

Multimodální dopravní plánovač Jihomoravského kraje

Jan Václavík*

Abstrakt

Cílem této práce je implementace webové aplikace, která kombinuje automobilovou a hromadnou dopravu pomocí přestupních bodů, kde dochází k přechodu z auta na hromadnou dopravu. Aplikace je vhodná zejména pro uživatele, kteří denně dojíždějí do práce a chtějí spojit rychlost a flexibilitu auta s pohodlím a nízkými emisemi hromadné dopravy, případně bydlí v místě, kde jsou spoje hromadné dopravy málo frekventované.

*xvcla35@stud.fit.vutbr.cz, *Fakulta informačních technologií, Vysoké učení technické v Brně*

1. Úvod

Poptávka po mobilitě neustále roste a s tím i počet automobilů, které jezdí po městě. To způsobuje dopravní zácpy, nehody a neprospívá to životnímu prostředí. Jednou z možností je použití hromadné dopravy, které je ale v mnoha případech pomalé, má zpoždění a na některých vesnicích nemusí mít frekventované spoje.

Tato práce přináší webovou aplikaci, která kombinuje automobilovou a hromadnou dopravu, a zachovává tak rychlost a flexibilitu auta a zároveň pohodlí a ekologičnost hromadné dopravy. Tyto dva způsoby dopravy spojuje pomocí přestupních bodů, kterými jsou železniční stanice a autobusová nádraží.

Existující řešení jako například Mapy Google umožňují kombinaci auta a veřejné dopravy, ale uživatel musí naplánovat obě trasy samostatně a rovněž musí znát místo, kde auto zaparkuje.

V této práci si uživatel může sám zvolit, který přestupní bod použije, nebo nechá vše na aplikaci a ta za něj nevhodnější přestupní bod vybere.

2. Architektura

Obrázek 1 ukazuje architekturu aplikace, která je rozdělena na klientskou část, která je implementována pomocí knihovny React, serverovou část implementovanou v jazyce TypeScript s běhovým prostředím Deno a datovou vrstvou. Klient přijímá od serveru vypočtené trasy a přestupní body a ty poté zobrazuje na mapě. Klient serveru poté posílá požadavky na trasy. Tato komunikace probíhá pomocí REST API.

Na serveru je poté prováděna hlavní část výpočtu. Pro získání jednotlivých posloupností spojů komunikuje s aplikací OpenTripPlanner 2¹ (OTP) pomocí GraphQL. Pro získání zpoždění a tras pro jednotlivé spoje pak pomocí REST API komunikuje s aplikací Lissy.

Přestupní body jsou periodicky získávány serverem z datové vrstvy, která sestává z GTFS souborů² pro daný týden a skriptu, který z nich tyto body extrahuje. Kromě základních informací o přestupních bodech jako jsou jejich jména a souřadnice, je uchovávána i hodnota říkající, kde se v jejím okolí nachází veřejné parkoviště. Tyto informace jsou získány pomocí služby Overpass, která dokáže získávat data z OpenStreetMap.

3. Funkce a implementace

Aplikace kromě jiného zaujme i tím, že je aplikovatelná na jakýkoliv region, který obsahuje GTFS soubory. Zároveň nabízí i přehledné mobilní zobrazení, které ukazuje **Obrázek 2**. Avšak nabízí i další funkce, které uživatelům zpřijemňují její používání.

3.1 Zobrazení tras a zpoždění

Součástí každého spoje je jeho trasa a rovněž předpokládané zpoždění. Jelikož GTFS soubory pro Jihomoravský kraj neobsahují trasy linek, bylo toto implementováno s použitím aplikace Lissy³ Juraje Lazúra

¹<https://docs.opentripplanner.org/en/latest/>.

²Tyto soubory jsou získávány z data.brno.cz.

³<https://pclazur.fit.vutbr.cz/lissy/>.

[1], která tyto trasy uchovává. Zobrazení tras a zároveň celé mapy na klientské straně obstarává knihovna React Leaflet a ukazuje ji [Obrázek 3](#).

Lissy rovněž shromažďuje informace o zpožděních pro jednotlivé linky za minulé dny a lze tedy pro každý spoj zobrazit, jak se v jeho zpoždění v minulosti vyvíjelo. To funguje i v případě této práce, kdy uživatel vidí pouze průměrné zpoždění za dané dny, ale může se podívat i na tabulku s vývojem zpoždění za minulé dny.

3.2 Funkce vyzvednutí

Součástí aplikace je rovněž funkce vyzvednutí, která lze použít v případě, že bude použito auto na začátku i na konci cesty. Uživatel si může zvolit bod vyzvednutí, odkud bude pokračovat autem do cílové destinace. Může se jednat o taxi nebo o příbuzného, který daného uživatele vyzvedne například na nádraží.

3.3 Funkce návratu zpět

Jestliže uživatelé cestují nějakou formou do cílové destinace, tak je velmi pravděpodobné, že pojedou i zpět. Aplikace umožňuje vyplnit datum a čas návratu přímo během plánování cesty tam. Výsledkem jsou poté trasy, které vedou z počátečního bodu do cíle, ale i naopak.

3.4 Hledání vhodného přestupního bodu

Celý problém, který je třeba řešit, tkví v nalezení přestupního bodu, který nabídne nejlepší trasu. Pro menší počet těchto bodů je možné vyzkoušet všechny, ale pro více bodů je to obtížnější, neboť výpočet trvá dlouhou dobu.

Proto jsou kandidátní přestupní body vybírány pomocí shlukovacího algoritmu [2], který přestupní body rozdělí do daného počtu shluků a jako kandidátní přestupní bod je vybrán ten, který se nachází nejbližší středu daného shluku.

Přestupní bod ze seznamu si však může uživatel i sám zvolit. V takovém případě jsou veškeré výpočty přeskočeny a plánování probíhá pouze přes tento bod.

3.5 Ohodnocení řešení

Po nalezení řešení pro kandidátní přestupní body dochází k jejich ohodnocení na základě celkového času cesty, zpoždění, emisí a přestupů. Nejprve dojde k odstranění řešení, která nejsou Pareto-optimální podle těchto kritérií, protože řešení, která jsou ve všech kritériích horší než jiná, nejsou důležitá.

Poté je provedena *Min-Max* normalizace, která všechny hodnoty převede do intervalu $< 0; 1 >$ kvůli tomu, že každé kritérium má jiný rozsah hodnot.

Skóre daného řešení je poté vypočítáno jako

$$s = \sum_{i=1}^N w_i * norm_i,$$

kde w_i je váha kritéria i a $norm_i$ je normalizovaná hodnota kritéria i a platí, že čím menší je hodnota s , tím je trasa lepší. Důležitost kritérií byla určena podle uživatelského dotazníku, který se na toto ptal. Váhy poté byly zkonstruovány pomocí metody AHP (*Analytical Hierarchy Process*), pomocí které lze určit důležitost kritéria vůči jiným kritériím [3].

Nejlepších 10 řešení pak putuje k uživateli v pořadí od nejlepšího po nejhorší. Uživatel si však může řešení seřadit buď podle rychlosti, nebo spotřeby.

4. Závěr

Výsledkem této práce je webová aplikace, která pomocí přestupních bodů dokáže kombinovat automobilovou a hromadnou dopravu a dokáže zobrazit trasu včetně zpoždění jednotlivých spojů. Na základě uživatelského testování vyplynula možná rozšíření jako je například výpočet ceny přepravy nebo získávání informací o kolonách pomocí Waze.

Poděkování

Rád bych poděkoval mému vedoucímu Ing. Jiřímu Hynkovi, Ph.D. za cenné rady a podporu během tvorby této diplomové práce.

Literatura

- [1] Ing. Juraj Lazúr. Analýza a vizualizace dat hromadné dopravy města brna. Diplomová práce, Vysoké učení technické v Brně, Fakulta informačních technologií, Ústav informačních systémů, Brno, 2023.
- [2] Mohiuddin Ahmed, Raihan Seraj, and Syed Mohammed Shamsul Islam. The k-means algorithm: A comprehensive survey and performance evaluation. *Electronics*, 9(8):1295, 2020.
- [3] Michael Bruhn Barfod and Steen Leleur. Multi-criteria decision analysis for use in transport decision making. 2014.