

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
НАЦІОНАЛЬНИЙ АВІАЦІЙНИЙ УНІВЕРСИТЕТ
ФАКУЛЬТЕТ КОМП'ЮТЕРНИХ НАУК ТА ТЕХНОЛОГІЙ

Кафедра Комп'ютерних інформаційних технологій

ДОПУСТИТИ ДО ЗАХИСТУ

Завідувач кафедри

Аліна САВЧЕНКО

«_____» _____ 2023 р.

КВАЛІФІКАЦІЙНА РОБОТА
(ДИПЛОМНИЙ ПРОЄКТ, ПОЯСНЮВАЛЬНА ЗАПИСКА)
ВИПУСКНИКА ОСВІТНЬОГО СТУПЕНЯ “БАКАЛАВР”
ЗА ОСВІТНЬО-ПРОФЕСІЙНОЮ ПРОГРАМОЮ
“ІНФОРМАЦІЙНІ УПРАВЛЯЮЧІ СИСТЕМИ ТА ТЕХНОЛОГІЇ”

Тема: «Мобільний Android-додаток для планування подорожі на мові Java»

Виконавець: студент групи УС-412Б Самойлов Олександр Миколайович

Керівник: _____ ст. викладач Харченко Олександр Григорович

Нормоконтролер: _____ Олександр ШЕВЧЕНКО

Київ – 2023

НАЦІОНАЛЬНИЙ АВІАЦІЙНИЙ УНІВЕРСИТЕТ

Факультет комп'ютерних наук та технологій

Кафедра Комп'ютерних інформаційних технологій

Галузь знань, спеціальність, освітньо-професійна програма: 12 “Інформаційні технології”, 122 “Комп'ютерні науки”, “Інформаційні управляючі системи та технології”

ЗАТВЕРДЖУЮ

Завідувач випускової кафедри

_____ Аліна САВЧЕНКО

«_____» _____ 2023р.

ЗАВДАННЯ

на виконання кваліфікаційної роботи студента

Самойлова Олександра Миколайовича

(прізвище, ім'я, по батькові)

- 1. Тема роботи:** «Мобільний Android-додаток для планування подорожі на мові Java» затверджена наказом ректора від 01.05.2023р. № 623/ст.
- 2. Термін виконання роботи:** з 15.05.2023р. по 25.06.2023 р.
- 3. Вихідні дані до роботи:** розробка Android-додатку для подорожі по визначним місцям України.
- 4. Зміст пояснювальної записки:** вступ, аналітичний огляд і постановка завдання, вибір інтегрованого середовища розробки, дослідження технологій та засобів, розробка додатку, аналіз виконаного, висновки.
- 5. Перелік обов'язкового графічного матеріалу:** скріншоти.

6. Календарний план-графік

<i>№ з/п</i>	<i>Завдання</i>	<i>Термін виконання</i>	<i>Підпис керівника</i>
1.	Огляд та аналіз предметної області	15.05.2023– 17.05.2023	
2	Збір, аналіз і опрацювання інформації за тематикою кваліфікаційної роботи	18.05.2023– 21.05.2023	
3	Вивчення інтегрованого середовища розробки Android Studio	22.05.2023- 25.05.2023	
4	Аналіз вимог до Android-додатку	27.05.2023– 29.05.2023	
5	Розроблення і тестування Android-додатку	30.05.2023– 02.05.2023	
6	Написання пояснювальної записки кваліфікаційного проекту	02.05.2023– 03.05.2023	
7	Подання готового проекту	03.06.2023– 22.06.2023	

7. Дата видачі завдання: 15 травня 2023 р.

Керівник дипломного проекту _____ Олександр ХАРЧЕНКО
(підпис керівника) (П.І.Б.)

Завдання прийняв до виконання _____ Олександр САМОЙЛОВ
(підпис випускника) (П.І.Б.)

РЕФЕРАТ

Пояснювальна записка до кваліфікаційної роботи «Мобільний Android-додаток для планування подорожі на мові Java» викладена на 69 сторінках, містить 42 рисунки та 10 літературно-наукових джерел.

Об'єкт дослідження: мобільний Android-додаток, створений для планування подорожей.

Предмет дослідження: розробка та реалізація мобільного Android-додатку, який допомагає користувачам планувати подорожі.

Мета кваліфікаційної роботи: розробка та реалізація мобільного Android-додатку для планування подорожей на мові Java. Головним завданням роботи є створення зручного та функціонального додатку, який надасть користувачам засоби для ефективного планування та організації своїх подорожей.

Метод дослідження: у даній роботі буде використано комбінацію теоретичних та практичних методів дослідження. Серед теоретичних методів будуть використовуватись аналіз наукової літератури, огляд існуючих аналогічних додатків та аналіз відгуків користувачів. Практичні методи будуть включати розробку та реалізацію мобільного додатку з використанням мови програмування Java та фреймворку для розробки Android-додатків.

Результат проекту: після завершення проекту, користувачі зможуть зручно та ефективно планувати свої подорожі за допомогою розробленого додатку.

АРХІТЕКТУРА ANDROID-ДОДАТКУ, МОВА ПРОГРАМУВАННЯ JAVA, МОДЕЛЮВАННЯ ТА АНАЛІЗ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ, ТЕСТУВАННЯ ЗРУЧНОСТІ ІНТЕРФЕЙСУ, АНАЛІЗ БЕЗПЕКИ ДАНИХ.

ЗМІСТ

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ ТА СКОРОЧЕНЬ.....	7
ВСТУП.....	8
РОЗДІЛ 1. ПРОБЛЕМИ РОЗРОБКИ ANDROID-ДОДАТКІВ ТА ВИМОГИ ДО НИХ	10
1.1. Загальні положення	10
1.2. Опис і аналіз предметної області.....	11
1.3. Методи розробки мобільних додатків.....	15
1.4. Формування та постановка задач розробляемого мобільного додатку	16
1.5. Аналіз вимог до програмного забезпечення	17
1.5.1. Розроблення функціональних вимог	18
1.5.2. Розроблення не функціональних вимог	18
1.6. Аналіз відомих програмних продуктів.....	18
1.6.1. Мобільний застосунок «Triplt»	19
1.6.2. Мобільний застосунок «PackPoint»	20
1.6.3. Мобільний застосунок «Booking»	22
1.6.4. Мобільний застосунок «Airbnb»	24
1.6.5. Мобільний застосунок «Couchsurfing».....	25
1.7. Висновок до 1 розділу	27
РОЗДІЛ 2. АНАЛІЗ ТА ПРОЕКТУВАННЯ.....	29
2.1. Проектування архітектури додатку	29
2.2. Вибір мови програмування та середовища розробки.....	30
2.3. Розробка інтерфейсу користувача	31
2.4. Висновок до 2 розділу	34
РОЗДІЛ 3. РЕАЛІЗАЦІЯ ТА ТЕСТУВАННЯ.....	35
3.1. Розробка компонентів додатку	35
3.1.1. Реєстрація / авторизація у додатку	38

3.1.2. Головна активність додатку	41
3.1.3. Запис задач до подорожі	44
3.1.4. Запрошення до групової подорожі	48
3.2. Підключення сервісів	50
3.2.1. Google maps	50
3.2.2. База даних	53
3.3. Збірка і встановлення	55
3.3.1. Створення арк-файлу.....	55
3.3.2. Завантаження додатку на смартфон	55
3.3.3. Робота додатку	57
3.4. Висновок до 3 розділу	67
ВИСНОВКИ	68
СПИСОК БІБЛІОГРАФІЧНИХ ПОСИЛАНЬ ВИКОРИСТАНИХ ДЖЕРЕЛ.	70

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ ТА СКОРОЧЕНЬ

БД	База даних
ПЗ	Програмне забезпечення
API	Інтерфейс програмування додатків
CRUD	Створення, читання, оновлення, видалення
GPS	Глобальна система позиціонування
HTTPS	Захищений протокол передачі гіпертексту
JSON	Формат обміну даними
MVC	Модель-представлення-контролер
SDK	Набір розробничих інструментів
SQL	Структурована мова запитів
UI	Інтерфейс користувача
UX	Користувацький досвід

ВСТУП

Кваліфікаційна робота розроблена для представлення Android-додатку, ціль якого полегшити мандрівникам подорож.

Мета роботи - розробка і реалізація мобільного додатку на платформі Android, який дозволить користувачам планувати свої подорожі зручним та ефективним способом. Додаток буде надавати широкий набір функцій, включаючи вибір та бронювання готелів, пошук та планування маршрутів, отримання інформації про визначні місця та події, а також керування витратами та розкладами.

Ідея роботи - зручного та універсального інструменту для подорожуючих, щоб допомогти їм ефективно організовувати свої подорожі та отримувати максимальне задоволення від них. Додаток буде орієнтований на широку аудиторію користувачів, незалежно від їхнього досвіду у плануванні подорожей.

Об'єктом дослідження - мобільний Android-додаток для планування подорожей.

Предмет дослідження - розробка та реалізація функціоналу, який відповідає потребам користувачів у плануванні та управлінні подорожами.

Методи дослідження - аналіз існуючих подібних додатків, проектування та розробка програмного забезпечення, тестування та оцінка результатів.

Результат проекту - готовий мобільний Android-додаток, який забезпечить користувачів зручним і ефективним інструментом для планування та керування своїми подорожами.

Додаток буде протестований та оцінений з точки зору його функціональності, надійності та користувальницького досвіду.

Крім того, розробка мобільного Android-додатку для планування подорожей на мові Java є актуальною у зв'язку зі зростанням популярності мобільних пристроїв та їх впливом на наше повсякденне життя. Люди все більше використовують смартфони та планшети для здійснення поїздок та організації своєї подорожі.

РОЗДІЛ 1

ПРОБЛЕМИ РОЗРОБКИ ANDROID-ДОДАТКІВ ТА ВИМОГИ ДО НИХ

1.1. Загальні положення

У сучасному світі мобільні пристрої здобули велику популярність, і користувачі все більше часу проводять на своїх смартфонах. Це відкриває нові можливості для розробки мобільних додатків, які забезпечують широкий спектр функцій і сервісів.

Особливо важливою галуззю, де мобільні додатки мають великий потенціал, є економічне управління. Використання мобільних пристроїв у сфері економічного управління може значно покращити якість розрахунків і зробити процес прийняття економічних рішень більш ефективним. Мобільні засоби надають зручний доступ до потрібної інформації, що допомагає економістам збирати, обробляти та аналізувати дані.

Оскільки компанії, такі як Google, Apple і Microsoft, розробляють спеціалізовані мобільні платформи та засоби розроблення, розробка мобільних додатків стає все більш актуальним напрямком в ІТ-індустрії. Однак, розробка мобільних додатків вимагає певних компетенцій у сфері об'єктно-орієнтованого програмування, знань SQL для роботи з базами даних, дизайну користувацького інтерфейсу, розуміння мережевої взаємодії та тестування програмного забезпечення.

Кафедра КІТ				НАУ 23 35 41 000 ПЗ			
Виконавець	Самойлов О.М.			ПРОБЛЕМИ РОЗРОБКИ ANDROID-ДОДАТКІВ ТА ВИМОГИ ДО НИХ	Літера	Аркуш	Аркушів
Керівник	Харченко О.Г.					10	19
Консультант							
Н.Контроль	Шевченко О.П.						
					УС-412Б	122	

Отже, розробка мобільних додатків на мові Java для планування подорожей є актуальною і важливою задачею, оскільки вона задовольняє потреби користувачів у зручному і ефективному способі планування та організації своїх подорожей.

За допомогою додатку у користувача має з'явитися можливість використовувати додаток помічник для мандрівників для того щоб дізнаватися про всі неprimітні пам'ятки України, додавати до обраного та ставити статус переглянутого щоб у пошуковій системі не було тієї статті котра була вже переглянута також є можливість прокласти шлях до них.

Тепер мандрівник не повинен шукати, які пам'ятки існують в Україні і шукати їх на різних сайтах, тому що від тепер користувач матиме можливість шукати все в одному додатку з простим та доступним інтерфейсом. Додаток буде створено на платформі Android.

1.2. Опис і аналіз предметної області

Подорожі стали невід'ємною частиною нашого життя, і все більше людей користуються мобільними додатками для планування своїх подорожей. Ця дипломна робота спрямована на створення мобільного Android-додатку, який надає користувачам зручний і ефективний інструмент для планування подорожей.

Основна мета додатку полягає в наданні користувачам зручного способу планування і організації їх подорожей. Додаток повинен забезпечувати такі основні функціональні можливості:

1. Реєстрація та авторизація користувачів: Додаток повинен дозволяти користувачам створювати облікові записи та увійти в систему для збереження та синхронізації своїх подорожей на різних пристроях.

2. Створення подорожей: Користувачі повинні мати можливість створювати нові подорожі, надавати їм назви та опис, вибрати дати початку та закінчення, а також додавати місця, які вони планують відвідати.

3. Планування маршруту: Додаток повинен надати функцію планування маршруту, що допоможе користувачам ефективно організувати свої маршрути та визначити найкоротші шляхи між місцями, які вони планують відвідати. Це може включати інтеграцію з картами та навігаційними сервісами.

4. Збереження інформації про місця: Додаток повинен дозволяти користувачам зберігати інформацію про місця, які вони планують відвідати під час подорожі, такі як назви, адреси, контактні дані, рейтинги та коментарі. Користувачі повинні мати можливість переглядати цю інформацію та додавати нові місця до своїх подорожей.

5. Нагадування та повідомлення: Додаток може надавати можливість створення нагадувань про подорожі та повідомляти користувачів про заплановані дати початку та закінчення подорожі.

Аналіз предметної області

Передбачає огляд основних аспектів, які пов'язані з даною темою. Декілька ключових елементів, які варто розглянути в аналізі, включають наступне:

1. Функціональність додатку: дослідження основних функцій та можливості, які потенційно можуть бути включені в додаток для планування подорожі. Це може включати функції, такі як пошук та бронювання готелів, пошук рейсів, складання маршруту, нагадування про події, зберігання документів та інші.

2. Інтеграція зі зовнішніми сервісами: можливість інтеграції додатку з популярними зовнішніми сервісами, такими як карти Google, системи бронювання готелів або авіаквитків, погодні сервіси тощо. Це дозволить користувачам отримувати актуальну інформацію та забезпечувати зручний досвід планування подорожі.

3. Маршрутизація та навігація: можливості для розробки алгоритмів маршрутизації та навігації, які допоможуть користувачам скласти оптимальні маршрути для своїх подорожей. Врахуйте фактори, такі як

відстань, час подорожі, вартість, транспортні засоби та інші параметри, які можуть впливати на вибір маршруту.

4. Управління витратами: можливості для включення функцій, які дозволяють користувачам вести облік витрат під час подорожі. Це може включати стеження за витратами на транспорт, проживання, харчування, розваги та інші витрати, що пов'язані з подорожжю. Такий підхід дозволить користувачам керувати своїми фінансами під час подорожі.

5. Планування подій та активностей: можливості для включення функцій, які дозволяють користувачам створювати плани та пам'ятки для подорожі. Це може включати нагадування про бронювання, розклад подій, список необхідних речей та інші корисні деталі, що допоможуть забезпечити організованість та комфорт під час подорожі.

6. Безпека та конфіденційність: важливість забезпечення безпеки та конфіденційності в додатку.

Програмою для розробки додатку було використано середовище «Android Studio», середовище розробки адаптоване для виконання типових завдань, що вирішуються в процесі розробки застосунків для платформи Android. Утому числі у середовище включені засоби для спрощення тестування програм на сумісність з різними версіями платформи та інструменти для проектування застосунків, що працюють на пристроях з екранами різної діяльності.

Особливості цієї платформи:

- Живі макети;
- Консоль розробника;
- Базування на Gradle;
- Android-орієнтований рефакторинг та швидкі виправлення;
- Шаблони для створення поширених Android видів дизайну.

Коли було обрано створення додатку на платформі Android та обрано середовище «Android Studio», стало питання вибору мови розробки додатку між Java, C/C++ та Kotlin.

Зовсім недавно Kotlin офіційно став мовою програмування для платформи Android, тому дехто навіть припускає, що він може стати чимось на зразок мови Swift, але для Android.

Відмінність Kotlin полягає в тому що, вимагає набагато менше шаблонного коду «boilerplate code», тому його синтаксис читається легше. Також, на відміну від Java, Kotlin null-безпечний, тобто при спробі привласнення або повернення nullкод не скомпілюється. Також, Kotlin сумісний з Java 1.6 це дуже важливий момент, адже саме ця версія Java використовується у всіх сучасних версіях Android та не дивлячись на запланований перехід OpenJDK, восьма версія потрапить в руки розробникам під мобільні пристрої не так скоро як хотілося. Звичайно є ретролямбди і інші хитрощі, але Kotlin – це не тільки лямбда для Android, але і сучасна мова, що дозволяє зробити розробку під Android простіше і приємніше без особливих витрат.

Також «Android Studio» пропонує підтримку мови C та C++, але тільки при використанні Android NDK. Це означає, що код, написаний на C і C++ можна виконати на JVM, але виконати безпосередньо на самому пристрої, що дає більший контроль над такими речами, як, наприклад, пам'ять. Для вимогливих додатків це допоможе вичавити з пристрою максимум продуктивності.

Також ви маєте можливість використовувати бібліотеки, написані на C або C++, але великим мінусом є важка настройка Android NDK, велика кількість помилок низька гнучкість C і C++(складно додавати до існуючої програми нові функціональні можливості). Наприклад, якщо хочеться створити комп'ютерну гру, то на багато простіше та краще скористатися вже готовим ігровим движком. Мовою розробки додатку була використана мова Java, розроблена компанією Sun Microsystem.

Головними особливостями Java вважається надійність, безпека та продуктивність. Сама мова Java була випущена компанією Sun Microsystems в далекому 1995 році. Код Java можна виконати на будь-якому пристрої, так як спочатку він транслюється в спеціальний байт-код, код Java сумісний з різними платформами, тому потім цей байт-код виконується віртуально машиною JVM.

1.3. Методи розробки мобільних додатків

Існує 2 методи розробки мобільних додатків:

- Нативна розробка;
- Кросплатформенна, або гібридна розробка (ReactNative, Flutter).

Нативна розробка має на увазі створення програми для мобільного пристрою на конкретній мові під конкретну платформу. Нативні додатки досить продуктивні і не мають обмежень в розробці (Java і Kotlin – для Android, а Swift – для iOS).

Плюси і недоліки

До плюсів такої розробки можна віднести досить швидку реакцію на дії користувача, можливість мати прямий доступ до апаратної частини і розробити найбільш звичний для користувача конкретної платформи інтерфейс. До недоліків можна віднести досить високу вартість розробки і підтримки, і тривалий час, необхідний на розробку.

Розробляти один і той самий додаток під різні платформи (iOS / Android) довго і дорого, до того ж їх тестування також займе досить тривалий час (наприклад, компіляція оновлень для Андроїд займає близько 30 сек), з цього якщо потрібно створити простий додаток відразу для двох платформ, вдаються до гібридного способу розробки.

Кросплатформенна розробка проводиться за допомогою web-технологій – HTML, CSS і JavaScript, які дозволяють розробити додаток відразу на кілька платформ. Але для того, щоб додаток працював у

відповідності зі своєю платформою, його потрібно “перевести” на зрозумілу платформі мову, або додати проміжну ланку-перекладач. До переваг можна віднести низьку вартість розробки, адже для цього іноді достатньо буде задіяти одного фахівця.

А ось до недоліків можна віднести можливі труднощі в роботі всіх функцій і затримки в реакції на дію користувача (додаток може бути повільним), також інтерфейс буде досить простим і потрібно буде його додатково допрацьовувати.

1.4. Формування та постановка задач розробляемого мобільного додатку

Цей етап включає визначення основних цілей та функціональності додатку, які ви хочете досягти. Нижче наведено кілька кроків, які можуть допомогти у формуванні та постановці задач:

Визначення цілей: Визначте основну мету вашого мобільного додатку. Наприклад, ціль може полягати в наданні користувачам зручного та ефективного інструменту для планування подорожей, що допомагає їм знайти оптимальний маршрут, забронювати готель, купити квиток і т.д.

Виявлення потреб користувачів: Дослідіть потреби та очікування вашої цільової аудиторії. Врахуйте, які функції та можливості були б корисними для користувачів під час планування та здійснення подорожей. Наприклад, це можуть бути пошук готелів за критеріями, відстеження польотів, можливість зберігання документів тощо.

Формулювання функціональності

На основі виявлених потреб користувачів визначте основні функції та можливості додатку. Сформулюйте ці функції у вигляді конкретних задач. Наприклад, задачею може бути "Розробити модуль пошуку готелів, який дозволяє користувачам шукати готелі за назвою, місцезнаходженням, категорією, ціною тощо".

Встановлення пріоритетів

Розподіліть задачі за пріоритетом. Визначте, які функції є основними та невідкладними, а які можуть бути реалізовані на пізніших етапах. Це допоможе вам керувати розробкою та зосередитися на найважливіших аспектах додатку.

Формулювання вимог

Сформулюйте конкретні вимоги до додатку. Вимоги повинні бути чіткими, вимірюваними та реалізованими. Наприклад, вимогою може бути "Додаток повинен мати можливість відображати розклад польотів з онлайн-джерела та оновлювати його у режимі реального часу".

Перевірка цілей та задач

Переконайтеся, що сформульовані цілі та задачі відповідають потребам користувачів та мають практичну цінність. Розгляньте можливість зворотного зв'язку від потенційних користувачів або фахівців у галузі, щоб впевнитися, що ви на правильному шляху.

1.5. Аналіз вимог до програмного забезпечення

Для визначення вимог до програмного забезпечення необхідно визначити складність інтерфейсу та можливості в системі. Для досягнення мети розробки система повинна містити наступні можливості: –користувач; – адміністратор. Користувач може використовувати загальний функціонал додатку.

Загальний функціонал додатку:

- Перегляд інформації стосовно подорожей;
- Користуватися пошуковою системою;
- Редагувати створені розділи у списку;
- Використання Google Maps для прокладання шляху та додавання маркерів.

1.5.1. Розроблення функціональних вимог

В системі передбачені наступні варіанти використання:

- Реєстрація користувача, реєстрація користувача проходить в вікні після натискання на кнопку «Реєструватися», користувач повинен заповнити усі поля, якщо системи виявить помилку, то користувач отримає її;

- Авторизація користувача, авторизація проходить у вікні після натискання на кнопку «Увійти», де користувач повинен заповнити усі поля та натиснути на кнопку «Увійти», якщо користувач ввів не вірні дані, тоді він отримає повідомлення про помилку;

- Користувач може використовувати увесь функціонал застосунку але не може редагувати та видаляти будь-що з колективних записів, де не є адміністратором.

1.5.2. Розроблення не функціональних вимог

Програмне забезпечення повинне відповідати наступним нефункціональним вимогам:

1. Локалізація інтерфейсу – українська для української локалізації пристрою;
2. Підтримувана версія ОС для мобільних додатків Android 9.0 та вище;
3. Для передачі даних повинні використовуватися захищені канали передачі даних з використанням TLS.

1.6. Аналіз відомих програмних продуктів

Існує декілька аналогів програмного забезпечення(ПЗ) для допомоги мандрівникам у подорожуванні. Кожна з них розрахована на різну кількість користувачів, але функціонал у всіх різний.

1.6.1. Мобільний застосунок «Triplt»

Мобільний застосунок «Triplt» є відомим інструментом для планування та керування подорожами. Він пропонує широкий спектр функціональності, що спрямована на полегшення організації та виконання подорожей. Основні риси та функції Triplt включають:

1. Маршрут під час подорожі: «Triplt» надає можливість створювати та зберігати маршрути подорожей, включаючи рейси, готелі, прокат автомобілів та інші дії. Користувачі можуть додавати деталі подорожі вручну або імпортувати їх з електронної пошти чи календаря.

2. Управління резерваціями: Застосунок дозволяє користувачам керувати своїми резерваціями готелів, авіарейсів, прокату автомобілів та інших послуг. Він може автоматично імпортувати резерваційні дані з електронної пошти та зберігати їх на одному місці для зручності.

3. Нагадування та сповіщення: «Triplt» надсилає користувачам нагадування про майбутні рейси, готельні бронювання та інші події, щоб допомогти їм вчасно приготуватися до подорожі. Він також може надсилати сповіщення про зміни у розкладах або інші важливі оновлення.

4. Збереження документів: Застосунок дозволяє користувачам зберігати важливі документи, такі як паспорти, візи, страхові поліси та інші, у цифровому форматі. Це дозволяє мати легкий доступ до необхідної документації під час подорожі.

5. Поділ подорожей: «Triplt» дозволяє користувачам ділитися своїми подорожами з іншими людьми. Це корисна функція для сімейних подорожей, командних відряджень або подорожей з друзями, коли декілька осіб беруть участь у плануванні та організації.

Мобільний застосунок Triplt надає користувачам зручність та організацію під час планування та виконання подорожей. Його функціональність допомагає зберегти час та зусилля, а також покращити загальний досвід подорожей. Аналізуючи Triplt та інші подібні програмні

продукти, ви можете здобути ідеї та використовувати їх в розробці свого мобільного додатку для планування подорожей на платформі Android.

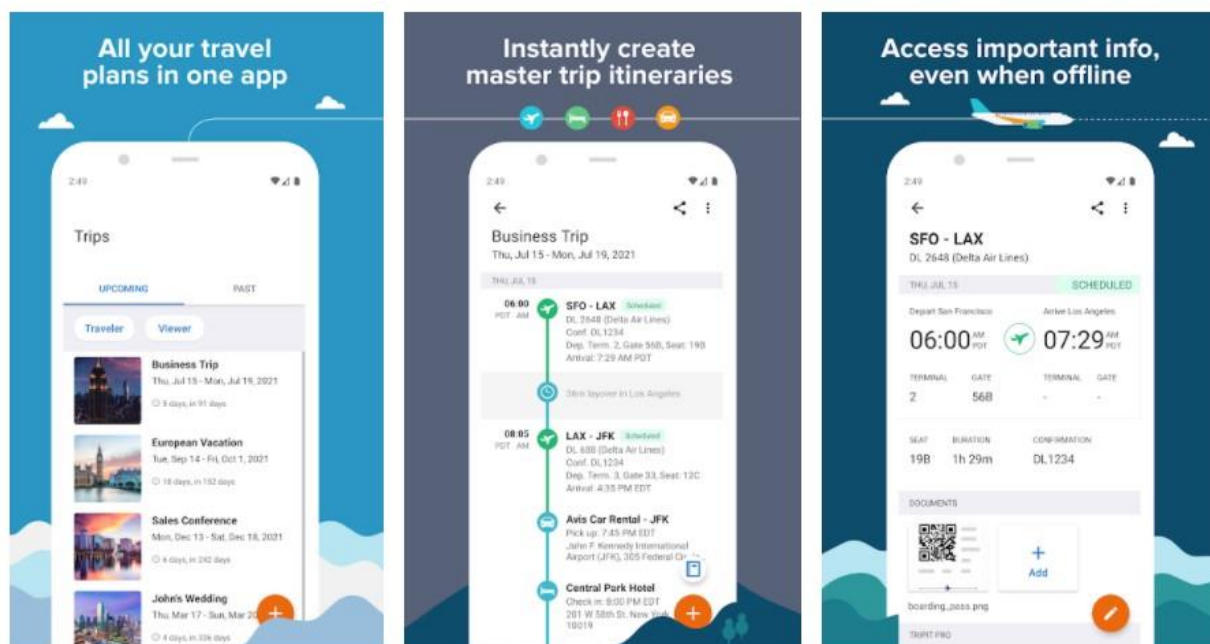


Рис 1.1. Головне меню запуску Triplt

1.6.2. Мобільний застосунок «PackPoint»

Мобільний застосунок «PackPoint» є ще одним відомим інструментом для планування подорожей.

Він спроектований для допомоги користувачам скласти список необхідних речей та пакувати їх перед від'їздом. Основні риси та функції «PackPoint» включають:

1. Автоматичне формування списку пакування: застосунок «PackPoint» автоматично створює персоналізований список пакування на основі інформації про місце призначення, тип подорожі, тривалість перебування та прогноз погоди. Він враховує різні фактори, такі як погодні умови, активності та подорожі з дітьми, щоб забезпечити повний та зручний список необхідних речей.

2. Додавання власних елементів: користувачі можуть додавати власні елементи до списку пакування. Це дозволяє персоналізувати список відповідно до своїх потреб і враховувати особисті речі, які можуть бути важливими для конкретної подорожі.

3. Керування списком: «PackPoint» дозволяє користувачам відзначати елементи, які вони вже запакували, та стежити за прогресом пакування. Він також надає можливість переглядати список пакування з раніше здійснених подорожей, що спрощує планування майбутніх поїздок.

4. Рекомендації щодо пакування: застосунок «PackPoint» надає корисні поради та рекомендації щодо пакування, зокрема стосовно одягу, аксесуарів, медичних засобів, документів та інших речей. Це може бути особливо корисним для людей, які подорожують в нові місця або не мають досвіду планування подорожей.

5. Інтеграція з іншими сервісами: «PackPoint» може інтегруватися з іншими сервісами та календарями, щоб отримувати інформацію про подорожі та резервації. Це спрощує організацію та планування подорожей.

6. Аналізуючи «PackPoint» та інші подібні програмні продукти, ви можете отримати ідеї та натхнення для свого мобільного додатку для планування подорожей на платформі Android. Врахуйте їх функціональність, дизайн та особливості, щоб покращити свій додаток та забезпечити зручний досвід користувача.

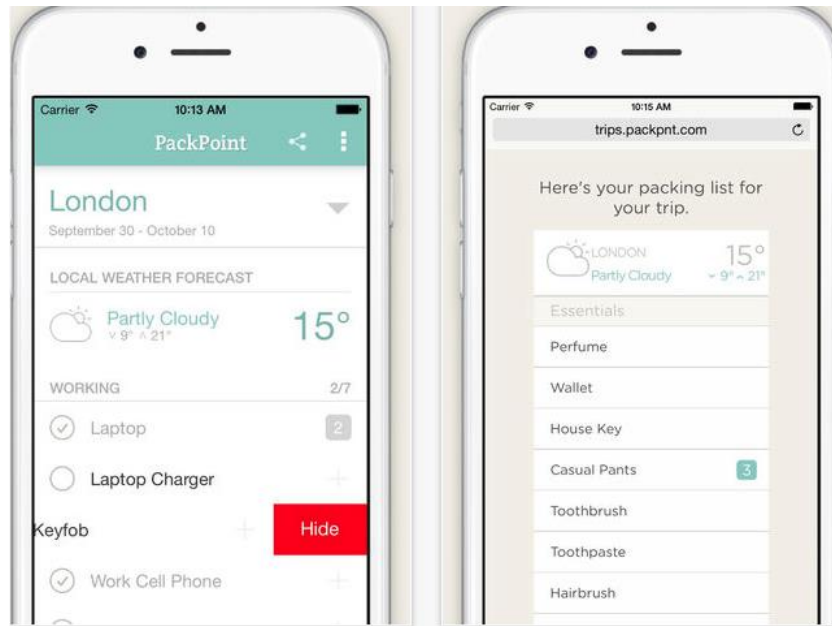


Рис 1.2. Функції «PackPoint»

1.6.3. Мобільний застосунок «Booking»

1. Мобільний застосунок «Booking» є одним з найвідоміших та популярних сервісів для бронювання готелів, апартаментів, гостьових будинків та інших типів помешкань. Основні риси та функції «Booking» включають:

2. Пошук та бронювання помешкань: Застосунок дозволяє користувачам здійснювати пошук помешкань за різними параметрами, такими як місце розташування, дати перебування, кількість гостей та багато іншого. Користувачі можуть переглядати фотографії, описи та відгуки про помешкання, а також забронювати їх безпосередньо через додаток.

3. Користувацькі відгуки та оцінки: «Booking» надає можливість користувачам залишати відгуки про помешкання, в яких вони перебували. Це допомагає іншим користувачам зробити інформований вибір та оцінити якість та зручність різних варіантів проживання.

4. Керування бронюваннями: За допомогою «Booking» користувачі можуть керувати своїми бронюваннями, переглядати деталі, внесені зміни,

скасовувати або змінювати резервації. Вони також отримують нагадування та сповіщення про наближення дати перебування та інші важливі події.

5. Фільтри та сортування: Застосунок надає різні фільтри та опції сортування, що дозволяють користувачам швидко знайти найбільш підходящі пропозиції за своїми вимогами. Вони можуть вибрати варіанти, які відповідають їх бюджету, типу помешкання, відстані від популярних місць та іншим параметрам.

6. Картки путівника: «Booking» також надає користувачам корисні інформаційні картки путівника, які містять дані про місце призначення, популярні пам'ятки, рекомендації щодо відвідування та іншу корисну інформацію.

Мобільний застосунок «Booking» допомагає користувачам знайти та забронювати помешкання під час подорожі. Аналізуючи «Booking» та інші подібні програмні продукти, ви можете отримати ідеї та використовувати їх в розробці свого мобільного додатку для планування подорожей на платформі Android.

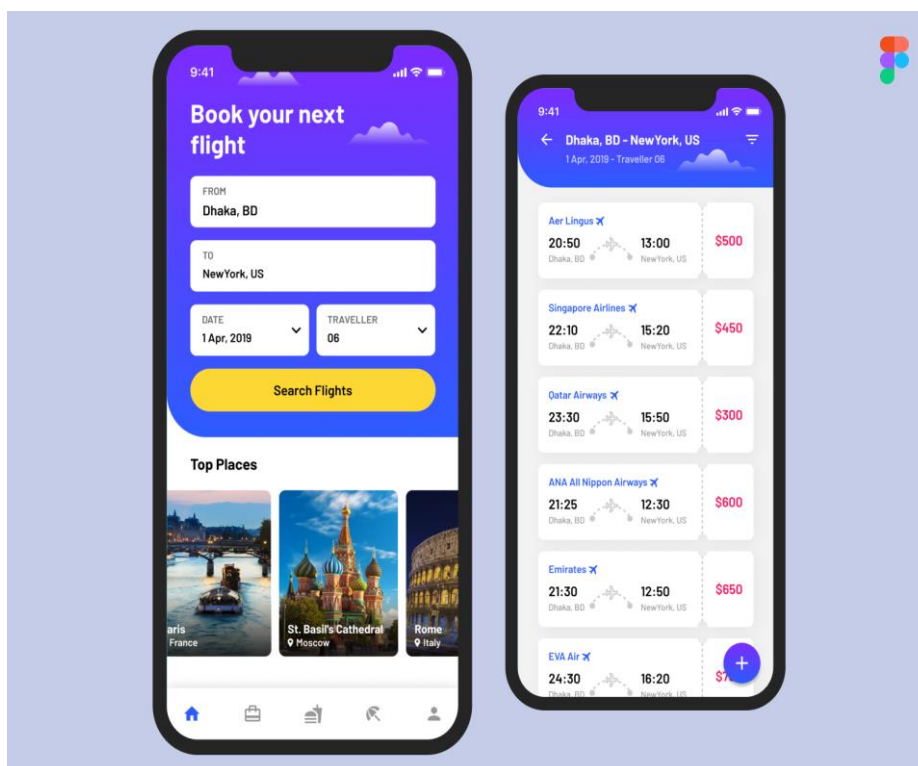


Рис 1.3. Функції «Booking»

1.6.4. Мобільний застосунок «Airbnb»

Мобільний застосунок «Airbnb» є одним з найпопулярніших та відомих сервісів для бронювання житла та переживання автентичних місцевих вражень під час подорожі. Основні риси та функції «Airbnb» включають:

1. Пошук та бронювання унікальних помешкань: Застосунок «Airbnb» дозволяє користувачам знаходити різні типи житла, включаючи квартири, будинки, вілли, кімнати та навіть незвичайні помешкання, такі як будиночки на деревах або човни. Користувачі можуть шукати за місцем розташування, датами перебування, кількістю гостей та іншими фільтрами.

2. Користувацькі відгуки та оцінки: «Airbnb» надає можливість користувачам залишати відгуки про житло, в якому вони перебували, а також оцінювати господарів. Це допомагає іншим користувачам прийняти інформоване рішення щодо бронювання та оцінити якість та гостинність господарів.

3. Взаємодія з господарями: Користувачі можуть спілкуватися з господарями через вбудовану систему повідомлень в застосунку. Вони можуть задавати питання, уточнювати деталі про житло та домовлятися про додаткові послуги або зміни в резервації.

4. Досвіди та екскурсії: «Airbnb» також пропонує розділ, де користувачі можуть бронювати унікальні досвіди та екскурсії, проведені місцевими жителями. Це дозволяє пізнати культуру та автентичність місцевих місць під керівництвом експертів.

5. Керування бронюваннями та оплатою: Користувачі можуть керувати своїми бронюваннями, переглядати деталі, змінювати дати перебування, скасовувати бронювання та здійснювати оплату за допомогою зручних платіжних методів.

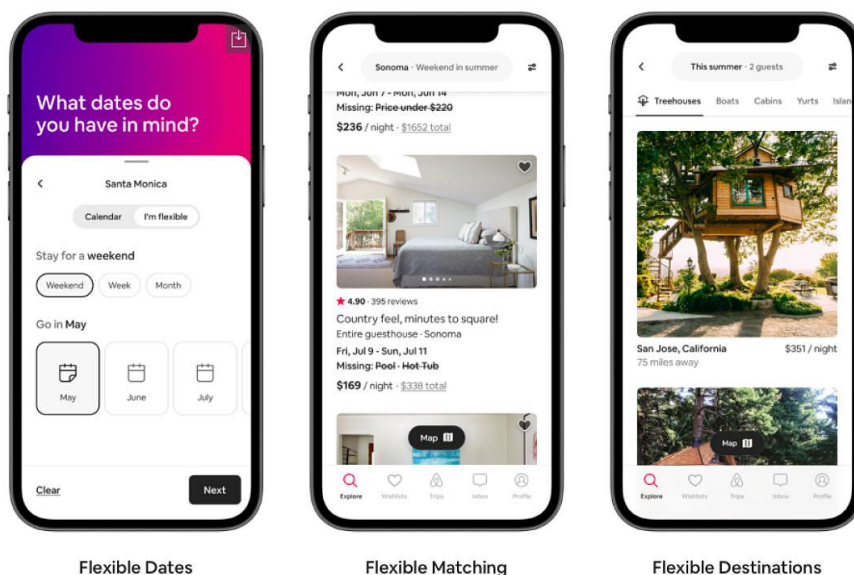


Рис 1.4. Функції «Airbnb»

1.6.5. Мобільний застосунок «Couchsurfing»

1. Мобільний застосунок «Couchsurfing» є популярною платформою, яка з'єднує мандрівників з місцевими жителями, які готові надати безкоштовне помешкання або поділитися своїм простором під час подорожей. Основні риси та функції «Couchsurfing» включають:

2. Пошук та зв'язок з господарями: Застосунок дозволяє користувачам знайти місцевих господарів, які готові прийняти їх у своєму житлі. Користувачі можуть переглядати профілі господарів, читати відгуки та зв'язуватися з ними для узгодження деталей перебування.

3. Безпечність та перевірка профілів: «Couchsurfing» надає можливість користувачам підтвердити свою особу та проходити перевірку, що додає додатковий рівень безпеки. Користувачі можуть також залишати відгуки та оцінки про гостей та господарів, що сприяє створенню довіри в спільноті.

4. Поділ місцевих вражень: Користувачі «Couchsurfing» можуть пропонувати та приймати запрошення на зустрічі, екскурсії та події, які організують інші учасники спільноти. Це дозволяє мандрівникам

познайомитися з місцевими жителями та дізнатися більше про культуру та атмосферу різних місць.

5. Групи та заходи: «Couchsurfing» має функціонал груп та подій, де користувачі можуть приєднуватися до спеціалізованих груп за інтересами або приймати участь у вже запланованих подіях, таких як зустрічі, вечірки, спортивні заходи тощо.

6. Поширення гостинності: Основним принципом «Couchsurfing» є поділ гостинності. Користувачі можуть бути як гостями, так і господарями, що розширює можливості для міжкультурного обміну та взаєморозуміння.

Мобільний застосунок «Couchsurfing» допомагає мандрівникам знаходити безкоштовне помешкання та спілкуватися з місцевими жителями. Аналізуючи «Couchsurfing» та інші подібні програмні продукти, ви можете здобути ідеї та використовувати їх у розробці свого мобільного додатку для планування подорожей на платформі Android.

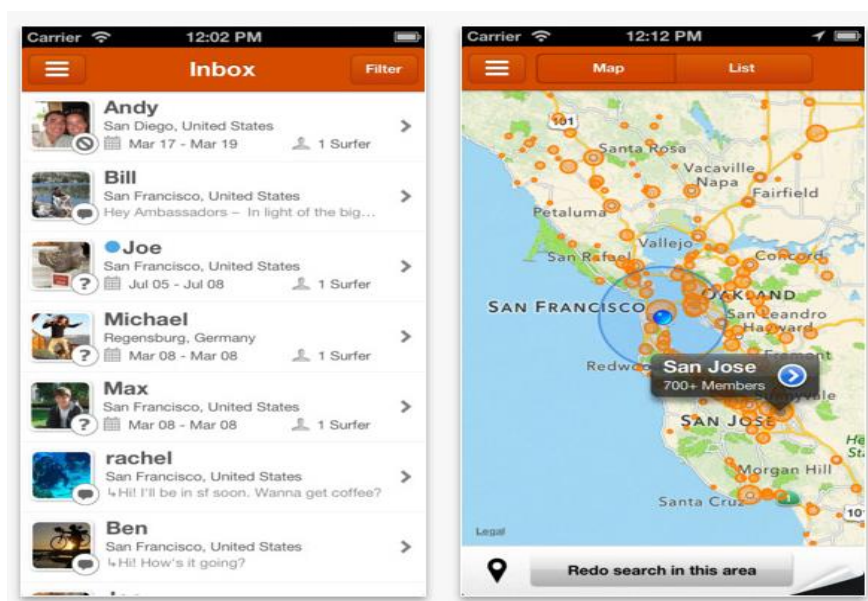


Рис 1.5. Функції «Couchsurfing»

1.7. Висновок до 1 розділу

У першому розділі дипломної роботи було проведено аналіз предметної області, а саме мобільних застосунків для планування подорожей. Було розглянуто технологічні методи досліджень, аналізовано вимоги до програмного забезпечення та проведено огляд відомих програмних продуктів.

Під час аналізу предметної області було встановлено, що популярні мобільні додатки для планування подорожей надають широкий набір функцій та можливостей. Зокрема, це включає пошук та бронювання помешкань, відображення інформації про місця відпочинку, взаємодію з місцевими жителями, організацію додаткових послуг, поділ вражень та досвіду, а також забезпечення безпеки та довіри.

Під час аналізу відомих програмних продуктів, таких як "Triplt", "PackPoint", "Booking", "Airbnb" та "Couchsurfing", були виявлені їх основні функції та особливості. Ці додатки надають зручність, широкий вибір та забезпечують задоволення потреб користувачів у плануванні та організації подорожей.

Висновок з аналізу показує, що розробка мобільного додатку для планування подорожей на платформі Android має потенціал стати цінним інструментом для користувачів, які шукають зручність, ефективність та персоналізацію у своїх подорожах.

Враховуючи розглянуті технологічні методи досліджень та аналізовані вимоги до програмного забезпечення, можна розробити інноваційний додаток, який задовольнить потреби користувачів у плануванні та організації подорожей.

У наступних розділах дипломної роботи буде розроблено концепцію та функціонал мобільного додатку, проведено його реалізацію на мові Java та здійснено його тестування. При цьому будуть враховані результати проведеного аналізу предметної області та вимог до програмного забезпечення.

РОЗДІЛ 2

АНАЛІЗ ТА ПРОЕКТУВАННЯ

2.1. Проектування архітектури додатку

Проектування архітектури додатку є важливим етапом у розробці мобільного додатку для планування подорожей. Правильно спроектована архітектура забезпечує модульність, розширюваність, підтримку підрядків та ефективність програмного забезпечення. Основні етапи проектування архітектури додатку включають:

1. Визначення архітектурних вимог: на цьому етапі визначаються основні вимоги до архітектури додатку, такі як масштабованість, безпека, швидкодія, розширюваність та інші;

2. Вибір архітектурного стилю: залежно від вимог та особливостей проекту можна обрати певний архітектурний стиль, наприклад, Model-View-Controller (MVC), Model-View-ViewModel (MVVM), Clean Architecture тощо.

3. Розбиття на модулі: додаток можна розбити на логічні модулі, що відповідають різним функціональним блокам або компонентам. Наприклад, модуль для роботи з базою даних, модуль для керування користувачем, модуль для відображення інтерфейсу тощо.

4. Проектування бази даних: якщо додаток використовує базу даних, потрібно спроектувати структуру бази даних, таблиці, зв'язки між ними та необхідні запити для роботи з даними;

Кафедра КІТ				НАУ 23 35 41 000 ПЗ			
Виконавець	Самойлов О.М.			АНАЛІЗ ТА ПРОЕКТУВАННЯ	Літера	Аркуш	Аркушів
Керівник	Харченко О.Г.					29	5
Консультант					<i>УС-412Б 122</i>		
Н.Контроль	Шевченко О.П.						

5. Вибір технологій: на цьому етапі визначаються технології, бібліотеки та фреймворки, які будуть використовуватися для реалізації архітектури додатку. Наприклад, вибір фреймворка для розробки інтерфейсу користувача, бібліотек для роботи з мережею або базою даних.

6. Створення діаграм архітектури: для кращого розуміння архітектури додатку можна створити діаграми, такі як діаграма компонентів, діаграма класів, діаграма послідовності та інші. Ці діаграми допомагають візуалізувати взаємодію між компонентами та основні потоки роботи додатку.

7. Реалізація та тестування: після проектування архітектури додатку можна приступити до його реалізації, використовуючи вибрані технології та бібліотеки. Після реалізації необхідно провести тестування для перевірки працездатності та відповідності вимогам.

Проектування архітектури додатку важливо проводити з урахуванням потреб користувачів, функціональних вимог, вимог до продуктивності та інших факторів, які можуть впливати на якість та ефективність додатку.

2.2. Вибір мови програмування та середовища розробки

Вибір мови програмування та середовища розробки є важливими аспектами при розробці мобільного додатку для планування подорожей. Вони визначають зручність розробки, продуктивність, підтримку різних функцій та інші фактори.

Мова програмування: Для розробки мобільних додатків для платформи Android найпоширенішою мовою програмування є Java. Java є офіційною мовою програмування для розробки Android-додатків, вона має велику спільноту розробників та багатий екосистему бібліотек і фреймворків. Використання Java дозволяє забезпечити широкую підтримку пристроїв та високу продуктивність додатку.

Однак, наразі також популярно використання мови Kotlin для розробки Android-додатків. Kotlin є сучасною мовою програмування, яка працює на віртуальній машині Java (JVM) і надає багато нових функцій та покращень у порівнянні з Java. Kotlin дозволяє писати більш компактний та зрозумілий код, що полегшує розробку та підтримку додатку.

Середовище розробки: Для розробки мобільних додатків для платформи Android можна використовувати різні середовища розробки (IDE). Офіційною рекомендацією для розробки Android-додатків є Android Studio. Android Studio є спеціалізованою IDE, яка надає багатий набір інструментів та функцій для розробки Android-додатків. Вона інтегрується з Android SDK та забезпечує зручне управління проектом, візуальний редактор інтерфейсу користувача, інструменти для збірки та тестування додатку.

Окрім Android Studio, є й інші популярні середовища розробки, такі як IntelliJ IDEA та Eclipse, які також підтримують розробку Android-додатків.

При виборі мови програмування та середовища розробки варто враховувати власні знання та вміння розробника, підтримку спільноти розробників, доступність документації та ресурсів для вивчення. Також важливо враховувати особливості проекту та вимоги до додатку.

2.3. Розробка інтерфейсу користувача

Розробка інтерфейсу користувача (UI) є важливою складовою частиною розробки мобільного додатку для планування подорожей. Якісний та зручний інтерфейс допомагає користувачам легко взаємодіяти з додатком та забезпечує позитивний досвід використання.

Основні кроки у розробці інтерфейсу користувача:

1. Аналіз вимог користувача: розуміння потреб та вимог користувачів є ключовим етапом у розробці інтерфейсу. Вивчення цільової аудиторії та її потреб допомагає визначити функціональність та елементи інтерфейсу, необхідні для задоволення користувачів.

2. Створення скетчів та проведення wireframing: скетчі та wireframes - це прості схематичні малюнки, які відображають основну структуру та компоненти інтерфейсу. Це допомагає візуалізувати розташування елементів, навігацію та загальну логіку інтерфейсу.

3. Дизайн UI: на цьому етапі розробляються детальні дизайн-макети інтерфейсу. Використовуються графічні редактори, такі як Adobe Photoshop або Sketch, для створення привабливого та сучасного вигляду інтерфейсу. Враховуються принципи дизайну, такі як схема кольорів, типографіка, розташування елементів, використання іконок тощо.

4. Прототипування: створення прототипу дозволяє інтерактивно протестувати та перевірити функціонал інтерфейсу. Використовуються інструменти прототипування, такі як InVision, Figma або Adobe XD, для створення динамічних прототипів, де користувачі можуть спробувати різні функції та взаємодіяти з інтерфейсом.

5. Тестування та вдосконалення: після створення прототипу важливо провести тестування залежності інтерфейсу на різних пристроях і з різними розмірами екранів. Зворотний зв'язок від користувачів може допомогти виявити проблеми та вдосконалити інтерфейс для кращої взаємодії.

6. Реалізація інтерфейсу: після завершення дизайну та тестування можна приступити до реалізації інтерфейсу. Використовуються мови програмування та інструменти розробки, такі як XML або Kotlin/Java для Android, для створення реальних екранів та елементів інтерфейсу.

7. Постійне вдосконалення: інтерфейс користувача повинен бути постійно покращуваний на основі отриманого відгуку користувачів, оновлень платформи та нових трендів у дизайні. Регулярні оновлення та вдосконалення дозволяють забезпечити високу якість інтерфейсу та задоволення користувачів.

Під час розробки інтерфейсу користувача важливо враховувати простоту взаємодії, консистентність дизайну, ергономіку, адаптивність до різних розмірів екранів та кращі практики у галузі UX/UI дизайну.

2.4. Висновок до 2 розділу

У другому розділі було проведено проектування архітектури додатку та вибір мови програмування та середовища розробки. Архітектура додатку була розроблена з урахуванням вимог до функціональності та ефективності додатку. Вибір мови програмування було обгрунтовано на основі доступності ресурсів та спільноти розробників.

Для розробки мобільного додатку для планування подорожей рекомендовано використовувати мову програмування Java або Kotlin для платформи Android. Обидві мови мають свої переваги та підтримуються відповідними інструментами розробки. Вибір мови залежить від власних знань та вподобань розробника.

Щодо середовища розробки, Android Studio є рекомендованим середовищем для розробки Android-додатків. Воно надає широкий набір інструментів та підтримку для розробки, тестування та налагодження додатку. Однак, також можна використовувати інші популярні середовища розробки, такі як IntelliJ IDEA або Eclipse, якщо вони відповідають потребам розробки.

Вибір мови програмування та середовища розробки є важливими кроками для успішної розробки додатку. Правильний вибір допоможе забезпечити продуктивність, ефективність та зручність розробки.

РОЗДІЛ 3

РЕАЛІЗАЦІЯ ТА ТЕСТУВАННЯ

3.1. Розробка компонентів додатку

З самого початку розробки додатку, запишемо конфігурацію додатка для платформи Android, його вимоги до пристроїв і дозволи, які будуть необхідні для коректної роботи. Основні елементи коду мають такі значення:

`versionCode="1"` - числовий ідентифікатор версії додатка, який використовується для внутрішніх потреб розробки;

`versionName="1.0"` - рядковий ідентифікатор версії додатка, який відображається користувачам;

`supports-screens` - вказує, які типи екранів підтримуються додатком (люба щільність, великі екрани, звичайні екрани і т.д.);

`uses-permission` - визначає дозволи, необхідні для функціонування додатка. Наприклад, доступ до Інтернету (`android.permission.INTERNET`), камери (`android.permission.CAMERA`), місцезнаходження (`android.permission.ACCESS_COARSE_LOCATION`, `android.permission.ACCESS_FINE_LOCATION`), зчитування стану телефону (`android.permission.READ_PHONE_STATE`) та багато інших.

`<uses-feature>` - вказує додаткові апаратні вимоги для пристроїв, на яких може бути встановлено додаток. У цьому випадку, додаток вимагає наявності GPS на пристрої (`android.hardware.location.gps`).

Кафедра КІТ				НАУ 23 35 41 000 ПЗ			
Виконавець	Самойлов О.М.			РЕАЛІЗАЦІЯ ТА ТЕСТУВАННЯ	Літера	Аркуш	Аркушів
Керівник	Харченко О.Г.					34	33
Консультант					<i>УС-412Б 122</i>		
Н.Контроль	Шевченко О.П.						

Код Manifest.xml.

```
<?xml version='1.0' encoding='utf-8'?>
  <manifest android:hardwareAccelerated="true" android:versionCode="1"
android:versionName="1.0" package="io.appery.project779708"
xmlns:android="http://schemas.android.com/apk/res/android">
    <supports-screens android:anyDensity="true" android:largeScreens="true"
android:normalScreens="true" android:resizeable="true"
android:smallScreens="true" android:xlargeScreens="true" />
    <uses-permission android:name="android.permission.INTERNET" />
    <application
        android:hardwareAccelerated="true"
        android:icon="@mipmap/ic_launcher" android:label="@string/app_name"
        android:supportsRtl="true">
        <activity
            android:configChanges="orientation/keyboardHidden/keyboard/screenSize/locale
/smallestScreenSize/screenLayout/uiMode" android:exported="true"
            android:label="@string/activity_name" android:launchMode="singleTop"
            android:name="MainActivity" android:theme="@style/Theme.App.SplashScreen"
            android:windowSoftInputMode="adjustResize">
            <intent-filter android:label="@string/launcher_name">
                <action android:name="android.intent.action.MAIN" />
                <category android:name="android.intent.category.LAUNCHER" />
            </intent-filter>
        </activity>
    </application>
    <uses-permission android:name="android.permission.CAMERA" />
    <uses-permission
        android:name="android.permission.ACCESS_COARSE_LOCATION" />
    <uses-permission
        android:name="android.permission.ACCESS_FINE_LOCATION" />
```

```

    <uses-permission
android:name="android.permission.ACCESS_LOCATION_EXTRA_COMMANDS
" />
    <uses-permission
android:name="android.permission.READ_PHONE_STATE" />
    <uses-permission android:name="android.permission.RECEIVE_SMS" />
    <uses-permission android:name="android.permission.RECORD_AUDIO" />
    <uses-permission
android:name="android.permission.MODIFY_AUDIO_SETTINGS" />
    <uses-permission android:name="android.permission.VIBRATE" />
    <uses-permission android:name="android.permission.READ_CONTACTS"
/>
    <uses-permission android:name="android.permission.WRITE_CONTACTS"
/>
    <uses-permission
android:name="android.permission.WRITE_EXTERNAL_STORAGE" />
    <uses-permission
android:name="android.permission.ACCESS_NETWORK_STATE" />
    <uses-permission android:name="android.permission.GET_ACCOUNTS" />
    <uses-permission
android:name="com.android.browser.permission.READ_HISTORY_BOOKMARK
S" />
    <uses-permission android:name="android.permission.WAKE_LOCK" />
    <uses-feature android:name="android.hardware.location.gps"
android:required="true" />
</manifest>

```

3.1.1. Реєстрація / авторизація у додатку

У додатку реалізовано особисті кабінети користувача з функціоналом реєстрації, авторизації та відновлення паролю. Для процесів реєстрації, авторизації та відновлення паролю, було створено 4 активності. На рис. 3.1 зображено першу активність, на яку потрапляє користувач, одразу після відкриття додатку – активність авторизації. У цій активності є назва самого додатку, описано тему активності – «Авторизація», є поле логіну та паролю, 2 посилання для відновлення паролю та переходу у активність реєстрації та кнопка для проходження процесу авторизації. У всіх полях інформації контролю (в даному випадку паролю) реалізовано функцію відображення / приховання тексту на екрані.

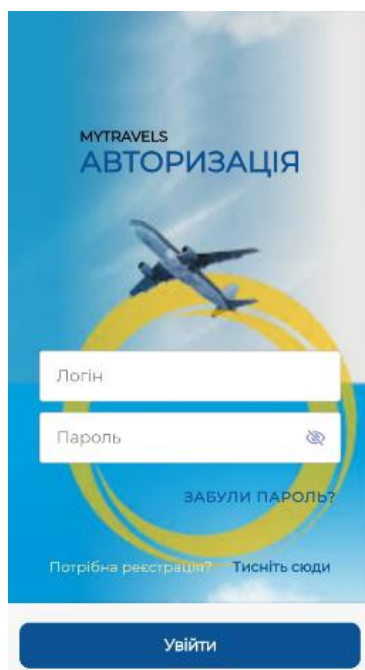
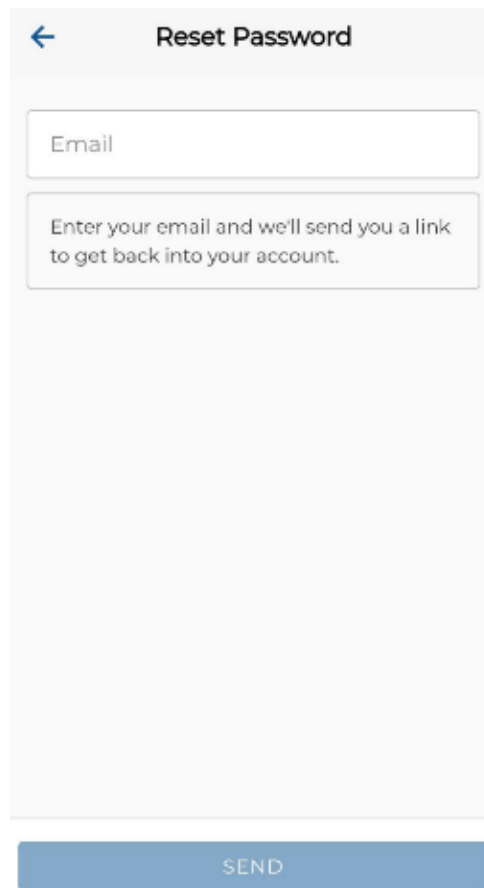


Рис. 3.1. Перша активність додатку – процес авторизації

На рис. 3.2. зображено активність відновлення паролю. У цій активності для користувача, який забув пароль від акаунту, створено інформаційну підказку стосовно дій для відновлення паролю. При вводі електронної адреси та натискання кнопки підтвердження, на неї буде відправлено лист з відновленням паролю.



The image shows a mobile application interface for resetting a password. At the top, there is a back arrow on the left and the title "Reset Password" in the center. Below the title is a text input field labeled "Email". Underneath the input field is a text box containing the instruction: "Enter your email and we'll send you a link to get back into your account." At the bottom of the screen is a blue button labeled "SEND".

Рис. 3.2. Активність процесу відновлення паролю

Після переходу користувачем за посиланням, яке вказано у листі, користувача перенаправляють на захищену веб-сторінку, у якій генерується тимчасовий новий пароль для входу. Після вводу в поля реєстрації пошти та тимчасового паролю, користувача перенаправляють на активність процесу зміни паролю (рис 3.3.).

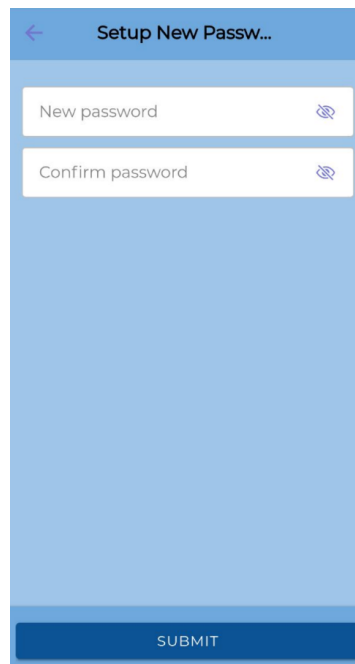


Рис. 3.3. Активність процесу зміни паролю

У активності зміни паролю, користувач вводить новий пароль у перше поле, і дублює його у друге поле. Після натискання кнопки підтвердження, пароль успішно буде змінено.

Якщо користувач ще не має власного акаунту, він проходить процес реєстрації. На рис. 3.4 зображено активність процесу реєстрації нового користувача. У цій активності, створено кнопку підтвердження і 5 полів для реєстрації:

- Ім'я;
- Прізвище;
- Адреса електронної пошти;
- Пароль;
- Дублювання паролю.

Після того, як користувач заповнить всі поля і натисне кнопку "Submit", з'явиться повідомлення про успішну реєстрацію та виконається перехід на «головну» сторінку додатка.

The image shows a mobile application registration screen. At the top, there is a back arrow and the title 'Registration'. Below the title are five text input fields stacked vertically: 'First Name', 'Last Name', 'Email', 'Password', and 'Confirm password'. The 'Password' and 'Confirm password' fields have a small eye icon to the right of the input area, indicating a toggle for password visibility. At the bottom of the form is a wide blue button with the text 'SUBMIT' in white capital letters.

Рис. 3.4. Активність процесу реєстрації нового користувача

3.1.2. Головна активність додатку

Після успішної реєстрації або авторизації, користувача перенаправить на «головну» сторінку, де він побачить головну активність додатку (див. рис. 3.5). З самого початку головної активності, реалізовано:

– Верхню панель навігації, де є:

- Кнопка перегляду отримання запрошення у нову активність подорожі;
- Привітання користувача виду: «Вітаю, (Ім'я користувача)»;
- Кнопка виходу з облікового запису;

– Основна панель зі списками:

- Подорожей, створених користувачем;

- Подорожей, у які користувача запросили і він прийняв запрошення;
 - Карта google maps для спрощення пошуку адреси, перегляду маршрутів, створення міток;
- Нижня панель з кнопкою створення нової подорожі.

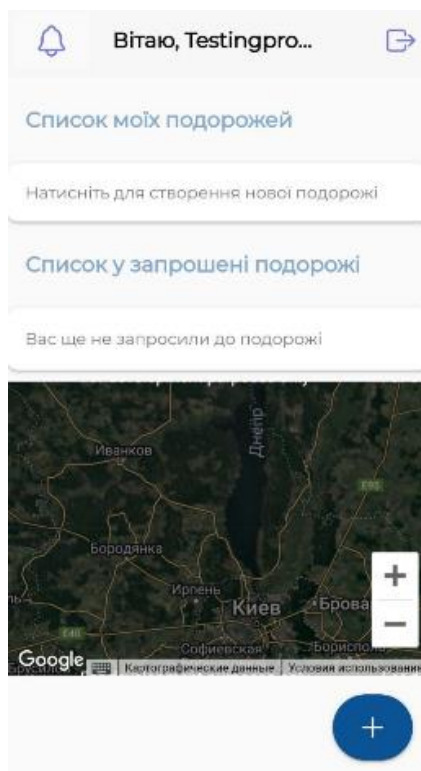


Рис. 3.5. Головна активність додатку

Для створення нової подорожі, користувач натискає у нижньому правому куті головної активності кнопку «+» - додати подорож, де його перенаправляє у активність процесу створення нової подорожі, яка зображена на рис. 3.6.

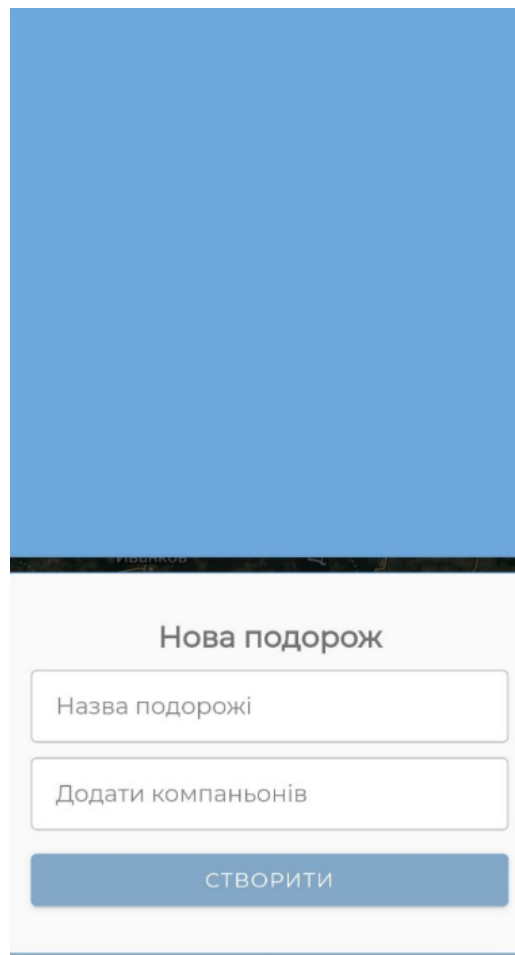


Рис. 3.6. Активність процесу створення нової подорожі

При створенні нової подорожі, користувач вводить тематичну назву подорожі та може додати інших зареєстрованих користувачів додатку у поле «Додати компаньонів», вказавши їх електронні адреси. Якщо користувач введе некоректну адресу, додаток повідомить про це негативним повідомленням. Користувач може залишити друге поле пустим, тоді подорож створиться лише для нього.

На рис. 3.7 зображено відображення щойно створеної нової подорожі з тематичною назвою «Подорож», ця подорож не має приєднаних користувачів.

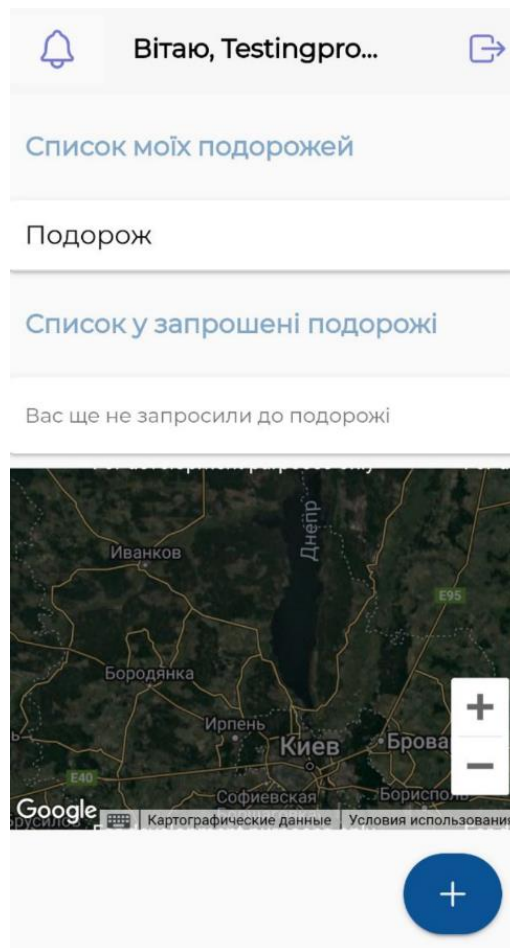


Рис. 3.7 Головна активність зі створеною новою подорожжю

3.1.3. Запис задач до подорожі

Після створення нової подорожі, для запису задач і завдань планування, користувач натискає на поле назви подорожі, після чого відкривається меню обраної подорожі.

На рис. 3.8. зображено активність меню обраної подорожі користувачем. В цьому випадку, користувач ще не додав ні однієї цілі до планування, тому поки що користувач бачить, що всі цілі досягнуто (якщо цілей немає, можна вважати що всі цілі досягнуто). Для створення нової цілі, користувач у правому нижньому кутку може натиснути кнопку створення нової цілі – «+».



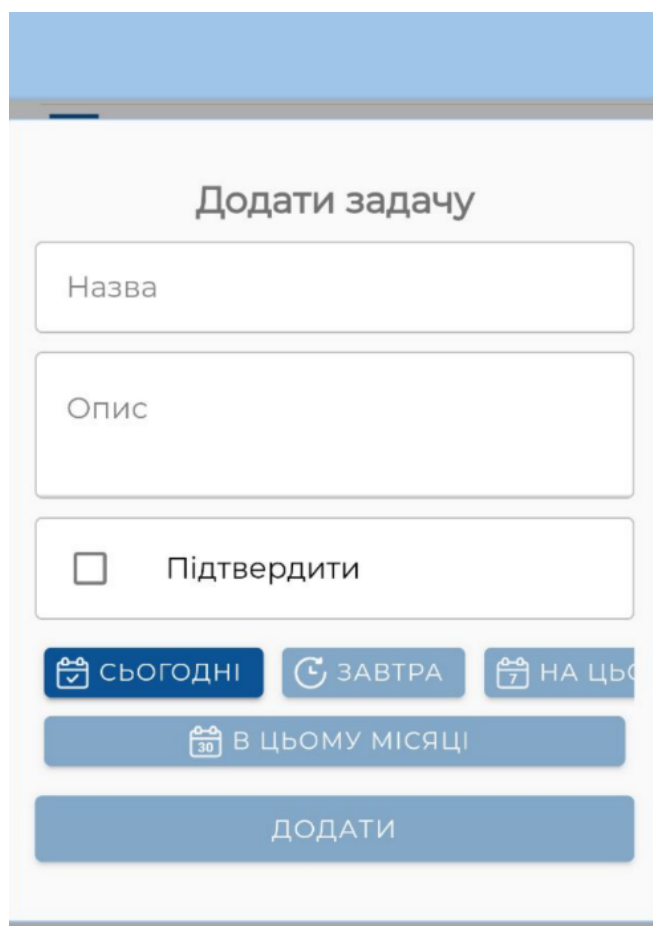
Рис. 3.8 Активність меню подорожі, обрана користувачем

Після натиснення на кнопку додавання нової цілі, користувача перенаправляє на активність створення нової цілі, що зображено на рис. 3.9.

При створенні нової цілі, користувачу потрібно ввести назву цілі у першому полі, детальний і зрозумілий для користувача опис кроків виконання цілі, підтвердити запис, встановивши прапорець у полі перевірки, обрати термін виконання та натиснути кнопку «додати». Термін виконання може бути:

- Сьогодні;
- Завтра;
- Протягом тижня;
- Протягом місяця.

При досягненні сформованої дати дедлайну, користувачу прийде спливаюче повідомлення на екран смартфона.



Додати задачу

Назва

Опис

Підтвердити

СЬОГОДНІ ЗАВТРА НА ЦЬОМУ ТИДНІ

В ЦЬОМУ МІСЯЦІ

ДОДАТИ

Рис. 3.9. Активність створення цілі подорожі

При успішному створенні цілі подорожі, нова комірка цілі з'явиться у списку подорожі, що зображено на рис. 3.10.

У комірці цілі подорожі, буде відображено:

- Дедлайн виконання;
- Назву цілі;
- Опис цілі;
- Та до кого вона відноситься (виконавці цілі, яких може бути декілька у груповій подорожі).

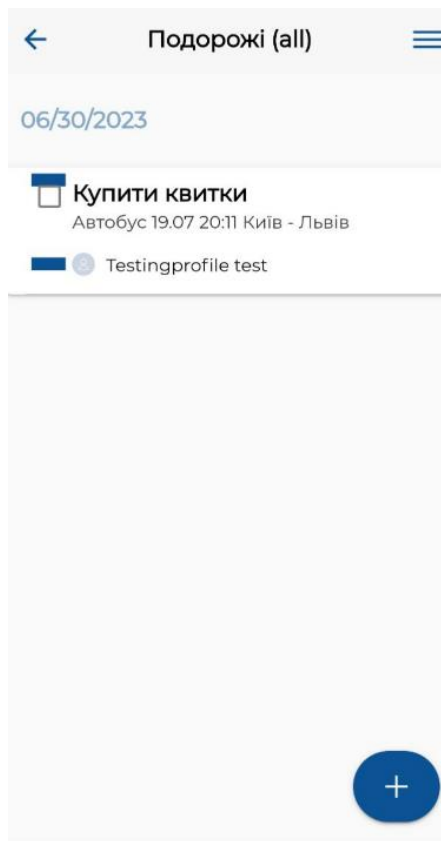


Рис. 3.10. Успішне створення цілі подорожі

При великій кількості цілей з різними параметрами дати, важливості та важкості, реалізовано фільтрацію відображення цілей (рис. 3.11), де користувач може обрати один із пунктів для фільтрації всіх активних цілей, за такими параметри як:

- За фільтром власності:

- Всі цілі;
- Особисті цілі (цілі, до яких має відношення користувач);

- За терміном виконання:

- Сьогодні;
- Завтра;
- Поточний тиждень;
- Поточний місяць.

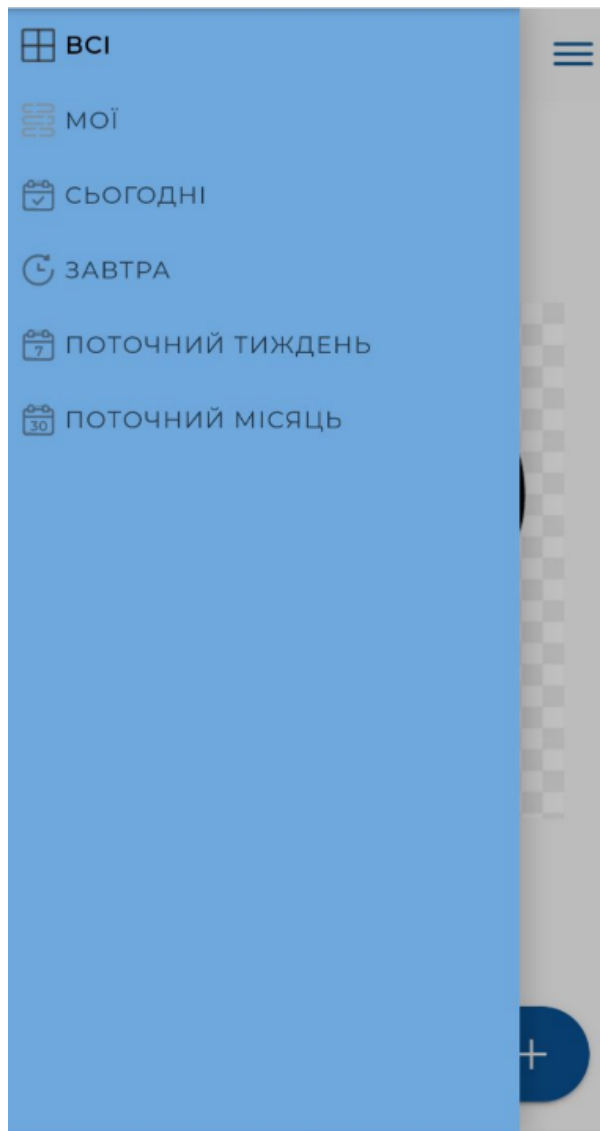


Рис. 3.11. Фільтри відображення цілей

3.1.4. Запрошення до групової подорожі

При створенні нової подорожі, можна вказати користувачів додатку, які будуть додатково приймати в неї участь (рис. 3.6). На рис. 3.12 зображено активність списку запрошень у нові подорожі без активних запрошень. Якщо при створенні нової подорожі, користувач вказав компаньонів для подорожі, вказаним користувачам у активність списку запрошень прийде нове повідомлення.

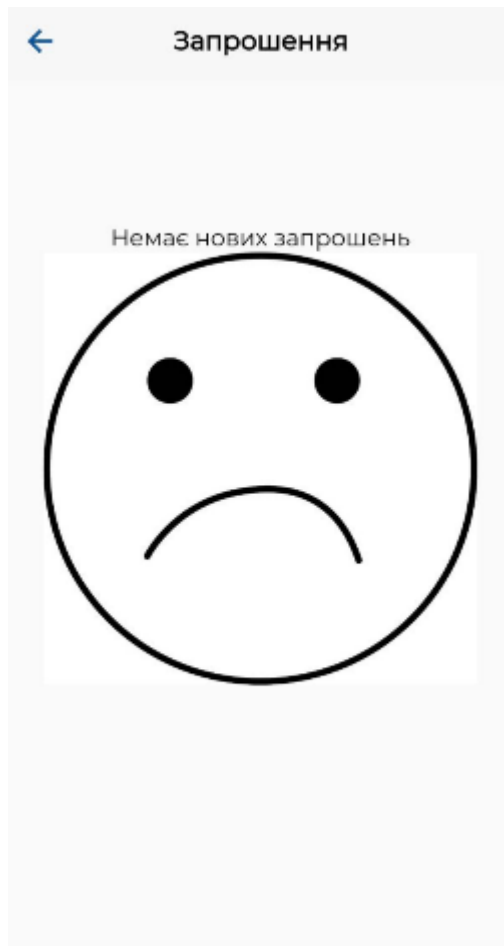


Рис.3.12. Активність списку запрошень без нових запрошень

На рис. 3.13 зображено одне запрошення у нову подорож. У запрошенні вказується назва подорожі, кнопки «прийняти» і «відхилити» запрошення та автора подорожі.

Якщо користувач відхиляє запрошення, автору нової подорожі висвітиться повідомлення, що компаньон відхилив запрошення. Якщо ж користувач приймає запрошення, користувачу у головній активності у списку запрошених подорожей, з'явиться нове поле створеної подорожі.

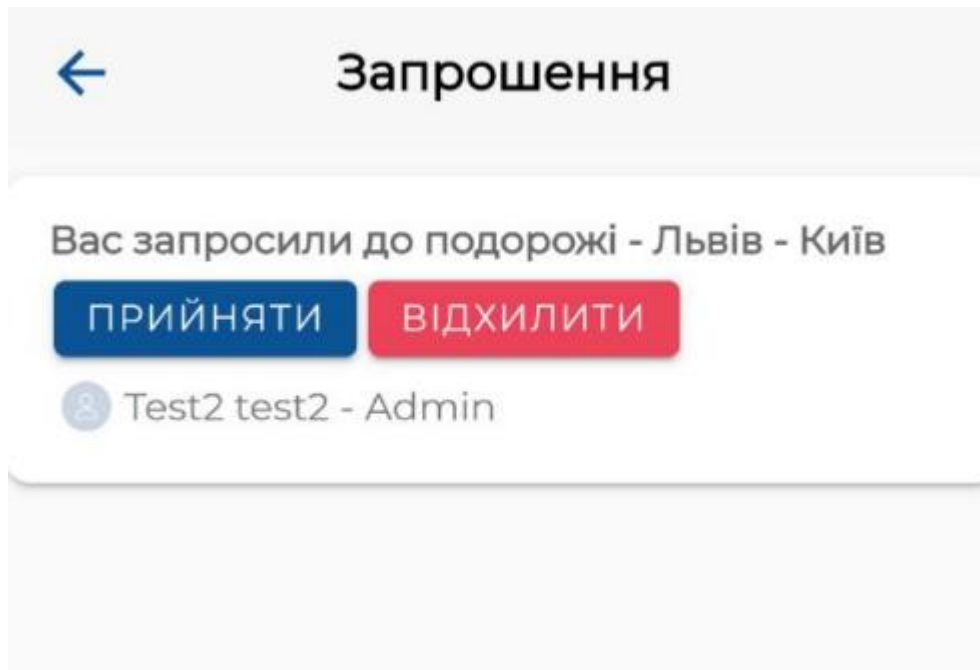


Рис. 3.13. Активність списку запрошень з одним новим запрошенням

3.2. Підключення сервісів

3.2.1. Google maps

У головній активності додатку, створено google мапу за допомоги google maps. При натисканні на мапу, додається маркер, а також прив'язується інформація (назва та опис) до маркера.

Код реалізації мапи.

```
implementation 'com.google.android.gms:play-services-maps:17.0.0'  
import android.os.Bundle;  
import android.widget.Toast;  
import androidx.annotation.NonNull;  
import androidx.appcompat.app.AppCompatActivity;  
import com.google.android.gms.maps.CameraUpdateFactory;  
import com.google.android.gms.maps.GoogleMap;  
import com.google.android.gms.maps.MapView;  
import com.google.android.gms.maps.OnMapReadyCallback;
```

```

import com.google.android.gms.maps.model.LatLng;
import com.google.android.gms.maps.model.Marker;
import com.google.android.gms.maps.model.MarkerOptions;
public class MainActivity extends AppCompatActivity implements
OnMapReadyCallback {
    private GoogleMap googleMap;
    private MapView mapView;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_layout);
        mapView = findViewById(R.id.mapView);
        mapView.onCreate(savedInstanceState);
        mapView.getMapAsync(this);
    }
    @Override
    public void onMapReady(@NonNull GoogleMap googleMap) {
        this.googleMap = googleMap;
        this.googleMap.setOnMapClickListener(new
GoogleMap.OnMapClickListener() {
            @Override
            public void onMapClick(LatLng latLng) {
                addMarker(latLng);
            }
        });
        LatLng initialLocation = new LatLng(37.7749, -122.4194);
        this.googleMap.moveCamera
(CameraUpdateFactory.newLatLngZoom(initialLocation, 10));
    }
    private void addMarker(LatLng latLng) {

```

```

        MarkerOptions markerOptions = new
MarkerOptions().position(latLng);
        Marker marker = googleMap.addMarker(markerOptions);
        marker.setTitle("Місьє");
        marker.setSnippet("Онуч місьє");
        marker.showInfoWindow();
    }
    @Override
    protected void onResume() {
        super.onResume();
        mapView.onResume();
    }
    @Override
    protected void onPause() {
        super.onPause();
        mapView.onPause();
    }
    @Override
    protected void onDestroy() {
        super.onDestroy();
        mapView.onDestroy();
    }
    @Override
    public void onLowMemory() {
        super.onLowMemory();
        mapView.onLowMemory();
    }
}
<com.google.android.gms.maps.MapView
    android:id="@ +id/mapView"

```

```

android:layout_width="match_parent"
android:layout_height="match_parent" />

```

3.2.2. База даних

Базу даних було реалізовано за допомоги SQLite та розміщено у хмарному рішенні.

У базі даних зберігається інформація про такі колекції та значення колекцій як:

- Users;
- Files;
- Devices;
- duties;
- groups;
- notifications;
- groupMembers.

На рис. 3.14 – 3.18 зображено інформацію з колекцій БД.

<input type="checkbox"/>	_id	username string	password password	firstName string	lastName string	updatePwd boolean	notificalons number	groups array	acl
<input type="checkbox"/>	648d41f490fe...	test213test@...	(hidden)	Test2	test2	<input type="checkbox"/>			{**:{read":tr...
<input type="checkbox"/>	648d40b290f...	testprofiles01...	(hidden)	Testingprofile	test	<input type="checkbox"/>		[648d42603c...	{**:{read":tr...
<input type="checkbox"/>	648d37093c6...	hdjdjd@gmail...	(hidden)	yeloeknf	hjdkdkd	<input type="checkbox"/>			{**:{read":tr...
<input type="checkbox"/>	648d34213c6...	TESTTEST@g...	(hidden)	TESTEST	TESTEST	<input type="checkbox"/>			{**:{read":tr...
<input type="checkbox"/>	648d244d90f...	test@gmail.co...	(hidden)	Тест	Тест	<input type="checkbox"/>			{**:{read":tr...
<input type="checkbox"/>	648d20be90f...	qwerty@gmai...	(hidden)	qwerty	qwerty	<input type="checkbox"/>			{**:{read":tr...
<input type="checkbox"/>	648d1fd23c6...	sbjvdjn@gmal...	(hidden)	asfasf	asfafas	<input type="checkbox"/>			{**:{read":tr...
<input type="checkbox"/>	648d1b063c6...	lfgdksm@gm...	(hidden)	цйукенн	уцкуенер	<input type="checkbox"/>			{**:{read":tr...
<input type="checkbox"/>	648d19a03c6...	lfgd@gmail.co...	(hidden)	esrtdyghuj	wertygnj	false	<input type="checkbox"/>		{**:{read":tr...
<input type="checkbox"/>	648d13bd90f...	jdfnsdlf@gma...	(hidden)	WERTYU	QWERTYU	<input type="checkbox"/>			{**:{read":tr...

Рис. 3.14. Вигляд БД колекції users

<input type="checkbox"/>	_id	user pointer<_users>	title string	description string	email string	dueDate date	group pointer<groups>	IsDone boolean	acl
<input type="checkbox"/>	648d42a23c6...	648d41f490fe...	Купити квитки	25.07 11:25 Л...		2023-06-30 2...	648d42603c6...		{**:{*read":tr...
<input type="checkbox"/>	648d41333c6...	648d40b290f...	Купити квитки	Автобус 19.07...		2023-06-30 2...	648d40fa3c62...		{**:{*read":tr...
<input type="checkbox"/>	648d373e3c6...	648d37093c6...	rtul	ryujk		2023-06-18 2...	648d372f3c6...	true	{**:{*write":tr...
<input type="checkbox"/>	648d1e0c90fe...	648d1b063c6...	апрондж	апронд		2023-06-17 2...	648d1d2790f...	true	{**:{*write":tr...
<input type="checkbox"/>	648d19ca90fe...	648d19a03c6...	qwrqwr	qweqwdasfa		2023-06-18 2...	648d19b190f...		{**:{*read":tr...
<input type="checkbox"/>	648d19c190fe...	648d19a03c6...	qqqq			2023-06-17 2...	648d19b190f...	true	{**:{*write":tr...
<input type="checkbox"/>	648d19b890f...	648d19a03c6...	serdfyghul			2023-06-17 2...	648d19b190f...	false	{**:{*write":tr...
<input type="checkbox"/>	648d195e90f...	648d13bd90f...	вапронд	вапрондл		2023-06-30 2...	648d13cc90fe...	false	{**:{*write":tr...
<input type="checkbox"/>	648d171a90f...	648d13bd90f...	енгшлд	івкеанпрг		2023-06-18 2...	648d170f90fe...	true	{**:{*write":tr...
<input type="checkbox"/>	648d15f090fe...	648d13bd90f...	ієнпш	5кебншщ		2023-06-17 2...	648d13cc90fe...	false	{**:{*write":tr...
<input type="checkbox"/>	648d141790f...	648d13bd90f...	sdfghjkl	dfghjkl		2023-06-17 2...	648d13cc90fe...	true	{**:{*write":tr...

Рис. 3.15. Вигляд БД колекції Duties

<input type="checkbox"/>	_id	admin pointer<_users>	name string	members array	acl
<input type="checkbox"/>	648d42603c6...	648d41f490fe...	Львів - Київ	[{"userid":"64...	{**:{*write":tr...
<input type="checkbox"/>	648d40fa3c62...	648d40b290f...	Подорож		{**:{*write":tr...
<input type="checkbox"/>	648d372f3c6...	648d37093c6...	trest		{**:{*write":tr...
<input type="checkbox"/>	648d20c690fe...	648d20be90f...	travel		{**:{*write":tr...
<input type="checkbox"/>	648d1d2790f...	648d1b063c6...	кенол		{**:{*write":tr...
<input type="checkbox"/>	648d19b190f...	648d19a03c6...	efghj		{**:{*write":tr...
<input type="checkbox"/>	648d170f90fe...	648d13bd90f...	апронл		{**:{*write":tr...
<input type="checkbox"/>	648d16163c6...	648d13bd90f...	кенанпшщз		{**:{*write":tr...
<input type="checkbox"/>	648d13cc90fe...	648d13bd90f...	ertyhjk		{**:{*write":tr...

Рис. 3.16. Вигляд БД колекції Groups

<input type="checkbox"/>	_id	group pointer<groups>	email string	acl
<input type="checkbox"/>	648d40fa03f5...	648d40fa3c62...	companion@...	{**:{*read":tr...
<input type="checkbox"/>	648d372f03f5...	648d372f3c6...	qwerty@gmal...	{**:{*read":tr...
<input type="checkbox"/>	648d19b133ff...	648d19b190f...	sdfghjkl@gma...	{**:{*read":tr...
<input type="checkbox"/>	648d161733ff...	648d16163c6...	ojo@gmail.co...	{**:{*read":tr...
<input type="checkbox"/>	648d161633ff...	648d16163c6...	sdfjn@gmail.c...	{**:{*read":tr...

Рис. 3.17. Вигляд БД колекції Notifications

<input type="checkbox"/>	<code>_id</code>	<code>group</code> pointer<groups>	<code>user</code> pointer<_users>	<code>acl</code>
<input type="checkbox"/>	648d42b33c6...	648d42603c6...	648d40b290f...	{**:{write":tr...

Рис. 3.18. Вигляд БД колекції Group member

3.3. Збірка і встановлення

3.3.1. Створення арк-файлу

Одним із останніх кроків розробки додатку є його збірка у файл формату арк для встановлення додатку на мобільному пристрої.

Для створення арк-файлу, виконуються такі кроки:

- Перевірка коректності залежностей, бібліотек та файлів конфігурації;
- Використовується автоматична збірка Gradle (файл build.gradle);
- Створюється цифровий підпис додатку ("Build" -> "Generate Signed Bundle/APK" у Android Studio);
- Обирається режим збірки;
- Запускається збірка.

Результатом збірки є створений файл розширення .apk. Цей файл є готовим для встановлення на мобільному пристрої.

3.3.2. Завантаження додатку на смартфон

Для перевірки коректності роботи додатка, завантажимо його версію на смартфон та встановимо.

Дій для встановлення зображено на рис. 3.19-3.21.

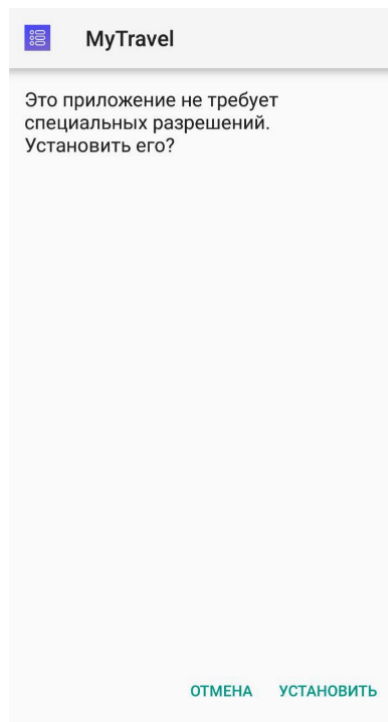


Рис. 3.19. Перший етап встановлення додатка на смартфон

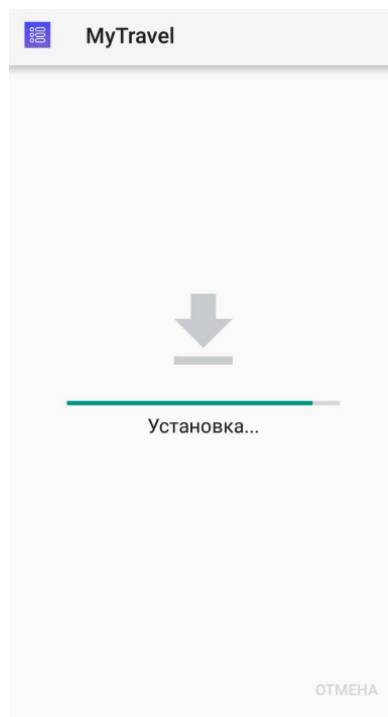


Рис. 3.20. Другий етап встановлення додатка на смартфон

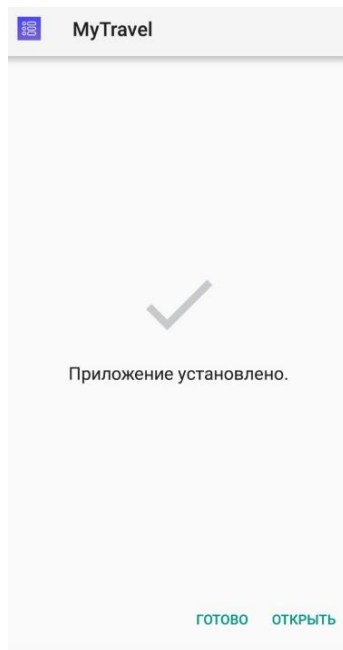


Рис. 3.21. Третій етап встановлення додатка на смартфон

3.3.3. Робота додатку

Виконаємо дослідження додатку на розроблений функціонал. Зайдемо в додаток і виконаємо реєстрацію особистого кабінету: заповнимо поля для реєстрації, натиснемо кнопку «submit» (рис. 3.22 – 3.23).

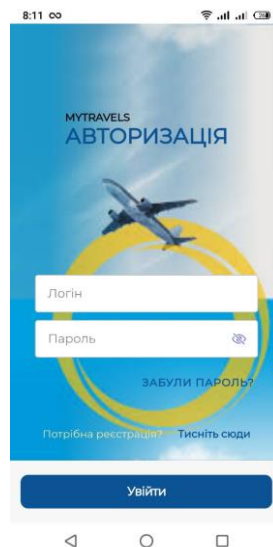


Рис. 3.22. Етап 1 дослідження додатку

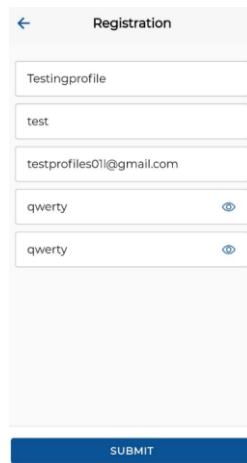


Рис. 3.23. Етап 2 дослідження додатку

Отримуємо підтвердження успішної реєстрації (рис. 3.24) і перехід на головну сторінку (рис. 3.25).

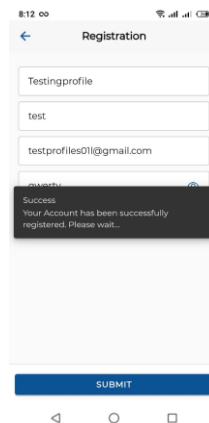


Рис. 3.24. Етап 3 дослідження додатку

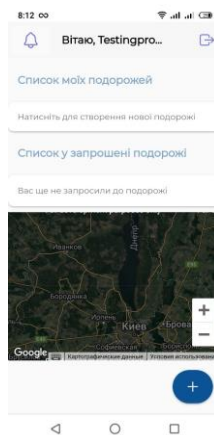


Рис. 3.25. Етап 4 дослідження додатку

Створимо нову подорож (рис. 3.26). Зайдемо у меню подорожі (рис. 3.27).

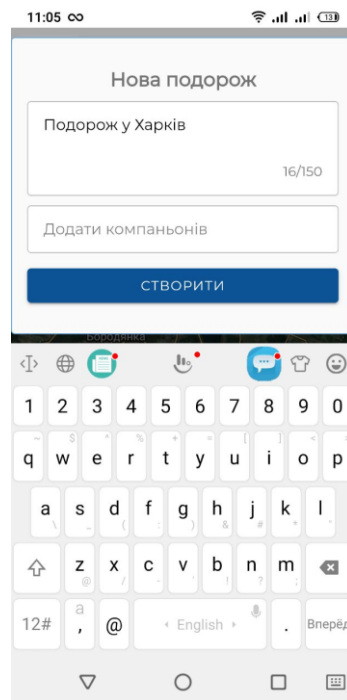


Рис. 3.26. Етап 5 дослідження додатку



Рис. 3.27. Етап 6 дослідження додатку

Створимо для подорожі «Подорож у Харків» ціль «Забронювати готель», запишемо дату поселення – 29.06 (рис. 3.28). Збережемо ціль і переглянемо успішність додавання (рис. 3.29).

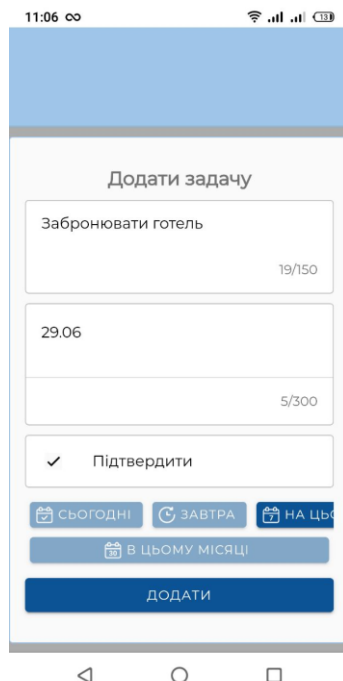


Рис. 3.28. Етап 7 дослідження додатку

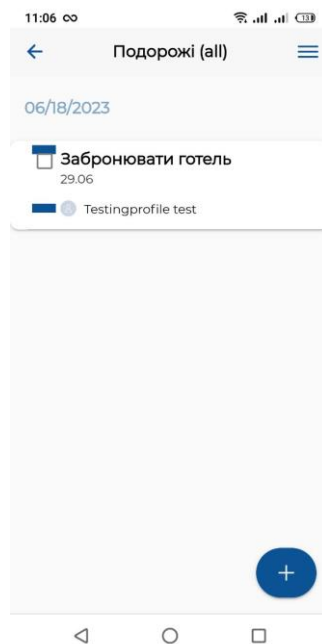


Рис. 3.29. Етап 8 дослідження додатку

Умовно виконали ціль. Видалимо ціль «Забронювати готель» (рис. 3.30 – 3.31).

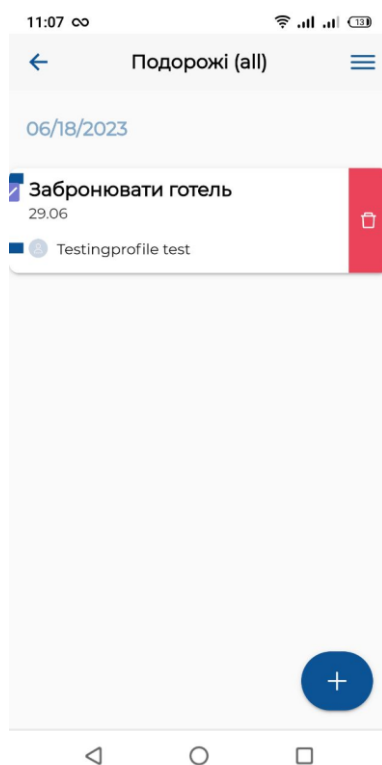


Рис. 3.30. Етап 9 дослідження додатку

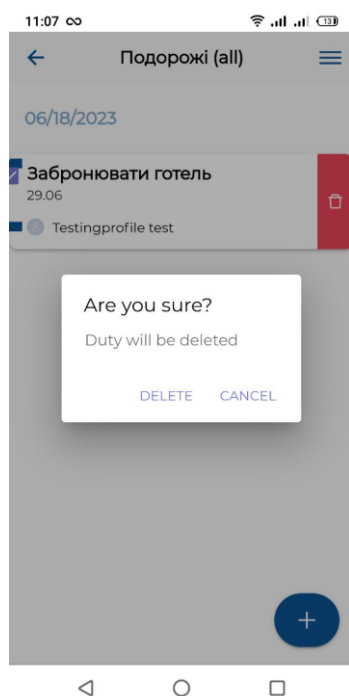


Рис. 3.31. Етап 10 дослідження додатку

Умовно інший користувач створив подорож і відправив нам запрошення (рис. 3.32). Прийmemo запрошення, додамо ціль «Дізнатися номер потягу» і переглянемо успішність створення цілі (рис. 3.33 – 3.34).

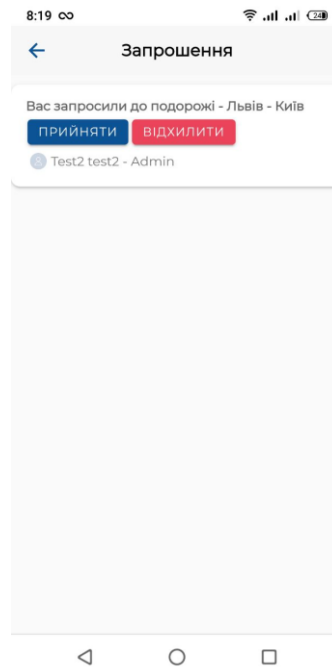


Рис. 3.32. Етап 11 дослідження додатку

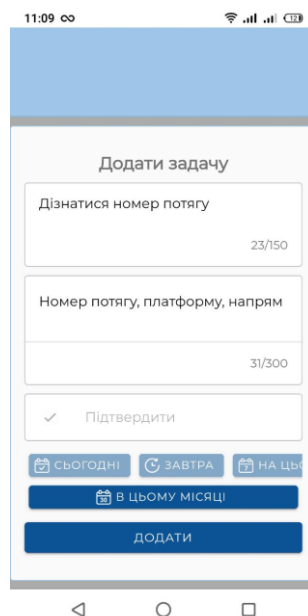


Рис. 3.33. Етап 12 дослідження додатку

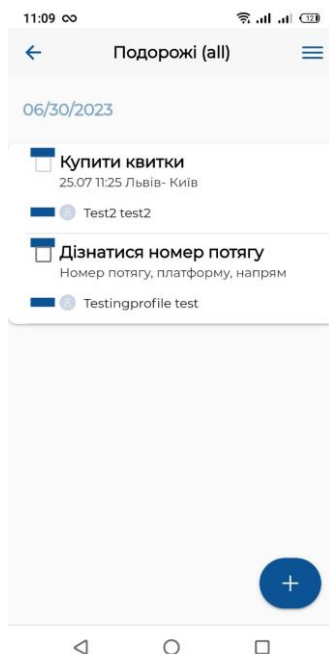


Рис. 3.34. Етап 13 дослідження додатку

Умовно виконали ціль «Купити квитки» - видалимо виконану ціль. (рис. 3.35 – 3.36). Перевіримо успішність видалення цілі (рис. 3.37).

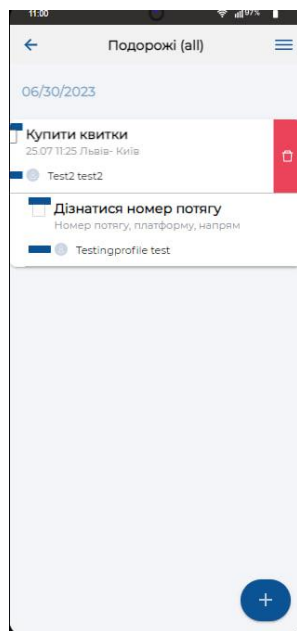


Рис. 3.35. Етап 14 дослідження додатку

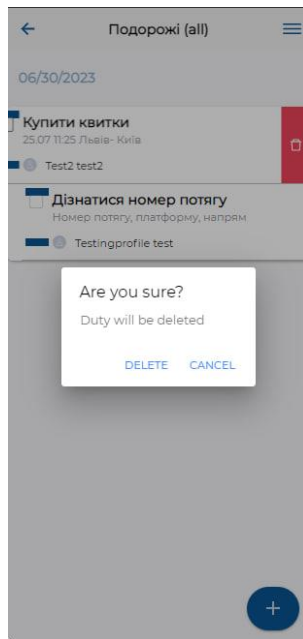


Рис. 3.36. Етап 15 дослідження додатку

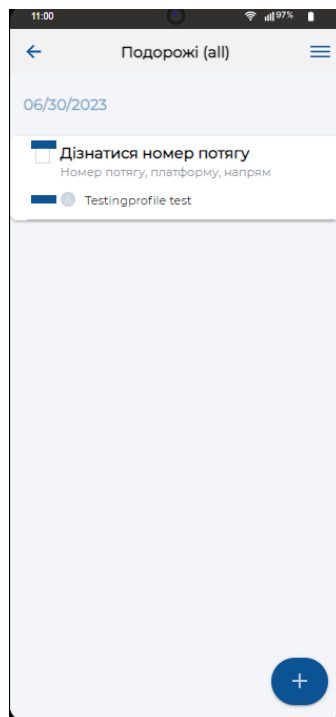


Рис. 3.37. Етап 16 дослідження додатку

Умовно змінилися плани – змінився маршрут шляху, а доданий користувач більше не приймає участь у колективній подорожі. Змінимо опис подорожі на «Львів – Житомир – Київ» (рис. 3.38 – 3.39) і видалимо користувача з подорожі (рис. 3.40).

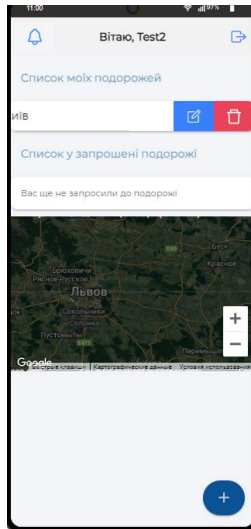


Рис. 3.38. Етап 17 дослідження додатку

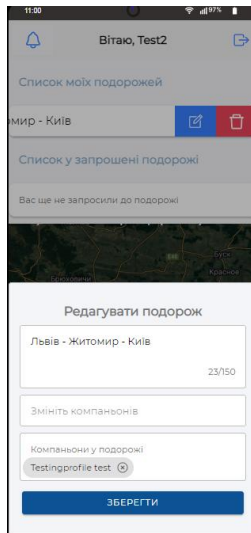


Рис. 3.39. Етап 18 дослідження додатку

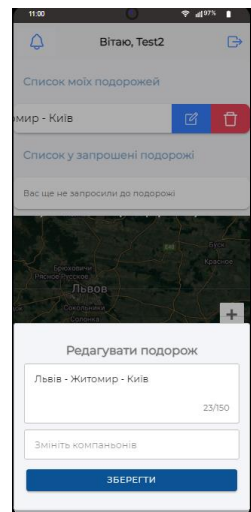


Рис. 3.40. Етап 19 дослідження додатку

Змінений список подорожей у раніше доданого користувача до спільної подорожі зображено на рис. 3.41.

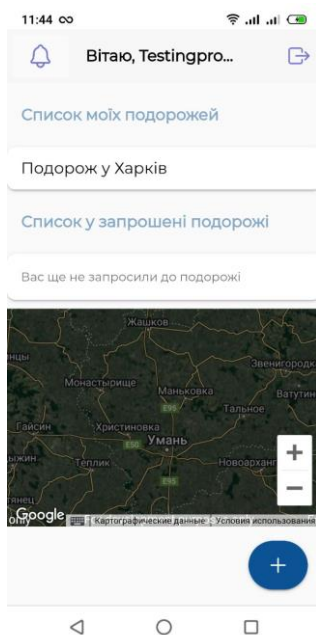


Рис. 3.41. Етап 20 дослідження додатку

Виконаємо вихід з особистого кабінету (рис. 3.42).

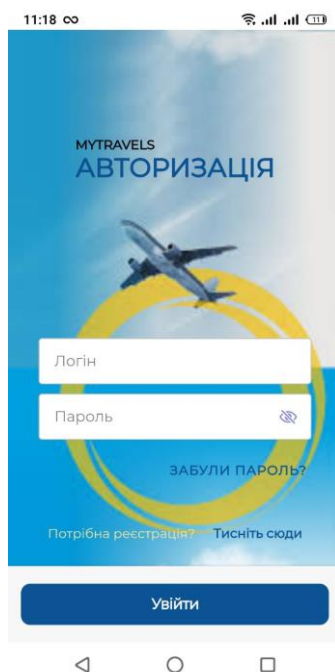


Рис. 3.42. Етап 21 дослідження додатку

3.4. Висновок до 3 розділу

В розділі 3 було проведено модульну розробку додатку для планування подорожей, було створено функціонал реєстрації, авторизації, відновлення паролю, створення, редагування і керування подорожами, цілями, користувачами та google мапою. Розроблені модулі включають функціональність для контролю планування нових маршрутів подорожей, редагування існуючих та зберігання даних БД.

Було зібрано арк файл для встановлення на мобільному пристрої додатку і проведено дослідження функціональності шляхом мануального тестування для забезпечення якості і працездатності.

Для розгортання ПЗ, були використані відповідні інструменти та сервіси, що дозволяють ефективно розгорнути додаток на обраній платформі.

В розділі 3 також було надано опис інтерфейсу системи, де було визначено основні елементи та їх взаємодію у додатку.

Загальний висновок з розділу 3 підтверджує успішне завершення розробки готового цілісного android-додатку, а також проведення тестування та розгортання програмного забезпечення. Розроблені модулі та проведені дослідження дозволяють забезпечити функціональність та стабільність, що сприяє задоволенню користувачів та ефективному використанню додатку для планування подорожей.

ВИСНОВКИ

У кваліфікаційній роботі було проведено аналіз предметної області планування подорожей та визначено основні вимоги до розроблюваного мобільного Android-додатку. Було проведено аналіз відомих програмних продуктів, таких як Triplt, PackPoint, Booking, Airbnb та Couchsurfing, для оцінки їхніх функціональних можливостей та особливостей інтерфейсу.

На основі аналізу було сформульовано задачі розробки мобільного додатку для планування подорожей, які включають створення та зберігання маршрутів, пошук та бронювання готелів, оренду житла, підбір туристичних атракцій та подій, а також обмін інформацією з іншими користувачами.

Для реалізації додатку було обрано мову програмування Java та середовище розробки Android Studio. Java є широко використовуваною мовою для розробки Android-додатків, а Android Studio надає потужні інструменти для розробки, тестування та налагодження додатків.

В процесі розробки було надано основні модулі для створення та зберігання маршрутів подорожей, а також було проведено тестування додатку на різних рівнях, включаючи передачу даних, інтерфейс, функціонал, безпеку та стабільність програмного забезпечення.

Описано інтерфейс системи, в якому визначено основні елементи та їх взаємодію у додатку, що сприяє зручній та ефективній взаємодії користувача з додатком.

Загальною метою роботи було створення мобільного додатку для планування подорожей на платформі Android. Результатом роботи є розроблений додаток з виконаними задачами, що задовольняють вимоги до функціональності, ефективності, безпеки та зручності використання. Додаток надає користувачам можливість зручно та ефективно планувати свої подорожі, знаходити та бронювати житло, а також обмінюватись інформацією з іншими користувачами.

Результати роботи можуть бути використані як основа для подальшого розвитку та вдосконалення додатку, а також для подальших досліджень у галузі планування подорожей та розробки мобільних додатків.

СПИСОК БІБЛІОГРАФІЧНИХ ПОСИЛАНЬ ВИКОРИСТАНИХ

ДЖЕРЕЛ

1. Роджерс Рік Android. Розробка додатків / Рік Роджерс Джон Ломбардо – М.: ЕКОМ Паблішерс, 2010. – 400 с.
2. Гаркуша Н.М. Моделі і методи прийняття рішень в аналізі та аудиті: навч. посіб. / Н.М. Гаркуша, О.В. Цуканова, О.О. Горошанська; — Київ, 2012. — 591 с.
3. Програми та додатки для Android [Електронний ресурс] – Режим доступу: <http://android-phones.ru/> (дата звернення 15.05.2023р.) – Назва з екрану.
4. Android Emulator [Електронний ресурс] – Режим доступу: <http://developer.android.com/tools/help/emulator.html>. (дата звернення 17.05.2023р.) – Назва з екрану.
5. Google Android [Електронний ресурс] – Режим доступу: <http://www.androidtalk.ru/google-android/> (дата звернення 20.05.2023р.) – Назва з екрану.
6. Все про Android [Електронний ресурс] – Режим доступу: <http://androiddocs.ru/> (дата звернення 21.05.2023р.) – Назва з екрану.
7. Архітектура Android-додатків [Електронний ресурс] – Режим доступу: <http://habrahabr.ru/post/141201/> (дата звернення 22.05.2023р.) – Назва з екрану.
8. Голощапов А. Л. Google Android: програмування для мобільних пристроїв / Голощапов А. Л. – СПб.: БХВ-Петербург, 2011. – 448 с.
9. Гарнаєв. А. WEB-програмування на Java та JavaScript / А. Гарнаєв , С. Гарнаєв. - М.: БХВ-Петербург, 2012. - 179 с.
10. Мобільна платформа Android [Електронний ресурс] – Режим доступу: <http://itc.ua/articles/mobilnaya-platforma-android-pyat-let-istorii/> (дата звернення 25.05.2023р.) – Назва з екрану.