

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Національний університет «Запорізька політехніка»

Факультет комп'ютерних наук і технологій

(повне найменування інституту, факультету)

Кафедра програмних засобів

(повне найменування кафедри)

Пояснювальна записка

до дипломного проєкту (роботи)

бакалавр

(ступінь вищої освіти)

на тему РЕАЛІЗАЦІЯ ПРОГРАМНОГО ЗАСТОСУНКУ

ЕЛЕКТРОННОЇ БІБЛІОТЕКИ.

SOFTWARE IMPLEMENTATION OF THE

ELIBRARY APPLICATION

Виконав: студент(ка) 4 курсу, групи КНТ-119сп

Спеціальності 121 Інженерія програмного

забезпечення

(код і найменування спеціальності)

Освітня програма (спеціалізація)

Інженерія програмного забезпечення

Сокур Є.О.

(прізвище та ініціали)

Керівник Субботін С.О.

(прізвище та ініціали)

Рецензент Гофман Є.О.

(прізвище та ініціали)

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Національний університет «Запорізька політехніка»
(повне найменування закладу вищої освіти)

Інститут, факультет ФКНТ
Кафедра програмних засобів
Ступінь вищої освіти бакалавр
Спеціальність 121Інженерія програмного забезпечення
(код і найменування)
Освітня програма (спеціалізація) Інженерія програмного забезпечення
(назва освітньої програми (спеціалізації))

ЗАТВЕРДЖУЮ

Завідувач кафедри ПЗ, д.т.н, проф.
С.О. Субботін
“ ” 2022 року

З А В Д А Н Н Я

НА ДИПЛОМНИЙ ПРОЄКТ (РОБОТУ) СТУДЕНТА(КИ)

Сокур Євгеній Олександрович
(прізвище, ім'я, по батькові)

1. Тема проєкту (роботи) Реалізація програмного застосунку електронної бібліотеки. Software Implementation of the eLibrary Application

керівник проєкту (роботи) Субботін Сергій Олександрович, д.т.н., професор,
(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)

затверджені наказом закладу вищої освіти від “27” квітня 2022 року № 102

2. Строк подання студентом проєкту (роботи) 7 червня 2022 року
3. Вихідні дані до проєкту (роботи) рекомендована література, технічне завдання

4. Зміст розрахунково-пояснювальної записки (перелік питань, які потрібно розробити) 1. Аналіз предметної області. 2. Матеріали і методи. 3. Опис програми. 4. Експлуатація, тестування та експериментальне дослідження програми.

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

6. Консультанти розділів проєкту (роботи)

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	прийняв виконане завдання
1-4 Основна частина	Субботін С.О., завідувач кафедри		
Нормоконтроль	Белова А.В., асистент		

7. Дата видачі завдання “ 27 ”квітня 2022 року.

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів дипломного проєкту (роботи)	Строк виконання етапів проєкту (роботи)	Примітка
1	Постановка завдання роботи.	1 тиждень	Завдання, ТЗ
2	Аналіз предметної області.	1 тиждень	Розділ 1
3	Вибір мови програмування та інших технологій розробки	2 тиждень	Розділ 2
4	Розробка архітектури програми.	2 тиждень	Розділ 3
5	Розробка програми.	3-4 тижні	Розділи 3,4
6	Тестування та експериментальне дослідження програмного забезпечення.	5 тиждень	Розділ 4
7	Оформлення пояснювальної записки та документів до неї.	6 тиждень	Додатки
8	Нормоконтроль та рецензування.	7 тиждень	
9	Захист роботи.	8 тиждень	

Студент(ка)

(підпис) Сокур Є.О.
(прізвище та ініціали)

Керівник проєкту (роботи)

(підпис) Субботін С.О.
(прізвище та ініціали)

РЕФЕРАТ

Пояснювальна записка до дипломної кваліфікаційної роботи бакалавра:
94 с., 42 рис., 3 табл., 4 дод., 28 джерел.

ЕЛЕКТРОННА БІБЛІОТЕКА, ЧИТАННЯ, ВИДАННЯ, РОЗРОБКА,
ПРОЕКТУВАННЯ, ВЕБ-ЗАСТОСУНОК, МОВА ПРОГРАМУВАННЯ,
ПРОГРАМНЕ ЗАБЕЗПЕЧЕННЯ.

Об'єкт дослідження – процес реалізація програмного застосунку електронної бібліотеки.

Предмет дослідження – програмні засоби для реалізації програмного застосунку електронної бібліотеки.

Метою роботи є реалізація програмного застосунку електронної бібліотеки у вигляді кроссплатформного вебзастосунку.

Матеріали, методи та технічні засоби: мова програмування PHP, система контролю контенту Joomla.

Результати. Розроблено програмне забезпечення у вигляді кроссплатформного вебзастосунку для електронної бібліотеки.

Практична цінність роботи полягає у реалізації програмного застосунку електронної бібліотеки у вигляді кроссплатформного вебзастосунку.

Висновки. Реалізовано програмний застосунок електронної бібліотеки у вигляді кроссплатформного вебзастосунку.

Галузь використання – користувачі мережі Інтернет, письменники.

ABSTRACT

Explanatory note to the diploma qualifying work of the bachelor: 94 pages, 42 figures, 3 tables, 4 appendixes, 28 sources.

ELIBRARY, READING, PUBLISHING, DEVELOPMENT, DESIGN, WEB APPLICATION, PROGRAMMING LANGUAGE, SOFTWARE.

The object of research is the process of implementing an electronic library software application.

The subject of the research is software tools for implementing the Electronic Library software application.

The aim of this work is to programmatically implement the Electronic Library software application in the form of a cross-platform web application.

Materials, methods, and technical tools: PHP programming language, Joomla content management system.

Result I. developed software in the form of a cross-platform web application for the Electronic Library.

The practical value of the work lies in the implementation of the Electronic Library software application in the form of a cross-platform web application.

Conclusions. Implemented an electronic library software application in the form of a cross-platform web application.

Scope of use is Internet users, writers.

ЗМІСТ

	С.
Перелік скорочень та умовних познач.....	8
Вступ	9
1 Аналіз предметної області.....	12
1.1 Електронні видання	12
1.2 Організація та функціонування електронних бібліотек.....	14
1.3 Порівняння існуючих аналогів	15
1.4 Постановка завдання.....	21
1.5 Висновки за розділом 1	25
2 Матеріали і методи.....	27
2.1 Вибір мови програмування та інших технологій розробки	27
2.2 Вибір та обґрунтування CMS.....	30
2.3 Структурна схема розробки	34
2.4 Висновки за розділом 2	36
3 Основні рішення щодо реалізації компонентів системи	37
3.1 Проектування моделей системи	37
3.2 Моделювання системи з точки зору прецедентів	40
3.3 Фізичне проектування	42
3.4 Проектування інтерфейсу	46
3.5 Висновки за розділом 3	48
4 Експлуатація, тестування та експериментальне дослідження програми.....	49
4.1 Призначення й умови застосування програми.....	49
4.2 Умови виконання програми	50
4.3 Інструкція по експлуатації програми	51
4.4 Методика та результати тестування.....	53
Висновки	57
Перелік джерел посилання	58
Додаток А Технічне завдання	62
Додаток Б Опис програми	67

Додаток В Текст програми	73
Додаток Д Слайди презентації.....	87

ПЕРЕЛІК СКОРОЧЕНЬ ТА УМОВНИХ ПОЗНАК

AES – Advanced Encryption Standard;

CMS – система управління контентом;

SEO – пошукова оптимізація;

UI – інтерфейс користувача;

UML – Unified Modeling Language;

БД – база даних;

ОС – операційна система;

ПЗ – програмне забезпечення;

ПК – персональний комп'ютер;

СКБД – система керування базами даних.

ВСТУП

Актуальність теми роботи полягає в тому, що відмінною рисою сьогоденного етапу розвитку суспільства є те, що інформація існує як в традиційній друкованій, так і в електронній формі [1]-[3]. Більш того, сучасні інформаційні технології дозволили не тільки приступити до широкомасштабного перекладу накопиченої людством інформації в електронну форму, а й до створення великого числа нових інформаційних ресурсів відразу в електронному вигляді. Ця форма подання інформації крім значного прискорення комунікативних процесів дозволяє на якісно новому рівні організувати процеси виробництва, зберігання і поширення інформації [1]-[3].

Інформаційні методи і технології все глибше проникають практично в усі види діяльності. Інформатизація, конвергенція інформаційних, комунікаційних технологій і мультимедіа, перехід до застосування сучасних інформаційних систем у сфері науки і освіти забезпечують принципово новий рівень отримання та узагальнення знань, їх поширення і використання [1]-[3].

Забезпечення публічного (в тому числі віддаленого) доступу користувачів до електронних інформаційних ресурсів стало одним з першочергових завдань інформаційного обслуговування науки, освіти і культури [1]-[3]. У нинішніх умовах вже не вимагає доказів, що концепція інформаційного обслуговування, що базується тільки на основі друкованих носіїв застаріла, і їй на зміну прийшла інша, - заснована на електронному поданні найрізноманітнішої інформації, тиражованої в необмеженій кількості і миттєво доступною через глобальні мережі передачі даних незалежно від часу і місцезнаходження користувача [1]-[3].

У зв'язку з цим істотно змінюється роль і функції такого соціального інституту, як бібліотека – основного сховища і розповсюджувача інформації [1]-[3].

Загальноновизнано, що найбільш ефективно інформаційно – бібліотечне обслуговування для потреб науки, культури і освіти сьогодні досягається шляхом створення електронних бібліотек, не стільки складаються з колекцій електронних документів (як оцифрованих традиційних матеріалів, так і створених відразу в електронному вигляді), скільки представляють із себе систему, що реалізує уніфікований підхід до виробництва, зберігання і організації різноманітної інформації з метою пошуку, аналізу та доступу до неї з використанням глобальних комп'ютерних мереж [1]-[3].

Основна відмінність традиційної бібліотеки та електронної бібліотеки полягає в тому, що користувач звертається до сервісів останньої за необхідною інформацією, а не просто за відсиланнями до документа та/або самими документами [1]-[3]. Відмінною рисою електронної бібліотеки є можливість паралельного використання різних пошукових механізмів і засобів доступу до гетерогенних колекцій електронних даних [1]-[3]. Так як в якості відповіді на запит до електронної бібліотеки може бути представлений далеко не єдиний електронний документ (або його фрагмент) і зовсім не обов'язково в текстовій формі, необхідні такі інформаційні системи, які можуть забезпечити ефективний комплексний пошук і аналіз інформації в колекціях різнорідних об'єктів [1]-[3].

В силу цілого ряду причин періоду соціально-економічних перетворень рівень інформаційного забезпечення науки і освіти в Росії на один - два порядки нижче, ніж в розвинених країнах [1]-[3]. Внаслідок жорстких фінансових обмежень різко знизився рівень комплектування навіть провідних бібліотек та інформаційних центрів як вітчизняними, так і зарубіжними науково - технічними періодичними виданнями [1]-[3]. Очевидно, що в найближчі роки традиційними методами змінити ситуацію на краще не представляється можливим, тому стали необхідні нові нетрадиційні підходи для вирішення проблеми підвищення рівня інформаційного забезпечення вчених і фахівців на основі використання інформаційних, комунікаційних технологій і електронізації інформаційних фондів. Цим пояснюється та увага,

яку державні відомства проявили до діяльності зі створення та використання електронних бібліотек в Україні[1]-[3].

Мета дипломної кваліфікаційної роботи бакалавра дослідити особливості реалізації програмного застосунку електронної бібліотеки.

Завдання дослідження:

- характеристика загальних питань організації електронних ресурсів;
- огляд принципів взаємодії інтернет-ресурсів та процесу формування каталогів електронних видань;
- відстеження тенденцій розвитку організаційно-технологічних основ електронних бібліотек;
- аналіз концепції побудови електронних бібліотек та принципів організації інформаційних ресурсів у них;
- аналіз існуючих аналогічних рішень та принципів їх організації;
- акумулювання результатів аналізу та досліджень для реалізації програмного застосунку електронної бібліотеки, що якісно відрізняється від існуючих аналогів.

1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ

Перший розділ роботи буде присвячено ознайомленню із предметною областю електронних бібліотек. Буде досліджено принципи їх організації та функціонування. Додатковий аналіз існуючих аналогів буде проведено з метою виявлення переваг та недоліків для майбутнього формування низки функціональних та не функціональних вимог до майбутнього застосунку.

1.1 Електронні видання

Електронні видання виникли наприкінці 80-х років. Фірма «Columbia Tristar Home Video» (США) випустила першу серію електронних книг для використання – так, виник ринок «електронних книг» [4]-[9]. На той час у зв'язку з швидким поширенням персональних комп'ютерів (ПК) виробники електронних книг і пристроїв для їх читання стали орієнтуватися на стандарти CO-ROM і передбачали можливість підключення плеєрів до ПК [5]-[9].

У 1998 році, на американському ринку з'явилося нове покоління електронних книг: The Rocket eBook, SoftBook, Millenium Reader і EveryBook [5]-[9].

Безумовно перевагою електронної книги є компактність – електронна книга набагато місткіша звичайної книги і може вмістити не те, що інформацію багатотомної енциклопедії, а зміст книг цілої бібліотеки. Електронна книга скорочує час пошуку інформації, оскільки текст отримується у вигляді бітів через Інтернет, і будь-яка книга може бути доставлена миттєво[5]-[9]. Електронна книга забезпечує легкість анотування, дозволяючи додавати власні електронні зауваження, виділяти текст і вносити анотації, примітки, посилання [5]-[9].

Електронна книга економічно вигідніша для читача, ніж друковане видання, що пов'язано з відсутністю витрат на видання, мінімальною вартістю зберігання і доставки [5]-[9]. На практиці немає ніяких перешкод до того, щоб

класичну літературу та інші загальнодоступні тексти можна було завантажувати з електронних сховищ безкоштовно [5]-[9].

Серед електронних видань слід в першу чергу відзначити наукові та навчальні видання [5]-[9]. Як правило, електронні підручники та енциклопедії як аматорського, так і професійного характеру спочатку випускалися в основному на компакт-дисках, але потім все більш популярними стали аналогічні видання і в інтернет. Чим легше сам процес створення, тим більше число розробників починає виробляти подібні застосунки. Творцями електронних видань стають і книжкові видавництва (тобто власники інформації) і журналісти, що випускають електронні версії періодичних видань, і, нарешті, викладачі [5]-[9]. Проміжне положення займають групи, що об'єднують власників інформації та програмістів.

Характерні риси електронних видань, створених засобами прямого програмування: різноманітність стилів реалізації (колірна палітра, інтерфейс, структура електронного підручника, спосіб подачі матеріалу і т.д.); складність модифікації та супроводу; великі витрати часу і трудомісткість; відсутність апаратних обмежень, тобто можливість створення електронного підручника, орієнтованого на наявну в наявність технічну базу [5]-[9].

Гіпертекст – це спосіб нелінійної подачі текстового матеріалу, при якому в тексті є будь-яким чином виділені слова, що мають прив'язку до певних текстових фрагментів [5]-[9]. Таким чином, користувач не просто гортає по порядку сторінки тексту, він може відхилитися від лінійного опису за яким-небудь посиланням, тобто сам управляє процесом видачі інформації. У гіпермедіа системі в якості фрагментів можуть використовуватися зображення, а інформація може містити текст, графіку, відеофрагменти, звук [5]-[9].

Використання гіпертекстової технології задовольняє таким вимогам, що пред'являються до підручників, як структурованість, зручність в обігу. При необхідності такий підручник можна викласти на будь-якому сервері і його можна легко коригувати [5]-[9]. Але, як правило, їм властиві невдалий дизайн,

компонування, структура і т.д. в даний час існує безліч різних гіпертекстових форматів [5]-[9].

1.2 Організація та функціонування електронних бібліотек

Електронна бібліотека з одного боку виконує функції традиційної бібліотеки: надання інформації читачеві, з іншого-виконує роль, характерну для автоматизованих інформаційних бібліотечних систем [4], [10]-[12]. Організація і зберігання локальних і віддалених електронних ресурсів і доступу до них на основі комп'ютерних і телекомунікаційних технологій.

Останнім часом з'являються різні тлумачення терміну електронна бібліотека. Зазвичай же під електронною бібліотекою розуміється інформаційна система, що дозволяє надійно зберігати і ефективно використовувати різноманітні колекції електронних документів [4], [10]-[12].

Цілі створення електронних бібліотек:

- забезпечення збереження друкованого матеріалу [4], [10]-[12];
- створення умов для забезпечення більшої доступності друкованого матеріалу, розширення інформаційних послуг [4], [10]-[12];
- вирівнювання рівнів технологічного розвитку установ, що працюють на користь суспільству [4], [10]-[12].

Фундаментом, основою, що об'єднує традиційні та електронні бібліотеки, є принцип обслуговування користувачів [4], [10]-[12].

Функції електронної бібліотеки відрізняються від класичних бібліотечних. З формальної точки зору істотна частина електронних ресурсів – є копії друкованих версій і в цьому сенсі електронна колекція складається з копій, а не з оригіналів, першоджерел. Електронна бібліотека на сьогоднішній день вторинна по відношенню до традиційної класичної бібліотеки [4], [10]-[12].

Найчастіше структура електронної бібліотеки зводиться до:

- вимога до електронних ресурсів: їх повинно бути багато, потрібно перевершити якусь критичну масу, що б забезпечити привабливість фондів для користувача, як віддалених, так і локальних [4], [10]-[12];

- середовище передачі даних – Інтернет. Хороша якість зв'язку читачів з Інтернет буде характеризувати роботу бібліотеки, крім того Інтернет може виступати як розподілене сховище безлічі мережевих ресурсів [4], [10]-[12];

- добре підготовлений теоретично і професійно бібліотечний працівник, який знає не тільки бібліотечну справу, а й володіє навичками робота з комп'ютером, в середовищі Інтернет [4], [10]-[12].

Розвиток електронної бібліотеки йде за трьома напрямками:

- удосконалення існуючої системи електронних каталогів;
- формування фонду електронних ресурсів;
- розвиток мережевого доступу до ресурсів електронної бібліотеки на території закладу вищої освіти [4], [10]-[12].

Основа будь-якої електронної бібліотеки є електронний каталог і доступ до бібліографічних баз даних, бюлетені нових надходжень, інформація про поточні події в бібліотеці: конференціях, книжкових виставках. Створюється колекція електронних книг і статей-власне сама електронна бібліотека [4], [10]-[12].

1.3 Порівняння існуючих аналогів

На сьогоднішній день існує багато ресурсів, які розміщують у себе знані видання письменників класиків. Проте про ліцензійність та загальну законність такого підходу достеменно нічого не відомо. Саме тому під час аналізу існуючих аналогічних рішень будемо звертати увагу саме на ресурси, які представляються себе у якості саме бібліотечних платформ, а також мають вже певний перевірений статус.

1.3.1 Бібліотека української літератури «УкрЛіб»

Онлайн-бібліотека української літератури «УкрЛіб» була заснована в 2000 році [13], [14]. На сьогоднішній же день ресурс позиціонується, як чи найбільша в інтернеті електронна бібліотека української та світової літератури на українській мові. Використовуючи принцип розміщення творів виключно українською, засновники мають намір популяризувати українську мову, культуру та спадщину серед нових поколінь, що є найбільш активними користувачами мережі Інтернет [13], [14].

Сайт ресурсу простий та зрозумілий, а серед основних розділів виділяються [13], [14]:

- українська література;
- світова література;
- навчання;
- літературознавство;
- матеріали російською.

Цікавинками ресурсу є функція – «книга дня», яку ресурсавтоматично визначає та пропонує почитати у день відвідування, та розділ – «видатні діячі української літератури», де можна ознайомитись з біографією та найкращими творами визначних українських письменників [13], [14].

З творами можна ознайомитися безпосередньо на сайті, або завантажити в найбільш прийнятному форматі: DOCX, RTF, HTML, EPUB, MOBI, PDF [13], [14].

Головна сторінка ресурсу наведена рис. 1.1.

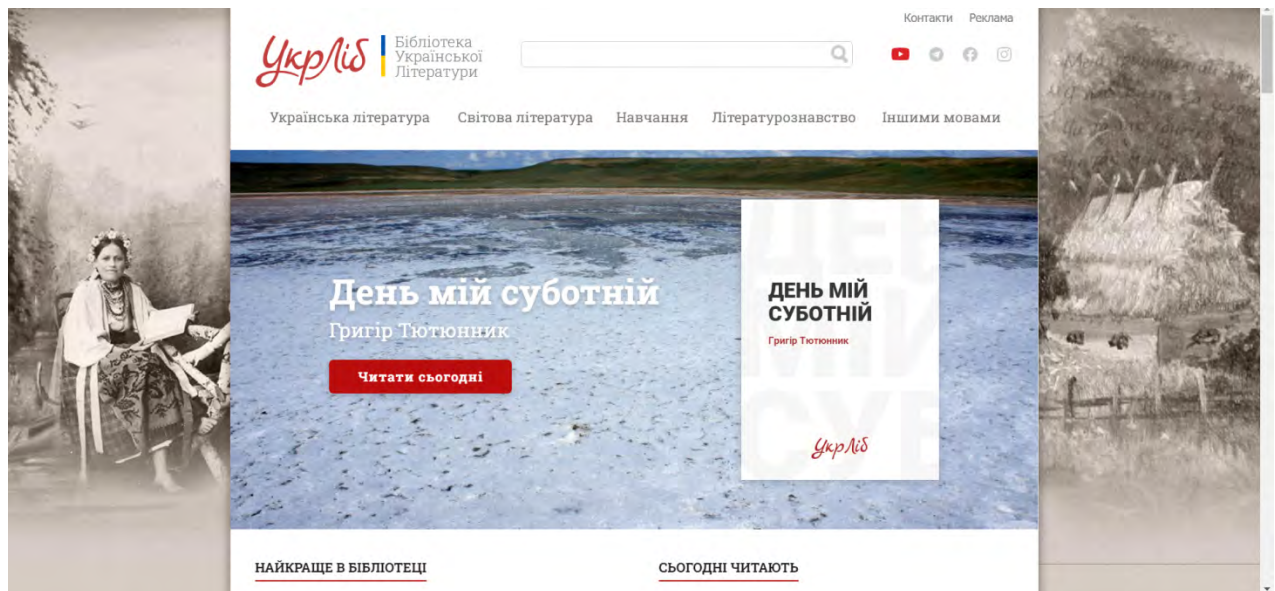


Рисунок 1.1 – Бібліотека української літератури «УкрЛіб» [14]

1.3.2 Українська електронна бібліотека «Libruk»

Платформа Libruk окрім розміщення українських та світових письменників класиків дозволяє авторам-початківцям розміщувати свої матеріали в розділі «Самвидав» [13], [15].

А загалом же платформа більш нагадує класичну бібліотеку з алфавітним каталогом авторів на головній сторінці. Проте, є цікавинка з виведенням письменників на мапі України, а для поглиблення знань з творчого шляху кожного автора можна ознайомитися із їх скороченими біографіями на платформі [13], [15].

З творами можна ознайомитися безпосередньо на сайті, або завантажити в найбільш прийнятному форматі: FB2, EPUB, MOBI, PDF або DJVU. MOBI було додано через популярність електронних видань Amazon Kindle [13], [15]. Сама ж платформа надає зручний інструментарій онлайн-читалки, що має низку налаштувань відображення тексту для людей із різними потребами, а приємним бонусом є – запам'ятовування останньої прочитаної сторінки [13], [15].

Головна сторінка сайту наведена рис. 1.2.

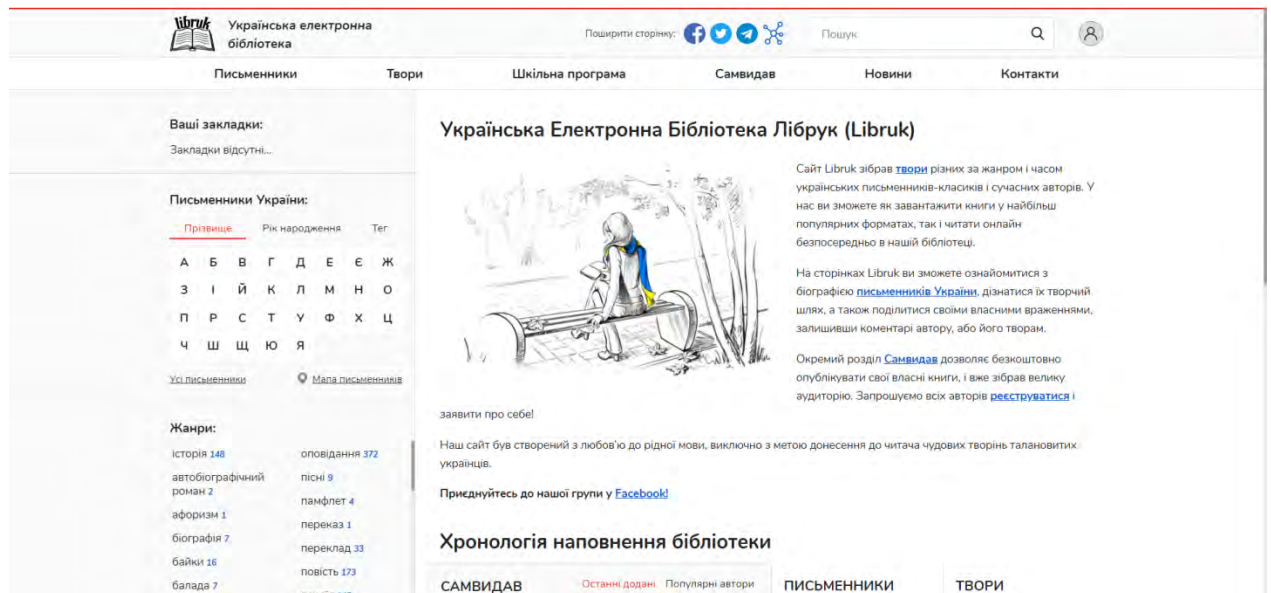


Рисунок 1.2 – Українська електронна бібліотека «Libruk» [15]

1.3.3 Сайт Національної бібліотека України імені В. І. Вернадського

Найбільша за обсягом фонду і площі приміщень, бібліотека Вернадського входить в 20-ку найбільших національних бібліотек світу [13], [16].

Заснована бібліотека була в 1918 році як Національна бібліотека української держави, і її першим головою Тимчасового комітету для заснування Національної бібліотеки очолив В.І. Вернадський, який також був і першим президентом Української Академії наук.

Вже в 1988 році Центральній науковій бібліотеці Академії наук УРСР, саме так вона називалася в ті роки, було присвоєно ім'я В.І. Вернадського. На сьогоднішній день в бібліотеці більше 40 підрозділів, серед яких чотири наукові інститути, 5 науково-дослідних центри і різні відділи [13], [16]. Склад бібліотечних фондів близько 15 млн примірників, серед яких різні видання від періодики до наукових робіт, картографічних видань та рідкісних книг [13], [16].

Головна сторінка сайту представлена на рис. 1.3.

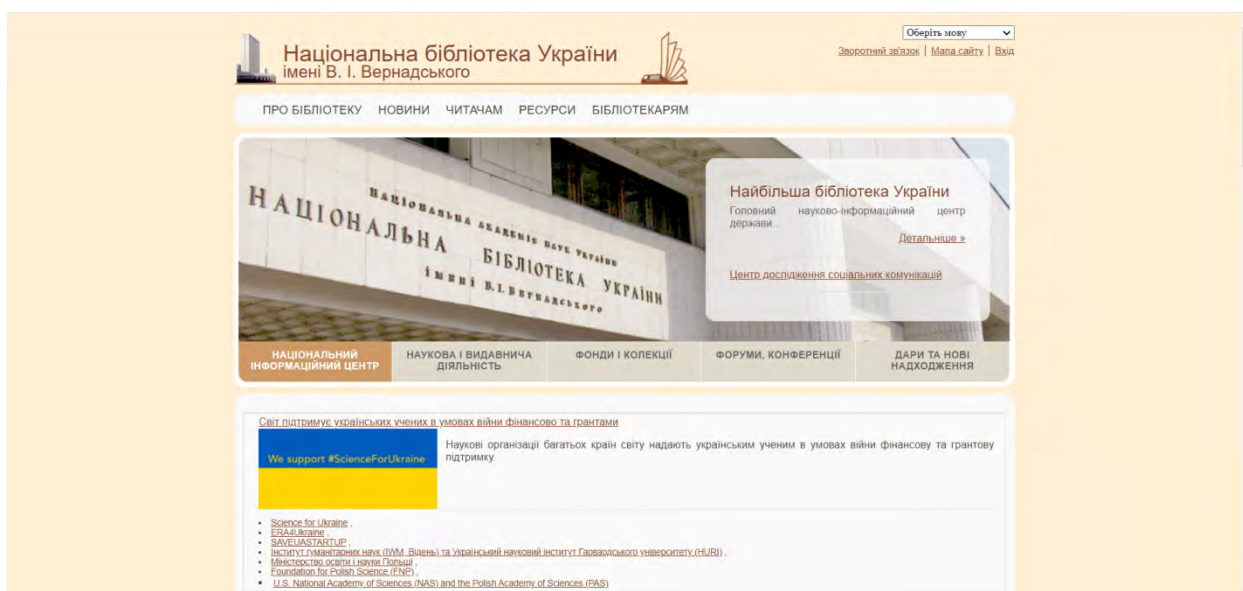


Рисунок 1.3 – Головна сторінка сайту Національної бібліотека України імені В. І. Вернадського [16]

1.3.4 Порівняння існуючих аналогів

Проведемо порівняння за певними критеріями, хід якого представимо у вигляді таблиці (табл. 1.1). Серед критеріїв виділимо наступні:

- представлення електронного каталогу та алфавітного вказівника – чи надає ресурс електронний каталог та алфавітний каталог для зручного пошуку існуючих видань, авторів, тощо;
- можливість опублікувати власні роботи – чи надає сервіс можливість публікувати свої власні роботи авторам-початківцям;
- рейтингування видань – чи надає ресурс можливість оцінки та перегляду рейтингового балу для видання;
- функції онлайн-читання обраних творів – чи надає ресурс можливість читати онлайн твори, що цікаві користувачу.

Таблиця 1.1 – Програмне забезпечення для підтримки освітньої діяльності

Критерії порівняння	УкрЛіб	Libruk	Національна бібліотека України імені В. І. Вернадського	Читанка
Представлення електронного каталогу та алфавітного вказівника	Так	Так	Ні (на сайті представлені ресурси)	Так
Можливість опублікувати власні роботи	Ні	Так	Ні	Так
Рейтингування видань	Ні	Так	Ні	Так
Функції онлайн-читання обраних творів	Так	Так	Ні	Так

З результатів порівняння можна зробити висновок, що на сьогоднішній день формат електронних онлайн-бібліотек достатньо популярний та розвинений, проте усі аналоги значно різняться в функціоналі. Так сайт УкрЛіб є найбільш привабливим на перший погляд для користувачів, проте пошук на сайті організовано недостатньо комфортно. Libruk для пошуку більш комфортний та й функція опублікування власних творів – є достатньо цікавою. Проте швидкість роботи та загальна оптимізація ресурсу недостатні – сайт періодично підвисає під час роботи. Національна бібліотека України імені В. І. Вернадського взагалі скоріше є адміністративним ресурсом: сайт дозволяє виконувати пошук робіт, більш того виводить результати пошуку в форматі бібліографічних стандартів опису джерел. Проте ознайомитися із роботами на сайті не можливо. А для переходу до алфавітного вказівника або ж каталогу треба довго чукати необхідні ресурси у відповідному пункті меню.

Саме тому актуальною задачею залишається реалізація програмного застосунку електронної бібліотеки у вигляді кроссплатформного вебзастосунку, що буде надавати більшість функцій звичайної бібліотеки із

розширеними можливостями читати видання онлайн та додавати книжки та видання авторам-початківцям.

1.4 Постановка завдання

Метою проєкту є реалізація програмного застосунку електронної бібліотеки.

Система має надавати доступ до книг, що зберігаються в її базі даних (БД). Вона надає можливість пошуку книг, перегляду їх анотації, читання книг (онлайн, скачування), завантаження своїх книг. Також система має надавати можливість особистого листування між зареєстрованими користувачами, додавання коментарів і оцінок до книг. Система повинна вести рейтинг найпопулярніших книг. Система підтримує можливість реєстрації користувачів в системі.

1.4.1 Межі проєкту

Система надає доступ до книг сучасних авторів і зареєстрованих користувачів. Кожному не зареєстрованому користувачеві дається можливість звичайного і розширеного пошуку книг, перегляду короткої інформації про книгу (автор, рік, анотація, рецензія, коментарі). Так само вони можуть читати книгу в он-лайн режимі. Для більш розширених прав користувачі повинні реєструватися.

Зареєстровані користувачі діляться на авторів і читачів.

Автори можуть додавати свої книги, які після перевірки модератором будуть додані в базу даних і видно іншим користувачам. Також вони можуть завантажувати книги, додавати коментарі. Є можливість особистого листування між зареєстрованими користувачами. Читачі можуть все теж, що і автори, крім додавання своїх книг.

Модератори здійснюють контроль над контентом сайту. Вони перевіряють книги на відповідність з жанрами, стежать за авторськими правами на сайті, не допускають дубляжу. Мають право заблокувати сторінки користувачів, які порушують умови сайту. Для видалення таких сторінок модератори повинні звертатися до адміністраторів. Оновлюють книжкові полиці книгами не користувачів сайту.

Адміністратори безпосередньо працюють з базою даних системи.

В системі будуть використовуватися такі готові модулі як «Пошук», «Розширений пошук», «Реєстрація», «Оцінка», «Особисте листування».

1.4.2 Функціональність системи

Система повинна забезпечити:

- реєстрація на сайті;
- можливість перегляду контенту;
- додавання нових книг;
- додавання нових користувачів;
- редагування вмісту;
- пошук книг;
- завантаження книг;
- читання он-лайн;
- редагування прав користувача (читач/автор);
- додавання коментаря;
- додавання оцінки добутку;
- сповіщення про нові особисті повідомлення;
- ведення рейтингу книг;
- особиста переписка між користувачами;
- блокування сторінок порушників правил сайту;

- видалення сторінок порушників правил сайту.

1.4.3 Вимоги до інтерфейсу

Інтерфейс системи повинен бути ергономічним, інтуїтивно-зрозумілим для користувачів системи. Вся інформація сайту повинна бути чітко структурована і розділена на розділи.

Розділ пошуку повинен мати настільки простий інтерфейс, наскільки можливо зробити без втрати його багатофункціональності.

Сторінки реєстрації та авторизації Користувача повинні мати підказки з приводу заповнення форм.

1.4.4 Вимоги до продуктивності

Проект повинен підтримувати до 30 запитів в секунду. Всі компоненти системи повинні працювати досить продуктивно, щоб користувачі мали можливість зручно працювати з системою. З максимальною продуктивністю повинен працювати модуль пошуку, забезпечуючи швидкість пошуку до 5000 патентів/патентних бюро за секунду. Також пропускна здатність системи реєстрації повинна забезпечувати одночасну обробку до 10 запитів на реєстрацію.

1.4.5 Вимоги до безпеки

Права користування системою залежать від статусу користувача

Повні права доступу маю адміністратори. Вони практично можуть управляти всією інформацією сайту

Модератори оновлюють систему новими поповненнями книг. Також вони перевіряють книги на відповідність короткої інформації та її змісту.

Автори можуть додавати свої книги

Читачі, як і всі зареєстровані користувачі, можуть завантажувати книги, додавати коментарі, ставити оцінки.

Незареєстровані користувачі, як і зареєстровані, можуть дивитися коротку інформацію про книгу, читати онлайн, дивитися коментарі, інформацію про автора.

Шифрування паролів буде проводитися методом Advanced Encryption Standard (AES).

1.4.6 Вимоги до даних

Всі дані повинні зберігатися на сервері. Користувачі повинні отримувати всю необхідну інформацію через запити до бази даних.

Книги закріплені за авторами.

Кожна книга повинна мати такі мітки як:

- назва;
- жанр;
- форма;
- року;
- статус (закінчена, редагується);
- рейтинг;
- коментар.

Користувачі повинні мати таку інформацію:

- ПІБ;
- логін;
- псевдонім;
- контактна інформація.

1.4.7 Експлуатаційні вимоги

Для користування додатком, користувач повинен мати девайс з доступом до мережі Інтернет, обладнаний веб-браузером. Для створення власного сховища користувач має зареєструватися.

1.4.8 Не функціональні вимоги до продукту

Мінімальні системні вимоги та програмне забезпечення (ПЗ), що необхідні для функціонування серверної частини мобільного застосунку:

- сервер (IBM-сумісна архітектура);
- серверна операційна система (ОС): Unix або Windows Server;
- інтерпретатор PHP5;
- MySQL-сервер для керування користувачами та організації доступу до БД.

Для експлуатації клієнтської частини необхідні такі програмно-технічні засоби:

- ПК з IBM-сумісною архітектурою;
- процесор із тактовою частотою від 1,7 GHz;
- обсяг оперативної пам'яті – від 3GB (для безперебійної роботи нових версій браузерів);
- підключення до мережі Internet;
- встановлений браузер.

1.5 Висновки за розділом 1

На поточному етапі роботи було виконане ознайомлення із предметною областю електронних бібліотек. Було досліджено принципи їх організації та функціонування. Додатковий аналіз існуючих аналогів був проведений з

метою виявлення переваг та недоліків для майбутнього формування низки функціональних та не функціональних вимог до майбутнього застосування.

2 МАТЕРІАЛИ І МЕТОДИ

В попередньому розділі було виконано ознайомлення із предметною областю електронних бібліотек. Було досліджено принципи їх організації та функціонування. Додатковий аналіз існуючих аналогів був проведений з метою виявлення переваг та недоліків для формування низки функціональних та не функціональних вимог до майбутнього застосунку.

Наразі, в другому розділі буде висвітлено низку технічних питань розробки: вибір технології та техніки розробки, принципи реалізації. В цьому розділі буде детально проаналізовано та обрано мову програмування для реалізації функціональної частини, програмні комплекси та засоби для оформлення ПЗ.

2.1 Вибір мови програмування та інших технологій розробки

PHP – це поширена скриптова мова з відкритим вихідним кодом, популярна для вебзастосунків. Хоча спочатку це означало: особиста домашня сторінка, PHP тепер є рекурсивною аббревіатурою від: гіпертекстового препроцесора [17]-[19].

У своїй найпростішій формі PHP – це серверна скриптова мова, вбудована в HTML. PHP дозволяє веб-розробникам створювати динамічний контент і взаємодіяти з базами даних. PHP відомий своєю простотою, швидкістю та гнучкістю – функціями, які зробили його наріжним каменем у світі веб-розробки [17]-[19].

PHP використовується 79,1% всіх веб-сайтів з відомим серверним програмуванням, і його використання зростає в останнє десятиліття, в порівнянні з 72% в 2010 році. PHP використовується на деяких найбільших веб-сайтах і платформах, включаючи Facebook, Wikipedia, WordPress і Zoom [17]-[19].

Як і будь-яка інша мова сценаріїв, PHP – це, по суті, інструмент, який ви використовуєте як для підключення до вашої бази даних для отримання інформації, так і для передачі цієї інформації на ваш веб-сервер для відображення у форматі HTML. Але є багато аспектів PHP, які відрізняють його від інших мов [17]-[19].

Простіше, PHP це:

- мова сценаріїв - мови сценаріїв інтерпретуються іншою програмою під час виконання (немає необхідності в компіляції). Мови сценаріїв можуть інтерпретуватися як на стороні сервера, так і на стороні клієнта (в браузері) [17]-[19];

- серверна – PHP це серверна мова сценаріїв, що обробляється інтерпретатором PHP на веб-сервері; результат (вихідні дані) відправляється в веб-браузер у вигляді звичайного HTML [17]-[19];

- відкритий вихідний код – PHP знаходиться у вільному доступі для завантаження і використання [17]-[19];

- об'єктно-орієнтованість – об'єктно-орієнтоване програмування (ООП) використовує концепцію об'єктів для зберігання даних і функцій, які допомагають створювати більш складні, повторно використовувані веб-додатки. ООП був доданий до PHP5[17]-[19];

- швидкість – PHP використовує власну пам'ять, мінімізуючи навантаження на сервер і підвищуючи продуктивність. PHP може бути на 382% швидше, ніж Python, і на 195% швидше, ніж Ruby[17]-[19];

- простота – синтаксис PHP легко зрозуміти і вивчити, незалежно від того, створюєте ви з нуля або використовуєте існуючі фреймворки або доповнення [17]-[19];

- хороша підтримка – PHP підтримує всі провідні бази даних (MySQL, SQLite, ODBC), сумісний з більшістю серверів (Apache, IIS і т.д.), переносимо на всі платформи (Windows, Mac OS, Linux і т. д.) і може додатково підтримуватися фреймворками PHP (Laravel, CodeIgniter, Symfony) і безліч добре укомплектованих і перевірених бібліотек [17]-[19].

RНР – це вільно типізована мова, яка використовує в цілому вісім типів даних, що використовуються для побудови змінних (в яких зберігаються дані різних типів) [17]-[19]. На відміну від інших мов програмування, RНР досить розслаблений, коли справа доходить до змінних, оцінки та вгадування типу даних. Всі змінні починаються зі знака долара, але можуть бути названі будь-яким ім'ям, що починається з букви[17]-[19].

На відміну від багатьох мов програмування, змінні RНР спочатку не є ні сильними, ніслабкими. Швидше за все, RНР динамічний і може використовувати як строго, так і слабо типізовані змінні, що вказують, наскільки строго або слабо відбувається перетворення[17]-[19].

RНР можна використовувати для розробки динамічних та інтерактивних веб-сторінок, додатків та платформ електронної комерції. RНР добре підходить для різних веб-завдань, від створення динамічних веб-сторінок, надсилання електронних листів та збирання веб-форм до надсилання та отримання файлів cookie. Далі, оскільки RНР може зберігати, видаляти і змінювати інформацію в базі даних, він стає міцною основою для створення веб-додатків, включаючи системи управління контентом (CMS), призначені для користувача онлайн-бази даних, веб-сайти електронної комерції, ігрові додатки і портали спільноти [17]-[19].

Коли справа доходить до надання надійного цифрового досвіду, який зміцнює відносини між брендом і клієнтами і підвищує ваші конкурентні переваги, RНР пропонує безпрецедентні можливості для розвитку [17]-[19].

Проведемо порівняння мов програмування для реалізації вебзастосунків, результати порівняння представимо в табличній формі (табл. 2.1). У якості критеріїв будемо використовувати наступні:

- простота розробки – матеріали, спільнота та швидкість вивчення мови програмування для початку розробки;
- навантаженість синтаксису – складність синтаксичних конструкцій мови програмування;

- механізми для спрощення розробки – конструкторські механізми для спрощення розробки;
- подальша масштабованість – складність розвитку (в тому числі накопичування функціоналу) застосунків під час/після розробки;
- програмні надбудови – фреймворки та CMS для спрощення та організованості розробки.

Таблиця 2.1 – Порівняння мов програмування

Критерій	PHP	Python	JavaScript
Простота розробки	++	++	++
Навантаженість синтаксису	+	+	++
Механізми для спрощення розробки	++	+	++
Подальша масштабованість	++	±	+
Програмні надбудови	++	+	+

З таблиці видно, що PHP вже довгий час є найбільш популярною мовою програмування для розробки вебзастосунків, саме тому має велику спільноту та найбільш оптимізовані сценаріїв підтримки на серверах, що робить її найоптимальнішим вибором для розробки.

2.2 Вибір та обґрунтування CMS

Joomla – це CMS з відкритим вихідним кодом, яка використовується для розробки веб-сайтів і додатків [20], [21]. Розробник може підключити Joomla до MySQL та інших баз даних для ефективної розробки сайтів і додатків. Він був випущений в 2005 році, і його основна увага була приділена зручності використання і розширюваності [20], [21].

Joomla – це комплексне рішення з відкритим кодом, яке кожен може безкоштовно завантажувати, використовувати та змінювати для створення

динамічних та інтерактивних сайтів для різних цілей та галузей [20], [21]. Від корпоративних сайтів до магазинів електронної комерції, блогів і сайтів соціальних мереж-все це створює Joomla [20], [21].

Joomla має велике і активне співтовариство розробників. На вибір доступні тисячі розширень і шаблонів, що робить його однією з найбільших платформ CMS, доступних сьогодні [20], [21].

Joomla підтримує 2 мільйони веб-сайтів по всьому світу. На це припадає майже 3% всієї мережі [20], [21].

Joomla може допомогти вам більше, ніж просто простий веб-сайт, використання інтегрованих технологій дозволяє розробникам створювати інноваційні рішення, які перевершують очікування клієнтів [20], [21].

Joomla – проста в освоєнні, швидка в налаштуванні і недорога у використанні система управління контентом [20], [21]. Це дозволяє створювати рішення для малих, середніх і великих підприємств. Joomla має широкий і широкий вибір плагінів, розширень, шаблонів та інших функціональних можливостей, розроблених сторонніми спільнотами, щоб дозволити навіть початківцям веб-розробникам максимально використовувати цю платформу [20], [21].

Joomla надає швидке і надійне рішення для розробки веб-сайтів:

- у Joomla є безліч перевірених сторонніх розширень [20], [21];
- тисячі високоякісних безкоштовних шаблонів [20], [21];
- також доступні платні шаблони з підтримкою світового класу;
- зміна кольорів, шрифтів, макетів та інші функції;
- дуже багатобезкоштовної документації та відео-тренінгів;
- потужні вбудовані інструменти пошукової оптимізації(SEO) [20], [21];
- пункти меню можуть мати свій власний мета-опис, ключові слова, що дозволяє налаштовувати параметри SEO [20], [21];
- Joomla підтримується і підтримується великою спільнотою добровольців [20], [21];

- безкоштовно назавжди;
- Joomla має вбудовану двофакторну аутентифікацію та рівні контролю доступу для забезпечення високого рівня безпеки. Спеціальна команда Security Strike Team завжди випереджає події і випускає виправлення до того, як нові експлойти потрапляють в екосистему [20], [21];
- вона сприяє розробці багатомовного сайту з використанням понад 70 пакетів перекладів;
- її легко адаптувати і розширювати відповідно до вимог;
- це стабільна і легка PHP CMS.

В даний час веб-розробка Joomla зазвичай використовується для програмного забезпечення [20], [21]. Але цим справа не обмежується. Вебзастосунків ростуть завдяки безперебійному зв'язку між користувачами в ці найгірші часи пандемій. Joomla використовується для застосунків на вимогу з кількох причин. Головна причина полягає в тому, щоб дозволити декільком розробникам зручно працювати над одним і тим же проектом [20], [21].

Таким чином, це допомагає їм створювати високоякісні вебзастосунків Відповідно до стандартів їх компанії. Крім того, Joomla дозволяє розробляти кроссплатформенні застосунки [20], [21]. Щоб їх користувачі не були обмежені пристроями при спробі отримати доступ до вебзастосунків, створених за допомогою Joomla [20], [21].

Варто відзначити, що вебзастосунки Joomla широко використовуються в сфері охорони здоров'я [20], [21]. Через Covid-19 люди шукають швидких медичних послуг. Вони не можуть відвідувати лікарні або клініки через норми соціального дистанціювання. Єдині рішення, які вони мають, - це використовувати веб-додатки, спілкуватися з фахівцями та ділитися з ними своїми звітами про стан здоров'я. Саме тут на перший план виходять веб-додатки Joomla [20], [21].

Платформи розробили програми, які вам не потрібно встановлювати на свої пристрої. Люди можуть отримати доступ до цих веб-додатків через

браузери. Ці програми розроблені з використанням високорівневих функцій безпеки і простого у використанні користувальницького інтерфейсу [20], [21].

Порівняльна характеристика найпоширеніших CMS для розробки інтернет-магазинів наведена у таблиці 2.2.

Серед критеріїв порівняння будемо розглядати наступні:

- вимоги до серверу та хостингу – важливий критерій, з огляду на предметну область та необхідність зберігати великі об’єми даних;
- стабільність та регулярність оновлень – оцінка стабільності, та регулярності оновлення CMS (з урахуванням оновлень для підвищення безпеки);
- масштабованість на наступна підтримка застосунків – легкість масштабованості та подальшої підтримки застосунків;
- взаємодія із системами керування БД (СКБД) – легкість взаємодії із СКБД та захищеність такої взаємодії.

З порівняння наведеного в таблиці можна зробити висновок, що використання Joomla є найбільш оптимальним варіантом для розробки враховуючи специфіку проблемної області та надмірні вимоги до стабільності майбутнього застосунку. Через її популярність більшість внутрішніх механізмів та процесів максимально оптимізовано під потреби розробників.

Таблиця 2.2 – Порівняння CMS

	Drupal	WordPress	Joomla!
Вимоги до серверу та хостингу	++	+	++
Стабільність та регулярність оновлень	+	++	++
Масштабованість на наступна підтримка застосунків	+	++	++
Взаємодія із системами керування БД (СКБД)	+	+	++

2.3 Структурна схема розробки

На рисунку 2.1 наведено структурну схему ПЗ, що розробляється.

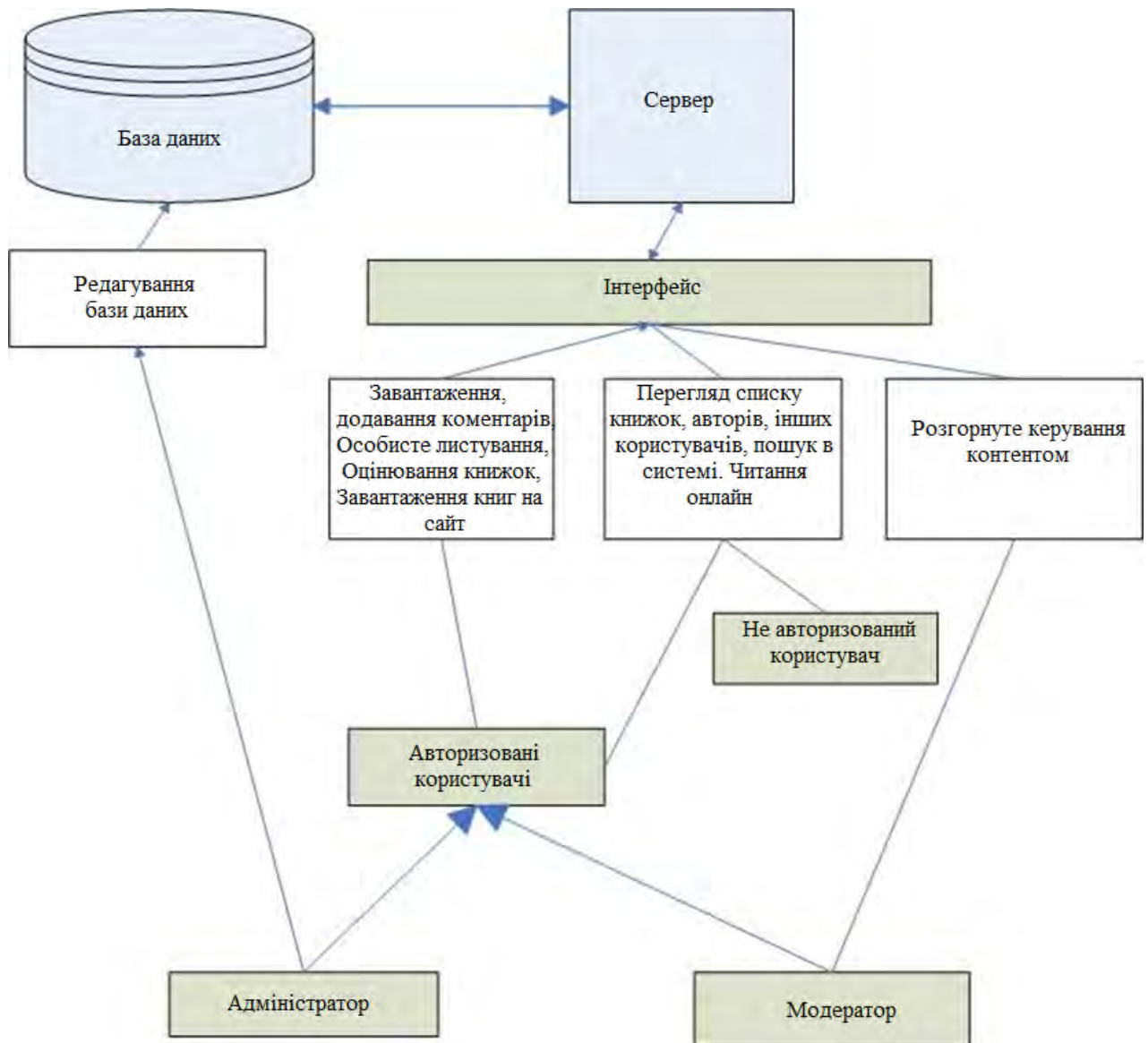


Рисунок 2.1 – Загальна схема системи за обраною темою

Управління контентом буде полягати у виконанні наступних функцій:

- додавання нових книг;
- додавання нових користувачів;
- редагування книг користувачів;
- видалення книг;

- видалення сторінок порушників правил сайту;
- реєстрація на сайті;
- додавання нових книг;
- редагування книг користувачів;
- додавання нових користувачів;
- редагування вмісту;
- пошук книг;
- завантаження книг;
- читання он-лайн;
- редагування прав користувача (читач / автор);
- додавання коментаря;
- додавання оцінки добутку;
- сповіщення про нові особисті повідомлення;
- ведення рейтингу книг;
- особиста переписка між користувачами;
- блокування сторінок порушників правил сайту;
- видалення сторінок порушників правил сайту.

Структура вебзастосунку, що розробляються з використанням Joomla зображена на рисунку 2.2.

3 ОСНОВНІ РІШЕННЯ ЩОДО РЕАЛІЗАЦІЇ КОМПОНЕНТІВ СИСТЕМИ

3.1 Проектування моделей системи

Unified Modeling Language (UML) — це мова моделювання загального призначення [22]-[24]. Основна мета UML - визначити стандартний спосіб візуалізації того, як була розроблена система. Він дуже схожий на креслення, які використовуються в інших галузях техніки [22]-[24].

UML – це не мова програмування, це скоріше візуальна мова. Ми використовуємо діаграми UML, щоб зобразити поведінку та структуру системи. UML допомагає інженерам програмного забезпечення, бізнесменам і системним архітекторам моделювати, проектувати та аналізувати [22]-[24]. Група управління об'єктами прийняла уніфіковану мову моделювання як стандарт у 1997 році. Відтоді нею керує OMG. Міжнародна організація зі стандартизації опублікувала UML як затверджений стандарт у 2005 році. UML переглядався протягом багатьох років і періодично переглядається [22]-[24].

Діаграма класів UML – це графічне позначення, яке використовується для побудови та візуалізації об'єктно-орієнтованих систем [22-24]. Діаграма класів на UML – це тип діаграми статичної структури, яка описує структуру системи, показуючи:

- класи;
- їх атрибути;
- операції (або методи);
- і відносини між об'єктами [22]-[24].

Діаграма класів вебзастосунку, що розробляється зображена на рис. 3.1.

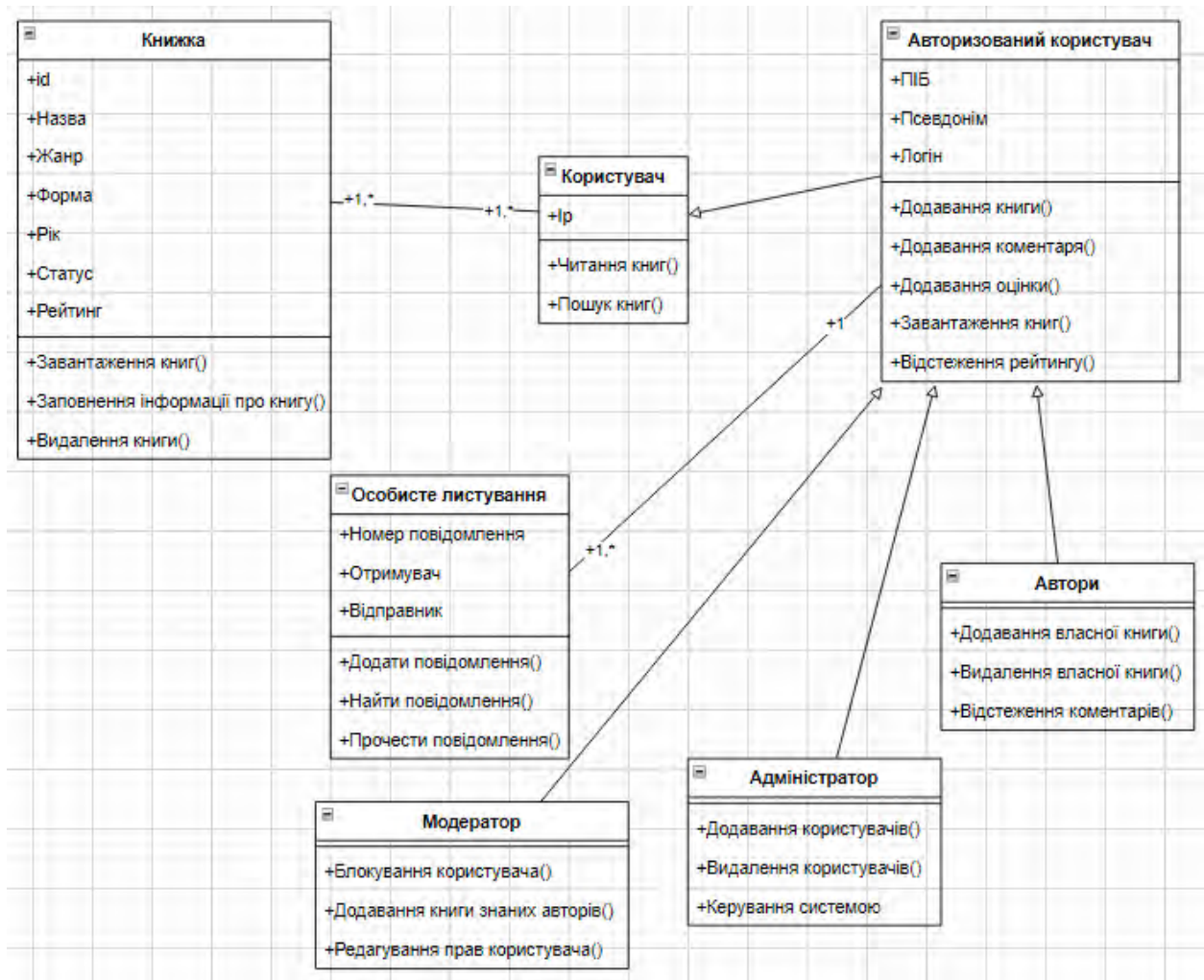


Рисунок 3.1 – Діаграма класів

Діаграми пакетів – це структурні діаграми, які використовуються для показу організації та розташування різних елементів моделі у вигляді пакетів [22]-[24]. Пакет – це групування пов’язаних елементів UML, таких як діаграми, документи, класи чи навіть інші пакети. Кожен елемент вкладений в пакунок, який зображений у вигляді папки файлів на діаграмі, а потім впорядкований ієрархічно в межах діаграми. Діаграми пакетів найчастіше використовуються для наочної організації багатoshарової архітектури в будь-якому класифікаторі UML, наприклад, у програмній системі [22]-[24].

Діаграма пакетів для вебзастосунку, що розробляється на рис. 3.2.

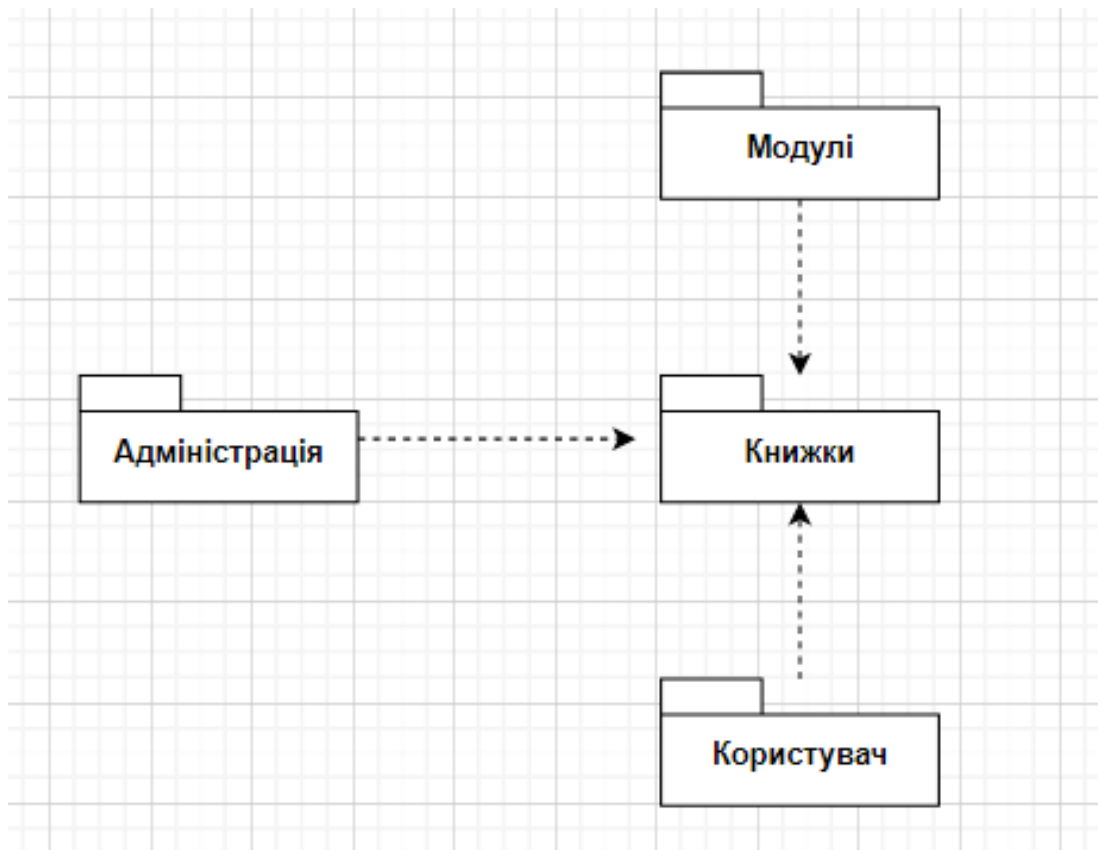


Рисунок 3.2 – Діаграма пакетів

Мета діаграми компонентів – показати взаємозв’язок між різними компонентами в системі. Для цілей UML 2.0 термін «компонент» відноситься до модуля класів, які представляють незалежні системи або підсистеми з можливістю взаємодії з рештою системи [22]-[24]. Існує цілий підхід до розробки, який обертається навколо компонентів: розробка на основі компонентів. У цьому підході діаграми компонентів дозволяють планувальнику ідентифікувати різні компоненти, щоб вся система виконувала те, що вона повинна робити. Частіше, в підході до програмування, діаграма компонентів дозволяє старшому розробнику групувати класи разом на основі спільної мети, щоб розробник та інші могли дивитися на проект розробки програмного забезпечення на високому рівні [22]-[24].

Діаграми компонентів для вебзастосунку, що розробляється зображена на рис. 3.3.

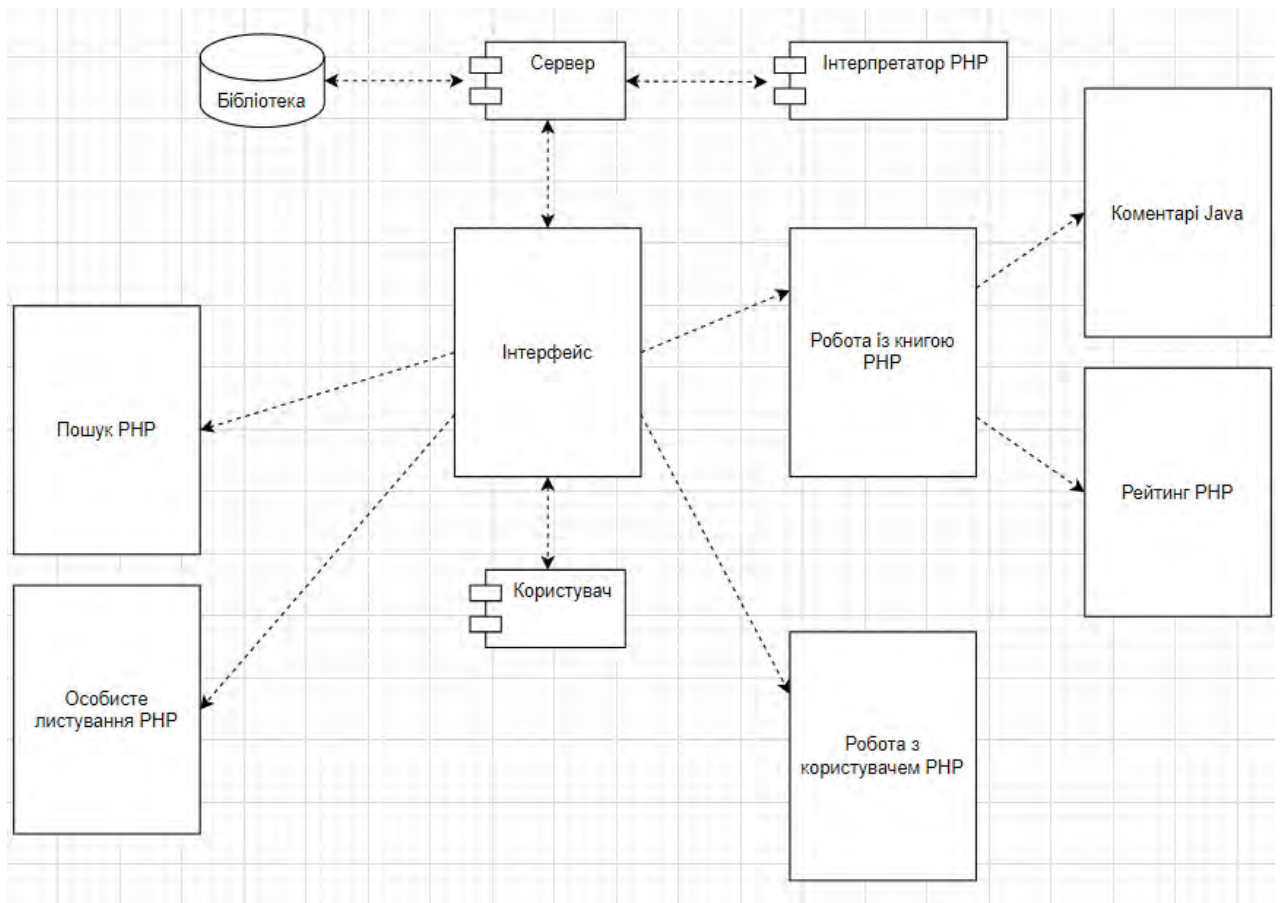


Рисунок 3.3 – Діаграма компонентів

3.2 Моделювання системи з точки зору прецедентів

Діаграма варіантів використання – це графічне зображення можливої взаємодії користувача з системою [22]-[24]. Діаграма варіантів використання показує різні варіанти використання та різні типи користувачів, яких має система, і часто супроводжується діаграмами інших типів. Варіанти використання представлені або колами, або еліпсами. Актори часто зображені у вигляді фігурок [22]-[24].

Діаграми прецедентів зображені на рис. 3.4–3.6.

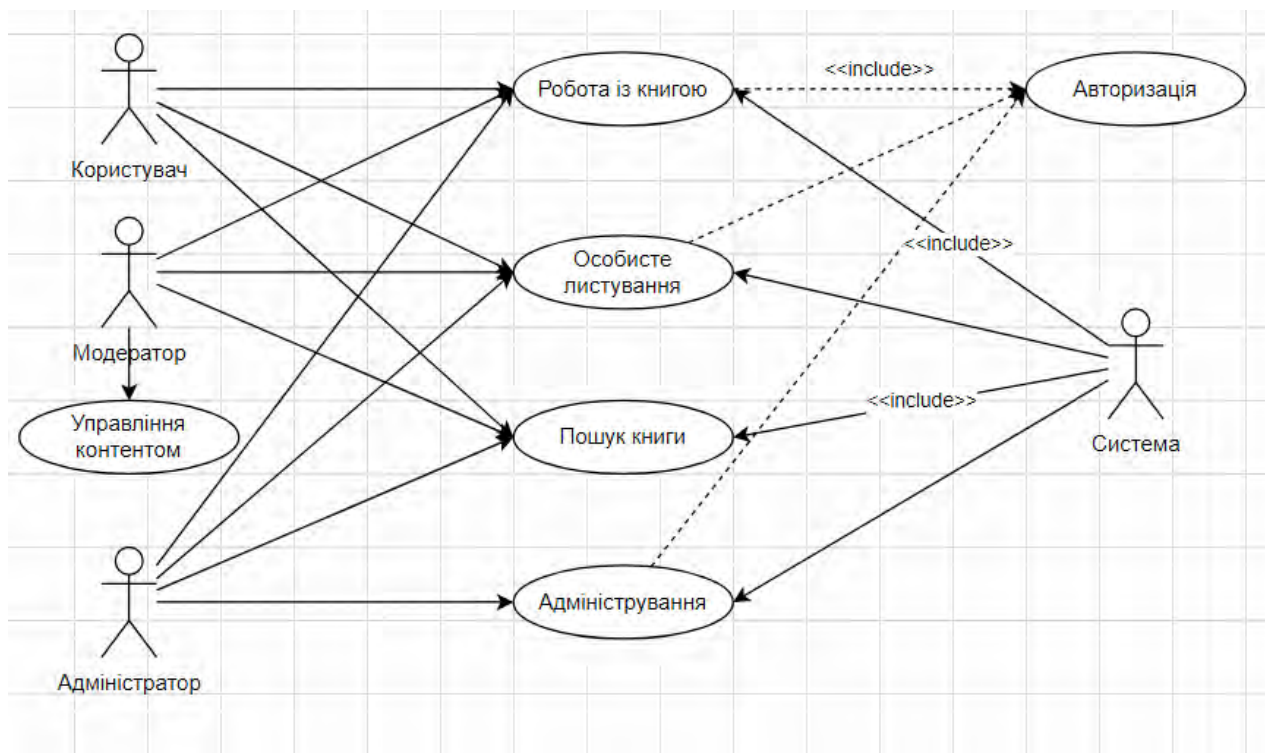


Рисунок 3.4 – Діаграма загального користування системою

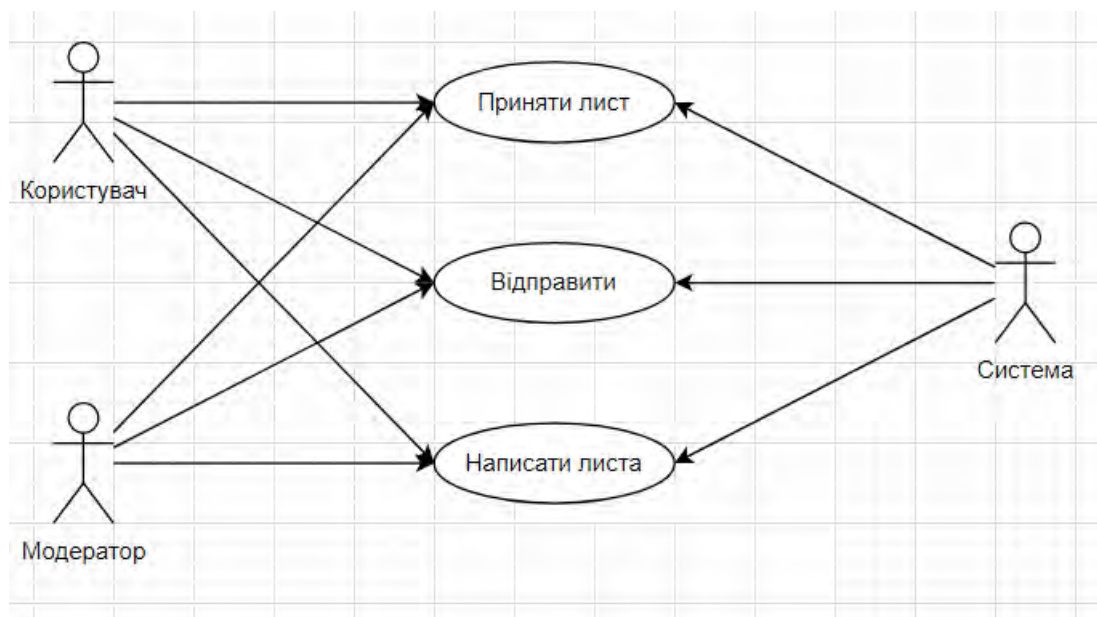


Рисунок 3.5 – Діаграма прецеденту «Особисте листування»

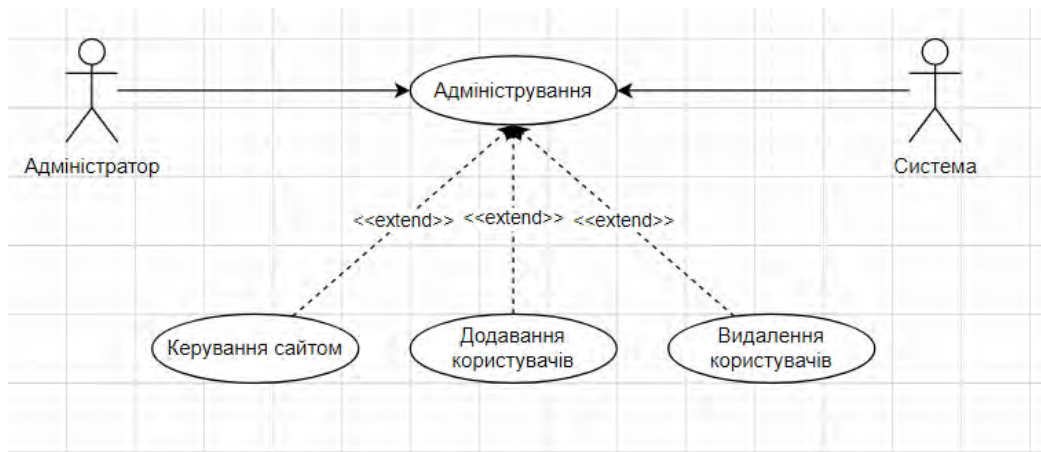


Рисунок 3.6 – Діаграма прецеденту «Адміністрування системи»

3.3 Фізичне проектування

Діаграма активностей (видів діяльності) – один із доступних видів діаграм, що підтримуються Flexberry Designer [22]-[24]. Вона, як і діаграма станів, відбиває динамічні аспекти поведінки системи. По суті, ця діаграма є блок-схемою, яка наочно показує, як потік управління переходить від однієї діяльності до іншої [22]-[24].

Діаграми діяльності та станів для вебзастосунку, що розробляється зображені на рис. 3.7–3.9.

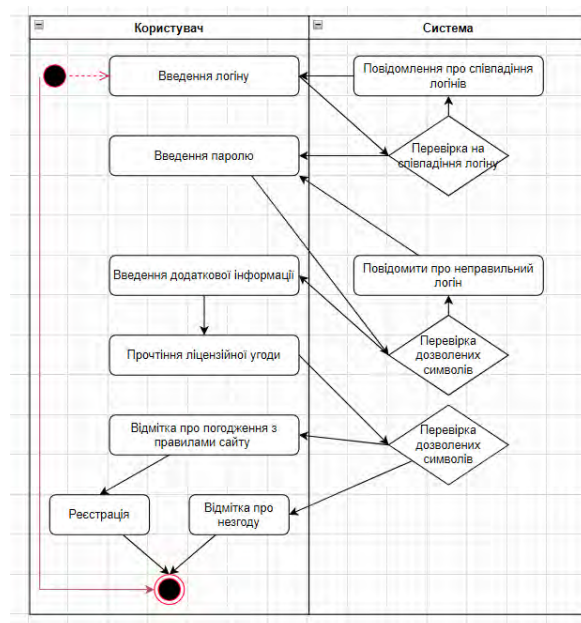


Рисунок 3.7 – Діаграма діяльності «Відправлення листа»

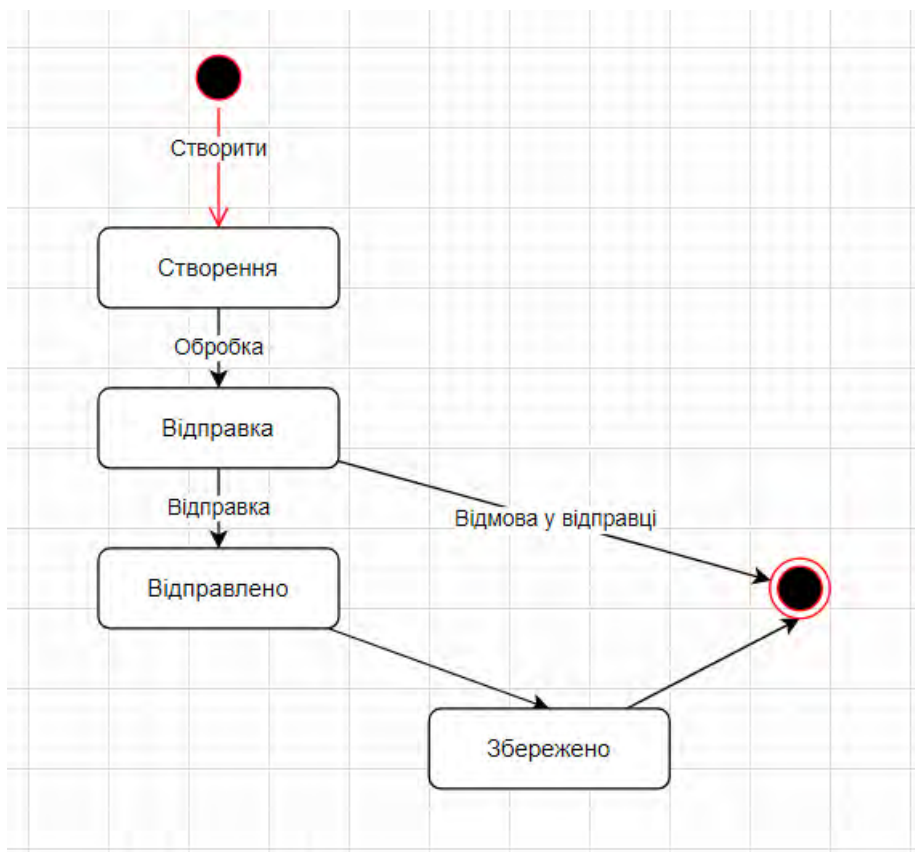


Рисунок 3.8 – Діаграма станів для відправки листа

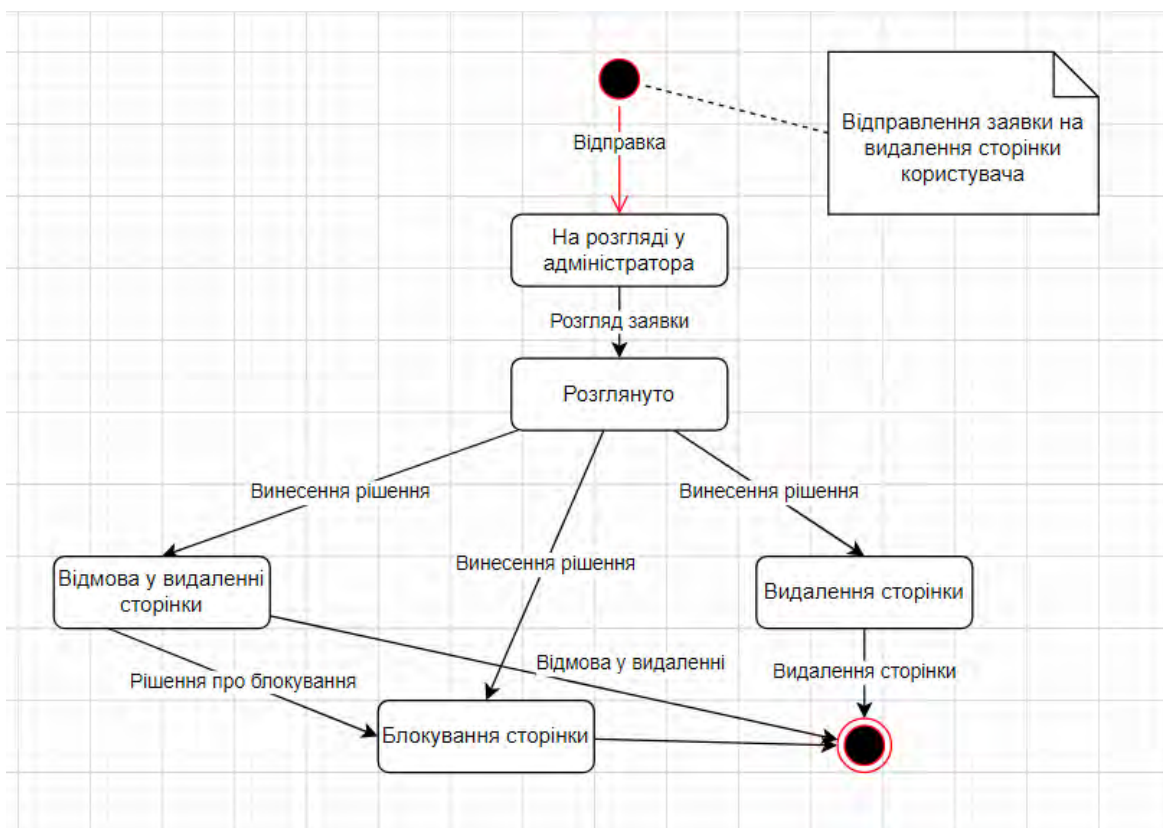


Рисунок 3.9 – Діаграма станів видалення користувача

Діаграми кооперацій використовуються, щоб показати, як об'єкти взаємодіють, щоб виконувати поведінку конкретного варіанту використання або частини варіанту використання [22]-[24]. Поряд із діаграмами послідовності, кооперація використовується дизайнерами для визначення й уточнення ролей об'єктів, які виконують певний потік подій у варіанті використання [22]-[24]. Вони є основним джерелом інформації, яка використовується для визначення обов'язків класу та інтерфейсів [22]-[24].

Діаграми кооперацій для вебзастосунку у вигляді файлообміннику, що розробляється зображені на рис. 3.10 та 3.11.

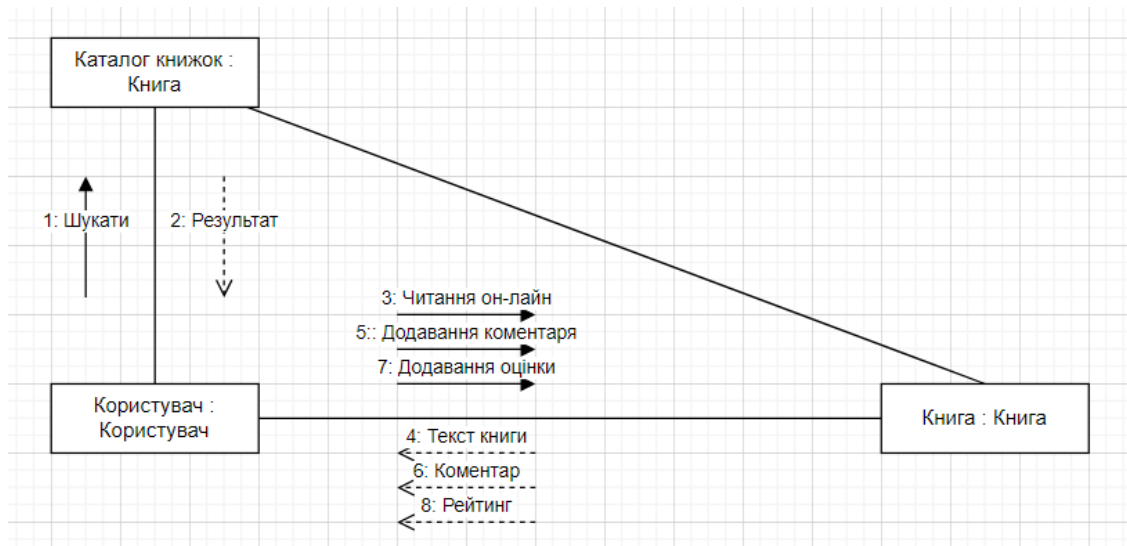


Рисунок 3.10 – Діаграма кооперації «Пошук книги»

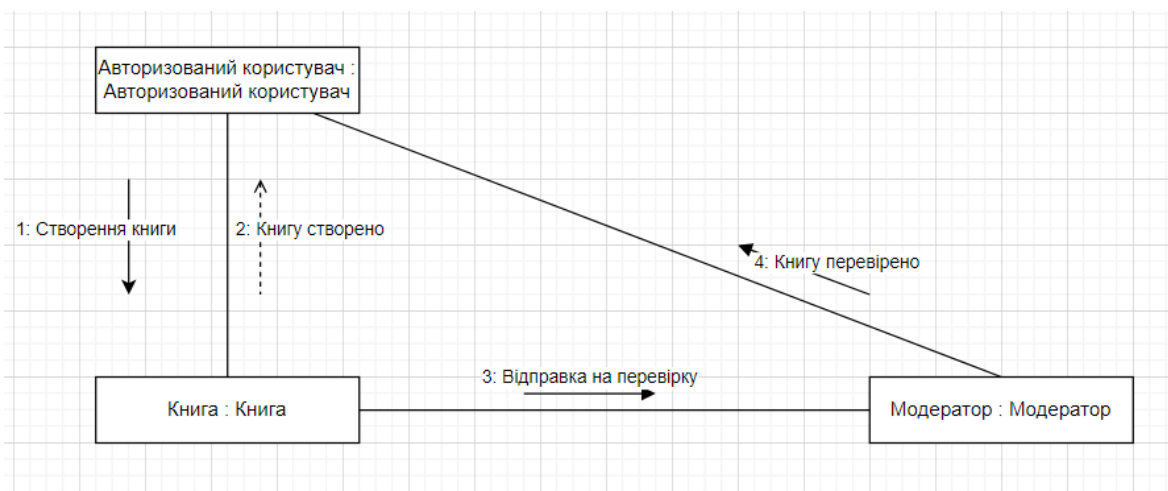


Рисунок 3.11 – Діаграма кооперації «Створення книги»

Діаграми послідовностей для вебзастосунку, що розробляється зображені на рис. 3.12 та 3.13.

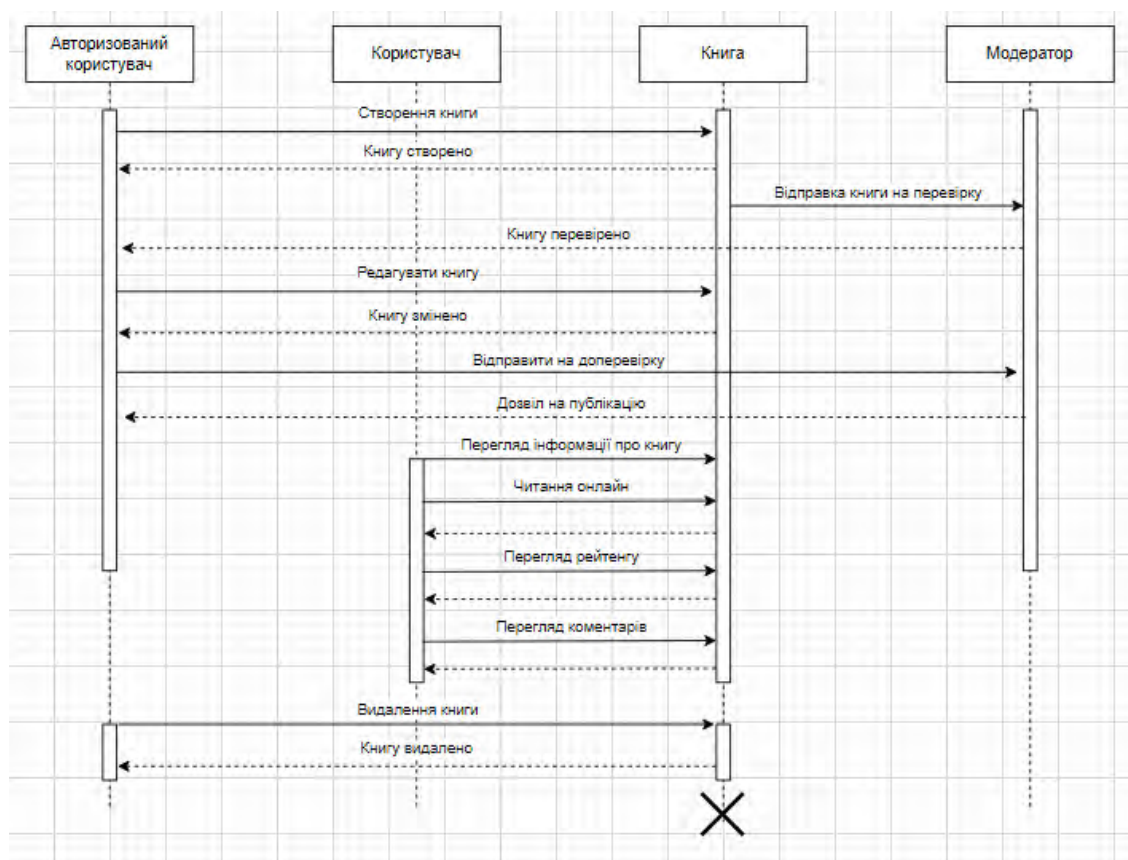


Рисунок 3.12 – Діаграма послідовностей «Створення книги»

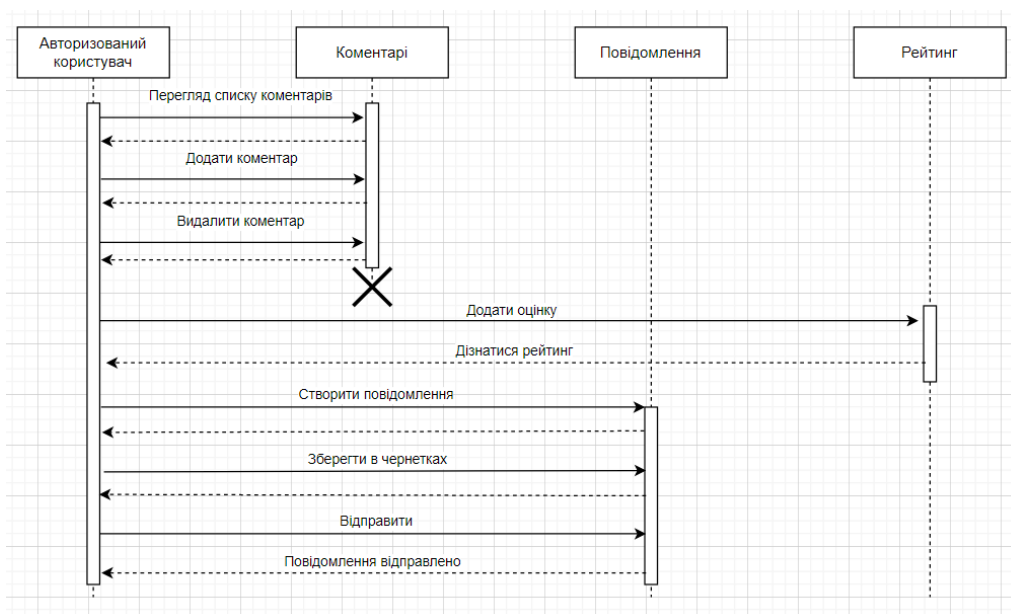


Рисунок 3.13 – Діаграма послідовностей «Робота із системою авторизованого користувача»

3.4 Проектування інтерфейсу

Дизайн інтерфейсу користувача (UI) фокусується на передбаченні того, що користувачам може знадобитися зробити, і забезпеченні того, щоб інтерфейс містив елементи, до яких легко отримати доступ, зрозуміти і використовувати для полегшення цих дій [25]-[27]. Інтерфейс користувача об'єднує концепції дизайну взаємодії, візуального дизайну та інформаційної архітектури [25]-[27].

Вибір елементів інтерфейсу. Користувачі вже знайомі з елементами інтерфейсу, що діють певним чином, тому намагайтеся бути послідовними і передбачуваними в своєму виборі і їх розташуванні. Це допоможе у виконанні завдань, підвищенні ефективності та задоволеності [25]-[27].

Елементи інтерфейсу включають, але не обмежуються ними:

а) елементи управління введенням [25]-[27]:

- 1) кнопки;
- 2) текстові поля;
- 3) прапорці, перемикачі;
- 4) випадаючі списки;
- 5) списки, перемикачі;
- 6) поле дати;

б) навігаційні компоненти [25]-[27]:

- 1) слайдер;
- 2) поле пошуку;
- 3) розбивка на сторінки;
- 4) слайдер;
- 5) теги;
- 6) значки;

в) інформаційні компоненти [25]-[27]:

- 1) спливаючі підказки;
- 2) значки;

- 3) індикатор виконання;
- 4) повідомлення;
- 5) вікна повідомлень;
- г) модальні вікна;
- д) контейнери [25]-[27]:
 - 1) гармошка.

Бувають випадки, коли для відображення вмісту може знадобитися кілька елементів. Коли це відбувається, важливо враховувати компроміси [25]-[27]. Наприклад, іноді елементи, які можуть допомогти вам заощадити місце, створюють велике психологічне навантаження на користувача, змушуючи його вгадувати, що знаходиться в випадаючому списку або яким може бути елемент [25]-[27].

Рекомендації щодо розробки інтерфейсу [25]-[27]. Все залежить від знання користувачів, включаючи розуміння їх цілей, навичок, уподобань і тенденцій. Як тільки розробник дізнається про свого користувача, обов'язково враховує наступне при розробці свого інтерфейсу [25]-[27]:

Зберігати інтерфейс простим. Кращі інтерфейси практично невидимі для користувача. Вони уникають непотрібних елементів і зрозумілі в мові, яку вони використовують на етикетках і в повідомленнях [25]-[27].

Створювати послідовність і використовувати загальні елементи інтерфейсу користувача. Використовуючи загальні елементи у вашому інтерфейсі користувача, користувачі відчують себе більш комфортно і можуть виконувати завдання швидше [25]-[27]. Важливо також створювати шаблони в мові, макеті та дизайні по всьому сайту, щоб підвищити ефективність. Як тільки користувач навчиться щось робити, він повинен мати можливість перенести цю навичку в Інші частини сайту [25]-[27].

Слід бути цілеспрямовані у верстці сторінки. Розглядати просторові відносини між елементами на сторінці та структуруйте сторінку на основі важливості [25]-[27]. Ретельне розміщення елементів може допомогти

привернути увагу до найважливіших частин інформації і може сприяти скануванню і читабельності [25]-[27].

Стратегічно використовувати колір і текстуру. Можна звернути увагу на предмети або відвернути його від них, використовуючи колір, Світло, контраст і текстуру в своїх інтересах [25]-[27].

Використовувати типографіку для створення ієрархії та ясності. Ретельно продумайте, як ви використовуєте шрифт. Різні розміри, шрифти і розташування тексту допомагають підвищити можливість сканування, розбірливість і читаність [25]-[27].

3.5 Висновки за розділом 3

Було проведено ознайомлення з видами структурних діаграм та особливостями їх застосування, розроблено абстрактну модель системи мовою UML на основі структурних діаграм. Розроблено абстрактну модель системи за обраною темою на основі діаграму класів, пакетів та компонентів, на основі діаграм поведінки, прецедентів, діяльності та станів, кооперацій та послідовностей.

4 ЕКСПЛУАТАЦІЯ, ТЕСТУВАННЯ ТА ЕКСПЕРИМЕНТАЛЬНЕ ДОСЛІДЖЕННЯ ПРОГРАМИ

4.1 Призначення й умови застосування програми

Система надає доступ до книг сучасних авторів і зареєстрованих користувачів. Кожному не зареєстрованому користувачеві дається можливість звичайного і розширеного пошуку книг, перегляду короткої інформації про книгу (автор, рік, анотація, рецензія, коментарі). Так само вони можуть читати книгу в он-лайн режимі. Для більш розширених прав користувачі повинні реєструватися.

Зареєстровані користувачі діляться на авторів і читачів.

Автори можуть додавати свої книги, які після перевірки модератором будуть додані в базу даних і видно іншим користувачам. Також вони можуть завантажувати книги, додавати коментарі. Є можливість особистого листування між зареєстрованими користувачами. Читачі можуть все теж, що і автори, крім додавання своїх книг.

Модератори здійснюють контроль над контентом сайту. Вони перевіряють книги на відповідність з жанрами, стежать за авторськими правами на сайті, не допускають дубляжу. Мають право заблокувати сторінки користувачів, які порушують умови сайту. Для видалення таких сторінок модератори повинні звертатися до адміністраторів. Оновлюють книжкові полиці книгами не користувачів сайту.

Адміністратори безпосередньо працюють з базою даних системи.

В системі будуть використовуватися такі такі готові модулі як «Пошук», «Розширений пошук», «Реєстрація», «Оцінка», «Особисте листування».

Веб-сайт забезпечує виконання наступних функцій:

- реєстрація на сайті;
- можливість перегляду контенту;
- додавання нових книг;

- додавання нових користувачів;
- редагування вмісту;
- пошук книг;
- завантаження книг;
- читання он-лайн;
- редагування прав користувача (читач/автор);
- додавання коментаря;
- додавання оцінки добутку;
- сповіщення про нові особисті повідомлення;
- ведення рейтингу книг;
- особиста переписка між користувачами;
- блокування сторінок порушників правил сайту;
- видалення сторінок порушників правил сайту.

4.2 Умови виконання програми

Мінімальні системні вимоги та програмне забезпечення (ПЗ), що необхідні для функціонування серверної частини мобільного застосунку:

- сервер (IBM-сумісна архітектура);
- серверна ОС: Unix або Windows Server;
- інтерпретатор PHP5;
- MySQL-сервер для керування користувачами та організації доступу до БД.

Для експлуатації клієнтської частини необхідні такі програмно-технічні засоби:

- ПК з IBM-сумісною архітектурою;
- процесор із тактовою частотою від 1,7 GHz;
- обсяг оперативної пам'яті – від 3GB (для безперебійної роботи нових версій браузерів);
- підключення до мережі Internet;

– встановлений браузер.

4.3 Інструкція по експлуатації програми

4.3.1 Характеристики програми

Папка `cache` – сюди централізована система кешування поміщає всі попередньо оброблені файли для прискорення видачі сторінок [20], [21].

Папка `administrator` – директорія відповідає за панель управління сайту.

Папка `components` – місце установки компонентів для роботи адміністратора [20], [21].

Папка `includes` – тут зберігаються файли реалізують Application Layer в додатку [20], [21].

Папка `language` – в цій папці зберігаються локалізації панелі управління сайту [20], [21].

Папка `modules` – папка з адміністраторськими модулями такими як `toolbar`, або панель швидкого доступу [20], [21].

Папка `components` – в цій директорії зберігаються файли всіх встановлених в системі компонентів.

Кожен компонент повинен мати певний набір файлів приписаний API і обраної моделлю [20], [21].

Інтерфейс прикладного програмування (іноді інтерфейс програмування додатків) – набір готових класів, функцій, структур і констант, що надаються додатком (бібліотекою, сервісом) для її використання в зовнішніх програмних продуктах [20], [21].

Папка `images` – директорія для зображень і файлів Joomla.

Папка `stories` – в яку зберігаються призначені для користувача зображення [20], [21].

Папка `banners` – тут розташовуються файли банерів для стандартного компонента банери [20], [21].

Папка `M_images` – тут розташовуються іконки і невеликі картинки для сайту.

Папка `includes` – папка з файлами реалізують Application Layer, проте безліч файлів в цій директорії залишені для сумісності зі старою версією Joomla [20], [21].

Папка `language` – локалізації для розширень користувальницького частини сайту [20], [21].

Папка `libraries` – зміст цієї директорії реалізує Framework Layer CMS, в ній зберігається як сам фреймворк Joomla, так і сторонні бібліотеки, які необхідні для роботи [20], [21].

Папка `logs` – файли логів розширень Joomla. Наприклад, сюди поміщаються записи таких подій, як невірний логін на сайт [20], [21].

Папка `modules` – це модулі користувальницької частини сайту .

Папка `plugins` – це обробники подій. Вони викликаються компонентами в певний час генерації контенту , наприклад перед його розміщенням [20], [21].

Папка `templates` – користувацькі шаблони Joomla,

Папка `temp` – папка для тимчасових файлів, наприклад – розпаковані архіви компонентів, тимчасові дані та інше.

Файл `index.php` – головний файл. Саме цей файл запускає Joomla при зверненні користувача до сервера.

Цей же файл запускає програму установки Joomla, якщо вона ще не була встановлена [20], [21].

Файл `templateDetails.xml` – опис шаблону. Налаштування шаблону. Перелік доступних позицій[20], [21].

Файл `component.php` – шаблон виведення компонентів.

Файл `robots.txt` – текстовий файл, який знаходиться в корені сайту і говорить пошуковим роботам як саме індексувати проект [20], [21].

4.3.2Звернення до програми

Веб-сайт доступний за наступною адресою – «<http://chytanka.com/>». Для того щоб зайти на сайт потрібно в адресному рядку браузера ввести адресу сайту, при цьому веб-сервер повинен бути запущений.

4.3.3Вхідні й вихідні дані

Всі дані повинні зберігатися на сервері. Користувачі повинні отримувати всю необхідну інформацію через запити до бази даних.

Книги закріплені за авторами.

Кожна книга повинна мати такі мітки як:

- назва;
- жанр;
- форма;
- року;
- статус (закінчена, редагується);
- рейтинг;
- коментар.

Користувачі повинні мати таку інформацію:

- ПІБ;
- логін;
- псевдонім;
- контактна інформація.

4.4 Методика та результати тестування

Наведемо загальний механізм тестування [28].

Для завантаження сайту і взаємодії з ним потрібно вказати в адресному рядку браузера «<http://chytanka.com/>». Незареєстрований користувач може

тільки переглянути новинну стрічку і зробити по ній пошук, інші дії доступні тільки після реєстрації.

Реєстрація відбувається на сторінці «вхід на сайт» за відповідним посиланням. Після реєстрації можна перейти на сторінку особистого профілю, щоб його відредагувати або залишити як є. Після реєстрації та переходу на сайт користувач бачить головну сторінку, що містить новинну стрічку, останні оновлення розділів і т.п. Пункт «Пошук» дозволяє знайти цікаву інформацію не тільки за назвою запису, але і з текстом по всіх записах. Пункт «Для авторів-початківців» дозволяє користувачі додати свої твори на сайт. На сайті є можливість додавання коментарів до творів. Для ведення рейтингу книг користувачам пропонується оцінювати за шкалою від 1 до 5 балів. Основні сторінки застосунку зображені на рис. 4.1 – 4.5.

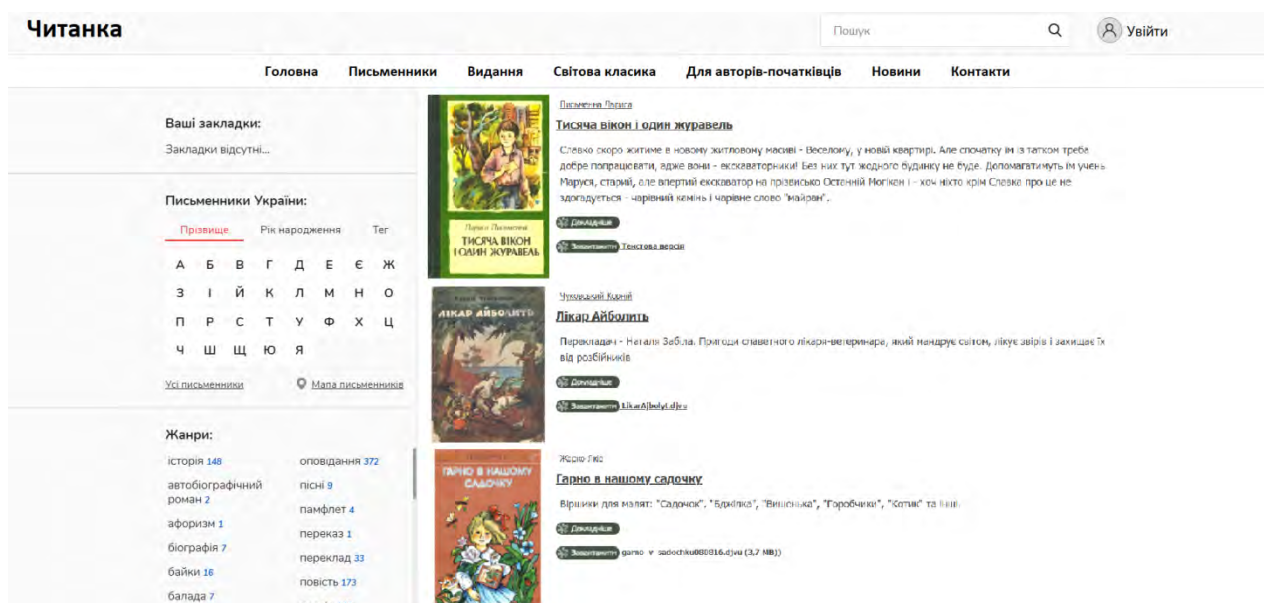


Рисунок 4.1 – Головна сторінка

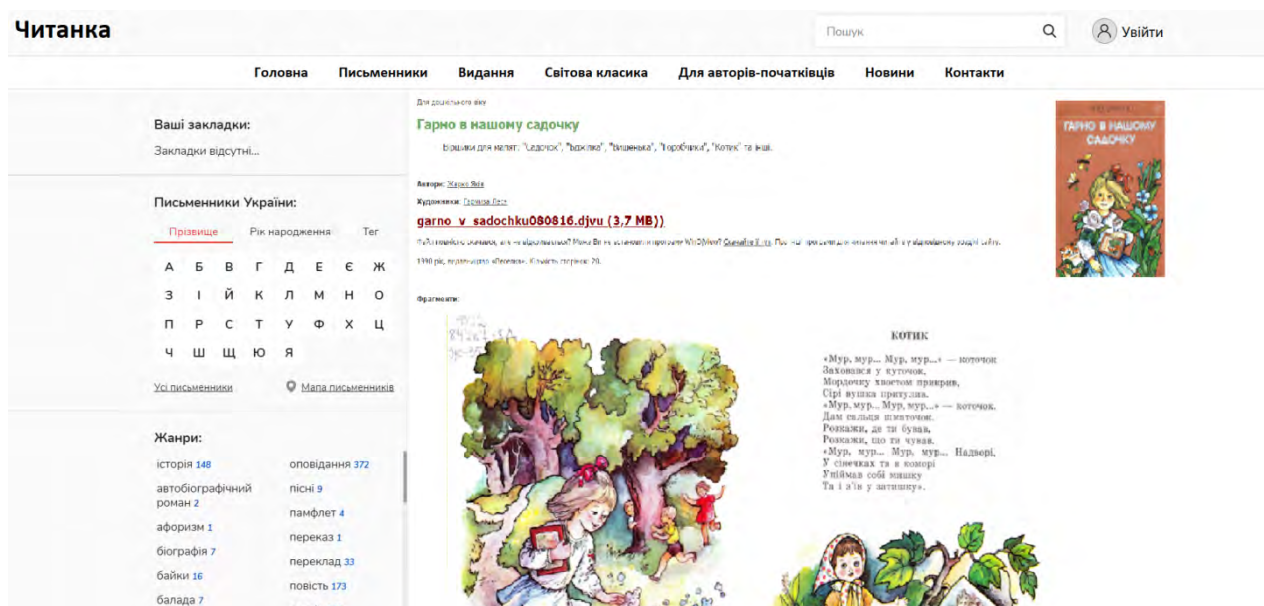


Рисунок 4.2 – Перегляд обраної книги

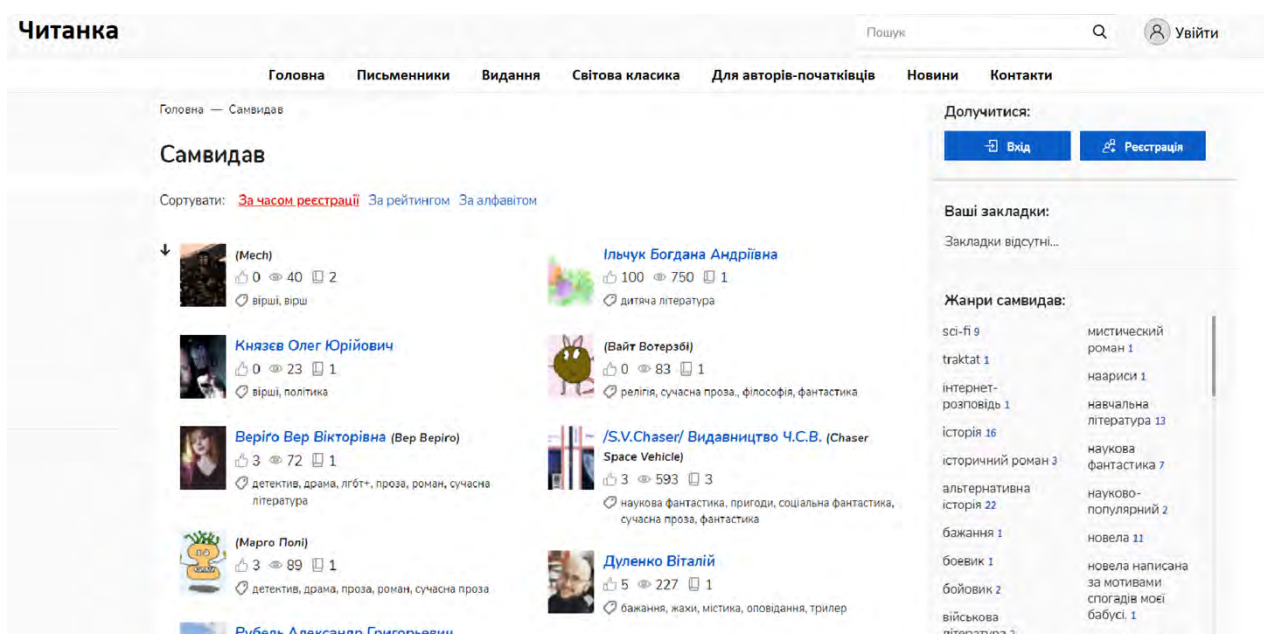


Рисунок 4.3 – Матеріали, що були додані авторами, зареєстрованими в системі

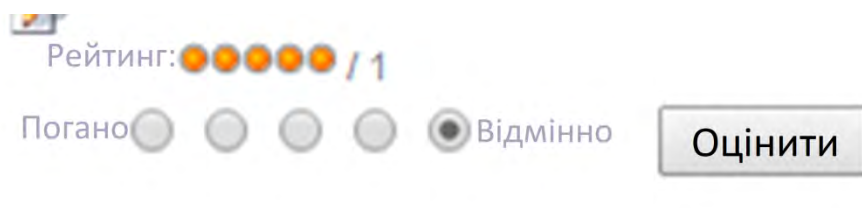


Рисунок 4.4 – Виставлення оцінки твору

Залишити коментар

Переконайтеся, що ви заповнили усі важливі поля (ті, що позначені *).
Введення HTML-коду - заборонено

Повідомлення*

Ім'я* Super User

Пошта* merilist@mail.ru

Залишити коментар

[Назад](#)

Рисунок 4.5 – Вікно для коментарів

ВИСНОВКИ

В ході виконання дипломної кваліфікаційної роботи бакалавра було реалізовано головну мету – дослідити особливості реалізації програмного застосунку електронної бібліотеки. Для цього було виконано ряд завдань:

- охарактеризовано загальні питання організації електронних ресурсів;
- оглянуто принципи взаємодії інтернет-ресурсів та процесу формування каталогів електронних видань;
- відстежено тенденції розвитку організаційно-технологічних основ електронних бібліотек;
- проаналізовано концепції побудови електронних бібліотек та принципи організацій інформаційних ресурсів у них;
- проаналізовано існуючі аналогічні рішення та принципи їх організації;
- акумульовано результати аналізу та досліджень для реалізації програмного застосунку електронної бібліотеки, що якісно відрізняється від існуючих аналогів.

Розроблена структура вебзастосунку була реалізована із використанням CMS Joomla.

Був розроблений сценарій та проведено тестування створеного вебзастосунку за різними напрямками та сценаріями. Отриманий результат свідчить про те, що тестування дало позитивний результат.

В результаті виконання завдання було отримано систему, що відповідає всім вимогам визначеним на початку розробки та закріпленим в технічному завданні. Розроблена система є зручною для використання та виконує всі необхідні функції.

Усі завдання дипломної кваліфікаційної роботи бакалавра повністю виконано.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Аналогова і електронна фотографія: історіографія, характеристика, порівняльний аналіз, сфери застосування в юриспруденції : навч. посіб. / П. Д. Біленчук [та ін.] ; ред. П. Д. Біленчук ; Київський національний ун-т ім. Тараса Шевченка, Київський національний ун-т внутрішніх справ, Навчально-науковий ін-т підготовки слідчих і криміналістів, Європейський ун-т управління, безпеки та інформаційно-правових технологій. – К. : Українська академія підприємництва, 2009. – 40 с.
2. Гарагуля С. С. Електронна бібліотека як інтегратор джерел наукової інформації : автореф. дис. ... канд. наук із соц. комунікацій : 27.00.03 / С. С. Гарагуля; НАН України, Нац. б-ка України ім. В. І. Вернадського. – Київ, 2018. – 19 с.
3. Власов П. Книга друкована і книга електронна: розмова бібліотекаря з книгарем [Електронний ресурс] / П. Власов, І. Журавльова // Бібліотечний вісник. – 2013. – № 3. – С. 16-22.
4. ДСТУ 7157:2010 Інформація та документація. Видання електронні. Основні види та вихідні відомості [Текст]. – Чинний від 2010-07-01. – К. : Держспоживстандарт України, 2010. – IV, 13 с.
5. Українські електронні та паперові енциклопедичні видання: основні здобутки й перспективи : [наук. зб.] / [редкол.: Т. І. Березюк та ін.] ; НАН України, Ін-т енциклопед. дослідж. – Київ : Ін-т енциклопед. дослідж. НАН України, 2015. – 249 с.
6. Киричок Т.Ю. Мережеві електронні видання : довідник / Т. Ю. Киричок, О. І. Лотоцька. – Київ : НТУУ "КПІ", 2016. – 298 с.
7. Любовець Н. Книжкові та електронні біографічні видання: спільне і відмінне, традиції і новаторство [Електронний ресурс] / Н. Любовець, В. Попик // Бібліотечний вісник. – 2008. – № 6. – С. 19-21.

8. Погореловська І. О. Електронні видання: ідентифікаційний аспект [Електронний ресурс] / І. О. Погореловська // Держава та регіони. Соціальні комунікації. – 2013. – № 2. – С. 136-141.
9. Петяк Ю. Ф. Визначення терміну "електронні видання" [Електронний ресурс] / Ю. Ф. Петяк // Поліграфія і видавнича справа. – 2011. – № 2. – С. 184-187.
10. Білоус В. С. Електронна бібліотека – основа інтегрованої технологічної системи інформаційного поля та швидкого доступу до інформації [Текст] / В. С. Білоус // Наукові праці Державної науково-педагогічної бібліотеки України імені В. О. Сухомлинського. – 2014. – Вип. 4. – С. 158-164.
11. Іванова М. Електронна бібліотека та право інтелектуальної власності [Текст] / М. Іванова, Ю. Іванов // Вісник Книжкової палати. – 2018. – № 3. – С. 43-47.
12. Гуревич Р. С. Електронна бібліотека у вищому навчальному закладі [Текст] / Р. С. Гуревич, О. В. Шестопалюк // Сучасні інформаційні технології та інноваційні методики навчання в підготовці фахівців: методологія, теорія, досвід, проблеми. – 2010. – Вип. 26. – С. 3-9.
13. 10 найкращих онлайн-бібліотек з електронними книжками українською мовою [Електронний ресурс] – Режим доступу : <https://megabite.ua/articles/10-najkrashih-onlayn-bibliotek-z-elektronnimi-knijkami-ukrainskou-movou.html>
14. Бібліотека української літератури «УкрЛіб» [Електронний ресурс]. – Режим доступу : <https://www.ukrlib.com.ua/>
15. Українська електронна бібліотека «Libruk» [Електронний ресурс]. – Режим доступу : <https://libruk.com.ua/>
16. Сайт Національної бібліотека України імені В. І. Вернадського [Електронний ресурс]. – Режим доступу : <http://nbuv.gov.ua/>

17. Древич А. І. Реалізації програмних засобів WEB на PHP [Текст] / А. І. Древич, Р. В. Моравецький // Збірник наукових праць Інституту проблем моделювання в енергетиці ім. Г. Є. Пухова. – 2014. – Вип. 71. – С. 54-58.
18. Нефьодов Л. І. Модель вибору PHP-фреймворків для розроблення веб-додатків інформатизації проектної організації [Текст] / Л. І. Нефьодов, Н. Ю. Філь, Д. С. Ковальов // Вісник Харківського національного автомобільно-дорожнього університету. – 2019. – Вип. 87. – С. 74-79.
19. Мельник В. М. Дослідження швидкодії виконання запиту в СУБД MySQL та MariaDB засобами Python та PHP [Текст] / В. М. Мельник, А. І. Нагорнюк, К. Я. Бортник // Комп'ютерно-інтегровані технології: освіта, наука, виробництво. – 2019. – № 36. – С. 123-127.
20. Колисниченко Д. Н. Joomla 1.5.15/1.6.0. Руководство пользователя [Текст] / Д. Н. Колисниченко. – М. ; СПб. ; К. : Диалектика : Вильямс, 2010. – 240 с.
21. AboutJoomla! [Електронний ресурс]. – Режим доступу : <https://www.joomla.org/about-joomla.html>
22. Постіл С. Д. UML. Уніфікована мова моделювання інформаційних систем : навч. посіб. / С. Д. Постіл. – Ірпінь : Ун-т держ. фіск. служби України, 2019. – 321 с.
23. Богдан І. Верифікація моделей об'єктно-орієнтованих програм: перевірка на несуперечливість та узгодженість [Текст] / І. Богдан // Техн. науки та технології. – 2017. – № 2. – С. 110-115.
24. Гудзовата О. О. Використання уніфікованої мови візуального моделювання UML (Unified Modeling Language) як інструменту підтримки проектування інформаційних систем [Текст] / О. О. Гудзовата, В. І. Костирко, І. В. Артищук // Підприємництво і торгівля. – 2019. – Вип. 24. – С. 108-114.
25. Злепко С. М. Принципи організації інтерфейсу користувача в інформаційній системі тестування персоналу [Текст] / С. М. Злепко, К. С. Навроцька // Вісник Хмельницького національного університету. Технічні науки. – 2015. – № 2. – С. 169-172.

26. Данчук В. Д. Побудова моделей бізнес-процесів колцентру, як інтегрованої частини компанії e-commerce та розробка на їх основі концептуальної архітектури та прототипів інтерфейсу користувача. [Текст] / В. Д. Данчук, Д. В. Луцюк // Вісник [Національного транспортного університету]. – 2013. – № 27. – С. 389-395.

27. Інформаційні технології. Інтерфейси користувача. Універсальна віддалена консоль [Текст]. – Київ : УкрНДНЦ, 2019 . – (Національний стандарт України). Ч. 4 : ДСТУ ISO/IEC 24752-4:2018 (ISO/IEC 24752-4:2014, IDT). Опис цілі. – Чинний від 2020-01-01. – 2019. – V, 15 с.

28. Коваленко О. В. Імітаційна модель технології тестування безпеки на основі положень теорії масштабування [Текст] / О. В. Коваленко // Безпека інформації. – 2018. – 24, № 2. – С. 110-117.

ДОДАТОК А
Технічне завдання

Вступ

Система має надавати доступ до книг, що зберігаються в її БД. Вона надає можливість пошуку книг, перегляду їх анотації, читання книг (онлайн, скачування), завантаження своїх книг.

Також система має надавати можливість особистого листування між зареєстрованими користувачами, додавання коментарів і оцінок до книг.

Система повинна вести рейтинг найпопулярніших книг.

Система підтримує можливість реєстрації користувачів в системі.

A.1 Підстава для розробки

Підставою для розробки є завдання на дипломну кваліфікаційну роботу на тему «Реалізація програмного застосунку електронної бібліотеки», затверджене наказом Національного університету «Запорізька політехніка» № 102 від 27квітня 2022 р.

A.2 Основні вимоги до програми, що розробляється

Метою проекту є створення електронної бібліотеки, надання доступу до книг сучасних авторів розважальної художньої літератури для ознайомлення. Система-електронна бібліотека з можливістю додавати свої книги. В системі можна реєструватися, завантажувати свої твори, завантажувати книги, проводити пошук, оцінювати інші твори.

A.2.1 Вимоги до функціональних характеристик

Система повинна забезпечити:

- реєстрація на сайті;
- можливість перегляду контенту;

- додавання нових книг;
- додавання нових користувачів;
- редагування вмісту;
- пошук книг;
- завантаження книг;
- читання он-лайн;
- редагування прав користувача (читач/автор);
- додавання коментаря;
- додавання оцінки добутку;
- сповіщення про нові особисті повідомлення;
- ведення рейтингу книг;
- особиста переписка між користувачами;
- блокування сторінок порушників правил сайту;
- видалення сторінок порушників правил сайту.

A.2.2 Вимоги до інтерфейсу програми

Інтерфейс системи повинен бути ергономічним, інтуїтивно-зрозумілим для користувачів системи. Вся інформація сайту повинна бути чітко структурована і розділена на розділи.

Розділ пошуку повинен мати настільки простий інтерфейс, наскільки можливо зробити без втрати його багатофункціональності.

Сторінки реєстрації та авторизації Користувача повинні мати підказки з приводу заповнення форм.

A.2.3 Вимоги до надійності

Права користування системою залежать від статусу користувача

Повні права доступу маю адміністратори. Вони практично можуть управляти всією інформацією сайту

Модератори оновлюють систему новими поповненнями книг. Також вони перевіряють книги на відповідність короткої інформації та її змісту.

Автори можуть додавати свої книги

Читачі, як і всі зареєстровані користувачі, можуть завантажувати книги, додавати коментарі, ставити оцінки.

Незареєстровані користувачі, як і зареєстровані, можуть дивитися коротку інформацію про книгу, читати онлайн, дивитися коментарі, інформацію про автора.

Шифрування паролів буде проводитися методом AES.

A.2.4 Умови експлуатації

Для користування додатком, користувач повинен мати девайс з доступом до мережі Інтернет, обладнаний веб-браузером. Для створення власного сховища користувач має зареєструватися.

A.2.5 Вимоги до складу та параметрів технічний засобів

Мінімальні системні вимоги та програмне забезпечення (ПЗ), що необхідні для функціонування серверної частини мобільного застосунку:

- сервер (IBM-сумісна архітектура);
- серверна ОС: Unix або Windows Server;
- інтерпретатор PHP5;
- MySQL-сервер для керування користувачами та організації доступу до БД.

Для експлуатації клієнтської частини необхідні такі програмно-технічні засоби:

- ПК з IBM-сумісною архітектурою;
- процесор із тактовою частотою від 1,7 GHz;

- обсяг оперативної пам'яті – від 3GB (для безперебійної роботи нових версій браузерів);
- підключення до мережі Internet;
- встановлений браузер.

A.2.6 Вимоги до маркування і пакування

Програма для роботи електронного щоденника здобувача освітиможе бути записана на будь-якому носії інформації. На пакуванні повинна бути назва програми – «Читанка».

A.3 Вхідні дані до роботи

Всі дані повинні зберігатися на сервері. Користувачі повинні отримувати всю необхідну інформацію через запити до бази даних.

Книги закріплені за авторами.

Кожна книга повинна мати такі мітки як:

- назва;
- жанр;
- форма;
- року;
- статус (закінчена, редагується);
- рейтинг;
- коментар.

Користувачі повинні мати таку інформацію:

- ПІБ;
- логін;
- псевдонім;
- контактна інформація.

ДОДАТОК Б
Опис програми

Б.1 Загальні відомості

Система має надавати доступ до книг, що зберігаються в її БД. Вона надає можливість пошуку книг, перегляду їх анотації, читання книг (онлайн, скачування), завантаження своїх книг. Також система має надавати можливість особистого листування між зареєстрованими користувачами, додавання коментарів і оцінок до книг. Система повинна вести рейтинг найпопулярніших книг. Система підтримує можливість реєстрації користувачів в системі.

Метою проекту є створення електронної бібліотеки, надання доступу до книг сучасних авторів розважальної художньої літератури для ознайомлення. Система-електронна бібліотека з можливістю додавати свої книги. В системі можна реєструватися, завантажувати свої твори, завантажувати книги, проводити пошук, оцінювати інші твори.

Б.2 Функціональне призначення

Система повинна забезпечити:

- реєстрація на сайті;
- можливість перегляду контенту;
- додавання нових книг;
- додавання нових користувачів;
- редагування вмісту;
- пошук книг;
- завантаження книг;
- читання он-лайн;
- редагування прав користувача (читач/автор);
- додавання коментаря;
- додавання оцінки добутку;
- сповіщення про нові особисті повідомлення;

- ведення рейтингу книг;
- особиста переписка між користувачами;
- блокування сторінок порушників правил сайту;
- видалення сторінок порушників правил сайту.

Б.3 Опис логічної структури

На рисунку Б.1 наведено структурну схему ПЗ, що розробляється.

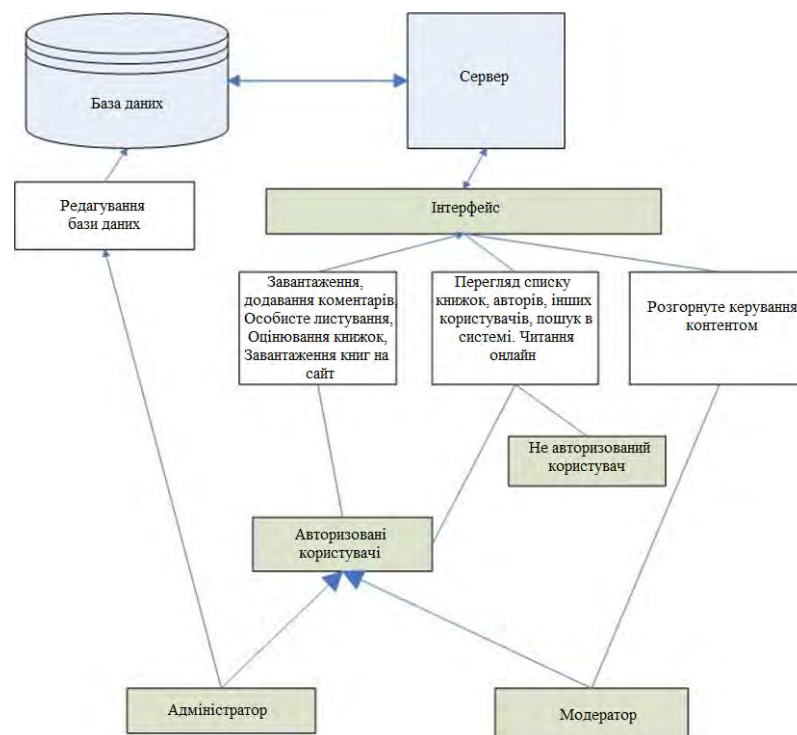


Рисунок Б.1 – Загальна схема системи за обраною темою

Управління контентом буде полягати у виконанні наступних функцій:

- додавання нових книг;
- додавання нових користувачів;
- редагування книг користувачів;
- видалення книг;
- видалення сторінок порушників правил сайту;
- реєстрація на сайті;

- додавання нових книг;
- редагування книг користувачів;
- додавання нових користувачів;
- редагування вмісту;
- пошук книг;
- завантаження книг;
- читання он-лайн;
- редагування прав користувача (читач / автор);
- додавання коментаря;
- додавання оцінки добутку;
- сповіщення про нові особисті повідомлення;
- ведення рейтингу книг;
- особиста переписка між користувачами;
- блокування сторінок порушників правил сайту;
- видалення сторінок порушників правил сайту.

Структура веб-застосунку, що розробляються з використанням Drupal зображена на рисунку Б.2.

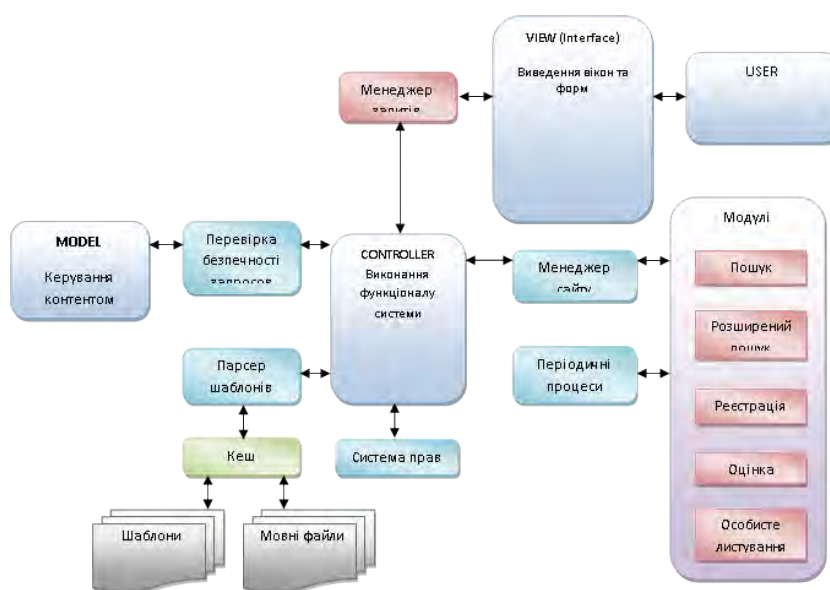


Рисунок Б.2 – Архітектура веб-застосунку

Б.4 Використані технічні засоби

Мінімальні системні вимоги та програмне забезпечення (ПЗ), що необхідні для функціонування серверної частини мобільного застосунку:

- сервер (IBM-сумісна архітектура);
- серверна ОС: Unix або Windows Server;
- інтерпретатор PHP5;
- MySQL-сервер для керування користувачами та організації доступу до БД.

Для експлуатації клієнтської частини необхідні такі програмно-технічні засоби:

- ПК з IBM-сумісною архітектурою;
- процесор із тактовою частотою від 1,7 GHz;
- обсяг оперативної пам'яті – від 3GB (для безперебійної роботи нових версій браузерів);
- підключення до мережі Internet;
- встановлений браузер.

Б.5 Виклик і завантаження

Веб-сайт доступний за наступною адресою – «<http://chytanka.com/>». Для того щоб зайти на сайт потрібно в адресному рядку браузера ввести адресу сайту, при цьому веб-сервер повинен бути запущений.

Б.6 Вхідні та вихідні дані

Всі дані повинні зберігатися на сервері. Користувачі повинні отримувати всю необхідну інформацію через запити до бази даних.

Книги закріплені за авторами.

Кожна книга повинна мати такі мітки як:

- назва;
- жанр;
- форма;
- року;
- статус (закінчена, редагується);
- рейтинг;
- коментар.

Користувачі повинні мати таку інформацію:

- ПІБ;
- логін;
- псевдонім;
- контактна інформація.

ДОДАТОК В
Текст програми

```

<?php
if(!isset($_SESSION)){
    session_start();
}
require_once("DBconn.php");

if(isset($_SESSION['logged'])){
if($_SESSION['logged']=='1'){
header("location:adminPage/adminAuth.php");

}
}

<head><title>ELIMS::Читанка</title>
<link rel="shortcut icon" href="pcts/icon.ico" />
<link rel="stylesheet" type="text/css" href="css/divs.css">
<link rel="stylesheet" type="text/css" href="css/bootstrap.css">
<script type="text/javascript" src="js/jquery.js"></script>
<script type="text/javascript" src="js/my_code.js"></script>

<link href="css/js-image-rotor.css" rel="stylesheet" type="text/css" />
<script src="js/js-image-rotor.js" type="text/javascript"></script>
<link href="generic.css" rel="stylesheet" type="text/css" />

<style>
#login{
margin-left:60px;
}
</style>

```

```
</head>
```

```
<body>
```

```
<div class="body1">
```

```
<div class="menubox">
```

```
<h4 class="loginh4" align="center">Меню</h4>
```

```
<form action="checkLogin.php" method="post">
```

```
<p class="error">
```

```
<?php
```

```
if(isset($_SESSION['complain'])){
```

```
$complain=$_SESSION['complain'];
```

```
echo $complain;
```

```
$_SESSION['complain']="";
```

```
}
```

```
?></p>
```

```
<input type="text" call_name="uName" id="uName1" placeholder="User  
call_name" />
```

```
<input type="pass" call_name="pass" id="pass1" placeholder="pass" />
```

```
<input type="submit" class="btn btn-success" id="login" value="login"/>
```

```
</form>
```

```
</div>
```

```
<div class="banner">
<h1>ELIMS<br></h1>
```

```
<h2>E-Library Information Management System</h2>
```

```
</div>
```

```
<?php
```

```
/*Adds the locations in the sub menu*/
```

```
?>
```

```
<ul id="menu">
```

```
<li><a href="index.php">Home</a></li>
```

```
<li>
```

```
<a href="site.php">Site</a>
```

```
<ul>
```

```
<li><a href="#">Eastern DR</a>
```

```
<ul>
```

```
<?php
```

```
while($row = mysql_fetch_array($edr)) {
```

```
$dist1=$row['dist'];
```

```
$siteEDR=mysql_query("SELECT * FROM site_maps
WHERE dist='$dist1'") or die(mysql_error());
```

```
while($rowEDR = mysql_fetch_array($siteEDR)) {
```

```

                                echo                                '<li><a
href="site.php?action='.$rowEDR['call_name'].'">'.$rowEDR['call_name'].'</a></li
>';

                                }

                                }

                                ?>

                                </ul>
                                </li>
                                <li><a href="#">Читанка</a>
                                <ul>

                                <?php

                                while($row = mysql_fetch_array($cdr)) {
                                $dist1=$row['dist'];
                                $siteDB=mysql_query("SELECT  *  FROM  site_maps
WHERE dist='$dist1'") or die(mysql_error());
                                while($rowDB = mysql_fetch_array($siteDB)) {
                                    $siteChytanka=$rowDB['call_name'];
                                    echo                                '<li><a
href="site.php?action='.$siteChytanka.'">'.$siteChytanka.'</a></li>';

                                    }

                                }

                                ?>

```

```

        </ul>

        </li>

        <li><a href="#">Western DR</a>

        <ul>

        <?php

        while($row = mysql_fetch_array($wdr)) {
            $dist1=$row['dist'];
            $siteWDR=mysql_query("SELECT * FROM site_maps
WHERE dist='$dist1'") or die(mysql_error());
            while($rowWDR = mysql_fetch_array($siteWDR)) {
                $siteChytanka=$rowWDR['call_name'];
                echo                                '<li><a
href="site.php?action='.$siteChytanka.'">'.$siteChytanka.'</a></li>';

                }

            }

        ?>

        </ul>

        </li>

        <li><a href="#">Читанка</a>

        <ul>

```

```

<?php

while($row = mysql_fetch_array($mwdr)) {
    $dist1=$row['dist'];
    $siteMWDR=mysql_query("SELECT * FROM site_maps
WHERE dist='$dist1'") or die(mysql_error());
    while($rowMWDR = mysql_fetch_array($siteMWDR)) {
        $siteChytanka=$rowMWDR['call_name'];
        echo '
        <li><a
href="site.php?action='.$siteChytanka.'">'.$siteChytanka.'</a></li>';

    }

}

?>
</ul>

</li>
<li><a href="#">Читанка</a>

<ul>

<?php

while($row = mysql_fetch_array($fwdr)) {
    $dist1=$row['dist'];
    $siteChyt=mysql_query("SELECT * FROM site_maps
WHERE dist='$dist1'") or die(mysql_error());

```

```

while($rowFWDR = mysql_fetch_array($siteChyt)) {
    $siteChytanka=$rowFWDR['call_name'];
    echo
    ' <li><a
href="site.php?action='.$siteChytanka.'">'.$siteChytanka.'</a></li>';

}

}

?>
</ul>

</li>

</ul>

</li>

</ul>

</li>

<li><a href="#">Читанка</a>
    <ul>
<li><a href="#">О нас</a></li>
<li><a href="#">Автори</a></li>
<li><a href="#">Видання</a></li>
<li><a href="#">Для молодих авторів</a></li>
</ul>
</li>

<li><a href="#">About</a></li>
<li><a href="#">Contact</a></li>

```

```
</ul>
```

```
<?php                                /*End of the drop down menu*/
```

```
?>
```

```
<div class="contentBox">
```

```
<?php /* rotor Part*/?>
```

```
<div id="rotorFrame">
```

```
<div id="rib"></div>
```

```
<div id="rotor">
```

```
<a href="#">
```

```

```

```
</a>
```

```
<a class="lazyImage" href="pcts/2.jpg" title="Helping">Customizable  
Transition Effects</a>
```

```
<a href="#">
```

```
<b data-src="pcts/3.jpg" data-alt="Serving">Image rotor</b>
```

```
</a>
```

```
<a class="lazyImage" href="pcts/4.jpg" title="Changing Lives">Plain  
Javascript rotor</a>
```

```
</div>
```

```
<div style="display: none;">
```

```
</div>
```

```

<!--tthumbnails-->
<div id="tthmbs">
<div class="tthmb"></div>
<div class="tthmb"></div>
<div class="tthmb"></div>
<div class="tthmb"></div>

</div>
</div>

</div>
<?php /*-----rotor Part ends-----*/?>

</div>

<div class="footer">
</div>
</body>
</html>
<joomla:post_id>82</joomla:post_id>
    <joomla:post_date>2022-03-22 21:27:34</joomla:post_date>
    <joomla:post_date_gmt>2022-03-22
17:27:34</joomla:post_date_gmt>
    <joomla:comment_status>open</joomla:comment_status>
    <joomla:ping_status>closed</joomla:ping_status>
    <joomla:post_name>need-for-speed-most-
wanted</joomla:post_name>
    <joomla:status>publish</joomla:status>
    <joomla:post_parent>0</joomla:post_parent>

```

```

    <joomla:menu_order>0</joomla:menu_order>
    <joomla:post_type>product</joomla:post_type>
    <joomla:post_password></joomla:post_password>
    <joomla:is_sticky>0</joomla:is_sticky>
    <category                                domain="product_tag"
nicename="race"><![CDATA[Race]]></category>
    <category                                domain="product_type"
nicename="simple"><![CDATA[simple]]></category>
    <joomla:postmeta>
        <joomla:meta_key>_edit_last</joomla:meta_key>

<joomla:meta_value><![CDATA[1]]></joomla:meta_value>
    </joomla:postmeta>
    <joomla:postmeta>
        <joomla:meta_key>_thumbnail_id</joomla:meta_key>

<joomla:meta_value><![CDATA[83]]></joomla:meta_value>
    </joomla:postmeta>
    <joomla:postmeta>
        <joomla:meta_key>_visibility</joomla:meta_key>

<joomla:meta_value><![CDATA[visible]]></joomla:meta_value>
    </joomla:postmeta>
    <joomla:postmeta>
        <joomla:meta_key>_stock_status</joomla:meta_key>

<joomla:meta_value><![CDATA[instock]]></joomla:meta_value>
    </joomla:postmeta>
    <joomla:postmeta>
        <joomla:meta_key>total_sales</joomla:meta_key>

```

```

<joomla:meta_value><![CDATA[0]]></joomla:meta_value>
  </joomla:postmeta>
  <joomla:postmeta>
    <joomla:meta_key>_downloadable</joomla:meta_key>

```

```

<joomla:meta_value><![CDATA[yes]]></joomla:meta_value>
  </joomla:postmeta>
  <joomla:postmeta>
    <joomla:meta_key>_virtual</joomla:meta_key>

```

```

<joomla:meta_value><![CDATA[yes]]></joomla:meta_value>
  </joomla:postmeta>
  <joomla:postmeta>
    <joomla:meta_key>_regular_price</joomla:meta_key>

```

```

<joomla:meta_value><![CDATA[290]]></joomla:meta_value>
  </joomla:postmeta>
  <joomla:postmeta>
    <joomla:meta_key>_sale_price</joomla:meta_key>

```

```

<joomla:meta_value><![CDATA[]]></joomla:meta_value>
  </joomla:postmeta>
  <joomla:postmeta>
    <joomla:meta_key>_purchase_note</joomla:meta_key>

```

```

<joomla:meta_value><![CDATA[]]></joomla:meta_value>
  </joomla:postmeta>
  <joomla:postmeta>
    <joomla:meta_key>_featured</joomla:meta_key>

```

```

<joomla:meta_value><![CDATA[no]]></joomla:meta_value>
  </joomla:postmeta>
  <joomla:postmeta>
    <joomla:meta_key>_weight</joomla:meta_key>

```

```

<joomla:meta_value><![CDATA[]]></joomla:meta_value>
  </joomla:postmeta>
  <joomla:postmeta>
    <joomla:meta_key>_length</joomla:meta_key>

```

```

<joomla:meta_value><![CDATA[]]></joomla:meta_value>
  </joomla:postmeta>
  <joomla:postmeta>
    <joomla:meta_key>_width</joomla:meta_key>

```

```

<joomla:meta_value><![CDATA[]]></joomla:meta_value>
  </joomla:postmeta>
  <joomla:postmeta>
    <joomla:meta_key>_height</joomla:meta_key>

```

```

<joomla:meta_value><![CDATA[]]></joomla:meta_value>
  </joomla:postmeta>
  <joomla:postmeta>
    <joomla:meta_key>_sku</joomla:meta_key>

```

```

<joomla:meta_value><![CDATA[]]></joomla:meta_value>
  </joomla:postmeta>
  <joomla:postmeta>

```

<joomla:meta_key>_product_attributes</joomla:meta_key>

<joomla:meta_value><![CDATA[a:0:{}]]></joomla:meta_value>

</joomla:postmeta>

<joomla:postmeta>

<joomla:meta_key>_sale_price_dates_from</joomla:meta_key>

<joomla:meta_value><![CDATA[]]></joomla:meta_value>

</joomla:postmeta>

<joomla:postmeta>

<joomla:meta_key>_sale_price_dates_to</joomla:meta_key>

<joomla:meta_value><![CDATA[]]></joomla:meta_value>

</joomla:postmeta>

<joomla:postmeta>

<joomla:meta_key>_price</joomla:meta_key>

<joomla:meta_value><![CDATA[290]]></joomla:meta_value>

</joomla:postmeta>

<joomla:postmeta>

<joomla:meta_key>_sold_individually</joomla:meta_key>

<joomla:meta_value><![CDATA[]]></joomla:meta_value>

>

ДОДАТОК Д
Слайди презентації



Сокур Є.О.
ст. групи КНТ-119сп
Субботін С.О.
д.т.н., професор, завідувач кафедри ПЗ

Рисунок Д.1 – Слайд 1



Об'єкт, предмет та мета роботи

- **Об'єкт дослідження** – процес реалізація програмного застосунку електронної бібліотеки.
- **Предмет дослідження** – програмні засоби для реалізації програмного застосунку електронної бібліотеки.
- **Метою роботи є** реалізація програмного застосунку електронної бібліотеки у вигляді кроссплатформного вебзастосунку.

Рисунок Д.2 – Слайд 2

Постановка завдання

Метою проєкту є реалізація програмного застосунку електронної бібліотеки.

Завдання роботи:

- характеристика загальних питань організації електронних ресурсів;
- огляд принципів взаємодії інтернет-ресурсів та процесу формування каталогів електронних видань;
- відстеження тенденцій розвитку організаційно-технологічних основ електронних бібліотек;
- аналіз концепції побудови електронних бібліотек та принципів організації інформаційних ресурсів у них;
- аналіз існуючих аналогічних рішень та принципів їх організації;
- акумулювання результатів аналізу та досліджень для реалізації програмного застосунку електронної бібліотеки, що якісно відрізняється від існуючих аналогів.

3

Рисунок Д.3 – Слайд 3

Визначені функціональні вимоги

Система повинна забезпечити:

- реєстрація на сайті;
- можливість перегляду контенту;
- додавання нових книг;
- додавання нових користувачів;
- редагування вмісту;
- пошук книг;
- завантаження книг;
- читання он-лайн;
- редагування прав користувача (читач/автор);
- додавання коментаря;
- додавання оцінки добутку;
- сповіщення про нові особисті повідомлення;
- ведення рейтингу книг;
- особиста переписка між користувачами;
- блокування сторінок порушників правил сайту;
- видалення сторінок порушників правил сайту.

4

Рисунок Д.4 – Слайд 4

Порівняння існуючих аналогів

Критерії порівняння	УкрЛіб	Libruk	Національна бібліотека України імені В. І. Вернадського	Читанка
Представлення електронного каталогу та алфавітного вказівника	Так	Так	Ні (на сайті представлені ресурси)	Так
Можливість опублікувати власні роботи	Ні	Так	Ні	Так
Рейтингування видань	Ні	Так	Ні	Так
Функції онлайн-читання обраних творів	Так	Так	Ні	Так

5

Рисунок Д.5 – Слайд 5

Порівняння мов програмування

Критерій	PHP	Python	JavaScript
Простота розробки	++	++	++
Навантаженість синтаксису	+	+	++
Механізми для спрощення розробки	++	+	++
Подальша масштабованість	++	±	+
Програмні надбудови	++	+	+

6

Рисунок Д.6 – Слайд 6

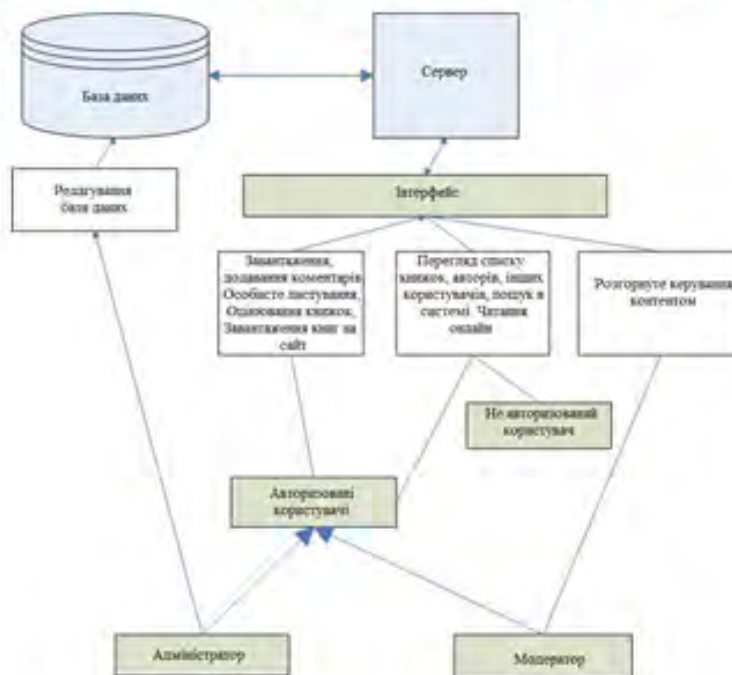
Порівняння CMS

	Drupal	WordPress	Joomla!
Вимоги до серверу та хостингу	++	+	++
Стабільність та регулярність оновлень	+	++	++
Масштабованість на наступна підтримка застосунків	+	++	++
Взаємодія із системами керування БД (СКБД)	+	+	++

7

Рисунок Д.7 – Слайд 7

Загальна схема системи за обраною темою



8

Рисунок Д.8 – Слайд 8

Структура веб-застосунку, що розробляється з використанням Joomla



Рисунок Д.9 – Слайд 9

Проектування моделей системи

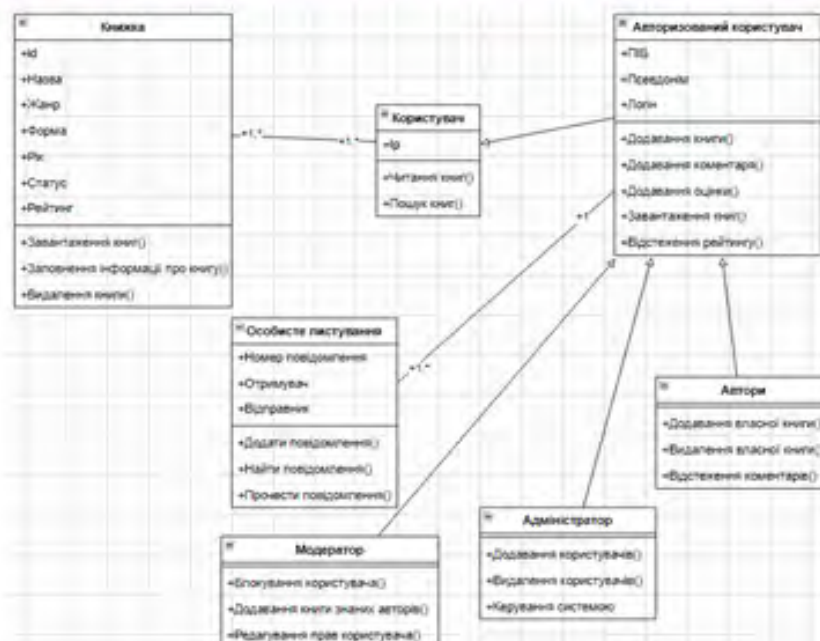
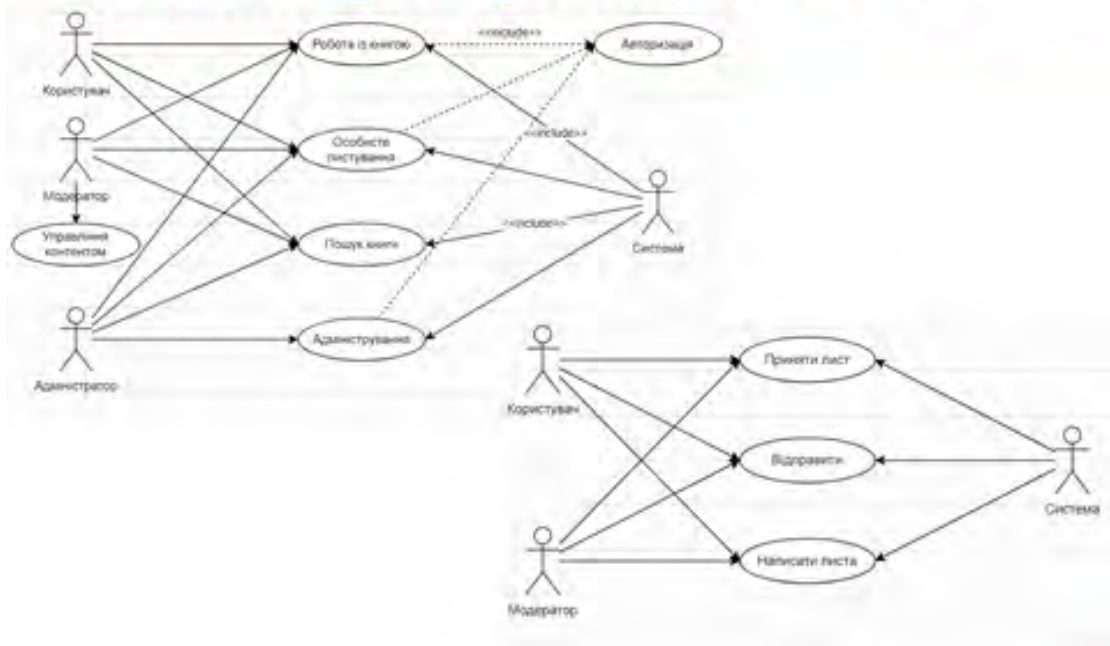


Рисунок Д.10 – Слайд 10

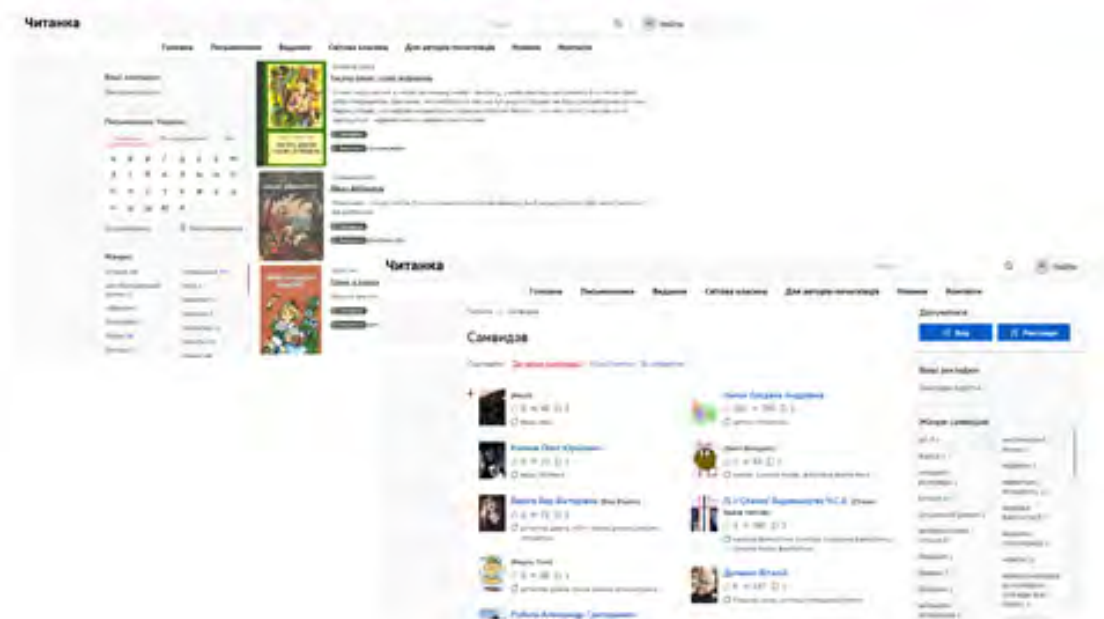
Логічне проектування взаємодії



11

Рисунок Д.11 – Слайд 11

Інтерфейс застосунку



12

Рисунок Д.12 – Слайд 12

Висновки

В ході виконання дипломної кваліфікаційної роботи бакалавра було реалізовано головну мету – дослідити особливості реалізації програмного застосунку електронної бібліотеки. Для цього було виконано ряд завдань:

- охарактеризовано загальні питання організації електронних ресурсів;
- оглянуто принципи взаємодії інтернет-ресурсів та процесу формування каталогів електронних видань;
- відстежено тенденції розвитку організаційно-технологічних основ електронних бібліотек;
- проаналізовано концепції побудови електронних бібліотек та принципи організацій інформаційних ресурсів у них;
- проаналізовано існуючі аналогічні рішення та принципи їх організації;
- акумулювано результати аналізу та досліджень для реалізації програмного застосунку електронної бібліотеки, що якісно відрізняється від існуючих аналогів.

В результаті виконання завдання було отримано систему, що відповідає всім вимогам представленим в технічному завданні, зручна для використання та виконує всі необхідні функції.

Усі завдання випускної дипломної роботи бакалавра повністю виконано.

Рисунок Д.13 – Слайд 13