

**Міністерство освіти і науки України
Луцький національний технічний університет
Факультет комп'ютерних та інформаційних технологій
Кафедра інженерії програмного забезпечення**

**КВАЛІФІКАЦІЙНА РОБОТА
ЗА СТУПЕНЕМ ВИЩОЇ ОСВІТИ «БАКАЛАВР»**

**РОЗРОБКА ANDROID-ДОДАТКУ ДЛЯ РЕКОМЕНДАЦІЇ
ТУРИСТИЧНИХ НАПРЯМКІВ З ВИКОРИСТАННЯМ
KOTLIN TA FIREBASE**

**DEVELOPMENT OF AN ANDROID APPLICATION FOR
RECOMMENDING TRAVEL DESTINATIONS USING KOTLIN
AND FIREBASE**

спеціальність 121 «Інженерія програмного забезпечення»
освітня програма «Інженерія програмного забезпечення»

Виконав: здобувач вищої освіти
Групи ІПЗ-42
Кінах Андрій Ігорович

Керівник:
Повстяна Юлія Славомирівна

Кваліфікаційну роботу
допущено до захисту
«10» 06 2025 р.
к.т.н., доцент
Гарант освітньої програми:
к.т.н., доцент
Ліщина Наталія Миколаївна

(підпис)

Луцьк – 2025 року

ЛУЦЬКИЙ НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ

Факультет комп'ютерних та інформаційних технологій

Кафедра інженерії програмного забезпечення

Ступінь вищої освіти: бакалавр

Галузь знань: 12 Інформаційні технології

Спеціальність: 121 «Інженерія програмного забезпечення»

Освітня програма: «Інженерія програмного забезпечення»

ЗАТВЕРДЖУЮ

Завідувач кафедри



«20» 12 2024 р.

ЗАВДАННЯ НА КВАЛІФІКАЦІЙНУ РОБОТУ ЗДОБУВАЧУ ВИЩОЇ ОСВІТИ

Кінаху Андрію Ігоровичу
(прізвище, ім'я, по батькові)

1. Тема кваліфікаційної роботи: «Розробка Android-додатку для рекомендації туристичних напрямків з використанням Kotlin та Firebase».

Керівник роботи: к.т.н., доцент, Повстяна Юлія Славомирівна

затверджені наказом закладу вищої освіти від «19» грудня 2024 р. № 474/01-02

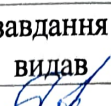
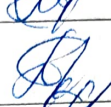
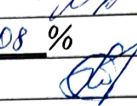
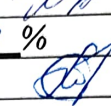
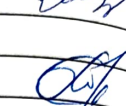
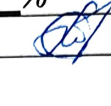
2. Строк подання здобувачем вищої освіти кваліфікаційної роботи «10» червня 2025 р.

3. Вихідні дані до роботи: технічне та програмне забезпечення, вимоги до розробки програмного забезпечення, методичні вказівки до виконання кваліфікаційної роботи бакалавра.

4. Зміст розрахунково-пояснювальної записки (перелік питань, що потрібно розробити): аналіз сучасного стану проблеми, існуючих методів і засобів її розв'язання, аналіз, визначення вимог до розроблюваного програмного забезпечення та проектування програмного забезпечення, опис функціонального наповнення об'єкта проектування, практична реалізація об'єкта проектування.

5. Перелік графічного матеріалу: 20 рисунків

6. Консультанти розділів роботи

Розділ	Прізвище, ініціали та посада консультанта	Підпис	
		завдання видав	завдання прийняв
Аналіз предметної області	Повстяна Ю. С.		
Специфікації вимог до розробленої системи	Повстяна Ю. С.		
Розробка об'єкта проектування	Повстяна Ю. С.		
Нормоконтроль	Вознюк А. В.		
Гарант ОП	Ліщина Н. М.		
Показник запозичень тексту	2,08 %		
Академічна доброчесність	Повстяна Ю. С.		

7. Дата видачі завдання «20» грудня 2024 р.

№	Назва етапів кваліфікаційної роботи бакалавра	Строк виконання етапів роботи	Примітка
1	Огляд літературних джерел по темі кваліфікаційної роботи бакалавра	до 04.02.2025 р.	
2	Аналіз проблеми розробки та впровадження об'єкту проектування	до 01.03.2025 р.	
3	Обґрунтування вибору шляхів, технологій (алгоритмів) і засобів вирішення поставленого завдання	до 15.03.2025 р.	
4	Розробка функціонально-структурної схеми роботи об'єкта проектування та проектування бази даних	до 29.03.2025 р.	
5	Практична реалізація об'єкта проектування та розробка бази даних	до 26.04.2025 р.	
6	Тестування та налагодження об'єкта проектування	до 03.05.2025 р.	
7	Здача чистового варіанту кваліфікаційної роботи бакалавра на кафедрі	до 10.06.2025 р.	

Здобувач вищої освіти



Андрій КІНАХ

Керівник кваліфікаційної роботи



Юлія ПОВСТЯНА

АНОТАЦІЯ

Кінах А. І. Розробка Android-додатку для рекомендації туристичних напрямків з використанням Kotlin Та Firebase. Рукопис. Кваліфікаційна робота бакалавра ОП «Інженерія програмного забезпечення» спеціальності «Інженерія програмного забезпечення». Луцький національний технічний університет. Луцьк, 2025.

Кваліфікаційна робота бакалавра складається зі вступу, трьох розділів, висновків, списку використаних джерел.

У першому розділі здійснено аналіз сучасного стану проблеми розробки рекомендації туристичних напрямків і сформульовано завдання. В другому розділі визначено вимоги до розроблюваної системи, а також засоби, методи і алгоритми розробки. У третьому розділі розроблено вебдодаток для рекомендації туристичних напрямків. У висновках узагальнено інформацію, відображену у попередніх частинах.

Ключові слова: Android-додаток, тур, карта, реєстрація.

ABSTRACT

Kinakh A. Development of an Android Application for Recommending Travel Destinations Using Kotlin and Firebase. Manuscript. Qualification work of the bachelor's program "Software engineering" in the specialty "Software engineering". Lutsk National Technical University. Lutsk, 2025.

The bachelor's qualification thesis consists of an introduction, three chapters, conclusions and recommendations, and a list of references.

The first chapter provides an analysis of the current state of the problem of developing tourist destination recommendation systems and formulates the objectives. The second chapter defines the requirements for the system being developed, as well as the tools, methods, and algorithms used in the development. In the third chapter, an Android application for recommending tourist destinations was developed. The conclusions summarize the information presented in the previous sections.

Keywords: Android application, tour, map, registration.

ЗМІСТ

ВСТУП	7
РОЗДІЛ 1 АНАЛІЗ ДОДАТКІВ ДЛЯ РЕКОМЕНДАЦІЇ ТУРИСТИЧНИХ НАПРЯМКІВ І ПОСТАНОВКА ЗАВДАНЬ НА КВАЛІФІКАЦІЙНУ РОБОТУ	10
1.1 Аналіз сучасного стану проблеми	10
1.2 Постановка завдання на кваліфікаційну роботу бакалавра	19
РОЗДІЛ 2 СПЕЦИФІКАЦІЯ ВИМОГ ДО РОЗРОБЛЮВАНОЇ СИСТЕМИ	21
2.1 Аналіз, визначення вимог до розроблюваного програмного забезпечення та проектування програмного забезпечення	21
2.2 Вибір засобів, методів і алгоритмів вирішення поставленого завдання	26
РОЗДІЛ 3 РОЗРОБКА ANDROID-ДОДАТКУ ДЛЯ РЕКОМЕНДАЦІЇ ТУРИСТИЧНИХ НАПРЯМКІВ З ВИКОРИСТАННЯМ KOTLIN ТА FIREBASE	30
3.1 Практична реалізація об'єкта проектування	30
3.2 Тестування та налагодження інформаційно-комп'ютерної системи	39
3.3 Розробка бази даних	42
3.4 SEO інформаційно-комп'ютерної системи	44
ВИСНОВКИ	46
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ	48

ВСТУП

Актуальність теми. Туризм став невід’ємною частиною життя людини. Якщо людина подорожує не одна, вона дуже часто чує та дізнається про пам’ятки від інших людей, які відвідали ці місця, або використовуючи різні засоби масової інформації на розвиток туризму в інших країнах. Коли людина вирішує відвідати нове місце, вона зазвичай не знає з чого почати дослідження, але цю проблему можна легко вирішити за допомогою мобільних пристроїв, як компаньйон, мобільний додаток може показати користувачеві найкращі туристичні напрямки, які можуть бути самотніми не знайшов би іншими шляхами. Простота Інтернету дозволяє мандрівникам знаходити цікаву інформацію та планувати свою діяльність. За останнє десятиліття відбулася революція в інформаційно-комунікаційних технологіях, яка не тільки змінює повсякденну поведінку людей, але й впливає на те, як вони взаємодіють один з одним.

Сучасні мобільні додатки відіграють ключову роль у сфері туризму, надаючи користувачам можливість швидкого пошуку та вибору цікавих місць для подорожей. Зростання популярності персоналізованих рекомендаційних систем обумовлює необхідність розробки ефективних програмних рішень, здатних аналізувати вподобання користувачів та надавати відповідні туристичні напрямки. У цьому контексті особливий інтерес викликає розробка мобільних додатків для платформи Android, оскільки вона є однією з найпоширеніших серед користувачів.

Одним із ключових підходів до створення таких систем є використання мови програмування Kotlin, яка забезпечує високу продуктивність та гнучкість у розробці Android-застосунків. Крім того, інтеграція з хмарною платформою Firebase дозволяє реалізувати надійне збереження даних, синхронізацію інформації в реальному часі та ефективне управління користувацькими запитами.

У процесі розробки мобільного додатка важливим аспектом є розробка алгоритмів персоналізованих рекомендацій, що базуються на аналізі вподобань користувачів. Використання Firebase дає змогу зберігати дані про взаємодію користувачів із додатком, їхні оцінки та вибрані категорії туристичних місць, що сприяє формуванню якісних рекомендацій.

Запропонований Android-додаток буде мати зручний інтерфейс, що дозволить користувачам переглядати туристичні місця, додавати оцінки та отримувати персоналізовані рекомендації відповідно до їхніх інтересів. Для покращення взаємодії система надаватиме можливість фільтрації результатів за певними критеріями, такими як популярність, фотогенічність або історична цінність об'єкта.

Розробка такого додатку сприятиме підвищенню ефективності планування подорожей та допоможе користувачам знаходити нові, цікаві місця відповідно до їхніх уподобань. Застосування сучасних технологій дозволить забезпечити масштабованість додатка та його подальший розвиток.

Метою даної кваліфікаційної роботи бакалавра є розробка Android-додатку для рекомендації туристичних напрямків із використанням Kotlin та Firebase, що забезпечує персоналізований підхід до вибору місць для подорожей на основі вподобань користувачів.

Об'єктом кваліфікаційної роботи бакалавра є процес розробки Android-додатку для рекомендації туристичних напрямків.

Предметом кваліфікаційної роботи бакалавра є методи та технології створення Android-додатку, що базуються на використанні Kotlin та Firebase.

Для досягнення поставленої мети необхідно виконати такі завдання:

- провести аналіз існуючих рішень у сфері мобільних рекомендаційних систем для туризму;
- визначити основні вимоги до мобільного додатка та розробити його архітектуру;
- реалізувати функціональність додатка, що включає персоналізовані рекомендації на основі категорій, оцінок і фільтрів;

- інтегрувати Firebase для збереження користувацьких даних і забезпечення реальної синхронізації інформації;
- провести тестування розробленого додатка для перевірки його коректної роботи та зручності використання;
- проаналізувати результати тестування та запропонувати шляхи подальшого вдосконалення системи.

Дана кваліфікаційна робота бакалавра спрямована на створення ефективного мобільного рішення, що допоможе туристам швидко знаходити актуальні рекомендації для подорожей відповідно до їхніх вподобань.

РОЗДІЛ 1

АНАЛІЗ ДОДАТКІВ ДЛЯ РЕКОМЕНДАЦІЇ ТУРИСТИЧНИХ НАПРЯМКІВ І ПОСТАНОВКА ЗАВДАНЬ НА КВАЛІФІКАЦІЙНУ РОБОТУ

1.1 Аналіз сучасного стану проблеми

Бізнес і застосування інформаційно-комунікаційних технологій (ІКТ) поступово покращується завдяки розробці електронних додатків, таких як електронна комерція, електронне навчання, інтернет-культура та путівники з туризму, інтернет-ігри, електронне здоров'я та електронне зондування та збір даних середовища з середини 1990-х років. З бездротовими технологіями та мобільними пристроями, нова тенденція мобільних додатків виникла відповідно до їх електронної попередників, які включають мобільну комерцію. Зростання використання мобільних телефонів і нові розробки в області мобільних комп'ютерів комунікаційні мережі (наприклад, GPS, Wi-Fi) пропонують найсучасніші вдосконалення системи рекомендацій у сфері туризму.

Люди пристосувалися до нових технологій і можуть використовувати пристрої інформаційних технологій (ІТ) не лише у повсякденному житті, а й під час подорожей. Розумний туризм видається логічним переходом від традиційного туризму до інноваційного підходу, коли споживачі покладаються на широке впровадження ІКТ у туризмі. Залучення інтелектуальності до туристичних напрямків, що динамічно об'єднує зацікавлених сторін через технологічну платформу, на якій можна миттєво обмінюватися інформацією, що стосується туристичної діяльності. Такий розвиток приносить користь індустрії туризму, забезпечуючи легкий доступ до інформації як для туристичних організацій, так і для туристів через інтегровані та централізовані платформи даних. Туризм на сьогоднішній день є актуальною темою, тому що він є чудовим джерелом доходу для будь-якої людини. Однак туризм є

сезонним заходом через відмінність клімату і погоди в усьому світі, а також через те, що туристи рідні має традиційні свята та пори року.

Мобільні програми є ефективним способом для туристичних компаній будувати лояльні стосунки з клієнтами. У сучасному бізнес-середовищі мобільні програми допомагають компаніям, пов'язаним з подорожами, надавати захоплюючий, цікавий та інноваційний досвід своїм клієнтам. У туристичній індустрії мобільні маркетингові пристрої допомагають споживачам ідентифікувати та зрозуміти більше туристичної інформації, продуктів і послуг і менеджери з подорожей часто використовують мобільні пристрої для розробки маркетингових стратегій для постачальників. Мобільні додатки приносять величезну користь споживачам, але насправді було дуже мало досліджень про те, які фактори можуть впливати на використання споживчими мобільних додатків; хоча мобільні програми викликають багато інтересів у багатьох людей. Щоб краще дослідити залучення споживачів до використання мобільних додатків, проблема полягає в тому, як зрозуміти різницю між поведінкою та прийняттям. Разом з розвитком інформаційних технологій (ІТ) туристи сьогодні часто використовують мобільні додатки, щоб вибрати туристичні напрямки як найефективніший спосіб. Тож різні чинники впливають на ставлення користувачів до використання мобільних додатків і бажання відвідати туристичні місця.

Мобільні рекомендаційні системи були дуже корисними з точки зору допомоги туристам прийняття рішень або отримання інформації.

Користувачі мобільних пристроїв погоджуються використовувати програми, коли виконують свою першу мобільну діяльність. Якщо говорити точніше, споживачі приймають програми, коли їх завантажують. Збільшення кількості людей, які користуються програмами для телефону, означає, що кількість людей, які приймають програми, більша, ніж тих, хто не приймає.

Пошук он-лайн інформації про туристичні напрямки є одним із найпопулярніших процесів, від якого туристи отримують все більше переваг. В останні роки спостерігається зростання використання онлайн-звітів про

туристичні напрямки та їх планування. Індустрія туризму – одна з найбільших і розвинених галузей у світі, яка приносить досить високий відсоток доходу туристичним країнам.

Туристи все частіше використовують інтелектуальні програми на своїх мобільних телефонах під час подорожей до туристичних місць, таким чином революціонізуючи спосіб подорожі та взаємодії з напрямками та різними агентами, які їх складають. Це означає, що ці технології мають значний вплив на такі напрямки, які зараз перебувають у процесі цифрової трансформації. Ці технології включають ряд мобільних додатків:

- карти та навігація: мобільні програми, які надають інтерактивні карти, які допомагають користувачам знайти своє місцезнаходження та напрямки до різних цікавих місць;

- інформація в реальному часі: інтеграція служб визначення місцезнаходження та постачальників даних у реальному часі для надання актуальної інформації про дорожній рух, погоду та місцеві новини;

- покращений туристичний досвід: використання таких технологій, як доповнена реальність і аудіогіди, щоб покращити відвідування музеїв, визначних місць і туристичних місць, пропонуючи додаткову інформацію та більш захоплюючий досвід;

- додаткові послуги: програми, які включають додаткові послуги, такі як інформація про наявність зарядних станцій для електромобілів, варіанти паркування, рекомендації щодо ресторанів у місті або бронювання таксі.

Мобільні додатки для туризму допомагають оптимізувати та збагатити досвід подорожі. Мобільні туристичні послуги привернули увагу як у практиці, так і в дослідженнях.

Система в домені ESTD.GR є експертною системою для туристичних напрямків, яка співпрацює з користувачем-туристом. Створення експертної системи для вебпорталу з використанням платформи з відкритим кодом (CMS) з базою даних MySQL [1]. ESTD.GR – це веб-платформа на основі визначення місцезнаходження. Списки стають частиною подорожі (рис. 1.1). Завдяки

можливості створювати списки на основі географічного розташування платформа стає чудовим інструментом для рекомендацій туристичних напрямків, створення туристичних напрямків, блогів про подорожі, путівників тощо в поєднанні з типами користувачів-туристів і рейтингами збережених місць у зірках. Основна ідея запропонованої системи полягає в тому, щоб отримати найкраще оптимізоване персоналізоване туристичне місце з бази даних, створеної користувачами.

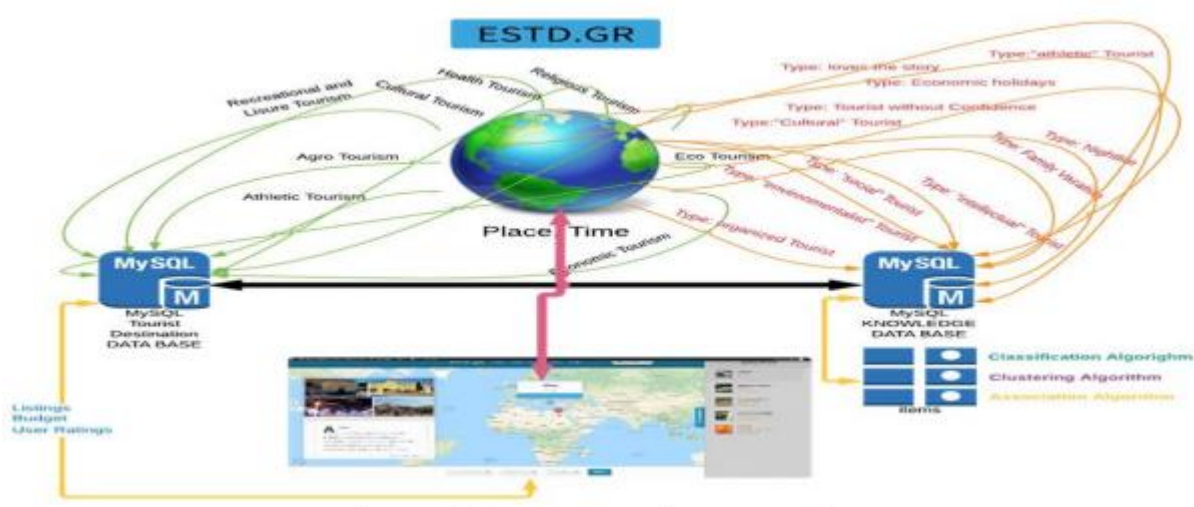


Рисунок 1.1 – Платформа туристичних рекомендацій [1]

У документі [2] представлено програму електронного туризму для підтримки туристів у регіоні. Додаток рекомендує туристичні об'єкти, які йому або їй цікаві, виходячи з туристичних уподобань і поточної ситуації в регіоні. Пам'ятки та їхні описи та зображення взяті з доступних Інтернет-джерел (наприклад, Wikipedia). Вони ранжируються спеціальним сервісом рекомендацій програми. Рекомендації базуються на оцінках туристів, які користуються додатком. У статті описано архітектуру додатків на основі послуг, онтологію для взаємодії інтелектуальних мобільних сервісів туристичних гідів та оцінку. Розроблений додаток доступний для завантаження в Google Play Market для користувачів пристроїв Android.

Додаток Visit Ukraine розроблено для надання актуальної інформації про туристичні місця, події та маршрути по всій Україні. Хоча конкретні технічні деталі розробки цього додатка не розкриті публічно, відомо, що організація Visit Ukraine активно використовує сучасні веб-технології для підтримки свого порталу visitukraine.today. Цей портал надає інформацію чотирма мовами та містить онлайн-чати для підтримки користувачів, що свідчить про інтеграцію з системами обробки запитів та багатомовної підтримки [3].

Ukraine Travel Guide – це детальний путівник по Україні, який містить інформацію про культурні та природні пам'ятки, музеї, парки та інші цікаві місця (рис. 1.2). Додаток також надає карти міст та регіонів, що допомагає у навігації під час подорожі [4].

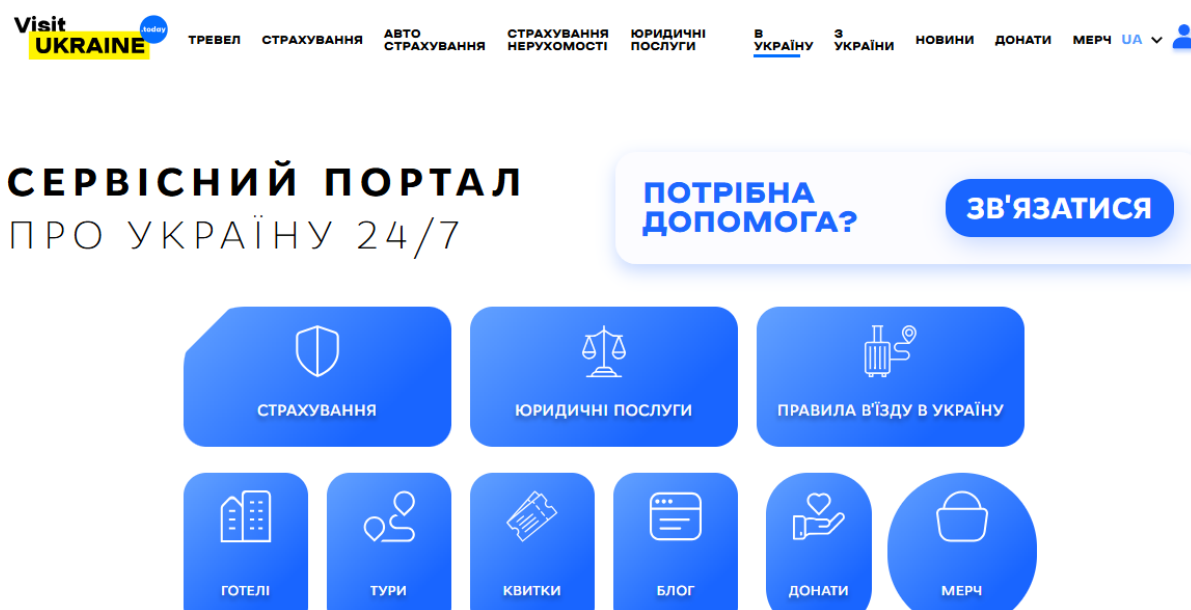


Рисунок 1.2 – Портал Ukraine Travel Guide [4]

Дія – офіційний державний додаток, який надає доступ до цифрових документів та послуг. Окрім цього, у розділі «Мандруй Україною» користувачі можуть знайти інформацію про туристичні маршрути, визначні місця та події в різних регіонах країни (рис. 1.3) [5].



Мандруй Україною: стартував проект популяризації внутрішнього туризму

#Волинь_унікальна

Завдяки ініціативі Офіса Президента України та Державного агентства розвитку туризму в особі Мар'яни Олесків у нас є унікальна можливість відкрити Волинь для всієї України та світу в цілому в нових кольорах і показати те, чого не бачили мільйони наших громадян.

Давайте зробимо це РАЗОМ.

Якщо у вашому населеному пункті або районі є цікаві місця, які є маловідомими, однак не менш мальовничими, мають

свою історію і можуть стати туристичними магнітами - просто напишіть про це, надсилайте фото локацій та короткий опис на tourism.volyn@gmail.com

Завдяки Вам, найкращі точки ми включимо до туристичної мапи Волині та загальнодержавних туристичних маршрутів.

Відвідуючи цікаві місця, відмічайте хештегом: [#МандруйУкраїною](#), [#Волинь_унікальна](#), щоб кожен міг знайти ці світлини та відвідати наш край.

Рисунок 1.3 – Проект популяризації внутрішнього туризму «Мандруй Україною» [5]

Ці додатки допоможуть вам отримати актуальну інформацію про туристичні напрямки в Україні та зробити вашу подорож комфортною та цікавою.

Нові технології, такі як штучний інтелект, можуть сприяти розробці нових додатків, які радикально змінюють досвід відвідувачів у сфері туризму та гостинності, дозволяючи створювати індивідуальні послуги та динаміку, які розробляються за допомогою технологій та/або людьми та доповнюються штучним інтелектом. Приклади застосування штучного інтелекту в туризмі включають інтелектуальні помічники в подорожах, розмовні системи та програми мовного перекладу. Це явище виникає через здатність штучного інтелекту створювати рішення, що імітують людську поведінку, використовуючи атрибути, схожі на людські характеристики, інкапсульовані в комп'ютерних алгоритмах.

BetterTravel – це платформа для планування подорожей на основі ШІ, яка створює індивідуальні маршрути відповідно до вподобань користувача (рис. 1.4). Використовуючи алгоритми машинного навчання, додаток аналізує інтереси та бюджет мандрівника, пропонуючи оптимальні варіанти розміщення, транспорту та розваг. Користувачі можуть

налаштовувати свої плани та отримувати оновлення в реальному часі, що робить подорожі більш комфортними та організованими [6].

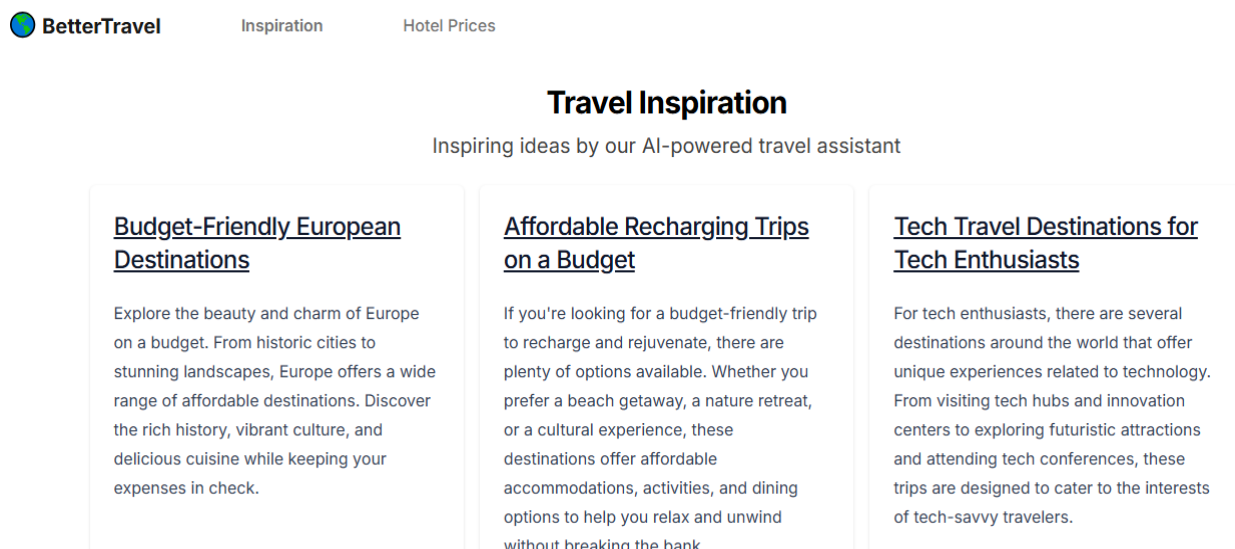


Рисунок 1.4 – Платформа «BetterTravel» [6]

Mindtrip – інноваційна туристична платформа, яка використовує великі мовні моделі та обробку природної мови для створення персоналізованих маршрутів. Користувачі можуть взаємодіяти з ШІ через інтерактивні розмови, отримуючи рекомендації щодо напрямків, заходів та місць проживання. Платформа також інтегрує візуальний контент, такий як фотографії та карти, забезпечуючи глибше розуміння запропонованих місць [7].

TripIT – це відомий додаток для планування подорожей зі штучним інтелектом, який пропонує унікальний підхід до планування подорожей, створюючи маршрути з матеріалів, наданих користувачами (рис. 1.5). Коли користувачі отримують електронні листи з підтвердженням бронювання (щодо розміщення, авіаквитків, оренди автомобілів тощо), вони можуть пересилати ці електронні листи до TripIT, який об'єднує інформацію у комплексний план подорожі. Додаток для подорожей також може синхронізуватися з календарями, щоб консолідувати всі деталі подорожі.

Окрім побудови маршруту, TripIT надає сповіщення про затримки рейсів, зміни виходу на посадку та скасування. Це збагачує досвід подорожей користувачів інформацією про певну країну та дозволяє ділитися деталями поїздки. Додаток для подорожей також має вигляд карти для візуалізації пунктів призначення, пошуку місць поблизу та зберігання документів. TripIT доступний офлайн і пропонує статистику поїздок [8].



Рисунок 1.5 – Додаток «TripIT» [8]

Fluxify – український мобільний додаток, розроблений на React Native і Python для туристів, які віддають перевагу самостійним подорожам без групових екскурсій (рис. 1.6).

Додаток дозволяє придбати тури, переглядати прев'ю маршрутів та обирати ті, що відповідають особистим вподобанням. Унікальна функція сканування будівель надає додаткову інформацію про архітектурні пам'ятки, а також дозволяє «подорожувати в часі», переглядаючи старовинні фотографії. Fluxify працює на основі власної нейронної мережі, що оптимізує використання пам'яті пристрою та забезпечує швидку обробку даних [9].



Рисунок 1.6 – Додаток «Fluxify» [9]

Ці додатки демонструють, як технології ШІ можуть покращити досвід мандрівників, надаючи їм індивідуальні рекомендації, інтерактивні гідів та зручні інструменти для планування подорожей.

Інше дослідження під назвою TravelAgent демонструє використання великих мовних моделей для створення персоналізованих маршрутів, що враховують різноманітні обмеження та вподобання користувачів. Це підкреслює потенціал ШІ у створенні гнучких та адаптивних туристичних сервісів [10].

Також варто згадати систематичний огляд мобільних туристичних додатків, який виявив, що основними розумними функціями є відстеження геолокації, генерація маршрутів подорожей та персоналізовані рекомендації в реальному часі. Однак, розробники стикаються з викликами, такими як забезпечення точності локації та конфіденційності даних користувачів [11].

Для подальшого вдосконалення мобільних туристичних додатків важливим є врахування поведінкових особливостей користувачів та аналіз їхніх уподобань у режимі реального часу. Інтеграція модулів штучного інтелекту дозволяє реалізувати персоналізовані рекомендації, які враховують

не лише геолокаційні дані, а й інтереси, історію пошуку та соціальні зв'язки користувача. Такі системи можуть автоматично підлаштовуватись під зміну погоди, локальні події або тимчасову недоступність об'єктів. Використання хмарних технологій значно підвищує масштабованість додатків і дозволяє зберігати великі обсяги інформації про туристичні об'єкти, маршрути та відгуки. Водночас важливо забезпечити надійність та безпеку персональних даних, що передаються через такі сервіси, зокрема із врахуванням вимог GDPR. Залучення користувачів до створення контенту, наприклад, через публікацію відгуків, фото та рейтингів, створює ефект «живого середовища», що стимулює нові відвідування. Додаткові можливості для розвитку мобільних додатків відкриваються завдяки доповненій реальності, яка дозволяє здійснювати віртуальні тури ще до фактичного відвідування місця. Застосування методів машинного навчання сприяє підвищенню точності прогнозів щодо популярності об'єктів у певні періоди часу. Аналіз «великих даних» у сфері туризму дозволяє виявляти приховані закономірності та формувати більш ефективні моделі поведінки користувачів. Таким чином, розробка сучасного Android-додатку з рекомендаційною системою для туристичних напрямків має спиратися на багаторівневий підхід до збору, обробки та візуалізації туристичної інформації.

1.2 Постановка завдання на кваліфікаційну роботу бакалавра

Для реалізації поставленої мети кваліфікаційної роботи необхідно вирішити низку завдань, що охоплюють як аналітичний, так і практичний аспекти розробки мобільного додатку. Ці завдання спрямовані на створення ефективного інструменту для надання персоналізованих туристичних рекомендацій, який відповідатиме сучасним вимогам зручності, надійності та інтерактивності. Зокрема, передбачається виконати такі кроки:

- провести аналіз існуючих рішень у сфері мобільних рекомендаційних систем для туризму;

- визначити основні вимоги до мобільного додатка та розробити його архітектуру;
- реалізувати функціональність додатка, що включає персоналізовані рекомендації на основі категорій, оцінок і фільтрів;
- інтегрувати Firebase для збереження користувацьких даних і забезпечення реальної синхронізації інформації;
- провести тестування розробленого додатка для перевірки його коректної роботи та зручності використання;
- проаналізувати результати тестування та запропонувати шляхи подальшого вдосконалення системи.

РОЗДІЛ 2

СПЕЦИФІКАЦІЯ ВИМОГ ДО РОЗРОБЛЮВАНОЇ СИСТЕМИ

2.1 Аналіз, визначення вимог до розроблюваного програмного забезпечення та проектування програмного забезпечення

У цій кваліфікаційній роботі бакалавра проблема великої кількості туристичного вмісту вирішується за допомогою багатьох критеріїв шляхом рекомендації. Мета – полегшити користувачам вибір туристичних напрямків на основі власних потреб тенденції. Завершена мобільна програма вимагає реєстрації користувача, а потім запитує користувача вибирає категорії, які його цікавлять, і пропонує персоналізований список на основі цих виборів туристичні напрямки. Список додатково персоналізується шляхом вибору фільтра, і таким чином створюється ранжований список користувач отримує інформацію про найкращі туристичні напрямки для його інтересів. Проводиться ранжування напрямків на основі середніх оцінок інших користувачів, які вже оцінили певний напрямок. Крім того, також є можливість переглянути пункт призначення на інтерактивній карті. Додаток створено за допомогою розробки середовищі Android Studio, використовуючи мову програмування Kotlin, і використовується для зберігання даних є Firebase.

Беручи до уваги переваги користувачів і існуючі туристичні напрямки місцевості, можливість створення рекомендацій на основі змісту, а також спільне фільтрування та ранжирування, функціональні та нефункціональні вимоги повинні бути визначені.

Функціональні вимоги – це твердження про послуги, які система повинна надавати, як система повинна реагувати на певні вхідні дані та як система повинна поводитися в певних ситуаціях. У деяких випадках функціональні вимоги можуть явно вказувати, що система не повинна робити. Функціональні вимоги часто залежать від типу програми чи послуги, що розробляється, а також залежать від середовища програмування, в якому він розроблений, а також важливо, для якого типу користувача він призначений. Вони також

визначають, що має робити система, у коротких примітках і часто написані простою мовою.

Мобільний додаток буде мати можливість входу та створення профілю за інтересами користувача, огляд і вибір мультимедіа представлених туристичних напрямків, оцінка та ранжування, багатокритеріальне рекомендування пунктів призначення. У практичній частині роботи з використанням запропонованої моделі та архітектури та відповідних мови програмування та технології (Kotlin, Firebase та інші за потреби) буде реалізовано програмно мобільний додаток з базою даних і системою створення рекомендацій.

Архітектура мобільного Android-додатку для багатокритеріального рекомендування туристичних напрямків складається з сторони користувача та сторони сервера. Серверна сторона передбачає використання двох баз даних, а саме бази даних користувачів і бази даних туристичних місць. Крім того, ця сторона він також використовує Google Maps API для функції геолокації. Логіка програми на стороні користувача це показано блок-схемою на рисунку 2.1. Технологічна схема показує послідовність використання мобільний Android-додаток для багатокритеріального рекомендації туристичних напрямків. Показано, що при під час реєстрації користувача програма зберігає дані в базі даних, а при вході в систему використовує збережені дані для автентифікації користувача. Після реєстрації користувач повинен вибрати бажані категорію, щоб потім створити список рекомендованих цільових місць, до яких буде застосовано вибрані фільтр. Користувач має можливість вибору між різними вкладками на панелі навігації, яка є показано стрілками в обох напрямках. Блок-схема також показує зв'язок з база даних туристичних місць. Дані витягуються з цієї бази даних під час відображення інформація про пункти призначення та оновлення, коли користувач призначає оцінки.

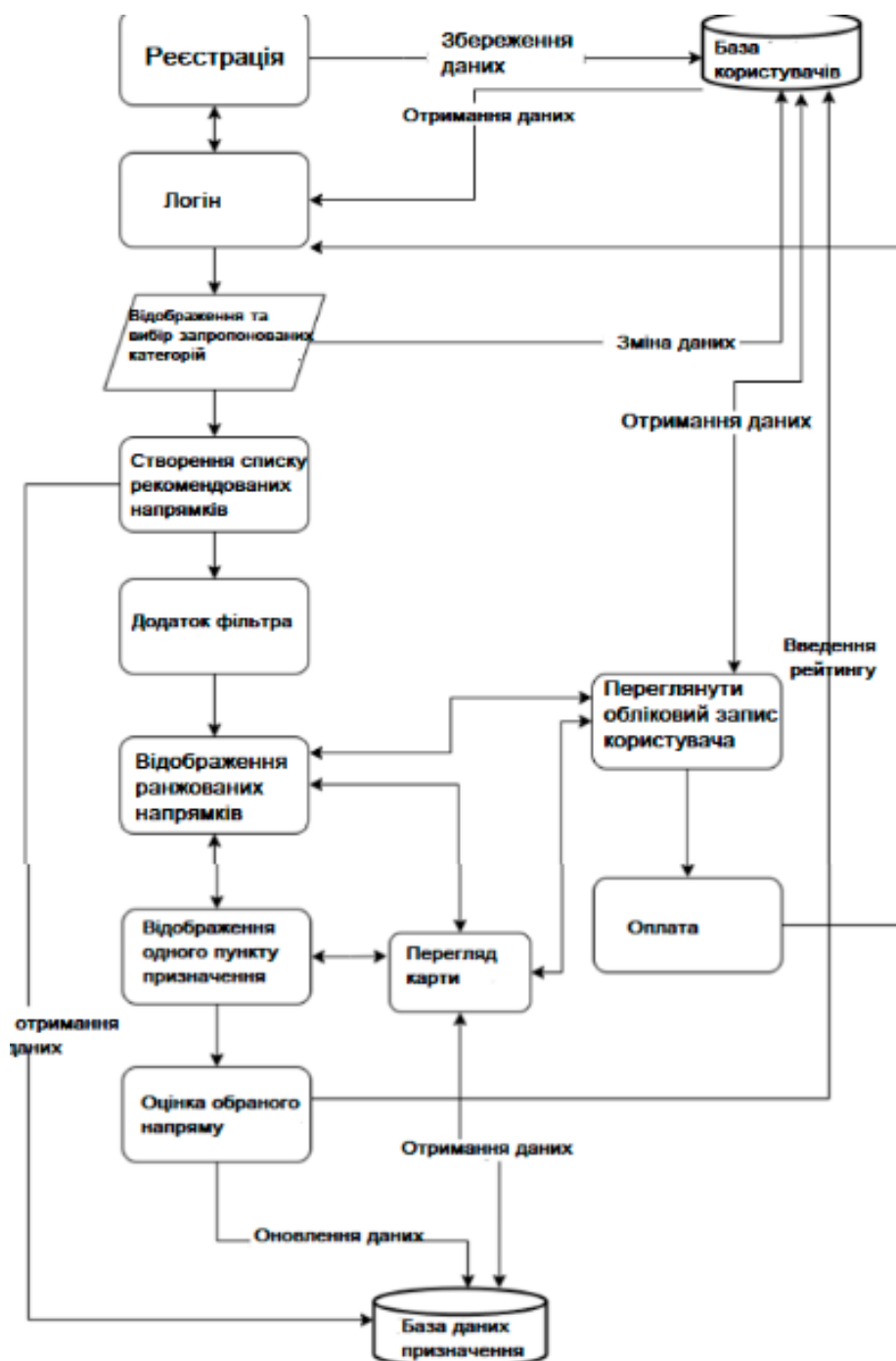


Рисунок 2.1 – Блок-схема мобільного Android-додатку для рекомендації за кількома критеріями туристичні напрямки

Для визначення вимог було використано аналіз сценаріїв використання та підхід, заснований на юзер-сторінці. Основні функціональні можливості додатку включають реєстрацію та автентифікацію користувачів, введення уподобань щодо туристичних напрямків, отримання рекомендацій, перегляд інформації про місця (фото, опис, рейтинг), збереження улюблених локацій та

використання карти для побудови маршрутів. Важливими нефункціональними вимогами є швидкодія застосунку, захист даних користувачів, підтримка автономного режиму для перегляду збережених місць та доступність на різних пристроях з Android.

Розробка Android-додатку вимагає ретельного аналізу вимог, формування моделі системи та побудови відповідних UML-діаграм. Для моделювання програмного забезпечення було використано нотацію UML. Діаграма випадків використання ілюструє основні взаємодії користувача із системою, включаючи авторизацію, перегляд рекомендацій, пошук туристичних місць та взаємодію з картою. Діаграма класів демонструє структуру даних та основні взаємозв'язки між ними: користувач, місце, рейтинг, категорія, рекомендація. Діаграма послідовності описує процес отримання рекомендацій, що включає надсилання запиту до бази даних Firebase, обробку даних та повернення списку результатів користувачеві.

Цей етап проектування включає проектування уніфікованої мови моделювання (UML), проектування інтерфейсу та проектування бази даних. Діаграма варіантів використання має на меті пояснити зв'язок між користувачем і системою, яка буде створена, тобто пояснити все, що система буде створена, і її функції. За допомогою діаграми варіантів використання можна надати огляд усіх видів використання системи, які включають систему, яка буде використовуватися для всіх і для яких функцій.

Представлена UML-діаграма класів (рис. 2.2) моделює структуру Android-додатку для рекомендації туристичних напрямків. Центральними класами є User, Place, Category, Rating та Recommendation. Клас User містить дані про користувача, його вподобання та методи взаємодії з системою (реєстрація, авторизація, оцінювання тощо). Category описує типи туристичних місць, які користувач може вибирати, а Place містить інформацію про конкретні локації. Rating пов'язує користувача з оцінками туристичних місць за різними критеріями. Клас Recommendation відповідає за збереження і логіку персоналізованих порад. Між класами реалізовано зв'язки: один користувач

має багато рейтингів та вподобаних категорій, місце має багато оцінок, а рекомендації базуються на аналізі зв'язку між користувачем і місцем. Така структура відображає логіку додатку, засновану на багатокритеріальному фільтруванні та персоналізованих пропозиціях.

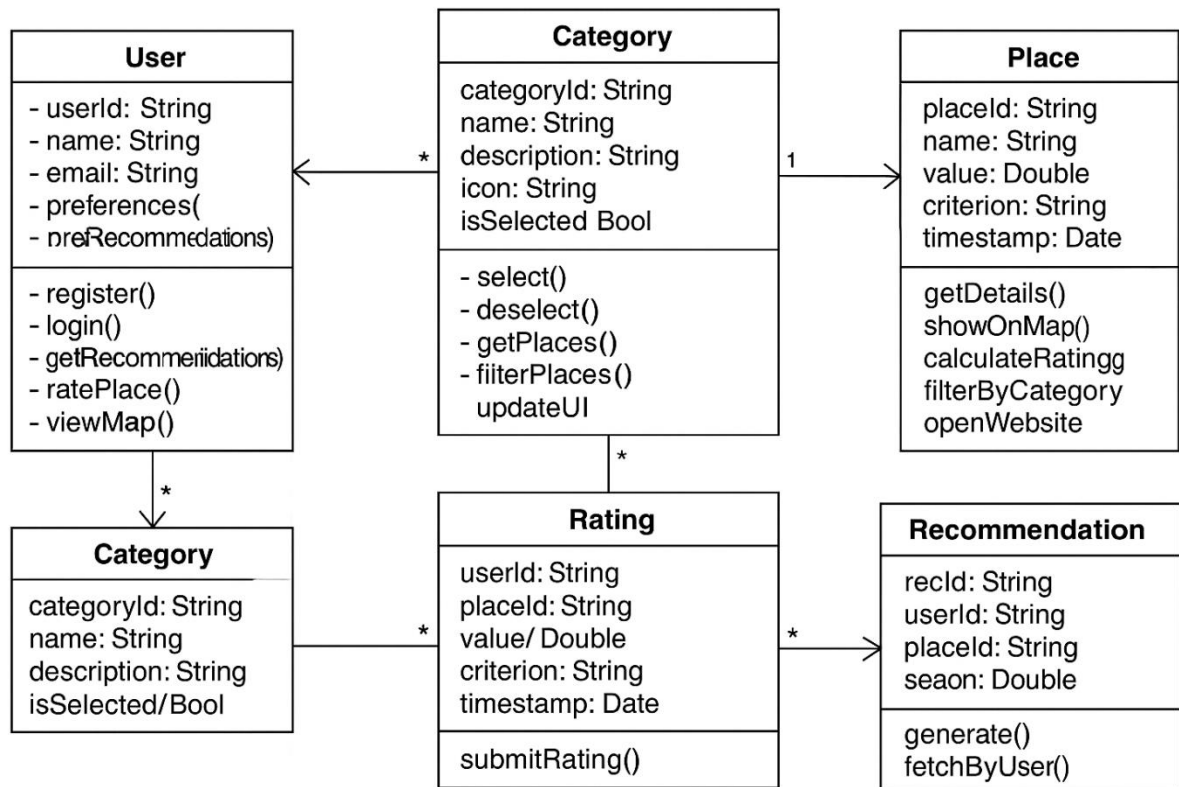


Рисунок 2.2 – UML-діаграма класів Android-додатку для рекомендації туристичних напрямків

Інтерфейс користувача спроектуємо з урахуванням принципів UX/UI-дизайну для забезпечення зручності та інтуїтивності взаємодії. Основні екрани додатку включатимуть сторінку входу та реєстрації, головний екран з рекомендованими місцями, сторінку деталей туристичного об'єкта з інформацією та кнопками взаємодії, а також екран карти з можливістю побудови маршрутів. Використання Material Design та анімацій покращить візуальне сприйняття застосунку.

Важливим аспектом є інформаційні потоки у системі. Додаток здійснюватиме запити до Firebase Firestore для отримання актуальних даних,

використовує Firebase Authentication для керування акаунтами користувачів та Firebase Cloud Messaging для надсилання персоналізованих сповіщень. Алгоритм рекомендацій базується на історії пошуків, оцінках користувача та категоріях місць, забезпечуючи релевантність отриманих результатів.

Таким чином, розроблювана система поєднує сучасні технології та підходи до мобільної розробки, дозволяючи користувачам отримувати персоналізовані туристичні рекомендації та взаємодіяти з ними у зручний спосіб.

2.2 Вибір засобів, методів і алгоритмів вирішення поставленого завдання

З розвитком розумних мобільних пристроїв і розвитком мережі трафіку у людей виникає бажання досліджувати нові та невідомі туристичні напрямки. Це призводить до зростання одного з галузі громадського харчування, туризм. Є багато нових напрямків, які туристи обирають відвідати вмісту та цікавих речей, які вони можуть відвідати та побачити, але їхня найбільша проблема – їх відсутність час. Розробка цієї програми надає користувачам можливість дуже швидкого пошуку туристичний напрямок, який був би цікавий для них особисто, а інші допомагають їм у цьому виборі користувачів шляхом присвоєння рейтингів туристичним напрямкам.

Початковий дизайн додатку для багатокритеріальної рекомендації туристичних напрямків створено в онлайн інструменті Canva. Використовуючи мову програмування Kotlin та XML для інтерфейсу користувача, додаток розроблено в середовищі програмування Android Studio. Firebase використовувався для зберігання даних у хмарі. Усі перераховані вище інструменти будуть докладніше пояснені в наступних розділах.

Canva – це веб-орієнтований інструмент, що надає можливість створення різноманітних графічних матеріалів, зокрема бізнес-презентацій, інфографіки, постерів, а також медіаконтенту для соціальних мереж [12]. Платформа містить

широкий вибір шаблонів, засоби для редагування текстових і візуальних елементів, а також бібліотеку зображень та іконок, що можуть бути використані користувачами для створення власних дизайнерських рішень. Популярність Canva зумовлена її інтуїтивним інтерфейсом, який дозволяє здійснювати розробку графічного контенту без необхідності глибокого опанування спеціалізованих дизайнерських інструментів та принципів. У межах даної кваліфікаційної роботи для створення логотипа (рис. 2.3) було використано сервіс Canva. Також за допомогою Canva було розроблено початковий макетний прототип додатка (рис. 2.4).

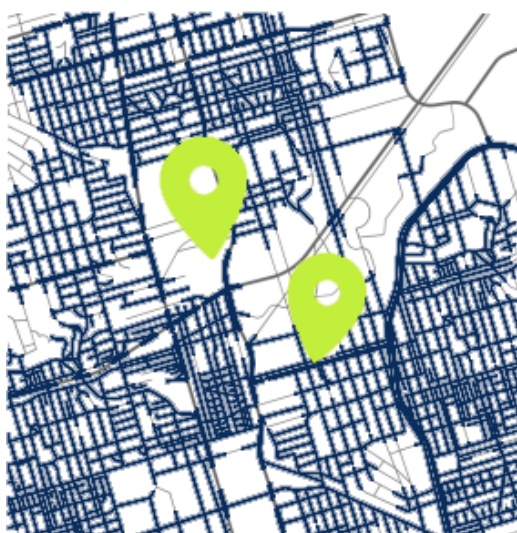


Рисунок 2.3 – Логотип програми

Android Studio є офіційним інтегрованим середовищем розробки (IDE) для створення додатків під операційну систему Android, що підтверджується у [13]. Це середовище базується на потужному редакторі коду та інструментах розробки IntelliJ IDEA, що забезпечує широкий спектр можливостей для підвищення продуктивності під час розробки мобільних застосунків. Серед ключових функцій Android Studio слід відзначити вбудований візуальний редактор інтерфейсу користувача, інструменти для керування створенням і упаковкою застосунків, а також віртуальний емулятор, який дозволяє тестувати додатки без потреби у фізичному пристрої. Крім того, середовище підтримує

кілька мов програмування, серед яких Java, Kotlin і C++, що надає розробникам гнучкість у виборі технологій.



Рисунок 2.4 – Ідея макета програми

Як зазначено у [14], Kotlin є сучасною мовою програмування, яка працює на віртуальній машині Java (JVM) та має можливість компіляції у JavaScript. Вона була розроблена компанією JetBrains, що також стоїть за IntelliJ IDEA, та позиціонується як більш лаконічна та виразна альтернатива мові Java. Kotlin усуває деякі недоліки Java, зокрема проблему нульових посилань (NullPointerException), надає можливості функціонального програмування та має компактний і читабельний синтаксис. Завдяки повній сумісності з Java Kotlin широко використовується для розробки Android-застосунків, дозволяючи розробникам спрощувати код та підвищувати його надійність.

XML (Extensible Markup Language), як зазначено у [15], є мовою розмітки, яка використовується для зберігання та передачі даних. Вона не є мовою програмування у традиційному розумінні, оскільки не містить операторів, які виконують дії, а натомість застосовується як транспортний механізм для представлення структурованої інформації у вигляді тегів. У контексті Android XML активно використовується для створення графічних інтерфейсів додатків, зокрема визначення макетів, стилів та ресурсів. Крім того, XML застосовується у конфігураційних файлах, що забезпечує гнучке керування параметрами застосунку без потреби у зміні коду.

Як стверджується у [16], Firebase є потужною платформою, що надає розробникам комплексний набір інструментів для створення, тестування та підтримки мобільних застосунків. Вона включає сервіси автентифікації користувачів, хмарні бази даних у реальному часі, хмарне зберігання файлів, аналітику, а також інструменти для розгортання та моніторингу застосунків. Firebase значно спрощує процес бекенд-розробки, дозволяючи програмістам зосередитися на функціональності клієнтської частини додатку. Завдяки інтеграції з іншими сервісами Google, такими як Google Analytics та Cloud Functions, платформа забезпечує ефективне керування даними, масштабованість та швидкий доступ до інформації. Це робить Firebase одним із найпопулярніших рішень для мобільної розробки, зокрема для Android-додатків, де важлива швидкість обробки запитів та зручність адміністрування.

РОЗДІЛ 3

РОЗРОБКА ANDROID-ДОДАТКУ ДЛЯ РЕКОМЕНДАЦІЙ ТУРИСТИЧНИХ НАПРЯМКІВ З ВИКОРИСТАННЯМ KOTLIN ТА FIREBASE

3.1 Практична реалізація об'єкта проєктування

Під час першого запуску програми користувачеві відображається екран входу. Якщо у користувача ще немає облікового запису, він може вибрати кнопку «Зареєструватися», ввести необхідні дані та підтвердити реєстрацію.

Для створення облікових записів використовується Firebase Authentication. Реєстрація здійснюється за допомогою методу `register`, структура якого наведена у лістингу 3.1. Метод приймає три параметри: адресу електронної пошти, пароль та ім'я користувача.

Система аутентифікації Firebase перевіряє коректність введеної електронної адреси, а також валідність пароля, який повинен містити щонайменше вісім символів, серед яких має бути хоча б одна літера та одна цифра. Якщо всі вимоги виконані, у Firebase створюється новий обліковий запис із автоматично згенерованим унікальним ідентифікатором.

Якщо ж користувач із такою електронною адресою вже існує, система виводить відповідне повідомлення у вигляді Toast-сповіщення: «Електронна адреса вже існує або недійсна». У разі некоректного пароля користувач також отримає сповіщення із вказівками щодо правильного введення даних для успішної реєстрації.

Лістинг 3.1 – Код способу реєстрації

```
fun performRegistration(userEmail: String, userPassword: String,
    userName: String) {
    auth.createUserWithEmailAndPassword(userEmail, userPassword)
        .addOnCompleteListener(this) { result ->
        if (result.isSuccessful) {
            Toast.makeText(baseContext, "Account created
successfully!", Toast.LENGTH_SHORT).show()
```

```

        val firebaseUser = auth.currentUser
        val userId = firebaseUser?.uid

        if (userId != null) {
            val profileData = mapOf("name" to userName)
            val defaultPreferences = mapOf("selectedChips" to
listOf("exists"))

            val documentsRef = db.collection("users")
                .document(userId)
                .collection("documents")

            documentsRef.document("user-info").set(profileData)
                .addOnFailureListener { exception ->
                    Log.e("FirestoreError", "Unable to save user
profile: ${exception.message}")
                }

            documentsRef.document("categoryPreferences").set(defaultPreferences)
                .addOnFailureListener { exception ->
                    Log.e("FirestoreError", "Unable to save user
preferences: ${exception.message}")
                }
        }

        val homeIntent = Intent(this,
MainActivity::class.java).apply {
            putExtra("registration", "true")
        }
        SavedStates.setNavigationBarIndex(1)
        startActivity(homeIntent)
        finish()

    } else {
        val error = result.exception?.localizedMessage ?:
        "Invalid or already used email"
        Toast.makeText(baseContext, error,
Toast.LENGTH_SHORT).show()
        Log.e("RegistrationError", "Sign-up failed: $error")
    }
}
}

```

кінець лістингу 3.1

При виборі маркера на карті або картки туристичного місця у списку рекомендованих локацій відкривається екран із детальною інформацією про обране місце. На цьому екрані відображаються зображення, назва, короткий

опис, а також, якщо доступно, посилання на офіційний ресурс. Для зберігання зображень використовується Firebase Storage.

Також передбачена кнопка «Показати на карті», яка дозволяє користувачеві переглянути розташування вибраного місця на карті. У нижній частині екрана розміщені чотири рейтингові шкали, кожна з яких відповідає певному критерію оцінки, а також група перемикачів. За допомогою цих елементів користувач може оцінити туристичне місце, наприклад, за шкалою від одного до п'яти, де один означає мінімальне задоволення, а п'ять – максимальне.

Оновлення рейтингу реалізоване у методі `submitRating`, схема роботи якого представлена у лістингу 3.2. Цей метод отримує як параметри значення оцінки, унікальний ідентифікатор туристичного місця та назву критерію, за яким здійснюється оцінювання. При кожному виклику методу необхідні дані завантажуються з бази даних, перевіряється їх наявність, і якщо вони відмінні від нуля, до них додається нова оцінка користувача. Після цього обчислюється оновлене середнє значення рейтингу, і всі оновлені дані зберігаються назад у базу даних. Крім того, оцінка, яку користувач призначив цьому місцю, зберігається у його профілі.

Лістинг 3.2 – Програмний код методу `submitRating`

```
fun submitRating(ratingWidget: RatingBar, locationId: String, category:
String) {
    if (ratingWidget.rating != 0.0f && !ratingWidget.isIndicator) {
        val locationRef = db.collection("places").document(locationId)

        locationRef.get().addOnSuccessListener { snapshot ->
            val countField = "${category}TotalVotes"
            val sumField = "${category}RatingSum"
            val averageField = "${category}AvgScore"
            var voteCount = snapshot.getDouble(countField) ?: 0.0
            var totalScore = snapshot.getDouble(sumField) ?: 0.0
            if (totalScore == 0.0) {
                handleZeroSumRating(ratingWidget, locationId, category,
totalScore)
            } else {
                totalScore += ratingWidget.rating.toDouble()
```

```

        voteCount += 1

        val newAverage = totalScore / voteCount

        user?.let { currentUser ->
            val ratingRef = db.collection("users")
                .document(currentUser.uid)
                .collection("documents")
                .document(locationId)

            ratingRef.update("${category}UserRating",
ratingWidget.rating.toDouble())
        }

        ratingWidget.isIndicator = true

        locationRef.update(mapOf(
            sumField to totalScore,
            countField to voteCount,
            averageField to newAverage
        )).addOnSuccessListener {
            Toast.makeText(this, "Rating submitted!",
Toast.LENGTH_SHORT).show()
        }
    }

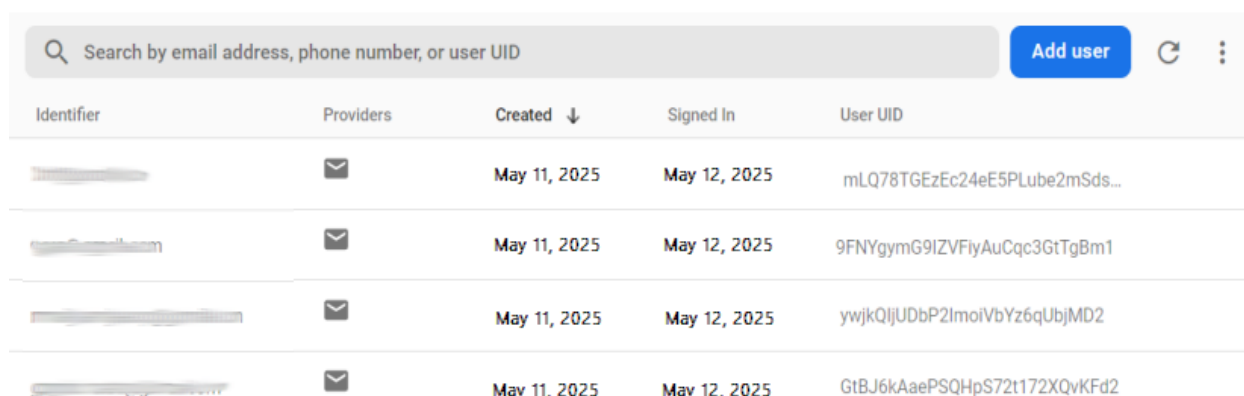
    updateAllAverages(locationId)
}
}
}
}
}

```

кінець лістингу 3.2

Для автентифікації користувачів у системі застосовується Firebase Authentication, яка надає вбудовані методи перевірки коректності введених даних. На рисунку 3.1 представлено кілька активних та збережених облікових записів користувачів у Firebase Authentication. Ця технологія зберігає такі відомості, як адреса електронної пошти, унікальний ідентифікатор користувача, дату створення та останнього входу в систему. Однак збережений пароль не може бути переглянутий або змінений безпосередньо через базу даних. Користувач має можливість самостійно ініціювати процедуру відновлення доступу, надіславши запит на скидання пароля через застосунок. Альтернативно, це може здійснити адміністратор бази даних через Firebase

Authentication, після чого користувач отримає електронний лист із подальшими інструкціями щодо відновлення доступу.



Identifier	Providers	Created ↓	Signed In	User UID
[Redacted]	✉	May 11, 2025	May 12, 2025	mLQ78TGEzEc24eE5PLube2mSds...
[Redacted]	✉	May 11, 2025	May 12, 2025	9FNYgymG9IZVFiyAuCqc3GtTgBm1
[Redacted]	✉	May 11, 2025	May 12, 2025	ywjKQlJUDbP2lmoiVbYz6qUbjMD2
[Redacted]	✉	May 11, 2025	May 12, 2025	GtBJ6kAaePSQHps72t172XQvKFd2

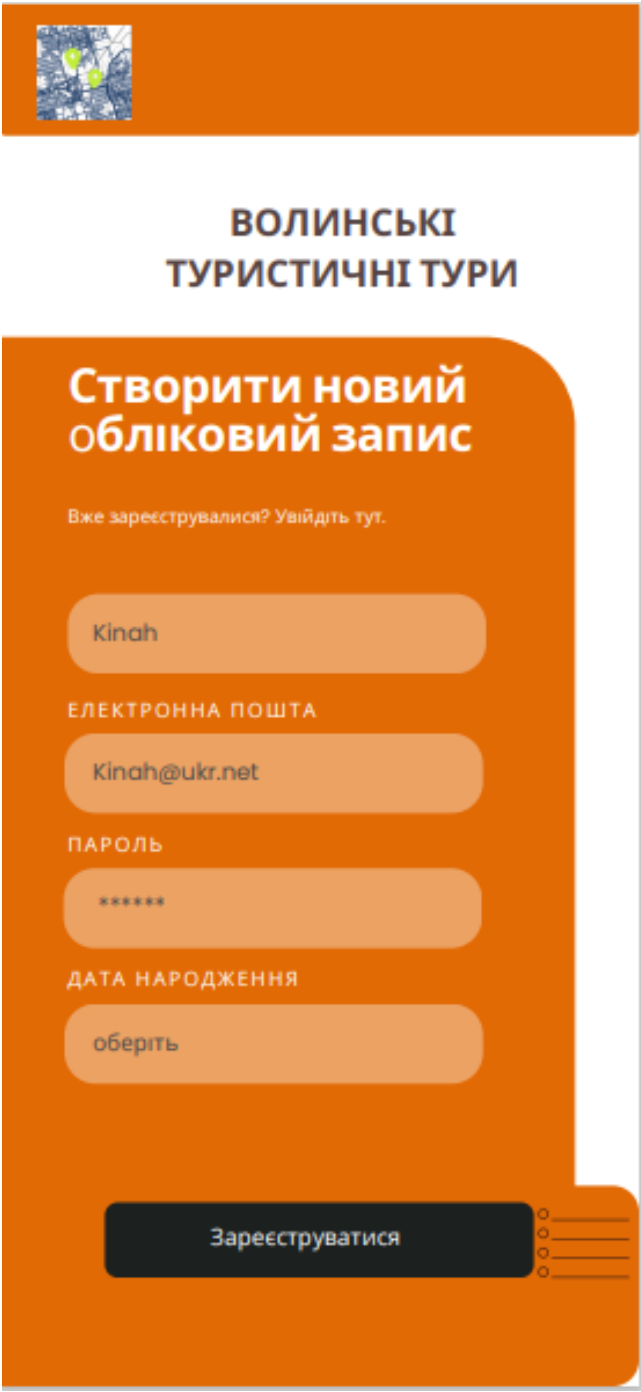
Рисунок 3.1 – Зареєстровані користувачі в системі

Перед початком використання програми її необхідно встановити. Після запуску користувач потрапляє на головний екран (рис. 3.2), на якому розташовані поля для автентифікації. Якщо користувач вже має обліковий запис, йому потрібно ввести електронну пошту та пароль у відповідні поля.

У разі втрати пароля надається можливість його відновлення. Натискання на посилання «Забули пароль?» відкриває спливаюче вікно, де користувач вводить електронну пошту, вказану під час реєстрації. Після цього система надсилає лист із подальшими інструкціями щодо зміни пароля.

Якщо обліковий запис відсутній, користувач може натиснути кнопку «РЕЄСТРАЦІЯ», що переведе його на екран реєстрації.

Для використання програми користувач повинен створити обліковий запис. Під час реєстрації необхідно ввести своє ім'я, адресу електронної пошти та пароль у відведених для цього полях, мінімальна довжина яких становить вісім символів. Пароль повинен містити принаймні одну букву і принаймні одну цифру. В іншому випадку буде показано користувачеві повідомлення про те, що пароль не відповідає вимогам.



**ВОЛИНСЬКІ
ТУРИСТИЧНІ ТУРИ**

**Створити новий
обліковий запис**

Вже зареєструвалися? Увійдть тут.

Kinah

ЕЛЕКТРОННА ПОШТА

Kinah@ukr.net

ПАРОЛЬ

ДАТА НАРОДЖЕННЯ

оберіть

Зареєструватися

Рисунок 3.2 – Інтерфейс входу

Після реєстрації користувачеві відображається спливаюче вікно з вітальним повідомленням, а потім екран, де потрібно вибрати категорії туристичних напрямків, які його цікавлять, що показано на рисунку 3.3.

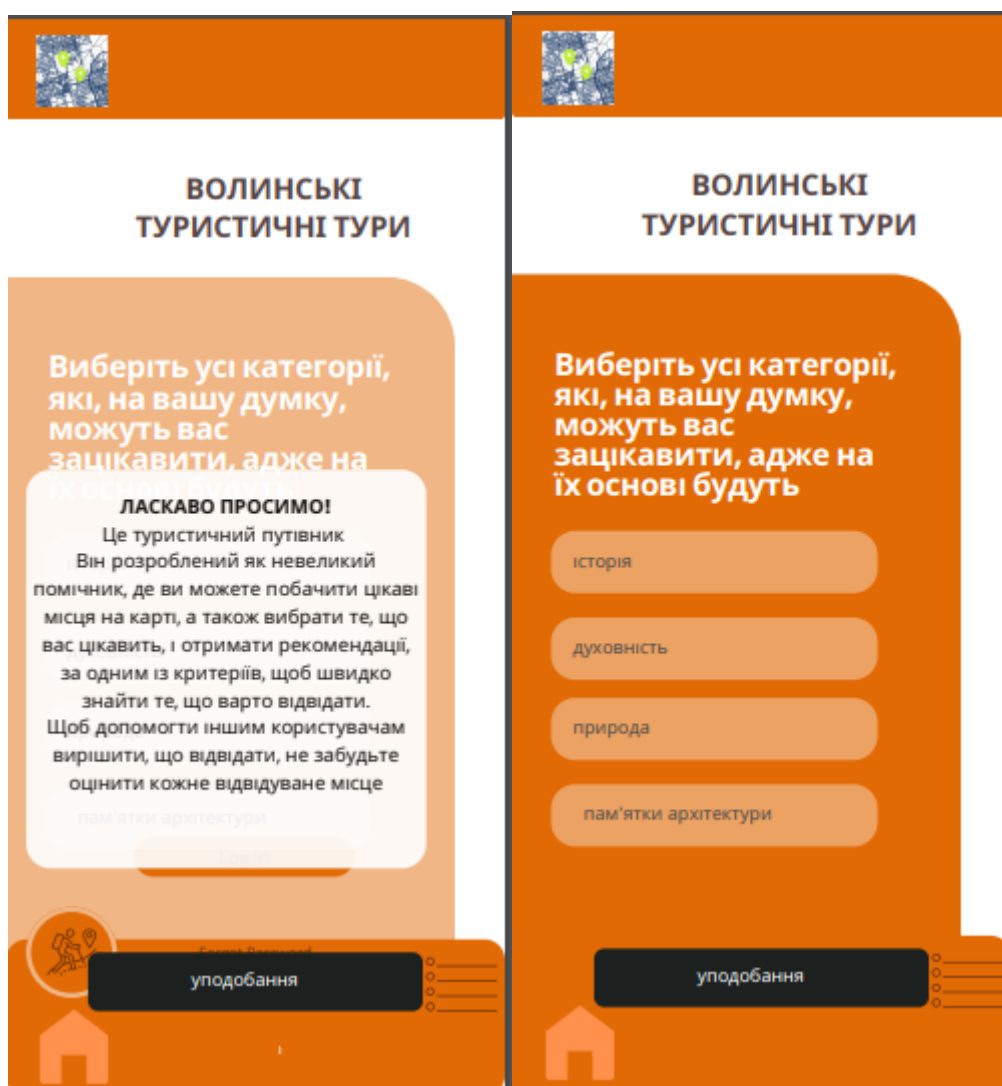


Рисунок 3.3 – Інтерфейс вибору категорій

Перша вкладка панелі навігації переведе користувача на екран карти, де всі введені туристичні напрямки відображаються шпильками – «червоними колами».

Це також головний екран за замовчуванням для користувача, який уже має обліковий запис користувача та який увійшов і залишився в системі під час останнього входу в програму.

Фрагмент карти показано на рисунку 3.4 та натискання будь-якої шпильки, відкриває новий екран для користувача з детальною інформацією про вибраний пункт призначення.



Рисунок 3.4 – Перегляд карти

Третя вкладка панелі навігації показує користувачеві екран із рекомендованими туристичними напрямками. За замовчуванням пункти призначення відсортовано за загальним рейтингом, і користувач сам обирає, який фільтр йому буде рекомендовано.

Приклад рекомендованих напрямків із застосуванням фільтра «Оригінал» показано на рисунку 3.5.



Рисунок 3.5 – Інтерфейс рекомендованих місць призначення

Користувач також може клацнути на будь-який пункт призначення, і на екрані, як показано на рисунку 3.6 показано інформаційну інформацію для певного туристичного місця, що містить зображення цього місця, його назву, опис і кнопку «ВІДМІТКА НА КАРТІ», яка повертає користувача до першої картки та виділяє пункт призначення, де він хотів знати, де воно знаходиться. Якщо є веб-сайт про пункт призначення, буде показано посилання, яке перемістить користувача на цей веб-сайт після натискання.

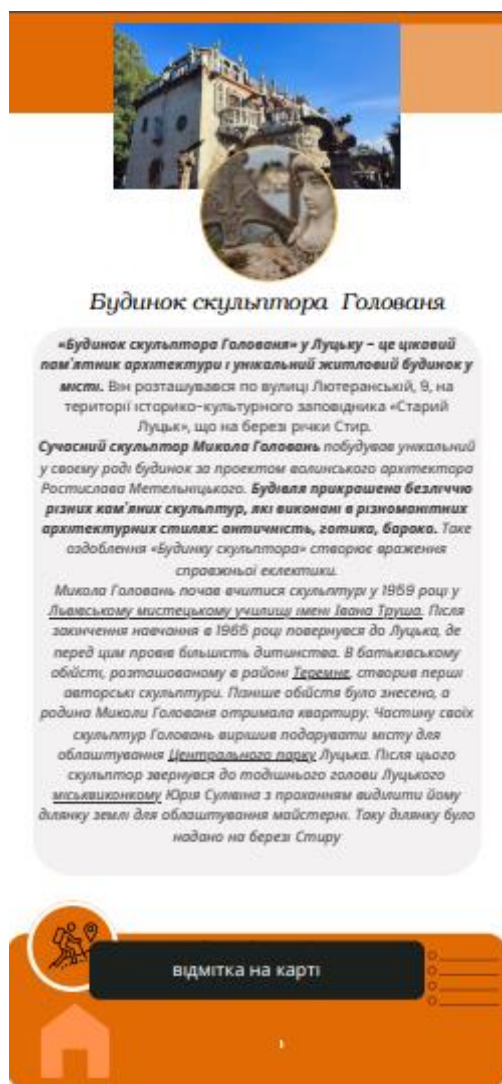


Рисунок 3.6 – Інтерфейс деталей призначення

3.2 Тестування та налагодження інформаційно-комп'ютерної системи

Метою тестування мобільного додатка є оцінка його коректної роботи, стабільності та відповідності очікуваним функціональним характеристикам. У процесі тестування розглядаються два ключові сценарії використання, що дозволяють оцінити ефективність роботи алгоритмів рекомендацій.

Сценарій 1 – відсутність вибраних категорій. На початковому етапі користувач може не обрати жодної категорії туристичних напрямків. У такому випадку система не надаватиме загальні рекомендації, а на вкладці «Рейтинг» відсутні туристичні місця. Натомість користувач отримує інструкцію щодо

переходу до вкладки «Налаштування» для вибору категорій, необхідних для персоналізації рекомендацій. Відсутність вибраних категорій не призводить до збоїв у роботі додатка або некоректного відображення інформації (рис. 3.7).



Рисунок 3.7 – Відображення екрана рекомендованих місць призначення, якщо не вибрано жодної категорії

Сценарій 2 – вибір фільтра для персоналізації рекомендацій. Додаток передбачає використання фільтрів для персоналізації туристичних рекомендацій. За замовчуванням застосовується фільтр «найвищий рейтинг»,

проте користувач може обрати альтернативні параметри, зокрема «доступність», «оригінальність» або «фотогенічність». Результати тестування підтверджують, що зміна фільтра коректно впливає на список рекомендованих напрямків, забезпечуючи персоналізований підхід до формування списку (рис. 3.8).



Рисунок 3.8 – Зміна місця призначення залежно від фільтра

Результати тестування засвідчили, що мобільний додаток функціонує відповідно до очікуваних параметрів. Відсутність вибраних категорій не спричиняє некоректної роботи системи, а користувач отримує інструкцію щодо подальших дій, а можливість використання фільтрів забезпечує персоналізацію рекомендацій. Таким чином, реалізований багатокритеріальний підхід до

рекомендації туристичних напрямків є ефективним та відповідає заявленій функціональності.

3.3 Розробка бази даних

У сучасних додатках, орієнтованих на користувача, надзвичайно важливо забезпечити зручне, швидке й масштабоване зберігання даних. Використання хмарних баз даних, таких як Cloud Firestore, дозволяє реалізувати гнучкі архітектурні рішення, які підтримують індивідуальні потреби користувачів і водночас забезпечують ефективний доступ до загальнодоступної інформації. У цьому контексті розробка двох взаємопов'язаних баз даних у Firestore дозволила створити збалансовану систему для зберігання як персоналізованих даних користувача, так і детальної інформації про об'єкти інтересу – зокрема, туристичні місця.

Для коректної роботи застосунку було використано дві бази даних, обидві з яких є частиною Cloud Firestore. Окрім цього, для зберігання зображень застосовувався Firebase Storage.

Перша база даних містить інформацію про користувачів, включаючи вибрані ними категорії, оцінки, які вони ставлять певним туристичним місцям, а також особисті дані. Кожному користувачеві призначається унікальний ідентифікатор, отриманий через Firebase Authentication, який генерується випадковим чином під час реєстрації.

Друга база даних містить інформацію про туристичні місця. Для кожного пункту призначення зберігається його унікальний ідентифікатор, назва, категорії, до яких він належить, опис, посилання на зображення, веб-сайт (за наявності), координати, а також дані про оцінки: кількість користувачів, які залишили відгуки, загальна кількість оцінок і середній рейтинг. Структура бази даних користувачів представлена на рисунку 3.9.

Крім того, для забезпечення швидкого доступу до інформації та оптимізації запитів використовувалися індекси, налаштовані в Cloud Firestore.

Завдяки інтеграції всіх цих компонентів забезпечується стабільна та ефективна робота застосунку навіть при зростанні кількості користувачів і даних.

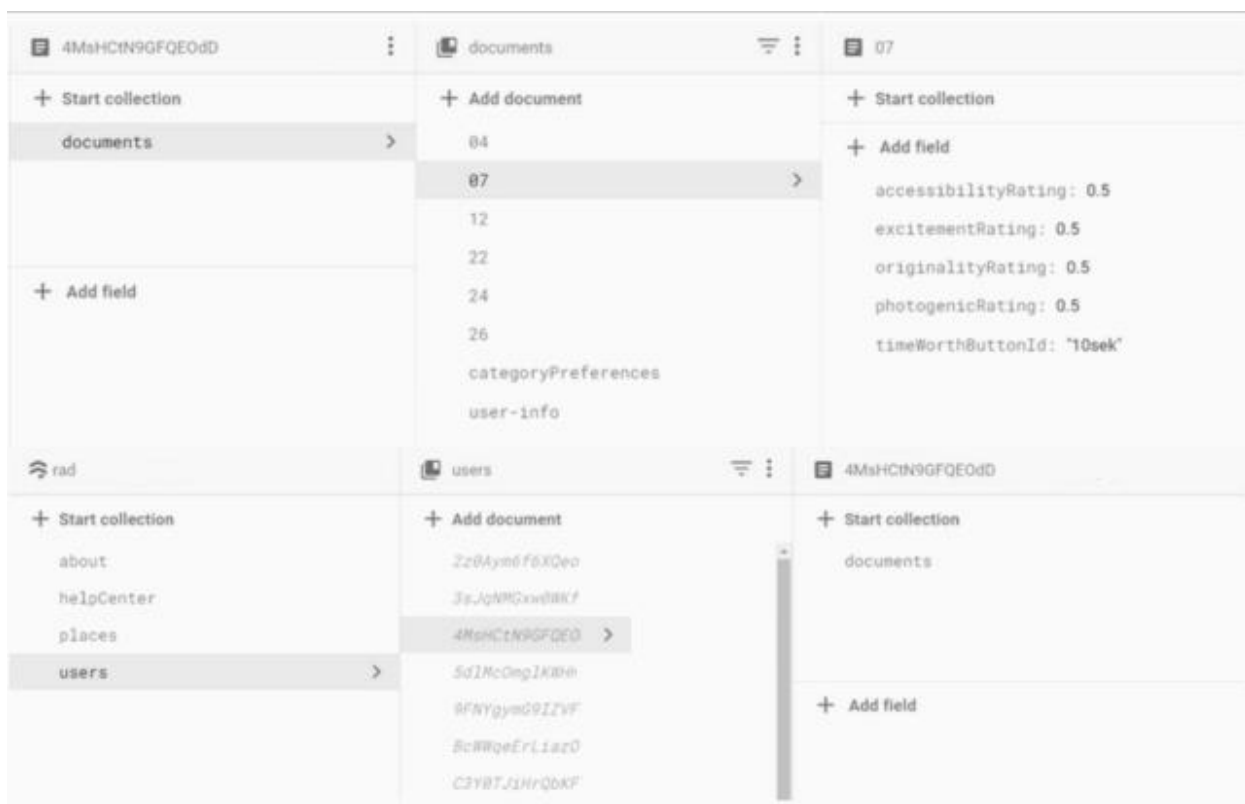


Рисунок 3.9 – База даних користувачів

Структура бази даних туристичних місць зображено на рисунку 3.10.

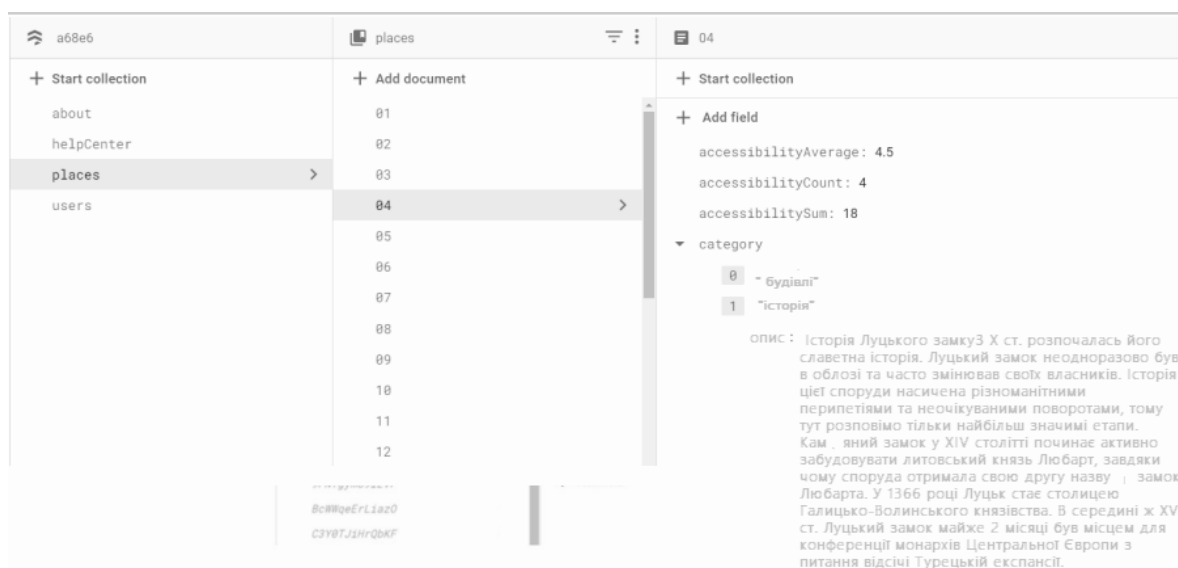


Рисунок 3.10 – База даних туристичних місць

3.4 SEO інформаційно-комп'ютерної системи

SEO (Search Engine Optimization) відіграє ключову роль у забезпеченні видимості та популярності мобільного додатку серед потенційних користувачів. Для Android-додатку з рекомендацією туристичних напрямків, розробленого на Kotlin із використанням Firebase, важливо застосовувати сучасні методи оптимізації для підвищення його рейтингу в пошукових системах Google Play та зовнішніх пошукових системах (Google, Bing).

Одним із головних етапів SEO-оптимізації є коректне налаштування метаінформації: вибір релевантної назви додатку, створення унікального та інформативного опису з використанням ключових слів (наприклад, «туристичні маршрути», «рекомендації подорожей», «мандрівки з AI»). Додатково важливо включити унікальні скріншоти, відеопрезентацію та зручну навігацію в Google Play Store.

Для зовнішнього просування застосовуються методи ASO (App Store Optimization), що включають роботу з відгуками користувачів, аналіз ключових слів конкурентів та A/B тестування описів у Google Play Console. Використання Firebase Analytics дозволяє відстежувати поведінку користувачів, що допомагає покращити юзабіліті та адаптувати функціонал до їхніх потреб.

Крім того, Firebase Dynamic Links дає змогу створювати глибокі посилання для прямого перенаправлення користувачів на конкретні розділи додатку, що значно підвищує його зручність та ефективність рекламних кампаній. Використання Google Ads, Facebook Ads та TikTok Ads дозволяє здійснювати таргетоване просування серед цільової аудиторії – туристів, блогерів, мандрівників.

Додатково важливим аспектом є технічна SEO-оптимізація: підвищення швидкості завантаження за допомогою Jetpack Compose, оптимізація роботи з мережею через Retrofit, а також мінімізація використання ресурсів Firebase Firestore для зменшення витрат на запити. Також слід забезпечити інтеграцію

додатку з веб-ресурсами через PWA (Progressive Web App) для покращення доступності через мобільні браузері.

Таким чином, комплексний підхід до SEO та маркетингового просування включає внутрішню оптимізацію (ASO), аналітику поведінки користувачів (Firebase Analytics), ефективну рекламу (Google Ads, Facebook Ads) та технічну оптимізацію (швидкість завантаження, інтеграція з веб-сервісами). Це дозволяє забезпечити максимальну ефективність додатку та залучити широку аудиторію користувачів.

ВИСНОВКИ

У процесі виконання кваліфікаційної роботи на тему «Розробка Android-додатку для рекомендації туристичних напрямків з використанням Kotlin та Firebase» було реалізовано комплекс завдань, сформульованих у роботі. Основна мета полягала у створенні ефективного мобільного застосунку, здатного надавати персоналізовані туристичні рекомендації на основі вподобань користувача, з використанням сучасних технологій та інструментів, таких як мова програмування Kotlin та хмарна платформа Firebase. У ході дослідження було вирішено низку практичних і теоретичних задач, що забезпечили досягнення поставленої мети.

Перш за все, було здійснено ґрунтовний аналіз сучасних рішень у сфері мобільних рекомендаційних систем, зокрема розглянуто такі додатки як TripIT, Visit Ukraine, Fluxify, Mindtrip та BetterTravel. Проаналізовано їхні функціональні можливості, архітектуру, принципи побудови рекомендацій, що дозволило виявити сильні сторони та недоліки існуючих підходів і обґрунтувати необхідність створення нового рішення, адаптованого до українського туристичного середовища та з орієнтацією на індивідуальні вподобання користувача.

Далі були визначені функціональні та нефункціональні вимоги до Android-додатку. Сформовано архітектуру системи, що включає дві основні бази даних (користувачів і туристичних місць), реалізовано механізми автентифікації через Firebase Authentication та інтеграцію з Firebase Firestore для зберігання і обробки даних. Важливим аспектом проектування стало використання UML-діаграм, зокрема діаграми варіантів використання, класів і послідовностей, що дозволили формалізувати логіку взаємодії між компонентами системи.

Було реалізовано персоналізовану функціональність: після реєстрації користувач обирає категорії туристичних напрямків, а система на основі цих уподобань формує список рекомендованих місць. Рекомендації сортуються за

такими критеріями, як «популярність», «оригінальність», «фотогенічність» та «доступність». Для кожного туристичного об'єкта передбачено можливість оцінювання за кількома шкалами, з подальшим розрахунком середніх значень, що впливають на ранжування. Користувач також має змогу переглядати обрані місця на інтерактивній карті Google Maps.

У рамках інтеграції Firebase реалізовано не лише зберігання основних даних, а й застосовано Firebase Storage для зображень, а також Firebase Cloud Messaging для персоналізованих сповіщень. Було налаштовано індексацію у Firestore, що дозволило підвищити продуктивність додатку при зростанні обсягів інформації.

Проведено тестування розробленої системи. Зокрема, перевірено поведінку додатку у випадку, якщо користувач не обрав категорії (додаток не зависає, а виводить підказку), а також ефективність роботи фільтрів при побудові персоналізованих рекомендацій. Тестування підтвердило стабільну роботу додатку, відповідність заявленому функціоналу та зручність взаємодії для кінцевого користувача.

На завершення було проаналізовано результати тестування, і запропоновано шляхи подальшого вдосконалення системи, зокрема – додавання функцій соціальної взаємодії (відгуки, спільні маршрути), вдосконалення алгоритмів рекомендацій за допомогою методів машинного навчання, а також розширення бази даних туристичних об'єктів шляхом залучення користувачів до створення контенту.

Отже, всі поставлені завдання, були успішно реалізовані, що свідчить про досягнення мети кваліфікаційної роботи та ефективність обраного технічного підходу.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Halkiopoulou C., Antonopoulou H., Gkintoni E., Giannoukou I. An Expert System for Recommendation Tourist Destinations: An Innovative Approach of Digital Marketing and Decision-Making Process. URL: <https://surl.li/KNWVSC> (дата звернення: 14.02.2025).
2. Application for e-Tourism: Intelligent Mobile Tourist Guide. URL: <https://ieeexplore.ieee.org/abstract/document/7373873> (дата звернення: 19.02.2025).
3. Visit Ukraine. URL: https://omore.city/articles/194376/-scho-ce-za-organizaciya-ta-chim-vona-zajmaetsya?utm_source=chatgpt.com (дата звернення: 19.02.2025).
4. Ukraine Travel Guide. URL: <https://www.ukraine.com/> (дата звернення: 19.02.2025).
5. Мандруй Україною: стартував проект популяризації внутрішнього туризму. URL: <https://voladm.gov.ua/new/mandruy-ukrayinoyu-startuvav-proekt-populyarizaciyi-vnutrishnogo-turizmu/> (дата звернення: 19.02.2025).
6. BetterTravel. URL: <https://aimojo.io/uk/tools/bettertravel/> (дата звернення: 19.02.2025).
7. 5 найкращих інструментів ШІ для планування подорожей. URL: <https://www.unite.ai/uk/best-ai-tools-for-travel-planning/> (дата звернення: 19.02.2025).
8. AI-Driven Trip Planning App Development: Benefits, Features, Process and Cost. URL: <https://www.zealousys.com/blog/how-to-build-ai-trip-planner-app/> (дата звернення: 19.02.2025).
9. Українська студентка розробила мобільний застосунок для туристів, які хочуть мандрувати без групових екскурсій. URL: <https://shotam.info/ukrainska-studentka-rozrobyla-mobilnyy-zastosunok-dlia-turystiv-iaki-khochut-mandruvaty-bez-hrupovykh-ekskursiy/> (дата звернення: 19.02.2025).

10. Aili Chen, Xuyang Ge, Ziquan Fu, Yanghua Xiao, Jiangjie Chen. TravelAgent: An AI Assistant for Personalized Travel Planning. UML: https://arxiv.org/abs/2409.08069?utm_source=chatgpt.com (дата звернення: 02.05.2025).
11. Systematic review of mobile travel apps and their smart features and challenges. URL: https://www.emerald.com/insight/content/doi/10.1108/jhti-02-2022-0087/full/html?utm_source=chatgpt.com (дата звернення: 02.05.2025).
12. Canva. URL: <https://www.canva.com/> (дата звернення: 13.02.2025).
13. The Use Of Integrated Development Environment (Ide) In Making An Android-Based Weight Calculator. URL: <https://ejournal.uika-bogor.ac.id/index.php/INOVA-TIF/article/view/9332> (дата звернення: 13.02.2025).
14. Kotlin Docs. URL: <https://kotlinlang.org/docs/home.html> (дата звернення: 08.03.2025).
15. What Is XML (Extensible Markup Language)? Meaning, Elements, and Benefits. URL: <https://www.spiceworks.com/tech/tech-general/articles/what-is-xml/> (дата звернення: 17.04.2025).
16. Get Started with Realtime Database – Firebase. Посібник для швидкого початку роботи з Firebase Database. URL: <https://firebase.google.com/docs/database/flutter/start> (дата звернення: 10.03.2025).