

Diplomová práce



České
vysoké
učení technické
v Praze

F3

Fakulta elektrotechnická
Katedra počítačů

Systém pro aktivitně-cestovní průzkum mobility

Bc. Tom Nováček
Otevřená informatika

Květen 2017
Vedoucí práce: RNDr. Michal Čertický, Ph.D.

Poděkování / Prohlášení

Rád bych touto cestou poděkoval Michalovi Čertickému za cenné rady a vstřícnost při vedení práce. Stejně tak bych chtěl poděkovat své rodině za podporu při studiu a v neposlední řadě participantům průzkumu za poskytnuté údaje.

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

V Praze dne 26. 5. 2017

.....

Abstrakt / Abstract

Dopravní plánování hraje významnou roli při utváření měst po celém světě. S rostoucím přepravním výkonem je stále složitější zajistit potřebné kapacity dopravní infrastruktury pro udržitelný ekonomický rozvoj oblastí. Důležitým podkladem pro pochopení potřeb a návyků obyvatel se staly průzkumy přepravního chování uskutečňované formou aktivitně-cestovních deníků. Aktivitní modely dopravy pak těchto informací využívají k lepší predikci dopadu případných změn a omezení. Získání takových dat je ale poměrně náročné a vyžaduje od respondentů aktivní přístup po delší časový interval. Tato práce se zabývá tvorbou uceleného systému, který by měl celý proces výrazně zjednodušit a přispět tak k celkové kvalitě výsledných dat. K tomuto účelu jsme poskytli mobilní aplikaci pro automatizovaný sběr cestovních dat a dále navrhli a implementovali algoritmy pro detekci dopravních módů i konkrétních aktivit.

Klíčová slova: Aktivitně-cestovní průzkum, cestovní deník, dopravní model, detekce dopravních módů

Transportation planning plays an important role in shaping cities all around the world. Due to increasing traffic volumes, it is more difficult to provide the necessary transport capacity for the sustainable economic development of the urban areas. Travel-activity diary surveys provide important background information for understanding the needs and habits of the population and activity-based models use this information for formulating city's transport policy. Obtaining such data is quite demanding and requires respondents to participate in the survey over a long period of time. This thesis deals with the development of a system which should significantly simplify the process of collecting and evaluating travel data and thus contribute to the quality of the survey results. Therefore, to address this issue, we proposed a mobile application and algorithms for transportation mode and activity detection.

Keywords: Activity-based survey, travel diary, transportation model, transportation mode detection

Title translation: Activity-Based Travel Survey System

Obsah /

1 Úvod	1
1.1 Cíl práce.....	2
1.2 Struktura práce	2
2 Popis problematiky.....	3
2.1 Dopravní průzkumy.....	3
2.2 Čtyřfázový dopravní model.....	4
2.2.1 Trip generation	5
2.2.2 Trip distribution	5
2.2.3 Modal split	5
2.2.4 Assignment.....	5
2.3 Aktivitně-cestovní model	6
2.3.1 Cestovní deníky	7
2.3.2 Modelování a validace aktivitních systémů	9
3 Analýza dostupných řešení	10
3.1 MEILI Travel Diary	10
3.2 Future Mobility Sensing	10
3.3 Související práce	11
3.3.1 Detekce dopravních módů za využití GPS....	11
3.3.2 Detekce dopravních módu za využití akcelerometru.....	13
3.3.3 Detekce aktivit	14
4 Návrh řešení	16
4.1 General Transit Feed Specification	16
4.1.1 Formát GTFS dat	17
4.1.2 Využití GTFS dat	18
4.2 Detekce dopravních módů	18
4.3 Detekce aktivit	21
4.4 Návrh architektury systému ...	21
5 Mobilní aplikace	23
5.1 Analýza požadavků a rizik	23
5.1.1 Funkční požadavky	23
5.1.2 Obecné požadavky	23
5.1.3 Analýza problémů a rizik.....	24
5.2 Popis architektury	24
5.2.1 Core layer	24
5.2.2 Client layer.....	25
5.2.3 Tracking layer.....	25
5.3 Implementace	25
5.3.1 Vybrané komponenty systému Android	25
5.3.2 Sledování polohy	26
5.3.3 Detekce dopravních módů	28
5.3.4 Odesílání dat na server ..	29
5.3.5 Automatické aktualizace	30
5.4 Zhodnocení a testování aplikace.....	31
6 Server	33
6.1 Analýza požadavků a rizik	33
6.1.1 Aplikační server	33
6.2 Popis architektury	34
6.2.1 Webový server	34
6.2.2 Databáze	35
6.2.3 Procesy pro vyhodnocení.....	36
6.3 Implementace	36
6.3.1 Komunikace klient - server	36
6.3.2 Detekce dopravních módů	37
6.3.3 Klasifikace aktivit.....	38
7 Vyhodnocení	39
8 Závěr	41
Literatura	43
A Seznam použitých zkratek	47

Tabulky /

2.1. Ukázka možné podoby za- znamenaných dat.....	9
3.1. Identifikované hodnoty jed- notlivých dopravních módu	12
5.1. Tabulka parametrů testova- cích zařízení	31
7.1. Tabulka distribuce testova- cích test.....	39

Kapitola 1

Úvod

Doprava hraje důležitou roli v každodenním životě většiny lidí, především pak obyvatel velkých měst. S tím, jak se naše společnost vyvíjí, stává se dojíždění za prací stále běžnější, zároveň však roste požadavek na zkracování doby strávené v dopravních prostředcích. Ze statistik vyplývá, že ačkoliv se počet přepravených osob výrazně nemění, lidé cestují častěji a na větší vzdálenosti¹. Například přepravní výkon, tedy jeden z parametrů hodnotící celkovou vytíženosť a efektivitu osobní dopravy, vzrostl v České republice mezi roky 2010 - 2015 o více než 10 % na celkových 113 miliard osobo-kilometrů².

Tento trend do jisté míry souvisí také s tím, jak se mění pohyby obyvatel mezi velkými městskými regiony a jejich nejbližšími okresy. Od 90.let minulého století dominuje ve vnitřní migraci (tedy stěhování v rámci jednoho státu - konkrétně ČR) proud směrem z center měst do periferií, nejvýraznějším příkladem u nás je okolí hlavního města Prahy. V důsledku procesu nazývaného jako suburbanizace vznikají v zázemí města na něm ekonomicky závislé nové rezidenční a komerční oblasti. Většina osob stěhujících se do těchto lokalit si však z důvodu atraktivnosti trhu práce ponechává své pracovní místo v Praze. Podíl takto dojíždějících obyvatel představoval v roce 2016 okolo 20 % celkové zaměstnanosti hlavního města, čímž vzniká velký tlak na dopravní infrastrukturu a systém městské hromadné dopravy [1]. Samosprávy jsou také nuceny řešit navazující problémy s parkováním a ekologickou zátěží. Pokud zůstaneme u hlavního města, výdaje na dopravu přesahovaly v roce 2016 částku 17 mld. Kč³, z čehož nejvyšší položkou jsou kompenzace na provoz Dopravnímu podniku hl. m. Prahy (DPP) - cca 13 mld. Kč. Tento podnik je navíc s téměř 11 tisíci zaměstnanci jedním z největších zaměstnavatelů v regionu. Doprava se tak proto v posledních letech stává stále častějším tématem společenských i politických diskuzí.

Pro záruku udržitelného rozvoje mobility je nezbytné zajistit optimální sladění přepravní nabídky s poptávkou. Důležitou roli při zjišťování preferencí obyvatel tak tvoří dopravní průzkumy, které slouží jako podklad pro plánování dopravy na daném území. Jejich dopad se liší podle účelu zaměření, některé se mohou soustředit na kvalitu poskytovaných služeb (dotazníkové průzkumy), jiné zase měřit intenzitu dopravních proudů (např. pomocí úsekového scítání). Klíčovým zdrojem dat pro efektivní analýzu se pak stávají průzkumy dopravního chování, které sledují více parametrů osobní přepravy. Kromě délky a trvání jednotlivých cest to může být například i její účel nebo zvolený dopravní prostředek. Pokud jsou navíc tyto informace doplněny o základní demografické údaje, stávají se velmi cennými při odhadu potenciálních potřeb a především možností cestujících. Nejčastěji jsou tyto průzkumy prováděny formou tzv. cestovních deníků,

¹ <https://www.mdcr.cz/Statistiky/Souhrnné-ukazatele/Osobní-doprava>

² Osobo-kilometr (osbkm) je jednotka přepravního výkonu a znamená přepravu jedné osoby na vzdálenost jednoho kilometru. Vypočítá se jako součin vzdálenosti, kterou ujede daný dopravní prostředek, a počtu přepravených osob. Přepravní výkon automobilu, který s jedním cestujícím ujede 100 km, je stejný jako výkon autobusu, který ujede se 100 cestujícími 1 km.

³ http://www.praha.eu/jnp/cz/o_meste/finance/rozpočet/rozpočet_na_rok_2016

jedná se však o poměrně náročný způsob získávání dat. Aby měl takový průzkum dostatečnou vypovídající hodnotu, musí být totiž prováděn v rámci delšího časového intervalu (např. týden) a ze své podstaty vyžadující aktivní přístup zúčastněných participantů může být náchylný na nepřesné výsledky. V neposlední řadě je třeba motivovat respondenty k vyplnění všech potřebných údajů, což ještě navyšuje pracnost příprav a organizace tohoto druhu výzkumu [2].

Pokrok v rozvoji informačních technologií, především pak v oblasti mobilních zařízení, dnes však umožňuje tvorbu inteligentních nástrojů, které by tento postup mohly do značné míry zjednodušit a umožnit tak efektivnější pokrytí většího množství cestujících. Městům by tak pomohly lépe pochopit požadavky a návyky obyvatel, což je základ pro řešení problému s plánováním dopravy.

1.1 Cíl práce

Cílem práce je poskytnout ucelený nástroj pro sběr a vyhodnocení dat průzkumu dopravního chování obyvatel prováděného formou cestovních deníků. Dosažení tohoto cíle dnes umožňuje rostoucí zájem o chytré mobilní zařízení (tzv. smartphones), která nabízejí nové možnosti v oblasti interakce s uživatelem i v lokalizaci jeho polohy. Kombinace technologií GPS a Wi-Fi spolu s příjemem signálu mobilního operátora zajišťuje velmi přesné určení pozice i v poměrně husté městské zástavbě [3]. Tato skutečnost je zásadním předpokladem toho, aby participant průzkumu nemusel uskutečněné trasy naznamenávat sám.

Dalším záměrem je nabídnout systém automatické detekce dopravních módů. Pro úspěšnou analýzu je totiž nutné jednoznačně určit, zdali daná jízda byla uskutečněna např. automobilem nebo za využití městské hromadné dopravy (MHD). Důležitou součástí systému by také mělo být propojení demografických údajů s účelem cesty, tedy aktivitou, která je pro ní bezprostředním podnětem.

Systém se proto bude skládat z mobilní aplikace zajišťující sběr dat a serverové části vyhodnocující naznamenaná data. Kombinace výše zmíněných faktorů sníží nutnost uživatelské interakce na minimum, což zjednoduší nasazení takových dopravních průzkumů a přispěje i ke snížení jejich celkových nákladů.

1.2 Struktura práce

V úvodních kapitolách (2 a 3) se seznámíme se základními pojmy a modely z oblasti dopravního plánování a dopravních průzkumů. Na základě porovnání souvisejících prací a analyzování současných možností budou diskutovány volby vedoucí k výběru daného řešení. Kapitoly 4 až 6 se poté zabývají návrhem a implementačními detaily jednotlivých částí výsledného systému. V závěrečných kapitolách jsou prezentovány výsledné statistiky a data spolu s možnostmi dalšího rozšíření systému.

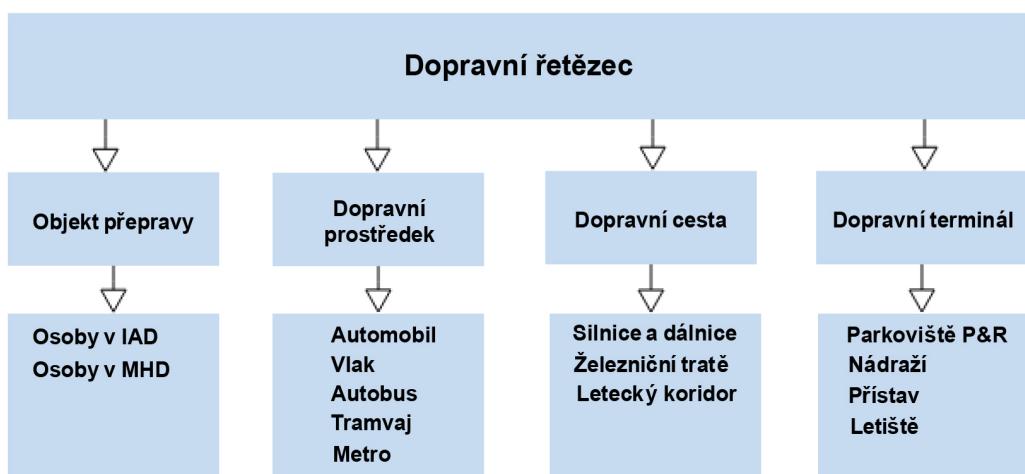
Kapitola 2

Popis problematiky

2.1 Dopravní průzkumy

Abychom mohli vytvořit efektivní systém pro potřeby dopravních průzkumů, musíme nejprve pochopit proces jejich tvorby, metodiky vyhodnocování a také jejich dělení. Je nutné na dopravní problematiku nahlížet ze širšího kontextu urbanismu, především ve spojitosti s územním plánováním. Pokud se totiž na dopravu podíváme čistě z pohledu ekonomické poptávky, jde ve většině případů, tedy až na situace, kdy se cestování děje z vlastní vůle (např. potěšení z cestování) o tzv. odvozenou poptávkou [4]. Jejím účelem je zprostředkování jiné služby, a poptávka po ní je tak logicky odvozena od poptávky po dané službě. Chceme-li se zaměřit na oblast osobní dopravy, znamená to, že umožňuje obyvatelům vykonat určitou aktivitu na jiném místě, než je místo jejich současné aktivity [5]. Příčiny vzniku takových aktivit mohou být různé, např. ekonomické (cesta do zaměstnání) nebo společenské (návštěva kulturní akce). Rozvoj dopravy tedy úzce souvisí s rozvojem dané oblasti, což je motivem již zmíněného územního plánování.

Dále bychom měli definovat základní pojmy a popsat tzv. přepravně-dopravní řetězec, jehož charakteristikami se průzkumy zabývají. Vzhledem k účelu a cílům této práce se budeme soustředit pouze na osobní přepravu [6].



Obrázek 2.1. Zobecněný model dopravního řetězce, převzato z [7].

- Objekt přepravy - definuje kategorii cestujícího, spolu s prostředkem tvoří dopravní jednotku
- Dopravní prostředek - definuje dopravní element, který se pohybuje po dopravní cestě
- Dopravní cesta - definuje prostor, ve kterém se pohybují dopravní jednotky
- Dopravní terminál - definuje prostor, kde dochází ke změně druhu dopravy

Následuje seznam pojmu souvisejících s oblastí průzkumů a jejich význam:

- Domácnost - jedná se o jednotlivce nebo skupinu osob, které sdílí společný bytový prostor a obvykle i dopravní prostředky.
- Jednotka průzkumu - jednotlivec nebo domácnost vybraná pro účely šetření.
- Reprezentativní výběr - soubor jednotek, z jehož vlastností můžeme usuzovat na vlastnosti celé populace.
- Dotazníkové šetření - sběr dat pomocí dotazníků
- Aktivita - účel cesty, činnost za účelem uspokojení některé z potřeb (zaměstnání, vzdělání atd.).
- Cesta - změna místa, přemístění od jedné aktivity ke druhé (např. cesta do zaměstnání se zastávkou na nákup se počítá jako dvě samostatné cesty). V některých případech může cesta samotná představovat účel (aktivitu). Jedná se o situace, kdy nevykonáváme aktivitu z vnějších, ale vnitřních důvodů - typicky výlety, projíždky na kole nebo venčení psa.

Každou z výše uvedených kategorií dopravního řetězce můžeme podrobit analýze v samostatném průzkumu. Můžeme se například v rámci dotazníkového šetření zabývat tím, jaký dopravní prostředek cestující preferují, či jak jsou spokojeni s kvalitou služeb MHD. V případě zkoumání dopravních cest a terminálů nás pro změnu bude spíše zajímat propustnost důležitých křižovatek, využití kapacity parkovišť či čekací doby na zastávkách. Všechny tyto způsoby zajišťování podkladů pro modelování dopravního systému se dají rozdělit do dvou skupin:

- Pozorování - prováděno bez vědomí pozorovaného
 - Dopravní sčítání na silnici
 - Sčítání cestujících v MHD (např. pomocí automatických sčítaců)
- Dotazování - vyžaduje aktivní přístup dotazovaného
 - Přepravní průzkumy v terénu
 - Telefonické dotazování (prováděno spíše v historii)
 - Dotazníky pro domácnost
 - Cestovní deník

Většina těchto průzkumů má jedno společné - zachycují současný stav dopravního řetězce. Hlavním úkolem dopravního plánování je však odhadnout vývoj celého systému, především pak dopad případných změn a omezení. Při uvedení takových změn do reality velmi brzy zjistíme, že různé skupiny obyvatel reagují na vzniklou situaci jinak. Co může být pro jednoho důvodem ke změně dopravního prostředku (např. prodloužení doby strávené v koloně, zvýšené ceny pohonných hmot), nemusí pro jiného být podstatné, třeba z důvodu komfortu. Pro identifikaci těchto skupin je proto důležité sledovat jejich dopravní chování s ohledem na demografické a socio-ekonomické ukazatele.

2.2 Čtyřfázový dopravní model

Tradiční přístupy pro plánování dopravy využívaly již od 50. let 20. století prognózy založené na tzv. čtyřfázovém dopravním modelu (z anglicky four-step model). Tento princip chápe přepravu jako výsledný efekt čtyř po sobě jdoucích, matematicky vyjádřených rozhodnutí - jednotlivých fází modelu [8].

- Generování cest (Trip generation)

- Volba cílů cest (Trip distribution)
- Volba dopravního prostředku (Modal split)
- Volba tras (Assignment)

■ 2.2.1 Trip generation

V této fázi je klíčové určení vstupních a cílových přepravních uzelů. Zajímá nás např. počet členů domácnosti, její struktura, dostupnost automobilu a dále charakteristiky jednotlivých území, tedy počet zaměstnanců nebo počet obchodů. Využívá se zde například Huffův model vycházející z gravitačního modelu, přisuzujícího všem přepravám stejnou váhu¹. V této fázi se snažíme najít odpověď na otázku: Kolik cest bude uskutečněno?

■ 2.2.2 Trip distribution

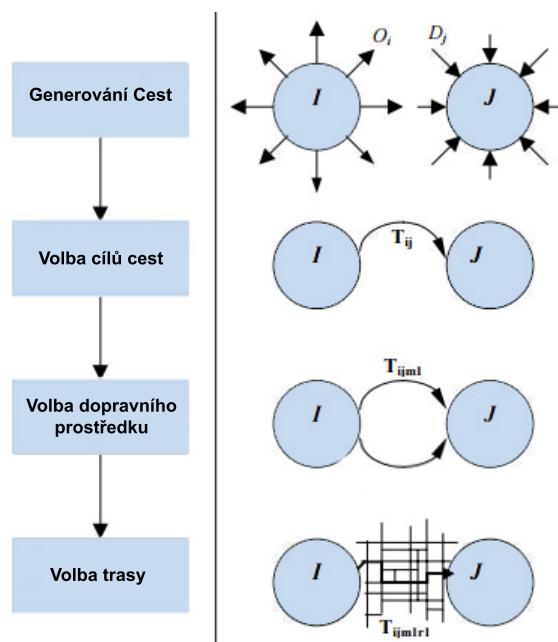
V druhé části tvoříme vazby mezi zdrojovými a cílovými oblastmi, např. formou matici přepravních vztahů. Zajímá nás tedy kam jednotlivé trasy vedou.

■ 2.2.3 Modal split

V další fázi se snažíme modelovat rozhodnutí, vyjádřené např. pravděpodobností, která vedou jednotlivce (či skupinu) k výběru daného prostředku - tedy především čas, náklady a dostupnost MHD.

■ 2.2.4 Assignment

Poslední částí je rozhodnutí o výběru konkrétní trasy pro dosažení cílové destinace. Cestující může například preferovat delší trasu z důvodu přeplněné kapacity na hlavním koridoru. Využívá se například ohodnocení jednotlivých tras pomocí časových nákladů spojených s průjezdem daného úseku.



Obrázek 2.2. Schéma čtyřfázového modelu [9]. Nejprve nalezneme množiny vstupních a výstupních uzelů (O a D). Poté jednotlivé uzly spojíme trasou T_{ij} za využití prostředku m_1 a trasy r_1 .

¹ <https://www.arcgis.com/home/item.html?id=f4769668fc3f486a992955ce55caca18>

Úskalím čtyřfázového modelu je jeho orientace na cestu (trip-based). Tento přístup nenahlíží na dopravu jako na prostředek k naplnění potřeb jedince (cesta do zaměstnání), ale zaměřuje se spíše na dopravní toky jako takové [10].

Pro parametrizaci jednotlivých fází využívá statistických modelů, což s sebou přináší nedostatečné pokrytí některých aspektů dopravního chování. Byl vyvinut jako podklad k vyhodnocování finančně náročných investičních projektů v oblasti infrastruktury, a to především v poválečném období, kdy nárůst přepravních objemů souvisejel s růstem počtu obyvatel [11].

Omezené jsou proto jeho možnosti při odhadování dopadu jednotlivých změn ve stávajícím dopravním systému, tedy například zavedení mýtného či omezení počtu parkovacích míst, naopak může být dobré využít při plánování o navýšení kapacity některých tras dopravní sítě [12].

Zaměřením se pouze na jednotlivé cesty logicky nemůžeme získat znalost o prostorových a časových vztazích mezi nimi, tedy ani hledat důvody například jejich návaznosti.

Analýzy založené na průzkumech dopravního chování v současnosti představují nebytnost při utváření dopravních strategií, ať již se jedná o rozhodnutí o výstavbě nové infrastruktury či plánování veřejné dopravy. Znalost potřeb a důvodů, které cestující ovlivňují při rozhodování o volbě jednotlivých dopravních prostředců jsou zásadní pro parametrizaci celkového modelu přepravní poptávky. V posledních letech proto vzrůstá poptávka po modelech založených na sledování tohoto přepravního chování, konkrétně na tzv. aktivitně založených modelech poptávky, kde aktivitou je chápán zásadní podnět pro uskutečnění cesty.

2.3 Aktivitně-cestovní model

Základní myšlenkou stojící za aktivitně-cestovním modelováním je skutečnost, že přepravní poptávka je odvozena od poptávky po jednotlivých aktivitách, tedy činností, které jednotlivec a domácnosti potřebují (nebo si přejí) uskutečnit. Hlavní rozdíl oproti tradičnímu (čtyřfázovému) modelu spočívá v tom, že aktivitně založený model hledá vzájemné spojitosti mezi několika aspekty daných profilů činností. Tyto aspekty bychom mohli zobecnit na místo, kde je aktivita vykonávána, dále denní dobu, ve které se uskutečňuje, délku trvání a zvolený dopravní prostředek [13].

Počátky tohoto přístupu sahají do 60. let 20. století, kdy se Francis Stuart Chapin zabýval výzkumem činností obyvatel měst [14]. Jeho výsledky však ještě ignorovaly prostorový kontext aktivit a nezabývaly se vztahem mezi cestováním a související aktivitou. Popré byly tyto vazby zahrnující čas a místo prezentovány v ucelené studii až v roce 1981 na konferenci „Travel Demand Analysis: Activity-based and Other New Approaches“.

Větší popularitě se aktivitně orientovaný přístup začíná těšit až od 90. let minulého století, ke čemuž vedly faktory jako rostoucí tlak na zvýšenou přesnost predikce účinnosti různých změn a omezení v dopravě, dále pak uvědomování si nedostatků tradičního modelu a v neposlední řadě rozvoj analytických nástrojů a výpočetní kapacity umožňující lepší sběr dat [15]. V poslední dekádě také roste využití těchto systémů pro modelování emisí či kvality vzduchu [16][17].

Tento způsob nahrazuje konvenční přístup ke studiu cestovního chování ucelenějším, takzvaným holistickým rámcem¹, ve kterém jsou v cestování analyzovány určité vícedenní vzory chování vycházející z rozdílů v životním stylu a účasti v různých druzích

¹ Holismus (z řeckého *holos*, celek) je filosofický názor nebo směr, který zdůrazňuje, že všechny vlastnosti nějakého systému nelze určit nebo vysvětlit pouze zkoumáním jeho částí. Zdroj: wikipedia.org

aktivit [18]. Aktivitní modely tedy explicitně zkoumají základní chování s ohledem na socio-ekonomické aspekty, a proto dokáží na rozdíl od tradičních modelů reflektovat individuální reakce na vzniklé změny.

Zavádějí do systému také zpětnou vazbu a především souvislosti mezi jednotlivými uskutečněnými cestami (na rozdíl od tradičního modelu, který cesty hodnotí nezávisle a jednotlivě).

Uvažujme například situaci jedince, který jezdí každý den do práce sám autem a po cestě domů se pravidelně stavuje na nákup zboží v obchodě. Volba dopravního prostředku pro obě cesty však v tomto případě není nezávislá. V případě potenciálních změn či omezení v dopravě tak tento cestující nemusí změnit dopravní prostředek, zkrátka protože je pro něj večerní cesta s nákupem pohodlnější vozem. Tradiční, cestovně orientovaný model by díky neznalosti této vazby mohl chybně předpovědět, že jedinec začne jezdit např. hromadnou dopravou [15].

Získání takto komplexních dat je ze své podstaty samozřejmě náročnější, a vyžaduje od participantů sdílení většího množství informací. Osvědčenou metodou pro toto dotazníkové šetření se staly tzv. cestovní deníky.

■ 2.3.1 Cestovní deníky

Cestovní deníky jsou standardním nástrojem pro sběr dat v rámci dopravních průzkumů. Využívají se jak pro longitudinální, tak průřezové studie¹ [19]. Jejich podstatou je zaznamenávání všech důležitých údajů v souvislosti s dopravou, a to na denní bázi. Pokud jsou využívány v rámci aktivitně-cestovního průzkumu, mluvíme pak spíše o aktivitně-cestovních denících, jejichž klíčovou součástí jsou rovněž relevantní socio-ekonomické a demografické charakteristiky [20].

Způsob jejich provádění se historicky využíval a první využití se datuje do 50. let minulého století, především ve velkých městských oblastech Spojených států amerických. Oslovaný byly zpravidla celé domácnosti, a to písemnou formou či telefonicky. V závislosti na zvoleném přístupu se pak lišila celá procedura. Deníky mohly být spolu s instrukcemi pro vyplnění zaslány poštou po prvním úspěšném kontaktu (PAPI)², případně s nástupem počítačů retrospektivně vyplňovány podle připraveného scénáře přes telefon (CATI)³. Obě tyto metody mají svá úskalí a hrozí u nich riziko nepřesných údajů, například při určování lokace prováděné aktivity (lidé si nemusí pamatovat název ulice či přesnou hodinu). Telefonické oslovení sice do jisté míry zvyšuje procento úspěšně dokončených průzkumů, zároveň ale zvyšuje jejich celkové náklady.

Tyto důvody vedly k vytvoření ustálených metodik, které snižují potřebu velkého procentuálního zastoupení domácností [21]. Posledním způsobem zaznamenávání těchto dat (stále však ještě vyžadující aktivní účast) je tzv. metoda CASI⁴, ve které participant zapisuje údaje přímo do počítače, případně mobilu.

Množství požadovaných informací v cestovních denících je rozdílné podle účelu daného průzkumu. Pro tradiční modely dopravy (trip-based) jsou to především lokační údaje o začátku a konci cesty, doba jejího trvání atd. Aktivitně orientované modely navíc k těmto informacím přidávají již zmíněné socio-ekonomické souvislosti.

Dopad cestovních deníků na aktivitní model dopravy závisí na tom, do jaké míry jsou schopny zaznamenat návyky a stereotypy jednotlivých účastníků. Zjišťování těchto informací však prodlužuje nezbytnou dobu pro účast v průzkumu i na několik týdnů.

¹ Longitudinální výzkum zahrnuje studování stejné skupiny po delší časové období, průřezový pak v konkrétní časový okamžik

² Z anglického paper-and-pencil interview

³ computer-assisted telephone interviews

⁴ computer-assisted-self-interview

Cesta <u>Jaký dopravní prostředek jste využil(a) pro přepravu za aktivitou?</u> <u>Jestli jste využil(a) auto/motorcykl, byl iste:</u> Řidič <input type="checkbox"/> Spoluždec <input type="checkbox"/> <u>Kolik osob bylo s Vámi se vozidle</u> <u>Kolik z těchto osob bylo členů Vaší domácnosti</u> <u>Bylo toto vozidlo z Vaší domácnosti:</u> Ano <input type="checkbox"/> Ne <input type="checkbox"/>	Aktivita <u>Čas přjezdu/příchodu na místo aktivity:</u> <u>Typ aktivity:</u> Práce / Vzdělání / Volnočasová aktivita / Cesta domů <u>Adresa místa dané aktivity:</u> <u>Čas odchodu z místa aktivity</u>
--	--

Obrázek 2.3. Zjednodušená ukázka papírové formy cestovního deníku.

Tento fakt může od vyplňování deníků odradit velkou část cestujících, především pak ty, kteří vykonávají větší počet aktivit a s tím spojených cest. Chybějící údaje od těchto potenciálních participantů přitom výrazně snižují validitu celého průzkumu [22].

Nezřídka proto musí být motivováni finančně, proto nepřekvapí, že rozsah a počet takto uskutečněných průzkumů nebývá velký. Cílem tvůrců všech aktivitně-cestovních průzkumů je tedy snaha omezit nutnou interakci s uživatelem na minimum.

Na počátku 70. let 20. století začal být ve Spojených státech připravován pasivní¹ družicový systém pro dálkové měření polohy, známý též jako Global Positioning System (GPS). Pro civilní použití byl systém GPS přístupný od 90. let, avšak do roku 2000 byla jeho přesnost pro veřejnost záměrně armádou omezena zhruba na sto metrů z důvodu bezpečnosti. Plně hodnotné zpřístupnění GPS přineslo velké výhody a také možnosti pro tvorbu cestovních deníků.

Zásluhou velké přesnosti (jednotky metrů) je mnohem jednodušší detektovat počátek a konec cesty včetně celé její trajektorie, snižuje se také uživatelská zátěž s ohledem na zapamatování si času nebo délky trvání jednotlivých aktivit a cest. V praxi je tento princip využit tak, že participanti obdrží GPS přijímač, který údaje sbírá automaticky. S ohledem na rozmach chytrých osobních zařízení, které tento přijímač mají většinou zabudovaný, se pak mohou využívat mobilní aplikace.

Zmenšení nároků na uživatele zlepšuje schopnost provádět vícedenní průzkumy, které mohou snížit potřebný reprezentativní vzorek o 20 až 30 % [23]. Stále však přetravává nedostatek v podobě nutnosti zpětného ohodnocení uskutečněných tras, tedy hlavně určení vykonané aktivity a volba konkrétního dopravního prostředku.

Metodika pro realizaci aktivitně-cestovního průzkumu byla v České republice certifikována Ministerstvem dopravy v roce 2014 [20]. Specifikuje doporučené údaje pro shromažďování v cestovních denících, ať již se jedná o domácnosti či jednotlivce. Mezi tyto patří:

- Dostupnost dopravních prostředků, včetně informace o vlastnění řidičského oprávnění a karty MHD
- Možnosti parkování mimo domov, tedy například v místě pracoviště/vzdělávání
- Hlavní aktivity - počet odpracovaných hodin týdně, možnost práce z domova

Dále se doporučuje získat informace o pohlaví, věku a ekonomické aktivitě.

Nejčastějšími příjemci vyhodnocených dat z aktivitně-cestovních průzkumů jsou veřejné instituce a úřady odpovědné za plánování dopravních opatření. Cenná však tyto

¹ Pasivní v tomto případě znamená, že signál pouze přijímáme, ale nevysíláme

data mohou být i pro soukromé společnosti, které se účastní realizace takovýchto opatření (projektantské společnosti, developeři), případně komerční firmy s cílem lokalizační analýzy daného trhu. V neposlední řadě jsou pak vyhodnocené průzkumy důležité pro poskytovatele dopravních služeb.

Datum a čas	Aktivita	Prostředek	Trvání[h]	Ulice
12/3/2011 9:00	Práce	MHD	8	Hlavní
12/3/2011 18:00	Nákup	MHD	0.5	Pražská
12/3/2011 19:00	Doma	Pěšky	1.5	Krásná
12/3/2011 21:00	Kino	Vůz	3	Akční

Tabulka 2.1. Ukázka jedné z možných podob zaznamenaných dat.

■ 2.3.2 Modelování a validace aktivitních systémů

Pro vyhodnocování výsledků dopravních průzkumů a na nich založených aktivitních modelech se v poslední době využívají moderní technologie z oblasti umělé inteligence. Jednou z takových možností, jak modelovat celý systém a predikovat budoucí chování je využití tzv. multi-agentního přístupu. Ten dokáže simulovat interakci mezi jednotlivými obyvateli (tedy *agenty*) i jejich interakci se systémem [24]. Náročnou úlohou je však validace takového komplexního systému, jedním z mála poskytovaných frameworků pro tento účel je například *VALFRAM*¹. Ten porovnává časové a prostorové charakteristiky aktivitních modelů oproti skutečným cestovním deníkům a umožňuje tak statistickou validaci konkrétního modelu. [25].

¹ Validation Framework for Activity-based Modesl

Kapitola 3

Analýza dostupných řešení

Vytvoření systému pro popsanou metodiku aktivitně-cestovního průzkumu, včetně odstranění stávajících nedostatků cestovních deníků, vyžaduje analýzu hned v několika oblastech. Zaprvé je nutné prozkoumat současné možnosti a omezení technologií, které umožňují automatizovaný sběr co nejpřesnějších dat.

S ohledem na co možná nejvyšší uživatelský komfort se jako optimální řešení jeví využití mobilního přístroje, který zpravidla většina osob nosí neustále u sebe a tudíž by pro ně nepředstavoval přidanou zátěž, na rozdíl od případu samostatného GPS přijímače. Dále je nutné analyzovat možnosti mobilních senzorů, především pak jejich využití pro odhad zvoleného dopravního prostředku. Poslední závislost na zpětné vazbě od uživatele, tedy určení dané aktivity, by pak mohly odstranit prostorové a časové znalosti o daném místě. Při dosažení všech těchto cílů, samozřejmě s ohledem na datovou a energetickou náročnost, by se tak omezily požadavky na participanta pouze na nainstalování dané aplikace a počáteční sdělení demografických údajů.

K využití těchto dat je potřeba zvolit efektivní způsob pro rekonstrukci dané trasy, tedy například zjištění konkrétní linky MHD v případě jejího využití. K zajištění přesných výsledků bude nutné získat znalosti o dopravní síti, včetně jízdních řádů a poloze zastávek.

3.1 MEILI Travel Diary

Jezdním z existujících systémů poskytujících podobnou funkcionalitu je open-source projekt MEILI¹ švédských vývojářů z univerzity KTH. Součástí je mobilní klient pro operační systémy Android a iOS, který zajišťuje sběr dat. K dispozici je také webové rozhraní, umožňující zobrazení zaznamenaných tras, celkových statistik a jejich vizualizaci.

Jako nevhodným pro účely práce se však jeví zvolený způsob řešení mobilní aplikace, a to z několika důvodů. Zaprvé je nutné aplikaci spouštět před každou zamýšlenou cestou, zadruhé je potřeba ručně zvolit aktuální dopravní prostředek. To je v rozporu s cílem přenesení co možná největší zátěže z uživatele na samotný systém. V neposlední řadě je klientská aplikace dlouhou dobu neaktualizována, a nemůže tudíž reprezentovat velké změny ve vývoji operačních systémů a mobilních senzorů.

3.2 Future Mobility Sensing

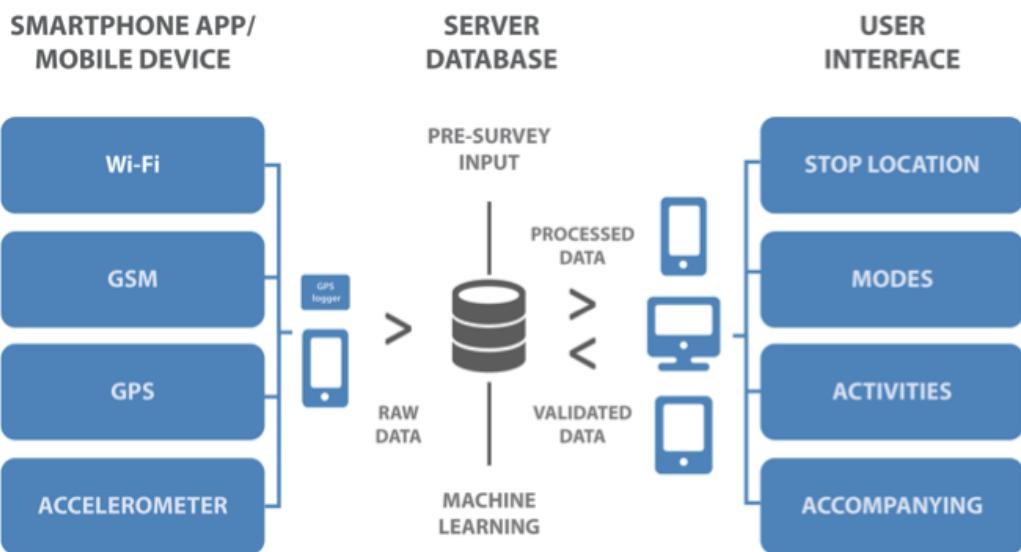
Future Mobility Sensing (FMS) je systém vyvinutý výzkumnou skupinou Singapore-MIT Alliance for Research and Technology (SMART)². Představený byl v roce 2013 a po úspěšném pilotním průzkumu byl nasazen mezi Singapurskými domácnostmi. [26] Průzkum trval od října 2012 do září 2013, a zúčastnilo se ho 1440 uživatelů.

¹ <https://github.com/Badger-MEILI>

² <https://its.mit.edu/future-mobility-sensing>

Systém má podobnou architekturu jako MEILI, před nainstalováním aplikace však vyžaduje vyplnění krátkého dotazníku a posléze validaci jednotlivých cest ze strany uživatele přes webové rozhraní. Z výše uvedeného počtu účastníků cesty validovalo 948 uživatelů, z toho 793 po dobu více než 5 dnů [27]. Vychází tedy, že cca 45 % účastníků cesty nevalidovalo, a neposkytlo tak důležitou zpětnou vazbu pro vyhodnocení úspěšnosti predikce módů a aktivit.

Výhodou je využití běhu aplikace na pozadí. Tento přístup neobtěžuje uživatele v situacích, kdy začíná cestu a musel by aplikaci zapnout sám. Samozřejmě, za standardních situacích se tento postup nedoporučuje, protože uživatel ztrácí kontrolu nad životním cyklem aplikace, což může negativně ovlivnit životnost baterie. Vzhledem k účelu průzkumu, kdy je uživatel dopředu s touto informací seznámen, se však jedná o žádoucí chování.



Obrázek 3.1. Schéma architektury systému FMS. Zdroj: <https://its.mit.edu/future-mobility-sensing>.

Z popsáné architektury řešení systému FMS není zřejmé, jakým způsobem je prováděna detekce dopravních módů, je však zcela jistě závislá na zpětné validaci. Naopak poměrně rozsáhle je popsána detekce aktivit, která využívá prostorové a časové závislosti nashromážděných GPS údajů. Princip této metody, spolu s porovnáním jiných přístupů, je vysvětlen v následující sekci.

3.3 Související práce

Výzkumem rozpoznávání dopravních módů či aktivit se v posledních letech začíná zabývat stále více pracovních skupin. Některé staví své výsledky na údajích z GPS, tedy v případě módů především rychlosti, jiné využívají pravděpodobnostních modelů obsahující časové a prostorové závislosti. Postupně se seznámíme se všemi publikovanými přístupy a budeme diskutovat vhodnost jejich využití pro nově vytvořený systém.

3.3.1 Detekce dopravních módů za využití GPS

Nejjednodušším způsobem jak odhadnout zvolený dopravní prostředek je pouze za využití GPS záznamů. Z průběhu trajektorie jsme schopni zpětně rekonstruovat hodnoty

průměrné i maximální rychlosti. Tímto přístupem se zabývá například [28]. Studie zkoumala parametry jednotlivých módů na základě dat získaných od 900 respondentů mezi říjnem a listopadem 2010. Celkově bylo zkoumáno přes 1 milion GPS záznamů z území městské aglomerace Peking.

parametr	Automobil	Jízdní kolo	Metro	Autobus	Chůze
V_p [km/h]	14.54	8.47	11.21	13.27	3.00
σ_v [km/h]	9.20	8.20	13.32	14.21	2.49
V_{max} [km/h]	53.25	20.87	44.51	50.23	15.35
$\sigma_{v_{max}}$ [km/h]	18.73	14.73	30.22	19.33	7.45

Tabulka 3.1. Identifikované hodnoty jednotlivých dopravních módu. V_p odpovídá průměrné rychlosti, V_{max} maximální rychlosti, σ je směrodatná odchylka. Zdroj [28].

Údaje byly získány zásluhou zpětné validace cestovních deníků jednotlivými účastníky.

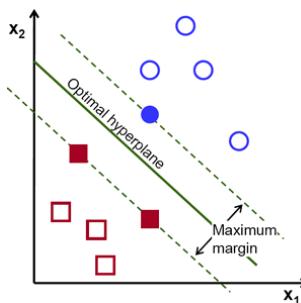
Z předchozí tabulky můžeme vyzorovat, že parametry průměrné rychlosti jsou velmi podobné pro kategorie automobilu, autobusu i metra. V případě maximální rychlosti se sice již rozdíly zvětšují (především u metra), stále však zůstávají natolik malé, aby bylo možné tyto kategorie relevantně odlišit.

Studie se proto zabývá možností uplatnění metod strojového učení, především tzv. metody Support vector machines (SVM). Ta spočívá v klasifikaci dat nalezením optimální nadroviny takové, že sada dat patřící do odlišných tříd leží v opačných poloprostorech. Pokud uvažujeme o množině dat jako o dvojicích (\vec{x}_i, y_i) kde \vec{x}_i je vektor našich hodnot a y_i je hodnota +1 nebo -1 podle toho, do jaké kategorie záznam patří, můžeme nadrovинu popsat rovnicí¹:

$$\vec{w} \cdot \vec{x} + b = 0$$

kde \vec{w} značí normálu nadroviny, $\|\vec{w}\|$ její Eukleidovskou normu a $\frac{|b|}{\|\vec{w}\|}$ vzdálenost nadroviny od počátku souřadného systému.

Optimální nadrovinu pak nalezneme maximalizací vzdálenosti nejbližšího bodu z obou poloprostorů od ní (tato vzdálenost se někdy nazývá margin). Tím vytvoříme jakési pásmo okolo nadroviny.



Obrázek 3.2. Ukázka použití metody SVM [29].

¹ https://en.wikipedia.org/wiki/Support_vector_machine

Výslednou nadrovinu tedy můžeme určit jako

$$\min_w \frac{1}{2} \|\vec{w}\|^2$$

$$z.p. \quad y_i(\vec{x}_i \cdot \vec{w} + b) - 1 \geq 0 \quad \forall i > 0$$

Na popis této nadroviny nám stačí najít krajní body z obou poloprostorů ležící na hranici tohoto pásma (tzv. hraniční pásmo) a jejich vektory - odtud tedy název Support vector (podpůrné vektory).

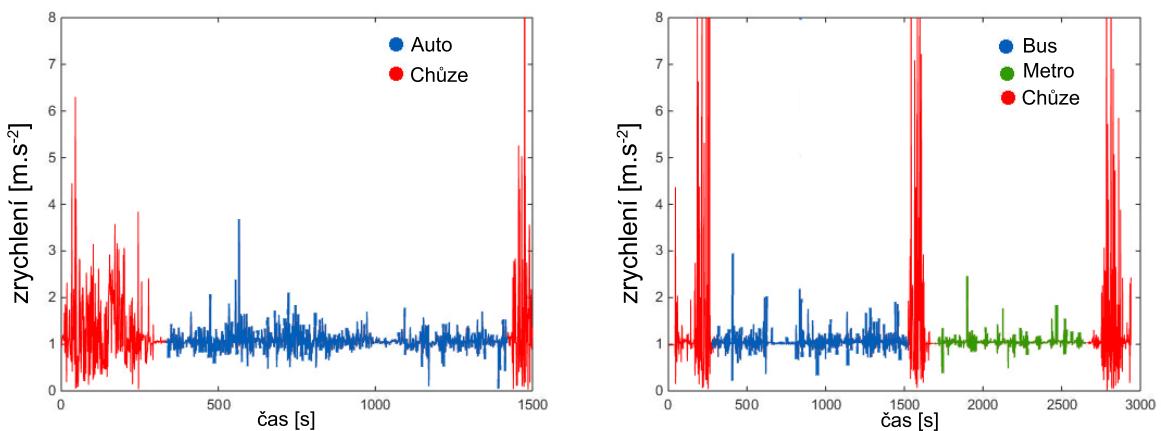
Úspěšnost predikce na zmíněných datech za použití optimalizované metody SVM dosahovala celkově 92 %, nicméně u samostatné kategorie aut pouze 80 % [28]. To může být zapříčiněno tím, že je velmi těžké odlišit rychlostní charakteristiku automobilu od autobusu, a proto byla nemalá část chybně vyhodnocena právě na tento dopravní mód.

Navíc je nutné pro tento přístup zaznamenat poměrně velké množství GPS bodů (především v městském provozu), aby se dostatečně projevily charakteristiky daného prostředku.

■ 3.3.2 Detekce dopravních módu za využití akcelerometru

Dalším způsobem detekce je využití některého z mobilních senzorů, konkrétně akcelerometru a gyroskopu. První senzor zachycuje okamžité zrychlení zařízení, druhé pak výkyvy jeho jednotlivých os při pohybu. Výhodou akcelerometru je také fakt, že dokáže při zvýšené snímkovací frekvenci zachytit i velmi malé vibrace zařízení, tedy například i jemné chvění od motoru při sezení v sedačce automobilu/autobusu [30].

Hodnoty z akcelerometru byly v rámci analýzy jeho využití pro potřeby práce zkoumány na 3 mobilních telefonech po dobu jednoho týdne. Porovnání výsledných údajů odhalilo velké rozdíly v zaznamenaných vibracích pro kategorii chůze a motorového vozidla¹, což ilustruje následující graf srovnání jedné cesty (tedy cesty mající společný start a cíl) pro různé dopravní prostředky.



Obrázek 3.3. Hodnoty zrychlení v závislosti na čase pro různé dopravní prostředky.

Bohužel z obrázku je patrné, že opět jako v případě rychlosti, i naměřené hodnoty zrychlení nevykazují natolik statisticky významné rozdíly, aby bylo možné rozlišit jednotlivé prostředky z kategorie motorových vozidel.

Podobným výzkumem se také zabývají dvě studie z let 2014 a 2016 [30][31]. První zkoumala hodnoty zrychlení na vzorku 18 participantů a 3 dopravních módech (chůze,

¹ Mezi motorová vozidla počítáme automobil, autobus, tramvaj, metro a vlak

jízdní kolo a vozidlo), druhá získala uživatelská data firmy HTC od 224 dobrovolníků z období let 2012-2014 a oproti první rozlišuje i běh.

Srovnání obou výzkumů se nabízí z důvodu použití stejné metody pro klasifikaci dat, konkrétně již zmíněné metody SVM. Zaměříme-li se na procentuální úspěšnost predikce, první ze studií dosahuje téměř[3] 99% celkové úspěšnosti, zatímco druhá zhrubá 87%. Nutno však poznamenat, že deklarované výsledky mohou být v případě první studie zkresleny poměrně nereprezentativním vzorkem, a to především z důvodu malého za-stoupení jízd vozidlem (cca 25 %). Pro tento dopravní mód totiž obě studie vykazují výrazně horší výsledky (obě okolo 85 %).

Publikované hodnoty průběhu zrychlení druhé studie mají podobné charakteristiky jako námi naměřené, lze proto předpokládat, že akcelerometr lze dobře využít při rozlišování dopravních módů chůze, jízdního kola a motorového vozidla. Bohužel v případě cestovních deníků by jeho trvalé použití pro získání uceleného vzorku znamenalo velkou zátěž na baterii, vzhledem k současně zaplému přijímači signálu GPS.

■ 3.3.3 Detekce aktivit

Druhou důležitou součástí automatizovaného cestovního deníku je rozpoznávání aktivity. Vezmeme-li v úvahu skutečnost, že pro sběr pozičních údajů potřebujeme příjem GPS signálu, získáváme základní rámec také pro vykonávané aktivity. Zásluhou jednotlivých koordinát jsme totiž schopni určit místo a čas jak počáteční aktivity, tak té cílové. Zásadní informací je také doba trvání dané aktivity, tedy rozdílu mezi dvěma navazujícími cestami.

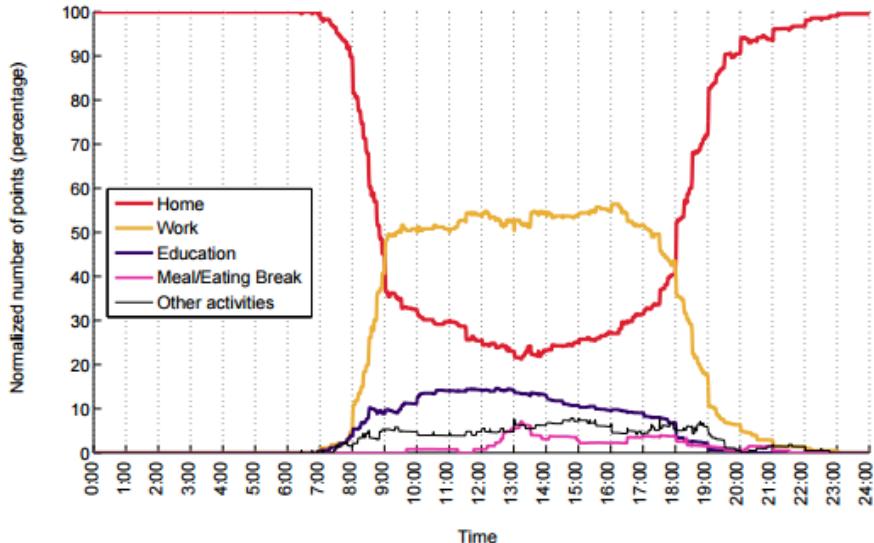
Tím však naše vstupní hodnoty získané od participanta končí, a další potřebné údaje musíme buď získat od třetí strany, či případně odvodit z určitých charakteristických vzorců typických pro různé kategorie aktivit. Do první skupiny spadají například informace o typu dané lokality. Můžeme rozlišovat rezidentní zóny, kde je větší pravděpodobnost aktivity „cesta domů“, a naopak komerční nebo průmyslové areály, do kterých se bude cestující vydávat spíše za účelem zaměstnání.

Nejzajímavějšími místy jsou pak tzv. points of interest (POI, česky většinou překládané jako body zájmu), které se svým významem vztahují přímo ke konkrétnímu adresnímu místu. Můžeme se tak například dozvědět, že v blízkosti místa cílové aktivity se nachází restaurace, obchodní dům, vzdělávací instituce atd. Tato znalost samozřejmě dále zvyšuje pravděpodobnost úspěšného odhadu typu aktivity. Poskytovatel takových databází jsou např. firmy Google, Foursquare nebo Yelp.

Využití těchto zdrojů má však smysl pouze v případě jejich propojení s časovým rámcem získaným z dat GPS. Uvažujme například situaci, kdy se bude adresa místa aktivity shodovat s údajem z databáze POI, a jejím typem bude „restaurace“. Mohli bychom tak logicky předpokládat, že cestující navštíví toto místo za účelem občerstvení. Důležité je ale zasadit tuto informaci do kontextu celkového setrvání na místě aktivity, neboť v případě jejího výrazně dlouhého trvání (8 hodin) se pravděpodobně bude jednat o aktivity „zaměstnání“, a naopak v kombinaci s krátkým intervalom bude zřejmě pravdivý původní odhad [32].

Dále si musíme uvědomit fakt, že pravděpodobnost vykonávání určitých typů aktivit je závislá na denní době. Je například velmi nepravděpodobné, že by participant průzkumu začal aktivitu „zaměstnání“ nebo „vzdělávání“ později večer, naopak v tuto dobu bude stoupat počet návštěv obchodů, restaurací či kin. Nemůžeme ovšem tyto nenadálé případy přímo vyloučit, je proto důležité průzkum provádět po delší období, aby se dopad případných odchylek minimalizoval.

Tyto časové souvislosti bychom mohli nazvat atraktivitou daného bodu zájmu v konkrétní denní době. Pro ilustraci můžeme uvést výsledky pozorování z již uvedeného průzkumu v Singapuru systémem FMS (viz 3.1), ve kterém byl z validovaných záznamů sestaven graf znázorňující atraktivitu pro základní typy aktivit.



Obrázek 3.4. Atraktivita daných aktivit v závislosti na denní době. Převzato z [33].

Kapitola 4

Návrh řešení

V předchozích kapitolách jsme identifikovali nejdůležitější části, které by systém pro aktivitně-cestovní průzkum mobility měl obsahovat. Zásluhou existující metodiky (2.3) také víme, o jaké údaje bychom v rámci vstupního formuláře měli účastníky požádat. V sekci 2.3.1 jsme se dozvěděli o potřebách koncipovat takový systém jako co možná nejméně závislý na uživateli, ať již z pohledu zpětné validace nebo samotného sběru dat. Porovnali jsme podobná řešení, a z výše uvedených důvodu se jako nejlepší jeví použití mobilní aplikace. Ta nám díky zabudovanému GPS přijímači umožní sbírat potřebná data, ze kterých můžeme určit důležité časové a prostorové souvislosti. Navíc se jedná o poměrně moderní oblast výpočetní technologie, která nabízí velké možnosti budoucího rozšíření.

Dále jsme diskutovali možnosti tvorby algoritmů pro predikci aktivit a určení dopravních módů na základě takto získaných dat. Zjistili jsme, že metodami strojového učení, například metodou SVM (sekce 3.2) můžeme dosáhnout poměrně přesné klasifikace některých kategorií dopravních prostředků. Vyplynulo ovšem, že při rozpoznání jednotlivých druhů motorových vozidel, především automobilu a autobusu, mohou být nespolehlivé. Vzhledem k poměrně velkému zastoupení těchto dvou způsobů přepravy ve městech, na která cílí aktivitně-cestovní průzkum především, mohou sloužit spíše jako orientační údaj pro další fáze zpracování.

Jednou z možností, jak zpřesnit odhad zvoleného dopravního módu, je využití znalosti o dopravní síti. Pokud bychom totiž věděli, že trajektorie cesty odpovídá možné trase nějakého spojení veřejnou dopravou, bylo by jednodušší rozhodnout o konkrétním dopravním prostředku.

Některé dopravní podniky v součastnosti vybavují svá vozidla tzv. GPS lokátory, především z důvodu kontroly plnění předepsaných standardů a dodržování jízdních řádů. V Praze je známý systém DORIS¹ pro sledování provozu tramvají. Bohužel data z něj slouží pouze interním potřebám.

V posledních letech se ale zásluhou zvýšeného zájmu o otevřená data² zvyšuje tlak na zpřístupnění informací o infrastruktuře městské hromadné dopravy. To však vyžaduje existenci standardizovaného formátu pro výměnu těchto dat.

4.1 General Transit Feed Specification

Patrně nejznámějším a jedním z nejrozšířenějších formátů pro sdílení jízdních řádů a souvisejících geografických informací je General Transit Feed Specification (GTFS) vyvinutý firmou Google a představený na přelomu let 2005-2006. Samotný formát sice není v současné době zapsán jako oficiální standard žádnou mezinárodní organizací, je

¹ <https://cs.wikipedia.org/wiki/DORIS>

² Otevřená data, anglicky Open data jsou informace a data zveřejněná na internetu, která jsou úplná, snadno dostupná, strojově čitelná, používající standardy s volně dostupnou specifikací, zpřístupněná za jasné definovaných podmínek užití dat s minimem omezení a dostupná uživatelům při vynaložení minima možných nákladů. Zdroj: wikipedia.org

za něj však všeobecně považován a proto se také někdy označuje jako de-facto standard [34].

Myšlenkou stojící za jeho vývojem bylo získat jednotná data od více dopravních společností, která by sloužila jako podklad pro plánovač cest Google Transit Trip Planner. Ten je dnes součástí známé služby Google Maps a nazývá se zkráceně Transit. Prvním, kdo takto začal poskytovat informace, bylo město Portland ve Spojených státech v prosinci roku 2005. Od té doby jsou součástí plánovače stovky měst a tisíce dopravců po celém světě¹.

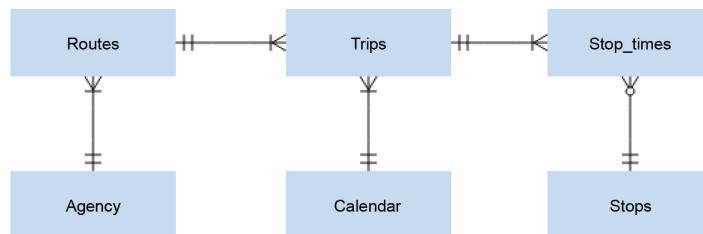
4.1.1 Formát GTFS dat

Balíček GTFS dat se skládá ze 13 samostatných souborů uložených ve formátu CSV (Comma-separated values). Jedná se o textovou reprezentaci dat ve formě tabulky, v níž jsou jednotlivé položky každého řádku odděleny znakem čárky (,). Přípustné jsou i jiné oddělovací znaky, nejčastěji středník (;), a to z důvodu použití čárky v desetinných číslech v některých jazycích. Pokud se v záznamu vyskytuje čárka, může být jeho text uzavřen do uvozovek, v případě výskytu uvozovek jsou pak tyto zdvojeny.

Ze 13 možných souborů je však povinných pouze 6:

- Agency.csv - obsahuje kontaktní údaje dopravce.
- Routes.csv - obsahuje popis, typ a názvy jednotlivých linek.
- Trips.csv - obsahuje jméno, směr a popis konkrétní trasy dané linky.
- Stops.csv - obsahuje název, zeměpisné souřadnice a další informace o zastávce, například její bezbariérovost.
- Stop_times.csv - obsahuje všechny časy odjezdů a příjezdů linek.
- Calendars.csv - obsahuje informaci o dnech, ve kterých dané linky jezdí.

O GTFS záznamech hovoříme jako o tabulkách z důvodu podobnosti s relačními databázemi, jelikož relace jsou v souborech zajištěny pomocí sdílené hodnoty (ekvivalent primárních a cizích klíčů).



Obrázek 4.1. Entity-Relationship diagram povinných souborů GTFS balíčku.

Jak konkrétně pak takové záznamy vypadají se můžeme podívat na následující ukázce pro linky (routes.csv).

```

route_id,agency_id,route_short_name,route_long_name,route_type,r_color
L991D1,1,A,"Nemocnice Motol - Petřiny - Skalka - Depo Hostivař",1,408000
L992D1,1,B,"Zličín - Černý Most",1,FFFF00
L993D1,1,C,"Letňany - Ládví - Háje",1,FF0000
  
```

Vidíme, že každé lince náleží unikátní identifikátor, na který jsou odkazovány jednotlivé spoje obsluhující danou linku v souboru trips.csv.

¹ <https://maps.google.com/landing/transit/cities/>

Pro zastávky (stops.csv) a časy odjezdů je situace obdobná.

```
stop_id,stop_name,stop_lat,stop_lon,location_type,parent_stn,wheelchair
U1071Z102,"Depo Hostivař",50.075541,14.51532,0,U1071N1,1
U953Z102,"Skalka",50.068435,14.507169,0,U953N2,1
U713Z102,"Strašnická",50.073336,14.490091,0,U713N3,1
```

Zbylých 7 souborů je nepovinných a přidávají informace např. o platnosti jízdních řádů, trajektorii jednotlivých linek nebo frekvenci určitých spojů.

■ 4.1.2 Využití GTFS dat

Vznik jednotného formátu pro sdílení dat o hromadné dopravě ve spojitosti s jeho rostoucí oblibou umožňuje vývoj velkého množství informačních systémů. V současné době jsou tato data využívána především pro plánovače cest a mapové aplikace, jako jsou například již zmíněné Google Maps nebo OpenTripPlanner. Dalšími okruhy jsou webové a mobilní aplikace pro vyhledávání časů nejbližších odjezdů, tisk jízdních řádů nebo pro vizualizaci sítě veřejné dopravy.

Z pohledu tvorby aktivitně-cestovního deníku je zajímavá především možnost propojení nashromážděných GPS záznamů s dopravní sítí. Na počátku kapitoly jsme identifikovali problém s rozlišením dopravního módu v případě automobilu a prostředků hromadné dopravy. Pokud bychom ale propojili naše údaje s GTFS záznamy pro danou oblast, mohli bychom na základě porovnání časových a prostorových parametrů toto riziko minimalizovat. S ohledem na přesnost GPS systému je totiž velmi pravděpodobné, že objevíme shodu mezi pozicí cestujícího a případných zastávek MHD.

Pouze porovnání pozice by však nestačilo, cestující by totiž například mohl okolo identifikované zastávky pouze projíždět v automobilu, nebo stát na semaforech. V tomto případě je nutné porovnat i další souvislosti, jmenovitě:

- způsob, jakým se cestující dostavil na případnou nástupní stanici.
- kontinuitu spojení - případná cesta MHD musí začínat a končit na zastávkách nějaké linky.
- způsob, jakým cestující pokračoval z případné výstupní stanice.
- shodu časů odjezdů a příjezdů dle jízdního řádu (samozřejmě s určitou tolerancí).
- shodu celkového intervalu cesty MHD s deklarovaným intervalem dle GTFS.
- shodu v trajektorii cesty s udávanou trajektorií dle GTFS.

V situacích, kdy bychom zjistili dostatečnou shodu s deklarovanými parametry bychom navíc velmi snadno zjistili i další údaje o trase (konkrétní linka MHD, zpoždění spojů atd.) Využití znalosti o dopravní síti proto bude tvořit důležitou součást poskytovaného systému.

■ 4.2 Detekce dopravních módů

Pro základní klasifikaci zvoleného dopravního prostředku 3.2 existují účinné metody kombinující data z GPS a mobilních senzorů (akcelerometr). Pokud se zamyslíme nad možnými způsoby vyhodnocování těchto dat, dojdeme ke dvěma závěrům. Zaprvé je možné detekci provádět kompletně retrospektivně, k čemuž ale potřebujeme poměrně velké množství přesných záznamů GPS i údajů o zrychlení. V případě sběru dat mobilní aplikací tento přístup tedy vyžaduje zapnutí více senzorů naráz, což logicky zvyšuje energetické nároky.

Druhou možností je detekce módu přímo v mobilním klientovi. Tento přístup by umožnil také zachytit konkrétní změny dopravních prostředků v rámci jedné cesty (např.

přestupy na zastávkách), čímž bychom získali cennou informaci o průběhu uskutečněné trasy. Dále bychom mohli určit, jakým způsobem se cestující pohyboval v určitých úsecích trasy, tedy například i zjistit, zdali se na danou zastávku MHD dostavil pěšky, případně zdali mezi dvěma možnými koncovými zastávkami opravdu jel vozidlem.

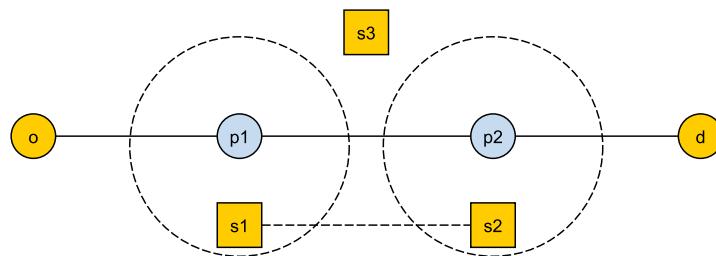
Detekce obecného dopravního módu (walk, vehicle, still) přímo na straně klienta by tak výrazně zjednodušila a zpřesnila detekci konkrétního módu (car, bus, tram, metro). Naštěstí není nutné mít zmíněné senzory zapnuté neustále, existují totiž veřejně dostupné knihovny pro mobilní systémy, které dokáží obecné dopravní módy rozpoznat velmi rychle (v řádu několika málo sekund) - například Google Activity Recognition API¹ nebo PathSense API². Zapnutí senzorů na několik málo vteřin například v minutových intervalech se jeví jako rozumný kompromis mezi úsporou energie a ziskem informací. Takový interval by totiž stále měl pokrýt případné změny módu při přestupech.

Při uvážení situace, kdy máme k dispozici GPS záznamy z cestovního deníku, dále GTFS reprezentaci dopravní sítě a informaci o obecném dopravním módu ke každé zaznamenané souřadnici, nám již zbývá jen klasifikovat konkrétní mód v případě vozidla. Abychom mohli rozlišit automobil a MHD, potřebujeme získat potenciální koncové body (zastávky) na uskutečněné trase.

Nechť bod $p(t, x, y, m)$ odpovídá jednomu záznamu GPS z uskutečněné trasy T . Potom $t \in \mathbb{N}$ odpovídá časové stopě v milisekundách, $x, y \in \mathbb{R}$ odpovídá zeměpisné šířce a zeměpisné délce, $m \in M = \{\text{Car}, \text{Still}, \text{Walk}\}$ odpovídá detekovanému dopravnímu módu v daném bodě. Funkce $\text{mode} : T \rightarrow M$ vrátí hodnotu dopravního módu pro daný bod.

Abychom nemuseli procházet všechny souřadnice z jedné cesty a porovnávat je se seznamem všech zastávek v síti, stačí nám vzít pouze body, ve kterých došlo ke změně dopravního módu. Takovou množinu označíme $T' = \{p_i \mid (p_i, p_{i+1} \in T) \wedge (\text{mode}(p_i) \neq \text{mode}(p_{i+1}))\}, \forall i \in 1, \dots, n-1\}$ kde n je velikost množiny T .

Definujme dále bod $s(x, y)$ jako zastávku z množiny všech zastávek v síti S a funkci $d : T \times S \rightarrow \mathbb{R}_0^+$ jako vzdálenost zastávky od zaznamenaného bodu v metrech. Potom množina $C = \{(s, p) \mid p \in T', s \in S, d(s, p) < 100\}$ je množina všech možných zastávek, okolo kterých cestující prošel nebo projel v okruhu 100 metrů.



Obrázek 4.2. Princip hledání možných zastávek na trase mezi body o a d . Zastávky s_1 a s_2 leží v blízkosti bodů p_1 a p_2 , na kterých došlo ke změně dopravního módu - proto jsou přidány do množiny C .

V množině C pak hledáme všechna spojení mezi každou dvojicí zastávek a vyřadíme takové, ke kterým nevede žádné spojení.

¹ Součást tzv. služeb Google Play pro systémy Android, v kontextu této knihovny je význam slova Aktivita chápan jako dopravní mód

² <http://pathsense.io/>

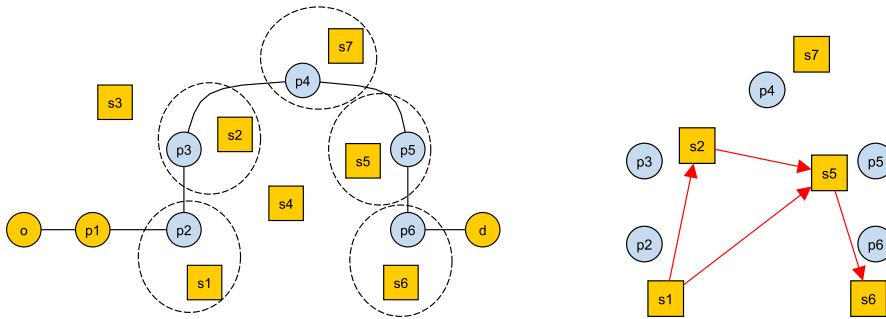
Vytvoříme tedy graf možných spojení $G = (V, E, g)$ kde

- V je množina všech zastávek z množiny C s nalezeným spojením,
- $E = \{(u, v) | (u, v) \in V \wedge (u \neq v)\}$ je množina orientovaných hran reprezentující dané spojení
- funkce $g : E \rightarrow \mathbb{N}$ je ohodnocení přiřazující každé hraně přirozené číslo.

$$g(s_1, s_2) = |t_{s_1d} - GTFS_{s_1d}| + |t_{s_2d} - GTFS_{s_2d}|$$

Funkce $g(s_1, s_2)$ ohodnocuje hranu (s_1, s_2) na základě podobnosti časů s jízdním řádem, kde t_{s_1d} a $GTFS_{s_1d}$ porovná časy odjezdů a t_{s_2d} , $GTFS_{s_2d}$ časy příjezdů.

Pokud nastane situace, že mezi uzly s_1 a s_2 existuje více možných spojení (tedy spojů od různých linek MHD), je vybráno to s lepším ohodnocením.



Obrázek 4.3. Konstrukce grafu G . Modře znázorněné body odpovídají souřadnicím, kde došlo ke změně dopravního módu a zároveň je v jejich blízkosti detekována zastávka MHD. Zastávky s_1, s_2, s_5 a s_6 jsou přidány do grafu G , zastávka s_7 je vyřazena z důvodu neexistujícího spojení s grafem.

Nyní potřebujeme najít nejlepší možné spojení mezi nástupní a výstupní zastávkou, které bude označovat případnou cestu MHD. Jelikož funkce našeho ohodnocení hledá minimum pro každou hranu, musíme vyhledat spojení mezi těmito koncovými body takové, že jeho výsledné ohodnocení bude minimální přes všechny možné. V našem případě tedy konkrétně hledáme nejkratší cestu v grafu G mezi první a poslední zastávkou. Pro problém hledání nejkratších cest v grafu existuje mnoho algoritmů, známý je například Bellman-Fordův pracující s asymptotickou složitostí $O(VE)$ nebo Dijkstrův algoritmus s lepší asymptotickou složitostí $O(E + V\log V)$. Existuje však případ, kdy se náročnost úlohy dá ještě snížit, a to pokud se jedná o orientovaný acyklický graf. V něm nalezneme topologické uspořádání vrcholů se složitostí $O(V + E)$ a poté z každého vrcholu prozkoumáváme výstupní hrany:

```

set dist[u] = inf for every vertex u
For every vertex u in topological order do:
    for every adjacent vertex v of u
        if (dist[v] > dist[u] + weight(u,v))
            dist[v] = dist[u] + weight(u,v)

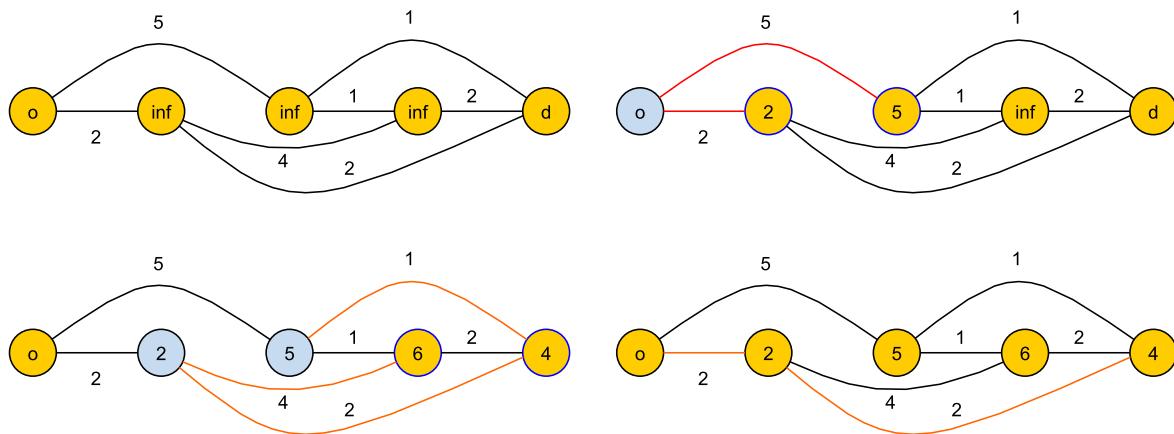
```

Náročnost tohoto algoritmu je $O(V + E)$, neboť pro každý vrchol jsou procházeny jeho výstupní hrany, každá však pouze jednou.

Pokud si uvědomíme, že zastávky jsou na dané trase cestujícího navštěvovány po stupně, a že linky MHD se po cestě nevrací dvakrát na stejnou zastávku, zaručit

poslední podmínu acyklického grafu. Tou je situace, kdy by se náhodou cestující v průběhu cesty rozhodl nečekaně vrátit. V takovém případě se tedy graf rozdělí na dva rozdílné. Odpadá také nutnost topologického uspořádávání, protože zastávky jsou již uspořádány z průběhu cesty. Výše uvedený algoritmus tak můžeme aplikovat i na nás případ.

Pro určení celkové trasy MHD je nutné kromě hodnoty nejkratší cesty zjistit také její průběh, konkrétně tedy zjistit po kterých hranách (spojeních mezi dvěma zastávkami) se uskutečnila. Proto je do grafu G přidána funkce $l : E \rightarrow R$, která každé hraně grafu G přiřazuje na základě jejího ohodnocení funkci g jednu linku z množiny všech linek R . Pro rekonstrukci trasy pak stačí hledat nejdélší úsek po stejně lince (tedy hrany se stejnou hodnotou funkce l , v případě změny hodnoty na hraně jsme zaznamenali přestup na jinou linku).



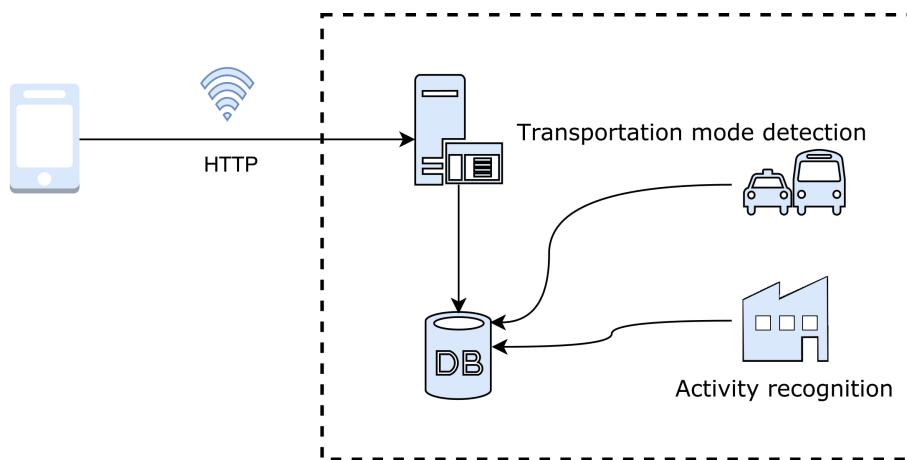
Obrázek 4.4. Ukázka algoritmu pro hledání nejkratší cesty mezi body o a d , na posledním obrázku je pak znázorněn i její průběh. V první fázi nastavíme ohodnocení všech uzlů na hodnotu inf a algoritmus začneme v takovém uzlu, která obsahuje pouze výstupní a žádnou vstupní hranu. V každé další iteraci je aktualizována hodnota uzlu s ohledem na její minimalizaci. Uzel je uzavřený v momentě, kdy začneme zpracovávat jeho potomky, neboť díky topologickému uspořádání a acyklicity se již nemůžeme vrátit.

4.3 Detekce aktivit

Pro detekci aktivit byla určena časová okna inspirovaná z [33]. Jako detekovanou aktivitou je pak zvolena ta, která vykazuje největší shodu v atraktivitě - tedy časových charakteristikách dle denní doby. Tyto údaje jsou navíc postupně aktualizovány, například pokud se zjistí opakované vykonávání aktivity na jednom místě (typicky zaměstnání nebo domov), jsou pak tyto aktivity doplněny u neohodnocených cest s odpovídajícím místem dané aktivity. Jako orientační údaj může sloužit např. název Wi-Fi sítě nebo nejbližší POI dle veřejně dostupných API.

4.4 Návrh architektury systému

Výše navrhované řešení počítá s využitím mobilního telefonu jako klientské části a serverové části pro vyhodnocení dat. Nashromázděná data, vzhledem k jejich struktuře, by měla být uchovávána v relační databázi, která umožňuje tvorbu potřebných vazeb a zajistí rychlé vyhledávání například podle požadovaného času. Dále je nutné vytvořit obdobné úložiště pro GTFS záznamy. V rámci aplikačního serveru je potřeba zajistit



Obrázek 4.5. Návrh architektury systému pro aktivitně-cestovní průzkum.

obslužné metody pro příjem a ukládání dat, stejně tak jako pro následnou detekci dopravních módů a aktivit.

Kapitola 5

Mobilní aplikace

Jako klientská část systému pro sběr dat cestovního deníku byl z důvodů zmíněných v kapitole 4.1 vybrán mobilní telefon. Při volbě operačního systému (OS) hrálo ve velké míře roli jeho procentuální zastoupení v populaci. V současné době mají na trhu dominantní postavení dva operační systémy - Android společnosti Google se 70 % a iOS společnosti Apple s téměř 20 %¹. Následující Androidu se však v posledních letech zvyšuje na úkor ostatních systémů, lze proto předpokládat, že bude v nejbližších letech stále nejpoužívanějším mobilním OS. Pro jeho volbu je také podstatné existující množství veřejně dostupných rozhraní (tzv. API²) poskytovaných nejenom firmou Google, ale také početnou komunitou vývojářů. Z nejdůležitějších jmenujme například Google Activity Recognition API pro rozpoznávání základních dopravních módů.

5.1 Analýza požadavků a rizik

Počáteční analýza a specifikace nezbytných částí aplikace slouží především jako výchozí bod pro návrhovou a implementační část.

5.1.1 Funkční požadavky

Funkční požadavky definují základní funkce, které by měla aplikace obsahovat. Každý požadavek by měl popisovat pouze jednu samostatnou a měřitelnou funkcionalitu.

- Aplikace umožní uživatelům registraci do systému
- Aplikace bude schopna sledovat polohu uživatele za využití systému GPS
- Aplikace bude také schopna detektovat základní dopravní mód přímo na straně klienta
- Aplikace bude funkční i v tzv. módu na pozadí
- Aplikace bude umožňovat automatické aktualizace
- Aplikace bude odesílat nashromážděná data na server
- Aplikace bude logovat případně chyby v systému a odesílat je na server
- Aplikace bude v provozu pouze při cestách uživatele
- V případě vypnutého přijímače GPS bude aplikace schopná sledovat polohu za využití Wi-Fi a signálu operátora.

5.1.2 Obecné požadavky

Tato sekce specifikuje omezující podmínky, které jsou kladený na systém, nikoliv na jeho funkcionalitu. Většinou se jedná o použité technologie a omezení týkající se kvality a výkonu. Proto se také těmto požadavkům někdy říká nefunkční.

- Aplikace bude implementována v jazyce Java pro operační systém Android

¹ <http://gs.statcounter.com/os-market-share/mobile/worldwide>

² Application Programming Interface označuje v informatice rozhraní pro programování aplikací. Jde o sbírku procedur, funkcí, tříd či protokolů nějaké knihovny, které může programátor využívat. Zdroj: wikipedia.org

- Aplikace poneše název GPS Tracker, který by měl co nejpřesněji vystihovat její účel.
- Aplikace bude funkční na systému Android od verze 4.1
- Aplikace nebude využívat mobilní data pro odesílání dat na server.
- Aplikace omezí energetické nároky na maximální možné minimum.
- Aplikace bude dostupná v češtině a angličtině.
- Aplikace bude obsahovat informace o účelu dopravního průzkumu
- Aplikace bude schopna zajistit základní demografické údaje

■ 5.1.3 Analýza problémů a rizik

Pro vývoj GPS cestovního deníku v podobě mobilní aplikace je zásadní několik skutečností.

Zaprve je nutné se vypořádat s problémem zvýšené energetické spotřeby, neboť sledování polohy (a tedy zapnutí přijímače GPS) patří k nejnáročnějším úkonům v oblasti mobilních systémů. Bude tedy nutné najít určitý kompromis mezi spotřebou baterie a zajištěním dostatečné přesnosti záznamů.

Dalším problémem je funkčnost aplikace v tzv. background módu¹, tedy situaci, kdy uživatel proces aplikace neukončí, ale přesune jej právě na pozadí běhu OS. V tom případě je nutné zajistit, aby hlavní proces aplikace nebyl nenadále přerušován a nedošlo tak k výpadku v záznamu dat. S tímto problémem samozřejmě souvisí i situace, kdy aplikace není zapnuta vůbec (tedy ani přesunutá uživatelem na pozadí). V takové chvíli je nutné proces záznamu dat spustit automaticky prostřednictvím OS.

Složité také může být zajištění kompatibility pro různé verze systému Android. Tento OS prochází neustálým a poměrně intenzivním vývojem, který s sebou logicky přináší velké změny. Každá nová verze definuje své rozhraní (tzv. API level), podle kterého vývojáři mohou používat dostupnou funkcionalitu. Během let 2012-2016 bylo vydáno celkem 9 takových verzí.

V neposlední řadě je nutné zabránit ztrátě dat například při nezdařeném odeslání na server. Pokud bychom totiž automaticky počítali s tím, že jednou odeslaná data můžeme z důvodu úspory kapacity úložiště vymazat, mohlo by dojít k nevratné ztrátě informací o zaznamenané cestě uživatele, tedy nejdůležitějšímu účelu aplikace.

■ 5.2 Popis architektury

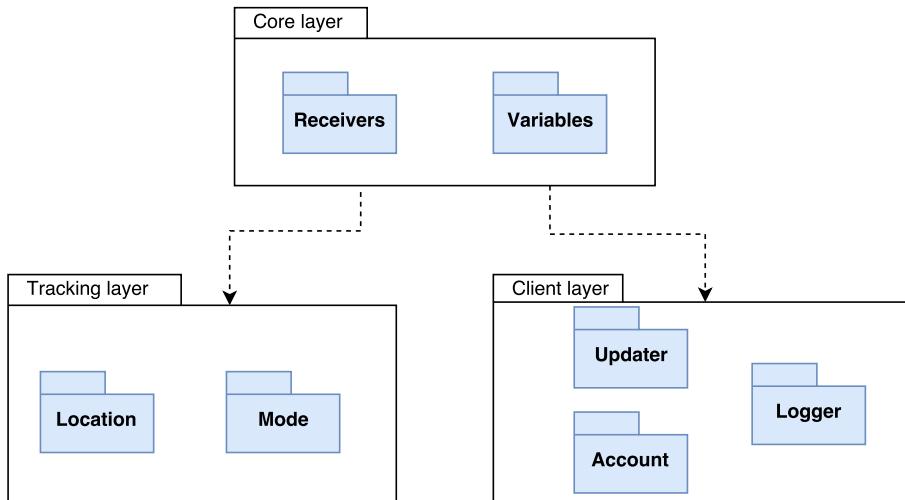
Aplikace GPS Tracker implementovaná v jazyce Java se skládá z několika balíčků tvořících ucelený systém. Každý z těchto balíčků obsahuje vzájemně související třídy a zastupuje tak konkrétní funkcionalitu aplikace. Jednotlivé balíčky bychom dále mohli rozdělit do třech vrstev aplikace, konkrétně:

- Core layer - základní vrstva aplikace odpovědná za řízení ostatních vrstev
- Client layer - obslužná vrstva pro interakci s uživatelem
- Tracking layer - vrstva odpovědná za proces sledování polohy a detekce módů

■ 5.2.1 Core layer

Tato řídící vrstva se stará o spouštění všech důležitých procesů aplikace. Dělí se na balíčky *Receivers* a *Variables*. První z balíčků, jak již název napovídá, obsahuje tzv. přijímače (receiver). Jedná se o komponenty, které umožňují registrovat a zachytit jak

¹ Background mode je režim aplikací, při kterém není ukončena jejich činnost, avšak některé funkce mohou být pozastaveny OS



Obrázek 5.1. Diagram balíčků a jejich rozdělení do jednotlivých vrstev aplikace.

systémové, tak aplikáční události. Takovou událostí může být například připojení na Wi-Fi nebo restartování operačního systému. Je také možné vytvářet a přijímat vlastní (aplikáční) události, například při dokončení určitého procesu.

Druhý balíček *Variables* obsahuje aplikáční proměnné a konstanty, tedy zejména url serveru a jeho koncových bodů pro různé služby.

■ 5.2.2 Client layer

V této vrstvě najdeme všechny komponenty aplikace, které se týkají bezprostřední interakce s uživatelem. Balíček *Account* zajišťuje například registraci a správu identifikačních údajů pro komunikaci se serverem. Balíček *Updater* se stará o kontrolu nově dostupných verzí aplikace na serveru a také o případné upozornění na ni přes uživatelské rozhraní. Poslední část vrstvy, balíček *Logger*, zaznamenává provozní údaje o chodu aplikace a jejich odeslání na server.

■ 5.2.3 Tracking layer

Vrstva Tracking je důležitá především z hlediska dat cestovního deníku. Balíček *Location* je odpovědný za záznam údajů GPS, jejich uložení do paměti a v případě řídicího signálu od některého z přijímačů z vrstvy Core také za odeslání dat na aplikáční server. Balíček *Mode* pak zpracovává údaje o aktuálním dopravním módu uživatele.

■ 5.3 Implementace

Pro přehlednost v následujících sekcích je nutné na začátku popsat některé důležité třídy a komponenty systému Android, kterých aplikace GPS Tracker využívá.

■ 5.3.1 Vybrané komponenty systému Android

Nejdůležitějším rozhraním operačního systému, které poskytuje všem procesům je *Context*. V něm se nachází globální informace o prostředí, ve kterém aplikace běží. Jeho prostřednictvím můžeme získat přístup k aplikáčním souborům a především pak spouštět nové procesy a vysílat řídicí signály pro přijímací *BroadcastReceivers*.

Jejich použití je založeno na návrhovém vzoru *Observer*, který definuje závislost 1:N mezi sledovaným objektem, někdy nazývaným jako *subject*, a sledujícími objekty - *observers*. Princip takové komunikace spočívá v tom, že pokud se změní stav sledovaného

objektu, jsou o této změně informováni všichni zaregistrovaní ze strany observers. Konkrétně v případě systému Android jsou příjemci takových zpráv právě objekty třídy `BroadcastReceiver`, které většinou využíváme pro sledování systémových událostí, případně si můžeme takovou událost definovat sami. Výhodou tohoto přístupu je fakt, že o spouštění zaregistrovaných přijímačů se stará sám operační systém, nemusíme proto kvůli tomu například zapínat aplikaci. Jedna se o klíčovou vlastnost z hlediska aplikace GPS Tracker, neboť umožňuje reagovat na změny v systému a to bez závislosti na uživateli. Objekty této třídy by však neměly být odpovědné za vykonávání delších procesů, jako je například zpracování dat. Jejich účelem je pouze zachytit signál v komunikačním kanálu a delegovat řízení na odpovídající části.

Jednou z takových částí je třída `Service` sloužící pro dlouhotrvající procesy, které nevyžadují interakci s grafickým rozhraním, tedy například pro stahování souborů z internetu nebo zpracování dat na pozadí. Jejím rozšířením je třída `IntentService`, která představuje základ pro asynchronní zpracování požadavků. Její životní cyklus je oddělen od hlavního vlákna aplikace, a slouží pro jednorázové vykonání akce na pozadí, po kterém je tento proces ukončen.

Další podstatnou částí jsou objekty třídy `Intent`. Ty poskytují abstrakci pro zapouzdření operace, kterou požadujeme po systému vykonat. Typickou ukázkou je použití takového objektu pro spuštění procesu (`Service`).

```
Context context = getApplicationContext();
Intent serviceExample = new Intent(context, SomeServiceClass.class);

context.startService(serviceExample);
```

S třídou `Intent` je úzce spjatá třída `PendingIntent`, která umožňuje poskytnout jiným aplikacím nebo procesům přístup k námi definované části kódu a vykonat ho. `PendingIntent` je tedy jakýsi token, kterým cílovému procesu předáváme referenci na námi určenou komponentu. Toho můžeme využít především pro spuštění procesů, které mají běžet na pozadí.

■ 5.3.2 Sledování polohy

Pro sledování polohy uživatele pomocí mobilního přístroje existují s ohledem na přesnost souřadnic tři možnosti. Zaprvé můžeme využít zabudovaného přijímače (tzv. *GPS provider*), který získává údaje ze satelitů GPS. Tento přístup zaručuje za optimálních podmínek nejvyšší přesnost záznamů, ta však závisí především na dobré viditelnosti oblohy a velikosti šumu signálu. Druhou možností je využít *Network provider*, který pozici odvozuje z hodnot signálu mobilního vysílače a také okolních Wi-Fi sítí. Jeho přesnost je velmi proměnlivá, pohybuje se zpravidla od desítek až po stovky metrů v závislosti na lokaci.

Poslední možností je využití kombinace obou providerů, při které můžeme dosáhnout vyvážené spotřeby baterie a relativně dobré přesnosti. Zjišťování aktuálních podmínek a přepínání mezi providerem by však bylo poměrně složité, naštěstí společnost Google nabízí v rámci tzv. služeb Google Play¹ tuto funkcionality pod názvem *Fused Location provider*.

Abychom mohli pozici zaznamenávat, musíme nejprve vytvořit objekt třídy `LocationRequest` se specifikovanými parametry:

- Interval - časový údaj požadované frekvence přijímání souřadnic GPS

¹ Google Play Services je balíček služeb v rámci obchodu s aplikacemi Google Play

- Priority - zde specifikujeme požadovanou přesnost záznamů, konkrétně:
 - High_accuracy - nejvyšší možná přesnost dostupná v danou chvíli, nejvyšší spotřeba baterie
 - Balanced_power - jedná se o určitý kompromis mezi spotřebou a přesností, využívá především *Network provider* a poskytuje přesnost v řádu stovek metrů.
 - Low_Power - nejnižší přesnost (10km), nejnižší spotřeba baterie
 - No_Power - pasivní mód ve kterém obdržíme pozici pouze pokud o ni požádá jiná aplikace
- Displacement - nejkratší požadovaná vzdálenost mezi dvěma GPS záznamy

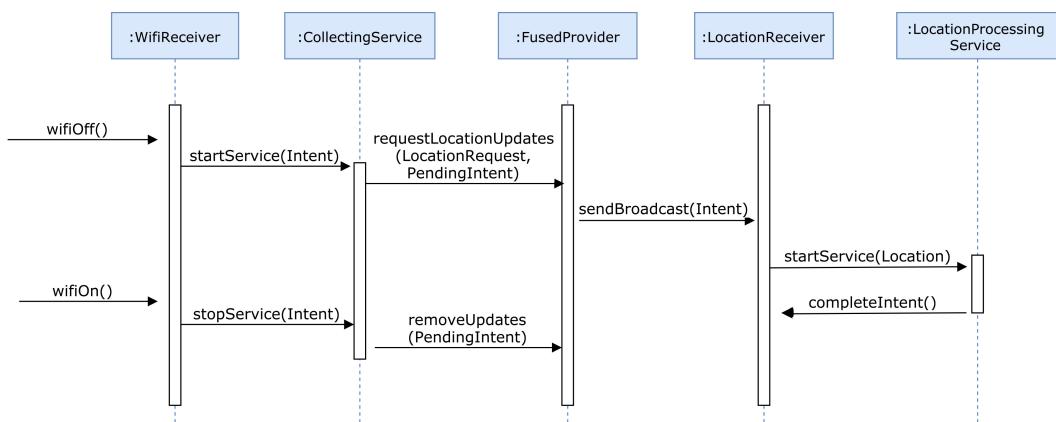
Takto specifikovaný požadavek *locationRequest* pak zaregistrujeme u klienta rozhraní *Fused Provider*, v ukázce pojmenovaném jako *fusedApiClient*:

```
Intent intent = new Intent(context, LocationReceiver.class);
PendingIntent pendingIntent = PendingIntent.getBroadcast(context,
    tokenID, intent, PendingIntent.FLAG_UPDATE_CURRENT);

LocationServices.FusedLocationApi.requestLocationUpdates(fusedApiClient,
    locationRequest, pendingIntent);
```

Pomocí objektu třídy *PendingIntent* vlastně zaregistrujeme náš přijímač *LocationReceiver* pro naslouchání o změnách v poloze. Parametr *FLAG_UPDATE_CURRENT* nám umožnuje sdělit systému, aby nevytvářel nový komunikační kanál mezi vysílačem *Fused Provider* a naším přijímačem, ale aby použil stávající spojení - to je důležité v situacích, kdy chceme pouze změnit některé parametry sledování pozice (např. interval).

Fused Provider nám pak v požadovaném intervalu doručuje objekty třídy *Location*, které kromě zeměpisných souřadnic obsahují také další informace jako rychlosť, nadmořskou výšku nebo přesnost záznamu v metrech. Tyto informace pomocí přijímače zachytíme, zpracujeme v odpovídající službě a vložíme do interního úložiště aplikace.



Obrázek 5.2. Diagram procesu zaznamenávání pozice.

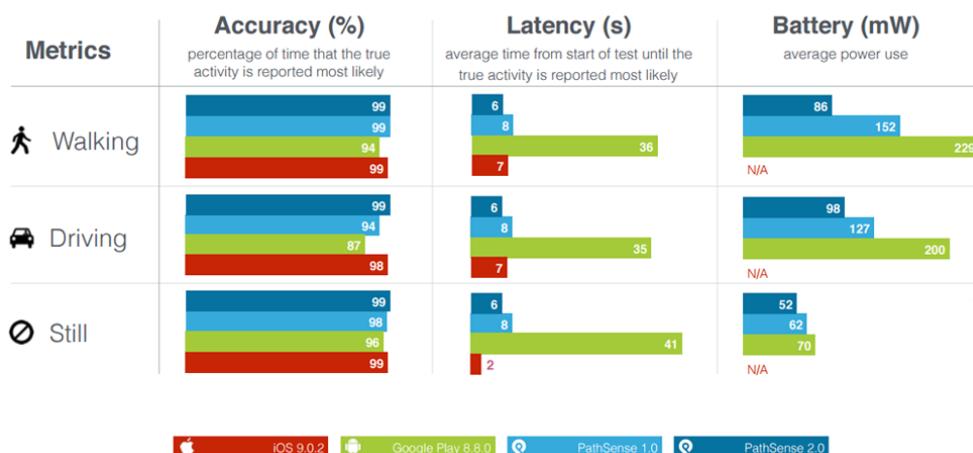
Popsaný proces sledování polohy je standardně řízen signálem o připojení k síti Wi-Fi. Pokud se tedy uživatel od sítě odpojí (začne cestu - např. vyjde z domova), je proces automaticky spuštěn pomocí *WifiReceiver*, při opětovném připojení se proces ukončí. Samozřejmě musíme počítat se situací, kdy se uživatel na Wi-Fi sítě standardně nepřipojuje. V takovém případě by se totiž sledování pozice neukončilo a zbytečně zatěžovalo baterii přístroje, proto je z těchto důvodů kontrolován čas posledního pohybu uživatele, a v případě, že nečinnost přesáhne dobu 5 minut, celý proces uspán. Více o tomto je popsáno v následující sekci.

5.3.3 Detekce dopravních módů

Algoritmus popsaný v sekci 4.2 počítá s tím, že budeme schopni detektovat základní dopravní módy (*Walk*, *Still*, *Vehicle*) a tedy i případné změny mezi nimi. Jednou z možností pro získání této informace je využití *Google Activity Recognition API*¹. Proces komunikace mezi knihovnou a naší aplikací je pak totožný jako v případě určování polohy, jen tentokrát nevytváříme objekt se specifikovanými parametry, ale definujeme pouze interval pro obnovování detekce.

V momentě kdy náš přijímač *TransportModeReceiver* obdrží od rozhraní informaci s detekovaným módem, je na tuto hodnotu nastavena proměnná, kterou využívá proces pro zpracování údajů GPS. Ke každému záznamu GPS tedy vytvoříme spojení s konkrétním dopravním módem a na server jsou tak tyto informace odesílány společně.

API od společnosti Google však není jediné nabízené, další možností je například PathSense API již zmíněné v sekci 4.2. To nabízí totožnou funkcionalitu, vykazuje však výrazně lepší časy pro potřebnou detekci a také nižší spotřebu energie. Jeho další výhodou je podpora operačního systému iOS, což by v případě možného budoucího rozšíření aplikace GPS Tracker i pro tento systém znamenalo snadnější a konzistentnější vývoj.



Obrázek 5.3. Porovnání výkonnostních parametrů knihoven *Google Activity Recognition API* a *PathSense API*. Převzato z <https://pathsense.com>

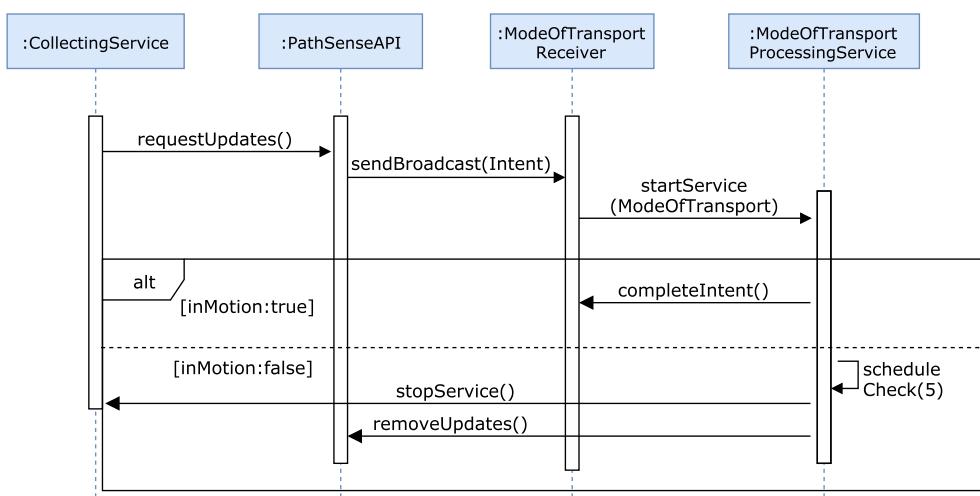
Jak již bylo v minulé sekci naznačeno, jedna uskutečněná cesta nemusí být vždy zakončena připojením na Wi-Fi síť. V případě, že bychom na toto spoléhali, mohla by nastat situace, kdy se budeme domnívat, že je uživatel stále na cestě, ale přitom již bude dálno na místě cílové aktivity.

Představme si například situaci, kdy se uživatel zdržuje delší dobu na stejném místě, kde žádná síť není k dispozici (např. v zaměstnání nebo restauraci). I v případě nehybného přístroje by systém Android stále ověřoval polohu, byť pravděpodobně s menší spotřebou. V těchto situacích aplikace GPS Tracker kontroluje dobu nečinnosti, tedy konkrétně pokud detekuje jako dopravní mód *Still*, ověřuje po dobu následujících 5 minut zdali nebyl přístroj znova uveden do chodu. Po překročení této hranice jsou procesy sledování polohy i detekce dopravních módů automaticky uspány. Abychom dokázali

¹ Musíme podotknout, že v kontextu rozpoznávání dopravních módů chápe firma Google slovo „Aktivita“ jako dopravní prostředek, nikoliv tedy jako činnost, kterou jsme definovali v kapitole o aktivitně-cestovních modelech.

zachytit uživatelův případný další pohyb, je v opakujícím se intervalu 5 minut spouštěn proces detekce dopravních módů, vždy však pouze na pár vteřin. Tato zátěž pro baterii je srovnatelná například se synchronizací nových e-mailu nebo dat ze sociálních sítí, umožňuje však v případě zaznamenaného pohybu opětovně spustit proces sledování polohy.

Obdobným způsobem funguje sledování pohybu po připojení na Wi-Fi síť, neboť může nastat situace, kdy se uživatel připojí na pohybující se síť - například ve vlaku. Na rozdíl od předchozího případu je zde však interval pro případné uspání zkrácen na dobu jedné minuty, a to z důvodu situací, kdy se uživatel připojí na rozlehlou Wi-Fi síť například v kancelářském komplexu, ale nějakou dobu ještě setrvává v pohybu (než se dostaví do své kanceláře). Pokud by se na této cestě zdržel více než 5 minut, mohla by aplikace takový pohyb klasifikovat jako pokračování v cestě, a zbytečně by tak po dobu dalších 5 minut zaznamenávala polohu.



Obrázek 5.4. Diagram uspání procesů při nečinnosti uživatele

5.3.4 Odesílání dat na server

Podstatnou částí aplikace je komunikace se serverem. Ta probíhá při několika situacích, zaprvé při uživatelově registraci, kdy je ze serveru vrácen unikátní klíč user_id, kterým se aplikace hlásí později při odesílání zaznamenaných dat, což tvoří druhý případ komunikace. Dále připojení na server probíhá při odesílání logů a kontrole nových verzí.

Každá tato operace probíhá uvnitř instancí třídy *AsyncTask*, která vykonává příkazy asynchronním způsobem a neblokuje tak hlavní vlákno aplikace. Samotná komunikace se serverem probíhá pomocí protokolu HTTP a jeho dotazovacích metod, konkrétně metody GET pro získání dat a metody POST pro odeslání dat. Ve druhém zmíněném případě jsou data odesílána ve formátu JSON¹. Na celkový dotaz se můžeme podívat na následující ukázce. Pro zjednodušení byly vynechány některé parametry GPS záznamů:

```

POST http://serverUrl/endpoint HTTP/1.1
Host: 127.0.0.1:3000
Content-type: application/json; charset=utf-8
Content-Length: 216
{
  ...
}
  
```

¹ Formát dat JSON (JavaScript Object Notation) slouží k jednoduché výměně dat po internetu. Jeho výhodou je snadná čitelnost pro člověka a platformová nezávislost.

```

"user_id": 0, "location_values" : [
  { "accuracy": 20,
    "latitude":50.0755,
    "longitude":14.4378,
    "transport_mode":"Vehicle"
  },
  { "accuracy": 41,
    "latitude":50.0698,
    "longitude":14.4215,
    "transport_mode":"Walk"
  }
]
}

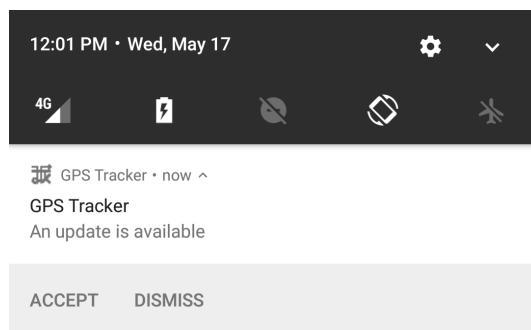
```

Případné ztrátě dat při nenadálých událostech na serveru nebo při přenosu je zabráňeno tak, že aplikace čeká na odpověď signalizující úspěšné uložení všech záznamů, a teprve poté je vymaže z interního úložiště. V případě neúspěšného odeslání, například z důvodu odpojení od Wi-Fi sítě, se tak data jednoduše odešlou při příštém úspěšně navázaném připojení.

■ 5.3.5 Automatické aktualizace

Především v počáteční fázi vývoje bylo nutné pro potřeby testování vyvinout systém automatických aktualizací. Jelikož testování v reálných podmínkách probíhalo více uživateli na rozdílných zařízeních (více v následující sekci), bylo by poměrně nepohodlné v případě opravy chyb vždy žádat uživatele o novou instalaci aplikace. Z tohoto důvodu byl implementován systém automatické distribuce nových verzí systému. Po připojení na Wi-Fi síť tak aplikace odešle na server dotaz pomocí metody GET, která vrátí číslo nejnovější verze umístěné na serveru. Pokud se toto číslo neshoduje s aktuálně nainstalovanou verzí, je uživateli skrze notifikaci hlavního rozhraní operačního systému nabídnuta možnost aktualizace.

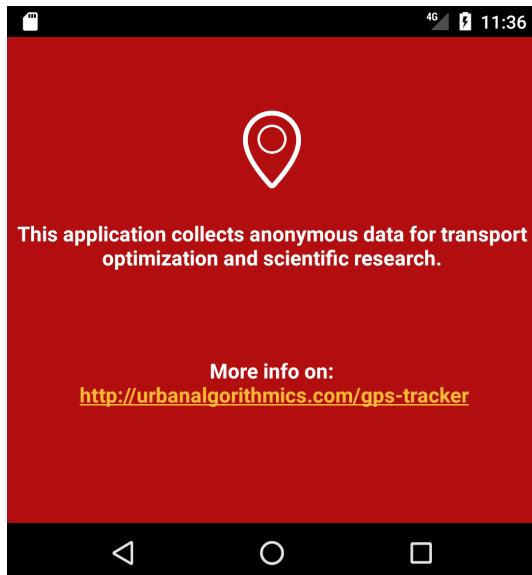
Pro tyto potřeby jsou vytvořeny vlastní události, kterým naslouchá třída *UpdateService*. Uživatel tedy musí pouze potvrdit stažení nového souboru, případně tuto akci odmítnout a smazat tak danou notifikaci.



Obrázek 5.5. Ukázka notifikace při dostupné aktualizaci

5.4 Zhodnocení a testování aplikace

Aplikace GPS Tracker¹ poskytuje podporu pro operační systém Android od verze 4.1, tedy pro více než 95 % uživatelů všech verzí. Výsledné procento potenciálních uživatelů může být do jisté míry sníženo závislostí na nainstalovaném balíčku *Google Play Services*, vzhledem k jeho silné provázanosti s obchodem aplikací *Google Play*, kde počet jeho stažení přesahuje 1 miliardu, lze však předpokládat jeho přítomnost v cílovém zařízení.



Obrázek 5.6. Ukázka uživatelského rozhraní aplikace GPS Tracker.

Prvotního uživatelského testování se zúčastnilo celkem 6 osob s rozdílnými typy mobilních přístrojů. Testovací scénář byl vzhledem k účelu aplikace jediný - aplikaci nainstalovat, zaregistrovat se a poté aplikaci vypnout. Následně byl pomocí logovacích zpráv ověřován proces sledování polohy na pozadí. Tato fáze pomohla odhalit chyby v systému plynoucí především z rozdílných verzí operačních systémů a přispěla k nalezení optimálních hodnot nastavení citlivosti a frekvence přijímání souřadnic GPS a detekce dopravních módů. Zároveň také sloužila pro testování uživatelského rozhraní, byť v případě aplikace GPS Tracker je tato skutečnost omezena pouze na případ registrace.

Model přístroje	Verze OS	API level
BB Classic	4.3.1	18
MI 25	4.1	16
SM-G930F	6.0.1	23
D5803	5.0	21
LG-D855	5.0	21
SM-G360F	5.1.1	22

Tabulka 5.1. Tabulka parametrů testovacích zařízení. API level odpovídá poskytovanému rozhraní služeb dané verze operačního systému.

¹ <http://urbanalgorithmics.com/gps-tracker>

Jelikož většina částí aplikace využívá služeb třídy *Service*, tedy procesu vykonávajícího příkazy bez interakce s uživatelem, bylo nutné ověřit chování těchto částí integračními testy. Pro tyto případy existuje API s názvem Android Testing Support Library využívajícího frameworku JUnit 4. Tímto způsobem však bohužel nejde otestovat chování tříd *IntentService*¹, bylo proto nutné jejich funkcionality otestovat převážně unit testy². Pro potřeby testování komunikace se serverem bylo využito zástupného objektu (tzv. *mock objectu*), který simuluje potřebné chování pro unit testy.

S ohledem na možnost detekce případných chyb v systému bylo také nutné vypořádat se s problémem vzdáleného hlášení problému a pádů aplikace. Systém Android zaznamenává všechny zprávy, ať již systémové nebo aplikační, pomocí nástroje *Logcat*. Abychom nemuseli s odesláním hlášení obtěžovat uživatele, byla do projektu přidána knihovna ACRA³, která v případě výskytu chyby nebo pádu aplikace odešle na cílový server všechna požadovaná data.

¹ <https://developer.android.com/training/testing/integration-testing/service-testing.html>

² Unit testy ověřují správnost vykonávání kódu na úrovni nejmenších jednotek - tedy nejčastěji metod.

³ <https://github.com/ACRA/acra>

Kapitola 6

Server

Mobilní aplikace GPS Tracker tvoří klientskou část poskytovaného systému a je odpovědná za sběr dat cestovních deníků. Zbývá proto vytvořit další komponentu systému pro aktivitně-cestovní průzkum, kde se bude odehrávat zpracování a vyhodnocení - tedy serverovou část.

6.1 Analýza požadavků a rizik

Je potřeba si nejprve uvědomit, jaké důležité prvky musíme pro takový proces zajistit, a poté analyzovat dostupné možnosti i technologie, neboť vhodnost jejich použití se může lišit podle daného účelu.

První důležitou vlastností serveru musí být schopnost přijímat HTTP požadavky ze strany klienta. Je totiž klíčové zajistit obousměrnou komunikaci mezi klientem a serverem na takové úrovni, aby nebylo nutné získávat potřebná data od participanta průzkumu jinými prostředky. Z toho tedy plyne druhý požadavek na server, a to schopnost identifikovat jednotlivé uživatele. S ohledem na potřeby aktivitně-cestovního průzkumu je také nutné uchovávat požadované demografické a socio-ekonomické ukazatele spolu s jejich vazbami k uskutečněným cestám. Pro tyto potřeby se jako optimální jeví použití relační databáze, server by tedy měl být schopný takovou databází obslužit. Dále potřebujeme nad získanými daty spouštět samostatné procesy pro vyhodnocování aktivit i zvolených dopravních prostředků jednotlivých cest.

Pokud se zamyslíme nad riziky, jeví se jako největším problémem případný výpadek serveru a s tím související údržba. Toto riziko můžeme minimalizovat například využitím tzv. cloudového řešení, které nám správu velmi usnadní, navíc s sebou většinou přináší i garanci v podobě zaručené doby provozuschopnosti. Z důvodu možné ztráty databázových dat je také nutné provádět pravidelné zálohy, ideálně na jiném, od serveru fyzicky odděleném úložišti. Dalším rizikem může být nesprávný HTTP dotaz od klienta, který by mohl například způsobit nekonzistentní data.

6.1.1 Aplikační server

Pokud bychom se rozhodli pojmit server pouze jako cílové místo pro příjem a ukládání uživatelských dat, pravděpodobně by nám k tomuto účelu dobře posloužil jednoduchý webový server. S ohledem na možná budoucí rozšíření, případné propojení systému se službami třetích stran a tedy kombinaci několika různých technologií se jako vhodnější jeví využití komplexního cloudového řešení, a to z několika důvodů. V první řadě si musíme uvědomit, že aktivitně-cestovní průzkumy pro velké městské oblasti potřebují pro účinné plánování zajistit stovky až tisíce respondentů - potenciálních uživatelů systému. To s sebou samozřejmě přináší zvýšené nároky na obsluhu uživatelů a bezchybné zpracování všech HTTP požadavků a zpráv. V další řadě je také potřebné zajistit dostatečnou kapacitu pro ukládání uživatelských záznamů a pro data GTFS. Nashromázděná data však nemusí sloužit jen pro potřeby detekce aktivit a dopravních módů, ale mohou nad nimi být vykonávány i další procesy, jako například vizualizace. Ve všech případech

se jedná o činnosti, u kterých je důležité zajistit dynamickou škálovatelnost, a přesně pro tyto účely jsou cloudová řešení navržena.

Při rozhodnutí o volbě konkrétního poskytovatele musíme vzít v potaz skutečnost, že cloudová řešení se v zásadě rozdělují do tří modelů služeb:

- Software as a Service (SaaS)
- Platform as a Service (PaaS)
- Infrastructure as a Service (IaaS)

První z jmenovaných služeb nám poskytuje přímý přístup k aplikaci bez nutnosti instalace. Nemusíme se také zabývat nastavením nebo navazováním spojení, většinou nám ke spuštění stačí pouze webový prohlížeč. Příkladem takové služby je například známé úložiště *Dropbox* nebo oblíbený textový editor *Google Docs*. Pro potřeby práce je však tato služba nedostatečná z důvodu omezených možností použitých technologií a prakticky nulové možnosti kontroly nad prováděním programu.

Platform as a Service poskytuje vývojářům větší množství funkcí (skrze svá API) a rozšířené možnosti nastavení, jak však již z názvu vyplývá, je závislá na zvolené platformě. Nabízí uživatelům určitý druh frameworku, tedy pracovní prostředí s potřebnými nástroji. Na druhé straně ale musíme počítat s omezeními vyplývajícími z fixace např. na daný programovací jazyk. Obecně má tato služba omezenou míru kontroly nad prováděným procesem, což může být rizikem z hlediska správy uživatelsky citlivých dat. Toto řešení naopak může být vhodné pro situace, kdy vyvíjíme software s ohledem na nástroje daného poskytovatele a nechceme se zabývat správou, nasazením a aktualizacemi našeho programu. Mezi poskytovateli můžeme jmenovat například Heroku nebo Google App Engine.

Poslední služba IaaS nabízí největší možnosti z hlediska vývoje software i kontroly jeho životního cyklu. Uživatel má plně ve své režii virtuální stroj i poskytnutou infrastrukturu. Velkou výhodou je dynamické přidělování výpočetního výkonu i dalších zdrojů podle potřeby. Zjednodušeně by se dalo říct, že poskytovatel služby IaaS nám vlastně nabízí kus svého fyzického hardwaru skrze virtuální prostředí. Nemusíme se tedy starat o jeho nákup ani údržbu a platíme pouze za využitý výkon. Patrně nejznámějšími poskytovateli jsou Windows Azure, Amazon AWS nebo Google Compute Engine.

Při rozhodování o volbě konkrétní služby hrála největší roli především možnost kontroly nad spouštěním a nasazováním procesů, míra škálovatelnosti v případě většího počtu uživatelů a v neposlední řadě také kontrola nad uživatelskými daty. Z těchto důvodů byla zvolena služba IaaS.

Jako konkrétní poskytovatel pak byl vybrán Amazon AWS z důvodu dobrých zkušeností při předchozím vývoji a velkému množství přidružených služeb. Pro tyto potřeby nabízí tzv. *EC2* instance lišící se pouze výkonnostními charakteristikami. Na každou takovou instanci můžeme nasadit jednu z mnoha předinstalovaných platform s potřebnými nástroji.

6.2 Popis architektury

Tato sekce popisuje všechny komponenty serverové části systému pro aktivitně-cestovní průzkum.

6.2.1 Webový server

První z komponent je odpovědná za příjem a zpracování HTTP požadavků od klienta. Pro tyto účely byl na instanci ec2 serveru vytvořen jednoduchý webový server

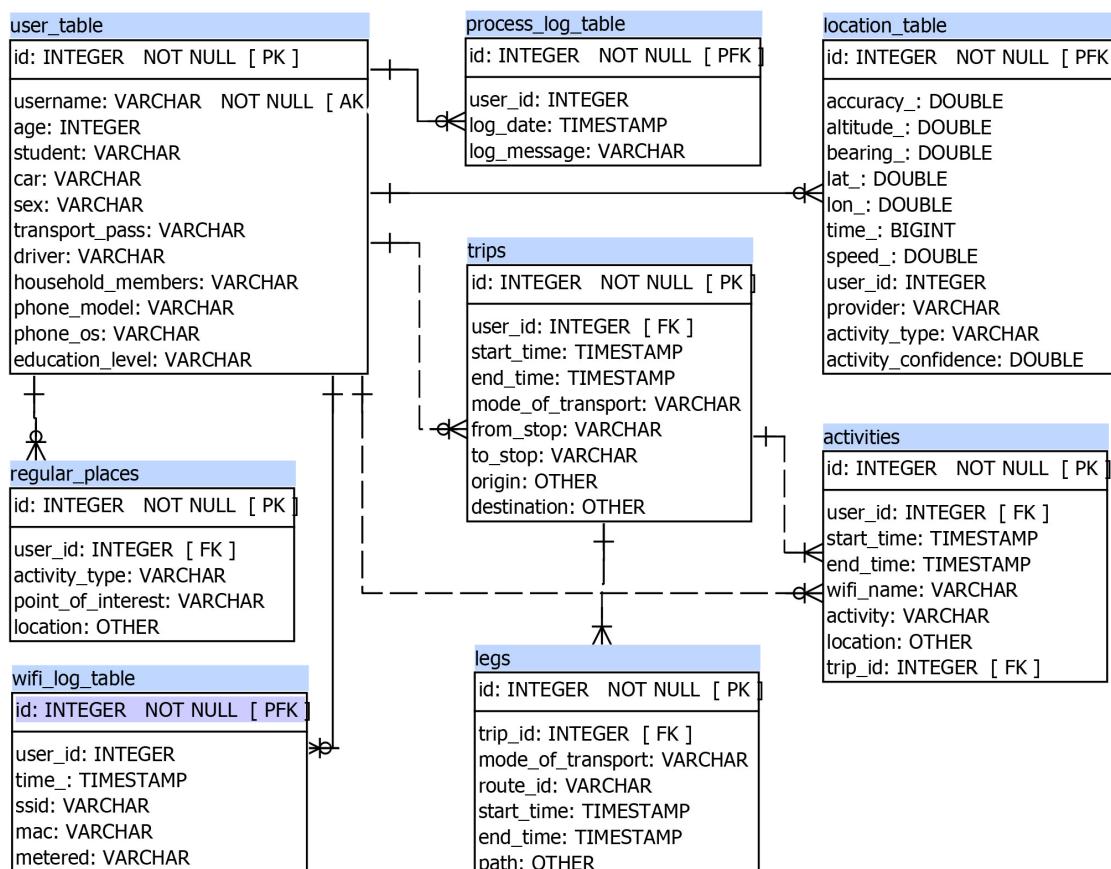
za využití technologie Node.js¹. Ta je sice založena na jazyku JavaScript, který je primárně určen pro běh programů v klientské části, přináší ale běhové prostředí *V8 engine* vytvořené firmou Google pro prohlížeč Chrome. Node.js využívá událostmi řízeného zpracování, jednotlivé požadavky tedy zpracovává neblokujícím asynchronním způsobem. Jeho výhodou je snadné propojení s velkým množstvím JavaScriptově založených knihoven, například pro zpracování souborů JSON nebo pro komunikaci s databází.

6.2.2 Databáze

Další částí je relační databáze pro záznam GTFS dat i údajů GPS z uskutečněných cest. Pro realizaci této části byla vybrána open-source databáze PostgreSQL, a to především z důvodu existujícího rozšíření PostGIS, které přináší množství funkcí pro zpracování a úpravu prostorových dat, což je přesně případ údajů z cestovních deníků.

V rámci této databáze byla vytvořena tři schémata:

- raw_data - určené pro uchování dat souvisejících s uživatelem a jeho cestami
- trip_data - určené pro zpracované a vyhodnocené cesty
- gtfs - úložiště GTFS záznamů o dopravní síti

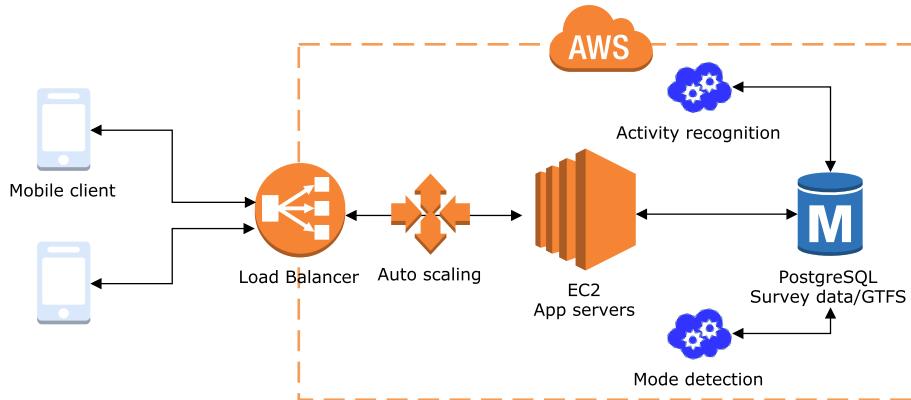


Obrázek 6.1. Databázové schéma všech tabulek systému pro aktivitně-cestovní průzkum.

¹ <https://nodejs.org/en/>

6.2.3 Procesy pro vyhodnocování

Pro potřeby vyhodnocování dat z cestovních deníků mohou být spouštěny dva samostatné procesy vytvořené v programovacím jazyce Java, které jsou umístěny na aplikačním serveru. K potřebným datům z databáze se však lze připojit i vzdáleně skrze rozhraní serveru, lze tedy tyto procesy v případě potřeby spouštět i na jiných instancích.



Obrázek 6.2. Architektura poskytovaného systému.

6.3 Implementace

6.3.1 Komunikace klient - server

Mobilní aplikace GPS Tracker odesílá všechny požadavky a dotazy na server formou HTTP dotazů s přiloženými daty ve formátu JSON. Každá taková zpráva může směřovat na rozdílný koncový bod, neboť chceme zajistit odpovědnost konkrétních metod pouze za jednu funkcionalitu. Pro tyto potřeby webový server využívá knihovny *router*, která dokáže přesměrovávat dotazy směřující na rozdílná url serveru a rozlišuje je také podle typu dané HTTP metody.

Pokud se tedy například uživatel se svým mobilním přístrojem připojí na síť WiFi, odešle klient dotaz na případnou aktualizaci a požádá o navrácení dat ve formátu JSON:

```

GET serverUrl/versionCode HTTP/1.1
.
.
Accept: application/json
  
```

Server tento požadavek přijme a router jej předá příslušné obslužné metodě, která číslo verze obalí do požadovaného formátu:

```

router.get('/versionCode', function (req, res) {
  response.set('Content-Type', 'application/json');
  return res.status(200).json(version);
});
  
```

Obdobným způsobem funguje i metoda POST pro vkládání dat, v tomto případě se však v rámci metody ještě navazuje spojení s klientem DB.

■ 6.3.2 Detekce dopravních módů

V sekci 4.2 jsme popsali navrhovaný algoritmus pro rozlišení konkrétních dopravních módů, tedy nejen pro *Vehicle*, ale i například *Bus*, *Car*, *Metro*.

Pro potřeby tvorby našeho grafu a nad ním prováděných operací byly vytvořeny zástupné třídy, jejichž instance představují konkrétní objekty v síti veřejné dopravy.

- Třída *Graph* představuje nejvyšší míru abstrakce pro model dopravní sítě
- Třída *Node* zastupuje konkrétní fyzické místo jedné zastávky, její parametry tedy obsahují souřadnice GPS, GTFS identifikátor a seznam linek obsluhující danou zastávku
- Třída *Route* představuje konkrétní linku v síti veřejné dopravy, obsahuje svůj identifikátor (číslo linky) a seznam konkrétních spojů, které ji obsluhují.
- Třída *Trip* zastupuje konkrétní spoj v síti, obsahuje tedy především informace o časech odjezdů a příjezdů k jednotlivým zastávkám své trasy.

Dále byly z důvodu potřebné abstrakce pro „nefyzické“ části modelu, tedy konkrétně cestu, GPS souřadnice a dopravní mód vytvořeny zástupné třídy:

- ModeOfTransport - výčtový typ *Enum* pro dopravní mód
- GPSPoint - zástupný objekt pro souřadnice GPS. Tato třída byla vytvořena především z důvodu vyhodnocování jednotlivých úseků cest, obsahuje tedy informace o dopravním prostředku, kterým se uživatel na daný bod dopravil, a dále také časovou stopu záznamu
- Leg - tato třída představuje konkrétní úsek dané cesty, který byl vykonán jedním dopravním prostředkem

Na začátku zpracování každé cesty je nalezení seznamu potenciálních přestupních zastávek. Tyto kandidáty vyhledáme v grafu podle jejich identifikátorů a utvoříme mezi nimi vazby v případě existujících spojů. To je velmi jednoduché a rychlé, neboť u každých dvou zastávek stačí porovnat pouze čísla obslužných linek, kterých standardně bývá malý počet. Poté je každé takovéto spojení ohodnoceno celkovým zpožděním v minutách, tedy rozdílu skutečného času oproti udávanému dle jízdního řádu.

Algoritmus pro hledání nejvhodnější cesty (tedy takové, která minimalizuje zpoždění mezi nástupní a výstupní zastávkou) je pak spouštěn na jednotlivých podgrafech začínajících v možných počátečních zastávkách. Těch může být více, protože na cestě MHD nemusí vždy platit, že na sebe zastávky plynule navazují, a cestující proto musí ujít vzdálenost mezi nimi například pěšky.

Každý takto vzniklý úsek cesty (tedy instance třídy Leg) pak prochází několika stupni klasifikace. V první úrovni se zjišťuje pravděpodobnost, že daná trasa se uskutečnila pomocí veřejné dopravy, a to na základě porovnání časů pro jednotlivé dopravní prostředky. Pro MHD tuto znalost máme zásluhou dat GTFS, dobu předpokládaného trvání jízdy autem pak ověříme dotazem přes plánovač Google Maps API. Je nutné podotknout, že tento údaj je pouze orientační, může se velmi lišit pro různé denní doby z důvodu dopravních špiček apod. Umožní nám však v případě výrazného rozdílu mezi časy odstranit nepravděpodobné kandidáty. Tento princip je využíván především v situacích, kdy se zaznamenaný čas blíží času auta a zároveň je rychlejší než deklarují intervaly jízdních řádů.

Druhou fází klasifikace je ověřování podobnosti trasy dané linky se skutečně zaznamenanou. Abychom neprocházeli všechny body trasy jednotlivě, jsou ověřovány pouze výskyty na zastávkách mezi danými dvěma koncovými body, samozřejmě s jistou mírou tolerance nepřesnosti záznamů GPS. V případě metra je tento postup vyneschán, neboť

praxe ukázala, že signál v podzemí je velmi nevyzpytatelný a nelze se na tyto záznamy spoléhat. Na druhou stranu právě z tohoto důvodu je metro poměrně snadno detektovatelné, protože v případě jakéhokoliv jiného prostředku bychom měli k dispozici více záznamů.

Poslední fází klasifikace je propojování jednotlivých úseků v ucelenou trasu. Při tomto procesu můžeme korigovat některé špatné odhady tím, že porovnáme vývoj celé cesty. Je například nepravděpodobné, že by cestující začal trasu ve vozidle, poté na malý úsek přesedl do MHD (byť by se shodovaly časové údaje i trajektorie), a zbytek trasy strávil opět ve voze.

Takto ohodnocená trasa je poté uložena do databáze, společně s propojením na jednotlivé úseky i s informacemi o dopravním módu, případně zvolených linkách MHD.

■ 6.3.3 Klasifikace aktivit

Určování cílových aktivit jednotlivých tras je prováděno na základě porovnání časové „atraktivity“, viz sekce 4.4. V současné době systém rozlišuje tři základní druhy aktivit:

- Home - dlouhodobě prováděná aktivita započatá ve večerních hodinách a ukončena v ranních hodinách.
- Work - střednědobá až dlouhodobá aktivita odehrávající se přes den
- Leisure - krátkodobá aktivita vykonává spíše v odpoledních hodinách

Pro zkvalitnění odhadu je také využito detekování tzv. pravidelných lokalit a aktivit. Pokud je například konkrétní místo ve spojitosti s daným uživatelem detekováno jako místo výkonu zaměstnání, můžeme pak takto ohodnotit i nečekané výskytu v této lokalitě mimo předpokládanou dobu. Pro budoucí rozšíření by také tato informace mohla sloužit k lepší detekci aktivit jiných uživatelů, kteří například dojíždějí na stejné místo.

Kapitola 7

Vyhodnocení

Abychom mohli určit míru úspěšnosti poskytovaného systému na reálných datech, požádali jsme v rámci prvotního uživatelského testování aplikace participanty, aby své cesty kromě automatizovaného záznamu přes mobilní aplikaci také zapisovali ručně. Z důvodu vytvoření testovacího grafu pouze z dat GTFS pro město Praha do validace nebyly zahrnutu cesty uskutečněné na jiném území. Tyto údaje jsou samozřejmě součástí vyhodnocených záznamů v databázi, nelze u nich však určit míru úspěšnosti predikce, neboť je u nich zaznamenán pouze obecný dopravní mód a nikoliv konkrétní. Nelze proto garantovat, že by se u těchto dat v případě spuštění procesu detekce MHD pro dané území dopravní módy nezměnily.

Celkově bylo pro tyto potřeby nashromázděno 305 jízd od 8 lidí z období prosince 2016 až dubna 2017. Každá jízda se přitom mohla skládat z několika částí, celkově tak všechny testovací jízdy vytvořily 355 úseků, na kterých došlo k samostatnému vyhodnocení dopravního módu.

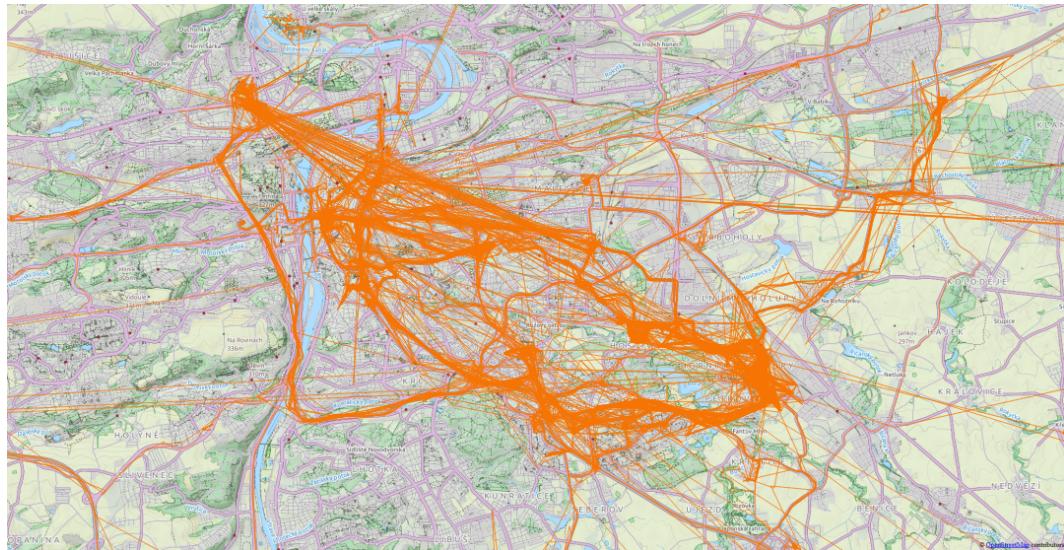
Vyhodnoceno	Auto	Bus	Tram	Metro	Chůze
Úspěšně	125	64	42	40	50
Neúspěšně	4	5	2	1	5
Úspěšnost[%]	96,90	92,75	95,45	97,56	96,90

Tabulka 7.1. Tabulka rozložení úspěšnosti detekce dopravních módů pro jednotlivé prostředky.

Mezi těmito záznamy chybí celkem 17 jízd, u kterých nebylo možné rozhodnout o dopravním módu. Všechny tyto jízdy jsou od 2 uživatelů, při jejichž cestách mobilní přístroj nebyl schopný věrohodně klasifikovat základní dopravní mód. To může být způsobeno jejich stářím a tedy nepřesnými či chybějícími senzory, které využívá knihovna PathSense API. Druhým ovlivňujícím faktorem mohla být možnost vypnutého GPS přijímače či přepnutí mobilního telefonu na pasivní sledovací mód, čemuž by u jednoho uživatele nasvědčovaly i velmi nepřesné souřadnice pozic. Pravděpodobně toto chování nemá spojitost s verzí operačního systému, neboť jiný participant se stejnou verzí vykazoval uspokojivé výsledky.

Na těchto jízdách byl poté spuštěn proces určení aktivity, prozatím jedné ze tří základních - Zaměstnání (Work), Domov (Home) a Volnočasová aktivita (Leisure). Celkově bylo pro těchto 8 uživatelů rozpoznáno 20 lokalit z obvyklým výskytem aktivit a určeno 176 aktivit.

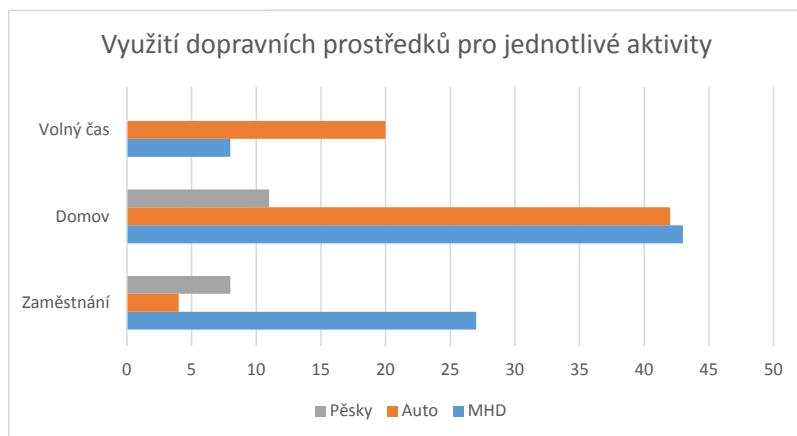
Při validaci rozpoznaných míst se ukázalo, že u jednoho uživatele systém chyběně vyhodnotil aktivitu „Vzdělávání“ na aktivitu „Zaměstnání“, a to celkem v 6 případech. Pokud se zamyslíme nad časovou charakteristikou těchto dvou aktivit, je to pochopitelné - lidé mohou trávit srovnatelný čas ve škole jako v zaměstnání, navíc ve velmi podobné denní době. Proto bylo zpětně využito dvou důležitých informací ke korekci výsledku.



Obrázek 7.1. Vizualizace uskutečněných jízd pro potřeby testování.

Zaprvé se ověřilo, jestli daná osoba není studentem, a zdali není v blízkosti této lokality bod zájmu s typem „Vzdělávání“. V obou případech bylo kritériu vyhověno a aktivita tak upravena na správný typ.

Celkově tak bylo u 305 jízd detekováno 100 aktivity typu „Domov“, 42 typu „Zaměstnání“, 28 typu „Volnočasová aktivita“ a 6 typu „Vzdělávání“.



Obrázek 7.2. Graf využití jednotlivých dopravních prostředků pro dané aktivity.

Kapitola 8

Závěr

Cílem práce bylo vytvořit nástroj, který by výrazně zjednodušil možnosti nasazení aktivitně-cestovních průzkumů na reprezentativní vzorek městské populace. Aby výsledky takového průzkumu měly dostatečnou vypovídající hodnotu a mohly se tak stát relevantním základem celého aktivitního modelu dopravy, potřebují většinou zajistit velký počet respondentů na delší časový interval. Připravit celý proces sběru i zpracování dat vytváří enormní náklady z hlediska času i lidských zdrojů. Námi poskytovaný systém umožňuje část těchto nákladů snížit a zároveň také minimalizovat chybovost vzniklou lidským faktorem hned z několika důvodů.

Zaprve při tradičním způsobu dotazování formou cestovních deníků hrozí skutečnost, že lidé zapomenou potřebné údaje zapisovat, případně je tato činnost po krátkém čase omrzí. Taková data pak mohou být velmi nepřesná a od některých respondentů se informace po úspěšném zahájení průzkumu nepodaří získat zpět vůbec. Naše mobilní aplikace toto riziko výrazně snižuje a především nabízí větší přesnost záznamů zásluhou využití přijímače GPS. Toto řešení navíc poskytuje velký potenciál rozšíření do budoucna, například možnosti vizualizace či zobrazení statistik jednotlivých cest zatím nebyly naplno využity. Velkým příslibem je také vývoj v oblasti mobilních technologií a využívání dat z jejich senzorů, který je znatelný především v posledních letech.

Samotné implementaci předcházela rozsáhlá analýza dostupných možností a souvisejících prací na téma dopravních modelů a průzkumů. Jako nejlepší současnou možností se ukázalo využití kombinace nashromážděných dat se znalostí o dopravní sítí MHD.

Hlavní přínos práce vidíme v navrhnutém a úspěšně implementovaném způsobu rozlišování konkrétních dopravních prostředků ze zaznamenaných dat, který na testovací sadě dosahoval přesnosti přesahující 90 % pro všechny módy. Další nespornou výhodou systému je automatizovaný sběr pozicičních údajů prostřednictvím mobilní aplikace, která byla úspěšně otestována na několika odlišných přístrojích s rozdílnými verzemi operačního systému Android. Navrhovanou architekturu serverové části se podařilo úspěšně nasadit v rámci cloudového prostředí, a vytvořit tak ve spojitosti s klientem ucelený nástroj. Celý systém byl také otestován na datech získaných v reálném prostředí městského provozu.

Při vývoji se ukázalo, že největším problémem navrhovaného systému může být jeho energetická náročnost na klientský mobilní přístroj. S ohledem na tuto skutečnost byly provedeny určité optimalizace, například uspávání procesů v případě nečinnosti nebo využití signálu Wi-Fi a mobilní sítě pro zajištění dostatečně přesných souřadnic v případě vypnutí přijímače GPS. Byly porovnány dostupné varianty knihoven pro rozpoznávání základních dopravních módů, ze kterých se nakonec jako nejlepší z hlediska spotřeby ukázalo PathSense API. V této oblasti však spatřujeme do budoucna velký potenciál, za zmínku stojí například možnost rozpoznání opakujících se stereotypů v přepravním chování, tedy určení obvyklých tras pro danou denní dobu, a na základě těchto odhadů poté snižovat zátěž kladenou na klienta.

8. Závěr

Věříme, že výsledky práce mohou přinést zlepšení procesu dopravního plánování, jenž se v posledních letech stává celospolečenským problémem a dotýká se každodenního života nás všech.

Literatura

- [1] MEJSTRÍK, Jiří. *Dojíždka a vyjíždka do zaměstnání do/z hl. m. Prahy*. Praha: Institut plánování a rozvoje hlavního města Prahy, Kancelář strategie a rozvoje, 2014. ISBN 978-80-87931-54-7.
- [2] JANOŠ, Vít. Úvod do předmětu přepravní vztahy [online]. [cit. 2017-05-09]. Dostupné z:
http://techno.kvalitne.cz/1_Uvod_prep_vztahy.pdf Studijní materiál. Ústav řízení dopravních procesů a logistiky ČVUT v Praze, Fakulta dopravní
- [3] GREAVES, S. et al. *A Web-Based Diary and Companion Smartphone app for Travel/Activity Surveys*. In: Transportation Research Procedia, 2015. p. 297-310. ISSN 2352-1465. DOI 10.1016/j.trpro.2015.12.026.s
- [4] COLE, Stuart. *Applied transport economics: policy, management and decision making*. 3rd ed. London: Kogan Page, 2005. ISBN 0-7494-4102-X.
- [5] RICHARDSON, A.J., ELIZABETH S. AMPT a ARNIM H. MEYBURG. *Survey methods for transport planning*. Melbourne, Vic., Australia: Eucalyptus Press, 1995. ISBN 978-0-6462-1439-9.
- [6] SVOBODA, Vladimír a SVÍTEK, Miroslav *Telematika nad dopravními sítěmi*, vyd. Praha: ČVUT, 2004, ISBN 80-01-03087-3
- [7] TICHÝ, Tomáš. Řídicí systémy dopravy: Dopravní telematika [online]. 2004 [cit. 2017-05-09]. Dostupné z:
<http://www.lss.fd.cvut.cz/Members/tichy/dokumenty-k-vyuce/ITS> Učební text. ČVUT v Praze, Fakulta dopravní.
- [8] BHAT, Chandra R. a KOPPELMAN, Frank S. *Activity-Based Modeling of Travel Demand*. In: Handbook of Transportation Science, 1999. p. 35-61. ISBN 978-1-4615-5203-1
- [9] BANG, C., *Integrated Model to Plan Advanced Public Transportation Systems*, Ph.D. Dissertation, Department of Civil and Environmental Engineering, Virginia Polytechnic Institute and State University, Blacksburg, Virginia. 1998.
- [10] SCHONFELDER, Stefan. a K. W. AXHAUSEN. Urban rhythms and travel behaviour: spatial and temporal phenomena of daily travel. Burlington, Vt.: Ashgate, c2010. Transport and society. ISBN 978-0-7546-9951-4.
- [11] MCNALLY, Michael G. *The Four-Step Model*, in David A. Hensher , Kenneth J. Button (ed.) *Handbook of Transport Modelling* (Handbooks in Transport, Volume 1) , pp.35 - 53, Amsterdam: Elsevier, 2007. ISBN 978-0-08-045376-7
- [12] CLARKE, P. et al. *Migrating Four-Step Models to an Activity Based Modelling Framework in Practice*. In: European Transport Conference, 2008
- [13] WANG, Kelvin C. P. Applications of advanced technology in transportation: proceedings of the ninth international conference, 2006. p. 319-324. ISBN 0784407991.
- [14] CHAPIN, F. Stuart. Human activity patterns in the city: things people do in time and in space. New York: Wiley, 1974. ISBN 9780471145639.

- [15] BHAT, Chandra R a PINJARI, Abdul R. *Activity-based travel demand analysis*. In: A Handbook of Transport Economics, 2011. p. 213-248. ISBN 9781847202031
- [16] BECKX, C. et al. *Assessing Activity-Related Vehicle Emissions through an Integrated Activity-Based Modelling Framework*. In: International Congress on Environmental Modelling and Software, 2008. ISBN 978-1-4020-6009-0
- [17] BECKX, C. et al. *The contribution of activity-based transport models to air quality modelling: A validation of the ALBATROSS-AURORA model chain*. In: Science of The Total Environment, 2009. p. 3814-3822.
- [18] JONES, P.M et al. *Activity analysis: the state of the art and future directions*. In: Developments in Dynamic and Activity-based Approaches to Travel Analysis, 1990. p. 34-55. ISBN 978-0566070235.
- [19] MURAKAMI, Elaine a UMBERG, Cyrus *The Puget Sound Transportation Panel*. In: Panels for Transportation Planning, 1997. p. 159-192. ISBN 978-1-4757-2642-8
- [20] Biler, S., Kouřil, P., Rusý, P., Staněk, M., a Šenk, P. (2014). Metodika aktivitně-cestovního průzkumu. Dostupné online z:
http://www.cyklokonference.cz/cms_soubory/rubriky/369.pdf
- [21] FOSTER, R.E, BENSON, J.D a STOVER, V. *Consequences of small sample 0-D data collection in the transportation planning process*, 1978, Washington : Dept. of Transportation, Federal Highway Administration
- [22] RASOULI, Soora a H. J. P. TIMMERMANS. Mobile technologies for activity-travel data collection and analysis. ISBN 9781466661707.
- [23] Stopher, P R, Kockelman, K, Greaves, S P, Clifford, E. *Reducing burden and sample sizes in multi-day household travel surveys*. Transportation Research Board Annual Meeting, 87th, 2008, Washington, DC, USA.
- [24] ČERTICKÝ, M., DRCHAL, J., CUCHÝ, M. a JAKOB, M. *Models and Technologies for Intelligent Transportation Systems (MT-ITS)*. IEEE International Conference, 2015. p. 229-236.
- [25] DRCHAL, J., ČERTICKÝ, M. a JAKOB, M. *Data Driven Validation Framework for Multi-agent Activity-based Models*. Springer International Publishing, 2015. p. 55-67.
- [26] COTTRILL, C. et al. *Future Mobility Survey: Experience in Developing a Smart-Phone-Based Travel Survey in Singapore*. In: Transportation Research Record: Journal of the Transportation Research Board, issue 2354, 2013. p.59-67.
- [27] PEREIRA, F.C. et al. *Activity recognition for a smartphone and web based travel survey*. In: CoRR, 2015, abs/1502.03634
- [28] ZONG, F. et al. *Identifying Travel Mode with GPS Data Using Support Vector Machines and Genetic Algorithm*. In: Information, 2015. p. 212-217. doi:10.3390/info6020212. ISSN 2078-2489
- [29] Introduction to Support Vector Machines [online]. Dostupné z:
http://docs.opencv.org/2.4/doc/tutorials/ml/introduction_to_svm/introduction_to_svm.html
- [30] FANG, Shih-Hau, Hao-Hsiang LIAO, Yu-Xiang FEI, Kai-Hsiang CHEN, Jen-Wei HUANG, Yu-Ding LU a Yu TSAO. *Transportation Modes Classification Using Sensors on Smartphones*. Sensors [online]. 2016, 16(8), 1324-1340. DOI: 10.3390/s16081324. ISSN 1424-8220. Dostupné z:
<http://www.mdpi.com/1424-8220/16/8/1324>

-
- [31] XIA, Hao, Yanyou QIAO, Jun JIAN a Yuanfei CHANG. Using Smart Phone Sensors to Detect Transportation Modes. *Sensors* [online]. 2014, 14(11), 20843-20865. DOI: 10.3390/s141120843. ISSN 1424-8220. Dostupné z:
<http://www.mdpi.com/1424-8220/14/11/20843/>
 - [32] WOLF, Jean, Randall GUENSLER a William BACHMAN. *Elimination of the Travel Diary: Experiment to Derive Trip Purpose from Global Positioning System Travel Data*. *Transportation Research Record: Journal of the Transportation Research Board*, 2001. p.125-134. DOI: 10.3141/1768-15. ISSN 0361-1981.
 - [33] CARRION, C. et al. *Evaluating fms: A preliminary comparison with a traditional travel survey*. *Transportation Research Board 94th Annual Meeting*, 2014.
 - [34] CATALA, Martin, DOWGLIN, Samuel, HAYWARD, Donald. *Expanding the Google Transit Feed Specification to Support Operations and Planning*, University of South Florida, 2011.

Příloha A

Seznam použitých zkratek

API	■ Application Programming Interface
CSV	■ Comma Separated Values
DPP	■ Dopravní podnik hl. m. Prahy
GPS	■ Global Positioning System
GTFS	■ General Transit Feed Specification
HTTP	■ Hypertext Transfer Protocol
IAD	■ Individuální automobilová doprava
JSON	■ JavaScript Object Notation
MHD	■ Městská hromadná doprava
OS	■ Operační systém
POI	■ Point of Interest
SVM	■ Support Vector Machines