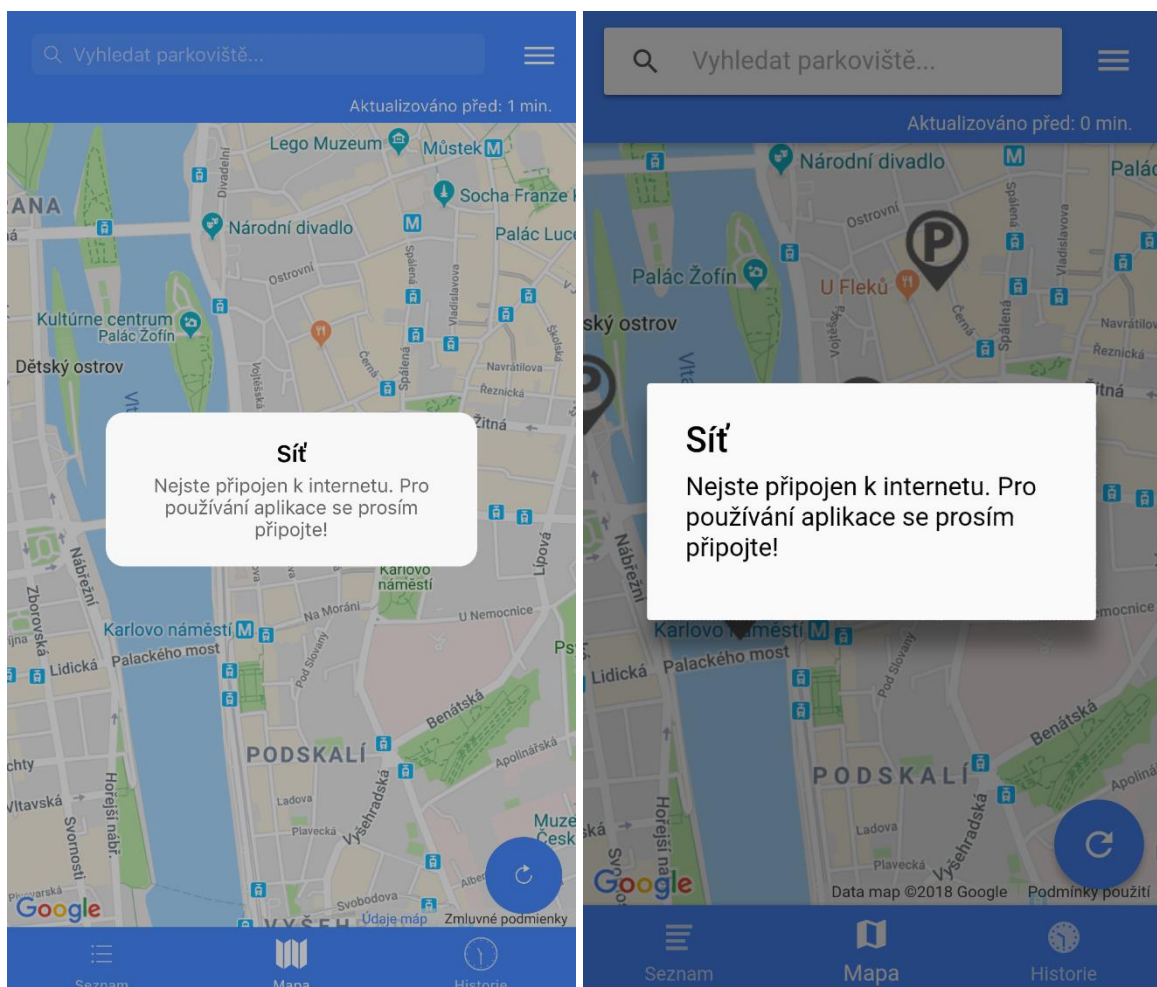
- 1) Údaj „dostupnost“ byl přejmenován na původní údaj „obsazenost“. To již uživatelům přišlo lépe pochopitelné.
- 2) Tato chyba byla způsobena prohozeným znaménkem a byla odstraněna.
- 3) Vzhledem k design guidelines (pokyny tvorby návrhu) jsem tento prvek ponechal ve stávající podobě.
- 4) Položka menu „Kontaktujte nás“ byla přejmenována na „Podpora“ a do ní byl vložen odkaz na stránku „Legenda“.



Obrázek 44: Ukázka obrazovky Podpora a Legenda

4.2 Nedostupnost Internetu

Aplikace po celou dobu své činnosti sleduje, zdali má uživatel internetové připojení. V případě, že neneviduje žádné připojení, ať již pomocí Wi-Fi či sítě GSM, vyskočí okno upozorňující na tento fakt a oznamující, že bez připojení k Internetu není možné aplikaci používat. Tato varianta je zvolena, protože všechna data o parkovištích si aplikace stahuje ze serveru a v případě, že nemá žádná aktuální data, pozbývá její spuštění významu.



Obrázek 45: Ukázka chybového hlášky při vypnutí Internetu

4.3 Nedostupnost GPS

Nedostupnost polohy se v aplikaci projeví různě při třech různých příležitostech, a to při:

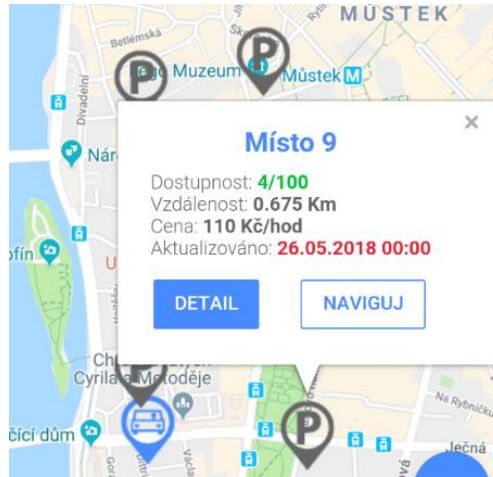
- startu aplikace – zobrazí se vyskakovací okno s hláškou u nedostupnosti polohy, nejsou mu zobrazeny žádné údaje o parkovacích místech ani na mapě nevidí značku aktuální polohy. Na prázdné mapě je také zobrazena informace o tomto problémovém stavu.
- době mezi dvěma aktualizacemi dat – v takovém případě není žádná chyba zaregistrována. Poloha je volána až vždy při aktualizaci dat.
- aktualizaci dat – zobrazí se vyskakovací okno, na mapě je uživateli zobrazena příslušná informace a v seznamu je u položky vzdálenost zobrazena hláška „*!NEAKTUÁLNÍ!*“, data ze serveru jsou ovšem načítána.

4.4 Výpadek serveru

Při výpadku serveru je zásadní, zdali výpadek proběhl tak, že zasáhne načítání dat. Pokud dojde k výpadku během načtených dat na dobu menší, než je další kolo aktualizace, není takovýto výpadek registrován. V momentě výpadku v době požadavku na aktualizaci, se uživateli jednoduše data nezobrazí a v aplikaci je zobrazena chybová hláška odpovědi serveru. Aplikace se dále pokouší o periodické aktualizace nebo je možné, aby uživatel zkontroloval, zdali server již funguje, manuální aktualizací. Jelikož server běží na stabilní platformě třetí strany a při vývoji nebyl žádný výpadek registrován, není počítáno s rozsáhlejšími výpadky, a tak je uživateli zobrazeno chybové hlášení v případě, že není schopen stáhnout data.

4.5 Výpadek dodávky dat jednoho parkoviště nebo skupiny parkovišť

V případě, že parkoviště nenahraje data na server v požadovaném intervalu, nezmění se hodnota „updatedAt“ v databázi viz Obr. 38 na stránce 56. Což je hodnota, kterou aplikace zobrazuje a v případě dat starších než 15 minut takové parkoviště také znázorní. V mapě je to viditelné pomocí zešedivění značky. Dále se tento údaj zobrazí zbarvením na červenou barvu v informačním okně zobrazeném po zvolení mapové značky a také v detailu a seznamu.



Obrázek 46: Ukázka zobrazení neaktuálních (zastaralých) dat

4.6 Zrušení parkoviště

Pokud dojde ke zrušení parkoviště (záznam bude vymazán z databáze), tak po aktualizaci databáze dojde i k jeho smazání v aplikaci. Smazáno bude na všech obrazovkách včetně historie.

4.7 Návrhy na vylepšení aplikace

Při řešení stávající verze aplikace bylo naraženo na několik prvků, které by mohly a měly aplikaci v budoucnu vylepšit. Jedná se například o systém přihlašování, kdy by uživatel mohl svůj profil zobrazit na jakémkoliv mobilním zařízení. S tím spojenou možností vybrat si oblíbená parkoviště, která by se ukládala na server. Dalším vylepšením by mohla být možnost zadat geografické souřadnice a prohlédnout si tak všechna parkoviště včetně údajů o vzdálenosti, potencionální poloze a parkovacích objektech. Měla by být zavedena možnost ukládání historických dat na serveru a následného odhadu budoucích stavů, v určitých dnech a časech, na základě těch minulých. To by mělo uživatelům pomoci při výběru vhodného cíle. Dále by šlo především o rozšíření aplikace o větší množství reálných parkovišť a navázání spolupráce s provozovateli těchto parkovišť za účelem povolení stahování dat z jejich serverů. Další poměrně zajímavou inovací by bylo vytvoření webových stránek, které by byly propojeny s aplikací a nabízely podobné a rozšířené funkce, tentokrát v prostředí webové aplikace. Následně by na toto rozhraní mohly být nabaleny další funkce, které by odpovídaly analýze a novým funkčním požadavkům.

5 Závěr

Hlavní motivací pro vznik této diplomové práce bylo vytvoření mobilní aplikace pro usnadnění hledání volných parkovacích objektů. Při práci bylo postupováno na základě zadání a obecně platných postupů při vývoji mobilní aplikace.

V první kapitole byla nastíněna problematika informačních systémů u dopravy v klidu, jaké jsou jejich cíle, jaké typy informací předávají, a zvláště jaké informace předávají hromadné parkovací objekty. Dále byly popsány hlavní Operační systémy pro mobilní telefony, jejich zastoupení na trhu a přístupy vývoje mobilních aplikací. Byly uvedeny geografické systémy, které se využívají při tvorbě mobilních aplikací, a byla provedena podrobná analýza mobilních aplikací s podobnou tematikou.

Ve druhé kapitole již bylo přistoupeno k návrhu samotné aplikace. Nejdřív byly určeny, na základě kapitoly předchozí, uživatelské požadavky. Tyto požadavky byly dále přeformulovány do případů užití, kde byla rozmyšlena základní logika aplikace. Po kontrole, že všechny funkční požadavky byly v případech užití použity, došlo na první krok návrhu.

První krok návrhu spočíval v základním a hrubém návrhu uživatelského prostředí na papír. Zde byla určena struktura aplikace a její základní rozložení tak, aby obsáhla všechny funkční požadavky. Aplikace byla rozvržena do tří základních stránek: mapy, seznamu a historie. Dále byly navrženy prvotní prvky těchto obrazovek.

V druhém kroku byl návrh převeden do interaktivního návrhového programu. Byly zhodnoceny a přepracovány některé prvky. Zvoleny byly barvy aplikace, vytvořeno logo a vytvořeno kritérium pro zobrazování barev na základě poměru počtu obsazených míst k celkovému počtu míst.

Třetí kapitola začala výběrem, že aplikace bude tvořena na obě platformy pomocí hybridního přístupu vývoje a v prostředí Ionic Framework. Následně byla představena struktura projektu a byl vytvořen návrh uživatelského rozhraní ve zvoleném prostředí, kde byl ukázán vzhled jednotlivých obrazovek na obou platformách a dále byly vysvětleny některé specifické prvky. Bylo vytvořeno mapové rozhraní a kompletní grafický výstup aplikace. V další části jsem nastínil samotnou logiku aplikace, kde jsem popsal nejdůležitější funkce v aplikaci, ukázal propojení se serverem a poukázal na to, že současná verze aplikace data zatím simuluje, proto dalším krokem by bylo stahování dat od provozovatelů parkovišť.

V neposlední řadě byly provedeny uživatelské testy, za jejichž účelem jsem vytvořil jednoduchý dotazník a několik úloh pokrývajících funkční požadavky aplikace a případy užití. Tyto úlohy jsem poté zadal participantům, kteří je do formuláře zhodnotili a dali podnět

k odstranění několika chyb. V neposlední řadě jsem provedl analýzu spolehlivosti při možných poruchách, jako je například nedostupnost internetového připojení.

Stávající poslední verze aplikace není konečná. Jak nastínila kap. 4.7, jsou zde poměrně široké možnosti rozšíření a několik zajímavých funkcí k implementaci. Mezi ně bude patřit možnost přihlášení uživatele a s tím spojená možnost označit si oblíbené parkovací objekty, které se následně uloží a uživatel k nim bude moci přistupovat z jakéhokoliv mobilního zařízení.

Parkování je nezbytnou složkou fungování měst a jeho správná organizace zvyšuje životní úroveň. Vedle uživatelských aplikací je tak nutné investovat do samotných parkovacích objektů včetně záchytných parkovišť tak, aby nabídka dokázala dostatečně uspokojovat poptávku. Jedině tak lze zajistit udržitelnou dopravu i udržitelný život ve velkých městských aglomeracích pro budoucí generace. Tato aplikace může pomoci při výběru vhodného místa a pozitivně tak ovlivňovat dopravu.

Seznam použitých zdrojů

Knihy

- [1] PŘIBYL, Pavel a Miroslav SVÍTEK. Inteligentní dopravní systémy. Praha: BEN – technická literatura, 2001. ISBN 80-7300-029-6.
- [2] VLČKOVÁ, Veronika. Kudy kam geoinformačním inženýrstvím. Praha: Fakulta dopravní ČVUT, 2011. ISBN 9788001049518.

Internetové zdroje

- [3] Operating system market share. Market share for mobile, browsers, operating systems and search engines | NetMarketShare [online]. Copyright © 2017 [cit. 29.05.2018]. Dostupné z: <https://netmarketshare.com/operating-system-market-share.aspx?options=%7B%22filter%22%3A%24and%22deviceType%22%3A%24in%22Mobile%22%2C%22dateLabel%22%3A%22Trend%22%2C%22attributes%22%3A%22share%22%2C%22group%22%3A%22platform%22%2C%22sort%22%3A%22share%22%2C%22id%22%3A%22platformsMobile%22%2C%22dateInterval%22%3A%22Monthly%22%2C%22dateStart%22%3A%222017-05%22%2C%22dateEnd%22%3A%222018-04%22%2C%22segments%22%3A%22-1000%22%7D>
- [4] eParkomat – Aplikace pro Android ve službě Google Play. [online]. Copyright © 2018 Google Inc. [cit. 05.02.2018]. Dostupné z: <https://play.google.com/store/apps/details?id=cz.csp.eparkomat&hl=cs>
- [5] Parkování v Praze – Aplikace pro Android ve službě Google Play. [online]. Copyright © 2018 Google Inc. [cit. 07.02.2018]. Dostupné z: <https://play.google.com/store/apps/details?id=is.stokkur.parking.android&hl=cs>
- [6] Průvodce parkování v Praze – Aplikace pro Android ve službě Google Play. [online]. Copyright © 2018 Google Inc. [cit. 08.02.2018]. Dostupné z: <https://play.google.com/store/apps/details?id=org.hlousek.parking.prague&hl=cs>
- [7] PADĚLEK, Tomáš. Provoz a projektování místních komunikací: Doprava v klidu. [online]. Copyright © 2018 FD CVUT [cit. 08.02.2018]. Dostupné z: http://www.lss.fd.cvut.cz/Members/langr/2010-drup/2010-drup-2-pdf/at_download/file
- [8] Smart Parking: A System that Could Help Cities Rethink Parking. [online]. Copyright © 2018 Siemens [cit. 08.02.2018]. Dostupné z:

<https://www.siemens.com/press/pool/de/events/2015/corporate/2015-09-iaa/background-smart-parking-e.pdf>

- [9] TICHÝ, Tomáš. Řídicí systémy dopravy – dopravní telematika [online]. Copyright © 2018 FD CVUT [cit. 14.02.2018]. Dostupné z: <http://www.lss.fd.cvut.cz/Members/tichy/dokumenty-k-vyuce/ITS>
- [10] Geografické informační systémy – Prohloubení nabídky dalšího vzdělávání v oblasti zeměměřičtví a katastru nemovitostí ve Středočeském kraji [online]. Copyright © Střední průmyslová škola zeměměřická v Praze [cit. 15.02.2018]. Dostupné z: <http://spszem.cz/storage/files/1332/ArcGIS1.pdf>
- [11] Jak vyvíjet mobilní aplikace [online]. Copyright © Jan Václavík [cit. 15.02.2018]. Dostupné z: <http://janvaclavik.cz/jak-vyvijet-mobilni-aplikace/>
- [12] Hybridní, nativní nebo webové aplikace? [online]. Copyright © iQuest s.r.o. [cit. 15.02.2018]. Dostupné z: <http://www.hybridniaplikace.cz/srovnani>
- [13] Weby versus aplikace [online]. Copyright © Martin Michálek [cit. 15.02.2018]. Dostupné z: <https://www.vzhurudolu.cz/prirucka/weby-vs-aplikace>
- [14] Parken in Mannheim – Aplikace pro Android ve službě Google Play. [online]. Copyright © 2018 Google Inc. [cit. 15.02.2018]. Dostupné z: <https://play.google.com/store/apps/details?id=de.parkenmannheim.parkenmpb&hl=cs>
- [15] Google Developers [online]. © 2018 Google Inc. [cit. 15.02.2018]. Dostupné z: <https://developers.google.com/maps/documentation/javascript/tutorial>
- [16] Leaflet – a JavaScript library for interactive maps. Leaflet – a JavaScript library for interactive maps [online]. Copyright © Vladimir Agafonkin [cit. 04.03.2018]. Dostupné z: <http://leafletjs.com>
- [17] Build apps with HERE Maps API and SDK Platform Access – HERE Developer. Build apps with HERE Maps API and SDK Platform Access – HERE Developer [online]. Copyright © HERE [cit. 05.03.2018]. Dostupné z: <https://developer.here.com>
- [18] A Cry For Looking To Other Methods For User Centered Design – Usabilla Blog. Usabilla Blog [online]. Dostupné z: <http://blog.usabilla.com/a-cry-for-looking-to-other-methods-for-user-centered-design/>
- [19] GitHub – SebastianM/angular-google-maps: Angular 2+ Google Maps Components. The world's leading software development platform · GitHub [online]. Copyright © 2018 [cit. 21.05.2018]. Dostupné z: <https://github.com/SebastianM/angular-google-maps>

- [20] Ionic Native – Diagnostic. Build Amazing Native Apps and Progressive Web Apps with Ionic Framework and Angular [online]. Copyright © 2018 [cit. 21.05.2018]. Dostupné z: <https://ionicframework.com/docs/native/diagnostic/>
- [21] GitHub – katzer/cordova-plugin-email-composer: Edit and send email messages. The world's leading software development platform · GitHub [online]. Copyright © 2018 [cit. 21.05.2018]. Dostupné z: <https://github.com/katzer/cordova-plugin-email-composer>
- [22] GitHub – apache/cordova-plugin-geolocation: Mirror of Apache Cordova Plugin geolocation. The world's leading software development platform · GitHub [online]. Copyright © 2018 [cit. 21.05.2018]. Dostupné z: <https://github.com/apache/cordova-plugin-geolocation>
- [23] GitHub – dpa99c/phonegap-launch-navigator: Phonegap/Cordova plugin which launches native route navigation apps for Android, iOS and Windows. The world's leading software development platform · GitHub [online]. Copyright © 2018 [cit. 21.05.2018]. Dostupné z: <https://github.com/dpa99c/phonegap-launch-navigator>
- [24] GitHub – apache/cordova-plugin-network-information: Mirror of Apache Cordova Plugin network-information. The world's leading software development platform · GitHub [online]. Copyright © 2018 [cit. 21.05.2018]. Dostupné z: <https://github.com/apache/cordova-plugin-network-information>
- [25] PhoneGap. PhoneGap [online]. Copyright © 2016 Adobe Systems Inc. All rights reserved. [cit. 26.05.2018]. Dostupné z: <https://www.phonegap.com>
- [26] Dokumentace pro Xamarin. [online]. Copyright © 2016 Microsoft. All rights reserved. [cit. 26.05.2018]. Dostupné z: <https://docs.microsoft.com/cs-cz/xamarin/>
- [27] Cross Platform IOS, Iphone Mobile App Development Using Java. Cross Platform IOS, Iphone Mobile App Development Using Java [online]. Copyright © 2012 All Rights Reserved [cit. 26.05.2018]. Dostupné z: <https://www.codenameone.com>
- [28] React Native · A framework for building native apps using React. [online]. Copyright © 2018 Facebook Inc. [cit. 26.05.2018]. Dostupné z: <https://facebook.github.io/react-native/>
- [29] Build Amazing Native Apps and Progressive Web Apps with Ionic Framework and Angular. Build Amazing Native Apps and Progressive Web Apps with Ionic Framework and Angular [online]. Copyright ©2018 Ionic [cit. 26.05.2018]. Dostupné z: <https://ionicframework.com>
- [30] FALTUS, Vladimír a Roman SRP, Doprava a rozvoj ITS v Pražské aglomeraci [online]. Copyright © [cit. 29.05.2018]. Dostupné z:

http://www.sdt.cz/dokumenty/2018_Pozicni_dokument_SDT_Doprava_a_rozvoj_ITS_v_PAG.pdf

Softwarové zdroje

- [31] Getting Started with Ionic. Build Amazing Native Apps and Progressive Web Apps with Ionic Framework and Angular [online]. Copyright ©2018 Ionic [cit. 29.05.2018].
Dostupné z: <https://ionicframework.com/getting-started>

Seznam obrázků

Obrázek 1: Ukázka poskytovaných dat od TSK Praha	12
Obrázek 2: Architektura telematického parkovacího systému	12
Obrázek 3: Hlavní obrazovka aplikace eParkomat	17
Obrázek 4: Detail parkoviště v aplikaci eParkomat	18
Obrázek 5: Úvodní obrazovka aplikace Parkování v Praze	20
Obrázek 6: Ukázka platby v aplikaci Parkování v Praze	20
Obrázek 7: Detail v aplikaci Parkování v Praze	21
Obrázek 8: Úvodní obrazovka aplikace Průvodce parkování v Praze	23
Obrázek 9: Detail parkoviště v aplikaci Průvodce parkování v Praze	24
Obrázek 10: Hlavní obrazovka aplikace Parken in Mannheim	26
Obrázek 11: Detail aplikace Parken in Mannheim	27
Obrázek 12: Princip UCD [18]	29
Obrázek 13: Navrhovaný Use Case diagram aplikace	31
Obrázek 14: Prvotní návrh úvodní obrazovky	37
Obrázek 15: Prvotní návrh obrazovky se seznamem	37
Obrázek 16: Prvotní návrh obrazovky detailu	38
Obrázek 17: Hlavní logo a zároveň splash screen aplikace	39
Obrázek 18: Hlavní obrazovka aplikace v druhém kole návrhu	39
Obrázek 19: Detailní informace o aplikaci v druhém kole návrhu	40
Obrázek 20: Seznam parkovišť a historie navštívených míst v druhém kole návrhu	41
Obrázek 21: Průřez strukturou projektu	43
Obrázek 22: Záhlaví aplikace bez a se zobrazeným menu (nahore Android, dole iOS)	44
Obrázek 23: Navigační menu aplikace (nahore Android, dole iOS)	45
Obrázek 24: Úvodní stránka aplikace (vlevo Android, vpravo iOS)	45
Obrázek 25: Přehled vytvořených mapových značek	46
Obrázek 26: Zobrazení informačního okna v případě aktuálních i starých dat	47
Obrázek 27: Obrazovka detailu parkoviště (vlevo Android, vpravo iOS)	48
Obrázek 28: Základní vzhled obrazovky (vlevo Android, vpravo iOS)	49
Obrázek 29: Ukázka okna pro seřazení podle (vlevo Android, vpravo iOS)	50
Obrázek 30: Ukázka seznamu s rozbaleným filtrem (vlevo Android, vpravo iOS)	50
Obrázek 31: Ukázka obrazovky historie (vpravo Android, vlevo iOS)	51
Obrázek 32: Obrazovky „o nás“ a „kontaktuje nás“	52
Obrázek 33: Ukázka import pluginu Geolocation v souboru app.module.ts	52
Obrázek 34: Ukázka případu při vypnuté GPS, alert a chybová hláška	54
Obrázek 35: Druhý typ chybové hlášky	54

Obrázek 36: Ukázka staré polohy na obrazovce seznamu.....	54
Obrázek 37: Algoritmus určení polohy	55
Obrázek 38: Ukázka struktury tabulky a jednoho zápisu v prostředí serveru.....	56
Obrázek 39: Ukázka dotazu na server a uložení dat do pole	57
Obrázek 40: Algoritmus aktualizace dat.....	58
Obrázek 41: Ukázka podmínky pro výběr barvy značky.....	59
Obrázek 42: Ukázka výběru požadované aplikace k navigaci	60
Obrázek 43: Ukázka formuláře pro uživatelské testování.....	61
Obrázek 44: Ukázka obrazovky Podpora a Legenda	63
Obrázek 45: Ukázka chybového hlášky při vypnutí Internetu	63
Obrázek 46: Ukázka zobrazení neaktuálních (zastaralých) dat.....	65

Seznam tabulek

Tabulka 1: Klady a zápory prostředků předávání informací.....	11
Tabulka 2: Porovnání Androidu a iOS.....	14
Tabulka 3: Zhodnocení kladů a záporů jednotlivých přístupů vývoje aplikací [11][12][13]	15
Tabulka 4: Význam barev v aplikaci eParkomat.....	17
Tabulka 5: Klady a zápory aplikace eParkomat.....	19
Tabulka 6: Klady a zápory aplikace Parkování v Praze.....	23
Tabulka 7: Význam barev v aplikaci Průvodce parkování v Praze.....	24
Tabulka 8: Klady a zápory aplikace Průvodce parkování v Praze	25
Tabulka 9: Klady a zápory aplikace Parken in Mannheim	27
Tabulka 10: Zhodnocení aplikací	28
Tabulka 11: Postup návrhu a tvorby.....	29
Tabulka 12: Vztah funkčních požadavků a případů užití	36
Tabulka 13: Barevné rozlišení značek	40
Tabulka 14: Porovnání nástrojů pro vývoj hybridních aplikací	42

Seznam příloh

Přílohy jsou k nalezení na přiloženém CD – ROM.

Příloha A: Zdrojový kód aplikace

Kod.zip

Příloha B: Interaktivní návrh

Navrh.zip

Příloha C: Instalační soubor pro platformu Android

Instalace.zip

Příloha D: Dotazníky s uživatelskými testy

Dotazniky.pdf