FAKULTA INFORMAČNÍCH TECHNOLOGIÍ ČVUT V PRAZE

ZADÁNÍ BAKALÁŘSKÉ PRÁCE

Název:	Aplikace pro prezentaci dat z hodnocení výzkumných organizací
Student:	Pavel Švagr
Vedoucí:	Ing. Karel Klouda, Ph.D.
Studijní program:	Informatika
Studijní obor:	Webové a softwarové inženýrství
Katedra:	Katedra softwarového inženýrství
Platnost zadání:	Do konce letního semestru 2018/19

Pokyny pro vypracování

1) Prostudujte metodiku hodnocení vědy pro předchozí roky a data dostupná na stránkách rvvi.cz.
2) Navrhněte databázi pro uložení těchto dat a vytvořte nástroj, který automaticky provede import dat do této databáze.
3) Prostudujte existující nástroje prezentující tato data.
4) Navrhněte, implementujte a otestujte webové rozhraní k této databázi, které umožní mimo jiné toto:
- vyhledávat v datech podle zvolených kritérií;
- získat přehled o kvantitativním vědeckém výkonu jednotlivých osob a výzkumných organizací a pracovišť;
- odhalit vybrané nekonzistence a chyby v datech.
5) Zvažte napojení nástroje na systémy Scopus a Web of Science.

Seznam odborné literatury

Dodá vedoucí práce.

Ing. Michal Valenta, Ph.D.
vedoucí katedry

doc. RNDr. Ing. Marcel Jiřina, Ph.D.
děkan

V Praze dne 19. prosince 2017

FAKULTA
INFORMAČNÍCH
TECHNOLOGIÍ
ČVUT V PRAZE

Bakalářská práce

Aplikace pro prezentaci dat z hodnocení výzkumných organizací

Pavel Švagr

Katedra softwarového inženýrství
Vedoucí práce: Ing. Karel Klouda, Ph.D.

14. května 2018

Poděkování

Chtěl bych poděkovat Ing. Karlu Kloudovi, Ph.D za vedení mé bakalářské práce a ochotu mi vždy se vším pomoci. Další obrovské poděkování patří mé rodině, která mi dala příležitost studovat na vysoké škole a byla mi vždy velkou oporou, a také mé přítelkyni a jejím rodičům, kteří mě po dobu celého studia velmi podporovali.

Prohlášení

Prohlašuji, že jsem předloženou práci vypracoval(a) samostatně a že jsem uvedl(a) veškeré použité informační zdroje v souladu s Metodickým pokynem o etické přípravě vysokoškolských závěrečných prací.

Beru na vědomí, že se na moji práci vztahují práva a povinnosti vyplývající ze zákona č. 121/2000 Sb., autorského zákona, ve znění pozdějších předpisů. V souladu s ust. § 46 odst. 6 tohoto zákona tímto uděluji nevýhradní oprávnění (licenci) k užití této mojí práce, a to včetně všech počítačových programů, jež jsou její součástí či přílohou, a veškeré jejich dokumentace (dále souhrnně jen „Dílo"), a to všem osobám, které si přejí Dílo užít. Tyto osoby jsou oprávněny Dílo užít jakýmkoli způsobem, který nesnižuje hodnotu Díla, a za jakýmkoli účelem (včetně užití k výdělečným účelům). Toto oprávnění je časově, teritoriálně i množstevně neomezené. Každá osoba, která využije výše uvedenou licenci, se však zavazuje udělit ke každému dílu, které vznikne (byť jen zčásti) na základě Díla, úpravou Díla, spojením Díla s jiným dílem, zařazením Díla do díla souborného či zpracováním Díla (včetně překladu), licenci alespoň ve výše uvedeném rozsahu a zároveň zpřístupnit zdrojový kód takového díla alespoň srovnatelným způsobem a ve srovnatelném rozsahu, jako je zpřístupněn zdrojový kód Díla.

V Praze dne 14. května 2018 ……

Odkaz na tuto práci

Švagr, Pavel. *Aplikace pro prezentaci dat z hodnocení výzkumných organizací.* Bakalářská práce. Praha: České vysoké učení technické v Praze, Fakulta informačních technologií, 2018.

Abstrakt

Tématem bakalářská práce je zpracování otevřených dat z hodnocení vědeckých výsledků v České republice publikovaných Radou pro výzkum, vývoj a inovace. Práce se zabývá analýzou datových souborů, při které odhaluje nejzávažnější nekonzistence a chyby, a implementací modulu pro jejich zpracování. Následně se zaměřuje na analýzu, návrh a implementaci aplikace pro prezentaci zpracovaných dat, která umožní vyhledávání výsledků v hodnoceních a zobrazí přehledy vědeckých aktivit výzkumných organizací, pod ní spadajících jednotek a autorů.

Klíčová slova hodnocení vědeckých výsledků, Rejstřík informací o výsledcích, vědecká činnost, analýza dat, reporting, VaVaI

Abstract

The subject of this bachelor thesis is the processing of an open data set derived from ratings of research results published by Research, Development and Innovation Council. This thesis deals with the analysis of files, implementation of parsing module and reveals inconsistencies and errors. Subsequently it is focused on analysis, design and implementation of a web application which enables seaching in ratings and displays an overview of scientific activity of reseach organizations, their units and authors.

Keywords evaluation of research results, Index of information about results, scientific activity, data analysis, reporting, VaVaI

Obsah

Seznam obrázků

Úvod

Hodnocení výsledků výzkumných organizací je dokument vycházející v ročním rozmezí[1] a poskytující informace o hodnocení Radou pro výzkum, vývoj a inovace (dále RVVI) k vědeckých výsledkům, které je stěžejním faktorem při rozdělování finančních prostředků všem výzkumným organizacím v České republice. [1]

Aktuálním místem pro prezentaci těchto hodnocení je Informační systém výzkumu, experimentálního vývoje a inovací [2] (dále IS VaVaI). Ten nabízí příslušné dokumenty pouze ve formátu XLSX a odděluje je od detailních informací o jednotlivých vědeckých výsledcích dostupných ze subsystému Rejstříku informací o výsledcích[2] (dále RIV), které poskytují výzkumné organizace.

Samotný systém umožňuje vyhledávat vědecké výsledky pouze dle jednotlivých kategorií, které jsou předdefinovány pro odevzdávaná data a neposkytuje žádná srovnání jejich hodnocení nebo činnosti organizací a autorů. Navíc prezentovaná hodnocení obsahují hned několik nekonzistencí a chyb, která jsou vzhledem k důležitosti těchto dokumentů a faktu, že slouží jako informační zdroj pro několik set až tisíce lidí, nezanedbatelné.

Zájem o téma vývoje reportingových[3] a analytických webových aplikací a možnost pomoci poskytnout organizovaná a ucelená data z IS VaVaI s možností viditelnosti nekonzistencí v datech byly hlavní motivací autora pro vypracování této bakalářské práce.

[1] Stát má ale často se zveřejňováním výsledků hodnocení výrazné zpožedění.

[2] `https://www.rvvi.cz/riv`

[3] Tj. podávající přehled o průběhu a stavu daných aktivit.

Cíle práce

Hlavním cílem této práce je návrh a implementace databáze a skriptu pro zpracování a uložení dat z hodnocení výsledků výzkumných organizací a následná implementace webové aplikace nad těmito daty. Ta by měla poskytnout přehlednější a ucelenější pohled na výskyty vědeckých výsledků v hodnocení a srovnání činnosti jednotlivých autorů v rámci organizací i celé České republiky a odhalení zásadních nekonzistencí při přepočítávání bodů.

V rámci zpracování dat je cílem získat přehled o formátování a struktuře dat, které jsou v rámci hodnocení z let 2013–2016 veřejně přístupné a zanalyzovat jejich použitelnost, případně jejich napojení na portály Web of Science[4] (dále WoS) či Scopus[5]. Na základě této analýzy by mělo být později možné data rozdělit do logických struktur (například na organizace, výskyty vědeckých výsledků a autory), které bude možné ve výsledné webové aplikaci třídit a vyhledávat.

Jedním z cílů je také snaha o podrobné seznámení autora s frameworkem Symfony a jazykem Python, ve kterých bude praktickou část této práce implementovat.

Struktura práce

Práce je rozdělena do tří hlavních částí. První popisuje pohled na data, jejich strukturu, nalezené nekonzistence a chby a zpracování těchto dat pro využití aplikace. Druhá část se věnuje samotné webové aplikaci pro prezentaci dat. Konkrétně analýze požadavků a stávajících řešení, návrhu aplikace a implementaci. Na závěr je věnována krátká kapitola zhodnocení dosažených výsledků včetně možnosti napojení aplikace a dat na další webové služby.

[4]Databáze odborných časopisů, dostupná z `https://webofknowledge.com/`.
[5]Databáze odborné recenzované literatury, dostupná z `https://www.scopus.com/`.

Kapitola 1

Analýza dat

Hodnocení pro výzkum vědeckých organizací vychází každoročně na webovém portálu Rady pro výzkum, vývoj a inovace (dále už jen RVVI) ve specifické struktuře, která se může v rámci let výrazně lišit a převážně se odvíjí od aktuálního usnesení Vlády České republiky pro určité časové období. Výchozím formátem je pro poslední roky formát XLSX. Tedy specifikace Office Open XML navržena firmou Microsoft a v dnešní době formát multiplatformní, používaný především ve spreadsheetových editorech, kde se s nimi pracuje jako s tabulkami [3].

Data jsou běžně rozdělena do souborů pro každou registrovanou výzkumnou organizaci[6] (dále VO) a poskytují informace o hodnocení všech vědeckých výsledků, které byly organizací pro toto časové období registrovány, dle aktuálně vydané metodiky Úřadem vlády ČR [4]. Metodika pro roky 2013–2016 byla schválená dne 19. 6. 2013 [5], nová metodika pro roky od 2017 stále není zcela dopracována [6]. Z toho důvodu je veškerá analýza zpracování dat v této práci řízena metodikou pro roky 2013–2016.

Každá organizace poskytuje data z vlastních webových portálů pomocí importu do RIV, či ručně pomocí administrace přímo v systému[7]. Rozsah těchto dat je často mnohem větší než se objevuje v samotných výsledcích hodnocení. Hodnocení jsou navíc zveřejňována ve speciální aplikaci a nejsou tedy přístupná přímo z vyhledávání v systému RIV.

[6] Oficiální seznam výzkumných organizací je dostupný na: `http://www.msmt.cz/vyzkum-a-vyvoj-2/seznam-vyzkumnych-organizaci`.

[7] Nápověda k rozhraní dostupná z: `https://www.rvvi.cz/is?s=napoveda`.

1.1 Struktura datových souborů

Soubory příslušné jednotlivým organizacím obsahují několik listů. Kromě úvodního listu s přehledem hodnocení celé organizace obsahuje tři další a to konkrétní hodnocení dle jednotlivých pilířů:

Pilíř I. Oborové hodnocení výsledků.

Pilíř II. Hodnocení kvality vybraných výsledků.

Pilíř III. Hodnocení patentů a nepublikačních výsledků aplikovaného výzkumu.

Hlavním z pilířů určující oborové hodnocení výsledků je Pilíř I. (v souboru hodnocení list `Tab3_Pil1`). Nachází se v něm přehled všech pod organizaci spadajících výskytů vědeckých výsledků, které jsou hodnoceny v rámci jejich oborové skupiny. Každá z těchto oborových skupin má předem určený bodový limit, který je pak v hodnocení podílově rozdělen mezi jednotlivé výsledky. V rámci prvního pilíře jsou hodnoceny následující typy výsledků:

J Článek v odborném periodiku. Dále se rozlišuje:

- $\mathbf{J_{imp}}$ Článek v časopise evidovaném na WoS.
- $\mathbf{J_{SC}}$ Článek ve zdroji evidovaném na Scopus a neevidován na WoS.
- $\mathbf{J_{neimp}}$ Článek v recenzovaném časopise v databázi ERIH[8] neevidovaném na WoS ani Scopus.
- $\mathbf{J_{rec}}$ Článek v českém recenzovaném časopise neevidovaném ve WoS, Scopus ani v ERIH.

B Odborná kniha

C Kapitola v odborné knize

D Článek ve sborníku

Názvy a počet rozdělení těchto typů se může v jednotlivých letech lišit dle aktuální upřesňující metodiky od RVVI, nicméně jejich význam zůstává stále stejný.

Ostatní pilíře jsou vzhledem k nízké roli v celkovém podílu na bodech, které daná organizace získá a malému množství hodnocených výsledků v rámci této práce vynechány. Jejich doplnění do systému je jeden z námětů pro rozšíření a vylepšení do budoucna.

[8] Databáze akademických časopisů dostupná z: `https://dbh.nsd.uib.no/publiseringskanaler/erihplus/`

Každému ohodnocenému výsledku odpovídá v tabulce nejméně jeden řádek (dále výskyt výsledku) za každou organizační jednotku, které je přiřazen (dále předkladatel výsledku), a za každého poskytovatele výsledku, který předložený výsledek financuje (dále jen poskytovatel).

Ve sloupcích jednotlivého výskytu výsledku jsou uvedena základní data k výsledku a jeho hodnocení. Každý z těchto sloupců má svůj vlastní název a univerzální zkratku, která je totožná v rámci všech tabulek. Přesný popis jednotlivých sloupců byl vydán naposledy v roce 2013 [8] a jejich význam není pro novější roky nikde vysvětlen, z toho důvodu je následující popis jednotlivých sloupců určen na základě jejich podrobného prostudování v rámci jednotlivých výzkumných organizací a na základě dokumentu popisující dodávku dat do systému RIV pro poskytovatele výsledku [7].

Vzhledem k velmi odlišné struktuře datového souboru z roku 2013 oproti dalším rokům, který by vyžadoval speciální pozornost při zpracování dat, a popsání struktury sloupců v tomto roce ve výše zmiňovaném dokumentu, jsou v této bakalářské práci zpracovány a popisovány pouze data z let 2014–2016.

1.1.1 Data o výsledku

Základní data k výskytům výsledku v souborech hodnocení z let 2014–2016 mají stejně definované zkratky sloupců i jejich počet.

Hodnoty většiny sloupců jsou uvedeny v tzv. konsolidovaných hodnotách. Ty jsou použity pro záznamy, které mají v RIV nejednoznačné hodnoty. (Například pro dvě výzkumné organizace nalezneme stejný výsledek s odlišným rozsahem stran). V takovém případě je jako konsolidovaná hodnota označena ta, která je v záznamech uvedena nejčastěji, v případě shodných počtů je to hodnota nejvyšší.

Následující seznamy obsahují název sloupce (odpovídající hodnocení z roku 2014) a v závorce je uvedena zkratka sloupce.

Konsolidovaný rok uplatnění výsledku (ROKUPLEFF) je rok, ve kterém byl výsledek uplatněn. Do hodnocení je přebrán ten, který má v době hodnocení nejčastější výskyt u jednotlivých záznamů v datech výzkumných organizací.

Druh výsledku v RIV (VYSDRUKOD) obsahuje zkratky pro jednotlivé druhy výsledku registrovaném v RIV. Ty se mohou lišit od výsledného druhu výsledku v hodnocení na záladě pravidel aktuální metodiky.

Příklad: *„D“*

Seznam domácích tvůrců výsledku (AUTSZNDOM) obsahuje tvůrce daného výsledku, oddělené středníkem (případně středníkem a mezerou), kteří měli pracovněprávní nebo studijní vztah k organizační jednotce nebo předkladateli, uvedeném ve stejném řádku výsledku, v roce uplatnění. V případě, že se

jedná o zahraniční či organizační jednotkou nezaměstnané autory, nevyžaduje RIV žádné informace.

Tedy v těchto datech, mohou být o takových autorech udány libovolné informace. Každý zadaný tvůrce zde má uvedeno své jméno, příjmení a v případě, že má i české občanství a bylo tedy zasláno do RIV jeho rodné číslo, i jedinečný identifikátor *vedidk* (hovorově „vědík“) generovaný z rodného čísla [9]. Tyto informace jsou odděleny čárkou (případně čárkou a mezerou).

Hodnota sloupce by logicky neměla být prázdná, nicméně se vyskytují i případy, kdy je zřejmě výsledek přiřazen k organizační jednotce, aniž by k ní žádný z autorů měl nějaký vztah, v tomto případě je zřejmě možná i prázdná hodnota.

Příklad: *„Novák, Jiří, 2921715; Novák, Pavel, 3432742; Mikš, Antonín, 2832143“ (Hodnocení pro ČVUT, rok 2014, řádek 13).*

Název výsledku v původním jazyce (VYSNAZORI). Vyplněn je vždy.

Příklad: *„Generalized Carré Multi-Step Phase-Shifting Algorithms“ (Hodnocení pro ČVUT, rok 2014, řádek 13).*

Kód jazyka výsledku (VYSJAZKOD) je zkratkou jazyka, ve kterém je výsledek napsán.

Například: *„eng“*
(Hodnocení pro ČVUT, rok 2014, řádek 13).

Obor výsledku (CR_OBR) je obor, ve kterém je výsledek hodnocen, ten se může lišit v rámci let u stejného výsledku na základě posouzení rady pro hodnocení. Případně se může jeho hodnota změnit v případě znovu odeslání stejného vědeckého výsledku k posouzení. V souborech je zmíněna buďto zkratka (rok 2015) nebo zkratka společně s názvem.

Například: *„BH - Optika, masery a lasery“*
(Hodnocení pro ČVUT, rok 2014, řádek 13).

Skupina oborů dle metodiky (RVVOBRSKU) označuje skupinu oborů, v rámci které je výsledek hodnocen. Tedy zároveň definuje počet bodů, který může daný výsledek získat. Tato hodnota by se neměla měnit vzhledem ke své návaznosti na metodiku hodnocení ani při opakovaných odesláních výsledku k posouzení. Zadán bývá dvoumístným číslem a názvem skupiny oborů.

Například: *„08 - Fyzikální vědy“*
(Hodnocení pro ČVUT, rok 2014, řádek 13).

Číslo patentu (PATCIS) je vyplněno pouze tehdy, pokud je k výsledku registrován patent.

WoS Accession Number (ISIUT_KOD) je číslo odkazující na výsledek ve službě Web of Science. Vyplněno je pouze pokud je výsledek v této službě registrován a pokud tento údaj byl odeslán do RIV.

Naříklad: *„000208219100001“*
(Hodnocení pro ČVUT, rok 2014, řádek 5536).

DOI výsledku (VYSDOI) je unikátní identifikátor pro výsledky dostupné v digitální podobě (převážně pro vědecké sborníky a časopisy). Vyplněn je pouze pokud byl údaj zaslán do RIV.

Například: *„10.1007/JHEP12(2010)060“*
(Hodnocení pro ČVUT, rok 2014, řádek 5536).

Webová stránka s plným textem výsledku (VYSWWW). Odkaz na vlastní stránky výsledku, pokud byla taková stránka do RIV s údaji zaslána.

Webová stránka s detaily výsledku v hodnocení (VYSWEBURL). Odkaz na detailní informace o výsledku do RIV. V případě roku 2014 odkazuje do původního systému IS VaVaI, jehož provoz byl v roce 2016 ukončen [10]. Uvedeno u všech výskytů vědeckého výsledku.

Identifikační kód výsledku s označením dodávky dat dle RIV (VYSIDKPER) je jedinečné id pro každý výskyt výsledku dle dodaných dat, organizační jednotky a poskytovatele výsledku, který projekt financoval. Jelikož jsou výskyty výsledků přepisovány do dalších let v rámci pětiletého okénka, jedná se identifikátor propojující záznamy během několika let. Ve svém plném znění obsahuje kód organizační jednotky i finančního poskytovatele. Zbylá čísla udávají indetifikaci vědckého výsledku v RIV.

Například: *„RIV/68407700:21260/10:00202487!RIV13-MSM-21260____“*
(Hodnocení pro ČVUT, rok 2014, řádek 5536)

První dvojice čísel po prvním zpětném lomítku oddělena dvojtečkou je IČO organizace a kód organizační jednotky. Za dalším zpětným lomítkem je část identifikátoru vědeckého výsledku z RIV a část za vykřičníkem je definice poskytovatele.

Identifikační kód celku sjednoceného výsledku (VYSNIDCEL) je vyplněn pouze pro knihy a kapitoly v knihách. Vygenerován pro celek ve kterém je výsledek obsažen.

Identifikační kód sjednoceného výsledku (VYSNID) je vygenerovaný identifikátor, který sjednocuje výskyty toho samého výsledku v rámci hodnocení z jednoho roku. Každý rok má vygenerovanou jinou sadu identifikátorů a hodnoty se tak pro jednotlivé výskyty v ročních hodnoceních ve většině případů liší.

Například: *„08h8i0gXFMDx9vPVi7LZ5uXmHSA=“*
(Hodnocení pro ČVUT, rok 2014, řádek 5536)

Kód stupně důvěrnosti údajů (STUDUVKOD) má hodnoty dle číselníků RIV. Uveden je vždy.

Například: *„S“*

Násobnost výsledku (NASOB) udává počet výskytů výsledku v RIV. Uvedena je vždy.

ISSN (ISSSPR) je jednoznačný osmiciferný identifikátor periodické publikace. Vyplněn je tehdy, pokud výsledek spadá pod periodické publikace.

Ročník periodika (PERROC) je soubor všech čísel periodika vydaných během jednoho roku (podle rozvržení vydavatele). Vyplněn je tehdy, pokud výsledek spadá pod periodické publikace.

Číslo periodika v rámci ročníku (PERROCCIS) je vyplněno tehdy, pokud výsledek spadá pod periodické publikace.

Název publikačního kanálu (DOKNAZPRI) udává název portálu, publikace, knihy či periodika, v rámci kterého byl výsledek zveřejněn či tvoří jednu z jeho částí.

ISBN (ISBSPR) je číselný kód určený pro jednoznačnou identifikaci knižních vydání. Vyplněn je tehdy, pokud výsledek spadá pod knižní publikace.

Rozsah stran (STRROZ) je vyplněn pokud výsledek spadá pod knižní publikace či periodika.

Počet stran výsledku (POCSTREFF) je vyplněn pokud tato informace byla nalezena v RIV.

Počet stran knihy (KNIPOCSTRE) je vyplněn pokud je výsledek součástí knihy a tato informace byla zaslána do RIV.

Edice nebo název svazku (EDINAZSVA) jsou zadány v případě že tato informace byla zaslána do RIV a výsledek spadá pod knižní publikace.

Název nakladatele (NAKNAZ) je vyplněn pouze pokud tato informace byla zaslána do RIV.

Příznak, že nakladatel je z ČR (PRINAKCZ) obsahuje hodnoty: „A“ (Ano) a „N“ (Ne). Je vyplněno pouze pokud informace byla nalezena v RIV.

Popis (BTBPOLPOP) je krátký doprovodný text k hodnocení. Například se může jednat o informaci o tom, zdali je hodnocení výsledku přebráno z předchozích let.

IČO (INSICOPKL) označuje identifikační číslo organizace. Zde identifikuje výzkumnou organizaci pod kterou je jednotka přiřazena. Jednotné pro daný soubor s hodnocením.

Název instituce (INSNAZPKL) označuje název výzkumné organizace. Název je vždy jednotný pro celý soubor s hodnocením stejně jako IČO.

Kód organizační jednotky (ORJKODPKL) identifikuje pod organizaci spadající jednotku (například fakulty). V případě, že organizace žádné jednotky nemá, či výsledek spadá pod celou organizaci, je tato hodnota prázdná.

Název organizační jednotky (ORJNAZPKL) je vyplněn tehdy, pokud je výskyt výsledku k nějaké přiřazen.

1.1.2 Data k hodnocení

Hodnocení samotné je uvedeno ve dvou kategoriích – celkové bodové ohodnocení výsledku a podíl výzkumné organizace. Veškerá data k hodnocení jsou uvedena v následujících sloupcích:

Druh výsledku v hodnocení (DRUHODKOD) označuje druh, v rámci kterého je výsledek ohodnocen a může se tedy lišit od druhu výsledku registrovaného v RIV.

Kód hodnocení (BTBPOLKOD) označuje kategorii do jaké byl výsledek zařazen. Například zdali patří výsledek mezi vyřazené či hodnocené.

Bodové ohodnocení výsledku (VYSBOD) je celkový bodový zisk v oboru pro předkladatele výsledku. V případě knihy je započítán i bodový podíl části na celé knize. Vyplněno pouze u posledního záznamu dodaného do RIV.

Upravené bodové ohodnocení výsledku (VYSKBO) označuje upravené body sjednoceného výsledku (připojeny například bonifikace za projekty či soutěže). Vyplněno pouze u posledního dodaného záznamu výsledku do RIV.

Podíl hodnocené VO na výsledku (VYSPDL) určuje procentuální podíl bodů předkladatele výsledku.

Bodový podíl hodnocené VO na výsledku (PKLBOD) je přepočítaný bodový zisk dle podílu.

Upravený bodový podíl hodnocené VO na výsledku (PKLKBO) označuje přepočítaný upravený bodový zisk na základě podílu.

1.2 Nalezené nekonzistence a chyby

Data obsažená převážně v listu s hodnocením dle Pilíře I obsahují hned několik nekonzistencí, jejichž smysl není pospán v metodice vyhodnocení, ani v jiném dokumentu vydaným RVVI, z toho důvodu je možné je považovat za chyby.

Jejich původ je zpětně těžko odhadnutelný, jelikož přímo v RIV se některé tyto chyby nevyskytují a chybovost jednotlivých dokumentů se v průběhu let liší společně s měnící se strukturou a formátem dat.

Tyto nedostatky zároveň mohou velice snadno způsobit problémy při jejich strojovém zpracování a z toho důvodu je nutné je dále řešit. Z technického pohledu je možné rozdělit je do tří kategorií: Chyby logické, chyby ve formátování a nekonzistence formátování mezi jednotlivými roky.

1.2.1 Logické chyby

Do chyb logických zařazuji takové, které brání v porozumění struktuře dat či mohou způsobit zmatení čtenáře. Základním znakem je například nekonzistence dat mezi jednotlivými záznamy nebo chybějící data tam, kde by uvedena být měla.

Chybně zadaná jména autorů

Jednou z těchto chyb jsou nepřesná a měnící se jména jednoho a toho samého autora v hodnoceních. U žen je to při vstupu do manželství pochopitelné. U jiných autorů se ale často vyskytují překlepy a chyby ve jménech i příjmeních. Jelikož data o autorech jsou zasílány jednotlivými organizacemi, zřejmě se jedná převážně o problém u poskytovatelů dat.

Například:

> *„Roman Horváth“ a „Roman Horvath“ (Hodnocení Moravské zemské knihovny v Brně z roku 2016) či „Ilona Matejko-Peterka“ a „Ilona Matejko Peterka“ (Hodnocené Slezského zemského muzea z roku 2016)*

Řešením takovéto situace je neodkazování se na autory s identifikátorem přes jejich jména, ale pouze přes jejich identifikátor. U autorů bez identifikátoru je jejich rozeznání prakticky nemožné, protože se může jednat o jiného autora s podobným jménem, o kterém je v jiném záznamu udána informace. Zpracovávání dat od nejnovějšího hodnocení dále pomůže vybrat nejaktuálnější jména žen.

Identifikátory autorů

Další a neméně matoucí chybou jsou identifikátory autorů samotné. Vzhledem k požadavkům, které RIV stanovuje, na data, která byla popsána výše při definici jednotlivých sloupců, je obsah s výčtem autorů prakticky volný. Jediná jistota tedy je, že u výskytu výsledku budou zmínění autoři spojení s danou organizací a pokud jsou to čeští autoři, bude zde zmíněno jejich *vedidk*. Problém nastává, když organizace uvede i další autory s organizací nespojené. U autorů bez českého občanství totiž povinnost udávat bližší údaje organizacím odpadá a není jim zakázáno udávat informace o autorech z jiných organizačních jednotek či organizací.

Jedna z takových situací je například pozorovatelná v datech ČVUT u prof. Ing. Róbert Lórencze, CSc., který jakožto domácí tvůrce Fakulty informačních technologií je ve výskytech výsledků uplatňovaných v letech 2009–2011 v hodnocení 2014 zaregistrován s *vedidk* a v letech od 2012 dále uváděn již bez *vedidk*. *Například hodnocení z roku 2014 pro ČVUT v Praze, řádek 10 723 je autor s vedidk a řádek 11 517 obsahuje autora bez vedidk*. Tato chyba údajně[9] vznikla při odesílání dat do systému RIV, kdy se v letech 2012–2015 odesílali zahraniční autoři bez podrobnějších osobních údajů (včetně rodného čísla) a v letech dřívějších byli odeslání s podrobnějšími údaji.

Závažnost tohoto problému tedy opět závisí na každé organizaci a její jednotce a je kvůli němu prakticky nemožné algoritmicky jistě určit pouze podle hodnocení identity jednotlivých autorů.

Řešením této situace může být určení skupiny dat, která budou autory bez *vedidk* rozeznávat, ovšem s vědomím, že se může jednat o rozeznání silně nepřesné. Takovou logickou skupinou dat se může jevit trojice: jméno, příjmení a výzkumná organizace pod kterou autor spadá. V případě dvou totožných jmen v rámci jedné organizace ale může stále dojít ke spojení dvou odlišných autorů. V tomto případě se nabízí zřejmě jediná další logická varianta – zapojení do původní trojice údajů ještě skupinu oborů.

Tak už je poměrně slušná šance, že nebudou sloučeni dva různí autoři, bohužel je ale zároveň i možný případ (pokud autor působí v organizaci napříč obory), že nebude rozeznán a vzniknou tak dva různí autoři, ač se jedná o tutéž osobu. Nicméně spojení autora s *vedidk* a autora bez něj je zřejmě logicky nemožné a takovéto případy by musely být řešeny jedině individuálně.

Zkrácení zápisu autorů

Dalším nedostatkem v hodnocení je vynechání některých podstatných informací. To může způsobit nepřesnosti v porozumění při zpracovávání dat. Jedna z těchto chyb může nastat v případě počtu autorů výsledku, kde je zápis zkrácen třemi tečkami na konci v případě, že výsledek má autorů více a tedy v těchto datech neobsahuje domácí autory všechny. Výsledky hodnocení zřejmě

[9] Informace získaná z rozhovoru s prof. Ing. Róbertem Lórenczem, CSc.

počítají s dohledáváním daných publikací přímo v RIV, případně nedostatek vznikl již při zadávání dat danou organizací. *Například zkrácený zápis autorů se třemi tečkami v hodnocení Fakultní nemocnice Brno z roku 2014, řádek 84.*

Tento problém zřejmě nelze řešit bez složitějšího zásahu, kterým je například dohledávání chybějících tvůrců v RIV či jiných zdrojích. V případě této práce, která zpracovává pouze soubor s hodnocením, zůstává problém tedy nevyřešen.

Chybějící data v bodovém hodnocení

V části 1.1.2 byl uveden seznam dat v hodnocení. Ve sloupcích s celkovými body výskytu a upravenými body výskytu se bodové ohodnocení vyskytuje jen v případě, že se jedná o nejpozději přidaný záznam do RIV.

To ovšem opět může při pohledu na data způsobit jisté zmatení. Obzvlášť v případě, že se celkové bodové hodnocení nevyskytuje u některých tabulek z organizací vůbec a jediným odkazem na celkový počet bodů je násobnost výsledku v RIV. Vše je tedy na čtenáři. Ten si musí buďto dohledat všechny výskyty výslekdu pro jiné organizace, abyse dostal k celkovému počtu bodů, případně si musí bodovou hodnotu dopočítat sám dle uvedených podílů na bodech pro organizační jednotku či organizaci. Vzhledem k tomu, že se v datech mohou vyskytovat chyby, nemá čtenář ani jistě zaručeno, že vypočítaná hodnota je v souladu s hodnotou, která je uvedena v záznamu s celkovým bodovým hodnocením.

Řešení takového problému v kontextu aplikace zpracovávající kompletní data není složitá. Možné je ukládat výsledek až v případě, kde je u něj dané hodnocení uvedeno. Případně hodnocení ukládat do jiné datové struktury než zbytek informací o výsledku.

1.2.2 Chyby ve formátování

Chyby ve formátování jsou pro data z RIV mnohem častější. Jejich původ může být jak různé nastavení kódování znakových sad (například přechod formátu UTF8 a Windows-1250), případně špatná validace na straně nástroje pro import dat z jednotlivých organizací.

Escapování znaků

Základní chybou, která může velmi komplikovat čtení dat strojem, je špatné escapování znaků. Prakticky ve všech záznamech z posledních tří let lze nalézt takové hodnoty sloupců, které mají nahrazené pouze některé speciální znaky, mají je nahrazené pouze někdy, nebo nejsou nahrazeny pomocí únikových znaků vůbec. Typickým příkladem jsou znaky apostrofů a uvozovek. Ty ve většině programovacích jazyků a také v regulárních výrazech zastupují funkci únikových znaků a nemusí se znaky speciální v nich umístěné znovu escapovat. Nalézt se dají ve všech zmiňovaných variacích:

- Nejsou v prefixu s únikovým znakem.
  Příklad: *„"UNSP 908741"“ (Hodnocení z roku 2014, Fakultní nemocnice Brno, řádek 75).*

- Jsou v prefixu s únikovým znakem jen někdy.
  Příklad: *„"054103-2\"-"054103-13\"“ (Hodnocení z roku 2014, Západočeská univerzita v Plzni, řádek 5153).*

- Jsou v prefixu v celé hodnotě sloupce.
  Příklad: *„Sborník příspěvků z odborné konference \"Zvýšení bezpečnosti provozu vozidel ozbrojených sil\"“ (Hodnocení z roku 2014, České vysoké učení technické v Praze, řádek 15 601).*

Pro stroj, který data čte a chce je dále zpracovávat, tak nastává problém, jelikož bez předzpracování nemůže se všemi hodnotami pracovat stejným způsobem. Například při vkládání do relační databáze, kde se textové řetězce uvozují právě apostrofy případně uvozovkami, pokud nebudou správně dané znaky v řetězci escapované, vznikne problém v syntaxi jazyka SQL.

Řešením by mohly být složitější překladače, které by výskyty takových znaků náležitě ošetřily, nicméně jejich implementace je složitější a režie s nimi spojená je zbytečná, jelikož se chyby nevyskytují tak často. Jednodušším řešením může být smazání všech únikových znaků (v tomto případě zpětného lomítka) a následné nahrazení obou problémových znaků. Problémem, který může nastat, je úmyslné zadání zpětného lomítka například do názvu vědeckého výsledku, tam bude totiž zpětné lomítko z názvu odstraněno. Tento případ bude zřejmě velmi ojedinělý a řešení tohoto problému tedy opět zůstává jako možné vylepšení zpracování do budoucna.

Chybně zadaná data

V některých případech je možné nalézt nesmyslné záznamy. Ty většinou obsahují překlepy nebo znaky, které do dané hodnoty nepatří. Tyto překlepy mohou stejně jako chyby v escapování znaků způsobit problémy při zpracovávání. Proto je nutné opět všechna tato data testovat proti výskytu problematických znaků a nahrazovat je. Problém ovšem zůstává nevyřešen, jelikož rekonstrukce původních dat zahrnuje opět větší režii mimo zpracování souboru s hodnocením.

Příklady:

Jméno autorky Marie Hübnerové, v záznamech uvedena jako „H"ubnerová“ (Rok 2014 organizace Fakultní nemocnice Brno, řádek 10).

Chybný odkaz na celé znění výsledku: „http://www.intechopen.com/books/multiple-myeloma-an-overview" title="Multiple Myeloma - An Overview">Multiple Myeloma - An Overview“
(Rok 2014 organizace Technická univerzita v Liberci, řádek 3 002).

Pomlčka místo vedidk: „Franky, ,-“
(Rok 2016 organizace Univerzita Karlova, řádek 3 274).

Udání zbytečného údaje navíc: „Hořava,ml., Vladimír, 4166515“
(Rok 2016 organizace Ostravská univerzita, řádek 1 323).

Chybné znaky pro cizojazyčné názvy výsledků

Zřejmě přechodem z jednoho formátu znakových sad do druhého, jak bylo zmíněno v úvodu této kapitoly, došlo k záměně některých znaků (například znaků azbuky) za znaky neplatné. Tato chyba není častá, nicméně opět není bez složitějšího zásahu (hledání přesného znění daného textu na různých portálech) řešitelná a tak zůstává v této práci nedořešena.

Příklad:

Název vědeckého výsledku: „? ????????? ???????? ??? ?? ??????? ???????????? ??????? : ?? ??? ???? ??? ?? ??????? ?????? 1946-1968“ (Hodnocení Univerzity Karlovy z roku 2015, řádek 18 573)

1.2.3 Nekonzistence formátování

V rámci jednotlivých let, které spadají pod stejnou metodiku hodnocení, můžeme nalézt změny ve formátování, aniž by to bylo explicitně řečeno při vydávání souborů s výsledky. Ač se jedná o chyby, které výrazněji nebrání porozumění, v případě zpracovávání těchto dat hromadně jedním strojem může dojít k chybě či duplikaci stejných dat. V letech 2013 – 2016 mohou být nalezeny následující změny:

Výpis oborů u kterých se liší formát v jednotlivých souborech s hodnocením. V některých je uveden celý název (tedy zkratka pomlčka a název oboru) a v některých je uvedena pouze zkratka.

Začlenění některých organizačních jednotek tak, že jednou se jedná o organizační jednotku, jindy však o výzkumnou organizaci. Například pro rok 2014, jsou odděleny ústavy Ministerstva vnitra do jednotlivých souborů. V roce 2016 jsou však tyto ústavy sloučeny do jednoho souboru a jsou odděleny jako organizační jednotky.

Oddělovače s bílými znaky jsou v průběhu let změněny. Například v roce 2014 jsou autoři odděleni středníkem a mezerou, v roce 2016 už bez mezery.

1.3 Metodikami odlišně definovaná data

Každý rok hodnocení se řídí upřesňující metodikou. To znamená, že některé struktury dat si v daných hodnoceních neodpovídají. Hned několik takových změn proběhlo při přechodu z hodnocení 2014 do roku 2015, a následující rok 2016 poté kombinuje definice z obou let předchozích.

Typy výsledků v hodnocení jsou pro roky 2014 a 2016 shodné s obecnou metodikou. V roce 2015 se rozlišují některé podtypy a jejich názvy jsou výrazně odlišné.

Například: *„Jimp“ (2014) a „1JI“ (2015).*

Kód hodnocení se liší ve všech letech a konkrétně v roce 2014 má výrazně odlišný formát. V letech 2015 a 2016 se přechází na jeden znak s případnou příponou podle poddruhu, které jsou pro oba roky lehce odlišné.

Například: *„uzn/H14/apl/P/vyssi/JP“ (2014) a „A“ (2015 a 2016).*

Tyto upřesňující roční metodiky zároveň znemožňují porovnávání výsledků z různých let. Získané body a upravené body se vždy řídí jinými pravidly a je proto nutné, aby uživatel aplikace měl alespoň základní přehled o významu dat.

Kapitola 2

Zpracování dat

Následující kapitola se věnuje zpracování dat. Nejdříve shrnuje jednotlivé požadavky na výsledný modul zpracovávající data a požadavky na strukturu výsledných dat. Poté se věnuje implementaci modulu, kde nejdříve popisuje zvolené technologie, řešení problémů z předchozí kapitoly a na závěr modul samotný.

2.1 Požadavky na zpracování dat

Na základě zadání této práce a analýze dat z předchozí kapitoly je možné vytyčit hned několik funkčních požadavků, které by měl implementovaný modul na zpracování dat splňovat.

2.1.1 Funkční požadavky na modul

(F1.1) Modul bude obsahovat rozhraní pro stažení dat z IS VaVaI. Toto rozhraní umožní zadat specifické roky (případně konkrétní url odkazy na organizace), které by měly být staženy a vytvoří datovou strukturu, která bude obsahovat data všech organizací v souborech pro daný rok.

(F1.2) Modul bude umět zpracovávat XLSX soubory. V rámci stahování dat je nutné zpracovávat XLSX soubory, tedy umět jednotlivé výsledky převést do struktury, ze které bude čtení dat pro programátora jednodušší a rychle připravené pro další zpracování.

(F1.3) Modul nabídne rozhraní pro uložení dat do relační databáze. Jelikož cílem modulu je převést data do formátu, který bude možné uložit do databáze dle jednotlivých logických struktur, je časově i uživatelsky pohodlné řešení provádět ukládání do databáze již během zpracování, nikoli později. Stejně tak vzhledem ke konečnému stavu dat je možné ihned při parsování vytvořit několik předpočítaných tabulek a hodnot pro rychlejší práci budoucí aplikace.

(F1.4) Modul bude obsahovat parser, který umožní zpracování dat. Jelikož data obsahují mnoho nekonzistencí a formátových chyb, které byly uvedeny v minulé kapitole, je nutné tyto data zpracovat a roztřídit tak, aby se pro budoucí použití dala bezpečně uložit do databáze v logických celcích.

2.1.2 Funkční požadavky na parser

(F2.1) Parser bude řešit problémy z kapitoly 1. Parser by měl dle požadavků na výsledná data vyřešit chyby, které jsou uvedeny v minulé kapitole, obzvlášť ty nejzávažnější, které by mohly znehodnotit výsledná data.

(F2.2) Parser bude umět zpracovat data z let 2014–2016. Jelikož se formát datových struktur mění během jednotlivých období a metodika definující struktury pro nové období od roku 2017 stále není hotová, budou zpracovávany jen nejnovější tři roky, jelikož mají podobnou strukturu dat.

(F2.3) Parser bude data třídit do struktur. Tedy bude využívat navržená schémata jednotlivých objektů pro lepší rozšiřitelnost (například o export zpracovaných dat do CSV souboru) a možnost přímé spolupráce s relační databází.

2.1.3 Požadavky na strukturu výsledných dat

(N1.1) Data budou roztříděna do logických celků. Z důvodu ukládání dat do relační databáze je nutná normalizace těchto dat, tedy přeorganizování struktury dat tak, aby využívala výhody relační databáze. Zatímco v původních souborech jsou data vložená do jedné velké tabulky s výsledky hodnocení, výsledná data vycházející z parseru by měla být utříděná dle návrhu databázové struktury, aby bylo jednoduše a rychle možné zajistit získání dat dle zadaných kritérií, například výskyt výsledku dle zadaného autora, hodnocení výskytu dle zadané organizace a další.

(N1.2) Data by měla mít stejný výstupní formát. Ačkoliv jednotlivé roky mají ve struktuře dat pro některé sloupce různý formát, po zpracování by měla mít data v těchto sloupcích formát stejný či podobný (alespoň v případě, kde to udělat lze), aby je bylo možné lépe srovnávat.

2.2 Implementace

2.2.1 Volba technologie

Pro tvorbu modulu se zpracováním dat si autor zvolil jazyk Python (verze 3) v interaktivním prostředí nástroje Jupyter a rozšířením za pomoci knihovny Pandas. Tato kombinace tvoří snadno ovladatelný a moderní nástroj pro analýzu velkého množství dat. Samotný jazyk Python je velmi oblíbený pro svou jednoduchost, čitelnost, rychlost a univerzálnost, díky které Python pohání všemožné aplikace i stroje. Dalším důvodem pro zvolení Pythonu je snaha autora práce naučit se jazyk, jelikož s ním do této doby nezískal žádné zkušenosti.

Jupyter notebook

Jupyter je webová aplikace nahrazující vestavěnou konzoli jazyka Python. Mezi její hlavní přednosti patří ukládání jednotlivých příkazů, které jsou vypisovány do bloků a je tedy možné se k nim vracet, měnit jejich pořadí, případně jednotlivé bloky jednoduše umazat. Tyto bloky vracejí výstupy podobné Python konzoli s rozdílem, že se dají formátovat a některé knihovny (například právě Pandas) zde mají speciální styl zobrazování. [11]

Jupyter mimo jiné podporuje i další sady jazyků, nikoli pouze Python, například Julia[10] a R[11], (podle kterých společně s Pythonem vznikl název Jupyter) a dají se stáhnout jádra i pro další jazyky.

Externí moduly

Pandas je knihovna sloužící k analýze a práci s tabulkovými daty, tedy například daty z relačních databází či spreadsheetových editorů jakým je například Excel. Základní datový typ je `DataFrame`, který jednoduše simuluje tabulku a obsahuje několik užitečných metod pro převod dat například právě z tabulkových formátů jakým je i XLSX. [12]

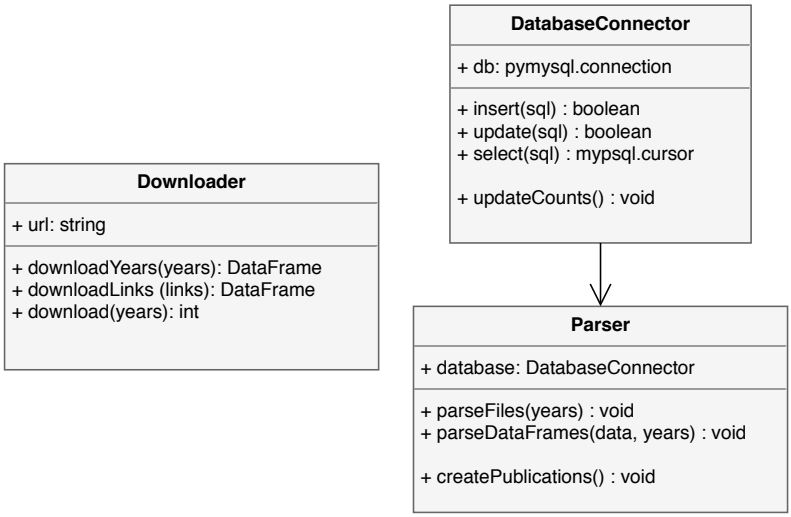
Beauftiful Soup je parser HTML/XML, který se zabývá získáváním dat z webových stránek. Hlavní předností je, že ignoruje špatnou syntaxi, tedy v případě, že obdrží špatně strukturované či formátované HTML, vrátí strom, který se snaží kopírovat zdroj a nevrátí chybu. Díky tomu se dají získat data i z chybně napsaných stránek bez nutnosti implementovat vlastní parser. [13] V této práci bude využit převážně pro získávání odkazů na jednotlivé hodnocení výsledků ze stránek VaVaI.

[10] Dynamický programovací jazyk pro vědecké výpočty. Více na `https://julialang.org/`.
[11] Programovací jazyk pro statistickou analýzu. Více na `https://www.r-project.org/`.

PyMySql je konektor pro práci s MySQL[12] databází. Nabízí základní rozhraní pro připojení k datbázi a operacemi nad ní. Kromě jednoduchých SQL příkazů umožňuje například i tvorbu transakcí. [14]

2.2.2 Stavba modulu

Python, stejně jako řada dalších programovacích jazyků, rozděluje jednotlivé třídy do modulů. Na rozdíl od jazyka C++, kde se dává přednost vytváření nového souboru pro každou třídu, je však zvykem třídy slučovat do souborů dle logického využití. [15] Zde je základním modulem soubor `revy.py`[13], který obsahuje všechny potřebné třídy pro zpracování dat. Velmi zjednodušený popis tříd popisuje diagram 2.1.



Obrázek 2.1: Zjednodušený diagram tříd v modulu revy

Stahování dat

Pro stažení dat slouží třída `Downloader`. Ta nabízí metody pro stažení dle zadaných let buďto přímo do objektů `DataFrame` nebo pouze pro uložení na disk. Při stahování jednotlivých let využívá knihovnu BeautifulSoup, díky které získává odkazy na jednotlivé soubory XLSX.

Parsování dat

Pro parsování dat slouží třída Parser. Ta data třídí do logických celků pro organizace, autory, vědecké výsledky (publikace) a hodnocení. Dále registruje

[12] Systém pro řízení relační databáze dostupný z `https://www.mysql.com/`.
[13] Zkratka pro „Reporting výsledků hodnocení“ užitá i pro název výsledné aplikace.

pomocné struktury, přispívající normalizaci dat a přípravě na vložení do databázové vrstvy, jako například: skupiny oborů, obory, jazyky a vztahy mezi jednotlivými objekty (v databázi vztahy M:N). Následně data odesílá ke vložení do databáze.

Přístup k celému zpracování parseru odráží snahu vyhnout se databázovým operacím z důvodu efektivity. Místo hledání již existujících výskytů v databázi si samotný Parser uchovává informaci o tom, jaké a kolik záznamů již do databáze vložil. Nicméně vzhledem k požadavku na rozlišení autorů a vědeckých výsledků se kterými jsou spojeny nekonzistence a složité prohledávání souborů, je výsledné zpracování výpočetně velmi náročné.

Ukládání dat

Funkce modulu PyMySQL obsahují vše nutné pro základní práci s databází[14], nicméně neumožňují ignorování chyb, které v použitých SQL příkazech vznikají. Navíc logika řízení vkládání záznamů do databáze se nehodí do struktury třídy `Parser`. Proto jsou zpracovaná data předána třídě `DatabaseConnector`, která sbírá informace o provedených příkazech a umožňuje nastavit toleranci chyb. To přispívá k budoucí využitelnosti skriptu právě v aplikaci Jupyter, jelikož při zahrnování nových roků s odlišnou strukturou bude možno jednoduše s výpisem chyb sledovat v čem a kdy chyba nastala a zpracování dat (které zřejmě bude trvat i několik desítek minut) nebude dále zastaveno, pokud se jedná o nezávažnou chybu.

V třídě se také nachází definice jednotlivých tabulek a jejich SQL příkazů.[15] Při vytváření nového databázového spojení je automaticky otestováno, zdali databáze vlastní veškeré tabulky, které jsou potřeba pro uložení dat.

2.2.3 Rozlišování autorů

Základním požadavkem na aplikaci a zpracovaná data je identifikace a sjednocení autorů. K tomu se váže problém s *vedidk*. Ten byl vyřešen v parseru rozdělením autorů do dvou skupin: autoři s *vedidk* a autoři bez *vedidk*. Jelikož pro obě skupiny bylo nutné vytvořit společný jedinečný identifikátor, který je bude v databázi sjednocovat pod jednu tabulku, byla využita vestavěná funkce jazyka Python `hash()`[16].

Tato hešovací funkce nabízí oproti jiným funkcím návratovou hodnotu v podobě celého čísla, což je pro účely databáze vhodnější volba než textový typ, jelikož porovnání čísel je v zásadě mnohem rychlejší a umožňuje tedy i rychlejší vyhledávání.

[14] Provádění SQL příkazů, prohledávání databáze, změny nastavení a další.
[15] Jelikož základní verze PyMySQL neumožňuje mapování tabulek do tříd.
[16] Dokumentace dostupná z `https://docs.python.org/3/library/functions.html`.

Skupina autorů s *vedidk* do této funkce předává pouze svůj identifikátor. Skupina bez identifikátoru je zadávána pomocí svého jména, příjmení, skupině oborů a organizaci.

Ukázkový příklad:

> *Ve výskytu výsledku v hodnocení pro České vysoké učení technické (IČO 68407700) s číslem skupiny oborů 3, jsou uvedeni dva autoři: „Josef Novák, 152637; Jan Švarc“. Tento vstup je oddělen pomocí středníku na informace o obou autorech.*
> *První autor, Josef Novák, je uveden s vedidk. Jeho identifikátor pro databázi je vytvořen funkcí* `hash(152637)`.
> *Druhý autor, Jan Švarc, je uveden bez vedidk. Jeho identifikátor pro databázi je vytvořen funkcí* `hash(set(Jan Švarc, 68407700, 3))`

Pro výsledek je důležitější rozlišení odlišných autorů, než nerozeznání autora stejného. Zahrnutí skupin oborů bude ve větším množství případů vést spíše k rozdělení jednoho a toho samého autora, ale zároveň tím může parser přispět k větší šanci na rozeznání dvou jiných autorů se stejným jménem.

2.2.4 Rozlišování vědeckých výsledků

Nejen autoři, ale také vědecké výsledky, tvoří při zpracování dat problém. Pro porovnávání činnosti organizací v rámci hodnocení jednoho roku stačí zpracovávat výskyty v daném roce. Při porovnávání mezi roky je ovšem nutné výskyty vědeckých výsledků spojovat do skupin, které budou reprezentovat vědecký výsledek napříč všemi organizacemi, aby nebyly stejné výskyty započítány vícekrát.

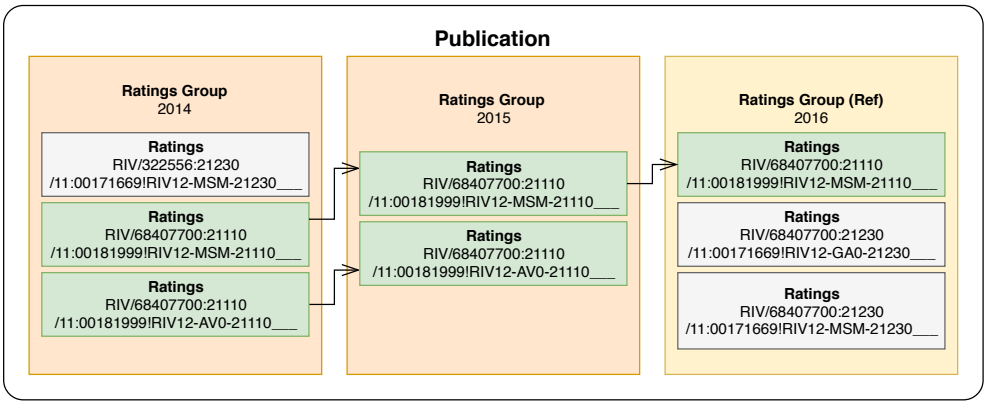
Toho nelze jednoduše dosáhnout, jelikož RIV definuje výsledek pro každou organizaci. Lze tedy nalézt v RIV dva různé záznamy pro stejný výsledek v případě, že jej předaly dvě různé organizace. *Například „The guarding game is E-complete“ má dva záznamy pro Univerzitu Karlovu*[17] *a pro ČVUT v Praze*[18]. Definujme proto pro následující kapitoly tyto struktury:

Hodnocení (Ratings) je výskyt výsledku v daném roce, tedy jakýkoliv řádek v souboru hodnocení, který vlastní unikátní identifikační kód výsledku s označením dodávky dat dle RIV.

Skupina hodnocení (Ratings Group) je skupina výskytů výsledků v daném roce. Tedy takové výskyty, které mají společný identifikační kód sjednoceného výsledku v rámci jednoho roku.

[17] `https://www.rvvi.cz/riv?s=jednoduche-vyhledavani&ss=detail&n=0&h=RIV%2F68407700%3A21240%2F14%3A00214334`

[18] `https://www.rvvi.cz/riv?s=rozsirene-vyhledavani&ss=detail&n=0&h=RIV%2F00216208%3A11320%2F14%3A10286913`



Obrázek 2.2: Příklad pro demonstraci struktur vědeckých výsledků

Publikace (Publication) je spojení skupin hodnocení přes stejné výskyty výsledků. Tato struktura pak definuje hodnocení výsledku v pozorovaných letech a jedná se o abstrakci vědeckého výsledku.

Referenční skupina hodnocení je skupina v rámci jedné publikace, která je obsažena v nejnovějším hodnocení. Slouží jako zdroj informací o publikaci.

Ukázkový příklad:

> *Vědecký výsledek „Lineární útvary v prostoru“ má ve výsledku hodnocení z roku 2014 výskyty označené identifikátory:* `RIV/A:B!X`, `RIV/A:B!Y` *a mají přiděleno sjednocené id* `K`.
>
> *Tyto výskyty se opakují v roce 2015 s novým výskytem* `RIV/A:C!X` *a mají sjednocené id* `L`. *V roce 2016 již nemá výsledek žádný výskyt.*
>
> *Výsledek tedy má v kontextu výše uvedených struktur 5 hodnocení (dvě v roce 2014 a tři v roce 2015) a dvě skupiny hodnocení* `K` *a* `L`. *Tyto dvě skupiny tvoří publikaci výsledku, jelikož dvě jejich hodnocení mají stejný identifikátor. Skupina* `L` *je označena jako referenční skupina, jelikož je z nejnovějšího hodnocení.*

S takto definovanými strukturami je možné rozlišit jednotlivé výskyty, seskupit je pod jeden celek a označit za vědecký výsledek jak je ukázáno na obrázku 2.2. Problémové mohou být případy kdy je z nějakého důvodu odeslán stejný výsledek k hodnocení podruhé. Přesný důvod tohoto počínání závisí na konkrétním případu, nicméně nově vzniklé výskyty, například s roční pauzou v roce 2015, by mohly způsobovat nesprávné spojení skupin hodnocení a tedy znehodnocení spojení skupin hodnocení pod publikaci. V některých případech se skupiny hodnocení překrývají (mají stejné sjednocené id v různých letech). Z toho důvodu jsou identifikátorům přidány prefixy roku, ze kterého skupina i samotný výskyt pocházejí.

Ve třídě `Parser` jsou pro tento účel využity vestavěné slovníky jazyka Python (`dict`) `occurre_group_trees`, `publications` a `occurre_groups_rated`. Parser nejdříve projde všechna získaná data a zjistí, které skupiny hodnocení jsou referenční a jaké mají id publikace. Druhý zmíněný slovník slouží pro ukládání bodů skupiny hodnocení, jelikož je uvedeno až v posledním výskytu. Pro efektivnost vkládání do MySQL databáze jsou využity transakce.

S pomocí těchto dat získaných v prvním průchodu je pak při vkládání skupin hodnocení předem dané, jaké mají mít id publikace a zdali jsou referenční. Tento způsob je mnohem rychlejší než procházet data pouze jednou a zpětně volat `UPDATE` příkazy nad skupinami hodnocení po uložení jejich id publikace.

2.2.5 Výpočet vlastních hodnot

Jelikož jsou data získaná z hodnocení statická a neprobíhá jejich pozdější úprava, dají se vypočítat některé hodnoty, které by pro relační databázi znamenaly velkou výpočetní náročnost za běhu aplikace. Vypočteny jsou následující údaje:

1. Počty bodů autorů a referenčních skupin hodnocení pro roky.
2. Počty bodů získaných v oborech a oborových skupinách, které jsou získávány na základě sečtení počtu bodů referenčních skupin.
3. Počty publikací v oboru a skupinách oborů, tedy počty referenčních skupin registrovaných pro daný obor.
4. Počty autorů v oboru a skupinách oborů.
5. Počty vědeckých výsledků organizace a jejích jednotek (tj. například fakult) vypočteny přes referenční skupiny.
6. Počty bodů organizace a jejích jednote, které jsou získány sečtením všech upravených bodů u hodnocení registrovaných k dané organizaci pouze u referenčních skupin.
7. Počty autorů organizace a jejích jednotek.
8. Počet autorů skupiny hodnocení.
9. Počet bodů hodnocení na jednoho autora, který je získán vydělením počtů upravených bodů organizace u hodnocení počtem domácích autorů.
10. Počet vědeckých výsledků autora, který je vypočten na základě skupin ke kterým je autor registrován.
11. Počet bodů autora, který je vypočten součtem všech bodů na autora v hodnocení (bod číslo 9), u kterých byl autor uveden.

Při vypočítávání bodu 1 byly vypočteny i statistiky pro všechny roky hodnocení. Ty byly označeny jako rok 0. Tato data jsou využita hlavně k vypočítání celorepublikových průměrů pro všechny roky (tj. například průměrný počet bodů referenční skupiny hodnocení)

U bodu číslo 10 se bere v potaz vždy jen jedna skupina vědeckého výsledku (publikace). Tento vztah není vázán na referenční skupiny hodnocení, jelikož u nich nemusí být uvedeni všichni autoři, kteří byli například v roce minulém. To by způsobilo, že autorům neuvedeným v referenční skupině by nebyl započítán výsledek.

V bodu 11 je přepočítáván zisk autora nikoli z celkových bodů výsledku, ale z upravených bodů organizace, která má na daném výsledku podíl. Je tak učiněno na základě nejistého počtu autorů skupiny hodnocení. Bez něj totiž není možné získat body autora pomocí vydělení získaných bodů skupiny hodnocení počtem jejích autorů. Přepočítávány jsou body opět pouze přes referenční skupiny hodnocení.

Ukázkový příklad:

> *Josef Novák je jediným autorem tří vědeckých výsledků, které napsal pro ČVUT v Praze. Každý má jeden výskyt v hodnocení 2014, který je opsán do roku 2015 i 2016 s korekturami. Existuje tedy 9 hodnocení, 9 skupin hodnocení a 3 referenční skupiny (výskyty z roku 2016).*
>
> *V roce 2014 mají výskyty bodové ohodnocení pro organizaci 10, 20 a 30 upravených bodů. Stejně je to s body i v roce 2015. V roce 2016 jsou upravené body organizace u výsledků 12, 22, 30. U každého hodnocení je vypočten počet bodů na autora v hodnocení, které budou stejné jako upravené body organizací, jelikož jediným autorem je Josef Novák.*
>
> *Počet bodů autora je tedy 64, jelikož referenční skupina je z roku 2016 a součet bodů hodnocení na autora je součet bodů 12, 22 a 30.*

Výpočet těchto hodnot umožňuje modul dvěma způsoby. První je volán automaticky při parsování dat a jedná se o metodu `createCounts()` třídy `Parser`. Ta využije své hodnoty, které byly pravidelně vypočítávány při zpracovávání dat a vkládání do databáze, a použije je pro upravení záznamů.

Druhý způsob je volání metody `udateCounts()` třídy `DatabaseConnector`, ten využije předepsané SQL skripty, které data dopočítají. Tento způsob je ale velice neefektivní[19].

[19] Při zpracování všech dat je upravováno několik milionů záznamů.

Kapitola 3

Analýza aplikace

3.1 Analýza současného stavu řešení

Aplikací nad daty z hodnocení výsledků vědeckých organizací nevzniká mnoho. Kromě oficiálního systému VaVaI vznikly takové, které se snaží o lepší zpracování vyhledávání nad daty, jejich lepší zobrazení, případně napojení na jiné webové služby. Mezi ně patří Trendy oborové publikační výkonnosti pracovišť výzkumných organizací v České republice v letech 2008–2014 [16] (dále pouze Trendy) od Institutu pro demokracii a ekonomickou analýzu (IDEA) a Vyhledávání ve výsledcích Hodnocení VaVaI [17] od RNDr. Pavla Šmerka, Ph.D. z Fakulty informatiky na Masarykově univerzitě.

3.1.1 Oficiální systém VaVaI

Informační systém VaVaI má jakožto jediný ze známých portálů plný přístup k veškerým datům předložených organizacemi. Systém je rozdělen do centrální evidence projektů[20] (CEP), centrální evidenci aktivit[21] (CEA), evidence veřejných soutěží[22] (VES) a rejstřík informací o výsledcích[23] (RIV), kde se nachází pro tuto práci důležité vědecké výsledky.

RIV poskytuje vyhledávání výsledků dle téměř všech kategorií, které jsou pro dané vědecké výsledky definovány při zadávání dat. Formuláře pro vyhledávání jsou dvojího typu: základní a rozšířené, což přispívá k větší přívětivosti uživatelského rozhraní, ale obsahují pro většinu kategorií pouze jednoduchou selekci s možností výběru jedné možnosti či fulltextové vyhledávání pro organizace či autory, kde by bylo vhodné uživateli umožnit zvolit více než pouze jednu položku. Celková doba vyhledávání se odlišuje podle zadaných para-

[20] `https://www.rvvi.cz/cep`
[21] `https://www.rvvi.cz/cea`
[22] `https://www.rvvi.cz/ves`
[23] `https://www.rvvi.cz/riv`

metrů, ovšem například vyhledání pouze podle jmen a příjmení autorů mívá v rámci odezvu delší než 20 sekund.

Hlavní nedostatkem celého systému spočívá ve zobrazování výsledků vyhledávání. Ty totiž RIV neumožňuje dále řadit ani filtrovat a není v nich tedy možné, například při vyhledání vědeckých výsledků pouze podle organizací nebo podle autorů, žádným logickým způsobem listovat. Při nutnosti snížit počet výsledků, či je jinak uspořádat, musí uživatel zadat či zvolit jiné filtry do původního formuláře, ke kterému se musí dostat pomocí tlačítka zpět.

Při kliknutí na libovolný výsledek vyhledávání se zobrazí detail zvoleného vědeckého výsledku a uživatel získá přehled o všech do systému zadaných informací. RIV tedy zpřístupňuje pouze získaná data, netřídí je na logické celky a nezpracovává žádná porovnání. Dokonce v detailech výsledků neukazuje ani výsledky hodnocení, ty jsou dostupné pouze v souborech XLSX přes speciálně upravené stránky se seznamem hodnocení tříděných dle jednotlivých organizací:

- rok 2014 dostupný z `http://hodnoceni14.rvvi.cz/www/`,
- rok 2015 dostupný z `http://hodnoceni15.rvvi.cz/www/`,
- rok 2016 dostupný z `http://hodnoceni16.rvvi.cz/www/`.

Na těchto stránkách se autorovi této práce nepodařilo nalézt žádnou domovskou stránku, odkud by bylo možné se dostat k jednotlivým ročním hodnocením, tedy odkazy na tyto stránky lze získat pouze přes novinky v archivu IS VaVaI. Hodnocení lze získat pro jednotlivé organizace stažením XLSX souborů, třídění je umožněné pouze zadáním názvu organizace. Navíc je celkové hodnocení (tedy počet bodů a počet upravených bodů) možné vidět jen u posledního do RIV dodaného záznamu o výskytu výsledku, tedy v případě, že se výsledek zařazuje mezi několik organizací, které na něm mají podíl, musí uživatel projít všechny výskyty vědeckého výsledku u všech organizací, pro které je daný výsledek zařazen nebo si body dle podílu organizace dopočítat. Jak však bylo zjištěno v analýze dat v předchozí části této práce, není přesný výsledek při takovém dopočítání zcela jistý.

3.1.2 Aplikace Trendy od IDEA

Webová stránka Trendy nabízí oproti oficiálnímu systému odlišný pohled na data. Nejedná se o vyhledávač vědeckých výsledků, nýbrž o jednoduchou aplikaci zobrazující podíly publikačního výkonu zvoleného významu na celkovém publikačním výkonu oboru v rámci let 2008–2009 a 2013–2014 a to pouze pro články registrované na Web of Science (tedy součty publikací ve WoS pro organizace v poměru s celkovým počtem celé ČR).

Otázkou je, zdali aplikace vůbec zpracovává výsledky hodnocení z RIV nebo pouze vyhledává výsledky uvedené ve WoS. Nicméně jedná se o jednu z mála aplikaci snažící se o porovnání vědecké činnosti organizací, ač nemá přístup k finančním prostředkům ani velikosti výzkumných týmů. [18]

Uživatelské rozhraní aplikace Trendy se skládá z jednoduchého vyhledávacího formuláře se základními filtry pro získání výsledku pro zadané organizace, její jednotky a obor. Pro získání výsledku je nutné mít vybranou ze všech těchto kategorií alespoň jednu položku, případně označit možnost: „Vybrat všechny“ u každé z nich. Výsledky pak ovlivňuje zvolené třídění oborů a použité jednotky pro podíly. Všechny části formuláře jsou doplněny o užitečné nápovědy, avšak samotné uživatelské rozhraní doprovázejí, vzhledem k jednoduchosti formuláře, opravdu dlouhé instrukce, jak celé prostředí používat.

V poslední řadě rozhraní nabízí možnost přepnout výsledky do grafu, tím se ukáže jeden velmi jednoduchý sloupcový graf ukazující pouze poměr podílů na oboru z let 2008–2009 a 2013–2014 zvolených organizací. Tyto grafy neukazují žádné informace, které by nebylo možné vyčíst ze samotného výsledku vyhledávání, kde je dokonce možné vidět i vyjádřený poměr těchto dvou podílů na oboru. Navíc grafy při zvolení více organizací nezobrazují vše, pouze pár prvních organizací, které se ve výsledcích vyhledávání ukazují.

3.1.3 Vyhledávání ve výsledcích Hodnocení VaVaI

Jednoduchá aplikace od RNDr. Pavla Šmerka, Ph.D. se funkcionalitami podobá oficiálnímu IS VaVaI. Rozšiřuje detaily výskytů vědeckých výsledků a vyhledávání v RIV o hodnoty z výsledků hodnocení a umožňuje zadat větší množství kategorií (například pro organizace) do vyhledávání. Kombinuje tedy jak data získaná z RIV, tak data ze stránek s hodnoceními. Data z RIV poskytuje ze stažených statických stránek detailů výskytů výsledků přímo ze systému pomocí url odkazu v datech výsledků hodnocení[24] (v případě roku 2014 musí poskytovat vlastní zálohu, jelikož systém odkázaný v roce 2014 již není dostupný jak bylo zmíněno v části 1.1.1.

Data jsou v aplikaci tříděna dle hodnocení, tedy stejně jako v oficiálním systému, což znemožňuje vyhledávat mezi hodnocenými výsledky v rámci let. V případě nalezené shody výskytů jsou u dvou různých hodnoceních propojeny odkazem na předchozí či následující rok, kde byla shoda nalezena.

Velkou předností oproti systému RIV je samotné vyhledávání s využitím velkého formuláře, který umožňuje hledat prakticky téměř podle čehokoliv, co se dá z oficiálního systému získat. Vyhledávání dokonce poskytuje možnost pro uživatele definovat vlastní dotaz na vyhledávání a vlastní definici řazení, které je ale dostupné opět pouze v původním formuláři.

Aplikace také pracuje se srovnáním výskytů výsledků v hodnoceních v rámci let. Při zobrazení detailu výskytu výsledku můžeme nalézt porovnání s před-

[24] Informace získaná přímo od RNDr. Pavla Šmerka, Ph.D.

cházejícími roky, které v systému jsou. Aktuálně se zřejmě jedná o jedinou takto fungující aplikaci pro vyhledání výsledků hodnocení a informací o vědeckých výsledcích. Opět ovšem vzhledem k závislosti na původních datech z RIV neposkytuje aplikace žádné roztřídění výsledků na jednotlivé organizace či autory. Tento projekt je stále ve vývoji a je možné, že autor do aplikace přidá i další funkcionality a možná později vylepší i aktuálně velmi jednoduché uživatelské rozhraní.

3.2 Funkční požadavky

3.2.1 Vyhledávání výsledků z hodnocení

Základním problémem výsledků hodnocení v oficiálním systému je nemožnost vyhledat jednotlivé vědecké výsledky a získat tak okamžitý pohled na získaná bodová ohodnocení. Z toho důvodu by aplikace měla obsahovat vyhledávání výskytů výsledků v hodnocení na základě všech informacích, které je možné z výsledků hodnocení získat. Kromě podrobnosti zadávacích kritérií může být vyžadována jejich kombinovatelnost, tedy například zadat několik organizací, pro které se má výsledek zobrazit.

Organizace dat do jednotlivých struktur pak umožní vyhledávat nejen výskyty výsledků, ale rovněž autory, obory či organizace ke kterým jsou v hodnoceních výskyty přiděleny. Vzhledem k jejich vysokému počtu se vyplatí implementovat vyhledávání i přímo pro tyto struktury s podobnými požadavky, které byly výše vytyčeny pro výskyty výsledků v hodnocení.

3.2.2 Zobrazení detailů činnosti

Jelikož hodnocení obsahují poměrně vysoký počet informací o vědeckých výsledcích, je možné získat i základní informace o jednotlivých autorech a organizacích jako například jejich specializaci v oborech. Na základě snahy vytvořit převážně reportingovou aplikaci, je vhodné tato data zpracovávat a umožnit tak uživateli zobrazení náhledu s detaily jednotlivých objektů. To poslouží nejen k lepší identifikaci daného objektu, ale také přehledu o jeho činnosti. Vzhledem k získávání dat z několika let je ideální data numerického typu zobrazovat v grafech či pomoci jiných grafických komponent, jak tomu bývá právě u reportingových aplikací.

3.2.3 Filtrování výsledků

Nedostatkem všech zmiňovaných systémů bylo třídění získaných výsledků. Z toho důvodu by aplikace měla umožňovat získané výsledky vyhledávání řadit a přehledy o činnosti objektů filtrovat podle nejdůležitějších kategorií daného vyhledávání či přehledu. Díky tomu bude možno jednoduše a rychle srovnávat jednotlivé výsledky hodnocení nebo aktivity autorů a organizací.

3.3 Případy užití

Jelikož se jedná o systém nad otevřenými daty přístupný každému, platí pro aplikaci velice jednoduchá pravidla přístupu. Typ aplikace a přístup k datům nevyžaduje přihlášení, tedy z pohledu případů užití existuje pouze jediný aktér – uživatel aplikace.

UC1 Vyhledání výsledků v hodnoceních

Základní případ užití, ve kterém se uživatel snaží získat informace o hodnocení, vědeckém výsledku, jeho autorech a organizacích, které na něm mají podíl.

Scénář případu užití

1. Systém zobrazí úvodní stránku.
2. Uživatel zvolí možnost „vědecké výsledky“.
3. Systém zobrazí formulář vyhledávání ve výskytech výsledků.
4. Uživatel zadá kritéria a odešle formulář.
5. Systém zobrazí seznam výsledků relevantních pro zadaná kritéria.

Alternativní scénář

4.1. Pokud uživatel zadal neplatný vstup, systém na skutečnost uživatele upozorní a nedovolí formulář odeslat.

4.2. Uživatel opraví neplatný vstup/vstupy a vrací se k bodu 4.

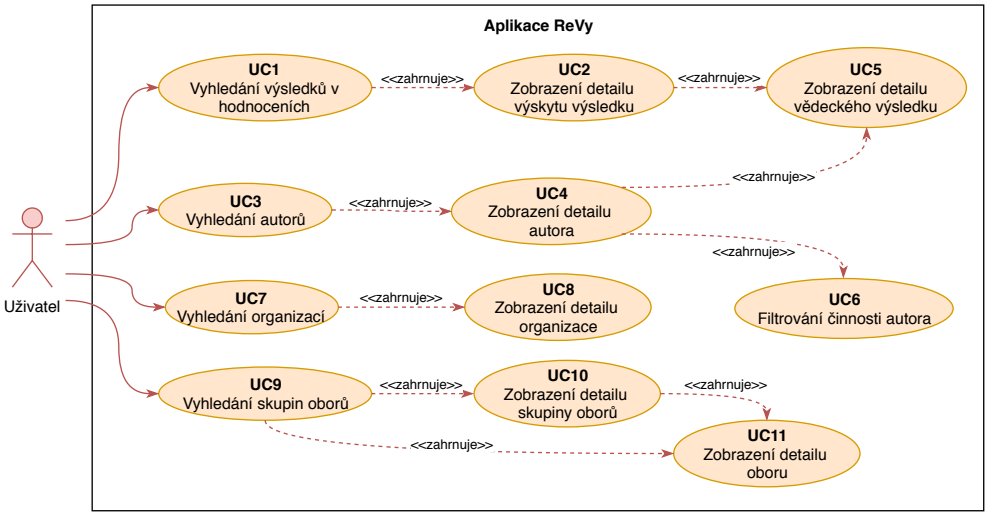
UC2 Zobrazení detailu výskytu výsledku

V tomto případu užití se uživatel snaží zobrazit detail výskytu výsledku s jeho informacemi a odkazy na detail v RIV či na jiné s výsledkem související objekty. (autoři, organizace)

Zahrnuje případ: UC1 Vyhledání výsledku v hodnocení

Scénář případu užití

1. UC1 Vyhledání výsledku v hodnocení.
2. Uživatel vybere daný výskyt výsledku a klikne na jeho název.
3. Systém zobrazí detail výskytu výsledku.



Obrázek 3.1: Diagram případů užití v aplikaci ReVy

UC3 Vyhledání autorů

Uživatel se snaží získat přehled o některém z autorů, který byl ve výsledcích hodnocení uveden.

Scénář případu užití

1. Systém zobrazí úvodní stránku.
2. Uživatel zvolí možnost „autoři“.
3. Systém zobrazí formulář vyhledávání v seznamu autorů.
4. Uživatel zadá kritéria a odešle formulář.
5. Systém zobrazí seznam autorů relevantních pro zadaná kritéria. Přednost dá těm, kteří mají *vedidk* (Tedy jsou jasně identifikování v rámci hodnocení).

Alternativní scénář

4.1. Pokud uživatel zadal neplatný vstup, systém na skutečnost uživatele upozorní a nedovolí formulář odeslat.

4.2. Uživatel opraví neplatný vstup/vstupy a vrací se k bodu 4.

UC4 Zobrazení detailu autora

V tomto případu užití se uživatel snaží získat detailní informace o autorovi a jeho činnosti

Zahrnuje případ: UC3 Vyhledání autorů

Scénář případu užití

1. UC3 Vyhledání autora.
2. Uživatel vybere daného autora a klikne na jeho jméno.
3. Systém zobrazí detail autora s grafy zobrazující jeho činnost, která je získatelná z výsledků hodnocení.

UC5 Zobrazení detailu vědeckého výsledku

V tomto případu užití se uživatel snaží získat přehled o vědeckém výsledku, ke kterému patří dané výskyty v hodnocení. V aplikaci se jedná o stejný případ užití jako zobrazení jako zobrazení detailu výskytu jen s rozdílem, že se jedná o výskyt referenční.

Zahrnuje případ: UC4 Zobrazení detailu autora nebo
UC2 Zobrazení detailu výskytu výsledku

Scénář případu užití

1. UC4 Zobrazení detailu autora.
2. Uživatel vybere daný vědecký výsledek autora.
3. Systém zobrazí detail vědeckého výsledku s odkazy na další objekty s ním související.

Alternativní scénář

1. UC2 Zobrazení detailu o výskytu výsledku.
2. Uživatel klikne na odkaz na referenční výskyt výsledku.
3. Systém zobrazí detail vědeckého výsledku s odkazy na další objekty s ním související.

UC6 Filtrování činnosti autora

V tomto případu užití se uživatel snaží porovnat autorovu tvorbu v rámci let či organizace a jejích jednotek. (Pokud je to pro daného autora relevantní)

Zahrnuje případ: UC4 Zobrazení detailu autora

Scénář případu užití

1. UC4 Zobrazení detailu autora.
2. Uživatel vybere danou kategorii ve formuláři u činnosti autora.
3. Systém zobrazí činnost autora relevantní k zadaným filtrům.

UC7 Vyhledání organizací

Uživatel se snaží získat přehled o některé z ogranizací a jeho organizačních jednotek.

Scénář případu užití

1. Systém zobrazí úvodní stránku.
2. Uživatel zvolí možnost „organizace“.
3. Systém zobrazí formulář vyhledávání v seznamu organizací.
4. Uživatel zadá kritéria a odešle formulář.
5. Systém zobrazí seznam organizací a jejích jednotek, pokud nějaké má.

Alternativní scénář

4.1. Pokud uživatel zadal neplatný vstup, systém na skutečnost uživatele upozorní a nedovolí formulář odeslat.

4.2. Uživatel opraví neplatný vstup/vstupy a vrací se k bodu 4.

UC8 Zobrazení detailu organizace

V tomto případu užití se uživatel snaží získat detailní informace o organizaci

Zahrnuje případ: UC7 Vyhledání organizací

Scénář případu užití

1. UC5 Vyhledání organizací.
2. Uživatel vybere danou organizaci a klikne na její název.
3. Systém zobrazí detail organizace seznamem organizačních jednotek a grafy pro dané organizační jednotky, pokud organizace nějaké jednotky vlastní.

UC9 Vyhledání skupiny oborů

Uživatel se snaží získat přehled o skupině oborů, která byla použita ve výsledcích hodnocení.

Scénář případu užití

1. Systém zobrazí úvodní stránku.
2. Uživatel zvolí možnost „obory“.
3. Systém zobrazí formulář vyhledávání v seznamu skupin oborů.
4. Uživatel zadá kritéria a odešle formulář.
5. Systém zobrazí seznam skupin oborů relevantních pro zadaná kritéria.

Alternativní scénář

4.1. Pokud uživatel zadal neplatný vstup, systém na skutečnost uživatele upozorní a nedovolí formulář odeslat.

4.2. Uživatel opraví neplatný vstup/vstupy a vrací se k bodu 4.

UC10 Zobrazení detailu skupiny oborů

V tomto případu užití se uživatel snaží získat detailní informace o skupině oborů.

Zahrnuje případ: UC9 Vyhledání skupin oborů

Scénář případu užití

1. UC9 Vyhledání skupin oborů.
2. Uživatel vybere danou skupinu oborů a klikne na její název.
3. Systém zobrazí detail skupiny oborů se seznamem podoborů a grafy s rozdělením dle podoborů.

UC11 Zobrazení detailu oborů

V tomto případu užití se uživatel snaží získat informace o oboru, který byl použit v hodnocení z některého roku.

Zahrnuje případ: UC9 Vyhledání skupin oborů nebo
UC10 Zobrazení detailu skupiny oborů

Scénář případu užití

1. UC9 Vyhledání skupin oborů.
2. Uživatel vybere danou skupinu oborů a klikne na tlačítko s označením počtu podoborů.
3. Systém zobrazí seznam podoborů skupiny.
4. Uživatel vybere daný obor a klikne na jeho název.
5. Systém zobrazí informace o oboru.

Alternativní scénář

1. UC10 Zobrazení detailu skupiny oborů.
2. Uživatel vybere daný podobor uvedený v seznamu podoborů.
3. Systém zobrazí informace o oboru.

Kapitola 4

Návrh

4.1 Návrh databáze

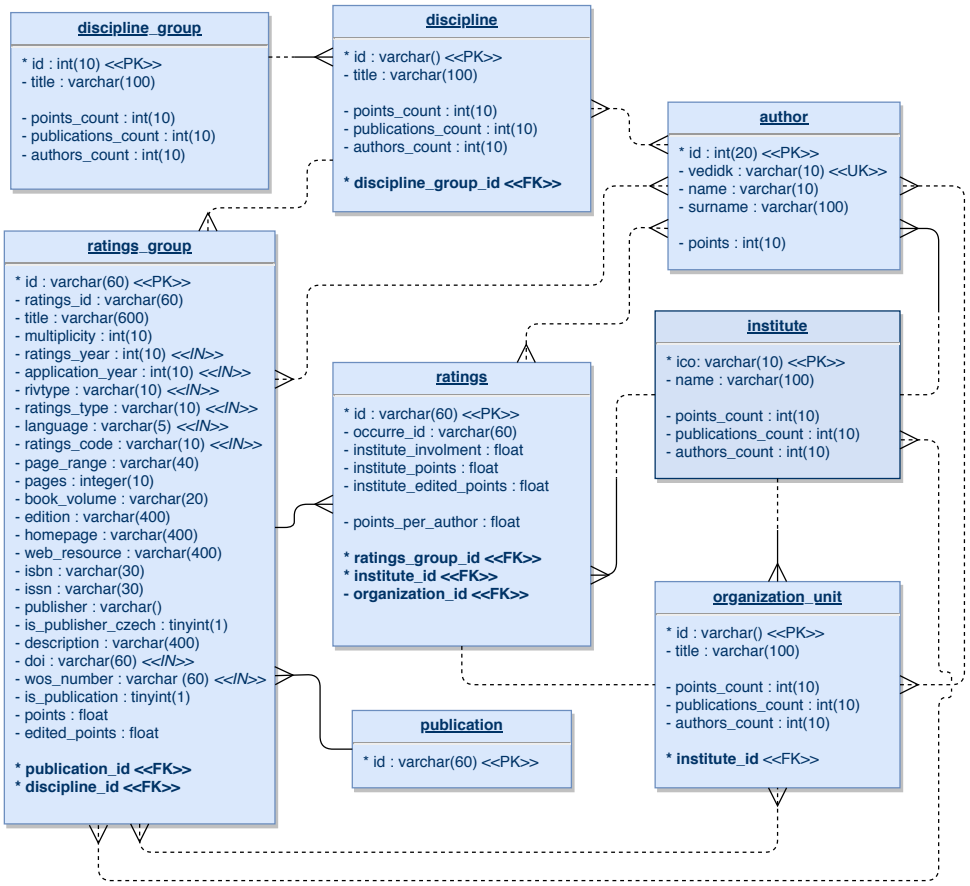
Návrh databáze vychází z analýzy a zpracování dat. Základem je snaha o co největší normalizaci získaných dat do tabulek, dle kterých bude možné následně provádět jejich vyhledávání a zobrazování jejich detailů.

Pro zefektivnění vyhledávní byly přidány následující pomocné tabulky pro objekty, které výrazně pomohou při výstavbě formulářových prvků (v závorce je uveden název tabulky):

- typ v RIV (`rivtype`),
- typ v hodnocení (`ratings_type`),
- roky hodnocení (`ratings_years`),
- roky uplatnění (`years`),
- kódy hodnocení (`ratings_code`),
- jazyky (`language`).

Organizace (tabulka `institute`) a pod ní spadající jednotky (tabulka `organization_unit`) jsou v návrhu odděleny. Každá jednotka totiž může sama o sobě nést stejné množství informací jako organizace samotná a většina výzkumných organizací v hodnocení žádné jednotky nevlastní.

Nositelem získaných informací o výsledku v hodnocení je skupina hodnocení (`ratings_group`) z toho důvodu, že se i některé informace (například seznam autorů) může v hodnoceních v odlišných letech měnit a je proto vhodné dané změny reflektovat. Ze všech výskytů vědeckého výsledku přebírá skupina hodnocení seznamy autorů, organizací a jednotek a utváří tak obraz vědeckého výsledku v rámci roku hodnocení. Tyto skupiny jsou pak sloučeny identifikátorem publikace (tabulka `publication`).



Obrázek 4.1: Zjednodušené databázové schéma bez dekompozice vztahů M:N

Spojení mezi skupinou hodnocení a organizací tvoří tabulka hodnocení (`ratings`). Ta nese informace o zapojení organizace a organizační jednotky a informace o oboru a domácích autorech organizace. Pro jednoduchý přehled tabulek a vztahů slouží zjednodušené schéma s nedekomponovanými vztahy na obrázku 4.1.

4.2 Uživatelské rozhraní

Cílem této práce je vytvořit webovou aplikaci pro prezentaci dat pro nespecifikované užívatele. Jelikož pro informační účely jsou využívány již jiné aplikace zmíněné v analýze současného řešení, či aplikace interní, které jistě poskytují větší rozsah informací (Například interní systém ČVUT – V3S [19]), rozhodl se autor této práce přiblížit uživatelské rozraní spíše moderním reportingovým systémům jako například Google Analytics [20] či Matomo [21] (Dříve známý jako Piwik).

Tento návrh se snaží o vytvoření pro uživatele jednoduchého a přehledného zobrazení propojených dat. Porušuje přitom některé zásady přístupnosti [22] a použitelnosti [23] s ohledem na grafické zaměření aplikace.

4.2.1 Základní struktura

Základním prvek na obrazovce je vždy menu s logem aplikace a drobečková navigace. Pro širší obrazovky se autor rozhodl využít umístění menu vlevo s vertikálním rozložením položek. Pro užší obrazovky a mobilní zařízení je pak menu přemístěno do horní části obrazovky s horizontálním rozložením, aby byla maximalizována obsahová část stránek. Drobečková navigace je umístěna vždy nejvýše v obsahové části stránky.

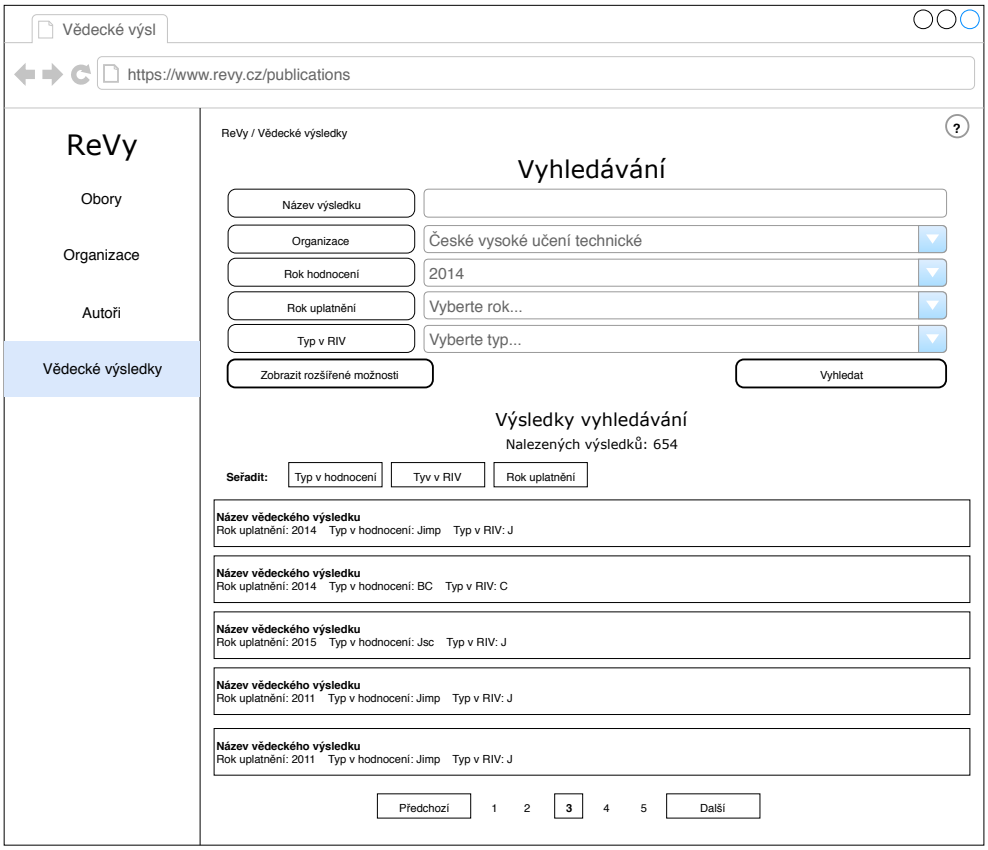
V případě, že stránka poskytuje rozhraní, které pro uživatele nemusí být intuitivně ovladatelné, obsahuje v pravém rohu malý kruhový odkaz, který zobrazí nápovědu. Ten se pohybuje společně s drobečkovou navigací.

Obsahová stránka celé aplikace se dá rozdělit na dvě základní rozhraní poskytované uživateli a to vyhledávání a detaily vyhledané struktury.

4.2.2 Vyhledávání

Rozhraní umožňující vyhledávání je rozděleno do dvou částí. První zobrazuje vyhledávací formulář, který je v případě obsahu mnoha prvků omezen na základní a nabízí možnost zobrazení pokročilých možností. V případě zadání hodnot, které nesouvisí se strukturou vyhledávaných parametrů (například zadání ISBN výsledku v nepředepsaném formátu) je formulář označen za nevalidní a zobrazí se chybová hláška v hlavičce formuláře společně s dodatečným komentářem u chybně zadaného parametru. Zřejmě nejsložitější formulář bude mít vyhledávání výskytů vědeckých výsledků, jehož návrh rozhraní je na obrázku 4.2.

Druhá část rozhraní vyhledávání je seznam samotných výsledků. Ten je poskytován ve stránkovací podobě s možností řadit výsledky dle základních kritérií, které jsou nadepsané v hlavičce vyhledávání. Hlavička dále obsahuje i počet nalezených výsledků pro zadaná kritéria.



Obrázek 4.2: Návrh rozhraní pro vyhledávání

Samotné výsledky v seznamu jsou reprezentovány svým názvem, který odkazuje na jejich detail, a výčtem základních atributů, pomocí kterých je výsledek možné identifikovat.

V případě, že se jedná o nerozhodný výsledek (například autoři bez *vedidk* u vyhledávání autorů), je celé okénko s výsledkem označeno odlišnou barvou než výsledky ostatní.

4.2.3 Detaily

Detaily jednotlivých objektů jsou děleny na bloky jako je tomu právě například u Google Analytics či Matomo. To umožňuje strukturovat obsah stránky na další objekty. Bloky se na na stránkách dělí do dvou horizontálně oddělených skupin.

První skupina zobrazuje všechny základní informace o objektu, které byly získány přímo ze souboru s hodnocením. Pro autory jsou to tedy například organizace, obory a vědecké výsledky, pro vědecké výsledky zase obory, autoři

a organizace. V případě, že se jedná o objekty, které v rámci aplikaci také obsahují detailní stránku, je jejich název odkazem na příslušný detail.

Druhá skupina bloků obsahuje grafy a grafické prvky zobrazující agregace nad získanými daty, pokud je relevantní pro daný objekt nějaké agregace sledovat. Většinu takto agregovaných hodnot je vzhledem k množství dat výpočetně a časově náročné získat. Proto jsou u vysoce abstraktních objektů s mnoha registrovanými daty (skupiny oborů, obory, organizace) poskytovány bloky zobrazující pouze předpočítané hodnoty uvedené v části 2.2.5. Příkladem může být detail autorů na obrázku 4.3.

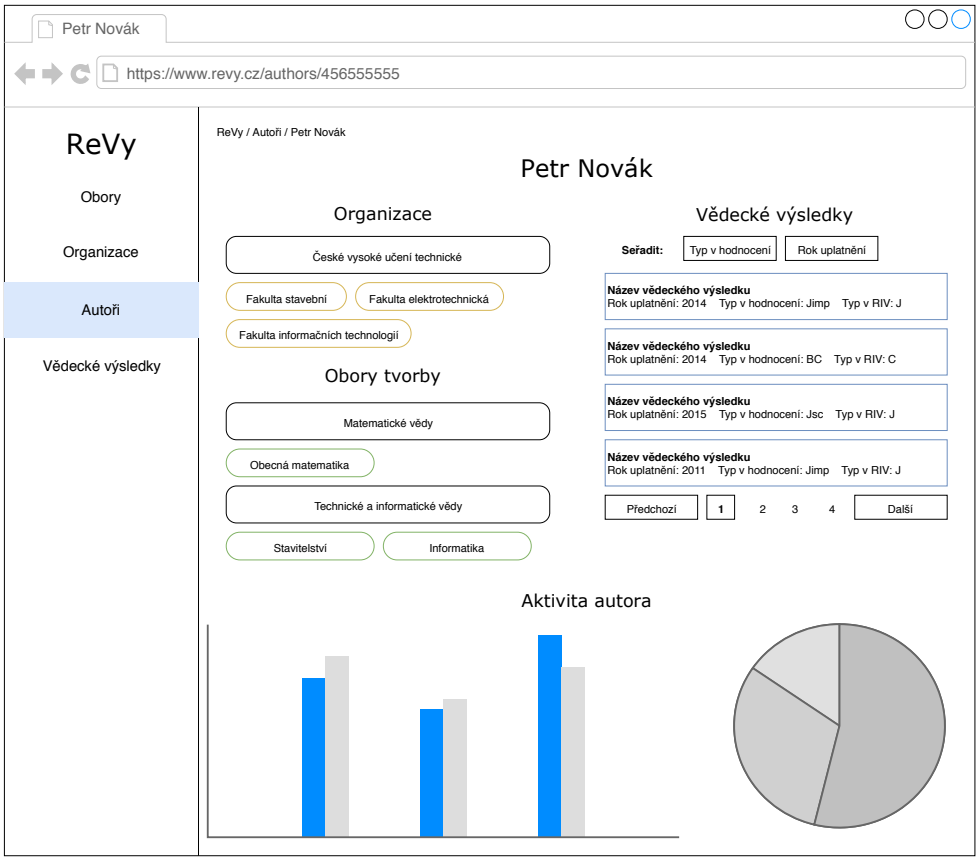
4.2.3.1 Bloky pro autory

Detaily autorů obsahují přehled jejich tvůrčí činnosti, kterou bude možné filtrovat. Poskytnutý je proto formulář nad filtrovanými bloky, pomocí kterého bude možné získat informace jen pro specifickou autorovu organizaci, její jednotku či obor. Ze získaných dat se rozhodl autor práce poskytnout následující bloky:

- Počet vědeckých výsledků autora v poměru s filtrovaným objektem. Pokud není zadán filtr, zobrazí se republikový průměr.
- Počet získaných bodů autora v poměru s filtrovaným objektem. Pokud není zadán filtr, zobrazí se republikový průměr.
- Počet bodů na publikaci v poměru s filtrovaným objektem. Pokud není zadán filtr, zobrazí se republikový průměr.
- Seznam publikací autora dle roku uplatnění výsledku tříděných dle typu výsledku ve sloupcovém grafu
- Počet získaných bodů dle typu výsledků v koláčovém grafu.
- Počet bodů získaných autorem v poměru s body, které získaly autorovy výsledky dle roku uplatnění výsledku v čárovém grafu.
- Pavučinový graf zobrazující počet výsledků autora v daném oboru. Pro přehlednost je vhodné vybrat jen několik nejvýznamnějších.

4.2.3.2 Bloky pro organizace a jednotky

V rámci organizací je zajímavé porovnání jednotlivých organizačních jednotek a to jak v počtu bodů, tak v počtu publikací. K rozmanitosti celého rozhraní je vhodné nabídnout dva grafy rozdílného typu, například sloupcový a koláčový graf. To vše lze samozřejmě jen v případě, že zadaná organizace nějaké organizační jednotky vlastní. Pro všechny organizace a případně i její jednotky je dále poskytnut žebříček nejlepších autorů dle počtu bodů i dle počtu výsledků a seznam nejlépe ohodnocených vědeckých výsledků podle bodů.



Obrázek 4.3: Návrh rozhraní pro detail autora

4.2.3.3 Bloky pro skupiny hodnocení

U skupin hodnocení je z pohledu uživatele zajímavé získat přehled o jiných skupinách hodnocení pro stejný vědecký výsledek (publikaci), pokud existují. Proto je vhodný blok se srovnáním pomocí tabulky jednotlivých skupin v letech hodnocení. Další podstatné informace jsou o hodnocení pro jednotlivé organizace. I pro tento účel bude vhodná tabulka například s přehledem všech organizací případně organizačních jednotek, které měly v hodnocení uveden nějaký podíl na celkovém výsledku. Obě tyto tabulky by zároveň měly sledovat nekonzistence mezi podíly a hodnoceními a případně daný údaj graficky zvýraznit (například jinou barvou), aby bylo uživteli jasné, že se v datech vyskytuje chyba.

4.3 REST API

Aplikace, vzhledem k možnosti poskytování nových struktur oproti výsledkům hodnocení jako například autorů či organizací, bude nabízet RESTful rozhraní. Pro účely jiných systémů či využití a napojení aplikace na jiné služby se mohou hodit základní metody pro získání dat příslušejících k výzkumným organizacím, autorům a jeho výsledkům ukázaných v hodnocení, či výsledkům z hodnocení samotných dle jednotlivých identifikátorů.

Pro tyto potřeby jsou nabízeny HTTP metody dostupné z následujících adres:

GET api/authors/{vedidk} vrátí autora se všemi informacemi z databáze (tj. i s autorem spojené organizace a obory) dle zadaného *vedidk*.

GET api/authors/{vedidk}/ratings vrátí všechny skupiny hodnocení, u kterých byl uveden autor dle zadaného *vedidk*.

GET api/institutes/{ico} vrátí organizaci se seznamem jejích organizačních jednotek dle zadaného IČO.

GET api/institutes/{ico}/authors vrátí seznam domácích autorů spojených v hodnocení s organizací dle zadaného IČO.

GET api/organization-unit/{id}/authors vrátí seznam domácích autorů spojených s organizační jednotkou dle zadaného identifikátoru organizační jednotky.

GET api/ratings/{id} vrátí skupinu hodnocení podle id v databázi. To je tvořeno ve formátu: *„rok hodnocení-identifikátor výsledku s označením dodávky dat do RIV*[25]*“*.

GET api/ratings Vrátí skupiny hodnocení ze všech let na základě alespoň jednoho z následujích url query parametrů:

wosNumber udává přístupové číslo do WoS.[26]

doi udává DOI výsledku.

riv udává identifikátor výsledku s označením dodávky dat do RIV.

[25] Identifikátor popsán v části 1.1.2

[26] `https://images.webofknowledge.com/images/help/WOS/hs_accession_number.html`

Kapitola 5

Implementace

5.1 Použité technologie

Pro implementaci aplikace autor zvolil jazyk PHP[27] z důvodu osobní preference v oblasti webového inženýrství. Pro usnadnění práce byl zvolen framework Symfony verze 4 [24], ač se pro takto jednoduchou a graficky zaměřenou aplikaci více hodí jednodušší nástroje jako například frameworky Nette[28], CakePHP[29], či například Slim[30]. Autor tak učinil z důvodu snahy rozvinout své znalosti v nejnovější verzi Symfony.

V Symfony existují stahovatelné komponenty nazývané „balíčky“, ty je možné do projektu jednoduše zakomponovat například pomocí nástroje Composer[31]. Vzhledem k povaze projektu bylo použito hned několik následujících knihoven a komponent.

Doctrine[32] je knihovna poskytující komunikaci s relační databází s využitím objektově relačního mapování (dále ORM). To umožňuje automatické mapování objektů na data z databáze a abstrakci nad databází, která může být nahrazena za jinou podporovanou relační databázi.

Less[33] je preprocessor CSS. Nabízí zjednodušený syntax, který je později možné snadno zkompilovat a získat tak plný soubor CSS použitelný přímo na webu.

Webpack[34] je balíčkovací systém, který umožňuje rozsáhlé soubory se závislostmi kompilovat do statických minimalizovaných zdrojů.

[27] `http://php.net/`
[28] Český framework dostupný z `https://nette.org/`
[29] Framework inspirovaný Ruby on Rails `https://cakephp.org/`
[30] Minimalistický PHP framework dostupný z `https://www.slimframework.com/`
[31] `https://getcomposer.org/`
[32] `https://www.doctrine-project.org/`
[33] `http://lesscss.org/`
[34] `https://webpack.js.org/`

JQuery[35] je zřejmě nejpopulárnější javascriptovou knihovnou. Zjednodušuje javascriptové metody a je jednoduchý na naučení [25].

Highcharts[36] je javascriptová knihovna pro vizualizaci a práci s interaktivními grafy. Obsahuje šablony pro velké množství grafů počínaje obyčejnými sloupcovými, čárovými či koláčovými až po složitější kombinované. Knihovna je pod ochranou Creative Commons (CC) Attribution–NonCommercial licencí, která zakazuje bezplatné využití knihovny pro komerenční účely [26]. V práci bude využita pro vizualizaci činnosti autorů a organizací. Ke svému používání vyžaduje knihovnu JQuery.

Chosen[37] je javascriptová a CSS knihovna nahrazující HTML select prvky. Podobných knihoven existuje mnoho (např. JQuery multiselect[38], Selectize[39] a další). Tato knihovna byla vybrána z důvodu, že umožňuje vyhledávání ve výběru možností a do svého stylování zahrnuje třídy definované v oddělených souborech CSS, které lze vlasními styly snadno přepsat a změnit tak snadno vzhled výsledných prvků.

ObHighchartsBundle[40] je balíček pro framework Symfony umožňující objektový návrh nad knihovou Highcharts. Pomocí jednoduchého rozhraní zpřístupňuje všechny metody Highcharts api a umožňuje tak objektový přístup ke grafům v jazyku PHP, místo psaní čistého javascriptu, který je později na základě vytvořeného objektu generován.

FOSRestBundle[41] je komponenta Symfony podporovaná autory frameworku. Jedná se o nástroj pro zjednodušení práce s RESTful[42] aplikačním rozhraním.

JMSSerializerBundle[43] je další balíček pro Symfony, který je využíván FOSRestBundle pro serializaci objektů a jejich následné odeslání.

[35] `http://jquery.com/`
[36] `https://www.highcharts.com/`
[37] `https://harvesthq.github.io/chosen/`
[38] `http://loudev.com/`
[39] `https://selectize.github.io/selectize.js/`
[40] `https://github.com/marcaube/ObHighchartsBundle`
[41] `https://github.com/FriendsOfSymfony/FOSRestBundle`
[42] Architektura rozhraní, navržená pro distribuované prostředí užívající HTTP [27]
[43] `https://github.com/schmittjoh/JMSSerializerBundle`

5.2 Struktura aplikace

Základní struktura aplikace je převzatá z oficiální dokumentace frameworku Smyfony a veškeré dělení složek odpovídá tamním doporučením. Symfony stejně jako velké množství jiných webových frameworků využívá návrhový vzor MVC (Model-View-Controller) [29]. Ten dělí aplikaci do tří logikých vrstev.

Model tvoří v aplikaci hlavní funkční základ. Velmi často se dělí do dalších vrstev dle logiky konkrétní aplikace. Model by neměl znát strukturu a pracovat s dalšími dvěma vrstvami, jedná se tedy o nejnižší stavební kámen.

View je vrstva, která obstarává vizuální část aplikace. Ve frameworku Symfony je pro tyto účely využita šablonová komponenta Twig[44], která se stará o generování HTML a Javascriptu. Dle konvencí jsou všechny tyto šablony uloženy ve složce `templates`.

Controller neboli řadič je vrstva zpracovávající požadavky uživatele a na jejich základě pak používá logiku zbylých vrstev. Na výstup uživateli pak vrací vykreslená data z vrstvy view.

Od nové verze 4 framewoku Smyfony se architektura aplikace přestává striktně dělit na bundles[45] a tento balíčkovací vzor zůstává doporučen pouze pro kód, který má být později využitelný i pro jiné aplikace [28]. Jelikož většina prvků aplikace je závislá na komponentách z kapitoly 5.1 a není v ní mnoho prostoru pro členění do dalších balíčků dle speciálních funkcionalit, je aplikce tvořena jednolitě. Název aplikace autor zvolil stejný jako u modulu pro zpracování dat a to ReVy.

5.2.1 Přístup k databázi

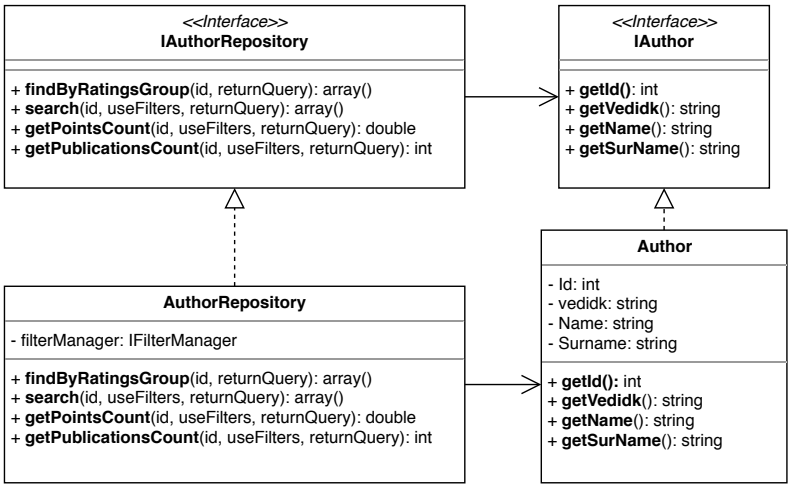
Komponenta Doctrine využívá návrhového vzoru Repository [30], který byl použit i při implementaci aplikace ReVy.

Entity je čistě mapovaná třída na tabulku v databázi. Pomocí jejích metod dochází k získávání dat z tabulky databáze a jejich uložení do atributů objektu.

Repository slouží k provádění agregací a dotazů nad databází. Třída se tak stará o načítání, ukládání a úpravu objektů `Entity`.

Pro možnou abstrakci celé databázové vrstvy aplikace jsou pro třídy Repository a Entity vytvořené interface, které definují jejich rozhraní. Ty bude nutné v případě výměny databáze znovu implementovat.

[44] `https://twig.symfony.com/`
[45] Název pro balíčky funkcionalit (tj. moduly) ve frameworku Symfony.



Obrázek 5.1: Diagram tříd pro návrhový vzor Repository pro autory.

5.2.2 Bloky a grafické prvky

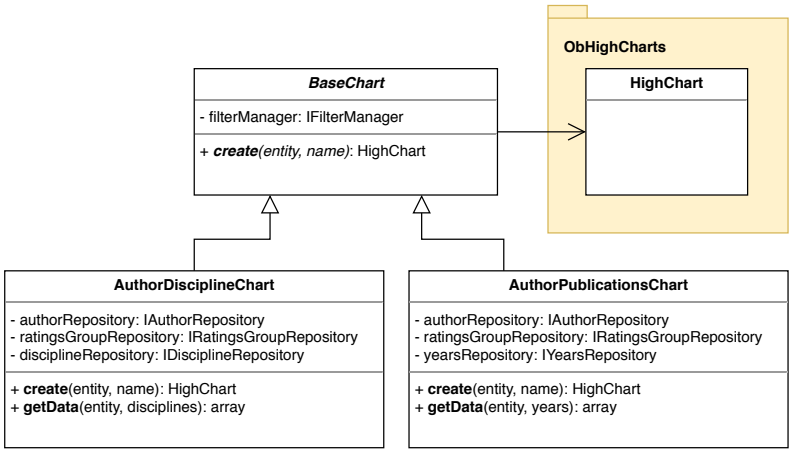
Aplikace je vizuálně zaměřená a místo rozsáhlé buisness logiky má nabízet pestrou prezentaci výsledných dat. Proto byly pro grafické prvky obsažené v blocích vytvořeny nové komponenty. Ty, které využívají externích komponent pak využívají návrhového vzoru Factory Method. Jedná se o vzor, který implementuje metodu pro vytvoření jiného objektu. Dochází tak k oddělení složitého inicializování některých objektů od dalších vrstev. V kontextu aplikace je tento vzor využit pro grafy a formuláře.

Grafy jsou v aplikaci umístěné ve složce `Graphics/Charts`. V nich jsou vytvářeny objekty komponenty ObHighCharts, které mají obsáhlé inicializování odpovídající api knihovny HighCharts[46]. Každý graf s odlišnými daty a jiným vizuálním zobrazením zde vlastní svou vlastní třídu. Výsledné vykreslení získaných objektů je vždy stejné, jelikož se jedná o kódy v javascriptu o které se starají využité knihovny a proto objekty nevyžadují speciální šablony.

Formuláře jsou vytvářeny dle dokumentace Symfony s využitím základní komponenty Forms[47]. Díky ní je možné jednoduše mapovat data na entity a provádět jejich jednoduchou validaci jak u klienta, tak při zpracovávání požadavku na serveru. Formuláře je možné najít ve složce `Forms`.

Stránkování využívají komponenty KnpPagination a stejně jako grafy vrací vytvořený objekt se specifickými daty. Pro zobrazení dané komponenty jsou definovány speciální šablony, jelikož každý objekt ve stránkování vyžaduje zobrazení jiných informací. Ty se nachází ve složce `templates/pagination`.

[46] `https://api.highcharts.com/highcharts/`
[47] `https://symfony.com/doc/current/forms.html`



Obrázek 5.2: Diagram tříd pro návrhový vzor „Factory method“ užitý u grafů.

Číselná porovnávání jsou v aplikaci implementovány jako třídy `Stars` ve složce `Graphics/Stars`. Ty získávají data z databáze a generují vlastní šablonu, která zobrazuje hodnotu pro daná data, průměr v dané oblasti a příslušný počet hvězd ukazující srovnání hodnoty s průměrem.

5.3 Vyhledávání

Pro vyhledávání v datech jsou využívány přednosti MySQL databáze. Problémové z pohledu efektivity může být fulltextové vyhledávání v názvech vědeckých výsledků a jmen a příjmení autorů.

Pro tyto potřeby by mohl být využit index `FULLTEXT` v databázi MySQL. Jak ovšem plyne z některých testů, je vyhledávání s tímto indexem velmi pomalé při kombinaci s obyčejnými MySQL indexy [31]. Proto je takové vyhledávání implementováno pomocí wildcard operátorů[48] v klauzuli `LIKE`.

Při vyhledávání je pro zadaná kritéria uplatňováno logické násobení (tedy výsledek musí splňovat všechna kritéria). Při zvolení více prvků jednoho kritéria je uplatňován logický součet (alespoň jeden prvek kritéria musí být splněn).

Pro přenos získaných argumentů z formulářů pro vyhledávání je vytvořena třída `FilterManager` implementující rozhraní `IFilterManager`. Ta ukládá jednotlivé argumenty zadané uživatelem a zpřístupňuje je pro třídy `Repository`, které následně provádějí vyhledávání v datech dle zadaných kritérií.

Jelikož jsou grafické prvky odděleny v třídách, může se stát, že se dotazují na stejná data z databáze. Z toho důvodu byla implementována třída `CalculationManager` rozhraní `ICalculationManager`, která funguje jako cache pro již jednou dotazovaná data.

[48] Operátor „%“. Značí, že místo něj může být jakýkoliv jiný řetezec.

5.4 Nacházení nekonzistencí a chyb

Pro hledání nekonzistencí a chyb v datech je vše připraveno díky databázovému návrhu z kapitoly 4.1. Ukládání všech řádek v hodnocení a dělení skupin hodnocení na jednotlivé roky totiž poskytuje možnost jejich srovnání.

5.4.1 Chybějící data

Pokud pro výpočet či zobrazení některých dat v databázi chybí data a tedy nebyla v hodnoceních uvedena, je v blocích vykreslena chybová hláška. Může se tak stát například u některých výskytů vědeckého výsledku, který nemohl být z nějakého důvodu ohodnocen. Důvod této chyby bývá uveden ve sloupci s popisem, který je v aplikaci zobrazen u informací o výsledku.

5.4.2 Nekonzistence mezi výskyty v hodnocení

V detailech skupin hodnocení jsou v tabulkách srovnávány jednotlivé výskyty a další skupiny stejného výsledku. Při nalezení nekonzistence je položka v tabulce označena žlutou barvou. Testovány jsou získané body a uvedený počet autorů.

5.4.3 Chybně dopočtené hodnoty

Při zobrazování získaných bodů dané organizace či její jednotky jsou kontrolovány výpočty podílu bodů. Celkový počet bodů a upravených bodů je vynásoben podílem organizace a testuje se, zdali je výsledek v souladu s uvedenými hodnotami v hodnocení.

V případě, že hodnota nesedí o méně než 0,5 bodu, je považována za správnou. V případě, že je chybná v rozmezí od 0,5 až do 3 bodů, je považována za lehkou chybu a je označena žlutě. V případě, že nesedí o více jak 3 body, je považována za hrubou chybu a je označena červeně.

Na závěr je proveden součet všech podílů organizací. Od tohoto součtu jsou odečteny hodnoty aktuálně zvolené skupiny a výsledek je zobrazen do posledního řádku tabulky. V případě, že jsou hodnoty záporné či nulové (součet podílů organizací je nižší než v celkovém hodnocení), je tato hodnota označena za správnou. V případě, že jsou hodnoty kladné, jedná se o chybu a dané hdonoty jsou opět zbarveny žlutě. Testován jsou kromě bodového ohodnocení i procenta podílů.

5.5 Vzhled aplikace

Při vytváření vzhledu aplikace se autor snažil vytvořit příjemné a barevně slazené prostředí, které bude plně responzivní a tedy přístupné i z nejmenších zařízení. Každý typ objektu dostává vlastní barvu, kterou se v aplikaci prezentuje. Vzhledem k tomu, že uživatelé využívající aplikaci, budou zřejmě uživatelé aplikace RIV, byly některé prvky jako například výsledky vyhledávání graficky inspirované právě aplikací RIV.

Pro grafickou reprezentaci kategorií v menu se autor rozhodl použít CSS font. Pro zobrazení vektorových obrázků mohl být použit formát SVG, nicméně ten je podporovaný až v pozdějších verzích některých prohlížečů [32]. Definici fontu aplikace autor vytvořil přes službu Fontastic[49], která umožňuje skloubit nezjnámější existující fonty s uživatelem nově vytvořenými ikonami. Ikony vytvořené pro aplikaci ReVy jsou zobrazeny na obrázku 5.3, zbylé byly převzaty z fontu Font Awesome[50].

Obrázek 5.3: Sada vytvořených ikon

[49] `http://fontastic.me`
[50] `https://fontawesome.com`

Kapitola 6

Testování a další možná rozšíření

6.1 Testování aplikace

Testování aplikace probíhá pomocí frameworku PHPUnit, který má pro podporu v Symfony vestavěnou komponentu PHPUnit Bridge[51]. S jeho pomocí byly otestovány nejproblémovější části kódu pomocí funkcionálních testů. Jednotkové testy v Symfony slouží převážně k testování buisness logiky [33], která v této aplikaci, vzhledem k její jednoduchosti, téměř není. Testy byly změřeny jak na databázové tak grafické části aplikace a REST API. Byla tak ověřena jejich základní i pokročilá funkčnost. Věkeré testy se nachází ve složce `tests`.

Dále proběhlo uživatelské testování vybráním dvou dobrovolníků, kteří byli seznámeni se základní strukturou dat v RIV. Ověřovala se funkčnost a intuitivnost uživatelského rozhraní aplikace. Na základě výtek těchto dobrovolníků bylo ostraněno hned několik grafických nedostatků jako špatné zobrazování šipek u řazení či špatně se přizpůsobující bloky při změně velikosti obrazovky. Dále byly přidány zaokrouhlení některých zobrazených hodnot z databáze a další nápovědy pro nabízená rozhraní jako například pro filtrování aktivit autora.

6.2 Propojení s IS VaVaI

Vzhledem k poskytování pouze částečných informací o výsledcích by mohlo být užitečné aplikaci napojit na oficiální systém VaVaI a získat tak přístup k informacím, které jsou shromažďovány v RIV, ale ne ve výsledcích hodnocení. Možnosti jak aktuální aplikaci napojit na IS VaVaI přichází v úvahu hned dvě.

[51] `https://symfony.com/components/PHPUnit%20Bridge`

První z nich je vyhledání výskytů vědeckých výsledků v RIV pomocí klasického webového formuláře (lze zvolit velmi široká kritéria jako například roky uplatnění zahrnující všechny výsledky). Následně lze provedést export nalezených výsledků do formátů, které jsou nabízeny (XML, ODS[52], CSV) [34]. Identifikátory nalezených výskytů výsledků[53] pak budou odpovídat v databázi aplikace sloupci `occurre_id` v tabulce `Ratings`. Získaná data je pak po zpracování možné uložit do nové tabulky či upravené tabulky skupin hodnocení a následně implementovat zobrazení těchto informací na detailních stránkách jednotlivých skupin hodnocení.

Druhá možnost může kopírovat styl, kterým pracuje aplikace Vyhledávání ve výsledcích Hodnocení VaVaI, uvedena v kapitole 3.1.3. Tedy stažením statických stránek pro jednotlivé skupiny hodnocení pomocí odkazu do RIV, který je uveden ve výsledcích hodnocení pro každý výskyt výsledku. To ovšem zabraňuje vlastní definici struktury stránek, jinak by bylo nutné stažené HTML soubory parsovat a získat z nich potřebná data, která by mohla být později opět uložena do databáze jak bylo navženo v předchozím odstavci.

6.3 Napojení na služby Web of Science a Scopus

Dalšími službami pro získání informací o časopisech a dodatečných informací o výsledcích, jako například citačních indexů, jsou databáze WoS a Scopus. Obě tyto služby poskytují rozhraní, pomocí kterého lze v jejich databázích vyhledávat údaje o vědeckých článcích. Díky tomu je možné v budoucnu implementovanou aplikaci o informace z těchto služeb snadno rozšířit.

Web of Science nabízí pro získání záznamů SOAP[54] API [35]. Dokumentace k němu ovšem není veřejně přístupná. K přístupu do API je nutné zažádat o vlastní klíč. Jelikož jsou v hodnocení výsledků uvedeny čísla pro vyhledání ve WoS, je možné je použít a snadno tak data z WoS získat a uložit například do nové tabulky v databázi navázané například na referenční skupinu hodnocení či tabulku s identifikátorem publikace.

Scopus nabízí pro získání dat RESTful API [36]. Pro jeho používání je opět nutné získat vlastní klíč. Nicméně v datech hodnocení není žádný odkaz do této webové služby. EID[55] a informace ze Scopus by ovšem mohly být získány z identifikátoru DOI [37]. Takové vyhledávání by pak mohlo být prováděno v případě, že se jedná o typ J_{sc} (tj. článek evidovaný na Scopus), čímž by mělo být zajištěno, že bude výsledek ve Scopus nalezen. Propojení s aplikací je v databázi opět možné jednoduchým přidáním další tabulky s referencí na skupinu hodnocení či tabulku s identifikátorem publikace.

[52] Otevřený formát tabulkového procesoru.
[53] V rozšířeném vyhledávání v RIV označeny jako *specifikace*.
[54] Protokol pro výměnu zpráv ve formátu XML.
[55] Identifikátor článků užitý ve Scopus.

Závěr

V této bakalářské práci byla provedena podrobná analýza dat z hodnocení vědeckých výsledků. Nejprve byla na základě informací z metodik hodnocení a popisů dat dodaných do RIV podrobně prozkoumána jejich struktura a poté byl popsán obsah jednotlivých sloupců v XSLX souborech s hodnoceními. Následně byly uvedeny nejzávažnější problémy a nekonzistence v datech, které komplikují další zpracování dat a byla navržena řešení, která se snaží chyby eliminovat případně omezit.

Na základě této analýzy bylo implementováno zpracování dat, které se kromě řešení základních chyb ve formátování snaží o rozeznání autorů a vědeckých výsledků co nejpřesnějším způsobem tak, aby bylo možné s co nejmenší chybovostí sledovat vědecké výsledky jednotlivých autorů a organizací. Implementovaný modul umožňuje stažení souborů s hodnocením na disk, či do struktur nad kterými může být v nástroji Jupyter prováděna další analýza a následné zpracování a uložení dat do MySQL databáze.

Druhá část práce byla věnována analýze, návrhu a implementaci webové aplikace, která zobrazuje informace získané z hodnocení. V aplikaci bylo umožněno vyhledání nejen hodnocení výsledků, ale také organizací, jejích jednotek a autorů dle nejdůležitějších kritérií. Na detailních stránkách pak aplikace zobrazuje nejzákladnější statistiky vědecké aktivity v oborech, organizacích a jejích jednotkách a snaží se o detailní přehled aktivit autorů s možností jejich filtrování. V přehledech hodnocení bylo zpřístupněno srovnání stejných hodnocení v průběhu let a byly zobrazeny nalezené nekonzistence v bodovém ohodnocení výsledků.

Výsledná aplikace byla otestována jak automatizovanými testy, tak testy uživatelskými. Na závěr bylo promyšleno napojení na systém RIV, služby Web of Science a Scopus a navrženo, jak nově získaná data napojit do implementované aplikace.

Modul na zpracování dat je společně s databází a aplikací připravený na další rozšíření. Možností, jak modul i výslednou aplikaci vylepšit či obohatit o nové funkcionality, je opravdu mnoho. Například připadá v úvahu zpřístup-

není možnosti exportu dat ze zpracování do formátu CSV, či předzpracování vybraných hodnot pro organizace či jednotky v rámci jednotlivých hodnocení a zpřístupnění tak filtrování i nad jejich aktivitami.

Jak však z analýzy dat a metodik vyplynulo, otázkou pro budoucí vývoj zůstává, zdali přehledy hodnocení nad všemi roky jsou relevantním směrem získávání informací a určování žebříčků autorů a organizací. Zároveň může vývoj této aplikace silně ovlivnit nová metodika hodnocení, která by mohla definovat naprosto odlišný styl bodování a v případě zapojení nových let pod touto metodikou, by tak mohlo dojít k ještě většímu znehodnocení již chybami ovlivněných hodnot, se kterými aplikace pracuje. Výsledek této práce tak prozatím může zřejmě sloužit pouze k obecnému přehledu a porovnávání pro jednotlivé organizace či autory a rychlého a přehledného přístupu k datům z hodnocení, nikoli jakožto plnohodnotný analytický nástroj.

Literatura

[1] MUSILOVÁ, Jana, Systém rozdělování financí – faktor omezující autonomii institucí v oblasti výzkumu, *Autonomie univerzit – je ohrožena?* [online]. Praha, Seminář Odborné skupiny ČFS JČMF Organizace výzkumu, 26. 3. 2009, [cit. 9. 3. 2018] Dostupné z `https://jcmf.cz/sites/default/files/osov/akademicke_forum_II_P4_musilova.pdf`

[2] ÚŘAD VLÁDY ČESKÉ REPUBLIKY. *Informační systém výzkumu, experimentálního vývoje a inovací* [online]. 2016 [cit. 19. 4. 2018]. Dostupné z: `https://www.rvvi.cz/`

[3] XLSX File Format. *WhatIs* [online]. 2010 [cit. 29. 4. 2018]. Dostupné z: `https://whatis.techtarget.com/fileformat/XLSX-Microsoft-Excel-Open-XML-Document`

[4] VLÁDA Č ESKÉ REPUBLIKY. Usnesení vlády ze dne 23. 6. 2004 k hodnocení výzkumu a vý voje a jeho výsledků. In: *Výzkum a vývoj v ČR* [online]. 23. 6. 2004 [cit. 10. 3. 2018]. Dostupné z: `http://www.vyzkum.cz/FrontClanek.aspx?idsekce=18748`

[5] Metodika hodnocení výsledků výzkumných organizací a hodnocení výsledků ukončených programů (platná pro léta 2013 až 2016). In: *Vývoj a výzkum v ČR* [online]. Úřad vlády ČR, 2013 [cit. 10. 4. 2018]. Dostupné z: `http://www.vyzkum.cz/storage/att/471EC8E44A7C3AA09C01B666F1ED6B30/M2013-0815-kor2.pdf`

[6] RADA PRO VÝZKUM, VÝVOJ A INOVACE. Hodnocení výzkumných organizací a hodnocení programů účelové podpory výzkumu, vývoje a inovací. In: *Výzkum a vývoj v ČR* [online]. 2015 [cit. 29. 4. 018]. Dostupné z: `http://www.vyzkum.cz/FrontClanek.aspx?idsekce=799796&ad=1&attid=825165,%20http://metodikahodnoceni.blogspot.cz/2018/02/hodnoceni-2017-z-prvni-ruky-3-dil.html`

[7] Předávání údajů do Informačního systému výzkumu, experimentálního vývoje a inovací RIV – Rejstřík informací o výsledcích 2016. In: *Výzkum a vývoj v ČR* [online]. Rada pro výzkum, vývoj a inovace, 6. 1. 2016 [cit. 12. 4. 2018]. Dostupné z: http://www.vyzkum.cz/storage/att/21BE166AE38CA071AD6B51251E2D193C/RIV16s0v2.pdf

[8] Popis předávaných tabulek: Etapa 3. In: *Výzkum a vývoj v ČR* [online]. Rada pro výzkum, vývoj a inovace [cit. 29. 4. 2018]. Dostupné z: http://www.vyzkum.cz/storage/att/CF7D2CBC496B32E414D0CD367552D07B/H13e3_Popis_tabulek.pdf

[9] Identifikátory. *ASEP* [online]. Knihovna AV ČR [cit. 15. 4. 2018]. Dostupné z: https://www.lib.cas.cz/asep/pro-zpracovatele/identifikatory/#RIV

[10] Tiskové prohlášení Konsorcia ČVUT v Praze a InfoScience Praha s.r.o. k Informačnímu systému výzkumu, experimentálního vývoje a inovací (IS VaVaI). In: *České vysoké učení technické v Praze: Zpravodajský servis* [online]. 2015, 10. 6. 2016 [cit. 15. 4. 2018]. Dostupné z: https://aktualne.cvut.cz/tiskove-zpravy/20160610-tiskove-prohlaseni-konsorcia-cvut-v-praze-a-infoscience-praha-sro-k

[11] JUPYTER STEERING COUNCIL. *Project Jupyter* [online]. 2014 [cit. 20. 4. 2018]. Dostupné z: http://jupyter.org/

[12] MCKINNEY, Wes. *Pandas* [online]. [cit. 27. 4. 2018]. Dostupné z: https://pandas.pydata.org/

[13] RICHARDSON, Leonard. Beautiful Soup. *Crummy: The Site* [online]. 2018 [cit. 27. 4. 2018]. Dostupné z: https://www.crummy.com/software/BeautifulSoup/

[14] *PyMySQL* [online]. 2018 [cit. 27. 4. 2018]. Dostupné z: https://pymysql.readthedocs.io/en/latest/

[15] THE PYTHON SOFTWARE FOUNDATION. Modules. *Python* [online]. 2018 [cit. 26. 4. 2018]. Dostupné z: https://docs.python.org/3/tutorial/modules.html

[16] Trendy oborové publikační výkonnosti pracovišť výzkumných organizací v České republice v letech 2008–2014. *IDEA-Apps* [online]. [cit. 17. 4. 2018]. Dostupné z: https://ideaapps.cerge-ei.cz/Trendy/

[17] ŠMERK, Pavel. Vyhledávání ve výsledcích Hodnocení VaVaI. *Fakulta Informatiky Masarykovy univerzity* [online]. [cit. 17. 4. 2018]. Dostupné z: https://www.fi.muni.cz/riv/ Dostupné z: https://www.fi.muni.cz/riv/

[18] ŠEBEK, Michael. IDEA - Trendy oborové publikační výkonnosti výzkumných pracovišť. In: *Diskusní fórum FEL* [online]. 24. 09. 2017 [cit. 8. 5. 2018]. Dostupné z: https://forum.fel.cvut.cz/topic/4143/idea-trendy-oborove-publikacni-vykonnosti-vyzkumnych-pracovist/

[19] VÝPOČETNÍ A INFORMAČNÍ CENTRUM. *V3S* [software]. České vysoké učení technické v Praze, 2015 [cit. 27. 4. 2018]. Dostupné z: https://v3s.cvut.cz/login

[20] GOOGLE, INC. *Google Analytics* [software]. 2005 [cit. 24. 4. 2018]. Dostupné z: https://www.google.com/intl/cs_ALL/analytics/index.html

[21] INNOCRAFT LTD. *Matomo* [software]. 2007 [cit. 27. 4. 2018]. Dostupné z: https://matomo.org/

[22] ŠPINAR, David. Pravidla tvorby přístupného webu. *Přístupnost* [online]. [cit. 27. 4. 2018]. Dostupné z: http://pristupnost.nawebu.cz/texty/pravidla-standardy.php?full

[23] NIELSEN, Jakob. *Usability 101: Introduction to Usability.* Nielsen Norman Group [online]. 4.1.2012 [cit. 20. 4. 2018]. Dostupné z: https://www.nngroup.com/articles/usability-101-introduction-to-usability/

[24] THE PHP GROUP. *Symfony* [software]. [cit. 27. 4. 2018]. Dostupné z: https://symfony.com/

[25] JQuery Tutorial. REFSNES DATA. *W3Schools* [online]. [cit. 15. 4. 2018]. Dostupné z: https://www.w3schools.com/jquery/

[26] Attribution-NonCommercial 3.0 Unported (CC BY-NC 3.0). CREATIVE COMMONS. *Creative Commons* [online]. [cit. 25. 4. 2018]. Dostupné z: https://creativecommons.org/licenses/by-nc/3.0/

[27] HANÁK, Drahomír. Stopařův průvodce REST API. *ITnetwork* [online]. [cit. 29. 4. 2018]. Dostupné z: https://www.itnetwork.cz/nezarazene/stoparuv-pruvodce-rest-api

[28] The Bundle System. REFSNES DATA. *Symfony* [online]. [cit. 27. 4. 2018]. Dostupné z: https://symfony.com/doc/current/bundles.html

[29] Symfony versus Flat PHP. *Symfony* [online]. [cit. 28. 4. 2018]. Dostupné z: https://symfony.com/doc/current/introduction/from_flat_php_to_symfony2.html

[30] Databases and the Doctrine ORM. *Symfony* [online]. [cit. 28. 4. 2018]. Dostupné z: https://symfony.com/doc/current/doctrine.html

[31] GRILLY, Nicolas. Don't Waste Your Time With MySQL Full-Text Search. In: *Hacker Noon* [online]. 5. 12. 2017 [cit. 28. 4. 2018]. Dostupné z: https://hackernoon.com/dont-waste-your-time-with-mysql-full-text-search-61f644a54dfa

[32] COYIER, Chris. Inline SVG vs Icon Fonts. In: *CSS* Tricks [online]. 22. 4. 2014 [cit. 28. 4. 2018]. Dostupné z: https://css-tricks.com/icon-fonts-vs-svg/

[33] Testing (Symfony Docs). *Symfony* [online]. [cit. 4. 5. 2018]. Dostupné z: https://symfony.com/doc/current/testing.html

[34] Export výsledků vyhledávání - RIV. ÚŘAD VLÁDY ČESKÉ REPUBLIKY. *Informační systém výzkumu, experimentálního vývoje a inovací* [online]. 2016 [cit. 27. 4. 2018]. Dostupné z: https://www.rvvi.cz/riv?s=rozsirene-vyhledavani&ss=export&n=0

[35] Web of Science web services (APIs). CLARIVATE. *Clarivate Analytics* [online]. 2018 [cit. 28. 4. 2018]. Dostupné z: http://wokinfo.com/products_tools/products/related/webservices/

[36] Elsevier Scopus APIs. ELSEVIER B.V. *Elsevier Developers* [online]. 2018 [cit. 28. 4. 2018]. Dostupné z: https://dev.elsevier.com/sc_apis.html

[37] KITCHIN, John. Getting a Scopus EID from a DOI. In: *Kitchin Research Group* [online]. 7. 6. 2015 [cit. 28. 4. 2018]. Dostupné z: http://kitchingroup.cheme.cmu.edu/blog/2015/06/07/Getting-a-Scopus-EID-from-a-DOI/

Dodatek A

Seznam použitých zkratek

API Application Programming Interface

CEA Centrální evidence aktivit

CEP Centrální evidence projektů

CSS Cascading Style Sheets

CSV Comma-separated values

ČVUT České vysoké učení technické

DOI Digital Object Identifier

HTML HyperText Markup Language

IDEA Institut pro demokracii a ekonomickou analýzu

IS VaVaI Informační systém výzkumu, experimentálního vývoje a inovací

MVC Model-View-Controller

ODS Open Document Spreadsheet

ORM Objektově relační zobrazení

PHP Hypertext Preprocessor

REST Representational State Transfer

RIV Rejstřík informací o výsledcích

SOAP Simple Object Access Protocol

SQL Structured Query Language

SVG Scalable Vector Graphics

VES Evidence veřejných soutěží

WoS Web of Science

XML Extensible markup language

Dodatek B

Obsah přiloženého DVD

- `readme.txt` ..... stručný popis obsahu DVD
- `README.md` ..... popis instalace aplikace
- `parser` ..... adresář obsahující implementaci zpracování dat
  - `revy.py` ..... modul pro zpracování dat
  - `download_data.ipynb` ..... notebook s parsovacím skriptem
  - `README.md` ..... návod na používání modulu
- `database` ..... adresář s databází pro aplikaci
- `revy` ..... složka se zdrojovými kódy aplikace
- `text` ..... složka s textem práce
  - `thesis.pdf` ..... text práce ve formátu PDF

Dodatek C

Instalační příručka

C.1 Požadavky

- **MySQL 5.5** a vyšší (5.7.21 doporučeno)
- **PHP 7.1** a vyšší (7.1 doporučeno)
- **Composer 1.6** a vyšší (1.6.3 doporučeno)
- **Python 3** a vyšší (3.5 doporučeno) s následujícími moduly:
  - Pandas
  - BeautifulSoup
  - PyMySQL
- **Jupyter** verze 4 a vyšší.

C.2 Instalace

1. Otevřete složku `revy`.
2. Spusťte příkaz `php composer.phar install`.
3. Otevřete soubor `.env` a přepište údaje pro připojení do vaší MySQL databáze.
4. Spuštěním příkazu `php bin/console doctrine:database:create` vytvoříte příslušnou databázi.
5. Příkazem `php bin/console doctrine:schema:update --force` vytvoříte databázové schéma, či aktualizujete aktuální schéma databáze na schéma, které používá aplikace.

6. Zpracujte data podle následující sekce C.3 či zkopírujte databázi z DVD do vaší složky s MySQL. Na Linuxu bývají databáze uloženy v `\var\lib\mysql`.

7. Příkazem `php bin/console server:run` spustíte built-in server Symfony a připojením na *localhost* v prohlížeči otevřete aplikaci.

C.3 Zpracování dat

1. Otevřete složku `parser`.

2. Spusťte příkazy `PYTHONHASHSEED=0` a `jupyter notebook`. Proměnná `PYTHONHASHSEED` je nastavena, aby metoda `hash()` vytvářela při každém zapnutí pro stejného autora stejný hash. Bez této možnosti nebudou správně fungovat testy v aplikaci.

3. V rozhraní aplikace Jupyter otevřete soubor `data_download.ipynb`.

4. V souboru se nachází příkazy pro zpracování všech dat či pouze testovacího vzorku. Spusťe jednotlivé příkazy v blocích za sebou podle toho, jaká data chcete získat. Zpracování dat pro všechna hodnocení trvala na testovacím stroji (4 GB RAM) přibližně 9 hodin. V druhé části skriptu se nachází testovací případ, který trval na testovaím stroji pouhých 40 minut a obsahuje přibližně 25% všech hodnocení.

5. Nyní máte data připravená v databázi, vraťte se proto k bodu 7 pro spuštění aplikace.

6. Více o používání modulu naleznete v souboru `parser/README.md`.