

Bakalářská práce



České  
vysoké  
učení technické  
v Praze

**F3**

Fakulta elektrotechnická  
Katedra počítačů

## Chatbot jako turistický průvodce Prahou

Ruf Shamikh

Vedoucí: Ing. Jan Zídek  
Obor: Softwarové inženýrství a technologie  
Květen 2021



## I. OSOBNÍ A STUDIJNÍ ÚDAJE

Příjmení: **Shamikh** Jméno: **Ruf** Osobní číslo: **477994**  
Fakulta/ústav: **Fakulta elektrotechnická**  
Zadávající katedra/ústav: **Katedra počítačů**  
Studijní program: **Softwarové inženýrství a technologie**

## II. ÚDAJE K BAKALÁŘSKÉ PRÁCI

Název bakalářské práce:

**Chatbot jako turistický průvodce Prahou**

Název bakalářské práce anglicky:

**Chatbot as a tourist guide in Prague**

Pokyny pro vypracování:

1. Proveďte rešerši zdrojů dat o cestování a turismu.
2. Proveďte rešerši dostupných chatbotů a jejich využití v praxi.
3. Na základě rešerše implementujte chatbota, který pomůže s vyhledáním turistických cílů na základě komunikace s uživatelem.
4. Chatbot by měl například umět: vyhledat body zájmu v okolí, vyhledat informace o konkrétním místě, naplánovat trasu, nabídnout již ověřenou trasu, apod.
5. Implementujte uživatelsky přívětivé rozhraní s důrazem na využití aplikace na mobilních zařízeních.
6. Správnost řešení ověřte formou testování použitelnosti na skupině alespoň 5 účastníků.

Seznam doporučené literatury:

DREW, Martin, G. Woodside ARCH a Dehuang NING. Etic interpreting of naïve subjective personal introspections of tourism behavior: Analyzing visitors' stories about experiencing Mumbai, Seoul, Singapore, and Tokyo [online]. 2007 [cit. 2020-10-13]. ISSN 1750-6182. Dostupné z: doi:10.1108/17506180710729592

GLIOZZO, Dr. Alfio, Chris ACKERSON, Rajib BHATTACHARYA, et al. IBM. Building Cognitive Applications with IBM Watson Services: Volume 1 Getting Started [online]. I. vydání. online: IBM Redbooks, 2017 [cit. 2020-10-13]. ISBN SG24-8387-00. Dostupné z: <http://www.redbooks.ibm.com/abstracts/sg248387.html?Open>

AISHWARYA GUPTA. Introduction to AI Chatbots. International Journal of Engineering Research and [online]. 2020, V9(07) [cit. 2020-10-13]. ISSN 2278-0181. Dostupné z: doi:10.17577/IJERTV9IS070143

RASHCHUPKINA, Svetlana. ZERO WASTE WEBOVÁ APLIKACE. Praha, 2020. Dostupné také z: <https://dspace.cvut.cz/handle/10467/87585>. Bakalářská práce. České vysoké učení technické v Praze.

Jméno a pracoviště vedoucí(ho) bakalářské práce:

**Ing. Jan Zídek, Centrum znalostního managementu**

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

Datum zadání bakalářské práce: **12.02.2021**

Termín odevzdání bakalářské práce: **21.05.2021**

Platnost zadání bakalářské práce: **30.09.2022**

\_\_\_\_\_  
Ing. Jan Zídek  
podpis vedoucí(ho) práce

\_\_\_\_\_  
podpis vedoucí(ho) ústavu/katedry

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

### III. PŘEVZETÍ ZADÁNÍ

Studentka bere na vědomí, že je povinna 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 studentky

## Poděkování

Chtěla bych poděkovat svému vedoucímu Ing. Janu Zídkovi za pomoc, zpětnou vazbu a vedení této práce. Jsem vděčná Martině Klimešové za designové rady a taky kolegům Adamu Kováři a Adamu Lipowski za konzultaci ohledně JavaScriptového kódu. Dále chci poděkovat svým rodičům, přátelům, kolegům a mému příteli za podporu a za zajímavé nápady.

## Prohlášení

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

V Praze, 21. května 2021

Ruf Shamikh

## Abstrakt

Tato práce se zaměřuje na dvě témata, cestovní ruch a technologii chatbotů, a taky se skládá ze tří částí. Po vyhledání informací o těchto tématech vznikla první teoretická část této práce. Druhá část této práce je praktická a popisuje návrh a implementaci chatbota, který byl postaven na platformě od IBM Watson Assistant. Výsledkem práci je chatbot pro turisty v Praze, kde je možné pomocí komunikace vytvořit vlastní prohlídku, projít tuto prohlídku a také najít zajímavá místa v okolí. Poslední část je zaměřená na testování a hodnocení provedené práce.

**Klíčová slova:** cestování, prohlídka, cesta, chatbot, místo, IBM Watson Assistants

**Vedoucí:** Ing. Jan ZídekIng. Jan Zídek

## Abstract

This work focuses on two topics, tourism and chatbot technology, and also consists of three parts. The first theoretical part of this work was created after finding information on these topics. The second part of this work is practical and describes the design and implementation of a chatbot, which was built on a platform from IBM Watson Assistant. The result of the work is a chatbot for tourists in Prague, with which it is possible to create your own tour, go through it and also find interesting places in the area. The last part is focused on testing and evaluation of work.

**Keywords:** traveling, tour, path, chatbot, place, IBM Watson Assistants

**Title translation:** Chatbot as a tourist guide in Prague

# Obsah

<b>1 Úvod</b>	<b>1</b>		
1.1 Předmluva	1		
1.2 Motivace a cíl práce	1		
1.3 Struktura práce	2		
<b>Část I</b>			
<b>Teoretická část</b>			
<b>2 Cestování</b>	<b>5</b>		
2.1 Definice cestování	5		
2.2 Druhy cestování	5		
2.3 Cestovní ruch v Praze	6		
2.4 Organizace pěších výletů	6		
2.5 Shrnutí kapitoly	7		
<b>3 Chatbot</b>	<b>9</b>		
3.1 Definice chatbotů	9		
3.2 Historie chatbotů	9		
3.3 Klasifikace chatbotů	12		
3.3.1 Klasifikace chatbotů podle způsobu vývoje	12		
3.3.2 Klasifikace chatbotů podle obchodních cílů	12		
3.4 Výhody a nevýhody technologie chatbot	13		
3.4.1 Výhody	13		
3.4.2 Nevýhody	13		
3.5 Platformy pro tvorbu chatbotů	14		
3.5.1 Neprogramovací platformy	14		
3.5.2 Programovací platformy	14		
3.6 Cestovní chatboti	15		
3.6.1 Rezervační cestovní chatboti	15		
3.6.2 Chatboti o hledání míst a prohlídkách	16		
3.7 Shrnutí kapitoly	16		
<b>Část II</b>			
<b>Praktická část</b>			
<b>4 Návrh aplikace</b>	<b>21</b>		
4.1 Případy užití	21		
4.2 Výběr technologie	22		
4.2.1 IBM Watson Assistant	22		
4.2.2 Programovací jazyk	22		
4.2.3 Frontend	23		
4.2.4 Backend	23		
4.2.5 Databáze	23		
4.3 Výběr externích zdrojů	24		
4.3.1 Google Maps API	24		
4.3.2 Wiki API	25		
4.4 Architektura aplikace	25		
4.5 Návrh databáze	26		
4.6 Shrnutí kapitoly	26		
<b>5 Implementace</b>	<b>27</b>		
5.1 Vytvoření chatbota	27		
5.1.1 Dialog	27		
5.1.2 Intent	28		
5.1.3 Entity	29		
5.1.4 Kontextová proměnná	30		
5.1.5 Webhook	30		
5.1.6 Typy zpráv od chatbota	30		
5.1.7 Testování chatbota na platformě	31		
5.1.8 Průchod dialogem	31		
5.2 Frontend	33		
5.2.1 Tvorba designu	33		
5.2.2 Vývoj	34		
5.3 Backend	36		
5.4 Shrnutí kapitoly	38		
<b>Část III</b>			
<b>Testování a vyhodnocení implementované aplikace</b>			
<b>6 Testování chatbota</b>	<b>41</b>		
6.1 Cílová skupina	41		
6.2 Testovací scénáře	41		
6.2.1 První scénář	42		
6.2.2 Druhý scénář	42		
6.2.3 Třetí scénář	42		
6.2.4 Čtvrtý scénář	43		
6.3 Výsledky testování	43		
<b>7 Vyhodnocení</b>	<b>45</b>		
7.1 Výsledek práce	45		
7.2 Návrhy vylepšení stávající funkcionality	45		
7.3 Návrhy další možné funkcionality	46		

<b>8 Závěr</b>	<b>47</b>
<b>Přílohy</b>	
<b>A Literatura a zdroje</b>	<b>51</b>
<b>B Seznam použitých technologií, knihoven a nástrojů</b>	<b>57</b>
<b>C Instalace a spuštění</b>	<b>59</b>
C.1 Příprava prostředí . . . . .	59
C.1.1 Databáze MongoDB . . . . .	59
C.1.2 IBM Watson Assistant . . . . .	59
C.1.3 Soubor s požadovanými údaji	60
C.2 Spuštění aplikace . . . . .	60



## Obrázky

## Tabulky

3.1 Historie chatbotů .....	9
4.1 UML diagram případů užití ....	21
4.2 Návrh architektury .....	25
4.3 Návrh databáze .....	26
5.1 Dialog na platformě IBM Watson Assistant .....	28
5.2 Intents na platformě IBM Watson Assistant .....	29
5.3 Intents na platformě IBM Watson Assistant .....	30
5.4 Průchod dialogem .....	31
5.5 Design webové aplikace - hlavní část .....	33
5.6 Holešovice tour .....	34
5.7 Design webové aplikace - další části .....	34
5.8 Stránka s dialogem na webu ....	35
5.9 Vytváření session Id .....	37



# Kapitola 1

## Úvod

Tato kapitola popisuje témata, která tato práce pokrývá. Dále budou stanoveny motivace a cíle. Na konci stručně popíšu strukturu této práce.

Cílem této práce je implementovat turistického chatbota, který pomůže turistům poznat město Praha pomocí prohlídek. První část popisuje téma cestovního ruchu a technologii chatbotů. Následující kapitoly popisují návrh, implementaci a testování chatbota. Výsledkem této práce je aplikace, pomocí které může turista vyhledávat zajímavá místa v okolí, vytvořit prohlídku nebo zvolit hotovou prohlídku a poté jí projít pomocí chatbota.

### 1.1 Předmluva

Hlavními tématy, kterými se tato práce zabývá, jsou cestovní ruch a chatboti. Cestování je skvělou příležitostí jak poznat kulturu a vlastnosti jiné země nebo regionu. Turisté mohou mít různé cestovní cíle: nakupování, zábavu, práci, jídlo a mnoho dalších věcí. Ale cílem pro většinu z nich je procházka městem při hledání historických nebo nových struktur. Mohou se seznámit se slavnými památkami, navštívit muzea, galerie a parky toho města, ve kterém se nachází.

Technologie chatbotů je v současnosti velmi populární a často se používá v turistickém průmyslu. Pomáhají turistům s rezervací hotelů, letenek, dávají rady, atd. Turisté pro poznávání města obvykle volí samostatné procházky nebo prohlídky s průvodcem. Rozhodla jsem se implementovat chatbota, který by byl jejich průvodcem po městě, a také by pomohl vytvořit osobní prohlídky.

### 1.2 Motivace a cíl práce

Moje hlavní motivace je zavedení chatbota do prostředí cestování s cílem pomoci návštěvníkům se seznámit s městem. Výhodou chatbota je konverzace s uživatelem pomocí dialogu. Taková interakce s chatbotem je pro velké množství lidí vhodnější, než čtení článku nebo knihy, a hledání informace z různých zdrojů.

Tato práce má dva cíle. Prvním cílem této práce je teoretické seznámení s tématem cestování, technologií chatbotů, jakož i analýza využití chatbotů v oblasti cestování. Dalším cílem je vytvořit aplikaci, kde může uživatel pomocí komunikace s chatbotem vytvářet prohlídky, procházet připravené prohlídky nebo hledat místa v okolí.

## 1.3 Struktura práce

Práce obsahuje 8 kapitol, které zahrnují teoretickou a praktickou část. Tato kapitola popisuje úvod, který dává člověku pochopení toho, o čem bude hlavními tématy v této práci.

Následující dvě kapitoly jsou teoretická část, kde dělám rešerše informací a seznamuji se s tématy. Druhá kapitola je věnována cestování, druhům cestování, turistice v Praze a organizaci turistických výletů. Třetí kapitola popisuje definici chatbota, historii chatbotů, jejich klasifikaci, silné a slabé stránky této technologie a také platformy, pomocí kterých lze chatboty vytvářet.

Další tři kapitoly jsou praktické. Čtvrtá kapitola popisuje plán vývoje aplikace, kde jsou uvedeny případy užití a které technologie jsou vybrány pro vývoj. Pátá kapitola popisuje proces vývoje aplikace, tedy jak byl vyvinut chatbot, frontend a backend. Šestá kapitola se zaměřuje na testování, kde je možné najít testovací scénáře a výsledky testování.

Dále je kapitola o vyhodnocení práce. Popisují tam co by šlo vylepšit na základě provedeného testování. Rovněž budou uvedeny nápady, jak by bylo možné tuto aplikaci rozšířit.



# Část I

## Teoretická část



## Kapitola 2

### Cestování

Tato kapitola se zabývá tématem cestování. Popíšu definici cestovního ruchu a druhy cestování. Budu se zabývat turistikou v Praze za účelem zjištění oblíbeného typu cestovního ruchu mezi turisty v tomto městě. V této kapitole také identifikuji způsoby, jak organizovat pěší túry.

#### 2.1 Definice cestování

Existuje mnoho definic cestovního ruchu. Následující definice je volný překlad ze zdroje [1]:

"Cestovní ruch je pohyb osob z místa jejich trvalého pobytu do jiné země nebo lokality v jejich zemi ve volném čase pro zábavu a rekreaci."

Turismus se stal dostupnějším a oblíbenějším po rozvoji dopravy, rozšířilo se cestování a vznikly cestovní agentury. Dříve si to mohli dovolit jen bohatí lidé, ale po průmyslové revoluci si každý mohl udělat krátký výlet. Než začala pandemie Covid-19 cestovní ruch tvořil 12% celkového světového obchodu a toto číslo rostlo každý rok o 8%. [1]

#### 2.2 Druhy cestování

Jedna z hlavních klasifikací cestovního ruchu závisí na potřebách cestovatelů. Vzhledem k rychlému rozvoji této oblasti existuje mnoho druhů cestování. Dále je uvedeno pouze několik hlavních pro představu čtenáře. Tato sekce je kompilací informací ze zdrojů: [1, 2, 3, 4].

**Rekreační turistika** se provádí za účelem zotavení, rekreace a zábavy. Cestovatelé toho dosáhnou změnou prostředí.

**Lékařská turistika** se provádí za účelem zlepšení zdraví, léčby nemocí a chirurgických operací. To se děje mimo trvalé bydliště. Taková turistika se obvykle provádí kvůli tomu, že v zemi bydliště je nezbytná léčba na nižší úrovni nebo se neprovádí vůbec.

**Náboženská turistika** je individuální nebo skupinový výlet na svatá místa. Důvody, proč se lidé rozhodnou pro takové cestování, mohou být životní krize, duchovní spása, oslavení náboženských událostí a společenství s jinými věřícími.

**Dobrodružná turistika** je druh aktivního cestovního ruchu, který může zahrnovat určitá rizika. Aby se člověk mohl tomuto typu turistiky věnovat, musí mít dobrou fyzickou zdatnost a speciální dovednosti. Cestovatelé se snaží vystoupit ze své komfortní zóny.

**Kulturní turistika** je návštěvou historických, kulturních nebo geografických zajímavostí. Cestovatel se zapojuje do kultury země, seznamuje se s tradicemi obyvatel a jejich způsobem života. Tento druh turistiky je mezi cestovateli nejoblíbenější.

Abych mohla určit, na jaký typ cestovního ruchu bych se měla zaměřit, musím zvolit aktivní druh cestovního ruchu ve městě, který bude spojen s chatbotem. Proto se v další kapitole budu věnovat tématu cestovního ruchu v Praze.

## 2.3 Cestovní ruch v Praze

Tato sekce je kompilací informací ze zdrojů: [5, 6, 7].

Po vstupu České republiky do Evropské unie v roce 2004 se odvětví cestovního ruchu začalo aktivně rozvíjet. V roce 2018 navštívilo Prahu 7,9 milionu lidí, z toho 6,7 milionu (85%) byli zahraniční turisté. Počet turistů každým rokem roste.

Tak velký počet turistů vyžaduje ve městě vývoj různých směrů, které lákají turisty. Hlavní typy cestovního ruchu, o které se hosté města zajímají, jsou takové druhy turistiky: rekreační, dobrodružná, přírodní, kulturní, průmyslová.

Návštěva historických památek a kultury je motivací pro 41% zahraničních turistů. Historické centrum Prahy je považováno za největší městskou rezervaci na světě. Turisté v Praze se na památky většinou dívají zvenčí díky zachovalé historické architektuře. Hlavním úkolem turistického chatbota v rámci mé bakalářské práce bude poskytnout pěší túry návštěvníkům města. K tomu je třeba určit způsoby organizace turistických pěších výletů, které budou popsány v následující kapitole.

## 2.4 Organizace pěších výletů

Procházky po městě jsou nedílnou součástí městského cestovního ruchu. Existují dva způsoby organizace pěších výletů: je to možnost využít služeb průvodce nebo samostatné plánování výletu. Tato sekce je kompilací informací ze zdrojů: [8, 9, 10, 11, 12].

Turistický průvodce turisty doprovází, ukazuje jim památky a zajímavá místa. Mezi hlavní dovednosti turistického průvodce patří vypracování a realizace plánu turistických tras. Všechno by mělo být jasně promyšlené a organizované. Zaměstnávání turistických průvodců lze rozdělit do tří oblastí: průvodce po historických památkách, podnikový průvodce a průvodce přírodou.



Historičtí průvodci seznamují turisty s památkami, muzei, galeriemi, atd. Jedním z jejich úkolů je vyprávět zajímavý příběh místa nebo památníku, který ukazují.

Firemní průvodci obvykle podnikají prohlídky továren, vyprávějí příběh podniku, popisují výrobní proces a cíle podnikání. Mohou také provádět výlety do muzeí a galerií.

Průvodci přírodou provádějí výlety s důrazem na přírodní zdroje a pozoruhodná místa. Jedná se obvykle o procházky do přírodních rezervací, národních parků a hor.

Existují průvodci, kteří stanoví pevnou cenu za prohlídku, a jiní, kteří mají otevřenou cenu výletu. To znamená, že sám turista určuje, kolik chce průvodci zaplatit. Bohužel kvůli situaci s koronavirem mnoho turistických průvodců neprovádí prohlídky.

Při samostatném plánování prohlídek města s cílem poznat nová místa se turista bude muset hodně snažit. Pro kvalitní a zajímavou procházku po městě musí turista trávit spoustu času hledáním informací o památkách a dalších kulturních místech města. Může používat zdroje jako je internet, knihy, číst mnoho článků, fóra, hledat pomoc od přátel nebo sledovat videoblogy. Také na internetu existují nezávislé pěší túry, které popisují samotnou cestu, strávený čas, vzdálenost cesty, body na mapě. Tato metoda je ekonomičtější, ale není tak zajímavá. Některé klíčové body budou pravděpodobně vynechány.

Turisté si obvykle mohou dovolit využít služeb turistických průvodců, ale turisté s omezeným rozpočtem dávají přednost navrhování vlastních cest. Při výběru jedné z metod je věnována pozornost nejen rozpočtu, ale také času strávenému ve městě, zájmům jednoho nebo skupiny turistů.

## ■ 2.5 Shrnutí kapitoly

V této kapitole jsem analyzovala pojem slova „cestování“ a definovala jeho typy. Po prozkoumání tématu cestovního ruchu v Praze jsem se rozhodla zaměřit na kulturní cestovní ruch. Analyzovala jsem téma organizování pěších výletů s cílem určit druh, který by vyhovoval jak bohatým, tak ekonomickým turistům.



## Kapitola 3

### Chatbot

Účelem této kapitoly je seznámit se s definicí slova „chatbot“, dozvědět se více o této technologii pomocí historie chatbota a jeho klasifikací. Bude důležité analyzovat výhody a nevýhody této technologie a dozvědět se více o platformách, na kterých můžete chatbota vytvořit. Na konci si povíme o existujících cestovních chatbotech.

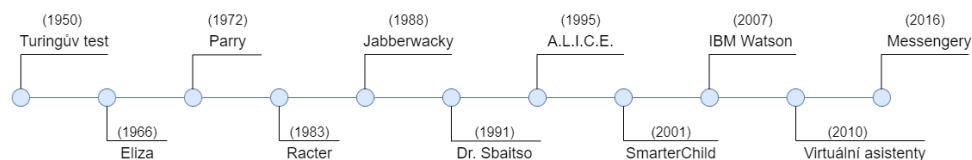
#### 3.1 Definice chatbotů

Termín „Chatbot“ původně vytvořil vědec Michael Mauldin v roce 1994. To znamenalo interaktivní program, který mohl simulovat lidskou komunikaci. [13] S rozvojem této technologie a její implementací v různých sférách života zní v dnešní době definice slova „chatbot“ takto:

"Chatbot je program s umělou inteligencí, který je schopen simulovat komunikaci s uživatelem pomocí předem vypočítaných klíčových frází, zvukových nebo textových signálů." [14]

#### 3.2 Historie chatbotů

Chatboti mají dlouhou historii vývoje. Vyvinuli se z jednoduchého textového chatu do pokročilé interaktivní komunikační platformy. Tato část popisuje, jaký byl jejich vývoj. Na obrázku 3.1 je vidět časová osa vývoje.



Obrázek 3.1: Historie chatbotů

#### Turingův test.

Historie chatbotů začala právě tímto testem, který vytvořil v roce 1950 slavný logik, kryptoanalytik a objevitel počítačové vědy Alan Turing. Ve



### ■ Dr. Sbaitso.

Velmi důležitý chatbot zvaný Dr. Sbaitso byl vytvořen v roce 1991 společností Creative Labs. K vytvoření chatbota byla použita zvuková karta Sound Blaster. Tento přístup byl revoluční technologickou inovací, protože dokázal syntetizovat řeč. Nereprodukoval složitou komunikaci, byl v roli psychologa, který kladl jednoduché otázky. [15, 16].

### ■ A.L.I.C.E.

Vědec Richard Wellis v roce 1995 vytvořil prvního chatbota založeného na AIML<sup>1</sup>, který dostal název ALICE, což je zkratka pro Artificial Linguistic Internet Computer Entity. Jazyk AIML pracuje se schématy XML, to znamená, že definuje parametry dialogu. Díky tomuto jazyku mohl chatbot vést více lidských rozhovorů a poskytovat složitější odpovědi. ALICE však nemohla projít Turingovým testem. Poté, co Richard Wellis v roce 2001 zveřejnil specifikaci AIML, mnoho vývojářů vytvořilo open source ALICE v různých programovacích jazycích a chatbot zvládl nové cizí jazyky. [15, 16].

### ■ SmarterChild.

První rozšířený chatbot SmarterChild se objevil v roce 2001. Byl vytvořen ve společnosti ActiveBuddy. Chatbot byl používán na platformách pro zasílání zpráv, které byly v té době populární. Používalo jej více než 30 milionů uživatelů. Díky použití online databází a získávání aktuálních informací přineslo konečnému uživateli skutečné výhody. [21, 22].

### ■ IBM Watson Assistant.

Watson chatbot od IBM byl vytvořen v roce 2007. Hlavním úkolem bylo bojovat ve hře „Jeopardy!“. Hráč v této hře musí rychle pochopit složité otázky, které mohou obsahovat složité formulace, dvojznačnost, hříčky atd. Velkou roli hraje také porozumění přirozenému jazyku. V roce 2011 porazil dva z nejlepších hráčů. [16, 23].

### ■ Virtuální asistenty.

V roce 2010 společnost Apple vydala Siri, která byla prvním široce dostupným osobním digitálním asistentem. Od té doby Google, Amazon a Microsoft vydaly své vlastní modely takových asistentů. Tito chatboti dobře rozumějí přirozenému jazyku, a protože mají přístup ke kontaktům, kalendáři a dalším datům, mohou jménem uživatele provádět mnoho různých úkolů. [16, 21].

---

<sup>1</sup>Artificial Intelligence Markup Language



Takoví chatboti musí rozumět kontextu, aby mohli poskytovat kvalitní služby zákazníkům. S tím se pojí další vlastnosti jako je individualita a vícečetná komunikace. Takoví chatboti se nacházejí v digitálním marketingu, maloobchodu nebo třeba zdravotnictví.

**Chatboti s dovednostmi** provádějí příkazy zadávané uživatelem, takže nemusí rozumět kontextu. Nejpohodlnější formou komunikace s takovým chatbotem je hlasová komunikace, takže uživatel nemusí používat žádné zařízení. Chatbot by měl fungovat rychle, aby uživatel nebyl rozptylován od hlavních činností.

**Chatboti asistenti** si vzali to nejlepší ze dvou předchozích typů. Díky tomu jsou zlatou střední cestou. Tito chatboti by měli pokrývat širokou škálu témat, aby mohli s uživatelem o všem mluvit. Individualita chatbota by měla být zajímavá a zábavná, aby uživatel chtěl s chatbotem komunikovat znovu a znovu.

Tyto typy jsou základní, ale v praxi existuje mnoho různých kombinací těchto typů chatbotů. Analýza těchto kombinací není obsahem této práce. Dále se zaměřuji na typ chatboti asistenti.

## 3.4 Výhody a nevýhody technologie chatbot

Vývoj technologie chatbot odhalil určité výhody jejího používání v životě, ale také identifikoval oblasti, které je třeba zlepšit. Tato sekce popisuje výhody a nevýhody používání technologie chatbot. Následující podsekce jsou kompilací informací ze zdrojů: [29, 30].

### 3.4.1 Výhody

Díky dostupnosti 24 hodin denně je technologie chatbot pro společnost výhodnějším řešením než zaměstnanec. Pokrytí takové podpory lidmi je mnohem obtížnější. Vzhledem k tomu, že chatboti přijímají požadavky kdykoli během dne, klient obdrží odpověď okamžitě, bez ohledu na denní dobu. To pomáhá udržet zájem klienta o společnost a nenuťtí jej, aby se obrátil na služby jiné společnosti, která bude reagovat rychleji.

Prostřednictvím chatbotů společnost shromažďuje údaje o zákaznících. Díky sběru těchto údajů společnost lépe chápe potřeby a problémy klienta a snaží se je řešit.

Chatboti mohou zvýšit prodej zasíláním reklam na nové produkty a aktualizaci produktů. Vzhledem k tomu, že komunikace probíhá s klientem v soukromí, je to vnímáno jako osobní nabídka chatbota. Lidé na takové reklamy reagují lépe.

### 3.4.2 Nevýhody

Chatboti jsou naprogramovány tak, aby řešily určité problémy, takže pokud dojde k neočekávaným událostem, chatbot nemusí správně reagovat. Někteří lidé stále očekávají komunikaci se skutečnou osobou, ale ne s robotem.

Budování chatbota od nuly je velmi obtížný úkol, který trvá dlouho. Pro vývoj chatbota je třeba mít znalosti z celé škály oblastí. Je zřejmé, že čím efektivnější je chatbot, kterého potřebujete vytvořit, tím více síly a energie bude vyžadovat.

Aby chatbot ve společnosti fungoval dobře, potřebujete jeho neustálou údržbu. Je nutné provést analýzu rozhovoru se zákazníky, aby chatbot mohl odpovědět na otázky, které dosud nezná. Společnost neustále aktualizuje a přidává produkty, což by měl chatbot vzít v úvahu. Aby to fungovalo dobře, musí společnost chatbota často kontrolovat a testovat.

## 3.5 Platformy pro tvorbu chatbotů

K vytvoření chatbota můžete použít dva typy platform. Jedna z nich vyžaduje programátorské dovednosti a druhá je vhodná pro všechny lidi. V následujících podsekcích se podrobněji podíváme na tyto dva typy platform.

### 3.5.1 Neprogramovací platformy

Protože tento pohled nevyžaduje žádné znalosti programování, je jeho použití celkem snadné. Je možné vytvořit lehkého chatbota, který nepoužívá jazykové zpracování nebo ho používá na nízké úrovni. Mezi tyto platformy patří Chatfuel, Botsify, Flow Xo, KITT.AI, atd. [31].

Jelikož je tato práce zaměřena na používání programovacích platform, není nutné se této oblasti věnovat podrobněji.

### 3.5.2 Programovací platformy

Tato sekce je kompilací informací ze zdrojů: [platforms\_for\_creating\_chatbots, 31, 32, 33, 34, 35].

Platformy vyžadující znalosti programování lze rozdělit na dva další typy: platformy využívající AIML a platformy od technologických gigantů.

AIML je značkový jazyk, který se používá k simulaci interakce s uživatelem. Použitím klíčových slov ve větách a určením způsobu jejich použití generuje konkrétní odpověď. Datové objekty, které tvoří AIML, se skládají z témat obsahujících název, atribut a kategorie, které obsahují šablonu pro generování odpovědí. Tento jazyk je lehký, flexibilní a výkonný. Jakmile je však šablona vytvořena, je obtížné ji změnit.

Platformy poskytované technologickými giganty zase využívají snadno pochopitelné entity, úmysly a kontext. Tyto platformy přebírají část logiky chatbota. Nevýhodou je, že tyto platformy poskytují konkrétní domény, které pro účely chatbota nemusí stačit.

Níže jsou uvedeny hlavní rozdíly mezi nejpobulárnějšími platformami - IBM Watson a Microsoft Bot Framework.

IBM Watson se skládá ze 3 hlavních komponent: intenty, entity a dialog. Intenty obsahují ukázkové fráze, které mohou uživatelé použít při komunikaci s chatbotem. To znamená, že jedna intenta definuje cíl, který lze identifikovat



podle vstupu uživatele. Například, intenta s názvem `store_hours` odpovídá na otázky týkající se otevírací doby. Dialog zahrnuje záměry. Pokud graficky znázorníte posloupnost dialogů, bude to strom a každá nová větev je intenta, kterou chatbot zpracovává. Entity obsahují výraz nebo objekt, který poskytuje kontext pro záměr. Microsoft Bot Framework se skládá ze dvou hlavních komponent: Bot Builder SDK<sup>3</sup> pro vývoj a Microsoft LUIS<sup>4</sup>, která představuje vstup uživatele jako kontext a je motorem NLP / NLU<sup>5</sup>.

IBM Watson poskytuje velké množství SDK pro programovací jazyky Node, Java, Python, iOS a Unity. Oproti tomu konkurenční Microsoft Bot Framework má pro vývoj komponentu jako Bot Builder SDK. Toto SDK podporuje .NET a Node.js.

IBM Watson podporuje dva jazyky - angličtinu a japonštinu. Microsoft Bot Framework zase podporuje 30 různých jazyků a automatický překlad, díky čemuž je oblíbenější.

Pokud jde o licenci, produkt IBM Watson má tři možnosti: bezplatnou, standardní a prémiovou. Sada Bot Builder SDK od Microsoft Bot Framework je open source a je k dispozici komukoli na Githubu.

## 3.6 Cestovní chatboti

Chatboti využívaní na webových stránkách cestovních kanceláří nebo rezervačních webech jsou nejoblíbenějšími pomocníky zákazníků. Odpovídají hlavně na často kladené otázky a mohou vám pomoci s orientací na webu. Příkladem mohou být chatboti: Alison - chatbot pro rezervaci dovolené na pláži, Ana - virtuální cestovní agentka Copa Airlines, Julie - poskytuje pomoc při navigaci na Amtrak.com. Tito chatboti jsou poměrně primitivní, tak se jimi dále nebudu zabývat. [36]

### 3.6.1 Rezervační cestovní chatboti

Další populární cestovní chatboti jsou přes Facebook Messenger a Slack. Jsou pokročilejší než předchozí typ, protože pomáhají uživateli s rezervací cesty. Na druhou stranu je to jiné rozhraní pro rezervaci kromě použití samotného webu cestovní kanceláře. Příklady takových chatbotů jsou popsány níže. [36]

#### Expedia bot.

Tento chatbot byl vyvinut cestovní agenturou Expedia pro Facebook Messenger, který umožňuje turistům najít vhodný hotel. Turista nejprve zadá informace o datu a městě příjezdu a poté chatbot nabídne pět možností. Když je vybrán jeden z navrhovaných hotelů, uživateli následuje odkaz na webovou stránku Expedia a provede tam rezervaci. Poté chatbot odešle zprávu s informacemi o rezervaci do Messengeru. Zdroje: [37, 38].

<sup>3</sup>Software Development Kit

<sup>4</sup>Language Understanding Intelligent Service

<sup>5</sup>Natural Language Understanding



nevýhody této technologie. Analýza turistických chatbotů poskytla pochopení nejoblíbenějších oblastí cestovního ruchu, kde lze tuto technologii použít.





## Část II

### Praktická část



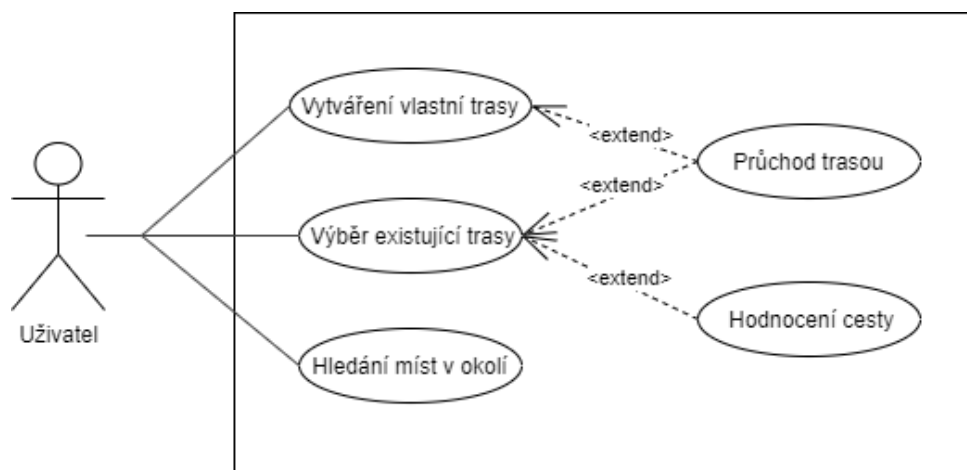
## Kapitola 4

### Návrh aplikace

Předtím, než se začne programovat aplikaci, je nutné se zamyslet nad základem toho, z čeho se bude práce skládat. Tato kapitola popisuje návrh aplikace. V části o případech užití popisují, jaké akce bude systém provádět. To poskytne představu o tom, co od systému bude požadováno. Následující část popisuje výběr technologií, tedy to, co se použije k implementaci chatbota, backendu, frontendu a databázi.

#### 4.1 Případy užití

Pro lepší pochopení hlavních cílů, kterých budou uživatelé schopni dosáhnout, je nutné uvést případy užití webové aplikace. Obrázek 4.1 zobrazuje UML diagram případů užití. Každý z nich je popsán níže.



Obrázek 4.1: UML diagram případů užití

##### UC1. Výběr existující trasy

Webová aplikace umožňuje uživateli vybrat a projít jednu z nabízených tras. Při výběru jedné z navrhovaných prohlídek budou uživateli poskytnuty informace o prohlídce. Těmi mohou být například popis prohlídky, jak dlouho





s chatbotem. Pro vývoj webových aplikací je jedním z populárních programovacích jazyků JavaScript, který jsem si nakonec vybrala. Použité zdroje k této podkapitole: [42], [43].

Jeho výhodou oproti jiným programovacím jazykům je, že skripty jsou podporovány všemi populárními prohlížeči a je jediným programovacím jazykem, který je zabudován do webového prohlížeče. Díky své popularitě je k dispozici velké množství open-source řešení.

Jednou z nevýhod tohoto jazyka je problém práce s datovými typy, což může vést k řadě chyb v programu. Řešením by bylo použít TypeScript, který JavaScriptu dodává statické datové typy. V mé bakalářské práci, která nepoužívá velké množství dat, nebude problém použít JavaScript.

JavaScript-stack je kombinací technologií, které lze použít k vytvoření webové aplikace. Existuje několik populárních JavaScript-stacků. Například MEAN stack, který má následující komponenty: MongoDB pro databáze, Express.js jako backend framework, Angular.js jako frontend framework, Node.js jako cross-platform JavaScript runtime. Nebo například další MERN-stack, kde se místo frameworku Angular.js pro frontend použije knihovna React.js. Pro svou práci jsem si vybrala MERN stack. V následujících podkapitolách popisují výhody tohoto výběru, kde je možné najít informace o každé z komponent tohoto stacku.

### ■ 4.2.3 Frontend

Open-source knihovna React se snadno naučí a je jednou z populárních knihoven. Používá syntaxi podobnou jako HTML s názvem JSX. Použití knihovny React zrychluje načítání a rendering stránky. Rychlé načítání stránek je důležité zejména pro uživatele telefonů. Dalším plusem je opětovné použití komponent, to znamená, že jakýkoli prvek rozhraní lze v projektu použít několikrát jednoduchým voláním. [44]

### ■ 4.2.4 Backend

Na straně serveru je populárním řešením použít Node.js. Jedná se o modul runtime JavaScript, který spouští framework na straně serveru. Jako doplněk k funkcím Node.js se používá framework Express.js, který je flexibilní a jeho použití výrazně snižuje psaní kódu, což znamená, že se snižuje čas na implementaci. Vývoj ve stejném programovacím jazyce ze strany serveru i klienta zajišťuje rychlou synchronizaci a efektivní práci. NPM (Node Package Manager) je správce balíčků Node.js, který neustále roste a obsahuje obrovské množství open-source knihoven, což zrychluje vývoj. Platforma IBM poskytuje sadu SDK pro vytváření komunikace chatbota, která je založena na Node.js, což je pro moji práci velký plus. [45], [46].

### ■ 4.2.5 Databáze

Ve své práci jsem se rozhodla použít databázi NoSql z několika důvodů. Ukládání dat v relačních tabulkách, jak se to děje v databázích SQL, vyžaduje

pečlivě promyšlené struktury dat data k ukládání, protože je příliš nákladné provádět změny. Ve své práci bych chtěla mít flexibilnější schémata, abych v případě jakýchkoli změn v datech dokázala rychle zvládnout situaci a přizpůsobit se. To je jedna z výhod databází NoSql. Z důvodu normalizace dat v databázích SQL je pro vytvoření dotazu nutné kombinovat data z několika tabulek. V databázích NoSQL jsou data ukládána společně za účelem optimalizace dotazů. Proto jsou dotazy rychlé a nevyžadují operaci join. [47]

Jednou z populárních databází NoSQL je MongoDB. Jedná se o databázi s otevřeným zdrojovým kódem, kde lze jako dotazovací jazyk použít JavaScript. Mezi výhody patří vynikající uživatelské rozhraní. Jedním z hlavních důvodů pro výběr MongoDB v mé práci je to, že je součástí MERN stacku, který použiji pro implementaci.

## 4.3 Výběr externích zdrojů

Při výběru externích zdrojů jsem se zaměřila na tři body: plánování trasy s více body na mapě, hledání zajímavých míst v okolí a použití informací o vybraném bodě. Dalším důležitým bodem byla možnost využívat zdroje zdarma nebo nepřekročit limit bezplatného použití. Aby bylo možné rychle pochopit použití konkrétního zdroje, musí mít kvalitní dokumentaci. V následujících podkapitolách popisují zdroje, které jsem vybrala.

### 4.3.1 Google Maps API

Nejznámějším zdrojem souvisejícím s mapami jsou Google Maps. Má rozsáhlou databázi, skvělou dokumentaci a celou řadu souvisejících API. Platba se provádí "Pay-As-You-Go", to znamená za každý požadavek. Cena závisí na počtu požadavků za měsíc a pohybuje se od 3 USD do 10 USD za 1000 požadavků. Google Maps dává každému uživateli měsíční kredit 200 USD a v rámci mé bakalářské práce bude tato částka kreditu dostatečná. Pro komunikaci s API se využívá protokol HTTPS a odpověď přichází ve formátu JSON. Níže popisují související API od Google Maps, které jsem si vybrala.

### Places API

Google Maps má službu Places API, pomocí které je možné hledat místa. Používá se několik kategorií míst, mezi nimi například jsou podniky a slavné památky. Pro svou práci použiji několik metod tohoto API k vyžádání místa.

Jedna z nich je Place Search, která vrací seznam míst spolu s informacemi o každém místě. Hledání se provádí buď na základě umístění uživatele, nebo na textovém řetězci. Následující dotaz Place Details se používá k načtení dalších informací o místě, jako je úplná adresa, telefonní číslo, hodnocení uživateli, recenze nebo reference k fotkám. Důležitou součástí mé práce je také doprovázet každé místo fotografiemi. Služba Place Photos poskytuje přístup k milionům fotografií uložených v databázi Places. [48].

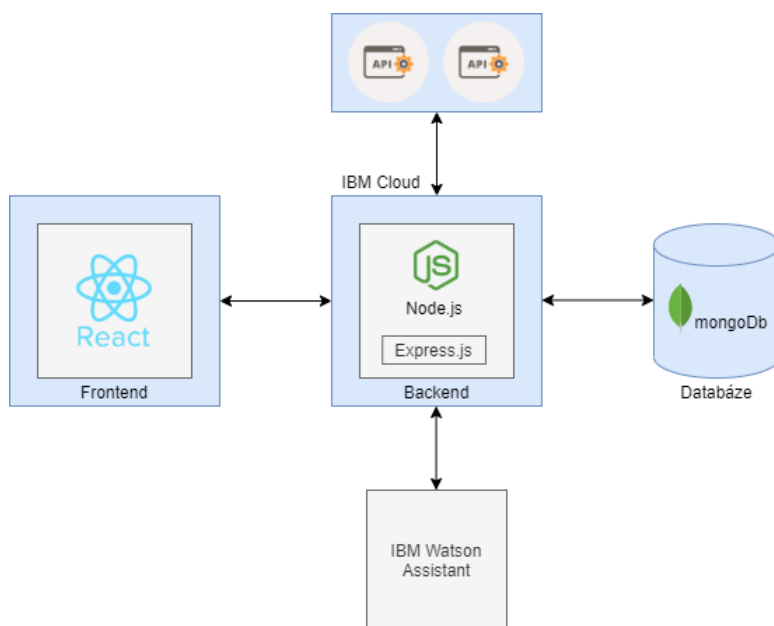
## ■ Directions API

Služba Directions API vrací trasy mezi místy. Je možné zvolit způsob dopravy, například veřejnou dopravu, autem, pěšky nebo na kole. Schopnost vytvořit složenou trasu, tj. přes několik bodů, je přesně to, co moje aplikace vyžaduje. Jednou z výhod je, že jsou vráceny nejefektivnější trasy. Místa lze prohledávat pomocí zadávání textu nebo podle ID místa, které mohu získat z Places API. [49].

### ■ 4.3.2 Wiki API

Aby se chatbot nejvíce podobal průvodci, musí poskytovat podrobnější informace o místě, které uživatel navštěvuje. Pro svoji práci jsem se rozhodla použít online encyklopedii Wikipedia, která obsahuje spoustu informací o různých zajímavostech, které se v Praze nacházejí. Výhodou používání tohoto API je možnost využívat zdarma a k tomu ještě má dobrou dokumentaci. [50]

## ■ 4.4 Architektura aplikace



**Obrázek 4.2:** Návrh architektury

Návrh architektury zobrazený na obrázku 4.2 je založen na vybraných technologiích a případech užití. Frontend komunikuje s backendem pomocí REST API. Backend zpracovává požadavky na základě toho, co frontend potřebuje, což mohou být informace z databázi nebo odpověď od IBM Watson Assistant. Některé odpovědi od asistenta budou vyžadovat data z externích zdrojů. Připravená odpověď bude vrácena do frontendu.

## 4.5 Návrh databáze

Databáze bude sloužit především k ukládání informací o dokončených prohlídkách a místech. Při použití databáze není nutné pokaždé odesílat požadavek na externí zdroje. Jak již bylo zmíněno, za každý požadavek odeslaný do rozhraní Google Maps API je účtován poplatek. To znamená, že použití databáze ušetří náklady. Na obrázku 4.3 je vidět návrh schémat kolekcí Tour a Waypoint. Je nutné uložit všechny informace, které je třeba uživatelům pak poskytnout.

Tour	Waypoint
<pre>{   "_id": ObjectID,   "nameTour": String,   "waypoints": [Waypoint.name],   "tourDescription": String,   "mapsLink": String,   "ratingAmount": {     "five": Number,     "four": Number,     "three": Number,     "two": Number,     "one": Number,   },   "totalRating": Number,   "duration": String, }</pre>	<pre>{   "_id": ObjectID,   "name": String,   "formatted_address": String,   "url": String,   "photo1": String,   "photo2": String, }</pre>

Obrázek 4.3: Návrh databáze

## 4.6 Shrnutí kapitoly

V této kapitole byly řešeny věci ohledně návrhu aplikace. Definice případů užití stanoví pro chatbota jasné cíle a vhodně vybrané technologie zkrátí dobu vývoje. Návrh architektury a databáze dává lepší představu o aplikaci. Toto poslouží jako dobrý základ pro přechod do další fáze - vývoje aplikace, která bude popsána v následující kapitole.

## Kapitola 5

### Implementace

V této kapitole popíšu postup vytváření aplikace mé práce a hlavní body vývoje. Implementace se skládá ze tří částí: vývoj chatbota, backendu a frontendu. Část chatbot popisuje proces vytváření, z čeho se chatbot skládá a taky průchod dialogem. V další části je popsán proces vytváření frontendu aplikace. Pak následuje část o backendu, kde popisují práci s databází, použití externích zdrojů a komunikace s frontendem.

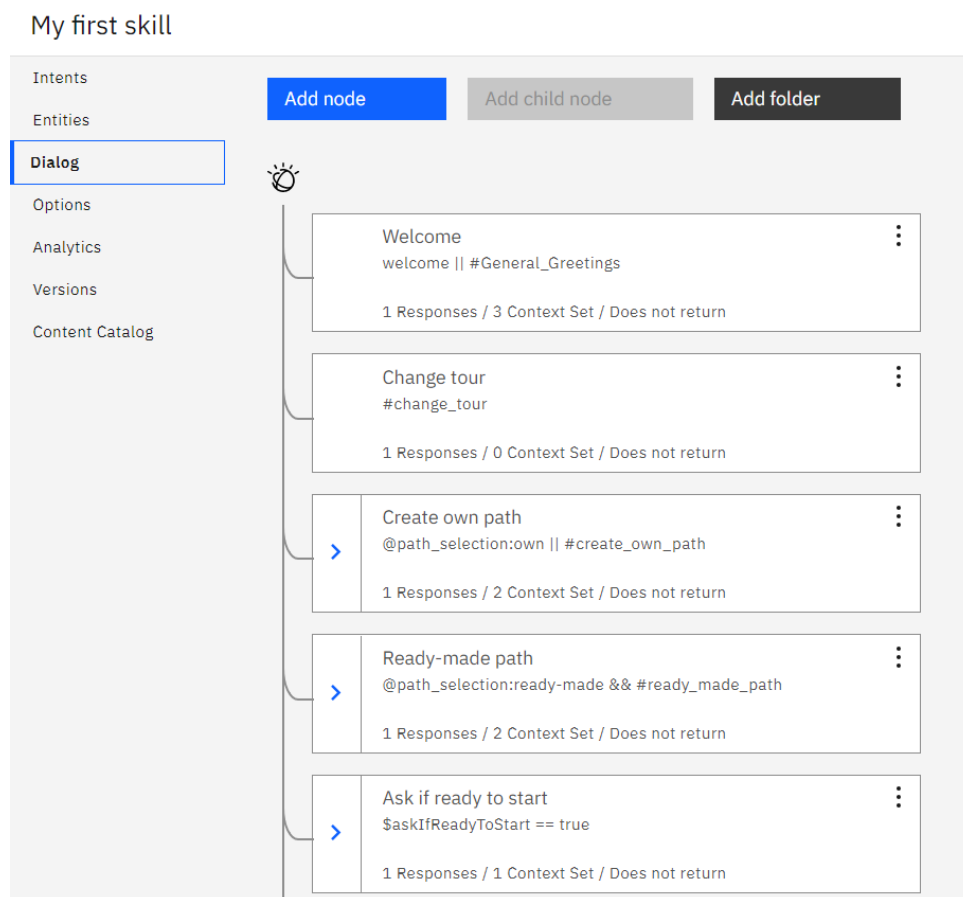
#### 5.1 Vytvoření chatbota

Implementace chatbota se skládá z několika kroků. První krok je před prací s platformou chatbot. Je potřeba se rozhodnout, jaké požadavky uživatelů splní. Na základě případů užití, které byly popsány v minulé kapitole v sekci 4.1, hlavním účelem chatbota bude vytvoření vlastní trasy, průchod po připravené trase a hledání zajímavých míst v blízkosti.

Další krok zahrnuje práci se samotným produktem IBM Watson Assistant. V dalších sekcích popíšu hlavní části platformy, které jsem potřebovala k vytváření chatbota.

##### 5.1.1 Dialog

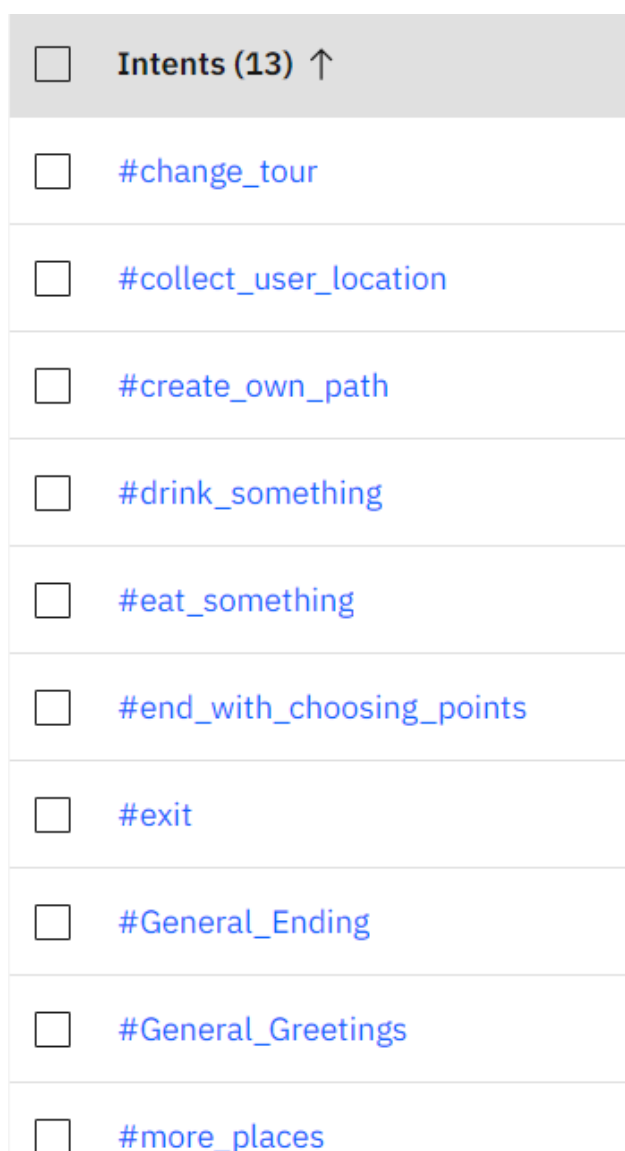
Dialog graficky představuje strom. Každá větev stromu je zodpovědná za zpracování určitého záměru (Intents) od uživatele. Větev se zase skládá z uzlů. Uzel se minimálně skládá z podmínky a odpovědi. Zpracování uzlu lze považovat za konstrukci `if / then`: pokud je tato podmínka pravdivá, vraťte tuto odpověď. Můj dialog se skládá z 10 větví, některé z nich můžete vidět na obrázku 5.1. Následující důležité části systému, které v uzlu mohou fungovat jako podmínka jsou intent, entita a kontextová proměnná. [51]



Obrázek 5.1: Dialog na platformě IBM Watson Assistant

### 5.1.2 Intent

Jak již bylo zmíněno v části 3.5.2, intenty jsou cíle nebo úkoly uživatele, které obsahují vzorové fráze, které budou případně odeslány chatbotu. Jedná se o nějaký druh akce, kterou chce uživatel udělat, to znamená, že se nejčastěji jedná o sloveso. Jsou označeny znakem #. Můj chatbot má 13 intent. Některé z nich můžete vidět na obrázku 5.2. Tři hlavní intenty, které chatbot používá, jsou **#create\_own\_path** - pokud si uživatel chce vytvořit vlastní cestu, **#ready\_made\_path** - jestli chce jít hotovou cestou, **#more\_places** - pokud chce najít další zajímavá místa. [52].



Obrázek 5.2: Intents na platformě IBM Watson Assistant

### ■ 5.1.3 Entity

Dalším krokem je vytvoření entit k identifikaci informací od uživatele, které jsou relevantní pro jeho cíl. Jedná se o nějaký druh objektu nebo kontextu akce, kterou chce uživatel provést, nejčastěji jde o podstatné jméno. Jsou označeny znakem @. Například můj chatbot má entitu, která se nazývá @location, která určuje adresu uživatele, když ji odešle. Tuto adresu používám k vyhledání míst v blízkosti uživatele, to znamená pro intent, který se nazývá #more\_places. [53].

### 5.1.4 Kontextová proměnná

K uložení potřebných informací během dialogu s uživatelem se používají kontextové proměnné. Jsou definovány v uzlu a lze jim přiřadit výchozí hodnotu. Mají znak \$. Na obrázku 5.3 je vidět, jak jsem implementovala kontextovou proměnnou nazvanou \$waypoints, kde později uložím název míst, která chce uživatel přidat, když vytvoří svou osobní cestu. [54].

#### Then set context

The conversation\_start node should be used to set context. [Learn more](#)

Variable	Value
waypoints	[]

Obrázek 5.3: Intents na platformě IBM Watson Assistant

### 5.1.5 Webhook

Nejdůležitější částí vyvinutého chatbota je použití v dialogu takového mechanismu, jako je **webhook**. Z dokumentace Watson Assistant webhook umožňuje přístup k externímu programu v závislosti na tom, co se ve vašem programu děje. Pro použití webhooku je nutné definovat jednu adresu URL, na kterou bude odeslán požadavek metodou POST, a také povolit tuto funkci v potřebném uzlu. Při zpracování tohoto uzlu se aktivuje definovaný webhook. Zpracování požadavku by mělo trvat maximálně 8 sekund. V požadavku lze zaslat další proměnné, což mohou být také kontextové proměnné. S tímto mechanismem realizují požadavky na externí zdroje, které používám v této aplikaci. [55].

Čtyři uzly v tomto chatbotu mají povolenou tuto funkci, která se používá k následujícím účelům:

1. vyhledání místa a informací o místě, které si uživatel vybral,
2. vytvoření cesty z vybraných bodů,
3. vyhledávání dalších informací o místě,
4. hledání míst v okolí.

### 5.1.6 Typy zpráv od chatbota

Chatbot může posílat různé druhy zpráv pro lepší uživatelský komfort. Defaultní odpověď je text. Kromě toho existuje pět dalších typů. Typ možnosti, kde odpověď asistenta nabízí seznam jedné nebo více možností. Když uživatel klikne na jednu z možností, asistent obdrží hodnotu s ní spojenou. Dalším typem odpovědi je obrázek, kde je třeba zadat adresu URL, kde je obrázek uložen. Soubor musí být veřejně dostupný. Moje aplikace používá fotografie



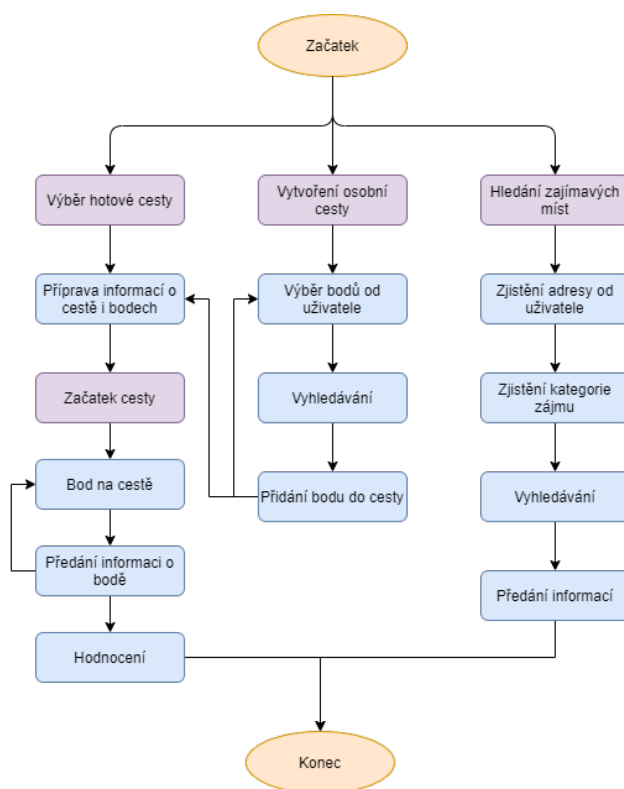
uložené v Google Maps, takže to nebyl problém. Vyvinutý chatbot používá tyto tři typy odpovědí: text, možnosti a obrázky. Existují ale 3 další typy: pauza - čekání na zadaný počet milisekund před dalším zpracováním, přenos kanálu - odeslán požadavek na přenos dialogu na jiný kanál, připojení k lidskému agentovi - kde se volá služba k přenosu konverzace na osobu. [56].

### 5.1.7 Testování chatbota na platformě

Pro kontrolu správného procházení dialogem je nutné ho neustále testovat. Platforma poskytuje vynikající příležitost otestovat chatbota přímo na platformě pomocí dialogového okna. Tímto způsobem lze všechny chyby opravit ihned na platformě a okamžitě je otestovat.

### 5.1.8 Průchod dialogem

V této části popíšu, jak probíhá dialog s chatbotem. Zahrnuty jsou pouze hlavní části dialogu. Schéma je vidět na obrázku 5.4.



Obrázek 5.4: Průchod dialogem

#### Začatek

Chatbot hned na začátku uživatele pozdraví a poskytne o sobě informace. Rovněž seznámí uživatele s tím, co může pomocí tohoto chatbota dělat. Poté

chatbot odešle tři možnosti, které si uživatel může vybrat: hotovou cestu, vytvořit si vlastní cestu nebo hledání zajímavých míst.

### ■ Výběr hotové cesty

Uživatel má možnost vybrat si jednu z čtyř navrhovaných hotových cest. Backend zachytí název cesty a v databázi se prohledají informace o vybrané cestě. Připravené informace se uloží do proměnné na backendu a přiřadí se kontextové proměnné v chatbotu.

### ■ Vytvoření osobní cesty

Existuje několik kroků k vytvoření vlastní prohlídky. Hlavní věc je výběr míst, která chce uživatel navštívit. Pořadí výběru míst je následující: nejprve je vybráno počáteční místo, poté konečné a potom místa, která chce mezi nimi navštívit. Omezení počtu vybraných míst pro návštěvu se pohybují od 2 do 9. Tato volba počtu bodů je způsobena skutečností, že trasu v Google Maps lze provést maximálně z 9 bodů. Názvy míst od uživatele se odesílají na backend aplikace pomocí webhooku, kde se místo vyhledává pomocí požadavku na Places API. Poté, co uživatel vybere všechna místa, která chce navštívit, provede se následující typ požadavku na backend, kde je vybrána nejlepší varianta pořadí míst mezi prvním a posledním bodem. To se provádí odesláním požadavku na Direction API. Vytvoří se také odkaz na Google Maps, kde je uvedena celá cesta. Pak začíná samotná prohlídka.

### ■ Průchod cestou

Prohlídka zahrnuje několik základních kroků. Nejprve je uživateli zaslán název a adresa místa spolu s odkazem na Google Maps, kde je místo označeno. Dále musí uživatel potvrdit, že na dané místo dorazil. Poté se odešlou další informace o místě. K tomu je odeslán požadavek na backend, kde je pomocí požadavku na Wiki Api vyhledán článek o daném místě. Navíc je odeslán odkaz na článek a web, aby si uživatel mohl přečíst více informací. Pokud uživatel prošel existující cestu, tak na konci ji může ohodnotit. Jinak po skončení vlastní prohlídky chatbot prostě poděkuje a nabídne další možnosti, které uživatel může udělat pomocí chatbota.

### ■ Hledání zajímavých míst

Na začátku tohoto procesu se chatbot zeptá na adresu, kde se uživatel nachází. Adresa je pak uložena v kontextové proměnné. Chatbot se poté zeptá, o co má uživatel zájem. Uživatel si může vybrat z následujících kategorií: restaurace a bary, muzea, galerie, parky a památky. Kategorie se také uloží do kontextové proměnné. Dále se odešle požadavek na backend aplikace, kam se odešlou data shromážděná chatbotem, tj. adresa a kategorie zájmu. Vyhledávání se provádí pomocí externích zdrojů. Nalezené informace se odešlou zpět uživateli.

## ■ Konec konverzace

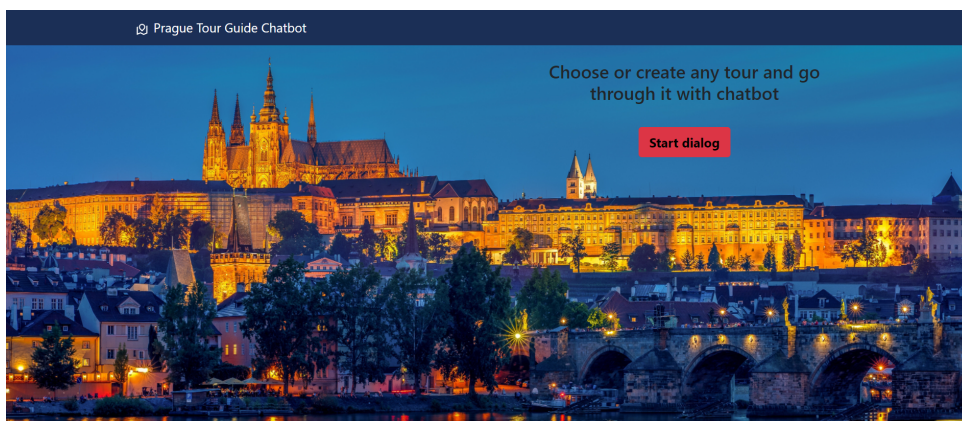
Na konci dialogu se chatbot s uživatelem rozloučí.

## ■ 5.2 Frontend

Vytvoření přátelského uživatelského rozhraní je důležitou součástí vytváření jakékoli aplikace. Vývoj frontendu byl rozdělen do dvou kroků. Prvním je vytvoření designu aplikace a následné programování.

### ■ 5.2.1 Tvorba designu

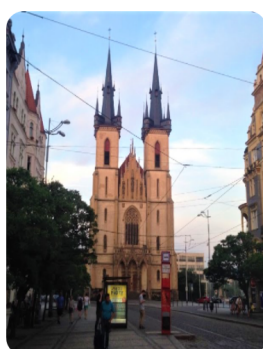
Nástroj, který jsem používala pro tvorbu designu je Figma<sup>1</sup>. Celkově jsem vytvořila dva různé návrhy webových stránek. Každý z nich byl konzultován s odborníkem v této oblasti. Bylo rozhodnuto, že první návrh byl příliš složitý a měl příliš mnoho zbytečných částí, které by odváděly pozornost uživatele od hlavních cílů. Poté byl vytvořen druhý a konečný design, kde již byly viditelné hlavní části webové aplikace. Uživatel musí pochopit, co mu komunikace s chatbotem poskytne. Na základě toho byly vyrobeny následující části webu.



**Obrázek 5.5:** Design webové aplikace - hlavní část

Část webu, kterou lze vidět na obrázku 5.5, je hlavní, protože obsahuje hlavní tlačítko, pomocí kterého je možné se rovnou přepnout na dialog s chatbotem. Aby mu uživatel věnoval pozornost, liší se velikostí a barvou od ostatních tlačítek na webu. Na obrázku 5.6 vidíte, jak byla představena každá z existujících prohlídek. Bylo důležité poskytnout všechny potřebné informace, aby si uživatel mohl představit, co ho čeká, pokud si zvolí tuto prohlídku. Následující obrázek 5.7 ukazuje část webu, která popisuje další cíle, kterých může uživatel dosáhnout prostřednictvím komunikace s chatbotem: vytvořit vlastní cestu a najít místa v okolí. Když byl návrh připraven, přešla jsem na následující krok, což je psaní kódů, kterou popíšu v příští podsekcí.

<sup>1</sup><https://www.figma.com>



### Holesovice tour

**Rating:** 2.5  
**Duration:** 1 hour  
**Waypoints:** 8

Holešovice is the most interesting and fast changing area in the north of Prague. Previously, it was an industrial suburb, where many factories and plants were concentrated. Today they have been reconstructed and converted into apartment buildings, offices, and places for creative people. There are many cafes and bistros, alternative cultural venues and designer shops.

**What you will see:** Cross Club - Tipsport arena - Lapidarium of the National Museum - Industrial Palace - Prague Planetarium - National Gallery Prague - Trade Fair Palace - Letná Park - Prague Metronome

Start tour

Obrázek 5.6: Holešovice tour

Do you want more?

#### Create your own tour

With the chatbot, you can create your own tour. To do this, you need a start point, an end point, and the rest of the points you want to visit. The chatbot will create a path for you, and guide you along it, telling you information about each point along the way!

Create tour

#### Find interesting places nearby

If you want to find something interesting near you, use this chatbot. Provide your address and select the category of your interest, and the chatbot will find open places for you to visit.

Find places

Obrázek 5.7: Design webové aplikace - další části

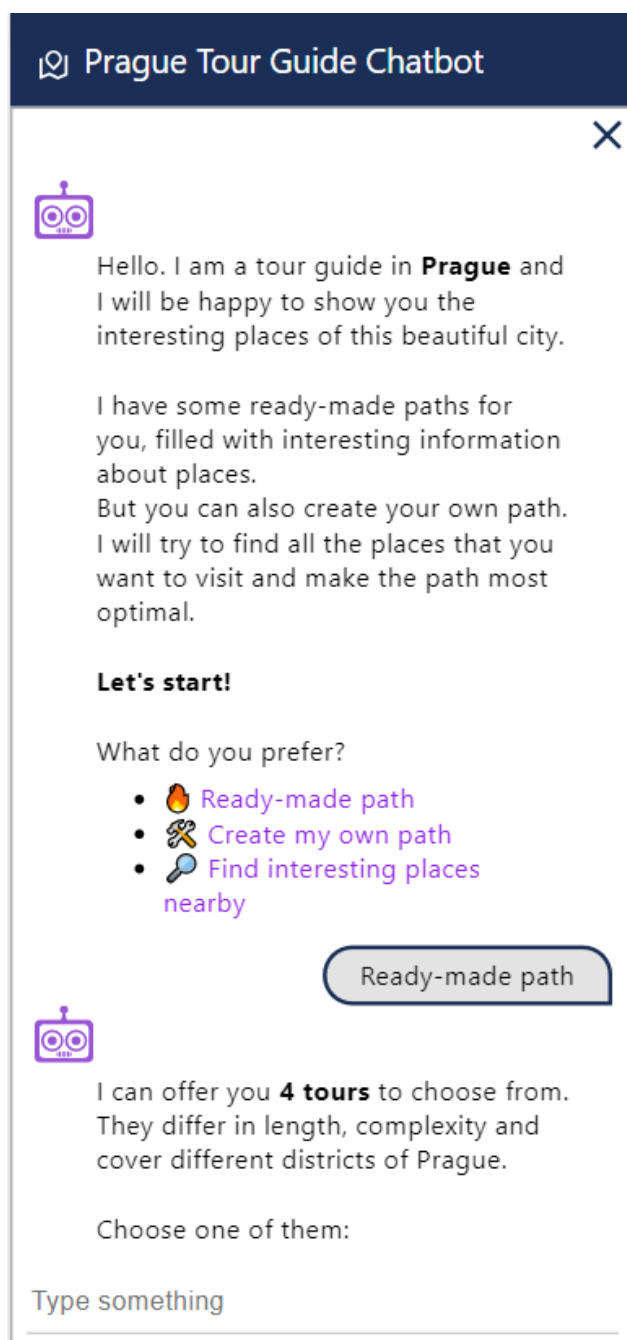
## 5.2.2 Vývoj

Jak již bylo uvedeno v sekci 4.2.3, k vytvoření frontendu byla použita knihovna React.js. Pro kaskádové styly jsem použila knihovnu css Bootstrap.

Práce obsahuje dvě stránky. Hlavní stránka obsahuje informace o tom, jaké cíle může uživatel pomocí chatbota dosáhnout, a také informace o připravených prohlídkách. Prohlídková data se používají z databáze. K ukládání těchto dat do jednoho místa, odkud je pak můžu používat z jakékoli komponenty, jsem použila knihovnu Redux.js, která pomáhá ovládat stav aplikace.

Na stránce jsou 4 typy tlačítek, která přenesou uživatele do druhé stránky s dialogem a okamžitě zahájí dialog s tím, co uživatel potřebuje. Za tímto účelem je chatbotu zaslána počáteční zpráva. Tlačítko „Spustit dialog“ - zahájí dialog od začátku a nabízí možnosti, které si uživatel může vybrat. Chatbotu se odešle prázdná zpráva k zahájení konverzace. Další tlačítko „Zahájit prohlídku“ - spustí konkrétní prohlídku, kterou si uživatel vybral, odeslaná zpráva obsahuje název prohlídky. Poté tlačítko „Vytvořit prohlídku“ - zahájí proces vytváření cesty a nakonec tlačítko „Najít místa“ - zahájí proces hledání míst v okolí.

Druhá stránka obsahuje dialog s chatbotem. První zpráva pochází od chatbota. Na frontendu se zpracovává každý použitý typ zprávy od chatbota, který je popsán v sekci 5.1.6. Poté může uživatel odesílat zprávy nebo vybrat jednu z možností nabízených chatbotem kliknutím na tuto možnost. Stránka s dialogem byla vytvořena stejným způsobem jako v šabloně pro backend. K tomu jsem také použila stejné styly.



Obrázek 5.8: Stránka s dialogem na webu

Většina turistů se bude pohybovat po městě, takže jedním z cílů práce bylo vytvořit přátelské rozhraní pro používání chatbota na telefonech. K tomu byly použity styly css, které se mění v závislosti na velikosti obrazovky. Na obrazce 5.8 je vidět jak vypadá dialog na mobilu iPhone X.

## 5.3 Backend

Backend aplikace je postavena na Node.js. Backend se používá hlavně pro komunikaci s frontendem, stejně jako k čtení informací o hotových prohlídkách z databáze. Byla použita připravená šablona od IBM, kterou je možné najít na GitHubu<sup>2</sup>. Tuto šablonu jsem doplnila prací s databází, přidáním práce s kontextovými proměnnými a zpracováním požadavků na externí zdroje.

### Informace o připravených prohlídkách z databáze

Za databází jsem použila cloudovou databázovou službu MongoDB Atlas. Databáze se používá k ukládání informací o připravených cestách a informacích o bodech, které tyto cesty obsahují. Tím je zajištěno rychlé zpracování zpráv od chatbota. Asistent obdrží tato data v okamžiku, kdy se v těle požadavku objeví určité kontextové proměnné, konkrétně boolean proměnná `findTourFromDb`, která musí být `true`. Hledání prohlídky se provádí pomocí požadavku GET do endpointu `api/tours` a hledání informací o bodech v cestě do endpointu `api/waypoint`. Poté se shromážděné informace přidávají do kontextových proměnných, se kterými asistent pracuje.

### Webhook a externí zdroje

Jak již bylo zmíněno v sekci 5.1.5, aby asistent mohl zpracovávat požadavky na různé externí zdroje, používá se webhook. Protože webhook je adresa URL a bylo třeba, aby byla veřejně dostupná, bylo nutné nahrát backend na server. K tomu IBM má runtime aplikace založené na Cloud Foundry. Podle návodů jsem nahrála aplikaci na Bluemix server a bylo možné zpracovat žádosti od asistenta.

Asistent vytvoří požadavek na webhook pro koncový bod `/myplaces`. První uzel hledá místo podle jména, které uživatel zadá. K tomu byly použity 2 typy požadavků z rozhraní Places API. Place Search hledá ID místa a Place Details pomocí ID místa poskytuje zbytek informací, což jsou celé jméno místa, adresa, URL a odkaz na fotografii. Všechny nalezené informace se odesílají zpět chatbotovi.

Další uzel používá webhook pro získání správného pořadí vybraných bodů a pro vytváření odkazu na Google Maps, kde je vidět cestu. K tomu backend na začátku provede požadavek na Direction API. Data, která se mají odeslat v těle požadavku: způsob jízdy, počáteční bod, koncový bod a body mezi nimi. V aplikaci se používají dvě možnosti způsobu dopravy, a to autem a pěšky. Veřejnou dopravu nelze použít ve spojení s optimalizací cesty. Rozhodla

<sup>2</sup><https://github.com/watson-developer-cloud/assistant-simple>

jsem se cyklistickému způsobu dopravy se vyhnout, protože zahrnuje pouze cyklistické stezky. Po obdržení odpovědi se správným pořadím, vytváří se odkaz na Google Maps s vytvořenou cestou. Asistentu se odešle tato informace. Tím se uzavírá vytvoření vlastní cesty.

Další potřeba použít webhook je, když je nutné získat informace z Wiki Api. Pro zpracování tohoto požadavku je vyžadován pouze název místa. Dále je možné zadat kolik vět z článku odeslat v odpovědi, a zda přidat k odpovědi link na článek. Nevýhodou používání Wiki Api je omezený počet existujících článků o pražských zajímavých místech. Uživatel může ke své prohlídce přidat různá místa, ale to nezaručuje, že bude o tomto místě existovat článek v Wikipedii. Pro budoucí vývoj aplikace bude tedy nutné použít i jiný zdroj informací. Další věc je to, že při práci s Wiki API se provádí vyhledávání ve všech článcích existujících ve Wikipedii. Problém může nastat, když je například nutné najít informace o Národním muzeu. Existuje mnoho článků o národních muzeích v různých městech. Aby bylo hledání konkrétnější přidávám ke každému názvu místa slovo „Prague“, aby se nalezený článek týkal tohoto města.

Další typ dotazu se používá k vyhledání místa v okolí. K tomu používám Nearby Search požadavek z rozhraní Places API. Požadované parametry pro použití tohoto dotazu jsou místo a poloměr hledání. U místa je nutné uvést zeměpisnou šířku a délku místa, kolem kterého bude probíhat hledání. Proto se na začátku dělá požadavek na Place Search, kde je uvedena adresa, kterou uživatel zaslal. Po obdržení této nutné informace, probíhá Neraby Search požadavek. Nastavila jsem 1000 metrů jako poloměr hledání. Člověk tuto vzdálenost projde za 15 minut, což není příliš dlouhá doba.

## ■ Komunikace mezi frontendem a backendem

Na stránce s dialogem probíhá komunikace s Watson Assistant. Tato aplikace používá Watson Assistant V2 API. Pro posílání zpráv od uživatele a získání odpovědi, je nutné mít session Id. Za tímto účelem se odešle požadavek GET do koncového bodu „api/session“. K tomu ještě je potřebné mít assistant Id, které je k nalezení v nastavení na IBM Watson Assistant. Na obrázku 5.9 s kódem je vidět, jak se na backendu vytváří session Id.

```
app.get("/api/session", function (req, res) {
  assistant.createSession(
    {
      assistantId: process.env.ASSISTANT_ID || "{assistant_id}",
    },
    function (error, response) {
      if (error) {
        return res.send(error);
      } else {
        return res.send(response);
      }
    }
  );
});
```

**Obrázek 5.9:** Vytváření session Id

Tělo každé zprávy od uživatele musí obsahovat session Id, assistant Id a taky objekt s textem. Zpráva je odeslána pomocí požadavku POST do endpointu „api/message“.

Tělo požadavku může mít nejen text zprávy od uživatele, ale také kontextové proměnné, které budou asistentovi k dispozici. Například když chce uživatel z hlavní stránky zahájit dialog okamžitě procházením určité prohlídky, do těla zprávy se odesílají potřebné kontextové proměnné, které umožní zahájit dialog ze správného místa.

## ■ 5.4 Shrnutí kapitoly

V této kapitole byl popsán proces implementace webové aplikace. Také je popsána práce s platformou IBM pro vytvoření chatbota. V části o frontendu je popsáno, jak jsem vytvořila návrh designu a samotná implementace. Také se zde nachází popis hlavní části backendu aplikace. Kód aplikace a návod k jejímu spuštění je možné nalézt v příloze C. Další krok moji práce je uživatelské testování webové aplikace s integrovaným chatbotem, které popíšu v následující kapitole.





## **Část III**

### **Testování a vyhodnocení implementované aplikace**



## Kapitola 6

### Testování chatbota

Každý krok ve vývoji chatbota byl okamžitě testován, aby bylo možné vidět požadovaný výsledek práce. Bylo provedeno uživatelské testování za účelem získání dalších informací o správnosti chatbota. Tato kapitola popisuje cílovou skupinu, jaké testovací scénáře byly připraveny a taky výsledky testování.

#### 6.1 Cílová skupina

Pro výběr účastníků testování, je nutné se nejprve rozhodnout jaká je cílová skupina aplikace. Implementovaný chatbot souvisí s tématem cestování. Lidé různého věku, kteří se věnují různým činnostem a žijí v různých zemích, rádi cestují. Proto je hlavním kritériem pro cílovou skupinu mít internet a zařízení pro používání chatbota.

Testování se účastnilo 5 dobrovolníků ve věkové kategorii od 19 do 30 let. Dva z nich jsou studenti ekonomického oboru a ostatní pracují v oblasti financí, designu a IT. Testování proběhlo voláním přes Microsoft Teams, kde je možné převést kontrolu nad obrazovkou. Mým úkolem bylo sledovat, jak testeři procházejí scénáři a zjistit, kde nastávají potíže.

Za účelem získání zpětné vazby od účastníků byly po testování položeny následující otázky:

- Líbila se vám aplikace?
- Jste spokojeni s komunikací s chatbotem?
- Používali byste chatbota při svých cestách?
- Co vám chybělo a co byste chtěli přidat pro větší komfort při komunikaci s chatbotem?
- Doporučili byste tuto aplikaci?

Díky takovým otázkám bude možné aplikaci vyhodnotit.

#### 6.2 Testovací scénáře

Pro pokrytí celého průchodu dialogu s chatbotem byly připraveny 4 testovací scénáře. Byly založeny na případech užití popsanych v sekci 4.1.

### 6.2.1 První scénář

Scénář představuje výběr existující cesty a její průchod. Úkolem testera bylo vybrat jednu z navrhovaných cest, projít celou cestu a na konci ji vyhodnotit. Kroky, které měl tester udělat:

1. Vyberte možnost projít připravenou cestou.
2. Vyberte jednu z připravených cest.
3. Dokončete všechna místa zahrnutá v cestě.
4. Uveďte své hodnocení.
5. Ukončete dialog.

### 6.2.2 Druhý scénář

Scénář zahrnuje vytvoření osobní cesty pomocí chatbota. Tester musí vybrat počáteční bod, koncový bod a několik dalších míst a poté projít celou cestu, kterou vytvořil. Kroky, které měl tester udělat:

1. Vyberte možnost vytvořit osobní cestu.
2. Napište adresu výchozího bodu.
3. Napište adresu koncového bodu.
4. Přidejte do cesty několik dalších bodů.
5. Dokončete výběr bodů.
6. Projděte vytvořenou cestu.
7. Ukončete dialog.

### 6.2.3 Třetí scénář

Scénář zahrnuje hledání zajímavých míst v okolí. Tester by měl poslat adresu a vybrat kategorii vyhledávání.

1. Vyberte možnost najít zajímavá místa v okolí.
2. Napište adresu.
3. Vyberte kategorii zájmu.
4. Ukončete dialog.

#### 6.2.4 Čtvrtý scénář

Scénář implementuje situaci, kdy uživatel chce při průchodu cestou najít zajímavé místo v okolí. Úkolem testera je vybrat prohlídku, začít ji procházet a poté se uprostřed dialogu pokusit najít zajímavá místa v okolí.

1. Vyberte možnost projít připravenou cestou.
2. Vyberte jednu z připravených cest.
3. Začnete procházet cestou.
4. Po několika bodech napište chatbotovi, že chcete v okolí najít něco zajímavého.
5. Napište adresu.
6. Vyberte kategorii zájmu.
7. Po obdržení odpovědi od chatbota pokračujte po cestě k dalšímu bodu.
8. Dokončete všechna zbývající místa zahrnutá v cestě.
9. Ukončete dialog.

### 6.3 Výsledky testování

Účelem testování bylo vyhodnotit práci chatbota a najít neočekávané akce od uživatelů. Během testování bylo také zjištěno neočekávané chování chatbota.

Jedním z bodů bylo nepřesné vysvětlení toho, co by měl uživatel dělat. To znamená nesprávné formulování vět. Například, někde bylo uvedeno „vybrat místo“ místo „napsat místo“. To bylo opraveno přímo na platformě IBM Watson.

Při výběru míst pro vlastní prohlídku tester uvedl adresu místo názvu. Proto bylo přidáno vysvětlení toho, co je žádoucí zvolit jako místa k návštěvě, tj. název muzeí, parků, galerií, atd. To je důležité při hledání informací o místě z Wikipedie.

Při vytváření prohlídky si uživatel nevěšil informace o pořadí výběru míst, tj. nejprve první, potom poslední místo na cestě a poté již místa mezi nimi. Aby tomu uživatel věnoval pozornost, označila jsem důležité části textu velkým tučným písmem.

Po provedení testů a pohovorech s testery se ukázalo, že funkčnost aplikace se všem líbila. Odpovědi byly takové, že dosáhly svých cílů, a že komunikace s chatbotem je velmi zajímavá. Jeden účastník uvedl, že komunikace s chatbotem je docela intuitivní. V další kapitole bude provedeno vyhodnocení práce, které je založeno na tom, zda by lidé používali chatbota. Hodnocení se provádí primárně na základě výsledků testování.



# Kapitola 7

## Vyhodnocení

Na konci každé provedené práce je užitečné ji vyhodnotit. V budoucnu by se také mohlo rozšířit používání chatbota k vytváření prohlídek po městě. V této kapitole bude popsáno hodnocení výsledku implementace. Popíšu také nápady na vylepšení aplikace a možnost dalšího rozvoje.

### 7.1 Výsledek práce

Oblast cestovního ruchu používá chatboty k různým účelům. Po hledání chatbotů s podobnou funkcí v České republice nebylo nic nalezeno. Vytvořený chatbot je jedinečný v tom, že jej lze použít k vytváření prohlídek po Praze i jako doprovodu během prohlídky. Po testování čtyři z pěti testerů uvedla, že by místo obvyklého hledání míst a vytváření cest v Google Maps zvolila chatování s chatbotem. Z toho lze pochopit, že komunikace s chatbotem je pro lidi přívětivější než běžné vyhledávání na internetu.

### 7.2 Návrhy vylepšení stávající funkcionality

Po absolvování všech testovacích scénářů byl každý tester dotázán, co jim nejvíce chybí při komunikaci s chatbotem a co je podle jejich názoru třeba přidat. Tato část na základě jejich doporučení popisuje, co by mohlo zlepšit uživatelskou přívětivost.

Jednou z hlavních věcí, kterou je třeba vylepšit, bude přidání map. To by bylo lepší při zobrazení celé cesty nebo bodu na mapě než používání odkazů na Google Maps, protože uživatel nemusí otevírat další okna. Také přidání map při výběru míst, kdy uživatel vytváří cestu by zlepšilo interakci uživatele. Uživatel tedy nemusí znát název místa, ale mohl by je jednoduše vybrat na mapě.

Další věcí je použití geolokace. Při hledání míst v okolí by bylo vhodné požádat o povolení k použití geolokace, aby uživatel nemusel ručně hledat svou polohu, kde by navíc mohl udělat chybu. To by urychlilo proces hledání míst, protože by byla použita přesná data. Také pomocí geolokace by bylo možné vylepšit proces a pomoci uživateli dostat se na místo na prohlídce, odesláním mapy s cestou z místa uživatele na další místo na prohlídce.





## Kapitola 8

### Závěr

Tato práce měla dva cíle, a to je seznámení s tématy cestovního ruchu a technologie chatbot a taky implementaci samotného chatbota. Chatbot by měl být v roli pražského průvodce, který také pomáhá s tvorbou prohlídek a hledá zajímavá místa v okolí.

Teoretická část této práce se zabývá seznámením s tématy cestování a technologie chatbot. Pro to jsou vyhrazeny první dvě kapitoly, kde jsem udělala rešerše těchto témat. V první kapitole byla uvedena definice cestování, druhy cestovního ruchu, cestovní ruch v Praze a organizace turistických tras. Ve druhé kapitole jsem popsala definici chatbota, jeho historii, typy chatbotů, výhody a nevýhody této technologie, platformy pro vytváření chatbotů a stávající chatboti v cestovním ruchu. Po prozkoumání turistických chatbotů jsem se přesvědčila, že v Praze neexistují žádní chatboti, kteří by turisty po městě provázeli a vyprávěli jim informace o památkách. Díky tomu jsem pochopila, že téma mé bakalářské práce je docela aktuální.

Před zahájením implementace chatbota jsem vytvořila návrh aplikace. V této kapitole rozhoduji o hlavních akcích, které může chatbot pro uživatele provádět, a také si vybírám technologie, které při implementaci použiji.

V další kapitole popisuji, jak probíhala implementace. Výsledkem je chatbot, který má pro uživatele tři hlavní možnosti. Chatbot nabízí vytvoření cesty, výběr hotové prohlídky, a průchod cestou spolu s uživatelem. Jako další funkci je možné vyhledávat zajímavá místa v okolí.

Po implementaci chatbota jsem provedla uživatelské testování. Díky tomu bylo neočekávané chování chatbota opraveno. Na základě testování v další kapitole jsem vyhodnotila práci, kde jsem popsala přání uživatelů ke zlepšení. Byly popsány také nápady pro další vývoj chatbota.

Jako přínos pro sebe bylo to, že jsem se naučila vytvářet chatboty na platformě IBM Watson Assistant. Také jsem získala více zkušeností s programováním v jazyce JavaScript, který byl použit k vytvoření aplikace.





## Přílohy



# Příloha A

## Literatura a zdroje

1. *Tourism* [online]. Tourism Notes [cit. 2021-12-12]. Dostupné z: <https://tourismnotes.com/travel-tourism/>.
2. ANA-MARIA, Ghete. Classification Of Various Forms Of Tourism. *Annals of Faculty of Economics* [online]. 2016 [cit. 2020-12-12]. Dostupné z: <https://ideas.repec.org/a/ora/journal/v1y2016i2p313-319.html>.
3. JALIM, Jeshna. *What are the types of tourism?* [online]. 2018 [cit. 2020-12-12]. Dostupné z: <https://www.quora.com/What-are-the-types-of-tourism>.
4. TUREAC, Cornelia Elena; TURTUREANU, Anca. *Types and Forms of Tourism* [online]. 2008 [cit. 2020-12-12]. Dostupné z: <http://www.journals.univ-danubius.ro/index.php/oeconomica/article/view/60/56>.
5. VANICEK, Jiri. *Position of Prague on the International Tourism Market* [online]. 2020 [cit. 2020-12-14]. Dostupné z DOI: /10.18775/ijied.1849-7551-7020.2015.56.2003.
6. NOVOTNA, Marketa; KUNC, Josef; KUBICKOVA, Helena. Special Interest Tourism in the Czech Republic: Introduction and Overview. *Czech Journal of Tourism* [online]. 2019, roč. 8, s. 49–63 [cit. 2020-12-14]. Dostupné z DOI: 10.2478/cjot-2019-0004.
7. *Praha má novou strategii cestovního ruchu: více domácích turistů a potřeby města na prvním místě* [online]. Institut plánování a rozvoje hl. m. Prahy, 2020 [cit. 2020-12-14]. Dostupné z: <https://www.iprpraha.cz/clanek/2057/praha-ma-novou-strategii-cestovniho-ruchu-vice-domacich-turistu-a-potreby-mesta-na-prvnim-miste>.
8. *What does a tour guide do?* [online]. Career Explorer, 2012 [cit. 2020-12-14]. Dostupné z: <https://www.careerexplorer.com/careers/tour-guide/>.
9. DZIENDZIURA, Michal. *Prague Old Town* [online]. 2020 [cit. 2020-12-14]. Dostupné z: <https://freewalkingtourprague.eu/en/tours/free/prague-old-town>.

10. *Prague Tours* [online]. Tours By Locals, 2020 [cit. 2020-12-14]. Dostupné z: <https://www.toursbylocals.com/Prague-Tours>.
11. PRIYANKA. *5 Steps To Planning A Walking Tour* [online]. 2017 [cit. 2020-12-14]. Dostupné z: <https://medium.com/altertrips/5-steps-to-planning-a-walking-tour-1455fe16f2cf>.
12. *Discover the beauty of Prague* [online]. Smart Guide, 2020 [cit. 2020-12-14]. Dostupné z: <https://smart-guide.org/destinations/en/prague/>.
13. ABBAS, Siraj. *History and future of chatbots* [online]. 2017 [cit. 2021-01-02]. Dostupné z: <https://medium.com/@sirajea/history-and-future-of-chatbots-a1c2521f56e7>.
14. *Definition from Techopedia - What does Chatbot mean?* [online]. Techopedia, 2019 [cit. 2021-01-02]. Dostupné z: <https://www.techopedia.com/definition/16366/chatbot>.
15. ZEMČÍK, Tomáš. *A Brief History of Chatbots* [online]. 2019 [cit. 2021-01-02]. ISSN 2475-8841. Dostupné z DOI: 10.12783/dtcse/aicae2019/31439.
16. DSOUZA, Jeremy. *History of chatbots: What was, what is and what will be* [online]. 2020 [cit. 2021-01-02]. Dostupné z: <https://www.engati.com/blog/history-of-chatbots>.
17. *A history of chatbots* [online]. Chatbot Pack, 2018 [cit. 2021-01-02]. Dostupné z: <https://www.chatbotpack.com/a-history-of-chatbots/>.
18. *Racter* [online]. Chatbots, 2009 [cit. 2021-01-03]. Dostupné z: <https://www.chatbots.org/chatbot/racter>.
19. B, Sean. *What is the Racter Chatbot?* [online]. 2020 [cit. 2021-01-03]. Dostupné z: [https://yakbots.com/what-is-the-racter-chatbot/?no\\_cache=1](https://yakbots.com/what-is-the-racter-chatbot/?no_cache=1).
20. INA. *The History of Chatbots* [online]. 2017 [cit. 2021-01-03]. Dostupné z: <https://onlim.com/en/the-history-of-chatbots/>.
21. EDWARDS, Nick. *A Brief History of Chatbots* [online]. 2019 [cit. 2021-01-03]. Dostupné z: <https://mhrglobal.com/us/en/blog/a-brief-history-of-chatbots>.
22. NEWMAN, Jared. *How The New, Improved Chatbots Rewrite 50 Years Of Bot History* [online]. 2016 [cit. 2021-01-03]. Dostupné z: <https://www.fastcompany.com/3059439/why-the-new-chatbot-invasion-is-so-different-from-its-predecessors>.
23. FERRUCCI, David; LEVAS, Anthony; BAGCHI, Sugato; GONDEK, David; T.MUELLER, Erik. *Watson: Beyond Jeopardy!* [online]. 2013 [cit. 2021-01-03]. Dostupné z DOI: 10.1016/j.artint.2012.06.009.
24. SOLUTIONS, Artificial. *Types of Chatbot Technology* [online]. 2019 [cit. 2021-01-03]. Dostupné z: <https://medium.com/voice-tech-podcast/types-of-chatbot-technology-72d095df2540>.

25. *Different types of chatbots: Rule-based vs. NLP* [online]. Flow.ai, 2020 [cit. 2021-01-03]. Dostupné z: <https://flow.ai/blog/kb-different-kinds-of-chatbots>.
26. *Types of chatbots* [online]. Fresh Works, 2019 [cit. 2021-01-03]. Dostupné z: <https://www.freshworks.com/live-chat-software/chatbots/three-types-of-chatbots/>.
27. MASON, Mitch. *3 types of business chatbots you can build* [online]. 2017 [cit. 2021-01-03]. Dostupné z: <https://www.ibm.com/blogs/watson/2017/12/3-types-of-business-chatbots-you-can-build/>.
28. MANPREET. *Top 5 Types of Chatbots For Solving Business Problems* [online]. 2018 [cit. 2021-01-03]. Dostupné z: <https://www.hurekatek.com/blog/types-of-chatbots>.
29. *Advantages and Disadvantages of Chatbots You Need to Know* [online]. Avio, 2019 [cit. 2021-01-04]. Dostupné z: <https://www.aivo.co/blog/advantages-and-disadvantages-of-chatbots>.
30. KAUR, Harkiran. *What are the Advantages and Disadvantages of Chatbots in Business?* [online]. 2020 [cit. 2021-01-04]. Dostupné z: <https://www.geeksforgeeks.org/what-are-the-advantages-and-disadvantages-of-chatbots-in-business/>.
31. KUMAR, Saroj. *Top 10 Powerful Platforms for Chatbot Development* [online]. 2018 [cit. 2021-01-05]. Dostupné z: <https://dzone.com/articles/top-10-powerful-platforms-for-chatbot-development>.
32. ANURAG. *Top 5 Chatbot Platforms to Develop Bots – Make a Choice* [online]. 2018 [cit. 2021-01-05]. Dostupné z: <https://www.newgenapps.com/blog/top-5-chatbot-platforms-develop-bots-sdk-solution-tools/>.
33. NATH, Jay. *How to Identify the Right Platform for Your Chatbot* [online]. 2018 [cit. 2021-01-05]. Dostupné z: <https://chatbotsmagazine.com/identifying-the-platform-for-your-chatbot-312e46ef9e8a>.
34. *About Watson Assistant* [online]. IBM, 2020 [cit. 2021-01-05]. Dostupné z: <https://cloud.ibm.com/docs/assistant?topic=assistant-index>.
35. BEKAROO, Girish; SANTOKHEE, Aditya; TECKCHANDANI, Nishant. *AIML and Sequence-to-Sequence Models to Build Artificial Intelligence Chatbots: Insights from a Comparative Analysis* [online]. 2019 [cit. 2021-01-05]. Dostupné z DOI: 10.1007/978-3-030-18240-3\_30.
36. SHEFFIELD, Jenna. *The Ultimate Travel Bot List* [online]. 2016 [cit. 2021-01-05]. Dostupné z: [https://www.30secondstofly.com/ai-software/ultimate-travel-bot-list/#Customer\\_Service\\_Travel\\_Bots](https://www.30secondstofly.com/ai-software/ultimate-travel-bot-list/#Customer_Service_Travel_Bots).
37. MARQUES, Mariana. *Top 3 chatbots that are changing the travel industry* [online]. 2018 [cit. 2021-01-05]. Dostupné z: <https://medium.com/hijiffy/top-3-chatbots-that-are-changing-the-travel-industry-d325082c50b8>.

38. STREET, Tarran. *Introducing Expedia bot for Facebook Messenger* [online]. 2016 [cit. 2021-01-05]. Dostupné z: <https://viewfinder.expedia.com/introducing-expedia-bot-facebook-messenger/>.
39. SAWERS, Paul. *Skyscanner launches Facebook Messenger bot to automate flight search conversationally* [online]. 2016 [cit. 2021-01-05]. Dostupné z: <https://venturebeat.com/2016/05/26/skyscanner-launches-facebook-messenger-bot-to-automate-flight-search-conversationally/>.
40. *Meet Golem: Prague Visitor's handy new Facebook Chatbot* [online]. Prague Resident, 2018 [cit. 2021-01-05]. Dostupné z: <https://www.praguevisitor.eu/meet-golem-prague-visitors-handy-new-facebook-chatbot/>.
41. *Chatbot with exciting routes through Moscow* [online]. The Moscow Agency for Recreation a Tourism, 2018 [cit. 2021-01-05]. Dostupné z: <http://mosroutes.com/page2974878.html>.
42. *Advantages and Disadvantages of JavaScript* [online]. geeksforgeeks.org [cit. 2021-03-06]. Dostupné z: <https://www.geeksforgeeks.org/advantages-and-disadvantages-of-javascript/1>.
43. MATHUR, Rahul. *MEAN vs MERN vs MEVN vs LAMP stack for development* [online] [cit. 2021-03-06]. Dostupné z: <https://www.arkasoftwares.com/blog/mean-vs-mern-vs-mevn-vs-lamp-stack-for-development/>.
44. MAŁEK, Piotr. *Why Use React JS?* [online] [cit. 2021-03-06]. Dostupné z: <https://railsware.com/blog/why-use-react/>.
45. GAWRON, Karolina. *What is Node.js and Why You Should Use It For Enterprise Software?* [online] [cit. 2021-03-10]. Dostupné z: <https://www.monterail.com/blog/nodejs-development-enterprises>.
46. *Express/Node introduction* [online]. MDN Web Docs [cit. 2021-03-10]. Dostupné z: [https://developer.mozilla.org/en-US/docs/Learn/Server-side/Express\\_Nodejs/Introduction](https://developer.mozilla.org/en-US/docs/Learn/Server-side/Express_Nodejs/Introduction).
47. SCHAEFER, Lauren. *NoSQL vs SQL Databases* [online]. mongodb.com [cit. 2021-03-10]. Dostupné z: <https://www.mongodb.com/nosql-explained/nosql-vs-sql>.
48. *Places API - Overview* [online]. Google Maps Platform [cit. 2021-04-15]. Dostupné z: <https://developers.google.com/maps/documentation/places/web-service/overview>.
49. *Getting directions through the Directions API* [online]. Google Maps Platform [cit. 2021-04-15]. Dostupné z: <https://developers.google.com/maps/documentation/directions/get-directions>.
50. *API:Main page* [online]. Media Wiki [cit. 2021-04-15]. Dostupné z: [https://www.mediawiki.org/wiki/API:Main\\_page](https://www.mediawiki.org/wiki/API:Main_page).







## Příloha B

### Seznam použitých technologií, knihoven a nástrojů

- Technologie
  - NPM 6.14.4
  - Node.js 14.0.0
- Backend knihovny
  - ibm-watson 5.1.0
  - mongoose 5.12.3
  - node-fetch 2.6.1
  - jsonwebtoken 8.5.1
  - babel-jest 26.6.3
  - bcryptjs 2.4.3
  - concurrently 6.0.2
  - cross-fetch 3.1.3-alpha.6
  - dotenv 6.2.0
  - express 4.17.1
  - request 2.88.2
  - babel-eslint 10.1.0
  - casperjs 1.1.4
  - codecov 3.6.1
  - eslint 7.11.0
  - nodemon 2.0.7
  - phantomjs-prebuilt 2.1.16
  - supertest 4.0.2
- Frontend knihovny
  - react 17.0.2

- axios 0.21.1
- bootstrap 4.6.0
- react-bootstrap 1.5.2
- react-dom 17.0.2
- react-icons 4.2.0
- react-redux 7.2.3
- react-router-dom 5.2.0
- react-scripts 4.0.3
- reactstrap 8.9.0
- redux 4.0.5
- redux-thunk 2.3.0
  
- Nástroje
  - Visual Studio Code 1.56.2
  - MongoDB Compass 1.26.1
  - Postman 8.4.0

## Příloha C

### Instalace a spuštění

V této příloze popisuji, jak nainstalovat potřebné běhové prostředí a spustit aplikaci.

#### C.1 Příprava prostředí

Pro spuštění backendu a frontendu je nutné nainstalovat Node Package Manager (NPM). V aplikaci byla použita verze 6.14.4. NPM jde nainstalovat spolu s Node.js na stránkách <https://nodejs.org/en/>. V aplikaci byla použita verze Node.js 14.0.0. Verze dalších použitých knihoven a nástrojů najdete v příloze B.

##### C.1.1 Databáze MongoDB

Tady popisuji jak se připojit cloudovou databázovou službu MongoDB Atlas. Musíte se vytvářet účet na MongoDb Atlas<sup>1</sup>. Pak musíte si vytvořit nový cluster. Dále je nutné v tom clusteru přejít na složku "connect" a tam najdete "connect your application", kde bude "connection string", kterou musíte přidat do .env souboru jako proměnnou s názvem MONGODB\_URI viz C.1.3.

##### C.1.2 IBM Watson Assistant

Vytvořte si účet na IBM Cloud. Na platformě IBM Watson Assistant vytvořte dialog. Stáhněte si soubor tour-guide-chatbot.json, který najdete na githubu <https://github.com/shamiruf/shamiruf-thesis>. V instanci Watson Assistant klikněte na ikonu "Import". Klikněte na "Choose a file" a poté vyberte soubor tour-guide-chatbot.json, který jste si stáhli dříve a dokončete kliknutím na "Import". Přejděte na nastavení svého dialogu. Zde najdete požadovaná service credentials, a to jsou ASSISTANT\_ID, ASSISTANT\_IAM\_APIKEY, ASSISTANT\_URL, ASSISTANT\_IAM\_NAME. Ty data potřebujete pro spuštění aplikaci.

---

<sup>1</sup><https://www.mongodb.com/cloud/atlas>

### ■ C.1.3 Soubor s požadovanými údaji

Vytvořte soubor `.env` a tam přidejte následující data:

- `MONGODB_URI`,
- `ASSISTANT_ID`,
- `ASSISTANT_IAM_APIKEY`,
- `ASSISTANT_URL`,
- `ASSISTANT_IAM_NAME`.

## ■ C.2 Spuštění aplikace

1. Přejděte do složky `client` a nainstalujte dependencies pro frontend pomocí příkazového řádku **`npm install`**.
2. Přejděte do `root directory` a nainstalujte dependencies pro backend pomocí příkazového řádku **`npm install`**.
3. Backend a frontend spustíte pomocí příkazového řádku **`npm run dev`** z `root directory`.

Backend běží na `http://localhost:5000`. Frontend běží na `http://localhost:3000`.  
Kód je dostupný online na <https://github.com/shamiruf/shamiruf-thesis>.