



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA INFORMAČNÍCH TECHNOLOGIÍ

FACULTY OF INFORMATION TECHNOLOGY

ÚSTAV INTELIGENTNÍCH SYSTÉMŮ

DEPARTMENT OF INTELLIGENT SYSTEMS

ZPRACOVÁNÍ VÝMĚNNÉHO FORMÁTU INFORMAČNÍHO SYSTÉMU KATASTRU NEMOVITOSTÍ

PROCESSING OF THE EXCHANGE INFORMATION FORMAT OF THE REAL ESTATE CADASTRE

INFORMATION SYSTEM

BAKALÁŘSKÁ PRÁCE

BACHELOR'S THESIS

AUTOR PRÁCE

AUTHOR

MARTIN NIZNER

VEDOUcí PRÁCE

SUPERVISOR

RADEK KOČÍ, Ing., Ph.D.

BRNO 2019

Zadání bakalářské práce



21999

Student: **Nizner Martin**
Program: Informační technologie
Název: **Zpracování výměnného formátu Informačního systému katastru nemovitostí**
Processing of the Exchange Information Format of the Real Estate Cadastre Information System

Kategorie: Databáze

Zadání:

1. Seznamte se s Registrem územní identifikace, adres a nemovitostí (RÚIAN) spravovaným Českým úřadem zeměměřickým a katastrálním. Nastudujte výměnné formáty pro přenos dat z RÚIAN.
2. Navrhněte nástroj, který umožňuje zpracování dat získaných z RÚIAN. Nástroj musí být jednoduše adaptabilní na změnu výměnného formátu a musí být schopen detekovat změny dat od posledního zpracování.
3. Navrhněte model dat vhodný pro uchování informací z RÚIAN pro účely genealogických databází.
4. Navržený nástroj implementujte a realizujte model dat ve vhodném databázovém systému.
5. Vyhodnoťte dosažené výsledky a diskutujte možná rozšíření nástroje.

Literatura:

- ČÚZK: Veřejný dálkový přístup (VDP) Uživatelská dokumentace, dostupné on-line <https://www.cuzk.cz/VDP-uzivatelska-dokumentace>, říjen 2018.

Pro udělení zápočtu za první semestr je požadováno:

- První tři body zadání.

Podrobné závazné pokyny pro vypracování práce viz <http://www.fit.vutbr.cz/info/szz/>

Vedoucí práce: **Kočí Radek, Ing., Ph.D.**
Vedoucí ústavu: Hanáček Petr, doc. Dr. Ing.
Datum zadání: 1. listopadu 2018
Datum odevzdání: 15. května 2019
Datum schválení: 1. listopadu 2018

Abstrakt

Cielom tejto bakalárskej práce je implementovať databázu, ktorá bude vhodná pre uloženie dát registru územnej identifikácie adries a nehnuteľností spolu s nástrojom, ktorý umožní vkladanie a aktualizovanie prvkov v danej databáze. Úvodná časť tejto práce sa venuje vysvetleniu riešenej problematiky. Jadro práce sa venuje návrhu a implementácií databázy a nástroja pre jej obsluhu. Záver práce tvoria ukázkové dotazy nad implementovanou databázou spolu s nápadmi na vylepšenie.

Abstract

The aim of this bachelor thesis is to implement the database, that will be appropriate to save the data of the register of territorial identification of addresses and real estate along with the tool that will make it possible to insert and update the items of the database. The introductory part of the work is devoted to explanation of the solved issue. The core of the work is devoted to implemented database and the tool for its maintenance. The conclusion of the thesis consists of sample queries over the implemented database along with ideas for improvement.

Kľúčové slová

databáza, RÚIAN, súbory, URL, VDP, VFR, XML

Keywords

database, files, RÚIAN, URL, VDP, VFR, XML

Citácia

NIZNER, Martin. *Zpracování výměnného formátu Informačního systému katastru nemovitostí*. Brno, 2019. Bakalářská práce. Vysoké učení technické v Brně, Fakulta informačních technologií. Vedoucí práce Radek Kočí, Ing., Ph.D.

Zpracování výměnného formátu Informačního systému katastru nemovitostí

Prehlásenie

Prehlasujem, že som túto bakalársku prácu vypracoval samostatne pod vedením pana Radka Kočího ,Ing., Ph.D. Uviedol som všetky literárne pramene a publikácie, z ktorých som čerpal.

.....

Martin Nizner

14. mája 2019

Podakovanie

Rád by som chcel poďakovať vedúcemu práce, Radkovi Kočímu, Ing., Ph.D., za konzultácie a odbornú pomoc pri vývoji aplikácie.

Obsah

1	Úvod	3
2	Vysvetlenie riešenej problematiky	4
2.1	Register územných adries a nehnuteľností	4
2.2	Typy poskytovania údajov z RÚIAN	4
2.2.1	Aplikácia verejného diaľkového prístupu	4
2.2.2	Výmenný formát RÚIAN	5
2.2.3	Webové služby informačného systému základných registrov	5
2.2.4	Súbory s hodnotami oddelenými čiarkami	5
2.3	Výmenný formát registru územných adries a nehnuteľností	6
2.4	Formát súborov VFR	6
2.5	Kolekcie údajov vo VFR	6
2.6	Základné dátové typy VFR	9
2.7	Typy súborov VFR	10
2.7.1	Rozdelenie podľa obsahu prvkov RÚIAN	11
2.7.2	Rozdelenie podľa obsahu atribútov	11
2.7.3	Rozdelenie podľa časovej platnosti	11
2.7.4	Rozdelenie podľa pokrytia prvkov RÚIAN	12
2.8	Špeciálny VFR	12
2.9	Možnosti získania súborov VFR	13
2.9.1	Získanie súborov VFR z aplikácie verejného diaľkového prístupu	13
2.9.2	Získanie súborov VFR pomocou webových služieb registru územných adries a nehnuteľností	14
2.10	Štruktúra URL adries získaných pomocou aplikácie VDP	15
2.10.1	Príklady URL adries súborov VFR	15
3	Návrh databázového modelu	17
3.1	Zvolenie prvkov RÚIAN a ich atribútov pre export	17
3.2	Databázový model	18
3.2.1	Základná geometria	18
3.2.2	Generalizované hranice	19
3.2.3	Skloňovacie pády	20
3.2.4	Súbory číselníkov	21
3.2.5	Prvá časť prvkov RÚIAN	22
3.2.6	Druhá časť prvkov RÚIAN	24
4	Implementácia nástroja pre spracovanie prvkov RÚIAN	25
4.1	Zvolenie implementačných nástrojov	25

4.2	Štruktúra skriptu	25
4.3	Spracovanie argumentov príkazového riadku VFR	26
4.3.1	Získanie URL adres a extrakcia súborov VFR z archívov	26
4.4	Spracovanie extrahovaných XML súborov	27
4.5	Práca s databázou	27
5	Testovanie databázy	29
6	Záver	32
	Literatúra	33
A	Obsah priloženého CD	35
B	Manuál	36
B.1	Vytvorenie databázy a jej prepojenie s PHP skriptom	36
B.2	Spustenie skriptu	36

Kapitola 1

Úvod

Táto bakalárska práca má za cieľ implementovať databázu, ktorá bude vhodná pre uloženie dát registru územnej identifikácie, adries a nehnuteľností. O obsluhu implementovanej databázy sa bude starať nástroj, ktorý je súčasťou výslednej bakalárskej práce. Tento nástroj bude užívateľom umožňovať naplnenie prázdnej implementovanej databázy všetkými potrebnými dátami z registru územnej identifikácie adries a nehnuteľností spolu s aktualizáciou tejto databázy. Navrhnutý nástroj spolu s databázou by mali uľahčiť rôznym informačným systémom a mapovým klientom prístup k dátam registru územnej identifikácie, adries a nehnuteľností tak, aby nemuseli sami získavať tieto dáta pomocou výmenného formátu, ale mohli si ich jednoducho načítať z navrhutej databázy, ktorá bude umiestnená na ich serveri.

Úvodná časť tejto práce sa venuje vysvetleniu riešenej problematiky od objasnenia pojmu RÚIAN až po získanie dát RÚIAN pomocou výmenného formátu (kapitola 2). Jadro práce sa venuje návrhu databázového modelu (kapitola 3) a implementácií nástroja pre spracovanie prvkov RÚIAN (kapitola 4). Záver práce tvoria ukážkové dotazy nad implementovanou databázou (kapitola 5) spolu s nápadmi na vylepšenie výsledného implementovaného nástroja, ktoré sú súčasťou záveru (kapitola 6).

Kapitola 2

Vysvetlenie riešenej problematiky

Táto kapitola sa venuje vysvetleniu a objasneniu pojmov, ktoré sa týkajú riešenej problematiky. Kapitola v úvode popisuje register územných adries a nehnuteľností, ktorého dáta boli v tejto práci získané. V jadre kapitoly je podrobne popísaný výmenný formát RÚIAN, pomocou ktorého boli dáta RÚIAN získané.

2.1 Register územných adries a nehnuteľností

V polovici roku 2012 bol úspešne spustený systém základných registrov verejnej správy Českej republiky. Jeden zo štyroch základných registrov je i register územnej identifikácie, adries a nehnuteľností. RÚIAN je verejný zoznam, ktorý umožňuje užívateľom z verejnej, ale i komerčnej a akademickej sféry, diaľkový prístup cez internet [18].

2.2 Typy poskytovania údajov z RÚIAN

Existuje niekoľko spôsobov, ktoré umožňujú získanie údajov z RÚIAN. Všetky existujúce postupy sú uvedené v tejto podkapitole.

2.2.1 Aplikácia verejného diaľkového prístupu

Spôsob umožňuje nazerať a získavať dáta základného registru RÚIAN a tiež niektoré dáta editačného agendového informačného systému územnej identifikácie [5] a informačného systému katastru nehnuteľností [6] pomocou aplikácie verejného diaľkového prístupu [15]. Pre prístup do aplikácie VDP¹ nie je potrebná žiadna registrácia. Poskytované dáta z VDP sú zadarmo. Dáta poskytované prostredníctvom VDP nie sú referenčné, majú iba informačný charakter. Aplikácia VDP obsahuje nasledujúce funkcionality:

- vyhľadanie aktuálnych i zrušených prvkov
- zobrazenie detailných informácií k vybranému prvku
- export jednotlivých prvkov RÚIAN vo formátoch PDF [16], CSV [4] a XML [7]
- zobrazenie RÚIAN prvkov na mape pomocou mapového klienta Marushka [12]

¹Verejný diaľkový prístup

- poskytnutie informácií o vybranom prvku k určitému dátumu v minulosti (iba od 1.7.2011 nakoľko VDP eviduje dáta až od tohto dátumu)
- poskytnutie informácií o vlastníkovi pozemku alebo stavebného objektu (budovy) evidovaného v katastri nehnuteľností pomocou presmerovania do aplikácie nazerania do katastru nehnuteľností [13].
- overenie adresy vrátane vyhľadania adresy na základe neúplných údajov.
- poskytnutie obsahu registru RÚIAN a ISÚI² vo výmennom formáte RÚIAN, ktorý je načrtnutý v podkapitole 2.2.2.

Ku všetkým funkcionalitám aplikácie VDP sa dá dostať pomocou úvodnej obrazovky aplikácie VDP[17].

Keďže účelom projektu je získať dáta RÚIAN pomocou výmenného formátu RÚIAN, tak zo všetkých poskytovaných funkcionalít aplikácie VDP bola využitá iba funkcionalita slúžiaca pre poskytnutie obsahu registru RÚIAN vo výmennom formáte RÚIAN.

2.2.2 Výmenný formát RÚIAN

Spôsob umožňuje získavať dáta z RÚIAN pomocou výmenného formátu RÚIAN.[8] Jedná sa o súbory vo formáte XML. Tieto súbory je možné sťahovať nasledujúcimi spôsobmi:

- prostredníctvom aplikácie VDP (v rámci projektu bol zo všetkých možných spôsobov získania dát RÚIAN zvolený práve tento jeden spôsob)
- z internetových (URL) adries vrátených webovými službami informačného systému základných registrov [14] – dostupné iba orgánom štátnej správy a samosprávy (nie sú voľne prístupné širokej verejnosti)

Analýze VFR³ sa ďalej podrobne venuje kapitola 2.3

2.2.3 Webové služby informačného systému základných registrov

V rámci ISZR⁴ fungujú štyri základné registre verejnej správy (**register osôb, register obyvateľov, RÚIAN, register práv a povinností**) [10]. Webové služby ISZR⁵ slúžia pre komunikáciu a výmenu dát medzi registrovanými agendovými informačnými systémami[9] so základnými registrami verejnej správy. Služby sú publikované na eGON [11] rozhraní systémov základných registrov.

2.2.4 Súbory s hodnotami oddelenými čiarkami

Súbory obsahujú dáta RÚIAN vo formáte CSV a sú generované raz za mesiac z úplného VFR súboru. Pre každú obec existuje jeden samostatný CSV súbor[4]. Tieto CSV súbory je možné získať pomocou aplikácie nazerania do katastru nehnuteľností [13].

²Informačný systém územnej identifikácie

³Výmenný formát RÚIAN

⁴Informačný systém základných registrov

⁵Informačný systém základných registrov

2.3 Výmenný formát registru územných adries a nehnuteľností

VFR predstavuje jednu z foriem získavania a poskytovania dát RUIÁN, pričom dáta RUIÁN sú pri tejto forme uložené vo forme súborov, ktoré obsahujú tieto dáta vo formáte VFR. Keďže účelom projektu je získanie dát RUIÁN práve pomocou VFR, tak je tejto kapitole v rámci teoretickej časti venovaná najväčšia pozornosť, keďže je nevyhnutné pochopiť do detailu tento formát dát, nakoľko má byť extrahovaný do databázy. Budú teda vysvetlené a objasnené všetky dôležité súčasti VFR od formátu súborov VFR až po formát URL [2] adries, na ktorých sú tieto VFR súbory uložené.

2.4 Formát súborov VFR

Súbory VFR sú väčšinou uložené a komprimované na URL adresách vo formáte **gzip** (niekedy sa vyskytnú prípady, keď sú uložené vo formáte **zip**). V týchto archívoch sa vždy nachádza presne jeden súbor vo formáte XML, v ktorom sa nachádza stromová štruktúra zložená z elementov, ktoré ohraničujú kolekciu údajov a elementov, ktoré ohraničujú hodnoty atribútov týchto kolekcí. Nakoľko táto stromová štruktúra obsahuje geografické prvky (napríklad. súradnice definičného bodu), je uložená vo formáte GML (konkrétne sa jedná o GML 3.2.1)[3].

2.5 Kolekcie údajov vo VFR

Vo VFR existuje niekoľko rôznych kolekcí údajov. V tejto podkapitole budú bližšie predstavené tie kolekcie údajov, ktoré budú využité v navrhovanom databázovom modeli pre projekt. Pochopenie štruktúr týchto kolekcí údajov je nevyhnutné z pohľadu správneho návrhu jednotlivých databázových entít navrhovaného databázového modelu.

Prvok VFR je základnou kolekciou údajov vo VFR. Je to vlastne prvok RUIÁN zapísaný vo formáte GML a obsahuje vo svojej kolekcií všetky atribúty a kolekcie, ktoré spadajú pod daný prvok. Tieto kolekcie a atribúty slúžia pre popis územného celku, ktorý spadá pod daný prvok VFR (napríklad v prípade prvku VFR **Obec** sú popísané všetky územné prvky, ktoré spadajú pod územný celok obec - ulice, časti obce a podobne). Každý prvok VFR teda popisuje určitý územný celok. V tabuľke 2.1 sú uvedené všetky prvky VFR, ktoré boli vo forme tabuliek zahrnuté do navrhovaného databázového modelu (názvy prvkov VFR odpovedajú ich názvom v súboroch VFR). V tabuľke je u každého prvku VFR uvedená tiež jeho nadradená kolekcia, ktorá v súboroch VFR obaľuje všetky prvky VFR s daným názvom.

Tabuľka 2.1: Prehľad prvkov VFR

Názov prvku VFR	Popis	Nadradená kolekcia
Stat	popisuje územie štátu	Staty
RegionSoudrznosti	popisuje územie regiónu súdržnosti	RegionySoudrznosti
Vusc	popisuje územie vyššieho územne samosprávneho celku	Vusc
Kraj	popisuje územie kraja podľa zákona číslo č.36/1960 Sb.	Kraje
Okres	popisuje územie okresu	Okresy
Orp	popisuje správny obvod obce s rozšírenou pôsobnosťou	Orp
Pou	popisuje správny obvod obce s rozšírenou pôsobnosťou	Pou
Obec	popisuje územie obce	Obce
CastObce	popisuje územie časti obce	CastiObci
Mop	popisuje územie mestského obvodu v meste Praha	Mop
SpravniObvod	popisuje územie správneho obvodu v meste Praha	SpravniObvody
Momc	popisuje územie mestského obvodu a mestskej časti územne členeného statuárneho mesta	Momc
KatastralniUzemi	popisuje územie katastrálneho územia podľa katastrálneho zákona č. 344/1992 Sb.	KatastralniUzemi
Ulice	popisuje územie ulice alebo iného verejného priestranstva	Ulice
Zsj	popisuje územie základnej sídelnej jednotky	Zsj

Všetky prvky VFR uvedené v tabuľke sú spravované v agendovom systéme ISÚI a sú ukladané vo svojej nadradenej kolekcií údajov (napríklad kolekcia údajov **Obce** môže obsahovať jednu alebo niekoľko kolekcií údajov **Obec**). Prvok **Stat** tvorí najväčšiu kolekciu údajov vo VFR, nakoľko obsahuje všetky podradené prvky katastrálnych území, obcí a pod. spolu z ich atribútami a kolekciami údajov.

Cudzí kľúč je kolekciou údajov, ktorá je súčasťou kolekcie prvku VFR a vždy obsahuje kód nadradeného prvku VFR. Príklad cudzieho kľúča v súbore VFR:

```
<vf:Obec gml:id="0B.592251">
  <obi:Okres>
    <oki:Kod>3705</oki:Kod>
  </obi:Okres>
</vf:Obec>
```

Tento príklad obsahuje cudzí kľúč kolekcie **Okres** v nadradenej kolekcií **Obec**. Pomocou tohto cudzieho kľúča je možné zistiť, do akého okresu daná obec patrí.

MluvnickeCharakteristiky - táto kolekcia údajov obsahuje skloňovacie pády (konkrétne druhý až siedmy pád okrem piateho pádu), ktoré sú využité u niektorých prvkov RÚIAN (napríklad prvok **Obec**). Každý skloňovací pád je znakový typ o dĺžke 48 znakov. Príklad skloňovacích pádov v súbore VFR:

```
<vf:Obec gml:id="0B.592251">
  <obi:MluvnickeCharakteristiky>
    <com:Pad2>Prahy</com:Pad2>
    <com:Pad3>Praze</com:Pad3>
    <com:Pad4>Prahu</com:Pad4>
    <com:Pad6>Praze</com:Pad6>
    <com:Pad7>Prahou</com:Pad7>
  </obi:MluvnickeCharakteristiky>
</vf:Obec>
```

Jedná sa o skloňovacie pády mesta Prahy.

Geometria Táto kolekcia obsahuje informácie o geometrickom určení prvku VFR. Každý atribút kolekcie geometrie má v súboroch VFR prefix **gml** a jedná sa o časť XML súboru, ktorá je vo formáte GML. Kolekcia má viac typov, pričom v tejto práci boli využité iba dva typy tejto kolekcie :

- **DefiniciBod** - tento typ kolekcie geometrie obsahuje **x**-ovú a **y**-ovú súradnicu definičného bodu prvku VFR a jednoznačne tak určuje presnú pozíciu daného prvku VFR na mape. Príklad kolekcie definičného bodu:

```
<vf:Obec gml:id="OB.592251">
  <obi:Geometrie>
    <gml:MultiPoint gml:id="DOB.554782">
      <gml:pointMembers>
        <gml:Point gml:id="DOB.554782.1">
          <gml:pos>-743100.00 -1043300.00</gml:pos>
        </gml:Point>
      </gml:pointMembers>
    </gml:MultiPoint>
  </obi:Geometrie>
</vf:Obec>
```

Súradnice definičného bodu sa nachádzajú v elemente **gml:pos**, pričom v mapovom kliente (Marushka) sú použiteľné iba v prípade, keď sa odstráni úvodný znak pomlčky. Ostatné elementy sú z pohľadu obsahu elementu **obi:Geometrie** pre využitie v mapových klientoch nezaujímavé.

- **GeneralizovaneHranice** - tento typ kolekcie geometrie obsahuje všetky súradnice polygónu, ktorý definuje územie daného prvku VFR. Príklad kolekcie generalizovaných hraníc:

```
<obi:GeneralizovaneHranice3>
  <gml:MultiSurface gml:id="GOB.558567.3">
    <gml:surfaceMember>
      <gml:Polygon gml:id="GOB.558567.3.1">
        <gml:exterior>
          <gml:LinearRing>
            <gml:posList>-833663.85 -1078072.78 -833596.36
            -1078192.40 -833506.20 -1078221.94
            -833534.02 -1078394.63 -833488.64
            </gml:posList>
          </gml:LinearRing>
        </gml:exterior>
      </gml:Polygon>
    </gml:surfaceMember>
  </gml:MultiSurface>
</obi:GeneralizovaneHranice3>
```

Jednotlivé súradnice sa nachádzajú v elemente **gml:posList** a sú oddelené medzerou. Pre ich využitie v mapovom kliente je potrebné rovnako ako v prípade kolekcie definičného bodu odstrániť úvodný znak pomlčky.

Okrem týchto typov kolekcie geometrie ešte existujú ďalšie dva typy tejto kolekcie (**OriginalniHranice** a **DefinicniCara**), ktoré však nebudú popísané z dôvodu ich nevyužitia v tejto práci.

Okrem vyššie uvedených kolekcií ešte VFR obsahuje ďalšie kolekcie údajov, ktoré však v tejto práci nie sú využité a preto nebudú bližšie špecifikované (**NespravneUdaje**, **BonitovaneDily**, **ZpusobyOchranyPozemku + ZpusobyOchrany**, **CislaDomovni**, **DetailniTEA**).

2.6 Základné dátové typy VFR

Existuje niekoľko základných dátových typov, ktoré sa vyskytujú v súboroch VFR a je potrebné sa s nimi zoznámiť z dôvodu správneho návrhu databázového modelu. Ku každému dátovému typu je v nasledujúcom zozname uvedený príklad atribútov, v ktorých sa daný dátový typ používa. Zoznam všetkých atribútov, ktoré sa nachádzajú v súboroch VFR vrátane ich popisu a využitia v prvkoch VFR je uvedený v tabuľke 2.2.

Zoznam základných dátových typov VFR:

- **Integer** - tento dátový typ má celočíselný rozsah od -2147483648 do 2147483647. Dátový typ sa najčastejšie vyskytuje v kóde prvku RÚIAN, keďže každý prvok RÚIAN musí obsahovať kód, ktorý je jedinečný a je jeho jednoznačným identifikátorom. Ďalej sa tento typ vyskytuje v rôznych číselných údajoch (napríklad počet bytov, počet podlaží u stavebného objektu).
- **String** - jedná sa o znakový typ rôznej dĺžky. Najčastejšie sa tento typ vyskytuje v atribúte Názov prvku VFR, keďže každý prvok VFR musí obsahovať názov. Ďalej sa tento typ vyskytuje napríklad u prvkov obcí, kde dosahuje v niektorých prípadoch až štyritisíc znakov (atribúty VlajkaText a ZnakText).
- **Boolean** - jedná sa o dátový typ, ktorý môže nadobúdať iba jednu z dvoch hodnôt pravda / nepravda (true/false). Tento dátový typ sa vyskytuje napríklad v atribúte NespravnyUdaj, ktorý určuje, či je daný VFR prvok správny, alebo či nie je jeho platnosť spochybnená.
- **DateTime** - tento typ definuje výpis dátumu a času v tvare RRRR-MM-DDTHH:MM:SS (napríklad 2019-05-01T00:00:00). Typ sa najčastejšie vyskytuje v atribúte PlatiOd, ktorý u každého prvku VFR určuje začiatok platnosti (začiatok platnosti v RÚIAN).
- **Long** - jedná sa o dátový typ s celočíselným rozsahom väčším než Integer. U prvkov VFR sa teda používa v prípadoch, u ktorých je potrebné použiť väčšiu číselnú hodnotu, než umožňuje Integer (používa sa iba u atribútov IdTransakce a Globalni-IdNavrhuZmeny).
- **base64Binary** - tento dátový typ sa používa pre vykreslenie obrázkov u prvkov VFR. Využíva sa teda v atribútoch VlajkaObrazek a ZnakObrazek (prvky VFR Obec a Stat).

Tabuľka 2.2: Prehľad atribútov VFR a ich využitia v prvkoch VFR

Atribút	Typ	Dĺžka	Popis	Prvky VFR
Kod	Integer	2 až 9	Kód prvku VFR	(*)
Nazev	String	32 až 48	Názov prvku vo VFR	(*)
Nespravny	Boolean		Príznak nesprávnosti údaju	(*)
PlatiOd	DateTime		Začiatok platnosti	(*)
PlatiDo	DateTime		Koniec platnosti	(*)
IdTransakce	Long	18	identifikátor prvku VFR v databáze RÚIAN	(*)
GlobalniIdNavrhuZmeny	Long	18	identifikátor návrhu zmeny v ISÚI	(*)
Geometrie	Geometrie		Geometrické údaje	(*)
NespravneUdaje	NespravneUdaje		Zoznam nesprávnych referenčných údajov	(*)
DatumVzniku	DateTime		Skutočný dátum vzniku prvku	(*) - (Ulice)
MluvnickeCharakteristiky	MluvnickeCharakteristiky		Druhý až siedmy skloňovací pád	Obec, CastObce, MOMC, KatastralniUzemi
NutsLau	String	6	Kód územného prvku VFR v NUTS / LAU	Stat, RegionSoudrznosti, VUSC, Okres, Obec
Stat	Cudzí kľúč		Nadradený Stat	RegionSoudrznosti, Kraj
Vusc	Cudzí kľúč		Nadradené Vúsc	Okres, Orp
SpravniObecKod	Integer	6	Kód nadradenej správnej obce	Orp, Pou
VlajkaText	String	4000	Text popisujúci vlajku obce	Obec, Mome
VlajkaObrazek	base64Binary		Obrázok vlajky obce	Obec, Mome
ZnakText	String	4000	Text popisujúci znak obce	Obec, Mome
ZnakObrazek	base64Binary		Obrázok znaku obce	Obec, Mome
Orp	Cudzí kľúč		Nadradená Orp	Pou
StatusKod	Integer	2	Status obce	Obec
Okres	Cudzí kľúč		Nadradený Okres	Obec
Pou	Cudzí kľúč		Nadradená Pou	Obec
CleneniSmRozsahKod	Integer	2	Rozsah členenia statuárneho mesta na MOMC	Obec
CLeneniSmTypKod	Integer	2	Typ MOMC, na ktoré je statuárne mesto rozčlenené	Obec
ExistujeDigitalniMapa	Boolean		Príznak existencie digitálnej mapy	KatastralniUzemi
RizeniID	Long	18	identifikátor riadenia v ISKN	KatastralniUzemi
Kraj	Cudzí kľúč		Nadradený Kraj	Okres
SpravniMomeKod	Integer	6	Kód správneho MOMC	SpravniObvod
Vymera	Long	18	Vymera ZSJ v m2	ZSJ
CharakterZsjKod	Integer	4	Prevažujúci spôsob využitia ZSJ	ZSJ

2.7 Typy súborov VFR

Súbory VFR je možné na základe prvkov VFR rozdeliť na niekoľko rôznych typov podľa rôznych kritérií. V nasledujúcej podkapitole budú uvedené všetky typy súborov, ktoré VFR poskytuje pri vyhľadávaní pomocou aplikácie VDP. Na základe typov súborov, ktoré VFR poskytuje, je možné rozdeliť typy súborov VFR podľa štyroch základných kritérií, na ktoré bude táto podkapitola rozdelená :

- obsah prvkov RÚIAN
- obsah atribútov
- časová platnosť
- pokrytie prvkov RÚIAN

Na základe každého z vyššie uvedených kritérií sa dá určiť, o aký typ súboru VFR sa jedná. O súbore VFR sa teda dá vždy povedať, že sa skladá z viacerých typov (má typ určený na základe obsahu atribútov, typ určený na základe časovej platnosti a podobne). Každé kritérium obsahuje vždy dva typy súborov VFR. Všetky uvedené typy súborov sa dajú naklikat' vo vyhľadávacom formulári VDP tak, aby sme získali požadovaný súbor VFR.

2.7.1 Rozdelenie podľa obsahu prvkov RÚIAN

Toto kritérium rozdelenia určuje, aké prvky RÚIAN sa v súbore VFR budú vyskytovať a obsahuje tieto typy súborov VFR:

- **Súbory typu štát až zsj** - Súbory zahŕňajú tieto prvky RÚIAN: Stat, Region-Soudrznosti, Vusc, Orp, Kraj, Okres, Obec, CastObce, Momc, Mop, SpravniObvod, KatastralniUzemi, ZSJ. Tento typ súborov sa oplatí využiť, keď potrebujeme prístupit' k základným územným celkom (nepotrebujeme analyzovať najnižšie územné celky).
- **Súbory typu obec a jej podradené prvky** - súbory zahŕňajú tieto prvky RÚIAN: Obec, CastObce, Momc, Mop, SpravniObvod, KatastralniUzemi, Zsj, Ulice, Parcela, StavebniObjekt, AdresniMisto. Tento typ súborov sa oplatí využiť, keď potrebujeme analyzovať najnižšie územné celky spadajúce pod obec, ktoré sa nenachádzajú v súboroch typu **štát až zsj**. Pre každú obec existuje jeden samostatný súbor tohto typu.

2.7.2 Rozdelenie podľa obsahu atribútov

Toto kritérium rozdelenia určuje, aké atribúty prvkov RÚIAN sa budú v súboroch VFR nachádzať a obsahuje tieto typy súborov VFR:

- **Súbory základnej dátovej sady** - súbory zahŕňajú iba textové atribúty prvkov RÚIAN a u atribútu **Geometrie** iba jeden typ - definičný bod. Tieto súbory sa oplatí použiť, keď nám postačia iba textové údaje o jednotlivých prvkoch RÚIAN a plánujeme lokalizovať jednotlivé prvky RÚIAN iba na základe ich definičného bodu.
- **Súbory kompletnej dátovej sady** - súbory môžu obsahovať okrem textových atribútov i binárne dáta (napríklad atribút **VlajkaObrazek**) a okrem definičného bodu obsahujú jeden alebo viac ďalších typov atribútu **Geometrie** (generalizované hranice a podobne). To, ktoré z týchto ďalších atribútov sa budú v súboroch VFR nachádzať sa určí pomocou vyhľadávacieho formulára VDP. Tento typ súborov sa oplatí využiť, keď plánujeme ukladať všetky údaje o jednotlivých RÚIAN prvkoch alebo plánujeme lokalizovať celé územie jednotlivých prvkov RÚIAN pomocou polygónov (napríklad pomocou generalizovaných hraníc).

2.7.3 Rozdelenie podľa časovej platnosti

Na základe tohto rozdelenia možno prvky VFR rozdelit' na tieto typy súborov:

- **Platné súbory VFR** - jedná sa o prvky RÚIAN, ktoré nemajú určený atribút **PlatiDo**, ktorý určuje koniec platnosti prvku RÚIAN. Vo väčšine prípadov sa využíva vždy tento typ súborov VFR.
- **Historické súbory VFR** - tieto súbory VFR obsahujú prvky RÚIAN, ktoré už nie sú platné a boli vyradené z databázy platných prvkov RÚIAN. Súbory sa odlišujú od platných prvkov RÚIAN atribútom **PlatiDo** určujúcim koniec ich platnosti.

2.7.4 Rozdelenie podľa pokrytia prvkov RÚIAN

Toto kritérium bolo zámerne uvedené pod vyššie uvedenými kritériami, nakoľko určuje, či sa v súboroch VFR budú nachádzať všetky prvky RÚIAN zvolené podľa vyššie uvedených kritérií alebo len niektoré. Kritérium je teda na najvyššej vrstve a obaluje všetky ostatné kritéria. Na základe tohto kritéria možno rozdeliť prvky VFR na tieto typy súborov:

- **Mesačné stavové súbory** - Obsahujú úplne všetky prvky RÚIAN zvolené podľa vyššie uvedených kritérií. Tieto typy súborov sa generujú vždy posledný deň v mesiaci a sú k dispozícii nasledujúci deň (prvý deň v nasledujúcom mesiaci). Každý mesiac je teda vygenerovaný vždy práve jeden súbor tohto typu. Tento typ súborov sa oplatí využiť v prípade, keď plánujeme exportovať všetky prvky RÚIAN spadajúce pod vyššie uvedené kritéria (napríklad ak vytvárame novú databázu súborov základnej dátovej sady).
- **Denné zmenové súbory** - Tieto typy súborov môžu byť generované každý deň a majú rovnakú štruktúru ako mesačné stavové súbory. Generovanie v danom dni závisí na podmienke, či sú v danom dni nejaké nové zmeny oproti poslednému mesačnému stavovému súboru (nikdy sa nevytvára prázdny zmenový súbor). Nový mesačný stavový súbor (generovaný v posledný deň aktuálneho mesiaca) sa tvorí z posledného stavového súboru predchádzajúceho mesiaca doplneného o prvky RÚIAN, ktoré sa nachádzajú v denných zmenových súboroch za aktuálny mesiac - tento typ súborov teda tvorí dôležitú súčasť pri tvorbe mesačných stavových súborov. Oproti mesačným stavovým súborom však má niekoľko nedostatkov:
 - Pokiaľ dôjde k zmene akéhokoľvek atribútu prvku RÚIAN, tak do zmenového súboru sa dostane vždy celý tento prvok RÚIAN so všetkými jeho atribútmi (generujú sa teda aj nezmenené atribúty, ktoré generované byť nemuseli).
 - Najmenšou časovou jednotkou, ktorá sa zaznamenáva u zmenových súborov je jeden deň. To má za následok to, že pokiaľ dôjde v jednom dni k viac zmenám (napríklad založenie prvku, zmena prvku, zrušenie prvku) u nejakého prvku RÚIAN, tak sa do zmenového súboru dostane iba posledná vykonaná zmena (napríklad ak v jednom dni prvok zrušíme aj založíme, tak sa vykoná iba jeho založenie - bude duplikovaný).
 - Zmenové súbory sa negenerujú pre súbory typu obec a jej podradené prvky - nie je možné teda pravidelne aktualizovať tieto územné celky.

Tento typ súborov sa oplatí využiť, pokiaľ plánujeme aktualizovať už existujúcu databázu prvkov RÚIAN.

2.8 Špeciálny VFR

Všetky doteraz uvedené informácie o VFR sa týkali štandardného VFR. Od verzie VFR 1.8 však bol pridaný ku štandardnému VFR aj špeciálny VFR, ktorý sa delí na dva typy súborov:

- **Súbory číselníkov** - Obsahujú dodatočné informácie k prvkom RÚIAN. Každý súbor číselníkov sa vždy skladá zo šiestich základných atribútov: kód, názov, skratka, popis, začiatok platnosti a koniec platnosti. Na súbory číselníkov odkazujú prvky

štandardného VFR (napríklad prvok **Zsj** a jeho atribút **CharakterZsjKod**, ktorý odkazuje na kolekciu údajov **CharakterZsj** v číselníku).

- **Súbory volebných okrskov** - tento typ súborov obsahuje kolekciu údajov **VolebneOkrsky**, ktorá obsahuje všetky prvky RÚIAN s názvom **VolebniOkrsek**. Jedná sa o jedinú kolekciu prvkov RÚIAN, ktorá nie je súčasťou štandardného VFR.

Pokiaľ uvažujeme rozdelenie typov súborov VFR na základe kritérií uvedených v podkapitole 2.7, tak súbory špeciálneho VFR sa dá rozdeliť iba na základe rozdelenia podľa pokrytia prvkov RÚIAN, pričom súbory volebných okrskov obsahujú pri tomto kritériu iba typ mesačných stavových súborov (generujú sa iba raz mesačne - vždy tretí deň v mesiaci).

2.9 Možnosti získania súborov VFR

Aktuálne je možné súbory VFR získať pomocou dvoch spôsobov:

- Pomocou aplikácie VDP.
- Pomocou webových služieb ISZR.

2.9.1 Získanie súborov VFR z aplikácie verejného diaľkového prístupu

Táto funkcionálnosť aplikácie VDP bola využitá z dôvodu získania potrebných URL, na ktorých sa nachádzajú súbory VFR s dátami RÚIAN pre navrhovanú aplikáciu, ktorá bude k týmto súborom pristupovať pomocou získaných URL. Do hlavného formuláru pre vyhľadávanie súborov štandardného VFR sa dostaneme po kliknutí na tlačítko **Výmenný formát** nachádzajúce sa na úvodnej obrazovke aplikácie VDP.

Formulár pre vyhľadávanie súborov štandardného VFR je možné prepnúť na formulár pre vyhľadávanie súborov špeciálneho VFR a opačne pomocou navigačného menu **Výmenný formát**, ktoré sa nachádza v ľavej časti stránky. Vo formulári pre vyhľadávanie súborov štandardného VFR je možné nastaviť tieto vyhľadávacie kritéria:

- platnosť údajov (**platné** alebo **historické**)
- časový rozsah (**úplna kópia** alebo **zmenové súbory od určitého dátumu**)
- územné prvky (**štát** alebo **obec a podradené** - samostatné súbory obcí)
- datová sada (**základná** alebo **kompletná**)
- výber z údajov (**základné údaje**, **generalizované hranice**, **originálne hranice a vlajky a znaky**)
- územné obmedzenie (**celá ČR**, **len konkrétny kraj**, **len konkrétna obec s rozšírenou pôsobnosťou** a **len konkrétna obec**)

V prípade, že je za územný prvok zvolený štát a súčasne je zvolená kompletná datová sada, tak je potrebné vo výbere z údajov zvoliť aspoň jednu hodnotu okrem hodnoty základné údaje. Formulár pre vyhľadávanie súborov špeciálneho VFR obsahuje iba položky časový rozsah a výber z údajov (**číselníky** a **volebné okresy**). Po kliknutí na tlačítko vyhľadať vo formulári sa pod formulárom zobrazí tabuľka s výsledkami, ktoré spĺňajú zadané vyhľadávacie kritéria. Z tohto zoznamu výsledkov je možné vyčítať URL adresy, ktoré sú potrebné

pre navrhovanú aplikáciu (adresy sú uložené pod ikonou diskety v poslednom stĺpci tabuľky (**uložiť**)). Aplikácia VDP bola teda z pohľadu projektu využitá ako pomôcka pre získanie týchto URL adries. Podrobnému popisu týchto URL adries sa venuje podkapitola 2.10. V nasledujúcom obrázku je uvedený príklad formulára s vyplnenými vyhľadávacími kritériami a tabuľkou výsledkov pre dané vyhľadávacie kritéria.

Výměnný formát Nové zadání

Platnost údajů: Platné Historické

Časový rozsah: Úplná kopie Přírůsteky od data: 10.04.2019

Územní prvky: Stát až ZSJ Obec a podřazené

Datová sada: Základní Kompletní

Výběr z údajů: Základní údaje Gen. hranice Originální hranice Vlajky a znaky

Územní omezení: ČR Kraj (VÚSC): nevybráno

Obec s rozšířenou působností (ORP): nevybráno

Obec (kód):

Seznam linků Vyhledat

Obec	Platnost údajů	Výběr z údajů	Název souboru	Velikost souboru [MB]	Uložit
	Platné	Základní	20190331_ST_UZSZ.xml.zip	4,30	
	Platné	Základní	20190301_ST_UZSZ.xml.zip	4,30	
	Platné	Základní	20190131_ST_UZSZ.xml.zip	4,27	

< Předchozí Strana: 1 Další > Celkem záznamů: 3

Obr. 2.1: Hlavný formulár pre vyhľadávanie súborov štandardného VFR spolu s tabuľkou výsledkov.

2.9.2 Získanie súborov VFR pomocou webových služieb registru územných adries a nehnuteľností

Pri tomto spôsobe sú URL, na ktorých sa nachádzajú súbory VFR získané pomocou webových služieb RÚIAN. Tieto služby sú poskytované na eGON rozhraní ISZR. Aktuálne sú poskytované dva typy týchto webových služieb:

- **E39 - ruianSouboryZmen** - služba vracia zoznam súborov VFR zadaného typu, ktoré obsahujú zmenové súbory zvoleného prvku RÚIAN od zvoleného dátumu do súčasnosti. Dátum, od ktorého sú zmeny požadované, nesmie byť starší než tri mesiace. V prípade, že je zadaný typ súboru, ktorý nie je zmenovým typom, tak bude ako odpoveď vrátený prázdny XML súbor.
- **E40 - ruianSouboryDat** - Služba vracia zoznam súborov VFR zadaného typu, ktoré obsahujú všetky údaje zvoleného prvku RÚIAN k poslednému dňu minulého mesiaca. V prípade, že je zadaný typ súboru, ktorý je zmenovým typom, tak bude ako odpoveď vrátený prázdny XML súbor.

Tieto webové služby sú dostupné iba orgánom štátnej správy a samosprávy. Nakoľko tento spôsob získania súborov VFR nie je voľne prístupný širokej verejnosti, tak bol pre účely projektu jasne zvolený spôsob získania súborov VFR pomocou aplikácie VDP, ktorá je voľne prístupná širokej verejnosti.

2.10 Štruktúra URL adries získaných pomocou aplikácie VDP

Na základe experimentovania s aplikáciou VDP založenom na vyplňovaní rôznych kombinácií v hlavnom formulári pre vyhľadávanie súborov VFR bolo zistené, že kostru verejných URL adries, na ktorých sa nachádzajú súbory VFR, tvorí vždy tento základný tvar: `vdp.cuzk.cz/a/b_c_defg.xml.i`, pričom význam jednotlivých zvýraznených neznámych je nasledujúci:

- **a** je u štandardného VFR typ súboru podľa kritéria časovej platnosti - **soucasna** pre platné súbory a **historicka** pre historické súbory. U špeciálneho VFR má táto neznáma v URL vždy pevnú hodnotu - **specialni**
- **b** je formát dátumu v tvare **YYYYMMDD** (deň, mesiac, rok)
- **c** je typ súboru podľa kritéria obsahu prvkov RÚIAN - **ST** pre súbory typu štát a **OB_kód-obce** pre súbory typu obec a podradené. U špeciálneho VFR má táto neznáma v URL vždy pevnú hodnotu **ST**
- **d** je typ súboru podľa pokrytia prvkov RÚIAN - **U** pre mesačné stavové súbory a **Z** pre denné zmenové súbory.
- **i** je typ archívu - gz alebo zip.

Neznáme **e**, **f** a **g** majú tvoria vždy u súborov špeciálneho VFR nasledujúce hodnoty:

- **CIS** pre súbory číselníkov
- **VOH** pre súbory volebných okrskov.

Naopak u súborov štandardného VFR má každá z týchto neznámych iný význam:

- **e** je typ súboru podľa kritéria obsahu atribútov - **Z** pre súbory základnej dátovej sady a **K** pre súbory kompletnej dátovej sady.
- **f** je typ súboru podľa časovej platnosti - **S** pre platné súbory VFR a **H** pre historické súbory VFR
- **g** obsahuje dodatočné informácie k obsahu atribútov - **Z** - iba atribúty základnej dátovej sady, **G** - atribúty základnej dátovej sady spolu s geometriou generalizovaných hraníc, **H** - základná dátová sada spolu s geometriou originálnych hraníc a **O** - základná dátová sada spolu s obrázkami vlajok a znakov obcí.

2.10.1 Príklady URL adries súborov VFR

V tejto podkapitole sú uvedené typické príklady URL adries na základe ich štruktúry popísanej v kapitole 2.10 :

- `vdp.cuzk.cz/soucasna/20190301_ST_UZSZ.xml.gz` - platný, mesačný stavový súbor základnej dátovej sady s prvkami RÚIAN štát .
- `vdp.cuzk.cz/soucasna/20190315_ST_ZKSG.xml.gz` - platný, denný zmenový súbor kompletnej dátovej sady s geometriou generalizovaných hraníc a prvkami RÚIAN štát .

- vdp.cuzk.cz/soucasna/20190301_OB_500020_ZKSO.xml.gz - platný, mesačný stavový súbor kompletnej dátovej sady s obrázkami vlajok a znakov obcí a prvkami RÚIAN obec a podradené.
- vdp.cuzk.cz/specialni/20190303_ST_UVOH.xml.gz - platný, mesačný stavový súbor kompletnej dátovej sady volebných okrskov.
- vdp.cuzk.cz/specialni/20190303_ST_UCIS.xml.gz - platný, mesačný stavový súbor kompletnej dátovej sady číselníkov

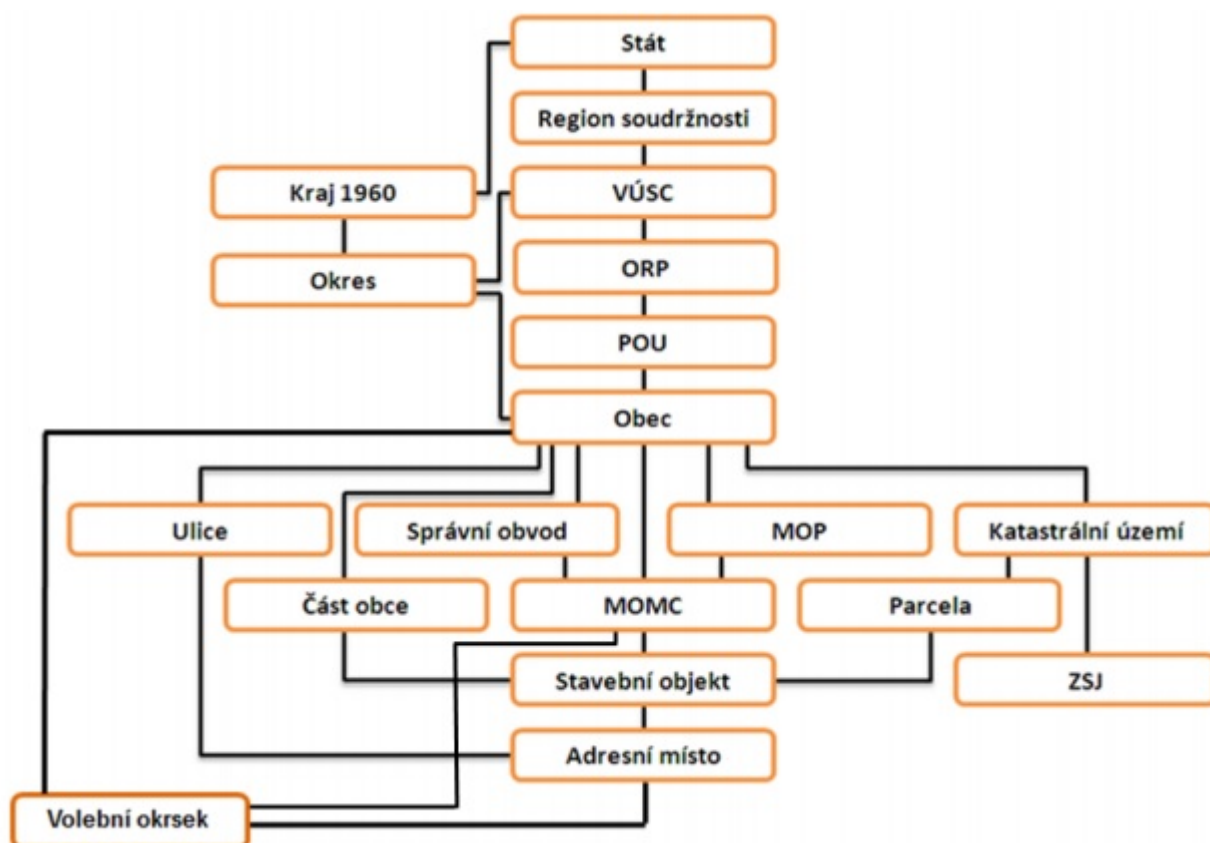
Kapitola 3

Návrh databázového modelu

V tejto kapitole bude popísane, ktoré prvky RÚIAN boli zahrnuté do databázového modelu a budú podrobne vysvetlené jednotlivé časti databázového modelu.

3.1 Zvolenie prvkov RÚIAN a ich atribútov pre export

Na nasledujúcom obrázku sa nachádza zjednodušená schéma (stromová štruktúra) všetkých prvkov vedených v RÚIAN:



Obr. 3.1: Zjednodušená schéma prvkov RÚIAN [15].

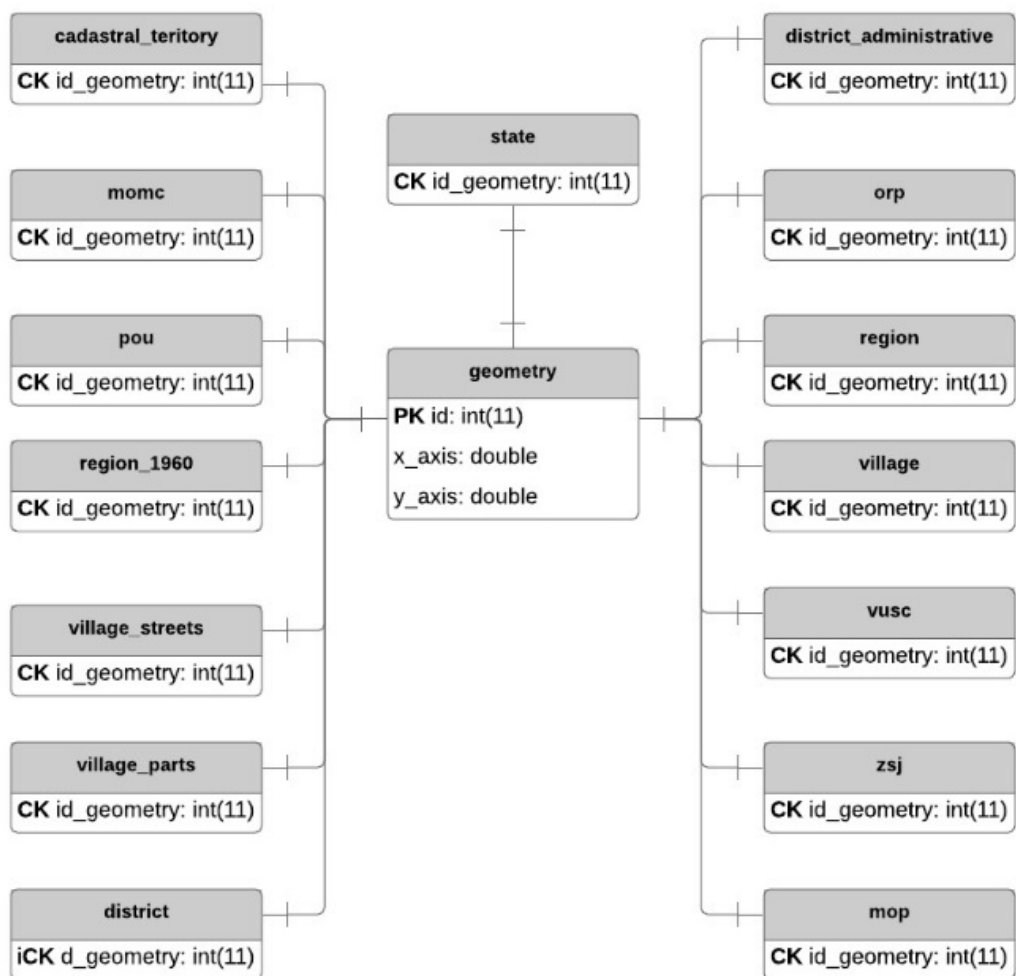
Keďže cieľom bakalárskej práce je exportovať predovšetkým súbory typu štát základnej dátovej sady, tak medzi prvky RÚIAN určené pre návrh nebol zahrnutý prvok špeciálneho VFR (Volební okrsek) a niektoré prvky, ktoré sa nachádzajú v súboroch VFR typu obec a podradené (Parcela, Stavební objekt, Adresní místo). Všetky ostatné prvky RÚIAN, ktoré sú uvedené v obrázku 3.1, boli do návrhu zahrnuté. Čo sa týka atribútov prvkov RÚIAN, tak do návrhu boli zahrnuté všetky atribúty uvedené v tabuľke 2.2 okrem atribútov **NespravneUdaje** (súčasťou návrhu aktuálne nie je kontrola správnosti údajov), **GlobalniIdNavrhuZmeny** (cieľom navrhovanej aplikácie je práca s databázou RÚIAN a toto id sa týka databázy ISÚI) , **RizeniID** (toto id sa týka databázy ISKN) a obrazových atribútov (ZnakObrazek, VlajkaObrazek), ktoré sú súčasťou kompletnej dátovej sady. Z typov geometrie bol do návrhu pre všetky prvky RÚIAN zahrnutý základný typ geometrie okrem prvkov **KatastralniUzemi** a **Obec**, u ktorých boli do návrhu zahrnuté aj generalizované hranice. Originálne hranice neboli u žiadneho prvku RÚIAN do návrhu zahrnuté.

3.2 Databázový model

Navrhnutý databázový model tvorí aktuálne dvadsaťštyri entít a preto bol pre väčšiu prehľadnosť a ľahšie pochopenie rozdelený do niekoľko podkapitol, pričom každá podkapitola popisuje špecifickú časť databázového modelu. Primárny kľúč každej entity tvorí vždy hodnota atribútu **Kod** a jeho veľkosť sa postupne navyšuje, nakoľko tvorí unikátny identifikátor v rámci celej databázy. Cudzie kľúče sú vždy uvedené v tvare **id_<nazov tabuľky>** a ich hodnotu tvorí hodnota primárneho kľúča tabuľky, na ktorú odkazujú. V prvých troch podkapitolách sú popísané časti modelu, ktoré nie sú súčasťou databázového modelu RÚIAN a pre zjednodušenie sú v nich uvedené len atribúty, ktoré sú pre tieto časti nevyhnutné. Posledné dve podkapitoly popisujú časti modelu, ktoré sú súčasťou databázového modelu RÚIAN a udávajú vzťahy medzi jednotlivými prvkami RÚIAN.

3.2.1 Základná geometria

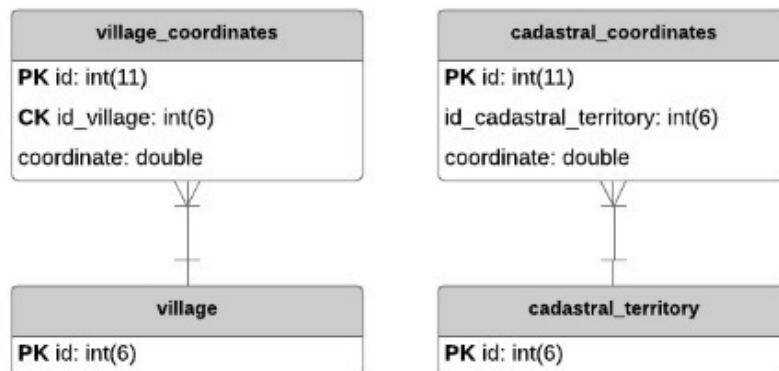
Táto časť databázového modelu slúži pre uloženie definičných bodov všetkých prvkov RÚIAN a jej ER diagram je zobrazený na obrázku 3.2. Túto časť modelu tvorí vždy vzťah **1:1** - každý prvok RÚIAN má práve jeden jedinečný definičný bod v rámci celej databázy. Entita geometrie obsahuje okrem primárneho kľúča atribúty **x_axis** a **y_axis**, ktoré určujú definičný bod príslušného prvku RÚIAN. Táto časť databázového modelu bola zavedená z dôvodu redukcie počtu atribútov v jednotlivých entitách (v databázovom modeli RÚIAN každá entita obsahuje atribúty geometrie, ale v navrhovanom modeli bude vďaka tomuto riešeniu obsahovať atribúty geometrie iba jedna entita - **geometry**).



Obr. 3.2: Entita geometrie a jej vzťahy s entitami prvkov RÚIAN.

3.2.2 Generalizované hranice

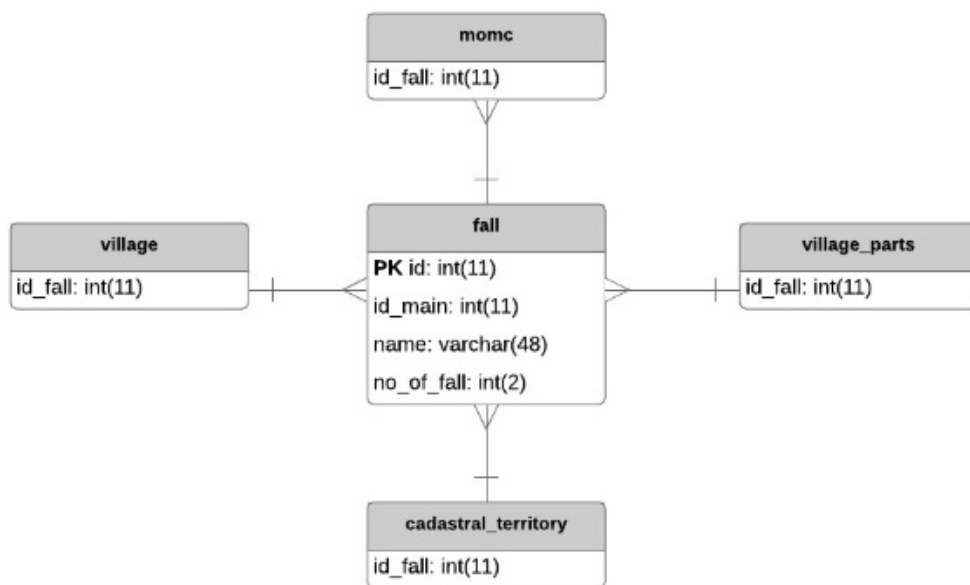
ER diagram tejto časti databázového modelu je uvedený na obrázku 3.3. V tejto časti modelu sa nachádza vždy vzťah **1:N**, pričom na ľavej strane vzťahu sa nachádza prvok RÚIAN a na pravej strane vzťahu všetky súradnice jeho generalizovaných hraníc. V entite generalizovaných hraníc sa nachádza vždy jedna súradnica z kolekcie generalizovaných hraníc a cudzí kľúč, ktorý odkazuje na príslušný prvok RÚIAN. Táto časť databázového modelu tiež nie je súčasťou databázového modelu RÚIAN a bola zavedená z rovnakého dôvodu, ktorý je uvedený v podkapitole 3.2.1 (v tomto prípade každá entita obsahuje jeden atribút generalizovaných hraníc, v ktorom sú jednotlivé súradnice oddelené čiarkou). Oproti entite **geometry** je tu však jedna nevýhoda - každý prvok RÚIAN, u ktorého sa ukladajú generalizované hranice, musí mať vlastnú entitu generalizovaných hraníc z dôvodu ukladania cudzieho kľúča v danej entite. Aktuálne sa ukladajú generalizované hranice pre entity **village** a **cadastral_territory** - entity **cadastral_coordinates** a **village_coordinates**. Ukladanie generalizovaných hraníc je do budúcnosti možné rozšíriť o ďalšie entity.



Obr. 3.3: Entity generalizovaných hraníc spolu s entitami, ktoré ich využívajú.

3.2.3 Skloňovacie pády

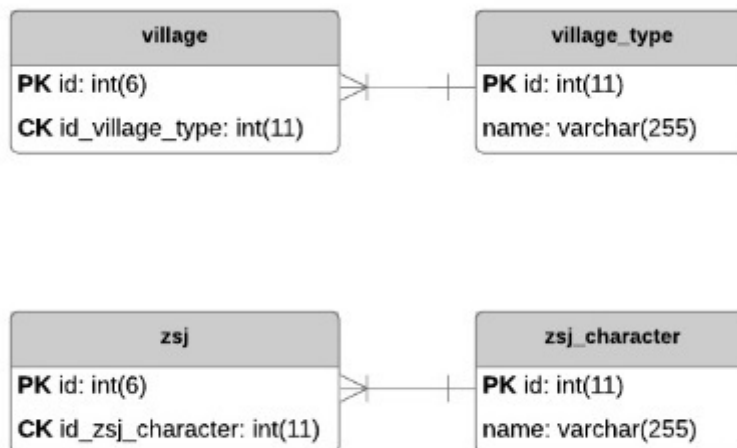
Časť databázového modelu zabezpečuje uloženie skloňovacích pádov pre prvky RÚIAN, ktoré ich obsahujú. Návrh tejto časti modelu je uvedený na obrázku 3.4. Vzťahy medzi jednotlivými entitami sú vždy typu **1:N**, pričom na ľavej strane vzťahu je prvok RÚIAN a na pravej strane vzťahu jeho skloňovacie pády. V entite skloňovacích pádov **fall** sa nachádza atribút **id_main**, ktorý vždy odkazuje na úvodný druhý pád prvku RÚIAN a všetky entity prvku RÚIAN obsahujú cudzí kľúč **id_fall**, ktorý takisto odkazuje na druhý pád v entite **fall**. Pomocou entity prvku RÚIAN je teda možné získať vždy iba druhý pád pre túto entitu. Následne je potrebné z entity **fall** získať všetky ostatné pády pomocou atribútu **id_main**. Toto riešenie umožňuje mať iba jednu tabuľku skloňovacích pádov pre všetky prvky RÚIAN, ktoré obsahujú skloňovacie pády. Atribút **no_of_fall** udáva číslo pádu podľa českého skloňovania [1]. Databáza RÚIAN ukladá skloňovacie pády vždy priamo v tabuľke prvku RÚIAN a preto až štyri tabuľky databázy RÚIAN obsahujú skloňovacie pády. V tomto návrhu obsahuje skloňovacie pády iba jedna tabuľka a sú v ňom zahrnuté všetky entity, ktoré skloňovacie pády využívajú.



Obr. 3.4: Návrh pre ukládanie skloňovacích pádov.

3.2.4 Súbory číselníkov

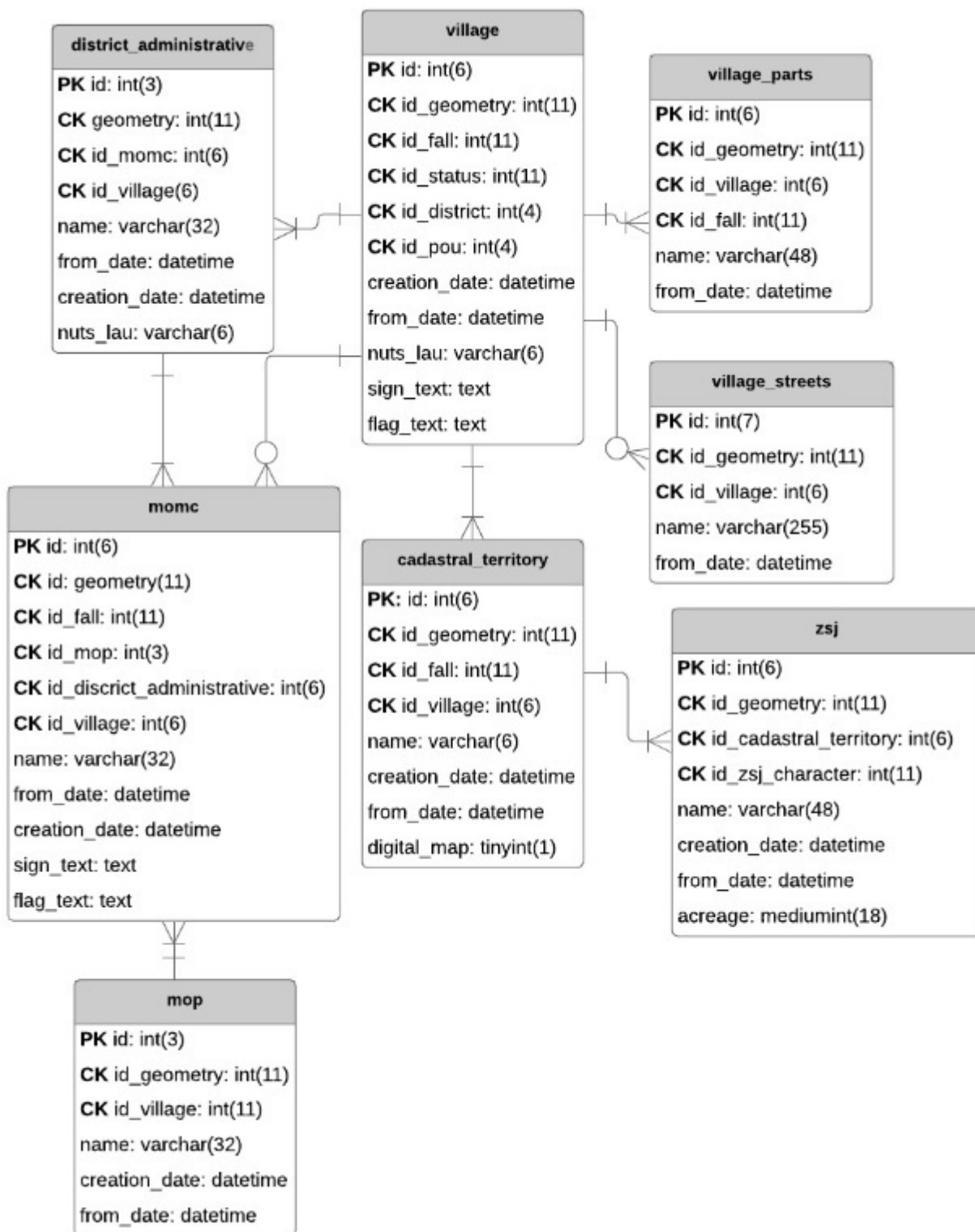
Návrh tejto časti databázového modelu je uvedený na obrázku 3.5. Dva prvky RÚIAN, ktoré sú súčasťou návrhu databázového modelu (**Obec** a **Zsj**) obsahujú atribúty (**CharakterZsjKod** a **StatusKod**), ktoré odkazujú na súbory číselníkov (v súboroch číselníkov sa pod touto číselnou hodnotou nachádza názov charakteru Zsj respektíve typu obce). Z tohto dôvodu boli do návrhu zahrnuté entity **village_type** a **zsj_character**, ktoré uchovávajú potrebné názvy zo súborov číselníkov a nie je tak potrebné vyhľadávať tieto názvy v daných súboroch.



Obr. 3.5: Návrh pre ukládanie súborov číselníkov.

3.2.5 Prvá časť prvkov RÚIAN

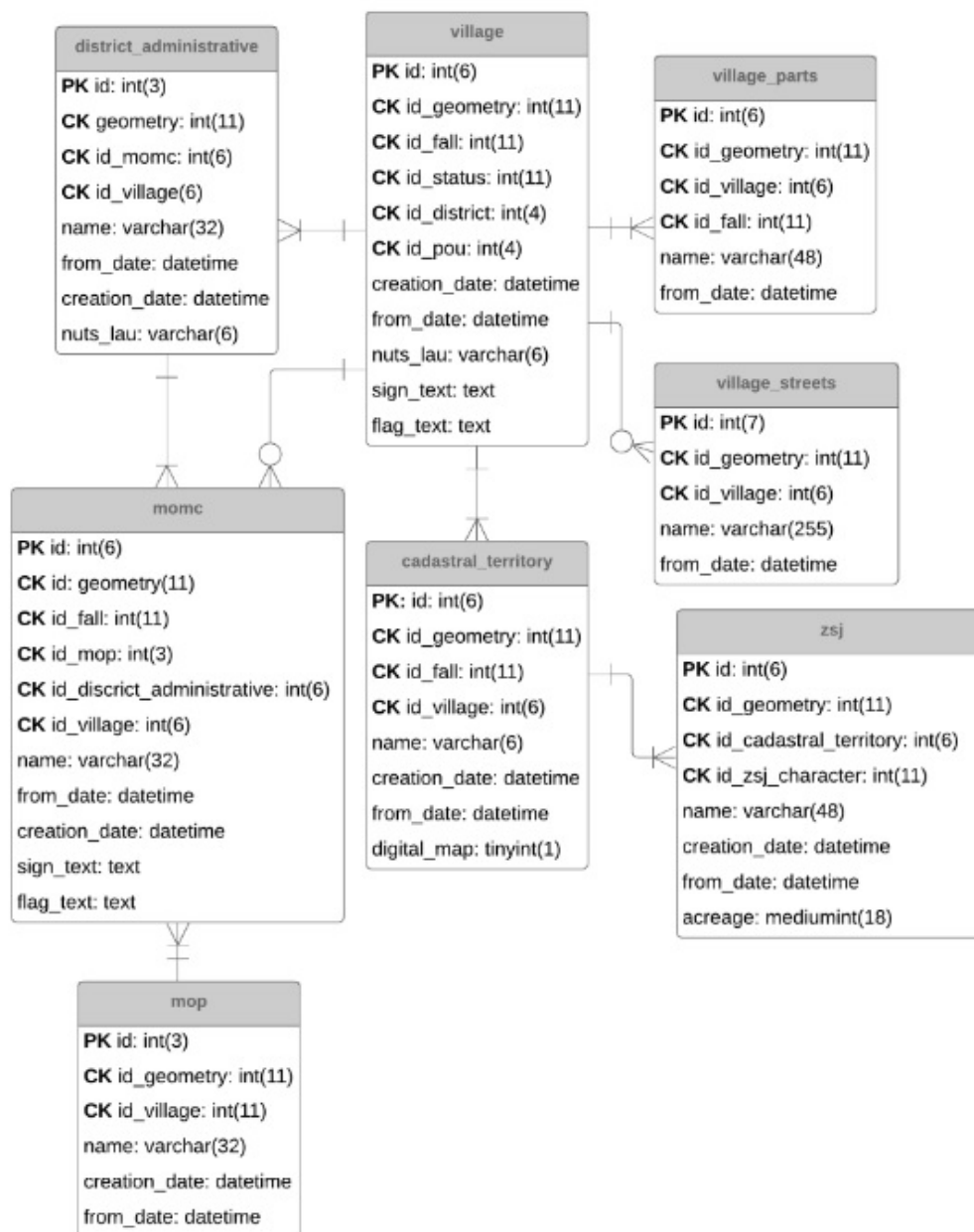
ER diagram tejto časti modelu je zobrazený na obrázku 3.6. Táto časť databázového modelu je vo väčšine prípadov tvorená vzťahom **1:N**, kde na ľavej strane vzťahu je nadradená entita (nadradený väčší územný - napríklad **state**) a na pravej strane vzťahu podradená entita (podradený menší územný celok - napríklad **region_1960**). Jedná sa o reťazové odkazovanie na väčšie nadradené územné celky (obec odkazuje na nadradenú entitu pou, následne pou odkazuje na nadradenú entitu orp, následne orp odkazuje na nadradenú entitu vusc, následne vusc odkazuje na nadradenú entitu region a na záver región odkazuje na nadradený štát). Okrem vzťahu 1:N sa v tejto časti ešte používa vzťah **1:0..1**, ktorý sa týka entít **pou**, **orp** a **village**, kde entity **orp** a **pou** sú obce, ale neobsahujú všetky informácie, ktoré možno o obciach získať. Preto obsahujú cudzí kľúč odkazujúci na entitu **village**, aby bolo možné o nich získať detailnejšie informácie.



Obr. 3.6: Prvá část prvků RÚIAN

3.2.6 Druhá časť prvkov RÚIAN

ER diagram tejto časti modelu je zobrazený na obrázku 3.7



Obr. 3.7: Druhá časť prvkov RÚIAN.

Táto časť databázového modelu sa týka územného celku obce a jej podradených územných celkov. Vo väčšine prípadov sa v tejto časti používa medzi entitami vzťah **1:N**, kde na ľavej strane sa nachádza entita **village** a na pravej strane jej podradený územný celok (niekedy je použitý vzťah **1:0..N** - napríklad vzťah medzi entitami **village** a **momc**, nakoľko niektoré obce vôbec nemajú mestský obvod).

Kapitola 4

Implementácia nástroja pre spracovanie prvkov RÚIAN

Implementácia nástroja pre spracovanie prvkov VFR a ich uloženie do databázy je rozdelená na niekoľko dôležitých častí. Budú vysvetlené všetky súčasti implementovaného nástroja od použitých jazykov až po implementáciu databázy.

4.1 Zvolenie implementačných nástrojov

Implementovaný nástroj by mal jednorázovo vykonať potrebné operácie a následne ukončiť svoju činnosť. Z tohto dôvodu bolo rozhodnuté, že navrhovaný nástroj bude implementovaný v jazyku PHP.

Pre implementovanie databázy bol využitý z dôvodu bezproblémovej kompatibility na rôznych operačných systémoch a bezplatnej dostupnosti MySQL systém riadenia bázy dat.

Na základe týchto zvolených nástrojov je navrhovaný nástroj vlastne PHP skript pracujúci s MySQL databázou.

4.2 Štruktúra skriptu

Skript je rozdelený na niekoľko modelov a jednu pomocnú triedu. Každý z týchto modelov spolu s pomocnou triedou tvorí samostatný PHP súbor:

- **en.php** - pomocná trieda, ktorá obsahuje statické pole. Toto pole slúži pre preklad názvov prvkov a kolekcii súborov VFR do anglických názvov, ktoré sa využívajú v implementovanej databáze (kľúč tvorí anglický názov a hodnota tvorí český názov v súboroch VFR). Trieda sa nachádza v zložke languages.
- **Language.php** - model, ktorý obsahuje metódy pre prácu s pomocnou triedou **en.php**.
- **XmlRuian.php** - model, ktorý slúži pre spracovanie súborov VFR - získa a roztriedi z nich dáta tak, aby boli pripravené pre vloženie do databázy
- **RuianDatabase.php** - model, ktorý sa stará o operácie nad databázou - vykonáva vkladanie, odstraňovanie a aktualizovanie prvkov v implementovanej databáze.
- **Ruian.php** - tento model tvorí jadro skriptu - začne vykonávať svoju činnosť ihneď po spustení skriptu. Model sa stará sa o spracovanie argumentov príkazového riadku a riadenie činnosti ostatných modelov.

4.3 Spracovanie argumentov príkazového riadku VFR

Po spustení skriptu prebehne v modeli **Ruian.php** inicializácia ostatných modelov a kontrola argumentov príkazového riadku. Na základe argumentov príkazového riadku sa môže skript dostať do jedného z týchto dvoch stavov:

- Pokiaľ je skript spustený bez parametrov, tak prebehne zmazanie všetkých záznamov aktuálnej databázy a budú sa spracovávať **mesačné stavové súbory** VFR. Získane dáta zo súborov sa uložia do prázdnej databázy.
- Pokiaľ je skript spustený s argumentom **update**, tak prebehne spracovanie **denných zmenových súborov**, ktorých spracované dáta sa následne použijú na vytvorenie nových alebo aktualizáciu existujúcich záznamov v databáze.

4.3.1 Získanie URL adres a extrakcia súborov VFR z archívov

Po spracovaní argumentov príkazového riadku predá riadiaci model **Ruian.php** riadenie skriptu modelu **XmlRuian.php**. Tento model následne nastaví potrebné dátumové konštanty (prvý deň predchádzajúceho mesiaca pre mesačné stavové súbory a tretí deň predchádzajúceho mesiaca pre súbory číselníkov). Pri spracovaní denných zmenových súborov sa v cykle kontroluje výskyt zmenových súborov od dátumu poslednej aktualizácie, ktorý je uložený v pomocnej tabuľke **logs** až po dnešný dátum. Každá dátumová konštantka obsahuje niekoľko fáz kontroly, ktorá preverí existenciu URL adresy pre daný dátum:

- **kontrola prípony archívu** - táto fáza sa vykonáva pri každej dátumovej konštantke. Kontrola najprv skontroluje, či existuje URL adresa s príponou súboru **.gz**. V prípade neúspechu sa následne kontroluje existencia URL adresy s príponou **.zip**.
- **zmena dátumovej konštanty** - táto fáza sa vykoná iba v prípade neúspechu predošlej fázy a týka sa iba mesačných stavových súborov a súborov číselníkov. Fáza zmení v prípade neúspechu prvej fázy dátumovú konštantu pre mesačný stavový súbor na prvý deň aktuálneho mesiaca, nakoľko bolo občas narazené na takéto prípady vygenerovania súborov. V prípade súborov číselníkov sa dátumová konštantka upraví na tretí deň aktuálneho mesiaca. Po úprave dátumových konštant sa celá kontrola opakuje.

Pokiaľ bola nájdená URL adresa pre zadanú dátumovú konštantu, tak prebehne uloženie archívu na disk. Následne sa vykoná extrakcia XML súboru z archívu (každý archív obsahuje práve jeden XML súbor). Subory XML sa ukladajú vždy do týchto zložiek v koreňovom adresári:

- **files/vdp/base** pre mesačné stavové súbory kompletnej dátovej sady typu štát až ZSJ.
- **files/vdp/villages** pre mesačné stavové súbory základnej dátovej sady typu obec a podradené.
- **files/vdp/updates** pre denné zmenové súbory kompletnej dátovej sady
- **files/vdp/dial** pre súbory číselníkov.

Samostatná zložka **files** slúži pre dočasné ukladanie stiahnutých archívov.

4.4 Spracovanie extrahovaných XML súborov

Z mesačného stavového súboru typu štát až zsj je možné získať všetky potrebné prvky VFR okrem prvku **Ulice**, ktorý sa nachádza v mesačných stavových súboroch typu obec a podradené. Keďže pre každú obec existuje samostatný súbor s jej podradenými prvkami, tak je potrebné v prípade vytvárania novej databázy (denné zmenové súbory neobsahujú podradené prvky obce) vždy stiahnuť pre každú obec jej samostatný XML súbor (približne 6000 súborov), čo výrazne spomaľuje vykonávanie skriptu.

Model **XmlRuian** obsahuje pre každý prvok VFR deklarované multidimenzionálne globálne pole, ktoré popisuje, čo všetko sa bude z daného prvku VFR extrahovať do databázy. Príklad takéhoto poľa pre prvok **Stat**:

```
private $state_collection = array('vf:Staty', 'vf:Stat',  
    array('sti:Kod', 'sti:Nazev', 'sti:PlatiOd', 'sti:NutsLau',  
        'sti:DatumVzniku'), array());
```

Prvý index v poli obsahuje názov nadradenej kolekcie, ktorá obaluje všetky prvky VFR s názvom, ktorý je uvedený v druhom indexe. Tretí index obsahuje pole názvov atribútov prvku **Stat**, ktoré nie sú kolekciami. Posledný štvrtý index obsahuje pole názvov atribútov prvku **Stat**, ktoré sú kolekciami (okrem kolekcie geometrie, ktorá sa spracúva samostatne a extrahuje sa povinne u každého prvku VFR). Tieto globálne polia je možné kedykoľvek upraviť a zmeniť tak dáta, ktoré budú zahrnuté do databázy.

Na začiatku spracovania XML súborov je deklarované pole **all**, ktoré určuje, ktoré všetky prvky VFR sa budú extrahovať do databázy. Ak sa bude prvok VFR extrahovať do databázy, tak jeho multidimenzionálne globálne pole bude zahrnuté do poľa **all**. Aktuálne pole **all** obsahuje všetky multidimenzionálne polia prvkov VFR, ktoré sú deklarované v modeli **XmlRuian.php**.

Samotné spracovanie XML súborov prebieha pre každý samostatný XML súbor v cykle pomocou php rozšírenia **XmlReader**. Toto rozšírenie bolo vybraté z dôvodu rýchlejšieho spracovania XML súborov (narozdiel od niektorých iných nástrojov pre spracovanie XML tento nástroj neukladá celý obsah XML súboru do pamäte počítača, ale načíta potrebné dáta priamo z XML súboru, vďaka čomu prebieha spracovanie XML súboru rýchlejšie a je menej náročné pre pamäť počítača). Počas spracovania XML súboru sa získavané atribúty ukladajú do multidimenzionálneho poľa **ruianStructure**, ktorého kľúče prvej úrovne sú názvy tabuliek databázy (napríklad **Stat**). Tieto kľúče obsahujú ďalšie pole s číselným indexovaním ([0], [1], [2]...), kde každý takýto číselný index predstavuje jeden záznam konkrétnej tabuľky a je tvorený ďalším poľom, ktorého kľúče sú názvy stĺpcov danej tabuľky a obsahujú hodnotu priradenú danému stĺpcu tabuľky (napríklad ['name'] => 'Brno'). Po kompletnom naplnení poľa **RuianStructure** (pre denné zmenové súbory sa jedná o pole **RuianStructuresFromUpdate**, ktoré obsahuje pre každý nájdený zmenový súbor samostatné pole **RuianStructure**) sa ukončí činnosť modelu **XmlRuian.php**, ktorý predá toto pole riadiacemu modelu **Ruian.php**.

4.5 Práca s databázou

Prácu s databázou vykonáva databázový model **RuianDatabase.php**. V tomto modeli sú definované konštanty (**HOST**, **DBNAME**, **USER** a **PASS**), ktoré slúžia pre úvodnú inicializáciu databázy a je možné ich kedykoľvek upraviť v závislosti na používanom databázovom serveri. Databázový model vykonáva prístup k databáze a dotazy nad ňou pomocou

PDO inštancie, ktorá je rozšíriteľnou súčasťou knižnice pre PHP. Spojenie s databázou sa vytvorí pri úvodnom vytvorení inštancie databázového modelu.

Ukladanie obsahu multidimenzionálneho poľa do databázy je rozdelené do niekoľkých fáz:

- **príprava dát pre uloženie do databázy** - v tejto fáze riadiaci model **Ruian.php** pre každý kľúč prvej úrovne poľa **RuianStructure** zavolá metódu **prepareData**, ktorá hodnoty pod daným kľúčom upraví pre konečné uloženie do databázy - uloží predčasne do databázy atribúty pod daným kľúčom, ktoré sú kolekciami prvkov a majú v databáze vytvorenú samostatnú tabuľku (generalizované hranice, skloňovacie pády). Upravené hodnoty sa nachádzajú v novom poli **data_to_insert**. Kľúče tohto poľa sú preložené kľúče poľa **RuianStructure**, v ktorom názvy kľúčov odpovedajú názvom prvkov a ich atribútov v súboroch VFR. Preklad kľúčov poľa **RuianStructure** prebieha pomocou modelu **Language.php**. Preložené kľúče poľa **database_to_insert** odpovedajú názvom tabuliek a atribútov v implemetovanej databáze.
- **uloženie dát do databázy** - v tejto fáze sa pre každý prvok poľa **data_to_insert** zavolá v riadiacom modeli **Ruian.php** metóda **insert** (pokiaľ záznam s daným prvkom už v databázi existuje, tak sa zavolá metóda **update**) databázového modelu **RuianDatabase.php**. Táto metóda vykoná uloženie tohto prvku do databázy. Jeden prvok poľa **data_to_insert** predstavuje jeden záznam konkrétnej tabuľky a pole **data_to_insert** predstavuje celkový obsah tejto tabuľky.

Kapitola 5

Testovanie databázy

V tejto kapitole sa nachádzajú dotazy, ktoré boli využité pre otestovanie implementovanej relačnej databázy. Jedná sa o ukázkové dotazy, ktoré slúžia ako názorná ukážka práce s implementovanou relačnou databázou. Všetky dotazy sú uvedené v jazyku SQL a boli zadané v nástroji phpMyAdmin. Pod každým dotazom je uvedený podrobnejší popis daného dotazu.

Prvý dotaz - Počet obcí v Českej Republike.

```
SELECT COUNT(*) FROM village WHERE 'id\_status' = '2'
```

Tento dotaz patrí medzi najjednoduchšie dotazy nad vytvorenou databázou. Dotaz pracuje iba s jednou tabuľkou (tabuľka **village**) a jej jedným atribútom (**id_status**), ktorý je použitý vo **where** klauzule. Dotaz zobrazí počet záznamov v tabuľke **village**, ktoré sú typu obce (jedná sa o jeden zo šiest typov). Pokiaľ chceme zistiť počet záznamov, ktoré sú iného typu tak zmeníme hodnotu atribútu **id_status** na hodnotu atribútu **id** z tabuľky **village_type** tohto typu (napríklad hodnota **id_status** bude 4 pre zobrazenie počtu statúárnych miest v Českej Republike).

Druhý dotaz - Vypísanie súradníc polygónu, ktorý tvorí územie konkrétneho katastrálneho územia.

```
SELECT cadastral_coordinates.coordinate FROM cadastral_coordinates  
INNER  
JOIN cadastral_territory ON  
cadastral_territory.id=cadastral_coordinates.id_cadastral_territory  
WHERE cadastral_territory.name = 'Mesto Brno'
```

Oproti predchádzajúcemu dotazu je tento dotaz už o niečo zložitejší, nakoľko pracuje s dvoma tabuľkami a využíva klauzulu **INNER JOIN**. Tento konkrétny dotaz vypíše všetky súradnice katastrálneho územia mesta Brna. Tieto súradnice tvoria polygón, ktorý na mape zobrazuje územie tohto katastrálneho územia. Súradnice sa dajú využiť v rôznych mapových klientoch pre vykresľovanie súradníc konkrétneho katastrálneho územia v mape. Dotaz využíva dve tabuľky (**cadastral_territory** a **cadastral_coordinates**), ktoré sú medzi sebou prepojené na základe atribútov **id_cadastral_territory** a **id** pomocou klauzule **INNER JOIN**. Zároveň dotaz obsahuje klauzulu **WHERE**, ktorá udáva názov katastrálneho územia, pre ktoré chceme získať súradnice polygónu.

Tretí dotaz - Výpis **x**-ovej a **y**-ovej súradnice definičného bodu ľubovoľného prvku.

```
SELECT geometry.x_axis, geometry.y_axis FROM geometry
INNER
JOIN mop ON mop.id_geometry=geometry.id
WHERE mop.name = 'Praha 2'
```

Tento typ dotazu možno aplikovať na tabuľku akéhokoľvek územného celku, nakoľko každá takáto tabuľka obsahuje atribút **id_geometry**, ktorý odkazuje na príslušný záznam v tabuľke **geometry**. Tento konkrétny dotaz slúži pre vypísanie **x**-ovej a **y**-ovej súradnice definičného bodu mestského obvodu v Prahe s názvom Praha 2. Dotaz využíva tabuľku **mop** a tabuľku **geometry**, ktoré sú medzi sebou prepojené na základe atribútov **id_geometry** a **id** pomocou klauzule **INNER JOIN**. Klauzula **where** určuje názov mestského obvodu v Prahe. Pokiaľ budeme chcieť vyhľadať súradnice definičného bodu v inej tabuľke na základe názvu územného celku tejto tabuľky, tak postačí zmeniť iba obsah klauzule **FROM** (napríklad na **'village'** pre vyhľadávanie definičných bodov obcí). Tieto definičné body môžeme rovnako ako v prípade súradníc polygónov využiť v rôznych mapových klientoch pre zobrazenie definičného bodu na mape. Súradnice je tiež možné použiť v **Marushke** klikneme na **Hledat** v hornej časti pravej strany klienta a následne zvolíme možnosť **Najdi souřadnice v S-JTSK**. Po zvolení tejto možnosti sa zobrazí formulár, do ktorého zadáme súradnice získané z databázového dotazu a formulár odošleme kliknutím na tlačítko **Hledat**. Po odoslaní formuláru **Marushka** zobrazí na mape definičný bod daného územného celku.

Štvrtý dotaz - výpis skloňovacích pádov konkrétnej obce

```
SELECT fall.name, fall.id_fall_name FROM fall
INNER JOIN village ON fall.id_main = village.id_fall
WHERE village.name = 'Blatec'
```

Tento dotaz vypíše druhý až šiesty skloňovací pád konkrétnej obce. V tomto príklade sa jedná o obec Blatec. Dotaz využíva tabuľky **fall** a **village**, ktoré sú medzi sebou prepojené pomocou klauzule **INNER JOIN** na základe atribútov **id_main** a **id_fall**. Klauzula **WHERE** určuje názov obce, z ktorej chceme zobraziť skloňovacie pády.

Piaty dotaz - Výpis názvu a definičných bodov všetkých základných sídelných jednotiek typu priemyselný areál, ktoré sa nachádzajú v meste Brno

```
SELECT zsj.name, geometry.x_axis, geometry.y_axis FROM zsj
INNER
JOIN cadastral_territory ON zsj.id_cadastral_territory = cadastral_territory.id
INNER JOIN village ON cadastral_territory.id_village = village.id
INNER JOIN geometry ON zsj.id_geometry = geometry.id
WHERE village.name = 'Brno' AND zsj.id\_zsj\_character = '3';
```

Tento dotaz je oproti predchádzajúcim dotazom značne zložitejší a zobrazí názov všetkých základných sídelných jednotiek spolu s ich definičným bodom, ktoré sa nachádzajú v meste Brno. Dotaz pracuje s tabuľkami **zsj**, **cadastral_territory**, **village** a **geometry**.

Dotaz najprv prepojí tabuľky **zsj** a **cadastral_territory** pomocou prvej **INNER JOIN** klauzule na základe atribútov **id_cadastral_territory** a **id**. Druhá klauzula **INNER JOIN** prepojí tabuľky **cadastral_territory** a **village** na základe atribútov **id_village** a **id**. Posledná klauzula **INNER JOIN** získa definičné body základných sídelných jednotiek pomocou prepojenia tabuliek **zsj** a **geometry** na základe atribútov **id_geometry** a **id**. Klauzula **WHERE** určuje, že chceme získať základné sídelné jednotky z mesta Brna typu priemyselný areál. Pokiaľ by sme chceli zmeniť typ sídelnej jednotky, tak zmeníme hodnotu atribútu **id_zsj_character** vo **WHERE** klauzuli na hodnotu atribútu **id** z tabuľky **zsj_character** tohto typu (napríklad hodnota **id_zsj_character** bude 10 pre základné sídelné jednotky typu lesná plocha). Bolo by možné vyhľadávať aj podľa názvu typu základnej sídelnej jednotky rozšírením o ďalšiu klauzulu **INNER JOIN**, ktorá by pracovala s tabuľkami **zsj** a **zsj_character**.

Šiesty dotaz - Výpis názvov miest Juhomoravského kraja

```
SELECT village.name FROM village
INNER JOIN village_type ON village.id_status = village_type.id
INNER JOIN district ON village.id_district = district.id
INNER JOIN region_1960 ON district.id_region_1960 = region_1960.id
WHERE region_1960.name = 'Jihomoravský' AND
(village_type.name = 'Město' OR village_type.name = 'Statutární město'
OR village_type.name = 'Hlavní město') ORDER BY village.name;
```

Jedná sa o dotaz, ktorý znie jednoducho, ale na základe väzieb medzi jednotlivými tabuľkami databázy je jeho štruktúra komplikovaná. Prvá klauzula **INNER JOIN** spojí obec s jej typom pomocou prepojenia tabuliek **village** a **village_type** na základe atribútov **id_status** a **id**. Druhá klauzula **INNER JOIN** prepojí obec s jej okresom prepojením tabuliek **village** a **district** na základe atribútov **id_district** a **id**. Posledná klauzula **INNER JOIN** prepojí okres s krajom prepojením tabuliek **district** a **region_1960** na základe atribútov **id_region_1960** a **id**. Klauzula **WHERE** určuje názov kraja (**Jihomoravský**) a typ obce (keďže chceme získať všetky mestá je potrebné zahrnúť aj statuárne mestá a hlavné mesto). Na záver dotaz zoradí všetky mestá vzostupne podľa abecedy pomocou klauzule **ORDER BY**.

Kapitola 6

Záver

Na základe získaných poznatkov z naštudovania problematiky VFR bol vytvorený skript, ktorý umožňuje spracovávať súbory VFR a ukladať ich spracovaný obsah do naimplementovanej databázy. Skript aktuálne spracúva iba textové dáta bez obrazových dát vlajok a znakov.

Vznikol nástroj, ktorý umožňuje dostať údaje, ktoré sa nachádzajú v databáze RÚIAN pomocou VFR do naimplementovanej databázy, ktorá môže byť súčasťou akéhokoľvek webového rozhrania, informačného systému prípadne mapového klienta. Nástroj tak umožní spoločnostiam, ktoré ho budú využívať prístup k dátam RÚIAN pomocou naimplementovanej databázy (spoločnosti nebudú musieť využívať aplikáciu VDP a riešiť spracovanie VFR ale rovno pristúpia k dátam RÚIAN pomocou naimplementovanej databázy).

V prílohe B je uvedený manuál slúžiaci na spoznanie skriptu a databázy pri prvom spustení.

Na záver je uvedených ešte niekoľko nápadov na vylepšenie:

- Výsledný skript by mohol byť rozšírený o ďalšie typy geometrie (napríklad definičné čiary ulíc), vďaka čomu by mohli rôzni mapový klienti zobrazovať na základe dát z databázy i územie ulíc.
- Implementovaná databáza by mohla byť v budúcnosti rozšírená o ďalšie územné celky (hlavne územné celky zo súborov VFR typu **Obec a podradené**, keďže z týchto súborov implementovaná databáza aktuálne uchováva iba prvok **Ulice**).
- Skript by mohol v budúcnosti kontrolovať pri spracovaní súborov VFR príznak **NespravneUdaje** tak, aby sa do databázy dostali iba územné celky, ktoré sú platné k dátumu, v ktorom majú byť uložené do databázy.
- Pre skript a databázu by mohlo byť do budúcnosti vytvorené webové rozhranie, s ktorým bude spúšťanie skriptu a zobrazenie údajov databázy príjemnejšie a jednoduchšie (dve tlačítka pre obsluhu skriptu - **Create Database** a **Update Database** a v prípade databázy výpis jednotlivých dát priamo vo webovom rozhraní bez nutnosti pristupovať k databáze napríklad pomocou nástroja phpMyAdmin).

Literatúra

- [1] *Pády v českém jazyce*. [Online; navštíveno 08.05.2019].
URL <https://cesky-jazyk.okhelp.cz/sklonovani/pady-v-ceskem-jazyce.php>
- [2] *RFC 1738 - Uniform Resource Locators (URL)*. December 1994, [Online; navštíveno 21.04.2019].
URL <https://tools.ietf.org/html/rfc1738>
- [3] *RFC 1738 - Uniform Resource Locators (URL)*. December 1994, [Online; navštíveno 21.04.2019].
URL <https://tools.ietf.org/html/rfc1738>
- [4] *RFC 4180 - Common Format and MIME Type for Comma-Separated Values (CSV) files*. Október 2005, [Online; navštíveno 20.04.2019].
URL <https://tools.ietf.org/html/rfc4180>
- [5] *ISUI - příručka uživatele*. Január 2012, [Online; navštíveno 20.04.2019].
URL https://www.cuzk.cz/isui-elearning/tools/ISUI_prirucka_uzivatele.pdf
- [6] *ČÚZK - Informační systém katastru nemovitostí - IKSIN*. December 2014, [Online; navštíveno 20.04.2019].
URL <https://www.cuzk.cz/Katastr-nemovitosti/0-katastru-nemovitosti/Informacni-system-katastru-nemovitosti-ISKIN.aspx>
- [7] *Extensible Markup Language (XML)*. Október 2016, [Online; navštíveno 20.04.2019].
URL <https://www.w3.org/XML/>
- [8] *Struktura a popis výměnného formátu RÚIAN (VFR)*. Február 2016, [Online; navštíveno 29.04.2019].
URL [https://www.cuzk.cz/Uvod/Produkty-a-sluzby/RUIAN/2-Poskytovani-udaju-RUIAN-ISUI-VDP/Vymenny-format-RUIAN/Vymenny-format-RUIAN-\(VFR\)/Struktura-a-popis-VFR-1_8_0.aspx](https://www.cuzk.cz/Uvod/Produkty-a-sluzby/RUIAN/2-Poskytovani-udaju-RUIAN-ISUI-VDP/Vymenny-format-RUIAN/Vymenny-format-RUIAN-(VFR)/Struktura-a-popis-VFR-1_8_0.aspx)
- [9] *Agendové informační systémy, Správa základních registrů*. 2019, [Online; navštíveno 20.04.2019].
URL <http://www.szrcr.cz/registr-osob/agendove-systemy>
- [10] *Co (ne)jsou ZR, Správa základních registrů*. 2019, [Online; navštíveno 20.04.2019].
URL <http://www.szrcr.cz/co-jsou-to-zakladni-registry>
- [11] *eGON rozhraní, Správa základních registrů*. 2019, [Online; navštíveno 20.04.2019].
URL <http://www.szrcr.cz/egon-rozhrani>

- [12] *Mapový klient Marushka pre vyhľadávanie RÚIAN prvkov*. 2019, [Online; navštíveno 20.04.2019].
URL <https://vdp.cuzk.cz/marushka/>
- [13] *Nahlížení do katastru nemovitostí*. 2019, [Online; navštíveno 20.04.2019].
URL <https://nahlizenidokn.cuzk.cz/StahniAdresniMistaRUIAN.aspx>
- [14] *O SZR, Správa základních registrů*. 2019, [Online; navštíveno 20.04.2019].
URL <http://www.szrcr.cz/sprava-zakladnich-registru>
- [15] *Verejný dálkový přístup (VDP) Uživatelská dokumentace*. 2019, [Online; navštíveno 20.04.2019].
URL <https://www.cuzk.cz/VDP-uzivatelska-dokumentace>
- [16] *What is PDF ? Adobe Portable Document Format / Adobe Acrobat DC*. 2019, [Online; navštíveno 20.04.2019].
URL <https://acrobat.adobe.com/sk/en/acrobat/about-adobe-pdf.html>
- [17] *Úvodná obrazovka aplikácie verejného diaľkového prístupu*. 2019, [Online; navštíveno 20.04.2019].
URL <https://vdp.cuzk.cz/>
- [18] *ČÚZK - Informace o RÚIAN*. 2019, [Online; navštíveno 20.04.2019].
URL [https://www.cuzk.cz/Uvod/Produkty-a-sluzby/RUIAN/RUIAN-\(1\).aspx](https://www.cuzk.cz/Uvod/Produkty-a-sluzby/RUIAN/RUIAN-(1).aspx)

Príloha A

Obsah priloženého CD

- **src** - zdrojové súbory skriptu
- **tex** - zdrojové súbory pre vytvorenie výsledného PDF dokumentu
- **php** - súbory obsahujúce PHP verziu, ktorá bola použitá pre testovanie skriptu
- **sql** - SQL súbory pre vytvorenie a testovanie databázy

Príloha B

Manuál

V tejto prílohe je popísaný návod na vytvorenie databázy a spustenie skriptu.

B.1 Vytvorenie databázy a jej prepojenie s PHP skriptom

Vytvorenie databázy je popísane v nasledujúcich krokoch:

- **1.** Určíme si server, na ktorom budeme chcieť databázu založiť (napríklad lokálny MySQL server pomocou nástroja Xampp - <https://www.apachefriends.org/download.html>)
- **2.** Na danom serveri založíme novú databázu, do ktorej budeme chcieť ukladať dáta RÚIAN.
- **3.** Nad založenou databázou spustíme sql `sql/db.sql` (cesta z koreňového adresára skriptu)
- **4.** v súbore `RuianDatabase.php`, ktorý je v koreňovom adresári skriptu vyplníme konštanty `HOST`, `DBNAME`, `USER` a `PASS` podľa konfigurácie serveru, na ktorom sme vytvorili databázu. Od tohto momentu je PHP skript prepojený s databázou.

B.2 Spustenie skriptu

Skript je nutné spúšťať s verziou PHP 5 a vyššou. Daná verzia PHP musí mať nainštalované rozšírenie, ktoré umožňuje použiť triedu `PDO`. Skript je možné spúšťať pomocou jedného z dvoch spôsobov:

- **Spustenie skriptu bez parametrov** - slúži pre prvotné naplnenie databázy a používa sa pri prvom kontakte s databázou. Príklad spustenia v operačnom systéme Linux:

```
php Ruian.php
```

- **Spustenie skriptu s parametrom update** - slúži pre aktualizáciu vyplnenej databázy a používa sa v prípade, keď chceme aktualizovať databázu o nové prvky v období od poslednej práce s databázou po súčasný dátum. Príklad spustenia v operačnom systéme Linux:


```
php -update Ruian.php
```

Po naplnení databázy prvkami RÚIAN môžeme databázu otestovať pomocou ukázkových sql súborov umiestnených v priečinku **sql** koreňového adresára skriptu.