

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

Національний університет кораблебудування імені адмірала Макарова
Навчально-науковий інститут комп'ютерних наук та управління проектами
Кафедра інформаційних управляючих систем та технологій

"Допущений до захисту"

Завідувач кафедри ІУСТ



к.т.н, доц. Михелєв І.Л.

" ____ " _____ 2025 р.

КВАЛІФІКАЦІЙНА РОБОТА

на здобуття ступеня вищої освіти "бакалавр"

на тему: «Розробка інформаційної системи з WEB-доступом для муніципальної бібліотеки»

Виконав: студент групи 4141


(підпис)

Тимофєєва Н.В..

Керівник роботи:

Викладач

(посада, науковий ступінь вчене звання)


(підпис)

Книрик К.О.

м. Миколаїв – 2025 р.

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

Національний університет кораблебудування імені адмірала Макарова
Навчально-науковий інститут комп'ютерних наук та управління проектами
Кафедра інформаційних управляючих систем та технологій
Спеціальність 122 "Комп'ютерні науки"
Освітня програма "Комп'ютерні науки"

"ЗАТВЕРДЖУЮ"

Гарант освітньої програми

к.т.н, доц. Гайда А.Ю.

" ____ " _____ 2025 р.

ЗАВДАННЯ НА КВАЛІФІКАЦІЙНУ РОБОТУ на здобуття ступеня вищої освіти "бакалавр"

Студенту Тимофєєвій Наталії Вікторівни

(прізвище, ім'я, по батькові)

1. Тема роботи «Розробка інформаційної системи з WEB-доступом для муніципальної бібліотеки»

Керівник роботи Книрик К.О.

Затверджені наказом ректора № 295-уч від "27" березня 2025 року.

2. Термін подання роботи: 20 червня 2025 р.

3. Вихідні дані до проекту (роботи) ДСТУ щодо обробки інформації, літературні джерела, технічна документація на існуючі аналоги систем,

4. Перелік питань, що належать до розробки (найменування розділів) Перелік питань, що належать до розробки (найменування розділів) _____

- Титульний аркуш, завдання на кваліфікаційну роботу, анотація (українською та англійською мовами), зміст, перелік умовних позначень, символів, одиниць та термінів (при необхідності)
- ВСТУП
- ОПИС ПРЕДМЕТНОЇ ГАЛУЗІ ДЛЯ АВТОМАТИЗАЦІЇ ДІЯЛЬНОСТІ БІБЛІОТЕКИ
- ПРОЕКТУВАННЯ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ ІНФОРМАЦІЙНОЇ СИСТЕМИ З WEB-ДОСТУПОМ ДЛЯ МУНІЦИПАЛЬНОЇ БІБЛІОТЕКИ (ескізний, технічний та робочий проекти)
- РЕЗУЛЬТАТИ РОЗРОБКИ •
- ОХОРОНИ ПРАЦІ
- ВИСНОВКИ
- СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ
- ДОДАТКИ (технічне завдання, текст програми, інструкція користувача)

5. Перелік презентаційних матеріалів

Титульний слайд, Мета роботи, Аналоги програмних продуктів, Контекстна діаграма інформаційної системи для муніципальної бібліотеки DFD0; Контекстна діаграма інформаційної системи для муніципальної бібліотеки DFD1; Діаграма варіантів використання системи (use-case), Концептуальна модель даних, Логічна модель даних, Результати розробки програмного забезпечення сайт бібліотеки, Приклад коду для сайту, Висновки

6. Консультанти розділів роботи

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв
1	Книрік К.О., викладач	03.02.2025	22.02.2025
2	Книрік К.О., викладач	22.02.2025	21.03.2025
3	Книрік К.О., викладач	21.03.2025	25.04.2025
4	Книрік К.О., викладач	25.04.2025	07.06.2025

7. Дата видачі завдання 03 лютого 2025 р.

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів дипломного проекту (роботи)	Строк виконання етапів роботи	Примітка
1	Аналіз предметної галузі	13.02.2025	
2	Постановка задачі	15.02.2025	
3	Розробка концепції	22.02.2025	
4	Розробка проекту ІС	21.03.2025	
5	Реалізація проекту	25.04.2025	
6	Розробка питань з охорони праці	07.06.2025	
7	Розробка графічних матеріалів	12.06.2025	
8	Подання роботи на перевірку на плагіат	10.06.2025	
9	Подання роботи рецензенту	15.06.2025	
10	Подання роботи на захист	20.06.2025	

Студент


(підпис)

Тимофєєва Н.В.
(ІПБ)

Керівник роботи


(підпис)

Книрік К.О.
(ІПБ)

АНОТАЦІЯ

Кваліфікаційна робота присвячена розробці інформаційної системи з WEB-доступом для муніципальної бібліотеки.

Програмне забезпечення дозволяє автоматизувати облік книжок та процес обслуговування читачів в бібліотеці, за допомогою WEB-сайту. Робота містить проект програмного забезпечення, розділ з охорони праці, а також технічне завдання на розробку програмного забезпечення, текст програми, програму та методику випробувань, інструкцію користувача, опис програм.

Робота викладена українською мовою на 93 сторінках друкованого тексту, містить 8 таблиць, 19 рисунка, 3 додатків та список використаних джерел з 23 найменувань.

Ключові слова:

Бібліотека, книга, твір, автор, опис, література, жанр, письменник, резервування, сайт, користувач, база даних, розробка.

ABSTRACT

The qualification work is devoted to the development of an information system with WEB access for a municipal library.

The software allows you to automate the accounting of books and the process of servicing readers in the library, using a WEB site. The work contains a software project, a section on labor protection, as well as a technical task for the development of software, the text of the program, the program and testing methodology, user manual, description of programs.

The work is presented in Ukrainian on 93 pages of printed text, contains 8 tables, 19 images, 3 appendices and a list of sources used from 23 titles.

Keywords:

Library, book, work, author, description, literature, genre, writer, reservation, website, user, database, development.

ЗМІСТ

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ, СКОРОЧЕНЬ І ТЕРМІНІВ	7
ВСТУП	9
РОЗДІЛ 1. ОПИС ПРЕДМЕТНОЇ ГАЛУЗІ ДЛЯ АВТОМАТИЗАЦІЇ ДІЯЛЬНОСТІ БІБЛІОТЕКИ.....	10
1.1. Опис предметної галузі муніципальної бібліотеки	10
1.2. Підхід до програмного забезпечення для автоматизації діяльності бібліотеки	13
1.3. Аналіз існуючих аналогів.....	14
1.4. Постановка задачі на розробку програмного забезпечення	16
Висновки	18
РОЗДІЛ 2. ПРОЕКТУВАННЯ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ ІНФОРМАЦІЙНОЇ СИСТЕМИ З WEB-ДОСТУПОМ ДЛЯ МУНІЦИПАЛЬНОЇ БІБЛІОТЕКИ	19
2.1. Ескізний проект ПЗ інформаційної системи з WEB-доступом для муніципальної бібліотеки.....	19
2.1.1. Функціональна модель системи	19
2.2. База даних	24
2.2.1. Розробка концептуальної моделі.....	24
2.2.2. Логічна модель	29
2.2.3. Фізична модель.....	30
2.4. Створення веб-сайту	33
2.4.1. Мета створюваного сайту	33
2.4.2. Визначення концепції сайту та структури сайту	34
2.4.3. Малювання блокової структури сторінок	34
2.4.4. Розробка дизайну та верстка сторінок	36
Висновки	38

РОЗДІЛ 3. РЕЗУЛЬТАТИ РОЗРОБКИ ПРОГРАМНОГО ПРОДУКТУ ДЛЯ МУНІЦИПАЛЬНОЇ БІБЛІОТЕКИ	39
3.1. Результати розробки програмного коду	39
3.1.1. Програмні файли	39
3.1.2. HTML-сторінки	39
3.1.3. Розроблений програмний код	50
3.2. Результати тестування програми	50
Висновки	51
РОЗДІЛ 4. ОХОРОНА ПРАЦІ	52
4.1. Аналіз небезпечних і шкідливих факторів	52
4.2. Розробка заходів для зменшення впливу шкідливих факторів	58
4.3. Ергономіка робочого місця	61
4.4. Режим праці	64
Висновки	65
ВИСНОВКИ	66
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ	67
ДОДАТОК А. ТЕХНІЧНЕ ЗАВДАННЯ НА РОЗРОБКУ «РОЗРОБКА ІНФОРМАЦІЙНОЇ СИСТЕМИ З WEB-ДОСТУПОМ ДЛЯ МУНІЦИПАЛЬНОЇ БІБЛІОТЕКИ»	70
ДОДАТОК Б. КОД ПРОГРАМИ	77
ДОДАТОК В. ІНСТРУКЦІЮ КОРИСТУВАЧА	86

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ, СКОРОЧЕНЬ І ТЕРМІНІВ

А/м - ампер на метр.

АІС - автоматична ідентифікаційна система (англ. Automatic Identification System, AIS).

В/м - вольт на метр.

Гц - герц (позначається Гц, Hz) - одиниця вимірювання в системі SI частоти періодичних процесів (н-д, коливань).

дБ, - децибел (українське позначення: дБ, міжнародне: dB) — популярна одиниця вимірювання рівня гучності, підсилення потужності сигналу або власне потужності сигналу, значення якої дорівнює одній десятій бела.

Діапазон радіочастот: **ВЧ, УВЧ і СВЧ** - радіохвилі — діапазон електромагнітних хвиль з довжиною хвилі від 10–5 до 1010 метра.

ЕКГ - електрокардіограма

ІКТ - це технології опрацювання інформації за допомогою комп'ютера та телекомунікаційних засобів.

ІС - інформаційна система (англ. Information system) — сукупність організаційних і технічних засобів для збереження та обробки інформації з метою забезпечення інформаційних потреб користувачів.

кВ/м - кіловольт на метр.

КІСП - комп'ютерна інформаційна система підприємства.

лк - люкс (позначення: лк, міжнародне: lx, від лат. lux - «світло») - одиниця вимірювання освітленості в Міжнародній системі одиниць (SI). 1 люкс - це освітленість світловим потоком 1 лм, рівномірно розподіленим по поверхні площею 1 м².

ЛПО - люмінесцентні світильники

МБ - Мегабайт (megabyte) — одиниця вимірювання обсягу даних. Рівна 1 048 576 (220) байт або 1024 кілобайт.

мкбер/год - радіаційний фон:

мкВт/м² - щільності потужності ультрафіолетового й інфрачервоного випромінювань.

ООП - об'єктно-орієнтоване програмування (англ. Object-oriented programming, OOP).

ПЕОМ - узагальнена назва групи засобів обчислювальної техніки

Пристрій CD-RW - перезаписуваний компакт-диск - різновид компакт-диску (CD).

СУБД - система управління базами даних (СУБД, СКБД англ. Database Management System, DBMS).

ERP - система (англ. enterprise resource planning system) - Система планування ресурсів підприємства)

UML (Unified Modeling Language) - уніфікована мова візуального моделювання

ВСТУП

Мета роботи – розроблення інформаційної системи з веб-доступом для муніципальної бібліотеки, автоматизація інформаційно-бібліотечної діяльності, надання онлайн-доступу читачам до бібліотечного фонду, бронюванню книг що допоможе зробити роботу бібліотекарів більш швидкою та ефективною і підніме рівень обслуговування читачів у бібліотеці на більш якісний рівень.

Головним завданням муніципальної бібліотеки є надання якісних бібліотечних інформаційних послуг.

Робота бібліотек ґрунтується на однакових принципах в різних країнах, а саме накопичення та систематизація бібліотечного фонду, обслуговування читачів та популяризація літератури серед населення.

Інформаційно-освітній простір бібліотеки є динамічно організованим середовищем, створеним в результаті спільної діяльності всіх структурних підрозділів бібліотеки і часто за участю громадських організацій. Тому застосування широкого спектру новітніх інформаційних технологій підвищує рівень роботи бібліотеки.

Створення сучасного інформаційно простору бібліотеки і розширення спектра бібліотечних послуг передбачає:

- автоматизацію основних бібліотечних процесів;
- наповнення інформаційного простору бібліотеки (інформаційно ресурсна база бібліотеки);
- створення віртуального інформаційного простору бібліотеки (вебсайту);
- технічне та комунікаційне оснащення бібліотеки.

Щоб розв'язати ці завдання знадобився комплексний підхід під час проектування та реалізації проекту.

РОЗДІЛ 1. ОПИС ПРЕДМЕТНОЇ ГАЛУЗІ ДЛЯ АВТОМАТИЗАЦІЇ ДІЯЛЬНОСТІ БІБЛІОТЕКИ

1.1. Опис предметної галузі муніципальної бібліотеки

Предметна область даного проекту – бібліотека.

Муніципальні бібліотеки є популярними культурно-освітніми закладами, які потребують достатньо об'ємної документації.

В бібліотеці ведеться облік книг, та обслуговування читачів. Основне завдання бібліотеки надання якісних послуг читачам бібліотеки. Діяльність бібліотек в Україні регулюється наступними законодавчими актами: «Про інформацію»[14], «Про бібліотеки та бібліотечну справу»[13], «Про Національний архівний фонд та архівні установи»[15], «Про стандартизацію»[20].

Головною матеріальною цінністю в бібліотеці є книжковий фонд який складається з різноманітної друкованої продукції, а саме: книг різних жанрів та іноді інших матеріалів.

Для задоволення попиту у бібліотеці зазвичай зберігається більше одного екземпляру однієї й тієї ж книги. Крім того одна книга може бути виготовлені різними видавництвами в різний час – в такому випадку кількість сторінок та дата видання може відрізнитись в залежності від екземпляру. Тому в даних про екземпляр книги зберігають наступне:

- 1) Ідентифікатор книги;
- 2) Назва;
- 3) Автори;
- 4) Жанр;
- 5) Видавництво;
- 6) Рік видання;
- 7) Кількість сторінок;
- 8) Дата видання;
- 9) Ціна екземпляру книги.

При обслуговуванні читачів для пошуку книг працівники бібліотеки можуть використовувати назву книги, автора, назву твору (якщо у книзі більше одного твору), жанр, опис й т.д.. Й працівники бібліотеки часто шукають книги за деякими параметрами.

Важливо зазначити, що одна книга може мати більш, ніж одного автора чи жанра та містити один чи більше творів. По кожному з цих пунктів ведеться облік.

Стосовно авторів записують: прізвище, ім'я, ім'я по батькові, якщо воно є. Жанрів: їх назву й іноді опис. Творів: назву, автора(ів), короткий опис.

Важливим моментом для обслуговування читачів є й їх облік. Він ведеться починаючи з реєстрації в бібліотеці. Для цього до формуляру читача заносять наступні дані:

- 1) Номер читача (паспорта);
- 2) Прізвище;
- 3) Ім'я;
- 4) Домашня адреса;
- 5) Електронна адреса;
- 6) Номер телефону.

Працівник бібліотеки немає права видавати книги незареєстрованим читачам. Й кожна видача повинна бути задокументована.

Книжки в бібліотеці видаються на термін 15 днів, після чого їх необхідно повернути до бібліотеки, у іншому разі читача запишуть в боржники й не будуть видавати інші книжки до повернення вже виданих. У випадку втрати книги читач зобов'язаний виплатити її подвійну вартість, так як книжки для бібліотеки є матеріальною цінністю. Також, якщо це можливо, знайти аналогічний екземпляр. Регулярно один раз на рік в бібліотеці проводять інвентаризацію книжкового фонду.

Читачам дозволяється брати не більше 5 екземплярів різних книг (не можна брати декілька екземплярів однієї й тієї ж книги) за один раз.

У разі коли екземпляру необхідної книги немає в наявності, читач може стати за нею в чергу. Коли в бібліотеці з'явиться необхідний екземпляр читачу подзвонять та проінформують про наявність необхідної літератури в бібліотеці.

Коли книгу видають чи заносять читача в чергу, то у бібліотеці зазначають:

- книгу (назву, код, авторів), читача (прізвище, ім'я, телефон), дату видачі/внесення в чергу. Коли книгу повертають, то дописують ще й дату повернення.

В роботі муніципального бібліотекаря і досі залишається багато рутинних операцій пов'язаних з паперовими носіями, що має ряд недоліків:

- при заповненні документів можуть виникати помилки, у зв'язку з чим може виникати плутаниця для виправлення якої витрачається багато часу;
- часто інформація в різних документах дублюється, щоб усе було зрозуміло, але через це інформація займає багато місця, швидкість її обробки/поповнення уповільнюється, а у наслідку й обслуговування читачів;
- читачі обмежені у роботі з інформацією, через нестачу часу, та неавтоматизований пошук.

Для обробки статистичних цифрових даних використовуються такі програми як Word та Excel, що мають майже аналогічні проблеми. Також форми для занесення даних не завжди зручні і потребують додаткового часу при користуванні ними.

1.2. Підхід до програмного забезпечення для автоматизації діяльності бібліотеки

Сучасне інформаційне суспільство потребує якісної та швидкої роботи з інформацією. Важливим напрямком розвитку бібліотеки є автоматизація саме технологічних бібліотечних процесів, а саме обслуговування читачів за їх запитамі. Це дає можливість зробити роботу бібліотекаря більш ефективною за рахунок усунення рутинних операцій та поліпшити комунікацію з читачами.

Вивчивши предметну галузь, в даному разі це муніципальна бібліотека маємо такі результати, що потребують вирішення:

- робота в бібліотеці з паперовими формулярами та носіями має ряд недоліків;
- читачі обмежені у роботі з інформацією.

Інформаційна технологія сучасної бібліотеки ґрунтується насамперед на АБІС[6] (автоматичні бібліотечні інформаційні системи), що забезпечує аналітико-синтетичне опрацювання та представлення користувачам як традиційного (паперового) документного фонду бібліотеки, так і електронних інформаційних ресурсів, обслуговує бібліотечні фонди на всіх етапах – від придбання до замовлення.

Зазвичай АБІС складається з реляційної бази даних, програмного забезпечення, що взаємодіє з базою даних, і двох графічних користувацьких інтерфейсів (один – для читачів, другий – для персоналу). Окремі функції програмного забезпечення більшості АБІС є функціонально завершеними модулями, об'єднаними в загальний інтерфейс. Оперативний доступ до інформаційних ресурсів АБІС передбачає надання читачам таких можливостей:

- 1) доступ до електронного каталогу;
- 2) доступ до електронної інформації, що міститься в базах даних бібліотеки; віддалене замовлення літератури;
- 3) користування електронною поштою.

1.3. Аналіз існуючих аналогів

ПЗ для бібліотек призначено для підтримки ієрархічних класифікацій і тезаурусів, надавати можливість бібліотекам здійснювати обмін (експорт/імпорт) бібліографічними записами, створювати електронні бібліотеки, вести каталог читачів, забезпечувати захист бази даних від несанкціонованого доступу та в разі аварійної ситуації, відновлювати дані.

В бібліотеках України використовується різне програмне забезпечення: MAPK-SQL, ІРБІС, ALEPF, УФД/Бібліотека, UNILIB і т. ін.

Майже все ці програмні продукти складаються з однакових функціональних модулів, а саме:

- «Комплектування», «Каталогізатор», «Читач», «Книговидача», WEB - модуль, «Адміністратор».

АІБС "МАРК-SQL"

Комплексна автоматизована інформаційно-бібліотечна система, призначена для автоматизації бібліотечної діяльності. Повний технологічний цикл бібліотечно-бібліографічної роботи від замовлення літератури до видачі читачеві.

Принципові особливості АІБС "МАРК-SQL":

- АІБС "МАРК-SQL" функціонує в архітектурі клієнт-сервер.
- АІБС «МАРК-SQL» розроблялася на основі принципу відкритих систем і має наступні властивості:
 - мультизадачність;
 - розширюваність;
 - масштабованість;
 - мобільність/переносність;
 - здатність до взаємодії з іншими інформаційними до системами;
 - дружність до користувача;
 - модульність структури;

<<Liber>>

Розробку комерційного варіанту системи здійснила французька фірма "Relais Informatique International" у 1987 році.

В АБІС "Liber" автоматизовано всі основні бібліотечні функції. За основу бібліотечної системи "Liber" взято операційну систему РІСК. Ця система спрямована працювати з базами даних. Наразі ОС РІСК немає широкого поширення, що можна розцінювати як недолік системи. З метою пристосування АБІС "Liber" К сучасним вимогам фірма-розробник постачає спеціальну оболонку, яка функціонує під керуванням UNIX.

Ще одним недоліком системи "Liber" є її закритість. З системою не постачається необхідна документація для розробки користувачами нових програмних функціональних модулів, тому користувач, придбавши "Liber", змушений проводити модернізацію, замовляючи за окрему плату програми у фірми-розробника або її партнера.

Система "Liber" є достатніми для автоматизації університетських бібліотек, але недоліки що перераховані вище не дозволяють рекомендувати її для застосування в публічних бібліотеках. Використання даної АБІС можливе, але малоперспективне з причин зазначених вище.

IRBIS: бренд, виробник, дистриб'ютор IRBIS

IRBIS – російська торгова марка, під якою випускаються планшети, ноутбуки, смартфони та комп'ютерна техніка для дому та офісу. Торгова марка IRBIS належить компанії Treolan, що входить до складу ГК Ланіт - однієї з найбільших груп компаній на російському ринку інформаційних технологій.

ІРБІС64

Автоматизована бібліотечно-інформаційна система (АБІС) забезпечує створення автоматизованих робочих місць для традиційних бібліотечних процесів (комплектування, науково-технічна обробка, облік фондів, видача та прийом книг і т.п.) і створення ефективної системи подання інформації маційних ресурсів для користувачів.

І якщо технічні процеси в цій програмі реалізовані на високому рівні то подання інформації не завжди задовольняє користувача.

У багатьох існуючих АБІС модуль надання ресурсів читачам реалізовано у вигляді спеціального додатка. Таке рішення обмежує сферу застосування модуля читальною залю бібліотеки, в той час як технологія веб-доступу дозволяє практично будь-якому користувачеві з будь-якого комп'ютера отримати доступ до ресурсів через Інтернет.

Інший недолік існуючих підходів до організації електронних ресурсів полягає у використанні лінійної моделі електронного каталогу. У лінійній моделі доступу електронний каталог представляється таблицею записів, пошукові запити до якої формуються за певними критеріями. Графова модель електронного каталогу відрізняється від лінійної наявністю системи навігації між індивідуальними ресурсами або їх групами, що дозволяє користувачеві знаходити потрібні ресурси швидше, роблячи роботу з електронним каталогом більш ефективними.

Загалом перераховані готові системи є іноземними. Українських розробок в даному напрямку фактично не має. Тому робота над створення власної програми для автоматизації бібліотек є корисною.

1.4. Постановка задачі на розробку програмного забезпечення

За результатом опису предметної галузі муніципальної бібліотеки та її аналізу є можливість створити інформаційну систему з WEB-доступом та розробити програмне забезпечення для її функціонування.

Мета роботи – розроблення інформаційної системи з веб-доступом для муніципальної бібліотеки, тобто сайту бібліотеки, щоб зробити більш зручним користування бібліотечним фондом та надати можливість бронювання книг

читачам. Працівникам бібліотеки це допоможе покращити обслуговування та зробити роботу більш швидкою та ефективною. Створення самого веб-сайту передбачає роботу з базою даних, сеанси, автентифікацію тощо. Автоматизація процесу обслуговування читачів в муніципальній бібліотеці покликана забезпечити:

- усунути дублювання даних в документах (зменшити їх об'єм);
- багатоцільове їх використання;
- пошук даних за різними ознаками;
- автоматизацію комплектування фонду бібліотеки;
- автоматичне формуванням бронювання і автоматичний їх облік їх виконання;
- автоматизацію ведення бібліотечної статистики.

Електронний варіант ведення документації допоможе позбутися перерахованих у бібліотеці недоліків і таким чином покращити показники ефективності праці персоналу бібліотеки та підвищити рейтинг даного культурно-освітнього закладу.

Програмне забезпечення, що розроблено в кваліфікаційній роботі:

- повинно вільно розповсюджуватись;
- повинно виконувати необхідний перелік функцій для даної категорії бібліотек;
- бути незалежним від інших програмних продуктів (окрім операційної системи).

Розроблена інформаційна система дозволить перейти до ведення обліку у електронному вигляді

Вимоги до «Інформаційної системи з веб-доступом для муніципальної бібліотеки», що розроблена в рамках кваліфікаційної роботи наведено у технічному завданні (додаток А).

Висновки

У першому розділі каліфіційної роботи проведено аналіз предметної галузі, знайдені вивчені аналоги подібних програмних продуктів, поставлені цілі та завдання для створення інформаційної системи муніципальної бібліотеки.

РОЗДІЛ 2. ПРОЕКТУВАННЯ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ ІНФОРМАЦІЙНОЇ СИСТЕМИ З WEB-ДОСТУПОМ ДЛЯ МУНІЦИПАЛЬНОЇ БІБЛІОТЕКИ

Для виконання поставленої задачі при розробці програмного забезпечення інформаційної системи з WEB-доступом для муніципальної бібліотеки була проведена робота з ескізного, технічного та робочого проектування.

2.1. Ескізний проект ПЗ інформаційної системи з WEB-доступом для муніципальної бібліотеки

Для створення ескізного проекту програмного забезпечення була вивчена та проаналізована предметна галузь муніципальної бібліотеки. В технічному завданні на створення інформаційної системи описані вимоги до результатів виконання програми. В залежності від цих вимог був розроблений загальний підхід та схема функціонування системи, структури вхідних і вихідних даних, були визначені дії користувача ПЗ та стан і дії програми при відповіді на них, розроблений інтерфейс для WEB-доступу.

2.1.1. Функціональна модель системи

Функціональну модель створена для оцінки функцій та динаміки системи, визначення модулів, що входять до її складу та функціонування програмного забезпечення в цілому.

Контекстна діаграма ПЗ інформаційної системи для муніципальної бібліотеки

Демонстрацією функціонування моделі є набір діаграм потоків даних. На ній демонструється як вхідні дані перетворюються у вихідні, при цьому не розглядається порядок чи спосіб реалізації перетворень.

Діаграма потоків даних забезпечує візуальне уявлення того, як потоки даних (інформації) надходять від зовнішніх (по відношенню до процесу) суб'єктів і переробляються в виконання процесу, зберігаються, передаються зовнішнім (по відношенню до процесу) суб'єктам.

На рисунку 2.1 представлена контекстна діаграма інформаційної системи для муніципальної бібліотеки

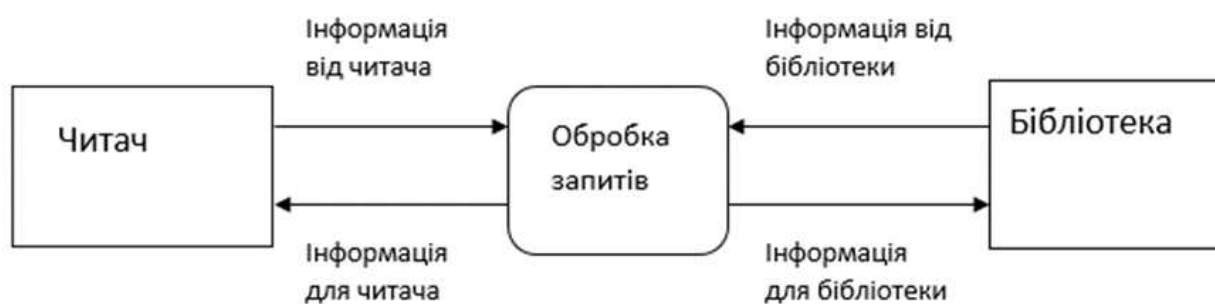


Рис. 2. 1 DFD0

За результатами попереднього етапу - збору вимог можна виділити основні функції системи, типи учасників взаємодії і приблизний набір інформаційних даних, які в ній циркулюють.

На рис. 2.2 зображено діаграму декомпозиції процесу «Обробка запитів». Перед початком обробки запиту на літературу перевіряється наявність авторизації користувача (процес «Перевірка даних клієнта»). Перевірка авторизації користувача відбувається шляхом перевірки вводу логіна користувача на сайті. Після підтвердження права користувача йому надається можливість резервувати книги доступні у системі. Читач здійснює вибір

необхідної літератури. У відповідь система надає список доступної літератури та описи для заданого жанру, користувач із наданого списку вибирає книгу. У відповідь система надає можливість зарезервувати книгу для її отримання у бібліотеці на протязі 15 днів пвсля цього (процес «Запит на резервування»).

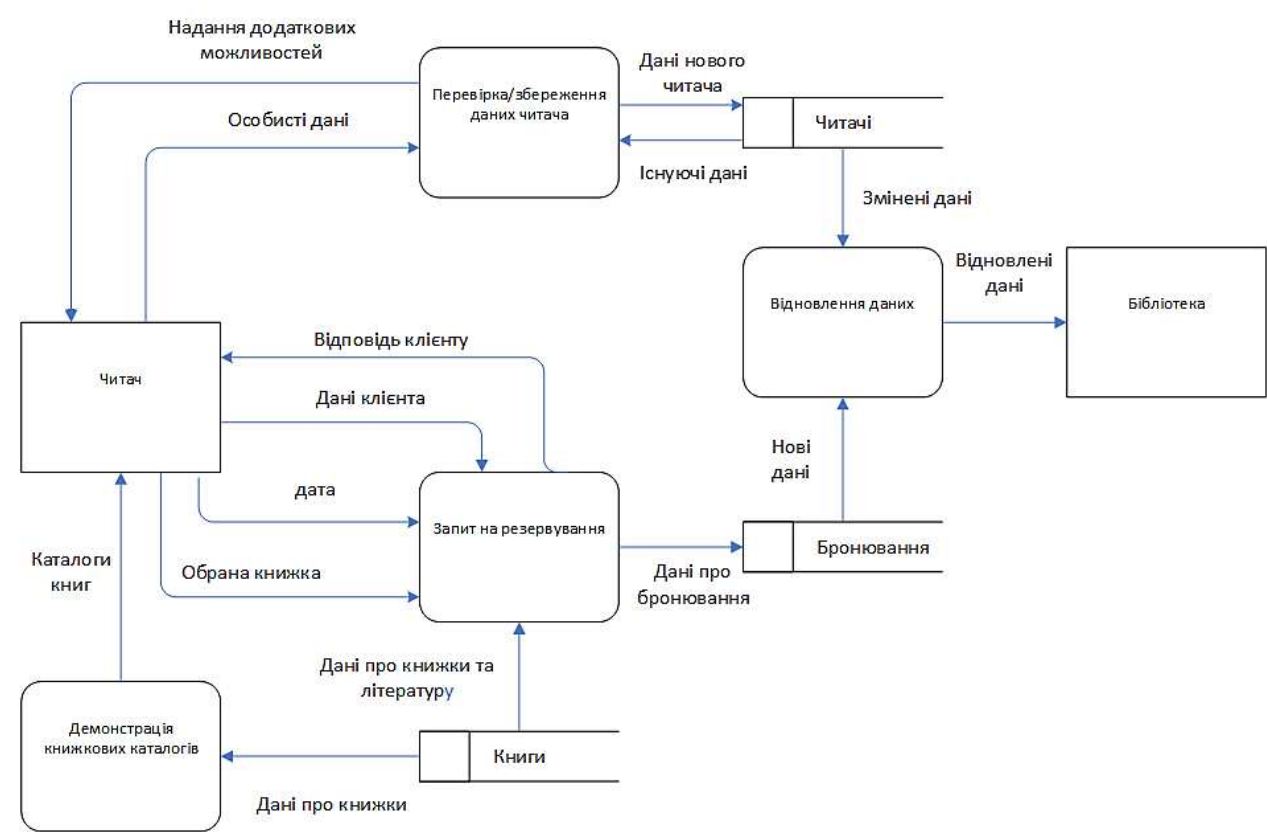


Рис. 2. 2 DFD1

У таблиці 2.1 приведена відповідність потоків даних й атрибутів.

Таблиця 2. 1

Відповідність потоків даних й атрибутів

Потоки даних DFD0	Потоки даних DFD1	Атрибути
Інформація від клієнта	Дані клієнта	Інформація про клієнта.
	Обрані книжки	Номер обраного ресторану.
	Дата	Дата зробленого резервування.
	Особисті дані	Особиста інформація клієнта (ПІБ, адреса, телефон, емейл, пароль).

Інформація для клієнта	Каталоги книг	Списки книжок (усі або за критеріями).
	Відповідь клієнту	Чи є книга в наявності, чи клієнта занесли у лист очікуючих на книгу. Терміни, коли книгу можна забрати з бібліотеки.
	Надання додаткових можливостей	Якщо надані клієнтом особисті дані проходять перевірку, то клієнт буде мати більше прав у системі.
Інформація від бібліотеки		Дані з системи (наявні книги, списки користувачів, співробітників, тощо).
Інформація для бібліотеки	Відновлені дані	Змінення у базі даних та системі.

UML-діаграма

Розглянемо вимоги до функціоналу системи з боку користувачів.

На рис. 2.3 наведено діаграму варіантів використання (use-case) системи.

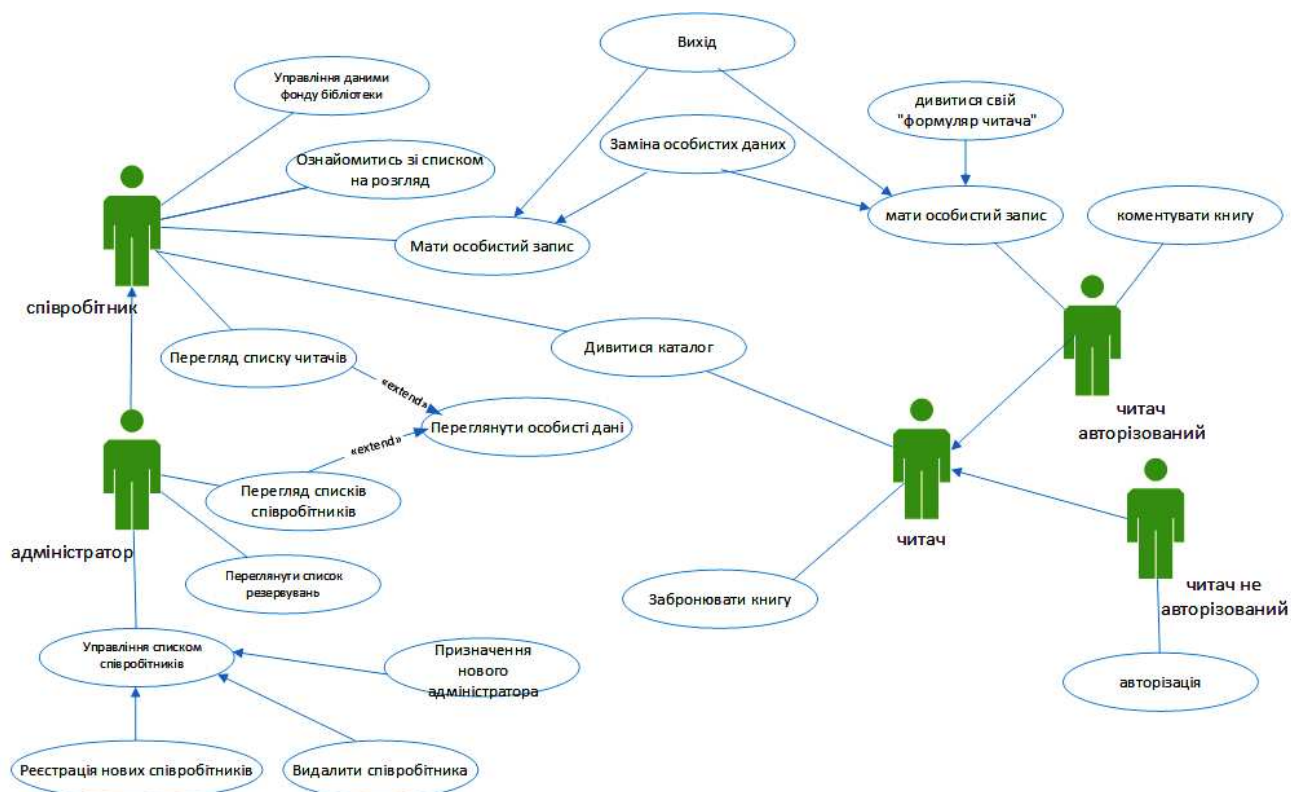


Рис. 2. 3 UML-діограма «Use-case»

Згідно з діаграмою в системі будуть виділені наступні користувачі:

- Клієнт (авторизований, неавторизований);
- співробітник (звичайний, адміністратор).

Найменший набір прав у неавторизованого користувача. Згідно з вимогами, такий користувач не має доступу до резервування літератури. У межах системи неавторизований користувач має доступ лише до функції реєстрації/входу та перегляду інформації про книги (пошук йому теж доступний). Після реєстрації користувач набуває статусу авторизованого користувача. Авторизований користувач є основним учасником системи. Такий користувач має можливість розпочати роботу з системою (увійти в систему) та закінчити роботу (вийти із системи). Для ідентифікації користувача достатньо даних обов'язкових полів реєстрації – логіна та пароля. Для реалізації основної потреби авторизований користувач використовує функції вибору потрібної книги та отримання доступу до читачького кабінету з усіма його функціями та можливістю бронювання книжок.

Функціональна роль співробітника бібліотеки: додавати та списувати книжки, управляти фондом бібліотеки, видавати та контролювати повернення книжок від читачів, контролювати список на резервування книжок.

Функціональна роль адміністратора. Адміністратор має право як працівник бібліотеки виконувати всі дії що і інші співробітники та має додаткові обов'язки (створює облікові записи працівників та контролює роботу співробітників). Також адміністратор має можливість блокування (видалення) акаунтів несумлінних користувачів.

Загальні вимоги до системи, виділені раніше в основні функціональні групи:

- Процеси забезпечення ідентифікації користувача:
 - реєстрація;
 - вхід.

- Процеси обслуговування читача:
 - надання списку доступної літератури;
 - надання описів літератури;
 - надання можливості написання коментарів до книжок
- Процес управління бібліотечною документацією:
 - додавання даних книги;
 - видалення книги;
 - редагування/додавання опису книги;
 - редагування списку співробітників.

2.2. База даних

Сайт передбачає використання бази даних для збереження та легкого доступу до актуальної інформації, уникнення дублювання даних та їх пошуку.

2.2.1. Розробка концептуальної моделі

На базі проведеного аналізу предметної галузі за допомогою потоків даних та діаграми варіантів використання системи, спочатку створюється концептуальна модель даних інформаційної системи для муніципальної бібліотеки. Цей етап являється обов'язковим при проектуванні програмного забезпечення.

При концептуальному проектуванні як основна модель використовується модель типу "сутність зв'язок" (ER-модель). Спочатку визначають сутності та їх атрибути. Це реалізовано в таблиці 2.2.

Таблиця 2. 2

**Сутності та атрибути предметної області інформаційної системи
муніципальної бібліотеки**

Назва	Значення	Тип	Властивість	Приклад
Author				
ID		Int	IDENTIFIANT	1
LAST_NAME	Прізвище автора	Varchar	Not null	Роулінг
FIRST_NAME	Ім'я автора	Varchar	Not null	Джоан
Genre				
ID		Int	IDENTIFIANT	1
TITLE	Назва жанру	Varchar	Not null	Роман
Literary_Work				
ID		Int	IDENTIFIANT	1
IMG	Посилання на зображення	Text		//https:...
TITLE	Назва твору	Varchar	Not null	Гаррі Поттер
DESCRIPTION	Опис твору	Text		Історія про хлопчика, який вижив.
Publishing_House				
ID		Int	IDENTIFIANT	2
NAME	Назва видавництва	Varchar	Not null	А-ба-ба-га-ла-ма-га
Book				
ID		Int	IDENTIFIANT	4
IMG	Обкладинка книги (посилання на зображення)	Text		//https:...
TITLE	Назва книги	Varchar	Not null	Гаррі Поттер і філософський камінь
DESCRIPTION	Опис книги	Text		Історія про хлопчика, який вижив. Частина перша.
YEAR	Рік випуску	Year	Not null	2023
LANGUAGE	Мова книги	Varchar	Not null	Українська

LANG_ORIG	Мова оригіналу	Varchar	Not null	Англійська
NB_PAGES	Кількість сторінок	Int	Not null, >0	168
NB_EX	Кількість екземплярів	Int	Not null, >=0	2000
PRIX	Ціна	Money	Not null, >0	489
POSITION	Розташування книги в бібліотеці	Int	Not null,	1
Status				
ID		Varchar	IDENTIFIANT	
STATUS	Посада працівника	Varchar	Not null	Administrateur
Personnel				
ID		Int	IDENTIFIANT	5
LAST_NAME	Прізвище працівника	Varchar	Not null	Ганько
FIRST_NAME	Ім'я працівника	Varchar	Not null	Лів
ADDRESS	Адреса працівника	Varchar	Not null	вул. Вишнева, д. 15, кв. 4
EMAIL	Email працівника	Varchar	Not null	ganco@biblio.com
TEL	Телефон працівника	Varchar	Not null	050 111 11 11
PASSWORD	Пароль працівника	Varchar	Not null	Admin1
User				
ID		Int	IDENTIFIANT	
LAST_NAME	Прізвище користувача	Varchar	Not null	Ткач
FIRST_NAME	Ім'я користувача	Varchar	Not null	Олена
ADDRESS	Адреса користувача	Varchar	Not null	вул. Вишнева, д. 4, кв. 45
EMAIL	Email користувача	Varchar	Not null	ot1987@gmil.com
TEL	Телефон користувача	Varchar	Not null	069 222 22 22

PASSWORD	Пароль користувача	Varchar	Not null	User2209
State				
ID		Varchar	IDENTIFIANT	
STATE	Стан бронювання	Varchar	Not null	У черзі
Lending				
ID		Varchar	IDENTIFIANT	3
DATE_RECEIVE	Дата видачі книги	Date	Not null	25/06/2025
DATE_RETURN	Дата повернення книги	Date		05/07/2025
Comment				
ID		Int	IDENTIFIANT	3
COMMENT	Текст коментарію	Text	Not null	Цікава книга.

Після визначення сутностей та атрибутів виявляються залежності між двома і більше сутностями, сутностями та атрибутами, зв'язками атрибутів між собою. Це представлено у таблиці 2.3.

Таблиця 2. 3

Зв'язки та їх атрибути концептуальної моделі бібліотеки

Перша сутність	Назва зв'язку	Друга сутність	Назва атрибуту	Обмеження
Author	Author_Work	Literary_Work	-	
	1,n : 1,n			
Genre	Genre_Work	Literary_Work	-	
	0,n : 1,n			
Literary_Work	Work_Book	Book	-	
	1,n : 1,n			
Publishing_House	Publish	Book	-	
	0,n : 1,1			
Book	Be	Lending	-	
	0,n : 1,1			

State	Belong	Lending	-	
	0,n : 1,1			
Personnel	Confirm	Lending	-	
	0,n : 0,1			
Status	Belong	Personnel	-	
	0,n : 1,1			
User	Do	Lending	-	
	0,n : 1,1			
Book	Have	Comment	-	
	0,n : 1,1			
User	Write	Comment	-	
	0,n : 1,1			
Comment	Respond	Comment	-	
	0,n : 0,1			

На самій концептуальній моделі відображаються сутності і зв'язків між ними за допомогою візуальної концептуальної схеми рис.2.4.

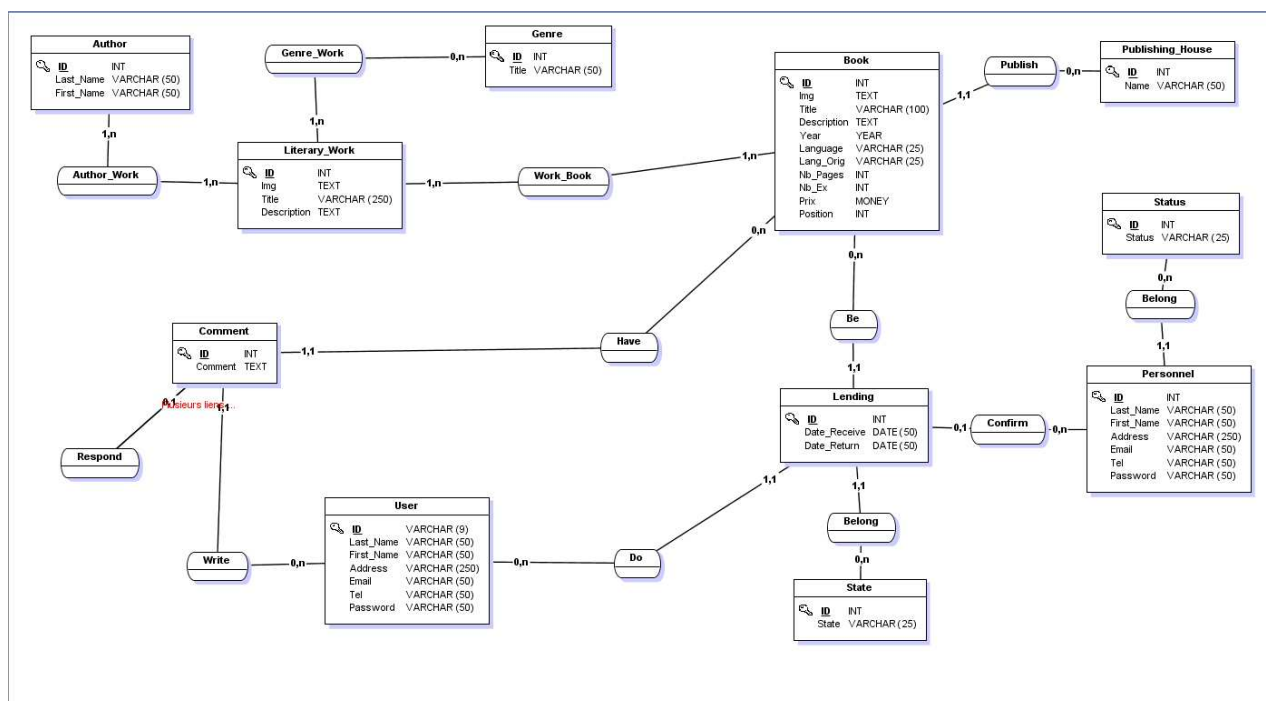


Рис. 2. 4 Концептуальна модель даних

2.2.2. Логічна модель

Логічне проектування полягає в створенні логічної моделі на основі вибраної моделі даних.

У якості логічній моделі бази даних була обрана реляційна модель, оскільки саме реляційна модель використовується у більшості розвинених СУБД.

На створеній концептуальній моделі (рис. 2.5) є зв'язки багато до багатьох, тому на логічній моделі були створені додаткові таблиці. Таким чином в логічній моделі всі зв'язки встановлені як один до багатьох або багато до одного. Для перевірки вірності логічної моделі застосовано правила нормалізації

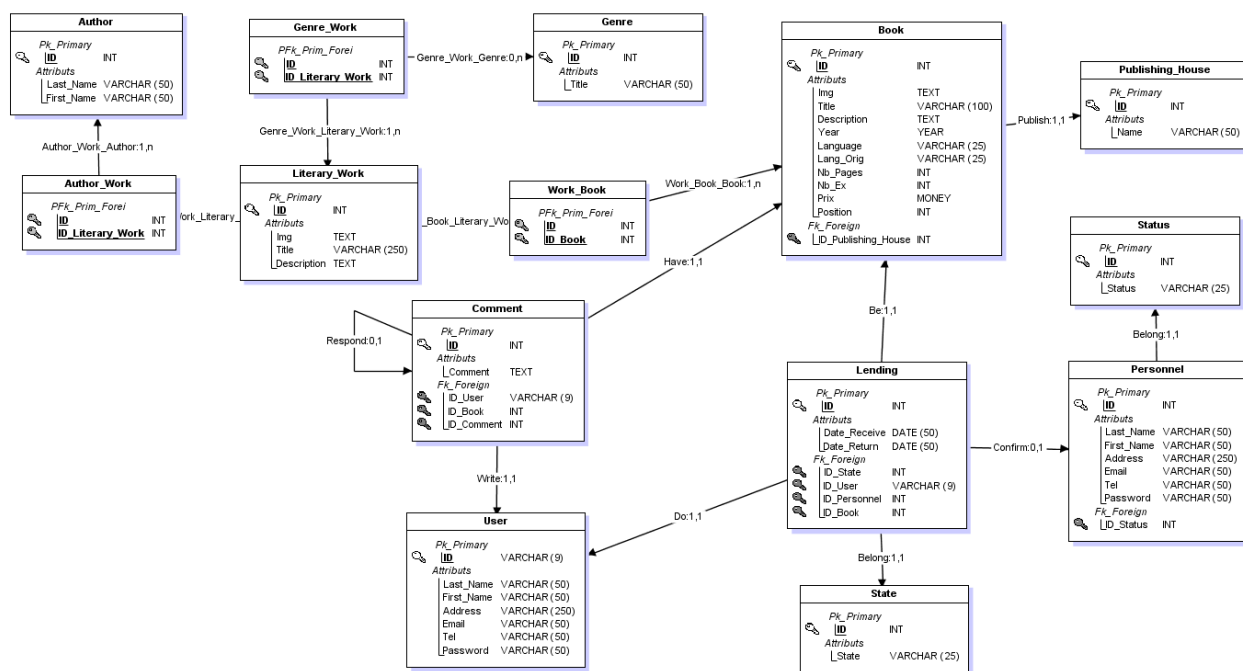


Рис. 2. 5 Логічна модель даних

2.2.3. Фізична модель

Фізичне проектування полягає в описі засобів фізичної реалізації логічного проекту БД. Фізичні моделі визначають засоби розміщення даних в середовищі зберігання і засоби доступу до цих даних, які підтримуються на фізичному рівні. У процесі проектування визначається структура реляційної БД (склад таблиць, їх структура і логічні зв'язки). Структура таблиці визначається складом стовпців, типом даних і розмірами стовпців, ключами таблиці.

В таблиці 2.4 приведена специфікація БД (вказані таблиці, назви атрибутів, визначено типи даних атрибутів, визначено розмір поля в байтах по кожному з атрибутів, визначено первинні та зовнішні ключі).

Таблиця 2. 4

Специфікація БД

Назва	Тип	Розмір поля (байт)	Ключ
Author			
ID	Intenger	1	Первинний
LAST_NAME	Text	50	
FIRST_NAME	Text	50	
Genre			
ID	Intenger	1	Первинний
TITLE	Text	50	
Literary_Work			
ID	Intenger	1	Первинний
IMG	Text	250	
TITLE	Text	100	
DESCRIPTION	Text	250	
Author_Work			
ID_AUTHOR	Intenger	1	Первинний

ID_WORK	Intenger	1	Первинний
Genre_Work			
ID_GENRE	Intenger	1	Первинний
ID_WORK	Intenger	1	Первинний
Publishing_House			
ID	Intenger	1	Первинний
NAME	Text	50	
Book			
ID	Intenger	1	Первинний
IMG	Text	250	
TITLE	Text	100	
DESCRIPTION	Text	250	
YEAR	Intenger	4	
LANGUAGE	Text	25	
LANG_ORIG	Text	25	
NB_PAGES	Intenger	3	
NB_EX	Intenger	3	
PRIX	Numeric	4	
POSITION	Intenger	2	
ID_PUBLISHING_HOUSE	Intenger	1	Зовнішній
Status			
ID	Intenger	1	Первинний
STATUS	Text	25	
Personnel			
ID	Intenger	1	Первинний
LAST_NAME	Text	50	
FIRST_NAME	Text	50	
ADDRESS	Text	250	
EMAIL	Text	50	

TEL	Text	12	
PASSWORD	Text	8	
ID_STATUS	Intenger	1	Зовнішній
User			
ID	Text	9	Первинний
LAST_NAME	Text	50	
FIRST_NAME	Text	50	
ADDRESS	Text	250	
EMAIL	Text	50	
TEL	Text	12	
PASSWORD	Text	8	
State			
ID	Intenger	1	Первинний
STATE	Text	25	
Lending			
ID	Intenger	1	Первинний
DATE_RECEIVE	Text (формат Date)	10	
DATE_RETURN	Text (формат Date)	10	
ID_USER	Intenger	1	Зовнішній
ID_BOOK	Intenger	1	Зовнішній
ID_PERSONNEL	Intenger	1	Зовнішній
ID_STATE	Intenger	1	Зовнішній
Comment			
ID	Intenger	1	Первинний
COMMENT	Text	250	
ID_USER	Intenger	1	Зовнішній
ID_BOOK	Intenger	1	Зовнішній
IS_ANSWER	Intenger	1	Зовнішній

Всього: 2658

У якості СУБД обрано SQLite. Тому у столбці «Розмір поля» вказані приблизні значення. У даній СУБД ці розміри залежать від збережених у полі даних.

2.4. Створення веб-сайту

Веб-сайт – це сукупність багатьох веб-сторінок, доступних через інтернет, об'єднаних загальною темою і пов'язаних між собою гіпертекстовими та іншими навігаційними посиланнями. Простіше – це певне місце в інтернеті, де можна розмістити будь-яку інформацію, зробивши її доступною з будь-якої точки світу.

У підхід до розробки сайту зазвичай виділяють такі етапи:

- 1) визначення концепції сайту та функціонала;
- 2) розробка структури сайту;
- 3) малювання блокової структури сторінок;
- 4) розробка дизайну;
- 5) верстка та створення адаптивної версії;
- 6) розробка бекенду сайту;
- 7) наповнення сайту контентом;
- 8) кінцеве тестування функціонала сайту;
- 9) використання додаткових технічних правок.

2.4.1. Мета створюваного сайту

Підвищити рівень бібліотеки як культурно-просвітницького центру, за рахунок забезпечення зручного доступу читачів до муніципальної бібліотеки, та зробити роботу працівників бібліотеки більш зручною та якісною.

2.4.2. Визначення концепції сайту та структури сайту

Сайт муніципальної бібліотеки – це сайт послуг.

На цьому сайті продумується і розробляється правильна структура.

В даному випадку буде використовуватись гібридна структура, що буде включати у себе:

- Блочну. Так як планується, що на усі основні сторінки можна буде потрапити з будь-якого місця через верхнє або бічне меню, з елементами ієрархії.
- Ієрархічну. Бо деякі сторінки будуть мати підкатегорії.
- Орієнтовану на цільову аудиторію. Такий метод побудови структури ефективний завдяки тому, що кожен користувач залежно від умов можуть отримати додаткові функції веб-сайту.

2.4.3. Малювання блокової структури сторінок

Для деяких сторінок заздалегідь створюються шаблони сторінок, щоб у подальшому основні блоки не забути враховувати у дизайні та наступній верстці сайту.

Тому, щоб побачити приблизну картину сайту, були створені наступні шаблони (рис. 2.6-2.8):

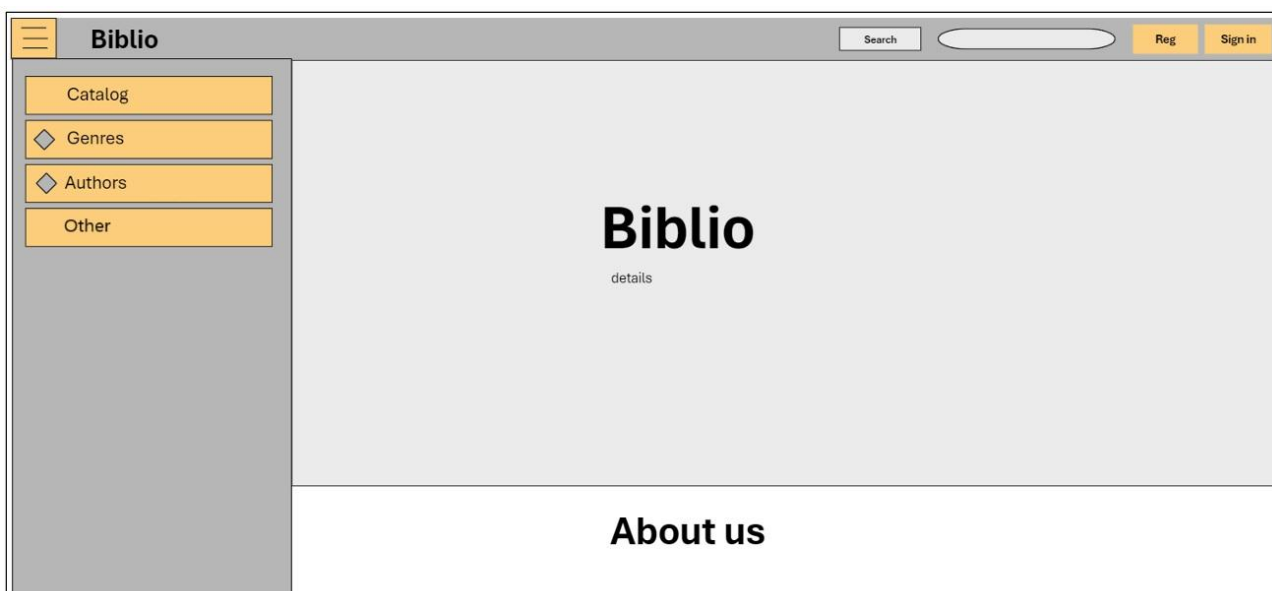


Рис. 2. 6 Шаблон головної сторінки

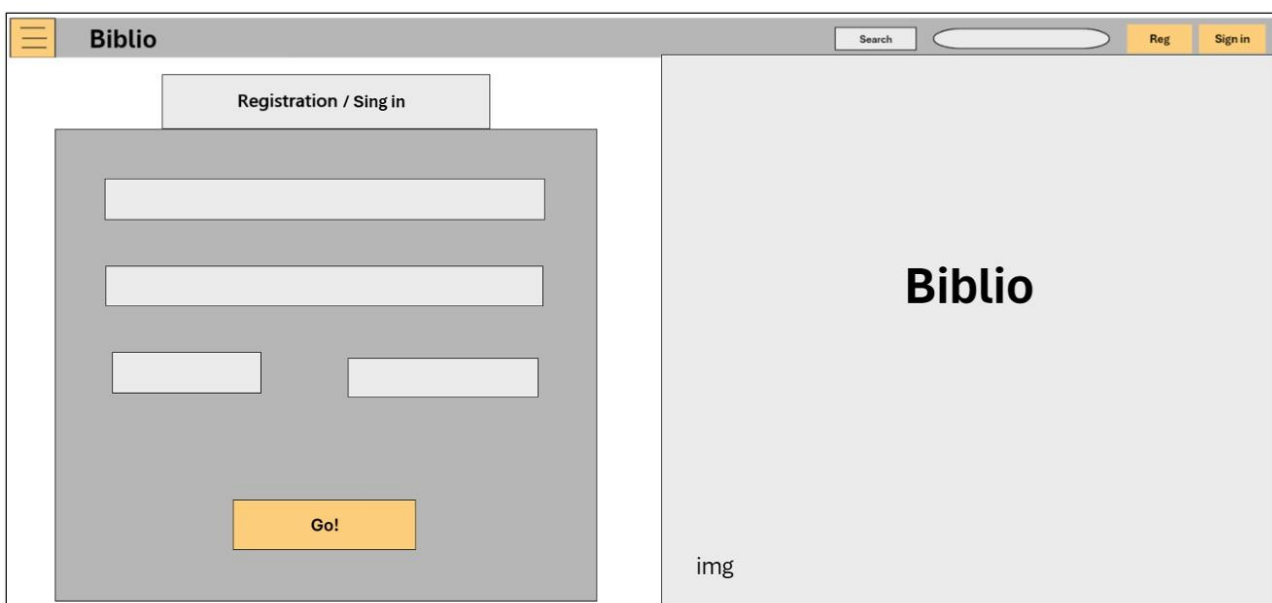


Рис. 2. 7 Шаблон сторінки входу/реєстрації



Рис. 2. 8 Шаблон сторінки каталогу

2.4.4. Розробка дизайну та верстка сторінок

На основі шаблонів було визначено дизайн (рис. 2.10):



Рис. 2. 9 Демонстрація головної сторінки сайту

Для верстки (створення фронтенду) були використані традиційні інструменти:

- **HTML з Template Engine** для створення HTML-шаблонів сторінок (Template Engine допомагає пришвидшити та оптимізувати цей процес);
- **CSS** для додавання стилів;
- **JavaScript** для додавання інтерактивності.

2.4.5. Backend: вибір інструментів для розробки

Python, а точніше його фреймворк Flask, добре підходить для розробки веб-сайтів. Python також добре підходить для створення чат-ботів.

Flask практичний завдяки своїм функціям:

- Мінімальний набір готових до використання інструментів. Крім того, вони не нав'язують жодної жорсткої архітектури чи структури проекту. Ми можемо самі вирішувати, як і що ми будемо створювати.
- Гнучкість. За допомогою Flask програміст може вибрати лише необхідні вбудовані інструменти та підключити додаткові зовнішні інструменти, не перевантажуючи проект зайвими модулями.
- Розширюваність. Flask пропонує численні розширення та плагіни для швидкого додавання нових функцій, таких як управління базами даних та форми.
- Простота. Flask має простий синтаксис, що дозволяє швидко створювати прототипи веб-додатків.

За допомогою Python Flask зручно використовувати SQLite, на який ми створюємо базу даних, оскільки для роботи з нею нам потрібно лише імпортувати бібліотеку sqlite3.

2.4.6. Тестування функціонала сайту

Після того як відбулося повне наповнення сайту, здійснюється повна перевірка функціонування:

- відображення та виведення елементів;
- коректність роботи навігації;
- виведення даних;
- функціонування фільтрів і сортувань;
- перевірка форм;
- працездатність реєстрації/входу на сайті;
- перевірка внутрішнього функціонала з оформлення бронювань.

Але це ще не останній етап створення сайту. У подальшому використанні важливий і невід’ємний є подальший розвиток і підтримка сайту. Необхідно буде постійно оновлювати функціональність і контент, аналізувати дані про відвідуваність ресурсу, моніторити його роботу і проводити аналіз поведінки користувачів. Виходячи зі зворотного зв’язку користувачів, потрібно вносити необхідні зміни, покращувати наявні функції, оптимізувати продуктивність і надавати користувачам безпеку сайту.

Висновки

В другому розділі описано процес проектування інформаційної системи з WEB-доступом для муніципальної бібліотеки. Всі необхідні етапи проектування успішно завершені.

РОЗДІЛ 3. РЕЗУЛЬТАТИ РОЗРОБКИ ПРОГРАМНОГО ПРОДУКТУ ДЛЯ МУНІЦИПАЛЬНОЇ БІБЛІОТЕКИ

3.1. Результати розробки програмного коду

3.1.1. Програмні файли

Основний файл **app.py** запускає застосунок. Він містить у собі основну логіку роботи сайту, у ньому записана поведінка кожної створеної сторінки сайту та її взаємодія з базою даних. Також якраз у ньому зазначено підключення Фласк й бази даних **data_base.db**, розглянутої нижче.

Було створено базу даних **data_base.db**, за допомогою sql коду у файлі **data_base.sql** цього проекту.

Для користування базою даних було створено клас **FDataBase** у файлі **FDataBase.py**, функції у якому інтретретьють необхідні sql-запити.

3.1.2. HTML-сторінки

Усі html-сторінки проекту знаходяться у папці **templates** за замовчуванням.

Верхнє та нижнє меню. Шаблон

«Шапка» та низ сайту прописано у файлі **Menus.html**. У ньому ж вже написано спільне для усіх сторінок **head**, у тому числі й підключення до створених **css** файлів.

Цей файл не є повноцінною сторінкою сайту, але його можна назвати шаблоном усіх інших сторінок, адже цей каркас, верхнє та нижнє меню (як і код **html**, **head** та **body**), є у кожній сторінці, змінюється лише контекст між ними та назва самої сторінки. Тому зручно написати все це один раз, а потім викликати та вставляти у нього необхідний контекст з назвою.

Верхнє містить у собі кнопку виклику бокового меню (дія якого прописана у файлі js, про який ми поговоримо пізніше), логотип сайту зі ссилкою на головну сторінку, рядок пошуку по сайту та дві кнопки: реєстрація і вхід для неавторизованого користувача; мої бронювання та вихід для авторизованого.

Функції рядку пошуку прописані для кожної сторінки у їх функціях файлу app.py.

У нижньому меню просто зазначені можливі контакти.

3.1.2.1. Загальнодоступні сторінки

Головна сторінка

Головна сторінка являє собою представлення бібліотеки, її короткий опис.

Саме з цієї сторінки запускається сайт. Це прописано у файлі app.py, а саме у функції home(), яка викликає html-сторінку **Home.html**, яка у свою чергу використовує шаблон Menus.html. Назва сторінки зазначена як Biblio, а у контексті вся інша необхідна інформація. Результат можна побачити на рис.3.1.

З цієї сторінки, як і з решти, можна перейти на будь-яку іншу за допомогою верхнього та бічного меню.

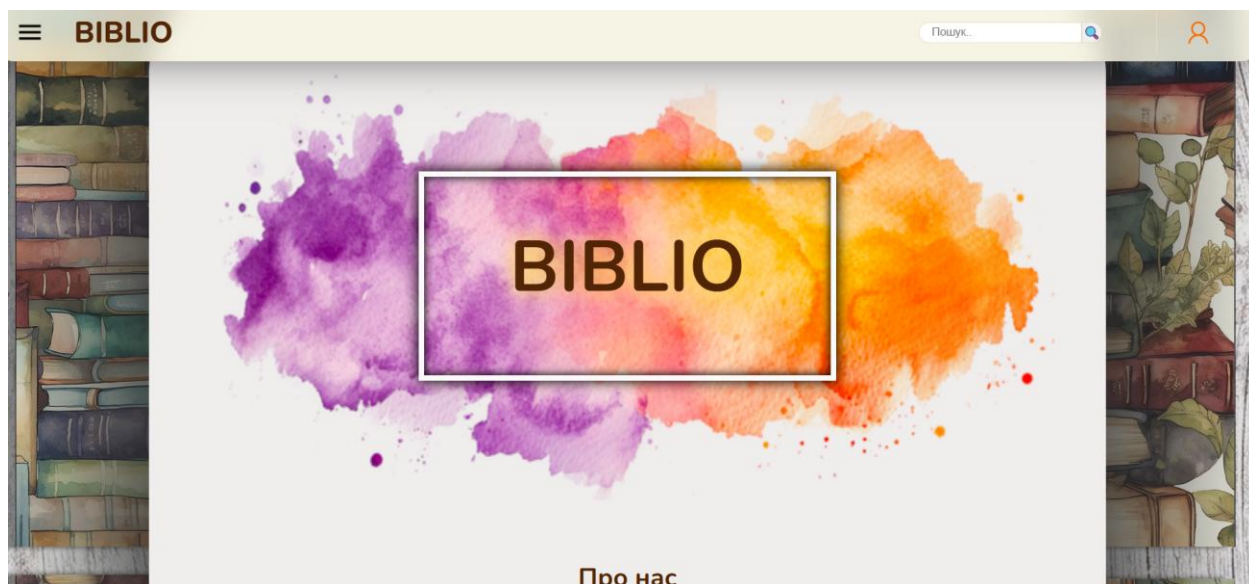


Рис. 3. 1 Головна сторінка

Загальний каталог

Каталог — це повний перелік усіх наявних книг у бібліотеці у тому порядку, як вони занесені у базу даних. Перелік створено таким чином, що користувач баче: обкладинку книги, її назву (що є посиланням на сторінку самої книги), автора(ів) (при натисканні перейдемо на каталог книг автора), жанри (посилання на каталог жанру) й навіть короткий синопсис (рис.3.2).

Цей перелік створено у файлі **ListBooks.html** для подальшого використання у html-сторінках, так як такий перелік знадобиться нам не єдиний раз. Для його відображення необхідні дані, які ми можемо взяти з заповненої бази даних. Робимо ми це в функції **catalog()** файлу **app.py** за допомогою різних функцій **get** класу **FDataBase**.

Вона також викликає саму html-сторінку **Catalog.html**, яка знов використовує **Menus.html**, має назву **BiblioCatalog** та у контексті просто викликає **ListBooks.html**.

До нього можна перейти звідусіль і будь-кому через бічне меню.

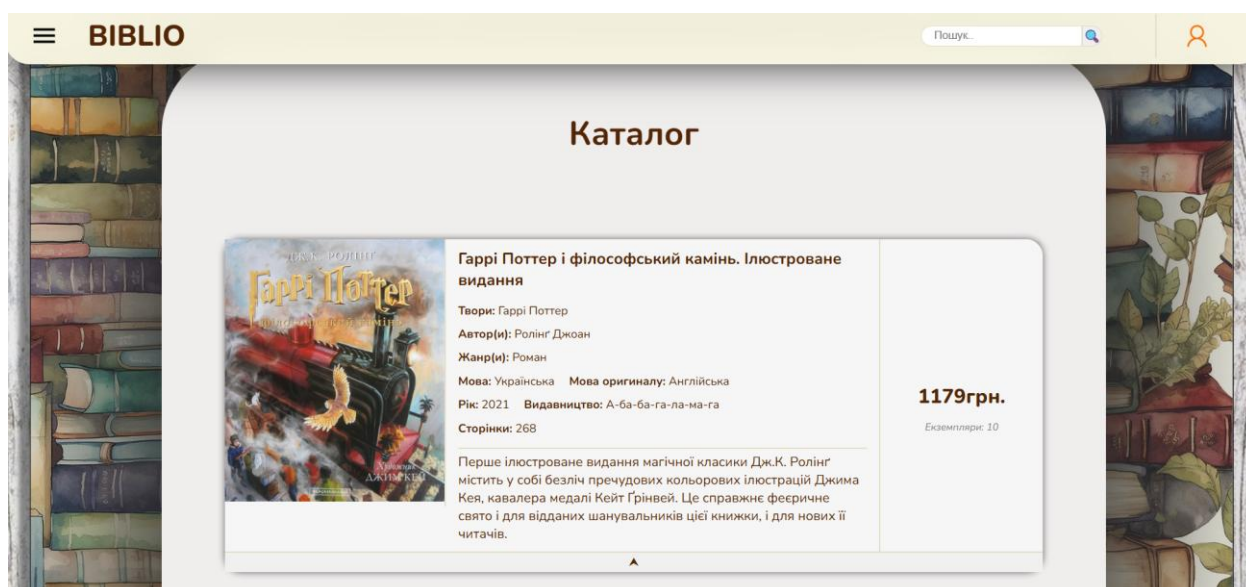


Рис. 3. 2 Основний каталог

Відфільтровані каталоги

Аналогічну структуру мають й вже відфільтровані каталоги. На сайті є три типи таких: за автором, жанром та твором. На них можна перейти зі сторінок списків творів, авторів та жанрів, посилання на які знаходиться у бічному меню.

Списки

Як було сказано раніше, це сторінки, з яких можна перейти на відповідно відфільтровані каталоги. На самі сторінки списків можна зайти через бічне меню.

Структура цих сторінок проста – шаблон Menus.html + необхідний список з бази даних.

Один з цих списків продемонстровано на рис.3.3.

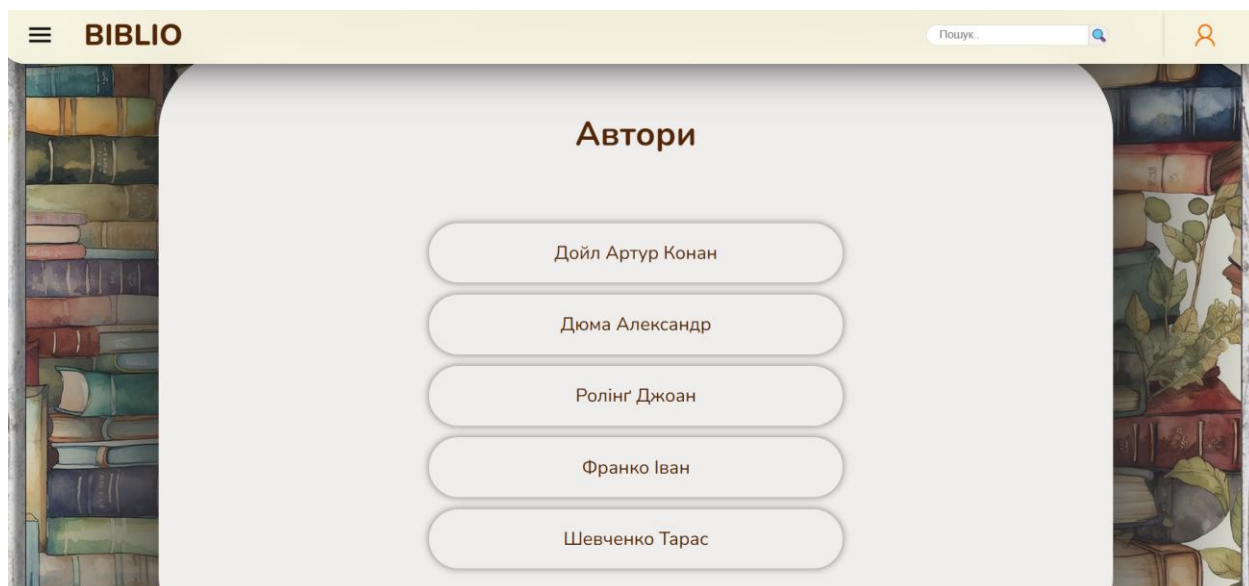


Рис. 3. 3 Список авторів

Пошук

Вище перелічене, звісно, у разі спрощує орієнтування по сайту, але якщо нам потрібно швидко щось знайти, не дуже зручно шукати потрібний результат по категоріям (рис.3.4). Простіше просто зробити запит у пошук, який на сайті присутній.

Сторінка прописана у **Search.html**, що викликається функцією `search()`.

Пошук по сайту шукає книги відразу по чотирьом напрямкам: книга(назва чи опис), твір, автор та жанр (рис. 3.).

Пошук розроблено таким чином, що він може знаходити відповідні книжки та інше, навіть якщо користувач зробив якусь помилку при написанні запиту (наприклад, написав Потер, замість Поттер). Якщо ж книг з такою чи схожою назвою не знайдено, вам можуть запропонувати варіанти книг, де слова з вашого запиту зустрічались у описі.

Таким же чином виконується й пошук творів, авторів, творів.

Все це за допомогою наступної функції з файлу `FDataBase.py`:

```

def search(self, s, table):
    try:
        search = ""
        s = s.lower()
        selected = "*"
        searchTitle = "TITLE"
        if table == 'Author':
            selected = "ID, (LAST_NAME || ' ' || FIRST_NAME) AS NAME"
            searchTitle = "NAME"
        elif table == 'Genre':
            selected = "ID, TITLE"
        if len(s) < 3:
            search += "SELECT " + selected + " FROM " + table + "\n" + \
                "WHERE lower("+searchTitle+") LIKE '%" + s + "%'\n"
        else:
            for i in range(len(s) - 2):
                search += "SELECT " + selected + " FROM " + table + "\n" + \
                    "WHERE lower("+searchTitle+") LIKE '%" +
s[i:i+3]+"%'\n"
                if i != len(s) - 3:
                    search += "UNION ALL\n"
            if table in ["Book", "Literary_Work"]:
                listW = s.split()
                forDescrip = '%'.join(listW)
                search += "UNION ALL\n" + \
                    "SELECT * FROM " + table + "\n" + \
                    "WHERE lower(DESCRIPTION) LIKE '%" + forDescrip +
"%'\n"
            self.__cur.execute("SELECT *, count(ID) AS ACCURACY FROM (\n" +
                search + ")\n" +
                "GROUP BY ID\n" +
                "ORDER BY ACCURACY DESC, " + searchTitle)
            res = self.__cur.fetchall()
            if res:
                return res
        except sqlite3.Error as e:
            print("Error:      " + str(e))
    return []

```

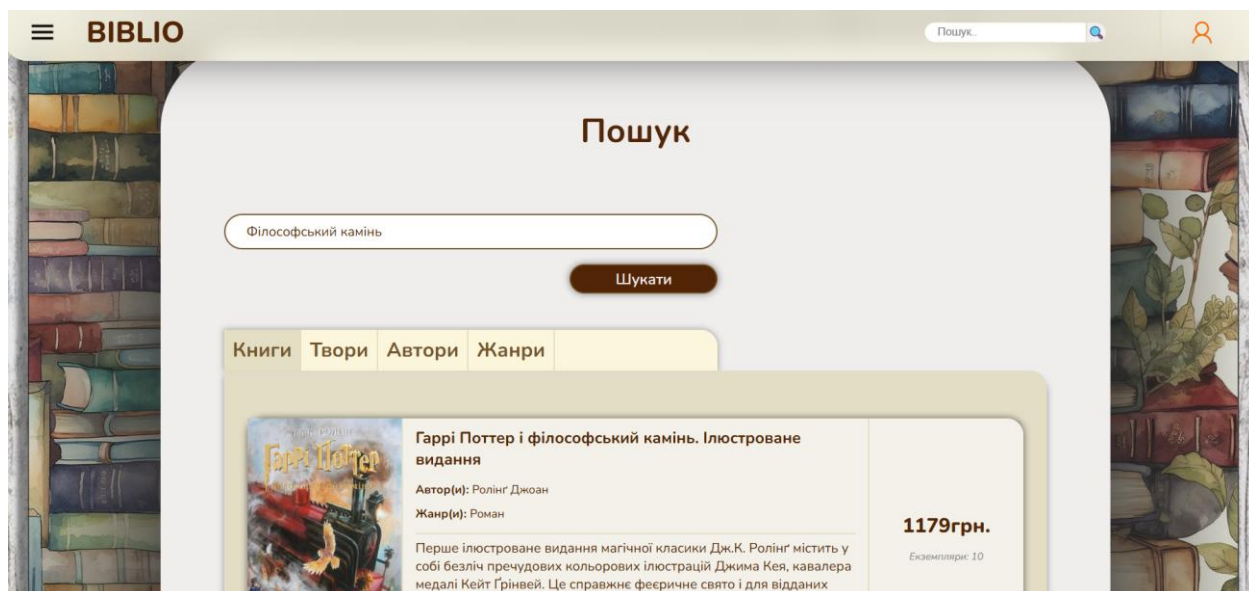


Рис. 3. 4 Пошук

Сторінка книжки

Не важливо який каталог використовує користувач, з кожного можна перейти на сторінку обраної книги (рис.3.5), натиснувши на її назву.

Сторінка книги містить у собі майже той самий опис, що і у каталозі, та кнопку бронювання з відділом коментарів, функції яких доступні лише авторизованим читачам. Сторінка викликається функцією `book(b_id)`, де прописана також взаємодія з базою, html-код записан у **Book.html**.

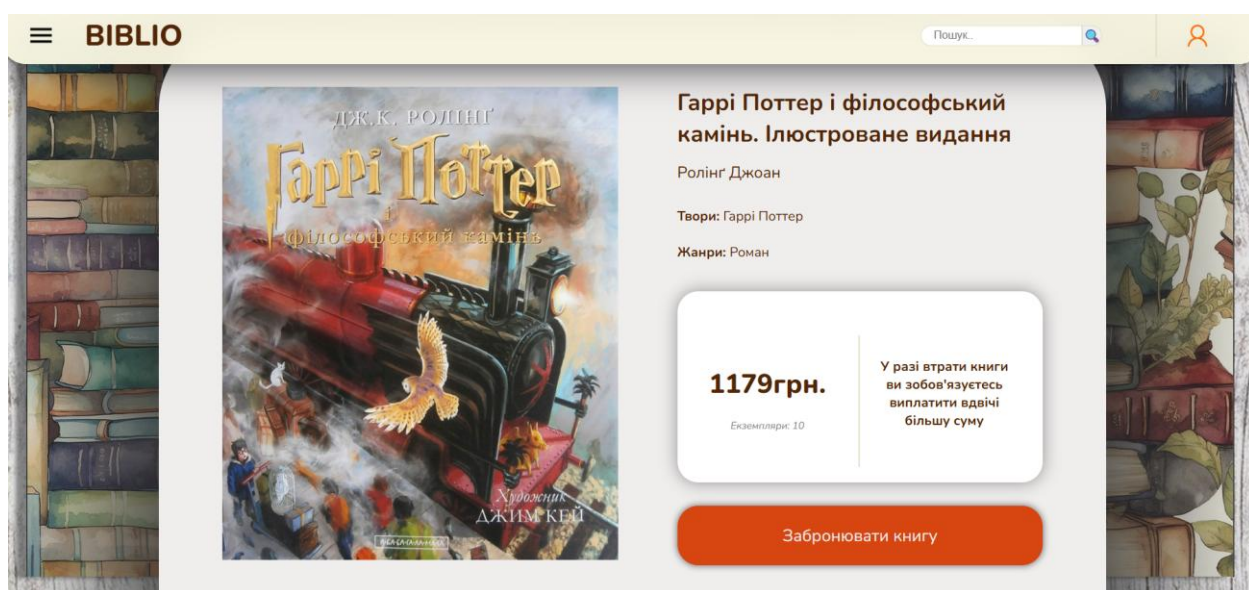


Рис. 3. 5 Сторінка книги

Реєстрація та вхід

Якщо користувач хоче забронювати екземпляр книги, то він повинен бути авторизованим. Увійти у аккаунт можна за допомогою сторінки «Вхід» (рис.3.6). На неї можна перейти нажавши відповідну кнопку у верхньому меню.

Для входу користувачу необхідно ввести свій емейл та пароль, який він вносив при реєстрації. У разі внесення некоректних даних буде спливати відповідне повідомлення, це прописано у функції login().

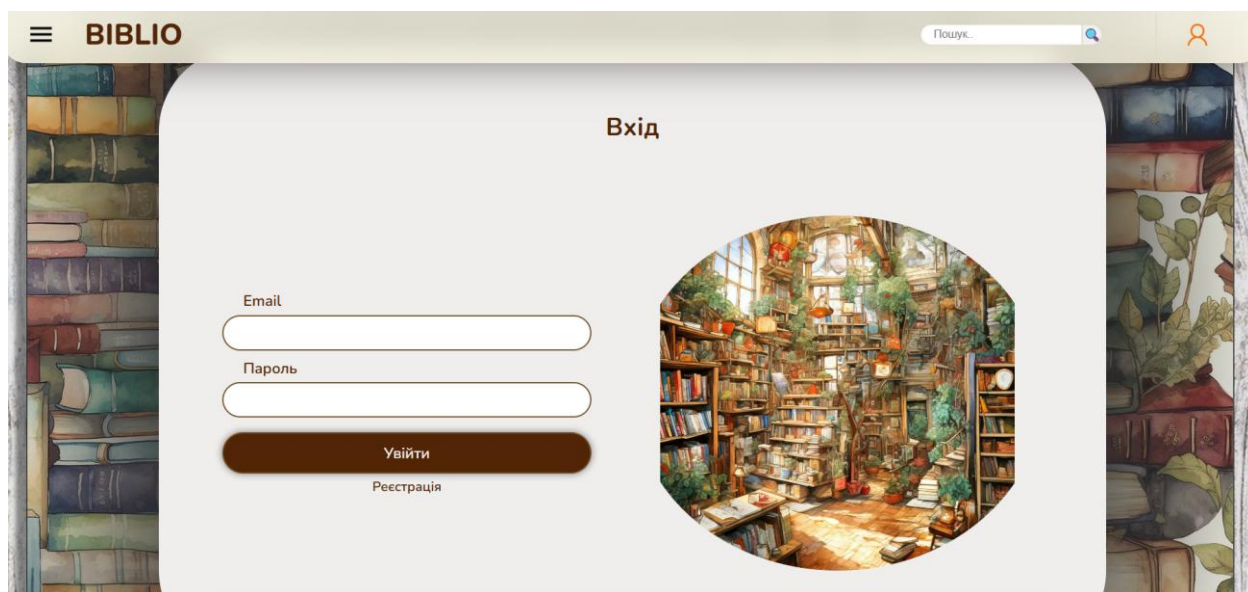


Рис. 3. 6 Вхід

У разі якщо користувач ще не був зареєстрованим йому слід перейти на сторінку реєстрації (рис.3.7) за посиланням на сторінці входу.

Сторінка реєстрації схожа на сторінку входу, але має додаткові поля крім емейла та пароля. Такі як: id, ім'я, прізвище, номер телефону. Кожне з цих полів є обов'язковим. Якщо якесь поле пропущене або введено некоректно, про це з'явиться відповідне повідомлення.

Рис. 3. 7 Реєстрація

Форма входу та реєстрації обидві працюють з базою даних, але по різному. Форма входу лише звіряє введені дані зі взятими з бази, а форма реєстрації вносить у неї зміни, а саме вводить нові дані у базу.

Задля безпеки персональних даних користувачів пароль від їх аккаунта зберігається у базі у закодованому вигляді.

3.1.2.2. Сторінки доступні авторизованим користувачам

Аккаунт

Після реєстрації кожен користувач (й читач, й працівник) отримує свій особистий кабінет. На його сторінці користувач може переглядати свій «формуляр читача», а працівник може побачити видані безпосередньо ним книги з інформацією про замовника.

Існування аккаунту також надає можливість користувачам бачити та модифікувати свої особисті дані.

3.1.2.3. Сторінки доступні працівникам

Всі користувачі

Працівники бібліотеки мають право передивлятися списки усіх зареєстрованих читачів та їх бронювання.

Додавання нової інформації

Працівники мають право вносити зміни у базу даних, у каталог бібліотеки.

Задля того, щоб додати нову книгу (рис.3.8) необхідно ввести у форму всі необхідні дані: її назву, посилання на зображення (усі зображення у базі даних зберігаються у виді посилань), її автора чи авторів з переліку (у разі якщо автора у ньому немає, його можливо теж додати, використавши відповідну функцію), жанр(и) (система дій аналогічна з вибором авторів і т.д..

The screenshot shows the 'BIBLIO' library management system interface. At the top, there is a navigation bar with a menu icon, the word 'BIBLIO', a search bar labeled 'Пошук...', and a user profile icon. The main content area is titled 'Додати книгу' (Add book). It contains several input fields: 'Назва *' (Title), 'Посилання на обкладинку' (Cover link), 'Твори *' (Works), 'Видавництво *' (Publisher), and a section for 'Опис' (Description) with a 'Description...' label. There is also a dashed box for an image with an orange icon. At the bottom, there is a large brown button labeled 'Додати' (Add).

Рис. 3. 8 Додати книгу

Інструменти

Посилання на всі сторінки працівників можна знайти у вкладці «Інструменти» бічного меню. Її вигляд продемонстровано на рис.3.9

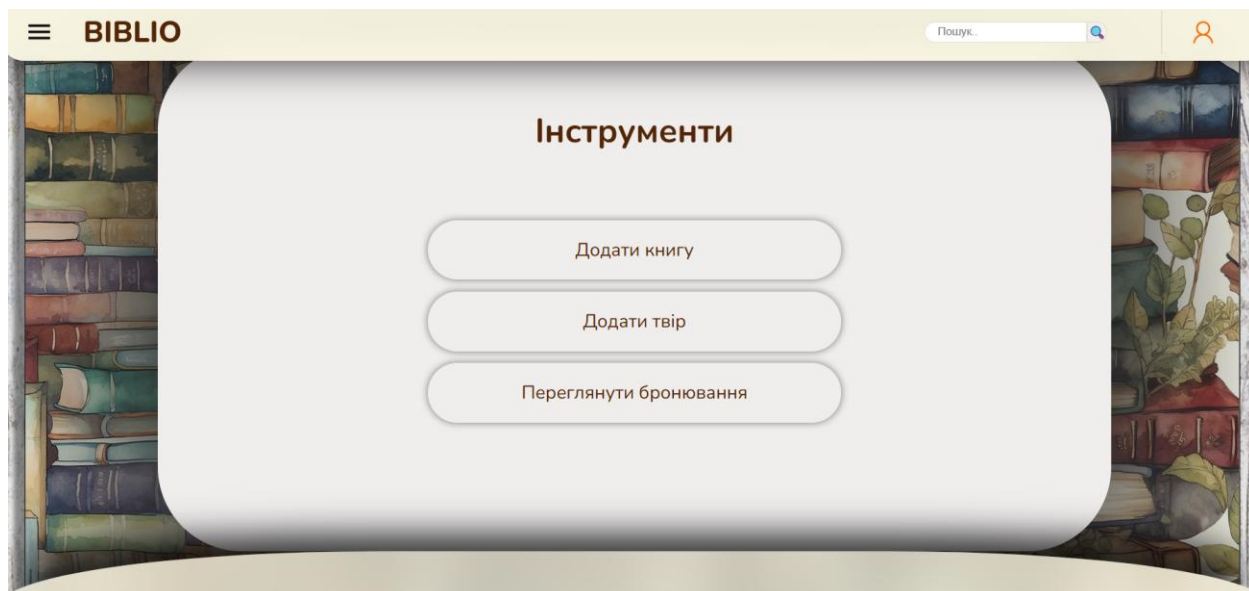


Рис. 3. 9 Сторінка «Інструменти»

3.1.2.4. Сторінки доступні адміністраторам

Всі працівники

Адміністратори бібліотеки мають право передивлятися списки усіх працівників та назначати нових адміністраторів.

Реєстрація нових працівників

У обов'язки адміністраторів також входить створення облікових записів нових працівників. Сторінка реєстрації працівників аналогічна до реєстрації читачів.

3.1.2.5. Файли JavaScript та CSS

За всі інтерактивні дії на сайті відповідає JavaScript (файл script.js. У ньому прописана поведінка різних функцій після натискання кнопок і т.д. (наприклад поведінка бічного меню).

Щоб все працювало, файл викликається сторінками html рядком `<script src="/static/js/script1.js"></script>` блока `<head>` у сторінці-шаблоні.

За естетичну сторону сайту відповідають файли CSS, де описуються стилі, кольорова гамма й просто загальний вигляд сайту на коді .

Підключаються вони до сторінок у блоці `<head>` рядком `<link rel="stylesheet" href="/static/css/style____.css">`.

3.1.3. Розроблений програмний код

Приклад розробленого програмного коду знаходиться у Додатку Б.

3.2. Результати тестування програми

Всі тестування пройдені успішно. Рисунки приведені вище це підтверджують.

Висновки

В результаті виконання кваліфікаційної роботи було розроблено програмне забезпечення для інформаційної системи муніципальної бібліотеки з WEB-доступом, а саме сайт.

В кваліфікаційній роботі було розроблено ескізний, технічний та робочий проекти, проведено кодування, тестування та випробування програмного забезпечення, створено наочний та інтуїтивно-зрозумілий інтерфейс сайту бібліотеки. Робота на сайті забезпечує виконання різноманітних запитів користувачів. На сайті передбачено пошук літератури по різним критеріям, а також реєстрацію користувачів, додавання коментарів до книжок користувачами сайту, резервування необхідної літератури.

Програмне забезпечення виконує функції, що були зазначені в технічному завданні (додаток А), створено базу даних та сайт бібліотеки.

Створений програмний продукт було протестовано. У ході тестування програмне забезпечення показало свою повну працездатність, а також високий рівень роботи з виконанням усіх необхідних задач, поставлених при його проектуванні.

Випробування програмного забезпечення було проведено відповідно до програми й методики випробувань.

В ході виконання кваліфікаційної роботи була створена інструкція для користувача сайту (Додаток В).

РОЗДІЛ 4. ОХОРОНА ПРАЦІ

4.1. Аналіз небезпечних і шкідливих факторів

Охорона праці — це система правових, соціально-економічних, організаційно-технічних, санітарно-гігієнічних і лікувально-профілактичних заходів та засобів, спрямованих на збереження життя, здоров'я і працездатності людини у процесі трудової діяльності.

Мета охорони праці — робота без травм, захворювань та аварій з збереженням працездатності, що здійснюється за рахунок реалізації її практичних та наукових задач. До практичних задач відносять розробку заходів для досягнення поставленої мети, а до наукових — отримання нових нетрадиційних знань, заснованих на виявлених законах, математичних описах нових явищ або нових схемах технологічного процесу.

При організації умов праці необхідно враховувати вплив на працюючих небезпечних і шкідливих виробничих факторів, які можуть привести до травми або іншого раптового різкого погіршення здоров'я та захворювання або зниження працездатності.

Всі небезпечні і шкідливі виробничі чинники відповідно до ГОСТ 12.0.003-74 підрозділяються на **фізичні, хімічні, біологічні та психофізіологічні**.

До фізичних небезпечних і шкідливих факторів відносяться: рухомі елементи машин і механізмів; вироби, що переміщуються; матеріали; заготовки; руйнування конструкцій; надлишкова запиленість і загазованість повітря в робочій зоні; невідповідна температура поверхні обладнання; підвищена або понижена температура повітря в робочій зоні; підвищений рівень шуму, вібрацій, ультразвуку, інфразвукових коливань; підвищений або понижений барометричний тиск; підвищена або знижена відносна вологість; рух, іонізація повітря; підвищений рівень статичної електрики, електромагнітних коливань, відсутність або недостатнє природне освітлення;

недостатня штучна освітленість робочої зони; підвищена яскравість світла; прямий та відбитий блиск; підвищена пульсація світлового потоку; підвищені рівні ультрафіолетової та інфрачервоної радіації; гострі краї, жорсткість поверхні деталей, інструментів та обладнання; розташування робочих місць на значній висоті відносно землі; невагомість.

До хімічних небезпечних і шкідливих виробничих факторів відносяться хімічні речовини, які за характером дії на організм людини поділяються на токсичні, подразнюючі, сенсibilізуючі, канцерогенні, мутагенні, котрі впливають на репродуктивну функцію. За шляхом проникнення в організм людини вони поділяються на такі, що проникають через дихальні шляхи, шлунково-кишковий тракт, слизові оболонки і поверхні тіла людини.

До біологічних небезпечних і шкідливих виробничих факторів відносяться патогенні мікроорганізми (бактерії, віруси, гриби) і продукти їх життєдіяльності, а також макроорганізми (рослинні і тваринні).

До психофізичних небезпечних і шкідливих виробничих факторів відносяться фізичні (статичні і динамічні) і нервово-психічні перевантаження (розумове перенапруження, перенапруження каталізаторів, монотонність праці, емоційне перенавантаження). Нервово - психічні навантаження спричиняються переробкою великої кількості інформації і стосуються операторів пультів управління при порушенні режиму праці і відпочинку.

Науково-технічний прогрес серйозно вплинув на умови виробничої діяльності працівників розумової праці. Їх праця стала більш інтенсивною, напруженою, потребуючою значних витрат розумової, емоційної і фізичної енергії. Це зажадало комплексне рішення проблем ергономіки, гігієни і організації праці, регламентації режимів праці і відпочинку.

В даний час комп'ютерна техніка широко застосовується у всіх областях діяльності людини. При роботі з комп'ютером людина піддається дії ряду небезпечних і шкідливих виробничих чинників: електромагнітних полів

(діапазон радіочастот: ВЧ, УВЧ і СВЧ), інфрачервоного і іонізуючого випромінювань, шуму і вібрації, статичної електрики і ін. 2.

Робота з комп'ютером характеризується значною розумовою напругою і нервово-емоційним навантаженням операторів, високою напруженістю зорової роботи і достатньо великим навантаженням на м'язи рук при роботі з клавіатурою ЕОМ. Велике значення має раціональна конструкція і розташовує елементів робочого місця, що важливе для підтримки оптимальної робочої пози людини-оператора.

В процесі роботи з комп'ютером необхідно дотримувати правильний режим праці і відпочинку. Інакше у персоналу наголошуються значна напруга зорового апарату з появою скарг на незадоволеність роботою, головні болі, дратівливість, порушення сну, втомленість і хворобливі відчуття в очах, в поясниці, в області шиї і руках.

Правильно спроектоване і виконане виробниче освітлення поліпшує умови зорової роботи, знижує стомлюваність, сприяє підвищенню продуктивності праці, благотворно впливає на виробниче середовище, створюючи позитивний психологічний вплив на працюючого, підвищує безпеку праці і знижує травматизм.

Недостатність освітлення приводить до напруги зору, послаблює увагу, приводить до настання передчасної стомленості. Надмірно яскраве освітлення викликає осліплення, роздратування й різь в очах. Неправильний напрямок світла на робочому місці може створювати різкі тіні, відблиски, дезорієнтувати працюючого. Всі ці причини можуть привести до нещасного випадку або профзахворювань, тому настільки важливий правильний розрахунок освітленості.

Існує три види освітлення — **природне, штучне й сполучене.**

Природне освітлення — освітлення приміщень денним світлом, що проникає через світлові прорізи в зовнішніх конструкціях, що огорожують, приміщень. Природне освітлення характеризується тим, що змінюється в

широких межах залежно від часу дня, пори року, характеру області й ряду інших факторів.

Штучне освітлення застосовується при роботі в темний час доби й удень, коли не вдається забезпечити нормовані значення коефіцієнта природного освітлення (похмура погода, короткий світловий день).

Освітлення, при якому недостатнє По нормах природне освітлення доповнюється штучним, називається **сполученим освітленням**.

Штучне освітлення підрозділяється на робоче, аварійне, евакуаційне, охоронне.

Робоче освітлення, у свою чергу, може бути загальним або комбінованим.

Загальне - освітлення, при якому світильники розміщуються у верхній зоні приміщення рівномірно або стосовно до розташування устаткування.

Комбіноване — освітлення, при якому до загального додається місцеве освітлення.

Вимоги до освітленості в приміщеннях, де встановлені комп'ютери, що впливають: при виконанні зорових робіт високої точності загальна освітленість повинна становити 300лк, а комбінована — 750лк; аналогічні вимоги при виконанні робіт середньої точності — 200 й 300лк відповідно.

Крім того усе поле зору повинно бути освітлене досить рівномірно – це основна гігієнічна вимога. Іншими словами, ступінь освітлення приміщення і яскравість екрана комп'ютера повинні бути приблизно однаковими, тому що яскраве світло в районі периферійного зору значно збільшує напруженість очей і, як наслідок, приводить до їх швидкої стомлюваності.

Параметри мікроклімату можуть мінятися в широких межах, в той час як необхідною умовою життєдіяльності людини є підтримка сталості температури тіла завдяки терморегуляції, тобто здатності організму регулювати віддачу тепла в навколишнє середовище. Принцип нормування мікроклімату – створення оптимальних умов для теплообміну тіла людини з навколишнім середовищем.

Обчислювальна техніка є джерелом істотного тепловиділення, яке може привести до підвищення температури і зниження відносної вологості в приміщенні.

У приміщеннях, де встановлені комп'ютери, повинні дотримуватися визначені параметри мікроклімату. У санітарних нормах СН-245-71 встановлені величини параметрів мікроклімату, що створюють комфортні умови. Ці норми встановлюються в залежності від пори року, характеру трудового процесу і характеру виробничого приміщення (таблиця 4.1).

Таблиця 4. 1

Параметри мікроклімату для приміщень, де встановлені комп'ютери

Період року	Параметр мікроклімату	Величина
Холодний	Температура повітря в приміщенні	22...24°C
	Відносна вологість	40...60%
	Швидкість руху повітря	до 0,1 м/с
Теплий	Температура повітря в приміщенні	23...25°C
	Відносна вологість	40...60%
	Швидкість руху повітря	0,1...0,2 м/с

Шумом називають усякий несприятливо діючий на людину звук. З фізичної точки зору звук являє собою механічні коливання пружного середовища.

Слуховий орган людини сприймає у вигляді чутного звуку коливання пружного середовища, які мають частоту приблизно від 20 до 20000 Гц, але найбільш важливий для слухового сприйняття інтервал від 45 до 10000 Гц.

Сприйняття людиною звуку залежить не тільки від його частоти, але і від інтенсивності і звукового тиску.

Несприятлива дія шуму на людину залежить не тільки від рівня звукового тиску, але і від частотного діапазону шуму, а також від рівномірності впливу протягом робочого часу.

У результаті несприятливого впливу шуму на працюючу людину відбувається зниження продуктивності праці, збільшується брак у роботі, створюються передумови до виникнення нещасливих випадків. Усе це обумовлює велике оздоровче й економічне значення заходів щодо боротьби із шумом. Граничні рівні звуку можна побачити у таблиці 4.2.

Таблиця 4. 2

Граничні рівні звуку, дБ, на робочих місцях

Категорія напруженості праці	Категорія важкості праці			
	I легка	II Середня	III Важка	IV Дуже важка
I Мало напружена	80	80	75	75
II Помірковано напружена	70	70	65	65
III Напружена	60	60	-	-
IV Дуже напружена	50	50	-	-

Більшість учених вважає, що як короткочасний, так і тривалий вплив усіх видів випромінювання від екрана монітора не небезпечний для здоров'я персоналу, що обслуговує комп'ютери. Однак вичерпних даних щодо безпеки впливу випромінювання від моніторів на працюючих з комп'ютерами не існує і дослідження в цьому напрямку продовжуються.

Максимальний рівень рентгенівського випромінювання на робочому місці оператора комп'ютера зазвичай не перевищує 10мкбер/год, а інтенсивність ультрафіолетового й інфрачервоного випромінювань від екрана монітора знаходиться в межах 10...100 мВт/м² (таблиця 4.3).

Таблиця 4. 3

**Припустимі значення параметрів неіонізуючих електромагнітних
випромінювань (відповідно до СанПіН 2.2.2.542-96)**

Найменування параметра	Припустимі
Напруженість електричної складової електромагнітного поля або на відстані 50 см від поверхні відеомонітора	10 В/м
Напруженість магнітної складової електромагнітного поля або на відстані 50 см від поверхні відеомонітора	0,3 А/м
Напруженість електростатичного поля не повинна перевищувати: - для дорослих користувачів - для дітей дошкільних установ і учнів середніх спеціальних і вищих навчальних закладів	20 кВ/м 15 кВ/м

4.2. Розробка заходів для зменшення впливу шкідливих факторів

Організація і поліпшення умов праці на робочому місці є одним найважливіших резервів продуктивності праці й економічної ефективності виробництва, а також подальшого розвитку самої працюючої людини. У цьому головний прояв соціального й економічного значення організації і поліпшення умов праці.

Фарбування приміщень і меблів повинні сприяти створенню сприятливих умов для здорового сприйняття, гарного настрою.

Джерела світла, такі як світильники і вікна, що дають відображення від поверхні екрана, значно погіршують точність знаків і спричиняють перешкоди фізіологічного характеру, які можуть виразитися в значній напрузі, особливо при тривалій роботі. Відображення, включаючи відображення від вторинних джерел світла, повинне бути зведене до мінімуму. Для захисту від надлишкової яскравості вікон можуть бути застосовані штори й екрани.

У залежності від орієнтації вікон рекомендується наступна фарбування стін і підлоги:

- вікна орієнтовані на південь: стіни – зеленувато-блакитний або світло-блакитний кольори; підлога – зелений;
- вікна орієнтовані на північ: стіни – світло-жовтогарячий або оранжево-жовтий кольори; підлога – червонясто-жовтогарячий;
- вікна орієнтовані на схід: стіни – жовто-зеленого кольору; підлога – зелений або червонясто-жовтогарячий;
- вікна орієнтовані на захід: стіни – жовто-зелений або блакитнувато-зелений кольори; підлога – зелений чи червонясто-жовтогарячий.

У приміщеннях, де знаходиться комп'ютер, необхідно забезпечити наступні величини коефіцієнта відображення: для стелі – 60...70%, для стін – 40...50%, для підлоги – близько 30%. Для інших поверхонь і робочих меблів – 30...40%.

Зниження шуму, створюваного на робочих місцях внутрішніми джерелами, а також шуму проникаючого ззовні, є дуже важливою задачею. Зниження шуму в джерелі надходження можна забезпечити застосуванням пружних прокладок між підставою машини, приладу й опорною поверхнею. В якості прокладки використовуються гума, повсть, пробка, різної конструкції амортизатори. Під настільні шумливі апарати можна підкладати м'яку прокладку із синтетичних матеріалів, а під ніжки столів, на яких вони встановлені, – прокладки з м'якої гуми, повсті, товщиною 6-8 мм. Кріплення прокладок можливе шляхом приклейки їх до опорних частин.

Можливе також застосування звукоізолюючих кожухів, що не заважають технологічному процесу. Не менш важливим для зниження шуму в процесі експлуатації є питання правильного і своєчасного регулювання, змазування і заміни механічних вузлів шумливого устаткування.

Раціональне планування приміщення і розміщення устаткування є важливим чинником, що дозволяє знизити шум при існуючому устаткуванні

ЕОМ. Приміщення необхідно розташовувати в далечині від шумливого і вібруючого устаткування.

Зниження рівня шуму, що проникає у виробниче приміщення ззовні, може бути досягнуте збільшенням звукоізоляції конструкцій для захисту, ущільненням по периметру притворів вікон, дверей.

Таким чином для зниження шуму, створюваного на робочих місцях внутрішніми джерелами, а також шуму, що проникає з поза, слід:

- послабити шум самих джерел (застосування екранів, звукоізолюючих кожухів);
- знизити ефект сумарного впливу відбитих звукових хвиль (звуковбирні поверхні конструкцій);
- застосовувати раціональне розташування устаткування;
- використовувати архітектурно-планувальні і технологічні рішення ізоляцій джерел шуму.

Нормалізація мікроклімату приміщення може бути досягнута поліпшенням вентиляції приміщення за рахунок витяжних вентиляторів і нормального функціонування вбудованої вентиляції будинку.

У приміщеннях з обчислювальною технікою, як правило, застосовується бічне природне освітлення. Робочі кімнати і кабінети повинні мати природне освітлення. В інших приміщеннях допускається штучне освітлення.

Освітленість на поверхні столу в зоні розміщення робочого документа повинна бути 300-500 лк, також допускається установка світильників місцевого освітлення для підсвічування документів, але з такою умовою, щоб воно не створювало відблисків на поверхні екрана і не збільшувало освітленість екрана більш ніж на 300 лк.

В якості джерела світла при штучному висвітленні повинні застосовуватися переважно люмінесцентні лампи типу ЛБ. При порядкуванні відбитого освітлення в адміністративно-суспільних приміщеннях допускається застосування металогалогених ламп потужністю до 250 Вт. Допускається застосування ламп накаливання у світильниках місцевого освітлення.

Загальне освітлення варто виконувати у вигляді суцільних або переривчастих ліній світильників, розташованих збоку від робочих місць, паралельно лінії зору користувача при рядному розташуванні ПЕОМ. При периметральному розташуванні комп'ютерів лінії світильників повинні розташовуватися локалізовано над робочим столом, ближче до його переднього краю, зверненого до оператора.

Для забезпечення нормованих значень освітленості в приміщеннях використання ПЕОМ варто проводити чищення стікол віконних рам і світильників не рідше двох разів у рік і проводити своєчасну заміну перегорілих ламп.

4.3. Ергономіка робочого місця

Проектування робочих місць, оснащених відеотерміналами, відноситься до числа важливих проблем ергономічного проектування в області обчислювальної техніки.

Робоче місце і взаємне розташування всіх його елементів повинне відповідати антропометричним, фізичним і психологічним вимогам. Велике значення має також характер роботи. Зокрема, при організації робочого місця програміста повинні бути дотримані наступні основні умови: оптимальне розміщення устаткування, що входить до складу робочого місця і достатній робочий простір, що дозволяє здійснювати всі необхідні рухи і переміщення.

Ергономічними аспектами проектування відеотермінальних робочих місць, зокрема, є: висота робочої поверхні, розміри простору для ніг, вимоги до розташування документів на робочому місці (наявність і розміри підставки для документів, можливість різного розміщення документів, відстань від очей користувача до екрана, документа, клавіатури і т.д.), характеристики робочого крісла, вимоги до поверхні робочого столу, регулюючих елементів робочого місця.

Головними елементами робочого місця програміста є стіл і крісло. Основним робочим положенням є положення сидячи.

Робоча поза сидячи викликає мінімальне стомлення програміста. Рациональне планування робочого місця передбачає чіткий порядок і сталість розміщення предметів, засобів праці і документації. Те, що потрібно для виконання робіт частіше, розташовується в зоні легкої досяжності робочого простору.

Для комфортної роботи стіл повинний задовольняти наступним умовам:

- висота столу повинна бути обрана з урахуванням можливості сидіти вільно, у зручній позі, при необхідності спираючись на підлокітники;
- нижня частина столу повинна бути сконструйована так, щоб програміст міг зручно сидіти, не був змушений підгортати ноги;
- поверхня столу повинна мати властивості, що виключають появу відблисків у полі зору програміста;
- конструкція столу повинна передбачати наявність висувних шухляд (не менше ніж 3 для збереження документації, лістингів, канцелярських предметів);
- висота робочої поверхні рекомендується в межах 680-760 мм. Висота поверхні, на яку встановлюється клавіатура, повинна бути близько 650 мм.

Велике значення надається характеристикам робочого крісла. Так, рекомендована висота сидіння над рівнем підлоги знаходиться в межах 420-550 мм. Поверхня сидіння м'яка, передній край закруглений, а кут нахилу спинки – регульований.

Необхідно передбачати при проектуванні можливість різного розміщення документів: збоку від відеотерміналу, між монітором і клавіатурою і т.п.. Крім того, у випадках, коли відеотермінал має низьку якість зображення, наприклад помітні мелькання, відстань від очей до екрана роблять більше (близько 700 мм), чим відстань від ока до документа (300-450 мм). Узагалі при високій якості

зображення на відеотерміналі відстань від очей користувача до екрана, документа і клавіатури може бути рівною.

Велике значення також надається правильній робочій позі користувача. При незручній робочій позі можуть з'явитися болі в м'язах, суглобах і сухожиллях. Вимоги до робочої пози користувача відеотермінала наступні:

- голова не повинна бути нахилена більш ніж на 20° ,
- плечі повинні бути розслаблені,
- лікті – під кутом $80^{\circ}\dots 100^{\circ}$,
- передпліччя і кисті рук – у горизонтальному положенні.

Причина неправильної пози користувачів обумовлена наступними факторами: немає гарної підставки для документів, клавіатура знаходиться занадто високо, а документи – низько, нікуди покласти руки і кисті, недостатній простір для ніг.

З метою подолання зазначених недоліків даються загальні рекомендації: краще пересувна клавіатура; повинні бути передбачені спеціальні пристосування для регулювання висоти столу, клавіатури й екрана, а також підставка для рук.

Істотне значення для продуктивної і якісної роботи на комп'ютері мають розміри знаків, щільність їхнього розміщення, контраст і співвідношення яскравості символів і фону екрана. Якщо відстань від очей оператора до екрана дисплея складає 60...80 см, то висота знаку повинна бути не менше ніж 3 мм, оптимальне співвідношення ширини і висоти знаку складає 3:4, а відстань між знаками – 15...20% їхньої висоти. Співвідношення яскравості фону екрана і символів – від 1:2 до 1:15.

Під час користування комп'ютером медики радять установлювати монітор на відстані 50-60 см від очей. Фахівці також вважають, що верхня частина відеодисплея повинна бути на рівні очей або трохи нижче. Коли людина дивиться прямо перед собою, її очі відкриваються ширше, ніж коли вона дивиться вниз. За рахунок цього, площа огляду значно збільшується, викликаючи зневоднювання очей. До того ж якщо екран установлений високо, а

очі широко відкриті, порушується функція моргання. Це значить, що очі не закриваються цілком, не омиваються слізною рідиною, не одержують достатнього зволоження, що приводить до їх швидкої стомлюваності.

4.4. Режим праці

Для підтримки тривалої працездатності людини велике значення має режим праці і відпочинку. Під раціональним фізіологічно обґрунтованим режимом праці і відпочинком мається на увазі таке чергування періодів роботи з періодом відпочинку, при якому досягається висока ефективність суспільно-корисної діяльності людини, гарний стан здоров'я, високий рівень працездатності і продуктивності праці.

Ефективність перерв підвищується при сполученні з виробничою гімнастикою або організацією спеціального приміщення для відпочинку персоналу зі зручними м'якими меблями, акваріумом, зеленою зоною і т.п..

Всі професійні користувачі ПК повинні проходити обов'язкові попередні медичні огляди при вступі на роботу, періодичні медичні огляди з обов'язковою участю терапевта, невропатолога і окуліста, а також проведенням загального аналізу крові та ЕКГ.

Не допускаються до роботи на ПК жінки з часу встановлення вагітності та в період годування груддю.

Короткозорість, далекозорість та інші порушення рефракції повинні бути повністю коригувати очками. Для роботи повинні використовуватися окуляри, підібрані з урахуванням робочого відстані від очей до екрана дисплея. При більш серйозних порушеннях стану зору питання про можливість роботи на ПК вирішується лікарем-офтальмологом.

Для зняття втоми акомодаций м'язів і їх тренування використовуються комп'ютерні програми типу Relax.

Інтенсивно працюючим доцільно використовувати такі новітні засоби профілактики зору, як окуляри ЛПО-тренер і офтальмологічні тренажери ДАК і «Снайпер-ультра».

Дозвілля рекомендується використовувати для пасивного та активного відпочинку (заняття на тренажерах, плавання, їзда на велосипеді, біг, гра в теніс, футбол, лижі, аеробіка, прогулянки по парку, лісі, екскурсії, прослуховування музики і т.п.). Двічі на рік (навесні і пізньої осені) рекомендується проводити курс вітамінотерапії протягом місяця. Слід відмовитися від куріння. Категорично має бути заборонено куріння на робочих місцях і в приміщеннях з ПК.

Висновки

Було вивчено основні правила охорони праці для роботи з комп'ютерною технікою та організації робочого місця і часу.

ВИСНОВКИ

В результаті створення в рамках дипломного проекту інформаційної системи муніципальної бібліотеки було повністю виконано технічне завдання і враховані побажання замовника.

Під час роботи над проектом:

- були використані всі необхідні етапи проектування: ескізний, технічний та робочий проекти;
- вивчена предметна галузь та проведено її аналіз на основі чого створено концептуальну і логічну модель інформаційної системи;
- розроблено базу даних бібліотеки;
- розроблено проект сайту муніципальної бібліотеки та успішно реалізовано його на практиці.
- створена інформаційна система виконана в єдиному стилі, має зручний і зрозумілий інтерфейс.

Взаємодія бази даних і сайту протестована. Програма виконує всі завдання, що потребували вирішення в рамках дипломного проекту.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. freshit.ua. URL: [https:// freshit.ua](https://freshit.ua) (дата звернення: 23.03.2025).
2. <http://www.ndu.edu.ua/liceum/web.pdf> (дата звернення: 07.03.2025)
3. <https://dspace.onua.edu.ua/server/api/core/bitstreams/ae28ae9f-d248-4ec9-8ac2-deb56416f8c2/content>
4. <https://dspace.onua.edu.ua/server/api/core/bitstreams/ae28ae9f-d248-4ec9-8ac2-deb56416f8c2/content>
5. <https://serviceclub.com.ua/> URL: [https:// freshit.ua](https://freshit.ua) (дата звернення: 23.03.2025).
6. <https://ube.nlu.org.ua/article/%D0%9F%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BD%D0%B5%20%D0%B7%D0%B0%D0%B1%D0%B5%D0%B7%D0%BF%D0%B5%D1%87%D0%B5%D0%BD%D0%BD%D1%8F#:~:text=%D0%92%20%D0%B1%D1%96%D0%B1%D0%BB%D1%96%D0%BE%D1%82%D0%B5%D0%BA%D0%B0%D1%85%20%D0%A3%D0%BA%D1%80%D0%B0%D1%97%D0%BD%D0%B8%20%D0%B2%D0%B8%D0%BA%D0%BE%D1%80%D0%B8%D1%81%D1%82%D0%BE%D0%B2%D1%83%D1%94%D1%82%D1%8C%D1%81%D1%8F%20%D1%80%D1%96%D0%B7%D0%BD%D0%B5,%2D%20%D0%BC%D0%BE%D0%B4%D1%83%D0%BB%D1%8C%2C%20%C2%AB%D0%90%D0%B4%D0%BC%D1%96%D0%BD%D1%96%D1%81%D1%82%D1%80%D0%B0%D1%82%D0%BE%D1%80%C2%BB>
7. <https://wedex.com.ua/blog/etapi-rozrobki-sajtu/>
8. <https://zakon.rada.gov.ua/laws/show/1556-18>
9. <https://zakon5.rada.gov.ua/laws/show/2145-19>
10. technari.com. URL: [https:// technari.com](https://technari.com) (дата звернення: 03.03.2025).
11. Вовк, Н. С. Автоматизація підсистеми обслуговування читачів у системі електронної бібліотеки / Вовк Наталія Степанівна // Бібліотекознавство. Документознавство. Інформологія. – 2020. – № 2.

- С. 27–34. – Бібліогр.: 9 назв. – Текст статті представлено також в інтернеті: <http://journals.uran.ua/bdi/article/view/221159>.
12. Полякова, Т. М. Вебліографія в практичній діяльності бібліотек. Хмарні сховища / Тетяна Миколаївна Полякова // Джерело пед. інновацій : інформ. технології в б-ці загальноосвіт. навч. закл. – 2015. – Вип. 1 (9). – С. 24–30. – Бібліогр.: 8 назв.
 13. Про бібліотеки і бібліотечну справу: Закон від 27.01.1995 р. № 32/95-ВР. Дата оновлення: 01.01. 2017. Відомості Верховної Ради України (ВВР), 2017. № 7. С. 45. URL: <https://zakon.rada.gov.ua/laws/show/32/95-%D0%B2%D1%80>
 14. Про інформацію: Закон України від 02.10.1992 № 2657-XII. Дата оновлення: 16.07.2019. Відомості Верховної Ради України (ВВР), 1992. № 48. С. 650. URL: <https://zakon.rada.gov.ua/laws/show/2657-12>
 15. Про Національний архівний фонд та архівні установи: Закон України № 3814-XII. Дата оновлення: 21.05. 2015. Відомості Верховної Ради України (ВВР), 1994. № 15. С. 86. URL: <https://zakon.rada.gov.ua/laws/show/3814-12>
 16. Про охорону здоров'я: Закон від 19.11.1992 № 49, ст.19. Дата оновлення: 13.12.2020. URL: <https://zakon.rada.gov.ua/laws/show/2801-12#Text> (дата звернення: 07.03.2025).
 17. Про охорону праці: Закон від 14.10.1992 № 49, ст.668. Дата оновлення: 27.02.2025. URL: <https://zakon.rada.gov.ua/laws/show/2694-12#Text> (дата звернення: 07.03.2025).
 18. Про пожежну безпеку: Закон від 01.07.2013. URL: <https://zakon.rada.gov.ua/laws/show/3745-12#Text> (дата звернення: 07.03.2025).
 19. Про пожежну безпеку: Закон від 27.02.2021. URL: <https://zakon.rada.gov.ua/laws/show/322-08#Text> (дата звернення: 07.03.2025)

- 20.Про стандартизацію: Закон України № 1315-VII. Дата оновлення: 10.02.2016. Відомості Верховної Ради України (ВВР), 2014. № 31. С. 1058. URL: <http://zakon5.rada.gov.ua/laws/show/1315-18>
- 21.Сучасна бібліотека: веб-сервіси сучасної бібліотеки : метод. матеріали / Департамент культури і туризму Харків. облдержадмін., Харків. обл. універс. наук. б-ка ; [укладач Державна науково-педагогічна бібліотека України імені В. О. Сухомлинського НАПН України <http://dnrb.gov.ua/ua/> 51 В. С. Ручка]. – Харків : [ХОУНБ], 2014. – 38 с. – (Сучасна бібліотека). – Текст видання представлено також в інтернеті: <http://library.kharkov.ua/libdruk/LibKh00000000142.pdf>.
- 22.Що таке сайт. URL: <https://www.alkosto.ru/blog/chto-takoe-sajt/> (дата звернення: 03.03.2025).
- 23.Що таке сайт. URL: <https://fo.ru/blog/chto-takoe-sajt/> <https://www.alkosto.ru/blog/chto-takoe-sajt/> (дата звернення: 03.03.2025)..

ДОДАТОК А. ТЕХНІЧНЕ ЗАВДАННЯ НА РОЗРОБКУ «РОЗРОБКА ІНФОРМАЦІЙНОЇ СИСТЕМИ З WEB-ДОСТУПОМ ДЛЯ МУНІЦИПАЛЬНОЇ БІБЛІОТЕКИ»

Вступ

Повне найменування програмного продукту: «Розробка інформаційної системи з WEB-доступом для муніципальної бібліотеки».

Коротка характеристика області застосування: програмне забезпечення планується використовувати для читачів та працівників муніципальної бібліотеки.

1. Підстава для розробки:

Підставою для розробки є виконання завдання бакалаврської кваліфікаційної роботи.

2. Призначення розробки

2.1. Функціональне призначення

Програмне забезпечення для інформаційної системи муніципальної бібліотеки повинно:

- ✓ усунути помилки/необхідність дублювати дані в документах;
- ✓ впровадити багатоцільове використання даних;
- ✓ зробити пошук книжок та інформації в бібліотеці за різними ознаками автоматичним;
- ✓ автоматизувати комплектування фонду бібліотеки;
- ✓ автоматизувати ведення обліку читачів бібліотеки;

- ✓ надати можливість читачам бібліотеки самостійно реєструватися на сайті бібліотеки;
- ✓ автоматизувати формуванням замовлень читачів для отримання необхідної літератури;
- ✓ автоматизувати облік виконання замовлень читачів;
- ✓ автоматизувати ведення бібліотечної статистики.

2.2. Експлуатаційне призначення

Програма призначена для експлуатації на робочих комп'ютерах бібліотеки працівниками бібліотеки так і через мережу Інтернет зареєстрованими користувачами муніципальної бібліотеки.

3. Вимоги до програми або програмного продукту

3.1. Вимоги для функціональних характеристик

3.1.1. Вимоги до переліку виконуваних функцій

Необхідно надати бібліотекарю можливість вирішувати наступні задачі:

- Реєструвати отриманих книг;
- Росподіляти книги різні категорії;
- Можливість управління бібліотечним фондом;
- Створення та управління кабінетом читача;
- Виконувати облік видачі та повернення книг читачами. Видача книг передбачає пошук екземпляру книги, яку хоче взяти читач, серед доступних у бібліотеці з наступним оформленням видачі з вказанням дати видачі та дати повернення. Повернення книги передбачає внесення дати фактичного повернення книги.

3.1.2. Вимоги для організації вхідних та вихідних даних

Вхідними даними є інформація про книгу (її авторів, жанр, видавництво) й персональна інформація про читача.

Вихідні дані будуть мати наступний вигляд:

- Записи про книги (твори у них, їх жанри, видавництва, авторів, тощо);
- Записи про читачів/працівників;
- Записи про видачу/повернення книг.

3.2. Вимоги до надійності

3.2.1. Вимоги до забезпечення надійного функціонування програми.

При виникненні збою в роботі апаратури, відновлення нормальної роботи програми повинне виконуватися після перезавантаження операційної системи. Програма повинна забезпечувати коректну обробку інформації.

Для забезпечення надійної роботи програмного забезпечення також необхідно передбачити можливість резервного копіювання інформації. У разі введення користувачем некоректної інформації система повинна повідомити про помилку і надати можливість виправити її.

3.2.2. Час відновлення після відмови

Час відновлення після відмови, не пов'язаної з роботою програми, повинен складатися з: часу перезапуску операційної системи; часу повторного введення або зчитування з носіїв необхідних даних.

3.2.3. Відмови з причини не коректних дій оператора

Відмови програми можливі з причин некоректних дій користувача при взаємодії з операційною системою. З метою запобігання відмов такого роду слід надати користувачу лише необхідний мінімум прав користувача.

3.3. Умови експлуатації

3.3.1. Кліматичні умови експлуатації

Кліматичні умови експлуатації, при яких повинні забезпечуватися задані характеристики, повинні задовольняти вимогам, що пред'являються до технічних засобів в частині умов їх експлуатації

3.3.2. Вимоги до видів обслуговування

Програма не вимагає проведення будь-яких обов'язкових видів обслуговування.

3.3.3. Вимоги до чисельності та кваліфікації персоналу

Вимоги до чисельності та кваліфікації персоналу не висуваються.

3.4. Вимоги до складу та параметрів технічних засобів

Даний програмний продукт потребує від комп'ютеру, на якому він буде встановлений, наступних характеристик, які треба розглядати як мінімальні:

Операційна система Microsoft Windows Vista та вище;

- Процесор Intel Pentium/Celeron 1800 МГц і вище;
- Оперативна пам'ять 1 Гбайт та вище;
- Вільне місце на диску: 500 Мбайт та більше;

- Відеокарта 256 МБайт та більше.

3.5. Вимоги до технічної та програмної сумісності

3.5.1. Вимоги до інформаційних структур та методів розв’язку

Для зберігання даних та розподілення їх по категоріям необхідно використовувати базу даних. В якості СУБД рекомендовано використати SQLite

3.5.2. Вимоги до захисту інформації та програм

Для захисту інформаційного забезпечення доступ до управління даними надається користувачам залежно від їх статусу у системі (звичайний користувач, працівник бібліотеки, адміністратор)

3.6. Вимоги до маркування та упаковки

Програма буде розповсюджуватись через мережу інтернет без використання фізичних носіїв, тому вимоги до маркування, упаковки, транспортування та зберігання – не висуваються.

3.6.1. Спеціальні вимоги

Програма має забезпечувати взаємодію з користувачем за допомогою графічного інтерфейсу користувача, розробленого згідно з побажаннями замовника

4. Вимоги до програмної документації

4.1. Склад програмної документації має включати в себе

- 1) Технічне завдання.
- 2) Текст програми.
- 3) Програму та методику випробувань.
- 4) Інструкцію користувача.

5. Техніко-економічні показники

Орієнтовна економічна ефективність не розраховується.

6. Стадії та етапи розробки

Розробка має бути проведена в п'ять етапів, перелік етапів наведено в таблиці А.1:

Таблиця А. 1

Стадії та етапи розробки

№ з/п	Назва етапів дипломного проекту (роботи)	Строк виконання етапів роботи	Примітка
1	Аналіз предметної галузі	13.02.2025	
2	Постановка задачі	15.02.2025	
3	Розробка концепції	22.02.2025	

4	Розробка проекту ІС	21.03.2025	
5	Реалізація проекту	25.04.2025	
6	Розробка питань з охорони праці	07.06.2025	
7	Розробка графічних матеріалів	12.06.2025	
8	Подання роботи на перевірку на плагіат	10.06.2025	
9	Подання роботи рецензенту	15.06.2025	
10	Подання роботи на захист	20.06.2025	

ДОДАТОК Б. КОД ПРОГРАМИ

HTML-код верхнього й бічного меню (шаблону сторінок), файл Menus.html:

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-
scale=1.0">
  <link rel="stylesheet" href="/static/css/styleGeneral.css">
  <link rel="stylesheet" href="/static/css/styleHome.css">
  <link rel="stylesheet" href="/static/css/styleCatalogs.css">
  <link rel="stylesheet" href="/static/css/styleLists.css">
  <link rel="stylesheet" href="/static/css/styleBook.css">
  <link rel="stylesheet" href="/static/css/styleAuthorisation.css">
  <link rel="stylesheet" href="/static/css/styleOrder.css">
  <link rel="stylesheet" href="/static/css/styleSearch.css">
  <link rel="stylesheet" href="/static/css/styleFlash.css">
  <script src="/static/js/script.js"></script>

  <link rel="preconnect" href="https://fonts.googleapis.com">
  <link
href="https://fonts.googleapis.com/css2?family=Nunito:ital,wght@0,200..
1000;1,200..1000&family=Plus+Jakarta+Sans:ital,wght@0,200..800;1,200..8
00&display=swap" rel="stylesheet">

  <title>{% block title %}{% endblock %}</title>
</head>
<body>
  <div class="page">
    <header class="header">
      <nav class="header__container">
        <ul class="header__left">
          <li class="header__burger"
onclick="burgerOn()"></li>
          <li class="home"><a href="{{ url_for('home') }}">
BIBLIO </a></li>
        </ul>
```

```

<div class="header__right">
    <form method="get" class="header__search"
name="quickSearch" action="{{ url_for('search') }}">
        <input type="text" placeholder="Поиск.."
class="search-box-input" name="search">
        <button class="search-box-button"
type="submit">#128269</button>
    </form>
    <ul class="header__buttons">
        {#
            {% if userRole == 'user' %}#}
        {#
            <li><a href=""></a></li>#}
        {#
            {% else %}#}
        {#
            <li><a href="{{ url_for('instruments')
}}"></a></li>#}
        {#
            <li><a href="{{ url_for('basket')
}}"></a></li>#}
        {#
            {% endif %}#}
            {% if userAuthorised %}
                <li>
                    <a href="" title="Menu de compte"></a>
                    <ul>
                        <li><a href="{{ url_for('account')
}}">Compte</a></li>
                        <li><a href="{{ url_for('exit')
}}">Exit</a></li>
                    </ul>
                </li>
            {% else %}
                <li><a href="{{ url_for('login') }}"
title="Login"></a></li>
            {% endif %}
        {#
            {% if userRole != 'user' %}#}
        {#
            <li><a href="#"></a></li>#}
        {#
            {% endif %}#}
    </ul>
</div>

```

```

        </nav>
    </header>
    <section class="burger">
        <nav class="burger__bar">
            <ul>
                <li><a href="{{ url_for('catalog')
}}">Каталог</a></li>
                <li><a href="{{ url_for('genres')
}}">Жанри</a></li>
                <li><a href="{{ url_for('auteurs')
}}">Автори</a></li>
                <li><a href="{{ url_for('ouvrages')
}}">Твори</a></li>
            </ul>
            {% if userRole != 'user' %}
            <ul>
                <li><a href="{{ url_for('instruments')
}}">Інструменти</a></li>
                <li><a href="{{ url_for('usersList')
}}">Користувачі</a></li>
                {% if userRole == 'admin' %}
                <li><a href="{{ url_for('workersList')
}}">Працівники</a></li>
                {% endif %}
            </ul>
            {% endif %}
        </nav>
    </section>
    <div class="flash-container">
        {% for cat, msg in get_flashed_messages(True) %}
            <div class="flash {{ cat }}">
                <label class="label-part">{{ msg }}</label>
                <button class="close-button"
onclick="closeButton(this)">x</button>
            </div>
        {% endfor %}
    </div>
    <main class="content">
        <div class="content__container">
            {% block content %}{% endblock %}
        </div>
    </main>
    <footer class="footer">
        <div class="footer__container">
            <nav class="footer__bar">
                <div class="footer__central">

```

```

        <ul class="footer__mess">
            <li><i class="fa fa-instagram">1</i></li>
            <li><i class="fa fa-facebook">2</i></li>
            <li><i class="fa fa-github">3</i></li>
        </ul>
        <div class="footer__names">
            Nataliia TYMOFIEIEVA
        </div>
    </div>
</nav>
</div>
</footer>
</div>
</body>
</html>

```

HTML-код головної сторінки, Home.html:

```

{% extends 'Menus.html' %}
{% block title %} BOOKShop {% endblock %}

{% block content %}
    <div class="main__h1">
        <h1>BIBLIO</h1>
    </div>
    <section>
        <div class="titre-seconde">
            <h2>Про нас</h2>
        </div>
        <div>
            <p> Ми раді вітати вас у нашій бібліотеці, де кожен знайде
щось для себе. Від класики до новинок, від наукових праць до художньої
літератури, наша колекція постійно оновлюється та розширюється. Ми
віримо, що книги мають силу змінювати життя, і прагнемо, щоб кожен міг
знайти тут натхнення та знання. </p>
        </div>
    </section>
{% endblock %}

```

На ній реалізовано анімацію логотипу при наведенні за допомогою CSS у файлі styleHome.css:

```
.main__h1{
    display: block;
    width: 120%;
    align-self: center;
    text-align: center;
    background: url('/static/img/mainh1.png') 0 -15vh / 100% 140% no-
repeat;
    height: 75vh;
    transition: all 0.8s ease 0s;
    transform-style: preserve-3d;
    padding: 20vh;
    margin-bottom: 10vh;
}

.main__h1 h1 {
    display: inline-block;
    font-size: 12vh;
    padding: 7vh 0;
    /*transform: translate(0, 10vh);*/
    backface-visibility: hidden;
    text-shadow: 0 0 25px #ffd736;
    text-align: center;
    width: 70vh;
    height: 35vh;
    border: 7px solid #FAFAFA;
    box-shadow: 0 0 15px 1px rgba(0,0,0, 1), inset 0 0 10px 1px
rgba(0,0,0, 1);
}

.main__h1:hover{
    transform: rotateX(-180deg);
}
```

Викликається ця сторінка за допомогою функції Python Flask:

```
@app.route("/", methods=["POST", "GET"])
def home():
    return render_template("Home.html",
        userAuthorised=session['user']['authorised'],
        userRole=session['user']['role'])
```

Сторінка реєстрації має на собі форму та вже взаємодіє з базою даних за допомогою функцій Python Flask:

HTML-код сторінки реєстрації:

```
{% extends 'Menus.html' %}
{% block title %} Login {% endblock %}
{% block content %}

    <div class="titre-seconde log-res">
        <h2>Реєстрація</h2>
    </div>

    <section class="authorisation">
        <div class="authorisation-deco deco-reg">
            
        </div>
        <form action="{{url_for('registration')}}" method="post"
class="authorisation-form">
            <div class="label-container">
                <label for="Passport">Паспорт (ID)</label>
                <input class="input" type="text" name="Passport"
value="" required/>
            </div>
            <div class="label-container">
                <label for="Surname">Прізвище</label>
                <input class="input" type="text" name="Surname"
value="" required/>
            </div>
            <div class="label-container">
                <label for="Name">Ім'я</label>
                <input class="input" type="text" name="Name" value=""
required/>
            </div>
            <div class="label-container">
                <label for="Email">Email</label>
                <input class="input" type="email" name="Email" value=""
required/>
            </div>
            <div class="label-container">
                <label for="Telephone">Телефон</label>
                <input class="input" type="tel" name="Telephone"
value="" required/>
            </div>
```



```

        <div class="label-container">
            <label for="Password">Пароль</label>
            <input class="input" type="password" name="Password"
placeholder="Мінімум 4 символи" value="" required/>
        </div>
        <input class="input-button" type="submit"
value="Зареєструватися">
    </form>
</section>

{% endblock %}

```

Код Python Flask:

```

@app.route("/Registration", methods=["GET", "POST"])
def registration():
    if request.method == 'POST':
        email = request.form['Email']
        if not dbase.search_by_email(email):
            password = request.form['Password']
            if (len(password) > 3):
                hash = generate_password_hash(password)
                if session['user']['role'] == role[1]:
                    flash("Vous êtes inscrit en tant que responsable.",
category='success')
                    dbase.registration_worker(request.form['Surname'],
request.form['Name'],
                                email,
request.form['Telephone'], hash, 2)
                else:
                    flash("Vous vous êtes inscrit en tant
qu'utilisateur.", category='success')
                    dbase.registration_user(request.form['Surname'],
request.form['Name'],
                                request.form['Email'],
request.form['Telephone'], hash)
                    session['user']['authorised'] = True
                    session['user']['id'] =
dbase.get_user_by_email(email)[0]
                    session.modified = True
                    return redirect(url_for('home'))
            else:

```

```

        flash("Le mot de passe doit être composé d'au moins 4
caractères.", category='error')
    else:
        flash("Un compte avec cet e-mail existe déjà",
category='error')
    return render_template('Registration.html',
userAuthorised=session['user']['authorised'],
userRole=session['user']['role'])

```

```

def search_by_email(self, email):
    try:
        res = self.get_user_by_email(email)
        if not res:
            res = self.get_worker_by_email(email)
    except sqlite3.Error as e:
        print("Error:  " + str(e))
        return False
    return res

```

```

def get_user_by_email(self, email):
    try:
        self.__cur.execute('''SELECT * FROM User
                                WHERE EMAIL = ?''', (email,))
        res = self.__cur.fetchone()
    except sqlite3.Error as e:
        print("Error:  " + str(e))
        return False
    return res

```

```

def new_client(self, nom, name, email, tel):
    try:
        self.__cur.execute("INSERT INTO Client (NOM, PRENOM, EMAIL,
TEL) VALUES(?, ?, ?, ?)",
                                (nom, name, email, tel))
        self.__db.commit()
    except sqlite3.Error as e:
        print("Error" + str(e))
        return False
    return True

```

```

def registration_user(self, nom, name, email, tel, password):
    try:
        self.new_client(nom, name, email, tel)
        clients = self.get_all_clients()
        client = clients[len(clients)-1][0]
        self.__cur.execute("INSERT INTO Utilisateur (PASSWORD,
ID_CLIENT) VALUES(?, ?)",
                                (password, client))
        self.__db.commit()
    except sqlite3.Error as e:
        print("Error" + str(e))
        return False
    return True

def registration_worker(self, nom, name, email, tel, password,
statut):
    try:
        self.__cur.execute("INSERT INTO Personnel (NOM, PRENOM,
EMAIL, TEL, PASSWORD, ID_STATUT) VALUES(?, ?, ?, ?, ?, ?)",
                                (nom, name, email, tel, password,
statut))
        self.__db.commit()
    except sqlite3.Error as e:
        print("Error" + str(e))
        return False
    return True

```

ДОДАТОК В. ІНСТРУКЦІЮ КОРИСТУВАЧА

Інструкція користувача призначена для ознайомлення користувача з порядком виконання операцій за допомогою програмного забезпечення для бібліотеки, щоб досягти бажаного результату при користуванні ним.

Інформація про бібліотеку і книжки її фонду знаходиться на сайті бібліотеки. При вході на сайт користувачу доступна головна сторінка сайту рисунок Б1 з горизонтальним і бічним меню при зверненні до якого будуть виконуватись вказані в інструкції дії.



Рис. В. 1 Головна сторінка сайту

Дані авторизованих і неавторизованих користувачів сайту доступ до інформації на сайті буде різним. Неавторизовані читачі можуть тільки переглядати дані про книги бібліотечного фонду.

Авторизовані користувачів сайту мають доступ до всіх опцій залежно від того до якої категорії вони відносяться (працівники бібліотеки чи читачі). Вважається, що всі авторизовані користувачі є читачами, працівники бібліотеки мають додаткові опції доступу до керування інформацією на сайті.

В інструкція користувача описано дії користувача з порядком виконання операцій для читачів бібліотеки.

Авторизація користувача сайту

Авторизація користувачів сайту виконується при натисканні на кнопку з «іконкою» «Вхід» і введені пароллю, якщо користувач пройшов процедуру авторизації і ввів свої дані до системи. Якщо процедуру реєстрації на сайті не пройдено, йому буде запропоновано ввести особисті дані в форму реєстрації на сайті і створити пароль для подальшого входу до свого кабінету читача. При створенні пароллю потрібно використовувати не менше 6 знаків різног реєстру клавіатури і хоча б 1 цифру. Підтвердження про реєстрацію прийде на вказану електронну скриньку користувача.

Додавання запису про книгу

Для додавання запису про книгу користувач повинен вказати наступну інформацію:

- назву;
- повне ім'я автора (якщо він є);
- жанр;
- вказати інформацію про видавництво: назву, місто, адресу, контактний телефон;
- кількість сторінок;
- ціну (може бути опущено);
- вказати місце розміщення книги: відділ (якщо він є), шафу, полицю.

Після цього користувач повинен натиснути на кнопку «Додати». Запис про книгу буде додано до бази даних системи.

Перегляд записів про книги

Для перегляду записів про книги користувач повинен натиснути на кнопку «Записи про книги». Система відобразить перелік усіх книг, записи про

які зберігаються у базі даних системи у відповідній таблиці на вкладці «Книги».

Пошук книг

Для пошуку книг користувач повинен в поле пошуку вписати критерій пошуку. Критеріями пошуку є назва книги або автор. Пошук книг розпочнеться автоматично при заповненні поля пошуку книг. Результат пошуку буде відображено у відповідній таблиці на вкладці «Книги».

Видалення книги

Для виконання операції видалення книги користувач спочатку повинен знайти цю книгу за допомогою операції перегляду записів про книгу або пошуку книг. Після цього користувач повинен обрати запис про книгу, натиснувши на нього. Обравши книгу необхідно натиснути на кнопку «Видалити». Після цього буде відкрито діалогове вікно, в якому система попросить підтвердити видалення книги. У випадку натискання на кнопку «Видалити», обраний запис про книгу буде видалено з бази даних системи. У випадку натискання на кнопку «Скасувати» змін до бази даних системи не буде зроблено.

Редагування даних про книги

Для виконання операції редагування книги користувач спочатку повинен знайти цю книгу за допомогою операції перегляду записів про книгу або пошуку книг. Після цього користувач повинен обрати запис про книгу, натиснувши на нього. Обравши книгу необхідно натиснути на кнопку «Редагувати». Після цього буде відкрито діалогове вікно, в якому система відобразить перелік даних про книгу, які можна відредагувати: назву, жанр, автора, кількість сторінок, інформацію про видавництво. У випадку натискання на кнопку «Редагувати», обраний запис про книгу буде відредаговано у базі даних системи. У випадку натискання на кнопку «Скасувати» змін до бази даних системи не буде зроблено.

Списання книги

Для виконання операції списання книги користувач спочатку повинен натиснути на опцію списання книг та обрати одну з опцій-причин списання. Списання можливе з наступних причин: списання книг, які не придатні для подальшого зберігання та використання («Планове списання») та списання з причини втрати книги читачем («Втрата книги»).

У випадку вибору опції планового списання система відкриє вікно, в якому користувач повинен вказати інвентарний номер книги та натиснути на кнопку «Знайти». Система виконає пошук вказаного екземпляру книги та відобразить інформацію про нього. У випадку успіху операції пошуку екземпляру книгу користувач може натиснути на кнопку «Виконати списання». Система видалить запис про знайдений раніше екземпляр та створить документ про планове списання книги у форматі .pdf, в якому буде зазначено наступну інформацію про списану книгу:

- назва;
- автор;
- жанр;
- вказати інформацію про видавництво: назву, місто, адресу, контактний телефон;
- кількість сторінок;
- ціну;
- дату та час виконання операції списання.

У виборі опції списання з причини втрати книги читачем, система відкриє вікно, в якому користувач повинен вказати дані читача (ПІБ або телефон) та натиснути на кнопку «Знайти». Система виконає пошук книг, які отримав читач та не повернув, та відобразить цей перелік. Після цього необхідно обрати книгу, яку читач загубив та натиснути на кнопку «Виконати списання». вказаного екземпляру книги та відобразить інформацію про нього». Система видалить запис про знайдений раніше екземпляр та створить документ

про списання книги у зв'язку з загубленням у форматі .pdf, в якому буде зазначено наступну інформацію про списану книгу:

- назва;
- автор;
- жанр;
- вказати інформацію про видавництво: назву, місто, адресу, контактний телефон;
- кількість сторінок;
- ціну;
- дату та час виконання операції списання;
- інформація про читача.

У випадку натискання на кнопку «Скасувати» виконання операції списання книги буде перервано.

Створити «Абонемент читача»

Для створення абонементу читача користувач вносить дані про користувача (ПІБ, дату народження, контактний телефон) у відповідні поля на вкладці створення абонементу читача та натискає на кнопку «Створити абонемент». Вказання телефону для створення запису не є обов'язковим. У випадку вказання валідних даних буде створено абонемент читача.

Перегляд записів про читачів

Для перегляду записів про книги користувач повинен натиснути на кнопку «Записи про читачів». Система відобразить перелік усіх читачів, записи про яких зберігаються у базі даних системи у відповідній таблиці на вкладці «Читацькі абонементи».

Пошук читачів

Для пошуку читачів користувач повинен в поле пошуку вписати критерій пошуку. Критеріями пошуку є назва книги ПІБ або контактний телефон. Пошук

читачів розпочнеться автоматично при заповненні поля пошуку читачів. Результат пошуку буде відображено у відповідній таблиці на вкладці «Читацькі абонементи».

Видалення абонементу читача

Для виконання операції видалення абонементу читача користувач спочатку повинен знайти запис про читача за допомогою операції перегляду записів про читачів або пошуку читачів. Після цього користувач повинен обрати запис,

«Видалити». Після цього буде відкрито діалогове вікно, в якому система попросить підтвердити видалення абонементу читача. У випадку натискання на кнопку «Видалити», обраний запис про читача буде видалено з бази даних системи. У випадку натискання на кнопку «Скасувати» змін до бази даних системи не буде зроблено.

Редагування абонементу читача

Для виконання операції редагування інформації про читача користувач спочатку повинен знайти запис про читача за допомогою операції перегляду записів про читачів або пошуку читачів. Після цього користувач повинен обрати запис,

натиснувши на нього. Обравши запис про читача необхідно натиснути на кнопку «Редагувати». Після цього буде відкрито діалогове вікно, в якому система відобразить перелік інформації про читача, що підлягає редагуванню: прізвище, ім'я, по-батькові, дату народження, контактний телефон. У випадку натискання на кнопку «Редагувати», обраний запис про читача буде відредаговано у базі даних системи. У випадку натискання на кнопку «Скасувати» змін до бази даних системи не буде зроблено.

Видача книг

Для виконання операції видачі книг користувач повинен натиснути на відповідний напис. Після цього буде відкрито вікно, в якому буде поле для пошуку книг, пошуку читачів та таблиця для відображення книг, які отримає читач в результаті виконання операції. Для пошуку книг необхідно у відповідне поле написати назву книги. Пошук книги розпочнеться автоматично при заповненні поля пошуку книги. За результатами пошуку натиснути на іконку «+» для додавання книги до переліку книг до видачі. До переліку можна додати не більше 5 книг.

Після завершення додавання необхідних книги до переліку необхідно знайти читача. Для цього у відповідне поле вказати інформацію про читача (ПІБ), пошук читача розпочнеться автоматично при заповненні поля пошуку читачів.

Після знаходження читача необхідно вказати дату повернення книг. Для оформлення видачі книг необхідно натиснути на кнопку «Оформити видачу»

Повернення книг

Для виконання операції повернення користувач повинен натиснути на відповідний напис. Після цього буде відкрито вікно, в якому буде поле для читача та таблиця для відображення книг, які були отримані читачем. Для пошуку читача необхідно вказати інформацію про читача у відповідне поле вказати інформацію про читача (ПІБ), пошук читача розпочнеться автоматично при заповненні поля пошуку читачів.

За результатом пошуку у таблиці будуть відображені книги, які отримав читач. В тому випадку, коли читач втратив одну з книг, необхідно натиснути на цю книгу в списку та натиснути на кнопку «Оформити втрату». Після цього відкриється вікно в якому можна буде оформити втрату книги. Дії користувача будуть подібними до операції «Списання книги за втратою», окрім пошуку читача.

Далі користувач має натиснути на кнопку «Оформити повернення», після чого книги, які брав читач, знов будуть доступні до отримання.