

Міністерство освіти і науки України
Тернопільський національний технічний університет імені Івана Пулюя

Факультет комп'ютерно-інформаційних систем і програмної інженерії
(повна назва факультету)

Кафедра комп'ютерних наук
(повна назва кафедри)

КВАЛІФІКАЦІЙНА РОБОТА

на здобуття освітнього ступеня

бакалавр

(назва освітнього ступеня)

на тему: Розробка мобільного застосунку «Інтерактивна карта для туриста»

Виконав: студент IV курсу, групи СНС-42

спеціальності 122 Комп'ютерні науки

(шифр і назва спеціальності)

Яремко І.В.
(підпис) (прізвище та ініціали)

Керівник Ясній О.П.
(підпис) (прізвище та ініціали)

Нормоконтроль Марценко С.В.
(підпис) (прізвище та ініціали)

Завідувач кафедри Боднарчук І.О.
(підпис) (прізвище та ініціали)

Рецензент Бойко І.В.
(підпис) (прізвище та ініціали)

Тернопіль
2024

Міністерство освіти і науки України
Тернопільський національний технічний університет імені Івана Пулюя

Факультет комп'ютерно-інформаційних систем і програмної інженерії
(повна назва факультету)

Кафедра комп'ютерних наук
(повна назва кафедри)

ЗАТВЕРДЖУЮ

Завідувач кафедри

Боднарчук І.О.
(підпис) (прізвище та ініціали)

«__» червня 2024 р.

**ЗАВДАННЯ
НА КВАЛІФІКАЦІЙНУ РОБОТУ**

на здобуття освітнього ступеня Бакалавр
(назва освітнього ступеня)

за спеціальністю 122 Комп'ютерні науки
(шифр і назва спеціальності)

Студенту Яремко Ігор Васильович
(прізвище, ім'я, по батькові)

1. Тема роботи Розробка мобільного застосунку «Інтерактивна карта для туриста»

Керівник роботи Ясній Олег Петрович, доктор технічних наук, професор кафедри МН
(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)

Затверджені наказом ректора від «29» квітня 2024 року № 4/7-472

2. Термін подання студентом завершеної роботи 24 червня 2024р.

3. Вихідні дані до роботи Наукові публікації, електронні ресурси, підручники, посібники

4. Зміст роботи (перелік питань, які потрібно розробити)
Вступ. Розділ 1. Аналіз предметної області. 1.1 Аналіз ринку цифрових карт. 1.2 Аналіз існуючих рішень. 1.3 Визначення потреб користувачів. 1.4. Висновки до розділу. Розділ 2. Проектна частина. 2.1 Огляд середовища розробки 2.2 Формалізація вимог до системи та моделювання 2.2.1 Вимоги до програмного застосунку. 2.2.2 Вимоги до інтерфейсу. 2.2.3 Вимоги до Бази Даних. 2.3 Висовки до розділу. Розділ 3. Практична частина. 3.1 Імпортування Maps SDK. 3.2 Розробка на базі Maps SDK. 3.3 Висновки до розділу. Розділ 4. Безпека життєдіяльності, основи охорони праці. 4.1 Природне середовище і його забруднення. 4.2 Гігієнічні вимоги до організації та обладнання робочих місць з ВДТ. 4.3 Висновки до розділу. Висновки. Перелік джерел посилання.

5. Перелік графічного матеріалу (з точним зазначенням обов'язкових креслень, слайдів)
1. Тема. 2. Мета роботи. 3. Аналіз ринку. 4-5. Огляд існуючих рішень. 5. Вимоги до застосунку. 6. Огляд середовища розробки. 7. Імпортування Maps SDK. 8-11. Розробка Застосунку.

АНОТАЦІЯ

Створення мобільного застосунку інтерактивна карта для туриста // Кваліфікаційна робота освітнього рівня «Бакалавр» // Яремко Ігор Васильович // Тернопільський національний технічний університет імені Івана Пулюя, факультет комп'ютерно-інформаційних систем і програмної інженерії, кафедра комп'ютерних наук, група СНс-42 // Тернопіль, 2024 // С. 55, рис. - 14, бібліогр. - 36.

Ключові слова: мобільний застосунок, цифрові карти, maps SDK, база даних, ОС Android, kotlin, android studio.

Кваліфікаційна робота присвячена розробці мобільного застосунку інтерактивна карта для туриста. В першому розділі здійснено опис предметної області. Проведено аналіз ринку цифрових карт, та темпи його росту. Далі було проведено детальний аналіз існуючих рішень на ринку. Це допомогло оцінити конкуренцію і можливості для розробки.

У Другому розділі зосереджено увагу на проектній частині, включаючи огляд середовища розробки та формування вимог до системи. Було обрано найзручніше програмне середовище, описано вимоги до застосунку, а також визначена мова програмування.

Третій розділ буд призначений для практичної частини. Він охоплював імпортування та розробку додатка на базі Maps SDK. Це включало створення власного API ключа, встановлення необхідних залежностей, імпортування модулів та налаштування середовища розробки. Після цього була проведена розробка, використовуючи можливості даного SDK.

ANNOTATION

Development of a mobile application: interactive map for tourists // Qualification work of the educational level "Bachelor" // Yaremko Ihor Vasylovych// Ternopil Ivan Pulyu National Technical University, Computer and Information Systems and Software Engineering Faculty, Computer Sciences Department, group Snc-42 // Ternopil, 2024 // P. 55, fig. - 14, references - 36.

Keywords: mobile application, digital maps, maps SDK, database, android OS, kotlin, android studio.

The qualification work is dedicated to developing a mobile application: an interactive map for tourists. The first chapter provides an overview of the subject area, including an analysis of the digital maps market and its growth rates. A detailed analysis of existing solutions on the market follows, helping to assess competition and development opportunities.

The second chapter focuses on the project aspects, covering an overview of the development environment and requirements elicitation. The most suitable development environment was chosen, application requirements were described, and the programming language was selected.

The third chapter is dedicated to the practical part, encompassing the import and development of the application based on Maps SDK. This includes creating a custom API key, installing necessary dependencies, importing modules, and configuring the development environment. Development was then conducted utilizing the capabilities of the SDK.

**ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ,
СКОРОЧЕНЬ І ТЕРМІНІВ**

БД – База даних

API – Application Programming Interface

GPS – Global Positioning System.

SDK – software Development Kit

UML – Unified Modeling Language

ЗМІСТ

ВСТУП	7
РОЗДІЛ 1. АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ ТА ПОСТАНОВКА ЗАВДАННЯ.....	9
1.1 Аналіз ринку цифрових карт.....	9
1.2 Аналіз існуючих рішень	14
1.3 Визначення потреб користувачів.....	19
1.4 Висновки до першого розділу.....	20
РОЗДІЛ 2. ПРОЕКТНА ЧАСТИНА	22
2.1 Огляд середовища розробки.....	22
2.2 Формалізація вимог до системи та моделювання.....	26
2.2.1 Вимоги до програмного застосунку	29
2.2.2 Вимоги до інтерфейсу.....	30
2.2.3 Вимоги до Бази Даних	31
2.3 Висновки до розділу.....	32
РОЗДІЛ 3. ПРАКТИЧНА ЧАСТИНА	33
3.1 Імпортування Maps SDK.....	33
3.1 Розробка на базі Maps SDK	36
3.3 Висновки до розділу.....	42
РОЗДІЛ 4. БЕЗПЕКА ЖИТТЄДІЯЛЬНОСТІ, ОСНОВИ ОХОРОНИ ПРАЦІ	43
4.1 Природне середовище і його забруднення	43
4.2 Гігієнічні вимоги до організації та обладнання робочих місць з ВДТ.....	46
4.3 Висновки до розділу.....	49
ВИСНОВКИ.....	50
ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ	52

ВСТУП

Актуальність теми. В сучасному світі мобільні технології стали невід'ємною частиною повсякденного життя, і туристична галузь не є винятком. З розвитком глобалізації та збільшенням доступності подорожей зростає попит на інформаційні технології, які можуть забезпечити туристів необхідними даними під час перебування в нових місцях. Мобільні додатки для туристів, такі як інтерактивні карти, відіграють ключову роль у цьому процесі, оскільки вони надають можливість швидко отримувати інформацію про визначні пам'ятки, ресторани, готелі та інші важливі об'єкти.

Серед багатьох наявних рішень, таких як Google Maps, туристи часто стикаються з обмеженнями, зокрема недостатньою адаптацією до конкретних потреб мандрівників, відсутністю інтеграції з місцевими сервісами та нестачею можливостей для взаємодії з іншими користувачами. Відповідно, розробка спеціалізованого мобільного застосунку, який би враховував ці потреби та пропонував додаткові функції, є надзвичайно актуальною.

Така інтерактивна карта може значно покращити досвід подорожей, надаючи більш детальну та корисну інформацію, допомагаючи туристам орієнтуватися в незнайомих місцевостях, знаходити приховані перлини та унікальні місця, а також ділитися своїм досвідом з іншими мандрівниками. Це особливо важливо в умовах пандемії, коли питання безпеки та актуальної інформації стали ще більш критичними.

Мета і задачі дослідження. Мета полягає в розробці мобільного застосунку "Інтерактивна карта для туриста", який би забезпечував користувачів необхідною інформацією для комфортного та безпечного перебування в нових місцях, а також сприяв взаємодії між туристами та місцевими сервісами. Задачі дослідження включають:

- Вивчення поточних мобільних застосунків, визначення їхніх переваг та недоліків, а також задоволення потреб користувачів, які не існуючими продуктами.

- Визначення вимог до функціональних можливостей застосунку на основі проведеного аналізу, включаючи необхідні інтерактивні елементи, інформаційні модулі та сервіси.

- Розробка архітектурної моделі мобільного додатку, визначення основні компонентів, їх взаємодію та способи інтеграції з зовнішніми сервісами.

- Реалізація інтуїтивно зрозумілого та зручного інтерфейсу, який би дозволяв користувачам легко взаємодіяти з додатком та отримувати необхідну інформацію.

- Розробка функцій взаємодії користувачів, таких як відгуки, рекомендації, спільне планування маршрутів та інші соціальні функції.

- Тестування й оптимізація застосунку для виявлення та усунення можливих помилок, забезпечення швидкодії та стабільності.

- Розробка стратегії запуску та просування додатку на ринку.

Таким чином, реалізацією даного проєкту буде продукт, який задовольнить потреби сучасних туристів, підвищуючи рівень комфорту під час подорожей.

РОЗДІЛ 1. АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ ТА ПОСТАНОВКА ЗАВДАННЯ

1.1 Аналіз ринку цифрових карт

Цифрова карта відноситься до процесу створення та відображення географічної інформації у цифровому форматі. Це включає в себе використання комп'ютерної технології для збору, зберігання, обробки, аналізу та подання просторових даних. Цифрові карти можуть представляти різні об'єкти та явища, такі як рельєф, дороги, будівлі, річки, кордони та об'єкти цікавості. Ці карти зазвичай створюються за допомогою програмного забезпечення географічних інформаційних систем (ГІС), що дозволяє інтегрувати та аналізувати кілька шарів просторових даних.

Однією з найбільш значних переваг цифрового картографування є його здатність надавати точну та актуальну інформацію про певне місце. Цифрові карти можна постійно оновлювати новими даними, що дозволяє користувачам бачити зміни в реальному часі. Це робить цифрове картографування особливо корисним у надзвичайних ситуаціях, де своєчасна та точна інформація може бути вирішальною для заходів реагування та відновлення. Крім того, цифрове картографування також може допомогти покращити планування та прийняття рішень у різних галузях промисловості. Наприклад, у міському плануванні цифрові карти можуть використовуватися для визначення областей з високою щільністю населення або заторами на дорогах, що допомагає міським чиновникам краще планувати транспортні маршрути та громадські служби. У сільському господарстві цифрові карти можуть використовуватися для аналізу типів ґрунтів та рівнів поживних речовин, допомагаючи фермерам оптимізувати врожаї та зменшувати втрати. Крім того, цифрове картографування також може сприяти зусиллям у збереженні навколишнього середовища. Шляхом картографування чутливих екосистем та місць існування,

дослідники та зберігачі можуть визначити області, які потребують захисту, та розробити стратегії для мінімізації впливу людини.

Фактори, такі як зростання у прийнятті мобільних обчислювальних пристроїв для навігації, широке використання 3D платформ та передові технології для обстеження та створення цифрових карт, очікуються прискорити зростання ринку цифрових карт у майбутньому. Крім того, зростаючий попит на послуги на основі місцезнаходження у різних секторах, таких як транспорт, логістика та роздрібна торгівля, використовуючи цифрові карти для покращення своєї діяльності, поліпшення вражень клієнтів та оптимізації розподілу ресурсів. Ці тренди на ринку цифрових карт обумовлені зростанням поширення смартфонів, пристроїв з GPS та потребою у реальному часі в навігації та геопросторових даних. Однак наявність безкоштовних карт з колективним створенням та правові виклики обмежують зростання. Крім того, стрімке збільшення прийняття карт у реальному часі створює сильну можливість для галузі цифрових карт. Крім того, інтеграція передових технологій у цифрове картографування, наприклад машинного навчання і штучного інтелекту, також очікується, що забезпечить прибуткові можливості для зростання ринку протягом прогнозного періоду. Крім того, зростання пристроїв та датчиків Інтернету речей (IoT) генерує величезні обсяги місцезнайдених даних, що додатково підкріплює попит на передові технології картографування [1].

Ринок цифрових карт - це динамічна сфера, де інновації та технології зливаються, щоб визначити спосіб, яким ми навігуємо та розуміємо наш світ. З появою пристроїв з GPS та розвитком програмного забезпечення для картографування цифрові карти стали невід'ємними інструментами для осіб, бізнесів та урядів.

Ринок цифрових карт передбачається зростати з CAGR 14,65% з 2023 по 2030 рік, з приблизною вартістю USD 18,6 мільярда у 2022 році до USD 55,52 мільярда у 2030 році [2]. Дані тенденції зображено на рисунку 1.1

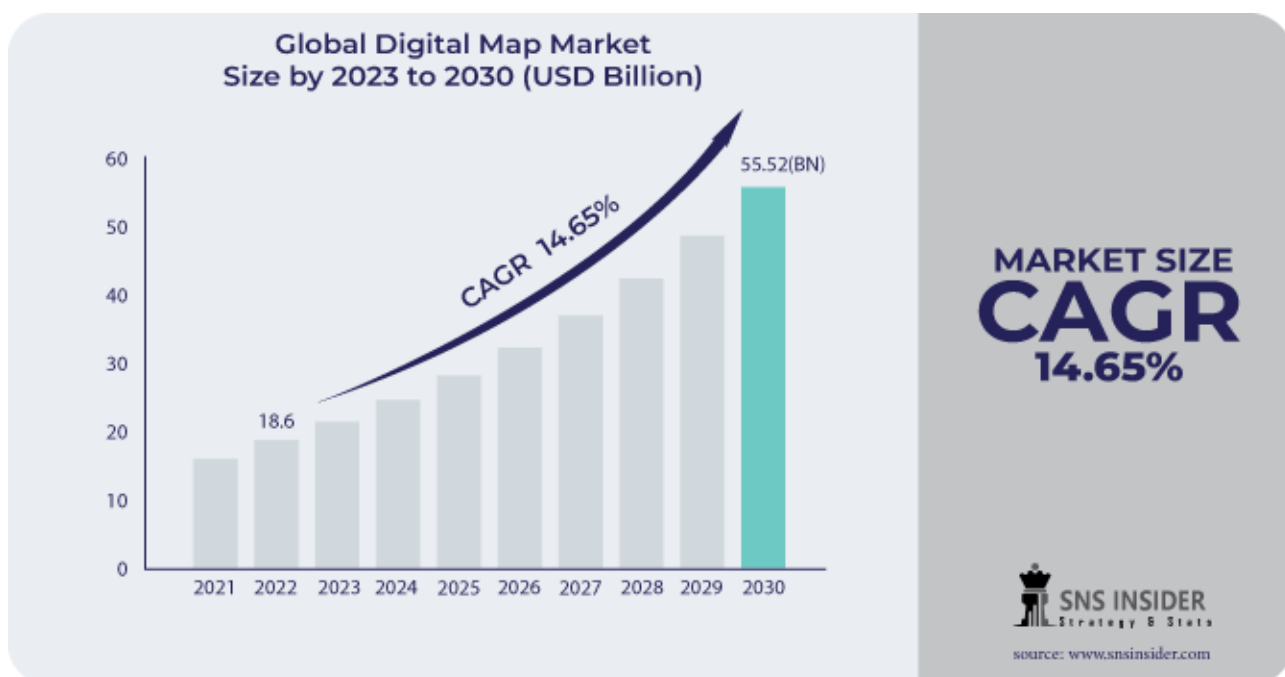


Рисунок 1.1 - Тенденції розвитку цифрових карт

За застосуванням ринок поділено на автомобільну промисловість, військово-оборонний комплекс, мобільні пристрої, підприємницькі рішення, логістику, подорожі та транспорт, розвиток інфраструктури та будівництво та інші. Їх розподіл показано на рисунку 1.2.

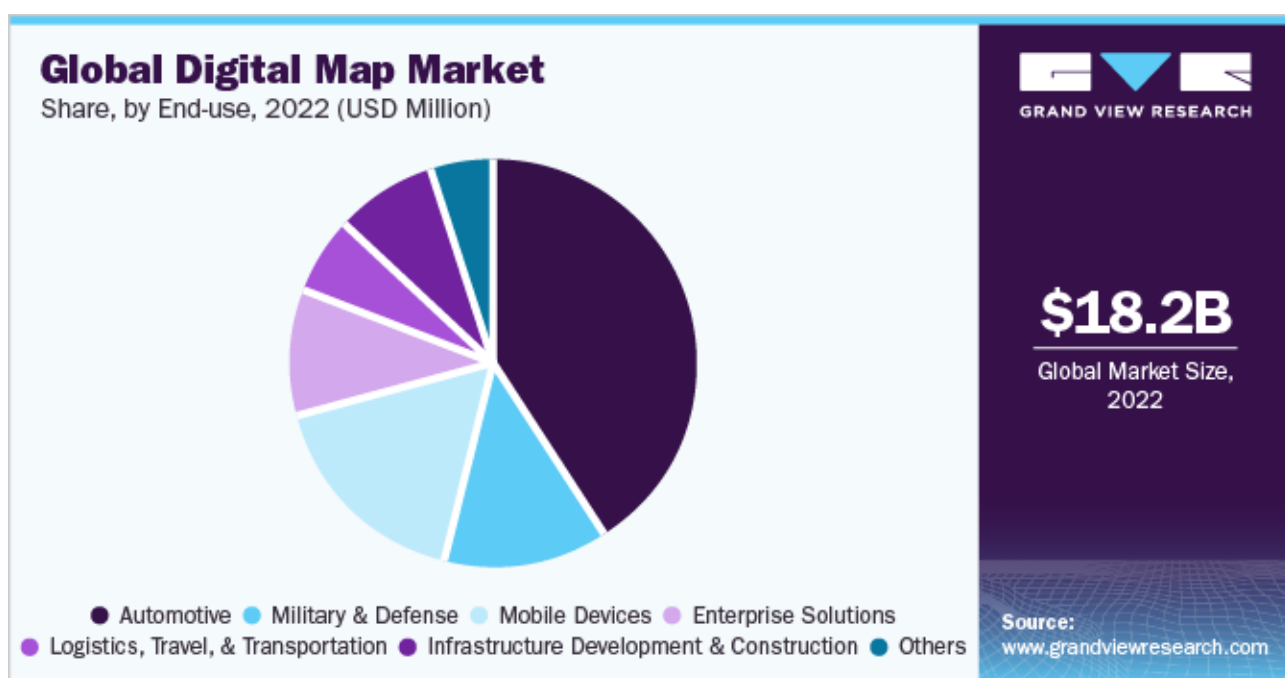


Рисунок 1.2 - Розподіл ринку цифрових карт

Сегмент автомобільної промисловості домінував на ринку, зареєструвавши найвищу частку ринку приблизно 40,0% у 2022 році. Це пов'язано з швидким зростанням попиту на автомобільну навігацію та досягненнями в галузі підключених автомобілів. Це також може бути пов'язано з відновленим інтересом автомобільних компаній до впровадження автономних автомобілів як майбутнього способу переміщення. Інші ключові напрямки в застосуванні цифрових карт для автомобільної промисловості включають управління автопарком, системи передової допомоги водієві (ADAS) та системи контролю логістики.

Сегмент мобільних пристроїв також передбачається стати найшвидше зростаючим сегментом з річним темпом зростання на рівні 17,9% протягом прогнозованого періоду. Зростаюча інтеграція різних технологій дистанційного зондування та навігації на спільну рухому платформу сформувала великий попит на цифрові карти. Крім того, зростаюча потреба серед користувачів у більш ефективній та беззусильній обробці інформації про маршрутизацію в реальному часі сформувала велику базу для ринку цифрових карт у сегменті мобільних пристроїв[3].

Одним із ключових факторів, що сприяють зростанню ринку, є зростаючий попит на послуги на основі місцеположення (LBS). Від додатків для замовлення таксі до логістики доставки, LBS спираються на точні та актуальні цифрові карти для надання користувачам інформації в реальному часі та персоналізованих досвідів. Цей попит очікується подальше зростання, оскільки галузі, такі як роздрібна торгівля, туризм та міське планування, використовують потужність місцевого інтелекту для оптимізації операцій та підвищення залучення клієнтів.

Основними факторами попиту на цифрові карти є сегменти кінцевого використання автомобілів та мобільних пристроїв, які значно впливають на регіональне прийняття цифрових карт. На рисунку 1.3 показано регіональні тренди росту.

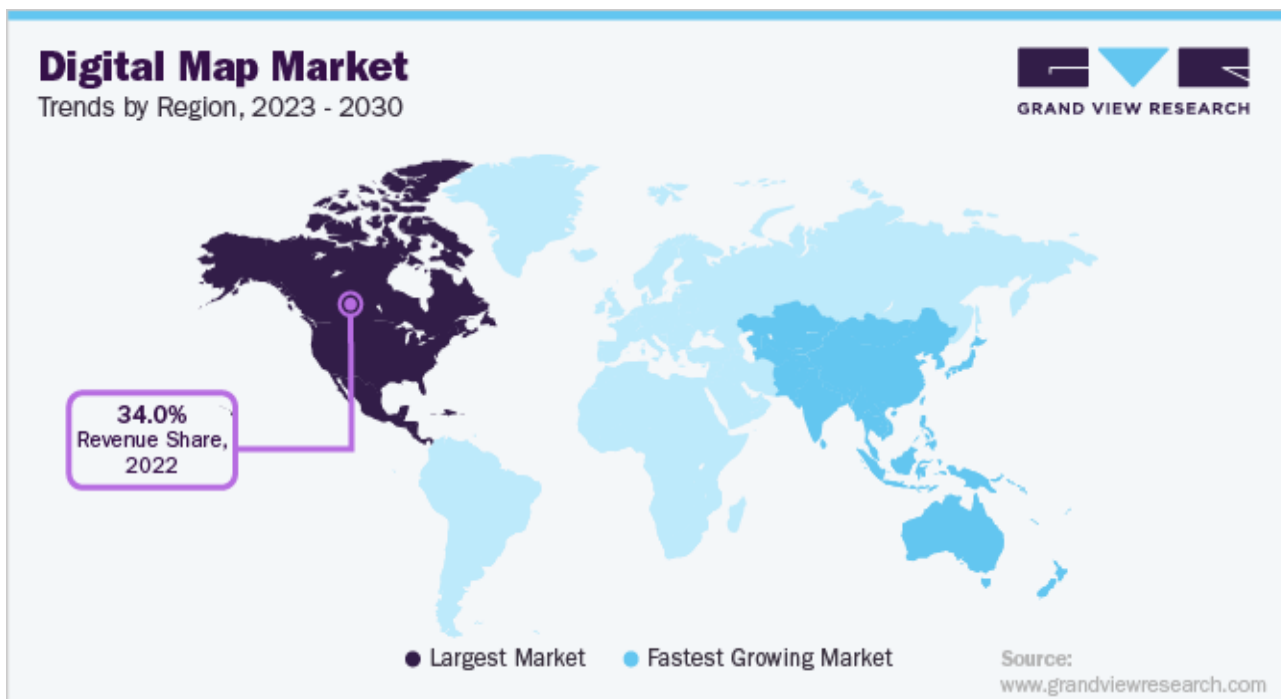


Рисунок 1.3 – Регіональні тренди росту ринку

В результаті Північна Америка очолила глобальний ринок цифрових карт за обсягом виручки. Регіон зареєстрував найвищу частку ринку приблизно 34,0% у 2022 році. Це можна пояснити великою кількістю гравців у галузі, розвиненим автомобільним сектором та недавнім зростанням у злиттях та поглибленнях.

Азійсько-тихоокеанський регіон передбачається стати найшвидше зростаючим регіоном, який спостерігатиме річний темп зростання на рівні 17,0% протягом прогнозованого періоду через швидкий попит на системи ГІС, геопросторові дані, а також пов'язані послуги в регіоні. Зростаюче використання цифрових карт урядами для розвитку сільських та сільськогосподарських районів, проектів розумних міст, а також розвитку міської та транспортної інфраструктури, серед інших, виступає як основні регіональні драйвери зростання. Крім того, розвиток інфраструктури в країнах, таких як Китай та Індія, відкрив нові бізнес-можливості для постачальників, що сприяє росту ринку [3].

1.2 Аналіз існуючих рішень

Важливим елементом даного ринку являються Google Maps. Даний сервіс надає вичерпні дані про географічні місця та об'єкти у всьому світі. Окрім самих простих карт, Google Maps пропонує аеро- та супутникові зображеннялюбих місць. В багатьох містах Google Maps також пропонує вуличні відображення, які включають фотографії, виконані з автомобілів [4]. В його основі знаходиться широкий арсенал картографічних даних, отриманих від мільйонів джерел. Цей багатогранний набір даних постійно оновлюється та вдосконалюється. Дані карти є одними з найпопулярніших додатків для навігації та отримання інформації про різні об'єкти, їх вигляд показано на рисунку 1.4.

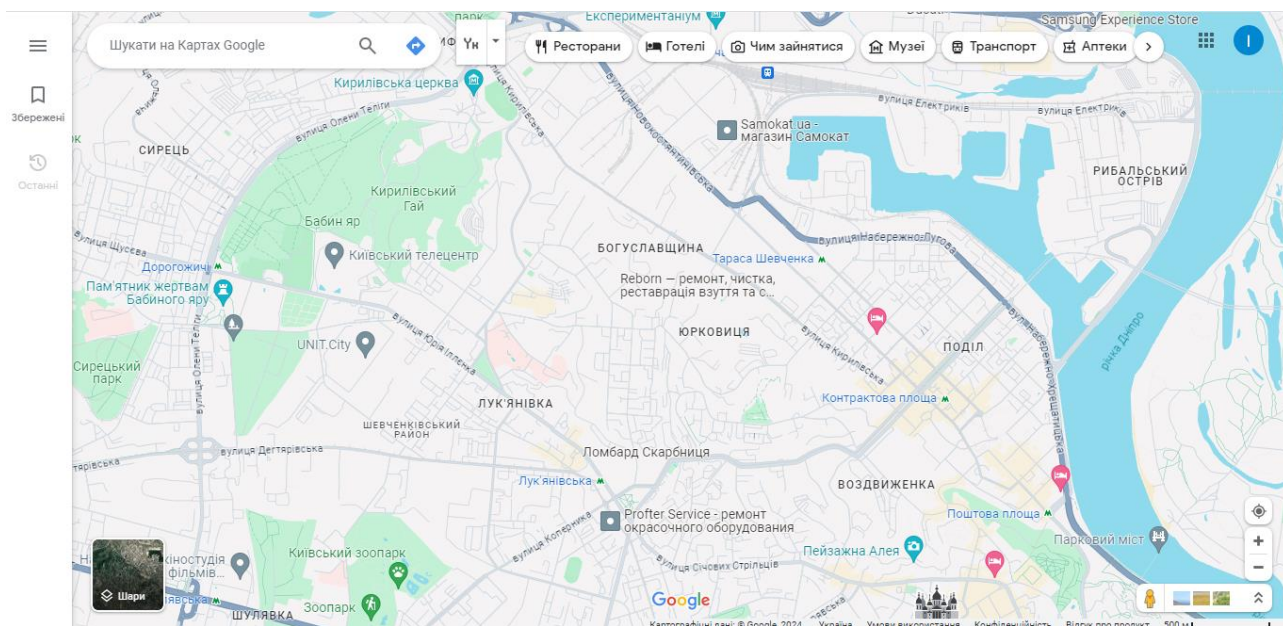


Рисунок 1.4 - Інтерфейс Google Maps

Однією з головних переваг Google Maps є його широкий охоплення географічних областей. Система карт включає дані практично з усього світу, що робить її незамінною для мандрівників у будь-якому куточку планети. Додаток також постійно оновлюється і покращується, додавши нові функції, такі як реальний час у деяких областях, підтримка віртуальної реальності, інтеграція з

громадським транспортом тощо. Однією з ключових функцій є навігація в реальному часі, яка надає користувачам точні та актуальні відомості про маршрути, дорожні умови та пробки. Ця функція допомагає мінімізувати час та зусилля, потрібні для подорожі, і забезпечує більш ефективне використання часу. Додаток також надає користувачам можливість додавати свої власні фотографії та відгуки про відвідані місця, що дозволяє створювати більш особистий та унікальний досвід використання додатку. Ще однією важливою перевагою є його дружній інтерфейс користувача, він простий у використанні, що робить його доступним для широкого кола користувачів, від досвідчених подорожників до тих, хто тільки починає вивчати мобільні додатки.

Однак Google Maps має й деякі недоліки. Наприклад, для використання деяких функцій, таких як офлайн-карти або деякі додаткові шари інформації, потрібне підключення до Інтернету. Це може бути незручно в областях з поганим покриттям мобільного зв'язку або в умовах обмеженого доступу до Інтернету. Крім того, він може бути досить ресурсоемким для деяких пристроїв, особливо для старіших моделей смартфонів або тих, які мають обмежені технічні можливості. Це може призвести до повільної роботи програми або виснаження заряду батареї пристрою.

Незважаючи на ці недоліки, Google Maps залишається одним з найбільш функціональних та корисних інструментів для навігації та орієнтування, доступних на ринку мобільних додатків.

Другою програмою яку слід розглянути є TripAdvisor. Це онлайн-платформа, яка спеціалізується на наданні відгуків, рекомендацій та інформації про подорожі для мандрівників у всьому світі. Даний веб-сайт та мобільний додаток дозволяють користувачам знаходити та оцінювати різноманітні готелі, ресторани, визначні місця та інші туристичні об'єкти, а також ділитися своїми власними враженнями та досвідом з іншими користувачами.

Однією з основних переваг TripAdvisor є велика база даних відгуків та рекомендацій від реальних користувачів. Це дозволяє мандрівникам отримати об'єктивну інформацію про якість послуг та умови перебування у різних

зкладах, що допомагає приймати обґрунтовані рішення щодо вибору готелю, ресторану чи екскурсії. Крім того, TripAdvisor надає можливість знаходити цікаві та непересічні місця для відвідування, які можуть бути прихованими скарбами або маловідомими туристичними атракціями. Велика кількість відгуків та порад від інших мандрівників дозволяє відкривати нові місця та досліджувати незвідані куточки світу.

Потрібно завжди пам'ятати інформація, надана користувачами, може бути суб'єктивною та не завжди об'єктивною. Відгуки можуть бути спотвореними або підробленими, що може призвести до неточностей та невірної інформації. Крім того, TripAdvisor не завжди має повну або актуальну інформацію про всі заклади та місця. Деякі готелі чи ресторани можуть бути відсутніми в базі даних, або інформація про них може бути застарілою, що ускладнює процес планування подорожі.

TripAdvisor вважається корисним інструментом для мандрівників, який дозволяє отримувати інформацію та рекомендації від реальних користувачів і відкривати нові туристичні можливості.

Третім застосунком який буде розглядатись буде Maps.me. MAPS.ME – симбіоз офлайн-карт, онлайн-довідника і GPS-навігатора для Android-пристроїв. Може бути використаний як для пошуку об'єктів, побудови маршрутів або позиціонування в режимі реального часу, так і для навігації в умовах відсутності зв'язку. Вміє запам'ятовувати прокладені маршрути, зберігати їх в закладки і розшарювати обраним контактам з допомогою гіперпосилань [5].

Він є безкоштовним мобільним додатком для навігації та картографії, який дозволяє користувачам завантажувати та переглядати карти для використання в офлайн-режимі. Основна перевага полягає у можливості використовувати його без доступу до Інтернету, що особливо корисно для подорожей у віддалені або з поганим покриттям мобільного зв'язку областях. Його вигляд показано на рисунку 1.5.



Рисунок 1.5 - Інтерфейс застосунка Maps.me

Крім того, Maps.me має дуже широке охоплення географічних областей і включає в себе детальні карти практично з усього світу. Це робить його відмінним інструментом для подорожей до різних країн та регіонів, де інші картографічні додатки можуть бути менш ефективними.

MAPS.ME працює по-іншому, ніж більшість популярних навігаційних додатків. Замість використання Інтернет-з'єднання для отримання та показу карт в реальному часі, додаток працює повністю офлайн. Це привертає іншу групу користувачів - часто це подорожуючі або мешканці областей з поганим Інтернет-з'єднанням. Це надає додатку важливе конкурентне перевагу порівняно з іншими картографічними сервісами. Ця позиція важлива, оскільки додаток використовується на мобільних платформах, які належать відповідно Google і Apple.

Очевидно, що в такій програмі будуть обмеження. Наприклад, оновлення карт можуть бути менш частими порівняно з іншими сервісами, це може призвести до застарілої або неточної інформації про деякі місця. Крім того, функціонал додатку обмежений порівняно з більш розширеними картографічними сервісами, і він може не має деяких додаткових функцій, які доступні в інших програмах.

Останнім можна переглянути застосунок Citymapper, що спеціалізується на навігації великими містами та використанні громадського транспорту. Його вигляд можна переглянути на рисунку 1.6.

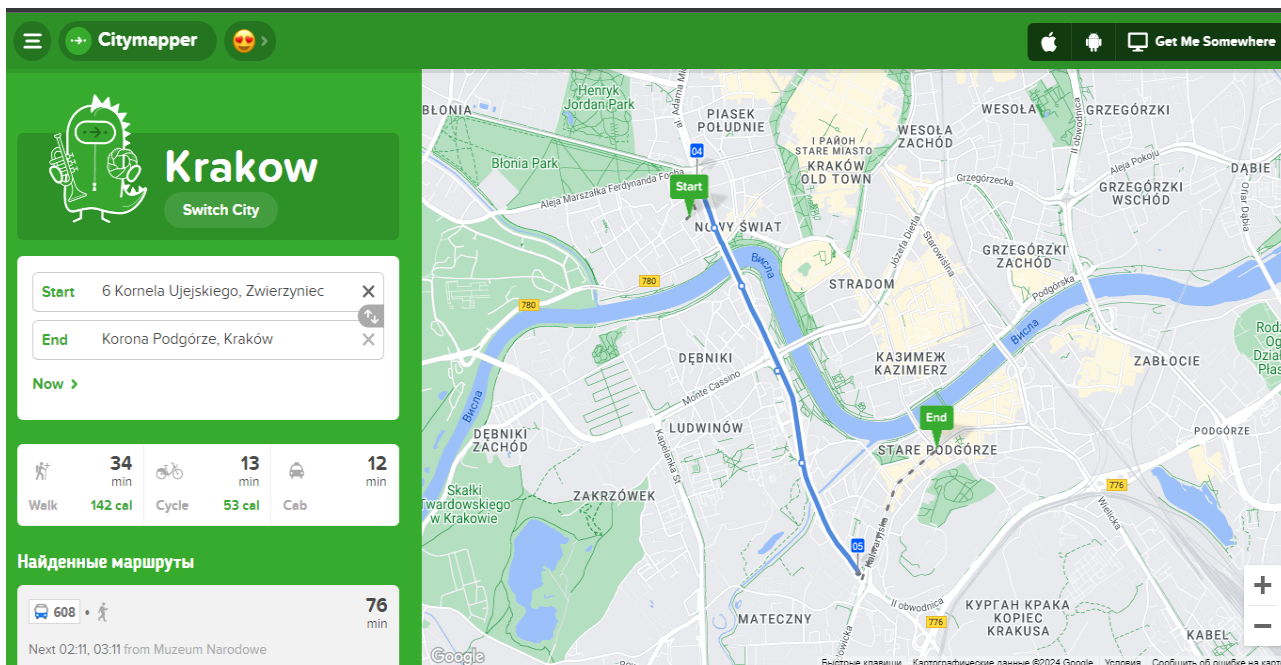


Рисунок 1.6 - Інтерфейс Citymapper

Він надає користувачам інформацію про розклади руху автобусів, тролейбусів, трамваїв, метро, потягів, велопрокат і навіть таксі у більшості великих міст світу. Основна перевага Citymapper полягає в його спроможності інтегрувати різні види транспорту та надавати користувачам оптимальні маршрути для переміщення з одного пункту до іншого у місті. Крім того, додаток включає інформацію про вартість проїзду, тривалість поїздки, а також можливість зберігати улюблені маршрути для швидкого доступу. Citymapper також надає користувачам актуальну інформацію про транспортні затримки, ремонти, й інші події, що можуть впливати на мобільність у місті. Однак, серед недоліків Citymapper можна відзначити обмежену доступність - додаток підтримує лише обрані великі міста, і для деяких регіонів може бути менш актуальним або навіть недоступним. У деяких містах функціонал може бути обмеженим, залежно від наявності та якості даних про громадський транспорт.

Ринок подібних сервісів для навігації та картографії супроводжується значною конкуренцією, проте це сприяє розвитку інновацій та покращенню сервісів для користувачів. Кожен застосунок має свої унікальні особливості та переваги, і дана різноманітність дозволяє користувачам вибирати сервіс, який найбільш відповідає їх потребам та умовам використання. При цьому важливо враховувати якість даних, актуальність інформації та зручність використання, щоб забезпечити найкращий досвід користувача. Такий ринок стимулює розвиток нових технологій та підвищення якості сервісів, що призводить до постійного удосконалення навігаційних рішень для користувачів.

1.3 Визначення потреб користувачів

Потреби користувачів є кутовими каменями усіх успішних інновацій у сфері користувацького досвіду. Якщо ваш продукт, рішення або функціонал не відповідає існуючим або потаємним потребам цільових користувачів, то всі ваші зусилля з метою створення відмінного користувацького досвіду для цього рішення можуть бути марні. Іншими словами, ваша інновація в користувацькому досвіді ніколи не буде успішною, якщо ви не задовольняєте існуючих або потаємних потреб кінцевих користувачів [6].

Визначення потреб користувачів у контексті розробки мобільного застосунку - це процес аналізу та ідентифікації ключових вимог та очікувань, які користувачі можуть мати від цього застосунку. Детальне вивчення потреб користувачів допомагає зрозуміти, як саме вони планують використовувати застосунок та які функції та можливості є для них найважливішими.

Перш за все, важливо врахувати потребу у зручному та швидкому способі знаходження маршрутів та орієнтування в невідомих місцях. Це може включати в себе наявність детальних картографічних даних, оновлених інформаціях про громадський транспорт, а також можливість використовувати застосунок в офлайн-режимі. Само собою додаток повинен мати зручний та зрозумілий

інтерфейс з різноманітними функціями таких як отримання рекомендацій та відгуків від інших людей, пошук різних місць, тощо.

Також важливо враховувати залежності від індивідуальних вимог та ситуацій. Наприклад, дехто може шукати оптимальні маршрути для пішоходів, велосипедистів або водіїв, тоді як інші можуть цікавитися більше інформацією про громадський транспорт або можливість використання додаткових послуг, таких як пошук таксі чи прокат велосипедів. В загальному можна виділити кілька важливих вимог:

- Потреба у зручному пошуку місць.
- Потреба у навігації.
- Потреба у збереженні місць та маршрутів.
- Потреба у офлайн-режимі.
- Потреба у персоналізації.

Завданням даного проєкта є розробка мобільного застосунку, який надає зручний та ефективний інструмент для навігації та відкриття нових місць у містах та регіонах. Крім того, він повинен мати можливість планування оптимальних маршрутів від одного місця до іншого, а також надавати можливість збереження улюблених місць та маршрутів для подальшого використання або поділу з іншими користувачами. Важливо також забезпечити можливість використання застосунку в офлайн-режимі, щоб користувачі могли користуватися ним у місцях з обмеженим доступом до Інтернету.

1.4 Висновки до першого розділу

В першому розділі було приділено увагу детальному аналізу предметної області для розробки мобільного додатку. Початковим етапом був проведений огляд ринку цифрових карт, що виявив високий попит на ці технології у сучасному туристичному секторі. Зокрема, виявлено зростаючу популярність мобільних додатків, які надають користувачам можливість швидко та зручно орієнтуватися в новому місцевості. Далі був проведений детальний аналіз

існуючих рішень на ринку. Виявлено, що багато з наявних додатків мають обмежені функціональні можливості. Більшість з них, наприклад, концентруються на навігації без додаткових інтерактивних можливостей, які можуть покращити загальний досвід використання.

Наступним кроком було визначення потреб потенційних користувачів. Були ідентифіковані ключові аспекти, які важливі для туристів під час подорожей: зручність використання, наявність додаткової інформації про визначені місця, можливість персоналізації контенту тощо.

Отже, результати цієї роботи послужать основою для подальшої розробки концепції інтерактивної карти, що відповідає сучасним вимогам та потребам туристів.

РОЗДІЛ 2. ПРОЕКТНА ЧАСТИНА

2.1 Огляд середовища розробки

Додаток розробляється для платформи Android з використанням Android Studio. Це дуже популярне середовище для розробки, створене компанією Google, яке дозволяє ефективно створювати та налагоджувати мобільні додатки. Однією з головних переваг є його офіційна підтримка з боку Google, що забезпечує високий рівень сумісності з оновленнями платформи Android та іншими сервісами компанії. Вбудована підтримка SDK дозволяє розробникам легко налаштовувати проекти. Android Software Development Kit (SDK) вміщає в себе багато утиліт, які знадобляться для побудови застосунку, а також обширний інструментарій. Він дає можливість інсталиювати Android API, і перевірити сховище на присутність архівів та пакетів, що не були встановленні. [7].

Інтерфейс середовища розроблений з метою забезпечити розробникам зручний та ефективний робочий процес. Його зображено на рисунку 2.1.

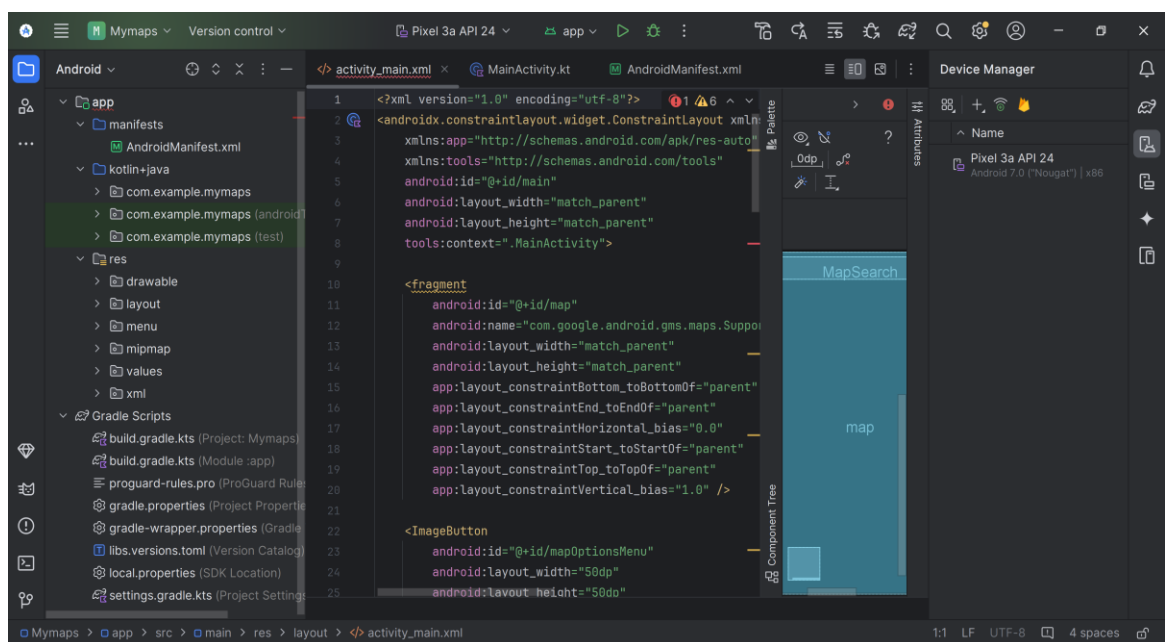


Рисунок 2.1 – Інтерфейс Android Studio

Після запуску програми користувач опиняється в головному робочому просторі, де він може створювати нові проекти, відкривати вже існуючі, налаштовувати робочі параметри та виконувати інші дії.

У верхній частині вікна розташоване меню з різноманітними командами для керування проектами, відлагодженням, налаштуваннями та іншими функціями програми. Під меню зазвичай розташовується панель інструментів, де розміщені кнопки швидкого доступу до основних операцій, наприклад, запуску додатку, відлагодження коду, вибору конфігурацій запуску, інструментів тестування та іншого.

Центральна частина інтерфейсу зазвичай відведена для відображення вмісту проекту. Тут може розміщуватися вікно редактора коду, візуальний редактор макетів або інші робочі області, в залежності від того, яку частину проекту ви редагуєте або переглядаєте.

У нижній частині вікна можна знайти панель стану. Тут відображається стан проекту та самого ІСР, а також будь-які попередження або повідомлення [8]. Також є додаткові інструменти, наприклад, швидкий доступ до логів виконання, аналізатора профілювання та інших засобів.

Додаткові вікна, такі як вікно відлагодження, вікно керування проектом, вікно терміналу тощо, можуть відкриватися за потребою користувача та розміщатися в різних частинах інтерфейсу, наприклад, у вигляді вкладок або панелей.

Загалом, інтерфейс спроектований з урахуванням зручності та продуктивності розробників, надаючи їм необхідні інструменти та можливості для створення якісних додатків для платформи.

Для ефективного створення власного проекту відведений різноманітний інструментарій. Основні компоненти цього інструментарію включають в себе вбудовані редактори, аналізатори коду, інструменти тестування та відлагодження, а також інші корисні засоби.

Редактор макетів дозволяє розробникам створювати та редагувати інтерфейс користувача за допомогою візуального інтерфейсу, що спрощує

процес розробки UI. Він також надає можливість перегляду вигляду додатку на різних пристроях та в різних конфігураціях екрану.

Редактор коду забезпечує зручне середовище для написання та редагування коду додатка. Він має широкий набір функцій, таких як підсвічування синтаксису, автодоповнення коду, швидкий доступ до документації та інші.

Аналізатор коду допомагає виявляти та виправляти помилки в коді, а також проводити аналіз стилю коду та його оптимізацію для підвищення якості та продуктивності програми.

Інструменти тестування включають в себе різноманітні засоби для проведення автоматичних та ручних тестів, включаючи тести одиниць, інтеграційні тести та інші. Ці інструменти допомагають забезпечити надійність та стабільність додатку перед його випуском.

Відлагоджувач надає можливість виявлення та виправлення помилок в програмі шляхом відстеження виконання коду та перевірки значень змінних.

Профілер дозволяє аналізувати роботу додатку в реальному часі, вимірюючи його використання ресурсів, таких як пам'ять та процесор, що дозволяє виявляти та усувати проблеми з продуктивністю.

Загалом, Android Studio є потужним середовищем розробки мобільних застосунків, що користується найбільшою популярністю у всьому світі і регулярно оновлюється.

Розробка проєкту велась мовою програмування kotlin. Вона була створена компанією JetBrains. Kotlin створювався для виправлення певних проблем Java. В даній мові фактично, всі змінні за замовчуванням є ненульовими, і ризики виникнення NullPointerException значно зменшуються ще на етапі компіляції. Згідно слів прихильників мови, синтаксис простіший та призводить до значно меншого роздування коду. Це допомагає краще зосередитися на розв'язанні потрібної задачі, а не на ускладненому синтаксисі [9].

Серед всіх мов програмування kotlin ніколи не знаходився високо в топі популярності що показано на рисунку 2.2. Проте для створення програм орієнтованих на мобільні пристрої, це дуже хороший вибір.

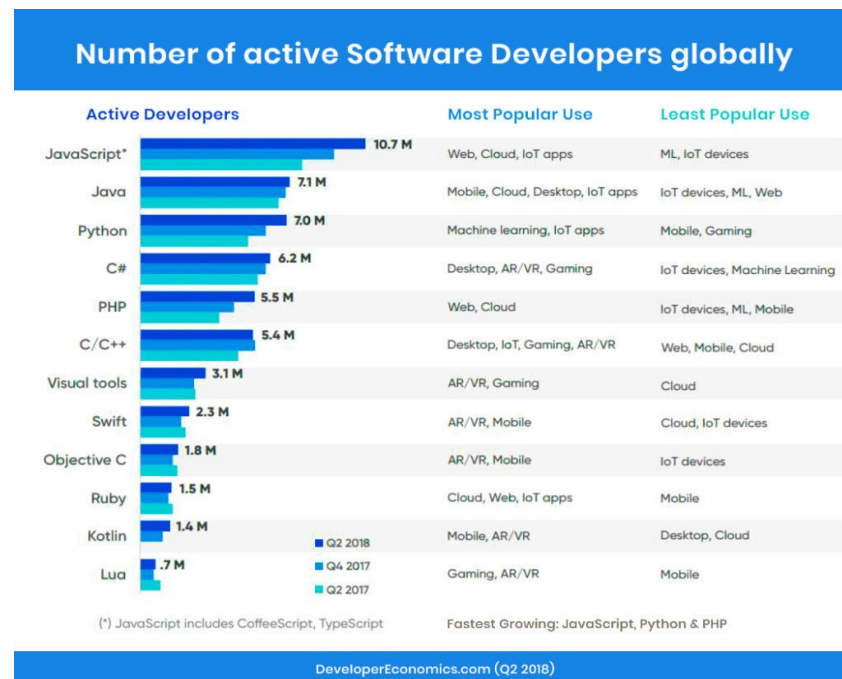


Рисунок 2.2 - Рейтинг популярності мов програмування

Понад 50% професійних розробників Android використовують Kotlin як основну мову програмування, тоді як лише 30% працюють з Java. 70% розробників, для яких основна мова - Kotlin, стверджують, що він збільшує їх продуктивність [10].

Набула широкої популярності завдяки своїй простоті, читабельності та потужності. Однією з ключових особливостей Kotlin є його безпека. Він вбудовує в себе низку механізмів, які допомагають уникнути типових помилок програмістів, таких як помилки нульових вказівників (NullPointerException). Також використовує систему нульової безпеки, яка унеможливорює використання нульових значень без відповідного перевірки.

Додатковою перевагою Kotlin є його експресивний та лаконічний синтаксис. Код часто коротший та зрозуміліший, ніж аналогічний на Java,

завдяки вбудованим функціональним можливостям та іншим вдосконаленням. Крім того, він підтримує різні структурні підходи до програмування, такі як функціональне програмування, що дозволяє писати більш зрозумілий та компактний код. Дана мова програмування також надає багато інструментів для роботи з асинхронним кодом. В нього вбудована підтримка coroutines, яка дозволяє легко створювати асинхронні операції без потреби в складних конструкціях зворотних викликів або інших механізмів. Код написаний на котліні може бути легко інтегрований в існуючі проекти, а також використовувати бібліотеки та інструменти.

Kotlin працює на різних платформах (Windows, Mac, Linux, Raspberry Pi та інші) і на 100% сумісний з Java. Використовується багатьма великими компаніями, такими як Google, Netflix, Slack, Uber і інші, для розробки своїх додатків на базі Android [11].

Загалом, Kotlin - це потужна та сучасна мова програмування, яка пропонує широкий спектр можливостей для розробки програмного забезпечення на будь-яких платформах та для різних завдань. Її простота використання, безпека та ефективність роблять її відмінним вибором для будь-якого проекту.

2.2 Формалізація вимог до системи та моделювання

Формалізація вимог до системи - це процес узгодження та документування функціональних та нефункціональних вимог до програмного забезпечення. Цей процес є важливим етапом у розробці будь-якої системи, оскільки він дозволяє зрозуміти потреби та очікування користувачів та інших зацікавлених сторін. Він включає в себе визначення функціональності системи, її можливостей та обмежень. Це може включати в себе опис різних сценаріїв використання, які показують, як користувачі будуть взаємодіяти з системою та очікувані результати взаємодії. Крім того, формалізація вимог може включати в себе опис нефункціональних вимог, таких як швидкодія системи, її надійність,

безпеку та інші аспекти, які не пов'язані безпосередньо з функціональністю, але є важливими для успішної реалізації системи.

Об'єднана мова моделювання (UML) є візуальною мовою програмування і корисним інструментом для дизайнерів і програмістів. Вона допомагає поєднувати візуалізацію з стандартизацією для досягнення вищої якості, кращої відповідності та підвищеної продуктивності. Для професіоналів в галузі інформаційних технологій, які розробляють, тестують або аналізують програмне забезпечення, UML може забезпечити ці переваги і підтримати їхню роботу [12]. Вона має власний синтаксис і правила для оформлення моделей [13].

Для забезпечення для візуального відображення структури та взаємозв'язків класів у системі використовуються діаграми класів. У цих діаграмах класи зображуються у вигляді прямокутників, кожен з трьох секціями для назви класу, атрибутів і методів. Лінії, що з'єднують класи, показують асоціації, що відображають відносини, такі як один-до-одного або один-до-багатьох [14].

Діаграма активностей є ще однією важливою поведінковою діаграмою в UML для опису динамічних аспектів системи. Вона суттєво розширює звичайну блок-схему, моделюючи потік від однієї активності до іншої. Також описує порядок виконання викликів методів і моменти їх завершення [15].

Діаграма розгортання UML зображує статичний вигляд конфігурації часу виконання апаратних вузлів та програмних компонентів, які працюють на цих вузлах. Діаграми розгортання UML показують апаратне забезпечення вашої системи, програмне забезпечення, що встановлене на цьому апаратному забезпеченні, і проміжне програмне забезпечення, яке використовується для з'єднання різних машин одна з одною [16].

Діаграма об'єктів UML відображає конкретний екземпляр діаграми класів в певний момент часу. Вона фокусується на атрибутах набору об'єктів і на їхніх взаємозв'язках. Діаграми об'єктів легко створювати: вони складаються з

об'єктів, зображених у прямокутниках, з'єднаних між собою лініями [17].
Діаграми об'єктів можуть бути корисні у таких ситуаціях:

1. Під час аналізу проекту можна створити діаграму класів для опису структури системи, а потім створити набір діаграм об'єктів як тестові випадки для перевірки повноти діаграми класів.

2. Перед створенням діаграми класів можна створити діаграму об'єктів для виявлення фактів про конкретні модельні елементи та їх зв'язки [18].

Діаграма послідовності представляє зразковий уривок процесу програмної системи. Вона моделює взаємодії і активності, і може бути розширена виразами OCL. Також описує порядок виконання викликів методів і час їх завершення. Отже, подібно до діаграм станів, діаграми послідовності моделюють поведінкові аспекти [19].

Графічним зображенням функціональних вимог до системи, яке відображає взаємодію між користувачами та системою є діаграма варіантів використання (Use Case Diagram). Діаграма варіантів використання демонструє набір ВВ і діючих осіб-акторів (актантів) (які є спеціальним видом класів), а також їх зв'язки. Вони описують статичний вигляд ВВ системи і важливі для організації та моделювання поведінки системи [20].

Вона допомагає визначити, як користувачі будуть використовувати систему та які функції вони будуть виконувати. Діаграма варіантів використання дозволяє зосередитися на тому, що робить система для користувачів, а не на технічних деталях її реалізації.

Графічне зображення діаграми варіантів використання може бути представлене у вигляді еліпсів для акторів та прямокутників для варіантів використання. Стрілки з акторів до варіантів використання показують, які дії може виконати кожен актор.

Дане представлення допомагає розуміти, як користувачі будуть взаємодіяти з системою, та служить основою для подальшого проектування та

реалізації системи. Вона є важливим інструментом для забезпечення відповідності функціональних вимог до системи потребам користувачів.

2.2.1 Вимоги до програмного застосунку

У кожного програмного забезпечення є свої особливості та вимоги. Серед них деякі є практично обов'язковими:

- Надійність. Визначає шанс стабільної роботи ПЗ протягом певного проміжку часу [21].
- Гнучкість. Проект повинен бути створений таким чином, щоб можна було додавати нові можливості без суттєвого внесення змін до існуючого модуля. Він завжди має бути готовим до нових вимог.
- Ефективність. Вона означає що продукт виконує свої функції без зайвих витрат часу та ресурсів. Ресурсами можуть бути зовнішня пам'ять, оперативна пам'ять, пропускна здатність каналу та інші.
- Масштабованість. система повинна зростати без негативного впливу на її продуктивність. Це означає обслуговування більше користувачів, обробку більше даних і виконання більше транзакцій [21].
- Портативність. Програма повинна підтримуватися багатьма різними версіями Android, компілюватися та працювати безперебійно. Через швидкий розвиток апаратного та програмного забезпечення в наші дні зміна платформи є звичайним явищем.
- Продуктивність. Додаток повинен працювати швидко та без затримок, забезпечуючи миттєвий відгук на дії користувача. Завантаження карт, пошук інформації та інші операції повинні виконуватися без затримок.
- Безпека. Додаток повинен забезпечувати високий рівень безпеки, охоплюючи захист особистих даних користувачів, безпечну автентифікацію та авторизацію, і також захист від хакерських атак, таких як SQL-ін'єкції та XSS.

Дані вимоги є основними для проєктування структури системи, і виначаються на початковому етапі.

У випадку вимог з безпеки працездатності системи, в ній виділяються різні категорії користувачів, які мають право доступу до тих або інших функцій або даних, та передбачаються додаткові функції системи з перевірки доступу. Коли важливо обмежити доступ до певних даних, то в програмі може передбачатися, мандатний захист. Для створення кращого захисту системи від недобросовісного втручання, користувачі проходять реєстрація та аутентифікацію, щоб мати доступ до ПЗ.

2.2.2 Вимоги до інтерфейсу

Інтерфейс є ключовим аспектом, який визначає взаємодію між користувачем і програмою. Він об'єднує в собі елементи, які бачить користувач, та з якими він може взаємодіяти, коли користується програмою. Головними складовими частинами в інтерфейсі користувача програмного забезпечення є:

- сукупність керуючих елементів та їх розташування;
- графічне середовище;
- засоби взаємодії системи з користувачем;
- Основні вимоги до користувацького інтерфейсу.

Елементи керування інтерфейсом користувача (UI) є будівельними блоками будь-якого програмного інтерфейсу. Це можуть бути кнопки, вікна, списки, тощо. Вони дозволяють користувачу здійснювати різні дії з програмною системою. Вимоги до інтерфейсу можуть бути такими:

- Інтерфейс повинен бути легким у використанні, інтуїтивно зрозумілим для кінцевого користувача.
- Всі елементи повинні мати однаковий стиль, вигляд і поведінку, щоб уникнути замішання та покращити користувацький досвід.
- Інтерфейс повинен бути ефективним у виконанні завдань користувача.

- Хоча функціональність є ключовою, привабливий зовнішній вигляд також важливий. Все повинно бути естетично приємним і привабливим для користувача.
- Інтерфейс повинен бути адаптивним до різних розмірів екранів та пристроїв. Він повинен добре виглядати як на маленьких мобільних екранах, так і на великих моніторах.
- Користувач повинен мати можливість отримувати повідомлення про результати своїх дій і про можливі помилки. Це може включати відповідні повідомлення про помилки, підказки або підтвердження.
- Контрольованість є важливою, оскільки потрібно знайти золоту середину між автоматизацією роботи і повним контролем користувача. Вимога контрольованості стосується можливості впливу користувача на систему [22].

Архітектура мобільного застосунку має проєктуватися таким чином, щоб забезпечити масштабованість, ефективність та гнучкість. Основні компоненти архітектури включають клієнтську частину, серверну частину, базу даних та API. Інтерфейс же повинен бути інтуїтивно зрозумілим, зручним та привабливим. Він включатиме головний екран з картою, меню для навігації, а також додаткові опції.

2.2.3 Вимоги до Баз Даних

База даних - це організована структура даних, що зберігається та управляється з метою ефективного доступу, оновлення та аналізу. Проєктування БД являє собою структурування різних даних відповідно до певної моделі. Конструктор буде визначати, що за дані необхідно зберігати і як елементи даних взаємопов'язані.

При проєктуванні потрібно розуміти які вимоги повинна задовільняти БД. Вимоги до бази даних відносяться до конкретних потреб і очікувань, які зацікавлені сторони мають до бази даних. Вони служать основою для проєктування та впровадження ефективної системи бази даних. Документуючи

їх, розробник формулює мету, функціональність та обмеження бази даних, забезпечуючи спільне розуміння всіх учасників проекту [23]. Вимоги до БД можуть бути такими:

- цілісність даних;
- легкість оновлення;
- сумісність;
- мінімалізація надлишковості даних;
- часте використання даних;
- захищеність від несанкціонованого доступу до даних;
- швидкі пошук і робота бази даних.

База даних є ключовою складовою будь-якої інформаційної системи та забезпечує основу для зберігання та організації даних, необхідних для функціонування програмного забезпечення.

2.3 Висновки до розділу

Даний розділ фокусується на проектній частині, включаючи огляд середовища розробки та формалізацію вимог до системи з подальшим моделюванням. Починаючи з огляду середовища розробки, було визначено, що для реалізації найбільш підходять сучасні інтегровані середовища розробки (IDE), такі як Android Studio для платформи Android.

Формалізація вимог до системи була проведена на основі результатів аналізу ринку та вимог користувачів, включаючи як функціональні, так і нефункціональні вимоги. Функціональні вимоги включають основні функції додатка, такі як навігація, інтерактивність з об'єктами на карті, можливості персоналізації. Нефункціональні вимоги охоплюють аспекти продуктивності, безпеки та зручності використання.

РОЗДІЛ 3. ПРАКТИЧНА ЧАСТИНА

3.1 Імпортування Maps SDK

Maps SDK для Android є частиною платформи Google Play services і дозволяє розробникам включати карти та налаштовану картографічну інформацію в їх додатки для платформи Android [24].

Google Maps SDK для Android Studio - це набір інструментів, що дозволяє розробникам інтегрувати функціональність карт Google безпосередньо в їх додатки для платформи Android. Автоматично обробляє доступ до серверів, відображення карт і реагування на жести користувача, такі як кліки і перетягування [25]. За допомогою Maps SDK ви можете налаштовувати мапу відповідно до ваших потреб: змінювати тип мапи (звичайна, супутникова, гібридна), відображати маркери для позначення точок інтересу, налаштовувати інфо-вікна для маркерів, а також додавати полігони та лінії для відображення маршрутів. Ви також можете додавати взаємодію з користувачем, таку як обробка подій натискання на маркери або переміщення по мапі жестами.

Протягом років Google постійно вдосконалює даний сервіс, додаючи нові функції, такі як 3D-відображення, відображення в реальному часі, можливості індивідуалізації карт і покращену інтерактивність

Використання Maps SDK дозволяє створювати додатки з вражаючими картографічними можливостями, які полегшують навігацію та взаємодію з користувачем, розширюючи функціональність вашого додатка та покращуючи користувацький досвід.

Для імпортування даного SDK перш за все потрібно мати акаунт в google cloud, і створити там свій проєкт. Після цього відкриється доступ до бібліотеки, де будуть доступні для скачування різні сервіси. Вигляд бібліотеки показано на рисунку 3.1.

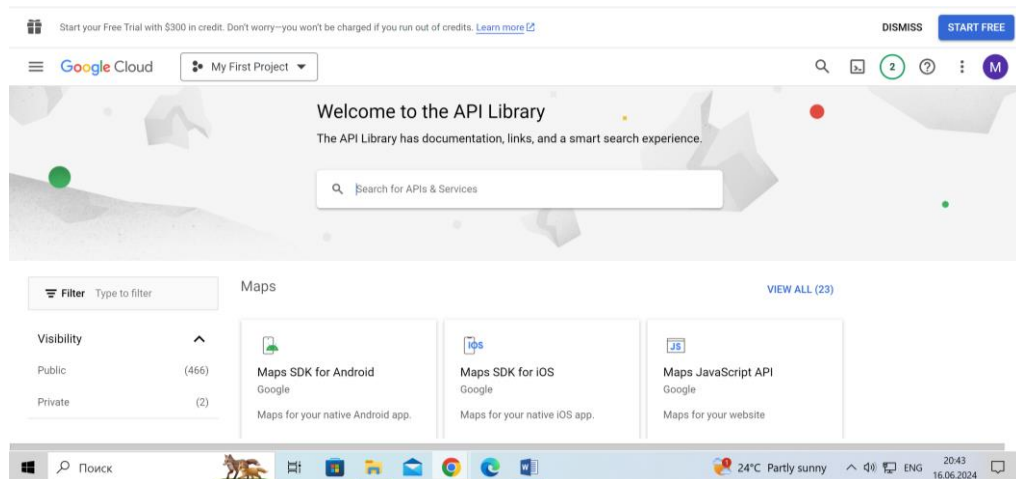


Рисунок 3.1 - Бібліотека в google cloud

Коли відповідний варіант буде вибраний, відкриється сторінка на якій буде вказана інформація про сервіс, а також кнопка для скачування. Дана сторінка зображена на рисунку 3.2.

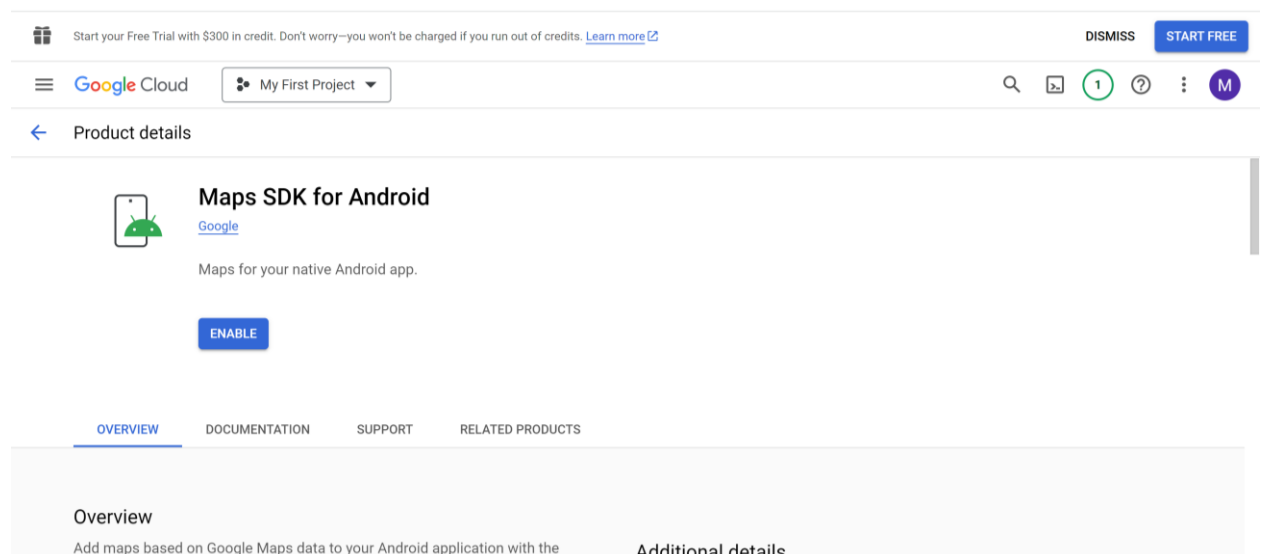


Рисунок 3.2 - Сторінка для завантаження Maps SDK

Роботу з Maps SDK немає сенсу розпочинати без API ключа. Він є унікальним ідентифікатором, який використовується для автентифікації проекту, що викликає увімкнений API [26]. Цей ключ визначає програму як дозволена для використання конкретного набору функцій та можливостей, що пропонуються Maps SDK. Він асоціює запит з проектом Google Cloud для цілей

рахунків і квот [27]. Також використовується для автентифікації додатку під час звернень до сервісів Google Maps, таких як відображення мап, пошук місць або прокладання маршрутів. API-ключ може мати обмеження, пов'язані з обсягом запитів, швидкістю виконання запитів, а також доступом до певних функцій, які можуть бути активовані або вимкнені через панель керування «Google Cloud Console». Ці обмеження дозволяють контролювати використання додатку та забезпечувати оптимальну роботу з серверами Google. Для створення цього ключа потрібно перейти на вкладку «Credentials» і натиснути кнопку «Create Credentials». Після цього ключ буде сформовано, як це показано на рисунку 3.3.

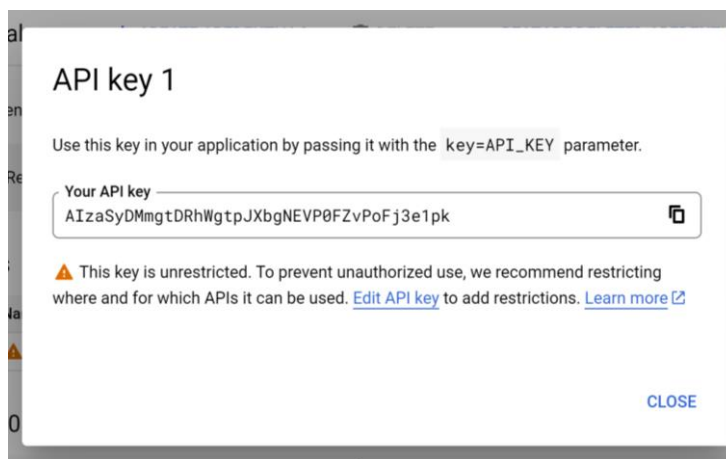


Рисунок 3.3 - Згенерований API ключ

Завершивши роботу в google cloud, можна перейти до android studio. Необхідно імпортувати модуль для роботи з картою вписавши відповідну стрічку коду. Також важливо дати доступ дл локацій та ключа. Відкривши файл «AndroidManifest.xml» треба написати декілька рядків (див. лістинг 3.1).

Лістинг 3.1 – Підключення локацій та API ключа

```
<manifest
xmlns:android="http://schemas.android.com/apk/res/android"
  xmlns:tools="http://schemas.android.com/tools">
  <uses-permission android:name="android.permission.INTERNET"/>
  <uses-permission
android:name="android.permission.ACCESS_FINE_LOCATION"/>
```

```

<uses-permission
android:name="android.permission.ACCESS_COARSE_LOCATION"/>
<application
  <meta-data
    android:name="com.google.android.geo.API.KEY"
    android:value="AIzaSyDMmgtDRhWgtpJXbgNEVP0FZvPoFj3e1pk"
  >

```

В результаті, до проєкту було підключено необхідний SDK Manager, а також базові модулі для роботи з ним.

3.1 Розробка на базі Maps SDK

Першим кроком після підключення карт до проєкту, є виведення їх на екрані. Це реалізовується в файлі `Activity_main.xml`, Який відповідає за елементи інтерфейсу.

Для коректного відображення, необхідно вказати ID, та параметри розташування (див. лістинг 3.2).

Лістинг 3.2 – Виведення карти на екран смартфона

```

<fragment
  android:id="@+id/map"
  android:name="com.google.android.gms.maps.SupportMapFragment"
  android:layout_width="match_parent"
  android:layout_height="match_parent"
  app:layout_constraintBottom_toBottomOf="parent"
  app:layout_constraintEnd_toEndOf="parent"
  app:layout_constraintHorizontal_bias="1.0"
  app:layout_constraintStart_toStartOf="parent"
  app:layout_constraintTop_toTopOf="parent"
  app:layout_constraintVertical_bias="1.0" />

```

Макет (layout) є ресурсом. Ресурс - це частина додатка, яка не є кодом, наприклад такі речі, як файли зображень, аудіофайли та файли XML [28]. Карта є основним елементом додатка на який завжди буде орієнтуватись користувач, її вигляд наведено на рисунку 3.4.

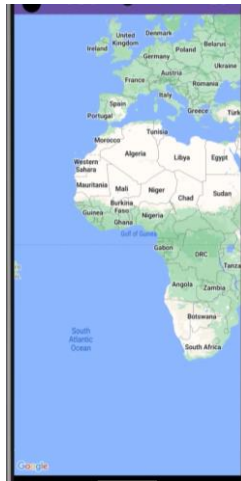


Рисунок 3.4 - Вигляд карти в емуляторі

В залежності від ситуації користувачу можуть бути потрібні інші види карт. Наприклад, людині яка йде в похід важливо мати зображення рельєфу місцевості, а комусь більш зручно розглядати супутникові зображення. На рисунку 3.5 показано вигляд рельєфної карти.



Рисунок 3.5 - Рельєфна карта в емуляторі

Реалізацію потрібно розпочинати зі створення відповідного меню. Для цього слід зробити окремий файл з опціями «Map_options.xml» (див. лістинг. 3.3).

Лістинг 3.3 - Опції для вибору карти

```
<menu xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto">
    <item
        android:id="@+id/normal_map"
        android:title="Normal Map"
        app:showAsAction="never"/>
    <item
        android:id="@+id/hybrid_map"
        android:title="Hybrid Map"
        app:showAsAction="never"/>
    <item
        android:id="@+id/satellite_map"
        android:title="Satellite Map"
        app:showAsAction="never"/>
    <item
        android:id="@+id/terrain_map"
        android:title="Terrain Map"
        app:showAsAction="never"/>
</menu>
```

Після написання параметрів розміщення меню на екрані, можна переходити до самої реалізації функції за допомогою властивості `mapType` (див. лістинг 3.4)

Лістинг 3.4 - Розробка функції вибору виду карти

```
val mapOptionButton:ImageButton =
    findViewById(R.id.mapOptionsMenu)
    val popupMenu = PopupMenu(this, mapOptionButton)
    popupMenu.menuInflater.inflate(R.menu.map_options, popupMenu.menu)
    popupMenu.setOnMenuItemClickListener { menuItem ->
        changeMap(menuItem.itemId)
        true
    }
    mapOptionButton.setOnClickListener {
        popupMenu.show()
    }
private fun changeMap(itemId: Int) {
    when(itemId)
    {
        R.id.normal_map -> mMap?.mapType =
        GoogleMap.MAP_TYPE_NORMAL
        R.id.hybrid_map -> mMap?.mapType =
        GoogleMap.MAP_TYPE_HYBRID
        R.id.satellite_map -> mMap?.mapType =
        GoogleMap.MAP_TYPE_SATELLITE
```

```
R.id.terrain_map -> mMap?.mapType =
GoogleMap.MAP_TYPE_TERRAIN
```

Цей код демонструє створення і обробку подій для PopupMenu в Android, що використовується для вибору типу карти в Google Maps SDK. Спочатку визначається «mapOptionButton» як «ImageButton», який знаходиться за допомогою «findViewById». Наступний рядок створює новий об'єкт «PopupMenu» з контекстом поточної активності та зв'язує його з раніше створеною кнопкою. За допомогою метода «menuInflater.inflate», відбувається надання меню з ресурсу в «PopupMenu». Зазвичай «menuInflater» використовується для створення об'єктів меню з XML-файлів [29]. При виборі елемента меню викликається функція «changeMap», яка приймає ідентифікатор обраного елемента меню. У методі «setOnClickListener» викликається «popupMenu.show()», що показує меню при кліку на кнопку. В методі «changeMap» відбувається зміна типу карти на основі обраного елемента меню. Кожен раз, коли користувач вибирає елемент меню з «PopupMenu», відповідний тип карти встановлюється для об'єкту «mMap».

Одним з найважливіших елементів, подібного додатка є функція пошуку потрібного місця, як це показано на рисунку 3.6

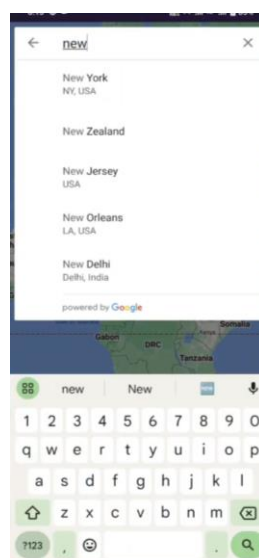


Рисунок 3.6 - Функція пошуку в емуляторі

Спочатку потрібно вставити на панелі поле для введення тексту (див лістинг 3.5). Найкращим варіантом буде його розміщення з самого верху по середині.

Лістинг 3.5. Розміщення текстового поля для пошуку

```
<SearchView
    android:id="@+id/MapSearch"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_margin="10dp"
    android:queryHint="Search..."
    android:iconifiedByDefault="false"
    android:elevation="5dp"
    android:background="@drawable/bg_search_view"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintTop_toTopOf="parent"
/>
```

Реалізація функціоналу пошуку буде здійснюватись за допомогою методу «SearchView» в Android, де користувачі можуть вводити запити про місцеположення, і програма відображає його на карті (див. лістинг 3.6).

Лістинг 3.6 – Впровадження функції пошуку

```
mapSearchView.setOnQueryTextListener(object :
    SearchView.OnQueryTextListener {
        override fun onQueryTextSubmit(query: String?): Boolean {
            query?.let { location ->
                val geocoder = Geocoder(this@MainActivity)
                try {
                    val addressList =
geocoder.getFromLocationName(location, 1)
                    if (addressList.isNotEmpty()) {
                        val address = addressList[0]
                        val latLng = LatLng(address.latitude,
address.longitude)

mMap.addMarker(MarkerOptions().position(latLng).title(location))

mMap.animateCamera(CameraUpdateFactory.newLatLngZoom(latLng, 10F))
                    } else {

Toast.makeText(this@MainActivity, "Місцеположення для '$location'
не знайдено", Toast.LENGTH_SHORT).show()
                }
            } catch (e: IOException) {
```

```

        e.printStackTrace()
        // Обробити помилку геокодування
        Toast.makeText(this@MainActivity, "Сталася
помилка геокодування", Toast.LENGTH_SHORT).show()
    }
}
return false
}

```

В даному коді використовується інтерфейс «OnQuareyTextListener» для відправки пошукового запиту. Клас «geocoder» надає можливість геокодування. Об'єкт «LatLng» ставить точку у відповідному місці. Також Повертає об'єкт «CameraUpdate», який переміщує центр екрану до заданої широти і довготи за допомогою об'єкта «LatLng», і переміщується до вказаного рівня масштабування [30]. Також буда написана обробка помилок аби покращити стабільності.

Подорожуючи транспортом по містах, кожному туристу буде не лишнім знати про пробки, які можуть бути в нього на шляху. Для цього слід створити окрему кнопку, яка буде підсвічувати різним кольором рівень загруженості доріг. Віджет буде знаходитись справа внизу, а для його роботи треба виокористати властивість “isTrafficEnable”, яка відповідає за відображення заповненості шляху транспортом (див. лістинг 3.7).

Лістинг 3.7 – Реалізація відображення трафіка

```

var trafficonoff:ImageView = findViewById(R.id.trafficonof)
    mMap.isMyLocationEnabled = true;
    trafficonoff.setOnClickListener {
        if (!issofed){
            mMap.isTrafficEnabled = true;

trafficonoff.setImageResource(R.drawable.trafficonn)
            issofed = true
        }
        else{
            mMap.isTrafficEnabled = false;

trafficonoff.setImageResource(R.drawable.trafficoff)
            issofed = false
        }
    }
}

```

За логікою коду, спочатку створюється змінна, яка прив'язується до віджета, що має ідентифікатор “trafficonof”. При натисканні на кнопку, іконка змінюється і показується загруженість доріг. При повторному натисканні все вертається назад.

3.3 Висновки до розділу

Третій розділ присвячений практичній частині, охоплює імпортування та розробку додатка на базі Maps SDK. Початковим етапом було успішно імпортовано Maps SDK, яке є ключовим інструментом для інтеграції картографічних можливостей у мобільний додаток. Це включало створення власного API ключа, встановлення необхідних залежностей, імпортування модулів та налаштування середовища розробки для більш ефективної роботи з SDK.

Далі була показана розробка часткового функціоналу додатка, використовуючи можливості Maps SDK. Основний акцент був зроблений на інтеграції інтерактивної карти з функціоналом навігації, відображенням пунктів інтересу та іншими корисними елементами для туристичного використання. Важливою частиною розробки було забезпечення зручного та інтуїтивно зрозумілого інтерфейсу користувача, що сприяє покращенню загального враження від використання додатку.

РОЗДІЛ 4. БЕЗПЕКА ЖИТТЄДІЯЛЬНОСТІ, ОСНОВИ ОХОРОНИ ПРАЦІ

4.1 Природне середовище і його забруднення

Розглядаючи безпеку людини як поняття, що стосується сутності людського життя, сфери її діяльності і взаємозв'язків з навколишнім середовищем, потрібно розглянути таке поняття, як «середовище», в якому живе і діє людина. Навколишнє природне середовище в ширшому розумінні — космічний простір, а у вузкому — біосфера, зовнішня оболонка Землі, яка охоплює частину атмосфери, гідросферу і верхню частину літосфери, що взаємопов'язані складними біогеохімічними циклами міграції речовин і енергії.

Атмосфера — це газова оболонка Землі, яка обертається разом з нею. Атмосфера поділяється на шари, в яких з висотою змінюється температура. На висоті 8 -1 0 км температура становить 40—50 °С нижче нуля, а на висоті біля 60—70 км знаходиться повітряний шар з досить помірною температурою, близькою до 0 °С. Причини виникнення цього теплого шару атмосфери пояснюються явищем абсорбції молекулами озону і кисню ультрафіолетового випромінювання Сонця. Саме тут проходить захист життя Землі від жорсткого випромінювання Сонця. З віддаленням від Землі змінюється [31].

Літосфера – зовнішня тверда оболонка Землі, яка включає всю земну кору з частиною верхньої мантії Землі й складається з осадових, вивержених і метаморфічних порід. Нижня межа літосфери нечітка й визначається різким зменшенням в'язкості порід, зміною швидкості поширення сейсмічних хвиль і збільшенням електропровідності порід. Товщина літосфери на континентах і під океанами різниться і становить в середньому, відповідно, 25-200 і 5-100 км. У межах літосфери періодично відбуваються грізні екологічні процеси які мають величезне значення для формування екологічних ситуацій у певному регіоні планети [32].

Гідросфера – це водяна оболонка. До надземної частини гідросфери, що вкриває 70% поверхні земної кулі, належать океани, моря, озера, ріки, а також льодовики. Основна частина води перебуває у глибинних зонах Землі — в її мантії. Підземна частина гідросфери охоплює ґрунтові, підґрунтові, напірні й безнапірні води, тріщинні води і води карстових порожнин у легкокорозчинних гірських породах (вапняках, гіпсах тощо). Усі форми водних мас переходять одна в одну у процесі перетворення. Для величезної кількості живих організмів, особливо на ранніх етапах розвитку біосфери, вода була середовищем зародження та розвитку. Без води неможливий фотосинтез, який відбувається в зелених рослинах і лежить в основі біологічного кругообігу речовин на нашій планеті [31].

Область існування живих організмів на Землі називається біосферою (сферою життя). Живий світ Землі, її біосфера, складається з організмів трьох основних типів:

- 1) продуценти, або автотрофи – це організми, що створюють органічну речовину за рахунок утилізації сонячної енергії, води, вуглекислого газу та мінеральних солей;
- 2) консументи або гетеротрофи – організми, що одержують енергію за рахунок харчування автотрофами чи іншими консументами;
- 3) редуценти – мікроорганізми, що розкладають органічну речовину продуцентів і консументів до простих сполук.

Вся ця величезна кількість живих істот знаходиться в надзвичайно складних взаємовідносинах між собою і неживою речовиною. У цілому біосфера дуже схожа на єдиний гігантський суперорганізм, у якому автоматично підтримується гомеостаз – динамічна сталість фізико – хімічних і біологічних властивостей внутрішнього середовища [32].

Якісний стан компонентів середовища буття людини значно впливає на рівень її здоров'я, життєдіяльності та тривалості життя. У процесі життя і виробничої діяльності людина неминуче вносить у середовище, що її оточує, певні зміни, які стосуються як хімічного і біологічного стану середовища, так і

умов існування людини. Ці зміни впливають на параметри клімату, особливо в районах з високою концентрацією населення та виробництва, і ведуть до зміни складу атмосфери. У результаті енергетичного забруднення підвищується енергетичний вплив на людину, який створюють промислові та енергетичні підприємства, об'єкти енергетики, зв'язку, транспорту. Збільшення енергоозброєності промисловості сільського господарства і особливо побуту підвищує загрозу негативного її впливу на людей і збільшує ризик небезпеки [33].

Для послаблення впливу шкідливих речовин на організм людини, для визначення ступеня забрудненості довкілля та впливу на рослинні та тваринні організми, проведення екологічних експертиз стану навколишнього середовища або окремих об'єктів чи районів в усьому світі користуються такими поняттями, як гранично допустимі концентрації (ГДК) шкідливих речовин (полютантів), гранично допустимі викиди (ГДВ), гранично допустимі екологічні навантаження (ГДЕН), максимально допустимий рівень (МДР), тимчасово погоджені викиди (ТПВ) та орієнтовно безпечні рівні впливу (ОБРВ) забруднюючих речовин у різних середовищах.

Доки існують шкідливі види антропогенної діяльності, щоб обмежити їх вплив на природне середовище, потрібно нормувати кількість шкідливих речовин, які викидаються в повітря, фунти, води всіма типами забруднювачів, постійно контролювати викиди різного типу об'єктів, прогнозуючи стан довкілля та приймаючи відповідні санкції і рішення щодо порушників законів про охорону природи. В основу нормування всіх забруднювачів у нормативах різних країн покладено визначення ГДК у різних середовищах. За основу приймають найнижчий рівень забруднення, що ґрунтується на санітарно-гігієнічних нормах. Слід зазначити, що ГДК забруднювачів у нормативах різних країн часто різняться, хоча й незначно. ГДК шкідливої речовини — це такий вміст її у природному середовищі, який не знижує працездатності та самопочуття людини, не шкодить здоров'ю у разі постійного контакту, а також не викликає небажаних (негативних) наслідків у нащадків. Визначаючи ГДК,

враховують ступінь впливу не лише на здоров'я людини, але й на диких та свійських тварин, рослини, гриби, мікроорганізми й природні угруповання в цілому [31].

4.2 Гігієнічні вимоги до організації та обладнання робочих місць з ВДТ

Завдяки досягненням сучасних технологій більшість так би мовити, «канцелярської роботи» в офісі здійснюється з використанням комп'ютерної техніки. Якщо згадати, що в середньому робочий день офісного працівника складає 7-8 годин при п'яти – або шестиденному робочому тижні, можна зробити висновок, наскільки багато часу доводиться проводити віч-на-віч з комп'ютером.

Перелік нормативно-правових актів, що так чи інакше регулюють дане питання, є досить широким. Так, обов'язки роботодавця щодо забезпечення працівникам комфортних та безпечних умов для здійснення роботи, а також права працівників на такі умови передбачено частиною 2 ст. 2 та ч. 1 ст. 21 КЗпП, а також ст. 13 Закону України «Про охорону праці». Даний закон визначає основні положення щодо реалізації конституційного права працівників на охорону їх життя і здоров'я у процесі трудової діяльності. Більшість актів у даній сфері становлять акти підзаконного рівня, а саме, численні правила, інструкції, державні санітарні правила і норми (ДСанПН) тощо, якими врегульовуються окремі моменти щодо власне конструкції електронно-обчислювальної техніки, особливостей облаштування приміщень для роботи з нею та низки інших подібних вимог [34].

Під час проектування устаткування й організації робочого місця необхідно враховувати антропометричні показники жінок (якщо працюють тільки жінки) і чоловіків (якщо працюють тільки чоловіки); якщо устаткування обслуговують жінки й чоловіки, у якості граничних за можливості необхідно застосовувати антропометричні показники жінок 5-го перцентиля для

максимальних розмірів і антропометричні показники чоловіків 95-го перцентилля – для мінімальних.

Конструкція виробничого устаткування й робочого місця повинна забезпечувати оптимальне положення тіла працюючого, що досягається регулюванням:

- висоти робочої поверхні, сидіння й простору для ніг. Регульовані параметри потрібно вибирати за номограмою, наведеною на рисунку 4, а також відповідно до вимог ДСТУ ISO 14738;

- висоти сидіння та підставки для ніг (у разі нерегульованої висоти робочої поверхні). У цьому разі висоту робочої поверхні встановлюють за номограмою для працюючих зростом 1800 мм. Оптимальна робоча поза для менших за зростом працюючих досягається збільшенням висоти робочого сидіння й підставки для ніг на величину, що дорівнює різниці між висотою робочої поверхні для працюючого зростом 1800 мм і висотою робочої поверхні, оптимальної для зросту даного працюючого [35].

Розміри крісла встановлюють з антропометричних даних (згідно з ДСТУ EN 547-3 і ДСТУ ISO 7250) дорослого населення з урахуванням поправки на спеціальний одяг і спорядження, а у випадку обмеженого робочого простору — з урахуванням стандартних об'ємів, що виділяють на об'єктах для робочого місця оператора (згідно з ДСТУ EN 547-1, ДСТУ EN 547-2) [36].

Параметри крісла вимірюють відповідно до вихідного положення спинки крісла. Для не профільованої спинки вихідним є положення, за якого кут її нахилу дорівнює 95° . Для профільованої спинки вихідним є таке положення, при якому дотична до кривизни поперекової опори в точці а буде перпендикулярною горизонтальній площини, що проходить через поверхню сидіння [36].

У конструкції крісла повинні регулюватися висота поверхні сидіння над підлогою й кут нахилу спинки. За необхідності, мають регулюватися також такі параметри: висота спинки над сидінням, висота підлокітників над сидінням, кут

нахилу підлокітників, висота підголівника над сидінням, висота підставки для ніг, кут нахилу підставки для ніг [36].

Регулювання параметрів може бути плавне або східчасте. Крок східчастого регулювання для лінійних параметрів — 15—25 мм [36].

Конструкція крісла, форма, розміри та розташування його елементів повинні забезпечувати оптимальні умови посадки та виходу оператора з крісла, а також швидке залишення оператором крісла в аварійних ситуаціях [36].

Характеристики образного і об'ємно-пластичного вирішення крісла мають бути узгоджені з характером формоутворення інших елементів робочого місця. Колірно-фактурні характеристики конструктивних і оздоблювальних матеріалів мають відповідати вимогам дизайну [36].

Дуже часто використовувані засоби відображення інформації, які вимагають точного й швидкого зчитування показань, повинно розміщувати у вертикальній площині під кутом $\pm 15^\circ$ від нормальної лінії погляду та в горизонтальній площині під кутом $\pm 15^\circ$ від сагітальної площини. Часто використовувані засоби відображення інформації, що вимагають менш точного й швидкого зчитування показань, допускається розташовувати у вертикальній площині під кутом $\pm 30^\circ$ від нормальної лінії погляду й у горизонтальній площині під кутом $\pm 30^\circ$ від сагітальної площини [35].

Форму робочої поверхні слід встановлювати з урахуванням характеру виконуваної роботи. Вона може бути прямокутною, мати виріз для тіла працюючого або поглиблення для настільних машин і т.д. У разі необхідності робочу поверхню можна обладнати підлокітниками. Підставка для ніг повинна бути регульованою по висоті. Її ширина повинна бути не менше ніж 300 мм, довжина - не менше ніж 400 мм. Поверхня підставки повинна бути рифленою. По передньому краю доцільно передбачати бортик висотою 10 мм [35].

Таким чином, для того щоб особи, які працюють з комп'ютерною технікою, меншою мірою втомлювались і зберігали високий рівень працездатності, потрібно раціонально організувати їхні робочі місця [34].

4.3 Висновки до розділу

В даному розділі кваліфікаційної роботи було описано по одному питанню з безпеки життєдіяльності та охорони праці. Першою темою для розгляду було навколишнє середовище та його забруднення. Детально описано його види і роль в житті людини. Також проаналізовано антропогенний вплив та методи нормування для охорони природи. Далі було розглянуто гігієнічні вимоги до організації та обладнання робочих місць з ВДТ. З приводу даної теми можна підкреслити, що забезпечення безпеки та здоров'я працівників вимагає комплексного підходу, що охоплює організаційні та технічні заходи. Важливо розробляти та впроваджувати ефективні стратегії управління ризиками, спрямовані на зменшення впливу шкідливих факторів на здоров'я працівників і покращення їхньої робочої продуктивності.

ВИСНОВКИ

В результаті проведеної роботи було розроблено мобільний застосунок інтерактивної карти на базі Maps SDK. Даний проєкт спрямований на поліпшення навігаційних можливостей для подорожуючих людей.

Окрім того, при виконанні даної роботи було розглянуто різноманітний інструментарій для розробки в Android. Підсумком роботи став невеликий і зручний додаток, який чудово підійде тим, хто не сидить на одному місці, а часто подорожує.

В першому розділі проаналізовано предметну область для розробки мобільного додатку. Спочатку здійснювався огляд ринку цифрових карт, що виявив високий попит на ці технології. Зокрема, виявлено зростаючу популярність мобільних додатків, які надають користувачам можливість швидко та зручно орієнтуватися в новому місцевості. Далі був проведений детальний аналіз існуючих рішень на ринку. Він допоміг оцінити конкуренцію і можливості для розробки. Після цього було визначення потреби потенційних користувачів.

Другий розділ зосереджений на проєктній частині, включаючи огляд середовища розробки та формування вимог до системи. Було обрано найзручніше програмне середовище, описано його інтерфейс, а також визначена мова програмування.

Формалізація вимог до системи була зроблена на основі результатів аналізу ринку та вимог користувачів, включаючи як функціональні, так і нефункціональні вимоги. Функціональні вимоги включають основні функції додатка, такі як навігація, інтерактивність з об'єктами на карті, можливості персоналізації. Нефункціональні вимоги охоплюють аспекти продуктивності, безпеки та зручності використання.

Третій розділ призначений для практичної частині, охоплює імпортування та розробку додатка на базі Maps SDK. Успішне імпортування даної платформи є ключовим моментом для інтеграції картографічних

можливостей у мобільний застосунок. Воно включало створення власного API ключа, встановлення необхідних залежностей, імпортування модулів та налаштування середовища розробки для більш ефективної роботи з SDK.

Після цього була проведена розробка, використовуючи можливості Maps SDK. Акцент був зроблений на інтеграції карти з функціоналом навігації, відображенням пунктів інтересу та іншими корисними елементами для туристичного використання. Важливою частиною розробки було створення зручного та інтуїтивно зрозумілого інтерфейсу користувача, що сприяє полегшенню використання застосунку для звичайних людей.

В четвертому розділі кваліфікаційної роботи було розписано питання з безпеки життєдіяльності та охорони праці. Першою темою для огляду являється навколишнє середовище та його забруднення. Визначено його види і роль в житті людини. Також наведено види впливу та способи нормування для охорони навколишнього середовища. Далі було описано гігієнічні вимоги до організації та обладнання робочих місць з ВДТ. Переглянуто багато електронних ресурсів та ДСТУ на дану тему.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Digital Map Market Size, Share, Industry Forecast - 2032 [Електронний ресурс] – Режим доступу: <https://www.alliedmarketresearch.com/digital-map-market> (дата звернення 1.05.2024).
2. Digital Map Market Size, Share & Trends Analysis [Електронний ресурс] – Режим доступу: <https://www.snsinsider.com/reports/digital-map-market-3556> (дата звернення 1.05.2024).
3. Global Digital Map Market Size & Share Analysis Report, [Електронний ресурс] – Режим доступу: <https://www.grandviewresearch.com/industry-analysis/digital-map-market> (дата звернення 1.05.2024).
4. What is Google Maps and how do you use it? [Електронний ресурс] – Режим доступу: <https://www.techtarget.com/whatis/definition/Google-Maps> (дата звернення 2.05.2024).
5. MAPS.ME для Android: веб-сайт. URL: <https://winsoft.com.ua/android/internet/navigation/maps-me-for-android> (дата звернення 2.05.2024).
6. Kraft C. User Experience Innovation/Christian Kraft - Apress Berkeley, 2016. - 228 с.: ил. - 978-1-4302-4150-8 (дата звернення 3.05.2024).
7. Конспект лекцій з дисципліни «Програмування для мобільних пристроїв» для студентів денної форми навчання спеціальності 126 «Інформаційні системи та технології» / Укладачі: Готович В.А., Михайлович Т.В. – Тернопіль : Тернопільський національний технічний університет імені Івана Пулюя, 2020. - 216 с. (дата звернення 3.05.2024).
8. Мови програмування для мобільної розробки [Електронний ресурс] - Режим доступу: <https://code.tutsplus.com/uk/articles/mobiledevelopment-languages--cms-29138> (дата звернення 3.05.2024).

9. Get to know the Android Studio UI [Електронний ресурс] – Режим доступу: <https://developer.android.com/studio/intro/user-interface> (дата звернення 4.05.2024).
10. Офіційний сайт Kotlin. – Електрон. дан. – Режим доступу: <https://kotlinlang.org/docs/android-overview.html> (дата звернення 4.05.2024).
11. Kotlin - Overview [Електронний ресурс] – Режим доступу: https://www.tutorialspoint.com/kotlin/kotlin_overview.htm (дата звернення 4.05.2024).
12. What Is Unified Modeling Language (UML)? Types and Uses [Електронний ресурс] – Режим доступу: <https://www.indeed.com/career-advice/career-development/what-is-uml> (дата звернення 4.05.2024).
13. UML для бізнес-моделювання: для чого потрібні діаграми процесів [Електронний ресурс] - Режим доступу: <https://evergreens.com.ua/ua/articles/uml-diagrams.html> (дата звернення 4.05.2024).
14. Class Diagram | Unified Modeling Language [Електронний ресурс] – <https://www.geeksforgeeks.org/unified-modeling-language-uml-class-diagrams/> (дата звернення 5.05.2024).
15. What is Activity Diagram: веб-сайт. URL: <https://www.visual-paradigm.com/guide/uml-unified-modeling-language/what-is-activity-diagram/> (дата звернення 5.05.2024).
16. Ambler S. The Elements of UML 2.0 Style/Scott W. Ambler - Cambridge University Press, 2010. - 188 с. ил. – 9780511817533 (дата звернення 5.05.2024).
17. Офіційний сайт Lucidchart. – Електрон. дан. – Режим доступу: <https://www.lucidchart.com/pages/uml-object-diagram> (дата звернення 6.05.2024).
18. Object diagrams [Електронний ресурс] – Режим доступу: <https://www.ibm.com/docs/en/dma?topic=diagrams-object> (дата звернення 6.05.2024).

19. Rumpe B. Modeling with UML/Bernhard Rumpe - Springer International Publishing Switzerland, 2016. - 281 с. ил. - 978-3-319-33933-7 (дата звернення 6.05.2024).

20. Петрик М.Р. Моделювання програмного забезпечення : науково методичний посібник / М.Р. Петрик, О.Ю. Петрик – Тернопіль : Вид-во ТНТУ імені Івана Пулюя, 2015. – 200 с. (дата звернення 6.05.2024).

21. Functional and Nonfunctional Requirements: Specification and Types [Електронний ресурс] – Режим доступу: <https://www.altexsoft.com/blog/functional-and-non-functional-requirements-specification-and-types/> (дата звернення 7.05.2024).

22. Guide to mapping requirements to user interface specifications [Електронний ресурс] – Режим доступу: https://www.researchgate.net/publication/243533328_Guide_to_mapping_requirements_to_user_interface_specifications (дата звернення 7.05.2024).

23. Documenting Your Database Requirements [Електронний ресурс] – Режим доступу: <https://www.linkedin.com/pulse/documenting-your-database-requirements-turneroundtech-no5uf> (дата звернення 7.05.2024).

24. Maps and places [Електронний ресурс] – Режим доступу: <https://developer.android.com/develop/sensors-and-location/location/maps-and-places> (дата звернення 8.05.2024).

25. Maps SDK for Android: веб-сайт. URL: <https://console.cloud.google.com/apis/library/maps-android-backend.googleapis.com?hl=en-GB> (дата звернення 8.05.2024).

26. Google Maps API Key [Електронний ресурс] – Режим доступу: <https://developer.android.com/develop/sensors-and-location/location/maps-and-places> (дата звернення 8.05.2024).

27. Офіційний сайт Microsoft. – Електрон. дан. – Режим доступу: <https://learn.microsoft.com/ru-ru/previous-versions/xamarin/android/platform/maps-and-location/maps/obtaining-a-google-maps-api-key?tabs=windows> (дата звернення 8.05.2024).

28. Phillips B., Stewart C., & Marsicano K. Android Programming: The Big Nerd Ranch Guide - Big Nerd Ranch, 2015. - 618 с. ил. - 978-0134171494 (дата звернення 8.05.2024).
29. MenuInflater [Електронний ресурс] – Режим доступу: <https://developer.android.com/reference/android/view/MenuInflater> (дата звернення 8.05.2024).
30. Class CameraUpdateFactory [Електронний ресурс] – Режим доступу: <https://www.cercalia.com/api/android/sdk/com/nexusgeographics/cercalia/maps/CameraUpdateFactory.html> (дата звернення 9.05.2024).
31. Желібо Є. П., Заверуха Н. М., Запарний В. В. Ж516 Безпека життєдіяльності: Навч. посіб. / За ред, Є. П. Желібо. 6-е вид.- К.: Каравела, 2008. — 344 с. (дата звернення 9.05.2024).
32. Структура природного середовища [Електронний ресурс] - режим доступу: <https://kc.pnu.edu.ua/wp-content/uploads/sites/11/2021/02/General-and-chemical-ecology1.pdf> (дата звернення 9.05.2024).
33. Толлок А.О. Крюковська О.А. Безпека життєдіяльності: Навч. посібник. – 2011. – 215 с. (дата звернення 9.05.2024).
34. Гігієнічні вимоги до організації робочих місць [Електронний ресурс] - режим доступу: <https://oppb.com.ua/news/gigiyenichni-vymogy-do-organizaciyi-i-obladnannya-robochuh-misc> (дата звернення 9.05.2024).
35. ДСТУ 8604:2015 Дизайн і ергономіка. Робоче місце для виконання робіт у положенні сидячи. Загальні ергономічні вимоги (дата звернення 9.05.2024).
36. ДСТУ 7951:2015 Дизайн і ергономіка. Крісло оператора. Загальні ергономічні вимоги (дата звернення 9.05.2024).