

**МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
КИЇВСЬКИЙ НАЦІОНАЛЬНИЙ ЕКОНОМІЧНИЙ УНІВЕРСИТЕТ
ІМЕНІ ВАДИМА ГЕТЬМАНА
Навчально-науковий інститут
«Інститут інформаційних технологій в економіці»
Кафедра інформаційних систем в економіці**

ОСВІТНЬО-ПРОФЕСІЙНА ПРОГРАМА «КОМП'ЮТЕРНІ НАУКИ»

галузь знань 12 «Інформаційні технології»

спеціальність 122 «Комп'ютерні науки»

Форма навчання: денна

КВАЛІФІКАЦІЙНИЙ БАКАЛАВРСЬКИЙ ПРОЕКТ

на тему

**РОЗРОБКА ЗАСТОСУНКУ ДЛЯ ВІДСТЕЖЕННЯ ПРОГРЕСУ ЧИТАННЯ
КНИГ**

здобувача Пучко Оксани Костянтинівни _____

Науковий керівник:

ст. викладач

_____ Сквордякова О.О.

**Проект допущений до захисту перед
екзаменаційною комісією з атестації
здобувачів вищої освіти**

завідувач кафедри:

к.е.н., доцент

_____ Тішков Б.О.

Київ 2024

Міністерство освіти і науки України
Київський національний економічний університет імені Вадима Гетьмана
Навчально-науковий інститут «Інститут інформаційних технологій в економіці»
Кафедра інформаційних систем в економіці

ОСВІТНЬО-ПРОФЕСІЙНА ПРОГРАМА «КОМП'ЮТЕРНІ НАУКИ»

галузь знань 12 «Інформаційні технології»

спеціальність 122 «Комп'ютерні науки»

ПОГОДЖЕНО:

Керівник проектної групи(гарант)
освітньо-професійної програми

_____Іванченко Г.Ф.
“_____” _____ 2024 р.

ЗАТВЕРДЖУЮ:

Завідувач кафедри
_____Тішков Б.О.

“_____” _____ 2024 р.

ІНДИВІДУАЛЬНЕ ЗАВДАННЯ

здобувача вищої освіти ***Пучко Оксани Костянтинівни***

очної (денної) форми навчання

на підготовку кваліфікаційного бакалаврського проекту
на тему: «Розробка застосунку для відстеження прогресу читання книг»

Тему затверджено наказом ректора Університету від «11» _03_ 2024 р. № 529-ст.

**Кваліфікаційний бакалаврський проект виконується на матеріалах,
отриманих з мережі Інтернет**

План кваліфікаційного бакалаврського проекту

Розділ I. Характеристика та аналіз предметної області

Розділ II. Розробка вимог і моделювання інформаційної системи

Розділ III. Проектування та реалізація компонентів системи

Об'єкт дослідження: процеси та механізми автоматизації моніторингу
читання книг.

Предмет дослідження: методи автоматизації та інструменти
програмного забезпечення для моніторингу читання книг, а також їх вплив
на користувацький досвід і ефективність взаємодії користувачів з
застосунком.

Мета кваліфікаційного бакалаврського проекту: розробка і
впровадження ефективного програмного забезпечення для моніторингу
читання книг, яке дозволить користувачам зручно відстежувати свій прогрес.

Конкретні завдання, які студент повинен виконати для досягнення поставленої мети:

У розділі I: Описати предметну область, визначити основні поняття та терміни. Проаналізувати поточний стан застосунків з моніторингу читання книг.

У розділі II: Сформулювати функціональні та не функціональні вимоги до інформаційної системи, описати загальну постановку задачі та обґрунтувати різні методи її розв'язання.

У розділі III: Розробити інформаційне забезпечення системи, включаючи загальну характеристику, організацію збору та передавання первинної інформації, побудову системи класифікації, структуру інформаційних масивів, вибір СКБД та створення інфологічної моделі бази даних, описати технічне забезпечення, зокрема загальні положення, схему автоматизації, структуру комплексу технічних засобів, а також представити результати реалізації інформаційної системи.

**Завдання підготував
науковий керівник _____**

Сквордякова Ольга Олегівна

“__15__” ____ 03 ____ 2024 р.

**Завдання одержав
студент _____**

Пучко Оксана Костянтинівна

“__15__” ____ 03 ____ 2024 р.

Відгук
про кваліфікаційний бакалаврський проект
здобувача навчально-наукового інституту
«Інститут інформаційних технологій в економіці»
освітньо-професійної програми
«Комп'ютерні науки»

Пучко Оксани Костянтинівни

на тему

«Розробка застосунку для відстеження прогресу читання книг»

1. **Актуальність** дослідження полягає в необхідності підтримки та стимулювання читацької активності в умовах цифрової епохи. Застосунок з моніторингу читання книг допоможе зробити читання більш організованим, мотивуючим та насиченим. Такий застосунок є важливим інструментом для розвитку освітнього та культурного середовища.
2. **Позитивні риси кваліфікаційного бакалаврського проекту:** здобувачем було застосовано актуальні методи аналізу та синтезу інформації, а також програмного забезпечення для розробки та тестування застосунку для відстеження прогресу читання книг.
3. **Наявність самостійних розробок автора:** здобувачем було спроектовано та розроблено застосунок для моніторингу читання, запропоновано своє бачення використання методів покращення процесу відстеження прогресу читання.
4. **Цінність теоретичних висновків та практичних рекомендацій:** розроблена система моніторингу процесу читання книг є ефективним інструментом для покращення організації навчання та підвищення мотивації до читання. Результати пілотного впровадження підтверджують доцільність та ефективність використання системи в освітніх та бібліотечних установах.
5. **Наявність недоліків:** постановка задачі потребує системного аналізу, інформаційна модель задачі – усунення неточностей, а розроблена система – подальшого вдосконалення за рахунок розширення функціональних можливостей.
6. **Загальна оцінка кваліфікаційного бакалаврського проекту та його допущення до захисту перед ЕК:** кваліфікаційний бакалаврський проект здобувача Пучко О.К. відповідає встановленим вимогам методичних вказівок, рекомендується до захисту і заслуговує позитивної оцінки.

Науковий керівник



ст. викладач Сквордякова О.О.

“17” _____ червня _____ 202

АНОТАЦІЯ

Даний кваліфікаційний бакалаврський проєкт присвячений розробці застосунку для відстеження прогресу читання книг. Метою роботи є створення зручного та функціонального інструменту, який допоможе користувачам ефективно відстежувати свій прогрес у читанні, організовувати бібліотеку книг, встановлювати цілі та аналізувати свої читанські звички. Розроблено інфологічну модель бази даних для зберігання інформації про книги та користувацькі досягнення, обрано відповідну СКБД. Результатом роботи є повнофункціональний застосунок, який може бути використаний широкою аудиторією для підвищення мотивації до читання та ефективного управління особистою бібліотекою.

РЕФЕРАТ

Бакалаврський кваліфікаційний проект містить 69 сторінок, 3 таблиці, 31 рисунок, 4 додатки, список використаних джерел з 14 найменувань.

Предметом дослідження є процеси та методи розробки застосунку для відстеження прогресу читання книг, включаючи аналіз вимог користувачів, проектування архітектури системи, розробку інформаційної моделі, вибір відповідних технологій та інструментів для реалізації застосунку, а також методи забезпечення зручності використання і продуктивності системи.

Об'єктом дослідження є сам застосунок з моніторингу читання книг, який має бути створений, розроблений та впроваджений для вирішення вказаних вище завдань.

Метою даного проекту є створення функціональної та зручної системи, яка допоможе користувачам ефективно організовувати свою читацьку діяльність, відстежувати прогрес та збагачувати читацький досвід.

Відповідно до поставленої мети були визначені такі завдання:

- аналіз ринку подібних застосунків;
- розробка детальної специфікації вимог до системи, включаючи функціональні та нефункціональні вимоги;
- створення інфологічної моделі бази даних для зберігання інформації про книги, користувачів та їх прогрес у читанні;
- вибір СКБД, яка найкраще відповідає вимогам системи;
- реалізація серверної та клієнтської частини застосунку, забезпечивши їх інтеграцію.

Практичне значення отриманих результатів полягає в створенні зручного та функціонального застосунку для відстеження прогресу читання книг, який допоможе користувачам ефективно організовувати своє читання, встановлювати та досягати читанських цілей, аналізувати свої звички та підвищувати мотивацію до читання.

Рік виконання кваліфікаційного бакалаврського проекту 2024.

Рік захисту кваліфікаційного бакалаврського проекту 2024.

Зміст

ВСТУП	3
РОЗДІЛ 1. ХАРАКТЕРИСТИКА ТА АНАЛІЗ ПРЕДМЕТНО ОБЛАСТІ	5
1.1 Характеристика предметної галузі та об'єкта дослідження	5
1.2. Аналіз літературних джерел та практичного досвіду використання ІС і технологій в предметній галузі	9
РОЗДІЛ 2. РОЗРОБКА ВИМОГ І МОДЕЛЮВАННЯ ІНФОРМАЦІЙНОЇ СИСТЕМИ/ПІДСИСТЕМ	12
2.1. Аналіз і специфікація вимог до інформаційної системи	12
2.2. Постановка та алгоритм розв'язання задачі	17
2.2.1 Характеристика задачі.....	17
2.2.3 Вихідні повідомлення.....	19
2.2.2 Вхідні повідомлення	21
2.3. Моделювання інформаційної підсистеми	22
РОЗДІЛ 3. ПРОЕКТУВАННЯ ТА РЕАЛІЗАЦІЯ КОМПОНЕНТІВ ПІДСИСТЕМИ	27
3.1. Інформаційне забезпечення	27
3.1.1 Загальна характеристика	27
3.1.2 Організація збору і передавання первинної інформації	29
3.1.3 Побудова системи класифікації та кодування	31
3.1.4 Структура інформаційних масивів	33
3.1.5 Вибір СКБД	34
3.1.6 Інфологічна модель бази (сховища) даних	36
3.2. Технічне забезпечення	38
3.2.1 Загальні положення та схема автоматизації.	38
3.2.2 Структура комплексу технічних засобів.	40
3.2.3 Опис автоматизованого робочого місця	44
3.3. Програмне забезпечення	45
3.3.1 Структура програмного забезпечення.....	46
3.3.2 Системне програмне забезпечення.....	47
3.3.3 Прикладне програмне забезпечення	50
3.3.4 Програмна документація.....	53
3.4. Результати реалізації інформаційної підсистеми	59
ВИСНОВОК	67
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ	68
ДОДАТКИ	69

ВСТУП

У сучасному цифровому часі, коли технології стають не тільки нашими помічниками, але й співробітниками у розвитку особистості, питання моніторингу та підтримки читацьких звичок виявляються критичними. Дана тема дослідження, що стосується застосунків з моніторингу читання книг, вибрана на фоні необхідності підтримки і популяризації читацької активності, що стає важливим аспектом освіти та розвитку суспільства в цілому.

Обґрунтування вибору теми дослідження: Інформаційна перевантаженість та постійний стрес сучасного життя приводять до того, що люди звертаються до читання як до способу релаксації, розвитку та збагачення світогляду. Проте, в умовах зростання обсягу інформації стає складніше відстежувати власний прогрес читання та знаходити мотивацію для продовження. Застосунок з моніторингу читання книг вирішує цю проблему, забезпечуючи користувачам зручні інструменти для стеження та аналізу їхньої читацької активності.

Предметна галузь цього дослідження охоплює розробку та впровадження програмного забезпечення, що дозволяє користувачам вести облік прочитаних книг, встановлювати цілі, аналізувати статистику та отримувати рекомендації щодо подальшого розвитку читацьких навичок.

Об'єктом дослідження є сам застосунок з моніторингу читання книг, який має бути створений, розроблений та впроваджений для вирішення вказаних вище завдань.

Метою даного проекту є створення функціональної та зручної системи, яка допоможе користувачам ефективно організовувати свою читацьку діяльність, відстежувати прогрес та збагачувати читацький досвід.

Основні завдання проектування включають:

Розробку інтерфейсу застосунку, що забезпечує зручний доступ до усіх функцій. Імплементацію механізмів моніторингу та аналізу читацької активності.

Розробку системи рекомендацій та персоналізованих порад для користувачів.

Тестування та оптимізацію застосунку для забезпечення стабільної та ефективної роботи.

Актуальність даної теми полягає в необхідності підтримки та стимулювання читацької активності в умовах цифрової епохи. Застосунок з моніторингу читання книг допоможе зробити читання більш організованим, мотивуючим та насиченим. Такий застосунок є важливим інструментом для розвитку освітнього та культурного середовища.

Сучасний рівень та призначення теми кваліфікаційного бакалаврського проекту:

На сучасному етапі розвитку інформаційних технологій застосунки з моніторингу читання книг вже використовуються, але існують можливості для подальшого вдосконалення та розширення їхніх можливостей. Даний кваліфікаційний бакалаврський проект спрямований на створення нового та вдосконалення існуючого застосунку для забезпечення кращої якості обслуговування користувачів та сприяння розвитку читацьких навичок.

РОЗДІЛ 1. ХАРАКТЕРИСТИКА ТА АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ

1.1 Характеристика предметної галузі та об'єкта дослідження

Предметна галузь цього дослідження охоплює застосунки з моніторингу читання книг, які є частиною ширшої галузі інформаційних технологій у сфері освіти та саморозвитку. Ця галузь зосереджена на розробці програмного забезпечення, яке сприяє підвищенню ефективності освітнього процесу, розвитку навичок самоорганізації та мотивації до читання.

Економічний зміст задач у цій галузі полягає в оптимізації процесу читання, зменшенні часу на пошук відповідних книг та забезпеченні постійного прогресу у читанні. Це досягається за допомогою функцій моніторингу, аналізу читацької активності та надання персоналізованих рекомендацій. Впровадження таких застосунків може сприяти підвищенню продуктивності користувачів, що є важливим як у освітніх, так і в професійних контекстах.[1]

Нижче наведено перелік об'єктів предметної галузі.

Освітні заклади: школи, коледжі університети. В цих установах застосунки для моніторингу читання можуть використовуватися для підвищення ефективності навчального процесу.

Незалежні користувачі: індивідуальні читачі, книжкові клуби.

Незалежні користувачі можуть використовувати ці застосунки для особистого розвитку, організації читацької діяльності та взаємодії з іншими читачами.

У предметній галузі застосунків для моніторингу читання книг основними проблемними ситуаціями є:

Низька мотивація до читання: Багато користувачів відчують труднощі з підтримкою регулярного читання через брак мотивації. Ціль вирішення – підвищити мотивацію користувачів через гейміфікацію, відстеження прогресу та надання персоналізованих рекомендацій.

Складність у відстеженні прогресу читання: Користувачі часто не мають зручних інструментів для відстеження свого прогресу в читанні. Ціль вирішення

– створення зручних функцій для реєстрації прочитаних сторінок, вимірювання часу читання та надання детальної статистики.

Персоналізація досвіду користувача: Не всі застосунки можуть надавати персоналізовані рекомендації, що знижує якість користувацького досвіду. Ціль вирішення – впровадження алгоритмів машинного навчання для аналізу вподобань користувачів та надання персоналізованих рекомендацій.

Особи, що приймають рішення у цій галузі, включають:

Розробники програмного забезпечення: Відповідальні за технічну реалізацію функцій та інтеграцію з іншими системами.

Менеджери проектів: Координують розробку та впровадження, визначають пріоритети та стежать за виконанням завдань.

Аналітики даних: Аналізують читацьку активність користувачів для покращення функцій та персоналізації.

Дизайнери користувацького інтерфейсу: Забезпечують зручність та інтуїтивність взаємодії користувачів із застосунком.

Кінцеві користувачі: Їхні відгуки та побажання враховуються для покращення застосунку.

Принципи узгодження індивідуальних переваг:

Комунікація: Регулярні зустрічі та обговорення між різними групами для виявлення пріоритетів та очікувань.

Компромід: Пошук взаємовигідних рішень, які задовольняють основні вимоги всіх зацікавлених сторін.

Фокус на кінцевого користувача: Орієнтація на створення цінності для кінцевого користувача, що є спільною метою для всіх груп.

Прозорість: Відкрите обговорення можливих рішень та їхніх наслідків для всіх зацікавлених сторін.

Адаптивність: Гнучкість у підходах та готовність змінювати стратегії відповідно до отриманих результатів та відгуків.

Прийняття рішень у предметній галузі застосунків для моніторингу читання книг залежить від ряду факторів, які можуть значно впливати на

характер і ступінь прийняття рішень. Нижче наведені основні фактори та їхній вплив:

Технологічні фактори:

Рівень розвитку технологій: Впливає на можливості реалізації складних функцій, таких як інтеграція з іншими платформами, використання алгоритмів машинного навчання для персоналізації рекомендацій тощо.

Сумісність з різними пристроями: Необхідність забезпечити роботу застосунку на різних операційних системах (iOS, Android) і типах пристроїв (смартфони, планшети).

Економічні фактори:

Бюджет проекту: Обмеження фінансових ресурсів можуть впливати на можливість реалізації деяких функцій або використання певних технологій.

Вартість розробки та підтримки: Необхідність оцінки витрат на розробку, впровадження та підтримку системи в довгостроковій перспективі.

Соціальні фактори:

Потреби користувачів: Розуміння потреб та очікувань кінцевих користувачів є ключовим для прийняття рішень щодо функціональності та дизайну застосунку.

Тенденції у читанні: Зміни у звичках читання (наприклад, зростання популярності електронних книг) можуть впливати на прийняття рішень щодо підтримуваних форматів та функцій.

Правові фактори:

Законодавчі вимоги: Необхідність дотримання законодавства щодо захисту персональних даних (наприклад, GDPR) та авторських прав.

Ліцензійні обмеження: Вимоги щодо використання ліцензійного контенту та програмного забезпечення.

У процесі розробки застосунку для моніторингу читання книг важливо розуміти, як прийняті раніше рішення впливають на поточні та майбутні рішення, а також як ці рішення взаємопов'язані. Взаємозв'язки рішень можна поділити на кілька основних категорій:

Технічні рішення:

Вибір платформи та технологій: Рішення про використання певних технологій (наприклад, мови програмування, фреймворки, бази даних) на початковому етапі визначають можливості та обмеження для подальшої розробки. Наприклад, вибір мови програмування може впливати на продуктивність, сумісність з іншими системами та доступність бібліотек.

Архітектура системи: Визначена на початковому етапі архітектура (наприклад, мікросервіси або монолітна архітектура) впливає на масштабованість, гнучкість та можливість інтеграції з іншими системами. Це рішення також впливає на майбутні можливості розширення функціональності.[1]

Функціональні рішення:

Визначення ключових функцій: Рішення про основні функціональні можливості застосунку, такі як реєстрація користувачів, додавання книг, початок читання, оцінка та коментування книг, визначають основний обсяг роботи та подальші функціональні вимоги. Наприклад, якщо на ранньому етапі було вирішено додати функцію коментування книг, це рішення впливатиме на структуру бази даних, розробку інтерфейсу користувача та механізми модерації коментарів.

Інтеграція з зовнішніми сервісами: Рішення про інтеграцію з іншими сервісами (наприклад, платіжні системи, соціальні мережі) визначають необхідність реалізації API, забезпечення безпеки даних та підтримку сумісності з версіями зовнішніх сервісів.

Організаційні рішення:

Розподіл ролей та обов'язків: Визначення ролей у команді розробників (наприклад, фронтенд та бекенд розробники, тестувальники, менеджери проектів) впливає на організацію процесу розробки та комунікацію між членами команди. Це рішення також визначає ефективність роботи команди та швидкість реалізації проекту.

Стратегічні рішення:

Цільова аудиторія: Рішення про визначення цільової аудиторії (наприклад, студентів, професіоналів, шанувальників літератури) впливає на функціональність та дизайн застосунку. Це рішення визначає, які функції будуть пріоритетними, а які можуть бути реалізовані на більш пізніх етапах.

Монетизація: Вибір стратегії монетизації (наприклад, платні підписки, реклама, покупка окремих функцій) визначає фінансову модель проекту та впливає на функціональність застосунку. Це рішення може також впливати на досвід користувачів та їхню задоволеність продуктом.

Рішення, прийняті на ранніх етапах, створюють основу для подальших рішень і визначають можливі напрями розвитку проекту:

Забезпечення сумісності та інтеграції: Ранні рішення про вибір технологій та архітектури впливають на можливості інтеграції з новими сервісами та системами в майбутньому.

Можливість масштабування: Архітектурні та технологічні рішення визначають, наскільки легко буде масштабувати систему для підтримки більшої кількості користувачів та обсягу даних.

Гнучкість розвитку: Рішення щодо методології розробки та організації команди впливають на здатність проекту швидко адаптуватися до змін та реалізувати нові функції.

Фінансова стабільність: Рішення про стратегію монетизації впливають на фінансову стабільність проекту та можливості подальшого розвитку.

1.2. Аналіз літературних джерел та практичного досвіду використання ІС і технологій в предметній галузі

У сучасному світі застосунки для моніторингу читання книг набувають все більшої популярності. Вони дозволяють користувачам не лише стежити за своєю активністю у читанні, але й взаємодіяти з іншими читачами, ділитися відгуками та рекомендаціями. У цьому підрозділі розглянемо сучасний стан розвитку таких систем, проаналізуємо існуючі теорії та підходи, а також критично оцінюємо відповідні літературні джерела, включаючи зарубіжні дослідження.

Різноманітні дослідження та розробки у сфері інформаційних систем (ІС) та технологій показують, що застосунки для моніторингу читання книг мають значний потенціал для вдосконалення користувацького досвіду читання. Ці системи дозволяють ефективно організовувати читання, аналізувати прогрес та стимулювати інтерес до читання.

Згідно з дослідженням А. Smith (2020), сучасні застосунки для моніторингу читання книг забезпечують користувачам інтуїтивний інтерфейс та потужні функціональні можливості для відстеження прочитаних сторінок, часу, проведеного за читанням, та обміну враженнями про книги. Інше дослідження В. Johnson (2019) підкреслює важливість соціального аспекту таких застосунків, де користувачі можуть ділитися рекомендаціями та створювати спільноти за інтересами.[1]

Критичний огляд літературних джерел

Smith, A. (2020). "Innovations in Reading Applications". Ця книга надає детальний огляд сучасних технологій та інновацій у застосунках для моніторингу читання. Автор аналізує функціональність, інтерфейси та користувацький досвід, наголошуючи на важливості інтеграції соціальних функцій.

Johnson, B. (2019). "Social Aspects of Reading Monitoring Apps". Дослідження акцентує увагу на соціальних взаємодіях користувачів застосунків для моніторингу читання книг. Автор вказує на важливість комунікації між користувачами для підвищення мотивації до читання.

Williams, C., & Brown, D. (2018). "User Experience in Digital Reading". У цій роботі розглядаються аспекти користувацького досвіду у цифрових читальних системах. Автори аналізують зручність інтерфейсів та важливість адаптації під різні пристрої.

Kumar, R. (2021). "Future Trends in Reading Applications". Ця книга досліджує майбутні тенденції у розвитку застосунків для читання, включаючи використання штучного інтелекту для персоналізації контенту та аналізу поведінки користувачів.

Garcia, M. (2017). "Digital Transformation in Reading Habits". Автор обговорює вплив цифрових технологій на звички читачів, зазначаючи, що застосунки для моніторингу читання сприяють зростанню інтересу до книг завдяки інтерактивним функціям.

Аналізуючи літературні джерела, можна зробити висновок, що основні аспекти теми достатньо висвітлені, проте є деякі проблемні питання, що потребують подальшого дослідження:

Персоналізація контенту. Необхідно дослідити методи персоналізації рекомендацій на основі індивідуальних уподобань користувачів.

Аналіз даних. Важливо розробити ефективні алгоритми для аналізу поведінки користувачів та виявлення патернів у їхніх читальних звичках.

Інтеграція з іншими системами. Потребує подальшого дослідження питання інтеграції застосунків для моніторингу читання з іншими інформаційними системами, наприклад, бібліотеками або освітніми платформами.

РОЗДІЛ 2. РОЗРОБКА ВИМОГ І МОДЕЛЮВАННЯ ІНФОРМАЦІЙНОЇ СИСТЕМИ

2.1. Аналіз і специфікація вимог до інформаційної системи

Діаграма бізнес-вимог (Business Requirements Diagram) - це інструмент в системному аналізі, який використовується для візуалізації та організації бізнес-вимог до системи або проекту. Вона дозволяє чітко визначити та відобразити ключові функції, процеси та вимоги, які повинна виконувати система з погляду бізнес-користувачів. [7]

Діаграма бізнес-вимог є важливим інструментом при аналізі та уточненні вимог до системи. Ось кілька причин, чому вона є корисною:

Уточнення бізнес-потреб:

Діаграма допомагає розкрити основні потреби бізнесу, визначити цілі та завдання, які система повинна вирішувати.

Визначення функцій системи:

Дозволяє ідентифікувати основні функції та можливості, які повинна мати система для задоволення потреб бізнесу.

Розуміння взаємодії з іншими системами:

Допомагає визначити, як система взаємодіє з іншими системами, сервісами або компонентами, які можуть бути використані.

Визначення обмежень та умов:

Дозволяє визначити обмеження та умови, які повинні бути враховані при розробці системи.

Діаграма бізнес-вимог для застосунку з моніторингу читання книг описує основні функціональні вимоги та процеси, які повинна виконувати система. Основні елементи цієї діаграми включають:

Реєстрація користувача

Авторизація: Користувач вводить свої облікові дані або створює новий обліковий запис, якщо він новий користувач. Ця функція дозволяє користувачам мати доступ до особистих налаштувань та зберігання своїх даних.

Створення облікового запису: Користувач заповнює реєстраційну форму з особистими даними (ім'я, електронна адреса тощо) та отримує доступ до системи.

Додавання книги

Пошук книги: Користувач може шукати книги за назвою, автором або категорією, що дозволяє зручно знаходити потрібні матеріали.

Додавання книги до облікового запису: Користувач обирає книгу з результатів пошуку та додає її до свого списку книг. Ця функція спрощує управління читанням та надає зручний доступ до книг.

Читання книги

Початок читання: Користувач вибирає книгу зі свого списку та починає читання, вказавши початкову сторінку. Це дозволяє користувачеві зручно читати книги в онлайн форматі.

Завершення читання: Користувач вказує кінцеву сторінку після завершення читання, щоб система враховувала його прогрес та відображала відповідну статистику.

Оцінка та коментарі

Додавання оцінки: Користувач може оцінювати книгу за шкалою від 1 до 5 зірочок, щоб інші користувачі могли зрозуміти якість матеріалу.

Додавання коментаря: Користувач може залишити коментар або рецензію до книги для інших користувачів, ділившись своїми думками та враженнями.

Статистика читання

Час читання: Система відстежує час, який користувач витрачає на читання книги, щоб надати інформацію про ефективність читання.

Прогрес читання: Система відображає прогрес читання книги у відсотках, щоб користувачі могли відслідковувати свій прогрес та визначати завершеність книги.

В процесі розробки інформаційної системи для відстеження прогресу читання книг важливим етапом є визначення бізнес-вимог, які відображають основні потреби та очікування користувачів. На Рисунку 2.1 – Діаграма бізнес-

вимог представлена структурована схема, що демонструє ключові вимоги до системи



Рисунок 2.1 – Діаграма бізнес-вимог

Функціональні вимоги до системи - це конкретні задачі, дії або функції, які система повинна виконувати для задоволення потреб користувачів та досягнення цілей проекту. Це можуть бути функції, що вимагають від системи здійснення певних операцій, обробки даних, забезпечення взаємодії з користувачами тощо [7].

Діаграма функціональних вимог для застосунку з моніторингу читання книг корисна з кількох позицій:

- Дозволяє уточнити функціональні вимоги до системи, визначаючи конкретні функції та дії, які має виконувати програмне забезпечення.
- Дозволяє зрозуміти, які функції важливі для користувачів застосунку та як система має взаємодіяти з ними.
- Допомогає визначити пріоритетність функцій та вимог, що є важливим для впровадження та розвитку системи.

У діаграмі функціональних вимог для застосунку з моніторингу читання книг є такі елементи:

Аутентифікація та реєстрація

Вхід в систему

Тестовий сценарій: Користувач вводить свої дані для входу в систему.

Очікуваний результат: Успішний вхід в систему або повідомлення про помилку.

Реєстрація користувача

Тестовий сценарій: Новий користувач заповнює форму реєстрації з обов'язковими полями.

Очікуваний результат: Успішна реєстрація або повідомлення про помилку.

Операції з книгами

Додавання книги

Тестовий сценарій: Користувач додає нову книгу до свого списку читання.

Очікуваний результат: Книга додається до списку користувача.

Видалення книги

Тестовий сценарій: Користувач видаляє книгу зі свого списку читання.

Очікуваний результат: Книга видаляється зі списку користувача.

Оцінка книги

Тестовий сценарій: Користувач оцінює прочитану книгу.

Очікуваний результат: Оцінка зберігається в системі та відображається користувачу.

Коментар до книги

Тестовий сценарій: Користувач залишає коментар до прочитаної книги.

Очікуваний результат: Коментар зберігається в системі та відображається користувачу.

На Рисунку 2.2 – Функціональні вимоги до ІС представлено деталізовану діаграму, яка описує основні функціональні вимоги до інформаційної системи для відстеження прогресу читання книг.



Рисунок 2.2 – Функціональні вимоги до ІС

Діаграма нефункціональних вимог - це інструмент аналізу та документування вимог до системи, які не визначають функціональні аспекти

програмного забезпечення, але визначають необхідність певних якостей, характеристик або обмежень системи. Основні аспекти діаграми нефункціональних вимог включають такі пункти:

Інтерфейс користувача:

- Ергономіка і зручність використання інтерфейсу для різних категорій користувачів.
- Наявність інтуїтивно зрозумілих елементів управління та навігації.
- Сумісність з різними пристроями та роздільними здатностями екранів.

Надійність та безпека:

- Забезпечення стійкості системи до виникнення помилок та збоїв.
- Механізми резервного копіювання та відновлення даних.
- Захист інформації від несанкціонованого доступу та втрати.
- Моніторинг та логування подій для виявлення можливих загроз та атак.

Продуктивність:

- Швидкість відгуку системи на запити користувачів.
- Ефективне використання ресурсів (пам'яті, обчислювальної потужності).
- Масштабованість для обробки зростаючого обсягу даних та користувачів.

Відповідність стандартам:

- Дотримання вимог та нормативів, які стосуються безпеки, конфіденційності, доступності.
- Сумісність зі стандартами та протоколами в області інформаційних технологій.

На Рисунку 2.3 – Діаграма нефункціональних вимог представлена схема, що описує основні нефункціональні вимоги до інформаційної системи для відстеження прогресу читання книг.

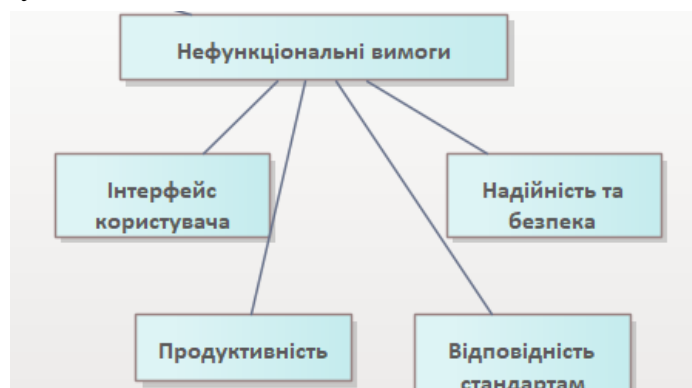


Рисунок 2.3 – Діаграма нефункціонованих вимог

2.2. Постановка та алгоритм розв'язання задачі

2.2.1 Характеристика задачі

Задача моніторингу читання книг призначена для автоматизації процесу відстеження та аналізу читання літератури користувачами. Це включає реєстрацію книг, фіксацію прогресу читання, оцінювання та коментування прочитаних книг. Основна мета – надати користувачам інструмент для організації та покращення їх читання, а також створити платформу для обміну враженнями та рекомендаціями.

Техніко-економічна сутність задачі полягає в оптимізації процесів збирання та аналізу даних про читання книг, що дозволяє знизити витрати часу та покращити якість рекомендацій. Використання ЕОМ (електронно-обчислювальних машин) забезпечує швидке та точне оброблення великих обсягів інформації, автоматизацію рутинних операцій і підвищення загальної ефективності системи.

Перелік об'єктів, за управління якими розв'язується задача:

1. Користувачі: фізичні особи, які використовують застосунок для моніторингу свого читання.
2. Книги: літературні твори, які користувачі додають до своєї бібліотеки для читання.
3. Читання: процес відстеження прогресу читання кожної книги користувачем.
4. Оцінки: числові значення, які користувачі надають книгам після їх прочитання.
5. Коментарі: текстові відгуки та враження користувачів про прочитані книги.
6. Призначення й використання вихідної інформації:
7. Статистика читання: звіти про кількість прочитаних сторінок, час, витрачений на читання, та інші показники.
8. Оцінки книг: рейтинг книг на основі оцінок користувачів.
9. Коментарі користувачів: текстові відгуки, що надають додаткову інформацію та рекомендації іншим користувачам.

Ця інформація використовується для аналізу прогресу читання, покращення рекомендаційних алгоритмів та створення спільноти навколо обговорення книг.

Задача моніторингу читання книг виконується постійно в реальному часі. Дані оновлюються при кожній взаємодії користувача із застосунком (додавання книги, оновлення статусу читання, залишення оцінки або коментаря). Вихідна інформація може генеруватися на вимогу користувача або автоматично у встановлені інтервали (наприклад, щотижневі або щомісячні звіти про прогрес читання).

Умови, за яких припиняється автоматизоване розв'язання задачі.

Автоматизоване розв'язання задачі може бути припинено в таких випадках:

- Відсутність доступу до інтернету:
- Система не зможе оновлювати дані в режимі реального часу без інтернет-з'єднання.
- Проблеми з сервером або базою даних, що обробляє інформацію про користувачів і книги.

Завершення читання книги:

- Після завершення книги користувачем автоматизований моніторинг цієї книги припиняється.
- Зв'язки даної задачі з іншими задачами

Рекомендаційні системи:

- Використання даних про читання для надання персоналізованих рекомендацій книг.

- Системи управління контентом:

- Адміністрування бази даних книг та інформації про користувачів.

Технічні засоби:

- Обробка та зберігання даних.
- Генерація звітів та статистики.
- Автоматизовані рекомендації та аналіз даних.

Персонал:

- Налаштування системи та введення початкових даних.
- Моніторинг роботи системи та вирішення технічних проблем.
- Обслуговування та оновлення бази даних книг.

Цей розподіл дозволяє забезпечити ефективну роботу системи та мінімізувати вплив людського фактору на процес прийняття рішень.

На Рисунку 2.4 – Інформаційна модель вирішення задачі застосунку з моніторингу читання представлена схема, що ілюструє структуру даних та взаємозв'язки між основними компонентами системи.

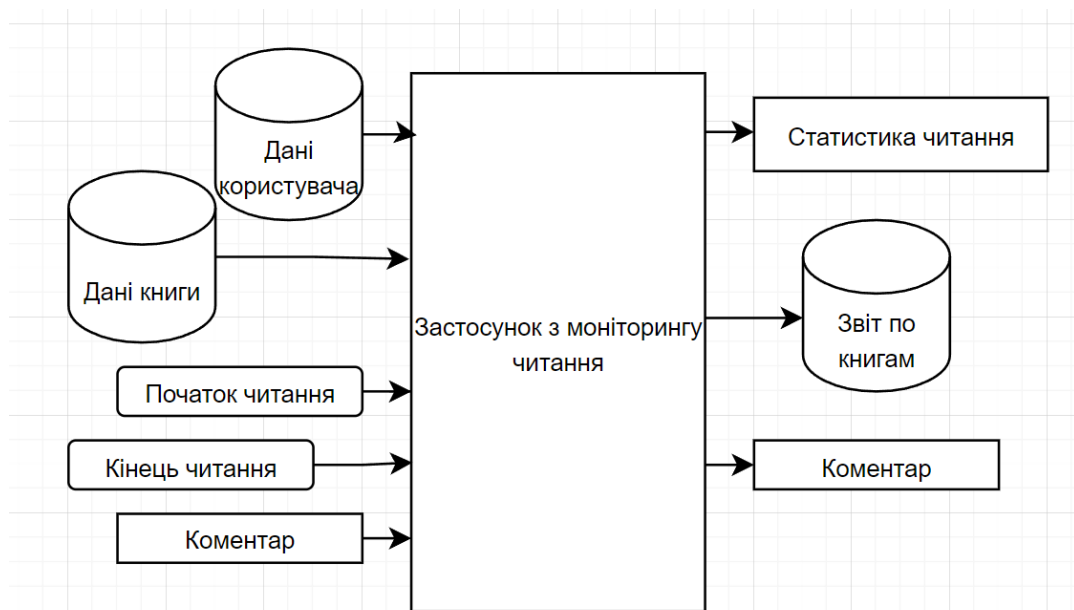


Рисунок 2.4 – Інформаційна модель вирішення задачі застосунку з моніторингу читання

2.2.3 Вихідні повідомлення

Вихідна інформація є кінцевим продуктом обробки даних в інформаційній системі. Її призначення полягає в наданні користувачам даних, які допомагають у прийнятті рішень, оцінці ефективності процесів і контролі за виконанням завдань. У випадку системи для моніторингу читання книг вихідна інформація може включати різні звіти, статистичні дані, огляди прогресу користувачів, а також коментарі та оцінки книг.

Призначення і використання вихідної інформації

Статистика читання: Використовується для аналізу активності користувачів, визначення популярності книг та оцінки загального часу, витраченого на читання.

Звіти по книгам: Включають дані про кількість прочитаних книг, середній час читання і оцінки, що дозволяє користувачам і адміністраторам системи аналізувати переваги читачів.

Коментарі та оцінки: Допмагають іншим користувачам приймати рішення щодо вибору книг на основі відгуків та оцінок.

У Таблиці 2.2.3.1 – Вихідні повідомлення представлений перелік основних повідомлень, які генерує інформаційна система для відстеження прогресу читання книг. Ці повідомлення включають різні типи інформації, що надсилається користувачам для інформування про їхній прогрес та інші важливі події.

Таблиця 2.2.3.1 – Вихідні повідомлення

№	Назва вихідного повідомлення	Ідентифікатор	Форма подання і вимоги до неї	Періодичність видання	Термін видання і допустимий час затримки	Користувачі інформації
1	Статистика читання	STAT_READ	Таблиця / графік, CSV, PDF	Щомісяця	1-й день місяця, 1 доба	Адміністратори , користувачі
2	Звіт по книгам	BOOK_REPORT	Таблиця PDF	Щоквартально	5-й день кварталу, 2 доби	Адміністратори
3	Коментарі та оцінки	REVIEWS	Текст, HTML	В режимі реального часу	Не застосовується	Користувачі

Опис структурних одиниць вихідних повідомлень

Статистика читання (STAT_READ)

Кількість прочитаних книг: Загальна кількість книг, прочитаних за період.

Середній час читання: Середній час, витрачений на читання однієї книги.

Кількість активних користувачів: Кількість користувачів, які читали книги протягом періоду.

Звіт по книгам (BOOK_REPORT)

Назва книги: Ідентифікатор книги.

Кількість прочитань: Скільки разів книгу було прочитано.

Середня оцінка: Середня оцінка книги, надана користувачами.

Коментарі та оцінки (REVIEWS)

Текст коментаря: Відгук, залишений користувачем.

Оцінка книги: Оцінка книги за шкалою від 1 до 5.

Автор коментаря: Ідентифікатор користувача, який залишив коментар.

Картка клієнта (USER_CARD)

Ім'я користувача: Ідентифікатор користувача.

Список прочитаних книг: Перелік книг, прочитаних користувачем.

Оцінки та коментарі: Оцінки та коментарі, залишені користувачем.

2.2.2 Вхідні повідомлення

Вхідні повідомлення в контексті інформаційної системи - це дані або інформація, яка надходить до системи ззовні для її обробки, збереження або використання у подальшій роботі. Вони можуть надходити від користувачів, інших систем або сенсорів, і включають різні форми введення, такі як текст, числові дані, команди, зображення, звіти тощо.

У Таблиці 2.2.2.2 – Вхідні повідомлення представлений перелік основних повідомлень, які система отримує від користувачів для обробки та зберігання інформації. Ці повідомлення включають різні типи даних, що надсилаються користувачами для оновлення їхнього прогресу читання та інших дій у системі.

Таблиця 2.2.2.2 – Вхідні повідомлення

№	Назва вхідного повідомлення	Ідентифікатор	Форма подання і вимоги до неї	Періодичність видання	Термін видання і допустимий час затримки	Користувачі інформації
1	Дані користувача	USER_DATA	Електронна	Під час реєстрації та входу	Негайно	Система моніторингу читання
2	Дані книги	BOOK_DATA	Електронна	Під час додавання нової книги	Негайно	Система моніторингу читання
3	Початок читання	START_READING	Електронна	Під час початку читання	Негайно	Система моніторингу читання
4	Закінчення читання	END_READING	Електронна	Після завершення читання	Негайно	Система моніторингу читання
5	Коментар	COMMENT_DATA	Електронна	Після додавання коментаря	Негайно	Система моніторингу читання

2.3. Моделювання інформаційної підсистеми

Моделювання поведінки системи - це процес створення моделі, яка описує те, як система поводить себе в різних ситуаціях та умовах. Цей процес включає у себе аналіз і уявлення того, як система взаємодіє з своєю оточуючою середовищем, як вона реагує на вхідні дані та події, які відбуваються у системі, і як вона змінює свій стан або надає результати відповідно до цих подій.

Діаграма прецедентів - це графічний інструмент моделювання в рамках методології UML (Unified Modeling Language), який використовується для визначення функціональних вимог до системи з точки зору її користувачів. Основна мета діаграми прецедентів - це визначення того, як користувачі взаємодіють з системою, які функції вони можуть виконувати та як система реагує на їх дії.

Застосунок з моніторингу читання має функцію для користувачів, які реєструються, додають книги, використовують таймер для вимірювання часу читання, додають оцінки та коментарі до прочитаних книг.

На Рисунку 2.5 – Діаграма прецедентів представлено основні сценарії взаємодії користувачів із системою для відстеження прогресу читання книг. Діаграма прецедентів ілюструє, як різні актори (користувачі, адміністратори) взаємодіють із системою для виконання різних завдань.

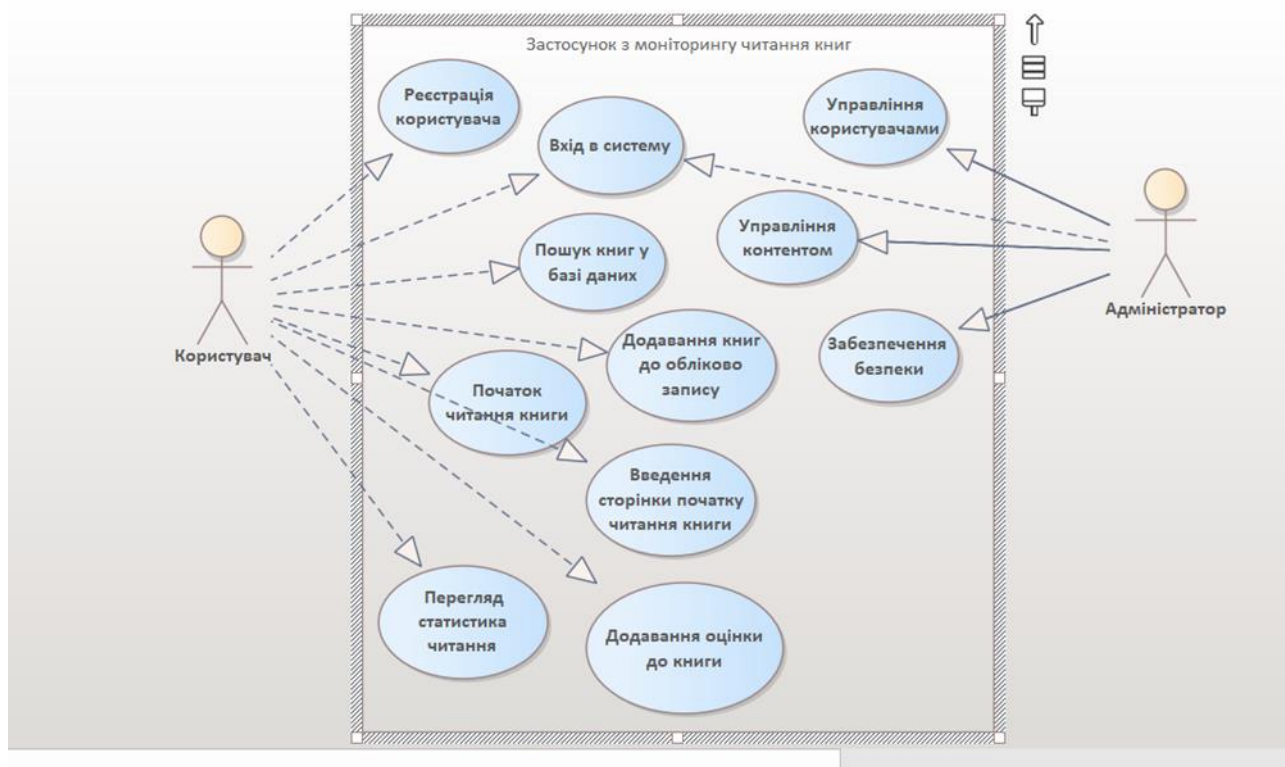


Рисунок 2.5 – Діаграма прецедентів

Реєстрація користувача: Користувач заповнює форму реєстрації, вводить необхідні особисті дані (ім'я, електронна пошта, пароль тощо). Актори: Користувач, Адміністратор
Умови виконання: Користувач має доступ до мережі Інтернет та введе правильні дані.

Вхід в систему: Користувач вводить свій логін та пароль для входу в систему. Актори: Користувач, Система. Умови виконання: Користувач має зареєстрований обліковий запис та вводить правильні дані.

Пошук книги у базі даних: Користувач виконує пошук книги за назвою або автором у базі даних системи. Актори: Користувач, Система. Умови виконання: Користувач має доступ до функції пошуку та введе правильні дані для пошуку.

Додавання книги до облікового запису: Користувач додає нову книгу до свого облікового запису, вказуючи її назву, автора тощо. Актори: Користувач, Система. Умови виконання: Користувач має права на додавання книг та вводить правильні дані.

Початок читання книги: Користувач обирає книгу для читання та починає читати. Актори: Користувач, Система. Умови виконання: Користувач має доступ до книг та вибирає книгу для читання.

Вимірювання часу читання: Система розпочинає вимірювання часу, коли користувач починає читання книги. Актори: Користувач, Система. Умови виконання: Користувач починає читання книги.

Введення номера сторінки початку: Користувач вводить номер сторінки, з якої починає читати книгу. Актори: Користувач, Система. Умови виконання: Користувач починає читання книги та вводить номер сторінки.

Введення номера сторінки кінця: Користувач вводить номер сторінки, на якій завершує читання книги. Актори: Користувач, Система. Умови виконання: Користувач завершує читання книги та вводить номер сторінки.

Перегляд статистики читання: Користувач може переглянути статистику свого читання, таку як час, кількість сторінок тощо. Актори: Користувач, Система. Умови виконання: Користувач вибирає опцію перегляду статистики читання.

Додавання оцінки до книги: Користувач може оцінити книгу, яку він прочитав. Актори: Користувач, Система. Умови виконання: Користувач має можливість оцінювати книгу та вводить відповідну оцінку.

На Рисунку 2.6 – Матриця відповідності вимог і прецедентів представлено таблицю, яка показує взаємозв'язок між вимогами до системи та прецедентами, що описують сценарії використання системи.

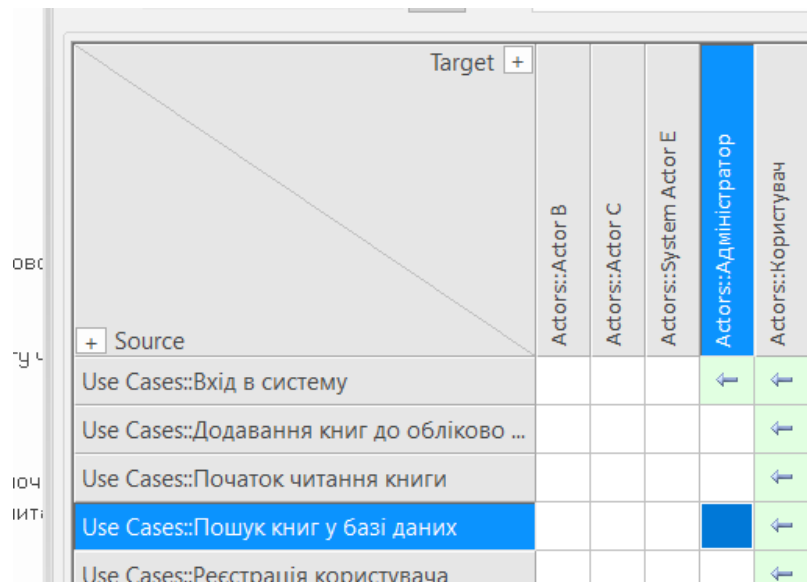


Рисунок 2.6 – Матриця відповідності вимог і прецедентів

Діаграма послідовності - це UML-діаграми, що відображають взаємодію між об'єктами в системі у вигляді послідовності повідомлень чи подій. Основна мета цієї діаграми - показати, як об'єкти спілкуються між собою та в якому порядку вони обмінюються повідомленнями або виконують дії.

На Рисунку 2.7 – Діаграма послідовностей представлено послідовність взаємодій між різними компонентами системи та користувачами для виконання певних функцій у застосунку для відстеження прогресу читання книг.

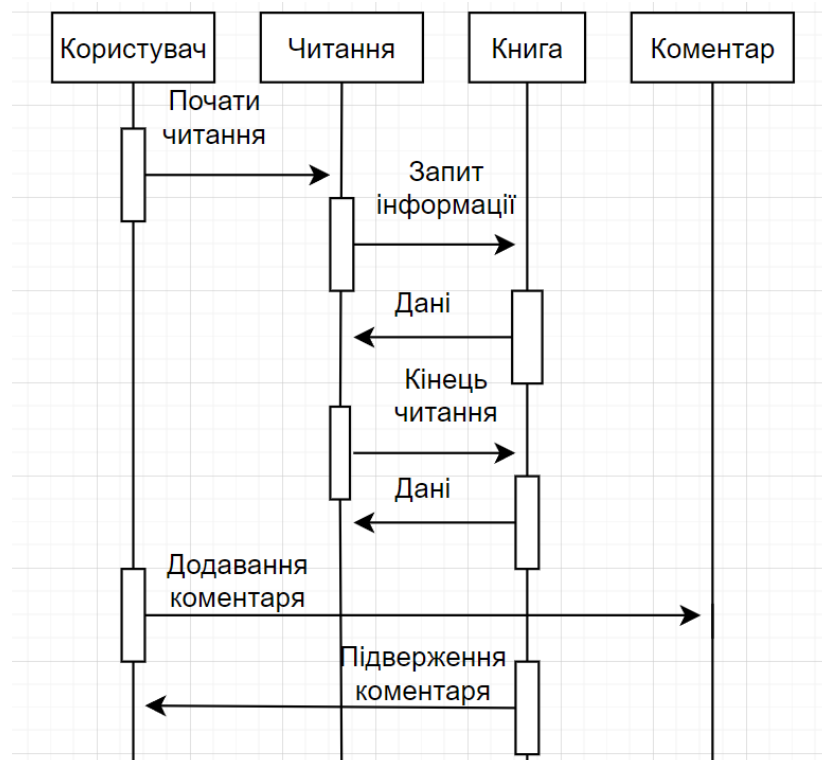


Рисунок 2.7 – Діаграма послідовностей

На цій діаграмі зображено, що користувач вибирає книгу для читання. Система запитує базу даних про цю книгу. База даних повертає системі дані про книгу. Система відображає книгу для читання користувачеві. Користувач натискає на кнопку "Почати читання". Система запускає таймер часу читання. Після завершення читання користувач натискає кнопку "Кінець читання". Система зупиняє таймер та відображає статистику читання та кінцеву сторінку книги користувачеві.

Діаграма класів є одним із видів діаграм у моделюванні систем за допомогою UML (Unified Modeling Language). Вона описує структуру системи, показуючи її класи, їхні атрибути, методи та взаємозв'язки між ними. Це основний елемент статичного моделювання, що дозволяє розробникам побачити "скелет" системи та зрозуміти, як компоненти системи взаємодіють між собою.

На Рисунку 2.8 – Діаграма класів представлена структура класів системи для відстеження прогресу читання книг. Діаграма класів вказує на основні класи об'єктів у системі та їхні взаємозв'язки.

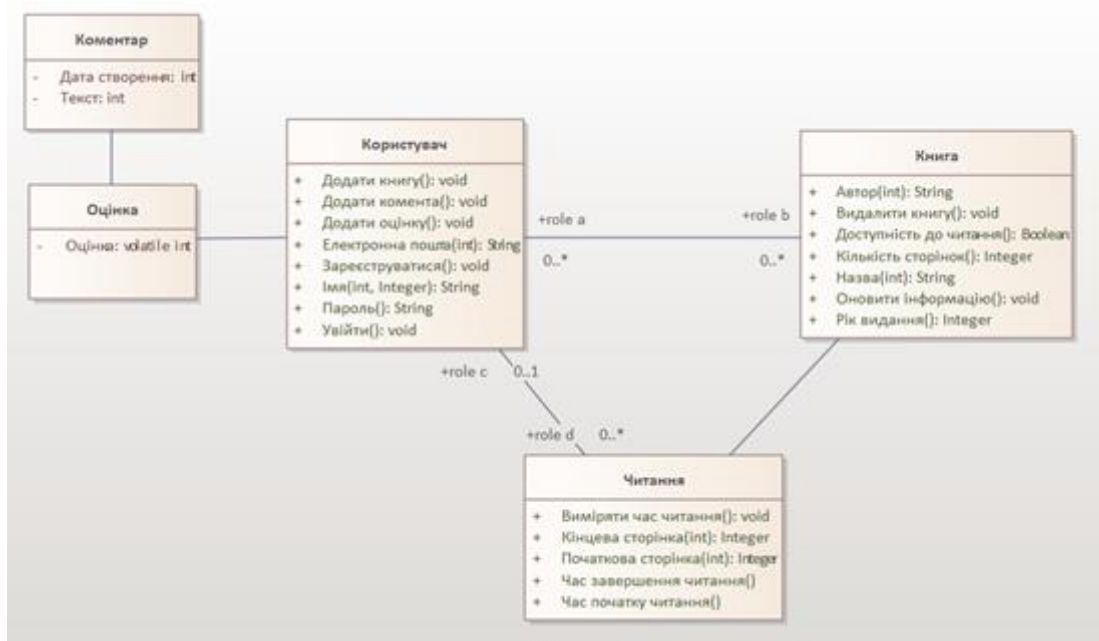


Рисунок 2.8 – Діаграма класів

Система моніторингу читання призначена для відстеження прогресу читання користувачів, зберігання інформації про прочитані книги, коментарі та оцінки. Основними користувачами системи є читачі, які хочуть відслідковувати свої успіхи, обмінюватися думками та оцінювати книги.

РОЗДІЛ 3. ПРОЕКТУВАННЯ ТА РЕАЛІЗАЦІЯ КОМПОНЕНТІВ ПІДСИСТЕМИ

3.1. Інформаційне забезпечення

3.1.1 Загальна характеристика

Інформаційне забезпечення (ІЗ) є однією з найважливіших складових частин автоматизованих інформаційних систем (ІС). Воно включає в себе сукупність методів і засобів для збору, зберігання, обробки, передачі і відображення інформації, необхідної для функціонування ІС. Для застосування з моніторингу читання книг інформаційне забезпечення відіграє вирішальну роль у забезпеченні ефективного функціонування системи.

Склад і структура інформаційного забезпечення.[5]

Інформаційне забезпечення системи моніторингу читання книг складається з наступних компонентів:

Бази даних (БД):

1. Користувачі
2. Книги
3. Сесії читання
4. Оцінки
5. Коментарі

Механізми обробки даних:

1. Модулі реєстрації і аутентифікації користувачів
2. Модулі додавання і редагування інформації про книги
3. Модулі відстеження сесій читання
4. Модулі обробки оцінок і коментарів

Принципи організації інформаційного забезпечення

Інформаційне забезпечення організовано на основі наступних принципів:

1. Цілісність даних: Забезпечення цілісності даних шляхом підтримки актуальності і точності інформації.

2. Доступність: Гарантування доступу до даних для авторизованих користувачів у будь-який час.

3. Конфіденційність: Захист даних від несанкціонованого доступу.

4. Надійність: Використання резервного копіювання та відновлення даних для забезпечення безперервності роботи системи.

5. Достовірність: Використання механізмів перевірки достовірності інформації для мінімізації помилок.[10]

Тип носія даних

Для зберігання даних використовуються сучасні реляційні бази даних, розміщені на серверних носіях з високим рівнем надійності. Це забезпечує швидкий доступ до даних, масштабованість та відмовостійкість системи.

Обґрунтування вибору СКБД та основні її характеристики

Для реалізації інформаційного забезпечення системи моніторингу читання книг обрана система керування базами даних (СКБД) PostgreSQL. Вибір обумовлений наступними характеристиками:[4]

Підтримка ACID-транзакцій: Забезпечення надійності і цілісності даних.

Масштабованість: Можливість роботи з великими обсягами даних і підтримка розподілених систем.

Безпека: Механізми аутентифікації і авторизації користувачів, шифрування даних.

Гнучкість: Підтримка складних запитів і розширень.

Відкритий код: Можливість модифікації і налаштування під конкретні вимоги.

Методи контролю інформації

Контроль інформації здійснюється за допомогою наступних методів:

Валідація даних: Перевірка даних на етапі введення.

Верифікація даних: Перевірка достовірності і актуальності даних шляхом порівняння з існуючими записами.

Аудит логів: Відстеження дій користувачів і змін в базі даних для запобігання несанкціонованому доступу.

Вимоги до надійності і достовірності інформації

Надійність і достовірність інформації забезпечуються наступними заходами:

Резервне копіювання: Регулярне створення резервних копій бази даних.

Відновлення після збоїв: Механізми швидкого відновлення системи після аварійних ситуацій.

Моніторинг системи: Постійний моніторинг стану бази даних і системи в цілому.

Механізми шифрування: Використання сучасних алгоритмів шифрування для захисту конфіденційних даних.

Загальна схема інформаційного забезпечення

Загальна схема ІЗ включає наступні елементи:

Сервер бази даних: PostgreSQL сервер, що зберігає всі дані системи.

Клієнтські додатки: Інтерфейси користувачів для взаємодії з системою.

Мережеві компоненти: Забезпечують зв'язок між клієнтськими додатками і сервером бази даних.

Системи резервного копіювання: Механізми для створення і зберігання резервних копій даних.

Системи моніторингу і аудиту: Інструменти для відстеження і контролю дій в системі.

Перелік конкретних елементів, що використовуються при функціонуванні ІС:

Таблиці бази даних для зберігання інформації про користувачів, книги, сесії читання, оцінки і коментарі.

Форми і інтерфейси для введення і редагування даних.

Звіти і аналітичні інструменти для оцінки ефективності системи.

Системи безпеки і шифрування для захисту даних.[4]

3.1.2 Організація збору і передавання первинної інформації

Організація збору і передавання первинної інформації є важливим аспектом функціонування системи моніторингу читання книг. Цей процес

забезпечує своєчасне і коректне надходження необхідних даних до системи для подальшої обробки та аналізу.

Перелік джерел і носіїв інформації

Джерела інформації для системи моніторингу читання книг можуть бути різноманітними. Основними джерелами є:

Користувачі системи:

Кінцеві користувачі (читачі) вводять дані про початок і закінчення сесій читання, оцінюють книги, залишають коментарі.

Адміністратори вводять та редагують дані про книги, управляють користувачами та контентом.

Зовнішні джерела:

Онлайн-каталоги книг (наприклад, Open Library, Google Books API) забезпечують автоматичне завантаження інформації про книги.

Соціальні мережі та інші веб-ресурси можуть бути джерелами додаткової інформації про книги та рецензії.

Внутрішні джерела:

База даних системи зберігає всю актуальну інформацію про книги, користувачів, сесії читання, оцінки та коментарі.

Опис загальних вимог до організації збору і передання інформації

Збір і передавання інформації повинні відповідати наступним вимогам:

Своєчасність: Інформація повинна надходити в систему без затримок, щоб забезпечити актуальність даних.

Достовірність: Дані повинні бути точними і надійними, щоб забезпечити коректну обробку та аналіз.

Безперервність: Процес збору даних повинен бути безперервним і автоматизованим, щоб мінімізувати вплив людського фактора.

Безпека: Забезпечення захисту даних під час передачі і зберігання.

Опис вхідних повідомлень:

Дані про користувача: Включають інформацію про реєстрацію, авторизацію, профіль користувача. Ця інформація вводиться користувачами через веб-інтерфейс.

Інформація про книгу: Включає назву, автора, жанр, опис, обкладинку книги. Дані можуть бути автоматично завантажені з онлайн-каталогів книг або введені вручну адміністраторами.

Дані про сесію читання: Включають дату і час початку та завершення читання, кількість прочитаних сторінок. Ця інформація вводиться користувачами під час читання.

Коментарі: Включають рейтинги книг і текстові відгуки, введені користувачами після завершення читання книги.

Оновлення даних про книги: Адміністратори регулярно оновлюють інформацію про книги, щоб забезпечити актуальність бази даних.

3.1.3 Побудова системи класифікації та кодування

Основними об'єктами класифікації у нашому застосунку є: книги, користувачі, сесії читання та коментарі.

Класифікація книг здійснюється за наступними параметрами: автор, рік видання.

Класифікація користувачів базується на: типі користувача (звичайний користувач, адміністратор), рівні активності (наприклад, кількість прочитаних книг, залишених коментарів)

Класифікація сесій читання включає: тривалість сесії, прогрес читання (відсоток прочитаної книги)

Класифікація коментарів охоплює: тема коментаря (зміст, стиль, сюжет), найменування використовуваних класифікаторів і кодів.

Для кожного об'єкта застосовується відповідний класифікатор та код, що дозволяє унікально ідентифікувати кожен елемент. Основні класифікатори включають:

Класифікатор книг (Book Classification, BC)

Класифікатор користувачів (User Classification, UC)

Класифікатор сесій читання (Reading Session Classification, RSC)

Класифікатор оцінок та коментарів (Review Classification, RC)

Методи кодування об'єктів класифікації

Метод кодування: Серійно-порядковий

Структура коду: [Жанр]-[Автор]-[Рік]-[Номер]

Приклад коду: FIC-ORW-1949-001 (Жанр: Fiction, Автор: George Orwell, Рік: 1949, Номер: 001)

Користувачі (UC)

Метод кодування: Порядковий

Структура коду: [Тип користувача]-[Номер]

Приклад коду: 01-1001 (Тип користувача: Звичайний користувач, Номер: 1001)

Сесії читання (RSC)

Метод кодування: Паралельний

Структура коду: [Користувач]-[Дата]-[Сесія]

Приклад коду: 1001-20230515-01 (Користувач: 1001, Дата: 15 травня 2023 року, Сесія: 01)

Оцінки та коментарі (RC)

Метод кодування: Послідовний

Структура коду: [Користувач]-[Книга]-[Рейтинг]-[Коментар]

Приклад коду: 1001-001-5-01 (Користувач: 1001, Книга: 001, Рейтинг: 5, Коментар: 01)

Структура і довжина коду

Структура коду:

Жанр (Genre): Три літери (FIC для Fiction, NF для Non-Fiction, та ін.)

Автор (Author): Три літери від прізвища автора

Рік (Year): Чотири цифри

Номер (Number): Три цифри

Тип користувача (User Type): Дві цифри (01 для звичайного користувача, 02 для адміністратора)

Дата (Date): Вісім цифр (YYYYMMDD)

Сесія (Session): Дві цифри

Рейтинг (Rating): Одна цифра (від 1 до 5)

Коментар (Comment): Дві цифри

Довжина коду:

Код книги: До 16 символів (наприклад, FIC-ORW-1949-001)

Код користувача: До 6 символів (наприклад, 01-1001)

Код сесії читання: До 18 символів (наприклад, 1001-20230515-01)

Код оцінки та коментаря: До 17 символів (наприклад, 1001-001-5-01)

Приклади класифікаторів

Класифікатор книг (Book Classification, BC):[3]

3.1.4 Структура інформаційних масивів

Інформаційні масиви — це структуровані набори даних, які зберігаються в пам'яті комп'ютера та використовуються для обробки і зберігання інформації в інформаційних системах. Інформаційні масиви можуть включати різноманітні типи даних, такі як текстові, числові, логічні, мультимедійні дані тощо. Вони є основою для ефективної організації, зберігання та обробки великих обсягів інформації.

Інформаційні масиви складаються з окремих записів, кожен з яких містить набір полів. Поля визначаються типами даних і представляють різні аспекти інформації, що зберігається в масиві. Масиви можуть бути організовані різними способами, такими як послідовна, індексна або хеш-організація, що залежить від вимог до швидкості доступу та обробки даних.

При розробці інформаційної системи для відстеження прогресу читання книг було враховано різноманітні типи даних, що зберігаються у системі. Для зручності керування та організації цих даних були визначені та описані різні масиви. Ось їхні описи в таблиці 3.1

Таблиця 3.1 – Опис масивів

Найменування масиву	Ідентифікатор у програмі	Умове позначення у формулах	Формат					Логічні чи семантичні зв'язки
				Первинний/вторинний ключ	Умове позначення	Обов'язкове поле	Індексне поле	
Книги	BOOKS	Книги	Текст	PK	INDEX BOOKS	Так	ІДД	Немає
Користувачі	USERS	Користувачі	Текст	PK	INDEX USERS	Так	ІДД	Немає
Сесії читання	SESSIONS	Сесії читання	Текст	PK	INDEX_SESSIONS	Так	ІДД	Немає
Коментарі	REVIEWS	Коментарі	Текст	PK	INDEX REVIEWS	Так	ІНД	Немає

Структура інформаційних масивів, яка описана вище, дозволяє організувати зберігання даних у застосунку з моніторингу читання книг. Вибір методів організації, ключів упорядкування та типів носіїв інформації обґрунтовано потребою забезпечити високу ефективність доступу до даних та їх обробки. Кожен масив має свій ідентифікатор і ключі упорядкування, що забезпечує швидкий пошук та сортування інформації, а також інтеграцію з іншими масивами та системами.[4]

3.1.5 Вибір СКБД

Вибір системи керування базами даних (СКБД) є критично важливим етапом розробки інформаційної системи, оскільки від цього залежить ефективність зберігання, доступу та обробки даних. Для застосунку з моніторингу читання книг вибір СКБД обґрунтовується наступними критеріями:

Продуктивність: СКБД повинна забезпечувати високу швидкість обробки запитів та масштабованість для роботи з великими обсягами даних.[4]

Надійність і відмовостійкість: СКБД повинна забезпечувати надійне зберігання даних з можливістю відновлення після збоїв.

Безпека даних: СКБД повинна підтримувати механізми контролю доступу та захисту даних від несанкціонованого доступу.

Підтримка різних типів даних: СКБД повинна дозволяти зберігати та обробляти різні типи даних, такі як текстові, числові, мультимедійні тощо.

Зручність використання: СКБД повинна мати зручні інструменти для адміністрування, розробки та оптимізації запитів.

Інтеграція: СКБД повинна легко інтегруватися з іншими системами та програмними компонентами, які використовуються в проекті.

Обґрунтування вибору СКБД

Для застосунку з моніторингу читання книг було обрано СКБД MySQL з наступних причин:

Продуктивність і масштабованість: MySQL забезпечує високу швидкість обробки запитів завдяки оптимізованому механізму індексації та підтримці кластеризації для масштабування системи.[12]

Надійність і відмовостійкість: MySQL має вбудовані механізми реплікації та резервного копіювання, що забезпечують високий рівень надійності та відмовостійкості.

Безпека даних: MySQL підтримує розширені механізми контролю доступу, шифрування даних та ведення журналу аудиту, що забезпечує високий рівень безпеки даних.

Підтримка різних типів даних: MySQL дозволяє зберігати різноманітні типи даних, включаючи текстові, числові, датовані та бінарні дані, що робить її універсальною для різних застосунків.

Зручність використання: MySQL має зручні інструменти адміністрування, такі як MySQL Workbench, що дозволяють легко керувати базою даних, створювати та оптимізувати запити.

Інтеграція: MySQL підтримує різні API та драйвери для інтеграції з різними мовами програмування та середовищами розробки, що спрощує інтеграцію з іншими компонентами системи.

Основні характеристики MySQL

1. Тип СКБД: Реляційна база даних.
2. Архітектура: Клієнт-сервер.
3. Механізми зберігання: InnoDB (за замовчуванням), MyISAM та інші.
4. Інструменти адміністрування: MySQL Workbench, phpMyAdmin.
5. Підтримка транзакцій: Так (InnoDB).
6. Інтерфейси доступу: SQL, API для C, C++, Java, PHP, Python та інших мов.[12]

Вибір MySQL як СКБД для застосунку з моніторингу читання книг обґрунтований її високою продуктивністю, надійністю, безпекою, зручністю

використання та широкими можливостями інтеграції. Ці характеристики дозволяють створити ефективну та масштабовану інформаційну систему, яка відповідає всім вимогам проекту.[6]

3.1.6 Інфологічна модель бази (сховища) даних

Інфологічна модель бази даних для застосунку з моніторингу читання книг відображає фізичну структуру бази даних у середовищі вибраної СКБД. У даному випадку використовується MySQL. Модель включає таблиці, індекси, ключі та інші компоненти, що забезпечують ефективне зберігання та обробку даних.

На Рисунку 3.9 – Інфологічна модель бази даних представлена структура бази даних інформаційної системи для відстеження прогресу читання книг. Ця модель включає в себе основні сутності та їхні взаємозв'язки, які визначають логічну організацію даних у системі.

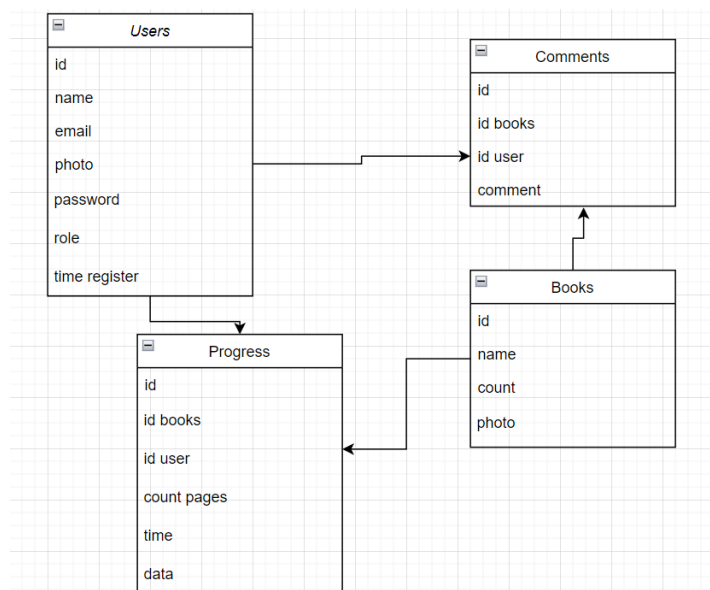


Рисунок 3.9 – Інфологічна модель бази даних

Опис складових бази даних у термінах мови опису даних (DDL)

Таблиця користувачів (Users):

```
CREATE TABLE `users` (  
  `id` int(11) NOT NULL,  
  `name` varchar(255) NOT NULL,
```

```

`email` varchar(255) NOT NULL,
`photo` varchar(255) NOT NULL,
`password` varchar(255) NOT NULL,
`role` varchar(255) NOT NULL,
`time_register` timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP ON
UPDATE CURRENT_TIMESTAMP
) ENGINE=InnoDB DEFAULT CHARSET=utf8;

```

Таблиця книг (Books):

```

CREATE TABLE `books` (
  `id` int(11) NOT NULL,
  `name` varchar(255) NOT NULL,
  `count` int(11) NOT NULL,
  `photo` varchar(255) NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8;

```

Таблиця коментарів:

```

CREATE TABLE `comments` (
  `id` int(11) NOT NULL,
  `id_book` int(11) NOT NULL,
  `id_user` int(11) NOT NULL,
  `comment` varchar(255) NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8;

```

Таблиця прогресу:

```

CREATE TABLE `progress` (
  `id` int(11) NOT NULL,
  `id_book` int(11) NOT NULL,
  `id_user` int(11) NOT NULL,
  `count_pages` int(11) NOT NULL,
  `time` varchar(255) NOT NULL,
  `date` timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP ON UPDATE
CURRENT_TIMESTAMP

```

) ENGINE=InnoDB DEFAULT CHARSET=utf8;

Призначення таблиць

Users: Зберігає інформацію про користувачів застосунку, таку як: ім'я, електронна пошта, фото, пароль, роль та день реєстрації

Books: Містить дані про книги, включаючи ім'я автора, назву книги кількість сторінок, та фото.

Progress: Відстежує процеси читання книг користувачами, включаючи статус читання.

Comments: Залишину користувачем коментар під певною книгою.

Ця структура бази даних забезпечує ефективне зберігання та доступ до необхідної інформації для моніторингу процесу читання книг користувачами.[3]

3.2. Технічне забезпечення

3.2.1 Загальні положення та схема автоматизації.

Інформаційна система для застосунку з моніторингу читання книг передбачає наявність віддаленого сервера (можливо орендованого), на якому буде розташовано ядро системи, базу даних та необхідні файли. Доступ користувачів до системи буде здійснюватися через використання веб-сайту та/або мобільного додатку з доступом до мережі Інтернет.

Фізичне розташування необхідного обладнання передбачає наступне:

Серверна інфраструктура: Центральний сервер, на якому розміщено базу даних та основний програмний код, буде розташовано в дата-центрі. Можливе використання хмарних сервісів, таких як AWS, Google Cloud, чи Azure для забезпечення масштабованості та надійності системи.[9]

Клієнтська частина: Користувачі будуть взаємодіяти із системою через веб-браузери або мобільні додатки на своїх пристроях (смартфони, планшети, ПК). Клієнтська частина додатку буде підключатися до серверу через мережу Інтернет.

Забезпечення безпеки: Для захисту даних користувачів та забезпечення безпечного з'єднання між клієнтами та сервером буде використовуватися

протокол HTTPS. Крім того, сервер буде обладнаний засобами захисту від несанкціонованого доступу та кібератак.[9]

Система буде функціонувати наступним чином:

Ядро системи та база даних: Розміщені на віддаленому сервері або у хмарному середовищі, забезпечуючи централізоване зберігання та обробку даних.

Веб-доступ: Користувачі зможуть заходити на веб-сайт системи з будь-якого пристрою, що підтримує веб-браузер, для доступу до своїх акаунтів, перегляду книг, відстеження прогресу та додавання коментарів.

Мобільний застосунок: Альтернативний спосіб доступу до системи через мобільні додатки для iOS та Android, що дозволить користувачам мати більш зручний і швидкий доступ до функціоналу системи.

Таким чином, система буде забезпечувати зручний і безпечний доступ до функціоналу моніторингу читання книг, використовуючи сучасні технології та інфраструктуру.

Загальна схема автоматизації включає декілька основних компонентів і їх взаємодію. Ось як може виглядати така схема для застосунку з моніторингу читання книг:

Клієнтська частина:

Веб-додаток: Забезпечує доступ користувачів через веб-браузери на різних платформах (ПК, ноутбуки, планшети).

Серверна частина:

Веб-сервер: Приймає запити від клієнтської частини, обробляє їх і взаємодіє з базою даних.

База даних: Зберігає дані користувачів, книги, прогрес читання, коментарі тощо.

Інфраструктура:

Хмарні сервіси: AWS, Google Cloud, Azure або інші, що забезпечують масштабованість і надійність системи.

Системи безпеки: Використання SSL/TLS для шифрування даних, механізми аутентифікації та авторизації (наприклад, OAuth, JWT).

На Рисунку 3.10 – Загальна схема автоматизації показана структура та взаємозв'язки між основними компонентами автоматизованої системи для відстеження прогресу читання книг. Ця схема включає в себе різні модулі та системи, які спільно працюють для забезпечення функціональності та продуктивності системи.

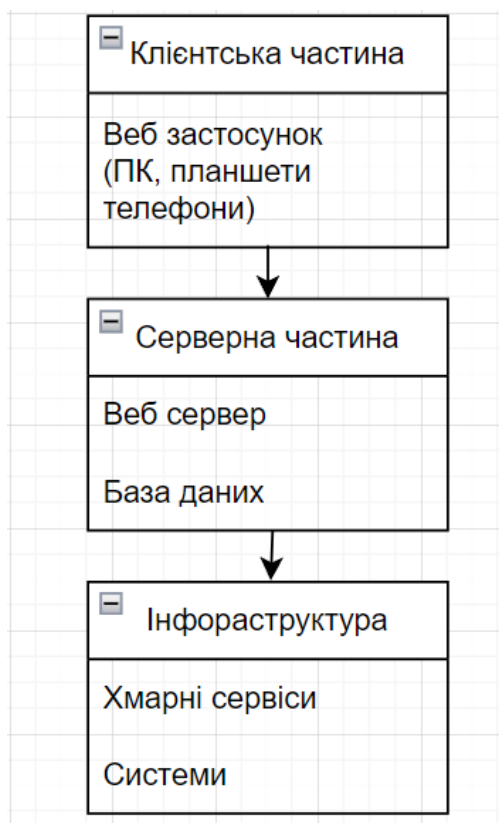


Рисунок 3.10 – Загальна схема автоматизації

3.2.2 Структура комплексу технічних засобів.

Вибір структури комплексу технічних засобів для застосунку з моніторингу читання книг базується на забезпеченні ефективності, надійності та безпеки системи. Основними критеріями вибору є потужність, масштабованість, сумісність з іншими системами та відповідність вимогам користувачів.

Технічні рішення по обміну даними з іншими автоматизованими системами

Для забезпечення ефективного обміну даними з іншими автоматизованими системами застосовується використання API (Application Programming Interface). Це дозволяє легко інтегруватися з іншими системами, такими як бібліотеки електронних книг, платформи для навчання або соціальні мережі, де користувачі можуть ділитися своїми досягненнями.

Мінімально допустима кількість АРМ по різних відділах підприємства

Для ефективного функціонування системи моніторингу читання книг необхідна мінімально допустима кількість автоматизованих робочих місць (АРМ) по різних відділах підприємства:

Адміністративний відділ: 1 АРМ

Відділ технічної підтримки: 2 АРМ

Відділ розвитку та тестування: 2 АРМ

Відділ маркетингу та аналітики: 1 АРМ

Загальні технічні відомості серверу

Для розміщення ядра системи та бази даних використовуватиметься хостинг-провайдер. Обраний провайдер забезпечує необхідну масштабованість та надійність послуг.

Провайдер: AWS (Amazon Web Services)

Тип серверу: EC2 (Elastic Compute Cloud)

Конфігурація серверу:

CPU: 4 vCPU

RAM: 16 GB

Storage: 100 GB SSD

Операційна система: Ubuntu 20.04 LTS

Кількість та характеристики інших апаратних рішень

Клієнтські пристрої: Мобільні телефони (Android, iOS), ПК та ноутбуки

Інші апаратні рішення:

Маршрутизатор та мережеве обладнання: Забезпечують швидкий та безпечний доступ до серверу

Резервні копії та відновлення даних: Використовуються додаткові сервери для резервного копіювання важливої інформації

Опис технічних компонентів

Мобільні пристрої:

Платформи: Android, iOS

Процесори: ARM Cortex-A53 або вище

Оперативна пам'ять: Мінімум 2 GB RAM

Мережа: Wi-Fi, 4G/5G

ПК та ноутбуки:

Операційні системи: Windows 10/11, macOS, Linux

Процесори: Intel Core i5 або вище, AMD Ryzen 5 або вище

Оперативна пам'ять: Мінімум 8 GB RAM

Мережа: Ethernet, Wi-Fi

Маршрутизатор:

Модель: Cisco RV340

Швидкість: Gigabit Ethernet

Функції: VPN, Firewall, QoS

Резервні сервери:

Конфігурація: Аналогічна основному серверу, з акцентом на зберігання даних

Розташування: Віддалені дата-центри AWS для географічної віддаленості

Ця структура комплексу технічних засобів забезпечує надійну, масштабовану та безпечну роботу застосунку з моніторингу читання книг. Використання хмарних технологій дозволяє легко адаптуватися до змін в обсягах даних і користувачів, а сучасні апаратні рішення забезпечують високу продуктивність і стабільність системи.

При розробці системи моніторингу читання книг важливим є правильний вибір периферійних технічних засобів, які забезпечують отримання, контроль, підготовку, збір, реєстрацію, зберігання і відображення інформації. Основні рішення щодо вибору таких засобів включають:

Засоби отримання інформації:

Мобільні пристрої (смартфони, планшети): Забезпечують зручний спосіб для користувачів вводити дані про прочитані книги, оновлювати статус читання та залишати коментарі.

Засоби контролю інформації:

Сервери з базами даних: Центральні сервери, що забезпечують надійне зберігання та доступ до даних. Важливо обрати надійного хостинг-провайдера, який забезпечить захист даних і високу доступність сервісу.

Системи резервного копіювання: Для захисту даних від втрат використовуються регулярні резервні копії на окремих серверах або в хмарних сховищах.

Засоби підготовки та збору інформації:

Комп'ютерні робочі станції: Забезпечують можливість адміністрування системи, внесення змін до бази даних та обробки великих обсягів даних.

Програмне забезпечення для аналізу даних: Використовується для обробки та аналізу інформації про читання книг, що дозволяє виявляти тренди та покращувати роботу системи.

Засоби реєстрації та зберігання інформації:

Системи управління базами даних (СКБД): Вибір СКБД, таких як MySQL, PostgreSQL або Oracle, залежить від обсягу даних, що зберігаються, і вимог до продуктивності.

Хмарні сховища: Зручні для зберігання великих обсягів даних та забезпечення доступу до них з різних місць.

Засоби відображення інформації:

Монітори та екрани: Використовуються для відображення інформації на робочих місцях адміністраторів та користувачів.

Інформаційні панелі та інтерфейси користувача: Мають бути зручними та інтуїтивно зрозумілими, забезпечувати легкий доступ до інформації про книги, статус читання та коментарі.

3.2.3 Опис автоматизованого робочого місця

Обґрунтування та опис рішень з вибору технічних засобів АРМ:

Для забезпечення ефективної та надійної роботи системи моніторингу читання книг, необхідно вибрати технічні засоби з урахуванням мінімальних системних вимог, що гарантують коректну роботу інформаційної системи (ІС). Основні компоненти АРМ включають:

Процесор: Рекомендується використовувати процесори Intel Core i5 або AMD Ryzen 5 і вище для забезпечення достатньої продуктивності.

Оперативна пам'ять: Мінімально допустимий об'єм оперативної пам'яті — 8 ГБ. Оптимально — 16 ГБ для більш швидкої обробки даних та багатозадачності.

Жорсткий диск: Рекомендується використовувати SSD об'ємом не менше 256 ГБ для зберігання операційної системи та програмного забезпечення, а також HDD об'ємом 1 ТБ для зберігання великих обсягів даних.

Графічний процесор: Для базових задач візуалізації достатньо інтегрованої графіки, але для більш вимогливих задач рекомендується дискретна відеокарта, наприклад, NVIDIA GeForce GTX 1050 або AMD Radeon RX 560.

Вибір периферійних технічних засобів АРМ

Обґрунтування та опис рішень вибору периферійних технічних засобів:

Монітори: Рекомендується використовувати монітори з роздільною здатністю Full HD (1920x1080) або вище. Діагональ монітора — від 21 дюймів. Для зручності роботи можна використовувати два монітори.

Клавіатура та миша: Базові периферійні пристрої. Оптимально обирати ергономічні моделі для тривалої роботи.

Розрахунок числа технічних засобів:

Для забезпечення коректної роботи ІС на підприємстві необхідно визначити кількість АРМ з урахуванням кількості співробітників та обсягу виконуваних завдань. Припустимо, що в організації працюють 20 співробітників, і кожен з них потребує робоче місце з описаними вище технічними характеристиками. Таким чином, необхідно:

20 комп'ютерів з відповідними характеристиками.

20 комплектів периферійних пристроїв (монітори, клавіатури, миші, гарнітури).

5 сканерів для використання співробітниками різних відділів.

5 принтерів (МФУ) для друку та сканування документів.

Потреби в мережевому обладнанні:

Для забезпечення з'єднання всіх АРМ до мережі Інтернет та внутрішньої мережі організації необхідне наступне мережеве обладнання:

Маршрутизатор (Router): Для забезпечення з'єднання з Інтернетом. Рекомендується використовувати маршрутизатор з підтримкою VPN та апаратного шифрування для забезпечення безпеки даних.

Комутатори (Switches): Для з'єднання всіх АРМ у локальну мережу. Необхідно мінімум 2 комутатори з 24 портами кожен.

WiFi точка доступу: Для забезпечення бездротового з'єднання мобільних пристроїв. Рекомендується використовувати точки доступу з підтримкою стандарту Wi-Fi 6 для забезпечення високої швидкості та стабільності з'єднання.

Кабелі Ethernet: Для з'єднання стаціонарних АРМ з мережею.

Спосіб підключення АРМ до мережі:

Ethernet: Основний спосіб підключення стаціонарних АРМ до мережі для забезпечення стабільного та швидкого з'єднання.

WiFi: Використовується для підключення мобільних пристроїв та пристроїв, що знаходяться в зонах, де прокладка кабелів неможлива або небажана.

VPN: Використовується для забезпечення захищеного доступу до внутрішньої мережі організації з віддалених місць.

Таким чином, запропонована структура АРМ забезпечує ефективну роботу системи моніторингу читання книг, забезпечуючи надійне зберігання та обробку даних, а також зручний доступ до інформації для всіх користувачів.

3.3. Програмне забезпечення

3.3.1 Структура програмного забезпечення.

Базове або системне програмне забезпечення

Операційна система: Наприклад, Windows, macOS, Linux для десктопного застосунку або Android, iOS для мобільного застосунку.

Системні бібліотеки та фреймворки: Пакети та бібліотеки, що забезпечують робоче середовище для розробки та виконання програм, наприклад, .NET, Java, Node.js.

База даних: СУБД, що використовується для зберігання даних користувачів та книг, MySQL.

Прикладне або спеціалізоване програмне забезпечення

Інтерфейс користувача:

Фронтенд: Веб-інтерфейс або мобільний додаток, який взаємодіє з користувачем.

Інструменти розробки інтерфейсу: HTML, CSS, JavaScript, React, Flutter.

Серверна частина:

Бекенд: Серверна логіка, що обробляє запити від фронтенду та взаємодіє з базою даних.

Фреймворки для бекенду: Express.js для Node.js, Django для Python, Spring для Java.

Логіка додатку:

Авторизація та аутентифікація: Система реєстрації та входу користувачів.

Трекінг читання: Модулі для запуску таймеру, фіксації часу читання та сторінок.

Аналітика: Аналіз даних про читання для формування статистики.

Розробник сценаріїв (Scenario Builder):

Сценарії користувача: Опис взаємодії користувача з додатком, включаючи попередні та наступні умови, обмеження.

Інструменти для створення сценаріїв: UML діаграми, інструменти моделювання процесів.

На Рисунку 3.11 – Схема програмного забезпечення представлена детальна схема основних компонентів програмного забезпечення, яке використовується в інформаційній системі для відстеження прогресу читання книг.

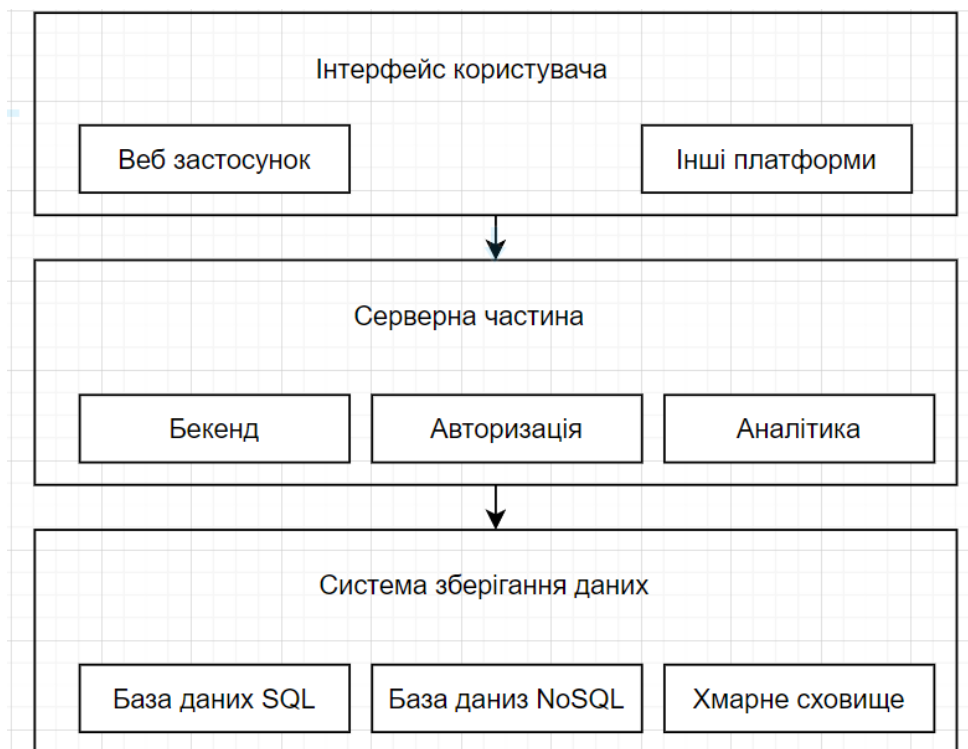


Рисунок 3.11 – Схема програмного забезпечення

3.3.2 Системне програмне забезпечення.

Для забезпечення ефективної та надійної роботи додатку для моніторингу процесу читання книг, необхідно обрати відповідне системне програмне забезпечення. У цьому параграфі розглянемо рішення щодо вибору операційних систем, RAD-засобів розробки проекту, систем керування контентом, інструментів для аналізу неструктурованих даних, сервісних програм, програм технічного обслуговування, програмного забезпечення захисту інформації, програм машинної графіки, програм обробки мовних сигналів та інструментального ПЗ.

Операційні системи

Операційна система (ОС) є основою для роботи будь-якого комп'ютера або мобільного пристрою. Вибір ОС залежить від цільової платформи, на якій буде працювати додаток:

Windows: Популярна ОС для десктопів з широким набором інструментів для розробки.

macOS: Вибір для розробки додатків для екосистеми Apple.

Linux: Відкрита ОС, популярна серед розробників завдяки своїй гнучкості та стабільності.

Android та iOS: Основні операційні системи для мобільних пристроїв.

Windows Server: Використовується для розгортання серверної частини додатку.

Novell Open Enterprise Server (OES): Використовується у корпоративних середовищах.

Для розробки і оформлення графічних схем, діаграм і специфікацій моделей баз даних можна використовувати такі CASE-пакети:

1. AllFusion ERWin Data Modeler: Інструмент для моделювання баз даних.
2. IBM Rational Rose: Засіб для створення UML-діаграм.
3. Sybase PowerDesigner: Засіб для розробки архітектури інформаційних систем.
4. MsSQL Server та MsAccess: Для проектування та управління базами даних.
5. MySQL Workbench: Інструмент для роботи з базами даних MySQL.
6. Oracle Designer: Інструмент для проектування баз даних Oracle.

Системи керування контентом (CMS)

Для створення сайту або веб-інтерфейсу додатку можуть використовуватись як готові системи керування контентом, так і власноруч розроблені сайти з використанням веб-технологій:

1. CMS: WordPress, Joomla, Drupal.
2. Веб-технології: HTML, PHP, MySQL, ASP, JavaScript, React, Angular.
3. Інструменти для аналізу неструктурованих даних

Для аналізу неструктурованих даних та гіпертекстової інформації можуть використовуватись такі програмні пакети:

1. Deductor: Інструмент для аналізу даних та бізнес-аналітики.

2. NeuroPro: Засіб для аналізу великих обсягів даних з використанням нейронних мереж.
3. MineSet: Інструмент для візуалізації та аналізу даних.
4. Polyanalyst: Програмне забезпечення для аналізу даних.
5. Text Analyst: Засіб для аналізу текстової інформації.
6. WizWhy: Інструмент для аналізу та прогнозування на основі даних.
7. CubiCalc: Засіб для багатовимірної аналізу даних.
8. Evolver MineSet: Інструмент для візуалізації та аналізу даних.

Сервісні програми

До сервісних програм належать утиліти та оболонки, які забезпечують зручність роботи з системою:

Оболонки: Bash (для Linux), PowerShell (для Windows).

Утиліти: Архіватори, файлові менеджери, інструменти для управління процесами.

Антивірусні засоби: Norton, McAfee, Kaspersky, Avast.

Програми для технічного обслуговування включають тестові програми та програми контролю:

Тестові програми: MemTest86 (для перевірки оперативної пам'яті), Prime95 (для стрес-тестування CPU).

Програми контролю: Моніторинг системних ресурсів (наприклад, HWMonitor).

Захист інформації є критичним для будь-якого програмного забезпечення:

1. Шифрування: OpenSSL для реалізації криптографічних протоколів.
2. Брандмауери: iptables (для Linux), Windows Firewall.
3. Інструменти для забезпечення безпеки мереж: Snort (система виявлення вторгнень).

Для обробки та створення графічних матеріалів використовуються програми машинної графіки:

1. Adobe Photoshop: Для редагування растрових зображень.
2. Adobe Illustrator: Для створення векторної графіки.

3. CorelDRAW: Засіб для роботи з векторною графікою.

Для розробки програмного забезпечення необхідні транслятори мов програмування, компілятори та інтерпретатори:

1. Компілятори: GCC (для C/C++), javac (для Java).
2. Інтерпретатори: Python, Ruby.
3. Інструменти для управління версіями: Git, Subversion.

Вибір системного програмного забезпечення є критично важливим етапом розробки додатку для моніторингу процесу читання книг. Правильний вибір ОС, інструментів для розробки, систем керування контентом та інших компонентів забезпечить стабільну та ефективну роботу додатку.

3.3.3 Прикладне програмне забезпечення

Для створення та функціонування інформаційної системи моніторингу процесу читання книг необхідно розробити комплекс прикладного програмного забезпечення. Це ПЗ можна поділити на декілька категорій: ПЗ загального призначення, методо-орієнтоване ПЗ, проблемно-орієнтоване ПЗ, ПЗ для роботи глобальних мереж та ПЗ для організації (адміністрування) обчислювального процесу.

ПЗ загального призначення

Ця категорія включає програми, які забезпечують основні функції користувачам, наприклад, текстові редактори, електронні таблиці та інші офісні інструменти. Для розробки нашої системи моніторингу процесу читання книг можна використовувати такі програми:

Microsoft Office: Для створення документації.

Google Docs: Для спільної роботи над документами.

Методо-орієнтоване ПЗ

Методо-орієнтоване ПЗ включає інструменти та програми, що забезпечують реалізацію конкретних методологій і підходів у процесі розробки:

UML-діаграми: Для проектування структури системи використовуються інструменти, такі як Microsoft Visio або Lucidchart.

CASE-засоби: Такі як IBM Rational Rose, для моделювання та проектування програмного забезпечення.

Проблемно-орієнтоване ПЗ

Ця категорія включає програми, що забезпечують рішення специфічних задач у певних галузях:

Спеціалізовані інструменти для аналізу даних: Як Deductor або NeuroPro, для аналізу даних про читання.

Системи управління базами даних: Наприклад, MySQL, PostgreSQL, MongoDB для зберігання даних про користувачів та їхній прогрес у читанні.

ПЗ для роботи глобальних мереж

Для забезпечення мережевої взаємодії необхідно використовувати програми, що забезпечують роботу в глобальних мережах:

Веб-сервери: Apache, Nginx для обробки запитів від клієнтів.

Протоколи зв'язку: HTTPS для безпечної передачі даних.

ПЗ для організації (адміністрування) обчислювального процесу

Ця категорія включає інструменти для управління та моніторингу обчислювальних ресурсів:

Системи моніторингу: Nagios, Zabbix для відстеження стану серверів та баз даних.

Інструменти управління конфігурацією: Ansible, Puppet для автоматизації налаштувань систем.

Вибір засобів розробки прикладного програмного забезпечення

При виборі засобів розробки прикладного програмного забезпечення для нашої інформаційної системи необхідно враховувати наступні критерії:

Функціональні можливості:

Інструменти розробки повинні забезпечувати підтримку необхідних функціональних можливостей, таких як створення користувацьких інтерфейсів, обробка даних, інтеграція з базами даних.

Інтеграція з СКБД:

Обрані засоби розробки повинні забезпечувати легку інтеграцію з обраною системою керування базами даних (наприклад, MySQL, PostgreSQL).

Підтримка RAD (Rapid Application Development):

Інструменти повинні задовольняти основним вимогам до програмних продуктів категорії RAD:

Наявність інтегрованого середовища розробки (IDE): Наприклад, Visual Studio, IntelliJ IDEA, PyCharm.

Наявність засобів для швидкого візуального проектування компонентів ПЗ: Використання таких фреймворків як React, Vue.js для фронтенду, а також Django, Flask для бекенду.

Комплекс програмного забезпечення для розробки

Інтегроване середовище розробки (IDE):

Visual Studio Code: Легке та функціональне IDE, що підтримує різні мови програмування.

IntelliJ IDEA: Потужне IDE для розробки на Java та інших мовах.

Фреймворки та бібліотеки:

Фронтенд: React, Vue.js для створення інтерактивних користувацьких інтерфейсів.

Бекенд: Django, Flask для створення серверної частини.

Системи керування базами даних (СКБД):

MySQL: Для зберігання даних про користувачів та книги.

PostgreSQL: Як альтернатива MySQL, що забезпечує високу надійність та продуктивність.

Інструменти для аналізу та візуалізації даних:

Tableau, Power BI: Для створення візуальних звітів та аналізу даних.

Системи контролю версій:

Git: Для відстеження змін у коді та співпраці розробників.

GitHub, GitLab: Для розміщення репозиторіїв та організації роботи над проектом.

Інструменти автоматизації:

Docker: Для контейнеризації додатку та забезпечення його роботи в різних середовищах.

Jenkins: Для автоматизації процесу збирання та розгортання додатку.

Для розробки інформаційної системи моніторингу процесу читання книг необхідно вибрати та використовувати різноманітні прикладні програмні засоби, що відповідають функціональним вимогам проекту. Вибрані інструменти та засоби повинні забезпечувати швидку та ефективну розробку, інтеграцію з базами даних та високий рівень продуктивності і безпеки системи.

3.3.4 Програмна документація

Програмна документація є важливою частиною розробки інформаційної системи моніторингу процесу читання книг. Вона містить детальний опис програмних засобів, їх можливості, режими роботи, порядок використання та вимоги до інформаційного і технічного забезпечення. Документація допомагає користувачам і системним адміністраторам ефективно експлуатувати та налаштовувати програмне забезпечення.

Формуляр описує основні характеристики програми, її комплектність та відомості про експлуатацію. Він включає:

Назва програми: Система моніторингу процесу читання книг.

Версія: 1.0.

Основні функції:

1. Реєстрація та авторизація користувачів.
2. Додавання та управління книгами з бази даних.
3. Запуск таймера для відстеження часу читання.
4. Введення сторінки завершення читання.
5. Генерація статистики читання.

Комплектність:

1. Виконуваний файл програми.
2. Інструкція користувача.
3. Файли конфігурації.

4. Технічна документація.

Відомості про експлуатацію:

1. Операційні системи: Windows, macOS, Linux.
2. Мови програмування: Python (Django), JavaScript (React).
3. СУБД: MySQL або PostgreSQL.
4. Веб-сервери: Nginx або Apache.

Відомості про призначення програми

Програма призначена для моніторингу процесу читання книг користувачами. Основні області використання:

Бібліотеки: Для відстеження читання книг користувачами.

Освітні заклади: Для аналізу читання студентами.

Персональне використання: Для самостійного моніторингу читання.

Обмеження на застосування та мінімальна конфігурація технічних засобів

Програма вимагає мінімальної конфігурації технічних засобів для ефективного функціонування:

Процесор: Двоядерний процесор з тактовою частотою 2.0 ГГц або вище.

Оперативна пам'ять: 4 ГБ.

Місце на диску: 100 МБ для установки програми та бази даних.

Операційна система: Windows 10 або вище, macOS 10.12 або вище, Linux Kernel 4.0 або вище.

Мережеве з'єднання: Інтернет для синхронізації та оновлень.

Керівництво системного програміста

Керівництво системного програміста містить інформацію для перевірки, забезпечення функціонування та налаштування програми відповідно до умов конкретного застосування:

Завантаження та установка залежностей (Python, Django, MySQL/PostgreSQL, Node.js, Nginx/Apache).

Налаштування середовища (виробниче або розробницьке).

Конфігурація:

Налаштування файлів конфігурації (settings.py для Django, конфігураційний файл для Nginx/Apache).

Налаштування бази даних (створення схем та таблиць, налаштування прав доступу).

Запуск та обслуговування:

Запуск сервера (Django development server або WSGI-сервер для виробничого середовища).

Перевірка логів та моніторинг продуктивності.

Налаштування резервного копіювання бази даних.

Опис контрольного прикладу

Контрольний приклад демонструє функціонування програми на реальному сценарії:

Реєстрація користувача: Користувач створює обліковий запис.

Додавання книги: Користувач додає книгу з бази даних.

Початок читання: Користувач натискає кнопку "Почати читання", таймер запускається.

Закінчення читання: Користувач натискає кнопку "Закінчити читання" та вводить сторінку, на якій зупинився.

Статистика читання: Система генерує звіт про прочитані сторінки та витрачений час.

Тексти програми

Тексти програми включають вихідний код усіх компонентів інформаційної системи:

Backend (Django):

Моделі (models.py): Опис структури даних.

В'юхи (views.py): Логіка обробки запитів.

URL-маршрутизація (urls.py): Визначення маршрутів для запитів.

Серіалізатори (serializers.py): Перетворення даних для API.

Frontend (React):

Компоненти (App.js, компоненти користувацького інтерфейсу).

Redux (actions, reducers): Управління станом додатку.

Сервіси (API.js): Обробка запитів до серверу.

Програмна документація забезпечує детальний опис програмних засобів, їх можливостей та режимів роботи, а також надає необхідну інформацію для ефективної експлуатації та налаштування програмного забезпечення. Вона є важливим інструментом для користувачів та системних адміністраторів, допомагаючи забезпечити стабільну та надійну роботу інформаційної системи.

Налаштування Середовища Розробки

Встановили MAMP, завантажили та встановили MAMP з офіційного сайту, налаштували кореневий каталог документів та порт для Apache і MySQL.

Створення бази даних:

Відкрили phpMyAdmin через MAMP, створили базу даних book_management, створили необхідні таблиці: users, books, comments, progress, additions. (Додаток 1)

На Рисунку 3.12 можна побачити результат створення бази даних у програмі phpMyAdmin.

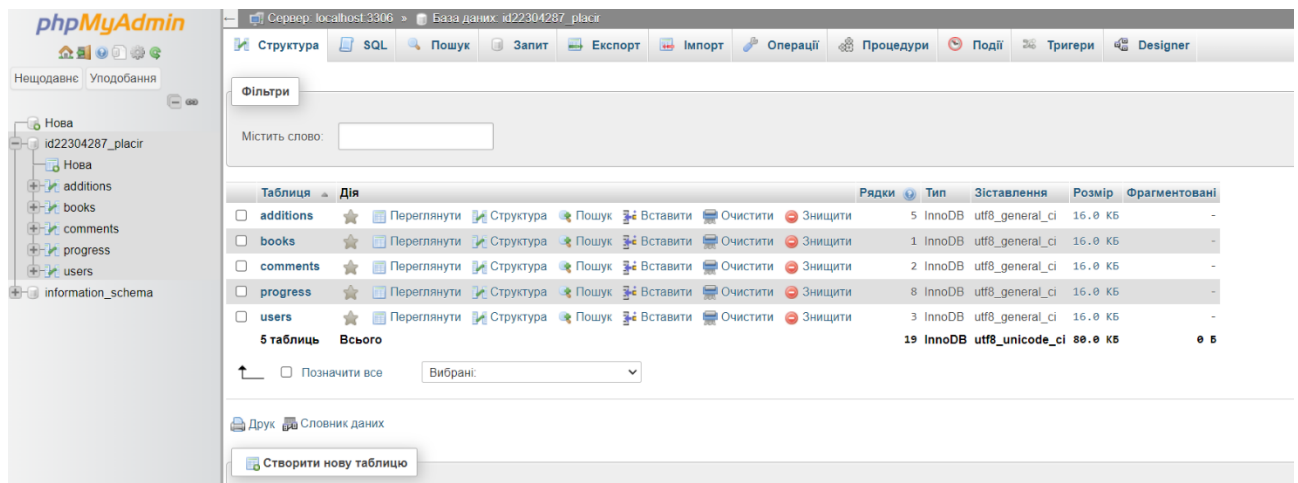


Рисунок 3.12 – Створення бази даних

Реалізація Фронтенду

HTML & CSS: Розробили базову структуру HTML для відображення списку книг та коментарів, створили стилі за допомогою CSS для оформлення інтерфейсу. (Додаток 2)

На Рисунку 3.13 зображено реалізацію фронтенду для інформаційної системи з відстеження прогресу читання книг. У цьому зображенні використано середовище розробки Visual Studio Code для написання коду.

```
body,
html {
  height: 100%;
}

.wrapper {
  max-width: 480px;
  background: #191919;
  margin: 0 auto;
  height: 100%;
  min-height: 100%;
}

.container {
  width: 90%;
  margin: 0 auto;
}

.welcome-page__title {
  font-size: 32px;
  font-weight: 700;
}

.link-btn {
  border: 1px solid #000000;
  border-radius: 25px;
  background: #000000;
  height: 45px;
  max-width: 290px;
  width: 100%;
  text-decoration: none;
  color: #000000;
  font-size: 24px;
  font-weight: 700;
}
```

Рисунок 3.13 – Реалізація фронтенду

Реалізація Бекенду. Функції для роботи з базою даних: реалізували функції для отримання списку книг, додавання книг до списку користувача та отримання коментарів до книг. (Додаток 3)

На Рисунку 3.14 представлена реалізація бекенду для інформаційної системи з відстеження прогресу читання книг. У цьому зображенні можна побачити код, написаний у середовищі розробки Visual Studio Code.

```

1  <?php
2  session_start();
3  function db_connect()
4  {
5      $connect = mysqli_connect('localhost', 'id22304287_placir', 'Placir24@', 'id22304287_placir');
6
7      if (!$connect) {
8          return 'Помилка підключення до бази даних';
9      } else {
10         return $connect;
11     }
12 }
13
14 function model_user_register($name, $email, $password, $photo, $role)
15 {
16     $connect = db_connect();
17
18     $query_check_email = "SELECT * FROM `users` WHERE `email` = '$email'";
19     $result_check_email = mysqli_query($connect, $query_check_email);
20
21     if (mysqli_num_rows($result_check_email) > 0) {
22         return false;
23     } else {
24
25         $photo_path = NULL;
26         if (isset($_FILES['photo']) && $_FILES['photo']['error'] === UPLOAD_ERR_OK) {
27             $target_dir = "uploads/";
28             $target_file = $target_dir . basename($_FILES["photo"]["name"]);
29             $uploadOk = 1;
30             $imageFileType = strtolower(pathinfo($target_file, PATHINFO_EXTENSION));
31
32             $check = getimagesize($_FILES["photo"]["tmp_name"]);
33             if ($check !== false) {
34                 $uploadOk = 1;
35             } else {
36                 $uploadOk = 0;
37             }

```

Рисунок 3.14 – Реалізація Бекенду

Контролер для обробки списку книг: Включає логіку для відображення книг та коментарів, а також обробку додавання книг до списку користувача. (Додаток 4)

На Рисунок 3.15 представлена реалізація контролерів. У цьому зображенні можна побачити код, написаний у середовищі розробки Visual Studio Code.

```

1  <?php
2
3  function welcome_controller()
4  {
5      require_once './views/welcome.php';
6  }
7  function login_controller()
8  {
9      if ($_SERVER['REQUEST_METHOD'] == 'POST') {
10         $email = $_POST['email'];
11         $password = $_POST['password'];
12
13         $result_login = model_user_login($email, $password);
14
15         if ($result_login) {
16
17             if (isset($_SESSION['user_role'])) {
18                 if ($_SESSION['user_role'] == 'user') {
19                     header('Location: ./success');
20                 } elseif ($_SESSION['user_role'] == 'admin') {
21                     header('Location: ./admin');
22                 }
23             }
24
25         } else {
26             $error_message = "Неправильний email або пароль";
27         }
28     }
29
30     require_once './views/login.php';
31 }
32
33 function register_controller()
34 {
35     $name = $_POST['name'];
36     $email = $_POST['email'];
37     $password = $_POST['password'];

```

Рисунок 3.15 – Опис контролерів

3.4. Результати реалізації інформаційної підсистеми

У процесі виконання кваліфікаційної бакалаврської роботи було розроблено та впроваджено комплекс програмних засобів для моніторингу процесу читання книг. Основою цієї розробки стали оригінальні пропозиції щодо методів відстеження та аналізу часу читання, а також проектні та експериментальні дослідження, виконані під керівництвом наукового керівника.

Вхід з боку адміністрації (див. рисунок 3.16, 3.17, 3.18, 3.19)

Вхідні дані: Адміністратор вводить свої облікові дані, які складаються з електронної пошти та пароля.

Процес аутентифікації: Система перевіряє введені дані на відповідність в базі даних. Якщо дані правильні, адміністратор отримує доступ до панелі керування.

Вихідні дані: У разі успішної аутентифікації адміністратор перенаправляється на панель керування. У разі помилки аутентифікації виводиться повідомлення про помилку.

Панель керування адміністратора

Основні функції: Панель керування містить розділи для управління книгами.

Вихідні дані: Дані про поточний стан системи, звіти, список книг, доступ до налаштувань.

Вхід з боку користувача (див. рисунок 3.20, 3.21, 3.22, 3.23)

Вхідні дані: Користувач вводить свої облікові дані, які складаються з електронної пошти та пароля.

Процес аутентифікації: Система перевіряє введені дані на відповідність в базі даних. Якщо дані правильні, користувач отримує доступ до головної сторінки застосунку.

Вихідні дані: У разі успішної аутентифікації користувач перенаправляється на головну сторінку. У разі помилки аутентифікації виводиться повідомлення про помилку.

Після входу користувач бачить перелік доступних книг. Можливість пошуку книг за параметрами, такими як автор та назва.

Користувач може переглянути та редагувати свою особисту інформацію, таку як ім'я, фото, електронна пошта.

Історія читання

Функціонал: Користувач може переглядати свою історію читання, включаючи список прочитаних книг, поточні книги в процесі читання та закладені книги.

Вихідні дані: Оновлений список книг в історії читання користувача.

Процес читання книги (див. рисунок 3.24, 3.25, 3.26, 3.27)

Користувач вибирає книгу зі списку доступних книг для читання. Користувач починає читання вибраної книги.

Продовження читання

Користувач може зупинити читання і зберегти сторінку кінця, щоб продовжити читання пізніше з того ж місця. Система автоматично зберігає прогрес читання.

Закінчення читання

Після закінчення книги користувач може оцінити книгу та залишити відгук. Прогрес читання оновлюється в історії читання користувача.

Процеси:

Введення оцінки та коментаря.

Збереження коментаря в базі даних.

Вихідні дані: Оновлений рейтинг та коментарі до книги.

Додавання коментаря (див. рисунок 3.28, 3.29, 3.30, 3.31)

Вибір книги для коментування. Користувач вибирає книгу, яку хоче прокоментувати, зі списку прочитаних книг або зі списку книг, які зараз читаються, далі переходить до форми для написання коментаря. Форма містить поле для введення тексту. Після введення коментаря тест зберігається в базі даних та стає доступним для перегляду іншими користувачами.

Ці процеси забезпечують зручне використання системи як для адміністрації, так і для користувачів, дозволяючи їм ефективно взаємодіяти з книгами, відстежувати прогрес читання та обмінюватися враженнями через коментарі.

Проведене пілотне впровадження системи моніторингу процесу читання книг показало високу практичну цінність розробки. Основні переваги системи включають:

Мотивація користувачів:

Можливість відстежувати свій прогрес у читанні мотивує користувачів до регулярного читання та досягнення цілей.

Аналіз продуктивності:

Система дозволяє користувачам аналізувати свій час читання, що допомагає у плануванні та оптимізації часу, витраченого на читання.

Зручність використання:

Інтуїтивний інтерфейс та простота введення даних забезпечують легкість використання системи навіть для недосвідчених користувачів.

Інтеграція з освітніми платформами:

Система може бути легко інтегрована з існуючими освітніми платформами, що дозволяє використовувати її для моніторингу читання навчальної літератури.

У порівнянні з існуючими на ринку альтернативами, такими як Goodreads, Bookly, і Libib, розроблена система має кілька ключових переваг:

Goodreads: фокусується на соціальних функціях, таких як обмін рецензіями та рекомендаціями книг, але не має функціоналу для точного вимірювання часу читання.

Bookly: забезпечує детальний облік часу читання та аналіз швидкості читання, проте має обмежені можливості для інтеграції з освітніми платформами.

Libib: використовується переважно для управління бібліотечними колекціями і не пропонує функціоналу для моніторингу часу читання.

Наша система, крім відстеження часу читання, надає детальну статистику, можливості інтеграції з освітніми платформами та зручний інтерфейс, що робить її унікальною на ринку.

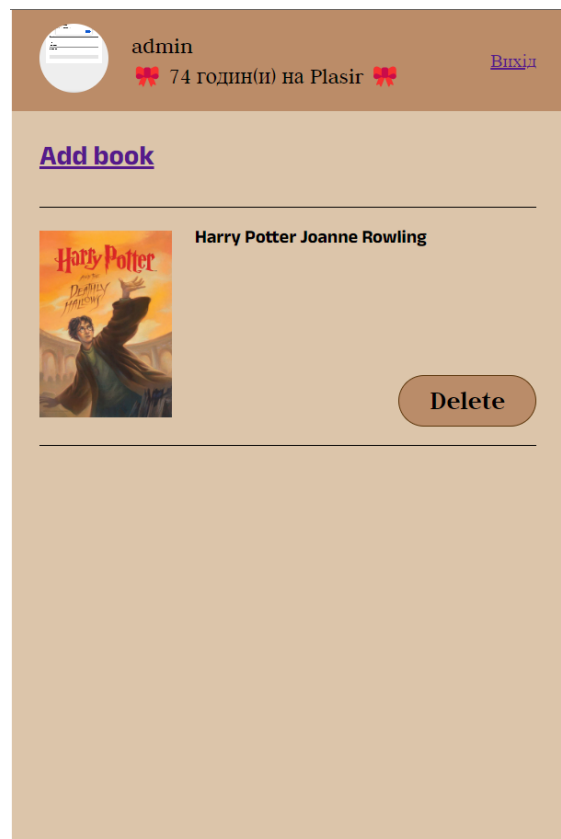
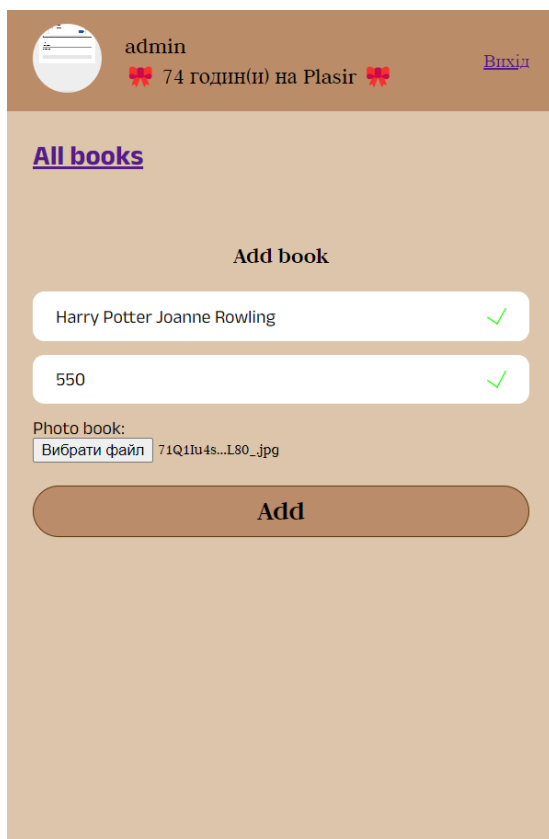
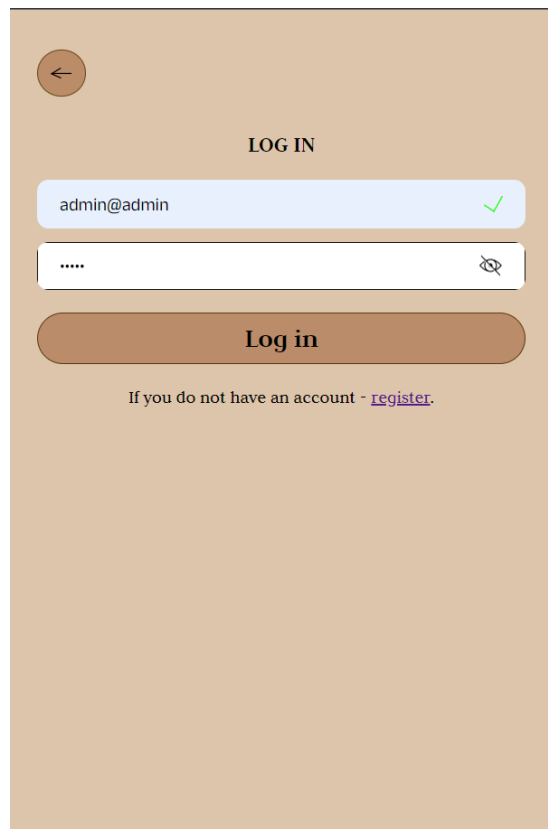
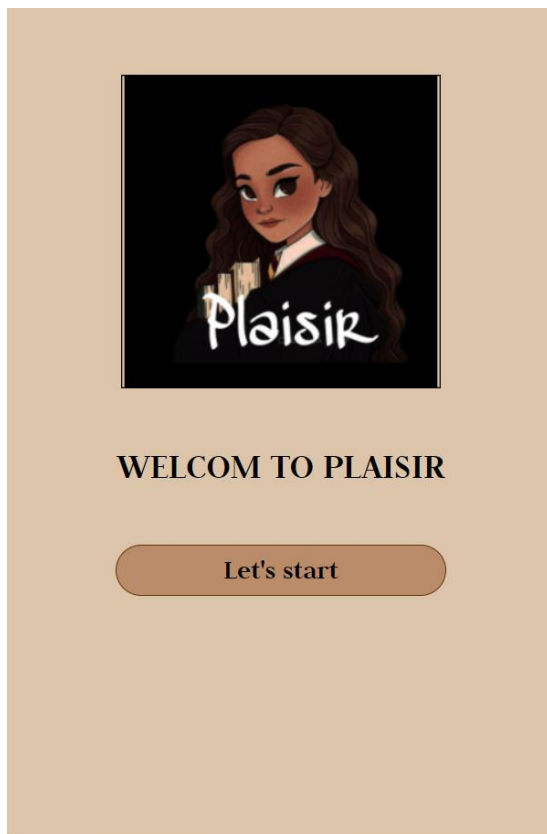


Рисунок 3.16, 3.17, 3.18, 3.19 - Вхід з боку адміністрації

log in.'."/>

←

OR LOG IN WITH EMAIL

Oksana ✓

oksanapuchko@ukr.net ✓

..... 🔍

Your photo:
Вибрати файл Файл не вибрано

Get started

if you have an account - [log in.](#)

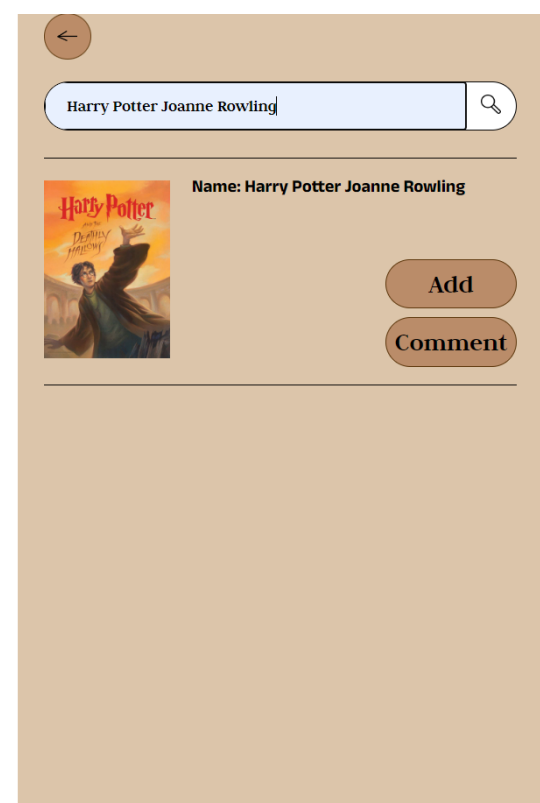
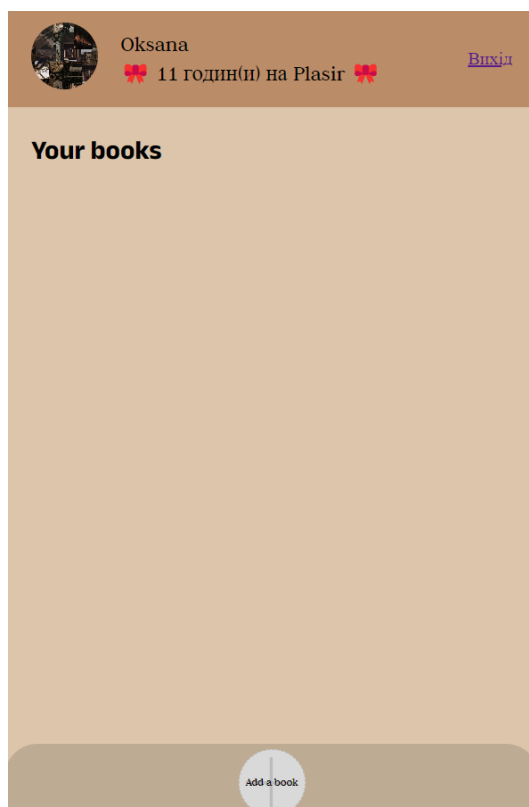
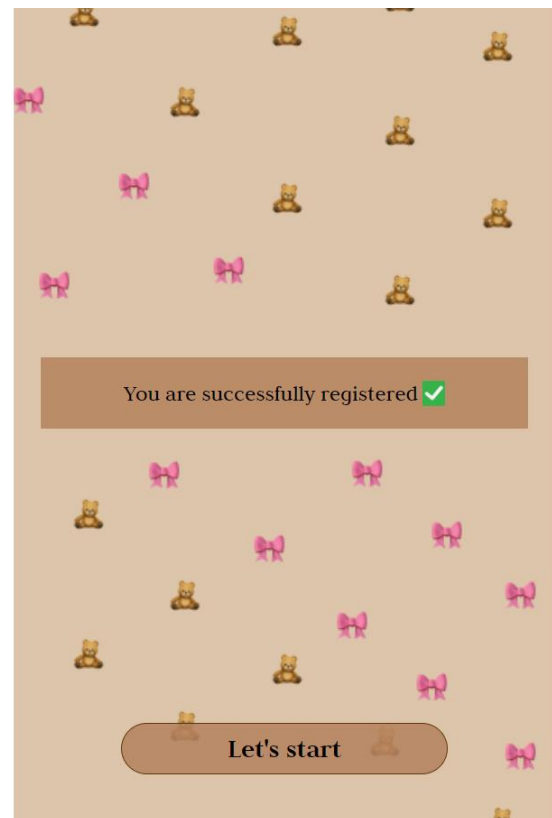


Рисунок 3.20, 3.21, 3.22, 3.23– Вхід з боку користувача

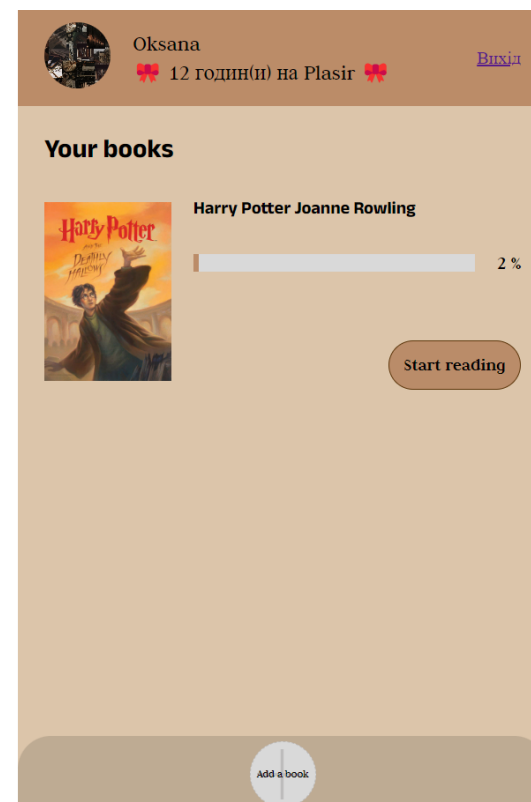
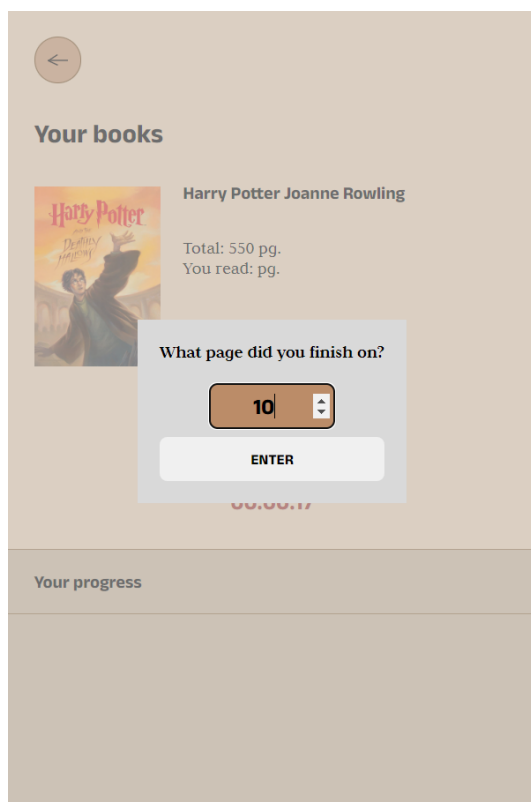
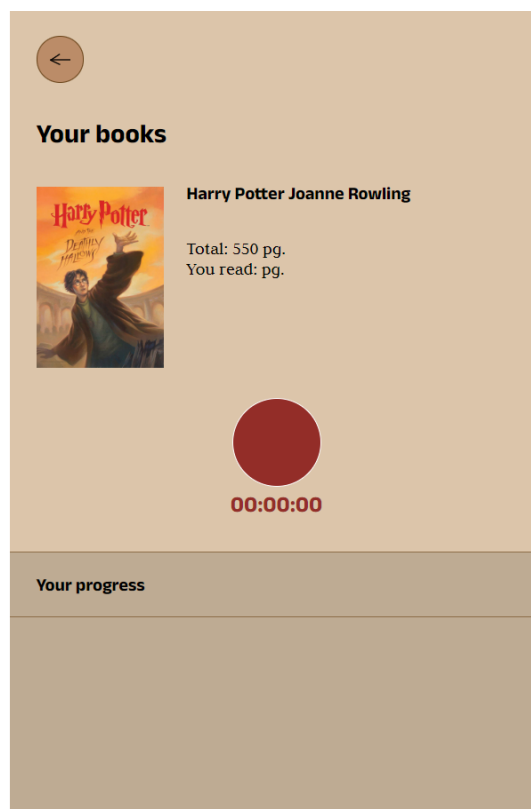
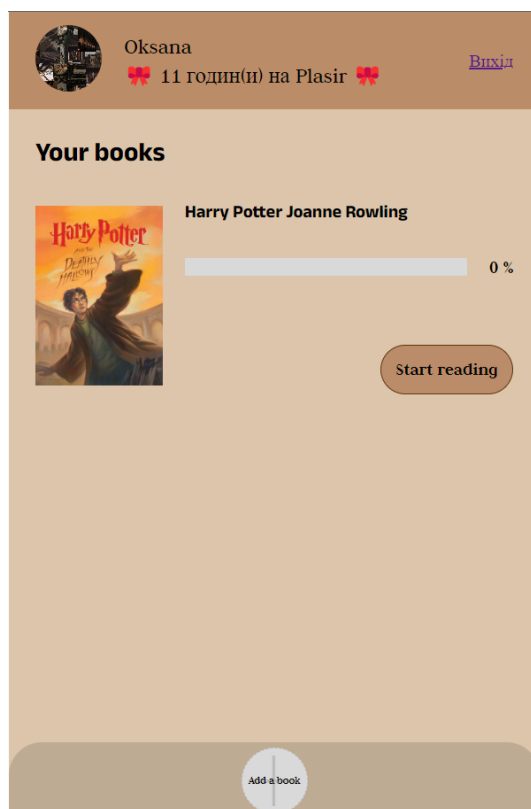


Рисунок 3.24, 3.25, 3.26, 3.27 – Процес читання книги

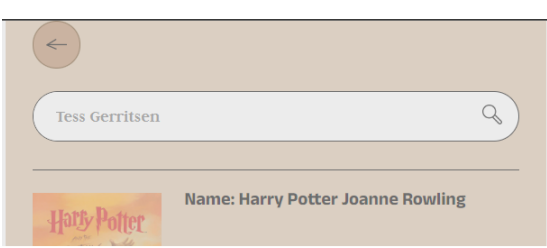
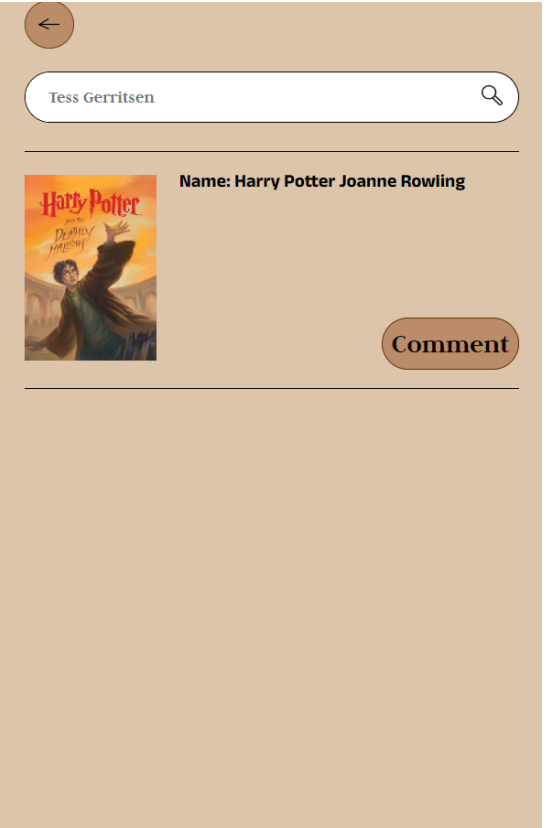
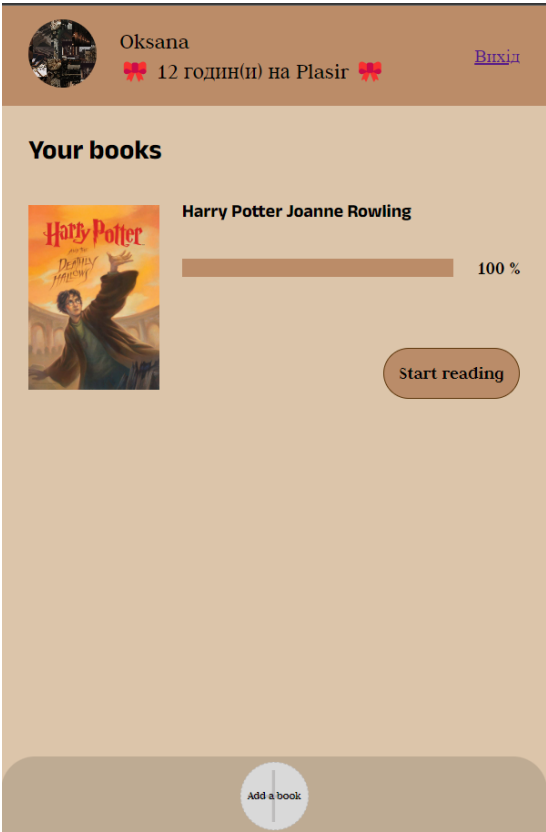


Рисунок 3.28, 3.29, 3.30, 3.31 – Додавання коментаря

ВИСНОВОК

Кваліфікаційний бакалаврський проект був присвячений розробці системи моніторингу процесу читання книг. У ході дослідження було проведено аналіз існуючих рішень, визначено їх недоліки та запропоновано оригінальні методи для покращення процесу відстеження часу читання.

Основні результати роботи включають:

Розробка системи:

Створено програмне забезпечення, яке дозволяє користувачам відстежувати час, витрачений на читання книг, та вводити дані про прочитані сторінки.

Перспективи впровадження включають розширення використання системи у школах, університетах, бібліотеках та онлайн-платформах.

Рекомендації щодо вдосконалення:

Рекомендується розширити функціональність системи, додавши можливості для відстеження читання електронних книг та інтеграції з іншими навчальними системами.

Подальші дослідження можуть включати використання економіко-математичних методів для аналізу ефективності читання та оптимізації навчальних процесів.

Таким чином, розроблена система моніторингу процесу читання книг є ефективним інструментом для покращення організації навчання та підвищення мотивації до читання. Результати пілотного впровадження підтверджують доцільність та ефективність використання системи в освітніх та бібліотечних установах. Подальші дослідження та вдосконалення дозволять розширити можливості системи та підвищити її конкурентоспроможність на ринку.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Грудинін, А., & Іванов, П. (2010). Методи проектування інформаційних систем. Київ: Видавничий дім «Освіта».
2. Дейкстра, Е. (1976). Мистецтво програмування. Лондон: Видавництво «Комп'ютерна література».
3. Добровольський, В. (2007). Основи програмної інженерії. Львів: Видавництво «Техніка».
4. Лапчинська, І. (2013). Основи проектування баз даних. Дніпро: Видавництво «Дніпро».
5. Нікольський, Ю. (2016). Основи розробки програмного забезпечення. Харків: Видавництво «Прометей».
6. Романенко, В. (2011). Інтеграція програмних систем. Одеса: Видавництво «Чорномор'я».
7. Стоянов, А. (2008). Моделювання інформаційних систем. Дніпро: Видавництво «Дніпро».
8. Чернишов, А. (2014). Методології проектування програмних систем. Київ: Видавничий дім «Либідь».
9. Шляхтин, В. (2018). Програмне забезпечення: Розробка та тестування. Львів: Видавництво «Техніка».
10. Goodreads. (2024). Retrieved from <https://www.goodreads.com/>.
11. Bookly. (2024). Retrieved from <https://booklyapp.com/>.
12. Libib. (2024). Retrieved from <https://www.libib.com/>.
13. ДСТУ 3973-2000. (2000). Інформаційні технології. Програмна документація. Вимоги до змісту і оформлення. Київ: Держстандарт України.
14. ISO/IEC 9126. (2001). Software engineering - Product quality. International Organization for Standardization.

ДОДАТКИ

Додаток 1 - Створення бази даних

```
CREATE TABLE `users` (  
  
  `id` int(11) NOT NULL AUTO_INCREMENT,  
  
  `username` varchar(255) NOT NULL,  
  
  `password` varchar(255) NOT NULL,  
  
  PRIMARY KEY (`id`)  
  
);
```

```
CREATE TABLE `books` (  
  
  `id` int(11) NOT NULL AUTO_INCREMENT,  
  
  `name` varchar(255) NOT NULL,  
  
  `photo` varchar(255) NOT NULL,  
  
  `count` int(11) NOT NULL,  
  
  PRIMARY KEY (`id`)  
  
);
```

```
CREATE TABLE `comments` (  
  
  `id` int(11) NOT NULL AUTO_INCREMENT,  
  
  `id_book` int(11) NOT NULL,  
  
  `comment` text NOT NULL,  
  
  PRIMARY KEY (`id`)  
  
);
```

```
CREATE TABLE `progress` (  
  
  `id` int(11) NOT NULL AUTO_INCREMENT,
```

```
`id_user` int(11) NOT NULL,  
  
`id_book` int(11) NOT NULL,  
  
`count_pages` int(11) NOT NULL,  
  
PRIMARY KEY (`id`)  
  
);  
  
CREATE TABLE `additions` (  
  
`id` int(11) NOT NULL AUTO_INCREMENT,  
  
`id_user` int(11) NOT NULL,  
  
`id_book` int(11) NOT NULL,  
  
PRIMARY KEY (`id`)  
  
);
```

Додаток 2 - Реалізація Фронтенду

```
<!DOCTYPE html>  
  
<html lang="en">  
  
<head>  
  
<meta charset="UTF-8">  
  
<meta name="viewport" content="width=device-width,  
initial-scale=1.0">  
  
<title>Book Management</title>  
  
<link rel="stylesheet" href="./assets/style/index.css">  
  
</head>  
  
<body>
```

```
<div class="wrapper">

<div class="container">

<h1>Book Management System</h1>

<form action="" class="form-search" method="get">

<input type="text" placeholder="Search by book
name" name="name">

<button type="submit">Search</button>

</form>

<div class="book-list">

<?php foreach ($all_books as $book): ?>

<div class="book-item">

">

<h2><?= $book['name'] ?></h2>

<div class="comments">

<?php foreach ($book['comments']
as $comment): ?>

<p><?= $comment['comment']
?></p>

<?php endforeach; ?>

</div>

<form action="" method="post">

<input type="hidden" name="id">
```

```

value="<?= $book['id'] ?>">

<button type="submit">Add to My

List</button>

</form>

</div>

<?php endforeach; ?>

</div>

</div>

</div>

<script src="./assets/js/app.js"></script>

</body>

</html>

```

Додаток 3 - Реалізація Бекенду

```

function model_show_all_book() {

    $connect = db_connect();

    $books = mysqli_query($connect, "SELECT * FROM `books`");

    $books = mysqli_fetch_all($books, MYSQLI_ASSOC);

    return $books;

}

function model_get_comments_by_book_id($book_id) {

    $connect = db_connect();

    $query = "SELECT comment FROM comments WHERE id_book = ?";

    $stmt = $connect->prepare($query);

```

```

$stmt->bind_param("i", $book_id);

$stmt->execute();

$result = $stmt->get_result();

$comments = $result->fetch_all(MYSQLI_ASSOC);

$stmt->close();

$connect->close();

return $comments;

}

```

Додаток 4 - Контролери

```

function book_my_list_controller() {

    $id_user = $_SESSION['user_id'];

    $search_name = isset($_GET['name']) ? $_GET['name'] : "";

    if ($search_name) {

        $all_books = model_search_books_by_name($search_name);

    } else {

        $all_books = model_show_all_book();

    }

    foreach ($all_books as $key => $book) {

        $all_books[$key]['is_in_user_list'] =

        model_is_book_in_user_list($book['id'], $id_user);

        $all_books[$key]['comments'] =

        model_get_comments_by_book_id($book['id']);

    }
}

```

```
if ($_SERVER['REQUEST_METHOD'] == 'POST') {  
  
    $id_book = $_POST['id'];  
  
    if (!model_is_book_in_user_list($id_book, $id_user)) {  
  
        $add_book = model_add_user_book($id_book, $id_user);  
  
        if ($add_book) {  
  
            header('Location: ./profile');  
  
            exit;  
  
        }  
  
    }  
  
    }  
  
    require_once './views/book-my-list.php';  
  
}
```