

Міністерство освіти і науки України
Національний університет "Києво-Могилянська Академія"
Кафедра інформатики Факультету Інформатики

Розробка android-додатку з геоданими

Текстова частина до курсової роботи
за спеціальністю “Комп’ютерні науки та інформаційні технології”
112

Керівник курсової роботи
ст. викладач, доцент Жежерун О. П.

(підпис)

“ ____ ” _____ 2021 р.

Виконав студент ФІ-4
Урсатий О. В.

“ ____ ” _____ 2021 р.

Київ 2021

Міністерство освіти і науки України
Національний університет "Києво-Могилянська Академія"
Кафедра інформатики Факультету Інформатики

ІНДИВІДУАЛЬНЕ ЗАВДАННЯ
для курсової роботи

Для студента Факультету Інформатики 4 курсу
ТЕМА Розробка android-додатку з геоданими

Складові текстової частини курсової роботи:

Титульний аркуш

Індивідуальне завдання

Календарний план

Зміст

Вступ

Основна частина

Висновок

Посилання

Додатки

Дата "17" __ грудня 2020 р. Керівник _____
(підпис)

Завдання отримано _____
(підпис)

Тема: Розробка android-додатку з геоданими

Календарний план написання курсової роботи:

№	Назва етапу	Дата	Помітки
1.	Узгодження теми	01.11.2020	
2.	Пошук літератури	20.11.2020	
3.	Вивчення архітектури мобільних додатків	10.12.2020	
4.	Дослідження сфери геоданих	10.01.2021	
5.	Ознайомлення з технологіями та фреймворками	10.01.2021	
6.	Дослідження розробки додатків з картами	21.01.2021	
7.	Визначення специфікації	1.02.2021	
8.	Розробка документації	8.02.2021	
9.	Розробка застосунку	15.02.2021	
10.	Написання першої частини	15.03.2021	
11.	Аналіз роботи з керівником	22.03.2021	
12.	Написання другої частини	29.03.2021	
13.	Аналіз роботи з керівником	1.04.2021	

14.	Внесення змін згідно з зауваженнями керівника	4.04.2021	
15.	Створення презентації	7.04.2021	
16.	Захист курсової	18.04.2021	

Студент Урсатий О. В.

Керівник Жежерун О. П.

“ ”
— —

Зміст

Зміст	4
Вступ	6
Технології картографії	9
1.1 Огляд технологій	9
Google Maps	9
Open Street Maps	11
Yandex Maps	12
MapBox	14
Apple Maps	14
1.2 Порівняння картографічних сервісів.	16
Функціональність відображення мапи	16
Навігація	18
Мультиплатформенність	18
Пошуковий функціонал	19
Складність впровадження	20
Висновки	20
Технічне завдання та інструменти	22
2.1 Аналіз проблеми	22
2.2 Технічне завдання	23
2.3 Вибір технологій	24
Огляд мобільних ОС	24
Порівняння розробки під IOS та Android	25

	5
Кросс-платформенна розробка	26
Розробка додатку	28
3.1 Створення проекту	28
3.2 Робота з мапою	30
3.3 Імплементация бази даних	32
3.4 Діаграма класів застосунку	35
3.5 Скріншоти роботи програми	36
Висновок	40
Посилання	41
Додатки	43
Лістинг програмного коду 1	43
Лістинг програмного коду 2	44

Вступ

Швидкий розвиток інформаційних технологій впливає практично на кожну сферу людської діяльності. Задача пошуку місця розташування та окреслення шляхів є нагальною вже не одну тисячу років. Сьогодні, у зв'язку з трендом на мобільність та глобалізацією зросла кількість людей, що подорожують. А також мобільних пристроїв, що докорінно змінило уявлення про карти, котрі висять на стінах та лежать у бардачку. “За 300 метрів на кільці поверніть праворуч” - певне чув кожен з нас, а сучасні сервіси доставки та таксі спрощують нам життя та зберігають час.

Я вважаю, що наразі ринок картографічного програмного забезпечення доволі монополізований додатком Google Maps. Цей додаток має потужний функціонал та задовольняє більшість потреб пов'язаних з пошуком місць та навігацією, і щомісяця має понад мільярд користувачів. Хоча це покриває більшість необхідного функціоналу, але таким чином перевантажує користувачів з більш вузькими потребами.

Через те, що найпопулярніший у світі додаток (Google Maps) намагається закрити потреби всіх груп користувачів ми отримуємо перевантажений функціями продукт. Це подекуди заважає простим юзерам отримати позитивний досвід користування. Наприклад, літні люди, не розуміють як поділитися своїм місцезнаходженням. Тому з часом ринок почав пропонувати альтернативи під конкретні потреби

користувачів. Наприклад Navigator Truck Pro для водіїв дальnobійних маршрутів. [\[1\]](#)

Функціональність картографування є невід'ємною складовою розробки сучасного програмного забезпечення. Картографія використовується у різних сферах бізнесу по всьому світі, від логістичних сервісів до веб-сайтів маленьких виробників з фізичними точками.

Хоча просто отримати свою геолокацію (координати) доволі просто в наш час, коректно відобразити особливості кожної конкретної місцевості є складнішим завданням. І знову ж таки кожна група користувачів потребує різних "уточнень". Якщо водіям цікаво знати інформацію про затори на дорозі, то, наприклад, хітчайкерам важливіше знати стан поверхні та природні труднощі які можуть трапитися їм на дорозі. Таким чином ми дійшли до необхідності сегрегації функціоналу картографічного застосунку. У випадку нашого проекту я вирішив взяти конкретну групу людей та прибрати все зайве з найпопулярнішого картографічного застосунку, залишивши лише важливі для них функції.

Я вирішив виокремити таку групу людей як шукачі грибів та створити аналог такого додатку саме для їх потреб. Також додаток може знайти застосування у інших людей, котрі користуються мапами та зберігають точки.

В загальному функціонал такого додатку звужується до збереження/поширення точок на мапі. І саме це може стати великою перевагою для грибних цінителів так як ніякі ресторани, кафе чи пробки не будуть відволікати їх увагу від лісових масивів, особливо після дощу. Програмно такий продукт являє собою використання картографічного API, роботу з мобільним інтерфейсом через нативну розробку, налаштування взаємодії з SQL базою даних і налаштування java бібліотек.

1. Технології картографії

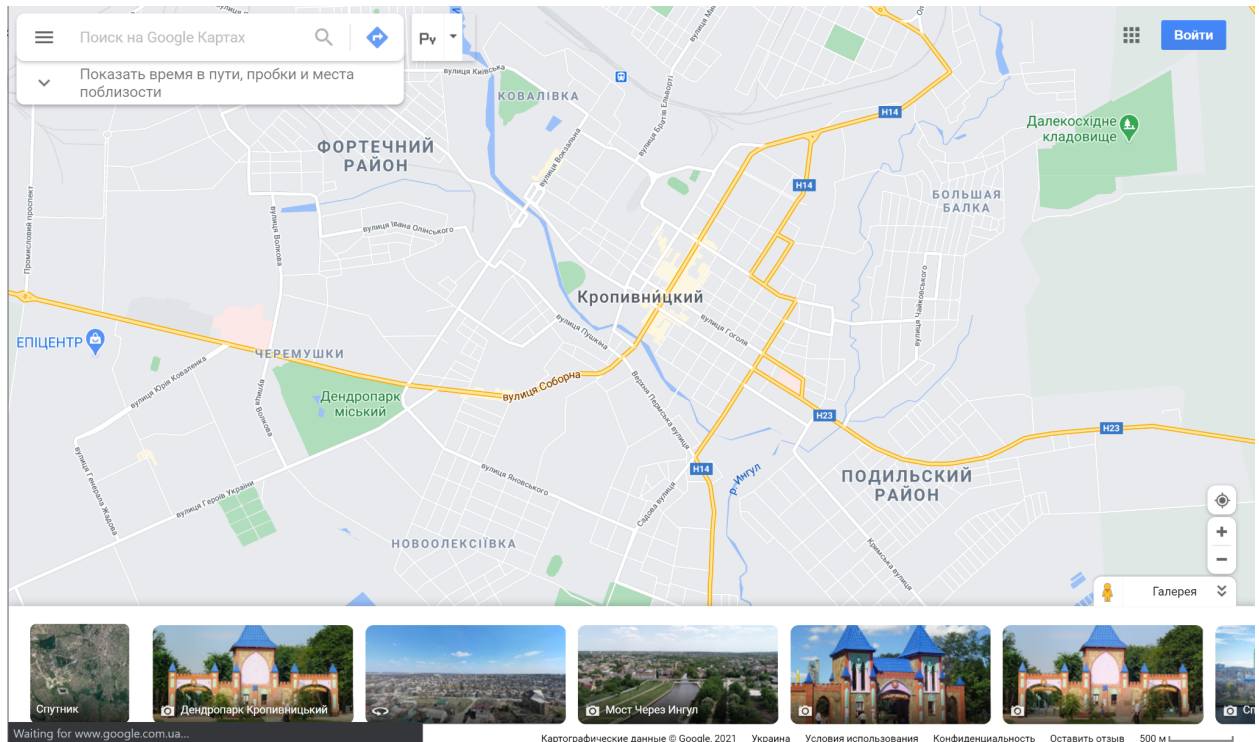
1.1 Огляд технологій

Сьогодні сервіси картографії пропонуються різними компаніями, кожна з яких пропонує власний продукт, хоча значна частина функціоналу є тотожною. Практично кожний постачальник карт надає послуги відображення мапи, пошуку, навігації та інші послуги. Google Maps - сервіс постачання карт, що першим спадає мені на думку, хоча поширені також Open Street Maps, Apple Maps, Яндекс Карти. При виборі постачальника карт слід враховувати функціональні вимоги до застосунку, а правильний вибір інструменту картографування зменшить витрати (послуги деяких компаній є платними) та спростить розробку.

У цьому розділі розглянуто послуги основних постачальників геоінформаційних систем - Google Maps, OpenStreetMap, Яндекс Maps та MapBox.

Google Maps

Google Maps - один з основних продуктів компанії Google. Сервіс існує з 2005 року, за цей було випущено ряд продуктів та програмних сервісів для розробників. Сьогодні Google Maps Platform є комплексом сервісів, що надають картографічні послуги, що побудовані на основі власних розробок. [\[2\]](#)



Додаток 1.1.1 Головний екран Google Maps Web

Google Maps Platform забезпечує покриття 99% поверхні Землі та має понад 1 000 000 000 активних користувачів щомісяця [3]. Високий рівень кастомізації та якісна документація роблять продукт дуже поширеним, як серед великих компаній, так і серед стартапів, хоча їх карти є не найдешевшим рішенням.

Google виокремлює для розробників три основних продукти:

- **Google Maps API.** Сервіс для відображення інтерактивних карт. Для розробника доступні різноманітні фігури, маркери та зображення на мапі, у тому числі зроблені власноруч. Google Street View надає можливість “прогулятися” вулицями за допомогою тривимірної проекції міст. Ця функціональність

реалізована з використанням спеціального обладнання для панорамного фотографування.

- **The Directions API.** Ще один веб-сервіс, для створення маршруту між точками заданими користувачем. Надає вичерпний результат, з урахуванням трафіку на дорогах в реальному часі.
- **Places API.** Надає інформацію про місця по всьому світу. Саме цей сервіс допомагає вам знайти потрібний де поїсти в новому місті, чи потрібний будинок. Доступна система відгуків для місць, пошук місця за назвою та за місцем розташування.

Open Street Maps

OSM - open-source картографічний проект, що забезпечує картами велику кількість веб-ресурсів та додатків. Геосервіс забезпечує безкоштовне використання для більшості функцій. Завдяки ентузіастам карти поповнюються новими точками та кращими маршрутами, що є перевагою сервісу у порівнянні з конкурентами. , через що сервіс славиться хорошою деталізацією. Інтерфейс для розробника складається з декількох сервісів.



Додаток 1.1.2 OpenStreetMap

Основний сервіс є має потужний функціонал для взаємодії з мапами. Він підтримується з 2005 року. OverPass API є розподіленим сервером і в порівнянні з основним API оптимізовано для отримання даних. Він містить вибрані частини основного сервісу. Цей сервіс містить серйозну підтримку від ком'юніті та має імплементації для різних платформ. Під час виконання роботи було досліджено простіший OverPass API від OpenStreetMap, зокрема бібліотеку для його Android імплементації.

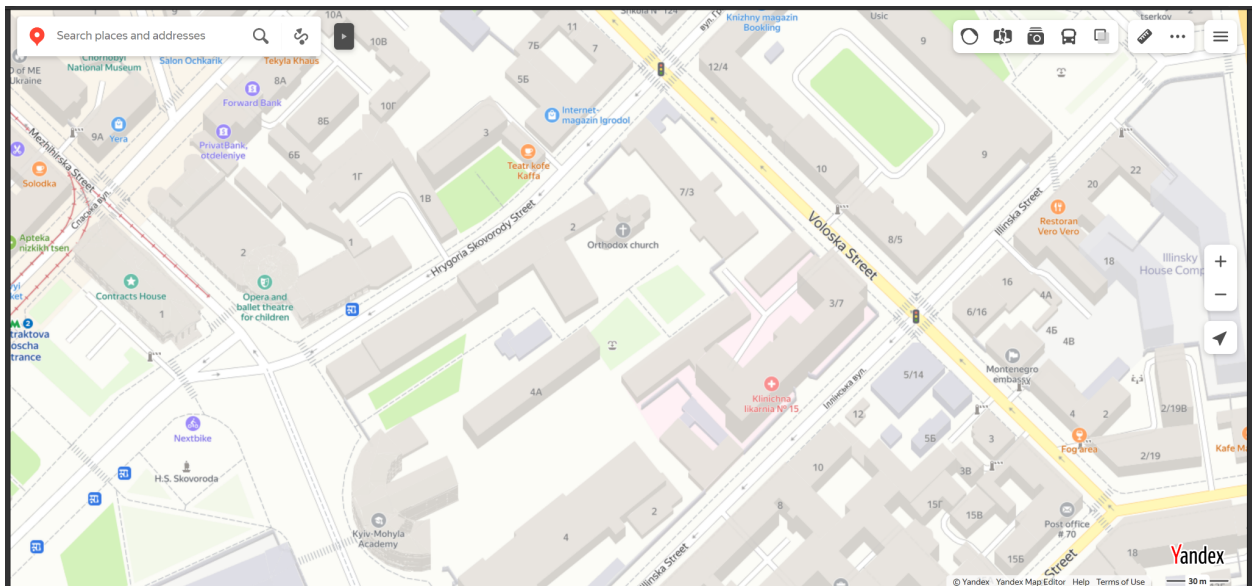
На основі Open Street Maps побудовані відомі системи, наприклад MAPS.ME, Wikipedia та Strava.

Yandex Maps

Сервіс розроблений російською корпорацією Яндекс. Для розробників компанія пропонує два програмних продукти - Yandex Maps API, що має потужні налаштування та використовується для

веб-розробки. Для створення мобільних додатків використовується Yandex Maps Kit. Також компанія використовує власні наробки для створення нових продуктів, як от Яндекс.Нерухомість.

Минулого року у Яндекс змінили цінову політику, а саме обмежили безкоштовне використання з 25 000 запитів на добу до 1 000 запитів на добу. Частину компаній-користувачів не попередили про зміни, що призвело до серії скандалів та відмов від використання сервісів Яндекс. [\[4\]](#)



Додаток 1.1.3 Yandex Maps

Попри це, основною перевагою цього провайдера є висока якість даних для регіонів СНГ, що полюбилося користувачам та використовується, навіть після заборони роботи сервісів Яндекс в Україні.

MapBox

MapBox - ще один відомий постачальник картографії. Їх продукт один з найпотужніших, на рівні з Google Maps. Це векторні карти, що побудовані за принципом нашарування візуальної інформації. Їх карти широко застосовуються розробниками різних калібрів, зокрема, відомих Facebook, Porsche, компаній у сфері логістики, як DHL, Instacart та інших. [\[5\]](#) [\[6\]](#)

Окремої уваги заслуговує продукт для кастомізації мап та стилів - MapBox Studio, що має багато можливостей та налаштувань. [\[7\]](#) Його інтерфейс нагадує Photoshop, де ви можете обрати свій унікальний дизайн, обравши потрібні кольори, шрифти, тощо та вигляд різноманітних географічних об'єктів, від зелених насаджень до державних кордонів. MapBoxStudio можна запустити просто в браузері, змінювати та спостерігати зміни в реальному часі, а створений стиль можна застосувати для всіх платформ.

Є унікальна платформа для розробки додатків у доповненій реальності, де реалізовано прив'язка об'єктів до локацій. Підтримується розробка у середовищі Unity та React Native. [\[8\]](#)

У MapBox є безкоштовна версія, що дозволяє використання сервісу до 25 000 активних користувачів щомісяця для Maps SDKs for Mobile та до 50 000 користувачів для веб-ендпоінтів, відповідно.

Apple Maps

Apple Inc. має власну розробку для відображення та роботи з геофізичними картами. Раніше вони використовували технології Google

Maps, але у 2012 було випущено Apple Maps, а згодом і набір інструментів для розробників - MapKit. [\[9\]](#) Лише у 2018 було реалізовано можливість інтеграції мап для сторонніх сайтів-розробників. Хоча деякі ресурси, як-от пошуковик DuckDuckGo і раніше використовували Apple Maps у своїх розробках.

Фреймворк від Apple пропонує вбудоване відображення мап, можливість кастомізації кольорів та маркерів. Є вбудована навігація (недоступно в деяких регіонах), API для пошуку місць - Points of Interest, що працює як за назвою, так і за розташуванням.

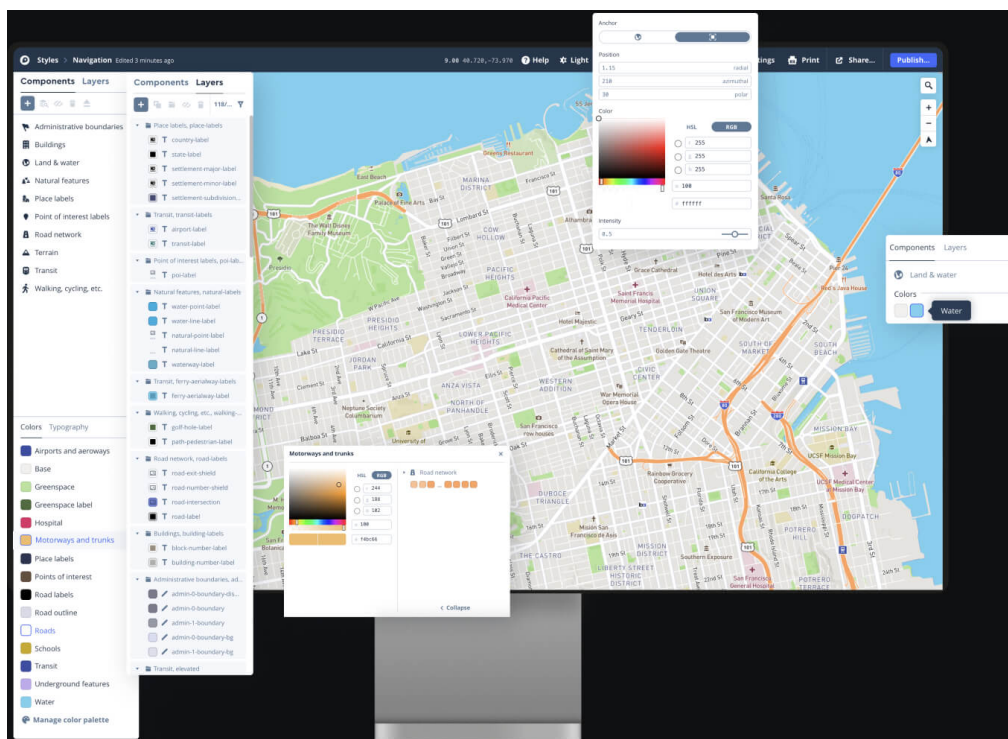
Також 2019 року було створено Street View, власний аналог Google Street View. Сервіс також пропонує до огляду кругові панорами та перегляд доріг. [\[10\]](#)

1.2 Порівняння картографічних сервісів.

Функціональність відображення мапи

Підтримується всіма провайдерами карт. Apple Kit та Google Maps SDK мають нативну підтримку на власних платформах та при розробці додатків не вимагає великих завантажень. Платформи не особливо відрізняються відображенням та пропонують схожий функціонал - відображення доріг, зелених насаджень, будинків.

З відмінного, деякі провайдери мають кращу деталізацію. Наприклад Яндекс у СНД відображає навіть нумерацію під'їздів, а OpenStreetMap, котрий якісніше має потужнішу базу доріг та об'єктів, завдяки ком'юніті.

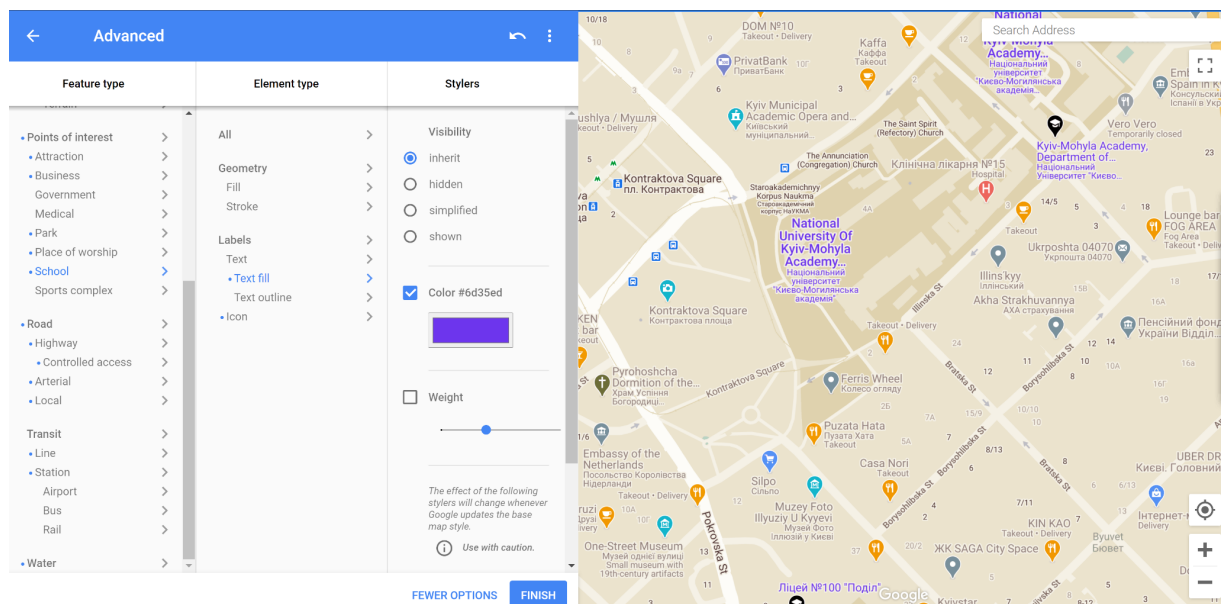


Додаток 1.2.1. MapBox Studio, редагування стилів мапи

Для відображення об'єктів використовуються маркери, що можуть бути замінені на зображення пропоновані користувачем.

Найконсервативнішим в контексті кастомізації є Apple Maps, але цього функціоналу достатньо для відображення декількох стилів та шарів.

Google Maps та MapBox мають власні платформи для створення стилів відображення. У налаштуваннях ви можете змінити відображення практично будь-яких об'єктів, їх форму, розміри та кольори. Google Maps також підтримує навігацію у приміщеннях, як от аеропорти чи торгові центри.



Додаток 1.2.2 Редагування стилів у Google Maps

Навігація

Навігація є однією з основних функцій геосервісів. Вона реалізована за допомогою REST API, що приймає дві, або більше точок та повертає складний об'єкт маршруту. Маршрути також доступні для побудови офлайн. Google Maps Platform та Яндекс пропонують функціонал для відображення завантаженості доріг в реальному часі, та використовують ці дані для побудови оптимальних маршрутів. Apple Maps з 2017 року теж відображає трафік на дорогах України, використовуючи сторонній TomTom Traffic. [\[11\]](#)

Веломаршрути. Жоден з провайдерів фактично не пропонує підтримку веломаршрутів, хоча Apple Maps та Google мають такий функціонал у інших країнах.

Мультиплатформенність

Усі оглянуті платформи пропонують інтеграції з різними платформами. Google Maps Platform має SDK для обох мобільних систем. Також є Flutter SDK для впровадження Google карт [\[12\]](#). OpenStreetMap має окремий OverPass API, що реалізує доступ для відображення мап та деталей на них. Для роботи з мобільними додатками OSM пропонує ряд бібліотек, як для IOS, так і для Android. [\[13\]](#)

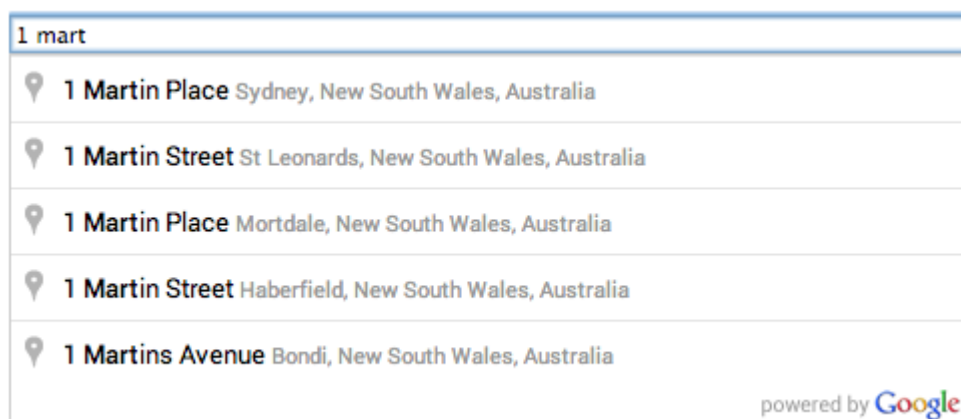
Apple Kit пропонується тільки для пристроїв Apple, та для відображення карт на сайтах.

Графічна бібліотека, яку пропонує MapBox, працює водночас як для персональних комп'ютерів, так і для мобільних пристроїв чи

веб-сайтів. Також, як і Google [14], MapBox постачає програмне забезпечення для автомобілів [15]. є лідером з кількості можливого функціоналу при роботі з мапами.

Пошуковий функціонал

Кожен карт-провайдер має можливість пошуку, як за назвою, так і за фізичними координатами. Цей функціонал реалізують за допомогою REST API, де у форматі JSON сервер повертає інформацію про цікаві місця - різноманітні заклади, автозаправки чи то зони для відпочинку. При виборі карт-провайдера важливо розуміти чи будете ви використовувати пошук, та порівняти покриття і функціональність.



Додаток 1.2.3 Автодоповнення пошуку з використанням Places API

Google Places API, про який було зазначено у огляді технологій є одним з лідерів у пошуку та кількості місць. Також доступні перегляд відгуків, побудова маршрутів та підтягування фотографій за результатами пошуку.

Складність впровадження

В залежності від бажаного функціоналу на це питання можна відповісти по-різному. Зважаючи на технічне завдання, у цьому пункті коротко порівняно якість документації для технологій відображення мап, пошуку та збереження точок і навігації, наявність прикладів і підтримку ком'юніті. Документація в Google Maps та MapBox порівняно однаково вичерпна, але Google Maps має більшу кількість прикладів через свою популярність.

Для імплементації карт у мобільний додаток необхідно спершу встановити фреймворк для їх відображення. Для доступу до API використовується спеціальний ключ-токен, його необхідно отримати на сайті геопровайдера та вказати в додатку у спеціальному місці, аби досягнути серверу.

OpenStreetMaps містить у собі веб-застосування, що ви можете розмістити на власному сервері. Для простіших застосувань та додавання мап на сайти є OverPass, котрий не вимагає ключів та може відповідати на запити.

Висновки

Серед оглянутих технологій немає сервіса, котрий однозначно є кращим. Під час вибору та впровадження картографії у розробку слід враховувати функціональні вимоги. Важливо також врахувати навантаженість та оцінити витрати. У цій роботі цінові політики не оглядались, але було зауважено, що MapBox у безкоштовному плані пропонує підтримку до 50 000 запитів на місяць.

OSM пропонує потужний функціонал, і є повністю open-source. Водночас це означає, що ви маєте запустити власний сервер для взаємодії з мапою. Як було зазначено, також є простіший функціонал, що реалізований у окремому API.

Google Maps Platform є найпоширенішим рішенням для імплементації карт. Основні переваги - зручність імплементації, велика кількість прикладів та можливість кастомізації. Також пропонується Places API, що є фактично хранилищем даних про різноманітні заклади та місця, зі вбудованими відгуками та фотографіями. Щоправда, цей сервіс недоступний у безкоштовній версії, а запити коштують від 5 до 30 доларів за 1000, що досить дорого. Але у цей же час, Google Maps Platform для відображення мап є повністю безкоштовним для необмеженої кількості мобільних пристроїв.

Зважаючи на наявність прикладів та документації, та достатній функціонал для задоволення технічних вимог до продукту, було вирішено використати безкоштовну версію Google Maps Platform і як провайдера геоданих.

2. Технічне завдання та інструменти

2.1 Аналіз проблеми

Було проаналізовано декілька картографічних додатків, зокрема поширені Google Maps, OpenStreetMaps, Waze, Bing Maps. Це складні додатки, що мають велику кількість функцій. Під час дослідження було обрано просту схему - під час прогулянки зберегти кілька точок по маршруту та поділитися ними.

Серед наведених додатків такий функціонал представлено у OpenStreetMaps, з можливістю додати точку. Меню додавання містить можливість додавання маркеру. Також у додатку наявна функція створення списків, для збереження місць. При збереженні такого місця є можливість додати опис та мультимедіа файли. Інтерфейс цього додатку є не очевидним, і дещо перевантаженим.

У мобільній версії Google Maps - після довгого натискання додаток відображає координати широти та довготи, котрі можна скопіювати та використати на власний розсуд.

Під час аналізу проблеми з науковим керівником, було узгоджено проблему та персону, для кого ми розробляємо додаток.

Персона - Микола Іванович, 45 років. Працівник заводу побутових приладів. Микола з товаришами щоосені вирушає з колегами відпочивати до Чорного лісу, де вони шукають гриби та мешкають у наметах. Цього року Микола поїхав раніше, аби знайти влучне місце

для намету та паркування автівок. Кожного року це є проблемою - важко орієнтуватися у лісі, а пошуки автівки взагалі перетворюються на муки. Такий додаток може значно спростити життя Миколи, запаркував машину - збережи точку, розбив намет - збережи точку, треба зорієнтувати своїх колег - поділися точками у месенджері. Микола може здобути славу логістичного генія, а якщо зберігати точки, де були знайдені добрі гриби, то з кожним роком пошуки будуть ставати більш вдалимими!

2.2 Технічне завдання

Створити мобільний додаток, метою якого є можливість створення, збереження та перегляду точок на мапі, додавання назв та агрегації їх у списки. Також передбачається функція поділитися списком точок та відтворення таких отриманих списків.

Точки мають ім'я та опис, можливо впровадження збереження фотографій та інших мультимедіа файлів. Точки можна переглядати, змінювати та видаляти. Точка може належати тільки одному списку, та не може існувати окремо від списку.

Список точок має ім'я та опис, і може бути порожнім. Кожен список може містити необмежену кількість точок. Список можна редагувати, при відображенні списку слід відображати усі точки, що йому належать.

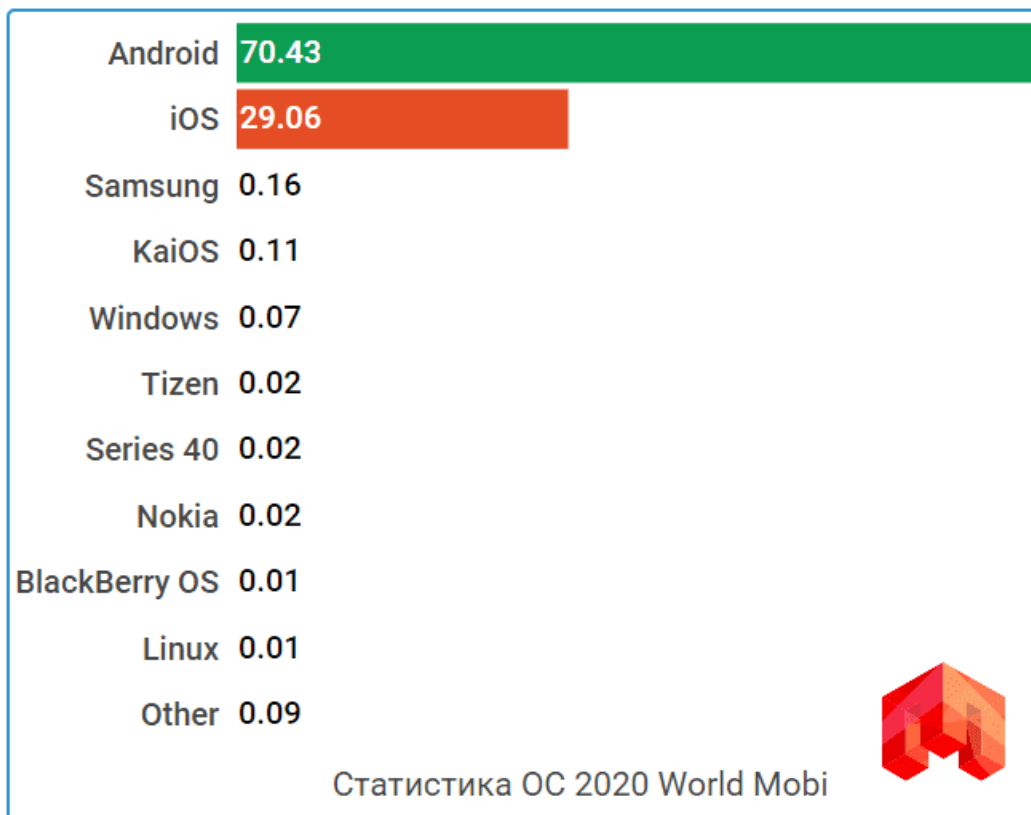
2.3 Вибір технологій

Огляд мобільних ОС

З розвитком технологій та плином часу пристрої зменшуються, а технології покращуються. Сьогодні ми носимо у кишені обчислювальні машини, що значно потужніші, за комп'ютери, що не так давно було важко навіть зрухати з місця.

Мобільні пристрої працюють за допомогою операційних систем, що встановлюються розробниками “з заводу”. Завдяки такому програмному забезпеченню ми з вами можемо використовувати велику кількість функцій, вже відразу після покупки, без додаткових налаштувань. [\[16\]](#)

Операційна система мобільного пристрою є зручним інтерфейсом взаємодії з апаратною частиною пристрою [\[16\]](#), крім цього забезпечуючи стабільну роботу. Найпоширенішими системами на сьогодні є Android, IOS та Windows Phone. Android та IOS займають сьогодні понад 95% ринку мобільних пристроїв, як в Україні так і по світу [\[17\]](#), і ця частка тільки зростає, тому ми враховували лише пристрої з цими ОС.



Додаток 2.3.1 Графік частки популярних мобільних систем, 2020

Порівняння розробки під IOS та Android

Apple розвиває потужну екосистему, що включає у себе як кілька лінійок комп'ютерів, так і носимі гаджети, крім телефонів це годинник, навушники та окуляри. Завдяки цій системі розробнику простіше орієнтуватися як в гайдлайнах, що визначаються Apple, так і в коді, що має спільні вимоги та навіть розробляється у тому ж середовищі.

Особливість цієї системи - замкненість, під IOS працюють лише пристрої випущені Apple, з комплектуючими від Apple, що позитивно впливає на ефективність використання ресурсів та безпеку. Водночас це стосується і середовища розробки - нативна розробка під IOS

можлива лише з Mac OS. Apple є значно більш консервативною у ставленні до сторонніх розробників. [\[18\]](#) Додатки мають проходити декілька стадій перевірок та відповідати вимогам гайдлайнів [\[19\]](#) та продуктивності.

Що стосується навчальних ресурсів, їх велика кількість доступна для обох платформ, однак через більшу розповсюдженість спільнота Android-розробників є ширшою.

До переваг системи від Google слід віднести відкритість та гнучкість. Основний інструмент розробника, Android Studio в доступі для всіх операційних систем. Створення додатків під цю платформу менш вимогливе у порівнянні з IOS.

Кросс-платформенна розробка

Також існує можливість розробляти додатки одночасно для обох систем, що з роками стає популярнішим. Першим серйозним кроком у цьому напрямі стали наробки компанії Xamarin, випущені 2011. Це open-source інструмент, що використовує мову C#. 2016 року, компанія була придбана Microsoft. Сьогодні частка цієї платформи спадає, а бібліотеки значною мірою застаріли.

Сучасними інструментами кроссплатформної розробки є React Native, Flutter та Kotlin Multiplatform. Основна перевага кросс-платформенної розробки - економія ресурсів на розробці повноцінних додатків. Зазвичай такий тип розробки використовується для розробки нескладних продуктів, та широко поширений серед стартапів.

Основною проблемою такого підходу є часткова втрата функціональності, пов'язана з частотою виходу оновлень [\[20\]](#) та підтримкою коду. Нативні додатки значно краще оперують функціоналом операційної системи, як-от сповіщеннями та роботою з картами зокрема.

Значно цікавіше виглядає Kotlin Multiplatform [\[21\]](#), що також дозволяє одночасно розробляти мобільні додатки для обох ОС. Його особливістю є нативна інтеграція для системи Android, розробка відбувається в Android Studio, а код для IOS генерується та підтримується у XCode. Цей інструмент ще дещо новий, але вже використовується такими відомими компаніями, як Netflix [\[22\]](#) та Memrise.

У підсумку, для розробки цього додатку цільовою системою було обрано Android OS, зокрема через, його кращу поширеність, вже знайому мову Kotlin для нативної розробки та наявність такого пристрою, за відсутності продукції від Apple.

3. Розробка додатку

3.1 Створення проекту

Розробка виконується у середовищі Android Studio 4.1 У розробці використовуються бібліотеки, зокрема Android Maps SDK. Для впровадження бібліотек слід змінити файл налаштувань Gradle. Gradle є інструментом для автоматизованої побудови та запуску додатку. Тут ви можете писати власні скрипти з використанням мови Groovy. У лістингу наведено імпорт усіх використаних бібліотек.

```
dependencies {
```

```
    // Room Database
```

```
    implementation "androidx.room:room-runtime:2.2.6"
```

```
    kapt "androidx.room:room-compiler:2.2.6"
```

```
    implementation 'androidx.room:room-ktx:2.2.6'
```

```
    // Navigation Component
```

```
    implementation 'androidx.navigation:navigation-fragment-ktx:2.3.5'
```

```
    implementation 'androidx.navigation:navigation-ui-ktx:2.3.5'
```

```
    // Google Maps Services
```

```
    implementation
```

```
    'com.google.android.gms:play-services-maps:17.0.0'
```

```
implementation  
'com.google.android.gms:play-services-location:18.0.0'
```

```
// Lifecycle components
```

```
implementation "androidx.lifecycle:lifecycle-extensions:2.2.0"
```

```
implementation "androidx.lifecycle:lifecycle-viewmodel-ktx:2.3.1"
```

```
implementation "androidx.lifecycle:lifecycle-common-java8:2.3.1"
```

```
// Coroutines
```

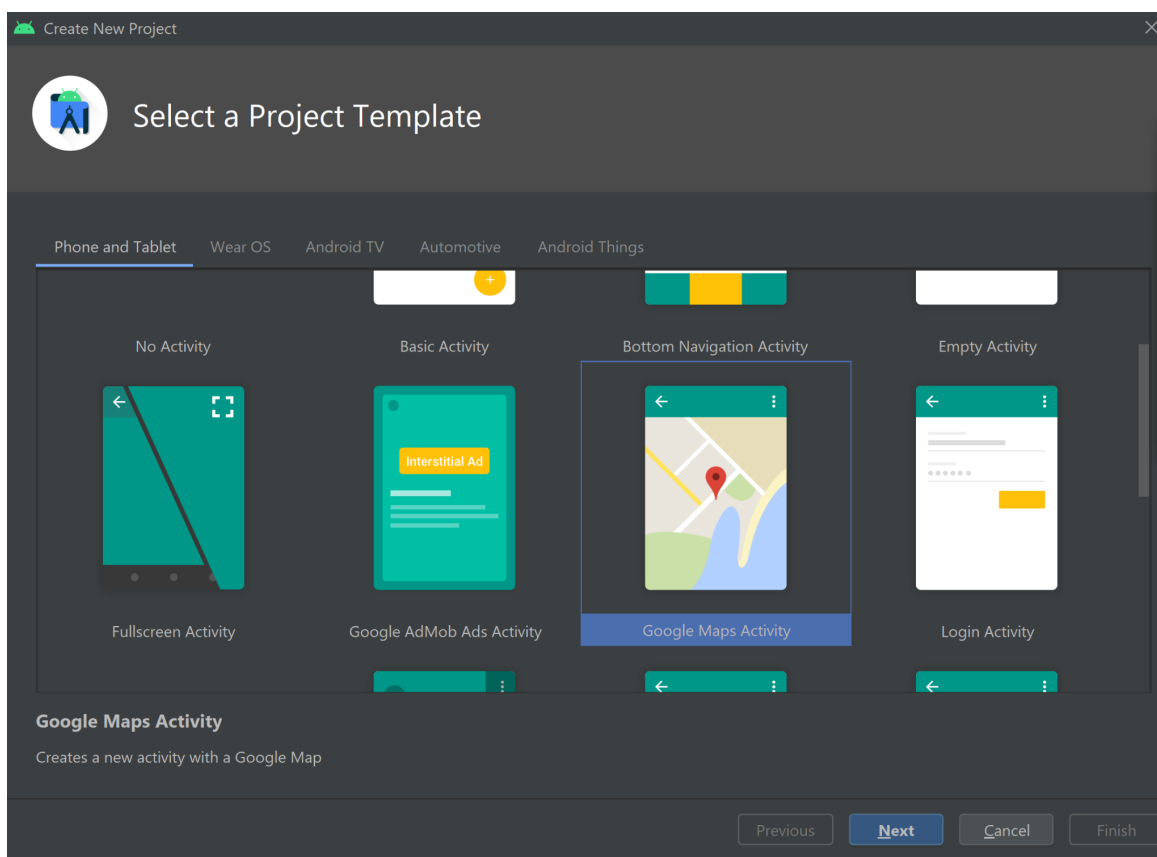
```
implementation 'org.jetbrains.kotlinx:kotlinx-coroutines-core:1.4.3'
```

```
}
```

3.2 Робота з мапою

Для додавання екрану з мапою можна під час створення проекту обрати Google Maps Activity, або додати після створення порожнього проекту. Після додавання екрану слід впровадження унікального ключа від Google Maps Platform.

Для впровадження мапи у ваш додаток слід виконати інструкцію з сайту Google Developers. [\[23\]](#) Отримати API Key та вписати його у відповідне поле у файлі налаштувань.



Додаток 3.2.1 Меню вибору Activity при створенні нового додатку

У Android розробка інтерфейсу полягає у взаємодії між програмним класом мовою Java/Kotlin, та файлом розмітки, зазвичай xml. Для впровадження мапи використовується fragment з бібліотеки Maps SDK. Фрагменти - один з основних елементів на рівні з Activity, що є контейнерами для інших візуальних компонентів. Лістинг коду XML-макету для Активіті з картою.

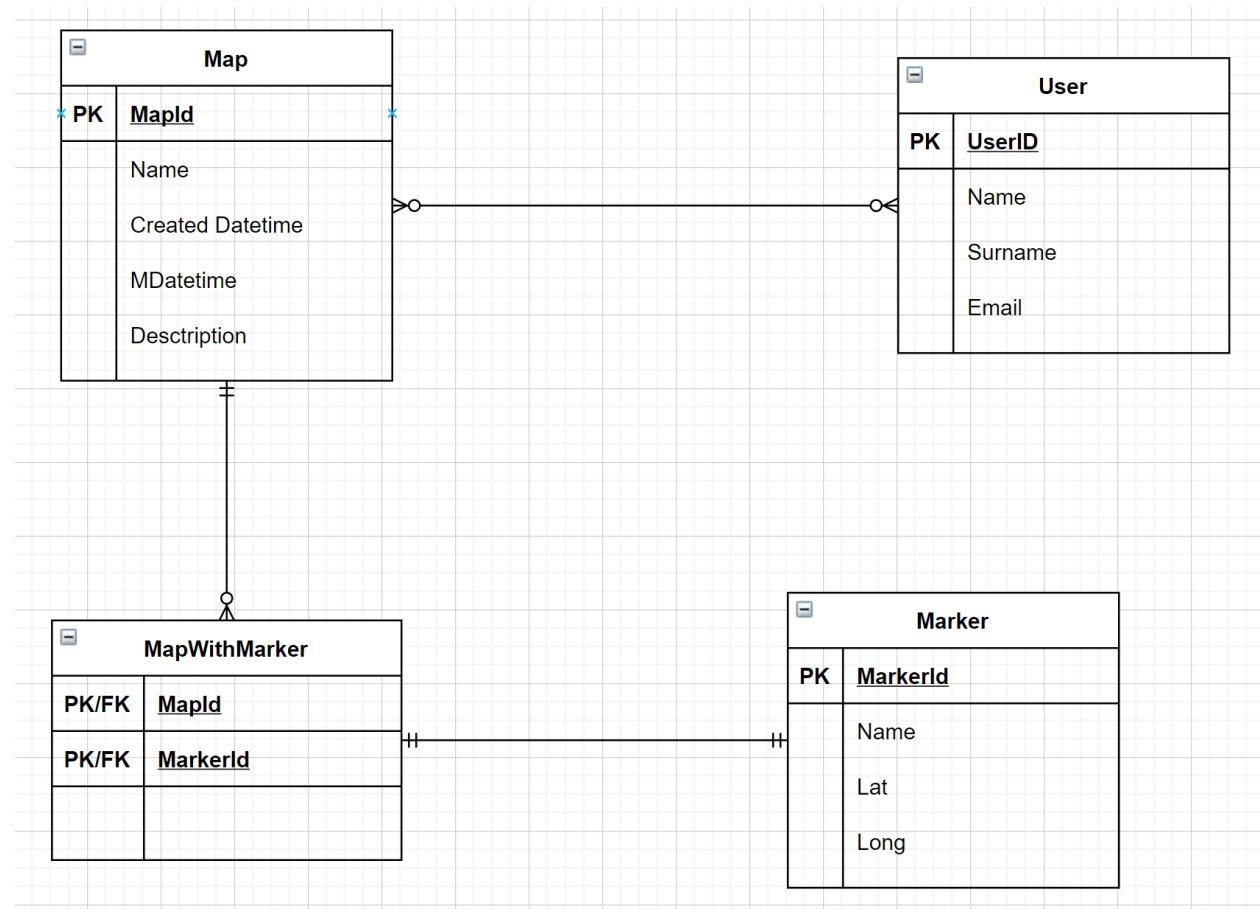
```
<?xml version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.ConstraintLayout
xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".MapActivity">

    <fragment
        android:id="@+id/map"
        android:name="com.google.android.gms.maps.SupportMapFragment"
        android:layout_width="match_parent"
        android:layout_height="0dp"
        app:layout_constraintBottom_toTopOf="@+id/guideline2"
        app:layout_constraintTop_toTopOf="parent"
        app:layout_constraintVertical_bias="0.0"
        tools:layout_editor_absoluteX="-25dp"
        tools:ignore="FragmentTagUsage" />
</androidx.constraintlayout.widget.ConstraintLayout>
```

3.3 Імплементация бази даних

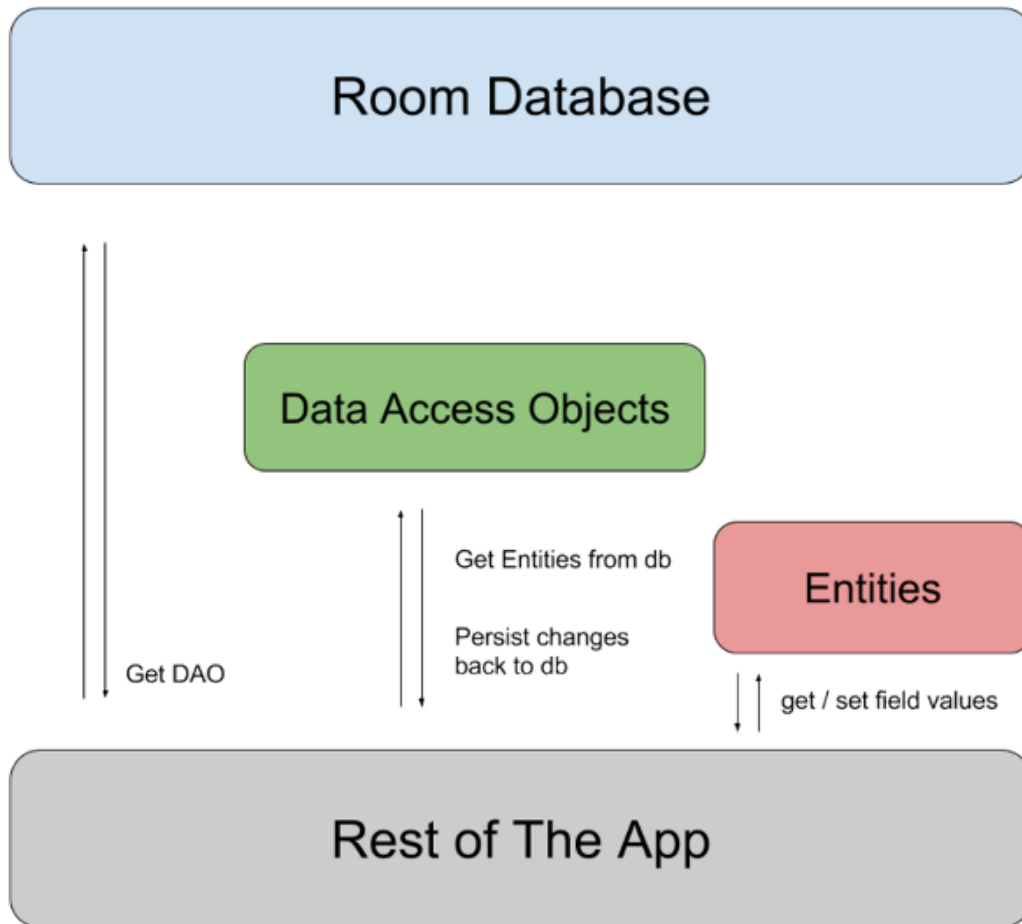
Схема бази даних

Поточна версія БД складається з лише таблиці Maps. База даних може бути доповнена, при впровадженні функцій додавання мультимедіа об'єктів зокрема.



Додаток 3.3.1 Діаграма сутностей бази даних

Для збереження точок на пристрої використовується бібліотека Room, що надає інтерфейс для взаємодії зі сховищем даних SQLite. Архітектура Room складається з трьох основних частин - база даних, об'єкти доступу даних (DAO) та таблиць.



Додаток 3.3.2. Діаграма архітектури бібліотеки Room [\[24\]](#)

Архітектура Room

Це фактично класи таблиць, що зберігають об'єкти даних. Для реалізації функціоналу Room використовуються анотації, що опрацьовуються на етапі побудови додатку. У додатку Лістинг коду 1

DAO є інтерфейсом, що містить методи для виклику CRUD-запитів до бази даних. На відміну від прямих викликів, завдяки DAO Room забезпечує відокремлення представлення даних від бізнес-логіки, що є одним з фундаментальних засад розробки ПЗ, та Android додатків зокрема.

У анотації до методів вказано тип запиту, його SQL-код та додаткові параметри [25]. Додатки лістинг 1 та лістинг 2 містять код таблиці Map та інтерфейсу DaoMaps.

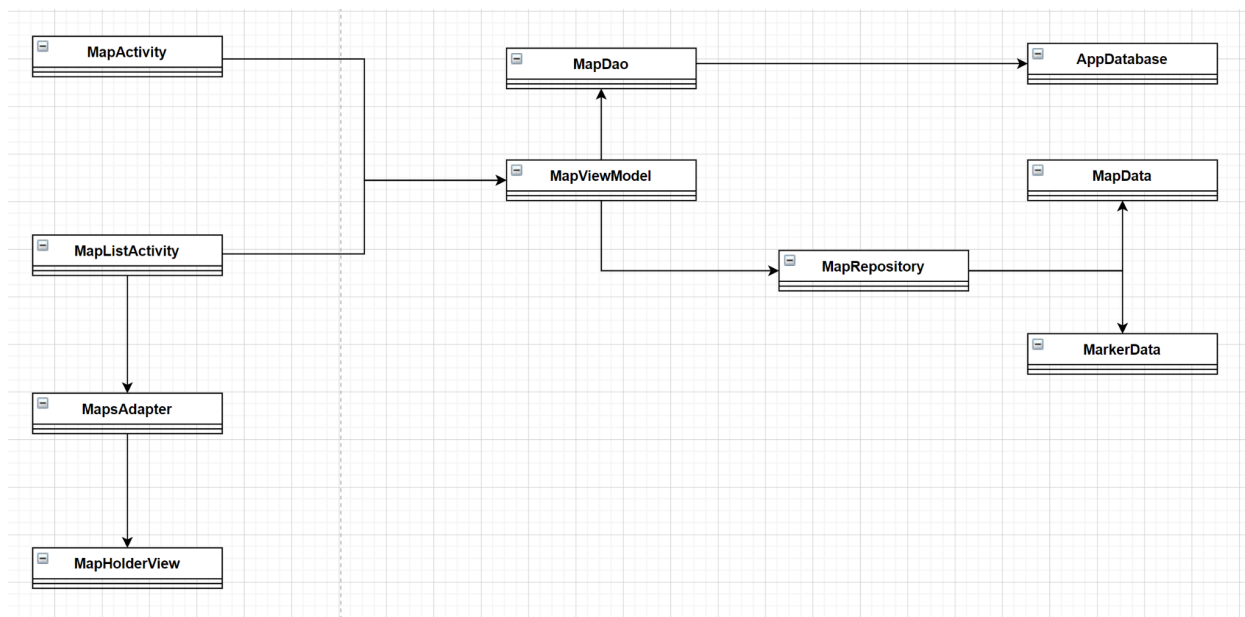
База даних є файлом, що зберігається у пам'яті пристрою. Робота з даними відбувається за схемою Model-View-Controller. Доступ до даних та виклики Dao реалізовані у класі MapRepository.

Один з запитів інтерфейсу Dao:

```
@Update(onConflict = OnConflictStrategy.REPLACE)
```

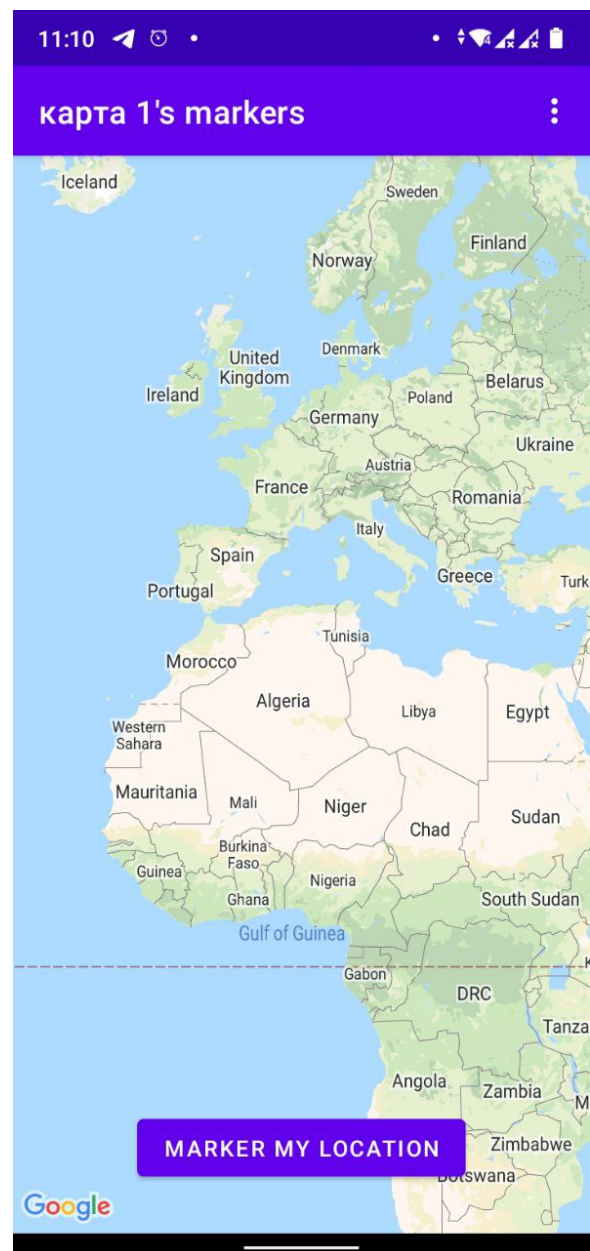
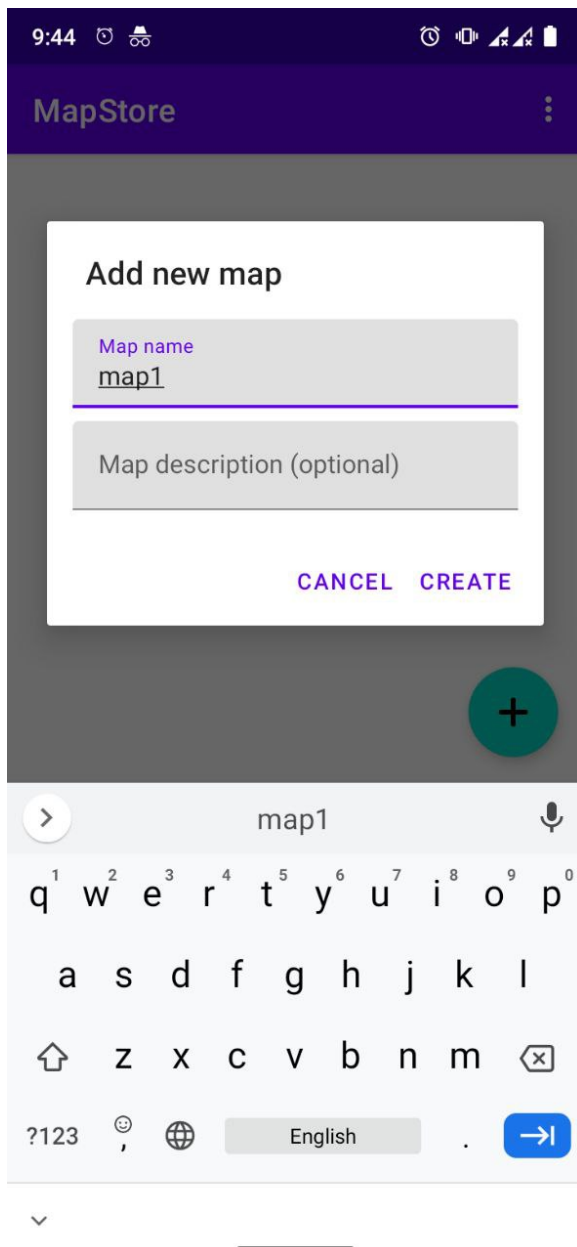
```
suspend fun update(map: MapData)
```

3.4 Діаграма класів застосунку



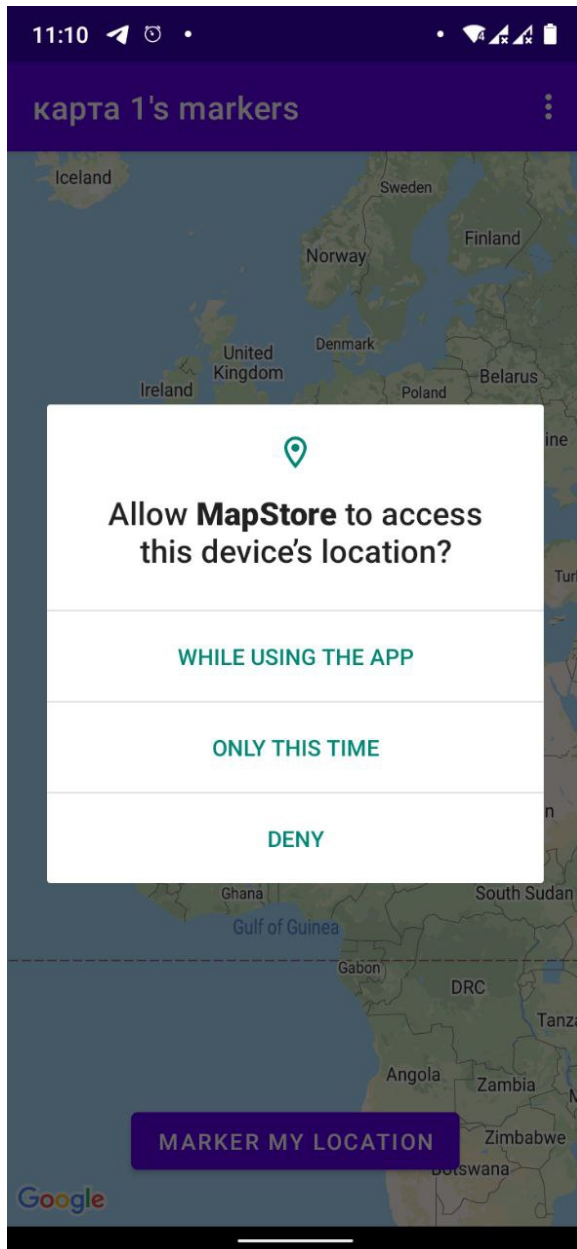
Додаток 3.4.1

3.5 Скріншоти роботи програми

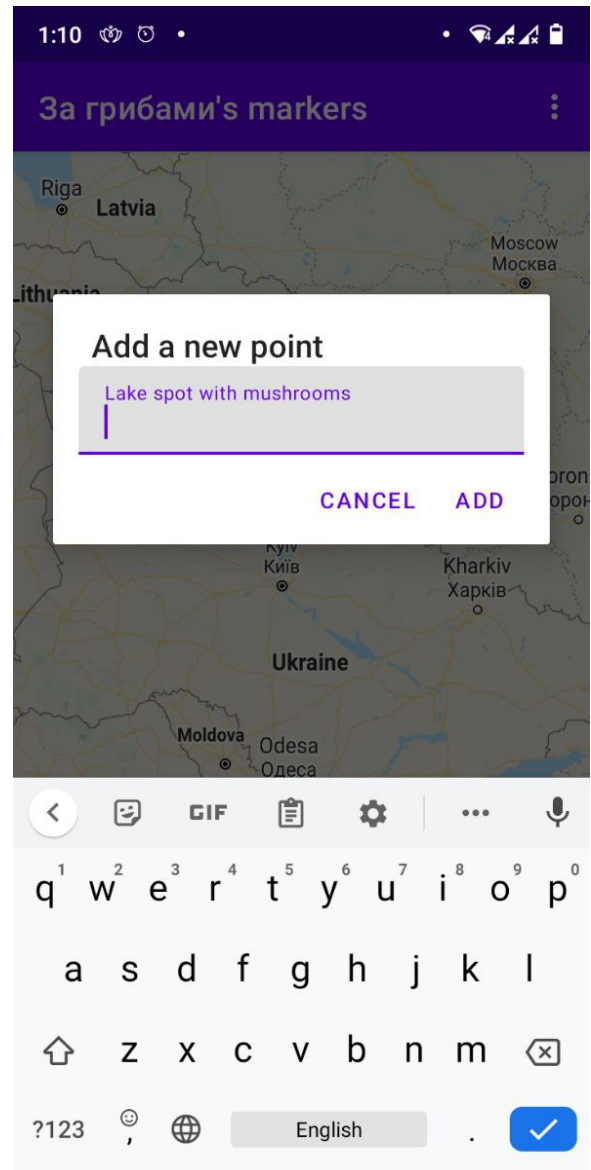


Додаток 3.5.1 Діалог створення нової карти

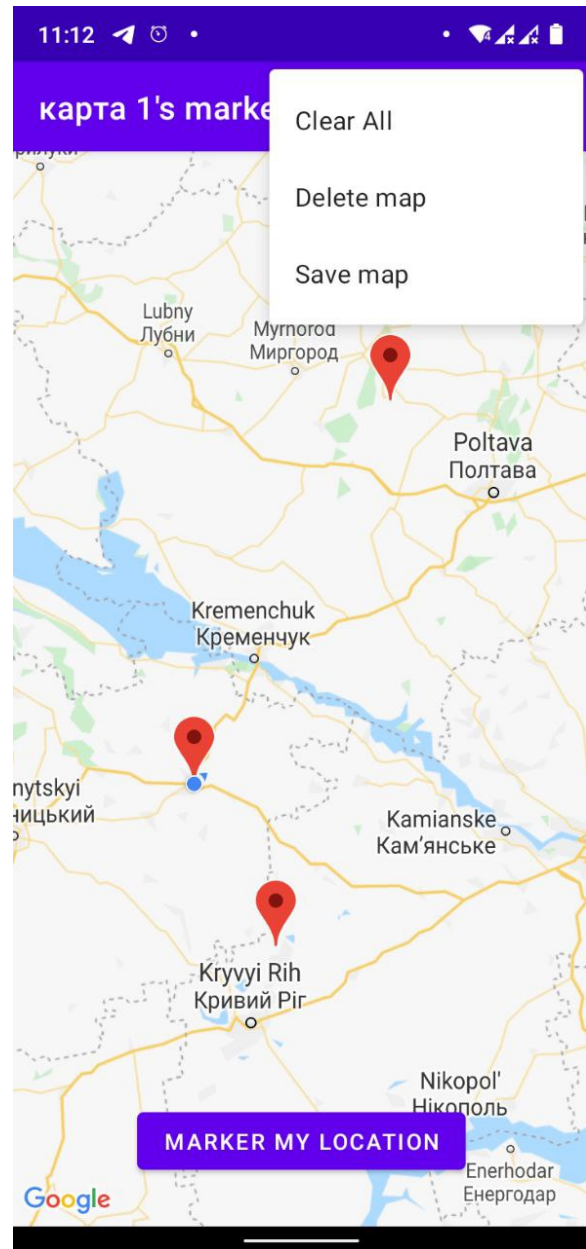
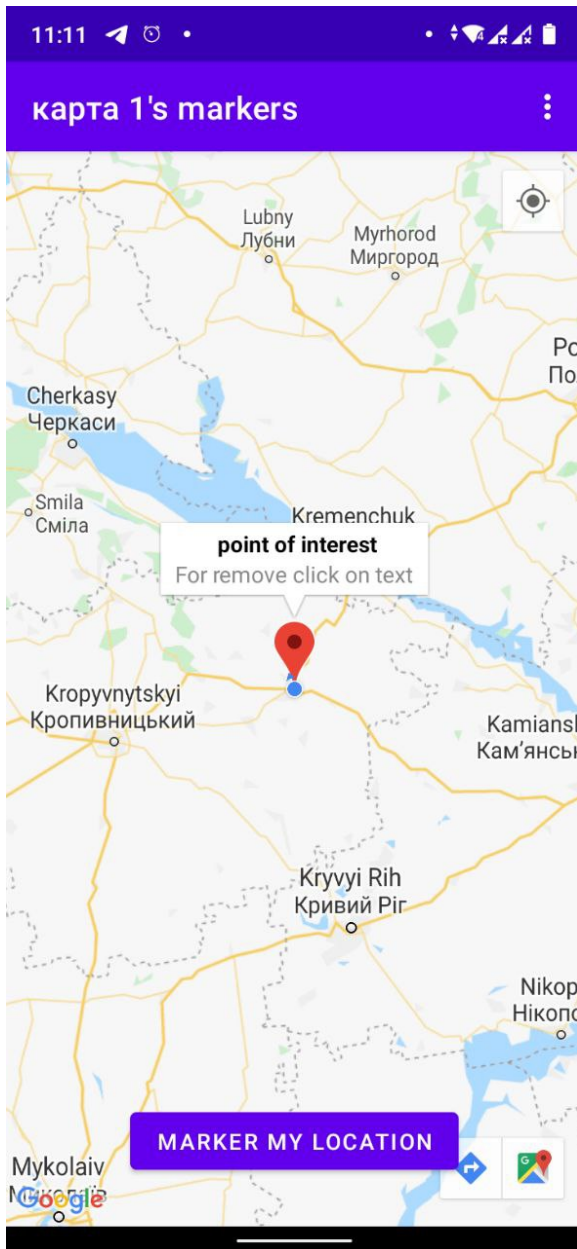
Додаток 3.5.2. MapActivity з порожньою картою



Додаток 3.5.3. Діалог дозволу використання локації

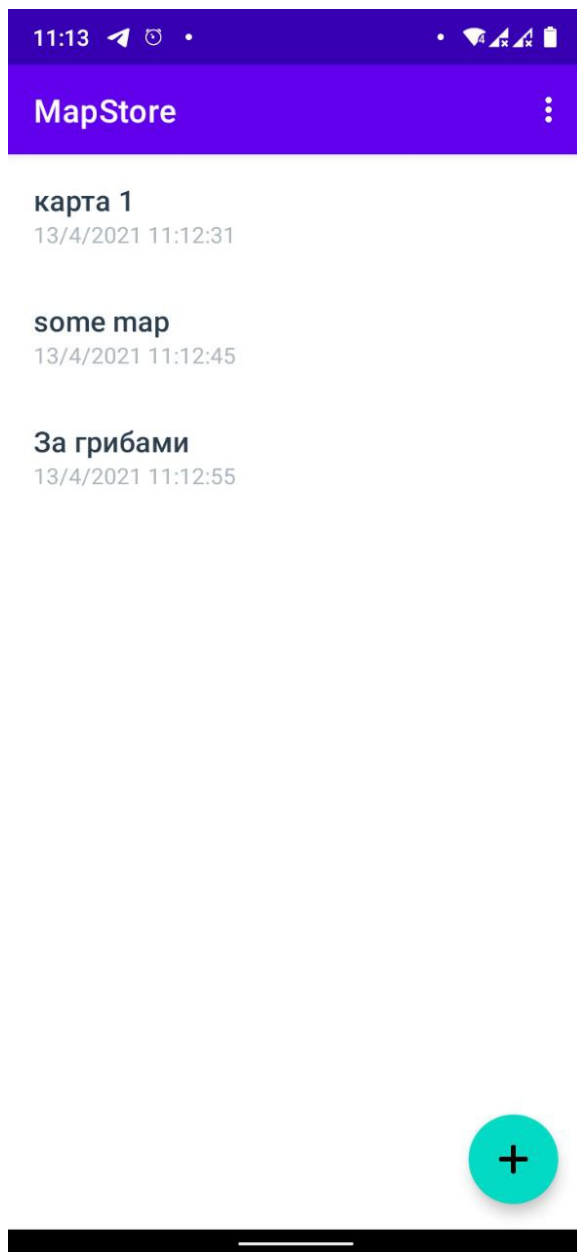


Додаток 3.5.4. Діалог додавання нової точки



Додаток 3.5.5 Новостворена точка на карті

Додаток 3.5.6 Меню у MapActivity



Додаток 3.5.7 MapsListActivity, з декількома мапами

Висновок

Глобальні зміни у сфері інформаційних технологій вплинули також і на картографію. Вже 10 років тому за допомогою мобільного пристрою ви могли будувати маршрути між містами, або ж переглядати панорами вулиць. Під час дослідження було оглянуто технології різних сучасних провайдерів геоданих.

Було вирішено розробити додаток для мобільних пристроїв, що має спростити орієнтацію в просторі та дати простий потрібний функціонал для грибників та інших туристів та дослідників. Для розробки було використано картографічний провайдер Google Maps, через потужні можливості та вичерпний функціонал безкоштовної версії API.

Додаток розроблено з використанням інструменту Android Studio, логіка додатку реалізована мовою Kotlin. Для зберігання даних використовується SQLite база даних, що імплементована бібліотекою Room. Для викликів асинхронного коду в проекті слугують модні Kotlin Coroutines, а сам додаток спроектований за принципами Clean Architecture.

При розробці картографічного застосунку завжди виникають певного роду труднощі, зокрема питання вибору технологій. Слід враховувати функціональні вимоги до картографії та навантаженість системи для того, аби отримати якісне рішення і водночас не використовувати ресурси надмірно.

%D0%BD%D0%B0%D1%8F_%D1%81%D0%B8%D1%81%D1%82
 %D0%B5%D0%BC%D0%B0

17. <https://marketer.ua/stats-operating-system-2020/>
18. <https://easternpeak.com/blog/android-vs-ios-development-which-platform-first/>
19. <https://developer.apple.com/design/human-interface-guidelines/ios/overview/themes/>
20. <https://dzone.com/articles/pros-and-cons-of-cross-platform-app-development>
21. <https://kotlinlang.org/lp/mobile/>
22. <https://netflixtechblog.com/netflix-android-and-ios-studio-apps-kotlin-multiplatform-d6d4d8d25d23>
23. <https://developers.google.com/maps/documentation/android-sdk/start?hl=ru>
24. <https://developer.android.com/training/data-storage/room>
25. <https://developer.android.com/training/data-storage/room/accessing-data>

Додатки

Лістинг програмного коду 1

```
package com.example.mapstore.model

import androidx.room.*

@TypeConverters(MapData.DataConverter::class)
@Entity(tableName = "maps")
data class MapData(

    @PrimaryKey(autoGenerate = true) var id: Int = 0,

    @ColumnInfo var name: String,

    @ColumnInfo var description: String,

    @ColumnInfo var createdDatetime: String,

    @ColumnInfo var pointsMap: List<MarkerData>

)
```


Лістинг програмного коду 2

```
package com.example.mapstore.dao

import androidx.lifecycle.LiveData
import androidx.room.*
import com.example.mapstore.model.MapData

@Dao
interface MapDao {

    @Query("SELECT * FROM maps")
    fun getAll(): LiveData<List<MapData>> // list on youtube

    @Query("SELECT * FROM maps WHERE id =(:mapId)")
    fun getById(mapId: Int): MapData

    @Insert
    suspend fun insert(map: MapData)

    @Update(onConflict = OnConflictStrategy.REPLACE)
    suspend fun update(map: MapData)
```

```
@Query("DELETE FROM maps WHERE id =(:mapId)")
```

```
suspend fun deleteMap(mapId: Int)
```

```
@Query("DELETE FROM maps")
```

```
suspend fun deleteAllFromMaps()
```

```
}
```