

Vylepšení aplikace profily autorů

Dominik Palán

Bakalářská práce
2023



Univerzita Tomáše Bati ve Zlíně
Fakulta aplikované informatiky

Univerzita Tomáše Bati ve Zlíně
Fakulta aplikované informatiky
Ústav informatiky a umělé inteligence

Akademický rok: 2022/2023

ZADÁNÍ BAKALÁŘSKÉ PRÁCE

(projektu, uměleckého díla, uměleckého výkonu)

Jméno a příjmení: **Dominik Palán**
Osobní číslo: **A19083**
Studijní program: **B3902 Inženýrská informatika**
Studijní obor: **Softwarové inženýrství**
Forma studia: **Prezenční**
Téma práce: **Vylepšení aplikace profily autorů**
Téma práce anglicky: **Enhancement of an Author Profiles application**

Zásady pro vypracování

1. Proveďte literární rešerši na dané téma.
2. Shromážděte požadavky na rozhraní.
3. Navrhněte technické řešení ve formě úprav webové aplikace.
4. Zdůvodněte výběr jednotlivých komponentů technického řešení.
5. Realizujte a otestujte výsledné technické řešení ve spolupráci s uživatelem.
6. Věnujte pozornost zabezpečení aplikace.



Forma zpracování bakalářské práce: **tištěná/elektronická**

Seznam doporučené literatury:

1. CHAFFER, Jonathan a Karl SWEDBERG. Mistrovství v jQuery: [kompletní průvodce vývojáře]. Brno: Computer Press, 2013, 384 s. Mistrovství v. ISBN 9788025141038.
2. YORK, Richard. Web development with jQuery. Indianapolis, IN: Wrox, a Wiley brand, [2015], 1 online resource. Programmer to programmer. ISBN 9781119209430. Dostupné také z: <https://onlinelibrary.wiley.com/doi/book/10.1002/9781119209430>.
3. NIXON, Robin. Learning PHP, MySQL & JavaScript: with jQuery, CSS & HTML5. Fourth edition. Beijing: O'Reilly, 2014, xxvii, 780 s. ISBN 9781491918661.
4. JQuery – kuchařka programátora. Brno: Computer Press, 2010, 436 s. ISBN 9788025131527.
5. Architectural Styles and the Design of Network-based Software Architectures. ics.uci.edu [online], 2004 [cit. 2021-11-08], Dostupné z: <https://www.ics.uci.edu/fielding/pubs/dissertation/top.htm>
6. Web Services Architecture. w3c.org [online], 2004 [cit. 2021-11-08], Dostupné z: <https://www.w3.org/TR/2004/NOTE-ws-arch-20040211>.
7. Standard ECMA-404: The JSON Data Interchange Syntax. Ecma International [online]. 2017, December 2017. Dostupné z: <https://ecma-international.org/publications/files/ECMA-ST/ECMA-404.pdf>.

Vedoucí bakalářské práce: **doc. Ing. Zdenka Prokopová, CSc.**
Ústav počítačových a komunikačních systémů

Datum zadání bakalářské práce: **2. prosince 2022**
Termín odevzdání bakalářské práce: **26. května 2023**

doc. Ing. Jiří Vojtěšek, Ph.D. v.r.
děkan



prof. Mgr. Roman Jašek, Ph.D., DBA v.r.
ředitel ústavu

Ve Zlíně dne 7. prosince 2022

Prohlašuji, že

- beru na vědomí, že odevzdáním bakalářské práce souhlasím se zveřejněním své práce podle zákona č. 111/1998 Sb. o vysokých školách a o změně a doplnění dalších zákonů (zákon o vysokých školách), ve znění pozdějších právních předpisů, bez ohledu na výsledek obhajoby;
- beru na vědomí, že bakalářská práce bude uložena v elektronické podobě v univerzitním informačním systému dostupná k prezenčnímu nahlédnutí, že jeden výtisk bakalářské práce bude uložen v příruční knihovně Fakulty aplikované informatiky Univerzity Tomáše Bati ve Zlíně;
- byl/a jsem seznámen/a s tím, že na moji bakalářskou práci se plně vztahuje zákon č. 121/2000 Sb. o právu autorském, o právech souvisejících s právem autorským a o změně některých zákonů (autorský zákon) ve znění pozdějších právních předpisů, zejm. § 35 odst. 3;
- beru na vědomí, že podle § 60 odst. 1 autorského zákona má UTB ve Zlíně právo na uzavření licenční smlouvy o užití školního díla v rozsahu § 12 odst. 4 autorského zákona;
- beru na vědomí, že podle § 60 odst. 2 a 3 autorského zákona mohu užít své dílo – bakalářskou práci nebo poskytnout licenci k jejímu využití jen připouští-li tak licenční smlouva uzavřená mezi mnou a Univerzitou Tomáše Bati ve Zlíně s tím, že vyrovnání případného přiměřeného příspěvku na úhradu nákladů, které byly Univerzitou Tomáše Bati ve Zlíně na vytvoření díla vynaloženy (až do jejich skutečné výše) bude rovněž předmětem této licenční smlouvy;
- beru na vědomí, že pokud bylo k vypracování bakalářské práce využito softwaru poskytnutého Univerzitou Tomáše Bati ve Zlíně nebo jinými subjekty pouze ke studijním a výzkumným účelům (tedy pouze k nekomerčnímu využití), nelze výsledky bakalářské práce využít ke komerčním účelům;
- beru na vědomí, že pokud je výstupem bakalářské práce jakýkoliv softwarový produkt, považují se za součást práce rovněž i zdrojové kódy, popř. soubory, ze kterých se projekt skládá. Neodevzdání této součásti může být důvodem k neobhájení práce.

Prohlašuji,

- že jsem na bakalářské práci pracoval samostatně a použitou literaturu jsem citoval. V případě publikace výsledků budu uveden jako spoluautor.
- že odevzdaná verze bakalářské práce a verze elektronická nahraná do IS/STAG jsou totožné.

Ve Zlíně, dne

Dominik Palán, v. r.
podpis studenta

ABSTRAKT

Tato bakalářská práce se zabývá úpravou webové aplikace Věda, která je součástí webových stránek Knihovny UTB. V teoretické části jsou popsány použité technologie, informace o bibliografických databázích a metrikách. Je zde také popsána již existující nová verze webové aplikace Věda. V praktické části je popsána tvorba tabulek pomocí pluginu DataTables, vyhledávání pomocí Typeahead nebo tvorba grafů s využitím knihoven ChartJS a Vega Lite.

Klíčová slova: JavaScript, jQuery, Datatables, Treetables, ChartJS, Javascript, XML, Bibliografie, UTB, knihovna, publikace, autor

ABSTRACT

This bachelor's thesis deals with the modification of the web application called "Věda," which is part of the UTB Library website. The theoretical part describes the used technologies, information about bibliographic databases, and metrics. It also includes a description of the existing new version of the "Věda" web application. The practical part covers the creation of tables using the DataTables plugin, search functionality using Typeahead, and graph creation using ChartJS and Vega Lite libraries.

Keywords: JavaScript, jQuery, Datatables, Treetables, ChartJS, XML, Bibliography, UTB, library, publication, author.

Chtěl bych poděkovat mé vedoucí doc. Ing. Zdence Prokopové, CSc. za poskytnutí cenných rad při tvorbě teoretické části bakalářské práce. Dále bych chtěl poděkovat Ing. Ivanu Masárovi, který mi poskytoval rady při tvorbě praktické části a následně také dohlížel na správné zpracování přidaných nástrojů a funkcí. Děkuji také své rodině, která mě celou dobu studia podporovala.

OBSAH

ÚVOD	9
I TEORETICKÁ ČÁST	10
1 TECHNOLOGIE PRO TVORBU WEBOVÝCH APLIKACÍ	11
1.1 HTML.....	11
1.2 CSS.....	13
1.3 JAVASCRIPT.....	15
1.3.1 XML.....	17
1.3.2 JSON.....	18
1.4 JQUERY.....	19
1.4.1 Typeahead.....	19
1.4.2 DataTables.....	20
1.4.3 AJAX.....	21
2 BIBLIOMETRIE	23
2.1 BIBLIOMETRICKÉ DATABÁZE.....	23
2.1.1 Web of Science.....	23
2.1.2 Scopus.....	28
2.2 BIBLIOMETRICKÉ METRIKY.....	32
2.2.1 Journal Impact Factor (JIF).....	32
2.2.2 Article Influence Score (AIS).....	32
2.2.3 Scimago Journal Rank (SJR).....	32
2.2.4 CiteScore.....	33
3 SEZNÁMENÍ S ROZPRACOVANOU APLIKACÍ VĚDA	34
II PRAKTICKÁ ČÁST	37
4 SHROMÁŽDĚNÍ POŽADAVKŮ NA ROZHRANÍ	38
4.1 ZÁKLADNÍ POŽADAVKY.....	38
4.2 POŽADAVKY PRO PROJEKT.....	39
4.2.1 Graf metrik.....	39
4.2.2 Beamplot.....	39
5 NÁVRH TECHNICKÉHO ŘEŠENÍ ÚPRAV	40
5.1 VÝVOJOVÉ PROSTŘEDÍ.....	40
5.2 WEBOVÝ SERVER.....	40
5.3 WEBOVÝ PROHLÍZEČ.....	40
5.3.1 Chrome DevTools.....	40
5.4 KLIENT PRO DATABÁZI POSTGRESQL.....	41
5.5 VPN.....	41
5.6 GIT.....	41

5.7	PRŮZKUM KNIHOVEN	41
5.7.1	ChartJS	41
5.7.2	NVD3	42
5.7.3	Vega Lite	42
5.8	ZDŮVODNĚNÍ VÝBĚRU KOMPONENTŮ	42
6	IMPLEMENTACE A TESTOVÁNÍ NAVRŽENÉHO ŘEŠENÍ	44
6.1	ÚPRAVA STRÁNKY PUBLIKACE	44
6.1.1	Fazeta na filtrování autorů	44
6.1.2	Tlačítko pro export	49
6.2	ÚPRAVA STRÁNKY ČASOPISU	50
6.2.1	Graf metrik	50
6.2.2	Informace z databáze SHERPA/RoMEO	52
6.3	FILTROVÁNÍ PODLE JEL KÓDŮ	57
6.3.1	Implementace stromového zobrazení kódů	57
6.3.2	Zobrazování publikací podle kódů	60
6.3.3	Vyhledávání podle kódu	61
6.4	IMPLEMENTACE NÁSTROJE BEAMPLLOT	63
6.5	ZABEZPEČENÍ APLIKACE	72
	ZÁVĚR	73
	SEZNAM POUŽITÉ LITERATURY	74
	SEZNAM POUŽITÝCH SYMBOLŮ A ZKRATEK	75
	SEZNAM OBRÁZKŮ	77
	SEZNAM TABULEK	80
	SEZNAM PŘÍLOH	81

ÚVOD

Tato bakalářská práce se zaměřuje na rozšíření funkcionalit již rozpracované nové verze webové aplikace Věda, která slouží jako užitečný nástroj pro autory. Aplikace bude hostována na doméně veda.k.utb.cz, kde momentálně funguje stará verze, která již nedostačuje požadavkům. Nová verze aplikace je vyvíjena s využitím moderních technologií, například frameworku Bootstrap.

V teoretické části práce jsou popsány webové technologie, které byly použity při úpravě aplikace. Konkrétně jsou zde popsány například základní funkcionality technologií HTML, JavaScript a jQuery, které jsou doplněny o ukázky jejich použití. Dále jsou v práci rozebrány dvě nejčastěji používané bibliografické databáze, Web of Science a Scopus, včetně popisu metrik, které se z těchto databází vypočítávají. Poslední část teoretické části se věnovala seznámení s již rozpracovanou webovou aplikací Věda. Tato část slouží k uvedení čtenáře do kontextu a poskytnutí přehledu o existujícím stavu aplikace před zahájením praktické implementace nových funkcionalit.

V praktické části práce je nejprve popsán sběr požadavků na rozhraní. Dále jsou v práci popsány aplikace, které byly potřebné pro úspěšnou úpravu stávajícího technického řešení. Poslední a zároveň hlavní část práce se zaměřuje na implementaci a testování nových funkcionalit do webové aplikace.

Praktickým výstupem této práce je přidání nového vizualizačního nástroje Beamplot. Dále byl přidán graf metrik, který je zobrazen na stránce časopisů. Zde byl pro každý časopis přidán přehled zásad vydavatele. Na stránce autorů byla přidána vyhledávací fazeta pro filtrování publikací podle jmen autorů. Na stránce FAME bylo přidáno stromové zobrazení členění kategorií podle kódu JEL. Následně byla přidána stránka, která zobrazuje publikace podle kódu. JEL kódy byly také doplněny do vyhledávání pomocí vyhledávacího pole.

Hlavním cílem této bakalářské práce bylo přidat nové funkcionality, které by sloužily zaměstnancům UTB při získávání informací o autorech a vědeckých publikacích. Z toho důvodu práce vznikala pod dohledem zaměstnance knihovny UTB, který sledoval průběh práce, aby bylo dosaženo maximální spokojenosti s výsledkem.

I. TEORETICKÁ ČÁST

1 TECHNOLOGIE PRO TVORBU WEBOVÝCH APLIKACÍ

V následující kapitole jsou popsány technologie pro tvorbu webových aplikací, které byly použity při úpravě.

1.1 HTML

HTML (HyperText Markup Language) je značkovací jazyk, který se používá k vytváření struktury webové stránky. Obsah může být strukturován například do odstavců, tabulek nebo různých seznamů. Dále je možné vkládat také obrázky, odkazy nebo videa. Kód v HTML je tvořen pomocí značek, které se uzavírají do lomených závorek. Do nich se také vkládají atributy, které se umisťují do uvozovek. Značky se dělí do dvou skupin [1]:

Párové značky – například:

Tabulka 1. Párové značky HTML

Název značky	Popis
<code><head></head></code>	Hlavička dokumentu
<code><body></body></code>	Tělo dokumentu
<code></code>	Tučný text
<code><h1></h1></code>	Hlavní nadpis
<code><p></p></code>	Odstavec
<code><a></code>	Vložení odkazu

Nepárové značky – například:

Tabulka 2. Nepárové značky HTML

Název značky	Popis
<code></code>	Vložení obrázku
<code>
</code>	Nový řádek
<code><input></code>	Vstupní pole
<code><hr></code>	Vodorovná čára

Základní dokument HTML může být definován nepárovou značkou `<!doctype>`, která je doplněna klíčovým slovem `html`. Dále se skládá ze základních párových značek `<html>`,

<head> a <body>. Kompletní ukázka základní struktury zdrojového kódu v HTML je uvedena na Obrázku 1.

```
<!DOCTYPE html>
<html>
  <head>
    <title>Název stránky</title>
  </head>
  <body>
    <h1>Nadpis</h1>
    <p>Text odstavce</p>
  </body>
</html>
```

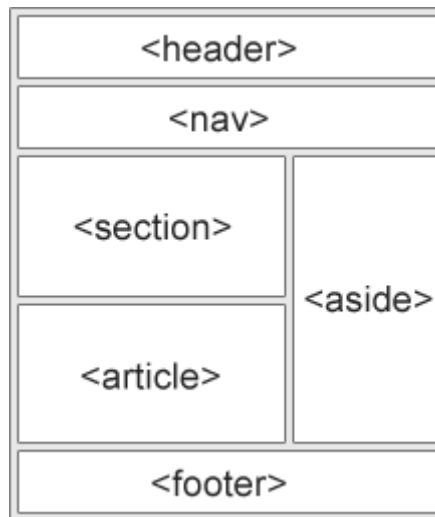
Obrázek 1. Základní struktura HTML

V současnosti je k dispozici již verze HTML5, která nabízí spoustu nových značek. Některé z nich jsou popsány v Tabulce 3 [2].

Tabulka 3. Nové značky v HTML5

Název nové značky	Popis
<audio>	Přehrává zvuk
<video>	Přehrává video
<embed>	Vykresluje externí interaktivní obsah
<canvas>	Vykresluje dynamickou bitmapovou grafiku za běhu
<time>	Reprezentuje čas

Před příchodem verze HTML5 bylo potřebné layout vytvářet pomocí značek <div>. To se s příchodem nové verze změnilo a byly vytvořeny speciální značky přímo pro layout. Konkrétně se jedná o <header> a <footer>, které představují hlavičku a patičku stránky, dále značku <nav>, ta se využívá pro navigační prvky a <section>, <article> a <aside> na tvorbu obsahu. [3]



Obrázek 2. Základní struktura v HTML5

1.2 CSS

CSS (Cascading Style Sheets) [4] je samostatný jazyk, který se užívá v kombinaci s HTML. Byl vytvořen v roce 1996 a od té doby prošel výraznou změnou. Používá se ke stylování a formátování obsahu. S CSS lze docílit také například jednotného vzhledu více dokumentů, jednoduché změny barvy pozadí a textu, stylování tlačítek a dalších grafických úprav. Lze ho tedy použít k většině vizuálních úprav u webových stránek.

Kód v CSS je tvořen pravidly, ta se skládají z tzv. selektorů a bloku deklarácí, který je obalen složenými závorkami. Na obrázku níže je vidět příklad selektoru pro třídu s názvem navigace, který obsahuje tři deklaráce zahrnuté do deklaračního bloku. Jednotlivé deklaráce přebarví text na červenou, vycentruje jej na střed a nastaví velikost fontu na 20 pixelů.

```
.navigace {  
  color: red;  
  text-align: center;  
  font-size: 20px;  
}
```

Obrázek 3. Ukázka CSS deklaračního bloku

Selektorů máme několik druhů [5]:

Univerzální selektor

- Univerzální selektor je značen hvězdičkou a vybere všechny HTML tagy v dokumentu. Lze jej využít v kombinaci s jiným názvem značky nebo elementu. To znamená, že například veškeré elementy obalené značkou `<footer>` budou nastýlovány podle jednotlivých deklarací.

```
footer * { color: red; }
```

Obrázek 4. Ukázka univerzálního selektoru

Selektor třídy

- Selektor třídy je značen pomocí tečky před názvem. Jedná se pravděpodobně o nejpoužívanější selektor, který aplikuje styly na všechny elementy se stejným názvem třídy.

```
.barva { color: red; }
```

Obrázek 5. Ukázka selektoru třídy

Selektor identifikátoru

- Selektor identifikátoru je definován pomocí mřížky, která je umístěna před názvem. Poté se jednotlivé deklarace aplikují na všechny elementy se stejnou hodnotou atributu ID.

```
#barva { color: red; }
```

Obrázek 6. Ukázka selektoru identifikátoru

Selektor elementu

- Selektor elementu vybere konkrétní element podle názvu a aplikuje styly.

```
Nav { color: red; }
```

Obrázek 7. Ukázka selektoru elementu

Vkládání stylů do HTML dokumentu

Vložit CSS kód do HTML dokumentu lze více způsoby [6]:

- Pomocí párové značky `<style>`, která může být umístěna kamkoliv v rámci hlavičky dokumentu. Mezi značky `<style>` pak zapisujeme pravidla. Tento způsob je vhodný především pro krátké zápisy.
- Pomocí atributu `style`, který můžeme zapsat přímo do elementu. Tento způsob je vhodný především pro velmi krátké zápisy. Například pro obarvení nadpisu nebo zvětšení písma.
- Pomocí externího souboru s příponou `.css`, který je nutné přilinkovat do HTML dokumentu pomocí nepárového tagu `<link>`. Ke specifikování cesty k souboru využijeme atribut `href`, do kterého vepíšeme cestu k souboru. Dále přidáme ještě atributy `rel` a `type`, které slouží k určení vztahu mezi dokumenty a typu připojovaného dokumentu. Nejčastěji se používá hodnota `stylesheet` pro atribut `rel` a `text/css` pro atribut `type`.

```
<head>
  <title>Název stránky</title>
  <link rel="stylesheet" href="style.css" type="text/css">
</head>
```

Obrázek 8. Ukázka importu CSS stylů

1.3 Javascript

JavaScript, vlastním názvem ECMAScript, je skriptovací jazyk, který běží na straně klienta a je interpretován za běhu. Byl vyvinut v roce 1996 ve firmě Netscape pod názvem LiveScript, název se ale neujal, a tak nedopatřením vznikl název JavaScript. Tento název je původně neoficiální a může být matoucí, protože jazyk nemá přímou spojitost s Javou. Později v roce 1997 po spolupráci s Microsoftem byl vytvořen standard pod názvem ECMAScript. [7]

DOM

DOM, celým názvem Document Object Model, je standard využívaný ve spojení s JavaScriptem. Jedná se o stromovou reprezentaci XML nebo HTML dokumentu, která je vytvořena při načítání stránky v prohlížeči. Pomocí JavaScriptu lze poté manipulovat s načtenými elementy v DOM, například je skrývat, mazat nebo upravovat.[7]

Struktura kódu

Kód v JavaScriptu je tvořen jednotlivými příkazy, které jsou řazeny za sebou. Tvoření kódu je podobné jako u jiných programovacích jazyků. Základem jsou proměnné, které slouží k ukládání dat do paměti. JavaScript má několik základních datových typů, například `string` pro textové hodnoty, `number` pro čísla, `boolean` pro hodnoty `true/false` a `object` pro složitější struktury, jako jsou pole.

Dříve se deklarace proměnných prováděla pomocí klíčového slova `var`, které bylo později doplněno o nové slovo `let`.

Vkládání skriptů do dokumentu

JavaScriptový kód můžeme vložit do HTML dokumentu několika způsoby [8]:

- Prvním způsobem je vkládání pomocí párové HTML značky `<script>`, do které lze následně vložit JavaScriptový kód. Značky můžeme umístit kamkoliv do dokumentu. Tato možnost ovšem není úplně vhodná, jelikož může zpomalovat načítání webové stránky.
- Pravděpodobně nejvyužívanějším způsobem je vytvoření externího souboru s příponou `.js`, který je poté načten do dokumentu pomocí značky `<script>`. Do značky je potřeba doplnit atribut `src` společně s cestou k souboru. U tohoto způsobu je také nutné podle okolností rozhodnout kam značky vložit. Nejčastěji se vkládají na konec dokumentu před ukončovací znak `</body>`. Důvod je ten, že knihovny se při načtení stránky stahují do mezipaměti a stahování na začátku by mohlo výrazně zpomalit zobrazení ostatních elementů.
- Další způsob je pomocí JavaScriptových událostí, které se umisťují do značek jako atributy. Stručný přehled nejčastěji používaných událostí je uveden v Tabulce 4 [9].

Tabulka 4. Nejpoužívanější značky v HTML5

Název atributu	Popis
onLoad	Aktivuje se při načtení stránky, nejčastěji je použit v kombinaci se značkou body.
onClick	Aktivuje se při kliknutí na prvek (např. tlačítko)
onMouseOver	Aktivuje se při najetí myši na prvek. Například zobrazení upozornění po najetí na obrázek.
onKeyPress	Pokud je element aktivní, pak se aktivuje po stisknutí jakékoli klávesy na klávesnici.

Pokud z nějakého důvodu nebude JavaScript podporován, můžeme využít párovou značku `<noscript>`. Mezi značky můžeme vložit text, který se zobrazí v případě, že JavaScript nebude dostupný nebo povolený.

1.3.1 XML

XML (Extensible Markup Language) je značkovací jazyk vyvinutý konsorciem W3C, který se využívá především k výměně dat. Jeho hlavní výhodou je možnost vlastního pojmenování značek, což by mělo usnadňovat práci s daty. Nicméně, problém může nastat, pokud jsou značky pojmenovány různě v různých systémech nebo dokumentech. To může vést k nesouladům a obtížím při zpracování dat.

Pro řešení tohoto problému byla vytvořena schémata, která popisují, jak správně pojmenovávat značky pro konkrétní problematiku. Tato schémata definují strukturu, datové typy a omezení pro správnou validaci XML dat. Například sdružení BizTalk od Microsoftu vyvinulo schémata pro podnikovou integraci a výměnu dat. [10]

Struktura XML

Každý XML dokument může obsahovat hlavičku, ve které je uvedena verze XML a typ kódování. Hlavička však není povinná a pokud se rozhodneme ji použít, musí být umístěna na začátek dokumentu. Hlavička může mít následující podobu:

```
<?xml version="1.0" encoding="UTF-8"?>
```

Poté následují vlastní značky a konkrétní struktura souboru. Značky mohou být dále doplněny o atributy, které umožňují přidat další informace k jednotlivým prvkům a usnadňují jejich filtrování a zpracování v kódu programu. [10]

```
<?xml version="1.0" encoding="UTF-8"?>
<skladObchodu>
  <polozka druh="pecivo" nazev="rohlik">
    <nazev>Rohlik</nazev>
    <pocetKusu>150</pocetKusu>
    <hmotnost>42</hmotnost>
    <cena>3.5</cena>
  </polozka>
  <polozka druh="pecivo" nazev="chleba">
    <nazev>Chleba</nazev>
    <pocetKusu>55</pocetKusu>
    <hmotnost>1200</hmotnost>
    <cena>35</cena>
  </polozka>
</skladObchodu>
```

Obrázek 9. Ukázka XML souboru

1.3.2 JSON

JSON (JavaScript Object Notation) [11] je textový formát pro výměnu dat, který není omezen pouze na použití v JavaScriptu. Je široce podporován mnoha jazyky a je snadno čitelný a zapisovatelný. Jednou z hlavních výhod je jeho jednoduchost a snadná kompatibilita s ostatními jazyky. JSON data mohou být také uložena v souborech s příponou `.json` a mohou být načítána a zpracovávána pomocí technologií jako je AJAX. JSON je často používán ve spojení s API, které poskytuje data ve formátu JSON.

Struktura JSON

Struktura formátu JSON může být opravdu libovolná. Můžeme využívat i zanoření do dalších úrovní. Jednoduchý příklad JSON struktury může vypadat například takto:

```
{ "polozky": [
  {
    "nazev": "Rohlik",
    "druh": "pecivo",
    "hmotnost": "42",
    "cena": "3.5"
  },
  {
    "nazev": "Chleba",
    "druh": "pecivo",
    "hmotnost": "1200",
    "cena": "35"
  }
] }
```

Obrázek 10. Ukázka struktury JSON

JSON v JavaScriptu

V JavaScriptu máme dvě hlavní funkce pro práci s JSON objekty [12] a to jsou:

- `JSON.parse()` – Převod řetězce JSON na JavaScript objekt
- `JSON.stringify()` – Převod JavaScript objektu na řetězec JSON

1.4 jQuery

jQuery je knihovna vyvinuta v roce 2006 [13], která je napsána v jazyce JavaScript a slouží k rozšíření možností interakce s HTML na webových stránkách. Knihovna je navrhnutá tak, aby usnadňovala interakci na webu přímo za běhu. Obsahuje spousty již naprogramovaných funkcí, které by v jazyce JavaScript obsahovali i několik desítek řádků. S pomocí jQuery můžeme snadno provádět animace bez použití technologie Flash, skrývat a zobrazovat elementy, měnit styly a texty za běhu a mnoho dalšího.

Zásuvné moduly pro jQuery

Pro knihovnu jQuery existuje mnoho rozšíření ve formě zásuvných modulů, takzvaných pluginů, které ještě více usnadňují práci s knihovnou. Například plugin Typeahead umožňuje dynamické vykreslování dat při psaní ve vyhledávacím poli. Dalším příkladem je plugin DataTables, který poskytuje pokročilé funkce pro správu a zobrazení tabulek. Kromě toho existují také pluginy pro galerie, video přehrávače, jednoduché nahrávání souborů a mnoho dalšího. Tyto zásuvné moduly můžeme importovat přímo z lokálního úložiště do svého projektu nebo využít CDN (Content Delivery Network), které slouží k rychlému a spolehlivému doručování obsahu ze vzdálených serverů.

1.4.1 Typeahead

Typeahead je plugin inspirovaný vyhledávačem na Twitteru, který umožňuje vykreslování dat při zadávání textu pomocí manipulace s DOM (Document Object Model). Jeho hlavním účelem je poskytovat našeptávač při vyhledávání. Plugin zobrazuje výsledky z tzv. datových sad, které jsou předány pluginu jako zdroje dat. Může pracovat s více datovými sadami současně. Pro vyhledávání a generování návrhů využívá plugin nástroj Bloodhound, který umožňuje efektivní manipulaci s daty a vyhledávání v nich. [14]

```
$('#typeahead').typeahead ({
  minLength:3,
  highlight:true
}),
{
  name:'nazev_datasetu',
  source:zdrojDat
});
```

Obrázek 11. Ukázka inicializace Typeahead

Na Obrázku 11 můžeme vidět základní inicializaci Typeahead. V první části jsou nastaveny parametry `minLength` pro určení minimální délky slova pro spuštění vyhledávání a `highlight` pro obalení výsledků ve značce `` pro zvýraznění.

V druhé části se specifikuje název datové sady pomocí parametru `name` a poté se předává zdroj dat (`source`). Jako zdroj dat můžeme použít různé přístupy. Můžeme přímo předat pole s daty, nebo využít inicializaci funkce Bloodhound, která je pokročilejší a umožňuje efektivnější manipulaci s daty a vyhledávání v nich.

1.4.2 DataTables

DataTables [15] je plugin, který usnadňuje vytváření a manipulaci s tabulkami. Může čerpat data z datových sad nebo přímo z HTML kódu. Data v datových sadách musí být ve formátu objektů nebo polí. DataTables zjednodušuje práci s tabulkami nabízením funkcí jako stránkování, vyhledávání a řazení dat dle sloupců. K dispozici je také zpoplatněný editor, který přidává možnosti přidávání, editace a mazání dat z tabulky.

```
$(document).ready( function () {
  $('#id_tabulky').DataTable();
} );
```

Obrázek 12. Ukázka inicializace DataTables

Parametr dom

V parametru `dom` jsou definovány prvky, které budou obsaženy v tabulce. Tyto prvky mohou zahrnovat již zmíněné funkcionality jako vyhledávání, stránkování, filtrování pomocí fazet nebo možnost ukládání do různých formátů, které budou popsány dále. Hodnoty v parametru `dom` jsou určeny kombinací malých a velkých písmen. Nejdůležitější je parametr `t`, který zobrazuje samotnou tabulku. Další parametry zahrnují `p` pro stránkování, `f` pro filtrování, a `i` pro informace o tabulce. Pořadí písmen je důležité, protože určuje pořadí prvků v tabulce.

Ukázka základního parametru `dom`:

```
dom: `lfrtip`
```

Vyhledávací fazety

Vyhledávací fazety jsou rozšířením, které není součástí přímo DataTables, a proto je potřeba jej importovat samostatně. Slouží k filtrování dat v tabulce a umožňuje filtrování pomocí několika hodnot současně. Pro zobrazení fazet je nutné do parametru `dom` přidat velké písmeno `P`. Obvykle se tento parametr přidává na začátek `dom`, aby byly fazety zobrazeny nahoře nad tabulkou.

Exportování dat

Pomocí parametru `buttons` můžeme přidávat tlačítka pro export dat. DataTables nabízí možnost exportu do formátů CSV, Excel nebo PDF. Další možností je tlačítko "Copy", které slouží ke zkopírování dat do schránky, nebo tlačítko "Print" pro zobrazení modálního okna pro tisk. Pro zobrazení těchto tlačítek v `dom` použijeme malé písmeno `b`.

1.4.3 AJAX

AJAX (Asynchronous JavaScript and XML) je asynchronní technologie, která umožňuje získávání dat ze serveru bez nutnosti znovu načítat celou stránku. Název AJAX naznačuje, že tato technologie pracuje asynchronně a umožňuje stahování i velkého množství dat na pozadí.

Implementace technologie AJAX je součástí jazyka JavaScript. V knihovně jQuery existuje funkce, která výrazně usnadňuje práci s AJAXem.

1.5 PHP

PHP (Hypertext Preprocessor) [16] je skriptovací jazyk s otevřeným zdrojovým kódem. Byl navržen pro webové aplikace a je překládán na straně serveru. Byl vyvinut v roce 1994 a v současnosti existuje již sedmá verze tohoto jazyka. PHP se používá na přibližně třech čtvrtinách všech webových stránek.

Struktura kódu v PHP je podobná jako u jiných programovacích jazyků. Základem jsou proměnné, které slouží k ukládání dat. K dispozici jsou také funkce pro manipulaci s proměnnými, jako jsou podmínky, cykly nebo přepínače (switch).

Od verze PHP 5 byla přidána vylepšená podpora pro objektově orientované programování. Tato funkčnost zahrnuje prvky jako dědičnost, rozhraní, abstraktní třídy nebo konstruktory.

Knihovny

Jazyk PHP disponuje množstvím vestavěných knihoven, které výrazně usnadňují práci. Mezi ně patří například knihovny pro generování grafů, obrázků ve formátu GIF a vykreslování geometrických tvarů, jako jsou čáry, obdélníky a kruhy. Existují také knihovny pro práci s databázemi, čtení souborů a různé výpočetní knihovny. Umí také jednoduše posílat poštovní zprávy nebo generovat PDF soubory.

2 BIBLIOMETRIE

Bibliometrie je vědecká disciplína, která se zabývá analýzou vědeckých publikací a dalších výstupů vědeckého výzkumu. Tato kapitola se zaměřuje především na nejrozšířenější bibliometrické databáze. Dále budou popsány bibliometrické metriky, které jsou využívány v těchto databázích, a budou doplněny o metriky použité v praktické části.

2.1 Bibliometrické databáze

V této části jsou popsány nejrozšířenější bibliometrické databáze, a to Web of Science a Scopus. Popis se zaměřuje na informace o těchto databázích, včetně vyhledávání a zobrazování dat.

2.1.1 Web of Science

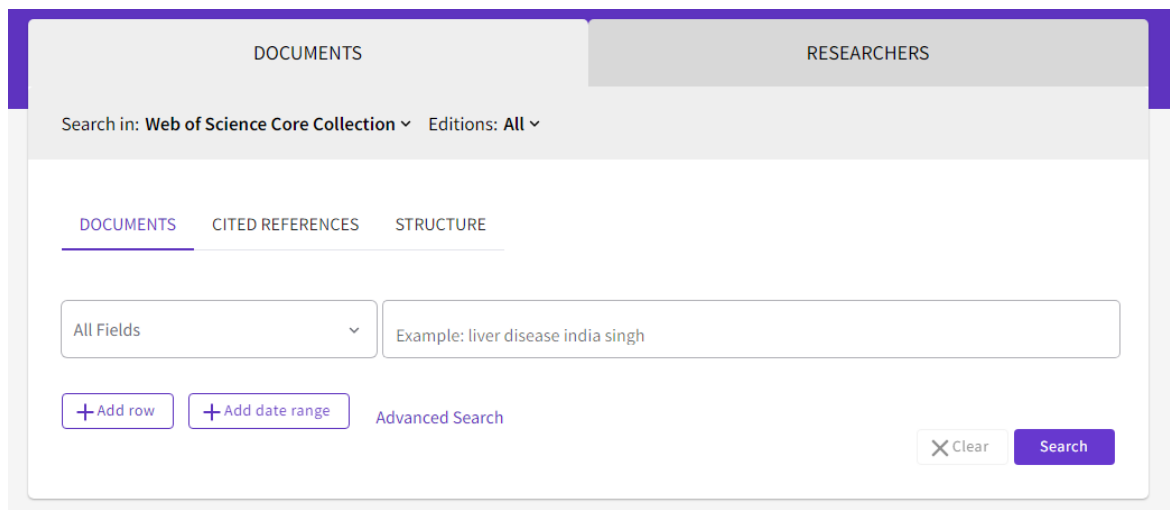
Web of Science [17] je jedna z největších a nejdůvěryhodnějších citačních databází od firmy Clarivate. Její jádro tvoří multioborová databáze s názvem Core Collection, která obsahuje více než 171 milionů publikací a důvěřuje jí přes 9 tisíc akademických, firemních a vládních institucí. Sdružuje také více než 20 milionů vědeckých pracovníků po celém světě, kteří publikují okolo 8 tisíc nových článků denně.

Databáze obsahuje širokou škálu formátů, včetně článků, knih, recenzí a technických článků. Tyto formáty jsou kategorizovány do více než 250 oborů, které zahrnují přírodní vědy, sociální vědy, humanitní vědy a umění.

Uchovávání a vyhledávání dat

V roce 1960 byl v Americe založen Institut pro vědecké informace, kde se nacházela původní papírová verze databáze Web of Science. S postupem času byla databáze přesunuta na kompaktní disky (CD) a později byla dostupná prostřednictvím serverů s webovým rozhraním.

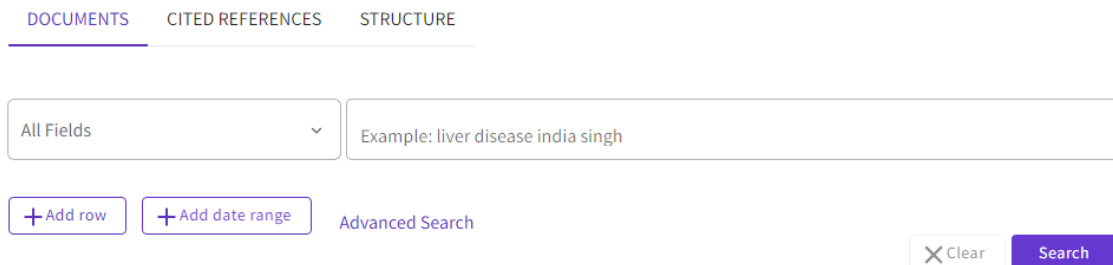
Webové rozhraní značně usnadňuje vyhledávání publikací a je možné ho lokalizovat do 10 světových jazyků, mezi které patří angličtina, španělština, ruština nebo čínština.



Obrázek 13. WoS - Vyhledávání

Vyhledávací tabulka obsahuje dvě záložky, dle kterých můžeme vyhledávat publikace nebo přímo publikující vědecké pracovníky. Záložka pro vyhledávání autorů obsahuje pouze pole pro zadání jména a příjmení. U publikací jsou celkem tři možnosti hledání.

1. Hledání podle názvu publikace



Obrázek 14. WoS - Hledání podle názvu publikace

První možností je základní hledání podle názvu, které nabízí několik filtrů. Mezi tyto filtry patří například jazyk, autor, nadpis nebo rok publikace. Je možné kombinovat filtry pomocí tlačítka umístěného níže. Tímto způsobem lze provádět detailnější a přesnější vyhledávání publikací podle konkrétních kritérií.

Tlačítko "Add row"

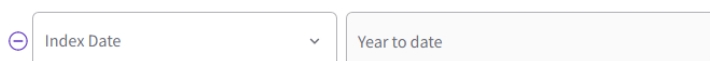
Pomocí tlačítka "Add row" lze přidat další řádek filtru, ve kterém je možné vybrat jednu ze tří možností and, or a not. Tyto možnosti lze přeložit jako obsahuje, nebo a neobsahuje.



Obrázek 15. WoS – Nový řádek přidáný pomocí tlačítka Add Row

Tlačítko "Add date range"

Zde můžeme vyhledávat publikace podle data jejich publikace nebo indexace.



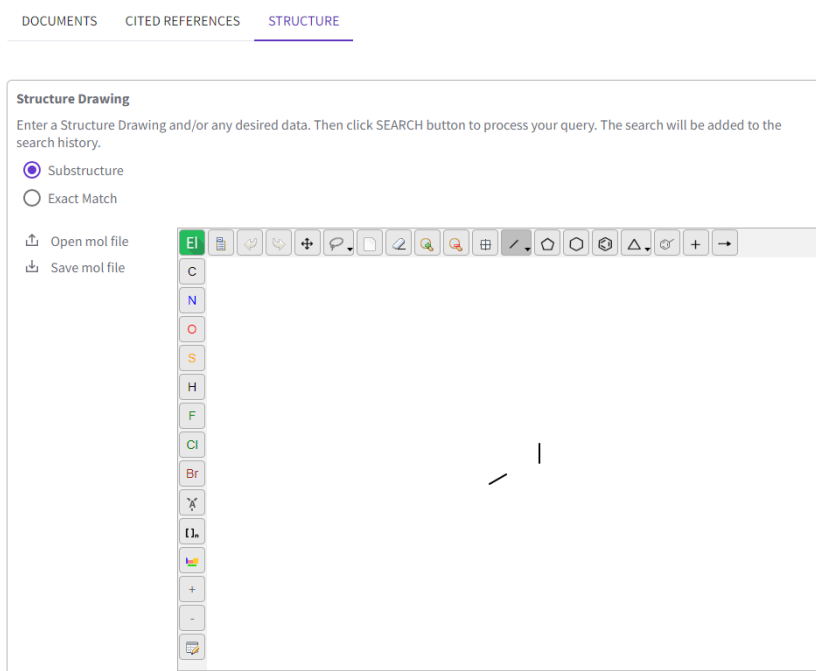
Obrázek 16. WoS – Nový řádek přidáný pomocí tlačítka Add date range

2. Hledání podle citací

Další možností je hledání publikací podle citací. Zde je k dispozici filtr, který umožňuje vyhledávat podle roku, autora nebo názvu. K dispozici jsou také dvě tlačítka, která byla popsána výše.

3. Hledání podle struktury

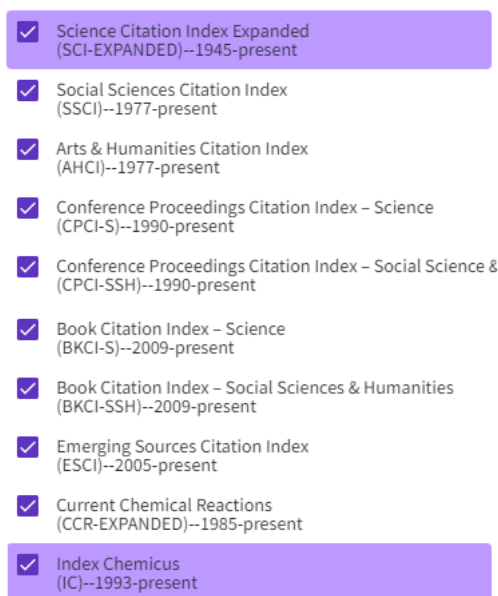
Hledání podle struktury nám umožňuje vyhledávat pomocí kreslicího nástroje, který můžeme využít například pro vyhledávání chemických sloučenin.



Obrázek 17. WoS – Hledání podle struktury

Vyhledávací indexy

U databáze Web of Science také probíhá vyhledávání pomocí 10 indexů, které lze vybrat v nabídce Editions. Seznam indexů je uveden na Obrázku 18.



Obrázek 18. WoS – Vyhledávací indexy

Zobrazení výsledků

Po provedení vyhledávání zadaného výrazu a použití filtrů se zobrazí stránka s výsledky. Zde můžeme také použít další filtry nebo například exportovat výsledky do různých formátů, jako je Excel. Výsledky lze zobrazit také ve formě citačního reportu nebo analýzy výsledků. Na Obrázku 19 a 20 jsou zobrazeny formáty výsledků po zadání výrazu "Tomas Bata University".

Clarivate English Products

Web of Science Search Marked List History Alerts Dominik Palán

Results for tomas bata univ... Analyze Results: tomas bata university (All Fields)

6,289 results from Web of Science Core Collection for:

Q tomas bata university (All Fields) Analyze Results Citation Report Create Alert

Copy query link

Publications You may also like...

Refine results

Search within results for...

Quick Filters

- Highly Cited Papers 15
- Hot Papers 1
- Review Articles 83
- Early Access 36
- Open Access 2,121
- Associated Data 39

Publication Years

- 2022 81
- 2021 401
- 2020 375
- 2019 572
- 2018 548

0/6,289 Add To Marked List Export

Sort by: Relevance 1 of 126

1 The Historical Heritage of **Tomas Bata** as an Incitement for Application of Creativity in Marketing
 Proceedings of the 3rd International Conference on Strategic Innovative Marketing (IC-SIM) 2015 | PROCEEDINGS OF THE 3RD INTERNATIONAL CONFERENCE ON STRATEGIC INNOVATIVE MARKETING (IC-SIM 2014) 175, pp.342-349 4 References

2 ENHANCING LEARNING OF MULTIMEDIA SUBJECTS AT SECONDARY SCHOOLS AND FACULTY OF HUMANITIES, **TOMAS BATA UNIVERSITY** IN ZLIN, CZECH REPUBLIC
 10TH International Conference on Education and New Learning Technologies (EDULEARN) 2018 | EDULEARN18: 10TH INTERNATIONAL CONFERENCE ON EDUCATION AND NEW LEARNING TECHNOLOGIES, pp.6407-6416 27 References

Obrázek 19. WoS – Zobrazení výsledků

Publications 6,289 Total From 1945 to 2022

Citing Articles 37,579 Total 35,005 Without self-citations

Times Cited 48,476 Total 41,036 Without self-citations

72 H-Index Average per item 7.71

Times Cited and Publications Over Time

Publications Citations

Year	Publications	Citations
2000	0	0
2001	0	0
2002	0	0
2003	0	0
2004	0	0
2005	0	0
2006	0	0
2007	200	0
2008	150	0
2009	180	0
2010	220	0
2011	350	0
2012	300	0
2013	350	0
2014	350	0
2015	600	1000
2016	620	2000
2017	700	3500
2018	550	5000
2019	580	6500
2020	380	7500
2021	400	9000
2022	80	1500

6,289 Publications Sort by: Citations: highest first 1 of 126

	2018	2019	2020	2021	2022	Average per year	Total
Total	4,953	6,385	8,072	9,861	2,009	2,308.38	48,476
Worldwide trends in body-mass index, underweight, overweight, and obesity from 1975 to 2016: a pooled analysis of 2416 population-based measurement studies in 128.9 million children, adolescents, and adults	263	595	865	953	194	479.83	2,879

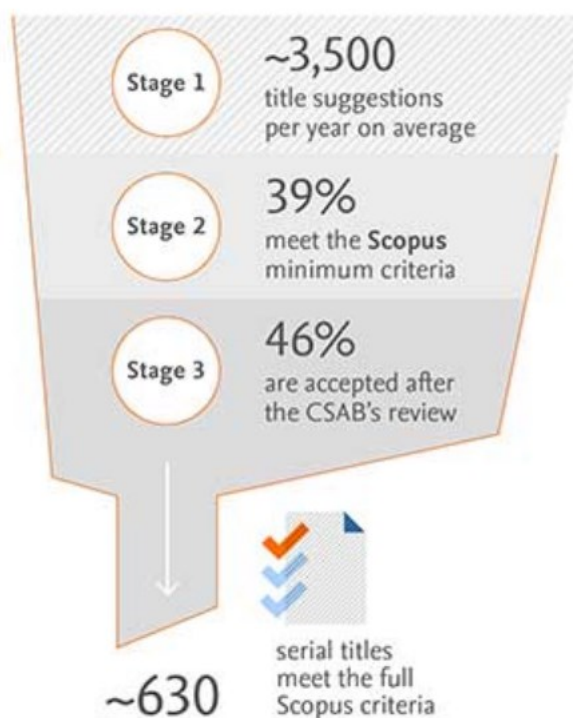
Obrázek 20. WoS – Zobrazení výsledků

2.1.2 Scopus

Scopus [18] je důvěryhodnou databází od firmy Elsevier, do které přispívá více než 7000 vydavatelů zahrnujících 340 oborů. Jedná se o relativně novou databázi, která obsahuje více než 82 milionů publikací a 1.7 miliardy citovaných referencí. Nachází se zde publikace od více než 17 milionů autorů z celého světa. Databáze zahrnuje formáty jako knižní série, časopisy nebo sborníky.

Přidávání nových publikací

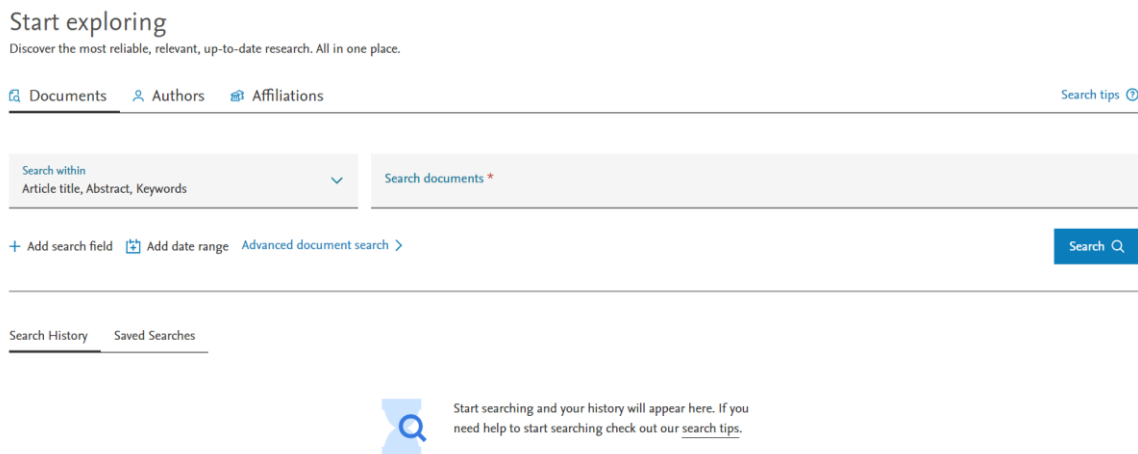
Scopus má poměrně přísná kritéria. Do databáze je navrženo pro každý rok přibližně 3500 titulů z který je schváleno pouze kolem 600. Detailní počty přidávaných publikací jsou na Obrázku 21.



Obrázek 21. Scopus – Kritéria přidávání publikací

Vyhledávání

Ve Scopusu máme k dispozici vyhledávací formulář, který je rozdělen na tři záložky. Na první záložce je vyhledávání podle publikací, kde se nachází rozbalovací seznam s bližšími kategoriemi, jako jsou autoři, publikace, klíčová slova nebo ISSN. Vedle seznamu se nachází samotné vyhledávací pole.

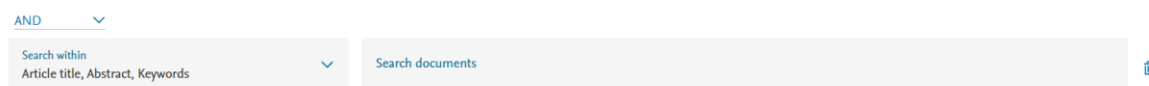


Obrázek 22. Scopus – Vyhledávání

Nachází se zde také dvě tlačítka, která rozšiřují možnosti filtrování.

Tlačítko - "Add search field"

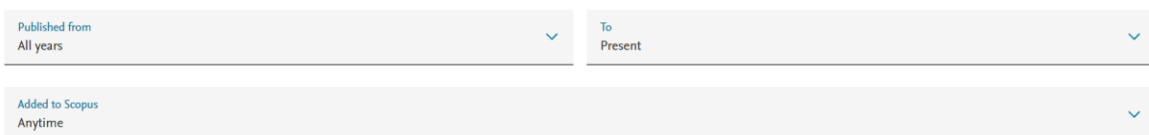
Tlačítko "Add search field" přidává další řádek, kde máme na výběr logické operátory and, or a not, spolu s rozbalovacím seznamem obsahujícím další vyhledávací pole.



Obrázek 23. Scopus – Nový řádek přidáný pomocí tlačítka Add search field

Tlačítko – "Add date range"

Tlačítko "Add date range" umožňuje přidat filtrování podle rozsahu roků, ve kterých byla publikace zveřejněna a také rozsahu roků, ve kterých byla přidána do databáze Scopus.



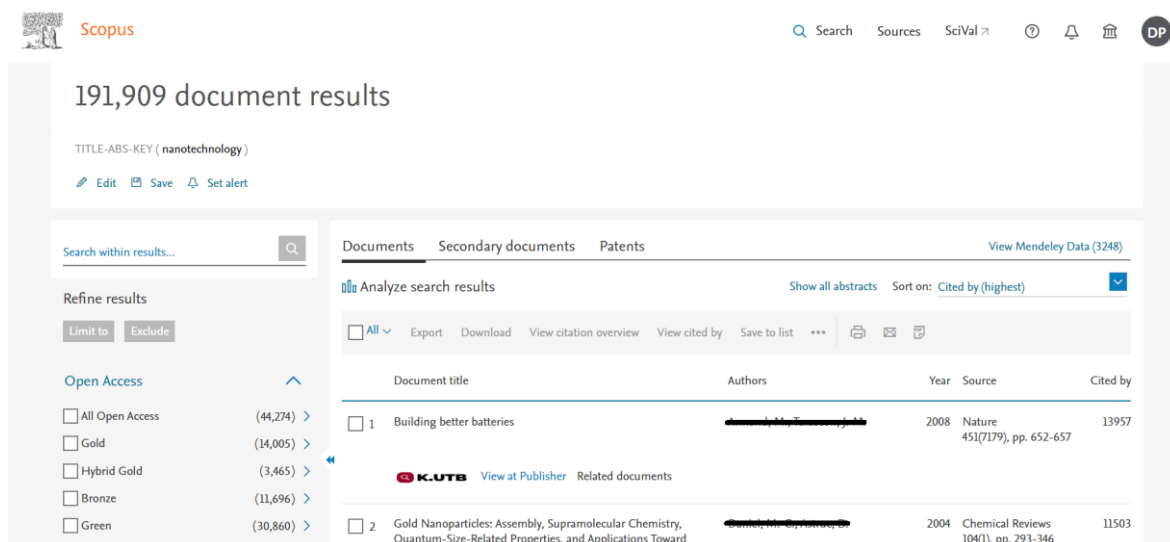
Obrázek 24. Scopus – Nový řádek přidáný pomocí tlačítka Add date range

Další možností vyhledávání je podle autora. Zde můžeme vyhledávat pomocí jména a příjmení autora nebo unikátního kódu ORCID. Máme také možnost přidat filtrování podle instituce, za kterou autor publikuje. Poslední možností je vyhledávání podle instituce, kde je k dispozici pole pro zadání názvu instituce.

Výhodou je také možnost procházení historie hledání a uložených hledání, které lze nalézt v dolní části formuláře.

Výsledky vyhledávání

Po provedení vyhledávání je zobrazen seznam výsledků, kde máme k dispozici různé možnosti filtrování a exportu do různých formátů. K dispozici jsou také podrobnější informace o jednotlivých publikacích, které lze zobrazit po kliknutí na jejich název.



The screenshot shows the Scopus search results page for the query 'TITLE-ABS-KEY (nanotechnology)'. The page displays 191,909 document results. The search filters are set to 'Documents', 'Secondary documents', and 'Patents'. The results are sorted by 'Cited by (highest)'. The first two results are:

Document title	Authors	Year	Source	Cited by
1 Building better batteries	[Redacted]	2008	Nature 451(7179), pp. 652-657	13957
2 Gold Nanoparticles: Assembly, Supramolecular Chemistry, Quantum-Size-Related Properties, and Applications Toward	[Redacted]	2004	Chemical Reviews 104(1), pp. 293-346	11503

Obrázek 25. Scopus – Zobrazení výsledků

Document type
Review

Source type
Journal

ISSN
14764679

DOI
10.1038/451652a

[View more](#) ▾

Building better batteries

[Save all to author list](#)

[Save all to author list](#)

^a LRCS, CNRS UMR-6007, Université de Picardie Jules Verne, Amiens, France

13 957 ^{99th percentile}
Citations in Scopus

13,6
FWCI [?]

3 482
Views count [?] ↗

[View all metrics](#) >

[Full text options](#) ▾ [Export](#)

Indexed keywords

SciVal Topics

Chemicals and CAS Registry
Numbers

Metrics

Indexed keywords ▾

SciVal Topics [?] ▾

Chemicals and CAS Registry Numbers ▾

Metrics ▾

References (28)

[View in search results format](#) >

Obrázek 26. Scopus – Zobrazení výsledků

2.2 Bibliometrické metriky

V následující části jsou popsány metriky bibliometrických databází.

2.2.1 Journal Impact Factor (JIF)

Impact Factor je jednou z nejznámějších metrik, která se počítá z databáze Web of Science. Používá se k vyjádření vlivu časopisu a vypočítává se tak, že se počet citací za konkrétní rok dělí počtem článků publikovaných v tomto časopise za předchozí dva roky.

Nevýhodou této metriky je, že menšinový podíl, který je citován vícekrát, má větší vliv na hodnocení než většinový podíl, který je citován méně. Dále není vhodné srovnávat Impact Factor mezi různými obory, protože některé obory jsou obecně více citované než jiné.

$$JIF = \frac{\text{počet citací v konkrétním roce}}{\text{počet článků za předchozí dva roky}}$$

2.2.2 Article Influence Score (AIS)

Article Influence Score (AIS) je další metrikou počítanou z databáze Web of Science. Jedná se o průměrný vliv všech článků v časopise za posledních pět let. Průměrné skóre je nastaveno na hodnotu 1,00. Pokud skóre časopisu vyjde menší než 1,00, znamená to, že články v daném časopise mají podprůměrný vliv. Naopak, pokud skóre vyjde vyšší než 1,00, pak články v časopise mají nadprůměrný vliv.

$$AIS = \frac{0.01 * \text{EigenFactor Score}}{\text{počet článku v časopise za 5 let vydělený} \\ \text{počtem článků ve všech časopisech za 5 let}}$$

2.2.3 Scimago Journal Rank (SJR)

Scimago Journal Rank (SJR) je jednou z nejpoužívanějších metrik, která je počítána z databáze Scopus. Výpočet SJR je podobný jako u AIS, avšak s tím rozdílem, že se bere v úvahu průměrný počet vážených citací obdržných v konkrétním roce, který je následně vydělen počtem článků publikovaných v předchozích třech letech.

Jedním z významných rozdílů mezi SJR a AIS je, že SJR zohledňuje také prestiž časopisu. To znamená, že citace z časopisů s vyšší prestiží budou mít větší váhu než citace z časopisů s nižší prestiží.

2.2.4 CiteScore

CiteScore je další z metrik, která je vypočítávána pro databázi Scopus. Její výpočet je podobný Impact Factoru a provádí se jednou za rok, vždy následující rok v létě. Pro výpočet CiteScore se vezme počet citací z konkrétního roku a vydělí se počtem článků publikovaných za předchozí tři roky.

$$CiteScore = \frac{\text{počet citací v konkrétním roce}}{\text{počet článků za předchozí tři roky}}$$

3 SEZNÁMENÍ S ROZPRACOVANOU APLIKACÍ VĚDA

Tato kapitola popisuje již rozpracovanou webovou aplikaci s názvem Věda, která dosud nebyla uveřejněna. Aplikace bude umístěna na subdoméně `veda.k.utb.cz`, na které se momentálně nachází stará verze. Souběžně s touto starou verzí je vyvíjena nová rozpracovaná verze, která využívá technologie HTML a CSS pro zobrazení a stylování obsahu.

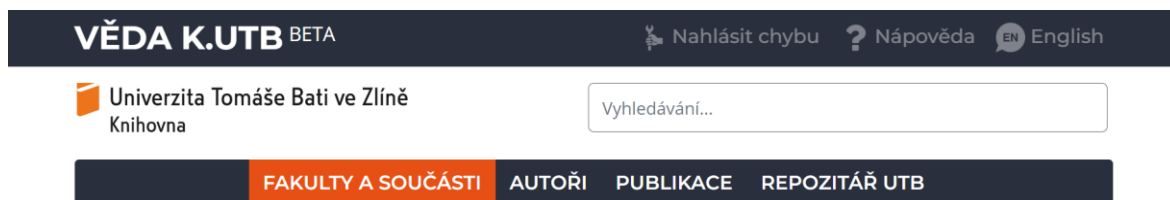
Na serverové straně aplikace je používána technologie PHP, která zajišťuje zpracování dat z databáze a serverů API.

Na klientské straně aplikace je využit JavaScript, který zajišťuje interaktivitu mezi uživatelem a webovou aplikací. JavaScript umožňuje interakci s uživatelem, validaci dat, dynamické načítání obsahu a další funkcionality.

Webový portál Věda slouží k prezentaci informací o autorech a publikacích, stejně jako informací o fakultách UTB a vědeckých časopisech. Tyto informace jsou zobrazovány pomocí přehledných tabulek, které využívají knihovnu DataTables. Tabulky umožňují rychlé a interaktivní vyhledávání, třídění a filtrování dat, což zajišťuje uživatelům pohodlný přístup k informacím.

Kromě tabulek je na portálu Věda také využívána knihovna ChartJS pro generování grafů. Tato knihovna umožňuje vizuálně atraktivní a interaktivní zobrazení dat prostřednictvím různých typů grafů, jako jsou sloupcové grafy, koláčové grafy nebo linkové grafy.

Rozpracovaná nová verze se skládá ze tří částí. První část je hlavička stránky, která je umístěna v souboru `header.html`. V této části se nachází logo, vyhledávací formulář a kompletní menu, které je rozděleno na čtyři položky: Autoři, Publikace, Fakulty a Repozitář UTB.



Obrázek 27. Hlavička stránky

Další část je patička stránky, která je rovněž oddělena a nachází se v souboru `footer.html`. V této části se nachází například kompletní seznam fakult s odkazy nebo tlačítka pro sociální média.



Obrázek 28. Patička stránky

Poslední částí je samotný obsah jednotlivých stránek, do kterých jsou importovány předchozí dvě části.

Stránka fakulty a součásti

V této části se nachází seznam všech fakult, který je doplněn o univerzitní institut a knihovnu. Při rozkliknutí jednotlivých fakult získáme přístup k informacím o autorech a publikacích pocházejících z dané instituce.

Stránka autoři

Na stránce autoři je k dispozici přehledná tabulka se seznamem všech autorů v databázi UTB. Tato tabulka je vytvořena pomocí pluginu DataTables. Po kliknutí na jméno konkrétního autora je uživatel přesměrován na jeho kompletní profil, kde najde podrobnější informace a údaje o jeho publikační činnosti.

Stránka publikace

Na stránce publikace je k dispozici tabulka se všemi publikacemi, která je vytvořena také pomocí pluginu DataTables. Tabulka nabízí rozsáhlé vyhledávání pomocí fazetového okna, které umožňuje filtrovat publikace podle různých kritérií, jako je typ dokumentu, datum vydání nebo název fakulty. Dále je možné kliknout na název publikace a zobrazit si její podrobnosti.

Repozitář UTB

Poslední sekce v menu je repozitář UTB, která přesměruje uživatele na web publikace.k.utb.cz. Na této stránce se nachází bibliografie publikační činnosti autorů působících na UTB.

Stránka vědeckých časopisů

Webová aplikace rovněž zahrnuje prozatím skrytou sekci, která obsahuje seznam vědeckých časopisů. V této sekci je možné vyhledávat časopisy podle názvu nebo kódu ISSN. Po vyhledání konkrétního časopisu je uživatel přesměrován na stránku s podrobnými informacemi o daném vědeckém časopisu.

II. PRAKTICKÁ ČÁST

4 SHROMÁŽDĚNÍ POŽADAVKŮ NA ROZHRAŇÍ

4.1 Základní požadavky

Před výběrem vhodné JavaScriptové knihovny pro zobrazení dat formou grafu je nezbytné stanovit základní požadavky, které musí být splněny.

Open Source

Prvním kritériem je, že knihovna musí mít otevřený zdrojový kód. Existuje několik důvodů pro tuto volbu. Prvním z nich je finanční stránka, protože open source knihovny jsou zdarma a neobsahují žádné uzamčené nebo placené funkce. Dalším důvodem je transparentnost, jelikož každý může zkontrolovat kód knihovny a hledat případné chyby. Tím se zvyšuje bezpečnost, protože chyby mohou být identifikovány a opraveny komunitou vývojářů.

Přehledná dokumentace

Dalším kritériem je kvalitní dokumentace, která by měla být přehledná a měla by obsahovat všechny důležité informace o knihovně. Je důležité, aby dokumentace poskytovala jasný popis jednotlivých funkcí knihovny a byla doplněna o příklady použití, které pomohou lépe porozumět funkčnosti. Kvalitní dokumentace umožňuje vývojářům rychleji a efektivněji pracovat s knihovnou a získat potřebné informace bez zbytečného hledání nebo odhadování.

Potřebné funkce

Dalším důležitým kritériem je zohlednění požadavků a potřeb projektu. Je důležité provést předem důkladný průzkum dostupných knihoven a zjistit, zda nabízejí funkcionalitu, která odpovídá požadavkům projektu. Tímto přístupem lze předejít situaci, kdy se v průběhu projektu zjistí, že vybraná knihovna nedokáže splnit požadavky a je nutné přejít na jinou knihovnu. Předem určené požadavky a provedený průzkum umožňují zvolit nejvhodnější knihovnu, která bude schopna poskytnout potřebnou funkcionalitu a usnadnit vývoj projektu.

Pravidelné aktualizace

Posledním kritériem je pravidelné aktualizování knihovny. Je důležité zjistit, zda je knihovna pravidelně aktualizována vývojáři. Aktualizace zahrnují opravy chyb, zlepšení výkonu a bezpečnostní aktualizace. Pravidelné aktualizace naznačují, že knihovna je aktivně udržována a podporována vývojáři, což je důležité pro dlouhodobou udržitelnost a spolehlivost projektu.

4.2 Požadavky pro projekt

V této kapitole budeme stanovovat konkrétní požadavky na knihovny, které budou následně použity v projektu.

4.2.1 Graf metrik

Graf metrik bude zobrazen na stránce časopisu a měl by znázorňovat hodnoty pěti vypočtených metrik pro každý časopis. Na ose x budou vypsány jednotlivé roky a na ose y bude zobrazena hodnota metriky. Pro tuto funkcionalitu je potřeba vybrat knihovnu, která umožňuje zobrazení čárového grafu s body odpovídajícími jednotlivým hodnotám. Důležitým kritériem je také přítomnost legendy a informačního okna, které se zobrazí při najetí myši na bod. V informačním okně by měly být zobrazeny rok, název a hodnota metriky. Další výhodou je možnost přiřadit bodům barvu odpovídající čáře, na které leží.

4.2.2 Beamplot

Beamplot je graf založený na publikacích a citačních datech. Na ose x budou zobrazeny procentní hodnoty a na ose y jednotlivé roky. V grafu budou umístěny jednotlivé body na linkách. Dále budou zobrazeny mediány za jednotlivé roky a celkový medián. Pro tuto funkcionalitu je potřeba vybrat knihovnu, která umožňuje práci s více vrstvami. Kromě vodorovných linek a bodů musí knihovna také podporovat zobrazení horizontální linky pro celkový medián a kruhových bodů pro mediány za jednotlivé roky. V informačním okně by měly být zobrazeny název publikace, procentní hodnota, kategorie a rok.

5 NÁVRH TECHNICKÉHO ŘEŠENÍ ÚPRAV

Dalším krokem k úpravě již rozpracované webové aplikace Věda bylo se seznámit s použitými technologiemi. Stránka byla vytvořena přehledně a jednoduše pomocí technologií HTML, PHP a JavaScript. K úpravě jsme tedy potřebovali vhodné vývojové prostředí a webový server, který dokáže zpracovat PHP. Jelikož databáze byla umístěna na serveru, ke kterému byl omezen přístup pouze na vnitřní školní síť, bylo nezbytné využít školní VPN.

5.1 Vývojové prostředí

Pro úpravu již rozpracované webové aplikace bylo použito IDE (Integrated Development Environment) Visual Studio Code od společnosti Microsoft. Zmiňované IDE je zdarma a je možné jej stáhnout z oficiálních stránek code.visualstudio.com. Byla použita momentálně nejnovější verze 1.66.1. Toto prostředí je přehledné a zvládá všechny požadované technologie, které byly potřebné. Obsahuje také mnoho užitečných pluginů, například Prettier, který umí automaticky formátovat zdrojový kód.

5.2 Webový server

Webový server byl zprovozněn pomocí programu XAMPP ve verzi v3.3.0. Tento program je šířen pod licencí GNU GPL. Byl vybrán převážně z důvodu předchozích zkušeností a své jednoduchosti. Stáhnutí proběhlo ze stránek [apachefriends.org](https://www.apachefriends.org). Konkrétně byl použit webový server Apache ve verzi 2.4.

5.3 Webový prohlížeč

Nevyhnutelným programem byl také webový prohlížeč, který byl použit ve verzi 100. Konkrétně se jednalo o Google Chrome od společnosti Google. Při připojování k webovému serveru Apache jsme využívali doménu localhost nebo IP adresu 127.0.0.1.

5.3.1 Chrome DevTools

Hojně využit byl také nástroj pro vývojáře, který je možné zobrazit pomocí klávesové zkratky CTRL + SHIFT + I. Využita byla převážně konzole, která může posloužit pro detekování chyb a zobrazování testovacích výpisů například při volání API. Dále byl použit debugger na záložce Source, kde jsme mohli testovat jednotlivé soubory z projektu.

5.4 Klient pro databázi PostgreSQL

Pro přístup k databázi byl použit program HeidiSQL ve verzi 11.3. Jde o bezplatný open-source nástroj, který je distribuován pod licencí GPLv2. Výhodou této aplikace je přehledné a uživatelsky přívětivé rozhraní pro připojení k databázím. K připojení bylo nutné znát IP adresu databáze, název konkrétní databáze, a samozřejmě uživatelské jméno a heslo.

5.5 VPN

Jelikož databáze běží na serveru UTB, bylo nutné se z odkazu localhost připojit k této databázi. K tomuto účelu byl použit VPN klient. Konkrétně byl využit klient s názvem Cisco AnyConnect Secure Mobility Client ve verzi 4.7. Připojení bylo realizováno prostřednictvím domény vpn.utb.cz a použití školních přihlašovacích údajů.

5.6 Git

Pro správu verzí webové aplikace je používán software s názvem Gitea, který je šířen pod licencí MIT. Aktuální verze Gitea je GO1.18 a přístup k němu je zajištěn pomocí domény git.k.utb.cz. Druhou možností komunikace, která byla využita, je propojení s vývojovým prostředím (IDE). Toto propojení bylo realizováno přímo ve Visual Studio Code pomocí rozšíření s názvem GitHub Pull Request and Issues. Po instalaci rozšíření se na bočním panelu objevilo tlačítko pro práci s verzovacím systémem Git, které umožňovalo sledovat změny a nahrávat je přímo na Git.

5.7 Průzkum knihoven

Všechny uvedené knihovny, které byly vyzkoušeny, splňují základní kritérium, kterým je otevřený zdrojový kód. Při jejich výběru jsme se také zaměřili na jejich popularitu, jelikož nejpoužívanější grafy mají větší komunitu a je snazší nalézt řešení případných problémů.

5.7.1 ChartJS

První knihovna, která byla zkoumána, je ChartJS. Byla vybrána z důvodu jejího již existujícího použití ve stávající verzi webové aplikace. Během prozkoumávání přehledné dokumentace jsme zjistili, že knihovna je velmi jednoduchá a stručná. Definice grafu se provádí pomocí funkce `Chart`, která přijímá parametry `type`, `data` a `options`. Parametr `type` slouží k určení typu grafu. Mezi základní typy grafů, které lze vybrat, patří čárový, sloupcový a bodový graf. Parametr `data` slouží k definici zdrojových dat, která lze přímo

vložit pomocí proměnné. Posledním parametrem je `options`, který umožňuje základní úpravy, například nastavení os, barev grafu nebo legendy. Jednou z nevýhod knihovny je omezená vlastní přizpůsobitelnost, například vytváření vlastních vrstev, které jsou potřebné pro Beamplot. Nicméně knihovna je pravidelně aktualizována, což znamená, že případné chyby budou v budoucnu opraveny bez větších problémů.

5.7.2 NVD3

Druhou vybranou knihovnou bylo NVD3. Původním záměrem bylo najít knihovnu, která by umožňovala provádět pokročilé úpravy grafů. Avšak, ve srovnání s ChartJS, měla knihovna NVD3 poněkud omezenější dokumentaci, avšak stále dostatečnou. Zjištěno bylo, že knihovna NVD3 nebyla aktualizována od roku 2014, což nevyhovuje našim požadavkům. Proto jsme se rozhodli tuto knihovnu nepoužívat.

5.7.3 Vega Lite

Třetí vybranou knihovnou byla Vega Lite. Tato knihovna disponuje přehlednou dokumentací, která obsahuje mnoho ukázek a detailních popisů. Základní struktura pro tvorbu grafů v knihovně Vega Lite se skládá z parametrů `data`, `mark` a `encoding`. V parametru `data` je definován zdroj dat. U knihovny Vega Lite je možné staticky zapsat data pomocí vnořeného parametru `values`. Data je také možné předat ve formátech JSON nebo CSV. Knihovna také podporuje kombinování vrstev grafů pomocí parametru `mark`. V parametru `encoding` je možné určit, která data budou zobrazena na které ose.

Při bližším prozkoumání jsme zjistili, že knihovna Vega Lite nabízí široké možnosti úprav grafů. Obsahuje funkce pro transformaci dat, základní výpočty a také umožňuje použití podmínek. Pro zobrazení grafů lze využít přidruženou knihovnu Vega Embed, kde stačí definovat cílové ID elementu a proměnnou obsahující definici dat pro graf.

5.8 Zdůvodnění výběru komponentů

Po zhodnocení výhod a nevýhod jsme se rozhodli zvolit knihovnu ChartJS pro graf metrik. Důvodem bylo, že všechny grafy na stránce časopisů jsou vytvořené pomocí této knihovny. Použitím stejné knihovny dosáhneme tedy konzistentního vzhledu s ostatními grafy. Nicméně pro graf Beamplot jsme zjistili, že ChartJS není dostatečně flexibilní, protože nedokáže implementovat více vrstev do jednoho grafu, což je pro Beamplot velký

nedostatek. Po důkladném zvážení jsme se tedy rozhodli pro knihovnu Vega Lite. Díky její schopnosti transformace dat a rozsáhlým funkcím je pro nás vhodnější a lépe využitelná.

6 IMPLEMENTACE A TESTOVÁNÍ NAVRŽENÉHO ŘEŠENÍ

Následující kapitola popisuje nově přidané funkce do webové aplikace Věda. Jedná se především o doplňkové grafy a nástroje, které budou pomáhat autorům v lepší orientaci. Všechny části byly testovány přímo při vývoji a následně také zaměstnancem knihovny UTB při předávání. Předávací protokol je přiložen v Příloze č. 1.

6.1 Úprava stránky publikace

Jednou z prvních úprav byla úprava na stránce publikací, kde jsou všechny publikace zobrazeny spolu s jejich autory, typem dokumentu, jménem fakulty a rokem vydání. Na této stránce je možné filtrovat výsledky pomocí horních fazet a vyhledávacího pole pod fazetami. Ukázku původní stránky před úpravou můžeme vidět na Obrázku 29.

The screenshot shows a web application interface for a publication list. At the top, there are three filter panels: 'Typ dokumentu' (Document Type) with 10 items, 'Vydáno' (Year) with 6 items, and 'Fakulta' (Faculty) with 9 items. Below the filters is a search bar with the text 'např. polymer, liquid crystals or Lehocky'. The main content area is a table with columns: 'Název' (Title), 'Autoři' (Authors), 'Typ dokumentu' (Document Type), 'Vydáno' (Year), 'Fakulta' (Faculty), and 'Výzkumná témata' (Research Topics). Two rows of publications are visible.

Typ dokumentu	Vydáno	Fakulta
Kapitola v knize (438)	2000 (6)	FAI (2869)
Odborná kniha (187)	2001 (36)	FAME (2042)
Patent (3)	2002 (57)	FHS (621)
Prototyp (43)	2003 (103)	FLKR (291)
Příspěvek ve sborníku (3619)	2004 (139)	
Review (116)	2005 (172)	

Název	Autoři	Typ dokumentu	Vydáno	Fakulta	Výzkumná témata
Social media adoption in SMEs sustainability: Evidence from an emerging economy	Amoah, John; Bruce, Emmanuel; Shurong, Zhao; Bankuoru Egala, Sulemana; Kwarteng, Kofi	article	2023	FAME	social media, technology adoption, SME, sustainability, TOE
Atomic Fe on hierarchically ordered porous carbon towards high-performance Lithium-sulfur batteries	Chen, Xinan; Zhu, Zhengju; Vargun, Elif; Li, Yiwen; Sáha, Petr; Cheng, Qilin	article	2023	UNI	lithium-sulfur batteries, single atom catalyst, polysulfide conversion, hierarchically porous carbon, electrochemistry

Obrázek 29. Stránka publikace před úpravou

Úkolem bylo doplnit stránku o fazetu autora ve stejném stylu jako již existující fazety. Dále mělo být přidáno tlačítko pro export seznamu publikací do formátu CSV.

6.1.1 Fazeta na filtrování autorů

Stránka publikace je vytvořena pomocí pluginu s názvem DataTables. Data do tabulky jsou dodávána z databáze UTB ve formátu JSON, který je stahován pomocí PHP. Fazety jsou přidávány pomocí pluginu SearchPanes, který přebírá data ze sloupců v tabulce.

V první řadě bylo zapotřebí vyřešit, jak stahovat data pro fazetu, aby byla data korektně naformátována. V tabulce publikací můžeme vidět, že jedna publikace může mít více autorů. Tato informace již byla přebírána z databáze jako celek, ale nebylo možné ji rozdělit na jednotlivé autory. Nebylo tedy možné použít stejný způsob zdroje dat jako u ostatních fazet, kde se data přebírají přímo z jednotlivých sloupců. Řešením mohlo být stahování seznamu autorů přímo z API, jako se to děje na stránce autorů. Avšak vyskytl se problém, protože do tabulky nebylo možné přidat další zdroj dat, konkrétně tedy seznam autorů ve formátu JSON, který by se zobrazoval pouze ve fazetě autora. Jedinou možností bylo upravit zdroj dat publikací již při stahování pomocí PHP a v tabulce použít funkci orthogonálního vykreslování, která umožní zobrazovat původní data před úpravou ve vlastnosti `render`.

Na následujícím obrázku můžeme vidět formát jedné publikace, která je stahována z databáze.

```
▼ [0 ... 99]
▼ 0:
  authors: "{\"Gautam, Shweta\" , \"Lapčík, Lubomír\" , \"Lapčíková, Barbora\" , \"Gál, Robert\"}"
  faculty: "{\"Faculty of Technology\"}"
  handle: "10563/1011457"
  internal_authors: "{\"Gautam, Shweta\" , \"Lapčík, Lubomír\" , \"Lapčíková, Barbora\" , \"Gál, Robert\"}"
  issued: "{2023}"
  subjects: "{\"emulsion coatings\" , rheology , \"phase separation\" , \"thermal analysis\" , \"antimicrobial agents\" , meat}"
  title: "{\"Emulsion-based coatings for preservation of meat and related products\"}"
  type: "{review}"
▶ [[Prototype]]: Object
```

Obrázek 30. Ukázka jedné položky před úpravou `authors` a `internal_authors`

Konkrétně zde jsou položky `authors` a `internal_authors`, které obsahují seznam autorů, jež bylo potřebné rozdělit. K tomuto účelu byla vytvořena jednoduchá funkce s názvem `get_formated_authors`, která odstraňuje nepotřebné znaky z textového řetězce a následně je rozděluje do pole dle jmen autorů.

```
function get_formated_authors($data){
    $symbols = array('{', '}', '"');
    $authorsData = str_replace('"', '', $data);
    $authorsData = str_replace($symbols, "", $authorsData);
    $authorsArr = explode(";", $authorsData);
    return $authorsArr;
}
```

Obrázek 31. Funkce `get_formated_authors`

Funkce `get_formated_authors` má jeden vstupní parametr, kterým je seznam autorů v textové podobě uložený v položce `authors`. Jako výstup vrací pole s jednotlivými autory, které je poté vloženo zpět do položky `authors`. Výsledek můžeme vidět na Obrázku 32.

```

▼ [0 ... 99]
▼ 0:
  ▼ authors: Array(7)
    0: "Asabuwa Ngwabebhoh, Fahanwi"
    1: "Zandraa, Oyunchimeg"
    2: "Sáha, Tomáš"
    3: "Stejskal, Jaroslav"
    4: "Kopecký, Dušan"
    5: "Trchová, Miroslava"
    6: "Pfleger, Jiří"
    length: 7
    ▶ [[Prototype]]: Array(0)
    faculty: "{\\University Institute\\"}
    handle: "10563/1011504"
  ▼ internal_authors: Array(4)
    0: "Asabuwa Ngwabebhoh, Fahanwi"
    1: "Zandraa, Oyunchimeg"
    2: "Sáha, Tomáš"
    3: "Stejskal, Jaroslav"
    length: 4
    ▶ [[Prototype]]: Array(0)
  issued: "{2023}"
  subjects: "{polypyrrole,poly(N-vinylpyrrolidone),\\conducting leather\\,\\sheet resistance\\,\\bending tests\\,\\organic dyes\\"}"
  title: "{\\Coating of leather with dye-containing antibacterial and conducting polypyrrole\\"}"
  type: "{article}"
  ▶ [[Prototype]]: Object

```

Obrázek 32. Ukázka jedné položky po rozdělení `authors` a `internal_authors`

V souboru s inicializací tabulky byla upravena vlastnost `searchPanels`, která obsahuje inicializaci fazet. Konkrétně byly upraveny položky `targets` a `layout`.

- **Targets** – Zajišťuje propojení fazet s jednotlivými sloupci. Bylo přidáno číslo 1, což přidává propojení se sloupcem autora. Pořadí je taktéž důležité, jelikož určuje pořadí fazet na stránce.
- **Layout** – Určuje rozložení fazet. Původní hodnota `columns-3` byla upravena na `columns-4`.

```
searchPanes: {
  cascadePanes: true,
  viewTotal: true,
  targets: [1, 2, 3, 4],
  layout: 'columns-4',
},
dom: 'PfrtBip',
columnDefs: [
  {
    searchPanes: {
      show: true,
      cascadePanes: true,
    },
    targets: [1]
  },
],
```

Obrázek 33. Ukázka z kódu inicializace tabulky

V části `ColumnDefs`, která slouží k individuálnímu nastavení sloupců fazet, byla následně přidána vlastnost `searchPanes` pro fazetu autora. Parametr `show` slouží k zobrazení fazety a `cascadePanes` umožňuje propojení vyhledávání současně s ostatními fazetami. Za vlastností `searchPanes` se nachází hodnota `targets`, která propojuje nastavení sloupce se sloupcem autorů.

```
{
  "data": "authors",
  "render": {
    "_": function (data, type, row) {
      var authors = data;
      var internalAuthors = row.internal_authors;

      var authorsString = "";

      for (var i = 0; i < authors.length; i++) {
        var author = authors[i];
        if (internalAuthors.indexOf(author) > -1)
          authorsString += '<a href="author.html?id='
            + author + '>' + author + '</a>';
        else
          authorsString +=
            '<a style="color:#292F3C" href="publications.html?author='
            + author + '>' + author + '</a>';

        if (i !== authors.length - 1)
          authorsString += '<b>;</b> ';
      }

      return authorsString;
    },
    sp: '[]'
  },
  "searchPanes": {
    orthogonal: 'sp',
  }
},
```

Obrázek 34. Nastavení orthogonálního zobrazování dat ve fazetě autora.

Následně byla přidána vlastnost `sp` do funkce pro vykreslování sloupce, která definuje, že data budou vykreslena do fazety v původním, nepozměněném formátu, tak jak byla stažena a upravena v PHP. Vlastnost `orthogonal` v části `searchPanes` je použita k propojení vlastnosti `sp` přímo s fazetou autora. Na Obrázku 35 je zobrazena funkční fazeta autora.



Autoři	
Pross, M.	1
Provazník, Ivo	11
Provazník, Michal	1
Pruba, M.	1
Prucková, Zdeňka	13
Prudil, Luděk	2

Obrázek 35. Fazeta autora

6.1.2 Tlačítko pro export

Dalším požadavkem na stránce publikací bylo umožnit export publikovaných dat do formátu CSV. K tomu byla využita vestavěná funkce v pluginu DataTables, která poskytuje možnost exportu do různých formátů.

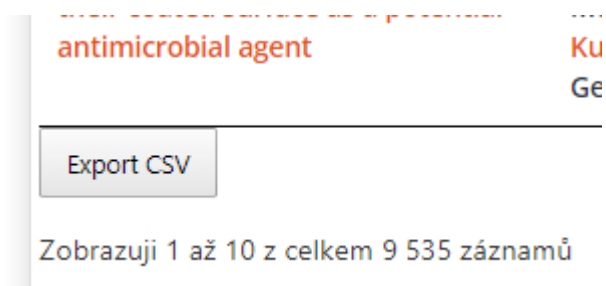
V inicializační funkci tabulky byla přidána sekce `buttons`, ve které byly nastaveny vlastnosti `extend`, `bom` a `text`.

- `extend` – Určuje, do kterých formátů bude možné exportovat.
- `text` – Specifikuje název tlačítka pro export.
- `bom` – Pomáhá správně naformátovat text s diakritikou.

```
dom: 'PfrtBip',
buttons: [
  {
    extend: 'csv',
    bom: true,
    text: 'Export CSV',
  }
],
```

Obrázek 36. Sekce `buttons` pro export do CSV

Pro umístění tlačítka bylo nutné upravit parametr dom, kde bylo přidáno velké písmeno B. Toto písmeno je umístěno mezi malým t, které slouží k zobrazení tabulky, a i, které zobrazuje počet zobrazených záznamů. Tímto nastavením je tlačítko pro export umístěno na levou stranu přímo pod tabulkou.



Obrázek 37. Tlačítko pro export do CSV

6.2 Úprava stránky časopisu

Stránka časopisu zatím není veřejně přístupná, ale nabízí funkce pro vyhledávání časopisů a zobrazování informací o nich. Úkolem bylo přidat graf metrik a informace o zásadách vydavatele z databáze SHERPA/RoMEO.

6.2.1 Graf metrik

Na stránce časopisů byl přidán vizualizační nástroj nazvaný graf metrik, který slouží k zobrazení vývoje určitých metrik souvisejících s daným časopisem v průběhu času. Tento graf zobrazuje nejpoužívanější metriky, jako je JCR, SNIP, SJR, IPP a CiteScore. Data pro tyto metriky jsou získávána pomocí již existujícího API serveru, který je napojen na databázi UTB.

V JavaScriptovém souboru s grafy byla přidána funkce `drawMetricsChart`, která má dva vstupní parametry. Prvním parametrem jsou data určená pro vykreslení grafu a druhým parametrem je místo, kam má být graf umístěn. Dále byla přidána funkce `getData`, která přijímá typ metriky jako vstupní parametr a následně vybírá pouze hodnoty, které se vztahují k této konkrétní metrice.

```
function getData(type) {  
  var newData=[];  
  data.forEach(item => {  
    if(item.source==type){  
      newData.push({ x: item.year, y: item.value });  
    }  
  });  
  return newData;  
}
```

Obrázek 38. Funkce `getData` pro výběr dat jednotlivých metrik

Následně byl inicializován graf, do kterého byla data rozdělena podle metrik, takže pro každou metriku byla vytvořena vlastní linka. Na Obrázku 39 je vidět jedna z pěti položek v datasetu.

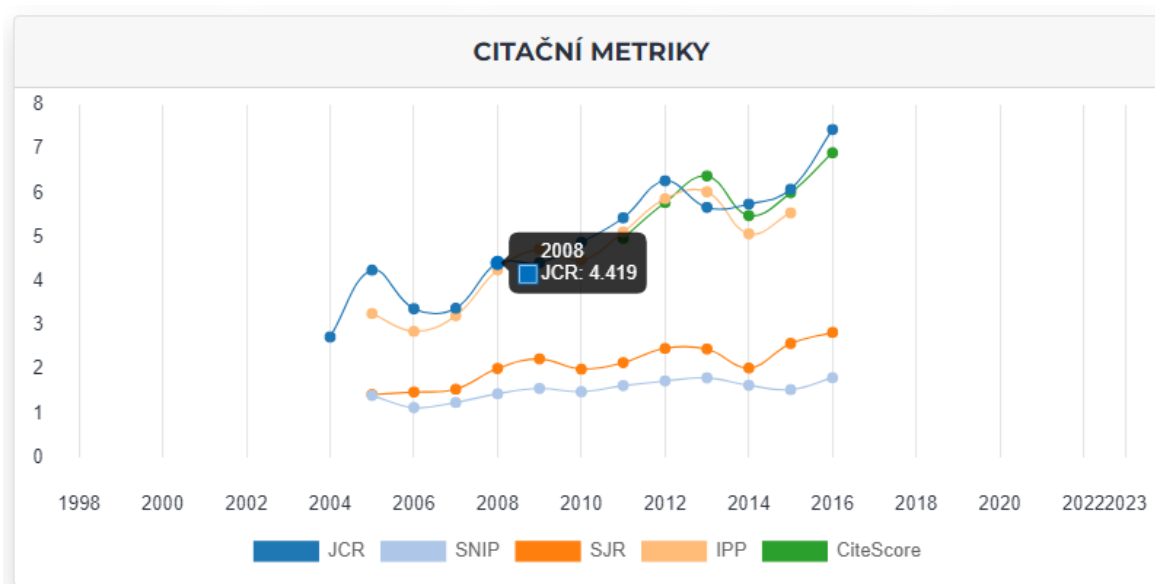
```
datasets: [  
  {  
    data: getData("JCR"),  
    label: "JCR",  
    backgroundColor: "#1F77B4",  
    borderColor: "#1F77B4",  
    fill: false,  
    borderWidth: 1,  
    showLine: true,  
  },  
]
```

Obrázek 39. Ukázka jedné položky v datasetu

Každá položka má svůj vlastní zdroj dat, který získáváme pomocí funkce `getData`. Této funkci je předáván vstupní parametr s názvem metriky, který slouží k filtrování dat a vrácení pouze hodnot relevantních pro danou metriku. Dále je možné nastavit název metriky, barvu a šířku linek.

Další důležitou vlastností je `scales`, ve které definujeme osy X a Y pro náš graf. Na ose X jsou umístěny jednotlivé roky od roku 1998 do současnosti, zatímco na ose Y jsou umístěny hodnoty metrik pro daný časopis. Aktuální rok je získáván pomocí funkce `getFullYear`, která vrací aktuální rok a tato hodnota je předána do vlastnosti `max` osy X. Kromě toho byla přidána nápověda, která se zobrazí při najetí myši na body v grafu. V této nápovědě jsou zobrazeny informace o konkrétním roce, názvu metriky a hodnotě metriky pro daný rok.

Pro zobrazení grafu byl použit prvek canvas, který slouží k vykreslování grafických dat. Prvek má nastavené ID `journalMetricsGraph`, které je předáno do funkce pro inicializaci grafu. Tabulka kolem grafu byla stylována pomocí tříd poskytovaných frameworkem Bootstrap, aby odpovídala vzhledu již existujících prvků na stránce. Výsledný graf je zobrazen na Obrázku 40.



Obrázek 40. Graf citačních metrik

6.2.2 Informace z databáze SHERPA/RoMEO

Úkolem bylo zajistit stahování z databáze SHERPA/RoMEO do databáze UTB a následně zobrazovat zásady vydavatele na stránce časopisu.

6.2.2.1 *Stahování dat a vkládání do databáze*

Pro stahování dat z databáze SHERPA/ROMEO bylo zapotřebí získat API klíč. Klíč byl zdarma dostupný po registraci. Využito bylo API s názvem Object Retrieval API, které má parametry popsané v Tabulce 5.

Tabulka 5. Parametry pro Object Retrieval API ze SHERPA/ROMEO

Název	Popis	Povinné
Item-type	Typ objektu	Ano
Api-key	Klíč API	Ano
Format	Formát výstupu	Ano
Order	Pořadí záznamů	Ne
Limit	Limit počtu záznamů, maximální je 100 položek	Ne
Offset	Slouží tomu, abychom při každém stahování nezačínali zase od začátku. Například když bude offset nastaven na 100, tak API začne stahovat až 101 položku	Ne
filter	Pole, které obsahuje podmínky filtru	Ne

Stahování dat je zajištěno funkcí `get_sherpa_api_page` v PHP souboru, která využívá metodu `fetch`. Při stahování používáme povinný parametr `item-type`, do kterého předáváme hodnotu `publication`, a nastavujeme formát na JSON. Dále předáváme parametr `api-key`, kam vkládáme API klíč. Limit je nastaven na maximálně 100 položek, a proto používáme `offset` také nastavený na 100. Stažená data jsou dále předávána do funkce `romeo_insert_json`, která je následně zpracovává.

Funkce `romeo_insert_json` má za úkol rozdělit data na jednotlivé části. Přijímá jako vstupní parametr JSON obsahující 100 položek, které následně rozdělí a uloží do databáze pod odpovídajícím ID, které je převzato z databáze SHERPA/RoMEO. Data jednotlivých položek jsou ukládána do atributu `data` v databázi ve formátu JSON.

```
function romeo_insert_json($jsonArr) {
    global $veda_dsn, $veda_user, $veda_pass, $veda_opt;

    $pdo = new PDO($veda_dsn, $veda_user, $veda_pass, $veda_opt);

    try {
        $stmt = $pdo->beginTransaction();

        $stmt = $pdo->prepare("INSERT INTO sherpa.romeo_raw (id, data) VALUES (:id, :json);");
        foreach ($jsonArr as $jsonItem) {
            if($jsonItem!=null){
                $json = json_encode($jsonItem);
                $stmt->bindParam(":id", $jsonItem['id'], PDO::PARAM_INT);
                $stmt->bindParam(":json", $json, PDO::PARAM_STR);
                $stmt->execute();
            }
        }
        $rows = $stmt->fetchAll();
        if (count($rows) == 1) {
        } else {
            echo "<div class='box-error'>Error inserting</div>";
            exit;
        }

        $stmt = $pdo->commit();
    } catch (PDOException $e) {
        $message = 'Invalid query: ' . $e->getMessage() . "<br>\n";
        $message .= 'Whole query: ';
        die("<div class='box-error'>$message</div>\n");
    }
}
```

Obrázek 41. Funkce romeo_insert_json, která slouží k rozdělení dat

Jelikož databáze SHERPA/RoMEO omezuje počet položek na stránku na 100, bylo potřebné implementovat automatickou obnovu stránky po vložení každých 100 položek do databáze. K tomu jsme vytvořili proměnné `limit` a `offset`, které získávají aktuální hodnotu offsetu pomocí proměnné `$_GET`. Poté nastavíme nový offset pro příští požadavek a voláme funkci `get_sherpa_api_page`, která nám poskytne 100 položek z databáze. Ty jsou následně předány funkci `romeo_insert_json`, která je zpracuje a uloží do databáze. Pomocí metody `header` obnovíme stránku s novým offsetem a stáhneme dalších 100 položek. Tento postup se opakuje, dokud nejsou stažena všechna data, a script je poté ukončen.

sherpa.romeo_raw: 28 400 řádků celkem,

662	{"system_metadata":{"date_created":"2010-07-14 10:58:54","publicly_visible_phras...	publication	2022-04-04
663	{"system_metadata":{"uri":"https://v2.sherpa.ac.uk/id/publication/663","publ...	publication	2022-04-04
664	{"id":664,"issns":[{"issn":"1367-0751","type_phrases":[{"value":"print","language":"...	publication	2022-04-04
667	{"system_metadata":{"uri":"https://v2.sherpa.ac.uk/id/publication/667","id":6...	publication	2022-04-04
669	{"publisher_policy":{"internal_moniker":"Policy F","urls":{"description":"Policy F"...	publication	2022-04-04
674	{"type_phrases":[{"phrase":"Journal","value":"journal","language":"en"}],"publishe...	publication	2022-04-04
677	{"type":"journal","listed_in_doaj":"no","system_metadata":{"id":677,"publicly_visib...	publication	2022-04-04
679	{"title":{"language_phrases":[{"value":"en","language":"en","phrase":"English"}],"l...	publication	2022-04-04
681	{"listed_in_doaj_phrases":[{"value":"no","language":"en","phrase":"No"}],"issns":[{"...	publication	2022-04-04
682	{"system_metadata":{"date_created":"2010-07-14 11:07:35","publicly_visible_phras...	publication	2022-04-04
683	{"system_metadata":{"date_modified":"2020-01-29 15:25:35","uri":"https://v2.sh...	publication	2022-04-04
684	{"listed_in_doaj":"no","type":"journal","system_metadata":{"date_modified":"2020...	publication	2022-04-04
685	{"type":"journal","listed_in_doaj":"no","system_metadata":{"id":685,"publicly_visib...	publication	2022-04-04
687	{"system_metadata":{"date_created":"2010-07-14 11:16:20","publicly_visible_phras...	publication	2022-04-04

Obrázek 42. Ukázka dat uložených v databázi

6.2.2.2 Zobrazování dat z databáze

Data, která budou zobrazována, jsou zásady vydavatele rozdělené na předloženou verzi, přijatou verzi a zveřejněnou verzi. Pro jejich zobrazení musel být vybrán plugin, který umožňuje zobrazit každou z těchto verzí odděleně. Jako vhodný plugin byl zvolen jQuery UI, který poskytuje možnost vytvoření tzv. efektu harmoniky neboli skládacího obsahu.




Nejprve byl vytvořen PHP soubor pro stahování dat z databáze. V tomto souboru byly implementovány dvě funkce. První funkce se nazývá `get_id` a má jeden vstupní parametr, kterým je ISSN identifikátor. Tato funkce vyhledá řádek, ve kterém se daný identifikátor nachází, a vrátí ID tohoto řádku. Získané ID je poté předáváno do druhé funkce s názvem `get_json_by_id`. Tato funkce stáhne a vrátí celý řádek ve formátu JSON na základě poskytnutého ID.

Následně je PHP soubor volán pomocí funkce `POST`, ve které jsou vybírána data pro zobrazení zásad vydavatele. Tato data jsou následně rozdělena do dvou proměnných `header` a `body`. V proměnné `header` je specifikován název obsahu, například `Published Version`, `Submitted Version` nebo `Accepted Version`. V proměnné `body` jsou uloženy konkrétní položky zásad vydavatele, které jsou vypsány níže.







- OA Fee
- OA Publishing
- Embargo
- Licence

- Publisher Deposit
- Location
- Conditions

Jelikož ne všechny verze zahrnují všechny tyto položky, byly prázdné položky skryty. Poté byly nastaveny vhodné ikony, velikost textu a styl tabulky. Pro rozbalený skládací obsah byla vybrána ikona mínus a pro zabalení ikona plus. Dále byl nastaven parametr `collapsible` na hodnotu `true`, čímž se zajistilo, že při načtení stránky bude obsah skrytý. Proměnné body a header byly přiřazeny k prvku s ID `publisher_policy` pomocí metody `append`, a následně byl inicializován skládací obsah. Výsledek je zobrazen na Obrázcích 43 a 44.

– Submitted Version	
Name	Description
 Embargo	No Embargo
 Location	Author's Homepage Preprint Repository
 Conditions	Published source must be acknowledged Must link to publisher version with DOI
+ Accepted Version	
+ Accepted Version	
+ Published Version	

Obrázek 43. Zásady vydavatele – Submitted Version

+ Submitted Version	
+ Accepted Version	
+ Accepted Version	
– Published Version	
Name	Description
 OA Fee	Yes
 OA Publishing	Yes
 Embargo	No Embargo
 Licence	CC BY undefined
 Publisher Deposit	PubMed Central
 Location	Any Website Institutional Repository Named Repository Subject Repository Journal Website

Obrázek 44. Zásady vydavatele – Published Version

6.3 Filtrování podle JEL kódů

JEL kódy jsou standardizovaný systém kategorizace, který se využívá v ekonomických publikacích. Úkolem bylo přidat na stránku Fakulty managementu a ekonomiky stromové zobrazení těchto kódů. Následně také přidat tyto kódy do vyhledávání a vytvořit stránku, která zobrazí publikace v konkrétní kategorii.

6.3.1 Implementace stromového zobrazení kódů

Jelikož JEL kódy jsou hierarchicky strukturované, bylo nutné použít tabulku ve formátu stromového zobrazení. Nejprve bylo zváženo využití pluginu dataTables, ale tento plugin bohužel neumožňuje vykreslování dat ve formě stromového zobrazení. Proto byl přidán plugin treeTables, který slouží jako rozšíření pro dataTables a umožňuje práci s daty ve stromovém formátu.

```
1 <?xml version="1.0"?>
2 <data>
3   <classification level="1" language="English">
4     <code>A</code>
5     <description><![CDATA[General Economics and Teaching]]></description>
6   <classification level="2" language="English">
7     <code>A1</code>
8     <description><![CDATA[General Economics]]></description>
9   <classification level="3" language="English">
10    <code>A10</code>
11    <description><![CDATA[General]]></description>
12  </classification>
13  <classification level="3" language="English">
14    <code>A11</code>
15    <description><![CDATA[Role of Economics &bull; Role of Economists &bull; Market for
16  </classification>
17  <classification level="3" language="English">
18    <code>A12</code>
19    <description><![CDATA[Relation of Economics to Other Disciplines]]></description>
20  </classification>
21  <classification level="3" language="English">
22    <code>A13</code>
23    <description><![CDATA[Relation of Economics to Social Values]]></description>
24  </classification>
25  <classification level="3" language="English">
26    <code>A14</code>
27    <description><![CDATA[Sociology of Economics]]></description>
28  </classification>
29  <classification level="3" language="English">
30    <code>A19</code>
31    <description><![CDATA[Other]]></description>
32  </classification>
33 </classification>
```

Obrázek 45. Ukázka XML s JEL kódy

Data byla získána z oficiálních stránek Journal of Economic Literature ve formátu XML. K zpracování těchto dat byla vytvořena funkce s názvem `createJsonForJel`, která přijímá data ve formátu XML jako vstupní parametr. Pomocí technologie AJAX jsou data načítána z XML souboru a předávána do této funkce. Data jsou strukturována podle "levelů", přičemž vyšší číslo levelu vždy závisí na menším čísle. V rámci funkce jsou data transformována do odpovídajícího formátu nezbytného pro správné vykreslení tabulky z pluginu `treeTable`.

Formát pluginu `treeTable` je pevně stanovený a zahrnuje tyto parametry:

- `tt_key` – Identifikační číslo řádku
- `tt_parent` – Číslo nadřazeného řádku
- `name` – Jméno kategorie

Formát byl rozšířen o parametr `code`, který slouží k ukládání JEL kódu odděleně od názvu.

Zobrazování je zajištěno pomocí HTML tabulky, která obsahuje dvě položky – kód a název. Tabulka je stylována pomocí Bootstrap frameworku, aby se co nejlépe hodila do existujícího řešení. Kromě toho je do ní přidán prvek s ID `onlyFame`, který zajistí, že se tabulka zobrazí pouze na dané fakultě. Podmínka, která ovlivňuje zobrazení, je zobrazena na Obrázku 46.

```
if(facultyId!="fame"){  
    $('#onlyFame').hide();  
}
```

Obrázek 46. Podmínka pro skrytí tabulky na ostatních fakultách

Při vytváření podmínky byla využita existující proměnná `facultyId`, která obsahuje zkratku fakulty, na které se aktuálně nacházíme. Pokud stránka nemá ID `fame`, tabulka se nezobrazí.

Data pro tabulku jsou sestavována pomocí funkce `showTable`, která má jeden vstupní parametr pro předání již naformátovaných dat. Parametry pluginu `treeTable` jsou popsány pod následujícím obrázkem.

```
function showTable(data){
  $('#codesTable').treeTable({
    "data": data,
    "collapsed": true,
    "columns": [
      {
        "data": "code",
      },
      {
        "data": "name",
      }
    ],
    "order": [[1, 'asc']],
    "language": {
      "url": "https://cdn.datatables.net/plug-ins/1.10.20/i18n/Czech.json"
    },
  });
}
```

Obrázek 47. Funkce showTable pro zobrazení tabulky

- `data` – Zdroj dat, v daném formátu.
- `collapsed` – Definuje, zda má být stromové zobrazení rozbaleno nebo zabaleno. Nastaveno na `false`, aby tabulka při načtení stránky nezabírala moc místa.
- `columns` – Rozdělení dat do sloupců. Máme dva sloupce. První s názvem kód do kterého předáváme data z položky `code` a druhý sloupec název, do kterého předáváme data z položky `name`.
- `order` – V tomto parametru nastavujeme řazení tabulky. Parametr má dvě hodnoty. První hodnotou je číslo sloupce, dle kterého bude řazena tabulka a druhá hodnota určuje vzestupné řazení podle abecedy.
- `language` - Parametr, který nám zajišťuje překlad tabulky do češtiny.

Na Obrázku 48 je zobrazená výsledná tabulka. Většina parametrů je již povolena v základním nastavení, a proto můžeme vidět tabulku s doplněnými základními funkcemi. V levém horním rohu se nachází výběr počtu záznamů, které mají být zobrazeny na stránce a v pravém je vyhledávací pole. V levém dolním rohu je uveden počet záznamů zobrazených na stránce a v pravém je stránkování. Názvy kategorií jsou doplněny o odkaz, který slouží k přesunu na stránku kategorie.

Zobraz záznamů 10 Hledat:

Kod	Název
▼ A	General Economics and Teaching
- A1	General Economics
- A2	Economic Education and Teaching of Economics
- A3	Collective Works
▶ B	History of Economic Thought, Methodology, and Heterodox Approaches
▶ C	Mathematical and Quantitative Methods
▶ D	Microeconomics
▶ E	Macroeconomics and Monetary Economics
▶ F	International Economics
▶ G	Financial Economics

Zobrazuji 1 až 10 z celkem 23 záznamů (filtrováno z celkem 159 záznamů) Předchozí 1 2 3 Další

Obrázek 48. Tabulka s JEL kódy na stránce FAME

6.3.2 Zobrazování publikací podle kódů

Pro stránku kategorie byl vytvořen HTML soubor, který pomocí funkce POST přijímá parametr a zobrazuje pouze publikace se stejným kódem. Pro zobrazení publikací byla využita tabulka z pluginu DataTables a byla inicializována pomocí funkce `createJelCategoryTable`. V době vytváření souboru nebyla k dispozici skutečná data s kódy, které by mohly být zobrazeny, takže byla vytvořena proměnná s testovacími daty. V příkladu, který je uveden níže, jsou zobrazena data v kategorii A1. Za sloupcem "Výzkumná témata" se nachází skrytý sloupec, který definuje kategorii, ve které se daná publikace nachází.

SEZNAM PUBLIKACÍ V KATEGORII A1

Hledat:

Název	Autoři	Typ dokumentu	Vydáno	Fakulta	Výzkumná témata
Název publikace	Příjmení, Jméno	article	2022	FAME	text

Zobrazuji 1 až 1 z celkem 1 záznamů Předchozí 1 Další

Obrázek 49. Ukázka tabulky pro kategorii A1

6.3.3 Vyhledávání podle kódu

Vyhledávání je umístěno na každé stránce v pravém horním rohu a je vytvořeno pomocí pluginu Typeahead. Je doplněno nástrojem pro návrhy s názvem Bloodhound. V původním nastavení vyhledávání probíhalo pouze pro autory. Úkolem bylo rozšířit ho o vyhledávání kódů pro JEL kategorie.

Původní nastavení umožňovalo zobrazovat deset autorů v nápovědě vyhledávání. Bylo tedy nutné přidat omezení na pět položek, aby se uvolnil prostor pro JEL kódy.

Následně byla vytvořena funkce `getCodesJelFromXML`, která pomocí funkce AJAX převádí data z XML do požadovaného formátu. Formát je rozdělen následovně:

- `name` – Jméno kategorie
- `code` – kód kategorie
- `type` – statický text "jel", který odděluje data od autorů.

Vytvoření Bloodhound návrhu zahrnuje následující parametry:

- `datumTokenizer` – Povinný parametr, který rozděluje data na pole slov.
- `queryTokenizer` – Povinný parametr, který odkazuje na vytvořený dotaz.
- `limit` – Nastavuje limit výsledků, které se zobrazí.
- `local` – Předávání dat, pomocí funkce `getCodesJelFromXML`.

```
var codes = new Bloodhound({
  datumTokenizer: Bloodhound.tokenizers.obj.whitespace('name'),
  queryTokenizer: Bloodhound.tokenizers.whitespace,
  limit: 5,
  local: getCodesJelFromXml()
});
codes.initialize();
```

Obrázek 50. Bloodhound návrh

Do inicializace Typeahead byla následně přidána sekce pro JEL kódy, která obsahuje následující parametry:

- `name` – Název datové sady.
- `displayKey` – Určuje, ze kterého parametru budeme brát data pro nápovědu.
- `source` – Zdroj dat. `ttAdapter` je klíčová metoda, která spouští vyhledávání.
- `templates` – návrh vzhledu.
 - `header` – Zde určujeme styl a text nadpisu sekce vyhledávání.
 - `suggestion` – Pomocí funkce přidáváme ke každému datu odkaz, který umožňuje přechod na stránku kategorie.

```
{
  name: 'codes',
  displayKey: 'name',
  source: codes.ttAdapter(),
  templates: {
    header: '<h3 class="header">JEL Codes</h3>',
    suggestion: function (data) {
      return '<a href="jel-category.html?category=' + data.code + '>' + data.name + '</a>';
    }
  }
}
```

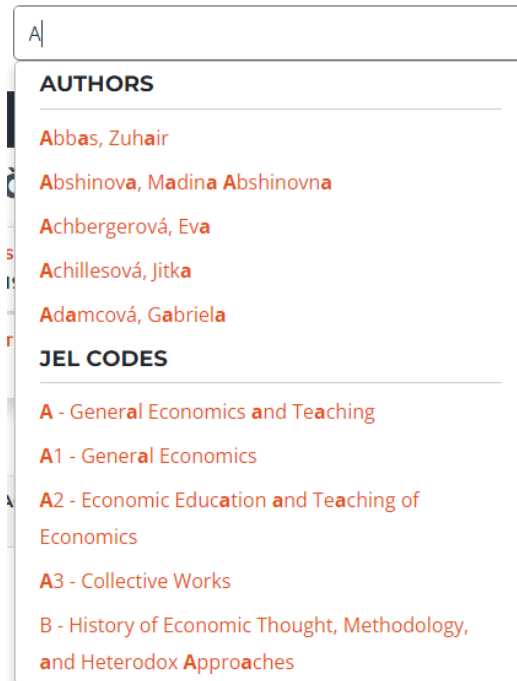
Obrázek 51. Inicializace Typeahead pro JEL kódy

Jelikož při pohybu šipkami a výběru pomocí klávesy Enter nedocházelo k správnému přesměrování na stránku, byla za inicializaci Typeahead přidána metoda `on`, která tuto chybu napraví. Rozdíl mezi autorem a JEL kódem je identifikován podle typu, který je přidáván již ve funkci `getCodesJelFromXML`.

```
}).on('typeahead:select', function (e, item) {
  if(item.type=="jel"){
    window.location.href = "jel-category.html?category=" + item.code;
  }
  else{
    window.location.href = "author.html?id=" + item.name;
  }
});
```

Obrázek 52. Oprava odkazu u výběru z klávesnice

Výsledné zobrazení můžeme vidět na následujícím obrázku, kde je nápověda rozdělena na dvě části. V první části je zobrazeno pět nalezených výsledků pro autory a v druhé části je pět výsledků pro JEL kódy.



Obrázek 53. Ukázka vyhledávání JEL kódů

6.4 Implementace nástroje Beamplot

Beamplot je vizualizační nástroj, který využívá publikace a citační data pro konkrétního autora a fakultu. Nástroj zobrazuje jednotlivé publikace jako body, přičemž na ose X je zobrazen citační percentil a na ose Y jsou jednotlivé roky. Beamplot také vypočítává medián pro každý rok a celkový medián pro všechny roky.

Data jsou získávána z databáze UTB pomocí již předem připraveného PHP souboru. Soubor je volán pomocí funkce `POST`, která následně data ve formátu JSON předává do inicializační funkce `drawBeamPlot`. Tato funkce má dva vstupní parametry, první parametr přijímá data ve formátu JSON a druhý přijímá ID HTML značky, ve které bude graf vykreslen. Pro tvorbu grafu byl využit plugin Vega Lite, který zvládá kombinovat více vrstev v jednom grafu. Zdrojová data můžeme vidět na Obrázku 54.

```

▼ 0:
  best_ais_percentile: "0.5988"
  handle: "10563/1007152"
  issn: "{0263-2241}"
  issued: "2017"
  source: "Measurement: Journal of the International Measurement Confederation"
  title: "Appropriateness of gait analysis for biometrics: Initial study using FDA method"
  type: "article"
  wos_category: "ENGINEERING, MULTIDISCIPLINARY"
  Symbol(vega_id): 1
▶ [[Prototype]]: Object

```

Obrázek 54. Ukázka zdrojových dat pro Beampot.

Tabulka 6. Popis zdrojových dat pro Beampot

Název	Popis
best_ais_percentile	Citační percentil publikace
handle	Identifikační kód publikace
issn	Osmiciferný identifikátor publikace
issued	Rok vydání
source	Zdroj měření citačních dat
title	Název publikace
type	Typ publikace
wos_category	Název kategorie ve Web of Science

V inicializační funkci `drawBeamPlot` byla nejprve vytvořena proměnná `options`, do které byla vepsána veškerá nastavení grafu. Do proměnné byla nejprve přidána vlastnost `data`, do které jsou ze vstupního parametru předána data pro `Beampot`. Použitím vlastnosti `transform` v inicializační funkci byla následně ke zdrojovým datům přidána položka `url`, která vytvoří odkaz na jednotlivé publikace. Kromě toho byla také přidána položka `percent`, která vypočítá procentuální hodnotu z hodnoty `best_ais_percentile`, jelikož citační percentil ve zdrojových datech nebyl uváděn v procentech. Následně pomocí vlastnosti `filter` je kontrolováno, zda položky "handle" nebo "title" nejsou prázdné.

```

"transform": [
  { "filter": "datum.handle!=null" },
  { "filter": "datum.title!=null" },
  { "calculate": "100*datum.best_ais_percentile", "as": "percent" },
  { "calculate": "'publication.html?handle=' + datum.handle", "as": "url" }
],

```

Obrázek 55. Definice nových proměnných `url` a `percent`. Kontrola správnosti dat

Následně byla přidána vlastnost `encoding`, ve které definujeme vlastnosti pro osu. Sekci `encoding` můžeme vidět na Obrázku 56.

```
"encoding": {  
  "x": {  
    "field": "percent",  
    "type": "quantitative",  
    "axis": {  
      "grid": false,  
      "title": "Citační percentil"  
    }  
  },  
  "y": {  
    "field": "issued",  
    "type": "ordinal",  
    "sort": "ascending",  
    "axis": {  
      "grid": true,  
      "title": "Rok vydání"  
    }  
  },  
},
```

Obrázek 56. Sekce `encoding`, definice os

Na ose X je předávána položka `percent`, která obsahuje citační percentil publikace, a na ose Y je hodnota `issued`, která udává rok vydání. Pro dosažení správných výsledků byly použity vhodné typy os a hodnoty na ose Y byly seřazeny vzestupně, což umožňuje lepší orientaci. Vlastnost `axis` byla nastavena tak, aby zobrazovala linky mřížky pouze na ose Y, na kterých budou ležet jednotlivé body. Dále byly přidány popisky pro obě osy pomocí vlastnosti `title`.

V další části kódu byly přidány jednotlivé vrstvy. První vrstva je samotná linka, která propojuje jednotlivé body publikací. Jelikož se jedná o horizontální linku, byla přiřazena k ose Y, která zobrazuje jednotlivé roky. Byla také nastavena oranžová barva, aby se hodila k již existujícímu řešení.

```

{
  "mark": "line",
  "encoding": {
    "detail": {
      "field": "issued",
      "type": "ordinal"
    },
    "color": {"value": "#E65014"}
  }
},

```

Obrázek 57. Linková vrstva na spojení bodů

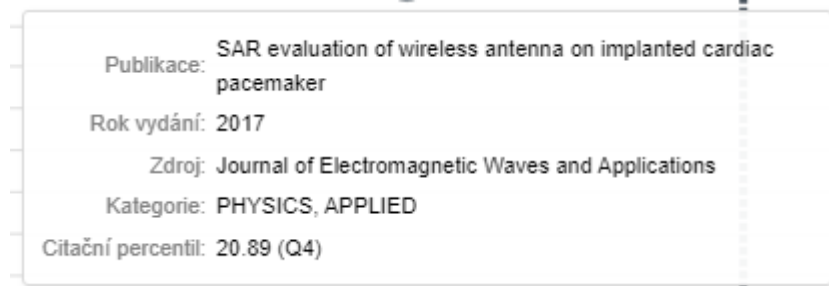
Následně byla přidána další vrstva pro jednotlivé body publikací. Pro tuto vrstvu byl zvolen typ `point`, který zobrazuje jednotlivé body na linkách. Body jsou umístěny podle citačního percentilu publikace, a proto je vrstvě přiřazena položka `percent`. Pro každý bod byla také přidána nápověda, která poskytuje všechny dostupné informace o publikaci. Byla nastavena velikost bodů, které jsou vyplněné, a barva byla nastavena na oranžovou. Kromě toho byl pomocí vlastnosti `href` nastaven odkaz na stránku publikace.

```

{
  "mark": {
    "type": "point",
    "tooltip": true,
    "filled": true,
    "size": 80
  },
  "encoding": {
    "detail": {
      "field": "percent",
      "type": "quantitative"
    },
    "href": dataJson.length < limit ? {"field": "url", "type": "nominal"} : '',
    "color": {"value": "#E65014"},
    "tooltip": [
      dataJson.length < limit ? {"field": "title", "type": "nominal", "title": "Publikace"} : '',
      {"field": "issued", "type": "ordinal", "title": "Rok vydání"},
      {"field": "source", "type": "nominal", "title": "Zdroj"},
      {"field": "wos_category", "type": "nominal", "title": "Kategorie"},
      {"field": "percent", "type": "quantitative", "title": "Citační percentil"},
    ]
  }
},

```

Obrázek 58. Bodová vrstva reprezentující publikace



Obrázek 59. Nápověda pro bodovou vrstvu publikací

Další vrstva byla přidána pro výpočet mediánu pro každý konkrétní rok. Medián pro každý rok je zobrazen pomocí bodů. Pro lepší rozlišení od bodů publikací byl bod mediánu nastaven na šedou barvu, větší velikost a nevyplněné tělo bodu. Dále byla nastavena nápověda, která zobrazuje rok a medián pro daný rok. Zdrojová data pro tuto vrstvu jsou předávána z funkce `getAnnualMedian`, která je popsána níže.



Obrázek 60. Nápověda pro bodovou vrstvu ročního mediánu

```
{
  "mark": {
    "type": "point",
    "tooltip": true,
    "filled": false,
    "size": 250,
    "strokeWidth": 2.5
  },
  "data": {"values": getAnnualMedian(dataJson)},
  "encoding": {
    "x": {
      "field": "median",
      "type": "quantitative",
    },
    "color": {"value": "#46505A"},
    "tooltip": [
      {"field": "median", "type": "quantitative", "title": "Roční medián"},
      {"field": "issued", "type": "ordinal", "title": "Rok"}
    ]
  }
},
```

Obrázek 61. Bodová vrstva pro roční medián

Funkce `getAnnualMedian` přijímá jeden vstupní parametr, a to `data` ve formátu JSON. Funkce postupně prochází všechna `data` a vybírá jednotlivé roky. Pro každý rok vytváří pole s citačními percentily a toto pole následně předává do funkce pro výpočet mediánu.

```
function getAnnualMedian(data) {
  var listYear=[];
  var numberList=[];
  var medianList=[];
  data.forEach(element => {
    listYear.push(element.issued)
  });
  var uniqueListYear = [...new Set(listYear)];
  uniqueListYear.forEach(year => {
    data.forEach(row => {
      if (year == row.issued) {
        numberList.push(Number(row.best_ais_percentile*100));
      }
    });
    var median = getMedian(numberList);
    medianList.push({"issued": year, "median": median});
    numberList=[];
  });
  return medianList;
};
```

Obrázek 62. Funkce `getAnnualMedian` pro výpočet ročního mediánu

Ve funkci `getMedian` jsou `data` seřazena vzestupně a poté je vypočítán samotný medián. Pro medián platí, že minimálně polovina hodnot je menších nebo rovných a zároveň polovina hodnot je větších nebo rovných. Po provedení výpočtu je hodnota mediánu předána zpět do funkce `getAnnualMedian`, kde je přiřazena ke konkrétnímu roku.

```
function getMedian(data) {
  var median;
  data.sort(function(a, b) {
    return a - b;
  });
  if (data.length % 2 === 0) {
    median = (data[data.length / 2 - 1] + data[data.length / 2]) / 2;
  } else {
    median = data[(data.length - 1) / 2];
  }
  return median;
}
```

Obrázek 63. Funkce `getMedian` pro výpočet mediánu

Poslední vrstvou je celkový medián, který je vypočítán ze všech publikací. Medián bude zobrazen pomocí přerušované vertikální linky, proto byl nastaven typ na `rule`. Linka byla obarvena na šedou barvu a pomocí vlastnosti `strokeDash` upravena na přerušovanou. Byla přidána nápověda, která zobrazuje hodnotu mediánu. Celkový medián je vypočítán pomocí funkce `getOverallMedian`, kterou můžeme vidět na Obrázku 64.

```

{
  "data": {"values": [{"overallMedian": getOverallMedian(dataJson)}]},
  "mark": {
    "type": "rule",
    "tooltip": true,
    "strokeDash": [4, 2],
    "size": 4
  },
  "encoding": {
    "x": {
      "datum": getOverallMedian(dataJson),
    },
    "color": {"value": "#46505A"},
    "tooltip": [
      {"field": "overallMedian", "type": "quantitative", "title": "Celkový medián: "},
    ]
  }
}

```

Obrázek 64. Vrstva s vertikální linkou pro celkový medián

Funkce `getOverallMedian` má vstupní parametr `data` ve formátu JSON. Z těchto dat funkce vybere všechny citační percentily a vloží je do pole. Toto pole je následně předáno do funkce `getMedian`, která byla již popsána v předchozí vrstvě. Funkce `getMedian` vypočítá a vrátí hodnotu mediánu pro všechny publikace.

```

function getOverallMedian(data) {
  var overallMedianArr=[];
  data.forEach(element => {
    overallMedianArr.push(element.best_ais_percentile*100);
  });
  var oMedian = getMedian(overallMedianArr);
  return oMedian;
};

```

Obrázek 65. Funkce `getOverallMedian` pro výpočet celkového mediánu

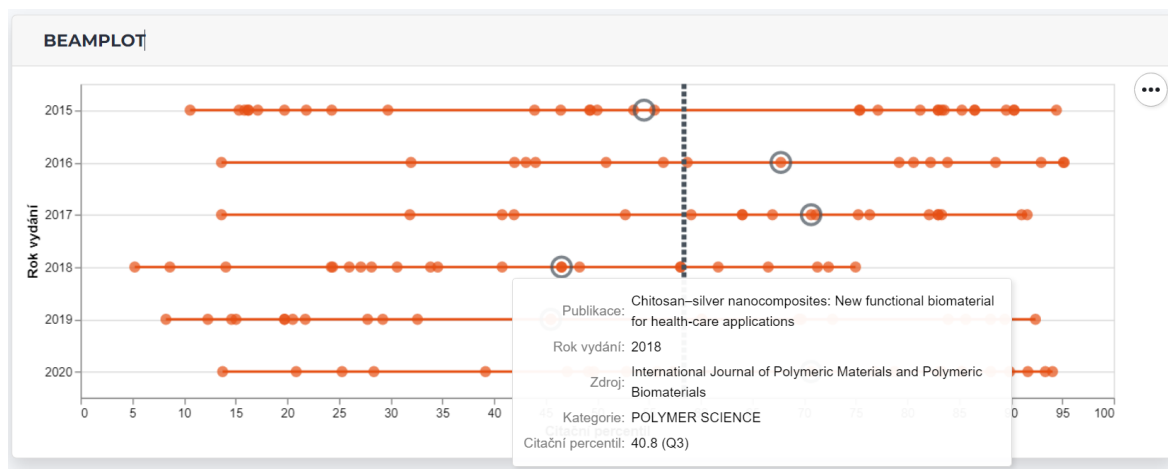
Zobrazování nástroje `Beamplot` je zajištěno pomocí dalšího pluginu s názvem `VegaEmbed`. Tento plugin obsahuje jednu hlavní funkci, a to `VegaEmbed`, která má dva vstupní parametry. Prvním parametrem je proměnná `options` s nastavením grafu a druhým je HTML značka, do které bude graf vykreslován. Tato funkce byla použita na konci inicializační funkce.

Při volání funkce `drawBeamPlot` v dokumentu HTML byla následně přidána podmínka, která kontroluje, zda nejsou obdržena data prázdná. Pokud jsou prázdná, Beamplot je skrytý.

```
$.post("db/author_beamplot.php" + window.location.search)
  .done(
    function (data) {
      var jsonObj = JSON.parse(data)
      if(jsonObj.length!==0){
        drawBeamPlot(jsonObj, '#authorBeamplotChart')
      }
      else{
        $('#authorBeamplot').hide();
      }
    }
  );
```

Obrázek 66. Volání funkce `drawBeamPlot` a kontrola obdržených dat

Pro vykreslení nástroje Beamplot byla vytvořena značka, která byla stylizována pomocí frameworku Bootstrap, aby se začlenila mezi ostatní grafy. Na Obrázku 67 je zobrazený již vykreslený Beamplot společně s nápovědou k publikaci.



Obrázek 67. Vizualizační nástroj Beamplot pro autora

Jelikož inicializační funkce byla vyvíjena tak, aby zvládla zobrazit stejný graf i pro fakultu. Stačilo na stránce fakulty přidat HTML značku pro vykreslení grafu, která byla také stylizována pomocí frameworku Bootstrap. Kvůli odlišnému `facultyId` knihovny na stránce fakult a ID v databázi byla přidána podmínka, která nastavuje ID knihovny pro Beamplot na K. Poté bylo možné pomocí funkce `POST` stáhnout data pro fakultu a následně je předat do funkce `drawBeamplot` s názvem značky.

```

if(facultyId=="kutb"){
    facultyIdForBeamplot = "K";
}
else{
    facultyIdForBeamplot=facultyId;
}

```

Obrázek 68. Podmínka pro změnu ID u knihovny

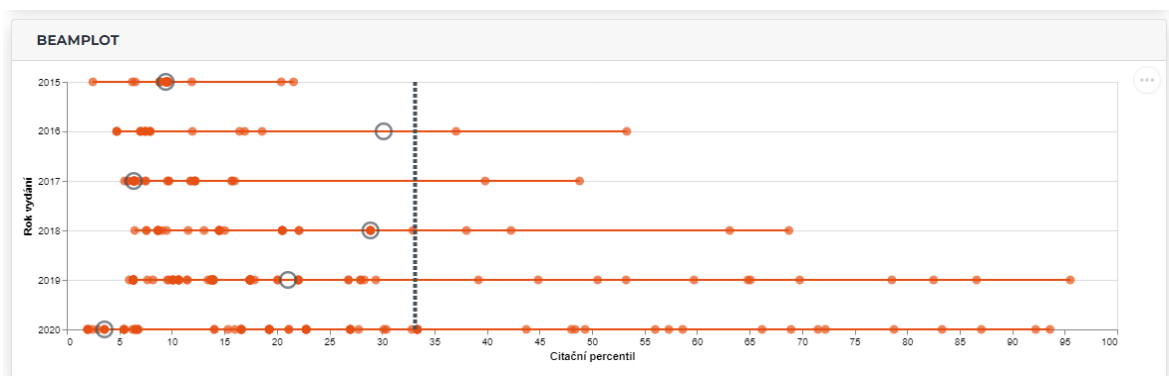
```

$.post("db/faculty_beamplot.php?faculty=" + facultyIdForBeamplot.toUpperCase())
    .done(
        function (data) {
            var jsonObj = JSON.parse(data)
            drawBeamPlot(jsonObj, '#facultyBeamplotChart')
        });

```

Obrázek 69. Volání souboru `faculty_beamplot` a následné předávání dat do funkce `drawBeamPlot`

Na Obrázku 70 můžeme vidět vykreslený vizualizační nástroj Beamplot pro Fakultu managementu a ekonomiky.



Obrázek 70. Beamplot pro Fakultu managementu a ekonomiky

6.5 Zabezpečení aplikace

Na zajištění bezpečnosti bylo nutné brát ohled převážně při stahování dat z API SHERPA/RoMEO. Pro zajištění bezpečnosti API klíče je nejlepší uchovávat ho na straně serveru namísto klienta. Při práci s API byl klíč umístěn do proměnné v souboru PHP. Tato proměnná je následně přímo vkládána do odkazu, který je volán pomocí metody `fetch`. Tímto způsobem jsme zajistili, že klíč není snadno přístupný neoprávněným osobám.

ZÁVĚR

Cílem bakalářské práce bylo přidat další funkcionality do již rozpracované nové verze webové aplikace Věda, která slouží jako užitečný nástroj pro vědecké publikace a autory. Aplikace byla vytvořena podle požadavků Knihovny UTB, pro kterou je vyvíjena.

Tato práce se v teoretické části zabývala technologiemi, které byly použity při úpravě aplikace. Byla zde popsána základní funkčnost těchto technologií a doplněna o ukázky a různé zajímavosti. Dále jsou zde popsány dvě nejpoužívanější databáze, Web of Science a Scopus, a jejich metriky, které jsou v praktické části využívány. Poslední část je věnována seznámením s již rozpracovanou webovou aplikací Věda.

Praktická část se nejprve zabývala sběrem požadavků na rozhraní. Dále byly popsány aplikace, které byly navrženy pro úpravy stávajícího technického řešení. Hlavní část se zaměřuje na implementaci a testování navržených úprav. Stručně popisuje postupy při tvorbě úprav a doplňuje je ukázkami kódu.

Do aplikace byl implementován vizualizační nástroj Beamplot, který využívá publikace a citační data. Dále byl přidán nový graf metrik na stránce časopisů, který zobrazuje metriky JCR, SNIP, SJR, IPP a CiteScore. Pro každý časopis byla přidána sekce zásad vydavatele, která je automaticky získávána z databáze SHERPA/ReMEO. Také byla implementována podpora JEL kódů pro kategorizaci ekonomických publikací. Na stránce FAME byla přidána tabulka s hierarchickým zobrazením těchto kódů a stránka umožňující vyhledávání publikací v dané kategorii. Na stránce autorů byla přidána vyhledávací fazeta autora a tlačítko pro export do formátu CSV.

Úpravy byly provedeny s cílem zvýšit kvalitu nadcházející verze aplikace Věda. Hlavním důvodem bylo přidání dalších informací o autorech, časopisech a publikacích a zlepšení jejich přehlednosti a orientace v nich.

SEZNAM POUŽITÉ LITERATURY

- [1] JANOVSKEÝ, Dušan. Abecední rejstřík HTML tagů. In: *Jak psát web* [online]. 2022 [cit. 2022-04-11]. Dostupné z: <https://www.jakpsatweb.cz/html/rejstrik.html>
- [2] HTML5 - New Tags (Elements). In: *Tutorials Point* [online]. [cit. 2022-04-11]. Dostupné z: https://www.tutorialspoint.com/html5/html5_new_tags.htm
- [3] HTML Layout Elements and Techniques. In: *W3Schools* [online]. Refsnes Data, c1999-2022 [cit. 2022-05-06]. Dostupné z: https://www.w3schools.com/html/html_layout.asp
- [4] CASTRO, Elizabeth a Bruce HYSLOP. *HTML5 a CSS3: názorný průvodce tvorbou WWW stránek*. Brno: Computer Press, 2012. ISBN 978-80-251-3733-8.
- [5] HEROUT, Tomáš. Základní selektory v CSS. In: *HelpMark* [online]. [cit. 2022-04-11]. Dostupné z: <https://www.helpmark.cz/navody/html-css/pravidla-css/562-zakladni-selektory-v-css>
- [6] JANOVSKEÝ, Dušan. CSS prakticky. In: *Jak psát web* [online]. c1999-2022 [cit. 2022-05-06]. Dostupné z: <https://www.jakpsatweb.cz/css/css-prakticky.html>
- [7] SUEHRING, Steve. *JavaScript: krok za krokem*. Brno: Computer Press, 2008. Krok za krokem (Computer Press). ISBN 978-80-251-2241-9.
- [8] MICHÁLEK, Martin. Vše o vkládání JavaScriptu do HTML. In: *Vzhůru dolů* [online]. [cit. 2022-04-11]. Dostupné z: <https://www.vzhurudolu.cz/prirucka/html-script>
- [9] JANOVSKEÝ, Dušan. Události JavaScriptu. In: *Jak psát web* [online]. 2022 [cit. 2022-04-11]. Dostupné z: <https://www.jakpsatweb.cz/javascript/udalosti.html>
- [10] Úvod, Syntaxe XML. KOSEK, Jiří. *XML pro každého: podrobný průvodce*. Praha: Grada, 2000, s. 9-26. ISBN 80-7169-860-1.
- [11] *Úvod do JSON* [online]. [cit. 2022-04-25]. Dostupné z: <https://www.json.org/json-cz.html>
- [12] JSON - Introduction. In: *W3Schools* [online]. Refsnes Data, c1999-2022 [cit. 2022-04-25]. Dostupné z: https://www.w3schools.com/js/js_json_intro.asp
- [13] MARGORÍN, Marián. *JQuery bez předchozích znalostí*. Brno: Computer Press, 2011. ISBN 978-80-251-3379-8.
- [14] HARDING, Jake, Veljko SKARICH a Tim TRUEEMAN. Typeahead.js. In: *GitHub* [online]. 2013 [cit. 2022-05-06]. Dostupné z: <https://github.com/twitter/typeahead.js>
- [15] Manual. In: *DataTables* [online]. Scotland: SpryMedia, c2007-2022 [cit. 2022-04-25]. Dostupné z: <https://datatables.net/manual/>
- [16] WELLING, Luke a Laura THOMSON. *Mistrovství PHP a MySQL*. Brno: Computer Press, 2017. ISBN 978-80-251-4892-1.
- [17] Web of Science. In: *Clarivate* [online]. [cit. 2022-04-16]. Dostupné z: <https://clarivate.com/webofsciencegroup/solutions/web-of-science/>
- [18] Scopus. In: *Elsevier* [online]. Elsevier, 2022 [cit. 2022-05-06]. Dostupné z: <https://www.elsevier.com/solutions/scopus/how-scopus-works/content>

SEZNAM POUŽITÝCH SYMBOLŮ A ZKRATEK

AIS	Article Influence Score
AJAX	Asynchronous JavaScript and XML
API	Application Programming Interface
CD	Compact Disc
CDN	Content Delivery Network
CSS	Cascading Style Sheets
CSV	Comma-separated values
DOM	Document Object Model
ES	ECMA Script
FAME	Faculty of Management and Economics
GIF	Graphics Interchange Format
GNU GPL	GNU General Public License
HTML	HyperText Markup Language
IDE	Integrated Development Environment
ID	Identification
IP	Internet Protocol
IPP	Impact per Publication
ISSN	International Standard Serial Number
JEL	Journal of Economic Literature
JIF	Journal Impact Factor
JS	JavaScript
JSON	JavaScript Object Notation
PDF	Portable Document Format
PHP	Hypertext Preprocessor
RoMEO	Rights Metadata for Open-archiving

SHERPA	Securing a Hybrid Enviroment for Research Preservation and Access
SJR	Scimago Journal Rank
SQL	Structured Query Language
UI	User Interface
UTB	Univerzita Tomáše Bati
VS	Visual Studio
W3C	World Wide Web Consortium
WoS	Web of Science
XAMPP	Cross-Platform (X), Apache (A), MariaDB (M), PHP (P), Perl (P)
XML	Extensible Markup Language

SEZNAM OBRÁZKŮ

Obrázek 1. Základní struktura HTML	12
Obrázek 2. Základní struktura v HTML5	13
Obrázek 3. Ukázka CSS deklaračního bloku.....	13
Obrázek 4. Ukázka univerzálního selektoru	14
Obrázek 5. Ukázka selektoru třídy	14
Obrázek 6. Ukázka selektoru identifikátoru	14
Obrázek 7. Ukázka selektoru elementu	14
Obrázek 8. Ukázka importu CSS stylů	15
Obrázek 9. Ukázka XML souboru.....	18
Obrázek 10. Ukázka struktury JSON.....	18
Obrázek 11. Ukázka inicializace Typeahead.....	20
Obrázek 12. Ukázka inicializace DataTables	20
Obrázek 13. WoS - Vyhledávání	24
Obrázek 14. WoS - Hledání podle názvu publikace.....	24
Obrázek 15. WoS – Nový řádek přidáný pomocí tlačítka Add Row.....	25
Obrázek 16. WoS – Nový řádek přidáný pomocí tlačítka Add date range.....	25
Obrázek 17. WoS – Hledání podle struktury	25
Obrázek 18. WoS – Vyhledávací indexy.....	26
Obrázek 19. WoS – Zobrazení výsledků	27
Obrázek 20. WoS – Zobrazení výsledků	27
Obrázek 21. Scopus – Kritéria přidávání publikací	28
Obrázek 22. Scopus – Vyhledávání	29
Obrázek 23. Scopus – Nový řádek přidáný pomocí tlačítka Add search field.....	29
Obrázek 24. Scopus – Nový řádek přidáný pomocí tlačítka Add date range	29
Obrázek 25. Scopus – Zobrazení výsledků.....	30
Obrázek 26. Scopus – Zobrazení výsledků.....	31
Obrázek 27. Hlavička stránky.....	34
Obrázek 28. Patička stránky	35
Obrázek 29. Stránka publikace před úpravou	44
Obrázek 30. Ukázka jedné položky před úpravou authors a internal_authors	45
Obrázek 31. Funkce get_formated_authors	45
Obrázek 32. Ukázka jedné položky po rozdělení authors a internal_authors.....	46
Obrázek 33. Ukázka z kódu inicializace tabulky.....	47
Obrázek 34. Nastavení orthogonálního zobrazování dat ve fazetě autora.....	48

Obrázek 35. Fazeta autora	49
Obrázek 36. Sekce buttons pro export do CSV	49
Obrázek 37. Tlačítko pro export do CSV	50
Obrázek 38. Funkce getData pro výběr dat jednotlivých metrik	51
Obrázek 39. Ukázka jedné položky v datasetu	51
Obrázek 40. Graf citačních metrik.....	52
Obrázek 41. Funkce romeo_insert_json, která slouží k rozdělení dat	54
Obrázek 42. Ukázka dat uložených v databázi	55
Obrázek 43. Zásady vydavatele – Submitted Version.....	56
Obrázek 44. Zásady vydavatele – Published Version.....	56
Obrázek 45. Ukázka XML s JEL kódy.....	57
Obrázek 46. Podmínka pro skrytí tabulky na ostatních fakultách	58
Obrázek 47. Funkce showTable pro zobrazení tabulky.....	59
Obrázek 48. Tabulka s JEL kódy na stránce FAME.....	60
Obrázek 49. Ukázka tabulky pro kategorii A1	60
Obrázek 50. Bloodhound návrh	61
Obrázek 51. Inicializace Typeahead pro JEL kódy	62
Obrázek 52. Oprava odkazu u výběru z klávesnice.....	62
Obrázek 53. Ukázka vyhledávání JEL kódů.....	63
Obrázek 54. Ukázka zdrojových dat pro Beampot.	64
Obrázek 55. Definice nových proměnných url a percent. Kontrola správnosti dat.....	64
Obrázek 56. Sekce encoding, definice os	65
Obrázek 57. Linková vrstva na spojení bodů	66
Obrázek 58. Bodová vrstva reprezentující publikace	66
Obrázek 59. Nápopvěda pro bodovou vrstvu publikací	67
Obrázek 60. Nápopvěda pro bodovou vrstvu ročního mediánu	67
Obrázek 61. Bodová vrstva pro roční medián	67
Obrázek 62. Funkce getAnnualMedian pro výpočet ročního mediánu	68
Obrázek 63. Funkce getMedian pro výpočet mediánu	68
Obrázek 64. Vrstva s vertikální linkou pro celkový medián	69
Obrázek 65. Funkce getOverallMedian pro výpočet celkového mediánu.....	69
Obrázek 66. Volání funkce drawBeamPlot a kontrola obdržených dat.....	70
Obrázek 67. Vizualizační nástroj Beamplot pro autora	70
Obrázek 68. Podmínka pro změnu ID u knihovny	71

Obrázek 69. Volání souboru fakulty_beamplot a následné předávání dat do funkce drawBeamPlot.....71

Obrázek 70. Beamplot pro Fakultu managementu a ekonomiky.....71

SEZNAM TABULEK

Tabulka 1. Párové značky HTML.....	11
Tabulka 2. Nepárové značky HTML	11
Tabulka 3. Nové značky v HTML5	12
Tabulka 4. Nejpoužívanější značky v HTML5.....	17
Tabulka 5. Parametry pro Object Retrieval API ze SHERPA/ROMEO	53
Tabulka 6. Popis zdrojových dat pro BeampLOT.....	64

SEZNAM PŘÍLOH

Příloha P 1: Předávací protokol

Příloha P 2: BSD-2-Clause Licence

Příloha P 3: Soubory na disku

PŘÍLOHA P 1: PŘEDÁVACÍ PROTOKOL

Předávací protokol

Předmět předání: Úpravy aplikace v rámci bakalářské práce “Vylepšení aplikace profily autorů”

- Přidat do tabulky publikací fazetu Autor
- Stromové zobrazení kategorií JEL
- Graf typu Beamplot u autora a fakulty
- Graf metrik u fakulty a časopisu
- Stahování informací o časopisech přes API služby SHERPA/ROMEO
- Export publikací do CSV

V rámci řešení bakalářské práce byla předávajícím předána upravená aplikace.

Výše uvedená aplikace byla předána v elektronické podobě ve formě zdrojového kódu.

Předávající a přebírající se shodli, že plnění bylo provedeno v odpovídajícím rozsahu a termínech dle požadavků přebírajícího.

Datum předání: 19.5.2023

Předávající: Dominik Palán

Přebírající: Ing. Ivan Masár, Knihovna UTB

.....

.....

PŘÍLOHA P 2: BSD-2-CLAUSE LICENCE



Note: This license has also been called the "Simplified BSD License" and the "FreeBSD License". See also the 3-clause BSD License.

Copyright <YEAR> <COPYRIGHT HOLDER>

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

PŘÍLOHA P 3: SOUBORY NA DISKU

Viz příložené CD, které obsahuje kompletní webovou aplikaci Věda.

Upravované soubory:

- author.html
- journal.html
- publications.html
- faculty.html
- scripts/publicationTableDefinition.js
- scripts/charts.js
- scripts/common.js
- scripts/search.js
- db/publications.php
- assets/css/styles.css

Nově přidané soubory:

- jel-category.html
- scripts/beamplot.js
- scripts/jelCategoryTable.js
- queries/journals_romeo_get.php
- db/Romeo_insert_json.php
- db/journal_sherpa.php

Nově importované soubory:

- assets/xml/classificationTree.xml
- assets/js/treeTable.js
- assets/css/tree-table.css