

Міністерство освіти і науки України
Луцький національний технічний університет
Факультет комп'ютерних та інформаційних технологій
Кафедра інженерії програмного забезпечення

КВАЛІФІКАЦІЙНА РОБОТА
ЗА СТУПЕНЕМ ВИЩОЇ ОСВІТИ «БАКАЛАВР»

РОЗРОБКА СОЦМЕРЕЖІ ОБМІНУ КНИГАМИ «BOOK FIND
LNTU» З ВИКОРИСТАННЯМ JAVA, REACT TA EXPRESS.JS

DEVELOPMENT OF "BOOK FIND LNTU" BOOK SHARING
SOCIAL NETWORK USING JAVA, REACT AND EXPRESS.JS

спеціальність 121 Інженерія програмного забезпечення
(шифр і назва спеціальності)

освітня програма «Інженерія програмного забезпечення»
(назва освітньої програми)

Виконав: здобувач вищої освіти
Групи ІПЗ-41
Лісковчук Сергій Михайлович


(підпис)

Керівник:
асистент
Вознюк Анастасія Вадимівна


(підпис)

Кваліфікаційну роботу
допущено до захисту
« 8 » 06 2024 р.
к.т.н., доцент
Гарант освітньої програми:
Ліщина Наталія Миколаївна

(підпис)

Луцьк – 2024 року

ЛУЦЬКИЙ НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ

Факультет комп'ютерних та інформаційних технологій
Кафедра інженерії програмного забезпечення
Ступінь вищої освіти: бакалавр
Галузь знань: 12 Інформаційні технології
Спеціальність: 121 Інженерія програмного забезпечення
Освітня програма: «Інженерія програмного забезпечення»

ЗАТВЕРДЖУЮ

Завідувач кафедри

М. М. Миченко
« 2 » 02 2024 р.

ЗАВДАННЯ НА КВАЛІФІКАЦІЙНУ РОБОТУ ЗДОБУВАЧУ ВИЩОЇ ОСВІТИ

Лісковчуку Сергію Михайловичу

(прізвище, ім'я, по батькові)

1. Тема кваліфікаційної роботи: Розробка соціальної мережі обміну книгами «Book Find LNTU» з використанням Java, React та Express.js

Керівник роботи: асистент, Вознюк Анастасія Вадимівна

затверджені наказом закладу вищої освіти від «30» грудня 2023 р. № 477/01-02

2. Строк подання здобувачем вищої освіти кваліфікаційної роботи «5» червня 2024 р.

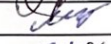
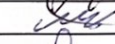
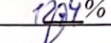
3. Вихідні дані до роботи: технічне та програмне забезпечення ЕОМ, вимоги до розробки об'єкту проектування.

4. Зміст розрахунково-пояснювальної записки (перелік питань, що потрібно розробити):

Аналіз сучасного стану проблеми, існуючих методів і засобів її розв'язання, аналіз, визначення вимог до розроблюваного сайту та проектування сайту, опис функціонального наповнення об'єкта проектування, практична реалізація об'єкта проектування.

5. Перелік графічного матеріалу: 13 рисунків; 1 таблиця; 2 діаграми.

6. Консультанти розділів роботи

Розділ	Прізвище, ініціали та посада консультанта	Підпис	
		завдання видав	завдання прийняв
Аналіз предметної області	Вознюк А. В.		
Специфікації вимог до розробленої системи	Вознюк А. В.		
Розробка об'єкта проектування	Вознюк А. В.		
Нормоконтроль	Повстяна Ю. С.		
Гарант ОП	Ліщина Н. М.		
Показник запозичень тексту		12,4%	
Академічна доброчесність	Яцук А. А.		

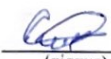
7. Дата видачі завдання «2» лютого 2024 р.

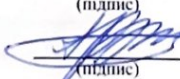
КАЛЕНДАРНИЙ ПЛАН

№ 3/п	Назва етапів кваліфікаційної роботи бакалавра	Строк виконання етапів роботи	Примітка
1	Огляд літературних джерел по темі кваліфікаційної роботи бакалавра	07.02.2024 р.	Зачитано
2	Аналіз проблеми розробки та впровадження об'єкту проектування	27.02.2024 р.	Зачитано
3	Обґрунтування вибору шляхів, технологій (алгоритмів) і засобів вирішення поставленого завдання	12.03.2024 р.	Зачитано
4	Розробка функціонально-структурної схеми роботи об'єкта проектування та проектування бази даних	17.04.2024 р.	Зачитано
5	Практична реалізація об'єкта проектування та розробка бази даних	18.05.2024 р.	Зачитано
6	Тестування та налагодження об'єкта проектування	01.05.2024 р.	Зачитано
7	Здача чистового варіанту кваліфікаційної роботи бакалавра на кафедру	05.06.2024 р.	Зачитано

Здобувач вищої освіти

Керівник кваліфікаційної роботи


(підпис)


(підпис)

Лісковчук С. М.
(прізвище, ініціали)

Вознюк А. В.
(прізвище, ініціали)

Міністерство освіти і науки України
Луцький національний технічний університет

**Рецензія
на кваліфікаційну роботу**

Здобувач вищої освіти: Лісковчук Сергій Михайлович

Тема: «Розробка соцмережі обміну книгами "Book Find LNTU" з використанням Java, React та Express.js»

Коротка характеристика кваліфікаційної роботи: Робота присвячена розробці соціальної мережі для обміну книгами "Book Find LNTU" з використанням Java, React та Express.js. Вона спрямована на створення платформи, яка сприятиме взаємодії між користувачами та полегшить процес обміну літературними ресурсами. Використання сучасних технологій забезпечує високу продуктивність та зручність в обслуговуванні проекту.

Самостійні розробки і пропозиції автора: У роботі узагальнено методи створення соціальної мережі для обміну книгами з використанням Java, React та Express.js. Доведено ефективність запропонованих технологій та архітектурних рішень, що забезпечують високу продуктивність, масштабованість та зручність користування платформою.

Практичне значення роботи: Розроблена методика успішно використовувалась для створення соціальної мережі обміну книгами. Кваліфікаційна робота має важливе практичне застосування, демонструючи потенціал для впровадження в реальному середовищі та підтверджуючи його ефективність.

Недоліки: В розробці можна було б удосконалити документацію проекту, щоб забезпечити більшу зрозумілість та доступність для майбутніх розробників та адміністраторів системи.

Загальний висновок: Робота відзначається високим рівнем та системним підходом до розробки соціальної мережі. Її логічна структура та дотримання вимог підтверджують знання автора у сфері програмування та веб-розробки. Результати роботи можуть бути успішно представлені на захисті та заслуговують на позитивну оцінку.

Рецензент: к.т.н., доцент, Редько Ольга Іванівна

Рецензент кваліфікаційної роботи


(підпис)

«07» червня 2024 р.

Міністерство освіти і науки України
Луцький національний технічний університет

**Факультет комп'ютерних та інформаційних технологій
Кафедра інженерії програмного забезпечення**

ВІДГУК
КЕРІВНИКА НА КВАЛІФІКАЦІЙНУ РОБОТУ БАКАЛАВРА
Здобувач вищої освіти Лісковчук Сергій Михайлович
група ІІЗ-41

Тема кваліфікаційної роботи: Розробка соцмережі обміну книгами «Book Find LNTU» з використанням Java, React та Express.js

Керівник: асистент Вознюк Анастасія Вадимівна

Актуальність теми: тема дослідження і розробки соціальної мережі для обміну книгами є надзвичайно актуальною в сучасних умовах розвитку інформаційного суспільства. Незважаючи на те, що існуючі платформи для обміну книгами та рекомендацій вже існують, все ще залишаються незаповнені ніші в таких аспектах, як соціальна взаємодія між користувачами, зручність користування, можливості інтеграції з іншими сервісами та персоналізація рекомендацій. Така мережа може не лише сприяти популяризації читання та обміну літературою, але й створити активну спільноту книголюбів, що буде сприяти розвитку культури читання та спільного використання ресурсів.

Об'єкт дослідження – це процес обміну книгами між користувачами та результат розробки соціальної мережі для цього обміну.

Характеристика теоретичного рівня, наявності самостійних розробок і практичної значимості роботи: створено ефективну соціальну мережу для обміну книгами, яка сприяє взаємодії між користувачами та забезпечує зручний доступ до літературних ресурсів; розроблено цілісну сучасну архітектуру застосунку, яка включає фронт-енд, бек-енд частини та базу даних, виокремлено компоненти та сервіси; впроваджено систему рекомендацій на основі інтересів та вподобань користувачів; протестовано дану та проведено порівняльний аналіз розробленого рішення з існуючими платформами для обміну книгами.

Зауваження та недоліки: істотних недоліків не виявлено.

Загальний висновок робота виконана на високому рівні та, за умови успішного захисту, заслуговує на позитивну оцінку.

Керівник


(підпис)

Вознюк Анастасія Вадимівна
(прізвище, ім'я, по батькові)

«07» червня 2024 р.

АНОТАЦІЯ

Лісковчук С. М. Розробка соцмережі обміну книгами «Book Find LNTU» з використанням Java, React та Express.js. Рукопис.

Кваліфікаційна робота бакалавра ОП «Інженерія програмного забезпечення». Луцький національний технічний університет. Луцьк, 2024.

Кваліфікаційна робота бакалавра складається зі вступу, трьох розділів, висновків і пропозицій, списку використаних джерел та додатків.

У першому розділі здійснено аналіз сучасного стану проблеми розробки сайтів соціальних мереж для обміну книгами і сформульовано завдання. В другому розділі визначено вимоги до розроблюваної системи, а також засоби, методи і алгоритми розробки. У третьому розділі розроблено систему для обміну книгами. У висновках узагальнено інформацію, відображену у попередніх частинах. Результати розробки демонструють можливості соціальної мережі для обміну книгами.

Ключові слова: соціальні мережі, Java, Express.js, React.

ABSTRACT

Liskovchuk S. M. Development of the book sharing social network "Book Find LNTU" using Java, React and Express.js. Manuscript.

Bachelor's qualifying thesis of the educational program "Software Engineering". Lutsk National Technical University. Lutsk, 2024.

The bachelor's qualifying thesis consists of an introduction, three sections, conclusions and proposals, a list of used sources and annexes.

In the first chapter, an analysis of the current state of the problem of developing developing social networking sites for book sharing was carried out and the task was formulated. The second section defines the requirements for the developed system, as well as means, methods and development algorithms. In the third chapter, a system for sharing books was developed and tested. The conclusions summarize the information presented in the previous parts. The development results demonstrate the possibilities of a social network for sharing books.

Keywords: social networks, Java, Express.js, React.

ЗМІСТ

ВСТУП	7
РОЗДІЛ 1 АНАЛІЗ СОЦМЕРЕЖ ДЛЯ ОБМІНУ КНИГАМИ ТА ПОСТАНОВКА ЗАВДАНЬ НА КВАЛІФІКАЦІЙНУ РОБОТУ	8
1.1 Аналіз сучасного стану проблеми	8
1.2 Постановка завдання на кваліфікаційну роботу бакалавра	14
Висновки до розділу 1	14
РОЗДІЛ 2 СПЕЦИФІКАЦІЯ ВИМОГ ДО РОЗРОБЛЮВАНОЇ СИСТЕМИ	16
2.1 Аналіз, визначення вимог до розроблюваного програмного забезпечення та проектування програмного забезпечення.....	16
2.2 Вибір засобів, методів і алгоритмів вирішення поставленого завдання....	18
Висновки до розділу 2	23
РОЗДІЛ 3 РОЗРОБКА СИСТЕМИ ДЛЯ ОБМІНУ КНИГАМИ	25
3.1 Практична реалізація об'єкта проектування	25
3.2 Тестування та налагодження інформаційно-комп'ютерної системи	27
3.3 Розробка бази даних	30
3.4 Захист інформаційно-комп'ютерної системи	32
Висновки до розділу 3	39
ВИСНОВКИ.....	40
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ.....	42
ДОДАТКИ.....	44

ВСТУП

Актуальність теми. Соціальні мережі, присвячені книгам, стають дедалі популярнішими. Вони дозволяють користувачам обмінюватися рекомендаціями, обговорювати та відстежувати новинки серед бестселерів.

Мета цих соціальних мереж – допомагати користувачам відкривати цікаві книги для читання, рекомендувавши видання на основі їхніх прочитаних книг або орієнтуючись на те, що читають їхні друзі.

Метою кваліфікаційної роботи є створення соцмережі обміну книгами «Book Find LNTU» з використанням Java, React та Express.js

Об'єктом дослідження є процес розробки соцмережі для обміну книгами, що реалізується з використанням технологій, таких як Java, React та Express.js.

Предметом дослідження є методи, технології та процеси, які використовуються для розробки соціальної мережі обміну книгами.

Для досягнення мети кваліфікаційної роботи, необхідно виконати наступні завдання:

- провести дослідження та аналіз поточного стану проблеми в обраній галузі;
- виявити та формалізувати вимоги до об'єкту проектування;
- оцінити та обрати оптимальні засоби, методи та алгоритми для розробки об'єкту проектування;
- розробити власне рішення у вигляді соціальної мережі для обміну книгами;
- спроектувати та реалізувати базу даних, необхідну для функціонування системи;
- розробити та впровадити заходи для забезпечення безпеки та захисту даних в системі.

РОЗДІЛ 1

АНАЛІЗ СОЦМЕРЕЖ ДЛЯ ОБМІНУ КНИГАМИ ТА ПОСТАНОВКА ЗАВДАНЬ НА КВАЛІФІКАЦІЙНУ РОБОТУ

1.1 Аналіз сучасного стану проблеми

Сучасний стан проблеми показує, що соціальна когнітивна теорія (SCT) представляє собою психологічний підхід, що підкреслює значення соціального середовища у формуванні поведінки, отриманні знань і самоконтролю. Теорія, створена Альбертом Бандурою, передбачає, що мотивація визначається особистими цілями індивіда та його очікуваннями щодо наслідків своїх дій. SCT широко досліджується у сфері управління знаннями, особливо у контексті поведінки студентів, включаючи вплив медіа, ефективне навчання, взаємодію та залучення в освітній процес. Дослідники також застосовували SCT для аналізу використання людьми онлайн-платформ, таких як YouTube, для обміну інформацією. Вони з'ясували, що такі чинники, як поінформованість і очікувані результати, впливають на частоту обміну знаннями. Проте, теми різноманітності та культурних бар'єрів, методологічні аспекти та довгострокові наслідки втручань залишаються важливими питаннями, які потребують подальшого вивчення в рамках досліджень SCT.

Дослідження Alexander et al. [2] підкреслив важливість спостереження за учасниками для розуміння мотивів, які стоять за рішеннями людей брати участь у обміні інформацією на сайтах соціальних мереж. Огляд літератури показує, що потрібно проводити більше досліджень щодо впливу соціальних медіа на обмін знаннями серед студентів університетів. Дослідження мало на меті дослідити поведінку кількох користувачів SNS, зокрема спілкування, виробництво знань та інформаційних матеріалів, а також обмін документами. Результати свідчать про позитивний зв'язок між обміном файлами, обміном інформацією та дискусіями офлайн і онлайн [3]. Попередні дослідження також підтверджують використання соціальних медіа для спільної практики та припускають, що репутація може бути фактором, який впливає не лише на прагнення, але й на когнітивні здібності та

результати винагороди [4]. Результати мета-аналізу показали, що електронні пристрої широко використовуються для навчання серед студентів.

У дослідженні [1] обговорювалося, як розвивався Інтернет, коли сайти соціальних мереж перетворювалися з технічного центру соціалізації на освітню, економічну та соціальну адміністративну установу. Дослідження [1] було зосереджено на важливості розуміння контексту під час дослідження використання соціальних медіа студентами коледжів у Малайзії, країні, що розвивається, де університети все ще перебувають у зародковому стані. У дослідженні було відзначено, що в ринкових умовах необхідно провести додаткові аналізи соціальних медіа. Зокрема, була відзначена значимість розуміння специфіки різних платформ соціальних мереж, таких як Facebook, Instagram, LinkedIn і Twitter. Дослідження також підкреслило недавній розвиток і визнання сайтів соціальних мереж або Web 2.0 і те, як використання соціальних медіа для роботи може вплинути на обмін інформацією між організацією. Дослідження припустило, що чотири характеристики соціальних медіа можуть чергувати підходи до участі у відкритих дебатах про знання. Деякі прийоми мають позитивний вплив, а інші – згубний. Технічні фактори можуть вплинути на обмін знаннями між студентами університету нафти та мінералів імені короля Фахда в Саудівській Аравії. Дослідження показало позитивний зв'язок між використанням соціальних медіа та обміном знаннями студентів. Також підкреслило обмеження оригінального дослідження, практичні результати та дослідницький потенціал для майбутніх досліджень [4].

Для педагогів і вчених участь студентів є загадкою. Ця робота пропонує орієнтовну теоретичну парадигму для активного навчання, яка згадує електроніку як особливо впливовий компонент. Були досліджені наслідки для майбутніх теоретичних досліджень щодо взаємозв'язку між участю студентів і цифровим навчанням, а також пропозиції для викладачів [4], або щоразу, коли дані обмінюються опосередковано (наприклад, за допомогою технологій), часто відбувається обмін знаннями [12]. Дехто стверджував, що підключені пристрої можуть покращити знання та навчання учнів [4]. Деякі стверджували, що, хоча

люди беруть участь у цифрових розмовних заходах, навчальні спільноти можуть сприяти обміну знаннями. Крім того, попередні дослідники виявили, що коли люди засновують організації, основною метою обміну інформацією є розширення можливостей інших шляхом розширення повноважень самих або отримання переваги. Він стверджував, що це було для отримання похвали. Наприклад, репутація може сприяти бажанню віртуальних груп ділитися інформацією [12].

Вплив мобільних додатків і соціальних мереж на різні аспекти освіти та навчання охоплює рівновагу між особистим задоволенням і практичними аспектами використання мобільних додатків, а також переваги соціальних медіа для обміну знаннями і активної участі студентів. Дослідження Akdim et al. [1], Rasheed et al. [10] і Naeem [9] підтримують ідею про те, що соціальні медіа можуть мати позитивний вплив на результати навчання, головним чином через віртуальні зв'язки з однолітками та експертами з предметів. Дослідження Almutairi [12]. На практиці більшість студентів мають доступ і використовують соціальні медіа, які можна легко включити в розробку навчальної програми. Соціальні медіа можуть підтримувати інтелектуальні форми студентів і заохочувати залучення через швидку взаємодію та співпрацю. Зростання соціальних медіа також призвело до широких наслідків збільшення використання, як з позитивними, так і з негативними наслідками. У дослідженні Manu et al. [11] досліджували, як учні сприймають соціальні мережі як цінний освітній інструмент. Це показало, наскільки відкрито соціальні медіа використовуються в освіті, розкривши, як і чому вони використовуються та мають теоретичне та дидактичне значення.

Вплив програм для телефону та соціальних медіа на різні аспекти освіти та навчання включає баланс між суб'єктивним задоволенням і утилітарними факторами в програмах для телефону та переваги соціальних мереж для обміну знаннями та залучення студентів. Багато науковців підтримують думку, що соціальні мережі можуть сприятливо впливати на академічні результати завдяки

віртуальним взаємодіям із ровесниками та фахівцями в різних галузях. Також підкреслюється значення усвідомлення ролі соціальних медіа в період пандемії.

Соціальні мережі та цифрові групи на книжкову тему стають все більш популярними, дозволяючи учасникам ділитися хорошими книгами, обговорювати чудові історії та переглядати останні бестселери. Для аналізу було обрано 4 чудових соціальних мережі на основі яких ми будемо брати за зразок для розробки власної соцмережі, зосереджених на книгах, для завзятих читачів.

Мета Goodreads (рис. 1.1) допомагати користувачам знаходити чудові книги, рекомендувавши нові видання на основі їхніх попередніх читань або вподобань їхніх друзів. Система також допомагає уникати книг, які можуть не відповідати їхнім смакам.

Goodreads дозволяє створювати список книг, оцінювати та рецензувати книги, а також дізнаватися, що читають друзі. Функції спільноти включають величезну різноманітність груп для участі, а також обговорення та функцію «Запитати автора».

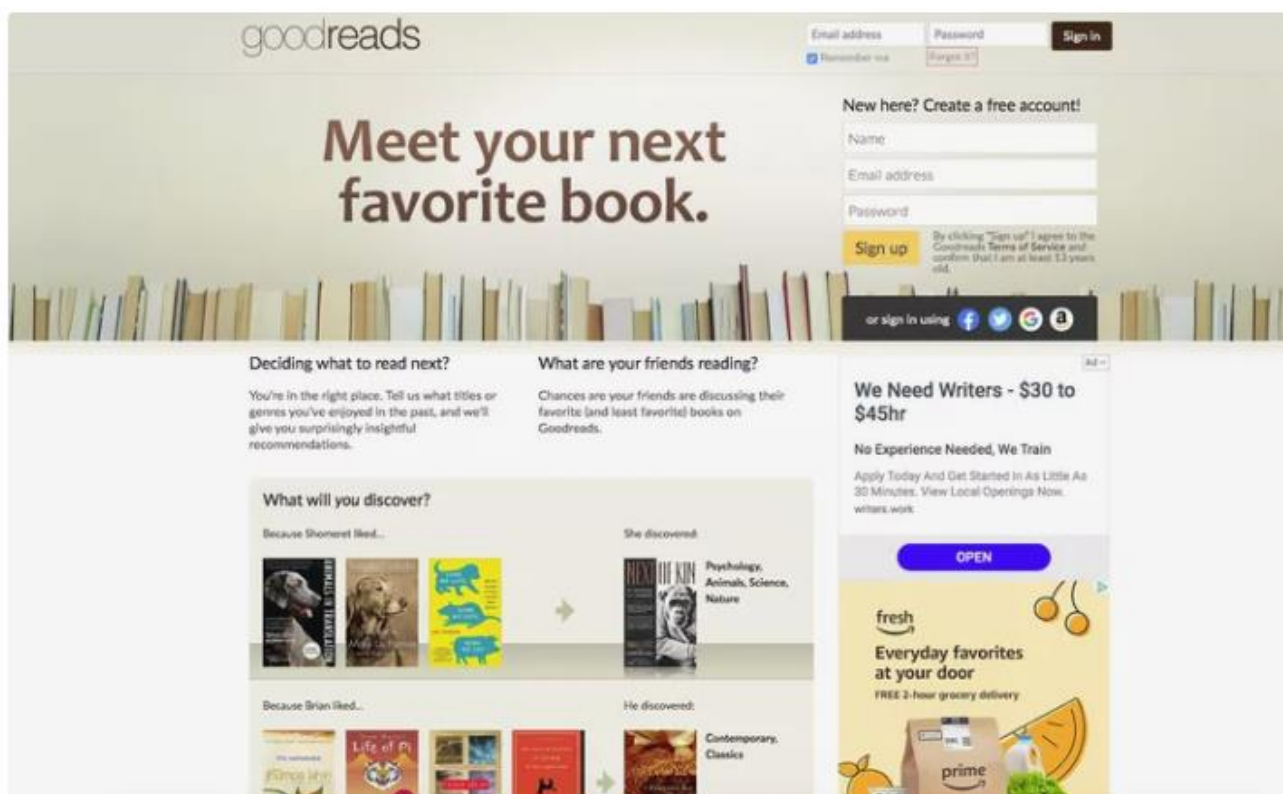


Рисунок 1.1 – Сайт Goodreads [7]

LibraryThing (рис. 1.2) є чудовим способом організувати свій список для читання. Книжкова платформа діє як простий у користуванні каталог у стилі бібліотеки зі спільнотою, що налічує понад два мільйони учасників. Каталогуйте книги безпосередньо з Amazon. Бібліотеки Конгресу та понад 1000 інших бібліотек. Ви навіть можете використовувати його для каталогізації своїх фільмів і музики, якщо хочете. Існує величезна кількість груп, до яких можна приєднатися, і дискусійних форумів, у яких можна взяти участь, з активною, зацікавленою спільнотою.

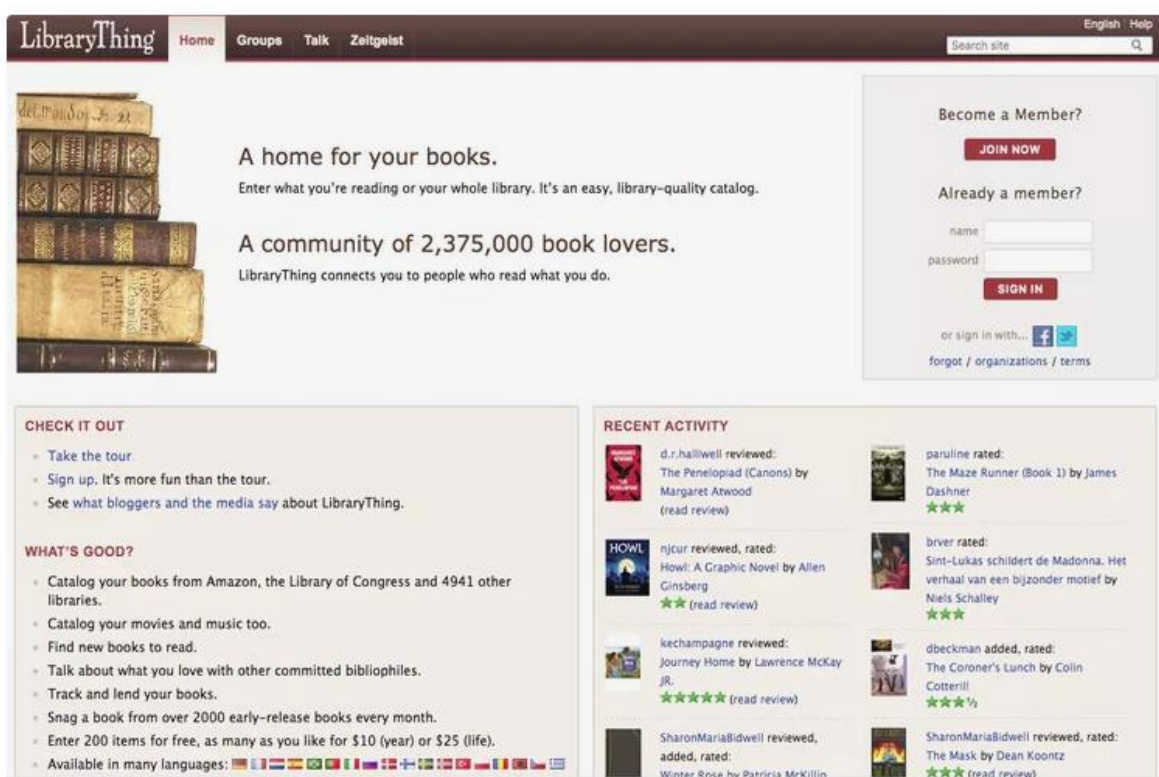


Рисунок 1.2 – LibraryThing [6]

BookCrossing (рис. 1.3) – це книжкова соціальна мережа, учасники якої випускають книги назад у публіку, залишаючи їх на лавках у парку, у спортзалі чи в школі. BookCrossing, яка є частиною соціальної мережі та частиною соціального експерименту, дозволяє вам брати участь у поверненні світу літератури, передаючи свої улюблені книги.

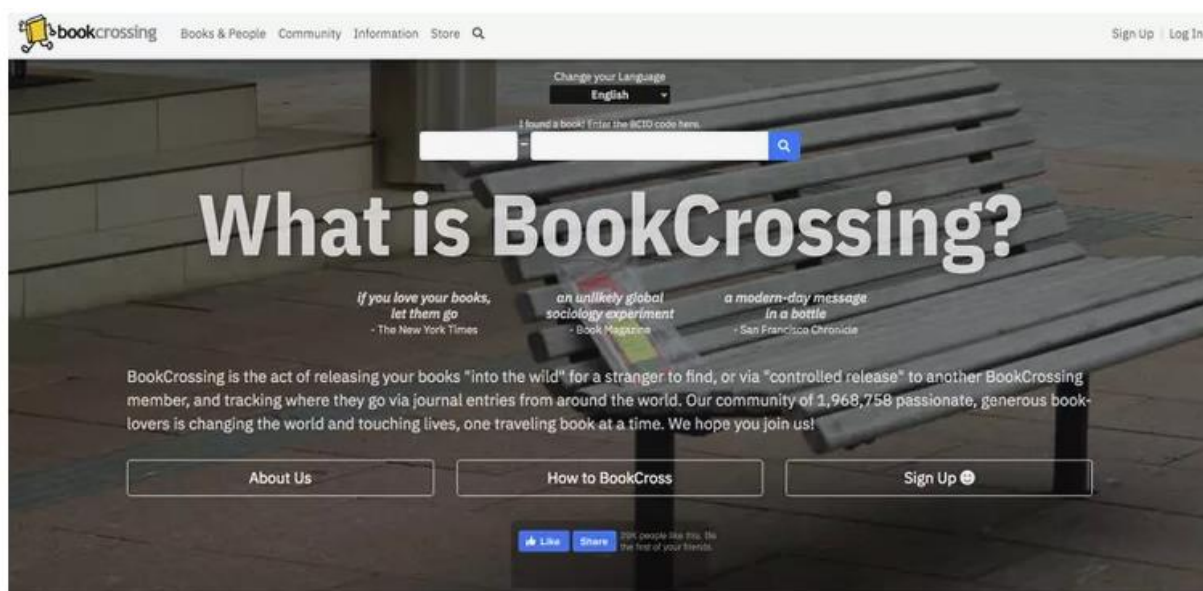


Рисунок 1.3 – Соціальна мережа BookCrossing [5]

Соціальні функції включають форуми спільнот, відгуки, рекомендації тощо.

Litsy (рис. 1.4) – це соціальний сайт і додаток, орієнтований на книги, щось середнє між Instagram і Goodreads. Учасники Litsy можуть переглядати, ділитися та обговорювати свої улюблені книги, а також створювати та організовувати списки для читання.

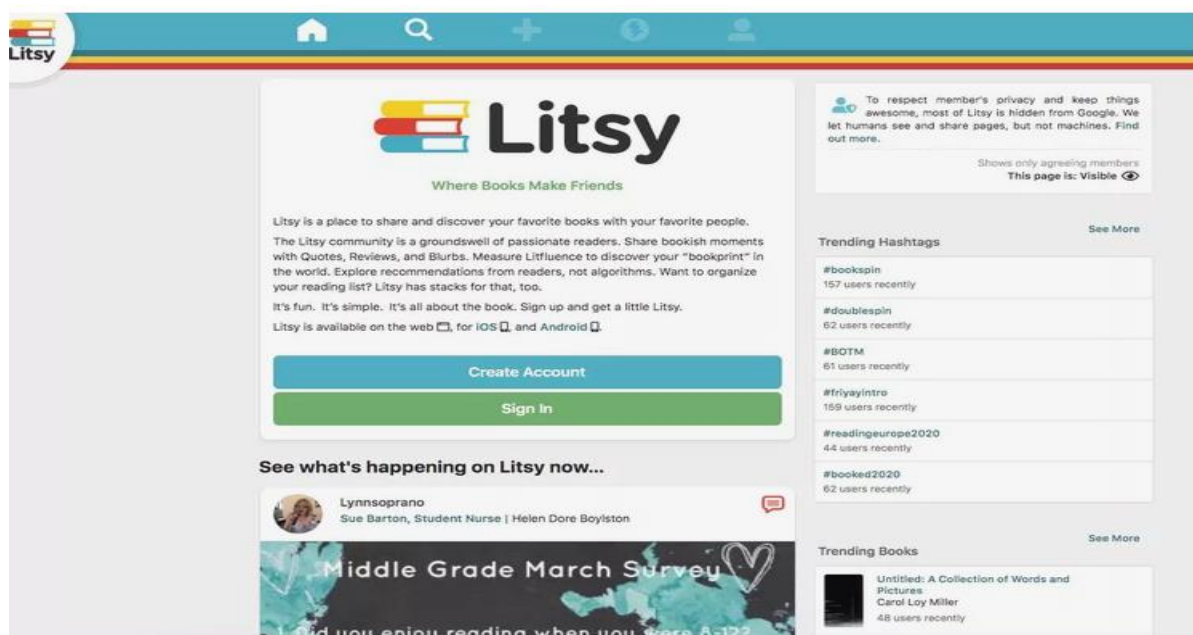


Рисунок 1.4 – Соціальна мережа Litsy [8]

Соціальні аспекти сайту включають глибокі та пристрасні обговорення книг, а також рекомендації. Оскільки все зосереджено на книгах, ви можете знайти всі публікації, що стосуються книги, здійснивши пошук за назвою.

1.2 Постановка завдання на кваліфікаційну роботу бакалавра

Після проведеного аналізу, для досягнення мети кваліфікаційної роботи бакалавра були поставлені наступні завдання:

- провести дослідження та аналіз поточного стану проблеми в обраній галузі;
- виявити та формалізувати вимоги до об'єкту проектування;
- оцінити та обрати оптимальні засоби, методи та алгоритми для розробки об'єкту проектування;
- розробити власне рішення у вигляді соціальної мережі для обміну книгами;
- спроектувати та реалізувати базу даних, необхідну для функціонування системи;
- розробити та впровадити заходи для забезпечення безпеки та захисту даних в системі;

Висновки до розділу 1

Було здійснено аналіз соціальних мереж мета яких – сприяти користувачам у відкритті чудових книг, рекомендувавши нові видання на основі їхніх попередніх читань або вподобань колег по роботі, а також допомагаючи уникати невідповідних книг. Соціальна когнітивна теорія (SCT) є психологічним підходом, який підкреслює значущість соціального середовища в процесі формування поведінки, набуття знань і саморегуляції. Згідно з теорією, розробленою Бандурою, мотивація визначається цілями індивіда та його очікуваннями щодо результатів. Вона досліджує різні аспекти студентської

поведінки, такі як вплив мас-медіа, мотивація до навчання, взаємодія та зацікавлення студентів. Дослідники також використовували SCT для вивчення того, як люди використовують онлайн-платформи, такі як YouTube, для обміну інформацією, виявивши, що такі фактори, як обізнаність і очікування результатів, впливають на те, як часто люди діляться своїми знаннями. Однак питання про різноманітність і культурні труднощі, методології та довгострокові наслідки втручань є важливими темами, які все ще потребують розгляду в дослідженнях SCT. Також для розробки аналогу було проаналізовано 4 сайти з подібною тематикою.

РОЗДІЛ 2

СПЕЦИФІКАЦІЯ ВИМОГ ДО РОЗРОБЛЮВАНОЇ СИСТЕМИ

2.1 Аналіз, визначення вимог до розроблюваного програмного забезпечення та проектування програмного забезпечення

Ідея полягає в тому, щоб створити систему обміну книгами, яка допоможе студентам отримати вживані книги легко і дешево. Використовуючи цю онлайн систему взаємодопомоги, люди можуть отримувати книги, які вони шукають, а також ділитися власними ресурсами з іншими.

Причиною вибору жанру підручника замість романів та художніх книг був фактор необхідності, який несе з собою підручник у порівнянні з іншими жанрами.

Метою цієї роботи є створення спільної онлайн-платформи для обміну книжками, які потрібні студентам. Вона також забезпечує платформу для обміну та обговорення переваг та недоліків книг, які вони використовують.

Цей вебсайт реалізований на Java, React та Express.js.

Існуюча система повністю автономна або залежить від інших соціальних платформ, які не призначені спеціально для цього завдання. Офлайн-процес є повільним і не впорядкованим. Довідники дуже важливі для будь-якого студента вищого навчального закладу, але ці книги часто дорогі або їх важко знайти. Існуюча система не має жодних можливостей для вичерпного пошуку таких книг в одному місці. Таким чином, студент повинен покладатися на свою соціальну мережу друзів, щоб знайти потрібні йому книги. Це може бути не для всіх можливим, а також виснажливим і трудомістким процесом.

З популяризацією інтернету, онлайн-покупки стали новою унікальною тенденцією. Від одягу до електроніки – все доступно в інтернеті. Продовжуючи цю тенденцію, відчувається потреба в системі онлайн-обміну книгами, оскільки вона може ще більше покращити існуючу систему.

Запропонована система (рис. 2.1), надасть користувачеві вичерпну базу даних доступних книг. Це буде першою зупинкою для будь-якого студента, щоб знайти будь-яку книгу.



Рисунок 2.1 – Діаграма USE CASE застосунку

Після того, як користувач знайде відповідного власника книги, система надасть йому можливість спілкуватися з ним в чаті. Основними цілями цієї системи є:

- вона зручна і проста у використанні як для читачів, так і для продавців;
- споживачі також можуть самостійно змінювати власну бібліотечну сторінку. Вони можуть додавати/видаляти книги;

- продавці можуть додавати нові книги у відповідні категорії;
- покупці можуть переглядати історію своїх книг та інформацію про них.

2.2 Вибір засобів, методів і алгоритмів вирішення поставленого завдання

Ключові елементи цієї системи (рис. 2.2) включають реєстрацію, вхід до системи, перегляд книг, замовлення і перегляд інформації про книги. Коли користувач визначає потрібного власника книги, система забезпечує платформу для безпосереднього спілкування. Запропонована система також надає платформу для всіх книголюбів, щоб взаємодіяти та обговорювати книги, для цього в системі передбачена система оцінювання та рецензування певних книг, щоб читачі могли ділитися своїми думками про прочитані ними книги.

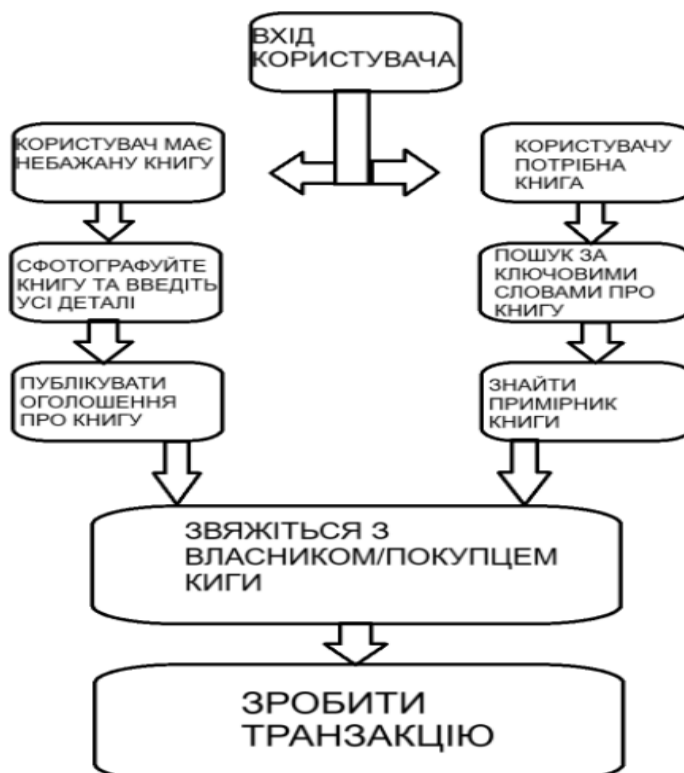


Рисунок 2.2 – Компоненти проектованої системи

Користувачі повинні зареєструватися, тільки тоді вони зможуть переглядати книги. Зареєстровані користувачі повинні спочатку увійти в

систему, щоб ділитися книгами. Після цього користувачі повинні вибрати необхідні книги, а потім зв'язатися з продавцем через функцію чату. Після цього вони можуть домовитися про ціну та характер угоди.

Архітектура системи (рис. 2.3) складається з: графічного інтерфейсу користувача (GUI), фронтенду та бекенду. Основними технологіями, що використовуються при розробці є React та Express.js. Це фреймворк веб-додатків з відкритим вихідним кодом, який підтримується компанією Google та спільнотою індивідуальних розробників і корпорацій для вирішення багатьох проблем, що виникають при розробці односторінкових додатків. Він має на меті спростити як розробку, так і тестування таких додатків, надаючи фреймворк для клієнтських архітектур «модель-вид-контролер» (MVC) та «модель-вид-модель» (MVVM), а також компоненти, які зазвичай використовуються в багатофункціональних інтернет-додатках.

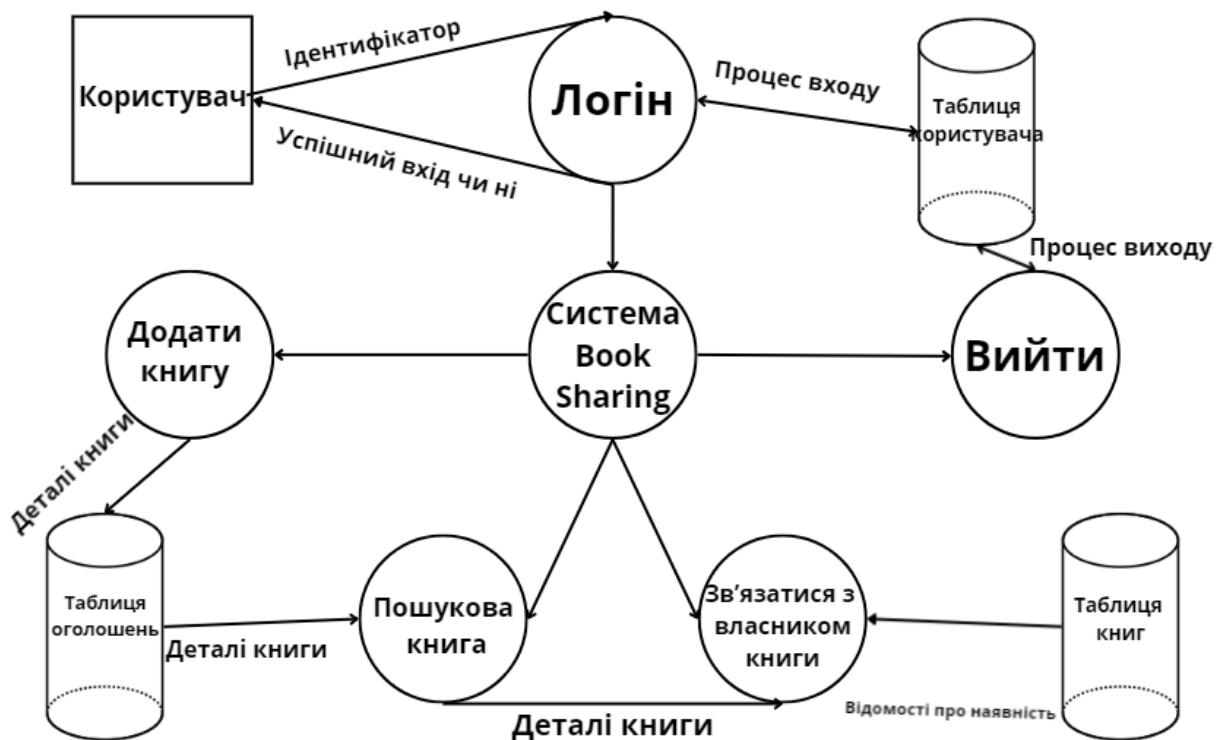


Рисунок 2.3 – Архітектура системи

Java – це об'єктно-орієнтована мова програмування. Об'єктно-орієнтоване програмування (ООП) – це парадигма програмування, яка організовує дизайн

програмного забезпечення навколо даних, а не функцій і логіки. Об'єкт – це поле даних із власним набором властивостей і поведінки. Наприклад, студентський клас може мати такі змінні, як ім'я, клас, номер списку, ім'я батька, оцінки та методи для зберігання та отримання цих значень.

У об'єктно-орієнтованому програмуванні (ООП) увага зосереджена на самих об'єктах, а не на логіці їх обробки. Це робить ООП ідеальним вибором для великих, складних проектів, які потребують регулярного оновлення і підтримки. Це охоплює програмне забезпечення для виробництва та проектування, а також мобільні програми; наприклад, ООП можна використовувати для моделювання виробничих систем. Основні концепції ООП:

- об'єкт;
- клас;
- спадщина;
- поліморфізм;
- абстракція;
- інкапсуляція.

Java використовується в мережевому та розподіленому середовищі. Наступні особливості роблять Java безпечною мовою:

- немає використання явних покажчиків;
- програма Java працює у віртуальній пісочниці (Віртуальна машина Java або JVM);
- завантажувач класів: середовище виконання Java (JRE) включає завантажувач класів, який використовується для динамічного завантаження класів Java у віртуальну машину Java. Він покращує безпеку, ізолюючи пакет для класів локальної файлової системи від класів, імпортованих із мережеских джерел;
- менеджер безпеки: визначає ресурси, до яких має доступ клас, наприклад, читання та запис на локальний диск.

ReactJS, створена Facebook – це бібліотека JavaScript, призначена для створення динамічних та інтерактивних програм, вдосконалення UI/UX для веб

і мобільних платформ. Працюючи як інтерфейсна бібліотека на основі компонентів із відкритим кодом, React присвячений дизайну інтерфейсу користувача та оптимізує налагодження коду за допомогою компонентно-орієнтованого підходу.

DOM (рис. 2.4) означає об'єктну модель документа. Це найважливіша частина мережі, оскільки вона ділиться на модулі та виконує код. Зазвичай JavaScript Frameworks оновлює весь DOM одночасно, що робить веб-програму повільною. Але React використовує віртуальний DOM, який є точною копією справжнього DOM. Щоразу, коли у веб-програмі відбувається зміна, спочатку оновлюється весь віртуальний DOM і виявляється різниця між реальним і віртуальним DOM.

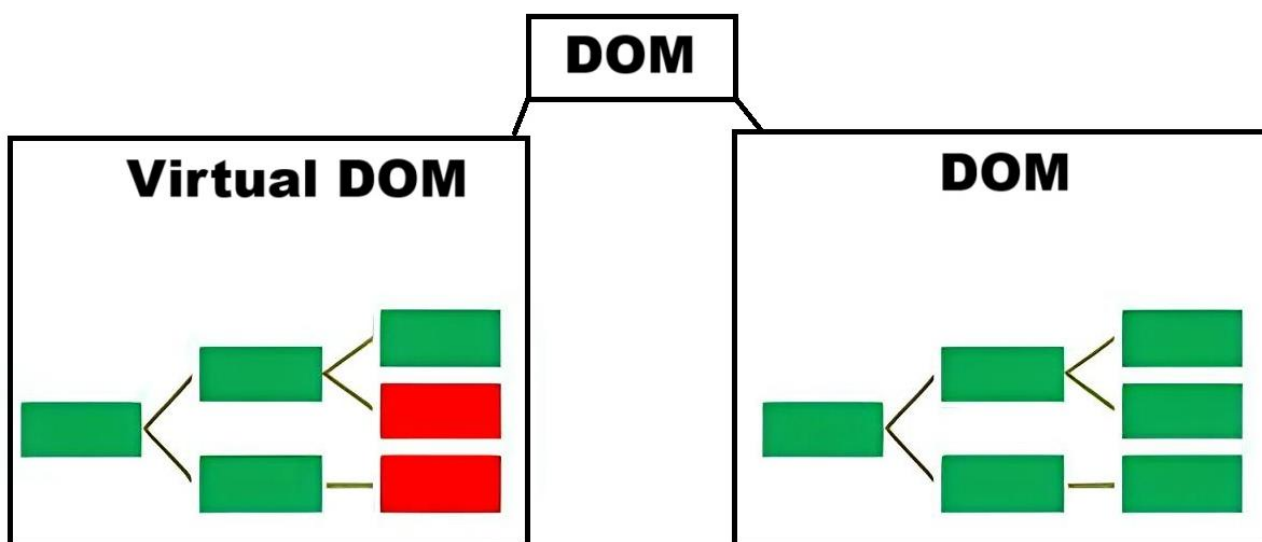


Рисунок 2.4 – Модель DOM

На наведеному вище рисунку 2.4, коли весь віртуальний DOM оновлено, відбувається зміна дочірніх компонентів. Отже, тепер DOM знаходить різницю та оновлює лише змінену частину.

React.js заснований на компонентах, що робить код багаторазовим, React.js використовує JSX, який є комбінацією HTML і JavaScript.

Це робить код легким для розуміння та налагодження та містить менше коду.

Express.js, або просто Express – це фреймворк веб-додатків для створення RESTful API за допомогою Node.js, який доступний для безкоштовного використання з відкритим кодом згідно з ліцензією MIT. Він призначений для розробки веб-додатків і API. Його назвали де-факто серверною структурою Node.js. TJ Holowaychuk, оригінальний автор, описав його як сервер, натхненний Сінатрою, маючи на увазі, що він відносно простий з багатьма функціями, доступними як плагіни.

Express разом із програмним забезпеченням бази даних MongoDB і інтерфейсом або бібліотекою JavaScript є базовим компонентом популярних стеків розробки, таких як стек MEAN, MERN або MEVN.

Express було створено для легкого створення API та веб-додатків, це заощаджує багато часу на кодування, майже вдвічі.

Ще одна причина використання Express полягає в тому, що він написаний на javascript і його дуже легко вивчити та реалізувати. Причиною створення експрес-фреймворку для node.js є:

- економія часу;
- швидко;
- ефективний;
- легко навчатися;
- асинхронний.

Швидка розробка на стороні сервера – функції node.js допомагають значно заощадити час.

Проміжне програмне забезпечення – це обробник запитів, який має доступ до циклу запит-відповідь програми.

Маршрутизація – це стосується того, як URL-адреси кінцевої точки програми відповідають на запити клієнта.

Шаблони – надає механізми створення шаблонів для створення динамічного вмісту на веб-сторінках шляхом створення шаблонів HTML на сервері.

Налагодження – Express полегшує роботу, оскільки визначає точну частину, де є помилки.

Переваги використання Express.js:

- експрес безперервний, і ми можемо налаштувати його;
- для обробки запитів ми можемо використовувати Middleware;
- одна мова використовується для інтерфейсу та серверної розробки;
- express швидко зв'язує його з такими базами даних, як MySQL, MongoDB тощо;
- express дозволяє динамічно відображати HTML-сторінки на основі передачі аргументів у шаблони.

Обмеження використання Express.js:

- іноді використання Express.js обмежується тим, що відсутній чіткий структурний підхід до організації коду, що може призвести до нерозуміння коду;
- виникає багато проблем із зворотніми викликами;
- повідомлення про помилки, які надходять, важко зрозуміти.

Express.js, або просто Express – це фреймворк для веб-додатків. Ми використовуємо його для створення RESTful API з Node.js. База даних була обрана SQL.

Висновки до розділу 2

Основними компонентами цієї системи є реєстрація, вхід, перегляд книг, замовлення та перегляд інформації про книгу. Система являє собою додаток, який надасть користувачам вичерпну базу даних доступних книг. Це буде першим місцем, куди звертатимуться студенти для пошуку будь-якої книги. Після того, як користувач знаходить потрібного власника книги, система пропонує власний чат для спілкування. Це не тільки полегшує взаємодію для користувачів, але й забезпечує захист їхньої приватності від можливих зловмисників.

У ООП основний акцент робиться на об'єктах, що використовуються розробниками, а не на створенні логіки для маніпулювання ними. Цей метод програмування ідеально підходить для великих і складних проектів, що потребують регулярного обслуговування та оновлень.

Для виконання роботи було обрано Java, React та Express.js спираючись на їхні переваги та на простоту у побудові застосунку.

РОЗДІЛ 3

РОЗРОБКА СИСТЕМИ ДЛЯ ОБМІНУ КНИГАМИ

3.1 Практична реалізація об'єкта проєктування

Переходимо до створення застосунку для обміну книгами між користувачами.

Ця система, надасть користувачам повну базу даних доступних книг. Це буде зручним для будь якого студента, щоб знайти книгу. Після того, як користувач знайде відповідного власника книги, система надасть йому можливість спілкуватися з ним в чаті. Основними цілями цієї системи є:

- вона зручна і проста у використанні як для читачів, так і для продавців;
- споживачі також можуть самостійно змінювати власну бібліотечну сторінку, вони можуть додавати/видаляти книги;
- продавці можуть додавати нові книги у відповідні категорії;
- покупці можуть переглядати історію своїх книг та інформацію про них.

Діаграма класів UML – це графічна нотація, яка використовується для побудови та візуалізації об'єктно-орієнтованих систем. Діаграма класів в уніфікованій мові моделювання (UML) – це тип статичної структурної діаграми, яка описує структуру системи, показуючи системні:

- класи;
- атрибути;
- операції (або методи);
- зв'язки між об'єктами.

Клас – це схема об'єкта. Об'єкти та класи йдуть рука об руку. Ми не можемо говорити про одне, не говорячи про інше. І вся суть об'єктно-орієнтованого проєктування полягає не в об'єктах, а в класах, тому що ми використовуємо класи для створення об'єктів. Отже, клас описує, чим буде об'єкт, але це не сам об'єкт.

Кожен об'єкт був створений з однакового набору креслень і тому містить однакові компоненти (властивості та методи). Стандартне значення полягає в

тому, що об'єкт є екземпляром класу та об'єкта. Об'єкти мають стани та поведінку (рис 3.1).

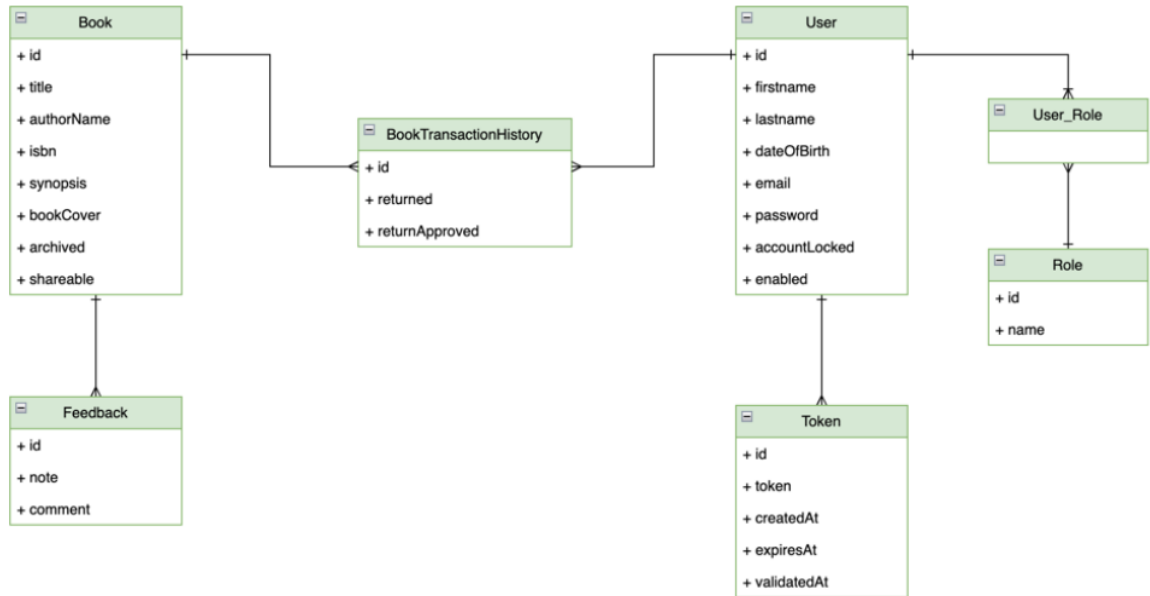


Рисунок 3.1 – Діаграма класів для розробленого застосунку

Діаграма варіантів використання є важливим інструментом у проектуванні системи, вона надає візуальне представлення, як користувачі взаємодіють із системою. Він служить схемою для розуміння функціональних вимог системи з точки зору користувача, допомагає в спілкуванні між зацікавленими сторонами та керує процесом розробки.

На фрагменті нижче (лістинг 3.1) ми розглянемо формування сторінки авторизації користувача.

Лістинг 3.1 – Формування сторінки авторизації

```

if [ -z "$MAVEN_SKIP_RC" ] ; then

    if [ -f /usr/local/etc/mavenrc ] ; then
        . /usr/local/etc/mavenrc
    fi

    if [ -f /etc/mavenrc ] ; then
        . /etc/mavenrc
    fi
fi
  
```

```

fi

if [ -f "$HOME/.mavenrc" ] ; then
. "$HOME/.mavenrc"
fi

fi

# OS specific support.  $var _must_ be set to either true or
false.
cygwin=false;
darwin=false;
mingw=false
case "$(uname)" in
  CYGWIN*) cygwin=true ;;

```

Кінець лістингу 3.1

Діаграма варіантів використання – це тип діаграми уніфікованої мови моделювання (UML), яка представляє взаємодію між учасниками і системою, що розглядається, для досягнення певних цілей. Він забезпечує загальне уявлення про функціональність системи, ілюструючи різні способи взаємодії користувачів з нею.

Межа системи – це візуальне представлення обсягу або меж системи, яку ви моделюєте. Він визначає, що всередині системи, а що зовні. Межа системи зазвичай представляється прямокутною рамкою, яка оточує всі варіанти використання системи.

Визначення обсягу: воно чітко окреслює межі системи, вказуючи, які компоненти є внутрішніми для системи, а які є зовнішніми акторами.

Окреслюючи сферу дії системи, діаграма може зосередитися на ілюстрації основних функціональних можливостей, які надає система, без зайвих деталей про зовнішні сутності.

3.2 Тестування та налагодження інформаційно-комп'ютерної системи

Робота починається з головної сторінки на якому розміщуються література розміщена іншими користувачами. Авторизований користувач має змогу так

само додати свою книгу, під книгами додати опис та коментувати публікації інших користувачів (рис. 3.2). Буде виконане функціональне тестування-перевірка працездатності всіх форм та функцій сайту.

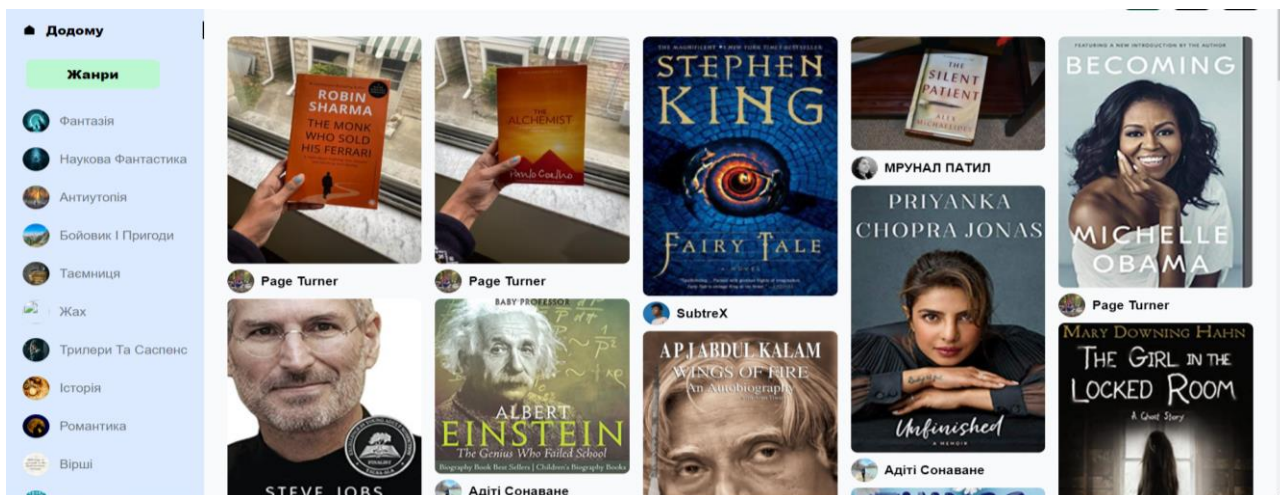


Рисунок 3.2 – Головна сторінка

По навігаційній панелі авторизований користувач може фільтрувати книги по жанрам, щоб швидше знайти бажану для прочитання книгу (рис. 3.3).



Рисунок 3.3 – Навігаційна панель жанрів

Також кожен користувач має свою особисту сторінку з інформацією про нього (рис. 3.4).

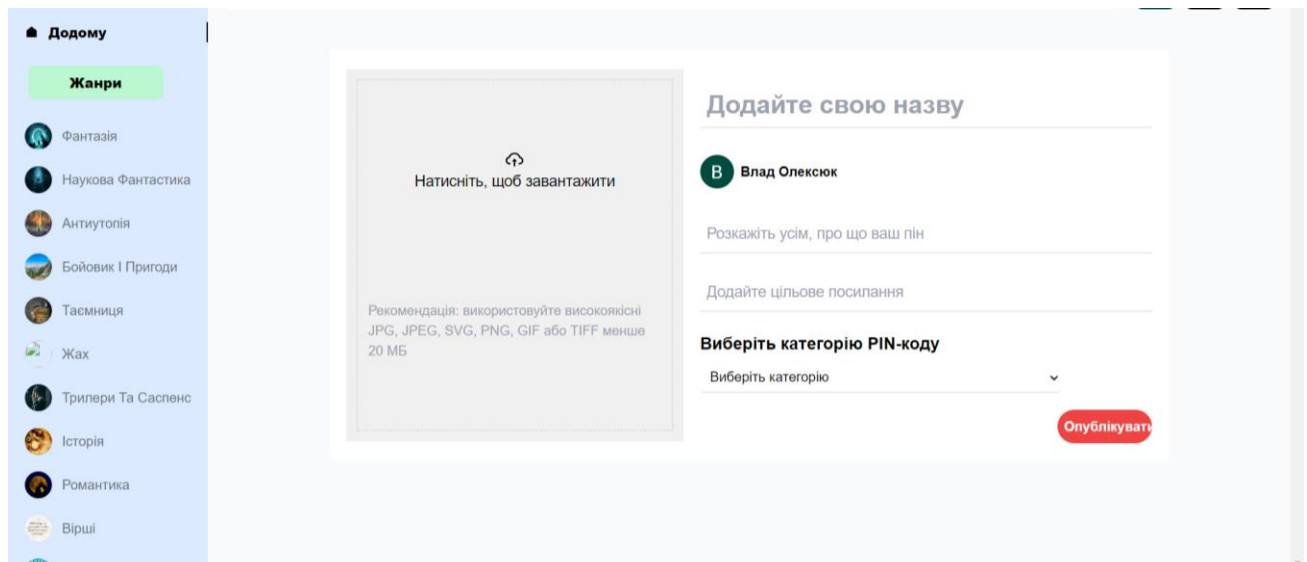


Рисунок 3.4 – Сторінка інформації про користувача

Додатково перевіримо функцію фільтрації книг по жанру (рис. 3.5).



Рисунок 3.5 – Фільтрація жанру наукова фантастика

Кожна розміщена книга має свій опис, який містить важливі деталі щодо її жанру, сюжету, головних персонажів, а також можливо про автора та його стиль написання. Опис слугує не лише інформаційною основою для читачів, але й сприяє їхньому зацікавленню та розумінню змісту книги, перед її прочитанням, завдяки ньому, читачі можуть зробити свідомий вибір та знайти саме ті книги, які відповідають їхнім вподобанням та інтересам (рис. 3.6).

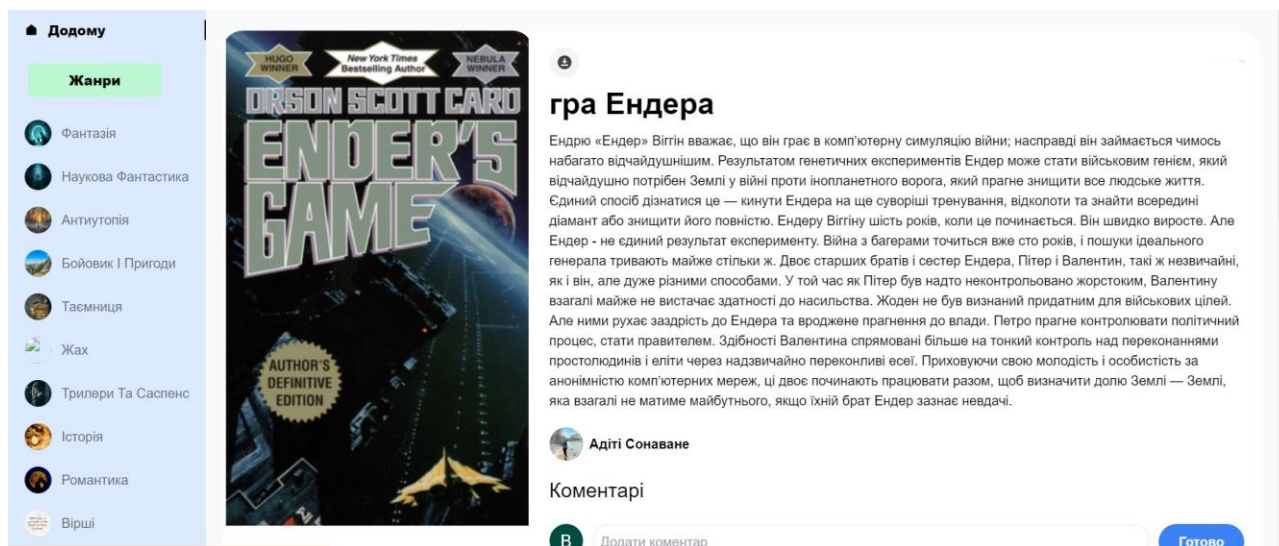


Рисунок 3.6 – Опис книги

Перевіривши всі функції сайту бачимо що додаток виконує всі функції які необхідні для виконання завдання.

3.3 Розробка бази даних

Програми для соціальних мереж є одними з найбільших споживачів даних, тому потреба в надійних базах даних завжди важлива в цій галузі. Програмам соціальних мереж потрібні бази даних для зберігання інформації про користувачів, розповсюдження вмісту та забезпечення взаємодії між онлайн-спільнотами.

Прикладами такої інформації можуть бути контакти клієнтів (ім'я, прізвище, номер телефону, адреса тощо).

Соціальна програма генерує величезні обсяги даних, наприклад, профілі, коментарі, зображення, відео, публікації наприклад, профілі користувачів з їхніми особистими даними, коментарі та відгуки, масу зображень і відео-файлів, а також різноманітні публікації у вигляді текстового контенту.

Соціальні програми потребують сповіщень і оновлень у режимі реального часу, а це означає, що вони мають бути швидкими й надійними на комп'ютерах і мобільних пристроях.

Бази даних «ключ-значення» та бази даних документів найкраще підходять для частих оновлень, оскільки вони дозволяють користувачам легко додавати, видаляти або редагувати свої дані.

Реляційна база даних (рис. 3.7) використовує таблиці наведені в додатку А та рядки для зберігання даних. Вона ідеально підходить для інформації, яка має стандартизовану послідовну структуру.

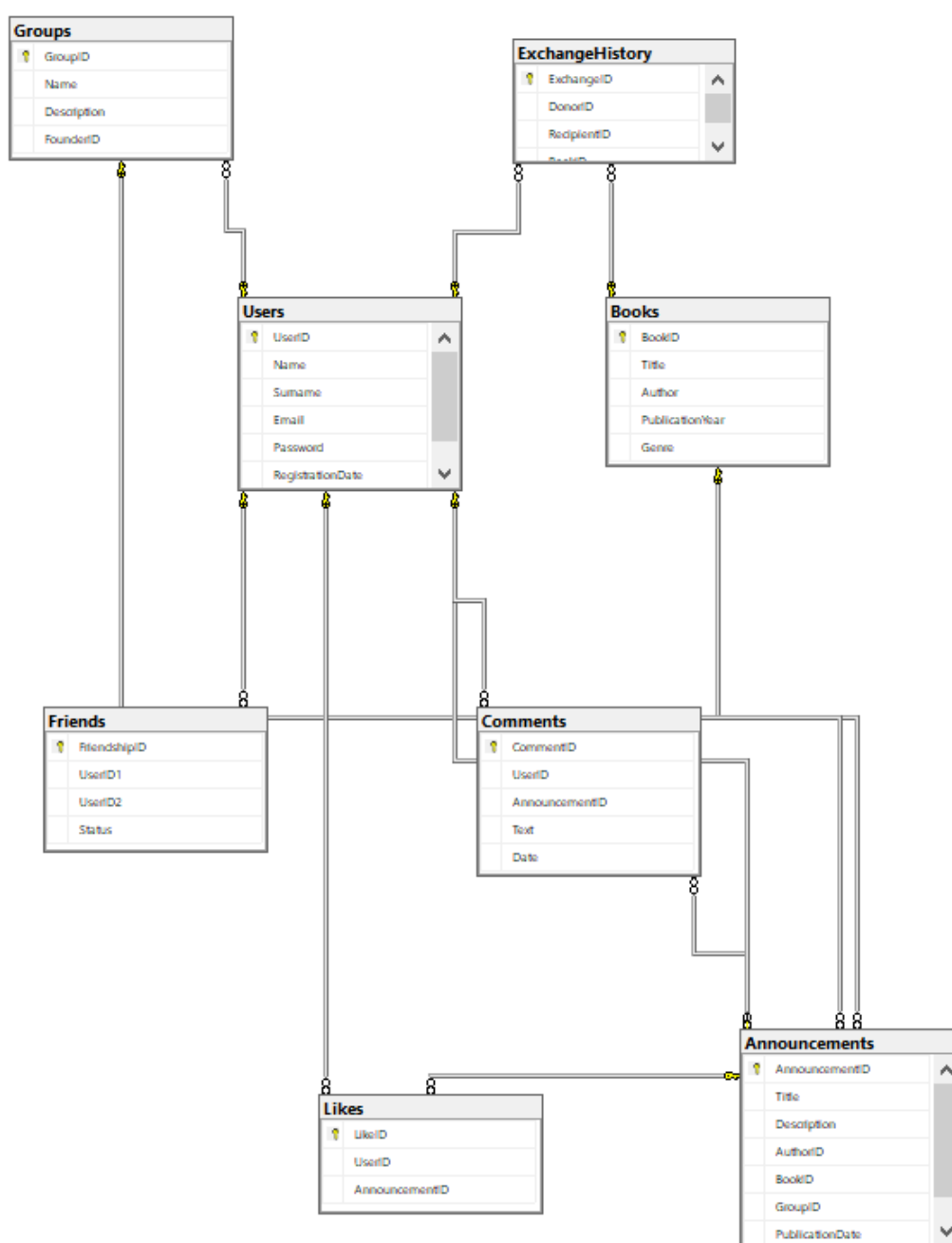


Рисунок 3.7 – Схема бази даних соцмережі

3.4 Захист інформаційно-комп'ютерної системи

Захист системи в операційній системі відноситься до механізмів, реалізованих операційною системою для забезпечення безпеки та цілісності системи. Захист системи передбачає методи запобігання неавторизованому доступу, неправильному використанню або модифікації операційної системи.

Існує кілька способів, якими операційна система може забезпечити захист системи:

- автентифікація користувача, операційна система вимагає від користувачів автентифікації перед доступом до системи. Для цього зазвичай використовуються імена користувачів і паролі;
- контроль доступу, операційна система використовує списки контролю доступу (ACL), щоб визначити, які користувачі або процеси мають дозвіл на доступ до певних ресурсів або виконання певних дій;
- шифрування, операційна система може використовувати шифрування для захисту конфіденційних даних і запобігання несанкціонованому доступу;
- брандмауер, це програма, яка відстежує та контролює вхідний і вихідний мережевий трафік на основі попередньо визначених правил безпеки;
- антивірусне програмне забезпечення, використовується для захисту системи від вірусів, шкідливих програм та іншого шкідливого програмного забезпечення;
- оновлення та виправлення системи, операційна система повинна постійно оновлюватися за допомогою останніх виправлень безпеки та оновлень, щоб запобігти використанню відомих вразливостей.

Впроваджуючи ці механізми захисту, операційна система може запобігти несанкціонованому доступу до системи, захистити конфіденційні дані та забезпечити загальну безпеку та цілісність системи.

Захист відноситься до механізму, який контролює доступ програм, процесів або користувачів до ресурсів, визначених комп'ютерною системою. Ми можемо взяти захист як помічник для операційної системи багатопрограмного

програмування, щоб багато користувачів могли безпечно спільно використовувати загальний логічний простір імен, наприклад каталог або файли.

Потреба в захисті:

- для запобігання доступу неавторизованих користувачів;
- щоб переконатися, що кожна активна програма або процес у системі використовує ресурси лише відповідно до зазначеної політики.

Автентифікація – це процес перевірки ідентичності користувачів, які отримують доступ до веб-додатків. Це досягається через запит користувачів надати особисті дані, такі як ім'я користувача та пароль, для підтвердження їхньої ідентичності.

Коли користувачі надають свої облікові дані, вони зазвичай автентифікуються сервером, який перевіряє облікові дані та надає або забороняє доступ до програми. Хоча автентифікація гарантує, що користувачі є тими, ким вони себе видають, авторизація визначає, що їм дозволено робити після автентифікації. Наприклад, користувач може бути автентифікований як адміністратор, але може не мати доступу до певних частин програми. Інший приклад: користувач може мати дозволи лише на доступ до своїх власних даних і їх змінення, але не на дані інших користувачів.

Автентифікація на основі сеансу була однією з найпоширеніших стратегій автентифікації та була стандартним способом автентифікації. Він передбачає використання файлів cookie для зберігання інформації про сеанс на стороні клієнта.

Коли користувач входить в систему, сервер створює сеанс і зберігає його в базі даних. Це означає, що підключення є станом. Потім сервер надсилає ідентифікатор сеансу клієнту у формі файлу cookie.

Коли користувач відправляє запит на сервер, маркер авторизації включається у заголовок з префіксом «Bearer». Потім сервер перевіряє цей маркер і, якщо він є дійсним, надає доступ до програми. У випадку недійсного маркера користувачеві буде відмовлено в доступі і, можливо, він буде перенаправлений на головну або сторінку входу.

Коли користувач виходить із системи, сеанс руйнується, а файл cookie видаляється.

Переваги:

- легко реалізувати;
- маленький розмір;
- тривалий термін служби.

Недоліки:

- не масштабується, оскільки сервер повинен зберігати інформацію про сеанс у базі даних. Це означає, що сервер повинен обробляти багато трафіку та запитів, що може бути дорогим і трудомістким;
- небезпечно, оскільки ідентифікатор сеансу зберігається в файлі cookie, який схильний до атак XSS і CSRF.

Автентифікація на основі маркерів дуже схожа на автентифікацію на основі сеансу, але замість того, щоб зберігати інформацію про сеанс у базі даних, сервер генерує унікальний маркер JWT, який надсилається клієнту та зберігається у файлі cookie. Це означає, що з'єднання не має стану і ми не маємо накладних витрат на зберігання інформації про сеанс у базі даних.

Маркер JWT – це рядок, який містить об'єкт JSON, закодований за допомогою Base64. Він підписується за допомогою секретного ключа, який використовується для перевірки маркера.

Переваги:

- масштабований, оскільки не зберігається в базі даних;
- безпечний, оскільки токен підписується за допомогою секретного ключа, який використовується для перевірки токена;
- ви можете зберегти інформацію про користувача в маркері, таку як роль користувача, дозволи тощо.

Недоліки:

- нелегко реалізувати, оскільки вам потрібно реалізувати багато логіки для перевірки маркера та обробки помилок. Але ви можете використовувати зовнішню бібліотеку, щоб зробити це простіше;

- коротший термін служби;
- накладні витрати на дані, оскільки зазвичай вони більші за ідентифікатор сеансу, і чим більше даних ви збережете в маркері, тим більшим буде маркер;
- все ще вразливий до атак XSS і CSRF, оскільки маркер зберігається в файлі cookie.

Використовуйте прапорець `HttpOnly`, який запобігає доступу JavaScript до файлів cookie. Це означає, що доступ до файлів cookie може отримати лише сервер.

Використовуйте прапор безпеки, який гарантує, що файли cookie надсилаються лише через HTTPS. Це означає, що сервер може отримати доступ до файлу cookie лише через HTTPS.

Установіть для прапорця `SameSite` значення `Strict`, що гарантує, що файли cookie надсилатимуться лише із запитом, що надходить із того самого сайту.

Якщо ви готові пройти зайву мильову, ви можете використовувати маркер оновлення та маркер доступу, це спосіб, якому я віддаю перевагу.

OAuth розшифровується як «відкрита авторизація», що є стандартом авторизації, який зазвичай використовується як спосіб для користувачів надавати веб-сайтам або програмам доступ до своєї інформації на інших веб-сайтах, але без надання їм паролів. Цей механізм надають такі компанії, як Facebook, Twitter і Google, щоб дозволити користувачам ділитися інформацією про свої облікові записи зі сторонніми програмами або веб-сайтами.

OAuth 1.0 проти OAuth 2.0 – це дві різні версії протоколу OAuth. Хоча обидві версії служать тій же основній меті, вони мають деякі відмінності у своїй реалізації та функціях.

Ось коротке порівняння між OAuth 1.0 і OAuth 2.0:

OAuth 1.0:

- використовує метод автентифікації на основі підпису, де кожен запит від клієнта містить підпис на основі спільного секрету та позначки часу. Це робить його безпечним і надійним для автентифікації;

- використовує як маркер, так і секрет для автентифікації клієнта та доступу до захищених ресурсів;
- складний потік, використовує тристоронній потік автентифікації, який передбачає отримання маркера запиту, авторизацію користувача, а потім обмін маркера запиту на маркер доступу.

OAuth 2.0:

- використовує простіший двосторонній потік автентифікації, який передбачає отримання маркера доступу безпосередньо з сервера авторизації;
- використовує метод автентифікації на основі маркерів, коли клієнт представляє маркер доступу в кожному запиті на доступ до захищених ресурсів;
- підтримує кілька типів дозволів, включаючи код авторизації, неявні, пароль і облікові дані клієнта, що забезпечує більшу гнучкість сценаріїв автентифікації;
- використовує області для визначення дозволів, запитуваних клієнтом, які можуть бути грубими або дрібнозернистими, залежно від реалізації;
- набув широкого поширення та широко використовується популярними веб-службами та API для автентифікації та авторизації.

Підсумовуючи, OAuth 1.0 – це складний потік на основі підпису з детальними дозволами, тоді як OAuth 2.0 – це простіший потік на основі маркерів із підтримкою кількох типів дозволів, потоків і областей. OAuth 2.0 ширше поширений, але вибір між OAuth 1.0 і OAuth 2.0 залежить від конкретних вимог програми та бажаного рівня безпеки та складності.

Один із найбільш популярних та широко використовуваних механізмів в рамках OAuth 2.0 – це потік коду авторизації. Коли клієнтська програма потребує доступу до захищених ресурсів, вона ініціює запит на авторизацію до сервера авторизації. Цей процес передбачає взаємодію з кількома кроками, починаючи з ініціації запиту та закінчуючи отриманням необхідного доступу. Ось покроковий огляд процесу коду авторизації:

- користувач ініціює процес авторизації: користувач натискає кнопку «Увійти за допомогою (клієнтської програми)» або виконує іншу дію, щоб ініціювати процес авторизації в клієнтській програмі;
- клієнт перенаправляє користувача на сервер авторизації: клієнтська програма перенаправляє користувача на сервер авторизації із запитом, який містить ідентифікатор клієнта, URI перенаправлення та бажану область доступу;
- автентифікація користувача на сервері авторизації: користувачеві пропонується пройти автентифікацію на сервері авторизації, надавши свої облікові дані (наприклад, ім'я користувача та пароль), щоб підтвердити свою особу;
- користувач надає авторизацію: після успішної автентифікації користувачеві відкривається екран згоди, де він може переглянути та надати або відхилити запит клієнта на доступ до своїх захищених ресурсів;
- сервер авторизації перенаправляє користувача назад до клієнта: якщо користувач надає авторизацію, сервер авторизації перенаправляє користувача назад до URI перенаправлення клієнтської програми з кодом авторизації, включеним у URL-адресу;
- клієнт обмінюється кодом авторизації на маркер доступу: клієнтська програма надсилає запит на сервер авторизації для обміну коду авторизації на маркер доступу. Цей запит містить код авторизації, ідентифікатор клієнта, секрет клієнта та URI перенаправлення;
- сервер авторизації перевіряє та видає маркер доступу: сервер авторизації перевіряє код авторизації, ідентифікатор клієнта, секрет клієнта та URI перенаправлення. Якщо вони дійсні, сервер авторизації видає клієнту маркер доступу та, можливо, маркер оновлення;
- клієнт використовує маркер доступу для доступу до захищених ресурсів: потім клієнт може використовувати отриманий маркер доступу, щоб робити авторизовані запити до захищених ресурсів від імені користувача. Маркер доступу служить доказом авторизації.

Потік коду авторизації широко використовується у веб-додатках. Однак це також додає додаткової складності через кілька залучених кроків, що робить його більш придатним для програм на стороні сервера.

OpenID Connect (OIDC) – це рівень автентифікації на основі протоколу OAuth 2.0. Він додає додатковий маркер JWT, який називається ідентифікатором, до потоку коду авторизації. Ідентифікаційний маркер містить інформацію про користувача, наприклад його ім'я та адресу електронної пошти. Ця інформація може бути використана для персоналізації взаємодії з користувачем і встановлення особи користувача.

Security Assertion Markup Language (SAML) – це стандарт на основі XML для обміну даними автентифікації та авторизації між доменами безпеки. Він використовується компаніями, щоб дозволити співробітникам отримувати доступ до програм і послуг від кількох постачальників за допомогою єдиного набору облікових даних.

Веб-автентифікація (WebAuthn) – це новий стандарт W3C для автентифікації без пароля в Інтернеті. Це дозволяє користувачам автентифікуватися на веб-сайтах за допомогою біометричної автентифікації, як-от відбитків пальців або розпізнавання обличчя, або ключів безпеки, таких як Yubikeys.

Він працює за допомогою криптографічної системи з відкритим ключем для створення унікальної пари криптографічних ключів для кожного користувача. Приватний ключ зберігається на пристрої користувача, а відкритий ключ зашифрований і зберігається на сервері. Коли користувач намагається увійти, сервер надсилає виклик на пристрій користувача, який користувач потім підписує своїм закритим ключем. Потім сервер перевіряє підпис за допомогою відкритого ключа, і якщо він дійсний, користувач аутентифікується.

Для складних програм із кількома користувачами я б рекомендував використовувати OIDC. Він більш безпечний і широко поширений. Лише для веб-додатків я б рекомендував використовувати автентифікацію на основі

маркерів із застереженнями, зазначеними вище. SAML – найкращий вибір, якщо ви створюєте корпоративну програму.

WebAuthn – це ще один чудовий варіант для веб-додатків, але він все ще знаходиться на ранніх стадіях і ще не широко прийнятий, і це вводить одну нову проблему – що, якщо користувач втратить автентифікатор, то може втратити доступ до свого облікового запису, що створює серйозні незручності та потенційні ризики. Існують інші способи автентифікації користувачів, як-от 2-факторна автентифікація, одноразові паролі тощо, але вони зазвичай використовуються в поєднанні з одним із методів, згаданих вище.

Висновки до розділу 3

Під час виконання розділу було розроблено та протестовано спеціальну соціальну мережу, призначену для обміну книгами між студентами та іншими користувачами, які мають доступ до цієї платформи.

Було розроблено можливість додавати книги, різними користувачами, коментувати та додавати в друзі.

ВИСНОВКИ

Зростає популярність книжкових тематичних груп у соціальних мережах, де учасники можуть ділитися улюбленими книгами, обговорювати чудові сюжети та переглядати останні бестселери.

Мета цих соціальних мереж – допомагати користувачам знайти цікаві книги, рекомендуючи нові видання на основі прочитаних раніше або вподобань їхніх знайомих з соціальних мереж, а також сприяючи уникненню книг, які можуть їм не сподобатися.

Основні аспекти цієї системи включають процеси реєстрації, входу, перегляду книжкового каталогу, замовлення книг та перегляд інформації про кожну книгу. Пропонована система, надасть користувачам детальну базу даних доступних книг. Для студентів це буде основним місцем пошуку будь-якої книги. Після того, як користувач знайде відповідного власника книги, система надає власну систему чату для взаємодії, що не тільки зручно для користувачів, але й допомагає захистити їх особистість від осіб зі зловмисними намірами.

В ході виконання роботи було розроблено та протестовано соціальну мережу для обміну книгами між студентами та людьми які мають доступ до цього ресурсу. Було розроблено можливість додавати книги, різними користувачами, коментувати та додавати в друзі.

В результаті роботи були виконані такі задачі:

- було проведено дослідження та аналіз ринку існуючих рішень для обміну книгами. Зокрема розглянуті такі соціальні мережі та цифрові групи на книжкову тему як: Goodreads, LibraryThing, BookCrossing та Litsy. Виявлено основні недоліки та можливості для покращення у доступних платформах, проаналізовано потреби та очікування користувачів, що дозволило визначити ключові функції, які мають бути реалізовані в новій соціальній мережі;

- виявлено та формалізовано вимоги до об'єкту проектування. Основні вимоги: реєстрація та автентифікація користувачів, можливість створення та редагування особистих профілів, додавання та пошук книг, обмін книгами,

оцінювання та рецензії, інтуїтивно зрозумілий інтерфейс. Технічні вимоги: продуктивність, масштабованість, безпека;

- оцінено та обрано оптимальні засоби, методи та алгоритми для розробки, виявлені основні функціональні вимоги до системи, зокрема, необхідність у реєстрації користувачів, управлінні бібліотекою книг, обміні книгами, забезпеченні безпеки та продуктивності, проведено порівняльний аналіз різних технологій, що дозволило обрати Java для бекенду, React для фронтенду та Express.js для створення RESTful API;

- було розроблено власне рішення соцмережі для обміну книгами, що включає серверну та клієнтську частину; базу даних, використовуючи Java та Express.js було створено серверну частину, за допомогою React було розроблено зрозумілий інтерфейс, спроектовано та реалізовано базу даних та впроваджено заходи для забезпечення безпеки системи;

- було спроектовано та реалізовано базу даних, необхідну для функціонування системи. Спроектowana БД підтримує всі необхідні функції системи, є масштабованою та забезпечує ефективне зберігання та доступ до інформації про користувачів, книги та операції обміну, для реалізації бази даних було використано технологію SQL, що забезпечує швидку та надійну роботу системи;

- розроблено та впроваджено заходи для забезпечення безпеки та захисту даних в системі, такі як, механізми шифрування даних, автентифікації та авторизації користувачів для захисту персональної інформації, реалізовані заходи для захисту системи від потенційних атак, таких як SQL – ін'єкції та XSS.

Виконання цих завдань дозволило створити функціональну, безпечну та масштабовану соціальну мережу обміну книгами «Book Find LNTU», яка відповідає сучасним вимогам та потребам користувачів, сприяє обміну знаннями та покращенню освітнього процесу.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Akdim K., Casalo L. V., Flavián C. The role of utilitarian and hedonic aspects in the continuance intention to use social mobile apps. *Journal of Retailing and Consumer Services*. 2022. Vol. 66. P. 102888. URL: <https://doi.org/10.1016/j.jretconser.2021.102888> (дата звернення: 15.03.2024).

2. Alexander D. S., Cao C., Alfonso M. L. Examining Whether the Social Cognitive Theory Concepts Predict Childhood Obesity Prevention Outcome Expectations. *International Quarterly of Community Health Education*. 2020. Vol. 41, no. 2. P. 143–151. URL: <https://doi.org/10.1177/0272684x20915383> (дата звернення: 22.03.2024).

3. Are Veterinary Students Using Technologies and Online Learning Resources for Didactic Training? A Mini-Meta Analysis / E. Muca et al. *Education Sciences*. 2022. Vol. 12, no. 8. P. 573. URL: <https://doi.org/10.3390/educsci12080573> (дата звернення: 18.03.2024).

4. Bond M., Bedenlier S. Facilitating Student Engagement Through Educational Technology: Towards a Conceptual Framework. *Journal of Interactive Media in Education*. 2019. Vol. 2019, no. 1. URL: <https://doi.org/10.5334/jime.528> (дата звернення: 25.03.2024).

5. BookCrossing. URL: <https://bookcrossing.com/> (дата звернення: 30.03.2024).

6. LibraryThing. URL: <https://www.librarything.com/> (дата звернення: 30.03.2024).

7. Goodreads. URL: <https://www.goodreads.com/> (дата звернення: 31.03.2024).

8. Litsy. URL: <https://www.litsy.com/web/home> (дата звернення: 31.03.2024).

9. Naeem M. Do social networking platforms promote service quality and purchase intention of customers of service-providing organizations?. *Journal of Management Development*. 2019. Vol. 38, no. 7. P. 561–581. URL: <https://doi.org/10.1108/jmd-11-2018-0327> (дата звернення: 04.04.2024).

10. Rasheed M. I., Murtza M. H. Dark Side of Competitive Psychological Climate: The Role of Workplace Envy and Trait Self-Esteem. *Academy of Management Proceedings*. 2023. Vol. 2023, no. 1. URL: <https://doi.org/10.5465/amproc.2023.13953abstract> (дата звернення: 11.04.2024).

11. Student engagement and social media in tertiary education: The perception and experience from the Ghanaian public university / B. D. Manu et al. *Social Sciences & Humanities Open*. 2021. Vol. 3, no. 1. P. 100100. URL: <https://doi.org/10.1016/j.ssaho.2020.100100> (дата звернення: 22.04.2024).

12. The Value of Social Media Use in Improving Nursing Students' Engagement: A Systematic Review / M. Almutairi et al. *Nurse Education in Practice*. 2022. P. 103455. URL: <https://doi.org/10.1016/j.nepr.2022.103455> (дата звернення: 01.05.2024).

ДОДАТКИ

Додаток А

Код створення таблиць (користувачів, книг, груп, оголошень, історії обміну,
друзів, коментарів, вподобань)

```
CREATE DATABASE Bool_SharingSocial  
GO
```

```
USE Bool_SharingSocial
```

```
-- Створення таблиці Користувачів
```

```
CREATE TABLE Users (  
    UserID INT PRIMARY KEY,  
    Name NVARCHAR(50),  
    Surname NVARCHAR(50),  
    Email NVARCHAR(100),  
    Password NVARCHAR(100),  
    RegistrationDate DATE  
);
```

```
-- Створення таблиці Книг
```

```
CREATE TABLE Books (  
    BookID INT PRIMARY KEY,  
    Title NVARCHAR(100),  
    Author NVARCHAR(100),  
    PublicationYear INT,  
    Genre NVARCHAR(50)  
);
```

```
-- Створення таблиці Груп
```

```
CREATE TABLE Groups (  
    GroupID INT PRIMARY KEY,
```

```
Name NVARCHAR(100),  
Description NVARCHAR(255),  
FounderID INT,  
FOREIGN KEY (FounderID) REFERENCES Users(UserID)  
);
```

-- Створення таблиці Оголошень

```
CREATE TABLE Announcements (  
    AnnouncementID INT PRIMARY KEY,  
    Title NVARCHAR(100),  
    Description NVARCHAR(255),  
    AuthorID INT,  
    BookID INT,  
    GroupID INT,  
    PublicationDate DATE,  
    FOREIGN KEY (AuthorID) REFERENCES Users(UserID),  
    FOREIGN KEY (BookID) REFERENCES Books(BookID),  
    FOREIGN KEY (GroupID) REFERENCES Groups(GroupID)  
);
```

-- Створення таблиці Історії Обміну

```
CREATE TABLE ExchangeHistory (  
    ExchangeID INT PRIMARY KEY,  
    DonorID INT,  
    RecipientID INT,  
    BookID INT,  
    ExchangeDate DATE,  
    FOREIGN KEY (DonorID) REFERENCES Users(UserID),  
    FOREIGN KEY (RecipientID) REFERENCES Users(UserID),  
    FOREIGN KEY (BookID) REFERENCES Books(BookID)
```

```
);
```

```
-- Створення таблиці Друзів
```

```
CREATE TABLE Friends (
    FriendshipID INT PRIMARY KEY,
    UserID1 INT,
    UserID2 INT,
    Status NVARCHAR(50),
    FOREIGN KEY (UserID1) REFERENCES Users(UserID),
    FOREIGN KEY (UserID2) REFERENCES Users(UserID)
);
```

```
-- Створення таблиці Коментарів
```

```
CREATE TABLE Comments (

    CommentID INT PRIMARY KEY,
    UserID INT,
    AnnouncementID INT,
    Text NVARCHAR(255),
    Date DATE,
    FOREIGN KEY (UserID) REFERENCES Users(UserID),
    FOREIGN KEY (AnnouncementID) REFERENCES
Announcements(AnnouncementID)
);
```

```
-- Створення таблиці Лайків
```

```
CREATE TABLE Likes (
    LikeID INT PRIMARY KEY,
    UserID INT,
    AnnouncementID INT,
```

```
FOREIGN KEY (UserID) REFERENCES Users(UserID),  
FOREIGN KEY (AnnouncementID) REFERENCES  
Announcements(AnnouncementID) );
```