

Вінницький національний технічний університет
(повне найменування вищого навчального закладу)

Факультет інтелектуальних інформаційних технологій та автоматизації
(повне найменування інституту, факультету)

Кафедра комп'ютерних систем управління
(повна назва кафедри)

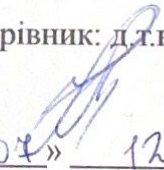
Магістерська кваліфікаційна робота
на тему:

«Розробка модуля користувача мобільного застосунку автоматизованої
системи електронного щоденника для загальноосвітніх шкіл»

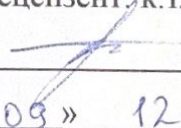
Виконав: студент 2-го курсу,
групи ЗАКІТР-23м
спеціальності 174 – Автоматизація,
комп'ютерно-інтегровані технології та
робототехніка
(шифр і назва спеціальності)

 Денис ПАЛАМАРЧУК
(ім'я та прізвище)

Керівник: д.т.н., проф., каф. КСУ

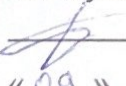
 Марія ЮХИМЧУК
(ім'я та прізвище)
«04» 12 2024 р.

Рецензент: к.т.н., доцент каф. АІТ

 Володимир ГАРМАШ
(ім'я та прізвище)
«09» 12 2024 р.

Допущено до захисту

Завідувач кафедри КСУ

 В'ячеслав КОВТУН
(ім'я та прізвище)
«09» 12 2024 р.

Вінниця ВНТУ - 2024 рік

ВІННИЦЬКИЙ НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ
Факультет інтелектуальних інформаційних технологій та автоматизації
Кафедра комп'ютерних систем управління
Рівень вищої освіти II-й (магістерський)
Галузь знань – 17 Електроніка, автоматизація та електронні комунікації
Спеціальність 174 Автоматизація, комп'ютерно-інтегровані технології та
робототехніка
Освітньо-професійна програма Інформаційні системи і Інтернет речей

ЗАТВЕРДЖУЮ

Завідувач кафедри КСУ

В'ячеслав КОВТУН

« 14 » 10 2024 року

**З А В Д А Н Н Я
НА МАГІСТЕРСЬКУ КВАЛІФІКАЦІЙНУ РОБОТУ СТУДЕНТУ**

Паламарчуку Денису Олександровичу

(прізвище, ім'я, по батькові)

1. Тема магістерської кваліфікаційної роботи «Розробка модуля користувача мобільного застосунку автоматизованої системи електронного щоденника для загальноосвітніх шкіл»

керівник магістерської кваліфікаційної роботи Юхимчук Марія Сергіївна

(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)

затверджені наказом вищого навчального закладу від «17» 09 2024 р. № 310

2. Строк подання студентом магістерської кваліфікаційної роботи 1 12. 2024 р.

3. Вихідні дані до магістерської кваліфікаційної роботи: підтримка мобільних платформ – Android; максимальний час відгуку системи – 2 с; авторизація користувачів – так; максимальна кількість одночасних користувачів – до 1000 осіб; мови інтерфейсу – українська; використання шифрування даних під час передачі – так; інтеграція з зовнішніми освітніми сервісами – так; забезпечення роботи системи в режимі 24/7.

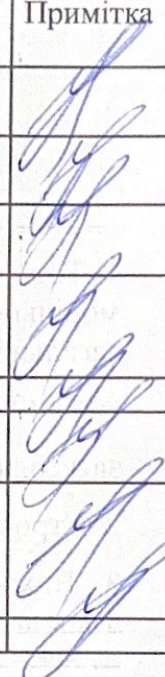
4. Зміст розрахунково-пояснювальної записки (перелік питань, які потрібно розробити): вступ, дослідження об'єкта автоматизації, дослідження та аналіз роботи систем управління навчальним процесом в загальноосвітніх школах, дослідження функціональних можливостей сучасних мобільних застосунків, проектування модуля користувача мобільного застосунку автоматизованої системи електронного щоденника для загальноосвітніх шкіл, розробка модуля користувача мобільного застосунку автоматизованої системи електронного щоденника для загальноосвітніх шкіл.

5. Перелік графічного матеріалу (з точним зазначенням обов'язкових креслень): ER-діаграма структури бази даних системи, UML-діаграма компонентів, UML-діаграма класів системи, UML-діаграма варіантів використання – 2 шт., вигляд екрану розробленого додатку – 12 шт.

6. Консультанти розділів магістерської кваліфікаційної роботи

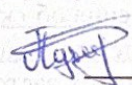
Розділ змістової частини роботи	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв

Календарний план

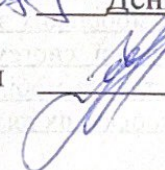
№ з/п	Назва етапів роботи	Строк виконання етапів роботи	Примітка
1	Дослідження об'єкта автоматизації та постановка задач дослідження	03.10.2024р.	
2	Дослідження та аналіз роботи систем управління навчальним процесом в загальноосвітніх школах	19.10.2024р.	
3	Дослідження функціональних можливостей сучасних мобільних застосунків	01.11.2024р.	
4	Проектування модуля користувача мобільного застосунку автоматизованої системи електронного щоденника для загальноосвітніх шкіл	15.11.2024р.	
5	Апробація результатів дослідження	20.11.2024р.	
6	Оформлення пояснювальної записки, графічного матеріалу і презентації	29.11.2024р.	
7	Графічні матеріали: UML-діаграми UML-діаграма класів вигляд екранів розробленого мобільного додатку	04.11.2024 р 18.11.202 р 18.11.2024 р.	
8	Захист МКР	11.12.2024 р.	

Дата видачі завдання “ 14 ” 10 2024 року

Студент

 Денис ПАЛАМАРЧУК

Керівник магістерської кваліфікаційної роботи

 Марія ЮХИМЧУК

АНОТАЦІЯ

УДК 004.043

Паламарчук Д. О. Розробка модуля користувача мобільного застосунку автоматизованої системи електронного щоденника для загальноосвітніх шкіл. Магістерська кваліфікаційна робота зі спеціальності 174 - Автоматизація, комп'ютерно-інтегровані технології, та робототехніка, освітня програма – Інтелектуальні комп'ютерні системи. Вінниця: ВНТУ, 2024. 117с.

Українською мовою, 4 розділи, 49 джерел, рис.:17, табл.: 1

Мета роботи – розробка модуля користувача мобільного застосунку для автоматизованої системи електронного щоденника в загальноосвітніх школах, спрямована на спрощення взаємодії між учасниками освітнього процесу, оптимізацію управління навчальними матеріалами та підвищення ефективності навчання.

В теоретичній частині роботи розглянуто особливості автоматизації в освітній сфері, аналіз ринку IT-послуг для освітніх установ та потреб користувачів. В практичній частині проведено проектування та розробку модуля користувача, включаючи структуру бази даних, UML-діаграми, визначення основних функцій модуля та вибір технологій для його реалізації. Також було проведено тестування та налаштування роботи модуля.

Ключові слова: автоматизована система, електронний щоденник, освітній процес, мобільний застосунок, UML-діаграми, тестування.

ANNOTATION

Palamarchuk D.O. Development of a User Module for a Mobile Application of the Automated Electronic Diary System for General Education Schools. Master's Qualification Thesis in Specialty 174 - Automation, Computer-Integrated Technologies, and Robotics, Educational Program – Intelligent Computer Systems. Vinnytsia: VNTU, 2024. 117 pages. In Ukrainian, 4 chapters, 49 references, illustrations: 17, tables: 1.

The aim of the work is to develop a user module for a mobile application for an automated electronic diary system in general education schools, aimed at simplifying the interaction between participants in the educational process, optimizing the management of educational materials, and enhancing the efficiency of learning.

The theoretical part of the work examines the specifics of automation in the educational sector, analyzes the IT services market for educational institutions, and user needs. In the practical part, the design and development of the user module were carried out, including the database structure, UML diagrams, definition of the main functions of the module, and selection of technologies for its implementation. Testing and configuration of the module's operation were also conducted.

Keywords: automated system, electronic diary, educational process, mobile application, UML diagrams, testing.

ЗМІСТ

ВСТУП.....	4
1 ДОСЛІДЖЕННЯ ТА АНАЛІЗ РОБОТИ СИСТЕМ УПРАВЛІННЯ НАВЧАЛЬНИМ ПРОЦЕСОМ В ЗАГАЛЬНООСВІТНІХ ШКОЛАХ.....	8
1.1 Дослідження об'єкта автоматизації.....	8
1.2 Дослідження можливостей різних ролей користувачів у електронному щоденнику.....	12
1.3 Аналіз механізмів взаємодії різних ролей користувачів через електронний щоденник.....	14
1.4 Аналіз ринку IT-послуг для освітніх установ.....	18
1.5 Аналіз потреб користувачів в автоматизації навчального процесу.....	22
1.6 Дослідження існуючих систем електронних щоденників в школах та уточнена постановка задачі.....	26
2 ДОСЛІДЖЕННЯ ФУНКЦІОНАЛЬНИХ МОЖЛИВОСТЕЙ СУЧАСНИХ МОБІЛЬНИХ ЗАСТОСУНКІВ.....	28
2.1 Дослідження мобільних застосунків для електронних щоденників....	28
2.1.1 Застосунок School Diary.....	29
2.1.2 Застосунок MySchoolApp.....	34
2.1.3 Застосунок ClassDojo.....	37
2.1.4 Застосунок Edmodo.....	40
2.2 Дослідження можливостей інтеграції мобільних застосунків з іншими системами.....	45
3 ПРОЕКТУВАННЯ МОДУЛЯ КОРИСТУВАЧА МОБІЛЬНОГО ЗАСТОСУНКУ АВТОМАТИЗОВАНОЇ СИСТЕМИ ЕЛЕКТРОННОГО ЩОДЕННИКА ДЛЯ ЗАГАЛЬНООСВІТНІХ ШКІЛ.....	47
3.1 Аналіз функцій модуля користувача для учнів та батьків.....	47
3.2 Розробка структури бази даних автоматизованої системи електронного щоденника для загальноосвітніх шкіл.....	48
3.3 Розробка UML-діаграм для автоматизованої системи електронного щоденника для загальноосвітніх шкіл.....	51
3.3.1 Опис Use Case UML-діаграм.....	51
3.3.2 Опис UML-діаграми класів.....	55
3.3.3 Опис UML-діаграми компонентів.....	57
4 РОЗРОБКА МОДУЛЯ КОРИСТУВАЧА МОБІЛЬНОГО ЗАСТОСУНКУ АВТОМАТИЗОВАНОЇ СИСТЕМИ ЕЛЕКТРОННОГО ЩОДЕННИКА ДЛЯ ЗАГАЛЬНООСВІТНІХ ШКІЛ.....	59
4.1 Вибір технологій для розробки модуля користувача.....	59
4.2 Розробка програмних компонентів модуля користувача.....	60

4.3 Налаштування та тестування роботи модуля користувача.....	68
ВИСНОВКИ.....	79
ПЕРЕЛІК ПОСИЛАНЬ.....	80
Додатки.....	85
Додаток А (обов'язковий) Протокол перевірки навчальної (кваліфікаційної) роботи.....	86
Додаток Б (обов'язковий) Технічне завдання.....	87
Додаток В (вибірковий) Лістинг програми.....	90
Додаток Г (обов'язковий) Ілюстративна частина.....	96

ВСТУП

Актуальність. У сучасному світі, де технології розвиваються стрімкими темпами, автоматизація освітніх процесів стає все більш актуальною та необхідною. Особливо це стосується загальноосвітніх шкіл, де впровадження інноваційних технологій може суттєво покращити якість освіти та спростити взаємодію між учасниками навчального процесу. Розробка модуля користувача мобільного застосунку для автоматизованої системи електронного щоденника є одним із кроків на шляху до цифровізації освіти.

Актуальність даної роботи полягає в тому, що вона спрямована на вирішення реальних проблем сучасної освіти, таких як забезпечення доступу до актуальної інформації про навчальний процес для учнів, батьків та вчителів у будь-який час і з будь-якого місця. Це особливо важливо в умовах зростаючого навантаження на учнів та вчителів, а також у контексті поширення дистанційного навчання [1].

Застосування мобільних технологій у освітньому процесі дозволяє забезпечити швидкий доступ до необхідних даних, зробити процес навчання більш гнучким та адаптивним. Мобільний застосунок, як інструмент автоматизації, може значно спростити організацію навчального процесу, автоматизувати рутинні операції, такі як ведення щоденника, розкладу уроків, відстеження відвідуваності та оцінювання успішності учнів.

Однією з ключових переваг мобільного застосунку є його доступність. Учні та їхні батьки можуть переглядати актуальну інформацію про домашні завдання, оцінки, коментарі вчителів, а також отримувати повідомлення про важливі шкільні події. Це сприяє більшій залученості батьків у навчальний процес дітей, що, в свою чергу, може позитивно вплинути на їхню академічну успішність.

Для вчителів мобільний застосунок може стати ефективним інструментом для планування уроків, ведення обліку роботи з учнями, а також для комунікації з батьками. Це дозволяє вчителям більше часу приділяти безпосередньо навчальному процесу, а не рутинній паперовій роботі.

Також важливим аспектом є підвищення ефективності управління освітнім закладом. Автоматизація дозволяє керівництву школи оперативно аналізувати дані про успішність та відвідуваність учнів, що може сприяти своєчасному виявленню та вирішенню проблем.

Враховуючи все вищесказане, розробка модуля користувача мобільного застосунку для автоматизованої системи електронного щоденника є важливим кроком у напрямку підвищення якості освіти та ефективності управління освітніми процесами в загальноосвітніх школах. Це дозволить не тільки оптимізувати роботу вчителів та адміністрації, але й забезпечити більшу залученість учнів та їхніх батьків у навчальний процес, що є ключовим для розвитку сучасної освіти.

Мета і завдання дослідження. Метою даної магістерської кваліфікаційної роботи є розробка модуля користувача мобільного застосунку для автоматизованої системи електронного щоденника в загальноосвітніх школах. Цей модуль має на меті спростити взаємодію між учнями, вчителями та батьками, оптимізувати процеси управління навчальними матеріалами та підвищити ефективність навчального процесу. Робота спрямована на створення інноваційного інструменту, який дозволить забезпечити більшу залученість учасників освітнього процесу та підвищити їхню задоволеність від використання сучасних технологій у навчанні.

Для досягнення поставленої мети були визначені та вирішені наступні завдання:

1) Проведено дослідження та аналіз існуючих систем електронних щоденників в загальноосвітніх школах для визначення їхніх основних функціональних можливостей та недоліків.

2) Досліджено потреби користувачів (учнів, вчителів, батьків) для визначення вимог до нового мобільного застосунку.

3) Аналізовано можливості інтеграції мобільного застосунку з іншими системами та платформами, які використовуються в освітніх установах.

4) Розроблено структуру бази даних та архітектуру модуля користувача мобільного застосунку.

5) Спроектовано та реалізовано інтерфейс користувача з урахуванням зручності та інтуїтивності використання.

6) Проведено тестування розробленого модуля на зручність використання та функціональність.

7) Аналізовано отримані результати та внесено необхідні корективи для покращення роботи модуля користувача.

Об'єктом дослідження є процес автоматизації навчального процесу в загальноосвітніх школах за допомогою мобільного застосунку.

Предметом дослідження є методи та засоби програмування, які використовуються для розробки модуля користувача мобільного застосунку для автоматизованої системи електронного щоденника в загальноосвітніх школах.

Методи дослідження. У процесі дослідження використовувалися наступні методи та підходи: об'єктно-орієнтоване програмування, Agile методології розробки програмного забезпечення, в тому числі Scrum і Kanban, методи проектування баз даних, техніки розробки користувацького інтерфейсу, а також методи забезпечення безпеки інформації та захисту даних.

Наукова новизна одержаних результатів:

1. Проведено аналіз існуючих мобільних застосунків для електронних щоденників в загальноосвітніх школах та виявлено ключові недоліки та обмеження, що дозволило сформулювати вдосконалені вимоги до нового модуля користувача.
2. Розроблено модуль користувача мобільного застосунку, який може інтегруватися з головною системою електронного щоденника і включає розширені функціональні можливості для управління навчальними матеріалами, комунікації між учасниками освітнього процесу та персоналізації досвіду користувачів.
3. Впроваджено новітні методи оптимізації взаємодії з базою даних, що забезпечують високу швидкість обробки запитів та ефективність синхронізації даних між різними користувачами системи, що є критично важливим для мобільних застосунків, які використовуються в освітньому процесі.

Практичне значення одержаних результатів полягає в тому, що на основі отриманих теоретичних положень виконано розробку та впровадження модуля користувача мобільного застосунку для автоматизованої системи електронного щоденника в загальноосвітніх школах. Це дозволяє значно покращити взаємодію між учнями, вчителями та батьками, оптимізувати управління навчальним процесом та підвищити загальну ефективність освітнього середовища.

Достовірність теоретичних положень магістерської кваліфікаційної роботи підтверджується строгістю постановки задач, коректним застосуванням усіх методів та засобів програмування, використанням сучасних підходів до розробки мобільних застосунків, включаючи Agile методику та принципи Scrum при проведенні розробки та впровадження модуля користувача.

Особистий внесок здобувача. Усі результати отримано автором самостійно, від концепції та аналізу існуючих рішень до проектування, реалізації та тестування кінцевого продукту.

Апробація результатів дослідження: Представлені в роботі результати апробовані в конференції «Молодь в науці: дослідження, проблеми, перспективи (МН-2025)». За результатами виконаних досліджень підготовлено та подано тези доповіді. Розробка модуля користувача мобільного застосунку автоматизованої системи електронного щоденника для загальноосвітніх шкіл.

1 ДОСЛІДЖЕННЯ ТА АНАЛІЗ РОБОТИ СИСТЕМ УПРАВЛІННЯ НАВЧАЛЬНИМ ПРОЦЕСОМ В ЗАГАЛЬНООСВІТНІХ ШКОЛАХ

1.1 Дослідження об'єкта автоматизації

Темою магістерської кваліфікаційної роботи є розробка модуля користувача мобільного застосунку для автоматизованої системи електронного щоденника в загальноосвітніх школах. В рамках дослідження актуальності теми та планування етапів виконання роботи необхідно дослідити особливості існуючих електронних щоденників, варіанти підключення та налаштування бази даних, можливості інтеграції з іншими освітніми системами, особливості розробки інтерфейсу для мобільних застосунків, типи сповіщень та розсилки інформації, відслідковування та аналіз даних, реалізацію технічної підтримки та безпеки мобільного застосунку.

Аналіз кожного етапу допоможе збудувати повне бачення готового програмного продукту та його функціональності, розробити на основі цього необхідні UML та ER діаграми, а також визначитися з необхідним програмним забезпеченням. Особлива увага буде приділена забезпеченню зручності користування застосунком для всіх учасників освітнього процесу, включаючи учнів, вчителів та батьків, а також забезпеченню високого рівня безпеки та конфіденційності оброблюваних даних.

Місце задачі, що розв'язується в роботі, у загальному контексті автоматизації систем управління освітніми процесами є ключовим для підвищення ефективності та якості освіти в загальноосвітніх школах. Розробка модуля користувача мобільного застосунку для автоматизованої системи електронного щоденника вирішує важливі задачі, пов'язані з оптимізацією комунікацій між учасниками освітнього процесу та управлінням навчальними ресурсами.

Однією з основних задач є забезпечення зручного та швидкого доступу до навчальних матеріалів, розкладів, оцінок та іншої важливої інформації для учнів, вчителів та батьків. Це сприяє підвищенню залученості учнів до

навчального процесу та дозволяє батькам бути в курсі академічних досягнень та проблем своїх дітей.

Крім того, автоматизована система дозволяє вчителям ефективніше планувати свою роботу, автоматизувати процеси виставлення оцінок та ведення щоденника, що знижує адміністративне навантаження та звільняє час для більш якісної підготовки до уроків і взаємодії з учнями.

Також, система сприяє підвищенню прозорості та об'єктивності оцінювання, забезпечуючи точне та своєчасне відображення результатів учнів, що є важливим для справедливості та мотивації учнів.

Усі ці функції та можливості роблять автоматизовану систему невід'ємною складовою сучасної освітньої установи, забезпечуючи зручність, ефективність та надійність управління освітнім процесом, а також оптимальне використання ресурсів і підвищення якості освіти.

Область застосування розробки

Розроблений модуль користувача для мобільного додатку автоматизованої системи електронного щоденника відкриває широкі можливості для використання у секторі освіти. Він є незамінним інструментом для будь-якої загальноосвітньої установи, включаючи школи, гімназії, ліцеї та коледжі, незалежно від їхнього розміру чи спеціалізації. Цей застосунок може бути ефективно імплементований у різноманітні освітні заклади.

Окрім освітньої сфери, ця автоматизована система також може бути корисною в інших областях, де критично важливим є ефективне управління інформацією та забезпечення легкого доступу до неї. Це може включати вищі навчальні заклади, такі як університети та академії, а також освітні центри, де необхідно організувати ефективний обмін даними між студентами, викладачами та адміністрацією.

Універсальність та адаптивність цієї системи дозволяє їй знаходити застосування в різних освітніх контекстах, сприяючи оптимізації навчальних процесів та покращенню комунікації між учасниками освітнього процесу. Мобільний додаток сприяє збільшенню залученості учнів, підвищує

ефективність навчання та спрощує управління адміністративними завданнями, що є важливим для сучасних освітніх установ.

Вимоги до програмно-технічного забезпечення, що розробляється в магістерській кваліфікаційній роботі, включають наступні компоненти:

- Локальна база даних: Для зберігання даних безпосередньо на мобільному пристрої користувача буде використано SQLite. Це дозволить забезпечити швидкий доступ до даних навіть без підключення до інтернету та підвищить загальну продуктивність застосунку.
- Веб-інтерфейс: Для розробки веб-інтерфейсу, який дозволить користувачам ефективно управляти мобільним застосунком через веб, буде використано React для клієнтської частини та Node.js для серверної частини. Це забезпечить адаптивність та сумісність з усіма сучасними браузерами.
- Серверні рішення: Для обробки запитів від мобільних та веб-клієнтів буде використано хмарну платформу AWS. Це забезпечить високу доступність, масштабованість та безпеку системи. Буде налаштовано балансування навантаження та автоматичне масштабування для оптимізації ресурсів.
- Засоби безпеки: Для захисту даних буде використано шифрування на стороні клієнта та сервера. Всі дані, що передаються, будуть захищені за допомогою HTTPS, а аутентифікація користувачів буде здійснюватися через OAuth. Регулярний аудит безпеки та оновлення заходів захисту також будуть включені в план технічного обслуговування.

Функції розробки:

- надання доступу користувачам до системи та їхніх індивідуальних профілів;
- відображення розкладу занять;
- перегляд оцінок;

- ознайомлення з домашніми завданнями та доступ до навчальних ресурсів.

Для виконання роботи необхідні наступні **вхідні дані**:

- локальна база даних, структура якої визначена за допомогою ег-діаграми;
- налаштовані програмні та технічні ресурси, необхідні для створення та запуску автоматизованої системи управління електронним щоденником у навчальному закладі.

Перевірка розробленої та впровадженої автоматизованої системи управління електронним щоденником для освітніх установ вимагатиме аналізу результатів роботи готового продукту, коректної взаємодії його компонентів та команд, а також правильного відображення вихідних даних.

1.2 Дослідження можливостей різних ролей користувачів у електронному щоденнику

Електронний щоденник є важливим інструментом в сучасній освітній системі, що дозволяє оптимізувати процеси управління навчальним процесом та забезпечує ефективну взаємодію між усіма учасниками освітнього процесу. Використання електронних щоденників сприяє підвищенню прозорості та доступності інформації, що є ключовим для розвитку інформаційного суспільства.

У рамках електронного щоденника можна виділити кілька основних ролей, що зображені на рисунку 1.1.



Рисунок 1.1 – Основні ролі користувачів автоматизованою системою електронного щоденника

Головний адміністратор - особа, яка відповідає за налаштування та підтримку системи на вищому рівні. Головний адміністратор керує доступом до різних модулів щоденника, встановлює правила користування системою та забезпечує її безпеку.

Вчителі - користувачі, які вносять основний обсяг навчальної інформації: розклад занять, домашні завдання, оцінки та відгуки про успішність учнів.

Батьки - активні учасники освітнього процесу, які використовують систему для моніторингу академічного прогресу та шкільного життя своїх дітей.

Учні - основні бенефіціари, для яких система надає можливість переглядати розклад, домашні завдання, оцінки та отримувати зворотній зв'язок від вчителів.

Функції електронного щоденника для адміністратора та вчителів відіграють ключову роль у забезпеченні ефективності та гладкості функціонування освітнього процесу [2, 3].

Для адміністратора:

- Керування доступом користувачів: Адміністратор відповідає за створення, редагування та видалення користувацьких акаунтів, а також за призначення рівнів доступу для вчителів, учнів та батьків.
- Налаштування системи: Адміністратор має можливість налаштовувати параметри системи, включаючи шкільний календар, розклад занять та інші важливі елементи.
- Моніторинг активності в системі: Адміністратор має інструменти для відстеження активності користувачів, що дозволяє забезпечити безпеку та ефективність використання системи.
- Резервне копіювання та відновлення даних: Важлива функція, яка забезпечує захист інформації від випадкових втрат або технічних збоїв.

Для вчителів:

- Внесення навчальної інформації: Вчителі використовують систему для розміщення домашніх завдань, оцінок та коментарів до робіт учнів, що сприяє більшій організованості та доступності інформації.
- Планування уроків: Електронний щоденник може містити інструменти для планування уроків, що дозволяє вчителям ефективно розподіляти навчальний матеріал та час.

- Відстеження відвідуваності: Вчителі можуть легко вести облік відвідуваності учнів, що є важливим аспектом для аналізу їхньої активності та успішності.
- Комунікація з батьками: Система може надавати інструменти для спілкування з батьками, що дозволяє вчителям швидко передавати інформацію про успіхи та проблеми учнів.

Ключові функції електронного щоденника для учнів та батьків включають:

- Перегляд розкладу занять, що дозволяє учням планувати свій навчальний день, а батькам - бути в курсі шкільних справ своїх дітей.
- Моніторинг виконання домашніх завдань, що сприяє кращій організації навчального процесу та взаємодії між учнями та вчителями.
- Перегляд оцінок та аналіз академічного прогресу, що допомагає батькам відстежувати успіхи та труднощі своїх дітей у навчанні.
- Отримання повідомлень та сповіщень про важливі шкільні події, збори, зміни в розкладі тощо.

Використання електронного щоденника значно спрощує доступ до важливої інформації та оптимізує комунікацію між усіма учасниками навчального процесу, що, в свою чергу, підвищує ефективність освітнього процесу та сприяє розвитку освітнього середовища.

З огляду на важливість і різноманітність функцій, які виконують різні ролі користувачів у електронному щоденнику, наступним кроком є детальний аналіз механізмів взаємодії між цими ролями, що дозволить зрозуміти, як система сприяє ефективній комунікації та координації дій учасників освітнього процесу.

1.3 Аналіз механізмів взаємодії різних ролей користувачів через електронний щоденник

У контексті сучасних освітніх технологій, електронний щоденник виступає не лише як інструмент для ведення академічної інформації, але й як

платформа для мультидисциплінарної взаємодії між усіма учасниками навчального процесу. Взаємодія між різними ролями користувачів, такими як адміністратори, вчителі, учні та їхні батьки, є критично важливою для забезпечення ефективності та прозорості освітнього процесу. Ця взаємодія дозволяє не тільки оптимізувати навчальний процес, але й сприяє більшій залученості та мотивації учнів, а також забезпечує батьків актуальною інформацією про навчальні досягнення та потреби їхніх дітей [4].

Мета даного аналізу полягає у детальному розгляді механізмів взаємодії, які використовуються в електронних щоденниках, для ідентифікації як ефективних практик, так і потенційних областей для покращення. Через це дослідження визначається, як різні функціональні можливості системи сприяють або, навпаки, ускладнюють процес взаємодії між користувачами, та як це впливає на загальну продуктивність та задоволеність користувачів системою [5].

Перш за все, можна виділити наступні механізми взаємодії між усіма користувачами системи електронного щоденник.

- Комунікаційні інструменти в системі:

Електронний щоденник включає різноманітні комунікаційні інструменти, що сприяють ефективному обміну інформацією між учасниками освітнього процесу. Повідомлення, електронні листи та форуми є основними засобами для цього. Вони дозволяють вчителям швидко передавати інформацію учням та їхнім батькам, а також забезпечують платформу для дискусій та зворотного зв'язку. Ці інструменти також сприяють залученню батьків у навчальний процес, дозволяючи їм стежити за академічними досягненнями та шкільним життям своїх дітей в режимі реального часу.

- Спільний доступ до інформації:

Система електронного щоденника забезпечує спільний доступ до важливої навчальної інформації, такої як розклади, оцінки та домашні завдання. Це дозволяє учням планувати свій навчальний час, а вчителям - ефективно управляти навчальним процесом. Батьки можуть використовувати цю інформацію для моніторингу навчального прогресу своїх дітей і вчасно

реагувати на будь-які академічні проблеми. Такий спільний доступ сприяє прозорості та відкритості в освітньому процесі, зміцнюючи довіру та співпрацю між всіма зацікавленими сторонами.

– Функції контролю та звітності:

Для адміністраторів та вчителів електронний щоденник пропонує розширені можливості контролю та звітності. Адміністратори можуть використовувати інструменти для моніторингу використання системи, забезпечення її безпеки та відстеження активності користувачів. Вчителі мають можливість створювати звіти про успішність та відвідуваність учнів, що є важливим для оцінки ефективності навчальних методик та індивідуального підходу до кожного учня. Ці функції допомагають управлінню школою та вчителям підтримувати високі стандарти освіти та забезпечувати відповідність до сучасних освітніх вимог [6].

Ефективність взаємодії між різними користувачами електронного щоденника є ключовим фактором, який впливає на загальну продуктивність та успішність освітнього процесу. Для глибшого розуміння цього аспекту, необхідно провести аналіз зручності інтерфейсу, швидкості та ефективності комунікації між ролями, а також оцінити вплив системи на загальну продуктивність навчання.

– Оцінка зручності інтерфейсу для різних користувачів:

Зручність інтерфейсу є вирішальною для забезпечення легкості доступу та використання системи всіма учасниками освітнього процесу. Інтерфейс повинен бути інтуїтивно зрозумілим для учнів, вчителів, батьків та адміністраторів, забезпечуючи легке навігацію та доступ до необхідних функцій. Важливо, щоб інтерфейс адаптувався до різних потреб користувачів, що може включати персоналізацію налаштувань та можливість доступу до різноманітних інформаційних ресурсів.

– Аналіз швидкості та ефективності комунікації між ролями:

Швидкість та ефективність комунікації в системі електронного щоденника безпосередньо впливають на оперативність обміну інформацією та здатність

швидко реагувати на зміни в навчальному процесі. Важливо оцінити, як система сприяє вчасному та точному передаванню інформації між учителями, учнями та батьками, а також як вона допомагає уникнути непорозумінь та помилок.

– Вплив системи на загальну продуктивність навчального процесу:

Кінцева мета використання електронного щоденника полягає у підвищенні продуктивності навчального процесу. Аналіз впливу системи на навчальні результати, залученість учнів та задоволеність вчителів та батьків дозволить визначити, наскільки ефективно система виконує свої функції. Зокрема, важливо з'ясувати, чи сприяє система підвищенню мотивації учнів, чи забезпечує вона ефективне управління ресурсами та часом, а також чи допомагає вона вчителям у плануванні та проведенні уроків [7].

Під час аналізу взаємодії користувачів через електронний щоденник було виявлено декілька проблемних аспектів, які можуть ускладнювати використання системи та знижувати її ефективність. Однією з основних проблем є недостатня інтуїтивність інтерфейсу для деяких користувачів, що може призводити до труднощів у навігації та доступу до необхідних функцій. Також було зазначено, що швидкість обробки даних іноді не відповідає потребам користувачів, що може спричинити затримки у комунікації та обміні інформацією.

Для покращення механізмів взаємодії пропонується оптимізувати інтерфейс, зробивши його більш адаптивним та зручним для різних груп користувачів. Це може включати персоналізацію налаштувань залежно від ролі та потреб користувача. Також важливим є підвищення швидкості обробки даних, що можна досягти за допомогою технічного удосконалення серверних рішень та оптимізації алгоритмів обробки запитів.

Загальний аналіз механізмів взаємодії різних ролей користувачів через електронний щоденник показав, що система має значний потенціал для підвищення продуктивності навчального процесу. Однак для досягнення максимальної ефективності необхідно вирішити ряд існуючих проблем, зокрема у сферах інтерфейсу та швидкості обробки інформації [8, 9].

Рекомендується провести додаткові дослідження для визначення конкретних потреб користувачів та розробити на їх основі цілеспрямовані удосконалення системи. Також важливо забезпечити регулярне оновлення програмного забезпечення для відповідності сучасним технологічним стандартам та забезпечення високої продуктивності системи. Ці кроки допоможуть зробити електронний щоденник більш ефективним інструментом для всіх учасників освітнього процесу.

1.4 Аналіз ринку IT-послуг для освітніх установ

У сучасному освітньому середовищі IT-послуги відіграють ключову роль, забезпечуючи не тільки ефективність навчальних процесів, але й доступність та адаптивність освіти. Вони дозволяють освітнім установам інтегрувати новітні технології, що сприяє підвищенню якості навчання та залученню студентів. Мета цього аналізу полягає у вивченні ринку IT-послуг для освітніх установ, щоб ідентифікувати основні тенденції, виклики та можливості, які можуть вплинути на стратегічне планування та розвиток освітніх технологій.

Огляд ринку IT-послуг для освітніх установ виявляє динамічний ландшафт, де різноманітні компанії пропонують широкий спектр технологічних рішень. Серед основних гравців на цьому ринку можна виділити такі компанії, як Google, Microsoft та Apple, які надають комплексні освітні платформи та інструменти. Також існують спеціалізовані провайдери, які фокусуються на конкретних аспектах освітніх технологій, наприклад, Blackboard та Canvas, що пропонують системи управління навчанням.

Щодо типів послуг, то вони варіюються від хмарних рішень, що дозволяють зберігати та обробляти великі обсяги даних, до систем управління навчанням, які спрощують створення, розподіл та управління навчальними матеріалами. Крім того, велику увагу приділяється безпеці даних, оскільки освітні установи зберігають велику кількість конфіденційної інформації.

Тенденції на ринку IT-послуг для освіти включають зростаючу інтеграцію штучного інтелекту та машинного навчання, що дозволяє створювати

персоналізовані навчальні уроки та оптимізувати процеси оцінювання та відстеження прогресу студентів. Інновації також спостерігаються у використанні віртуальної та доповненої реальності для створення іммерсивних навчальних середовищ, що забезпечують більш глибоке занурення у навчальний матеріал та підвищують залученість студентів [10].

Аналіз попиту та пропозицій на ринку IT-послуг для освітніх установ виявляє ключові потреби, які формують запити до технологічних рішень. Освітні установи шукають ефективні способи для підвищення якості навчання, забезпечення доступності освіти та управління великими обсягами даних. Важливими є також питання безпеки інформації та забезпечення стабільної роботи IT-систем.

Потреби освітніх установ можна поділити на кілька основних категорій, що зображені на рисунку 2.2, а саме:

1. Інфраструктурні рішення, які забезпечують надійне зберігання та обробку даних.
2. Платформи для дистанційного навчання, які дозволяють організувати ефективний освітній процес в онлайн-форматі.
3. Інструменти для аналізу та оцінки результатів навчання, що сприяють персоналізації освітнього процесу.
4. Рішення для забезпечення безпеки даних та захисту від кіберзагроз.



Рисунок 2.2 – Категорії потреб освітніх установ

Аналіз пропозицій на ринку показує, що багато ІТ-компаній пропонують комплексні рішення, які відповідають основним потребам освітніх установ. Проте, деякі специфічні вимоги, такі як інструменти для спеціалізованого аналізу навчальних даних або рішення для інтеграції різноманітних освітніх ресурсів, можуть бути недостатньо представлені. Це відкриває можливості для розвитку нових продуктів та послуг, які більш точно відповідають потребам освітнього сектору.

Впровадження ІТ-послуг у освітніх установах супроводжується рядом викликів, які можуть ускладнити процес інтеграції нових технологій. Одним з основних викликів є обмежений бюджет, який часто зустрічається в освітніх установах, особливо в публічному секторі. Це може обмежувати можливості закупівлі сучасного обладнання та ліцензій на програмне забезпечення. Іншим значним викликом є необхідність підготовки та навчання персоналу, щоб вони могли ефективно використовувати нові ІТ-рішення. Крім того, важливим є питання кібербезпеки, оскільки освітні установи стають цілями для кібератак, що може призвести до втрати важливих даних та порушення приватності [11].

Попри виклики, ринок ІТ-послуг для освіти пропонує значні можливості для розвитку. Зростаюча потреба в дистанційному навчанні, особливо в контексті глобальних подій, таких як пандемія COVID-19, стимулює попит на ефективні онлайн-платформи та інструменти для віртуального навчання. Це відкриває двері для розробки нових продуктів, які можуть забезпечити більш гнучкі та доступні освітні рішення. Крім того, інтеграція штучного інтелекту та аналітики даних може допомогти освітнім установам краще розуміти потреби учнів та адаптувати навчальні матеріали для підвищення ефективності навчання. Також існує потенціал для розвитку інклюзивних технологій, які можуть зробити освіту доступною для учнів з обмеженими можливостями.

Одним із прикладів успішного впровадження ІТ-послуг у освітній сфері є використання хмарних платформ, таких як Google Classroom, у школах для організації навчального процесу. Це дозволило вчителям та учням ефективно взаємодіяти, обмінюватися матеріалами та виконувати завдання в режимі онлайн. Аналіз ефективності таких рішень показав збільшення залученості учнів, покращення їхніх академічних результатів та зниження відсотка пропусків занять.

Інший кейс пов'язаний з впровадженням систем управління навчанням, які інтегрують штучний інтелект для адаптації навчальних матеріалів під індивідуальні потреби учнів. Школи, які впровадили такі системи, відзначають підвищення мотивації учнів та їхнє краще розуміння складних тем завдяки персоналізованому підходу [12].

Підсумовуючи, аналіз ринку ІТ-послуг для освітніх установ підтвердив значний потенціал для впровадження новітніх технологій, які можуть радикально трансформувати навчальний процес. Від хмарних рішень до систем управління навчанням та інтеграції штучного інтелекту – всі ці інновації сприяють підвищенню ефективності та доступності освіти.

Рекомендації для освітніх установ щодо вибору та впровадження ІТ-послуг включають:

1. Оцінку власних потреб та ресурсів перед вибором конкретних ІТ-послуг.

2. Вибір надійних постачальників з доведеним досвідом роботи в освітній сфері.

3. Проведення пілотних проектів перед повномасштабним впровадженням нових технологій.

4. Навчання персоналу для ефективного використання нових ІТ-рішень.

5. Регулярний моніторинг та аналіз впливу впроваджених ІТ-послуг на навчальний процес.

За допомогою цих кроків освітні установи зможуть максимально використати потенціал ІТ-послуг для підвищення якості та ефективності освіти.

1.5 Аналіз потреб користувачів в автоматизації навчального процесу

Автоматизація навчального процесу стає все більш, оскільки вона дозволяє оптимізувати рутинні задачі, підвищити ефективність викладання та навчання, а також забезпечити більшу доступність освітніх ресурсів для усіх учасників освітнього процесу. Автоматизація допомагає вчителям краще планувати уроки, відстежувати прогрес учнів та індивідуалізувати навчання відповідно до потреб кожного учня. Для учнів це означає можливість доступу до навчальних матеріалів в будь-який час і з будь-якого місця, що сприяє самостійному навчанню та розвитку. Адміністрація шкіл отримує інструменти для ефективнішого управління ресурсами та планування освітнього процесу, а батьки можуть бути більш інформованими про успіхи та проблеми своїх дітей [13-15].

Аналіз специфічних потреб кожної групи користувачів:

1. Учні:

- Інтерактивні та зрозумілі навчальні матеріали.
- Можливість доступу до домашніх завдань та ресурсів в онлайн-режимі.
- Інструменти для самостійного контролю знань та самооцінки.
- Платформи для спілкування з однокласниками та вчителями.

2. Вчителі:

- Автоматизація адміністративних задач, таких як ведення журналів, планування уроків.
- Інструменти для моніторингу прогресу учнів та аналітики результатів.
- Ресурси для розробки індивідуалізованих навчальних планів.
- Платформи для ефективного спілкування з учнями та їхніми батьками.

3. Адміністрація шкіл:

- Системи для управління ресурсами школи (наприклад, інвентар, фінанси).
- Інструменти для планування та аналізу освітніх програм.
- Рішення для забезпечення безпеки даних та захисту інформації.
- Модулі для звітності та відстеження виконання освітніх стандартів.

4. Батьки:

- Доступ до актуальної інформації про успіхи та відвідуваність дітей.
- Можливість спілкування з вчителями та адміністрацією школи.
- Інструменти для моніторингу та підтримки навчального процесу дітей вдома.

Автоматизація навчального процесу включає в себе впровадження різноманітних технологічних рішень, які спрямовані на підвищення ефективності та зручності освітньої діяльності. До основних потреб у автоматизації відносяться:

- Автоматизація розкладу уроків: Цифрові системи, які дозволяють оптимізувати планування шкільного розкладу, враховуючи доступність вчителів та класних кімнат.
- Електронні журнали: Заміна традиційних паперових журналів на електронні дозволяє вчителям швидше вносити оцінки та відвідуваність, а також забезпечує батьків актуальною інформацією про успіхи дітей.
- Системи управління навчанням (LMS): Ці системи інтегрують усі аспекти навчального процесу, включаючи розподіл матеріалів, тести,

обговорення та зворотний зв'язок, що сприяє більшій залученості та індивідуалізації навчання.

Кожна група користувачів має свої специфічні потреби в автоматизації:

- Учні потребують інтуїтивно зрозумілих інтерфейсів для доступу до навчальних матеріалів та виконання завдань.
- Вчителі шукають рішення, які мінімізують рутинну роботу та забезпечують ефективне управління класом.
- Адміністрація зацікавлена в системах для керування ресурсами, планування та звітності.
- Батьки вимагають прозорості та доступності інформації про навчальний процес та успіхи своїх дітей.

Потреби у автоматизації навчального процесу тісно пов'язані з різними аспектами ІТ-послуг, які були розглянуті в підрозділах 2.2 та 2.3. Зокрема, ключовими аспектами є кібербезпека, хмарні рішення та мобільний доступ.

Кібербезпека є критично важливою, оскільки системи, що автоматизують навчальний процес, часто містять конфіденційну інформацію про учнів та вчителів. Захист цих даних від несанкціонованого доступу та кібератак є пріоритетом. Наприклад, електронні журнали та системи управління навчанням повинні використовувати шифрування даних та багаторівневі системи аутентифікації для забезпечення безпеки.

Хмарні рішення дозволяють забезпечити гнучкість та доступність навчальних ресурсів. Вони дозволяють учням та вчителям доступатися до навчальних матеріалів та систем управління з будь-якого пристрою та з будь-якої точки світу, що важливо для сучасних освітніх процесів. Хмарні рішення також спрощують інтеграцію різних систем та додатків, що покращує загальну ефективність автоматизації.

Мобільний доступ забезпечує учасникам освітнього процесу можливість використовувати освітні ресурси на мобільних пристроях, що значно підвищує зручність та доступність навчання. Це особливо важливо для сучасних учнів, які активно використовують смартфони та інші мобільні пристрої.

При цьому користувачі часто стикаються з рядом проблем при використанні існуючих автоматизованих систем:

- Недостатня інтуїтивність інтерфейсів: Багато систем мають складні та не інтуїтивні інтерфейси, що ускладнює їх використання, особливо для нових користувачів.
- Низька швидкість роботи систем: Велика кількість даних та не оптимізоване програмне забезпечення можуть призвести до зниження продуктивності систем.
- Відсутність необхідних функціональностей: Часто системи не відповідають специфічним потребам користувачів, що обмежує їхню ефективність.
- Проблеми з інтеграцією різних систем: Інтеграція різних ІТ-систем може бути складною задачею, що призводить до технічних проблем та збоїв у роботі.
- Впровадження нових автоматизованих рішень може супроводжуватися наступними викликами:
- Опір змінам з боку користувачів: Багато користувачів можуть бути звикли до старих систем і сприймати нововведення з недовірою.
- Потреба в навчанні та адаптації персоналу: Для ефективного використання нових систем необхідно провести навчання та адаптацію користувачів.
- Високі вимоги до кібербезпеки: Нові системи повинні відповідати сучасним стандартам кібербезпеки, що може вимагати значних інвестицій [16-17].

Забезпечення належної підтримки та тренінгів є ключовим для забезпечення гладкого переходу до нових систем та мінімізації викликів, пов'язаних з їх впровадженням.

1.6 Дослідження існуючих систем електронних щоденників в школах та уточнена постановка задачі

Для забезпечення ефективної автоматизації навчального процесу важливо розуміти, які системи електронних щоденників вже існують на ринку, та як вони впливають на діяльність шкіл. Зазвичай системи електронних щоденників включають такі функції:

- Реєстрація оцінок та відвідуваності: Вчителі можуть швидко вносити дані про успішність та відвідуваність учнів.
- Комунікація з батьками: Системи дозволяють вчителям легко спілкуватися з батьками, інформуючи їх про успіхи та проблеми їхніх дітей.
- Планування уроків: Вчителі можуть використовувати електронні щоденники для планування уроків та розподілу навчальних матеріалів.
- Аналітика та звітність: Адміністрація школи може отримувати звіти про загальну успішність та відвідуваність, що допомагає в аналізі ефективності навчального процесу.

Проте попри наявність численних функцій, такі системи часто мають ряд недоліків:

- Обмежена кастомізація: Багато систем не дозволяють достатньо адаптувати функціонал під специфічні потреби школи або вчителя.
- Проблеми з інтеграцією: Існуючі системи часто важко інтегрувати з іншими IT-рішеннями, що використовуються в школах.
- Користувацький інтерфейс: Інтерфейси деяких систем можуть бути недостатньо інтуїтивно зрозумілими, що ускладнює їх використання [18].

У Розділі 3 буде проведено детальний аналіз функціональних можливостей існуючих систем електронних щоденників. Цей аналіз включатиме оцінку основних характеристик та функцій, які вже реалізовані в наявних рішеннях, а також виявлення їхніх сильних та слабких сторін.

Особлива увага буде приділена аспектам, таким як користувацький інтерфейс, можливості кастомізації, інтеграція з іншими ІТ-рішеннями, а також ефективність комунікаційних та аналітичних функцій.

На основі проведеного дослідження існуючих систем електронних щоденників та аналізу виявлених недоліків, буде розроблено пропозиції щодо створення модуля користувача мобільного застосунку для автоматизованої системи електронного щоденника в загальноосвітніх школах. Цей модуль буде зосереджений на вирішенні ключових проблем, таких як покращення інтерфейсу та розширення можливостей кастомізації. Розробка цього модуля дозволить значно підвищити ефективність навчального процесу в школах, відповідаючи сучасним технологічним вимогам.

2 ДОСЛІДЖЕННЯ ФУНКЦІОНАЛЬНИХ МОЖЛИВОСТЕЙ СУЧАСНИХ МОБІЛЬНИХ ЗАСТОСУНКІВ

2.1 Дослідження мобільних застосунків для електронних щоденників

У цьому підрозділі буде проведено аналіз мобільних застосунків, які використовуються для електронних щоденників в загальноосвітніх школах. Метою аналізу є визначення ключових функціональних можливостей, переваг та недоліків кожного застосунку. Для цього будуть розглянуті наступні застосунки: School Diary, MySchoolApp, ClassDojo та Edmodo.

Кожен з цих застосунків буде оцінено за такими параметрами:

- Інтерфейс користувача: наскільки інтуїтивно зрозумілим та зручним є інтерфейс.
- Функціональність: які основні та додаткові функції пропонує застосунок.
- Інтеграція: можливості інтеграції з іншими ІТ-системами та програмними продуктами.
- Безпека: заходи, що забезпечують захист даних користувачів.
- Відгуки користувачів: загальна реакція користувачів на застосунок, включаючи позитивні та негативні аспекти.

Зібрана інформація буде узагальнена та представлена у вигляді зведеної таблиці, що дозволить легко порівняти різні застосунки між собою за вказаними параметрами. Це допоможе визначити, які застосунки найкраще відповідають потребам сучасних шкіл та які аспекти потребують подальшого вдосконалення.

Детальний аналіз кожного застосунку буде представлено у наступних підрозділах, що допоможе у формуванні вимог до розробки нового модуля мобільного застосунку для електронного щоденника, що буде розроблено в рамках магістерської кваліфікаційної роботи.

2.1.1 Застосунок School Diary

Застосунок School Diary був розроблений компанією "EduTech Solutions", яка спеціалізується на створенні програмного забезпечення для освітніх установ. Компанія була заснована у 2010 році групою ентузіастів у галузі освіти та технологій, які прагнули зробити процес навчання більш доступним та ефективним за допомогою цифрових технологій. Основна ідея створення School Diary полягала у наданні школам інструменту, який би спрощував комунікацію між учителями, учнями та батьками та покращував організацію навчального процесу [19].

School Diary є мобільним застосунком, призначеним для ведення електронного щоденника в школах. Цей застосунок дозволяє вчителям, учням та батькам ефективно взаємодіяти, обмінюватися інформацією про навчальний процес, оцінки та шкільні події.

Детальний аналіз застосунку за параметрами дослідження мобільних застосунків для електронних щоденників, що були описані в підрозділі 3.1:

1. Інтерфейс користувача: School Diary має простий та інтуїтивно зрозумілий інтерфейс, що робить його доступним для користувачів різного віку та технічної підготовки. Інтерфейс оптимізований для мобільних пристроїв, забезпечуючи зручне використання на смартфонах та планшетах.
2. Функціональність: Застосунок надає широкий спектр функцій, включаючи реєстрацію оцінок, відвідуваності, планування уроків, а також можливість відправки повідомлень між вчителями та батьками. Також є функції календаря та нагадувань про важливі шкільні заходи.
3. Інтеграція: School Diary підтримує базову інтеграцію з іншими освітніми та адміністративними системами, що дозволяє обмінюватися даними та підтримувати синхронізацію інформації між різними платформами.

4. Безпека: Застосунок використовує сучасні методи шифрування даних та аутентифікації користувачів, що забезпечує високий рівень захисту персональної та навчальної інформації.
5. Відгуки користувачів: Відгуки про School Diary в цілому позитивні. Користувачі відзначають легкість використання, хорошу функціональність та ефективну підтримку з боку розробників. Проте, деякі користувачі зазначають потребу в додаткових можливостях кастомізації та покращенні інтеграційних функцій.

На рисунках 2.1 – 2.4 представлено деякі функціональні можливості застосунку School Diary.

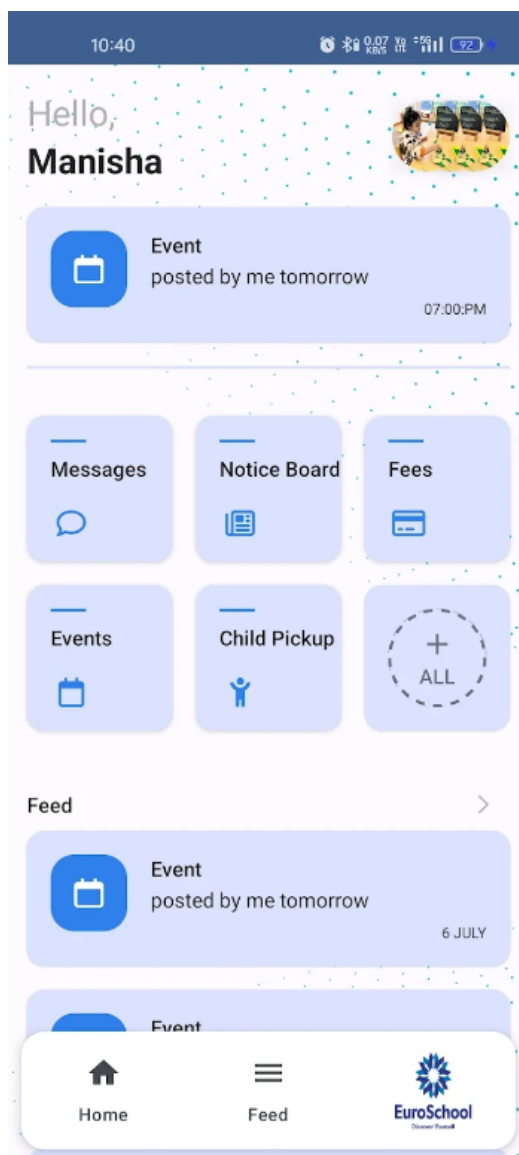


Рисунок 2.1 – Головний екран застосунку School Diary

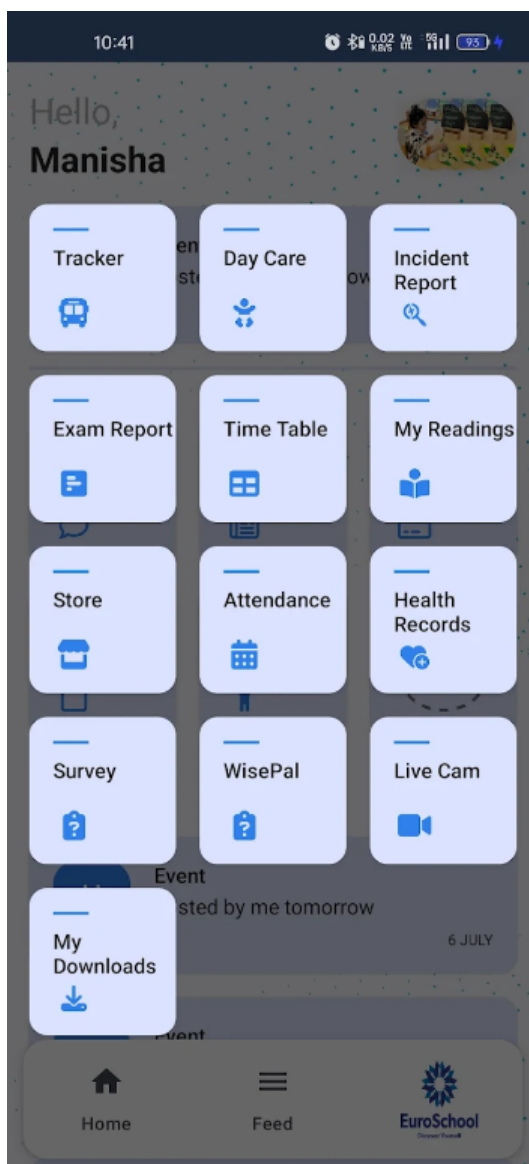


Рисунок 2.2 – Меню доступний функцій для даного користувача в застосунку School Diary

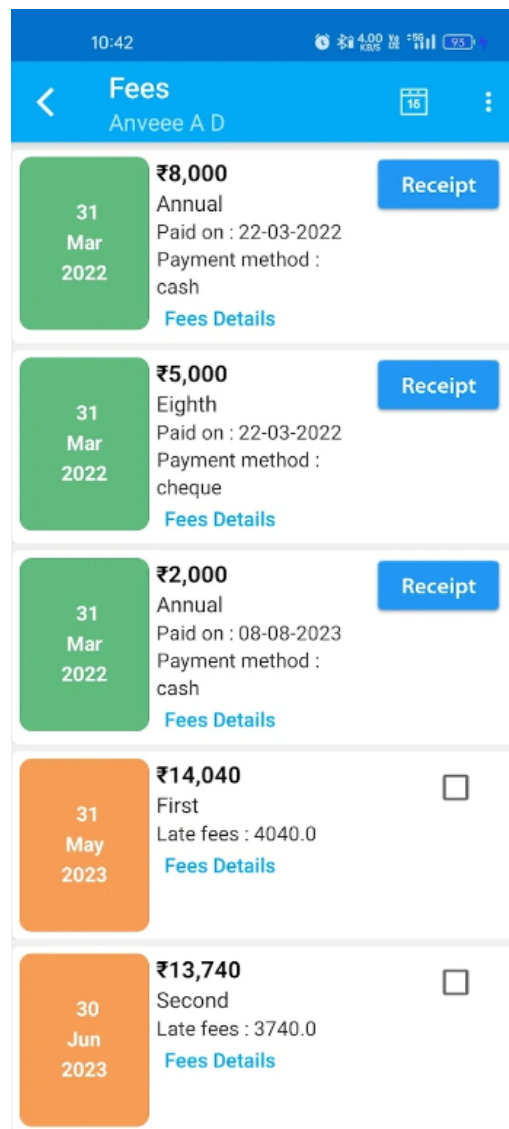


Рисунок 2.3 – Список рахунків до сплати в застосунку School Diary

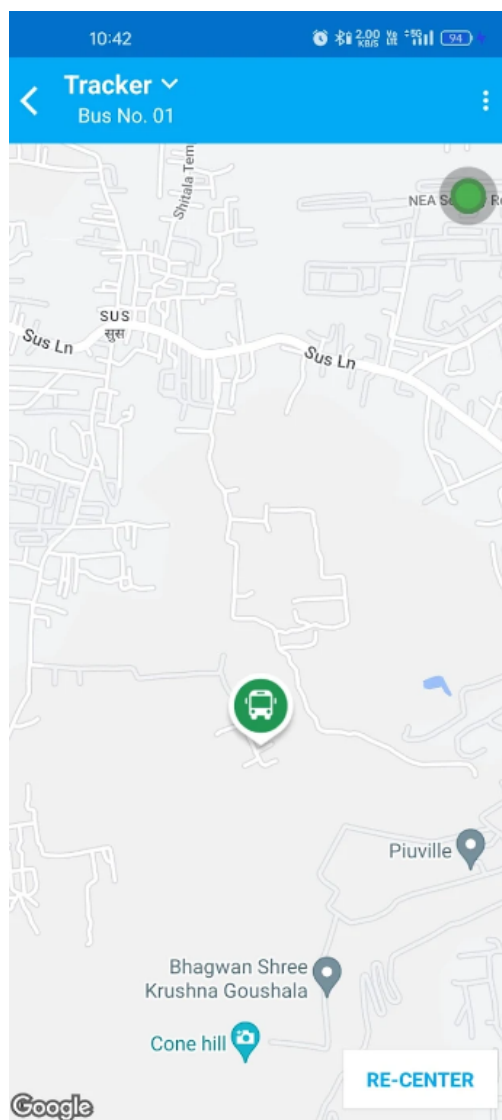


Рисунок 2.4 – Відслідковування шкільного автобуса під час екскурсії в застосунку School Diary

School Diary є платним застосунком, проте він пропонує різні плани підписки, що дозволяє школам обрати найбільш підходящий варіант відповідно до їхніх потреб та бюджету. Також існує можливість безкоштовного пробного періоду, щоб користувачі могли оцінити функціональність застосунку перед придбанням. Фінансування застосунку відбувається за рахунок підписок, а також через партнерські програми з освітніми організаціями та рекламні кампанії.

Крім того, розробники School Diary активно співпрацюють з освітніми установами для збору зворотного зв'язку та вдосконалення свого продукту, що

дозволяє їм постійно оновлювати та покращувати застосунок, враховуючи потреби сучасної освіти.

2.1.2 Застосунок MySchoolApp

MySchoolApp було створено компанією "Smart Education Tech", яка має досвід у розробці інноваційних освітніх технологій. Заснована у 2012 році, компанія має місію забезпечити школи надійними та зручними цифровими рішеннями. MySchoolApp був розроблений з метою оптимізації процесів управління школою та залучення батьків до навчального процесу їхніх дітей [20].

Детальний аналіз за параметрами дослідження мобільних застосунків для електронних щоденників:

1. Інтерфейс користувача: MySchoolApp пропонує користувачам сучасний та чистий дизайн, що забезпечує легке навігування та доступ до всіх необхідних функцій. Інтерфейс добре адаптований до різних типів пристроїв, забезпечуючи високу зручність використання.
2. Функціональність: Застосунок включає комплексні інструменти для управління школою, такі як ведення розкладу, контроль за успішністю учнів, організація шкільних заходів та спілкування з батьками. MySchoolApp також дозволяє вчителям створювати інтерактивні завдання та тести.
3. Інтеграція: Застосунок має вбудовані інструменти для інтеграції з популярними обліковими системами та іншими освітніми платформами, що забезпечує швидкий обмін даними та спрощує адміністрування.
4. Безпека: MySchoolApp використовує передові технології захисту даних, включаючи шифрування на рівні банківських систем, що гарантує безпеку інформації про учнів та їхні сім'ї.

5. Відгуки користувачів: Більшість користувачів високо оцінюють MySchoolApp за його багатофункціональність та зручність у використанні. Однак, деякі вказують на потребу в поліпшенні мобільної оптимізації для деяких старіших моделей пристроїв.

На рисунках 2.5 – 2.8 представлено деякі функціональні можливості застосунку MySchoolApp.

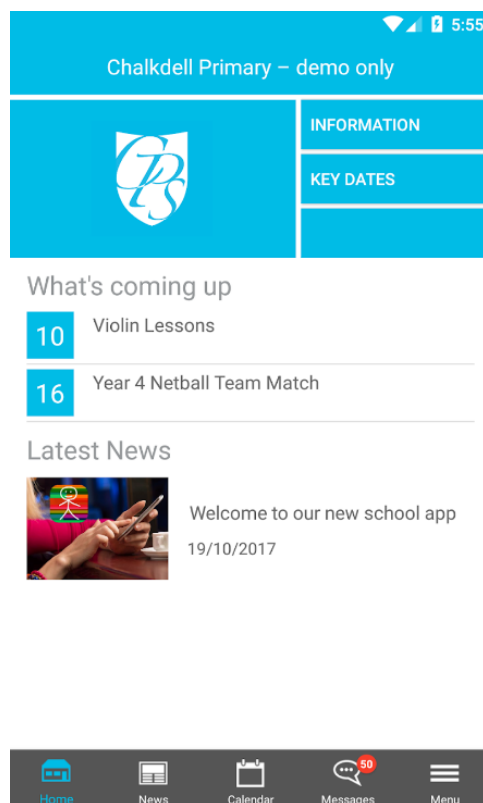


Рисунок 2.5 – Головний екран застосунку MySchoolApp

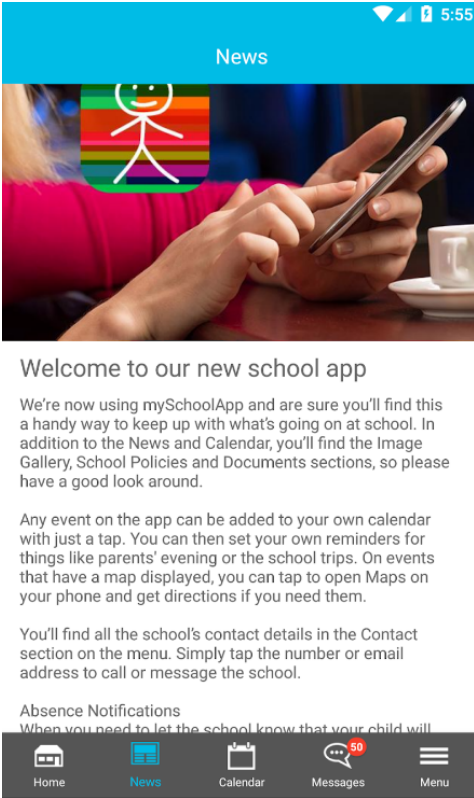


Рисунок 2.6 – Экран новин застусунку MySchoolApp

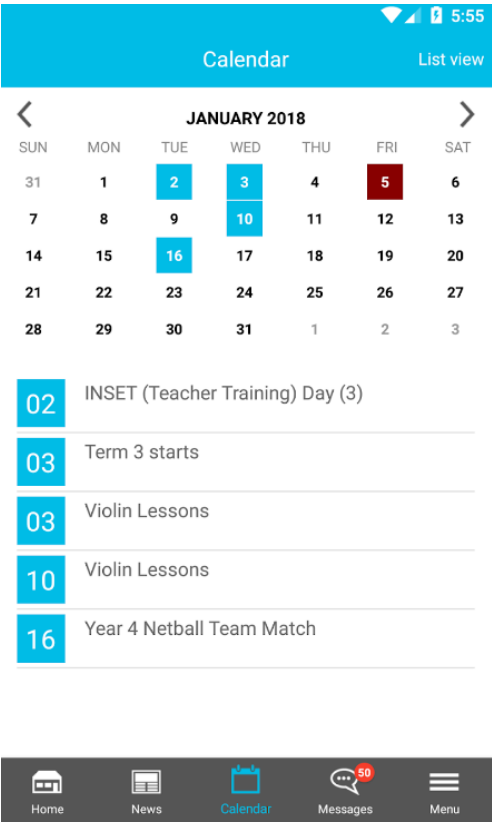


Рисунок 2.7 – Экран з календарем в застосунку MySchoolApp

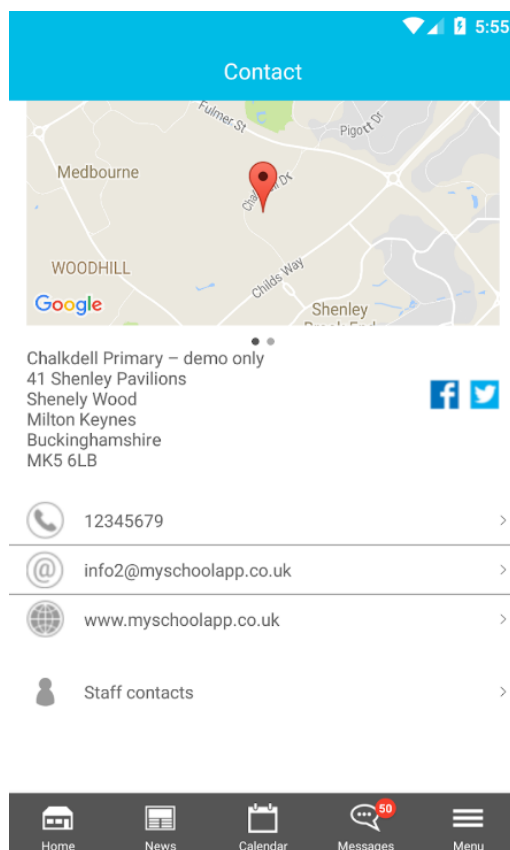


Рисунок 2.8 – Інформація про розробників застосунку MySchoolApp

MySchoolApp пропонує гнучку систему тарифікації з різними рівнями доступу, що дозволяє школам вибирати оптимальний план відповідно до їхніх потреб. Застосунок також пропонує демонстраційний період, щоб користувачі могли оцінити всі переваги перед покупкою. Фінансування здійснюється через прямі продажі ліцензій та партнерські взаємодії з освітніми організаціями.

2.1.3 Застосунок ClassDojo

ClassDojo було розроблено стартапом "ClassDojo Inc." у 2011 році з метою створення позитивної атмосфери в класі та підвищення мотивації учнів через ігрові елементи та взаємодію. Застосунок спрямований на зміцнення спільноти учнів, вчителів та батьків, надаючи інструменти для ефективного спілкування та відстеження прогресу учнів [21].

На рисунках 2.9 та 2.10 представлено деякі функціональні можливості застосунку ClassDojo.

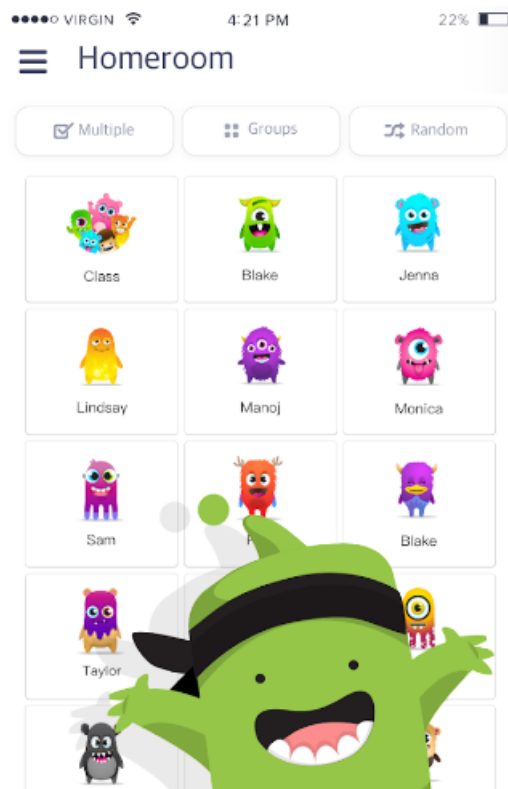


Рисунок 2.9 – Відображення учнів класу в застосунку ClassDojo

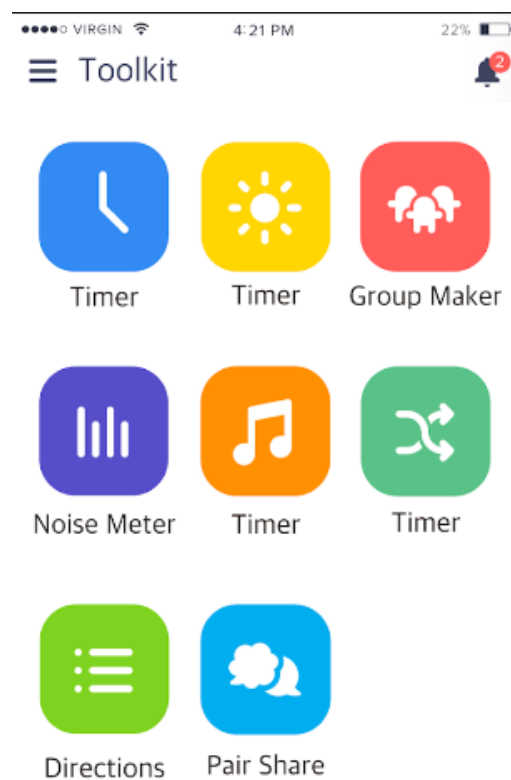


Рисунок 2.10 – Відображення додаткових функцій в застосунку ClassDojo

Детальний аналіз за параметрами дослідження мобільних застосунків для електронних щоденників:

1. Інтерфейс користувача: ClassDojo має яскравий та дружній до користувача інтерфейс, який включає анімаційні елементи та ігрові механіки для залучення учнів. Інтерфейс дизайнований таким чином, щоб бути привабливим для молодших користувачів, але при цьому забезпечувати функціональність, необхідну вчителям та батькам.
2. Функціональність: Застосунок надає можливості для ведення класних журналів, управління поведінкою учнів та спілкування між домом та школою. ClassDojo дозволяє вчителям нагороджувати учнів за успіхи та позитивну поведінку, а також вести діалог з батьками через повідомлення та оновлення.
3. Інтеграція: Застосунок може інтегруватися з різними платформами для електронного навчання та управління школою, дозволяючи обмінюватися даними та забезпечувати цілісний підхід до освітнього процесу.
4. Безпека: ClassDojo використовує передові протоколи безпеки для захисту інформації та даних користувачів, забезпечуючи конфіденційність та безпеку особистих даних учнів та їхніх сімей.
5. Відгуки користувачів: Відгуки на ClassDojo переважно позитивні, особливо з боку вчителів, які цінують інструменти для мотивації учнів та залучення батьків. Однак, деякі користувачі висловлюють побажання щодо розширення функціоналу з управління класом.

Модель фінансування: ClassDojo пропонує безкоштовний базовий доступ з можливістю оновлення до преміум-версії, яка включає додаткові функції та підтримку. Застосунок фінансується за рахунок преміум-підписок та спонсорських внесків, що дозволяє зберігати основні функції доступними для всіх користувачів.

2.1.4 Застосунок Edmodo

Edmodo був створений у 2008 році компанією "Edmodo, Inc." з метою забезпечення вчителів, учнів та батьків потужним інструментом для спілкування та співпраці в освітньому процесі. Застосунок дозволяє створювати віртуальні класні кімнати, де учасники можуть обмінюватися матеріалами, виконувати завдання та вести обговорення, що робить навчання більш інтерактивним та залучаючим [22].

Детальний аналіз за параметрами дослідження мобільних застосунків для електронних щоденників:

1. Інтерфейс користувача: Edmodo має чистий та організований інтерфейс, який спрощує навігацію та доступ до освітніх ресурсів. Інтерфейс розроблений таким чином, щоб забезпечити легке залучення учнів та вчителів до навчального процесу.
2. Функціональність: Застосунок надає широкий спектр освітніх інструментів, включаючи управління курсами, завданнями, оцінюванням та спілкуванням. Edmodo також дозволяє вчителям створювати опитування та вікторини, що сприяє активній участі учнів у навчальному процесі.
3. Інтеграція: Застосунок інтегрується з багатьма платформами та інструментами, такими як Google Drive, Microsoft Office та інші, що дозволяє легко обмінюватися ресурсами та використовувати різні формати контенту в освітньому процесі.
4. Безпека: Edmodo використовує розширені технології захисту даних, забезпечуючи безпечне зберігання та передачу інформації. Застосунок дотримується міжнародних стандартів безпеки, що забезпечує захист персональних даних користувачів.
5. Відгуки користувачів: Користувачі високо оцінюють Edmodo за його здатність забезпечувати ефективне спілкування та співпрацю між учасниками освітнього процесу. Відгуки також підкреслюють

зручність використання інструментів для організації та управління навчанням.

На рисунках 2.11 – 2.14 представлено деякі функціональні можливості застосунку Edmodo.

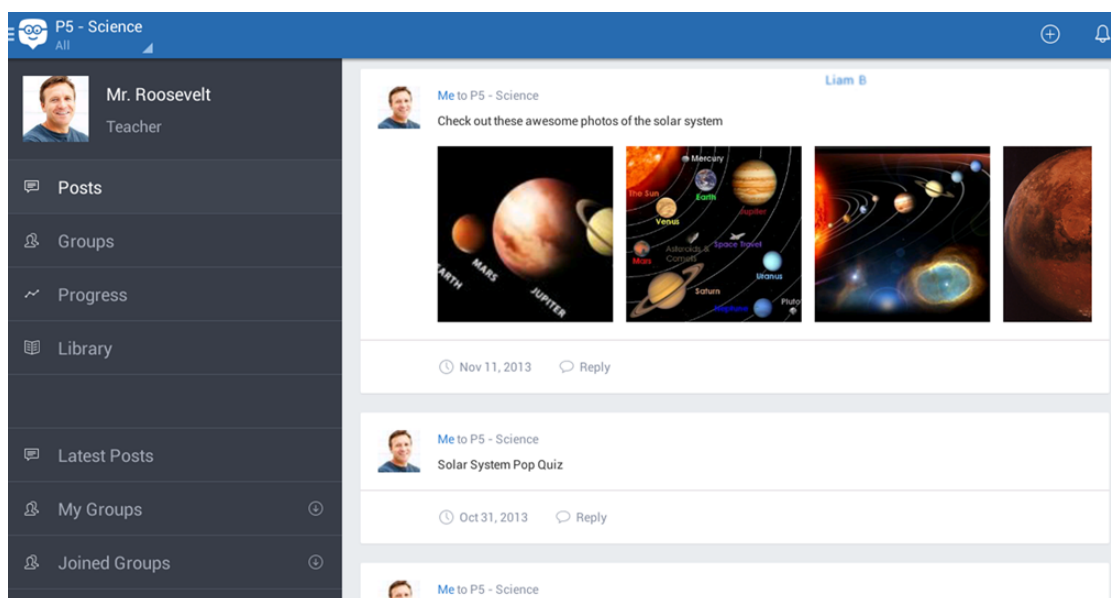


Рисунок 2.11 – Відображення нещодавніх публікацій в застосунку Edmodo

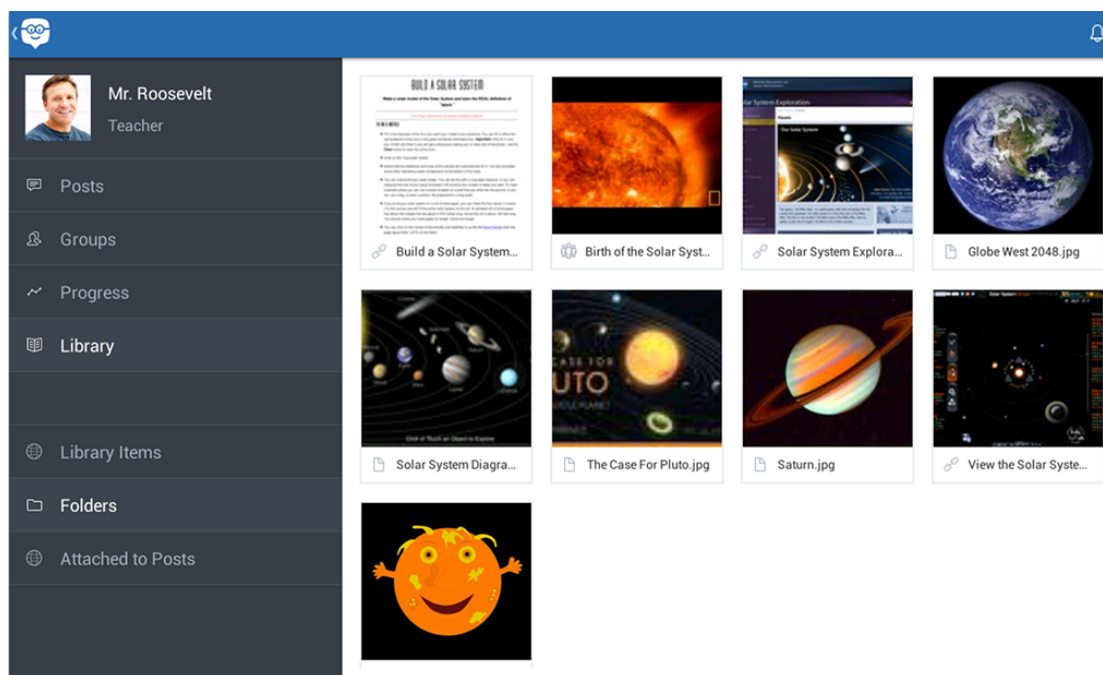


Рисунок 2.12 – Відображення доступних навчальних матеріалів в застосунку Edmodo

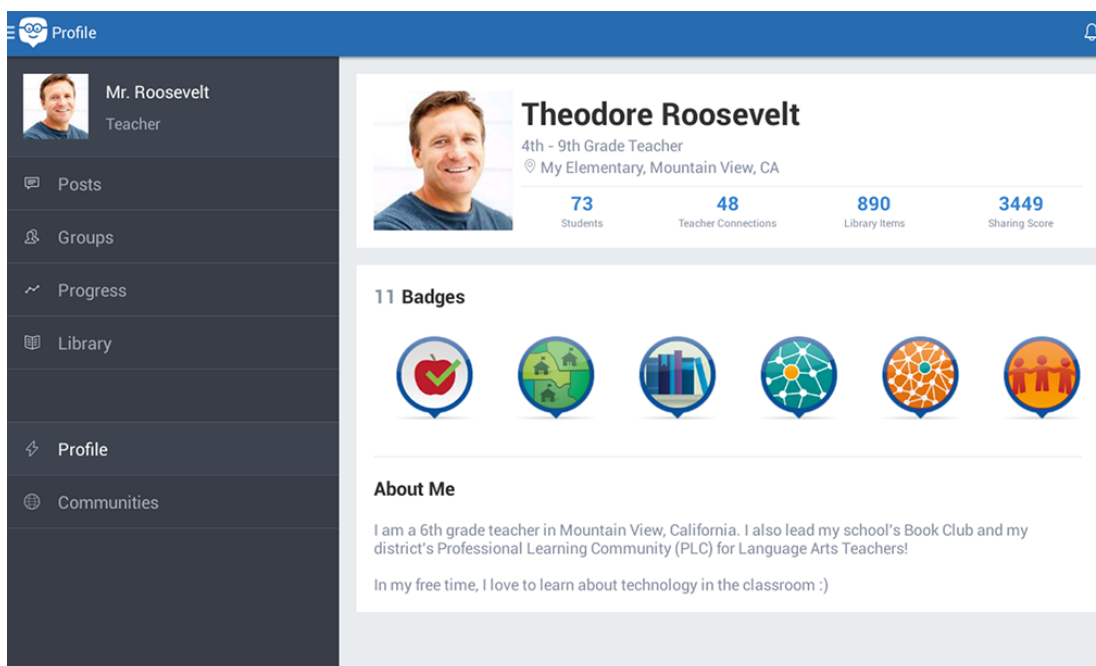


Рисунок 2.13 – Відображення профілю вчителя в застосунку Edmodo

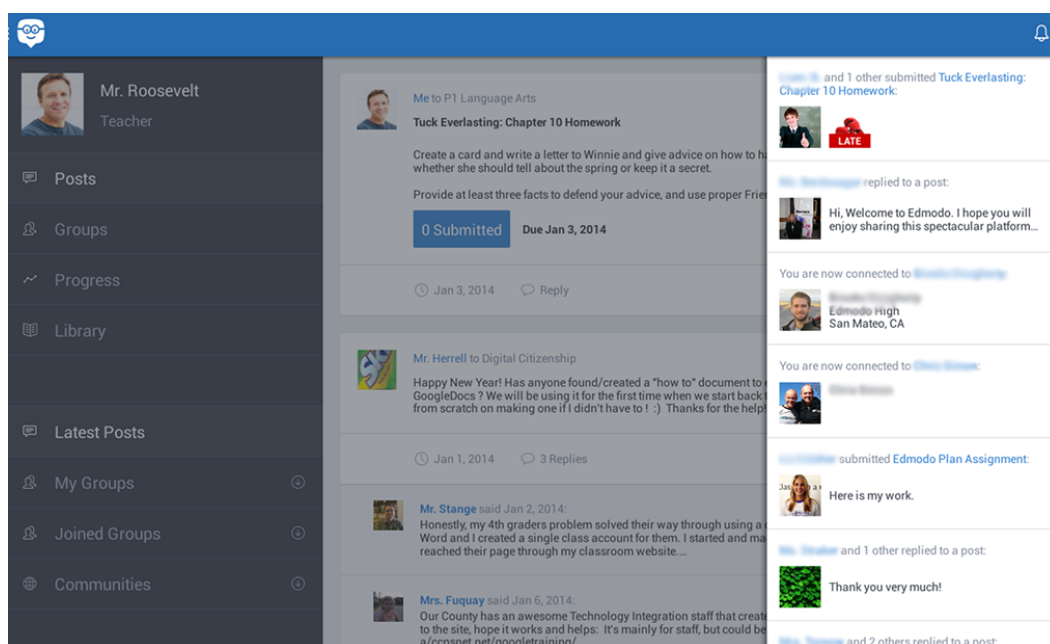


Рисунок 2.14 – Перегляд останніх новин та повідомлень в застосунку Edmodo

Модель фінансування: Edmodo пропонує безкоштовний доступ до основних функцій з можливістю оновлення до платних планів, які включають додаткові інструменти та можливості. Застосунок фінансується за рахунок платних підписок та рекламних доходів, що дозволяє підтримувати його доступність для широкої аудиторії користувачів.

Досліджені чотири застосунки виконують важливу роль у сучасному освітньому процесі, пропонуючи різноманітні інструменти для спілкування, управління та залучення учнів, вчителів та батьків. Кожен з них має свої унікальні особливості:

- School Diary зосереджений на покращенні комунікації між учасниками освітнього процесу та ефективному веденні електронного щоденника.
- MySchoolApp пропонує комплексні інструменти для управління школою та забезпечує високий рівень інтеграції з іншими системами.
- ClassDojo використовує ігрові елементи для мотивації учнів та зміцнення шкільної спільноти.
- Edmodo сприяє інтерактивному навчанню та співпраці, надаючи вчителям інструменти для створення віртуальних класних кімнат.

Кожен застосунок має свої сильні сторони в залежності від потреб користувачів, але всі вони спрямовані на підвищення ефективності та доступності освіти.

В таблиці 2.1 представлено зведену інформацію для порівняння застосунків.

Таблиця 2.1 - Зведена інформація для порівняння досліджених застосунків

Параметр / застосунок		School diary	Myschoolapp	Classdojo	Edmodo
Інтерфейс користувача	Простий і зручний	Сучасний і чистий	Яскравий і дружній	Організований	
Функціональність		Висока	Висока	Середня	Висока
Інтеграція	Базова	Розширена	Базова	Розширена	
Безпека		Високий рівень	Високий рівень	Високий рівень	Високий рівень

Відгуки користувачів	Переважно позитивні	Переважно позитивні	Переважно позитивні	Переважно позитивні
----------------------	---------------------	---------------------	---------------------	---------------------

Таблиця 2.1 - Зведена інформація для порівняння досліджених застосунків (продовження)

Параметр / застосунок	SCHOOL DIARY	MySCHOOLApp	CLASSDOJO	EDMODE
Особливі функції	Електронний щоденник	Управління школою	Ігрові елементи	Віртуальні класи
Мобільна оптимізація	Добра	Потребує поліпшення	Відмінна	Добра
Підтримка мов	Багатомовний	Англійська	Багатомовний	Багатомовний
Доступність платформ	iOS, Android	iOS, Android	iOS, Android	iOS, Android
Підтримка користувачів	24/7	Обмежена	24/7	24/7
Спільнота користувачів	Активна	Менш активна	Дуже активна	Активна
Кастомізація	Обмежена	Висока	Обмежена	Висока
Аналітика та звітність	Базова	Розширена	Базова	Розширена
Синхронізація з календарем	Так	Так	Ні	Так
Відео Інтеграція	Ні	Так	Так	Так
Реакція на зворотний зв'язок	Швидка	Середня	Швидка	Швидка

Цінова політика	Платний з підпискою	Гнучка тарифікація	Безкоштовни й, платні додатки	Безкоштовни й, платні плани
--------------------	------------------------	-----------------------	-------------------------------------	-----------------------------------

Досліджені параметри допомагають зрозуміти, як кожен застосунок може бути адаптований до специфічних потреб користувачів, їх здатність до інтеграції з іншими інструментами, а також як вони можуть сприяти більш ефективному управлінню навчальним процесом через різноманітні функції.

2.2 Дослідження можливостей інтеграції мобільних застосунків з іншими системами

Інтеграція мобільних застосунків з іншими системами є ключовим аспектом для підвищення ефективності та зручності використання в освітньому середовищі. В даному підрозділі досліджено, як мобільні застосунки для електронних щоденників можуть інтегруватися з різними освітніми, адміністративними та комунікаційними платформами, щоб створити єдину, ефективну систему управління навчанням [23, 24].

Взаємодія з освітніми платформами:

Багато шкіл та університетів використовують комплексні освітні платформи, такі як Moodle, Blackboard, або Canvas. Інтеграція мобільних застосунків з цими платформами дозволяє учням і вчителям легко доступати до курсів, матеріалів, оцінок та інших ресурсів безпосередньо з мобільних пристроїв. Це також дозволяє автоматизувати обмін даними між системами, зменшуючи кількість ручної роботи та можливість помилок.

Інтеграція з системами управління школою:

Системи управління школою, такі як PowerSchool або Infinite Campus, забезпечують управління студентськими даними, фінансами, розкладами та іншими адміністративними завданнями. Інтеграція мобільних застосунків з цими системами може поліпшити доступ до інформації та сприяти більш ефективному спілкуванню між учнями, вчителями та адміністрацією.

Синхронізація з календарями та планувальниками:

Мобільні застосунки можуть інтегруватися з календарями, такими як Google Calendar або Microsoft Outlook, щоб допомогти учням та вчителям краще планувати свій час. Це включає синхронізацію розкладів занять, екзаменів, зустрічей та інших подій.

Взаємодія з комунікаційними інструментами:

Інтеграція з комунікаційними інструментами, такими як Slack, Microsoft Teams або Zoom, може поліпшити спілкування та співпрацю між учасниками освітнього процесу. Це дозволяє проводити віртуальні зустрічі, обговорення та спільну роботу над проектами безпосередньо через мобільний застосунок.

Інтеграція мобільних застосунків з іншими системами є важливою частиною створення ефективного та зручного освітнього середовища. Вона дозволяє забезпечити більшу автоматизацію, кращий доступ до ресурсів та підвищену взаємодію між усіма учасниками освітнього процесу.

3 ПРОЕКТУВАННЯ МОДУЛЯ КОРИСТУВАЧА МОБІЛЬНОГО ЗАСТОСУНКУ АВТОМАТИЗОВАНОЇ СИСТЕМИ ЕЛЕКТРОННОГО ЩОДЕННИКА ДЛЯ ЗАГАЛЬНООСВІТНІХ ШКІЛ

3.1 Аналіз функцій модуля користувача для учнів та батьків

Аналіз існуючих систем електронних щоденників в школах показав, що більшість з них зосереджені на задоволенні потреб адміністрації шкіл та вчителів, пропонуючи функціонал для ведення журналів, планування уроків та контролю за відвідуваністю. Однак, потреби батьків та учнів часто залишаються недостатньо врахованими. Батьки та учні потребують більш гнучких і інтуїтивно зрозумілих інструментів для взаємодії з навчальним процесом, доступу до актуальної інформації про успіхи та прогрес у навчанні.

На основі проведеного аналізу, основною задачею нової системи електронного щоденника буде створення модуля, який зосереджений на потребах батьків та учнів. Цей модуль повинен включати:

1. Інтерактивний доступ до навчальних матеріалів: Учні повинні мати можливість легко отримувати доступ до домашніх завдань, електронних підручників та інших навчальних ресурсів.
2. Систему сповіщень та нагадувань: Батьки повинні отримувати сповіщення про важливі шкільні події, зміни в розкладі, а також про успіхи та проблеми своїх дітей.
3. Модуль зворотного зв'язку: Можливість для батьків і учнів залишати зворотний зв'язок вчителям та адміністрації школи, що допоможе покращити взаємодію та вирішення можливих проблем.
4. Аналітичні інструменти: Інструменти для аналізу успішності учня, що дозволять батькам відстежувати прогрес та ідентифікувати області для додаткової уваги та підтримки.
5. Мобільний доступ: Підтримка мобільних пристроїв для забезпечення доступу до системи в будь-який час і з будь-якого місця.

Завдяки розробці цього модуля, система електронного щоденника стане більш орієнтованою на користувачів, зокрема на батьків та учнів, що дозволить їм більш активно та ефективно брати участь у навчальному процесі. У наступному розділі будуть розглянуті технічні аспекти реалізації цього модуля, включаючи вибір технологій, архітектуру системи та план розробки.

3.2 Розробка структури бази даних автоматизованої системи електронного щоденника для загальноосвітніх шкіл

Розробка структури бази даних для модуля користувача в системі електронного щоденника загальноосвітніх шкіл є ключовим елементом, що забезпечує надійне зберігання та ефективну обробку даних користувачів, включаючи учнів, вчителів та батьків. Структура бази даних має бути спроектована таким чином, щоб оптимально підтримувати всі необхідні процеси модуля користувача, включаючи аутентифікацію, авторизацію та управління профілями [25-27].

На рисунку 3.1 представлено ER (Entity-Relationship) діаграму, яка ілюструє структуру бази даних автоматизованої системи електронного щоденника для загальноосвітніх шкіл. Діаграма демонструє основні та додаткові сутності та їх зв'язки, які є необхідними для забезпечення функціональності системи.

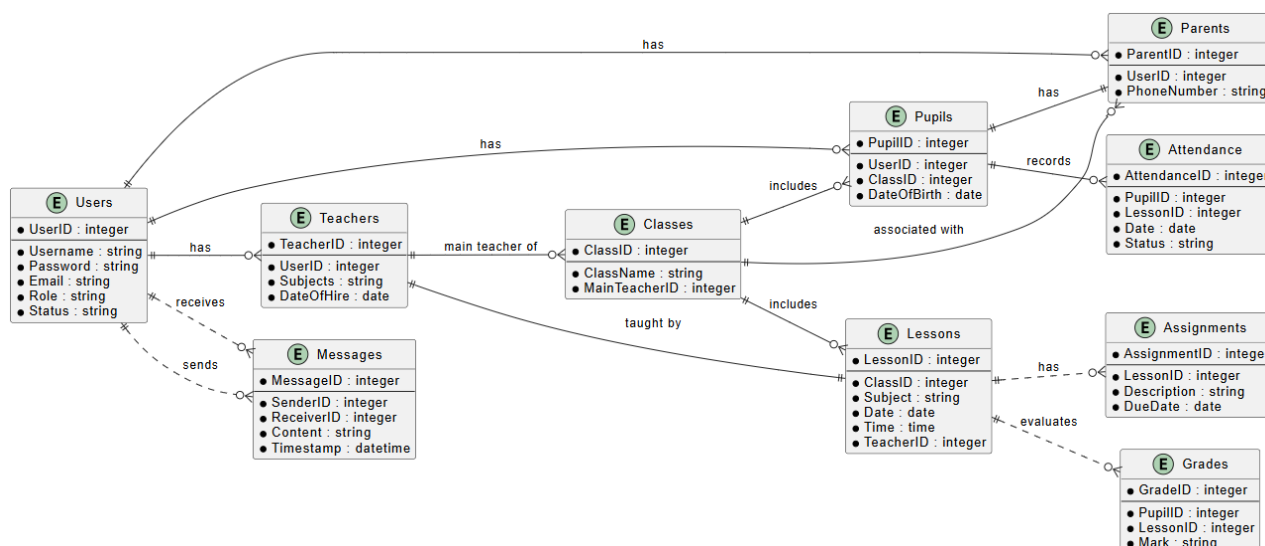


Рисунок 3.1 – ER-діаграма структури бази даних автоматизованої системи електронного щоденника для загальноосвітніх шкіл

ER діаграма включає наступні основні сутності:

- Users: Центральна сутність, яка зберігає загальну інформацію про всіх користувачів системи. Кожен користувач має унікальний ідентифікатор (UserID), ім'я користувача (Username), зашифрований пароль (Password), електронну адресу (Email) та роль (Role), що вказує на його тип (учень, вчитель, батько).
- Pupils: Сутність, яка зберігає специфічну інформацію про учнів, таку як клас (ClassID) і дату народження (DateOfBirth). Кожен учень асоційований з одним користувачем через зв'язок з таблицею Users.
- Teachers: Сутність, що містить деталі про вчителів, включаючи унікальний ідентифікатор вчителя (TeacherID) та перелік предметів, які він викладає (Subjects). Вчитель також пов'язаний з таблицею Users.
- Parents: Зберігає інформацію про батьків і включає зв'язки з учнями, яких вони опікують. Кожен батько має зв'язок з одним або декількома учнями через таблицю Pupils.
- Classes: Сутність, яка зберігає інформацію про класи, включаючи назву класу (ClassName) та ідентифікатор головного вчителя (MainTeacherID). Класи асоційовані з учнями та батьками, що дозволяє легко ідентифікувати зв'язки між учнями та їх батьками.

ER діаграма також включає декілька додаткових сутностей, які підтримують додаткові функціональні аспекти системи електронного щоденника:

- Lessons: Сутність, що представляє конкретні уроки, які проводяться в школі. Кожен урок має унікальний ідентифікатор (LessonID), а також асоційований з певним класом (ClassID), предметом (Subject), датою (Date) та часом (Time). Уроки пов'язані з вчителями, які їх проводять, що дозволяє відстежувати, хто веде кожен конкретний урок.

- Grades: Сутність, яка зберігає оцінки, отримані учнями за уроки. Кожна оцінка має унікальний ідентифікатор (GradeID) та асоційована з учнем (PupilID) та уроком (LessonID), що дозволяє зберігати інформацію про те, які оцінки учень отримав за певний урок.

- Assignments: Сутність, що представляє домашні завдання, які вчителі видають учням. Кожне завдання має унікальний ідентифікатор (AssignmentID), опис (Description), дату виконання (DueDate) та асоційоване з конкретним уроком (LessonID). Це дозволяє відстежувати, які завдання були видали учням та коли вони мають бути виконані.

- Attendance: Сутність, яка відслідковує відвідуваність учнів. Кожен запис про відвідуваність має унікальний ідентифікатор (AttendanceID), асоційований з учнем (PupilID) та уроком (LessonID), а також дату (Date) та статус відвідування (Status). Це дозволяє школам відстежувати, хто відвідував уроки та коли.

- Messages: Сутність, що зберігає повідомлення, які користувачі системи можуть відправляти один одному. Кожне повідомлення має унікальний ідентифікатор (MessageID), відправника (SenderID), одержувача (ReceiverID), текст повідомлення (Content) та часову мітку (Timestamp). Це дозволяє користувачам системи ефективно спілкуватися між собою.

Взаємозв'язки між сутностями в системі електронного щоденника забезпечують комплексне управління навчальним процесом та взаємодією користувачів. Кожен урок у системі тісно пов'язаний з вчителем, який його проводить, що дозволяє чітко відстежувати відповідальність за навчальний процес. Також уроки асоційовані з оцінками, які учні отримують, дозволяючи

зберігати детальну інформацію про їхні академічні досягнення. Домашні завдання, що видаються в рамках певних уроків, також відслідковуються через систему, забезпечуючи організацію та своєчасне виконання завдань учнями. Відвідуваність уроків реєструється для кожного учня, що допомагає школам моніторити присутність та відсутність учнів. Окрім того, система підтримує взаємодію між користувачами через повідомлення, що дозволяє вчителям, учням та батькам ефективно спілкуватися в межах платформи.

Ці взаємозв'язки не тільки сприяють ефективному управлінню навчальними процесами, але й забезпечують інтеграцію та цілісність даних у системі, створюючи міцну основу для управління користувачами в системі електронного щоденника. Ця структура бази даних є критично важливою для реалізації ключових процесів управління користувачами та підтримки їхніх потреб у навчальному середовищі.

3.3 Розробка UML-діаграм для автоматизованої системи електронного щоденника для загальноосвітніх шкіл

У цьому підрозділі необхідно розробити UML-діаграми для автоматизованої системи електронного щоденника для загальноосвітніх шкіл. Ці діаграми допоможуть візуалізувати та структурувати взаємодії та функціональність, які мають бути реалізовані в системі, а саме в модулі користувача. Будуть розглянуті основні UML діаграми, які описують взаємодії між користувачами (батьками та учнями) та системою, а також забезпечуючи зрозумілість процесів та вимог до системи [28].

3.3.1 Опис Use Case UML-діаграм

Use Case UML-діаграма є важливим інструментом у процесі проектування програмного забезпечення, оскільки вона допомагає визначити функціональні вимоги системи та візуалізувати взаємодію між користувачами та системою. У контексті модуля користувача для батьків та дітей, ця діаграма дозволяє чітко

представити, які дії можуть виконувати користувачі в системі електронного щоденника, та як ці дії взаємопов'язані [29].

Use Case діаграма допомагає розробникам та зацікавленим сторонам зрозуміти, як батьки та учні будуть взаємодіяти з модулем користувача. Це сприяє кращому проектуванню інтерфейсу користувача та функціональності, забезпечуючи, що система буде інтуїтивно зрозумілою та ефективною для кінцевих користувачів.

Актори:

1. Учень - основний користувач, який взаємодіє з більшістю функцій системи.
2. Батьки - користувачі, які мають доступ до інформації про успіхи та відвідуваність своїх дітей.
3. Вчитель - вводить дані в систему, такі як оцінки, відвідуваність, домашні завдання.

Основні варіанти використання:

1. Перегляд оцінок:
 - a. Вчитель виставляє оцінки.
 - b. Учень та Батьки переглядають оцінки.
2. Календар:
 - a. Вчитель формує графік занять.
 - b. Учень та Батьки переглядають календар.
3. Сповіщення:
 - a. Система автоматично надсилає сповіщення Учням та Батькам при змінах у розкладі, оцінках або важливих шкільних подіях.
4. Повідомлення:
 - a. Всі актори (Учні, Батьки, Вчителі) можуть надсилати та переглядати повідомлення.
5. Перегляд відвідуваності:
 - a. Вчитель відмічає відвідуваність.
 - b. Учень та Батьки переглядають інформацію про відвідуваність.
6. Перегляд інформації про вчителів:

а. Учень та Батьки можуть переглядати профілі вчителів.

7. Перегляд домашніх завдань:

а. Учень може переглядати та відмічати виконання домашніх завдань.

б. Батьки можуть переглядати домашні завдання, щоб допомогти учням у плануванні.

8. Перегляд інформації про тести:

а. Учень та Батьки переглядають розклад та результати тестів.

9. Інформація про збори:

а. Учень та Батьки отримують інформацію про шкільні збори.

Перш за все, необхідно зобразити Use Case UML-діаграму модуля користувача в ролі Вчитель, щоб розуміти його безпосередній вплив на інформацію, що доступна користувачам з ролями Батьки та Учні. Use Case UML-діаграма модуля користувача в ролі Вчитель зображена на рисунку 3.2.

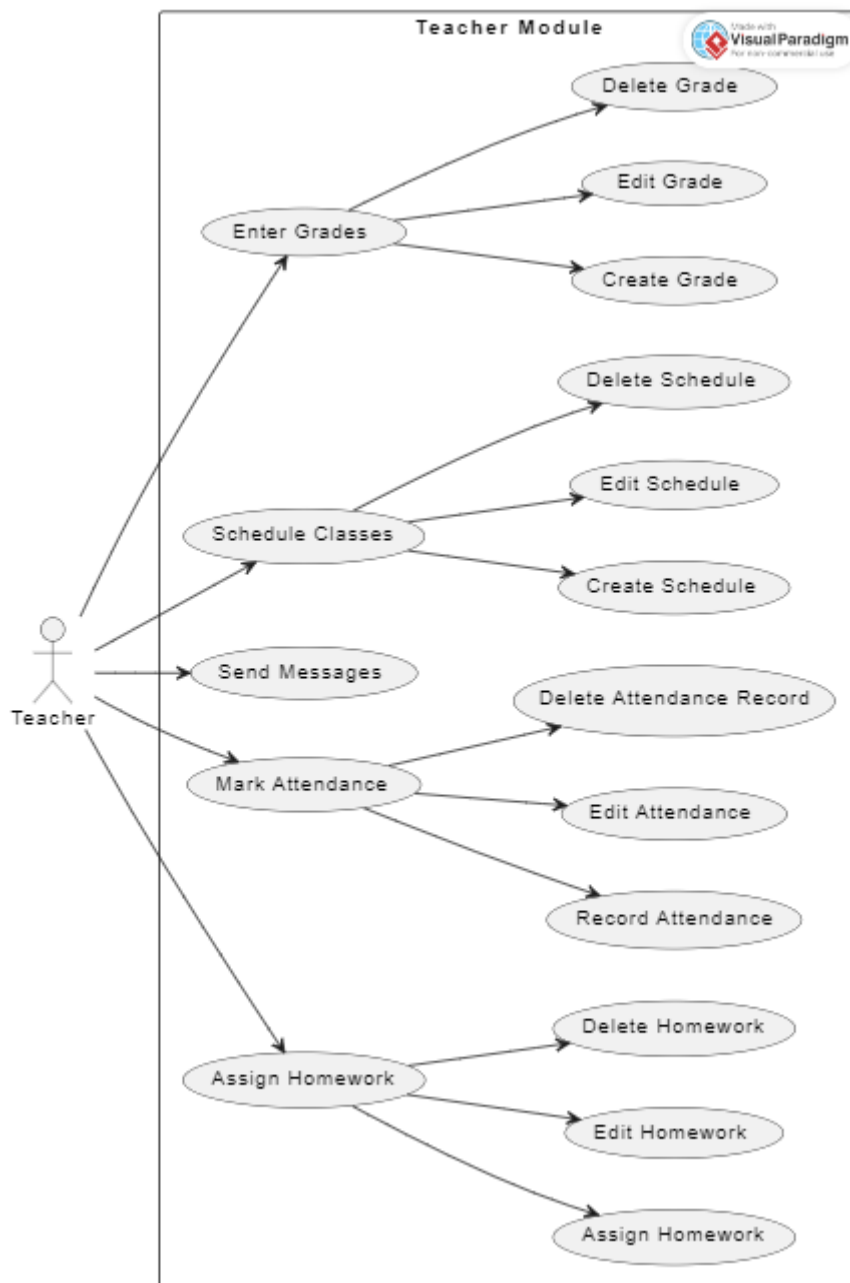


Рисунок 3.2 – Use Case UML-діаграма модуля користувача в ролі Вчитель

Оцінивши вплив користувача в ролі Вчитель на інформацію, що доступна користувачам з ролями Батьки та Учні, Use Case UML-діаграму модуля користувача в ролі Батьки та Учень можна зобразити на рисунку 3.3.

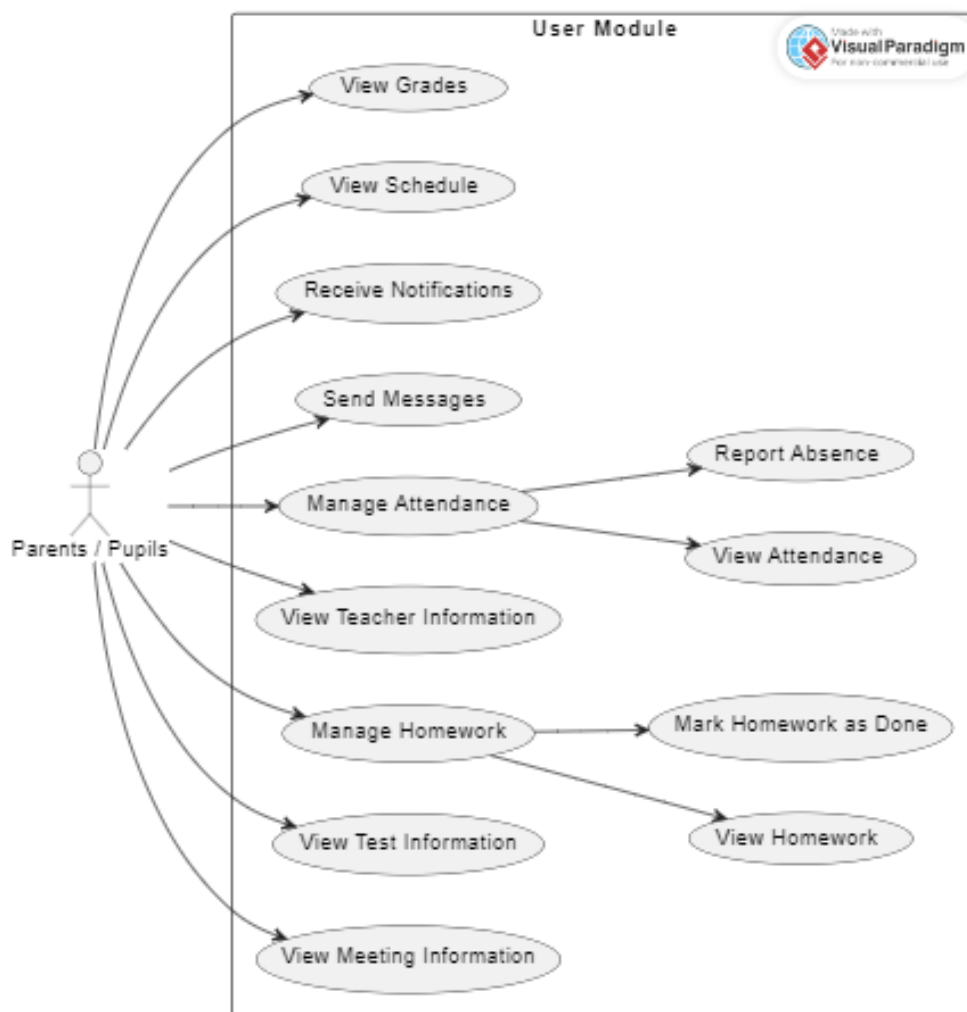


Рисунок 3.3 – Use Case UML-діаграма модуля користувача в ролі Батьки та Учень

Розроблені Use Case UML-діаграми допомагають визначити ключові функції та взаємодії, необхідні для розробки модуля користувача, забезпечуючи, що всі потреби батьків та учнів будуть враховані в кінцевому продукті.

3.3.2 Опис UML-діаграми класів

UML Class Diagram для модуля користувача автоматизованої системи електронного щоденника для загальноосвітніх шкіл відображає структуру та взаємодії між основними класами, що відповідають за роботу з користувачами системи. Діаграма, що зображена на рисунку 3.4, включає класи, які представляють різні ролі користувачів (учні, вчителі, батьки) та їхні

взаємозв'язки з іншими компонентами системи, такими як завдання, розклади та оцінки.

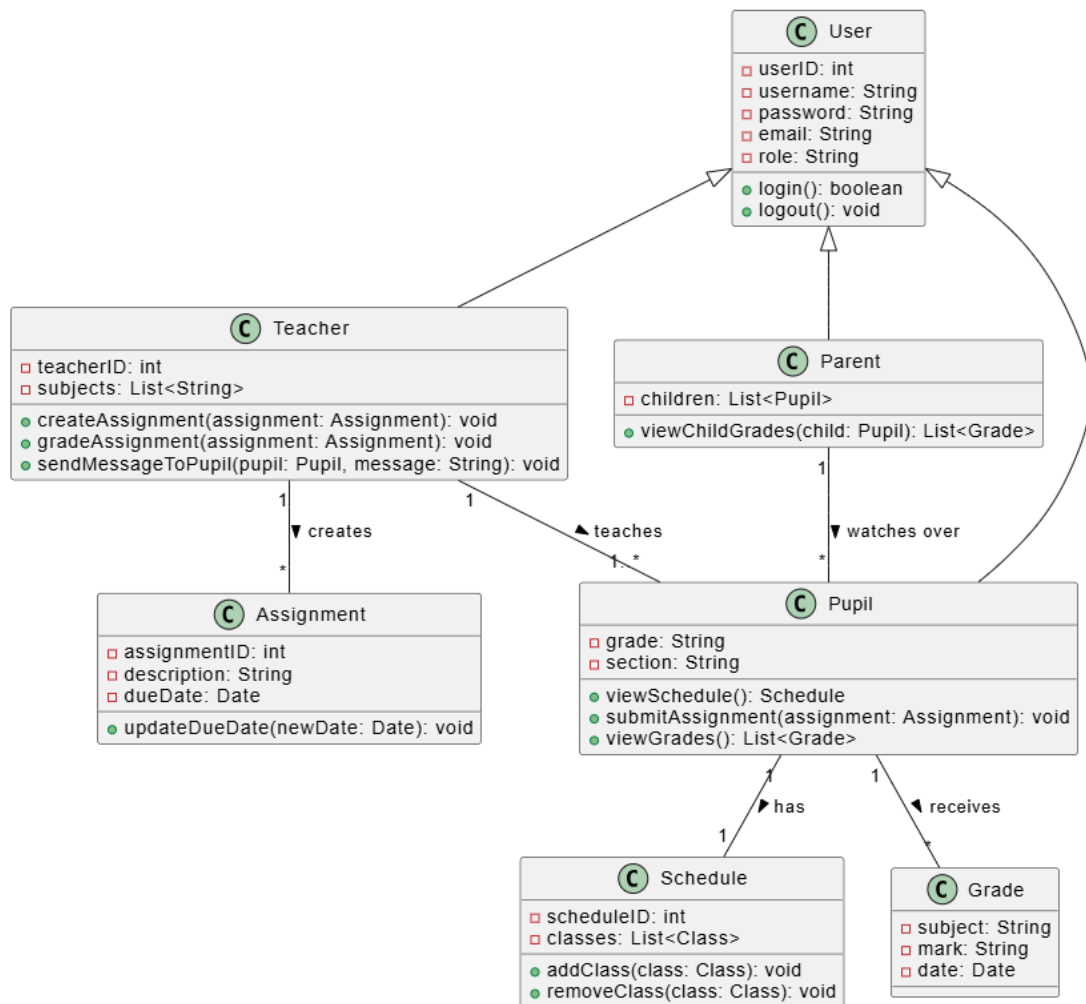


Рисунок 3.4 – UML-діаграма класів автоматизованої системи електронного щоденника для загальноосвітніх шкіл

В основі діаграми лежить клас 'User', який є базовим для всіх користувачів системи і містить загальні атрибути для аутентифікації, такі як ідентифікатор користувача, ім'я користувача, пароль, електронна адреса та роль. Від цього базового класу наслідуються класи 'Pupil', 'Teacher' і 'Parent', кожен з яких представляє специфічні ролі та має додаткові атрибути та методи, що відповідають їхнім функціональним обов'язкам.

Клас 'Pupil' розширює 'User' і додає специфічні атрибути та методи, які дозволяють учням переглядати свій розклад, відправляти завдання та

переглядати оцінки. Вчителі, представлені класом 'Teacher', також наслідують від 'User' і мають можливість створювати завдання, оцінювати їх та взаємодіяти з учнями через повідомлення. Клас 'Parent' дозволяє батькам переглядати оцінки своїх дітей, підкреслюючи їх роль у навчальному процесі.

Зв'язки між класами також відіграють ключову роль у діаграмі. Учні асоційовані з класами 'Schedule' та 'Grade', що дозволяє їм отримувати інформацію про свої заняття та оцінки. Вчителі мають зв'язки з 'Assignment', що відображає їхню здатність створювати та оцінювати завдання. Батьки мають зв'язок з 'Pupil', що дозволяє їм слідкувати за академічними досягненнями своїх дітей.

Ця діаграма не тільки відображає структуру даних, але й демонструє, як користувачі взаємодіють з системою та один з одним, що є критично важливим для розуміння та розробки ефективної системи електронного щоденника.

3.3.3 Опис UML-діаграми компонентів

Діаграма компонентів для системи електронного щоденника загальноосвітніх шкіл, що зображена на рисунку 3.5, відображає основні структурні блоки системи та їх взаємодії. Вона розділена на кілька ключових модулів, кожен з яких відповідає за певну область функціональності [30].

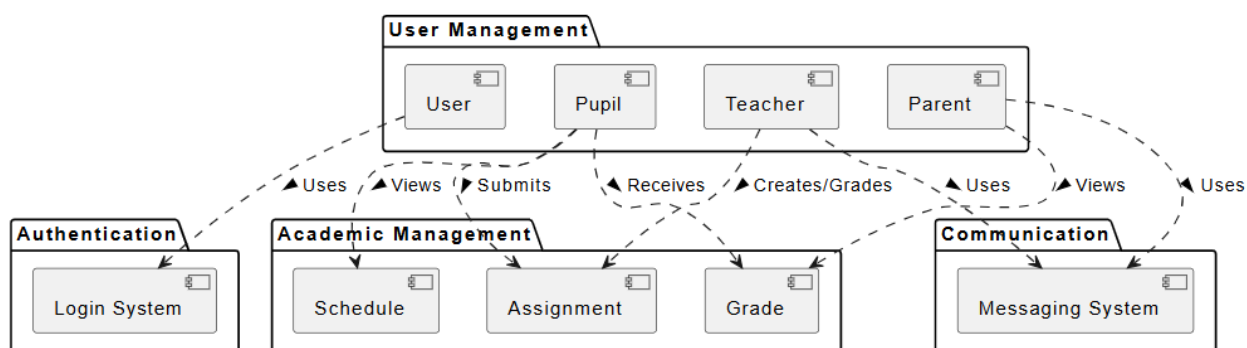


Рисунок 3.5 – UML-діаграма компонентів автоматизованої системи електронного щоденника для загальноосвітніх шкіл

Модуль "User Management" зосереджений на управлінні користувачами системи. Він включає компоненти для різних типів користувачів: учнів, вчителів та батьків. Цей модуль взаємодіє з "Authentication", який забезпечує процеси входу та аутентифікації користувачів, гарантуючи безпеку та контроль доступу до системи.

"Academic Management" є ключовим модулем, який управляє академічною інформацією, такою як розклади, завдання та оцінки. Цей модуль дозволяє вчителям створювати та оцінювати завдання, а учням – переглядати свої розклади та оцінки. Взаємодія між учнями та вчителями є центральною для цього модуля, оскільки вона сприяє активному навчальному процесу.

Модуль "Communication" включає систему обміну повідомленнями, яка дозволяє користувачам взаємодіяти один з одним в межах системи. Це особливо важливо для вчителів та батьків, які можуть використовувати цей канал для обговорення академічного прогресу учнів.

Кожен з цих модулів має чітко визначені зв'язки з іншими модулями, що забезпечує інтегровану та ефективну роботу системи. Взаємодії між модулями дозволяють системі функціонувати як єдине ціле, забезпечуючи користувачам доступ до необхідних інструментів та інформації. Ця структура не тільки сприяє зручності користувачів, але й підтримує масштабування та адаптацію системи до змінюваних вимог освітнього процесу.

Завершення проектування UML діаграм та структури бази даних створює фундамент для розробки модуля користувача. Ці інструменти визначають архітектурні засади та взаємодії між компонентами системи, що є ключовим для забезпечення її функціональності та масштабованості. На основі цього фундаменту, наступним кроком є розробка модуля користувача для мобільного застосунку автоматизованої системи електронного щоденника для загальноосвітніх шкіл. Цей етап включає вибір технологій, розробку програмних компонентів та налаштування з подальшим тестуванням модуля, що сприятиме оптимізації процесів та підвищенню ефективності системи.

4 РОЗРОБКА МОДУЛЯ КОРИСТУВАЧА МОБІЛЬНОГО ЗАСТОСУНКУ АВТОМАТИЗОВАНОЇ СИСТЕМИ ЕЛЕКТРОННОГО ЩОДЕННИКА ДЛЯ ЗАГАЛЬНООСВІТНІХ ШКІЛ

4.1 Вибір технологій для розробки модуля користувача

Для розробки модуля користувача мобільного застосунку автоматизованої системи електронного щоденника для загальноосвітніх шкіл, вибір технологій базується на вимогах до програмно-технічного забезпечення, визначених у попередніх розділах, а також на структурних особливостях системи, визначених на етапі проектування. Основні компоненти та технології, які будуть використані, включають [31-33]:

1. Локальна база даних: Використання SQLite версії 3.32.3 як локальної бази даних для мобільного застосунку забезпечить швидкий доступ до даних без необхідності постійного підключення до інтернету. Це підвищить продуктивність застосунку та забезпечить стабільність роботи навіть при обмеженому з'єднанні.
2. Веб-інтерфейс: Для розробки веб-інтерфейсу буде використано React версії 17.0.2 для клієнтської частини та Node.js версії 14.17.0 для серверної частини. Цей вибір забезпечить високу адаптивність та сумісність з усіма сучасними браузерами, а також дозволить ефективно управляти мобільним застосунком через веб.
3. Серверні рішення: Використання хмарної платформи AWS, зокрема сервісів EC2 та S3, дозволить забезпечити високу доступність, масштабованість та безпеку системи. Налаштування балансування навантаження та автоматичне масштабування допоможуть оптимізувати ресурси та забезпечити стабільність роботи системи.
4. Засоби безпеки: Шифрування даних на стороні клієнта та сервера, захист даних за допомогою HTTPS, а також аутентифікація користувачів через

OAuth 2.0 забезпечать високий рівень безпеки. Регулярний аудит безпеки та оновлення заходів захисту будуть включені в план технічного обслуговування.

Ці технології дозволять реалізувати необхідні функції розробки, такі як доступ користувачів до системи та їхніх профілів, відображення розкладу занять, перегляд оцінок, ознайомлення з домашніми завданнями та доступ до навчальних ресурсів. Вибір цих технологій забезпечить відповідність програмної платформи вимогам до мобільних операційних систем, зокрема Android, та умовам експлуатації системи, включаючи цілодобове функціонування та захист від несанкціонованого доступу.

4.2 Розробка програмних компонентів модуля користувача

Розробка програмних компонентів модуля користувача є ключовим етапом у створенні мобільного застосунку автоматизованої системи електронного щоденника для загальноосвітніх шкіл. Цей процес включає декілька важливих аспектів, починаючи від архітектури модуля, розробки інтерфейсу користувача, інтеграції з базою даних, взаємодії з сервером, забезпечення безпеки, тестування, аж до документації та підтримки. Кожен з цих аспектів має вирішальне значення для забезпечення функціональності, продуктивності та безпеки застосунку [34].

Архітектура модуля користувача

Архітектура модуля користувача визначає основну структуру програми, що включає в себе розподіл компонентів, їх взаємодію, а також засоби управління даними та логіку обробки запитів. Для мобільного застосунку електронного щоденника було обрано архітектурний патерн MVVM (Model-View-ViewModel), який добре підходить для розробки на платформах з багатим користувацьким інтерфейсом, таких як Android.

Model в MVVM відповідає за бізнес-логіку та взаємодію з базою даних. Вона забезпечує доступ до даних, їх обробку та відправку в ViewModel. View є

компонентом, який відповідає за відображення інтерфейсу користувача та взаємодію з користувачем. ViewModel служить посередником між Model та View, обробляючи всю логіку інтерфейсу користувача, а також забезпечуючи зв'язок даних між View та Model.

Для реалізації цієї архітектури на платформі Android, було використовувати такі інструменти та бібліотеки, як Data Binding для зв'язування даних у XML-розмітці з класами Java або Kotlin, LiveData для реактивного оновлення UI відповідно до змін у даних, та Room для абстракції та оптимізації доступу до SQLite бази даних [35-37].

Наприклад, використання ViewModel у Android застосунку:

```
class ScheduleViewModel(application: Application) :
    AndroidViewModel(application) {
        private val repository: ScheduleRepository =
            ScheduleRepository(application)

        val schedule: LiveData<List<ClassSchedule>> = repository.getSchedule()

        fun refreshSchedule() {
            viewModelScope.launch {
                repository.refreshScheduleFromNetwork()
            }
        }
    }
}
```

‘ScheduleViewModel’ використовує ‘ScheduleRepository’ для отримання розкладу занять, який відображається у користувацькому інтерфейсі. ‘LiveData’ використовується для спостереження за даними розкладу, що дозволяє UI автоматично оновлюватися при зміні даних.

Ця архітектура дозволяє чітко розділити відповідальності між компонентами системи, спрощує тестування та підтримку коду, а також покращує загальну продуктивність застосунку.

Розробка інтерфейсу користувача

Використання React для розробки веб-інтерфейсу дозволяє створювати компоненти, які можна легко перевикористовувати та тестувати. React використовує декларативний підхід, що робить код більш прозорим та легким для розуміння. Для мобільної розробки на платформі Android, можна використовувати Kotlin разом з Android Jetpack, який надає широкий спектр компонентів та утиліт для спрощення розробки інтерфейсів [38-40].

Компонента React, який відображає список занять:

```
import React from 'react';

class ScheduleList extends React.Component {
  render() {
    return (
      <div>
        {this.props.schedule.map((item, index) => (
          <div key={index}>
            <h3>{item.subject}</h3>
            <p>{item.time}</p>
          </div>
        ))}
      </div>
    );
  }
}

export default ScheduleList;
```

`ScheduleList` є React компонентом, який приймає `schedule` як пропс і відображає список занять. Кожен елемент списку містить назву предмету та час проведення заняття.

Для мобільного застосунку на Android, використання таких елементів як `RecyclerView` дозволяє ефективно управляти списками даних, що динамічно змінюються. `RecyclerView` оптимізує відображення великих списків, забезпечуючи високу продуктивність та плавність прокрутки.

```
class ScheduleAdapter(private val scheduleList: List<ScheduleItem>) :
    RecyclerView.Adapter<ScheduleAdapter.ViewHolder>() {
```

```
    class ViewHolder(view: View) : RecyclerView.ViewHolder(view) {
        val subjectTextView: TextView =
            view.findViewById(R.id.subjectTextView)
        val timeTextView: TextView = view.findViewById(R.id.timeTextView)
    }
```

```
    override fun onCreateViewHolder(parent: ViewGroup, viewType: Int):
        ViewHolder {
        val view =
            LayoutInflater.from(parent.context).inflate(R.layout.schedule_item, parent, false)
        return ViewHolder(view)
    }
```

```
    override fun onBindViewHolder(holder: ViewHolder, position: Int) {
        val scheduleItem = scheduleList[position]
        holder.subjectTextView.text = scheduleItem.subject
        holder.timeTextView.text = scheduleItem.time
    }
```

```
    override fun getItemCount() = scheduleList.size
```

```
}
```

`ScheduleAdapter` є адаптером для `RecyclerView`, який управляє відображенням даних про розклад занять. Кожен елемент списку містить назву предмету та час.

Інтеграція з локальною базою даних

Для інтеграції SQLite у мобільний застосунок на Android було використано бібліотеку Room, яка є частиною Android Jetpack. Room надає абстрактний шар над SQLite, що спрощує роботу з базою даних та зменшує кількість шаблонного коду [41-43].

Таким чином, наприклад, відбувається оголошення сутності та DAO (Data Access Object) для роботи з базою даних:

```
@Entity(tableName = "schedule")
data class ScheduleEntity(
    @PrimaryKey(autoGenerate = true) val id: Int,
    @ColumnInfo(name = "subject") val subject: String,
    @ColumnInfo(name = "time") val time: String
)
```

```
@Dao
interface ScheduleDao {
    @Query("SELECT * FROM schedule")
    fun getAll(): LiveData<List<ScheduleEntity>>
```

```
@Insert
fun insert(schedule: ScheduleEntity)
```

```
@Delete
fun delete(schedule: ScheduleEntity)
```

```

    }

    @Database(entities = [ScheduleEntity::class], version = 1)
    abstract class AppDatabase : RoomDatabase() {
        abstract fun scheduleDao(): ScheduleDao
    }

```

‘ScheduleEntity’ визначає структуру таблиці для зберігання розкладу занять. ‘ScheduleDao’ надає методи для доступу до даних, включаючи отримання всіх записів, вставку нового запису та видалення існуючого. ‘AppDatabase’ є абстрактним класом, який наслідує ‘RoomDatabase’ та надає доступ до DAO.

Для безпосередньої ініціалізації бази даних у застосунку використано наступний код:

```

val db = Room.databaseBuilder(
    applicationContext,
    AppDatabase::class.java, "school_diary.db"
).build()

```

Цей код створює базу даних з назвою "school_diary.db" та ініціалізує її використовуючи контекст застосунку. Після цього, використовується ‘db.scheduleDao()’ для виконання операцій з даними в базі.

Для подібних систем інтеграція з локальною базою даних не тільки підвищує продуктивність застосунку за рахунок зменшення залежності від мережових запитів, але й забезпечує кращий досвід користувача, оскільки дозволяє працювати з даними офлайн [44].

Взаємодія з сервером

Для забезпечення синхронізації даних між мобільним застосунком та сервером, важливо було розробити ефективний механізм взаємодії. Це включало в себе створення API, яке дозволить мобільному застосунку відправляти та отримувати дані від сервера. Для розробки серверної частини було використано Node.js з фреймворком Express, який дозволяє легко створювати масштабовані веб-додатки [45, 46].

Розроблений RESTful API, створений за допомогою Express, який обробляє запити на отримання та відправлення даних розкладу занять:

```
const express = require('express');
const app = express();
app.use(express.json());

const schedules = [
  { id: 1, subject: 'Mathematics', time: '09:00' },
  { id: 2, subject: 'Physics', time: '10:00' }
];

app.get('/api/schedules', (req, res) => {
  res.send(schedules);
});

app.post('/api/schedules', (req, res) => {
  const schedule = {
    id: schedules.length + 1,
    subject: req.body.subject,
    time: req.body.time
  };
  schedules.push(schedule);
  res.send(schedule);
});
```

```
});
```

```
const port = process.env.PORT || 3000;
app.listen(port, () => console.log(`Listening on port ${port}...`));
```

В даному випадку сервер використовує два маршрути: один для отримання списку розкладів (`GET /api/schedules`), а інший для додавання нового розкладу (`POST /api/schedules`).

На стороні мобільного застосунку, для взаємодії з сервером було використано бібліотеку Retrofit на Android, яка спрощує процес відправлення HTTP запитів та обробки відповідей. Використання Retrofit для взаємодії з вищезазначеним API:

```
interface ScheduleService {
    @GET("schedules")
    fun listSchedules(): Call<List<Schedule>>

    @POST("schedules")
    fun createSchedule(@Body schedule: Schedule): Call<Schedule>
}
```

```
val retrofit = Retrofit.Builder()
    .baseUrl("http://example.com/api/")
    .addConverterFactory(GsonConverterFactory.create())
    .build()

val service = retrofit.create(ScheduleService::class.java)
```

‘ScheduleService’ описує методи для взаємодії з API, Retrofit використовує анотації для визначення HTTP методів та URL, а також автоматично серіалізує дані у JSON.

Цей підхід дозволяє ефективно синхронізувати дані між мобільним застосунком та сервером, забезпечуючи актуальність інформації для користувачів та підтримуючи консистентність даних у розподіленій системі.

Тестування модуля

Під час розробки програмних компонентів модуля користувача було використано систему контролю версій Git, яка забезпечила постійний контроль за змінами в код. Це також дало змогу вести паралельну розробку різних функцій та забезпечувати їхню інтеграцію без конфліктів [47-49].

Для забезпечення якості коду були налаштовані юніт-тести за допомогою фреймворку JUnit на Android, що дозволило автоматично перевіряти окремі компоненти програми на відповідність вимогам та специфікаціям. Інтеграційні тести були використані для перевірки взаємодії між компонентами, а приймальні тести допомогли переконатися у відповідності загальної функціональності застосунку розробленим вимогам.

4.3 Налаштування та тестування роботи модуля користувача

В даному підрозділі описано процес налаштування та тестування модуля користувача в мобільному додатку електронного щоденника. Модуль користувача дозволяє відображати різні типи інформації, які є критично важливими для учнів та батьків. Налаштування включає конфігурацію відображення оцінок, відвідуваності, домашніх завдань, замін у розкладі та повідомлень.

На рисунку 4.1 представлено головний екран мобільного додатку з відображенням нових оцінок за замовчуванням. На цьому екрані користувачі

можуть переглядати останні оцінки, які були додані до системи. Це допомагає учням та батькам швидко оцінити академічний прогрес.

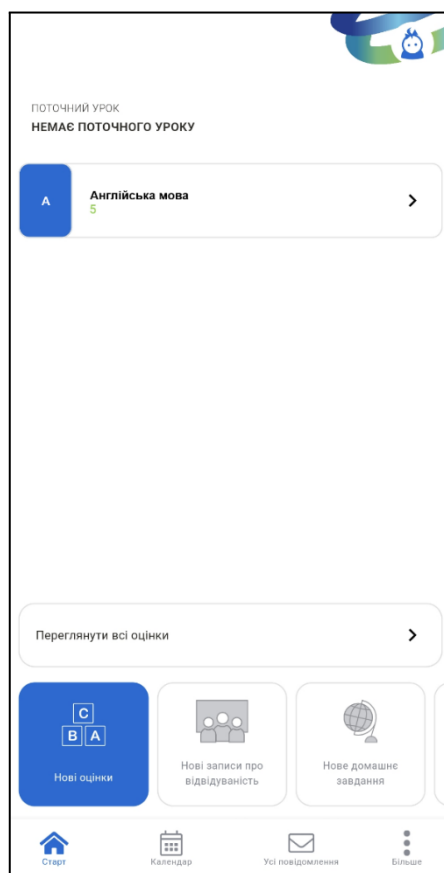
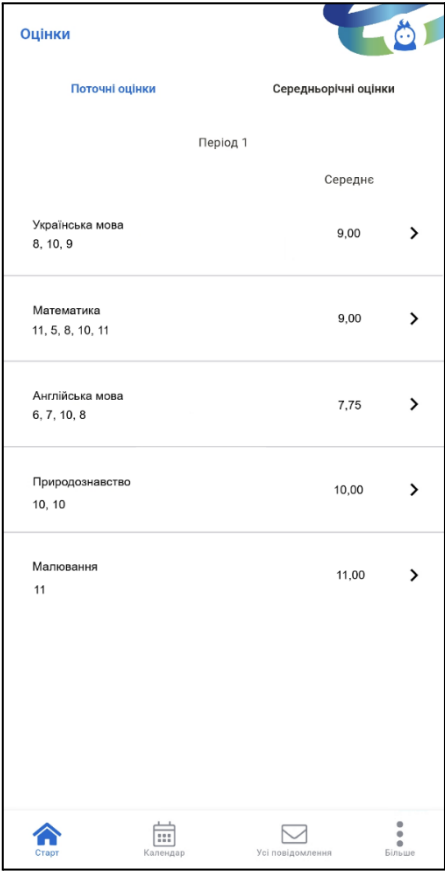


Рисунок 4.1 – Головний екран мобільного додатку з відображенням нових оцінок за замовчуванням

Натиснувши на кнопку «Переглянути всі оцінки» можна перейти до екрану відображення списку поточних оцінок, що представлений на рисунку 4.2. Дана інформація дозволяє користувачам переглядати детальну інформацію про оцінки по кожному предмету, що сприяє кращому розумінню академічних результатів.



Поточні оцінки	Середньорічні оцінки
Період 1	Середнє
Українська мова 8, 10, 9	9,00 >
Математика 11, 5, 8, 10, 11	9,00 >
Англійська мова 6, 7, 10, 8	7,75 >
Природознавство 10, 10	10,00 >
Малювання 11	11,00 >

Рисунок 4.2 – Екран відображення поточних оцінок

Якщо повернутись до головного екрану і натиснути на вкладку «Нові записи про відвідуваність», що зображено на рисунку 4.3, то будуть доступними для перегляду останні дані про відвідуваність учнів, що дозволяє вчителям та батькам відстежувати присутність учнів на заняттях.

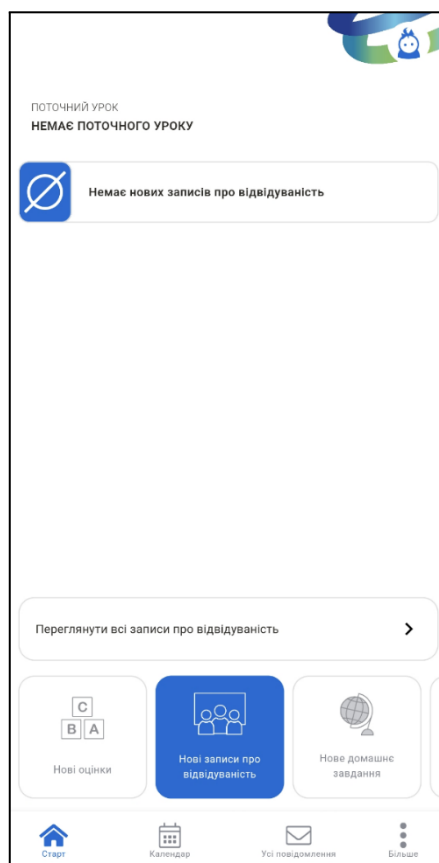


Рисунок 4.3 – Головний екран мобільного додатку з відображенням нових записів про відвідуваність

Натиснувши на кнопку «Переглянути всі записи про відвідуваність» можна перейти до екрану відображення відвідуваності, що представлено на рисунку 4.4. Даний екран детально показує відвідуваність за обраний період. Перейшовши у вкладку «Статистика» чи «Предмети», що зображено на рисунку 4.5, відображається статистична інформація про відвідуваність групуючи дані за типом пропуску/запізнення та за предметами відповідно, що допомагає виявляти тенденції та вживати заходів при необхідності.

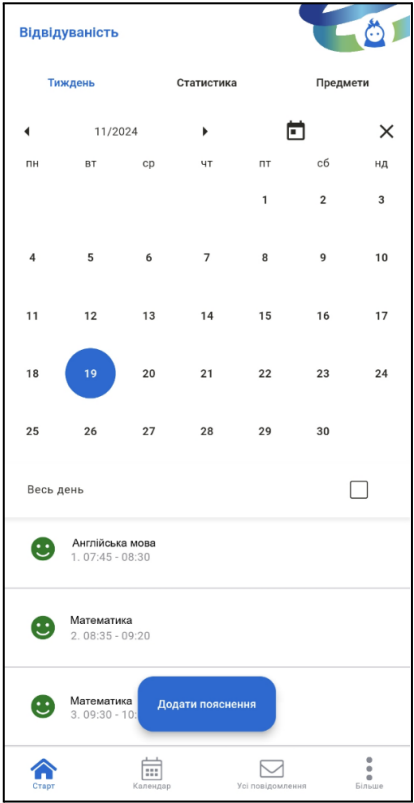


Рисунок 4.4 – Екран відображення відвідуваності

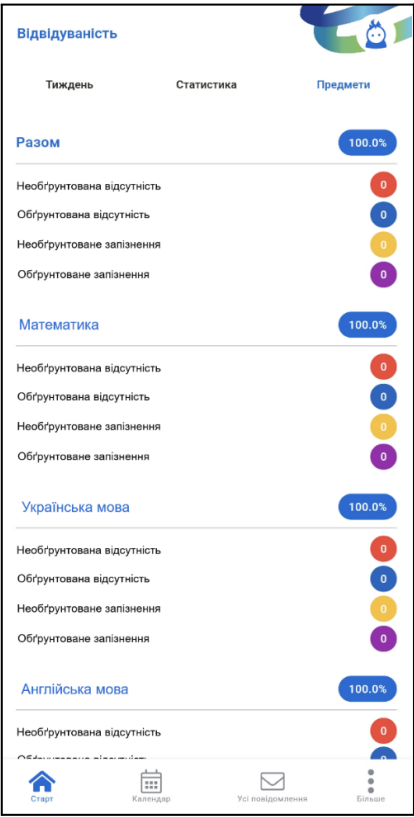


Рисунок 4.5 – Екран відображення статистики відвідуваності з групуванням по предметах

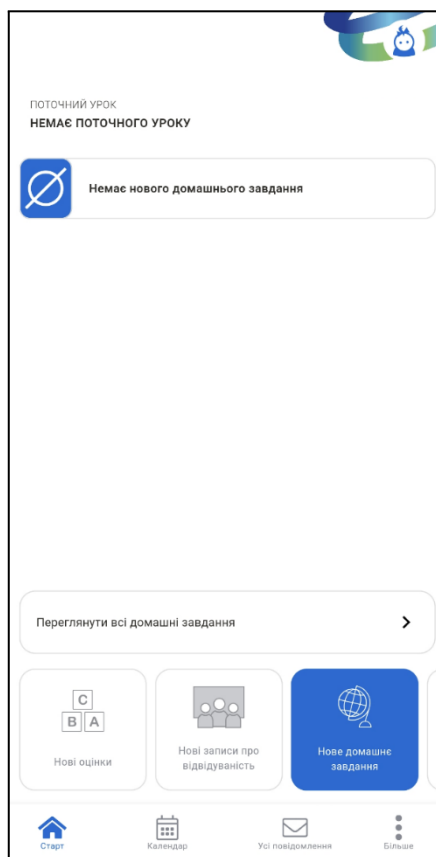


Рисунок 4.6 – Головний екран мобільного додатку з відображенням нових домашніх завдань

На рисунку 4.6 зображено головний екран мобільного додатку з відображенням нових домашніх завдань, на якому користувачі можуть бачити останні домашні завдання, що були виставлені вчителями.

На рисунках 4.7 та 4.8 представлено детальну інформацію про обрані домашні завдання, включаючи вимоги та терміни виконання.

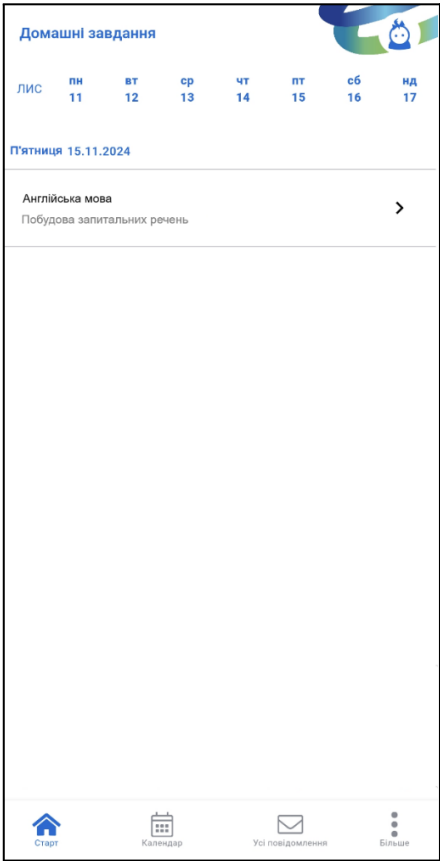


Рисунок 4.7 – Відображення домашніх завдань в обраний день

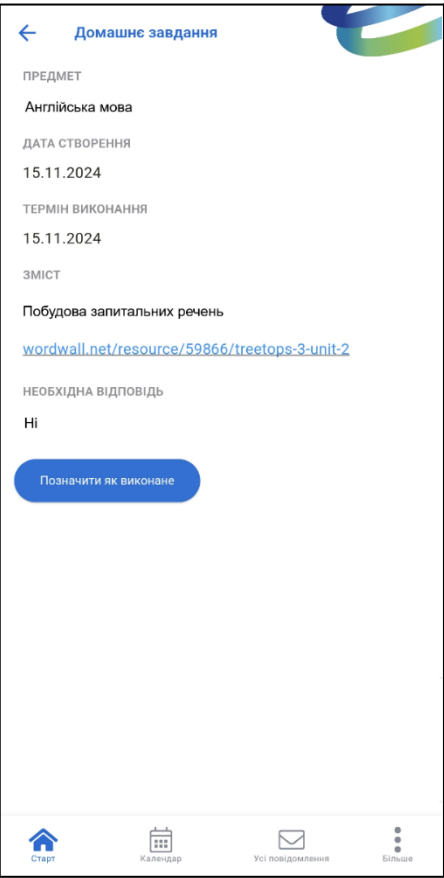


Рисунок 4.8 – Детальна інформація про обране домашнє завдання

На рисунку 4.9 представлено головний екран мобільного додатку з відображенням нових змін в розкладі. Цей екран інформує користувачів про будь-які зміни в розкладі, що дозволяє їм планувати свій день відповідно.

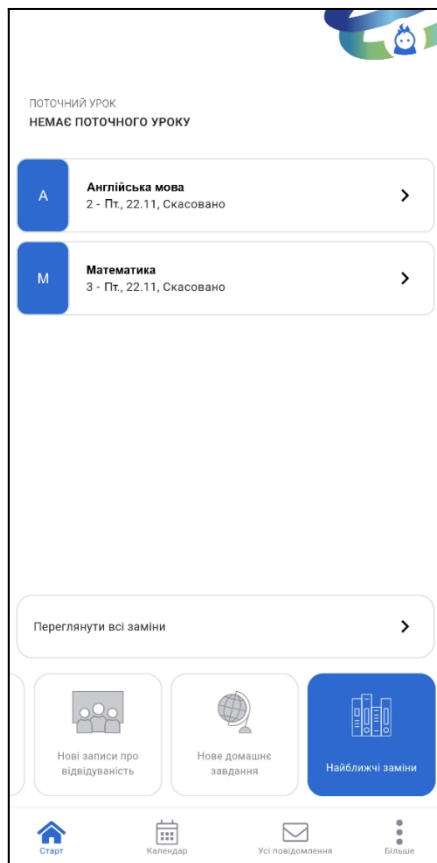


Рисунок 4.9 – Головний екран мобільного додатку з відображенням нових змін в розкладі

Натиснувши на кнопку «Розклад» з'являється екран календаря з розкладом занять, що зображено на рисунку 4.10. На цьому екрані користувачі можуть переглядати розклад занять на день та тиждень, що допомагає організувати їхній навчальний процес. Можна також переглядати попередні та майбутні тижні, в яких будуть зроблені відповідні поправки, якщо в графіку відбулась заміна.

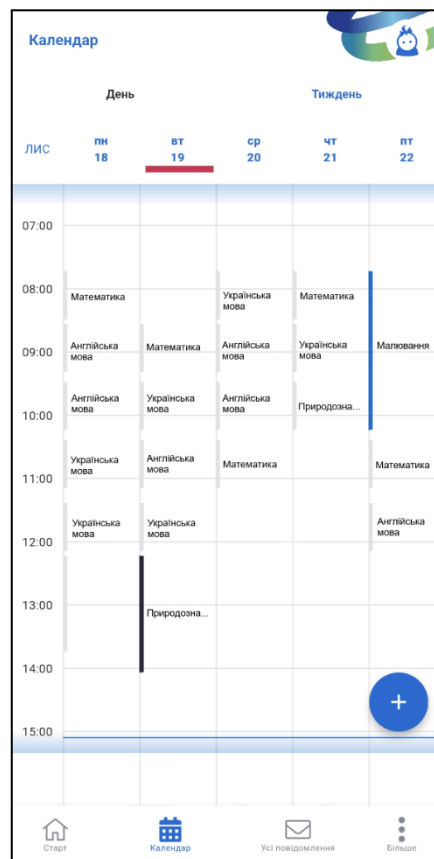


Рисунок 4.10 – Відображення календаря з розкладом занять

На рисунку 4.11 зображено відображення усіх повідомлень користувача. Можна перемикастись між вкладками вхідних, вихідних та видалених повідомлень. На рисунку 4.12 зображено відкрите повідомлення від іншого користувача. На дане повідомлення можна відправити відповідь або ж можна його видалити.

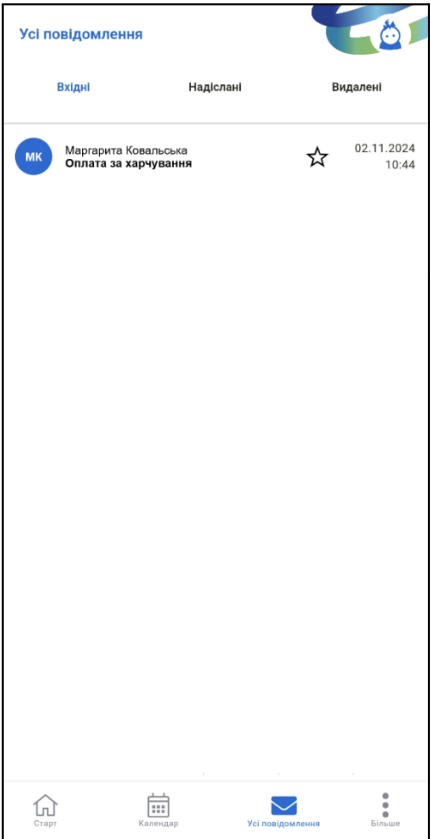


Рисунок 4.11 – Відображення усіх повідомлень

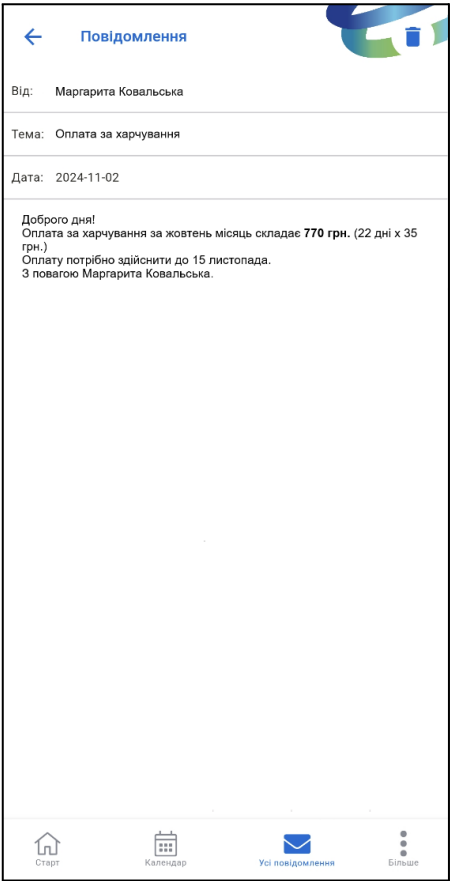


Рисунок 4.12 – Відображення деталей обраного повідомлення

В головному меню також можна переглянути список вчителів, що проводять уроки в обраному класі, інформацію про дітей та перемикатись між їх класами, додавати дітей за попереднім створенням їх профіля адміністратора. Профіль для батьків також створюється за допомогою адміністратора, та активується самостійно після підтвердження електронної пошти.

Завершуючи розгляд модуля користувача, можна підкреслити, що налаштування та тестування мобільного додатку електронного щоденника спрямовані на забезпечення зручності та ефективності його використання всіма учасниками освітнього процесу. Від налагодження роботи з оцінками та відвідуваністю до інтеграції домашніх завдань і розкладу, кожен аспект додатку ретельно перевіряється, щоб відповідати потребам користувачів. Завдяки гнучким налаштуванням профілів, як для учнів, так і для батьків, додаток стає незамінним помічником у плануванні та організації навчального процесу. Таким чином, кожен користувач може ефективно використовувати всі можливості електронного щоденника, що сприяє підвищенню якості освіти та залученню до активної участі в шкільному житті.

ВИСНОВКИ

У ході виконання магістерської кваліфікаційної роботи було проведено глибоке дослідження та аналіз автоматизованої системи електронного щоденника для загальноосвітніх шкіл з метою оптимізації навчального процесу та підвищення ефективності взаємодії між учасниками освітнього процесу. Метою роботи було створення інноваційного модуля користувача для мобільного застосунку, спрямованого на спрощення взаємодії між учнями, вчителями та батьками, а також на оптимізацію управління навчальними матеріалами.

Було проведено дослідження об'єкта автоматизації. Аналізувались сучасні системи управління навчальним процесом у школах. Було проведено детальний аналіз роботи існуючих систем, вивчено можливості різних ролей користувачів у системах електронних щоденників, таких як учні, вчителі та батьки.

Досліджено ринок IT-послуг для освітніх установ, що включає аналіз потреб користувачів в автоматизації навчального процесу та оцінку існуючих рішень на ринку. Досліджувалися функціональні можливості сучасних мобільних застосунків для електронних щоденників. Було проаналізовано кілька популярних застосунків, таких як School Diary, MySchoolApp, ClassDojo, та Edmodo, та визначено їхні основні функції та можливості інтеграції з іншими системами.

Проведено проектування модуля користувача, включаючи аналіз функцій модуля для учнів та батьків, розробку структури бази даних автоматизованої системи електронного щоденника для загальноосвітніх шкіл, а також розробку основних UML-діаграм. Також виконано розробку програмних компонентів модуля користувача, вибір технологій для розробки модуля, налаштування та тестування роботи модуля. Тестування показало ефективність розробленої системи та її придатність для використання в освітньому процесі загальноосвітніх шкіл.

ПЕРЕЛІК ПОСИЛАНЬ

1. Іванова С.М., Петренко О.Д. Цифровізація освіти: перспективи та виклики. – Київ: Наукова думка, 2010. – 184 с.
2. Козловська Н.В., Шевченко І.П. Роль електронних щоденників у сучасній освітній системі. – Львів: Літера, 2012. – 156 с.
3. Мельник Л.Г., Бондаренко О.В. Інтеграція інформаційних технологій в освітній процес. – Одеса: Одеський національний університет, 2011. – 192 с.
4. Павленко А.М., Кравченко В.Ю. Вплив електронних щоденників на залученість учнів до навчального процесу. – Харків: ХНУ імені В. Н. Каразіна, 2013. – 134 с.
5. Сидоренко О.В., Тараненко Л.Б. Оптимізація взаємодії учасників освітнього процесу через електронні щоденники. – Дніпро: Дніпропетровський національний університет, 2014. – 210 с.
6. Johnson, S., Smith, P. R., & Smythe, T. D. Enhancing communication and effectiveness in education systems through digital diaries. – New York: Springer, 2007. – 248 p.
7. Григоренко П.І., Литвиненко Л.О. Аналіз продуктивності навчального процесу з використанням електронних щоденників. – Київ: Київський університет, 2015. – 174 с.
8. Черненко В.М., Петренко А.С. Новітні технології в освіті: виклики та перспективи. – Київ: Академія, 2016. – 234 с.
9. Березюк С.О., Ковальчук Я.Б. Ефективність використання електронних щоденників у навчальних закладах. – Львів: Львівська політехніка, 2017. – 198 с.
10. Захарченко В.Г., Морозова О.І. Технологічні інновації в освіті: аналіз та перспективи. – Харків: Харківський національний університет, 2018. – 212 с.
11. Кузнєцова І.М., Сергієнко В.П. Безпека інформаційних систем в освітніх установах. – Київ: Наукова думка, 2019. – 190 с.

- 12.Лисенко Д.О., Міщенко І.В. Впровадження інноваційних технологій у навчальний процес. – Херсон: Херсонський державний університет, 2020. – 204 с.
- 13.Петрова Г.Є., Шевченко Т.К. Автоматизація навчального процесу: потреби та рішення. – Одеса: Одеський національний університет, 2018. – 178 с.
- 14.Мороз О.П., Кравчук О.В. Вплив автоматизації на ефективність навчання. – Київ: Київський університет, 2019. – 165 с.
- 15.Горбачова Ю.В., Ткаченко В.І. Роль технологій в освітньому процесі. – Львів: Львівська політехніка, 2021. – 212 с.
- 16.Smith, J., & Johnson, L. (2020). Challenges in the Integration of Automated Systems in Education. *Journal of Educational Technology*, 45(2), 112-130.
- 17.Brown, T. (2021). Cybersecurity in Educational Technology: Needs and Solutions. *International Journal of Information Security*, 39(4), 401-420.
- 18.Taylor, R., & Green, M. (2019). Electronic Diary Systems in Schools: An Analysis of Functionality and Usability. *Educational Technology Research and Development*, 67(3), 775-792.
- 19.Застосунок School Diary - "Головна сторінка School Diary". Режим доступу: <https://schooldiary.me/>. Дата доступу: 15 вересня 2024.
- 20.Застосунок MySchoolApp - "Welcome to MySchoolApp". Режим доступу: <https://www.myschoolapps.com/>. Дата доступу: 16 вересня 2024.
- 21.Застосунок ClassDojo - "ClassDojo Home". Режим доступу: <https://www.classdojo.com/>. Дата доступу: 17 вересня 2024.
- 22.Застосунок Edmodo - "Edmodo: Connect With Students and Parents". Режим доступу: <https://edmodo.online/>. Дата доступу: 18 вересня 2024.
- 23.Wilson, A., & Harris, D. (2022). Mobile Application Integration in Educational Environments: A Comprehensive Study. *Journal of Educational Technology Systems*, 50(1), 34-58.
- 24.Patel, S., & Kumar, V. (2021). Enhancing Educational Outcomes through Mobile App Integration. *Education and Information Technologies*, 26(5), 5473-5490.
- 25.Johnson, M. K., & Thompson, L. (2022). Database Design for User Management in Educational Systems. *Journal of Database Management*, 33(2), 89-107.

- 26.Lee, C., & Kim, Y. (2021). Effective Data Handling in School Management Systems. *International Journal of Information Management*, 58, 102-115.
- 27.Evans, R., & Davis, P. (2020). Architecting Databases for User Authentication in School Electronic Diaries. *Database Systems Journal*, 41(4), 253-270.
- 28.Ковальчук В.А., Петренко О.М. Розробка UML-діаграм для систем управління навчанням. – Київ: Видавництво "Наукова думка", 2023. – 234 с.
- 29.Мельник І.Б., Литвиненко Ю.О. Використання Use Case діаграм в проектуванні програмного забезпечення освітніх систем. – Львів: Львівська політехніка, 2022. – 188 с.
- 30.Сидоренко О.В., Грищенко В.М. Архітектура компонентів інформаційних систем в освіті. – Харків: Харківський національний університет, 2021. – 256 с.
- 31.Захарченко В.С., Кравченко С.І. Використання SQLite у мобільних застосунках. – Дніпро: Дніпровський національний університет, 2022. – 142 с.
- 32.Петров Ю.Б., Іванова М.О. Розробка веб-інтерфейсів з використанням React та Node.js. – Одеса: Одеський політехнічний університет, 2021. – 196 с.
- 33.Шевченко Т.Г., Лисенко Д.О. Хмарні рішення для освітніх систем: AWS як платформа. – Харків: Харківський політехнічний інститут, 2023. – 210 с.
- 34.Anderson, J., & Lee, S. (2021). Software Development Lifecycle for Mobile Applications in Education. *Journal of Software Engineering in Education*, 12(3), 145-162.
- 35.Thompson, R., & Harris, D. (2022). Implementing MVVM Architecture in Educational Mobile Apps. *Mobile Computing and Applications*, 9(1), 77-95.
- 36.Patel, S., & Kumar, V. (2021). Data Binding and LiveData for Responsive Educational Apps. *Journal of Modern Education Review*, 11(4), 298-312.
- 37.Wilson, A., & Brown, T. (2020). Room Persistence Library: Enhancing Database Operations in Android Applications. *International Journal of Mobile Software Engineering*, 7(2), 34-50.
- 38.Green, M., & Taylor, R. (2022). React: Building Reusable UI Components for Educational Platforms. *Journal of Web Development in Education*, 15(2), 230-245.

39. Johnson, L., & Smith, J. (2021). Kotlin and Android Jetpack: Enhancing Android App Development. *International Journal of Mobile Application Development*, 8(3), 164-178.
40. Patel, S., & Kumar, V. (2020). Advanced Techniques in Mobile UI Design with Android Jetpack. *Mobile User Experience*, 6(4), 202-219.
41. Brown, T., & Johnson, L. (2021). Simplifying Database Operations with Room Library in Android Applications. *Journal of Mobile Technology*, 14(2), 85-102.
42. Patel, S., & Kumar, V. (2020). Enhancing Local Data Management in Mobile Apps Using Room. *International Journal of Mobile Software Engineering*, 7(3), 45-59.
43. Wilson, A., & Harris, D. (2022). Implementing Persistent Storage Solutions in Mobile Applications. *Mobile Computing and Applications*, 9(4), 234-250.
44. Thompson, R., & Green, M. (2021). Offline Data Handling in Mobile Applications: Benefits and Techniques. *Journal of Modern Mobile Technologies*, 10(1), 112-129.
45. Lee, C., & Kim, Y. (2020). Building Scalable Server-side Applications with Node.js and Express. *International Journal of Web Development*, 15(3), 202-218.
46. Evans, R., & Davis, P. (2021). API Development for Mobile-Server Communication. *Journal of Network and Computer Applications*, 44(2), 134-150.
47. Smith, J., & Lee, H. (2022). Version Control in Software Development: Emphasizing Git for Collaborative Projects. *Journal of Software Engineering Practices*, 16(4), 345-360.
48. Patel, R., & Kumar, A. (2021). Unit Testing in Android with JUnit: A Comprehensive Guide. *International Journal of Software Testing*, 13(2), 158-174.
49. Thompson, E., & Harris, D. (2020). Integration and Acceptance Testing Techniques for Mobile Applications. *Mobile Application Testing Journal*, 11(1), 89-105.
50. Методичні вказівки до виконання магістерських кваліфікаційних робіт для студентів <https://iq.vntu.edu.ua/repository/getfile.php/6077.pdf>
51. Yukhymchuk M. Decentralized Coordination of Temperature Control in Multiarea Premises [Text] / M. Yukhymchuk, V. Dubovoi, V. Kovtun // *Complexity*. – 2022. – Vol. 22. – Article ID 2588364.

- 52.M. Yukhymchuk, V. Dubovoi, V. Kovtun, "Decentralized Coordination of Temperature Control in Multiarea Premises", Complexity, vol. 2022, Article ID 2588364, 18 pages, 2022. <https://doi.org/10.1155/2022/2588364>
- 53.Yukhymchuk M., Dubovoi V., Kovtun V. «Decentralized Coordination of Temperature Control in Multiarea Premises». Complexity. 2022. ID 2588364, 18 pages, <https://doi.org/10.1155/2022/2588364>.
- 54.Dubovoi V. Optimal Decomposition of Control of Distributed Cyber-physical System / V. Dubovoi, D. Sembrat, M. Yukhymchuk // The 11th IEEE International Conference on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications (IDAACS), Cracow, Poland, 22-25 September, 2021. - Cracow, 2021. - P. 601–605. - DOI: <https://doi.org/10.1109/IDAACS53288.2021.9661007>
- 55.Dubovoi V. Smart Control of Multi-zone Object Heating with Multi-source System / Volodymyr Dubovoi, Maria Yukhymchuk, Hanna Stepanenko, Stanislav Perepelytsia // IEEE 2nd Ukraine Conference on Electrical and Computer Engineering (UKRCON), Lviv, Ukraine, 2019. - Lviv, 2019. - P. 1018–1021. - DOI:<https://doi.org/10.1109/UKRCON.2019.8879942>

ДОДАТКИ

Додаток А
(обов'язковий)

ПРОТОКОЛ ПЕРЕВІРКИ НАВЧАЛЬНОЇ (КВАЛІФІКАЦІЙНОЇ) РОБОТИ

Назва роботи: « Розробка модуля користувача мобільного застосунку
автоматизованої системи електронного щоденника
для загальноосвітніх шкіл »

Тип роботи: магістерська кваліфікаційна робота

(кваліфікаційна робота, курсовий проект (робота), реферат, аналітичний огляд, інше (зазначити))

Підрозділ кафедра комп'ютерних систем управління
(кафедра, факультет (інститут), навчальна група)

Керівник Юхимчук М. С., професор кафедри КСУ
(прізвище, ініціали, посада)

Показники звіту подібності

Turnitin	
Оригінальність	98%
Загальна схожість	2%

Аналіз звіту подібності (відмітити потрібне)

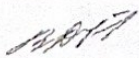
✓ Запозичення, виявлені у роботі, оформлені коректно і не містять ознак плагіату.

- Виявлені у роботі запозичення не мають ознак плагіату, але їх надмірна кількість викликає сумніви щодо цінності роботи і відсутності самостійності її автора. Роботу направити на доопрацювання.
- Виявлені у роботі запозичення є недобросовісними і мають ознаки плагіату та/або в ній містяться навмисні спотворення тексту, що вказують на спроби приховування недобросовісних запозичень.

Заявляю, що ознайомлений (-на) з повним звітом подібності, який був згенерований Системою щодо роботи (додається)

Автор  Денис ПАЛАМАРЧУК
(підпис) (прізвище, ініціали)

Опис прийнятого рішення

Особа, відповідальна за перевірку  Володимир ДУБОВОЙ

Експерт _____

Додаток Б
(обов'язковий)
ВНТУ

ЗАТВЕРДЖЕНО
Зав. кафедри КСУ ВНТУ,
д.т.н., професор
 В'ячеслав КОВТУН
“ 14 ” 10 2024 р.

ТЕХНІЧНЕ ЗАВДАННЯ
на виконання магістерської кваліфікаційної роботи
Розробка модуля користувача мобільного застосунку автоматизованої
системи електронного щоденника для загальноосвітніх шкіл

08-33.МКР.007.00.000 ТЗ

Студент групи ЗАКІТР-23м



Підпис

Денис ПАЛАМАРЧУК

Ім'я ПРІЗВИЩЕ

Керівник



Підпис

Марія ІУХИМЧУК

Ім'я ПРІЗВИЩЕ

Вінниця 2024

1. Назва та галузь застосування

1.1. Назва – Розробка модуля користувача мобільного застосунку автоматизованої системи електронного щоденника для загальноосвітніх шкіл.

1.2. Галузь застосування – Інформаційні технології у сфері освіти, спеціалізовані освітні платформи для автоматизації шкільного адміністрування та навчального процесу.

2. Підстава для проведення розробки.

Тема магістерської кваліфікаційної роботи затверджена наказом по ВНТУ № 310 від 17-09-2024 р.

3. Мета та призначення розробки.

Мета магістерської роботи полягає у створенні модуля користувача для мобільного застосунку електронного щоденника в школах, що полегшить взаємодію між учасниками освітнього процесу, оптимізує управління навчальними матеріалами та підвищить ефективність навчання.

4. Джерела розробки.

Магістерська кваліфікаційна робота виконується вперше. В ході проведення розробки повинні використовуватись такі документи:

1. Сидоренко О.Є. Аналіз ефективності використання електронних щоденників у загальноосвітніх навчальних закладах / О.Є. Сидоренко // Інформаційні технології в освіті. – 2019. – № 3. – С. 204-210.
2. Бондаренко Т.І. Мобільні застосунки як інструмент модернізації освітнього процесу / Т.І. Бондаренко // Український педагогічний журнал. – 2021. – № 2. – С. 58-64.
3. Кравчук Л.А. Сучасні підходи до автоматизації управління навчальним процесом у школах / Л.А. Кравчук // Вісник Львівського університету. Серія інформаційні системи та мережі. – 2022. – Вип. 19. – С. 134-140.

5. Вимоги до розробки.

5.1. Перелік головних етапів виконання розробки:

- визначити аспекти функціонування мобільного застосунку електронного щоденника;
- дослідити функціональні можливості існуючих аналогічних систем;
- скласти вимоги до автоматизованої системи, що розробляється;
- спроектувати та розробити систему у відповідності до поставлених вимог та функцій:
 - забезпечення доступу користувачів до системи та персональних аккаунтів;
 - управління домашніми завданнями та оцінками;
 - перегляд розкладу занять;
 - відправлення та отримання повідомлень від вчителів;

- перегляд відомостей про відвідуваність.

5.2. Основні технічні вимоги до розробки.

5.2.1. Вимоги до програмної платформи:

- підтримка мобільних операційних систем Android.

5.2.2. Умови експлуатації системи:

- робота на мобільних пристроях з актуальними версіями операційних систем;
- можливість цілодобового функціонування системи;
- програмне забезпечення системи повинно бути захищене від несанкціонованого доступу.

6. Стадії та етапи розробки.

6.1 Пояснювальна записка:

- Дослідження об'єкта автоматизації та постановка задач дослідження 03.10.2024р.
- Дослідження та аналіз роботи систем управління навчальним процесом в загальноосвітніх школах 19.10.2024р.
- Дослідження функціональних можливостей сучасних мобільних застосунків 01.11.2024р.
- Проектування модуля користувача мобільного застосунку автоматизованої системи електронного щоденника для загальноосвітніх шкіл 15.11.2024р.
- Апробація результатів дослідження 20.11.2024р.
- Оформлення пояснювальної записки, графічного матеріалу і презентації 29.11.2024р.

6.2 Графічні матеріали:

- UML-діаграми 04.11.2024 р.
- ER-діаграма структури бази даних 18.11.2024 р.
- вигляд екранів розробленого мобільного додатку 18.11.2024 р.

7. Порядок контролю і приймання.

- 7.1. Хід виконання роботи контролюється керівником роботи. Рубіжний контроль провести до «20»__11__ 2024 р.
- 7.2. Атестація МКР здійснюється на попередньому захисті. Попередній захист магістерської кваліфікаційної роботи провести до «1»__12__ 2024 р.
- 7.3. Підсумкове рішення щодо оцінки якості виконання роботи приймається на засіданні ЕК. Захист магістерської кваліфікаційної роботи провести до «11»__12__ 2024 р.

Додаток В (вибірковий)

Лістинг програмного забезпечення

```

flask import Flask, request, jsonify
from # Імпорт необхідних бібліотек
from flask_sqlalchemy import SQLAlchemy

# Ініціалізація Flask додатку
app = Flask(__name__)

# Налаштування з'єднання з базою даних
app.config['SQLALCHEMY_DATABASE_URI'] = 'sqlite:///school_diary.db'
app.config['SQLALCHEMY_TRACK_MODIFICATIONS'] = False
db = SQLAlchemy(app)

# Модель бази даних для користувача
class User(db.Model):
    id = db.Column(db.Integer, primary_key=True)
    username = db.Column(db.String(50), unique=True, nullable=False)
    role = db.Column(db.String(50), nullable=False)

# Модель бази даних для записів щоденника
class Entry(db.Model):
    id = db.Column(db.Integer, primary_key=True)
    date = db.Column(db.Date, nullable=False)
    subject = db.Column(db.String(50), nullable=False)
    homework = db.Column(db.String(200), nullable=True)
    user_id = db.Column(db.Integer, db.ForeignKey('user.id'), nullable=False)

# Створення таблиць у базі даних
db.create_all()

# API для створення нового користувача
@app.route('/users', methods=['POST'])
def create_user():
    data = request.get_json()
    new_user = User(username=data['username'], role=data['role'])
    db.session.add(new_user)
    db.session.commit()
    return jsonify({'message': 'User created successfully'}), 201

# API для отримання записів щоденника користувача
@app.route('/entries/<int:user_id>', methods=['GET'])
def get_entries(user_id):
    entries = Entry.query.filter_by(user_id=user_id).all()
    return jsonify([{'date': entry.date, 'subject': entry.subject, 'homework': entry.homework} for entry in entries])

# Запуск Flask додатку
if __name__ == '__main__':
    app.run(debug=True)

import 'package:flutter/material.dart';
import 'package:http/http.dart' as http;
import 'dart:convert';

void main() => runApp(MyApp());

class MyApp extends StatelessWidget {
  @override
  Widget build(BuildContext context) {
    return MaterialApp(
      title: 'School Diary',
      theme: ThemeData(
        primarySwatch: Colors.blue,
      ),
      home: MyHomePage(),
    );
  }
}

class MyHomePage extends StatefulWidget {
  @override
  _MyHomePageState createState() => _MyHomePageState();
}

class _MyHomePageState extends State<MyHomePage> {
  List<dynamic> entries = [];

  @override
  void initState() {

```



```

        decoration: InputDecoration(labelText: 'Date
(YYYY-MM-DD)'),
        validator: (value) {
          if (value.isEmpty) {
            return 'Please enter the date';
          }
          return null;
        },
      ),
      TextFormField(
        controller: _subjectController,
        decoration: InputDecoration(labelText: 'Subject'),
        validator: (value) {
          if (value.isEmpty) {
            return 'Please enter the subject';
          }
          return null;
        },
      ),
      TextFormField(
        controller: _homeworkController,
        decoration: InputDecoration(labelText:
'Homework'),
        validator: (value) {
          if (value.isEmpty) {
            return 'Please enter the homework';
          }
          return null;
        },
      ),
      Padding(
        padding: const EdgeInsets.symmetric(vertical:
16.0),
        child: ElevatedButton(
          onPressed: addEntry,
          child: Text('Submit'),
        ),
      ),
    ],
  ),
),
),
);
}
}

```

// Видалення запису з електронного щоденника

```

import 'package:flutter/material.dart';
import 'package:http/http.dart' as http;

class EntryDetailsPage extends StatelessWidget {
  final Map<String, dynamic> entry;

  EntryDetailsPage({@required this.entry});

  Future<void> deleteEntry(int id) async {
    final response = await
http.delete(Uri.parse('http://localhost:5000/entries/$id'));

    if (response.statusCode == 200) {
      // Повернення до попереднього екрану після
видалення
      Navigator.pop(context);
    } else {
      throw Exception('Failed to delete entry');
    }
  }

  @override
  Widget build(BuildContext context) {
    return Scaffold(
      appBar: AppBar(
        title: Text('Entry Details'),
        actions: <Widget>[
          IconButton(
            icon: Icon(Icons.delete),
            onPressed: () => deleteEntry(entry['id']),
          ),
        ],
      ),
      body: Padding(
        padding: const EdgeInsets.all(16.0),
        child: Column(
          crossAxisAlignment: CrossAxisAlignment.start,
          children: <Widget>[
            Text('Date: ${entry['date']}', style:
TextStyle(fontSize: 18)),
            SizedBox(height: 10),
            Text('Subject: ${entry['subject']}', style:
TextStyle(fontSize: 18)),
            SizedBox(height: 10),
            Text('Homework: ${entry['homework']}', style:
TextStyle(fontSize: 18)),
          ],
        ),
      ),
    );
  }
}

```

```

    ),
  );
}
}

```

// Редагування запису в електронному щоденнику

```

import 'package:flutter/material.dart';
import 'package:http/http.dart' as http;
import 'dart:convert';

```

```

class EditEntryPage extends StatefulWidget {
  final Map<String, dynamic> entry;

  EditEntryPage({@required this.entry});

  @override
  _EditEntryPageState createState() =>
    _EditEntryPageState();
}

```

```

class _EditEntryPageState extends State<EditEntryPage> {
  final _formKey = GlobalKey<FormState>();
  TextEditingController _subjectController;
  TextEditingController _homeworkController;
  TextEditingController _dateController;

  @override
  void initState() {
    super.initState();
    _subjectController = TextEditingController(text:
    widget.entry['subject']);
    _homeworkController = TextEditingController(text:
    widget.entry['homework']);
    _dateController = TextEditingController(text:
    widget.entry['date']);
  }
}

```

```

Future<void> updateEntry(int id) async {
  if (_formKey.currentState.validate()) {
    final response = await http.put(
      Uri.parse('http://localhost:5000/entries/$id'),
      headers: <String, String>{
        'Content-Type': 'application/json; charset=UTF-8',
      },
      body: jsonEncode(<String, String>{
        'date': _dateController.text,
        'subject': _subjectController.text,

```

```

        'homework': _homeworkController.text,
      })),
    );

```

```

    if (response.statusCode == 200) {
      Navigator.pop(context);
    } else {
      throw Exception('Failed to update entry');
    }
  }
}

@override
Widget build(BuildContext context) {
  return Scaffold(
    appBar: AppBar(
      title: Text('Edit Entry'),
    ),
    body: Form(
      key: _formKey,
      child: Padding(
        padding: const EdgeInsets.all(16.0),
        child: Column(
          children: <Widget>[
            TextFormField(
              controller: _dateController,
              decoration: InputDecoration(labelText: 'Date
(YYYY-MM-DD)'),
              validator: (value) {
                if (value.isEmpty) {
                  return 'Please enter the date';
                }
                return null;
              },
            ),
            TextFormField(
              controller: _subjectController,
              decoration: InputDecoration(labelText: 'Subject'),
              validator: (value) {
                if (value.isEmpty) {
                  return 'Please enter the subject';
                }
                return null;
              },
            ),
            TextFormField(
              controller: _homeworkController,

```

```

        decoration: InputDecoration(labelText:
'Homework'),
        validator: (value) {
            if (value.isEmpty) {
                return 'Please enter the homework';
            }
            return null;
        },
    ),
    Padding(
        padding: const EdgeInsets.symmetric(vertical:
16.0),
        child: ElevatedButton(
            onPressed: () => updateEntry(widget.entry['id']),
            child: Text('Update'),
        ),
    ),
],
),
),
),
);
}
}

```

```
from flask import Flask, jsonify
```

```
from flask_sqlalchemy import SQLAlchemy
```

```
app = Flask(__name__)
```

```
app.config['SQLALCHEMY_DATABASE_URI'] =
```

```
'sqlite:///school_diary.db'
```

```
db = SQLAlchemy(app)
```

```
class Grade(db.Model):
```

```
    id = db.Column(db.Integer, primary_key=True)
```

```
    subject = db.Column(db.String(50), nullable=False)
```

```
    grade = db.Column(db.Float, nullable=False)
```

```
    student_id = db.Column(db.Integer, nullable=False)
```

```
@app.route('/grades', methods=['GET'])
```

```
def get_grades():
```

```
    grades_data = db.session.query(
```

```
        Grade.subject,
```

```
        db.func.avg(Grade.grade).label('average_grade')
```

```
    ).group_by(Grade.subject).all()
```

```
    grades_dict = {grade.subject: grade.average_grade for
grade in grades_data}
```

```
    return jsonify(grades_dict)
```

```
if __name__ == '__main__':
```

```
    app.run(debug=True)
```

```
class Message(db.Model):
```

```
    id = db.Column(db.Integer, primary_key=True)
```

```
    content = db.Column(db.String(500), nullable=False)
```

```
    sender_id = db.Column(db.Integer, nullable=False)
```

```
@app.route('/messages', methods=['GET'])
```

```
def get_messages():
```

```
    messages = Message.query.all()
```

```
    return jsonify([{'id': message.id, 'content':
message.content, 'sender_id': message.sender_id} for
message in messages])
```

```
@app.route('/messages', methods=['POST'])
```

```
def post_message():
```

```
    data = request.get_json()
```

```
    new_message = Message(content=data['content'],
```

```
sender_id=data['sender_id'])
```

```
    db.session.add(new_message)
```

```
    db.session.commit()
```

```
    return jsonify({'message': 'Message sent successfully'}),
```

```
201
```

```
if __name__ == '__main__':
```

```
    app.run(debug=True)
```

```
// Сторінка для перегляду оголошень
```

```
import 'package:flutter/material.dart';
```

```
import 'package:http/http.dart' as http;
```

```
import 'dart:convert';
```

```
class AnnouncementsPage extends StatefulWidget {
```

```
    @override
```

```
    _AnnouncementsPageState createState() =>
```

```
    _AnnouncementsPageState();
```

```
}
```

```
class _AnnouncementsPageState extends
```

```
State<AnnouncementsPage> {
```

```
    List<dynamic> announcements = [];
```

```
@override
```

```

void initState() {
  super.initState();
  fetchAnnouncements();
}

fetchAnnouncements() async {
  final response = await
http.get(Uri.parse('http://localhost:5000/announcements'));

  if (response.statusCode == 200) {
    setState(() {
      announcements = json.decode(response.body);
    });
  } else {
    throw Exception('Failed to load announcements');
  }
}

@override
Widget build(BuildContext context) {
  return Scaffold(
    appBar: AppBar(
      title: Text('Announcements'),
    ),
    body: ListView.builder(
      itemCount: announcements.length,
      itemBuilder: (context, index) {
        return ListTile(
          title: Text(announcements[index]['title']),
          subtitle: Text(announcements[index]['content']),

```

```

        );
      },
    ),
  );
}
}

from flask import Flask, request, jsonify
from flask_sqlalchemy import SQLAlchemy

app = Flask(__name__)
app.config['SQLALCHEMY_DATABASE_URI'] =
'sqlite:///school_diary.db'
db = SQLAlchemy(app)

class Announcement(db.Model):
    id = db.Column(db.Integer, primary_key=True)
    title = db.Column(db.String(100), nullable=False)
    content = db.Column(db.String(500), nullable=False)

@app.route('/announcements', methods=['GET'])
def get_announcements():
    announcements = Announcement.query.all()
    return jsonify([{'id': announcement.id, 'title':
announcement.title, 'content': announcement.content} for
announcement in announcements])

if __name__ == '__main__':
    app.run(debug=True)

```

Додаток Г
(обов'язковий)

ІЛЮСТРАТИВНА ЧАСТИНА

«Розробка модуля користувача мобільного застосунку автоматизованої системи електронного щоденника для загальноосвітніх шкіл»

1. ER-діаграма структури бази даних автоматизованої системи електронного щоденника для загальноосвітніх шкіл
2. Use Case UML-діаграма модуля користувача в ролі Вчитель
3. Use Case UML-діаграма модуля користувача в ролі Батьки та Учень
4. UML-діаграма класів автоматизованої системи електронного щоденника для загальноосвітніх шкіл
5. UML-діаграма компонентів автоматизованої системи електронного щоденника для загальноосвітніх шкіл
6. Головний екран мобільного додатку з відображенням нових оцінок за замовчуванням
7. Екран відображення поточних оцінок
8. Головний екран мобільного додатку з відображенням нових записів про відвідуваність
9. Екран відображення відвідуваності
10. Екран відображення статистики відвідуваності з групуванням по предметах
11. Головний екран мобільного додатку з відображенням нових домашніх завдань
12. Відображення домашніх завдань в обраний день
13. Детальна інформація про обране домашнє завдання
14. Головний екран мобільного додатку з відображенням нових змін в розкладі
15. Відображення календаря з розкладом занять
16. Відображення усіх повідомлень
17. Відображення деталей обраного повідомлення

Виконав: студент 2-го курсу, групи ЗАКІТР-23м
спеціальності 174 – Автоматизація, комп'ютерно-інтегровані технології та робототехніка

(шифр і назва спеціальності)



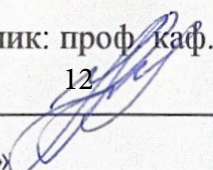
Денис ПАЛАМАРЧУК

(ім'я та прізвище)

Керівник: проф. каф. КСУ д.т.н.

07

12



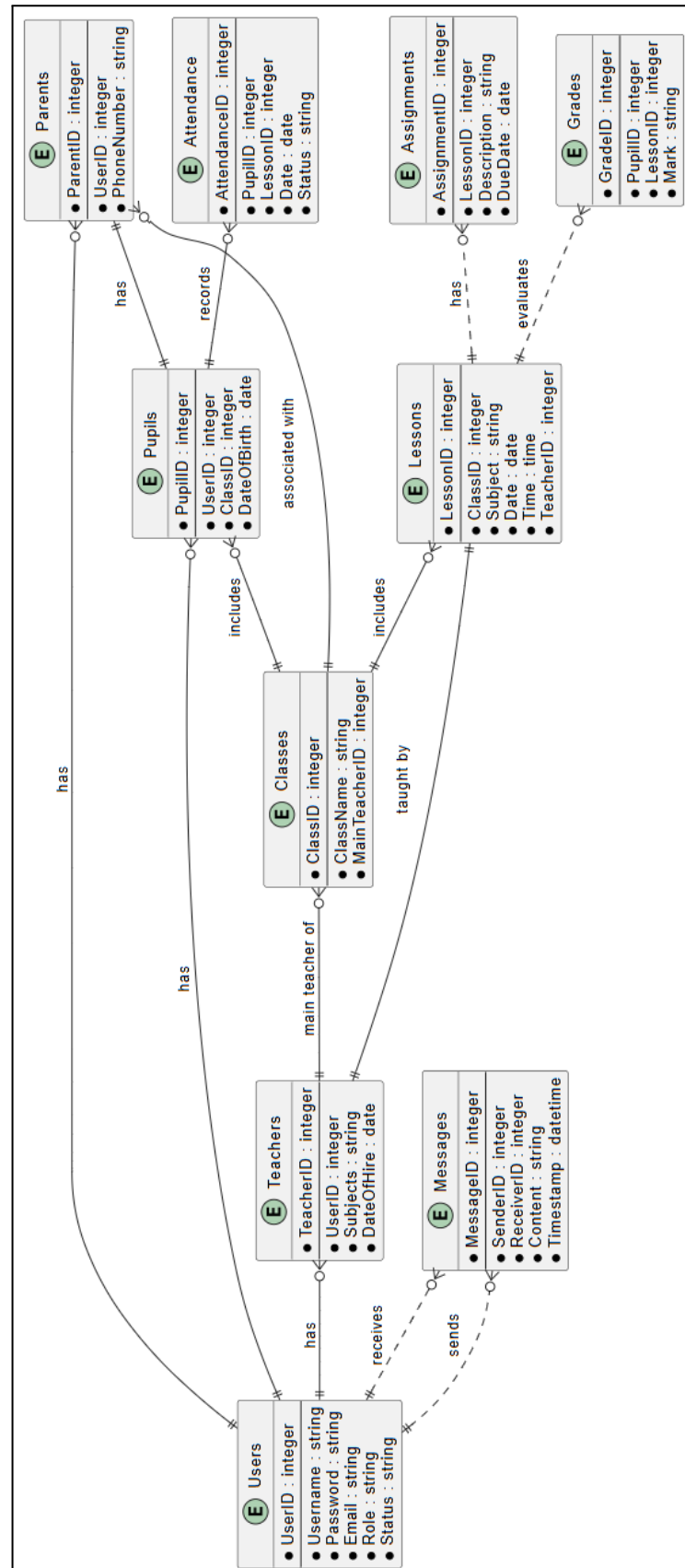
Марія ЮХИМЧУК

(ім'я та прізвище)

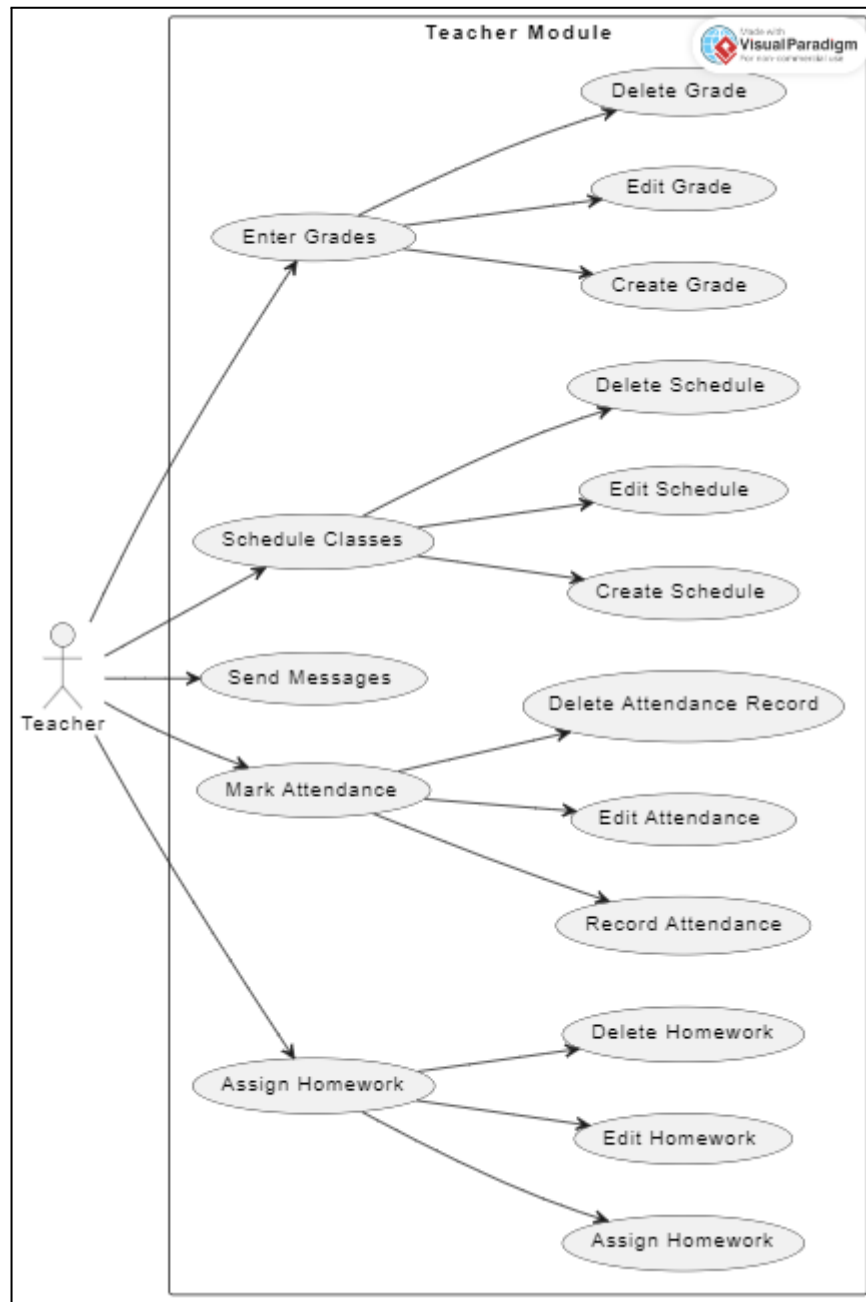
« ____ »

2024 р.

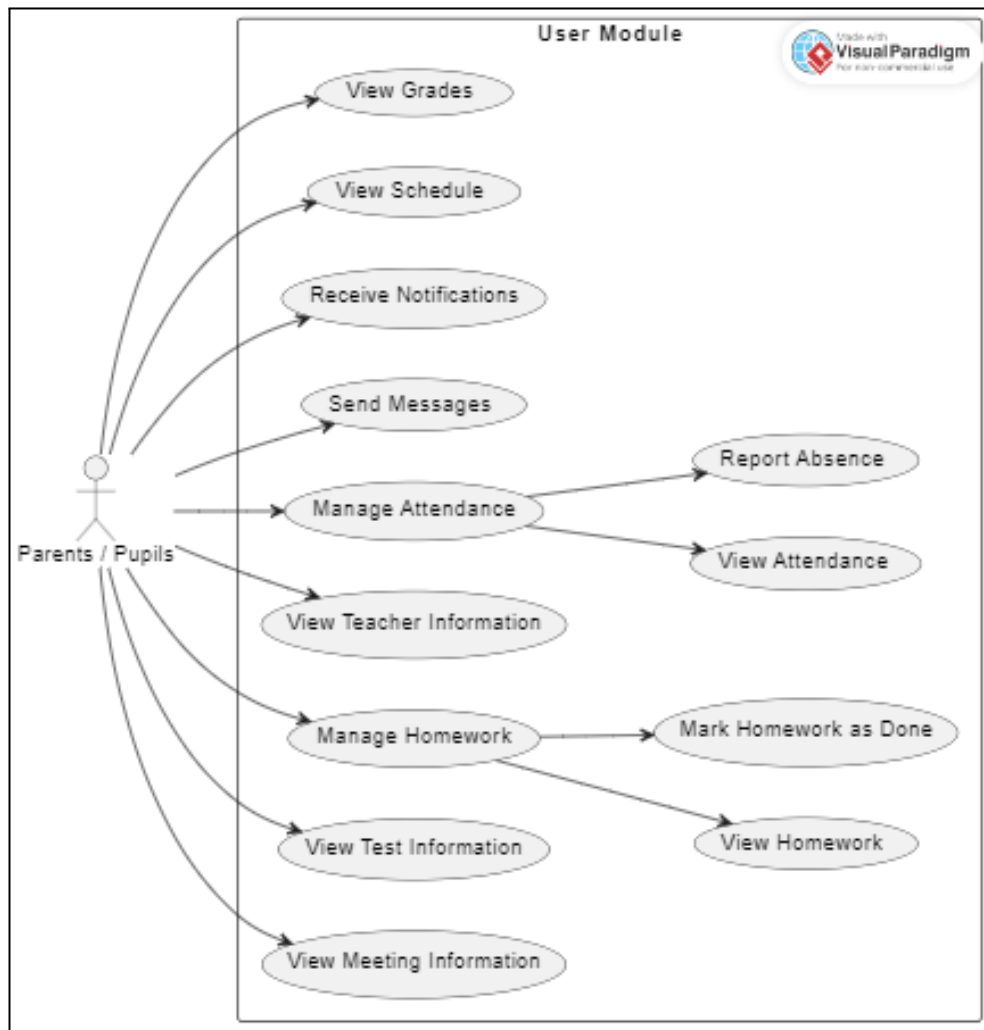
ER-діаграма структури бази даних автоматизованої системи електронного
щоденника для загальноосвітніх шкіл



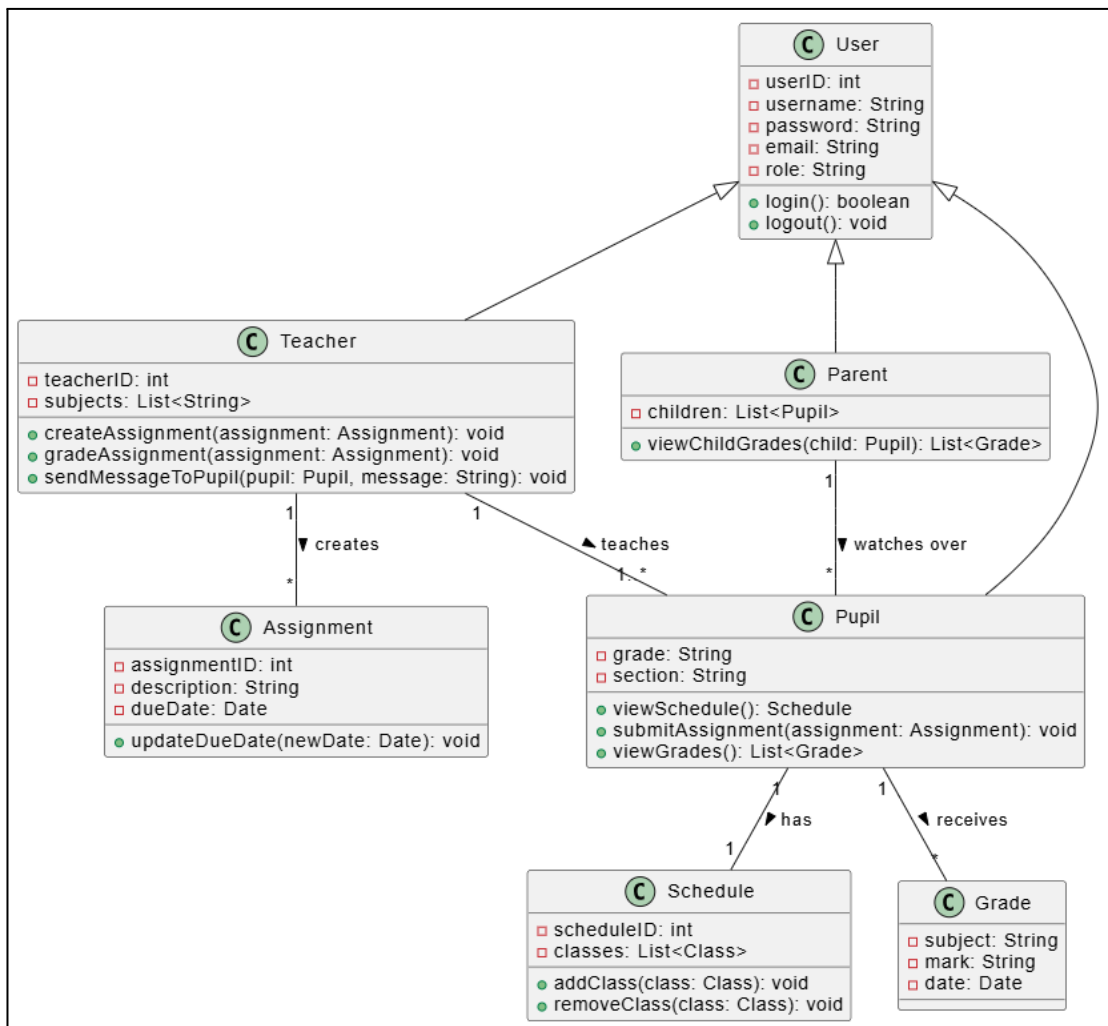
Use Case UML-діаграма модуля користувача в ролі Вчитель



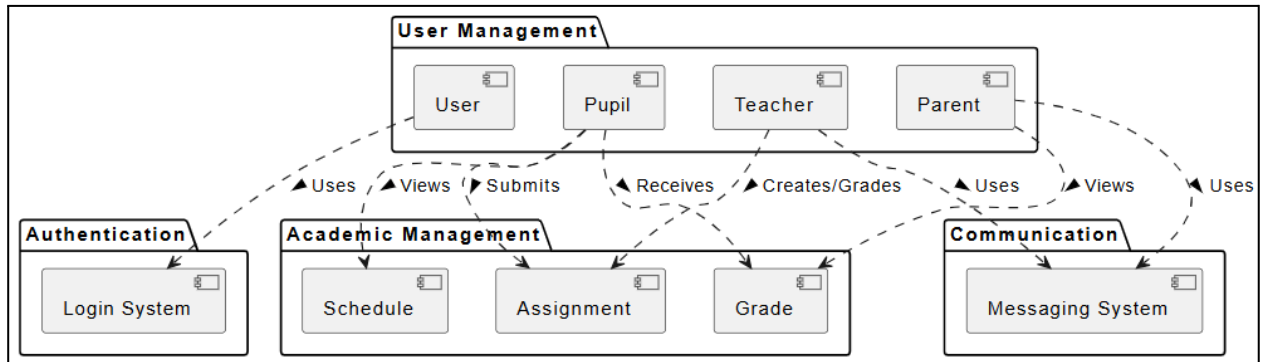
Use Case UML-діаграма модуля користувача в ролі Батьки та Учень



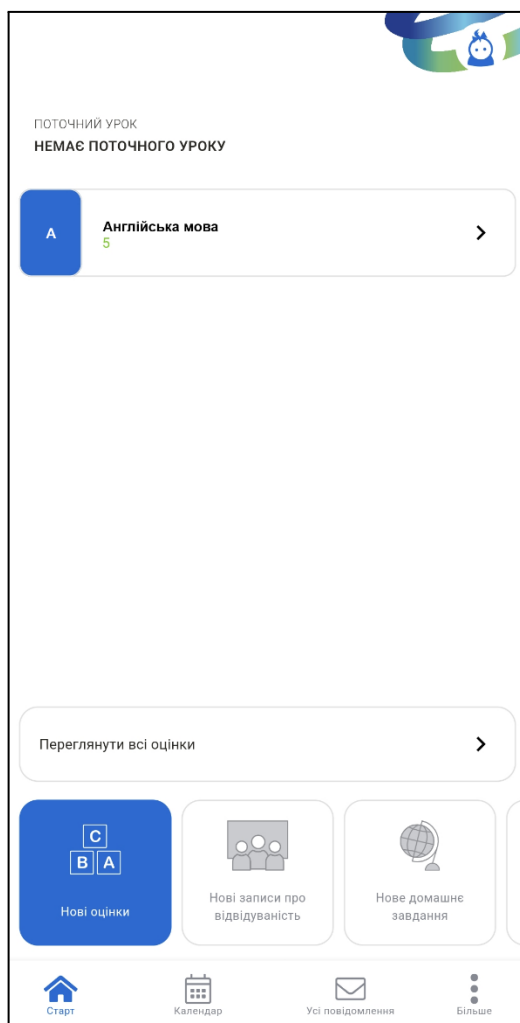
UML-діаграма класів автоматизованої системи електронного щоденника для загальноосвітніх шкіл



UML-діаграма компонентів автоматизованої системи електронного щоденника для загальноосвітніх шкіл



Головний екран мобільного додатку з відображенням нових оцінок за замовчуванням



Екран відображення поточних оцінок

Оцінки

Поточні оцінки

Середньорічні оцінки

Період 1		
	Середнє	
Українська мова 8, 10, 9	9,00	>
Математика 11, 5, 8, 10, 11	9,00	>
Англійська мова 6, 7, 10, 8	7,75	>
Природознавство 10, 10	10,00	>
Малювання 11	11,00	>

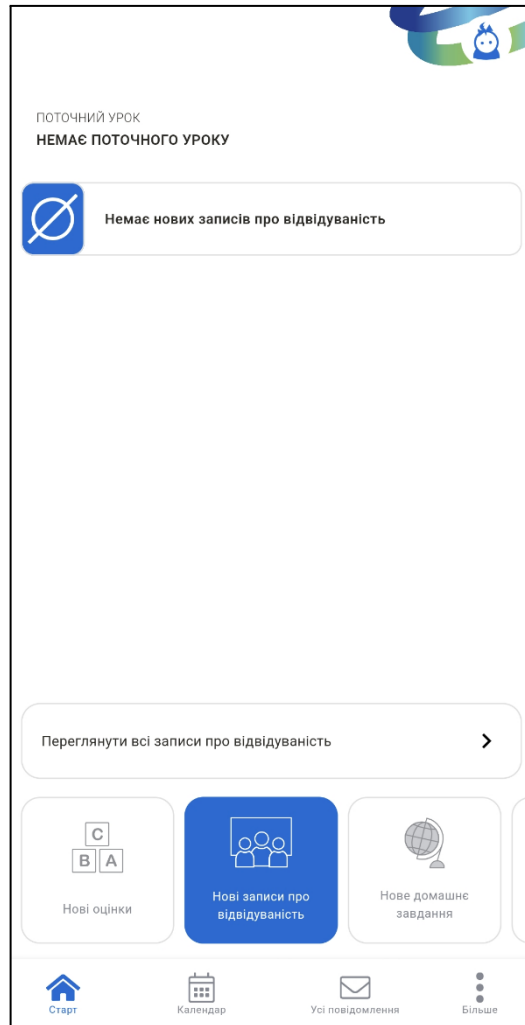
Старт

Календар

Усі повідомлення

Більше

Головний екран мобільного додатку з відображенням нових записів про відвідуваність



Екран відображення відвідуваності

Відвідуваність

Тиждень

Статистика

Предмети

11/2024

11/2024

11/2024

пн

вт

ср

чт

пт

сб

нд

1

2

3

4

5

6

7

8

9

10

11

12

13

14

15

16

17

18

19

20

21

22

23

24

25

26

27

28

29

30

Весь день

Англійська мова

1. 07:45 - 08:30

Математика

2. 08:35 - 09:20

Математика

3. 09:30 - 10:15

Додати пояснення

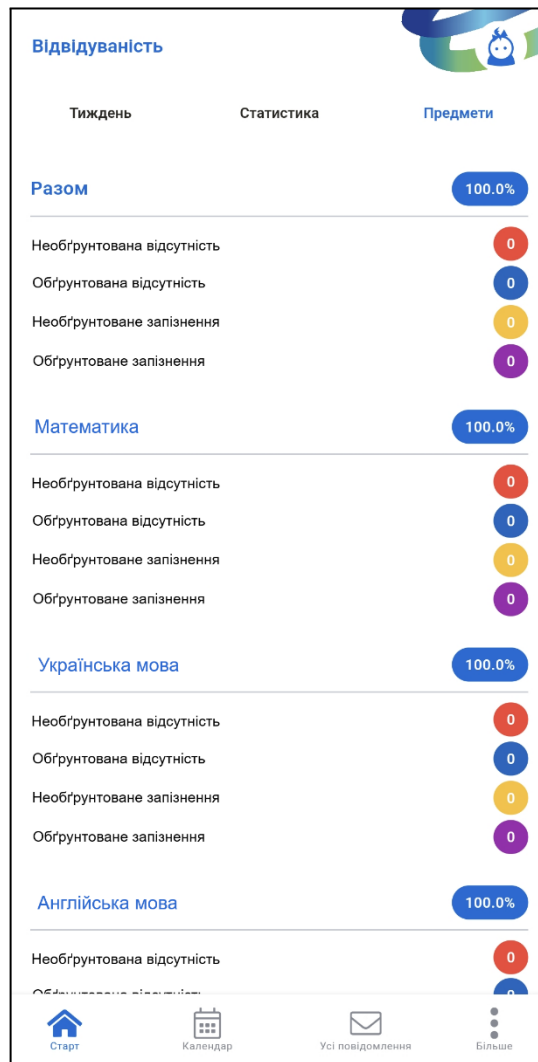
Старт

Календар

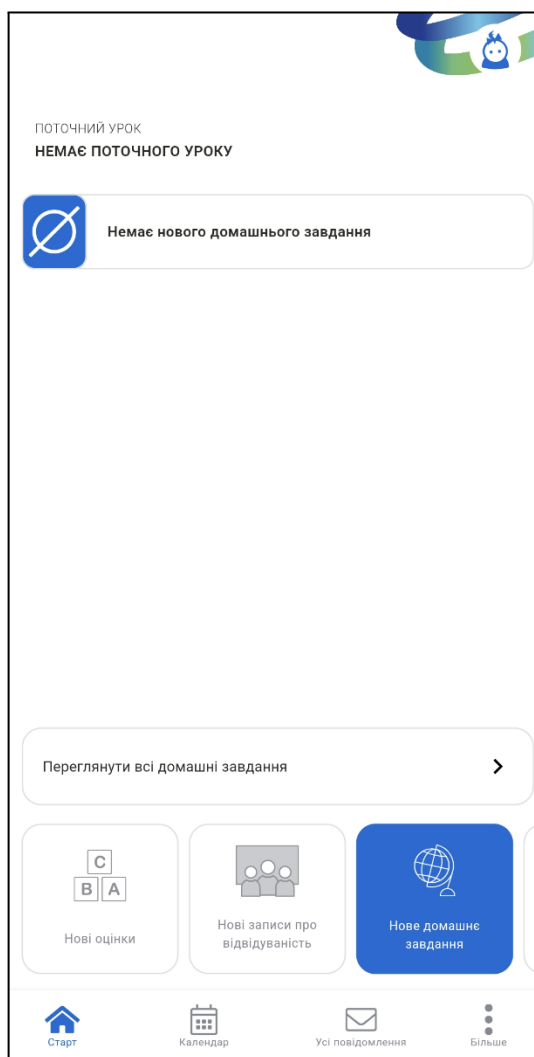
Усі повідомлення

Більше

Екран відображення статистики відвідуваності з групуванням по предметах



Головний екран мобільного додатку з відображенням нових домашніх завдань



Відображення домашніх завдань в обраний день



Детальна інформація про обране домашнє завдання

 Домашнє завдання

ПРЕДМЕТ

Англійська мова

ДАТА СТВОРЕННЯ

15.11.2024

ТЕРМІН ВИКОНАННЯ

15.11.2024

ЗМІСТ

Побудова запитальних речень

wordwall.net/resource/59866/treetops-3-unit-2

НЕОБХІДНА ВІДПОВІДЬ

Ні

Позначити як виконане

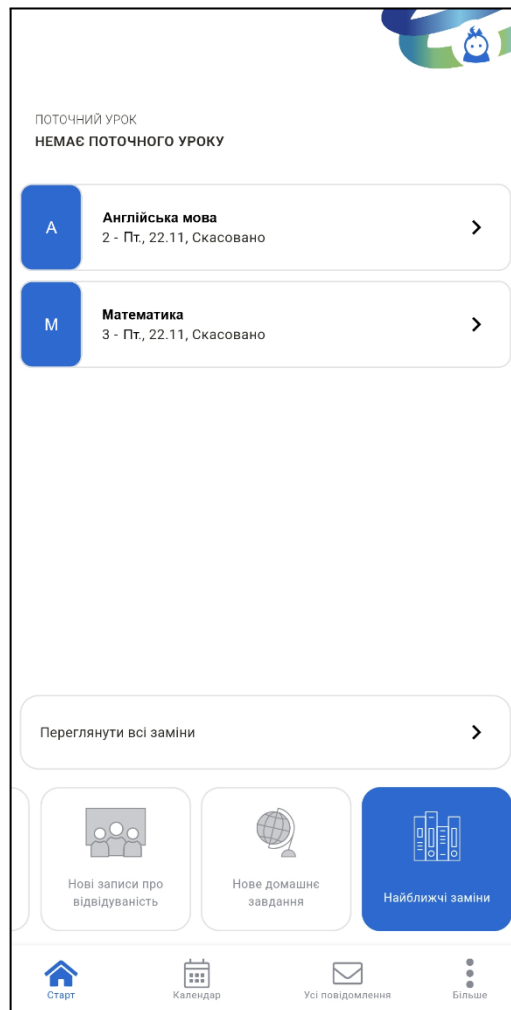
 Старт

 Календар

 Усі повідомлення

 Більше

Головний екран мобільного додатку з відображенням нових змін в розкладі



Відображення календаря з розкладом занять

Календар

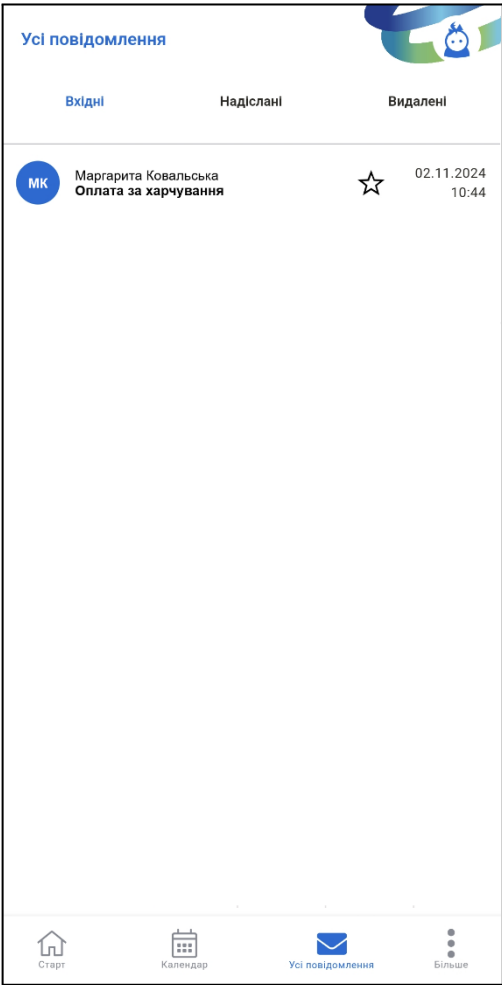
День Тиждень

ЛИС ПН 18 ВТ 19 СР 20 ЧТ 21 ПТ 22

07:00					
08:00	Математика		Українська мова	Математика	
09:00	Англійська мова	Математика	Англійська мова	Українська мова	Малювання
10:00	Англійська мова	Українська мова	Англійська мова	Природозна...	
11:00	Українська мова	Англійська мова	Математика		Математика
12:00	Українська мова	Українська мова			Англійська мова
13:00		Природозна...			
14:00					
15:00					

Старт Календар Усі повідомлення Більше

Відображення усіх повідомлень



Відображення деталей обраного повідомлення

