



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA INFORMAČNÍCH TECHNOLOGIÍ

FACULTY OF INFORMATION TECHNOLOGY

ÚSTAV INFORMAČNÍCH SYSTÉMŮ

DEPARTMENT OF INFORMATION SYSTEMS

NÁSTROJ PRO POKROČILOU VIZUALIZACI INTENZITY CYKLISTICKÉ DOPRAVY V BRNĚ

TOOL FOR ADVANCED VISUALISATION OF CYCLING TRAFFIC INTENSITY IN BRNO

BAKALÁŘSKÁ PRÁCE

BACHELOR'S THESIS

AUTOR PRÁCE

AUTHOR

VEDOUCÍ PRÁCE

SUPERVISOR

RADEK ŠEREJCH

Ing. JURAJ LAZÚR,

BRNO 2023

Zadání bakalářské práce



154342

Ústav: Ústav informačních systémů (UIFS)
Student: **Šerejch Radek**
Program: Informační technologie
Název: **Nástroj pro pokročilou vizualizaci intenzity cyklistické dopravy v Brně**
Kategorie: Informační systémy
Akademický rok: 2023/24

Zadání:

1. Seznamte se s problematikou ukládání, sdílení a vizualizace geografických dat.
2. Prostudujte dostupné datové zdroje obsahující data o cyklistické dopravě ve městě Brně a metodu jejich integrace, navrženou ing. Radoslavom Eliášom.
3. Seznamte se s problematikou vizualizací velkých objemů dat v prostoru.
4. Navrhněte nástroj umožňující pokročilé vizualizace dat nad integrovaným modelem intenzit cyklistické dopravy města Brna.
5. Po konzultaci s vedoucím implementujte navržené řešení.
6. Otestujte funkčnost a použitelnost systému ve spolupráci s vybraným vzorkem uživatelů.

Literatura:

- Eliáš, Radoslav. Model of Cycling Traffic Intensity in Brno. Brno, 2023. Master's thesis. Brno University of Technology, Faculty of Information Technology. Supervisor Ing. Jiří Hynek, Ph.D.
- Pluto-Kossakowska, Joanna, et al. "Dashboard as a platform for community engagement in a city development—a review of techniques, tools and methods." *Sustainability* 14.17 (2022): 10809.
- Terh, Shin Huoy, and Kai Cao. "GIS-MCDA based cycling paths planning: a case study in Singapore." *Applied geography* 94 (2018): 107-118.
- Oliveira, F., Nery, D., Costa, D. G., Silva, I., & Lima, L. (2021). A survey of technologies and recent developments for sustainable smart cycling. *Sustainability*, 13(6), 3422.
- Leaflet: Leaflet API reference [online]. 2023 [cit. 2023-09-30]. Dostupné z: <https://leafletjs.com/reference-1.9.4.html>

Při obhajobě semestrální části projektu je požadováno:
Body 1. až 4.

Podrobné závazné pokyny pro vypracování práce viz <https://www.fit.vut.cz/study/theses/>

Vedoucí práce: **Lazúr Juraj, Ing.**
Vedoucí ústavu: Kolář Dušan, doc. Dr. Ing.
Datum zadání: 1.11.2023
Termín pro odevzdání: 9.5.2024
Datum schválení: 30.10.2023

Abstrakt

V důsledku stále rostoucího trendu cyklistické dopravy vyvstává potřeba cyklistickou dopravu analyzovat. Datový portál města Brna poskytuje data, které je možné použít nej-různějšími způsoby. Záznamy v těchto datových sadách používají ovšem různé podkladové mapy. Tento problém řeší Radoslav Eliáš ve své diplomové práci, kde za pomoci algoritmů vytváří spojený datový model. Pro vizualizaci těchto dat používá aplikaci ArcGIS, která však není dostatečně pružná pro snadné vytváření pokročilejších vizualizací. V této práci je tedy navržena a implementována aplikace, která nahrazuje stávající způsob vizualizace. Pro jednoduché předzpracování dat je aplikace rozdělena na backendovou a frontendovou část. Backendová část zajišťuje předzpracování dat, které následně poskytuje skrze API. Frontendovou část tvoří webová aplikace, která zajišťuje veškerou tvorbu grafů a vizualizací. Implementované řešení přináší širší možnosti jakými lze data analyzovat, a také snazší reagování na změny v datových sadách.

Abstract

Due to the continuously growing trend of cycling transportation, there arises a need to analyze cycling transportation. The data portal of the city of Brno provides data that can be utilized in various ways. However, the records in these datasets use different underlying maps. This problem is addressed by Radoslav Eliáš in his master's thesis, where he creates a connected data model using algorithms. For visualizing this data, he uses the ArcGIS application, which, however, is not sufficiently flexible for easy creation of advanced visualizations. Therefore, in this thesis, an application is proposed and implemented to replace the existing method of visualization. For simple data preprocessing, the application is divided into a backend and frontend part. The backend part handles data preprocessing, which is subsequently provided through an API. The frontend part consists of a web application that handles all graph and visualization creation. The implemented solution brings broader possibilities for analyzing data and also facilitates responding to changes in the datasets.

Klíčová slova

cyklistika, geografická data, vizualizace, uživatelské rozhraní, Brno, React, Leaflet

Keywords

cycling, geographic data, visualization, user interface, Brno, React, Leaflet

Citace

ŠEREJCH, Radek. *Nástroj pro pokročilou vizualizaci intenzity cyklistické dopravy v Brně*. Brno, 2023. Bakalářská práce. Vysoké učení technické v Brně, Fakulta informačních technologií. Vedoucí práce Ing. Juraj Lazúr,

Nástroj pro pokročilou vizualizaci intenzity cyklistické dopravy v Brně

Prohlášení

Prohlašuji, že jsem tuto bakalářskou práci vypracoval samostatně pod vedením pana Ing. Juraje Lazúra. Uvedl jsem všechny literární prameny, publikace a další zdroje, ze kterých jsem čerpal.

.....
Radek Šerejch
7. května 2024

Poděkování

Chtěl bych touto cestou poděkovat vedoucímu práce Ing. Juraji Lazúrovi za jeho cenné rady při psaní, a celkově za jeho přátelský a profesionální přístup při vedení práce.

Obsah

1	Úvod	3
2	Geografická data	4
2.1	Souřadný systém	4
2.2	Projekce	4
2.2.1	Azimutální	5
2.2.2	Kuželová	6
2.2.3	Válcová	6
2.3	Prvky interaktivní mapy	7
2.3.1	Bod	7
2.3.2	Čára	8
2.3.3	Oblast	8
2.3.4	Pole	9
2.4	Geospatial data	10
2.5	Ukládání dat	10
2.5.1	NoSQL databáze	11
2.5.2	NewSQL databáze	11
2.5.3	Příklady databází s geografickými daty	11
3	Stávající řešení	13
3.1	Ride with GPS	13
3.2	Město Spokane	13
3.3	DVRPC	14
3.4	Brno	16
3.5	Strava	16
4	Analýza	20
4.1	Analýza uživatelů	20
4.2	Analýza stávajícího řešení	21
4.3	Požadavky na řešení	21
5	Návrh	22
5.1	Architektura	22
5.2	Datové vstupy	22
5.2.1	BikeToWork	23
5.2.2	City Census	23
5.2.3	Cyklodetektory	23
5.2.4	Spojený datový model	23

5.2.5	Cyklistické nehody	24
5.3	Návrh GUI	24
5.3.1	Návrh grafů pro cyklodetektory	26
6	Implementace	27
6.1	Backend	27
6.2	Frontend	28
6.3	Problémy při implementaci	33
7	Testování	34
7.1	Uživatelské testování	34
7.1.1	Testovací scénář	34
7.1.2	Testovací subjekt 1	34
7.1.3	Testovací subjekt 2	35
7.1.4	Testovací subjekt 3	35
7.1.5	Závěr testování	36
7.2	Budoucí vývoj	37
8	Závěr	38
	Literatura	39

Kapitola 1

Úvod

V dnešní době jsou silnice přehlceny automobilovou dopravou a je těžké se ve městech v autě pohybovat i parkovat. Další nevýhodou osobní automobilové dopravy je, že je poměrně drahá, ať už mluvíme o pořizovací ceně nebo o provozu. Navíc řidiči, kterým řízení úplně nejde, mohou mít ve městech problém se složitějšími dopravními situacemi, a tak mohou často způsobit vážné dopravní nehody. Proto lidé stále častěji volí jiné způsoby dopravy.

Cyklistika je alternativou, která se dnes těší stále větší oblibě, jelikož lidé mají stále větší zájem o zdravý životní styl a pohyb. Navíc jízda na kole nevytváří žádné znečištění ovzduší ani žádný hluk. Kolo může být také velmi rychlým dopravním prostředkem, jelikož se na něm můžete lehce vyhnout dopravním zácpám. Také parkování je s kolem daleko snazší kvůli jeho značně menším rozměrům oproti osobnímu automobilu a parkovací stojany jsou dnes po městech hojně rozmístěny.

Některá města např. v Dánsku jsou koncipovaná primárně pro cyklisty, avšak rozvoj měst pro cyklisty je u nás teprve v počátcích. Vytváření takových cest je navíc velmi problematické, jelikož šířka silnice se ve městě ve většině případů zvětšit nedá, a tak se musí cyklisté dělit o prostor s chodci nebo motoristy. Aby města dokázala zlepšovat svoji infrastrukturu, a tím napomocť cyklistům s pohybem po městě, potřebují nějaký nástroj, pomocí kterého by mohla analyzovat provoz cyklistů ve svých ulicích.

Tato práce se zabývá pokročilými vizualizacemi cyklistických dat ve městě Brno, které využívají datový model navržený Radoslavem Eliášem. Teoretická část práce je rozdělena na dvě části. První část, kapitola 2 se zabývá geografickými daty. Nejprve je zde popsán souřadný systém, který se v mapách používá, a mapové projekce. Dále jsou zde popsány prvky, které se používají v interaktivních mapách. Závěr kapitoly se věnuje specifikám geografických dat a jejich ukládáním. Druhá část, kapitola 3 popisuje aplikace, které se věnují zobrazování cyklistického provozu. Praktická část je rozdělena do čtyř částí. První část, kapitola 4 se zabývá analýzou uživatelů a stávajícího řešení, z čehož vyplynou požadavky na aplikaci. V druhé části, kapitole 5 je vytvořen jak návrh architektury celé aplikace, tak návrh grafického uživatelského rozhraní. Dále jsou zde také popsány datové modely, se kterými aplikace pracuje. Třetí část, kapitola 6 obsahuje popis implementace aplikace. Kromě frontendu a backendu jsou zde také popsány problémy, které při implementaci vyvstaly. Čtvrtá část, kapitola 7 obsahuje informace o průběhu testování aplikace a jeho výsledky. Dále jsou zde shrnuty návrhy na budoucí vývoj aplikace.

Kapitola 2

Geografická data

S mapovými aplikacemi je možné se setkat téměř na denní bázi například při cestování. Tyto bývají ve většině případů obohaceny o další prvky, jako například ikony restaurací, benzinových pump nebo zvýraznění aktuální situace v dopravě. Aby tyto prvky mohly být do map zobrazeny, musí být ukotveny do prostoru mapy. K tomu je třeba použít vhodně zvolený souřadný systém, který je specifický svým použitím na kulovém tvaru Země. K tvaru zeměkoule se také vztahují mapové projekce, jelikož je třeba převést nějakým rozumným způsobem její kulový povrch na spojitou 2D plochu monitoru.

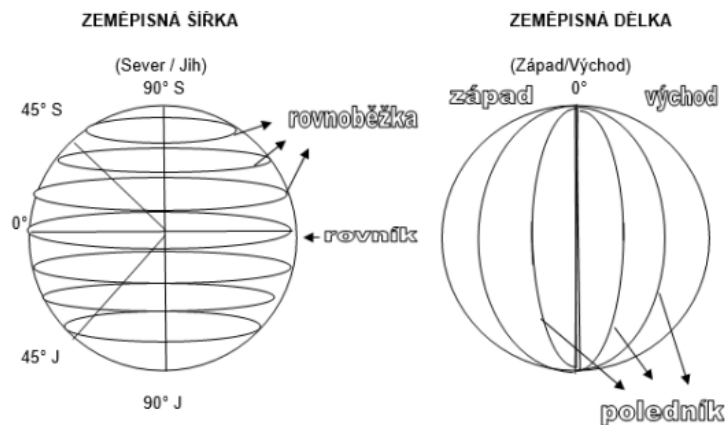
2.1 Souřadný systém

Základem geografických vizualizací je umístování reálných objektů do digitálního prostoru. Umístění každého objektu závisí na jeho zeměpisných souřadnicích. Zeměpisné souřadnice, v angličtině známé jako GCS (geographical coordinate system), používají k abstrakci tvaru Země trojrozměrný kulový povrch. Poloha každého bodu na této kouli je určena zeměpisnou délkou a šířkou. Tyto hodnoty udávají úhel, který je měřený od středu zeměkoule k bodu na jejím povrchu, který se udává nejčastěji ve stupních.[9]

Na kulový povrch je pak možné promítnout dva druhy čar, jedny budou vést mezi póly, a druhé budou vést od východu k západu. Čáry mezi póly jsou tvořeny body, které mají stejnou zeměpisnou délku a nazývají se poledníky. Čáry od východu k západu jsou tvořeny body, které mají stejnou zeměpisnou šířku a nazývají se rovnoběžky. Rovnoběžky a poledníky je možné vidět na obrázku 2.1. Zeměpisná šířka se dále specifikuje jako severní nebo jižní v rozsahu od 0° do 90° a zeměpisná délka jako východní nebo západní v rozsahu od 0° do 180° . Spojení rovnoběžek a poledníků pak vytváří mříž, která se nazývá zeměpisná síť. Tyto souřadnice se pak používají pro určení polohy na ploché mapě. [16]

2.2 Projekce

Projekce je důležitou součástí mapových aplikací, jelikož je třeba převést zemi z trojrozměrného objektu na dvojrozměrný. Převádění se provádí kvůli lepší čitelnosti mapy a lepší práci s ní. Projekci mapy je možné si zjednodušeně představit jako proces, kde je nejprve okolo Země obtočen nějaký papír či 2d plocha, na kterou se bude promítat. Poté jsou ze středu Země vedeny čáry, které protnou povrch Země a plochu, na kterou promítáme. Bod, který tato čára protne na povrchu Země, se pak promítne na plochu do místa, kde se protla s čarou [9]. Vzniklá projekce pak také používá klasický souřadný systém, kde se poloha bodu

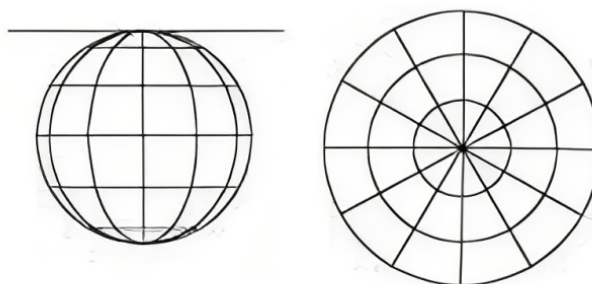


Obrázek 2.1: Rovnoběžky a poledníky ¹

určuje podle hodnoty x a y. Způsobů, jakým je možné přiložit plochu k Zemi a jakého tvaru tato plocha je, je však několik, a tak vzniká více typů projekcí, kde pro každou projekci vzniká jiné zobrazení, které se pak musí kompenzovat vizualizacemi. Každá tato projekce má své výhody a nevýhody a hodí se pro různá použití. [17]

2.2.1 Azimutální

Azimutální projekce promítá body na rovnou plochu, která se protíná se Zemí v jediném bodě. Tento bod pak určuje, na které místo se bude daná aplikace projekce zaměřovat, jelikož okolo tohoto bodu bude mít nejmenší zkreslení. Variace této projekce se mohou lišit bodem, ze kterého budeme vysílat čáry, které se následně spojí s bodem na povrchu Země, a který se následně promítne na plochu. Tento bod je možné zvolit buď uprostřed Země, nebo jej posouvat po přímce, která je kolmá k ploše, na kterou promítáme. Nejčastěji se pak volí body buď na opačné straně zeměkoule, nebo v nekonečné vzdálenosti, což simuluje situaci, kdy se body budou promítat kolmo na plochu. Tato projekce se nejčastěji používá k zobrazení polárních oblastí[9]. Ukázkou tohoto zobrazení je možné vidět na obrázku 2.2



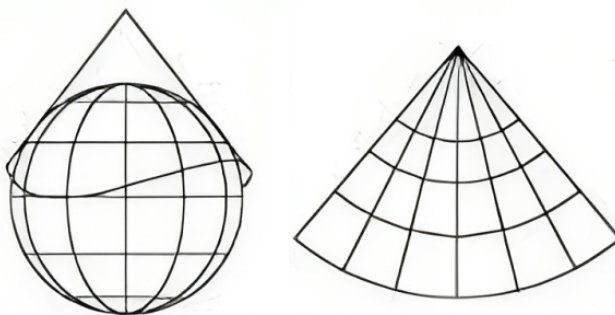
Obrázek 2.2: Ukázkou azimutální projekce a její výsledek²

¹zdroj: https://inkluzivniskola.cz/sites/default/files/uploaded/z_zemepisne_souradnice.pdf

²zdroj: https://slideplayer.cz/slide/15246244/92/images/6/KARTOGRAFICK%C3%81_ZOBRAZEN%C3%8D.jpg

2.2.2 Kuželová

Tato projekce používá k promítnutí kuželovou plochu, která se přiloží k Zemi tak, že se s ní protíná v jedné kružnici. Tato kružnice se nazývá standardní rovnoběžka. U této projekce obecně platí, že čím více se kuželová plocha vzdaluje od Země, tím je výsledek zkreslenější. Z toho plyne, že nejpřesnější zobrazení dostaneme právě okolo místa protnutí. Kuželovou projekci je také možné modifikovat tak, že se bude Země protínat místo s jednou kružnicí se dvěma. Tím je docíleno toho, že výsledná projekce bude celkově méně zkreslená. Dále je také možné kužel okolo Země libovolně posouvat a zmenšovat ho či zvětšovat, čímž je možné zacílit na různé oblasti[9]. Ukázku této projekce je možné vidět na obrázku 2.3. Azimutální a kuželové projekce a jejich variace používají aplikace jako např. Generic Mapping Tools, nebo PROJ.4. V těchto aplikacích lze vytvářet statické mapy, s použitím různých projekcí. Tyto mapy se pak dají využívat pro atlasy, které se zaměřují na určitou oblast, popř. pokud chce mapa nějakým způsobem demonstrovat tvar Zeměkoule. [10]

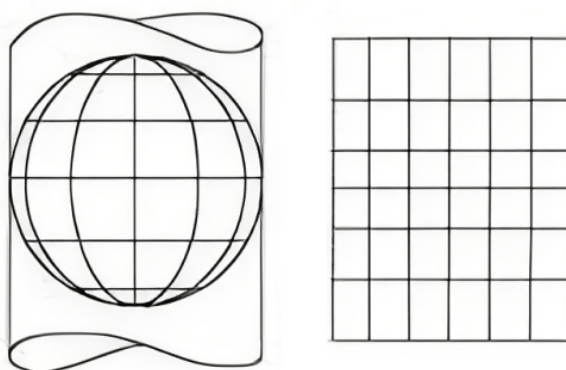


Obrázek 2.3: Ukázka kuželové projekce a její výsledek ³

2.2.3 Válcová

Válcová projekce používá k promítnutí bodů kuželovou plochu, která se přiloží k Zemi tak, aby se s ní protínala v rovníku. Rovnoběžníky jsou pak vykresleny matematicky a poledníky geometricky na plášť válce. Ve výsledku dostaneme mříž, kde rovnoběžníky a poledníky spolu svírají pravý úhel. Promítnuté poledníky mají mezi sebou stále stejné mezery, zatímco mezery mezi rovnoběžníky se zvětšují směrem k pólům. Tato projekce má nejmenší zkreslení okolo kružnice, ve které se protíná se Zemí. Tuto kružnici, a tím i oblast, kde bude projekce nejpřesnější, je možné měnit tím, že budeme válec okolo Země různě natáčet[9]. Tuto projekci nejčastěji používají mapy, ve kterých lze přibližovat z globálního do lokálního měřítko. [10]. Variaci válcového zobrazení, tzv. Mercator projekci, používá mnoho velkých poskytovatelů online map, jako jsou např. Google Maps, nebo OpenStreetMap. Tato variace má kružnici průniku se Zemí na rovníku nebo dvě v jeho blízkém okolí, a tak je v těchto místech nejpřesnější. Některé polární oblasti se touto projekcí nemusí vůbec zobrazit, jelikož nepromítá celý rozsah zeměpisné šířky [19]. Ukázku této projekce je možné vidět na obrázku 2.4

³zdroj: https://slideplayer.cz/slide/15246244/92/images/6/KARTOGRAFICK%C3%81_ZOBRAZEN%C3%8D.jpg



Obrázek 2.4: Ukázka válcové projekce a její výsledek ⁴

2.3 Prvky interaktivní mapy

Pro zobrazení dodatečných dat, se kterými je možné nějak dále pracovat, je třeba použít místo klasické mapy mapu interaktivní. Ta se odlišuje tím, že je potřeba do klasické mapy přidat prvky, se kterými lze nějak interagovat, a které nám budou zobrazovat nějaká data. Tyto prvky se dělí podle zobrazovaného obrazce a jsou jimi bod, čára, oblast a pole. Pro zobrazení těchto prvků jsou používány dva přístupy: imitace a schematizace. Imitace přiděluje prvkům souřadnice, a tím je umísťuje do zobrazované plochy, kde se pak zobrazují. Schematizace zjednodušuje nebo upravuje geografický kontext (neboli mapu v pozadí), aby zvýraznila informace, které chce zobrazit[7]. Toto rozdělení lze uplatnit na všechny základní prvky interaktivní mapy, a proto je použito v následujících úsecích. Ukázku jednotlivých prvků a jejich technik je možné vidět na obrázku 2.5.

2.3.1 Bod

Body se používají k označení míst, na která se daná aplikace zaměřuje. Bod může například zobrazovat místa dopravních nehod.

Imitace

Imitace bodů používá často různé ikonky, popřípadě v kombinaci s nějakým štítkem pro zobrazení informací do mapy. Výběr ikon je zde klíčový, protože je třeba zvolit takové, na které jsou uživatelé zvyklí, a jejichž význam je intuitivní. Výsledná mapa je pak čitelná do té míry, jak je uživatel schopný na první pohled rozeznat význam ikon. Při vizualizaci takovýchto prvků se často používají různé úrovně důležitosti, jako je vidět na obrázku 2.5. Tato vlastnost se často zobrazuje pomocí různých velikostí fontů písma nebo ikon[3]. To je většinou kombinováno s funkcí, kdy jsou pro aktuální míru přiblížení zobrazovány pouze určité úrovně důležitosti.

⁴zdroj: <https://slideplayer.cz/slide/15246244/92/images/6/KARTOGRAFICK%C3%81+ZOBRAZEN%C3%8D.jpg>

Schematizace

Schematizační techniky se místo odstraňování prvků kompletně zaměřují na jejich přesouvání. Jedna metoda se zabývá problémem, který nastává v místech s hustějším zaplněním, kde by se dvě ikony vedle sebe nevešly. Tento problém se řeší pomocí algoritmu, který přesouvá body z míst s hustším zaplněním do míst, kde bodů není tolik. Druhá metoda se zabývá uzly grafů. Tato metoda přemísťuje uzly například v místech, kde se kříží hrany grafu tak, aby byla splněna vizuální kritéria výsledného grafu, a aby se zvýšila jeho čitelnost [1].

2.3.2 Čára

Čáry se nejčastěji používají k vizualizaci cest, silnic nebo k vyznačení obrysů nějaké oblasti.

Imitace

Při imitaci čáry se používají dvě techniky - zaměřené na obrysy a na hrany. Techniky zaměřené na obrysy používají k vytvoření čar body, které mohou získat například z ostrých rohů kolem obrysu, nebo dlouhých úseků na obrysu, které neobsahují ohnutí. Algoritmy se zde musejí potýkat s tím, aby se vyhly pravidelným obrysům. Toho mohou docílit tím, že do výpočtu čar přidají nějaký náhodný prvek, a tím zkreslí pravidelný obrys. Dalším způsobem je na výsledné čary použít nějakou geometrickou transformaci v případě, že detekují pravidelné obrysy. Techniky zaměřené na hrany se používají k vytvoření trasy mezi dvěma body. Hlavní problém, se kterým se tyto algoritmy potýkají je, aby vzniklé trasy nezanesly do mapy mnoho čar, a tak ji udělaly nečitelnou. Jedna cesta, jak tento problém eliminovat, je spojovat čáry, které jdou podobnou cestou, a tím tak eliminovat počet čar potřebných k vykreslení v hustě pokrytých oblastech [22].

Schematizace

Schematizace čáry používá dvě techniky - zaměřené na okraje a na cesty. Techniky zaměřené na okraje se zaměřují na zjednodušení obrysu nějakého regionu. Hlavní úlohou těchto algoritmů je nalézt na obrysu body, které lze vynechat, nebo úseky, které lze modifikovat. Poté se aplikuje transformace, která zajistí rozumnou míru schematizace. Techniky zaměřené na cesty se zabývají zjednodušováním komplexnějších tras. Jedním z nejpoužívanějších přístupů je narovnání cest a redukování počtu možných úhlů ohybů mezi cestami (např. kroky po 45 stupních), jehož ukázkou je vidět na obrázku 2.5.

2.3.3 Oblast

Oblasti zobrazují do různě velkých a tvarovaných obrazců nějaký atribut, jako např. počet obyvatel v dané oblasti.

Imitace

Imitace oblasti používá k zobrazení různých oblastí tři techniky. První technika je zaměřená na mřížky a vytváří pravidelnou mříž obrazců. Nejčastěji se pro vytváření mříže používá šestiúhelník, jelikož jeho tvar vypadá nejméně pravidelně [14]. Tato technika rozdělí mapovou plochu právě do této mříže a každé pole pak nese nějakou informaci. Druhá technika je geometrická teselace a vytváří nepravidelnou mříž obrazců. V této technice je každý bod,

který reprezentuje nějaká data, uzavřen do buňky, která ho obklopuje. Tyto buňky pak mohou být podle různých kritérií slučovány do větších buněk. Buňky jsou generovány tak, že pozice, která nepatří žádnému bodu, je přiřazena k tomu nejbližšímu. Při práci s regiony, kde je zobrazována vodní plocha, se přidávají virtuální body do vody, aby se tyto plochy obkreslily. Třetí technika generuje plochy pouze kolem shluků bodů místo toho, aby zaplňovala celou plochu. Tato technika často zobrazuje úroveň hustoty v zobrazovaných datech.

Schematizace

Schematizace používá k zobrazení oblastí tři druhy technik. První druh používá k zobrazení oblastí nějakou oblast z podkladové plochy, kterou dále různými způsoby deformuje, aby zobrazil dodatečná data. Deformace musí proběhnout tak, aby byla výsledná oblast stále podobná té zdrojové. Pro tyto algoritmy je klíčové, aby dobře adaptovaly oblasti z mapy. Asi nejjednodušším řešením deformace je zvětšování a zmenšování zdrojové oblasti. Náročnější řešení se zaměřují na udržení topologických vztahů mezi oblastmi po deformaci. Další technika používá k zobrazení geometrické objekty a nevychází tak z tvaru podkladové oblasti. U této techniky musí algoritmy řešit vhodné místo pro zobrazení obrazců, aby byla zachována alespoň částečná topologická přesnost. Tato technika dosahuje vysoké statistické přesnosti na úkor topologické a geografické přesnosti. Třetí technika zobrazuje data do vygenerované mřížky geometrických objektů. Tato technika je schopna dosáhnout vysoké statistické přesnosti, jelikož zobrazovaná data se namapují na nějakou vizuální vlastnost buňky (např. podbarvení), a přesto si zachovat velkou geografickou a topologickou přesnost, jelikož buňky kopírují obrysy v podkladu.

2.3.4 Pole

Pole se používají k zobrazení nějakého kontinuálního fenoménu, jako je například riziko zemětřesení.

Imitace

Imitace využívá pro vytváření polí dvě techniky - obarvovací a konturovací. Obarvovací používá k zobrazení rozložení zobrazované hodnoty barevnou škálu, kterou je pokryta mapa. Konturovací technika zobrazuje oblasti se stejnými hodnotami pomocí spojnic. Tyto spojnice zvýrazňují přechod mezi hodnotami. Čím jsou spojnice blíže k sobě, tím je rozdíl mezi hodnotami větší.

Schematizace

Schematizace zobrazuje do pole kontinuální data, která je potřeba pro použití zdiskretizovat do pravidelné mřížky. K zobrazení dále používá dvě techniky. První technika používá vizuální roztahování oblastí na mapě. Ve většině případů algoritmus nejprve položí přes mapu pravidelnou mříž, kde buď buňka nebo uzel představuje nějakou hodnotu. Poté se roztáhnou buňky nebo uzly podle velikosti jejich hodnot. Zatímco geografická a statistická přesnost této techniky je průměrná, její topologická přesnost je vysoká, jelikož hranice sousedících oblastí zůstávají vždy stejně napojené, jako je vidět na obrázku 2.5. Druhá technika používá pro zobrazení dat vizuální hustotu prvků. Pro tento účel jsou nejprve kontinuální

data zdiskretizována a následně uložena do mříže. Hodnota uzlů je pak vizuálně odlišena pomocí velikosti, vzdálenosti nebo sytosti barvy.

Perspektiva	Imitace				Schematizace			
	Bod	Čára	Oblast	Pole	Bod	Čára	Oblast	Pole
Kategorie								
Technika	úrovně důležitosti 	obrysy 	mřížky 	obarvení 	přemístění bodů 	okraje 	deformace 	roztahování
	ikony 	hrany 	teselace 	konturování 	přemístění uzlů 	cesty 	objekty 	hustota
			shluky 				mříž 	

Obrázek 2.5: ukázka prvků a jejich technik⁵

2.4 Geospatial data

Geografická data jsou specifická tím, že každému objektu musí specifikovat souřadnice a k nim jsou přidány nějaké další informace. Při ukládání do databáze mají geospatial data tři části: data o lokaci, data o atributu a o čase. Data o lokaci popisují umístění popisovaného objektu, a podle toho se může do mapy vykreslovat různými tvary. Data o atributu popisují určitou hodnotu, kterou chceme sledovat a která se vztahuje k danému místu. Tato hodnota se pak většinou různými způsoby škáluje a určuje způsob vizualizace (např. podbarvení). Data o čase nám udávají čas, ve kterém byly data o atributu zaznamenána. Tato data jsou dnes velmi užitečná, jelikož většinou sledujeme nějaký fenomén v dlouhodobějším horizontu, raději než v malém časovém výseku [11]. Příkladem uložení takto strukturovaných dat může být nějaký panelový dům, jehož poloha je určena souřadnicemi. Data o atributu mohou být například kolik lidí v tomto domě bydlí, které jsou spojené s časovým údajem, jelikož počet obyvatelů domu se může měnit. Mezi nejpoužívanější datové formáty patří Shapefile, KML nebo GeoJSON [13].

2.5 Ukládání dat

K ukládání dat mnoho dekad sloužila relační databáze. V důsledku stále většího objemu dat ale začala narážet na svoje limity a začaly stále častěji vyvstávat potřeby pro efektivnější ukládání velkých objemů dat. Hlavní problém s SQL databází je ten, že pro zajištění principů ACID a pro zajištění bezpečnosti transakcí potřebuje mnoho výpočetního času. V dnešní době je také daleko větší potřeba ukládat nestrukturovaná data, k čemuž relační databáze není úplně vhodná [15]. Dnes se tak používají technologie, které jsou schopny obsloužit velké množství nestrukturovaných dat, ovšem za cenu toho, že nejsou vždy dodrženy principy ACID.

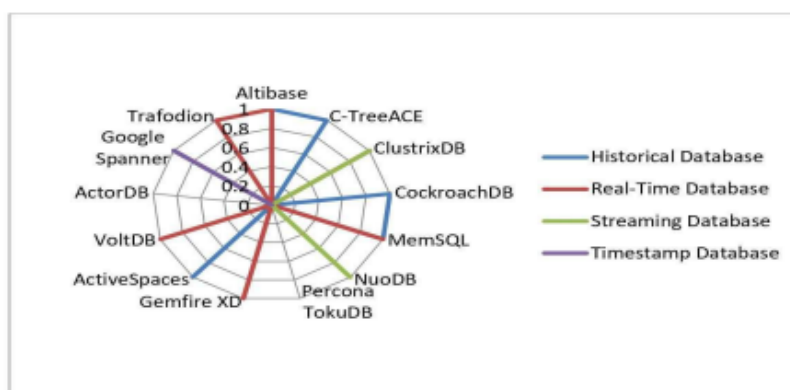
⁵ zdroj: [7]

2.5.1 NoSQL databáze

Označení NoSQL se používá pro databáze, které používají nízkoúrovňové nestandardizované rozhraní. Tato vlastnost je dělá náročnější pro integraci do stávajících aplikací, očekávajících SQL rozhraní, či pro změnu poskytovatele [20]. Příklady NoSQL Databáze jsou např. HBase, Cassandra, DynamoDB, MongoDB, Riak, Redis, Accumulo, Couchbase [6].

2.5.2 NewSQL databáze

NewSQL databáze jsou moderní formou relační databáze. Tyto databáze cílí na podobnou škálovatelnost jako NoSQL databáze, zatímco si zachovávají transakční záruky tradičního databázového systému. Charakteristikou těchto databází je zachování principů ACID a použití SQL jako hlavního mechanismu pro interakci[2]. Jejich ukázkou můžeme vidět na obrázku 2.6



Obrázek 2.6: Ukáзка databází a jejich typů ⁶

2.5.3 Příklady databází s geografickými daty

- **PostgreSQL**

PostgreSQL je open source relační databáze, která využívá modelu klient-server. PostgreSQL může používat replikaci dat na různých serverech, aby zajistil vysokou dostupnost, balancování zátěže a rychlé provádění dotazů[8]. Do PostgreSQL mohou být také přidávána různá rozšíření, která nám získají další funkcionalitu či datové typy. Jedním z takových rozšíření je PostGIS, které se zaměřuje na geografická data. Součástí tohoto rozšíření jsou další datové typy, jako například body či mnohoúhelníky. Toto rozšíření používá zvláštní indexy pro rychlý přístup k prostorovým datům, které jsou založené na jejich lokaci. Dále jsou zde funkce, jako např. měření vzdálenosti či plochy oblastí. Data zde lze také generalizovat nebo zjednodušovat.

- **MongoDB**

MongoDB je NoSQL databázový systém, který nabízí pokročilou podporu pro geografické indexy a dotazování na geografická data. Tyto indexy umožňují efektivní ukládání a dotazování na geografická data, jako jsou body, oblasti a cesty. Existují dva hlavní typy geografických indexů, které MongoDB podporuje. 2d index je vhodný pro

⁶zdroj: [2]

jednoduché geografické úlohy, jako jsou vyhledávání bodů v určité oblasti. 2dsphere index je určen pro složitější geografické operace, jako jsou vyhledávání bodů v blízkosti určitého místa, hledání oblastí zahrnujících daný bod apod.

- **MySql**

MySQL je open-source relační databázový systém vhodný pro širokou škálu aplikací. MySQL nabízí rozšíření nazvané MySQL Spatial, které umožňuje práci s geografickými daty. Toto rozšíření poskytuje podporu pro ukládání, indexaci a dotazování na geografické objekty, jako jsou body, linie a polygony. MySQL Spatial poskytuje vytváření geografických indexů či širokou škálu funkcí a operátorů pro práci s geografickými daty. Díky podpoře standardních formátů a funkcí MySQL Spatial je možné integrovat MySQL databázi s dalšími nástroji a aplikacemi pro geografickou analýzu a vizualizaci, jako jsou například QGIS nebo ArcGIS.

Kapitola 3

Stávající řešení

Analýzou cyklistických dat se dnes, v důsledku stoupajícího trendu cyklistické dopravy[21], zabývá mnoho institucí, a tak je možné na internetu najít stále více aplikací, které se touto problematikou zabývají. Některé jsou i volně přístupné, avšak ve většině případů takové aplikace neposkytují velkou informační hodnotu. Aplikace, které nejsou volně dostupné, jako např. strava metro nebo bike citizens analytics, nám poskytnou mnohem víc užitečných nástrojů, avšak je těžké k nim získat přístup. Tyto aplikace mohou dnes daleko lépe fungovat, jelikož sbírají data ze sportovních mobilních aplikací, které se těší čím dál tím větší oblibě. Tomu pomáhá, že dnes jsou chytré mobilní telefony velmi rozšířené, a chytré sportovní hodinky v kombinaci s aplikací, kde si lze zaznamenávat svoje sportovní výkony, používá stále více lidí[12]. V následujících kapitolách je popsáno několik vybraných aplikací.

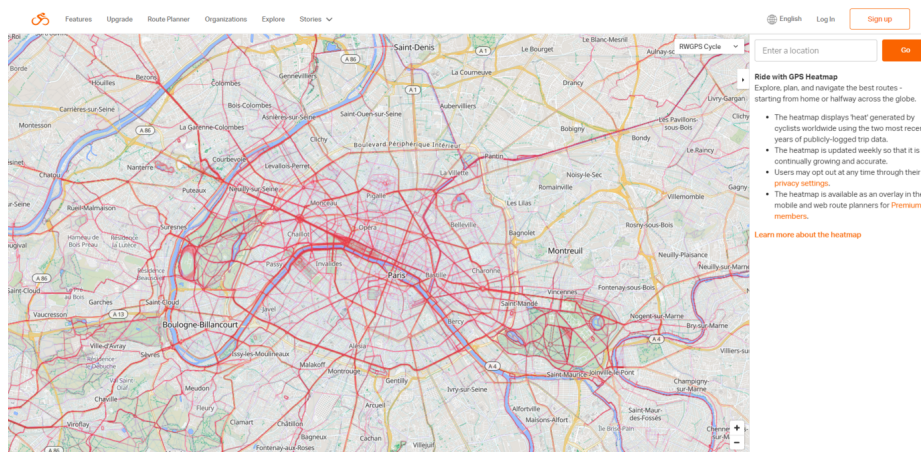
3.1 Ride with GPS

Tato aplikace zobrazuje pomocí heatmapy cyklistická data z aplikace Ride with GPS: Bike navigation. Výhodou této aplikace je, že ji používá velké množství uživatelů po celém světě, a tak si zde můžeme prohlédnout data z různých koutů světa. Tato aplikace ovšem postrádá jakékoliv další možnosti s daty dále pracovat, filtrovat je, nebo vytvářet nějaké statistiky, což je velká nevýhoda pro celkovou využitelnost aplikace. Jediná interaktivní akce, co lze s aplikací provádět, je změnit podkladovou mapu. Ukázkou aplikace je možné vidět na obrázku 3.1.

3.2 Město Spokane

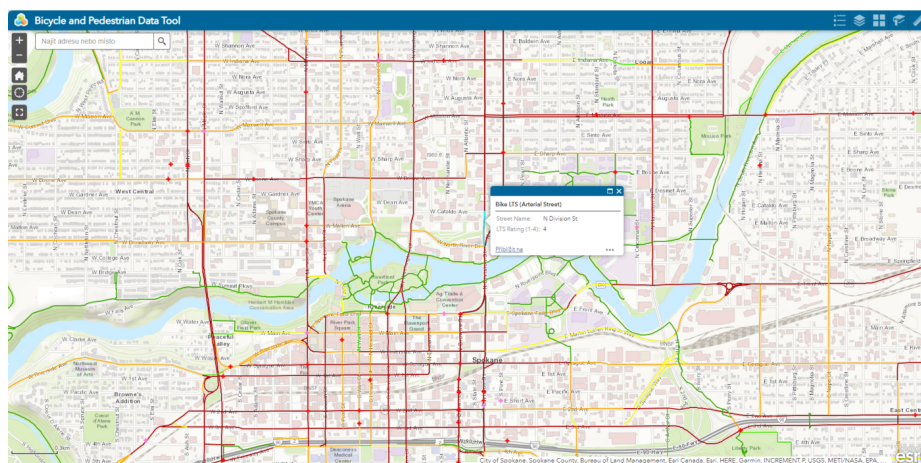
Aplikace, kterou provozuje město Spokane v USA, zobrazuje hodnoty bicycle LTS pomocí nástroje ArcGis do heatmapy. LTS je zkratka pro level of traffic stress, která se zde zaměřuje na cyklisty. Tato hodnota se přiřazuje cestám nebo ulicím a ukazuje, jak je cesta vhodná pro užívání cyklisty z hlediska bezpečnosti a pohodlí jízdy. Z toho pak můžeme vyhodnotit, jak zkušený cyklisté by si měli na takové cesty troufnout. Bike LTS rozděluje cesty do čtyř kategorií ohodnocených od 1 do 4 (1 jsou cesty s nejmenším stresem, 4 s největším). Do rozhodování, jak cesty do těchto kategorií rozřadit, může vstupovat mnoho faktorů, jako je třeba počet jízdních pruhů na dané cestě, počet cyklopruhů, maximální povolená rychlost a tak dále.

Hodnota LTS se zde konkrétně počítá podle toho, zda silnice disponuje jízdním pruhem, popř. zda je vybaven parkováním na ulici, či jakou má pruh šířku. Dalšími faktory jsou



Obrázek 3.1: Ukázka aplikace Ride with GPS

maximální povolená rychlost nebo počet průběžných pruhů a další méně významné faktory [18]. Cesty, které jsou fyzicky odděleny od ostatní dopravy, jsou automaticky považovány za cesty úrovně 1. V heatmapě si lze rozkliknout jednotlivé části ulic pro zobrazení detailu, kde se můžeme dočíst, jaký je název zvolené ulice a hodnotu LTS pro cyklistickou dopravu. Tato aplikace je zvláštní tím, že umožňuje do mapy promítnout další vrstvy, díky kterým můžeme do mapy promítnout další užitečné informace, jako jsou například místa nehod cyklistů nebo kde se nachází semaforey a tak dále... a tím tak zvýšit přehled o dopravní situaci cyklistů ve městě. Těto aplikaci ale chybí nějaká vrstva, která by ukazovala reálný provoz cyklistů ve městě. Ukázku aplikace je možné vidět na obrázku 3.2.



Obrázek 3.2: Ukázka aplikace města Spokane

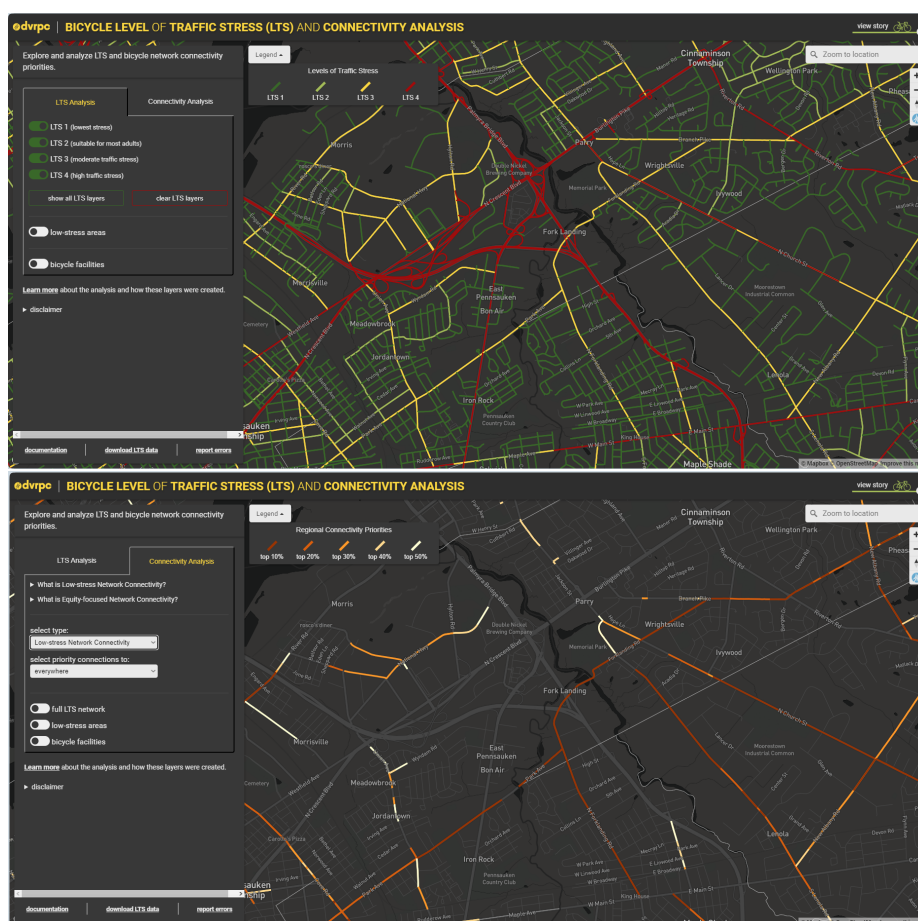
3.3 DVRPC

Aplikace DVRPC¹, jako jedna z dalších, zobrazuje do mapy ulic LTS pro cyklistickou dopravu města Philadelphia, přičemž si lze jednotlivé úrovně skrýt nebo zobrazit. Její ukázku

¹<https://www.dvrpc.org/webmaps/bike-lts/analysis/>

je možné vidět na obrázku 3.3. Také si zde lze zobrazit cyklistické stezky nebo pruhy pro cyklisty. Zvláštností této aplikace je, že si zde lze zobrazit analýzu Low-stress Network Connectivity, která nám ukazuje, jaké ulice LTS úrovně 3 jsou nejvíce používané. Tato funkčnost je zde implementována proto, že město chtělo zlepšit provoz na silnicích s nižší zátěží, a tak ulevit přetíženým hlavním proudům. To by mělo za následek snadnější cestování cyklistů po městě. Jelikož ale nelze pracovat na všech cestách zároveň, je potřeba určit, které cesty jsou nejvíce používané, a tudíž které je potřeba vylepšovat prioritně. Proto je zde implementován algoritmus, který si vytvoří nejkratší cestu z jedné cesty LTS 3. úrovně a spojí ji s každou další cestou LTS 3. úrovně do vzdálenosti 5 mil za pomoci shortest path algoritmu. Při spojování cest jsou používány pouze cesty LTS 1., 2. a 3. úrovně. Procenta zobrazená do mapy jsou nazvána relativním potenciálem dopadu, což ve zkratce říká jak moc je cesta využívána nějkraťšími cestami, které jsme vygenerovali[4]. Z toho můžeme právě vyčíst, která cesta je vhodná pro zlepšení.

U této analýzy si lze zobrazit i typ equality-focused network connectivity, který se liší od předchozího tím, že zahrnuje informace o populaci, která žije v počáteční a cílové oblasti. Cílem tohoto typu zobrazení je nalézt cesty, které by měly při vylepšení největší potenciál pomoci znevýhodněným skupinám a udělat tak pro ně cestování na kole jednodušší. Nevýhodou této aplikace je, že neposkytuje žádné informace o reálném provozu cyklistů.



Obrázek 3.3: Ukázka aplikace DVRPC

3.4 Brno

Aplikace města Brna zobrazuje do mapy cyklistická data zpracovaná R. Eliášem popsaná v jeho diplomové práci[5].

Data sbírá z různých aplikací, které ovšem používají jiné datové modely, a proto vyvztala potřeba vytvořit datový model, který bude všechny spojovat dohromady. Při spojování datových modelů však nastal problém s inkonzistencí dat, jelikož každá aplikace používá jiné mapy, a tím každá ulice může mít jiné ukončení nebo být o kousek posunuta. Proto bylo potřeba vytvořit algoritmus, který bude ulice, které se v cílové mapě neshodují, spojovat do jedné.

K tomu používá tři přístupy, pomocí kterých porovnává ulice mezi sebou. Prvním přístupem je offset tolerance. Tento přístup využívá zaokrouhlování souřadnic bodů, které vymezují ulice, a tím spojuje nejbližší ulice dohromady. Druhým přístupem je maximal accepted angle. Tento přístup porovnává úhel, který mezi sebou svírají dvě ulice a diskvalifikuje pro spojení ty, které přesáhnou určitou hodnotu. Většina ulic však není tvořena jednoduchými přímkami, a je proto třeba takovou ulici uzavřít do pomocného obdélníku a přímku vytvořit z jeho úhlopříčky. Třetí přístup je best match score. Tento přístup nám konečně vytvoří jakési ohodnocení, jak jsou si dvě ulice podobné. Toho docílí tak, že kolem dvou porovnávaných ulic vytvoří pomocný obdélník. Podobnost ulic získá z procenta překrytí těchto obdélníků. Pro každou ulici pak získá její nejlepší shodu (popř. že se s žádnou ulicí v cílové mapě neshoduje) a propojí tyto dvě ulice. Vyhledávání mezi všemi ulicemi města Brna by však bylo výpočetně velmi náročné, a tak zde byl algoritmus optimalizován pomocí rozdělení ulic do segmentů. Shody pro jednotlivé ulice se následně nemusejí hledat ve všech ostatních ulicích, ale pouze v těch, které jsou ve stejném segmentu s ulicí, pro kterou hledáme shodu.

Po spojení ulic algoritmem můžeme vytvořit tabulku, která bude spojovat všechny datové modely. Tabulka bude obsahovat pouze informaci, které ulice s kterými souvisí. Data z jednotlivých aplikací si zachovají datové modely jednotlivých aplikací. Výsledný datový model pak zobrazuje pomocí aplikace GIS do heatmapy.

Po rozkliknutí jednotlivých úseků v této heatmapě si můžeme prohlédnout informace získané z různých datových zdrojů a seskupené podle roků. Kromě úseků zde také najdeme automatické cyklodetektory, které nám poskytují informace o počtu chodců a cyklistů v jednotlivých směrech. Dále zde můžeme také vidět počet cyklistických nehod. Všechny tyto vrstvy lze vypínat i zapínat. Ukázkou aplikace je možné vidět na obrázku 3.4.

3.5 Strava

Heatmap

Zdarma dostupná verze aplikace Strava s názvem heatmap, ukázaná na obrázku 3.5, zobrazuje cyklistická data z aplikace Strava: Run, Bike, Hike. Výhodou této aplikace je, že ji používají lidé po celém světě, a tak si můžeme pomocí dat získaných z této aplikace udělat celkem reálný obrázek o tom, jak cyklistická doprava vypadá. Data jsou agregována za poslední rok a jsou aktualizována jednou měsíčně. Tato heatmapa ale kromě vybírání typu pohybu, který zobrazuje (např. jízda na kole nebo chození), neposkytuje žádnou další práci s daty.



Obrázek 3.4: Ukázka aplikace města Brna

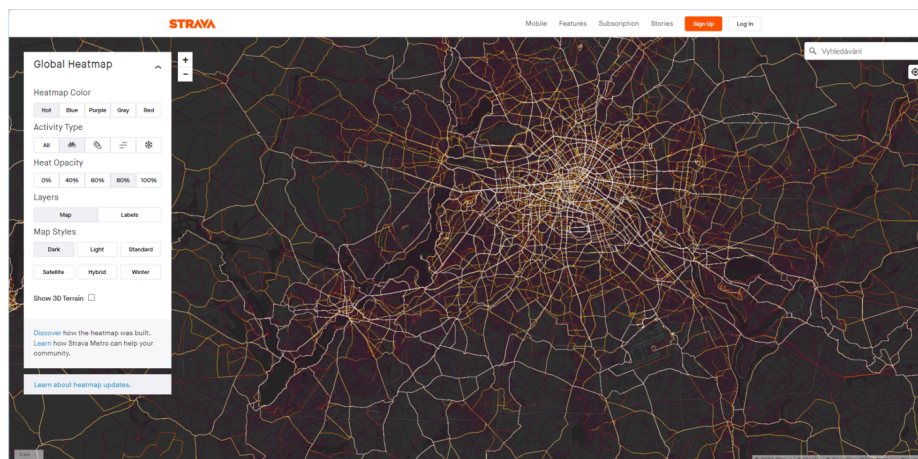
Metro

Asi nejpropracovanější aplikací je Strava metro, jejíž ukázky je možné vidět na obrázku 3.6. Data používá stejné jako Strava heatmap, ale možnost dále s daty pracovat je zde daleko větší. V aplikaci si můžeme vybrat mezi pěti pohledy: Heatmap, Travel Times, Origins & Destination, Routes, Streets.

Pohled Heatmap nám ukazuje podobnou heatmapu jako Strava heatmap, s tím rozdílem, že rozlišení je mnohem vyšší, a tak je mapa daleko čitelnější.

Po zvolení pohledu Travel Times můžeme kliknout kamkoliv do mapy a po chvíli se vygeneruje mapa šestiúhelníků. Velikost plochy, na které se šestiúhelníky vygenerují, je vypočítána podle zvolené maximální doby cestování. Po najetí myši na jiný šestiúhelník na mapě se nám zobrazí odhadovaná délka jízdy. Data pro výpočet lze upravit podle období, ve kterém byly cesty pro výpočet uskutečněny pomocí výběrových polí. Čas jízdy ze zdrojového pole můžeme také zhruba vyčíst z podbarvení ostatních polí v mapě.

Pohled Origins & Destination nám vytvoří mapu šestiúhelníků, kde pro každé pole můžeme vidět, kolik cest v něm začínalo nebo kolik končilo. Data v těchto polích můžeme upravovat podle účelu cesty (volnočasová nebo dojíždění), části dne, dne v týdnu nebo podle roku. Po vybrání pole se nám zobrazí všechny cílové pole, do kterých vedly cesty z vybraného pole, a po zvolení cílového pole můžeme vidět statistiku všech jízd, které vedly ze zvoleného pole do cílového. Můžeme zde vyčíst průměrný čas a vzdálenost jízdy, zda se

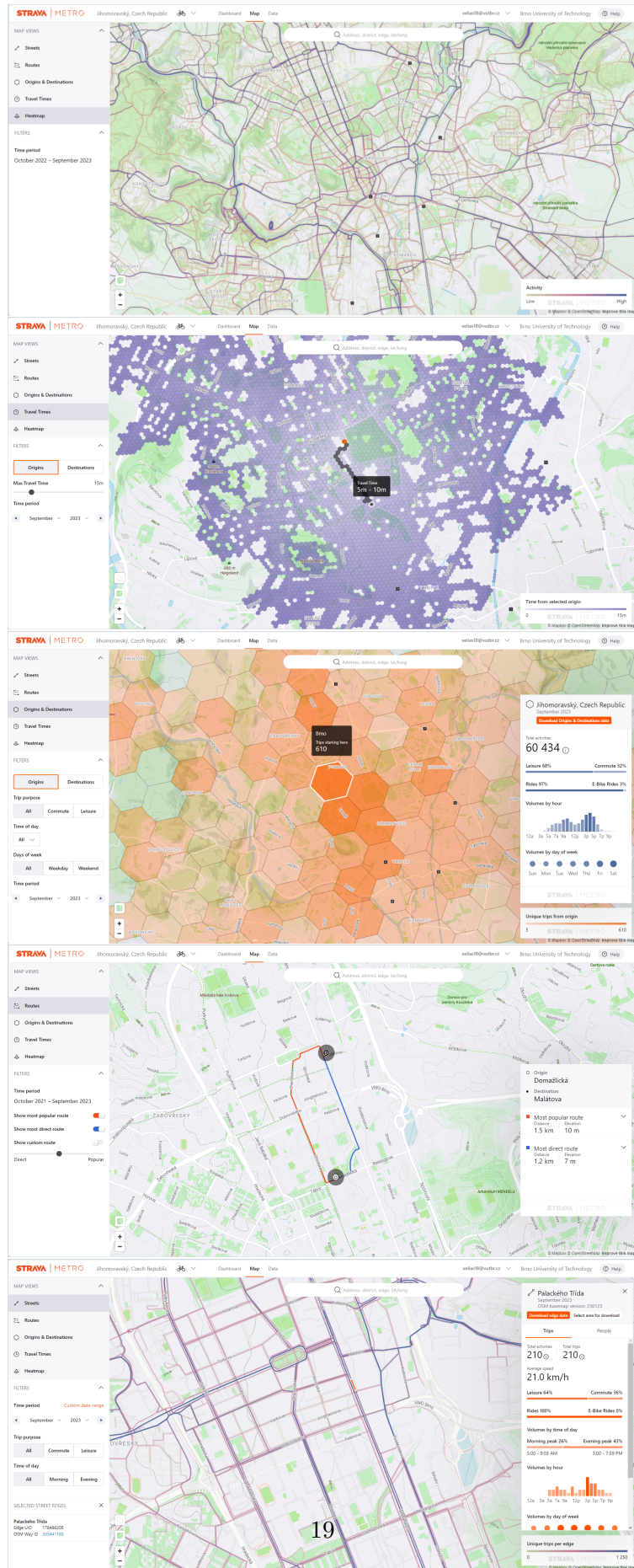


Obrázek 3.5: Ukázka aplikace strava heatmap

jednalo o volnočasovou jízdu nebo dojíždění, a které hodiny, dny v týdnu a měsíce byly nejvíce vytíženy.

Pohled Routes umožňuje najít cestu mezi dvěma body, které si vybereme. Mezi těmito body se nám vygeneruje nejpřímější a nejpoblíbenější cesta. Pohled Streets ukazuje heatmapu a celkovou statistiku zaznamenaných cest v daném kraji. Ve statistice můžeme vyčíst stejná data, jako v pohledu Origins & Destination, a k tomu je zde také možné si zobrazit podobnou statistiku, která je agregovaná podle uživatelů. Po zvolení ulice nebo její části se nám statistika vyfiltruje podle daného úseku. Zde je navíc uvedena průměrná rychlost.

Největší nevýhodou této aplikace je, že není volně dostupná a je potřeba požádat o přístup do ní, který je velmi omezený a společnost strava jej uděluje jen výjimečně.



Obrázek 3.6: Ukázka aplikace strava metro

Kapitola 4

Analýza

Pro správný návrh řešení, které budou uživatelé používat, je nejprve třeba si takové potencionální uživatele představit a zanalyzovat jejich potřeby a požadavky. Dále je třeba si prostudovat stávající řešení a zjistit jeho nedostatky a přednosti. Z těchto dvou analýz nám následně vyplynou výsledné požadavky na řešení.

4.1 Analýza uživatelů

Cyklistická data v Brně jsou veřejně dostupná, a tak k nim mohou přistupovat všichni uživatelé, kteří o ně mají zájem. Tyto uživatele můžeme rozdělit na dvě hlavní skupiny. První skupinou je široká veřejnost. Uživatelé z této skupiny budou používat námi vytvořené řešení primárně v rámci své zájmové aktivity. Druhou skupinou jsou pracovníci, kteří budou možné řešení používat k analyzování dopravní situace.

- **Veřejnost**

První skupinou uživatelů, která bude potenciální řešení používat, je široká veřejnost, pro niž představuje důležitý zdroj zajímavých informací a trendů v oblasti cyklistické dopravy v rámci města. Jejich motivací k využití možného řešení je především získání zajímavých informací a sledování trendů v oblasti cyklistické dopravy v městském prostředí. Tyto informace následně mohou využít při plánování svých cyklistických výletů nebo každodenních tras. Možné řešení jim umožní efektivněji využít dostupná data tím, že jim poskytne informace o nejvytíženějších trasách, které jsou preferované cyklisty. Díky analýze dat mohou uživatelé sledovat trendy v dopravě, například změny intenzity provozu v průběhu jednotlivých hodin či v různé dny v týdnu.

Nabízená funkcionality budoucího řešení jim umožní získat ucelený přehled o situaci v cyklistické dopravě, ať už se jedná o oblíbené trasy, časy největšího provozu nebo místa s výraznou intenzitou cyklistů. Tato data mohou být pro uživatele klíčová pro plánování trasy a volbu optimálního času pro cestování. Zároveň jim umožní lépe porozumět dynamice městské cyklistické dopravy, a přispět tak k vytváření udržitelnějších a efektivnějších dopravních systémů. S pomocí informací z cyklo-detektorů budou uživatelé schopni získat detailní pohled na provoz na konkrétních úsecích a identifikovat klíčové trasy, které mohou ovlivnit jejich rozhodnutí ohledně cestování.

- **pracovníci Data Brno**

Druhou skupinou uživatelů jsou odborní pracovníci institucí specializujících se na plánování a analýzu dopravy. Oproti široké veřejnosti mají tito uživatelé odlišný přístup

k získaným informacím, neboť je využívají na profesionální úrovni k účelům spojeným s jejich pracovními povinnostmi. Jejich hlavním cílem je zlepšovat životní podmínky obyvatel prostřednictvím efektivního řízení dopravy a infrastrukturních úprav, a za tímto účelem budou chtít použít nějakou aplikaci, která jim poskytne užitečné informace a statistiky o dopravě.

Pro tyto pracovníky představují data o cyklistické dopravě cenný zdroj informací. Pomáhají jim například při plánování nových cyklistických pruhů či cyklostezek tím, že analyzují, kde se cyklisté nejčastěji pohybují a kde by taková infrastrukturní úprava byla nejefektivnější a nejžádanější. Dále mohou využít historická data k hodnocení provedených úprav a sledování vývoje trendů v oblasti cyklistické dopravy. Tato analýza jim umožňuje lépe porozumět potřebám cyklistů a navrhnout opatření, která odpovídají aktuálním trendům a potřebám městského prostředí.

Díky kombinaci praktických znalostí a datové analýzy mohou tito odborníci efektivně přispívat k rozvoji cyklistické infrastruktury a celkovému zlepšení dopravní situace ve městech.

4.2 Analýza stávajícího řešení

Aktuální vizualizace je realizována pomocí ArcGIS platformy, popsané v kapitole 3.4. Tato platforma má však nedostatky při snaze vytvořit pokročilejší vizualizace, jelikož například přidávání vlastních prvků nebo předzpracování dat je velmi náročné a neefektivní. Tato platforma používá pro zobrazení dodatečných informací o cestách nebo o bodech pouze tabulkové zobrazení. Další vytváření pohledů na data, například ve formě grafů, a jejich různé filtrování je možné pouze ve zvláštním dashboardu, který je od mapy oddělen. Také zde chybí možnost filtrovat data pro heatmapu, například pokud by jsme chtěli data pouze za určitý rok.

4.3 Požadavky na řešení

V této sekci jsou rámcově shrnuty požadavky na řešení vyplývající z analýzy uživatelů a jejich využití potenciálního řešení a z analýzy stávajícího řešení. Tyto požadavky tvoří základ pro návrh nástroje, který by uspokojil potřeby uživatelů a vylepšil stávající aplikaci.

- Zobrazovaná data vykreslit do heatmapy - pro dobrou vizualizaci modelů s velkým počtem záznamů, které se vztahují k ulicím města, je vhodné zobrazení do heatmapy, jelikož poskytne uživateli na první pohled hrubou představu o dopravě v Brně.
- Pro jednotlivé úseky a detektory zobrazit detail - pro podrobnější prohlížení dat o úsecích či ulicích je vhodné tato data zobrazit až v detailním pohledu, jelikož není možné v rámci přehlednosti je zobrazit hned na první pohled.
- Filtrovat data podle časového intervalu - tato funkcionality se hodí pro případy, kdy chce uživatel vidět data pouze v určitém časovém intervalu.
- Vytvoření grafů a tabulek pro detailní zhodnocení provozu - umožní nám více pohledů na data.
- Automatické aktualizování datových sad z API města Brno - aby správa dat byla na straně poskytovatele dat (tedy město Brno), a také aby nevznikala duplicita dat.

Kapitola 5

Návrh

V této kapitole je zpracován návrh možného řešení na základě analýz a požadavků na řešení popsaných v předchozí kapitole. V následujících podkapitolách je nejprve popsána navrhovaná architektura, následně jsou popsány datové vstupy řešení a nakonec je popsán návrh grafického rozhraní.

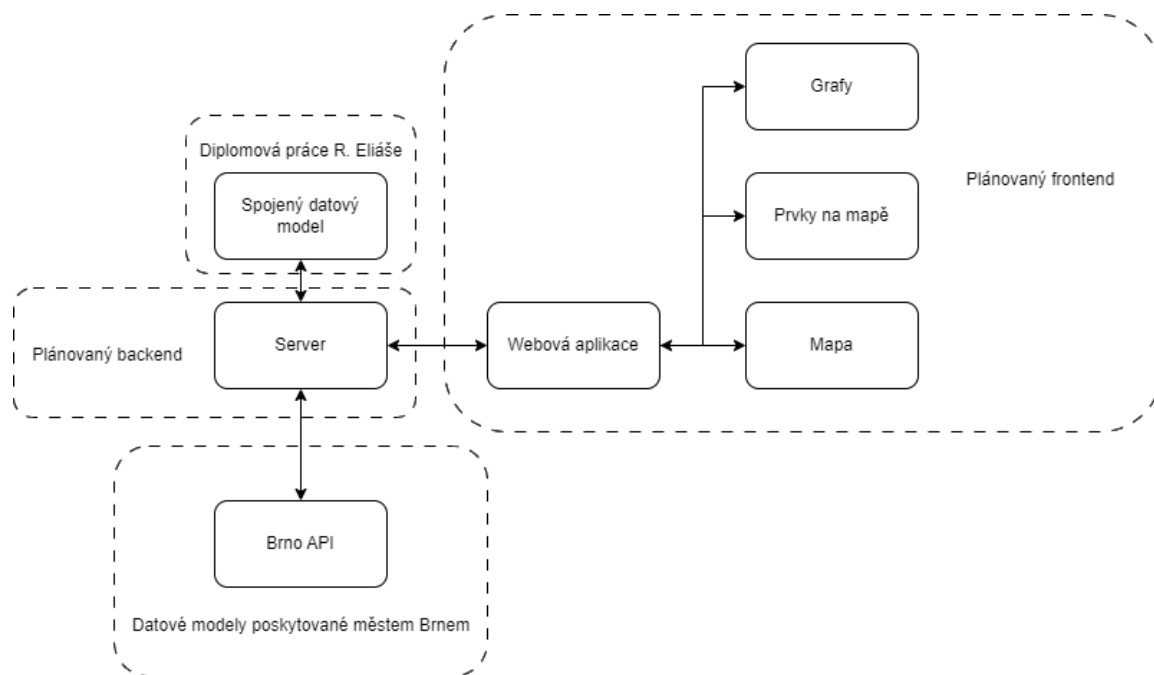
5.1 Architektura

Navrhované řešení nad daty města Brna tedy bude spočívat ve vytvoření grafické aplikace, která nahradí stávající ArcGis aplikaci. Výsledné řešení bude zobrazovat data z datového portálu města Brna, který je poskytuje skrz ArcGis API. Toto API však umožňuje na jeden požadavek poskytnout pouze 1000 - 2000 záznamů. To je v tomto případě nevyhovující, jelikož datová sada aplikace BikeToWork obsahuje více záznamů. Z toho důvodu bude potřeba vytvářet více požadavků a jednotlivé výsledky spojovat. Proto bude potřeba vytvořit samostatný backend server, který nám bude tato data poskytovat přes jeden endpoint pro výslednou aplikaci. Využití tohoto serveru je také výhodné z důvodu filtrování záznamů například pro cyklo detektory, protože Arcgis API využívá specifické rozhraní, ve kterém je třeba určit podmínku, požadované sloupce a výstupní formát. Toto rozhraní následně nahradíme standartním API ve tvaru například `/cyklodetektory/{id}`. Možné řešení si také bude potřebovat vyčíst datový model, který spojuje ostatní datové modely, vytvořený pomocí programu R. Eliáše. Tento model je uložen v GeoJson souboru, a tak je nutné ho zpřístupnit skrz API, jako je vidět na obrázku 5.1.

Z požadavků nám vyplývá, že bude nejlepší vytvořit webovou aplikaci spolu s lehkým backendovým API serverem.

5.2 Datové vstupy

V této sekci jsou popsány datové modely, se kterými bude výsledné řešení pracovat, a které bude zpracovávat vytvořené API. Jedná se o modely, které používá ve své diplomové práci R. Eliáš, a k nim je přidána historie cyklistických nehod v Brně. Tyto modely poskytují data v podobě GeoJSON a jsou všechny poskytovány městem Brno kromě spojeného datového modelu, který je výstupem diplomové práce R. Eliáše.



Obrázek 5.1: Návrh architektury

5.2.1 BikeToWork

Data pro tento model jsou získávána při každoroční květnové akci „Do práce na kole“, pořádané spolkem Automat. Datová sada nám ukazuje počty pěších a cyklistů v daném silničním úseku, a to seskupená za jednotlivé roky.

5.2.2 City Census

Data pro tento model pravidelně zpracovávají Brněnské komunikace, a.s. Data jsou od roku 2016 získávána na vybraných ulicích a v pravidelných intervalech po dvou letech. Pro každou ulici je pak pro vybraný rok zaznamenán počet cyklistů během všedního dne a v neděli. Hodnoty jsou udávány v desítkách cyklistů.

5.2.3 Cyklodetektory

Tento model získává data ze strategicky umístěných detektorů. Detektory počítají cyklisty a chodce, některé v obou a některé pouze v jednom směru. Zaznamenaná data se následně ukládají po hodinových blocích, které jsou identifikované koncem tohoto intervalu.

5.2.4 Spojený datový model

Spojený datový model je vytvořen pomocí programu vytvořeného v diplomové práci R. Eliáše. Tento model obsahuje souřadnice každé cesty, ke které připojí pomocí cizího klíče ostatní datové sady.

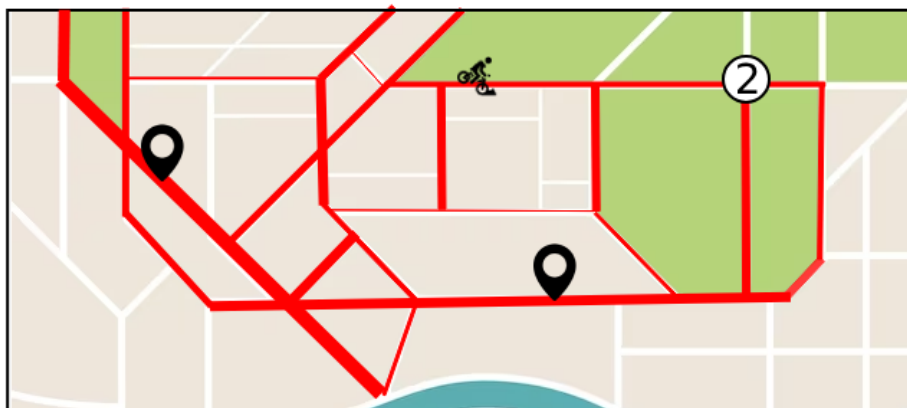
5.2.5 Cyklistické nehody

Tento model nám zobrazuje podrobnosti o cyklistických nehodách v Brně od roku 2010. Data se aktualizují jednou ročně. Zdrojem data je Policie ČR.

5.3 Návrh GUI

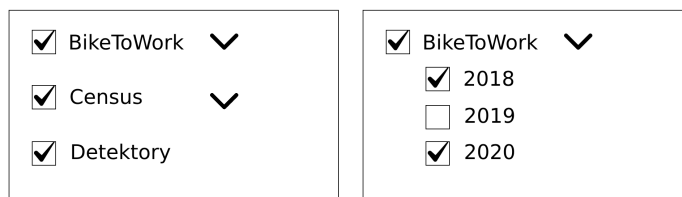
Grafická část výsledného řešení bude implementována na základě následujícího návrhu. Uživatelské rozhraní bude obsahovat následující části:

- **Hlavní část s mapou** - Hlavní část bude obsahovat mapu, do které se budou zobrazovat datové sady. Cyklodetektory se budou zobrazovat pomocí ikony. Po kliknutí na ikonu se nám zobrazí detail cyklodetektoru. Ostatní sady se budou zobrazovat pomocí zvýraznění cest. Data těchto sad se budou znázorňovat pomocí šířky jednotlivých cest, které budou tvořit takzvanou heatmapu, jako je vidět na obrázku 5.2. Pro určení šířky se nejprve sečtou všechny hodnoty jednotlivých datových sad podle filtru. Rozsah těchto hodnot se následně namapuje na šířku cesty. Po kliknutí na cestu se zobrazí detail jednotlivých cest a vybraná cesta se zvýrazní. Další vrstvu budou tvořit cyklistické nehody. Jelikož je záznamů s nehodami po městě hodně a zobrazovat ikony pro všechna místa nehod by bylo značně nepřehledné, bude třeba na místech s vysokou hustotou tyto ikony sloučit do takzvaných clusterů. Tyto clustery budou zobrazovány pomocí ikony a budou obsahovat číslo reprezentující počet ikon, které tento cluster zastupuje, jako je možné vidět na obrázku 5.2.

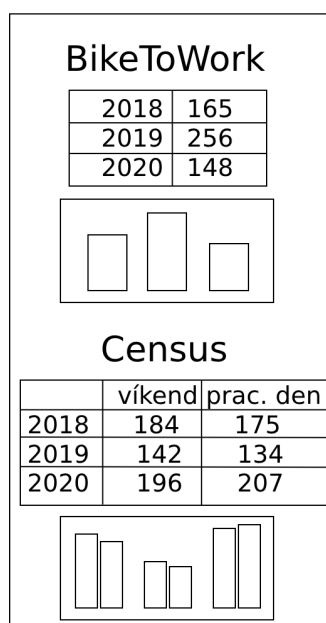


Obrázek 5.2: Hlavní část s mapou

- **Filtr na datové sady** - Ve filtru na datové sady bude možné zobrazovat a skrývat jednotlivé datové sady. Pro cyklodetektory bude mít tento filtr pouze funkci vypnutí/zapnutí. Pro cesty dostaneme zajímavější funkčnost, jelikož každá datová sada nám změní vzhled celé heatmapy, a tak budeme moci sledovat datové data po celém městě. Pro každou datovou sadu bude možné zobrazovat jednotlivé časové úseky, např. pouze bikeToWork z let 2018 a 2020, jako je vidět na obrázku 5.3.
- **Detail cesty** - V detailu cesty budeme moci vidět všechna data, která se vztahují k dané cestě, jako je vidět na obrázku 5.4. Pokud na dané cestě nebudou zaznamenána data z některého datasetu, vizualizace se pro něj nezobrazí. Detaily se budou zobrazovat jako boční panel v hlavní části mapy.

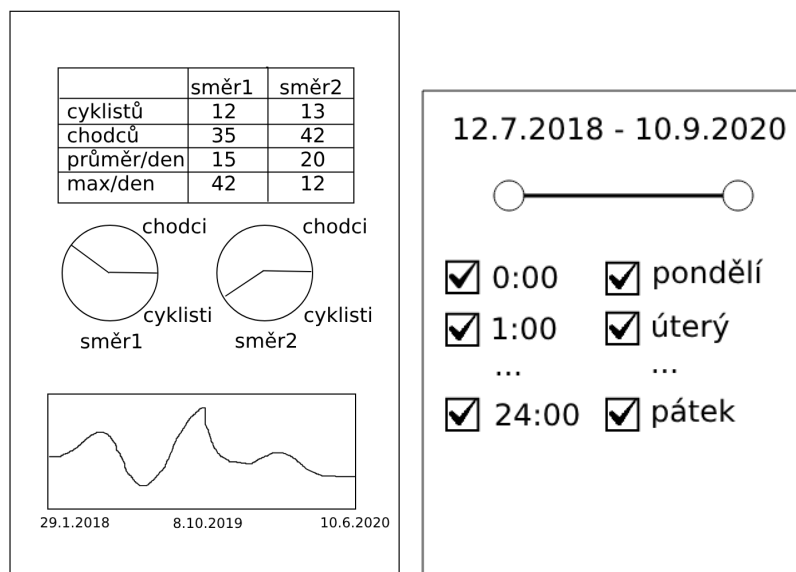


Obrázek 5.3: Filtry datových sad

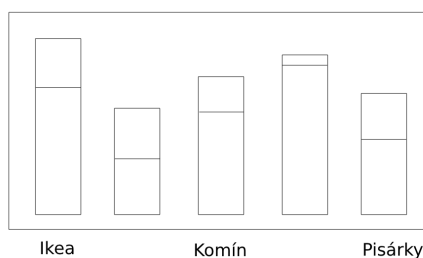


Obrázek 5.4: Detail cesty

- **Detail cyklo-detektoru** - V detailu cyklo-detektoru budeme moci vidět různé grafy pro oba směry, které detektor zaznamenává, jako například podíl chodců a cyklistů. Dále zde bude tabulka zobrazující kromě základních informací, jako je počet cyklistů a chodců v jednotlivých směrech, také další funkce jako průměr za den či hodinu. Všechna tato data budou závislá na časových filtrech. Tyto filtry budou moci omezit data jak podle data pořízení, tak zde půjde vyfiltrovat také jednotlivé dny či hodiny, jako je vidět na obrázku 5.5. Výsledná data, která se zobrazí v detailu, budou omezena intervalem datumů, a v tomto intervalu se zobrazí pouze specifické dny a hodiny podle filtru. Po kliknutí na jednotlivé grafy se nám otevře detail grafu.
- **Detail grafu** - V detailu grafu bude možné pozorovat stejnou problematiku, jako zobrazuje graf, na který bylo kliknuto, ale bude ji sledovat napříč všemi detektory, jako je vidět na obrázku 5.6. K zobrazení použije sloupcový graf, kde každý detektor bude mít jeden sloupec a každý sloupec se bude skládat ze dvou částí podle sledované problematiky.
- **Detail cyklistické nehody** - Detail cyklistické nehody bude zobrazovat tabulku obsahující informace o cyklistické nehodě, které nám poskytuje API datového portálu.



Obrázek 5.5: Detail cyklodetektoru s filtry



Obrázek 5.6: Detailu grafu

5.3.1 Návrh grafů pro cyklodetektory

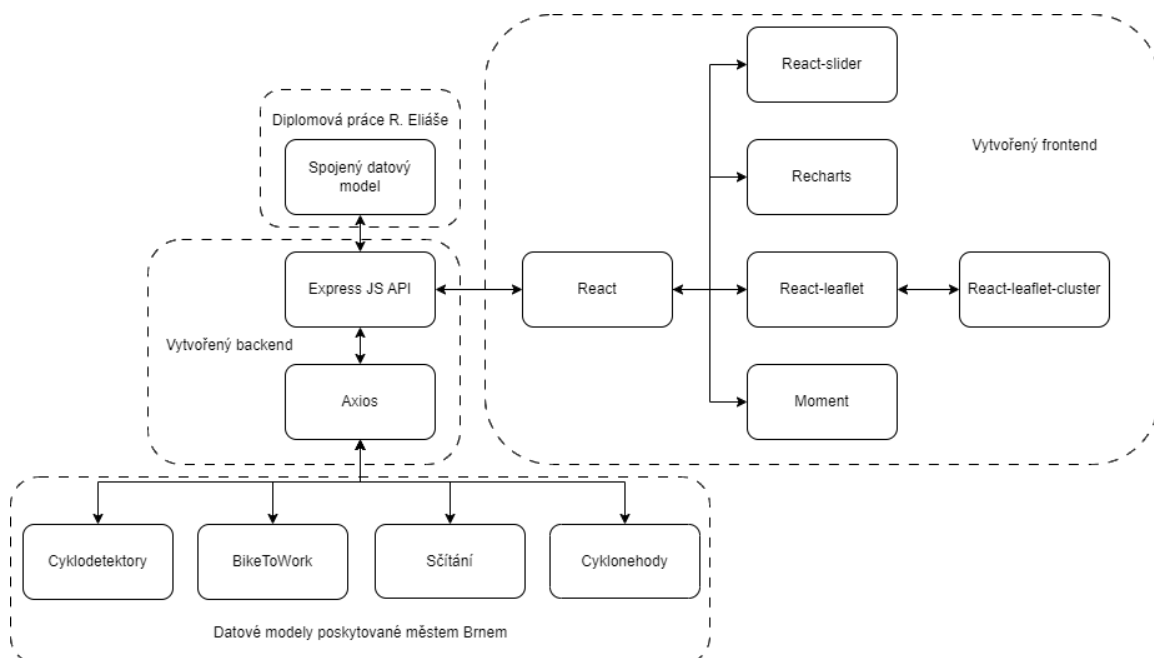
Cyklodetektory oproti ostatním datovým sadám nabízí díky frekvenci záznamů mnohem větší možnosti, jak data zobrazovat, nebo je nějakým způsobem filtrovat. Veškeré grafy budou závislé na časovém filtru detailu cyklodetektoru. V následující podkapitole jsou popsány příklady možných grafů:

- Koláčový graf s podílem chodců a cyklistů v každém směru. Tento graf sleduje celkový podíl chodců a cyklistů v každém směru.
- Lineární graf s podílem chodců a cyklistů celkem. Tento graf sleduje vývoj cyklistů a chodců v historii detektoru.
- Koláčový graf s podílem cestujících (jak cyklistů, tak chodců) o víkendu a ve všední dny.
- Koláčový graf s podílem cestujících (jak cyklistů, tak chodců) ve dne a v noci (den je od 6:00 do 22:00).
- Sloupcový graf pro porovnání poměru cyklistů a chodců ve všech cyklodetektorech.

Kapitola 6

Implementace

Tato kapitola se zabývá popisem implementace aplikace. Následující podkapitoly se zaměřují na dvě části implementace - na backendovou a frontendovou část. Ukázkou architektury systému lze vidět na obrázku 6.1



Obrázek 6.1: Výsledná architektura

6.1 Backend

Backendová část zpracovává požadavky frontendové aplikace, a pro tyto požadavky předzpracovává data. Pro vytvoření API byl vybrán express.js. Tato knihovna byla vybrána, aby byla udržena jednotnost v použití programovacích jazyků a pro její jednoduchost. Naše API bude obsahovat následující endpointy:

- **detectors**

Endpoint *detectors* pouze načítá data z ArcGis API a přeposílá je dále. Datová sada pro tento endpoint obsahuje jen malé množství záznamů, proto v tomto případě stačí

pouze jeden požadavek na ArcGis Api. Pro dotazy na ArcGis Api použijeme knihovnu `axios`.

- **bikeToWork**

Endpoint `bikeToWork` načítá data z ArcGis Api a přeposílá je dále. Zde je ale problém s tím, že datová sada `BikeToWork` obsahuje větší množství záznamů, než je Api schopné poskytnout na jeden dotaz. Proto zde posíláme dotazy na Api, dokud nám posílá data. Tyto dotazy limitujeme podle ID záznamu.

- **fullModel**

Endpoint `fullModel` nám poskytuje datový model vytvořený R. Eliášem. Tento model je čten ze souboru. Po přečtení jsou z dat odstraněny záznamy, kterým nebylo přiděleno id ani z `bikeToWork`, ani ze sčítání. Dále je bodům, které určují ulice, prohozena zeměpisná šířka a délka, jelikož frontend pracuje s opačným pořadím souřadnic.

- **census**

Endpoint `census` používá stejnou logiku jako `detectors`, ale jiný ArcGis endpoint.

- **detectorsHis/{id}**

Endpoint `detectorsHis` nám poskytuje veškerá historická data, která se vztahují k určitému detektoru. Tato data ale obsahují více záznamů, než je možné poskytnout na jeden dotaz, a tak je nutné opět posílat dotazy, dokud nedostaneme všechny záznamy.

- **compareCyclistsPedestrians**

Endpoint `compareCyclistsPedestrians` nám poskytuje data z historie cyklo-detektorů, které se seskupí podle id lokace, a porovnává podíl cyklistů a chodců.

- **compareWeekWeekend**

Endpoint `compareWeekWeekend` nám poskytuje data z historie cyklo-detektorů, které se seskupí podle id lokace, a porovnává podíl provozu o víkendu a ve všední den.

- **compareDayNight**

Endpoint `compareDayNight` nám poskytuje data z historie cyklo-detektorů, které se seskupí podle id lokace, a porovnává podíl provozu ve dne a v noci.

- **accidents**

Endpoint `accidents` nám zprostředkovává data o cyklistických nehodách stejně jako ostatní endpointy, které nemusí zasílat více dotazů.

6.2 Frontend

Frontendová část aplikace zajišťuje zobrazení dat a interakci s uživatelem. Pro vytvoření webové aplikace podle návrhu byla vybrána Javaskriptová knihovna `React`. Tato knihovna byla zvolena, jelikož je v dnešní době nejvíce používána. Při implementaci byly vytvořeny pro různé funkční bloky komponenty. Jednotlivé komponenty jsou následně popsány:

- **Aplikace**

Aplikace je hlavní komponentou, která zajišťuje načítání dat a zobrazování dalších komponent, pro které drží důležité proměnné. Dále zajišťuje filtraci dat pro heatmapu a výpočet vah pro zobrazení.

Pro výpočet vah je třeba nejprve si pro každý časový úsek zaznamenaný v datových sadách (například sčítání o víkendu 2020, nebo bikeToWork 2019, atd.) vypočítat maximální hodnotu. Poté jsou při inicializaci a při změně zobrazovaných datových modelů přepočítávány hodnoty každé cesty. Hodnota šířky je počítána od 0 do maximální hodnoty součtu všech atributů zvolených datových sad na dané cestě. Rozsah hodnot je následně namapován na šířku od 0.5 do 10 pixelů.

Tato komponenta také obsahuje hlavní mapovou komponentu s podkladovou mapou. Pro zobrazení mapy a ostatních prvků mapy je použita knihovna react-leaflet. Pro práci s daty je v aplikaci použita knihovna moment.

- **Heatmapa**

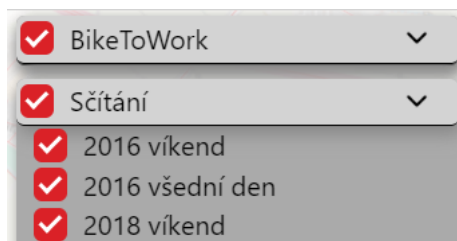
Heatmapa zobrazuje data z datových modelů City Census a BikeToWork do heatmapy pomocí zvýraznění cest, na kterých byla data nasbírána. Intenzitu provozu vykresluje pomocí šířky zvýrazněných cest jako je vidět na obrázku 6.2. Barva je zvolena červená pro udržení stylu se stránkami města Brna.



Obrázek 6.2: Ukázka heatmapy

- **Výběr datových sad**

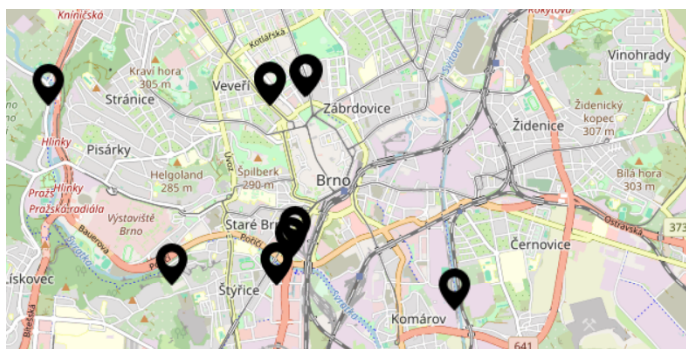
Výběr datových sad umožňuje vybírat datové sady, které chceme do mapy zobrazit. Sady *BikeToWork* a *Sčítání* ovlivňují vykreslení heatmapy. Pro obě sady je možné si rozkliknout detail a vybrat jednotlivé atributy datových sad, které chceme do heatmapy promítnout, jako je vidět na obrázku 6.3. Sady *Detektory* a *Nehody* se zobrazují pomocí markerů.



Obrázek 6.3: Ukázka výběru datových sad

- **Detektory**

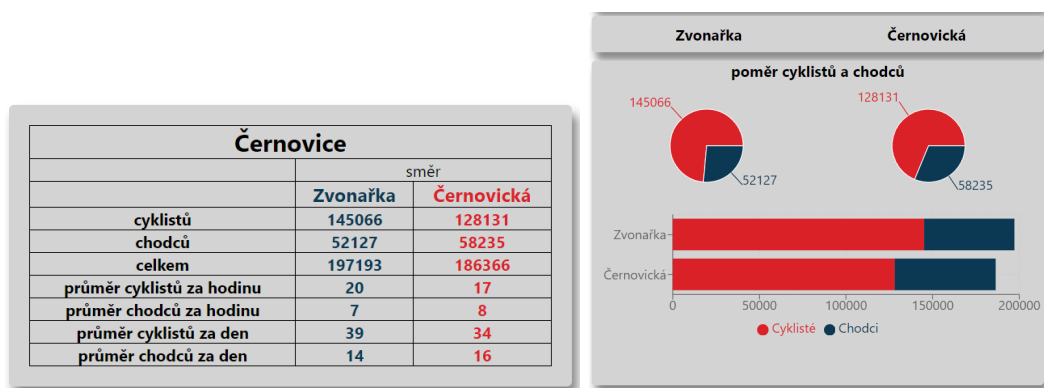
Komponenta *Detektory* zobrazuje do mapy detektory města Brna. Detektory jsou vizualizovány pomocí ikon, jako je vidět na obrázku 6.4. Pro ikony bylo třeba vybrat barvu, která bude kontrastní k barvě heatmapy, proto byla zvolena černá. Po najetí myší na ikonu se zobrazí popup s názvem cyklodetektoru a po kliknutí na ikonu se zobrazí jeho detail.



Obrázek 6.4: Ukázka cyklodetektorů na mapě

- **Detail detektoru**

V této komponentě můžeme vidět detailní informace o detektoru. Informace jsou zde prezentovány nejprve v tabulce, kde je možné vidět počty cyklistů a chodců v obou směrech. Dále je zde vidět také součet provozu a průměry chodců a cyklistů za hodinu a za den v obou směrech, jako je vidět na obrázku 6.5. Kromě informací v tabulce jsou zde také grafy pro poměr cyklistů a chodců, poměr provozu ve všední den a o víkendu a poměr provozu ve dne a v noci. Poměr chodců a cyklistů je vyzobrazen pomocí koláčových grafů, kde můžeme snáze sledovat poměr chodců a cyklistů a také pomocí sloupcového grafu, kde je lépe vidět rozdíl celkového provozu v obou směrech. Ostatní poměry jsou vyzobrazeny pouze pomocí koláčových grafů. Všechny koláčové grafy jsou zobrazeny po dvojicích vedle sebe pro oba směry a názvy obou směrů jsou uvedeny pouze jednou nad všemi grafy, jako je vidět na obrázku 6.5. Po kliknutí na jeden z grafů se zobrazí detail, který je popsán v odrážce *DiagramDetail*. Dále je zde také lineární graf pro zobrazení vývoje jak cyklistů, tak chodců za každou zaznamenanou hodinu. Veškerá data pro grafy a pro tabulku závisí na datovém filtru, který je popsán v dalších dvou odrážkách. Pro tvorbu grafů je použita knihovna *recharts*.

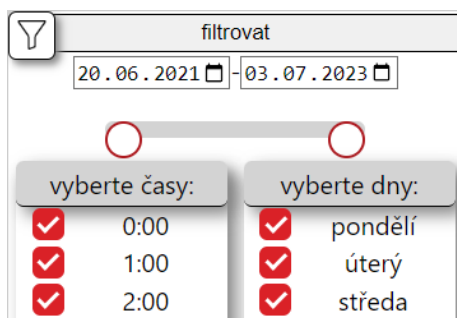


Obrázek 6.5: Ukázka částí detailu cyklodetektoru

- **Filtr Datumů**

Filtr Datumů je skryt pod ikonou a zobrazuje se až po kliknutí na ni, jako je vidět na obrázku 6.6. Filtr obsahuje dvě části. První částí je posuvník, který obsahuje část filtru pro rozsah datumů. Teto datum můžeme vybrat jak pomocí zadávacích polí, tak pomocí posuvníku s dvěma běžci. Posuvník byl vytvořen pomocí knihovny react-slider. Tato knihovna však neumožňuje použít posuvník pro data ve stringovém formátu, a proto bylo třeba datumy namapovat na rozsah od 0 do maximální hodnoty. Maximální hodnotu jsme vypočítali jako rozdíl dní minimálního data a maximálního data. Výsledný číselný rozsah je následně při změně převeden zpět na datum a zobrazí se do výběrových polí. Filtrování dat probíhá až po uvolnění posouvaného běžce pro lepší výkon aplikace.

Druhou částí filtru detektoru je výběr dnů a hodin, pomocí kterého můžeme vybírat, které dny a hodiny budou zahrnuté v datech pro grafy a tabulku. Data můžeme filtrovat pomocí checkboxů, které jsou zobrazeny ve dvou sloupcích, jeden pro hodiny a druhý pro dny, jako je vidět na obrázku 6.6. Filtrování dat proběhne až po kliknutí na tlačítko *filtrovat* pro lepší výkon aplikace.

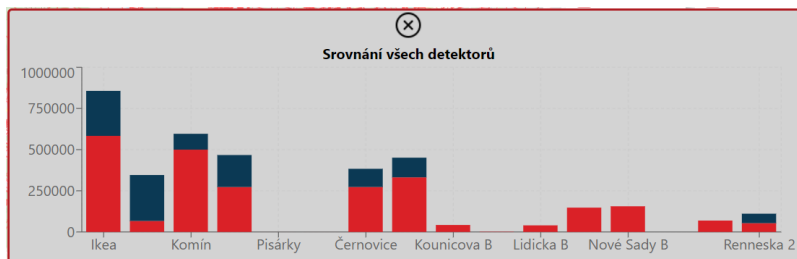


Obrázek 6.6: Ukázka filtru

- **Detail diagramu**

Detail diagramu ukazuje sloupcový graf, který sleduje stejný fenomén, jako koláčový graf, na který bylo kliknuto, jako je vidět na obrázku 6.7. Data grafu jsou však vybrána pro všechny detektory. Načítání dat je ale zdlouhavé, jelikož je pro historii detektorů mnoho záznamů a tyto záznamy nelze nějak agregovat podle datumu. To je způsobeno

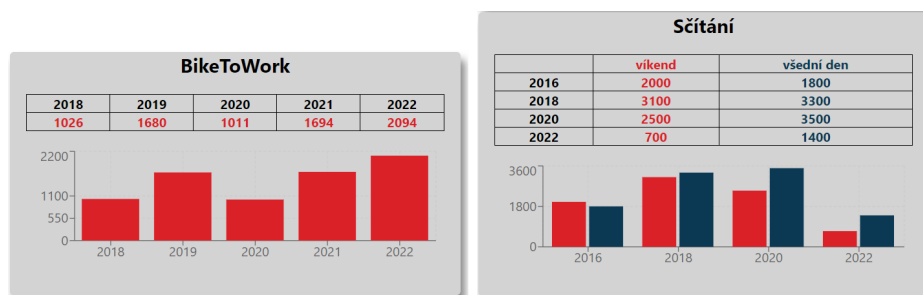
tím, že datумы jsou uloženy v českém formátu (DD.MM.YYYY) a nelze je tak filtrovat pomocí podmínky `EndOfInterval > timestamp '2022-01-01 00:00:00'`



Obrázek 6.7: Ukázka detailu diagramu

- **Detail cesty**

Detail cesty zobrazuje detailnější pohled na data jednotlivých sad promítnutých do heatmapy. Pro každou cestu se zobrazuje pouze detail sady, která je k dané cestě navázaná. Pro každou datovou sadu je pak zobrazená tabulka s jednotlivými záznamy a k ní sloupcový graf, jako je vidět na obrázku 6.8



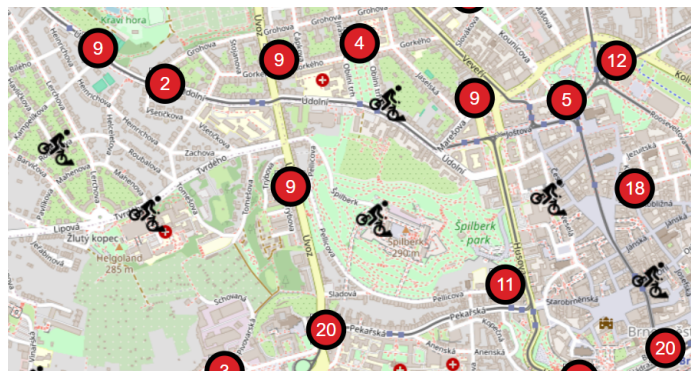
Obrázek 6.8: Ukázka detailů cesty

- **Nehody**

Komponenta *Nehody* přidává další vrstvu do zobrazované mapy a to dopravní nehody. Jednotlivé nehody jsou vykreslovány pomocí ikon, které se ve velkých shlucích spojují do tzv. clusterů, jako je vidět na obrázku 6.9. Tyto clustery jsou vytvořeny pomocí knihovny react-leaflet-cluster. Jednotlivé clustery zobrazují počet bodů, které zastupují. Po kliknutí na cluster se mapa přiblíží, a pokud je to možné, tak se zobrazí ikony, popřípadě se ve velkých shlucích opět spojí do clusteru. Po kliknutí na ikony se zobrazí detail nehody. Vrstva nehod se do mapy nezobrazuje defaultně, a proto je potřeba ji nejprve zapnout. Vrstvu nezobrazujeme defaultně pro zvýšení přehlednosti aplikace při prvním pohledu na ni, a pro urychlení prvotního načítání aplikace.

- **Detail nehody**

Detail nehody zobrazuje tabulku se všemi informacemi, jako je vidět na obrázku 6.10. V tabulce jsou uvedeny všechny důležité informace, které datový portál Brna nabízí, jako je například zavinění nehody nebo stav vozovky v době nehody.



Obrázek 6.9: Ukázka nehod a shluků

Brno-Jundrov	
Datum a čas	26.04.2019 10:00
Zavinění nehody	řidičem nemotorového vozidla
Stav vozovky v místě nehody	povrch suchý, neznečištěný
Povětrnostní podmínky v době nehody	neztížené
Viditelnost v době nehody	ve dne, viditelnost nezhoršená vlivem povětrnostních podmínek
Následky nehody	nehoda s následky na životě nebo zdraví
Druh vozidla účastníka	jízdní kolo
Alkohol u viníka	Ne
Druh komunikace nehody	komunikace místní
Stav cyklisty 24 hodin po nehodě	těžké zranění
Věková skupina	71 a více
Rozhledové poměry v místě nehody	dobré
Místo nehody	parkoviště přiléhající ke komunikaci
Druh nehody	havárie
Příčina nehody	řidič se plně nevěnoval řízení vozidla

Obrázek 6.10: Ukázka detailu nehody

6.3 Problémy při implementaci

Detail diagramu

Při vytváření detailu digramu a speciálního API pro každý diagram bylo zjištěno, že stáhnutí a seskupení veškerých historických dat trvá velmi dlouhou dobu, jelikož backend musí poslat velké množství požadavků (cca 180). Jako řešení se nabízí seskupit data podle časového úseku, například podle let. Následně bylo ale testováním zjištěno, že na GisApi není možné filtrovat data podle datumu. Testování probíhalo zadáním podmínky ve verzích *EndOfInterval* \geq *timestamp* '01-01-2022' a *EndOfInterval* \geq *timestamp* '2022-01-01', a v obou případech API posílalo i záznamy před tímto datem. Další možností je si při prvním přístupu stáhnout data na backend a dále je brát pouze z backendu. Tento přístup je však v rozporu s tím, že jsme nechtěli duplikovat data a brát je přímo z API města Brna, a tak byl ponechán původní přístup s tím, že uživatel musí čekat delší dobu na načtení dat.

Kapitola 7

Testování

Tato kapitola se zabývá testováním aplikace. Testování je důležitou součástí vývoje aplikace, jelikož pomáhá odhalit chyby a nedostatky v systému, podává přehled o výkonu aplikace a o reakcích na neočekávané stavy a chování. Testování aplikace probíhalo jak při jejím vývoji, tak po jejím dokončení různými uživateli. Tato kapitola se zaměřuje na testování uživateli a výsledky tohoto testování.

7.1 Uživatelské testování

Uživatelské testování je důležitou součástí testování, jelikož nám poskytuje pohled potenciálních uživatelů na aplikaci. Tento pohled je cenný především proto, že pohled programátora je velice odlišný od běžného uživatele, a tak může snadno přehlédnout nedostatky aplikace. Abychom mohli takové testování provést, je nejprve nutné si vytvořit testovací scénář, kterým se bude testování řídit. Postupování podle tohoto scénáře při testování je důležité, aby všichni testovaní uživatelé měli stejné podmínky. Uživatelé do testování jsou vybráni napříč věkovým a profesním spektrem.

7.1.1 Testovací scénář

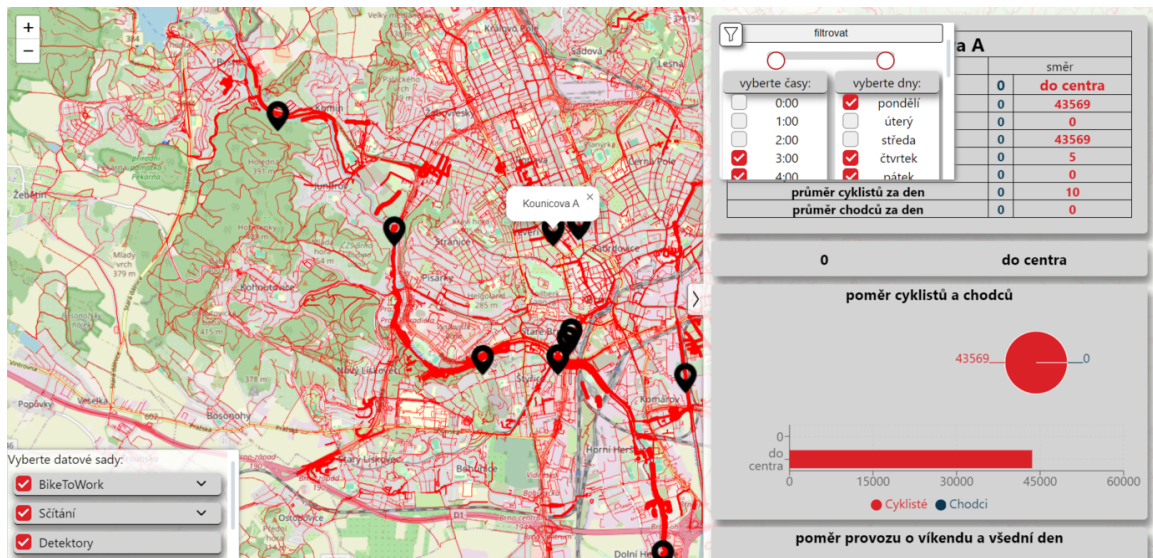
Při uživatelském testování této aplikace budeme postupovat následně: Nejprve uvedeme uživatele do hypotetické situace, ve které by se mohl setkat s aplikací. Uživateli je řečeno, že na datovém portálu města Brna narazil na aplikaci sledující cyklistickou dopravu po městě. Následně je uživatel ponechán, aby si aplikaci proklikal, čímž sledujeme, zda je aplikace dostatečně intuitivní a uživatel objeví všechny její funkcionality. Pokud nějakou funkcionalitu neobjeví, je mu ukázána, aby ji mohl zhodnotit. Poznatky těchto uživatelů, jsou následně zaznamenány. Ukázka aplikace při jejím testování je vidět na obrázku [7.1](#)

7.1.2 Testovací subjekt 1

Prvním testovacím subjektem je 48 letá skupinová vedoucí. Ačkoliv se testovací subjekt v profesním životě setkává s informačními systémy, není s jejich užíváním příliš sžit a většinou mu trvá se v aplikacích zorientovat.

Výsledky testovacího subjektu 1:

- V první řadě měl testovací subjekt problém s identifikací různých prvků na mapě, nebo s pochopením grafových zobrazení.



Obrázek 7.1: Ukázka aplikace při testování

- Při snaze filtrovat data na detailu detektoru si uživatel stěžoval, že pokud chce odfiltrovat pouze určitou hodinu, musí nejprve odklikat všechny ostatní.
- Při úzkých úsecích na mapě měl uživatel problémy se na ně trefit.
- Při upravení zobrazovaných datových sad byl testovací subjekt zmaten z absence vizualizace načítání.

7.1.3 Testovací subjekt 2

Druhým testovacím subjektem je 23 letý student Fakulty informačních technologií v Brně. Tento testovací subjekt je také programátorem, avšak o testovaném systému neměl do doby testování žádné informace.

Výsledky testovacího subjektu 2:

- Uživatel si nebyl na první pohled jist, co které datové sady znamenají.
- Při úzkých úsecích na mapě měl uživatel problémy se na ně trefit.
- U detailu detektoru by mohlo být popsáno, co přesně zobrazuje. Nebylo na první pohled jasné, že se dá na detail prokliknout.
- Vysoké číselky by mohly být po třech řádech vizuálně odděleny.
- Scrolování u datových sad není na první pohled dobře viditelné.

7.1.4 Testovací subjekt 3

Třetím testovacím subjektem je 24 letá asistentka pedagoga na základní škole. Tento subjekt se s informačními systémy a aplikacemi setkává v hojné míře mimo svůj profesní život.

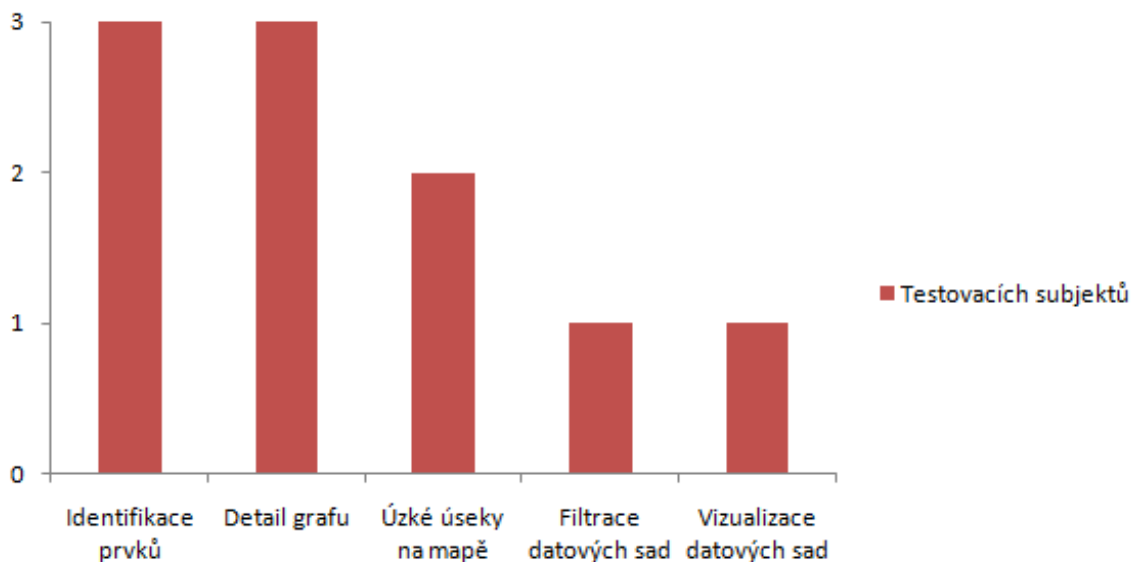
Výsledky testovacího subjektu 3:

- Subjekt si nebyl na první pohled jist, co znamenají jednotlivé ikony.

- Detail grafů u detektoru byl špatně dohledatelný.
- Subjekt přišel na časový filtr detailu až po delším pátrání.

7.1.5 Závěr testování

- Všechny testovací subjekty měly ve směr problém rozpoznat význam zobrazované heatmapy a ikon, jako je vidět na obrázku 7.2. Tento problém by se dal vyřešit přidáním legendy, která by zobrazovala základní popis ikon a heatmapy, popřípadě při kliknutí na ni by mohl být zobrazen detailnější popis vykreslených dat.
- Další častý problém byl při snaze kliknout na heatmapě na užší ulice. Tento problém by šel vyřešit dvěma způsoby. Prvním řešením by mohlo být nastavení minimální šířky pro kliknutí. Nemáme však ověřené, že knihovna pro vykreslení cest takovou funkci poskytuje. Dalším řešením je vykreslit přes cesty které jsou příliš úzké další průhlednou cestu, která bude širší a která bude sloužit pouze pro klikání.
- Testovací subjekty měly dále problém přijít na to, že existuje detail grafů. To že je detail grafu klikatelný je vizualizováno pouze obtažením daného bloku s grafem při najetí myši. Indikaci klikatelnosti by šlo rozšířit například o ikonu značící detail, která by se zobrazovala někde v bloku s grafem.
- Jedním z dalších nedostatků objevených při testování je absence možnosti zakliknout při filtrování na detailu detektoru všechny dny či hodiny. Tato funkcionality je zde užitečná zejména z důvodu vysokého počtu zaklikávacích polí.



Obrázek 7.2: Nejčastější nalezené problémy

7.2 Budoucí vývoj

I když je aplikace zcela použitelná pro monitorování cyklistické dopravy, stále existují možnosti, jak ji dále rozšiřovat. První nabízenou možností je přidání dalších grafů a statistik pro detail cyklodetektoru, jelikož historie cyklodetektorů nám poskytuje velké množství způsobů, jak data agregovat. Příkladem může být vývoj provozu v jednotlivých měsících nebo letech. Pro korektní statistiku by bylo také dobré pro každý graf uvádět intenzitu vzorkování, abychom si ověřili, že například v noci se nedělají vzorky méně často než ve dne, a nevzniká tak odchylka. Jednou z dalších možností, jak aplikaci rozšířit, je přidání filtrace nehod, které se zobrazí do mapy. Filtrovat nehody by šlo například podle toho, zda byl u účastníků přítomen alkohol nebo podle způsobené škody. Dále by bylo užitečné sestavit grafy pro dopravní nehody podle jejich atributů, například podle věkových skupin nebo podle nehod s vážným nebo lehkým zraněním.

Kapitola 8

Závěr

Cílem mojí práce bylo navrhnout a implementovat aplikaci, která umožní analyzovat datové sady poskytnuté městem Brnem spojených datovým modelem vytvořeným R. Eliášem v jeho diplomové práci. Samotnému návrhu a implementaci aplikace předcházelo důkladné nastudování geografických dat, mapových aplikací a jejich prvků. V další řadě bylo třeba zmapovat aktuální řešení a provést jejich analýzu spolu s analýzou potenciálních uživatelů a jejich požadavků na aplikaci. Po následném návrhu a implementaci byla aplikace uživatelsky otestována. Na základě výsledků testů byly navrženy budoucí vylepšení aplikace.

Výsledkem je aplikace skládající se z backendové a frontendové části. Backendová část nám vytváří API, které zprostředkovává data z API města Brna a z vytvořeného spojeného modelu. Toto API nám umožňuje jednoduše předzpracovávat data pro vizualizaci. Frontendová část následně zobrazuje data do heatmapy a pro jednotlivé prvky poskytuje detaily. V těchto detailech si uživatel může prohlédnout různé grafy a pohledy na jednotlivé datové sady. Výhodou této aplikace je, že ji lze snadno rozšířit o další datové sady a pohledy na data. Uživatelé pak mohou datové sady lépe analyzovat. Další výhodou je, že aplikace může snadno reagovat na změny v datových sadách. Nevýhodou aplikace je, že se k datům musí dostávat skrze ArcGIS API, což zpomaluje práci s datosety s velkým množstvím záznamů. Aplikace je také navržena tak, aby bylo možné snadno změnit datové sady, a tak ji lze nasadit i jinde než jen v Brně.

Literatura

- [1] ABELLANAS, M., AIELLO, A., HERNÁNDEZ, G. a SILVEIRA, R. I. Network drawing with geographical constraints on vertices. *Actas XI Encuentros de Geom. Comput.* 2005, s. 111–118.
- [2] ALMASSABI, A., BAWAZEER, O. a ADAM, S. Top NewSQL Databases and Features Classification. *International Journal of Database Management Systems*. Academy and Industry Research Collaboration Center (AIRCC). duben 2018, sv. 10, č. 2, s. 11–31. DOI: 10.5121/ijdms.2018.10202. ISSN 0975-5705. Dostupné z: <http://dx.doi.org/10.5121/ijdms.2018.10202>.
- [3] BRATH, R. a BANISSI, E. Font attributes enrich knowledge maps and information retrieval: Skim formatting, proportional encoding, text stem and leaf plots, and multi-attribute labels. *International Journal on Digital Libraries*. Springer Science and Business Media LLC. únor 2016, sv. 18, č. 1, s. 5–24. DOI: 10.1007/s00799-016-0168-4. ISSN 1432-1300. Dostupné z: <http://dx.doi.org/10.1007/s00799-016-0168-4>.
- [4] DVRPC. *Bicycle LTS & Connectivity Analysis* [online]. 2021. [cit. 2023-01-23]. Dostupné z: <https://www.dvrpc.org/webmaps/bike-lts/>.
- [5] ELIÁŠ, R. *Model intenzit cyklistické dopravy v Brně*. Brno, 2023. Diplomová práce. Vysoké učení technické v Brně, Fakulta informačních technologií. Vedoucí práce ING. JIŘÍ HYNEK, P. Dostupné z: <https://www.vut.cz/studenti/zav-prace/detail/141152>.
- [6] HAMMAD, K. A. I., FAKHARALDIEN, M. A. I., ZAIN, J. a MAJID, M. Big data analysis and storage. In: *International conference on operations excellence and service engineering*. 2015, s. 10–11.
- [7] HOGRAEFER, M., HEITZLER, M. a SCHULZ, H. The State of the Art in Map-Like Visualization. In: Wiley, červen 2020, sv. 39, č. 3, s. 647–674. DOI: 10.1111/cgf.14031. ISSN 1467-8659. Dostupné z: <http://dx.doi.org/10.1111/cgf.14031>.
- [8] JUBA, S. a VOLKOV, A. *Learning PostgreSQL 11: A beginner's guide to building high-performance PostgreSQL database solutions*. Packt Publishing Ltd, 2019.
- [9] KENNEDY, M. a KOPP, S. *Understanding map projections*. 2000.
- [10] KESSLER, F. C., BATTERSBY, S. E., FINN, M. P. a CLARKE, K. C. Map projections and the Internet. *Choosing a Map Projection*. Springer. 2017, s. 117–148.

- [11] KRAAK, M.-J. a ORMELING, F. *Cartography: Visualization of Geospatial Data*. CRC Press, červenec 2020. ISBN 9780429464195. Dostupné z: <http://dx.doi.org/10.1201/9780429464195>.
- [12] OLIVEIRA, F., NERY, D., COSTA, D. G., SILVA, I. a LIMA, L. A Survey of Technologies and Recent Developments for Sustainable Smart Cycling. *Sustainability*. MDPI AG. březen 2021, sv. 13, č. 6, s. 3422. DOI: 10.3390/su13063422. ISSN 2071-1050. Dostupné z: <http://dx.doi.org/10.3390/su13063422>.
- [13] ORLIK, A. a ORLIKOVA, L. Current trends in formats and coordinate transformations of geospatial data — based on MyGeoData Converter. *Open Geosciences*. Walter de Gruyter GmbH. leden 2014, sv. 6, č. 3, s. 354–362. DOI: 10.2478/s13533-012-0178-8. ISSN 2391-5447. Dostupné z: <http://dx.doi.org/10.2478/s13533-012-0178-8>.
- [14] PANG, C.-I. a BIUK AGHAI, R. P. Wikipedia world map: method and application of map-like wiki visualization. In: *Proceedings of the 7th International Symposium on Wikis and Open Collaboration*. ACM, říjen 2011, s. 124–133. WikiSym '11. DOI: 10.1145/2038558.2038579. Dostupné z: <http://dx.doi.org/10.1145/2038558.2038579>.
- [15] SIDDIQA, A., KARIM, A. a GANI, A. Big data storage technologies: a survey. *Frontiers of Information Technology & Electronic Engineering*. Springer. 2017, sv. 18, s. 1040–1070.
- [16] SNYDER, J. P. *Flattening the earth: two thousand years of map projections*. University of Chicago Press, 1997.
- [17] SNYDER, J. P. *Map projections used by the US Geological Survey*. US Government Printing Office, 1982. Dostupné z: <http://dx.doi.org/10.3133/b1532>.
- [18] SPOKANE. *City of Spokane Bicycle Level of Traffic Stress* [online]. 2019. [cit. 2023-01-20]. Dostupné z: <https://storymaps.arcgis.com/stories/c307b4dd61d64f90bfada67b7aa46bbf>.
- [19] STEFANAKIS, E. Web mercator and raster tile maps: two cornerstones of online map service providers. *GEOMATICA*. Canadian Science Publishing. červen 2017, sv. 71, č. 2, s. 100–109. DOI: 10.5623/cig2017-203. ISSN 1195-1036. Dostupné z: <http://dx.doi.org/10.5623/cig2017-203>.
- [20] STROHBACH, M., DAUBERT, J., RAVKIN, H. a LISCHKA, M. Big data storage. *New Horizons for a Data-Driven Economy: A Roadmap for Usage and Exploitation of Big Data in Europe*. Springer International Publishing. 2016, s. 119–141.
- [21] TERH, S. H. a CAO, K. GIS-MCDA based cycling paths planning: a case study in Singapore. *Applied Geography*. Elsevier BV. květen 2018, sv. 94, s. 107–118. DOI: 10.1016/j.apgeog.2018.03.007. ISSN 0143-6228. Dostupné z: <http://dx.doi.org/10.1016/j.apgeog.2018.03.007>.
- [22] ZHOU, H., XU, P., YUAN, X. a QU, H. Edge bundling in information visualization. *Tsinghua Science and Technology*. Tsinghua University Press. duben 2013, sv. 18, č. 2, s. 145–156. DOI: 10.1109/tst.2013.6509098. ISSN 1007-0214. Dostupné z: <http://dx.doi.org/10.1109/tst.2013.6509098>.