

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Таврійський державний агротехнологічний університет
імені Дмитра Моторного
Факультет енергетики і комп'ютерних технологій

ЗАТВЕРДЖУЮ

Зав. каф. "Комп'ютерні науки"

доц. _____ **Сергій ШАРОВ**

«16» червня 2025 р.

Пояснювальна записка

до кваліфікаційної роботи здобувача СВО Бакалавр

(ступінь вищої освіти)

на тему: «Мобільний сервіс для автоматизації формування бібліографії
наукових публікацій»

52/4КНД.11323044.000000ПЗ

Виконав: здобувач вищої освіти 4 курсу,

групи 21С(МС)КН

спеціальності 122 Комп'ютерні науки

за ОПП Комп'ютерні науки

(шифр і назва спеціальності та ОПП)

_____ **Євген БЄЛІКОВ**

(підпис)

Керівник доц. _____ **Дмитро ЛУБКО**

(підпис)

Консультант _____ **Лариса БОЛТЯНСЬКА**

(підпис)

Консультант _____ **Михайло ЗОРЯ**

(підпис)

Нормоконтроль, ст. викл. _____ **Ольга ЗИНОВ'ЄВА**

(підпис)

Рецензент _____ **Олексій НАУМУК**

(підпис)

Запоріжжя – 2025 рік

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

Таврійський державний агротехнологічний університет імені Дмитра Моторного

Факультет: Енергетики і комп'ютерних технологій

Кафедра: Комп'ютерні науки

Ступінь вищої освіти: Бакалавр

Спеціальність: 122 Комп'ютерні науки

ОПП: Комп'ютерні науки

ЗАТВЕРДЖУЮ
Завідувач кафедри КН
к.пед.н., доц. _____ Сергій ШАРОВ
“10” жовтня 2024 року

З А В Д А Н Н Я НА КВАЛІФІКАЦІЙНУ РОБОТУ ЗДОБУВАЧУ ВИЩОЇ ОСВІТИ

Бєлікову Євгену Юрійовичу

(прізвище, ім'я, по батькові)

1. Тема кваліфікаційної роботи «Мобільний сервіс для автоматизації формування бібліографії наукових публікацій»

керівник проєкту Лубко Дмитро Вікторович, к.т.н., доцент

(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)

затверджені наказом університету від “30” вересня 2024 року № 452-С

2. Строк подання здобувачем вищої освіти роботи 12 червня 2025

3. Вихідні дані до роботи: дані обстеження об'єкту, статистичні дані, технічне завдання на дипломне проєктування, матеріали виробничих практик, нормативні документи, науково-технічна література, електронні ресурси та ін.

4. Зміст пояснювальної записки (перелік питань, які потрібно розробити)

1. Опис предметної області та аналіз існуючих рішень.
2. Специфікація вимог до експертної системи.
3. Вибір та обґрунтування технологій розробки експертної системи.
4. Дослідна експлуатація експертної системи.
5. Тестування експертної системи.

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

Діаграма використання, діаграми IDEF0, розкадровка інтерфейсу, структурна схема, візуальні компоненти інтерфейсу, алгоритм роботи, інтерфейс програми
Презентація - 11 слайдів

1. Титульний слайд

2. Предмет, об'єкт, мета
3. Завдання дослідження
4. Порівняльна характеристика аналогів програмного продукту
5. Діаграма варіантів використання
6. Інструментальні засоби
7. Інтерфейс мобільного сервісу
8. Інтерфейс і робота мобільного сервісу
9. Тестування
10. Висновок
11. Дякую за увагу

6. Консультанти розділів кваліфікаційної роботи

Розділ/ Підрозділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв
4.3	Болтянська Л. О.	15.10.2025	15.06.2025
4.4	Зоря М. В.	15.10.2025	15.06.2025

7. Дата видачі завдання 10 жовтня 2024 р.

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів кваліфікаційної роботи	Строк виконання етапів кваліфікаційної роботи	Примітка
1	Аналіз предметної області	03.10.2024	Виконано
2	Специфікація вимог до експертної системи	17.10.2024	Виконано
3	Проектування експертної системи	28.10.2024	Виконано
4	Обґрунтування інструментальних засобів	18.11.2024	Виконано
5	Розробка експертної системи	16.12.2024	Виконано
6	Тестування та налагодження експертної системи	11.04.2025	Виконано
7	Підпис керівником роботи	14.06.2025	Виконано
8	Підпис завідувачем кафедри	16.06.2025	Виконано

Здобувач вищої освіти

(підпис) Євген БЄЛІКОВ
(власне ім'я та прізвище)

**Керівник
кваліфікаційної
роботи**

(підпис) Дмитро ЛУБКО
(власне ім'я та прізвище)

РЕФЕРАТ

Кваліфікаційна робота бакалавра: 95 с., 25 рис., 11 табл., 25 джерел, 2 додатки.

Актуальність роботи зумовлена сучасними викликами у сфері наукового цитування, зокрема недостатньою підтримкою українських стандартів цитування у існуючих бібліографічних менеджерах.

У багатьох відомих рішеннях відсутній зручний мобільний доступ до функцій, що значно ускладнює роботу користувачів у польових умовах або поза робочим місцем.

Крім того, більшість популярних бібліографічних сервісів мають складні та непрозорі інтерфейси, що знижує їхню зручність і доступність для студентів і науковців.

Висока вартість комерційних продуктів є ще одним бар'єром, що обмежує широку доступність таких інструментів.

У зв'язку з цим створення спеціалізованого мобільного онлайн-сервісу набуває особливої важливості, оскільки він може значно спростити і прискорити процес формування бібліографії, забезпечуючи водночас відповідність як українським, так і міжнародним стандартам.

Об'єктом дослідження є процеси формування бібліографічних посилань та списків літератури у наукових публікаціях, що є невід'ємною частиною академічної діяльності.

Предметом дослідження виступає мобільний онлайн-сервіс, який розробляється з метою автоматизації формування бібліографії, що значно підвищує ефективність і точність оформлення посилань.

Головною метою роботи є створення такого сервісу, який забезпечить підтримку різних стилів цитування та адаптивність інтерфейсу для зручного використання на мобільних пристроях.

Задачі дослідження:

- аналіз існуючих рішень для управління бібліографією;
- розробка технічного завдання та функціональної моделі;
- проектування архітектури та об'єктної моделі системи;
- реалізація алгоритмів форматування за стандартами ДСТУ 8302:2015, APA та IEEE;
- створення адаптивного інтерфейсу з підтримкою мобільних пристроїв;
- впровадження функціоналу прогресивного веб-додатку (PWA).

Основні характеристики розробленого рішення:

- архітектура PWA для забезпечення офлайн-функціональності;
- підтримка українського стандарту ДСТУ 8302:2015 та міжнародних стилів цитування;
- адаптивний інтерфейс з підтримкою світлої та темної тем;
- модульна організація JavaScript-коду;
- мобільно-орієнтований дизайн з оптимізацією для сенсорних пристроїв.

Розроблений онлайн-сервіс дозволяє суттєво скоротити часові витрати на оформлення бібліографії та мінімізувати ризики помилок, що підтверджує актуальність розробки.

Ключові слова: бібліографічне посилання, стиль цитування, ДСТУ 8302:2015, мобільний онлайн-сервіс, прогресивний веб-додаток, адаптивний дизайн, автоматизація форматування, JavaScript, PWA, мобільна оптимізація.

SUMMARY

Bachelor's thesis: 95 p., 25 fig., 11 tables, 25 sources, and 2 appendices.

The relevance of the thesis is driven by contemporary challenges in the field of scientific citation, particularly the insufficient support for Ukrainian citation standards in existing bibliographic management tools.

Many well-known solutions lack convenient mobile access to essential functions, which significantly complicates the work of users in the field or outside the workplace.

Furthermore, most popular bibliographic services have complex and non-intuitive interfaces, reducing their usability and accessibility for students and researchers.

The high cost of commercial products is another barrier limiting widespread access to such tools.

Therefore, the creation of a specialized mobile online service becomes particularly important, as it can greatly simplify and accelerate the bibliography formation process while ensuring compliance with both Ukrainian and international standards.

The object of research is the processes of forming bibliographic references and bibliographies in scientific publications, which are an integral part of academic activities.

The subject of research is the mobile online service developed to automate bibliography formation, thereby improving the efficiency and accuracy of citation formatting.

The main purpose of the thesis is to create a service that supports various citation styles and features an adaptive interface for convenient use on mobile devices.

Research tasks:

- analyze existing bibliography management solutions;
- develop technical specifications and a functional model;
- design of the system's architecture and object model;
- implementation of formatting algorithms for DSTU 8302:2015, APA, and IEEE standards;
- creation of an adaptive interface with support for mobile devices;
- implementation of Progressive Web Application (PWA) functionality.

Main characteristics of the developed solution:

- PWA architecture providing offline functionality;
- support for the Ukrainian DSTU 8302:2015 standard and international citation styles;
- adaptive interface with light and dark themes;
- modular organization of JavaScript code;
- mobile-first design optimized for touch devices.

The developed online service significantly reduces the time spent on formatting bibliography and minimizes the risk of errors, which confirms the relevance and practical value of the solution.

Keywords: bibliographic reference, citation style, DSTU 8302:2015, mobile online service, progressive web application, adaptive design, formatting automation, JavaScript, PWA, mobile optimization.

ЗМІСТ

ВСТУП	9
РОЗДІЛ 1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ ТА ІСНУЮЧИХ ПРОГРАМНИХ РІШЕНЬ	11
1.1 Узагальнена характеристика предметної області	11
1.2 Огляд і аналіз існуючих аналогів системи	14
1.3 Розробка технічного завдання	23
РОЗДІЛ 2 СПЕЦИФІКАЦІЯ ВИМОГ ДО ІНФОРМАЦІЙНОЇ СИСТЕМИ	30
2.1 Глосарій.....	30
2.2 Концептуальна модель використання інформаційної системи	31
2.3 Розробка функціональної моделі.....	33
РОЗДІЛ 3 ОПИС ПРИЙНЯТИХ ПРОЄКТНИХ ТА ТЕХНОЛОГІЧНИХ РІШЕНЬ.....	39
3.1 Розробка об'єктної моделі	39
3.2 Розробка архітектури.....	40
3.3 Проєктування інтерфейсу програмної системи	42
3.4 Обґрунтування вибору мови програмування	48
3.5 Опис програмної реалізації.....	53
РОЗДІЛ 4 ДОСЛІДНА ЕКСПЛУАТАЦІЯ ІНФОРМАЦІЙНОЇ СИСТЕМИ	59
4.1 Інструкція користувача ІС.....	59
4.2 Основи тестування веб-додатків	67
4.3 Техніко-економічні показники ІС. Ошибка! Закладка не определена.	
4.4 Охорона праці.....	79
ВИСНОВКИ	85
СПИСОК ЛІТЕРАТУРИ	86
ДОДАТОК А	89
ДОДАТОК Б.....	90

ВСТУП

Актуальність теми дослідження. В умовах стрімкого розвитку інформаційних технологій зростає потреба у зручних і доступних інструментах для науковців, які забезпечують не лише коректне оформлення посилань, а й інтеграцію з мобільними пристроями.

Сучасний науковий процес вимагає ефективних рішень, що сприяють підвищенню якості та оперативності роботи дослідників у різних галузях.

Це сприяє не лише уніфікації оформлення, а й підвищенню престижу української науки на міжнародній арені.

Врахування українських стандартів цитування є важливим для збереження національної наукової ідентичності та підвищення якості наукових публікацій.

Це ускладнює роботу дослідників і студентів, які потребують швидких і зручних рішень для оформлення бібліографії.

Існуючі бібліографічні менеджери мають обмежену підтримку українських стандартів, складні інтерфейси, недостатню мобільність і високу вартість, що ускладнює роботу науковців.

Автоматизація формування бібліографії за допомогою мобільного сервісу значно скорочує час оформлення посилань.

Об'єкт дослідження – процеси формування бібліографічних посилань у наукових публікаціях.

Предмет дослідження – мобільний онлайн-сервіс для автоматизації формування бібліографії, його функції, інтерфейс і алгоритми форматування.

Мета дослідження – розробити мобільний онлайн-сервіс для автоматичного створення бібліографічних посилань за стандартами (ДСТУ 8302:2015, APA, IEEE) з урахуванням мобільного використання та підтримки українських норм.

Завдання дослідження:

1. Проаналізувати існуючі рішення, виявити їхні недоліки;
2. Розробити технічне завдання і функціональну модель;
3. Спроекувати архітектуру системи з урахуванням мобільності;
4. Реалізувати алгоритми форматування за трьома стандартами;
5. Створити адаптивний інтерфейс для мобільних пристроїв;
6. Впровадити PWA для офлайн-режиму;
7. Провести тестування та оптимізацію.

Методи дослідження:

1. Теоретичний аналіз літератури та існуючих рішень;
2. Моделювання архітектури;
3. Розробка прогресивного веб-додатку;
4. Тестування і порівняльний аналіз ефективності.

Практична значущість полягає у створенні зручного інструменту для автоматизації бібліографії з підтримкою українських стандартів, що зменшує час оформлення, мінімізує помилки та забезпечує мобільний доступ і офлайн-роботу.

Сервіс може застосовуватися у вишах, наукових установах і видавництвах.

Структура роботи:

Робота складається з чотирьох розділів:

1. Аналіз предметної області та існуючих рішень — огляд і обґрунтування теми;
2. Специфікація вимог — опис вимог і моделей системи;
3. Проєктування — архітектура, інтерфейс і технології розробки;
4. Тестування — перевірка функціональності та ефективності розробленого сервісу

РОЗДІЛ 1. АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ ТА ІСНУЮЧИХ ПРОГРАМНИХ РІШЕНЬ

1.1 Узагальнена характеристика предметної області

Вступ до предметної області

Предметна область даного дослідження охоплює процеси формування бібліографічних посилань та списків літератури у наукових публікаціях. Правильне використання цитат та оформлення посилань у наукових роботах є важливою складовою дотримання принципів академічної доброчесності та сприяє якісному інтегруванню українських дослідників у світову систему наукової комунікації [8].

У науковому середовищі існують різноманітні стандарти оформлення бібліографічних посилань, які відрізняються залежно від галузі науки, географічного регіону та вимог конкретних наукових видань. Серед найбільш поширених стандартів - український ДСТУ 8302:2015, міжнародні стилі APA (American Psychological Association) та IEEE (Institute of Electrical and Electronics Engineers). Кожен з цих стандартів встановлює специфічні правила щодо порядку розміщення авторів, формату дат, пунктуації та інших елементів бібліографічного опису.

Опис завдань, що вирішуються у предметній області

У сфері формування бібліографічних посилань науковці та дослідники стикаються з низкою завдань:

- коректне форматування бібліографічних описів відповідно до вимог різних стилів цитування, що часто потребує спеціальних знань і навичок;
- управління бібліографічною інформацією - збір, зберігання та організація даних про наукові джерела;

- конвертування між стилями цитування - необхідність переформатування бібліографічних описів при підготовці публікацій для різних видань;
- забезпечення точності бібліографічних даних для уникнення помилок у наукових комунікаціях;
- оптимізація часових витрат на оформлення списків літератури, особливо при роботі з великою кількістю джерел;

Особливої актуальності ці завдання набувають в умовах зростання кількості наукових публікацій та підвищення вимог до їх оформлення.

Для науковців, особливо початківців, дотримання всіх вимог стандартів цитування може бути складним і трудомістким процесом

Створення та оформлення бібліографічних посилань є трудомістким процесом, що вимагає уваги до деталей.

Щоб полегшити цю задачу, сучасні технології пропонують низку інструментів, які автоматизують і спрощують роботу з бібліографією [10].

Опис наявних рішень та огляд ринку програмних продуктів

На ринку програмних продуктів для управління бібліографією існують різні типи рішень:

- бібліографічні менеджери - програми, розроблені для зберігання бібліографічних даних, повних текстів і оформлення посилань і списків літератури [1] (Mendeley, Zotero, EndNote);
- онлайн-генератори цитат - веб-сервіси для швидкого створення окремих бібліографічних описів (Citation Machine);
- плагіни для текстових редакторів - інтегровані рішення для додавання цитат безпосередньо у текст документа (Zotero для Word, Mendeley Cite);
- мобільні додатки - спрощені версії бібліографічних менеджерів для використання на мобільних пристроях.

Ці рішення варіюються за функціональністю, зручністю використання, ціною та підтримкою локальних стандартів цитування.

Однак, незважаючи на різноманітність доступних інструментів, існує потреба у спеціалізованих рішеннях, орієнтованих на мобільне використання та українські стандарти цитування.

Обґрунтування актуальності розробки.

Актуальність розробки мобільного онлайн-сервісу для автоматизації формування бібліографії наукових публікацій обумовлена такими факторами:

- недостатня підтримка українських стандартів цитування у існуючих рішеннях, особливо ДСТУ 8302:2015, що створює труднощі для українських науковців;
- обмежений мобільний доступ до повноцінних функцій існуючих бібліографічних менеджерів, що не відповідає сучасним тенденціям використання мобільних пристроїв у професійній діяльності;
- складність інтерфейсів більшості наявних рішень, що потребує тривалого освоєння і не підходить для швидкого оформлення окремих посилань;
- висока вартість комерційних рішень з повним функціоналом, що обмежує доступ до них для багатьох українських науковців та студентів.

Розробка спеціалізованого мобільного онлайн-сервісу дозволить вирішити ці проблеми, надавши українським науковцям, студентам та редакторам зручний інструмент для автоматизації процесу формування бібліографії відповідно до національних та міжнародних стандартів.

Моделювання бізнес-процесів предметної області

Традиційний спосіб формування бібліографічних посилань включає ряд етапів, які потребують значних часових витрат та містять ризики помилок через людський фактор.

Автоматизація цього процесу дозволить значно скоротити час на оформлення та мінімізувати ризики помилок.

Для оцінки ефективності автоматизації процесу формування бібліографії розглянемо кількісні показники часових витрат на оформлення списку літератури різними методами (табл. 1.1).

Таблиця 1.1 - Порівняння часових витрат на формування бібліографії різними методами

Метод формування бібліографії	Час на оформлення одного джерела, хв	Час на оформлення списку з 30 джерел, хв	Ризик помилок
Ручне форматування	5-7	150-210	Високий
Використання десктопного бібліографічного менеджера	2-3	60-90	Середній
Використання онлайн-генератора цитат	1-2	30-60	Низький
Розроблюваний мобільний сервіс	0.5-1	15-30	Мінімальний

Як бачимо з таблиці 1.1, використання спеціалізованого мобільного сервісу для автоматизації формування бібліографії дозволяє значно скоротити часові витрати та мінімізувати ризики помилок, що підтверджує актуальність розробки.

1.2 Огляд і аналіз існуючих аналогів системи

Для об'єктивної оцінки необхідності розробки нового програмного продукту було проведено аналіз існуючих аналогів систем управління бібліографією, що представлені на ринку.

Особливу увагу приділено функціональності, зручності використання, мобільному доступу та підтримці українських стандартів цитування.

Mendeley - це безкоштовний універсальний бібліографічний менеджер та академічна соціальна мережа [24], розробка компанії Elsevier. Він дозволяє зберігати, організовувати та анотувати наукові публікації, а також генерувати списки літератури в різних стилях цитування.

Функціональні можливості:

- зберігання бібліографічних даних та PDF-файлів публікацій;
- організація джерел за колекціями та тегами;
- автоматичне вилучення метаданих з PDF-файлів;
- генерація списків літератури у різних стилях;
- інтеграція з MS Word та LibreOffice;
- синхронізація бібліотеки між пристроями;
- соціальні функції для взаємодії з іншими науковцями.

Недоліки:

- обмежена підтримка українських стандартів цитування, відсутність повноцінної підтримки ДСТУ 8302:2015;
- складний інтерфейс, що потребує значного часу для освоєння;
- обмежена функціональність мобільної версії, яка не дозволяє повноцінно користуватися всіма можливостями сервісу.

На рисунку 1.1 представлено інтерфейс бібліографічного менеджера Mendeley.

Zotero

Zotero - це безкоштовне програмне забезпечення для управління посиланнями з відкритим кодом, яке дозволяє зберігати та організовувати бібліографічні посилання, допомагає легко додавати джерела, автоматично генерувати цитати і бібліографії [25].

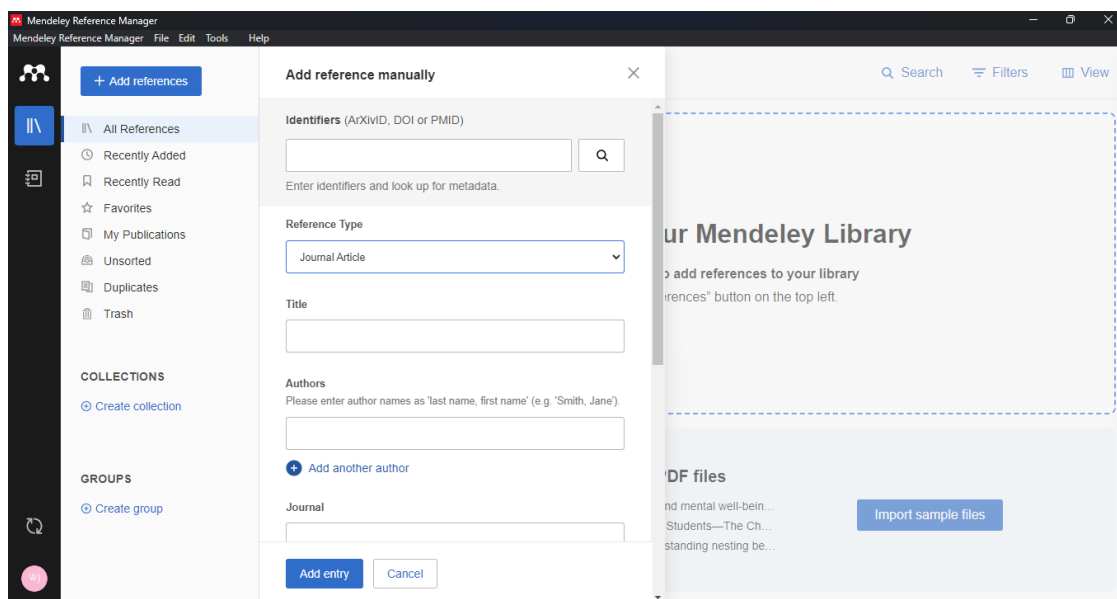


Рисунок 1.1 - Інтерфейс бібліографічного менеджера Mendeley

1. Функціональні можливості:

- автоматичний збір метаданих із веб-сторінок;
- організація джерел за колекціями та тегами;
- підтримка групової роботи та спільного доступу до бібліотек;
- інтеграція з текстовими редакторами;
- можливість створення користувацьких стилів цитування;
- відкритий програмний код, що дозволяє створювати додаткові розширення.

2. Недоліки:

- обмежене хмарне сховище у безкоштовній версії;
- відсутність повноцінної мобільної версії, лише веб-інтерфейс, який не оптимізований для мобільних пристроїв;
- складний процес налаштування користувацьких стилів цитування;
- недостатня локалізація для українських користувачів.

На рисунку 1.2 представлено інтерфейс бібліографічного менеджера Zotero

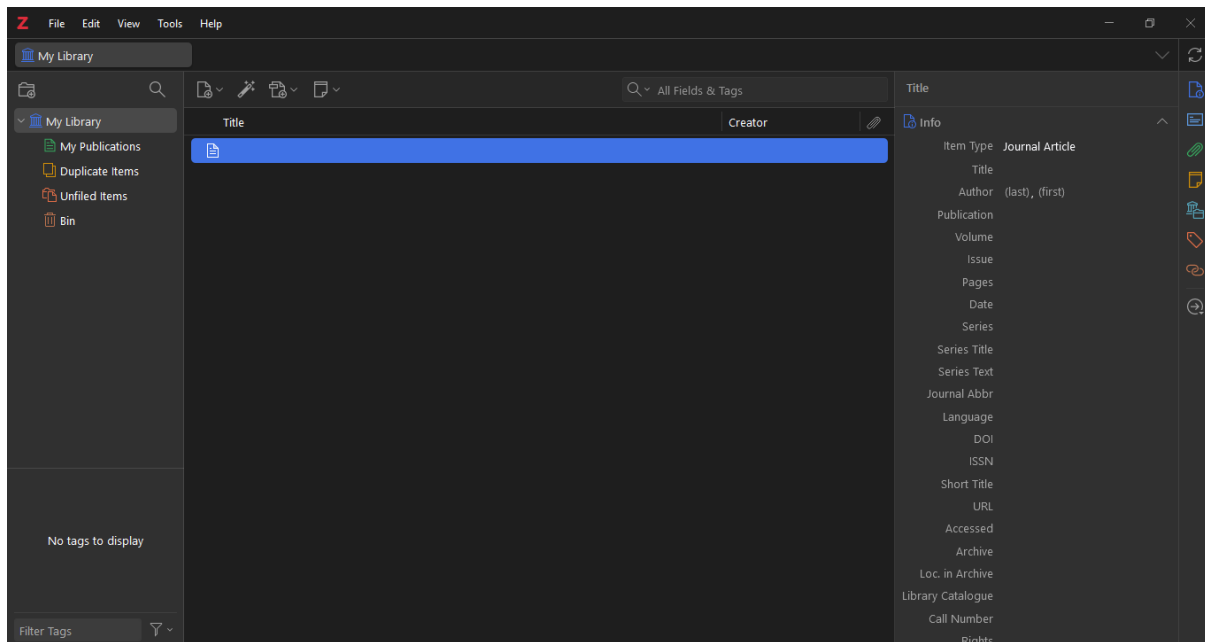


Рисунок 1.2 - Інтерфейс бібліографічного менеджера Zotero

EndNote - це комерційна система управління бібліографічною інформацією, що застосовується для управління посиланнями і бібліографією [16], розроблена компанією Clarivate Analytics, який широко використовується в академічному середовищі для управління бібліотеками джерел та створення списків літератури.

1. Функціональні можливості:

- збір метаданих із наукових баз даних;
- організація бібліотеки джерел;
- інтеграція з текстовими редакторами;
- велика база стилів цитування;
- спільний доступ до бібліотек для колективної роботи;
- розширені можливості пошуку та фільтрації джерел;

2. Недоліки:

- висока вартість ліцензії для повної версії;
- обмежена підтримка українських стандартів цитування;
- складний інтерфейс для початківців;
- обмежений мобільний доступ до повного функціоналу;

На рисунку 1.3 представлено інтерфейс бібліографічного менеджера EndNote.

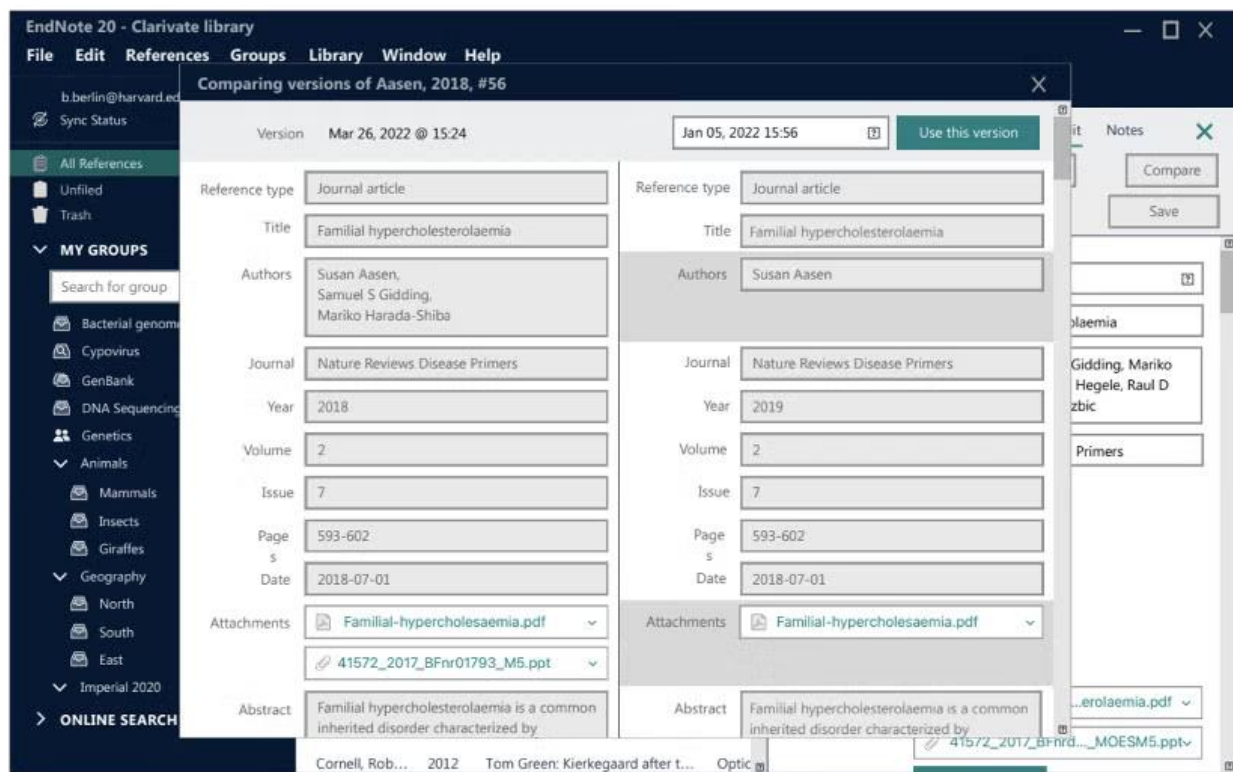


Рисунок 1.3 - Інтерфейс бібліографічного менеджера EndNote

Citation Machine - це онлайн-сервіс для спрощення створення бібліографічних посилань для академічних робіт, розроблений компанією Chegg [14]. Сервіс орієнтований на студентів та забезпечує форматування посилань за популярними стилями цитування.

1. Функціональні можливості:

- швидке генерування окремих бібліографічних посилань;
- підтримка основних міжнародних стилів цитування (APA, MLA, Chicago);
- простий інтерфейс, орієнтований на швидке створення посилань;
- наявність мобільної версії веб-сайту.

2. Недоліки:

- обмежений функціонал у безкоштовній версії;

- відсутність підтримки українських стандартів цитування;
- відсутність можливості зберігати бібліотеку джерел;
- наявність реклами, що ускладнює використання.

На рисунку 1.4 представлено інтерфейс онлайн-сервісу Citation Machine.

Finalize book details

In-Print E-book Online database

⚠ We recommend filling in the highlighted fields.

What I'm Citing

Citing

The whole book ▼

Contributors

Role	First Name	Middle Name	Last Name	Suffix
Author ▼				

➕ Add another contributor

In print publication info

Source title

Рисунок 1.4 - Інтерфейс онлайн-сервісу Citation Machine

Порівняльна характеристика аналогів

Для систематизації інформації про проаналізовані аналоги проведено їх порівняльну характеристику за ключовими параметрами, що є важливими для цільової аудиторії розроблюваного продукту (табл. 1.2).

Таблиця 1.2 - Порівняльна характеристика програмних продуктів

Види вимог	Mendeley	Zotero		EndNote	Citation Machine
1	2	3		4	5
Функціональні вимоги					
Підтримка ДСТУ 8302:2015	Частково	Через користувацькі стилі		Ні	Ні
Підтримка міжнародних стилів (APA, IEEE)	Так	Так		Так	Так
Зберігання бібліотеки джерел	Так	Так		Так	Обмежено
Організація джерел за категоріями	Так	Так	Так		Ні
Автоматичне вилучення метаданих	Так		Так	Так	Частково
Технічні вимоги					
Мобільний доступ	Обмежений		Обмежений	Обмежений	Так
Інтеграція з текстовими редакторами	Так		Так	Так	Ні
Експорт/імпорту даних	Так		Так	Так	Обмежено
Робота без встановлення	Ні	Через браузер		Веб-версія	Так
Комерційні вимоги					
Ціна	Платні функції	Безкоштовно	Платно		Безкоштовно з рекламою
Технічна підтримка	Так	Спільнота	Так		Обмежена

Продовження таблиці 1.2

1	2	3	4	5
Локалізація для українських користувачів	Часткова	Часткова	Ні	Ні
Ергономічні вимоги				
Простота інтерфейсу	Складний	Середній	Складний	Простий
Наявність довідкової інформації	Так	Так	Так	Обмежена
Інтуїтивність використання	Середня	Середня	Низька	Висока

Аналіз конкурентних переваг та недоліків існуючих рішень

Конкурентні переваги існуючих рішень, які варто врахувати при розробці нового продукту:

- багатofункціональність та інтеграція з науковими базами даних (EndNote, Mendeley);
- соціальні функції для наукової співпраці (Mendeley, Zotero);
- відкритий код та можливість розширення функціоналу спільнотою (Zotero);
- простота та швидкість формування окремих посилань (Citation Machine).

При цьому існують також спільні недоліки, які необхідно подолати в новому рішенні:

- недостатня підтримка українських стандартів цитування;
- обмежена функціональність на мобільних пристроях;
- складність інтерфейсу або, навпаки, обмеженість функціоналу;
- відсутність локалізації для українських користувачів;
- висновки щодо доцільності розробки нового програмного продукту.

Проведений аналіз існуючих аналогів систем управління бібліографією дозволяє зробити такі висновки:

Існуючі бібліографічні менеджери (Mendeley, Zotero, EndNote) є потужними інструментами для управління бібліотеками джерел, але вони мають складні інтерфейси, що потребують тривалого освоєння та не підходять для користувачів, які потребують лише швидкого форматування окремих посилань.

Жоден з проаналізованих аналогів не забезпечує повноцінної підтримки українського стандарту ДСТУ 8302:2015, що створює труднощі для українських науковців при оформленні публікацій.

Існуючі рішення мають обмежену функціональність на мобільних пристроях, що не відповідає сучасним тенденціям використання смартфонів у професійній діяльності.

Комерційні рішення з повним функціоналом мають високу вартість, що обмежує їх доступність для багатьох українських студентів та науковців.

Таким чином, розробка спеціалізованого мобільного онлайн-сервісу для автоматизації формування бібліографії наукових публікацій є актуальною та доцільною.

Новий програмний продукт має зосередитися на:

- забезпеченні повної підтримки українського стандарту ДСТУ 8302:2015 та популярних міжнародних стилів цитування;
- створенні простого та інтуїтивно зрозумілого інтерфейсу, орієнтованого на мобільне використання;
- оптимізації процесу форматування бібліографічних посилань для економії часу користувачів.

Це дозволить надати українським науковцям, студентам та редакторам зручний інструмент для автоматизації процесу формування бібліографії, що відповідає їхнім специфічним потребам.

1.3 Розробка технічного завдання

На основі проведеного аналізу предметної області та існуючих аналогів розроблено технічне завдання на створення мобільного онлайн-сервісу для автоматизації формування бібліографії наукових публікацій.

1.3.1. Призначення та сфера застосування

Мобільний онлайн-сервіс призначений для автоматизованого формування бібліографічних посилань відповідно до різних стилів цитування (ДСТУ 8302:2015, APA, IEEE) та їх конвертування між собою. Сервіс забезпечує зручний мобільний доступ до функціоналу.

Сфера застосування системи включає:

- формування бібліографічних описів наукових статей, книг та інших джерел;
- автоматизація роботи з бібліографічною інформацією в академічному середовищі;
- забезпечення коректного оформлення списків літератури у наукових публікаціях.

Цільовою аудиторією сервісу є:

- науковці та дослідники;
- студенти та аспіранти;
- викладачі закладів вищої освіти;
- редактори наукових видань та видавництва;
- бібліотекарі та інформаційні працівники.

1.3.2. Функціональні вимоги

Система має забезпечувати такі функціональні можливості:

1. Введення бібліографічних даних:

- можливість вказати авторів публікації з підтримкою різних форматів запису (Прізвище І.Б., Прізвище І.Б.);

- введення назви публікації;
- зазначення року видання;
- введення DOI (Digital Object Identifier);
- додавання інформації про сторінки;
- введення даних про видавництво та місце видання;
- зазначення тому та номеру видання для журнальних статей.

2. Форматування бібліографічних посилань:

- підтримка стилю ДСТУ 8302:2015 з дотриманням усіх вимог стандарту;

- підтримка міжнародного стилю APA (American Psychological Association);

- підтримка стилю IEEE (Institute of Electrical and Electronics Engineers);

- можливість одночасного форматування за кількома стилями;
- динамічне відображення результатів форматування.

3. Управління форматуванням:

- можливість вибору стилів форматування;
- динамічна зміна полів форми залежно від обраних стилів;
- маркування обов'язкових полів для кожного стилю.

4. Експорт та поширення:

- копіювання сформованих бібліографічних описів у буфер обміну;

- функція друку результатів;
- можливість поділитися сформованими посиланнями;
- збереження посилань у текстовому форматі.

1.3.3. Вимоги до інтерфейсу

Загальні вимоги:

- інтуїтивно зрозумілий інтерфейс, орієнтований на користувачів без спеціальних технічних знань;
- адаптивний дизайн, що забезпечує коректне відображення на різних пристроях (смартфони, планшети, комп'ютери);
- дотримання стандартів доступності для користувачів з обмеженими можливостями;
- підтримка світлої та темної теми інтерфейсу.

Структура інтерфейсу:

- розділ для введення бібліографічних даних у вигляді форми;
- блок вибору стилів форматування;
- секція відображення результатів форматування;
- панель дій для експорту та поширення результатів.

Елементи управління:

- чітке маркування обов'язкових полів для введення;
- підказки та опис полів для полегшення введення даних;
- кнопки дій з інтуїтивно зрозумілими іконками та підписами.

Розміщення елементів:

- логічне групування полів введення за категоріями;
- ієрархічна організація інтерфейсу з пріоритезацією важливих елементів;
- оптимальне використання доступного простору екрану на різних пристроях.

На рисунку 1.5 представлено прототип інтерфейсу розроблюваного мобільного онлайн-сервісу.

Введіть бібліографічні дані

☒ ДСТУ 8302:2015
 ☒ APA
 ☒ IEEE

Тип джерела:

Стаття

Книга

Конференція

Веб-сторінка

Дисертація

Автори *

Моторний Д.К.

Вкажіть прізвище та ініціали кожного автора, розділяючи їх крапкою з комою (;)

Т Назва *

Назва статті

Рисунок 1.5 - Прототип інтерфейсу мобільного онлайн-сервісу для формування бібліографії

1.3.4. Нефункціональні вимоги:

1. Надійність:
 - забезпечення доступності сервісу не менше 99,5% часу;
 - коректна обробка некоректного введення без аварійного завершення роботи;
 - збереження даних користувача при втраті з'єднання або перезавантаженні сторінки.
2. Масштабованість:
 - модульна архітектура, що дозволяє додавати нові стилі цитування без значних змін у коді;
 - можливість розширення функціональності без перебудови всієї системи.
3. Сумісність:

- підтримка усіх сучасних браузерів (Chrome, Firefox, Safari, Edge);
- коректна робота на різних операційних системах (Windows, iOS, Android);
- відповідність стандартам HTML5, CSS3 та ECMAScript.

4. Безпека:

- безпечна передача даних через HTTPS;
- відсутність збереження чутливих даних користувача на сервері.

1.3.5. Технічні вимоги:

1. Технології розробки:

- фронтенд: HTML5, CSS3, JavaScript;
- використання сучасних підходів до розробки адаптивних інтерфейсів;

- оптимізація для мобільних пристроїв.

2. Архітектура системи:

- клієнтська частина з мінімальними залежностями;
- модульна структура, що дозволяє незалежно розвивати компоненти;
- використання локального сховища браузера для збереження налаштувань користувача.

1.3.6. Етапи розробки

Розробка мобільного онлайн-сервісу для автоматизації формування бібліографії включає такі етапи:

1. Етап аналізу та проектування:

- аналіз вимог та уточнення технічного завдання;
- проектування архітектури системи;
- розробка прототипів інтерфейсу;

- визначення алгоритмів форматування для різних стилів цитування.

Б. Етап розробки:

- створення базової структури веб-додатку;
- реалізація алгоритмів форматування бібліографічних посилань;
- розробка адаптивного інтерфейсу користувача.

В. Етап тестування:

- функціональне тестування форматування посилань;
- тестування інтерфейсу на різних пристроях;
- тестування продуктивності та навантаження;
- перевірка коректності форматування для різних типів джерел.

Г. Етап впровадження:

- розгортання системи на хостингу;
- збір та аналіз зворотного зв'язку.

І. Етап підтримки:

- виправлення виявлених помилок;
- оновлення алгоритмів форматування при зміні стандартів;
- додавання нових стилів цитування;
- оптимізація продуктивності на основі реальних даних використання.

1.3.7 Критерії прийому роботи

Розробка вважається успішно завершеною за умови виконання таких критеріїв:

А. Функціональні критерії:

- система коректно форматує бібліографічні посилання відповідно до стандартів ДСТУ 8302:2015, APA та IEEE;
- інтерфейс адаптується до різних розмірів екрану та працює на всіх цільових пристроях;

- користувачі можуть зберігати, копіювати та поширювати сформовані бібліографічні описи.

Б. Нефункціональні критерії:

- час відгуку системи відповідає встановленим вимогам;
- досягнуто заплановану доступність сервісу;
- система коректно обробляє некоректне введення даних;
- забезпечено безпечну передачу даних через HTTPS.

В. Користувацькі критерії:

- позитивні відгуки від тестової групи користувачів;
- успішне виконання типових сценаріїв використання;
- висока оцінка зручності інтерфейсу користувачами.

Розробка мобільного онлайн-сервісу для автоматизації формування бібліографії наукових публікацій дозволить надати українським науковцям, студентам та редакторам зручний інструмент, що відповідає їхнім специфічним потребам у оформленні наукових робіт згідно з вітчизняними та міжнародними стандартами.

РОЗДІЛ 2. СПЕЦИФІКАЦІЯ ВИМОГ ДО ІНФОРМАЦІЙНОЇ СИСТЕМИ

2.1 Глосарій

Глосарій є словником основних використовуваних термінів, що забезпечує спільне розуміння основної термінології замовником і розробником. Він слугує відправною точкою для побудови більш розгорнутих моделей предметної області, які на стадії реалізації інформаційної системи лягають в основу об'єктної моделі та моделі даних. У таблиці 2.1 далі наведено глосарій згідно теми нашої роботи.

Таблиця 2.1 - Глосарій

Термін	Опис терміну
1	2
Основні поняття та категорії предметної області	
Бібліографічне посилання	Сукупність бібліографічних відомостей про цитований, розглядуваний або згадуваний документ, що необхідні й достатні для його загальної характеристики, ідентифікації та пошуку[3].
Бібліографічний опис	Сукупність бібліографічних відомостей про документ, наведених за певними правилами, що встановлені відповідним стандартом, і призначених для ідентифікації та загальної характеристики документа[2].
Стиль цитування	Набір правил, що регламентують оформлення списку бібліографічних посилань та наведення цитат у тексті праці[11].
ДСТУ 8302:2015	Національний стандарт України, що встановлює основні види бібліографічних посилань, правила та особливості їх складання й розміщування у документах[6].

Продовження таблиці 2.1

1	2
Стиль APA	Стиль цитування, розроблений Американською психологічною асоціацією, широко використовується у сфері соціальних та поведінкових наук[12].
Стиль IEEE	Стиль цитування, розроблений Інститутом інженерів з електротехніки та електроніки, переважно використовується у технічних публікаціях[20].
DOI	Digital Object Identifier - однозначний і постійний цифровий ідентифікатор об'єкта, що зберігає метадані і є міжнародним стандартом ISO 26324:2012[22].
Користувачі системи	
Науковець	Особа, яка займається науковою діяльністю та оформлює результати своїх досліджень у вигляді публікацій.
Студент	Особа, яка навчається в закладі вищої освіти та готує наукові роботи (курсові, дипломні) з використанням бібліографічних посилань.
Редактор	Фахівець, який перевіряє та виправляє наукові тексти перед публікацією, включаючи правильність оформлення бібліографії.
Вхідні та вихідні документи	
Інформація про публікацію	Дані про наукову публікацію (автори, назва, рік, видавництво тощо), які вводяться користувачем.
Бібліографічний опис	Вихідний документ, що містить відформатоване бібліографічне посилання згідно з обраним стилем цитування.
Шаблон цитування	Структурований формат для створення бібліографічного опису відповідно до конкретного стилю.

2.2 Концептуальна модель використання інформаційної системи

Специфікація функціональних вимог

Функціональні вимоги визначають перелік сервісів, які повинна виконувати система, включаючи реакції на вхідні дані та поведінку в певних ситуаціях.

Таблиця 2.2 - Специфікація функціональних вимог

Ідентифікатор вимоги	Назва вимоги (варіанта використання)	Атрибути вимог
ФВ-1	Введення бібліографічних даних	пріоритет: обов'язкове
ФВ-2	Форматування бібліографічних посилань за стилем ДСТУ 8302:2015	пріоритет: обов'язкове
ФВ-3	Форматування бібліографічних посилань за стилем APA	пріоритет: обов'язкове
ФВ-4	Форматування бібліографічних посилань за стилем IEEE	пріоритет: обов'язкове
ФВ-5	Експорт бібліографічних записів у різних форматах	пріоритет: рекомендоване
ФВ-6	Поділитися сформованими бібліографічними посиланнями	пріоритет: опційне

Специфікація нефункціональних вимог

Нефункціональні вимоги описують характеристики системи та її оточення, включаючи обмеження, що накладаються на дії і функції системи.

Таблиця 2.3 - Специфікація нефункціональних вимог

№	Назва вимоги	Характеристика
1	2	3
1	Застосовність	
1.1	Час для навчання користувачів	Користувач повинен освоїти основні функції системи за час не більше 15 хвилин без спеціального навчання
1.2	Вимірюваний час відгуку для типових завдань	Час формування бібліографічного опису не перевищує 2 секунди
2	Надійність	
2.1	Середній час безвідмовної роботи	Не менше 720 годин безперервної роботи
2.2	Середнє напрацювання до ремонту	Відновлення повної функціональності системи після збою має відбуватися не більше ніж за 2 години

Продовження таблиці 2.3

1	2	3
3	Робочі характеристики	
3.1	Швидкодія для запиту	Час відповіді сервера на запит користувача не більше 1 секунди
4	Проектні обмеження	
4.1	Мова програмування	JavaScript
5	Вимоги до документації, призначеної для користувача, і до системи допомоги	
5.1	Підказки в інтерфейсі	Контекстні підказки при введенні даних та виборі опцій
6	Інтерфейси	
6.1	Інтерфейси користувача	Адаптивний дизайн для мобільних, планшетних та десктопних пристроїв, підтримка світлої та темної теми
6.2	Комунікаційні інтерфейси	HTTPS для безпечної передачі даних
7	Вимоги до ліцензування	
7.1	Тип ліцензії	Open-source з вільним використанням для некомерційних цілей
7.2	Обмеження використання	Заборона використання для комерційних цілей без дозволу розробника
8	Вживані стандарти	
8.1	Стандарти цитування	Відповідність ДСТУ 8302:2015, APA, IEEE
8.2	Веб-стандарти	HTML5, CSS3, ECMAScript

2.3 Розробка функціональної моделі

Діаграма варіантів використання є важливим елементом концептуальної моделі інформаційної системи. Вона ілюструє взаємодію між різними користувачами (акторами) та системою, визначаючи основні функції, які система повинна виконувати.

Діаграма відображає всі етапи процесу, починаючи від введення бібліографічних даних і закінчуючи взаємодією з сформованими посиланнями.

Основними користувачами системи є користувачі та розробник.

Користувач - це особа, яка безпосередньо взаємодіє з веб-системою для виконання завдань, пов'язаних з бібліографічними посиланнями.

Він вводить дані про публікації, обирає стиль цитування та отримує сформовані бібліографічні посилання.

Розробник - це особа, відповідальна за створення, налаштування та підтримку веб-системи. Він проектує та реалізує функціональні вимоги, розробляє інтерфейси, тестує систему та забезпечує її підтримку та оновлення.

На діаграмі варіантів використання (рис. 2.1) чітко позначено всі можливі варіанти взаємодії між користувачами та системою.

Крім того, діаграма варіантів використання служить важливим інструментом для розробників, допомагаючи в точному визначенні вимог до функціональності та визначенні етапів розробки програмного забезпечення. Вона дозволяє ідентифікувати основні сценарії використання системи, що сприяє більш ефективному проектуванню та тестуванню.

Таким чином, діаграма варіантів використання є важливим етапом в процесі проектування інформаційної системи, оскільки вона допомагає зрозуміти, як система буде взаємодіяти з користувачами та які функціональні можливості слід реалізувати для задоволення потреб всіх типів користувачів.

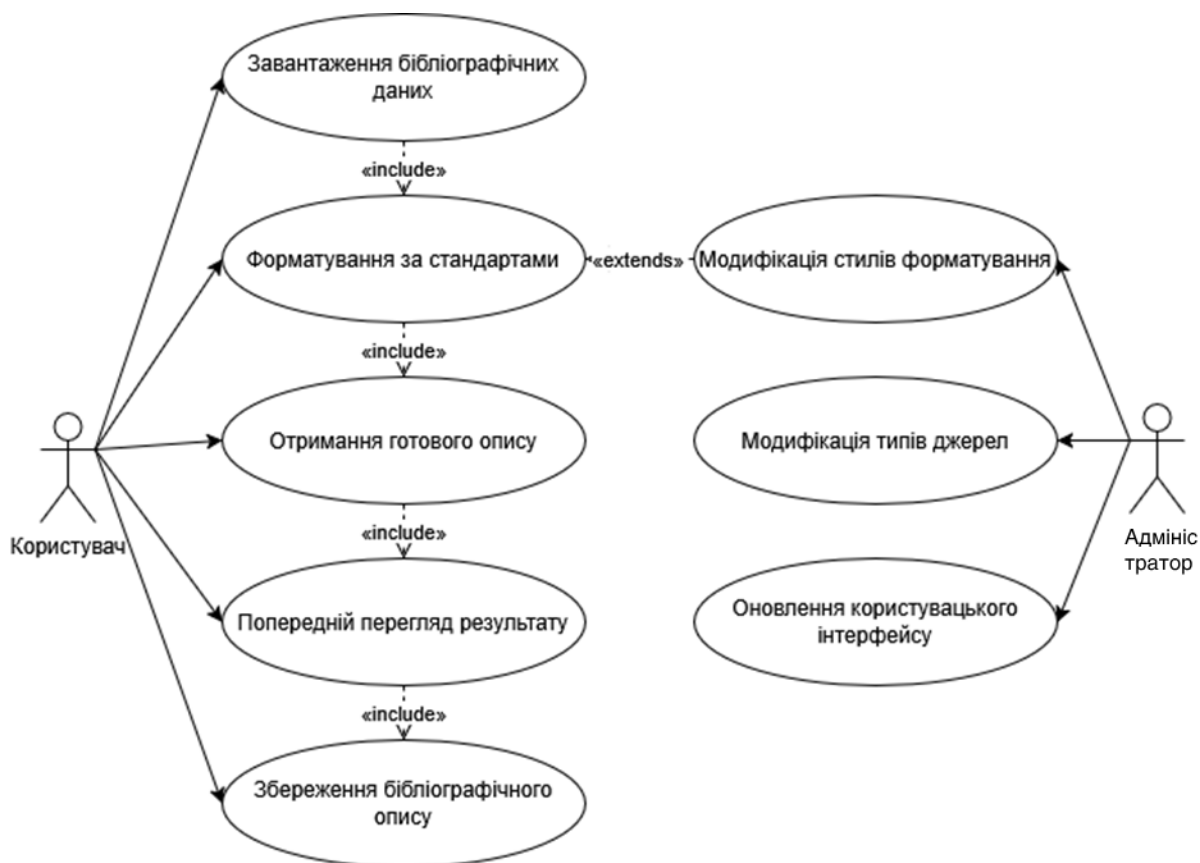


Рисунок 2.1 - Діаграма варіантів використання системи

Специфікація варіантів використання

Нижче (табл. 2.4) наведено детальний опис основних варіантів використання, що реалізують ключову функціональність системи.

Таблиця 2.4 - Варіант використання «Форматування бібліографічних посилань за стилем ДСТУ 8302:2015»

Варіант використання	Форматування бібліографічних посилань за стилем ДСТУ 8302:2015
1	2
Контекст використання	Користувач потребує оформити бібліографічне посилання на наукове джерело відповідно до вимог українського стандарту ДСТУ 8302:2015
Дійові особи	Основна дійова особа: Користувач Учасники та інтереси: Система, яка забезпечує правильне форматування

Продовження таблиці 2.4

1	2
Передумова	Користувач має доступ до системи. Користувач ввів необхідні бібліографічні дані джерела
Тригер	Користувач обирає стиль ДСТУ 8302:2015 та натискає кнопку «Сформувати»
Сценарій	1. Користувач вводить бібліографічні дані джерела (автори, назва, місце видання, рік тощо) 2. Користувач обирає стиль ДСТУ 8302:2015 3. Система перевіряє наявність всіх необхідних для обраного стилю полів 4. Система форматує введені дані відповідно до вимог стандарту ДСТУ 8302:2015 5. Система відображає користувачу сформоване бібліографічне посилання 6. Користувач може скопіювати сформоване посилання, експортувати його
Постумова	Користувач отримує коректно сформоване бібліографічне посилання у форматі ДСТУ 8302:2015

Розкадровка варіантів використання представляє логічний і концептуальний опис функціональних можливостей системи для певних сценаріїв, включаючи взаємодію між системою та її користувачами.

2.3 Розробка функціональної моделі

Функціональна модель розробки програмного забезпечення є структурованим зображенням функцій, що виконуються в ході його проектування, а також інформації, що зв'язує між собою ці функції. Мета функціональної моделі: визначити функції системи, показати перелік і взаємозв'язок завдань, склад вхідних і вихідних документів.

Для визначення функціональних вимог до нашого онлайн-сервісу розроблено функціональну модель з використанням стандарту IDEF0.

IDEF0 - це модель, створена для допомоги в аналізі та вдосконаленні процесів. Суть моделі IDEF0 полягає в тому, що вона дозволяє спочатку

ігнорувати те, що відбувається всередині процесу, і зосередитися на тому, як процес взаємодіє з навколишнім світом [15]. Важливою характеристикою IDEF0 є можливість розрізняти чотири типи інтерфейсів:

- входи від людей або інших процесів;
- виходи до людей або інших процесів;
- взаємодії з джерелами, які контролюють процес;
- взаємодії з механізмами, що забезпечують процес.

Це дозволяє чітко зображати взаємозв'язки і ролі різних елементів, таких як стандарти цитування і правила форматування, що необхідні для виконання головної функції системи - «Формування бібліографічних посилань».

Контекстна діаграма.

Контекстна діаграма представляє систему як єдине ціле та показує зв'язок із зовнішнім середовищем (рис. 2.2).



Рисунок 2.2 - Контекстна діаграма інформаційної системи в нотації IDEF0

Основна функція системи «Формування бібліографічних посилань». На вхід подаються бібліографічні дані джерел та запити користувачів.

Стрілки управління - це стандарти цитування (ДСТУ 8302:2015, APA, IEEE) та правила форматування бібліографічних записів. На виході отримуємо сформовані бібліографічні посилання та експортовані бібліографічні записи. У ролі механізмів виступають: користувачі системи (науковці, студенти, редактори) та веб-платформа.

Декомпозиція контекстної діаграми

Декомпозиція контекстної діаграми дозволяє розглянути функції системи та їх взаємодію з середовищем.

Вона полягає в розбитті головної функції на більш дрібніші підфункції, що дає можливість глибше зрозуміти структуру системи і її складові.

У нашій системі основною функцією є «Формування бібліографічних посилань». Декомпозиція цієї функції дозволяє чітко визначити всі етапи її виконання та необхідні ресурси для кожного етапу.

Це включає обробку даних, вибір стилю, перевірку, форматування та видачу готового посилання.

Крім того, декомпозиція дозволяє визначити, які механізми та інтерфейси беруть участь у виконанні кожної підфункції. В цьому випадку це можуть бути користувачі системи (науковці, студенти, редактори) та веб-платформа, які взаємодіють з системою. Важливою частиною є також управлінські інтерфейси, наприклад, стандарти цитування та правила форматування, які контролюють процес генерації посилань.

Завдяки декомпозиції можна більш точно визначити, де можуть виникати потенційні проблеми чи неефективності, а також де необхідно внести покращення. Також цей етап дозволяє краще організувати процеси в системі, зробити їх зрозумілими для всіх учасників, що в свою чергу спрощує розробку, тестування та підтримку системи.

Вона сприяє чіткому розмежуванню підсистем і полегшує реалізацію інформаційної системи.

РОЗДІЛ 3. ОПИС ПРИЙНЯТИХ ПРОЄКТНИХ ТА ТЕХНОЛОГІЧНИХ РІШЕНЬ

3.1 Розробка об'єктної моделі

Об'єктна модель є ключовою складовою проєктування програмного забезпечення, оскільки дозволяє структурувати складні взаємозв'язки між елементами системи[9].

Для мобільного онлайн-сервісу автоматизації формування бібліографії наукових публікацій розроблено об'єктну модель з використанням діаграми класів у нотації UML.

3.1.1 Діаграма класів системи

Діаграма класів відображає структуру системи, демонструючи класи системи, їх атрибути, методи та взаємозв'язки між ними[7]. На рисунку 3.1 представлено діаграму класів розроблюваного онлайн-сервісу.

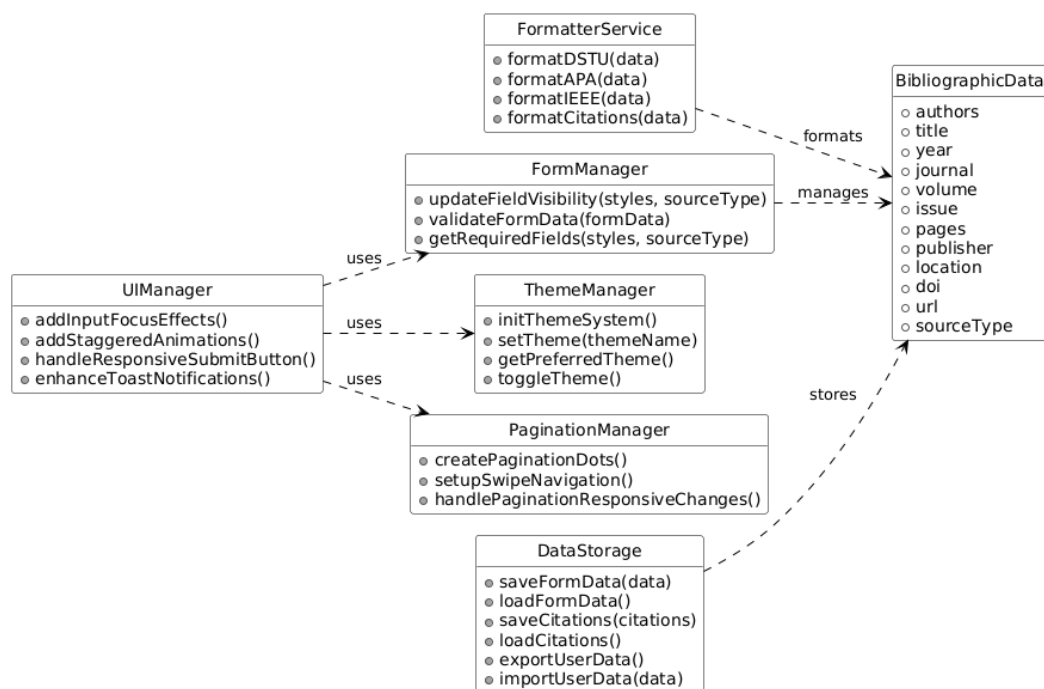


Рисунок 3.1 - Діаграма класів мобільного онлайн-сервісу

Хоча розроблюваний онлайн-сервіс використовує для зберігання даних локальне сховище браузера (localStorage), для забезпечення можливості майбутнього розширення функціональності розроблено ER-діаграму, що відображає структуру даних системи (рис. 3.2).

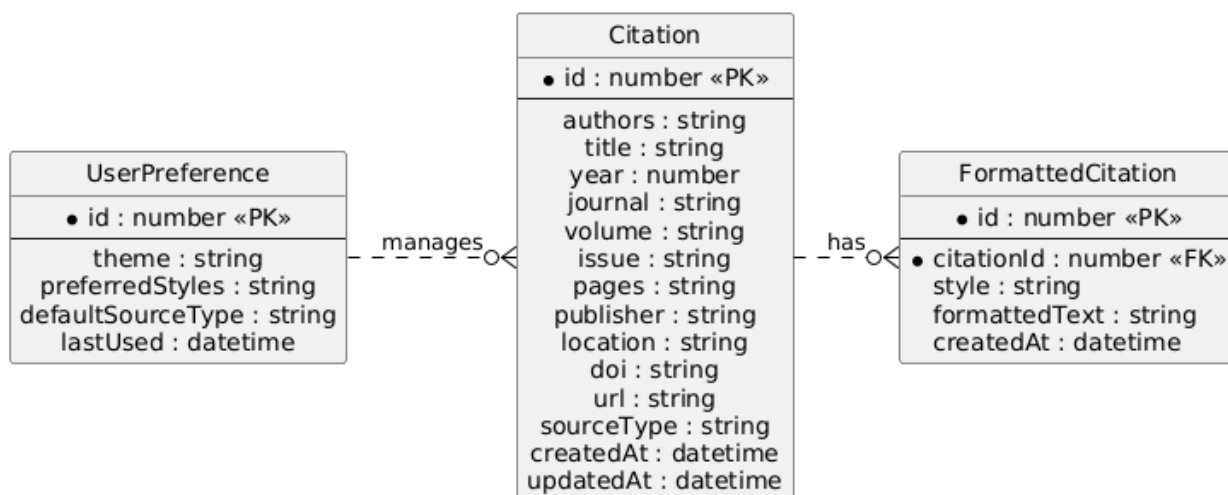


Рисунок 3.2 - ER-діаграма мобільного онлайн-сервісу

Розроблена об'єктна модель та ER-діаграма забезпечують структурну основу для реалізації функціональних вимог системи, визначених у технічному завданні, і дозволяють ефективно організувати дані та логіку роботи мобільного онлайн-сервісу для формування бібліографії.

3.2 Розробка архітектури

Архітектура мобільного онлайн-сервісу для автоматизації формування бібліографії наукових публікацій розроблена з урахуванням вимог щодо мобільного доступу, простоти використання та автономності роботи. Для забезпечення цих вимог обрано архітектуру Single-Page Application (SPA) з мінімальною залежністю від серверної частини.

3.2.1 Загальна архітектура системи

На рисунку 3.3 представлено діаграму компонентів розроблюваного онлайн-сервісу.

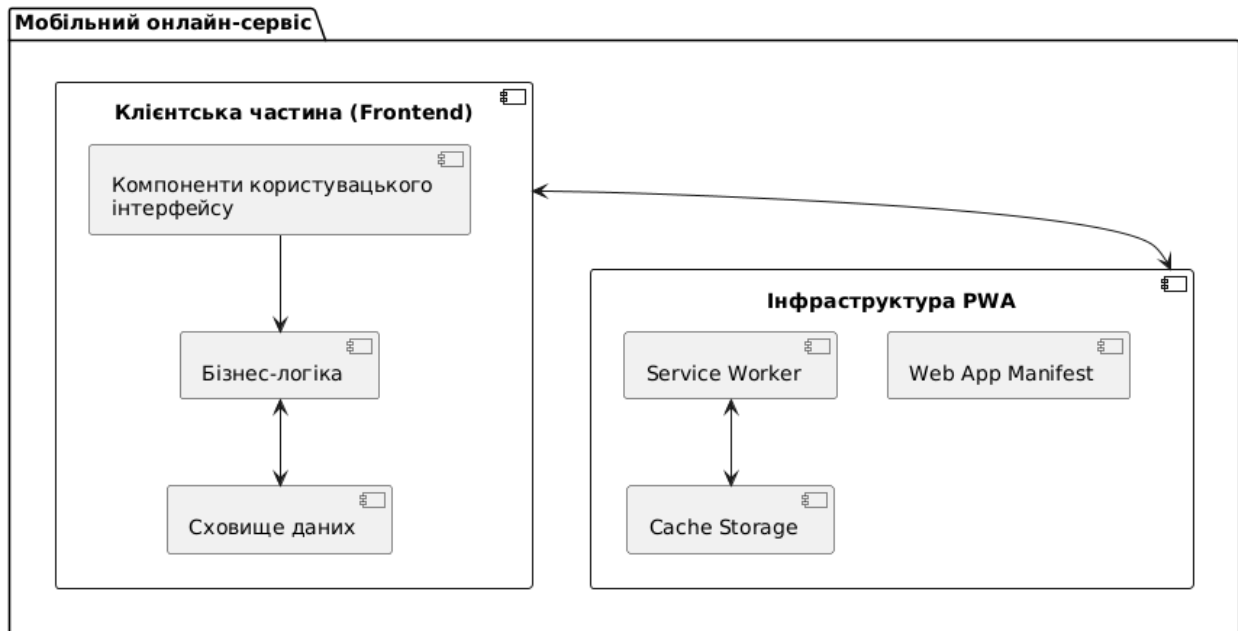


Рисунок 3.3 - Діаграма компонентів мобільного онлайн-сервісу

Архітектура системи базується на принципах прогресивного веб-додатку (Progressive Web Application, PWA), що забезпечує можливість використання сервісу в офлайн-режимі та додавання ярлика на домашній екран мобільного пристрою[23].

PWA було обрано, оскільки цей сучасний тип веб-додатків поєднує переваги нативних мобільних застосунків та можливості веб-технологій. Вони забезпечують швидку, надійну та зручну роботу незалежно від якості інтернет-з'єднання, використовуючи новітні веб-функції. Крім того, під час розробки прогресивних веб-застосунків можна впроваджувати такі можливості, як push-повідомлення, додавання ярлика на головний екран пристрою та суттєве прискорення завантаження сторінок. Такий підхід дозволяє поетапно трансформувати звичайний вебсайт у сучасний застосунок з високою функціональністю[13].

Система складається з двох основних пакетів компонентів:

Клієнтська частина (Frontend):

- компонент користувацького інтерфейсу (UI Component) взаємодіє з користувачем через UI Interface;
- компонент бізнес-логіки (Business Logic) відповідає за обробку даних та форматування бібліографічних посилань;
- компонент сховища даних (Data Storage) забезпечує збереження даних користувача через Storage API.

Інфраструктура PWA:

- компонент Service Worker відповідає за кешування та роботу в офлайн-режимі;
- компонент Web App Manifest забезпечує метадані для встановлення додатку на пристрій;
- компонент Cache Storage взаємодіє з Service Worker через Cache API для зберігання необхідних ресурсів.

Взаємодія між компонентами реалізована через чітко визначені інтерфейси, що забезпечує модульність та розширюваність системи.

3.3 Проектування інтерфейсу програмної системи

Інтерфейс є ключовим елементом мобільного онлайн-сервісу для автоматизації формування бібліографії. Розроблений інтерфейс забезпечує адаптивність для різних пристроїв та інтуїтивність використання.

Проектування інтерфейсу здійснювалося з урахуванням специфіки мобільних пристроїв як основної платформи використання.

Необхідність у адаптивності зростає, оскільки кількість людей, що використовують пристрої з меншими екранами для перегляду Інтернету, зростає зі все збільшуваною швидкістю [18]

3.3.1 Основні компоненти інтерфейсу

Інтерфейс мобільного онлайн-сервісу складається з наступних основних компонентів:

1. Шапка (Header):
 - логотип сервісу;
 - перемикач теми (світла/темна).
2. Секція вибору параметрів:
 - Перемикачі стилів цитування (ДСТУ 8302:2015, APA, IEEE);
 - Перемикачі типу джерела (стаття, книга, веб-сторінка тощо).
3. Форма введення даних:
 - поля для введення бібліографічних даних;
 - підказки та опис полів;
 - індикатори обов'язкових полів;
 - кнопка відправки форми.
4. Секція результатів:
 - табові перемикачі між стилями на мобільних пристроях;
 - блоки з відформатованими бібліографічними посиланнями;
 - кнопки для копіювання, друку та поширення результатів.

Взаємодія користувача з інтерфейсом відбувається за наступним сценарієм:

1. Вибір стилів цитування та типу джерела;
2. Заповнення форми з бібліографічними даними;
3. Відправка форми та отримання результатів;
4. Взаємодія з результатами (копіювання, друк, поширення).

Розташування елементів інтерфейсу спроектовано з урахуванням логічної послідовності дій користувача, забезпечуючи інтуїтивний шлях від вибору параметрів до отримання результатів.

3.3.2 Адаптивність інтерфейсу

Для забезпечення комфортного використання сервісу на різних пристроях реалізовано адаптивний дизайн, що автоматично підлаштовується під розмір екрану (рис. 3.4).

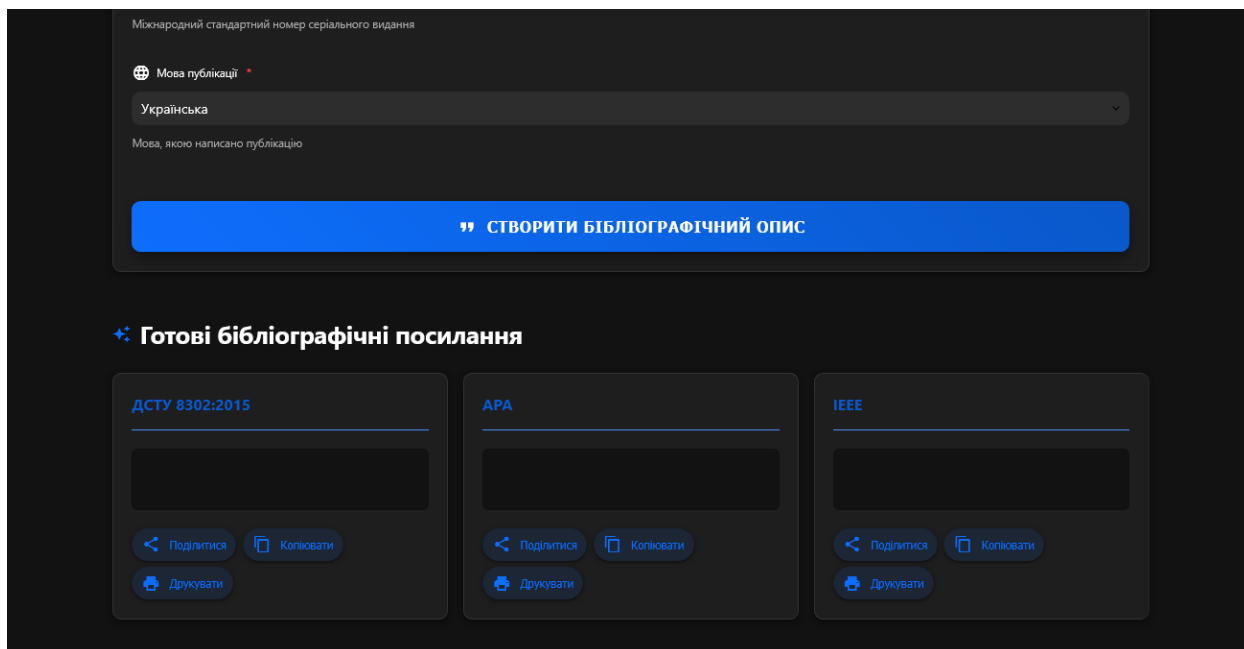


Рисунок 3.4 - Адаптивність інтерфейсу для десктопних пристроїв

Приклад світлої теми на десктопному пристрої наведено у Додатку А.

Адаптивність інтерфейсу реалізовано за допомогою наступних технологічних рішень:

- CSS медіа-запити (@media queries) для визначення типу пристрою;
- гнучкі контейнери на основі CSS Flexbox та Grid;
- відносні одиниці виміру (em, rem, vw, vh) замість абсолютних (px);
- перебудова компонентів залежно від розміру екрану.

На мобільних пристроях реалізовано специфічні адаптації інтерфейсу:

- вертикальне розташування секцій для комфортного скролінгу;

- збільшені елементи управління для зручності взаємодії на сенсорних екранах;
- табова навігація між результатами для різних стилів цитування;
- оптимізація полів форми для мобільного введення.

3.3.2 Темна та світла теми

Для забезпечення комфортного використання сервісу в різних умовах освітлення реалізовано підтримку світлої та темної тем оформлення (рис. 3.5).

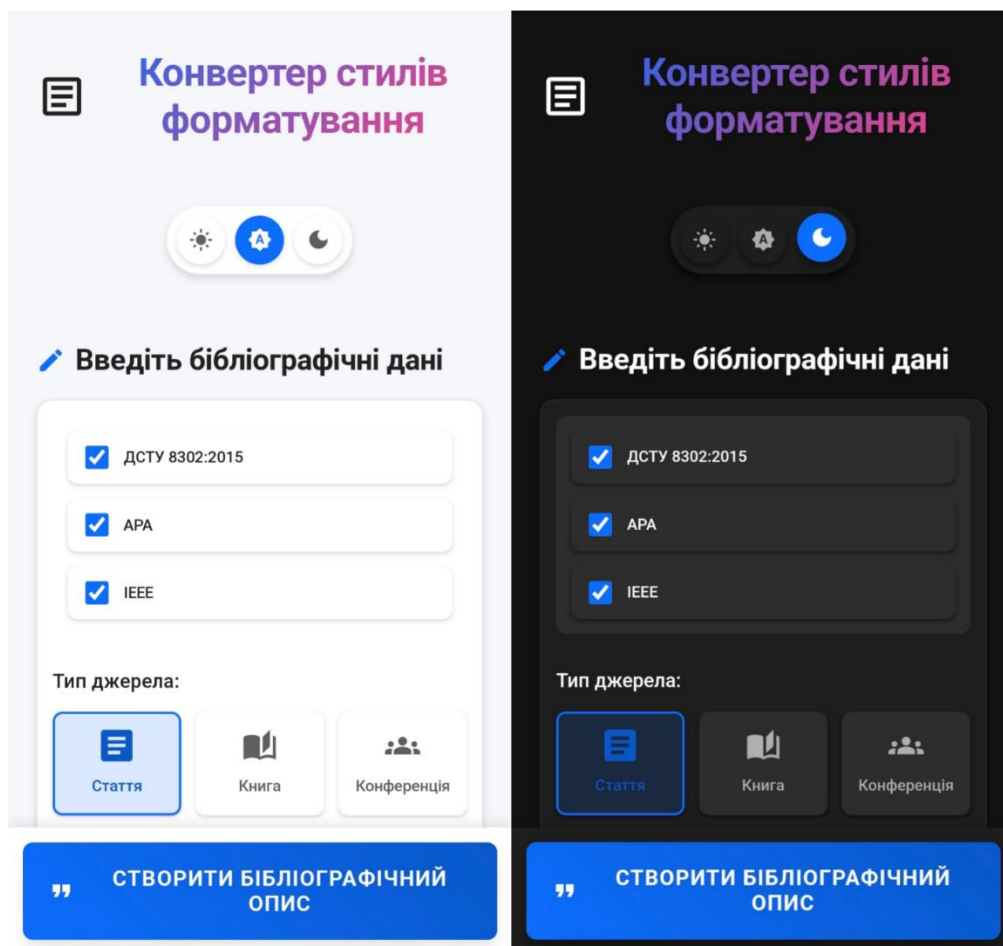


Рисунок 3.5 - Світла та темна теми інтерфейсу

Система автоматично визначає бажану тему на основі системних налаштувань користувача, але також надає можливість ручного

перемикання. Вибір користувача зберігається в localStorage і відновлюється при повторному відвідуванні сервісу.

Технічна реалізація підтримки тем базується на використанні CSS змінних (Custom Properties) та атрибуту data-theme.

Фрагмент лістингу коду:

```
1. :root {  
2.   -background-color: #ffffff;  
3.   -text-color: #333333;  
4.   -accent-color: #4A90E2;  
5.   /* інші змінні для світлої теми */  
6. }  
7.  
8. [data-theme="dark"] {  
9.   -background-color: #1E1E1E;  
10.  -text-color: #E0E0E0;  
11.  -accent-color: #61AFEF;  
12.  /* змінні для темної теми */  
13. }  
14.
```

Custom properties поведуться як свого роду scoped змінна, тому що значення успадковуються дочірніми елементами. Цей механізм дозволяє перевизначати значення змінних для елементів, що знаходяться в межах області дії селектора [data-theme="dark"], перекриваючи значення з :root[19].

Програмно перемикання між темами реалізовано у модулі themeManager.js, який:

1. Перевіряє системні налаштування за допомогою `window.matchMedia('(prefers-color-scheme: dark)')`;
2. Зберігає налаштування в localStorage;
3. Встановлює атрибут `data-theme` для кореневого елемента документа.

3.3.3 Принципи проєктування інтерфейсу

При розробці інтерфейсу дотримано наступних принципів:

1. Mobile First - проєктування інтерфейсу починалося з мобільної версії;

- розташування елементів оптимізовано для вертикального перегляду;

- первинний фокус на основних функціях, необхідних мобільному користувачу;

- поступове додавання розширених можливостей для більших екранів.

2. Progressive Disclosure - поступове розкриття необхідної інформації

- базові поля форми видимі відразу;

- додаткові поля з'являються при виборі відповідного типу джерела;

- підказки відображаються при фокусі на полях введення;

- обов'язкові поля позначені особливим чином.

3. Consistency - єдиний стиль та логіка взаємодії

- послідовне використання кольорів, шрифтів та іконок;

- єдиний стиль кнопок, полів форми та інших елементів інтерфейсу;

- узгоджене розташування повторюваних елементів на різних екранах;

- єдина система повідомлень про помилки та підказок.

4. Feedback - чіткий зворотний зв'язок про результати дій

- візуальне підтвердження успішного завершення операцій;

- явні повідомлення про помилки з детальними поясненнями;

- індикація прогресу при тривалих операціях;

- інтерактивна валідація форми в процесі введення даних.

Вибір інтерфейсних рішень базувався на сучасних практиках веб-розробки та принципах UX/UI дизайну, що дозволило створити інтуїтивно зрозумілий, адаптивний та візуально привабливий інтерфейс для будь-якого пристрою.

Особлива увага приділялась забезпеченню зручності використання на мобільних пристроях, що є критичним для цільової аудиторії сервісу.

3.4 Обґрунтування вибору мови програмування

Вибір мови програмування та супутніх технологій є ключовим рішенням, що впливає на всі аспекти розробки програмного продукту. Для мобільного онлайн-сервісу автоматизації формування бібліографії наукових публікацій вибір технологічного стеку базувався на аналізі функціональних вимог та особливостей використання системи.

3.4.1 Критерії вибору мови програмування

При виборі мови програмування були визначені наступні ключові критерії:

1. Підтримка веб-платформи - ефективне виконання у веб-браузерах;
2. Підтримка мобільних пристроїв - оптимальна робота на різних платформах;
3. Можливість розробки PWA - підтримка офлайн-режиму та інших функцій;
4. Продуктивність - швидка обробка запитів і взаємодія з користувачем;
5. Підтримка спільноти - наявність активної спільноти та готових рішень;
6. Простота розробки - зрозумілий процес розробки.

3.4.2 Порівняльний аналіз мов програмування

Порівняння технологій за ключовими критеріями представлено в таблиці 3.1.

Таблиця 3.1 - Порівняння мов програмування за визначеними критеріями

Критерій / Технологія	JavaScript	TypeScript	Java	Python	Dart
Підтримка веб-платформи	Висока	Висока	Середня	Середня	Середня
Підтримка мобільних пристроїв	Висока	Висока	Середня	Середня	Висока
Можливість розробки PWA	Висока	Висока	Низька	Середня	Середня
Продуктивність	Висока	Висока	Висока	Середня	Висока
Підтримка спільноти	Висока	Висока	Висока	Висока	Середня
Простота розробки	Висока	Середня	Низька	Висока	Середня

На основі проведеного аналізу для розробки мобільного онлайн-сервісу обрано JavaScript як основну мову програмування.

3.4.3 Обґрунтування вибору JavaScript

Вибір JavaScript обумовлений наступними факторами:

1. Нативна підтримка веб-браузерами - це мова, нативно підтримувана всіма браузерами без додаткових плагінів, що критично для доступності сервісу;
2. Ефективна робота на мобільних пристроях - сучасні мобільні браузери мають оптимізовані JavaScript-рушії, забезпечуючи високу продуктивність навіть на пристроях з обмеженими ресурсами;
3. Підтримка PWA - JavaScript надає всі необхідні API для реалізації прогресивних веб-додатків, включаючи Service Workers, Web App Manifest та Push Notifications;
4. Асинхронна модель виконання - вбудована підтримка асинхронного програмування (Promise, async/await) забезпечує ефективну роботу з мережевими запитами без блокування інтерфейсу;

5. Модульна архітектура - підтримка модульної організації коду через ES Modules дозволяє створювати масштабовані додатки;

6. Автономність клієнтської частини - можливість реалізації логіки форматування бібліографічних посилань на клієнтській стороні забезпечує швидку реакцію та автономність роботи.

Таким чином, згідно з джерелом, JavaScript є універсальним та повсюдним для вебу, включений у браузері на всіх сучасних пристроях і, в поєднанні з API, що надаються браузером, дозволяє створювати складні вебдодатки з можливостями, близькими до нативних додатків, включаючи збереження даних та роботу офлайн. Мова сама по собі є *robust and efficient*, підтримується активною спільнотою, яка розробила численні бібліотеки та фреймворки, що спрощують процес розробки та вирішують питання сумісності між браузерами[17]

3.4.4 Обрані технології та інструменти

Для розробки мобільного онлайн-сервісу обрано наступний технологічний стек:

Фреймворки та бібліотеки:

- Vanilla JavaScript - для основної логіки додатку без залежності від важких фреймворків;
- ES Modules - для модульної організації коду (main.js, formatters.js, fieldManager.js, uiManager.js, тощо);
- Service Workers - для забезпечення офлайн-режиму та стратегій кешування.

Інфраструктура PWA:

- Web App Manifest - для налаштування встановлення додатку на пристрої користувача;
- Cache API - для кешування статичних ресурсів та офлайн-доступу;

- localStorage - для зберігання користувацьких налаштувань та даних форми;
- IndexedDB - для зберігання бібліографічних записів.

UI та UX інструменти:

- CSS Custom Properties - для гнучкої стилізації та темної/світлої тем;
- CSS Grid та Flexbox - для адаптивного дизайну без зовнішніх CSS фреймворків;
- динамічна валідація форм - для покращення зручності користування.

Підтримка мобільних пристроїв:

- адаптивний дизайн - для оптимізації під різні розміри екранів;
- жести свайпінгу (swipe) - для навігації між секціями на мобільних пристроях;
- тактильний зворотній зв'язок - для покращення UX на сенсорних пристроях.

3.4.5 Приклад використання обраних технологій

Нижче представлена імплементація модуля форматування бібліографічних посилань за стилем ДСТУ 8302:2015. Фрагмент лістингу коду:

```
1. /
2.  * Модуль форматування за стилем ДСТУ 8302:2015
3.  */
4.
5. // Експорт функції форматування для використання в інших
   модулях
6. export function formatDSTU(data) {
7.     try {
8.         const elements = [];
9.         const sourceType = data.sourceType || 'article';
10.
11.         // Форматування авторів
12.         elements.push(formatAuthors(data.authors));
13.
14.         // Додавання назви публікації
```

```

15.         elements.push(data.title);
16.
17.         // Додавання специфічних для типу джерела
елементів
18.         switch (sourceType) {
19.             case 'article':
20.                 if (data.journal)
elements.push(`_${data.journal}_`);
21.                 if (data.volume && data.issue) {
22.
elements.push(`${data.volume}(${data.issue})`);
23.                 } else if (data.volume) {
24.                     elements.push(`${data.volume}`);
25.                 } else if (data.issue) {
26.                     elements.push(`№ ${data.issue}`);
27.                 }
28.                 break;
29.
30.             case 'book':
31.                 if (data.volume) elements.push(`T.
${data.volume}`);
32.                 break;
33.             }
34.
35.         // Додавання загальних елементів
36.         if (data.location || data.publisher) {
37.             elements.push([data.location, data.publisher]
38.                 .filter(Boolean)
39.                 .join(': '));
40.         }
41.
42.         elements.push(data.year);
43.         if (data.pages) elements.push(`C. ${data.pages}`);
44.         if (data.doi) elements.push(`DOI: ${data.doi}`);
45.
46.         // Об'єднання всіх елементів з роздільниками
47.         return elements.filter(Boolean).join('. ');
48.     } catch (error) {
49.         // Запасний варіант форматування у випадку помилки
50.         return `${data.authors}. ${data.title}.
${data.year}`;
51.     }
52. }
53.
54. // Допоміжна функція для форматування списку авторів
55. function formatAuthors(authors) {
56.     if (!authors) return 'Автор не вказаний';
57.
58.     const authorsList = authors.split(';').map(author =>
author.trim());
59.

```

```

60.     if (authorsList.length > 3) {
61.         return `${authorsList[0]} та ін.`;
62.     }
63.
64.     return authorsList.join(', ');
65. }

```

3.4.6 Особливості реалізації

Успішна розробка сервісу супроводжувалася рядом технічних рішень, що забезпечили якість програми:

1. Модульна організація коду - розподіл коду на функціональні модулі (formatters.js, fieldManager.js, formHandler.js, uiManager.js, themeManager.js, dataStorage.js);
2. Особливості алгоритмів форматування - реалізовано три стилі (ДСТУ, APA, IEEE) з обробкою різних типів джерел та механізмом запасного форматування;
3. Офлайн-функціональність - використання Service Worker для кешування ресурсів та localStorage для збереження даних користувача;
4. Адаптивний дизайн - забезпечення коректної роботи на різних пристроях з підтримкою сенсорного введення та різних орієнтацій екрану.

Реалізовані деталі імплементації забезпечують високу якість програмного продукту, його надійність, швидкодію та зручність використання на різних пристроях, як онлайн, так і в автономному режимі.

3.5 Опис програмної реалізації

Програмна реалізація сервісу базується на сучасних веб-технологіях з урахуванням вимог до продуктивності, масштабованості та кросплатформенності. Усі функціональні компоненти реалізовані як окремі модулі, що дозволяє забезпечити гнучкість розробки, простоту супроводу та можливість подальшого розширення системи.

Розробка сервісу базується на сучасних підходах до веб-програмування, що забезпечують високу продуктивність та зручність для користувачів. Архітектура коду модульна, що сприяє легкому супроводу та подальшому розширенню функціональності. Особлива увага приділяється адаптивності інтерфейсу, щоб додаток коректно працював на різних пристроях — від десктопів до мобільних телефонів. Важливою складовою є підтримка офлайн-режиму, що гарантує стабільність роботи навіть за відсутності мережевого з'єднання, що особливо актуально для користувачів із нестабільним інтернетом. Крім того, реалізація прогресивного веб-додатку дозволяє користувачам встановлювати сервіс на свої пристрої, забезпечуючи нативний досвід використання. Такий підхід відповідає сучасним вимогам до веб-додатків і сприяє підвищенню загальної якості продукту.

3.5.1 Загальна структура програмного забезпечення

Структура проєкту організована для чіткого поділу відповідальності між компонентами.

Ключові файли та директорії:

- index.html - головна HTML-сторінка додатку;
- manifest.json - маніфест для реалізації функцій PWA;
- service-worker.js - сервіс-воркер для забезпечення офлайн-режиму;
- Директорії styles/, js/ та assets/ - містять CSS, JavaScript модулі та статичні ресурси.

3.5.2 Модульна структура JavaScript-коду

Код організований у модулі, де кожен відповідає за певний аспект функціональності:

- main.js - точка входу в додаток;
- formatters.js - алгоритми форматування бібліографічних посилань;

- fieldManager.js - керування полями форми;
- formHandler.js - обробка введення даних;
- uiManager.js - інтерфейс користувача;
- paginationManager.js - навігація між результатами;
- themeManager.js - керування темами оформлення;
- dataStorage.js - збереження та відновлення даних;
- utils.js - допоміжні функції.

3.5.3 Реалізація ключових функціональних модулів

1. Модуль форматування (formatters.js):

- реалізує алгоритми форматування за стандартами ДСТУ 8302:2015, APA та IEEE;

- використовує об'єктно-орієнтований підхід з приватними методами для форматування окремих елементів та публічними методами для отримання повних посилань.

Детальний код реалізації алгоритмів форматування бібліографічних посилань наведено у Додатку Б. У цьому модулі реалізовано ключову бізнес-логіку сервісу, що відповідає за трансформацію введених користувачем даних у коректно сформатовані бібліографічні посилання згідно з вимогами відповідних стандартів.

2. Модуль керування полями (fieldManager.js):

- динамічно регулює видимість та обов'язковість полів форми залежно від обраного типу джерела та стилю цитування через матрицю вимог.

3. Модуль обробки форми (formHandler.js):

- відповідає за збір даних, валідацію та формування результатів, забезпечуючи зв'язок між UI та логікою форматування.

4. Модуль пагінації (paginationManager.js):

- забезпечує зручну навігацію між результатами форматування на мобільних пристроях, включаючи свайп-навігацію та індикатори.

3.5.4 Реалізація PWA-функціональності

1. Сервіс-воркер забезпечує офлайн-функціональність через стратегії кешування;
2. Web App Manifest містить метадані для встановлення додатку на пристрій користувача, що дозволяє запускати його у повноекранному режимі.

3.5.5 Алгоритми форматування бібліографічних посилань

1. ДСТУ 8302:2015:

Враховує особливості українського стандарту: форматування авторів, обробка елементів залежно від типу джерела, додавання видавничих даних та відповідних розділових знаків.

2. АРА:

Реалізує вимоги 7-го видання: форматування авторів у стилі "Прізвище, І. І.", додавання року в дужках, використання Sentence case для назв та курсиву для видань.

3. IEEE:

Форматує авторів як "І. І. Прізвище", розміщує назву публікації в лапках, використовує префікси "vol.", "no.", "pp." та форматує URL з "[Online]. Available: ".

3.5.6 Адаптивність інтерфейсу

Адаптивність реалізована через:

- CSS Grid та Flexbox;
- media queries;
- відносні одиниці вимірювання.

Для мобільних пристроїв реалізовано:

- табову навігацію між результатами;
- свайп-жести;
- збільшені елементи керування;
- вертикальне розташування полів форми.

3.5.7 Темний режим

Темний режим реалізовано через CSS-змінні.

Фрагмент лістингу коду:

```
1. :root {  
2.   -background-color: #ffffff;  
3.   -text-color: #333333;  
4.   /* інші змінні */  
5. }  
6.  
7. [data-theme="dark"] {  
8.   -background-color: #1e1e1e;  
9.   -text-color: #e0e0e0;  
10.  /* перевизначення для темної теми */  
11. }
```

Система визначає системні налаштування через `media query` та зберігає вибір користувача в `localStorage`.

3.5.8 Управління даними

Дані форми та історія форматування зберігаються в `localStorage` для відновлення при повторному відвідуванні. Реалізовано експорт та імпорт даних у форматі JSON.

3.5.9 Оптимізація та безпека

Для оптимізації та безпеки реалізовані такі функції:

- швидке завантаження модулів;
- мінімізація перемальовувань DOM;
- відкладене виконання некритичних операцій;
- валідація введених даних;
- обробка винятків при роботі з localStorage.

РОЗДІЛ 4. ДОСЛІДНА ЕКСПЛУАТАЦІЯ ІНФОРМАЦІЙНОЇ СИСТЕМИ

4.1 Інструкція користувача

У даному підрозділі описується функціональність розробленого програмного продукту з точки зору користувача.

4.1.1 Загальний опис інтерфейсу

Мобільний онлайн-сервіс для автоматизації формування бібліографії наукових публікацій представляє собою веб-додаток із інтуїтивно зрозумілим інтерфейсом.

Головне вікно програми складається з таких основних елементів:

1. Верхня панель - містить назву сервісу та перемикач теми оформлення (світла/темна/автоматична);
2. Панель вибору типу джерела - дозволяє обрати тип бібліографічного джерела (стаття, книга, веб-сторінка тощо);
3. Панель вибору стилів - дозволяє обрати стандарти оформлення бібліографічних посилань (ДСТУ 8302:2015, APA, IEEE);
4. Форма введення даних - містить поля для введення бібліографічної інформації;
5. Кнопка формування опису - запускає процес генерації бібліографічного опису;
6. Секція результатів - відображає результати форматування у вигляді готових бібліографічних описів.

4.1.2 Послідовність дій користувача

Наведено послідовність дій користувача при використанні програмного продукту:

Крок 1: Вибір типу джерела

Для початку роботи користувач повинен обрати тип бібліографічного джерела (рис. 4.1).

Доступні наступні типи:

- стаття - для наукових статей у журналах та періодичних виданнях;
- книга - для монографій, підручників та інших неперіодичних видань;
- матеріали конференції - для доповідей, тез та збірників конференцій;
- веб-сторінка - для електронних ресурсів та онлайн-джерел;
- дисертація - для дисертацій, авторефератів та інших кваліфікаційних робіт.

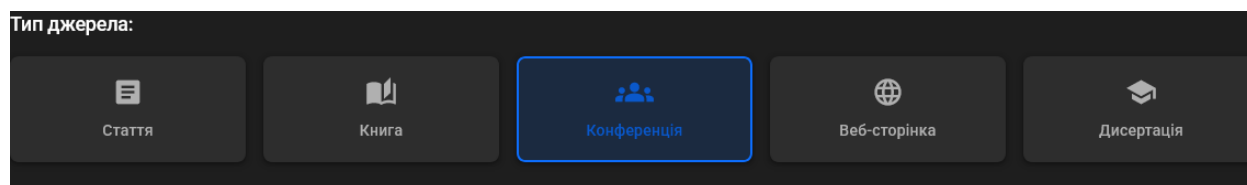


Рисунок 4.1 - Вибір типу джерела

Крок 2: Вибір стилів форматування (рис. 4.2).

Після вибору типу джерела користувач повинен обрати один або кілька стилів форматування бібліографічних посилань.

Доступні наступні стилі:

- ДСТУ 8302:2015 - український національний стандарт бібліографічного опису;
- APA (7th Edition) - стиль Американської психологічної асоціації, широко використовується у соціальних науках;

- IEEE - стиль Інституту інженерів електротехніки та електроніки, використовується в технічних публікаціях.

При наведенні курсора на назву стилю відображається його короткий опис.

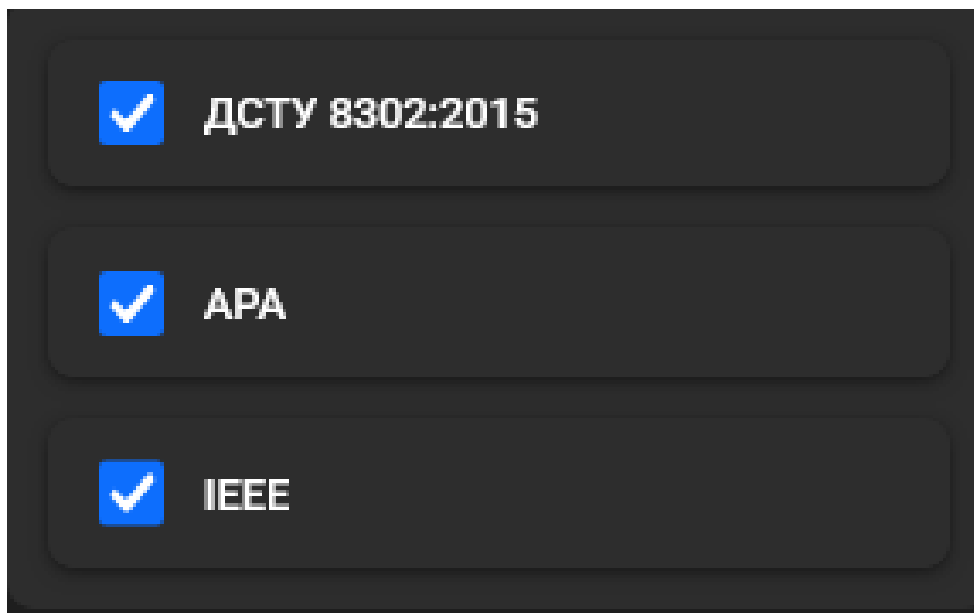


Рисунок 4.2 - Вибір стилів форматування

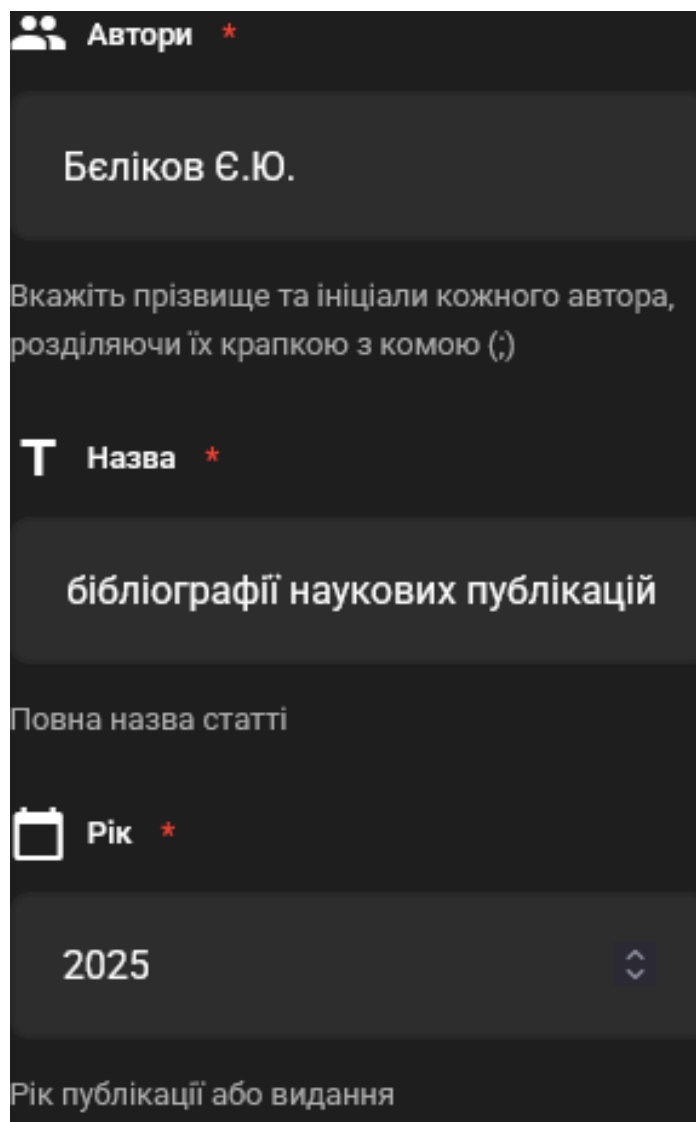
Крок 3: Заповнення форми

Залежно від обраного типу джерела та стилів форматування, форма автоматично адаптується, показуючи необхідні поля.

Обов'язкові поля позначені спеціальним маркером і додатково виділені кольором.

Основні поля, спільні для всіх типів джерел:

- автори - прізвища та ініціали авторів, розділені комами;
- назва - повна назва публікації;
- рік - рік видання;
- DOI - цифровий ідентифікатор об'єкта (якщо є).



Автори *

Беліков Є.Ю.

Вкажіть прізвище та ініціали кожного автора, розділяючи їх крапкою з комою (,)

Т Назва *

бібліографії наукових публікацій

Повна назва статті

Рік *

2025

Рік публікації або видання

Рисунок 4.3 - Заповнення форми

Крок 4: Генерація бібліографічного опису

Після заповнення всіх необхідних полів користувач повинен натиснути кнопку "Створити бібліографічний опис".

Система перевірить коректність введених даних і, у разі відсутності помилок, згенерує бібліографічні описи відповідно до обраних стилів (рис.4.4).

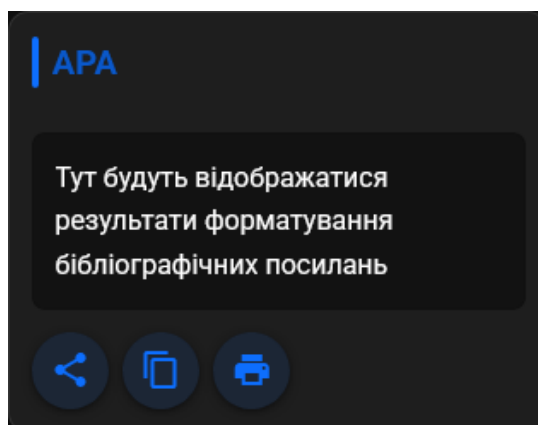


Рисунок 4.4 - Результати форматування

Приклад сформованих результатів форматування наведено у Додатку А.

Крок 5: Робота з результатами

Секція результатів відображає згенеровані бібліографічні описи. Для кожного стилю форматування створюється окремий блок з результатом.

Користувач може:

- копіювати опис до буфера обміну, натиснувши кнопку "Копіювати";
- роздрукувати опис, натиснувши кнопку "Друкувати";
- поділитися описом через стандартний діалог поширення.

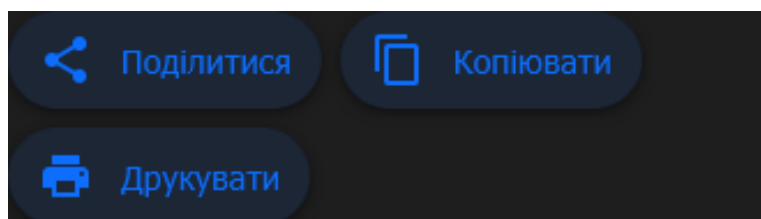


Рисунок 4.5 - Робота з результатами

4.1.3 Додаткові можливості

Збереження та відновлення даних

Програма автоматично зберігає всі введені дані локально в браузері. При повторному відвідуванні сервісу, форма буде заповнена останніми введеними значеннями (рис.4.6).

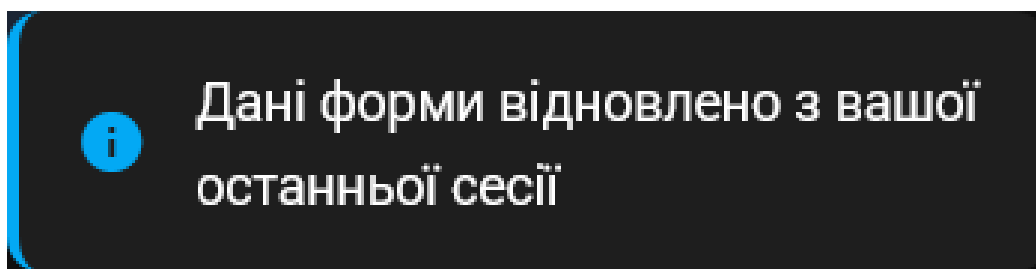


Рисунок 4.6 - Відновлення даних.

Робота в офлайн-режимі

Сервіс підтримує роботу без підключення до Інтернету. Після першого відвідування, всі необхідні ресурси кешуються, і програма залишається повністю функціональною навіть при відсутності з'єднання.

При роботі в офлайн-режимі в правому нижньому куті екрану відображається індикатор "Offline Mode" (рис. 4.7).



Рисунок 4.7 - Індикатор офлайн-режиму

Адаптація до мобільних пристроїв

На мобільних пристроях інтерфейс автоматично адаптується:

- результати форматування відображаються в форматі слайдів з можливістю переходу між ними свайпами;
- кнопка "Створити бібліографічний опис" фіксується внизу екрана для зручності;
- елементи керування збільшуються для комфортної роботи на сенсорних екранах.

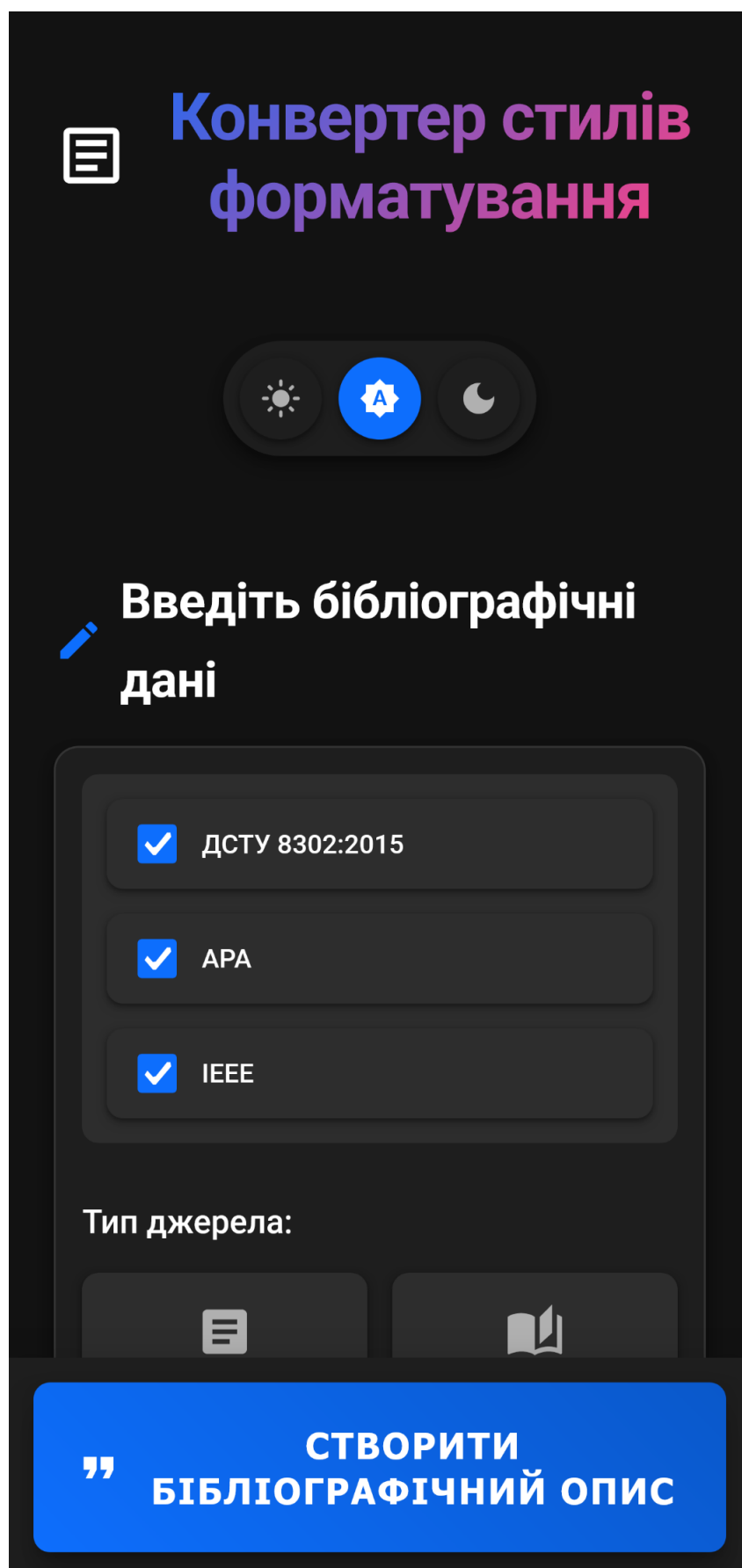


Рисунок 4.8 - Інтерфейс на мобільному пристрої

Перемикання теми оформлення

Користувач може обрати (рис. 4.9) один із трьох варіантів оформлення інтерфейсу:

- світла тема - світлий фон, темний текст;
- темна тема - темний фон, світлий текст;
- авто - автоматичне перемикання теми згідно з системними налаштуваннями пристрою.

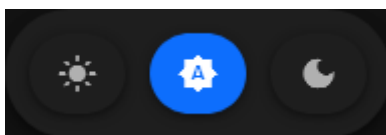


Рисунок 4.9 - Перемикання теми оформлення

4.1.4 Повідомлення про помилки

При виникненні помилок система відображає інформативні повідомлення з описом проблеми:

- помилки валідації - відображаються при відсутності обов'язкових полів або некоректних даних;
- системні помилки - відображаються при збоях у роботі програми.
-

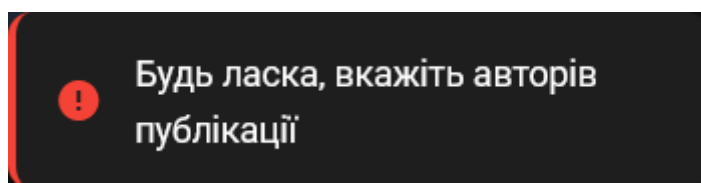


Рисунок 4.10 - Приклад повідомлення про помилку

Всі повідомлення відображаються у вигляді спливаючих повідомлень (toast notifications) у верхній частині екрану і автоматично зникають через кілька секунд.

4.1.5 Встановлення як PWA

Сервіс реалізовано як прогресивний веб-додаток (Progressive Web Application, PWA), що дозволяє "встановити" його на пристрій:

1. На мобільному пристрої: відкрийте сервіс у браузері та виберіть опцію "Додати на головний екран" у меню браузера.
2. На комп'ютері: відкрийте сервіс у Chromium-based браузері і натисніть кнопку "Встановити" у правій частині адресного рядка.

Після встановлення сервіс буде доступний через іконку на головному екрані пристрою і працюватиме як звичайний мобільний додаток.

4.2 Основи тестування веб-додатків

Тестування веб-додатків є невід'ємним процесом розробки сучасних інформаційних систем. Для веб-додатків характерні особливості, що відрізняють їх від звичайного програмного забезпечення: багатоплатформність, різноманіття браузерів, адаптивність до різних пристроїв та підтримка офлайн-режимів роботи. Тестування веб-додатків — це процес перевірки відповідності системи функціональним вимогам з урахуванням особливостей роботи в веб-середовищі.

4.2.1 Методологія тестування

Для забезпечення якості розробленого мобільного онлайн-сервісу для автоматизації формування бібліографії наукових публікацій було застосовано комплексний підхід до тестування, що враховує специфіку веб-додатків та PWA.

Рівні тестування:

1. Модульне тестування (Unit Testing) - перевірка окремих компонентів системи (форматерів, валідаторів, утилітарних функцій);

2. Компонентне тестування - перевірка компонентів на коректне відображення та взаємодію;

3. Інтеграційне тестування - перевірка взаємодії між компонентами та сервісами;

4. End-to-End тестування - перевірка повних сценаріїв використання.

Типи тестування:

1. Крос-браузерне тестування - перевірка коректності роботи в різних браузерах;

2. Адаптивне тестування - перевірка відображення на різних пристроях;

3. Тестування швидкодії - аналіз завантаження та роботи додатку;

4. PWA-тестування - перевірка функцій прогресивного веб-додатка;

5. Тестування офлайн-функціональності - перевірка роботи системи без підключення до Інтернету;

6. Тестування доступності (Accessibility) - визначення ефективності зусиль із забезпечення доступності [21].

4.2.2 Визначення тестових ситуацій для бібліографічного сервісу

Для систематичного тестування мобільного онлайн-сервісу для автоматизації формування бібліографії було визначено набір тестових ситуацій, що охоплюють різні аспекти роботи системи.

1. Функціональне тестування

- перевірка формування бібліографічних описів для різних типів джерел та стилів;

- перевірка валідації форми введення даних;

- перевірка копіювання, друку та поширення результатів;

- перевірка збереження та відновлення даних з локального сховища;

- перевірка експорту та імпорту даних.

2. Тестування зручності використання

- перевірка доступності елементів інтерфейсу на різних пристроях;
- перевірка адаптивності інтерфейсу на мобільних пристроях;
- перевірка навігації між результатами на мобільних пристроях;
- перевірка інформативності повідомлень про помилки.

3. Тестування продуктивності

- перевірка швидкості форматування для великих обсягів даних;
- перевірка швидкості завантаження сторінки;
- перевірка використання ресурсів пристрою.

4. Тестування офлайн-функціональності

- перевірка доступності системи без підключення до Інтернету;
- перевірка кешування ресурсів для офлайн-використання;
- перевірка повідомлень про статус підключення.

5. Тестування сумісності

- перевірка роботи в різних браузерях (Chrome, Firefox, Safari, Edge);
- перевірка роботи на різних мобільних пристроях (iOS, Android);
- перевірка встановлення та роботи як PWA.

4.2.3 Результати тестування

Результати випробування програмного забезпечення мобільного онлайн-сервісу для автоматизації формування бібліографії наукових публікацій представлені в таблиці 4.1.

Таблиця 4.1 - Тестові ситуації

№	Очікуваний результат	Статус
1	Коректно сформований опис з усіма заповненими полями	Успішно
2	Відображення повідомлення про помилку	Успішно
3	Текст скопійовано до буфера обміну	Успішно
4	Зміна теми інтерфейсу	Успішно
5	Перехід між результатами різних стилів	Успішно
6	Форма відправляється	Успішно

7	Відображення відповідних полів для обраного типу джерела	Успішно
8	Система працює в офлайн-режимі	Успішно
9	Відображення індикатора при відсутності підключення	Успішно
10	Додаток встановлюється і працює як нативний	Частково
11	Формування має відбуватися за <1 секунди	Успішно
12	Завантаження за <3 секунд на мобільному з'єднанні	Успішно
13	Завантаження за <1 секунду	Успішно

4.2.4 Виявлені проблеми та їх вирішення

У процесі тестування було виявлено ряд проблем, які були успішно вирішені:

1. Проблема: Некоректне форматування імен авторів у стилі IEEE. Рішення: Виправлено алгоритм трансформації авторів у форматері IEEE;

2. Проблема: Втрата даних форми при оновленні сторінки. Рішення: Додано автоматичне збереження даних форми в localStorage після кожної зміни;

3. Проблема: Недостатня контрастність у темній темі. Рішення: Покращено колірну схему для забезпечення відповідності вимогам доступності;

За результатами тестування можна зробити наступні висновки:

1. Розроблений мобільний онлайн-сервіс для автоматизації формування бібліографії наукових публікацій відповідає функціональним вимогам та специфікації;

2. Система має високу сумісність з більшістю сучасних браузерів та пристроїв, хоча й існують певні обмеження для Safari на iOS;

3. Інтерфейс користувача є інтуїтивно зрозумілим, адаптивним та забезпечує ефективну роботу як на десктопних, так і на мобільних пристроях;

4. Система успішно працює в офлайн-режимі, забезпечуючи повну функціональність без підключення до Інтернету;

5. Продуктивність системи відповідає сучасним вимогам до веб-додатків, забезпечуючи швидке завантаження та обробку даних.

Таким чином, розроблений програмний продукт має високу якість та готовий до впровадження в експлуатацію.

4.3 Техніко-економічні показники

Для об'єктивної оцінки ефективності впровадження інформаційної системи необхідно розрахувати основні техніко-економічні показники, які характеризують ефективність впровадження продукту.

4.3.1 Витрати, пов'язані з розробкою продукту.

До даних витрат належать витрати на оплату праці розробників з нарахуваннями, амортизаційні відрахування, витрати на матеріали, електроенергію та інші супутні витрати.

Витрати на оплату праці з нарахуваннями – складаються з витрат на основну заробітну плату, додаткову заробітну плату та нарахувань згідно чинного законодавства (22%).

Витрати на основну заробітну плату – визначаються виходячи з місячного посадового окладу або вартості 1 години роботи розробника та витрат часу, понесених на розробку даного продукту.

Розрахунок основної оплати праці здійснюється за формулою:

$$ЗП_{осн} = \frac{ЗП_{міс}}{ФРЧ} \cdot Тр; \quad (4.1)$$

де $ЗП_{осн}$ – основна заробітна плата, грн;

$ЗП_{міс}$ – місячний посадовий оклад розробника, на підприємстві становить 20500 грн/міс;

ФРЧ – місячний фонд робочого часу, становить 176 (22 дні по 8 год)
год./міс;

Тр – трудомісткість (затрати часу) виготовлення програмного продукту,
год.

Трудомісткість (затрати часу) розробника на створення продукту визначається виходячи з кількості витрачених на розробку робочих днів та тривалості робочого дня:

$$\text{ФРЧ} = \text{Крдм} \cdot \text{Трз}; \quad (4.2)$$

де Крд – кількість робочих днів на створення продукту, приймаємо 18 дн.

Трз – тривалість робочого дня (зміни), год.

Отже,

$$\text{Тр} = 18 \cdot 8 = 144 \text{ год}; \quad (4.3)$$

Таким чином:

$$\text{ЗПосн} = \frac{20500}{176} \cdot 144 = 16761,4 \text{ грн.};$$

У випадку, якщо при розробці продукту задіяні декілька працівників, розмір основної заробітної плати розраховується за кожним.

Додаткова заробітна плата – передбачає різноманітні види доплат за рівень кваліфікації, стаж роботи тощо. Розмір нарахувань здійснюється відповідно до норм прийнятих на підприємстві-замовника.

Відсоток нарахувань додаткової заробітної плати коливається в межах 10-25%. На даному підприємстві додаткова оплата праці становить 15%. Сума додаткової заробітної плати розраховується від розміру основної:

$$\text{ЗПдод} = \text{ЗПосн} \cdot \frac{\% \text{дод.}}{100}. \quad (4.3)$$

де %дод – відсоток нарахувань додаткової заробітної плати, %

$$ЗП_{\text{дод}} = 16761,4 \cdot \frac{20}{100} = 2514,2 \text{ грн.}$$

Нарахування на заробітну плату здійснюються від суми основної та додаткової зарплати.

Розмір нарахувань становить 22%.

$$Нзп = (ЗП_{\text{осн}} + ЗП_{\text{дод}}) \cdot \frac{\%_{\text{нар}}}{100} \quad (4.4)$$

де %нар – відсоток нарахувань заробітної плати,

$$Нзп = (16761,4 + 2514,2) \cdot \frac{22}{100} = 4240,6 \text{ грн.}$$

Розрахунок загальної суми витрат на оплату праці з нарахуваннями розробника наведемо у вигляді таблиці 2.

Таблиця 2 – Розрахунок витрат на оплату праці з нарахуваннями розробників програмного продукту

Найменування посади	Основна заробітна плата, грн.	Додаткова заробітна плата, год.	Розмір нарахувань на оплату праці, грн.	Всього витрат на оплату праці з нарахуваннями, грн.
Інженер-програміст	16761,4	2514,2	4240,6	23516,2

Розрахунок суми амортизаційних відрахувань обладнання, яке використовується при розробці продукту здійснюється найбільш поширеним прямолінійним методом. При створенні продукту

використовувався ноутбук та принтер. Відповідно до податкового кодексу України електронно-обчислювальні машини (ПК) відносять до 4 групи, з терміном корисного використання 2 роки. Відповідно річна норма амортизації становить 50%. Балансова вартість ПК (або ноутбуку) на момент придбання (розрахунку) становить 22000 грн, принтера - початкова вартість 8500 грн, приймаємо річну норму амортизації в розмірі 20% (термін використання 5 років).

Розрахунок річної суми амортизаційних відрахувань:

$$A_{річ} = B_{Вобл} \cdot \frac{\%_{аморт.}}{100}; \quad (4.5)$$

де $B_{Вобл}$ - ціна придбання, або балансова вартість комп'ютера на момент придбання, грн.

$\%_{аморт}$ – річний відсоток нарахувань додаткової заробітної плати, %

$$A_{річ. комп.} = 15000 \cdot \frac{50}{100} = 11000 \text{ грн.}$$

$$A_{річ. принт.} = 10000 \cdot \frac{20}{100} = 1700 \text{ грн.}$$

Сума амортизаційних відрахувань, що увійде до собівартості програмного продукту розраховується з врахуванням часу використання ноутбуку та принтера при розробці продукту.

$$A = A_{річ} \cdot \frac{T_{вик(днів)}}{365} \quad (4.6)$$

де $T_{вик}$ - термін використання персонального комп'ютера та принтера при розробці програмного продукту, міс. або дні.

Таким чином, загальна сума амортизаційних відрахувань, становитиме:

$$A = (11000 + 1700) \cdot \frac{18 \text{ днів}}{365} = 625,5 \text{ грн.}$$

Витрати на матеріали – визначаються відповідно до кількості витрачених матеріалів на розробку продукту та ціни на даний матеріал, на момент розрахунку. Витрати на матеріали, що були використані на розробку продукту наведені в таблиці 3.

Таблиця 3 – Розрахунок матеріальних витрат на створення продукту

Найменування матеріалу	Ціна за одиницю матеріалу, грн./од	Витрачено, од	Вартість витрачених матеріалів, грн.
Чорнила, банка	320	3	960
Папір, уп.	200	3	600
Разом витрат			1560

Витрати на електроенергію – з потужності обладнання, що використовується при розробці продукту ($\Pi=0,55$), коефіцієнту використання потужності ($K_{\text{вп}}=0,95$), фактичного часу (трудомісткості) на розробку та ціни електроенергії, що склалась на момент розрахунку ($\text{Ц}_{1\text{кВт}}=4,32$ грн.):

$$\text{Вел.} = \Pi \cdot K_{\text{вп}} \cdot T_{\text{р}} \cdot \text{Ц}_{1\text{кВт}} = 0,55 \cdot 0,95 \cdot 144 \cdot 4,32 = 325 \text{ грн.}; \quad (4.7)$$

Розрахунок суми витрат на розробку відображені даними таблиці 4.

Таблиця 4 – Розрахунок витрат на створення продукту

Найменування витрат	Вартість, грн.
Витрати на оплату праці з нарахуваннями, грн.	23516,2
Амортизаційні відрахування, грн	625,5

Найменування витрат	Вартість, грн.
Витрати на матеріали, грн	1560
Витрати на електроенергію, грн	325
Разом витрат	26026,7

4.3.2. Витрати на розміщення продукту.

Для публічного розміщення мобільного сервісу для автоматизації формування бібліографії також необхідно врахувати вартість хостингу та домену. Річна вартість хостингового обслуговування становить орієнтовно 250 грн/ міс, або 3000 грн/рік), а річна вартість доменного імені — приблизно 600 грн.

Таким чином, загальні витрати на розміщення інформаційної системи наведено 3600 грн.

4.3.3. Розрахунок собівартості одиниці копії програмного продукту

Сукупні витрати на створення та розміщення програмного продукту складаються з загальної суми витрат:

$$\text{Соб. прод} = \text{Вроз.} + \text{Врозм.} + \text{Врек} + \text{Вінш.} \quad (4.8)$$

де Врозр. – витрати на розробку програмного продукту, грн;

Врозм.с. - витрати, пов'язані з розміщенням інформації на сервері та вартість хостингу сайту, грн;

Врек – витрати на рекламу, приймаємо на рівні - 3200 грн.

Вінш – інші витрати, які не включені до вищеперелічених витрат, коливаються в межах 10-30% , приймаємо на рівні 20%.

$$\text{Вінш.} = (26026,7 + 3600 + 3200) \cdot \frac{20}{100} = 6565,3 \text{ грн.}$$

Таким чином, витрати на створення продукту складатиме:

$$\text{Вствор.} = 26026,7 + 3600 + 3200 + 6565,3 = 39392 \text{ грн.}$$

Оскільки, програмний продукт приймається та впроваджується за завданням замовника, а також може реалізовуватись іншим покупцям відповідно доцільно визначити ціну продажу. Ціна реалізації залежить від кількості реалізованого продукту. В нашому випадку плануємо реалізацію

15 од продукту. Таким чином, витрати на створення продукту розподіляються на кількість запланованих продажів та становлять собівартість одиниці продукту для продажу. Таким чином, собівартість одиниці продукту становить:

$$\text{Спрод.} = \frac{39392}{15} = 2626,1 \frac{\text{грн}}{\text{од.}}$$

4.3.4. Розрахунок ціни реалізації проектного рішення

Ціна реалізації визначається виходячи з рівня запланованої прибутковості (рентабельності) задля задоволення цільової аудиторії та прибутковості замовника. Нами було проведено аналіз ринкової вартості данного продукту, відповідно до результатів, рівень рентабельності за згодою замовника може коливається в межах 40-80%. Приймаємо рівень прибутковості 70%, рівень податку на додану вартість – згідно чинного законодавства становить 20%.

Таким чином, ціна реалізації продукту розраховується за формулою:

$$\text{Цр} = \text{Спрод} \cdot \left(1 + \frac{P}{100}\right) \cdot \left(1 + \frac{\text{ПДВ}}{100}\right) \quad (4.9)$$

де P – прийнятий норматив рентабельності, 70%.

ПДВ – ставка податку на додану вартість, приймається на рівні 20%.

$$\text{Цр} = 2626,1 \cdot \left(1 + \frac{70}{100}\right) \cdot \left(1 + \frac{20}{100}\right) = 4127,9$$

4.3.5 Розрахунок прибутку від створення та продажу програмного продукту.

Розрахунок чистого прибутку від реалізації запланованого обсягу програмного продукту здійснюється з врахуванням річного прибутку від реалізації та податку на прибуток, що існує на момент розрахунку. При розрахунку прибутку ціна реалізації приймається без ПДВ та становить ціну реалізації власника:

$$\text{Цр власн.} = 2626,1 \cdot \left(1 + \frac{70}{100}\right) = 4464,4 \text{ грн.}$$

Прибуток від продажу розраховується за формулою:

$$П = (\text{Цр власн.} - \text{Спрод}) \cdot \text{Опрод} \quad (4.10)$$

де Опрод. – кількість одиниць продажу програмного продукту, од;

Таким чином загальна сума прибутку від продажу становить:

$$П = (4464,4 - 2626,1) \cdot 15 = 27574,5 \text{ грн.}$$

Відповідно чистий прибуток за відрахуванням податку на прибуток становитиме:

$$\text{ЧП} = П - \left(П \cdot \frac{\%ПП}{100}\right) \quad (4.11)$$

де %ПП – відсоток податку на прибуток, приймається на рівні 18%.

$$\text{ЧП} = 27574,5 - \left(27574,5 \cdot \frac{18}{100}\right) = 22611,1 \text{ грн.}$$

4.3.6. Термін окупності.

Термін окупності характеризує період часу протягом якого окупляться витрати на розробку програмного продукту замовником.

Розрахунок періоду окупності здійснюється за формулою:

$$\text{Ток} = \frac{\text{Вствор.}}{\text{ЧП}}; \quad (4.12)$$

$$\text{Ток} = \frac{39390,8}{22611,1} = 1,74 \text{ роки.}$$

Таким чином, створення та подальший продаж розробленого програмного продукту є достатньо ефективним для замовника. Витрати на створення продукту становлять майже 39,4 тис. грн. При плануванні обсягу подальшого продажу на рівні 15 од., розмір отриманого прибутку для замовника становить 22,6 тис. грн., а період окупності при даних показниках – 1,7 роки. Слід зауважити, що показники ефективності розраховувались за рівнем прибутковості нижчим ринкового показника, також при розрахунках використовували мінімальну кількість проданого продукту, відповідно на перспективу продаж даного продукту є достатньо ефективним.

4.4 Охорона праці

Під час організації робочого місця в умовах домашнього середовища важливо дотримуватися вимог охорони праці з метою забезпечення безпеки, зниження ризику професійних захворювань і підтримки високої працездатності користувача.

4.4.1 Розрахунок освітлення в приміщенні

Коротка характеристика приміщення і виконуваних робіт: у приміщенні розташовано один ноутбук. До складу основного обладнання входять ноутбук та струменевий принтер. Обладнання розміщується таким чином, щоб був забезпечений вільний прохід та зручність використання. Робота в основному пов'язана з використанням ноутбука.

Планування і розміщення робочої зони: відповідно до сучасних ергономічних вимог за ДБН В.2.2-15:2019 "Житлові будинки. Основні положення" та ДСТУ 8604:2015 "Дизайн і ергономіка", меблі та обладнання повинні відповідати антропометричним, фізіологічним, психофізіологічним властивостям людини та обумовленим цими властивостями гігієнічним вимогам з метою збереження здоров'я та забезпечення комфорту при використанні[5].

Стіл робочої зони має висоту 800 мм, довжину 1400 мм, ширину 800 мм, висота сидіння 450 мм, висота простору для ніг - 650 мм, що забезпечує зручність користування. Висота і конструкція робочого столу вибрані так, щоб було комфортно працювати в положенні сидячи.

Розрахунок освітлення: Кімната має розміри 7м на 6,5м. Таким чином, площа складе $S = 45,5 \text{ м}^2$. При висоті $H = 2,5 \text{ м}$ об'єм приміщення складе $V = 113,75 \text{ м}^3$. Природне освітлення - одностороннє.

Приміщення має три вікна висотою 1,5 м, і шириною 2 м, розташованих на одній стіні. Згідно ДБН В.2.5-28:2018 "Природне і штучне освітлення", коефіцієнт природної освітленості (КПО) у житлових приміщеннях при бічному освітленні повинен складати $e_n = 1,5\%$ [4].

Примітка: Коефіцієнт природної освітленості (КПО) - це процентне відношення природної освітленості у будь-якій точці в середині приміщення до одночасно виміряної на тому ж рівні освітленості зовнішньої горизонтальної площини рівномірно розсіяним (дифузійним) світлом усього небосхилу.

Враховуючи коефіцієнт світлового клімату $m = 0,8$ і коефіцієнт сонячності клімату при тому, що вікна приміщення виходять на східний бік, $C = 0,7$, визначаємо нормоване значення КПО для даного приміщення:

$$e = e_n \cdot m \cdot C = 1,5 \cdot 0,8 \cdot 0,7 = 0,84\%.$$

При бічному освітленні КПО можна оцінити за формулою:

$$e = \frac{\tau_0 \cdot r_1 \cdot S_0}{K_3 \cdot \eta_0 \cdot K_{зд} \cdot S_{\Pi}} \cdot 100\%$$

де S_{Π} - площа підлоги приміщення, м²;

S_0 - площа світлових прорізів, м²;

K_3 - коефіцієнт запасу (приймається в межах від 1,2 до 2,0 залежно від можливого забруднення світлових прорізів);

η_0 - світлова характеристика вікон;

r_1 - коефіцієнт, що враховує підвищення коефіцієнта завдяки світлу, відбитому від поверхні приміщення та підстилаючих шару, який прилягає до будівлі;

τ_0 - загальний коефіцієнт світлопропускання, що визначається за формулою:

$$\tau_0 = \tau_1 \cdot \tau_2 \cdot \tau_3 \cdot \tau_4 ; (4.2)$$

де τ_1 - коефіцієнт світлопропускання матеріалу (для різних типів скла приймається в межах від 0,65 до 0,9);

τ_2 - коефіцієнт, що враховує втрати світла в межах вікна (в залежності від виду палітурки приймається в межах від 0,5 до 0,9);

τ_3 - коефіцієнт, що враховує втрати світла в несучих конструкціях (від 0,8 до 0,9);

τ_4 - коефіцієнт, що враховує втрати світла в сонцезахисних пристроях (при їх відсутності $\tau_4 = 1$, при їх наявності тоді приймається від 0,6 до 0,9).

Площа світлових прорізів становить $S_0 = 9\text{ м}^2$. Площа підлоги приміщення становить $S_{\Pi} = 45,5\text{ м}^2$. Для житлових приміщень коефіцієнт запасу приймається $K_3 = 1.2$.

Виходячи зі співвідношення розмірів приміщення і вікон, світлова характеристика вікон $\eta_0 = 10,5$.

Коефіцієнт, що враховує КПО за рахунок відбиття від поверхонь приміщення і підстилаючого шару, для даного приміщення становить $r_1 = 1,3$.

Для вікон з подвійними рамами коефіцієнт світлопропускання $\tau_1 = 0,8$.

Оскільки палітурка вікна подвійна металопластикові, то $\tau_2 = 0,6$.
При бічному освітленні $\tau_3 = 1$.

Як сонцезахисних пристроїв використовуються регульовані внутрішні жалюзі, тому $\tau_4 = 0,9$.

Таким чином, отримуємо:

$$\tau_0 = 0,8 \cdot 0,6 \cdot 1 \cdot 0,9 = 0,43;$$

$$e = \frac{100 \cdot 0,43 \cdot 1,3 \cdot 9}{1,2 \cdot 10,5 \cdot 1 \cdot 45,5} = 0,98\%.$$

Тож коефіцієнт природної освітленості $e = 0,98\%$ задовольняє нормі для житлових приміщень.

Природне освітлення істотно залежить від погодних умов, отже, необхідно передбачити штучне освітлення в похмуру погоду.

Штучне освітлення необхідно ще й тому, що в приміщенні виконуються роботи не тільки у світлий час доби, але і в темний.

4.4.2 Обґрунтування організації робочого місця

Виконаємо обґрунтування організації робочого місця у квартирі на основі прикріпленого зображення (рис. 4.11).



Рисунок 4.11 - Організація робочого місця у квартирі

Як видно з рисунка 4.11, робоче місце організовано з урахуванням ергономічних вимог відповідно до ДСТУ 8604:2015. Робоча зона включає:

1. Дерев'яний стіл з розмірами, що відповідають антропометричним вимогам: висота 800 мм, довжина 1400 мм, ширина 800 мм, з достатнім простором для ніг;
2. Ергономічне офісне крісло з регульованою висотою сидіння та опорою для спини, що забезпечує правильне положення тіла під час роботи;
3. Ноутбук, розташований на оптимальній відстані від краю столу (100-300 мм), що дозволяє комфортно розмістити руки та зберігати правильну поставу;
4. Струменевий принтер, розташований у зоні досяжності користувача;
5. Природне освітлення через вікно, доповнене шторами для регулювання рівня освітленості робочої поверхні.

При роботі за ноутбуком слід дотримуватись наступних рекомендацій для забезпечення комфорту та запобігання перевтомі:

- розташовувати ноутбук таким чином, щоб верхній край екрану знаходився на рівні очей або трохи нижче;
- сидіти прямо, спираючись на спинку крісла, уникаючи нахилу вперед;
- робити перерви кожні 45-60 хвилин для відпочинку очей та зміни положення тіла;
- забезпечувати достатнє освітлення робочої поверхні, уникаючи відблисків на екрані.

Така організація робочого простору в квартирі забезпечує ергономічні умови для навчання та особистого користування комп'ютерною технікою, що сприяє підвищенню продуктивності та зменшенню втоми при тривалій роботі.

ВИСНОВКИ

У результаті виконаної кваліфікаційної роботи досягнуто поставлену мету - розроблено мобільний онлайн-сервіс для автоматизації формування бібліографічних посилань з підтримкою українських стандартів.

Основні результати дослідження:

1. Аналіз існуючих рішень виявив недоліки: недостатню підтримку українських стандартів, обмежену мобільність та високу вартість;
2. Спроектовано архітектуру з використанням принципів прогресивного веб-додатку (PWA), що забезпечує автономність роботи та функціонування офлайн;
3. Реалізовано алгоритми форматування за стандартами ДСТУ 8302:2015, APA та IEEE;
4. Створено адаптивний інтерфейс з підтримкою мобільних пристроїв та світлої/темної тем;
5. Тестування підтвердило відповідність вимогам на різних пристроях.

Практична значущість: сервіс скорочує час оформлення бібліографії в 5-10 разів, мінімізує ризик помилок та забезпечує мобільний доступ без додаткового ПЗ.

Рекомендації щодо практичного використання: впровадження сервісу сприятиме підвищенню якості оформлення наукових публікацій та дотриманню вимог до бібліографії.

Подальший розвиток проєкту може включати розширення підтримки стилів цитування та інтеграцію з текстовими редакторами.

СПИСОК ЛІТЕРАТУРИ

1. Бібліографічне посилання. URL: https://ube.nlu.org.ua/article/Бібліографічне_посилання (дата звернення: 19.05.2025)
2. Бібліографічний опис. URL: https://uk.wikipedia.org/wiki/Бібліографічний_опис (дата звернення: 02.04.2025)
3. Бібліографічні менеджери. URL: <https://chmnu.edu.ua/wp-content/uploads/Bibliografichni-menedzheri.pdf> (дата звернення: 09.05.2025)
4. ДБН В.2.5-28:2018. Природне і штучне освітлення. Київ : Мінрегіон України, 2018. С. 20-22.
5. ДСТУ 8604:2015 Дизайн і ергономіка. Робоче місце для виконання робіт у положенні сидячи. Загальні ергономічні вимоги. Київ : ДП «УкрНДНЦ», 2017.
6. ДСТУ 8302:2015 Інформація та документація. Бібліографічне посилання. URL: https://kubg.edu.ua/images/stories/podii/2017/06_21_posylannia/dstu_8302.pdf (дата звернення: 06.04.2025)
7. Для чого потрібні UML діаграми? URL: <https://foxminded.ua/uml-diagramy/> (дата звернення: 18.04.2025)
8. Міжнародні правила цитування та посилання в наукових роботах. URL: <https://nauka.gov.ua/information/mizhnarodni-pravylyatsytuvannia-ta-posylannia-v-naukovykh-robotakh/> (дата звернення: 05.05.2025)
9. Об'єктна модель документа. URL: <https://ua5.org/information/1955-obyektna-model-dokumenta.html> (дата звернення: 16.04.2025)
10. Сучасні інструменти для створення бібліографії наукової статті. URL: <https://www.akademprostir.com/l/suchasni-instrumenti-dlya-stvorennya-bibliografiji-naukovoji-statti/> (дата звернення: 07.05.2025)

11. За яким стилем оформлювати список літератури? URL: <https://www.grafiati.com/uk/blogs/a-guide-to-citation-styles-which-one-to-choose/> (дата звернення: 04.04.2025)
12. APA style. URL: https://en.wikipedia.org/wiki/APA_style (дата звернення: 08.04.2025)
13. Ater T. Building Progressive Web Apps: Bringing the Power of Native to the Browser. Sebastopol, CA : O'Reilly Media, 2017. 5 с.
14. Citation Machine by Chegg Review. URL: <https://jenni.ai/blog/citation-machine-by-chegg-review> (дата звернення: 17.05.2025)
15. Deming, IT, and BPM IDEF0 Diagrams. URL: <https://bptrends.info/deming-it-and-bpm-idef0-diagrams/> (дата звернення: 14.04.2025)
16. EndNote. URL: <https://en.wikipedia.org/wiki/EndNote> (дата звернення: 15.05.2025)
17. Flanagan D. JavaScript: The Definitive Guide. Sebastopol, CA: O'Reilly Media, 2011. 1 с.
18. Frain B. Responsive Web Design with HTML5 and CSS. 3rd ed. Birmingham : Packt Publishing, 2020. 1 с.
19. Grant K. CSS in Depth. 3rd ed. Shelter Island, NY: Manning Publications, 2018. С. 52-53.
20. Institute of Electrical and Electronics Engineers (IEEE). URL: <https://guides.lib.vt.edu/find/citation-style-manuals/ieee> (дата звернення: 10.04.2025)
21. Kalbag L. Accessibility for Everyone. New York : A Book Apart, 2017. 121 с.
22. Object identifiers. URL: <https://www.tuwien.at/en/research/rti-support/research-data/rdm-infos-tips/persistent-identifiers/object-identifiers> (дата звернення: 12.04.2025)

23. Progressive web apps. URL: https://developer.mozilla.org/en-US/docs/Web/Progressive_web_apps (дата звернення: 20.04.2025)
24. Reference Management: Mendeley. URL: <https://libguides.cam.ac.uk/mendeley/> (дата звернення: 11.05.2025)
25. Zotero. URL: <https://oakland.libguides.com/Zotero> (дата звернення: 13.05.2025)

ДОДАТОК А

Приклади роботи розробленого програмного забезпечення

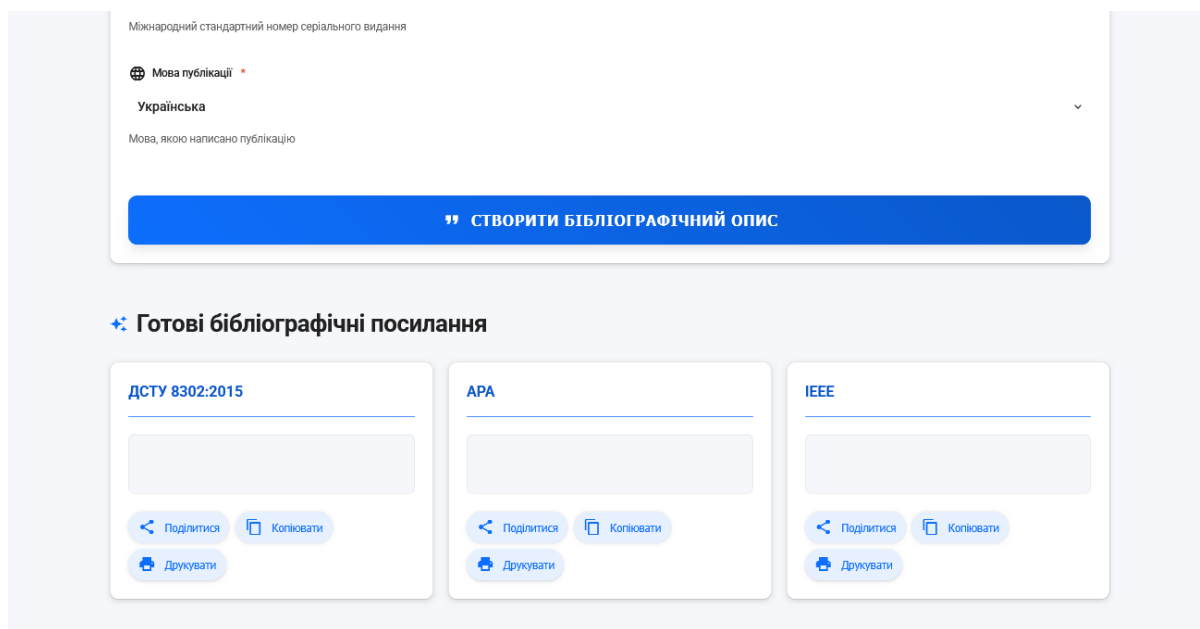


Рисунок А.1 - Приклад світлої теми на десктопному пристрої

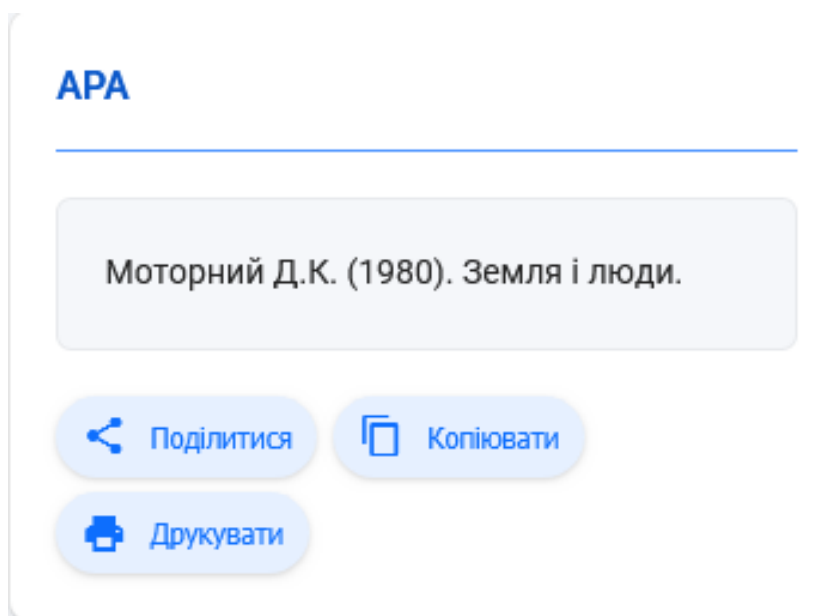


Рисунок А.2 - Сформовані результати форматування


```

32.                case 'book':
33.                    if (data.volume) elements.push(`Т.
${data.volume}`);
34.                    break;
35.
36.                case 'conference':
37.                    if (data.conferenceName)
elements.push(`Матеріали конференції
"${data.conferenceName}"`);
38.                    break;
39.
40.                case 'webpage':
41.                    elements.push('[Електронний
ресурс]');
42.                    if (data.url) elements.push(`URL:
${data.url}`);
43.                    if (data.accessDate) {
44.                        const date = new
Date(data.accessDate);
45.                        const formattedDate =
date.toLocaleDateString('uk-UA', {
46.                            day: '2-digit',
47.                            month: '2-digit',
48.                            year: 'numeric'
49.                        });
50.                        elements.push(`дата звернення:
${formattedDate}`);
51.                    }
52.                    break;
53.
54.                case 'thesis':
55.                    const thesisTypeMap = {
56.                        'PhD': 'дис. ... доктора
філософії',
57.                        'masters': 'магістерська робота',
58.                        'bachelors': 'бакалаврська
робота',
59.                        'doctoral': 'дис. ... докт. наук'
60.                    };
61.
elements.push(thesisTypeMap[data.thesisType] || 'дисертація');
62.                    if (data.institution)
elements.push(data.institution);
63.                    break;
64.                }
65.
66.                // Common elements for all source types
67.                if (data.location || data.publisher) {
68.                    elements.push([data.location,
data.publisher]
69.                        .filter(Boolean)

```

```

70.             .join(': ');
71.         }
72.
73.         elements.push(data.year);
74.         if (data.pages) elements.push(`C.
${data.pages}`);
75.         if (data.doi) elements.push(`DOI:
${data.doi}`);
76.
77.         return elements.filter(Boolean).join('. ');
78.     } catch (error) {
79.         return `${data.authors}. ${data.title}.
${data.year}.`;
80.     }
81. },
82.
83. /**
84.  * Format citation according to APA standard
85.  * @param {Object} data - Citation data
86.  * @returns {string} - Formatted citation
87.  */
88. apa: data => {
89.     try {
90.         const sourceType = data.sourceType ||
'article';
91.         let citation = `${data.authors}
(${data.year}). ${data.title}`;
92.
93.         switch (sourceType) {
94.             case 'article':
95.                 if (data.journal) citation += `
_${data.journal}_`;
96.                 if (data.volume || data.issue) {
97.                     citation += [
98.                         data.volume && `,
99.                         data.issue &&
`(${data.issue})`
100.                    ].filter(Boolean).join('');
101.                 }
102.                 if (data.pages) citation += `,
${data.pages}`;
103.                 break;
104.
105.             case 'book':
106.                 if (data.publisher) {
107.                     if (data.location) citation += `
${data.location}: ${data.publisher}`;
108.                     else citation += `
${data.publisher}`;
109.                 }

```

```

110.                break;
111.            }
112.
113.            if (data.doi) citation += `
https://doi.org/${data.doi}`;
114.
115.            return citation;
116.        } catch (error) {
117.            return `${data.authors} (${data.year}).
${data.title}`;
118.        }
119.    },
120.
121.    /**
122.     * Format citation according to IEEE standard
123.     * @param {Object} data - Citation data
124.     * @returns {string} - Formatted citation
125.     */
126.    ieee: data => {
127.        try {
128.            // Transform authors to put initials first
129.            const transformAuthors = authors => {
130.                return authors.split(', ').map(author =>
131.                {
132.                    const parts = author.split(' ');
133.                    if (parts.length <= 1) return author;
134.                    const lastName = parts.pop();
135.                    const initials = parts.map(name =>
136.                    `${name.charAt(0)}.`).join(' ');
137.                    return `${initials} ${lastName}`;
138.                }).join(', ');
139.            };
140.            const elements =
[`${transformAuthors(data.authors)}, "${data.title}"`];
141.            const sourceType = data.sourceType ||
'article';
142.            // Додаємо специфічну для типу джерела
інформацію
143.            switch (sourceType) {
144.                case 'article':
145.                    if (data.journal) elements.push(`in
_${data.journal}_`);
146.                    break;
147.
148.                case 'book':
149.                    elements.push('book');
150.                    if (data.publisher) {
151.                        if (data.location)
elements.push(`${data.location}: ${data.publisher}`);

```

```

152.                                     else
elements.push(`${data.publisher}`);
153.                                     }
154.                                     break;
155.                                     }
156.
157.                                     // Додаємо том, випуск, сторінки та рік
158.                                     if (data.volume) elements.push(`vol.
${data.volume}`);
159.                                     if (data.issue) elements.push(`no.
${data.issue}`);
160.                                     if (data.pages) elements.push(`pp.
${data.pages}`);
161.                                     elements.push(data.year);
162.
163.                                     let citation = elements.join(', ');
164.
165.                                     // Додаємо DOI, якщо він є
166.                                     if (data.doi) {
167.                                         citation += `.`;
168.                                         citation += ` doi: ${data.doi}`;
169.                                     }
170.
171.                                     return citation;
172.                                 } catch (error) {
173.                                     return `${data.authors}, "${data.title}",
${data.year}`;
174.                                 }
175.                            }
176.    };
177.
178.    /**
179.     * Format citations using the provided data
180.     * @param {Object} data - Citation data
181.     * @returns {Object} - Object with formatted citations
182.     */
183.    export function formatCitations(data) {
184.        // Перевіряємо наявність обов'язкових полів
185.        if (!data.authors || !data.title || !data.year) {
186.            // Повертаємо форматкування за замовчуванням
187.            return {
188.                dstu: `${data.authors || 'Автор не
вказаний'}. ${data.title || 'Назва не вказана'}. ${data.year
|| 'Рік не вказаний'}`,
189.                apa: `${data.authors || 'Автор не вказаний'}
(${data.year || 'n.d.'}). ${data.title || 'Назва не
вказана'}`,
190.                ieee: `${data.authors || 'Автор не
вказаний'}, "${data.title || 'Назва не вказана'}", ${data.year
|| 'n.d.'}`

```

```
191.         };  
192.     }  
193.  
194.     // Форматуємо цитування за кожним стилем  
195.     return {  
196.         dstu: formatters.dstu(data),  
197.         apa: formatters.apa(data),  
198.         ieee: formatters.ieee(data)  
199.     };  
200. }  
201.
```

Опис алгоритму форматування.

Наведений фрагмент коду реалізує основні алгоритми форматування бібліографічних посилань для трьох різних стандартів:

1. ДСТУ 8302:2015 - український національний стандарт
2. АРА - стиль Американської психологічної асоціації
3. IEEE - стиль Інституту інженерів електротехніки та електроніки

Кожен алгоритм враховує особливості відповідного стандарту та типу джерела (стаття, книга, веб-сторінка, матеріали конференції, дисертація). Функція `formatCitations` забезпечує зручний інтерфейс для форматування посилань за всіма підтримуваними стилями одночасно, а також містить механізм обробки неповних даних.