

**ХЕРСОНСЬКИЙ НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ
ФАКУЛЬТЕТ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ ТА ДИЗАЙНУ
КАФЕДРА ІНФОРМАТИКИ І КОМП'ЮТЕРНИХ НАУК**

Пояснювальна записка
до дипломної бакалаврської роботи

на тему:

**Розробка рекомендаційної системи для оптимізації управління
персональною бібліотекою на базі Node.js**

Виконав: студент 4 курсу, групи 4КН
спеціальності 122 «Комп'ютерні науки»
Поповідченко В.О.

Керівник: Вороненко М. О.

Рецензент: Огнєва О. Є.

Факультет	<u>Інформаційних технологій та дизайну</u>
Кафедра	<u>Інформатики і комп'ютерних наук</u>
Рівень вищої освіти	<u>Перший (бакалаврський) рівень</u>
Галузь підготовки	<u>12 «Інформаційні технології»</u>
Освітньо – професійна програма	<u>Комп'ютерні науки</u>
Спеціальність	<u>122 «Комп'ютерні науки»</u>

ЗАТВЕРДЖУЮ

Завідувач кафедри ІКН,

д.т.н., професор

_____ В.І. Литвиненко

«_____» _____ 2025 року

ЗАВДАННЯ

НА ДИПЛОМНУ РОБОТУ СТУДЕНТУ

Поповідченка Вадима Олексійовича

(прізвище, ім'я, по батькові)

1. Тема роботи: : **«Розробка рекомендаційної системи для оптимізації управління персональною бібліотекою на базі Node.js»**

2. Керівник роботи Вороненко Марія Олександрівна, кандидат технічних наук, доцент кафедри інформатики і комп'ютерних наук

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

затверджені наказом ХНТУ від « » 2025 р. № №

3. Вихідні дані до роботи: Аналіз функціональних і нефункціональних вимог до рекомендаційних систем, дослідження існуючих сервісів управління медіаконтентом, визначення основних принципів персоналізації. Вивчення

особливостей застосування технологій Node.js, MongoDB, Redis і React у побудові сучасних вебсистем.

4. Зміст розрахунково-пояснювальної записки: Вступ; 1. Аналіз предметної області та існуючих рішень; 2. Проєктування системи; 3. Розробка програмного продукту; 4. Тестування та аналіз результатів; 5. Перспективи розвитку; Висновки;

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

Рисунків – 68.

Таблиць – 15.

Формул – 4.

6. Консультанти розділів роботи

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв
1.	Вороненко М.О., к.т.н., доцент кафедри інформатики і комп'ютерних наук	05 лютого 2025 року	20 травня 2025 року
2.	Вороненко М.О., к.т.н., доцент кафедри інформатики і комп'ютерних наук	19 лютого 2025 року	27 травня 2025 року
3.	Вороненко М.О., к.т.н., доцент кафедри інформатики і комп'ютерних наук	24 березня 2025 року	02 червня 2025 року
4.	Вороненко М.О., к.т.н., доцент кафедри інформатики і комп'ютерних наук	29 квітня 2025 року	06 червня 2025 року
5.	Вороненко М.О., к.т.н., доцент кафедри інформатики і комп'ютерних наук	03 травня 2025 року	08 червня 2025 року

Дата видачі завдання 26.02.2025

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів дипломного роботи	Строк виконання етапів роботи	Примітка
1	Формування теми дипломної роботи, постановка задачі та написання вступу.	26.02.2025 – 13.03.2025	Виконано
2	Аналіз існуючих систем рекомендацій та технологічного стеку.	14.03.2025 – 24.03.2025	Виконано
3	Проектування архітектури та бази даних.	25.03.2025 – 31.03.2025	Виконано
4	Розробка серверної частини(Node.js + MongoDB).	01.04.2025 – 13.04.2025	Виконано
5	Реалізація клієнтської частини(React).	14.04.2025 - 10.05.2025	Виконано
6	Розробка системи рекомендацій, кешування та онбордингу.	11.05.2025 – 16.05.2025	Виконано
7	Проведення тестування та оцінка ефективності.	17.05.2025 – 30.05.2025	Виконано
8	Оформлення пояснювальної записки, підготовка до захисту.	31.05.2025 – 11.06.2025	Виконано

Студент

Поповідченко В. О.

(підпис)

(прізвище та ініціали)

Керівник роботи

Вороненко М.О.

(підпис)

(прізвище та ініціали)

ЗМІСТ

АНОТАЦІЯ.....	7
ABSTRACT	8
ПЕРЕЛІК УМОВНИХ СКОРОЧЕНЬ.....	9
ВСТУП.....	11
РОЗДІЛ 1. АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ ТА ІСНУЮЧИХ РІШЕНЬ.....	14
1.1.Огляд рекомендаційних систем.....	14
1.2. Аналіз існуючих систем управління персональними бібліотеками	17
1.3. Обґрунтування вибору технологічного стеку.....	21
1.4. Аналіз підходів до персоналізації	25
1.5. Постановка задачі розробки	27
1.6. Економічне обґрунтування розробки.....	30
РОЗДІЛ 2. ПРОЄКТУВАННЯ СИСТЕМИ	33
2.1. Архітектура системи.....	33
2.2. Проєктування бази даних	37
2.3. Проєктування алгоритмів рекомендацій.....	41
2.4. Проєктування системи обробки зворотного зв'язку	45
2.5. Проєктування системи кешування	48
2.6. Проєктування пояснювальної системи рекомендацій.....	51
2.7. Проєктування інтерфейсу користувача.....	52
РОЗДІЛ 3. РОЗРОБКА ПРОГРАМНОГО ПРОДУКТУ	55
3.1. Розробка серверної частини (Backend)	55
3.2. Розробка сервісу рекомендацій	61
3.3. Розробка системи кешування	70
3.4. Розробка клієнтської частини (Frontend)	73

3.5. Інтеграція компонентів системи	84
РОЗДІЛ 4. ТЕСТУВАННЯ ТА АНАЛІЗ РЕЗУЛЬТАТІВ	91
4.1. Методологія тестування системи	91
4.2. Функціональне тестування	94
4.3. Тестування продуктивності	103
4.4. Оцінка якості рекомендацій	113
Адаптивність системи	116
4.5. Аналіз стабільності та надійності	118
РОЗДІЛ 5. ПЕРСПЕКТИВИ РОЗВИТКУ	121
5.1. Технічні покращення	121
5.2. Функціональні розширення	124
5.3. Алгоритмічні покращення	125
5.4. Комерційні перспективи	126
ВИСНОВКИ	128
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ	130
Додаток А. Код основних алгоритмів рекомендацій	133
Додаток Б. Діаграми архітектури системи	140
Додаток В. API документація (Swagger)	141

АНОТАЦІЯ

Пояснювальна записка до кваліфікаційної роботи: 141 сторінок, 52 рисунки, 15 таблиць, 4 формули, 30 використаних джерел.

Об'єкт дослідження: процес персоналізованого керування мультимедійною бібліотекою із залученням сучасних рекомендаційних алгоритмів. У фокусі автоматизація добору контенту відповідно до інтересів користувача, а також оптимізація структури зберігання медіаданих.

Предмет дослідження: вебтехнології, які застосовуються для створення клієнт-серверних систем: JavaScript/React для розробки інтерфейсу користувача, Node.js з Express як серверна основа, MongoDB як база даних, Redis для кешування, а також використання зовнішніх API з метою наповнення бібліотеки й формування рекомендацій.

Мета дослідження: створення повнофункціональної вебплатформи для зручного керування персональною колекцією медіа — фільмів, серіалів, книг і музики. У межах реалізації було спроектовано архітектуру клієнт-серверної взаємодії, розроблено власне API для роботи з медіаданими, впроваджено кешування, створено систему онбордингу нових користувачів, а також проведено тестування й аналіз ефективності рекомендаційного модуля.

Ключові слова: рекомендаційна система, медіабібліотека, персоналізація, Node.js, React, MongoDB, кешування, API, Redis.

ABSTRACT

Explanatory note to the qualification work: 141 pages, 52 figures, 15 tables, 4 formulas, 30 sources.

Research object: The process of personalized management of a multimedia library using modern recommendation algorithms. The focus is on automating content selection based on user interests and optimizing the structure of media data storage.

Research subject: Web technologies applied to the development of client-server systems: JavaScript/React for frontend development, Node.js with Express as the backend framework, MongoDB as the database, Redis for caching, and integration of external APIs (TMDB, Spotify, Google Books) to populate the library and generate personalized recommendations.

Purpose of the study: To develop a fully functional web platform for convenient management of a personal media collection, including movies, TV shows, books, and music. The implementation includes designing the architecture of client-server interaction, developing a custom API for media data handling, introducing caching mechanisms, building a user onboarding system, and conducting functional testing along with an efficiency analysis of the recommendation module.

Keywords: recommendation system, media library, personalization, Node.js, React, MongoDB, caching, API, Redis.

ПЕРЕЛІК УМОВНИХ СКОРОЧЕНЬ

Скорочення, термін, позначення	Пояснення
API	Application Programming Interface – інтерфейс прикладного програмування
JWT	JSON Web Token – веб-токен у форматі JSON, використовується для авторизації
JSON	JavaScript Object Notation – нотація об’єктів JavaScript
HTTP	HyperText Transfer Protocol – протокол передавання гіпертексту
CRUD	Create, Read, Update, Delete – створення, читання, оновлення, видалення
REST	Representational State Transfer – архітектурний стиль «передача репрезентативного стану» для веб-сервісів
UI	User Interface – інтерфейс користувача
UX	User Experience – досвід користувача
SQL	Structured Query Language – структурована мова запитів
NoSQL	Not Only SQL – нереляційні бази даних, дослівно «не тільки SQL»
GDPR	General Data Protection Regulation – Загальний регламент захисту даних
CORS	Cross-Origin Resource Sharing – механізм спільного використання ресурсів між різними доменами

DOM	Document Object Model – об’єктна модель документа
MVP	Minimum Viable Product – мінімально життєздатний продукт
SaaS	Software as a Service – програмне забезпечення як послуга
SWOT	Strengths, Weaknesses, Opportunities, Threats – аналіз сильних і слабких сторін, можливостей та загроз
RAM	Random Access Memory – оперативна пам’ять
HTML	HyperText Markup Language – мова розмітки гіпертексту
npm	Node Package Manager – менеджер пакетів Node.js
JSDoc	JavaScript Documentation – формат документування коду JavaScript
СУБД	система управління базами даних
РС	рекомендаційна система

ВСТУП

Актуальність теми

У сучасну цифрову добу обсяг доступної інформації зростає з небаченою швидкістю, особливо в сфері розваг. Щодня з'являються нові фільми, серіали, книги, музичні альбоми — усе це створює ефект перенасичення інформацією. Згідно з даними компанії Nielsen, користувач витрачає в середньому 18 хвилин лише на те, щоб обрати фільм для перегляду [6]. Наприклад, на платформі Netflix представлено понад 15 000 позицій контенту, а на Disney+ — близько 7 500. У результаті виникає так званий «парадокс вибору»: чим більше доступних варіантів, тим складніше ухвалити рішення [5].

Аналогічна ситуація спостерігається і в Україні, де через зростання популярності стрімінгових сервісів користувачам стає дедалі важче орієнтуватися в океані контенту. У таких умовах постає нагальна потреба в інструментах, що дозволяють швидко знаходити релевантні матеріали, адаптовані до індивідуальних уподобань.

У цьому контексті особливу роль відіграють рекомендаційні системи — програмні інструменти, здатні автоматично формувати персоналізовані добірки на основі аналізу активності користувача. Такі системи вже активно функціонують на платформах на зразок Netflix, Spotify чи YouTube, однак у сегменті персональних медіабібліотек подібні рішення практично не застосовуються. Нерідко користувачі формують власні списки улюблених фільмів, серіалів, книг чи музики, але не мають зручного способу ефективного управління ними або отримання персоналізованих порад на їхній основі. Це створює підґрунтя для розробки універсального масштабованого інструменту, який би поєднував функції медіатеки та інтелектуальних рекомендацій.

Мета роботи

Головна мета дипломного проєкту — створити вебсервіс, що надає змогу ефективно управляти особистою медіабібліотекою користувача та генерувати персоналізовані рекомендації, застосовуючи сучасні алгоритми персоналізації.

Завдання роботи

У межах поставленої мети було визначено такі основні завдання:

- 1) Дослідити сучасні підходи до розробки рекомендаційних систем.
- 2) Спроектувати архітектуру з чіткою структурою логічних компонентів.
- 3) Реалізувати серверну частину з використанням Node.js і Express.js.
- 4) Розробити клієнтську частину інтерфейсу на базі React.
- 5) Імплементувати базові алгоритми рекомендацій: контентні, колаборативні та гібридні.
- 6) Інтегрувати API TMDb для отримання інформації про фільми та серіали.
- 7) Реалізувати функціонал управління особистою бібліотекою користувача.
- 8) Побудувати систему зворотного зв'язку з адаптацією рекомендацій.
- 9) Впровадити механізм кешування для підвищення продуктивності.
- 10) Провести тестування і здійснити оцінку ефективності впроваджених рішень.

Об'єкт роботи

Об'єктом дослідження є процес управління персональною медіатекою користувача та генерація рекомендацій на основі його активності.

Предмет роботи

Предметом є алгоритми побудови рекомендацій, методи персоналізації, принципи веброзробки, а також практичне застосування таких технологій як Node.js, React, MongoDB, Redis.

Методи, що застосовуються

У процесі реалізації дипломного проєкту були використані такі методи:

- системний та функціональний аналіз існуючих рішень;
- об'єктно-орієнтоване моделювання структури майбутньої системи;

- використання технологій веброзробки у середовищі Node.js та React;
- застосування статистичних методів для формування рекомендацій;
- тестування системи з використанням відповідних метрик якості.

Практична значущість

Створена система дає змогу користувачам ефективно організовувати власні медіаархіви та отримувати персоналізовані поради, що підвищує якість користувацького досвіду. Крім того, вона має потенціал для подальшого масштабування та комерційного застосування. У майбутньому проєкт може стати основою для стартапу, використовуватися в навчальних цілях або інтегруватися у вже існуючі освітні платформи.

Апробація: *Поповідченко Вадим, Вороненко Марія, Створення рекомендаційної системи для оптимізації управління персональною бібліотекою на базі Node.js. Матеріали Міжнародної науково-практичної інтернет-конференції «Ways of Science Development in Modern Crisis Conditions» (4 червня 2025 р., м. Дніпро) : Збнаук.праць.*

– Vol. 8 (№ 1). м. Дніпро. Україна : FOP Marenichenko V.V., 2025, 343 с.
– ISBN 978-617-8293-48-2