

# VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA INFORMAČNÍCH TECHNOLOGIÍ  
ÚSTAV INFORMAČNÍCH SYSTÉMŮ

FACULTY OF INFORMATION TECHNOLOGY  
DEPARTMENT OF INFORMATION SYSTEMS

## EGOVERNMENT: ZPRACOVÁNÍ INFORMACÍ O EKONOMICKÝCH SUBJEKTECH

BAKALÁŘSKÁ PRÁCE

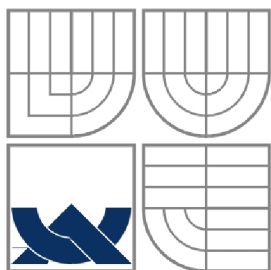
BACHELOR'S THESIS

AUTOR PRÁCE

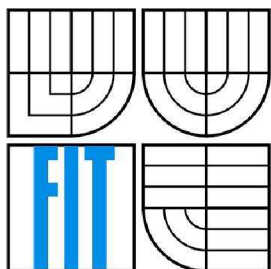
AUTHOR

PETR KONVALINKA

BRNO 2013



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ  
BRNO UNIVERSITY OF TECHNOLOGY



FAKULTA INFORMAČNÍCH TECHNOLOGIÍ  
ÚSTAV POČÍTAČOVÝCH SYSTÉMŮ

FACULTY OF INFORMATION TECHNOLOGY  
DEPARTMENT OF COMPUTER SYSTEMS

# EGOVERNMENT: ZPRACOVÁNÍ INFORMACÍ O EKONOMICKÝCH SUBJEKTECH

EGOVERNMENT: PROCESSING OF INFORMATION ABOUT ECONOMIC SUBJECTS

BAKALÁŘSKÁ PRÁCE

BACHELOR'S THESIS

AUTOR PRÁCE

AUTHOR

PETR KONVALINKA

VEDOUCÍ PRÁCE

SUPERVISOR

Ing. PAVEL OČENÁŠEK, Ph.D.

BRNO 2013

## **Abstrakt**

Tato bakalářská práce se zabývá elektronickou veřejnou správou v ČR, jak po stránce informačních technologií, tak i po stránce legislativní. Popisuje vybrané rejstříky státní správy, zejména informační systém ARES, který je využit v návrhu praktické části práce. Výsledkem práce je aplikace, která získává údaje z více rejstříků. Údaje pak spojuje a vizuálně interpretuje uživateli. Aplikace je vytvořena pomocí knihoven QT v jazyce C++.

## **Abstract**

This bachelor's thesis deals with eGovernment of the Czech Republic, as in terms of information technology, as well as legislative. The thesis describes selected registers of eGovernment of the Czech Republic, especially ARES information system, which is used in proposition of practical part of the thesis. The outcome of the thesis is the application that retriever data from multiple registers. Data are combined are visually interpreted to user. The application is created with assistance of QT Libraries of language C++.

## **Klíčová slova**

eGovernment, rejstříky, aplikace, QT Framework, státní správa, vizualizace, ekonomické subjekty, ARES, eGon, uživatelské rozhraní.

## **Keywords**

eGovernment, registers, application, QT Framework, administration, visualization, economic subjects, ARES, eGon, user interface.

## **Citace**

Konvalinka Petr: eGovernment: Zpracování informací o ekonomických subjektech, bakalářská práce, Brno, FIT VUT v Brně, 2013

# **eGovernment: Zpracování informací o ekonomických subjektech**

## **Prohlášení**

Prohlašuji, že jsem tuto bakalářskou práci vypracoval samostatně pod vedením pana Ing. Pavla Očenáška, Ph.D.

Uvedl jsem všechny literární prameny a publikace, ze kterých jsem čerpal.

.....

Petr Konvalinka

15. května 2013

## **Poděkování**

Tímto bych chtěl poděkovat vedoucímu mé bakalářské práce panu Ing. Pavlu Očenáškovvi, Ph.D. za odbornou pomoc, vstřícnost a čas, který mi věnoval při její tvorbě.

© Petr Konvalinka, 2013

*Tato práce vznikla jako školní dílo na Vysokém učení technickém v Brně, Fakultě informačních technologií. Práce je chráněna autorským zákonem a její užití bez udělení oprávnění autorem je nezákonné, s výjimkou zákonem definovaných případů..*



# Obsah

Obsah .....	1
1 Úvod.....	3
2 Vymezení pojmů .....	4
2.1 Ekonomický subjekt .....	4
2.2 Informační systém .....	4
2.3 eGovernment.....	4
3 Informace ve státní správě .....	6
3.1 ISVS.....	6
3.1.1 Požadavky na ISVS .....	6
3.2 Elektronické dokumenty .....	8
4 eGovernment v ČR .....	9
4.1 eGon.....	9
4.1.1 Czech POINT.....	9
4.1.2 Komunikační infrastruktura.....	9
4.1.3 Zákon o eGovernmentu .....	9
4.1.4 Základní registry .....	10
4.2 Klaudie.....	10
5 Rejstříky státní správy v ČR .....	11
5.1 ARES .....	11
5.2 Obchodní rejstřík .....	11
5.3 Živnostenský rejstřík .....	13
5.4 Registr ekonomických subjektů.....	13
5.5 Rejstřík politických stran a hnutí .....	14
6 Analýza a návrh aplikace .....	16
6.1 Analýza požadavků.....	16
6.1.1 Existující aplikace.....	17
6.2 Specifikace požadavků .....	18
6.3 Návrh .....	19
6.3.1 Strukturální návrh .....	19
6.3.2 Návrh chování.....	20
6.3.3 Údaje z rejstříků.....	22
6.3.4 Datový model.....	22
6.3.5 Uživatelské rozhraní .....	23
7 Implementace.....	26

7.1	Použité nástroje.....	26
7.1.1	Vývojové prostředí QT .....	26
7.1.2	SQLite.....	26
7.2	Struktura aplikace .....	26
7.3	Princip spojování subjektů.....	29
7.4	Rejstřík PSH .....	29
8	Testování.....	31
8.1	Výsledná aplikace.....	31
8.2	Konkrétní data .....	34
8.3	Výsledky testování.....	34
9	Závěr .....	35
	Obsah CD .....	38

# 1 Úvod

Neustálý vývoj informačních technologií se dotýká i státní správy. V současné době musí stát evidovat mnoho informací jak o občanech, tak i o ekonomických subjektech. Veřejnost potřebuje mít přístup k těmto informacím z různých důvodů, např. pro rozvíjení obchodních vztahů. Dříve k tomuto účelu sloužily přeplněné kartotéky. K získání vůbec nějakých informací jsme si museli vystát frontu na úřadě. Dnes jsou tyto informace skrze internet dostupné z pohodlí domova.

Hlavním cílem této práce je navrhnout a implementovat aplikaci, která zpracovává data o ekonomických subjektech. Tyto data bude získávat z různých rejstříků státní správy. Informace budou vizualizovány uživateli ve formě grafu, který bude interaktivní. Práce se skládá z devíti kapitol.

Druhá kapitola se věnuje pojmu eGovernment. Vysvětlí jeho význam, a jaký má přínos pro občany. Třetí kapitola popisuje uchování informací ve státní správě a zaměřuje se i na stránku legislativní.

Ve čtvrté kapitole najdeme vývoj eGovernmentu v České republice. Zjistíme, jaké jsou používány nástroje a prostředky k jeho realizaci.

Poslední teoretickou částí práce je pátá kapitola. Ta nás nutně seznámí s rejstříky státní správy, které budeme chtít využít v aplikaci. Zjistíme zde, jaké informace obsahují jednotlivé rejstříky.

Šestá kapitola už se zaměřuje na praktickou část práce. Najdeme zde návrh a analýzu požadavků na výslednou aplikaci.

Implementační detaily spolu s použitými nástroji na realizaci aplikace jsou popsány v kapitole sedmé.

Předposlední kapitola popisuje praktické ověření funkčnosti aplikace na různých datech a ukázky výsledné aplikace.

V poslední závěrečné kapitole nalezneme zhodnocení dosažených výsledků a možnosti budoucího rozšíření aplikace.

## 2 Vymezení pojmů

### 2.1 Ekonomický subjekt

Ekonomický subjekt [1] je jednotka, která vystupuje v hospodaření. V naší práci budeme pojem ekonomický subjekt využívat hlavně pro právnické subjekty a fyzické osoby s postavením podnikatele.

### 2.2 Informační systém

*„Informační systém je systém, umožňující účelné uspořádání sběru, uchování, zpracování a poskytování informací.“* [2]

Informace můžeme chápat jako data s nějakým přidaným významem – např. do informačního systému se uloží informace, že pan Novák je podnikatel. V naší práci se budeme zabývat informačními systémy, které poskytují informace o ekonomických subjektech široké veřejnosti.

### 2.3 eGovernment

Pojem eGovernment si můžeme představit jako elektronizaci státní správy a samosprávy. Většinou se do eGovernmentu zahrnuje i elektronizace celého výkonu veřejné moci. Přesné vymezení je složité a definice se liší dle zdrojů. Některé si zde uvedeme.

Definice Ministerstva vnitra České republiky (dále jen MVČR) zní: *„eGovernment představuje transformaci vnitřních a vnějších vztahů veřejné správy pomocí informačních a komunikačních technologií s cílem optimalizovat interní procesy. Jejím cílem je pak rychlejší, spolehlivější a levnější poskytování služeb veřejné správy nejširší veřejnosti a zajištění větší otevřenosti veřejné správy vztahu ke svým uživatelům.“* [3]

Dle OECD (Organizace pro hospodářskou spolupráci a rozvoj) je eGovernment zaměřen na využití nových informačních a komunikačních technologií, která se vztahují na celou řadu vládních funkcí. Důležitý je hlavně internet, jakožto prostředek umožňující lepší splnění úkolů veřejné správy. [4]

Definice podle [10] zní: *„eGovernment je využívání informačních technologií veřejnými institucemi pro zajištění výměny informací s občany, soukromými organizacemi a jinými veřejnými institucemi za účelem zvyšování efektivity vnitřního fungování a poskytování rychlých, dostupných a kvalitních informačních služeb.“*

Hlavní výhody elektronizace státní správy jsou v rychlosti, kvalitě a neomezené časové dostupnosti služeb občanům, finančních úsporách a jednoduchosti. eGovernment usnadňuje lidem práci, odstraňuje fronty na úřadech. Občané mohou získávat informace a stahovat formuláře z pohodlí domova. Snižuje se rozhodovací činnost a komunikace uvnitř státního aparátu. Zmenšují se finanční prostředky nutné k úřadování – např. poštovné. Dále dochází k úspoře personálu, protože některé úkony usnadňují informační technologie – např. agenda. Velkou výhodou je rychlost úřadování. Úředník má přístup k databázi (katalog, registr, evidence), kde jsou všechny informace seřazeny a lze v nich jednoduše vyhledávat. eGovernment zajišťuje lepší transparentnost veřejné správy. Lidé mají jednodušší přístup k informacím a mohou tedy lépe kontrolovat činnost úřadů – snižuje se možnost korupce, protekce atd. Naopak se zvyšuje možnost zapojit se aktivně do dění veřejné správy – existují různé ankety, diskusní fóra, průzkumy a v některých státech už jsou zavedené i eVolby – elektronické hlasování. [6]

Nevýhodou jsou nemalé vstupní investice – finanční náklady na informační technologie, školení pracovníků – musí umět pracovat s počítačem, efektivně využívat informace a pracovat s internetem atd. Dále je zde problém bezpečnosti. Hrozí úniky dat, roste počítačová kriminalita. [7]

Pro ještě lepší pochopení pojmu eGovernment si uvedeme některé oblasti a činnosti, které do něj zapadají. Jsou to informační systémy veřejné správy, elektronická komunikace, ochrana osobních údajů, elektronický podpis, elektronické veřejné zakázky, registry veřejné správy, konverze dokumentů a jiné. Pod pojem eGovernment patří i související legislativa, která umožňuje výměnu informací mezi komunikačními subjekty ve veřejné správě. [10]

## 3 Informace ve státní správě

Tato kapitola popisuje, jakým způsobem a v jaké formě jsou ve státní správě uchovávány informace o ekonomických subjektech. Zabývat se zde budeme i související legislativou v ČR.

### 3.1 ISVS

Informační systémy veřejné správy (ISVS) jsou informační systémy, které jsou definovány zákonem č. 365/2000 Sb. [8] a slouží pro výkon veřejné správy. Tento zákon tyto informační systémy popisuje. Zákon vznikl 23. 10. 2000 a byl několikrát novelizován.

Základní myšlenkou zákona je, aby se o informační systém starali odborníci, aby informační systém dělal právě ty věci, které má a nakonec, aby byl informační systém opatřen návodem. Rozeznat normální informační systém od ISVS není jednoduché, a proto vznikl dokonce metodický pokyn „Co je a co není ISVS“ [9], který popisuje, jaký informační systém je ISVS a vysvětluje, jak přistupovat k popisu ISVS v informační koncepci.

**Podle [9] můžeme za ISVS považovat:**

- Informační systémy, které jsou uvedeny v právním předpisu a tento zákon je označuje jako ISVS.
- Rejstříky, Registry, Evidence uvedené v právním předpisu, který je odkazuje na zákon ISVS.
- Rejstříky, registry, evidence, seznamy, databáze atd. uvedené v právním předpisu, který je označuje jako ISVS, ale na zákon o ISVS neodkazuje.
- Právní předpis o informačním systému, který upravuje jeho vedení jako ISVS, ale jako ISVS jej neoznačuje.

#### 3.1.1 Požadavky na ISVS

Na ISVS a na orgán veřejné správy, který jej spravuje, jsou kladeny určité požadavky [10].

**Správce ISVS musí uveřejňovat číselníky.** - Číselníkem je myšlen seznam přípustných hodnot datových prvků ve formě dvojic. Dvojice obsahuje kód a hodnotu. „*Datový prvek je jednotka dat, která je v daném kontextu dále považována za nedělitelnou a je jednoznačně definována.*“ [3]. Příkladem datového prvku může být např. jméno nebo telefonní číslo. Tyto datové prvky jsou standardizovány v zákoně [8]. Standardizace je nutná, protože budeme-li vyměňovat data mezi ISVS a datové definice budou rozdílné, tak vyměňovaná data mohou být nečitelná nebo je budeme muset náročně převádět. Příkladem standardizace je datový prvek pohlaví, který má tři přístupné hodnoty a k nim číselník: 1 - muž, 2 – žena, 0 - neznámo. Ukázku je možné vidět na obrázku 3.1. Správce

tohoto datového prvku je Ministerstvo vnitra ČR. Správci číselníků jsou stanoveni zvláštním právním předpisem. Dle zákona [8] je vytvořen ISVS o datových prvcích (ISDP) [11]. Tento IS je veřejný a obsahuje všechny standardizované číselníky a datové prvky.

### Detail číselníku pro JeDP: POHLAVÍ OSOBY

<b>Aktuální stav číselníku:</b>	Zapsán	
<b>Typ číselníku:</b>	Interní číselník	
<b>SDP číselníku:</b>	Ministerstvo vnitra	<b>Oblast DP:</b>
<b>POS číselníku:</b>	Není ustanoven	<b>Verze:</b>
<b>Identifikátor:</b>	AL0042	<b>Název</b>
<b>URL adresa číselníku:</b>	https://www.sluzby-isvs.cz/isdp/DatovyPrvek/DetailCiselnikProJeDP.aspx?id=EEC9F679-9EBA-4DE4-A65E-6F9F98FE6903	
<b>URL adresa dokumentace:</b>	https://www.sluzby-isvs.cz/isdp/DatovyPrvek/DetailCiselnikProJeDP.aspx?id=EEC9F679-9EBA-4DE4-A65E-6F9F98FE6903	
<b>Plánované datum vyhlášení:</b>	01.01.2007	<b>Datum vyhlášení:</b>
<b>Externí identifikátory:</b>		
<b>Ostatní</b>		
<b>Autorská práva:</b>		<b>Dokumentace licence:</b>
<b>Držitel práv:</b>		<b>Popis změn:</b>
<b>Položky interního číselníku</b>		
1		
Kód položky	Název položky	Zkrácený název položky
0	neznámo	
1	mužské	
2	ženské	
9	neuveдено	

Obrázek 3.1: Číselník Pohlaví osoby z ISDP [11]

**Orgán veřejné správy musí Ministerstvu vnitra ČR v elektronické podobě zpřístupňovat informace o jím vedených IS.** - Vznikl informační systém o ISVS. Do tohoto systému jsou povinni zpřístupňovat informace všechny ISVS ze dvou důvodů: tvorby nových informačních systémů a komunikace mezi nimi. Komunikace probíhá mezi ISVS a jiným informačním systémem. Přesun dat probíhá automatizovaným způsobem. V informační systému o ISVS můžeme najít informace o možnosti připojení k různým ISVS, ze kterých pak můžeme získávat volně dostupná data. K tomuto účelu je nutné znát datový formát, model přenosu, port a další informace. [10]

**Komunikace všech ISVS musí probíhat přes referenční rozhraní.** - Toto se týká pouze komunikujících ISVS. Definice referenčního rozhraní je následující:

*„Referenční, sdílené a bezpečné rozhraní informačních systémů veřejné správy je souhrn právních, technických, organizačních a jiných opatření vytvářejících jednotné integrační prostředí informačních systémů veřejné správy, které poskytuje kvalitní soustavu společných služeb včetně služeb výměny oprávněně vyžadovaných informací mezi jednotlivými informačními systémy orgánů veřejné správy a dalšími subjekty, a to i se systémy mimo Českou republiku.“ [8].*

Komunikace mezi ISVS se řídí určitými pravidly:

- povinné zapsání komunikujícího ISVS do IS o ISVS,
- použití vyhlášených datových prvků, které jsou v ISDP,
- všechna komunikace musí vytvářet logy – kdo a jaké informace požadoval po ISVS,
- komunikace ISVS podléhá atestaci, kterou udělí akreditovaná organizace (pověřena MVČR). [10]

**Dlouhodobé řízení ISVS.** - Orgány veřejné správy musí vytvořit informační koncepci. Koncepce obsahuje charakteristiku, financování a odpovědnost u spravovaných ISVS. Navíc obsahuje dlouhodobé cíle v řízení kvality a bezpečnosti ISVS. [10]

**Dokumentace.** - Na základě informační koncepce jsou orgány veřejné správy povinni vydávat provozní dokumentaci k ISVS, která obsahuje bezpečnostní politiku, systémovou a uživatelskou příručku a jiné provozní dokumenty. [10]

## 3.2 Elektronické dokumenty

*„Elektronický dokument je označení digitálně zpracovaného dokumentu, který je možno editovat pomocí počítače. Není fyzicky na papíře, ale je virtuálně uložen v počítačové paměti.“*[12]

V současné době se většina informací ve státní správě uchovává v elektronické podobě, protože je oproti papírové mnohem výhodnější. Největší výhodou je úspora času pracovníků a peněz, ale jsou zde i určité nevýhody. [13]

### **Výhody elektronického dokumentu:**

- Úspora finančních prostředků – např. ušetření nákladů za poštovné nebo vyhotovení dokladu.
- Rychlá komunikace – získání informací během okamžiku.
- Jednoduchá manipulace.
- Automatizované účetnictví – lze nastavit automatické účtování dokladů.
- Omezení chyb – při přepisování papírových dokumentů mohou vznikat chyby. [13]

### **Nevýhody elektronického dokumentu:**

- Investice zavedení – při přechodu z papírových dokumentů na elektronické je nutné vynaložit nemalé výdaje za nové řešení.
- Ochrana – je nutné dokument zálohovat a archivovat.
- Zabezpečení – dokument je při procházení sítí vystaven riziku odpozorování, zneužití či změnění a musí se řešit také důvěryhodnost dokumentu.



# 4 eGovernment v ČR

Tato kapitola popíše vývoj eGovernmentu v České republice a jeho nástroje, které se u nás používají.

## 4.1 eGon

Symbolem eGovernmentu v ČR je postavička jménem eGon. Projekt eGon sahá do roku 2006, kdy vznikl s úmyslem elektronizovat veřejnou správu za účelem usnadnění života občanům a zvýšení efektivity veřejné správy. eGon je znázorňován jako živý organismus, který obsahuje čtyři existenčně důležité části – prsty, oběhovou soustavu, srdce a mozek. Fungování částí mezi sebou vzájemně souvisí. [14]

### 4.1.1 Czech POINT

Czech POINT neboli Český Podací Ověřovací a Informační Národní Terminál je kontaktní místo veřejné správy. Symbolizuje eGonovy prsty. Přináší občanům ověřené údaje z různých rejstříků státní správy pomocí rozsáhlých a snadno dostupných poboček. Lidé tráví méně času ve frontách a různé záležitosti se vyřizují na stejném místě. Czech POINTy nabízejí například: výpis z obchodního rejstříku, výpis z živnostenského rejstříku, výpis bodového hodnocení řidiče, datové schránky atd. Czech POINTy jsou dostupné např. z obecního a městského úřadu, na pobočkách České pošty, u vybraných notářů atd. [14]

### 4.1.2 Komunikační infrastruktura

eGonovým oběhovým systémem je komunikační infrastruktura veřejné správy (KIVS), která sjednocuje různé datové linky subjektů veřejné správy do jediné. Projekt KIVS začal vznikat v roce 2007 z důvodu velkého přibývání datových linek od úřadů. Hlavním cílem projektu bylo vytvořit bezpečnou, jednotnou datovou síť s vysokým standardem. Projekt přinesl veliké finanční úspory. KIVS slouží např. jako propojení mezi orgány veřejné správy a Czech POINTy. [14]

### 4.1.3 Zákon o eGovernmentu

Zákonem o eGovernmentu se nazývá Zákon o elektronických úkonech a autorizované konverzi dokumentů č.300/2008 Sb. [23]. Tento zákon symbolizuje eGonovo srdce, jenž ho přivádí k životu. Výsledkem zákona je zrovnoprávnění elektronických a papírových dokumentů a tedy umožnění vedení elektronických spisů ve správních řízeních.

Zákon o eGovernmentu umožnil vznik Datových schránek, které představují princip komunikace se státní správou a orgány veřejné moci v elektronické podobě. Datová schránka je ze zákona zřizována každé právnické osobě a fyzické osobě, která si o ni požádá. [14]

#### **4.1.4 Základní registry**

Základní registry jsou eGonovým mozkem. Přináší jednotnost a aktuálnost klíčových databází. Tyto databáze obsahují referenční údaje. Tzn. údaje, které jsou platné a zaručené bez nutnosti ověření. Při změně údajů stačí změnit referenční údaje, přičemž se tato změna projeví ve všech ostatních rejstřících. [14]

**Základní registry jsou celkem čtyři:**

- Registr obyvatel – obsahuje údaje o občanech. Uložené údaje jsou jméno, příjmení, datum a místo narození, úmrtí a státní občanství.
- Registr práv a povinností – obsahuje údaje o působnosti orgánů ve veřejné moci.
- Registr osob – obsahuje údaje o ekonomických subjektech a orgánech veřejné moci.
- Registr územní identifikace, adres a nemovitostí - obsahuje údaje o základních územních a správních prvcích.

## **4.2 Klaudie**

Dlouhou dobu byl jediným symbolem v eGovernmentu v ČR eGon. Ale v dubnu roku 2011 si vzal za manželku<sup>1</sup> Klaudii. Tato nová postavička přináší nový model vývoje počítačových technologií a to cloud computing. Klaudie má za úkol zajistit, aby všechny informační a komunikační technologie byly efektivnější a levnější. Význam cloud computingu je ve sdílení HW i SW prostředků pomocí sítě. [5]

---

<sup>1</sup> Klaudie je uváděna jako eGonova nová životní partnerka. URL <<http://www.mvcr.cz/clanek/klaudie-od-spravy-majetku-k-modelu-poskytovani-a-odebirani-sluzeb.aspx?q=Y2hudW09Mg%3d%3d>>

# 5 Rejstříky státní správy v ČR

Definice rejstříku: „Rejstřík je veřejný seznam právně důležitých skutečností, kde by měly být zachyceny všechny případy.“ [15]

Ve veřejné správě ČR existuje mnoho rejstříků a registrů. Většina z nich poskytuje veřejnosti informace skrze webové stránky. Uvedeme si zde pouze ty, které budeme chtít využít v naší aplikaci.

## 5.1 ARES

Administrativní registr ekonomických subjektů (ARES) [16] je informační systém, který poskytuje veřejnosti údaje o ekonomických subjektech z IS veřejné správy. Obsahuje majoritní a ostatní zdroje. Z majoritních zdrojů (z tzv. zdrojových registrů) čerpá data a ukládá je do databáze. Další zdroje jsou doplněny ve formě odkazů – nejsou uloženy v databázi ARESu. Všechny údaje, které ARES poskytuje, jsou pouze informačního charakteru. ARES spravuje Ministerstvo financí ČR. Na obrázku 5.1 můžeme vidět ukázkou vyhledání osoby.

Zdrojové rejstříky ARESu jsou: obchodní rejstřík, živnostenský rejstřík, registr ekonomických subjektů, registr církví a náboženských společností, registr zdravotnických zařízení, seznam občanských sdružení a spolků, evidence zemědělského podnikatele, seznam politických stran a hnutí a rejstřík škol a školských zařízení.

Data, která ARES nabízí, jsou i ve formě XML, lze tedy k němu přistupovat automatizovaně. K IS ARES budeme přistupovat v naší aplikaci. Využijeme ale pouze některé rejstříky a ty si blíže představíme.

## 5.2 Obchodní rejstřík

Obchodní rejstřík (OR) je veřejný seznam. Je vytvořen podle zákona č. 513/1991 Sb, o obchodním zákoníku [17]. Ukládají se do něj údaje o podnikatelích.

**Do obchodního rejstříku se dle zákona [18] zapisují tyto skutečnosti:**

- firma,
- sídlo (právnícké osoby), místo podnikání nebo bydliště (fyzické osoby),
- předmět podnikání,
- právní norma právnícké osoby,
- u fyzické osoby rodné číslo a datum narození,
- identifikační číslo (IČO),
- jméno a bydliště prokuristy a způsob jakým jedná,

- jméno, bydliště (firma a sídlo) osoby, která je statutárním orgánem právnické osoby nebo jeho členem, a den vzniku a zániku její funkce,
- jméno a bydliště nebo firma u veřejné obchodní společnosti a sídlo jejích společníků,
- další skutečnosti, o kterých to stanoví právní předpis.

Údaje do OR se zapisují o Obchodních společnostech, družstvech a některých zahraničně podnikajících osobách. Fyzické osoby se zapisují, pokud o to sami požádají anebo to stanoví zákon. OR je spravován příslušným „rejstříkovým soudem“ – krajské soudy a Městský soud v Praze. OR je každému přístupný a je dostupný na internetu<sup>1</sup>. [17]

IČ	Obchodní firma Místo podnikání	Odkazy
72325488	Karel Novák Třebíč, Vnitřní Město, Jihlavská brána 6/2	<a href="#">RES</a> <a href="#">RŽP</a> <a href="#">CEDR</a> <a href="#">DPH</a>
44953259	Ing. František Novák Třebíč, Podklášteří, Týnská 745/58	<a href="#">RES</a> <a href="#">RŽP</a> <a href="#">DPH</a>
12152048	Pavel Novák Třebíč, Podklášteří, Hrádek 292/19	<a href="#">RES</a> <a href="#">RŽP</a> <a href="#">DPH</a>
74723286	Jiří Novák Třebíč, Podklášteří, Dr. Suzy 588/2	<a href="#">RES</a> <a href="#">RŽP</a>
47445131	Jiří Novák Třebíč, Podklášteří, Nová 901/35	<a href="#">RES</a> <a href="#">RŽP</a>
75633337	Ing. Petr Novák Třebíč, Podklášteří, 9. května 174/72	<a href="#">RES</a> <a href="#">RŽP</a>
74825712	Bc. Lukáš Novák Třebíč, Podklášteří, Krátká 891/8	<a href="#">RES</a> <a href="#">RŽP</a>
75552981	Karel Novák Třebíč, Podklášteří, Žerotínovo nám. 10/9	<a href="#">RES</a> <a href="#">RŽP</a>

Obrázek. 5.1: Vyhledání ekonomického subjektu v IS ARES. Po kliknutí na hypertext ve sloupci *Odkazy* lze zjistit informace o daném subjektu. [16]

<sup>1</sup> Obchodní rejstřík, URL <<https://or.justice.cz/ias/ui/rejstřík>>

## 5.3 Živnostenský rejstřík

Živnostenský rejstřík nebo někdy také registr živnostenského podnikání (RŽP) je podle zákona č. 455/1991 Sb, o živnostenském podnikání [18] informační systém veřejné správy spravovaný Živnostenským úřadem České republiky. RŽP je veřejný seznam vedený v elektronické podobě – je dostupný na internetu<sup>1</sup>. Do RŽP se zapisují podnikající fyzické i právnické osoby.

**Do Živnostenského rejstříku se dle zákona [18] zapisují následující údaje:**

- u fyzických osob jméno, příjmení, státní občanství, bydliště, datum narození, rodné číslo,
- u právnických osob obchodní firma, název, sídlo, a u fyzických osob, které jsou statutárními orgány nebo jeho členy stejné údaje, jak o bod výše,
- identifikační číslo (IČO),
- obchodní firma,
- předmět podnikání, druh živnosti,
- provozovna, v níž je živnost provozována,
- doba platnosti, vzniku a pozastavení živnostenského oprávnění,
- rozhodnutí o úpadku
- a další doplňující údaje dle zákona.

## 5.4 Registr ekonomických subjektů

Registr ekonomických subjektů (RES) je veřejný seznam, který je veden podle zákona č. 89/1995 Sb., o státní statistické službě [19]. RES má pouze evidenční význam. Zapisují se do něj údaje získaných podle zvláštních zákonů. Např. z živnostenského zákonu, z obchodního zákoníku, ze zákona o správě daní a poplatků a ze zákona o soukromém podnikání občanů. Pojem ekonomický subjekt je vysvětlen v kapitole 2.1. RES je dostupný na internetu<sup>2</sup>.

**Do Registru ekonomických subjektů se dle zákona [19] zapisují následující skutečnosti:**

- identifikační číslo (IČO) a obchodní firma,
- místo nebo sídlo podnikání, poštovní adresa, příp. Tel.,
- právní forma a předmět podnikání,
- datum vzniku podnikání, datum zániku podnikání a jeho příčina,
- kategorie velikosti ekonomického subjektu a klasifikace,
- u fyzické osoby rodné číslo,

---

<sup>1</sup>Živnostenský rejstřík, URL <<http://www.rzp.cz>>

<sup>2</sup> Registr ekonomických subjektů, URL <<http://registry.czso.cz/irsw/>>

- počet zaměstnanců a výše příjmu
- a další údaje dle zákona.

The screenshot shows the website of the Ministry of Industry and Trade of the Czech Republic, specifically the RŽP (Registry of Entrepreneurs) section. The page displays public information for a business owner.

**Údaje z veřejné části Živnostenského rejstříku**

Platnost k 24.01.2013 10:40:42

**Subjekt**

Jméno a příjmení:	<b>Ing. Rostislav Novák</b>
Datum narození:	<b>25.12.1943</b>
Občanství:	<b>Česká republika</b>
Bydliště:	<b>Janouškova 221/12, 674 01, Třebíč - Horka-Domky</b> (zapsáno od 16.10.1992)
Místo podnikání:	<b>Janouškova 221/12, 674 01, Třebíč - Horka-Domky</b> (zapsáno od 16.10.1992)
Identifikační číslo:	<b>12151114</b> (zapsáno od 16.10.1992)

**Živnostenská oprávnění**

**Živnostenské oprávnění č. 1:**

Předmět podnikání:	<b>Výroba, obchod a služby neuvedené v přílohách 1 až 3 živnostenského zákona</b>
Obory činnosti:	<b>Obchodní činnost</b> Zahájeno: 30.09.1992
Druh živnosti:	<b>Ohlašovací volná</b>
Vznik oprávnění:	<b>30.09.1992</b>
Doba platnosti oprávnění:	<b>na dobu neurčitou</b>
Přerušeni provozování živnosti oznámené podnikatelem:	od 11.12.2007 do 10.12.2009

Obrázek 5.2: Veřejné údaje o podnikateli v RŽP.

## 5.5 Rejstřík politických stran a hnutí

Rejstřík politických stran a hnutí je veřejný seznam vedený Ministerstvem Vnitra. Rejstřík je veden podle zákona č. 424/1991 Sb., o sdružování v politických stranách a v politických hnutích [22]. Do rejstříku se zapisují údaje o politických stranách a hnutích. Samotný rejstřík má pouze informativní charakter a je dostupný na internetu<sup>1</sup>.

**Do rejstříku politických stran a hnutí se dle zákona [22] zapisují následující údaje:**

- název, zkratka a sídlo strany (hnutí),
- datum registrace,
- zánik strany a právní důvod, zrušením a pozastavení strany,
- identifikační číslo osoby přidělené straně,

<sup>1</sup> Rejstřík politických stran a hnutí, URL <<http://aplikace.mvcr.cz/seznam-politickyh-stran/>>

- jméno, příjmení, datum narození a adresa místa pobytu osob, které jsou statutárními orgány, strany nebo jeho členi, dále způsob, jakým jednají,
- vstup do likvidace, zahájení insolvenčního řízení.

# 6 Analýza a návrh aplikace

Tato kapitola se bude zabývat analýzou požadavků a návrhem aplikace.

**Každý vývoj softwaru je rozdělen na několik etap:**

- analýza a specifikace požadavků,
- návrh,
- implementace,
- testování,
- provoz a údržba.

Vývoj naší aplikace tedy nebude výjimkou a projde také všemi výše uvedenými etapami. Tím bychom měli docílit toho, že náš softwarový projekt bude úspěšný. Problém, kterým se budeme zabývat, není jednoduchý, proto zvolíme k návrhu a vývoji aplikace objektově orientovaný přístup. Hlavní výhodou tohoto přístupu je stabilita navrhovaných prvků a jednoduchost změn. Používanou technologií, která sjednocuje jednotlivé procesy objektově orientovaného vývoje, je jazyk UML. Tento jazyk nabízí prostředky pro unifikovanou tvorbu modelů různých aspektů navrhovaných systémů. Základními prostředky UML jsou diagramy, kterých poskytuje několik. Diagramy budou v naší analýze a návrhu využívány. K jejich kreslení je použit volně šiřitelný program yEd Graph Editor<sup>1</sup>. [20]

## 6.1 Analýza požadavků

Tématem aplikace je získávání informací o ekonomických subjektech z již probraných rejstříků státní správy. Získané informace se budou spojovat a vizuálně interpretovat uživateli. Jako zdroj informací bude použit informační systém ARES. Aplikace by měla mít grafické uživatelské rozhraní, které bude intuitivní a přívětivé k uživateli. S aplikací budou pracovat různí uživatelé, kteří ale nejsou potřeba nijak odlišovat, proto není potřeba řešit uživatelské role a s nimi spojená autentizace. Výsledná vizualizace se bude moci uložit v počítači ve formě obrázku. Aplikace by měla řešit problém, kdy se na server odesílají stejné požadavky, čímž se zbytečně zatěžuje. Aby aplikaci mohlo používat nejvíce uživatelů, tak musí být multiplatformní – spustitelná tedy nejen na počítači s operačním systémem Windows, ale i ostatních, např. unixových systémech.

Aby bylo možné pracovat se systémem ARES, je nutné respektovat podmínky používání [16]. Tyto podmínky se musí zohlednit v naší aplikaci.

---

<sup>1</sup>Více informací na webových stránkách [http://www.yworks.com/en/products\\_yed\\_applicationfeatures.html](http://www.yworks.com/en/products_yed_applicationfeatures.html)



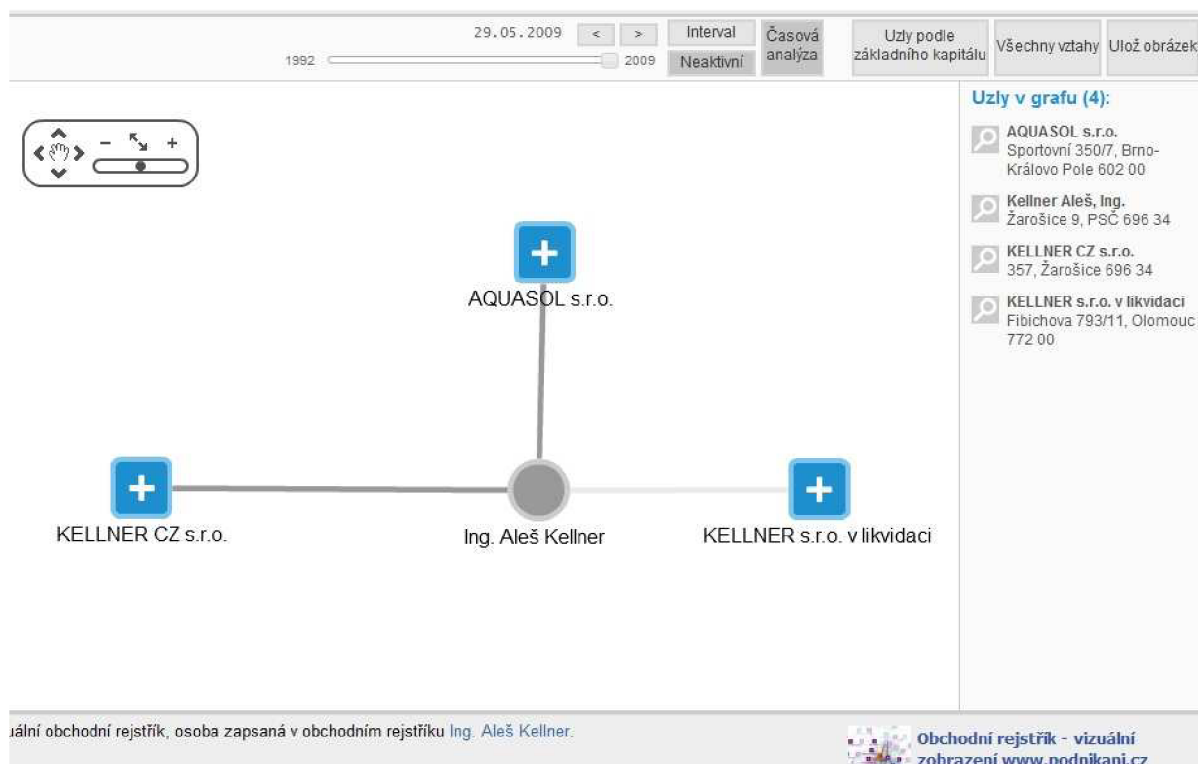
## Podmínky pro využívání služeb systému ARES jsou:

- mezi 8:00 a 18:00 je možno odeslat maximálně 1000 dotazů,
- mezi 18:00 až 08:00 následujícího dne je možno odeslat maximálně 5000 dotazů,
- nesmí se opakovaně zasílat stejné dotazy,
- nesmí se opakovaně zasílat nesprávné dotazy,
- zákaz obcházení předchozích bodů s použitím jiných IP adres. [16]

Při nerespektování těchto podmínek nám může být správcem omezen provoz.

### 6.1.1 Existující aplikace

Při návrhu své vlastní aplikace se můžeme inspirovat jinou, již vytvořenou a volně dostupnou webovou aplikací. Jedná se o vizualizaci obchodního rejstříku [21]. Aplikaci zadáte údaj, který chcete vyhledat v obchodním rejstříku. Výsledkem je graf zobrazující vztahy mezi námi vybraným subjektem a jinými ekonomickými subjekty, jak můžeme vidět na obrázku 6.1. S jednotlivými částmi grafu lze hýbat a obrázek si můžeme poté uložit.



Obrázek 6.1: Vizuální obchodní rejstřík, vyhledaný subjekt Ing. Aleš Kellner [26]

## 6.2 Specifikace požadavků

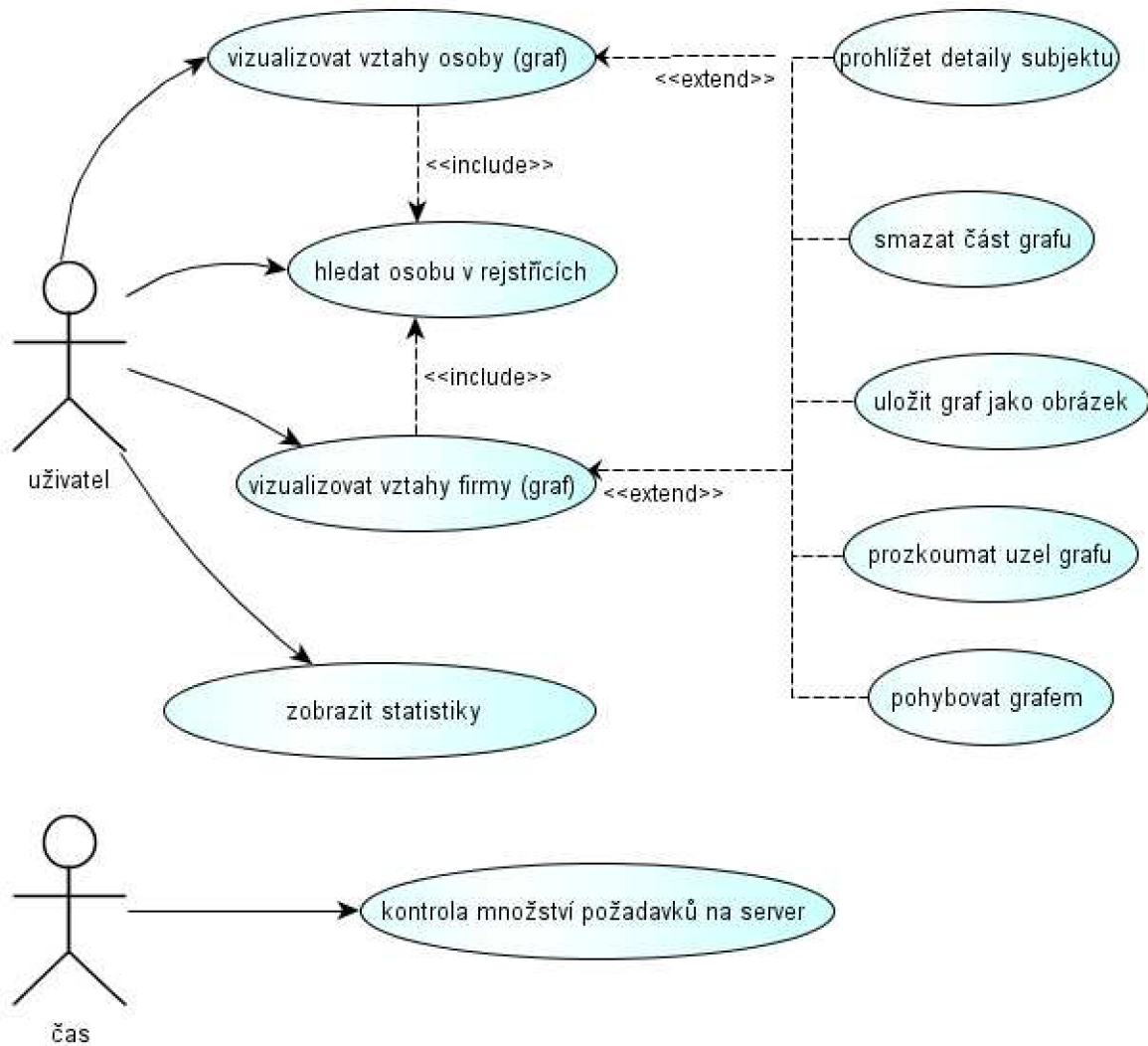
Specifikace požadavků navazuje na jejich analýzu a stanoví požadavky na funkcionalitu našeho systému.

Na základě nastudovaných informací budeme vytvářet aplikaci podobnou Vizualnímu obchodnímu rejstříku [26]. Aplikace vyhledá osobu a zjistí o ní, v jakých rejstřících veřejné správy se nachází. Následně budou informace vizualizovány. Jako zdroj informací využijeme údaje z IS ARES, který poskytuje data skrze XML rozhraní. Na rozdíl od [21] však nebudeme data získávat z pouze jednoho informačního zdroje (obchodní rejstřík), ale z více. Systém ARES pracuje s cca 15 informačními zdroji. Protože se naše aplikace bude zaměřovat primárně na osoby, můžeme některé zdroje vynechat (např. Registr zdravotnických zařízení nebo Rejstřík škol a školských zařízení). Dále vynecháme ty rejstříky, které neobsahují takové údaje o osobách, abychom je mohli správně identifikovat (např. registr církví a náboženských společností obsahuje čistě pouze jména osob). V poslední řadě nebudeme pracovat s registrem ekonomických subjektů, protože obsahuje stejné informace, které najdeme v ostatních rejstřících.

**Po vynechání nevhodných rejstříků bude aplikace pracovat s následujícími rejstříky:**

- obchodní rejstřík,
- živnostenský rejstřík,
- seznam politických stran a hnutí.

K zachycení požadavků se používá diagram případů užití z jazyka UML. Tento diagram, zachycující funkční požadavky na naši aplikaci, je na obrázku 6.2. V diagramu jsou použity zobecnění. Relace <<include>> znamená, že případ užití v sobě zahrnuje jeden nebo více jiných případů užití. V našem případě to znamená, že pokud budu chtít vizualizovat vztah osoby, tak předtím musím vždy nejprve vyhledat nějakou osobu. Naproti tomu relace <<extend>> znamená, že případ užití rozšiřuje jiný případ užití. Např. pokud budu vizualizovat vztah osoby, tak můžu, ale nemusím prohlížet detaily vztahu.



Obrázek 6.2: Diagram případů užití

## 6.3 Návrh

Tato podkapitola se bude zabývat návrhem aplikace. Návrh je rozdělen do několika částí. Statickou strukturu systému nám popíše strukturální návrh. Další částí je návrh chování, který popisuje, jak mezi sebou jednotlivé části systému spolupracují. Poté následuje návrh na použití údajů z rejstříků. V jakém tvaru budou data uloženy v databázi, nám popíše datový model. Poslední částí pak bude návrh uživatelského rozhraní aplikace.

### 6.3.1 Strukturální návrh

K popsání statické struktury systému slouží diagram tříd z jazyka UML. Tento diagram znázorňuje strukturu objektových tříd a jejich vztahy mezi sebou. Je to jeden z nejdůležitějších diagramů.

Na obrázku 6.3 najdeme diagram tříd naší aplikace. Kvůli přehlednosti zde uvádíme pouze názvy tříd bez atributů a operací.

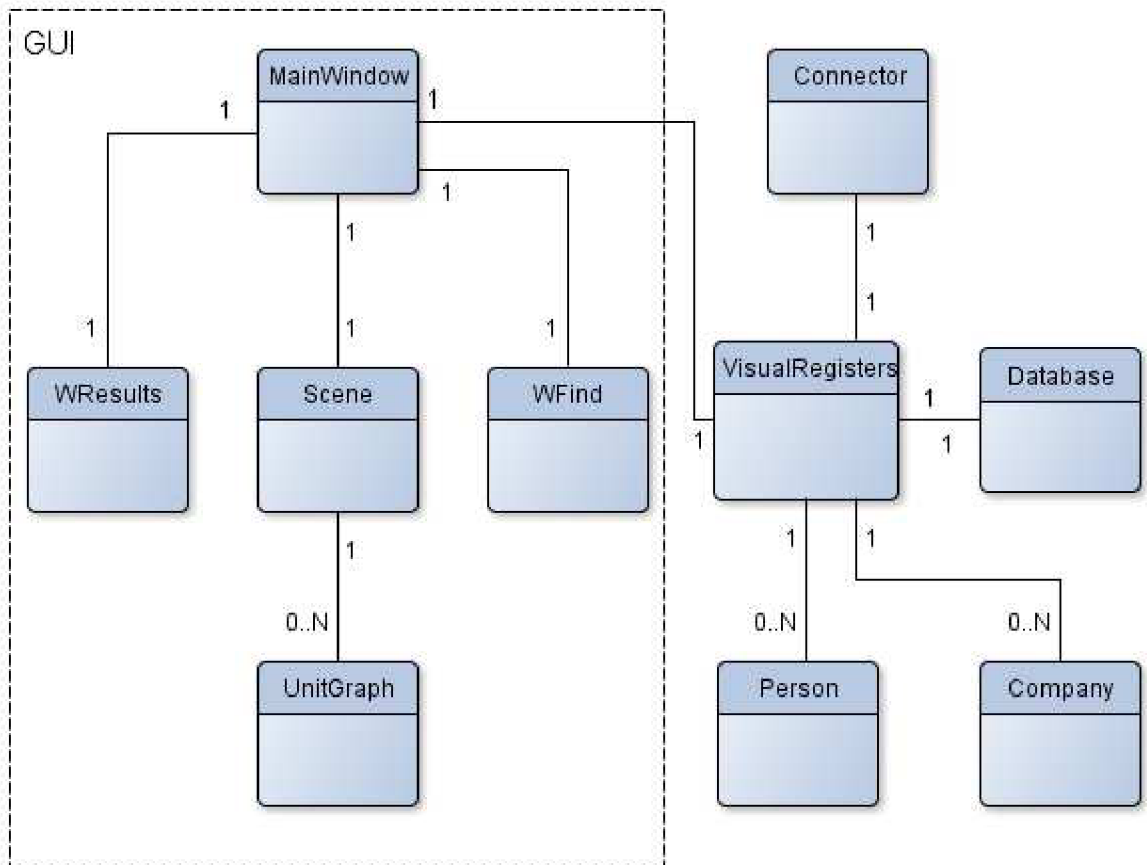
V ohraničeném rámečku se nachází třídy, které se týkají grafického uživatelského rozhraní. Třída `MainWindow` implementuje hlavní okno celé aplikace a zahrnuje všechny uživatelem vygenerované události (např. kliknutí na tlačítko). Třídy `WResult`, `Scene`, `WFind` jsou třídy pro jednotlivé části rozhraní aplikace, mezi kterými se lze přepínat. Třída `WFind` obsahuje formulář pro vyhledání osoby. Třída `WResult` obsahuje výsledky vyhledání osoby a Třída `Scene` je scéna pro vizualizaci grafu. Scéna se skládá z objektů třídy `UnitGraph`, které znázorňují uzly grafu.

Jádrem celé aplikace je třída `VisualRegisters`, jenž pracuje se všemi ostatními třídami. Třída `Connector` slouží ke komunikaci se serverem, ze kterého získává informace. K ukládání informací získaných ze serveru do interní databáze slouží třída `Database`. Třídy `Person` a `Company` modelují reálné fyzické osoby a firmy získané z rejstříků.

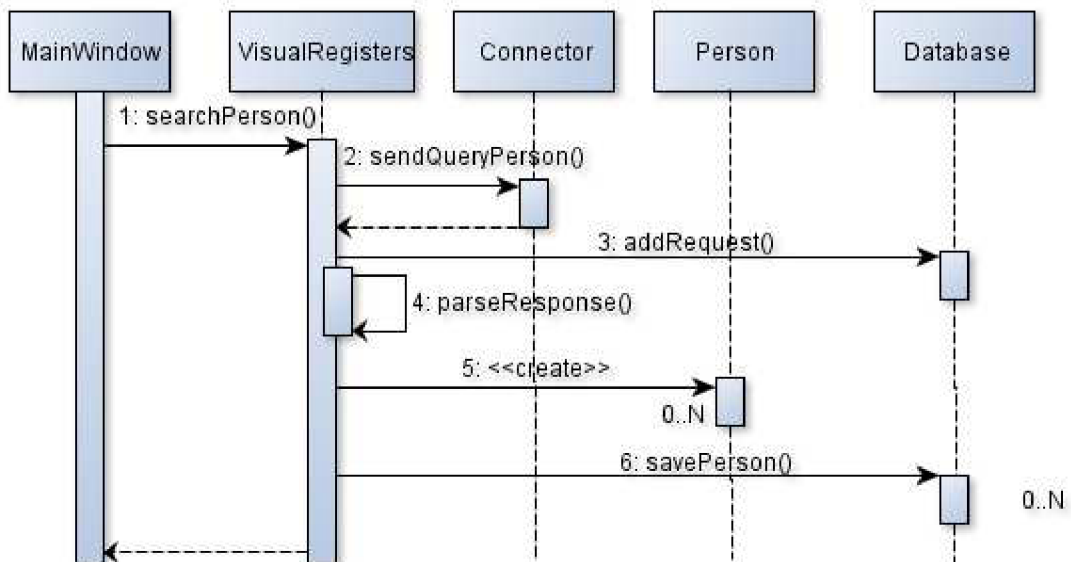
### 6.3.2 Návrh chování

Návrh chování popisuje, jak spolu jednotlivé části systému (objekty) interagují. Jazyk UML nabízí hned několik diagramů, které modelují chování systému. My si zvolíme pro nás nejvhodnější sekvenční diagram. Tento diagram znázorňuje posloupnost předávání zpráv mezi objekty v závislosti na čase. Posloupnost pak nejčastěji zachycuje v rámci jednoho detailu užití.

Jako demonstrativní ukázkou k naší aplikaci zvolíme detail užití, který vyhledá osobu v rejstříku statní správy. Jak můžeme vidět na obrázku 6.4, tak posloupnost zpráv zahajuje objekt třídy `MainWindow`, jenž zavolá metodu objektu třídy `VisualRegisters` `searchPerson`. Poté objekt třídy `Connector` odešle požadavek na server. Server nám vrátí odpověď jako výčet osob ve formátu XML. Metodou objektu třídy `addRequest` se inkrementuje počet požadavků na server. Z odpovědi serveru pak metoda `parseResponse` objektu třídy `VisualRegisters` vyextrahuje pro nás důležitá data, ze kterých se poté mohou vytvořit objekty `Person`. Objekty `Person` se pak uloží do interní databáze a vrací se zpátky jako návratová hodnota do objektu třídy `MainWindow`. U metod nejsou uvedeny parametry z důvodů přehlednosti.



Obrázek 6.3 Diagram tříd



Obrázek 6.4 Diagram sekvence pro případ užití hledání osoby v rejstřících

### 6.3.3 Údaje z rejstříků

Naše aplikace potřebuje získávat údaje o ekonomických subjektech z veřejných rejstříků, aby s nimi pak mohla dále pracovat. Jako zdroj těchto údajů nám poslouží informační systém ARES. V kapitole 5.2 jsme si uvedli, se kterými rejstříky budeme pracovat. V této kapitole si určíme, jakým způsobem budeme údaje získávat a v jakém formátu budou uloženy.

Systém ARES poskytuje XML rozhraní pro přístup k veřejným rejstříkům. Celý popis tohoto rozhraní je zde<sup>1</sup>. Rozhraní poskytuje různé formy dotazů na různé rejstříky. My pro prvotní vyhledání osoby budeme používat dotaz na *výběr z formuláře osob*, který nám vrátí údaje z obchodního rejstříku a živnostenského rejstříku. Jako parametry těchto dotazů dosadíme uživatelem zadané údaje. Tento dotaz nám vrátí seznam osob, které vyhovují zadaným kritériím.

Problémem je, že *výběr z formuláře osob* nám nevrátí seznam jedinečných osob, ale pouze záznam osoby s nekompletními údaji v rejstříku. Např. když pan P bude mít záznam ve firmě X a ve firmě Y, tak *výběr z formuláře osob* nám vrátí 2 záznamy o panu P – jeden ve firmě X a druhý ve firmě Y. Nepoznáme tedy, že pan P je vlastně jedna osoba. Z tohoto důvodu je nutné porovnat adresu a datum narození, které mají zadané ve firmě a podle těchto údajů identifikovat, že se jedná o jednu a tutéž osobu. Skutečnost, že dva různí lidé budou mít stejné jméno, adresu a datum narození, je velice nepravděpodobná. Naše aplikace nezaručuje stoprocentní pravdivost dat, proto si tento způsob identifikace můžeme dovolit.

Dále v naší aplikaci budou potřeba dotazy na jednotlivé rejstříky. Dotazy nám vždy vrátí odpověď ve formátu XML. Schémata všech odpovědí jsou v jediném souboru, který je dostupný na internetu<sup>2</sup>. Schémata jsou umístěny i v příloze s barevným vyznačením použitých elementů. Vrácených informací ze serveru je velice mnoho, je proto nutné, některé méně důležité věci vyřadit. Uživatel, který by si chtěl zobrazit kompletní informace o společnosti, si je může najít na webových stránkách z některých rejstříků. V aplikaci bude na ně přímo odkaz. Všechny údaje, se kterými budeme pracovat, budou znázorněny v datovém modelu aplikace v následující kapitole.

### 6.3.4 Datový model

Z požadavků na aplikaci vyplývá, že bychom neměli příliš zatěžovat server zasíláním stejných a tedy zbytečných požadavků. Řešením tohoto problému je vytvoření vlastní databáze, ve které si budeme uchovávat již získané požadavky ze serveru. Před dalším dotazem na server se tedy vždy podíváme do vlastní databáze, zda již takový výsledek neobsahuje.

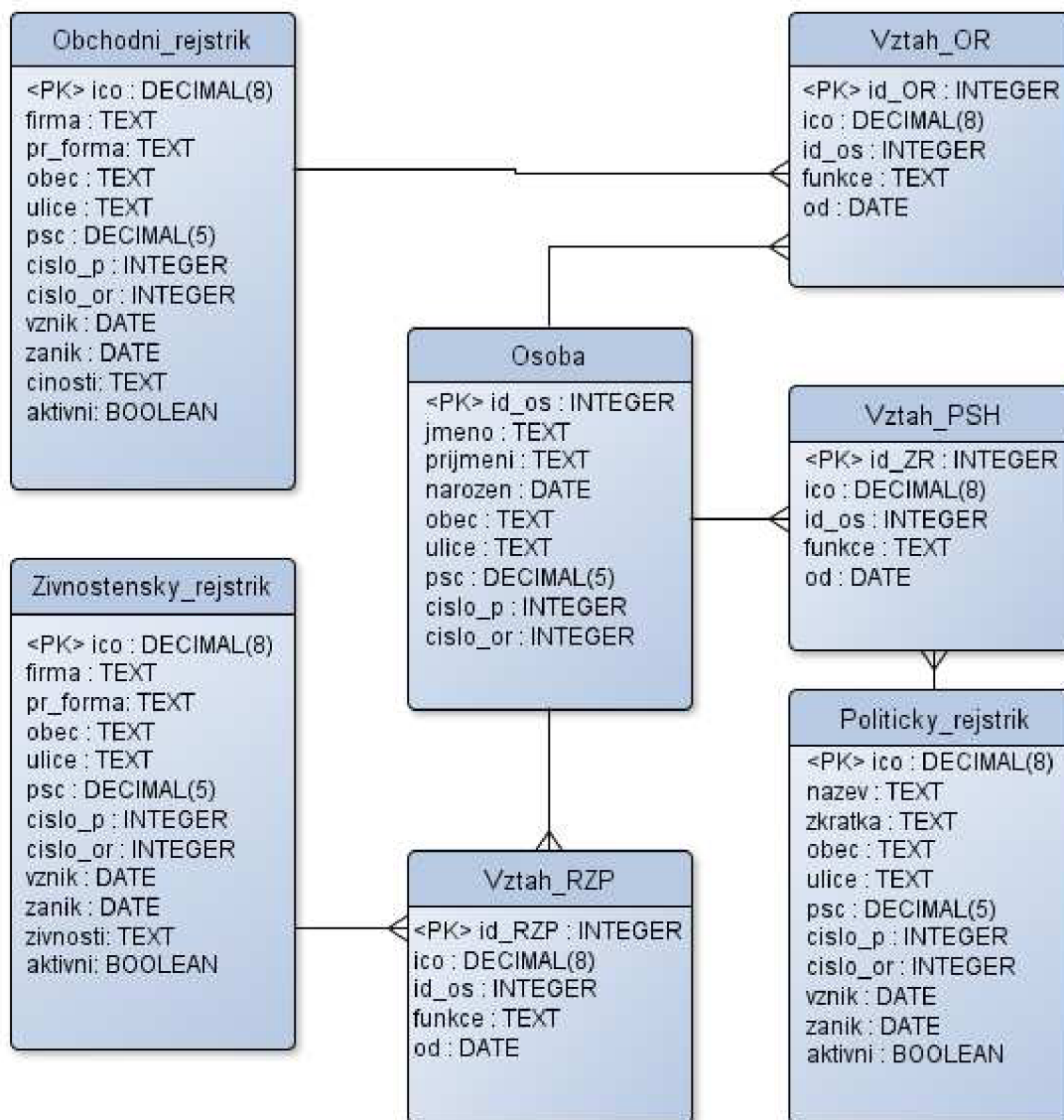
K vytvoření datového modelu se používá diagram entit a vztahů ERD, což je model popisující návrh uložených dat v systému na abstrakční úrovni.

---

<sup>1</sup> [http://wwwinfo.mfcr.cz/ares/xml\\_doc/schemas/index.html](http://wwwinfo.mfcr.cz/ares/xml_doc/schemas/index.html)

<sup>2</sup> [http://wwwinfo.mfcr.cz/ares/xml\\_doc/schemas/ares/ares\\_datatypes/v\\_1.0.4/ares\\_datatypes\\_v\\_1.0.4.xsd](http://wwwinfo.mfcr.cz/ares/xml_doc/schemas/ares/ares_datatypes/v_1.0.4/ares_datatypes_v_1.0.4.xsd)

ER diagram na obrázku 6.5 znázorňuje databázi naší aplikace. Hlavní entitou je *Osoba*, která modeluje osobu zapsanou v některé společnosti z rejstříků. Každý rejstřík, ke kterému budeme přistupovat, má svou entitu. Osoba může být zapsána v kterémkoli rejstříku a to dokonce vícenásobně (vztah M:N). Z tohoto důvodu je nutné vytvořit entity modelující vztah mezi osobou a rejstříkem.



Obrázek 6.5: ER diagram znázorňující vztahy mezi osobami a společnostmi v rejstřících

### 6.3.5 Uživatelské rozhraní

Naše aplikace bude mít grafické uživatelské rozhraní, což je velice důležitá věc pro zaujetí uživatele a jeho práci s aplikací. Uživatelské rozhraní by mělo být graficky přehledné a intuitivní. Tzn., že uživatel by neměl tápat, které prvky použít, ale měl by se s aplikací rychle naučit pracovat.

Samotná jeho práce by měla být co nejrychlejší (odstranit zbytečné klikání). V neposlední řadě by mělo uživatelské rozhraní vhodně vypadat. S takovou aplikací budou uživatelé rádi pracovat.

Na obrázku 6.6 můžete vidět návrh uživatelského rozhraní, které se zobrazí po spuštění aplikace. Na horní liště okna jsou dvě vysouvací menu. Menu nápověda bude obsahovat nápovědu, jak pracovat s aplikací a tvůrce programu. Na levé straně okna budou tři přepínací tlačítka signalizující, ve které fázi se nacházíme. Samotné okno první fáze obsahuje formulář k vyhledání osob. Dále zde nalezneme různé přepínače, kterými se upravují parametry hledání.

Stisknutím na tlačítko *Hledat* se přesuneme do druhé fáze, kde se zobrazí výsledky našeho hledání v podobě záznamu obsahujícího základní údaje. Kliknutím na záznam se označí. Toto okno ukazuje obrázek 6.7.

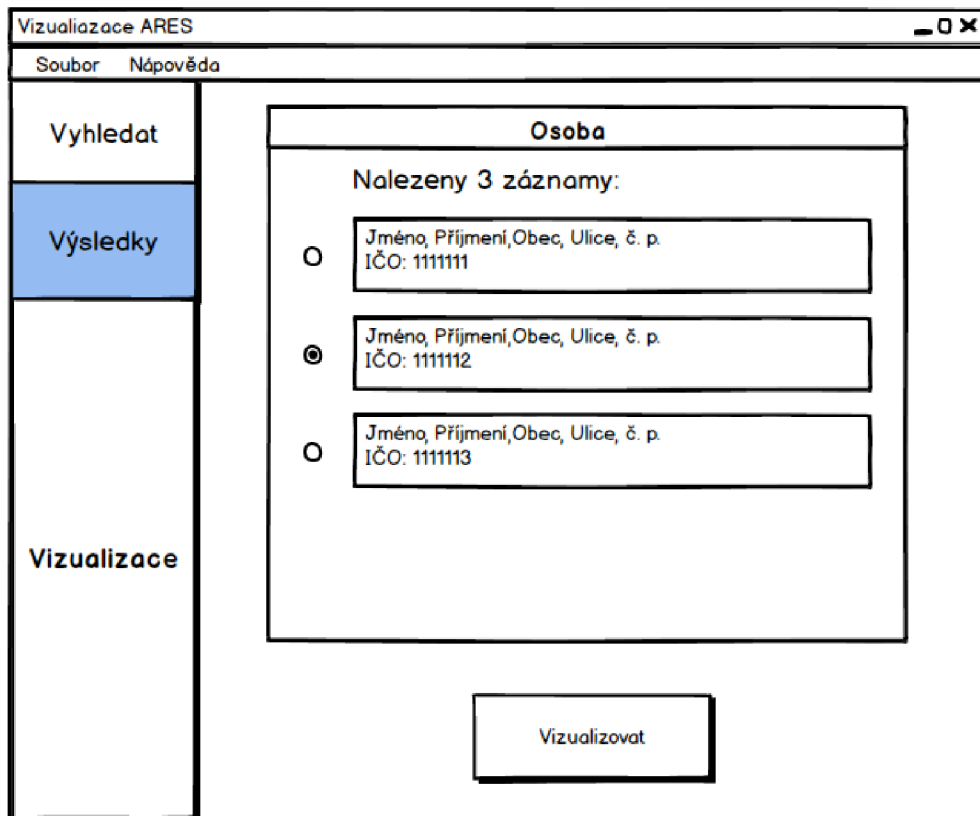
Pokud máme označený záznam, tak kliknutím na tlačítko *Vizualizovat* můžeme přejít do poslední fáze (obrázek 6.8), kterou je již samotná vizualizace výsledků v podobě grafu. Graf je na plátně, které je možné zmenšovat, zvětšovat a případně posouvat.

The image shows a window titled "Vizualizace ARES" with a menu bar containing "Soubor" and "Nápověda". On the left side, there is a vertical sidebar with three buttons: "Vyhledat" (highlighted in blue), "Výsledky", and "Vizualizace". The main area of the window contains search filters and options:

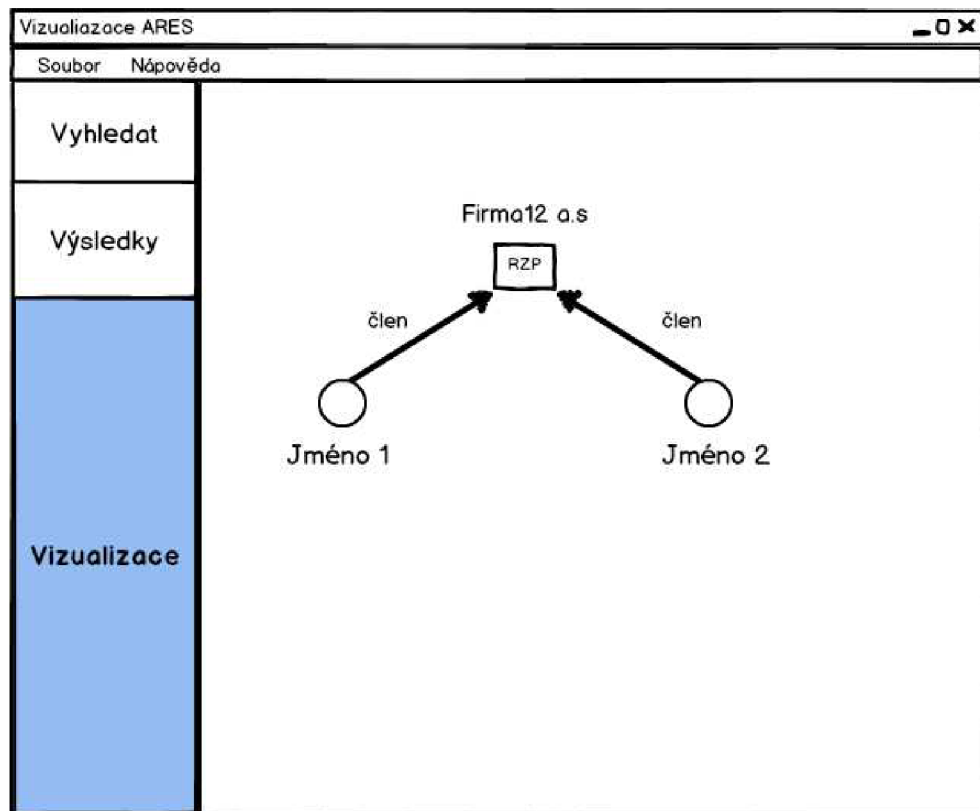
- Two checkboxes:  Hledat pouze v databázi and  Hledat i zaniklé subjekty.
- A section titled "Osoba" containing five text input fields: "Jméno", "Příjmení", "Č. domu", "Obec", and "Ulice".
- A section titled "Prohledávané rejstříky" containing three checked checkboxes:  Všechny,  Obchodní rejstřík,  Živnostenský rejstřík, and  Rejstřík politických stran.
- A "Hledat" button centered at the bottom.

Obrázek 6.6: Návrh okna po spuštění aplikace





Obrázek 6.7: Okno s vyhledanými osobami



Obrázek 6.8: Návrh výsledného grafu na plátně

# 7 Implementace

V této kapitole si popíšeme programovou realizaci aplikace a použité nástroje. Při implementaci aplikace se vycházelo z návrhu aplikace.

## 7.1 Použité nástroje

### 7.1.1 Vývojové prostředí QT

Jako vývojové prostředí pro naši aplikaci jsem zvolil knihovnu QT<sup>1</sup> vyvíjenou firmou Nokia. QT je multiplatformní nástroj pro vyvíjení softwarových projektů s grafickým uživatelským rozhraním. Aplikace jsou vyvíjeny s nativním vzhledem, takže se vždy přizpůsobí operačnímu systému. QT obsahuje knihovny pro více jazyků (C++, Python, Ruby, Pascal atd.). Já jsem si vybral objektově orientovaný jazyk C++. Tuto vývojovou knihovnu jsem zvolil z důvodu velmi přívětivého prostředí, zejména pak kvůli velké a přehledné dokumentaci. Knihovna QT obsahuje důležitou funkčnost, kterou jsou signály a sloty. Tyto zvláštní metody slouží ke komunikaci mezi objekty. Používají se například při kliknutí na tlačítko – vyšle se signál. Slot je pak metoda, která se provede v reakci na signál – obsluha po stisknutí tlačítka.

### 7.1.2 SQLite

SQLite<sup>2</sup> je relační databázový systém napsaný v jazyce C. Je to nejrozšířenější databázový systém na světě. Tato databáze není tvořena na principu klient – server jako ostatní databáze, ale je to pouze malá knihovna, se kterou pracuje aplikace pomocí jednoduchého rozhraní. Důležité je, že podporuje transakce. Vývojové prostředí QT obsahuje potřebné knihovny pro práci s touto databází. Pro naši aplikaci jsem ji zvolil z důvodu přenositelnosti a jednoduché obsluhy.

## 7.2 Struktura aplikace

Implementace aplikace je rozdělena do několika tříd. Ty nejdůležitějšími si zde uvedeme se stručným popisem.

**Třída `VisualRegisters`.** – Hlavní třída celého našeho projektu a také nejobsáhlejší. Volá metody objektu tříd `Database` a `Connector`. Pomocí nich získá informace o ekonomických

---

<sup>1</sup> Webové stránky knihovny QT, URL <<http://qt.digia.com/>>

<sup>2</sup> Webové stránky databázového systému SQLite, URL <<http://www.sqlite.org/index.html>>

subjektech. V případě získání dat ze serveru (pomocí třídy `Connector`) se nám informace vrátí ve formátu XML. K vyextrahování pro nás důležitých informací slouží metody začínající slovem `parse`. Získané informace se pak ukládají do třech seznamů:

- `listPerson` (objekty třídy `Person`) – seznam osob,
- `listCompany` (objekty třídy `Company`) – seznam společností,
- `listRelation` (objekty třídy `Relation`) – seznam vztahů mezi osobami a společnostmi.

S uvedenými seznamy se pak dále pracuje. Jsou zde implementovány různé metody (např. získání vztahů podle určité osoby), které se pak používají k výsledné vizualizaci.

Při získávání dat ze serveru musí třída počkat, než přijde odpověď (oznámení signálem). Abychom věděli, jak s přijatou odpovědí naložit, musíme vědět, ve kterém stavu se nacházíme. Z tohoto důvodu je zde tedy implementován konečný automat a třída se vždy nachází v nějakém stavu.

**Třída `Connector`.** – Třída obsahuje metody, které odesílají dotazy na server pomocí metody `GET`. Např. metoda `sendQueryOR` odešle požadavek na výpis společnosti z obchodního rejstříku podle parametru IČO.

Příklad dotazu na společnost z OR podle parametru IČO 27074358:

[http://www.wininfo.mfcr.cz/cgi-bin/ares/darv\\_or.cgi?ico=27074358](http://www.wininfo.mfcr.cz/cgi-bin/ares/darv_or.cgi?ico=27074358)

V průběhu implementace se objevily problémy s kódováním. Server totiž přijímá požadavky v kódování `windows-1250`, ale odpovědi ve formátu XML jsou v kódování `UTF-8`. V knihovně Qt jsem nenašel žádný způsob na překódování, a proto jsem musel ručně nahradit české znaky s diakritikou za znaky ve formátu `windows-1250`.

**Třída `Database`.** – Pro práci s databází slouží třída `Database`. Obsahuje metodu pro vytvoření tabulek a dále všechny metody pro práci s relační databází `SQLite`, např. pro získání osoby podle parametru `id` a k ní nalezených vztahů, slouží metoda `getPersonRelation`. Model databáze je zobrazen v kapitole 6.5. Kromě uvedených tabulek v ER diagramu obsahuje databáze ještě jednu tabulku s pouze jedním řádkem. Obsahem tabulky je datum posledního přístupu a počet přístupů na server z důvodu omezení dotazů na server, které je uvedeno v kapitole 6.1.

**Třídy `Person`, `Company`, `Relation`.** – Třídy `Person` a `Company` modelují reálné osoby a společnosti z rejstříků státní správy. Třída `Relation` pak modeluje vztah mezi těmito entitami.

Tyto třídy nemají žádné metody. Obsahují pouze atributy, které jsou kompatibilní se souvisejícími tabulkami v databázi.

**Třída MainWindow.** – Nyní se přesuneme na část grafického uživatelského rozhraní (GUI). Datová část je od GUI oddělena. Zprostředkovatelem mezi těmito dvěma částmi je třída `MainWindow`, která komunikuje s třídou `VisualRegisters`. Třída `MainWindow` představuje hlavní okno aplikace a dědí od třídy `QMainWindow`. Obsah okna je dále rozdělen do tří částí - formulář pro hledání osob, výsledky na vyhledání osob a scéna s grafem. Všechny tyto části jsou oddělenými třídami (`WFind`, `Scene`, `WResults`). Od těchto tříd třída `MainWindow` přijímá signály. Ovládací prvky tedy neimplementuje (pouze menu), ale obsahuje sloty na jejich obsluhu.

Např. při stisknutí na tlačítko Hledat, které implementuje třída `WFind`, se vyše signál, který třída `MainWindow` přijme a zavolá svou metodu `sButtonFind`. V jejím těle se pak zavolá příslušná metoda třídy `VisualRegisters`, načež se nám vrátí výsledky s vyhledanými osobami. Ty pak zobrazíme metodou třídy `WResults`.

**Třída WFind.** – Třída implementuje formulář pro vyhledání osob. Byla vytvořena pomocí `QT Designeru`, což je nástroj knihovny `QT` pro tvorbu grafických uživatelských rozhraní. Třída `WFind` obsahuje ovládací prvky knihovny `QT` – jako jsou textové pole, zaškrťovací políčka a tlačítka.

**Třída WResults.** – Pro zobrazení výsledků při hledání osoby slouží třída `WResults`. Je vytvořená také v `QT Designeru`. Abychom mohli zobrazit více prvků v jednom okně, je zde vytvořen kontejner – `QScrollArea`, do kterého se vloží jednotlivé osoby reprezentované jako `QWidgety`. Při zaplnění kontejneru se zobrazí rolovací lišta.

**Třída Scene.** – Třída `Scene` dědí z třídy `QGraphicScene`, která představuje plátno, do kterého lze umisťovat prvky třídy `QGraphicsItem`. Tuto třídu využíváme pro tvorbu grafu. Využíváme zde virtuálních metod – reakce na klik a pohyb myši, rolování kolečkem myši a na stisknuté klávesy.

Třída `Scene` obsahuje i metodu `findPoint`, která vyhledá na plátně nejbližší volné místo pro umístění dalšího prvku grafu. Parametrem této metody je uzel grafu, ke kterému se nově vkládaný prvek vztahuje. Vkládání probíhá kruhově okolo uzlu.

**Třída UnitGraph.** – Uzly grafu jsou reprezentovány objekty třídy `UnitGraph`, která dědí od třídy `QGraphicScene`. Vzhled uzlu tvoří ikony, které se liší podle druhu uzlu. Máme čtyři druhy – osoba, společnost z `OR`, společnost z `RŽP` a společnost z rejstříku politických stran a hnutí. Ikony

jsou získány pod licencí pro osobní použití z webových stránek společnosti *IconFinder*<sup>1</sup>. Pod uzlem nalezneme popis uzlu. Při změně místa uzlu se musí současně změnit i místo tohoto textu. Uzly jsou mezi sebou propojeny hranami – objekty třídy `ARC`, která dědí od `QGraphicLineItem`.

## 7.3 Princip spojování subjektů

Podle návrhu aplikace v kapitole 6.3.3 je první fází programu vyhledání osob z OR a RŽP dle zadaných parametrů. Vracené osoby ale mají údaje pouze o celém jménu a adrese. Osoby se stejným jménem a adresou spojíme a uložíme si, ve kterých společnostech se nacházejí. K porovnání použijeme regulární výrazy, abychom dosáhli co největší úspěšnosti. Následně osoby zobrazíme uživateli, který si z nich jednu vybere. Podle vybrané osoby zjistíme, které společnosti a jaké rejstříky máme prohledávat. Nejprve se podíváme do databáze. Pokud nalezneme společnost již tam, tak z ní vyjmeme všechny osoby, které k ní mají vztah. Když ji v databázi nenalezneme, tak odešleme požadavek na server. Vyextrahováním z formátu XML zjistíme stejné informace jako z databáze.

Osoby získané z rejstříků obsahují i data narození. K osobě, kterou si uživatel vybral, pak vyhledáme stejné osoby z rejstříků, které k ní patří. Poprvé se tedy osoba porovnává jen podle adresy a celého jména (i s tituly), ale poté se již porovnává i podle data narození. Nehrozí zde tedy, že bychom spojili osoby, které nejsou totožnou osobou. Všechno porovnání probíhá nezávisle na velikosti písmen. Pokud u osoby v rejstříku není datum narození (někdy se to může stát), tak tuto osobu vynecháme z prohledávání.

## 7.4 Rejstřík PSH

Systém ARES bohužel neposkytuje vyhledání osob z rejstříku politických stran a hnutí. Poskytuje ale vyhledání společností, které se v něm nacházejí. Protože společností v rejstříku je velice málo (v současné době asi jen 180), můžeme si do databáze stáhnout celý rejstřík. Každou stranu či hnutí si postupně vyextrahujeme z formátu XML a zjistíme údaje o členství osob. Osoby a strany pak ukládáme do databáze. Když v databázi najdeme osobu se stejným jménem, adresou a datem narození, tak k ní jen přidáme vztah k další společnosti. Problémem je, že celé jméno je zde uvedeno v jednom textovém řetězci a adresa také. Musíme si tedy jméno a adresu vydolovat. K tomuto účelu nám poslouží regulární výrazy.

**Ukázka vydolování částí jména z řetězce „Mgr.Ing. Jaroslav Novák, Ph. D.“ :**

Následující regulární výraz získá titul za jménem (v kulatých závorkách):

`,||s*(\\w*)||s*§`

---

<sup>1</sup> *IconFinder*, URL< <http://www.iconfinder.com/> >

Získanou část z řetězce vymažeme. Následně získáme titul před jménem:

$$||s^*([|w|.]*|.)$$

Opět získanou část vymažeme a nakonec zjistíme jméno a příjmení osoby:

$$||s^*(|w|w^*)||s+(|w|w^*)||s^*\$$$

Na stejném principu, i když obsáhlejším, získáme části adresy.

## 8 Testování

Tato kapitola se zabývá ověřením funkčnosti aplikace na jejích jednotlivých částech. Program byl testován na různých počítačích na platformě Windows 7. Ke spuštění aplikace jsou potřebné různé dynamicky linkované knihovny. Všechny jsou přiloženy ke spustitelné aplikaci na CD. Aplikaci jsem nazval *VisReg*.

Program obsahuje velké množství metod. Všechny metody bylo potřeba řádně otestovat hned při vývoji, protože později by se chyby hledaly mnohem složitěji. Ke snadnějšímu hledání chyb jsem použil ladící nástroj knihovny QT – debugger.

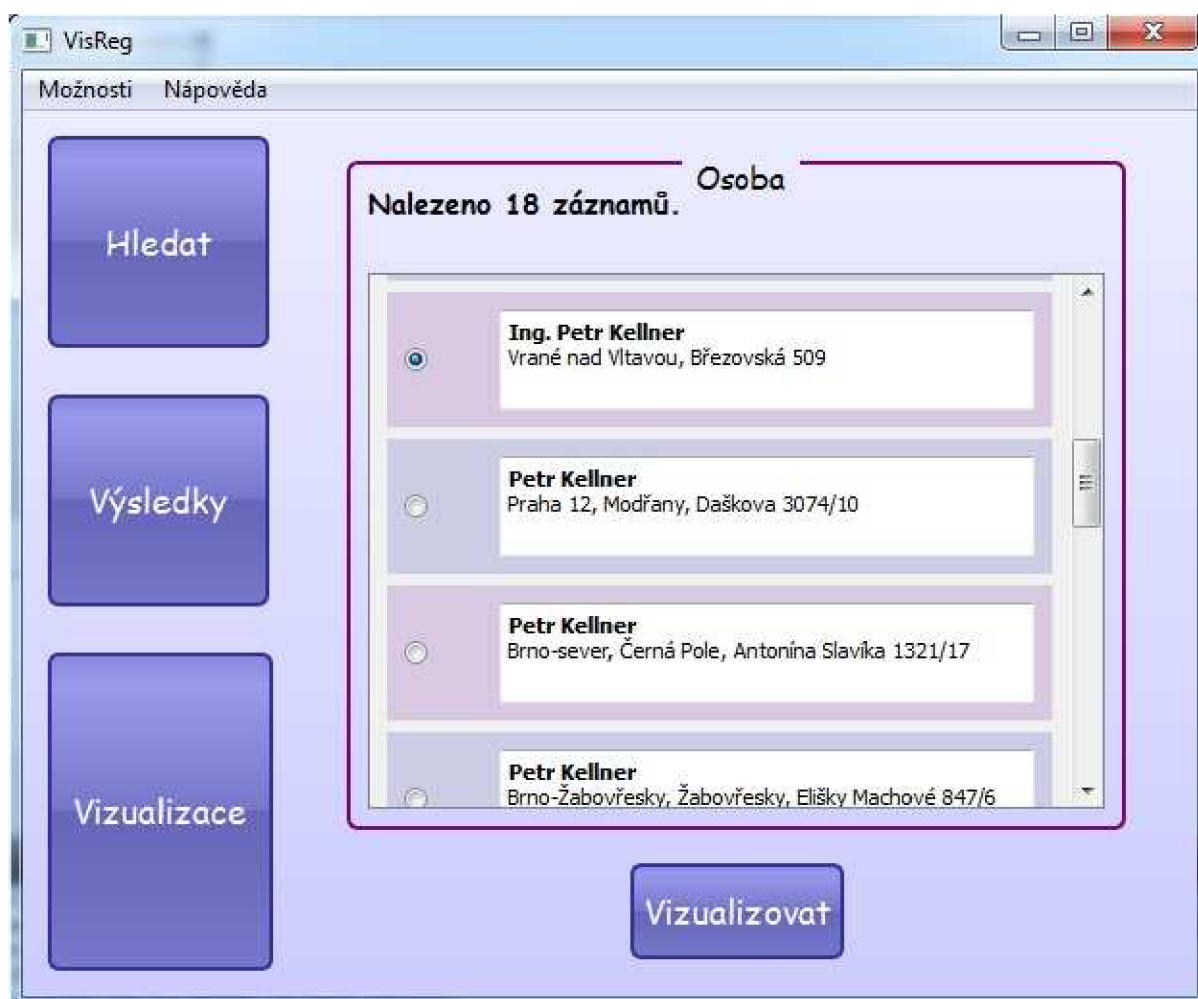
### 8.1 Výsledná aplikace

Na obrázku 8.1 nalezneme ukázkou z naší aplikace – byl proveden dotaz na jméno *Petr Kellner*. V levé části okna jsou přehledné přepínače, kterými se přepínáme mezi třemi fázemi programu. Zobrazení osob probíhá v ohraničeném rámci, který obsahuje rolovací lištu. Stisknutím na tlačítko *Vizualizovat*, začne program vyhledávat informace o dané osobě. Zatímco program vyhledává informace, tak se nám zobrazí dialog čekání. Tento dialog můžeme vidět na obrázku 8.2.

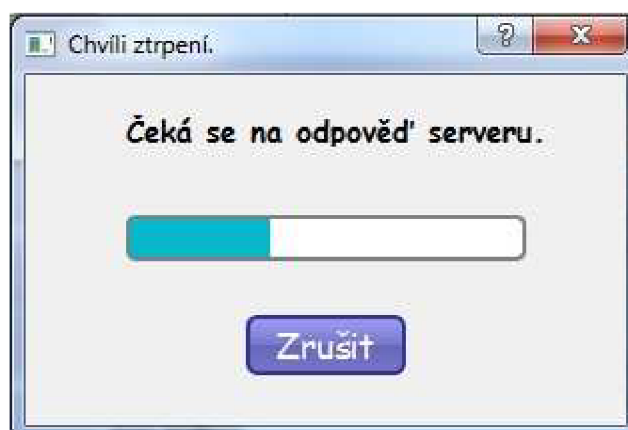
Ověřit správné výsledky vyhledání není jednoduché a nejde bohužel nijak zautomatizovat. Musíme si stejnou osobu vyhledat i v systému ARES a porovnávat výsledky, zda jsou ekvivalentní, nebo ne.

Obrázek 8.3 zobrazuje vizualizaci osoby, kterou jsme si vybrali v předchozím kroku. Ukazuje, ve kterých společnostech osoba působí. Zároveň jsme si prozkoumali další společnost – *Generali PPF Holding B. V.* V grafu jsou uzly barevně rozlišeny dle jeho druhu. Světlejší hrany znamenají, že společnost již zanikla. Kliknutím na uzel pravým tlačítkem se nám zobrazí kontextové menu. Můžeme zde uzel buď smazat, dále prozkoumat, nebo zobrazit všechny detaily v novém dialogovém okně (Obrázek 8.4). Při najetí myši nad uzel nebo hranu se nám zobrazí základní detaily. Uzly s tučně označeným popiskem ještě nebyly prozkoumány. S grafem se dá různě pracovat. Rolováním kolečkem myši se nám scéna zmenšuje a zvětšuje. Tažením myši si můžeme vybrat více objektů a pak je třeba všechny smazat. Zkratkou *CTRL+A* si pak označíme všechny uzly. Při držení *CTRL* a táhnutím myši se nám scéna posouvá dle libovolného směru. Samozřejmostí je možnost hýbání s uzly. V menu *Možnosti* si pak můžeme graf uložit ve formě obrázku typu PNG.

Otestování správné vizualizace je ještě složitější než vyhledávání subjektů. Musíme si totiž v IS ARES otevřít všechny společnosti, které zobrazujeme a v nich vyhledat, zda se tam opravdu osoba nachází.

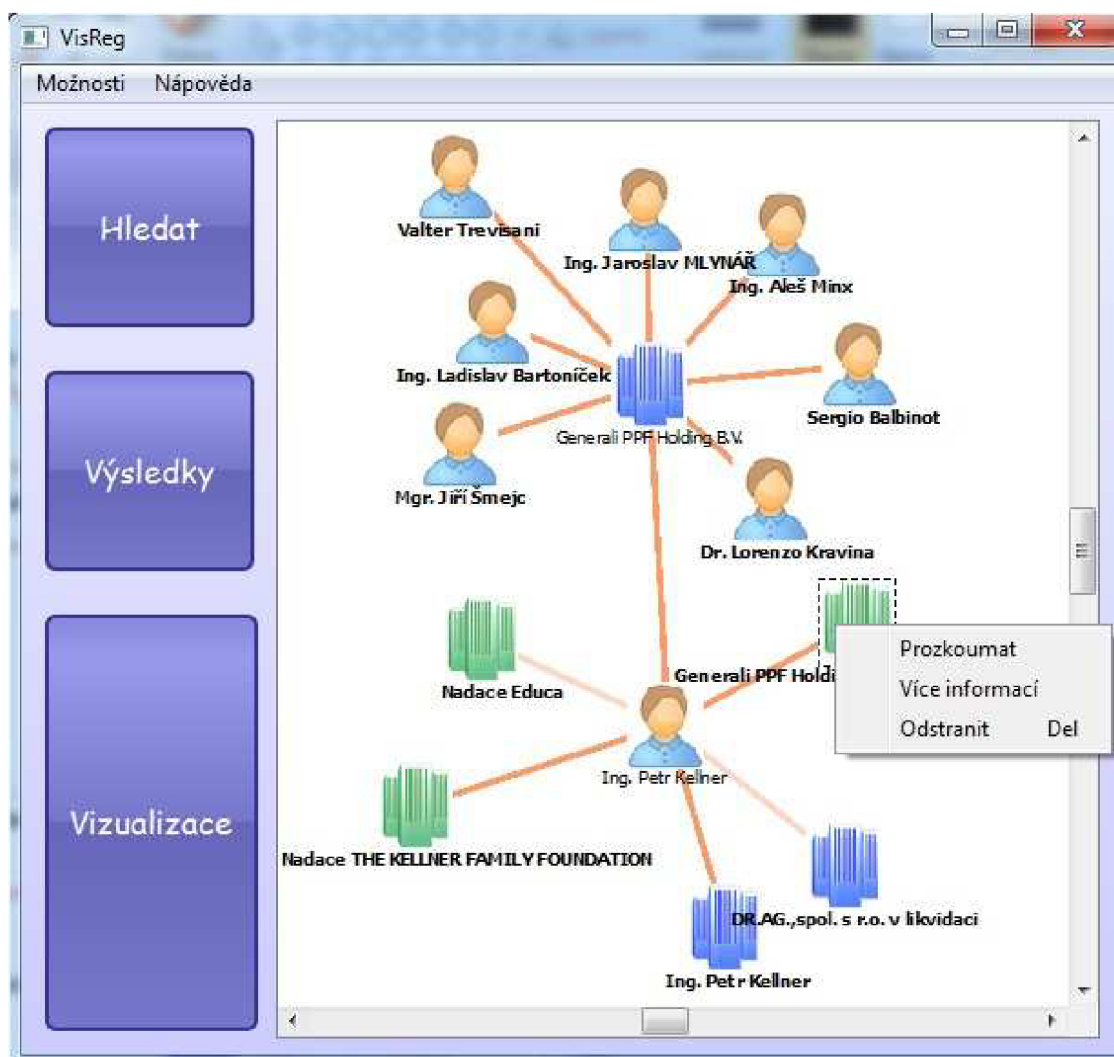


Obrázek 8.1: Vyhledané osoby na jméno *Petr Kellner*



Obrázek 8.2: Dialog na čekání, když program pracuje





Obrázek 8.3: Vizualizace subjektů ve formě grafu

Údaje o společnosti

## Generali PPF Holding B.V.

Záznam v rejstříku živnostenského podnikání.

Název Subjektu: **Generali PPF Holding B.V.**

IČO: **28239652**

Adresa: **Amsterdam, 1077XX, Strawinskylaan 933**

Právní forma: **Zahraniční osoba**

Stav: **Aktivní**

Datum zápisu: **2008-01-30**

Datum zániku:

Živnosti:

- výroba, obchod a služby neuvedené v přílohách 1 až 3 živnostenského zákona, Činnost

[Všechny informace naleznete zde.](#) OK

Obrázek 8.4: Zobrazení detailů o vybrané společnosti

## 8.2 Konkrétní data

Funkčnost aplikace zde představíme na třech osobách. Způsob testování je popsán v předchozí kapitole. Vybereme si ty osoby, které mají nejvíce vztahů u různých společnostech.

První testovanou osobou je *Ing. Petr Kellner*. V aplikaci si vyhledáme výsledky, které můžete vidět na obrázku 8.1. Za pomoci informačního systému ARES pak zjistíme, že nejvíce vztahů má *Ing. Petr Kellner* s adresou *Vrané nad Vltavou, Březovská 509*. V aplikaci si jej tedy vybereme a vytvoří se nám graf, který je možné vidět na obrázku 8.3. Naše aplikace našla šest společností, ve kterých má *Ing. Petr Kellner* nějaký záznam. Všechny tyto společnosti si otevřeme v systému ARES. Zjištěním, že se všechny výsledky shodují a u všech záznamů je i stejné datum narození. Angažmá u společností je také správné

Druhou osobou je *Jaroslav Navrátil*. Aby se zmenšila vyhledávací kritéria, tak si zvolíme město *Praha*. Vybereme si *Ing. Jaroslava Navrátila* s adresou *Praha, Mlynářská 1172/6*, protože v systému ARES má nejvíce vztahů u společností. Tyto společnosti jsou: *Falcon EU a.s. (OR)*, *Tepló Zlín, a.s. (OR)*, *GE Group a.s. (OR)*, *FOCUS DEVELOPMENT a.s. (OR)*, *ČESKÁ OBUV, s.r.o. (OR)*, *Ing. Jaroslav Navrátil (RŽP)*, *GE Group, a.s. (RŽP)*, *Tepló Zlín, a.s. (RŽP)*, *ČESKÁ OBUV, s.r.o. (RŽP)*. Aplikace nám ve výsledné vizualizaci zobrazí všech devět společností. Po podrobném zkoumání zjistíme, že i všechny údaje souhlasí.

Poslední zde předvedenou osobou je *Petr Bém*. Osob s tímto jménem je v ČR málo, proto nemusíme zadávat již žádné parametry vyhledání. Vybereme si *Petra Béma* s adresou *Praha 3, Vinohrady, Korunní 1172/81*. V systému ARES zjistíme, že má vztah ke třem společnostem - *AGIL PRAHA s.r.o. (OR)*, *AGIL PRAHA s.r.o. (RŽP)* a *Petr Bém (RŽP)*. U společnosti *AGIL PRAHA s.r.o. (OR)* má dokonce dvě angažmá – *Společník se vkladem* a *Člen statutárního orgánu*. Výsledná vizualizace nám, všechny tyto 3 společnosti zobrazí a při detailu vztahu se společností *AGIL PRAHA s.r.o. (OR)* nalezneme i dvě angažmá.

## 8.3 Výsledky testování

Po dalším testování aplikace jsem došel k zjištění, že ve většině případů se výsledky shodují. Někdy ovšem nastane problém, když je v datech rejstříku zadaná špatná informace. Např. v adrese je překlep - to se pak osoby bohužel nespojí a aplikace nám je označí jako dvě různé osoby.

Dále jsem při testování zjistil, že aplikace vždy spojí správné údaje. Při vizualizaci vztahů se někdy stane, že chybí určitý údaj. Důvodem jsou buď neúplné informace (např. v rejstříku chybí datum narození), nebo špatně zadané informace (překlepy).

## 9 Závěr

Hlavním cílem této bakalářské práce bylo vytvoření aplikace, která získává data o ekonomických subjektech z rejstříků státní správy. Aplikace získaná data sloučí a informace o působení různých osob v různých společnostech vizuálně interpretuje uživateli.

Prvním mým úkolem bylo nastudovat aktuální způsoby uchování informací ve státní správě. Zde jsem se zaměřil i na legislativní část. Dále jsem zjistil, jaké nástroje se u nás používají v oblasti eGovernmentu.

Před vývojem aplikace jsem si nejdříve vybral možné rejstříky, ze kterých jsem chtěl čerpat informace. Po jejich nastudování jsem se rozhodl využít informační systém ARES, který poskytuje údaje z více rejstříků ve formátu XML. Údaje z některých rejstříků ale nejsou kompletní a tak v naší aplikaci využíváme pouze data z obchodního rejstříku, živnostenského rejstříku a rejstříku politických stran a hnutí.

Vytvořením aplikace *VisReg* se mi podařilo dosáhnout hlavního cíle. Aplikace splňuje na ni kladené požadavky. Výsledný program umí vyhledat osobu z rejstříků a nalézt k ní vztahy k různým společnostem. Výsledek pak interpretuje uživateli ve formě grafu. S grafem lze provádět všechny důležité úkony – pohyb, mazání, přibližování a posun ve scéně tahem myši. Jednotlivé uzly můžeme dále prozkoumávat a lze tak vytvořit celou hierarchii vztahů různých osob a společností. Všechny získané informace se ukládají do lokální databáze, takže se zmenšuje vytížení serveru při opakovaném hledání.

Spojování stejných osob funguje na principu porovnání jména, adresy a data narození. Zde nastal problém. Někdy se totiž stává, že v rejstříku chybí datum narození, nebo jsou v adrese překlepy. Takové osoby pak vynecháváme z naší vizualizace. Je velice nepravděpodobné, skoro nemožné, aby aplikace spojila dvě různé osoby.

V budoucnosti by šla aplikace rozšířit i o vyhledávání společností z různých rejstříků, např. zemědělského rejstříku. Tato možnost by mohla získávat data opět pomocí informačního systému ARES. Druhou možností rozšíření je přidat vyhledávání osob i z jiných rejstříků např. z insolvenčního rejstříku. Ten ale informační systém ARES zatím nepodporuje, takže by se data musely získávat např. pomocí dolování z webových stránek rejstříku.

# Literatura

- [1] KUŽVARTOVÁ, Lenka. *Ekonomie - Ekonomie a ekonomika, vymezení pojmů* [online]. [cit. 2013-03-30]. Dostupné z: <http://www.imaturita.cz/maturitni-otazky/ekonomie/ekonomie-a-ekonomika,-vymezeni-pojmu/405/>
- [2] HRONEK, Jiří *Informační systémy* [online]. 2012 [cit. 2013-03-20]. Učební text. Dostupné z: <http://phoenix.inf.upol.cz/esf/ucebni/infoSys.pdf>
- [3] Webové stránky Ministerstva vnitra České republiky [online]. 2010 [cit. 2013-04-12]. Dostupné z: <http://www.mvcr.cz/egovernment.aspx>
- [4] Webové stránky OECD [online]. 6. 8. 2002 [cit. 2013-04-12]. Dostupné z: <http://stats.oecd.org/glossary/detail.asp?ID=4752>
- [5] NOHAVKOVÁ, Zdeňka. *Symboly eGovernmentu* [online]. [cit. 2013-03-20]. Dostupné z: <http://egovernment.euweb.cz/4.html/>
- [6] MATES, Pavel a Vladimír SMEJKAL. *E-government v českém právu*. Praha: Linde, 2006, 244 s. ISBN 80-7201-614-8.
- [7] E-government v ČR [online]. 2012 [cit. 2013-03-12]. Dostupné z: <http://e-government-v-cr.webnode.cz/vyhody-a-nevyhody-e-governmentu/>
- [8] Zákon č. 365/2000 Sb., o informačních systémech veřejné správy. In: Sběrka zákonů. 23. 10. 2000 [cit. 2013-04-12]. ISSN 1211-1244
- [9] Ministerstvo vnitra České republiky, *Co je a co není ISVS* [online]. 2010 [cit. 2013-03-17]. Dostupné z: <http://www.mvcr.cz/clanek/co-je-a-co-neni-isvs.aspx>
- [10] LIDINSKÝ, Vít. *EGovernment bezpečně*. 1. vyd. Praha: Grada, 2008, 160 s. ISBN 978-80-247-2462-1.
- [11] Informační systém o datových prvcích [online]. [cit. 2013-03-17]. Dostupné z: <https://www.sluzby-isvs.cz/ISDP/DefaultSSL.aspx>
- [12] IT SLOVNÍK.CZ. *Elektronický dokument* [online]. [cit. 2013-03-10]. Dostupné z: <http://it-slovník.cz/pojem/elektronicky-dokument>
- [13] ISVS.CZ. *Papírový versus elektronický dokument (27.díl)* [online]. [cit. 2013-03-10]. Dostupné z: <http://www.isvs.cz/papirovy-versus-elektronicky-dokument-27-dil/>
- [14] Ministerstvo vnitra České republiky, *eGON jako symbol eGovernmentu* [online]. 2010 [cit. 2013-03-23]. Dostupné z: <http://www.mvcr.cz/clanek/egon-jako-symbol-egovernmentu-moderniho-pratelskeho-a-efektivniho-uradu-252052.aspx>
- [15] Wikipedie, *Registr* [online]. 2010 [cit. 2013-03-15]. Dostupné z: <http://cs.wikipedia.org/wiki/Registr>
- [16] ARES [online]. 2012 [cit. 2013-01-10]. Dostupné z: <http://www.info.mfcr.cz/ares/ares.html.cz>
- [17] Zákon č. 513/1991 Sb. obchodní zákoník. In: Sběrka zákonů. 5. 11. 1991 [cit. 2013-01-20]. ISSN 1211 – 1244
- [18] Zákon č. 455/1991 Sb., o živnostenském podnikání. In: Sběrka zákonů. 2. 10. 1991 [cit. 2013-04-20]. ISSN 1211 – 1244
- [19] Zákon č.89/1995 Sb., o státní statistické službě. In: Sběrka zákonů. 20. 04. 1995 [cit. 2013-04-22]. ISSN 1211 – 1244
- [20] KŘENA, Bohuslav a Radek KOČÍ. *Úvod do softwarového inženýrství*. [cit. 2013-04-20] Brno : VUT, 2010. Studijní opora. VUT, FIT

- [21] Podnikani.cz, *Vizuální obchodní rejstřík* [online]. [cit. 2013-03-26].  
Dostupné z: <http://obchodni-rejstrik.podnikani.cz/>
- [22] Zákon č. 424/1991 Sb., o sdružování v politických stranách a v politických hnutích.  
In: Sbírka zákonů. 2. 10. 1991 [cit. 2013-03-23]. ISSN 1211 – 1244
- [23] Zákon č. 300/2008 Sb., o elektronických úkonech a autorizované konverzi dokumentů.  
In: Sbírka zákonů. 17. 7. 2008 [cit. 2013-03-23]. ISSN 1211 – 1244

# Příloha A

## Obsah CD

- Zdrojové texty aplikace v adresáři `src`.
- Spustitelný soubor `VisReg.exe` s potřebnými knihovnami v adresáři `bin`.
- Uživatelský manuál v souboru `manual.txt`.
- Programová dokumentace v adresáři `doc`.
- XML schémata se zvýrazněním použitých částí v adresáři `xml`.
- Elektronická verze této bakalářské práce v souboru `bp.pdf`.