

НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ імені ІГОРЯ СІКОРСЬКОГО»
(КПІ ім. ІГОРЯ СІКОРСЬКОГО)

ФАКУЛЬТЕТ БІОМЕДИЧНОЇ ІНЖЕНЕРІЇ
КАФЕДРА БІОМЕДИЧНОЇ КІБЕРНЕТИКИ

«До захисту допущено»
В.о. завідувача кафедри БМК

_____ Євген НАСТЕНКО

“ ___ ” _____ 2023р.

Дипломна робота
на здобуття ступеня бакалавра
за освітньо-професійною програмою
«Комп'ютерні технології в біології та медицині»
спеціальності 122 Комп'ютерні науки

на тему: Інформаційна система для онлайн бронювання медичного
обладнання

Виконав: студент IV курсу, групи БС-92

УСАЧ АНДРІЙ ОЛЕКСАНДРОВИЧ

(прізвище, ім'я, по батькові)

(підпис)

Керівник: ас. каф. біомедичної кібернетики (БМК)

Давидько Олександр Богданович

(посада, науковий ступінь, вчене звання, прізвище, ім'я, по батькові)

(підпис)

Рецензент: доцент каф. біомедичної інженерії, доцент, к.т.н.

Білошицька Оксана Костянтинівна

(посада, науковий ступінь, вчене звання, прізвище, ім'я, по батькові)

(підпис)

Засвідчую, що у цій дипломній роботі
немає запозичень з праць інших
авторів без відповідних посилань.

Студент

(підпис)

Київ – 2023 року

**Національний технічний університет України
«Київський політехнічний інститут імені Ігоря Сікорського»**

Факультет біомедичної інженерії

Кафедра біомедичної кібернетики

Рівень вищої освіти – перший (бакалаврський)

Спеціальність – 122 «Комп'ютерні науки»

Освітньо-професійна програма «Комп'ютерні технології в біології та медицині»

ЗАТВЕРДЖУЮ

В.о. завідувача кафедри БМК

_____ Євген НАСТЕНКО

«30» травня 2023 р.

ЗАВДАННЯ

на дипломну роботу студенту

УСАЧУ АНДРІЮ ОЛЕКСАНДРОВИЧУ

(прізвище, ім'я, по батькові)

1. Тема роботи

**Інформаційна система для онлайн бронювання
медичного обладнання**

Керівник роботи

Давидько Олександр Богданович

(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)

затверджені наказом по університету від «31» травня 2023 р. №2106-с

2. Термін подання студентом роботи:

08 червня 2023 року

3. Вихідні дані до роботи: Android-додаток для онлайн бронювання медичного обладнання

4. Зміст роботи: анотації (на двох мовах); перелік скорочень, умовних позначень, термінів; вступ; аналітичний огляд літературних джерел; теоретична частина, практична частина; загальні висновки; список використаних джерел.

5. Перелік ілюстративного матеріалу (із зазначенням плакатів, презентацій тощо):
18 слайдів презентації, **18** рисунків, **0** таблиць.

6. Дата видачі завдання **30 травня 2023 р.**

Календарний план

№ з/п	Назва етапів виконання дипломної роботи	Термін виконання етапів роботи	Примітка
1	Отримати завдання за темою ДР на практику	До 15.02.2023р.	<i>виконано</i>
2	Переддипломна практика	За графіком	<i>виконано</i>
3	Виконання розділів ДР (вступ, огляд спеціальної літератури, матеріали та методи)	До кінця практики	<i>виконано</i>
4	Виконання розділів ДР (результат і обговорення дослідження, загальні висновки, список використаних джерел)	Не пізніше 1 тижня до засідання кафедри	<i>виконано</i>
5	Перевірка ДР науковим керівником	Не пізніше 1 тижня до засідання кафедри	<i>виконано</i>
6	Подання в електронному вигляді ДР та анотації до неї на перевірку нормоконтролера та плагіат (UNICHECK).		<i>виконано</i>
7	Надання документів на засідання кафедри	За день до засідання	<i>виконано</i>
8	Передзахист ДР та допуск до захисту дисертації	Згідно плану кафедри	<i>виконано</i>
9	Подання ДР рецензенту. Отримання рецензії.	До подання пакету документів до ЕК	<i>виконано</i>
10	Подання пакету документів по ДР та супровідних до неї документів до захисту в ЕК ¹	За 5 днів до дати захисту ДР за графіком	<i>виконано</i>
11	Захист ДР в ЕК		

Студент

(підпис)

Андрій УСАЧ

(ім'я, ПРІЗВИЩЕ)

Керівник ДР

(підпис)

Олександр ДАВИДЬКО

(ім'я, ПРІЗВИЩЕ)

Нормоконтролер

(підпис)

Галина КОРНІЄНКО

(ім'я, ПРІЗВИЩЕ)

АНОТАЦІЯ

Виконано дипломну роботу під назвою «Інформаційна система для онлайн бронювання медичного обладнання», автором якої є *Усач Андрій Олександрович*, студент кафедри біомедичної кібернетики ФБМІ, за спеціальністю 122 *Комп'ютерні науки*. Робота виконана в рамках освітньо-професійної програми «*Комп'ютерні технології в біології та медицині*». Структура дипломної роботи містить наступні елементи: перелік скорочень, умовних позначень, термінів; вступ; 3 розділи (аналітичний огляд літератури, теоретична частина, практична частина), а також висновки до кожного з цих розділів; загальні висновки; список використаних джерел, який налічує 26 пунктів англomовної літератури. Загальний обсяг роботи 71 сторінок.

Актуальність теми. Медична галузь пережила значні зміни, пов'язані з технологічним прогресом та зростанням інтересу до медичного обладнання. В умовах постійного розвитку та зміни обладнання виникає потреба в удосконаленні процесів оренди або купівлі медичного обладнання для медичних закладів.

Мета і завдання роботи.

Мета даної роботи полягає у створенні інноваційного додатку, який надасть зручний та ефективний інструмент для пацієнтів та медичних закладів управляти процесом оренди або купівлі медичного обладнання.. Для досягнення поставленої мети необхідно вирішити наступний спектр завдань:

1. Вивчити та проаналізувати вітчизняні та зарубіжні джерела про способи створення мобільних додатків під різні платформи,
2. Визначити та застосувати сучасні практики та технології;
3. Визначити основну платформу для розробки, фреймворки бібліотеки та середовище розробки;
4. Реалізувати програмний продукт під Android.

Використані методи. Платформа Android, мова програмування Kotlin, бібліотеки: Koin, Retrofit, OkHttp, Coil, Room, RecyclerView, Android Paging, Lottie, Kotlin Coroutines, Kotlin Flow, Material Design, Android Navigation, Safe Args,

архітектура - MVVM.

Отримані результати. У ході виконання дипломної роботи були досягнуті всі цілі, відповідно до поставлених завдань:

1. Були проаналізовані вітчизняні та зарубіжні джерела на тему розробки мобільних додатків.

2. Було проаналізовано значущість мобільних додатків для онлайн бронювання медичного обладнання та аналіз існуючих аналогів, які використовуються на території України.

3. Були проаналізовані цільові платформи для розробки мобільних додатків, їхні переваги та недоліки, були обрані сучасні технології, практики та набір інструментів для реалізації додатку.

4. Було реалізовано мобільний додаток під Android для онлайн бронювання медичного обладнання, який є швидким, зручним та естетично привабливішим за свої аналоги. Це робить додаток потенційно корисним для користування різними групами людей.

Публікації. На основі виконаної роботи не планувались.

Ключові слова: Android, Kotlin, медичне обладнання, Room, Coil, Retrofit, MVVM, додаток.

Бібліографічний опис ДР

Усач А.О. Інформаційна система для онлайн бронювання медичного обладнання: дипломна роб. бакалавра : 122 Комп'ютері науки / Усач Андрій Олександрович. – Київ, 2023. – 71 с.

Annotation

The conducted thesis work titled "Information System for Online Booking of Medical Equipment" was completed by *Andrii Usach*, a student of the Department of Biomedical Cybernetics of the FBMI, specialty *122 Computer Science*. The work was carried out within the educational-professional program "Computer Technologies in Biology and Medicine". The structure of the thesis includes the following elements: list of abbreviations, symbols, and terms; introduction; 3 chapters (literature review, theoretical part, practical part), each accompanied by respective conclusions; general conclusions; a list of references consisting of 26 English-language sources. The total volume of the work is 71 pages.

Relevance of the topic. The medical field has undergone significant changes due to technological advancements and increased interest in medical equipment. In the context of ongoing development and equipment evolution, there is a need to improve the processes of renting or purchasing medical equipment for healthcare institutions.

Purpose and objectives of the work.

The *purpose* of this work was to create an innovative application that provides a convenient and efficient tool for patients and medical institutions to manage the process of renting or purchasing medical equipment. To achieve this goal, the following tasks were addressed:

1. Studying and analyzing domestic and foreign sources on mobile application development for various platforms.
2. Identifying and applying modern practices and technologies.
3. Determining the main development platform, frameworks, libraries, and development environment.
4. Implementing the software product for Android.

Methods used. The methods used in this work include the Android platform, the Kotlin programming language, libraries and frameworks such as Koin, Retrofit, OkHttp, Coil, Room, RecyclerView, Android Paging, Lottie, Kotlin Coroutines, Kotlin Flow, Material Design, Android Navigation, Safe Args, and the MVVM architecture.

Results obtained. As a result of the thesis work, all set goals and tasks were successfully achieved. The outcomes include:

1. Analysis of domestic and foreign literature on mobile application development.
2. Analysis of the significance of mobile applications for online booking of medical equipment and an examination of existing analogs used in Ukraine.
3. Analysis of target platforms for mobile application development, their advantages and disadvantages, and the selection of modern technologies, practices, and tools for the application's implementation.
4. Implementation of a mobile application for Android, which is fast, convenient, and aesthetically appealing compared to its analogs. This makes the application potentially useful for various user groups.

Publications. None were planned based on the work.

Keywords: Android, Kotlin, medical equipment, Room, Coil, Retrofit, MVVM, application.

ЗМІСТ

ПЕРЕЛІК СКОРОЧЕНЬ, УМОВНИХ ПОЗНАЧЕНЬ, ТЕРМІНІВ.....	10
ВСТУП.....	11
РОЗДІЛ 1 АНАЛІТИЧНИЙ ОГЛЯД ЛІТЕРАТУРНИХ ДЖЕРЕЛ	15
1.1 Мобільні додатки для купівлі медичного обладнання	15
1.2 Недоліки існуючих додатків	17
1.3 Огляд літератури для розробки мобільних додатків	18
Висновки до розділу 1	19
РОЗДІЛ 2 ТЕОРЕТИЧНА ЧАСТИНА	20
2.1 Android платформа	21
2.2 iOS платформа	22
2.3 Windows(phone) платформа	23
2.4 Native чи Cross-platform	24
2.5 Аналіз фреймворків.....	27
Висновки до розділу 2.....	29
РОЗДІЛ 3 ПРАКТИЧНА ЧАСТИНА	31
3.1 Розробка програмного продукту.....	31
3.1.1 Платформа.....	31
3.1.2 Мова програмування	31
3.1.3 Архітектура	32
3.1.4 Фреймворки та бібліотеки.....	35
3.1.5 Середовище розробки.....	37
3.2 Вхід в систему.....	39
3.3 Реєстрація.....	42

3.4 Головна сторінка.....	45
3.4 Пошук.....	48
3.4.1 Каталог	48
3.4.2 Сторінка пошуку.....	50
3.4.3 Фільтрація	54
3.4.4 Сортування.....	56
3.5 Публікація медичного обладнання.....	57
3.6 Сторінка медичного приладу.....	60
3.7 Інформування про помилки	62
3.7 Розрахунок економічного ефекту	63
Висновки до розділу 3.....	65
ЗАГАЛЬНІ ВИСНОВКИ.....	67
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ	69

ПЕРЕЛІК СКОРОЧЕНЬ, УМОВНИХ ПОЗНАЧЕНЬ, ТЕРМІНІВ

MVVM	Model-View-ViewModel
MVP	Model-View-Presenter
API	Application Programming Interface
OS	Operating System
SDK	Software Development Kit
JSON	JavaScript Object Notation
XML	Extensible Markup Language
HTTP	HyperText Transfer Protocol
UI	User Interface
DI	Dependency Injection
IDE	Integrated Development Environment
URL	Uniform Resource Locator
UX	User Experience

ВСТУП

Актуальність дипломної роботи

За останні десятиліття медична галузь пережила значні зміни, пов'язані з технологічним прогресом та зростанням інтересу до медичного обладнання. В умовах постійного розвитку та зміни обладнання виникає потреба в удосконаленні процесів оренди або купівлі медичного обладнання для медичних закладів. З метою поліпшення цих процесів та забезпечення зручності та ефективності, пропонується розробка мобільного додатку для оренди або купівлі медичного обладнання на платформі Android.

У процесі розробки додатку буде приділено увагу забезпеченню зручного та інтуїтивно зрозумілого інтерфейсу для користувачів, а також на створення безпечної системи для зберігання та обробки конфіденційної інформації про замовлення та медичне обладнання. Використовуватимуться сучасні методи та технології розробки мобільних додатків на платформі Android з метою створення стабільного та функціонального додатку для оренди або купівлі медичного обладнання.

Додаток буде базуватися на сучасних стандартах та найкращих практиках розробки мобільних додатків. Для цього будуть використані мова програмування Kotlin, що є основною мовою розробки для платформи Android. Будуть використані потужні інструменти розробки, такі як Android Studio та відповідні фреймворки, що дозволять ефективно створити функціональний та естетичний інтерфейс користувача.

Для підтримки обміну даними та комунікації між медичними закладами та постачальниками медичного обладнання, буде використана система баз даних, яка забезпечить надійне зберігання та доступ до інформації про товари, наявність, ціни та замовлення. Також будуть застосовані механізми аутентифікації та авторизації, що гарантуватимуть безпеку та захист персональних даних користувачів.

В процесі розробки додатку будуть враховані особливості вимог щодо медичного обладнання, його характеристик та варіантів оренди або купівлі. Буде

забезпечена можливість детального перегляду та порівняння обладнання, відображення фотографій, технічних характеристик та відгуків користувачів. Крім того, додаток буде мати функціонал для оформлення замовлення, оплати та відстеження стану доставки.

У підсумку, розробка мобільного додатку для оренди або купівлі медичного обладнання на платформі Android має на меті поліпшити процес управління медичним обладнанням, забезпечити зручність та швидкість вибору, оренди або купівлі необхідного медичного обладнання. Розробка такого додатку має потенціал значно спростити та прискорити процес придбання обладнання, знизити адміністративні витрати та покращити загальну ефективність медичних закладів.

Окрім того, застосування мобільних технологій та додатків у медичній сфері може сприяти покращенню якості надання медичних послуг. Лікарі та медичний персонал матимуть доступ до широкого спектру медичного обладнання через додаток, що дозволить їм швидше та більш ефективно здійснювати діагностику та лікування. Пацієнти також отримають зручну платформу для замовлення необхідного обладнання та отримання медичних послуг у своєму розпорядженні.

Об'єкт дослідження (модернізації/ розробки):

Створення програмного забезпечення під Android платформу, використання сучасних підходів так технологій у розробці мобільних додатків.

Вихідні дані: мобільний додаток під Android платформу для онлайн бронювання медичного обладнання

Мета і завдання дипломної роботи

Мета роботи: створення інноваційного додатку, який надасть зручний та ефективний інструмент для пацієнтів та медичних закладів управляти процесом оренди або купівлі медичного обладнання. Додаток буде надавати можливість користувачам зручно переглядати доступне обладнання, здійснювати бронювання або замовлення, отримувати інформацію про наявність нових пропозицій та стан замовлення. Крім того, постачальники медичного обладнання матимуть можливість пропонувати свої товари та отримувати замовлення через додаток, що

спростить процес комунікації та покращить обмін інформацією.

Завдання даної дипломної роботи:

1. Аналіз вітчизняних та зарубіжних джерел про способи створення мобільних додатків під різні платформи,
2. Застосування сучасних практик та технологій;
3. Визначення основної платформи для розробки, фреймворків бібліотек та середовища розробки;
4. Реалізування програмного додатку під Android.

Використані методи

У ході виконання дипломної роботи були розглянуті цільові платформи для мобільних додатків: Android, iOS, WindowsPhone, підходи до розробки мобільних додатків: Native та CrossPlatform, основні фреймворки для розробки за допомогою CrossPlatform підходу: ReactNative та Flutter, мови програмування для Native підходу: Java, Kotlin, Swift та основні бібліотеки, які використовуються для написання нативних додатків під Android та IOS. Для Android: Koin, Retrofit, OkHttp, Coil, Room, RecyclerView, Android Paging, Lottie, Kotlin Coroutines, Kotlin Flow, Material Design, Android Navigation, Safe Args. Для iOS: UIKit, SwiftUI, CoreData, Alamofire.

Отримані результати

В результаті виконання дипломної роботи були досягнуті усі цілі та отримано ряд позитивних результатів:

1. Під час роботи було проведено аналіз вітчизняних та зарубіжних джерел, що стосувалися різних способів створення мобільних додатків на різних платформах.
2. Також був проведений аналіз існуючих аналогів на ринку мобільних додатків для онлайн бронювання медичного обладнання.
3. У процесі розробки були використані сучасні практики та технології, що дозволило створити функціонально-багатий та естетично привабливий додаток.
4. Для реалізації програмного додатку були використані популярні фреймворки, бібліотеки та середовище розробки, що дозволило ефективно

створити функціональність додатку та забезпечити його високу продуктивність та якість.

5. Реалізовано Android додаток для онлайн бронювання медичного обладнання.

Обґрунтування прийнятих рішень за темою дипломної.

У ході аналізування було обрано та обґрунтовано використання: цільової платформи Android, мови програмування Kotlin, бібліотек та фреймворків: Koin, Retrofit, OkHttp, Coil, Room, RecyclerView, Android Paging, Lottie, Kotlin Coroutines, Kotlin Flow, Material Design, Android Navigation, Safe Args, архітектури MVVM.

Публікації. На основі виконаної роботи не планувались.

Структура роботи

Дипломна роботи за темою «Інформаційна система для онлайн бронювання медичного обладнання(графічний інтерфейс)» виконана студентом Усачем Андрієм Олександровичем зі спеціальності 122 «Комп'ютерні науки» за освітньо-професійною програмою «Комп'ютерні технології в біології та медицині», побудовано за класичним типом та викладена на 71 сторінках машинописного тексту. Звіт складається з: вступу; 3 розділів (аналітичний огляд літературних джерел за темою ДР, теоретична частина, практична реалізація задачі за темою ДР), висновків до кожного з цих розділів; загальних висновків; списку використаних джерел, який налічує 26 джерел та додатків (0 – на кирилиці). В звіті представлено 18 рисунків.

РОЗДІЛ 1

АНАЛІТИЧНИЙ ОГЛЯД ЛІТЕРАТУРНИХ ДЖЕРЕЛ

1.1 Мобільні додатки для купівлі медичного обладнання

Мобільні додатки для купівлі та бронювання медичного обладнання в Україні стають все більш популярними і користуваними. Це пояснюється зростанням свідомості та використання сучасних технологій в медичній сфері.

Медичні заклади дедалі більше орієнтуються на впровадження мобільних додатків, щоб забезпечити зручний доступ до медичного обладнання для своїх співробітників. Вони розуміють важливість ефективного управління та оптимізації процесу придбання та бронювання обладнання. Мобільні додатки надають зручний спосіб замовлення необхідного обладнання, відстеження його стану та швидкого отримання результатів.

Фахівці зі сфери охорони здоров'я також проявляють все більший інтерес до мобільних додатків для купівлі та бронювання медичного обладнання. Ці додатки надають зручний і прозорий спосіб замовлення та отримання обладнання, дозволяють відстежувати його доступність та забезпечити своєчасну доставку.

Одна з основних причин використання таких додатків полягає в зручності та доступності для користувачів. Вони дозволяють швидко та зручно здійснювати покупки або бронювання медичного обладнання просто зі свого смартфона або планшета. Це особливо важливо для фахівців з медичної сфери, які можуть швидко знайти та замовити необхідне обладнання для своєї практики.

Крім того, використання мобільних додатків сприяє збереженню часу та ефективній організації процесу купівлі або бронювання. Вони дозволяють користувачам швидко переглянути асортимент медичного обладнання, порівняти ціни та характеристики, а також здійснити оплату та отримати підтвердження усього в одному додатку.

Також використання таких додатків сприяє автоматизації процесу та забезпечує точність та надійність у роботі. Завдяки цьому, клієнти та фахівці

можуть бути впевнені у якості та доставці обладнання.

Проте попри росту популярності цих додатків, в Україні налічується досить мало мобільних застосунків цього типу. Найбільш популярними є Tabletki.ua та Rozetka, які є не цілеспрямованими на даний вид послуг.

Tabletki.ua - це популярний мобільний додаток в Україні, який надає користувачам зручну та доступну інформацію про ліки, аптеки та здоров'я. Завдяки цьому додатку, користувачі можуть швидко знайти необхідну інформацію про різноманітні медичні препарати, включаючи їх склад, дозування, показання до застосування, протипоказання та побічні ефекти.

Основною перевагою Tabletki.ua є можливість здійснювати онлайн замовлення препаратів, що дозволяє користувачам швидко та зручно придбати потрібні ліки без відвідування аптеки. Крім того, додаток надає можливість порівняти ціни на ліки у різних аптеках, що допомагає знайти найвигіднішу пропозицію та заощадити гроші.

Tabletki.ua також має функцію пошуку по різних аптеках, що дозволяє знайти найближчі аптеки та переглянути їхні контактні дані та графіки роботи. Користувачі можуть зручно вибрати найближчу або найзручнішу для них аптеку, а також переглянути відгуки і рейтинги аптек від інших користувачів.

Додаток Tabletki.ua дозволяє також налаштувати нагадування про прийом ліків, що допомагає не пропускати необхідний час та дозування препаратів. Користувачі можуть створювати персоналізовані розклади прийому ліків та отримувати повідомлення-нагадування від додатку.

Крім інформації про ліки, Tabletki.ua також надає можливість купувати просте медичне обладнання, що дозволяє користувачам придбати необхідні засоби для домашнього медичного догляду. В асортименті доступні такі предмети, як термометри, апарати для вимірювання тиску, пульсометри, стетоскопи, перев'язувальні матеріали та багато іншого.

Rozetka - це один з найпопулярніших онлайн-магазинів в Україні, який надає широкий вибір товарів, включаючи медичне обладнання. У додатку Rozetka існує велика кількість переваг, які роблять його привабливим для покупки медичних

пристроїв.

Однією з головних переваг Rozetka є широкий асортимент медичного обладнання. Ви можете знайти різноманітні медичні пристрої, включаючи термометри, апарати для вимірювання тиску, електрокардіографи, інгалятори, масажери та багато інших. Наявність широкого спектра продуктів дозволяє вам знайти саме те, що вам потрібно для задоволення вашого медичного обладнання.

Крім того, у додатку Rozetka є можливість порівняти характеристики та ціни на різні пристрої. Це дозволяє вам зробити інформований вибір, порівнюючи різні моделі та їх функціональність. Відгуки та рейтинги від інших покупців також доступні, що допомагає вам зрозуміти якість та ефективність продукту перед покупкою.

Процес замовлення в додатку Rozetka дуже зручний і простий. Ви можете швидко додати потрібний вам медичний пристрій до кошика, обрати зручний спосіб доставки та оплати, а потім зробити замовлення всього кількома клацаннями. Додаток також надає акції та знижки, що дозволяє зекономити гроші при покупці медичного обладнання.

1.2 Недоліки існуючих додатків

Варто враховувати, що хоча додатки, такі як Rozetka, Tabletki.ua та інші, надають можливість замовити медичне обладнання, вони можуть мати свої обмеження та недоліки, особливо коли мова йде про бронювання або оренду цих пристроїв.

По-перше, такі додатки зазвичай спрямовані на продаж товарів, а не на оренду. Вони можуть мати обмежені можливості щодо бронювання обладнання на обмежений період часу. Це означає, що якщо вам потрібно орендувати медичне обладнання на короткий термін, ці додатки можуть не мати відповідних функцій та опцій.

По-друге, додатки для замовлення медичного обладнання можуть не мати вичерпної інформації про умови оренди, такі як терміни, вартість та правила

повернення. Це може створити нестабільність та незручності при оренді медичного обладнання, оскільки може бути неясність стосовно умов та відповідальності.

1.3 Огляд літератури для розробки мобільних додатків

Під час проходження переддипломної практики було проведено аналіз даних з різних літературних джерел та веб-сайтів, пов'язаних з розробкою мобільних додатків під різні платформи. Дослідження охоплювало різноманітні аспекти розробки, включаючи мови програмування, фреймворки, інструменти та методології.

З літературних джерел були взяті у розгляд книги, які пропонували поглиблене вивчення платформ Android та iOS, такі як "Beginning Android Programming: Develop and Design" автора Кріса Гріффіта та "iOS Programming: The Big Nerd Ranch Guide" авторів Кріса Марфі та Білла Філіпса. Ці книги надали фундаментальні знання про основи розробки додатків під ці платформи, включаючи архітектуру, інструменти та практичні прийоми.

Крім того, були вивчені книги та публікації, присвячені розробці мобільних додатків з використанням кросплатформених фреймворків, таких як "React Native in Action" автора Натана Фотіуса та "Flutter for Beginners: An introductory guide to building cross-platform mobile applications with Flutter and Dart". Ці матеріали надали важливі відомості про побудову кросплатформених додатків та засоби розробки, що сприяли уніфікації та ефективності процесу розробки.

Крім літературних джерел, було проведено дослідження веб-сайтів, таких як офіційна документація Android та iOS, офіційні сайти React Native та Flutter, а також форум Stack Overflow. Підручники та документація на офіційних веб-сайтах React Native та Flutter також виявилися цінними джерелами інформації. Вони надають детальні пояснення щодо основних концепцій та принципів розробки мобільних додатків за допомогою цих фреймворків. Вони охоплюють різні аспекти, включаючи роботу зі зображеннями, навігацію, роботу зі звуком, а також інтеграцію зі сторонніми бібліотеками та сервісами.

Форум Stack Overflow є ще одним корисним ресурсом для розробників мобільних додатків. Він надає можливість знайти відповіді на конкретні технічні питання або вирішення проблем, з якими можуть стикатися розробники. Багато професіоналів з досвідом у розробці мобільних додатків активно взаємодіють на цьому форумі, надаючи корисні поради та рішення.

Всі ці ресурси разом надають розробникам широкий спектр інформації, необхідної для ефективної розробки мобільних додатків. Вони допомагають засвоїти основні принципи та інструменти, надають приклади коду та рекомендації щодо оптимальної структури додатків. Завдяки цим ресурсам розробники можуть ефективно вивчати, розвивати та покращувати свої навички у розробці мобільних додатків під різні платформи.

Також були використані інші літературні джерела для розробки Android додатків, офіційні документації використаних бібліотек та фреймворків, посібник з використання середовища розробки Android Studio.

Висновки до розділу 1

Користування мобільними додатками для купівлі та бронювання медичного обладнання має свої переваги, такі як зручність, швидкість та доступність. Вони дозволяють користувачам знайти необхідні товари, порівняти ціни та зробити замовлення всього в одному додатку. Однак, відповідно до конкретних потреб та вимог, необхідно розглянути альтернативні варіанти, такі як оренда медичного обладнання, які можуть вимагати інших рішень та підходів.

В результаті проходження переддипломної практики та аналіз даних з різних джерел літератури та веб-сайтів дозволили розширити розуміння процесу розробки мобільних додатків та отримати цінну інформацію щодо найкращих практик, новітніх технологій та інструментів.

РОЗДІЛ 2

ТЕОРЕТИЧНА ЧАСТИНА

Розробка мобільних додатків стала невід'ємною частиною сучасного цифрового світу. Із зростанням популярності мобільних пристроїв і залученням їх у всі сфери життя, розробники мають унікальну можливість створити інноваційні додатки, які забезпечують користувачам широкий спектр функціональності та відкривають нові перспективи.

Однак, розробники стикаються з різноманітними викликами під час розробки мобільних додатків, особливо коли розглядаються різні платформи. Наявність різних мобільних платформ, таких як Android, iOS та Windows, ставить перед розробниками завдання вибору платформи, що найкраще відповідає їх потребам та цілям проекту [1].

Цей розділ дипломної роботи присвячений дослідженню теоретичних аспектів розробки мобільних додатків під різні платформи. Головною метою цього дослідження є вивчення особливостей кожної платформи, а також порівняння їх переваг та недоліків.

В рамках цієї дипломної роботи будуть розглянуті ключові аспекти розробки мобільних додатків, включаючи мови програмування, інтегровані середовища розробки (IDE), фреймворки та інші інструменти. Крім того, будуть проаналізовані переваги та недолі кожної платформи з урахуванням функціональності, продуктивності, доступності ресурсів для розробників та популярності серед користувачів.

Практичне значення цього дослідження полягає в наданні розробникам інформації та рекомендацій щодо вибору оптимальної платформи для розробки мобільних додатків залежно від їх потреб, цілей та обмежень. Розуміння особливостей різних платформ і їх впливу на процес розробки дозволить розробникам прийняти обґрунтовані рішення щодо вибору платформи та оптимізації роботи над проектом.

2.1 Android платформа

Android — це операційна система для мобільних пристроїв, розроблена компанією Google. Її використовують на мільйонах смартфонів, планшетах і інших пристроях по всьому світу. Основна мета Android полягає в наданні зручного та масштабованого середовища для розробки та використання різноманітних мобільних додатків [10].

Аналізуючи розробку мобільних додатків під платформу Android, можна виявити ряд переваг та функцій, що роблять її привабливим вибором для розробників:

1. Широке поширення: Android займає значну частку ринку мобільних пристроїв, що відкриває великі можливості для досягнення широкої аудиторії. Маючи велику базу користувачів, розробники мають більше шансів на успіх своїх додатків та залучення великої кількості користувачів [10].

2. Відкрита платформа: Android є відкритою платформою, що дозволяє розробникам мати більше контролю над своїми додатками. Розробники можуть налаштовувати додатки, адаптувати їх під різні пристрої та операційні системи, використовувати різноманітні API та інструменти для створення багатофункціональних додатків [1].

3. Мова програмування Kotlin: За останні роки мова програмування Kotlin набула значного популярності серед розробників Android-додатків. Kotlin пропонує багато переваг, таких як безпеку типів, покращену підтримку функціонального програмування та сумісність з Java. Він дозволяє розробникам писати більш чистий, ефективний та експресивний код [6].

4. Багатофункціональність та інтеграція: Android надає розробникам широкий набір функцій та можливостей, включаючи доступ до камери, геолокації, сенсорів, роботи з базами даних та інше. Крім того, Android пропонує різні сервіси та API для інтеграції з іншими популярними сервісами, такими як Google Maps, Firebase, Google Pay та багатьма іншими. Це дозволяє розробникам створювати додатки, які максимально використовують можливості пристрою та забезпечують

взаємодію з різноманітними сервісами та платформами [1].

5. Розширена спільнота розробників: Android має велику та активну спільноту розробників, що забезпечує доступ до безлічі ресурсів, форумів, бібліотек та інструментів. Розробники можуть з легкістю знайти відповіді на свої запитання, обговорити проблеми та співпрацювати з іншими фахівцями. Це допомагає прискорити процес розробки та покращити якість додатків [10].

6. Можливості монетизації: Android надає розробникам різні можливості для монетизації своїх додатків. Вони можуть використовувати рекламу, внутрішні покупки, підписки, спонсорські контент та інші стратегії для отримання прибутку від своїх додатків. Багатий екосистема Google Play допомагає розробникам залучити користувачів та отримати дохід від їх додатків [10].

2.2 iOS платформа

iOS є операційною системою, розробленою компанією Apple для своїх мобільних пристроїв, таких як iPhone, iPad та iPod Touch. Вона володіє кількома особливостями та перевагами, які роблять її привабливою для розробників мобільних додатків [2].

Розробка мобільних додатків під платформу iOS привертає увагу розробників з різних причин і пропонує набір переваг та можливостей. При розгляді розробки під iOS слід враховувати декілька ключових факторів, які впливають на привабливість цієї платформи для розробників:

1. Широке поширення: iOS є однією з найпопулярніших мобільних платформ у світі, з великою базою активних користувачів. Це створює великі можливості для досягнення великої аудиторії та успіху додатку [2].

2. Висока якість додатків: Apple відома своєю високою якістю та вимогливістю до додатків, що допомагає забезпечити високу якість та надійність додатків на платформі iOS. Це дозволяє розробникам створювати користувачам продуктивні, добре працюючі додатки [2].

3. Екосистема Apple: iOS входить в екосистему Apple, що включає інші

пристрої та послуги, такі як iPhone, iPad, Apple Watch, Apple TV та інші. Розробка під iOS дозволяє інтегруватись з цими пристроями та послугами, надаючи розширені можливості для створення інноваційних та взаємодіючих додатків [16].

4. Мова програмування Swift: Запровадження мови програмування Swift спрощує розробку під iOS. Swift є сучасною, швидкою та безпечною мовою програмування, яка дозволяє швидко писати чистий та ефективний код. Вона має чіткий синтаксис, підтримку безпеки типів та багато інструментів для розробки, що робить розробку під iOS більш приємною та продуктивною [2].

5. Інтегрованість зі службами Apple: Розробка під iOS дозволяє використовувати широкий спектр служб та функцій, наданих Apple, таких як iCloud, Apple Pay, Apple Maps, HealthKit та інші. Це дозволяє розробникам створювати додатки, які максимально використовують можливості пристроїв та взаємодіють з екосистемою Apple [2].

6. Більш висока монетизація: iOS має репутацію платформи з вищими доходами для розробників порівняно з іншими платформами. Користувачі iOS частіше схильні до покупок додатків та внутрішніх покупок, що надає можливості для монетизації додатків та заробітку [16].

7. Більш високі стандарти безпеки: Apple відома своїми високими стандартами безпеки. Платформа iOS має вбудовані заходи безпеки, такі як шифрування даних, обмеження доступу до ресурсів та контроль над дозволами додатків, що допомагає забезпечити безпеку та конфіденційність користувачів [16].

2.3 Windows(phone) платформа

Windows Phone OS була операційною системою, розробленою компанією Microsoft для своїх мобільних пристроїв, таких як смартфони. Однак, Microsoft офіційно припинила підтримку Windows Phone OS в 2017 році, і внаслідок цього платформа втратила свою популярність та значущість на ринку [12].

Розробка мобільних додатків під Windows не має такої популярності та привабливості порівняно з іншими платформами, такими як Android та iOS. Ось

кілька факторів, які можуть впливати на вибір розробки під Windows та чому це може бути менш доцільним:

1. Нижча популярність: Windows Phone/Windows 10 Mobile не здобув великого успіху на ринку мобільних пристроїв порівняно з Android та iOS. Це означає, що користувачі Windows-пристроїв становлять меншу частку ринку, що обмежує потенційну аудиторію для мобільних додатків [12].

2. Обмежені можливості: Windows-платформа має обмежену набір функцій та сервісів порівняно з Android та iOS. Деякі функціональності, які є стандартними для мобільних додатків, можуть бути обмеженими або недоступними на Windows-платформі [12].

3. Відсутність екосистеми: Відсутність широкої екосистеми та інтеграції з іншими пристроями та послугами може бути ще однією причиною, чому розробка під Windows не є пріоритетною. Відсутність сильної підтримки від сторонніх розробників та обмежений доступ до популярних сервісів можуть ускладнити розробку та призвести до меншої функціональності додатків [12].

4. Менші можливості монетизації: З меншою аудиторією та обмеженими можливостями екосистеми Windows, монетизація додатків під цю платформу може бути менш вигідною. Користувачі Windows-пристроїв можуть бути менш схильні до покупок додатків або внутрішніх покупок, що може вплинути на дохід [12].

2.4 Native чи Cross-platform

При створенні мобільних додатків існують два головних підходи: нативна розробка та крос-платформна розробка. Кожен з цих підходів має свої переваги та обмеження.

Нативна розробка відноситься до процесу створення програмного забезпечення (у нашому випадку - мобільних додатків) для конкретної платформи або операційної системи. При нативній розробці використовуються інструменти, мови програмування та середовища, які специфічні для даної платформи.

Наприклад, для розробки нативних додатків під Android використовуються мови програмування Java або Kotlin та інструменти розробки, такі як Android Studio[1]. Для розробки нативних додатків під iOS використовуються мови програмування Objective-C або Swift, а основним інтегрованим середовищем розробки є Xcode [2].

Нативні додатки мають прямий доступ до функцій та можливостей операційної системи, що дозволяє досягти найвищого рівня продуктивності, ефективності та швидкодії. Вони можуть повністю використовувати функціональні можливості пристрою та оптимізовані для конкретної платформи [12].

Також нативні додатки можуть використовувати оновлення та нові функції, які надаються розробникам платформи. Це дозволяє створювати додатки з актуальними можливостями та використовувати нові функції, які надаються платформою [10].

Окрім цього нативні додатки маю найкращу інтеграцію з платформою та пристроєм. Даний вид додатків можуть повністю використовувати можливості пристрою та інтегруватися зі сторонніми сервісами, API та функціями платформи. Це дозволяє створювати додатки, які максимально використовують функціональність та можливості пристрою [14].

Однак, попри всі свої позитивні сторони нативна розробка також має свої обмеження.

Для створення додатків під кожен платформу потрібно написати окремий код, що вимагає більше зусиль та часу. Якщо вам потрібно розробити додаток для кількох платформ, вам доведеться писати окремий код для Android та iOS, що може призвести до збільшення витрат часу та ресурсів на розробку та підтримку [4].

Нативна розробка вимагає глибоких знань та експертизи в програмуванні для кожної платформи, оскільки вам потрібно розуміти особливості мов програмування, інструменти та API, що використовуються на кожній платформі.

Крос-платформна розробка - це підхід до створення програмного забезпечення (у нашому випадку - мобільних додатків), що дозволяє створювати один додаток, який може працювати на різних платформах, таких як Android, iOS,

Windows та інші [4].

Замість того, щоб розробляти окремі додатки для кожної платформи, використовується спільний код, який може бути використаний на різних платформах.

Крос-платформні фреймворки та інструменти дозволяють розробникам створювати додатки швидше, оскільки вони можуть використовувати спільний код для різних платформ. Це зменшує потребу у подвійній розробці та дозволяє економити час [3].

За рахунок використання спільного коду для різних платформ, крос-платформна розробка може знизити витрати на розробку та підтримку додатків. Розробники можуть зосередитися на написанні одного коду, який працюватиме на багатьох платформах [3].

Крос-платформні фреймворки дозволяють здійснювати одноразове оновлення для всіх платформ, що дозволяє швидко внести зміни та виправлення помилок на всіх пристроях. Це забезпечує більш просту та ефективну підтримку додатків [4].

Крос-платформні додатки можуть охоплювати більший ринок, оскільки вони можуть бути запущені на різних платформах, таких як Android, iOS, Windows та інші. Це дає можливість досягти більшої аудиторії та залучити більше користувачів [5].

Хоча крос-платформна розробка має свої переваги, вона також має деякі обмеження, які варто враховувати при виборі цього підходу.

Крос-платформні фреймворки можуть мати обмежену функціональність порівняно з нативною розробкою. Деякі продвинуті функції та можливості, що надаються платформами, можуть бути недоступні або складні для реалізації в крос-платформному середовищі [14].

Також крос-платформні рішення часто вимагають використання фреймворку, що може створювати залежність від стороннього розробника або організації. Це може призвести до обмежень у контролі та налагодженні додатку [14].

2.5 Аналіз фреймворків

Для нативної розробки під Android існує кілька важливих фреймворків, які допомагають розробникам створювати високоякісні мобільні додатки. Один з них - Android SDK, який включає в себе Android Studio (інтегроване середовище розробки), SDK-платформи, інструменти для компіляції та налагодження коду, а також API для доступу до функціональності пристрою [13]. Додатково, розробники можуть використовувати Kotlin - сучасну та безпечну мову програмування, що стала офіційною мовою для розробки Android-додатків [6]. Java також залишається популярною мовою програмування для розробки Android-додатків, з широкою підтримкою та великою кількістю ресурсів та документації [7]. Окрім цього, фреймворки, такі як Android Jetpack і Android Architecture Components, надають зручні інструменти та компоненти для ефективної розробки Android-додатків [10].

Крім Android SDK, Kotlin, Java, Android Jetpack, Android Architecture Components та RxJava, існує ще кілька значних фреймворків для нативної розробки під Android [1].

Для роботи з мережевими запитами часто використовується бібліотека Retrofit. Вона надає зручний інтерфейс для взаємодії з веб-сервісами, підтримує різні протоколи та формати даних, такі як JSON або XML. Retrofit спрощує роботу з API та дозволяє здійснювати мережеві запити ефективно та зручно [12].

Також однією з основних бібліотек для розробки Android додатків є Coil - легка та ефективна бібліотека для завантаження та відображення зображень в Android-додатках. Вона розроблена, щоб бути швидкою, простою у використанні та масштабованою [15].

Якщо у додатку планується використання баз даних – то найкраще підійде бібліотека Room. Це компонент Android Architecture Components, який надає легкий та зручний спосіб роботи з базою даних SQLite в Android-додатках. Room пропонує об'єктно-реляційне відображення (ORM) для простого доступу до даних та виконання SQL-запитів. Він автоматично генерує SQL-код на основі анотацій та дозволяє зручно взаємодіяти з базою даних [8].

При розробці додатку за допомогою мови програмування Kotlin, розробки отримує досить потужний інструмент для асинхронного програмування – Kotlin Coroutines.

Kotlin Coroutines - це бібліотека та підход до асинхронного програмування в мові Kotlin. Вона надає зручні та ефективні засоби для виконання асинхронних операцій, таких як мережеві запити, робота з базами даних та інші взаємодії зі зовнішніми джерелами даних. Корутини в Kotlin використовують легковажну підходу до асинхронності, де вони працюють на одному або кількох потоках, але не потребують відділення ниток оперативної системи. Це забезпечує ефективне використання ресурсів та підвищує продуктивність додатка [11].

Загалом Android розробники мають в арсеналі досить великий набір бібліотек, головною перевагою яких є офіційна підтримка власника та розробника платформи Android – Google, що надає впевненості в тому, що ці бібліотеки будуть сумісні з різними версіями платформи, підтримуватимуться довготривалий час, та з оновленнями не будуть мати змін, які зможуть привести до помилок, або ще гірше до крашів додатку [10].

Основні фреймворки для нативної розробки під iOS надають розробникам потужні інструменти для створення функціональних та естетично привабливих додатків. Ось кілька з них:

- **UIKit:** Цей фреймворк є основою для розробки інтерфейсу користувача в iOS-додатках. Він надає багатий набір компонентів, таких як кнопки, тексти, зображення, таблиці та інші, які допомагають побудувати інтерактивні та зручні інтерфейси [2].
- **SwiftUI:** Цей фреймворк є новітнім додатком до iOS-розробки, який пропонує декларативний підхід до створення інтерфейсів. За допомогою SwiftUI, розробники можуть швидко та легко створювати інтерактивні та адаптивні інтерфейси за допомогою простого коду [16].
- **Core Data:** Цей фреймворк надає засоби для роботи з базами даних в iOS-додатках. Завдяки Core Data, розробники можуть зберігати, отримувати та керувати даними в додатках, що дозволяє побудувати потужні та ефективні

функціональності.

- Alamofire: Це фреймворк для роботи з мережевими запитами в iOS-додатках. Він надає простий та зручний інтерфейс для виконання HTTP-запитів, обробки відповідей та взаємодії з веб-сервісами [2].

Загалом, ці фреймворки роблять розробку нативних додатків під iOS швидкою, ефективною та зручною, допомагаючи розробникам створювати додатки відповідно до вимог платформи з великою кількістю функціональних можливостей та високою якістю.

React Native - це один з найпопулярніших крос-платформних фреймворків, який базується на JavaScript та React. За допомогою React Native, розробники можуть створювати мобільні додатки, використовуючи спільний код, що працює як на Android, так і на iOS. Він надає доступ до нативних компонентів та може бути використаний для розробки швидких та ефективних додатків [3].

Flutter - це фреймворк, розроблений компанією Google, який використовує Dart для створення крос-платформних мобільних додатків. Flutter використовує власний движок рендерингу та надає доступ до віджетів, що виглядають як нативні компоненти. Він дозволяє створювати швидкі, красиві та високопродуктивні додатки для Android та iOS [4].

Висновки до розділу 2

У цьому теоретичному розділі дипломної роботи була проаналізована розробка мобільних додатків на різних платформах, зокрема на платформах Android, iOS та Windows Phone. Були розглянуті основні принципи розробки, переваги та недоліки кожної платформи, а також важливі фреймворки та інструменти для створення мобільних додатків.

Висновки з цього розділу вказують на те, що кожна платформа має свої унікальні особливості та переваги. Android є найпопулярнішою платформою з великою аудиторією користувачів, розширеними можливостями та розробчими інструментами. iOS, у свою чергу, відзначається своєю безпекою, стабільністю та

високою якістю додатків. Windows Phone, хоча втратив свою популярність, може бути варіантом для специфічних сценаріїв розробки.

Крім того, нативна розробка дозволяє отримати найкращу продуктивність та доступ до всіх можливостей платформи, а крос-платформний підхід може забезпечити швидку розробку для декількох платформ з використанням спільного коду.

Для успішної розробки мобільних додатків на різних платформах важливо враховувати потреби користувачів, характеристики платформи, особливості ринку та технологічні вимоги. Вибір платформи та підходу до розробки повинен здійснюватися на основі конкретних вимог та цілей проекту.

Даний теоретичний розділ є цінним джерелом інформації для подальшої розробки мобільного додатка для онлайн бронювання медичного обладнання.

РОЗДІЛ 3

ПРАКТИЧНА ЧАСТИНА

3.1 Розробка програмного продукту

3.1.1 Платформа

Аналізуючи теоретичний розділ дипломної роботи, було встановлено, що обрано платформу Android для розробки мобільного додатку для бронювання медичного обладнання з кількох вагомих причин.

По-перше, Android є найпоширенішою мобільною платформою на сучасному ринку. За статистикою, він має найбільшу частку ринку мобільних пристроїв, що означає широку аудиторію користувачів. Вибір Android-платформи дозволить досягти більшого охоплення та залучення більшої кількості користувачів до використання мобільного додатку для бронювання медичного обладнання [10].

По-друге, Android надає розробникам широкі можливості для створення різноманітних додатків. За допомогою нативних функцій та розширених API Android, розробники можуть реалізувати потрібну функціональність для бронювання медичного обладнання, включаючи доступ до камери, геолокації, сповіщень та інших функцій, які можуть бути важливими для медичного середовища [10].

По-третє, екосистема Android надає широкий вибір розробчих інструментів та підтримує різні мови програмування, зокрема Java та Kotlin. Це дозволяє розробникам обирати найбільш зручний та ефективний інструментарій для створення додатку. Наприклад, Kotlin є сучасною мовою програмування, що надає чистий та компактний код, покращує продуктивність та простоту розробки [1].

Нарешті, розглядаючи основні вигоди та можливості Android-платформи, було прийнято рішення обрати її для створення мобільного додатку для бронювання медичного обладнання.

3.1.2 Мова програмування

Обираючи мову програмування для розробки мобільного додатку, Kotlin є привабливим вибором порівняно з Java з кількох причин.

По-перше, Kotlin є сучасною мовою програмування, яка пропонує багато нововведень та покращень порівняно з Java. Вона має більш елегантний та компактний синтаксис, що дозволяє писати код більш зрозуміло та ефективно. Kotlin також підтримує безпеку типів, що допомагає уникати потенційних помилок та полегшує розробку стабільних та надійних додатків [10].

По-друге, Kotlin є інтероперабельною з Java, що означає, що існуючий код на Java може бути легко інтегрований в проект Kotlin. Це дає розробникам можливість поступово переходити до Kotlin, додавати новий код на Kotlin до існуючого проекту та використовувати переваги обох мов. Крім того, Kotlin підтримує всі фреймворки та бібліотеки, які доступні для розробки на Java, що робить його відмінним вибором для мобільної розробки [11].

По-третє, Kotlin має активну спільноту розробників, що забезпечує підтримку, документацію та навчальні матеріали. Це допомагає розробникам ефективно вивчати мову та швидко розвиватися в процесі розробки додатків [11].

Отже, обираючи Kotlin для розробки мобільного додатку, розробники отримують сучасну мову програмування з покращеними можливостями, високою сумісністю з Java, активною спільнотою розробників та широким спектром інструментів та бібліотек для мобільної розробки. Всі ці фактори роблять Kotlin привабливим вибором

3.1.3 Архітектура

В розробці мобільних додатків, вибір архітектури є важливим кроком, оскільки вона визначає спосіб організації коду та взаємодії між компонентами додатку. Дві основні архітектури, які широко використовуються в розробці Android додатків - це MVP (Model-View-Presenter) та MVVM (Model-View-ViewModel) [7].

MVP - це архітектура, в якій модель (Model) представляє даний додатку, представлення (View) відображає інтерфейс користувача, а презентер (Presenter) виступає посередником між моделлю та представленням. MVP дозволяє чітко розділити логіку бізнес-логіки від інтерфейсу користувача, що полегшує тестування та розширення додатку. Презентер відповідає за обробку взаємодії користувача та оновлення представлення на підставі даних з моделі [7].

Схематичне зображення архітектури зображено на рис. 3.1.

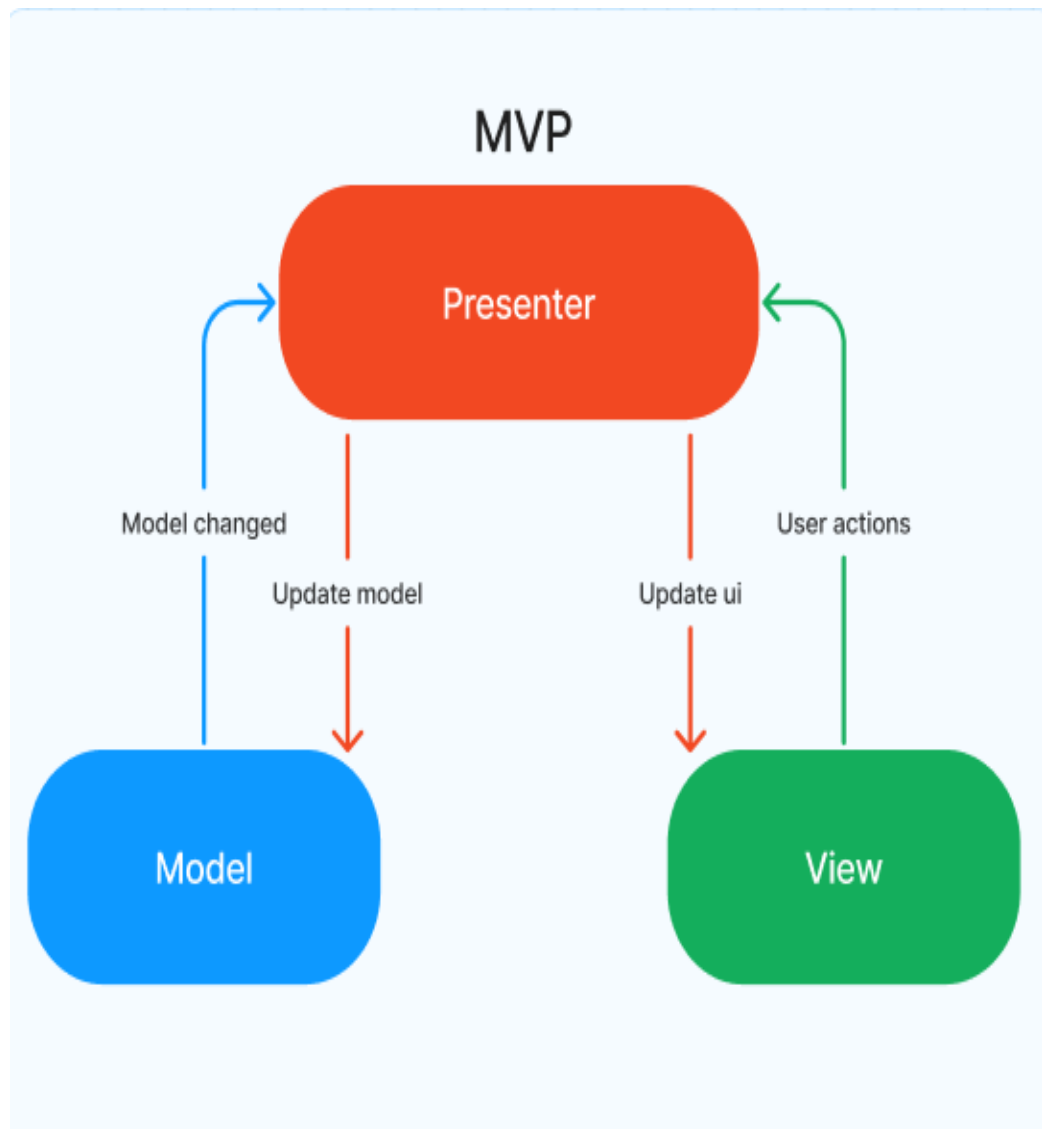


Рисунок. 3..1 - Шаблон архітектури MVP

MVVM - це архітектура, в якій модель (Model) містить бізнес-логіку та дані, представлення (View) відображає інтерфейс користувача, а вію-модель (ViewModel) відповідає за управління станом та взаємодією між моделлю та представленням. MVVM використовує прив'язку даних, що дозволяє автоматично оновлювати представлення при зміні даних у вію-моделі. Ця архітектура сприяє розділенню відображення даних та бізнес-логіки, забезпечуючи високу розширюваність та тестуваність [7]. Схематичне зображення архітектури зображено на рис. 3.2.

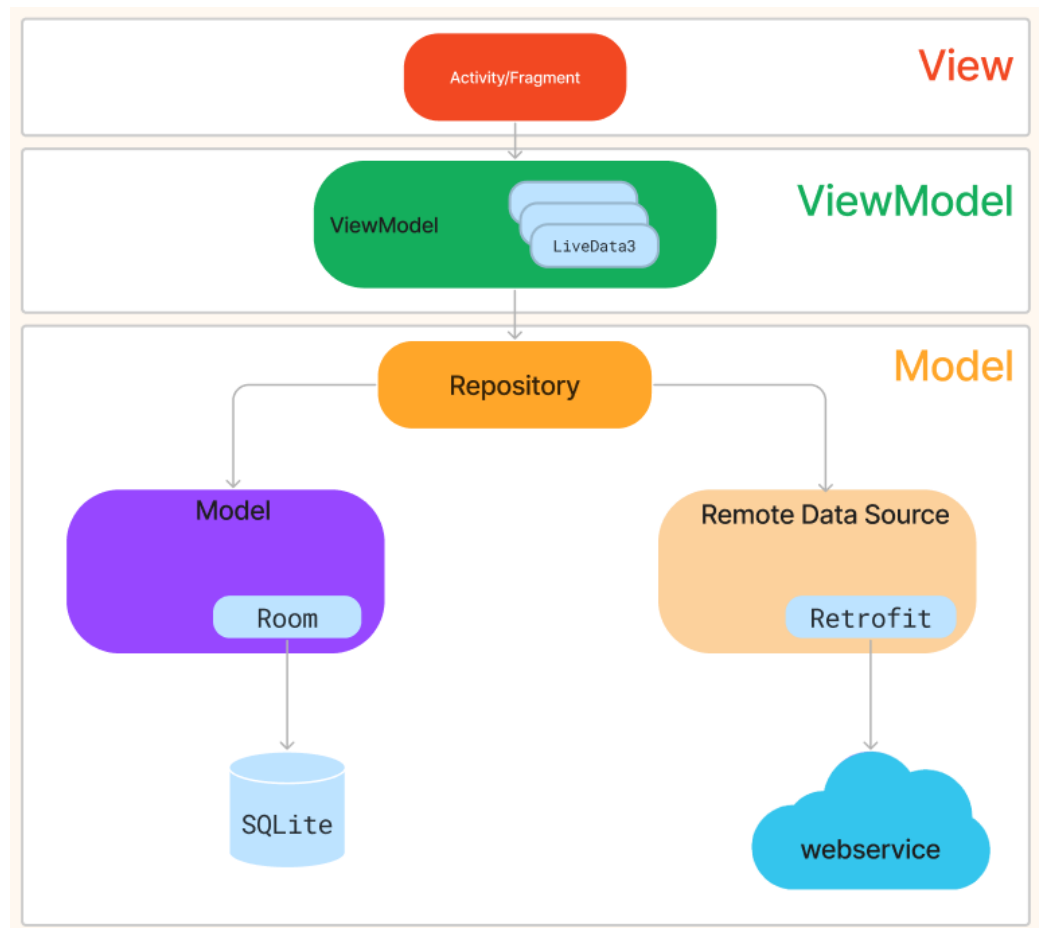


Рисунок.3.2 - Шаблон архітектури MVVM

Вибір архітектурного шаблону для розробки мобільного додатку є важливим кроком, і в даному випадку обрання архітектури MVVM має свої переваги порівняно з MVP:

1. Розділення логіки та представлення: MVVM дозволяє чітко розділити логіку додатку від його представлення. ViewModel виконує роль посередника між View і Model, управляючи потоком даних та взаємодією з бізнес-логікою. Це сприяє покращенню читабельності, тестуванню та підтримці коду [10].

2. Збереження стану інтерфейсу користувача: MVVM дозволяє зручно зберігати та відновлювати стан інтерфейсу користувача. Завдяки зв'язку даних між ViewModel і View, додаток може легко управляти станом UI під час повороту екрану або зміни конфігурації [8].

3. Більша гнучкість при тестуванні: MVVM сприяє більш простому і ефективному тестуванню додатків. ViewModel може бути легко тестований, оскільки він не має прив'язки до фреймворку UI. Це дозволяє швидко виявляти та

виправляти помилки, а також реалізовувати модульні та автоматизовані тести.

4. Легкість розширення та підтримки: MVVM забезпечує більшу гнучкість та легкість розширення додатків. Шар Model може бути замінений або розширений без змін у View і ViewModel. Це спрощує розробку нових функцій, впровадження змін та підтримку коду в майбутньому [8].

5. Підтримка реактивного програмування: MVVM добре поєднується з підходом реактивного програмування, таким як використання Kotlin Coroutines, Flow або LiveData. Це дозволяє створювати реактивні потоки даних та легко реагувати на їх зміни. Це особливо корисно для асинхронних операцій, таких як мережеві запити або отримання даних з бази даних [10].

Обираючи архітектуру MVVM, розробники отримують численні переваги, такі як зручне розділення логіки та представлення, збереження стану інтерфейсу користувача, полегшення тестування, легкість розширення та підтримки, а також можливість використання реактивного програмування. Всі ці фактори зробили архітектуру MVVM привабливим вибором для розробки мобільних додатків, зокрема для платформи Android. Обрання MVVM дозволяє розробникам зосередитись на розділенні відповідальностей, покращенні модульності та забезпеченні гнучкості і розширюваності додатку. Це сприяє покращенню розробки та підтримки додатку на протязі його життєвого циклу.

3.1.4 Фреймворки та бібліотеки

Для створення мобільного додатку під Android були використані наступні фреймворки та бібліотеки:

Koin - це легковаговий фреймворк для управління залежностями (dependency injection) в мобільних додатках на основі платформи Android. Він створений на основі мови Kotlin і пропонує простий та декларативний підхід до встановлення та використання залежностей [12].

Retrofit - це одна з найбільш використовуваних бібліотек для роботи з мережевими запитами у мобільних додатках, зокрема на платформі Android. Вона забезпечує простоту та зручність при взаємодії з веб-серверами та обробці отриманих даних.

Coil - це легковагова бібліотека для завантаження та відображення зображень в мобільних додатках на платформі Android. Вона розроблена з фокусом на простоту використання та ефективність [15].

Room - це бібліотека, що надає абстракцію бази даних SQLite для розробки мобільних додатків на платформі Android. Вона входить до складу Android Architecture Components і надає простий та зручний спосіб роботи з базою даних.

RecyclerView - це потужний компонент Android, який використовується для відображення списків даних з повторюваними елементами [10].

Android Paging - це компонент Android Architecture Components, який дозволяє розробникам ефективно завантажувати та відображати великі набори даних по частинам (сторінками) у спискових виглядах, таких як RecyclerView. Використання Android Paging допомагає зменшити використання пам'яті та оптимізувати швидкість завантаження даних [10].

Lottie - це бібліотека для Android, iOS та веб, яка дозволяє відтворювати анімації, створені у форматі JSON. Вона розроблена командою Airbnb і надає зручний спосіб використання векторних анімацій у мобільних додатках [12].

Kotlin Coroutines - це фреймворк для асинхронного програмування в мові Kotlin. Він надає спрощений підхід до конкурентності та асинхронності, що дозволяє зручно вирішувати завдання, які вимагають паралельної або асинхронної обробки [11].

Material Design - це дизайн-мова, розроблена Google, яка використовується для створення сучасних та стильних інтерфейсів користувача. Вона включає в себе набір принципів, стандартів та компонентів, які допомагають розробникам створювати додатки з високою якістю взаємодії та візуального враження [9].

Android Navigation - це компонент Android Jetpack, який дозволяє розробникам створювати зручну навігацію між екранами та переходи між фрагментами або активностями у мобільних додатках на платформі Android [10].

Safe Args - це частина Android Navigation Component, яка забезпечує безпечний і типобезпечний спосіб передачі аргументів між фрагментами або активностями. Використання Safe Args дозволяє уникнути помилок під час

передачі та отримання аргументів, а також надає перевагу статичній типізації [10].

3.1.5 Середовище розробки

Вибір правильного середовища розробки для побудови Android додатків є важливим з кількох причин. По-перше, ефективне середовище розробки допомагає прискорити процес розробки, надаючи потужні інструменти та підтримку для мов програмування, фреймворків і бібліотек. Це забезпечує більшу продуктивність розробників і дозволяє їм працювати швидше.

По-друге, правильне середовище розробки має багатий набір інструментів, які полегшують розробку Android додатків. Це можуть бути інтегровані редактори коду з функціями автодоповнення та перевірки помилок, налагоджувальні інструменти, емулятори пристроїв та інші. Функціональність цих інструментів сприяє створенню якісних додатків з більшою ефективністю.

По-третє, вибір середовища розробки пов'язаний з підтримкою мов програмування, таких як Kotlin і Java. Правильне середовище має вбудовану підтримку обраної мови програмування, забезпечуючи зручний синтаксис, автодоповнення та перевірку помилок. Це сприяє зручній розробці і покращує продуктивність розробників.

Наостанок, важливо враховувати сумісність середовища розробки з іншими інструментами і сервісами, які ви плануєте використовувати. Це можуть бути фреймворки, системи контролю версій, засоби тестування та інші інтеграції, які полегшують розробку та співпрацю в команді.

При розробці Android додатків з використанням Kotlin або Java, основними середовищами розробки, які можна використовувати, є Android Studio, Eclipse та IntelliJ IDEA.

Android Studio є офіційним інтегрованим середовищем розробки (IDE) для платформи Android. Воно базується на IntelliJ IDEA і забезпечує розширений набір інструментів для створення, редагування, налагодження та тестування Android додатків. Android Studio має вбудовану підтримку Kotlin та Java, надаючи розробникам багато функцій, таких як автодоповнення коду, вбудоване налагодження, інструменти для дизайну і багато іншого.

Eclipse - це інша популярна платформа розробки з відкритим вихідним кодом, яка також підтримує розробку Android додатків. Хоча підтримка для Android у Eclipse більше не активно оновлюється, воно все ще може використовуватись для розробки, особливо для тих, хто працює зі старішими проектами.

IntelliJ IDEA - це потужне комерційне середовище розробки, яке також надає зручні інструменти для розробки Android додатків. Воно має повну підтримку Kotlin і Java, дозволяючи розробникам працювати з будь-якою мовою програмування на свій вибір. IntelliJ IDEA пропонує багато розширень та інструментів, що полегшують розробку, відлагодження та управління проектами.

Використання сторонніх бібліотек є перевагою всіх трьох середовищ розробки для Android додатків (Android Studio, Eclipse, IntelliJ IDEA). Це дозволяє розробникам використовувати готові компоненти та інструменти для прискорення процесу розробки, таких як мережева взаємодія, робота з базами даних, графіка та інше.

Всі три середовища розробки (Android Studio, Eclipse, IntelliJ IDEA) надають можливість емуляції та тестування Android додатків на віртуальних пристроях з різними характеристиками. Це дозволяє розробникам перевірити, як їх додатки працюють на різних версіях Android та екранах різних розмірів.

Вбудовані інструменти для розробки інтерфейсу користувача є частиною всіх трьох середовищ розробки (Android Studio, Eclipse, IntelliJ IDEA). Розробники можуть використовувати візуальні редактори для швидкого створення та редагування макетів, налаштовувати розміщення елементів та змінювати кольори та стилі, спрощуючи процес розробки інтерфейсу.

Налагоджувальні інструменти, доступні в усіх трьох середовищах розробки (Android Studio, Eclipse, IntelliJ IDEA), допомагають розробникам виявляти та виправляти помилки у коді, забезпечуючи високу якість і надійність додатків.

Загалом, Android Studio може бути вважаний найкращим варіантом середовища розробки для Android додатків. Воно є офіційним середовищем розробки, розробленим спеціально для розробки Android додатків, і надає широкий набір інструментів та функцій для ефективної та продуктивної розробки.

Android Studio забезпечує повну підтримку мов Kotlin і Java, інтегровані інструменти для розробки інтерфейсу та налагодження, а також простоту використання Android SDK та Google сервісів. Воно також надає зручні емулятори та тестувальні інструменти для перевірки додатків на різних пристроях та версіях Android.

Загальне спільнота розробників Android вважає Android Studio найбільш популярним і надійним середовищем розробки для створення високоякісних Android додатків. Тому, вибір Android Studio може бути раціональним і вигідним рішенням для розробників, які прагнуть створювати потужні та успішні додатки для платформи Android.

3.2 Вхід в систему

Авторизація - це процес перевірки і підтвердження ідентичності користувача перед наданням доступу до певних функцій або ресурсів в додатку. Це забезпечує контроль над тим, хто має право використовувати додаток і його функціональні можливості.

Авторизація важлива з декількох причин. По-перше, вона забезпечує захист конфіденційності та приватності користувачів, оскільки несанкціоновані особи не зможуть отримати доступ до їх особистих даних. По-друге, авторизація дозволяє розробникам контролювати рівень доступу до різних функцій та ресурсів в додатку, забезпечуючи відповідність політикам безпеки і правам доступу.

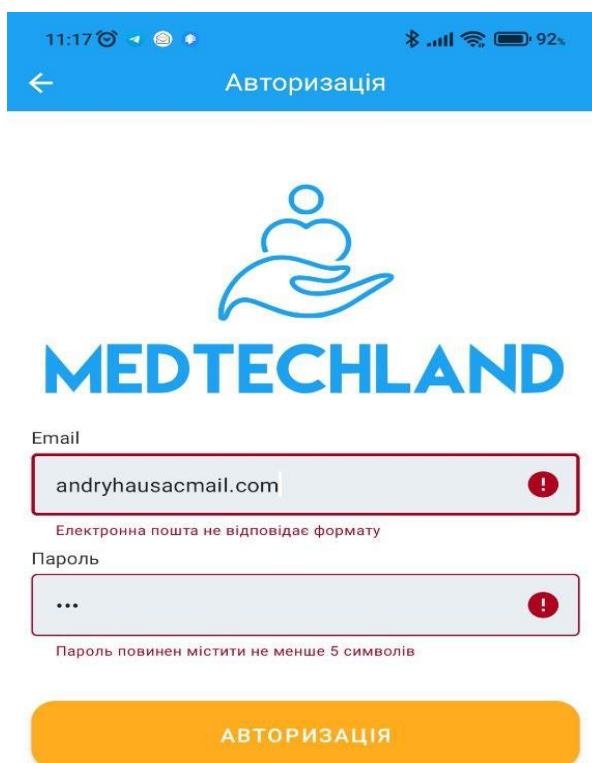
В додатку присутня можливість авторизації, яка забезпечує доступ до особистих облікових записів користувачів шляхом введення пошти та пароля. Цей механізм авторизації дозволяє забезпечити безпеку та обмеження доступу до конфіденційних даних тільки для авторизованих користувачів, забезпечуючи захист і приватність їх особистої інформації. Після введення пошти та пароля, вони перевіряються за допомогою сервера, і якщо дані вірні, користувач отримує повний доступ до функціоналу додатку та збереженим персональним налаштуванням.

На екрані наявні поля для вводу пошти та пароля, а також кнопка для входу



Рисунок 3.4 – Функція автозаповнення

На екрані авторизації виконується валідація полів пошти та пароля, що допомагає гарантувати правильність та коректність введених даних користувачем перед подальшою обробкою. Ця валідація включає перевірку формату пошти, наявності обов'язкових символів у паролі, а також може враховувати інші обмеження або правила, що встановлені для відповідних полів. При введенні некоректних даних у відповідних полях, відображаються повідомлення про помилку або підказки, що допомагають користувачеві виправити та внести потрібні зміни до введених даних перед продовженням процесу авторизації. Така валідація допомагає покращити взаємодію з додатком та забезпечує введення коректних даних для успішної авторизації користувача (рис. 3.5).



The screenshot shows the login interface for MedTechLand. At the top, the status bar displays the time 11:17, signal strength, Wi-Fi, and battery at 92%. Below the status bar is a blue header with a back arrow and the text "Авторизація". The MedTechLand logo, featuring a stylized person in a hand, is centered above the text "MEDTECHLAND". Below the logo are two input fields. The first field is labeled "Email" and contains the text "andryhausacmail.com". A red border and a red exclamation mark icon indicate an error, with the message "Електронна пошта не відповідає формату" below it. The second field is labeled "Пароль" and contains three dots. A red border and a red exclamation mark icon indicate an error, with the message "Пароль повинен містити не менше 5 символів" below it. At the bottom of the form is a large orange button with the text "АВТОРИЗАЦІЯ".

Рисунок 3.5 – Валідація полів на екрані входу в систему

3.3 Реєстрація

Реєстрація в додатку для онлайн бронювання медичного обладнання є важливим кроком для користувачів, які бажають отримати доступ до широкого спектру медичного обладнання та використовувати його відповідно до своїх потреб. Цей процес реєстрації дозволяє створити обліковий запис користувача в додатку, що відкриває доступ до функціоналу бронювання та використання медичного обладнання зручним та ефективним способом.

Реєстрація в додатку передбачає розподіл користувачів на декілька ролей, що надає їм відповідні функціональні можливості та доступ до різних функцій

додатку. Це дозволяє враховувати потреби та обмеження різних категорій користувачів для забезпечення більш гнучкого та персоналізованого досвіду. В передбачені такі ролі як: пацієнт(простий користувач), лікар(лікар повинен погоджувати оренду медичного обладнання свого пацієнта), завідувач медичного обладнання(працівник лікарні, який завідує медичним обладнанням лікарні) та постачальник медичного обладнання(людина, яка буде виставляти на продаж або оренду медичне обладнання).

Варто зазначити, що якщо користувач реєструється за роллю постачальника медичного обладнання, він повинен окремо пройти верифікацію своєї ідентичності та ділових документів, для повного відкриття функціоналу додатку. Цей процес верифікації має на меті перевірити достовірність та законність постачальника, щоб вони могли виставляти своє медичне обладнання на продаж або оренду іншим користувачам.

Під час верифікації, постачальник може бути запрошений надати певні документи, такі як ліцензії на надання медичних послуг, сертифікати безпеки та якості обладнання, рекомендації чи довідки від попередніх клієнтів тощо. Ці документи перевіряються додатковою командою або адміністраторами, які гарантують, що постачальник відповідає стандартам якості та безпеки.

Після успішної верифікації, постачальник медичного обладнання отримує повний доступ до функціоналу додатку, включаючи можливість розміщувати своє обладнання на продаж або оренду, встановлювати ціни, визначати умови використання та здійснювати управління своїм інвентарем.

Цей процес верифікації гарантує безпеку та надійність для клієнтів, які шукають медичне обладнання, а також сприяє створенню довіри та високої якості послуг в рамках додатку для онлайн бронювання медичного обладнання.

На цьому екрані присутні поля для введення особистих даних, таких як ім'я, прізвище, електронна пошта, пароль, підтвердження пароля та номер телефону, а також поле для вибору ролі та кнопка для завершення реєстрації (рис. 3.6).

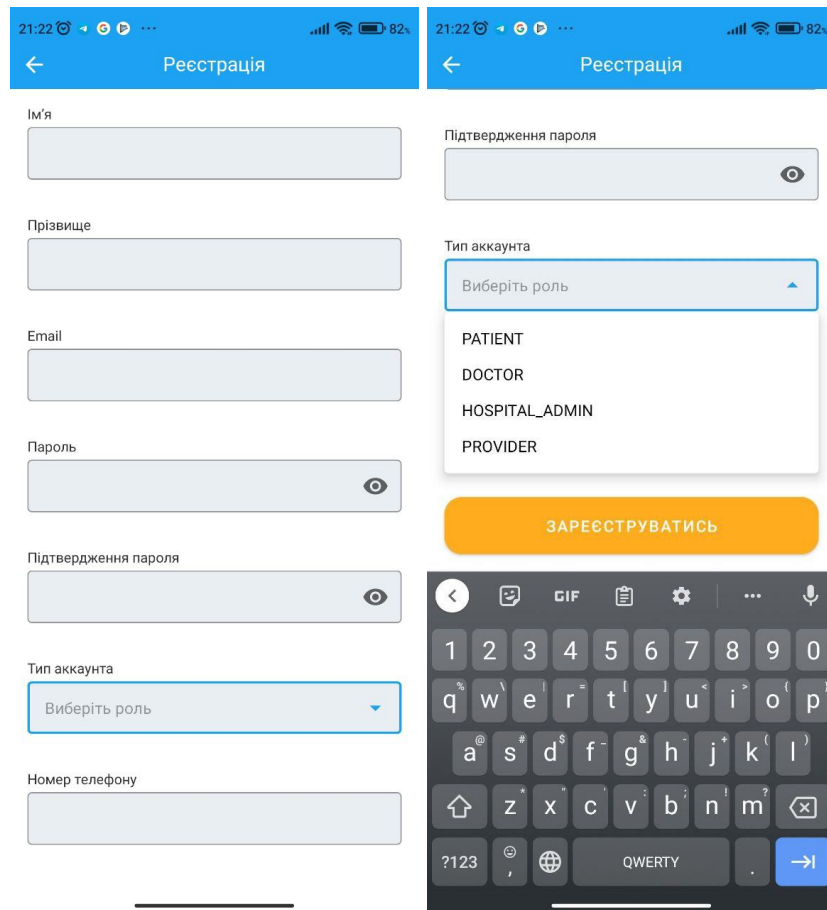


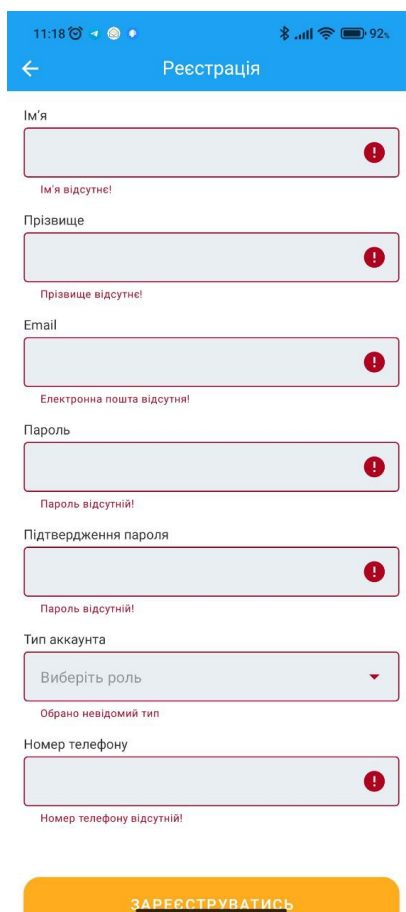
Рисунок 3.6 – Екран реєстрації та випадаючий список з типами аккаунта

Опис полів на екрані реєстрації:

1. Поле введення імені: Користувач вводить своє ім'я за допомогою цього поля.
2. Поле введення прізвища: Користувач вводить своє прізвище за допомогою цього поля.
3. Поле введення пошти: Користувач вводить свою електронну пошту, яка використовуватиметься як унікальний ідентифікатор облікового запису.
4. Поле введення пароля: Користувач вводить пароль для свого облікового запису. Пароль має відповідати вимогам безпеки, таким як мінімальна довжина та використання різних символів.
5. Поле підтвердження пароля: Користувач повторно вводить пароль для підтвердження, що введений пароль правильний і не містить помилок.
6. Поле вибору ролі: Користувач обирає свою роль в системі, наприклад, "Клієнт" або "Постачальник". Це поле дозволяє користувачеві визначити, які функції та можливості будуть доступні після реєстрації.

7. Поле введення номера телефону: Користувач вводить свій номер телефону, який може використовуватися для забезпечення зручності спілкування або додаткової перевірки ідентичності.

Також на екрані реєстрації присутня валідація для всієї введеної інформації користувачем. Валідація на екрані реєстрації наведена на рис. 3.7.



The screenshot shows a mobile application interface for registration. At the top, there is a blue header with a back arrow and the text "Реєстрація". Below the header, there are several input fields, each with a red exclamation mark icon indicating a validation error. The fields and their error messages are: "Ім'я" (Ім'я відсутнє!), "Прізвище" (Прізвище відсутнє!), "Email" (Електронна пошта відсутня!), "Пароль" (Пароль відсутній!), "Підтвердження пароля" (Пароль відсутній!), "Тип аккаунта" (Обрано невідомий тип), and "Номер телефону" (Номер телефону відсутній!). At the bottom, there is a yellow button with the text "ЗРЕЄСТРУВАТИСЬ".

Рисунок 3.7 – Валідація на екрані валідації

3.4 Головна сторінка

Головна сторінка додатку для онлайн бронювання медичного обладнання є ключовим місцем, де користувачі отримують огляд і доступ до основного функціоналу. На цій сторінці можна знайти перелік найпопулярніших товарів, а також рекомендованих постачальників обладнання. Вона пропонує зручну навігацію та швидкий доступ до основних функцій додатку.

На головній сторінці відображається список популярних товарів,

рекомендовані постачальники, поле пошуку.

Список популярних товарів: На головній сторінці відображається перелік самих популярних товарів медичного обладнання. Цей список дозволяє користувачам швидко ознайомитися з найбільш затребуваними та популярними продуктами.

Рекомендовані постачальники обладнання: Головна сторінка також надає можливість переглянути рекомендованих постачальників медичного обладнання. Це можуть бути надійні та перевірені постачальники, які пропонують високоякісне обладнання та мають гарну репутацію.

Швидкий доступ до сторінки пошуку: З головної сторінки можна безпосередньо перейти на сторінку пошуку. Це дозволяє користувачам швидко знайти конкретне медичне обладнання за певними параметрами, такими як назва, категорія, виробник тощо.

Головна сторінка додатку для онлайн бронювання медичного обладнання створена з метою забезпечення зручного та швидкого доступу до найпопулярніших товарів та рекомендованих постачальників. Вона також спрощує пошук потрібного обладнання шляхом прямого переходу на сторінку пошуку (рис. 3.8).

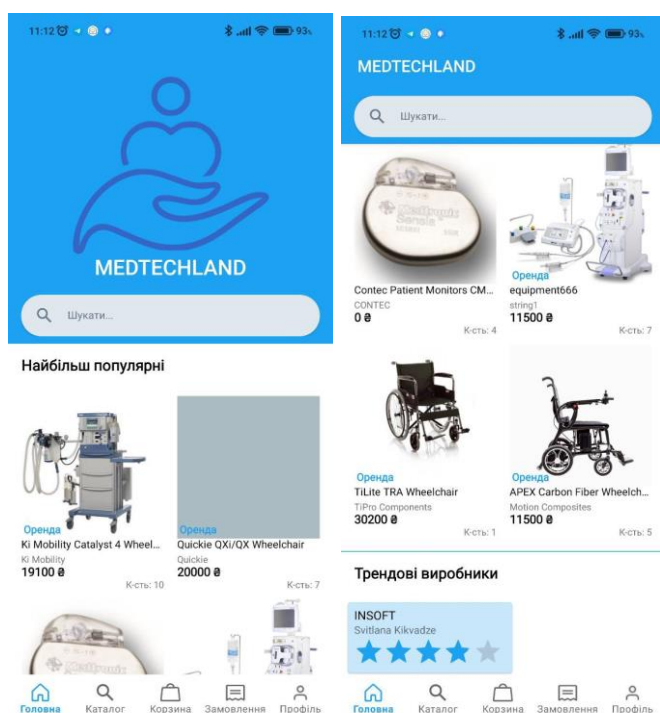


Рисунок 3.8 – Екран головної сторінки

Для покращення візуальної складової головної сторінки додатку був використаний `parallax` ефект, який активується при скролі сторінки. Цей ефект додає візуальну глибину та динаміку до інтерфейсу. При скролі картинка, що знаходиться на сторінці, поступово стає прозорою, дозволяючи з'явитися нижчерозташованим елементам. Заголовок екрану, який зазвичай містить назву додатку, поступово зсувається вліво, створюючи візуальний ефект руху.

Для реалізації цього ефекту була використана сучасна методика побудови UI для андроїд додатків, а саме `CoordinatorLayout`. `CoordinatorLayout` є розширеним контейнером, який дозволяє впроваджувати різноманітні ефекти та інтерактивність у вигляді взаємодії між елементами і їх розташуванням. Використання `CoordinatorLayout` дозволяє створити `parallax` ефект, а також інші цікаві візуальні ефекти та анімацію на головній сторінці додатку.

Цей підхід до побудови інтерфейсу додатку додає естетику, привабливість та більш інтерактивний досвід для користувачів. Візуальні ефекти, які реалізовані за допомогою `parallax` ефекту та `CoordinatorLayout`, створюють більш живу та привабливу головну сторінку додатку для онлайн бронювання медичного обладнання.

Варто зазначити, що весь контент, що відображається на головній сторінці, кешується за допомогою бібліотеки `Room`, яка забезпечує локальне зберігання та управління даними в базі даних. Це дозволяє додатку оперативно завантажувати та відтворювати збережений контент без необхідності постійного отримання даних з сервера.

Кешування контенту здійснюється з метою поліпшення швидкості завантаження та реагування додатку, а також для економії мобільного трафіку та зменшення навантаження на сервер. Бібліотека `Room` дозволяє зберігати та оновлювати дані локально, що сприяє плавній та безперебійній роботі додатку, навіть при відсутності інтернет-з'єднання.

Крім того, списки на головній сторінці відображаються за допомогою компонента `RecyclerView`. `RecyclerView` є потужним і гнучким інструментом для відображення списків у розширюваний спосіб. Він ефективно керує відображенням

великої кількості елементів, оптимізує використання ресурсів та надає можливості для анімації та взаємодії з користувачем. Використання RecyclerView дозволяє плавно прокручувати та оновлювати списки на головній сторінці, забезпечуючи зручне та ефективно взаємодію з контентом.

Застосування бібліотеки Room та компонента RecyclerView покращує продуктивність, швидкість та зручність використання головної сторінки додатку для онлайн бронювання медичного обладнання, забезпечуючи користувачам зручну та плавну навігацію та відображення контенту.

3.4 Пошук

Пошук є однією з головних функцій додатку для онлайн бронювання медичного обладнання. Ця функція надає користувачам можливість швидко та зручно знайти потрібне медичне обладнання за певними критеріями. Пошук дозволяє ефективно переглядати широкий асортимент обладнання, відповідно до індивідуальних потреб користувача.

При використанні функції пошуку, користувач може ввести ключові слова, категорії, параметри або інші фільтри, які відповідають його вимогам та специфікаціям обладнання. Додаток швидко обробляє ці дані і відображає релевантні результати, які відповідають пошуковому запиту користувача.

Пошукова функція є надзвичайно важливою, оскільки дозволяє користувачам зекономити час та знайти потрібне обладнання без зайвих зусиль. Вона сприяє швидкому з'ясуванню наявності, характеристик та цін на обладнання в різних постачальників. Пошукова функція також полегшує процес порівняння різних варіантів та вибір найбільш підходящого обладнання для конкретних потреб користувача.

3.4.1 Каталог

На екрані каталогу медичного обладнання користувачам надається зручний спосіб перегляду та пошуку різних категорій обладнання (рис. 3.9).



Рисунок 3.9 – Екран каталогу медичного обладнання

Екран включає такі елементи:

1. Поле пошуку: Натискання на поле пошуку відкриває головний екран пошуку, де користувачі можуть введення критеріїв пошуку для знаходження потрібного обладнання.
2. Список категорій: Представлення всіх доступних категорій обладнання у вигляді картинок та назв. Користувачі можуть прокручувати список та вибрати категорію, що відповідає їхнім потребам та інтересам.

Натиснувши на будь-яку категорію в списку відбувається перехід на екран пошуку, де будуть відображені результати для вибраної категорії обладнання. Це дозволяє користувачам швидко переходити до пошуку конкретного типу обладнання, спрощуючи процес навігації.

Екран каталогу медичного обладнання забезпечує зручну навігацію та швидкий доступ до різних категорій обладнання. Користувачі можуть шукати обладнання загалом або переходити до пошуку за конкретною категорією,

використовуючи відповідні елементи на екрані. Це забезпечує зручну та ефективну спосіб пошуку необхідного медичного обладнання.

3.4.2 Сторінка пошуку

Функція пошуку в додатку для онлайн бронювання медичного обладнання має п'ять станів, які відображають різні етапи та результати процесу пошуку.

1. Початковий стан: Цей стан активується, коли користувач вперше зайшов на екран пошуку. На цьому етапі користувач може ввести критерії пошуку, такі як ключові слова, категорії чи параметри обладнання. Початковий стан зображено на на рис. 3.10.
2. Початковий стан після запиту: Після виконання запиту та перегляду результатів, на екрані відображається історія запитів користувача. Це дає можливість користувачеві швидко повернутися до попередніх пошукових запитів та переглянути їх результати. Початковий стан після запиту зображено на на рис. 3.10.

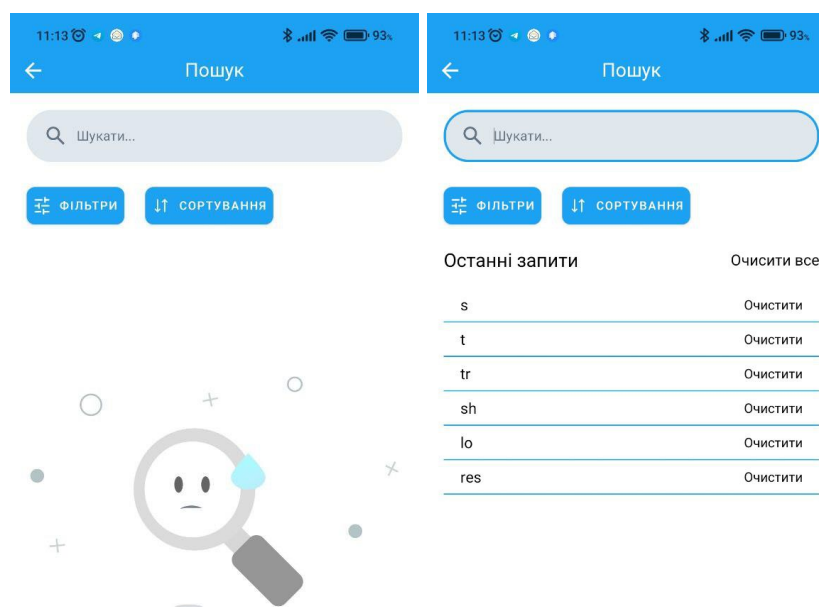


Рисунок 3.10 – Початковий стан та початковий стан після запиту пошуку

3. Процес пошуку: Коли користувач вводить пошуковий запит та натискає кнопку пошуку, відбувається анімація, що відображає процедуру пошуку. Цей етап показує, що додаток активно шукає відповідні результати до введених критеріїв. Процес пошуку зображено на рис. 3.11.
4. Стан пустого результату: Якщо пошук не дав результатів відповідно до введених критеріїв, відображається анімація або повідомлення, які показують, що не вдалося знайти жодного обладнання, що відповідає запиту користувача. Стан пустого результату зображено на рис. 3.11.

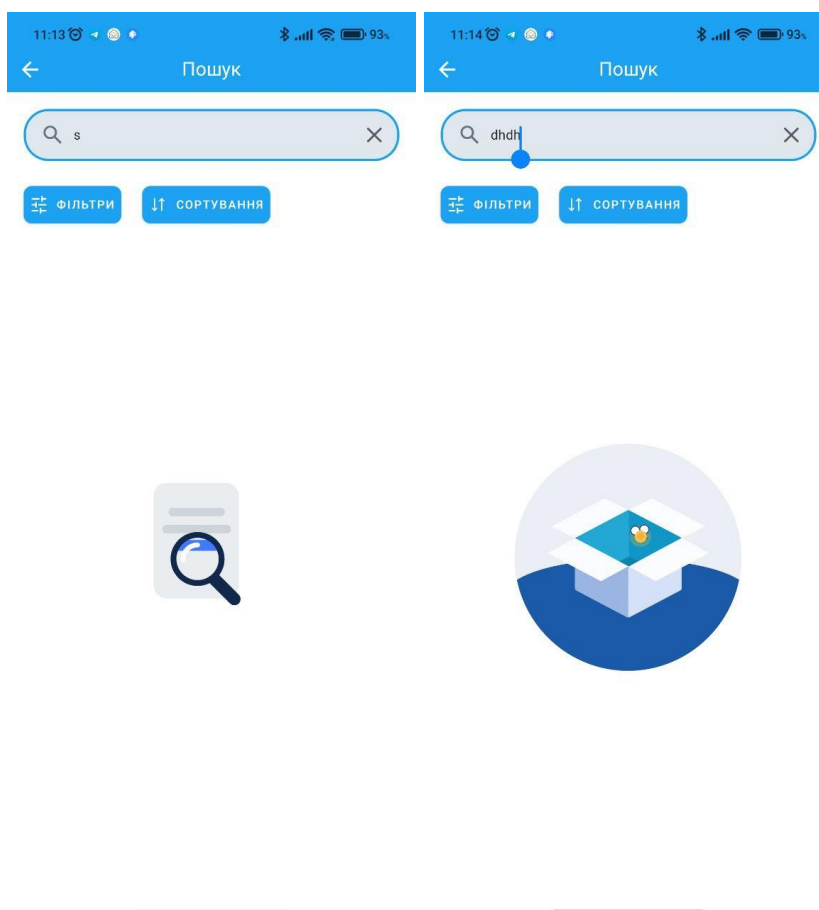


Рисунок 3.11 – Стан процесу пошуку та стан пустого результату

5. Стан результату: Після завершення пошуку відображається список з результатами, який відповідає пошуковому запиту користувача. У цьому стані користувач може переглядати та вибирати підходящі обладнання з отриманого списку. Стан результату зображено на рис. 3.12.

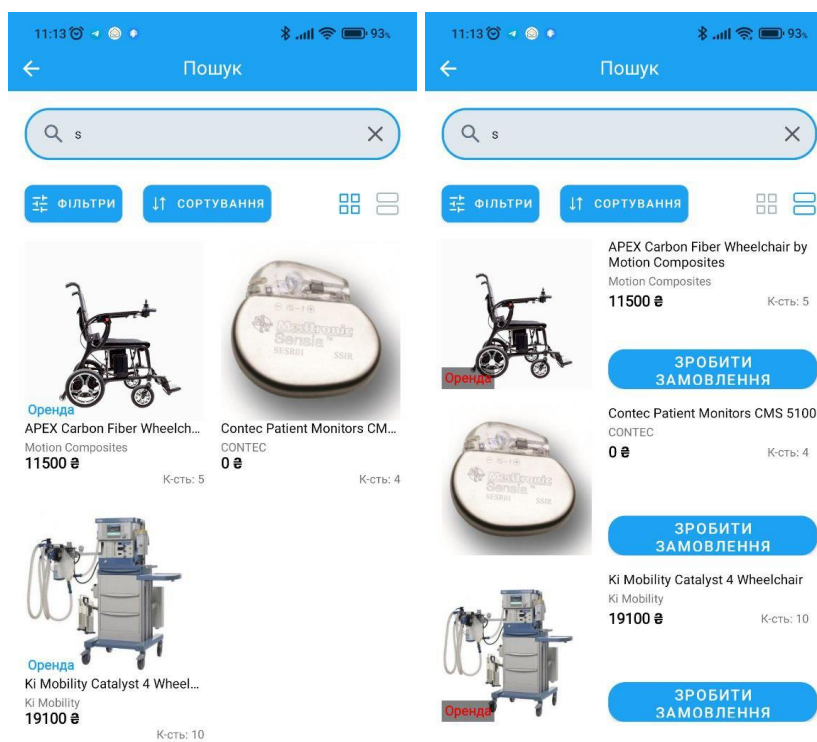


Рисунок 3.12 – Стан результату пошуку з сіткою 2 на 2 та лінійного режиму

Ці п'ять станів функції пошуку забезпечують користувачам зручну та інтуїтивно зрозумілу навігацію, а також повідомляють про стан процесу пошуку та результати пошуку. Вони допомагають користувачам ефективно здійснювати пошук медичного обладнання згідно їхніх вимог та потреб. Користувачі можуть легко вводити свої критерії пошуку і переглядати результати, які найкраще відповідають їхнім вимогам.

Екран пошуку в додатку реалізовано з використанням патерну State Machine, який забезпечує ефективне керування та перемикання станів пошуку. Цей патерн базується на ідеї, що процес пошуку може мати різні стани, такі як початковий стан, стан процесу пошуку, стан результатів та стан пустого результату. За допомогою State Machine, додаток може легко переходити між цими станами відповідно до дій користувача та результатів пошуку.

Кожен стан має свої властивості та поведінку, які відповідають його функціональності. У стані процесу пошуку може відобразитися анімація чекання,

а кнопка пошуку може бути вимкнена. У стані результатів відображаються відповідні результати пошуку, а у стані пустого результату показується відповідне повідомлення про відсутність результатів.

Застосування патерна State Machine у екрані пошуку допомагає створити зручний та інтуїтивно зрозумілий досвід для користувачів додатку. Він дозволяє ефективно керувати станами пошуку та забезпечує зручну навігацію та взаємодію з результатами пошуку.

Крім того, стани пошуку відображаються за допомогою візуальних елементів та анімації, що покращує взаємодію та користування додатком. Показання процесу пошуку та результатів допомагають користувачам зрозуміти, що відбувається в додатку та які результати можна очікувати.

Анімації в додатку для онлайн бронювання медичного обладнання відображаються за допомогою бібліотеки Lottie. Завдяки Lottie, розробники можуть використовувати анімації, створені за допомогою інструментів дизайну, таких як Adobe After Effects, і інтегрувати їх безпосередньо в додаток. Це дозволяє створювати багатогранні та динамічні анімації, які поліпшують візуальний ефект інтерфейсу та надають йому виразності та привабливості.

Стан результату на головній сторінці пошуку може відобразитися у двох режимах: за допомогою динамічної зміни сітки, яка включає сітку 2 на 2 та лінійний режим. Ці режими надають користувачам можливість вибору оптимального способу відображення результатів, враховуючи їхні вподобання та потреби.

У режимі сітки 2 на 2, елементи результатів пошуку відображаються у вигляді компактних плиток, розташованих у формі сітки. Цей режим забезпечує зручний та швидкий перегляд більшої кількості елементів на одному екрані. Користувач може швидко прокручувати та сканувати результати, отримуючи загальний огляд доступної інформації про обладнання.

У лінійному режимі, елементи результатів відображаються у вигляді вертикального списку, де кожен елемент займає більше місця на екрані. Цей режим надає користувачу більш деталізовану інформацію про кожен елемент результату. Користувач може бачити більшу кількість деталей та описів обладнання.

Динамічна зміна сітки відображення результатів дозволяє користувачам адаптувати вигляд інтерфейсу до своїх вподобань та потреб. Вони можуть змінювати режим відображення залежно від кількості результатів та рівня деталізації, який їм зручний у даному контексті.

Таким чином, використання режимів сітки 2 на 2 та лінійного режиму в стані результату пошуку надає гнучкість та варіативність користувачам, дозволяючи їм підлаштувати вигляд додатку до своїх вподобань та комфорту в сприйнятті.

3.4.3 Фільтрація

У додатку для онлайн бронювання медичного обладнання доступна функція фільтрації, яка дозволяє користувачам обмежити вибір обладнання за різними критеріями. При фільтрації, користувачі можуть встановлювати такі параметри, як ціна, тип обладнання та тип замовлення - купівля, оренда або обидва разом. Екран фільтрації представлений на рис. 3.13.

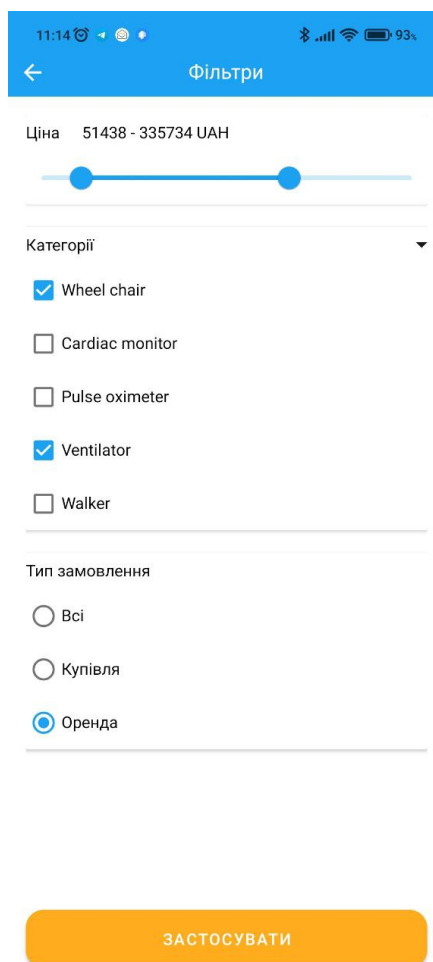


Рисунок 3.13 – Екран фільтрації пошуку

Фільтр "Ціна" дозволяє користувачам встановити максимальну та/або мінімальну ціну, яку вони готові платити за медичне обладнання. Це дозволяє користувачам звужити вибір і побачити тільки обладнання, що відповідає їхньому бюджету.

Фільтр "Тип обладнання" дозволяє користувачам обрати конкретний тип обладнання, яке вони шукають. Наприклад, вони можуть вибрати типи обладнання, такі як УЗД апарати, рентгенівське обладнання або кардіографи. Це дозволяє користувачам зосередитися на певних категоріях обладнання, що відповідають їхнім потребам та вимогам.

Фільтр "Тип замовлення" дозволяє користувачам вибрати, чи шукають вони обладнання для купівлі, оренди або обидвох варіантів одночасно. Це дає користувачам можливість знайти обладнання, яке відповідає їхнім потребам щодо типу транзакції.

Користувачі можуть використовувати один або кілька фільтрів одночасно для звуження результатів пошуку та відображення тільки тих варіантів обладнання, які відповідають їхнім вимогам. Це дозволяє зекономити час та зосередитися на найбільш підходящих варіантах медичного обладнання.

Екран фільтрації в додатку для онлайн бронювання медичного обладнання надає користувачам можливість налаштувати параметри фільтрації для точного підбору обладнання за їхніми потребами. На цьому екрані присутні такі елементи:

1. Слайдер ціни: Цей елемент дозволяє користувачам встановити діапазон ціни, використовуючи покажчики на слайдері. Вони можуть перетягувати покажчики, щоб встановити максимальну та мінімальну ціну, яку вони готові платити за обладнання.
2. Список з чекбоксами для типів медичного обладнання: На екрані присутній список з чекбоксами, де користувачі можуть вибрати потрібні типи медичного обладнання. Вони можуть активувати чекбокси, щоб відфільтрувати результати тільки за вибраними типами обладнання.
3. Радіо група з вибором типу замовлення: Цей елемент дозволяє користувачам вибрати тип замовлення - оренду, купівлю або обидва разом. Вони можуть

обрати відповідний варіант за допомогою радіо кнопок.

4. Кнопка застосування фільтрів: На екрані присутня кнопка, яку користувачі можуть натиснути після встановлення відповідних параметрів фільтрації. Після натискання кнопки, процес застосування фільтрів активується.

Процес застосування фільтрів використовує `FragmentManager`, що дозволяє передавати результати фільтрації між фрагментами. Після натискання кнопки застосування фільтрів, введені параметри фільтрації передаються до відповідного фрагменту, який обробляє ці параметри та оновлює список результатів відповідно до встановлених фільтрів.

3.4.4 Сортування

В додатку для онлайн бронювання медичного обладнання доступна функція сортування, що дозволяє користувачам організувати список результатів за різними критеріями. Сортування виконується за такими критеріями, як назва, ціна та дата створення замовлення. Крім того, користувачі можуть вибрати порядок сортування - від найменшого до найбільшого або навпаки.

Для зручності використання, сортування реалізовано за допомогою модального вікна, яке відображається на основі `BottomSheetDialogFragment`. При виборі опції сортування, користувачі можуть вибрати необхідний критерій та порядок за допомогою списку критеріїв та перемикача порядку. Модальне вікно сортування представлено на рис. 3.14.

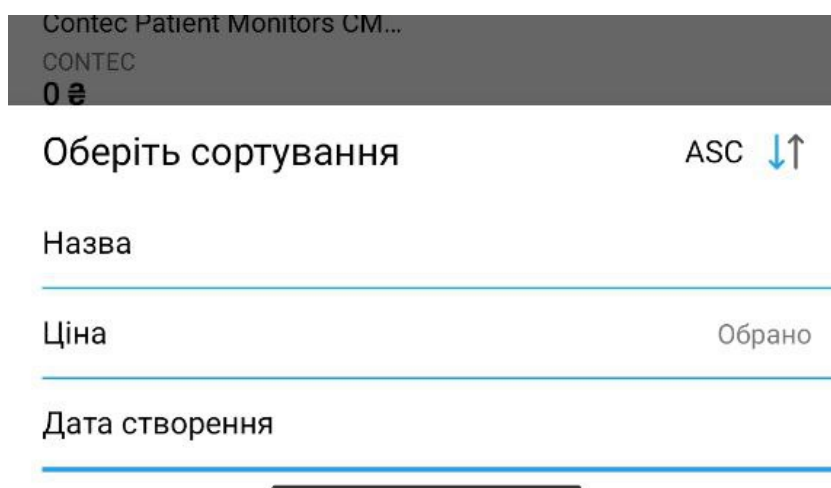


Рисунок 3.14 – Модальне вікно сортування результатів пошуку

Після вибору параметрів сортування та порядку, додаток застосовує ці параметри до списку результатів та відображає їх у вибраному порядку. Це дозволяє користувачам легко налаштувати спосіб сортування, щоб швидко знайти найбільш відповідні результати за їхніми вимогами.

Функція сортування забезпечує користувачам більшу гнучкість та контроль над відображенням результатів, допомагаючи зорганізувати їх у зручний спосіб. Вона полегшує процес пошуку та вибору медичного обладнання, дозволяючи швидко знайти найбільш відповідні варіанти за певними критеріями сортування.

3.5 Публікація медичного обладнання

У додатку для онлайн бронювання медичного обладнання постачальники мають можливість виставити своє обладнання на продаж або оренду. Ця функціональність дозволяє постачальникам активно залучати клієнтів та забезпечувати доступ до широкого спектру медичних пристроїв та обладнання.

Постачальники можуть використовувати додаток для створення профілю, в якому вони можуть описати свою компанію, представити асортимент медичного обладнання та встановити ціни та умови оренди або продажу. Вони також можуть додати фотографії та деталі про кожен пристрій, щоб надати користувачам детальну інформацію для вибору.

Користувачі додатку, у свою чергу, мають можливість переглядати список постачальників та їхнє доступне обладнання. Вони можуть ознайомитися з описом, фотографіями, цінами та умовами оренди або продажу. Користувачі можуть здійснювати бронювання або покупку обладнання, спілкуючись безпосередньо з постачальниками через додаток.

Важливо зазначити, що постачальники медичного обладнання в додатку повинні надати необхідні документи про оренду або продаж покупцям, які виявили інтерес до їх товару. Це включає документи, які підтверджують легальність та відповідність обладнання стандартам і вимогам, а також умови оренди або продажу.

Надання потрібних документів має важливе значення для покупців, оскільки це дозволяє їм переконатися в надійності та якості обладнання, а також узгодити умови та деталі транзакції. Покупці мають право отримати документацію, яка засвідчує правомірність володіння та використання обладнання, а також відповідність умовам, описаним постачальником.

Ці документи можуть включати орендні або продажні договори, сертифікати якості, гарантійні папери, інструкції з експлуатації та будь-яку іншу необхідну документацію, яка підтверджує правомірність та якість обладнання. Надання цих документів забезпечує довіру між постачальниками та покупцями, а також сприяє прозорості та відповідальності в угодах щодо медичного обладнання.

Екран для публікації медичного обладнання зображено на рис. 3.15.

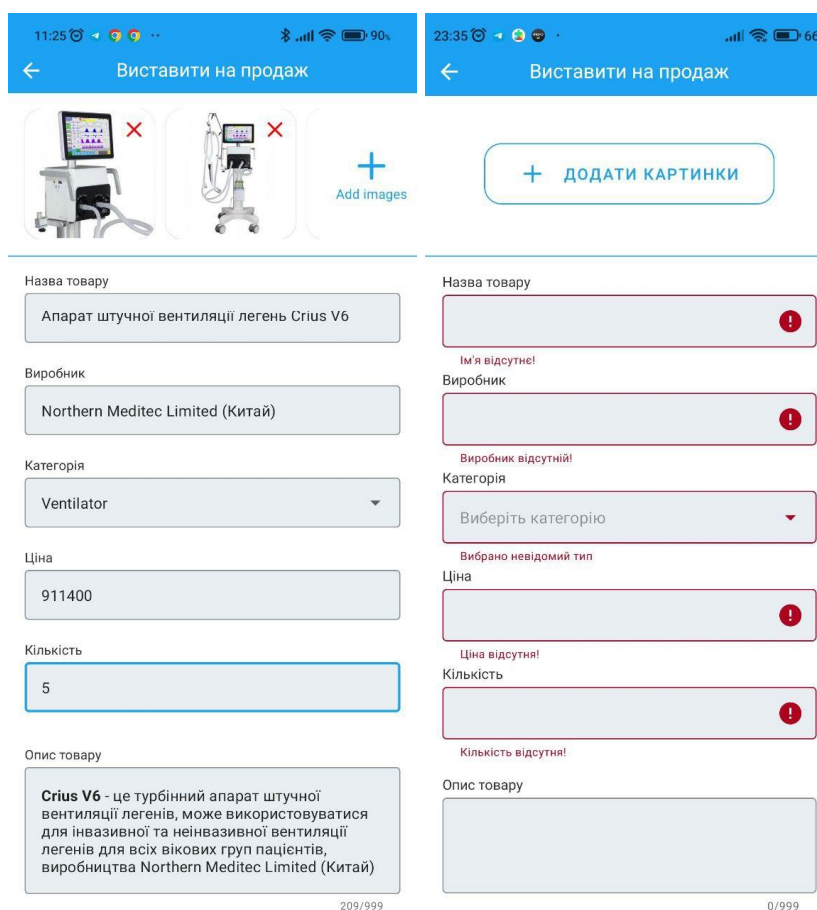


Рисунок 3.15 – Екран публікацію приладу на продаж чи оренду

На екрані публікації медичного обладнання користувачам надається можливість ввести необхідну інформацію та завантажити зображення з галереї для

представлення товару. Екран містить наступні поля та елементи:

1. Вибір зображень: Користувачі можуть обрати до 5 зображень зі своєї галереї, натиснувши на кнопку для вибору фотографій. Це дозволяє додати візуальну репрезентацію обладнання для публікації.
2. Поле для назви: Користувачі можуть ввести назву медичного обладнання, яке вони бажають опублікувати. Поле підтримує валідацію для перевірки коректності введених даних.
3. Поле для виробника: Користувачі можуть вказати виробника медичного обладнання. Це поле також підлягає валідації для забезпечення правильності введених даних.
4. Вибір категорії: Користувачі можуть обрати категорію, до якої належить медичне обладнання, за допомогою відповідного випадаючого списку.
5. Поле для вказання ціни: Користувачі можуть ввести ціну, за яку пропонується медичне обладнання. Це поле також має валідацію, щоб переконатися, що коректно введено числові дані.
6. Поле для кількості: Користувачі можуть вказати доступну кількість обладнання для продажу або оренди. Це поле також підлягає валідації, щоб перевірити коректність введених даних.
7. Поле для опису: Користувачі можуть ввести детальний опис медичного обладнання, включаючи його особливості та характеристики. Це поле обмежено до 999 символів, щоб забезпечити компактність та чіткість опису.

Після введення всіх необхідних даних, користувачі мають можливість натиснути кнопку "Публікувати" для збереження та надсилання даних на сервер. Зображення товару, які були вибрані з галереї, будуть завантажені на файловий хостинг Imgur, що забезпечує зберігання та доступ до них через URL адреси.

Після публікації, інформація про медичне обладнання буде доступна для перегляду іншим користувачам в додатку. Вони зможуть шукати та переглядати товари згідно зі своїми потребами та вимогами, знаходячи детальну інформацію та зображення, які були надані постачальниками.

3.6 Сторінка медичного приладу

Екран медичного обладнання в додатку є центральним місцем, де користувачі можуть ознайомитися з різноманітним асортиментом медичних пристроїв. Цей екран створений з метою надати користувачам повну інформацію про кожний прилад, щоб допомогти їм зробити обґрунтований вибір (рис. 3.16).



Рисунок 3.16 – Екран деталей медичного обладнання

На екрані медичного обладнання користувачі зможуть переглянути зображення приладу, ознайомитися з його назвою, ціною, виробником та іншими характеристиками. Детальний опис допоможе користувачам розібратися в функціональних особливостях та перевагах кожного пристрою.

Додаткова інформація, така як кількість доступних пристроїв, постачальник, контактна інформація та можливість оренди, надають користувачам додаткові деталі, які допоможуть у прийнятті рішення.

Кнопки для покупки та додавання до кошика забезпечують зручну навігацію

та можливість безпосередньо здійснити покупку або зберегти обладнання для подальшого розгляду.

Екран медичного обладнання в додатку є надійним джерелом інформації, що допомагає користувачам зробити важливе рішення стосовно медичних пристроїв, що відповідають їхнім потребам та вимогам.

На екрані медичного обладнання представлено широкий спектр інформації, що допомагає користувачам детально ознайомитися з кожним конкретним пристроєм. Екран містить наступні елементи:

1. **Зображення медичного приладу:** Зображення графічно відображає вигляд та характеристики приладу, надаючи користувачам загальне уявлення про його зовнішній вигляд та функціональні можливості.
2. **Назва приладу:** Унікальне ідентифікаційне позначення приладу, яке допомагає користувачам зрозуміти, про який саме медичний пристрій йдеться.
3. **Ціна:** Вказана вартість приладу, що допомагає користувачам оцінити його доступність та вирішити, чи він відповідає їхньому бюджету.
4. **Виробник:** Інформація про компанію, яка виробила прилад, що може мати значення для користувачів, що віддають перевагу певним брендам або мають попередній досвід з продукцією виробника.
5. **Тип приладу:** Вказує на функціональне призначення та класифікацію приладу, допомагаючи користувачам зорієнтуватися в різних категоріях медичного обладнання.
6. **Опис:** Детальний опис приладу, включаючи його основні характеристики, переваги та особливості. Ця інформація допомагає користувачам зрозуміти, як прилад може задовольнити їхні потреби та вимоги.
7. **Кількість:** Вказує на кількість доступних одиниць даного приладу, що може вплинути на рішення користувача щодо покупки або оренди.
8. **Постачальник:** Інформація про компанію або особу, яка постачає цей прилад. Це може бути важливо для користувачів, які мають впевненість в якості та надійності постачальника.

9. Контактна інформація: Номер телефону та адреса компанії, що дозволяє користувачам зв'язатися з постачальником для отримання додаткової інформації або уточнення деталей.

10. Кнопка для купівлі та додавання до кошика: Кнопки, що надають можливість користувачам придбати прилад або додати його до свого кошика для подальшої покупки.

Екран медичного обладнання забезпечує детальну інформацію та зручні функціональність для користувачів, щоб допомогти їм прийняти обґрунтоване рішення та здійснити вдалу покупку медичного обладнання.

3.7 Інформування про помилки

Обробка помилок є важливою складовою будь-якого додатку, включаючи наш додаток. Під час взаємодії з сервером або при виконанні певних дій можуть виникати різноманітні помилки, які необхідно адекватно обробляти, щоб забезпечити зручну та безперебійну роботу для користувачів.

Однією з типових ситуацій помилок є помилки під час виконання запитів до сервера. Це може статися через проблеми з мережевим з'єднанням, недоступністю сервера або некоректною відповіддю зі сторони сервера. У таких випадках, для зручності користувача, відображається модальне вікно зі заголовком помилки, який чітко вказує на характер помилки, описом, який роз'яснює причину помилки, а також кнопкою "Закрити", яка дозволяє користувачеві повернутися до попереднього стану додатку.

Крім того, іншою типовою ситуацією помилок є помилки пов'язані з авторизацією користувача. Якщо користувач намагається виконати дії, які вимагають авторизації, але він не авторизований або авторизація не вдалася, відображається відповідне повідомлення про помилку, яке повідомляє користувача про необхідність авторизуватися для доступу до цих функцій.

Загальний підхід до обробки помилок у нашому додатку полягає в тому, щоб чітко і зрозуміло повідомляти користувачам про виниклі помилки, надавати

достатньо інформації для розуміння причини помилки та надавати зручні засоби для їх вирішення. Модальне вікно помилки зображено на рис. 3.17.

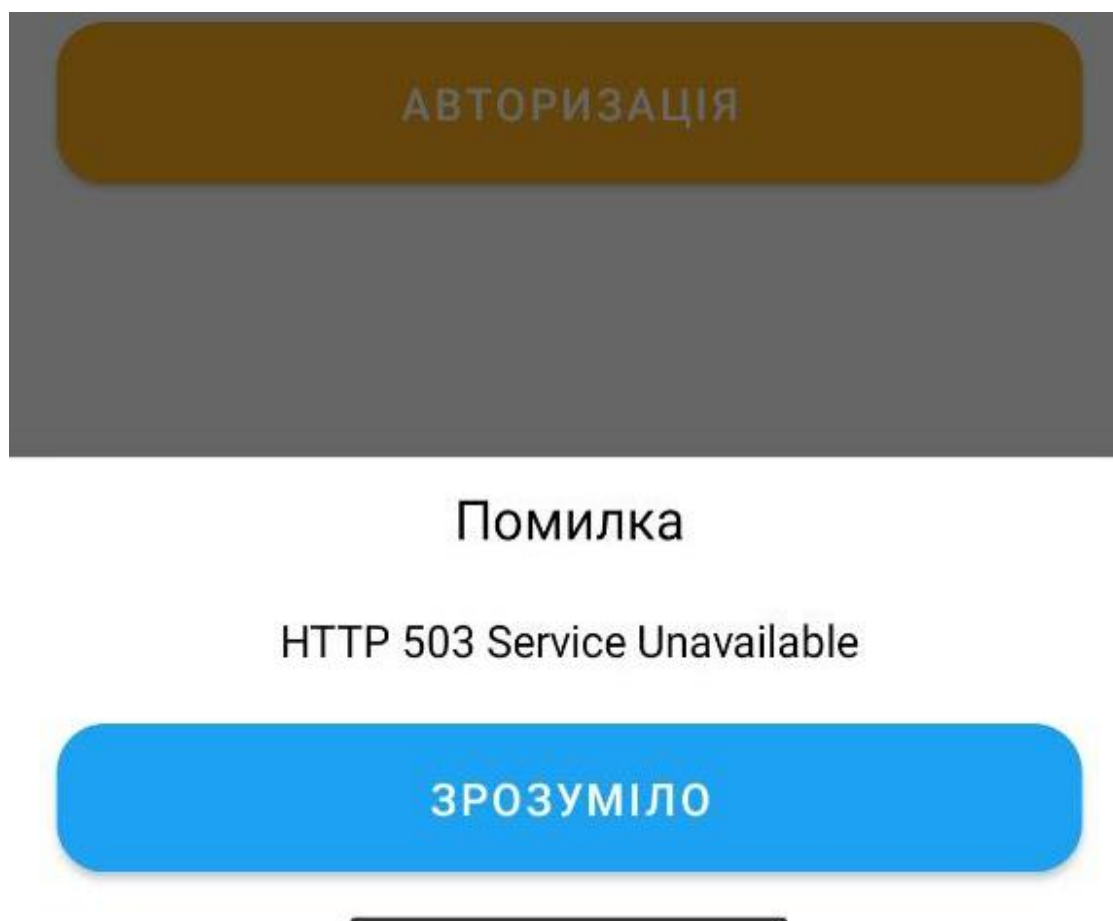


Рисунок 3.17 – Модальне вікно помилки

3.7 Розрахунок економічного ефекту

У даному розділі було проведено детальний аналіз програмного продукту, який призначений для Android додатку для онлайн бронювання медичного обладнання. З метою кращого розуміння функціональності програмного продукту, була розроблена морфологічна карта, яка дозволила систематизувати основні функції та визначити взаємозв'язки між ними.

На рис. 3.18 наведено морфологічну карту програмного продукту, який призначений для онлайн бронювання медичного обладнання.

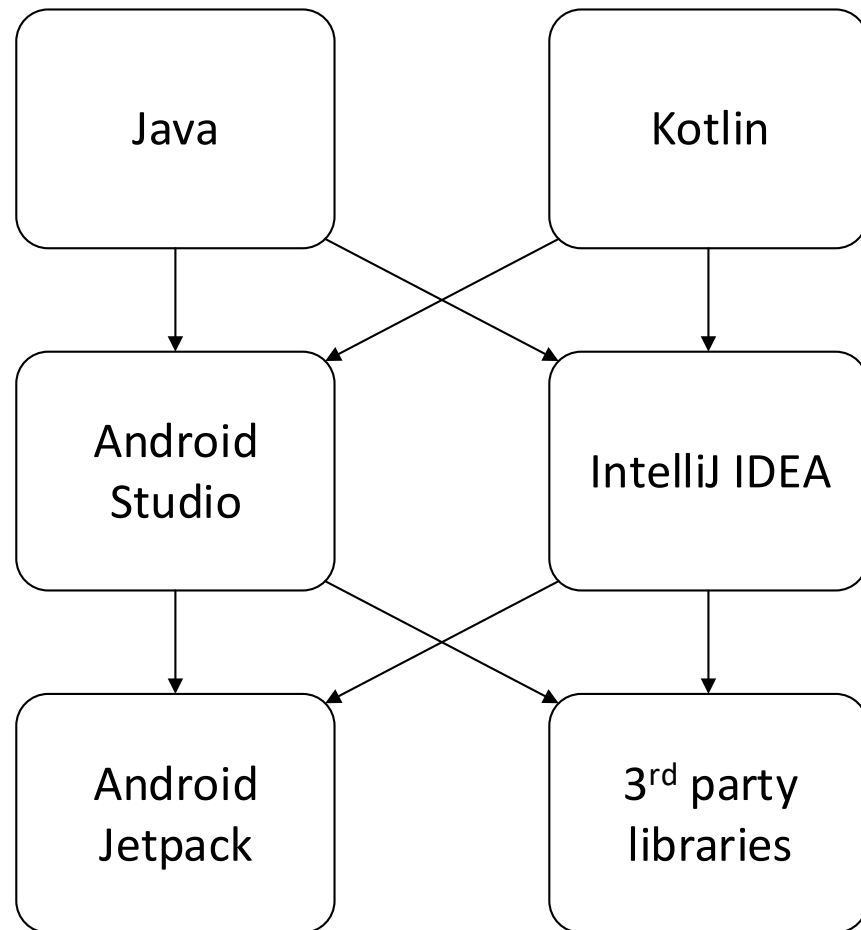


Рисунок 3.18 - Морфологічна карта варіантів реалізації функцій

Після ретельного вивчення економічного аспекту були узагальнені теоретичні знання, які мають важливе значення для ефективного використання їх в техніко-економічному обґрунтуванні програмних продуктів, застосовуючи метод функціонально-вартісного аналізу.

В ході аналізу функцій програмного продукту було виявлено чотири варіанти реалізації, серед яких найбільш перспективним виявився третій варіант, що мав найвищий коефіцієнт техніко-економічного рівня.

В результаті цей варіант передбачає:

1. Використання мови програмування Kotlin
2. Використання середовища розробки Android Studio
3. Використання набору бібліотек Android Jetpack

Витрати на реалізацію додатку складають 755 421.97 грн.

Висновки до розділу 3

У рамках третього розділу нашої дипломної роботи було проведено детальне дослідження та аналіз основних аспектів, пов'язаних з розробкою Android додатків, а також реалізації додатку для онлайн бронювання медичного обладнання.

Одним із головних висновків розділу є вибір платформи Android як основної для нашого додатку. Android є однією з найпоширеніших та найпопулярніших мобільних платформ, що надає широкі можливості для розробки та впровадження додатків на ринку. Вибір Android дозволяє залучити велику аудиторію користувачів та отримати значний ринковий потенціал.

Також було вирішено обрати мову програмування Kotlin для реалізації додатку. Kotlin є сучасною, розширеною та ефективною мовою програмування, що надає багато переваг у порівнянні з традиційною мовою Java. Використання Kotlin дозволяє писати більш компактний та зрозумілий код, зменшує кількість помилок та полегшує розробку та підтримку додатку.

Для побудови додатку було використано сучасний стек бібліотек, який включає Retrofit для роботи з мережевими запитами, Room для локального кешування даних, Coil для завантаження та відображення зображень, Lottie для анімацій та інші потужні інструменти. Цей стек допомагає забезпечити ефективну та швидку роботу додатку, забезпечуючи необхідний функціонал та візуальну привабливість.

Для розробки додатку було обрано Android Studio як основне середовище розробки. Android Studio є потужним та розширеним інструментом, спеціально розробленим для розробки Android додатків. Воно надає широкий набір функціональних можливостей, таких як налагоджувальні інструменти, емулятори пристроїв, інтеграція зі сторонніми бібліотеками, автоматична підказка коду та багато іншого. Використання Android Studio сприяє зручній та ефективній розробці, допомагає зберегти час та зусилля розробників.

У результаті роботи було створено сучасний, зручний та інтуїтивно

зрозумілий додаток для онлайн бронювання медичного обладнання. Додаток має основні функціональні можливості, такі як пошук, авторизація, перегляд медичного обладнання та виставлення на продаж або оренду. Було приділено особливу увагу деталям дизайну, використовуючи сучасні елементи і зручність для користувачів.

Всі основні функції додатку були успішно реалізовані, забезпечуючи зручну навігацію, швидкий доступ до інформації та зручне взаємодію з медичним обладнанням. Застосування правильної платформи, мови програмування та сучасного стеку бібліотек дозволило нам досягти високої якості та ефективності розробки.

ЗАГАЛЬНІ ВИСНОВКИ

В результаті виконання дипломної роботи було розроблено мобільний додаток під платформу Android для онлайн бронювання медичного обладнання. Даний додаток надає користувачам зручний та ефективний спосіб замовлення та оренди медичного обладнання через мобільні пристрої.

1. Аналіз джерел. Під час роботи було проведено аналіз вітчизняних та зарубіжних джерел, що стосувалися різних способів створення мобільних додатків на різних платформах. Отримані знання про різноманітні підходи та методи були узагальнені та використані для реалізації програмного додатку.

2. Аналіз існуючих аналогів. Також був проведений аналіз існуючих аналогів на ринку мобільних додатків для онлайн бронювання медичного обладнання. В процесі дослідження були виявлені деякі недоліки та обмеження в існуючих рішеннях, такі як неінтуїтивний інтерфейс користувача, обмежений функціонал або незадовільна швидкість роботи додатку.

Отриманий аналіз став основою для визначення переваг та унікальних особливостей розробленого додатку. Було зроблено акцент на створенні зручного, інтуїтивного та ефективного інтерфейсу користувача, а також на розширеному функціоналі, який включає різні опції пошуку, авторизації, перегляду та виставлення на продаж або оренду медичного обладнання.

3. Аналіз сучасних практик та технологій. У процесі розробки були використані сучасні практики та технології, що дозволило створити функціонально-багатий та естетично привабливий додаток. Вибір основної платформи падав на Android, яка є однією з найпопулярніших та широко використовуваних платформ для мобільних додатків.

4. Розробка мобільного додатку. Для реалізації програмного додатку були використані популярні фреймворки, бібліотеки та середовище розробки, що дозволило ефективно створити функціональність додатку та забезпечити його високу продуктивність та якість.

Усі поставлені цілі були досягнуті завдяки відповідальному підходу до

розробки, дослідженню сучасних технологій та використанню передових практик. Розроблений додаток для онлайн бронювання медичного обладнання показує себе як зручний, привабливий та інтуїтивно зрозумілий інструмент для користувачів. Його зручність полягає у широкому функціоналі, що дозволяє здійснювати пошук, авторизацію, перегляд та виставлення на продаж або оренду медичного обладнання виробниками.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Grant K. Beginning Android Programming: Develop and Design / K. Grant, C. Haseaman. – San Francisco: Peachpit Press, 2014. – 256 с. – (0321956567).
2. Keur C. Ios Programming: The Big Nerd Ranch Guide – Boston: Addison-Wesley Professional, 2017. – 461 с. – (6th edition). – (9780134682334).
3. Dabit N. React Native in Action / Nader Dabit. – New York: Manning, 2019. – 320 с. – (First Edition). – (1617294055).
4. Biessek A. Flutter for Beginners: An introductory guide to building cross-platform mobile applications with Flutter and Dart / Alessandro Biessek. – Birmingham: Packt Publishing, 2019. – 512 с. – (1788996089).
5. Stack Overflow [Електронний ресурс] – Режим доступу до ресурсу: <https://stackoverflow.com>.
6. Jemerov D. Kotlin in Action / D. Jemerov, S. Isakova. – New York: Manning, 2017. – 360 с. – (9781617293290).
7. Nudelman G. Android Design Patterns: Interaction Design Solutions for Developers / Greg Nudelman. – New Jersey: Wiley, 2013. – 559 с. – (1st edition). – (1118394151).
8. Vasic M. Mastering Android Development with Kotlin: Deep dive into the world of Android to create robust applications with Kotlin / Milos Vasic. – Birmingham: Packt Publishing, 2017. – 378 с. – (1788473698).
9. Clifton I. Android User Interface Design: Implementing Material Design for Developers / Ian Clifton. – Boston: Addison-Wesley Professional, 2015. – 448 с. – (2nd edition). – (0134191404).
10. Android Developer [Електронний ресурс] – Режим доступу до ресурсу: <https://developer.android.com/docs>.
11. Kotlin docs [Електронний ресурс] – Режим доступу до ресурсу: <https://kotlinlang.org/docs/home.html>.
12. Medium [Електронний ресурс] – Режим доступу до ресурсу: <https://medium.com/>.

13. Smyth N. Android Studio 4.0 Development Essentials - Kotlin Edition: Developing Android Apps Using Android Studio 4.0, Kotlin and Android Jetpack / Neil Smyth. – Raleigh: Payload Media, 2020. – 816 с. – (Kotlin Edition). – (1951442202).
14. C. Martin R. Clean Architecture: A Craftsman's Guide to Software Structure and Design / Robert C. Martin. – London: Pearson, 2017. – 432 с. – (1st Edition). – (0134494164)
15. Coil [Электронный ресурс] – Режим доступа до ресурсу: <https://coil-kt.github.io/coil/>.
16. Apple Developer [Электронный ресурс] – Режим доступа до ресурсу: <https://developer.apple.com>.
17. Android Programming: The Big Nerd Ranch Guide (3rd Edition) / Bill Phillips, Chris Stewart, Kristin Marsicano, Brian Hardy. - Boston: Big Nerd Ranch, 2019. - 624 p. - (9780134706054).
18. Professional Android 4 Application Development / Reto Meier. - Indianapolis: Wrox, 2012. - 864 p. - (1118102274).
19. Android Cookbook: Problems and Solutions for Android Developers / Ian F. Darwin. - Sebastopol: O'Reilly Media, 2017. - 838 p. - (1491977178).
20. Android Development with Kotlin / Marcin Moskala, Igor Wojda. - Birmingham: Packt Publishing, 2017. - 444 p. - (1787123682).
21. RESTful Web APIs: Services for a Changing World / Leonard Richardson, Mike Amundsen, Sam Ruby. - Sebastopol: O'Reilly Media, 2013. - 408 p. - (1449358063).
22. Retrofit: Love working with HTTP [Электронный ресурс] – Режим доступа до ресурсу: <https://square.github.io/retrofit/>
23. Android SQLite Essentials / Sunny Kumar Aditya. - Birmingham: Packt Publishing, 2014. - 166 p. - (1783282970).
24. Android High Performance Programming / Enrique López Mañas. - Birmingham: Packt Publishing, 2016. - 410 p. - (1785285385).
25. Head First Android Development / Dawn Griffiths, David Griffiths. - Sebastopol: O'Reilly Media, 2015. - 944 p. - (1449362184).

26. Android UI Design with XML: Tutorial Book - [Электронный ресурс] – Режим
доступу до ресурсу:
https://www.tutorialspoint.com/android/android_ui_design.htm