

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ЛЬВІВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ ІМЕНІ ІВАНА ФРАНКА

Факультет прикладної математики та інформатики

Кафедра прикладної математики

Магістерська робота
ІНФОРМАЦІЙНА СИСТЕМА ПОШУКУ ПРИМІЩЕНЬ У
БУДІВЛІ

Виконав: студент групи ПМППМ–22
спеціальності
113 – прикладна математика

_____ Божук В. В. _____
(прізвище та ініціали)

Керівник _____ Стягар А. О. _____
(прізвище та ініціали)

Рецензент _____
(прізвище та ініціали)

Львів – 2023

ЗМІСТ

ВСТУП.....	3
1. ПОСТАНОВКА ЗАДАЧІ ТА ВИКОРИСТАНІ ТЕХНОЛОГІЇ.....	4
1.1 Постановка задачі.....	4
1.2 Опис використаних технологій та середовищ розробки програм.....	5
1.2.1 Flutter.....	5
1.2.2 Firebase Database.....	6
1.2.3 Середовища програмування.....	7
1.2.4 Blender.....	8
1.2.5 .NET і WPF.....	9
1.2.6 Adobe Photoshop CC.....	10
2. ПРОЕКТУВАННЯ ТА ПРОГРАМНА РЕАЛІЗАЦІЯ ДОДАТКУ ВІДПОВІДНО ДО ВИМОГ.....	11
2.1 Загальний опис та характеристика продукту.....	11
2.2 Побудова карти поверхів.....	13
2.3 Загальна архітектура проекту.....	14
2.4 База даних проекту.....	16
2.5 Стиснення зображень.....	17
2.6 Особливості реалізації.....	18
2.7 Алгоритм пошуку.....	19
3. СЦЕНАРІЙ РОБОТИ МОБІЛЬНОГО ДОДАТКУ.....	20
ВИСНОВКИ.....	24
СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ ТА ІНТЕРНЕТ РЕСУРСІВ.....	25

ВСТУП

Протягом життя людина часто опиняється в ситуації, коли їй потрібно швидко знайти шлях до необхідного приміщення всередині великої будівлі. Багато поверхів, заплутані коридори та не завжди логічна нумерація дверей, або, навіть, її відсутність, дуже ускладнюють пошуки. Зазвичай, в таких ситуаціях ми починаємо шукати когось, хто підкаже шлях, але не завжди ця задача дається нам легко. Відсутність точної інформації про розташування необхідного приміщення створює незручності в його пошуку.

Для вирішення цієї проблеми вирішено розробити мобільний додаток, який допоможе користувачеві знайти найкоротший шлях до необхідного місця в будівлі. За допомогою цього додатку, користувач має можливість ввівши назву приміщення, отримати інформацію про найкоротший шлях до нього від вказаного місця перебування. Також користувач повинен мати доступ до віртуальної карти будівлі для ознайомлення з приміщеннями.

1. ПОСТАНОВКА ЗАДАЧІ ТА ВИКОРИСТАНІ ТЕХНОЛОГІЇ

1.1 Постановка задачі.

Завданням магістерської роботи є спроектувати мобільний додаток призначений для знаходження найефективнішого шляху до будь-якого приміщення в будівлі. В програмі повинно бути можливість ввести, вибрати зі списку або вказати на карті поточне та цільове місце розташування і отримати інформацію про найкоротший шлях між ними. Також, програма повинна містити віртуальну карту приміщення з можливістю довільного переміщення по визначених на ній точках. Інформація про маршрут повинна бути подана у вигляді двовимірної карти та доповнена панорамними фотографіями її вершин. Кожна фотографія відповідає вершині в орієнтованому зваженному графі з відміченими на ній точками для переходу на наступні вершини у напрямку вказаного приміщення. Найкоротший шлях вираховується з допомогою алгоритму Дейкстри.

Доступ до програмного забезпечення повинен здійснюватися із будь-якого сучасного смартфона з операційною системою Android.

Вимоги перед програмним продуктом:

- Відображення двовимірної карти будівлі.
- Вивід списку всіх приміщень в будівлі.
- Використання панорамних фотографій для подорожі будівлею.
- Можливість визначити своє місце розташування та вказати місце призначення.
- Можливість отримати інформацію про найкоротший шлях до вказаного приміщення.

1.2 Опис використаних технологій та середовищ розробки програм.

1.2.1 Flutter

Flutter є дуже привабливим рішенням для розробки мобільних додатків з багатофункціональним інтерфейсом. Деякі з причин, чому Flutter є хорошим вибором для розробки програмного забезпечення, включають:

1. Кросплатформність: Flutter дозволяє розробникам створювати мобільні додатки для Android та iOS за допомогою єдиного коду, що робить процес розробки більш ефективним та менш витратним.
2. Швидкість: Flutter використовує власний двигун для відображення графічних елементів, що робить його дуже швидким та ефективним. Це означає, що додатки, розроблені на Flutter, запускаються швидко та працюють із високою продуктивністю.
3. Гнучкість: Flutter дозволяє розробникам створювати різні типи додатків, включаючи мобільні додатки, веб-додатки та настільні додатки. Крім того, з Flutter можна створювати додатки з різними розмірами екрану та роздільною здатністю.
4. Відкритий код: Flutter є відкритим проектом, що означає, що розробники можуть внести свій внесок в проект, додавши нові функції та удосконалюючи наявні. Крім того, наявність великої спільноти розробників та розробників-фанатів також допомагає вирішувати проблеми та покращувати розробку додатків на Flutter.
5. Матеріальний дизайн: Flutter підтримує матеріальний дизайн від Google, що дозволяє розробникам швидко створювати красиві та зручні інтерфейси для своїх додатків.

1.2.2 Firebase Database

В якості бази даних було вирішено обрати Firebase Database. Це є хорошим рішенням для зберігання та синхронізації даних між додатками на різних пристроях та платформах. Деякі переваги Firebase Database включають:

1. Реальний час: Firebase Database підтримує режим реального часу, що дозволяє миттєво оновлювати дані на всіх підключених пристроях. Це дозволяє користувачам бачити оновлення даних миттєво, без необхідності оновлювати сторінку.
2. Простота використання: Firebase Database пропонує простий API для зберігання та отримання даних. Для розробників, які використовують Firebase, не потрібно писати складний SQL-код, що дозволяє прискорити процес розробки.
3. Масштабованість: Firebase Database підтримує масштабованість до будь-якої кількості користувачів та даних. Firebase може автоматично масштабуватись в залежності від навантаження, що дозволяє уникнути проблем з швидкістю та доступністю.
4. Безпека: Firebase Database пропонує безпечний доступ до даних, за допомогою прав доступу, які можуть бути настроєні на основі рівня доступу користувача.
5. Інтеграція: Firebase Database інтегрується з іншими сервісами Firebase, такими як Firebase Authentication, Firebase Storage та Firebase Cloud Messaging, що дозволяє створювати повноцінні додатки з використанням різних сервісів Firebase зі зручним API та інструментами.

1.2.3 Середовища програмування

В якості середовища програмування мобільного додатку було вирішено обрати Android Studio, тому що це є однією з найпопулярніших інтегрованих середовищ розробки для створення мобільних додатків на платформі Android. Ось кілька причин, чому Android Studio є хорошим рішенням для розробки мобільних додатків:

1. Інтегрованість: Android Studio містить всі необхідні інструменти для розробки, тестування, відлагодження та розгортання мобільних додатків на платформі Android. Це зменшує необхідність в користуванні різними інструментами та дозволяє розробникам працювати зручно та ефективно.
2. Інтеграція з Google: Android Studio підтримує інтеграцію з різними сервісами Google, такими як Google Cloud Platform, Firebase та Google Play. Це дозволяє розробникам швидко та легко використовувати різні сервіси Google для своїх проектів.
3. Відлагодження: Android Studio має різні інструменти для відлагодження, які дозволяють розробникам виявляти та виправляти помилки у своєму коді. Це допомагає покращити якість програмного забезпечення та зменшує кількість помилок, що зустрічаються користувачами.
4. Розширюваність: Android Studio має велику кількість плагінів, які дозволяють розширити функціональність IDE та забезпечити більш зручний досвід розробки.

Для розробки на Flutter в середовищі програмування Android Studio потрібно завантажити Flutter SDK[1]. Для цього можна завантажити zip-архів з сайту розробника або скопувати Github-репозиторій[2].

В якості середовища програмування програми для стиснення зображень було обрано Visual Studio.

1.2.4 Blender

Для моделювання поверхів будівлі було вирішено обрати Blender, тому що це потужним інструментом для моделювання 3D об'єктів, який використовується для створення детальних моделей плану будівлі. Blender має вбудовану бібліотеку засобів моделювання, включаючи інструменти для створення 3D форм, об'ємних тіл та текстур. Крім того, Blender підтримує імпорт і експорт файлів у багатьох форматах, що дає можливість обмінюватися моделями з іншими програмами для проектування будівель.

Blender також має функції рендерингу, які дозволяють створювати реалістичні візуалізації будівельного проекту, використовуючи різні освітлення та матеріали.

Крім того, Blender є відкритою і безкоштовною програмою з відкритим вихідним кодом, що робить її доступною для використання широкому колу користувачів. Blender має велику спільноту користувачів та розробників, які активно працюють над розвитком програми, розширюючи її можливості та функціонал.

1.2.5 .NET і WPF

Програма для стиснення зображень повинна працювати на операційній системі Windows. Для розробки такої програми ідеально підходять .NET і WPF технології.

.NET і WPF є дуже потужними інструментами для розробки невеликих десктопних програм. .NET Framework - це платформа розробки програмного забезпечення, що працює на різних операційних системах, включаючи Windows, macOS і Linux. WPF (Windows Presentation Foundation) є частиною .NET Framework і надає потужні інструменти для розробки віконних додатків з багатим графічним інтерфейсом користувача.

.NET Framework має широкий вибір стандартних класів та бібліотек, які забезпечують велику кількість функцій, що можуть бути використані в додатку.

WPF надає гнучкі можливості для розробки інтерфейсу користувача, дозволяючи розробникам створювати багатофункціональні вікна з високоякісними графічними ефектами. WPF також має дуже потужну систему прив'язки даних, що дозволяє зручно та ефективно працювати з даними в програмі.

1.2.6 Adobe Photoshop CC

В роботі над дизайном програми було вирішено використовувати Adobe Photoshop, тому що він є одним з найбільш популярних графічних редакторів на ринку. Цей редактор має широкий набір інструментів і функцій, які дозволяють створювати різноманітні графічні елементи, зокрема елементи дизайну програмного інтерфейсу.

За допомогою Adobe Photoshop можна створювати та редагувати різноманітні елементи дизайну, такі як іконки, кнопки, фони, логотипи та інші елементи програмного інтерфейсу. Він має широкі можливості роботи з кольорами, шрифтами, текстурами та іншими елементами дизайну.

Крім того, Adobe Photoshop має вбудовану систему шарів, яка дозволяє працювати з кожним елементом дизайну окремо та змінювати їх пізніше, не впливаючи на інші елементи. Це дозволяє зберігати організованість і зручність роботи над дизайном програмного інтерфейсу.

2. ПРОЕКТУВАННЯ ТА ПРОГРАМНА РЕАЛІЗАЦІЯ ДОДАТКУ ВІДПОВІДНО ДО ВИМОГ

Під час розробки даного додатку було використано вищезгадані технології: мова програмування Dart[8], середовище програмування Android Studio і Visual Studio, фреймворк Flutter, веб-сервіс контролю версій GitHub[9], програмний пакет для створення тривимірної комп'ютерної графіки Blender, графічний редактор Adobe Photoshop CC. Усі вони були описані у попередніх розділах.

2.1 Загальний опис та характеристика продукту.

Даний продукт належить до ПЗ, що повинне забезпечувати ефективний пошук приміщення в будівлі. Даний додаток був створений для вирішення наступного ряду вимог:

- Вивід списку всіх приміщень в будівлі.
- Можливість вільного пересування по віртуальній карті будівлі.
- Можливість визначити своє місце розташування та вказати місце призначення.
- Можливість отримати інформацію про найкоротший шлях між вказаними місцями.

Поведінка системи представлена на малюнку 2.1.

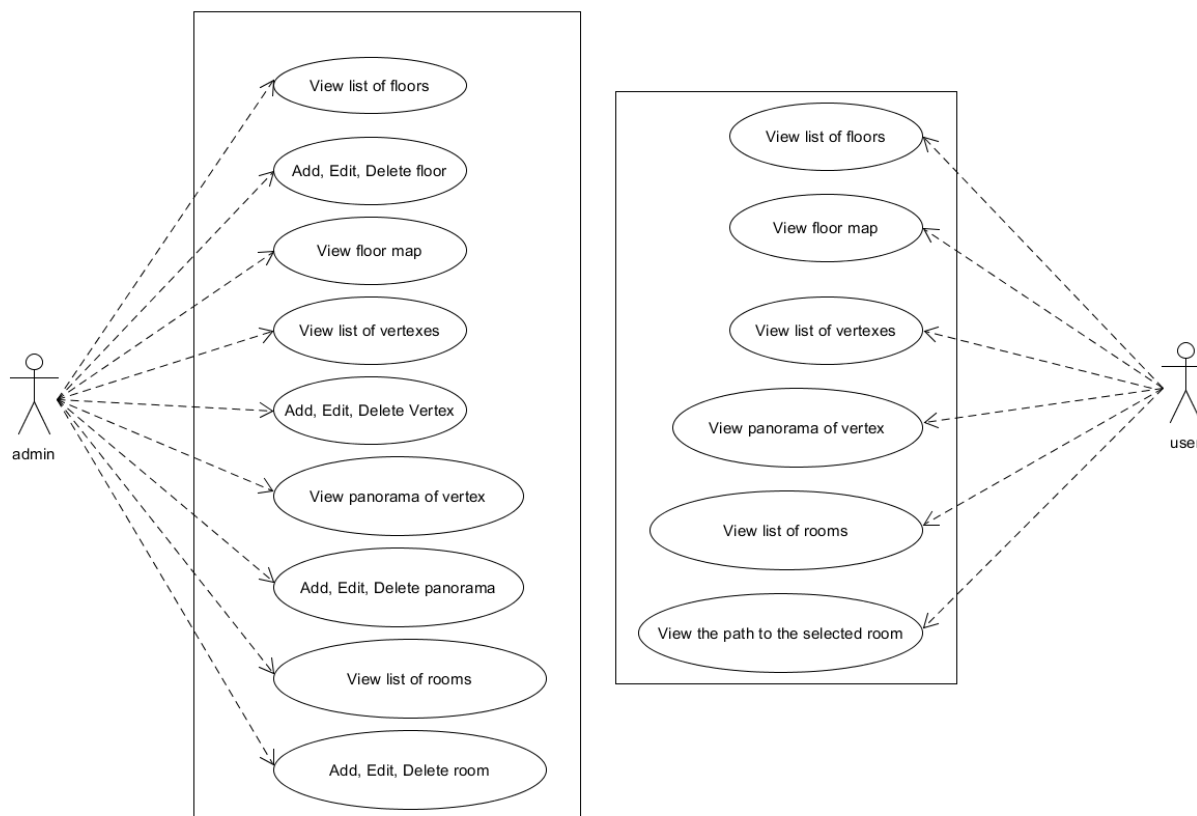


Рис. 2.1 UML-діаграма випадків використання системи.

Користувачу доступний список поверхів з двовимірними картами для кожного поверху будинку. Кожна карта містить вершини розташовані біля приміщень для їх ідентифікації. Вибравши приміщення йому буде виведено на екран панорамна фотографія найближчої вершини біля цього приміщення. Використовуючи віджети на екрані смартфона можна переміщатись по вершинах у вільному напрямку. Також, вибравши дві вершини або приміщення користувач може побудувати найкоротший шлях між ними, який буде зображений на графі.

Адміністратору доступний список поверхів, а також функціонал для редагування карти кожного поверху. Він може додавати вершини та приміщення, а також редагувати інформацію про них.

2.2 Побудова карти поверхів

Для побудови карти поверху використовується програмний пакет для створення тривимірної комп'ютерної графіки Blender. 3D модель поверху будинку була успішно розроблена за наступними етапами:

1. Планування та налаштування: перед розпочатком роботи був створений загальний план та встановлені налаштування для роботи з 3D моделями.
2. Створення основи моделі: була створена основа моделі, включаючи стіни та фундамент за допомогою інструментів моделювання.
3. Додавання деталей: були додані деталі, такі як двері, сходи та інші елементи, що доповнюють будівлю. Для цього використовувалися інструменти моделювання, такі як "Boolean" (логічне об'єднання).
4. Робота з текстурами: були додані текстури на поверхні будівлі за допомогою інструментів текстурювання, таких як "UV Mapping" (картування UV) та "Texture Paint" (фарбування текстур).
5. Освітлення: було додано освітлення до моделі, використовуючи різні типи джерел світла та колірні фільтри.

Після створення 3D моделі в Blender використано функцію рендерингу для створення 2D зображення з цієї моделі. Результатом цього процесу була 2D карта, яка відображає поверхню будинку з високим ступенем деталізації. На рисунку 2.2 зображено приклад 3D моделі першого поверху головного корпусу Львівського національного університету імені Івана Франка.

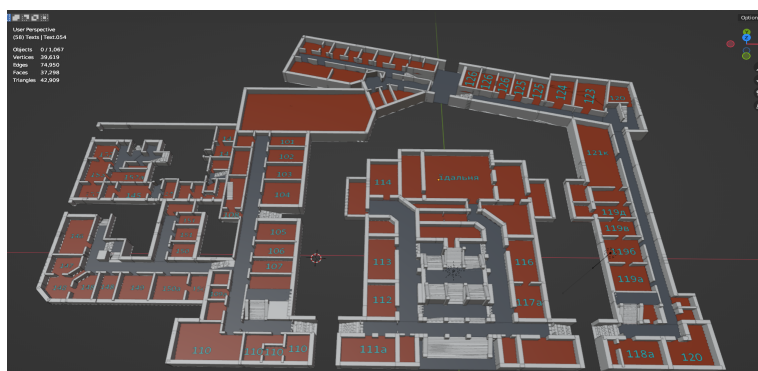


Рис. 2.2 3D модель першого поверху головного корпусу Львівського національного університету імені Івана Франка.

2.3 Загальна архітектура проекту

На рис. 2.3 зображена архітектура проекту

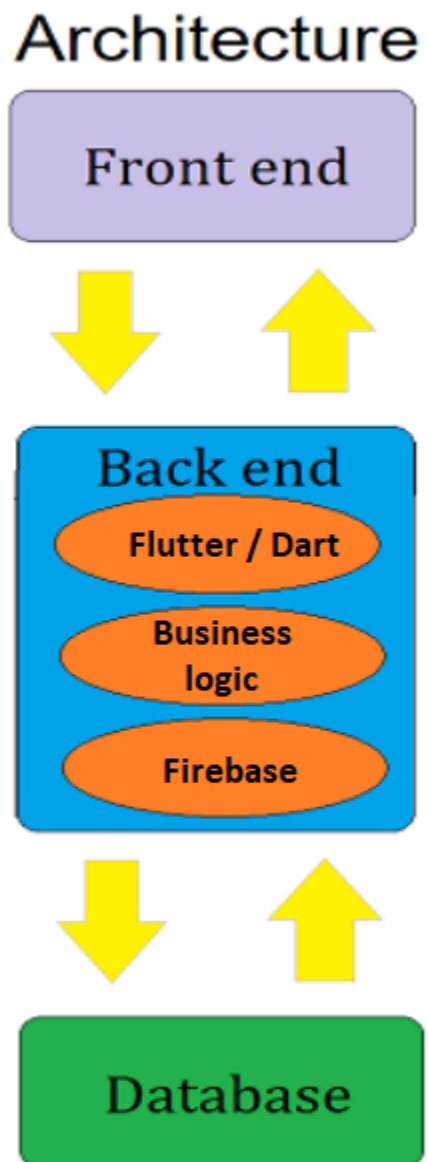


Рис. 2.3 Архітектура програми.

Ми маємо зручний користувацький інтерфейс на FrontEnd частині. На BackEnd частині архітектура поділена на 3 рівні:

- Flutter/Dart - це рівень, на якому знаходиться користувацька частина додатку, яка відповідає за передачу та прийом інформації.
- Business logic - на цьому рівні обробляється вся логіка, пов'язана з обробкою даних, коли вони переходять до користувача та коли зберігаються в сховищі.
- Firebase - цей рівень допомагає звертатися до бази даних для зберігання та отримання даних.

Дані програми зберігаються в сховищі даних, а також там є функції та процедури, які допомагають працювати з ними.

2.4 База даних проекту

Схема бази даних представлена на малюнку (рис.2.4).

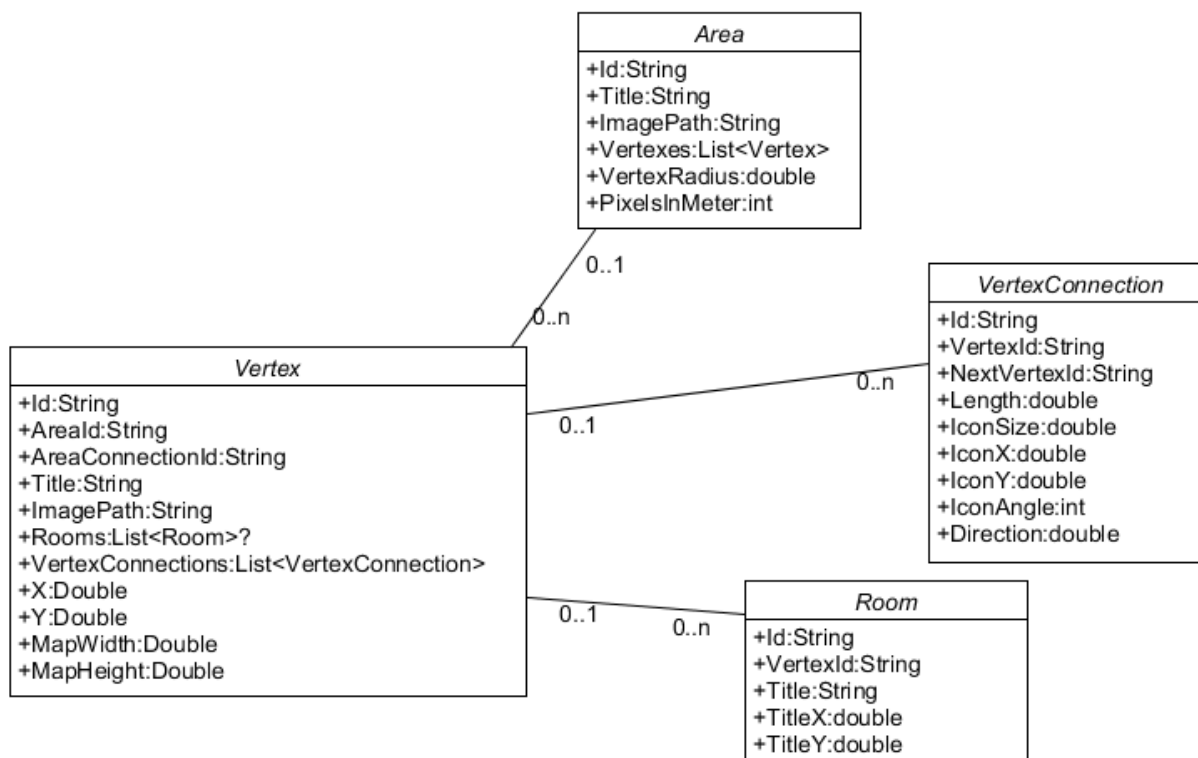


Рис. 2.4. Схема бази даних

Модель Area являється моделлю поверху або окремої частини будівлі, яка в собі містить список вершин. Кожна вершина містить список приміщень, які знаходяться близько до неї. Також кожна вершина має список VertexConnections в якому знаходяться моделі, які представляють зв'язок вершини з сусідніми вершинами.

2.5 Стиснення зображень

Кожна вершина графа містить панорамну фотографію місця розташування. Оскільки вона створюється у великому розширенні, тому задля економії пам'яті, трафіку і в результаті збільшення швидкодії додатку, кожне фото потрібно стиснути.

Для стискання зображення було розроблено окрему програму на мові програмування C# з використанням технологій .NET та WPF, яка дозволяє швидко конвертувати велику кількість фотографій за заданими параметрами розширення.

Провівши відповідні тестування, оптимальним розширенням ширини і висоти вважається 3500 і 1750 пікселів відповідно. Інтерфейс програми зображено на рисунку 2.5.

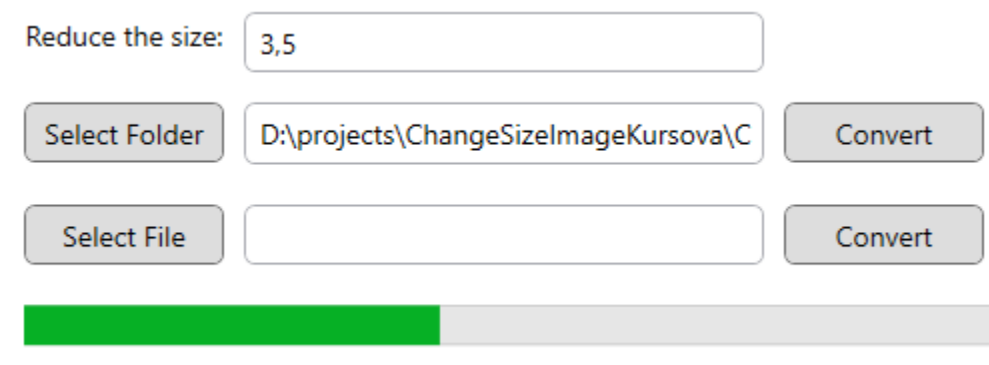


Рис. 2.5. Програма для стиснення зображень

2.6 Особливості реалізації

Концепція програми базується на ідеї, що будівля є великим графом з вагами, де вершини є важливими точками, такими як приміщення, повороти та перешкоди. Розташування вершин важливо забезпечити щільно, щоб кожна вершина була видима всім своїм сусідам. Між точками встановлюється приблизна відстань, яка використовується для пошуку найкоротшого шляху за допомогою алгоритму. Кожна вершина містить панорамну фотографію, яку можна зробити з допомогою будь-якого смартфона та безкоштовного додатку Google Street View. Хоча процес створення панорамних фотографій не є швидким і може займати близько 3-5 хвилин на кожную фотографію, він є більш доступним та економічним, ніж використання спеціальних панорамних фотоапаратів.

Після зйомки панорамних фотографій усіх вершин, вони мають бути стиснені за допомогою спеціальної програми, а потім відсортовані за папками для зручного доступу та подальшого редагування. Щоб мати онлайн доступ до всіх фотографій через програму, їх необхідно зберегти на хмарному носії.

2.7 Алгоритм пошуку

Алгоритм Дейкстри є ефективним рішенням для пошуку найкоротшого шляху в орієнтованому графі з невід'ємними вагами ребер. Онлайн-магазини, картографічні сервіси та маршрутизатори часто використовують цей алгоритм для побудови маршрутів та планування доставок.

Основна перевага алгоритму полягає в тому, що він забезпечує точний результат. Він відшукує найкоротший шлях між двома точками, навіть якщо граф має велику кількість вершин та ребер. Крім того, алгоритм Дейкстри забезпечує оптимальну ефективність, оскільки він не перебирає всі можливі шляхи, а шукає найкоротший шлях, спочатку від вершини старту до сусідніх вершин, потім до сусідніх вершин вже знайдених шляхів, і так далі.

Алгоритм працює шляхом поступового розширення множини відвіданих вершин. На кожному кроці він вибирає вершину з найменшою вагою та додає її до множини відвіданих. Далі, для кожної непройденої вершини, він оновлює найкоротший шлях до неї, якщо шлях через поточну вершину є коротшим за попередній найкоротший шлях.

Алгоритм завершується, коли всі вершини будуть відвідані або коли неможливо досягти бажаної вершини.

Алгоритм Дейкстри дозволяє знайти найкоротші шляхи в орієнтованому графі з невід'ємними вагами ребер за час, пропорційний кількості вершин та ребер графа. Це робить його ефективним інструментом для багатьох задач, таких як маршрутизація мережі, планування маршруту для роботів та автомобілів, пошук шляху в маршрутних листах та багато інших.

3. СЦЕНАРІЙ РОБОТИ МОБІЛЬНОГО ДОДАТКУ

Увійшовши в програму користувача зустріне стартова сторінка з логотипом та, за деякий час, сторінка зі списком поверхів. Після обрання поверху відкривається сторінка з моделлю на якій зображені вершини(рис. 3.1). Кожна точка на екрані являє собою вершину, з'єднану зі своїми сусідами. Після вибору початкової точки і визначення кінцевої на карті лишаються вершини, що утворюють шлях

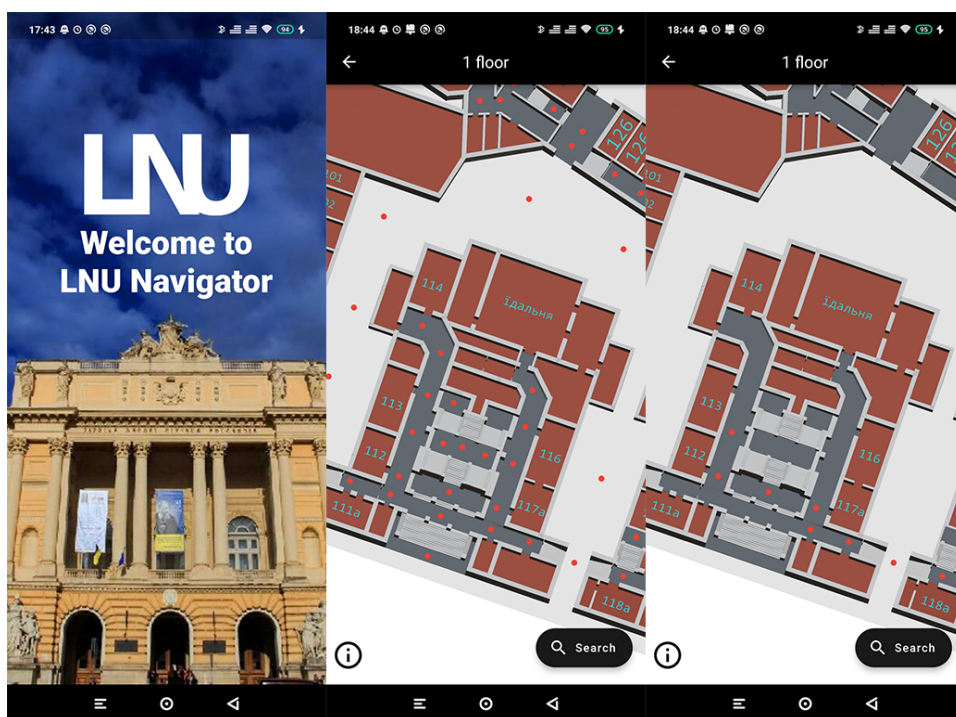


Рис. 3.1. Загрузка програми та сторінка зі списоко будівель

Натиснувши на кнопку пошуку шляху, ми потрапляємо на сторінку з вибором поточного місця і місця призначення, а також на сторінку з вибором конкретного приміщення(рис. 3.2)

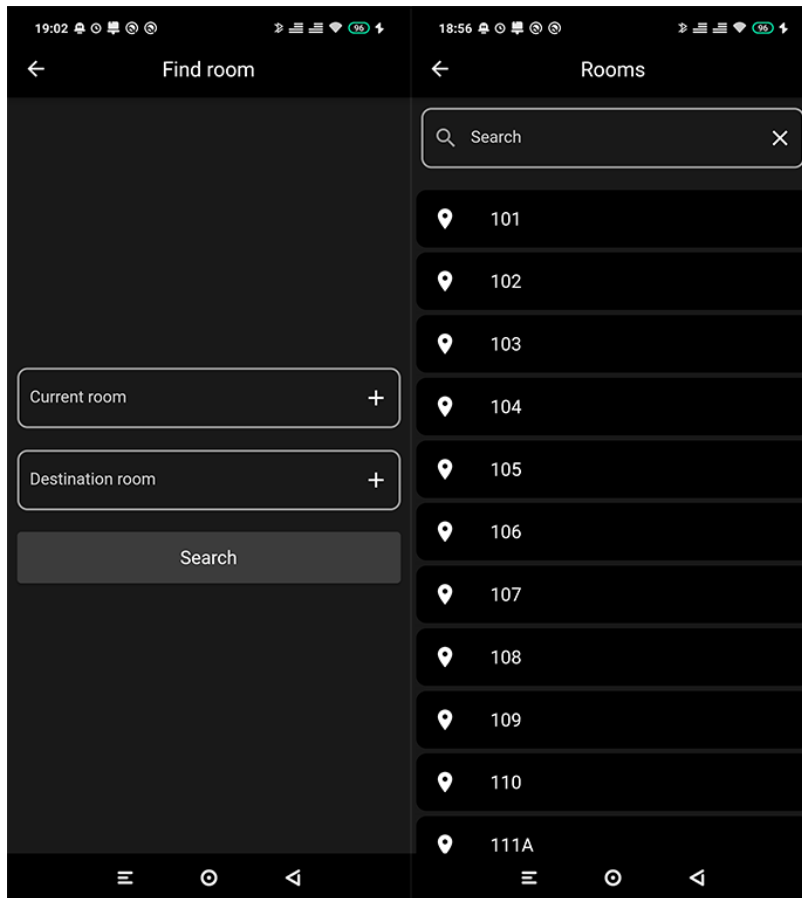


Рис. 3.2. Сторінки з вибором приміщень

Натиснувши на одну з вершин користувачу відкривається сторінка з відображенням панорамної фотографії, на якій виставлені орієнтири (рис. 3.4). Кожне приміщення підписане червоним шрифтом, а на шляху виставлені зелені іконки у напрямку наступної найближчої до вибраного приміщення контрольної точки. Натиснувши на іконку користувача відкривається наступна панорамна фотографія на цьому ж місці з вказаною наступною контрольною точкою, в кінці шляху програма виведе останню фотографію з відображенням потрібних дверей приміщення.

Якщо користувач не вкаже точки призначення, тоді натиснувши на вершину відкриється панорамна фотографія з усіма визначеними точками на ній, дозволяючи віртуально ознайомитись з будівлею.



Рис. 3.4. Сторінки з відображенням панорамних фотографій і контрольних точок

Процес побудови графа відбувається на сторінці адміністратора (рис. 3.5). Адмін виставляє точки на картинці, які являються вершинами. Зв'язки між вершинами позначаються червоними лініями для зручності в тестуванні.

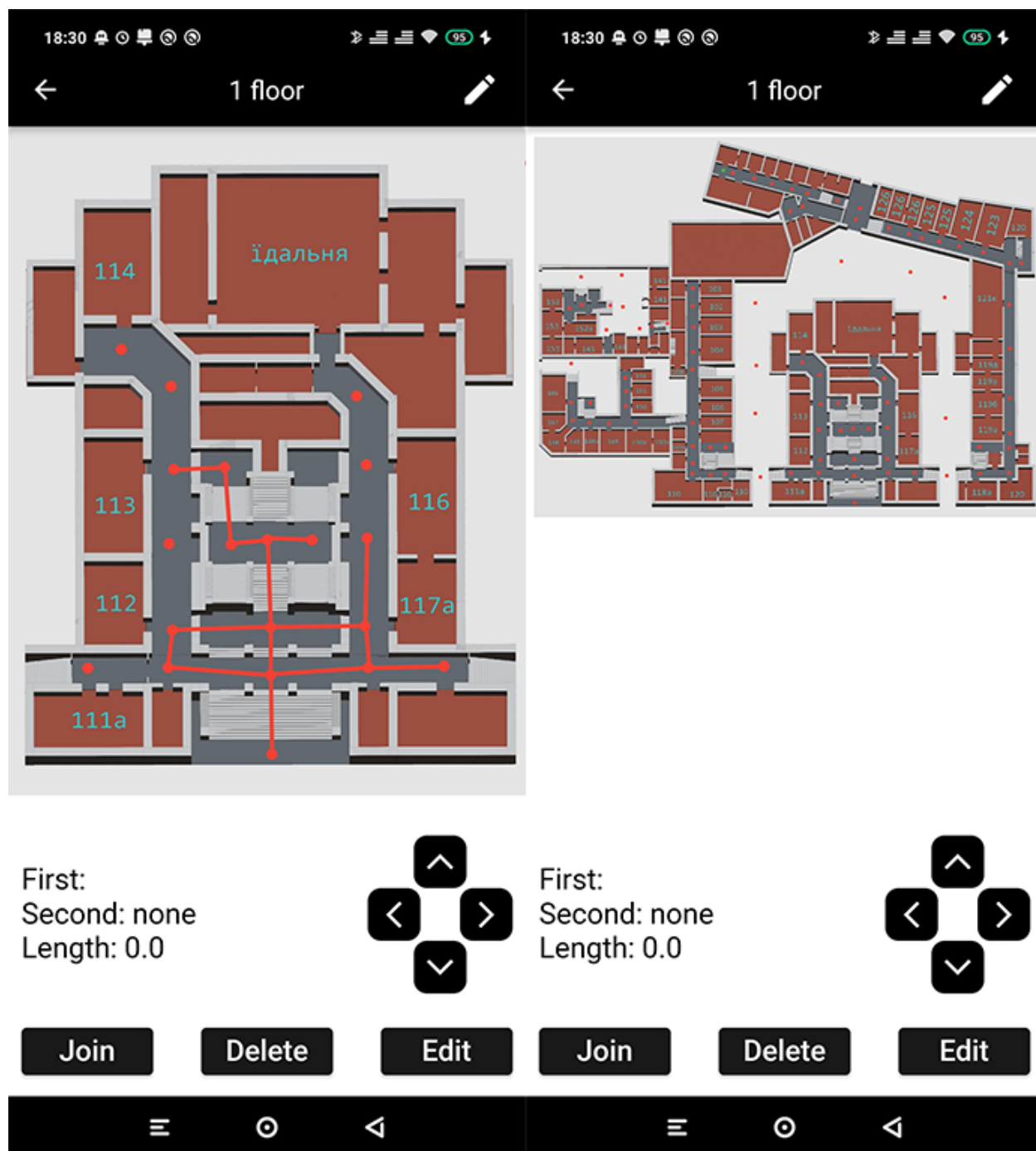


Рис. 3.5. Заповнення даних на сторінці адміністратора

ВИСНОВКИ

Дана робота присвячена розробці додатку для навігації по будівлі. Під час написання даної програми було використано чимало новітніх технологій (Flutter, Firebase, .NET, WPF). В результаті розроблено програму, яка дозволяє, використовуючи двовимірні карти та панорамні фотографії, орієнтуватися в незнайомій будівлі. Кожна двовимірна карта являє собою реальну частину моделі будинку, а панорамна фотографія на кожній вершині дозволяє орієнтуватися в приміщенні співставляючи реальні об'єкти навколо. Також, на кожній панорамній картинці вказані орієнтири які допомагають досягти заданого користувачем приміщення.

СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ ТА ІНТЕРНЕТ РЕСУРСІВ

1. Install - Flutter : [Electronic resource] – Available from : <https://flutter.dev/docs/get-started/install>
2. flutter/flutter: Flutter makes it easy and fast to build beautiful apps for mobile and beyond. : [Electronic resource] – Available from : <https://github.com/flutter/flutter>
3. Set up an editor – Flutter : [Electronic resource] – Available from : <https://flutter.dev/docs/get-started/editor?tab=androidstudio>
4. *Кормен, Т. Г., Лейзерсон Ч. Е., Рівест Р. Л, Стайн Кліфорд Вступ до алгоритмів. Видавництво <<К. І. С.>>. 2008. С. 144.*
5. *Кормен Т. Г Алгоритми доступно. Видавництво<<К.І.С>>. – 2013. – Вип. 20. – С. 85–94.*
6. Street View app : [Electronic resource] – Available from : <https://www.google.com/streetview/>
7. Dart overview | Dart : [Electronic resource] – Available from : <https://dart.dev/overview>
8. Github : [Electronic resource] – Available from : <https://github.com/>
9. Adobe Photoshop CC : [Electronic resource] – Available from: <https://adobe-photoshop.en.softonic.com/>