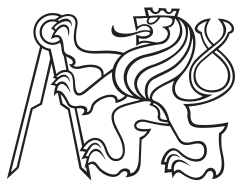


Bakalářská práce



České  
vysoké  
učení technické  
v Praze

**F3**

Fakulta elektrotechnická  
Katedra počítačů

## Optimalizace využití parkovacích míst

**Jakub Ďuriš**

Školitel: doc. Ing. Ivan Jelínek, CSc.  
Obor: Softwarové inženýrství a technologie  
Srpen 2023



## I. OSOBNÍ A STUDIJNÍ ÚDAJE

Příjmení: **Đuriš** Jméno: **Jakub** Osobní číslo: **499245**  
Fakulta/ústav: **Fakulta elektrotechnická**  
Zadávací katedra/ústav: **Katedra počítačů**  
Studijní program: **Softwarové inženýrství a technologie**

## II. ÚDAJE K BAKALÁŘSKÉ PRÁCI

Název bakalářské práce:

**Optimalizace využití parkovacích míst**

Název bakalářské práce anglicky:

**Optimizing the use of parking spaces**

Pokyny pro vypracování:

Cílem bakalářské práce je návrh a implementace aplikace pro optimální využití parkovacích míst parkoviště, kde část míst je volně přístupná a část míst je dedikována na konkrétní zaměstnance. Cílem optimalizace je dynamicky zpřístupnit aktuálně volná místa zaměstnanců, o nichž je známo, že v daný den nebudou využita zaměstnancem, aby byla k dispozici klientům firmy.

Provedte rešerši existujících přístupů k řešení, na základě této rešerše specifikujte a upřesněte požadavky na optimalizaci přidělování parkovacích míst.

Navrhněte vhodné implementační prostředky na základě rešerše těchto prostředků.

V souladu s požadavky zadavatelské společnosti vyberte optimální variantu výstupu a datové prezentace.

Navrhněte způsob testování aplikace a proveďte testování, výsledky testů vyhodnoťte.

Aplikace musí být otestována tak, aby správně pracovala i pod zátěžovým náporům uživatelů.

Výstupem bakalářské práce bude aplikace podle navržené specifikace tak, aby vyhovovala potřebám (a případným omezením) zadavatelské společnosti.

Seznam doporučené literatury:

James Ferreira, Google Script: Enterprise Application Essentials: Adding Functionality to Your Google Apps, 2012, O'Reilly Media

Maguire, M. Google Sheets Programming With Google Apps Script (2015 Revision Complete), 2015

Estes, Janelle; MacMillan, Andy User Tested: How the World's Top Companies Use Human Insight to Create Great Experiences. 2022. ISBN: 9781119844723.

Jméno a pracoviště vedoucí(ho) bakalářské práce:

**doc. Ing. Ivan Jelínek, CSc. kabinet výuky informatiky FEL**

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

Datum zadání bakalářské práce: **09.02.2023**

Termín odevzdání bakalářské práce: **15.08.2023**

Platnost zadání bakalářské práce: **22.09.2024**

doc. Ing. Ivan Jelínek, CSc.  
podpis vedoucí(ho) práce

podpis vedoucí(ho) ústavu/katedry

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

### III. PŘEVZETÍ ZADÁNÍ

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

\_\_\_\_\_  
Datum převzetí zadání

\_\_\_\_\_  
Podpis studenta

## Poděkování

Chtěl bych poděkovat panu doc. Ing. Ivanu Jelínkovi, CSc. za odborné vedení a cenné rady, které mi pomohly tuto práci zkompletovat.

## Prohlášení

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

## Abstrakt

Cílem této práce je vytvoření systému pro optimalizaci využití parkovacích stání na pozemku zadavatelské firmy. Součástí je detailní specifikace tohoto systému, řešení přístupů k řešení a následná implementace aplikace vyhovující všem požadavkům, které vycházejí z provedené analýzy. V neposlední řadě je zde zachycen proces testování, díky kterému je zaručen správný chod této aplikace.

**Klíčová slova:** Webová aplikace, Google Apps Script, JavaScript, parkování, optimalizace, digitalizace, projekt, specifikace

**Školitel:** doc. Ing. Ivan Jelínek, CSc.  
ČVUT FEL,  
Technická 2,  
16000 Praha 6

## Abstract

The aim of this work is to create a system for optimizing the utilization of parking spaces on the premises of the client's company. It includes a detailed specification of this system, research on various approaches to address the issue, and subsequent implementation of an application that meets all the requirements derived from the conducted analysis. Last but not least, the testing process is documented to ensure the proper functionality of this application.

**Keywords:** Web application, Google Apps Script, JavaScript, parking, optimization, digitization, project, specification

**Title translation:** Optimizing the use of parking spaces

## Obsah

<b>1 Úvod</b>	<b>1</b>	9.3 Shrnutí . . . . .	31
1.1 Cíl projektu . . . . .	2	<b>10 Testování</b>	<b>35</b>
<b>2 Rešerše existujících řešení</b>	<b>3</b>	10.1 Testy průchodů . . . . .	35
2.1 Aktuální situace . . . . .	3	10.1.1 Testovací sekvence . . . . .	36
2.2 Existující řešení . . . . .	3	10.2 Jednotkové testy . . . . .	36
<b>3 Analýza požadavků</b>	<b>5</b>	10.3 Tabulkové testy . . . . .	38
3.1 Technologická omezení . . . . .	5	10.4 Výkonnostní testy . . . . .	39
3.2 FURPS+ analýza . . . . .	6	10.5 Revize kódu . . . . .	39
3.2.1 Funkčnost . . . . .	6	10.6 Testování přijetí uživatelem . . .	39
3.2.2 Použitelnost . . . . .	6	<b>11 Závěr</b>	<b>41</b>
3.2.3 Spolehlivost . . . . .	6	11.1 Vyhodnocení implementace . . .	41
3.2.4 Výkon . . . . .	7	11.1.1 Požadavky . . . . .	41
3.2.5 Podpora . . . . .	7	11.1.2 Optimalizace . . . . .	41
3.2.6 Nefunkční požadavky . . . . .	7	11.1.3 Testování . . . . .	42
3.3 Požadavky na data systému . . . . .	7	11.2 Shrnutí výsledků . . . . .	42
3.4 Správcovský přístup . . . . .	8	11.3 Možnosti rozvoje . . . . .	43
3.5 Shrnutí . . . . .	9	<b>12 Příloha</b>	<b>45</b>
<b>4 Návrh řešení</b>	<b>11</b>	<b>Bibliografie</b>	<b>47</b>
4.1 Webová aplikace . . . . .	11		
4.2 Optimalizace . . . . .	11		
<b>5 Analýza vstupních dat</b>	<b>13</b>		
5.1 Princip fungování . . . . .	13		
5.2 Pozorované závěry . . . . .	14		
5.3 Shrnutí . . . . .	14		
<b>6 Požadavky</b>	<b>15</b>		
6.1 Funkční požadavky . . . . .	15		
6.2 Nefunkční požadavky . . . . .	15		
<b>7 Implementace</b>	<b>19</b>		
7.1 Návrh aplikace . . . . .	19		
7.2 Vývoj . . . . .	20		
<b>8 Využitá technologie</b>	<b>23</b>		
8.1 Google Apps Script . . . . .	23		
8.2 Uchování dat . . . . .	24		
8.3 Struktura projektu . . . . .	24		
8.4 Omezení . . . . .	25		
8.5 Shrnutí . . . . .	26		
<b>9 Uživatelské rozhraní</b>	<b>27</b>		
9.1 Rozdělení přístupu . . . . .	27		
9.2 Hlavní obrazovky . . . . .	27		
9.2.1 Rozložení pro mobilní telefon	28		
9.2.2 Rozložení pro počítač . . . . .	29		
9.2.3 Ostatní obrazovky . . . . .	29		
9.2.4 Správcovské zobrazení . . . . .	30		

## Obrázky

7.1 Diagram nasazení . . . . .	19
7.2 Diagram případů užití . . . . .	20
7.3 Procesní diagram základního průchodu systémem . . . . .	21
7.4 High fidelity prototyp . . . . .	21
7.5 Stavový diagram . . . . .	22
7.6 Fáze vodopádové metodiky[10] . . . . .	22
8.1 Náhled na editor Google Script aplikací . . . . .	23
9.1 Rozložení základních obrazovek na mobilním zařízení . . . . .	28
9.2 Rozložení základních obrazovek na počítači . . . . .	29
9.3 Obrazovky s různými druhy upozornění . . . . .	30
9.4 Správcovské zobrazení umožňující úpravu údajů o parkovacích místech . . . . .	31
9.5 Dialogové okno s možnostmi editace parkovacího stání . . . . .	31
9.6 Návodová vysvětlující barevné návrhy správcovské tabulky . . . . .	32
9.7 Upozornění varující uživatele před předčasným vypnutím aplikace . . . . .	32
10.1 Rezervace v diagramu aktivit . . . . .	36
10.2 Rezervace převedená na orientovaný graf . . . . .	37

## Tabulky

6.1 Seznam funkčních požadavků . . . . .	16
6.2 Seznam nefunkčních požadavků - 1. část . . . . .	17
6.3 Seznam nefunkčních požadavků - 2. část . . . . .	18
10.1 Testovací sekvence . . . . .	38
10.2 Pokrytí jednotkových testů . . . . .	38
10.3 Výsledky výkonnostního testování . . . . .	40



# Kapitola 1

## Úvod

Otázka nedostatku parkovacích míst se v dnešní společnosti stává stále důležitější. Většina domácností disponuje alespoň jedním vozem, kterým jezdí na denní bázi. Z historických důvodů je v evropských zemích nicméně většina ulic ve městech velmi úzká, protože často vznikaly v době, kdy byly uzpůsobené pro pěší provoz. [1]

Od dob výlučně pěšího přesunu z místa na místo nicméně už uběhla dlouhá doba a podstatná část prostorů mezi domy je v dnešní době uzpůsobena pro provoz aut. O to méně se nabízí míst pro parkování. Z tohoto důvodu je nezbytné nakládat efektivně s místy, která jsou k dispozici. To platí o to více u velkých společností, mezi které patří i zadavatel projektu. Do práce každý den přijíždí autem stovky zaměstnanců, kteří potřebují zaparkovat. Počet míst je ovšem omezen. Je proto potřeba co nejlépe všechna dostupná místa zužítkovat.

Právě z toho důvodu je cílem této práce vytvoření aplikace umožňující optimalizaci a digitalizaci parkování před budovou zadavatelské firmy. Tedy zajištění zužítkování i těch parkovacích stání, která by za běžných okolností zůstala prázdná.

Parkoviště je důležitou, ale často zanedbávanou možností firmy, jak udělat dobrý první dojem i na návštěvníky. To platí obzvláště v případě, že se jedná o parkoviště firmy zabývající se vývojem komponentů do aut, které zahrnují vše od elektrických součástek až po systémy autonomního řízení. Služba, jež pomůže každému zájemci najít dostupné parkovací místo, může v očích návštěvníka zvednout mínění o dané společnosti ještě před vstupem do budovy.

Elektronická správa parkoviště usnadňuje profesní život hlavně řadovým zaměstnancům. Ti díky ní získají schopnost najít místo k zaparkování rychleji a jednodušeji. Tím se pro ně zkracuje čas tím strávený a spolu s ním i celková doba transportu do práce a z ní. Zároveň představuje možnost, jak získat přehled o stavu kapacity parkoviště ještě před samotným příjezdem. Tento projekt se proto zabývá návrhem a vývojem aplikace zaměřené na využití těmito zaměstnanci, kteří budou v každodenním životě nejvíce profitovat z jejího zavedení do provozu a z optimalizace, kterou spolu s tím přinese.

## 1.1 Cíl projektu

Cílem projektu je systém pro optimální využití parkovacích míst parkoviště, kde část míst je volně přístupná a část míst je určena pro konkrétní zaměstnance. Cílem optimalizace je dynamicky zpřístupnit aktuálně volná místa zaměstnanců, o nichž je známo, že v daný den nebudou využita, jiným zájemcům z řad společnosti.

Součástí projektu je i splnění požadavků zadavatelské společnosti na podobu a funkčnost aplikace. Ty vyplývají z analytické části projektu, kde je rozebráno současné řešení a stav a vytvořena specifikace pro aplikaci.

Celý systém je pro firmu prvním krokem k automatizaci a digitalizaci procesu parkování a má poskytnout možnost v budoucnu snadno rozšířit svou funkcionalitu. Elektronická správa míst totiž umožňuje implementaci dalších nadstaveb, jakou je například aktuální zobrazení obsazenosti parkovacích stání. Tuto informaci je následně možné zajímavě prezentovat, ať už na interaktivním kiosku u vjezdu, v online aplikaci umožňující rezervaci konkrétního místa, nebo jednoduše na obrazovce informující o počtu volných stání.

## Kapitola 2

### Rešerše existujících řešení

#### 2.1 Aktuální situace

Parkoviště společnosti disponuje kapacitou zhruba dvou set parkovacích míst. Většina těchto míst není nikomu přiřazena, stát na nich smí všichni zaměstnanci. Zbytek parkovacích stání je dedikovaný konkrétnímu uživateli, nebo pro konkrétní účel. Může například být vyhrazené pro konkrétního zaměstnance ve vyšším managementu, případně pro parkování sdílených firemních vozů.

Všechna parkovací místa mají zaznamenaný aktuální způsob využití, tedy v případě, že místo slouží jako dedikované, je zřejmé, komu bylo přiřazeno. Tento zaměstnanec má následně výhradní právo k využívání místa, nehledě na skutečnost, jestli se zrovna nachází v kanceláři, potažmo jestli ho daný den využije. To znamená, že v případě, kdy parkování neobsadí, bude místo celý den prázdné, jelikož nikdo jiný k němu nemá přístup. Tento způsob využití dostupných parkovacích míst je velmi neefektivní, neboť ostatní parkovací místa se poměrně rychle zaplní a další zaměstnanci již nemají kde zaparkovat, i přesto, že spousta parkovacích stání zeje prázdnotou.

#### 2.2 Existující řešení

Po konzultaci se zadavatelským oddělením bylo vyrozuměno, že v současné době firma nedisponuje žádným systémem pro správu parkoviště ani parkovacích míst. Nově vytvořený systém tak bude úplně prvním krokem v rozvoji procesu parkování. Firma má sice k dispozici tabulky s obecnými informacemi o parkovišti, které jsou pravidelně aktualizovány, nicméně ty samy o sobě nemohou sloužit jako způsob optimalizace parkování. Jsou však dobrým zdrojem pro systém, který by data z nich zpracovával a prezentoval tak, aby se situace na parkovišti zlepšila.



## Kapitola 3

### Analýza požadavků

Na základě rozhovorů se zástupci vedení firmy byly sepsány požadavky, které určují, jak by systém měl vypadat. Následující kapitola se věnuje popisu těchto požadavků. Popisují obecný, na platformě nezávislý systém, který je postupně upřesňován podle kladených nároků. Na základě těchto dat byly následně zadavateli představeny návrhy možných řešení, které z nich vychází.

#### 3.1 Technologická omezení

Ve firmě probíhá běžná práce v aplikacích od společnosti Google – Sheets, Docs, Drive. Všechny automatické systémy se spouští nad tabulkou Google Sheets a jsou psány ve skriptovacím jazyce na bázi JavaScriptu, Google Apps Script. Ten využívá tabulkové funkce dostupné v Sheets a rozšiřuje je mimo jiné o webové aplikace s vlastním uživatelským prostředím[3]. Z tohoto důvodu je nutné celou aplikaci implementovat v tomto programovacím jazyce a nad jednou konkrétní tabulkou Google Sheet.

Potíž nastává s databází, která není ve firmě vůbec využívána. Data se ukládají do tabulek Google Sheets a stejný postup bude muset být dodržen i zde. Tabulky by se snad daly nazvat jakýmsi prostým databázovým souborem[4], nicméně oproti klasickým SQL databázím mají spoustu nevýhod. Mezi hlavní patří složitější přístup ke konkrétním datům z důvodu nutnosti simulace SQL dotazů jinými způsoby. Konkrétně jsou data z tabulek získávána pomocí funkcí ze třídy Sheet, což je interní třída Google Script aplikací. Hodnoty z listu konkrétní tabulky jsou uloženy do dvourozměrného pole, také nazývaného matice[5]. Zde vznikají hned dva problémy. Prvním je samotné načítání těchto hodnot, které je velmi pomalé, neboť vyžaduje otevření tabulky, načtení dat a jejich návrat. Při přístupu k těmto datům je tedy z důvodu zachování rozumné rychlosti běhu programu nutné načítat všechna data naráz, pracovat s nimi, a potom opět všechna aktualizovat. To je náročné na paměť.

Po načtení informací přichází druhý zádrhel, který je způsoben tím, že hodnoty jsou uloženy v matici, což si vynucuje dohledávání konkrétních hodnot procházením celého pole. To opět zpomaluje běh programu.

## ■ 3.2 FURPS+ analýza

FURPS analýza je nástroj, který analyzuje a prioritizuje požadavky (na systém). Skládá se z 5 částí: [6]

- Funkčnost (Functionality)
- Použitelnost (Usability)
- Spolehlivost (Reliability)
- Výkon (Performance)
- Podpora (Serviceability)

Symbol + v tomto případě znamená rozšíření o nefunkční požadavky. Ty nepopisují co bude systém dělat, ale jak to bude dělat[7]. Například upřesňují barevné rozložení systému, nebo v jakých jazycích má být dostupné uživatelské prostředí.

### ■ 3.2.1 Funkčnost

V systému bude možné zobrazit si aktuálně dostupná dedikovaná parkovací místa a zarezervovat si je pro vlastní použití. Přístup zaměstnanců k rezervaci bude nicméně omezen a bude se manuálně spouštět ve chvíli, kdy se ostatní parkovací místa zaplní. Nebude zároveň možné rezervovat si všechna dostupná místa na parkovišti, do systému budou vložena pouze dedikovaná parkovací stání, která jsou většinou přiřazena zaměstnancům managementu.

Aplikace bude obsahovat také rozhraní pro správu parkovacích míst, kterou budou moci využít zaměstnanci se správcovskými právy. V té bude možné konkrétní místo odstranit z nabídky, aby nebylo možné vytvořit si na něm rezervaci ani během nepřítomnosti jeho majitele. Dále bude možné místa přidat nebo odebrat z nabídky pouze pro daný den. Taktéž se odsud bude celá aplikace spouštět.

### ■ 3.2.2 Použitelnost

Webové rozhraní bude optimalizováno pro prohlížeče Google Chrome a Microsoft Edge. Očekává se primárně responzivní návrh umožňující jednoduché použití z mobilních telefonů s operačním systémem Android. To z důvodu, že většina uživatelů bude do aplikace přistupovat během cesty do práce, aplikace je proto nesmí nutit věnovat jí více pozornosti, než je nezbytně nutné.

### ■ 3.2.3 Spolehlivost

Přesnost vrácených dat je závislá výlučně na vstupních informacích, které nejsou ve správě této aplikace. Aplikace by měla zvládnout naráz obsloužit 100 souběžně přistupujících uživatelů. Zásadní je co nejvyšší spolehlivost v

ranních hodinách, kdy lze předpokládat nejvyšší nápor uživatelů. Informace o vytvořených rezervacích musí být správné za každé situace.

Aplikace nesmí obsahovat žádné kritické chyby, které by znemožňovaly její použití. Všechny tyto aspekty budou ověřeny testovací fází, ve které proběhne zejména kontrola správnosti běhu páteřních funkcí a robustnost při větším náporu uživatelů.

#### ■ 3.2.4 Výkon

Očekává se běžně uživatelsky akceptovaný výkon a rychlost načítání dat, v nejvytíženější moment může být aplikace naráz využívána více než sto zaměstnanci. Datový výstup by měl být zobrazen do 5 vteřin.

Zároveň je ale nutné ověřovat správnost dat v každém kroku vytváření rezervace uživatelem. Konkrétně to znamená, že každá akce ze strany uživatele musí být překontrolována aktualizací vstupních dat z tabulky, což bude běh poměrně zdržovat.

#### ■ 3.2.5 Podpora

Kompletní dokumentace systému bude ve vlastnictví zadavatele, ten bude zároveň zajišťovat udržování a podporu aplikace.

#### ■ 3.2.6 Nefunkční požadavky

Uživatelské rozhraní bude dostupné v anglickém jazyce a bude navrženo ve stylu co nejpodobnějším ostatním aplikacím zadavatele. To se týká především fontu písma a barevného návrhu. Všechny stránky a části aplikace musí také obsahovat logo společnosti.

### ■ 3.3 Požadavky na data systému

Systém by měl zaměstnancům umožnit zarezervovat si jedno z dostupných dedikovaných míst. Seznam dostupných míst s identifikačními čísly jejich majitelů a schválené absence pro daný den jsou ve správě jiných oddělení firmy, tedy požadavky na kvalitu těchto vstupních dat nejsou specifikovány, ale aplikace z nich má sama vycházet. Vše je uloženo v Google Sheets. Všechna data vystupující z aplikace musí být opět uložena do tabulky Google Sheets.

Dostupné informace se nachází ve 2 různých tabulkách. První tabulkou je prostý seznam identifikačních čísel všech zaměstnanců, kteří mají na daný den schválenou absenci v docházkovém systému. Tabulka je automaticky generovaná a aktualizuje se každý den v ranních hodinách. Druhá z tabulek je rozsáhlejší a obsahuje všechny údaje o parkovištích a místech. Je ve správě personálního oddělení.

Pro účely systému je nutné všechny listy se zdroji kopírovat do třetí tabulky, která může být na rozdíl od originálů editována. Nad tou poběží celá aplikace. V této tabulce se vytvoří seznam všech osob ze seznamu absencí,

kteří mají zároveň přiřazené parkovací místo. Tato stání se potom budou nabízet ostatním zájemcům k zaparkování. Musí být dále možné tento seznam manuálně upravit tak, aby bylo možné konkrétní místo z něj vyndat, nebo do něj naopak vložit. To přijde vhod, pokud vzniknou nesrovnalosti v přehledu absencí. Současně bude tabulka sloužit pro ukládání informací o systému. Bude tedy obsahovat mimo jiné stav jeho zapnutí, odkazy na obrázky nebo seznam emailových adres zaměstnanců se správcovským přístupem. Důležitým aspektem je, že aplikace nebude ukládat žádné časové údaje. Její běh bude ohraničen manuálním zapnutím ráno a automatickým vypnutím na konci dne. Bude totiž koncipována jako jakýsi nouzový způsob, jak vytvořit na parkovišti ještě o trochu více míst k zaparkování. Není totiž žádoucí, aby si kdokoliv mohl rezervovat konkrétní místo na několik dní dopředu, a to hned z několika důvodů. Prvním a nejdůležitějším je fakt, že dedikovaná místa jsou stále platným bonusem k manažerské pozici. Vzniká proto riziko následující situace. Manažer má schválenou několikadenní absenci. To někomu jinému umožní rezervovat si jeho místo na několik dní. Tento zaměstnanec už si pravděpodobně nebude každý den kontrolovat aplikaci, jestli má místo stále rezervované. I kdyby bylo řečeno, že se toto kontrolovat musí, může na to zapomenout, nebo může auto nechat v práci přes noc. Do toho se manažerovi změní rozvrh, najednou bude nečekaně potřebovat do kanceláře a při příjezdu zjistí, že je jeho místo obsazeno. Omezení vytváření rezervací pouze na jeden daný den toto riziko zmírňuje. Dalším argumentem proti několikadenním rezervacím je situace, kdy si zaměstnanec zablokuje parkování dopředu a pak ho nevyužije. Například dorazí do práce veřejnou dopravou. Tím je opět místo zabráno, ale nevyužito.

## 3.4 Správcovský přístup

Kromě rezervací parkovacích míst bude aplikace umožňovat i jejich správu. Přístup do této části bude umožněn vybraným zaměstnancům z personálního oddělení, kteří měli i doteď parkování na starost. Tito správci se mohou libovolně měnit. Proto musí pověřená osoba mít možnost upravit jejich seznam v aplikaci.

Osoba s přístupem do správcovského režimu bude mít možnost vstoupit do přehledu parkovacích míst, kde bude možné provést základní úpravy. Zejména pak takové, které budou mít přímý vliv na chod aplikace. Systém bude nicméně koncipován jako jednorázová možnost, jak vytvořit na parkovišti pár míst k zaparkování navíc. Proto ani správcovská konzole nebude poskytovat příliš rozšířenou funkcionalitu, přesto by ale měla umožnit následující úkony:

- Zapnutí a vypnutí aplikace
- Odebrání místa z nabídky natrvalo
- Odebrání místa z nabídky pro daný den
- Základní statistické údaje



## ■ 3.5 Shrnutí

Výstupem bude webová aplikace vytvořená nad tabulkou Google Sheets. Ta zaměstnancům firmy umožní jednoduchou rezervaci volného místa k zaparkování. Z vygenerovaného seznamu schválených absencí pro daný den se vyberou ti zaměstnanci, kteří mají přiřazeno dedikované parkovací stání, a ta se nabídnou ostatním zájemcům z řad společnosti.

Na aplikaci nebyly vzneseny žádné speciální požadavky ohledně rychlosti a dostupnosti dat. Očekává se proto dostatečný výkon, který nebude uživatele zásadním způsobem omezovat. Načtení dat by mělo proběhnout do 5 vteřin. Nejdůležitějším aspektem je poskytnutí dostatečně aktualizovaných a správných záznamů ze strany ostatních oddělení firmy. Dále musí být snadno použitelná na mobilním zařízení. Největší nápor uživatelů se očekává v ranních hodinách, hned po spuštění.

Hlavním uživatelským procesem je přihlášení zaměstnance k využívání daného parkovacího stání konkrétní den. Přístup k tomuto rezervačnímu systému bude spuštěn manuálně jeho správcem ve chvíli, kdy se kapacita volně dostupných míst zaplní.



## Kapitola 4

### Návrh řešení

Na základě sesbíraných požadavků byl vytvořen návrh řešení, který byl odsouhlasen zadavatelským oddělením společnosti.

#### 4.1 Webová aplikace

Vybraným řešením je samostatná webová aplikace. Ta bude zaměstnancům umožňovat pod svým jménem vytvořit rezervaci, která jim umožní parkovat na konkrétním parkovacím místě. Místa budou přiřazována automaticky ze seznamu. Ten budou tvořit ta parkovací stání, která jsou někomu přiřazena, většinou konkrétnímu manažerovi. Z nich ale budou pro možnost rezervace vybírána pouze taková místa, která nebudou daný den svým majitelem obsazena. To se zjišťuje z docházkového systému společnosti.

Aplikace dále přinese možnost využít tato parkovací místa, která dosud využít možné nebylo. Tím se optimalizuje proces výběru místa k zaparkování, protože se vytvoří nové parkovací možnosti.

#### 4.2 Optimalizace

Hlavním cílem práce je optimalizace využití parkovacích míst. Ta umožní, aby byla využita i taková parkovací stání, která dosud nebyla běžně dostupná. Ve firmě dosud neexistoval žádný systém, který by se přiřazováním parkovacích míst zabýval, není tedy z tohoto hlediska co vylepšovat, optimalizovat. Zde navržený systém je prvním krokem firmy k vylepšení využití dostupných parkovacích míst.

Optimalizace v rámci tohoto systému spočívá dle specifikace ve zvýšení kapacity parkoviště tím, že v nepřítomnosti majitele dedikovaného parkovacího stání uvolní aplikace konkrétní místo ostatním zaměstnancům. Všechna taková místa jsou zahrnuta v novém rezervačním systému, kde si každý zaměstnanec může o volné místo požádat. Krokem navíc je dále vylepšení pořadí, v jakém jsou místa řazena. Místo náhodného pořadí se přidělují podle toho, jak daleko od vchodu se fyzicky nachází. Nejbližší místa jsou nabízena prvním zájemcům a postupně se vzdalují. V případě, že uživatel zruší již existující rezervaci, jím uvolněné místo se opět zařadí na začátek seznamu.

Tímto se vylepší obsazenost celého parkoviště a prakticky i zvýší jeho užitná kapacita. Za optimalizovaný je potažmo možné považovat i proces zaplňování parkoviště, který je nově také řízen systémem. Oproti předchozímu stavu, kdy každý zaměstnanec zaparkoval na libovolném volném místě, nové řazení každého automaticky pošle na místo, které je aktuálně nejbližší ke vchodu.

Všechna zde popsaná vylepšení se v aktuální podobě týkají pouze manažerských stání, což vychází z domluvených požadavků zadavatele. Architektura systému ale umožňuje snadno rozšířit jejich působnost i na další parkovací místa, potažmo na celé parkoviště.

## Kapitola 5

### Analýza vstupních dat

Přechod na řešení, které umožňuje i řadovým zaměstnancům parkovat na manažerských místech, která jsou sama o sobě bonusem k vyšší pozici ve firmě, může být pro některé zaměstnance nepříjemný. Doteď platilo, že jim je jejich stání přístupné kdykoliv a na neomezenou dobu. Zároveň je nutné zajistit, aby zaměstnanci přistupovali k parkování na manažerských místech jako k nouzové výhodě a podle toho se chovali. Znamená to totiž, že v případě vytvoření rezervace bude nutné parkovat na správném místě a včas toto stání opět uvolnit. Tím se mitiguje riziko, že manažer nebude moci na svém místě zaparkovat, i když to bude potřebovat. Aby se při spuštění plánovaného systému předešlo nedorozuměním, byla vytvořena jednoduchá aplikace, která přiřazování manažerských stání zaměstnancům otestuje v praxi.

Tato aplikace zároveň sloužila jako ověření vstupních dat. Ta jsou uložena v podobě dvou oddělených tabulek Google Sheets, které jsou spravovány zadavatelským oddělením firmy. První obsahuje různé informace o parkovišti - převážně detaily parkovacích míst, jako je číslo stání, jeho účel, SPZ vozidla (pokud je místo vyhrazeno), identifikační číslo majitele. Druhá tabulka obsahuje seznam identifikačních čísel zaměstnanců, kteří mají na daný den schválenou absenci. Porovnáním tohoto seznamu a všech identifikačních čísel manažerů s přiřazeným parkovacím místem se vytvoří soupis parkovacích míst, která jsou neobsazená. Právě na ověření tohoto seznamu v praxi se zaměřila zmíněná testovací aplikace.

#### 5.1 Princip fungování

Základní logika testovací aplikace je jednoduchá. Z dostupných zdrojů se vytvoří seznam volných parkovacích míst na daný den a ten se vygeneruje v tabulce, kterou je možné vytisknout. Vytisknutý seznam míst dostane na starost pověřená osoba, v tomto případě vrátný, a ta má možnost parkovací stání po naplnění kapacity parkoviště nabízet příjezdějícím zaměstnancům. Ti potom na místě rovnou zaparkují a jejich jméno si pověřená osoba zapíše do zmíněného seznamu.

Toto triviální řešení naučí zaměstnance, že nyní existuje možnost dodatečného parkování v případě naplnění kapacity parkoviště a manažery, že jejich místa mohou být v době jejich nepřítomnosti využita někým jiným.

Zároveň je však správa přiřazování stání zaměstnancům ponechána člověku, který se fyzicky nachází na parkovišti a může flexibilně vygenerovaný seznam upravovat. Například vidí, že místo, které by mělo být prázdné, je přesto obsazeno, a může ho ze seznamu vyjmout. Navíc hlídá zaměstnance, aby parkovali na příslušných místech. Na druhou stranu se zde přidává práce jednomu zaměstnanci, který nemusí být ochotný přijíždějícím kolegům místa nabízet. Proto je vhodné nahradit přiřazování míst automatickým systémem.

## 5.2 Pozorované závěry

Běh testovací aplikace ukázal, že data sbíraná z docházkového systému jsou poměrně přesná, tedy že místa, která mají být prázdná, většinou skutečně prázdná jsou. Stále ale vznikaly v některých situacích problémy, že manažer s parkovacím místem přijel do práce, i když neměl. V zájmu plynulého chodu aplikace je nutné, aby se všichni zaměstnanci s novým chodem věcí ztotožnili, což znamená hlavně správné vyplňování docházkového systému manažery.

Změna vnitřního fungování firmy a zvyklostí manažerů ale není v silách nového systému. To závisí na jiných odděleních firmy (případně na vyšším managementu), která mají nad manažery a jejich docházkou kontrolu.

Pozorování nicméně ukázalo, že je obtížné zrušit již vytvořenou “rezervaci.” Tedy ve chvíli, kdy běžný zaměstnanec zaparkuje na propůjčeném stání a začne pracovat, už není jednoduché ho kontaktovat. K vytvořeným rezervacím se tedy bude přistupovat jako k finálním, nicméně kontaktní údaje o rezervující osobě budou i tak stále dostupné.

## 5.3 Shrnutí

Provoz zkušebního systému ukázal, že s dostupnými daty žádné zásadní problémy nejsou, ale že je nutná určitá zvykací fáze pro všechny uživatele rezervační aplikace. Zaměstnanci se musí chovat zodpovědně a neomezovat vlastníky míst nad dohodnutý rámec. Manažeři s přidělenými místy musí poctivě a přesně vyplňovat docházkový systém tak, aby se sami o místo nepřipravili a zároveň aby jej uvolnili ostatním v době osobní nepřítomnosti.



## Kapitola 6

### Požadavky



#### 6.1 Funkční požadavky

Podle vytvořeného návrhu byl domluven seznam funkčních požadavků, které aplikace musí splnit. Jejich výčet je uveden v tabulce 6.1.



#### 6.2 Nefunkční požadavky

Dále byl dohodnut i seznam nefunkčních požadavků, který udává pravidla, jak má aplikace plnit určité aspekty fungování. Ten je uveden v tabulkách 6.2 a 6.3.

Číslo a název	Detailní popis
FRQ01 - Vytvoření rezervace	Systém umožní zaměstnancům každý pracovní den vytvořit jednodenní rezervaci parkovacího místa.
FRQ02 - Informace o rezervaci	Systém umožní správcům získat aktuální informaci o tom, kdo parkuje na konkrétním místě.
FRQ03 - Automatické přihlášení	Systém umožní uživateli vytvářet rezervace bez nutnosti zadávat osobní údaje (emailová adresa a jméno) - ty se načtou automaticky.
FRQ04 - Zrušení rezervace	Systém umožní uživateli zrušit již vytvořenou rezervaci.
FRQ05 - Zobrazení na mapě	Systém umožní uživateli zobrazit si polohu přiděleného parkovacího místa na mapě parkoviště.
FRQ06 - Správa parkoviště	Systém umožní uživatelům se správcovským přístupem správu parkovacích míst.
FRQ07 - Využití dedikovaných míst	Systém nově umožní všem zaměstnancům parkovat i na parkovacích místech, která běžně patří konkrétní osobě.
FRQ08 - Zapnutí a vypnutí	Systém umožní správcům zapnout a vypnout rezervační systém.
FRQ09 - Úprava seznamu správců	Systém umožní správcům editaci seznamu osob se správcovským přístupem - přidání nových a odebrání existujících záznamů.
FRQ10 - Automatická aktualizace zdrojových dat	Systém umožní zaměstnancům upravovat již existující tabulky uchovávající informace o parkovišti. Nebude tedy nutné tyto změny duplikovat pro potřeby systému.

**Tabulka 6.1:** Seznam funkčních požadavků



Číslo a název	Detailní popis
NFRQ01 - Google Apps Script	Systém musí být implementován jako webová aplikace v prostředí Google Apps Script.
NFRQ02 - Mobilní rozložení	Systém musí být použitelný na mobilním zařízení.
NFRQ03 - Barevný návrh firmy	Systém musí být vizuálně stylizován podle barevného návrhu společnosti ve světlém odstínu zelené barvy.
NFRQ04 - Logo společnosti	Systém musí na všech stránkách obsahovat logo firmy.
NFRQ05 - Stabilita	Systém musí zvládnout obsloužit 100 souběžně přistupujících uživatelů bez náhlého ukončení.
NFRQ06 - Uživatelské testování	Systém bude uživatelsky otestován zadavatelským oddělením firmy.
NFRQ07 - Rychlost	Systém musí uživateli vytvořit rezervaci do 5 vteřin od stisknutí tlačítka.
NFRQ08 - Manuální spouštění	Systém bude spouštěn manuálně pověřeným zaměstnancem ve chvíli, kdy dojde k vyčerpání kapacity parkoviště.
NFRQ09 - Omezení na manažerská místa	Systém bude k rezervaci nabízet pouze (prázdná) manažerská místa.
NFRQ10 - Omezení na jeden den	Systém bude fungovat na bázi jednoho dne. Nebude tedy možné vytvářet rezervace předem, ani na konkrétní dobu.
NFRQ11 - Možnost odebrat místa z nabídky	Systém musí správcům umožnit správu seznamu míst nabízených k rezervaci. Každé místo může být ze seznamu odebráno natrvalo, nebo na jeden den.
NFRQ12 - Uživatelské rozhraní v angličtině	Systém musí být v anglickém jazyce.
Pokračování v tabulce 6.3	

**Tabulka 6.2:** Seznam nefunkčních požadavků - 1. část

Pokračování tabulky 6.2	
NFRQ13 - Zdrojová data	Systém bude získávat data z tabulek spravovaných jinými odděleními firmy. Data budou denně kontrolována pro případné změny.
NFRQ14 - Optimalizace přiřazování míst	Systém bude zaměstnancům parkovací místa přiřazovat postupně podle vzdálenosti od vchodu, počínaje nejbližším a konče nejvzdálenějším.
NFRQ15 - Dostupnost pouze pro vnitřní zaměstnance	Systém bude využíván pouze zaměstnanci firmy, nebude dostupný pro všechny zákazníky.
NFRQ16 - Optimalizace parkování	Systém bude parkovací místa přiřazovat podle jejich vzdálenosti od vchodu do budovy.

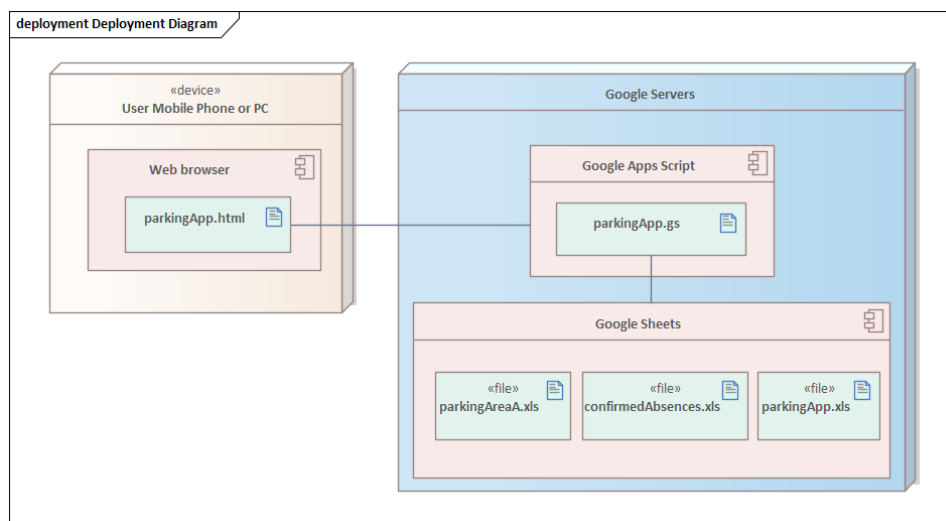
**Tabulka 6.3:** Seznam nefunkčních požadavků - 2. část

# Kapitola 7

## Implementace

### 7.1 Návrh aplikace

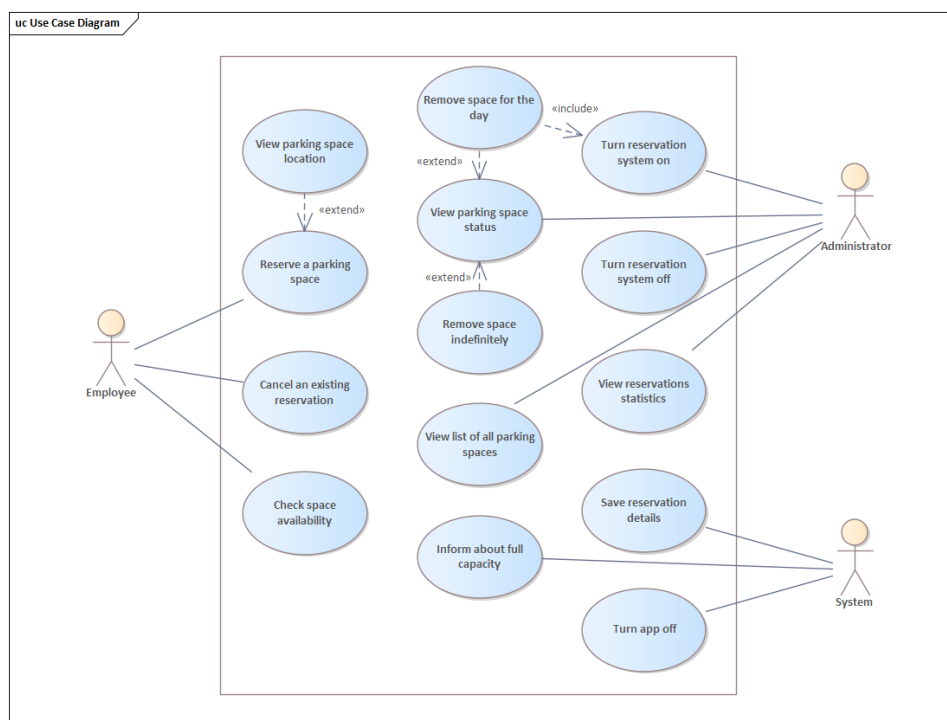
Aplikace byla dle požadavků navržena pro platformu Google Apps Script, konkrétně formou webové stránky. Detailní náhled architektury je k vidění v diagramu nasazení na obrázku 7.1. Jak z něj lze vyčíst, místo klasické databáze uchovává aplikace všechna data, se kterými pracuje, v tabulce Google Sheets. Tabulky "parkingAreaA.xls" a "confirmedAbsences.xls" naopak slouží jako zdroje dat.



Obrázek 7.1: Diagram nasazení

Diagram případů užití, anglicky Use Case diagram, viz obrázek 7.2, ukazuje základní funkcionality systému pro všechny zúčastněné strany. Z nich je zásadní zaměstnanec, pro kterého je celá aplikace určena. Ten ji využívá pro rezervaci parkovacího místa, na kterém následně může bezstarostně zaparkovat. Významnou součástí aplikace je i správcovská konzole. V ní může správce systému spravovat všechna dedikovaná místa. Mezi její základní prvky patří možnost odebrat místo ze seznamu nabízených.

Proces na obrázku 7.3 zobrazuje základní průchod aplikací ve chvíli, kdy



Obrázek 7.2: Diagram případů užití

se naplní kapacita zbytku parkoviště. Správce systému v tu chvíli webové prostředí spustí a zaměstnanci mohou začít žádat o parkovací místa.

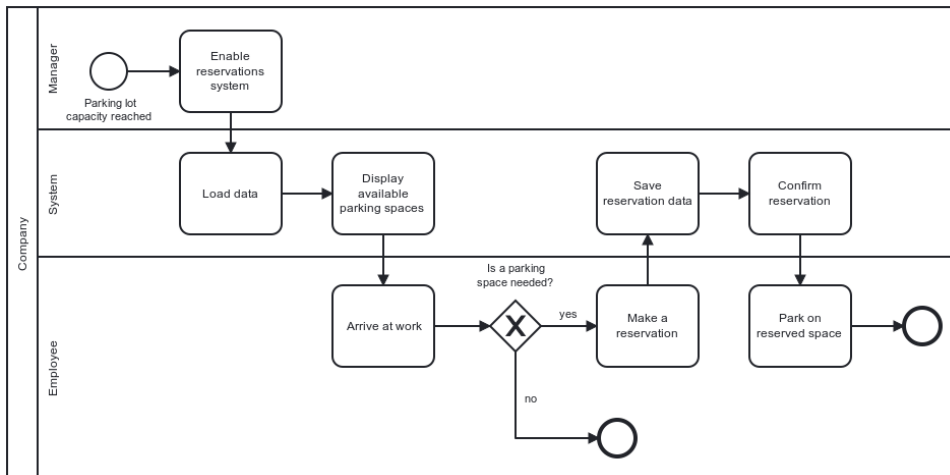
Stavový diagram na obrázku 7.5 zachycuje 4 základní stavy systému po nasazení:

- Not running (nespuštěný) - V tomto stavu je rezervační aplikace vypnutá a není možné vytvářet nové rezervace. Systém se zároveň automaticky vypne v daný čas po skončení pracovní doby.
- Running (spuštěný) - Přechodný stav po spuštění aplikace správcem, během kterého se načítají nová data ohledně parkovacích míst.
- Ready (připravený) - Systém je připravený na vytváření nových rezervací míst. V tomto stavu zůstává, dokud jsou stále volná místa k rezervaci.
- Full (plný) - Po vyčerpání všech dostupných míst už není možné vytvářet nové rezervace.

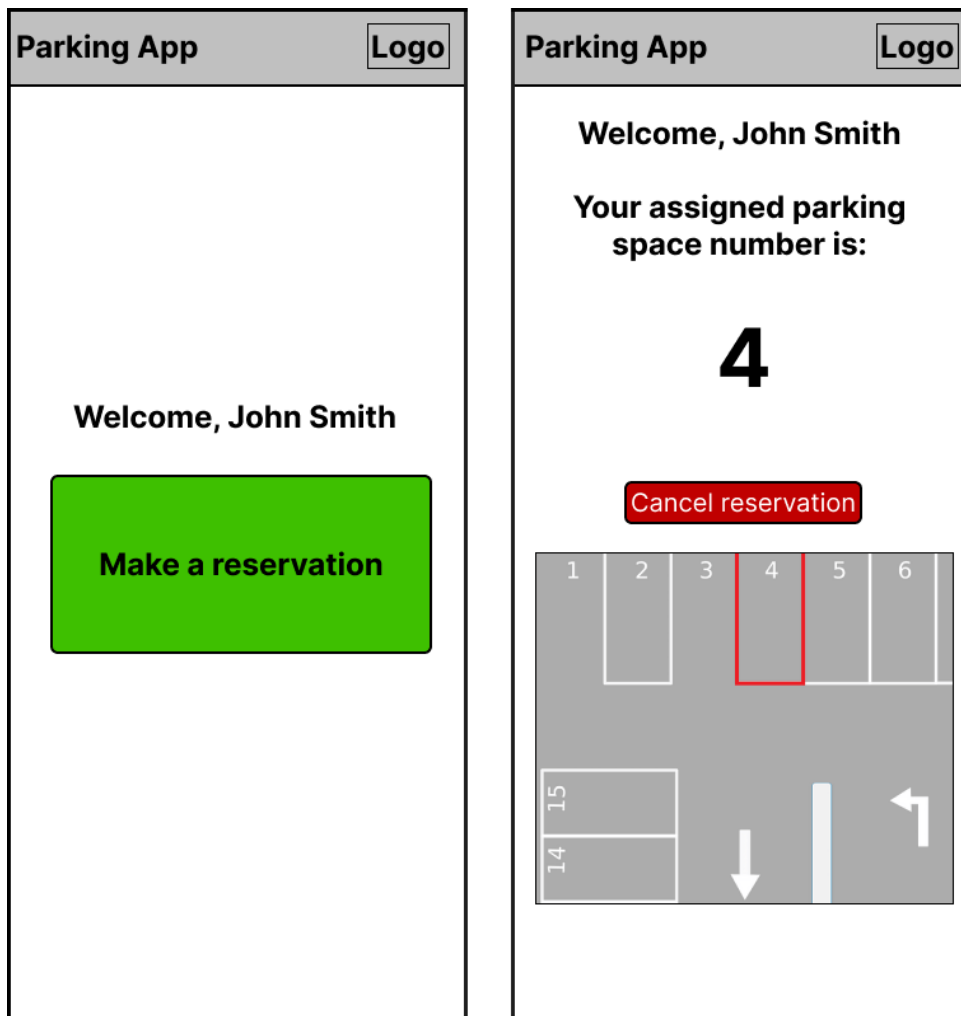
V rámci návrhu aplikace byl vytvořen i high fidelity prototyp [8] podoby mobilního rozložení, k vidění na obrázku 7.4.

## 7.2 Vývoj

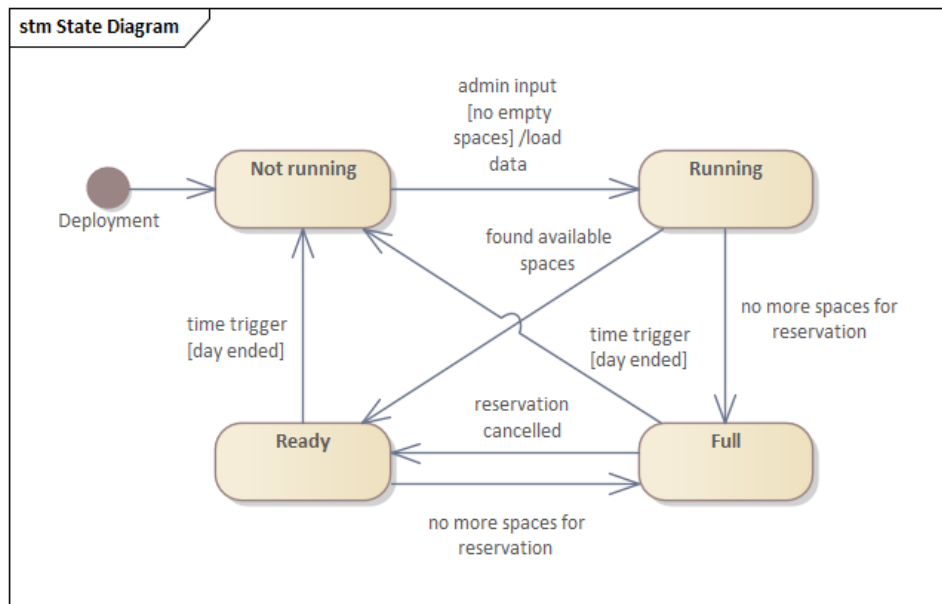
Proces vývoje samotné aplikace probíhal dle zásad metodiky zvané "waterfall", česky "vodopád". Ta spočívá v 5 fázích, viz. obrázek 7.6, které na sebe



Obrázek 7.3: Procesní diagram základního průchodu systémem

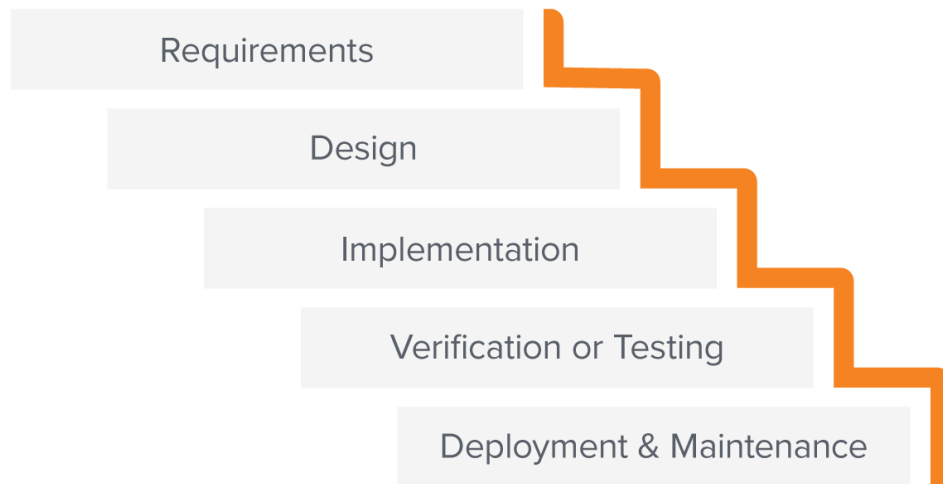


Obrázek 7.4: High fidelity prototyp



Obrázek 7.5: Stavový diagram

postupně navazují a společně vedou k vytvoření aplikace. Tento způsob sice oproti agilnímu vývoji neumožňuje reakci na změnu požadavků [9], ale pro tuto situaci je vhodný, jelikož téměř všechny požadavky byly definovány už na začátku projektu a neočekávaly se velké změny.



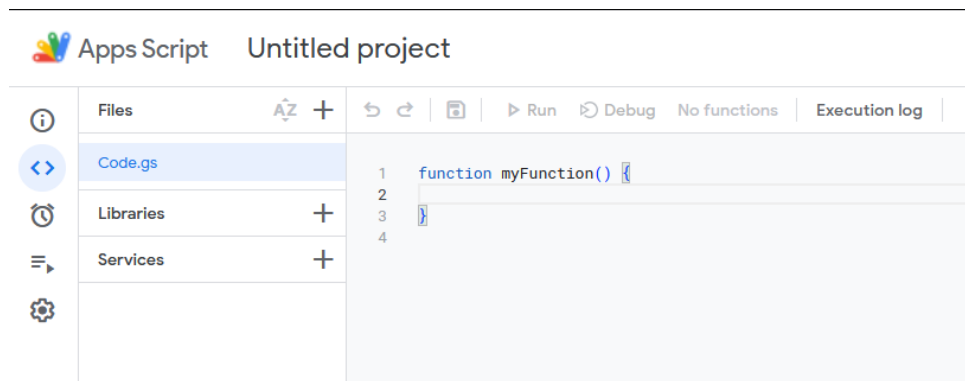
Obrázek 7.6: Fáze vodopádové metodiky[10]

## Kapitola 8

### Využitá technologie

#### 8.1 Google Apps Script

Google Apps Script, zkráceně GAS, je platforma pro tvorbu kódu a vývoj aplikací, zakomponovaná do prostředí Google Apps. Umožňuje přidání funkcionalit do tabulek, Gmailu, Sites a dalších služeb společnosti Google. Skripty mohou být vytvořeny jako samostatné soubory v úložišti Drive, uvnitř dokumentu nebo tabulky, případně jako Google Site[11]. GAS poskytuje možnost snadno a rychle vytvářet firemní aplikace, které se snadno propojí s Google Workspace. Vývoj probíhá v jazyce JavaScript, v editoru dostupném přímo v prohlížeči, viz. obrázek 8.1. Ten poskytuje přístup ke knihovnám umožňujícím práci s většinou aplikací[12].



**Obrázek 8.1:** Náhled na editor Google Script aplikací

V rámci projektu představuje Google Script vhodnou platformu pro aplikaci, protože zdrojová data jsou uložena v tabulkách Sheets. Nejvhodnější podobou je webová aplikace postavená nad jednou z těchto tabulek, která kopíruje data uložená v ostatních. Přizpůsobení aplikace pro použití ve webovém prohlížeči poskytuje několik výhod. První a nejdůležitější je uživatelský komfort. Díky kaskádovým stylům, tedy CSS, je možné nastylovat aplikaci tak, aby obsahovala velká tlačítka, která jsou snadno na dosah ruky při přístupu z mobilního telefonu. To by nebylo možné, pokud by se rezervace vytvářely například ručně přímo v tabulce. Druhou výhodou webu je právě

zmíněné odstínění uživatele od tabulky, díky čemuž není nutné, aby k ní měli přístup všichni potenciální uživatelé aplikace. V neposlední řadě je použití přes webovou stránku velmi jednoduché a pro uživatele intuitivní. Už jen proto, že není nutné stahovat žádné další programy.

## 8.2 Uchování dat

Aplikace vytvořené v GAS jsou navrženy tak, aby se daly využít zdroje uložené mimo jiné v tabulkách Google Sheets. Tím pádem není nutné v těchto skriptech využívat klasických databází. To nicméně přináší určitá omezení, která plynou ze složitějšího přístupu k datům. V základní podobě jsou hodnoty z tabulek získávány ve formátu dvourozměrných listů, kde jednotlivé záznamy prvního listu obsahují data z řádků, pozice (index) v každém z těchto řádků představuje sloupec [13]. Analogicky probíhá i uchování dat. Po zvolení konkrétního rozsahu buněk v tabulce, do níž chceme zapsat, jim lze přiřadit hodnoty z dvourozměrného listu.

Výhodou tohoto přístupu je jednoduchost implementace v aplikacích o menších rozměrech. S daty je totiž možné pracovat jako s běžným listem, který se následně snadno uloží. Problém nastává ve chvíli, kdy je nutné najít v tabulce konkrétní hodnotu nebo hodnoty. Protože nelze zadat vyhledávací parametry už během přístupu k uloženým informacím, je i z hlediska rychlosti nutné pracovat s celým rozsahem dat, ve kterém se následně vyhledává postupným procházením všech hodnot.

## 8.3 Struktura projektu

Obecně je webová aplikace v GAS v základu podobná klasickému webovému vývoji, má však svoje omezení. Základním rozdílem je, že je možné pracovat pouze se soubory typu .gs a .html. Soubory typu .gs obsahují funkce, které jsou vykonávány na serverové straně, tedy se pomocí těchto funkcí například přistupuje k datům z tabulek. Soubory .html obsahují zdrojový kód pro zobrazení webových stránek. K datům ze serveru se přistupuje speciálními metodami dostupnými přes JavaScript běžící na klientské straně. Ty jsou obsaženy v knihovnách přímo od Google. Skript může být publikován ve formě webové aplikace, pokud splní následující podmínky:[14]

- Obsahuje doGet(e) nebo doPost(e) funkci
- Ta vrací buď HTML service HtmlOutput objekt nebo Content service TextOutput objekt

HTML i Content service jsou příklady knihoven Google, které umožňují propojení klientské a serverové strany v Google Apps skriptu. HtmlOutput i TextOutput objekty obsahují kód ve formátu .html, který má být následně vyrenderován v prohlížeči. Je proto možné kontrolovat zadané podmínky v rámci funkce doGet() a podle jejich splnění či nesplnění zobrazovat konkrétní stránku. Obě metody se volají automaticky při přístupu na stránku.



V rámci tohoto projektu byl vytvořen jeden soubor `Code.gs`, který obsahuje všechny funkce potřebné pro přístup k datům a k jejich ukládání. Pro přehlednost a snadnou údržbu je využíván soubor `Constants.gs`, ve kterém jsou zaznamenány všechny důležité konstanty. Mezi ně patří identifikační kódy jednotlivých tabulek, názvy listů, sloupců a “souřadnice” konkrétních dat v příslušných tabulkách. Pro zajištění této možnosti je využito základní funkcionality editoru, která způsobuje, že všechny `.gs` soubory jsou mezi sebou propojené a mohou navzájem využívat metody a proměnné z ostatních.

Pro udržení základní struktury souborů webové aplikace má každá použitá “stránka” vlastní soubor ve formátu `html`. Aby se tyto soubory zmenšily, jsou všechna stylovací pravidla (která by běžně měla vlastní soubor nebo byla zapsána v HTML tagu “`<style>`”) pro každou stránku uložena v separátním souboru. Ten se jmenuje stejně jako odpovídající stránka, ale má přídomek “`CSS`.” Jeho obsah se sloučí se zbytkem během volání funkce `doGet()` a to pomocí funkce `include()`.

## 8.4 Omezení

GAS s sebou přináší určitá omezení, která se v běžné praxi nevyskytují. Příkladem je už fakt, že v souborové struktuře je možné pracovat pouze se soubory typu `.gs` a `.html`. To znemožňuje mimo jiné přidávání obrázků a dalších zdrojů přímo do struktury aplikace. Obrázky musí tudíž být nahrány přes veřejný odkaz, na konkrétním příkladu zde popisovaného systému to znamená práci navíc při zobrazování konkrétních obrázků se zvýrazněnou pozicí parkovacího místa. Každý obrázek je totiž nutné nahrát například na Google Drive, sdílet, a jeho ID vložit do odkazu, který obrázek na stránce správně zobrazí. Při zobrazování mapy ke konkrétnímu místu potom tedy nestačí zobrazit třeba soubor “`parkingspace12.jpg`,” kde se dá číslo snadno nahradit jiným, ale je nutné uložit si všechny odkazy a zpětně je dohledávat.

Další překážka vzniká při snaze přidat metadata do hlavičky `.html` stránky. Klasický postup, kde se jednotlivé značky jako například “`<title>`” nebo “`<meta>`” přidají do elementu “`<head>`,” zde selhává. Tyto informace je nutné přidat jako atributy k objektu `HtmlService`, který vrací funkce `doGet()`, jak je popsáno v sekci 8.3. Z běžného

```
<html>
  <head>
    <title>Example title</title>
    <meta name="viewport" content="width=device-width,
      initial-scale=1, minimum-scale=1" />
  </head>
</html>
```

tedy vznikne:

```
HtmlService.createTemplateFromFile("File").evaluate()
  .setTitle("Example title")
  .addMetaTag('viewport', 'width=device-width, initial-scale=1');
```

Poměrně zásadním problémem je fakt, že aplikace vytvořené v GAS neumožňují top-level navigaci nevyžádanou napřímo uživatelem. To v praxi znamená, že jakékoliv přesměrování může proběhnout jen a pouze po přímé interakci uživatele se stránkou. V kombinaci s vynucováním výlučně top-level navigace vzniká problém, že není možné automaticky znovu načíst stránku po dokončení běhu jiné funkce. V našem případě tedy není možné automaticky uživatele přesměrovat na stránku s potvrzením po úspěšném vytvoření rezervace parkovacího místa. Jedinou možností je nahrazení celé stránky novou pomocí klientského JavaScriptu. Tak je možné zajistit, že se nejprve ověří, zda-li rezervace byla vytvořena, a teprve potom je uživateli zobrazena stránka s potvrzením.

## 8.5 Shrnutí

Google Apps Script je platforma pro tvorbu business aplikací od společnosti Google, která pomocí knihoven umožňuje práci s ostatními aplikacemi, jako je Google Sheets nebo Google Docs. Zde je tato platforma využita pro vytvoření webové aplikace pracující s daty uloženými v tabulkách Sheets. Soubory typu .gs (Google Script) fungují na bázi skriptovacího jazyka JavaScript a představují serverovou stranu aplikace. Pomocí nich se mimo jiné přistupuje k datům uloženým v tabulkách. Ta jsou potom zobrazena na stránkách pomocí .html souborů.

## Kapitola 9

### Uživatelské rozhraní

#### 9.1 Rozdělení přístupu

Webová aplikace vytvořená v rámci tohoto projektu je rozdělena na dva typy uživatelů. Výchozí práva mají všichni uživatelé, kteří mají do aplikace přístup. Ti si mohou pouze vytvořit rezervaci, zobrazit číslo přiřazeného místa, jeho polohu na plánu parkoviště, případně mohou rezervaci také zrušit. Systém musí být pro tuto skupinu co nejjednodušeji přístupný, proto je omezen pouze na základní tlačítka a jasně srozumitelný text. Pro snadnou navigaci po stránkách jsou použity velké fonty, široké mezery mezi prvky a rozměrná navigační tlačítka.

Druhou skupinou uživatelů jsou správci. Správcem může být jakýkoliv zaměstnanec, jehož email je uveden v seznamu správců. Primárně se předpokládá, že jimi budou zaměstnanci personálního oddělení, kteří i dosud měli parkoviště a parkovací místa na starosti. Tito uživatelé mají základní funkcionalitu rozšířenou o možnost vstupu do správcovské konzole. To jim ale neodebírá schopnost vytvořit si vlastní rezervaci na parkování, tak jako běžný uživatel. Správcovský přístup obsahuje 3 hlavní prvky:

- Ovládání aplikace - zapnutí a vypnutí
- Základní přehled obsazenosti míst
- Seznam všech manažerských parkovacích stání

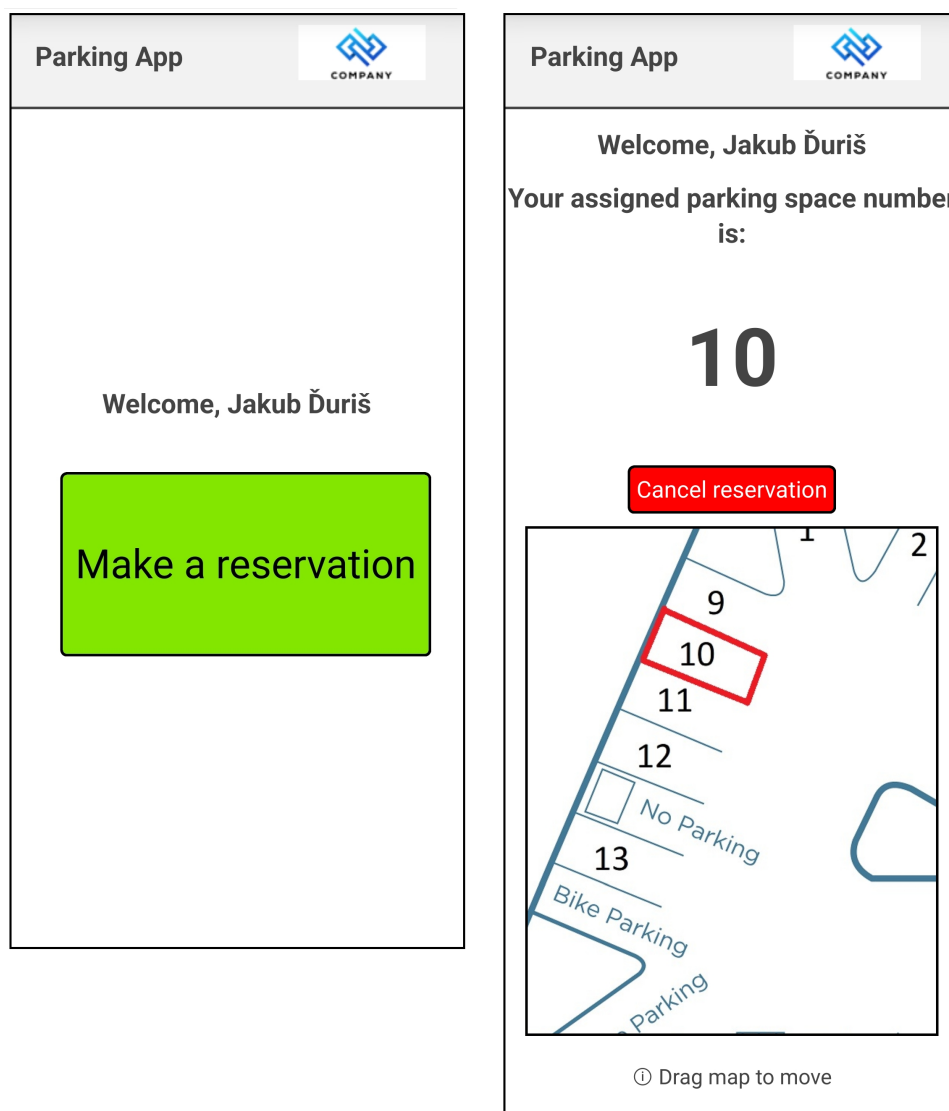
#### 9.2 Hlavní obrazovky

Pro běžného uživatele se za hlavní rozložení považuje mobilní, protože se předpokládá nejčastější přístup na stránku právě z telefonu. Správcovská konzole je zaměřena na počítačové rozložení, protože obsahuje velkou tabulku, která se pohodlněji spravuje právě na něm.

Samozřejmostí v každém rozložení jsou hlášky zpětné vazby, informující uživatele o tom, že interakce byla zaregistrována a probíhá zpracování dat. Při přístupu na každou stránku se nejprve zobrazí načítání, dokud nejsou všechny informace načteny. Správci mají v záhlaví každé stránky možnost

přepínat mezi běžným a správcovským zobrazením. Běžným uživatelům se toto tlačítko nezobrazuje.

### 9.2.1 Rozložení pro mobilní telefon

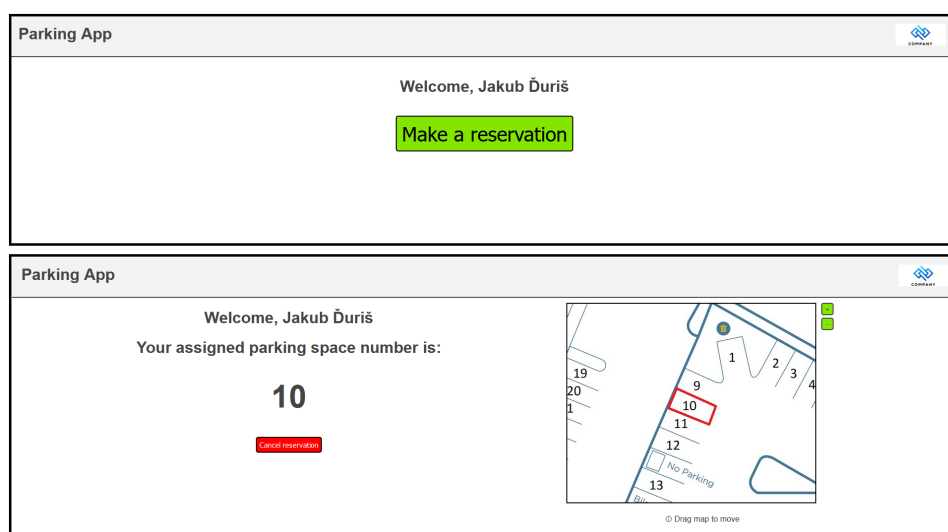


**Obrázek 9.1:** Rozložení základních obrazovek na mobilním zařízení

Rozložení pro mobilní telefon je koncipováno tak, aby bylo co nejjednodušší ke čtení i k ovládání. Hlavní stránka (viz. obrázek 9.1 vlevo) obsahuje velké tlačítko, které je na stránce navíc posunuto níže, aby se na něj dalo snadněji dosáhnout v případě, že je telefon držen v jedné ruce. Po jeho stisknutí je vytvořena rezervace a uživatel je přesměrován na stránku s potvrzením (viz. obrázek 9.1 vpravo). Tam je uvítán pozdravem se svým jménem, které má pasivně poskytovat zpětnou vazbu, že je vše v pořádku. Jméno uživatele se získává přes knihovnu přistupující k jeho seznamu kontaktů, ve kterém je

uložen jeho vlastní kontakt se jménem. Mimo jiné může sloužit jako kontrola, že je přihlášen správným účtem. Pod jménem se zobrazuje číslo systémem přiděleného parkovacího stání, tlačítko pro zrušení rezervace a obrázek s vyznačeným místem na plánu parkoviště. Parkovací místa jsou seřazena podle vzdálenosti od vchodu, to znamená, že nejprve jsou zarezervována místa nejbližší ke vchodu do budovy a pak postupně dále. Mapku je možné tažením posouvat, případně příslušnými gesty přiblížit a oddálit.

### 9.2.2 Rozložení pro počítač



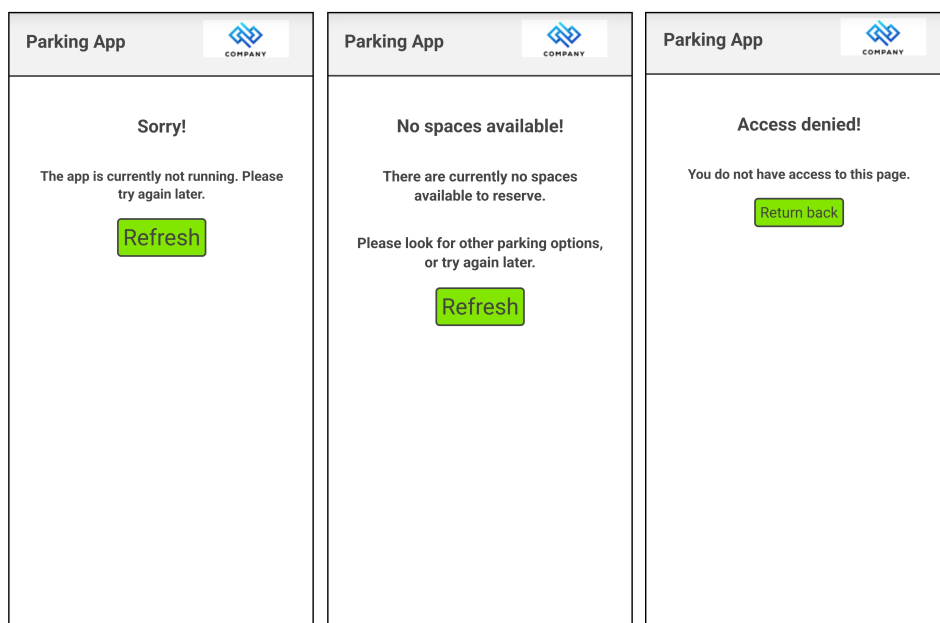
**Obrázek 9.2:** Rozložení základních obrazovek na počítači

Aplikace má funkční rozložení i pro počítač, viz. obrázek 9.2. To v podstatě vychází z rozložení pro mobilní telefon, ale snaží se více pracovat s prostorem do stran. Plánek parkoviště je přesunut doprava a objevila se tlačítka pro přiblížení a oddálení, která nahrazují gesta na mobilním telefonu.

### 9.2.3 Ostatní obrazovky

V případě nestandardní situace se mohou objevit varování, informující uživatele o stavu aplikace. Mezi ně patří:

- Neaktivní aplikace - Zobrazí se v případě, že aplikace daný den ještě nebyla správcem systému spuštěna. Uživatel je vyzván k trpělivosti. Na obrázku 9.3 vlevo.
- Plná kapacita - V případě, že se uživatel pokouší vytvořit rezervaci, nebo přichází na stránku ve chvíli, kdy jsou všechna dostupná místa zarezervována, je o této skutečnosti informován. Na obrázku 9.3 uprostřed.
- Přístup zamítnut - Pokud by se uživatel náhodou nebo záměrně dostal na odkaz vedoucí do správcovské konzole, proběhne kontrola, zdali se jeho



**Obrázek 9.3:** Obrazovky s různými druhy upozornění

emailová adresa nachází v seznamu správců. Pokud ne, je přesměrován na stránku s informací o neplatném přístupu. Na obrázku 9.3 vpravo.

#### 9.2.4 Správcovské zobrazení

Hlavním prvkem správcovské konzole je tabulka na obrázku 9.4, která zobrazuje informace o všech manažerských místech. Barevné zvýraznění označuje stav parkovacího stání během zapnuté aplikace. Modře podbarvená jsou ta místa, jejichž majitelé mají na daný den schválenou absenci a tato stání se tedy nabízí k rezervaci pro ostatní zaměstnance. Když je rezervace vytvořena, místo zezelená a objeví se emailová adresa zaměstnance, který rezervaci vytvořil.

Každé místo je možné modifikovat. Kliknutím na tlačítko “manage,” nebo dvojitým poklepáním na celý řádek, se zobrazí dialogové okno (viz. obrázek 9.5), které správci dává dvě možnosti:

- “Exclude space” - Zapnutí tohoto příznaku značí, že místo nebude nikdy nabízeno, ani během majitelovy nepřítomnosti. Nastavení zůstane vypnuté nebo zapnuté až do doby, kdy se manuálně přepne zpět. Text takto označeného místa se v tabulce zbarví šedě.
- “Available today” - Podobná funkcionalita jako “Exclude space,” ale pouze na jeden den. Zvolením možnosti “Yes” pro “Ano” nebo “No” pro “Ne” lze určit, jestli se místo daný den bude zobrazovat v nabídce k rezervaci. Spolu s vypnutím systému ale nastavení zmizí a příště se bude opět řídit daty z docházkového systému.

Nr.	Owner's Name	Excluded	Available today	Reserved by	Manage
1	Otakar Tyrol	No	No		Manage
2	Olga Procházková	No	No		Manage
3	Jarmila Satahřková	No	Yes		Manage
4	Jilka Hogná	No	Yes		Manage
5	Jaroslava Malá	Yes	No		Manage
6	Oldřich Coufal	No	Yes		Manage
7	Tomáš Vitek	No	Yes		Manage
8	Miroslav Šudoma	No	Yes		Manage
9	Lukáš Krutý	No	Yes		Manage
10	Eva Zúchová	No	Yes	durisp4@fd.cvut.cz	Manage
11	František Šíp	No	Yes		Manage
12	Zuzana Bílá	No	Yes		Manage
13	Václav Růžicka	No	No		Manage
14	Jana Josišová	No	Yes		Manage
15	Jan Vondráš	No	Yes		Manage
16	Julius Novák	No	Yes		Manage
17	Adéla Čížová	No	Yes		Manage

tabulka pokračuje

**Obrázek 9.4:** Správcovské zobrazení umožňující úpravu údajů o parkovacích místech

**Obrázek 9.5:** Dialogové okno s možnostmi editace parkovacího stání

Tyto informace jsou zároveň pro připomenutí uživateli zobrazeny po najetí myši na symbol otazníku vlevo nad tabulkou, viz. obrázek 9.6.

Konzole dále umožňuje zapnout a vypnout celou aplikaci, nicméně uživateli je silně doporučeno nevypínat běžící aplikaci před koncem dne. To proto, že vypnutí systému zruší všechny vytvořené rezervace, na což nejsou zaměstnanci nijak upozorněni. Aplikace se sama vypíná automaticky na konci dne. Viz. obrázek

## 9.3 Shrnutí

Přístup do systému se dělí na dvě úrovně - obyčejnou a správcovskou. Běžný uživatel aplikaci využívá k vytvoření rezervace parkovacího místa, často z mobilního telefonu. Rozložení je tedy zaměřeno na pohodlné použití v mobilních zařízeních. Při práci je uživateli poskytována zpětná vazba o provedených

**?** **Color coding**

**Nr.** **Blue** - Blue background color means the parking space is available for reservation today, but is not yet reserved.

1 **Green** - Green background color means a reservation was already made for this parking space. The occupant's email is also displayed in the table.

2

3 **Gray text** - When the text is gray, it means that the parking space is set to be excluded.

4 **Table columns**

5 **Excluded** - This column indicates if the parking space should automatically be offered for reservations in the owner's absence. This setting is persistent and will stay this way until manually changed back. When set to **Yes**,

6 the parking space will NOT be offered even during the owner's absence, as indicated in the cominfo.

7

8

9 **Available today** - This column indicates if the parking space is being offered today (the owner is absent). Can be manually toggled between **Yes** and **No**, but this

10 setting will be reset when the Parking App is turned off (or at the end of the day).

11

12

**Obrázek 9.6:** Nápořěda vysvětlující barevné návrhy správcořské tabulky

**Warning!**

This will irreversibly delete all reservations. Employees who made the reservations will NOT be notified about this.

The app does not have to be turned off manually, as it will happen automatically at the end of the day.

Are you sure you want to do this?

**Obrázek 9.7:** Upozornění varující uživatele před předčasným vypnutím aplikace



akcích v podobě informativních hlášek. Správci mají přístup jak k rezervaci míst, tak do správcovské konzole, která obsahuje seznam všech parkovacích a základní možnosti jejich spravování. Konzole je zaměřena na zobrazení na počítači, z důvodu rozměrů a nepřehlednosti na mobilním zařízení.



# Kapitola 10

## Testování

Stejně jako u tradičního softwaru, ověřovací a validační procesy mají za úkol kontrolu kvality implementace a odhalení deviací od požadavků uživatele. Dělí se na dva základní přístupy. Statické techniky se liší od dynamických tak, že nevyžadují “provedení” kódu (execution). Místo toho se zaměřují na artefakty, jako jsou návrhové dokumenty, například CSS, a zdroj HTML. Dynamické metody validace se zaměřují na prověřování systému zadáním vstupních dat a po proběhnutí porovnáním očekávaných výstupů se skutečnými[15].

Mezi konkrétní možnosti testování webových aplikací patří: [16]

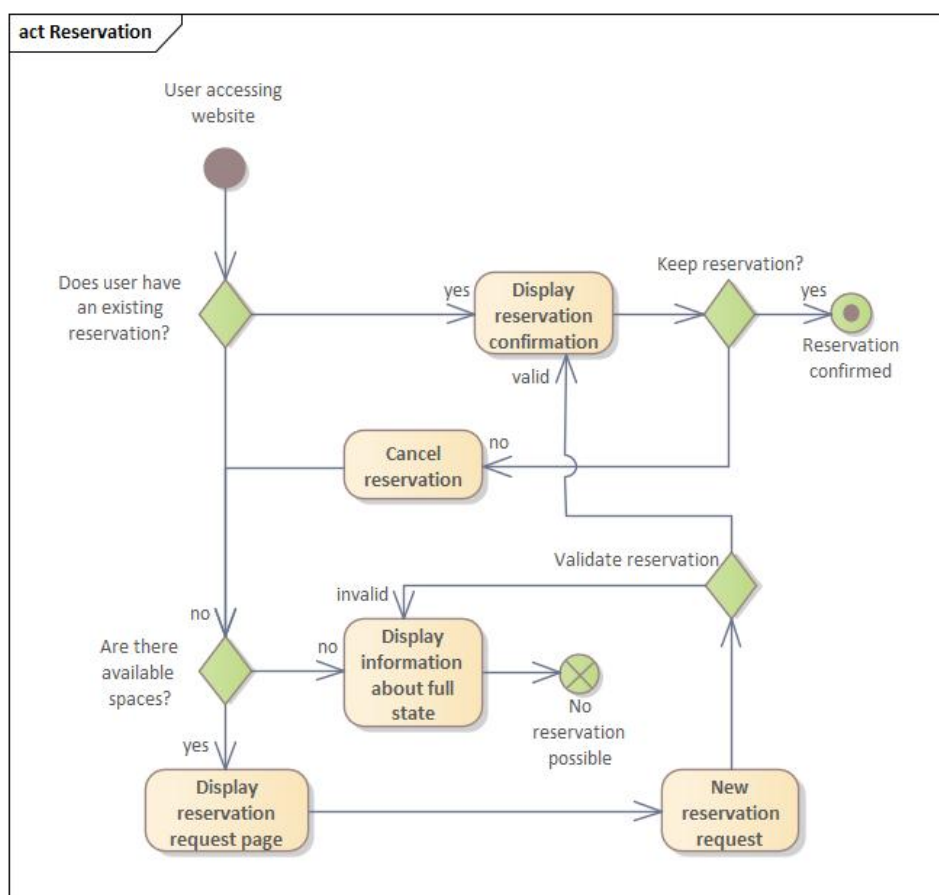
- Funkční testování (Functional Testing)
- Test použitelnosti (Usability Testing)
- Testování výkonu (Performance Testing)

Funkční testování se zaměřuje na ověření toho, že systém skutečně umožňuje všechny funkcionality, které vyžaduje uživatel a že tyto funkcionality fungují správně[17]. Testování použitelnosti se týká hodnocení produktu nebo služby jejich testováním s reprezentativními uživateli. Obvykle se během testu účastníci pokusí dokončit typické úkoly, zatímco pozorovatelé sledují, poslouchají a dělají si poznámky. Cílem je identifikovat případné problémy s použitelností, shromáždit kvalitativní a kvantitativní data a zjistit spokojenost účastníka s produktem[18]. Testování výkonu je proces zhodnocení toho, jak systém funguje z hlediska odezvy a stability při určité pracovní zátěži. Testy výkonu se obvykle provádějí za účelem prověření rychlosti, robustnosti a spolehlivosti aplikace[19].

V rámci tohoto projektu proběhlo několik testovacích strategií. Konkrétně byly zvoleny některé z důvodu vnitřních předpisů firmy, jiné proto, že dávají z hlediska webového vývoje největší smysl. Jednotlivé strategie jsou podrobně rozepsány v sekcích 10.1 až 10.6.

### 10.1 Testy průchodů

Nejdůležitějším procesem, který byl otestován, je vytvoření nové rezervace. Ten je pro fungování aplikace zásadní a základní. Jeho průchod je zobrazen v diagramu aktivit na obrázku 10.1.



Obrázek 10.1: Rezervace v diagramu aktivit

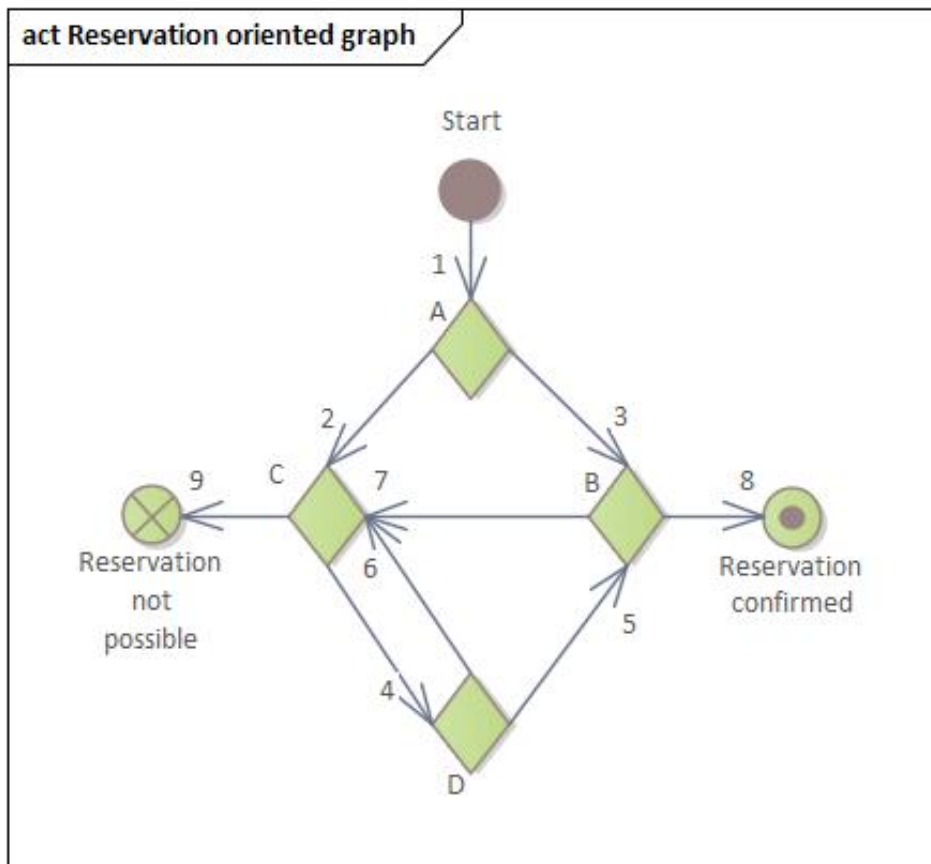
### 10.1.1 Testovací sekvence

Převedení průchodu procesu rezervace z obrázku 10.1 na orientovaný graf je vidět na obrázku 10.2.

Z něj byly vytvořeny následující testovací sekvence, viz. tabulka 10.1. Podle těchto plánů byly otestovány průchody tak, aby se ověřilo, že správně fungují.

## 10.2 Jednotkové testy

Jednotkové testy, anglicky “Unit Tests,” zahrnují testování softwarového kódu v jeho nejmenším funkčním bodě, tedy obvykle jedné třídy. Individuálním testováním každé jednotky lze většinu chyb, které mohou být zaneseny do kódu v průběhu projektu, velmi brzy odhalit a zabránit jejich šíření do dalších částí projektu[20]. Jednotkové testy spouští funkce, které testují, s konkrétními vstupními parametry a kontrolují, že funkce vrací očekávaný výstup. Například funkce sčítající dvě čísla by byla otestována tak, že by jí pomocí testu byla zadána dvě čísla, třeba 1 a 2 a očekávaný výsledek, tedy číslo 3. Pro každou funkci je třeba vytvořit takových testů více, aby se



Obrázek 10.2: Rezervace převedená na orientovaný graf

zkontrolovaly i krajní případy. Protože hlídat, že sčítací funkce vrátí správný výsledek při zadání dvou kladných nízkých čísel není příliš zajímavé. Problémy však už mohou nastat při snaze sečíst v naší funkci dvě nuly. Nebo dvě záporná čísla. Nebo záporné číslo a řetězec “abcd.” Je proto třeba ověřit nejen správné výsledky, ale i správné chování funkce v nestandardních situacích. To všechno jednotkové testy umožňují.

Pomocí těchto testů byly otestovány serverové funkce (ze souborů s příponou .gs, viz. sekce 8.3), které splňují dvě podmínky. Na vstupu přebírají jednu nebo více proměnných a vrací nějaký výstup. To jsou základní předpoklady pro použití jednotkových testů. Pro tento druh testování byla využita interní knihovna zadavatelské společnosti, která obsahuje všechny potřebné funkce ke spouštění vlastních konkrétních testů. Knihovna je soukromá v rámci firmy, proto ji nelze využít ve vzorové a anonymizované veřejné aplikaci. Více k tomuto tématu viz. kapitola 12

Pokrytí jednotkových testů dle různých aspektů je popsáno v tabulce 10.2. Relativně nízká úroveň pokrytí je způsobena tím, že většina funkcí nemá návratovou hodnotu, která by vycházela ze vstupních dat. Například pracují s daty v tabulce, tedy není nutné, by něco vracely. Nebo zpracují vstupní data a na výstupu posílají jiné hodnoty, které se využijí v dalších částech kódu.

Větvící bod	Vstupní akce	Výstupní akce	Kombinace pro test	Kombinace průchodu
A	1	2, 3	1-2, 1-3	1 2 4 5 8
B	3, 5	7, 8	3-7, 3-8, 5-7, 5-8	1 3 7 9
C	2, 6, 7	4, 9	2-4, 6-4, 7-4, 2-9, 6-9, 7-9	1 3 8
D	4	5, 6	4-5, 4-6	

Tabulka 10.1: Testovací sekvence

Popis	Poměr	Procentuálně
Počet otestovaných funkcí vůči celkovému počtu funkcí	5/35	14,29 %
Počet řádků kódu otestovaných funkcí vůči celkovému počtu řádků [21]	38/327 [22]	11,62 %

Tabulka 10.2: Pokrytí jednotkových testů

Takové funkce potom není možné otestovat jednotkovými testy. Počítání jednotlivých řádků kódu neukazuje zcela přesné pokrytí - některé funkce mohou být krátké, ale mít velký "účinek" - ale dává základní povědomí o tom, kolik funkcí bylo možné jednotkovými testy otestovat.

### 10.3 Tabulkové testy

Další knihovnou dostupnou v prostředí společnosti je knihovna tabulkových testů, Table Tests. Pomocí nich je možné ověřit, že zdrojové tabulky obsahují správná data. To přijde vhod v situacích, kde se data nahrávají z tabulek, které jsou zároveň jinými zaměstnanci firmy běžně k práci využívány. V takovém případě se snadno může stát, že správce tabulky náhodou nebo záměrně upraví buňky, které jsou aplikací využívány. To potom může způsobit problémy, zejména pokud se například číselná hodnota změní v jinou. Aplikace potom může zcela přestat fungovat. Je proto nutné zavést tyto kontroly zásadních hodnot. Knihovna zároveň nabízí možnost notifikace, pokud zachytí problém a tím umožňuje velmi rychlou nápravu chyby.

Zde je pomocí této knihovny kontrolována tabulka se seznamem zaměstnanců, jejich přidělených parkovacích míst a dalšími informacemi o parkovištích. Ta má však pro účely zavedení takovýchto kontrol omezení. Zejména v tom, že obsahuje i data navíc, která v aplikaci nejsou využívána a která mají jinou podobu. Nicméně však, protože jsou uložena ve stejné tabulce jako potřebná data, nedají se z důvodu způsobu implementace testovací knihovny z testů vyjmout. Tím jsou možnosti testování poměrně omezené, tyto testy proto byly využity alespoň pro taková data, která bylo možné zkontrolovat.

## 10.4 Výkonnostní testy

Přesto, že výkonnostní (performance) testy mohou být v řadě případů zanedbávány, jsou stále důležitou součástí vývoje a testování systémů[23]. Zde hraje výkon aplikace velkou roli, protože se předem očekává hlavní nápor uživatelů v danou chvíli, tedy ráno v pracovní dny. Aplikace byla proto otestována pomocí další interní firemní knihovny. Ta simulovala 200 přístupujících uživatelů naráz a jejich simultánní snahu o vytvoření rezervace na parkovací místo.

Z testu vyplynulo, že aplikace takový nápor vydržela bez problémů ohledně stability a rychlosti. Žádný z procesů, kterému se podařilo úspěšně vytvořit rezervaci, nezaznamenal problémy s dobou načítání a aplikace nespadla. Nicméně byl pozorován fenomén, že jednotlivé emailové adresy, se kterými se procesy připojovaly a které byly číselně seřazeny od 1 do 200, nebyly v seznamu rezervací uloženy v řadě. To je pravděpodobně zapříčiněno tím, že testovací knihovna neposílá jednotlivé procesy v řadě, tudíž se dopředu nedá zaručit, v jakém pořadí proběhnou. Každopádně i v případě, že by problém byl na straně aplikace, nebo Googlu, který serverové požadavky zpracovává, důležitějším výsledkem testu je, že se rezervace zapsaly a že aplikace nečekaně neskončila.

Výkonnostní testy byly v rámci projektu vybrány jako způsob, jak před nasazením odzkoušet chování aplikace pod náporem. Jejich cílem není otestovat rychlost serveru, ten je totiž spravován společností Google, ale zjistit, jak si aplikace poradí s mnoha žádostmi o rezervaci ve stejný moment. Konkrétně mají ověřit, jestli nebudou například již vytvořené rezervace přepsány jiným uživatelem, nebo v jakém pořadí budou přístupující uživatelé obslouženi. Konkrétní výsledky výkonnostního testování jsou uvedeny v tabulce 10.3.

## 10.5 Revize kódu

Po dokončení implementace aplikace proběhla v rámci firmy kontrola kódu seniorním zaměstnancem, neboli Code Review. Bylo zkontrolováno především dodržování interních požadavků na kód, jako jsou názvy proměnných, souborová struktura nebo dostatečná dokumentace a komentáře. Dále revizor poskytl návrhy a tipy na optimalizaci aplikace, které byly do systému vloženy.

## 10.6 Testování přijetí uživatelem

Finální verze byla představena zástupcům z oddělení, pro které je aplikace určena a které zadávalo požadavky na její podobu, aby proběhlo testování přijetí, neboli User Acceptance Testing. Aplikace byla předána k vyzkoušení, nicméně v době odevzdání této práce ještě není známa zpětná vazba. Ta je důležitá pro zakomponování pozorování od samotných uživatelů.

Pozorování	Výstup
Doba vytvoření rezervace	100 rezervací bylo vytvořeno během 20 sekund. Průměrná doba vytvoření rezervace je 0,2 sekundy. Jedná se však o čistou dobu bez uživatelského vstupu a grafického výstupu. Zpracování těchto dvou aspektů je primárně závislé na schopnostech zařízení koncového uživatele. Přesný přehled průměrné načítací doby pro uživatele by ukázal výstup z uživatelského testování.
Vytvoření rezervací	Všechny požadavky získaly rezervaci, pokud ještě zbylo volné místo. Později vytvořené požadavky nepřešly ty dřívější. V praxi by uživatel po stisknutí tlačítka "Rezervovat" obdržel chybovou hlášku o tom, že už nejsou volná místa. Během vytváření jeho rezervace byla totiž všechna zabrána uživateli, kteří žádost o rezervaci spustili o chvíli dříve.
Pořadí vytvořených rezervací	Jednotlivé rezervace byly vytvořeny v náhodném pořadí - pořadí, v jakém jednotlivé požadavky začaly, tedy nesouhlasí s tím, v jakém pořadí byla přiřazena parkovací místa. To má však v praktickém využití minimální dopad.

**Tabulka 10.3:** Výsledky výkonostního testování



# Kapitola 11

## Závěr

### 11.1 Vyhodnocení implementace

#### 11.1.1 Požadavky

Pro úspěšné dokončení bakalářské práce je nutné ověřit, že vytvořená aplikace splňuje všechny požadavky zadání bakalářské práce, v souladu s požadavky definovanými zadavatelem. Souhrnně lze říci, že byly splněny všechny, funkční i nefunkční. Testování ověřilo, že aplikace je schopná v jeden moment obsloužit 100 uživatelů, jak bylo definováno v NFRQ05. Podle NFRQ07 musí systém vytvořit rezervaci do 5 sekund. Bylo ověřeno, že rezervace je zapsána do příslušné tabulky v požadované době.

Nad rámec požadavků byl do správcovské konzole přidán jednoduchý štítek, který zobrazuje aktuální data o manažerských místech a jejich obsazenosti, viz. obrázek 9.4. Z nadefinovaných požadavků zadavatele na aplikaci není možné vytvořit jakýkoliv rozšířený systém statistik - například grafy zobrazující obsazenost parkoviště v jednotlivé dny v konkrétní hodinu, nebo přehled, která místa jsou nejvíce využívána. Je to proto, že aplikace má dle požadavků firmy omezenou funkčnost na jeden pracovní den, tedy nevytváří ani neukládá žádná data vytvořená ve dny předešlé. Dále jsou záznamy vedené pouze o manažerských místech, tudíž není možné získat informace o místech ostatních. Značnou optimalizaci aplikace přináší v podobě dynamického zpřístupnění parkovacích míst, která jsou fyzicky volná, ale dosud nemohla být využita, viz. odstavec 11.1.2.

#### 11.1.2 Optimalizace

Nasazení aplikace přináší všechny definované optimalizační výhody. Zejména nově umožňuje klientům firmy, kterými se dle specifikace rozumí její zaměstnanci, parkovat na místech, která pro ně dříve nebyla přístupná. Tím se zvětší množství volných parkovacích stání. Dále nyní aplikace spravuje řazení příjezdějících vozidel tak, aby se nejprve zaplnilo parkování nejbližší ke vchodu. Podle požadavků jsou tyto optimalizační prvky omezeny pouze na místa manažerská, ale nic nebrání rozšíření těchto pravidel i na ta zbylá.

### 11.1.3 Testování

Testování ověřilo, že aplikace je stabilní a vyhodnocuje data správně. Jednotkové testování odhalilo chyby v některých funkcích, které díky tomu mohly být opraveny. Velmi důležité bylo ověřit, že aplikace vydrží očekávaný nápor až nižších stovek uživatelů. Zde nebyly odhaleny žádné zásadní problémy, byť není zaručeno udržení pořadí přístupujících zaměstnanců. Podstatné ale je, že žádné rezervace nebyly po vytvoření přepsány jiným uživatelem a že se všechna dostupná místa naplnila.

## 11.2 Shrnutí výsledků

Navržená a vytvořená aplikace splňuje všechny body uvedené v zadání. Ve spolupráci se zadavatelskou firmou byl proveden návrh, podle něž byla následně aplikace implementována. Aplikace přináší optimalizaci parkování v podobě zpřístupnění míst manažerů ostatním zaměstnancům. Umožňuje i snadné rozšíření funkcionality na ostatní klienty firmy, nicméně dle požadavků není tato možnost použita. Firma je s výsledkem spokojena a systém využívá.

Smyslem celého systému je zefektivnění využití parkovacích míst před budovou zadavatelské společnosti. Parkoviště disponuje dvěma druhy míst, přičemž první část jsou místa nepřirazená, na kterých může parkovat kdokoliv, a druhá část jsou místa dedikovaná, která jsou přiřazená konkrétnímu zaměstnanci, nebo využívána za konkrétním účelem, například jsou na nich zaparkovány pracovní vozy. Toho bylo úspěšně docíleno.

Aplikace si klade za cíl umožnit zaměstnancům parkovat na těchto místech, na která běžně nemají přístup, v případě, že zrovna místa nejsou využívána. To aplikace zjišťuje ze seznamů spravovaných personálním oddělením - soupisem samotných míst a docházkovým listem. Jejich průnikem vzniká množina volných stání, která se mohou na daný den nabídnout k využití zaměstnancům přijíždějícím vlastním vozem. Součástí systému je návrh aplikace dle specifikací zadavatelského oddělení firmy.

Hlavním přínosem celého projektu je usnadnění procesu příjezdu do zaměstnání pro řadové zaměstnance, kteří ve firmě dlouhodobě trpí nedostatkem parkovacích míst. Možnost využít dedikovaná stání v případě jejich neobsazenosti je přínosná, zároveň vznik aplikace spravující parkovací místa otevírá další možnosti rozvoje a digitalizace v tomto směru, které by přinesly další výhody.

Systém byl implementován podle vytvořené specifikace a následně byl podroben několika různým druhům testování, které ověřily jeho robustnost a spolehlivost. Díky této kontrole je možné systém bezpečně nasadit do běžného provozu. Aplikace splňuje základní funkční požadavek, který upravuje zajištění systému parkovacích míst a umožňuje zaměstnancům využít volná stání, která pro ně doteď byla mimo dosah. Přidaná hodnota spočívá v zahájení možného rozvoje digitalizace celého parkoviště. Aktuálně aplikace pracuje výlučně s téměř 50 parkovacími místy, která jsou dedikována konkrétnímu uživateli. Celé parkoviště ale obsahuje přes 200 stání. Je tedy nyní možné zaměřit se

například i na tato místa a využít dodatečné systémy, například kamerový, k zjištění stavu obsazenosti a jeho následné prezentaci zaměstnancům. Systém proto vytváří prostor k aktualizaci běžného přístupu k parkování a k jeho rozvoji v digitálním světě.

## 11.3 Možnosti rozvoje

Aplikace má několik funkcí, které by v zájmu zvýšení uživatelského pohodlí bylo možné rozšířit. V první verzi byla upřednostněna jednodušší verze aplikace, která uživatelům umožní snazší seznámení s jejím používáním. Přidání funkcionalit firma plánuje na pozdější dobu, kdy už bude aplikace zavedena v běžném režimu.

Mezi možná rozšíření uživatelského prostředí patří možnost filtrovat výsledky ve správcovské tabulce podle jednotlivých sloupců - jméno, dostupnost, stav rezervovanosti. Také je možné rozšířit funkcionalitu o dlouhodobější přehled nebo export statistik obsazenosti a rezervovanosti všech míst.

Běžný uživatel by mohl mít zájem o zasílání informací o stavu jeho rezervace emailem. S tím by se dalo spojit i zaslání zprávy informující o zrušení rezervace správcem systému.

Velkým vylepšením by mohla být integrace celého systému s obrazem z bezpečnostních kamer umístěných v areálu. Z nich by se v daných časových intervalech načítaly obrázky, ze kterých by se automaticky zjistila aktuální obsazenost celého parkoviště, i s konkrétními parkovacími místy. Ta by se následně mohla automaticky zobrazit v systému jako dostupná k rezervaci. Tato idea je nejpravděpodobnějším směrem dalšího vývoje systému, aktuálně probíhá v rámci firmy analýza, jestli by takové řešení bylo proveditelné.

Tato i všechna další rozšíření nicméně přichází v úvahu až po zakomponování aplikace do běžného života zaměstnanců firmy. Všechny takové změny by také musely být konzultovány a schváleny se zákazníkem, tedy zadavatelským oddělením firmy.





## Kapitola 12

### Příloha

Činnost aplikace je možné ověřit v její anonymizované kopii, která byla vytvořena z důvodu uzavřeného systému zadavatelské firmy, na kterém originální aplikace běží. Kopie je veřejně dostupná [24] a věrně napodobuje originál. Z důvodu nezrušitelného automatického přesměrování ze strany společnosti Google je nemožné se do aplikace dostat, pokud je uživatel v prohlížeči přihlášen více než jedním účtem. Nejjednodušší způsob, jak v takovém případě do aplikace vstoupit, je přes "Anonymní režim." Tím se vyloučí nutnost odhlášení ze všech účtů Google v prohlížeči, což je samozřejmě další možnost. Přístup k originálu je možný po osobní dohodě s kontaktní osobou zadavatele: Ing. Jakub Hudec, jakub.hudec@valeo.com.





## Bibliografie

- [1] Carlton Reid. *Roads Were Not Built for Cars: How Cyclists were the First to Push for Good Roads & Became the Pioneers of Motoring*. Washington, DC: Island Press/Center for Resource Economics, 2015. ISBN: 978-1-61091-688-2. DOI: 10.5822/978-1-61091-688-2\_6.
- [2] Jakub Ďuriš. „Optimalizace využití parkovacích míst. Semestrální projekt“. Led. 2023.
- [3] Michael Maguire. *Google Sheets programming with Google Apps Script*. 2015.
- [4] *DATA INTEGRATION GLOSSARY*. Tech. zpr. 400 Seventh Street, S.W., HIAM-30, Washington, DC 20590: US Department of Transportation, srp. 2001, s. 10.
- [5] W3Schools. *C Multidimensional Arrays*. cit. 25. 3. 2023. URL: [https://www.w3schools.com/c/c\\_arrays\\_multi.php](https://www.w3schools.com/c/c_arrays_multi.php).
- [6] Mike Watson. *Managing Smaller Projects: A Practical Approach (Project Management Essentials Library)*. 2. Multi-Media Publications Inc., 2006, s. 117. ISBN: 9781895186857.
- [7] Lawrence Chung et al. „Non-Functional Requirements in Software Engineering“. In: Boston, MA: Springer US, 2000. ISBN: 978-1-4615-5269-7. DOI: 10.1007/978-1-4615-5269-7\_1. URL: [https://doi.org/10.1007/978-1-4615-5269-7\\_1](https://doi.org/10.1007/978-1-4615-5269-7_1).
- [8] Nick Babich. *Prototyping 101: The difference between low-fidelity and high-fidelity prototypes and when to use each*. cit. 7. 7. 2023. Lis. 2017. URL: <https://blog.adobe.com/en/publish/2017/11/29/prototyping-difference-low-fidelity-high-fidelity-prototypes-use>.
- [9] Adobe Communications Team. *Waterfall Methodology: A Complete Guide*. cit. 5. 7. 2023. Břez. 2022. URL: <https://business.adobe.com/blog/basics/waterfall>.
- [10] Adobe Communications Team. *Waterfall Methodology: A Complete Guide*. cit. 5. 7. 2023. Břez. 2022. URL: [https://business.adobe.com/blog/basics/media\\_1576e188070800895e8cf8e74a4a2582b9b20caaf.png?width=2000&format=webply&optimize=medium](https://business.adobe.com/blog/basics/media_1576e188070800895e8cf8e74a4a2582b9b20caaf.png?width=2000&format=webply&optimize=medium).

