

Slovenská technická univerzita v Bratislave
Fakulta informatiky a informačných technológií
Ilkovičova 2, 842 16, Bratislava 4

Deep Search

Dokumentácia k tímovému projektu

(Inžinierske dielo)

Vedúci tímu: Ing. Nadežda Andrejčíkova, PhD.

Členovia tímu: Bc. Peter Berta, Bc. Matej Adamov, Bc. Michal Krempaský, Bc.

Oliver Macko, Bc. Bronislava Pečíková

Akademický rok : 2017/2018

Obsah

1. Úvod	3
2. Globálne ciele projektu	3
3. Celkový pohľad na systém	3
3.1. Modul analýzy vstupov	5
3.1.1. Výber databázy pre ukladanie neštruktúrovaných dát	7
3.1.2. Výber databázy pre ukladanie štruktúrovaných dát	7
3.2. Modul spracovania textu	8
3.2.1. Požiadavky	8
3.2.2. Analýza prístupov cez jednotlivé služby	8
3.2.3. Návrh	11
3.2.4. Implementácia	11
3.3. Modul identifikácia entít	14
3.4. Modul správa používateľov	15
3.5. Modul používateľské rozhranie	16
3.5.1. Prihlásenie	16
3.5.2. Registrácia	17
3.5.3. Hlavná obrazovka	18
3.5.4. Vyhľadávanie	19
3.5.5. Vyhľadávanie vzťahov	19
3.5.6. Spracovanie textu	20

1. Úvod

Informačná explózia so sebou prináša aj viacero problémov. Napriek tomu, že v dnešnej dobe si nemožno sťažovať na nedostatok informácií, máme často problém nájsť to, čo práve potrebujeme. Väčšina dokumentov je totiž v neštruktúrovanej podobe a získať z nich informácie typu: kto v danom období pôsobil v určitom regióne je prakticky nemožné. Fulltextové vyhľadávanie má jeden vážny nedostatok a to, že nezohľadňuje sémantiku daných kľúčových slov. V našom projekte sa zameriavame na spracovanie prirodzeného jazyka a stanovili sme si pomerne ambiciózný cieľ, ktorým je extrakcia štruktúrovaných dát z neštruktúrovaného textu so zachytením ich významu. Pričom zameriavať sa budeme najmä na životopisy a iné dokumenty, z ktorých budeme môcť extrahovať informácie o tom kde a kedy dané osoby študovali, prípadne pôsobili. Pokúsime sa tiež z textov získať vzťahy typu: kolega alebo spolužiak. Našou úlohou bude teda rozpoznávať a pokiaľ to bude možné aj jednoznačne identifikovať, entity typu osoba, korporácia, geografická jednotka, dátácia a zároveň identifikovať udalosť štúdiom, prípadne pôsobenie a v rámci nich vzťahy medzi týmito entitami. Cieľom je tieto údaje uložiť v štruktúrovanej podobe tak, aby bolo možné v nich vyhľadávať a získať informáciu o tom kto a kedy v danom mieste študoval, s kým sa mohol poznať a vytvárať tak aj virtuálne komunity napr. pre určité zameranie.

2. Globálne ciele projektu

Cieľom tohto semestra je vytvorenie databázy na základe neštruktúrovaných životopisov. Je potrebné identifikovať nasledovné polia:

- Štúdium – dátum, inštitúcia, miesto, profesori, spolužiaci
- Práca – dátum, inštitúcia, miesto, kolegovia

V ideálnom prípade by bolo vhodné taktiež vizualizovať súvislosti medzi osobnosťami pomocou grafu. Jednotlivé vrcholy budú predstavovať osobnosti a hrany budú predstavovať súvislosti medzi nimi.

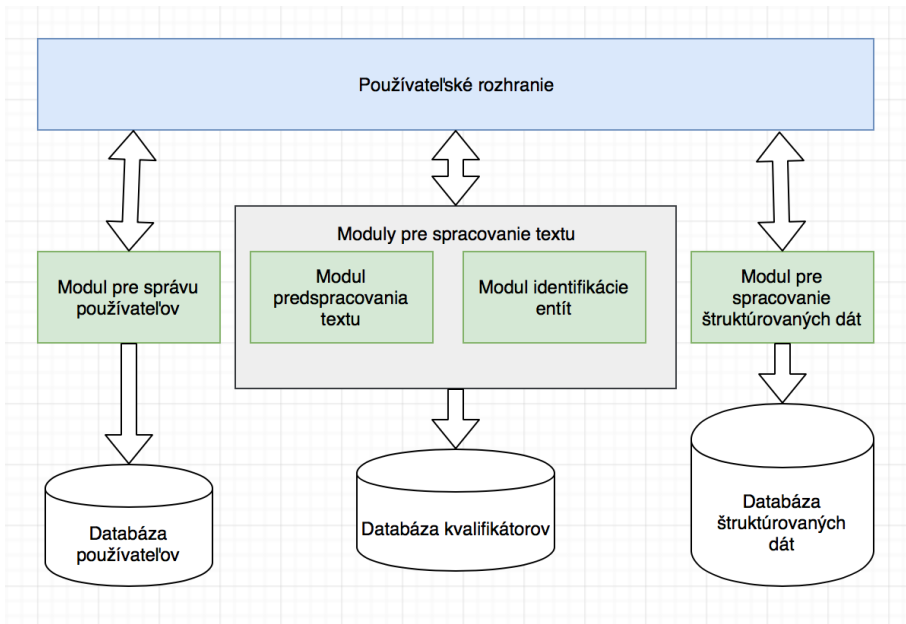
3. Celkový pohľad na systém

Systém pozostáva z troch modulov menovite sú to:

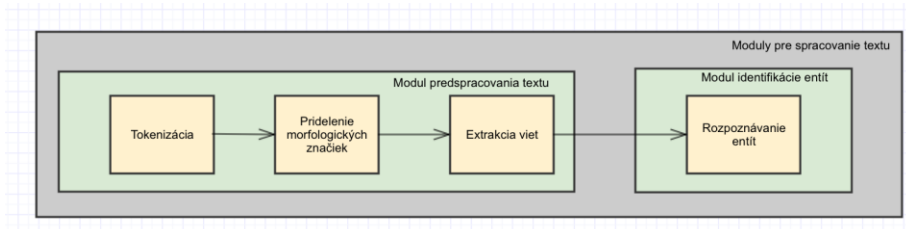
- modul používateľského rozhrania
- modul správa používateľov
- modul predspracovanie textu
- modul identifikácia entít
- modul pre spracovanie štruktúrovaných dát

Modul pre používateľské rozhranie bude implementovaný prostredníctvom webovej aplikácie vo frameworku Django a bude umožňovať používateľovi vyhľadávanie v štruktúrovaných dátach, zobrazenie grafu vzťahov, pridanie životopisu a mnohé ďalšie funkcionality (Obr. 3). Podrobnosti implementácie používateľského rozhrania sú rozpracované v kapitole modul používateľské rozhranie. Modul správa používateľov bude ponúkať metódy ako sú prihlásenie, odhlásenie a registráciu používateľa. Najdôležitejšími modulmi systému sú nepochybne modul predspracovanie textov a modul identifikácie entít. V implementácii spomínaných dvoch modulov bol využívaný architektonický štýl dátovody a filtre (obr.2). Modul pre spracovanie štruktúrovaných dát pokrýva funkcionality spojenú s ukladaním štruktúrovaných dát a dopytovaním sa nad týmito dátami a to najmä pre účely ich vizualizácie na používateľskom rozhraní (teda webovej aplikácii). Okrem

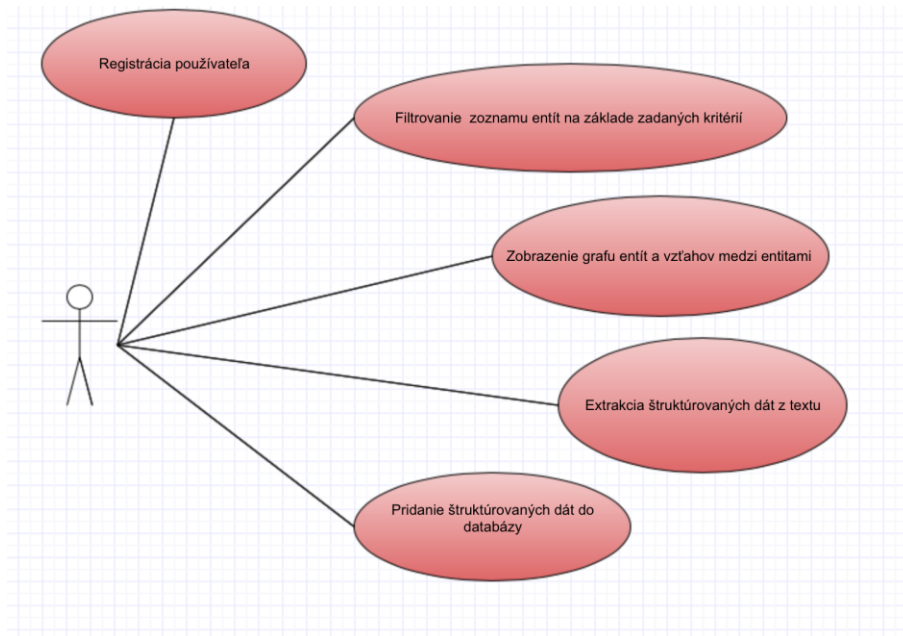
spomínaného bude tento modul zabezpečovať aj kompatibilitu s CIDOC CRM. Vzájomná interakcia uvedených modulov je zobrazená v diagrame Obr. 1.



Obr. 1: Moduly systému



Obr. 2: Architektonický štýl dátovody a filtre v projekte Deepsearch



Obr. 3 diagram prípadov použitia

3.1. Modul analýzy vstupov

V rámci analýzy vstupných textov sme identifikovali kvalifikátory pre štúdium a zamestnanie. Manuálne sme spracovali približne 100 životopisov, v ktorých sme zachytili rôzne vetné skladby v okolí identifikovaných kvalifikátorov. Z tejto množiny sme vytvorili univerzálnu štruktúru pre zachytenie výskytu kvalifikátorov spolu s pomenovanými entitami v ich okolí.

Pri zaznamenávaní pomenovaných entít sme použili dvojstupňovú kvalifikáciu pomenovaných entít pre český jazyk z článku “Czech Named Entity Corpus and SVM-based Recognizer”¹.

Skratky znázorňujú entity ktoré sa môžu nachádzať v okolí kvalifikátora, pričom čísla znázorňujú interval vzdialeností tejto entity od pozície kvalifikátora.

¹ KRAVALOVÁ, Jana; ŽABOKRTSKÝ, Zdeněk. Czech named entity corpus and SVM-based recognizer. In: Proceedings of the 2009 Named Entities Workshop: Shared Task on Transliteration. Association for Computational Linguistics, 2009. p. 194-201

Štúdium	Zamestnanie
studovať <IC -1 +2> <GU +2> <TY -1 +2> vystudovať <IC -1 +2> <GU +2> <TY -1 +2> absolvovať <IC +1> <GU +2> <TY -1> absolvováni <IC +1> <GU> <TY> být žák <IC -1> <GU -2 +3> <TY -3 +4> studium <IC +2> <GU +1 +3> <TY -1 +1> nastoupit <IC +1> <GU +2> <TY -1> vzdelání <IC +1> <GU +3> <TY +2>	pracovať <IC +2> <GU +2> <TY -1> být pracovník <IC> <GU> <TY> <IF +1> působit <IC -1 +1> <GU +3> <TY -1 +1> <IF> člen <IC +1> <GU +2> <TY -1> <IF> zakladatel <IC +1> <GU -1 +1> <TY> <IF> spoluzakladatel <IC +1> <GU +2> <TY +3> <IF> zamestnanec <IC> <GU> <TY -1> <IF +1>

Tab. 1. Kvalifikátory

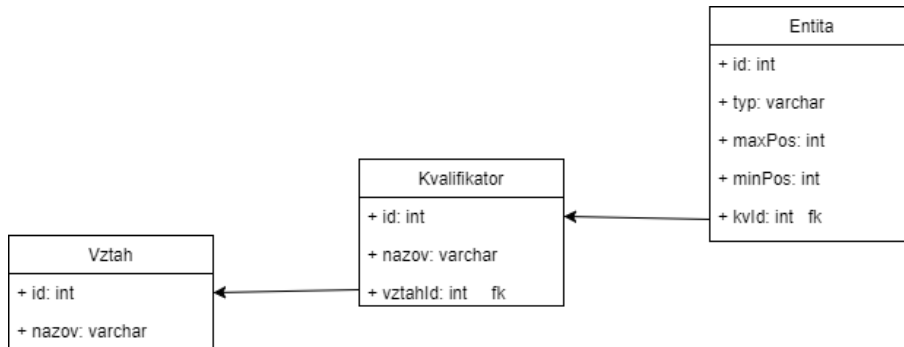
Túto štruktúru sme manuálne simulovali nad množinou nových životopisov a výsledky sme v nich farebne vizualizovali. Táto simulácia dosiahla 80% úspešnosť v rámci zachytenia kvalifikátora definovanou štruktúrou.

Hyliš, Petr,
Sochař Petr Hyliš se narodil 6. 3. 1956 v Havlíčkově Brodě jako syn
sochaře Karla Hyliše. V letech 1971-1975 vystudoval Střední
průmyslovou školu kamenickou a sochařskou v Hořicích. Během
středoškolského studia navštěvoval soukromé hodiny v ateliéru prof.
Karla Lidického. Ve studiích pokračoval v letech 1975-1981 na Vysoké
škole uměleckoprůmyslové v Praze u prof. Otto Eckerta. Vyučoval na
Pedagogické fakultě Jihočeské univerzity v Českých Budějovicích. Je
členem Unie výtvarných umělců České republiky, Sdružení výtvarných
umělců Vysočiny, Sdružení sochařů Čech, Moravy a Slezska a tvůrčí
skupiny EN FACE '91. Účastnil se mnoha kolektivních výstav u nás i v
zahraničí. Velkou samostatnou výstavu uspořádal v r. 2001 v několika
regionálních galeriích. Některá ze svých děl realizoval v
architektuře ve veřejných prostorech. Žije a pracuje v Havlíčkově
Brodě.

- kvalifikátor
- institúcia
- geografické pomenovanie
- datácia
- problémová oblasť

Obr. 3: Simulácia

Vzhľadom na výsledky simulácie tejto štruktúry, sme navrhli model relačnej databázy, ktorý sme následne implementovali, a naplnili doposiaľ identifikovanými kvalifikátormi.



Obr. 4: Relačný model

3.1.1. Výber databázy pre ukladanie neštruktúrovaných dát

Pre ukladanie neštruktúrovaných dát sme zvažovali tri alternatívy: ukladanie v databáze *ElasticSearch*, *Postgre* alebo ukladanie životopisov v klasických textových súboroch. Z dôvodu nedostatku pamäte na servery sme alternatívu *ElasticSearch* museli vylúčiť a napokon sme sa rozhodli pre *Postgre* nakoľko a tejto databáze máme uložené aj kvalifikátory a textové súbory by v budúcnosti nemuseli byť postačujúce.

V rámci analýzy dostupných možností pre ukladanie neštruktúrovaných dát sme vykonali aj proof of concept pre identifikáciu a pomenovanie entít prostredníctvom databázy *ElasticSearch*. Na základe tejto analýzy sme vyhodnotili, že databáza *ElasticSearch* nie je pre tento účel postačujúca, nakoľko jej posledná verzia neobsahuje funkcionality potrebnú pre identifikáciu a pomenovanie entít.

3.2. Modul spracovania textu

3.2.1. Požiadavky

Za účelom extrakcie poznatkov z vety je potrebné vytvoriť niekoľko nástrojov umožňujúcich zmysluplné dopytovanie nad textom. Pre tieto účely potrebujeme niekoľko metód, ktoré budú schopné spracovávať text a vytvárať množinu poznatkov o slovách vo vetách. Potrebné je vykonať nasledovné kroky:

- Tokenizácia - text potrebujeme vo forme tokenov - samostatných slov
- Lematizácia - potrebujeme určiť základný tvar slov aby sme mohli vyhľadávať kvalifikátory
- Pridelenie morfológických značiek - je potrebné každému slovu prideliť morfológické značky, ktoré nám následne umožnia identifikovať jeho slovný druh.

3.2.2. Analýza prístupov cez jednotlivé služby

Pre väčšinu programovacích jazykov existuje podpora prístupov k slovníkom obsahujúcim tieto vedomosti prostredníctvom webových služieb alebo voľne dostupných knižníc. V nasledujúcej časti je uvedená analýza dostupných knižníc a webových služieb.

3.2.2.1. NLTK

Natural Language Tool Kit - je knižnica v *Python*e, určená primárne pre účely spracovania anglického textu. Obsahuje mnoho slovníkov a funkcionalít, slúžiacich na generovanie tagov, lem, identifikáciu slovných druhov alebo tokenizáciu.

3.2.2.2. Lemmagen

Jedná sa o český voľne dostupný morfológický slovník a značkovač určený pre spracovanie textu v českom jazyku. Väčšina služieb tejto knižnice vykonáva morfológickú analýzu nad vloženým textom a ako výstup poskytuje morfológické značky, tokeny a lematizované formy slov na základe natrénovaného lingvistického modelu.

3.2.2.3. MorphoDita

Táto webová služba je dostupná aj cez rozhranie REST API, ktoré nám umožňuje na základe metód tokenize, analyze a tag vrátiť nad poslaným textom výstup vo forme JSON, kde je každé slovo tokenizované s pridelenou lemov a morfológickou značkou.

Ponúka webové rozhranie s REST API ako aj knižnicu v *C++* na lokálne použitie. Príklad použitia na webe je zobrazený na obrázku 5.



Token	Lemma	Tag
Byl	být	VpYS---XR-AA---
jsem	být	VB-S---1P-AA---
venku	venku	Db-----

Obr. 5: Príklad použitia na webe


```

model: "czech-morfflex-pdt-161115"
▼ acknowledgements:
  ▼ 0: "http://ufal.mff.cuni.cz/morphodita#morphodita_acknowledgements"
  ▼ 1: "http://ufal.mff.cuni.cz/morphodita/users-manual#czech-morfflex-pdt_acknowledgements"
▼ result:
  ▼ 0:
    ▼ 0:
      token: "Děti"
      ▼ analyses:
        ▼ 0:
          lemma: "dítě"
          tag: "POS=N|SubPOS=N|Gen=F|Num=P|Cas=1|Neg=A"
        ▼ 1:
          lemma: "dítě"
          tag: "POS=N|SubPOS=N|Gen=F|Num=P|Cas=4|Neg=A"
        ▼ 2:
          lemma: "dítě"
          tag: "POS=N|SubPOS=N|Gen=F|Num=P|Cas=5|Neg=A"
      space: " "
    ▼ 1:
      token: "pojedou"
      ▼ analyses:
        ▼ 0:
          lemma: "jet"
          tag: "POS=V|SubPOS=B|Num=P|Per=3|Ten=F|Neg=A|Voi=A"
      space: " "
  -

```

Obr. 6: Příklad použití metody *analyze*

```

model: "czech-morfflex-pdt-161115"
▼ acknowledgements:
  ▼ 0: "http://ufal.mff.cuni.cz/morphodita#morphodita_acknowledgements"
  ▼ 1: "http://ufal.mff.cuni.cz/morphodita/users-manual#czech-morfflex-pdt_acknowledgements"
▼ result:
  ▼ 0:
    token: "Děti"
    lemma: "dítě"
    tag: "NNFP1-----A----"
    space: " "
  ▼ 1:
    token: "pojedou"
    lemma: "jet-1_^(pohybovat_se,_ne_však_chůzi)"
    tag: "VB-P---3F-AA----"
    space: " "
  ▼ 2:
    token: "k"
    lemma: "k-1"
    tag: "RR--3-----"
    space: " "
  ▼ 3:
    token: "babičce"
    lemma: "babička"
    tag: "NNFS3-----A----"
  ▼ 4:
    token: "."
    lemma: "."
    tag: "Z:-----"
    space: " "

```

Obr. 7: Příklad použitia metódy tag

3.2.2.4. KonText

Jedná sa o rozhranie umožňujúce pristupovať ku slovníkom alebo korpusom českého jazyka. Každý z týchto korpusov reprezentuje vedeckú prácu výskumu akademikov českých univerzít. Tie následne poskytujú niekoľko operácií nad svojimi korpusmi. Nevýhoda uvedeného prístupu je, že nie je dostupná knižnica alebo prístup, ktorý by nám jednoducho umožnil lokálne sa dopytovať nad týmto rozhraním. Niektoré operácie dokonca vyžadujú prihlasovanie sa na webovej stránke. Tento koncept považujeme za nevýhodný a pokúsime sa nájsť alternatívne prístupy.

3.2.2.5. Text.fijt

Jedná sa o stránku výskumu fakulty FIIT, ktorá sa zaoberá spracovaním slovenského jazyka. Na tejto stránke máme k dispozícii metódy tokenizácie, lematizácie, morfológických značiek a určovania slovných druhov. Tieto metódy sú prístupné vo forme webových služieb a tento prístup sme zhodnotili ako vhodný pre spracovávanie slovenského jazyka.

3.2.3. Návrh

Z analýzy vyplýva možnosť použitia viacerých prístupov. Ako najvodnejšie riešenie pre účely našej práce sme identifikovali MorphoDitu. Nakoľko očakávame rôznu postupnosť procesov

spracovávaní textu, je vhodné navrhnúť volanie týchto služieb v samostatných pod moduloch. Tento prístup nám umožní vyššiu flexibilitu definovania procesu spracovávaní textu. Pre tento účel sa javí vhodné použitie architektonického štýlu dátovody a filtre, ktorý umožňuje rýchlu adaptáciu a konfiguráciu procesu spracovávaní textu.

3.2.4. Implementácia

Realizácia návrhu prebehla v duchu architektonického štýlu kde sme vytvorili spracovanie vo forme vstupných dát ako "slovník" a následne aplikujeme metódu, ktorá sa chová ako filter a spracuje text. Tento výstup následne použije pre vstup nasledujúceho filtra. Tento proces vieme konfigurovať.

Implementovali sme aj prístup k rozhraniu MorphoDity, kde je implementácia služieb volajúcich REST API, ktoré vrátia spracovaný JSON vo forme ako je vidieť v analýze.

Komentár od [1]: preformulovať zrozumiteľnejšie

3.3. Modul identifikácia entít

Tento modul implementuje identifikáciu pomenovaných entít v predspracovanom texte. Identifikácia entít je realizovaná prostredníctvom nástroja NameTag.

Vstupy:

Vstupom pre tento modul je JSON, ktorý uchováva záznamy viet pre každý typ vzťahu. Každý záznam vety obsahuje pôvodné znenie vety a pozícia kvalifikátora, ktorý tento vzťah definuje.

Komentár od [2]: preformulovať

```
[[
  {
    'relation': 'študent',
    'sentences': [{
      'sentence': 'Absolvoval Gymnázium M. M. Hodžu v Liptovskom Mikuláši (1985-1989) a Univerzitu Mateja Bela v Banskej Bystrici (slovenský jazyk - dejepis) (1989-1995).',
      'position': 0
    }
  ]
},
  {
    'relation': 'zamestnanec',
    'sentences': []
  }
]].
```

Výstup:

Výstupné dáta sú vo forme tripletov. Triplet obsahuje subjekt (v našom prípade meno osoby), predikát (vzťah), a následne objekt v podobe identifikovanej pomenovanej entity.

```
{
  "subject":Stanislav Káľavský,
  "predicate": "študent",
  "object": "Gymnázium M. M. Hodžu"
}
{
  "subject":Stanislav Káľavský,
  "predicate": "študent",
  "object": "Liptovský Mikuláš"
}
{
  "subject":Stanislav Káľavský,
  "predicate": "študent",
  "object": "1985-1989"
}
{
  "subject":Stanislav Káľavský,
  "predicate": "študent",
  "object": "Univerzitu Mateja Bela"
}
{
  "subject":Stanislav Káľavský,
  "predicate": "študent",
  "object": "Banská Bystrica"
}
{
  "subject":Stanislav Káľavský,
  "predicate": "študent",
  "object": "1989-1995"
}
```

Komentár od [3]: nezodpoveda realite. Upraviť a rozvinut (podrobne opísať štruktúru výstupu <https://drive.google.com/drive/u/0/folders/0B7zISiFSY1P0TFFvTHBlbEdkcmM>)

3.4. Modul správa používateľov

Naša aplikácia nezahŕňa len vyhľadávanie nad štruktúrovanými dátami, ale aj samotné pridávanie nových textov pre spracovanie našim systémom. Potrebujeme teda zabezpečiť, aby sa do systému nedostali nesprávne a nepravdivé dáta. Toto dosiahneme zavedením jednoduchšej správy používateľov.

Definovali sme si 3 základné používateľské role:

- Admin
Admin je používateľ, ktorý spravuje systém. Prideluje práva registrovaným používateľom a validuje registráciu.
- Anonymný používateľ
Anonymný používateľ je používateľ systému, ktorý si chce vyhľadať nejakú konkrétnu informáciu v už spracovaných štruktúrovaných dátach. Takýto používateľ nemá práva na vkladanie vlastných dát do systému na spracovanie.
- Registrovaný používateľ
Registrovaný používateľ je taký používateľ, ktorý sa zaregistroval do systému, bola mu schválená registrácia adminom a následne sa prihlásil do systému. Takto používateľ, ktorému boli nastavené práva adminom, dokáže do systému vložiť vlastné dáta, ktoré náš systém následne spracuje do štruktúrovanej podoby.

Registrácia používateľa prebieha pomocou jednoduchého formulára kde používateľ vyplní svoje meno, priezvisko, emailovú adresu a zvolí si používateľské meno a prihlasovacie heslo. Vyplnené dáta systém spracuje a vytvorí používateľovi záznam v databáze používateľov. Admin používateľovi registráciu buď schváli alebo zamietne. Pri prihlasovaní používateľ vyplní používateľské meno a heslo, systém skontroluje či existuje registrovaný používateľ s danými prihlasovacími údajmi a na základe výsledku vyhodnotí prihlásenie ako úspešné alebo neúspešné.

3.5. Modul pre spracovanie štruktúrovaných dát

3.5.1. Výber databázy pre ukladanie štruktúrovaných dát

Štruktúrované dáta získané zo spracovania životopisov je potrebné v nejakej forme ukladať. K dispozícii sú grafové databázy, objektové alebo klasické relačné databázy. Výhodou grafových databáz je, že vedú dobre reprezentovať vzťahy medzi objektami. V našom prípade by to bolo užitočné pre ukladanie vzťahov ako sú kolega, zamestnanec, spolužiak. Z grafových databáz sme zvažovali najmä *Neo4j* a *Cayley*. Na záver sme však zvolili výstupný formát tripletov, ktorý nám umožňuje využiť diplomovú prácu Martina Habdáka, spolu s úložiskom dát.

3.5.1.1. Neo4J

Jedna z najpopulárnejších *open-source* databáz, Neo4J je vyvíjaná v jazyku JAVA. Táto databáza umožňuje vizualizáciu dát na základe dopytov v jazyku cypher. Neo4j má široké využitie, napríklad v telekomunikačných službách pri detekcii *fraudov*, sieťových a IT operáciách atď.

Výhody:

- dlho žijúci projekt s veľkou komunitou
- podpora pre veľké množstvo jazykov
- *end to end control* - napr. nie je závislá na externých úložiskách ako *Cayley*
- *RESTful api*

Nevýhody:

- nie je *cloud based* (nemusí sa vždy jednať o nevýhodu)
- nepodporuje *shardovanie*
- spoľatná pre komerčné použitie

3.5.1.2. Cayley

Cayley je neoficiálny *opensource* produkt od spoločnosti *Google* vyvíjaný v jazyku *Go*, inšpirovaný grafovou databázou *Freebase*. Cieľom databázy *Cayley* je vytvorenie tzv. *toolboxu* pre pracovanie s prepojenými resp. grafovými dátami (sémantický web, sociálne siete, atď.).

Výhody:

- podporuje viac *query* jazykov ako napr. *Gizmo*, *GraphQL* alebo *MLQ*
- zabudovaný *query* editor a vizualizér
- *RESTful API*
- podpora viacerých *backend* úložísk: kvl (*Bolt*, *LevelDB*), *NoSQL* (*MongoDB*), *SQL* (*Postgres*, *CocroachDB*, *MySQL*)
- modulárny design (jednoduché rozširovanie s novými jazykmi a *backend* riešeniami)
- veľký dôraz na rýchlosť

Nevýhody:

- pomerne mladý projekt - ku dnešnému dňu má iba 1130 *commitov* a 59 prispievateľov
- neúplná dokumentácia
- *storage* rieši v externých databázach (môže byť aj výhoda)

3.5.2. Výber Neo4j Python driver-u

Pre účely integrácie systému s databázou Neo4j sme zvolili driver Neomodel. Nakoľko tento poskytuje všetku funkčnosť potrebnú pre účely vytváraného produktu.

Komentár od [4]: TO DO: doplniť

3.5.3. Dátový model

Okrem vhodnej prezentácie extrahovaných dát sme si dali za cieľ aj to aby bol náš systém bol kompatibilný s CIDOC CRM. Z tohoto dôvodu sme sa rozhodli vytvoriť v úložisku pre štruktúrované dáta dva modely - jeden pre účely prezentácie štruktúrovaných dát a druhý pre účely kompatibility z CIDOC CRM.

3.5.3.1. Prezentačný model

3.5.3.2. CIDOC CRM model

3.5.4. Import dát

3.5.4.1. Import školských organizácií

3.5.4.2. Import geografických entít

3.5.4.3. Prepojenie entít vzťahmi

3.6. Modul používateľské rozhranie

Systém bude sprístupnený používateľom v podobe webového rozhrania a bude ponúkať nasledovnú funkčnosť:

- vyhľadávanie informácií o zadaných entitách
- filtrovanie zoznamu entít na základe zadaných kritérií

- spracovanie a zobrazenie vlastného textu do štruktúrovanej podoby
- pridanie štruktúrovaných dát do databázy

Každá funkcionálnosť je dostupná na základe typu účtu pod ktorým je používateľ prihlásený. Preto je potrebné vytvorenie obrazoviek na prihlasovanie a registráciu, ktoré budú spolu s modulom správy používateľov poskytovať možnosť prihlasovania a správy účtu.

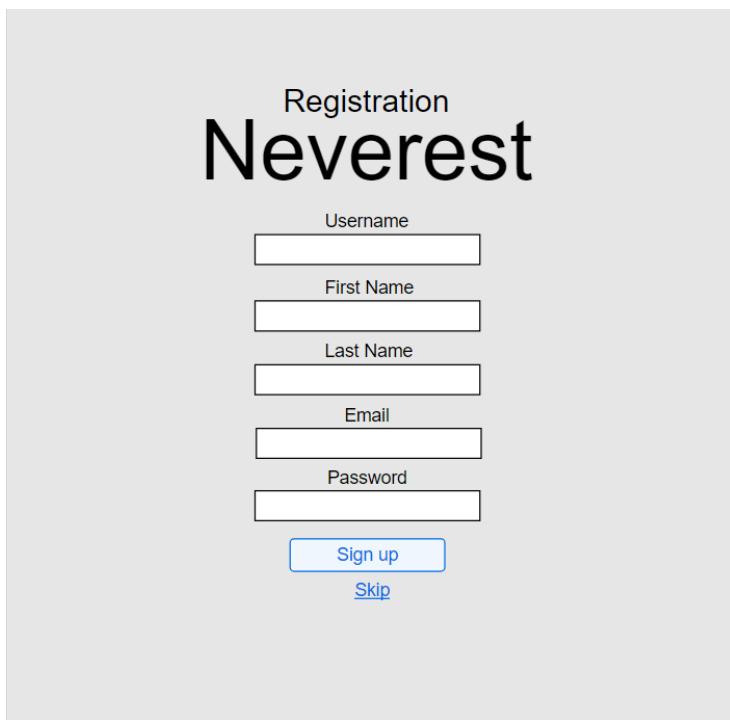
3.6.1. Prihlásenie

V obrazovke prihlasovania používateľ zadá svoje používateľské meno a heslo a pomocou tlačidla „Log in“ sa prihlási do systému. Nižšie pod týmto tlačidlom je dostupná registrácia nových používateľov, a taktiež preskočenie prihlásenia pre anonymných používateľov, ktorým je následne poskytnutá obmedzená funkcionálnosť systému.

Obr. 11: Obrazovka prihlásenia

3.6.2. Registrácia

Registrácia používateľa sa vykonáva na základe zadaného používateľského mena, hesla a emailu. Používateľ do formulára vyplní tieto údaje, a odošle registráciu tlačidlom „Sign up“. V prípade, ak používateľ nemá záujem o registráciu, má možnosť ju preskočiť a prístupíť k systému anonymne s obmedzenou funkcionálnosťou. Na tento úkon slúži odkaz „Skip“.



Registration
Neverest

Username

First Name

Last Name

Email

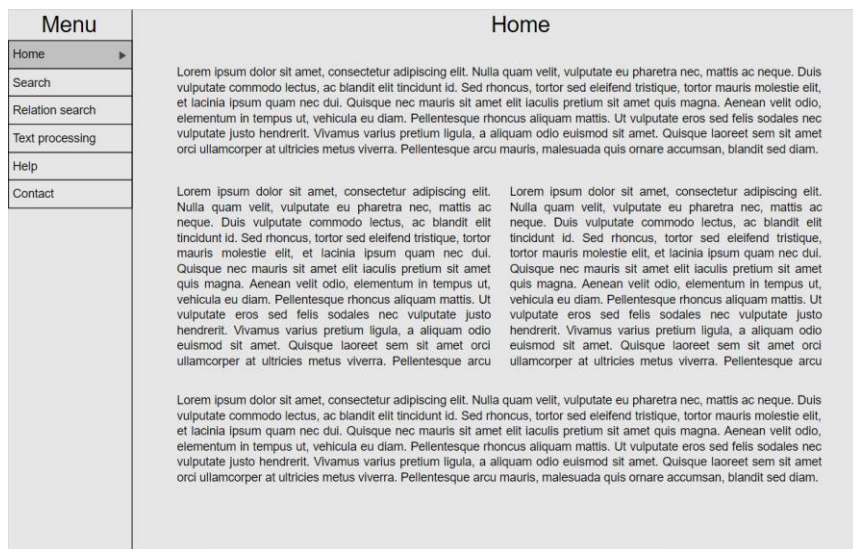
Password

[Skip](#)

Obr. 12: Obrázok registrácie

3.6.3. Hlavná obrazovka

Hlavná obrazovka je vertikálne rozdelená na dva panely. V ľavom paneli sa nachádza menu na vyber požadovaných funkcií. Tie sa následne zobrazujú v pravom paneli.



Obr. 13: Hlavná obrazovka

3.6.4. Vyhľadávanie

Funkcionalita vyhľadávania je v podobe jednoduchého vyhľadávacieho poľa v databáze na základe kľúčových slov zadaných používateľom. Používateľovi sa zobrazí zoznam výsledkov, v ktorom si následne môže zobrazíť detailne informácie o požadovaných entitách.

3.6.5. Vyhľadávanie vzťahov

Pri vyhľadávaní vzťahov si používateľ zadáva viacero kritérií vyhľadávania. Tieto kritéria sú v používateľskom rozhraní reprezentované filtrami, ktoré je možné pridávať a odoberať podľa potreby. V týchto filtroch používateľ vie definovať časovú os, korporáciu, miesto, vzťah, osobu, podľa toho čo vyhľadáva. Výsledkom tohto vyhľadávania je kolekcia všetkých záznamov, ktoré zodpovedajú zadaným kritériám.

The screenshot shows a web application interface with a search function. On the left is a vertical 'Menu' with links: Home, Search (highlighted with a right-pointing arrow), Relation search, Text processing, Help, and Contact. The main area is titled 'Search' and contains a search bar with the text 'Ludovít Štúr' and a 'Go' button. Below the search bar is a large white box containing the following information:

- Date of birth:** 28. októbra 1815, Uhrovec
- Date of death:** 12. januára 1856, Modra
- Books:** Slovanstvo a svet budúcnosti, Nauka reči Slovenskej, Dielo, VIAC
- Siblings:** Karol Štúr, Ján Štúr, Samuel Štúr, Karolína Štúrová
- Parents:** Samuel Štúr, Anna Štúrová
- Studies:** Evanjelické lýceum (1829–1836)

Obr. 14: Obrazovka vyhľadávania

3.6.6. Spracovanie textu

Spracovanie textu je nástroj webového rozhrania, ktorý umožňuje používateľovi z neštruktúrovaného textu životopisu vyťať informácie o štúdiu. Výsledné informácie sú vo formáte XML. Na spracovanie textu slúži tlačidlo „Process“. Tlačidlo „Upload“ slúži na vloženie štruktúrovaných dát do databázy. Táto možnosť je dostupná iba používateľom s potrebnými právami prístupu.

▼ Name	▼ Surname	▼ Date	▼ Corporation	▼ Position
Matej	Adamov	2014-2017	FIIT STU	Student
Peter	Berta	2014-2017	FIIT STU	Student
Michal	Krempaský	2014-2017	FIIT STU	Student
Oliver	Macko	2014-2017	FIIT STU	Student
Broňa	Pečíková	2014-2017	FIIT STU	Student

Obr. 15: Vyhľadávanie podľa vzťahov

Menu	Text processing
Home	Input - text: <input type="text"/>
Search	
Relation search	<input type="button" value="Process"/>
Text processing ▶	
Help	Output - XML: <input type="text"/>
Contact	
	<input type="button" value="Upload"/>

Obr. 16: Obrazovka spracovania textu