

Київський національний торговельно-економічний університет

Кафедра інженерії програмного забезпечення та кібербезпеки

ВИПУСКНА КВАЛІФІКАЦІЙНА РОБОТА

на тему:

Розробка програмного продукту «Книжковий органайзер»

Студента 4 курсу, 6 групи,
спеціальності 121 «Інженерія
програмного забезпечення»

підпис студента

Мороза Ярослава
Романовича

Науковий керівник
кандидат педагогічних наук,
ст. викладач

підпис керівника

Жирова Тетяна
Олександрівна

Гарант освітньої програми
кандидат технічних наук,
доцент

підпис керівника

Цензура Микола
Олександрович

КИЇВ — 2020

Київський національний торговельно-економічний університет

Факультет інформаційних технологій

Кафедра інженерії програмного забезпечення та кібербезпеки

Освітній ступінь бакалавр

Спеціальність 121 «Інженерія програмного забезпечення»

Спеціалізація / освітня програма 121 «Інженерія програмного забезпечення»

Затверджую

Зав. кафедри інженерії
програмного забезпечення
та кібербезпеки
Криворучко О. В.
"7" листопада 2019 р.

Завдання на випускню кваліфікаційну роботу студентів

Морозові Ярославу Романовичу

1. Тема випускної кваліфікаційної роботи: Розробка програмного продукту «Книжковий органайзер».

Затверджена наказом ректора від «02» грудня 2019 р. № 4112

2. Строк здачі студентом закінченої роботи _____

3. Цільова установка та вихідні дані до роботи:

Мета роботи: аналіз наявних електронних органайзерів книжок українських та світових фахівців, а також вивчення й аналіз питань розробки програм для створення продукту «Книжковий органайзер».

Об'єкт дослідження: облік книг домашньої бібліотеки.

Предмет дослідження: розробка програмного продукту «Книжковий органайзер».

4. Консультанти роботи із зазначенням розділів, які консультують:

Розділ	Консультант (прізвище, ініціали)	Підпис, дата	
		Завдання видав	Завдання прийняв

5. Зміст випускної кваліфікаційної роботи (перелік питань за кожним розділом)

ВСТУП

РОЗДІЛ 1. ТЕХНІЧНЕ ЗАВДАННЯ ТА МАРКЕТИНГОВЕ ДОСЛІДЖЕННЯ ПРОГРАМИ «КНИЖКОВИЙ ОРГАНАЙЗЕР»

1.1. Загальні відомості

1.1.1. Назва системи

1.1.2. Терміни роботи

1.1.3. Порядок оформлення і пред'явлення результатів робіт

1.1.4. Потенційні користувачі системи

1.2. Мета, призначення та передумови створення «Книжкового органайзера»

1.2.1. Призначення системи

1.2.3. Передумови для проведення робіт

1.3. Вимоги до системи

1.3.1. Вимоги до системи в цілому

1.3.2. Функціональні вимоги

1.3.3. Вимоги до надійності

1.3.4. Вимоги до надійності технічних засобів і програмного забезпечення

1.4. Вимоги до функцій (задач), що виконуються системою

1.4.1. Модуль «База даних»

1.4.4. Вимоги до програмного забезпечення

1.4.5. Вимоги до технічного забезпечення

1.4.6. Вимоги до організаційного забезпечення

1.4.7. Вимоги до лінгвістичного забезпечення

1.4.8. Вимоги до дизайну

1.5. Навігація по програмі «Книжковий органайзер»

1.6. Огляд ІТ-платформ для створення продукту «Книжковий органайзер»

1.6.1. Діяльність домашньої бібліотеки

1.6.2. Огляд програмних продуктів у бібліотеках

1.6.3. Вибір програм для створення продукту «Книжковий органайзер»

1.6.4. Вибір платформи для створення продукту «Книжковий органайзер»

1.6.5. Вибір мови програмування

1.7. Висновки до розділу 1

РОЗДІЛ 2. ПОБУДОВА ВАЙРФРЕЙМІВ ТА ТАБЛИЦЬ БАЗИ ДАНИХ ПРОГРАМИ «КНИЖКОВИЙ ОРГАНАЙЗЕР»

2.1. Створення вайрфреймів (wireframes)

2.1.1. Вибір платформи для створення вайрфреймів «Органайзера»

2.1.2. Створення першої візуалізації «Книжкового органайзера»

2.2. Побудова таблиць бази даних «Органайзера».

2.2.1. Концептуальна модель «Органайзера»

2.2.2. Логічна модель бази даних програми «Органайзер»

2.2.3. Фізична модель даних «Книжкового органайзера»

2.2.4. Внесення даних у MS SQL

2.3. Висновки до розділу 2

РОЗДІЛ 3. РЕАЛІЗАЦІЯ ПРОГРАМИ «КНИЖКОВИЙ ОРГАНАЙЗЕР»

3.1. Створення інтерфейсу «Органайзера»

3.2. Опис елементів керування меню «Органайзера»

3.3. Підключення баз даних до програми «Книжковий органайзер»

3.4. База даних як невід'ємна частина «Книжкового органайзера»

3.5. Висновки до розділу 3

ВИСНОВКИ ТА ПРОПОЗИЦІЇ

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

ДОДАТКИ

6. Календарний план виконання роботи

№ пор.	Назва етапів випускної кваліфікаційної роботи	Строк виконання етапів роботи	
		за планом	фактично
1	2	3	4
1.	<i>Вибір теми випускної кваліфікаційної роботи</i>		
2.	<i>Вступ та перелік літературних джерел</i>		
3.	<i>Розділ 1. Технічне завдання та маркетингове дослідження програми «Книжковий органайзер»</i>		
4.	<i>Розділ 2. Побудова вайрфреймів та таблиць бази даних програми «Книжковий органайзер»</i>		
5.	<i>Розділ 3. Реалізація програми «Книжковий органайзер»</i>		
6.	<i>Висновки</i>		
7.	<i>Здача випускної кваліфікаційної роботи на кафедру (перша перевірка)</i>		
8.	<i>Підготовка автореферату та презентації доповіді</i>		
9.	<i>Попередній захист випускної кваліфікаційної роботи</i>		
10.	<i>Зовнішнє рецензування випускної кваліфікаційної роботи</i>		
11.	<i>Здача прожитої випускної кваліфікаційної роботи на кафедру</i>		
12.	<i>Публічний захист випускної кваліфікаційної роботи</i>		

7. Дата видачі завдання «__» _____ 20__ р.

8. Науковий керівник випускної кваліфікаційної роботи

Жирова Т. О.

9. Гарант освітньої програми

Цензура М. О.

10. Завдання прийняв до виконання студент

Мороз Я. Р.

This image shows a single sheet of white paper with horizontal ruling lines. The lines are evenly spaced and run across the width of the page. There is no text or other markings on the paper.

(nidnuc, dama)

Жирова Т. О.

(ПІБ, підпис, дата)

Випускна кваліфікаційна робота студента Мороза Я. Р. може бути допущена до захисту екзаменаційній комісії.

Цензура М. О.

Криворучко О. В.

20 p.

АНОТАЦІЯ

Дипломна робота на тему «Розробка програмного продукту «Книжковий органайзер» містить 62 сторінки, 29 рисунків та 1 додаток. Перелік посилань налічує 29 найменувань.

Мета дослідження: аналіз наявних електронних органайзерів книжок та вивчення й аналіз питань розробки програм для створення продукту «Книжковий органайзер».

Об'єкт дослідження: облік книг домашньої бібліотеки.

Предметом дослідження є розробка програмного продукту «Книжковий органайзер».

У першому розділі складено технічне завдання, де визначено мету, призначення та передумови створення «Книжкового органайзера»; проведено маркетингове дослідження, де проаналізовано цільову аудиторію та вже існуючі програми для обліку й організації книг; створено вимоги до розробки програми та обраної платформи й засоби для її реалізації.

У другому розділі проаналізовано платформи для розробки першої візуалізації програми за допомогою вайрфреймів та надано зразки; описано процес роботи з базами даних: розробка концептуальної, логічної, фізичної моделей бази даних і внесення даних у програму MS SQL.

У третьому розділі описано реалізацію програми «Книжковий органайзер». Тут подано інформацію про інтерфейс програми, пояснено, як працювати з «Органайзером» та як підключати базу даних до застосунку.

ANNOTATION

The thesis on the topic "Development of a software product 'Book Organizer'" contains 62 pages, 29 drawings, and 1 appendix. The list of links includes 29 items.

The purpose of the study is to compare existing electronic book organizers, as well as define and analyze the problems of developing the program to create a product "Book Organizer".

Research subject: home library books accounting.

The study's subject is the development of a software product "Book Organizer".

In the first section, there are terms of reference with the aim, purpose, and prerequisites for creation "Book Organizer". This section has marketing research with the described target audience and examined existing book organization programs. It also has the requirements to the software development programs and the description of chosen platforms for their services.

The second section includes the analysis of the platforms for visualization of the program and provides samples of wireframes for this visualization. This section has a description of how to work with database: a conceptual, logical, physical model creation of the database and data transmission into MS SQL.

The third section describes the implementation of the program "Book Organizer". Information about the program interface, usage instructions, tips on how to attach the database to the "Organizer" provided here.

ЗМІСТ

ВСТУП.....	5
РОЗДІЛ 1. ТЕХНІЧНЕ ЗАВДАННЯ ТА МАРКЕТИНГОВЕ ДОСЛІДЖЕННЯ ПРОГРАМИ «КНИЖКОВИЙ ОРГАНАЙЗЕР».....	7
1.1. Загальні відомості	7
1.1.1. Назва системи	7
1.1.2. Терміни роботи	7
1.1.3. Порядок оформлення і пред’явлення результатів робіт.....	7
1.1.4. Потенційні користувачі системи.....	8
1.2. Мета, призначення та передумови створення «Книжкового органайзера».....	9
1.2.1. Призначення системи.....	9
1.2.3. Передумови для проведення робіт	10
1.3. Вимоги до системи.....	11
1.3.1. Вимоги до системи в цілому	11
1.3.2. Функціональні вимоги	12
1.3.3. Вимоги до надійності	12
1.3.4. Вимоги до надійності технічних засобів і програмного забезпечення	12
1.4. Вимоги до функцій (задач), що виконуються системою	12
1.4.1. Модуль «База даних».....	12
1.4.4. Вимоги до програмного забезпечення.....	13
1.4.5. Вимоги до технічного забезпечення	14
1.4.6. Вимоги до організаційного забезпечення	14

					КНТЕУ 121 06-15.БР			
Зм.	Аркуш	№ докум	Підпис	Дата	Розробка програмного продукту «Книжковий органайзер»	Стадія	Аркуш	Аркушів
Зав. кафедри	Криворучко О.В.					Зміст	2	58
Керівник	Жирова Т. О.					Факультет інформаційних технологій, 4 курс, 6 група		
Гарант	Цензура М. О.							
Розроб.	Мороз Я. Р.							
					Зміст			

1.4.7. Вимоги до лінгвістичного забезпечення.....	15
1.4.8. Вимоги до дизайну.....	15
1.5. Навігація по програмі «Книжковий органайзер».....	15
1.6. Огляд ІТ-платформ для створення продукту «Книжковий органайзер».....	16
1.6.1. Діяльність домашньої бібліотеки.....	16
1.6.2. Огляд програмних продуктів у бібліотеках.....	17
1.6.3. Вибір програм для створення продукту «Книжковий органайзер»	19
1.6.4. Вибір платформи для створення продукту «Книжковий органайзер».....	22
1.6.5. Вибір мови програмування	23
1.7. Висновки до розділу 1	24
РОЗДІЛ 2. ПОБУДОВА ВАЙРФРЕЙМІВ ТА ТАБЛИЦЬ БАЗИ ДАНИХ ПРОГРАМИ «КНИЖКОВИЙ ОРГАНАЙЗЕР».....	26
2.1. Створення вайрфреймів (wireframes).	26
2.1.1. Вибір платформи для створення вайрфреймів «Органайзера» ..	26
2.1.2. Створення першої візуалізації «Книжкового органайзера».....	28
2.2. Побудова таблиць бази даних «Органайзера».....	31
2.2.1. Концептуальна модель «Органайзера»	31
2.2.2. Логічна модель бази даних програми «Органайзер».....	32
2.2.3. Фізична модель даних «Книжкового органайзера»	33
2.2.4. Внесення даних у MS SQL	33
2.3. Висновки до розділу 2	38
РОЗДІЛ 3. РЕАЛІЗАЦІЯ ПРОГРАМИ «КНИЖКОВИЙ ОРГАНАЙЗЕР»... 40	40
3.1. Створення інтерфейсу «Органайзера»	40
3.2. Опис елементів керування меню «Органайзера»	42

					КНТЕУ 121 06-15.БР	Аркуш
						3
Зм.	Аркуш	№ докум	Підпис	Дата		

3.3. Підключення баз даних до програми «Книжковий органайзер».....	46
3.4. База даних як невід’ємна частина «Книжкового органайзера»	47
3.5. Висновки до розділу 3	51
ВИСНОВКИ ТА ПРОПОЗИЦІЇ	53
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ	56
ДОДАТКИ.....	59

					КНТЕУ 121 06-15.БР	Аркуш
						4
Зм.	Аркуш	№ докум	Підпис	Дата		

ВСТУП

У час технологій та всебічного розвитку навіть людини, яка є найрозумнішою істотою, важко впоратися з величезним об'ємом даних. Для того щоб впорядкувати інформацію, зберігати її, ділитися нею, оновлювати, потрібно затрачати багато енергії, що доволі часто викликає у пересічних людей прокрастинацію — бажання відкладати все на потім. Щоб зменшити навантаження та ефективно використовувати час, процес обробки інформації було повністю комп'ютеризовано.

Актуальність. У сучасному світі більшість людей отримує інформацію з інтернету або за допомогою різних програм. За дослідженням Грея Вільямса від 01 лютого 2020 року, більше 1 мільярда людей активно використовують соцмережу Facebook, а 59 % людей вважають за краще читати новини в Інтернеті, а не слухати або дивитися їх [21]. Ці дані підтверджують важливість повсякденного використання електронних засобів. Важко уявити особу, яка не має акаунту хоча би в одній із соціальних мереж, а те, що є люди, які жодного разу не відкривали сторінку браузеру й не здійснювали запиту в пошуку Google, який використовують, за згаданим вище дослідженням, 90,61 % користувачів, взагалі викликає недовіру (хоча подібні випадки й справді є).

Зараз настав такий період, коли любителі книжок усе частіше діляться прочитаними творами та поширюють у своїх соцмережах цитати, які найбільше припали до душі. Якщо ж читач однієї з таких акаунт-сторінок вподобав ідеї книги, доволі часто здійснюється уже популярне явище, як «буккросинг» —

					КНТЕУ 121 06-15.БР			
Зм.	Аркуш	№ докум	Підпис	Дата	Розробка програмного продукту «Книжковий організер»	Стадія	Аркуш	Аркушів
Зав. кафедри	Криворучко О.В.					В	5	58
Керівник	Жирова Т. О.					Факультет інформаційних технологій, 4 курс, 6 група		
Гарант	Цензура М. О.							
Розроб.	Мороз Я. Р.							
					Вступ			

обмін книгами. Щоб не загубити в переліку публікацій цитати, а крім цього, завжди мати інформацію, кому ви позичили свою книгу, запропоновано програму, яка має назву «Книжковий органайзер».

Мета дослідження випускної роботи: аналіз наявних електронних органайзерів книжок українських та світових фахівців, а також вивчення й аналіз питань розробки програм для створення продукту «Книжковий органайзер».

Об'єкт дослідження: облік книг домашньої бібліотеки.

Предметом дослідження є розробка програмного продукту «Книжковий органайзер».

Завдання дослідження:

- створити платформу для обліку книг домашньої бібліотеки;
- використати нові платформи для розробки програмного забезпечення;
- провести аналіз аудиторії користувачів програми;
- провести підготовку програми до її реалізації;
- створити бази даних домашньої бібліотеки.

Методи та технології розробки проводилися за методологією екстремального програмування на основі водоспадної моделі (waterfall). Програмування виконувалося за допомогою таких платформ, як MySQL, Microsoft Visual Studio 19, мова програмування — C#.

Результат роботи — розроблена легка в користуванні, конкурентоспроможна десктопна програма, для зберігання інформації про книги домашньої бібліотеки та їхнього обліку.

					КНТЕУ 121 06-15.БР	Аркуш
Зм.	Аркуш	№ докум	Підпис	Дата		6

РОЗДІЛ 1.

ТЕХНІЧНЕ ЗАВДАННЯ ТА МАРКЕТИНГОВЕ ДОСЛІДЖЕННЯ ПРОГРАМИ «КНИЖКОВИЙ ОРГАНАЙЗЕР»

1.1. Загальні відомості

1.1.1. Назва системи

Повне найменування системи: Книжковий органайзер.

Коротке найменування системи: Органайзер.

1.1.2. Терміни роботи

Початок роботи: жовтень 2019 р.

Закінчення роботи: квітень 2020 р.

1.1.3. Порядок оформлення і пред'явлення результатів робіт

Програмне забезпечення створюється у вигляді функціонуючого комплексу на базі засобів обчислювальної техніки та електротехнічного устаткування у визначені строки. Роботи зі створення програми «Книжковий органайзер» виконуються поетапно в наступній послідовності: етап, зміст роботи і результат:

- проведення маркетингового дослідження;
- створення технічного завдання;
- вибір платформи для роботи над проектом;
- проектування інтерфейсів та розробка графічного дизайну всіх модулів;
- створення прототипу (погодження функціоналу та дизайну програми);

					КНТЕУ 121 06-15.БР			
Зм.	Аркуш	№ докум	Підпис	Дата	Розробка програмного продукту «Книжковий органайзер»	Стадія	Аркуш	Аркушів
Зав. кафедри	Криворучко О.В.					P1	7	58
Керівник	Жирова Т. О.					Факультет інформаційних технологій, 4 курс, 6 група		
Гарант	Цензура М. О.							
Розроб.	Мороз Я. Р.							
					Технічне завдання та маркетингове дослідження програми «Книжковий органайзер»			

- розробка програмної частини модулів системи, інтеграція шаблонів, налаштування інтерфейсів;
- проєктування та створення бази даних системи Схема даних, таблиці для збереження даних та зв'язки між ними;
- тестування системи на контрольних прикладах.

Після реалізації цих пунктів, результат проведених робіт пред'являється за допомогою презентації.

1.1.4. Потенційні користувачі системи

Цільовою аудиторією проєкту є організовані люди, спрагли до нових знань, ті, хто хоче контролювати обіг своїх книг, або просто ті, хто має книжкову бібліотеку вдома.

Потенційних користувачів можна класифікувати за наступними критеріями:

1. Школярі – для контролю прочитаних книг та їх планування; за видом діяльності: школярі, студенти, працевлаштовані люди, безробітні та пенсіонери.
2. Активісти – для збереження найвдаліших цитат, які мотивуватимуть їх і надалі вдосконалювати як себе, так і суспільство.
3. Спікери та тренери – для швидкого доступу до даних про книги, особливо, коли потрібно порекомендувати їх своїй аудиторії.
4. Різні працівники – для планування прочитання книг, щоб підвищити свою кваліфікацію.
5. Домогосподарі – для обліку книг.

					КНТЕУ 121 06-15.БР	Аркуш
						8
Зм.	Аркуш	№ докум	Підпис	Дата		

Найбільше програма підійде користувачам, віком від 18 до 40 років, оскільки це той вік, коли люди використовують найбільше інформації і вони найлояльніші до нових продуктів ринку. Але «Органайзер» стане в нагоді кожному, незалежно від віку. Стать користувачів також не має значення – програма підходить усім.

1.2. Мета, призначення та передумови створення «Книжкового органайзера»

1.2.1. Призначення системи

Призначенням системи – навчити читачів колекціонувати свої книги, ставати більш відкритими до сучасних течій, таких як «буккросинг», при цьому не боятися загубити свої книжки.

Ціль програми:

- показати користувачам кількість прочитаних книг, цим самим змотивувавши їх до подальшого саморозвитку;
- дати можливість людям виписувати мудрі / цікаві / оригінальні цитати та тримати їх під рукою;
- надати можливість до швидкого доступу до книг: за допомогою пошуку перевірити наявність або вибрати матеріал для читання за датою написання / автором / твором / жанром / країною.

Програм також призначена, щоб власники, якщо вони позичили комусь книгу мали змогу бачити такі дані:

- кому була позичена книга;
- дата позики;
- назва книги;

					<i>КНТЕУ 121 06-15.БР</i>	Аркуш
						9
Зм.	Аркуш	№ докум	Підпис	Дата		

- умови позики;
- контакти людини, яка отримала книгу.

1.2.2. Мета створення системи

Розробити платформу для організації книг в особистій бібліотеці, їх обліку та збереження даних про найоригінальніші книги.

1.2.3. Передумови для проведення робіт

Сучасне суспільство живе в шаленому ритмі. Кожен бореться за своє місце на новій посаді або мотивується «бути кращим». Через те, що світ став більш відкритим до знань, відчувається тенденція потреби володіння інформацією про все на світі. Як зазначає Грей Вільямс, «чим більш освічена людина, тим більше вона схильна використовувати соціальні медіа» [21]. Хоча немає очевидних причин, чому так відбувається, але це може бути пов'язано з тим, що освіченіші люди працюють з комп'ютерами, маркетингом або мережею. Той, хто має диплом коледжу використовує соціальні мережі в середньому на 10% більше, ніж той, хто має нижчу освіту, та на 20% більше, ніж той, хто має атестат про закінчення середньої школи або подібних закладів (рис. 1.1).

Оскільки значна частина роботи переходить в онлайн, а в соціальних медіа люди діляться своїми думками, наприклад, про прочитані книги, з'явилася ідея, що користувачам мережі Інтернет потрібно платформа для впорядкування інформації.

Роздумуючи, що може зацікавити аудиторію міленіалів та зумерів, вирішено розробити програму для організації книг, що доторкається відразу до двох сфер життя: знання та відпочинку.

					<i>КНТЕУ 121 06-15.БР</i>	Аркуш
						10
Зм.	Аркуш	№ докум	Підпис	Дата		

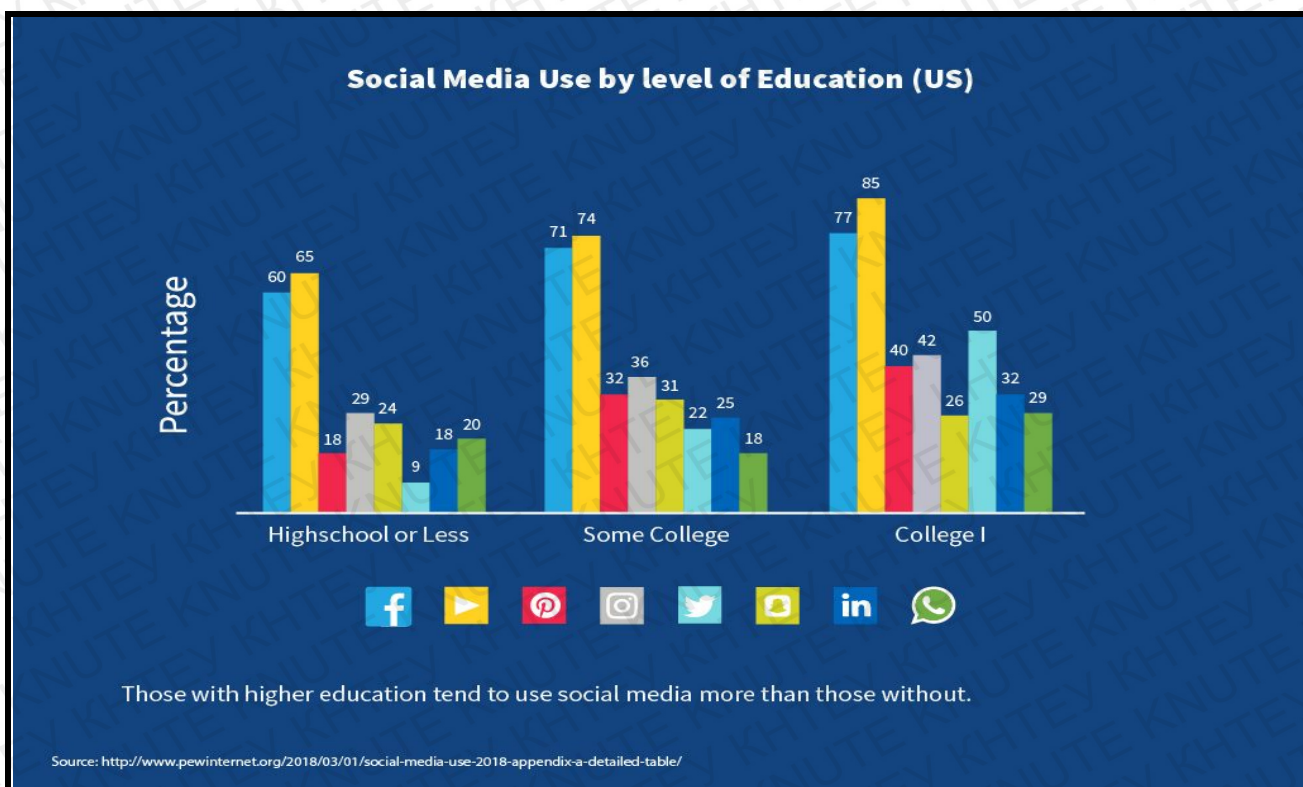


Рис. 1.1. Тенденція використання соціальних медіа різними категоріями населення

1.3. Вимоги до системи

1.3.1. Вимоги до системи в цілому

Програмний комплекс повинен забезпечувати зберігання та обробку даних з використанням бази даних, доступ до якої здійснюється в межах локальної обчислювальної мережі.

У склад програмного комплексу повинні входити програмні модулі:

- база даних;
- пошук і контроль книг;
- ведення своєї бібліотеки.

					КНТЕУ 121 06-15.БР	Аркуш
Зм.	Аркуш	№ докум	Підпис	Дата		11

1.3.2. Функціональні вимоги

Програмний продукт повинен володіти наступними функціональними характеристиками:

- наявність бази даних;
- збереження даних;
- пошук книги за різними критеріями;
- зручність інтерфейсу.

1.3.3. Вимоги до надійності

Надійне (стійке) функціонування програмного комплексу має бути забезпечене шляхом контролю коректності та повноти вхідних даних.

1.3.4. Вимоги до надійності технічних засобів і програмного забезпечення

Захист інформації від несанкціонованого доступу забезпечуватися користувачем, оскільки це офлайнова програма. Додаткового захисту не потрібно. «Органайзер» розрахований на те, щоб ділитися літературою.

1.4. Вимоги до функцій (задач), що виконуються системою

1.4.1. Модуль «База даних»

Запропонована програма вимагає до себе прикріпленої бази даних, щоб людина, яка працюватиме за «Книжковим органайзером» змогла додавати свої книги до списку, їх редагувати, якщо дані введені неправильно. Без жодного сумніву необхідною функцією буде можливість видаляти дані з бази.

					КНТЕУ 121 06-15.БР	Аркуш
						12
Зм.	Аркуш	№ докум	Підпис	Дата		

У «Книжковому органайзері» буде сім таблиць. Кожна з них матиме дані, де одна з колонок – ключ. Головна таблиця об'єднуватиме дані з усіх таблиць.

1.4.2. Модуль «Пошук і контроль книг»

Одним із основних завдань «Органайзера» – спростити пошук книг у своїй бібліотеці, зокрема пересвідчитися, що вони зараз лежать на своєму місці. У програмі додано кнопки різної функціональності, які допоможуть швидко і якісно знайти книгу за жанром, автором, назвою і навіть роками створення шедевр. Якщо користувач ще не обрав твору для читання, він чи вона має змогу вибрати випадкові варіанти характеристик й обрати найкращий варіант. Не виключається можливість відразу знайти книгу за заздалегідь визначеними характеристиками.

1.4.3. Модуль «Ведення своєї бібліотеки»

Одним із найважливіших пунктів є створення можливості ведення своєї бібліотеки. Потрібно зробити так, щоб користувачі після прочитання книги додавали їх до «Прочитаних», створювали самі короткий опис книги, щоб рекомендувати іншим. Оцінювання книги за шкалою п'яти зірок і створення рейтингу було б цікавою перспективою на майбутнє.

1.4.4. Вимоги до програмного забезпечення

Програмний комплекс повинен бути побудований на клієнт-серверній архітектурі з використанням технологій, що не вимагають додаткового ліцензування. Серверна частина програмного комплексу має базуватися на системі керування базами даних. Програма повинна працювати в режимі оффлайн. При розробці дозволяється застосовувати фреймворки. Клієнтська частина повинна бути побудована на стандартних технологіях, що не вимагають

					КНТЕУ 121 06-15.БР	Аркуш
						13
Зм.	Аркуш	№ докум	Підпис	Дата		

встановлення додаткових компонентів на комп'ютер (не рекомендується використовувати Flash або Java-аплети). Інтерфейс повинен бути комфортний. Дані, введені клієнтом повинні бути захищені від зовнішнього втручання та сторонньої модифікації. Також необхідно програмним шляхом забезпечити одностороннє архівування книг у резервну базу даних, що можна вивантажити у вигляді файлу.

1.4.5. Вимоги до технічного забезпечення

Програмний комплекс розрахований на функціонування при такому наборі технічних засобів:

- мінімальна апаратна конфігурація сервера;
- ядерний процесор з тактовою частотою від 1.4 ГГц;
- оперативна пам'ять об'ємом не менше 2 Гб;
- відеокарта та монітор з роздільною здатністю не менше 1024x768;
- клавіатура;
- маніпулятор «миша»;
- вільний дисковий простір не менш ніж 500 Мб;

1.4.6. Вимоги до організаційного забезпечення

Програма повинна бути компактна і не займати багато місця. Інтерфейс програмного комплексу – зрозумілим та зручним, не перевантаженим графічними елементами. Навігаційні елементи та компоненти інтерактивної взаємодії повинні бути виконані у зручній для користувача формі. Інтерфейс повинен мати можливість адаптації до людей із вадами зору або слуху для полегшення їх роботи із програмним комплексом.

					КНТЕУ 121 06-15.БР	Аркуш
Зм.	Аркуш	№ докум	Підпис	Дата		14

1.4.7. Вимоги до лінгвістичного забезпечення

Користувацький інтерфейс системи повинен бути виконаний українською мовою при першому запуску. У перспективі проєктування системи повинна – можливість перекладу інтерфейсу англійською. Додавляти книги можна й англійською мовою (або мовою оригіналу).

Уся документація, що розроблюється в рамках створення системи, повинна бути надана українською мовою. При розробці програмного забезпечення системи повинні використовуватися ефективні мови високого рівня. Вибір мов програмування виконується на стадії розробки програмної частини системи.

1.4.8. Вимоги до дизайну

Програма повинна бути виконана в стриманих кольорах, але таких, що приємні візуально. Найкращим варіантом буде комбінація чорно-зелених кольорів, оскільки вони чудово поєднуються та додають відчуття затишку. Жовтий та білий колір розбавить тьмяність.

Кнопки потрібно зробити з додаванням функції hover. Для деяких кнопок колір буде змінюватися, для інших змінюватиметься тільки колір тексту, а для останніх використовуватиметься заповнення іконок.

1.5. Навігація по програмі «Книжковий органайзер»

Створювана програма матиме два функціональні меню верхня (головне) та бокове.

Верхнє меню незмінне. На ньому розташовуються чотири кнопки: «Мої книги», «Прочитане», «Позичене» та «Пошук».

Бокове меню працює тільки при відкриванні кнопки «Мої книги», адже всі інші з головного меню мають свій функціонал.

					КНТЕУ 121 06-15.БР	Аркуш
						15
Зм.	Аркуш	№ докум	Підпис	Дата		

На боковій панелі розташуються наступні кнопки: Список книг, Автор, Країна, Жанр, Рік випуску, Додати до бази та Додати. Останні дві кнопки використовуватимуться для роботи з базами даних.

Облік книг можна перевірити, натиснувши на кнопку «Позичене». На даній панелі зліва є кнопки «Позичене комусь» і «Позичене мені».

Функціональними також є кнопки, розташовані у верхньому правому куті, що закривають, згортають програма та відкривають довідку.

1.6. Огляд ІТ-платформ для створення продукту «Книжковий органайзер»

1.6.1. Діяльність домашньої бібліотеки

Домашня бібліотека володіє широким асортиментом книг для своїх користувачів. Тут присутні жанри літератури на будь-який смак: науковий, художній, фантастичний, дитяча література, військова тощо. Будь-хто знайде тут книги різних епох та авторів усіх країн світу. Власники книг діляться творами митців для того, щоб популяризувати літературу та мистецтво серед людей, не залежно від їхнього віку.

Щоб запобігти ситуаціям, коли під час неформального буккросингу власник утрачає свої книги, а також, щоб упевнитися, що твори мистецтва повернуться додому запропоновано програму «Книжковий органайзер». У цій програмі потенційний читач заповнює форму з такими даними: прізвище, ім'я, по батькові, паспортні дані, електронна адреса та номер телефону.

Після того як було обрано книгу для обміну, відбувається обговорення дати повернення книги власникам та депозит. У разі не дотримання умов домовленості, депозит не повертається, але клієнт зобов'язаний повернути книгу.

					КНТЕУ 121 06-15.БР	Аркуш
						16
Зм.	Аркуш	№ докум	Підпис	Дата		

1.6.2. Огляд програмних продуктів у бібліотеках

На теренах інтернету присутні тисячі онлайн-бібліотек з мільонами електронних книжок. Найбільше пощастило американцям, в яких є Amazon і Barnes&Noble, де можна знайти будь-яку з існуючих книжок. Не гірша ситуація і в інших сегментах, де електронні бібліотеки працюють уже понад десятиліття.

За останні роки в Україні з'явилося кілька дуже цікавих онлайн-бібліотек, які спеціалізуються на книжках українською мовою, але їх не достатнього.

Є різні програмні забезпечення для автоматизації роботи бібліотеки. При їхньому виборі слід звертати увагу на особливості, які потрібні саме для вашої бібліотеки: розділення фонду підручників та іншої літератури, книжковий фонд, ким проводиться робота: дорослим чи дитиною. Нижче зазначено програми для роботи з бібліотекою.

Система «УФД/Бібліотека» має забезпечити автоматизацію основних виробничих циклів: каталогізацію видань – облік фонду – аналіз стану фонду; реєстрацію читачів. На жаль, дана система не дає можливості реалізувати багато функцій, специфічних для різних бібліотек, хоча має деяку надлишковість в інших функціях.

Система «UniLib» призначена для автоматизації бібліотечних процесів обслуговування читачів, а також комплектування, обробки та обліку літератури. Дана система є зручною і простою в користуванні, а ще має багато функціональних можливостей.

Система «Бібліограф» дозволяє в короткий термін сформувати електронний каталог, організувати пошук та замовлення літератури читачами та розпочати реєстрацію видачі й повернення книг та періодики. Програми пакету мають зручну та детальну довідкову систему, яка допоможе швидко навчитись

					КНТЕУ 121 06-15.БР	Аркуш
						17
Зм.	Аркуш	№ докум	Підпис	Дата		

ефективно їх використовувати. Недоліком системи є те, що даний продукт не підтримує пошук у каталогах інших бібліотек і зведених каталогах [2].

Система «Goodreads» – чудовий ресурс, щоб запланувати прочитати книги, стежити, які книги прочитані та поставити відповідний (суб’єктивний) рейтинг. Тут можна стежити за іншими користувачами й залишати відгуки щодо окремої книги. Але «Goodreads» має такі недоліки:

- це онлайн-платформа, тому потрібно дотримуватися її політики: «Goodreads» може надавати інформацію про ваше ведення акаунт третім сторонам, тому не можна бути впевненим у конфіденційності своєї інформації;
- якщо користуватися веб-сторінкою, блокування куків призведе до зменшення можливостей користування платформи;
- інтерфейс веб-сайту перенасичений, тому важко контролювати потрібну інформацію (рис. 1.2.);
- не має можливості для обліку передачі книг тощо.

					КНТЕУ 121 06-15.БР	Аркуш
						18
Зм.	Аркуш	№ докум	Підпис	Дата		

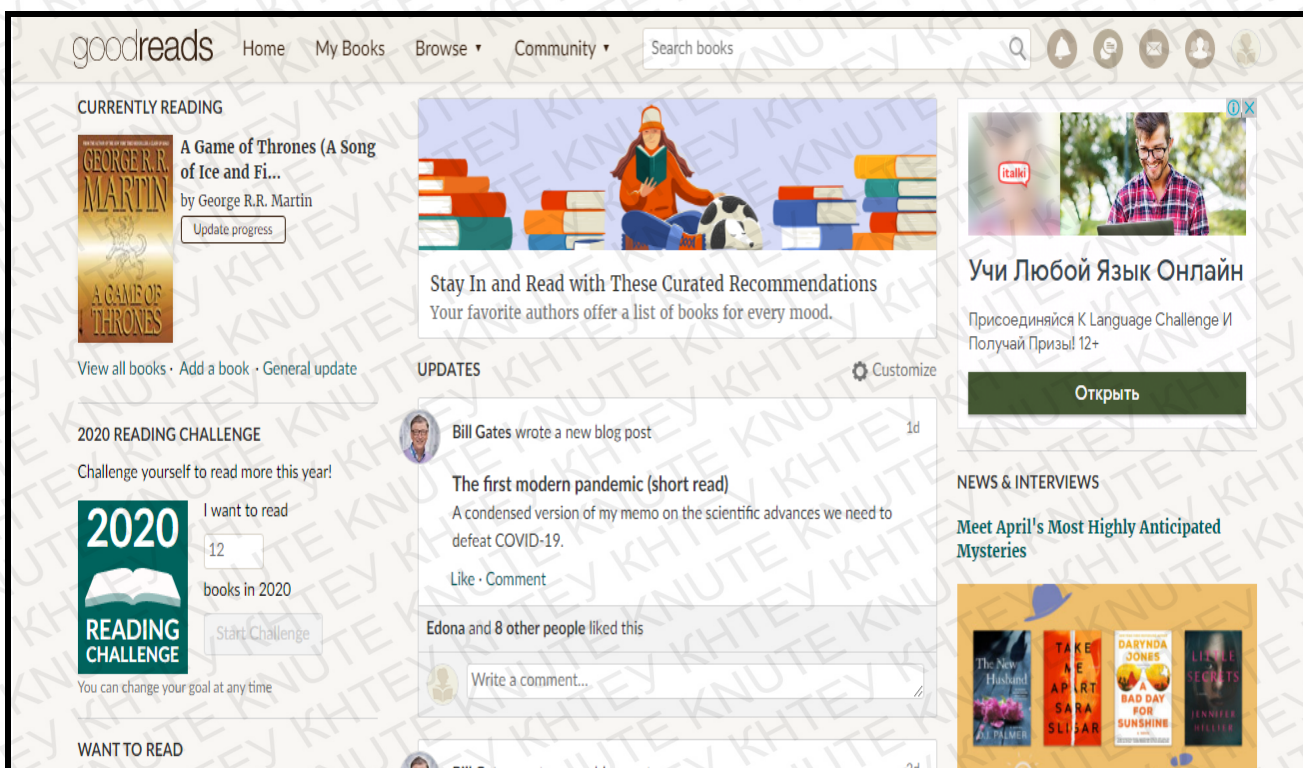


Рис. 1.2. Платформа «Goodreads»

1.6.3. Вибір програм для створення продукту «Книжковий органайзер»

Для того щоб створити базу даних, було порівняно декілька відповідних програм: MySQL, СУБД Oracle, Microsoft Office Access, PostgreSQL, Microsoft SQL Server.

MySQL – вільна система управління базами даних, яка є власністю компанії Oracle Corporation, що отримала її разом з поглиненою Sun Microsystems, яка розробляє і підтримує додатки. Розповсюджується на вибір: під час General Public License або під власною комерційною ліцензією.

Одна з переваг MySQL – чудова підтримка з боку численних мов програмування.

Також до переваг варто віднести такі фактори:

- MySQL є хорошим рішенням для малих і середніх додатків;

					КНТЕУ 121 06-15.БР	Аркуш
						19
Зм.	Аркуш	№ докум	Підпис	Дата		

- вихідні коди сервера компілюються на різних платформах;
- до MySQL входить бібліотека внутрішнього сервера, що дозволяє включати MySQL до самостійних програм;
- забезпечується гнучкість СУБД MySQL: користувачі вибирають, як таблиці типу MyISAM, що підтримують повнотекстовий пошук, так і таблиці InnoDB, що підтримують транзакції на рівні окремих записів [9].

Для некомерційного використання MySQL є безкоштовною. Можливості сервера MySQL:

- необмежена кількість користувачів, що одночасно працюють із базою даних;
- висока швидкість виконання команд;
- кількість рядків у таблицях до 50 млн.;
- простота у встановленні та використанні;
- наявність простої та ефективної системи безпеки.

Суттєвим недоліком MySQL є те, що в системі управління базами даних MySQL виявили збій, який породжує ризик використання платформи для зливу конфіденційної інформації на сервери зломисників.

СУБД **Oracle** – це найпотужніший програмний комплекс, який дозволяє створювати нові додатки, не залежно від їхньої складності. База даних – ядро цього комплексу, що зберігає інформацію, кількість якої за рахунок наданих засобів масштабування практично безмежна. З високою ефективністю працювати з цією інформацією одночасно може будь-яка кількість користувачів (за наявності достатніх ресурсів), без тенденції до зниження продуктивності системи при різкому збільшенні їхньої кількості.

Механізми масштабування в СУБД Oracle останньої версії дозволяють збільшувати швидкість і потужність роботи сервера Oracle та своїх додатків,

					КНТЕУ 121 06-15.БР	Аркуш
						20
Зм.	Аркуш	№ докум	Підпис	Дата		

додаючи все більше нових вузлів кластеру. Це не вимагає зупинки працюючих додатків та не вимагає перепису старих, які були розроблені для звичайної однамашинної архітектури. Крім того, до зупинки програми навіть не призводить вихід з ладу окремих вузлів кластера.

Ще одна перевага СУБД Oracle – постачання практично для всіх існуючих на сьогодні операційних систем. Працюючи під Windows, Sun Solaris, Linux або на іншій операційній системі, з продуктами Oracle не виникатиме жодних проблем у роботі. СУБД Oracle добре працює однаково на довільній платформі.

Microsoft Office Access – реляційна СУБД корпорації Microsoft. Вона має широкий асортимент функцій, включаючи зв'язані запити, зв'язок із базами даних та зовнішніми таблицями.

MS Access – це файл-серверна СУБД. Вона застосовується лише для маленьких додатків. У ній відсутній ряд механізмів, які необхідні в багатокористувацьких базах даних, наприклад, такі, як тригери.

MS Access дозволяє будувати повноцінні клієнт-серверні додатки на СУБД MS SQL Server, з можливістю поєднати з притаманною MS Access простотою інструменти для управління та керування базою даних і, звісно, засоби розробки [20].

PostgreSQL – об'єктно-реляційна система управління базами даних. Вона є альтернативою як комерційним СУБД (Microsoft SQL Server, IBM DB2, Oracle Database та інші), так і СУБД з відкритим кодом (SQLite, Firebird, MySQL).

Порівняно з іншими проєктами з відкритим кодом, такими як Apache, MySQL або FreeBSD, PostgreSQL не контролюється якоюсь однією компанією, її розробка можлива завдяки співпраці великої кількості компаній і людей, які бажають використовувати цю СУБД та впроваджувати в неї найновіші досягнення.

					<i>КНТЕУ 121 06-15.БР</i>	Аркуш
						21
Зм.	Аркуш	№ докум	Підпис	Дата		

Microsoft SQL Server – система управління реляційними базами даних, розроблена корпорацією Microsoft. Основна використовувана мова запитів – Transact-SQL, яку спільно створили Microsoft та Sybase. Transact-SQL – реалізація стандарту ANSI / ISO щодо структурованої мови запитів (SQL) з розширеннями. Вона використовується для роботи з базами даних розміром від персональних баз даних до баз великого масштабу підприємства, а також конкурує з іншими СУБД у цьому сегменті ринку.

Для забезпечення доступу до даних Microsoft SQL Server підтримує Open Database Connectivity (ODBC) – інтерфейс взаємодії додатків з СУБД. SQL Server надає можливість підключення користувачів через веб-сервіси, що використовують протокол SOAP. Це дозволяє клієнтським програмам, не призначеним для Windows, кросплатформно з'єднуватися з SQL Server [9].

Порівнявши всі платформи для роботи з базами даних, було зроблено висновки, що найкращою для «Книжкового органайзера» буде система Microsoft SQL Server. Оскільки однією з переваг MS SQL є те, що вона надає можливість підключення користувачів через веб-сервіси, то це дає можливість розширювати функціонал програми в майбутньому та розробити відповідну веб-сторінку.

Проблематика задачі полягала в тому, щоб розробити програму, якою користувалися б представники будь-якої категорії населення: від школярів до людей старшого віку. Для цього, окрім цікавого інтерфейсу, який би приваблював користувачів та не викликав утому для очей, потрібно було організувати правильно таблиці та інформацію бази даних.

1.6.4. Вибір платформи для створення продукту «Книжковий органайзер»

					<i>КНТЕУ 121 06-15.БР</i>	Аркуш
						22
Зм.	Аркуш	№ докум	Підпис	Дата		

Для створення «Органайзера» було вирішено використати платформу Microsoft Visual Studio 2019.

Microsoft Visual Studio – серія продуктів фірми Майкрософт, що включають інтегроване середовище розробки програмного забезпечення та ряд інших інструментальних засобів. Перевагою Visual Studio є те, що ці продукти дозволяють розробляти як консольні програми, так і програми з графічним інтерфейсом, включаючи підтримку технології Windows Forms, а також веб-застосунки, веб-сайти, веб-служби як у рідному, так і в керованому кодах для всіх платформ, які підтримуються такими програмами, як Windows Phone, Microsoft Windows, .NET Compact Framework, Windows Mobile, Windows CE, .NET Framework, та Microsoft Silverlight.

1.6.5. Вибір мови програмування

Якщо вибирати між такими мовами програмування, як C++, Java, C#, C тощо, то C# є найновішим з широковідомих мов програмування. Творцем мови C# є співробітник компанії Microsoft Андерс Хиджисберг, який раніше був провідним розробником середовища програмування Delphi. Мова C# створювалася як мова компонентного, об'єктно-орієнтованого програмування, і в цьому одна з головних переваг мови, що спрямована на можливість повторного використання створених компонентів. Інші переваги даної мови наступні:

- повністю об'єктно-орієнтована мова, де навіть убудовані в мову типи є об'єктами класів;
- потужна об'єктна мова з можливостями успадкування та універсалізації (створення узагальнених класів);
- спадкоємиця мови C ++, що зберігає кращі риси цієї популярної мови програмування [24].

					КНТЕУ 121 06-15.БР	Аркуш
						23
Зм.	Аркуш	№ докум	Підпис	Дата		

Ключові особливості мови C# є те, що у ній присутня компонента орієнтованість, код зібраний воедино (декларації і реалізації об'єднані разом). Щодо безпеки, то в C# наявна уніфікована система типів. Також у C# використовується єдина бібліотека класів – CLR, а робота з пам'яттю автоматична та мануальна.

Нормальний стиль програмування на C# передбачає написання безпечного коду, а це означає, що не використовуються жодні ніяких покажчики, жодні адресні арифметики, жодне управління розподілом пам'яті. Можливість роботи з покажчиками в дусі C++ обмежена небезпечними блоками. Небезпечний код для C#-програмістів буде швидше винятком, ніж правилом. Це дозволяє менше відволікатися на відстеження коректності роботи програми з пам'яттю, приділяючи більше уваги функціональній частини програми [23].

Проаналізувавши різні мови програмування, було зроблено висновок, що всі мови мають свої переваги та недоліки. Тому обирати ресурс для створення власної програми кожному програмісту потрібно обирати індивідуально. Оцінивши всі «за» і «проти», було прийнятим рішення для програми «Книжковий органайзер» обрати мову програмування C#, оскільки вона найпоширеніша та найновіша з перелічених вище варіантів, а це означає, що вона легша для написання коду та має більше функціоналу, що відсутній в інших мовах.

1.7. Висновки до розділу 1

Провівши аналіз аудиторії для програми, було визначено складено користувача: вісімнадцятирічний – сорокарічний освічений користувач, не залежно від професії та статі, що активно ділиться у соцмережах корисною інформацією з прочитаних книг.

					КНТЕУ 121 06-15.БР	Аркуш
						24
Зм.	Аркуш	№ докум	Підпис	Дата		

Проаналізувавши тенденцію збільшення користувачів платформи інтернет, переходу до електронних ресурсів та об'єму інформації, яку потрібно впорядковувати, а також платформи для контролю бібліотек, підтверджується, що програма «Книжковий органайзер» користуватиметься попитом, оскільки її функціонал надає можливості навести лад зі своїми книгами, їхнім оглядом та є більшим, ніж у її аналогів. Наявні різні органайзери книг, що мають свої унікальні характеристики, але перераховані вище програмні забезпечення створені для більш масштабного використання, а не для контролю домашньої бібліотеки. «Книжковий органайзер» розроблений індивідуально під кожного читача.

Щодо технічного забезпечення, то всі платформи та ресурси для створення програм мають як переваги, так і недоліки. Тому їхній вибір залежить суто від користувача / програміста та його вподобань.

Якщо порівнювати мови програмування, то найлогічнішим є вибрати найновішу мову, яка має функціонал попередніх, тільки дещо модифікованих. Саме тому для створення продукту «Органайзера» використано наступні програми:

- для створення бази даних – MS SQL Server;
- для створення програми – Visual Studio 2017;
- для написання коду – мова програмування C#.

					КНТЕУ 121 06-15.БР	Аркуш
						25
Зм.	Аркуш	№ докум	Підпис	Дата		

РОЗДІЛ 2.

ПОБУДОВА ВАЙРФРЕЙМІВ ТА ТАБЛИЦЬ БАЗИ ДАНИХ ПРОГРАМИ «КНИЖКОВИЙ ОРГАНАЙЗЕР»

2.1. Створення вайрфреймів (wireframes).

2.1.1. Вибір платформи для створення вайрфреймів «Органайзера»

Щоб почати працювати з програмою, варто узгодити, як вона буде виглядати. Візуальна концепція блоків допомагає зорієнтуватися, де і які кнопки будуть розташовані. Це необхідно, щоб визначити функціонал програми, її суть, а також, щоб уявити сам результат роботи, до якого прагне розробник.

Вайрфрейми – це схеми, які показують у деталях, яка інформація, контент й елементи управління повинні відображатися на кожній сторінці інтерфейсу будь-якої системи. У вайрфреймах розставляються акценти щодо різних елементів інтерфейсу: кнопок, тексту, заголовків, зображень тощо.

Створення вайрфреймів вважається одним із перших кроків у розробці інтерфейсів для програм та однією з найважливіших стадій, що впливає на майбутнє продукту, який розробляється. Вайрфрейм, що створюється на ранніх стадіях розробки інтерфейсів, визначає розташування елементів, які користувач бачить на головній панелі, а також те, як вони будуть взаємодіяти з ним.

Непродуманий користувацький інтерфейс часом призводить до того, що люди навіть не починають його використовувати. Зручне розташування кнопок, чекбоксів і інших елементів має велике значення, оскільки вони повинні забезпечувати просту та зручну навігацію всередині програми.

					КНТЕУ 121 06-15.БР			
Зм.	Аркуш	№ докум	Підпис	Дата	Розробка програмного продукту «Книжковий органайзер» Побудова вайрфреймів та таблиць бази даних програми «Книжковий органайзер»	Стадія	Аркуш	Аркушів
Зав. кафедри	Криворучко О.В.					Р2	26	58
Керівник	Жирова Т. О.					Факультет інформаційних технологій, 4 курс, 6 група		
Гарант	Цензура М. О.							
Розроб.	Мороз Я. Р.							

Варто пам'ятати, що вайрфрейми – це схеми сторінок, а не кінцевий дизайн системи. Усі розміри в них відносні, візуалізації елементів є спрощеною, але основні пропорції закладаються вже на цьому етапі. Чорно-сіро-білий кольори – це типова палітра вайрфрейму [29].

Без вайрфреймів важко розробити дизайн створюваної програми, а без продуманої концепції дизайну не варто братися за написання коду, адже чітке уявлення розташування об'єктів допомагає зрозуміти структуру, алгоритми та які блоки можна ділити / групувати за допомогою вибраної мови програмування.

Серед найпопулярніших платформ для створення вайрфреймів є draw.io та Figma.

draw.io – це цілком безкоштовний додаток Google Drive (TM) із вбудованими діаграмами, що уможливорює малювання:

- блок-схем;
- на уніфікованій мові програмування (UML);
- діаграм відношення логічних об'єктів;
- мережевих діаграм;
- моделей бізнес-процесів;
- організаційних діаграм;
- електронних схем.

Також додаток дозволяє створювати інтерактивні прототипи дизайну та макети [10].

Figma – векторний онлайн-сервіс розробки інтерфейсів та прототипування з можливістю організації спільної роботи, що розробляється однойменною компанією. Працює у двох форматах: у браузері та як клієнтський додаток на десктопі користувача. Зберігає онлайн-версії файлів, з якими працював

					КНТЕУ 121 06-15.БР	Аркуш
						27
Зм.	Аркуш	№ докум	Підпис	Дата		

користувач. Даний редактор підходить як для створення простих прототипів і дизайн-систем, так і складних проєктів (мобільні додатки, портали) [11].

Оскільки draw.io має простий інтерфейс і платформа не велика за масштабами, але при цьому цілком задовільняє всі потреби по створенню вайрфреймів, саме цей ресурс було використано для створення блок-схем для «Книжкового органайзера».

2.1.2. Створення першої візуалізації «Книжкового органайзера»

На рисунку 2.1. можна побачити прототип початкової панелі програми. Тут розташований основний функціонал «Органайзера»: верхнє меню, бокове меню, кнопки додавання інформації до бази даних, її редагування та список книг.

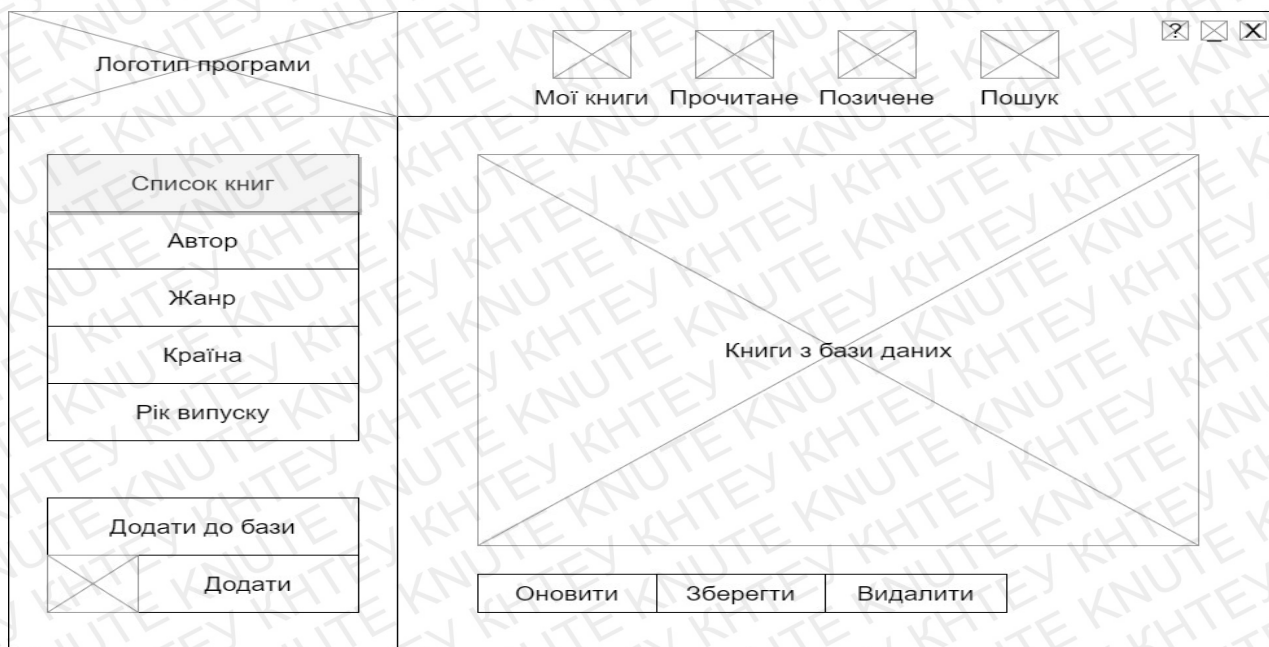


Рис. 2.1. Головна панель програми «Книжковий органайзер»

Прямокутники з діагоналями, що перетинаються означають, що у вибраному місці буде зображення, а ще, як у даному випадку, DataGridView зі списком книг із бази даних.

					КНТЕУ 121 06-15.БР	Аркуш
						28
Зм.	Аркуш	№ докум	Підпис	Дата		

Замальована кнопка свідчить про місце знаходження користувача на даному етапі. Кнопка «Список книг» показує вміст початкової панелі відразу при запуску «Книжкового органайзера».

Усі кнопки верхнього й нижнього меню переключатимуть панелі створюваної програми.

Рис. 2.2. Функціонал кнопки «Країна»

Рисунок 2.2. демонструє функціонал кнопки «Країна». Це визначається тим, що кнопка виділена сірим кольором. Помітно, що справа від бокового меню є Label «Країна», який доносить до відома користувача, що у відповідному текстовій вводить назва шуканої країни. Результат відобразиться нижче у блоці зі списком книг.

					КНТЕУ 121 06-15.БР	Аркуш
						29
Зм.	Аркуш	№ докум	Підпис	Дата		

Візуалізація кнопок «Автор», «Жанр», «Рік випуску» практично ідентична з кнопкою «Країна», тому окремі вайрфрейми для них не розроблялися.

На рисунку 2.3. користувач бачить панель, де відбувається додавання нових книг до програми, якщо такі були придбані / отримані в подарунок. Усі елементи розміщено так, аби не навантажувати потенційного користувача великою кількістю інформації.

Логотип програми		      			
		Мої книги Прочитане Позичене Пошук			
Список книг					
Автор		Країна		Додати	
Жанр		Жанр		Додати	
Країна		Рік випуску		Додати	
Рік випуску		Статус		Додати	
Додати до бази					
 Додати					

Рис. 2.3. Функціонал кнопки «Додати до бази»

На останньому вайрфреймі (рисунок 2.4), який помітно відрізняється від зазначених вище, зображено панель пошуку книги. Детальніше як працювати з даним пошуком зазначено в розділі 3.

					КНТЕУ 121 06-15.БР	Аркуш
						30
Зм.	Аркуш	№ докум	Підпис	Дата		

Логотип програми

Мої книги Прочитане Позичене Пошук

Автор

Книга

Пошук

Рис. 2.4. Панель для пошуку книг, наявних у «Книжковому органайзері»

2.2. Побудова таблиць бази даних «Органайзера».

2.2.1. Концептуальна модель «Органайзера»

Перед створенням бази даних, щоб зрозуміти, з якою інформацією доведеться працювати користувачу, розробляється її концептуальна модель. На рисунку 2.5. зображена концептуальна модель для бази даних органайзера книг для домашньої бібліотеки.

Вона складається з трьох блоків: читачі, книги та облік книг. З рисунку легко помітити, що блоки між собою пов'язані. Очевидним поясненням є те, що вони тісно між собою взаємодіють. Наприклад, у запропонованій домашній бібліотеці є книги з різними авторами, жанрами, датами випуску книг, але всі вони

					КНТЕУ 121 06-15.БР	Аркуш
						31
Зм.	Аркуш	№ докум	Підпис	Дата		

зберігаються в одному місці і, у даному випадку, в одному екземплярі. Кожна книга занесена до програми «Книжковий органайзер», де відбувається її облік та, за допомогою швидкого пошуку, перевіряється, чи кожна книга знаходиться на своєму місці.



Рис. 2.5. Концептуальна модель «Органайзера»

Читачі – безпосередні учасники дій над книгами. Вони клієнти, які перевіряють наявність певної книги, або шукатимуть її за різними критеріями там, де відбувається облік книг. Після того як вони отримують бажаний результат, відбувається домовленість за депозит та дату повернення, яка також записується в облік. Дані користувача записуються, з умовою, що два читачів не можуть мати ідентичний номер телефону та електронної адреси тощо.

2.2.2. Логічна модель бази даних програми «Органайзер»

Наступним кроком для побудови баз даних є створення їхньої логічної моделі. На цьому етапі власник повинен зрозуміти, як усі дані взаємодіють між

					КНТЕУ 121 06-15.БР	Аркуш
						32
Зм.	Аркуш	№ докум	Підпис	Дата		

собою, як вони пов'язані, як визначаються ключові поля та будуються зв'язки між ними.

Спочатку виконується побудова таблиць, які згодом зв'язуються в єдину схему. Важливо враховувати деталі об'єднання таблиць: не може бути так, щоб у таблиці з певним ключем було більше ніж одне ключове слово. Тобто дані не можуть повторюватися в деяких таблицях.

Вибір правильно типу даних також підлягає ретельному підбору, оскільки якщо зазначити неправильний тип, існує імовірність, що програма дасть збій відразу. Розробнику, очевидно, не потрібно такого результату.

Для «Книжкового органайзера» створено сім таблиць. Кожна з них має дані, де одна з колонок – Початковий ключ (Primary Key). Головна таблиця об'єднує дані з усіх таблиць. Схему бази даних можна побачити на рисунку 2.6.

2.2.3. Фізична модель даних «Книжкового органайзера»

Як було зазначено вище, для побудови таблиць без збоїв потрібно встановити правильні типи даних та ключі. Щоб реалізувати цей процес створюється фізична модель «Органайзера».

Така модель потрібна для того, щоб легше записувати дані у програму MS SQL (рисунок 2.7.).

2.2.4. Внесення даних у MS SQL

Наступним, що потрібно зробити для створення бази даних – унести створені таблиці на платформу MS SQL.

Для цього необхідно створити свою базу даних: перейти в пункти Databases, натиснути правку кнопку мишки (ПКМ) і вибрати пункти «Створити нову БД». Відкривається вікно, де можна назвати програму (у даному випадку назва KartotekaDB) і написати кнопку «ОК» (Рис. 2.8.).

					КНТЕУ 121 06-15.БР	Аркуш
						33
Зм.	Аркуш	№ докум	Підпис	Дата		

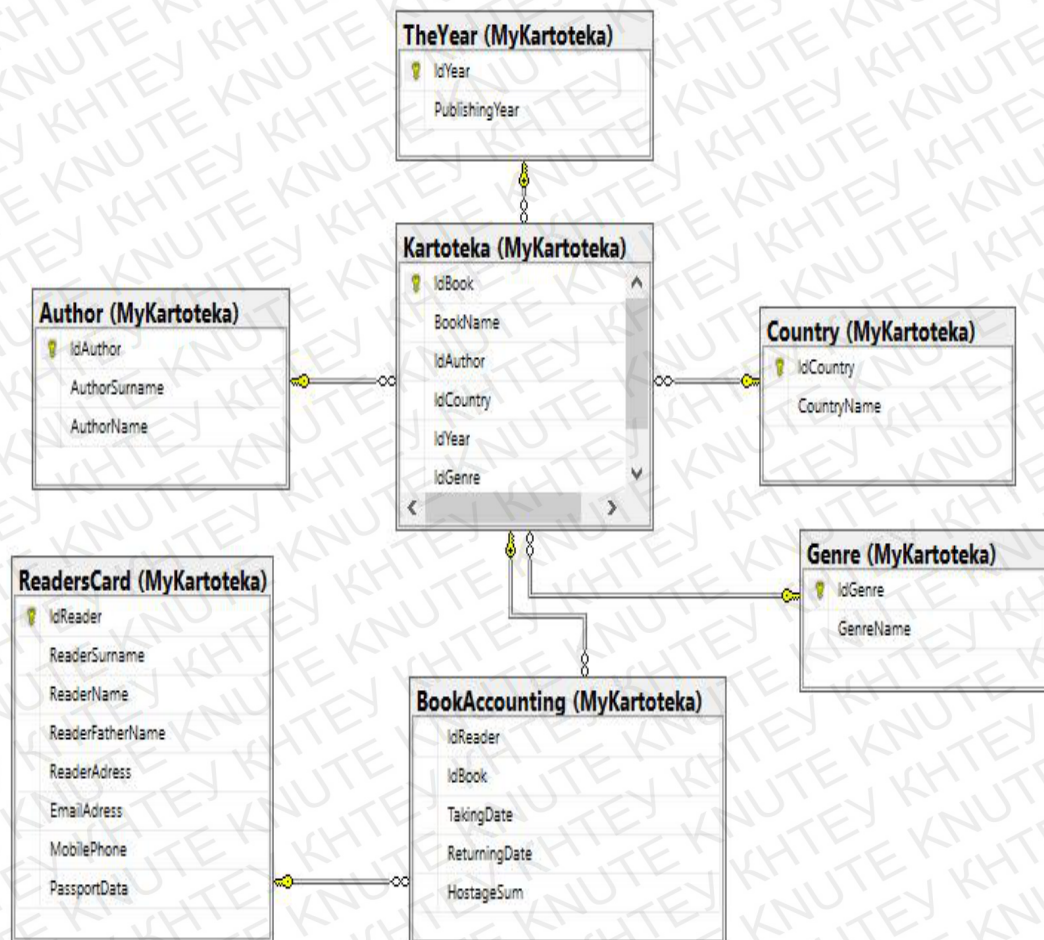


Рис. 2.6. Схема бази даних «Книжкового органайзера»

Перед створенням бази даних можливо ще зробити авторизацію. Для цього потрібно натиснути ПКМ на пункт Security/Logins – New Login, де користувач вибирає свої налаштування: логін, пароль, керування, як це продемонстровано на рисунку 2.9.

Після створення БД наступним кроком потрібно створити схему даних: у на панелі вгорі натиснути New Query. З'явиться місце для написання тексту.

					КНТЕУ 121 06-15.БР	Аркуш
						34
Зм.	Аркуш	№ докум	Підпис	Дата		

Author			Genre			Kartoteka		
Name	Data Type	Key	Name	Data Type	Key	Name	Data Type	Key
IdAuthor	int	Pr.K	IdGenre	int	Pr.K	IdBook	int	Pr.K
AuthorSurname	nvarchar(20)	-	GenreName	nvarchar(20)	-	BookName	nvarchar(60)	-
AuthorName	nvarchar(15)	-				IdAuthor	int	F.K
ReadersCard			TheYear			IdCountry	int	F.K
Name	Data Type	Key	Name	Data Type	Key	IdYear	int	F.K
ReaderSurname	int	Pr.K	IdYear	int	Pr.K	IdGenre	int	F.K
ReaderName	nvarchar(15)	-	PublishingYear	int	-			
ReaderFatherName	nvarchar(15)	-	Country			BookAccounting		
ReaderAddress	nvarchar(20)	-	Name	Data Type	Key	Name	Data Type	Key
EmailAdress	nvarchar(20)	-	IdCountry	int	Pr.K	IdReader	int	F.K
MobilePhone	int	-	CountryName	nvarchar(20)	-	IdBook	int	F.K
PassportData	nvarchar(10)	-				TakingDate	int	-
						ReturningDate	int	-
						HostageSum	int	-

АКТИ

Рис. 2.7. Фізична модель бази даних «Книжкового органайзера»

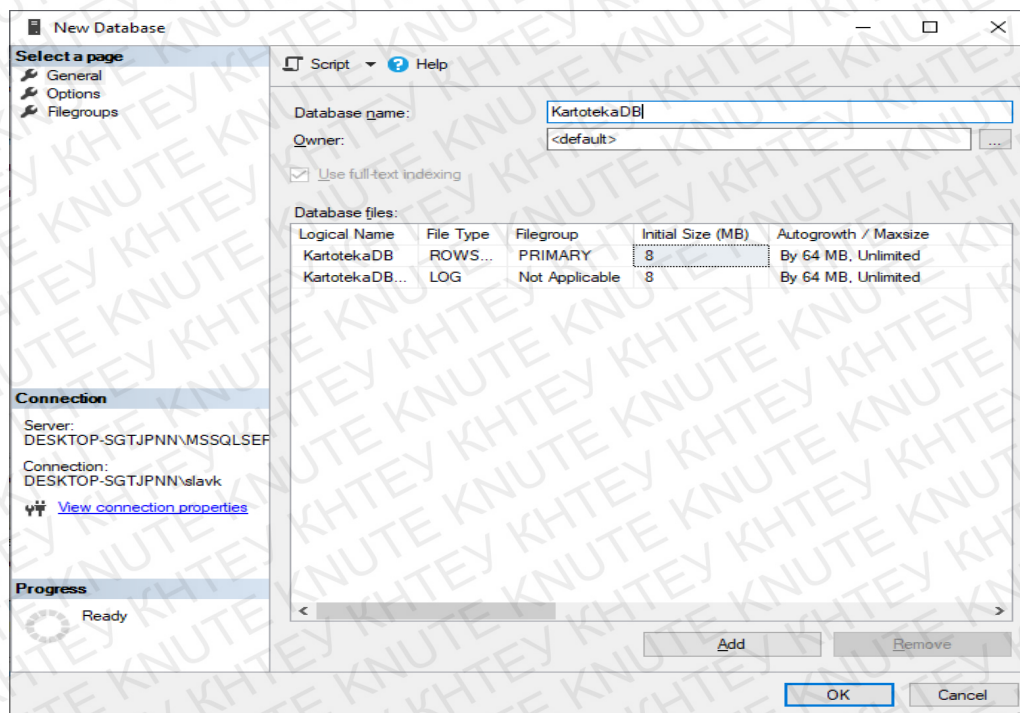


Рис. 2.8. Створення бази даних

					Аркуш	
					КНТЕУ 121 06-15.БР	
Зм.	Аркуш	№ докум	Підпис	Дата	35	

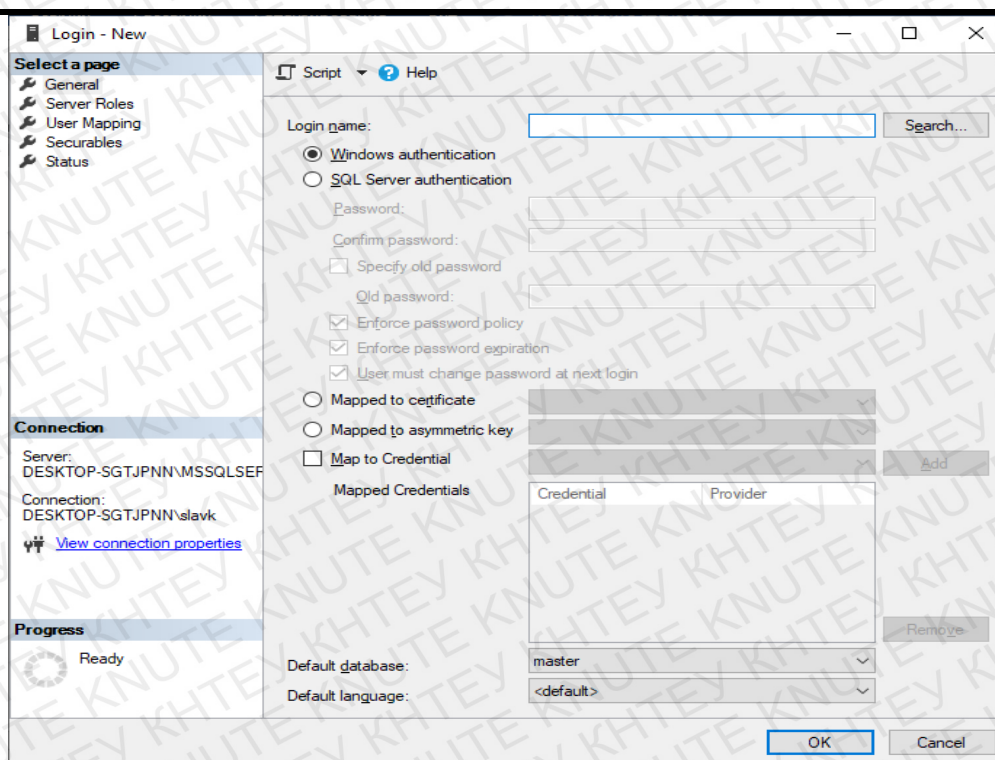


Рис. 2.9. Створення авторизації для баз даних

Щоб записати відповідні дані, використовується фізична модель, створена раніше й виконується код (рис. 2.10.) за зразком:

```
Use KartotekaDB
go
CREATE SCHEMA MyKartoteka
create table Author(IdAuthor int primary key, AuthorSurname nvarchar(20), AuthorName
nvarchar(15))
create table TheYear(IdYear int primary key, PublishingYear int)
create table Genre (IdGenre int primary key, GenreName nvarchar(40))
create table ReadersCard (IdReader int primary key, ReaderSurname nvarchar(15), ReaderName
nvarchar(15), ReaderFatherName nvarchar(15), ReaderAdress nvarchar(40), EmailAdress
nvarchar(40), MobilePhone int, PassportData nvarchar(10))
create table Country (IdCountry int primary key, CountryName nvarchar(20))
create table Kartoteka (IdBook int primary key, BookName nvarchar(20), IdAuthor int foreign key
references Author(IdAuthor), IdCountry int foreign key references Country(IdCountry), IdYear int
foreign key references TheYear(idYear), IdGenre int foreign key references Genre (IdGenre))
create table BookAccounting (IdReader int foreign key references ReadersCard (IdReader), IdBook
int foreign key references Kartoteka (IdBook), TakingDate int, ReturningDate int, HostageSum int)
```

Рис. 2.10. Лістинг створення схеми даних

					КНТЕУ 121 06-15.БР	Аркуш
						36
Зм.	Аркуш	№ докум	Підпис	Дата		

Щоб створити зв'язки між таблицями, потрібно натиснути праву кнопку мишки на пункт Tables/Table... і вибрати бажані таблиці для об'єднання. Після цього можна побачити створену схему даних (рис. 2.6.).

Для того щоб записати інформацію в таблиці, натискається права кнопка мишки на бажаній таблиці, а потім – на пункт Script Table as/INSERT to... і записується код (рис. 2.11).

```
USE [KartotekaDB]
GO

INSERT INTO [MyKartoteka].[Author]
([IdAuthor]
,[AuthorSurname]
,[AuthorName])
VALUES
(1, 'Турянський', 'Осип'),
(2, 'Багрянний', 'Іван'),
(3, 'Костенко', 'Ліна'),
(4, 'Шкляр', 'Василь'),
(5, 'Назарук', 'Осип')
```

Рис. 2.11. Уведення даних у бази даних «Органайзера»

Варто зауважити, що якщо ви ввели неправильний тип даних, то введення інформації стає неможливим. Щоб виправити цю ситуацію потрібно використати функцію Alter (Рис. 2.12.):

```
ALTER TABLE MyKartoteka.Author
ALTER COLUMN AuthorName nvarchar(30)
```

Рис. 2.12. Зміна типу даних у базі даних «Органайзера»

					КНТЕУ 121 06-15.БР	Аркуш
						37
Зм.	Аркуш	№ докум	Підпис	Дата		

Для того щоб вибирати потрібні дані, зв'язуються таблиці за допомогою функції Select та Join (Рис. 2.13.). Результат зв'язування зазначений на полі внизу.

Для використовувати вже створеної бази даних, потрібно приєднати її до коду у вибраній програмі – Visual Studio.

```
USE [KartotekaDB]
Go
Select reader.ReaderName, reader.ReaderSurname, kart.BookName, acco.HostageSum,
acco.TakingDate from MyKartoteka.BookAccounting acco
Join MyKartoteka.ReadersCard reader
On acco.IdReader = reader.IdReader
Join MyKartoteka.Kartoteka kart
On kart.IdBook = acco.IdBook
```

Рис. 2.13. Зв'язування таблиць у базі даних «Органайзера»

2.3. Висновки до розділу 2

Підготовка до розробки програми так само важлива, як і сам процес її реалізації. Щоб візуально побачити, як виглядатиме програма та об'єктивно оцінити її функціонал потрібно створювати блок-схеми – вайрфрейми. Для цього використано веб-застосунок draw.io.

Важливо пам'ятати, що вайрфрейми – тільки прототип, тому розмір блоків та відстані можуть відрізнятися від реальних.

Щодо бази даних, то передусім варто добре обдумати всі аспекти зв'язку між таблицями, де буде зберігатися інформація. Перш за все, потрібно зрозуміти, яким саме має бути продукт, визначити його цільову аудиторію та усвідомити, як ці всі речі пов'язані одне з одним.

					КНТЕУ 121 06-15.БР	Аркуш
						38
Зм.	Аркуш	№ докум	Підпис	Дата		

Концептуальна, логічна і фізична модель – це основи побудови бази даних. Вони допомагають зрозуміти, як взаємодіють між собою таблиці, для чого потрібні ключі, і як ці ключі впливають на внесені дані.

Базу даних можна легко створити, видалити, редагувати, змінювати типи даних за допомогою логічних функцій.

Зрозуміти, як працює база даних нескладно. Варто лише розібратися в принципах її роботи та користуватися підказками в інтернет-ресурсах.

					<i>КНТЕУ 121 06-15.БР</i>	Аркуш
Зм.	Аркуш	№ докум	Підпис	Дата		39

РОЗДІЛ 3.

РЕАЛІЗАЦІЯ ПРОГРАМИ «КНИЖКОВИЙ ОРГАНАЙЗЕР»

3.1. Створення інтерфейсу «Органайзера»

Після того як визначено інформацію в базах даних, яка буде основою створюваної програми, розробляється дизайн інтерфейсу згідно з вимогами, прописаними в технічному завданні та за визначеними вище вайрфреймами.

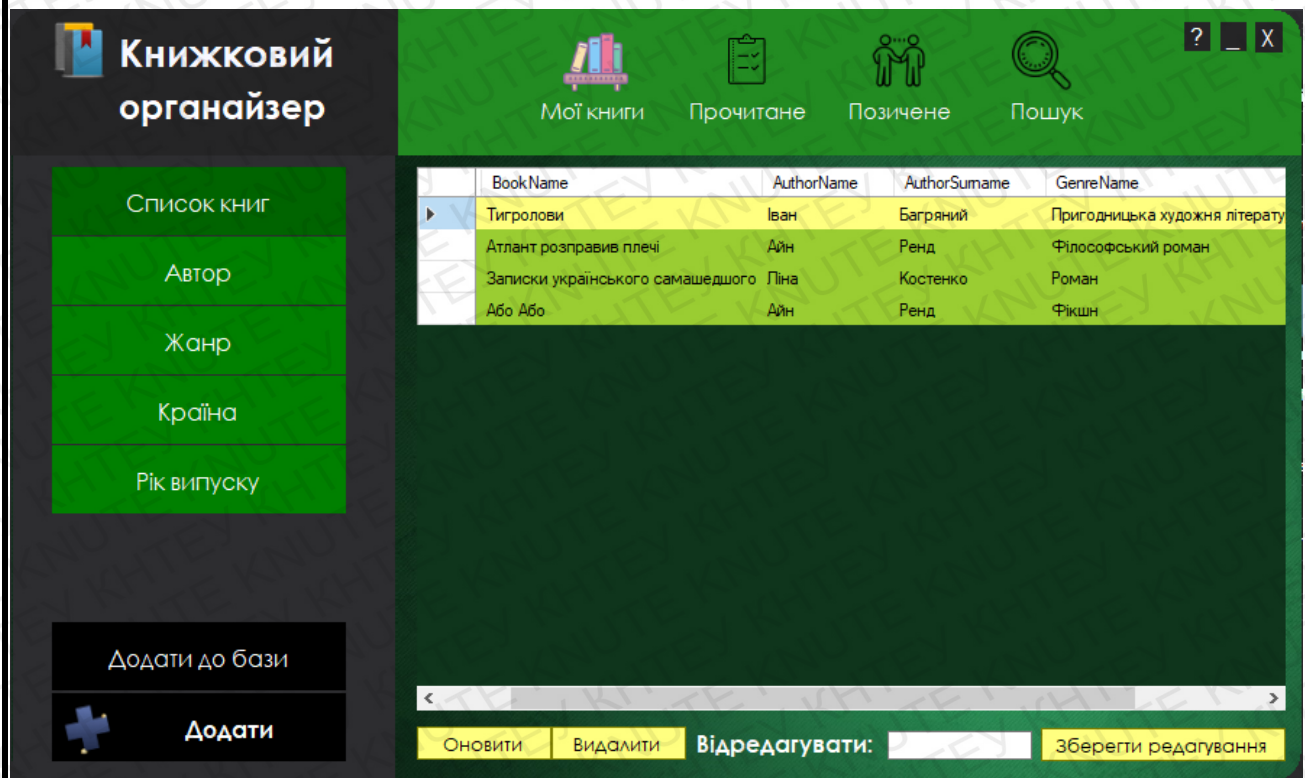


Рис. 3.1. Головна панель програми «Книжковий органайзер»

«Книжковий органайзер» отримав зелено-чорне забарвлення, оскільки ці кольори чудово комбінуються та не є занадто яскравими (рис. 3.1). Крім того,

					КНТЕУ 121 06-15.БР			
Зм.	Аркуш	№ докум	Підпис	Дата	Розробка програмного продукту «Книжковий органайзер»	Стадія	Аркуш	Аркушів
Зав. кафедри	Криворучко О.В.					РЗ	40	58
Керівник	Жирова Т. О.					Факультет інформаційних технологій, 4 курс, 6 група		
Гарант	Цензура М. О.							
Розроб.	Мороз Я. Р.				Реалізація програми «Книжковий органайзер»			

зелений колір вибраний, щоб користувачу було приємно дивитися на панель програми, оскільки відомо, що саме цей колір вгамовує розбурхані емоції, заспокоює і сприяє релаксації. Вибір зеленого завжди позитивний: він, звичайно ж, асоціюється з природою, яка діє оздоровчо та мотивує приязне ставлення до світу. Цей фактор є важливим у дизайні, адже саме він сприятиме залучення користувача до роботи з даним застосунком.

Щоб дизайн не був занадто тьмним, вирішено додати яскравих кольорів: білого та жовтого. Темний блок програми відрізняється від іншої частини програми, щоб користувач розумів, що йому доведеться здебільшого працювати з боковою панеллю, адже саме там можна додавати книги до бази даних, а також шукати книги за наперед визначеними характеристиками.

Інтерактивність у дизайні також потрібна. Тому для більшого залучення уваги зроблено таке:

- зелені кнопки на боковій панелі, коли вони натиснуті, стають темно-зеленими;
- виділена книга зі списку стає жовтою;
- при наведенні курсора мишки на одну з кнопок верхнього меню, зображення іконок забарвлюється (рис. 3.2).

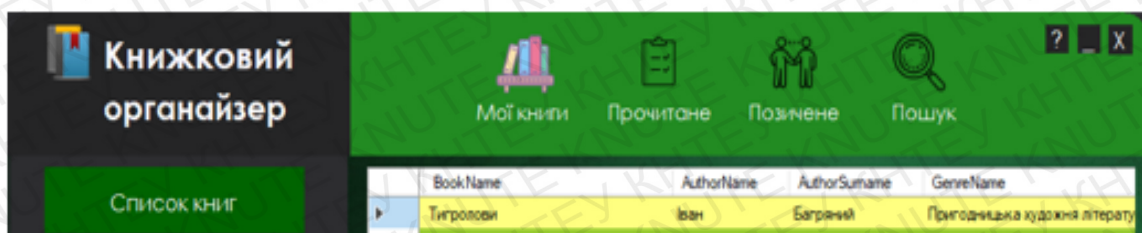


Рис. 3.2. Головне меню «Органайзера»

Так користувачі програми завжди бачитимуть, у якому розділі вони знаходяться саме зараз.

					КНТЕУ 121 06-15.БР	Аркуш
						41
Зм.	Аркуш	№ докум	Підпис	Дата		

3.2. Опис елементів керування меню «Органайзера»

Як було вже зазначено, у програмі створено два меню: верхнє та бокове. Кожне з них має свій функціонал. Головнішим, звісно, є верхнє меню, адже воно не зникає за жодних обставин, тоді як бокове присутнє не на всіх розділах «Органайзера».

Головне меню має тільки чотири кнопки «Мої книги», «Прочитане», «Позичене» та «Пошук». Кнопка «Мої книги» є початковою панеллю даної програми. Натискаючи на неї, можна повернутися до нашого списку книг та бокової панелі.

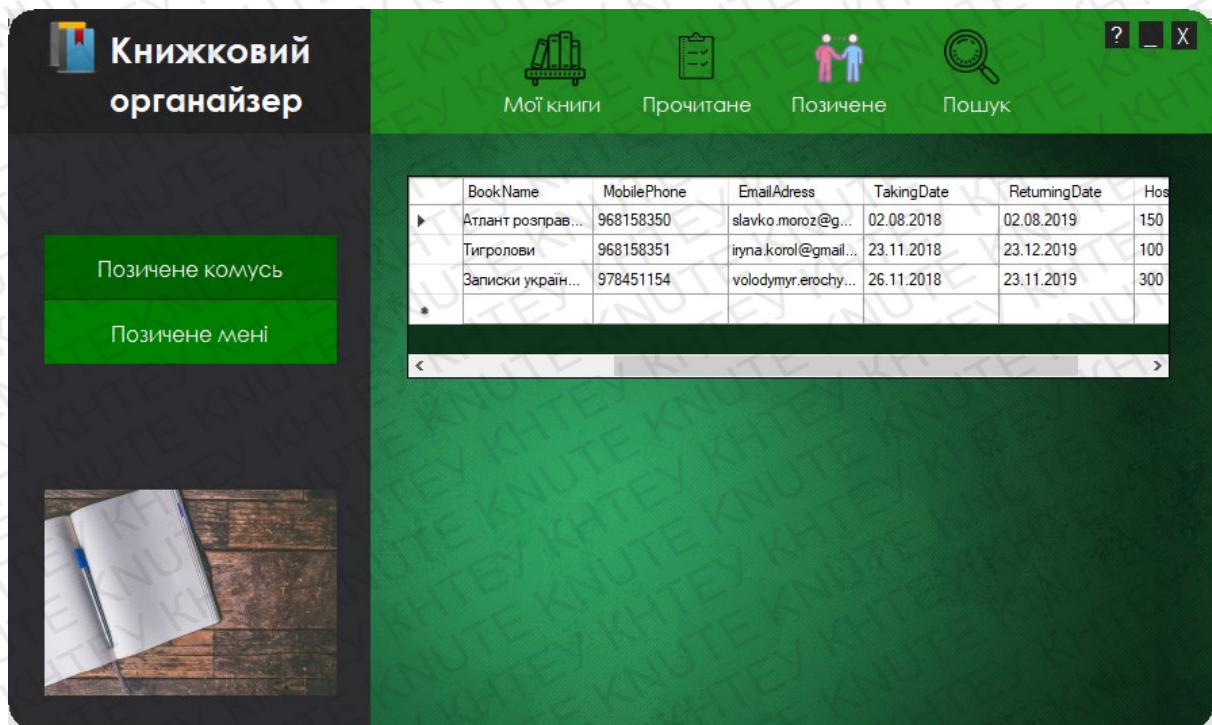


Рис. 3.3. Облік позичених книг

Кнопка «Прочитане» дозволить побачити список книг, відмічених автором, як прочитані. «Позичене» стане хорошим блоком для людей, які люблять

					КНТЕУ 121 06-15.БР	Аркуш
Зм.	Аркуш	№ докум	Підпис	Дата		42

буккросинг. Тут можна знайти книги, які ви позичили своїм знайомим, дізнатися час, коли це було зроблено, та отримати всю інформацію про відповідну особу:

- її ім'я та прізвище для ідентифікації;
- номер телефону, якщо виникне бажання дізнатися деталі повернення книги;
- електронна пошта, якщо не вдається скомунікувати по телефону.

Більше того, саме в цьому блоці можна побачити, скільки грошей доведеться заплатити людині, що позичила книгу, у разі її неповернення або зіпсування. Як правило, ціна оцінюється як собівартість книги (рис. 3.3).

Інша можливість розрахована для обліку книг, які власник програми позичив для себе. Для перегляду такої ж інформації варто натиснути кнопку «Позичене мені» (рис. 3.4).

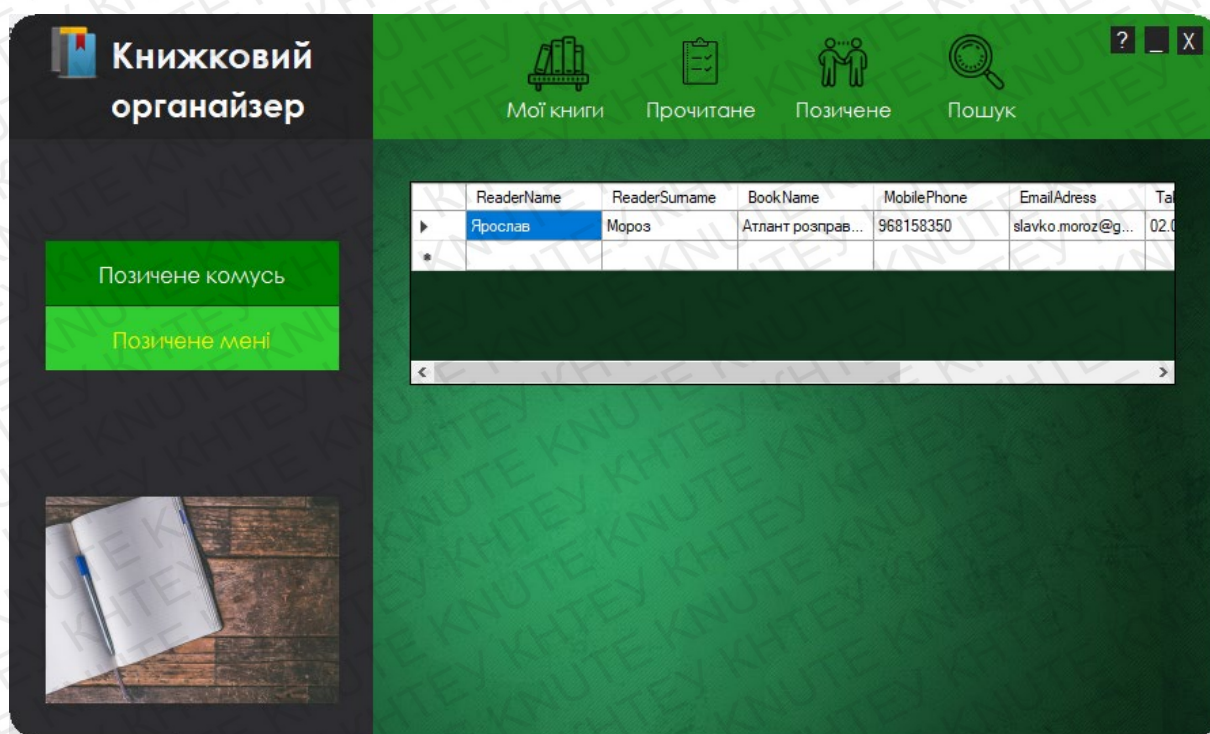


Рис. 3.4. Облік книг, позичених користувачем програми

					КНТЕУ 121 06-15.БР	Аркуш
						43
Зм.	Аркуш	№ докум	Підпис	Дата		

Пункт «Пошук» цікавий тим, що з його допомогою можна знайти книгу як «за автором», «за назвою книги», так і за комбінацією того й іншого. Пошук відбувається за будь-якими параметрами. До прикладу, можна написати в пошуку «Ліна Костенко» – знайдуться всі книги авторки. Якщо ввести просто «Ліна», відповідно знайдуться всі книги авторки з таким ім'ям. Для пошуку навіть не потрібно вводити повну назву, достатньо написати одну букву і почнеться пошук (рис. 3.5). Якщо дані не введені, то відкривається повідомлення з помилкою «Введіть дані».

Якщо повернутися до бокової панелі, то тут розташовані такі кнопки, як «Список книг», «Автор», «Жанр», «Країна», «Рік випуску», «Додати до бази» та «Додати».

Рис. 3.5. Пошук книги

Завдання кнопок – допомогти користувачу знайти категорію, яка допоможе знайти книгу за різними характеристиками. При натисканні на «Мої книги» для

					КНТЕУ 121 06-15.БР	Аркуш
						44
Зм.	Аркуш	№ докум	Підпис	Дата		

користувача автоматично відкривається категорія «Список книг», з переліком самостійно доданих у базу даних творів мистецтва.

На рисунку 3.6. показано, що знаходиться в блоці «Список книг». Тут користувач буде візуально бачити перелік усіх книжок, наявних у його бібліотеці та зможе редагувати їх назву, якщо вона була внесена в базу даних з помилками. Саме для цього є кнопки «Оновити», «Зберегти редагування», «Видалити», що виконують відповідні функції. Щоб змінити дані, потрібно просто натиснути на місце потенційного редагування та вписати правильну інформацію у відповідному полі нижче.

	IdBook	BookName	AuthorName	AuthorSurname	GenreName
	2	Тигролови	Іван	Багряний	Пригодницька худож.
	3	Атлант розправив плечі	Айн	Ренд	Філософський роман
	17	Записки українського самашедшого	Ліна	Костенко	Роман
▶	18	Або Або	Айн	Ренд	Фікшн

Оновити
Видалити
Відредагувати:
Зберегти редагування

Рис. 3.6. Редагування інформації про книгу

					КНТЕУ 121 06-15.БР	Аркуш
						45
Зм.	Аркуш	№ докум	Підпис	Дата		

Щодо кнопок «Автор», «Жанр», «Країна» і «Рік випуску», то вони мають однакове завдання. При їх натисканні з'являється поле з коміркою, де можна здійснити пошук за автором, жанром, країною і роком відповідно, вибираючи з поля той пункт, який найбільше до вподоби.

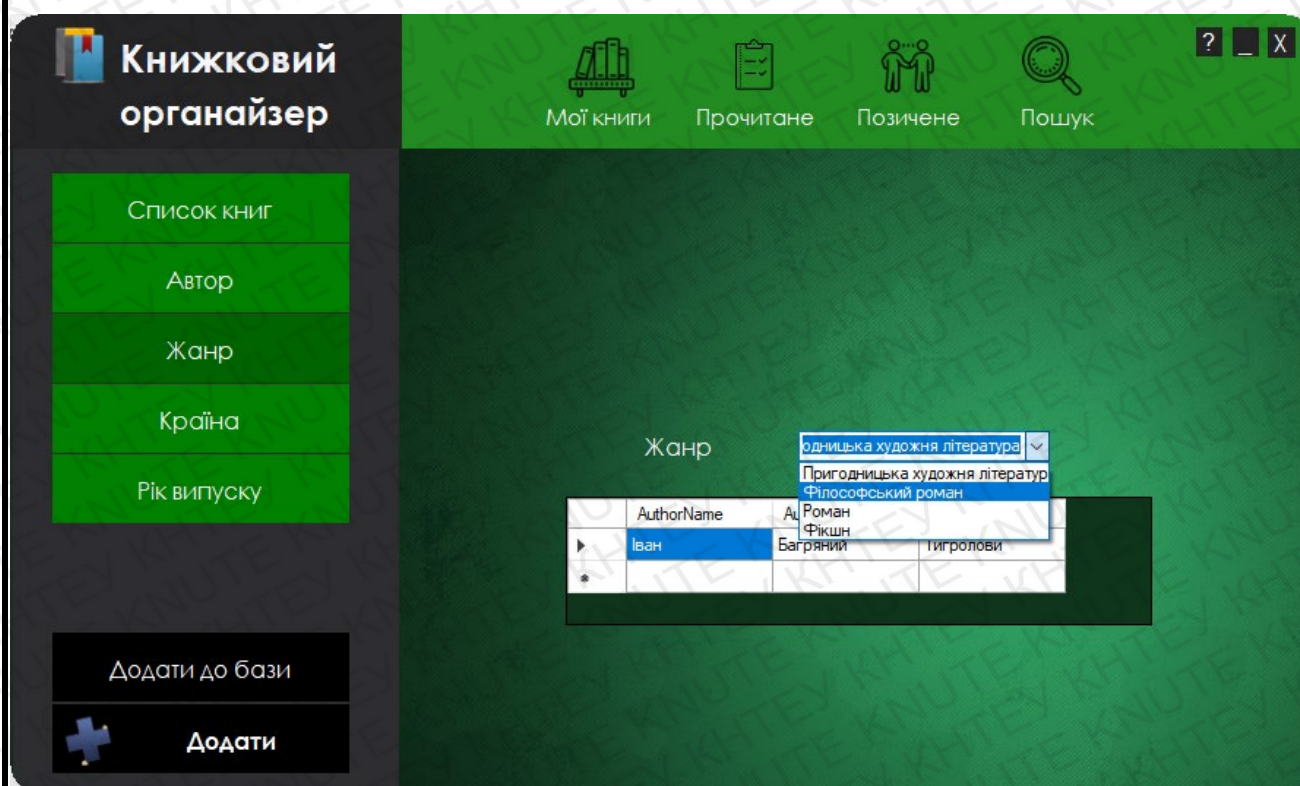


Рис. 3.7. Вибір книг за відповідними характеристиками

З іншого боку, це навіть допомагатиме користувачам відрізняти жанри книг та порівнювати стилі написання в різних країнах та епохах.

3.3. Підключення баз даних до програми «Книжковий органайзер»

Щоб приєднати базу даних до створеного продукту, потрібно використовувати код, зазначений нижче.

					КНТЕУ 121 06-15.БР	Аркуш
						46
Зм.	Аркуш	№ докум	Підпис	Дата		


```

using System.Data.SqlClient;

namespace Kartoteka_your_biblioteka
{
    public partial class asearch : UserControl
    {
        private const string ConnectionString = @"Data Source=DESKTOP-
SGTJPNN\MSSQLSERVER2;Initial Catalog=KartotekaDB;Integrated
Security=True";
        string a;
        SqlConnection connection = new SqlConnection(ConnectionString);

```

Рис. 3.8. Приєднання бази даних до «Органайзера»

3.4. База даних як невід’ємна частина «Книжкового органайзера»

Щоб звичайні користувачі не додавали книги через MySQL, існує кнопка «Додати до бази». Якщо перейти на цю панель, легко помітити поля, куди можна записати відповідну інформацію (рис. 3.9).

На цій формі є п’ять текстбоксів, для введення даних. Інформація записується, якщо читат придбав / отримав книгу автора, про якого ще не було згадок у базі даних. Після запису варто тільки натиснути кнопку «Додати» і база поновлюється новими даними. Щоб пересвідчитися, що це дійсно так, автоматично відкривається вікно-повідомлення про успішне додавання матеріалу, а в разі, коли дані продубльовано – що дані вже існують. Якщо ж не інформація відсутня, «Органайзер» попросить увести матеріал.

					<i>КНТЕУ 121 06-15.БР</i>	Аркуш
						47
Зм.	Аркуш	№ докум	Підпис	Дата		

Рис. 3.9. Додати до бази

Єдина кнопка, про яку ще не було розказано – «Додати» (рис. 3.10). Її функція одна з найважливіших, адже саме ця панель збирає докупи всю інформацію про книгу. Завдання панелі «Додати» – функцією якої є додавати зв'язану інформацію до бази даних про книги так, щоб усі комірки мали тип String. Якщо нічого не записати у комірки, то натиск на «Добавити» викличе повідомлення «Не всі комірки заповнені!».

Ще однією функцією блоку є автоматизація випадаючого списку. Якщо в панелі «Додати до бази» записувалися ім'я та прізвище автора – ця інформація стане в нагоді саме тут. Аби зрозуміти процес автоматизації, потрібно вибрати ім'я / прізвище автора. Після цього, відповідно, прізвище / ім'я автора книги з'явиться автоматично.

					КНТЕУ 121 06-15.БР	Аркуш
						48
Зм.	Аркуш	№ докум	Підпис	Дата		

Рис. 3.10. Додавання книги до списку

Як тільки всі дані про книгу додані, потрібно натиснути центральну кнопку «Додати» і все – книжка в базі даних. Щоб побачити її на відповідній панелі, варто перейти на блок «Список книг» і натиснути кнопку «Оновити» (рис. 3.11), щоб дані з'явилися у стрічці.

	BookName	AuthorName	AuthorSumame	GenreName
	Тигролови	Іван	Багрянний	Пригодницька художня літерат
	Атлант розправив плечі	Айн	Ренд	Філософський роман
	Записки українського самашедшого	Ліна	Костенко	Роман
	Або Або	Айн	Ренд	Фікшн
▶	Війна за реальність	Дмитро	Кулеба	Політинчна література

Рис. 3.11. Результат додавання книги до списку

Для того щоб з'єднати між собою дані таблиць, потрібно ввести відповідний код (рис. 3.12).

					КНТЕУ 121 06-15.БР	Аркуш
						49
Зм.	Аркуш	№ докум	Підпис	Дата		


```

public Spysokknyh()
{
    InitializeComponent();
    knyhy1.BringToFront();
    string str = @"Data Source=DESKTOP-SGTJPN\MSSQLSERVER2;Initial
Catalog=KartotekaDB;Integrated Security=True";
    SqlConnection connection = new SqlConnection(str);
    string sql = "Select gen.GenreName from [MyKartoteka].Genre gen" +
        " Join [MyKartoteka].Kartoteka kart" +
        " On gen.IdGenre = kart.IdGenre";
    SqlDataAdapter da = new SqlDataAdapter();
    SqlCommand cmd;
    DataSet ds = new DataSet();
    try{
        cmd = new SqlCommand(sql, connection);
        da.SelectCommand = cmd;
        da.Fill(ds);
        da.Dispose();
        cmd.Dispose();
        connection.Close();
        comboBox1.DataSource = ds.Tables[0];
        comboBox1.DisplayMember = "GenreName";
    }
    catch (Exception ex) {
        MessageBox.Show(ex.Message);
    }
}

```

Рис. 3.12. Код для приєднання таблиць

Усе відбувається поетапно:

- 1) прив'язується база даних до Microsoft Visual Studio;
- 2) додається код, щоб відбувався обмін інформацією між головною таблицею списку даних та іншими;
- 3) вибирається інформація, яку ми хочемо бачити в програмі (рис. 3.13).

Дані можна вводити і з самої програми. Щоб заповнювати їх автоматично в базу даних, потрібно використати функцію INSERT у коді, зразок якого зазначений нижче (рис. 3.14).

					КНТЕУ 121 06-15.БР	Аркуш
						50
Зм.	Аркуш	№ докум	Підпис	Дата		

```

private void comboBox1_SelectedIndexChanged(object sender, EventArgs e)
{
    SqlDataAdapter sda = new SqlDataAdapter("Select kart.BookName,
author.AuthorName, author.AuthorSurname, gen.GenreName from [MyKartoteka].[Kartoteka] kart"
+
    " Join [MyKartoteka].Genre gen" +
    " On gen.IdGenre = kart.IdGenre" +
    " Join [MyKartoteka].Author author" +
    " On kart.IdAuthor = author.IdAuthor" +
    " Join [MyKartoteka].BookAccounting acc" +
    " On kart.IdBook=acc.IdBook where GenreName like '" + comboBox1.Text + "' ",
connection);
    DataTable dt = new DataTable();
    sda.Fill(dt);
    dataGridView1.DataSource = dt;
}

```

Рис. 3.13. Вибір даних для виведення на екран

```

if (textBoxCountry.Text == "")
{
    MessageBox.Show("Введіть дані");
}
else
{
    string insertQuery = "INSERT INTO Country(CountryName) VALUES('" +
textBoxCountry.Text + "')";
    executeQuery(insertQuery);
    textBoxCountry.Text = "";
}
active();

```

Рис. 3.14. Додавання даних у базу

3.5. Висновки до розділу 3

Хоча програма й невелика, та вона має багато функцій і нею цікаво користуватися. Урешті-решт були знайдені відповіді на всі ключові запитання: які ІТ-платформи найкраще підходять для програми «Книжковий органайзер», чи є попит на програми обліку книжок, і чи створена програма зможе конкурувати на ринку. База даних «Органайзера» була успішно створена, дані записані,

					КНТЕУ 121 06-15.БР	Аркуш
						51
Зм.	Аркуш	№ докум	Підпис	Дата		

здатність додавати / видаляти інформацію присутня. Створений інтерфейс легкий для сприйняття та ненав'язливий для очей. Сам «Органайзер» зручний для користування, тому підходить навіть дітям, що ходять в молодші класи. Крім того, цю програму можна використовувати для саморозвитку.

					КНТЕУ 121 06-15.БР	Аркуш
						52
Зм.	Аркуш	№ докум	Підпис	Дата		

ВИСНОВКИ ТА ПРОПОЗИЦІЇ

Ритм двадцять першого століття дуже впливає на розвиток суспільства. Кожному школяру, студенту чи працівнику потрібно постійно вчитися, щоб устигати за розвитком нових технологій. Для цього варто тримати все в порядку: чи то очистити комп'ютер від зайвого мотлоху, чи впорядкувати інформацію в своєму віртуальному планері. А яке навчання без книг, які проносять не тільки користь, а ще й дають змогу відпочити?

Щоб розуміти, які книги є у власному домі, мати про них коротку інформацію, та ще й вести їхній облік, запропонована програма «Книжковий органайзер». Ця програма значно полегшить життя кожному, хто намагається привести до ладу своє життя.

Презентована програма «Органайзер» доволі успішна. Було реалізовано значну частину ідей, як має функціонувати даний ресурс. Розробка проводилася поетапно: проведено брейнсторминг ідей, створено технічне завдання, проведено маркетингове дослідження, створено вайрфрейми, вибрано платформи для реалізації, розроблено дизайн та, власне, реалізовано задум продукту.

Інтерфейс розроблено доволі простим та легким для сприйняття. Те, що він приваблює з першого погляду, дозволить заохотити нових користувачів спробувати покористуватися програмою. Великим успіхом є те, що користувачі матимуть змогу створювати особистий список прочитаних книг, а також завжди знатимуть, де знаходяться їхні книги.

Ця програма чудово покаже себе й у навчальному процесі, оскільки можна зберігати інформацію про величезну кількість книг, про те, хто їх автор, де вони

					КНТЕУ 121 06-15.БР			
Зм.	Аркуш	№ докум	Підпис	Дата	Розробка програмного продукту «Книжковий органайзер»	Стадія	Аркуш	Аркушів
Зав. кафедри	Криворучко О.В.					ВП	53	58
Керівник	Жирова Т. О.					Факультет інформаційних технологій, 4 курс, 6 група		
Гарант	Цензура М. О.							
Розроб.	Мороз Я. Р.				Висновки та пропозиції			

були написані, у якому жанрі. Згодом цю інформацію можна буде використовувати у якихось наукових цілях, порівняннях.

«Книжковий органайзер» написаний на мові C#. На цій мові легко програмувати, бо її новітні характеристики таки дійсно допомагають розвивати свої знання й уміння.

Для функціоналу програми було створено, а згодом і підключено базу даних. Важливими засобами для створення бази даних є її моделі: концептуальна, логічна і фізична. Усі вони обґрунтовують поведінку даних та показують, що потрібно зробити для функціональності продукту. А також, дані в цій базі можна легко редагувати за допомогою відповідних кнопок програми.

Невід’ємним пунктом є вибір платформи для написання коду та вибір мови програмування. Для того щоб обрати найвідповіднішу, потрібно порівняти кілька актуальних, найпопулярніших платформ і зрозуміти, чи саме вона задовольняє ваші потреби. Правда, підбір платформи відбувається індивідуально для кожного розробника.

Оскільки база даних робить програму надзвичайно корисною та легкою у використанні, її розробник має пропозицію зробити веб-версію «Книжкового органайзера» для майбутнього просування в маси. У цій версії планується створити рейтинг книг, можливість додавати відгуки та зробити сторінку з плануванням, де разом із друзями користувачі визначатимуть, яку книгу мають прочитати за вибраний заздалегідь обмежений час. Це не тільки зближуватиме друзів, доданих у мережу, а й сприятиме новому буккросингу.

Отже, «Книжковий органайзер» – це програма, яка не закликає людей проводити більше часу за гаджетами, а навпаки допомагає їм упорядкувати свої книги, щоб не витратити час на такі зайві речі, як пошук позиченої книги у величезній бібліотечці. З допомогою бази даних, ця програма зекономить усім

					КНТЕУ 121 06-15.БР	Аркуш
						54
Зм.	Аркуш	№ докум	Підпис	Дата		

час на вибір книжки та просто полегшити життя, щоб мати змогу нарешті зрозуміти цікаві та загадкові криптограми з дитинства.

					КНТЕУ 121 06-15.БР	Аркуш
						55
Зм.	Аркуш	№ докum	Підпис	Дата		

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

Основний

1. Берко А. Ю., Верес О. М., Пасічник В. В. Системи баз даних. Книга 2. Система управління базами даних та знань / А. Ю. Берко, В. В. Пасічник, О. М. Верес, 2013. – 680 с.
2. Бьюлі А. Вивчаємо SQL / Алан Бьюлі, 2007. – 312 с.
3. Документація Stack Overflow. Microsoft SQL Server, 2018 – 286 с.
4. Коноваленко І. В. Програмування мовою C# 6.0 / Ігор Коноваленко, 2016. – 227 с.
5. Костенко Л. Й. Бібліотечні електронні інноваційні технології / Л. Й. Костенко // Бібл вісн. – 2003. – № 6. – С. 13–17.
6. Серов Ю., Ковальчук М. Огляд програмних засобів для автоматизації роботи шкільної бібліотеки / Юрій Серов, Марина Ковальчук, 2012. – 279 с.
7. Сочинська Н. А. Організація роботи шкільної бібліотеки. / Н. А. Сочинська. – Миколаїв, 2008. – Комп'ютерні системи та мережі. – 72 с.
8. Шилдта Г. Повне керування C# / Герберта Шилдта, 2011. – 1056 с.

Інтернет-ресурси

9. Database (DB). [Електронний ресурс] – Режим доступу: <https://searchsqlserver.techtarget.com/definition/database>
10. Draw.io. [Електронний ресурс] – Режим доступу: <https://chrome.google.com/webstore/detail/diagramsnet/onlkggianjhjenigcpigpjehhpplldkc?hl=uk>

					КНТЕУ 121 06-15.БР			
Зм.	Аркуш	№ докум	Підпис	Дата	Розробка програмного продукту «Книжковий органайзер»	Стадія	Аркуш	Аркушів
Зав. кафедри	Криворучко О.В.					СД	56	58
Керівник	Жирова Т. О.					Факультет інформаційних технологій, 4 курс, 6 група		
Гарант	Цензура М. О.							
Розроб.	Мороз Я. Р.							
					Список використаних джерел			

11. Figma. [Електронний ресурс] – Режим доступу: <https://www.figma.com/pricing/>
12. Java Language basics. [Електронний ресурс] – Режим доступу: <https://developer.ibm.com/tutorials/j-introtojava1/>
13. Learn C++ Programming. [Електронний ресурс] – Режим доступу: <https://www.programiz.com/cpp-programming>
14. MySQL. [Електронний ресурс] – Режим доступу: <https://dev.mysql.com/doc/>
15. PostgreSQL. [Електронний ресурс] – Режим доступу: <https://uk.wikipedia.org/wiki/PostgreSQL>
16. SQL Tutorial. [Електронний ресурс] – Режим доступу: <https://www.w3schools.com/sql/>
17. SQL. Introduction for beginners. [Електронний ресурс] – Режим доступу: <https://learntocodewith.me/posts/sql-guide/>
18. SQL-програмування. [Електронний ресурс] – Режим доступу: <https://www.quora.com/What-is-SQL-programming>
19. AIBC Unilib. [Електронний ресурс] – Режим доступу: http://onua.edu.ua/index.php?option=com_content&view=article&id=205&Itemid=48&lang=uk
20. Бази даних. [Електронний ресурс] – Режим доступу: <http://lib.mdpu.org.ua/e-book/vstup/L5.htm>
21. Дослідженням Грея Вільямса [Електронний ресурс] – Режим доступу: <http://bit.ly/3a7zg9j>
22. Мова програмування C#. [Електронний ресурс] – Режим доступу: https://uk.wikibooks.org/wiki/C_Sharp

					КНТЕУ 121 06-15.БР	Аркуш
						57
Зм.	Аркуш	№ докум	Підпис	Дата		

23. Мова програмування C#. Огляд мови C#. [Електронний ресурс] – Режим доступу: <http://ci-sharp.ru/ukr/Teaching/mova-programmirovaniya-s-obzor-yazika-s.html>

24. Огляд мови програмування C#. [Електронний ресурс] – Режим доступу: http://stud.com.ua/97448/informatika/oglyad_movi_programuvannya

25. Пакет програм «Бібліограф». [Електронний ресурс] – Режим доступу: <http://www.politek-soft.kiev.ua/index.php?do=products&product=bibliographer>

26. Середовище розробки Microsoft Visual Studio. [Електронний ресурс] – Режим доступу: http://informatics.in.ua/programming_csharp/part_01.php

27. СУБД Oracle. [Електронний ресурс] – Режим доступу: <https://uk.wikipedia.org/wiki/Oracle>

28. Український фондовий дім. [Електронний ресурс] – Режим доступу: <http://ush.com.ua/#>

29. Що таке вайрфрейм (wireframe). [Електронний ресурс] – Режим доступу: <https://goldwebsolutions.com/uk/blog/shho-take-vajrfrejwm-wireframe-ta-dlya-chogovin-potriben-u-protsesi-rozrobki-sajtu-chi-dodatku/>

					КНТЕУ 121 06-15.БР	Аркуш
						58
Зм.	Аркуш	№ докум	Підпис	Дата		

ДОДАТКИ

Додаток А

```
Use KartotekaDB
go
CREATE SCHEMA MyKartoteka
create table Author(IdAuthor int primary key, AuthorSurname nvarchar(20), AuthorName
nvarchar(15))
create table TheYear(IdYear int primary key, PublishingYear int)
create table Genre (IdGenre int primary key, GenreName nvarchar(40))
create table ReadersCard (IdReader int primary key, ReaderSurname nvarchar(15), ReaderName
nvarchar(15), ReaderFatherName nvarchar(15), ReaderAdress nvarchar(40), EmailAdress
nvarchar(40), MobilePhone int, PassportData nvarchar(10))
create table Country (IdCountry int primary key, CountryName nvarchar(20))
create table Kartoteka (IdBook int primary key, BookName nvarchar(20), IdAuthor int foreign key
references Author(IdAuthor), IdCountry int foreign key references Country(IdCountry), IdYear int
foreign key references TheYear(idYear), IdGenre int foreign key references Genre (IdGenre))
create table BookAccounting (IdReader int foreign key references ReadersCard (IdReader), IdBook
int foreign key references Kartoteka(IdBook), TakingDate int, ReturningDate int, HostageSum int)
```

Рис. А.1. Лістинг створення схеми даних

```
using System.Data.SqlClient;

namespace Kartoteka_your_biblioteka
{
    public partial class asearch : UserControl
    {
        private const string ConnectionString = @"Data Source=DESKTOP-
SGTJPNM\MSSQLSERVER2;Initial Catalog=KartotekaDB;Integrated
Security=True";
        string a;
        SqlConnection connection = new SqlConnection(ConnectionString);
```

Рис. А.2. Приєднання бази даних до «Органайзера»

```

public Spysokknyh()
{
    InitializeComponent();
    knyhy1.BringToFront();
    string str = @"Data Source=DESKTOP-SGTJPN\MSSQLSERVER2;Initial
Catalog=KartotekaDB;Integrated Security=True";
    SqlConnection connection = new SqlConnection(str);
    string sql = "Select gen.GenreName from [MyKartoteka].Genre gen" +
        " Join [MyKartoteka].Kartoteka kart" +
        " On gen.IdGenre = kart.IdGenre";
    SqlDataAdapter da = new SqlDataAdapter();
    SqlCommand cmd;
    DataSet ds = new DataSet();
    try{
        cmd = new SqlCommand(sql, connection);
        da.SelectCommand = cmd;
        da.Fill(ds);
        da.Dispose();
        cmd.Dispose();
        connection.Close();
        comboBox1.DataSource = ds.Tables[0];
        comboBox1.DisplayMember = "GenreName";
    }
    catch (Exception ex) {
        MessageBox.Show(ex.Message);
    }
}

```

Рис. А.3. Створення схеми бази даних «Книжкового органайзера»

```

private void comboBox1_SelectedIndexChanged(object sender, EventArgs e)
{
    SqlDataAdapter sda = new SqlDataAdapter("Select kart.BookName,
author.AuthorName, author.AuthorSurname, gen.GenreName from [MyKartoteka].[Kartoteka] kart"
+
    " Join [MyKartoteka].Genre gen" +
    " On gen.IdGenre = kart.IdGenre" +
    " Join [MyKartoteka].Author author" +
    " On kart.IdAuthor = author.IdAuthor"+
    " Join [MyKartoteka].BookAccounting acc" +
    " On kart.IdBook=acc.IdBook where GenreName like '"+ comboBox1.Text + "' ",
connection);
    DataTable dt = new DataTable();
    sda.Fill(dt);
    dataGridView1.DataSource = dt;
}

```

Рис. А.4. Вибір даних у вже підключеній базі даних


```

if (textBoxCountry.Text == "")
{
    MessageBox.Show("Введіть дані");
}
else
{
    string insertQuery = "INSERT INTO Country(CountryName) VALUES('" +
textBoxCountry.Text + "')";
    executeQuery(insertQuery);
    textBoxCountry.Text = "";
}
active();

```

Рис. А.5. Додавання даних у базу

```

{
    connection.Open();
    string query = "update [MyKartoteka].Kartoteka set BookName = '" +
textBox6.Text + "' where IdBook = " + a;
    textBox6.Text = "";
    SqlDataAdapter adapter1 = new SqlDataAdapter(query, connection);
    DataTable table = new DataTable();
    adapter1.Fill(table);
    connection.Close();
    MessageBox.Show("Оновіть для отримання результату");
    try {
        cmdbl = new SqlCommandBuilder(adapter1);
        adapter1.Update(table);
    }
    catch (Exception ex)
    {
    }
}

```

Рис. А. 6. Оновлення даних у базі «Органайзера»


```

connection.Open();
string query = "delete from [MyKartoteka].BookAccounting where IdBook = " + a;
SqlCommand command = new SqlCommand(query, connection);
command.ExecuteNonQuery();
connection.Close();
connection.Open();
string query2 = "delete from [MyKartoteka].Kartoteka where IdBook = " + a;
SqlCommand command2 = new SqlCommand(query2, connection);
command2.ExecuteNonQuery();
connection.Close();

if (dataGridView1.CurrentRow != null)
{
    selectedrow = dataGridView1.CurrentRow.Index;
    dataGridView1.Rows.RemoveAt(selectedrow);
}
else
{
    MessageBox.Show("Дані видалено!");
}
}

```

Рис. А. 7. Видалення даних з бази «Органайзера»