VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA INFORMAČNÍCH TECHNOLOGIÍ
FACULTY OF INFORMATION TECHNOLOGY

ÚSTAV INFORMAČNÍCH SYSTÉMŮ
DEPARTMENT OF INFORMATION SYSTEMS

PORTÁL PRO SROVNÁVÁNÍ VĚDECKÝCH VÝSLEDKŮ

PORTAL FOR COMPARING SCIENTIFIC RESULTS

BAKALÁŘSKÁ PRÁCE
BACHELOR'S THESIS

AUTOR PRÁCE — **JOZEF HARAG**
AUTHOR

VEDOUCÍ PRÁCE — **Ing. RADEK BURGET, Ph.D.**
SUPERVISOR

BRNO 2019

Vysoké učení technické v Brně
Fakulta informačních technologií

Ústav informačních systémů (UIFS) Akademický rok 2018/2019

Zadání bakalářské práce

Student: **Harag Jozef**
Program: Informační technologie
Název: **Portál pro srovnávání vědeckých výsledků**
Portal for Comparing Scientific Results
Kategorie: Informační systémy

Zadání:

1. Seznamte se s existujícími datovými zdroji obsahujícími údaje o vědeckých výsledcích univerzit v UK.
2. Prostudujte současné technologie pro vizuální prezentaci rozsáhlých dat ve webových aplikacích zahrnující prvky Online Analytical Processing (OLAP).
3. Navrhněte architekturu systému umožňujícího sumarizaci a srovnání vědeckých výsledků univerzit podle různých kritérií a v různých časových intervalech.
4. Implementujte navržený systém s využitím vhodných technologií.
5. Proveďte testování systému na reálných datech.
6. Zhodnoťte dosažené výsledky.

Literatura:

- Dokumentace Google Charts, https://developers.google.com/chart/.
- Dokumentace projektu ClickHouse, https://clickhouse.yandex/docs/en/.
- Knoth, P. and Zdrahal, Z. (2012) CORE: Three Access Levels to Underpin Open Access, D-Lib Magazine, 18, 11/12, Corporation for National Research Initiatives.

Pro udělení zápočtu za první semestr je požadováno:

- Body 1 až 3.

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

Vedoucí práce: **Burget Radek, Ing., Ph.D.**
Vedoucí ústavu: Kolář Dušan, doc. Dr. Ing.
Datum zadání: 1. listopadu 2018
Datum odevzdání: 15. května 2019
Datum schválení: 16. října 2018

Abstrakt

Práca sa zaoberá tvorbou portálu pre porovnávanie vedeckých výsledkov medzi inštitúciami v Spojenom kráľovstve Veľkej Británie a Severného Írska. Cieľom práce je vytvorenie funkčného prototypu analytického nástroja, ktorý umožní užívateľom vytvárať vzorky dát na základe určitých spoločných charakteristík a nad týmito vzorkami realizovať pokročilú dátovú analýzu, ktorej výstupom sú rôzne typy grafov. Grafy je možné umiestniť na hlavný panel. Tento panel je možné prispôsobovať podľa vlastných predstáv a preto ho môže mať každý požívateľ rozdielny. Medzi analyzovateľné metriky patrí počet citácií, sociálnych referencií a čitateľov, ktoré získali jednotlivé vedecké publikácie.

Abstract

Thesis deals with the creation of a portal for comparing scientific results between institutions in the United Kingdom of Great Britain and Northern Ireland. This thesis aims to create a functional analytic tool prototype that allows users to generate data samples based on specific common characteristics and to perform advanced data analysis over these samples, resulting in different types of graphs. Graphs can be placed on the main panel. This panel can be customized as per their own preferences so each user can have it different. Analyzable metrics include the number of citations, social references, and readers that individual scientific publications have received.

Kľúčové slová

analytický nástroj, vedecké výsledky, metriky publikácií, Yandex Clickhouse, CORE, Microsoft Academic Graph, Mendeley Readership, Crossref Event Data, OLAP

Keywords

analytic tool, scientific results, metrics of publiations, CORE, Microsoft Academic Graph, Mendeley Readership, Crossref Event Data, OLAP

Citácia

HARAG, Jozef. *Portál pro srovnávání vědeckých výsledků.* Brno, 2019. Bakalářská práce. Vysoké učení technické v Brně, Fakulta informačních technologií. Vedoucí práce Ing. Radek Burget, Ph.D.

Portál pro srovnávání vědeckých výsledků

Prehlásenie

Prehlasujem, že som túto bakalársku prácu vypracoval samostatne pod vedením pána Ing. Radka Burgeta, PhD. Ďalšie informácie a prístup k databáze CORE a Microsoft Academic Graph mi poskytol Ing. Petr Knoth, PhD, ktorý bol mojím supervízorom na Erasmus stáži. Uviedol som všetky literárne zdroje a publikácie z ktorých som čerpal.

.....
Jozef Harag
5. mája 2019

Poďakovanie

Touto cestou by som sa chcel poďakovať vedúcemu práce Ing. Radkovi Burgetovi, PhD. za cenné rady, podporu a usmernenie počas vypracovávania teoretickej časti bakalárskej práce. Za pomoc pri vypracovávaní praktickej časti a konzultácie by som chcel poďakovať Ing. Petrovi Knothovi, PhD..

V neposledom rade sa chcem poďakovať mojej rodine a priateľom za podporu a trpezlivosť, najmä mojím sestrám Mgr. Alžbete Haragovej a Mgr. Paulíne Haragovej za pomoc s formálnou úpravou práce.

Obsah

Kapitola 1

Úvod

V súčasnosti je badateľný v takmer všetkých vedných odboroch trend digitalizácie údajov a ich presun do online verzie. Ako zdroj informácií už nie je potrebné používať len knižnice, pretože kvalitné odborné materiály je možné vyhľadať rýchlo a jednoducho z pohodlia domova. Vzniká však potreba rozlíšiť kvalitu prác od ich kvantity. Na tento účel je vhodné využiť analytický nástroj, vďaka ktorému si vedia zodpovedné osoby na univerzitách porovnať svoju inštitúciu s inými inštitúciami.

Vo svete existuje viacero analytických nástrojov pre porovnávanie vedeckých výsledkov[1]. Často ide o komerčné aplikácie, ktoré sú veľmi komplexné a na jednoduchú analýzu nevyhovujúce. Medzi najznámejšie a najkomplexnejšie patrí platený nástroj nazývaný *SciVal* od spoločnosti *Elsevier* ponúkajúci dáta z 10 500 výskumných inštitúcií, ktoré pochádzajú z 230 štátov sveta. Ďalším významným hráčom na trhu je nástroj *InCites* vyvíjaný firmou *Clarivate Analytics*[2]. Obvyklá cena týchto nástrojov je 35–50 tisíc libier ročne (v prepočte cca. 1–1.5 milióna českých korún).

V rámci Českej republiky som nenašial žiadny podobný analytický nástroj, avšak väčšina vyššie spomenutých nástrojov získava dáta z inštitúcií po celom svete, vrátane českých univerzít. Podľa môjho názoru preto nie je problém analyzovať aj dáta z univerzít v Českej republike.

1.1 Ciele práce

Cieľom práce bolo pomôcť univerzitám analyzovať ich výkonnosť v publikačnej oblasti pomocou rôznych metrík, získaných z otvorene dostupných zdrojov. Bola snaha proces analýzy jednotlivých metrík užívateľovi čo najviac uľahčiť. Analyzovateľné metriky dostupné v tomto nástroji boli hlavne: počet citácií, počet spomenutí na sociálnych médiách a počet čitateľov.

Ako je už vyššie spomenuté existuje viacero nástrojov na porovnávanie vedeckých prác, a preto sa môže vynárať otázka, čím bol mnou vyvíjaný nástroj odlišný. Hlavným rozdielom v porovnaní s konkurenciou bola z môjho pohľadu maximálna užívateľská jednoduchosť. Nového užívateľa nie je potrebné vôbec zaučiť do obsluhy systému a vie sa rýchlo a pohodlne dostať k potrebným informáciám, čo bol zároveň aj jeden z cieľov tejto práce. Pred tvorbou tohto portálu som mal možnosť vidieť ukážku nástroja *SciVal*, ktorý mal množstvo dobrej

[1] Podrobný prehľad nástrojov na porovnávanie a vyhodnocovanie vedeckých portálov je možné nájsť aj na stránke Wikipedia Comparison of research networking tools and research profiling systems

[2] Viac informácií o firme *Clarivate Analytics* je môžné získať priamo na stránkach týchto aplikácií – https://www.scival.com/, resp. https://clarivate.com/products/incites/

funkcionality, ale svojou robustnosťou nebol natoľko intuitívny, aby bol vhodný pre bežného používateľa. Podľa môjho názoru sa k základným informáciám dalo dostať pomerne zložito a vyžadovalo si to viackrokové preklikanie sa k zdroju informácií. Túto skutočnosť považujem za výrazné negatívum nástroja. Z vlastnej skúsenosti viem, že užívateľ chce často vidieť stále tu istú, avšak aktualizovanú informáciu a nepotrebuje využívať žiadne ďalšie funkcionality. Ďalšou veľkou výhodou portálu je využitie databázy projektu *CORE*, ktorá obsahuje *Open Access* (3.1.2 Pojem Open Access) dokumenty s abstraktom, resp. plným textom, vďaka čomu sa dajú vytvárať kvalitnejšie a presnejšie vzorky dát. V neposlednom rade je v tomto nástroji poskytnutá možnosť stiahnutia si údajov do svojho zariadenia, vďaka čomu si môže užívateľ tieto dáta importovať do rôznych programov, napr. do aplikácie *Excel* a pracovať s nimi ďalej podľa vlastnej potreby.

1.2 Logické členenie práce

V kapitole 2 On-Line Analytical Processing systémy je podrobne vysvetlený princíp *On-Line Analytical Processing* systémov. Portál obsahuje prvky týchto systémov a preto bolo nevyhnutné špecifikovať ich funkcionalitu.

Nasledujúce tri kapitoly 3 Zdroje dát, 4 Návrh a implementácia podporných služieb pre portál a 5 Štatistiky získaných dát, pojednávajú o tvorbe podporných služieb pre portál.

Ďalšie dve kapitoly 6 Návrh portálu pre porovnávanie vedeckých výsledkov a 7 Tvorba portálu pre porovnávanie vedeckých výsledkov sú zamerané na samotnú tvorbu portálu.

V poslednej kapitole 8 Záver som zhodnotil dosiahnuté výsledky a načrtol problémy a vylepšenia, ktoré by sa mohli v budúcich verziách realizovať.

Kapitola 2

On-Line Analytical Processing systémy

V teórii informačných systémov zaraďujeme *On-Line Analytical Processing* (ďalej len OLAP) systémy do kategórie systémov pre podporu rozhodovania. Tieto systémy nekladú dôraz až tak na aktuálnosť údajov ako na rýchlosť zodpovedania agregačných dotazov a ich prehľadnosť. Často sa v nich pracuje s dátami reprezentujúcimi určité časové obdobie – deň, týždeň, mesiac, rok [17].

V tejto kapitole je detailne predstavený pojem OLAP systémov, ktorý bol nevyhnutnou prerekvizitou pri tvorbe portálu (portál mal obsahovať prvky OLAP systémov).

2.1 Pojem OLAP

Pojem OLAP bol zavedený v septembri 1993 vo vedeckom článku s názvom *Providing OLAP (on-line analytical processing) to user-analysts: An IT mandate* publikovanom autorskou trojicou Edgar F. Codd, Sharon B. Codd a Clynch T. Salley [8]. Je to metodológia, ktorá poskytuje koncovému užívateľovi (zvyčajne ide o manážerov a analytikov firiem) prístup k rozsiahlím objemom dát a pomáha k ich pochopeniu. Typické pre OLAP sú totiž možnosti vizualizácie. Užívateľ má možnosť vytvárania rôznych typov grafov ako aj nahliadať na dáta v numerickej podobe [3].

Aby sa systém dal považovať za OLAP systém musí spĺňať 12 pravidiel, ktoré boli publikované vo vyššie spomenutom článku [8].

1. **Multidimenzionálny konceptuálny pohlaď** – systém by mal podporovať EIS (*Executive Information System*) *slice* a *dice* operácie (viď. 2.2.2 Základné operácie v OLAP kocke).

2. **Transparentnosť** – systém by mal byť súčasťou architektúry otvorených systémov. Technológia OLAP systému, podriadená databáza a architektúra výpočtu by mala byť pre užívateľa transparentná [20]. Na druhú stranu by užívateľ nemal vidieť zdroj dát, ktorý môže byť tak hetoregénny ako aj homogénny.

3. **Dostupnosť** – systém by mal pristupovať iba k tým údajom, ktoré sú skutočne potrebné na vykonanie požadovanej analýzy.

4. **Konzistentné vykazovanie** – užívateľ by nemal pocítiť spomalenie postupom času pri pribúdaní záznamov.

5. **Generická dimenzionalita** – systém by nemal byť obmedzený na 3D ani ovplyvnený žiadnym konkrétnym rozmerom. Funkcia aplikovaná na jeden rozmer by sa mala dať aplikovať aj na iný rozmer.

6. **Architektúra klient-server** – systém by mal spĺňovať princípy architektúty klient-server, pričom by nemalo byť náročné pripojiť ďalšieho klienta.

7. **Dynamická manipulácia s riedkými maticami** – fyzická štruktúra servera OLAP by mala manipulovať s riedkymi maticami (nie každá bunka musí nutne obsahovať nejakú hodnotu) optimálne.

8. **Podpora viacerých užívateľov** – systém musí mať podporu viacerých užívateľov, ktorí sú schopný pracovať nad rovnakým modelom.

9. **Neobmedzené krížové dimenzionálne operácie** – všetky dimenzie by mali byť vytvárané rovnocenne, takže všetky formy výpočtu musia byť povolené naprieč všetkými dimenziami [6].

10. **Intuitívna manipulácia s údajmi** – užívateľské rozhranie by malo umožnovať intuitívnu manipuláciu s dátami (viď. 2.2.2 Základné operácie v OLAP kocke) bez nutnosti preklikávania sa viacerými krokmi.

11. **Flexibilné vykazovanie** – systém by mal prezentovať informácie užívateľovi presne tým spôsobom, ktorý chce používateľ použiť.

12. **Neobmedzené dimenzie a agregácie** – systém by nemal byť žiadnym spôsobom obmedzený na počet dimenzií, resp. agregácií.

2.2 Dátová kocka OLAP

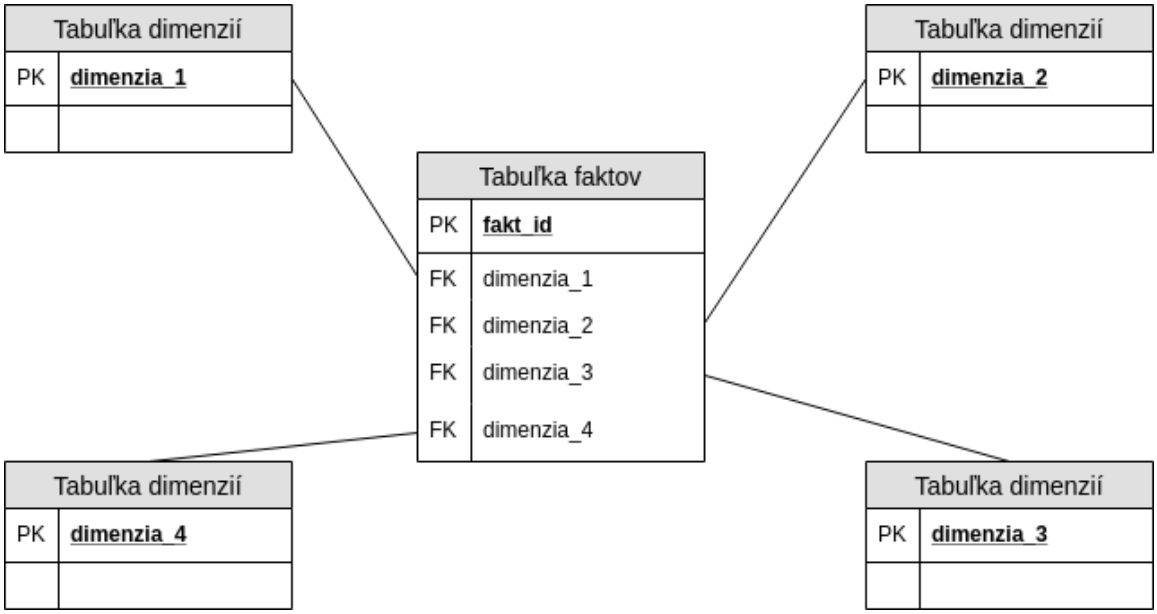
Technológia OLAP pracuje s multidimenzionálnymi dátami. Tie sú reprezentované viacrozmernou dátovou kockou nad ktorou sú definované operácie. Matematicky si pod dátovou kockou môžeme predstaviť viacrozmerné matice. Každá OLAP kocka obsahuje dva typy údajov – fakty a dimenzie. Fakty sú chápané ako numerické merné jednotky obchodovania, ktoré sú uložené v tabuľke faktov. Táto tabuľka je najväčšia v databáze a spravidla je len jedna. Dimenzie obsahujú logicky alebo organizačne hierarchicky usporiadané údaje. Tabuľky dimenzií sa nemenia tak často a sú menšie ako tabuľky faktov. Najčastejšie sa používajú časové, geografické a produktové dimenzie [20].

2.2.1 Reprezentácia dátovej kocky OLAP

Dátová kocka OLAP sa najčastejšie reprezentuje hviezdicovou schémou alebo schémou „snehovej vločky“. Je to topologické usporiadanie tabuľky faktov a tabuliek dimenzií.

Hviezdicová schéma

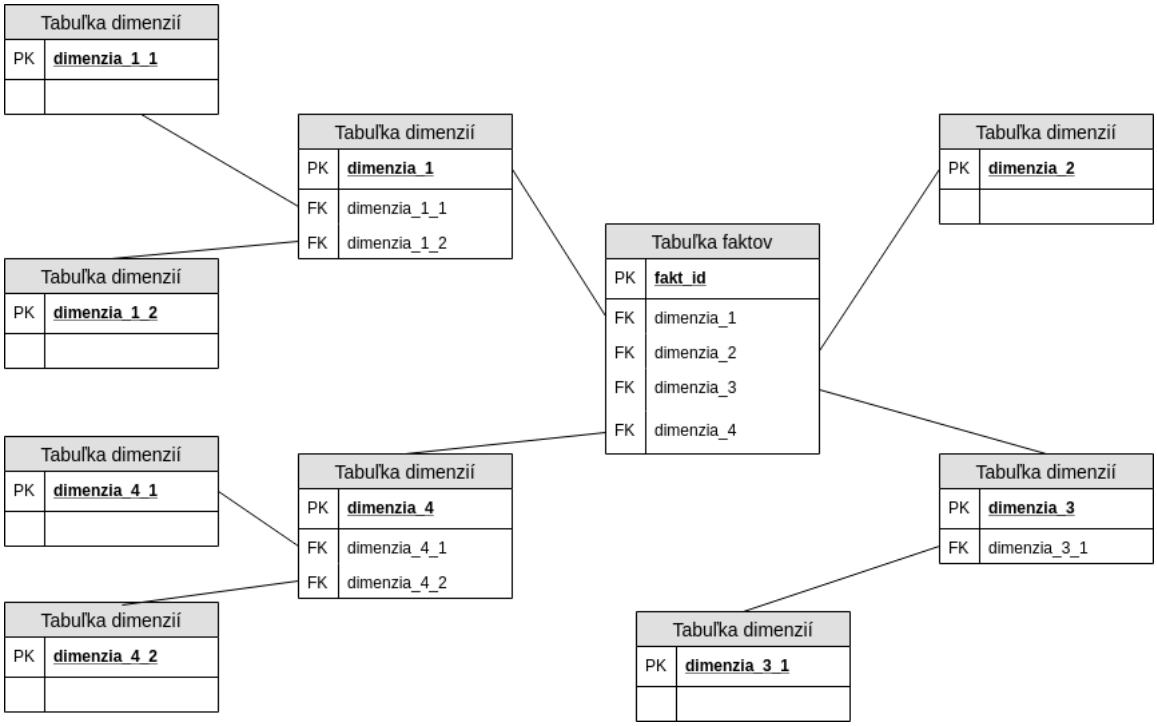
Hviezdicová schéma obsahuje tabuľku faktov, ktorá obsahuje cudzie kľúče odkazujúce na tabuľky dimenzií. Tabuľky dimenzií už ďalej neobsahujú cudzie kľúče na iné tabuľky. Dimenzie v tejto schéme nie sú normalizované. Táto schéma poskytuje vysoký dotazovací výkon [20]. Na obr. č. 2.1 je vidieť jednoduchý príklad hviezdicovej schémy.



Obr. 2.1: Príklad hviezdicovej schémy.

Schéma „snehovej vločky“

Schéma „snehovej vločky“ má podobne ako hviezdicová schéma tabuľku faktov, na ktorú naväzujú tabuľky dimenzií. Na rozdiel od hviezdicovej schémy v schéme „snehovej vločky“ môžu na tabuľky dimenzií naväzovať ďalšie tabuľky dimenzií. Vďaka tomu sú dimenzie normalizované, ale dotazovací výkon je podstatne nižší, pretože schéma obsahuje väčšie množstvo tabuliek. Na obr. č. 2.2 je vidieť jednoduchý príklad schémy „snehovej vločky“.



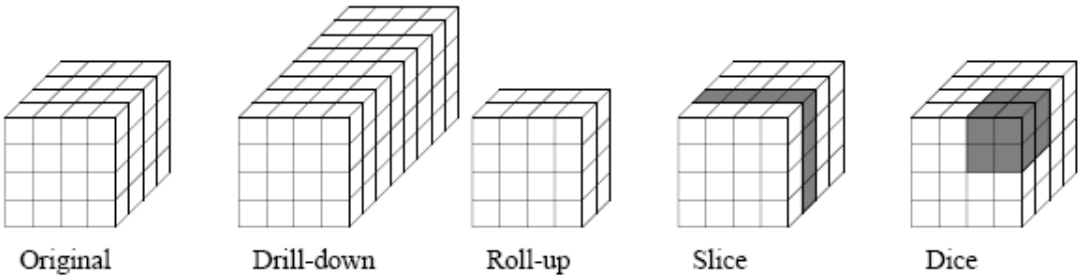
Obr. 2.2: Príklad schémy „snehovej vločky“.

2.2.2 Základné operácie v OLAP kocke

V OLAP kocke sú definované základné operácie, ktoré určujú ako sa dá s dátami manipulovať [9].[1] Prvé štyri operácie sú zobrazené aj na obr. č. 2.3.

- **Drill-Down** – umožňuje posun v hierarchickom smere k detailnejšej úrovni dimenzie. Je opakom operácie *Roll-Up*.
- **Roll-Up** – umožňuje posun v hierarchickom smere k obecnejšej úrovni dimenzie. Je opakom operácie *Drill-Down*.
- **Dicing** – umožňuje obmedziť jednu alebo viac dimenzií a vytvára podmnožinu obsahujúcu dva alebo viaceré prvky.
- **Slicing** – umožňuje výber jednej hodnoty pre jednu z dimenzií.
- **Pivoting** – umožňuje zmeniť osi datovej kocky OLAP a tým pádom je možný pohľad na dáta z rôznych uhlov.

[1] Základné názvy operácií definované nad OLAP kockou sú zachované v anglickom jazyku, pretože je to i zvykom medzi odborníkmi v obore.



Obr. 2.3: Grafické zobrazenie základných operácii v OLAP kocke. Prevzaté z [31].

Kapitola 3

Zdroje dát

Prvým krokom pri vytváraní nového portálu bolo získanie potrebných dát, vďaka ktorým si užívateľ môže vytvárať rôzne vzorky dát a tie následne analyzovať. Metriky jednotlivých vedeckých prác, ktoré boli zaujímavé z pohľadu analýzy, boli hlavne počet citácií, čitateľov a sociálnych signálov.

V tejto kapitole som vymedzil množinu dát, s ktorou som ďalej pracoval. V druhej časti tejto kapitoly som analyzoval dostupné dátové zdroje, ktoré poskytli potrebné metriky, t.j. citačné dáta, sociálne signály a množstvo čitateľov na platforme *Mendeley*.

3.1 Pojmy a množina dát zahrnutá v projekte

Vzhľadom nato, že na internete je dostupné veľké množstvo dát, či už legálnych alebo nelegálnych, bolo potrebné zadefinovať s akými dátami sa v tomto projekte bude pracovať.

3.1.1 Vymedzenie množiny dát pre projekt

V tejto práci som pracoval len s dokumentmi, ktoré

(a) spĺňali definíciu **Open Access** (viď 3.1.2 Pojem Open Access)

(b) a mali **jednoznačný identifikátor DOI** (viď. 3.1.3 Pojem DOI).

Ďalej všetky metriky dostupné k týmto dokumentom boli

(a) **voľne dostupné** na internete

(b) a bolo ich možné použiť podľa nimi definovaných licenčných podmienok.

3.1.2 Pojem Open Access

Pod pojmom *Open Access*, podľa Budapeštianskej iniciatívy pre voľný prístup (z angl. *Budapest Open Access Initiative*), sa rozumie literatúra, ktorá je voľne dostupná na internete.

> *„Každý užívateľ ma garantovanú možnosť čítať, sťahovať, kopírovať, distribuovať, tlačiť, vyhľadávať alebo odkazovať na plné texty týchto článkov, prehľadávať ich na indexovanie, odovzdať ich ako zdroj informácií pre software, alebo ich použiť na akýkoľvek iný zákonný účel, bez finančných, právnych, alebo technických prekážok, iných, ako sú tie, ktoré sú neoddeliteľné od získania prístupu k internetu*

ako takému. Jediným obmedzením v oblasti reprodukcie a distribúcie a jedinou úlohou autorských práv v tejto oblasti by malo byť poskytnutie kontroly integrity vlastnej práce autorom a právo byť riadne uznaný a citovaný.“ [25]

3.1.3 Pojem DOI

Každá vedecká práca má perzistentný a unikátny identifikátor používaný na jednoznačné určenie digitálneho objektu spravovaný neziskovou organizáciou *International DOI Foundation* (IDF)[13] a ďalej štandardizovaný Medzinárodnou organizáciou pre normalizáciu (ISO) v norme ISO 26324[1] z roku 2012 [11]. Tento identifikátor sa nazýva *Digital Object Identifier* (ďalej len DOI).

Pojem Digitálny repozitár

Pod pojmom digitálny repozitár alebo len repozitár sa rozumie úložisko metadát dokumentov, ktoré sú dostupné pomocou protokolu OAI PMH (viď. 3.1.4 Pojem OAI-PMH).

3.1.4 Pojem OAI-PMH

OAI-PMH (*Open Archives Initiative Protocol for Metadata Harvesting*) je protokol umožňujúcici získavanie metadát o dokumentoch uložených v digitálnych repozitároch, vytvorený Iniciatívou otvorených archívov (*Open Archives Initiative*, OAI).

3.2 Databáza projektu CORE

Databáza projektu CORE (z angl. ***Co**nnecting **Re**positories*) mi poslúžila ako hlavný a smerodajný zdroj dát. Cieľom projektu CORE je agregovať všetky práce s voľným prístupom z repozitárov a žurnálov z celého sveta a bezplatne ich sprístupniť širokej verejnosti. Na rozdiel od projektu *Google Scholar*[2] CORE obohacuje získané dáta o dalšie informácie a sprístupňuje ich abstrakt, resp. celý text, pokiaľ je dostupný [15]. Medzi informácie, ktoré CORE pridáva k získaným dokumentom patrí napríklad aj kategória. Vďaka pridaniu tejto kategórie je možné monitorovať trendy a odhadnúť rast výskumu v konkrétnych odboroch. Projekt CORE taktiež extrahuje citácie z publikácií a vytvára prepojenia medzi nimi [18].

3.2.1 Prístup k databáze CORE

K tejto databáze sa dalo pristupovať tromi rôznymi spôsobmi. V pravidelných intervaloch (cca. raz ročne) je vytváraný snímok (z angl. *snapshot*) celej databázy, ktorý je následne možné stiahnúť z webovej stránky projektu[3]. Išlo však o pomerne veľké množstvo dát, ktoré bolo treba spracovať a zaindexovať do vlastnej databázy.

Druhým spôsobom bol prístup cez REST API rozhranie[4]. Toto rozhranie je však značne limitované na počet požiadaviek poslaných na server v danom čase, a teda nepoužiteľné.

Poslednou možnosťou prístupu bolo získanie údajov priamo z databázy CORE. Vzhľadom na to, že portál pre porovnávanie vedeckých výsledkov som vytváral v rámci programu

[1] Norma ISO 26324 je dostupná na adrese https://www.iso.org/obp/ui/#iso:std:iso:26324:ed-1:v1:en.

[2] Viac informácií o projekte *Google Scholar* je možné získať na adrese https://scholar.google.com/intl/en-US/scholar/about.html.

[3] Všetky snímky CORE databázy sú dostupné na adrese https://core.ac.uk/servicesy#dataset.

[4] Viac o limitáciach CORE API je dostupné na adrese https://core.ac.uk/services#api.

Erasmus+ Traineeship na *The Open University*, bol mi umožnený prístup k tomuto zdroju dát.

3.3 Microsoft Academic Graph

Microsoft Academic Graph (ďalej len MAG) je heterogénny graf obsahujúci záznamy o vedeckých publikáciach a ich citačných vzťahov. Taktiež obsahuje informácie o autoroch, inštitúciách, žurnáloch, konferenciách a oblasti štúdia, ktorej sa daná publikácia venuje. Spoločnosť *Microsoft* ho využíva v projektoch *Bing*, *Cortana*, *Word* a v *Microsoft Academic*. Tento graf je aktualizovaný v týždenných intervaloch [29].

3.3.1 Prístup k databáze Microsoft Academic Graph

Podobne ako aj k databáze CORE, tak aj k databáze MAG bolo možné pristupovať pomocou REST API rozhrania[5], ktoré bolo dostupné buď v bezplatnej verzii (mesačný limit poslaných požiadaviek na server), alebo v platenej verzii. Vzhľadom na počet a náročnosť požiadaviek, ktoré bolo potrebné zaslať na servery, bola táto možnosť prístupu pre mňa nevyhovujúca.

Tento graf bol dostupný aj cez službu *Azure Storage*, ktorá bola síce spoplatnená, ale bolo možné stiahnuť si snímok tohto grafu a ďalej s ním pracovať lokálne. *The Open University* mala pre interné účely kópiu tohto grafu importovanú vo vlastnom skladisku veľkých dát (*Hadoop Distributed Filesystem* – HDFS). Nakoľko som pre tento projekt potreboval len podmnožinu dát, použil som riešenie, ktoré ponúkala *The Open University*. Existovala však aj možnosť získať tieto dáta pomocou voľne dostupnej sady dát pod názvom *Open Academic Graph*[6].

3.4 Crossref Event Data

Crossref je oficiálna nezisková organizácia pre registráciu DOI identifikátorov založená v januári 2000. Cieľom tejto organizácie je umožniť ľahko nájsť, citovať, prepojiť a hodnotiť vedecké výsledky [21]. K dosiahnutiu týchto cieľov *Crossref* vyvíja viacero podporných služieb. Jednou z nich je aj služba *Event Data*, ktorá zachytáva a ukladá rôzne udalosti z webových platforiem [32].

3.4.1 Udalosti zachytávané službou Event Data

Medzi udalosti, ktoré služba *Event Data* zachytáva patria hlavne komentáre, prepojenia, odkazy, zdieľania a referencie. Tieto udalosti sú zachytávané pre každú vedeckú prácu, resp. pre každý DOI objekt, pre ktorý má organizácia *Crossref* uložený záznam [32]. Kompletný zoznam ako aj zdroj dát, pre ktorú je udalosť zaznamenávaná, je uvedený v tabuľke 3.1.

[5] Dokumentácia REST API rozhrania pre MAG je dostupná na adrese https://azure.microsoft.com/en-us/pricing/details/cognitive-services/academic-knowledge-api/.

[6] Je nutné však podotknúť, že *Open Academic Graph* dataset bol naposledy aktualizovaný dňa 9. júna 2017 – https://aminer.org/open-academic-graph.

Zdroj dát	Názov zachytávanej udalosti
Cambia Lens	Citácie v patentoch
Crossref Metadata	Odkazy na registrovaný obsah DataCite
DataCite Metadata	Odkazy na registrovaný obsah Crossref
Hypothes.is	Anotácie v Hypothes.is
Newsfeed	Diskutované v správach a na blogoch
Reddit	Diskutované na Reddite
Reddit Links	Diskutované na stránkach, ktoré boli prepojené v subreddite
Stack Exchange Network	Diskutované na StackExchange stránkach
Twitter	Spomenutia v tweetoch
Wikipedia	Referencie na stránkach Wikipedie
Wordpress.com	Diskutované na stránkach Wordpress.com

Tabuľka 3.1: Kompletný zoznam udalostí s príslušným zdrojom dát pre službu *Crossref Event Data* [32].

3.4.2 Prístup k dátam zo služby Event Data

Jediným možným spôsobom ako sa dostať k dátam tejto služby bolo pomocou *Query API* rozhrania[7], ktoré bolo dostupné zdarma bez ďalších limitácií. Pre portál boli zaujímavé hlavne údaje o počte referencií na stránke *Wikipedia* a o počte spomenutí na sociálnej sieti *Twitter*.

3.5 Mendeley Readership dáta

Mendeley je bezplatný manažér referencií a akademická sociálna sieť vytvorená spoločnosťou *Elsevier*, ktorá má pomôcť výskumníkom organizovať ich výskum, spolupracovať s ostatnými online a vyhľadávať najnovšie informácie v konkrétnom výskume [26].

Mendeley Readership je metrika odzrkadlujúca angažovanosť vedeckých pracovníkoch na platforme *Mendeley*. Jednoducho povedané, ide o počet užívateľov, ktorí si pridali konkrétnu vedeckú prácu do svojej osobnej knižnice [5].

3.5.1 Prístup k Mendeley Readership dátam

Tak ako aj v predchádzajúcich prípadoch k *Mendeley Readership* dátam sa dalo dostať pomocou REST API rozhrania[8]. API rozhranie nebolo limitované, ale bolo nutné sa zaregistrovať a získať tak autorizačný token. K dispozícii bola aj SDK (*Software development kit*) knižnica pre rôzne programovacie jazyky, čo výrazným spôsobom uľahčovala prácu s týmito dátami.

[7] Dokumentácia *Crossref Query API* rozhrania je dostupná na adrese https://www.eventdata.crossref.org/guide/.

[8] Dokumentácia *Mendeley Readership* REST API rozhrania je dostupná na adrese https://dev.mendeley.com/methods/#introduction.

Kapitola 4

Návrh a implementácia podporných služieb pre portál

V tejto kapitole je detailne popísaný návrh a implementácia skriptov, ktoré zabezpečujú sťahovanie dátových zdrojov spomenutých v predchádzajúcej kapitole (3 Zdroje dát). Okrem iného je v tejto kapitole spomenutý aj princíp zlučovania dát a získania mapovania repozitárov univerzít.

4.1 Návrh

Cieľom bolo vytvorenie programu, resp. sťahovacích skriptov, ktoré umožnili stiahnúť každú metriku samostatne, nezávisle na sebe. Vďaka tomu bolo možné spustiť tieto procesy paralelne a urýchliť dobu sťahovania. Výstup každého skriptu bol súbor v textovom formáte *.csv*. Tieto súbory sa následne zobrali a pomocou finálneho skriptu spojili do jedného. Výsledný súbor slúžil pre import do databázy.

4.2 Implementácia

Pre implementáciu som sa rozhodol použiť programovací jazyk Python vo verzii 3.5. Dôvodom tohto výberu boli nielen moje predošlé skúsenosti so spomínaným programovacím jazykom, ale aj dostupnosť SDK knižnice pre dáta z *Mendeley Readership* platformy.

Program som vytvoril vo forme balíčka (z angl. *package*) s názvom `analytics`, ktorý obsahoval všetky potrebné skripty. Jedinou výnimkou bolo získavanie dát z MAG grafu, pretože jednoduchšie a rýchlejšie bolo napísať HiveQL dotazy, ktoré sa spustia manuálne v klustri ako samotný skript. Nebolo však možné plne paralelizovať proces sťahovania tak ako to bolo myslené v návrhu. Dôvodom bolo, že sťahovacie skripty pre metriky z MAG, *CrossRef Event Data* a *Mendeley Readership* potrebovali na vstup zoznam DOI identifikátorov, pre ktoré mali získať informácie. Z tohto dôvodu bolo prvým krokom stiahnutie všetkých dokumentov z projektu CORE publikovaných univerzitami v UK a obsahujúcich DOI identifikátor.

V programe sa nachádza jeden konfiguračný súbor umožňujúci zmeny rôznych nastavení, ako napríklad počet procesov sťahujúcich dáta z *Crossref Event Data*. Bližšie špecifikované nastavenia vrátane návodu na spustenie je možné nájsť v prílohe A Návod na spustenie skriptov na sťahovanie dát.

4.2.1 Získanie mapovania repozitárov britských univerzít

Vzhľadom na to, že tento projekt bol limitovaný len na dokumenty, ktoré boli publikované vedeckými inštitúciami v Spojenom kráľovstve Veľkej Británie a Severného Írska, nemohol som použiť všetky dáta z databázy CORE.

K získaniu potrebnej podmnožiny dát mi poslúžilo mapovanie repozitárov britských univerzít, ktoré som si manuálne vytvoril. Z dôvodu časovej náročnosti a možnej nepresnosti získaných výsledkov som nevytvoril žiadny skript, pomocou ktorého by to bolo možné automatizovať.

Projekt CORE harvestuje dáta z repozitárov pomocou protokolu OAI-PMH [18]. Pre každý repozitár má vytvorený unikátny identifikátor a k nemu priradený názov inštitúcie. Z týchto údajov som vybral len britské univerzity. Výsledkom bol súbor v textovom formáte *.csv*, ktorého časť je vyobrazená v tabuľke č. 4.1[1].

ID	Institute	Repositories IDs	Russel Group?
1	University of Aberdeen	[1]	0
2	Abertay University	[2]	0
3	Anglia Ruskin University	[6]	0
4	Aston University	[7]	0
5	Birkbeck, University of London	[8, 503]	0
6	Bournemouth University	[9]	0
7	University of Bradford	[10]	0
8	Brunel University	[14]	0
9	University of Reading	[17]	0
10	University of Central Lancashire	[18]	0

Tabuľka 4.1: Ukážka mapovania repozitárov univerzít. Celkový počet univerzít bol 139.

4.2.2 Russel group

> *„Russel group tvorí 24 najlepších UK univerzít, ktoré sa rozhodli viesť čo najlepší výskum, podávať vynikajúce výsledky v oblasti výučby a prepájať vzdelávanie s praxou v podnikoch a verejnom sektore. Skupina bola založená v roku 1994 a sídlo má v Londýne.“* [24]

V mapovaní (viď tabuľku č. 4.1) je možné vidieť stĺpec, ktorý odzrkadľuje, či sa daná univerzita nachádza (hodnota 1), alebo nenachádza (hodnota 0) v tejto skupine. Vo vedeckom portáli je vďaka tomu možné porovnávať nielen univerzity medzi sebou, ale aj vybranú univerzitu s univerzitami patriacimi do skupiny *Russel group*.

4.2.3 Stiahnutie britských prác z databázy CORE

Proces sťahovania britských prác z databázy CORE sa skladá z troch častí. Najskôr sa načítalo mapovanie repozitárov univerzít, následne sa poslal filtrovací dotaz na servery ElasticSearch projektu CORE, z ktorých sa stiahli postupne všetky dokumenty obsahujúce

[1] Je potrené podotknúť, že jedna univerzita môže mať aj viac repozitárov, preto je v stĺpci `Repositories IDs` uložené pole unikátnych identifikátorov.

DOI identifikátor a číslo repozitára sa nachádzalo v množine repozitárov získanej z mapovania. Sťahovali sa iba nasledovné informácie `CoreID` (unikátny idetifikátor dokumentu v rámci databázy CORE vo forme čísla), `DOI`, `PUB_YEAR` (rok publikácie dokumentu).

4.2.4 Získanie citačných dát z Microsoft Academic Graph

Získanie citačných dát riešili 4 HiveQL príkazy. V prvom kroku sa v Apache Hive vytvorila tabuľka z dát získaných z databázy projektu CORE. Nasledujúci HiveQL dotaz spojil importovanú tabuľku s relevantnými MAG tabuľkami a agregoval ich podľa DOI identifikátoru a roku citácie. Týmto dotazom sa získala informácia o tom, koľko citácií získali jednotlivé publikácie v konkrétnom čase. Po zmene agregácie na agregáciu iba podľa DOI sa získal celkový počet citácií pre jednotlivé publikácie.

4.2.5 Stiahnutie dát z platforiem CrossRef Event Data a Mendeley

Pre stiahnutie sociálnych signálov z platformy *Crossref Event Data* som vytvoril skript, ktorý sa vo viacerých vláknach dotazuje *CrossRef* API serveru na informácie o počte sociálnych signálov pre daný dokument. API server posiela všetky dostupné metriky, preto skript filtruje iba tie, ktoré som sa rozhodol integrovať do portálu, t.j. počet *Wikipedia* a *Twitter* referencií.

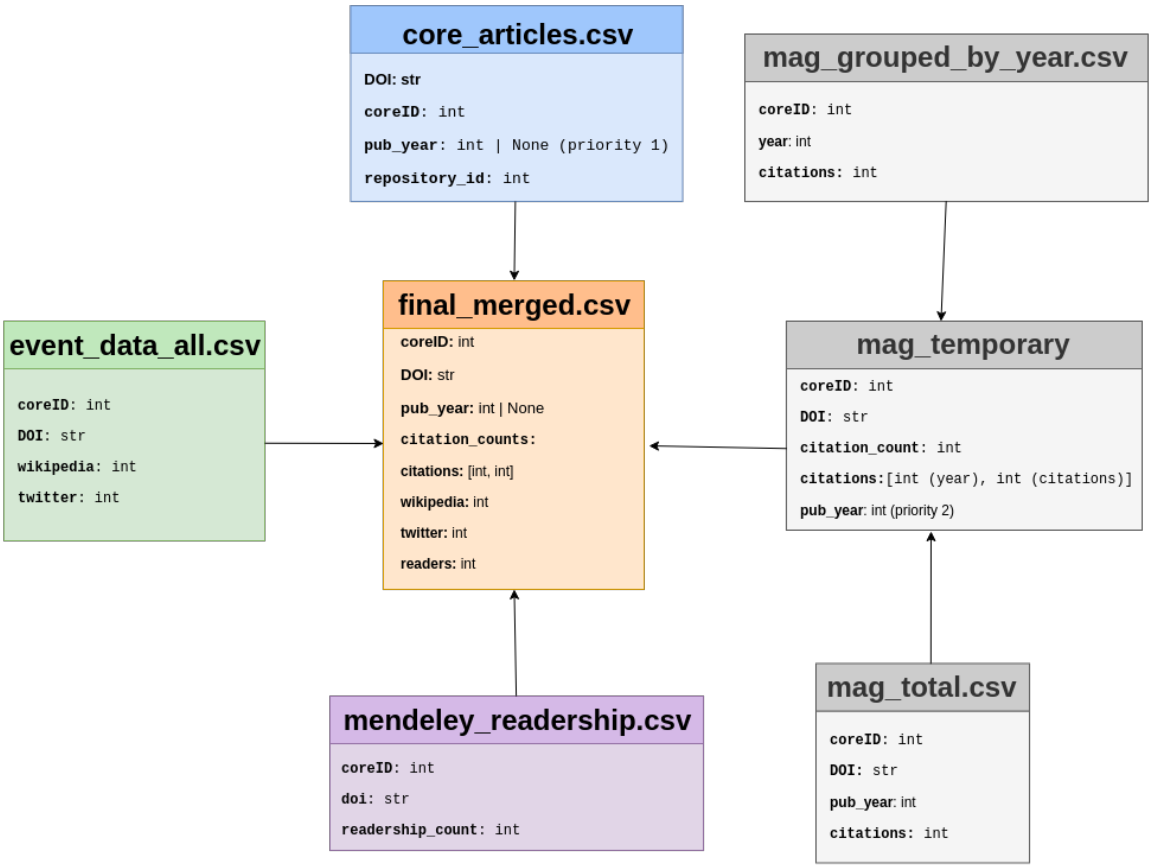
Pre získanie metriky o počte *Mendeley* čitateľov som využil oficiálnu knižnicu pre *Mendeley* API rozhranie napísanú v programovacom jazyku Python[2]. Práca s touto knižnicou bola jednoduchá a sťahovanie bolo priamočiare. Samozrejmosťou bolo sťahovanie dát vo viacerých procesoch.

4.2.6 Zlučovanie dát

Na účel zlúčenia dát do jedného súboru som naprogramoval skript, ktorý zobral všetky získané dáta a vytvoril z nich jeden súbor. Unikátnym kľúčom pri zlučovaní bol `coreID` identifikátor. Proces zlučovania dát je znázornený na obrázku č. 4.1. Údaj o roku publikovania bol dostupný v dvoch zdrojoch – CORE a MAG, pričom pri zlučovaní sa bral prioritne rok z dátovej sady CORE. Ak ho dátová sada CORE neobsahovala, bola zdrojom dátová sada MAG. Databáza projektu CORE totiž nemusela pri všetkých dokumentoch obsahovať rok publikácie a využitím údajov z MAG som zabezpečil presnejšiu dátovú sadu. Za povšimnutie stojí aj fakt, že v skripte sa zlučovali dáta podľa `coreID` identifikátora, čo je číslo, ktoré používa CORE databáza ako unikátny identifikátor. Bola to len preferencia. Rovnaký výstup by som získal, aj keby som ako unikátny kľúč pri zlučovaní použil DOI identifikátor.

Dáta z MAG boli dostupné v dvoch súboroch. V jednom súbore boli agregované podľa roku citácie a DOI identifikátora a v druhom iba podľa DOI identifikátora. Preto bolo nutné tieto údaje spojiť do jedného celku a následne ich zlúčiť s ostatnými zdrojmi dát. Z tohto dôvodu sa v pamäti vytvárala dočasná štruktúra, ktorá obsahovala `DOI`, `CORE_ID`, `citation_count` (celkový počet citácií pre daný dokument) a `citations` (pole, ktoré obsahovalo polia s dvoma prvkami – prvým prvkom je rok a druhým prvkom je počet citácií v danom roku.). Tento formát som zvolil kvôli tomu, že mi zjednoduchšil prácu pri importovaní dát do databázy portálu.

[2] *Mendeley* API knižnica pre programovací jazyk Python je dostupná na adrese https://github.com/Mendeley/mendeley-python-sdk.



Obr. 4.1: Postup pri zlučovaní dát do jedného súboru – `final_merged.csv`.

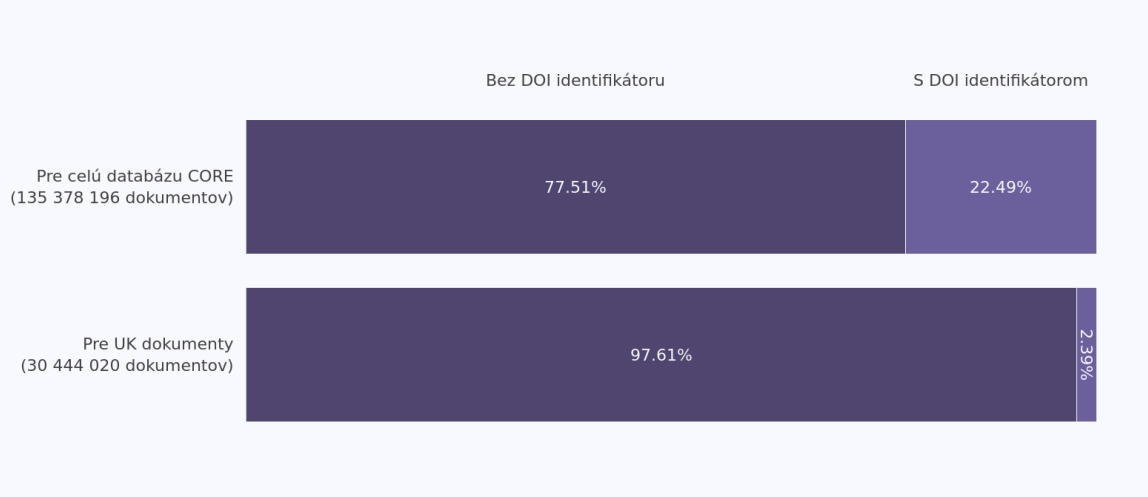
Kapitola 5

Štatistiky získaných dát

Pre portál nebolo potrebné len veľké množstvo získaných dát, ale aj ich presnosť a aktuálnosť. V tejto kapitole sú predstavené štatistiky jednotlivých metrík.

5.1 Štatistky pre dáta z CORE

Celkovo sa podarilo z britských repozitárov stiahnúť 728 728 dokumentov, ktoré obsahovali DOI identifikátor. Celkový počet dokumentov, ktoré boli na britských univerzitách publikované, bol 30 444 020 (29 715 292 bez DOI identifikátoru). Môže sa zdať, že pomer publikácií s DOI a bez DOI identifikátora je veľmi veľký. Je nutné podotknúť, že niektoré repozitáre obsahujú aj veľké množstvo nezaujímavých dokumentov (napríklad prezentácie, ktoré neboli publikované pod DOI identifikátorom). Pre zaujímavosť – projekt CORE obsahoval 135 378 196 dokumentov (30 444 020 s DOI a 104 934 176 bez DOI identifikátora). Štatistiky je možné vidieť v grafe č. 5.1.

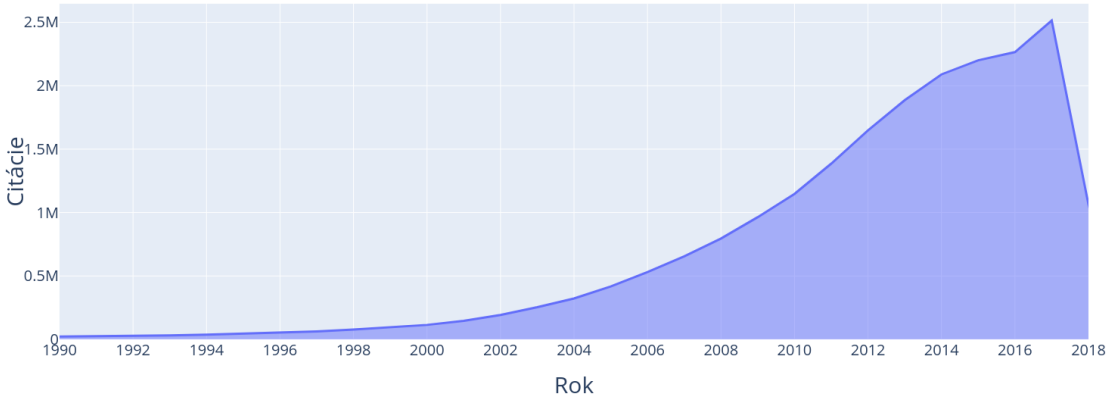


Obr. 5.1: Štatistiky získaných dát z projektu CORE ku dňu 12. 07. 2018.

5.2 Štatistiky z MAG

Pre 962 259 dokumentov s DOI identifikátorom získaných z CORE databázy sa mi podarilo získať 698 260 MAG záznamov, z toho 590 179 záznamov malo počet citácií väčších ako

nula, čo bolo viac ako 84%. V grafe č. 5.2 je znázornený počet citácií v rámci jednotlivých rokov v rozpätí 1990–2018. V priebehu rokov je možné vidieť rastúci počet citácií, čo je ovplyvnené zvyšujúcim sa počtom digitálnych dokumentov. Práce sa čoraz viac publikujú v digitálnej forme. Zaujímavosťou je, že prvá citácia v MAG grafe pre získané dokumenty z projektu CORE bola z roku 1898 a išlo o dokument s názvom *LXXXVIII. - Contributions from the Laboratories of the Heriot Watt College, Edinburgh. Note on the formation of anthraquinone from orthobenzoylbenzoic acid*[1]. V grafe č. 5.2 je možné vidieť v roku 2018 veľký pokles citácií. Príčinou bol fakt, že dáta boli získané z internej kópie databázy MAG, ktorá mi bola poskytnutá v rámci Erasmus stáže absolvovanej na *The Open University*. Dáta v internej kópii sa synchronizovali s databázou MAG v pravidelných, niekoľkomesačných intervaloch a ja som pracoval s verziou databázy z marca 2018.

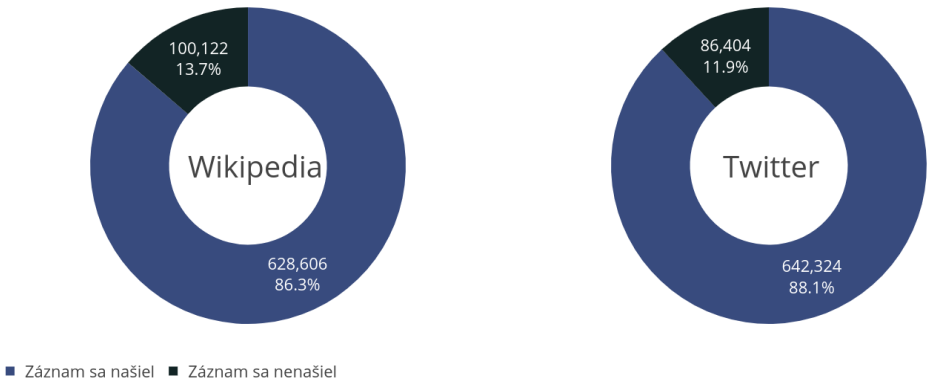


Obr. 5.2: Počet citácií UK dokumentov v rokoch 1990–2018 získaných z MAG grafu ku dňu 15. 07. 2018.

5.3 Štatistiky pre dáta z Crossref Event Data

Pre 728 728 dokumentov s DOI identifikátorom sa mi z projektu *Crossref Event Data* podarilo stiahnúť 628 606 dokumentov (86,3 %), ktoré obsahovali záznam o referenciách na stránke *Wikipedia*. Podobne pre portál *Twitter* sa mi podarilo stiahnúť 651 324 dokumentov (88,1%). Štatistiky je možné vidieť aj v grafe č. 5.3.

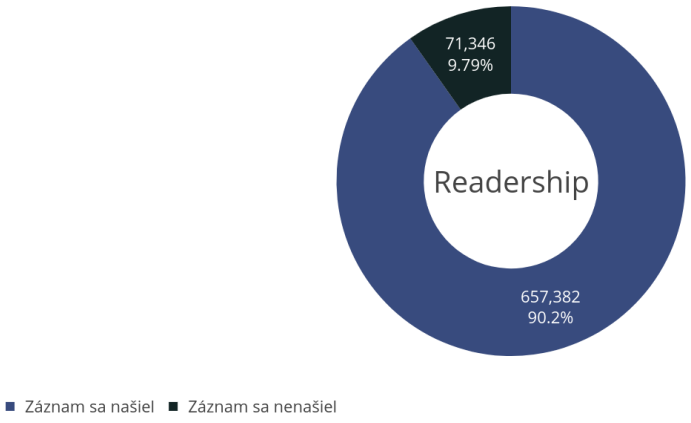
[1]Dokument *LXXXVIII. - Contributions from the Laboratories of the Heriot Watt College, Edinburgh. Note on the formation of anthraquinone from orthobenzoylbenzoic acid* je dostupný na adrese https://core.ac.uk/display/29178387



Obr. 5.3: Štatistiky získaných dát z projektu *Crossref Event Data* ku dňu 14. 07. 2018.

5.4 Štatistiky z Mendeley Readership

Posledným dátovým zdrojom pre portál boli *Readership* metriky z projektu *Mendeley*. Pre 728 728 dokumentov sa mi podarilo získať *Readership* metriku k 657 382 (90.2%) dokumentom. Grafické zobrazenie je možné vidieť v grafe č. 5.4.



Obr. 5.4: Štatistiky získaných dát z projektu *Mendeley* ku dňu 15. 07. 2018

Kapitola 6

Návrh portálu pre porovnávanie vedeckých výsledkov

V tejto kapitole som sa zameral na návrh samotného portálu. V podkapitolách boli predstavené jednotlivé podstránky portálu spolu s ich grafickým návrhom.

6.1 Časti portálu

Cieľom portálu bolo umožniť užívateľovi vytvárať vzorky dát a tieto vzorky následne ďalej analyzovať, čo značí vytvárať rôzne typy pokročilých grafov. Vytvorené analýzy mali byť prístupné na hlavnej stránke portálu, tzv. nástenke. Užívateľ mal mať možnosť pridávať, resp. odoberať jednotlivé grafy a takto si danú nástenku prispôsobovať podľa vlastných predstáv. Užívateľ mal mať priradenú jednu inštitúciu, ku ktorej patrí, resp. pre ktorú chce zobrazovať dáta (ide hlavne o dáta na hlavnom paneli).

6.2 Hlavný panel

Hlavný panel (*dashboard*) by mal byť zároveň hlavnou stránkou, ktorú užívateľ uvidí ako prvú ihneď po prihlásení. Táto stránka by sa mala skladať z dvoch častí.

- **Štatistiky inštitúcie**
- **Upravovateľná zóna**

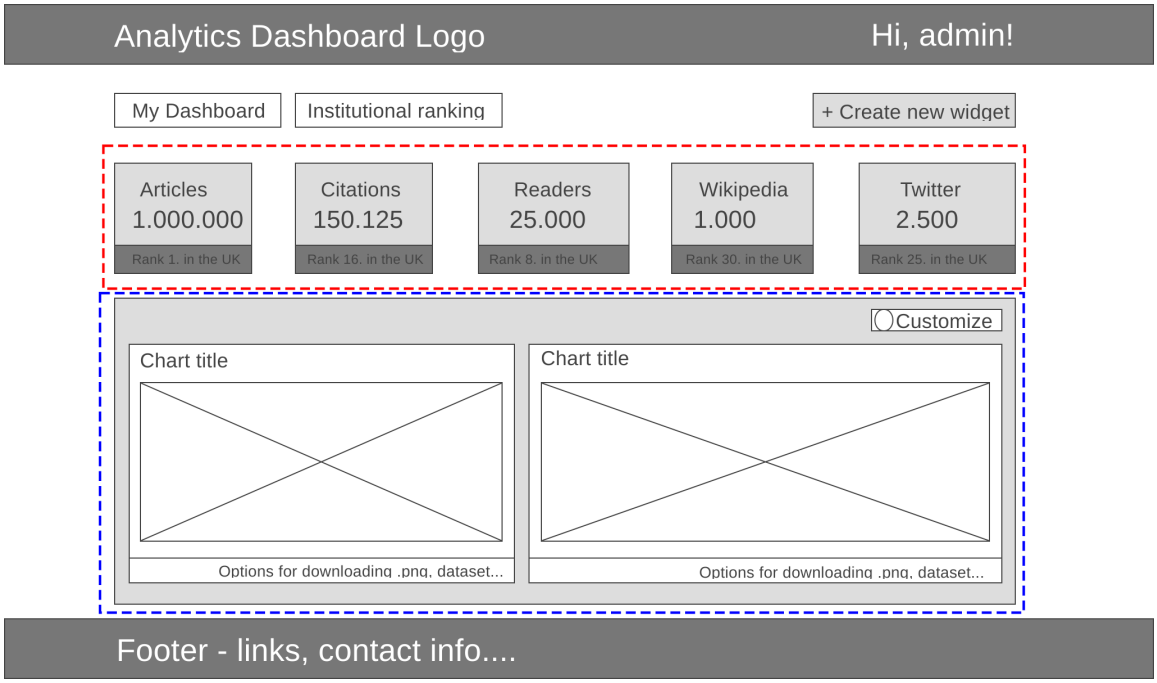
6.2.1 Štatistiky inštitúcie

V tejto časti by sa mali zobraziť globálne dáta o inštitúcii: počet dokumentov, pre ktoré existuje záznam v databáze, počet dokumentov, ktoré univerzita publikovala, počet citácií týchto dokumentov, počet čitateľov, počet referencií na stránke *Wikipedia* a počet spomenutí na sociálnej sieti *Twitter*. Okrem toho by sa mala v tejto časti zobrazovať aj ročná prognóza tam, kde to získané dáta dovolia. (viď sekciu 7.5.1 Výpočet ročnej projekcie). Na obrázku č. 6.1 je táto sekcia vyznačená červenou farbou.

6.2.2 Upravovateľná zóna

Túto časť by si mal každý užívateľ upravovať podľa seba. Na výber by mal mať základné preddefinované, ale aj vlastnoručne vytvorené grafy. Grafy by mali byť uložené v kniž-

nici grafov, ktorá sa zobrazí po kliknutí na tlačidlo `Show library`. Toto tlačidlo by sa malo užívateľovi zobraziť, ak by sa nachádzal v editačnom režime, ktorý indikuje prepínač `Customize` (na návrhu v obrázku č. 6.1 zobrazený v modrej zóne v pravom hornom rohu). Po otvorení knižnice by mal mať užívateľ možnosť vyhľadávať grafy a pridať ich na upravovateľný panel. Následne by mal mať možnosť na hlavnom paneli tieto grafy rozmiestňovať a zväčšovať alebo zmenšovať ich šírku a výšku. Každý graf by mal mať ešte aj dodatočnú možnosť stiahnutia v obrázkovom formáte *.png*, stiahnutia hlavných dát (vytvorených v prvom kroku tvorby grafu), ale aj dát, ktoré slúžia na vykreslenie samotného grafu (podmnožina dát z druhého kroku tvorby grafu). Na obrázku č. 6.1 je táto sekcia vyznačená modrou farbou.



Obr. 6.1: Návrh (*wireframe*) úvodnej stránky, resp. hlavného panelu.

6.3 Tvorba grafu

Do podstránky tvorba grafu by sa mal užívateľ dostať len z hlavného panelu, a to po kliknutí na tlačidlo `Create new widget`. Postup pri vytváraní jedného grafu by sa mal skladať z troch základných krokov:

1. **Definovanie vzorky dát** – vytvorenie dát na základe dotazu a filtrov.
2. **Voľba identifikátorov** – výber metrík, ktoré sa majú porovnávať a vyhodnocovať.
3. **Vizualizácia analýzy** – tvorba vizuálnej reprezentácie zvolenej vzorky.

6.3.1 Definovanie vzorky dát

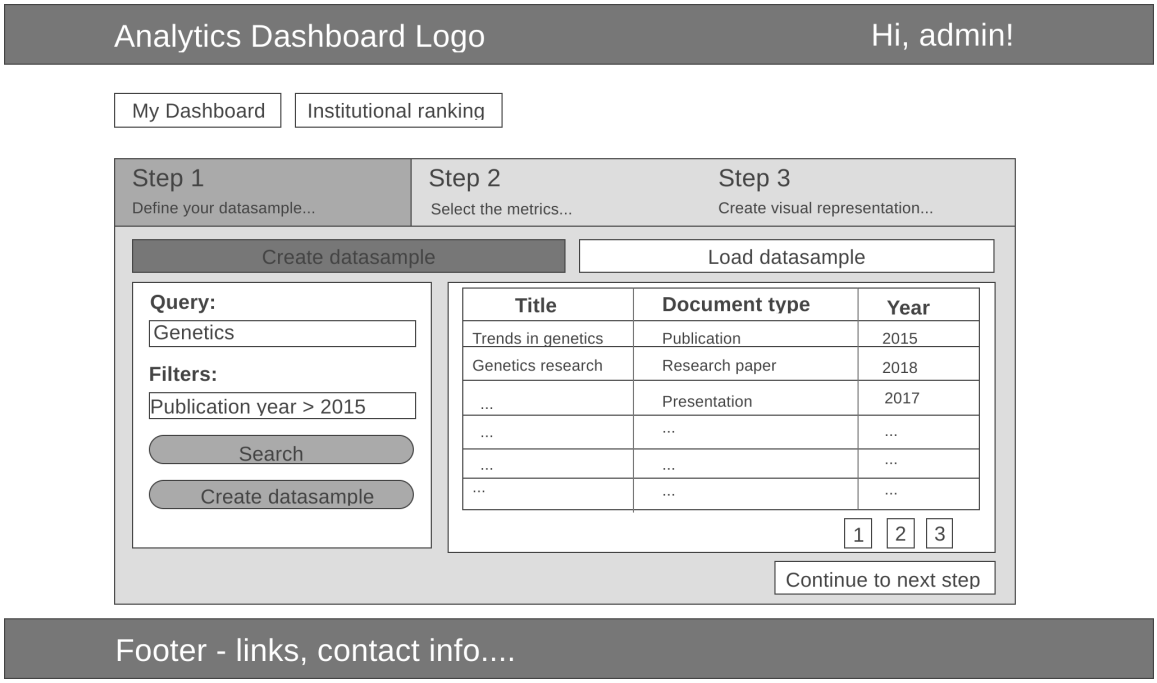
Definícia vzorky dát by mala byť prvým krokom pri tvorbe grafu. Užívateľ by mal mať možnosť zadefinovať a uložiť tie vzorky dát, s ktorými chce ďalej pracovať. Pod pojmom

vzorka dát sa rozumie podmnožina dát, ktorá má určitú spoločnú charakteristiku (napr. všetky dokumenty publikované v roku 2017 alebo všetky publikácie, ktoré majú 100 a viac citácií).

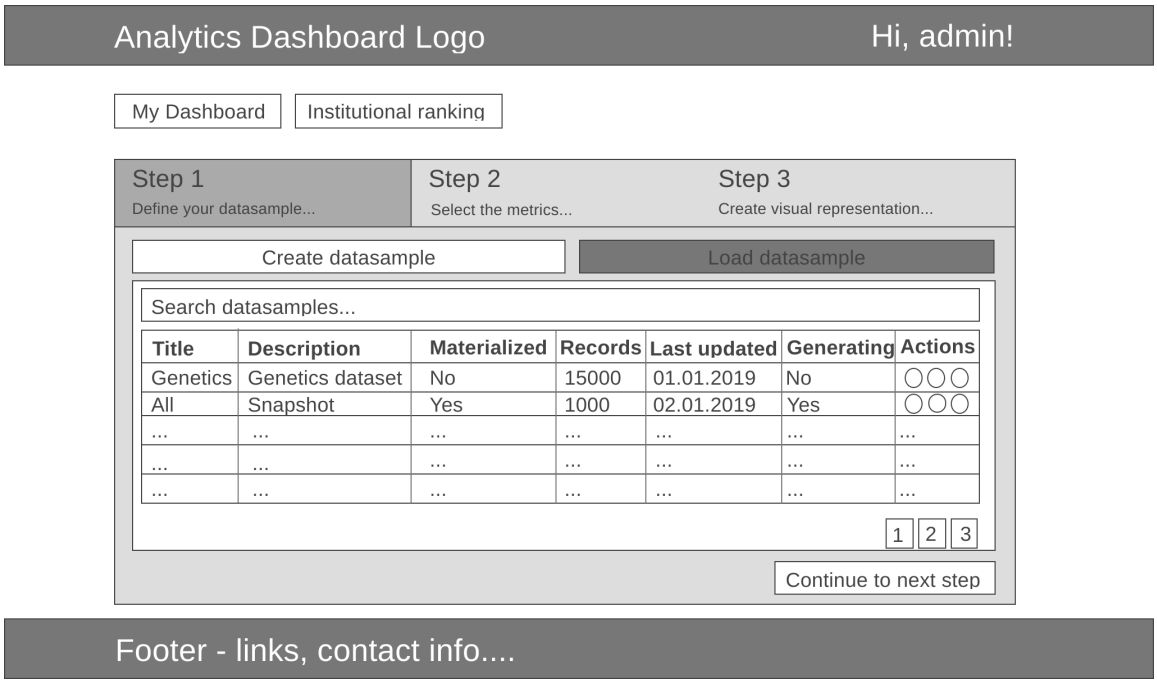
Tento krok by mal byť prepojený s databázou Elasticsearch projektu CORE. Tato databáza obsahuje všetky metadáta o prácach, napríklad: meno autora, rok publikovania, typ dokumentu (vedecká práca, prezentácia...), abstrakt a pri niektorých prác aj ich celý text. Posledné dve zmienené polia sú ideálnym kandidátom na textové vyhľadávanie, čo je aj primárna funkcia databázového systému ElasticSearch. Elasticsearch API umožňuje textové vyhľadávanie v reálnom čase. Výsledne dokumenty hodnotí podla internej ohodnocovacej funkcie a ponúka najrelevantnejšie výsledky, teda výsledky s najväčším skóre. Cieľom je vyplnenie formuláru, ktorý by sa na strane serveru pretransformoval na odpovedajúci ElasticSearch dotaz a ten sa zaslal na server projektu CORE. Spätne by sa obdržali všetky CORE identifikátory dokumentov, ktoré vyhovujú danému dotazu. Tieto čísla sa uložia do databázy portálu pod unikátnym identifikátorom – názvom vzorky. Je nutné podotknúť, že táto vzorka by mala byť dvojakého typu – aktualizovaná alebo zmrazená (z angl. *snapshot*). Pod pojmom aktualizovaná rozumiem vzorku, v ktorej by mali dáta byť vždy aktualizované. Naopak v zmrazenej vzorke by sa dáta nemenili a nebolo by možné ich už ďalej aktualizovať. Vzorky by malo byť taktiež možné exportovať v textovom formáte *.csv*. Návrh tejto podstránky je zobrazený na obrázku 6.2, resp. 6.3.

Prečo je však nutné vytvárať dva typy vzoriek? Najlepšie je to vysvetlené na nasledovnom príklade:

Pracovník univerzity chce zistiť, ako sa jeho univerzite, v porovnaní s inou univerzitou, darí vo výskume v oblasti genetiky. Jedným z ukazovateľov môže byť počet citácií, ktoré publikácie obsahujúce v nadpise alebo v abstrakte slovo *Genetika* získali. Pre túto potrebu si pracovník univerzity najskôr vytvorí vzorku dát. Vytvorí si dve rovnaké vzorky, jedna bude zmrazená a druhá bude aktualizovaná keď pribudnú nové dáta. Z každej vzorky si vytvorí rovnaký graf. Tieto grafy pripne na hlavný panel vedľa seba. Spočiatku grafy budú rovnaké, ale po pár mesiacoch by mohol graf, ktorý je vytvorený z aktualizovanej vzorky dát, zobrazovať iné hodnoty. Je to z toho dôvodu, že počet citácií ako aj iných metrík je získavaný v určitých intervaloch. Konkrétne citácie sú získané z internej kópie dátovej sady MAG, ktorá je synchronizovaná v pravidelných niekoľko mesačných intervaloch. Časom buď pribúdajú nové publikácie na tému genetika, alebo sa nájdu publikácie citujúce dokumenty zahrnuté vo vytvorených vzorkách. Tieto dáta by zmrazenú vzorku nemali ovplyvniť, ale aktualizovanú áno, čo sa automaticky odrazí na vytvorenom grafe.



Obr. 6.2: Návrh prvého kroku vytvorenia grafu – definovanie vzorky dát.



Obr. 6.3: Návrh prvého kroku vytvorenia grafu – načítanie už existujúcej vzorky dát.

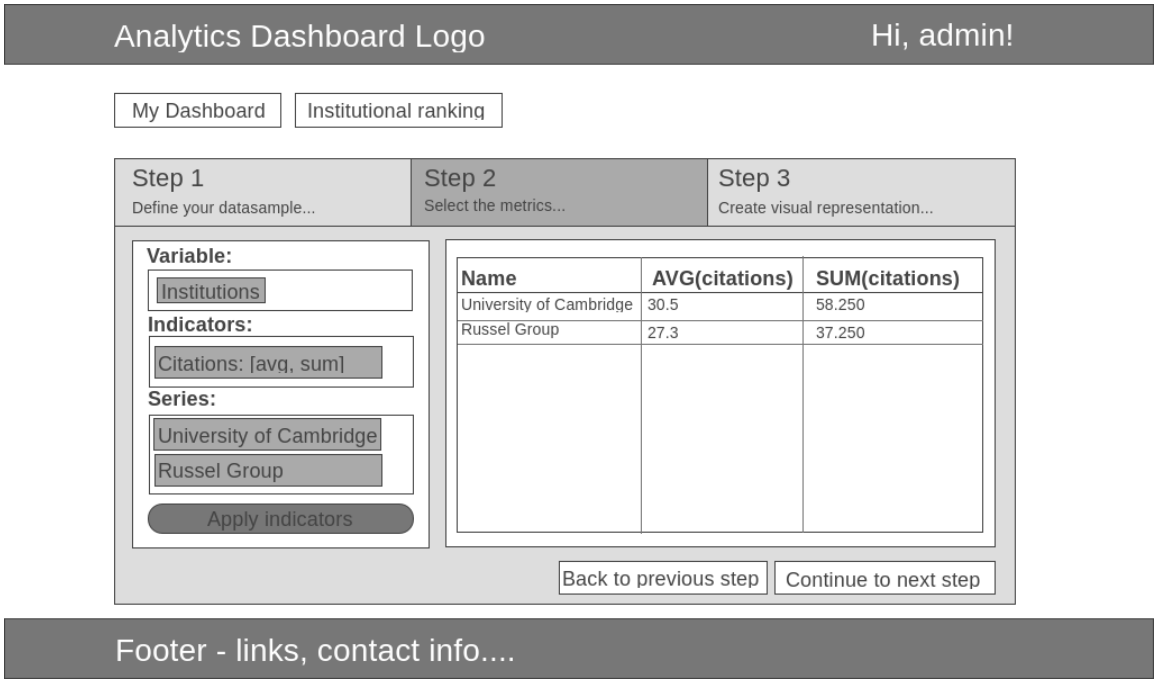
6.3.2 Voľba identifikátorov

Voľba identifikátorov reprezentuje druhý krok tvorby grafu. V tomto kroku by mal mať užívateľ možnosť bližšie špecifikovať, ktoré dáta z danej vzorky chce použiť. Vzorka z prvého

kroku obsahuje všetky informácie o dokumentoch, t.j. počet citácií, sociálnych signálov, metriky *Mendeley readership* a názov inštitúcie, ktorá daný dokument publikovala. Predpokladá sa, že užívateľ v grafe nechce zobraziť všetky tieto údaje, pretože graf by sa stal zbytočne neprehľadný. Z tohto dôvodu by tu mal byť formulár, ktorý zjednoduchší vzorku. Formulár by mal obsahovať tri polia:

- V prvom poli by mala byť možnosť výberu premennej (čas, alebo inštitúcia). Táto informácia indikuje to, čo sa bude nachádzať na x-ovej osi grafu.
- V druhom poli by mala byť možnosť výberu indikátorov, tento údaj reprezentuje hodnoty na y-ovej osy grafu. Ak by si užívateľ vybral premennú čas, tak ako indikátor by sa mu mali ponúknuť iba citácie (žiadne iné metriky neobsahujú časové dáta). V prípade výberu inštitúcie by už mala byť ponuka indikátorov bohatšia – referencie na stránkach *Twitter* a *Wikipedia*, počet čitateľov na platforme *Mendeley* a celkový počet citácií z dátovej sady MAG.
- V treťom poli by si mal užívateľ možnosť vybrať série – ktoré univerzity, resp. skupiny univerzít sa majú zobraziť.

Rozvrhnutie formulára a celkový náhľad na túto podstránku je vyobrazený na obrázku č. 6.4.

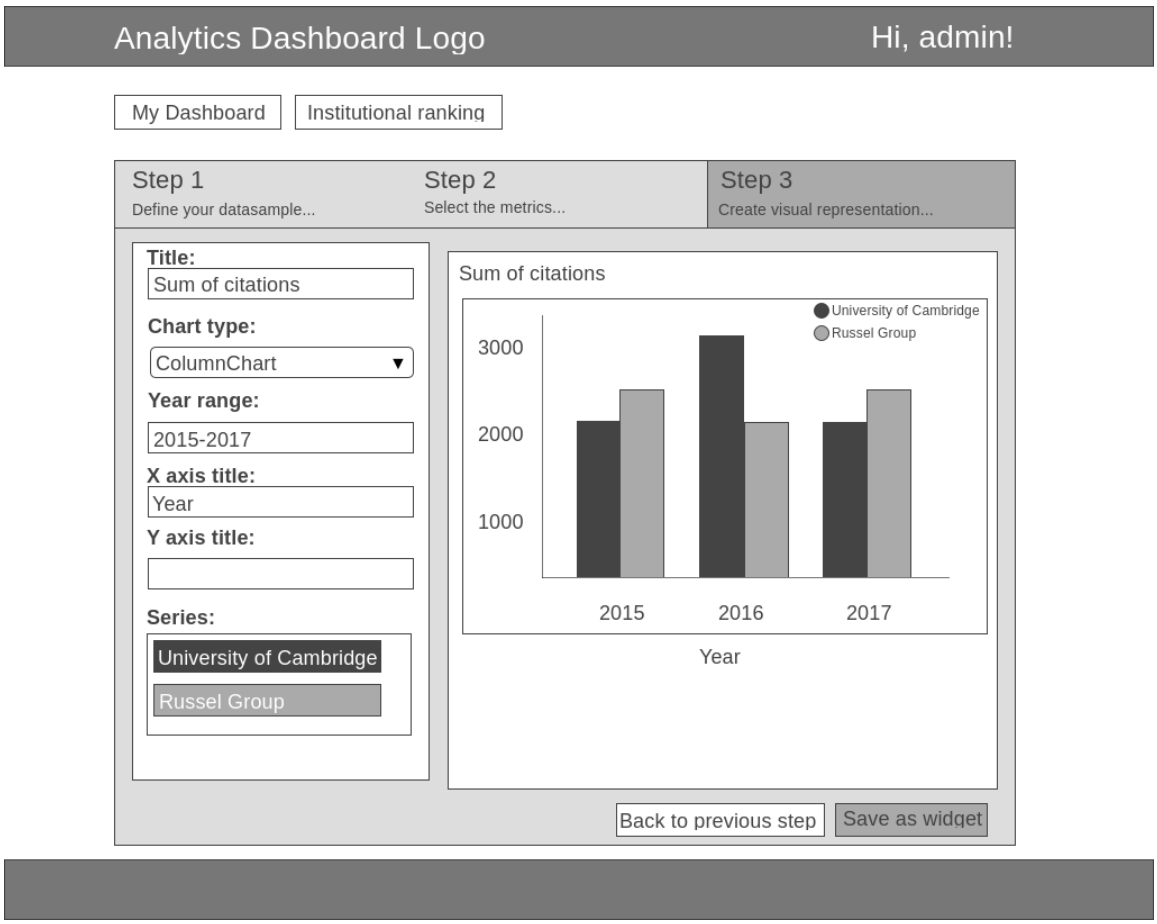


Obr. 6.4: Návrh druhého kroku vytvorenia grafu – voľba a aplikácia identifikátorov na dataset.

6.3.3 Vizualizácia analýzy

V poslednom, treťom kroku by mal mať užívateľ možnosť vybrať si jeden typ grafu (koláčový, stĺpcový, plošný, ...) a ďalej ho prispôsobovať (meniť popis jednotlivých osí, popis a farby jednotlivých sérií a poprípade niektoré série ešte dodatočne odstrániť). Vytvorený graf

by sa mal následne uložiť pod unikátnym názvom a zobraziť v knižnici grafov na hlavnej stránke.



Obr. 6.5: Návrh tretieho kroku vytvorenia grafu – vizualizácia analýzy.

6.4 Štatistiky univerzít

Na tejto podstránke by sa mali zobraziť všetky univerzity spolu so získanými informáciami. Malo by ísť o jednu veľkú tabuľku, v ktorej by sa malo dať vyhľadávať a radiť podľa určitých metrík. Takýmto spôsobom by mal užívateľ možnosť nájsť, ktorá univerzita má najviac citovaných prác, alebo ktorá má najviac sociálnych signálov. Užívateľ by mal mať aj možnosť pozrieť, ktoré univerzity sa nachádzajú pred alebo za ním sledovanou univerzitou.

6.5 Profil používateľa a dokumentácia

Každý používateľ by mal mať možnosť sa dostať na podstránku, na ktorej by si mohol upravovať základné informácie o sebe a meniť heslo. Mal by mať aj možnosť dostať sa na podstránku dokumentácie, kde by boli detailne popísané jednotlivé podstránky a sekcie. Nemal by tu chýbať aj obrázkový návod na vytvorenie grafu.

6.6 Administrátorský panel

Administrátorský panel by mal byť dostupný len pre užívateľov s právami administrátora. Tento panel by mal obsahovať tabuľku, v ktorej sa zobrazia všetci registrovaní užívatelia. V jednom stĺpci tabuľky by mali byť možnosti úprav, ako napríklad dočasné zakázanie prihlásenia užívateľovi alebo odstránenie užívateľa. Na tejto podstránke by mala byť tiež možnosť pridania užívateľa a možnosť zobrazenia štatistík portálu, t.j. koľko dokumentov sa nachádza v databáze, koľko užívateľov je registrovaných, koľko je vytvorených grafov a podobne.

Kapitola 7

Tvorba portálu pre porovnávanie vedeckých výsledkov

Táto kapitola je zameraná na implementáciu návrhu portálu z predchádzajúcej kapitoly (6 Návrh portálu pre porovnávanie vedeckých výsledkov). Okrem iného je tu detailne predstavený aj historický vývoj databázových systémov a výber vhodného databázového systému pre portál.

7.1 Voľba programovacieho jazyka a knižníc

Na návrh serverovej (*backend*) časti projektu som sa rozhodol použiť programovací jazyk PHP vo verzii 7.0. V tomto programovacom jazyku je napísaný framework Symfony[1], ktorý mi výrazne uľahčil prácu. Vďaka ďalším podporným knižniciam bola možná jednoduchá práca s databázou, registrácia a prihlásenie užívateľov a prinieslo to aj mnohé ďalšie výhody.

Na klientskú (*frontend*) časť bola použitá javascriptová knižnica JQuery vo verzii 3.3.1[2], na grafickú časť (CSS/HTML) bol použitý framework Bootstrap vo verzii 4.0[3]. Vykresľovanie grafov bolo riešené javascriptovou knižnicou Google Charts[4].

7.2 Voľba databázového systému

Portál mal spĺňať definíciu OLAP systému. Preto bol výber databázového systému (DBMS – *Database Management System*, v literatúre známy aj pod pojmom Systém riadenia bázy dát (SRBD)) kľúčový a venoval som mu výraznú pozornosť. Hlavným kritériom pri výbere bola rýchlosť zodpovedania komplexných agregačným dotazov. Ďalším kritériom bola dostupnosť knižnice pre programovací jazyk PHP a v neposlednom rade muselo ísť projekt otvoreného softwaru (z angl. *open source*), ktorý bolo možné nainštalovať lokálne a využívať bezplatne.

[1] Dokumentácia frameworku Symfony je dostupná na adrese https://symfony.com/doc/3.4/index.html#gsc.tab=0.

[2] Dokumentácia javascriptovej knižnice jQuery je dostupná na adrese https://api.jquery.com/.

[3] Dokumentácia k frameworku Bootstrap je dostupná na adrese https://getbootstrap.com/docs/4.0/getting-started/introduction/

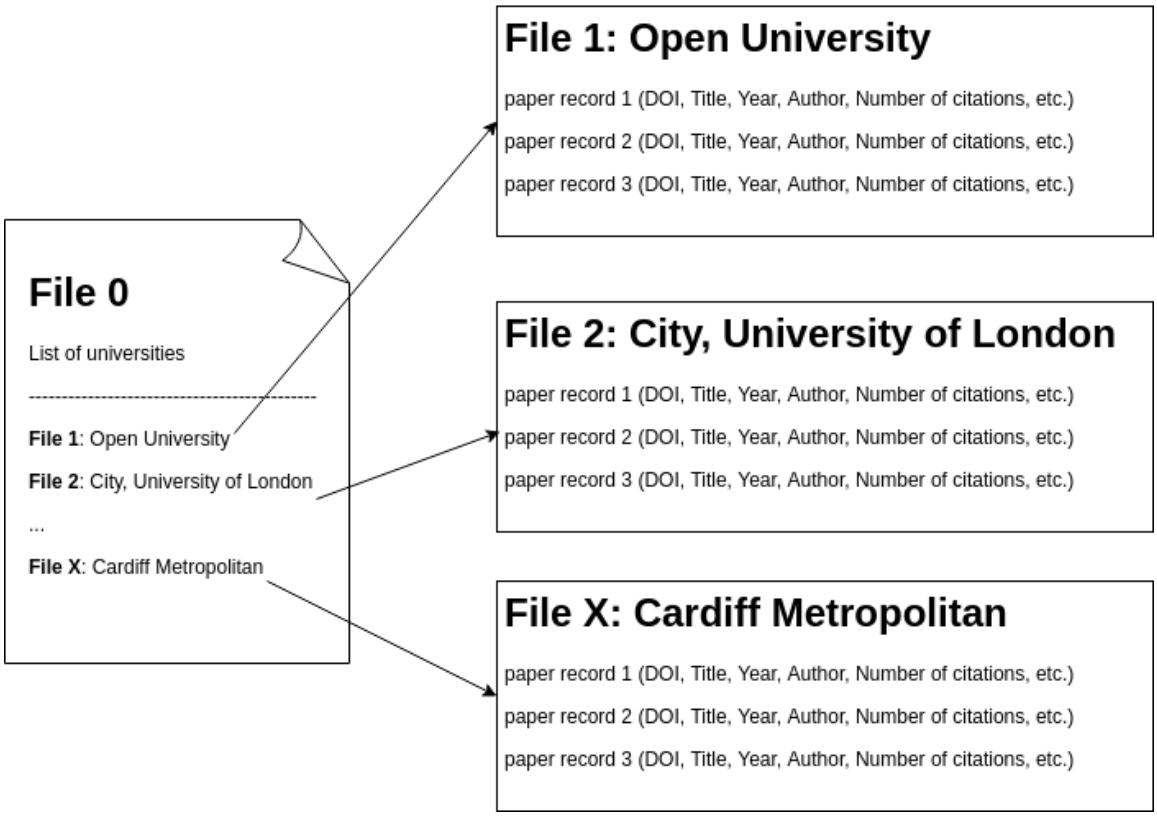
[4] Dokumentácia k Google Charts knižnici je dostupná na adrese https://developers.google.com/chart/.

7.2.1 Porovnanie databázových modelov

Databázový model určuje spôsob uloženia objektov do databázy a spôsob ich vzájomného prepojenia [23]. V tejto podkategórii sú rozobraté základné druhy modelov databáz a priblížené ich výhody a nevýhody. Modely sú zoradené chronologicky presne podľa ich časového vývoja.

Databáza otvorených súborov (z angl. Flat-file Database)

Otvorené súbory boli prvým spôsobom organizovania dát preferovaným v 50. rokoch 20. storočia. Ide o informácie oddelené zvyčajne oddelovačom a zapísané v sekvenčnom poradí. Indexovanie záznamov v týchto súboroch nie je prispôsobené rýchlemu vyhľadávaniu. Pri vyhľadávaní sa prechádza každý záznam sekvenčne až pokým sa nenarazí na ten hľadaný. Neefektívnosť tohto spôsobu uloženia dát rastie s ich množstvom. Ďalším problémom je poškodenie dát. Pokiaľ dôjde k neočakávanému vypnutiu aplikácie počas toho ako je súbor otvorený, nie je možné predikovať, čo sa s dátami stane (môžu sa napríklad stratiť). Ďalším problémom je konkurencia zápisu/čítania zo súborov [30]. Jednoduchý príklad ako by mohla vyzerať databáza portálu, ak by som využil tento model je znázornený na obrázku č. 7.1.

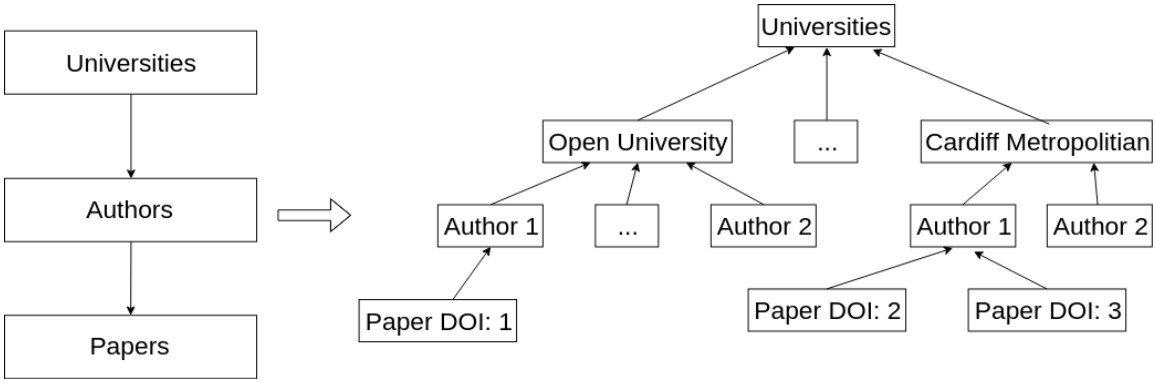


Obr. 7.1: Zjednodušený príklad uloženia dát v databázovom modeli otvorených súborov.

Hiererchický model (z angl. Hierarchical database model)

Hierarchický model bol prvýkrát predstavený v druhej polovici 60. rokov minulého storočia a vyvinul sa z prechádzajúceho modelu (pozn. modelu otvorených súborov). Názov je

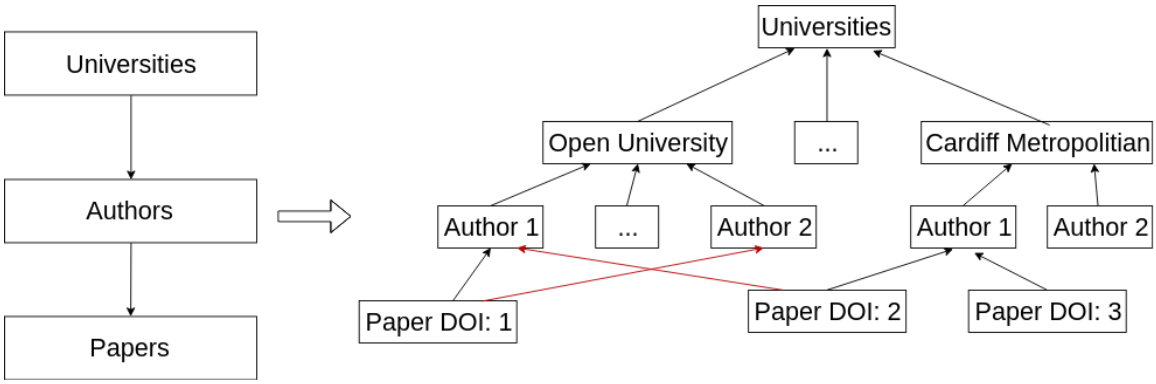
odvodený zo spôsobu uloženia dát. Každý zo súborov definovaných v systéme otvorených súborov je tu nahradený uzlom. Jednotlivé uzly sú prepojené ukazateľmi, ktoré definujú vzťah rodič-potomok (1:N - z angl. *one to many*). To znamená, že dáta sú organizované v stromovej štruktúre, pričom prvý uzol sa nazýva koreň. Vďaka tomu sa do určitej miery eliminovala redundancia spôsobená otvorenými súbormi. Hierarchický model má však svoje limitácie spôsobené tým, že uzol má vždy práve jedného rodiča [23]. To značí, že redundancia stále pretrváva. Ak má publikácia viac ako jedného autora a títo autori sú z rôznych univerzít, tak záznam musí byť duplikovaný. Na obrázku č. 7.2 je znázornený zjednodušený príklad uloženia dát v hierarchickom modeli databáz.



Obr. 7.2: Zjednodušený príklad uloženia dát v hierarchickom modeli databáz.

Sieťový model (z angl. Network model)

Sieťový model bol prvýkrát predstavený v roku 1971 a kombinoval individuálne hierarchické modely databáz do jednej veľkej. Záznamy sú poprepájané medzi sebou a vytvárajú akúsi sieť. V určitom zmysle je tento model veľmi podobný predchádzajúcemu, avšak prináša viaceré vylepšenia. Sieťový model môže mať viac ako jednu stromovú štruktúru a podporuje väzby typu M:N (z angl. *many to many*). Vyhľadávanie môže začať v ľubovoľnom zázname, nie nutne v koreni ako to bolo v hierachickom modeli. Napriek tomu, že sieťový model priniesol veľké množstvo vylepšení, priniesol aj problém údržby a implementácie [30]. Spomínané negatíva spôsobili, že sa tento model v praxi veľmi nepoužíva. Pretransformovaný hierarchický model na sieťový je možné vidieť na obrázku č. 7.3.



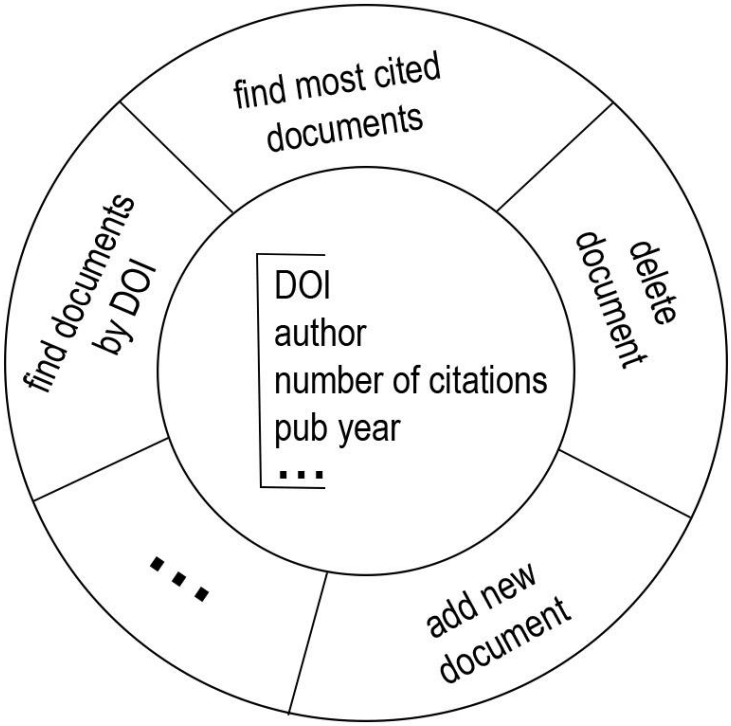
Obr. 7.3: Hierarchický model z príkladu 7.2 prepísaný do sieťového modelu databázy.

Relačný model (z angl. Relation model)

Relačný model mal vyriešiť problémy zmienených sieťových a hierarchických modelov. Ich spoločným nedostatkom bola aj nedostatočná flexibilita. Model je postavený na teórii z publikácie matematika Edwarda Franka Codda z *IBM Reasearch Laboratory* z roku 1970 [7]. Ten navrhoval ukladanie dát v dvojrozmerných tabuľkách, medzi ktorými vytváral relácie. Tabuľky v relačnom modeli databázy sú zložené z riadkov predstavujúcich záznamy a zo stĺpcov obsahujúcich dáta určitej položky. Každý záznam v tabuľke musel mať unikátny kľúč zložený z jedného alebo viacerých políčok záynamu. Tým sa zaručila jednoduchosť vyhľadávania a eliminovala redundancia. Vzhľadom na jednoduchost a zrozumiteľnosť bol tento model odbornou verejnosťou prijatý. Na základe opísaných výhod som sa rozhodol tento model použiť aj v mojej práci. Návrh databázy je zobrazený na obrázku č. 7.5.

Objektovo-orientovaný model (z angl. Object-Oriented model)

Objektovo-orientovaný model vznikol z dôvodu absencie podpory obrázkov, videí, zvukov a textových súborov v relačnom modeli dát. Pojem *objektovo-orientovaný model* bol prvýkrát použitý v roku 1985 a názov napovedá, že v tomto type databázy sa pracuje s objektami [2] [12]. Pod pojmom objekt sa rozumie zoskupenie príbuzných dát a programovej logiky, ktoré spoločne reprezentujú určitú vec alebo osobu z reálneho sveta [23]. Položky v tomto objekte sa nazývajú premenné. V objektovom-orientovanom modeli môže užívateľ pristupovať k premenným iba pomocou metód, ktoré sú na to vytvorené. Tým sa docieli zapúzdrenie (z angl. *encapsulation*) objektu. Môže sa zdať, že spôsob ukladania dát je veľmi neefektívny a uložené dáta môžu byť redundandné, avšak je tu aplikovateľný objektovo-orientovaný návrh. Každý objekt má mať len potrebné dáta a objekty sa majú zoskupovať do spoločných celkov a tým pádom sa dá aplikovať dedičnosť. Tento model sa často používa v CAD (z angl. *computer-aided design*) systémoch [23]. Zjednodušený návrh uloženia dát v objektovo-orientovanom modeli je zobrazený na obrázku č. 7.4.



Obr. 7.4: Zjednodušená ukážka uloženia dát v objektovo-orientovanom modeli.

Objektovo-relačný model (z angl. Object-Relation model)

Objektovo-relačný model dopĺňa objektové funkcie do relačného modelu dát. Prvá databáza tohto typu bola predstavená v júni 1997 spoločnosťou *Oracle* a niesla názov *Oracle8* [30]. Hlavným dôvodom pre vytvorenie objektovo-relačného modelu bola absencia jednorázových (z angl. *ad hoc*) dotazov v predchádzajúcom objektovo-orientovanom modeli.

Z prehľadu je zrejmé, že pre portál bol najvhodnejším kandidátom relačný model alebo objektovo-relačný model, ktoré spolu úzko súvisia.

7.2.2 Porovnanie spôsobu uloženia dát v relačných databázových systémoch

V relačných databázových systémoch sa dajú dáta ukladať po riadkoch alebo stĺpcoch, avšak najrozšírenejším spôsobom ukladania dát je ich ukladanie po riadkoch. Takýto spôsob je vhodný v databázach očakávajúcich veľké množstvo krátkych on-line transakcií – `INSERT`, `UPDATE` a `DELETE` (pozn. tieto systémy sa nazývajú OLTP – *Online Transaction Processing*). Operácie sú vďaka tomu rýchle a efektívne.

Spôsob ukladania dát po stĺpcoch je vhodný obzvlášť v databázach pre analytické účely, v ktorých sa predpokladá práca s menším množstvom stĺpcov, pričom zvyšok tabuľky je ignorovaný (pozn. využívané hlavne v OLAP systémoch). V stĺpcových databázach sú rýchlejšie agregačné dotazy. Spájanie (`JOIN`) viacerých tabuliek je v tomto spôsobe uloženia dát omnoho rýchlešie. Výhodou je aj rýchle pridávanie nových hodnôt pre všetky záznamy v jednom stĺpci. Tento typ databáz zaberá na disku menej miesta a kompresia dát je omnoho jednoduchšia. Na druhú stranu tu však existujú aj určité limitácie. Pridávanie a aktualizá-

cia nových záznamov sú podstatne pomalšie. Ďalšou nevýhodou je aj horšia škálovateľnosť [4].

Portál mal spĺňať prvky OLAP systému a dotazy na databázu boli hlavne agregačné, preto som v portáli použil stĺpcovo orientovaný databázový model.

7.2.3 Porovnanie stĺpcových relačných databáz

Poznáme veľké množstvo stĺpcových relačných databáz a porovnanie všetkých by presahovalo rozsah tejto práce. Rozhodoval som sa porovnať nasledovné stĺpcové relačné databázy: MariaDB ColumnStore[5], ClickHouse[6], MonetDB[7] a Apache Spark[8].

Prvý projekt, ktorý som vyradil bol projekt MonetDB. Bolo to z dôvodu nízkej komunity a z môjho pohľadu neprepracovanej dokumentácie. Existuje už viacero vypracovaných porovnaní, a preto nemalo zmysel tvoriť ďalšie. V marci 2017 uskutočnil Alexander Rubin nezávislé porovnania už zmienených troch databázových systémov – MariaDB ColumnStore, Clickhouse a Apache Spark. V článku s názvom *MariaDB ColumnStore vs. Clickhouse vs. Apache Spark* vzišiel ako víťaz databázový systém Yandex Clickhouse. V jeho testoch dosiahol viac ako 10 krát rýchlejšie dotazy a vďaka kompresii zaberali dáta v porovnaní s ostatnými testovanými databázmi na disku menej miesta [27].

V ďalšom porovnaní z januára 2019, ktoré uskutočnil Mark Litwintschik na vzorke 1.1 miliardy jázd taxíkom, sa Yandex ClickHouse umiestnil pred Apache Spark s viac ako 100 násobne rýchlejšou odpoveďou na dotaz [22]. Dátová sada obsahovala 51 stĺpcov a jeho celková veľkosť bola 500 GB. Autor porovnával viacero enginov, medzi ktorými boli aj Yandex Clickhouse a Apache Spark. Pred databázovým systémom ClikHouse sa umiestnili už len platené databázy alebo databázy, ktoré využívali GPU [22].

7.2.4 Konečný výber databázy

Rozhodol som sa použiť dva databázové systémy. Jeden v sebe uchovával všetky informácie o užívateľoch a nimi vytvorených grafoch a druhý uchovával všetky dáta popísane v kapitole 3 Zdroje dát.

Pre uloženie informácií o užívateľovi som použil klasickú relačnú databázu MariaDB vo verzii 10.0.

Pre uloženie metrík som využil databázu Yandex ClickHouse vo verzii 1.1.54388, ktorá bola jednoznačne najvhodnejšia. Nebola to však jediná možnosť, do úvahy pripadali aj iné alternatívy. Do úvahy treba brať aj aktuálnosť porovnaní a uvedomiť si, že niektoré merania môžu byť nepresné, Každý databázový systém má svoje výhody, ale aj nevýhody. Pri databáze ClickHouse je nevýhodou napríklad absencia klasickej operácie `DELETE` [16]. Vymazanie záznamov nie je možné. Dôvodom je, že databázový systém ClickHouse je orientovaný na rýchlosť a táto operácia je náročná na čas. Toto môže byť v novších verziách zmenené. Pre mňa táto skutočnosť nepredstavovala problém, pretože dáta v databáze sa neplánovali mazať[9].

[5] MariaDB ColumnStore vychádza z InfiniDB. Adresa projektu – https://mariadb.com/kb/en/library/mariadb-columnstore/.

[6] Adresa projektu Yandex ClickHouse – https://clickhouse.yandex/.

[7] Adresa projektu MonetDB – https://www.monetdb.org/Home.

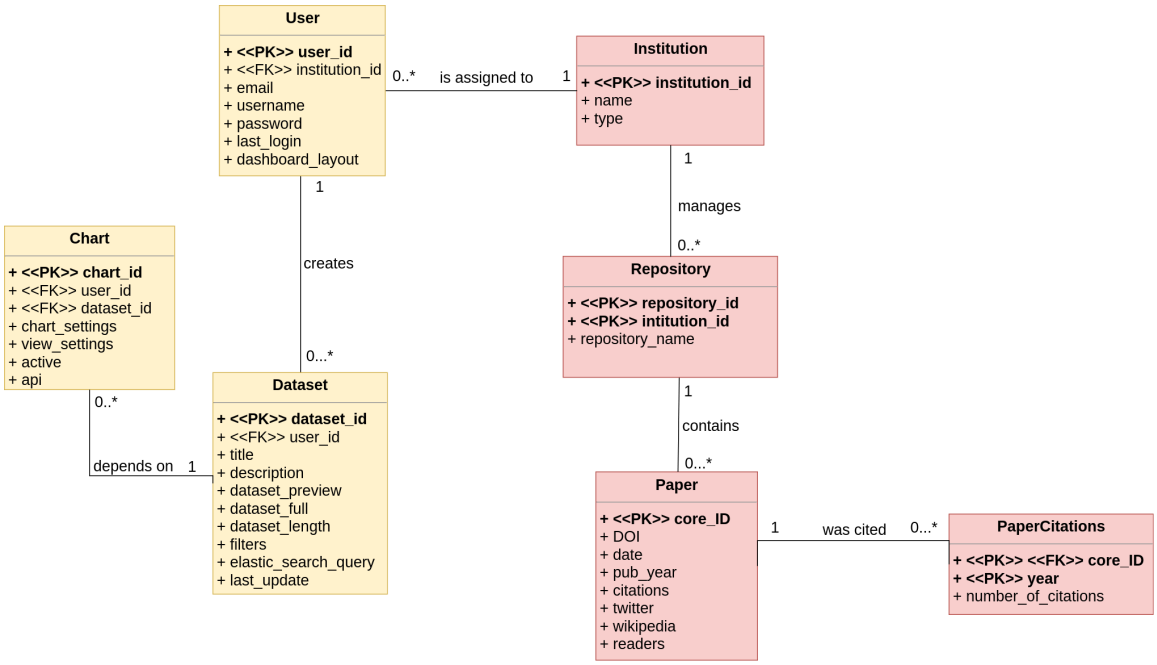
[8] Adresa projektu Apache Spark – https://spark.apache.org/.

[9] Je nutné však podotknúť, že dáta by sa mali v určitých časových intervaloch aktualizovať. Toto som vyriešil tak, že sa novo získané dáta vždy naimportujú do novej tabuľky a tá stará tabuľka sa odpojí a zmaže.

7.3 Návrh databázy - Entity Relationship Diagram

ER (z angl. *Entity Relationship*) je spôsob modelovania, pomocou ktorého sa dájú grafickou formou reprezentovať dáta uložené v databáze. V ER diagrame rozlišujeme entitu, entitnú množinu, atribút, vzťah a vzťahovú množinu [19].

Na obrázku č. 7.5 je namodelovaná databáza portálu. Žltou farbou sú vyznačené dáta, ktoré sú uložené v databáze MariaDB, t.j. užívateľské dáta a červenou farbou sú zobrazené dáta, ktoré sú uložené v databáze ClickHouse, t.j. dáta o publikáciách.



Obr. 7.5: Entity-Relationship diagram portálu.

7.4 Registrácia, prihlasovanie a profil užívateľa

Registráciu a prihlasovanie užívateľa som vyriešil pomocou rozšírenia do frameworku Symfony s názvom FOS User Bundle vo verzii 2.1[10]. Portál rozlišuje momentálne dva typy užívateľov. Prvý typ užívateľa je bežný užívateľ, ktorý ma prístup k tvorbe grafov a k štatistikám. Druhý typ užívateľa je administrátor, ktorý má okrem práv bežného užívateľa aj práva na vytváranie nových užívateľov a prístup ku štatistikám všetkých univerzít.

Administrátorský účet som vytvoril pomocou príkazu cez príkazový riadok (viac v prílohe B.4 Vytvorenie admin užívateľa). Po prihlásení má administrátor možnosť vytvoriť ďalšie profily. Ku každému profilu môže priradiť práve jednu univerzitu. Podľa priradenej univerzity sa užívateľovi na hlavnej stránke zobrazia štatistiky.

Každý užívateľ má prístup na podstránku, kde si môže zmeniť email a heslo k svojmu účtu a zároveň tam vidí informáciu aká univerzita bola k nemu priradená. Ak užívateľ zabudol heslo, má možnosť si ho resetovať na prihlasovacej stránke. Po kliknutí na `Forgot password?` sa užívateľovi zobrazí formulár, kde vyplní email alebo meno a po odoslaní mu

[10] Dokumentácia k rozšíreniu FOS User Bundle je dostupná na adrese – https://symfony.com/doc/master/bundles/FOSUserBundle/index.html

príde email s potrebnými inštrukciami na zmenu hesla. Implementované bolo aj automatické odhlásenie užívateľa po neaktivite viac ako 30 minút.

7.4.1 Špeciálny typ užívateľa – „default“

Špeciálnym typom užívateľa je užívateľ s prihlasovacím menom `default`, ktorý zároveň slúži ako šablóna pre všetkých ostatných nových užívateľov. Vytvorenie tohto užívateľa by malo nasledovať hneď po vytvorení administrátorského účtu. Všetky grafy a vzorky dát, ktoré sa zobrazia novému užívateľovi sú skopírované od užívateľa `default`. Vďaka tomu môže nový užívateľ vidieť plný potencial portálu hneď po prvom prihlásení.

Pri kopírovaní grafov od užívateľa `default` k novému užívateľovi sa reťazec `{name}` nahradí s názvom univerzity nového používateľa. Bližšie informácie o tom ako funguje šablónovanie sú dostupné aj na hlavnom paneli užívateľa `default`. Šablónovanie je možné vysvetliť na jednoduchom príklade:

Pre vytvorenie porovnania s názvom *Užívateľova univerzita vs. Russel Group v celkovom počte citácií* by sa mala pod účtom používateľa `default` vytvoriť vzorka dát pre všetky dokumenty. V druhom kroku by sa mali zvoliť dve premenné *Russel Group* a univerzita, ktorá je priradená užívateľovi `default`[11]. V treťom kroku pri uložení grafu by sa do názvu malo vyplniť *{name} vs. Russel Group v celkovom počte citácií* a všetky legendy, v ktorých je spomenutý názov univerzity `default`, napr. `(sum(University of Aberdeen - citations)` sa premenuje na `sum({name} - citations))`.

7.5 Štatistický panel

Štatistický panel je zobrazený na hlavnej stránke. Nachádza sa tam 5 malých okien, v ktorých sú zobrazené informácie o užívateľovej univerzite. Ďalej je v nich zobrazený počet publikácií, počet citácií, počet čitateľov na platforme *Mendeley*, počet *Wikipedia* referencií a počet tweetov na sociálnej sieti *Twitter*. Prvé dve spomenuté metriky obsahujú aj informáciu o ročnej projekcii (viac v 7.5.1 Výpočet ročnej projekcie). Každé okno nesie aj informáciu o umiestnení, t.j. číselna hodnota, ktorá symbolizuje koľko univerzít sa nachádza pred užívateľovou univerzitou. Štatistický bar je možné vidieť na obr. č. 7.6, kde je vyznačený červenou farbou.

7.5.1 Výpočet ročnej projekcie

Pri počte citácií a počte publikácií je možné vypočítať aj hodnotu ročnej projekcie, pretože k týmto údajom je dostupné aj časové obdobie. Získaný údaj symbolizuje ako sa darí univerzite v publikačnej činnosti v porovnaní s predchádzajúcim rokom a vypočíta sa podľa nasledovného vzorca:

[11] Nezáleží aká univerzita je priradená užívateľovi. Dôležité je však uvedomiť si, že sa nemôže porovnávať univerzita užívateľa `default` s univerzitou novo vytvoreného používateľa. Z tohto dôvodu je dobré používať v šablónach všeobecnejšie porovnania alebo použiť menej známu univerzitu, alebo v najlepšom prípade vytvoriť nový záznam pre fiktívnu univerzitu.

Pre sum_of_prev_year $\neq 0$:

$$annual\_year\_projection = \left(\frac{\frac{sum\_of\_current\_year * 12}{number\_of\_month}}{sum\_of\_prev\_year} - 1 \right) * 100$$

Pre sum_of_prev_year $= 0$:

$$annual\_year\_projection = \left(\frac{\frac{sum\_of\_current\_year * 12}{number\_of\_month}}{-100} - 1 \right) * 100$$

Pričom platí:

- **sum_of_current_year** predstavuje celkový počet dát získaných za sučasný rok
- **number_of_month** reprezentuje číslo aktuálneho kalendárneho mesiaca
- **sum_of_prev_year** predstavuje celkový počet dát získaných za minulý rok

Celkovým počtom dát sa v tomto prípade myslí počet citácií alebo počet publikácií.

7.5.2 Materializovaný pohľad na umiestnenia univerzít

Vypočet ročnej projekcie ako aj umiestnenia univerzít v daných kategóriach bol zdĺhavý. Spôsobovalo to dlhšiu dobu načítavania stránky a z toho dôvodu bolo potrebné tento proces zrýchliť. Databázový systém ClickHouse ako aj iné databázové systémy ponúkajú možnosť vytvárania pohľadov – „virtuálnych tabuliek“. Presnejšie ide o objekt v databáze, ktorý je definovaný SQL dotazom. Na tento objekt sa dá dotazovať rovnako ako na iné tabuľky. Poznáme databázový pohľad a materializovaný databázový pohľad. Materializovaný pohľad sa odlišuje od obyčajného pohľadu tým, že si uchováva výsledok SQL dotazu.

Rozhodol som sa použiť materializovaný pohľad a predpočítať si v ňom spomínané metriky. Pri každom importe dát je nutné ho zmazať a znova vytvoriť. Databáza ClickHouse neumožňuje aktualizovať materializovaný pohľad. Nie je to však podstatné, pretože sa dáta neplánujú meniť v tak častých intervaloch. Proces získavania umiestnenia sa výrazne urýchlil (z niekoľko sekúnd na milisekundy).

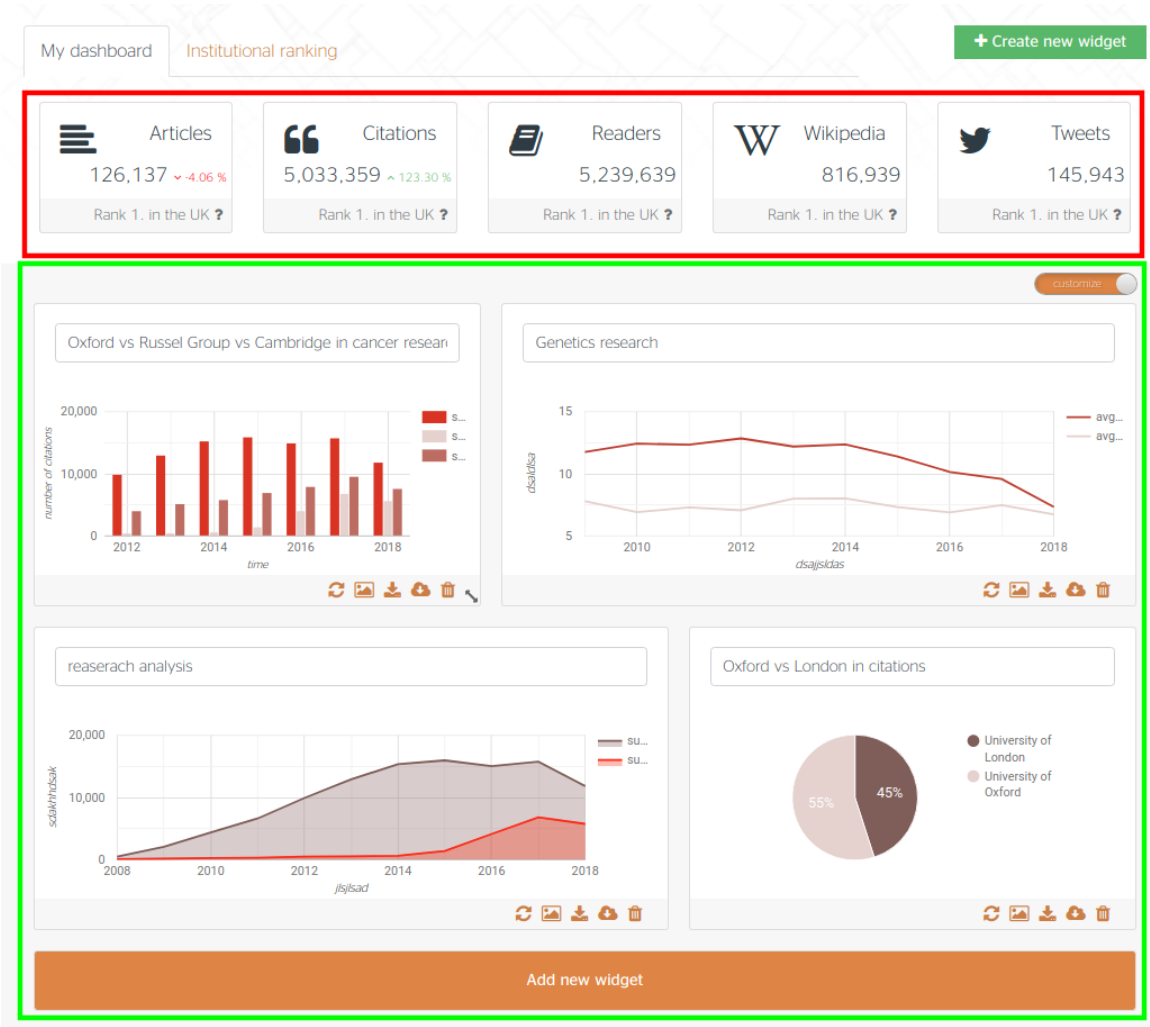
7.6 Upravovateľný panel

Hlavný panel sa nachádza pod štatistickým barom a obsahuje plochu, na ktorú sa dajú pridávať widgety, resp. grafy. Widgety sa dajú premiestňovať a je možné aj meniť ich veľkosť. Je to vyriešené pomocou javascriptovej knižnice GRIDSTACK.JS[12]. Ak chce užívateľ pracovať s týmito možnosťami, musí svoj panel prepnúť do editovacieho režimu, vďaka ktorému sa pri grafoch zobrazia možnosti presúvania a zmeny rozmerov grafu. Následne sa po upravení prepne do needitačného režimu, čo na pozadí vyvolá serializáciu rozloženia widgetov a následné uloženie do databázy. Vďaka tomu sa po ďalšom prihlásení užívateľovi zobrazí panel, ktorý si naposledy vytvoril.

Widgety sú na hlavnom paneli chápené ako grafy, ktoré sú obohatené o ďalšiu funkcionalitu:

[12]Dokumentácia javscriptovej knižnice GRIDSTACK.JS je dostupná na adrese http://gridstackjs.com/

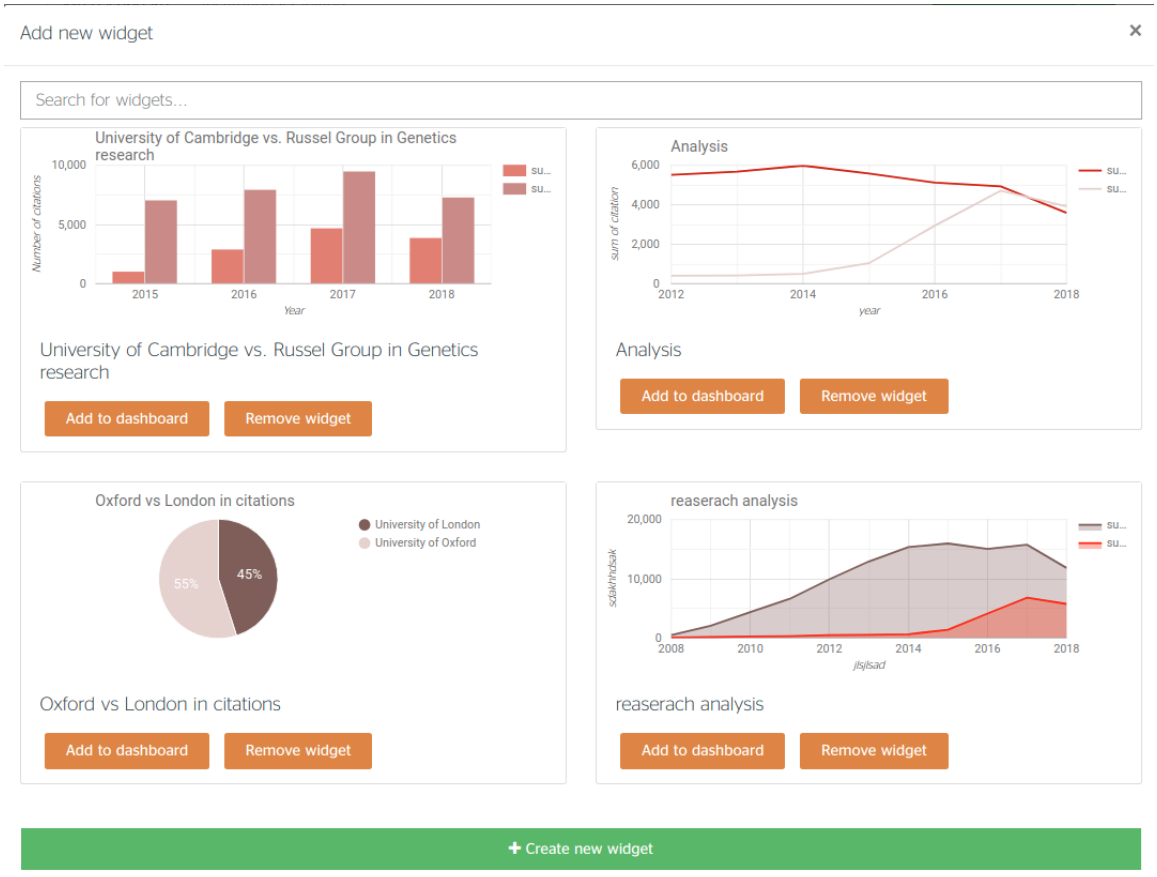
- Každý widget môže mať svoj názov. Vo východzom nastavení je názov widgetu totožný s názvom grafu. Unikátnosť názvu nie je nutná, dokonca názov nemusí byť zadaný vôbec.
- Ďalšou možnosťou je stiahnutie grafu v obrázkovej forme vo formáte *.png*.
- Užívateľ si môže vzorku dát, ktorú graf používal pri vytvorení, stiahnuť aj v textovom formáte *.csv*.
- Dáta, ktoré graf používa na vykreslenie, si môže užívateľ stiahnúť tiež v textovom formáte *.csv*.
- Poslednou fukncíou je odstránenie grafu z panelu.



Obr. 7.6: Príklad domovskej stránky, resp. hlavného panelu užívateľa. Červenou farbou je vyznačený štatistický panel a zelenou farbou je vyznačená upravovateľná zóna.

7.6.1 Knižnica widgetov

Pri režime editácie má užívateľ možnosť pridať nový widget po kliknutí na tlačidlo `Add new widget`, ktoré je umiestnené vždy naspodu upravovateľného panelu. Následne sa zobrazí modálne okno, ktoré obsahuje všetky dostupné widgety okrem tých, ktoré sú už umiestnené na hlavnom paneli. Medzi widgetami sa dá vyhľadávať podľa názvu a je možné ich mazať alebo pridávať na hlavný panel. Knižnicu widgetov je možné vidieť na obr. č. 7.7.



Obr. 7.7: Príklad domovskej stránky, resp. hlavného panelu užívateľa. Červenou farbou je vyznačený štatistický panel a zelenou farbou je vyznačená upravovateľná zóna.

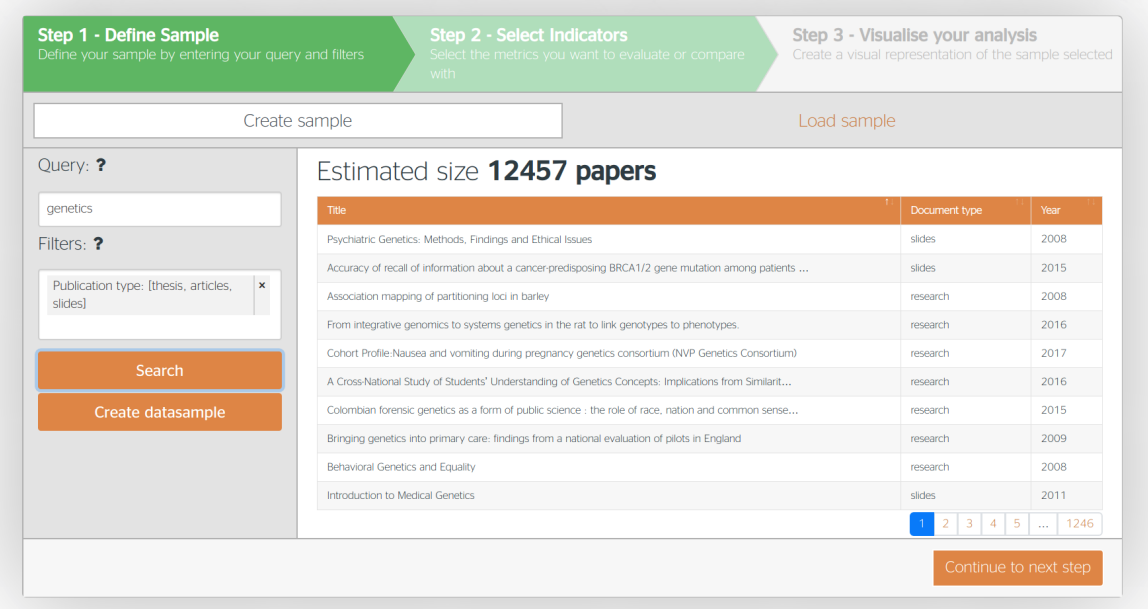
7.7 Vytváranie grafov

Aby bolo možné zobraziť grafy na hlavnom paneli, je nutné nejaký graf najprv vytvoriť. V tejte podkapitole su podrobne rozobraté tri kroky potrebné na vytvorenie grafu. Proces tvorby analýzy sa na portáli nazýva *Bechmarking wizard*. Vždy je na tejto podstránke zobrazený práve jeden krok. Medzi krokmi sa dá preklikávať a všetko podľa potreby meniť, napr. ak sa užívateľ nachádza v druhom kroku a rozhodne sa zmeniť vzorku dát, má možnosť sa vrátiť o krok späť a zmeniť ju bez nutnosti načítania stránky znova.

7.7.1 Definovanie vzorky dát

V tomto kroku sa vytvára vzorka dát, s ktorou sa bude ďalej v analýze pracovať. Každá vzorka musí mať unikátny názov a môže byť použitá pri viacerých porovnaniach. Užívateľ tu má možnosť rozhodnúť sa, či si chce vytvoriť novú vzorku dát, alebo použiť už existujúcu. Vzorka môže byť buď zmrazená (nemenná) alebo aktualizovaná (rozdiel je podrobnejšie vysvetlený v 6.3 Tvorba grafu). Proces vytvorenia vzorky dát môže trvať určitú dobu (pár sekúnd až pár minút). Užívateľ by musel dlho zotrvať v prvom kroku. Tento process preto nebolo možné robiť synchrónne, ale bolo nutné využiť asynchónnu funkcionalitu. Rozhodol som sa túto službu extrahovať a zaradiť ju do podporných služieb portálu, resp. skriptov, napísaných v programovacom jazyku Python. Naprogramoval som jednoduchú serverovú aplikáciu, ktorá prijme požiadavku na vytvorenie novej vzorky dát. Na pozadí spustí nový proces a vráti užívateľovi informáciu, že sa na tvorbe vzorky dát začalo pracovať. Vďaka tomu sa užívateľ môže presunúť k druhému kroku bez zbytočného čakania. V druhom kroku sa pracuje dočasne len so vzorkou 1000 dokumentov. Po vytvorení vzorky má užívateľ možnosť aktualizovať tieto dáta. Nič sa nestane, ak bude pokračovať len s 1000 dokumentami, avšak dáta v analýze môžu byť dočasne nepresné. Portál však ukáže upozornenie, ktoré je možné vidieť na obrázku č. 7.10. Návod na spustenie tejto podpornej služby je popísaný v prílohe B Spustenie serveru na tvorbu vzoriek dát. Vytvorenie vzorky dát, ktorá zachytáva všetky publikácie o genetike je zobrazené na obr. č. 7.8.

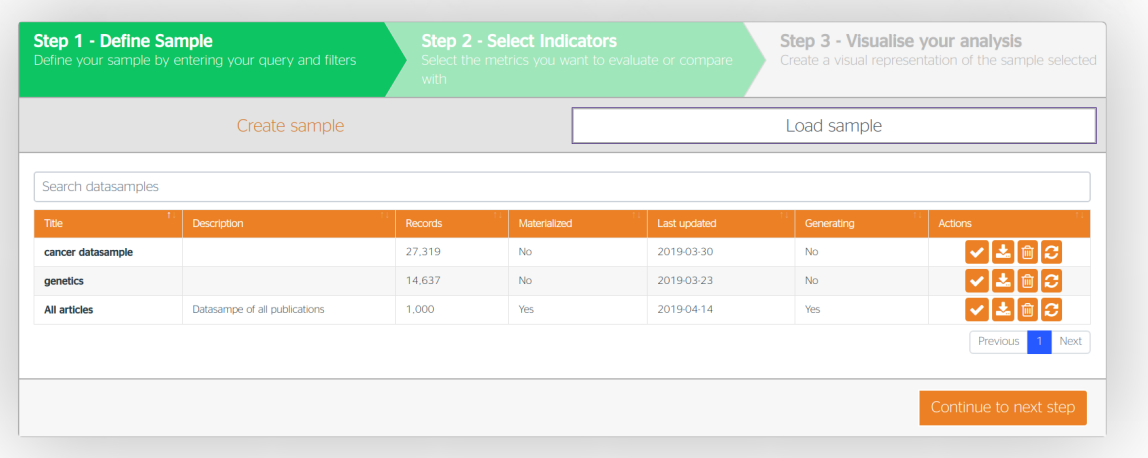
Pri vytváraní vzorky dát sa pri zmrazenom datasete okrem uloženia ID dokumentov získaných z projektu CORE skopíruju aj všetky dokumenty z databázy ClickHouse, ktoré sú obsiahnuté v tejto vzorke dát. Tieto dokumenty sa uložia do tabuľky `papers_data_materialized`, ktorá obsahuje rovnakú štuktúru ako tabuľka `papers_data` a obsahuje aj ID vzorky dát.



Obr. 7.8: Príklad vytvorenia vzorky dát.

Knižnica vzoriek dát

V knižnici vzoriek dát sa nachádzajú všetky už vytvorené vzorky, čo umožnuje na základe jednej vzorky vytvárať viac grafov. Knižnica je zobrazená ako tabuľka obsahujúca sedem stĺpcov, v ktorej sa dá vyhľadávať a radiť podľa jednotlivých hodnôt v stĺpcoch. Prvý stĺpec obsahuje názov vzorky dát, druhý obsahuje jej krátky popis, v treťom je aktuálny počet dokumentov, ktoré sa nachádzájú vo vzorke dát a v štvrtom je informácia či ide o materializovaný, alebo nematerializovaný pohľad. Ďalej sa tam nachádza dátum poslednej aktualizácie datasetu. V predposlednom stĺpci je informácia, či sa vzorka dát práve generuje, alebo už je vygenerovaná. V poslednom stĺpci sú zobrazené možnosti ako sa dá so vzorkou pracovať. Sú to štyri tlačidlá, pomocou ktorých sa dá vzorka dát aplikovať, aktualizovať [13], stiahnúť v textovom formáte *.csv* alebo odstrániť. [14]. Na obr. č. 7.9 je možné vidieť ukážku knižnice vzoriek dát.



Obr. 7.9: Reálny pohľad na knižnicu vzoriek dát.

7.7.2 Transformácia vzoriek dát

Druhý krok tvorby grafu sa zaoberá transformáciou vzorky dát na dáta použiteľné pre vykresľovanie grafu. Tento krok bol jeden z najzložitejších, pretože bolo nutné vytvoriť komplexný SQL dotaz, ktorý by bolo možné dynamicky meniť a aplikovať na všetky možné prípady.

Tvorba dynamického SQL dotazu

SQL dotaz sa dynamicky tvorí na základe údajov z formuláru v druhom kroku (na obr. č. 7.10 zobrazený vľavo) a pozostáva z piatich sekcií (v ukážkovom kóde 7.2 je každá sekcia vyznačená komentármi zelenej farby).

1. V prvej sekcii sa definuje, podľa akého kľúča sa bude agregovať. Ide o údaje, ktoré budú zobrazené na x-ovej osi grafu. Užívateľ má možnosť vybrať si medzi rokom (*year*) alebo inštitúciou (*institution*).

[13] Materializované vzorky dát nie je možné aktualizovať.
[14] Odstránenie vzorky dát nie je možné, ak ju ešte niektorý graf používa.

2. V druhej sekcii sa dynamicky definujú agregácie. Ak si užívateľ vybral ako kľúč agregácie (*variable*) čas, tak môže agregovať len citácie. V opačnom prípade, ak si užívateľ zvolil ako agregačný kľúč inštitúciu, tak môže agregovať nielen citácie ale aj čitateľov (*readers*). Medzi agregačnými funkciami, ktoré užívateľ môže použiť sú priemer (*mean*), celková suma (*sum*) a medián (*median*). Pri agregačných funkciách sa berú do úvahy len dokumenty, ktoré boli publikované na vybraných univerzitách.

3. V tretej sekcii je definované, ktoré dokumenty sa majú brať do úvahy. Ide o všetky dokumenty, ktoré boli definované v prvom kroku. Je tu možné vidieť optimalizačnú klauzulu `PREWHERE` typickú pre ClickHouse SQL dialekt, ktorá má rovnakú funkcionalitu ako klauzula `WHERE` s jedným rozdielom, a síce klauzula `PREWHERE` najskôr načíta iba stĺpce, ktoré sú nutné pre vykonanie príkazu v tejto klauzule a následne, ak je výraz pravdivý, tak sa načítajú aj ostatné stĺpce nutné pre vykonanie zvyšku dotazu [28].

4. V štvtej sekcii sa spája tabuľka inštitúcií s tabuľkou dokumentov. Vzhľadom na to, že relácia medzi týmito dvoma tabuľkami je typu M:N, bolo nutné použiť spojovaciu tabuľku `linking`.

5. Piata sekcia sa tvorí len v prípade, ak bol ako agregačný kľúč zvolený čas. Citačné dáta sú uložené v databáze trochu netradične vo forme dvojrozmerného pola. Prvý prvok je rok a druhý je počet citácií v konkrétnom roku. Databáza ClickHouse umožňuje natívnu podporu polí, vďaka čomu je možné využívať klauzulu `ARRAY JOIN` typickú pre ClickHouse SQL dialekt. Táto klauzula umožňuje expandovať prvky z polí do nových riadkov. Údaje sú kopírované zo zdrojového riadku a obohatené o prvky z pola [28] [1].

   Demonštrácia funkcionality `ARRAY JOIN`:

   Predstavme si tabuľku `paper_data` 7.1 s tromi stĺpcami – `coreID`, `DOI`, `time_array`, ktorá obsahuje jeden záznam. Prvé dva stĺpce nesú identifikátor dokumentu a tretí stĺpec ukladá citačné dáta vo formáte
   `[[rok, pocet_citacii], [rok, pocet_citacii]...]`.

coreID	DOI	time_array
5084	10.1088/0004-637x/706/1/l76	[[2009, 10], [2010, 5], [2018, 16]]

Tabuľka 7.1: Príklad uloženia dát v tabuľke `paper_data`

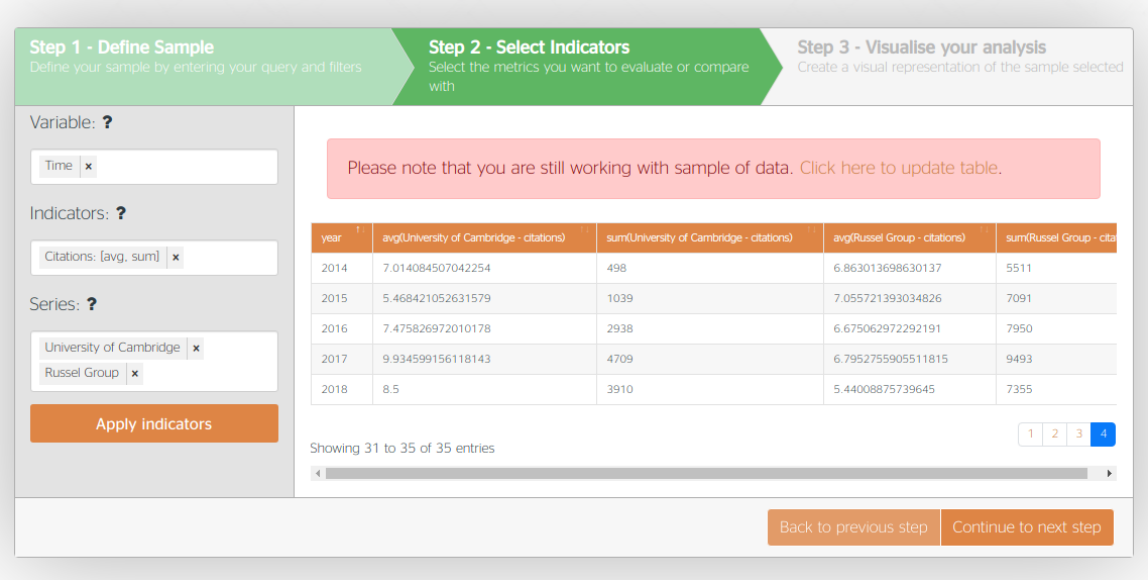
Po spustení nasledovného SQL dotazu:

```
SELECT coreId, DOI, t
FROM paper_data
ARRAY JOIN time_array AS t
```

Výpis 7.1: Demonštrácia `ARRAY JOIN` dotazu.

Získame tabuľku:

coreID	DOI	t
5084	10.1088/0004-637x/706/1/l76	[2009, 10]
5084	10.1088/0004-637x/706/1/l76	[2010, 5]
5084	10.1088/0004-637x/706/1/l76	[2018, 16]



Obr. 7.10: Reálny pohľad na druhý krok vizualizácie – transformácia vzorky dát.

```
SELECT
  -- Sekcia 1 - Kľúč zgrupovania, t.j. čo sa bude nachádzať na osi x
  year,
  -- Čiastočné ukončenie sekcie 1
  -- Sekcia 2: Dynamicky pridávateľné agregácie
  sum(
    if(institution_id = 140, arrayElement(t, 2) , null)
  ) AS 'sum(Russel Group - citations)',
  avg(
    if(institution_id = 140, arrayElement(t, 2) , null)
  ) AS 'sum(Russel Group - citations)',
  sum(
    if(institution_id = 15, arrayElement(t, 2) , null)
  ) AS 'sum(University of Cambridge - citations)'
  avg(
    if(institution_id = 15, arrayElement(t, 2) , null)
  ) AS 'sum(University of Cambridge - citations)'
  -- Koniec sekcie 2
FROM (
      SELECT
        coreID,
        repository_id,
        readers,
        pub_year,
        citations,
        time_array
      FROM papers_data
      -- Sekcia 3: Získanie údajov len z dokumentov, ktoré sú v datasete z prvého kroku.
      PREWHERE (
          repository_id IN (27,684 ...)
      )
      WHERE (
          coreID IN (77596237,41990676 ...)
      )
      -- Koniec sekcie 3.
```

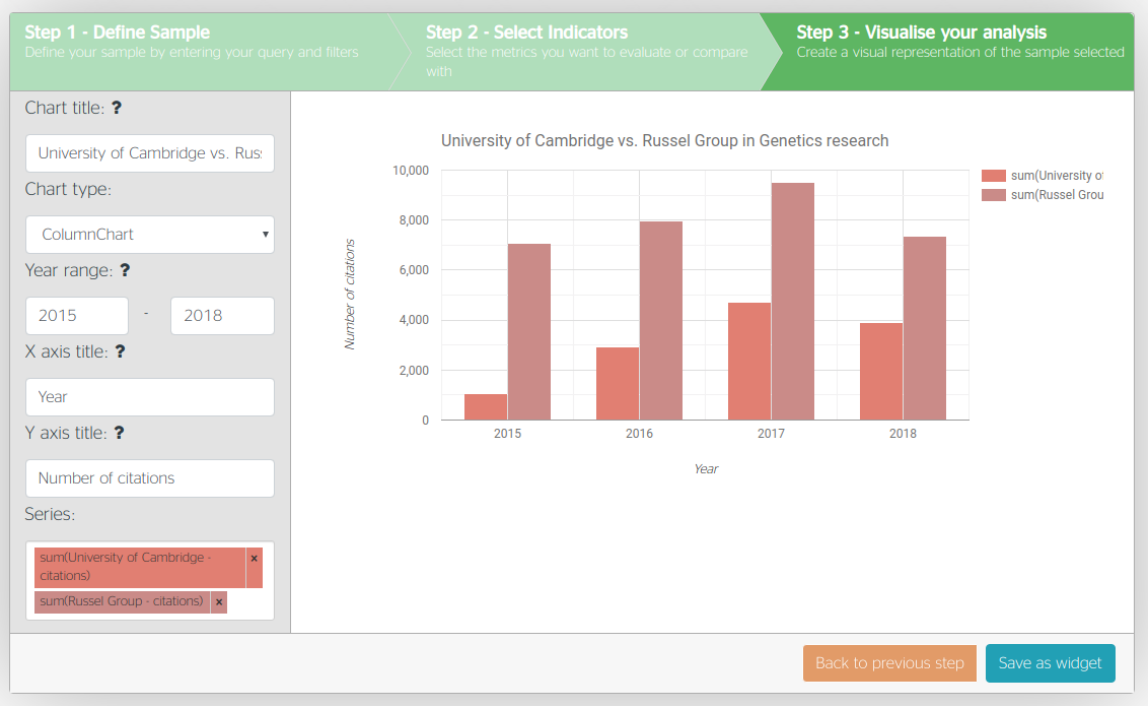
```
    -- Sekcia 4: Remapovanie repozitárov na univerzity
    ) ALL INNER JOIN (
      SELECT
        institution_id,
        repository_id,
        institution_name,
        institution_type
      FROM (
        SELECT
          id AS institution_id,
          name AS institution_name,
          type AS institution_type
        FROM institutions
        WHERE(
          institution_id IN (140,15)
        )
      ) ALL INNER JOIN (
        SELECT
          id_paper AS repository_id,
          id_institution AS institution_id
        FROM
          linking
      ) USING institution_id
    ) USING repository_id
    -- Koniec sekcie 4
-- Sekcia 5 (Definuje sa len, ak je agregačným kľúčom čas): rozvinutie citačných dát
ARRAY JOIN time_array AS t
WHERE (
  arrayElement(t, 1) IS NOT NULL AND arrayElement(t, 1) > 1800 AND arrayElement(t, 1) <= 2018
)
-- Koniec sekcie 5
-- Pokračovanie sekcie 1
GROUP BY arrayElement(t, 1) as year
-- Koniec sekcie 1
```

Výpis 7.2: Tvorba dynamického ClikcHouse SQL dotazu na transformáciu vzorky dát.

7.7.3 Vizualizácia

V poslednom kroku sa tvorí samotný graf. Dáta z predchádzajúceho kroku (pozn. 7.7.2 Transformácia vzoriek dát) sú prentrasformované na inštanciu triedy `DataView`, ktorá slúži ako zdroj údajov grafu [10].

Z knižnice Google Chart som použil 10 typov grafov, z nich tri su podporované, ak je zvolená premenná `institutions` – stĺpcový, pruhový a koláčový graf a sedem je podporovaných, ak je zvolená premenná `time` – stĺpcový, plošný, pruhový, spojnicový, kombinovaný, bodový a sparkline graf. Graf je možné upravovať vďaka voliteľným prispôsobeniam, ktoré sú reprezentované formulárom na ľavej strane. Názov grafu, x a y osi sa dynamicky mení pri zmene príslušných polí vo formulári. Časový rozsah rokov (*year range*) sa dá zvoliť iba ak sa pracuje s premennou `time`, v opačnom prípade sa táto možnosť užívateľoví nezobrazí. Tieto nastavenia sú ukladané vo formáte JSON a predávané metóde `draw()`. Metóda `draw()` je metóda definovaná nad objektom `google.visualization`, ktorý reprezentuje samotný graf [14]. Ukážka tvorby grafu je zobrazená na obr. č. 7.11.

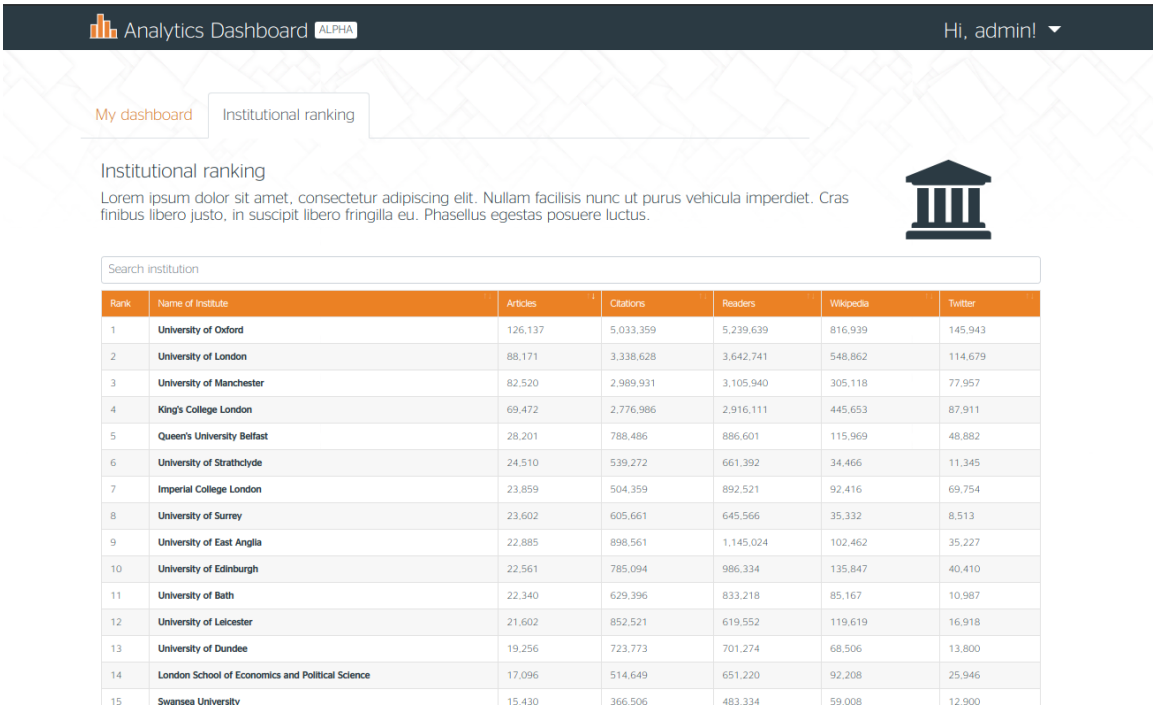


Obr. 7.11: Reálny pohľad na tretí krok vizualizácie – tvorba samotného grafu.

7.8 Štatitiky univerzít

Štatistiky univerzít sú zobrazené v tabuľke, v ktorej sa dá vyhľadávať podľa názvu univerzity a radiť dáta vo vzostupnom alebo zostupnom smere. K dosiahnutiu tejto funkcionality som využil jQuery rozšírenie s názvom DataTables[15]. Pre optimalizáciu rýchlosti načítavania týchto dát som použil už vytvorený materializovaný pohľad (7.5.2 Materializovaný pohľad na umiestnenia univerzít), ktorý dopredu agregoval všetky metriky pre jednotlivé univezity. Podstránku je možné vidieť na obr. č. 7.12.

[15]Dokumentácia rozšírenia DataTables je dostupná na adrese – https://datatables.net/manual/index



Rank	Name of Institute	Articles	Citations	Readers	Wikipedia	Twitter
1	University of Oxford	126,137	5,033,359	5,239,639	816,939	145,943
2	University of London	88,171	3,338,628	3,642,741	548,862	114,679
3	University of Manchester	82,520	2,989,931	3,105,940	305,118	77,957
4	King's College London	69,472	2,776,986	2,916,111	445,653	87,911
5	Queen's University Belfast	28,201	788,486	886,601	115,969	48,882
6	University of Strathclyde	24,510	539,272	661,392	34,466	11,345
7	Imperial College London	23,859	504,359	892,521	92,416	69,754
8	University of Surrey	23,602	605,661	645,566	35,332	8,513
9	University of East Anglia	22,885	898,561	1,145,024	102,462	35,227
10	University of Edinburgh	22,561	785,094	986,334	135,847	40,410
11	University of Bath	22,340	629,396	833,218	85,167	10,987
12	University of Leicester	21,602	852,521	619,552	119,619	16,918
13	University of Dundee	19,256	723,773	701,274	68,506	13,800
14	London School of Economics and Political Science	17,096	514,649	651,220	92,208	25,946
15	Swansea University	15,430	366,506	483,334	59,008	12,900

Obr. 7.12: Tabuľka štatistík všetkých univerzít.

Kapitola 8

Záver

Podarilo sa mi vytvoriť portál pre porovnávanie vedeckých výsledkov univerzít v Spojenom kráľovstve Veľkej Británie a Severného Írska. Získal som metriky o 728 728 dokumentoch z 139 univerzít. Ciele, ktoré som si v tejto práci stanovil boli splnené. Portál bol riadne otestovaný a je plne funkčný. Otestovať ho malo možnosť viacero ľudí, medzi ktorými bol aj môj supervízor na Erasmus stáži Ing. Petr Knoth, PhD., ktorý dohliadal na to, aby vývoj smeroval správnym smerom.

Je však nutné podotknúť, že výstupom práce bol len prototyp portálu. V budúcnosti je možné ho vylepšiť napríklad pridaním nových funkcionalít. V užívateľskom rozhraní by sa mohla pridať funkcionalita reeditácie vytvorených grafov, ktoré momentálne nie je možné spätne editovať. Do portálu by bolo možné pridať aj viac metrík. Ponúkajú sa napríklad metriky zo zdroja *CrossRef Event data*, pretože z celkových 11 dostupných zdrojov som vybral len dva najzaujímavejšie. Tieto návrhy však presahovali ciele práce a preto som sa im nevenoval.

Literatúra

[1] arrayJoin function. Yandex, [Online; navštívené 13. 04. 2019].
URL
https://clickhouse.yandex/docs/en/query_language/functions/array_join/

[2] Atwood, T. M.; aj.: An object-oriented DBMS for design support applications. *Proc. COMPINT*, 1985: s. 299–307.

[3] Berka, P.: *Dobývání znalostí z databází.* Praha: Academia, prvé vydanie, 2003, ISBN 8020010629, 35–45 s.

[4] Bhagat, V.; Gopal, A.: Comparative Study of Row and Column Oriented Database. In *2012 Fifth International Conference on Emerging Trends in Engineering and Technology*, IEEE, 2012, ISBN 9781479902767, s. 196–201.

[5] Bonasio, A.: A look at Mendeley Readership Statistics. September 2014, [Online; navštívené 27. 10. 2018].
URL https://blog.mendeley.com/2014/09/18/a-look-at-mendeley-readership-statistics/

[6] Choi, Y. M.: Intro to OLAP: Codd defined 12 Rules. Drexel University College of Computing & Informatics, [Online; navštívené 27. 04. 2019].
URL http://www.cis.drexel.edu/faculty/song/courses/info%20607/tutorial_olap/definition_Codd.htm

[7] Codd, E.: A relational model of data for large shared data banks. *Communications of the ACM*, ročník 13, č. 6, 1970: s. 377–387, ISSN 1557-7317.

[8] Codd, E. F.; Codd, S. B.; Salley, C. T.: Providing OLAP (on-line analytical processing) to user-analysts: An IT mandate. *Codd and Date*, ročník 32, 1993.

[9] Danel, R.: OLAP. [Online; navštívené 27. 04. 2019].
URL http://homel.vsb.cz/~dan11/is_skripta/IS%202010%20-%20Danel%20-%20OLAP.pdf

[10] DataTables and DataViews. Google, [Online; navštívené 14. 04. 2019].
URL https://developers.google.com/chart/interactive/docs/datatables_dataviews

[11] Denis, E. G.: Digital object identifier (DOI) becomes an ISO standard. Máj 2012, [Online; navštívené 20. 10. 2018].
URL https://www.iso.org/news/2012/05/Ref1561.html

[12] Derrett, N.; Lyngbaek, P.: Some aspects of operations in an object-oriented database. *quarterly bulletin*, 1985: str. 6.

[13] Digital Object Identifier System FAQs. International DOI Foundation, Júl 2018, [Online; navštívené 20. 10. 2018].
URL https://www.doi.org/faq.html

[14] Google Visualization API Reference. Google, [Online; navštívené 14. 04. 2019].
URL
https://developers.google.com/chart/interactive/docs/reference#visdraw

[15] How is CORE different from Google Scholar? CORE, [Online; navštívené 20. 10. 2018].
URL https://core.ac.uk/about#faqs

[16] How to Update Data in ClickHouse. Yandex, November 2016, [Online; navštívené 02. 03. 2019].
URL https://clickhouse.yandex/blog/en/how-to-update-data-in-clickhouse

[17] Hruška, T.; Křivka, Z.: Študijný opora pre predmet IIS - Pojem informačního systému, Data, Procesy, Transakce. Február 2012, [Online; navštívené 02. 03. 2019].

[18] Knoth, P.; Zdrahal, Z.: CORE: Three Access Levels to Underpin Open Access. *D-Lib Magazine*, ročník 18, č. 11, 2012, ISSN 1082-9873.
URL http://www.dlib.org/dlib/november12/knoth/11knoth.html

[19] Kudlička, M.: *Nástroj pro definici a evoluci databázového schématu*. Diplomová práca, VUT FIT Brno, Brno, 2006.

[20] Lacko, L.: *Databáze: datové sklady, OLAP a dolování dat s příklady v Microsoft SQL Serveru a Oracle*. Brno: Computer Press, prvé vydanie, 2003, ISBN 8072269690, 111–117 s.

[21] Lammey, R.: About us. Crossref, Jún 2018, [Online; navštívené 27. 10. 2018].
URL https://www.crossref.org/about/

[22] Litwintschik, M.: Summary of the 1.1 Billion Taxi Rides Benchmarks. Január 2019, [Online; navštívené 02. 03. 2019].
URL https://tech.marksblogg.com/benchmarks.html

[23] Oppel, A.: *Databáze bez předchodzích znalostí*. nám. 28. dubna 48, 63500 Brno: Computer Press, a.s., 2006, ISBN 80-251-1199-7, 22–31 s.

[24] Profile - Russel Group. The Russell Group, [Online; navštívené 12. 04. 2019].
URL https://russellgroup.ac.uk/media/5524/rg_text_june2017_updated.pdf

[25] Read the Budapest Open Access Initiative. Budapest Open Access Initiative, [Online; navštívené 20. 10. 2018].
URL https://www.budapestopenaccessinitiative.org/read

[26] Reference Manager and Academic Social Network - Mendeley Database | Elsevier Solutions. Elsevier B.V., Október 2018, [Online; navštívené 27. 10. 2018].
URL https://www.elsevier.com/solutions/mendeley

[27] Rubin, A.: Column Store Database Benchmarks: MariaDB ColumnStore vs. Clickhouse vs. Apache Spark. Marec 2017, [Online; navštívené 02. 03. 2019].
URL https://www.percona.com/blog/2017/03/17/column-store-database-benchmarks-mariadb-columnstore-vs-clickhouse-vs-apache-spark/

[28] SELECT Queries Syntax. Yandex, [Online; navštívené 13. 04. 2019].
URL https://clickhouse.yandex/docs/en/query_language/select/

[29] Sinha, A.; Shen, Z.; Song, Y.; aj.: An Overview of Microsoft Academic Service (MAS) and Applications. In *Proceedings of the 24th International Conference on World Wide Web – WWW 15 Companion*, ACM Press, 2015, s. 243–246, doi:10.1145/2740908.2742839.
URL https://doi.org/10.1145/2740908.2742839

[30] Stephens, R.; Plew, R.: *Sams Teach Yourself Beginning Databases*. 201 West 103rd St., Indianapolis, Indiana 46290 USA: Sams Publishing, prvé vydanie, 2003, ISBN 0-672-32492-X, 66–85 s.

[31] Wendt, K.: *Traffic Monitor: Data Display for Traffic Visualisation at Airports*. Dizertačná práca, Faculty of Computer Science and Mathematics, University of Passau, 01 2007.
URL https://www.researchgate.net/publication/201392539_Traffic_Monitor_Data_Display_for_Traffic_Visualisation_at_Airports

[32] Zach, A.: Event Data — open for your interpretation. Crossref, November 2017, [Online; navštívené 05. 04. 2019].
URL https://www.crossref.org/services/event-data/

Príloha A

Návod na spustenie skriptov na sťahovanie dát

V tejte prílohe je vysvetlený inštalačný proces skriptov na sťahovanie dát. V nasledujúcich podkapitolách je popísaný návod na spustenie jednotlivých skriptov aj s vysvetlenými parametrami, ktoré je možné použiť. V návode sa predpokladá, že používateľ má nainštalované určité prerekvizity. Prerekvizity sú vyznačená na začiatku každej podkapitoly *malým šikmým písmom*.

A.1 Inštalácia sťahovacích skriptov

Vychádza sa z predpokladu, že prerekvizita Python 3.5[1] *je úspešne nainštalovaná.*

Pred samotným spustením sťahovania dát je nutné vytvoriť virtuálne prostredie a v ňom si nainštalovať všetky potrebné knižnice. To sa dosiahne spustením nasledovných príkazov v zložke, kde sa nachádzajú skripty (pozn.: prvé tri príkazy je možné preskočiť a knižnice nainštalovať globálne).

```
$ mkdir .env
$ python3 -m venv .env/
$ . .env/bin/activate
$ pip3 install -r requirements.txt
```

A.2 Stiahnutie dokumentov z databázy CORE

Skripty na stiahnutie metrík z MAG, *CrossRef EventData* a *Mendeley Readership* musia dostať na vstup zoznam DOI identifikátorov, pre ktoré majú získať informácie. Skript pre sťahovanie všetkých UK dokumentov je možné spustiť pomocou príkazu:

```
$ python -m analytics download core
```

Je možné špecifikovať len určitý repozitár pomocou pozičného parametru `repository_id`:

```
$ python -m analytics download core <repository_id>
```

Dokumenty sa ukladajú do súboru `output/core_articles.csv`.

[1] Inštalačný súbor dostupný na adrese – https://www.python.org/downloads/.

A.3 Stiahnutie sociálnych signálov z CrossRef Event Data

Stiahnutie sociálnych signálov pre všetky britské dokumenty, resp. pre repozitár je možné spustiť príkazom, pričom je možné použiť nepovinný parameter `repository_id`, ktorým sa špecifikuje, pre aký repozitár sa majú stiahnúť dáta:

```
$ python -m analytics fetch event-data [<repository_id>]
```

V súbore `analytics/settings.py` je možné zmeniť počet súčastne pracujúcich vlákien (parameter `CROSS_REF_WORKERS_NUMBER`) [2] ako aj ďalšie relevantné parametre (ide o parametre obsahujúce prefix `CROSSREF_`). Získané údaje sa uložia do súboru s názvom `/output/event_data_all.csv`.

A.4 Získanie počtu čitateľov z platformy Mendeley

Skript na sťahovanie *Mendeley Readership* metrík je možné spustiť príkazom.

```
$ python -m analytics fetch mendeley-data \
$           (<repository_id> | --from-file <file_path>)
```

Je možné špecifikovať, pre ktorý repozitár sa majú dáta sťahovať alebo špecifikovať súbor, v ktorom sa nachádzajú DOI identifikátory (prednastavená cesta je `output/core_articles.csv`).

Skript sťahuje dáta vo viacerých vláknach súčasne a výsledky sú ukladané do súboru `/output/mendeley_readership.csv`. Pre správne fungovanie skriptu je nutné zadať `MENDELEY_CLIENT_ID` a `MENDELEY_CLIENT_SECRET` v súbore `analytics/settings.py`. Tieto údaje sa dajú bezplatne vygenerovať na oficiálnej stránke projektu[3]. Pomocou parametru `MENDELEY_WORKERS_NUMBER` je možné zmeniť aj počet súčasne pracujúcich vlákien[4].

A.5 Zlučovanie dát

Získané dáta je nutné zlúčiť do jedného súboru. O tento proces sa stará skript, ktorý sa dá spustiť pomocou príkazového riadku nasledovne:

```
$ python -m analytics merge
```

Skript zoberie všetky získané dáta zo zložiek `output/` a `input/` a vytvorí z nich jeden súbor, ktorý uloží do súboru `outputfinal_merged.csv`.

[2] Síce *Crossref* API rozhranie nie je limitované ale tento parameter je potrebné voliť rozumne, aby neboli servery príliš preťažované.

[3] *Mendeley* API dokumentácia je dostupná na adrese https://dev.mendeley.com/.

[4] Síce neexistujú žiadne limity poslaných požiadaviek na servery projektu *Mendeley* je však vhodné tento parameter voliť rozumne apríliš servery nezaťažovať.

Príloha B

Návod na vytvorenie databáz a spustenie portálu

V tomto dodatku je popísaný presný návod na spustenie portálu a jeho podporných služieb. Jednotlivé kroky na seba nadväzujú a ich preskočenie môže priniesť neočakávané chyby. V návode sa predpokladá, že používateľ má nainštalované určité prerekvizity. Prerekvizity sú vyznačená na začiatku každej podkapitoly *malým šikmým písmom*.

B.1 Proces inštalácie PHP knižníc

Vychádza sa z predpokladu, že prerekvizity PHP >= 7.2 a Composer, ktorý framework Symfony používa pre správu jeho závislostí [1] *sú úspešne nainštalované.*

Nasledovným príkazom sa nainštalujú všetky potrebné PHP knižnice a ich závislosti:

```
$ composer install
```

Počas inštalácie skript ponúkne možnosť upresnenia niektorých konfiguračných nastavení ako napríklad adresa databázy, adresa serveru ElasticSearch a adresa podporného serveru napísaného v programovacom jazyku Python. Východzie nastavenia je možné nájsť v súbore `app/config/parameters.yml.dist` a tieto hodnoty je možnémskopírovať.

B.2 Vytvorenie databáz a tabuliek

Pre tento krok je nutné mať už nainštalované databázové servery MariaDB vo verzii 10.0[2] *a Clickhouse Yandex vo verzii 1.1.54388*[3]*. Nevyhnutný je aj finálny súbor všetkých metadát o dokumentoch získaný v A.5 Zlučovanie dát*

B.2.1 Vytvorenie a import dát do Clickhouse databáze

Predpokladá sa, že sa nachádzate v zložke, kde sú uložené skripty napísané v programovacom jazyku Python a zároveň sú tieto skripty nainštalované a dáta sú úspešne stiahnuté (viac v A Návod na

[1] Proces inštalácie manažéra PHP knižníc Composer je dostupný na https://getcomposer.org/doc/00-intro.md.

[2] Inštalačný proces databázového systému MariaDB je možné nájsť na adrese https://mariadb.com/kb/en/library/getting-installing-and-upgrading-mariadb/.

[3] Inštalačný proces databázového systému Yandex ClickHouse je možné nájsť na adrese https://clickhouse.yandex/docs/en/getting_started/.

spustenie skriptov na sťahovanie dát).

Databáza ClickHouse ukladá metadáta o dokumentoch, získaných z kapitoly 3 Zdroje dát. Všetky nastavenia relatívne k databáze ClickHouse je možné upraviť v súbore `analytics/settings.py`. Ide hlavne o nastavenia s prefixom `CLICKHOUSE_`, t.j. názov databázy, URI adresa databázy. O vytvorenie a import do tabuľky sa stará skript napísaný v programovacom jazyku Python, ktorý je možné spustiť nasledovným príkazom:

```
$ python -m analytics import real data
```

Eventuálne je možné importovať len testovacie (náhodné) dáta príkazom:

```
$ python -m analytics random data
```

Tieto príkazy vytvoria novú tabuľku v databáze ClickHouse s názvom `papers_data`.

Ďalším príkazom sa vytvorí tabuľka inštitúcií s názvom `institutions` a spojovacia tabuľka s názvom `linking`. Dáta sú vzaté z manuálneho mapovania repozitárov univerzít (4.2.1 Získanie mapovania repozitárov britských univerzít).

```
$ python -m analytics import institutions table
```

B.2.2 Vytvorenie tabuliek v databáze MariaDB

Proces vytvorenie tabuliek v databáze MariaDB je priamočiary. Nasledovnými dvoma príkazmi sa vytvoria tri tabuľky – `ad_dashboard_users`, `ad_chart_settings` a `ad_dataset`, ktoré ukladajú informácie o užívateľoch, ich vzoriek dát a grafov.

```
$ php bin/console doctrine:database:create
$ php bin/console doctrine:schema:update --force
```

B.3 Spustenie serveru na tvorbu vzoriek dát

Predpokladá sa, že sa nachádzate v zložke so skriptami napísanými v programovacom jazyku Python a tieto skripty sú úspešne nainštalované.

Ako bolo spomenuté v 7.7.1 Definovanie vzorky dát, pre vytvorenie vzoriek dát bolo nevyhnutné naprogramovať externú službu, ktorú je možné spustiť príkazom:

```
$ python -m analytics run server
```

Číslo portu, na ktorom je server pripojený, je 8124. Toto číslo portu je možné upravovať zmenou nastavenia `SERVER_PORT` v súbore `analytics/settings.py`.

B.4 Vytvorenie admin užívateľa

Užívateľ *admin* sa vytvorí pomocou nasledovného príkazu, pričom je možné nastavit jeho meno, email a heslo. Posledný parameter indikuje, aké privilégia má užívateľ mať. Takto je možné vytvárať aj ostatných užívateľov, ale neodporúča sa to. Jednoduchší spôsob je pomocou administračného panelu portálu.

```
$ php bin/console fos:user:create admin admin@example.com p@ssword --super-admin
```

B.5 Spustenie portálu

Pre developerské účely je portál možne spustiť v režime *dev* v ktorom sú zmeny kódu reflektované hneď na portáli. Prvým príkazom sa spustí server Symfony a druhým príkazom sa spustí kompilácia štýlov CSS.

```
$ php bin/console server:run
$ yarn encore dev --watch
```

Pre spustenie produkčného serveru je nutné postupovať podľa návodu v officiálnej dokumentácii pre framework Symfony – https://symfony.com/doc/3.4/deployment.html