

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
СУМСЬКИЙ ДЕРЖАВНИЙ УНІВЕРСИТЕТ
КАФЕДРА КОМП'ЮТЕРНИХ НАУК

КВАЛІФІКАЦІЙНА МАГІСТЕРСЬКА РОБОТА

***на тему: «Інформаційна технологія
проектування рекомендаційної системи для
наукометричної бази даних Scopus»***

Завідувач випускної кафедри

Керівник роботи

Студент гр. ІНм-02

Довбиш А. С.

Берест О. Б.

Нікітін О. М.

СУМИ 2021

Факультет ЕЛІТ Кафедра Комп'ютерних наук
Спеціальність «122 - Комп'ютерні науки»

Затверджую:

зав.кафедрою _____

“ _____ ” _____ 20__ р.

ЗАВДАННЯ НА ДИПЛОМНИЙ ПРОЕКТ (РОБОТУ) СТУДЕНТОВІ

Нікітіну Олександр Миколайовичу

(прізвище, ім'я, по батькові)

1. Тема проекту (роботи) Інформаційна технологія проектування рекомендаційної системи для наукометричної бази даних Scopus

затверджую наказом по інституту від “ _____ ” _____ 20__ р. № _____

2. Термін задачі студентом закінченого проекту (роботи) _____

3. Вхідні данні до проекту (роботи) _____

4. Зміст розрахунково-пояснювальної записки (перелік питань, що їх належить розробити)

1. Інформаційний огляд;

2. Вибір методів рішення задачі;

3. Програмна реалізація та її опис;

4. Висновки.

5. Перелік графічного матеріалу (з точним зазначенням обов'язкових креслень) _____

6. Консультанти до проекту (роботи), із значенням розділів проекту, що стосується їх

Розділ	Консультант	Підпис, дата	
		Завдання видав	Завдання прийняв

7. Дата видачі завдання _____

Керівник

_____ (підпис)

Завдання прийняв до виконання

_____ (підпис)

КАЛЕНДАРНИЙ ПЛАН

№ п/п	Назва етапів дипломного проекту (роботи)	Термін виконання проекту (роботи)	Примітка
1.	Інформаційний огляд технологій		
2.	Постановка задачі та формування завдань дослідження.		
3.	Опис архітектури системи		
4.	Розробка системи		
5.	Оформлення пояснювальної записки до кваліфікаційної магістерської роботи		

Студент – дипломник

_____ (підпис)

Керівник проекту

_____ (підпис)

РЕФЕРАТ

Записка: 49 стор., 20 рис., 5 додатків, 15 джерел.

Об'єкт дослідження — Інформаційна технологія проектування рекомендаційної системи для наукометричної бази даних Scopus.

Мета роботи — розроблення інформаційної технології проектування рекомендаційної системи для наукометричної бази даних Scopus за допомогою API Elsevir.

Результати — спроектовано та програмно реалізовано інформаційну технологію для аналізу фрагментів тексту на ключові слова та використання цієї інформації для підбору рекомендованих публікацій з наукометричної бази Scopus на основі цих ключових слів, яка, швидко працює при великій кількості інформації, проста в дизайні і зрозуміла для користувача. Система повністю адаптивна до усіх пристроїв та працює у всіх сучасних браузерах. Доступ мають тільки авторизовані користувачі.

**ТЕХНОЛОГІЯ, NLP АЛГОРИТМ, НАУКОМЕТРИЧНА БАЗА,
API, SCOPUS, ELSEVIER, PHP, HTML, PYTHON, БАЗА ДАНИХ**

Зміст

ВСТУП	6
1. ІНФОРМАЦІЙНИЙ ОГЛЯД	7
1.1 Поняття API.....	7
1.2 Наукометрична база даних Scopus.....	9
1.3 Постановка задачі	11
2 ВИБІР МЕТОДІВ РІШЕННЯ ЗАДАЧІ	12
2.1 Методи прогнозування.....	12
2.2 Рекомендаційні системи	13
2.3 Мова програмування PHP.....	14
2.4 Scopus Search API	18
2.5 Дизайн веб-інтерфейсу.....	20
3 ПРОГРАМНА РЕАЛІЗАЦІЯ	22
3.1 Даталогічна модель	22
3.2 Розроблення веб-інтерфейсу	25
3.3 Алгоритм пошуку ключових слів	35
ВИСНОВКИ.....	38
СПИСОК ЛІТЕРАТУРИ.....	39
Додаток А.....	41
Додаток Б	43
Додаток В.....	45
Додаток Г	46
Додаток Д.....	48

ВСТУП

Рекомендаційна система — комплекс алгоритмів, програм та сервісів, завдання якого передбачити, що може зацікавити того чи іншого користувача. В основі роботи лежить інформація про профіль людини та інші дані.

Яскравий приклад рекомендаційної системи — Tik-Tok, про який останнім часом говорять дуже багато. Перші кілька днів користувачі бачать у стрічці все підряд, але потім програма показує користувачеві відео, що його цікавить, на основі попередніх оцінок.

В електронній торгівлі системи рекомендації націлені на надання людям допомоги у здійсненні покупок; на сайтах контент-проектів — окликані збільшити час перебування відвідувача на сайті, урізноманітнити перегляд та підвищити відвідуваність сайтів.

Навіть музичні програми не обходяться без рекомендаційних систем. Однією з найкращих має Spotify, яке нещодавно «прийшло» до України. Через кілька днів після встановлення програма підбирає для користувача треки, які з високою ймовірністю зацікавлять його.

Разом з цим багато науковців використовують наукометричні бази даних для пошуку статей, публікацій наукових журналів, матеріалів конференцій та книжкових видань.

Дана робота має мету розроблення інформаційної технології для аналізу фрагментів тексту на ключові слова та використання цієї інформації для підбору рекомендованих публікацій з наукометричної бази Scopus на основі цих ключових слів.

1. ІНФОРМАЦІЙНИЙ ОГЛЯД

1.1 Поняття API

"Прикладний програмний інтерфейс (API) — це сукупність засобів та правил, що дозволяють стандартизовану взаємодію між окремими модулями програмного забезпечення або між програмним та апаратним забезпеченням." [1]

Як працюють API?

API дають змогу вашому продукту взаємодіяти з іншими продуктами та послугами, не знаючи, як вони реалізовані. Це може спростити розробку додатків, заощадивши час і гроші. Коли ви розробляєте нові інструменти та продукти або керуєте існуючими, API надає вам:

- гнучкість,
- можливість спростити дизайн,
- можливість адміністрування та використання;
- можливості для інновацій.

API іноді вважаються контрактами з документацією, яка представляє угоду між сторонами: якщо сторона 1 надсилає запит, структурований особливим чином, саме так реагуватиме програмне забезпечення сторони 2. [8]

Оскільки API спрощує інтеграцію розробки нових програмних елементів в інфраструктуру, вони допомагають бізнес-бригадам та ІТ-командам працювати разом. Потреби бізнесу часто миттєво змінюються у відповідь на постійні мінливі цифрові галузі, коли нові суперники можуть перетворити цілий ринок за допомогою нової операції. Для того, щоб залишатися конкурентоспроможними, важливо підтримувати швидкий розвиток та розгортання інноваційних послуг. Одне із популярних рішень — розробка систем на основі хмарних технологій. Це перевірений спосіб збільшення швидкості розробки і він покладається на підключення мікро-сервісної архітектури через API.

API — це спрощений спосіб підключення вашої власної інфраструктури за допомогою розробки додатків у хмарі, вони також дозволяють ділитися своїми даними з клієнтами та іншими зовнішніми користувачами.

Загальнодоступні API представляють унікальну цінність для бізнесу, оскільки вони можуть спростити та розширити зв'язок із вашими партнерами, а також потенційно монетизувати ваші дані (популярним прикладом є API Карт Google).

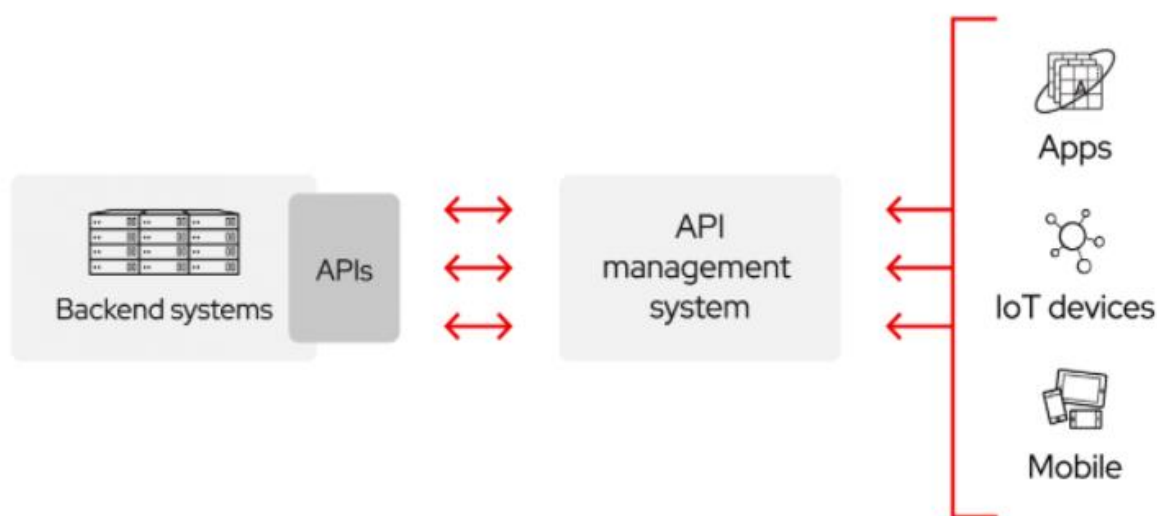


Рисунок 1.1 — Діаграма роботи API

Діаграма роботи API: серверні системи підключаються до API, які підключаються до системи керування API, які підключаються до додатків, пристроїв Інтернету речей і мобільних пристроїв. (рис. 1.1)

Наприклад, уявіть собі компанію з розповсюдження книг. Книжковий розповсюджувач може надати своїм клієнтам хмарний додаток, який дозволяє працівникам книжкових магазинів перевіряти наявність книг у розповсюджувача. Цей додаток може бути дорогим у розробці, обмеженим платформою та потребує тривалого часу розробки та постійного обслуговування.

Іншим підходом, розповсюджувач книг може надати API для перевірки наявності на складі. Цей підхід має кілька переваг:

- Надання клієнтам доступу до даних через API допомагає їм об'єднувати інформацію про свої запаси в одному місці.
- Розповсюджувач книг може вносити зміни до своїх внутрішніх систем, не впливаючи на клієнтів, доки поведінка API не змінюється.
- Завдяки загальнодоступному API розробники, які працюють на розповсюджувача книг, книгопродавців або треті сторони, можуть розробити додаток, щоб допомогти клієнтам знайти книги, які вони шукають. Це може призвести до збільшення продажів або інших можливостей для бізнесу.[8]

Отже, API дозволяють відкрити доступ до ваших ресурсів, зберігаючи безпеку та контроль. Як ви відкриваєте доступ і кому вирішувати вам. Підключення до API та створення додатків, які отримують дані або функціональні можливості, надані API, можна зробити за допомогою розподіленої інтеграційної платформи, яка з'єднує все, включаючи застарілі системи та Інтернет речей (IoT).[8]

1.2 Наукометрична база даних Scopus

Наукометрична база — бібліографічна та реферативна база даних, інструмент відстеження цитування наукових праць. Це пошукова машина, яка генерує статистичні дані, що характеризують стан і динаміку вказівників попиту, зусиль і впливу окремих вчених і дослідницьких асоціацій. [2]

Основні наукометричні показники

- Індекс цитування (Показчик цитування) — це свого роду бібліографічний показчик, індекс цитування між публікаціями, що дозволяє користувачеві легко встановити, які пізніші документи цитують які попередні документи. Кожне з них пропонує індекс цитувань між публікаціями та механізм визначення того, які документи цитуються на які інші документи.

- Індекс наукового цитування (SCI) – ключовий індекс, запроваджений Інститутом наукової інформації, який широко використовується в усьому світі для оцінки роботи дослідників і дослідницьких бригад. Показує, скільки разів твір, написаний певним автором, цитували в майстерні інших авторів за певний час. Бази даних вказують на цитування, наведене в складі списків публікацій, і дають кількісні показники цих посилань (загальна кількість цитат, показник Гірша тощо) [13].

- “Індекс Гірша (h-index) — наукометричний показник, запропонований у 2005р. американським фізиком Хорхе Хіршом з Університету Сан-Дієго, Каліфорнія, як альтернатива класичному «індексу цитування». Критерій базується на кількості публікацій дослідника та кількості цитувань цих публікацій. Його можна отримати за допомогою публічних наукометричних баз даних в Інтернеті (наприклад, Google Scholar, Science Index) і баз даних з передплатою за користування (наприклад, Scopus, Web of Science).” [13].

- Імпакт-фактор — це формальний числовий показник важливості наукового журналу, який щорічно розраховується Інститутом наукової інформації (Institute for Scientific Information) і публікується в Journal Citation Report. Він показує, скільки разів у середньому кожна опублікована в журналі стаття цитується протягом наступних двох років після публікації. Вважається, що чим вище значення імпакт-фактора, тим вище наукова цінність і авторитет журналу. [13].

Scopus — одна з наймасштабніших баз даних у світі і єдина багатодисциплінарна реферативна база даних наукових публікацій, яка є найбільшою базою наукових публікацій без повних текстів та щоденно оновлюється. Наукометрична база бази даних забезпечує облік публікацій

вчених та установ, в яких вони працюють, статистику їх цитування. Також Scopus надає гіперпосилання на повні тексти публікацій.

Інформація в цій базі використовується багатьма організаціями, що працюють у науковій сфері, аналізуючи дані, створюючи рейтинги. Глибина архіву датується 1823 роком.

База даних Scopus охоплює багато міжнародних видавництв, таких як Elsevier, SAGE, Springer, Brill, Кембриджський та Оксфордський університети та інші. Scopus відрізняється своєю політикою експертної оцінки та гарантує, що джерела ретельно розглядаються експертами.

1.3 Постановка задачі

Виконати проектування та програмну реалізацію інформаційної системи інформаційної системи для аналізу фрагментів тексту на ключові слова та пошуку у наукометричній базі Scopus. А саме:

1. Дослідити можливості Scopus API.
2. Виконати проектування даталогічної моделі системи для аналізу фрагментів тексту на ключові слова у базі Scopus.
3. Спроекувати та розробити графічний інтерфейс системи.
4. Розробити бізнес-логіку системи та програмно реалізувати.
5. Розробити систему доступу до сайту тільки авторизованих користувачів.
6. Розробити модуль пошуку ключових слів з тексту.
7. Розробити модуль рекомендацій на основі отриманих результатів.
8. Провести тестування.

Результатом роботи має стати веб-система, яка повинна бути швидкою при великій кількості інформації, проста в дизайні і зрозуміла для користувача. Система повинна бути адаптивною до всіх пристроїв, також повинна працювати у всіх сучасних веб-браузерах.

2 ВИБІР МЕТОДІВ РІШЕННЯ ЗАДАЧІ

2.1 Методи прогнозування

Нові та існуючі компанії, як правило, функціонують краще, якщо у них є візуальний довідник, який надає огляд очікуваних результатів і тенденцій. Успішні компанії часто використовують моделі прогнозування при плануванні майбутнього.

У цій статті ми обговоримо, як використовуються найпоширеніші типи моделей, і отримаємо огляд того, як створити базові моделі.

Моделі прогнозування є одним із багатьох інструментів, які підприємства використовують для прогнозування результатів щодо продажів, попиту та пропозиції, поведінки споживачів тощо. Ці моделі особливо вигідні в сфері продажів і маркетингу. Існує кілька методів прогнозування, які використовують підприємства, які надають різний рівень інформації. Від простого до складного, привабливість використання моделей прогнозування полягає в наявності візуального посилання на очікувані результати.

Хоча існує безліч способів прогнозування результатів бізнесу, існує чотири основних типи моделей або методів, які компанії використовують для прогнозування дій у майбутньому. Ви краще зрозумієте, як компанії використовують ці методи, щоб покращити свою бізнес-практику та покращити досвід клієнтів за допомогою наведених нижче прикладів поширених моделей прогнозування:

- Модель часового ряду
- Економетрична модель
- Осудна модель прогнозування
- Метод Delphi.

Компанії в галузі технологій використовують методи штучного інтелекту (ШІ) для прогнозування конкретної області зростання. Цей метод прогнозування дає надзвичайно точні результати за допомогою математичних

алгоритмів. Наука, що лежить в основі штучного інтелекту, передбачає численні результати для користувачів і допомагає генерувати ті пропозиції, які «вам також можуть сподобатися», які з'являються на певних сайтах.

Великі онлайн-компанії використовують штучний інтелект, щоб передбачити поведінку клієнтів на своїх сайтах, включаючи ймовірність покупки в майбутньому. Крім того, користувачі сайту отримують рекомендовані продукти за допомогою практики, яка називається «спільна фільтрація». Пропонування релевантних результатів покупцям відбувається шляхом групування та інтерпретації даних споживачів у поєднанні з інформацією профілю та демографічними даними. Більше даних дає більш якісні результати.

2.2 Рекомендаційні системи

Існує великий клас веб-додатків, які передбачають відповіді користувача на варіанти. Такий об'єкт називається рекомендаційною системою. Існує, два гарних приклади рекомендаційних систем:

1. Пропонування новинних статей читачам онлайн-газети на основі прогнозу читацьких інтересів.[15]
2. Пропонування клієнтам онлайн-роздрібного продавця пропозицій щодо того, що вони хотіли б придбати, на основі їх минулої історії покупок та/або пошуку продуктів.[15]

Рекомендаційні системи використовують ряд різних технологій. Ці системи можна розділити на дві великі групи.

- Системи на основі вмісту досліджують властивості елементів. Наприклад, якщо користувач Netflix переглянув багато ковбойських фільмів, порекомендуйте фільм, який класифікується в базі даних як жанр «ковбой».[15]
- Системи спільної фільтрації рекомендують елементи на основі показників подібності між користувачами та/або елементами. Рекомендовані

користувачеві елементи є тими, які віддають перевагу подібні користувачі.
[15]

Самі по собі технології недостатньо, і є деякі нові алгоритми, які виявилися ефективними для рекомендаційних систем.[15]

При побудові рекомендаційних систем в даний час використовується три основні типи алгоритмів:

- алгоритми колаборативної фільтрації;
- алгоритми, що ґрунтуються на змісті;
- гібридні алгоритми.

Основна ідея алгоритмів колаборативної фільтрації полягає у побудові рекомендацій для конкретного споживача, маючи певні дані про його профіль переваги, історію переглядів, оцінок) або думки (оцінки) групи однодумців. «Схожістю» користувачів у разі вважається кореляція векторів їх оцінок, що може розраховуватися безліччю способів. Можливість використання методів колаборативної фільтрації передбачає, що інтереси користувачів будуть представлені оцінками, які вони виставлять об'єктам після їхнього придбання, перегляду тощо.

Ніякої інформації про самих користувачів та оцінювані об'єкти не потрібно.

Алгоритми, засновані на змісті, навпаки, для створення рекомендацій передбачають наявність інформації про характеристики об'єктів. При цьому для роботи подібних алгоритмів необхідно, щоб показники були представлені у вигляді структурованих текстів.

2.3 Мова програмування PHP

PHP (Hypertext Preprocessor) — одна з найпопулярніших мов програмування, що використовується для розробки веб-додатків. В даний час PHP підтримується переважною більшістю хостинг-провайдерів, що робить

його чи не основною мовою, за допомогою якої можна розробити будь-який інтернет-проект — від простенького сайту до великого порталу.[11]

PHP — це мова сценаріїв загального призначення, орієнтована на веб-розробку. Спочатку він був створений датсько-канадським програмістом Расмусом Лердорфом у 1994 році. Довідкова реалізація PHP тепер розробляється The PHP Group. Спочатку PHP означав Personal Home Page, але тепер це означає рекурсивний ініціалізм PHP: Hypertext Preprocessor. PHP-код зазвичай обробляється на веб-сервері інтерпретатором PHP, реалізованим у вигляді модуля, демона або виконуваного файлу Common Gateway Interface (CGI). На веб-сервері результат інтерпретованого та виконаного PHP-коду — який може бути будь-яким типом даних, наприклад, згенерованим HTML або двійковим зображенням — утворює повну або частину відповіді HTTP.

Існують різні системи веб-шаблонів, системи керування веб-контентом та веб-фреймворки, які можна використовувати для організації або полегшення генерації відповідної реакції. Крім того, PHP можна використовувати для багатьох завдань програмування за межами веб-контексту, таких як автономні графічні програми та роботизоване керування дроном. PHP-код також можна виконати безпосередньо з командного рядка.

Стандартний інтерпретатор PHP, що працює на платформі Zend Engine, є безкоштовним програмним забезпеченням, випущеним під ліцензією PHP. PHP був широко портований і його можна безкоштовно розгорнути на більшості веб-серверів майже на кожній операційній системі та платформі. Мова PHP розвивалася без письмової офіційної специфікації або стандарту до 2014 року, при цьому оригінальна реалізація діяла як стандарт де-факто, якому інші реалізації мали на меті наслідувати. З 2014 року тривала робота над створенням офіційної специфікації PHP. W3Techs повідомляє, що станом на квітень 2021 року «PHP використовується 79,2% усіх веб-сайтів, чію мову програмування на стороні сервера ми знаємо».

Хоча сьогодні він не вважається найкращою з мов, ось кілька ключових переваг PHP, які допомагають пояснити, чому він все ще такий важливий у веб-розробці.

Його легко навчитися та використовувати: одна з головних причин, чому PHP став таким поширеним, полягає в тому, що його відносно просто розпочати. Навіть не маючи великих знань або досвіду в веб-розробці, більшість людей можуть створити веб-сторінку з одним файлом PHP за відносно короткий період часу. Синтаксис простий, а командні функції легко вивчити, а це означає, що перешкоди для входу в PHP нижчі, ніж у багатьох інших мовах.

Це відкритий вихідний код (і, отже, безкоштовний!): Це також допомагає розробникам розпочати роботу з PHP – його можна встановити швидко та за нульову вартість. Також є відкритий доступ до широкого спектру PHP-фреймворків, таких як Laravel та Symfony. Ця функція також приваблива для компаній, оскільки допомагає контролювати витрати на веб-розробку.

Чому PHP все ще важливий для веб-розробки:

1. Він універсальний: однією з основних переваг PHP є те, що він незалежний від платформи, тобто його можна використовувати в Mac OS, Windows, Linux і підтримує більшість веб-браузерів. Він також підтримує всі основні веб-сервери, завдяки чому його легко розгорнути на різних системах і платформах за мінімальних додаткових витрат.
2. Він користується сильною підтримкою спільноти: як ветеранська мова сценаріїв, яка широко використовується, PHP тепер має велику і лояльну базу спільнот для підтримки. Існує безліч навчальних посібників, поширених запитань та порад, які допоможуть початківцям розробникам PHP і продовжувати розширювати межі того, чого мова може досягти за допомогою регулярних оновлень.
3. Це швидко та безпечно: дві речі, які кожна організація хоче — щоб їхній веб-сайт чи додаток були швидкими та безпечними. PHP використовує

власну пам'ять і добре конкурує зі швидкістю, особливо при використанні новіших версій. У минулому були запитання щодо безпеки PHP, хоча важливо зазначити, що вона не є більш безпечною, ніж інші мови програмування. Однією з важливих переваг є те, що завдяки широкому використанню та підтримці спільноти тепер існує багато інструментів, фреймворків і найкращих практик, які допомагають виправити вразливості та захистити від кібератак.

4. Він добре пов'язаний з базами даних: PHP дозволяє легко безпечно підключатися майже до будь-якої бази даних. Це дає розробникам більше свободи при виборі бази даних, яка найкраще підходить для створюваної програми.
5. Це перевірено: однією з основних переваг існування протягом чверті століття є те, що PHP-код був випробуваний у всіх видах реального середовища. Основні помилки були знайдені та виправлені, завдяки чому мова стала більш стабільною та довіреною розробниками. Більше того, з часом було створено багато фреймворків та інструментів, які допомагають зробити веб-розробку PHP більш безпечною, ефективною та ефективною.
6. Існує багато застарілого коду: добре, це насправді не є перевагою використання PHP саме по собі, але коли так багато існуючих веб-сайтів написано на PHP, це стає важливим фактором. Простіше кажучи, зазвичай легше робити оновлення однією мовою, ніж намагатися переписати все іншою. Це допомагає PHP працювати навіть тоді, коли деякі молоді розробники можуть віддавати перевагу іншій мові. Цей застарілий ефект також означає, що зазвичай легше знайти розробника PHP для вашої організації, хоча, як ми вже писали в цьому блозі, ви хочете переконатися, що вони мають відповідні навички.

2.4 Scopus Search API

Оглянемо можливості API від Elsevier:

Для легкого розуміння роботи системи на порталі цього ресурсу є інтерактивна панель, де можна потренуватися у відправленні запитів та отримати відповіді. (рис. 2.1).

Elsevier Developers

My API key API Specification Interactive APIs How to Guides FAQ

Elsevier Scopus APIs - interactive documentation

Detailed interface documentation is located [here](#).

Affiliation_Search : Affiliation Search API

Show/Hide | List Operations | Expand Operations

Author_Search : Author Search API

Show/Hide | List Operations | Expand Operations

Scopus_Search : Scopus Search API

Show/Hide | List Operations | Expand Operations

Abstract_Retrieval : Abstract Retrieval API

Show/Hide | List Operations | Expand Operations

Affiliation_Retrieval : Affiliation Retrieval API

Show/Hide | List Operations | Expand Operations

Author_Retrieval : Author Retrieval API

Show/Hide | List Operations | Expand Operations

Abstract_Citation_Count : Abstract Citation count API

Show/Hide | List Operations | Expand Operations

Citations_Overview : Citations Overview API

Show/Hide | List Operations | Expand Operations

Serial_Title : Serial Title API

Show/Hide | List Operations | Expand Operations

[BASE URL: /content , API VERSION: 2.0]

Рисунок 2.1 — Інтерактивна панель

Обравши із безлічі варіантів для різних видів пошуку потрібний вид API, можна ввести у потрібні поля параметри запиту: Такі як стрічка запиту, ключ доступу та інші.

Response Content Type

Parameters

Parameter	Value	Description	Parameter Type	Data Type
query	<input type="text" value="all(gene)"/>	Scopus search query string	query	string
apiKey	<input type="text" value="7f59af901d2d86f78a1fd60c1bf9426a"/>	Your API key	query	string
httpAccept	<input type="text"/>	Requested content type, overrides HTTP header value	query	string
insttoken	<input type="text"/>	Specification for authorization, institution authtoken	query	string
access_token	<input type="text"/>	Specification for active session, secured authtoken	query	string

[Hide Response](#)

Curl

Рисунок 2.2 — Інтерфейс для тестових запитів

Після цього виводиться відповідь у форматі JSON або повідомлення з кодом помилки. За яким можна дізнатися у чому проблема. (рис. 2.3).

[Hide Response](#)

Curl

```
curl -X GET --header 'Accept: application/json' 'https://api.elsevier.com/content/search/scopus?query=all(gene)&apiKey=7f59af901d2d86f78a1fd60c1bf9426a'
```

Request URL

```
https://api.elsevier.com/content/search/scopus?query=all(gene)&apiKey=7f59af901d2d86f78a1fd60c1bf9426a
```

Response Body

```
{
  "prism:pageRange": "83-89",
  "prism:coverDate": "2020-11-07",
  "prism:coverDisplayDate": "7 November 2020",
  "prism:doi": "10.3329/bjms.v19i1.43876",
  "citedby-count": "0",
  "affiliation": [
    {
      "@_fa": "true",
      "affilname": "Kagawa University",
      "affiliation-city": "Takamatsu",
      "affiliation-country": "Japan"
    }
  ],
  "prism:aggregationType": "Journal",
  "subtype": "ar",
  "subtypeDescription": "Article",
  "source-id": "19900192321",
  "openaccess": "1",
  "openaccessFlag": true
},
```

Response Code

```
200
```

Response Headers

```
{
  "content-length": "5198",
  "content-type": "application/json;charset=UTF-8"
}
```

Рисунок 2.3 — Інтерфейс з відповіддю

На рисунку ми бачимо:

- URL,
- тіло запиту,
- код відповіді,
- заголовки відповіді.

2.5 Дизайн веб-інтерфейсу

Одним з перших етапів роботи над вирішенням поставленої задачі було створення прототипу графічного інтерфейсу веб-системи на якому повинні бути такі елементи:

- метрики, по яким здійснюється пошук,
- значення цих метрик,
- опис цих значень.

За допомогою графічного редактора було спроектовано дизайн зовнішнього інтерфейсу (рис. 2.4)

Веб-система

Параметри запиту

метрика1	Input text
метрика2	Input text
метрика3	Input text
метрика4	Input text
метрика5	Input text

опис значень та метрик

300x200
Advertisement

Radio button
 Selected radio button
 Radio button

ЗАПИТ

Рисунок 2.4 — Прототип графічного інтерфейсу інформаційної системи

А також проект сторінки відповіді(рис 2.5)



Рисунок 2.5 — Прототип сторінки-відповіді

Основне призначення системи — це обробка даних, отриманих за допомогою API Scopus та виведення їх у потрібному форматі. За допомогою технології API запит в заздалегідь установленому форматі відправляється на сервер бази даних Scopus, який відповідь відправляє потрібні дані. Схематично це показано на рисунку 2.6.

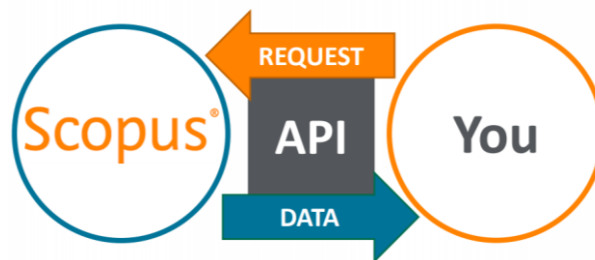


Рисунок 2.6 — Схема роботи програми

3 ПРОГРАМНА РЕАЛІЗАЦІЯ

3.1 Даталогічна модель

Для створення моделі даних було обрано реляційну СУБД MySQL.

MySQL — це реляційна система управління базами даних з відкритим вихідним кодом на основі мови SQL. Вона була розроблена та оптимізована для веб-додатків і може працювати на багатьох платформах. Вона має всі можливості, які потрібні веб-розробникам. База даних MySQL призначена для обробки мільйонів запитів та тисяч транзакцій, тому її часто вибирають компанії електронної комерції, яким потрібно керувати великою кількістю грошових переказів. Гнучкість у міру необхідності – основна характеристика MySQL.[14]

Даталогічна модель представлена на рисунку 3.1.

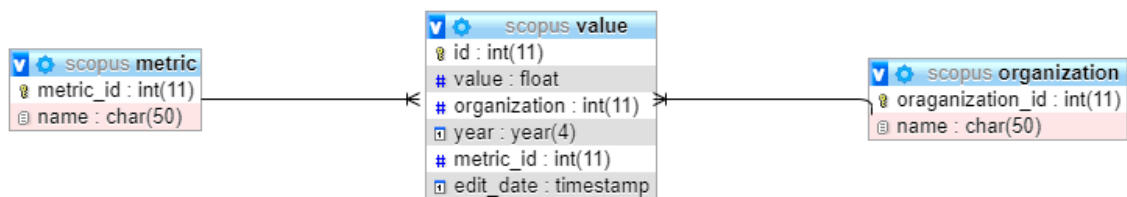


Рисунок 3.1 — Даталогічна модель

База даних сайту складається з наступних таблиць:

Таблиця 2.1 Організація

Назва поля	Опис поля	Тип даних
<u>oraganization id</u>	Номер організації	цілий
<u>name</u>	Назва організації	текстовий

Таблиця 2.2 Значення метрик

Назва поля	Опис поля	Тип даних
<u>id</u>	Номер значення	цілий
<u>organization id</u>	Номер організації	цілий
<u>metric id</u>	Номер метрики	цілий
<u>year</u>	Рік значення	рік
<u>Edit date</u>	Дата зміни	дата
<u>value</u>	Значення	float

Таблиця 2.3 Метрики

Назва поля	Опис поля	Тип даних
<u>metric id</u>	Номер метрики	цілий
<u>name</u>	Назва організації	текстовий

Сценарій створення реляційної БД MySQL:

```

DROP TABLE IF EXISTS `metric`;
CREATE TABLE IF NOT EXISTS `metric` (
  `metric id` int(11) NOT NULL AUTO_INCREMENT,
  `name` char(50) CHARACTER SET utf8mb4 COLLATE utf8mb4_0900_ai_ci
  DEFAULT NULL,
  PRIMARY KEY (`metric id`)
) ENGINE=MyISAM AUTO_INCREMENT=24 DEFAULT CHARSET=utf8mb4
COLLATE=utf8mb4_0900_ai_ci;

DROP TABLE IF EXISTS `organization`;
CREATE TABLE IF NOT EXISTS `organization` (
  `organization id` int(11) NOT NULL,
  `name` char(50) CHARACTER SET utf8mb4 COLLATE utf8mb4_0900_ai_ci
  DEFAULT NULL,
  PRIMARY KEY (`organization id`)
) ENGINE=MyISAM DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_0900_ai_ci;

DROP TABLE IF EXISTS `value`;
CREATE TABLE IF NOT EXISTS `value` (
  `id` int(11) NOT NULL AUTO_INCREMENT,
  `value` float DEFAULT NULL,
  `organization` int(11) NOT NULL,
  `year` year(4) DEFAULT NULL,
  `metric id` int(11) NOT NULL,
  `edit date` timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP,
  PRIMARY KEY (`id`)
) ENGINE=MyISAM AUTO_INCREMENT=46 DEFAULT CHARSET=utf8mb4
COLLATE=utf8mb4_0900_ai_ci;

DROP TABLE IF EXISTS `users`;
CREATE TABLE IF NOT EXISTS `users` (
  `id` int(10) NOT NULL AUTO_INCREMENT,
  `log` text NOT NULL,
  `pass` text NOT NULL,
  PRIMARY KEY (`id`)
) ENGINE=MyISAM AUTO_INCREMENT=2 DEFAULT CHARSET=utf8mb4
COLLATE=utf8mb4_0900_ai_ci;

```

Сценарій заповнення БД:


```

INSERT INTO `metric` (`metric_id`, `name`) VALUES
(22, 'OutputsInTopCitationPercentiles'),
(21, 'PublicationsInTopJournalPercentiles'),
(20, 'ScholarlyOutput'),
(19, 'HIndices'),
(18, 'FieldWeightedCitationImpact'),
(17, 'CitedPublications'),
(16, 'CollaborationImpact'),
(15, 'CitationsPerPublication'),
(14, 'CitationCount'),
(13, 'Collaboration');

INSERT INTO `organization` (`organization_id`, `name`) VALUES
(702756, 'Sumy State University');

INSERT INTO `users` (`id`, `log`, `pass`) VALUES
(1, 'admin', 'c406f4458b8c87722349d5486279dfc');
COMMIT;

```

3.2 Розроблення веб-інтерфейсу

За допомогою програмного забезпечення Apache NetBeans IDE було створено проект зі структурою, що показано на рис. 11.

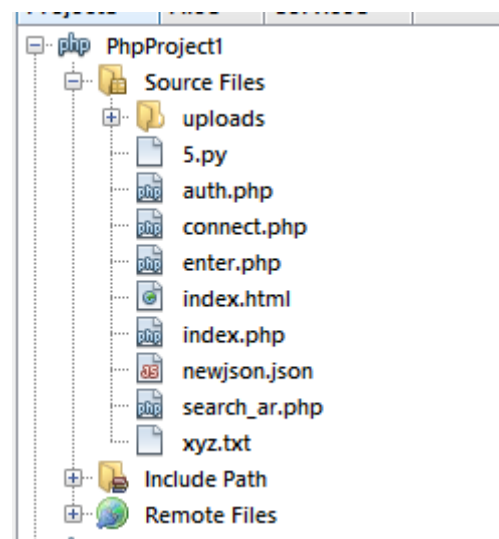


Рисунок 3.2 — Проект

Проект включає в себе веб-сторінку написану на мові HTML5 та PHP і Python скрипти для роботи з API Elsevier, внесення результатів до бази даних та перевірки авторизації і реалізації алгоритму виділення ключових слів.

PHP починався як невеликий проект з відкритим кодом, який розвивався, оскільки все більше людей дізнавалися, наскільки він корисний. Расмус Лерддорф випустив першу версію PHP ще в 1994 році.

- PHP є рекурсивним акронімом від «PHP: Hypertext Preprocessor».
- PHP — це мова сценаріїв на стороні сервера, яка вбудована в HTML. Він використовується для керування динамічним вмістом, базами даних, відстеження сеансів, навіть для створення цілих сайтів електронної комерції.
- Він інтегрований з низкою популярних баз даних, включаючи MySQL, PostgreSQL, Oracle, Sybase, Informix і Microsoft SQL Server.
- PHP приємно стрімкий у своєму виконанні, особливо коли він скомпільований як модуль Apache на стороні Unix. Після запуску сервер MySQL виконує навіть дуже складні запити з величезними наборами результатів за час встановлення рекордів.
- PHP підтримує велику кількість основних протоколів, таких як POP3, IMAP і LDAP. У PHP4 додано підтримку Java та архітектур розподілених об'єктів (COM і CORBA), що вперше зробило п-рівневу розробку можливою.
- PHP прощає: мова PHP намагається бути якомога більш прощаючим.
- Синтаксис PHP схожий на C.

За допомогою Bootstrap була створена форма для легкого вибору потрібної інформації (див. рис. 3.3). Форма дозволяє користувачу швидко обрати необхідні параметри запиту через графічний інтерфейс та звільняє від необхідності робити запити вручну. Також вона містить текстове поле вводу для аналізу текстів на ключові слова. Вміст форми буде описаний далі.

Bootstrap — це безкоштовна фреймворк розробки інтерфейсу з відкритим вихідним кодом для створення веб-сайтів і веб-програм. Фреймворк

Bootstrap побудований на HTML, CSS і JavaScript (JS) для полегшення розробки адаптивних сайтів і програм, орієнтованих на мобільні пристрої.

Адаптивний дизайн дає змогу веб-сторінці або додатку визначати розмір і орієнтацію екрана відвідувача та автоматично адаптувати відображення відповідно до цього; підхід на основі мобільних пристроїв передбачає, що смартфони, планшети та мобільні додатки для конкретних завдань є основними інструментами співробітників для виконання роботи та відповідає вимогам цих технологій у дизайні.

Web system of webometric indicators of SSU in the scientometric database Scopus

Parameter	Value	Description
metrics	<input type="text" value="CitationCount"/>	Names of the desired metrics
entity	<input type="text" value="Institution (default)"/>	Type of the requested entity
id	<input type="text" value="702756"/>	Id of the requested entity
yearRange	<input type="text" value="5yrs (default)"/>	Year range of returned metric
includeSelfCitations	<input type="text" value="true (default)"/>	Whether to include self-cited documents (not applicable to all metrics)
byYear	<input type="text" value="true (default)"/>	Whether to return results broken down by year (not applicable to all metrics)
includedDocs	<input type="text" value="AllPublicationTypes (default)"/>	Publication types to include (not applicable to all metrics)

[Search](#)

A recommendation system for the Scopus scientometric database based on keywords

Enter your text:

[Highlight keywords](#)

[Logout](#)

Рисунок 3.3 — Форма для конструювання запитів

Код сторінки `index.php` можна знайти у ДОДАТКУ А.

На ній ми бачимо декілька полів:

- Метрику на вибір. (Рисунок 13)
- Тип сутності (установа або автор). (Рисунок 14)
- Id сутності.

- Період часу, за який потрібні дані. (Рисунок 15)
- Самоцитати.
- Сортування по роках.
- Тип публікацій. (Рисунок 16)

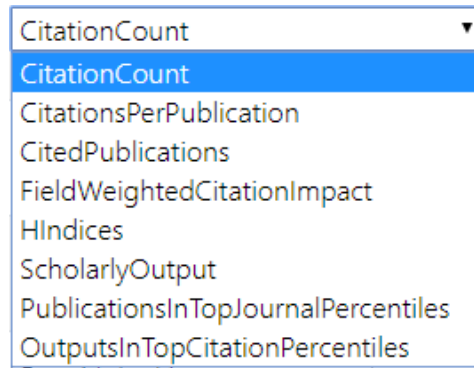


Рисунок 3.4 — Поле форми

Тут можна обрати метрику на вибір.

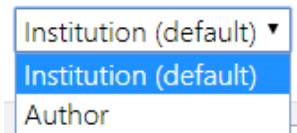


Рисунок 3.5 — Поле форми

Обираємо тип сутності.

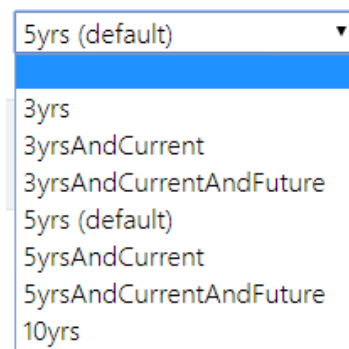


Рисунок 3.6 — Поле форми

Кількість років за які отримаємо метричні дані.

A screenshot of a web interface showing a dropdown menu. The selected option is "AllPublicationTypes (default)". The menu is open, displaying the following options: "AllPublicationTypes (default)", "ArticlesOnly", "ArticlesReviews", "ArticlesReviewsConferencePapers", "ConferencePapersOnly", "ArticlesConferencePapers", "BooksAndBookChapters", and "ArticlesReviewsConferencePapersBooksAndBookChapters".

Рисунок 3.7 — Поле форми

Обираємо типи публікацій що включаються у підрахунок.

Блок для пошуку ключових слів із тексту можна побачити на рисунку 3.8.

A screenshot of a web interface for a recommendation system. At the top, it says "A recommendation system for the Scopus scientometric database based on keywords". Below this is a search form with a text input field containing the placeholder "Enter your text:". To the right of the input field is a large empty rectangular area. Below the input field is a blue button labeled "Highlight keywords". Below the button is a "Logout" link. At the bottom of the page is the logo of Sumy State University, which consists of a stylized "U" and "J" with a globe and a torch, followed by the text "Сумський державний університет".

Рисунок 3.8 — Блок для пошуку ключових слів

Вставивши деякий текст, може отримати ключові слова тексту та здійснити пошук по ним у базі Scopus.

A recommendation system for the Scopus scientometric database based on keywords

Enter your text:

Delivering a comprehensive overview of the world's research output in the fields of science, technology, medicine social sciences and arts and humanities, Scopus features smart tools to track, analyze and visualize research. From researchers pursuing scientific breakthroughs to academic institutions and government agencies evaluating research, Scopus is the abstract and indexing database of choice. Worldwide, Scopus is used by more than 3,000 academic, government and corporate institutions and is the main data source that supports the Research Intelligence portfolio.

Highlight keywords

Keywords:

medicine social sciences

main data source

world's research output

Search articles

Рисунок 3.9 — Ключові слова

Також на PHP написаний скрипт, який обробляє введені користувачем дані та робить запит до системи Elsevier з використанням технології API.

```

$metrics = htmlspecialchars($_GET['metrics']);
$entity = htmlspecialchars($_GET['entity']);
$id = htmlspecialchars($_GET['id']);
$yearRange = htmlspecialchars($_GET['yearRange']);
$includeSelfCitations =
htmlspecialchars($_GET['includeSelfCitations']);
$byYear = htmlspecialchars($_GET['byYear']);
$includedDocs = htmlspecialchars($_GET['includedDocs']);

$ch =
curl_init('https://api.elsevier.com/analytics/scival/metrics/'. $m
etrics.'/'. $entity.'/'. $id.'?yearRange='. $yearRange.'&includeSelf
Citations='
.$includeSelfCitations.'&byYear='. $byYear.'&includedDocs='. $inclu
dedDocs.'&journalImpactType=CiteScore&showAsFieldWeighted=false&i
ndexType=hIndex&apiKey=<key>');

curl_setopt($ch, CURLOPT_RETURNTRANSFER, 1);
curl_setopt($ch, CURLOPT_HTTPHEADER, array('Accept:
application/json'));
curl_setopt($ch, CURLOPT_SSL_VERIFYPEER, FALSE);
$result = curl_exec($ch);

```

Далі отримується результат у форматі JSON, обробляється та переноситься до нашої БД

```

$mysqli = new mysqli('localhost', 'root', '', 'scopus');
if ($mysqli->connect_errno) {
    echo 'error with connection to db ';
    echo "Номер ошибки: " . $mysqli->connect_errno . "\n";
}

$years =
array_keys($data['results']['0']['metric']['valueByYear']);
$values =
array_values($data['results']['0']['metric']['valueByYear']);

$sql = "SELECT `organization_id` FROM `organization` WHERE
name = '" . $data['results']['0']['institution']['name'] . "'";
$result = $mysqli->query($sql);
$organization_id = mysqli_fetch_array($result);

$sql = "SELECT `metric_id` FROM `metric` WHERE name = '" .
$data['results']['0']['metric']['metricType'] . "'";
$result = $mysqli->query($sql);
$metric_id = mysqli_fetch_array($result);

foreach ($years as $key => $value) {
    $sql = "INSERT INTO `value` (`value`, `organization`,
`year`, `metric_id`) VALUES
('$values[$key].', '$organization_id[0].', STR_TO_DATE('$va
lue.', '%Y'), '$metric_id[0].')";
    $result = $mysqli->query($sql);
}

```

Повний код php скрипту у ДОДАТКУ Б.

Web system of webometric indicators of SSU in the scientometric database Scopus

Year	Organization	Metric	Value
2014	Sumy State University	CitationCount	1393
2015	Sumy State University	CitationCount	1318
2016	Sumy State University	CitationCount	1346
2017	Sumy State University	CitationCount	1411
2018	Sumy State University	CitationCount	946

[Back](#)



Рисунок 3.10 — Результат запиту

Результат виконання запиту можна побачити на рисунку 3.10.
На наступному рисунку ми бачимо приклад даних у базі даних.

	id	value	organization	year	metric_id	edit_date
4	1	1306	702756	2014	14	2020-03-03 14:58:55
4	2	1222	702756	2015	14	2020-03-03 14:58:55
4	3	1203	702756	2016	14	2020-03-03 14:58:55
4	4	1232	702756	2017	14	2020-03-03 14:58:55
4	5	725	702756	2018	14	2020-03-03 14:58:55

Рисунок 3.11 — Приклад даних з БД

Результат виконання пошуку публікацій на рисунку 3.12

Romantic involvement: A protective factor for psychological health in racially-diverse young sexual minorities

Title:	Romantic involvement: A protective factor for psychological health in racially-diverse young sexual minorities
Publication Name:	Journal of Abnormal Psychology
Publication Date:	April 2018
Author:	Whitton S.
Link:	https://api.elsevier.com/content/abstract/scopus_id/85041228517
Cited by count:	29

Analysis of professor ioan drăgoiu's opening lecture at the chair of histology from the cluj faculty of medicine, romania

Title:	Analysis of professor ioan drăgoiu's opening lecture at the chair of histology from the cluj faculty of medicine, romania
Publication Name:	Romanian Journal of Morphology and Embryology
Publication Date:	2015
Author:	Bârsu C.
Link:	https://api.elsevier.com/content/abstract/scopus_id/84953390555
Cited by count:	1

In search for the most informative data for feedback generation: Learning analytics in a data-rich context

Title:	In search for the most informative data for feedback generation: Learning analytics in a data-rich context
Publication Name:	Computers in Human Behavior
Publication Date:	June 2015
Author:	Tempelaar D.T.
Link:	https://api.elsevier.com/content/abstract/scopus_id/85027932823
Cited by count:	217

Рисунок 3.12 — Результат пошуку

Для цього виконується ще один запит до API Scopus та декодується ще одна відповідь:

```

$keywords = file("xyz.txt");
foreach ($keywords as $line_num => $keyword) {
    $keyword = str_replace(' ', '%20', $keyword);
    $keyword = trim($keyword);
    //echo $keyword;
    $ch =
    curl_init('https://api.elsevier.com/content/search/scopus?query=title%
7B' . $keyword.
'%7D&apiKey=7f59af901d2d86f78a1fd60c1bf9426a&count=3&sort=citedby-coun
t');
    curl_setopt($ch, CURLOPT_RETURNTRANSFER, 1);
    curl_setopt($ch, CURLOPT_HTTPHEADER, array('Accept:
application/json'));
    curl_setopt($ch, CURLOPT_SSL_VERIFYPEER, FALSE);
    $result = curl_exec($ch);
    If (curl_errno($ch) == 0) {
        $data = json_decode($result, true);
        } else {
        $data = false;
        echo "Error with connection to Elsevier: " . curl_errno($ch)
        . "\n";
        }
    curl_close($ch);
}

```

3.3 Алгоритм пошуку ключових слів

Найбільш основним і початковим кроком для аналізу обробки природної мови є виділення ключових слів, оскільки без вилучення ключових слів неможливо буде рухатися далі. Оскільки ми знаємо, що в НЛП ми отримуємо багато алгоритмів, які можуть допомогти нам у вилученні ключового слова для наших текстових даних.

Деякі з основних алгоритмів обробки тексту, які використовуються більшістю вчених даних і розробників машинного навчання

- TF-IDF
- TextRank
- RAKE

Rake, також відомий як швидке автоматичне вилучення ключових слів — є надзвичайно ефективним алгоритмом вилучення ключових слів, який працює з окремими документами, щоб уможливити застосування до динамічної колекції, його також можна дуже легко застосувати до нових

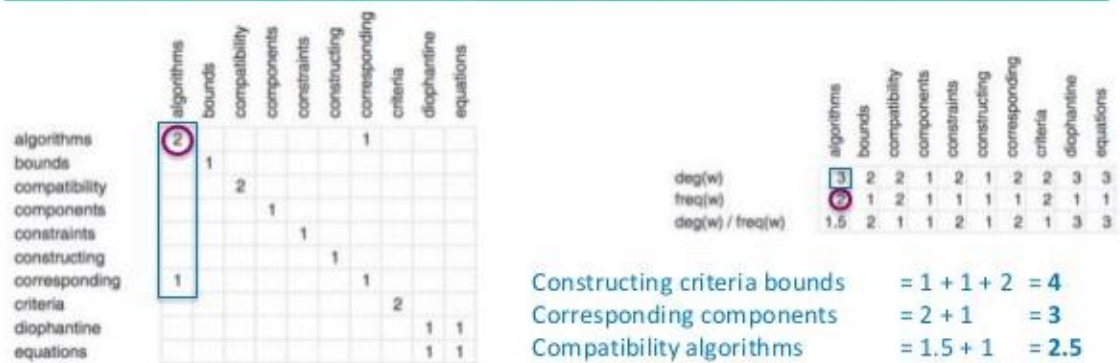
доменів, а також дуже ефективний при обробці кількох типів. документів, особливо тип тексту, який дотримується певних граматичних умов.

Rake базується на спостереженнях, що ключові слова часто містять кілька слів зі стандартними розділовими або стоп-словами, або ми можемо вимовляти функціональні слова, такі як «i», «of», «the» тощо з мінімальним лексичним значенням. Стоп-слова, як правило, відкидаються в усіх інформаційних системах, а також не включаються в різні аналізи тексту, оскільки вони вважаються безглуздими. Слова, які, як вважають, несуть значення, пов'язані з текстом, описуються як змістовні і називаються змістовними словами.

RAKE algorithm in one slide

"For search managers, developers & data scientists finding ways to innovate"

For **search managers, developers & data scientists** finding ways to innovate



Wolters Kluwer

Рисунок 3.13 — Опис алгоритму

Для реалізації алгоритму було обрано мову програмування Python. Він має такі переваги для використання:

- Гнучкість — це, на мою думку, основна перевага мови, оскільки завдяки своїй гнучкості мова набула популярності серед багатьох розробників.
- Розширюваність - один із слоганів мови звучить як - Just Import! — що повністю пояснює, наскільки мову розширюємо та було

розширено за останні роки. Існують бібліотеки та фреймворки під будь-який тип завдань та потреб. Також величезним плюсом є те, що ми можемо використовувати С-код з Python.

- Простота синтаксису. Синтаксис - це саме те, через що я закохався в Python, із синтаксису було прибрано все зайве, код чистий і зрозумілий без зайвих дужок та виразів.

Код реалізації алгоритму алгоритму можна побачити у додатку Д.

ВИСНОВКИ

В даній роботі було проведено аналіз вимог до інформаційної системи для аналізу фрагментів тексту на ключові слова та пошуку у наукометричній базі Scopus. Були визначені принципи методи та засоби, за допомогою яких буде створена архітектура та власне сам майбутній продукт.

Програмна реалізація включає в себе відправку запитів до API та подальша обробка відповіді серверу Scopus. Створена даталогічна модель даних та розроблено веб-інтерфейс системи. Доступ до системи мають тільки авторизовані користувачі.

Для реалізації веб-інтерфейсу був використаний фреймворк Bootstrap. Для серверної частини системи використана мова PHP. Реалізація алгоритму описана на скриптовій мові Python. В результаті роботи було отримано інформаційну веб-систему, яка швидко працює при великій кількості інформації, проста в дизайні і зрозуміла для користувача. Система повністю адаптивна до усіх пристроїв та працює на всіх сучасних браузерах. Доступ мають тільки авторизовані користувачі.

Проведено тестування роботи.

Дану систему надалі можна розширювати, масштабувати, не завдаючи шкоди існуючому функціоналу, а саме: можливе додавання більшої кількості типів метрик та розширення бази даних.

СПИСОК ЛІТЕРАТУРИ

1. API (Прикладний програмний інтерфейс) [Електронний ресурс]. – Режим доступу : <https://developer.mozilla.org/uk/docs/Glossary/API>
2. Науково-інформаційні та реферативні бази даних ХПУ [Електронний ресурс]. – Режим доступу : <http://www.nbu.gov.ua/node/1367>
3. Руководство по PHP [Електронний ресурс]. – Режим доступу: <https://www.php.net/manual/ru/intro-what-is.php>
4. Elsevier Developers [Електронний ресурс]. – Режим доступу: <https://dev.elsevier.com/>
5. Scopus Вікіпедія [Електронний ресурс]. – Режим доступу: <https://uk.wikipedia.org/wiki/Scopus>
6. Що таке RESTful API? [Електронний ресурс]. – Режим доступу: <https://codeguida.com/post/601>
7. Наукометричні бази даних, основні наукометричні показники, ідентифікація авторів? СНУ [Електронний ресурс]. – Режим доступу: <http://library.snu.edu.ua/index/naukometrija/0-12>
8. What is an API? [Електронний ресурс] – Режим доступу до ресурсу: <https://www.redhat.com/en/topics/api/what-are-application-programming-interfaces>.
9. Scopus LibGuide [Електронний ресурс] – Режим доступу до ресурсу: <https://elsevier.libguides.com/Scopus>.
10. Основні методи прогнозування: класифікація прогнозів [Електронний ресурс] – Режим доступу до ресурсу: https://pidru4niki.com/16400221/menedzhment/osnovni_metodi_prognozuvannya_klasifikatsiya_prognoziv.
11. Колисниченко Д. Н. PHP и MySQL. Разработка веб-приложений / Д. Н. Колисниченко., 2015. – 606 с

12. Мюллер Д. Базы данных та UML. проектування / Джон Мюллер., 2010. – 420 с. – (ЛЮРИ - М).

13. Indexes of citation and scientometrics of the database [Электронный ресурс] – Режим доступа до ресурсу:

<http://lib.nuos.edu.ua/%D1%96%D0%BD%D0%B4%D0%B5%D0%BA%D1%81%D0%B8->

<http://lib.nuos.edu.ua/%D1%86%D0%B8%D1%82%D1%83%D0%B2%D0%B0%D0%BD%D0%BD%D1%8F-%D1%82%D0%B0->

<http://lib.nuos.edu.ua/%D0%BD%D0%B0%D1%83%D0%BA%D0%BE%D0%BC%D0%B5%D1%82%D1%80%D1%96%D0%B9-%D0%B1%D0%B4/?lang=en>.

14. Что такое база данных? [Электронный ресурс] – Режим доступа до ресурсу: <https://www.oracle.com/ru/database/what-is-database/>.

15. Chapter 9. Recommendation systems - Stanford InfoLab [Электронный ресурс] – Режим доступа до ресурсу:

<http://infolab.stanford.edu/~ullman/mmds/ch9.pdf>.

Додаток А

Код сторінки index.php

```

1 <?php
2 require "auth.php";
3 ?>
4 <!DOCTYPE html>
5 <html>
6
7 <head>
8 <title></title>
9 <link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/4.0.0/css/bootstrap.min.css"
10 integrity="sha384-Gn5384xqQ1aoWXA+058RXPxPg6fy4IWvTNh0E263XmFcJlSAwiGgFAW/dAiS6JXm" crossorigin=
11 "anonymous">
12 <link href="https://fonts.googleapis.com/css2?family=Montserrat:wght@400;900&display=swap" rel=
13 "stylesheet">
14 </head>
15 <body>
16 <div class="container">
17 <p class="text-center p-5" style="font-family: 'Montserrat';font-size: 22px;">Web system of webometric
18 indicators of SSU in the scientometric database Scopus</p>
19 <form method="GET" action="connect.php">
20 <table class="table table-striped">
21 <thead>
22 <tr>
23 <th scope="col">Parameter</th>
24 <th scope="col">Value</th>
25 <th scope="col">Description</th>
26 </tr>
27 </thead>
28 <tbody>
29 <tr>
30 <td class="required"><label for="mmetrics0.57882158613877">metrics</label></td>
31 <td>
32 <select class="parameter required " name="metrics" id="mmetrics0.57882158613877">
33 <option value="CitationCount"> CitationCount </option>
34 <option value="CitationsPerPublication"> CitationsPerPublication </option>
35 <option value="CitedPublications"> CitedPublications </option>
36 <option value="FieldWeightedCitationImpact"> FieldWeightedCitationImpact </option>
37 <option value="ScholarlyOutput"> ScholarlyOutput </option>
38 </select>
39 </td>
40 <td>
41 <p>Names of the desired metrics</p>
42 </td>
43 </tr>
44 <tr>
45 <td class="code required"><label for="mentity0.706720277149572">entity</label></td>
46 <td>
47 <select class="parameter required" name="entity" id="mentity0.706720277149572">
48 <option selected="" value="Institution"> Institution (default) </option>
49 <option value="Author"> Author </option>
50 </select>
51 </td>
52 <td class="markdown">
53 <p><strong>Type of the requested entity</strong></p>
54 </td>
55 </tr>
56 <tr>
57 <td class="code required"><label for="mid0.6594872158411382">id</label></td>
58 <td>
59 <input class="parameter required" minlength="1" name="id" placeholder="(required)" id=
60 "mid0.6594872158411382" type="text" value="702756" required>
61 </td>
62 <td>
63 <strong><span class="markdown">
64 <p>Id of the requested entity</p>
65 </span></strong>
66 </td>
67 </tr>
68 <tr>
69 <td><label for="myearRange0.8096493667604483">yearRange</label></td>
70 <td>
71 <select class="parameter " name="yearRange" id="myearRange0.8096493667604483">
72 <option value=""></option>
73 <option value="3yrs"> 3yrs </option>
74 <option value="3yrsAndCurrent"> 3yrsAndCurrent </option>
75 <option value="3yrsAndCurrentAndFuture"> 3yrsAndCurrentAndFuture </option>
76 <option selected="" value="5yrs"> 5yrs (default) </option>
77 <option value="5yrsAndCurrent"> 5yrsAndCurrent </option>
78 <option value="5yrsAndCurrentAndFuture"> 5yrsAndCurrentAndFuture </option>
79 <option value="10yrs"> 10yrs </option>
80 </select>
81 </td>
82 <td>
83 <p>Year range of returned metric</p>
84 </td>
85 </tr>
86 <tr>
87 <td><label for="minincludeSelfCitations0.014460766339292386">includeSelfCitations</label></td>
88 <td>
89 <select class="parameter " name="includeSelfCitations" id=

```

```

84         <select class="parameter " name="includeSelfCitations" id=
85             "mincludeSelfCitations0.014460766339292386">
86             <option value=""></option>
87             <option selected="" value="true"> true (default) </option>
88             <option value="false"> false </option>
89         </select>
90     </td>
91     <td>
92         <p>Whether to include self-cited documents (not applicable to all metrics)</p>
93     </td>
94 </tr>
95 <tr>
96     <td class="code"><label for="mbyYear0.9595184413117974">byYear</label></td>
97     <td>
98         <select class="parameter " name="byYear" id="mbyYear0.9595184413117974">
99             <option selected="" value="true"> true (default) </option>
100         </select>
101     </td>
102     <td>
103         <p>Whether to return results broken down by year (not applicable to all metrics)</p>
104     </td>
105 </tr>
106 <tr>
107     <td class="code"><label for="mincludedDocs0.31360441734785005">includedDocs</label></td>
108     <td>
109         <select class="parameter " name="includedDocs" id="mincludedDocs0.31360441734785005">
110             <option value=""></option>
111             <option selected="" value="AllPublicationTypes"> AllPublicationTypes (default) </option>
112             <option value="ArticlesOnly"> ArticlesOnly </option>
113             <option value="ArticlesReviews"> ArticlesReviews </option>
114             <option value="ArticlesReviewsConferencePapers"> ArticlesReviewsConferencePapers </option>
115             <option value="ConferencePapersOnly"> ConferencePapersOnly </option>
116             <option value="ArticlesConferencePapers"> ArticlesConferencePapers </option>
117             <option value="ArticlesReviewsConferencePapersBooksAndBookChapters">
118                 ArticlesReviewsConferencePapersBooksAndBookChapters </option>
119         </select>
120     </td>
121     <td class="markdown">
122         <p>Publication types to include (not applicable to all metrics)</p>
123     </td>
124 </tr>
125 </tbody>
126 </table>
127 <button type="submit" class="btn btn-primary" style="width: 100%">Search</button>
128 </form>
129 <div>
130     <p class="text-center pt-5" style="font-family: 'Montserrat';font-size: 22px;">A recommendation
131     system for the Scopus scientometric database based on keywords</p>
132 </div>
133 <form method="post">
134     <div class="input-group py-3">
135         <span class="input-group-text">Enter your text:</span>
136         <textarea class="form-control" aria-label="With textarea" rows="8" name="text1"></textarea>
137     </div>
138     <button type="submit" class="btn btn-primary" style="width: 100%">Highlight keywords</button>
139 </form>
140 <div class="p-3">
141     <?php
142     if (isset($_POST['text1'])) {
143         $file = fopen('uploads/data.txt', 'w'); // открываем файл для добавления в конец
144         fputs($file, $_POST['text1']."\n"); // записываем содержимое текстовой области и добавляем
145         перенос на новую строку
146         fclose($file);
147         exec('C:\Users\PapiK\AppData\Local\Programs\Python\Python310\python.exe
148             C:\wamp64\www\ghlj\5.py');
149         //print_r($output);
150         $fd = fopen("xyz.txt", 'r') or die("не удалось открыть файл");
151         echo '<p class="text-center pt-5 font-weight-bold" style="font-family:
152             "Montserrat";font-size: 18px;">Keywords: </p>';
153         echo '<p class="text-center font-italic" style="font-family: "Montserrat";font-size:
154             18px;">';
155         while(!feof($fd))
156         {
157             $str = fgets($fd);
158             echo "$str <br>";
159         }
160         echo "</p>";
161         fclose($fd);
162         echo '<form action="search_ar.php"><button type="submit" class="btn btn-primary"
163             style="width: 100%">Search articles</button></form>';
164     }
165     ?>
166 </div>
167 <div class="pb-3">
168     <center><a href="index.php?do=logout" style="font-family: 'Montserrat';font-size: 18px;">Logout
169     </a></center>
170 </div>
171 <div class="col">
172     <center></center>
174 </div>
175 </div>
176 </body>
177 </html>

```

Додаток Б

Код скрипту connect.php

```

1 <!DOCTYPE html>
2 <html>
3 <head>
4 <title></title>
5 <link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/4.0.0/css/bootstrap.min.css"
  integrity="sha384-Gn5384xqQ1aoWXA+058RXPxPg6fy4IWvTNh0E263XmFcJlSAwiGgFAW/dAiS6JXm" crossorigin=
  "anonymous">
6 <link href="https://fonts.googleapis.com/css2?family=Montserrat:wght@400;900&display=swap" rel=
  "stylesheet">
7 </head>
8 <body>
9 <?php
10 require "auth.php";
11
12 $metrics = htmlspecialchars($_GET['metrics']);
13 $entity = htmlspecialchars($_GET['entity']);
14 $id = htmlspecialchars($_GET['id']);
15 $yearRange = htmlspecialchars($_GET['yearRange']);
16 $includeSelfCitations = htmlspecialchars($_GET['includeSelfCitations']);
17 $byYear = htmlspecialchars($_GET['byYear']);
18 $includedDocs = htmlspecialchars($_GET['includedDocs']);
19
20
21
22 $sch = curl_init('https://api.elsevier.com/analytics/scival/metrics/' . $metrics . '/' . $entity . '/' . $id .
  '?yearRange=' . $yearRange . '&includeSelfCitations='
23 . $includeSelfCitations . '&byYear=' . $byYear . '&includedDocs=' . $includedDocs .
  '&journalImpactType=CiteScore&showAsFieldWeighted=false&indexType=hIndex&apiKey=7f59af901d2d86f78a1fd60c1bf
  9426a');
24
25 curl_setopt($sch, CURLOPT_RETURNTRANSFER, 1);
26 curl_setopt($sch, CURLOPT_HTTPHEADER, array('Accept: application/json'));
27 curl_setopt($sch, CURLOPT_SSL_VERIFYPEER, FALSE);
28 $result = curl_exec($sch);
29
30 if (curl_errno($sch) == 0) {
31     $data = json_decode($result, true);
32 } else {
33     $data = false;
34     echo "Error with connection to Elsevier: " . curl_errno($sch) . "\n";
35 }
36
37 curl_close($sch);
38
39 $mysqli = new mysqli('localhost', 'root', '', 'scopus');
40 if ($mysqli->connect_errno) {
41     echo 'error with connection to db ';
42     echo "Номер ошибки: " . $mysqli->connect_errno . "\n";
43 }
44
45 $years = array_keys($data['results']['0']['metric']['valueByYear']);
46 $values = array_values($data['results']['0']['metric']['valueByYear']);
47
48
49
50 $sql = "SELECT `organization_id` FROM `organization` WHERE name = '" . $data['results']['0']['institution']
  [['name']] . "'";
51 $result = $mysqli->query($sql);
52 $organization_id = mysqli_fetch_array($result);
53 mysqli_free_result($result);
54
55 $sql = "SELECT `metric_id` FROM `metric` WHERE name = '" . $data['results']['0']['metric']['metricType'] .
  "'";
56 $result = $mysqli->query($sql);
57 $metric_id = mysqli_fetch_array($result);
58 mysqli_free_result($result);
59
60
61
62 //echo 'success';
63 ?>
64 <div class="container">
65 <p class="text-center p-5" style="font-family: 'Montserrat';font-size: 22px;">Web system of
  webometric indicators of SSU in the scientometric database Scopus</p>
66 <table class="table table-striped">
67 <thead>
68 <tr>
69 <th scope="col">Year</th>
70 <th scope="col">Organization</th>

```

```

70         <th scope="col">Organization</th>
71         <th scope="col">Metric</th>
72         <th scope="col">Value</th>
73
74     </tr>
75 </thead>
76 <tbody>
77     <?php
78     foreach ($years as $key => $value) {
79         $sql = "INSERT INTO `value` (`value`, `organization`, `year`, `metric_id`) VALUES ('.$values[$key].',".
80         $organization_id[0].",STR_TO_DATE('".$value."', '%Y')".$metric_id[0].")";
81         if ($mysqli->query($sql) === TRUE) {
82             echo "<tr>
83                 <td>".$value."</td>
84                 <td>".$data['results'][0]['institution']['name']."</td>
85                 <td>".$data['results'][0]['metric']['metricType']."</td>
86                 <td>".$values[$key]."</td>
87             </tr>";
88         } else {
89             echo "</tbody>
90             </table>Error: " . $sql . "<br>" . $mysqli->error;
91         }
92     }
93     <?>
94 </tbody>
95 </table>
96 <button type="submit" class="btn btn-primary" onclick="history.back();" style="width: 100%>
97     Back</button>
98 <div class="pb-3">
99     <center><a href="index.php?do=logout" style="font-family: 'Montserrat';font-size: 18px;">Logout
100     </a></center>
101 </div>
102 <div class="col">
103     <center></center>
105 </div>
</body>
</html>

```

Додаток В

Сторінка авторизації enter.php

```

1 <?php
2 session_start();
3
4 if(isset($_SESSION['admin'])){
5     header("Location: index.php");
6     exit;
7 }
8 ?>
9
10
11 <!DOCTYPE html>
12 <html>
13 <head>
14     <title></title>
15     <link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/4.0.0/css/bootstrap.min.css"
16         integrity="sha384-Gn5384xqQ1aoWXA+058RXPxPg6fy4IWvTNh0E263XmFcJlSAwiGgFAW/dAiS6JXm" crossorigin=
17         "anonymous">
18     <link href="https://fonts.googleapis.com/css2?family=Montserrat:wght@400;900&display=swap" rel=
19         "stylesheet">
20 </head>
21 <body>
22 <div class="container">
23     <p class="text-center p-5" style="font-family: 'Montserrat';font-size: 22px;">Web system of webometric
24     indicators of SSU in the scientometric database Scopus</p>
25     <div class="row">
26         <div class="col-sm" >
27             <form method="post" class="mx-auto my-5 border border-light p-5" style="max-width: 400px">
28                 <div class="form-group">
29                     <label for="exampleInputEmail1">Login</label>
30                     <input type="text" class="form-control" id="exampleInputEmail1" aria-describedby=
31                         "emailHelp" name="user">
32                 </div>
33                 <div class="form-group">
34                     <label for="exampleInputPassword1">Password</label>
35                     <input type="password" class="form-control" id="exampleInputPassword1" name="pass">
36                 </div>
37                 <input type="submit" class="btn btn-primary" name="submit" style="width: 100%"/>
38             </form>
39         </div>
40     </div>
41 <?php
42 $mysqli = new mysqli('localhost', 'root', '', 'scopus');
43 if ($mysqli->connect_errno) {
44     echo 'error with connection to db ';
45     echo "Номер ошибки: " . $mysqli->connect_errno . "\n";
46 }
47 if(isset($_POST['submit'])){
48     $sql = "Select id from users where log = '".$_POST['user']."' and pass = '".md5(md5($_POST['pass']))."'";
49     $result = $mysqli->query($sql);
50     if(mysqli_num_rows($result) > 0){
51         $_SESSION['admin'] = $_POST['user'];
52         header("Location: index.php");
53         exit;
54     }else echo '<p><center>Логин или пароль неверны!</center></p>';
55 }
56 $mysqli->close();
57 ?>
58 </div>
59 <div class="col">
60     <center></center>
62 </div>
63 </body>
64 </html>

```

Додаток Г

Код сторінки search_ar.php

```

1 <?php
2 require "auth.php";
3 ?>
4 <!DOCTYPE html>
5 <html>
6
7 <head>
8 <title></title>
9 <link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/4.0.0/css/bootstrap.min.css"
  integrity="sha384-Gn5384xqQ1aoWXA+058RXPxPg6fy4IWvTNh0E263XmFcJ1SAwiGgFAW/dAiS6JXm" crossorigin=
  "anonymous">
10 <link href="https://fonts.googleapis.com/css2?family=Montserrat:wght@400;900&display=swap" rel=
  "stylesheet">
11 </head>
12 <body>
13 <div class="container">
14 <?php
15 $keywords = file("xyz.txt");
16
17 foreach ($keywords as $line_num => $keyword) {
18 $keyword = str_replace(' ', '%20', $keyword);
19 $keyword = trim($keyword);
20 //echo $keyword;
21 $ch = curl_init('https://api.elsevier.com/content/search/scopus?query=title%7B'. $keyword.
  '%7D&apiKey=7f59af901d2d86f78a1fd60c1bf9426a&count=3&sort=citedby-count');
22
23 curl_setopt($ch, CURLOPT_RETURNTRANSFER, 1);
24 curl_setopt($ch, CURLOPT_HTTPHEADER, array('Accept: application/json'));
25 curl_setopt($ch, CURLOPT_SSL_VERIFYPEER, FALSE);
26 $result = curl_exec($ch);
27
28 If (curl_errno($ch) == 0) {
29 $data = json_decode($result, true);
30 } else {
31 $data = false;
32 echo "Error with connection to Elsevier: " . curl_errno($ch) . "\n";
33 }
34 curl_close($ch);
35 //echo $data['search-results']['opensearch:Query']['@searchTerms'];
36 if($data['search-results']['opensearch:totalResults'] > 0){
37 //echo $data['search-results']['entry']['1']['dc:creator'];
38 foreach ($data['search-results']['entry'] as $line_num => $key) {
39 echo'
40 <br>
41 <div>
42 <table class="table table-striped table-sm">
43 <thead>
44 <tr>
45 <th scope="col" colspan="2" ><p class="text-center p-2" style="font-size: 18px;">'.
  $data['search-results']['entry'][$line_num]['dc:title'].'</p></th>
46 </tr>
47 </thead>
48 <tbody>
49 <tr>
50 <th scope="row">Title:</th>
51 <td>'. $data['search-results']['entry'][$line_num]['dc:title'].'</td>
52 </tr>
53 <tr>
54 <th scope="row">Publication Name:</th>
55 <td>'. $data['search-results']['entry'][$line_num]['prism:publicationName'].'</td>
56 </tr>
57 <tr>
58 <th scope="row">Publication Date:</th>
59 <td>'. $data['search-results']['entry'][$line_num]['prism:coverDisplayDate'].'</td>
60 </tr>
61 <tr>
62 <th scope="row">Author:</th>
63 <td>'. $data['search-results']['entry'][$line_num]['dc:creator'].'</td>
64 </tr>
65 <tr>
66 <th scope="row">Link:</th>
67 <td><a href="'. $data['search-results']['entry'][$line_num]['prism:url'].' ">'. $data[
  'search-results']['entry'][$line_num]['prism:url'].'</a>
68 </td>
69 </tr>
70 </div>

```

```
70     </tr>
71     <tr>
72         <th scope="row">Cited by count:</th>
73         <td>'. $data['search-results']['entry'][$line_num]['citedby-count'],'</td>
74     </tr>
75 </tbody>
76 </table>
77 </div>
78     ';
79     }
80     //echo $result;
81 }
82 }
83 ?>
84 </div>
85 <div class="pb-3">
86     <center><a href="index.php?do=logout" style="font-family: 'Montserrat';font-size: 18px;">Logout
87     </a></center>
88 </div>
89 <div class="col">
90     <center></center>
92 </div>
</body>
</html>
```

Додаток Д

Код алгоритму

```

1  from __future__ import division
2  import operator
3  import nltk
4  import string
5
6  def isPunct(word):
7      return len(word) == 1 and word in string.punctuation
8
9  def isNumeric(word):
10     try:
11         float(word) if '.' in word else int(word)
12         return True
13     except ValueError:
14         return False
15
16     class RakeKeywordExtractor:
17
18     def __init__(self):
19         self.stopwords = set(nltk.corpus.stopwords.words())
20         self.top_fraction = 1 # consider top third candidate keywords by score
21
22     def _generate_candidate_keywords(self, sentences):
23         phrase_list = []
24         for sentence in sentences:
25             words = map(lambda x: "|" if x in self.stopwords else x,
26                         nltk.word_tokenize(sentence.lower()))
27             phrase = []
28             for word in words:
29                 if word == "|" or isPunct(word):
30                     if len(phrase) > 0:
31                         phrase_list.append(phrase)
32                         phrase = []
33                     else:
34                         phrase.append(word)
35             return phrase_list
36
37     def _calculate_word_scores(self, phrase_list):
38         word_freq = nltk.FreqDist()
39         word_degree = nltk.FreqDist()
40         for phrase in phrase_list:
41             degree = len(filter(lambda x: not isNumeric(x), phrase)) - 1
42             for word in phrase:
43                 word_freq.inc(word)
44                 word_degree.inc(word, degree) # other words
45         for word in word_freq.keys():
46             word_degree[word] = word_degree[word] + word_freq[word] # itself
47         # word score = deg(w) / freq(w)
48         word_scores = {}
49         for word in word_freq.keys():
50             word_scores[word] = word_degree[word] / word_freq[word]
51         return word_scores
52
53     def _calculate_phrase_scores(self, phrase_list, word_scores):
54         phrase_scores = {}
55         for phrase in phrase_list:
56             phrase_score = 0
57             for word in phrase:
58                 phrase_score += word_scores[word]
59             phrase_scores[" ".join(phrase)] = phrase_score
60         return phrase_scores
61
62     def extract(self, text, incl_scores=False):
63         sentences = nltk.sent_tokenize(text)
64         phrase_list = self._generate_candidate_keywords(sentences)
65         word_scores = self._calculate_word_scores(phrase_list)
66         phrase_scores = self._calculate_phrase_scores(
67             phrase_list, word_scores)
68         sorted_phrase_scores = sorted(phrase_scores.iteritems(),
69                                     key=operator.itemgetter(1), reverse=True)
70         n_phrases = len(sorted_phrase_scores)

```



```
70     n_phrases = len(sorted_phrase_scores)
71     if incl_scores:
72         return sorted_phrase_scores[0:int(n_phrases/self.top_fraction)]
73     else:
74         return map(lambda x: x[0],
75                   sorted_phrase_scores[0:int(n_phrases/self.top_fraction)])
76
77 def test():
78     rake = RakeKeywordExtractor()
79     keywords = rake.extract("""
80 Compatibility of systems of linear constraints over the set of natural
81 numbers. Criteria of compatibility of a system of linear Diophantine
82 equations, strict inequations, and nonstrict inequations are considered.
83 Upper bounds for components of a minimal set of solutions and algorithms
84 of construction of minimal generating sets of solutions for all types of
85 systems are given. These criteria and the corresponding algorithms for
86 constructing a minimal supporting set of solutions can be used in solving
87 all the considered types of systems and systems of mixed types.
88 """, incl_scores=True)
89     print keywords
90
91 if __name__ == "__main__":
92     test()
```