

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

Fakulta elektrotechniky
a komunikačních technologií

BAKALÁŘSKÁ PRÁCE

Brno, 2018

Jan Spurný



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA ELEKTROTECHNIKY A KOMUNIKAČNÍCH TECHNOLOGIÍ

FACULTY OF ELECTRICAL ENGINEERING AND COMMUNICATION

ÚSTAV TELEKOMUNIKACÍ

DEPARTMENT OF TELECOMMUNICATIONS

GEOGRAFICKÝ INFORMAČNÍ SYSTÉM V OBORU FINANČNÍCH SLUŽEB

GEOGRAPHIC INFORMATION SYSTEM IN THE FIELD OF FINANCIAL SERVICES

BAKALÁŘSKÁ PRÁCE

BACHELOR'S THESIS

AUTOR PRÁCE

AUTHOR

Jan Spurný

VEDOUCÍ PRÁCE

SUPERVISOR

doc. Ing. Petr Číka, Ph.D.

BRNO 2018

Bakalářská práce

bakalářský studijní obor **Teleinformatika**
Ústav telekomunikací

Student: Jan Spurný

ID: 186192

Ročník: 3

Akademický rok: 2017/18

NÁZEV TÉMATU:

Geografický informační systém v oboru finančních služeb

POKYNY PRO VYPRACOVÁNÍ:

Cílem práce je vytvořit geografický informační systém (GIS), který může sloužit jako klíčový nástroj a/nebo report pro rozhodování managementu subjektu podnikajícího v oblasti finančních služeb. Proveďte vhodnou lokalizaci dat vybrané instituce na mapové souřadnice, interní data instituce vhodně zkombinujte s daty veřejně publikovanými státní správou. Výsledek vizualizujte pomocí vámi vybrané volně dostupné mapové aplikace v podobě vhodně zvolených mapových kompozic. Systém implementujte ve vámi vybraném programovacím jazyku jako webovou aplikaci, která bude napojena na vzorek dat vámi vybrané instituce, která budou uložena ve vámi vybrané relační databázi.

DOPORUČENÁ LITERATURA:

- [1] WEISFELD, Matt A. The object-oriented thought process. Fourth edition. ISBN 978-0321861276.
- [2] KRYGIER, John a Denis WOOD. Making maps: a visual guide to map design for GIS. Third Edition. ISBN 978-1462509980.
- [3] Státní správa zeměměřictví a katastru: RÚIAN [online]. Praha: ČÚZK, 2016 [cit. 2017-08-23]. Dostupné z: <http://www.cuzk.cz/Uvod/Produkty-a-sluzby/RUIAN/RUIAN.aspx>

Termín zadání: 5.2.2018

Termín odevzdání: 29.5.2018

Vedoucí práce: doc. Ing. Petr Číka, Ph.D.

Konzultant: Ing. Stanislav Uchytíl, Ph. D.

prof. Ing. Jiří Mišurec, CSc.
předseda oborové rady

UPOZORNĚNÍ:

Autor bakalářské práce nesmí při vytváření bakalářské práce porušit autorská práva třetích osob, zejména nesmí zasahovat nedovoleným způsobem do cizích autorských práv osobnostních a musí si být plně vědom následků porušení ustanovení § 11 a následujících autorského zákona č. 121/2000 Sb., včetně možných trestněprávních důsledků vyplývajících z ustanovení části druhé, hlavy VI. díl 4 Trestního zákoníku č.40/2009 Sb.

ABSTRAKT

Práce pojednává o návrhu a vytvoření Geografického informačního systému pro firmu Partners Financial Service a.s.. Aplikace bude klíčový nástroj společnosti pro řízení a vytváření reportů.

Práce obsahuje návrh a realizaci tohoto systému s názvem GeoFiP a provázání veřejně dostupných dat různých institucí s interními daty společnosti.

Na základě získaných dat a zjištění možností pro zobrazení mapy byl vytvořen základ aplikace v podobě uživatelského rozhraní, logiky návaznosti dat, zobrazení mapového podkladu do aplikace a zajištění integrace aplikace GeoFiP do interního systému společnosti. Grafické rozhraní je vytvořeno pomocí jazyků HTML, JavaScript a CSS, přičemž logika zobrazení je tvořena back-end jazykem PHP s využitím MySQL databáze.

Výsledkem práce je funkční uživatelské menu zapadající do konceptu společnosti Partners Financial Services a.s., vytvoření metod pro zobrazení mapového podkladu a funkčním prvkem nastavení zobrazení dat, jen pro určitou část společnosti.

KLÍČOVÁ SLOVA

GIS, ČSÚ, ČÚZK, Partners, GeoFiP

ABSTRACT

The thesis deals with the analysis and implementation of a Geographical Information System for Partners Financial Service a.s. (GeoFiP). The application will be a mainly used by management to make decisions and for creating reports.

The thesis includes process of merging publicly available statistical data of various state institutions with internal company data.

Based on the input data and options for displaying the map the basic concept of the application was created such as user interface, the logic of merging the data, displaying of a map background and integration of GeoFiP into the company's internal system. Graphical interface is created by using HTML, JavaScript and CSS, while logic display is build up on PHP back-end using MySQL database.

The result of this thesis is a functional user menu that fits into the Partners Financial Services a.s design and user experience rules, implementation of methods for displaying a map background, and a functional element of data display settings for a specific part of the company.

KEYWORDS

GIS, ČSÚ, ČÚZK, Partners, GeoFiP

SPURNÝ, Jan. *Geografický informační systém v oboru finančních služeb*. Brno, 2018, 49 s. Bakalářská práce. Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, Ústav telekomunikací. Vedoucí práce: doc. Ing. Petr Číka, Ph.D.

PROHLÁŠENÍ

Prohlašuji, že svou bakalářskou práci na téma „Geografický informační systém v oboru finančních služeb“ jsem vypracoval(a) samostatně pod vedením vedoucího bakalářské práce a s použitím odborné literatury a dalších informačních zdrojů, které jsou všechny citovány v práci a uvedeny v seznamu literatury na konci práce.

Jako autor(ka) uvedené bakalářské práce dále prohlašuji, že v souvislosti s vytvořením této bakalářské práce jsem neporušil(a) autorská práva třetích osob, zejména jsem nezasáhl(a) nedovoleným způsobem do cizích autorských práv osobnostních a/nebo majetkových a jsem si plně vědom(a) následků porušení ustanovení § 11 a následujících autorského zákona č. 121/2000 Sb., o právu autorském, o právech souvisejících s právem autorským a o změně některých zákonů (autorský zákon), ve znění pozdějších předpisů, včetně možných trestněprávních důsledků vyplývajících z ustanovení části druhé, hlavy VI. díl 4 Trestního zákoníku č. 40/2009 Sb.

Brno

.....

podpis autora(-ky)

PODĚKOVÁNÍ

Rád bych poděkoval vedoucímu bakalářské práce panu doc. Ing. Petru Číkovi, Ph.D. , Ing. Stanislavu Uchytílovi, Ph.D. a panu Mgr. Petru Bartošovi za odborné vedení, konzultace, trpělivost a podnětné návrhy k práci.

Brno

.....

podpis autora(-ky)

OBSAH

Úvod	11
1 Technologie pro vývoj webových aplikací	12
1.1 Hypertext Preprocessor (PHP)	12
1.2 JavaScript	12
1.3 jQuery	13
1.4 Asynchronous JavaScript and XML (AJAX)	13
1.5 jQuery UI	14
1.6 Databázový systém MySQL	14
1.7 HyperText Markup Language (HTML)	14
1.8 Cascading Style Sheets (CSS)	15
2 Geografické informační systémy	16
2.1 GIS	16
2.2 Využití geografických informačních systémů	17
2.3 Tvorba map v GIS	18
3 Aplikace GeoFip	20
3.1 Architektura GeoFiPu	20
3.2 Hlavní softwarové nástroje	21
3.3 Zdroje dat pro mapové vrstvy	21
3.3.1 Provázání jednotlivých dat	21
3.3.2 Data z Českého úřadu zeměměřického a katastrálního	22
3.3.3 Data z Českého statistického úřadu	23
3.3.4 Data Partners Financial Services a. s.	24
3.3.5 Přidání dalších datových vrstev	24
3.4 Třídy a metody aplikace GeoFiP	26
3.4.1 Objektový model aplikace GeoFiP v PHP	27
3.4.2 Objekty v JavaScriptu aplikace GeoFiP	30
3.5 Vytváření mapové kompozice	31
3.6 Aplikační rozhraní Mapy firmy Seznam.cz	32
3.7 Integrace a zabezpečení aplikace GeoFiP	34
3.8 Interní systém FiP	35
4 Grafická část uživatelského rozhraní	36
4.1 Návrh grafického rozhraní GeoFiP	36
4.2 Komponenty menu	37
4.3 Realizace aplikace GeoFiP	38

5 Závěr	42
Literatura	43
Seznam symbolů, veličin a zkratk	46

SEZNAM OBRÁZKŮ

2.1	Použití datových vrstev	17
3.1	Architektura systému	20
3.2	Zjednodušený model návaznosti správních celků	22
3.3	Ukázka části webové rozhraní ČSÚ	23
3.4	Ukázka architektury vytváření mapy	28
3.5	Nákres architektury pro vytváření mapového prvku	29
3.6	Ukázka JavaScriptového objektu vrstva	30
3.7	Nákres posloupnosti při vytváření mapové vrstvy	31
3.8	Ukázka budoucí integrace aplikace GeoFiP do interního systému FiP	35
4.1	Návrh uživatelského prostředí	36
4.2	Základní zobrazení menu při načtení stránky	38
4.3	Otevřené menu výběru mapových vrstev	38
4.4	Příklad připnutí vrstvy do menu k zobrazení a vykreslení barevné škály	39
4.5	Zobrazení dat na různé územní celky	39
4.6	Výběr dat k zobrazení podle vybrané vertikální struktury společnosti	40
4.7	Zobrazení dat pro určitého uživatele	40
4.8	Nastavení barevného zobrazení vrstvy	41
4.9	Nastavení šrafovaného zobrazení vrstvy	41

SEZNAM TABULEK

3.1	Spojovací číselník	22
-----	------------------------------	----

SEZNAM VÝPISŮ

3.1	Ukázka zdrojového kódu AJAX Handleru	25
3.2	Ukázka zdrojového kódu PHP pro vytvoření objektu mapy	26
3.3	Inicializace Loaderu pro API Mapy	32
3.4	Ukázka JavaScriptového kódu pro zobrazení mapy	33

ÚVOD

Bakalářská práce na téma Geografický informační systém v oboru finančních služeb se týká návrhu architektury tohoto systému (dále GeoFiP) a tvorby grafického rozhraní webové aplikace. Aplikace bude integrována v rámci firemního interního systému (dále FiP), který nyní využívá více jak 1900 uživatelů ve společnosti.

GeoFiP bude nástroj pro rozhodování zejména managementu firmy Partners Financial Services a.s., nicméně aplikace může sloužit i jako nástroj pro zaměstnance společnosti na tvorbu reportů, neboť dle vertikální strukturou řízení definovaných uživatelských oprávnění umožňuje GeoFiP data zobrazit jen pro vybrané části společnosti.

Aplikace umožní zobrazovat interní data společnosti a vhodně je kombinovat s veřejně dostupnými daty z Českého statistického úřadu (dále ČSÚ), jako jsou například počty obyvatel, statistika dopravních nehod, průměrné výše důchodů a mezd atd. Jednotlivé datové vrstvy Partners Financial Service a.s. a ČSÚ se budou zobrazovat do mapových podkladů, na různé uzemní celky, dle dat Českého úřadu zeměměřického a katastrálního (dále ČÚZK), primárně nad územím České republiky, krajů, okresů a určitá data až do úrovně obcí s rozšířenou působností. Uživatel si bude moci určit, jakou podrobnost dat chce zobrazit a všechna vybraná data, v podobě mapových vrstev si nechat vykreslit nad mapovým podkladem. Charakter dat určuje způsob jejich reprezentace nad mapou v podobě bodových, liniových či polygonálních prvků, v rámci uživatelského rozhraní a přehlednosti zobrazení si uživatel dle svých potřeb bude moci sám upravit přednastavené grafické zobrazení mapové vrstvy, jako je výplň (barva nebo šrafa), barevná škála, průhlednost a pořadí zobrazovaných vrstev.

Podkladová mapa aplikace GeoFiP je získána od firmy Seznam.cz přes jejich veřejně dostupné aplikační rozhraní (API). Aplikace bude napsaná v jazyce PHP s využitím relační databáze MySQL. Grafické rozhraní aplikace bude kombinace HTML, JavaScriptu a jeho nástaveb v podobě jQuery a jQuery UI.

Bakalářská práce popisuje využívané jazyky a technologie, návrh a realizaci architektury systému a zpracování grafického rozhraní pro webovou aplikaci GeoFiP.

1 TECHNOLOGIE PRO VÝVOJ WEBOVÝCH APLIKACÍ

1.1 Hypertext Preprocessor (PHP)

PHP je skriptovací interpretovaný jazyk, pomocí něhož je možné vytvářet dynamické webové stránky. PHP je multiplatformní jazyk, běžící na straně serveru, který funguje na principu vykonání skriptu a k uživateli (klientovi) je ze serveru již odeslán celý výsledek [1].

Jedná se o dynamicky typovací jazyk což znamená, že datový typ není vázáný na proměnnou, ale na hodnotu této proměnné. PHP podporuje objektově orientované programování. Pro PHP existuje podpora v podobě mnoha knihoven, jako například Graphics Library (GD knihovna) pro zpracování obrázků nebo knihovny pro práci s PDF a XML dokumenty, které jsou od verze jazyka PHP5, jeho součástí [2].

PHP je obvykle využito s databázovým systémem MySQL, případně jinými databázovými systémy jako jsou PostgreSQL, nebo Firebird . Jedná se o nejrozšířenější skriptovací jazyk typu „*server-side*“ a je využit z více jak 80 % na webech a webových aplikacích [3].

Volbou tohoto jazyka pro aplikaci GeoFiP byla skutečnost, že ve společnosti Partners Financial Services a.s. je tento jazyk použit pro tvorbu dalších interních systémů.

1.2 JavaScript

Jedná se multiplatformní, interpretovaný, objektově orientovaný programovací jazyk běžící na straně klienta. Celý skript je v rámci *HyperText Markup Language* (HTML) kódu odeslán ke klientovi, kde se následně vykoná. Pomocí JavaScriptu lze dynamicky měnit obsah webových stránek, či webových aplikací, obsluhovat i tvořit menu a vytvářet interaktivní prvky.

Protože se jedná o technologii, která je prováděna na straně klienta je možnost, že někteří uživatelé mohou mít podporu skriptování zakázanou, respektive jejich webový prohlížeč nemusí skriptování vůbec podporovat. Dále JavaScriptové aplikace využívají přímo výkon uživatelského stroje a rychlost se odvíjí od výkonu zařízení. V rámci klientského zařízení, kde se vykonává skript je možné tento skript přechíst nebo stáhnout a uložit přes nástroj integrovaný ve většině internetových prohlížečů. Je závislý na internetovém prohlížeči, a proto v některých prohlížečích nemusí fungovat všechny naprogramované komponenty.

Zápis syntaxe technologie JavaScript je „case sensitive“, a proto záleží na velikosti písmen. Proměnné jsou dynamicky typované což znamená, že proměnná se typuje podle zapsané hodnoty. K objektům se JavaScript chová jako k asociativnímu poli. Pro JavaScript bylo vytvořeno mnoho knihoven, nástavb a frameworků například React, jQuery, D3.JS, Web Components nebo Angular JS [4, 5].

1.3 jQuery

jQuery je jedna z nejrozšířenějších knihoven pro JavaScript a jedním z jejich hlavních přínosů je řešení problémů s nekompatibilitou prohlížečů s JavaScriptem [6]. Je to nástroj pro snadnou a jednoduchou úpravu a manipulaci s webovými stránkami nebo webovou aplikací. Usnadňuje práci s JavaScriptovými objekty, kdy se vytvoří jQuery objekt a s ním se poté v rámci aplikace pracuje. Příkazy jQuery se zapisují přímo do kódu HTML. Skript pracuje s programovými prvky jako s objekty pomocí „tečkové“ notace.

Knihovna se do kódu linkuje, což znamená, že není součástí JavaScriptu. V naší aplikaci linkujeme jQuery verze „*jquery-3.2.1.js*“ [7, 8].

1.4 Asynchronous JavaScript and XML (AJAX)

AJAX využívá *XML HttpRequest*, což je rozhraní pro komunikaci se serverem, které odesílá a přijímá data ve formátu *JavaScript Object Notation* (JSON), *Extensible Markup Language* (XML), HTML, JavaScript a nebo textového formátu.

Tato technologie se využívá pro změnu obsahu webových stránek nebo aplikace bez toho, aniž by se musela celá webová stránka obnovovat (což je jinak standardní postup) [9]. Z toho plyne, že webová stránka nebo aplikace kontaktuje server a obdrží od něj jen požadovaná data bez přenosu dat, o které nežádáme. AJAX díky tomu urychlí práci, protože uživatel nemusí čekat na to, aby se načetl znovu celý obsah webu, aniž by se celý obsah změnil, ale žádá se změna pouze části obsahu webu. Díky tomu je menší zátěž na síťové připojení a využití kapacity linky, popřípadě nižší datová náročnost například při mobilním připojení.

Nevýhodou je, že nelze použít funkci zpět ve webovém prohlížeči (abychom dostali zpětně asynchroně načtený obsah), jelikož obsah je načítán mimo hlavní webovou aplikaci nebo webovou stránku.

Při využití technologie AJAXu je *Uniform Resource Locator* (URL) adresa většinou neměnná. Proto nejde stránku, ve které je například vyplněný formulář, odeslat

někomu jinému pomocí URL adresy s jeho vyplněnými údaji, pokud tuto funkci neumožňuje přímo aplikace jinou službou.

AJAX neošetřuje některé stavy, což při opakovaném dotazu na webovou službu může vést k problémům s přenosem a například zahlcením sítě [10, 11].

1.5 jQuery UI

jQuery UI je nadstavba klasického jQuery. Tato nadstavba obsahuje ovládací komponenty a jejich obslužné metody pro webové aplikace. Jsou to metody pro spojení prvků v aplikaci s funkcemi, které mají vykonávat výsledné grafické zobrazení. Práce s jQuery UI je stejná jako s jQuery. Tuto nadstavbu v podobně souboru knihovny, je vhodné mít přímo na serveru, kde je využita [12].

jQuery UI obsahuje:

- UI Core – což jsou základní funkce pro widgety, efekty a další funkce,
- Interactions – funkce k interakci mezi aplikací a uživatelem,
- Widgets – obsahuje grafické prvky jako jsou například tlačítka, kalendáře, posuvníky nebo dialogová okna [13].

1.6 Databázový systém MySQL

Jde o multiplatformní relační databázový systém, kde dotazovací jazyk je odvozený ze standardního databázového dotazovacího jazyka SQL (*Structure Query Language*). Systém MySQL pracuje na principu „klient-server“. Na straně serveru se řeší uložení dat, jejich organizace, údržba, ukládání a vyhledávání dat, spolu se správou uživatelů systému. Na straně uživatele je databázový klient, jehož funkcí je komunikace se serverem a na základě autorizace může žádat od serveru data nebo je přepisovat.

Systém MySQL využívá tradiční model relační databáze. Data jsou uložena v tabulkách, kde řádek tabulky představuje jeden záznam, sloupce tabulky pak atributy tohoto záznamu. Některý z atributů záznamu může být nastaven jako primární klíč, jehož hodnota je v rámci tabulky vždy jedinečná [14, 15].

1.7 HyperText Markup Language (HTML)

HTML je značkovací jazyk, určený pro tvorbu webových stránek. Pro svoji činnost využívá protokol HTTP (*HyperText Transfer Protocol*). Nejnovější verze jazyka je HTML 5.2, která opravuje předešlé verze a přidává některé funkcionality. Jde zejména o *Drag and Drop* umožňující elementy přemísťovat, canvas pro práci s vykreslováním 2D grafiky nebo změny v syntaxi.

Kód je kombinace klasického textu a značek jazyka, tyto značky se nazývají tagy. Jazyk není „case sensitive“, ovšem při zápisu URL cesty k dalším souborům je doporučeno zachovat konvenci ve velikosti znaků. Značky jazyka jsou párové a nepárové. Párové značky zavírají vždy obsah a neměly by se křížit. Nepárové značky neobklopují obsah, ale vykonávají příkaz dle svých atributů [16, 17, 18].

1.8 Cascading Style Sheets (CSS)

Jazyk CSS byl navržen za účelem oddělení popisu vzhledu a obsahu webové stránky. CSS popisuje vizuální projev webové stránky nebo aplikace. Tedy jak například má vypadat text, jeho barva, velikost, font, vzhled podkladu, tvar kurzoru v různých situacích atd. Nejnovější verze jazyka je CSS4.

Styly můžeme zapisovat několika způsoby:

- inline – styly vepisujeme přímo k elementům v kódu HTML pomocí atributu `style`,
- definice v HTML kódu – styl nadefinujeme v HTML kódu a voláme pomocí identifikátoru stylu,
- externím souborem – vytvoření stylového externího souboru připojeného do HTML souboru.

První dva způsoby se využívají ve velmi malém procentu případů, nejčastějším použitým způsobem je vytvoření a připojení externího souboru. Důvodem využití poslední varianty spočívá v přehlednosti kódů, jak v souboru HTML, tak v souboru CSS.

Příklad kódu pro nalinkování stylového souboru [19]:

```
< link href = 'style_files/styl_menu.css' rel = 'stylesheet' / >
```

K webovým elementům se přistupuje pomocí takzvaných selektorů.

Existují čtyři druhy selektorů:

1. typový selektor – styl se aplikuje na všechny prvky s příznakem,
2. selektor třídy – styl se aplikuje na všechny elementy dané třídy,
3. ID selektor – styl se aplikuje na všechny elementy s daným ID prvku,
4. selektor následníka – kde styl se aplikuje na všechny elementy uvnitř daného elementu.

Kaskádové styly disponují tzv. pseudotřídami, které obohacují pouze určitou část elementu, kde jsou styly aplikovány. V praxi to znamená, že elementům s určitým stylem a zároveň elementům dle definic pseudotřídy přidáme další styl [20].

2 GEOGRAFICKÉ INFORMAČNÍ SYSTÉMY

Kapitola zabývající se základní popisem geografických informačních systémů (GIS) v obecné rovině, s praktickým využitím pro aplikaci GeoFiP a metodikou tvorby geografických informačních systémů.

2.1 GIS

Geografické informační systémy jsou nástroje pro analýzu, manipulaci a vizualizaci dat obvykle nad mapou. Usnadňují porozumění nevizuálních dat pomocí vizualizace, kdy data jsou navázána na souřadnice souřadného systému mapy. Díky zobrazovacím možnostem těchto systémů můžeme porovnávat a analyzovat data v různých oblastech a tím zjišťovat vzájemné vazby mezi místy.

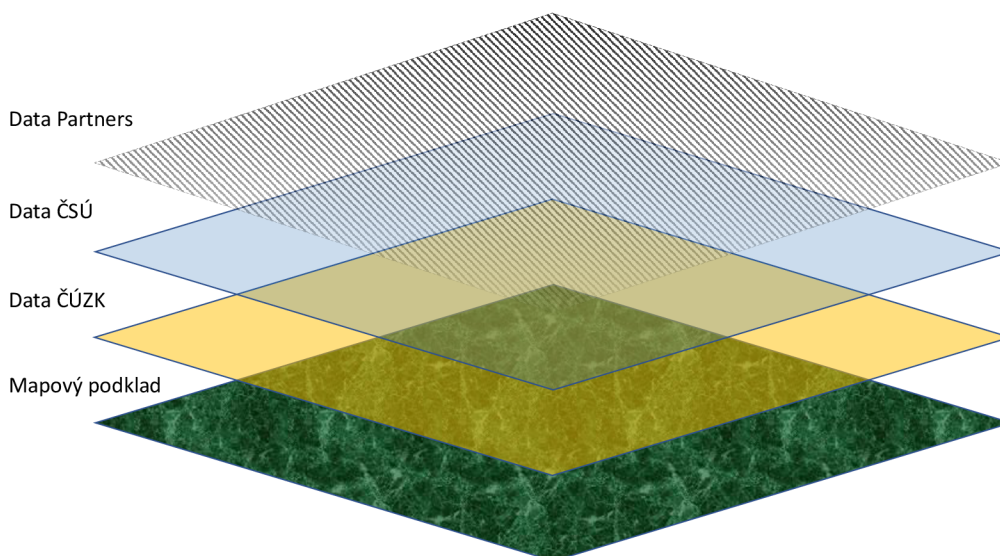
Aplikace mohou zahrnovat:

- kartografická data – informace například o poloze měst, silnic, ulic,
- fotografická data – letecké snímky kartografických dat,
- tabulková data – informace například o počtu obyvatel, průměrných mzdách a dopravních nehodách.

Systémy GIS využívají překrytí jednotlivých datových vrstev a tím umožňují zobrazit datovou kompozici nad mapovým podkladem a tak určit jejich geografické vztahy (obr. 2.1). Navíc můžeme zobrazovat data z různých zdrojů (ČSÚ, ČÚZK, interní data společnosti Partners Financial Services a. s.). Je pouze potřeba je převést do stejného souřadného systému a měřítka, který má použitý mapový podklad.

Typy formátů dat v GIS:

- Rastrové – ukládají informace v podobě pixelů, využívají se pro ukládání dat například satelitních snímků a ortofotomap,
- Vektorové – propojují data s popisy území a uchovávají data o území v podobě bodů, linií a polygonů (hranice krajů, okresů, měst,...)[21].



Obr. 2.1: Použití datových vrstev

2.2 Využití geografických informačních systémů

Nástroje GIS mají mnohostranné použití. Umožňují vytváření statistik, dlouhodobé sledování změn a vývoje, analýzy prodeje, tvorbu studií, analýz a modelů, které se vážou na dané území. Lze velmi jednoduše zjistit a porovnat například průměrnou výši mzdy s počtem společností v daném oboru na sledovaném území s jiným územím a poté plánovat rozvoj do jiných částí území. GIS systémy mají velký potenciál k zjednodušení a zrychlení práce a k zhodnocení vývoje na sledovaných územích.

Příklady využití GISů:

- státní správa a samospráva – tvorba územních plánů nebo strategie rozvoje,
- krizové řízení – využití integrovaného záchraného sboru (IZS ČR) v krizových situacích jako jsou například povodně,
- katastrální mapa ČÚZK – evidence majetku, nemovitostí, parcel,
- správa inženýrských sítí,
- vizualizace modelů a jejich zasazení do území [24].

Systém GeoFip přináší tyto přednosti GISů do oboru finančních služeb, kdy data firmy dokáže promítnout nad mapový podklad a tím nevizuálním datům dát vizuální charakter. Potenciál toho zobrazení je v tom, že usnadní a zrychlí analýzu dat, oproti tabulkovým datům, kde je porovnání pouze kvantitativní. Vizualizací kvantitativních dat a správným grafickým zobrazením, je usnadněné porovnání rozdílů

mezi jednotlivými oblastmi, a tím uživateli výrazně zjednodušuje a zpřehledňuje data než jejich klasické tabulkové vyjádření.

2.3 Tvorba map v GIS

Základní podoba geografických informačních systému je tvořena mapovými vrstvami. Mapové vrstvy jsou tzv. „průhledné fólie“, které jsou na sebe vrstveny a tím vytváří vizuální slučování mapového podkladu a vizualizací dat. Každá takto vytvořená mapová vrstva má svá pravidla tvorby.

Základní pravidla tvorby:

1. Logické oddělení různých typů dat do různých mapových vrstev.
2. Jedna mapová vrstva je pouze jeden datový soubor.
3. Vhodná barevná kombinace a typy zobrazení mapových vrstev.
4. Vhodné zobrazovací měřítko.
5. Vhodná podrobnost dat.

Prostorové vztahy mezi jednotlivými mapovými vrstvami musí být jednotné, to znamená, že pro každou mapovou vrstvu v mapové kompozici musí existovat jednotné měřítko zobrazení korespondující s ostatními mapovými vrstvami. Při tvorbě mapové vrstvy je nutné mít správný výběr vhodného zobrazovacího elementu pro zobrazovaná data. Pro různé typy dat je vhodné volit různé typy zobrazení v podobě bodů, linií, polygonů.

- Body – Zobrazují místo určení (poloha obchodu).
- Linie – Zobrazení sítí (rozvody energií).
- Polygony – Zobrazují data v ploše (kraje, okresy, lesy, pole).

Slučováním mapového podkladu a mapových vrstev vzniká výsledná mapa. Tato výsledná kompozice a prvky v mapě mají všeobecné zásady.

Všeobecné zásady tvorby map:

- Rozložení informací do celé mapy, bez hluchých míst.
- Jednoznačnost, bez příliš velkých výrazových přechodů.
- Čitelnost mapy jako celku, tak i detailů v ní zobrazených.
- Srozumitelnost mapy s vysvětlující legendou pro jednotlivé elementy, pokud je to nutné.
- Hlavní prvek mapy, který udává nejdůležitější informaci musí být dominantní.

Každá mapová kompozice by měla obsahovat prvky, které informují uživatele, co je zobrazeno.

Základní kompoziční prvky mapy:

- Název mapy
- Legenda
- Měřítko
- Mapové pole

Název mapy je údaj, který si mezi prvními uživateli přečte a zjistí co se mu tak zobrazuje bez jakékoliv předešlé zkušenosti.

Zásady tvorby názvů map jsou následující:

- Na prvním místě název zobrazovaných dat.
- Na druhém místě prostorové vymezení (územní celek).
- Poslední údaj je časové vymezení (rok zisku dat).

Legenda mapy informuje a vysvětluje význam použitých barevných stupnic, zobrazovacích značek a jedná se o klíčovou komponentu mapy z pohledu toho, že uživateli dává jasnou zprávu o tom, co se a jakým způsobem v mapové vrstvě zobrazuje.

Měřítko mapy zobrazuje informace o prostorovém rozlišení v mapové vrstvě.

Mapové pole je finální zobrazení mapy, ve kterém je obsáhlý mapový podklad a zobrazovaná data v podobě mapových vrstev.

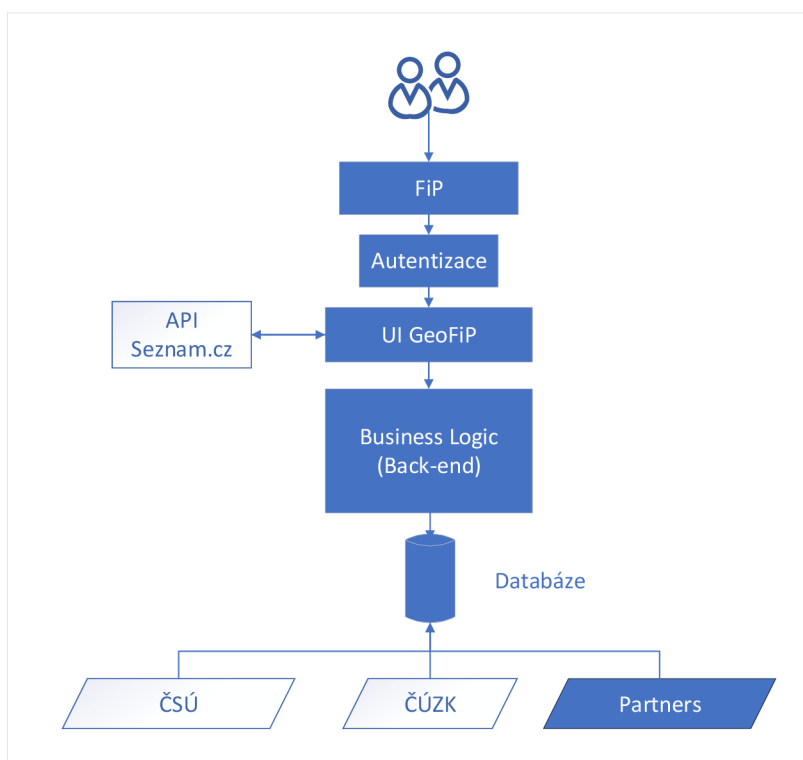
Základní metodou při tvorbě map je kombinace několika mapových vrstev přes sebe (použité v aplikaci GeoFiP) anebo kombinací předem vybraných dat do jedné mapové vrstvy. Při tvorbě map vzniklých kombinací dat do jedné mapové vrstvy, je veškerá rozlišovací schopnost dána pouze například jednou barevnou škálou nebo jedním typem zobrazení. Data dle toho, co zobrazují musí být upravena (nejčastěji přepočítána), pokud se nejedná o prosté zobrazení (př. počet dopravních nehod), ale kombinaci dvou a více hodnot (př. počet dopravních nehod na počet registrovaných vozidel), před zobrazením do mapové vrstvy. Po přepočtu, jsou data v seznamu, kde jedna výsledná hodnota je naasociována jednomu územně správnímu celku. Naopak při kombinaci několika mapových vrstev přes sebe se zobrazují prostá (neupravená) data, která kombinací s další mapovou vrstvou získávají vyšší vypovídající hodnotu. Lze tak z mapové kompozice vyčíst základní data jednotlivých vrstev, tak i jejich informace v podobě kombinace dvou vrstev [22, 23].

3 APLIKACE GEOFIP

Aplikace GeoFip bude součástí interního systému společnosti Partners Financial Services a. s. Bude se jednat o aplikaci pro nejvyšší management jako strategický nástroj pro rozhodování. Hlavní úkol aplikace bude tvorba reportů, vizualizace číselných hodnot na územní celek a tak možné porovnání jednotlivých oblastí.

3.1 Architektura GeoFiPu

Architektura systému GeoFip je tvořena několika bloky (viz obr.: 3.1).



Obr. 3.1: Architektura systému

- Uživatel FiPu – uživatel, který se na základě uživatelského jména a hesla dokáže přihlásit do systému.
- FiP – interní aplikace, do které je aplikace GeoFip integrována pomocí *iframe*.
- Autentizace – firemní autentizační služba vyvinutá firmou Partners Financial Services a. s. pro možnost zabezpečení aplikací, které nejsou přímo implementované v systému FiP (viz kap. 3.7).
- UI GeoFiP – uživatelské rozhraní aplikace GeoFip. (viz kap. 4.1)

- API Seznam.cz – rozhraní společnosti Seznam.cz pro přístup k mapovým podkladům a možnosti práce s mapou v rozsahu poskytovaných služeb API (viz kap. 3.6).
- Business Logic – jádro aplikace s logikou pro zpracování dat.
- Databáze – relační databáze, poskytující datové úložiště aplikace.
- ČSÚ – data poskytovaná pomocí definovaného výměnného formátu od Českého statistického úřadu (viz kap. 3.3.3).
- ČÚZK – data poskytovaná pomocí definovaného výměnného formátu od Českého úřadu zeměměřického a katastrálního (viz kap. 3.3.2).
- Partners – interní data společnosti Partners Financial Services a. s. (viz kap. 3.3.4).

3.2 Hlavní softwarové nástroje

- PhpStorm – vývojové prostředí pro jazyk PHP s podporou *front-end* jazyků (HTML, JavaScript, CSS). Prostředí má funkce ke správě databází, umí komunikovat s repositáři a synchronizovat soubory z lokálního úložiště na server.
- DataGrip – klientský nástroj pro správu databází. V rámci prostředí můžeme vytvářet databázové dotazy, pracovat s tabulkami databází a testování SQL databázových dotazů.
- Verzovací systém GitLab – decentralizovaný systém k uchovávání revizí kódů a vytváření záloh [27].

3.3 Zdroje dat pro mapové vrstvy

Aplikace GeoFiP využívá různá data pro zobrazení mapových vrstev, dle charakteru dat a definicí jednotlivých vrstev je pak rozhodnuto, jakým stylem se data zobrazí (bodově, liniově, polygonálně). Data se dělí dle zdroje a způsobu získání, v případě aplikace GeoFiP jsou zdroje ČÚZK, ČSÚ, Partners, kdy u prvních dvou institucí získání dat probíhá pomocí výměnných formátů dostupných na jejich webových stránkách. Jedním z úkolů bylo získání souřadnicových dat jednotlivých územních celků, přičemž tyto data poskytuje Český úřad zeměměřický a katastrální (viz kap. 3.3.2). Dalšími daty, která se dají navázat na území a mohou být zajímavá pro podnikatelské záměry firmy Partners Financial Services a. s. jsou data Českého statistického úřadu (viz kap. 3.3.3).

3.3.1 Provázání jednotlivých dat

Výše zmiňované instituce používají nekompatibilní označování územně správních celků, proto bylo nutno vytvořit společný, spojovací číselník těchto dat (tab. 3.1).

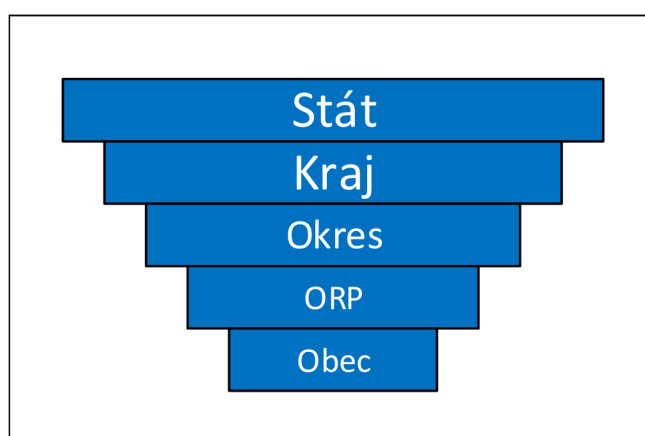
RSO_ORP	okres_kod	kraj_kod	VUSC_kod	id_okresu	RSO_okres	RSO_kraj	OBEC_KOD
1317	3702	37	116	54	40711	3115	582786
1325	3703	37	116	55	40720	3115	583120
1333	3703	37	116	55	40720	3115	583251

Tab. 3.1: Spojovací číselník

Číselník je tabulka, pro GeoFiP vytvořená v databázi, kde každý územní celek je navázán na vyšší správní celek (podle vydaného číselníku ČÚZK). GeoFiP navazuje data od úrovně obcí s rozšířenou působností na vyšší správní celky. Systém je možné rozšířit na jakékoliv územní celky, ke kterým je možné získat souřadnice. Identifikace správních územních celků patřících do vyššího územního celku je díky těmto číselníku velmi jednoduchá. Data ČSÚ mají sice svůj vlastní interní číselník, ale je k dispozici převodník, jenž umožňuje spojit tato data se správními celky ČÚZK. Data Partners Financial Services a. s. jsou navázána pomocí kódu adresního bodu, což představuje kód ČZÚK pro danou adresu. Převod adresy na kód je realizován pomocí algoritmu vyvinutého IT oddělením společnosti Partners Financial Services a. s.

3.3.2 Data z Českého úřadu zeměměřického a katastrálního

ČÚZK distribuuje svá data pro veřejnost pomocí registru územní identifikace, adres a nemovitostí (RÚIAN). RÚIAN je veřejný seznam umožňující uživatelům zdarma a bez registrace využívat data ČÚZK. Pomocí výměnného formátu RÚIAN (VFR) je možné stáhnout data například ve formátu *Extensible Markup Language* (XML), který byl pro aplikaci GeoFiP využit. Data jsou systematicky poskládaná dle návaznosti územních celků, jak je patrné z obrázku 3.2. Jednotlivé části jsou v rámci aplikace GeoFiP separátně odděleny v tabulkách v databázi.



Obr. 3.2: Zjednodušený model návaznosti správních celků

XML soubor je nutné tzv. „rozparsovat“, to v našem případě znamená, že z XML souboru jsou data převedena na datové objekty pomocí PHP skriptu. Z rozparsovaných dat se poté skládají řádky (záznamy) do databáze. Soubory XML obsahující informace pro jednotlivé územní celky mají vždy standardizovanou strukturu v podobě identifikátoru (kódu), názvu, platnosti, geometrie a další doplňující informace jako je například nadřazený územní celek [25].

3.3.3 Data z Českého statistického úřadu

Data ČSÚ jsou dostupná na webovém portále (www.czso.cz) pomocí jejich webové aplikace (obr.: 3.3) a jsou tříděna do kategorií, dle jejich charakteru. V rámci kategorie se data mohou dále větvit na podrobnější záznamy. V jednotlivých záznamech jsou informace na jakém územně správním celku jsou data získána, z jakého období jsou a jak často se data aktualizují.

Mzdy a náklady práce

Název tabulky	Období	Území
Počet zaměstnanců a průměrné hrubé měsíční mzdy	1998-2017	KR, Stát
Zaměstnanci a průměrné hrubé měsíční mzdy podle odvětví CZ-NACE	2000-2017	Stát
Zaměstnanci a průměrné hrubé měsíční mzdy podle sfér hospodářství	1998-2017	Stát
Průměrný evidenční počet zaměstnanců a průměrné hrubé měsíční mzdy podle odvětví CZ-NACE	2005-2016	KR, Stát
Zaměstnanci a hrubé měsíční mzdy - územní srovnání	2011-2017	KR, Stát
Průměrná hrubá měsíční mzda a medián mezd - mezikrajské srovnání	2011-2016	KR, Stát
Průměrné hrubé měsíční mzdy podle klasifikace zaměstnání - mezikrajské srovnání	2011-2016	KR, Stát
Podíly zaměstnanců, placený čas a hrubé měsíční mzdy podle hlavních tříd CZ-ISCO a pohlaví	2011-2016	KR, Stát
Podíly zaměstnanců, placený čas a hrubé měsíční mzdy podle vzdělání a pohlaví	2011-2016	Stát
Distribuce hrubých měsíčních mezd zaměstnanců podle hlavních tříd CZ-ISCO	2011-2016	Stát
Měsíční náklady práce	2009-2016	KR, Stát
Počet zaměstnanců a průměrné hrubé měsíční mzdy	2000-2017	KR, Stát

Obr. 3.3: Ukázka části webové rozhraní ČSÚ

Tato data je možné z webové stránky ČSÚ exportovat pomocí formátů XLSX, XML nebo PDF. Pro datový model aplikace GeoFiP bylo vhodné data vyexportovat pomocí XML. XML bylo „rozparsováno“ pomocí PHP skriptu a data následně uložena do databáze [26].

3.3.4 Data Partners Financial Services a. s.

Společnosti Partners Financial Services a. s. má data zobrazena v aplikaci GeoFiP uložena v databázi. Data jsou navázána na fyzickou osobu, která má mimo jiné v záznamech svoji adresu a svého finančního poradce. Díky adrese klienta a algoritmu vyvinutého IT oddělením společnosti je možné tuto adresu převést adresní místo dle definice ČÚZK, díky čemuž lze takto získané souřadnice navázat na příslušný územně správní celek. Návazností klienta, poradce a jeho kanceláře můžeme zjistit například kolik klientů má ve svém klientském kmeni poradce a odkud jsou, zda-li není vhodné například klienta předat jinému poradci, který je blíže jeho bydlišti.

3.3.5 Přidání dalších datových vrstev

Přidání dalších datových vrstev je možné díky modulárnosti systému pouhým vytvořením třídy pro tuto logickou skupinu. To je zajištěno tím, že pro každou logickou skupinu je definováno rozhraní, které tato třída musí implementovat a tím dojde ke snadné integraci do aplikace GeoFiP. Následuje přidání do ajaxového handleru (Výpis: 3.1), který má funkci takzvaného dispečera. Přes tento handler v různých částech prochází veškeré požadavky na vytvoření nebo smazání mapové vrstvy. Dle identifikátoru datové vrstvy se poté vytváří objekt pro tuto logickou datovou vrstvu, kde je následně řešen konkrétní mapová vrstva.

Výpis 3.1: Ukázka zdrojového kódu AJAX Handleru

```
if($data["idMenu"]=="partners") // zjistění jaký typ menu
{
/**
 * chování k partners vrstvam
 */
    $mapa = new PartnersDataRozcesti();//třída pro Partners data
    $JS .= $mapa->vytvorMapovyObjekt($data, $_POST,
        $poradiProSkaluZobrazeni);
}
if($data["idMenu"]=="csu")
{
    $mapa = new CsuDataRozcestnik();// třída pro ČSÚ data
    $JS .= $mapa->vytvorMapovyObjekt($data,
        $poradiProSkaluZobrazeni);
}
/**
 * tvorba objektů dle uzemního celku
 */
if($data["idMenu"]=="ruian")
{
    $spravniCeleg = new VytvorObjektyMapy();

    if($data["mapa"] == "Kraje soudržnosti") // kraje soudržnosti
    {
        $JS.=$spravniCeleg->vytvorObjekt("krajeS",NULL,$data["id"]);
    }
    elseif ($data["mapa"]=="Kraje") // kraje
    {
        $JS.=$spravniCeleg->vytvorObjekt("kraje",NULL,
            $data["id"]);
    }
    else // 0+ORP
    {
        preg_match("/^(\\w+)_([\\d,\\w]+)$/",
            $data["mapa"], $typMapy);
        $JS.=$spravniCeleg->vytvorObjekt($typMapy[1],
            $typMapy[2], $data["id"]);
    }
}
}
```

3.4 Třídy a metody aplikace GeoFiP

Prováděcí logika systému je psána v jazyce PHP. Pro účely bakalářské práce byly vytvořené metody pro zobrazení základní podkladové mapy. Model je tvořen třídou *Mapa* (viz výpis: 3.2) využívající metody pro nastavení typu souřadnic (JTSK, WGS84, UTM,...), identifikační parametr mapy, ke které chceme přistupovat (Výpis: 3.3) nebo například zdali chceme, aby si uživatel mohl měnit typ podkladu (základní mapa, ortofoto, sateltní,...). Po nastavení objektu *Mapa* se na tento objekt aplikuje metoda *vytvorJavascriptProMapu()*, která vytvoří z parametrizovaného objektu výsledný JavaScript pro API mapy Seznamu, který je možné vložit do webové aplikace a tím zobrazit mapu. Tato třída je obecná a velmi lehce, tak lze změnit na jiný typ API, jen dopsáním metod s jinými JavaScriptovými kódy, například od společnosti Google. Obdobným procesem se budou tvořit objekty pro další prvky k zobrazení do mapy, ať už se jedná o bodové, liniové a nebo polygonální elementy.

Výpis 3.2: Ukázka zdrojového kódu PHP pro vytvoření objektu mapy

```
$Mapa = Mapa::create('k')//parametr pro mapu (DIV)
->NastavCentrovani(false) //centrování mapy na prvky v ni
->NastavOvladaniMapy(true) //ovládání mapy +/-, posunování
->NastavTypSouradnic('JTSK') //typ souřadnic, které vkládáme
->NastavZmenuPodkladuMapy(true)
// možnost uživatele měnit mapový podklad
->NastavUkazatelMysi('hand'); //typ ukazatele myši
$vrstva_0 = Vrstva::create()
->NastavVrstvuDoMapy('polygonLayer_0')
->NastavTypVrstvy(Vrstva::$VRSTVA_GEOMETRICKA)
->DataZMapy($Mapa);
```

- `Mapa::create()` – Statická metoda vytvářející instance objektu mapy.
- `NastavCentrovani()` – Metoda nastavující parametr, zda se bude mapa po načtení centrovat dle zobrazených prvků.
- `NastavOvladaniMapy()` – Metoda nastavující přidání ovládacích prvků do mapy.
- `NastavTypSouradnic()` – Metoda nastavující typ souřadnic (JTSK, WGS, UTM).
- `NastavZmenuPodkladuMapy()` – Metoda nastavující přidání funkce pro změnu mapového podkladu (normální mapa, letecká, turistická, ...).
- `NastavUkazatelMysi()` – Metoda nastavující vizuální zobrazení ukazatele myši (pointer, hand, ...).
- `VytvorJavascripProMapu()` – Metoda vytvářející JavaScript pro aplikační rozhraní `Mapy.cz`, dle objektu nastaveného v PHP.

- `Vrstva::create()` – vytváří instanci objektu pro mapovou vrstvu.
- `NastavVrstvuDoMapy()` – přidá mapovou vrstvu s názvem.
- `NastavTypVrstvy()` – nastaví typ vrstvy, zda se jedná o vrstvu geometrickou a nebo bodovou.
- `DataZMapy($Mapa)` – každá vrstva musí mít data z Mapy z důvodu nastavení parametru `divu`

3.4.1 Objektový model aplikace GeoFip v PHP

Aplikace GeoFiP dodržuje standardy objektově orientovaného programování, aplikační logika je tedy zapouzdřena v odpovídajících třídách. Veškeré níže zmiňované třídy se jmenují stejně jako soubory, ve kterých jsou umístěny.

Základní třída pro mapu je umístěna v souboru *Mapa.php* s názvem *Mapa*. Zde se vytváří objekt mapy s nastavením základních parametrů, jak má mapový podklad vypadat, zejména se jedná o nastavení názvu HTML prvku, ve kterém je mapa na *front-endu* zobrazena, možnosti ovládání mapy a jiných obecných parametrů. Jednotlivé vrstvy, do kterých jsou pak vkládány mapové prvky, se tvoří pomocí třídy umístěné v souboru *Vrstva.php*. V této třídě jsou zejména metody pro nastavení názvu mapové vrstvy a typu prvků v ní obsažené, tedy liniová (geometrická) nebo bodová (markerová) vrstva.

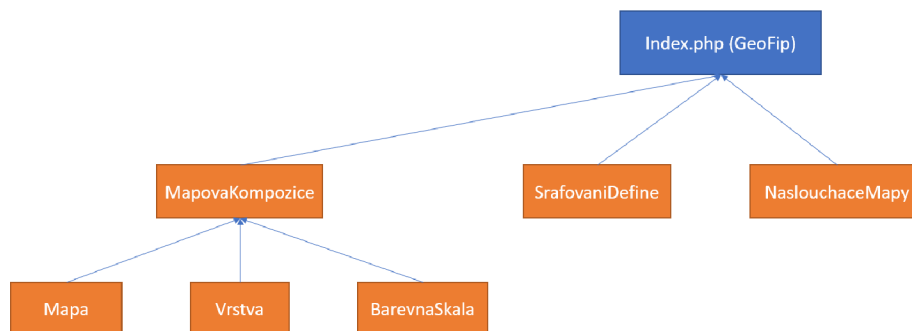
Tvorba barevné škály zabezpečuje třída nacházející se v souboru *BarevnaSkala.php*, která kromě nastavení legendy, výpočtu počtu prvků legendy (škála se tvoří dynamicky na základě zobrazovaných dat), nastavení barevnosti a vygenerování výsledné HTML tabulky, zajišťuje získání HTML kódu barvy pro jednotlivé mapové prvky. Pokud se jedná o geometrické prvky, tak i barvu vnitřní výplně a barvu okrajové hrany (pokud je zobrazována).

Šrafování polygonů se vytváří pomocí třídy *SrafovaniDefine.php*. Tato třída definuje šrafy, vytváří SVG kód a následně vrací tento kód do aplikace. V aplikaci GeoFiP jsou definované dva typy šraf, a to vodorovná a svislá. Každý typ šrafy má svou vlastní definici (šablonu), která je složena z barvy, hustoty šrafy a orientace (vertikální, horizontální). Šrafy mají svůj identifikátor na základě, kterého je aplikace schopna přiřadit správnou šrafu správnému mapovému prvku.

Třída obsažená v souboru *NaslouchaceMapy.php* vytváří nasloucháč pro mapu, tedy mechanismus, který sleduje definované uživatelské události nad mapu, například po kliknutí na prvek v mapě nám vrátí určité informace o něm a na základě nich můžeme dále pracovat s těmito údaji.

Každý mapový prvek, se kterým je možnost v rámci aplikace pracovat musí být známý třídě *MapovaKompozice* v souboru *MapovaKompozice.php*. Tato třída řeší ukládání do databáze nezbytně nutných dat pro mapovou kompozici (identifikátor

v jaké vrstvě se prvky nachází nebo které prvky se ve vrstvě nachází). Pro zobrazení podkladové mapy (převzaté z mapy.cz) stačí pouze třída *Mapa.php*, veškeré další proprietární prvky musí být ve zmíněné třídě *MapovaKompozice* (či nějakém jejím potomku). Atributy třídy jsou definované, tak že pokud nejsou nastavené parametry pro jednotlivé prvky, tak se použije defaultní nastavení.



Obr. 3.4: Ukázka architektury vytváření mapy

Hierarchie tříd mapových prvků zobrazených v mapových vrstev je následující(3.5). Rodičovská třída se nazývá *MapovyPrvek*. V této třídě se definují základní parametry pro všechny prvky, například ID – identifikátor prvku, přiřazení objektu vrstvy k prvkům mapy a nastavení možných vizitek u zobrazeného elementu.

Tato třída má dva potomky, a to dle toho jaký je typ zobrazovaného prvku. Pro body (markery) se využívá třída *JednySouradnice*, obsahující právě jednu dvojici souřadnic X a Y, včetně jejich typu (JTSK, WGS64 atd.). Pokud se jedná o liniové (geometrické) prvky je využita třída *OptionsGeometry*, kde se nastavují obecné parametry liniových / polygonálních prvků, a to zejména barva, šířka ohraničení, styl spojení jednotlivých souřadnic nebo průhlednost výplně. Z této třídy je zděděná třída *PoleSouradnic*, kde se definuje opět typ souřadnic a pole, ze kterého jsou poté vyčítány souřadnice mapového prvku.

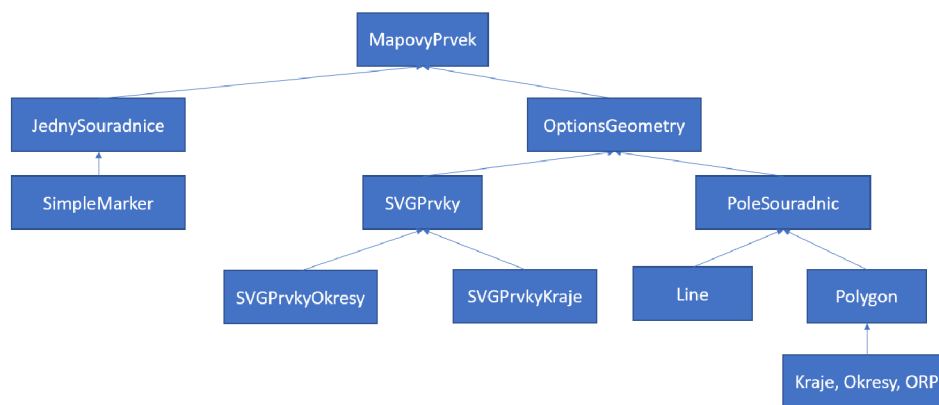
Z třídy *PoleSouradnic* se dědí třídy pro vytváření prvků čáry (*Line*) a polygon (*Polygon*). Tyto třídy vytváří geometrické prvky do mapy a zabezpečují vytvoření správných JavaScriptových kódů. Aplikace GeoFiP ve většině případů využívá třídu *Polygon*, jelikož se z ní dědí jednotlivé územně správní cely (kraje, okresy, obce s rozšířenou působností). Tyto třídy mají za úkol načíst souřadnice z databáze a transformovat je do příslušné podoby.

Tyto třídy mají metodu pro výběr přesnosti souřadnic, jelikož aplikace GeoFiP umí využívat souřadnice JTSK (Jednotná trigonometrická síť katastrální), které se

aplikují pouze na oblast ČR a Slovenska. Tento souřadnicový systém uplatňuje pravidlo, že na tak malém území se neprojeví zakřivení Země, a proto můžeme říct, že se jedná o pravoúhlou souřadnou síť. Díky tomu jsme schopni pomocí jednoduchých matematických operací spočítat a vybrat souřadnice, které mají určitou vzdálenost od sebe. Pro aplikaci GeoFiP byla zvolena defaultní přesnost 50 metrů, to znamená, že souřadnice polygonů mají mezi následnými body minimálně 50 metrů. Pole souřadnice není součástí dat, které se ukládají do databáze v rámci objektu Mapová kompozice.

Během vývoje aplikace bylo zjištěno, že některé územně správní celky jsou tzv. vnořené. Jedná se například o kraj Praha a Středočeský kraj. Zde nastává situace, kdy jeden kraj pohlcuje druhý. V tomto případě pokud by systém vykresloval oba kraje zároveň jejich zobrazení by se překrývalo, a tím se způsobila nepřehlednost a snížila by se rozeznávací schopnost. Proto bylo nutné najít způsob tzv. vyříznutí těchto dvou prvků, a tedy se seznámit s *Scalable Vector Graphics* (SVG). Díky SVG je možné vytvořit vyříznutí jednoho prvku v druhém.

V rámci logiky systému se muselo zabezpečit, pokud se vykreslují tyto prvky, tak aby se nevykreslovaly klasickou metodou, ale byly vytvořeny pomocí SVG. Z Třídy *SVG* vychází třídy *SVGPrvkyKraje*, *SVGPrvkyOkresy* a *ORP*. Tyto třídy řeší problémy tzv. vykrajování prvků, které jsou vnořené. Po nadefinování objektu mapový prvek se na tento objekt zavolá metoda *VytvorJS*, která z nastavovacího objektu vytvoří JavaScriptový kód, který je následně odeslán do API Mapy.cz [29].



Obr. 3.5: Nákres architektury pro vytváření mapového prvku

3.4.2 Objekty v JavaScriptu aplikace GeoFiP

Aplikace GeoFiP využívá JavaScript pro tvorbu záznamů nastavení. Hlavním nastavovacím objektem je objekt nazvaný *vrstva_x*, kde x je číslo 0-3. Tyto objekty mají za úkol mít nastavené parametry pro jednotlivé mapové vrstvy, které se aktuálně dají zobrazit. Zobrazení mapových vrstev je omezeno na hodnotu 4, z důvodu možné rozlišovací schopnosti v mapě. V objektu *vrstva_x* jsou informace o datech nastavených uživatelem, a to o jakou mapu se jedná, na jaké území se má mapa zobrazit, přičemž probíhá kontrola, na které územní celky je možné vrstvu zobrazit z hlediska dat v databázi, hodnotu vertikální struktury řízení (tz. jaká data se můžou uživateli zobrazit). Dalšími parametry jsou barva a její průhlednost nebo typ šrafy. V objektu je obsažena proměnná, která určuje, zda mapa již byla zobrazena anebo jestli se změnilo její nastavení. Tento objekt pro každou zobrazenou mapovou vrstvu je odesílán s aktualizací mapy.

```
> vrstva_0
< {idMenu: "partners", id: 0, mapa: "BJ na portfolio", userID: 100000, skala: "Modrá", ...}
  id: 0
  idMenu: "partners"
  iduzemi: 3115
  mapa: "BJ na portfolio"
  ▶ mozneZobrazeni: {kraje: true, okresy: true, ORP: true}
  pruhlednost: 100
  skala: "Modrá"
  srafura1: true
  srafura2: false
  userID: 100000
  uzemi: "ORP"
  zobrazena: true
```

Obr. 3.6: Ukázka JavaScriptového objektu vrstva

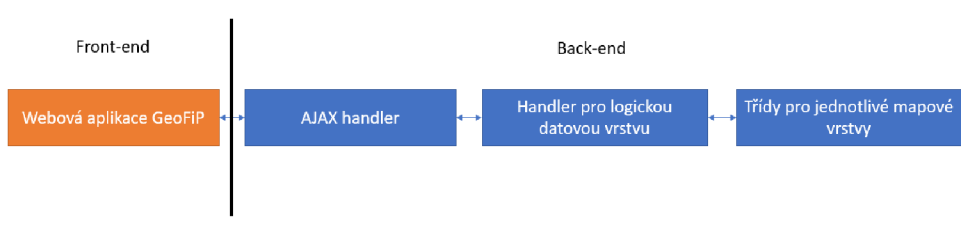
3.5 Vytváření mapové kompozice

Vytváření mapové kompozice probíhá v několika krocích. První krok je výběr uživatele, jakou mapovou vrstvu chce zobrazit. Po výběru mapové vrstvy dostane mapová vrstva předdefinované vlastnosti, zejména se jedná o nastavení územního celku a barvu výplně. Uživatel má možnost tyto předdefinované vlastnosti změnit v nastavovacím menu, kde si mapovou vrstvu může upravit dle vlastních požadavků. Po uložení nastavení a následném požadavku na aktualizaci mapy se javascriptové nastavovací objekty odešlou společně s dalšími parametry jako je například zobrazovaná vertikální struktura řízení pomocí technologie AJAX.

V ajaxovém handleru se zkontroluje, zda některá z mapových vrstev již nebyla zobrazena anebo je bez změny. Tato kontrola pracuje na principu, kontroly předešlého požadavku a porovnání s aktuálně přichozím požadavkem. Předešlý požadavek je uložený v databázi pod svým unikátním identifikátorem, který je tvořený názvem mapové aplikace a unikátního identifikátoru uživatele. Pokud se jedná o změnovou mapu musí se nejprve odstranit stará mapová vrstva. Po kontrole se přechází na rozhodnutí, jaká instance třídy se má vytvořit z hlediska logického členění datových vrstev (Partners data, ČSÚ data, ČÚZK data). Po správném zařazení je vytvořena instance třídy pro daný logický datový člen, který slouží jako další handler, dle toho, jakou mapovou vrstvu chceme zobrazit.

V tomto handleru se nachází již definované třídy pro jednotlivé uživatelsky vybrané mapové vrstvy. Ve kterých probíhá vyčtení dat pro konkrétní mapovou vrstvu dále se zde řeší vytvoření instance třídy pro územně správní celek a tvorba barevné škály, popřípadě zjištění dalších nutných informací k možnému zobrazení mapové vrstvy.

Po získání dat pro mapovou vrstvu, vytvoření správné barevné škály a instance pro územní celek se na tento objekt volá metoda `vratJSCelku`. Tato metoda vytváří z výše definovaných parametrů výsledný javascriptový kód pro aplikaci používané API mapy, který je následně vrácen přes všechny třídy zpět pomocí AJAXu do webové aplikace a ta díky API mapy.cz se zobrazí jako výsledná požadovaná mapová kompozice.



Obr. 3.7: Nákres posloupnosti při vytváření mapové vrstvy

3.6 Aplikační rozhraní Mapy firmy Seznam.cz

Firma Seznam.cz, a. s. provozuje aplikační rozhraní pro Mapy.cz umožňující uživatelům využívat jejich mapové podklady na vlastních webových stránkách a v rámci aplikací uživatelů. Poskytuje mapové služby, funkce v podobě mapových a obrazových podkladů a dává pro ně k dispozici uživatelské rozhraní.

Jeden z důvodů výběru této služby byl, že neomezuje počet dotazů na webový portál. Aktuální verze služby je *4.13 – Neil Armstrong* (k datu 25.11.2017). Použití je zcela zdarma a je možné ji využít i pro komerční účely, pokud jsou dodrženy licenční podmínky.

Funkčnost API Mapy zaručuje inicializace v hlavičce souboru (Výpis: 3.3) a dále pracují na principu vykonávání JavaScriptového kódu, kdy k příslušnému webovému elementu, pro GeoFiP bylo vybráno ID elementu „k“, vkládáme jejich JavaScriptové kódy obohacené o naše parametry (obr.: 3.4). Jedná se zejména o vizuální podobu zobrazovaných dat (např. šířka čar, barevnost, výplně polygonů, možnost nastavení vlastních bodových značek).

Výpis 3.3: Inicializace Loaderu pro API Mapy

```
<script
  type='text/javascript'
  src='https://api.mapy.cz/loader.js'>
</script>
<script type='text/javascript'>Loader.load()</script>
```

Výpis 3.4: Ukázka JavaScriptového kódu pro zobrazení mapy

```
var center = SMap.Coords.fromJTSK(1095790.58,695189.93);
var k = new SMap(JAK.gel("k"), center,8);
k.addDefaultLayer(SMap.DEF_BASE).enable();
  k.addDefaultLayer(SMap.DEF_OPHOTO);
  k.addDefaultLayer(SMap.DEF_OPHOTO0203);
  k.addDefaultLayer(SMap.DEF_OPHOTO0406);
  k.addDefaultLayer(SMap.DEF_TURIST);
  k.addDefaultLayer(SMap.DEF_HISTORIC);

var layerSwitch = new SMap.Control.Layer();
  layerSwitch.addDefaultLayer(SMap.DEF_BASE);
  layerSwitch.addDefaultLayer(SMap.DEF_OPHOTO);
  layerSwitch.addDefaultLayer(SMap.DEF_TURIST);
  layerSwitch.addDefaultLayer(SMap.DEF_OPHOTO0406);
  layerSwitch.addDefaultLayer(SMap.DEF_OPHOTO0203);
  layerSwitch.addDefaultLayer(SMap.DEF_HISTORIC);
  k.addControl(layerSwitch, {left:"8px", top:"9px"});

k.addDefaultControls();
k.setCursor("hand");
k.setPadding("left",5);
k.setPadding("right",85);
k.setPadding("bottom",20);
k.setPadding("top",5);
```

Na výpisu 3.4 je část JS kódu, který ovlivňuje následující vlastnosti mapy:

- Centrování základního zobrazení mapy na souřadnice a velikost přiblížení mapy s nastavením identifikátorem mapy „k“.
- Nastavení výběru mapových podkladů (základní mapa, letecká, turistická, historická).
- Nastavení definovaných mapových podkladů do možnosti výběru těchto map a přidání ovládání mapy.
- Příkazy set nastavují parametry přímo mapového divu. Jedná se o nastavení kurzoru a to na hodnotu „hand“, což je ukazatel ve vizuální podobě ruky. Parametry „padding“ nastavují šířku vnitřního okraje prvku.

3.7 Integrace a zabezpečení aplikace GeoFiP

GeoFiP je integrován do interního firemního systému FiP. Systém FiP je webová aplikace postavená na technologii C# ASP.NET. Integrace aplikace GeoFiP bude probíhat vložení *iframu* do určené části systému FiP.

K aplikaci FiP je vyžadován autorizovaný přístup pomocí uživatelského jména, hesla a v některých případech kódem, který je uživateli odeslán na jeho mobilní telefon. Firemní bezpečnostní politika neumožňuje propojit s interním autentizačním systémem aplikace třetích stran (např. i aplikaci GeoFiP). Aby se nemusel uživatel do aplikace GeoFiP přihlašovat dalším způsobem (např. BASIC autentizací na straně webového serveru), je pro autentizaci uživatele implementována autentizační služba vyvinuta společností Partners Financial Services a. s. Služba umožňuje sdílení přihlášení s webovou aplikací FiP a využívá tedy interní autentizační procesy.

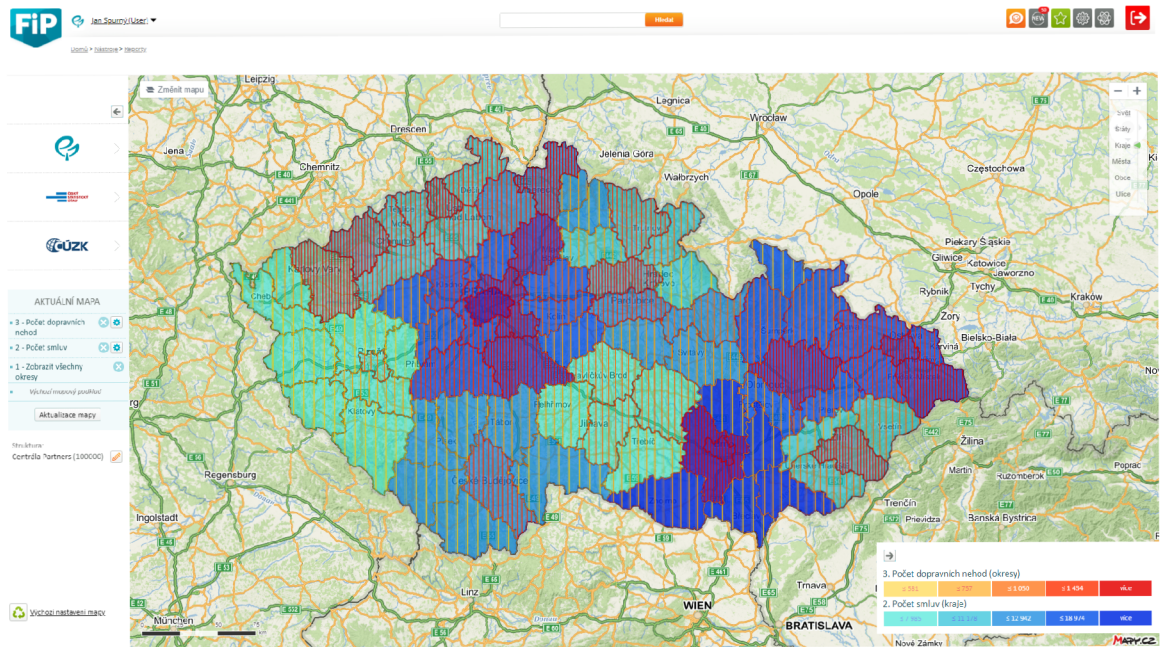
Přihlášení do aplikace GIS postupuje následovně:

1. Autentizace zkontroluje, zda má přihlášený uživatel v aplikaci FiP přístup k aplikaci GeoFiP.
2. Pokud uživatel disponuje právy k zobrazení aplikace GeoFiP je vygenerován tzv. autentizační token.
3. Autentizační token je předán metodou GET v URL adrese iFramu.
4. Aplikace GeoFiP při spuštění ověřuje tento autentizační token, zda je validní.
5. Při validitě autentizačního tokenu služba vrací identifikaci uživatele, kterou dále aplikace GeoFiP využívá, v opačném případě je vypsána zpráva o neoprávněném přístupu.

Expirace tokenu je volena tak, že při prvotní autentizaci je expirace krátká, z důvodu bezpečnosti a při následných opětovných autentizacích, kdy se uživatel dotazuje na nový obsah mapy, se prodlužuje. Autentizační služba musí být z bezpečnostních důvodů volána ve všech místech, kde je nutné ověřit, jaký uživatel je přihlášený. Tento způsob autentizace není sice nejbezpečnějším způsobem ověřování uživatelů, ale vedením společnosti Partners Financial Services a. s. byl přijat jako vyhovující.

3.8 Interní systém FiP

FiP je interní systém společnosti Partners Financial Services a.s. , který je určen především pro její poradenskou síť. Jedná se o komplexní nástroj pro správu veškerých poskytovaných služeb pro klienty firmy.



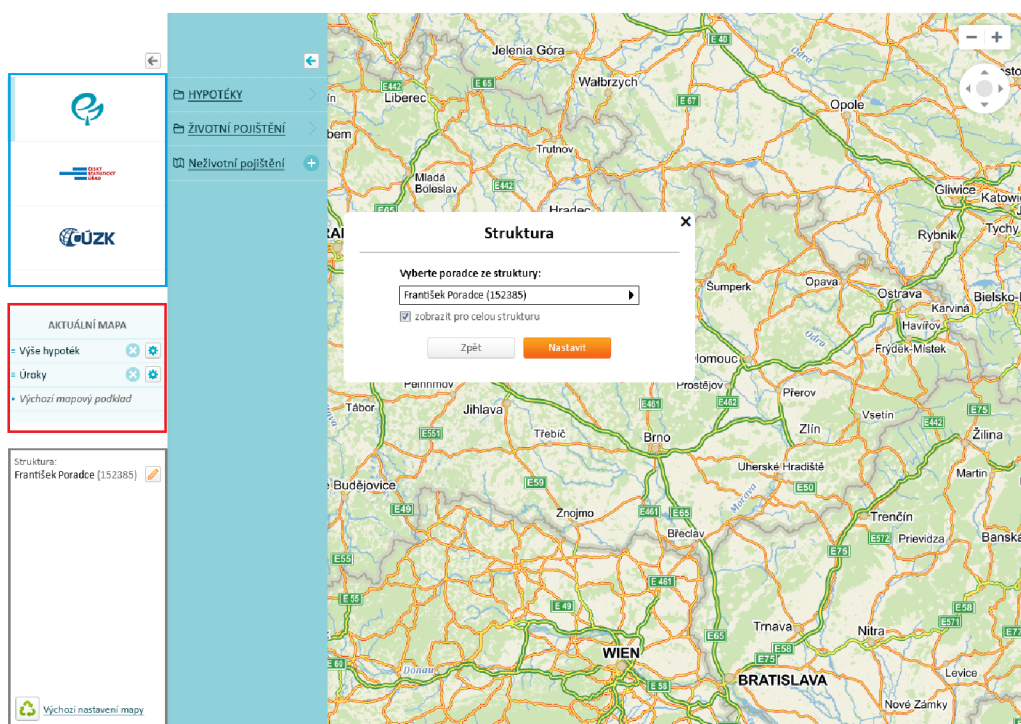
Obr. 3.8: Ukázka budoucí integrace aplikace GeoFiP do interního systému FiP

4 GRAFICKÁ ČÁST UŽIVATELSKÉHO ROZHRAŇÍ

Aplikace GeoFiP je zejména pro vytváření reportů, a vizualizaci dat do mapové kompozice. Proto bylo nutné vyřešit i grafické zobrazení menu, práci s menu a možné nastavení uživateli, dle vlastního uvážení, barevné zobrazení mapové kompozice s možností výběru pořadí jednotlivých mapových vrstev.

4.1 Návrh grafického rozhraní GeoFiP

Grafický návrh aplikace GeoFiP kombinuje několik funkcionalit, postupem vývoje prošel i změnami v podobě přidání některých grafických prvků, které mají uživateli usnadnit orientaci. Hlavním požadavkem ovládání je, aby byly všechny potřebné prvky soustředěny na jednom místě. Ovládání sdružuje jak prvky měnící se dynamicky (modrý rámeček na obrázku 4.1), ovládání aktuální mapové kompozice (červený rámeček na obrázku 4.1) výběr zobrazovaných dat dle vertikální řídicí struktury a obecná nastavení (šedý rámeček na obrázku 4.1).



Obr. 4.1: Návrh uživatelského prostředí

4.2 Komponenty menu

V dynamické části datového menu (modrý rámeček na obrázku 4.1) se po označení příslušné položky menu otevře další nabídka, kde jsou systematicky rozděleny jednotlivé tématické vrstvy. Z důvodu, že některé součásti jsou příliš rozsáhlé a komplexní, jsou zahrnuty pod jednou položkou, která po svém označení otevírá další menu.

Přidání mapové vrstvy do zobrazení je možno provést po výběru symbolu „+“ (znaménko plus) nebo kliknout na název. Po výběru z nabídky je vrstva přidána do pole „AKTUÁLNÍ MAPA“, kde se zobrazí vybrané vrstvy a je možné je upravit. Každá mapová vrstva má definované implicitní zobrazení, které si uživatel může změnit. K tomu se využívá symbol „ozubeného kolečka“. Po jeho výběru se otevře menu s možnostmi nastavení vizuálního zobrazení vrstvy.

Lze nastavit:

- Barevnou výplň – vrstva bude vyplněná pomocí přednastavené barvy s odpovídající barevnou škálou, která se bude zobrazovat v pravé části mapy s popisem hodnot,
- Šrafování – vrstva bude vyplněna šrafováními u nichž lze zvolit barevná škála, a tak například zvýraznit jeden typ sledovaného prvku.

Další možností pro zobrazení je výběr dat podle územně správního celku, kdy ve výběru jsou vždy všechny územně správní celky, na které je možné data zobrazit.

Vrstvu lze ze zobrazení odstranit výběrem symbolu „x“ (křížek) u daného popisu mapové vrstvy.

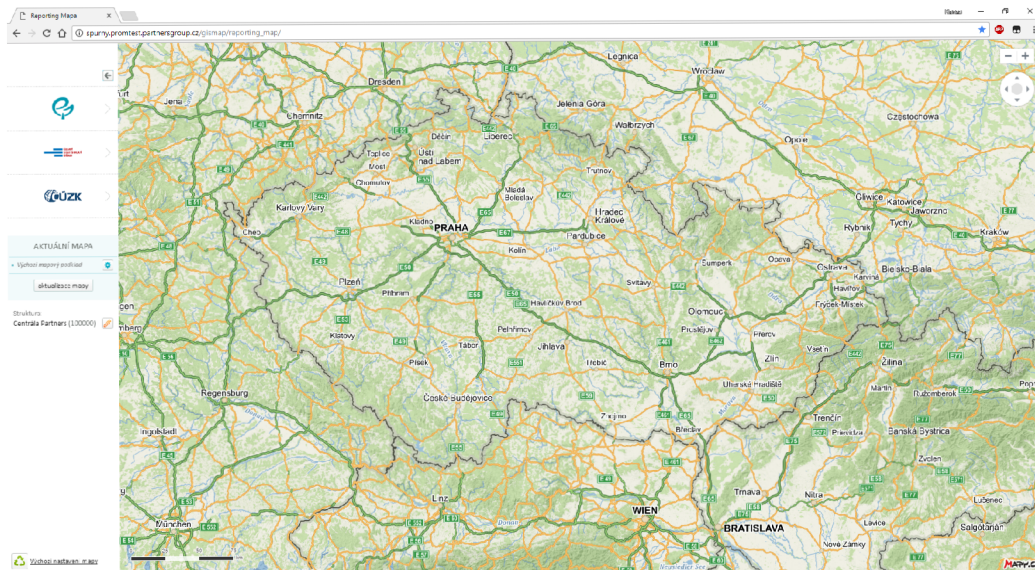
Volba „Výchozí mapový podklad“ určí, nad jakým typem mapového podkladu chceme data zobrazit (mapa obecná, letecká, satelitní).

Výběr organizační struktury, pro kterou se data mají zobrazit, se provede pomocí symbolu tužky v sekci „Struktura“ v šedém rámečku na obrázku 4.1. Volaná komponenta je tvořena nástavbou jQuery UI. Při prvotním načtení aplikace je zjištěno, k jakým datům společnosti má přihlášený uživatel práva přistupovat a dle jeho přístupových práv jsou vyselektována příslušná data pro zobrazování do mapové vrstvy. Tato funkce je realizována pomocí PHP třídy, která řeší příslušné dotazy do databáze, mimo to má každý uživatel možnost si zobrazit agregovaná data pro celou společnost.

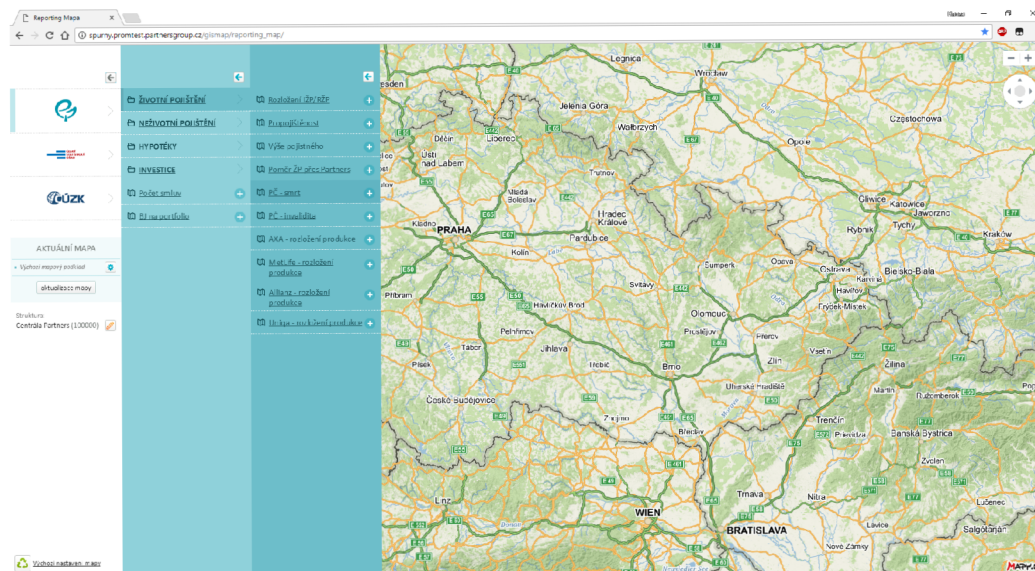
„Výchozí nastavení mapy“ uživatele vrátí k implicitnímu nastavení zobrazení (počáteční podoba) provede to obnovením (*refresh*) webové aplikace.

4.3 Realizace aplikace GeoFiP

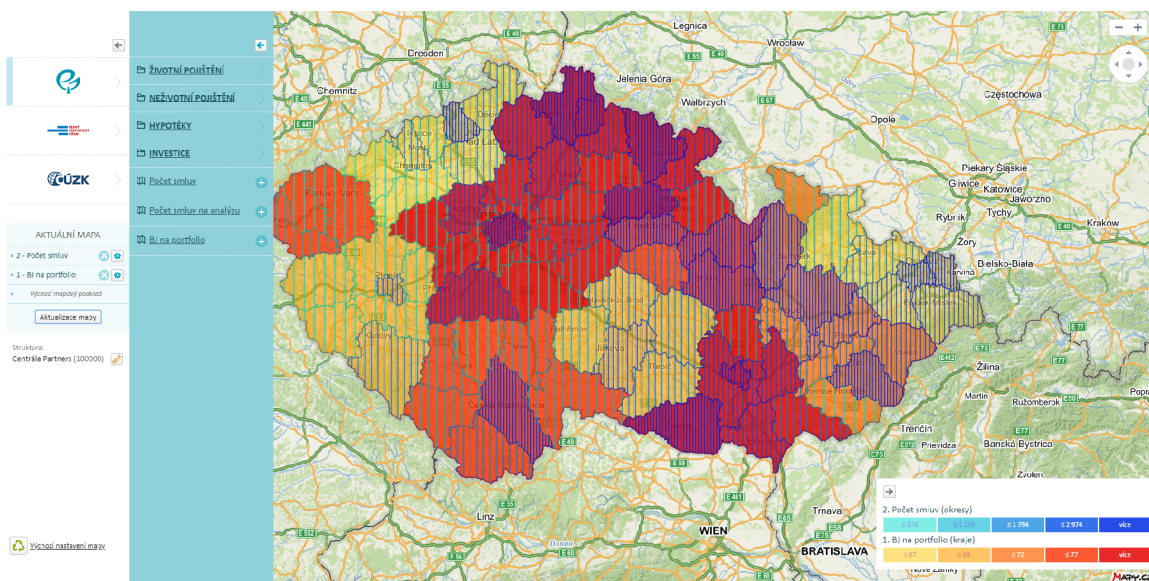
Výsledná realizace aplikace GeoFiP byla provedena dle výše uvedené kapitoly (viz. kap.: 3). Ukázky realizace jednotlivých částí aplikace můžete vidět níže na obrázcích 4.2, 4.3, 4.4, 4.5, 4.6, 4.7, 4.8, 4.9.



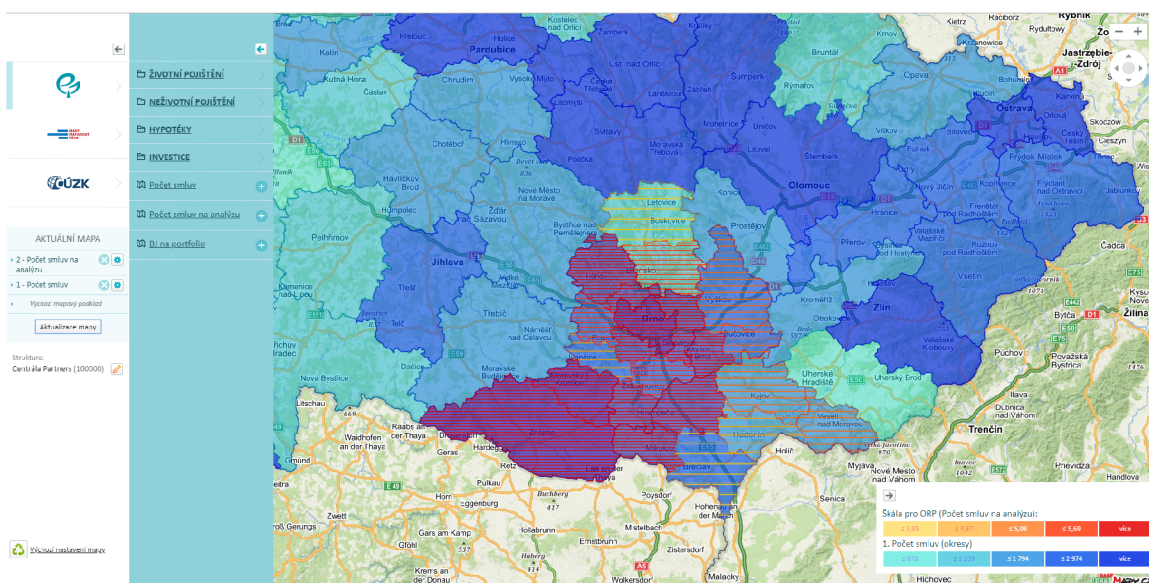
Obr. 4.2: Základní zobrazení menu při načtení stránky



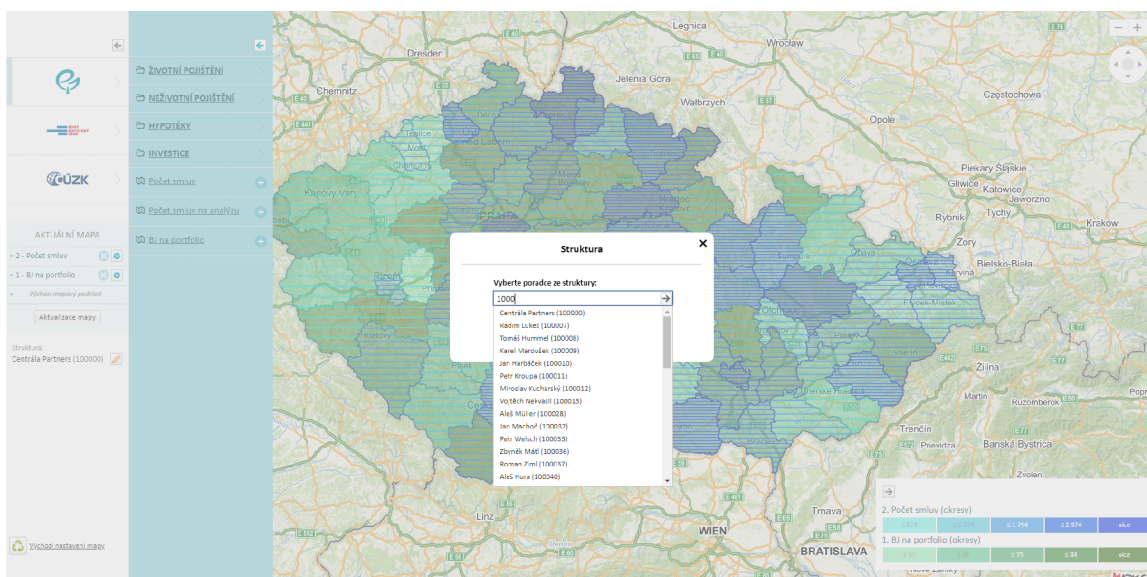
Obr. 4.3: Otevřené menu výběru mapových vrstev



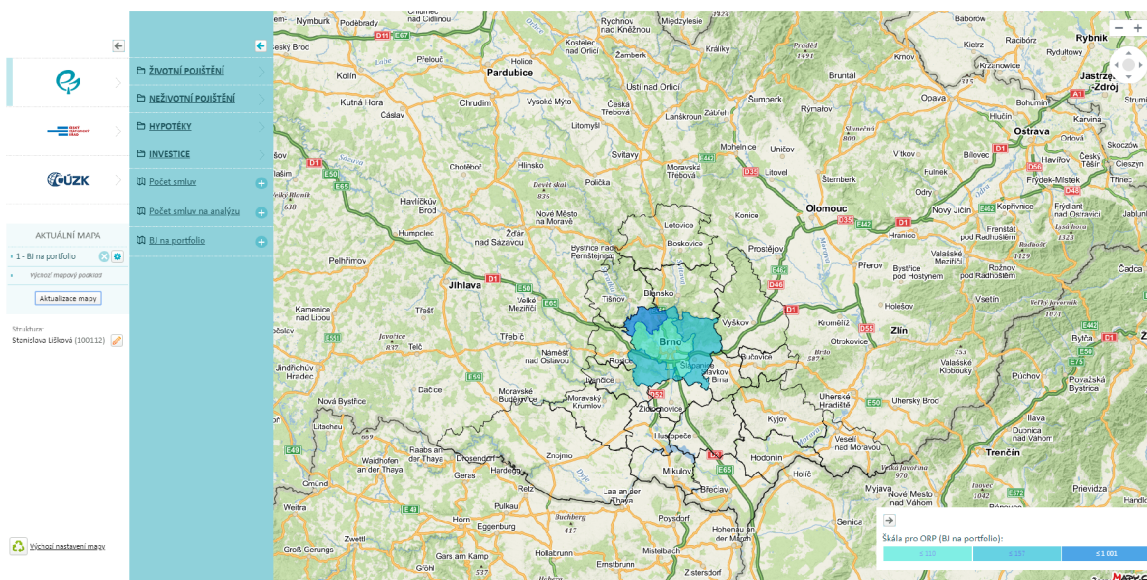
Obr. 4.4: Příklad připnutí vrstvy do menu k zobrazení a vykreslení barevné škály



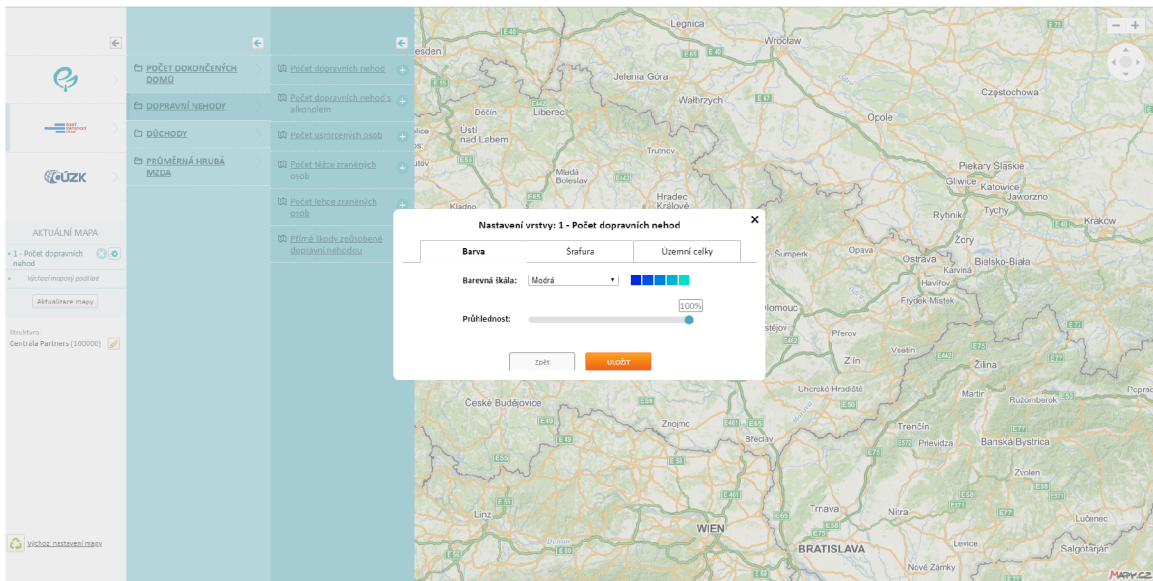
Obr. 4.5: Zobrazení dat na různé územní celky



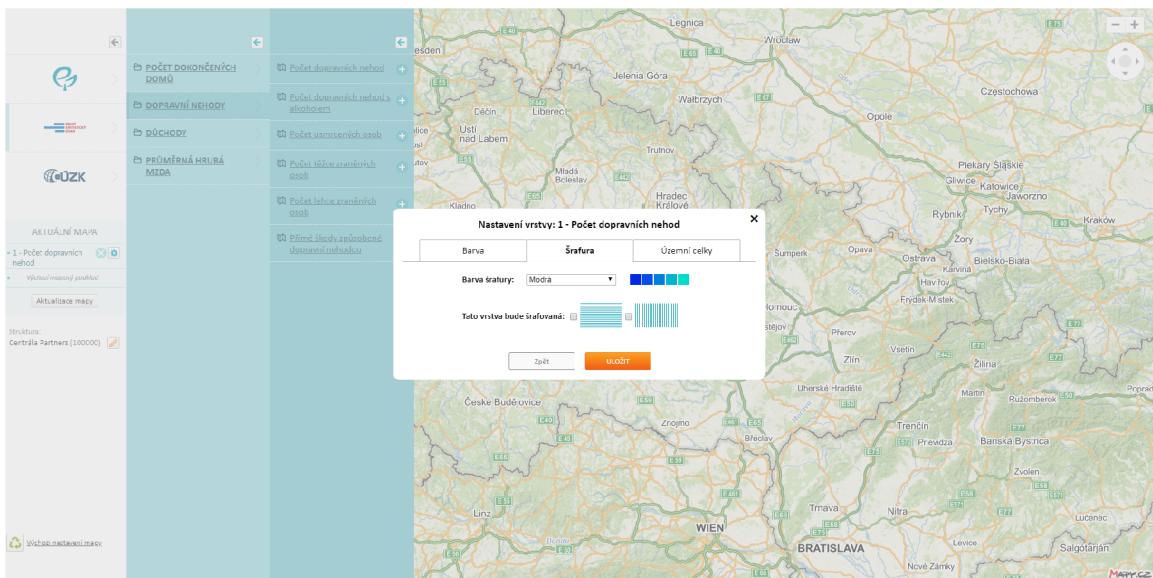
Obr. 4.6: Výběr dat k zobrazení podle vybrané vertikální struktury společnosti



Obr. 4.7: Zobrazení dat pro určitého uživatele



Obr. 4.8: Nastavení barevného zobrazení vrstvy



Obr. 4.9: Nastavení šrafovaného zobrazení vrstvy

5 ZÁVĚR

V rámci bakalářské práce byla nastudována problematika získání, transformace a spojení dat institucí (ČSÚ, ČÚZK, Partners Financial Services a. s.), jejichž data jsou využita. Dále byla nastudována možnost zabezpečení a integrace aplikace GeoFiP do interní aplikace FiP, využití API pro mapy od firmy Seznam.cz a byl jsem seznámen s firemní politikou zobrazování dat dle vertikálního řízení společnosti.

Pro aplikaci GeoFiP byl navrhnout a vytvořen datový model, který využívá relační databázi MySQL. Data (ČÚZK a ČSÚ) byla získána a uložena do výše zmiňované databáze, kde bylo nutno vytvořit převodník pro jednotlivé kódové označení dat od různých institucí a tím zabezpečit možnosti navázání jednotlivých dat na sebe.

Proběhl proces seznámení a nastudování API systému map od firmy Seznam.cz a byla vytvořena třída pro zobrazení základní podkladové mapy do aplikace. Dále byly zjištěny možnosti zobrazení bodových, liniových, polygonálních i prvků nadefinovaných pomocí SVG s využitím API.

Během bakalářské práce bylo vytvořeno grafické rozhraní (UI) aplikace GeoFiP, které má definované funkcionality, dle požadavků zadávající firmy Partners. Byla navrhnutá architektura systému GeoFiP a následný vývoj aplikace. V grafickém rozhraní byla vytvořena logika pro třídění dat do jednotlivých menu, funkce pro přidávání a odstraňování mapových vrstev z menu v aktuálně zobrazené mapové kompozici, obslužné funkce pro správnou funkcionalitu menu v podobě otevírání, zavírání nebo grafické úpravy menu.

Obslužné funkce pro menu jsou tvořeny JavaScriptem s využíváním technologie AJAX pro komunikaci s PHP obslužnými třídami.

Byly vytvořeny třídy a metody pro přidávání i odstraňování mapových prvků do mapy (body, linie, polygony), vytvoření způsobu přidávání nových datových vrstev a přidávání jednotlivých mapových vrstev bez nutné změny v systému.

Dále byla implementována metoda způsobu vytváření dynamické barevné škály na základě aktuálně požadovaných dat s rozprostřením, aby každý záznam v legendě měl stejný počet zobrazených prvků.

V rámci bakalářské práce jsou zprovozněny všechny mapové vrstvy pro ČÚZK, tři mapové vrstvy ze skupiny Partners Financial Services a. s. dat a jedna mapová vrstva z ČSÚ.

V systému GeoFiP je definovaný standard přidávání dalších mapových vrstev. Tento standard implementuje definované rozhraní, jakým způsobem mají vypadat třídy. Díky tomuto rozhraní je možné snadno rozšiřovat mapové vrstvy bez dalších znalostí systému. Mapové vrstvy se přidávají vytvořením třídy a přidáním třídy do příslušného logického handleru. Každá mapová vrstva tak má vlastní třídu pro vytváření dotazů do relační databáze na požadovaná data.

LITERATURA

- [1] Multiplatformní PHP. *Linuxsoft* [online]. 2005 [cit. 2017-10-21]. Dostupné z: <http://www.linuxsoft.cz/article.php?id_article=629>
- [2] KOSEK, Jiří. Přehled podpory XML v PHP5. *Zdrojak* [online]. 2009 [cit. 2017-10-21]. Dostupné z: <<https://www.zdrojak.cz/clanky/prehled-podpory-xml-v-php5/>>
- [3] *Historical trends in the usage of server-side programming languages for websites* [online]. [cit. 2017-10-21]. Dostupné z: <https://w3techs.com/technologies/history_overview/programming_language>
- [4] ČÁPKA, David. Úvod do JavaScriptu. In: *Itnetwork* [online]. 2013 [cit. 2017-10-22]. Dostupné z: <<https://www.itnetwork.cz/javascript/zaklady/javascript-tutorial-uvod-do-javascriptu-nepochopeny-jazyk/>>
- [5] Úvod do JavaScriptu. In: *Jak Psát Web* [online]. Praha [cit. 2017-10-22]. Dostupné z: <<https://www.jakpsatweb.cz/javascript/javascript-uvod.html>>
- [6] JELÍNEK, Lukáš. JQuery (1) — javascriptový framework, který (skoro) každý používá. In: *Linuxexpres* [online]. 2014 [cit. 2017-10-23]. Dostupné z: <<https://www.linuxexpres.cz/software/jquery-1-javascriptovy-framework-ktery-skoro-kazdy-pouziva>>
- [7] Vyšlo jQuery 1.7. *Jquery návod* [online]. [cit. 2017-10-23]. Dostupné z: <<http://jquery-navod.cz/kategorie-zpravicky/vyslo-jquery-1-7>>
- [8] *JQuery* [online]. [cit. 2017-10-23]. Dostupné z: <<http://jquery.com/>>
- [9] ADAMS, Cameron. AJAX: Usable Interactivity with Remote Scripting. In: *Sitepoint* [online]. 2005 [cit. 2017-10-23]. Dostupné z: <<https://www.sitepoint.com/remote-scripting-ajax/>>
- [10] VRÁNA, Jakub. AJAX. In: *ROOT.CZ_* [online]. 2005 [cit. 2017-10-23]. Dostupné z: <<https://www.root.cz/clanky/ajax/>>
- [11] Getting Started. *MDN Web Docs Moz://a* [online]. [cit. 2017-10-23]. Dostupné z: <https://developer.mozilla.org/en-US/docs/AJAX/Getting_Started>
- [12] JELÍNEK, Lukáš. JQuery (16) — úvod do jQuery UI. In: *Linuxexpres* [online]. 2015 [cit. 2017-10-23]. Dostupné z: <<https://www.linuxexpres.cz/software/jquery-16-uvod-do-jquery-ui>>

- [13] VANTOLL, T. J. *JQuery UI in action* [online]. Shelter Island, NY: Manning Publications, 2015 [cit. 2017-10-23]. ISBN 978-161-7291-937. Dostupné z: <https://livebook.manning.com/#!/book/jquery-ui-in-action/chapter-1/17>>
- [14] *MySQL databáze - český manuál* [online]. [cit. 2017-10-30]. Dostupné z: <http://www.junext.net/mysql/>>
- [15] ZAJÍC, Petr. MySQL (4) - něco terminologie. In: *Linuxsoft.cz* [online]. 2005 [cit. 2017-10-30]. Dostupné z: http://www.linuxsoft.cz/article.php?id_article=744>
- [16] Syntaxe HTML. *Jak Psát Web* [online]. [cit. 2017-11-04]. Dostupné z: <https://www.jakpsatweb.cz/html/syntaxe.html>>
- [17] HTML & CSS. *W3C* [online]. [cit. 2017-11-04]. Dostupné z: <https://www.w3.org/standards/webdesign/htmlcss>>
- [18] Jazyk webov HTML má novú verziu, HTML 5.1. *Digitálny Svet pod Lupou* [online]. 2016 [cit. 2017-11-04]. Dostupné z: <http://www.dsl.sk/article.php?article=19026>>
- [19] Zápisy CSS. *Jak Psát Web* [online]. [cit. 2017-11-04]. Dostupné z: <http://polopate.jakpsatweb.cz/index.php?page=zapis-css#externi>>
- [20] CSS2 — selektory, pseudotřídy a pseudoelementy. *Interval.cz* [online]. 2002 [cit. 2017-11-04]. Dostupné z: <https://www.interval.cz/clanky/css2-selektory-pseudotridy-a-pseudoelementy/>>
- [21] EVERS, Jeannie a Emdash EDITING, ed. GIS (geographic information system). In: *National Geographic* [online]. Caryl-Sue, National Geographic Society, 2017 [cit. 2017-11-28]. Dostupné z: <https://www.nationalgeographic.org/encyclopedia/geographic-information-system-gis/>>
- [22] ING. HRUBÝ, PH.D., Martin. *Geografické Informační Systémy (GIS) Studijní opora*. Brno, 2006. Učební text. Vysoké učení technické v brně.
- [23] KRYGIER, John a Denis WOOD. *Making maps: a visual guide to map design for GIS*. Third Edition. New York: Guilford Press, 2016. ISBN 978-1462509980.
- [24] CO JE GIS? *Geoportal Praha* [online]. Praha, 2010 [cit. 2017-11-28]. Dostupné z: <http://www.geoportalpraha.cz/cs/clanek/11/co-je-gis#.Wh2Tq0riaMp>>

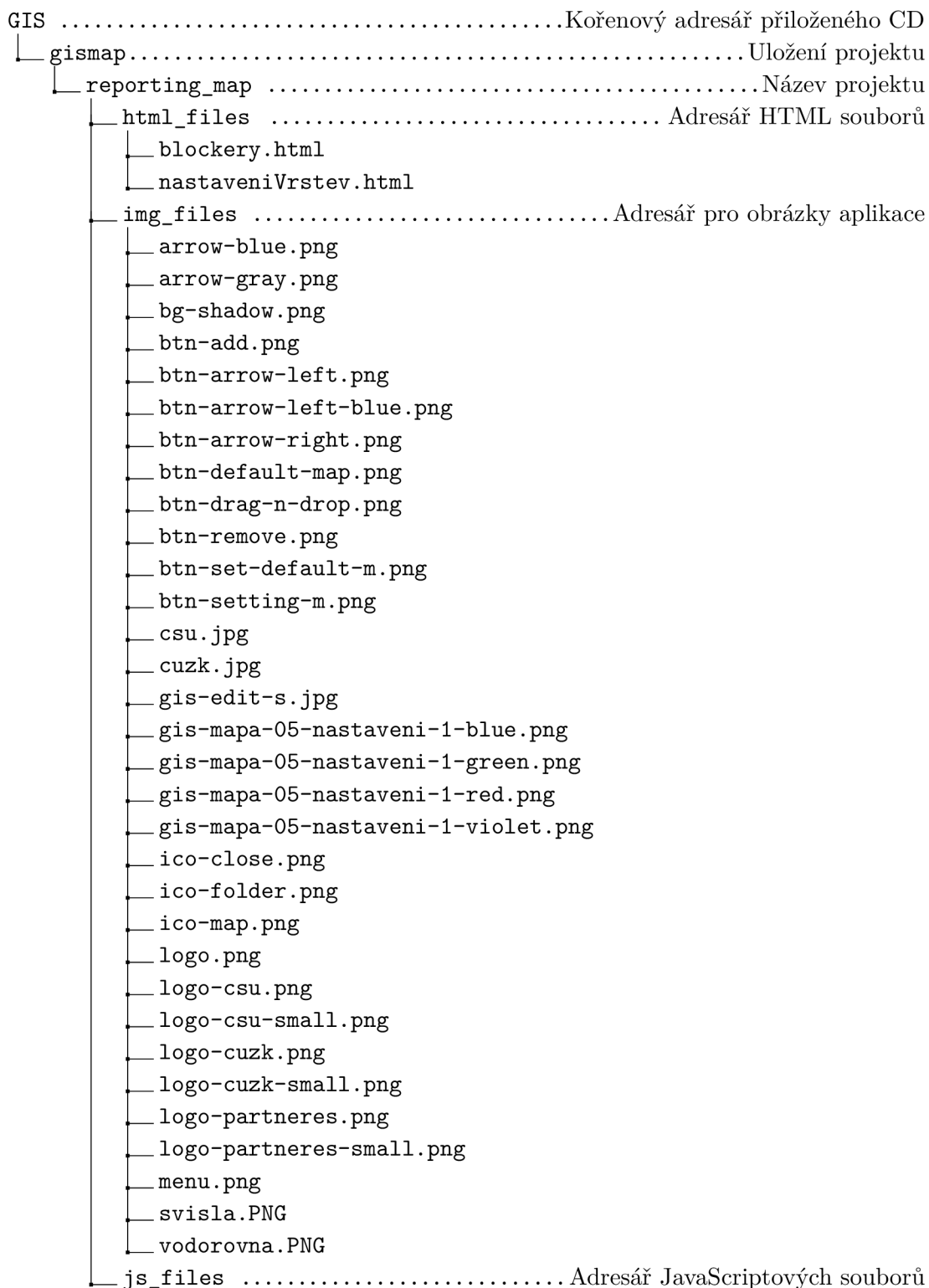
- [25] *Český úřad zeměměřický a katastrální* [online]. Praha, 2017 [cit. 2017-11-28]. Dostupné z: <<http://www.cuzk.cz/>>
- [26] *Český statistický úřad* [online]. Praha, 2017 [cit. 2017-11-28]. Dostupné z: <<https://www.czso.cz/>>
- [27] *GitLab* [online]. [cit. 2017-11-04]. Dostupné z: <<https://about.gitlab.com>>
- [28] *API mapy* [online]. Praha, 2018 [cit. 2018-05-19]. <<https://api.mapy.cz/>>
- [29] WEISFELD, Matt A. *The object-oriented thought process*. Fourth edition. Upper Saddle River, NJ: Addison-Wesley, 2013. ISBN 978-0321861276.

SEZNAM SYMBOLŮ, VELIČIN A ZKRATEK

GIS	Geografický informační systém
ČSÚ	Český statistický úřad
ČÚZK	Český úřad zeměměřický a katastrální
API	Application Programming Intergace
PHP	Hypertext Preprocessor
HTML	HyperText Markup Language
AJAX	Asynchronous JavaScript and XML
CSS	Cascading Style Sheets
IZS ČR	Integrovaný záchranný sbor České republiky
RÚIAN	Registr územní identifikace, adres a nemovitostí
XML	Extensible Markup Language
URL	Uniform Resource Locator
HTTP	Hypertext Transfer Protocol
SVG	Scalable Vector Graphics

PŘÍLOHA: CD MÉDIUM

Na přiloženém CD je projekt GeoFiP, který obsahuje adresářovou logiku, HTML, JavaScriptové, PHP a obrázkové soubory. PHP soubory jsou psány ve verzi jazyka PHP 5.6 a pro JavaScriptové kódy je využita knihovna jQuery verze jquery-3.2.1.js.





- KrajeSoudrznosti.php
- Line.php
- Mapa.php
- MapovaKompozice.php
- MapovyPrvek.php
- NaslouchaceMapy.php
- Okresy.php
- OptionsGeometry.php
- ORP
- PoleSouradnic.php
- Polygon.php
- SimpleMarker.php
- SrafovaniDefine.php
- SVGPrvky.php
- SVGPrvkyKraje.php
- SVGPrvkyOkresy.php
- Vrstva.php