

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Національний університет «Запорізька політехніка»

Інститут інформатики та радіoeлектроніки,
Факультет комп'ютерних наук і технологій
(повне найменування інституту, назва факультету)

Кафедра програмних засобів
(повне найменування кафедри)

Пояснювальна записка

до дипломного проєкту (роботи)

бакалавр
(ступінь вищої освіти)

на тему_

РОЗРОБКА ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ ОБЛІКУ РОБОТИ
БІБЛІОТЕКИ ВИЩОГО НАВЧАЛЬНОГО ЗАКЛАДУ
SOFTWARE DEVELOPMENT FOR TRACKING THE PERFORMANCE OF
A LIBRARY IN A HIGHER EDUCATION INSTITUTION

Виконав: студент(ка) 4 курсу, групи КНТз-217
Спеціальності 122 Комп'ютерні науки
(код і найменування спеціальності)

Освітня програма (спеціалізація)
Комп'ютерні науки

Разумей Р.І.
(прізвище та ініціали)

Керівник Корнієнко С.К.
(прізвище та ініціали)

Рецензент Неласа Г.В.
(прізвище та ініціали)

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

Інститут, факультет ІРЕ, ФКНТ
 Кафедра програмних засобів
 Ступінь вищої освіти бакалавр
 Спеціальність 122 Комп'ютерні науки
 Освітня програма (спеціалізація) Комп'ютерні науки
(назва освітньої програми (спеціалізації))

ЗАТВЕРДЖУЮ
 Завідувач кафедри ПЗ, д.т.н, проф.
С.О. Субботін
 “ ” 2021 року

З А В Д А Н Н Я
НА ДИПЛОМНИЙ ПРОЄКТ (РОБОТУ) СТУДЕНТА(КИ)

Разумєя Руслана Ігоровича
(прізвище, ім'я, по батькові)

1. Тема проєкту (роботи) Розробка програмного забезпечення обліку роботи бібліотеки вищого навчального закладу
Software Development for Tracking the Performance of a Library in a Higher Education Institution
 керівник проєкту (роботи) Корнієнко Сергій Костянтинович к.т.н., доцент,
(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)
 затверджені наказом закладу вищої освіти від «30» березня 2021 року № 103_
2. Строк подання студентом проєкту (роботи) 12 травня 2021
3. Вихідні дані до проєкту (роботи) рекомендована література, технічне завдання
4. Зміст розрахунково-пояснювальної записки (перелік питань, які потрібно розробити): 1 Вибір і обґрунтування структури системи. 2 Концептуальне моделювання предметної області. 3 Логічне проектування бази даних. 4 Методика роботи користувача з системою.
5. Перелік графічного матеріалу (з точним зазначенням обов'язкових креслень)
Слайди презентації

6. Консультанти розділів проєкту (роботи)

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	прийняв виконане завдання
1-4 Основна частина	Корнієнко С.К., доцент		
Нормоконтроль	Дейнега Л.Ю., ст. викладач		

7. Дата видачі завдання «28» лютого 2021 року.

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів дипломного проєкту (роботи)	Строк виконання етапів проєкту (роботи)	Примітка
1	Дослідження предметної області та постановка задачі проєктування	1 тиждень	ТЗ, Розділ 1
2	Концептуальне проєктування	2 тиждень	Розділ 2
3	Розробка інформаційного забезпечення	3 тиждень	Розділ 3
4	Розробка програмного забезпечення	4-7 тиждень	Розділ 3
5	Розробка інтерфейсу користувача	8-10 тиждень	Розділ 4
6	Оформлення пояснювальної записки	9 тиждень	
7	Захист роботи	9 тиждень	

Студент(ка)

Разумей Р.І.
(підпис) (прізвище та ініціали)

Керівник проєкту (роботи)

Корнієнко С.К.
(підпис) (прізвище та ініціали)

РЕФЕРАТ

Пояснювальна записка до дипломної кваліфікаційної роботи бакалавра: 62 с., 12 табл., 18 рис., 3 дод., 10 джерел

БАЗА ДАНИХ, БІБЛІОТЕКА, КНИГИ, ОБСЛУГОВУВАННЯ, АБОНЕМЕНТ, ЧИТАТАЧІ, СЕРВІС, ЗВІТИ.

Об'єкт проектування – програмне та інформаційне забезпечення автоматизованої системи.

Мета роботи – розробка системи, що забезпечує автоматизацію обліку обслуговування читачів багатoproфільної бібліотеки вищого навчального закладу.

Матеріали, методи та технічні засоби: модель «сутність-зв'язок», СКБД MS Access 2016, персональний комп'ютер з процесором Intel Core 2 Duo під управлінням операційної системи Microsoft Windows 10.

Результати. На основі аналізу предметної області за допомогою моделі «сутність-зв'язок» розроблено концептуальну модель предметної області та розроблено програмний застосунок, призначений для автоматизації обліку роботи бібліотеки вищого навчального закладу.

Висновки. Розроблена система дозволяє спростити та підвищити надійність обліку роботи бібліотеки.

Галузь використання – бібліотеки вищих навчальних закладів.

ABSTRACT

Explanatory note to the diploma qualifying work of the bachelor: 62 pages, 12 tables, 18 figures, 3 appendixes, 10 sources.

DATABASE, LIBRARY, BOOKS, MAINTENANCE, SUBSCRIPTION, READERS, SERVICE, REPORTS.

Object of design - software and information of the automated system. The purpose of the work is to develop a system that provides automation of accounting services to readers of the multidisciplinary library of higher education.

Materials, methods and technical means: «entity-communication» model, MS Access 2016 database, personal computer with Intel Core 2 Duo processor running Microsoft Windows 10 operating system.

Results. Based on the analysis of the subject area with the help of the model "essence-connection" the conceptual model of the subject area is developed and the software application intended for automation of the account of work of library of higher educational institution is developed.

Conclusions. The developed system allows to simplify and increase the reliability of accounting for the library.

Field of use – libraries of higher educational institutions.

ЗМІСТ

	С.
Перелік скорочень та умовних позначок	7
Вступ.....	8
1 Вибір і обґрунтування структури системи.....	10
1.1 Характеристика об'єкту проєктування	10
1.2 Обґрунтування структури системи.....	11
2 Концептуальне моделювання предметної області	13
3 Логічне проєктування бази даних	17
3.1 Вибір і обґрунтування інструментарію проєктування	17
3.2 Формування схеми даних.....	21
3.3 Нормалізація бази даних	25
3.4 Формування запитів	28
3.5 Макроси	32
3.6 Генерація звітів.....	33
4 Методика роботи користувача з системою.....	35
4.1 Розробка інтерфейсу користувача	35
4.2 Меню системи	36
4.3 Формуляр читача	38
4.4 Книги.....	38
4.5 Звіти	39
4.6 Замовлення літератури.....	39
4.7 Довідники	41
4.8 Пошукова система	43
Висновки.....	47
Перелік джерел посилання	48
Додаток А Технічне завдання.....	49
Додаток Б Зразки звітів	53
Додаток В Слайди презентації	56

ПЕРЕЛІК СКОРОЧЕНЬ ТА УМОВНИХ ПОЗНАК

SQL – Structured Query Language (Мова структурованих запитів);

АБД – автоматизований банк даних;

АІПС – інформаційно-пошукова система;

БД – база даних;

ВНЗ – вищий навчальний заклад;

ІК – інтерфейс користувача;

ММД – мова маніпулювання даними;

МОД – мова опису даних;

НФ – нормальна форма;

НФБК – нормальна форма Бойса-Кода;

ОЗП – оперативний запам'ятовуючий пристрій;

ПЗ – програмне забезпечення;

ПО – предметна область;

РБД – реляційна база даних;

СКБД – система керування базами даних.

ВСТУП

Сучасне світове співтовариство прямує до відкритого інформаційного простору, що передбачає відсутність будь-яких перешкод в отриманні інформації. Цей процес супроводжується змінами в усіх сферах людської діяльності, бібліотечна справа – не виняток. Бібліотеки переосмислюють свою роль у сьогодишньому суспільстві, виконуючи водночас свої традиційні функції.

Реалізація освітньої функції завжди була невід’ємною складовою частиною діяльності бібліотек. З іншого боку, функціонування освітніх закладів, реалізація освітніх програм неможливі без опору на бібліотеки та їхні інформаційні ресурси. Таким чином, діяльність таких соціальних інститутів, як бібліотеки та освітні установи, завжди тісно взаємопов’язана. Кардинальні зміни у світовій системі освіти, приєднання України до Болонського процесу впливають на зміни в реалізації бібліотеками освітньої функції [1].

Насамперед це стосується бібліотек вищих навчальних закладів. Державна політика в освітній сфері, необхідність адаптації до впливу мінливого світу формують нову парадигму діяльності бібліотек ВНЗ, вимагають постійного коригування стратегії, цілей і завдань. Модернізація бібліотек ВНЗ, відповідно до напрямів модернізації системи вищої освіти, стає закономірною.

Сучасна бібліотека ВНЗ під впливом інформаційних технологій переживає процес трансформації. Впровадження сучасних освітніх технологій передбачає активне звернення до цілого спектра інформаційних ресурсів. При цьому цільові установки освітніх технологій зумовлюють необхідність ефективного керування цими ресурсами, наявність оптимальних умов доступу до них, високого рівня інформаційної культури студентів. Такі установки є одночасно завданнями інформатизації бібліотек ВНЗ [1].

У сучасному світі глобальної інформатизації визначальним моментом є співіснування друкованих та електронних джерел інформації. а

впровадження сучасних освітніх технологій, що вимагає залучення широкого спектра інформаційних джерел нового типу, додатково стимулює процеси інформатизації бібліотек ВНЗ. Поняття «бібліотечний фонд» трансформується в більш конструктивне і відповідне сучасній місії бібліотеки поняття інформаційних ресурсів.

Даний дипломний проєкт саме й присвячений впровадженню інформаційних технологій в роботу бібліотеки навчального закладу.

ВИБІР І ОБҐРУНТУВАННЯ СТРУКТУРИ СИСТЕМИ

Характеристика об'єкту проєктування

Об'єктом проєктування даного дипломного проєкту є автоматизована інформаційно-пошукова система «Бібліотека». Метою системи є автоматизація обліку обслуговування читачів багатoproфільної бібліотеки.

При розробці системи враховуються бізнес-правила конкретної організації. Тому першим етапом проєктування автоматизованої системи є аналіз предметної області та технічного завдання (додаток А).

В результаті проведеного аналізу були виявлені такі характеристики системи.

Основні функції системи:

- облік читачів;
- облік обслуговування читачів;
- ведення замовлень;
- облік наявної літератури;
- підтримка пошукової системи;
- ведення довідників;
- ведення статистики роботи бібліотеки;
- генерація звітів.

Користувачеві автоматизованої інформаційної системи повинна бути надана можливість вводити, переглядати та редагувати інформацію про книжковий фонд, читачів, обслуговування читачів, а також довідкову інформацію.

Система повинна працювати автономно, тобто для її роботи не потрібно ніяких додаткових програм.

Система повинна бути забезпечена різними поясненнями та повідомленнями користувачеві.

При спробі ввести в якесь поле дані неприпустимого формату система

не повинна їх сприймати і повинна надати після цього можливість користувачеві знову ввести інформацію.

Система повинна бути розрахована на роботу з нею користувачів будь-якої кваліфікації.

Система повинна працювати при наявності операційної системи Windows 10 та СКБД Access 2016.

Обґрунтування структури системи

Структура інформаційно-пошукової системи (АПС) розроблена відповідно до технічного завдання й являє собою багаторівневу структуру, що складається з окремих програмних модулів.

Модульний підхід до проектування програмного забезпечення дозволяє розробнику краще орієнтуватися в своїй системі, а також зробити більш ефективним процес її написання і налагодження [2].

Головним модулем є диспетчер системи, реалізований у вигляді кнопкової форми. При цьому вибір потрібної функції системи здійснюється за допомогою відповідних кнопок.

До підлеглих модулів відносяться:

- «Книги»;
- «Читачі»;
- «Замовлення»;
- «Пошук»;
- «Довідники»;
- «Звіти».

Ці модулі, у свою чергу, включають сукупності окремих програм, реалізованих на Visual Basic або SQL.

Модульний підхід до проектування програмного забезпечення дозволяє розробнику зробити більш ефективним процес її написання та налагодження [2].

Вхідними даними для системи є така інформація:

- книги;
- читачі;
- довідники.

Крім того, в системі необхідно передбачити наявність довідників, призначених, з одного боку, для полегшення введення інформації, а з іншого боку, для контролю цієї інформації, оскільки користувач може вводити тільки дані з довідників. Це підвищує надійність і достовірність системи в цілому [2].

До таких довідників можна віднести, наприклад, інформацію про види документів, відділах бібліотеки, підрозділах навчального закладу, бібліотекарів.

Вихідними даними системи повинна бути така інформація:

- обслуговування читачів;
- статистика роботи за вказаний період.

Вихідна інформація формується в результаті виконання запитів і представляється у вигляді вихідних документів, що виводяться на екран або принтер.

Оскільки користувачами системи є люди, дуже далекі від обчислювальної техніки, необхідно в системі, що розробляється, передбачити засоби, що дозволяють полегшувати введення і коригування даних, а також здійснювати контроль достовірності інформації, що вводиться. З цією метою необхідно обладнати систему дружнім інтерфейсом користувача [2].

2 КОНЦЕПТУАЛЬНЕ МОДЕЛЮВАННЯ ПРЕДМЕТНОЇ ОБЛАСТІ

На етапі концептуального проектування в якості неформальної моделі предметної області (ПО) використовується модель типу «сутність-зв'язок» (ER-модель). Конструктивними елементами моделі служать сутності, атрибути та зв'язки. Основним конструктивним елементом є сутність [2,3].

Користувач описує цікаві для нього об'єкти ПО за допомогою сутностей, потім визначає властивості сутностей, використовуючи атрибути, і, нарешті, описує відповідності між сутностями за допомогою зв'язків [2].

Сутність – це збірне поняття, деяка абстракція реально існуючого об'єкта навколишнього світу, процесу або явища.

Атрибут – це поійменована характеристика сутності. Атрибут є засобом, за допомогою якого визначаються властивості сутності. Атрибути мають сенс тільки у відношенні з обумовленими ними сутностями.

Для ідентифікації конкретних екземплярів сутностей використовуються спеціальні атрибути – ідентифікатори (ключі). Це може бути один або кілька атрибутів, значення яких дозволяють однозначно відрізнити один екземпляр сутності від іншого [3].

Після визначення сутностей і атрибутів виявляються залежності між двома і більше сутностями, зв'язку атрибутів між собою.

Ці залежності в концептуальній моделі описуються за допомогою зв'язків. Зв'язки в якості засобу, за допомогою якого представляються відношення між двома і більше сутностями. Розрізняють три види зв'язку, що позначаються 1: 1 («один до одного»), 1: М («один до багатьох») і М: N («багато до багатьох») [2].

На основі результатів аналізу предметної області розробляється концептуальна модель ПО, яка враховує не тільки інформаційні інтереси користувачів системи, а й інформаційні потреби самої ПО [3].

Концептуальну модель прийнято представляти в графічному вигляді за

допомогою ER-діаграми, яка в наочній формі представляє зв'язку між сутностями і сьогодні є фундаментом моделювання логічних структур даних.

В результаті аналізу предметної області були виявлені такі типи сутностей:

- КНИГИ;
- ЧИТАЧІ;
- ЗАМОВЛЕННЯ;
- ОБСЛУГОВУВАННЯ_ЧИТАЧІВ.

Кожен виділений інформаційний об'єкт (кожна сутність) має свій набір характеристик (атрибутів), представлених у таблицях 2.1-2.4.

Таблиця 2.1 – Атрибути сутності КНИГИ

№п/п	Атрибут
1	Шифр_книги
2	Тематика
3	Вид документу
4	Назва
5	Автори
6	Видавництво
7	Місто
8	Рік
9	Сторінки
10	Ціна
11	Анотація
12	Ключові слова

Таблиця 2.2 – Атрибути сутності ЧИТАЧІ

№п/п	Атрибут
1	Номер
2	Прізвище
3	Ім'я_побатької
4	Кафедра
5	Тел_роб
6	Тел_моб
7	Адреса

Таблиця 2.3 – Атрибути сутності ЗАМОВЛЕННЯ

№п/п	Атрибут
1	Номер
2	ПІБ
3	Підрозділ
4	Дата
5	Автор
6	Назва
7	Виконання

Таблиця 2.4 – Атрибути сутності ОБСЛУГОВУВАННЯ ЧИТАЧІВ

№п/п	Атрибут
1	2
1	Ном
2	Шифр книги
3	Назва
4	Автори
5	Підрозділ

Продовження таблиці 2.4

1	2
6	Дата видачі
7	Видав
8	Дата повернення
9	Прийняв

Сутність ЧИТАЧІ пов'язана з сутностями ЗАМОВЛЕННЯ та ОБСЛУГОВУВАННЯ ЧИТАЧІВ зв'язком 1:М.

Сутність КНИГИ пов'язана з сутністю ОБСЛУГОВУВАННЯ ЧИТАЧІВ зв'язком 1: М.

3 ЛОГІЧНЕ ПРОЕКТУВАННЯ БАЗИ ДАНИХ

3.1 Вибір і обґрунтування інструментарію проектування

Ядром будь-якої системи баз даних (СБД) є система керування базами даних (СКБД). Сучасні СКБД базуються на використанні реляційної моделі даних [3].

СКБД поділяються на 2 основних типи:

- серверні;
- клієнтські.

Прикладом серверних є: Caché, DB2, Firebird, Informix, Ingres, InterBase, MSDE, MS SQL Server, MySQL, mSQL, Oracle, Pervasive SQL, PostgreSQL, Sybase ASE, Sybase ASA, Sybase IQ, Teradata, Mnesia, H2 [3].

Клієнтських : DataFlex, dBase, MS Access, OpenOffice.org Base, Paradox, Sav Zigzag [3].

MySQL – вільна система керування реляційними базами даних з відкритим кодом була створена як альтернатива комерційним системам. MySQL з самого початку була дуже схожою на mSQL, проте з часом вона все розширювалася й зараз MySQL – одна з найпоширеніших систем керування базами даних. Вона використовується, в першу чергу, для створення динамічних веб-сторінок, оскільки має чудову підтримку з боку різноманітних мов програмування [4].

MySQL — компактний багатонитевий сервер баз даних. Характеризується великою швидкістю, стійкістю і простотою використання.

MySQL вважається гарним рішенням для малих і середніх проєктів. Вихідні коди сервера компілюються на багатьох платформах. Найповніше можливості сервера виявляються в UNIX-системах, де є підтримка багатонитевості, що підвищує продуктивність системи в цілому.

Для некомерційного використання MySQL є безкоштовним. Можливості сервера MySQL [4]:

- простота у встановленні та використанні;
- підтримується необмежена кількість користувачів, що одночасно працюють із БД;
- кількість рядків у таблицях може досягати 50 млн.;
- висока швидкість виконання команд;
- наявність простої і ефективної системи безпеки.

OpenOffice Base. Base є невід'ємною частиною OpenOffice і є повнофункціональним додатком управління базами даних, які можуть бути використані як користувачами вдома, так і досвідченими користувачами для створення додатків баз даних і доступу до інформації за допомогою таблиць, запитів, форм і звітів. Base пропонує безліч налаштувань, які допоможуть вам легко і швидко створити дизайн бази даних з використанням власного вбудованого движка HSQL бази даних, а також має можливість підключення до інших СУБД, включаючи MySQL, MS Access, Adabas D і PostgreSQL. Крім того, Base забезпечують підтримку ODBC і JDBC драйверів, що дозволяє підключатися практично до будь-якої бази даних, яка присутня у системі [4].

Kexi. Це окремий додаток, який пропонує зручний інтерфейс для створення баз даних і доступу до інформації з них за допомогою запитів і сценаріїв. Цікава річ про Kexi: він має можливість зберігати дані на серверах баз даних, крім зберігання його на локальній машині. Ще одна цікава особливість в тому, що Kexi він сумісний з величезною кількістю платформ, включаючи Windows, Linux, Mac, Solaris і FreeBSD. Kexi поставляється з багатофункціональним дизайнером, який дозволяє конструювати всі види баз даних; крім того, додаток забезпечує виділений вид SQL для створення більш складних запитів. Також можна імпортувати існуючі дані з Microsoft Access або експортувати дані з бази Kexi на MySQL або популярний PostgreSQL сервера [4].

AxisBase. Вільна система баз даних, що складається з настільних інструментів і сервера бази даних. Вона використовує унікальний підхід до зберігання даних, видимих ззовні серверів баз даних (наприклад, MySQL або

SQL Server) і дозволяє обробляти їх так само, як локально збережені дані. AxisBase спрямована на надання кращого дизайнерського рішення кінцевим користувачам і намагається створити інструменти бази даних, які допоможуть їм у розробці додатків в простій і зручній формі [4].

Кожна з вищезазначених СУБД надає можливості для вирішення задач, поставлених у технічному завданні. Однак використання професійних СУБД в даній роботі не має сенсу через їхню складність та великий об'єм.

Таким чином, вибір був зроблений на користь MS Access, вона зручна у використанні, має дружній інтерфейс, не вимоглива до ресурсів, та забезпечує задовільний рівень швидкодії [5].

MS Access пропонує універсальний набір програмних засобів, які забезпечують широкі можливості для професійних розробників, а також можуть бути легко освоєні початківцями.

Особливості СКБД Access [5]:

- використання даних різних форматів з різних програм в знайомому інтерфейсі. Включення даних з різних джерел. Access 2003 підтримує різні формати даних, у тому числі XML, OLE, ODBC і формат служб Microsoft Windows SharePoint Services;

- зв'язки між бізнес-системами. Можна пов'язати таблиці таким чином, щоб одночасно отримувати доступ до даних з різних баз, працюючи з формами, звітами та сторінками доступу до даних в Access. Крім того, можна пов'язувати таблиці з інших баз даних Access, електронних таблиць Microsoft Excel, джерел даних ODBC, баз даних Microsoft SQL Server та інших джерел;

- максимально ефективно використання корпоративних даних. Можна включити дані Microsoft SQL Server до рішень Access. Можливе використання конструктора збережених процедур для створення і зміни простих процедур, які зберігаються в SQL Server;

- перегляд відомостей про залежності. Можна швидко знаходити таблиці, запити, форми та звіти, які пов'язані залежністю з конкретним об'єктом бази даних;

- допомога в усуненні помилок. Нова функція перевірки помилок позначає загальні помилки у формах і звітах, істотно прискорюючи процес тестування і виправлення. Після того як помилки помічені, користувачеві пропонуються варіанти їх виправлення, що значно економить час і підвищує якість форм і звітів;

- автоматичне відновлення властивостей. При зміні властивості поля в таблиці може бути виконано автоматичне зміна всіх форм і звітів, чий елементи управління пов'язані з нею;

- потужні функції аналізу даних. Можна перетягувати елементи керування у форму Access для створення зведеної таблиці Microsoft PivotTable, зведеної діаграми Microsoft PivotChart або електронної таблиці;

- простота архівування даних. Access 2003 дозволяє зберегти в іншому місці копію бази даних, з якою ви працюєте;

- створення форм і звітів, які можна зберігати в попередніх версіях Access, значно спрощує спільне використання даних;

- більш ефективне спільне використання даних. Є можливість виконання експорту та імпорту даних та створення зв'язку зі списками на вузлах Microsoft Windows SharePoint, де інші члени групи можуть отримувати доступ до них. Для створення таких вузлів необхідна наявність сервера Microsoft Windows Server, на якому виконуються служби Windows SharePoint Services;

- розробка веб-сторінок з використанням потужних програмних засобів. Можна публікувати форми і звіти в Інтернеті і пов'язувати їх з джерелом записів для відображення та оновлення даних у вихідній базі даних і роботи з ними;

- використання сумісних форматів файлів.

Також слід зазначити, що Access підтримує мову програмування Visual Basic for Application, і дозволяє створювати масиви, свої типи даних, викликати DLL-функції, а також за допомогою OLE Automation контролювати роботу додатків [5].

Слід зазначити, що Access добре узгоджується з мовою SQL. Мова структурованих запитів SQL на сьогодні стала де-факто стандартною мовою створення і роботи з базами даних. Практично усі СУБД реляційного типу підтримують цю мову, що дозволяє безболісно зістиковувати бази цих різних форматів і підтримувати технологію відкритого доступу до баз даних ODBC (Open Database Connectivity).

Аналіз наявного на сьогодні інструментарію програмування, призначеного для розробки прикладного програмного забезпечення в області розробки і ведення баз даних, а також аналіз технічного завдання дозволяє зробити висновок, що для автоматизованої системи, що розробляється, найбільш оптимальним буде використання СКБД Microsoft Access версії 2016.

3.2 Формування схеми даних

Першим етапом логічного проектування є розробка за допомогою обраної СКБД ACCESS інформаційного забезпечення системи, тобто схеми БД. Схема даних включає специфікацію таблиць БД і зв'язків між ними.

Схеми базових таблиць наведено в таблицях 3.1-3.5.

Таблиця 3.1 – Відношення Замовлення

Атрибут	Тип даних
Ном заказу	лічильник
Ном_Чит	числовий
ПІБ	текстовий
Підрозділ	текстовий
Дата	дата/час
Вид документу	текстовий
Автор	текстовий
Назва	текстовий
Виконання	логічний

Таблиця 3.2 – Відношення Книги

Атрибут	Тип даних
Шифр	лічильник
Вид документа	текстовий
Назва	текстовий
Автори	текстовий
Видавництво	текстовий
Місто	текстовий
Рік_видання	числовий
Номер_тому	числовий
Кільк_стор	числовий
Ціна	грошовий
Анотація	текстовий
Ключові_слова	текстовий

Таблиця 3.3 – Відношення Читачі

Атрибут	Тип даних
№_чит_билета	лічильник
Прізвище	текстовий
Ім'я_по_батькові	текстовий
Підрозділ/група	текстовий
Телефон_роб	текстовий
Телефон_дом	текстовий
Адреса	текстовий

Таблиця 3.4 – Відношення Місце зберігання

Атрибут	Тип даних
Шифр книги	числовий
Відділ бібліотеки	текстовий
Кількість	числовий

Таблиця 3.5 – Відношення Обслуговування читачів

Атрибут	Тип даних
№ чит_квитка	лічильник
Шифр книги	числовий
Назва	текстовий
Автори	текстовий
Відділ бібліотеки	текстовий
Дата видачі	дата/час
Видав	текстовий
Дата повернення	дата/час
Прийняв	текстовий

Оскільки реляційна база даних являє собою сукупність об'єктних відносин (базових таблиць), після розробки їх структури необхідно встановити зв'язки між ними. Для цього за допомогою Конструктора зв'язків СКБД Microsoft Access необхідно встановити зв'язок між тими полями, в яких міститься загальна інформація. Ці поля можуть мати різні імена, але тип даних і довжина полів, а також (що особливо важливо) інформація в обох полях відповідних записів повинні бути однаковими в обох таблицях [2].

Як правило, зв'язок встановлюється з'єднанням ключових полів таблиць: первинного ключа однієї таблиці і зовнішнього ключа - в іншій. Коли поле однієї таблиці посилається на інше поле (в іншій таблиці), воно носить назву зовнішнього ключа (foreign key); поле, на яке він посилається, носить

назву його батьківського ключа (parent key) [2].

Імена зовнішнього і батьківського ключів можуть бути неоднаковими. Якщо поле є зовнішнім ключем, то воно пов'язане з таблицею, на яку посилається, тобто кожне значення в цьому полі (зовнішньому ключі) безпосередньо пов'язаний із значенням в іншому полі (батьківському ключі). Це означає, що таблиця знаходиться в стані посилальної цілісності [2].

Зовнішній ключ, як і первинний, може бути визначений на будь-якій кількості полів, які розглядаються як єдине ціле (складовою ключ). Зовнішній ключ і батьківський, на який він посилається, мають бути визначені на однаковому безлічі полів (за кількістю полів, типам полів і порядку проходження полів) [2].

Таким чином, основна ідея посилальної цілісності полягає в тому, що всі значення зовнішніх ключів відсилаються до певної рядку батьківського ключа. Це означає, що кожне значення зовнішнього ключа повинно бути представлено в батьківському ключі один і тільки один раз. Як тільки значення розміщується в зовнішньому ключі, батьківський ключ перевіряє для того, щоб переконатися, що таке значення в ньому присутня; в іншому випадку команда відхиляється. Поле ключа, первинного або зовнішнього, перевіряється незалежно від того, додається воно, змінюється або видаляється. Якщо зміна ключа порушує зв'язок, значить, воно порушує цілісність даних, і запит відхиляється [2].

У розробленій базі даних відношення Читачі пов'язана з відношеннями Замовлення і Обслуговування_читачів зв'язком 1: М. Відношення Книги пов'язане з відношенням Обслуговування_читачів зв'язком 1:М. Відношення Місце_зберігання пов'язана з відношеннями Книги та Відділи_бібліотеки зв'язком М: 1. Схему бази даних наведено на рисунку 3.1.

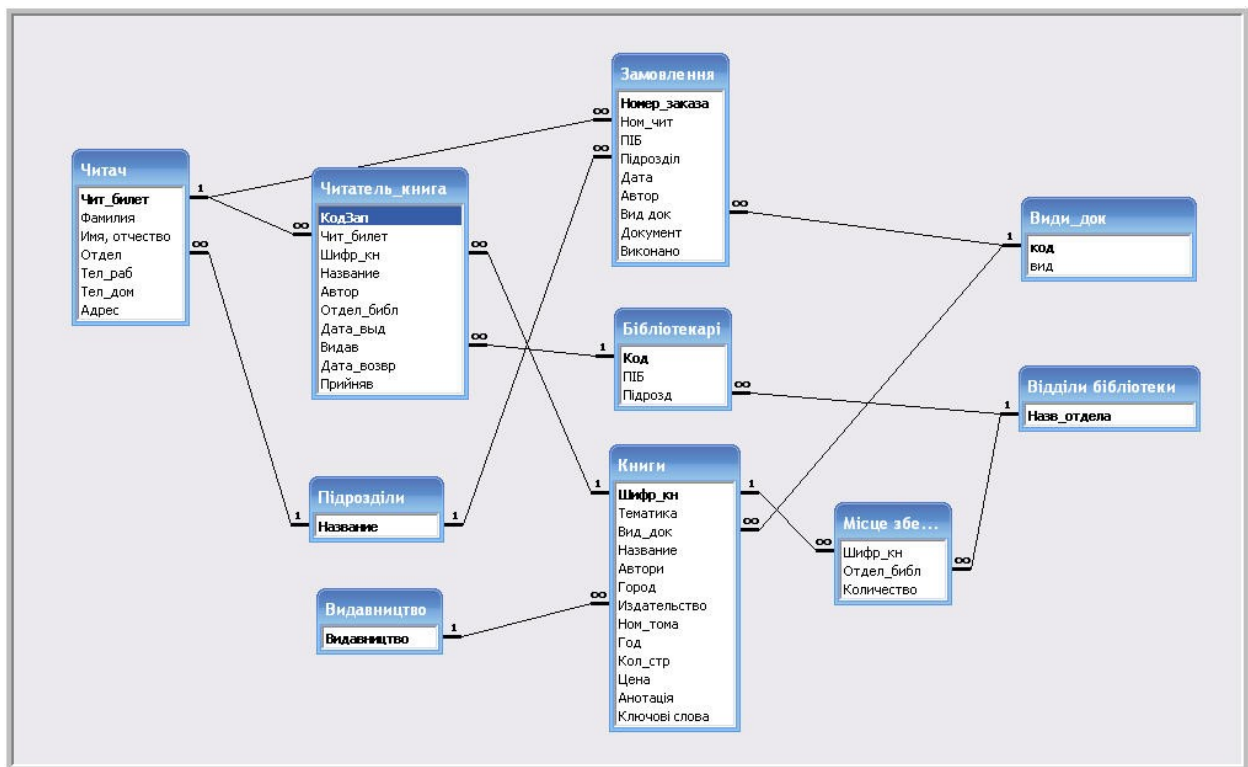


Рисунок 3.1 – Схема бази даних

3.3 Нормалізація бази даних

Основними цілями проектування баз даних є [6]:

- можливість зберігання всіх необхідних даних до бази даних;
- виключення надмірності даних;
- зведення числа збережених в базі даних відношень до мінімуму;
- нормалізація відношень.

Відправною точкою при проектуванні деякої бази даних є складання деякого коректного відношення, що містить всі дані, які передбачається розміщувати в базі даних у майбутньому, і званого універсальним відношенням бази даних [7].

Великий обсяг надлишкових даних у універсальному відношенні усувається на наступних етапах проектування.

Таким чином, перший етап нормалізації полягає в утворенні двовимірної таблиці, яка містить всі необхідні атрибути інформаційної моделі. Дані, представлені у вигляді двовимірної таблиці, у якій все описові атрибути

є атомарними, вважаються поданими в першій нормальній формі (1НФ) реляційної моделі даних.

Використання універсального відношення як завершеної бази даних представляється можливим, але небажаним, внаслідок виникнення аномалій додавання, видалення та оновлення даних [6].

Процес розбиття відношення з метою зменшення ймовірності виникнення аномалій називається декомпозицією. Ключовим для здійснення декомпозиції логічним методичним шляхом є концепція функціональних залежностей між атрибутами в розглянутому відношенні [7].

Якщо дано два атрибута A й B , то кажуть, що B функціонально залежить від A , якщо для кожного значення A існує рівно одне, пов'язане з ним значення B (в будь-який момент часу) [2].

Відношення знаходиться в другій нормальній формі (2НФ), якщо воно є відношенням в (1НФ) і кожен атрибут, який не є первинним ключем в цьому відношенні, залежить від будь-якого можливого ключа цього відношення. Приведення відношення до 2НФ полягає в забезпеченні повної функціональної залежності всіх атрибутів від ключа за рахунок розбиття таблиці на декілька, в яких всі наявні атрибути матимуть повну функціональну залежність від ключа цієї таблиці [2]. Таким чином, в процесі приведення моделі до 2НФ в основному виключаються аномалії дублювання даних.

Відношення знаходиться в третій нормальній формі (3НФ), якщо воно задано в 2НФ і кожен атрибут цього відношення, що не є первинним, не транзитивній залежить від кожного можливого ключа цього відношення. Транзитивне залежність виявляє дублювання даних до одному відношенні. Якщо A , B і C – три атрибута одного відношення і C залежить від B , а B від A , то говорять, що C транзитивній залежить від A [6].

Одним з найбільш важливих результатів в області реляційних баз даних стало доведене Коддом твердження про те, що більшість потенційних аномалій в базі даних буде усунуто в разі належної декомпозиції кожного відношення в нормальну форму Бойса-Кодда (НФБК) [7]. Ця форма

визначається наступним чином: Відношення знаходиться в НФБК, якщо і тільки якщо кожен детермінант відношення є можливим ключем.

Четверта нормальна форма (4НФ) усуває незалежні багатозначні залежності.

Хоча існують нормальні форми більш високого рівня, які накладають навіть сильніші обмеження на розроблювані відношення, на практиці більшість проектувальників намагаються отримати відношення в НФБК або 4НФ [2].

Таким чином, проектування структури БД можна розділити на наступні етапи:

- побудова ненормалізованих відношень;
- нормалізація відношення шляхом декомпозиції з метою приведення моделі до необхідному рівню нормальної форми;
- програмна реалізація нормалізованих відношення.

Аналіз схеми бази даних показує, що у відношеннях БД відсутні залежності описових атрибутів від частини складового ключа, що відповідає вимогам 2НФ.

Крім того, БД знаходиться в 3НФ, оскільки її відношення не містять транзитивні залежності.

У відношеннях БД усі детермінанти є потенційними ключами, що означає відповідність вимогам НФБК.

Кожна книга може зберігатися в декількох відділах, тобто між атрибутами Шифр_книги і Відділ_бібліотеки існує багатозначна залежність. Тому з метою уникнення надмірного дублювання даних необхідно відношення Книги привести до 4НФ шляхом виділення окремої підлеглої таблиці Місце_зберігання. Таким чином, спроектована база даних виявляється приведеної до рівня 4НФ, що є цілком достатнім [7].

3.4 Формування запитів

Наступним етапом логічного проектування БД є розробка користувальницьких застосувань. У нашому випадку це будуть запити та макроси.

Запити створюються за допомогою конструктора запитів СКБД ACCESS 2016 з подальшим коректуванням, при необхідності, за допомогою вбудованої мови структурованих запитів SQL [8].

Основними типами запитів є:

- на вибірку;
- оновлення;
- додавання;
- вилучення.

Як правило, запити на вибірку використовуються самостійно, а запити на зміну мають допоміжний характер і використовуються в складі макросів.

Запит «Замовлення» призначений для відбору інформації про ще невиконані замовлення читачів на літературу.

Текст SQL-запиту:

```
SELECT      Замовлення.Номер_замовлення,      Замовлення.Дата,
Замовлення. Ном_чит, Замовлення.ПІБ, Замовлення.Підрозділ, Замовлення.
[Вид_док], Замовлення.Автор, Замовлення.Документ
FROM Замовлення; FROM ЗАМОВЛЕННЯ WHERE (((ЗАКАЗИ.ном) =
[Forms]! [ЗАМОВЛЕННЯ]! [Ном]));
```

Запит «Кількість виданих книг сьогодні» призначений для підрахунку книг, виданих за поточну дату.

Текст SQL-запиту:

```
SELECT Читач.Книга.Відділ_бібл, Count (Читач.Книга. Шифр_кн) AS
[Кількість книг], Читач.Книга. Дата_вид
FROM Читач.Книга
```

```
GROUP BY Читач.Книга.Відділ_бібл, Читач.Книга.Дата_вид
HAVING (((Читач.Книга.Дата_вид) = Date ());
WHERE (((Замовлення.ном) = [Forms]! [Замовлення]! [Ном]));
```

Запит «Кількість виданих книг» призначений для підрахунку виданих за вказаний період.

Текст SQL-запиту:

```
SELECT Читач.Книга.Відділ_бібл, Count (Читач.Книга. Шифр_кн) AS
[Кількість книг], Читач.Книга.Дата_вид
FROM Читач.Книга
GROUP BY Читач.Книга.Відділ_бібл, Читач.Книга.Дата_вид
HAVING (((Читач.Книга.Дата_вид)> = [Введіть початкову дату] And
(Читач.Книга.Дата_вид) <= [Введіть кінцеву дату]));
```

Запит «Кількість обслугованих читачів» призначений для підрахунку обслугованих читачів за вказаний період.

Текст SQL-запиту:

```
SELECT Читач.Книга.Відділ_бібл, Count (Читач.Книга. Чіт_білет) AS
[Кількість Читачів], Читач.Книга.Дата_вид
FROM Читач.Книга GROUP BY Читач.Книга.Відділ_бібл,
Читач.Книга.Дата_вид
HAVING (((Читач.Книга.Дата_вид)> = [Введіть початкову дату] And
(Читач.Книга.Дата_вид) <= [Введіть кінцеву дату]));
```

Запит "Пошук за ключовим словом" призначений для пошуку літератури за вказаною ключовим словом.

Текст SQL-запиту:

```
PARAMETERS [Введіть ключовими словами] Text (255);
SELECT Книги.Шифр_кн, Книги.Названіє, Книги.Автори,
Віді_док.від, Книги. [Ключові слова], [Місце зберігання] .Відділ_бібл,
```

Книги.Місто, Книги.Видавництво, Книги.Рік, Книги.Кільк_стор

```
FROM (Види_док INNER JOIN Книги ON Види_док.код =
Книги.Вид_док) INNER JOIN [Місце зберігання] ON Книги.Шифр_кн =
[Місце зберігання].Шифр_кн
```

```
WHERE (((Книги. [Ключові слова]) Like "*" & [Введіть ключовими
словами] & "*"));
```

Запит «Пошук за назвою» призначений для пошуку літератури за неповною назвою.

Текст SQL-запиту:

```
SELECT Книги. *, Книги.Назва
FROM Книги
WHERE (((Книги.Назва) Like "*" & [Введіть ключовими словами
назви] & "*"));
```

Запит «Пошук за автором» призначений для пошуку літератури за автором.

Текст SQL-запиту: SELECT Книги.Шифр_кн, Книги.Від_док, Книги.Автори, Книги. Назва, Книги.Іздательство, Книги.Город, Книги.Год, Книги.Кол_стр, [Місце зберігання].Відділ_бібл

```
FROM Книги INNER JOIN [Місце зберігання] ON Книги.Шифр_кн =
[Місце зберігання].Шифр_кн
```

```
WHERE (((Книги.Автори) Like "*" & [Введіть автора] & "*"));
```

Запит "Список боржників" призначений для видачі інформації про співробітників, які не складуть вчасно (протягом 30 днів) літературу.

Текст SQL-запиту:

```
SELECT Читач.Книга.Дата_вид, Читач.Фамілія, Читач. [Ім'я, по
батькові], Читач.Відділ, Читач.Тел_раб, Читач.Книга.Шифр_кн,
Читач.Книга.Відділ_бібл FROM Читач INNER JOIN Читач.Книга ON
```

Читач.Чіт_білет = Читач.Книга.Чіт_білет

WHERE (((Читач.Книга.Дата_возвр) Is Null) AND ((Date () -
[Читач.Книга]! [Дата_вид])> 30));

Запит «Формуляр» призначений для видачі інформації про обслуговування конкретного читача.

Текст SQL-запиту:

SELECT Читач.Чит_квиток, Читач.Прізвище, Читач. [Ім'я, по батькові],
Читач.Книга. Назва, Читач.Відділ, Читач.Книга.Автор, Читач.Книга.
Шифр_кн, Читач.Книга.Дата_вид, Читач.Книга. Читач.Книга.Дата_возвр,
Читач.Книга.Прийняв

FROM Читач INNER JOIN Читач.Книга ON Читач.Чіт_білет =
Читач.Книга.Чіт_білет

WHERE (((Читач.Чіт_білет) = [Forms]! [Читач]! [Чіт_білет]));

Запит на оновлення «Вставка книги та автора в формуляр» призначений для автоматичного заповнення полів назву і автор книги таблиці Читач-книга.

Текст SQL-запиту:

UPDATE Книги INNER JOIN Читач.Книга ON Книги.Шифр_кн =
Читач.Книга.Шифр_кн

SET Читач.Книга.Названіє = Книги.названіє, Читач.Книга. Автор =
Книги.Автори

WHERE (((Читач.Книга.КодЗап) = [Forms]! [Читач]! [Підпорядкована
форма Читач.Книга]! [КодЗап]));

Запит на оновлення «Вставка прізвища в замовлення» призначений для автоматичного заповнення полів прізвище читача і місця роботи таблиці Читач-книга.

Текст SQL-запиту:

```
UPDATE (Підрозділі INNER JOIN Читач ON Підрозділи.Найменування
= Читач.Відділ) INNER JOIN Замовлення ON Читач.Чіт_білет =
Замовлення.Ном_чіт

SET Замовлення.ПІБ = Читач.фамілія, Замовлення.Підрозділ =
Читач.Відділ

WHERE (((Замовлення.Номер_заказа) = [Forms]! [Замовлення]!
[Номер_ замовлення]) AND ((Замовлення.Ном_чіт) = [Forms]! [Замовлення]!
[Ном_чіт]));
```

Запит на видалення «Видалення виконаних замовлень» призначений для видалення замовлень, у яких поле Виконано має значення Увімкнено.

Текст SQL-запиту:

```
DELETE      Замовлення.Номер_заказа,      Замовлення.Ном_чит,
Замовлення.ПІБ, Замовлення.Підрозділ, Замовлення.Дата, Замовлення.Автор,
Замовлення.[Вид док], Замовлення.Документ, Замовлення.Виконано

FROM Замовлення

WHERE (((Замовлення.Виконано)=On));
```

3.5 Макроси

Макрос – це прикладна програма, написана на мові Visual Basic. У СКБД ACCESS є вбудований конструктор макросів, що дозволяє легко проектувати макроси, не вдаючись до прямого написання програм. Призначення використаних макросів приведено в таблиці 3.6.

Таблиця 3.6 – Призначення макросів

Назва	Призначення
Вставка книги та автора до формуляру	Автоматична вставка назви та авторів книги до формуляру після вводу шифру книги
Заказ_вставка читача	Автоматична вставка прізвища читача та місця роботи після вводу № читацького квитка
Вилучання замовлення	Вилучання відомостей про виконані замовлення

3.6 Генерація звітів

У розробленій в дипломному проєкті автоматизованій системі передбачена видача звітів, призначення яких описано в таблиці 3.7

Таблиця 3.7 – Звіти

Назва	Призначення
Список боржників	Список читачів не здали книги більше 30 днів
Замовлення	Інформація про невиконані замовлення літератури
Кількість книг	Кількість виданих книг за період
Кількість читачів	Кількість читачів за період
Розподіл читачів	Діаграма розподілу читачів по відділах бібліотеки за період
Пошук за автором	Результати пошуку книг за автором
Пошук за ключовим словом	Результати пошуку книг за ключовим словом
Пошук за назвою	Результати пошуку книг за неповним назвою
Кількість виданих книг	Кількість виданих книг за вказаний день

Всі звіти реалізовані за допомогою вбудованого в СКБД ACCESS

конструктора звітів на підставі запитів, написаних на мові SQL. При бажанні звіт може бути роздрукований. Приклади звітів наведені в Додатку Б.

МЕТОДИКА РОБОТИ КОРИСТУВАЧА З СИСТЕМОЮ

Розробка інтерфейсу користувача

Під інтерфейсом користувача (ІК) розуміється комплекс засобів, які надаються прикладною програмою для керування нею, перегляду результатів її роботи, а також організації діалогу з користувачем [9].

Елементи користувацького інтерфейсу [9]:

- засоби відображення інформації, пристрої та технології введення даних;
- відображувана інформація, формати і коди;
- командні режими, мова користувач-інтерфейс;
- сценарії діалогу та самі діалоги;
- зворотний зв'язок з користувачем;
- підтримка прийняття рішень у конкретній ПО;
- порядок використання програми і документація на неї.

Проектування ІК перетворилося на самотійну проблему, яка найчастіше перевершує по складності проблему розробки кодів програми, і вимагає, як і процес проектування будь-якої складної системи, відповідних методів, засобів, і, природно, зусиль кваліфікованих фахівців [10].

Поняття дружності – це вектор, який містить відповідно до міжнародної класифікації сім вимог до ІК [10]:

- відповідність завдань, що вирішуються користувачем;
- легкість застосування;
- керованість;
- відповідність очікуванням користувача;
- стійкість до помилок;
- адаптованість / індивідуалізоване.

Більшість методів проектування, що реалізують ту чи іншу методологію, сьогодні тяжіє до технологічних рішень та інструментальних засобів, і здебільшого орієнтовано на реалізацію технічних рішень. При цьому

майже не приділяється уваги розгляду таких важливих завдань, пов'язаних з проектуванням, як збір та аналіз інформації, синтез і оцінка проектних рішень на ранніх етапах проектування. А між тим, інструментальні програмні засоби є лише інструментом і самі по собі не гарантують правильних проектних рішень [10].

Інтерфейс за допомогою інструментальних засобів «збирається» з готових елементів – «кубиків». Однак з одних і тих же кубиків можна «побудувати» інтерфейси різної якості та з різними ергономічними властивостями. Тут багато що залежить від проектувальників, їх знань, досвіду і кваліфікації в області організації людино-комп'ютерної взаємодії [10].

При розробці користувальницького інтерфейсу автоматизованої системи обліку роботи бібліотеки приділялася увага вимогам ергономіки та питань художнього сприйняття програмного продукту.

Так, форми для введення даних системи використовують сірі, сині та зелені відтінки, які не стомлюють очі користувача.

Система надає користувачеві різні пояснення та повідомлення. При введенні даних неприпустимого формату система їх не сприймає. Після цього користувачеві буде надана можливість знову ввести інформацію. При виконанні певних дій система запитує користувача про їхнє підтвердження, виключаючи таким чином помилки.

Вищевикладені чинники значно полегшують роботу користувача, підвищуючи тим самим продуктивність його праці. Оскільки система працює в середовищі WINDOWS, то всі налаштування інтерфейсу – кольорової палітри, розмірів і видів шрифтів – залежать від налаштувань в самому середовищі WINDOWS.

Весь інтерфейс програми розроблений за принципом діалогу з користувачем. Він складається з форм: форми меню, форми для введення і редагування даних і діалогові форми.

Інтуїтивно зрозумілий інтерфейс та проста система меню дозволяють користувачеві легко оволодіти навичками роботи з системою.

Меню системи

Після запуску застосунку на екран виводиться Головна форма – меню системи (рис. 4.1).



Рисунок 4.1 – Монітор системи

На формі є такі кнопки:

- «Формуляр читача»;
- «Книги»;
- «Звіти»;
- «Замовлення літератури»;
- «Довідники»;
- «Пошукова система»;
- «Вихід із системи».

Формуляр читача

Викликається однойменною кнопкою. Відповідна форма (рис. 4.2) містить відомості про читача: прізвище, ім'я, по батькові, місце роботи, домашня адреса й телефони. Крім того, на цій формі є підлегла форма «Формуляр читача», призначена для введення інформації про обслуговування даного читача. При видачі книги вводиться тільки її шифр, а назва та автори заповнюються автоматично.

Шифр_кн	Назва	Автор	Відділ_бібл	Дата_вид
681.3 Г68	Эффективная работа с СУБД	Горев А.П., Макашарипов А.П.	науковий абонемент	01.01.20
681.3 Д40	Проектирование реляционных баз дан	Джексон Дж.	науковий абонемент	02.02.20
681.3.06 Г1	СамоучительVBA	Абель П.	науковий абонемент	03.03.20
681.3 П50	Теория автоматизированных банков	Полищук П.Д., Хон П.А.	науковий абонемент	04.04.20
681.3.06 Н1	Разработка приложений в Access 97	Нортон П.	науковий абонемент	05.05.20
681.3.06 п2	С/С++. Структурное программирован	Джексон Дж.	науковий абонемент	08.09.20
681.3.06 Г2	Программирование в Mirosoft Office	Нортон П.	науковий абонемент	03.10.20
681.3 Д40	Проектирование реляционных баз дан	Джексон Дж.	науковий абонемент	03.10.20
681.3.06 Д1	MS-DOS	Абель П.	науковий абонемент	15.12.20
*				

Рисунок 4.2 – Формуляр читача

Книги

Викликається однойменною кнопкою. Відповідна форма (рис. 4.3) містить бібліографічний опис книги, а також відомості про відділи бібліотеки, де вона зберігається. Крім того, на формі є кнопка пошуку потрібної книги.

Книги

Шифр_кн: 681.3 Г68 **Автори:** Горев А.П., Макашарипов А.П.

Вид_док: навчальний посібник Том: Стор: 704

Назва: Эффективная работа с СУБД Рік: 2018

Видавництво: Питер Пресс Місто: С.-Петербург Ціна: 120,00 грн.

Анотація

Розглядаються основи побудови БД і основні засоби підвищення продуктивності інформаційних систем

Місце зберігання

Відділ	Кіл
науковий абонемент	5
читальний зал	5
*	0

Пошук

Ключові слова: БД, СКБД

Запись: 1 из 11 Нет фильтра Поиск

Рисунок 4.3 – Форма «Книги»

Звіти

Дана форма (рис. 4.4) викликається однойменною кнопкою. Містить кнопки генерації необхідних звітів. Зразки звітів наведені в додатку В пояснювальної записки.

Замовлення літератури

Відповідна форма (рис. 4.5) призначена для оформлення довгострокових замовлень читача на необхідну літературу, тобто літературу відсутню в даний момент у фонді.

ЗВІТИ

СПИСОК БОРЖНИКІВ	ЗАМОВЛЕННЯ
КІЛЬКІСТЬ ВИДАНИХ КНИГ ЗА ДЕНЬ	КІЛЬКІСТЬ ВИДАНИХ КНИГ ЗА ПЕРІОД
ДІАГРАМА ОБСЛУГОВУВАННЯ ЗА ПЕРІОД	КІЛЬКІСТЬ ЧИТАЧІВ ЗА ПЕРІОД



Рисунок 4.4 – Форма «Звіти»

Замовлення літератури

№ замовлення **Дата**

Ном_чит **ПІБ**

Підрозділ

Вид документу **Автор**

Назва

Виконано ☐

Вилучання виконаних замовлень

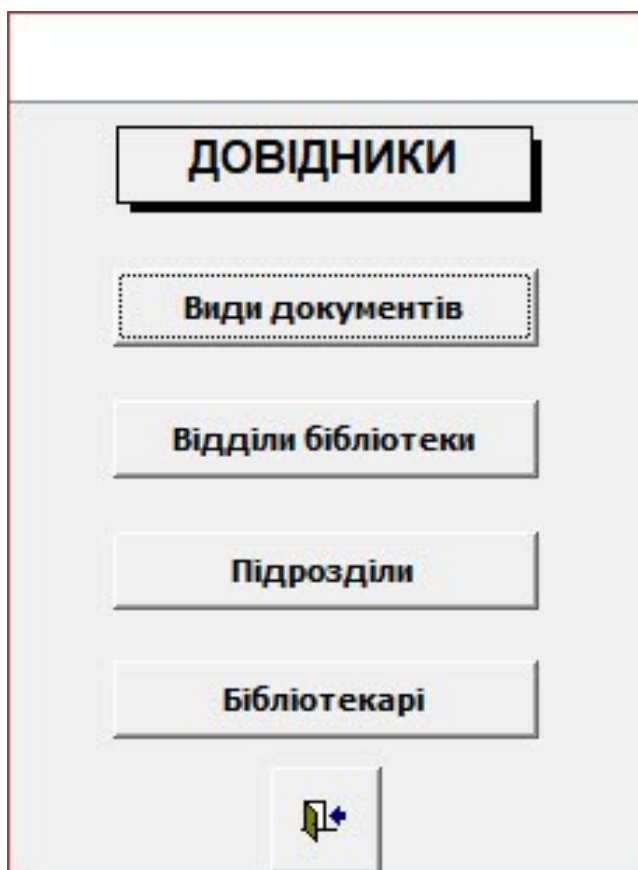
Запись: 1 из 10 Нет фильтра

Рисунок 4.5 – Форма «Замовлення літератури»

Після виконання замовлення включається прапорець Виконано. На формі є кнопка видалення виконаних замовлень. Видаляються ті замовлення, у яких включений відповідний прапорець.

Довідники

Дана форма (рис. 4.6) призначена для введення та редагування довідкових даних (рис. 4.7-4.10).



The image shows a software interface for managing reference data. It features a light gray background with a white title bar at the top containing the text 'ДОВІДНИКИ'. Below the title bar, there are five rectangular buttons with black text, stacked vertically. The buttons are labeled: 'Види документів', 'Відділи бібліотеки', 'Підрозділи', 'Бібліотекарі', and a bottom button with a folder icon and a blue plus sign.

Рисунок 4.6 – Форма «Довідники»

код	вид
1	методичні вказівки
2	конспект лекцій
3	монографія
4	навчальний посібник
5	словники

Запись: 1 из 6 Нет фильтра

Рисунок 4.7 – Форма «Види документів»

Відділ бібліотеки
абонемент молодших курсів
абонемент старших курсів
науковий абонемент
читальний зал
читальний зал періодики

Запись: 1 из 5

Рисунок 4.8 – Форма «Відділи бібліотеки»

Підрозділи	
Название	
	кафедра економіки
	кафедра захисту інформації
	кафедра комп'ютерних мереж
...	кафедра зварювання
	кафедра програмних засобів
	кафедра філософії
	кафедра філософії

Записи: 4 из 7

Рисунок 4.9 – Форма «Підрозділи»

Бібліотекарі		
Код	ПІБ	Підрозд
1	Петренко А.М.	абонемент старших курсів
2	Овчаренко О.О.	науковий абонемент
3	Паньків О.Р.	науковий абонемент
*(N°)		

Записи: 1 из 3 Нет фильтра Поиск

Рисунок 4.10 – Довідник співробітників бібліотеки

Пошукова система

Призначена для пошуку літератури за різними параметрами (рис. 4.11):

- за автором;
- за ключовим словом;

– за частиною назви.

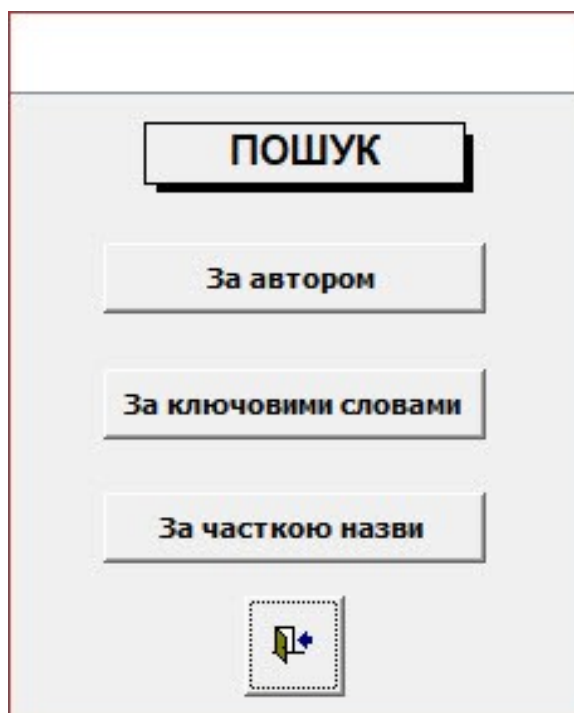


Рисунок 4.11 – Меню пошукової системи

Після натискання на кнопку «За автором» на екран виводиться форма (рис. 4.12) для введення прізвища автора. Після введення прізвища на екран виводяться результати пошуку (рис. 4.13).

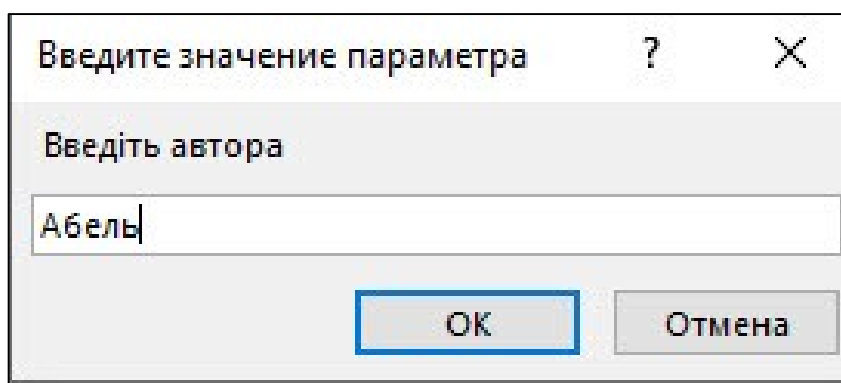


Рисунок 4.12 – Запит прізвища автора

Пошук за автором

Шифр_кн: 681.3.06 Г1 Вид_док: монографія

Автори: Абель П.

Назва Самоучитель VBA

Видавництво ВНУ Місто СПб

Рік 2012 Стор. 367

Запись: 1 из 2 Нет фильтра Поиск

Рисунок 4.13 – Результат пошуку за автором

Після натискання на кнопку «За ключовими словами» на екран виводиться форма (рис. 4.14) для введення ключового слова. Після введення слова на екран виводяться результати пошуку (рис. 4.15).

Введите значение параметра ? X

Введіть ключове слово

СУБД

ОК Отмена

Рисунок 4.14 – Запит ключового слова

Пошук за ключевыми словами

Шифр_кн: 681.3.06 A23 Вид_док:

Автори: Джексон Дж.

Назва C/C++. Структурное программирование. Практикум

Видавництво ВНУ Місто Київ

Рік 2014 Стор. 200

Запись: 5 из 5 Нет фильтра Поиск

Рисунок 4.15 – Результат пошуку за ключовим словом

Після натискання на кнопку «За назвою» на екран виводиться форма (рис. 4.16) для введення ключового слова. Після введення слова на екран виводяться результати пошуку (рис. 4.17).

Рисунок 4.16 – Запит ключового слова назви

Рисунок 4.17 – Результат пошуку за часткою назви

Інтерфейс системи відповідає вимогам ергономіки та є зручним для користувачів.

ВИСНОВКИ

В даному дипломному проєкті були виконані такі завдання:

- аналіз предметної області;
- розробка концептуальної моделі предметної області;
- вибір інструментарію для розробки системи;
- розробка користувальницького інтерфейсу системи;
- наповнення системи реальними даними та налагодження.

Практичним результатом проведеної роботи є створення автоматизованої інформаційно-пошукової системи, призначеної для використання в бібліотеках вищих навчальних закладів.

З розробленою системою може працювати будь-який користувач, який має мінімальні навички для роботи з комп'ютером.

При розробці системи враховувалися вимоги ергономіки, що знайшло відображення в раціональній системі меню, зручних формах введення інформації, системі підказок при роботі користувача, інтуїтивно зрозумілій програмному інтерфейсі.

Випробування системи показали, що в цілому вона має високу ефективність. Створений програмний продукт відповідає всім вимогам замовника.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Шилюк О. Діяльність вузівської бібліотеки в умовах інформатизації суспільства: наук.-практ. видання / О. Шилюк // Вища школа. – 2008. – № 5. – С. 52–54.
2. Корнієнко С.К. Проектування інформаційного забезпечення автоматизованих систем. Навч. посібник / С.К. Корнієнко. – Запоріжжя: ЗНТУ, 2015. – 210 с.
3. Сравнительный анализ СУБД [Электрон. ресурс]. – Режим доступа: <https://sibac.info/studconf/tech/xxxvi/43413>.
4. Шварц Б. MySQL. Оптимизация производительности. – СПб: Символ-Плюс, 2010. – 832 с.
5. Днепроv А. Microsoft Office Access 2007. Самоучитель / А. Днепроv. – СПб.: Питер, 2008. – 432 с.
6. Пасічник В.В. Організація баз даних та знань / В.В. Пасічник, В.А. Резніченко. – К.:ВНУ, 2006. – 384 с.
7. Дейт К. Введение в системы баз даних: Пер. с англ. / К. Дейт. – К.:Диалектика, 1998. – 781 с.
8. Боуман Д. Практическое руководство по SQL / Д.Боуман, С.Эмерсон, М.Дарновски. - Киев: Диалектика, 1997. – 320 с.
9. Корнієнко С.К. Системи баз даних: організація та проектування Навч. посібник /С.К. Корнієнко. – Запоріжжя: ЗНТУ, 2006. – 252 с.
10. Поморова, О.В. Проектування інтерфейсів користувача. Навч. посіб. / О.В. Поморова, Т.О. Говорущенко. – Хмельницький : ХНУ, 2011. – 206 с.

ДОДАТОК А
Технічне завдання

Вступ

Назва системи – «БІБЛІОТЕКА».

Область застосування – бібліотеки вищих навчальних закладів.

А.1 Підстава для розробки

Підставою для розробки є завдання на дипломну кваліфікаційну роботу на тему «Розробка програмного забезпечення обліку роботи бібліотеки вищого навчального закладу», затверджений наказом № 103 від 30 березня 2021 р. по Національному університету «Запорізька політехніка».

А.2 Призначення розробки

АППС призначена для автоматизації обліку обслуговування читачів багатoproфільної бібліотеки.

А.3 Основні вимоги до системи

А.3.1 Вимоги до функціональних характеристик

Система повинна виконувати такі функції:

- облік читачів;
- облік обслуговування читачів;
- ведення замовлень;
- облік наявної літератури;
- підтримка пошукової системи;
- ведення довідників;
- ведення статистики роботи бібліотеки;
- генерація звітів.

Вхідними даними для системи є така інформація:

- читачі;
- книги;
- довідкова інформація.

Вихідними даними системи повинна бути така інформація:

- обслуговування читачів;
- статистика роботи за вказаний період.

А.3.2 Вимоги до надійності

Програма повинна розпізнавати ситуації нерозуміння програмою користувача (користувачем програми), а також обробку збійних ситуацій.

А.3.3 Вимоги до умов експлуатації

Інформаційна система може бути перенесена на будь-якому з типів накопичувачів, ємністю не менше 1 Гб.

Експлуатація інформаційної системи відбувається згідно експлуатаційної документації на розроблену систему, яка відповідає стандартам і містить інформацію, необхідну для освоєння і експлуатації системи.

А.3.4 Вимоги до складу і параметрів технічних засобів

У якості апаратно-технічних засобів для експлуатації системи повинен використовуватися ІВМ-сумісний комп'ютер з характеристиками не нижче: процесор з частотою 800MHz, RAM 4 Гб, Video 32 Мб, HDD 200 Гб.

Інше обладнання для функціонування інформаційної системи не обов'язково.

A.3.5 Вимоги до маркування та упаковки

Система може поставлятися на оптичних дисках. На упаковці має бути назва системи «Ательє пошиття» та системні вимоги.

A.3.6 Вимоги до транспортування і збереження

Вимоги до транспортування і зберігання аналогічні вимогам, пред'явленим до оптичних дисків.

ДОДАТОК Б
Зразки звітів

Боржники						
ПІБ	Підрозділ	Дата_выд	Тел_роб	Відділ бібліотеки	Документ	
Сидоренко Іван Іванович	кафедра програмних засоб	15.12.2020	2-67	надуковий абонемент	MS-DOS	
		03.10.2020	2-67	надуковий абонемент	Проектирование реляционных баз данных для использования с мик	
		03.10.2020	2-67	надуковий абонемент	Программирование в Microsoft Office	
		08.09.2020	2-67	надуковий абонемент	C/C++. Структурное программирование	
		05.05.2020	2-67	надуковий абонемент	Разработка приложений в Access 97	
		04.04.2020	2-67	надуковий абонемент	Теория автоматизированных банков информации	
Шаблицкий Иван Григорьевич	кафедра зварювання	06.04.2020	2-52	абонемент старших курсів	Теория автоматизированных банков информации	

19 лютого 2021 р.

Страница 1 из 1

Страница: 1 1 Нет фильтра

Рисунок Б.1 – Список боржників

Замовлення							
№ чпг	ПІБ	Підрозділ	№ замовл	Дата	Вид док	Автор	Документ
1126	Сидоренко	кафедра зварювання	7	09.01.2020	монографія	Зиновьев П.О.	Теория автоматизированных банков информац
1654	Іваненко А.Р.	кафедра програмних засобів	2	12.02.2021	монографія	Петросян А.П.	Проектвання систем захисту інформації
			12	06.04.2020	конспект лекцій	Аркадьев В.Л.	С/С++. Структурное программирование. Прак
234	Іваненко	кафедра економіки	3	17.02.2021	монографія	Ампилов а.О.	С++. Базовый курс
			9	26.02.2020	монографія	Машков А.А.	Программирование в Microsoft Office
			10	13.02.2020	авчальний посібни	Аркадьев В.Л.	С/С++. Структурное программирование. Прак
			13				
3164	Полянский	кафедра зварювання	6	01.04.2020	авчальний посібни	Марков А.А.	Самоучитель VBA
352	Корнев	кафедра захисту інформації					

19 лютого 2021 р.

Страница 1 из 2

Рисунок Б.2 – Замовлення

ДОДАТОК В
Слайди презентації

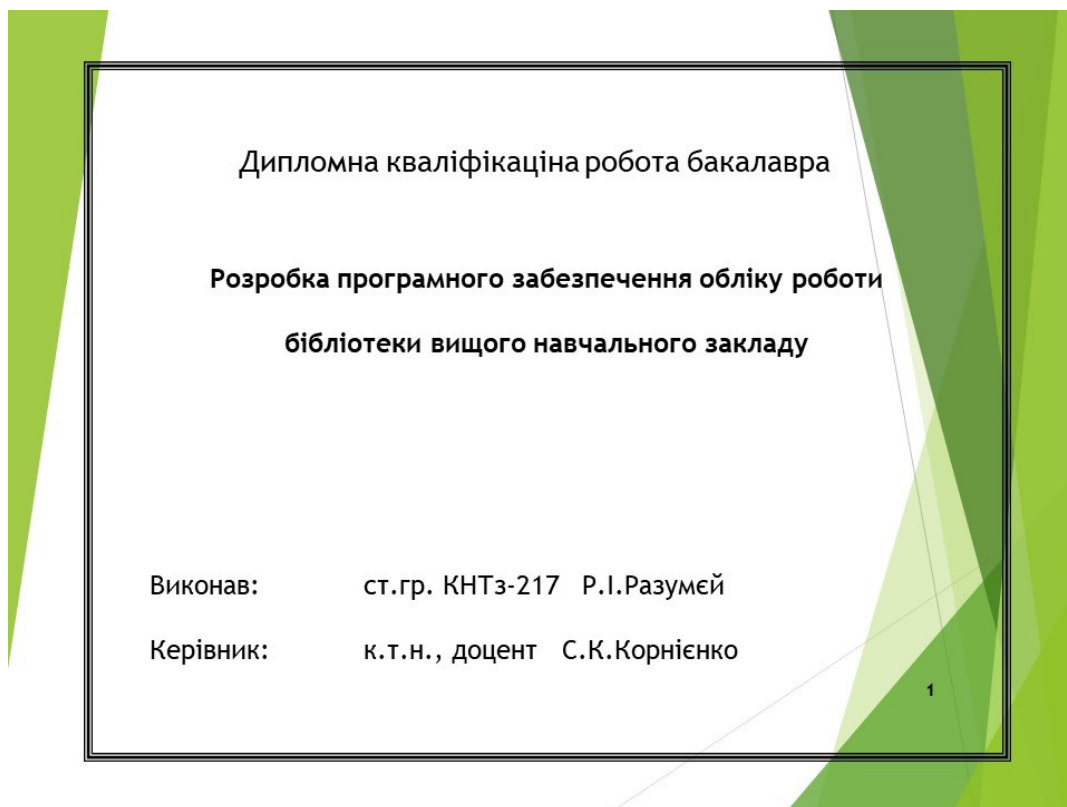


Рисунок В.1 – Слайд №1



Рисунок В.2 – Слайд №2

Завдання, що вирішувалися при виконанні дипломної роботи

- ▶ аналіз предметної області;
- ▶ розробка концептуальної моделі предметної області;
- ▶ вибір інструментарію для розробки системи;
- ▶ логічне проектування системи;
- ▶ розробка інтерфейсу користувача системи;
- ▶ наповнення системи реальними даними та її налагодження.

3

- 3





Рисунок В.5 – Слайд №5

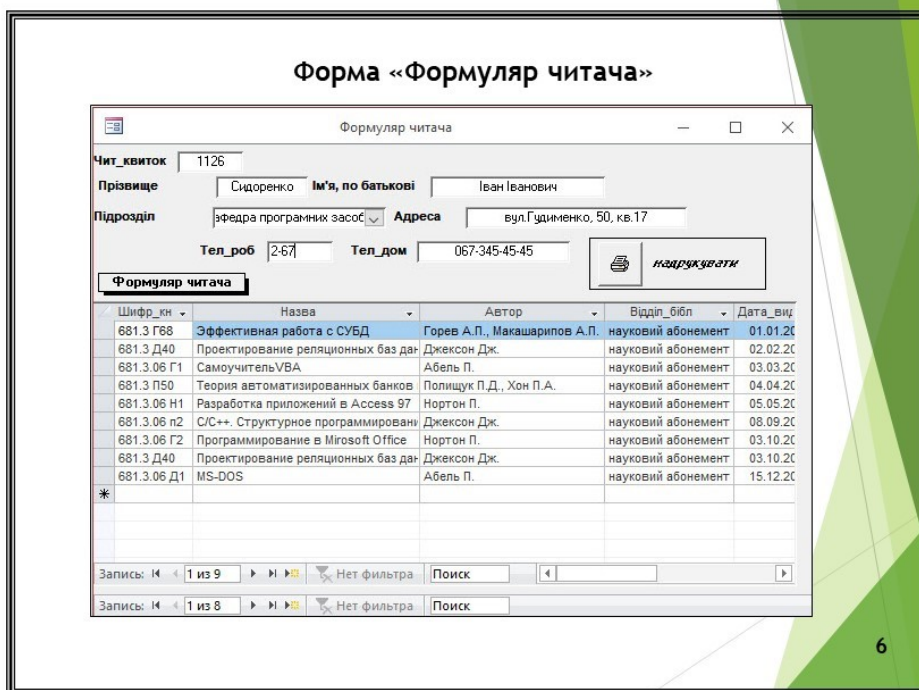


Рисунок В.6 – Слайд №6

Форма «Книги»

Книги

Шифр_кн: 681.3 Г68 Автор: Горев А.П., Макашарипов А.П.

Вид_док: навчальний посібник Том: Стор: 704

Назва: Эффективная работа с СУБД Рік: 2018

Видавництво: Питер Пресс Місто: С.-Петербург Ціна: 120,00 грн.

Анотація

Розглядаються основи побудови БД і основні засоби підвищення продуктивності інформаційних систем

Місце зберігання

Відділ	Кіл
науковий абонемент	5
читальний зал	5
*	0

Ключові слова: БД, СКБД

Пошук

Записи: 14 1 из 11 Нет фильтра Поиск

7

Рисунок В.7 – Слайд №7

Форма «Замовлення літератури»

Замовлення літератури

№ замовлення: 2 Дата: 12.02.2021

Ном_чит: 1654 ПІБ: Іваненко А.Р.

Підрозділ: кафедра програмних засобів

Вид документу: монографія Автор: Петросян А.П.

Назва: Проектування систем захисту інформації

Виконано ☐

Вилучення виконаних замовлень

Записи: 14 1 из 10 Нет фильтра Поиск

8

Рисунок В.8 – Слайд №8

Форма «Пошук літератури»

ПОШУК

За автором

За ключовими словами

За часткою назви

9

Рисунок В.9 – Слайд №9

Форма «Звіти»

ЗВІТИ

СПИСОК БОРЖНИКІВ

ЗАМОВЛЕННЯ

КІЛЬКІСТЬ ВИДАНИХ КНИГ ЗА ДЕНЬ

КІЛЬКІСТЬ ВИДАНИХ КНИГ ЗА ПЕРІОД

ДІАГРАМА ОБСЛУГОВУВАННЯ ЗА ПЕРІОД

КІЛЬКІСТЬ ЧИТАЧІВ ЗА ПЕРІОД

10

Рисунок В.10 – Слайд №10

Висновки:

В процесі виконання дипломної роботи були виконані такі завдання:

- аналіз предметної області;
- розробка концептуальної моделі предметної області;
- вибір інструментарію для розробки системи;
- логічне проектування системи;
- розробка інтерфейсу користувача системи;
- наповнення системи реальними даними та її налагодження.

Практичним результатом проведеної роботи є створення автоматизованої інформаційно-пошукової системи, призначеної для використання в багатопрофільних бібліотеках навчальних закладів.

Розроблено системою може користуватися будь-який працівник, який має мінімальні навички для роботи з комп'ютером.

11

Рисунок В.11 – Слайд №11