

Technická dokumentácia k projektu

Vyhľadávanie ohlasov

Autori: Norbert Jurík, Tomáš Maťko,
Adam Valentovič, Andrej Zemko

Obsah

1.	Katalóg požiadaviek	6
1.1.	Úvod	6
1.1.1.	Cieľ Katalógu požiadaviek	6
1.1.2.	Rozsah systému	6
1.1.3.	Slovník Pojmov	6
1.1.4.	Referencie	7
1.1.5.	Prehľad nasledujúcich kapitol	7
1.2.	Všeobecný Popis	7
1.2.1.	Perspektíva Systému	7
1.2.2.	Funkcie Systému	7
1.2.3.	Charakteristika používateľa	8
1.2.4.	Všeobecné Obmedzenia	8
1.2.5.	Predpoklady Závislostí	8
1.3.	Funkčné Požiadavky	8
1.3.1.	Databázy	8
1.3.1.1.	Využívanie Databáz	8
1.3.1.2.	Nedostupné Databázy	8
1.3.1.3.	Informácia o Nedostupných Databázach	8
1.3.2.	Autorizačný Kľúč	9
1.3.3.	Vstupné Prvky Úvodného Formulára	9
1.3.3.1.	v Prípade, Nevyplneného DOI	9
1.3.3.1.1.	Viaceré Roky/Rozsahy rokov Naraz	9
1.3.3.1.2.	Viaceré Afiliácie Naraz	9
1.3.3.2.	v Prípade Vyplneného DOI	9
1.3.3.2.1.	Viaceré DOI Naraz	9
1.3.4.	Zoznam Prác	9
1.3.5.	Vyhľadávanie	9
1.3.5.1.	Vyhľadávanie pri Vyplnenom DOI	9
1.3.5.2.	Vyhľadávanie pri Nevyplnenom DOI	9
1.3.5.3.	pri Vyplnenej Afiliácii	9
1.3.5.4.	pri Zadanom Roku/Rozsahu Rokov	10
1.3.6.	Zoradenie	10
1.3.7.	Informácia o Databáze v Ktojej Bola Položka Nájdená	10

1.3.8.	Zaškrtávacie Políčka	10
1.3.8.1.	Odškrtnutie	10
1.3.9.	Tlačidlo „Vyhľadaj Citácie“	10
1.3.10.	Zoznamy Citácií na Prácu/Práce	10
1.3.10.1.	Postupné Zobrazovanie Zoznamov Citácií	10
1.3.10.2.	Zobrazenie Zoznamov Citácií na Novej Karte.....	10
1.3.10.3.	Vylúčenie Autocitácií.....	10
1.3.10.3.1.	Možnosť Zobrazenia Vylúčených Autocitácií	10
1.3.10.4.	Totožné Citácie.....	10
1.3.10.5.	Zoradenie	11
1.3.10.5.1.	Zoradenie Citácií v Zoznamoch Citácií	11
1.3.10.5.2.	Zoradenie Zoznamov Citácií.....	11
1.3.10.6.	Zoznamy Citácií	11
1.3.10.6.1.	Citovaná Práca pri Zozname Citácií.....	11
1.3.10.6.2.	Citujúca Práca	11
1.3.10.7.	Počty Nájdenných Výsledkov pri Každom Zozname Citácií.....	11
1.3.10.8.	Chýbajúce Výstupné Údaje z Databáz.....	11
1.3.10.9.	Tlačidlo Kopírovania Záznamu v Internom Formáte do Podpoľa 591 v MARC 21 ..	11
1.3.10.9.1.	Totožné Citácie	11
1.3.10.9.2.	Informácia o Skopírovaní	11
1.3.11.	Formát Citácií	12
1.3.11.1.	Uprednostňovanie Informácií z WoS	12
1.3.11.2.	Konverzia Kapitalizácie Písmen	12
1.4.	Kvalitatívne Požiadavky.....	12
1.4.1.	Rozšíriteľnosť	12
1.4.2.	Dlhodobá Udržateľnosť.....	12
1.4.3.	Dostupnosť a Využiteľnosť.....	12
2.	Návrh.....	13
2.1.	Úvod	13
2.2.	Podrobná špecifikácia vonkajších interfejsov	13
2.2.1.	Používané technológie	13
2.2.1.1.	HTML a CSS	13
2.2.1.2.	Javascript.....	13
2.2.1.3.	Node JS.....	13

2.2.1.4. Scopus a Wos API	13
2.3. Komunikačné protokoly	14
2.3.1. Scopus communication protocol	14
2.3.1.1 Abstract Citation Count API	14
2.3.1.2. Citation Overview API	14
2.3.1.3. Serial Title API	14
2.3.1.4 Subject Classifications API.....	14
2.3.1.5. Abstract Retrieval API	14
2.3.1.6. Affiliation Retrieval API	15
2.3.1.7. Author Retrieval API.....	15
2.3.1.8. Affiliation Search API.....	15
2.3.1.9. Author Search API	15
2.3.1.10. Scopus Search API	15
2.3.2. WOS communication protocol.....	15
2.3.2.1. Query GET /	16
2.3.2.2. Query GET /query/{queryId}.....	17
2.3.2.3. Query GET /related	18
2.3.2.4. Query GET /citing	19
2.3.2.5. Query GET /references	20
2.3.2.6. Query GET /id/{uniqueId}	21
3.2.7. Query GET /recordids/{queryId}	22
2.3.2.8. Possible answers	22
2.4. Používateľské rozhranie	23
2.4.1. Úvodný formulár	23
2.4.2. Prehľad publikácií.....	23
2.4.3. Prehľad citácií.....	24
2.5. Popis Architektúry.....	24
2.6. Diagramy	26
2.6.1. Komponentový diagram.....	26
2.6.2. Triedny diagram	27
2.6.3. Stavový diagram.....	28
2.6.4. Sekvenčný diagram	29
2.6.4.1. Prvá fáza – Autentifikácia a vyhľadávanie publikácií	29
2.6.4.2. Druhá fáza – Vyhľadávanie citácií	30

3. Testovacie scenáre.....	31
3.1. Prihlásenie.....	31
3.2. Publikácie – doi vyhľadavanie	31
3.3. Publikácie – vyhľadavanie	31
3.4. Vypisovanie výsledkov pre dopyt publikácií	31
3.5. Označovanie publikácií.....	31
3.6. Formát výsledkov(citácií)	32
3.7. Kopírovanie citácie.....	32
3.8. Návrat na hlavnú stránku.....	32
3.9. Komplexné poradie	32

1. Katalóg požiadaviek

1.1. Úvod

1.1.1. Cieľ Katalógu požiadaviek

Tento katalóg požiadaviek popisuje požiadavky zadávateľa na funkcionality projektu „Automatické vyhľadávanie ohlasov“, ktorý zadala PhDr. Adriana Gersová ako projekt na predmet Tvorba informačných systémov odboru Aplikovaná informatika na Fakulte matematiky, fyziky a informatiky, Univerzity Komenského v Bratislave.

Katalóg požiadaviek vznikol po dohode so zadávateľom projektu a ako taký je záväzný pre obe zainteresované strany.

Špecifikované požiadavky budú počas vývoja, ako aj po ukončení vývoja softvérového produktu, slúžiť na vyhodnocovanie správnosti softvéru zadávateľom aj realizátorom.

1.1.2. Rozsah systému

Aplikácia, ktorej funkcionality bude realizovaná pomocou webovej aplikácie, API a skriptov, umožní používateľovi automaticky prehľadávať citácie k publikáciám v medzinárodných databázach.

Aplikácia cez vstupný formulár dostane potrebné údaje o citovanom dokumente, na základe ktorých pomocou API rozhrania vyhľadá relevantné záznamy v daných databázach a spracuje spätné dáta, ktoré zobrazí v naformátovanom tvare. Vyhľadané citácie bude jasne rozlišovať podľa zdroja, informácií, autocitácií, či duplikátov.

Výsledky bude možné jednoducho kopírovať pomocou tlačidla do iných aplikácií a systémov.

1.1.3. Slovník Pojmov

API: Rozhranie pre programovanie aplikácií.

Back end: Vrstva softvéru na prístup a spracovanie dát.

DOI: jednoznačný autetifikátor záznamu.

Front end: Vrstva softvéru na prezentáciu dát.

Layout: Rozloženie prvkov na obrazovke.

Užívateľské rozhranie (GUI): Používateľské rozhranie, ktoré umožňuje ovládať elektronické zariadenie pomocou súboru interaktívnych obrazových prvkov.

Webová aplikácia: Aplikácia typu klient-server vytvorená pre prostredie internetu alebo intranetu.

Skript: Program v skriptovacom jazyku pôvodne navrhnutý na uľahčenie (zautomatizovanie) operácií v počítači.

Citovaná publikácia: Publikácia, ktorá je citovaná (odkazovaná) inou publikáciou.

Citujúca publikácia: Publikácia, ktorá cituje (odkazuje sa na) inú publikáciu.

1.1.4. Referencie

Scopus API dokumentácia: https://dev.elsevier.com/sc_apis.html

WoS API dokumentácia: <https://developer.clarivate.com/apis/wos>

1.1.5. Prehľad nasledujúcich kapitol

V druhej kapitole sa nachádza všeobecný popis plánovaného systému prirodzeným jazykom, plynulými vetami bez toho, aby sme išli do veľkých podrobností. Popísané budú funkcionality systému, jeho charakteristiky a obmedzenia.

Tretia kapitola obsahuje zoznam funkčných požiadaviek.

Štvrtá kapitola obsahuje kvalitatívne požiadavky.

1.2. Všeobecný Popis

1.2.1. Perspektíva Systému

Systém bude vyvinutý pre účely vyhľadania publikácií a ich počtov citácií (ohlasov o nich). Z výsledkov vyhľadávania bude možné dohľadané informácie ľahko preniesť do iných systémov v textovej podobe (napr. do databázy evidencie publikačnej činnosti).

Systém je prednostne navrhnutý pre publikačnú činnosť Fakulty matematiky, fyziky a informatiky Univerzity Komenského. Systém bude rozšíriteľný aj pre aplikovanie na iných fakultách univerzity.

1.2.2. Funkcie Systému

Systém po zadaní mena a priezviska autora (povinné údaje), poprípade afiliácie alebo rozsahu rokov vydania publikácie, vyhľadá práce v daných databázach a zobrazí ich zoznam. Taktiež bude možné si tieto práce dohľadať podľa DOI. Tento zoradený zoznam bude obsahovať názvy publikácií, autora a rok vydania (1.3.6.).

Následne sa budú dať označiť práce pre ktoré sa majú vyhľadať ohlasy (citácie). Po označení a vyhľadaní sa objavia ďalšie informácie o práci v novom okne a informácie bude možné skopírovať do schránky v textovom formáte pre jednoduchý prenos do iných aplikácií a systémov.

1.2.3. Charakteristika používateľa

Systém bude dostupný len pre používateľov, ktorí sa autorizujú kľúčom, keďže niektoré databázy sú licencované. Osoby pracujúce so systémom budú mať znalosti so spracovaním podobných dát, nakoľko je systém určený pre účely knižníc, evidencie publikačných činností alebo iných odborne zameraných inštitúcií v ktorých je potrebné vyhľadávať citácie pre publikácie. Softvér bude používaný iba jedným typom používateľa ktorý bude mať prístup ku všetkým funkciám softvéru.

1.2.4. Všeobecné Obmedzenia

Niektoré z databáz, ktoré systém využíva, sú licencované zdroje, ktoré nie sú voľne prístupné ale sú prístupné len z univerzitných IP adries, alebo cez proxy s autorizáciou menom a heslom. Keďže systém bude fungovať na univerzitnom serveri, na ktorý majú prístup všetci a z ktorého je prístup do licencovaných databáz, bude systém využívať autorizáciu autorizačným kľúčom (bude len jeden kľúč, uložený v konfiguračných súboroch na serveri).

Obmedzenia pre tento systém bude klásť hlavne dostupnosť kľúču a rýchlosť internetového pripojenia. Rýchlosť dohľadania dát obmedzí aj odozva príslušných databáz a objem dát, ktorý bude závisieť nad rozsahom hľadaných údajov.

1.2.5. Predpoklady Závislostí

Pre správny chod bude systém vyžadovať prístup k internetu, povolený prístup k databázam citačných indexov a korektne zadané údaje vstupného formulára od používateľa. Pre optimalizáciu práce s danými údajmi musí byť systém dizajnovaný jednoduchý a prehľadný.

1.3. Funkčné Požiadavky

1.3.1. Databázy

1.3.1.1. Využívanie Databáz

Systém využíva predvolene všetky databázy, ktoré majú zadefinované implementované rozhranie API v zdrojovom kóde.

1.3.1.2. Nedostupné Databázy

V prípade nedostupnej databázy ju systém vynechá a vyhľadáva len v dostupných databázach.

1.3.1.3. Informácia o Nedostupných Databázach

Táto požiadavka je voliteľná.

V prípade nedostupných databáz na to systém zatiaľ neurčeným spôsobom upozorní používateľa (napr. formou upozornenia vo webovom prehliadači).

Spôsob závisí od náročnosti jeho implementácie.

1.3.2. Autorizačný Kľúč

V konfiguračných súboroch na serveri sa nachádza autorizačný kľúč. Pre sprístupnenie údajov (vyhľadávanie prác, zoznamov citácií) je potrebná autorizácia týmto kľúčom.

1.3.3. Vstupné Prvky Úvodného Formulára

1.3.3.1. v Prípade, Nevyplneného DOI

povinné: priezvisko autora (povinný údaj, bez diakritiky), meno autora (povinné aspoň jedno začiatkové písmeno, bez diakritiky).

nepovinné: rok/rozsah rokov, afiliácia.

1.3.3.1.1. Viaceré Roky/Rozsahy rokov Naraz

Možnosť vyplniť viaceré roky a rozsahy rokov oddelené oddeľovačom, ktorý od seba jednoznačne rozlíši roky a rozsahy rokov a bude špecifikovaný v dokumentácii.

1.3.3.1.2. Viaceré Afiliácie Naraz

Vzhľadom k nekonzistentnosti afiliácie v používaných databázach (napr.: UKOM FKAI, Katedra aplikovanej informatiky FMFI UK atď.) bude možné vyplniť viaceré formy jednej afiliácie oddelené oddeľovačom, ktorý jednotlivé afiliácie jednoznačne odlíši a bude špecifikovaný v dokumentácii.

Pre nájdenie všetkých výsledkov jednej afiliácie bude nutné zadať všetky jej formy, ktoré sú naprieč všetkými databázami ktoré systém využíva použité.

1.3.3.2. v Prípade Vyplneného DOI

V tomto prípade sa nebudú zohľadňovať žiadne iné údaje okrem údaje DOI, ktorý je v tomto prípade povinný.

1.3.3.2.1. Viaceré DOI Naraz

Možnosť zadať do formulára viaceré DOI naraz, ktoré budú oddelené oddeľovačom, ktorý ich medzi sebou jednoznačne odlíši a bude špecifikovaný v dokumentácii.

1.3.4. Zoznam Prác

Po kliknutí na tlačidlo „vyhľadaj záznamy“ sa vyhľadá zoznam prác podľa uvedených kritérií (resp. podľa údajov v úvodnom formulári (1.3.3.)).

1.3.5. Vyhľadávanie

1.3.5.1. Vyhľadávanie pri Vyplnenom DOI

Ak je vyplnené pole DOI, zoznam prác je vyhľadaný podľa DOI nezávisle od vyplnenia ostatných (vrátane povinných) údajov.

1.3.5.2. Vyhľadávanie pri Nevyplnenom DOI

Ak sú vyplnené všetky povinné údaje podľa (1.3.3.1.) a žiadne iné polia, vyhľadáva sa podľa nich.

1.3.5.3. pri Vyplnenej Afiliácii

Ak je okrem všetkých povinných údajov vyplnené aj pole afiliácia, vyhľadávajú sa len výsledky pre príslušnú afiliáciu.

1.3.5.4. pri Zadanom Roku/Rozsahu Rokov

Ak je okrem všetkých povinných údajov vyplnený rok/rozsah rokov (prípadne viaceré roky a rozsahy rokov), vyhľadávajú sa len práce z uvedeného (uvedených) roku/rozsahu rokov (rokov/rozsahov rokov).

1.3.6. Zoradenie

Zoznam prác bude zoradený najprv podľa roku, potom názvu práce abecedne.

1.3.7. Informácia o Databáze v Ktojej Bola Položka Nájdená

Vždy je nutné uviesť údaj, v ktorej databáze bola položka nájdená.

1.3.8. Zaškrťavacie Políčka

Pri každej položke zoznamu je zaškrťavacie políčko, ktorým vieme označiť práce, pre ktoré chceme vyhľadávať citácie.

1.3.8.1. Odškrtnutie

Po kliknutí na tlačidlo „vyhľadaj citácie“ sa všetky zaškrťavacie políčka odškrtnú, pretože je nepraktické odškrťávať ich pri ďalšom hľadaní manuálne, keďže vyhľadané citácie sa otvoria na novej karte.

1.3.9. Tlačidlo „Vyhľadaj Citácie“

Mimo zoznamu prác, sa bude nachádzať tlačidlo „vyhľadaj citácie“, ktoré vyhľadá zoznam (zoznamy) citácií na prácu (práce), ktorá (ktoré) mala (mali) zaškrtnuté zaškrťavacie políčko.

1.3.10. Zoznamy Citácií na Prácu/Práce

Po kliknutí na tlačidlo „vyhľadaj citácie“ sa zobrazí zoznam citácií pre každú zaškrtnutú položku zo zoznamu prác.

1.3.10.1. Postupné Zobrazovanie Zoznamov Citácií

V rámci technických možností a obmedzení API sa výsledky vyhľadávania (zoznamy citácií) budú zobrazovať postupne, ale zobrazené výsledky už budú finálne a nemenné.

1.3.10.2. Zobrazenie Zoznamov Citácií na Novej Karte

Zoznamy citácií sa zobrazia na novej karte (alebo v novom okne) vo webovom prehliadači.

1.3.10.3. Vylúčenie Autocitácií

V zoznamoch citácií na prácu/práce sú vylúčené všetky autocitácie (všetky tie, ktoré v zozname autorov obsahujú meno autora, ktorý a nachádza v zozname autorov citovaného textu).

1.3.10.3.1. Možnosť Zobrazenia Vylúčených Autocitácií

Zoznam vylúčených autocitácií bude zobraziteľný po kliknutí na tlačidlo „zobraziť vylúčené autocitácie“.

1.3.10.4. Totožné Citácie

V prípade súbežného vyhľadávania práce vo viacerých databázach (v Scopuse aj vo WoSe), pri nájdení totožných citácií vo viacerých databázach, je potrebné uviesť totožné citácie iba raz (pokiaľ je to možné).

Uprednostňujeme údaje z WoS (1.3.11.1.).

1.3.10.5. Zoradenie

1.3.10.5.1. Zoradenie Citácií v Zoznamoch Citácií

Zoznam citácií na citovanú prácu (resp. práce) bude zoradený podľa roku, priezviska a mena prvého uvedeného autora citujúcej práce (citácie).

1.3.10.5.2. Zoradenie Zoznamov Citácií

Zoznamy citácií, budú zoradené na základe práce na ktorú sa vzťahujú podľa pravidiel zoradenia zoznamu prác (1.3.6.).

1.3.10.6. Zoznamy Citácií

Výsledkom sú zoznamy citácií pre príslušné práce, pre ktorú boli citácie vyhľadávané (1.3.4.).

1.3.10.6.1. Citovaná Práca pri Zozname Citácií

Pri každom zozname citácií na prácu bude jasne uvedené, ktorá práca bola citovaná.

1.3.10.6.2. Citujúca Práca

Pri každej citácii na prácu (to sú položky zoznamov citácií) bude jasne uvedené, ktorá práca bola citujúca. Taktiež bude uvedené v ktorej databáze (resp. databázach) bola citácia nájdená.

1.3.10.7. Počty Nájdenných Výsledkov pri Každom Zozname Citácií

Pri každom zozname citácií, bude uvedené:

Počet Nájdenných Citácií v Zozname s Vylúčením Autocitácií

Počet Všetkých Citácií

Počet všetkých nájdených citácií, vrátane autocitácií, pre každý zoznam (1.3.10.6.).

1.3.10.8. Chýbajúce Výstupné Údaje z Databáz

Ak bude chýbať požadovaný údaj, zvýrazní sa vo výstupe jeho absencia. Požadované údaje sú všetky tie, ktoré sú popísané v dokumente Interný Formát do 591.

1.3.10.9. Tlačidlo Kopírovania Záznamu v Internom Formáte do Podpoľa 591 v MARC 21

Pri každej citácii bude tlačidlo „Kopírovať“, ktoré skonvertuje údaje o citácii do interného formátu, ktorého pravidlá sú v dokumente Interný Formát do 591, ktorý sa nachádza na githube:

<https://github.com/TIS2019-FMFI/hladanie-ohlasov> } <https://github.com/TIS2019-FMFI/hladanie-ohlasov> .

1.3.10.9.1. Totožné Citácie

V prípade totožných citácií treba prítomnosť citácie vo viacerých databázach vyznačiť podľa pravidiel z dokumentu Interný Formát do 591.

1.3.10.9.2. Informácia o Skopírovaní

Pri tlačidle sa po skopírovaní zobrazí informácia „Skopírované“.

1.3.11. Formát Citácií

1.3.11.1. Uprednostňovanie Informácií z WoS

Pri vyhľadávaní citácií, ktoré sú vo viacerých databázach, je potrebné v prípade totožných citácií (1.3.10.9.1.) uprednostňovať všetky názvy (aj názvy článkov, aj názvy časopisov a zdrojových dokumentov) z databázy WoS.

1.3.11.2. Konverzia Kapitalizácie Písmen

Aplikácia má za úlohu konvertovať kapitalizáciu písmen podľa nasledujúcich pravidiel:

Názvy článkov budú obsahovať veľké písmená iba na začiatku vety, prípadne podľa vlastných mien.

V prípade názvov časopisov a iných zdrojových dokumentov bude každé začiatkové písmeno slova veľké okrem predpôň a spojok. Neznáme krátke slovo sa celé vypíše UPPERCASE.

1.4. Kvalitatívne Požiadavky

1.4.1. Rozšíriteľnosť

Program bude implementovaný tak, aby ho bolo v budúcnosti možné rozšíriť aj pre prácu s inými databázami okrem WoS a Scopus.

1.4.2. Dlhodobá Udržateľnosť

Program bude dlhodobo udržateľný a správne funkčný.

1.4.3. Dostupnosť a Využiteľnosť

Program má byť využiteľný aj pre ďalšie knižnice UK.

2. Návrh

2.1. Úvod

Účelom tohto dokumentu je špecifikovať formu a technologické detaily vývoja aplikácie pre vyhľadávanie ohlasov. Dokument je určený pre vývojárov aplikácie.

2.2. Podrobná špecifikácia vonkajších interfejsov

Webová aplikácia bude bežať na serveri a dáta bude vyhľadávať na rôznych databázových serveroch (WoS, Scopus).

2.2.1. Používané technológie

2.2.1.1. HTML a CSS

Pomocou týchto skriptovacích jazykov vytvoríme používateľské rozhranie.

2.2.1.2 Javascript

Pomocou tohto programovacieho jazyka vytvoríme funkcionálnosť aplikácie ktorá sa bude vykonávať na strane užívateľa, teda v internetovom prehliadači.

2.2.1.3. Node JS

Táto technológia nám pomôže pri práci so stranou serveru, vďaka tomu nebude treba pre prácu na pozadí PHP alebo iná technológia.

2.2.1.4. Scopus a WoS API

Tieto technológie nám zabezpečia dáta z týchto databáz, ktoré budeme spracovávať podľa potreby užívateľa.

Užitočné linky:

Scopus - <https://dev.elsevier.com/>

WoS - <https://developer.clarivate.com/apis/wos>

2.3. Komunikačné protokoly

2.3.1. Scopus communication protocol

2.3.1.1 Abstract Citation Count API

Abstract Citation Count API - Returns Cited by in Scopus image given a Scopus document identifier.

This represents the interfaces to retrieve document citation counts (SCOPUS content) as a watermarked image or as metadata (JSON, XML). The metadata response also includes additional document identifiers, if available.

Formats: image/jpeg, text/html

Resource: <https://api.elsevier.com/content/abstract/citation-count>

2.3.1.2. Citation Overview API

Citation Overview API - Returns citation metadata, including counts and citation summaries, given a Scopus document identifier.

This represents the interfaces to retrieve document citation counts (SCOPUS content) broken down by year, with the option to exclude self-citations.

Formats: application/json, application/xml, text/xml

Resource: <https://api.elsevier.com/content/abstract/citations>

2.3.1.3. Serial Title API

Serial Title API - Returns metadata about serial title, including journal metrics (IPP, SJR and SNIP), given ISSN document identifier.

This provides interfaces to a search through serial titles (SCOPUS content).

Formats: application/json, application/xml, text/xml, image/gif

Resource: <https://api.elsevier.com/content/serial/title>
<https://api.elsevier.com/content/serial/title/issn/{issn}>

2.3.1.4 Subject Classifications API

Subject Classifications API - Returns subject classifications associated with Scopus content

This provides the ability to retrieve and search for the Subject Classifications associated with ScienceDirect or Scopus content.

Formats: application/json, application/xml, text/xml

Resource: <https://api.elsevier.com/content/subject/scidir>
<https://api.elsevier.com/content/subject/scopus>

2.3.1.5. Abstract Retrieval API

Abstract Retrieval API - Returns Scopus abstract given a document identifier. Abstract API response includes links various resources associated with an abstract, such as author and affiliation profiles.

This represents the interface to retrieve a Scopus abstract document. The full abstract has rich metadata representation, which includes links to author and affiliation profiles. The text of these abstracts are available for searching through use of the Scopus Search API.

Formats: application/json, application/xml, application/rdf+xml, text/xml

Resource: https://api.elsevier.com/content/abstract/scopus_id/{scopus_id}
<https://api.elsevier.com/content/abstract/eid/{eid}>
<https://api.elsevier.com/content/abstract/doi/{doi}>
<https://api.elsevier.com/content/abstract/pii/{pii}>

2.3.1.6. Affiliation Retrieval API

Affiliation Retrieval API - Returns Scopus affiliation profile given unique identifier.

This represents the interface to retrieve a SCOPUS Affiliation Profile. The response can contain links to Scopus Search and Author Profiles. The affiliation profiles are indexed and can be searched using Affiliation Search API.

Formats: application/json, application/xml, application/rdf+xml, text/xml

Resource: https://api.elsevier.com/content/affiliation/affiliation_id/{affiliation_id}
<https://api.elsevier.com/content/affiliation/eid/{eid}>

2.3.1.7. Author Retrieval API

Author Retrieval API - Returns Scopus author profile given unique identifier.

This represents retrieval of a Scopus Author Profile. The document can contain links to Scopus Search and Affiliation Profiles. The author profiles are indexed and can be searched using Author Search API.

Formats: application/json, application/xml, application/rdf+xml, text/xml

Resource: <https://api.elsevier.com/content/author>
https://api.elsevier.com/content/author/author_id/{author_id}
<https://api.elsevier.com/content/author/eid/{eid}>

2.3.1.8. Affiliation Search API

Affiliation Search API - Allows to search Scopus affiliation profiles given a search criteria.

This represents the search interfaces associated with Affiliation profiles. Each search result will, by definition, link to an Affiliation Profile.

Formats: application/json, application/xml, application/atom+xml

Resource: <https://api.elsevier.com/content/search/affiliation>

2.3.1.9. Author Search API

Author Search API - Allows to search Scopus author profiles given a search criteria.

This represents the search interfaces associated with Scopus-based Author profiles. Each search result will, by definition, link to an Author Profile. Search entries might also have links to the author's current Affiliation Profile.

Formats: application/json, application/xml, application/atom+xml

Resource: <https://api.elsevier.com/content/search/author>

2.3.1.10. Scopus Search API

Scopus Search API - Allows to search Scopus abstracts given a search criteria.

This represents the search interfaces associated with Scopus abstracts. Each search result will, by definition, link to a Scopus abstract. Search entries might also have links to a Full-Text article representation of the result.

Formats: application/json, application/xml, application/atom+xml

Resource: <https://api.elsevier.com/content/search/scopus>

2.3.2. WOS communication protocol

Všetky dopyty sú typu https, Wos podporuje odpovede JSON a XML.

Zdroj: <https://api.clarivate.com/swagger->

[ui/?url=https%3A%2F%2Fdeveloper.clarivate.com%2Fapis%2Fwos%2Fswagger](https://api.clarivate.com/swagger-?url=https%3A%2F%2Fdeveloper.clarivate.com%2Fapis%2Fwos%2Fswagger)

2.3.2.1. Query GET /

USE: The search operation submits a search query to the specified database edition and retrieves data. This operation returns a query ID that can be used in subsequent operations to retrieve more records.

REQUIERED PARAMETERS:

Name	Description
X-ApiKey *	Developer key parameter, which authenticates you to use the service.
string	
(header)	
databaseId *	Database to search. Must be a valid database ID, one of the following: BCI/BIOABS/BIOSIS/CCC/DCI/DIIDW/MEDLINE/WOK/WOS/ZOOREC. WOK represents all databases.
string	
(query)	
lang	Language of search. This element can take only one value: en for English. If no language is specified, English is passed by default.
string	
(query)	
usrQuery *	User query for requesting data, ex: TS=(cadmium). The query parser will return errors for invalid queries.
string	
(query)	
edition	Edition(s) to be searched. If null, user permissions will be substituted. Must include the name of the collection and edition name separated by '+', ex: WOS+SCI. Multiple editions are separated by ','. Editions available for collection (WOS) - AHCI,CCR,IC,ISSHP,ISTP,SCI,SSCI,BHCI,BSCI and ESCI.
string	
(query)	
publishTimeSpan	This element specifies a range of publication dates. If publishTimeSpan is used, the loadTimeSpan parameter must be omitted. If publishTimeSpan and loadTimeSpan are both omitted, then the maximum time span will be inferred from the editions data. Beginning and end dates should be specified in the yyyy-mm-dd format separated by '+', ex: 1993-01-01+2009-12-31.
string	
(query)	
loadTimeSpan	Load time span (otherwise described as symbolic time span) defines a range of load dates. The load date is the date a record was added to the database. If load date is specified, the publishTimeSpan parameter must be omitted. If both publishTimeSpan and loadTimeSpan are omitted, the maximum publication date will be inferred from the editions data. Any of D/W/M/Y prefixed with a number where D-Day, M-Month, W-Week, Y-Year allowed. Acceptable value range for Day(0-6), Week(1-52), Month(1-12) and Year(0-10), ex: 5D,30W,10M,8Y.
string	
(query)	
count *	Number of records to return, must be 0-100.
integer	
(query)	
firstRecord *	Specific record, if any within the result set to return. Cannot be less than 1 and greater than 100000.
integer	
(query)	
sortField	Order by field(s). Field name and order by clause separated by '+', use A for ASC and D for DESC, ex: PY+D. Multiple values are separated by comma.
string	
(query)	
viewField	Parameter to limit records to selected fields. Because many fields are

string	uniquely identified by a combination of element and attribute, you may not be able to limit record content to a precise degree. For example, the viewField parameter 'titles' will always return document titles, publication names (both full and abbreviated), and book titles. You cannot limit retrieval to only document (item) titles or only publication (source) titles because item and source are attribute values, not elements. You cannot include attributes in viewField parameters. Collection name and field name(s) separated by '+' ex: WOS+titles+addresses, where titles and addresses represent fields.
(query)	
optionView	Result representation. One of either FR or FS where FR=FullRecord and FS=Fields.
string	
(query)	
optionOther	Other option(s) of result representation. HL and On/Off separated by '+' where HL=HighLight: ex. HL+On.
string	
(query)	

2.3.2.2. Query GET /query/{queryId}

USE : Fetch record(s) by query identifier

REQUIERED PARAMETERS :

Name	Description
X-ApiKey *	Developer key parameter, which authenticates you to use the service.
string	
(header)	
queryId *	Retrieve records based on query identifier.
integer	
(path)	
count *	Number of records to return, must be 0-100.
integer	
(query)	
firstRecord *	Specific record, if any within the result set to return. Cannot be less than 1 and greater than 100000.
integer	
(query)	
sortField	Order by field(s). Field name and order by clause separated by '+', use A for ASC and D for DESC, ex: PY+D. Multiple values are separated by comma. If sortField was set on the original query, this parameter should be set as sorting is not a property of the query.
string	
(query)	
viewField	Parameter to limit records to selected fields. Because many fields are uniquely identified by a combination of element and attribute, you may not be able to limit record content to a precise degree. For example, the viewField parameter 'titles' will always return document titles, publication names (both full and abbreviated), and book titles. You cannot limit retrieval to only document (item) titles or only publication (source) titles because item and source are attribute values, not elements. You cannot include attributes in viewField parameters. Collection name and field name(s) separated by '+' ex: WOS+titles+addresses, where titles and addresses represent fields.
string	
(query)	
optionView	Result representation. One of either FR or FS where FR=FullRecord and FS=Fields.
string	

(query)	
optionOther	Other option(s) of result representation. HL and On/Off separated by '+' where HL=HighLight: ex. HL+On.
string	
(query)	

2.3.2.3. Query GET /related

USE : This operation finds related records for a given unique identifier. Related records will share cited references with the specified record. The operation returns the parent record along with the related records. The total number of related records for a given parent record is shown at the end of the response. The count parameter should be used to limit the number of related records returned by the operation.

REQUIERED PARAMETERS :

Name	Description
X-ApiKey *	Developer key parameter, which authenticates you to use the service.
string	
(header)	
databaseId *	Database to search. Must be a valid citation database ID, one of the following: BCI/DRCI/WOK/WOS. WOK represents all databases.
string	
(query)	
lang	Language of search. This element can take only one value: en for English. If no language is specified, English is passed by default.
string	
(query)	
uniqueId *	Primary item id to be searched, ex: WOS:000270372400005. Cannot be null or an empty string.
string	
(query)	
edition	Edition(s) to be searched. If null, user permissions will be substituted. Must include the name of the collection and edition name separated by '+', ex: WOS+SCI. Multiple editions are separated by ','. Editions available for collection(WOS) - AHCI,CCR,IC,ISSHP,ISTP,SCI,SSCI,BHCI,BSCI and ESCI.
string	
(query)	
publishTimeSpan	This element specifies a range of publication dates. If publishTimeSpan is omitted, then the maximum time span will be inferred from the editions data. Beginning and end dates should be specified in the yyyy-mm-dd format separated by '+', ex: 1993-01-01+2009-12-31.
string	
(query)	
refId	References for the parent record which are used to find related records. Multiple values are separated by comma.
string	
(query)	
count *	Number of records to return, must be 0-100.
integer	
(query)	
firstRecord *	Specific record, if any within the result set to return. Cannot be less than 1 and greater than 100000.
integer	
(query)	

sortField	Order by field(s). Field name and order by clause separated by '+', use A for ASC and D for DESC, ex: PY+D. Multiple values are separated by comma.
string	
(query)	
viewField	Parameter to limit records to selected fields. Because many fields are uniquely identified by a combination of element and attribute, you may not be able to limit record content to a precise degree. For example, the viewField parameter 'titles' will always return document titles, publication names (both full and abbreviated), and book titles. You cannot limit retrieval to only document (item) titles or only publication (source) titles because item and source are attribute values, not elements. You cannot include attributes in viewField parameters. Collection name and field name(s) separated by '+' ex: WOS+titles+addresses, where titles and addresses represent fields.
string	
(query)	
optionView	Result representation. One of either FR or FS where FR=FullRecord and FS=Fields.
string	
(query)	
optionOther	Other option(s) of result representation. HL and On/Off separated by '+' where HL=HighLight: ex. HL+On.
string	
(query)	

2.3.2.4. Query GET /citing

USE : Find citing items

REQUIERED PARAMETERS :

Name	Description
X-APIKey *	Developer key parameter, which authenticates you to use the service.
string	
(header)	
databaseId *	Database to be searched. Must be a valid citation database ID, one of the following: BCI/DRCI/WOK/WOS. WOK represents all databases.
string	
(query)	
lang	Language of search. This element can take only one value: en for English. If no language is specified, English is passed by default.
string	
(query)	
uniqueId *	Primary item id to be searched, ex: WOS:000270372400005. Cannot be null or an empty string.
string	
(query)	
edition	Edition(s) to be searched. If null, user permissions will be substituted. Must include the name of the collection and edition name separated by '+', ex: WOS+SCI. Multiple editions are separated by ','. Editions available for collection(WOS) - AHCI,CCR,IC,ISSHP,ISTP,SCI,SSCI,BHCI,BSCI and ESCI.
string	
(query)	
publishTimeSpan	This element specifies a range of publication dates. If publishTimeSpan is omitted, then the maximum time span will be inferred from the editions data. Beginning and end dates should be specified in the yyyy-mm-dd format separated by '+', ex: 1993-01-01+2009-12-31.
string	
(query)	
count *	Number of records to return, must be 0-100.

integer	
(query)	
firstRecord *	Specific record, if any within the result set to return. Cannot be less than 1 and greater than 100000.
integer	
(query)	
sortField	Order by field(s). Field name and order by clause separated by '+', use A for ASC and D for DESC, ex: PY+D. Multiple values are separated by comma.
string	
(query)	
viewField	Parameter to limit records to selected fields. Because many fields are uniquely identified by a combination of element and attribute, you may not be able to limit record content to a precise degree. For example, the viewField parameter 'titles' will always return document titles, publication names (both full and abbreviated), and book titles. You cannot limit retrieval to only document (item) titles or only publication (source) titles because item and source are attribute values, not elements. You cannot include attributes in viewField parameters. Collection name and field name(s) separated by '+' ex: WOS+titles+addresses, where titles and addresses represent fields.
string	
(query)	
optionView	Result representation. One of either FR or FS where FR=FullRecord and FS=Fields.
string	
(query)	
optionOther	Other option(s) of result representation. HL and On/Off separated by '+' where HL=HighLight: ex. HL+On.
string	
(query)	

2.3.2.5.Query GET /references

USE : Find cited references by id

REQUIERED PARAMETERS :

Name	Description
X-ApiKey *	Developer key parameter, which authenticates you to use the service.
string	
(header)	
databaseId *	Database to search. Must be a valid citation database ID, one of the following: BCI/DRCI/WOK/WOS. WOK represents all databases.
string	
(query)	
lang	Language of search. This element can take only one value: en for English. If no language is specified, English is passed by default.
string	
(query)	
uniqueId *	Primary item id to be searched, ex: WOS:000270372400005. Cannot be null or an empty string.
string	
(query)	
count *	Number of records to return, must be 0-100.
integer	
(query)	

firstRecord *	Specific record, if any within the result set to return. Cannot be less than 1 and greater than 100000.
integer	
(query)	
sortField	Order by field(s). Field name and order by clause separated by '+', use A for ASC and D for DESC, ex: PY+D. Multiple values are separated by comma.
string	
(query)	
optionOther	Other option(s) of result representation. HL and On/Off separated by '+' where HL=HighLight: ex. HL+On.
string	
(query)	

2.3.2.6. Query GET /id/{uniqueId}

USE : This operation returns a record identified by a unique identifier. You may specify multiple identifiers in a single request by separating each ID with a comma.

REQUIRED PARAMETERS :

Name	Description
X-ApiKey *	Developer key parameter, which authenticates you to use the service.
string	
(header)	
databaseId *	Database to search. Must be a valid database ID, one of the following: BCI/BIOABS/BIOSIS/CCC/DCI/DIIDW/MEDLINE/WOK/WOS/ZOOREC. WOK represents all databases.
string	
(query)	
lang	Language of search. This element can take only one value: en for English. If no language is specified, English is passed by default.
string	
(query)	
uniqueId *	Primary item(s) id to be searched, ex: WOS:000270372400005. Cannot be null or an empty string. Multiple values are separated by comma.
string	
(path)	
count *	Number of records to return, must be 0-100.
integer	
(query)	
firstRecord *	Specific record, if any within the result set to return. Cannot be less than 1 and greater than 100000.
integer	
(query)	
sortField	Order by field(s). Field name and order by clause separated by '+', use A for ASC and D for DESC, ex: PY+D. Multiple values are separated by comma.
string	
(query)	
viewField	Parameter to limit records to selected fields. Because many fields are uniquely identified by a combination of element and attribute, you may not be able to limit record content to a precise degree. For example, the viewField parameter 'titles' will always return document titles, publication names (both full and abbreviated), and book titles. You cannot limit retrieval to only document (item) titles or only publication (source) titles because item and source are attribute values, not elements. You cannot include attributes in viewField parameters.
string	
(query)	

	Collection name and field name(s) separated by '+' ex: WOS+titles+addresses, where titles and addresses represent fields.
optionView	Result representation. One of either FR or FS where FR=FullRecord and FS=Fields.
string	
(query)	
optionOther	Other option(s) of result representation. HL and On/Off separated by '+' where HL=HighLight: ex. HL+On.
string	
(query)	

3.2.7. Query GET /records/{queryId}

USE : This operation returns unique identifiers of records related to a query identifier. No other full record details are returned. This list of identifiers can be compared to an existing set of IDs (eg. supporting duplicate checks) and then be queried using /id/{uniqueId} to return the appropriate metadata.

REQUIRED PARAMETERS :

Name	Description
X-ApiKey *	Developer key parameter, which authenticates you to use the service.
string	
(header)	
queryId *	Query identifier for which to return results.
integer	
(path)	
count *	Number of record identifiers to return, number between 1-100.
integer	
(query)	
firstRecord *	Specific record, if any within the result set to return. number between 1-100,000.
integer	
(query)	
sortField	Order by field(s). Field name and order by clause separated by '+', use A for ASC and D for DESC, ex: PY+D. Multiple values are separated by comma.
string	
(query)	

2.3.2.8.Possible answers

Kód	Chybové hlásenie
200	Success, the fields available in the response will depend on the database searched.
400	Bad request
404	Not found
429	Throttle error
500	Internal server error

2.4. Používateľské rozhranie

2.4.1.Úvodný formulár

--- Povinné Údaje: ---

Hľadať podľa DOI: ☐

Meno: Priezvisko:

--- Voliteľné Údaje: ---

Roky a množiny rokov: ?

Afiliácia: ?

Hľadaj

Obrázok 1 Úvodný formulár

--- Povinné Údaje: ---

Hľadať podľa DOI: ☐

Meno: Priezvisko:

--- Voliteľné Údaje: ---

Roky a množiny rokov: ?

Afiliácia:

(zbalit' pomôcku)

Zadajte konkrétne afiliácie.

Jednotlivé afiliácie oddeľujte znakmi &&.

PRÍKLAD:

'Mars University && Hogwarts'

Vyhľadá iba záznamy s afiliáciami Mars University a Hoqwarts.

Hľadaj

Obrázok 2 Úvodný formulár s nápovedou

--- Povinné Údaje: ---

Hľadať podľa DOI: ☒

DOI:

--- Voliteľné Údaje: ---

Hľadaj

Obrázok 3 Úvodný formulár - vyhľadávanie podľa DOI

2.4.2. Prehľad publikácií

<div>Team workflow and peer review in a virtual learning environment</div> <div> <input checked="" type="checkbox"/> <div> source: Scopus year: 2017 volume: undefined pages: null doi: 10.1109/ITHET.2017.8067807 6 authors: Homola, Martin – Kubincova, Zuzana – Kl'Uka, Jan – Svolik, Martin – Darjanin, Milan – Zapalac, Peter </div> </div>
<div>Low-leakage modular regenerators for gas-turbine engines</div> <div> <input type="checkbox"/> <div> source: Scopus year: 1997 volume: 3 pages: null doi: 10.1115/97-GT-008 2 authors: Kluka, James Anthony – Wilson, David Gordon </div> </div>
Vyhľadaj citácie
Späť

Obrázok 4 Prehľad publikácií

2.4.3. Prehľad citácií

Zoznamy citácií

<div>Team workflow and peer review in a virtual learning environment</div> <div>doi: 10.1109/ITHET.2017.8067807</div> <div> <div>Collectivity from interference</div> <div> source: Scopus year: 2017 type: Article 4 authors: Blok, B. – Strikman, M. – Wiedemann, U. A. – Jakel, C. D. </div> <div>Kopírovať</div> </div>
<div>Proton-proton collisions prove stranger than expected</div> <div> source: Scopus year: 2017 type: Article 1 authors: Chang, S. </div> <div>Kopírovať</div>
<div>Review of bottomonium measurements from CMS</div> <div> source: Scopus year: 2019 type: Monograf 4 authors: Hu, Z. – Liu, T. – Leonardo, N. T. – Haytmyradov, M. </div> <div>Kopírovať</div>

Obrázok 5 Prehľad citácií

2.5. Popis Architektúry

Informačný systém bude realizovaný ako klient-server aplikácia. Serverová časť aplikácie bude umiestnená na školskom serveri a bude komunikovať voči externému prostrediu cez existujúce API, ktoré sú vypublikované nad databázami WoS a Scopus. Využívaná bude technológia node.js ako web server.

Používatelia budú pristupovať do aplikácie cez webový prehliadač. Klientska časť bude implementovaná v JavaScripte.

Interná Komunikácia medzi klientom a serverom bude prebiehať na základe požiadaviek od klienta pomocou http requestov, kde server zrealizuje dopyt a následne posiela spätné odpovede.

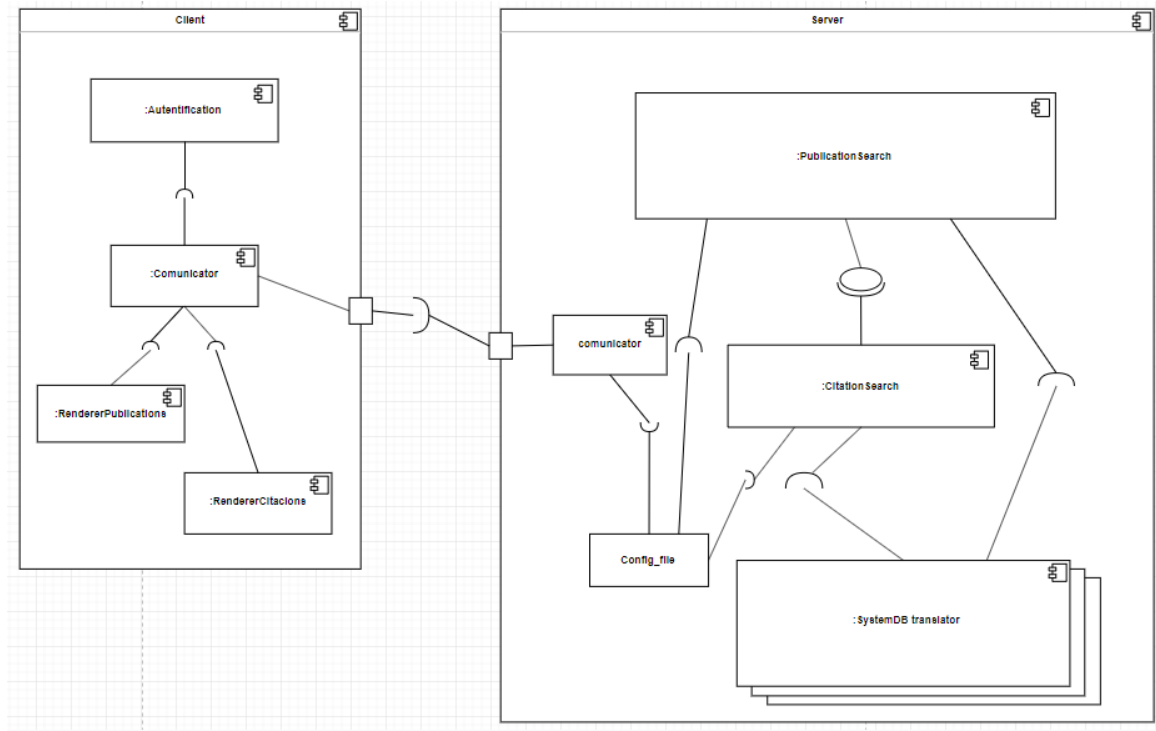
Aplikácia bude delená na moduly:

- Server (server bude komunikovať s databázami a spracovávať dáta)

- Komunikátor (modul zabezpečujúci komunikáciu medzi serverom a klientmi)
- Prihlásenie do aplikácie (prístup bude riešený 1 spoločným používateľským kontom, overeným voči uloženým údajom na serveri),
- Formulár (po korektnom vyplnení údajov formulára sa odošlú na server, ktorý ich spracuje)
- Publikácie (modul dostane dáta, ktoré budú získavané cez externé rozhranie, a následne ich spracuje a zobrazí u klienta vo webovom prehliadači),
- Citácie (modul dostane dáta, ktoré budú získavané cez externé rozhranie, a následne ich spracuje a zobrazí u klienta vo webovom prehliadači).

2.6. Diagramy

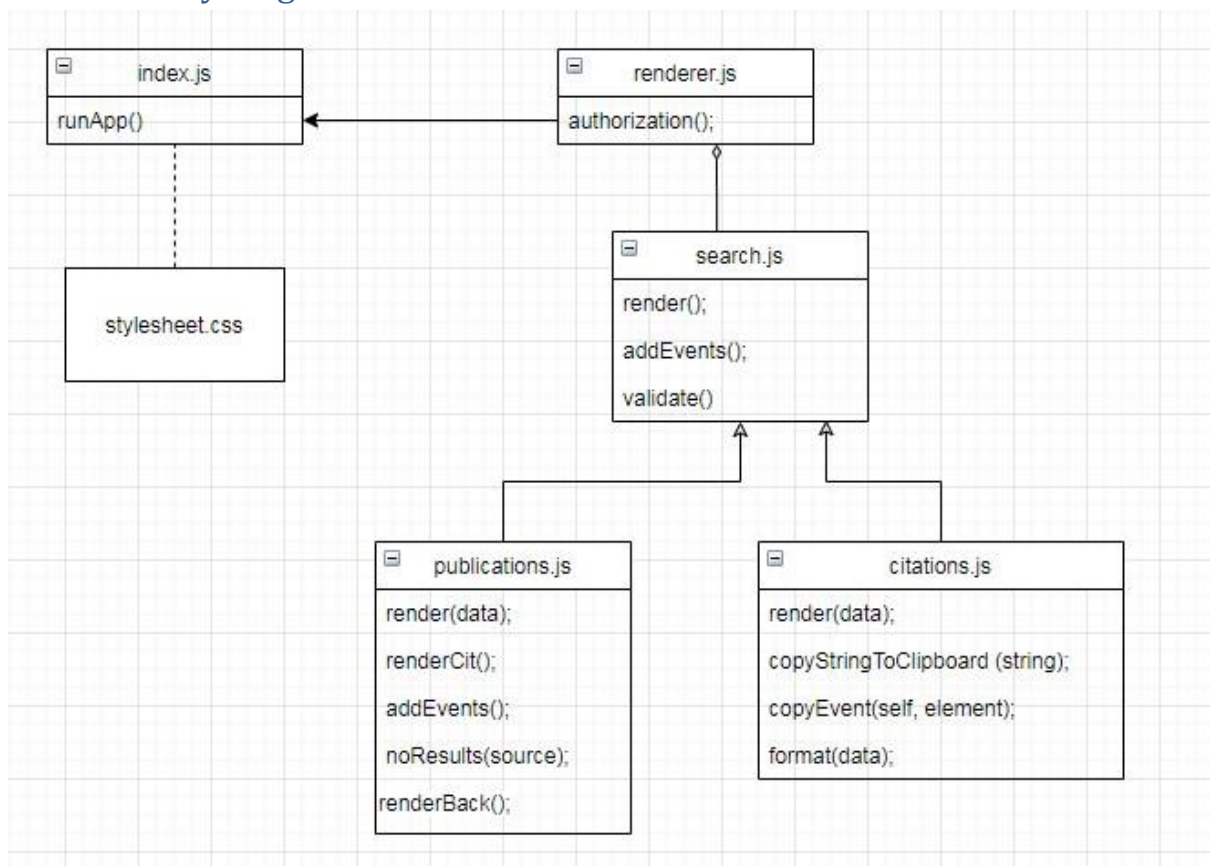
2.6.1. Komponentový diagram



Obrázok 6 Komponentový diagram

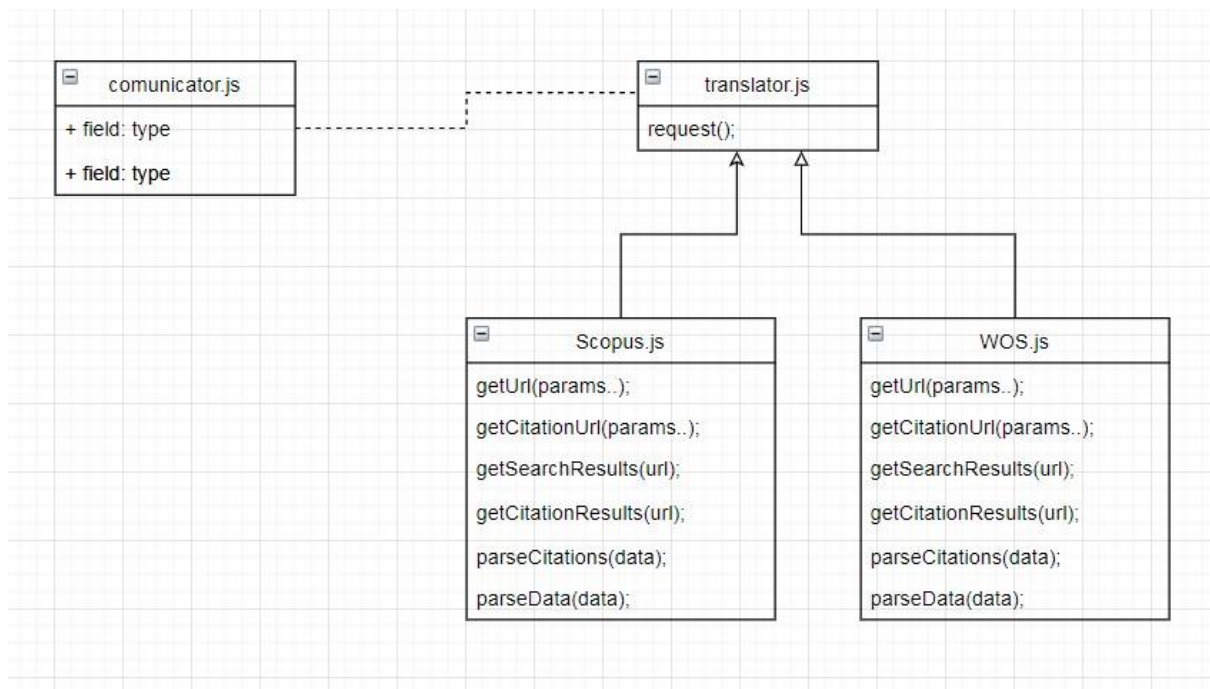
Tento diagram znázorňuje vzťah medzi jednotlivými komponentmi v aplikácii. Celá aplikácia je rozdelená na klient a server časť a jednotlivé časti majú svoje komponenty. Klient časť sa stará o vykresľovanie a zbieranie dát od používateľa. Server časť sa stará o spracovanie dát, rozposielanie dopytov do databáz tretích strán a taktiež o načítanie údajov z configu.

2.6.2. Triedny diagram



Obrázok 7 Triedny diagram - klient

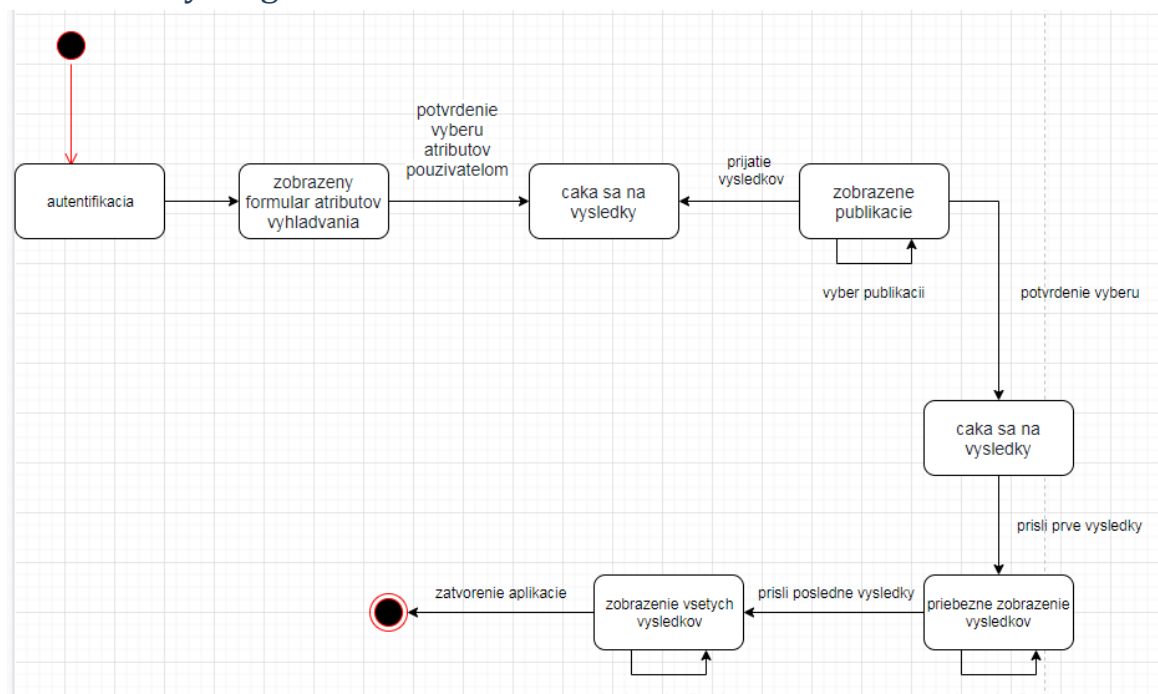
Triedny diagram zobrazuje hierarchiu tried(class) na strane klient časti aplikácie. Klient časť aplikácie sa stará o vykreslenie všetkých údajov a odoslanie zadanych údajov. Dizajn sa upravuje pomocou .css, funkcionality je .js suboroch.



Obrázok 8 Triedny diagram – server

Triedny(class) diagram pre serverovú časť zobrazuje hierarchiu medzi komunikátorom, ktorý dostáva údaje od klient časti a to ako je rozložene spracovanie v aplikácii, na serveri.

2.6.3. Stavový diagram



Obrázok 9 Stavový diagram

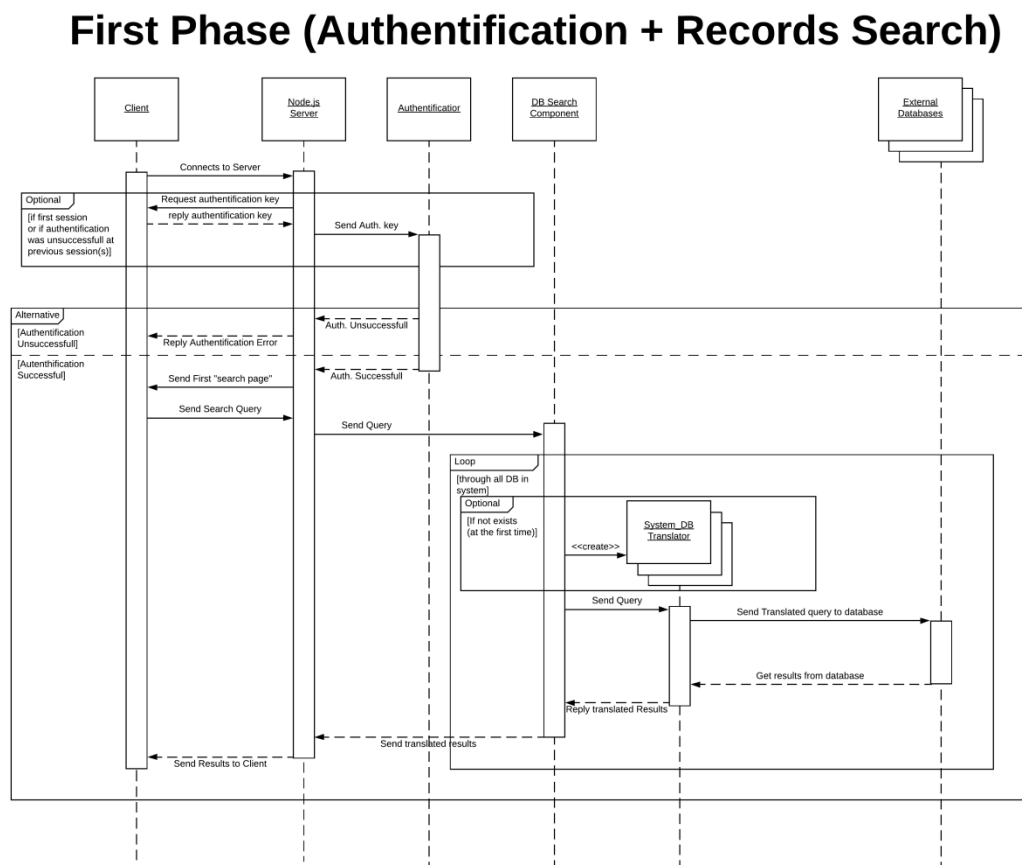
Stavový diagram zobrazuje stavy aplikácie na strane používateľa(browser).

2.6.4. Sekvenčný diagram

Funkcionalita navrhovaného riešenia je rozdelená do 2 fáz:

- Autentifikácia a vyhľadavanie publikácií
- Vyhľadavanie citácií

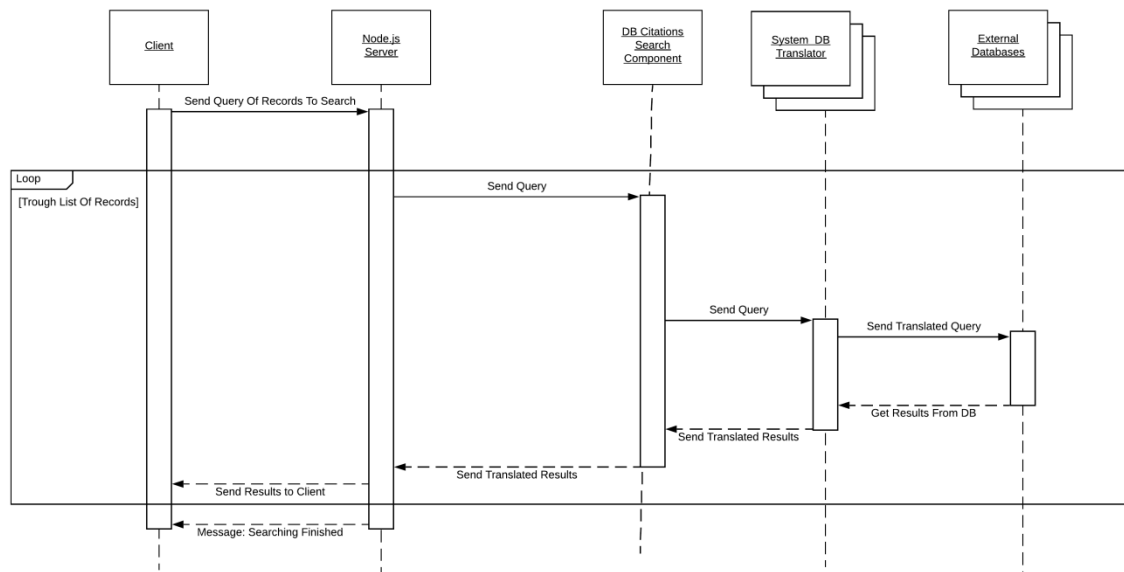
2.6.4.1. Prvá fáza – Autentifikácia a vyhľadavanie publikácií



Obrázok 10 Sekvenčný diagram - prvá fáza

2.6.4.2. Druhá fáza – Vyhľadávanie citácií

Second Phase (Citations Search)



Obrázok 11 Sekvenčný diagram - druhá fáza

Obr 10 zobrazuje sekvenčný diagram v prvej fáze vyhľadávania, taktiež je tam zahrnutá autentifikácia používateľa do aplikácie a následná možnosť pokračovať ďalej. Na obr 11 vidíme druhú fázu vyhľadávania, do ktorej sme sa dostali po prvej fáze(obr 10). Spolu oba sekvenčne diagramy znázorňujú chod celej aplikácie.

3. Testovacie scenáre.

3.1. Prihlásenie

Akcia: Používateľ pri vstupe na stránku zadá autentifikačné heslo.

Reakcia: Vložené heslo sa porovná s uloženým heslom v config súbore, ak sa zhodujú načíta sa úvodná stránka pre vyhľadávanie publikácií. Ak bolo nesprávne, vypíše sa oznam o tom a čaká sa na znovu zadanie hesla.

3.2. Publikácie – doi vyhľadávanie

Kroky ktoré sa vykonajú predtým: 1

Akcia: Používateľ označí, že chce použiť doi.

Reakcia: Formulár sa zmení podľa návrhu GUI.

Akcia: Používateľ zadá doi a stlačí vyhľadávanie.

Reakcia: Stránka odošle doi do databáz a vypíše výsledky(ak existujú), v prípade zlého doi vypíše o tom informáciu pre používateľa.

3.3. Publikácie – vyhľadávanie

Kroky ktoré sa vykonajú predtým: 1

Akcia: používateľ zadá informácie.

Reakcia: Overí sa, či sú zadané potrebné info a vypíšu sa výsledky. V prípade nedostatočných vstupných informácií sa vypíše o tom sprava pre používateľa.

3.4. Vypisovanie výsledkov pre dopyt publikácií

Výsledky sú vypísané po vykonaní vyhľadávania vo formáte ako v GUI návrhu.

3.5. Označovanie publikácií

Kroky ktoré sa vykonajú predtým: 1, 2

Akcia: Používateľ označí pre ktoré publikácie chce vyhľadávať citácie a dá vyhľadávať.

Reakcia: Načíta sa nová stránka a postupne sa budú pridávať výsledky.

3.6. Formát výsledkov(citácii)

Každý výsledok sa upraví do formátu MARC21.

3.7. Kopírovanie citácie

Kroky ktoré sa vykonajú predtým: 1, 2, 4

Akcia: Používateľ klikol na tlačidlo kopírovať pri danej citácii.

Reakcia: Do clipboardu sa skopíruje vybraná citácia vo formáte MARC21.

3.8. Návrat na hlavnú stránku

Kroky ktoré sa vykonajú predtým: 1, 3, 4, 7 alebo ľubovoľné iné

Akcia: Používateľ sa pokúsi navrátiť na pôvodnú stránku.

Reakcia: Všetky vyhľadávania sa zrušia a načíta sa úvodná stránka, už sa nevyžaduje prihlásenie.

3.9. Komplexné poradie

Kroky ktoré sa vykonajú: 1, 2, 4, 7, 8, 3, 5, 7

Aplikácia musí vždy zobrazíť správne údaje.