

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Чорноморський національний університет
імені Петра Могили
Факультет комп'ютерних наук
Кафедра інтелектуальних інформаційних систем

ДОПУЩЕНО ДО ЗАХИСТУ
Завідувач кафедри інтелектуальних
інформаційних систем, д-р техн. наук, проф.
_____Ю. П. Кондратенко
«____»_____2024 р.

КВАЛІФІКАЦІЙНА РОБОТА БАКАЛАВРА

**МОБІЛЬНИЙ ЗАСТОСУНОК ДЛЯ ІНФОРМАЦІЙНОЇ
ПІДТРИМКИ ВЛАСНИКІВ ДОМАШНІХ ТВАРИН**

Спеціальність 122 «Комп'ютерні науки»

122 – КРБ – 401з.221201004

Виконав студент 4-го курсу, групи 401з
_____ *А. О. Паламарчук*
«17» червня 2024 р.

Керівник: д-р фіз.-мат. наук, професор
_____ *Е. А. Лисенков*
«17» червня 2024 р.

Миколаїв – 2024

Кафедра інтелектуальних інформаційних систем
Мобільний застосунок для інформаційної підтримки власників домашніх тварин

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Чорноморський національний університет ім. Петра Могили
Факультет комп'ютерних наук
Кафедра інтелектуальних інформаційних систем

Рівень вищої освіти бакалавр
Спеціальність 122 «Комп'ютерні науки»
(шифр і назва)
Галузь знань 12 «Інформаційні технології»
(шифр і назва)

ЗАТВЕРДЖУЮ

Завідувач кафедри інтелектуальних
інформаційних систем, д-р. техн. наук, проф.
_____ Ю. П. Кондратенко
«___» _____ 2022 р.

ЗАВДАННЯ

на виконання кваліфікаційної роботи

Видано студенту групи 401з факультету комп'ютерних наук Паламарчуку Андрію Олександровичу.

1. Тема кваліфікаційної роботи «Мобільний застосунок для інформаційної підтримки власників домашніх тварин».

Керівник роботи Лисенков Едуард Анатолійович, д-р фіз.-мат. наук, професор.

Затв. наказом Ректора ЧНУ ім. Петра Могили від «28» грудня 2024 р. № 271

2. Строк представлення кваліфікаційної роботи студентом «25» червня 2024 р.

3. Вхідні (початкові) дані до роботи: предметна мобільні застосунки.

Очікуваний результат: система надання допомоги власникам домашніх тварин.

4. Перелік питань, що підлягають розробці (зміст пояснювальної записки):
– аналіз предметної сфери розробки мобільних застосунків;

Кафедра інтелектуальних інформаційних систем

Мобільний застосунок для інформаційної підтримки власників домашніх тварин

- обґрунтування вибору технологій і засобів розробки мобільного застосунку для інформаційної підтримки власників домашніх тварин;
- розробка і здійснення програмної реалізації застосунку для інформаційної підтримки власників домашніх тварин.

5. Перелік графічного матеріалу: презентація.

6. Завдання до спеціальної частини: «Охорона праці при виникненні надзвичайних ситуацій пов'язаних з пожежею».

7. Консультанти розділів роботи

Розділ	Прізвище, ініціали та посада консультанта	Підпис
Спеціальна частина з охорони праці	Григор'єва Л. І., д-р. біол. наук, професор кафедри екології	

Керівник роботи д-р фіз.-мат. наук, професор Лисенков Е. А.
(наук. ступінь, вчене звання, прізвище та ініціали)

_____ (підпис)

Завдання прийнято до виконання Паламарчук А. О.
(прізвище та ініціали)

_____ (підпис)

Дата видачі завдання «_____» _____ 2024 р.

Кафедра інтелектуальних інформаційних систем
Мобільний застосунок для інформаційної підтримки власників домашніх тварин

КАЛЕНДАРНИЙ ПЛАН
виконання кваліфікаційної роботи бакалавра

Тема: Мобільний застосунок для інформаційної підтримки власників домашніх тварин.

№	Найменування роботи	Початок	Закінчення	Примітки
1	Визначення керівника і теми КРБ. Подання заяви на затвердження теми КРБ	01.09.2023	05.02.2024	Виконано
2	Отримання завдання на виконання КРБ	10.02.2024	10.02.2024	Виконано
3	Складання календарного плану	11.02.2024	18.02.2024	Виконано
4	Огляд літератури за темою дослідження. Аналіз предметної сфери розробки мобільних застосунків	19.02.2024	12.03.2024	Виконано
5	Вибір технологій та інструментальних засобів розробки системи	13.03.2024	18.03.2024	Виконано
6	Створення дизайну, проєктування та програмна реалізація, тестування системи	19.03.2024	20.05.2024	Виконано
7	Робота над розділами фахової частини КРБ	21.05.2024	26.05.2024	Виконано
8	Проходження першої переддипломної практики, збір та аналіз матеріалів, остаточне оформлення розділів фахової частини КРБ	27.05.2024	27.05.2024	Виконано
9	Розробка спеціальної частини з охорони праці	28.05.2024	01.06.2024	Виконано
10	Обговорення отриманих результатів з керівником та попередній захист КРБ	02.06.2024	05.06.2024	Виконано
11	Корегування роботи за результатами попереднього захисту	06.06.2024	09.06.2024	Виконано
12	Проходження другої переддипломної практики, збір та аналіз матеріалів, остаточне оформлення розділів фахової частини КРБ	10.06.2024	10.06.2024	Виконано
13	Остаточне оформлення пояснювальної записки та слайдів доповіді до захисту	11.06.2024	20.06.2024	Виконано

Кафедра інтелектуальних інформаційних систем
Мобільний застосунок для інформаційної підтримки власників домашніх тварин

14	Подання рецензенту та рецензування КРБ	21.06.2024	22.06.2024	Виконано
15	Подання КРБ, її електронної копії та інших документів (відгуку, рецензії) до захисту	23.06.2024	25.06.2024	Виконано
16	Захист КРБ перед ЕК	28.06.2024	28.06.2024	

Розробив студент Паламарчук А. О.
(прізвище та ініціали)

_____ (підпис)

Керівник д-р фіз.-мат. наук, професор Лисенков Е. А.
(наук. ступінь, вчене звання, прізвище та ініціали)

_____ (підпис)

«_____» _____ 2024 р.

АНОТАЦІЯ

кваліфікаційної роботи студента групи 401з ЧНУ ім. Петра Могили

Паламарчука Андрія Олександровича

Тема: «Мобільний застосунок для інформаційної підтримки власників домашніх тварин»

Бакалаврська кваліфікаційна робота присвячена створенню мобільного застосунку для інформаційної підтримки власників домашніх тварин. На сьогоднішній день люди заводять собі надзвичайно велику кількість домашніх тварин, особливо екзотичних. Причина придбання домашньої тваринки є не тільки любов до тварин, а й допомога тваринам, які залишилися без притулку через природні катаклізми, війну та жорстоке ставлення минулих власників. Ще, як одну з причин можна назвати вибір домашнього улюбленця для психологічної допомоги, наприклад багато дітей й дорослих, які пережили досить важкі психологічні моменти життя за допомогою тварин можуть відволіктися від жахливості сьогодення та знайти в собі мотивацію продовжувати жити.

Враховуючи вище перелічені аспекти приходимо до висновку, що на сьогоднішній досить актуальною темою є розробка мобільного застосунку, що буде містити в собі інформацію для допомоги власникам домашніх тварин.

Об'єкт роботи – процес отримання інформації щодо домашніх тварин.

Предмет роботи – застосунок, що допомагає при пошуку інформації про домашніх тварин.

Мета роботи – підвищення якості життя людей та їх домашніх улюбленців.

Бакалаврська кваліфікаційна робота складається з фахової та спеціальної частини з охорони праці. Пояснювальна записка фахової частини складається зі вступу, трьох розділів, висновків та додатків. У першому

Кафедра інтелектуальних інформаційних систем

Мобільний застосунок для інформаційної підтримки власників домашніх тварин
розділі розкрито теоретичні засади. У другому розділі обґрунтовано вибір технологій і засобів розробки системи. У третьому розділі описано проєктування та програмну реалізацію системи аналізу і прогнозування курсу акцій.

Бакалаврська кваліфікаційна робота містить 63 сторінки (без додатків), 31 рисунок, 3 таблиці, 35 джерел, 1 додаток.

Ключові слова: мобільний застосунок, домашні тварини, Python, PyCharm.

ABSTRACT

qualification work of the student of group 401z at Petro Mohyla BSNU

Palamarchuk Andrii Oleksandrovych

Topic: "Mobile Application for Information Support of Pet Owners"

The bachelor's qualification work is dedicated to the creation of a mobile application for providing information support to pet owners. Nowadays, people are acquiring a significant number of pets, especially exotic ones. The reasons for acquiring pets are not only due to a love for animals but also to help animals left homeless due to natural disasters, war, and cruel treatment by previous owners. Additionally, one of the reasons could be choosing a pet for psychological assistance, as many children and adults who have experienced significant psychological challenges can find distraction from the horrors of the present and motivation to continue living through their pets.

Considering the above aspects, we conclude that the development of a mobile application that will contain information to help pet owners is a highly relevant topic today.

The object of the work is the process of obtaining information about pets.

The subject of the work is an application that helps in finding information about pets.

The goal of the work is to improve the quality of life of people and their pets.

The bachelor's qualification work consists of a professional and a special part on labor protection. The explanatory note of the professional part consists of an introduction, three chapters, conclusions, and appendices. The first chapter covers theoretical principles. The second chapter substantiates the choice of technologies and tools for system development. The third chapter describes the

Кафедра інтелектуальних інформаційних систем

Мобільний застосунок для інформаційної підтримки власників домашніх тварин

design and software implementation of the system for analyzing and forecasting stock prices.

The bachelor's qualification work contains 63 pages (excluding appendices), 31 figures, 3 tables, 35 sources, 1 appendices.

Keywords: mobile application, pets, Python, PyCharm, .

ПЕРЕЛІК СКОРОЧЕНЬ.....	11
ВСТУП	12
1 ТЕОРЕТИЧНІ ЗАСАДИ РОЗРОБКИ МОБІЛЬНОГО ЗАСТОСУНКУ ДЛЯ ІНФОРМАЦІЙНОЇ ПІДТРИМКИ ВЛАСНИКІВ ДОМАШНІХ ТВАРИН	14
1.1 Мобільні застосунки та їх особливості.....	14
1.2 Життєвий цикл додатків.....	14
1.3 Активності та фрагменти	18
1.4 Передумови створення застосунку для підтримки власників домашніх тварин, аналіз та порівняння існуючих застосунків.....	21
1.5 Збір та обробка інформації.....	28
Висновки до розділу 1	29
2 ТЕХНОЛОГІЇ ТА ЗАСОБИ РОЗРОБКИ МОБІЛЬНОГО ЗАСТОСУНКУ ДЛЯ ІНФОРМАЦІЙНОЇ ПІДТРИМКИ ВЛАСНИКІВ ДОМАШНІХ ТВАРИН	30
2.1 Важливість платформи Android для застосунку.....	30
2.2 Інтегроване середовище розробки Android Studio	32
2.3 Використання Vysor для розробки застосунків.....	34
2.4 Firebase як універсальна хмарна база.....	36
Висновки до розділу 2	38
РОЗДІЛ 3 ПРОГРАМНА РЕАЛІЗАЦІЯ ЗАСТОСУНКУ ДЛЯ ІНФОРМАЦІЙНОЇ ПІДТРИМКИ ВЛАСНИКІВ ДОМАШНІХ ТВАРИН	39
3.1 Додавання API на Google Maps	39
3.2 Налаштування Vysor для симуляції розробленого застосунку.....	40
3.3 Створення сховища для проекту у Firebase.....	46
3.4 Розробка функціоналу та інтерфейсу додатку.....	47
Висновки до розділу 3	60
ВИСНОВКИ.....	61
ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ	62
ДОДАТОК А.....	65

ПЕРЕЛІК СКОРОЧЕНЬ

- ПК** – Персональний комп'ютер
- API** – Application Programming Interface
- FCM** – Firebase Cloud Messaging
- IDE** – Integrated Development Environment
- IOS** – iPhone Operating System
- MVP** – Minimum Viable Product
- SDK** – Software Development Kit
- UI** – User Interface

ВСТУП

Актуальність. Надання інформаційної підтримки власникам домашніх тварин є важливим аспектом не тільки догляду та утримання домашніх улюбленців, а й попереднього вибору домашнього улюбленця.

Перспективним напрямком підвищення ефективності підтримки власників домашніх тварин є використання мобільних застосунків, які можуть автоматизувати процеси консультування, моніторингу та надання рекомендацій.

Мобільні застосунки можуть використовуватися для попереднього вибору домашнього улюбленця, що у майбутньому не призведе до відмови від тваринки.

Також такий додаток передбачає надзвичайно велику кількість можливостей, а саме: пошук ветеринарної допомоги, пошук “pet-friendly” готелів, читання відгуків інших власників, різноманітна допомога з притулком для тваринок та інші.

Вищеперелічені аспекти обумовили мету роботи, яка полягає у підвищенні якості надання інформаційної підтримки власникам домашніх тварин шляхом розробки мобільного застосунку.

Мета – забезпечення інформаційною підтримкою власників домашніх тварин.

Відповідно до поставленої мети було сформульовано завдання:

- здійснити аналіз предметної сфери надання інформаційної підтримки власникам домашніх тварин та дослідити теоретичні засади використання мобільних застосунків у цій сфері;
- обґрунтувати вибір технологій та інструментальних засобів розробки мобільного застосунку для інформаційної підтримки власників домашніх тварин;

Кафедра інтелектуальних інформаційних систем

Мобільний застосунок для інформаційної підтримки власників домашніх тварин

– розробити, здійснити програмну реалізацію мобільного застосунку для інформаційної підтримки власників домашніх тварин, дослідити якість його роботи та ефективність надання інформаційної підтримки.

Об’єкт роботи – процес надання інформаційної підтримки власникам домашніх тварин з використанням мобільного застосунку.

Предмет роботи – програмні засоби та методи надання інформаційної підтримки власникам домашніх тварин, що базуються на мобільних технологіях.

Практичне значення отриманих результатів полягає в тому, що використання розробленого мобільного застосунку дозволить підвищити ефективність надання інформаційної підтримки власникам домашніх тварин за рахунок автоматизації процесів консультування, моніторингу та надання рекомендацій, що сприятиме покращенню людей та їх улюбленців.

Кафедра інтелектуальних інформаційних систем
Мобільний застосунок для інформаційної підтримки власників домашніх тварин

1 ТЕОРЕТИЧНІ ЗАСАДИ РОЗРОБКИ МОБІЛЬНОГО ЗАСТОСУНКУ ДЛЯ ІНФОРМАЦІЙНОЇ ПІДТРИМКИ ВЛАСНИКІВ ДОМАШНІХ ТВАРИН

1.1 Мобільні застосунки та їх особливості

Мобільні застосунки – це програми, що спеціально створені для використання на мобільних пристроях, таких як смартфони та планшети.

Головна мета мобільних застосунків - це надання користувачам різних послуг і можливостей, які полегшують їхнє щоденне життя або взаємодію з технологіями, до таких послуг відносяться ігри, соціальні мережі, застосунки для слідкування за здоров'ям, освітні застосунки, застосунки для управління фінансами, онлайн магазини та інше.

Основні риси мобільних додатків включають їхню мобільність, що дає змогу користувачам отримувати доступ у будь-якому місці та часі, простоту використання з легким інтуїтивним інтерфейсом, функціональність, яка задовольняє потреби користувачів, здатність до праці на мобільних пристроях, враховуючи їхні обмеження, такі як розмір екрану та ресурси, а також регулярні оновлення та підтримка для покращення та виправлення помилок [1-4].

Типи мобільних додатків включають соціальні мережі, ігри, утиліти, освітні програми, фінанси, здоров'я та фітнес, подорожі, новини, електронну комерцію та багато інших. Вони реалізують різноманітні функції, включаючи роботу з мультимедіа, геолокацію, бази даних, інтеграцію з соціальними мережами, сповіщеннями, платіжними системами та інші.

1.2 Життєвий цикл додатків

Життєвий цикл застосунків має важливе значення при розробці, а саме детальне планування кожної фази дозволяє оптимізувати використання

Мобільний застосунок для інформаційної підтримки власників домашніх тварин ресурсів і часу розробників, збалансований підхід до кожної стадії дозволяє виявляти та виправляти помилки на ранніх етапах, що підвищує якість продукту, життєвий цикл дозволяє вчасно реагувати на потреби користувачів, що робить додаток конкурентоспроможним і актуальним та допомагає оптимізувати використання ресурсів та планувати успішні стратегії випуску додатку на ринок [1-5].

Розуміння та ефективне управління життєвим циклом застосунків є критично важливим для успішного розвитку в галузі розробки мобільних застосунків.

Етапи життя мобільного застосунку:

- Розробка;
- Тестування;
- Розгортання;
- Підтримка.

Фаза розробки мобільного застосунку є однією з найбільш важливих стадій життєвого циклу. На етапі розробки проводиться детальне планування, включаючи визначення функціональних та нефункціональних вимог, аналіз потреб користувачів та розробку архітектури додатку. Наступним кроком є проектування, де створюється інтерфейс користувача, дизайн та інші аспекти відповідно до вимог та рекомендацій платформи. Після цього розробники переходять до фази реалізації, де пишуть програмний код, використовуючи відповідні мови програмування (наприклад, Java або Kotlin для Android). Ця фаза включає в себе створення функціональності додатку, реалізацію бізнес-логіки та інтеграцію з різними сервісами та API. На кожному етапі розробки проводиться тестування для виявлення та виправлення помилок, а також для перевірки відповідності функціональності вимогам. У цій фазі також може бути створено прототипи або MVP (Minimum Viable Product), щоб швидко перевірити ідею та отримати зворотний зв'язок від користувачів. В цілому, фаза розробки є

Мобільний застосунок для інформаційної підтримки власників домашніх тварин ключовим етапом, що вимагає уваги до деталей, технічної експертизи та співпраці між розробниками та дизайнерами для створення мобільного застосунку, який задовольняє потреби користувачів та відповідає вимогам ринку [5-7].

Фаза тестування мобільного застосунку - це критично важлива частина його розробки, що включає в себе ретельну перевірку всіх аспектів додатку перед його випуском. Під час цієї фази проводяться різні види тестів, включаючи функціональне тестування для перевірки виконання основних функцій, інтерфейсне тестування для оцінки користувацького досвіду, тестування продуктивності для визначення швидкості та ефективності роботи, а також тестування на відповідність вимогам та безпеку. Крім того, важливим етапом є тестування на реальних пристроях, що дозволяє перевірити роботу додатку на різних моделях та версіях операційних систем. Всі ці процеси спрямовані на забезпечення високої якості, надійності та коректної роботи додатку перед його випуском, щоб забезпечити задоволення користувачів та успішний виступ на ринку мобільних додатків.

Фаза розгортання мобільного застосунку – це етап, де додаток готується до випуску та використання користувачами. Під час цієї фази розробники проводять підготовчі роботи для розміщення додатку на відповідних майданчиках, таких як Google Play Store для Android або App Store для iOS. Це включає в себе створення пакету додатку, включаючи скомпільований код, ресурси та метадані, а також підготовку опису, зображень та інших матеріалів для презентації додатку. Після цього додаток надсилається на платформу розгортання, де він проходить процес перевірки та схвалення відповідно до вимог і політики майданчика. Після успішного схвалення додаток стає доступним для завантаження та встановлення користувачами. Крім того, фаза розгортання також включає підготовку стратегії маркетингу та просування додатку для забезпечення його успішного запуску на ринку, а також підтримку та оновлення після випуску для

Мобільний застосунок для інформаційної підтримки власників домашніх тварин забезпечення продовженого функціонування та задоволення потреб користувачів. Таким чином, фаза розгортання є важливим етапом в процесі розробки мобільного застосунку, що дозволяє забезпечити його успішний випуск та ефективно використання користувачами [5-7].

Фаза підтримки мобільного застосунку - забезпечує тривалу та успішну експлуатацію додатку після його випуску. Під час цієї фази розробники відстежують роботу додатку, моніторять його використання та збирають відгуки користувачів для виявлення потенційних проблем чи покращень. Важливою складовою цієї фази є виправлення помилок - вчасне виявлення та усунення будь-яких несправностей або помилок, які можуть вплинути на користувацький досвід та роботу додатку.

Під час фази оптимізації та вдосконалення мобільного застосунку розробники аналізують використання додатку, збирають відгуки користувачів, та використовують дані аналітики для виявлення слабких місць та можливостей для покращення. Це може включати вдосконалення інтерфейсу користувача, оптимізацію продуктивності, виправлення помилок та удосконалення функціональності. Крім того, розробники можуть використовувати нові технології та підходи для оптимізації роботи додатку, такі як кешування даних, покращення алгоритмів обробки, а також використання спеціалізованих бібліотек чи фреймворків. Оптимізація та вдосконалення дозволяють підтримувати конкурентоспроможність додатку на ринку, забезпечуючи його здатність задовольняти зростаючі потреби користувачів та пристосовуватися до змін у технологічному середовищі. Таким чином, ця фаза дозволяє розробникам постійно вдосконалювати та пристосовувати свій продукт до найвищих стандартів якості та ефективності.

На стадії завершення життєвого циклу мобільного застосунку розробники приймають рішення щодо припинення підтримки та обслуговування додатку. Це може бути пов'язано з різними факторами, такими як зміни в бізнес-стратегії компанії, низька активність користувачів,

Мобільний застосунок для інформаційної підтримки власників домашніх тварин відсутність рентабельності або технологічна застарілість. Під час завершення життєвого циклу розробники припиняють видачу оновлень, роблять анонси про закриття додатку для нових завантажень, а також надають користувачам можливість зберегти свої дані або перейти до альтернативних рішень. Крім того, проводяться необхідні процедури звільнення ресурсів, таких як припинення оплати хостингу або припинення дії ліцензійних угод. Однак важливо зазначити, що завершення життєвого циклу не завжди означає повну видалення додатку. У деяких випадках він може продовжувати працювати на пристроях користувачів, проте без подальшої підтримки та оновлень. В цілому, фаза завершення життєвого циклу важлива для здійснення відповідальної та ефективного управління долею мобільного застосунку, забезпечуючи зручність користувачів та дотримання вимог законодавства та індустрійних стандартів.

1.3 Активності та фрагменти

Активності є вікнами додатку, з якими користувач може взаємодіяти, і кожна з них пройде свій власний життєвий цикл, який включає такі етапи, як створення, запуск, призупинення, відновлення та знищення.

При створенні додатку зазвичай визначаються різні активності для різних екранів або функціональностей. Наприклад, одна активність може бути призначена для головного екрану, інша для налаштувань, а ще одна для відображення деталей певного елемента.

Життєвий цикл активності розпочинається з методу `onCreate()`, де відбувається ініціалізація структури інтерфейсу користувача та налаштування додатку. Після цього можуть викликатися методи `onStart()`, `onResume()`, `onPause()` та `onStop()` залежно від взаємодії користувача з додатком. Життєвий цикл завершується методом `onDestroy()`, коли активність закривається або знищується [8-10].

Основні методи життєвого циклу активності, такі як `onCreate()`, `onStart()`, `onResume()`, `onPause()`, `onStop()` та `onDestroy()`, дозволяють розробникам керувати циклом життя активності та виконувати різні операції залежно від стану активності.

Фрагменти, з свого боку, є невеликими та повторно використовуваними частинами інтерфейсу користувача, які можуть динамічно комбінуватися на екрані. Це дозволяє розробникам створювати гнучкі та адаптивні інтерфейси, які легко адаптуються до різних умов використання. Життєвий цикл фрагментів співпадає з життєвим циклом активності, у якій вони розміщені, і включає різні етапи, які впливають на їхню поведінку та взаємодію з іншими елементами додатку.

Керування фрагментами в межах активності включає в себе використання ряду методів та транзакцій фрагментів для ефективного управління їхнім життєвим циклом та взаємодією з інтерфейсом користувача. Основні методи керування фрагментами в рамках активності включають додавання, видалення та заміну фрагментів. Метод `add()` використовується для додавання нового фрагмента на екран, `remove()` для видалення фрагмента, а `replace()` для заміни одного фрагмента іншим. Транзакції фрагментів дозволяють групувати операції додавання, видалення та заміни фрагментів, забезпечуючи атомарність операцій і виконання всіх або жодної з них. Крім того, вони дозволяють виконувати операції у певному порядку та забезпечують плавні переходи між фрагментами [8-10].

Взаємодія між фрагментами та активностями є важливим аспектом розробки мобільних додатків для платформи Android, оскільки вона дозволяє різним компонентам додатку обмінюватися даними та взаємодіяти між собою. Існують кілька способів комунікації між фрагментами та активностями. Один з них – використання інтерфейсів зв'язку, де фрагмент визначає інтерфейс, який реалізує активність, і може викликати методи цього інтерфейсу для передачі даних або сповіщення активності про події. Інший

Мобільний застосунок для інформаційної підтримки власників домашніх тварин
спосіб – виклик методів між фрагментами, де фрагменти можуть безпосередньо викликати методи інших фрагментів, що знаходяться в тій же активності, за допомогою посилань на фрагменти або методів менеджера фрагментів. Також можна використовувати *shared view models*, коли декілька фрагментів мають обмінюватися даними, через створення та використання спільного *view model*, який забезпечує доступ до даних для всіх фрагментів та активностей, які його використовують. І, нарешті, фрагментам можна передавати дані через *Bundle*, що містить аргументи, або використовувати методи фрагментів для передачі та отримання потрібних даних.

Узагальнюючи використання активностей та фрагментів в розробці мобільних додатків для *Android*, можна зазначити, що обидва компонента відіграють ключову роль у створенні гнучких та функціональних інтерфейсів користувача. Активності є базовими компонентами додатку, представляючи весь екран або один його шар, та управляючи життєвим циклом додатку. Вони відповідають за ініціалізацію та управління фрагментами, обробку подій та зв'язок з системою *Android*. З іншого боку, фрагменти дозволяють розбити інтерфейс на невеликі та повторно використовувані частини, що полегшує розробку, тестування та підтримку додатків. Їхній життєвий цикл більш гнучкий, оскільки вони можуть бути додані, видалені або замінені в межах активності за допомогою транзакцій фрагментів. Крім того, фрагменти дозволяють реалізувати адаптивний інтерфейс, який легко пристосовується до різних розмірів екрану та орієнтацій пристроїв. Порівнюючи життєві цикли, активності мають більш просту структуру, оскільки вони є окремими екранами або вікнами додатку, тоді як фрагменти можуть бути частинами цих екранів, додаватися або видалятися динамічно. Таким чином, активності та фрагменти доповнюють один одного, створюючи потужний інструментарій для розробки мобільних додатків, який забезпечує зручну і ефективну роботу з інтерфейсом користувача.

1.4 Передумови створення застосунку для підтримки власників домашніх тварин, аналіз та порівняння існуючих застосунків

Згідно статистики в Україні більше 60% людей мають домашню тварину, 75% – мали або планують завести. Україна посіла дев'яте місце серед інших країн світу по популярності заведення домашніх тварин. Лідеркою ж серед країн стала Аргентина з максимальним показником в 80%, друге місце – Мексика, і третє – Бразилія (75 відсотків) більш детальну статистику зображено на рисунку 1.1.

PET OWNERSHIP INTERNATIONALLY

Percentage of people living with different pets in 22 countries

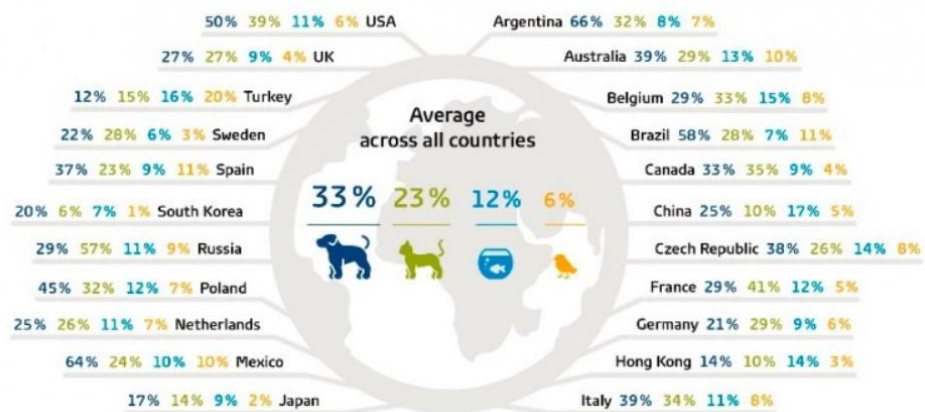


Рисунок 1.1 – Статистичні дані щодо власників домашніх тварин у різних країнах

Кожного року тільки збільшуються кількість людей, які планують завести домашнього улюбленця. Саме тому створення сервісу, який би допомагав власникам у повсякденному житті. Варто описати низку проблем, які можуть виникнути і які є можливість вирішити за допомогою створення застосунку.

1. Пошук ветклінік, грумерів для тварин. Час від часу кожній тваринці необхідні певні процедури. Не кожна людина, особливо яка вперше

Мобільний застосунок для інформаційної підтримки власників домашніх тварин стикнулася із утриманням тварин, знає, де знайти надійну ветеринарну клініку або професійного грумера. Застосунок може надати вичерпну базу даних з контактами, рейтингами та відгуками про клініки та грумерів, що дозволить швидко знайти потрібного спеціаліста поруч.

2. Моніторинг здоров'я та активності. Багато власників не знають, як часто їхнім улюбленцям потрібно гуляти, яку їжу давати або як стежити за їхнім здоров'ям. Застосунок може нагадувати про необхідність прогулянок, годування, візитів до ветеринара, а також відстежувати активність і вагу тварини, пропонуючи рекомендації щодо догляду.

3. Реєстрація і відстеження вакцинацій. Важливо своєчасно вакцинувати тварин, але легко забути про чергову вакцинацію. Застосунок може відстежувати графік вакцинацій і надсилати нагадування, забезпечуючи своєчасний захист від хвороб.

4. Розшук зниклих тварин. У випадку, якщо тварина загубиться, власник може оперативно повідомити про це через застосунок, розмістивши оголошення з фотографією та контактними даними. Інші користувачі застосунку можуть допомогти у пошуках, збільшуючи шанси на швидке повернення улюбленця.

5. Зоомагазини та товари для тварин. Часом важко знайти необхідні товари для тварин, такі як спеціальний корм, іграшки або аксесуари. Застосунок може пропонувати каталог зоомагазинів з можливістю замовлення та доставки необхідних товарів.

6. Соціалізація та взаємодія з іншими власниками. Власникам тварин корисно обмінюватися досвідом та порадами. Застосунок може включати функції соціальної мережі, де користувачі можуть спілкуватися, ділитися фотографіями своїх улюбленців, створювати спільноти за інтересами та організовувати зустрічі.

7. Освітні матеріали та поради. Новачки в утриманні тварин часто стикаються з різноманітними питаннями. Застосунок може містити базу

Мобільний застосунок для інформаційної підтримки власників домашніх тварин знань з порадами від експертів, відеоуроками та статтями про догляд за тваринами, їх тренуванням та поведінкою.

8. Пошук людини, яка могла б вигуляти тварину вранці чи ввечері. Часто у власників тварин не має можливості на прогулянки в певний час доби, тому було б непогано мати можливість якщо що попросити допомоги у когось другого, так само як наприклад няня для дитини.

9. Пошук готелів, спеціального простору або людей, які змогли б посидіти з домашнім улюбленцем на час відрадження або відпустки. Доречі, більша кількість людей не стала б утримувати якусь тваринку саме через необхідність постійної турботи і неможливість покинути дім на довгий період часу через відсутність знайомих які могли б посидіти та потурбуватися про домашнього улюбленця. Зараз, у 21 сторіччі ми завжди кудись біжимо, у нас немає часу і ми постійно подорожуємо і змушені продумати хто буде підкуватися про наших хвостиків. Саме через налічені причини було б непогано мати можливість знайти таких людей та домовитися про умови безпосередньо не виходячи із дому – через застосунок.

Створення застосунку для тварин має значні переваги як для власників домашніх улюбленців, так і для професіоналів у сфері догляду за тваринами. Такий сервіс може значно полегшити життя власників тварин, забезпечити їх улюбленцям найкращий догляд та сприяти зростанню обізнаності щодо правильного утримання тварин.

У світі вже існує чимало застосунків та сервісів для власників домашніх тварин, тож розглянемо існуючі застосунки, проаналізуємо їх можливості та зробимо порівняння.

Petfinder – це популярний сервіс для пошуку та адопції тварин. Основною метою цього застосунку є допомога у пошуку нового дому для безпритульних тварин. Petfinder пропонує базу даних тварин, доступних для адопції, з детальною інформацією про них (рис. 1.2).



Рисунок 1.2 – Логотип та зовнішній вигляд Petfinder

Розроблюваний застосунок також включатиме функцію адопції, але додатково пропонуватиме широкий спектр можливостей, таких як моніторинг здоров'я, пошук ветеринарів, грумерів, а також соціальні функції та освітні матеріали.

Rover – це сервіс, що спеціалізується на пошуку догсіттерів (осіб, які доглядають за тваринами) та осіб, що вигулюють собак. Rover дозволяє власникам тварин знайти надійних людей для догляду за своїми улюбленцями під час їхньої відсутності (рис. 1.3).

Кафедра інтелектуальних інформаційних систем
Мобільний застосунок для інформаційної підтримки власників домашніх тварин

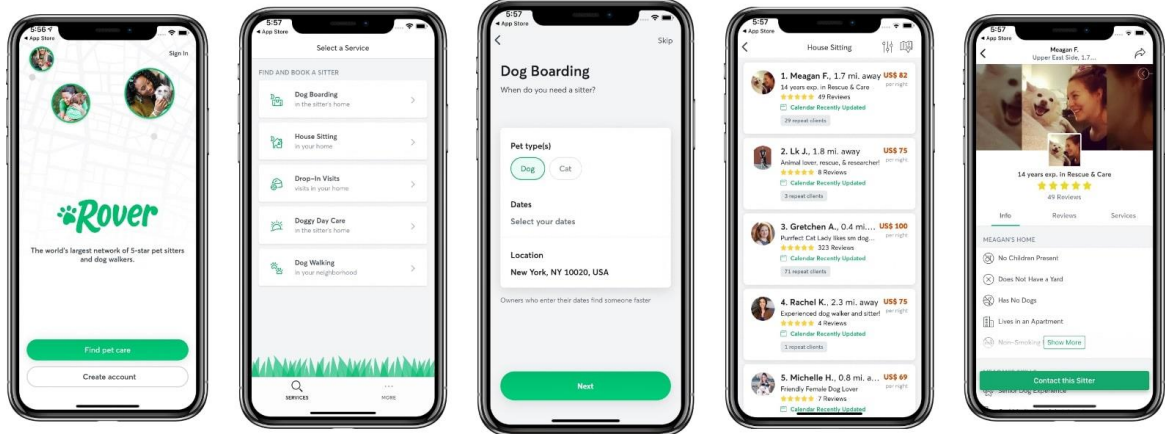


Рисунок 1.3 – Інтерфейс застосунку Rover

Розроблюваний застосунок також включатиме функцію пошуку догсіттерів і людей що вигулюють тварин, але додатково пропонуватиме багато інших корисних функцій, таких як пошук ветклінік, моніторинг здоров'я та активності, та підтримку зоомагазинів.

Веб-застосунок 11pets – застосунок для моніторингу здоров'я домашніх тварин, що включає можливості відстеження графіку вакцинацій, медичних записів та догляду за тваринами (рис. 1.4). Цей застосунок фокусується на здоров'ї та догляді за домашніми улюбленцями [11-13].

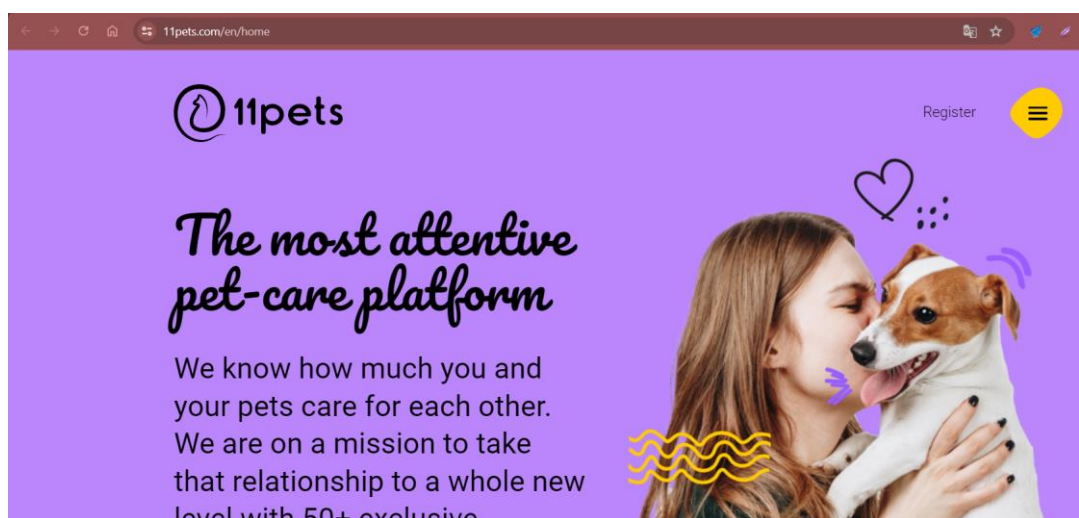


Рисунок 1.4 – Офіційна сторінка 11pets

Кафедра інтелектуальних інформаційних систем
 Мобільний застосунок для інформаційної підтримки власників домашніх тварин
 Розроблюваний застосунок надаватиме аналогічні функції моніторингу здоров'я, але додатково пропонуватиме інші можливості, такі як пошук ветеринарів та грумерів, соціальні функції, а також функції для адопції та пошуку зниклих тварин.

BringFido – це сервіс для пошуку готелів, ресторанів та інших місць, де дозволено перебувати з домашніми тваринами. Застосунок допомагає власникам тварин планувати подорожі та візити, враховуючи можливість перебування з улюбленцями (рис. 1.5).

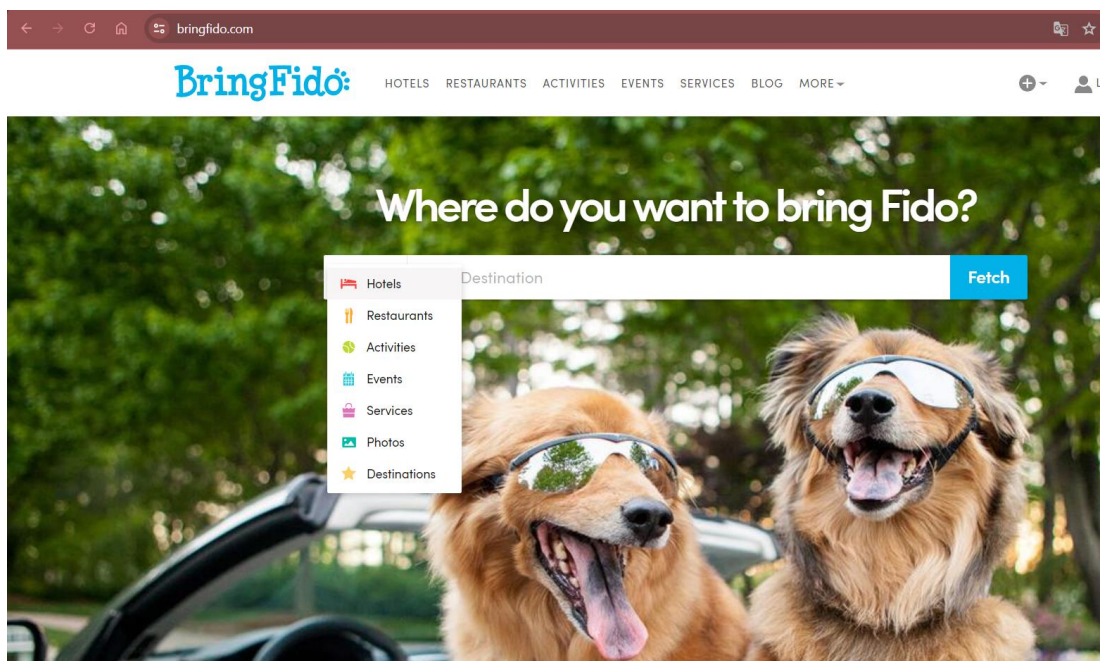


Рисунок 1.5 – Можливості застосунку BringFido

Розроблюваний застосунок включатиме функцію пошуку готелів і місць для перебування з тваринами, а також надаватиме широкий спектр інших можливостей, таких як пошук ветеринарів, грумерів, моніторинг здоров'я та соціальні функції.

Таблиця 1.1 – Порівняння існуючих застосунків

Функція / Застосунок	BringFido	11pets	Rover	Petfinder
Основна функціональність	Пошук готелів та інших місць, дружніх до тварин	Управління здоров'ям та доглядом за тваринами	Послуги догляду за домашніми тваринами	Пошук домашніх тварин для усиновлення
Цільова аудиторія	Подорожуючі з домашніми тваринами	Власники домашніх тварин	Власники домашніх тварин	Люди, що шукають домашніх тварин для усиновлення
Основні функції	Пошук готелів, дружніх до тварин; Пошук ресторанів, дружніх до тварин; Пошук парків та інших місць.	Відстеження здоров'я тварин; Планування годування, прийому ліків, щеплень; Історія лікування та медичних процедур; Нагадування про прийом ліків та вакцинації; Звіти про здоров'я тварини	Пошук доглядачів; Пошук тимчасових опікунів; Прогулянки з собаками; Догляд на дому.	Пошук домашніх тварин для усиновлення; Пошук притулків; Інформація про породи; Пошук за місцезнаходженням.
Платформи	iOS, Android	iOS, Android	iOS, Android, веб	iOS, Android, веб
Вартість	Безкоштовно	Безкоштовно з преміум функціями	Безкоштовно (з оплатою послуг доглядачів)	Безкоштовно
Особливості	Інтерактивні карти; Фото та опис місць.	Хмарне збереження даних; Мультипрофільність для декількох тварин.	Система відгуків;- Персоналізований профілі доглядачів.	Інтеграція з соціальними мережами; Велика база даних тварин.

Отже, хоча існує багато спеціалізованих застосунків для власників домашніх тварин, жоден з них не пропонує комплексного підходу, що поєднував би всі необхідні функції в одному місці. Розроблюваний застосунок буде універсальним рішенням, що задовольнить різноманітні потреби власників тварин, забезпечуючи зручність, ефективність та надійність у догляді за домашніми улюбленцями.

1.5 Збір та обробка інформації

Збір та обробка інформації є ключовими етапами розробки мобільного застосунку для інформаційної підтримки власників домашніх тварин. Використання сучасних технологій збору даних, їх обробки та аналізу дозволяє здійснювати досить вдалі рекомендації перед вибором, лікуванням та іншими аспектами утримання домашніх тварин.

1. Опитування та анкети. Створення анкет для збору даних про звички, потреби та проблеми, з якими стикаються власники тварин. Інтерв'ю з ветеринарами. проведення інтерв'ю з ветеринарами для отримання професійної інформації щодо догляду, профілактики та лікування тварин [14-17].

2. Аналіз існуючих джерел даних. Збирання даних з наукових публікацій, досліджень та звітів, що стосуються догляду за домашніми тваринами.

3. Дані від користувачів мобільного застосунку. Створення функціоналу, який дозволяє користувачам вводити інформацію про своїх тварин, їхній стан здоров'я, звички та потреби. Ці дані можуть бути використані для персоналізації рекомендацій та планів догляду. Впровадження системи зворотного зв'язку, яка дозволяє користувачам оцінювати корисність отриманої інформації та пропонувати вдосконалення.

Висновки до розділу 1

Було досліджено ключові аспекти розробки мобільного застосунку, що є основою для створення ефективного та корисного продукту. Перш за все, було розглянуто, що таке мобільний застосунок та особливості його створення.

Було розглянуто питання життєвого циклу додатків, який включає етапи від розробки до розгортання та підтримки. На кожному етапі життєвого циклу додатку, від планування до реалізації, тестування та підтримки, необхідно враховувати специфічні вимоги та умови, що впливають на кінцевий продукт.

Детально були розглянуті активності та фрагменти, які є основними будівельними блоками архітектури Android-додатків. Активності представляють собою окремі екрани інтерфейсу користувача, тоді як фрагменти дозволяють більш гнучко організовувати інтерфейс, розділяючи його на модульні та багаторазові компоненти.

Особливу увагу приділено процесу збору та обробки інформації, який є ключовим для функціонування застосунку для інформаційної підтримки власників домашніх тварин. Збір даних відбувається через опитування, інтерв'ю з ветеринарами, аналіз наукових статей та користувацькі введення. Обробка даних включає очищення, стандартизацію, аналіз за допомогою статистичних методів та машинного навчання, а також зберігання в реляційних та нереляційних базах даних та хмарних сховищах. Інтеграція даних через API та синхронізація забезпечують актуальність інформації, а візуалізація та система повідомлень покращують користувацький досвід.

Таким чином, дослідження предметної області забезпечило розуміння основних компонентів та процесів, необхідних для розробки мобільного застосунку, що надає якісну та своєчасну інформаційну підтримку власникам домашніх тварин.

Кафедра інтелектуальних інформаційних систем
Мобільний застосунок для інформаційної підтримки власників домашніх тварин

2 ТЕХНОЛОГІЇ ТА ЗАСОБИ РОЗРОБКИ МОБІЛЬНОГО ЗАСТОСУНКУ ДЛЯ ІНФОРМАЦІЙНОЇ ПІДТРИМКИ ВЛАСНИКІВ ДОМАШНІХ ТВАРИН

2.1 Важливість платформи Android для застосунку

Було обрано система Android над iOS з кількох важливих причин. Перш за все, Android займає значно більшу частку ринку мобільних операційних систем, що дозволяє охопити більшу аудиторію користувачів. Крім того, Android-пристрої представлені в ширшому діапазоні цінкових категорій, що робить їх доступнішими для більшої кількості людей. Це особливо важливо для розроблюваного застосунку, орієнтованого на широкий спектр користувачів. Також, Android надає більшу гнучкість у налаштуванні та персоналізації застосунків, що дозволяє створити більш адаптований та зручний інтерфейс для користувачів [18-19].

Таблиця 2.1 Порівняльна таблиця Android та iOS

Критерій	Android	iOS
Ринкова частка	Близько 72% світового ринку	Близько 27% світового ринку
Доступність	Широкий діапазон цінкових категорій	Переважно вищий цінковий сегмент
Кількість пристроїв	Велика різноманітність моделей	Обмежена кількість моделей (iPhone)
Гнучкість	Висока, можливість налаштування системи	Обмежена, строгий контроль Apple
Розробницьке середовище	Android Studio з підтримкою Java і Kotlin	Xcode з підтримкою Swift і Objective-C
Підтримка Google	Інтеграція з багатьма сервісами Google	Інтеграція з екосистемою Apple
Доступ до ринку	Google Play Store, менше обмежень	App Store, строгіші вимоги до застосунків

Кафедра інтелектуальних інформаційних систем
Мобільний застосунок для інформаційної підтримки власників домашніх тварин
Закінчення таблиці 2.1

Користувацька база	Більш різноманітна	Переважно користувачі високого сегменту
Розповсюдженість	Поширений у багатьох країнах, включаючи ринки, що розвиваються	Популярний у розвинених країнах

Ця порівняльна таблиця ілюструє ключові відмінності між Android та iOS, які були враховані при виборі платформи для розроблюваного застосунку. Враховуючи ці фактори, Android є оптимальним вибором для досягнення максимальної аудиторії та забезпечення гнучкості у розробці.

Платформа Android є однією з найпопулярніших операційних систем для мобільних пристроїв у світі. За даними Statista, станом на 2024 рік Android займає приблизно 72% ринку мобільних операційних систем. Це робить Android надзвичайно важливою платформою для розробки та впровадження будь-якого мобільного застосунку.

1. Доступність для широкої аудиторії. Розробка застосунку для Android дозволяє охопити велику кількість користувачів по всьому світу. Це особливо актуально для України, де багато людей користуються смартфонами на базі Android. Завдяки цьому, застосунок матиме значний потенціал для залучення великої аудиторії власників домашніх тварин.

2. Інтеграція з екосистемою Google. Android надає можливість легко інтегрувати застосунок з іншими сервісами та продуктами Google, такими як Google Maps, Google Calendar та Google Fit. Це дозволить додатково розширити функціональні можливості застосунку, наприклад, для пошуку ветклінік на мапі, створення нагадувань про візити до ветеринара або моніторингу активності тварини.

3. Гнучкість та налаштування. Android надає розробникам велику свободу у налаштуванні та персоналізації застосунків. Це означає, що можна створити унікальний інтерфейс, який буде зручним і інтуїтивно зрозумілим

Мобільний застосунок для інформаційної підтримки власників домашніх тварин для користувачів. Крім того, можливість підтримки різних версій Android дозволить забезпечити сумісність застосунку з широким спектром пристроїв, від бюджетних моделей до преміум-сегменту.

4. Переваги використання Android:

- З огляду на популярність Android, застосунок матиме потенціал охопити велику аудиторію як в Україні, так і за її межами;
- інтеграція з сервісами Google, що дозволить розширити функціональні можливості застосунку та зробити його більш корисним для користувачів;
- можливість створення унікального та зручного інтерфейсу, а також підтримка різних версій Android забезпечить зручність використання застосунку на різних пристроях;
- більшість смартфонів у середньому та бюджетному сегментах працюють на Android, що робить застосунок доступним для широкого кола користувачів.

Розробка застосунку для платформи Android є важливим кроком у створенні універсального сервісу для власників домашніх тварин, який допоможе вирішити багато щоденних проблем і значно полегшить життя як тваринам, так і їхнім власникам.

2.2 Інтегроване середовище розробки Android Studio

Android Studio є офіційним інтегрованим середовищем розробки (IDE) для створення застосунків на платформі Android. Це потужний інструмент, який надає розробникам всі необхідні засоби для ефективної та продуктивної роботи. Вибір Android Studio для розробки застосунку для тварин є обґрунтованим з наступних причин:

- офіційна підтримка Google. Android Studio розроблено та підтримується Google, що гарантує його постійну актуальність та

Мобільний застосунок для інформаційної підтримки власників домашніх тварин відповідність найновішим стандартам розробки для Android. Це означає, що розробники мають доступ до найновіших функцій та оновлень, які допоможуть створити сучасний та функціональний застосунок;

- інтеграція з інструментами розробки. Android Studio пропонує тісну інтеграцію з іншими інструментами розробки, такими як, Gradle для автоматизації процесів збірки та управління залежностями, Android SDK для доступу до всіх необхідних бібліотек та інструментів розробки, Emulator для тестування застосунків на різних версіях Android та типах пристроїв без необхідності фізичних девайсів;

- зручні інструменти для розробки UI. Android Studio надає розширені інструменти для створення інтерфейсу користувача (UI). Наприклад Layout Editor дозволяє візуально створювати та редагувати макети застосунку, використовуючи функцію перетягування елементів (drag-and-drop), Theme Editor допомагає налаштовувати вигляд застосунку, працюючи з темами та стилями, Real-Time Preview дозволяє миттєво бачити зміни у макеті на різних розмірах екранів та пристроях [20-21];

- розширені можливості налагодження та тестування. Android Studio пропонує потужні інструменти для налагодження та тестування, такі як, Logcat для перегляду логів та відстеження поведінки застосунку, Debugger для виявлення та виправлення помилок у коді, Profiler для аналізу продуктивності застосунку, виявлення проблем з продуктивністю, використанням пам'яті та іншими ресурсами;

- підтримка кількох мов програмування. Android Studio підтримує кілька мов програмування, включаючи Java та Kotlin. Kotlin є офіційною мовою для розробки Android-застосунків і пропонує сучасні можливості та синтаксичні переваги, що сприяє швидшій та зручнішій розробці.

Використання Android Studio для розробки застосунку для тварин є оптимальним вибором завдяки його офіційній підтримці Google, інтеграції з інструментами розробки, зручним засобам для створення UI, розширеним

Кафедра інтелектуальних інформаційних систем

Мобільний застосунок для інформаційної підтримки власників домашніх тварин можливостям налагодження та тестування, а також підтримці кількох мов програмування. Це середовище надає всі необхідні інструменти для створення високоякісного та функціонального застосунку, що допоможе ефективно вирішити проблеми власників домашніх тварин та забезпечить їх улюбленцям найкращий догляд.

2.3 Використання Vysor для розробки застосунків

Vysor – це інструмент, який дозволяє відображати та управляти вашим Android-пристроєм з комп'ютера. Це дуже корисний засіб для розробників, оскільки він забезпечує кілька важливих переваг під час розробки та тестування мобільних застосунків (рис. 2.1) [22-23].



Рисунок 2.1 – Інтерфейс застосунку Vysor

Переваги використання Vysor:

– Зручність налагодження і тестування. Vysor дозволяє розробникам швидко налагоджувати та тестувати застосунки на реальному пристрої, не відриваючись від свого робочого столу. Це значно спрощує

Мобільний застосунок для інформаційної підтримки власників домашніх тварин процес виявлення та виправлення помилок, а також перевірку працездатності застосунку на різних версіях Android;

- швидке відображення змін. Одна з ключових переваг Vysor – це можливість бачити зміни, внесені в код, в режимі реального часу на підключеному пристрої. Це дозволяє уникнути постійного перемикання між пристроями, що економить час і підвищує продуктивність;
- дистанційний контроль пристрою. Vysor забезпечує повний контроль над Android-пристроєм, дозволяючи виконувати всі дії, які доступні безпосередньо на телефоні або планшеті. Це особливо корисно для тестування жестів, багатозадачності та інших інтерактивних функцій;
- скріншоти та запис екрану. Інструмент надає можливість робити скріншоти та записувати відео з екрану пристрою. Це корисно для створення документації, навчальних матеріалів або демонстрації функціональності застосунку.

Як використовувати Vysor.

1. Встановлення Vysor. Для початку необхідно завантажити та встановити Vysor на комп'ютер. Vysor доступний для Windows, macOS та Linux.

2. Підключення пристрою. Підключіть ваш Android-пристрій до комп'ютера за допомогою USB-кабелю. Переконайтеся, що на пристрої увімкнено режим налагодження через USB.

3. Запуск Vysor. Запустіть Vysor на комп'ютері та натисніть кнопку "Find Devices". Виберіть пристрій зі списку доступних.

4. Відображення екрана. Після підключення екран Android-пристрою відобразиться у вікні Vysor, і можна буде почати взаємодію з ним.

Використання Vysor значно покращує ефективність процесу розробки та тестування мобільних застосунків. Завдяки можливості дистанційного контролю, швидкого відображення змін та зручного налагодження, розробники можуть економити час і ресурси, забезпечуючи високу якість

Кафедра інтелектуальних інформаційних систем
Мобільний застосунок для інформаційної підтримки власників домашніх тварин кінцевого продукту. Це робить Vysor незамінним інструментом для кожного, хто працює над створенням застосунків для Android.

2.4 Firebase як універсальна хмарна база

Firebase – це комплексна платформа для розробки мобільних і веб-застосунків, створена Google (рис. 2.2). Вона надає широкий спектр інструментів та послуг, які допомагають розробникам створювати високоякісні застосунки, покращувати їх продуктивність та забезпечувати ефективне управління даними. Використання Firebase значно спрощує багато аспектів розробки, від зберігання даних до автентифікації користувачів та аналітики [24-26].

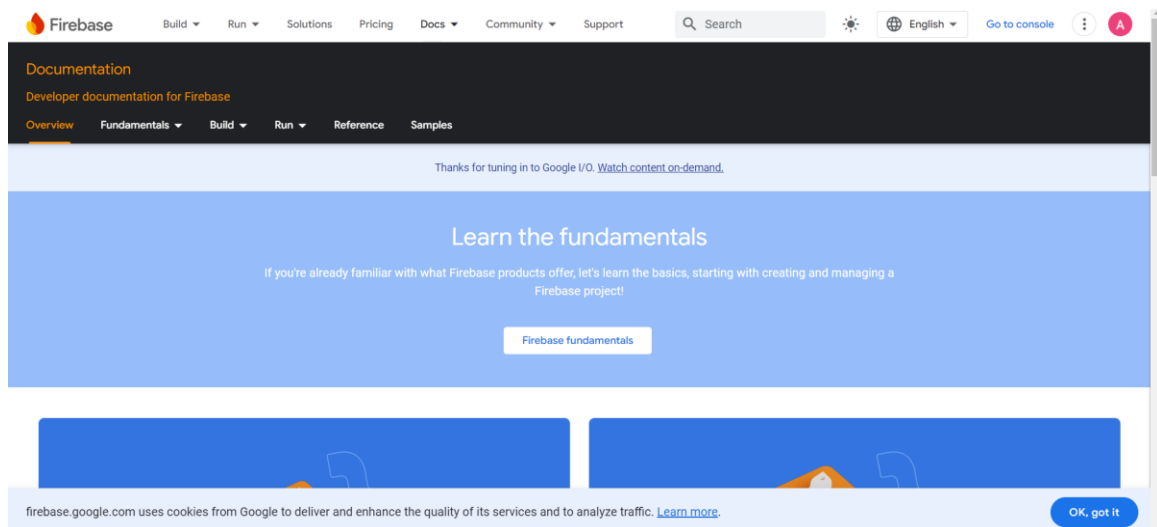


Рисунок 2.2 – Інтерфейс Firebase

Таблиця 2.2 Основні можливості Firebase

Основні можливості Firebase	Опис
Зберігання даних і бази даних в реальному часі	<p>Firebase пропонує Firestore та Realtime Database для зберігання та синхронізації даних у режимі реального часу. Це дозволяє розробникам створювати застосунки, які можуть миттєво реагувати на зміни в даних, забезпечуючи користувачам безперервний та інтерактивний досвід. Обидві бази даних забезпечують надійне та масштабоване зберігання даних, що підходить для застосунків будь-якого розміру.</p>
Аутентифікація користувачів	<p>Firebase Authentication дозволяє легко реалізувати аутентифікацію користувачів у застосунку за допомогою різних методів, включаючи електронну пошту та пароль, телефонні номери, а також сторонні провайдери, такі як Google, Facebook, Twitter та інші. Це забезпечує безпечний та зручний спосіб входу для користувачів, зменшуючи зусилля розробників на реалізацію цих функцій.</p>
Аналітика	<p>Firebase Analytics надає потужні інструменти для відстеження поведінки користувачів у застосунку. Це дозволяє розробникам збирати та аналізувати дані про використання застосунку, розуміти, як користувачі взаємодіють із застосунком, і приймати обґрунтовані рішення щодо покращення користувацького досвіду. Аналітика Firebase інтегрується з іншими сервісами Firebase, що дозволяє створювати персоналізовані та таргетовані кампанії.</p>
Хмарні функції	<p>Firebase Cloud Functions дозволяє виконувати серверний код у відповідь на події, що відбуваються у вашому застосунку або інших сервісах Firebase. Це допомагає автоматизувати багато процесів, наприклад, обробку даних, відправлення сповіщень або виконання складних бізнес-логік, без необхідності управління власними серверами.</p>
Сповіщення та обмін повідомленнями	<p>Firebase Cloud Messaging (FCM) дозволяє відправляти сповіщення та повідомлення на Android, iOS та веб-платформи. Це допомагає утримувати користувачів в курсі важливих подій, покращуючи взаємодію та залучення до застосунку.</p>
Хостинг	<p>Firebase Hosting надає безпечний, швидкий та надійний хостинг для веб-додатків та статичних веб-сайтів. Він забезпечує глобальну CDN, що робить ваші веб-застосунки доступними з будь-якого куточка світу з мінімальною затримкою.</p>

Firestore є потужною платформою для розробки мобільних і веб-застосунків, яка пропонує широкий спектр інструментів та сервісів для покращення продуктивності та управління даними. Вона значно спрощує процес розробки, дозволяючи розробникам зосередитися на створенні високоякісних застосунків, що забезпечують відмінний користувацький досвід. Використання Firestore може суттєво підвищити ефективність роботи команди розробників та допомогти швидко впроваджувати нові функції та покращення у вашому застосунку.

Висновки до розділу 2

Усі розглянуті технології та середовища розробки мають свої сильні та слабкі сторони. Вибір конкретного інструменту залежить від вимог проекту, цільової аудиторії та досвіду розробників.

За підсумками даного розділу можна сказати, що було проаналізовано технології розробки застосунків та на основі цих даних було обрано відповідні технології.

Кафедра інтелектуальних інформаційних систем
Мобільний застосунок для інформаційної підтримки власників домашніх тварин

РОЗДІЛ 3 ПРОГРАМНА РЕАЛІЗАЦІЯ ЗАСТОСУНКУ ДЛЯ ІНФОРМАЦІЙНОЇ ПІДТРИМКИ ВЛАСНИКІВ ДОМАШНІХ ТВАРИН

3.1 Додавання API на Google Maps

Інтеграція API Google Maps у розроблюваний застосунок або вебсайт може значно покращити його функціональність, дозволяючи користувачам взаємодіяти з картами, знаходити місця, прокладати маршрути та багато іншого (рис. 3.1) [27-28].

Розглянемо кроки, необхідні для додавання API Google Maps.

Для використання Google Maps API необхідно мати API-ключ. Цей ключ дозволяє Google ідентифікувати проєкт та надає доступ до його ресурсів.

Реєструємося на Google Cloud Platform та створюємо новий проєкт.

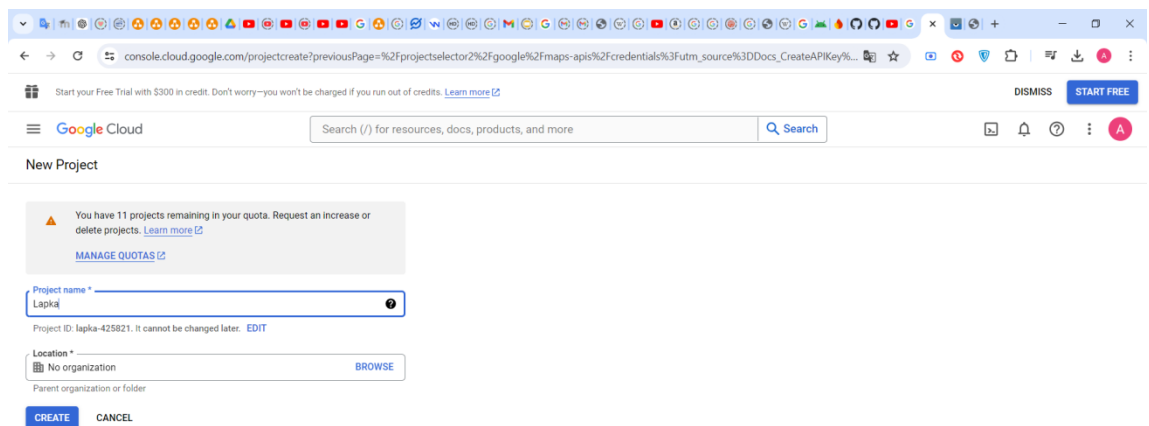


Рисунок 3.1 – Створення нового проєкту у Google Cloud

Натискаємо кнопку "Create" та активуємо Google Maps API, натиснувши у панелі навігації "API & Services" > "Library", знаходимо "Maps JavaScript API" і натискаємо "Enable". Після проведення всіх, вищеописаних дій переходимо до отримання ключа API, перейшовши до "API & Services" > "Credentials" та натиснувши "Create credentials" (рис. 3.2).

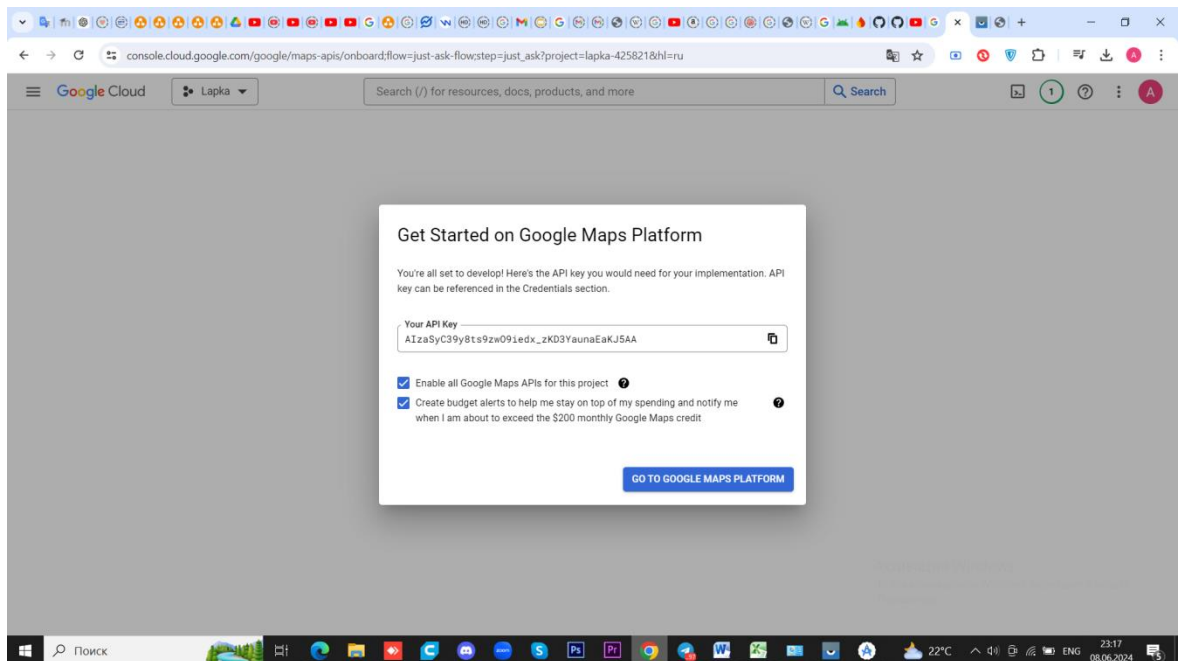


Рисунок 3.2 – Отриманий API-ключ

Зберігаємо отриманий ключ у безпечному місці, оскільки він знадобиться для подальших кроків.

3.2 Налаштування Vysor для симуляції розробленого застосунку

Після встановлення застосунку Vysor на персональний комп'ютер необхідно встановити спеціалізоване програмне забезпечення на смартфон з операційною системою Android для можливості тестування розроблюваного застосунку у реальному часі [29-30].

Для встановлення переходимо до Play Market та встановлюємо Vysor для Android (рис. 3.3).

22:03

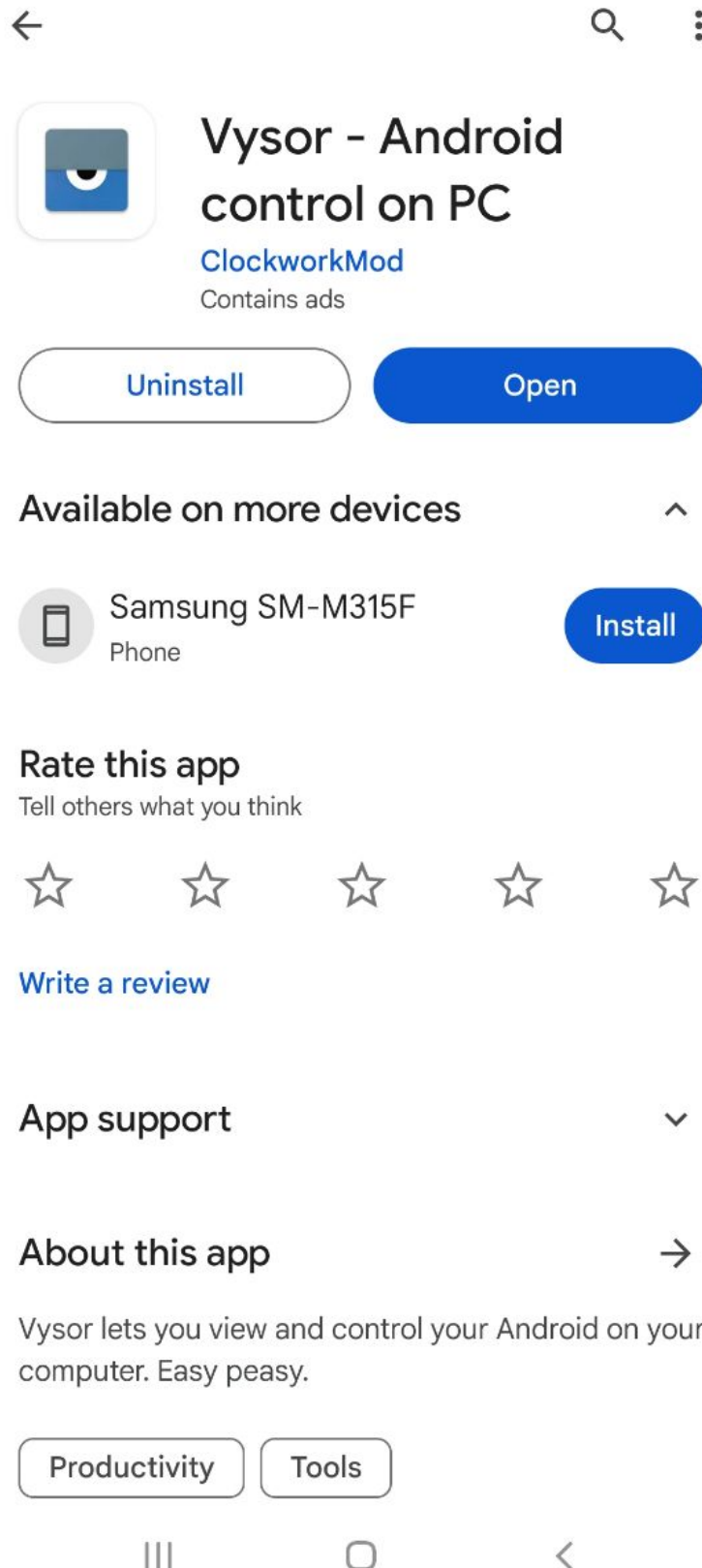


Рисунок 3.3 – Завантаження додатку Vysor для Android

Кафедра інтелектуальних інформаційних систем
Мобільний застосунок для інформаційної підтримки власників домашніх тварин
Проходимо процес встановлення (рис. 3.4).

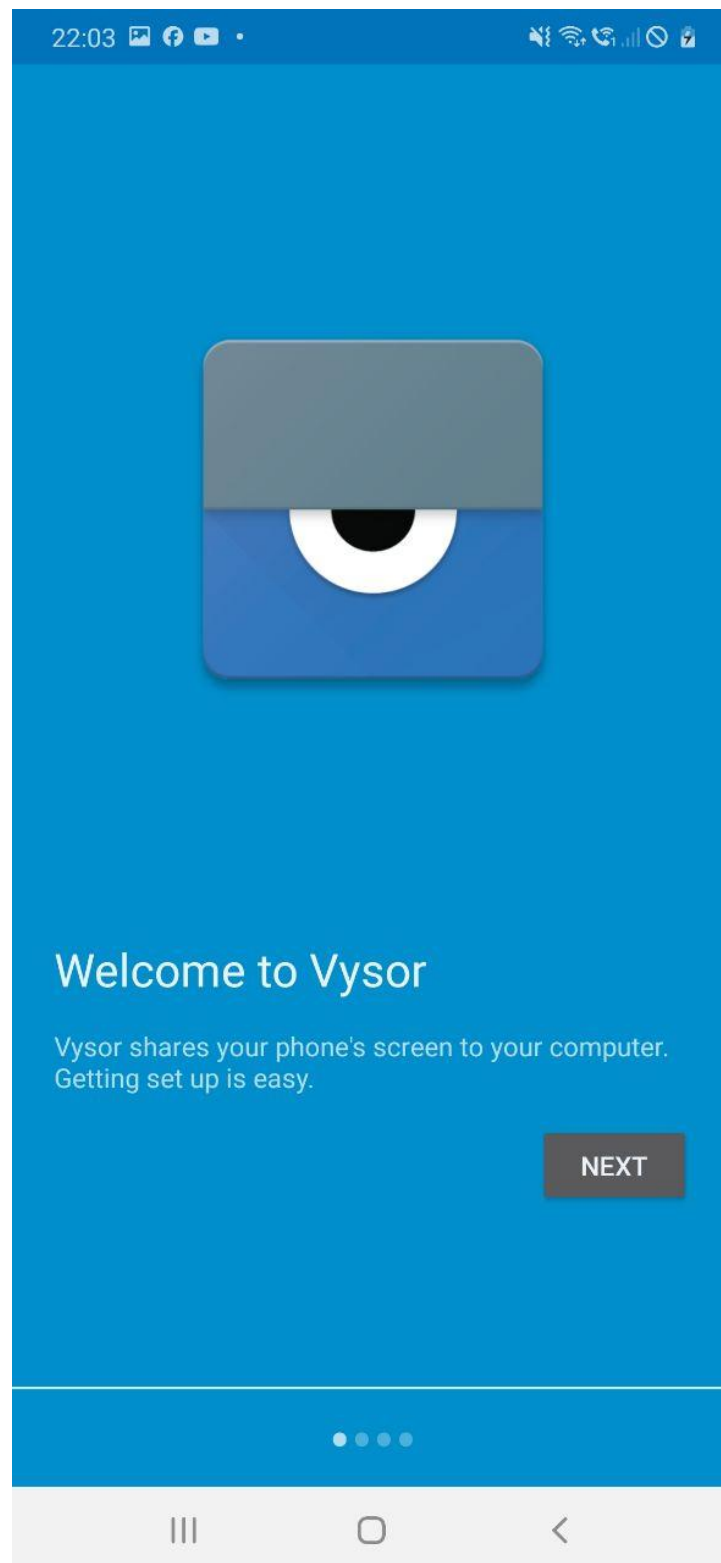


Рисунок 3.4 – Початок процесу встановлення застосунку

Кафедра інтелектуальних інформаційних систем
Мобільний застосунок для інформаційної підтримки власників домашніх тварин
Натискаємо кнопку “Send Download Link” для встановлення зв’язку з застосунком на ПК (рис. 3.5).

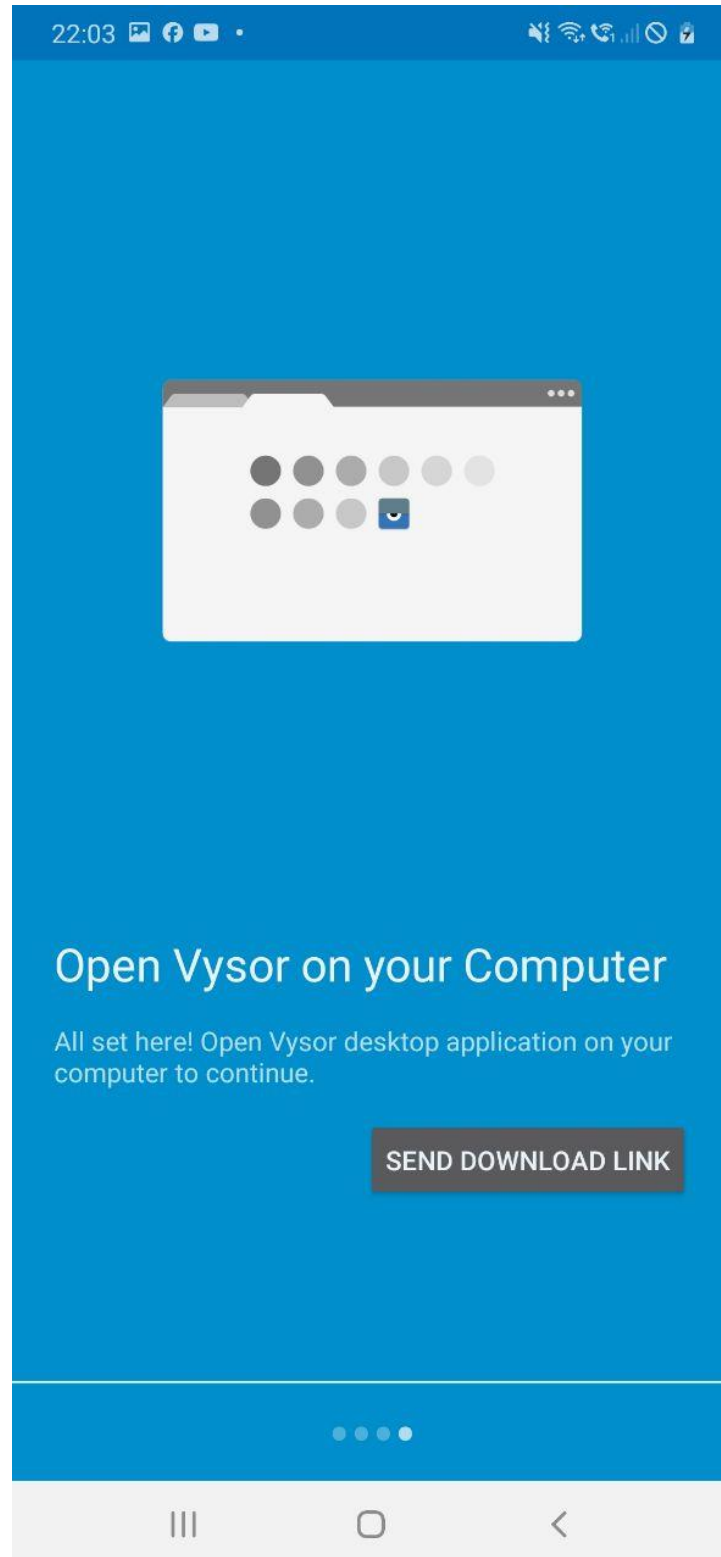


Рисунок 3.5 – Кнопка “Send Download Link”

Кафедра інтелектуальних інформаційних систем
Мобільний застосунок для інформаційної підтримки власників домашніх тварин
Переходимо у застосунок на ПК та натискаємо на кнопку запуску (рис. 3.6).

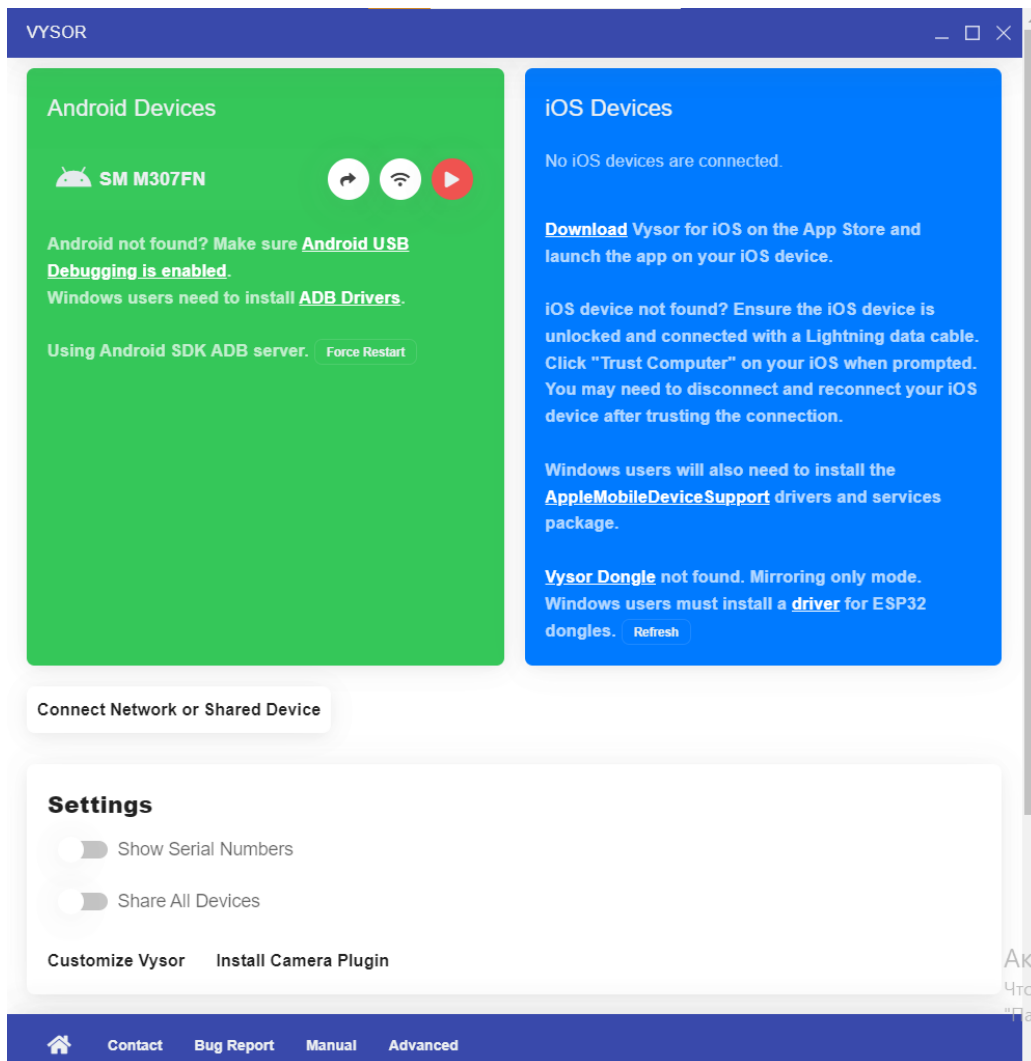


Рисунок 3.6 – Кнопка запуску у застосунку на ПК

Після цього повинно встановитися з'єднання зі смартфоном для майбутнього тестування (рис. 3.7 та рис. 3.8).

Кафедра інтелектуальних інформаційних систем
Мобільний застосунок для інформаційної підтримки власників домашніх тварин

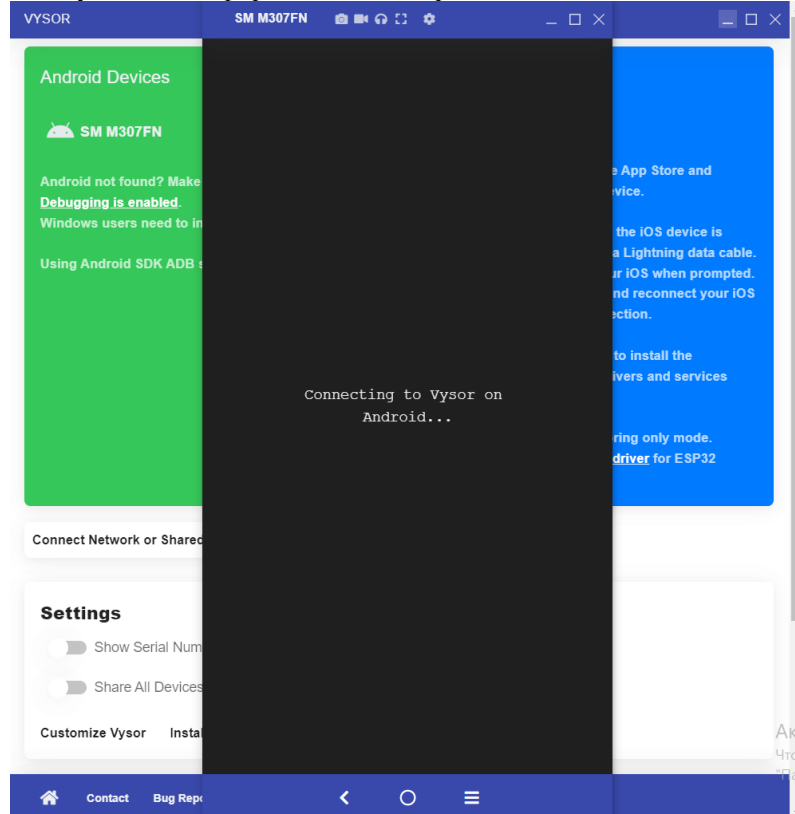


Рисунок 3.7 – Встановлення з'єднання

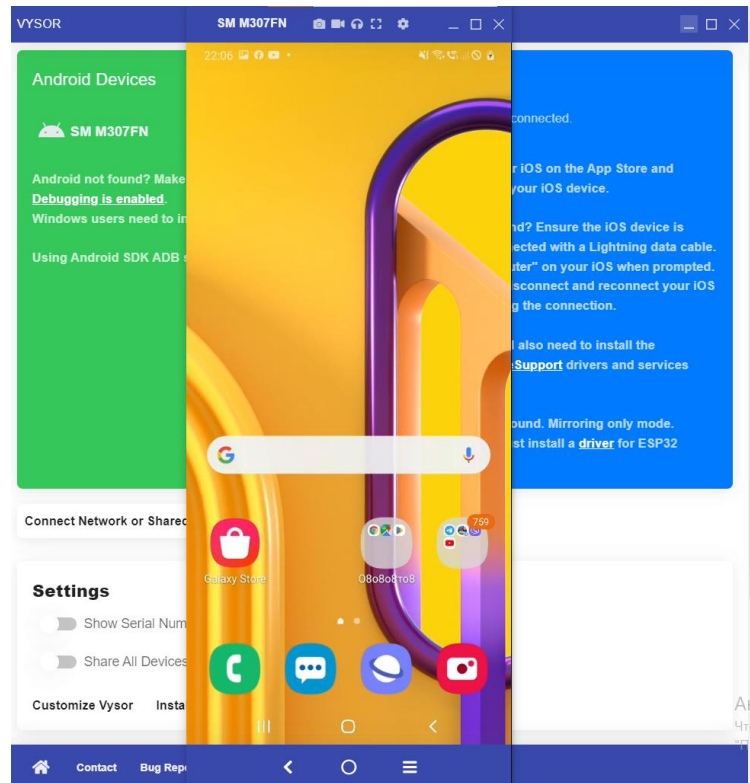


Рисунок 3.8 – Успішно встановлене з'єднання

Кафедра інтелектуальних інформаційних систем
Мобільний застосунок для інформаційної підтримки власників домашніх тварин
3.3 Створення сховища для проекту у Firebase

Спочатку необхідно створити копію проекту у хмарному сховищі (рис. 3.9).

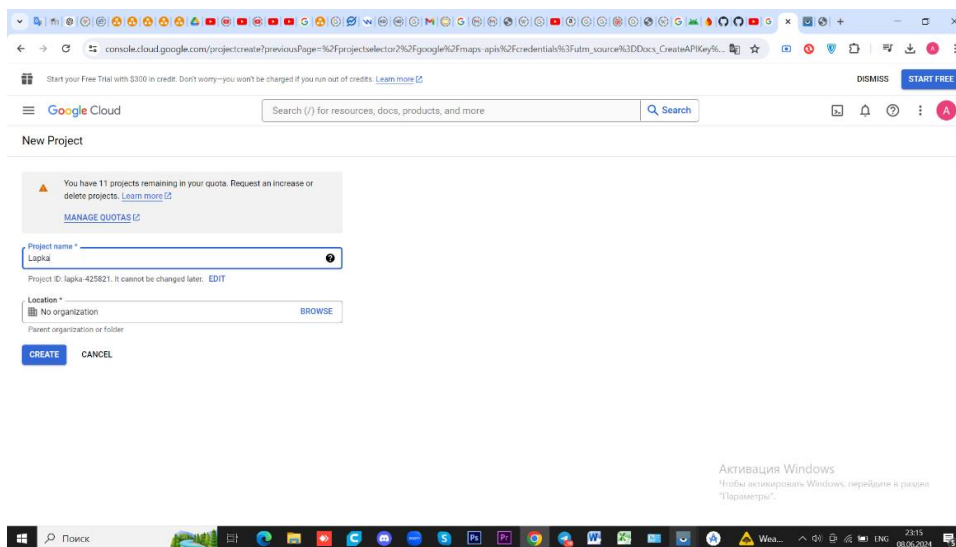


Рисунок 3.9 – Проект, створений у хмарному сховищі

Після успішного створення проекту у хмарному сховищі, додаємо цей проект у Firebase (рис. 3.10).

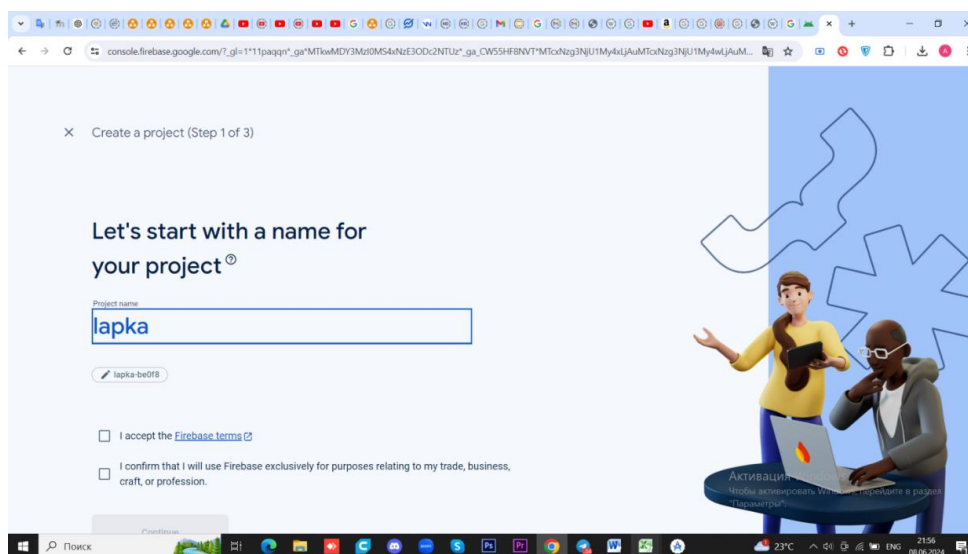


Рисунок 3.10 – Проект, доданий до Firebase

Кафедра інтелектуальних інформаційних систем
Мобільний застосунок для інформаційної підтримки власників домашніх тварин
Було успішно додано проект до Firebase з метою створення системи авторизації та збереження інформації [31-32].

3.4 Розробка функціоналу та інтерфейсу додатку

Додаємо список тварин, що зроблено за допомогою ListView (рис. 3.11-13).

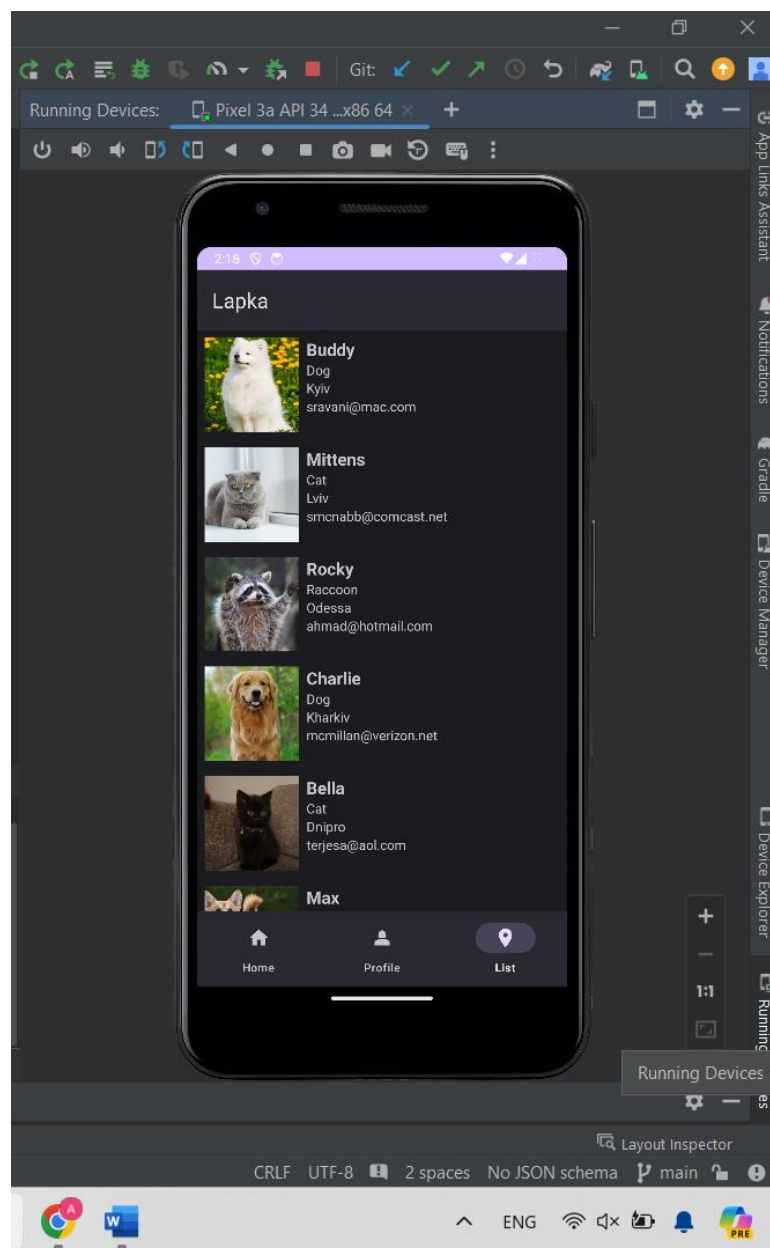


Рисунок 3.11 – Перше вікно доданих тварин

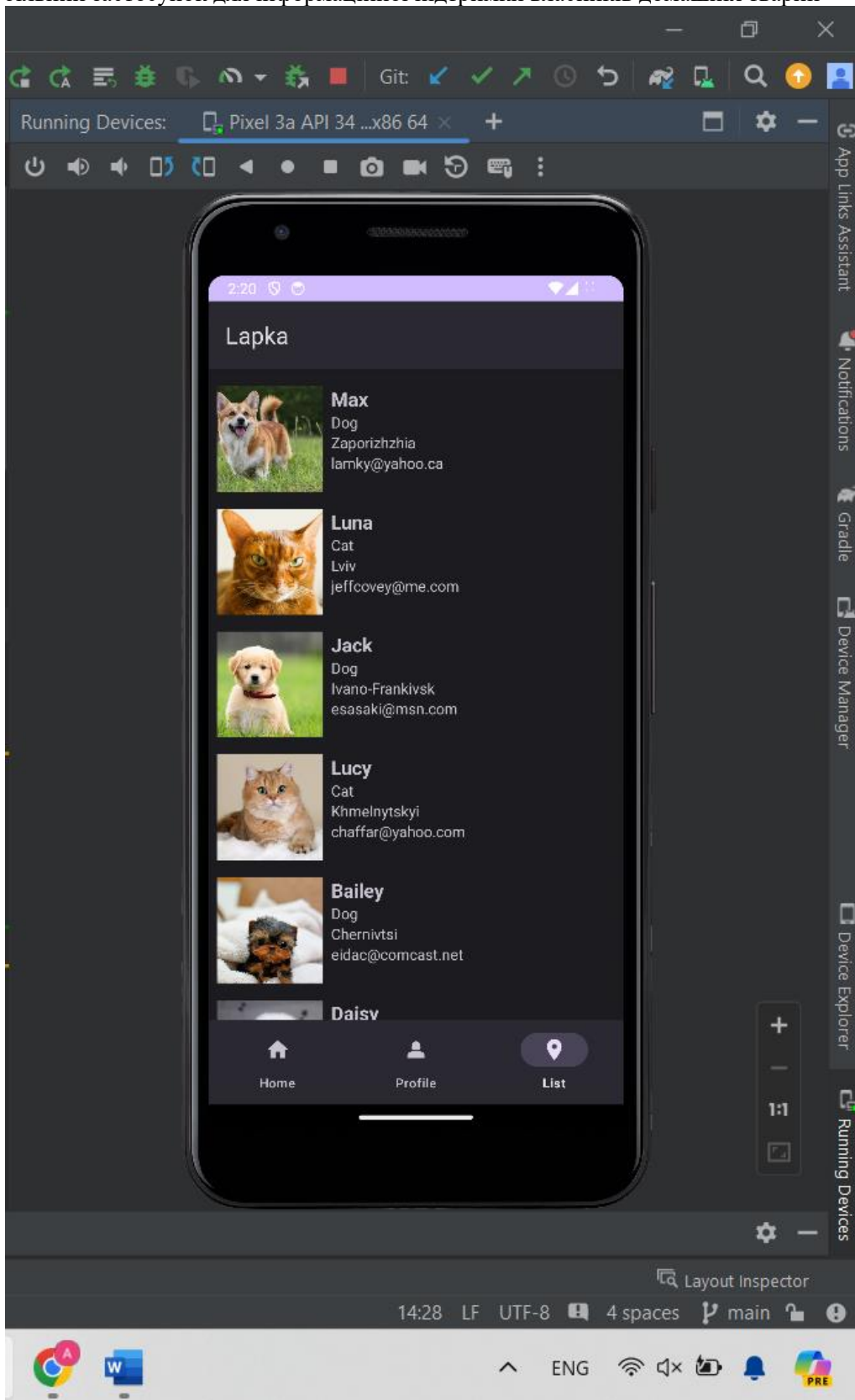


Рисунок 3.12 – Друге вікно доданих тварин

Кафедра інтелектуальних інформаційних систем
Мобільний застосунок для інформаційної підтримки власників домашніх тварин

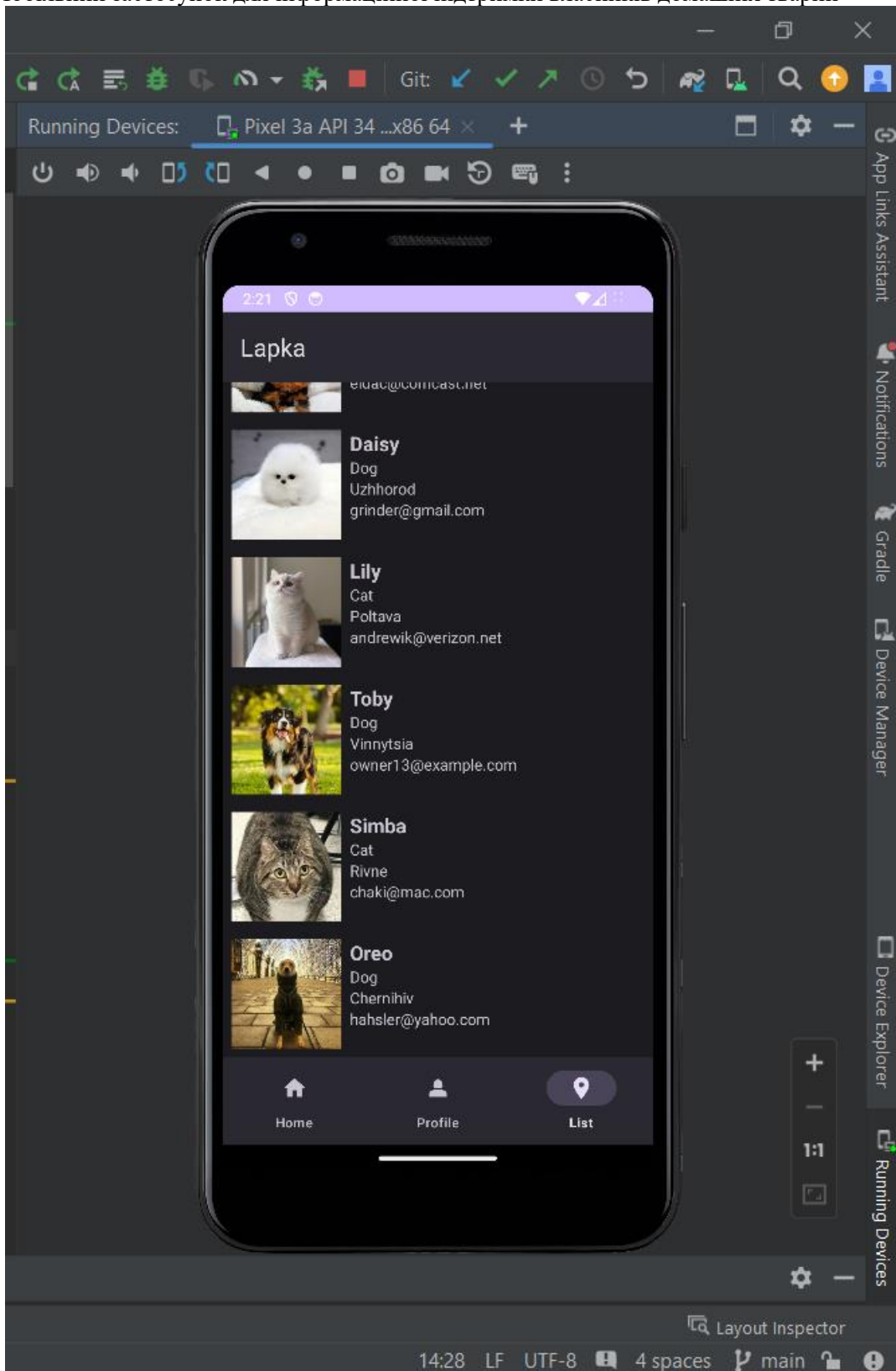


Рисунок 3.13 – Третє вікно доданих тварин

Розглянемо алгоритм роботи програми та основні функції, для ListView (рис. 3.14).

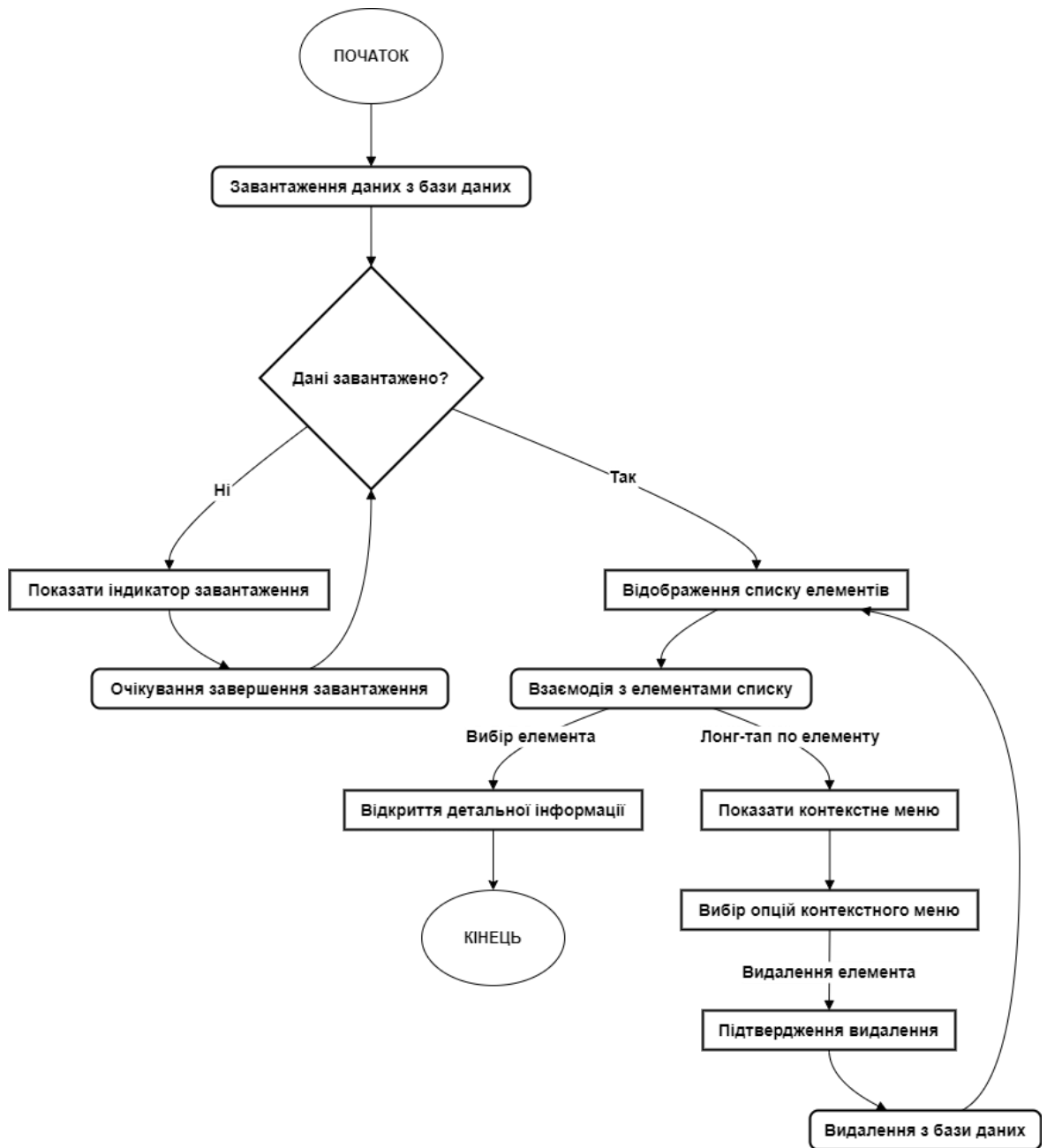


Рисунок 3.14 – Алгоритм роботи ListView

Алгоритм роботи ListFragment у додатку Android складається з наступних кроків:

Кафедра інтелектуальних інформаційних систем

Мобільний застосунок для інформаційної підтримки власників домашніх тварин

- починається все з ініціалізації, а саме Створення ListFragment та ініціалізації зв'язування макету (binding) [33-35];
- створення ієрархії представлень фрагмента у методі onCreateView. Ініціалізація RecyclerView та встановлення layoutManager у методі onCreateView. Ініціалізація RecyclerView та встановлення layoutManager у методі onCreateView [33-35];
- виклик методу loadPetsFromJson для завантаження даних з JSON-файлу. Читання JSON-даних з файлу активів у методі getJsonDataFromAsset. Перетворення JSON-даних у список об'єктів Pet за допомогою Gson. Додавання об'єктів Pet до petArrayList. Оновлення адаптера (myAdapter) [33-35] (рис. 3.16);
- показ детальної інформації про тварину у методі showPetDetail (рис. 3.15);
- видалення тварини зі списку у методі deletePet.

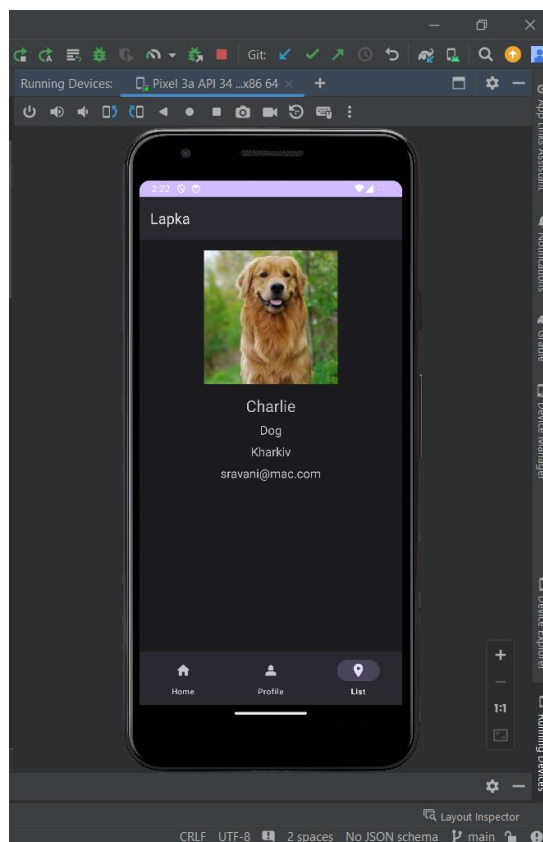


Рисунок 3.15 – Детальний перегляд методі showPetDetail

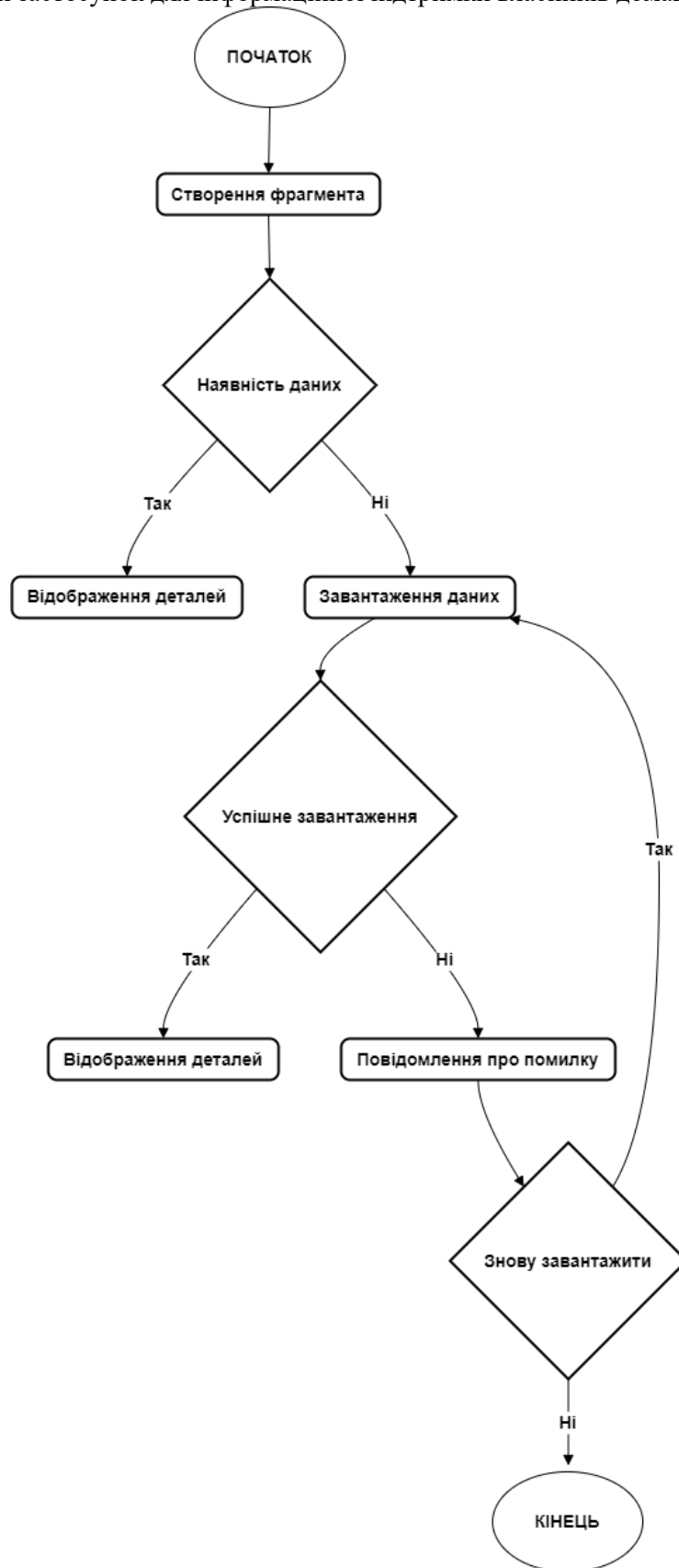


Рисунок 3.16 – Алгоритм роботи showPetDetail

Кафедра інтелектуальних інформаційних систем
Мобільний застосунок для інформаційної підтримки власників домашніх тварин
Дивлячись на рисунок 3.17 пояснимо алгоритм роботи методу

`showPetDetail`:

- створення нового екземпляра `PetDetailFragment` з об'єктом `Pet` у вигляді аргументу через метод `newInstance` [33-35];
- ініціалізація представлень. Виклик методу `onCreateView` для створення ієрархії представлень фрагмента, використовуючи прив'язування макету (`binding`) [33-35];
- отримання даних. Виклик методу `onViewCreated` після створення представлень. Отримання об'єкта `Pet` з аргументів фрагмента [33-35];
- відображення даних. Перевірка, чи об'єкт `Pet` не є `null`.

Заповнення текстових полів (`tvPetName`, `tvPetType`, `tvPetCity`, `tvOwnerContact`) даними з об'єкта `Pet`. Використання `Glide` для завантаження зображення тварини в `ImageView` (`ivPetPhoto`) [33-35].

Додаємо API на Google Maps (рис. 3.18 та рис. 3.19).

Кафедра інтелектуальних інформаційних систем
Мобільний застосунок для інформаційної підтримки власників домашніх тварин

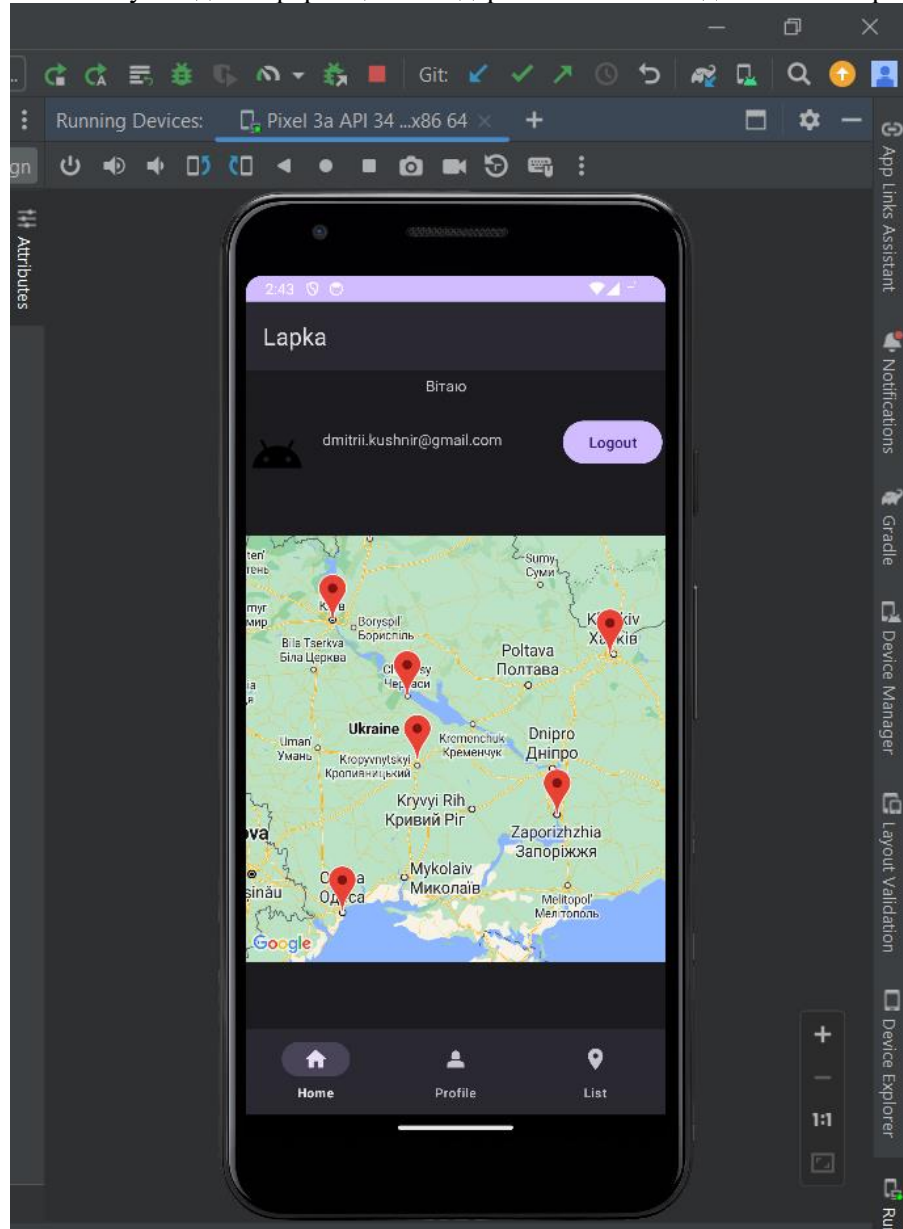


Рисунок 3.18 – Головний екран з локацією ветеринарних клінік

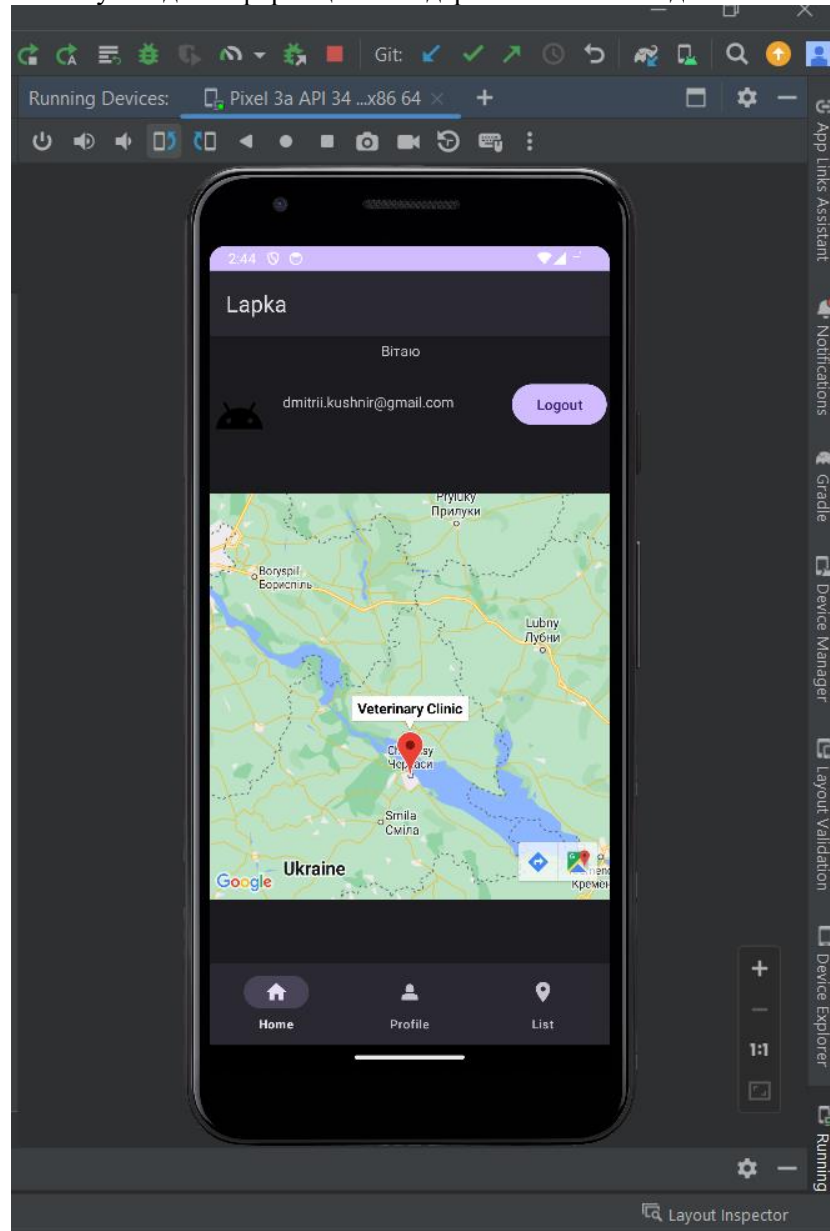


Рисунок 3.19 – Перевірка коректності роботи API на Google Maps

Як видно з рисунку 3.19 – додані API на Google Maps є інтерактивними, тож працюють коректно.

Для наповнення профілю та взагалі роботи додатку необхідно додати можливість дозволу застосунку під'єднуватися до інтернету, зчитувати дані про місцезнаходження та можливість експорту та імпорту необхідних файлів, для цього потрібно додати наступні дозволи:

- `<uses-permission android:name="android.permission.INTERNET"/>`.

Дозвіл на доступ до інтернету. Це дозволяє додатку використовувати

Мобільний застосунок для інформаційної підтримки власників домашніх тварин мережеві з'єднання, включаючи відправку та отримання даних через інтернет;

– `<uses-permission`

`android:name="android.permission.ACCESS_FINE_LOCATION"/`. Дозвіл на доступ до точного місцезнаходження пристрою. Використовується для отримання координат з високою точністю, зазвичай з GPS;

– `<uses-permission`

`android:name="android.permission.READ_EXTERNAL_STORAGE"/>`. Дозвіл на читання даних із зовнішнього сховища. Дозволяє додатку читати файли, збережені на зовнішніх накопичувачах, таких як SD-карти;

– `<uses-permission`

`android:name="android.permission.WRITE_EXTERNAL_STORAGE"/>`. Дозвіл на запис даних до зовнішнього сховища. Дозволяє додатку зберігати файли на зовнішніх накопичувачах, таких як SD-карти.

Дані дозволи необхідні для забезпечення відповідної функціональності додатку, пов'язаної з мережею, геолокацією та файловою системою.

Після надання усіх дозволів, можна наповнювати функціоналом профіль користувача (рис. 3.20-23).

Кафедра інтелектуальних інформаційних систем
Мобільний застосунок для інформаційної підтримки власників домашніх тварин

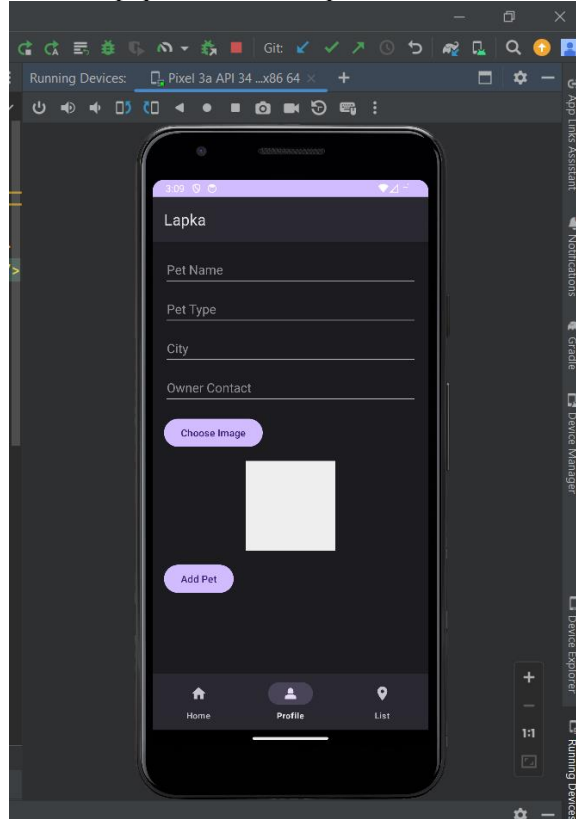


Рисунок 3.20 – Меню додавання домашнього улюбленця, що знаходиться в профілі користувача

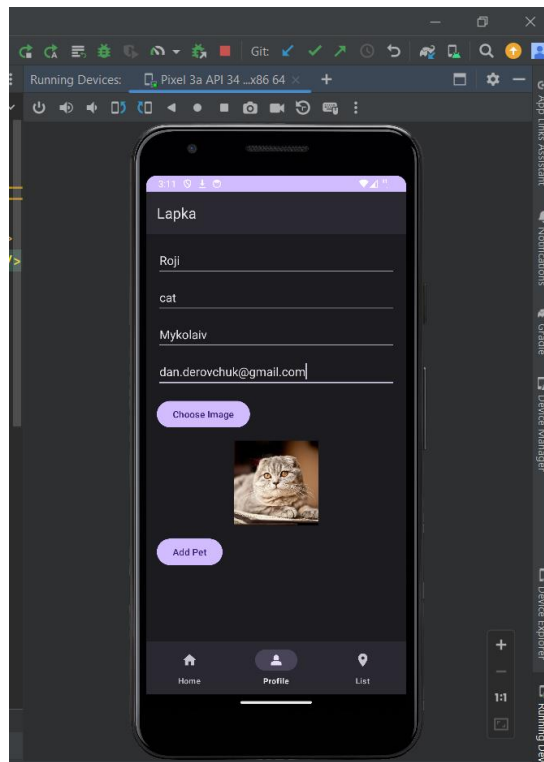


Рисунок 3.21 – Заповнене меню

Кафедра інтелектуальних інформаційних систем
Мобільний застосунок для інформаційної підтримки власників домашніх тварин

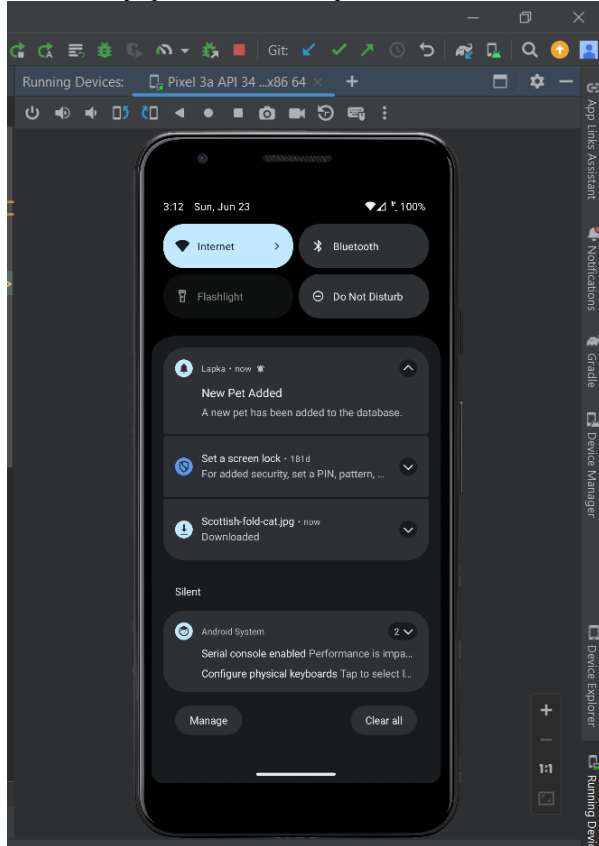


Рисунок 3.22 – Сповіщення про успішність додавання тварини

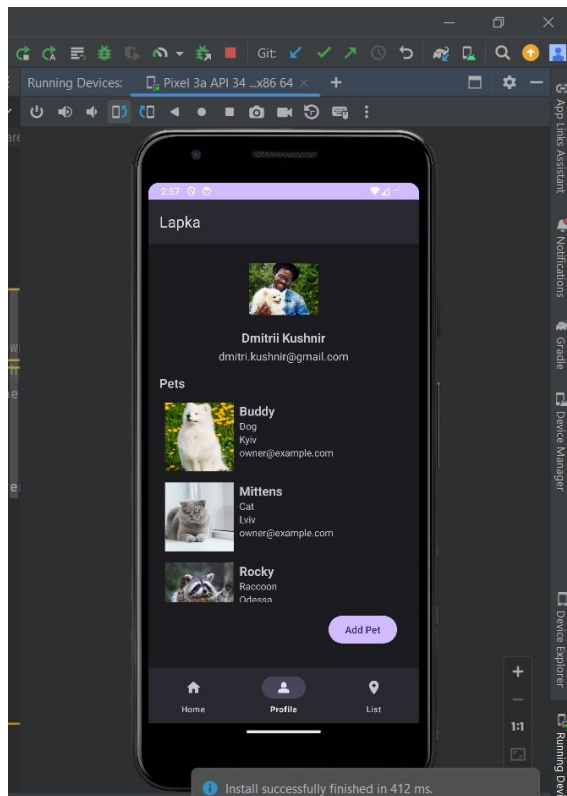


Рисунок 3.23 – Зовнішній вигляд профілю користувача

Після додання основних функцій до застосунку необхідно пояснити яким чином здійснена організація та управління текстовими ресурсами у додатку, а саме призначення файлу strings.xml (рис. 3.24).

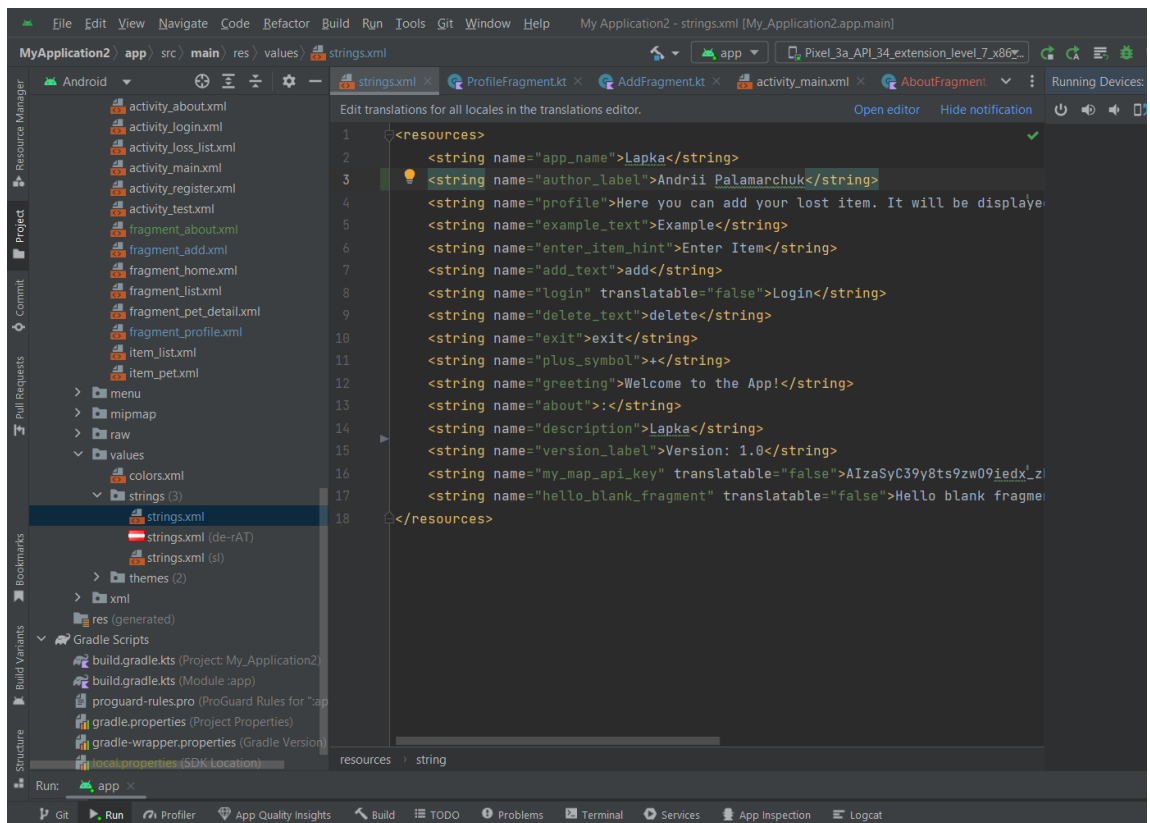


Рисунок 3.24 – Файл strings.xml

Зберігання текстових рядків у окремому файлі дозволяє легко перекладати додаток на інші мови. Для цього можна створювати окремі файли strings.xml для кожної мови у відповідних підкаталогах (наприклад, res/values-fr/strings.xml для французької мови).

Всі текстові ресурси зберігаються в одному місці, що полегшує їхнє управління та редагування. Це також знижує ймовірність виникнення помилок при використанні текстових рядків у коді.

Використання strings.xml дозволяє відокремити текстові дані від логіки додатку, що робить код більш чистим та зрозумілим.

Висновки до розділу 3

В ході виконання розділу 3 було реалізовано програму, що складається з кількох основних компонентів, які реалізовані у вигляді окремих фрагментів та активностей. Основна мета програми - надати користувачам можливість додавати, переглядати та аналізувати дані про домашніх тварин

Вона надає користувачам можливість додавати, переглядати та управляти інформацією про своїх домашніх улюбленців. Додаток також включає функціональність відображення ветеринарних клінік на карті та профілю користувача.

ВИСНОВКИ

В ході виконання даної роботи було проаналізовано патентну та теоретичну інформацію в сфері розробки додатків, а саме розробки додатків для власників домашніх тварин.

Обґрунтовано доцільність та актуальність виконання даної задачі та розглянуто технології завдяки яким можливо створити мобільний застосунок та наповнити його функціональністю.

В заключному розділі розглянуто програмну реалізацію застосунку для інформаційної підтримки власників домашніх тварин та її основні функціональні частини.

Застосунок, що було розроблено має багато різних функцій, таких як інтерактивні карти, на яких відображаються ветеринарні клініки, можливість створення профілю користувача та відслідковування місцезнаходження за допомогою GPS.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Петров, І. В. Розробка мобільних застосунків. Київ, 2020, 150 с.
2. Іваненко, А. В. Основи програмування на Java для Android. Харків, 2018, 200 с.
3. Johnson, K. Mobile Application Development. London, 2019, 250 с.
4. Сидоренко, Л. М. Мобільні технології та їх застосування. Київ, 2017, 180 с.
5. Smith, J. Android Development Basics. New York, 2020, 220 с.
6. Гнатюк, О. С. Основи розробки мобільних застосунків. Львів, 2019, 160 с.
7. Бойко, М. В. Мобільні додатки для початківців. Одеса, 2021, 140 с.
8. Campbell, M. Mastering Android Fragments. Boston, 2018, 300 с.
9. Коваленко, Ю. Г. Модульне програмування в Android. Дніпро, 2018, 190 с.
10. Лисенко, В. П. Проектування мобільних інтерфейсів. Полтава, 2019, 210 с.
11. Brown, L. Pet Care Apps: A Comparative Study. Chicago, 2020, 230 с.
12. Гриценко, Н. В. Аналіз ринку мобільних застосунків. Київ, 2019, 175 с.
13. Савченко, І. О. Інформаційні системи для підтримки тварин. Вінниця, 2020, 200 с.
14. Miller, A. Data Collection and Analysis for Apps. San Francisco, 2018, 270 с.
15. Василенко, П. Г. Методи збору інформації для мобільних застосунків. Київ, 2020, 165 с.

16. Федоренко, Л. М. Аналіз даних у мобільних застосунках. Харків, 2019, 185 с.
17. Thompson, R. Mobile App Development Insights. Seattle, 2019, 240 с.
18. Wilson, P. The Android Ecosystem. Berlin, 2020, 260 с.
19. Андрієнко, В. В. Платформа Android: можливості та перспективи. Київ, 2019, 195 с.
20. Harris, S. Effective Use of Android Studio. Boston, 2019, 280 с.
21. Пономаренко, О. Д. Використання Android Studio для розробки. Львів, 2020, 170 с.
22. Вишневецький, М. С. Інструменти для тестування мобільних додатків. Харків, 2020, 155 с.
23. Freeman, T. Testing with Vysor. San Francisco, 2019, 210 с.
24. Nelson, K. Firebase for Mobile Apps. New York, 2018, 230 с.
25. Шевченко, Д. В. Використання Firebase у мобільних застосунках. Київ, 2020, 200 с.;
26. Франчук, А. М. Хмарні технології в мобільних додатках. Одеса, 2020, 180 с.
27. Wilson, J. Google Maps API Integration. London, 2019, 220 с.
28. Величко, С. П. Робота з API в Android. Київ, 2019, 175 с.
29. Freeman, T. Advanced Vysor Configuration. San Francisco, 2019, 190 с.
30. Головка, О. І. Конфігурація та налаштування Vysor. Львів, 2020, 165 с.
31. Nelson, K. Advanced Firebase Techniques. New York, 2020, 250 с.
32. Шаповалов, В. Б. Організація баз даних у Firebase. Київ, 2020, 180 с.
33. Mitchell, R. Designing User Interfaces for Mobile Apps. Boston, 2018, 240 с.

Кафедра інтелектуальних інформаційних систем

Мобільний застосунок для інформаційної підтримки власників домашніх тварин

34. Черняк, Ю. І. Розробка інтерфейсів для мобільних додатків. Київ, 2020, 170 с.
35. Кравчук, М. О. Завершення проекту: аналіз та висновки. Харків, 2020, 190 с.

ДОДАТОК А***Лістинг коду для ListView:***

```
package com.example.myapplication2

import android.content.Context
import android.os.Bundle
import android.util.Log
import android.view.LayoutInflater
import android.view.View
import android.view.ViewGroup
import androidx.fragment.app.Fragment
import androidx.fragment.app.FragmentTransaction
import androidx.recyclerview.widget.LinearLayoutManager
import androidx.recyclerview.widget.RecyclerView
import com.example.myapplication2.databinding.FragmentListBinding
import com.google.gson.Gson
import com.google.gson.reflect.TypeToken
import java.io.IOException

class ListFragment : Fragment() {

    private lateinit var binding: FragmentListBinding
    private lateinit var petRecyclerview: RecyclerView
    private lateinit var petArrayList: ArrayList<Pet>
    private lateinit var myAdapter: MyAdapter
```

```
override fun onCreateView(  
    inflater: LayoutInflater, container: ViewGroup?,  
    savedInstanceState: Bundle?  
) : View {  
    binding = FragmentListBinding.inflate(inflater, container, false)  
    return binding.root  
}  
  
override fun onViewCreated(view: View, savedInstanceState: Bundle?) {  
    super.onViewCreated(view, savedInstanceState)  
  
    petRecyclerview = binding.recyclerView  
    petRecyclerview.layoutManager = LinearLayoutManager(requireContext())  
    petRecyclerview.setHasFixedSize(true)  
  
    petArrayList = arrayListOf()  
    myAdapter = MyAdapter(petArrayList, object :  
MyAdapter.OnItemClickListener {  
        override fun onItemClick(position: Int) {  
            showPetDetail(petArrayList[position])  
        }  
  
        override fun onItemLongClick(position: Int) {  
            deletePet(position)  
        }  
    })  
    petRecyclerview.adapter = myAdapter
```

```
// Load data from JSON file
```

```
loadPetsFromJson()
```

```
}
```

```
private fun getJsonDataFromAsset(context: Context, fileName: String): String?
```

```
{
```

```
    return try {
```

```
        val inputStream = context.assets.open(fileName)
```

```
        val size = inputStream.available()
```

```
        val buffer = ByteArray(size)
```

```
        inputStream.read(buffer)
```

```
        inputStream.close()
```

```
        String(buffer, Charsets.UTF_8)
```

```
    } catch (ex: IOException) {
```

```
        Log.e("ListFragment", "Error reading JSON file", ex)
```

```
        null
```

```
    }
```

```
}
```

```
private fun loadPetsFromJson() {
```

```
    val jsonString = getJsonDataFromAsset(requireContext(), "pets.json")
```

```
    if (jsonString != null) {
```

```
        val gson = Gson()
```

```
        val listPetType = object : TypeToken<List<Pet>>() {}.type
```

```
        val pets: List<Pet> = gson.fromJson(jsonString, listPetType)
```

```
        petArrayList.addAll(pets)
```

```
        myAdapter.notifyDataSetChanged()
```

```
    }
```

}

```
private fun deletePet(position: Int) {  
    petArrayList.removeAt(position)  
    myAdapter.notifyItemRemoved(position)  
}
```

```
private fun showPetDetail(pet: Pet) {  
    val petDetailFragment = PetDetailFragment.newInstance(pet)  
    val transaction: FragmentTransaction =  
parentFragmentManager.beginTransaction()  
    transaction.replace(R.id.frame_layout, petDetailFragment)  
    transaction.addToBackStack(null)  
    transaction.commit()  
}  
}
```