

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Національний університет «Запорізька політехніка»

Факультет комп'ютерних наук і технологій

(повне найменування факультету)

Кафедра програмних засобів

(повне найменування кафедри)

Пояснювальна записка

до дипломного проєкту (роботи)

бакалавр

(ступінь вищої освіти)

на тему ПРОГРАМНЕ ЗАБЕЗПЕЧЕННЯ ІНФОРМАЦІЙНОЇ СИСТЕМИ
ДЛЯ КЕРУВАННЯ НОТАТКАМИ КОРИСТУВАЧА
SOFTWARE FOR AN INFORMATION SYSTEM
FOR MANAGING USER NOTES

Виконав(ла): студент(ка) 4 курсу, групи КНТ-212сп
Спеціальності 122 Комп'ютерні науки
(код і найменування спеціальності)

Освітня програма (спеціалізація)
Комп'ютерні науки

ХЛЕПІТЬКО К.В.

(ПРИЗВИЩЕ та ініціали)

Керівник КОЛПАКОВА Т.О.

(ПРИЗВИЩЕ та ініціали)

Рецензент ІЛЛЯШЕНКО М.Б.

(ПРИЗВИЩЕ та ініціали)

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Національний університет «Запорізька політехніка»

Факультет КНТ
Кафедра програмних засобів
Ступінь вищої освіти бакалавр
Спеціальність 122 Комп'ютерні науки
(код і найменування)

Освітня програма (спеціалізація) Комп'ютерні науки
(назва освітньої програми (спеціалізації))

ЗАТВЕРДЖУЮ

Завідувач кафедри ПЗ, д.т.н, проф.
Сергій СУББОТІН
“ ” 2025 року

З А В Д А Н Н Я

НА ДИПЛОМНИЙ ПРОЄКТ (РОБОТУ) СТУДЕНТА(КИ)

ХЛЕПІТЬКА Кирила Віталійовича

(ПРИЗВИЩЕ, ім'я, по батькові)

1. Тема проєкту (роботи) Програмне забезпечення інформаційної системи для керування нотатками користувача. Software for an Information System for Managing User Notes

керівник проєкту (роботи) к.т.н., доцент, КОЛПАКОВА Тетяна Олексіївна,
(науковий ступінь, вчене звання, ПРИЗВИЩЕ, ім'я, по батькові)

затверджені наказом закладу вищої освіти від “ 28 ” квітня 2025 року № 209

2. Строк подання студентом проєкту (роботи) 03 червня 2025 року

3. Вихідні дані до проєкту (роботи) рекомендована література, технічне завдання

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

5. Перелік графічного матеріалу (з точним зазначенням обов'язкових креслень, кількість слайдів, плакатів)

Слайди презентації

6. Консультанти розділів проєкту (роботи)

Розділ	ПРИЗВИЩЕ, ініціали та посада консультанта	Підпис, дата	
		завдання видав	прийняв виконане завдання
1-4 Основна частина	КОЛПАКОВА Т.О, доцент		
Нормоконтроль	КАМІНСЬКА Ж.К., асистент		

7. Дата видачі завдання “ 28 ” квітня 2025 року.

КАЛЕНДАРНИЙ ПЛАН

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

Студент(ка)

(підпис) Кирило ХЛЕПІТЬКО
(Ім'я ПРИЗВИЩЕ)

Керівник проєкту (роботи)

(підпис) Тетяна КОЛПАКОВА
(Ім'я ПРИЗВИЩЕ)

РЕФЕРАТ

Пояснювальна записка до дипломної кваліфікаційної роботи бакалавра:
85 с., 3 табл., 35 рис., 3 дод., 15 джерел.

C#, MICROSOFT VISUAL STUDIO, ВЕБЗАСТОСУНОК,
ІНФОРМАЦІЙНА СИСТЕМА, КЕРУВАННЯ НОТАТКАМИ, МОВА
ПРОГРАМУВАННЯ.

Об'єкт дослідження – процес розробки інформаційних систем для керування нотатками.

Предмет дослідження – інформаційні системи для керування нотатками.

Мета роботи – розробка програмного забезпечення інформаційної системи для керування нотатками користувача.

Матеріали, методи та технічні засоби: мова програмування C#, середовище розробки Microsoft Visual Studio.

Результати. Розроблено проєктні рішення для створення програмного забезпечення інформаційної системи для керування нотатками користувача. Створено програмне забезпечення інформаційної системи для керування нотатками користувача. Проведено тестування розробленого програмного забезпечення інформаційної системи для керування нотатками користувача.

Висновки. Мету роботи досягнуто. Розроблено програмне забезпечення інформаційної системи для керування нотатками користувача за допомогою мови програмування C# та середовища розробки Microsoft Visual Studio.

Галузь використання – керування нотатками.

ABSTRACT

Explanatory note to the diploma qualifying work of the bachelor: 85 pages, 3 tables, 35 figures, 3 appendixes, 15 sources.

C#, MICROSOFT VISUAL STUDIO, WEB APPLICATION, INFORMATION SYSTEM, NOTE MANAGEMENT, PROGRAMMING LANGUAGE.

The object of research is the process of developing information system for managing notes.

The subject of the research is information systems for managing notes.

The purpose of this work is to develop the software of the information system for managing notes.

Materials, methods and technical tools: C# programming language, Microsoft Visual Studio development environment.

Results. Project solutions for the creation of software for the information system for managing notes has been designed. The software of the information system for managing notes has been developed. The developed software of the information system for managing notes has been tested.

Conclusions. The goal of the work has been achieved. The software tools of information system for managing notes using the C# programming language and the Microsoft Visual Studio development environment have been developed.

Scope of use – note management.

ЗМІСТ

	С.
Перелік скорочень та умовних познач.....	8
Вступ.....	9
1 Аналіз предметної області	11
1.1 Інформаційні системи для керування нотатками	11
1.2 Аналіз інформаційних систем для керування нотатками	21
1.3 Висновки за розділом 1.....	30
2 Матеріали і методи	32
2.1 Вибір мови програмування.....	32
2.2 Вибір середовища розробки для створення програмного забезпечення інформаційної системи для керування нотатками	34
2.3 Висновки за розділом 2.....	36
3 Опис програми	38
3.1 Структура програмного забезпечення інформаційної системи для керування нотатками	38
3.2 Функціонування програмного забезпечення інформаційної системи для керування нотатками	40
3.3 Розробка бази даних програмного забезпечення інформаційної системи для керування нотатками.....	46
3.4 Проектування інтерфейсу програмного забезпечення інформаційної системи для керування нотатками	47
3.5 Висновки за розділом 3.....	50
4 Експлуатація, тестування та експериментальне дослідження програми.....	52
4.1 Призначення й умови застосування програми	52
4.2 Характеристики програмного забезпечення інформаційної системи для керування нотатками	53
4.3 Інструкція по експлуатації програми.....	54
4.3.1 Звернення до програми.....	54
4.3.2 Вхідні й вихідні дані.....	54

4.3.3 Повідомлення.....	54
4.4 Виконання програмного забезпечення інформаційної системи для керування нотатками	55
4.5 Тестування програмного забезпечення інформаційної системи для керування нотатками	58
4.6 Висновки за розділом 4.....	59
Висновки.....	60
Перелік джерел посилань.....	63
Додаток А Технічне завдання.....	65
Додаток Б Фрагмент тексту програми	69
Додаток В Слайди презентації	78

ПЕРЕЛІК СКОРОЧЕНЬ ТА УМОВНИХ ПОЗНАК

БД – база даних;

ІС – інформаційна система;

КН – керування нотатками;

ПЗ – програмне забезпечення.

ВСТУП

Актуальність розробки програмного забезпечення для керування нотатками користувача пояснюється сучасними потребами у впорядкуванні та збереженні великого обсягу інформації. У сучасному світі обсяг даних, з якими доводиться працювати, постійно зростає, і для їх ефективного управління потрібні інструменти, які забезпечують швидкий доступ, пошук та організацію [1]-[3].

Програмне забезпечення інформаційних систем для керування нотатками часто адаптовані до потреб користувачів, надаючи функції для категоризації, швидкого пошуку та автоматизації завдань. Використання штучного інтелекту додає нові можливості, зокрема аналіз нотаток, створення нагадувань чи автоматичне формування записів [4], [5].

Збільшення кількості мобільних пристроїв вимагає доступу до нотаток у будь-який час та з будь-якого місця. Хмарні програмні рішення дозволяють синхронізувати дані між різними пристроями, забезпечуючи безперервність роботи [6], [7].

Інформаційні системи для керування нотатками відповідають вимогам сучасного цифрового суспільства, сприяють підвищенню продуктивності, зручності та безпеки користувачів. Розробка подібного програмного забезпечення є актуальною як для особистого, так і для корпоративного використання [7], [8].

Проте деякі програмні засоби інформаційних систем для керування нотатками вимагають стабільного доступу до Інтернету для синхронізації даних, що може стати проблемою в умовах обмеженого або нестабільного підключення. Це особливо актуально для хмарних платформ, які без доступу до мережі втрачають значну частину функціональності. У деяких інформаційних системах для керування нотатками спостерігається надмірна складність інтерфейсу. Розширений набір функцій іноді може заплутати користувачів, які потребують простих і зрозумілих інструментів для роботи.

Деякі програмні засоби інформаційних систем для керування нотатками можуть бути обмежені у функціональності або вимагати оплати за повноцінний доступ до всіх функцій, що може стати бар'єром для окремих категорій користувачів [1]-[7].

Тому актуальною є розробка інформаційної системи для керування нотатками користувача. У дипломній кваліфікаційній роботі бакалавра розв'язується актуальне завдання розробки програмного забезпечення інформаційної системи для керування нотатками користувача.

Для досягнення поставленої мети у кваліфікаційній роботі бакалавра необхідно розв'язати такі задачі:

- виконати аналіз предметної області та інформаційних систем для керування нотатками;
- здійснити проектування програмного забезпечення інформаційної системи для керування нотатками користувача;
- створити програмне забезпечення інформаційної системи для керування нотатками користувача;
- виконати тестування розробленого програмного забезпечення інформаційної системи для керування нотатками користувача.

1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ

1.1 Інформаційні системи для керування нотатками

Керування нотатками – це процес організації, створення, редагування, збереження та доступу до текстової, графічної чи іншої інформації у формі нотаток. Це може бути реалізовано за допомогою спеціалізованих інструментів або програмного забезпечення, які забезпечують зручність зберігання, пошуку, упорядкування та синхронізації нотаток між пристроями [1].

Основні аспекти керування нотатками включають: створення нотаток, організацію нотаток, їх редагування, збереження та пошук. Також важливими є синхронізація, тобто можливість автоматичного оновлення нотаток на різних пристроях, та захист даних, зокрема, використання паролів або шифрування для забезпечення конфіденційності [1], [2].

Інформаційні системи для керування нотатками являють собою програмні комплекси або платформи, створені для організації, обробки, збереження та обміну текстовою чи мультимедійною інформацією у вигляді нотаток. Вони забезпечують користувачам зручний інструмент для запису та управління даними в різних контекстах: особистому, навчальному чи професійному [1], [2].

Керування нотатками є важливим аспектом в організації роботи з інформацією в різних сферах, таких як навчання, професійна діяльність або творчий процес. В умовах постійного надходження нової інформації, легкість у записуванні та зберіганні важливих ідей стає критичною. Однак, незважаючи на доступність інструментів для записів, важливо правильно організувати їх, аби інформація не стала хаотичною та безладною [1], [2].

Нотатки можуть бути створені в різних форматах, залежно від переваг користувача. І хоча сам процес запису є індивідуальним, важливо знати, як впорядковувати записану інформацію, щоб згодом її можна було легко знайти.

Погано організовані записи можуть призвести до втрати часу та нервів, коли необхідно повернутися до важливих матеріалів, а їх не можна знайти [1], [2].

Для ефективного керування нотатками важливими є чітка структура та зрозуміла система організації. Добре впорядковані нотатки полегшують пошук потрібної інформації, прискорюють процес повторного перегляду та дозволяють швидше виявляти зв'язки між ідеями. Крім того, правильне впорядкування нотаток допомагає знижувати стрес та підвищує ефективність роботи з матеріалом [1], [2].

Інформаційні системи керування нотатками можуть бути представлені різноманітними інструментами, такими як спеціалізовані програми, вебплатформи та інші програмні рішення. Вибір інструменту залежить від конкретних потреб користувача, його стилю роботи та вимог до функціональності. Деякі користувачі обирають цифрові рішення, інші ж віддають перевагу традиційним методам запису на папері, що дозволяє знайти підхід, найбільш відповідний індивідуальним уподобанням. Пошук найбільш ефективного способу організації записів, відповідного до стилю навчання чи роботи, може значно полегшити доступ до необхідної інформації в майбутньому [1], [2].

Важливими є також регулярність і структура нотаток, адже це допомагає зберігати порядок і робить пошук інформації більш ефективним. Використання спеціальних програмних засобів та візуальних елементів, таких як заголовки, списки та маркери, дає можливість швидше орієнтуватися в матеріалі [1], [2].

Важливою є також читабельність нотаток, адже якщо записи незрозумілі, їх буде важко використовувати в майбутньому. Чіткість і розбірливість дозволяють зберегти інформацію, яка буде корисною на довший період часу. Крім того, при розробці інформаційної системи для керування нотатками варто передбачити можливість додавати основні дані, такі як дати або тематичні позначки, щоб значно зменшити час пошуку потрібної інформації [1], [2].

Після того, як вирішено, які нотатки та інформацію необхідно збирати, важливо створити ефективну систему їх класифікації, яка буде використовуватися у розроблюваній інформаційній системі. Це полегшує подальше зберігання та пошук необхідних даних. Категорії можуть бути різними в залежності від сфери діяльності, наприклад, для навчання це можуть бути курси, лекції або завдання. Важливо визначити підкатегорії, щоб організувати дані ще більш детально [1], [2].

Організація нотаток вимагає вибору методів, що підходять для особистих потреб. Для паперових записів це можуть бути спеціальні зошити, підшивки або використання кольорових роздільників для різних категорій. Для цифрових нотаток надається широкий вибір програм, які дозволяють структуровано зберігати записи, використовувати теги та функції пошуку. При роботі з текстовими редакторами, корисно створювати окремі папки для кожної категорії та дотримуватися чіткої системи іменування файлів [1], [2].

Ключовим є створення такої інформаційної системи для керування нотатками, яка буде зручною, швидкою та інтуїтивно зрозумілою, що дозволить регулярно користуватися нею без зайвих труднощів [1], [2].

Організація нотаток за допомогою інформаційних систем для керування нотатками є важливим елементом для забезпечення їх доступності та зручності використання. Створення змісту в таких системах дозволяє користувачам швидко знаходити необхідну інформацію, особливо коли нотатки містять великий обсяг даних або складні концепції. Завдяки функціям пошуку та автоматичним оновленням, впровадженим у сучасних інформаційних системах, зміст може бути інтерактивним, дозволяючи безпосередньо переходити до потрібних розділів, що спрощує навігацію та збереження загального огляду [1], [2].

Цифрові інформаційні системи для управління нотатками мають додаткові функції для інтеграції змісту, що дозволяє автоматично оновлювати розділи та забезпечувати зручне користування на різних платформах. Використання таких програм дозволяє не лише організовувати, але й

ефективно обробляти нотатки, забезпечуючи їх збереження та доступність у будь-який час [1], [2].

Для тих, хто працює з великим обсягом інформації або в команді, ефективно розроблені інформаційні системи для керування нотатками можуть включати функції спільного доступу, дозволяючи користувачам ділитися даними або працювати над ними разом. Такі системи забезпечують зручний доступ до необхідних матеріалів і дозволяють скоротити час на пошук або обробку важливої інформації [1], [2].

Важливою складовою ефективного використання системи для керування нотатками є її регулярне оновлення та перегляд. Враховуючи, що потреби користувача з часом можуть змінюватися, система повинна бути гнучкою і здатною адаптуватися до нових умов. Під час регулярного перегляду слід оновлювати категорії та структуру файлів, щоб забезпечити легкий доступ до актуальної інформації. Це дозволяє підтримувати ефективність системи в довгостроковій перспективі [1], [2].

Перегляд нотаток також дозволяє коригувати існуючу структуру та усувати невідповідності, що виникають через зміни в роботі або навчанні. Створення і підтримка такої організованої інформаційної системи для керування нотатками не лише полегшує пошук необхідних даних, а й забезпечує зручне користування та збільшує продуктивність, оскільки зберігається порядок і доступність інформації в будь-який час [1], [2].

Систематизація нотаток є важливою складовою ефективною інформаційної системи керування нотатками. Завдяки правильно організованій системі збереження і структуризації даних, яка використовується в інформаційних системах для керування нотатками, можна значно покращити процес запам'ятовування та зручність перегляду матеріалів. У таких системах можна не лише впорядковувати нотатки, а й оптимізувати процеси їх пошуку, що сприяє економії часу та зниженню рівня стресу. Належне впорядкування допомагає не лише в навчанні, але й підвищує ефективність роботи, зокрема для тих, хто постійно працює з великим обсягом інформації [3], [4].

Інформаційні системи для керування нотатками відіграють ключову роль у організації знань, покращуючи процес навчання та дозволяючи зосередитися на головному. Такі системи допомагають користувачам структурно підходити до зберігання та обробки даних, що значно спрощує доступ до необхідної інформації, зокрема під час підготовки до тестів чи важливих зустрічей. Регулярний перегляд і організація нотаток у системах дозволяють навчатися та працювати з максимальним ефектом, підвищуючи рівень успішності та зменшуючи час, витрачений на пошук матеріалів [3], [4].

Вміння правильно організовувати нотатки та ефективно їх управляти через спеціалізовані інформаційні системи стає основою для активного залучення до процесу навчання чи роботи. Завдяки такій організації користувачі мають можливість зберігати пильність та зосередженість, що особливо важливо під час лекцій або важливих робочих зустрічей. Ведення нотаток у таких інформаційних системах стимулює активну участь та ефективну обробку інформації, дозволяючи уникати відволікань та підтримувати увагу на ключових моментах [3], [4].

Вибір підходящого методу організації нотаток є ключовим для створення ефективних інформаційних систем для керування нотатками. Різні стратегії зберігання та обробки даних можуть задовольняти різні стилі навчання та вимоги до конкретних дисциплін. У таких системах важливо враховувати особливості кожного методу для забезпечення ефективної організації та доступу до необхідної інформації [3], [4].

Контурний метод є одним із варіантів організації нотаток, який дозволяє створювати чітку ієрархічну структуру для важливих тем і підпунктів. Це дозволяє створювати добре організовані документи, зручні для подальшого перегляду та підготовки до іспитів. Використання цього методу в інформаційних системах дозволяє зберігати структуровані дані і швидко знаходити необхідну інформацію. Однак для лекцій, що включають графіки та діаграми, цей метод може бути менш ефективним [3], [4].

Метод Корнела передбачає спеціальний формат для структуризації інформації, що полегшує перегляд і узагальнення ключових моментів. В інформаційних системах цей метод може бути адаптований через розділення нотаток на кілька блоків, що дає змогу зберігати нотатки в компактному вигляді та полегшує процес їхнього аналізу. Проте для лекцій, що містять складну термінологію та числові дані, цей метод може виявитися менш зручним [3], [4].

Інтелектуальні карти дозволяють візуально відображати зв'язки між різними аспектами теми, що є корисним для глибшого розуміння матеріалу. Інформаційні системи для керування нотатками можуть підтримувати створення таких карт, що допомагає користувачам візуалізувати та систематизувати складні ідеї та їхні підкатегорії. Цей метод є ефективним для тих, хто має візуальний стиль навчання, але він може бути складним у застосуванні під час швидких лекцій [3], [4].

Метод потоку дозволяє здійснювати більш динамічне та безперервне конспектування, використовуючи символи, кольори та стрілки для відображення зв'язків між ідеями. В інформаційних системах цей метод можна використовувати для активної взаємодії з матеріалом, що сприяє більш творчому підходу до навчання. Це дозволяє зберігати гнучкість у процесі запису, хоча для деяких типів інформації цей метод може бути недостатньо структурованим [3], [4].

Метод боксу, завдяки використанню цифрових інструментів для класифікації інформації, дозволяє легко організовувати нотатки в окремі блоки, кожен з яких відображає певну тему. Такі системи можуть бути особливо корисними для візуальних учнів, оскільки вони дозволяють створювати зручні та естетично привабливі документи. Однак цей метод вимагає значної уваги до деталей і може бути менш ефективним для запису швидких лекцій [3], [4].

Метод побудови діаграм сприяє організації нотаток у вигляді діаграм, що полегшує візуалізацію зв'язків між ідеями. Це дозволяє краще

запам'ятовувати факти та дані, що є важливими для багатьох дисциплін. В інформаційних системах можна використовувати цей метод для створення зручних для перегляду та ефективних діаграм, хоча для лекцій без чіткої структури цей метод може бути не таким зручним [3], [4].

Кожен з методів має свої переваги та обмеження, і їх застосування в інформаційних системах для керування нотатками дає можливість вибрати найефективніший підхід для конкретного стилю навчання та предмета. Експериментуючи з різними методами, можна знайти найбільш оптимальну стратегію, яка найкраще відповідає індивідуальним вимогам [3], [4].

Інтелектуальні карти є важливим інструментом для організації та збору нотаток у рамках інформаційних систем для керування нотатками. Вони суттєво відрізняються від традиційних методів запису і дозволяють візуалізувати думки та ідеї, що сприяє покращенню мозкового штурму. Такий підхід дає можливість створювати ефективні стратегії для обробки інформації, використовуючи зв'язки між ідеями через кольори, лінії та зображення [3], [4].

У розробці інформаційних систем для нотаток важливим аспектом є впровадження системи нумерації. Це дозволяє користувачам швидко знаходити необхідні нотатки, що значно підвищує ефективність навчання та роботи. Гарно структурована нумерація забезпечує чітке відстеження інформації, допомагаючи організувати великий обсяг даних для подальшого використання [3], [4].

Використання тематичних категорій в інформаційних системах для нотаток є ефективним способом організації матеріалу. Така організація допомагає не лише зберігати інформацію в порядку, а й спрощує процес вивчення та запам'ятовування, оскільки дозволяє швидко знаходити потрібну інформацію. Користувачі можуть створювати окремі розділи для різних тем чи проектів, що дає змогу організувати нотатки у рамках великих навчальних чи робочих задач [3], [4].

Тематичні блокноти в інформаційних системах сприяють зручному доступу до нотаток, зменшуючи час на пошук і підвищуючи продуктивність.

Такий підхід дозволяє організувати нотатки не лише за основними темами, але й за підтемами, що забезпечує ще більш детальну категоризацію та дає можливість швидко знаходити необхідні відомості [3], [4].

У контексті інформаційних систем керування нотатками особливе значення має створення показників для зручного доступу до інформації. Це дозволяє користувачам швидко орієнтуватися в нотатках, що значно покращує ефективність навчання та роботи. Показник, який організовує матеріал за основними та підтематичними категоріями, допомагає зекономити час на пошук і надає швидкий доступ до потрібних даних [3], [4].

Перехресні посилання, що використовуються в системах для керування нотатками, є потужним інструментом для інтеграції інформації. Вони допомагають зв'язувати між собою різні частини матеріалу, що значно покращує процес навчання та розуміння складних тем. Візуалізація таких зв'язків дає можливість побачити, як різні ідеї взаємопов'язані, що сприяє глибшому засвоєнню матеріалу та підвищенню ефективності навчальних процесів. Це також полегшує інтегроване вивчення, яке особливо важливе для складних предметів, що вимагають комплексного підходу [3], [4].

Використання підшивок як засобу впорядкування нотаток у відповідних інформаційних системах дає змогу легко структурувати матеріали, забезпечуючи зручність доступу до потрібної інформації. Завдяки можливості перегруповувати сторінки та додавати нові, підшивки є ефективним інструментом для тих, хто постійно працює з великою кількістю документів. Водночас, через свої габарити, вони не завжди зручні для транспортування, що може стати викликом у деяких ситуаціях. Цей підхід особливо виграє від використання сучасних інформаційних систем для керування нотатками, які дозволяють автоматизувати впорядкування та пошук даних [3], [4].

Інформаційні системи для організації нотаток, що підтримують візуалізацію та категоризацію, значно полегшують роботу з матеріалами. Завдяки функціям, які імітують розділювачі вкладок та кольорове кодування,

такі системи створюють цифровий аналог класичних підшивок. Додавання тегів, автоматичне сортування за темами та вбудований пошук дозволяють зосередитися на роботі, не витрачаючи час на ручне впорядкування. Періодична актуалізація даних та видалення застарілої інформації підвищують ефективність використання систем [5], [6].

Розробка інформаційних систем для керування нотатками включає можливість створення резюме та підсумків на основі текстових документів. Це дає змогу автоматично виділяти ключові положення з лекцій, зустрічей або досліджень, оптимізуючи підготовку до іспитів чи завдань. Такі системи інтегрують алгоритми семантичного аналізу, які розбивають текст на розділи, визначають основні ідеї та створюють скорочені огляди. Завдяки цьому користувачі отримують персоналізований навчальний матеріал, адаптований до їхніх потреб [5], [6].

Ефективна функція створення цифрових карток також входить до складу багатьох інформаційних систем управління нотатками. Ці інструменти дозволяють створювати інтерактивні картки для запам'ятовування важливих фактів чи понять, використовуючи двосторонній підхід. Застосування візуалізації, мнемонічних прийомів і автоматичного повторення підвищує ефективність навчання. Спільний доступ до таких карток між користувачами сприяє обміну знаннями та розвитку колективного інтелекту [7], [8].

Інтеграція інформаційних систем у процес навчання та роботи дозволяє перейти від паперових підшивок до цифрових платформ, які забезпечують багаторівневу організацію матеріалів. Вбудовані покажчики, посилання між нотатками та розширені функції пошуку сприяють швидкому доступу до потрібної інформації. Це особливо корисно для складних проектів чи тем, що вимагають аналізу великої кількості взаємопов'язаної інформації. У результаті такі системи створюють інтелектуальне середовище, яке полегшує навчання, організацію робочого процесу та збереження важливих даних [7], [8].

Інформаційні системи забезпечують гнучкість у створенні нотаток, дозволяючи структурувати їх у вигляді розділів, тем або концептуальних карт.

Вони також спрощують виділення ключових моментів і зведення інформації до суттєвих деталей. Використання цифрових інструментів з інтерактивними можливостями, такими як теги, індексація та перехресні посилання, допомагає зробити роботу з навчальними матеріалами більш продуктивною. Крім того, автоматизовані функції аналізу тексту можуть виділяти важливі концепції для підготовки до тестів чи практичних завдань [7], [8].

Інформаційні системи для керування нотатками дозволяють трансформувати нотатки зі статичних записів у активний навчальний інструмент, який допомагає запам'ятовувати та відтворювати інформацію. У таких системах також реалізовано можливості спільного доступу до нотаток, що сприяє груповому навчанню та обміну ідеями. Завдяки зручним функціям колективної роботи, як-от спільне редагування або коментування, інформаційні системи для керування нотатками забезпечують ефективну комунікацію між користувачами [7], [8].

Використання інформаційних систем для роботи з нотатками значно підвищує ефективність навчання та роботи. Такі інформаційні системи дозволяють створювати структуровані, адаптовані до потреб користувачів ресурси, які полегшують пошук, організацію та зберігання інформації. Завдяки таким технологіям підхід до організації знань стає сучасним, інтуїтивно зрозумілим і зручним, сприяючи розвитку продуктивності та інновацій у навчальному процесі [7], [8].

Визначено, що керування нотатками є процесом організації, створення, редагування, збереження та доступу до текстової, графічної чи іншої інформації у формі нотаток. Це може бути реалізовано за допомогою спеціалізованих інструментів або програмного забезпечення, які забезпечують зручність зберігання, пошуку, упорядкування та синхронізації нотаток між пристроями. Інформаційні системи для керування нотатками являють собою програмні комплекси або платформи, створені для організації, обробки, збереження та обміну текстовою інформацією у вигляді нотаток.

1.2 Аналіз інформаційних систем для керування нотатками

Проаналізуємо інформаційні системи для керування нотатками [5]-[11].

Програмне забезпечення Notion (рис. 1.1) являє собою багатофункціональну інформаційну систему, яка створена для організації та управління нотатками користувача, яка також охоплює завдання, проекти, записи й інші аспекти робочого процесу. Ця інформаційна система дозволяє об'єднувати всі інструменти для роботи з інформацією в одному цифровому просторі, забезпечуючи зручність, ефективність і гнучкість у керуванні особистими та командними даними [5], [6], [9].

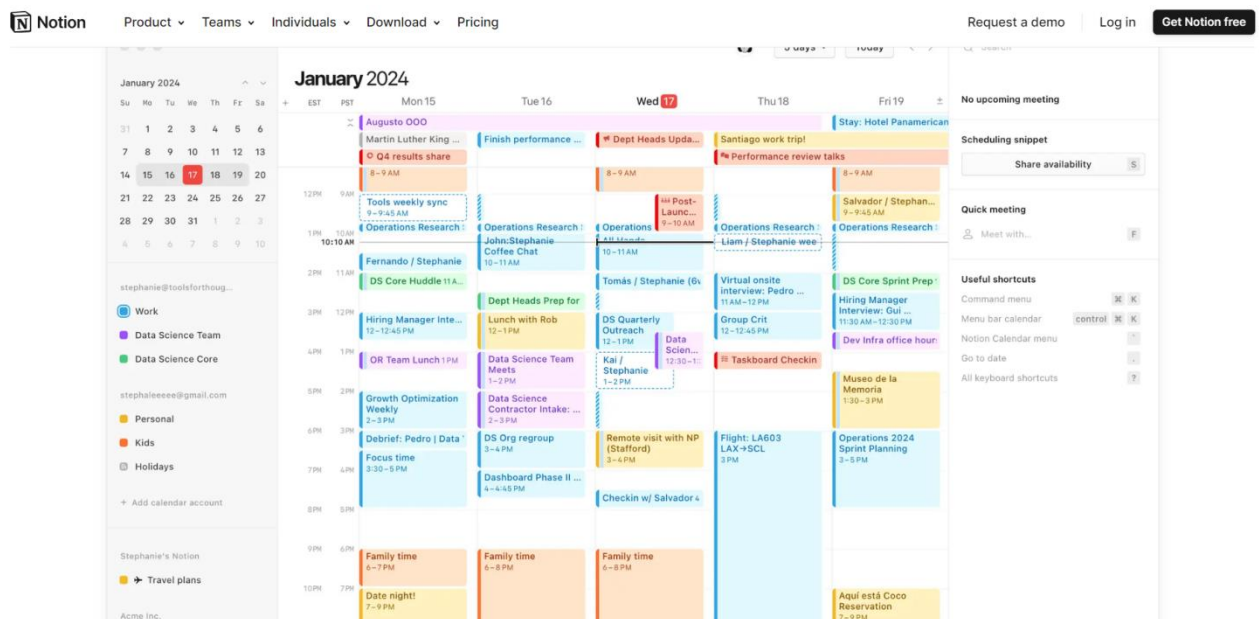


Рисунок 1.1 – Програмне забезпечення інформаційної системи Notion [9]

Основна увага в Notion зосереджена на можливості керувати нотатками, які можуть бути адаптовані під індивідуальні потреби користувача або вимоги команди. Інтерфейс сприяє створенню структурованих записів, каталогів, баз даних і шаблонів, які допомагають користувачам легко впорядковувати інформацію. Система підтримує налаштування для різних типів робочих процесів, дозволяючи створювати персоналізовані рішення для організації нотаток, проєктів, календарів і завдань [5], [6], [9].

Додатково Notion інтегрує можливості штучного інтелекту, який безпосередньо працює у всіх нотатках і документах, розширюючи їхню функціональність. Завдяки цьому стає можливим автоматизувати деякі процеси, прискорити створення контенту та аналіз даних. Використання штучного інтелекту усуває необхідність перемикатися між різними інструментами, надаючи всі необхідні функції в одному місці [5], [6], [9].

Система забезпечує зручність у спільній роботі, дозволяючи обмінюватися нотатками, створювати спільні бази даних, оновлювати документи та організовувати зустрічі. Завдяки доступу з будь-якого пристрою через мобільні додатки або вебверсію, користувачі можуть керувати своїми нотатками з будь-якого місця [5], [6], [9].

Notion надає функції для управління базами даних, документацією та експортом інформації, що дозволяє інтегрувати різноманітні робочі процеси. Додаток також підтримує інтеграцію з іншими сервісами, такими як Figma чи Google Sheets, що розширює його можливості для зручної роботи з нотатками та проектами [5], [6], [9].

Цей інструмент призначений для створення зручного та інтуїтивного середовища, яке допомагає оптимізувати процеси, спрощувати управління інформацією та забезпечувати ефективну організацію нотаток для різних потреб користувача [5], [6], [9].

Основними характеристиками програмного забезпечення інформаційної системи Notion є такі [5], [6], [9]:

- гнучкість у створенні та організації нотаток. Програмне забезпечення Notion дозволяє користувачам створювати й налаштовувати нотатки у вигляді тексту, таблиць, баз даних, списків, чек-листів або сторінок із вкладеними елементами. Система підтримує ієрархічну структуру, що полегшує впорядкування великих обсягів інформації [5], [6], [9];

- інтеграція даних і робочих процесів. Програмне забезпечення Notion надає можливість об'єднувати завдання, календарі, дорожні карти проєктів, документи та нотатки в одному місці. Інтеграція з іншими сервісами, такими

як Google Drive, Figma, Slack, забезпечує безперебійну роботу з зовнішніми даними [5], [6], [9];

- можливості спільної роботи. Програмне забезпечення Notion створено для командного використання, де користувачі можуть разом редагувати нотатки, обмінюватися коментарями та керувати проектами. Гнучка система доступів дозволяє налаштовувати рівні прав для окремих користувачів [5], [6], [9];

- підтримка баз даних. Програмне забезпечення інформаційної системи Notion забезпечує створення складних баз даних із різними типами записів, такими як текст, посилання, зображення або файли. Користувачі можуть створювати фільтри, сортування, а також переглядати дані у формі списків, таблиць або календарів [5], [6], [9];

- автоматизація з використанням штучного інтелекту. Інтеграція інструментів штучного інтелекту дозволяє автоматизувати частину роботи з нотатками, такі як створення текстів, аналіз даних або пропонування підказок. Це розширює можливості управління інформацією та спрощує виконання завдань [5], [6], [9];

- кросплатформеність. Доступ до програмного забезпечення Notion можливий із будь-якого пристрою — через веббраузери, мобільні додатки для iOS і Android, а також звичайні програми. Це забезпечує зручність у роботі незалежно від локації користувача [5], [6], [9];

- зручний інтерфейс. Програмне забезпечення Notion має мінімалістичний дизайн, що сприяє фокусуванню на вмісті без зайвих відволікань. Візуальна організація нотаток і баз даних робить інструмент інтуїтивно зрозумілим навіть для нових користувачів [5], [6], [9];

- підтримка шаблонів. Для швидкого початку роботи з програмним забезпеченням інформаційної системи Notion можна використовувати готові шаблони для різних цілей: від особистих списків завдань до складних дорожніх карт продуктів і баз знань [5], [6], [9];

– експорт і збереження даних. Програмне забезпечення Notion дозволяє експортувати створені дані у формати, зручні для архівування або подальшого використання, зокрема PDF, HTML чи CSV [5], [6], [9];

– особисте й корпоративне використання. Програмне забезпечення інформаційної системи Notion однаково добре підходить як для індивідуальних користувачів, що шукають ефективний спосіб організації нотаток і завдань, так і для компаній, які потребують інтегрованого середовища для співпраці та управління даними [5], [6], [9].

Серед недоліків програмного забезпечення Notion можна відзначити певні обмеження в не онлайн режимі, оскільки доступ до деяких функцій може бути обмежений без підключення до Інтернету. Також варто зазначити, що хоча інтерфейс програмного забезпечення Notion є зручним, для нових користувачів може знадобитися час на освоєння всіх можливостей, зокрема при роботі з базами даних і налаштуваннями інтеграцій. Ще одним недоліком є відсутність деяких функцій для роботи з великими обсягами даних, що може ускладнити використання для користувачів із великими проєктами або складними робочими процесами. Програмне забезпечення Notion також іноді має проблеми з продуктивністю при роботі з великою кількістю елементів або високим рівнем кастомізації [5], [6], [9].

Програмне забезпечення Simplenote (рис. 1.2) пропонує простий і ефективний спосіб створювати та організовувати нотатки без зайвих складнощів. Відмінною рисою цього інструменту є мінімалістичний підхід, що дозволяє користувачам зосередитись на основному — запису та управлінні нотатками. Програмне забезпечення забезпечує синхронізацію нотаток у реальному часі на всіх пристроях, завдяки чому вони завжди залишаються доступними незалежно від того, який пристрій використовується – мобільний телефон, планшет чи комп'ютер. Програмне забезпечення Simplenote дає можливість швидко і зручно створювати нотатки, списки або зберігати ідеї. Користувачі можуть додавати теги до своїх нотаток, що допомагає організувати їх за темами або проєктами [6], [7], [10].

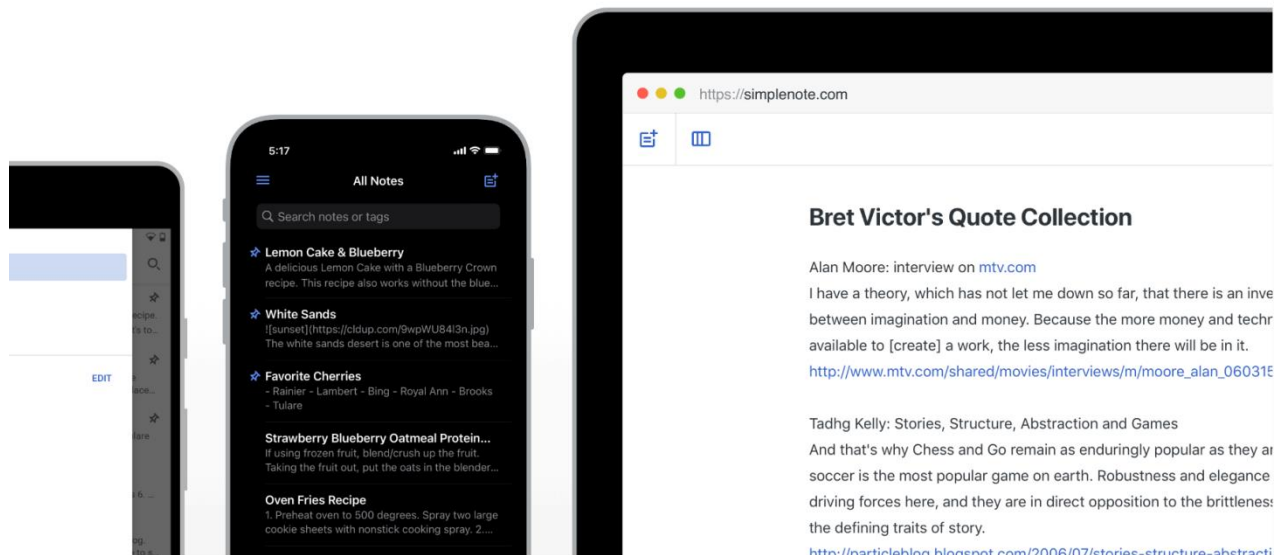


Рисунок 1.2 – Програмне забезпечення інформаційної системи Simplenote [10]

Програмне забезпечення Simplenote підтримує швидкий пошук і доступ до необхідної інформації. Крім того, Simplenote підтримує функцію спільної роботи, що дозволяє кільком користувачам працювати над одними нотатками в реальному часі, що робить платформу ідеальним інструментом для командної діяльності. Усі ці функції дозволяють ефективно керувати нотатками та організовувати робочі процеси, не перевантажуючи користувача зайвими можливостями [6], [7], [10].

Завдяки підтримці розмітки, користувачі можуть створювати більш складні та багатофункціональні нотатки, що додає гнучкості у використанні цього інструменту. Інтуїтивно зрозумілий інтерфейс програмного забезпечення Simplenote дозволяє навіть новачкам швидко освоїти програму, а синхронізація в реальному часі забезпечує безперервний доступ до нотаток. З цією системою можна легко впорядковувати і зберігати всі свої думки та ідеї, що робить її ідеальним інструментом для персонального використання або командної роботи [6], [7], [10].

Програмне забезпечення інформаційної системи Simplenote також має можливість інтеграції з іншими інструментами через Zapier, що дозволяє розширити функціональність програми і використовувати її в більш складних робочих процесах [6], [7], [10].

Програмне забезпечення Simplenote має такі особливості [6], [7], [10]:

- зручний інтерфейс: програмне забезпечення Simplenote надає Спрощений дизайн без зайвих елементів дозволяє користувачам зосередитись на процесі створення нотаток [6], [7], [10];

- синхронізація в реальному часі: нотатки автоматично синхронізуються між всіма пристроями користувача, забезпечуючи доступ до даних з будь-якого місця [6], [7], [10];

- підтримка тегів: користувачі програмного забезпечення Simplenote можуть додавати теги до нотаток, що допомагає ефективно організовувати інформацію та здійснювати пошук [6], [7], [10];

- швидкий пошук: завдяки потужному інструменту пошуку програмного забезпечення Simplenote можна швидко знаходити конкретні нотатки навіть серед великої кількості записів [6], [7], [10];

- спільна робота: програмне забезпечення Simplenote дозволяє кільком користувачам одночасно працювати над нотатками в реальному часі, що підходить для командної роботи [6], [7], [10];

- інтеграція через Zapier: програмне забезпечення Simplenote можна інтегрувати з іншими програмами за допомогою Zapier для автоматизації робочих процесів [6], [7], [10];

- безкоштовне використання: програмне забезпечення Simplenote доступне безкоштовно, що робить його привабливим варіантом для користувачів, які шукають просте рішення для ведення нотаток [6], [7], [10];

- підтримка різних платформ: програмне забезпечення Simplenote можна використовувати на пристроях iOS, Android, а також на десктопах через веб-браузер [6], [7], [10].

Серед недоліків програмного забезпечення Simplenote можна відзначити обмежену кількість розширених функцій порівняно з іншими програмами для нотаток. Наприклад, відсутність можливості вставляти зображення чи медіафайли у нотатки може бути незручним для користувачів, яким потрібно працювати з різними типами контенту. Крім того, інтерфейс

програми може виглядати застарілим порівняно з більш сучасними альтернативами. Це може бути проблемою для користувачів, які шукають більш функціональні або інтуїтивно зрозумілі рішення. Відсутність підтримки складних форматів тексту, таких як таблиці або складні елементи форматування, також може обмежувати застосування Simplenote для більш професійних завдань [6], [7], [10].

Програмне забезпечення Evernote (рис. 1.3) є інструментом, який спеціально розроблений для спрощення процесу створення та організації нотаток користувача. Завдяки інтеграції інтуїтивно зрозумілих функцій для створення візуальних нотаток, зокрема скріншотів та ескізів, користувачі можуть швидко додавати важливі деталі до своїх заміток. Ця система дозволяє ефективно коментувати й організовувати інформацію, а також полегшує її обмін між учасниками проекту чи командою [5], [6], [11].

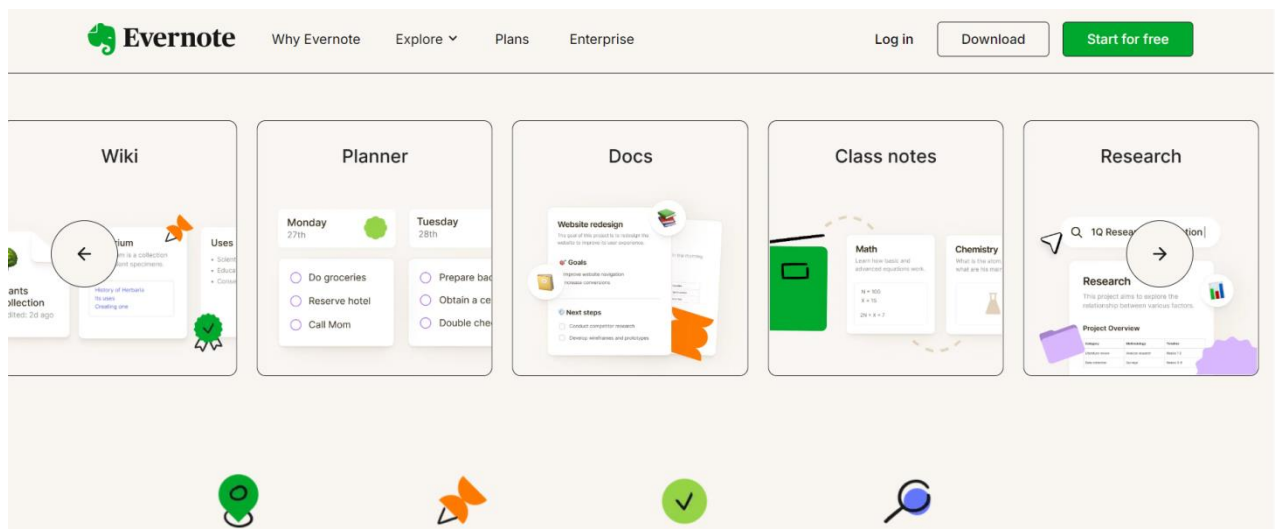


Рисунок 1.3 – Програмне забезпечення інформаційної системи Evernote [11]

Однією з основних переваг програмного забезпечення інформаційної системи Evernote є можливість використовувати хмарну синхронізацію, що забезпечує доступ до нотаток на різних пристроях у реальному часі. Користувачі можуть додавати текст, зображення, фігури та інші елементи, що допомагає зробити інформацію більш зрозумілою та візуально привабливою.

Інструменти для анотацій, які є частиною Evernote, дозволяють створювати докладні замітки та коментарі, що корисно для обговорень і планування робочих процесів [5], [6], [11].

Завдяки безперешкодній інтеграції з іншими програмами, програмне забезпечення інформаційної системи Evernote забезпечує зручний спосіб для команд співпрацювати й швидко обмінюватися ідеями. Це програмне забезпечення допомагає користувачам покращити організацію робочих процесів, підвищити рівень креативності та загальну ефективність завдяки оптимізації роботи з нотатками [5], [6], [11].

Програмне забезпечення Evernote має такі особливості [5], [6], [11]:

- інструменти для створення візуальних нотаток: програмне забезпечення Evernote дозволяє створювати копії екранних форм, ескізи та додавати анотації, що допомагає візуалізувати ідеї та процеси [5], [6], [11];

- простота в користуванні: завдяки інтуїтивно зрозумілому інтерфейсу, програмне забезпечення Evernote легко засвоюється навіть новими користувачами [5], [6], [11];

- хмарна синхронізація: нотатки синхронізуються в реальному часі на різних пристроях, що забезпечує доступ до них будь-де [5], [6], [11];

- можливість додавати різноманітні елементи: користувачі програмного забезпечення Evernote можуть вставляти зображення, текст, фігури та інші елементи в нотатки для більшої виразності [5], [6], [11];

- підтримка спільної роботи: програмне забезпечення Evernote дає змогу користувачам обмінюватися своїми нотатками та коментарями з іншими учасниками, що особливо корисно для команд [5], [6], [11];

- інтеграція з іншими інструментами: програмне забезпечення Evernote може працювати разом з іншими програмами для підвищення продуктивності та зручності роботи [5], [6], [11];

- простота в організації: користувачі програмного забезпечення Evernote можуть легко організовувати свої нотатки, що допомагає зберігати порядок у великих обсягах інформації [5], [6], [11];

– покращення креативності та продуктивності: Evernote сприяє покращенню процесів мозкового штурму, планування проектів та обміну ідеями, що підвищує загальну ефективність роботи [5], [6], [11].

Серед недоліків програмного забезпечення Evernote можна відзначити обмежену функціональність для створення текстових нотаток, оскільки основна увага приділяється візуальним елементам. Це може бути проблемою для користувачів, яким потрібно працювати з великими обсягами тексту або створювати складніші документи. Також, хоча програма підтримує хмарну синхронізацію, вона може мати обмеження щодо доступу до більш глибоких функцій у безкоштовній версії. Інтерфейс програми інколи може виглядати застарілим порівняно з іншими сучасними інструментами для створення нотаток, що також може впливати на загальний користувацький досвід. Крім того, Evernote не підтримує великої кількості форматів файлів, що обмежує варіанти використання в деяких робочих процесах [5], [6], [11].

Порівняльну характеристику інформаційних систем для керування нотатками наведено у таблиці 1.1.

Таблиця 1.1 – Порівняння інформаційних систем для керування нотатками

Критерій порівняння	Notion [9]	Simplenote [10]	Evernote [11]
Підтримка багатофункціональних нотаток	+	—	+
Простота використання	+—	+	+—
Безкоштовний план	+—	+	+—
Підтримка мультимедійних нотаток	+	—	+
Засоби керування проєктами	+	—	+—
Інтеграція з іншими програмами	+	+—	+

За результатами проведеного аналізу можна зробити висновок, що у наш час існує досить багато інформаційних систем для керування нотатками. Проте деякі інформаційні системи для керування нотатками вимагають стабільного доступу до Інтернету для синхронізації даних, що може стати проблемою в умовах обмеженого або нестабільного підключення. Це особливо актуально для хмарних платформ, які без доступу до мережі втрачають значну частину функціональності. У деяких інформаційних системах для керування нотатками спостерігається надмірна складність інтерфейсу. Розширений набір функцій іноді може заплутати користувачів, які потребують простих і зрозумілих інструментів для роботи. Деякі програмні засоби інформаційних систем для керування нотатками можуть бути обмежені у функціональності або вимагати оплати за повноцінний доступ до всіх функцій, що може стати бар'єром для окремих категорій користувачів. Тому актуальною є розробка програмного забезпечення інформаційної системи для керування нотатками.

При розробці програмного забезпечення інформаційної системи для керування нотатками необхідно забезпечити такі функціональні вимоги:

- підтримка можливості реєстрації та авторизації користувачів;
- підтримка можливості відображення списку нотаток користувача;
- можливість переходу між сторінками програми за допомогою відповідних кнопок;
- можливість додавання нотаток користувача;
- підтримка можливості ознайомлення з нотатками інших користувачів програми.

1.3 Висновки за розділом 1

Визначено, що керування нотатками є процесом організації, створення, редагування, збереження та доступу до текстової, графічної чи іншої інформації у формі нотаток. Це може бути реалізовано за допомогою спеціалізованих інструментів або програмного забезпечення, які забезпечують

зручність зберігання, пошуку, упорядкування та синхронізації нотаток між пристроями. Інформаційні системи для керування нотатками являють собою програмні комплекси або платформи, створені для організації, обробки, збереження та обміну текстовою інформацією у вигляді нотаток.

За результатами проведеного аналізу зроблено висновок, що у наш час існує досить багато інформаційних систем для керування нотатками. Проте деякі інформаційні системи для керування нотатками вимагають стабільного доступу до Інтернету для синхронізації даних, що може стати проблемою в умовах обмеженого або нестабільного підключення. Це особливо актуально для хмарних платформ, які без доступу до мережі втрачають значну частину функціональності. У деяких інформаційних системах для керування нотатками спостерігається надмірна складність інтерфейсу. Розширений набір функцій іноді може заплутати користувачів, які потребують простих і зрозумілих інструментів для роботи. Деякі програмні засоби інформаційних систем для керування нотатками можуть бути обмежені у функціональності або вимагати оплати за повноцінний доступ до всіх функцій, що може стати бар'єром для окремих категорій користувачів. Тому актуальною є розробка програмного забезпечення інформаційної системи для керування нотатками.

Сформульовано функціональні вимоги до програмного забезпечення інформаційної системи для керування нотатками.

2 МАТЕРІАЛИ І МЕТОДИ

2.1 Вибір мови програмування

При розробці програмного забезпечення інформаційної системи для керування нотатками було використано мову програмування C# [12], [13].

C# є мовою програмування, що базується на принципах відкритого коду, спрямованою на створення ефективного програмного забезпечення. Вона орієнтована на високу продуктивність системи, оптимізацію використання пам'яті та забезпечення швидкості виконання. Ця мова дозволяє розробляти програми, які функціонують на різних операційних системах. Об'єктно-орієнтований підхід, що підтримується C#, особливо ефективний у реалізації багатозадачності та роботи з паралельними процесами [12], [13].

C# забезпечує продуктивну компіляцію завдяки оптимізації коду, яка враховує типи даних, змінні та їх значення. Її функціонал широко застосовується у всіх аспектах створення програмного забезпечення: від операційних систем до браузерів і баз даних. У сфері техніки й медицини C# використовується для створення спеціалізованих програмних рішень [12], [13].

Серед переваг мови виділяють її популярність серед розробників за зручність у роботі, ефективність та швидкість створення коду. Вона дозволяє автоматично адаптуватися до доступного обладнання, що робить її придатною для клієнт-серверних систем. Ця мова також застосовується для розробки програмного забезпечення для настільних персональних комп'ютерів, скриптів і багатьох інших завдань, які виконуються на комп'ютері [12], [13].

Мова програмування C# забезпечує високу продуктивність завдяки своїй інтеграції з .NET Framework, що надає широкий спектр готових бібліотек і компонентів для розробки різних типів програм. Це дозволяє спростити реалізацію функцій, таких як авторизація, управління даними та інтерактивний інтерфейс [12], [13].

C# пропонує багатий набір функцій, які спрощують створення та підтримку програм. Завдяки своїй простоті й універсальності, мова підходить для створення інтуїтивно зрозумілих додатків. Також забезпечується зручність інтеграції з іншими програмними компонентами та можливість повторного використання коду [12], [13].

Обґрунтування вибору мови програмування для розробки програмного забезпечення інформаційної системи для керування нотатками наведено у таблиці 2.1.

Таблиця 2.1 – Обґрунтування вибору мови програмування для розробки програмного забезпечення інформаційної системи для керування нотатками

Критерій порівняння мов програмування	Мова програмування		
	C#	Java	Python
Зручність для створення ПЗ інформаційної системи для керування нотатками	+	—	—
Вбудовані інструменти для забезпечення безпеки	+	+	+—
Висока продуктивність	+	+	+—
Підтримка об'єктно-орієнтованого підходу	+	+	+—
Підтримка роботи з базами даних	+	+—	+

Отже, для реалізації програмного забезпечення інформаційної системи для керування нотатками обрано мову програмування C#, яка забезпечує високу продуктивність завдяки своїй інтеграції з .NET Framework, що надає широкий спектр готових бібліотек і компонентів для розробки програмних продуктів різного призначення. Крім того, ця мова є об'єктно-орієнтованою, що полегшує структурування коду і створення масштабованої архітектури,

завдяки чому можна легко підтримувати та розширювати систему в майбутньому.

2.2 Вибір середовища розробки для створення програмного забезпечення інформаційної системи для керування нотатками

В якості середовища розробки для створення програмного забезпечення інформаційної системи для керування нотатками було обрано середовище Microsoft Visual Studio [14], [15].

Середовище розробки Microsoft Visual Studio являє собою програму, створену корпорацією Microsoft, яка дозволяє розробляти графічні інтерфейси для програм різного типу, зокрема веб- і настільних додатків, використовуючи інструменти на основі Windows Forms і WPF. Це середовище підтримує інтеграцію з інструментами сторонніх розробників, такими як Subversion і Visual SourceSafe, а також містить розширені функції для моніторингу та розробки [14], [15].

За допомогою Visual Studio можна легко взаємодіяти з іншими інструментами, такими як Team Explorer у зв'язці з Team Foundation Server. Однією з переваг є вбудована система виявлення несправностей, яка автоматично повідомляє про проблеми й надає інформацію для їх вирішення. Тести, що виконуються в цьому середовищі, дозволяють швидко знаходити необхідні результати, які відповідають вашим запитам. Нові функції чи підтримка додаткових мов програмування можуть бути додані через установник Visual Studio [14], [15].

Перед виконанням тестування корисною є процедура безпечного очищення, яка запобігає можливим помилкам. Інструмент рефакторингу забезпечує можливості для редагування структури коду, такі як перейменування об'єктів, виділення окремих фрагментів у нові методи або зміна параметрів. Функція IntelliSense допомагає оптимізувати процес

написання коду, відображаючи підказки й дозволяючи автоматично вставляти фрагменти коду для зручності роботи [14], [15].

Середовище розробки Microsoft Visual Studio забезпечує спільну роботу в реальному часі, що дозволяє розробникам незалежно від типу програми чи мови програмування взаємодіяти більш ефективно. Це включає доступ до терміналів, запуск локальних програм чи спільну роботу через телефонний зв'язок. Також передбачено можливість переходу до визначення функцій чи типів даних через відповідні кнопки інтерфейсу [14], [15].

Visual Studio є потужним та інтегрованим середовищем, яке надає всі необхідні інструменти для розробки додатків на мові програмування C#. Це середовище пропонує зручний редактор коду, систему відладки, а також розширену підтримку для роботи з .NET Framework та .NET Core, що є ключовими для розробки ефективних інформаційних систем. Visual Studio підтримує створення багатоплатформних рішень, що дозволяє запускати розроблені програми на різних операційних системах, а також на мобільних пристроях [14], [15].

Також Visual Studio має великий набір розширень і плагінів для інтеграції з іншими інструментами розробки, включаючи системи контролю версій (наприклад, Git) і хмарні сервіси, що полегшує роботу команди розробників та забезпечує ефективне управління проектом [14], [15].

Додаткові функції, такі як «Ієрархія викликів» і CodeLens, сприяють аналізу й оптимізації коду. Вони дозволяють відстежувати зміни, знаходити помилки, покращувати продуктивність та навіть виконувати модульне тестування безпосередньо з редактора. Visual Studio пропонує широкий набір інструментів, який підходить для розробки програм на різних мовах програмування та для різноманітних задач [14], [15].

Обґрунтування вибору середовища розробки для створення програмного забезпечення інформаційної системи для керування нотатками наведено у таблиці 2.2.

Таблиця 2.2 – Обґрунтування вибору середовища розробки для створення програмного забезпечення інформаційної системи для керування нотатками

Критерій порівняння середовищ розробки	Середовища розробки		
	Microsoft Visual Studio	Eclipse	NetBeans
Підтримка мови C#	+	—	—
Зручність інструментів для розробки інтерфейсу користувача	+	+—	+—
Інтеграція з базами даних	+	+—	+
Інструменти для тестування	+	+	+—
Вбудована підтримка хмарних сервісів	+	—	—
Якість документації та навчальних матеріалів	+	+	+—

Таким чином, для створення програмного забезпечення інформаційної системи для керування нотатками обрано середовище розробки Microsoft Visual Studio, що є потужним та інтегрованим середовищем, яке надає всі необхідні інструменти для розробки програм на мові програмування C#, пропонує зручний редактор коду та засоби тестування.

2.3 Висновки за розділом 2

Для реалізації програмного забезпечення інформаційної системи для керування нотатками обрано мову програмування C#, яка забезпечує високу продуктивність завдяки своїй інтеграції з .NET Framework, що надає широкий

спектр готових бібліотек і компонентів для розробки програмних продуктів різного призначення. Крім того, ця мова є об'єктно-орієнтованою, що полегшує структурування коду і створення масштабованої архітектури, завдяки чому можна легко підтримувати та розширювати систему в майбутньому.

Для створення програмного забезпечення інформаційної системи для керування нотатками обрано середовище розробки Microsoft Visual Studio, що є потужним та інтегрованим середовищем, яке надає всі необхідні інструменти для розробки програм на мові програмування C#, пропонує зручний редактор коду та засоби тестування.

3 ОПИС ПРОГРАМИ

3.1 Структура програмного забезпечення інформаційної системи для керування нотатками

Структуру програмного забезпечення інформаційної системи для керування нотатками наведено на рис. 3.1.



Рисунок 3.1 – Структура програмного забезпечення інформаційної системи для керування нотатками

Структура програмного забезпечення інформаційної системи для керування нотатками користувача організована у вигляді п'яти модулів (модуль реєстрації та авторизації, модуль створення та керування нотатками,

модуль користувацького інтерфейсу, модуль управління даними, модуль синхронізації та резервного копіювання), які забезпечують ключові функції системи та зручність її використання.

Модуль реєстрації та авторизації забезпечує управління доступом до системи та ідентифікацію користувачів інформаційної системи для керування нотатками. Він виконує такі функції: реєстрація нових користувачів, авторизація користувачів за допомогою пароля, відновлення доступу у випадку втрати пароля, збереження сесій користувачів і управління терміном їхньої дії.

Модуль створення та керування нотатками є центральним модулем інформаційної системи, що відповідає за роботу з нотатками. Модуль створення та керування нотатками дозволяє виконувати створення нових текстових нотаток, редагування та видалення нотаток.

Модуль користувацького інтерфейсу відповідає за взаємодію користувача із програмним забезпеченням інформаційної системи для керування нотатками. Він включає інтуїтивно зрозумілий інтерфейс для створення, редагування, перегляду та організації нотаток.

Модуль управління даними відповідає за зберігання, оновлення та видалення даних. Основними компонентами цього модулю є база даних для збереження нотаток, логіка управління файлами, засоби обробки запитів на читання та запис даних.

Модуль синхронізації та резервного копіювання програмного забезпечення інформаційної системи для керування нотатками забезпечує збереження та синхронізацію даних між різними пристроями. Підтримує автоматичне резервне копіювання даних, а також відновлення даних у разі втрати чи збою.

Ця структура забезпечує гнучкість, зручність та безпеку програмного забезпечення інформаційної системи для керування нотатками, охоплюючи всі ключові аспекти роботи з нотатками.

3.2 Функціонування програмного забезпечення інформаційної системи для керування нотатками

Функціонування програмного забезпечення інформаційної системи для керування нотатками подамо за допомогою діаграми прецедентів, зображеної на рис. 3.2.

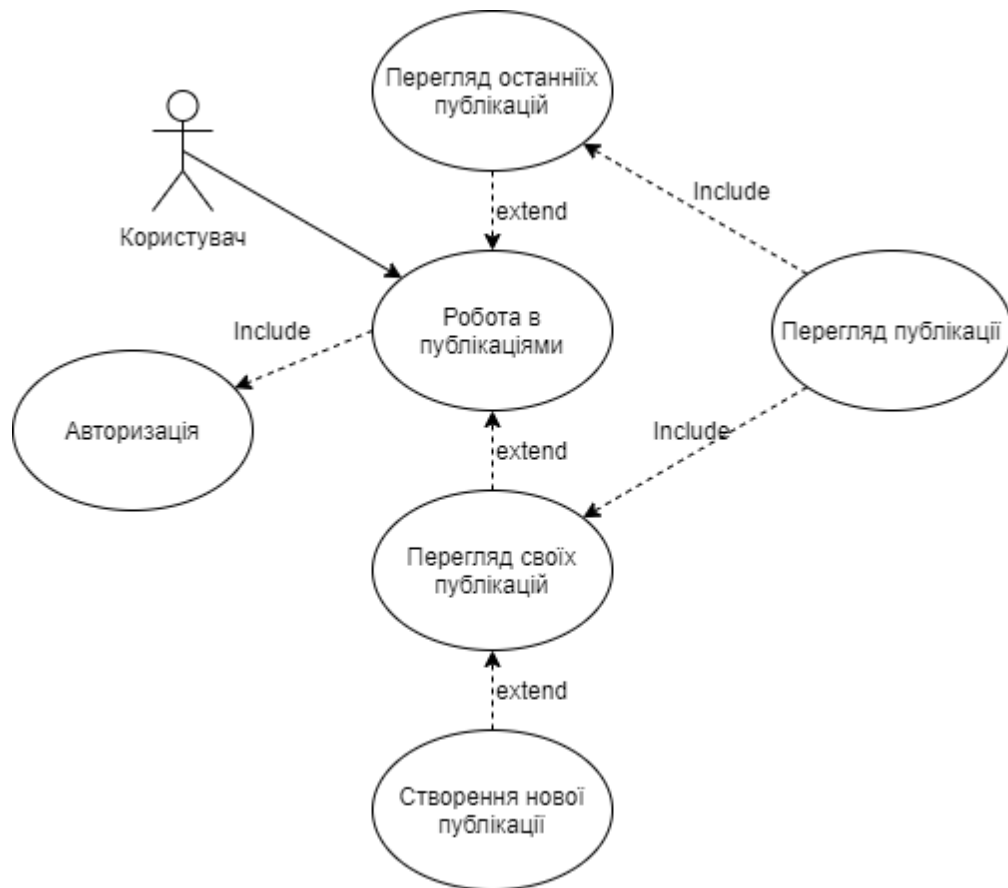


Рисунок 3.2 – Схема функціонування програмного забезпечення ІС для керування нотатками у вигляді діаграми прецедентів

Діаграма прецедентів для програмного забезпечення інформаційної системи керування нотатками описує взаємодію користувача із системою через кілька ключових сценаріїв. На ній зображено користувача, який є основним актором. Користувач взаємодіє із системою через функцію авторизації, що забезпечує вхід до системи за допомогою імені користувача та пароля.

Після успішної авторизації користувач програмного забезпечення інформаційної системи для керування нотатками отримує можливість працювати з публікаціями. Це включає перегляд існуючих нотаток. Також діаграма включає створення нових публікацій, що передбачає додавання контенту, його збереження та організацію у структурованому вигляді.

Ці дії взаємопов'язані, оскільки доступ до функцій перегляду та створення публікацій стає доступним тільки після успішної авторизації. Діаграма демонструє логічну послідовність та взаємодію між користувачем і системою в контексті ключових функцій.

Діаграму послідовності для створення нової нотатки користувача наведено на рис. 3.3.

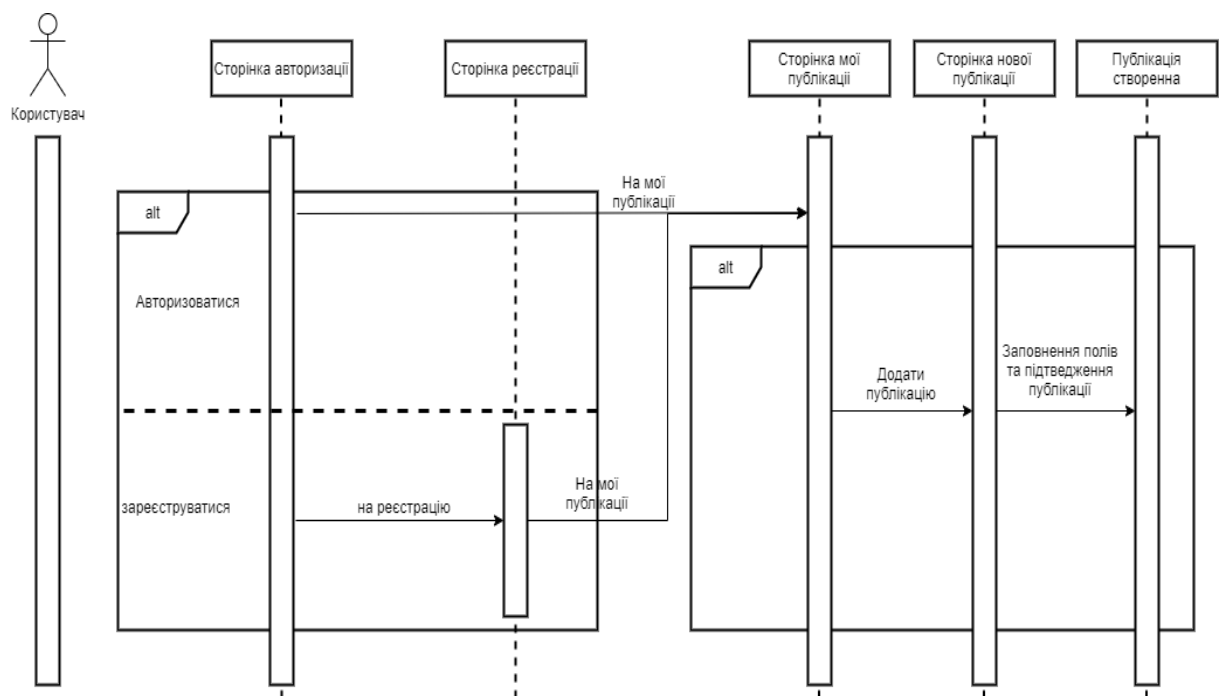


Рисунок 3.3 – Діаграма послідовності для створення нової нотатки користувача

Схему алгоритму створення публікації користувачем у програмному забезпеченні інформаційної системи для керування нотатками наведено на рис. 3.4.



Рисунок 3.4 – Схема алгоритму створення публікації користувачем у програмному забезпеченні ІС для керування нотатками

На основі рисунка 3.4 можна визначити ключові етапи роботи програмного забезпечення інформаційної системи для керування нотатками, які забезпечують створення нової нотатки користувачем. Спочатку

здійснюється завантаження сторінки авторизації, що дозволяє користувачеві виконати реєстрацію або авторизацію в системі. Після успішного входу відкривається сторінка «Мої публікації», яка надає доступ до особистого контенту. Для створення нового запису виконується перехід на сторінку «Нова публікація», де вводяться назва та основний контент майбутньої публікації. Завершальним етапом є збереження створеної публікації в базі даних програмного забезпечення інформаційної системи для керування нотатками для подальшого використання.

Діаграму послідовності для перегляду нотатки користувача наведено на рис. 3.5.

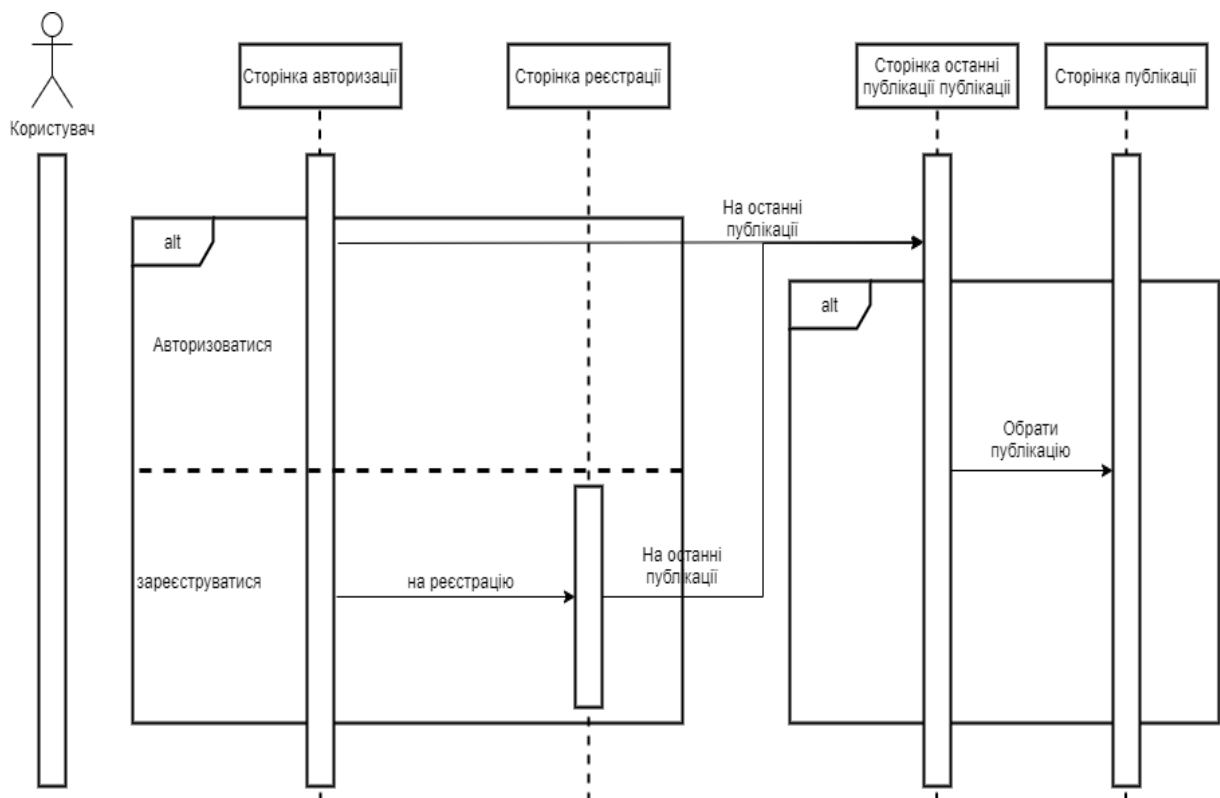


Рисунок 3.5 – Діаграма послідовності для перегляду нотатки користувача

Схему алгоритму перегляду публікації користувачем у програмному забезпеченні інформаційної системи для керування нотатками наведено на рис. 3.6.



Рисунок 3.6 – Схема алгоритму перегляду публікації користувачем у програмному забезпеченні ІС для керування нотатками

На основі рисунка 3.6 можна окреслити основні етапи, які виконує програмне забезпечення інформаційної системи для перегляду публікації. Спочатку здійснюється відкриття сторінки авторизації, де користувач проходить реєстрацію або авторизацію для отримання доступу до системи. Після цього завантажуються сторінка «Останні публікації», яка надає перелік

доступного контенту. Користувач обирає конкретну сторінку з публікацією, після чого відкривається сторінка, що містить обраний матеріал.

Діаграму переходів між сторінками програмного забезпечення інформаційної системи для керування нотатками наведено на рис. 3.7.

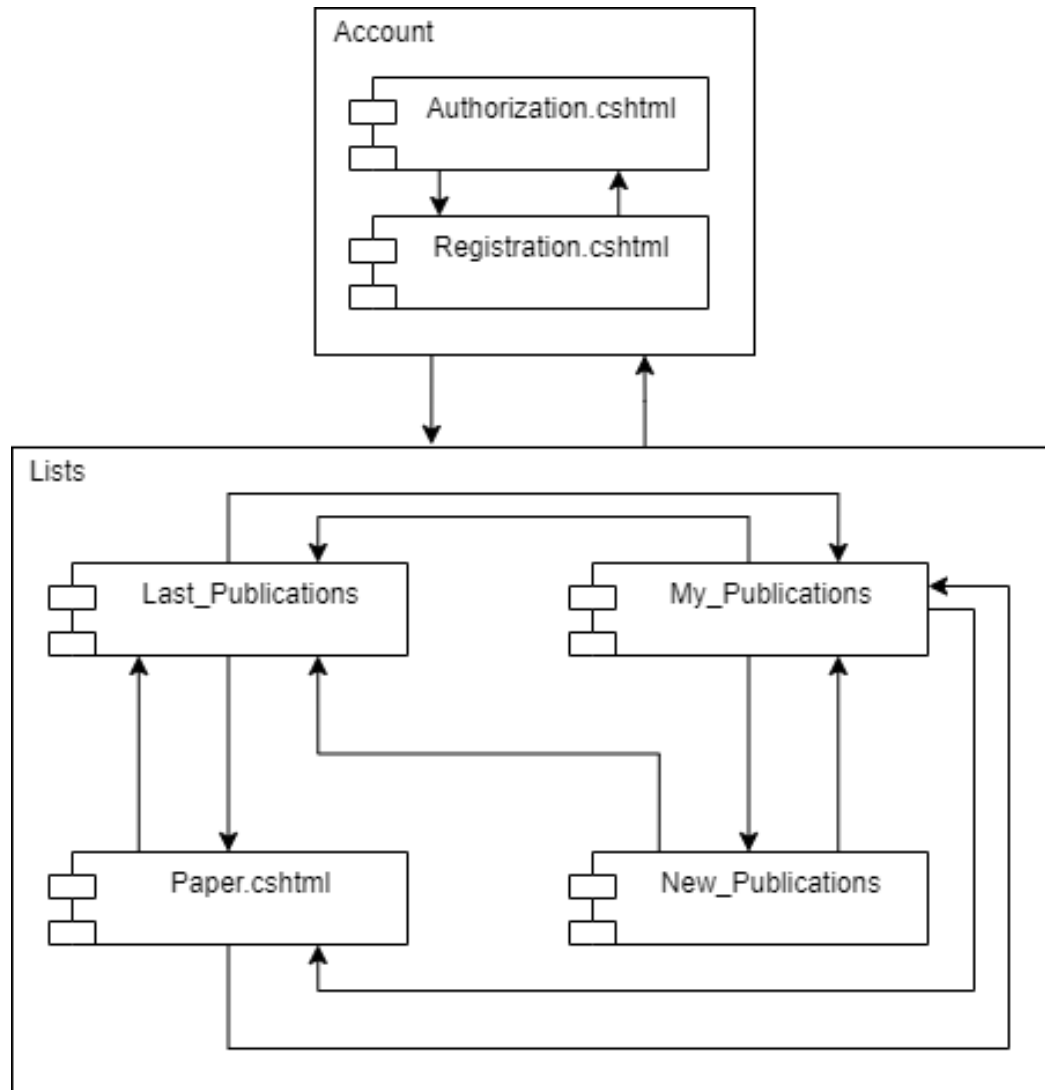


Рисунок 3.7 – Діаграма переходів між сторінками програмного забезпечення інформаційної системи для керування нотатками

Діаграма переходів між сторінками програмного забезпечення інформаційної системи для керування нотатками ілюструє можливі напрямки навігації в системі. Зокрема, забезпечується можливість переходу з авторизації на реєстрацію, а також доступ до сторінки авторизації з будь-якої іншої сторінки. Користувач може переходити на сторінки останніх публікацій з

будь-якого розділу інформаційної системи для керування нотатками. Крім того, зі сторінок своїх публікацій можна перейти до перегляду окремої публікації. Сторінка створення нової публікації доступна з розділу своїх публікацій. Така структура переходів забезпечує логічну послідовність і є зручною для користувачів програмного забезпечення інформаційної системи для керування нотатками.

3.3 Розробка бази даних програмного забезпечення інформаційної системи для керування нотатками

База даних інформаційної системи для керування нотатками складається з двох основних таблиць: «Користувачі» та «Нотатки».

Схему бази даних програмного забезпечення інформаційної системи для керування нотатками наведено на рис. 3.8.

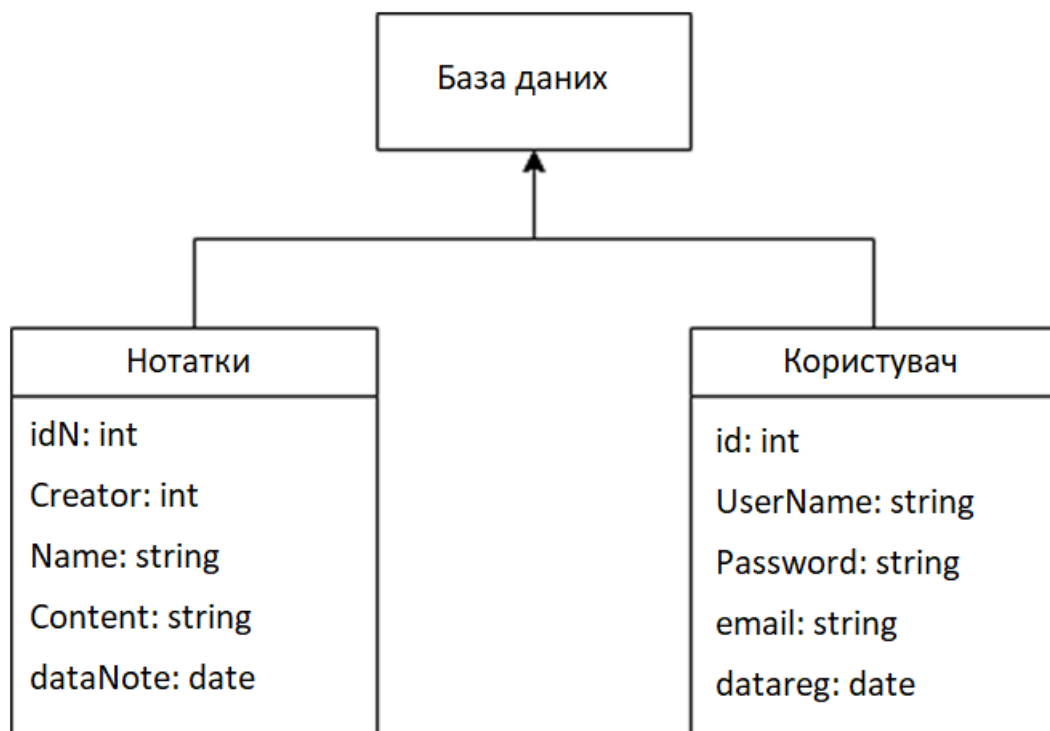


Рисунок 3.8 – База даних програмного забезпечення інформаційної системи для керування нотатками

Таблиця «Користувачі» зберігає дані про зареєстрованих користувачів і забезпечує управління доступом до системи. Кожен користувач має унікальний ідентифікатор, ім'я, електронну адресу, зашифрований пароль, а також інформацію про дату реєстрації та останній вхід до системи.

Таблиця «Нотатки» містить інформацію про всі створені записи. Вона пов'язана з таблицею «Користувачі» через поле, яке вказує на автора нотатки. У цій таблиці зберігаються унікальний ідентифікатор нотатки, її автор, заголовок, вміст, дата редагування.

Між таблицями реалізовано зв'язок за допомогою зовнішнього ключа, який забезпечує відповідність між користувачем і його нотатками. Така структура гарантує цілісність даних, дозволяє відслідковувати, кому належать записи, і підтримує багатокористувацький режим роботи системи.

3.4 Проєктування інтерфейсу програмного забезпечення інформаційної системи для керування нотатками

Проєктування інтерфейсу взаємодії користувача з програмним забезпеченням інформаційної системи для керування нотатками виконано з урахуванням вимог технічного завдання. При розробленні програмного забезпечення інформаційної системи для керування нотатками враховані функціональні вимоги до програми:

- підтримка можливості реєстрації та авторизації користувачів;
- підтримка можливості відображення списку нотаток користувача;
- можливість переходу між сторінками програми за допомогою відповідних кнопок;
- можливість додавання нотаток користувача;
- підтримка можливості ознайомлення з нотатками інших користувачів програми.

Інтерфейс головної форми програмного забезпечення інформаційної системи для керування нотатками наведено на рис. 3.9.

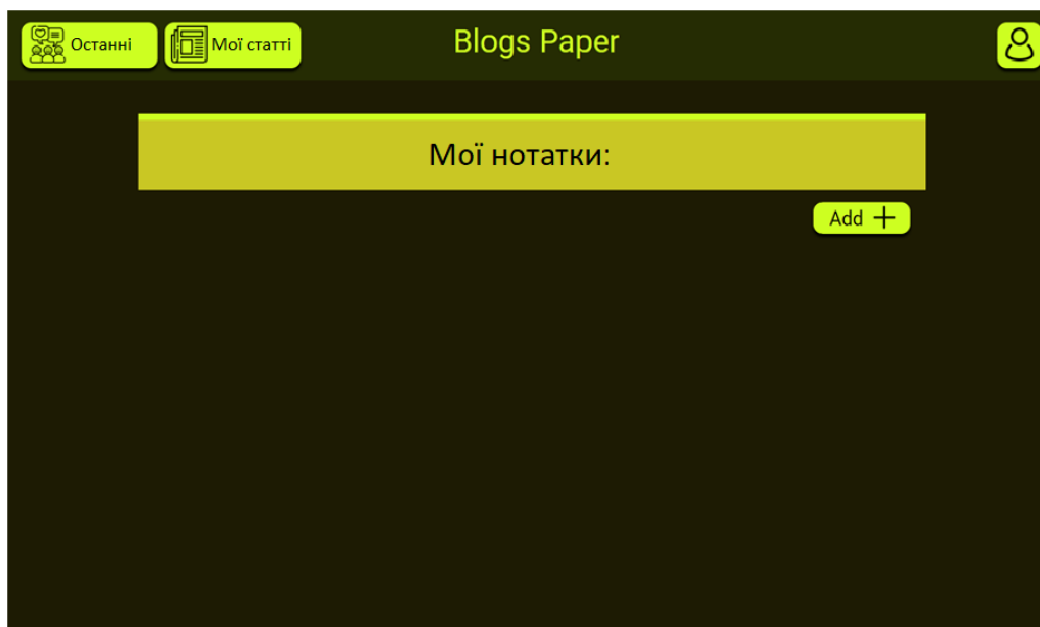


Рисунок 3.9 – Інтерфейс головної форми програмного забезпечення інформаційної системи для керування нотатками

Сторінку авторизації користувача програмного забезпечення інформаційної системи для керування нотатками наведено на рис. 3.10.



Рисунок 3.10 – Інтерфейс сторінки авторизації користувача програмного забезпечення інформаційної системи для керування нотатками

Сторінку реєстрації користувача програмного забезпечення інформаційної системи для керування нотатками наведено на рис. 3.11.

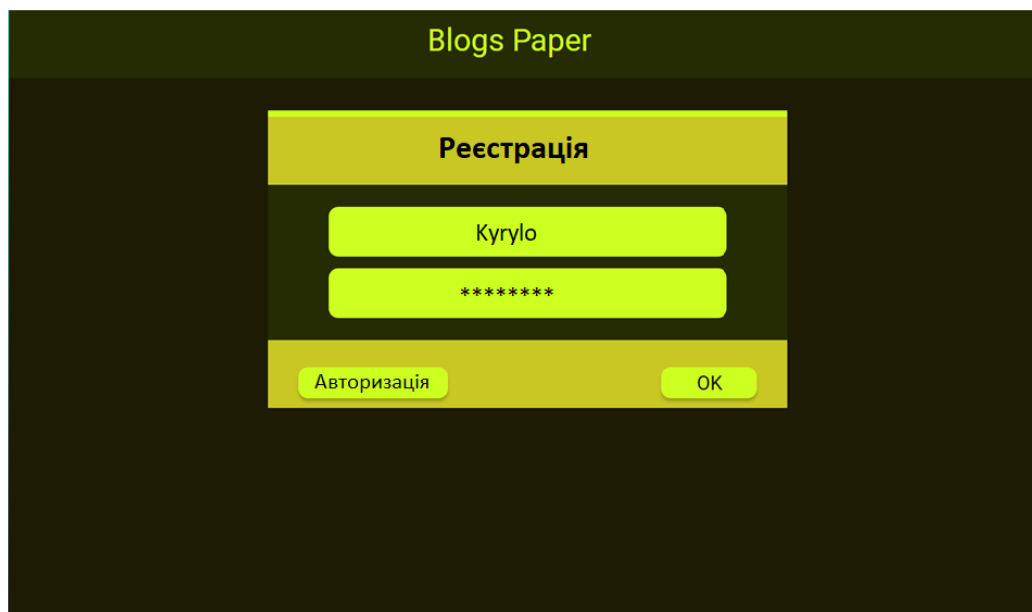


Рисунок 3.11 – Інтерфейс сторінки реєстрації користувача програмного забезпечення інформаційної системи для керування нотатками

Сторінку додавання нової нотатки наведено на рис. 3.12.

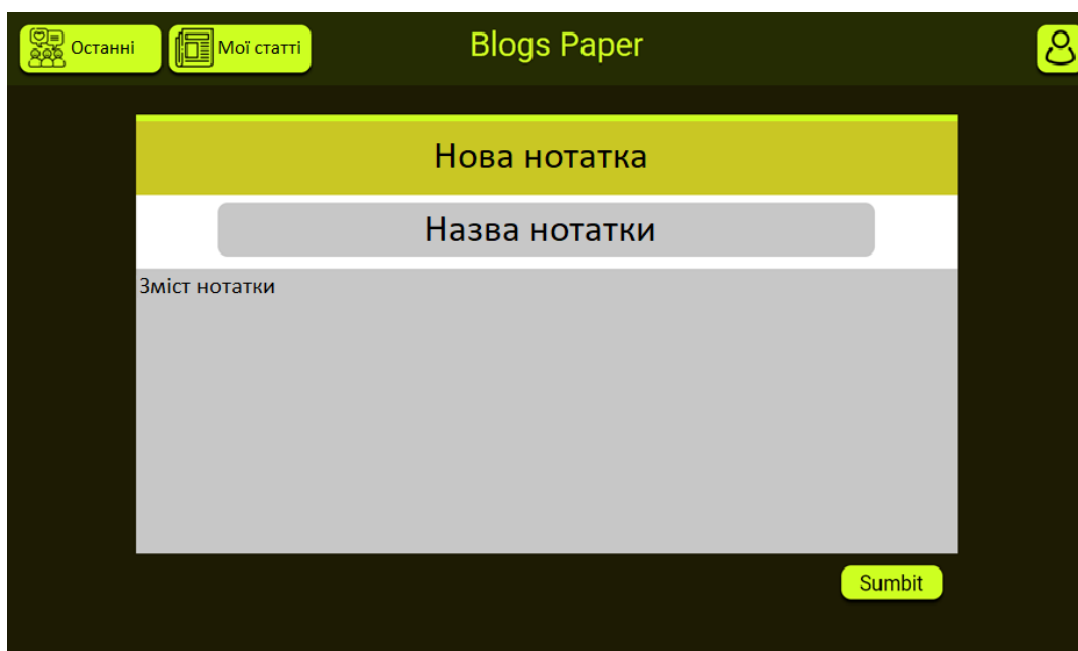


Рисунок 3.12 – Сторінка додавання нової нотатки

Сторінку з переліком останніх публікацій користувачів наведено на рис. 3.13.

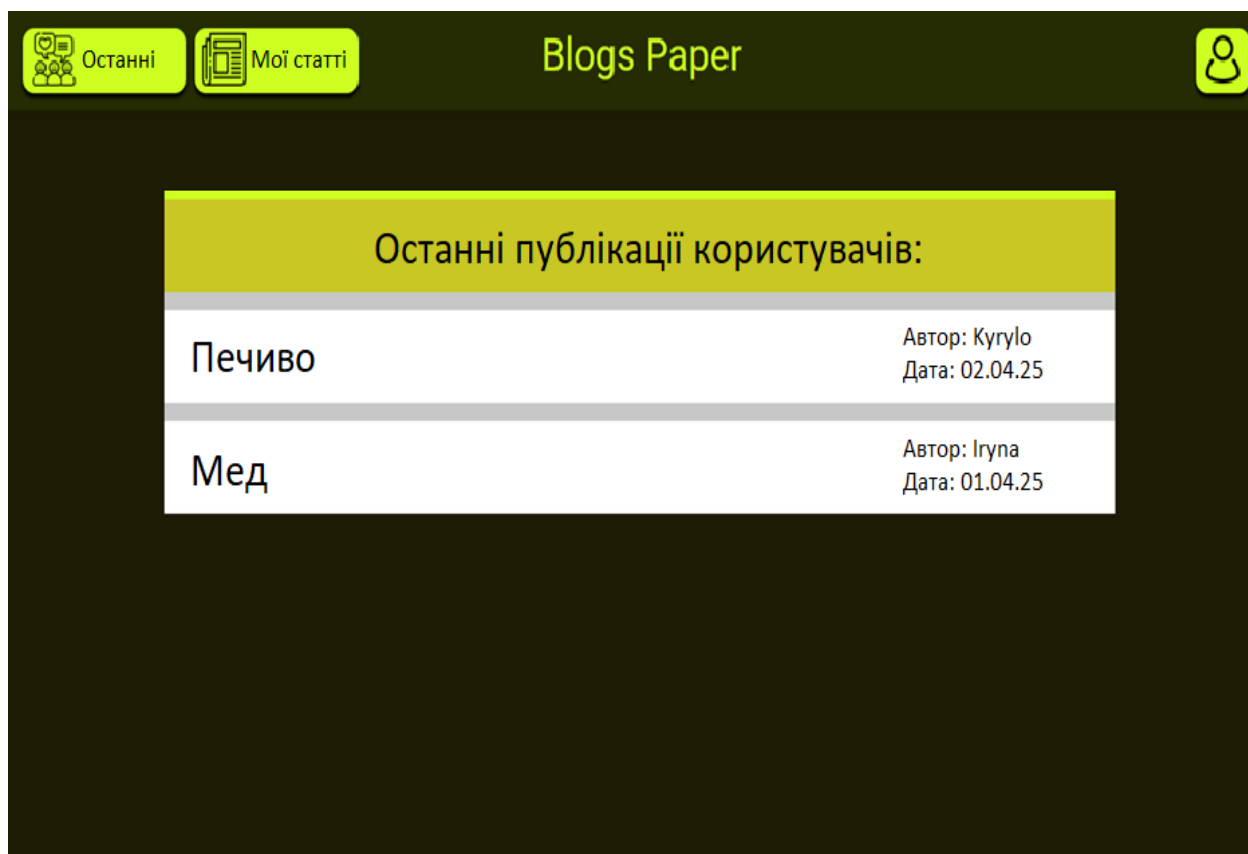


Рисунок 3.13 – Сторінка з переліком публікацій користувачів

3.5 Висновки за розділом 3

Розроблено програмне забезпечення інформаційної системи для керування нотатками.

Запропоновано структуру програмного забезпечення інформаційної системи для керування нотатками. Визначено, що структура програмного забезпечення інформаційної системи для керування нотатками користувача організована у вигляді п'яти модулів (модуль реєстрації та авторизації, модуль створення та керування нотатками, модуль користувацького інтерфейсу, модуль управління даними, модуль синхронізації та резервного копіювання), які забезпечують ключові функції системи та зручність її використання.

Розроблено базу даних програмного забезпечення інформаційної системи для керування нотатками. Визначено, що база даних інформаційної системи для керування нотатками складається з двох основних таблиць: «Користувачі» та «Нотатки». Таблиця «Користувачі» зберігає дані про зареєстрованих користувачів і забезпечує управління доступом до системи, таблиця «Нотатки» містить інформацію про всі створені записи.

Описано функціонування програмного забезпечення інформаційної системи для керування нотатками.

Описано особливості реалізації програмного забезпечення інформаційної системи для керування нотатками. Виконано проєктування інтерфейсу взаємодії користувача з програмним забезпеченням інформаційної системи для керування нотатками.

4 ЕКСПЛУАТАЦІЯ, ТЕСТУВАННЯ ТА ЕКСПЕРИМЕНТАЛЬНЕ ДОСЛІДЖЕННЯ ПРОГРАМИ

4.1 Призначення й умови застосування програми

Програма призначена для підтримки процесу функціонування для керування нотатками. Програма написана на мові C#, яка забезпечує високу продуктивність завдяки своїй інтеграції з .NET Framework, що надає широкий спектр готових бібліотек і компонентів для розробки програмних продуктів різного призначення. В якості середовища розробки для створення програмного забезпечення для керування нотатками обрано Microsoft Visual Studio, що є потужним та інтегрованим середовищем, яке надає всі необхідні інструменти для розробки програм на мові програмування C#, пропонує зручний редактор коду та засоби тестування.

Програмне забезпечення інформаційної системи для керування нотатками забезпечує виконання таких функцій:

- підтримка можливості реєстрації та авторизації користувачів;
- підтримка можливості відображення списку нотаток користувача;
- можливість переходу між сторінками програми за допомогою відповідних кнопок;
- можливість додавання нотаток користувача;
- підтримка можливості ознайомлення з нотатками інших користувачів програми.

Для роботи програмного забезпечення інформаційної системи для керування нотатками потрібні певні апаратні та програмні ресурси. До таких ресурсів належить процесор із тактовою частотою 3,6 ГГц та вбудованою графічною пам'яттю. Для роботи системи потрібна оперативна пам'ять обсягом 4 ГБ, блок живлення потужністю 450 Вт і накопичувач SSD на 500 ГБ для зберігання даних.

Функціональні характеристики розробленого програмного забезпечення інформаційної системи для керування нотатками наведено у

технічному завданні (додаток А). Фрагмент тексту програмного забезпечення інформаційної системи для керування нотатками наведено у додатку Б.

4.2 Характеристики програмного забезпечення інформаційної системи для керування нотатками

Програмне забезпечення інформаційної системи для керування нотатками користувача організована у вигляді п'яти модулів (модуль реєстрації та авторизації, модуль створення та керування нотатками, модуль користувацького інтерфейсу, модуль управління даними, модуль синхронізації та резервного копіювання), які забезпечують ключові функції системи та зручність її використання.

База даних інформаційної системи для керування нотатками складається з двох таблиць. Таблиця Користувачі зберігає дані про зареєстрованих користувачів і забезпечує управління доступом до системи. Кожен користувач має унікальний ідентифікатор, ім'я, електронну адресу, зашифрований пароль, а також інформацію про дату реєстрації та останній вхід до системи. Таблиця Нотатки містить інформацію про всі створені записи. Вона пов'язана з таблицею «Користувачі» через поле, яке вказує на автора нотатки. У цій таблиці зберігаються унікальний ідентифікатор нотатки, її автор, заголовок, вміст, дата редагування.

Програмне забезпечення інформаційної системи для керування нотатками надає функціональність, що дозволяє ефективно зберігати, організовувати та переглядати нотатки. Воно забезпечує простоту в користуванні, надаючи зручний інтерфейс для створення та редагування записів. Система підтримує безперешкодний доступ до даних користувача, що дозволяє швидко переглядати та редагувати нотатки, а також організовувати їх за допомогою тегів чи категорій.

Ще однією характеристикою програмного забезпечення інформаційної системи для керування нотатками є масштабованість. Програмне забезпечення

інформаційної системи для керування нотатками може підтримувати збільшення обсягу даних та кількості користувачів без втрати ефективності. Підтримка багатокористувацької роботи є важливою для організації доступу до нотаток, що дозволяє кільком користувачам працювати з програмою.

Завдяки всім цим характеристикам, програмне забезпечення є стабільним, надійним і зручним для користувачів, забезпечуючи ефективне керування нотатками у різних умовах.

4.3 Інструкція по експлуатації програми

4.3.1 Звернення до програми

Для запуску програмного забезпечення інформаційної системи для керування нотатками необхідно перейти на відповідну сторінку у браузері.

4.3.2 Вхідні й вихідні дані

Вхідними даними до програмного забезпечення інформаційної системи для керування нотатками є інформація про користувача, а також інформація про нотатку, зокрема, текст нотатки та ін.

Вихідними даними програмного забезпечення інформаційної системи для керування нотатками є створені у базі даних записи про користувачів та нотатки.

4.3.3 Повідомлення

Користувач програмного забезпечення інформаційної системи для керування нотатками може отримати повідомлення про те, що не заповнені усі поля введення.

4.4 Виконання програмного забезпечення інформаційної системи для керування нотатками

Після запуску програмного забезпечення інформаційної системи для керування нотатками користувачу відображається сторінка авторизації (рис. 4.1), де користувачу необхідно ввести свої логін та пароль, або пройти процедуру реєстрації, натиснувши відповідну кнопку.

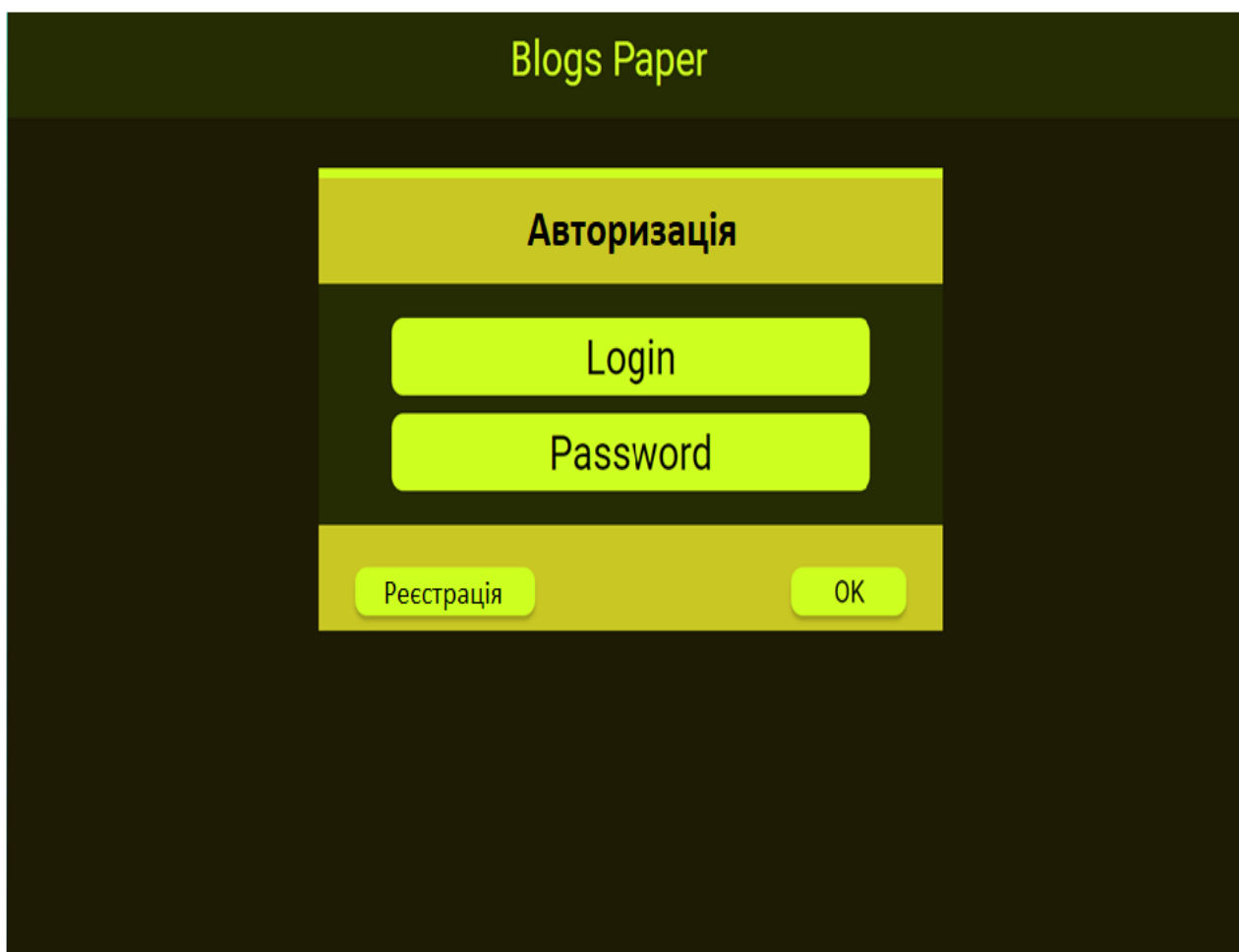


Рисунок 4.1 – Інтерфейс сторінки авторизації користувача програмного забезпечення інформаційної системи для керування нотатками

У випадку, якщо користувач натиснув на кнопку Реєстрація, відбувається перехід до сторінки реєстрації (рис. 4.2), де необхідно ввести свої реєстраційні дані.

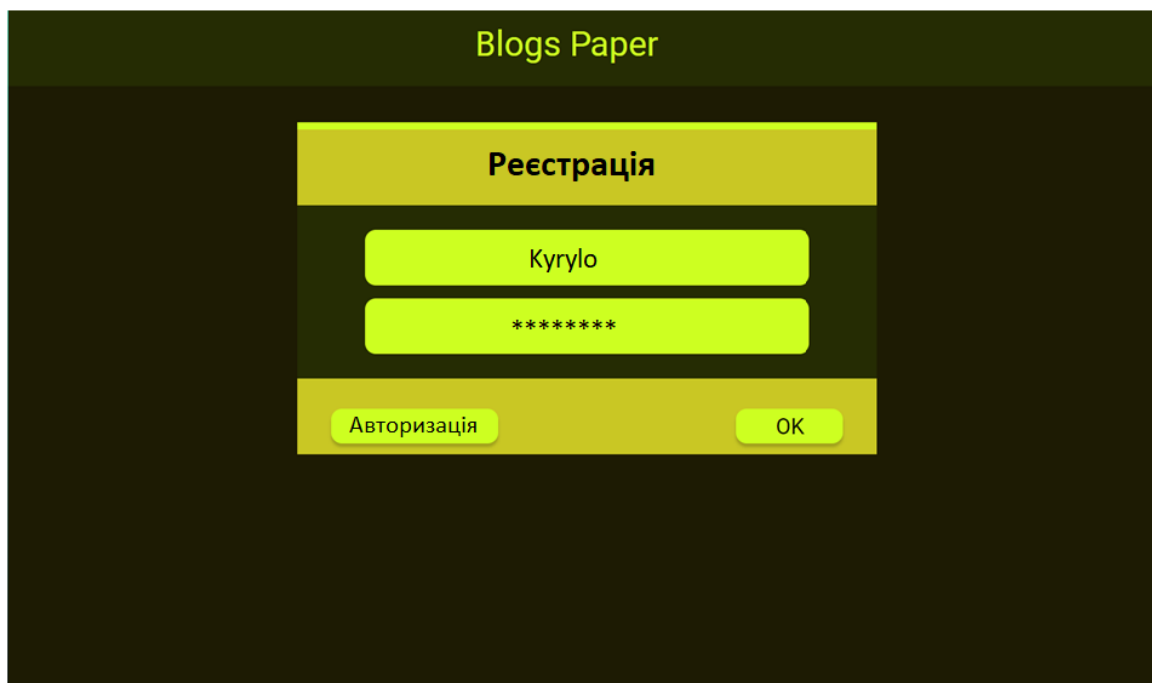


Рисунок 4.2 – Інтерфейс сторінки реєстрації користувача програмного забезпечення інформаційної системи для керування нотатками

Після входу у систему на головній сторінці (рис. 4.3) відображається список публікацій користувача. Головна сторінка також містить засоби для навігації між публікаціями та кнопку для відображення профілю користувача.

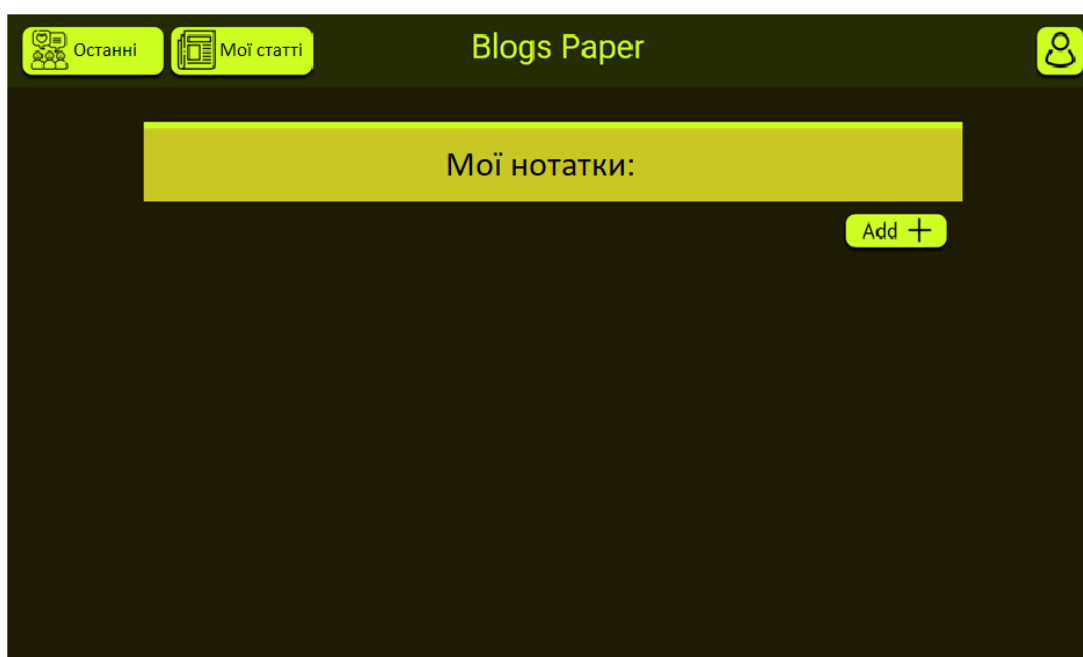


Рисунок 4.3 – Головна сторінка програми

Для створення нової нотатки необхідно натиснути кнопку Add (рис. 4.4) та ввести назву й текст нотатки.

The screenshot shows the 'Blogs Paper' app interface. At the top, there is a dark green header with two buttons on the left: 'Останні' (Last) and 'Мої статті' (My articles), and a user profile icon on the right. Below the header, the main content area has a yellow bar at the top with the text 'Нова нотатка' (New note). Underneath this bar is a form with a grey input field labeled 'Назва нотатки' (Note name) and a larger grey text area labeled 'Зміст нотатки' (Note content). At the bottom right of the form is a yellow button labeled 'Submit'.

Рисунок 4.4 – Додавання нової нотатки

Як результат додавання нотатки буде створено та збережено у базі даних відповідні записи з текстом та назвою публікації. Публікація також буде відображена на відповідній сторінці (рис. 4.5).

The screenshot shows the 'Blogs Paper' app interface with a published note. The header is the same as in the previous image. The main content area displays a note titled 'Печиво' (Biscuits). To the right of the title, it shows the author 'Автор: Kyrylo' and the date 'Дата: 02.04.25'. Below the title is a grey text box containing the following text: 'Печиво — кондитерські вироби з борошна. До тіста печива часто додають різні спеції та горіхи, сухофрукти чи цукрові вироби. Часто печиво покривають шоколадом, або виготовляють начинку та розміщують її всередині печива. В Україні печиво виготовляли ще здавна. Є багато старих рецептів приготування печива. Одним із найпоширеніших видів печива були медяники, що готувалися з меду та пшеничного чи житнього тіста.'

Рисунок 4.5 – Додана нотатка користувача

При переході за кнопкою Останні буде відображено перелік останніх публікацій користувачів (рис. 4.6).

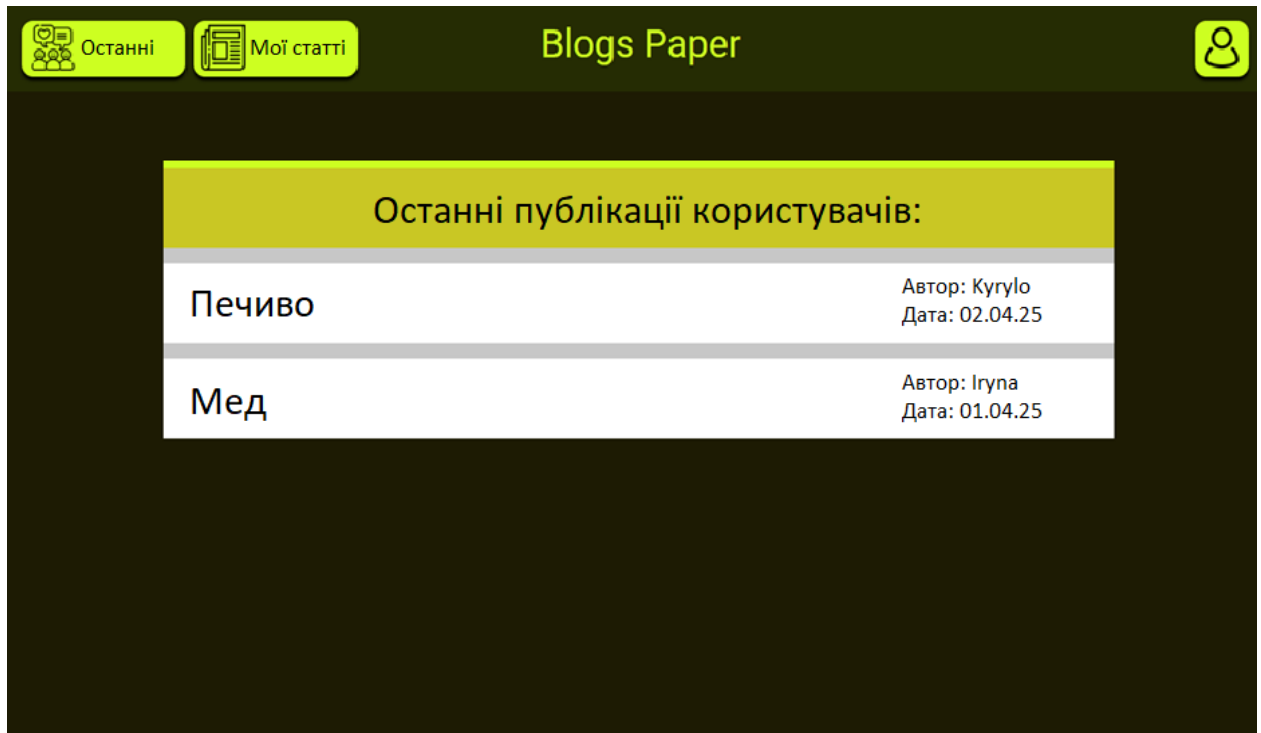


Рисунок 4.6 – Перелік публікацій користувачів

4.5 Тестування програмного забезпечення інформаційної системи для керування нотатками

Проведено тестування програмного забезпечення інформаційної системи для керування нотатками.

Підтверджено, що програмне забезпечення інформаційної системи для керування нотатками працює коректно і стабільно, виконуючи всі необхідні функціональні вимоги та успішно справляючись із основною задачею підтримки процесу підтримки діяльності користувача.

Розроблене програмне забезпечення забезпечує автоматизацію операцій інформаційної системи, пов'язаних із підтримкою процесів для керування нотатками.

4.6 Висновки за розділом 4

Описано програмне забезпечення інформаційної системи для керування нотатками.

Проведено тестування створеного програмного забезпечення інформаційної системи для керування нотатками. Результати тестування показали, що програмне забезпечення інформаційної системи для керування нотатками забезпечує автоматизацію процесів, пов'язаних із підтримкою діяльності користувача.

ВИСНОВКИ

В ході виконання дипломної кваліфікаційної роботи бакалавра було проаналізовано та досліджено процес розробки інформаційних систем для керування нотатками.

Визначено, що керування нотатками є процесом організації, створення, редагування, збереження та доступу до текстової, графічної чи іншої інформації у формі нотаток. Це може бути реалізовано за допомогою спеціалізованих інструментів або програмного забезпечення, які забезпечують зручність зберігання, пошуку, упорядкування та синхронізації нотаток між пристроями. Інформаційні системи для керування нотатками являють собою програмні комплекси або платформи, створені для організації, обробки, збереження та обміну текстовою інформацією у вигляді нотаток.

За результатами проведеного аналізу зроблено висновок, що у наш час існує досить багато інформаційних систем для керування нотатками. Проте деякі інформаційні системи для керування нотатками вимагають стабільного доступу до Інтернету для синхронізації даних, що може стати проблемою в умовах обмеженого або нестабільного підключення. Це особливо актуально для хмарних платформ, які без доступу до мережі втрачають значну частину функціональності. У деяких інформаційних системах для керування нотатками спостерігається надмірна складність інтерфейсу. Розширений набір функцій іноді може заплутати користувачів, які потребують простих і зрозумілих інструментів для роботи. Деякі програмні засоби інформаційних систем для керування нотатками можуть бути обмежені у функціональності або вимагати оплати за повноцінний доступ до всіх функцій, що може стати бар'єром для окремих категорій користувачів. Тому актуальною є розробка програмного забезпечення інформаційної системи для керування нотатками.

Сформульовано функціональні вимоги до програмного забезпечення інформаційної системи для керування нотатками.

Для реалізації програмного забезпечення інформаційної системи для керування нотатками обрано мову програмування C#, яка забезпечує високу продуктивність завдяки своїй інтеграції з .NET Framework, що надає широкий спектр готових бібліотек і компонентів для розробки програмних продуктів різного призначення. Крім того, ця мова є об'єктно-орієнтованою, що полегшує структурування коду і створення масштабованої архітектури, завдяки чому можна легко підтримувати та розширювати систему в майбутньому.

Для створення програмного забезпечення інформаційної системи для керування нотатками обрано середовище розробки Microsoft Visual Studio, що є потужним та інтегрованим середовищем, яке надає всі необхідні інструменти для розробки програм на мові програмування C#, пропонує зручний редактор коду та засоби тестування.

Розроблено програмне забезпечення інформаційної системи для керування нотатками.

Запропоновано структуру програмного забезпечення інформаційної системи для керування нотатками. Визначено, що структура програмного забезпечення інформаційної системи для керування нотатками користувача організована у вигляді п'яти модулів (модуль реєстрації та авторизації, модуль створення та керування нотатками, модуль користувацького інтерфейсу, модуль управління даними, модуль синхронізації та резервного копіювання), які забезпечують ключові функції системи та зручність її використання.

Розроблено базу даних програмного забезпечення інформаційної системи для керування нотатками. Визначено, що база даних інформаційної системи для керування нотатками складається з двох основних таблиць: «Користувачі» та «Нотатки». Таблиця «Користувачі» зберігає дані про зареєстрованих користувачів і забезпечує управління доступом до системи, таблиця «Нотатки» містить інформацію про всі створені записи.

Описано функціонування програмного забезпечення інформаційної системи для керування нотатками.

Описано особливості реалізації програмного забезпечення інформаційної системи для керування нотатками. Виконано проєктування інтерфейсу взаємодії користувача з програмним забезпеченням інформаційної системи для керування нотатками.

Проведено тестування створеного програмного забезпечення інформаційної системи для керування нотатками. Результати тестування показали, що програмне забезпечення інформаційної системи для керування нотатками забезпечує автоматизацію процесів, пов'язаних із підтримкою діяльності користувача.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАНЬ

1. How to organize notes [Electronic resource]. – Access mode: <https://able.ac/blog/how-to-organize-notes/>.
2. Tips for Managing Your Digital Notes [Electronic resource]. – Access mode: <https://www.pcmag.com/how-to/10-tips-for-managing-your-digital-notes>.
3. How to Organize Your Notes Effectively [Electronic resource]. – Access mode: <https://securespace.com/blog/how-to-organize-your-notes/>.
4. The Best Note-Taking Methods and Types [Electronic resource]. – Access mode: <https://www.sembly.ai/blog/the-best-note-taking-methods-and-types/>.
5. Best Note-Taking Software [Electronic resource]. – Access mode: <https://www.g2.com/categories/note-taking-software>.
6. Best Note Taking Apps for Boosting Productivity [Electronic resource]. – Access mode: <https://peoplemanagingpeople.com/tools/best-note-taking-apps/>.
7. Best Note-Taking Software [Electronic resource]. – Access mode: <https://www.softwareworld.co/note-taking-software/>.
8. Best Release Notes Management Software [Electronic resource]. – Access mode: <https://sourceforge.net/software/release-notes-management/>.
9. Notion [Electronic resource]. – Access mode: <https://www.notion.com/product>.
10. Simplenote [Electronic resource]. – Access mode: <https://simplenote.com/>.
11. Evernote [Electronic resource]. – Access mode: <https://evernote.com/>.
12. C#. Overview [Electronic resource]. – Access mode: https://www.tutorialspoint.com/csharp/csharp_overview.htm.
13. Base C# [Electronic resource]. – Access mode: <https://www.dotnetperls.com/base>.
14. Visual Studio product family documentation [Electronic resource]. – Access mode: <https://learn.microsoft.com/en-us/visualstudio/?view=vs-2022>.

15. Introduction to Visual Studio [Electronic resource]. – Access mode:
<https://www.geeksforgeeks.org/introduction-to-visual-studio/>.

ДОДАТОК А
Технічне завдання

Вступ

Програмне забезпечення може використовуватися для підтримки процесу діяльності інформаційної системи для керування нотатками.

А.1 Підстава для розробки

Підставою для розробки є завдання на дипломну кваліфікаційну роботу на тему «Програмне забезпечення інформаційної системи для керування нотатками користувача», затверджене наказом Національного університету «Запорізька політехніка» № 209 від 28 квітня 2025 р.

А.2 Призначення розробки

Програмний продукт призначений для підтримки процесу діяльності для керування нотатками.

А.3 Основні вимоги до програми, що розробляється

А.3.1 Вимоги до функціональних характеристик

Програмне забезпечення інформаційної системи для керування нотатками забезпечує виконання таких функцій:

- підтримка можливості реєстрації та авторизації користувачів;
- підтримка можливості відображення списку нотаток користувача;
- можливість переходу між сторінками програми за допомогою відповідних кнопок;
- можливість додавання нотаток користувача;
- підтримка можливості ознайомлення з нотатками інших користувачів програми.

А.3.2 Вимоги до інтерфейсу програми

Інтерфейс програмного забезпечення інформаційної системи для керування нотатками повинен бути зручним для користувачів. Повинна бути забезпечена можливість роботи в візуальному режимі.

А.3.3 Вимоги до надійності

Програмне забезпечення інформаційної системи для керування нотатками повинно забезпечити надійне функціонування.

А.3.4 Умови експлуатації

Для експлуатації програмного забезпечення інформаційної системи для керування нотатками необхідна наявність персонального комп'ютера.

А.3.5 Вимоги до складу та параметрів технічний засобів

Для роботи програмного забезпечення інформаційної системи для керування нотатками потрібні певні апаратні та програмні ресурси, зокрема, процесор із тактовою частотою 3,6 ГГц та вбудованою графічною пам'яттю. Для роботи системи потрібна оперативна пам'ять обсягом 4 ГБ, блок живлення потужністю 450 Вт і накопичувач на 500 ГБ для зберігання даних.

А.3.6 Вимоги до маркування і пакування

Програма для підтримки діяльності для керування нотатками може бути записана на будь-якому носії інформації.

На пакуванні повинна бути назва програми – «Програмне забезпечення інформаційної системи для керування нотатками користувача».

A.4 Вхідні дані до роботи

Вхідними даними до програмного забезпечення інформаційної системи для керування нотатками є інформація про користувача, а також інформація про нотатку, зокрема, текст нотатки та ін.

Вихідними даними програмного забезпечення інформаційної системи для керування нотатками є створені у базі даних записи про користувачів та нотатки.

ДОДАТОК Б
Фрагмент тексту програми

```

namespace NttkManagementApp
{
    public partial class LgnForm : Form
    {
        public LgnForm()
        {
            InitializeComponent();
        }

        private void btnLgn_Click(object sender, EventArgs e)
        {
            string krstnm = txtKrstnm.Text;
            string pswrd = txtPswrd.Text;

            if (DtMdl.AuthenticateKorystv(krstnm, pswrd))
            {
                MessageBox.Show("Авторизація успішна!");
                MnFrm mnFrm = new MnFrm(krstnm);
                mnFrm.Show();
                this.Hide();
            }
            else
            {
                MessageBox.Show("Невірне ім'я користувача або
пароль.");
            }
        }

        private void btnRgstr_Click(object sender, EventArgs e)
        {
            RgstrForm rgstrForm = new RgstrForm();
            rgstrForm.ShowDialog();
        }
    }

    public partial class RgstrForm : Form
    {
        public RgstrForm()
        {
            InitializeComponent();
        }

        private void btnRgstr_Click(object sender, EventArgs e)
        {
            string krstnm = txtKrstnm.Text;
            string pswrd = txtPswrd.Text;

```

```

        if (DtMdl.RgstrKorystv(krstnm, pswrd))
        {
            MessageBox.Show("Реєстрація успішна!");
            this.Close();
        }
        else
        {
            MessageBox.Show("Користувач із таким іменем вже
існує.");
        }
    }
}

public partial class MnFrm : Form
{
    private string currntUsr;

    public MnFrm(string krstnm)
    {
        InitializeComponent();
        currntUsr = krstnm;
        LoadNttks();
    }

    private void LoadNttks()
    {
        var Nttks = DtMdl.GetNttks(currntUsr);
        listBoxNttks.Items.Clear();
        foreach (var Nttk in Nttks)
        {
            listBoxNttks.Items.Add(Nttk.Zaglvk);
        }
    }

    private void btnNewNttk_Click(object sender, EventArgs e)
    {
        NttkEditorForm NttkEditor = new
NttkEditorForm(currntUsr);
        NttkEditor.ShowDialog();
        LoadNttks();
    }

    private void btnDeleteNttk_Click(object sender, EventArgs
e)
    {
        if (listBoxNttks.SelectedItem != null)
        {

```

```

        string zaglvk =
listBoxNttks.SelectedItem.ToString();
        DtMdl.DeleteNttk(crrntUsr, zaglvk);
        LoadNttks();
    }
}

private void btnBackup_Click(object sender, EventArgs e)
{
    if (BackupModule.BackupNttks(crrntUsr))
    {
        MessageBox.Show("Резервне копіювання успішно
виконано!");
    }
}

public partial class NttkEditorForm : Form
{
    private string crrntUsr;

    public NttkEditorForm(string krstnm)
    {
        InitializeComponent();
        crrntUsr = krstnm;
    }

    private void btnSave_Click(object sender, EventArgs e)
    {
        string zaglvk = txtZaglvk.Text;
        string zmst = txtZmst.Text;

        if (DtMdl.SaveNttk(crrntUsr, zaglvk, zmst))
        {
            MessageBox.Show("Нотатку збережено!");
            this.Close();
        }
        else
        {
            MessageBox.Show("Помилка збереження.");
        }
    }
}

public static class DtMdl
{
    private static Dictionary<string, string> korystvs = new
Dictionary<string, string>();

```

```

        private static Dictionary<string, List<Nttk>>
korystvNttks = new Dictionary<string, List<Nttk>>();

        public static bool RgstrKorystv(string krstnm, string
pswrd)
        {
            if (korystvs.ContainsKey(krstnm))
                return false;

            korystvs[krstnm] = pswrd;
            korystvNttks[krstnm] = new List<Nttk>();
            return true;
        }

        public static bool AuthenticateKorystv(string krstnm,
string pswrd)
        {
            return korystvs.ContainsKey(krstnm) &&
korystvs[krstnm] == pswrd;
        }

        public static List<Nttk> GetNttks(string krstnm)
        {
            return korystvNttks.ContainsKey(krstnm) ?
korystvNttks[krstnm] : new List<Nttk>();
        }

        public static bool SaveNttk(string krstnm, string zaglvk,
string zmst)
        {
            if (!korystvNttks.ContainsKey(krstnm))
                return false;

            korystvNttks[krstnm].Add(new Nttk { Zaglvk = zaglvk,
Zmst = zmst });
            return true;
        }

        public static void DeleteNttk(string krstnm, string
zaglvk)
        {
            if (korystvNttks.ContainsKey(krstnm))
            {
                korystvNttks[krstnm].RemoveAll(Nttk =>
Nttk.Zaglvk == zaglvk);
            }
        }
    }

```

```

public class Nttk
{
    public string Zaglvk { get; set; }
    public string Zmst { get; set; }
}

public static class BackupModule
{
    public static bool BackupNttks(string krstnm)
    {
        try
        {
            string backupFile = $"{krstnm}_backup.txt";
            var Nttks = DtMdl.GetNttks(krstnm);
            using (StreamWriter writer = new
StreamWriter(backupFile))
            {
                foreach (var Nttk in Nttks)
                {
                    writer.WriteLine($"Zaglvk:
{Nttk.Zaglvk}");
                    writer.WriteLine($"Zmst: {Nttk.Zmst}");
                    writer.WriteLine($"---");
                }
            }
            return true;
        }
        catch
        {
            return false;
        }
    }
}

public class PrflController : Controller
{
    private ApplicationKntx _kntx;
    public PrflController(ApplicationKntx kntx)
    {
        _kntx = kntx;
    }
    public IActionResult Registration()
    {
        return View();
    }
    public async Task<IActionResult>
Registration(RegistrationModel model)

```

```

        {
            if (ModelState.IsValid)
            {
                Korystv korystv = await
_kntx.Korystvs.FirstOrDefaultAsync(u => u.Krstnm ==
model.Krstnm);
                if (korystv == null)
                {
                    korystv = new Korystv { Krstnm =
model.Krstnm, Pswrd = model.Pswrd };
                    Role korystvRole = await
_kntx.Roles.FirstOrDefaultAsync(r => r.Name == "korystv");
                    if (korystvRole != null)
                        korystv.Role = korystvRole;

                    _kntx.Korystvs.Add(korystv);
                    await _kntx.SaveChangesAsync();

                    await Authenticate(korystv);

                    return RedirectToAction("Останні публікації",
"Lists");
                }
                else
                    ModelState.AddModelError("", "Некоректні
дані");
            }
            return View(model);
        }
        public IActionResult Authorization()
        {
            if (Korystv.Identity.Name != null)
            {
                HttpKntx.SignOutAsync();
                return RedirectToAction("Останні публікації",
"Lists");
            }
            return View();
        }
        public async Task<IActionResult>
Authorization(AuthorizationModel model)
        {
            if (ModelState.IsValid)
            {
                Korystv korystv = await _kntx.Korystvs
                    .Include(u => u.Role)
                    .FirstOrDefaultAsync(u => u.Krstnm ==
model.Krstnm && u.Pswrd == model.Pswrd);

```

```

        if (korystv != null)
        {
            await Authenticate(korystv);
            return RedirectToAction("Останні публікації",
"Lists");
        }
        ModelState.AddModelError("", "Некоректні дані");
    }
    return View(model);
}
private async Task Authenticate(Korystv korystv)
{
    var klmb = new List<Klmb>
    {
        new Klmb(KlmbIdentity.DefaultNameKlmbType,
korystv.Krstnm),
        new Klmb(KlmbIdentity.DefaultRoleKlmbType,
korystv.Role?.Name)
    };
    KlmbIdentity id = new KlmbIdentity(klmb,
"ApplicationCookie", KlmbIdentity.DefaultNameKlmbType,
KlmbIdentity.DefaultRoleKlmbType);
    await
HttpKntx.SignInAsync(CookieAuthenticationDefaults.Authentications
cheme, new KlmbPrincipal(id));
}
}
}

```

```

namespace LaDipl.Controllers
{
    public class ListsController : Controller
    {
        ApplicationKntx db;
        public ListsController(ApplicationKntx kntx)
        {
            db = kntx;
        }
        public IActionResult My_Ntkms()
        {
            List<Ntkm> LstNtk = new List<Ntkm>();
            foreach (var Pub in db.Ntkms){
                if (Pub.Creator == Korystv.Identity.Name) {
                    LstNtk.Add(Pub);
                }
            }
            return View(LstNtk);
        }
    }
}

```

```

    }
    public IActionResult Paper(int Id)
    {
        ViewBag.Ntkm = db.Ntkms.Find(Id);
        return View();
    }

    public IActionResult New_Ntkm()
    {
        return View();
    }
    public async Task<IActionResult> New_Ntkm(NtkmModel
model)
    {
        Ntkm ntkm = new Ntkm { Name = model.Name, Zmst =
model.Zmst), Creator = Korystv.Identity.Name, DateNtkm =
DateTime.Now.ToString("dd.mm.yyyy") };
        db.Ntkms.Add(ntkm);
        await db.SaveChangesAsync();

        return View();
    }
}
}

```

ДОДАТОК В
Слайди презентації

Міністерство освіти і науки України
Національний університет «Запорізька політехніка»
Кафедра програмних засобів

Програмне забезпечення для керування нотатками користувача

Виконав: Кирило ХЛЕПІТЬКО

ст. групи КНТ-212сп

Керівник: Тетяна КОЛПАКОВА

к.т.н., доцент НУ «Запорізька політехніка»

Рисунок В.1 – Слайд 1

Об'єкт, предмет та мета роботи

Об'єкт дослідження – процес розробки інформаційних систем для керування нотатками.

Предмет дослідження – інформаційні системи для керування нотатками.

Метою роботи є розробка програмного забезпечення інформаційної системи для керування нотатками користувача.

Рисунок В.2 – Слайд 2

Завдання роботи

Для досягнення поставленої мети у дипломній кваліфікаційній роботі бакалавра необхідно розв'язати такі задачі:

- виконати аналіз предметної області та інформаційних систем для керування нотатками;
- здійснити проектування програмного забезпечення інформаційної системи для керування нотатками користувача;
- створити програмне забезпечення інформаційної системи для керування нотатками користувача;
- виконати тестування розробленого програмного забезпечення інформаційної системи для керування нотатками користувача.

3

Рисунок В.3 – Слайд 3

Порівняння існуючих аналогів

Критерій порівняння	Notion[9]	Simplenote	Evernote [11]
Підтримка багатофункціональних нотаток	+	–	+
Простота використання	+–	+	+–
Безкоштовний план	+–	+	+–
Підтримка мультимедійних нотаток	+	–	+
Засоби керування проектами	+	–	+–
Інтеграція з іншими програмами	+	+–	+

4

Рисунок В.4 – Слайд 4

Порівняння мов програмування

Критерій порівняння мов програмування	Мова програмування		
	C#	Java	Python
Зручність для створення ПЗ інформаційної системи для керування нотатками	+	–	–
Вбудовані інструменти для забезпечення безпеки	+	+	+–
Висока продуктивність	+	+	+–
Підтримка об'єктно-орієнтованого підходу	+	+	+–
Підтримка роботи з базами даних	+	+–	+

5

Рисунок В.5 – Слайд 5

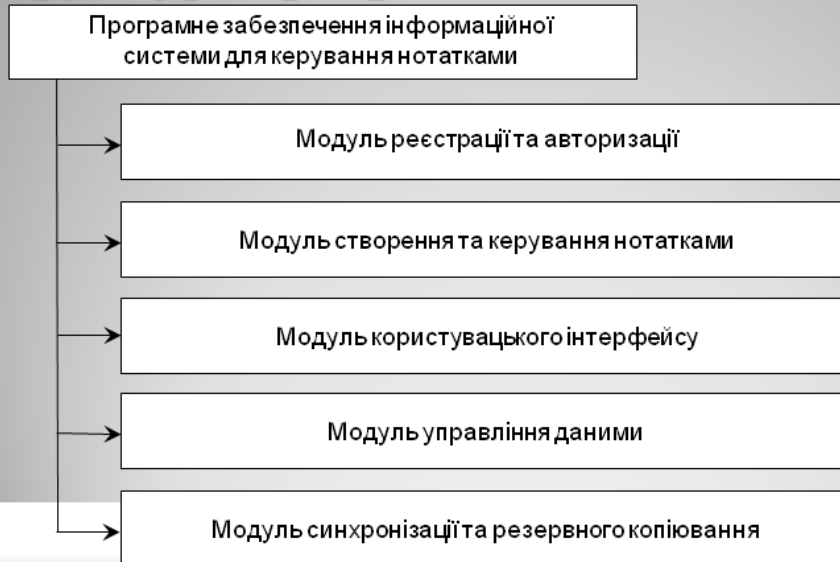
Порівняння середовищ розробки

Критерій порівняння середовищ розробки	Середовища розробки		
	Microsoft Visual Studio	Eclipse	NetBeans
Підтримка мови C#	+	–	–
Зручність інструментів для розробки інтерфейсу користувача	+	+–	+–
Інтеграція з базами даних	+	+–	+
Інструменти для тестування	+	+	+–
Вбудована підтримка хмарних сервісів	+	–	–
Якість документації та навчальних матеріалів	+	+	+–

6

Рисунок В.6 – Слайд 6

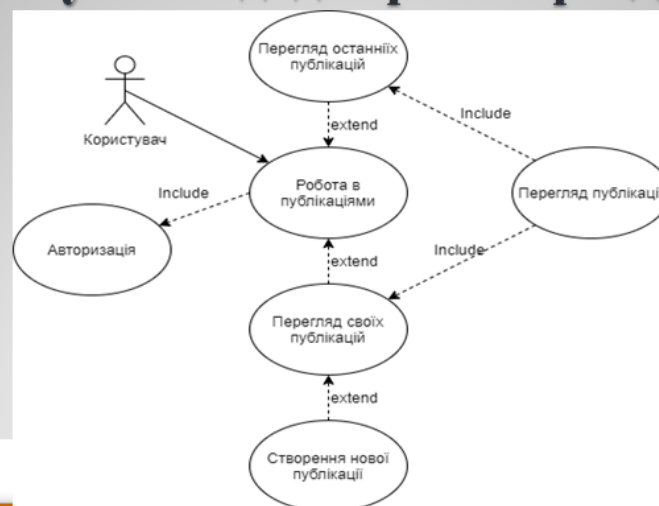
Структура програми



7

Рисунок В.7 – Слайд 7

Схема функціонування ПЗ ІС для керування нотатками у вигляді діаграми прецедентів



8

Рисунок В.8 – Слайд 8

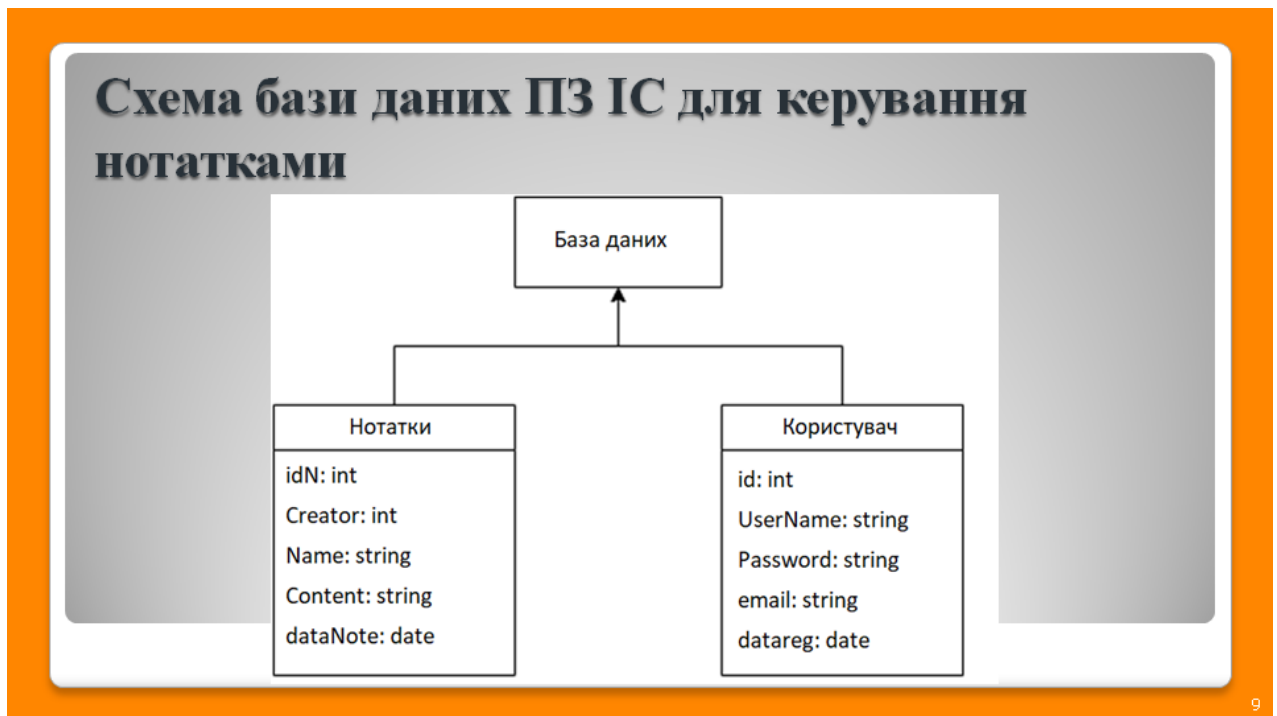


Рисунок В.9 – Слайд 9

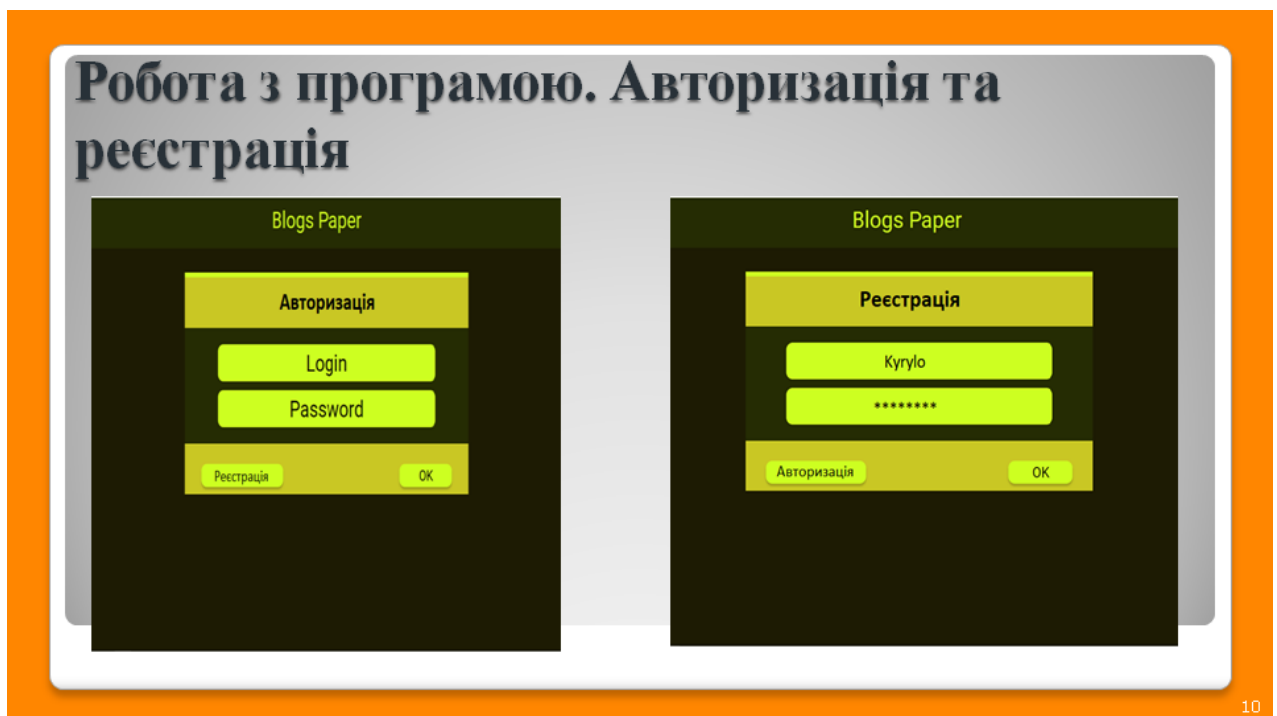


Рисунок В.10 – Слайд 10

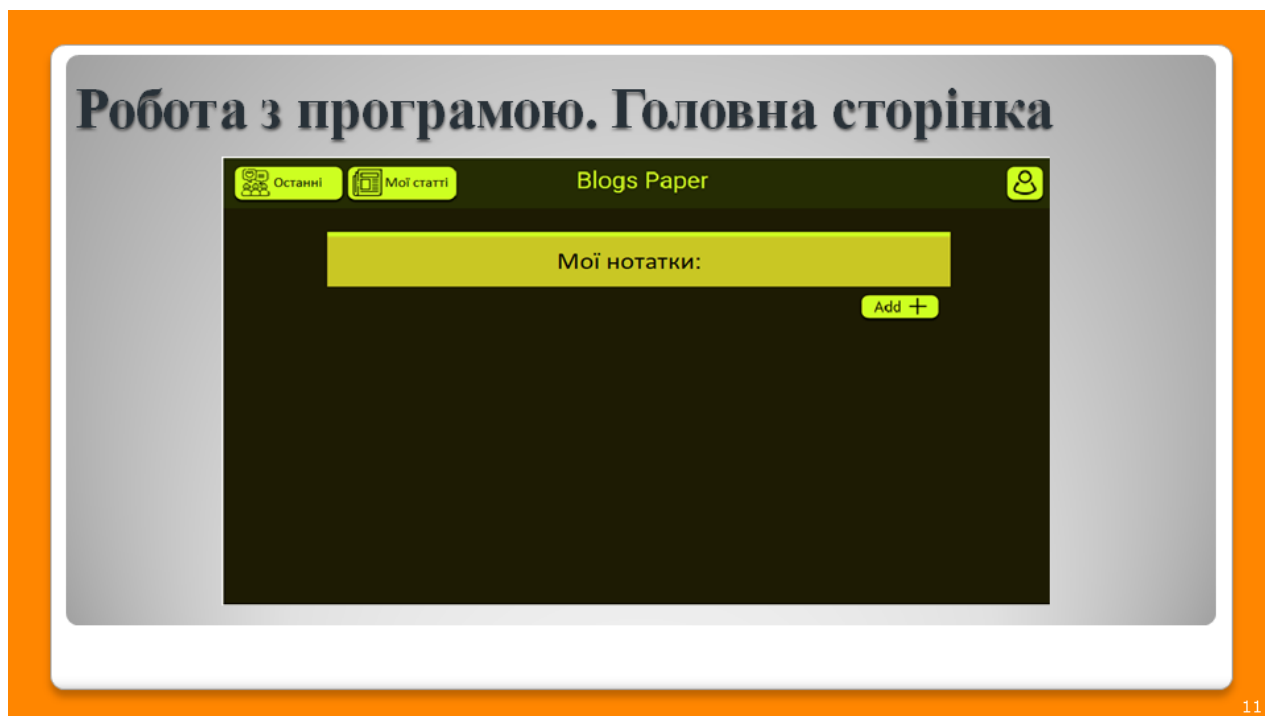


Рисунок В.11 – Слайд 11

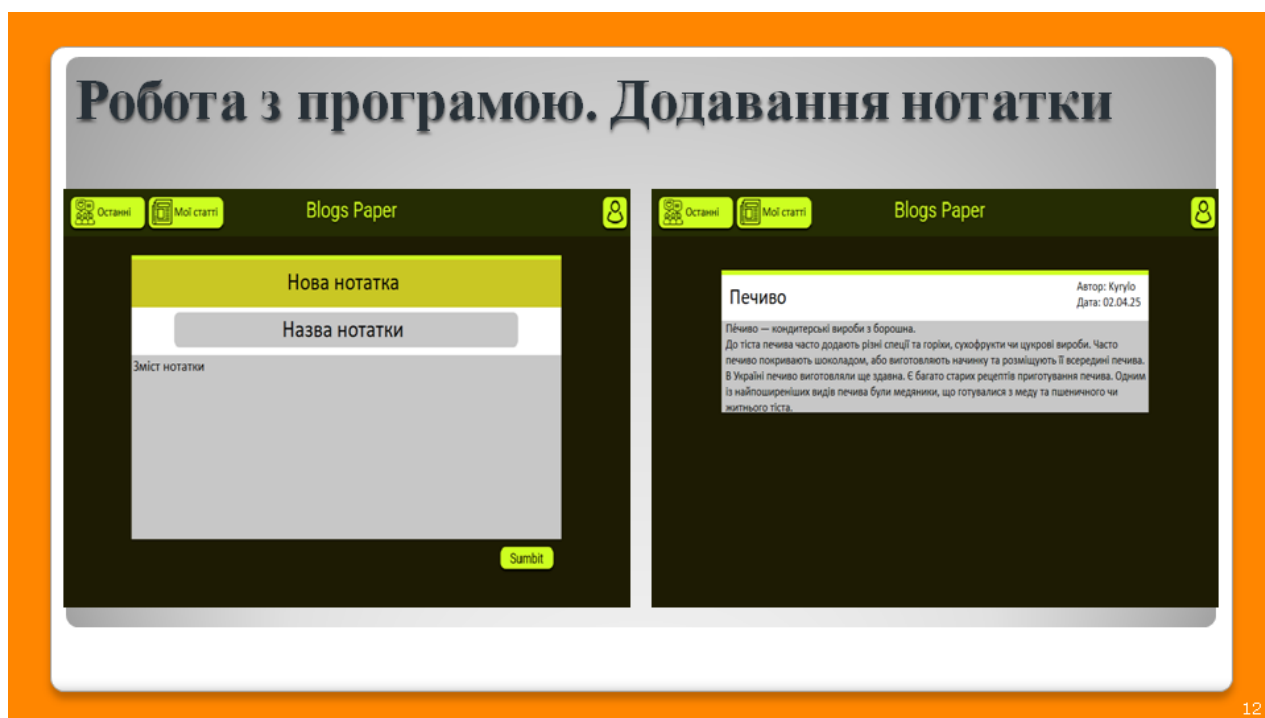


Рисунок В.12 – Слайд 12

Висновки

В ході виконання дипломної кваліфікаційної роботи бакалавра було проаналізовано та досліджено процес розробки інформаційних систем для керування нотатками.

Сформульовано функціональні вимоги до програмного забезпечення інформаційної системи для керування нотатками. Розроблено програмне забезпечення інформаційної системи для керування нотатками. Розроблено базу даних програмного забезпечення інформаційної системи для керування нотатками.

Виконано тестування програмного забезпечення інформаційної системи для керування нотатками.

Усі завдання випускної дипломної роботи бакалавра повністю виконано.

Рисунок В.13 – Слайд 13