

Міністерство освіти і науки України
Національний університет «Одеська політехніка»
Інститут комп'ютерних систем
Кафедра інформаційних систем

Шепелєв Данило Дмитрович,
Андрієнко Володимир Володимирович,
Петріков Богдан Андрійович,
студенти групи AI-214

ДИСЦИПЛІНА
Програмування мобільних пристроїв

КУРСОВА РОБОТА
Розробка мобільного додатку з перегляду статистики відеогри «Dota 2»

Спеціальність:
122 Комп'ютерні науки

Освітня програма:
Комп'ютерні науки

Керівник:
Годовиченко Микола Анатолійович,
кандидат технічних наук, доцент

Одеса – 2024

ЗМІСТ

Анотація	3
Вступ	4
1 Огляд систем-аналогів та технологій їх розробки	6
1.1 Особливості використання Android для розробки застосунків	6
1.2 Огляд схожих Android застосунків.....	7
1.2.1 Android застосунок «Статистика Dota 2»	8
1.3 Формування вимог до основних функцій нашого Android застосунку	9
1.4 Огляд інформаційних технологій для розробки Android застосунку	10
1.4.1 Jetpack Compose Navigation	10
1.4.2 Retrofit.....	11
1.5 Висновки до першого розділу.....	13
2 Проектування Android застосунку зі статистикою Dota 2	14
2.1 Мета та задачі Android застосунку	14
2.2 Визначення функціональних вимог до Android застосунку	15
2.3 Формування користувацьких історій Android застосунку.....	16
2.4 Визначення нефункціональних вимог до Android застосунку	16
2.5 Висновки до другого розділу	16
3 Програмна реалізація Android застосунку зі статистикою.....	18
3.1 Керування вихідним кодом Android застосунку	18
3.2 Інструкція користувача Android застосунку	18
3.3 Вихідний код Android застосунку	23
3.4 Висновки до третього розділу.....	23
ВИСНОВКИ.....	25
ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ.....	26

АНОТАЦІЯ

Курсова робота присвячена розробці Android застосунку «DOTA Tracker». Цей Android застосунок дозволяє переглядати статистику окремого гравця за його SteamID та статистику його ігор. Основною метою роботи є створення застосунку, який дозволить всім любителям відеогри «Dota 2» переглядати свою статистику зі свого смартфона. В ході розробки будуть використані сучасні технології та практики розробки Android застосунків, зокрема мова програмування Kotlin, Compose Navigation, взаємодія з API через Retrofit тощо.

ABSTRACT

The coursework is devoted to the development of the Android application "DOTA Tracker". This Android application allows you to view the statistics of an individual player by their SteamID and the statistics of their games. The main goal of the work is to create an application that will allow all fans of the video game "Dota 2" to view their statistics from their smartphone. In the course of the development, modern technologies and practices of Android application development will be used, in particular, the Kotlin programming language, Compose Navigation, interaction with the API through Retrofit, etc.

ВСТУП

У сучасному світі, де мобільні додатки набувають все більшої популярності, зростає потреба у зручних та функціональних інструментах для геймерів. Одним із таких інструментів є додатки для перегляду статистики відеоігор. Відеогра "Dota 2" має велику кількість прихильників, які цікавляться детальною статистикою своїх ігор та ігор інших гравців. Саме для таких користувачів був створений мобільний Android додаток "DOTA Tracker".

"DOTA Tracker" – це мобільний Android застосунок, який дозволяє переглядати статистику окремого гравця за його SteamID. Застосунок надає можливість користувачам отримувати як загальну статистику гравця, так і детальну статистику кожної його гри. Це дозволяє гравцям аналізувати свої досягнення, визначати сильні та слабкі сторони, а також порівнювати свої результати з результатами інших гравців.

Реалізація додатку здійснюється за допомогою спеціального загальнодоступного Open Dota API, який забезпечує необхідну функціональність для отримання та обробки статистичних даних. Завдяки цьому додаток може швидко і точно надавати користувачам актуальну інформацію про їхні ігрові досягнення та статистику.

"DOTA Tracker" орієнтований на любителів відеогри "Dota 2", а також на тих, хто цікавиться аналізом та статистикою ігрових процесів. Мета цього проекту – створити зручний та інтуїтивний інструмент для відстеження ігрової статистики, який допоможе гравцям покращувати свої навички та досягати нових висот у грі.

Для досягнення цієї мети необхідно виконати наступні задачі:

- здійснити огляд існуючих додатків для перегляду статистики відеоігор та подібних проектів;
- проаналізувати та обрати сучасні технології для розробки Android застосунку;
- провести проектування "DOTA Tracker", включаючи інтерфейс користувача та функціональні можливості;

- виконати програмну реалізацію додатку "DOTA Tracker" з використанням Open Dota API.

Розробка такого проекту вимагає використання мови програмування Java, сучасних технологій Android та архітектурних принципів, що дозволяють створити стабільний, ефективний та відмінний продукт. Додаток повинен мати інтуїтивний інтерфейс, який дозволить користувачам легко взаємодіяти з ним та отримувати необхідну інформацію у зручному форматі.

1 ОГЛЯД СИСТЕМ-АНАЛОГІВ ТА ТЕХНОЛОГІЙ ЇХ РОЗРОБКИ

1.1 Особливості використання Android для розробки застосунків

Операційна система Android є потужним інструментом для розробки мобільних застосунків, що дозволяють переглядати та аналізувати різноманітні статистичні дані. Ось декілька особливостей використання Android для розробки статистичних застосунків:

- Широкий охоплення аудиторії: Android є однією з найпопулярніших операційних систем для мобільних пристроїв у світі з мільйонами активних користувачів. Розробка статистичного застосунку на Android дозволяє досягти великої аудиторії і забезпечити ефективний доступ до статистичних даних для багатьох користувачів.
- Простота розробки: Android надає розробникам потужні інструменти та SDK (Software Development Kit) для створення застосунків. Мови програмування, такі як Java та Kotlin, дозволяють розробникам легко створювати функціональні та ефективні статистичні застосунки. Android Studio, офіційна інтегрована середа розробки (IDE), надає зручні інструменти для написання, тестування та відлагодження коду.
- Гнучкість інтерфейсу користувача: Android дозволяє створювати застосунки з інтуїтивними та привабливими інтерфейсами користувача. Бібліотеки для побудови графіків та візуалізації даних, такі як MPAndroidChart, дозволяють розробникам легко інтегрувати різноманітні графіки та діаграми для відображення статистичної інформації у застосунках.
- Потужні можливості обробки даних: Android надає широкі можливості для обробки та аналізу великих обсягів даних. Розробники можуть використовувати бази даних, такі як SQLite, та інші інструменти для зберігання і обробки статистичних даних. Це дозволяє створювати

застосунки, які можуть ефективно працювати з великими масивами інформації.

- Інтеграція з іншими сервісами та API: Android дозволяє легко інтегрувати статистичні застосунки з різноманітними сторонніми сервісами та API. Наприклад, розробники можуть використовувати відкриті API для отримання статистичних даних з різних джерел або інтегрувати свій застосунок з іншими аналітичними платформами для розширення функціональності.
- Можливості сповіщень: Android надає можливості для надсилання сповіщень користувачам через застосунки. Це дозволяє розробникам створювати системи сповіщень для інформування користувачів про оновлення статистичних даних, важливі події або інші зміни.
- Безпека та конфіденційність: Android забезпечує високий рівень захисту даних користувачів та конфіденційності. Використання сучасних засобів безпеки, таких як шифрування даних та автентифікація користувачів, дозволяє розробникам створювати безпечні статистичні застосунки.

Таким чином, використання Android для розробки статистичних застосунків дозволяє створювати потужні, зручні та безпечні інструменти для аналізу та візуалізації даних, які можуть ефективно задовольняти потреби користувачів.

1.2 Огляд схожих Android застосунків

Для розробки Android застосунку з оглядом статистики Dota 2 необхідно з'ясувати, які функції повинні бути доступні в ньому. Для цього потрібно дослідити та проаналізувати схожого за задумкою застосунку. Це дозволить встановити особливості роботи цих застосунків та з'ясувати позитивні та негативні моменти їх використання. Також цей аналіз допоможе краще зрозуміти потреби користувачів в цій області. Після проведення пошуку в Інтернеті, був знайдений найбільш популярний та відомий застосунок:

– Android застосунок «Статистика Dota 2»;

Далі потрібно провести аналіз цього застосунку з метою встановлення особливостей його роботи та з'ясування позитивних та негативних моментів його використання. Це допоможе краще зрозуміти, які функції повинні бути доступні Android застосунку з переглядом статистики Dota 2, та відповісти на потреби користувачів в цій області.

1.2.1 Android застосунок «Статистика Dota 2»

Android застосунок «Статистика Dota 2» є зручним застосунком для перегляду статистики гравців відеогри «Dota 2» (рис. 1.2).

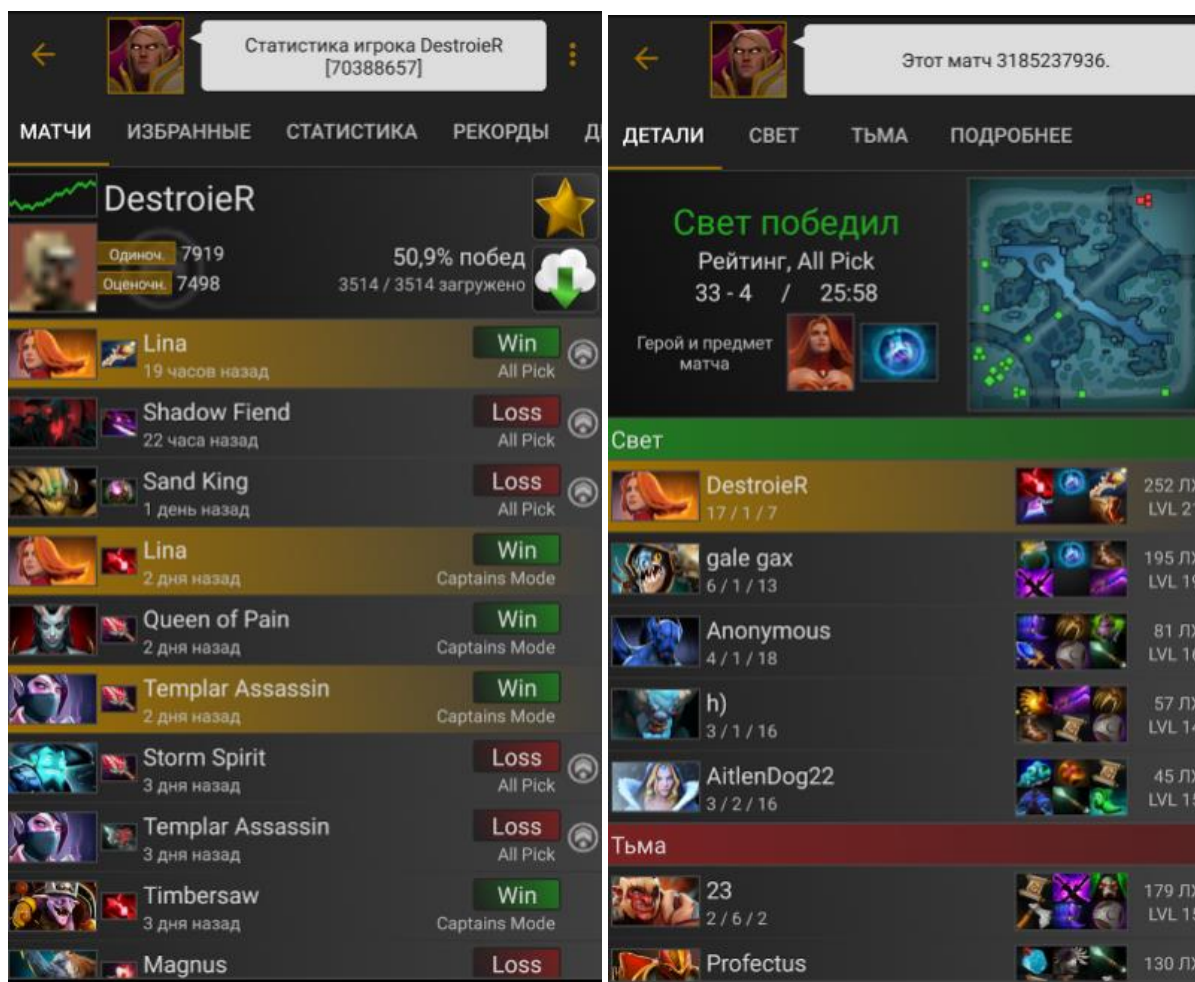


Рисунок 1.2 – Знімки екрану Android застосунку «Статистика Dota 2»
Основний функціонал додатку:

- повна історія матчів для будь-якого гравця, збережена на телефоні;
- настроювані фільтри, які дозволяють легко порівнювати гравців з різними комбінаціями героїв та предметів;
- можливість дивитись, які улюблені та найуспішніші герої та предмети;
- відсоток перемог та УСП;
- перегляд подробиці матчу та ознайомлення з предметом та гравцем матчу.

1.3 Формування вимог до основних функцій нашого Android застосунку

Аналіз Android застосунку «Статистика Dota 2» дозволив виділити його переваги та недоліки та сформуванати вимоги до власного Android застосунку.

Основний функціонал Android застосунку з переглядом статистики Dota 2 може включати наступні можливості:

- Перегляд загальної статистики гравця: Застосунок повинен дозволяти користувачам переглядати загальну статистику гравця, включаючи кількість зіграних матчів, перемоги, поразки, рейтинг та інші ключові метрики.
- Детальна статистика по кожній грі: Користувачі повинні мати можливість переглядати детальну статистику по кожному зіграному матчу. Це може включати кількість вбивств, смертей, допомог, вчинені дії, використані предмети та інші деталі.
- Історія матчів: Функція дозволяє користувачам переглядати історію їхніх матчів. Це включає перелік зіграних ігор з можливістю перегляду детальної статистики для кожного матчу.
- Рейтинг гравців: Користувачі можуть переглядати свої рейтинги та порівнювати їх з рейтингами інших гравців. Це може включати глобальні рейтинги, рейтинги по регіонам та друзям.
- Синхронізація з переглядом: Застосунок повинен забезпечувати синхронізацію з акаунтом Steam користувача який можна додати до фаворитів.

1.4 Огляд інформаційних технологій для розробки Android застосунку

1.4.1 Jetpack Compose Navigation

Jetpack Compose Navigation — це бібліотека від Google, яка надає інструменти для створення та керування навігацією в додатках Android, написаних за допомогою Jetpack Compose. Вона спрощує процес створення навігаційних графів і управління переходами між різними екранами у вашому застосунку.

Переваги Jetpack Compose Navigation:

- Нативна підтримка Compose: Jetpack Compose Navigation спроектована спеціально для використання з Jetpack Compose, що забезпечує плавну та ефективну інтеграцію.
- Декларативний підхід: Використання декларативного підходу Compose дозволяє визначати навігаційні маршрути чітко та зрозуміло.
- Спрощене управління навігацією: Навігаційні графи можна створювати безпосередньо в коді, що спрощує управління переходами між екранами.
- Вбудована підтримка аргументів: Підтримка передачі даних між різними екранами робить навігацію більш зручною та ефективною.
- Динамічне управління навігацією: Можливість динамічно змінювати навігаційний граф у відповідь на дії користувача або інші події.
- Робота з різними типами екранів: Підтримка різних типів екранів і переходів, включаючи модальні вікна та вкладені графи навігації.
- Підтримка ViewModel: Легка інтеграція з ViewModel для управління станом екранів.
- Життєвий цикл: Автоматичне управління життєвим циклом навігаційних компонентів.

Недоліки Jetpack Compose Navigation:

- Нова парадигма: Для розробників, які звикли до традиційної навігації в Android, може знадобитися певний час для адаптації до декларативного підходу Compose.

- Молода технологія: Jetpack Compose і Jetpack Compose Navigation є відносно новими технологіями, тому документація та приклади можуть бути не такими розгорнутими, як для традиційних підходів.
- Інтеграція з існуючими проектами: Може бути складніше інтегрувати Jetpack Compose Navigation в існуючі проекти, які вже використовують традиційну навігацію, без значних змін у кодовій базі.
- Потенційні баги: Як і з будь-якою новою технологією, можуть виникати баги та проблеми зі стабільністю, які ще не були виявлені або виправлені.

Jetpack Compose Navigation є потужним інструментом для управління навігацією в сучасних Android-застосунках, пропонуючи безліч переваг завдяки інтеграції з Jetpack Compose, але вимагає часу для освоєння та може стикатися з деякими проблемами, властивими новим технологіям.

1.4.2 Retrofit

Retrofit — це бібліотека для Android та Java, розроблена компанією Square, яка значно спрощує роботу з HTTP-запитами. Вона забезпечує простий та ефективний спосіб взаємодії з веб-сервісами, дозволяючи розробникам легко здійснювати HTTP-запити та обробляти відповіді. Retrofit використовує анотації для опису запитів і підтримує конвертацію даних у різні формати (JSON, XML) за допомогою конвертерів.

Переваги Retrofit:

- Анотації: Використання анотацій для опису HTTP-запитів робить код чистим і зрозумілим. Наприклад, для опису запиту GET або POST достатньо простої анотації над методом інтерфейсу.
- Автоматична конвертація даних: Retrofit підтримує автоматичну конвертацію JSON, XML та інших форматів у Java-об'єкти та навпаки, використовуючи бібліотеки конвертерів, такі як Gson або Moshi.
- Підтримка різних типів запитів: Retrofit підтримує всі основні типи HTTP-запитів, включаючи GET, POST, PUT, DELETE, PATCH та інші.

- **Можливість налаштування:** Легко налаштовується для роботи з різними API, підтримує обробку різних форматів даних і дозволяє легко додавати кастомні конвертери та адаптери.
- **Обробка асинхронних запитів:** Retrofit забезпечує просту реалізацію асинхронних HTTP-запитів, що дозволяє уникнути блокування основного потоку програми.
- **Інтеграція з OkHttp:** Retrofit використовує OkHttp для здійснення HTTP-запитів, що забезпечує високу продуктивність і додаткові можливості, такі як кешування, аутентифікація та інші.
- **Реактивне програмування:** Retrofit інтегрується з RxJava, що дозволяє використовувати реактивні підходи для обробки запитів і відповідей.
- **Підтримка корутин:** Також підтримує Kotlin Coroutines, що спрощує написання асинхронного коду в Kotlin.

Недоліки Retrofit:

- **Нові концепції:** Для розробників, які не знайомі з Retrofit, може знадобитися час на освоєння нових концепцій та підходів, таких як анотації та інтеграція з конвертерами.
- **Проблеми з великими об'ємами даних:** Обробка великих об'ємів даних може бути повільною або вимагати додаткових налаштувань для оптимізації продуктивності.
- **Залежність від інших бібліотек:** Retrofit покладається на інші бібліотеки, такі як OkHttp, Gson або Moshi, що збільшує кількість залежностей у проекті.
- **Стандартна обробка помилок:** Стандартні механізми обробки помилок можуть бути недостатніми для складних сценаріїв, що вимагатиме додаткової роботи від розробника для налаштування.

Retrofit є потужним і гнучким інструментом для взаємодії з веб-сервісами в Android та Java, який значно спрощує роботу з HTTP-запитами завдяки використанню анотацій та автоматичної конвертації даних. Проте, як і будь-яка інша бібліотека, Retrofit має свої недоліки, включаючи необхідність навчання та потенційні проблеми з продуктивністю при обробці великих об'ємів даних.

1.5 Висновки до першого розділу

У першому розділі курсової роботи був проведений огляд Android-застосунків зі схожими функціями та технологій їхньої розробки.

Було розглянуто питання особливостей використання Android для створення статистичних застосунків. Визначено, що платформа Android є потужним інструментом для розробки мобільних застосунків, які дозволяють користувачам отримувати доступ до різноманітної статистичної інформації. Особливості Android, такі як широке охоплення користувачів, гнучкість у розробці інтерфейсу та інтеграція з різними сервісами, роблять її ідеальною платформою для створення таких застосунків.

Далі був проведений огляд аналогічних застосунків для перегляду статистики Dota 2. Це дозволило визначити основні переваги та недоліки існуючих застосунків, встановити їхні ключові функції та можливості. На базі проведеного огляду були визначені основні вимоги до власного застосунку "DOTA Tracker", а також надано детальний опис цих вимог.

Було також проведено огляд інформаційних технологій для розробки Android-застосунку з переглядом статистики Dota 2. У якості мови програмування для розробки було вирішено обрати мову Kotlin, яка є офіційною мовою для розробки Android-застосунків і надає сучасні можливості та спрощує розробку. Для організації навігації в застосунку було вирішено використати Jetpack Compose Navigation, що дозволяє створювати гнучкі та інтуїтивні інтерфейси. Для здійснення HTTP-запитів та взаємодії з Open Dota API було обрано бібліотеку Retrofit, яка забезпечує зручний та ефективний спосіб обробки запитів та відповідей.

Таким чином, перший розділ курсової роботи надав детальний аналіз існуючих рішень та технологій, що дозволило визначити основні напрямки розробки застосунку "DOTA Tracker" та сформулювати вимоги до нього. Це забезпечує міцну основу для подальшої реалізації проекту, враховуючи виявлені переваги та недоліки аналогічних рішень, а також сучасні технології розробки.

2 ПРОЕКТУВАННЯ ANDROID ЗАСТОСУНКУ ЗІ СТАТИСТИКОЮ DOTA 2

2.1 Мета та задачі Android застосунку

Мета застосунку "DOTA Tracker" полягає в створенні зручного та функціонального інструменту для користувачів, що цікавляться відеогрою "Dota 2", який дозволить їм отримувати доступ до статистичної інформації про гравців та матчі у режимі реального часу. Головною метою є забезпечення користувачам зручного та швидкого способу відстежування своєї власної статистики та статистики інших гравців, а також отримання актуальної інформації про матчі.

Задачі застосунку "DOTA Tracker":

Отримання статистики гравців: Реалізувати можливість введення ідентифікатора гравця ("SteamID") та отримання детальної інформації про його статистику в грі "Dota 2", таку як загальна кількість перемог, програшів, середня кількість кіл та інше.

Перегляд історії матчів: Забезпечити можливість перегляду історії останніх матчів гравця, включаючи деталі кожного матчу, такі як результат, тривалість, кількість убивств та смертей.

Пошук матчів: Реалізувати функціонал пошуку матчів за різними параметрами, такими як рівень гравця, тип матчу, карта тощо.

Графічне відображення статистики: Реалізувати інтуїтивне та зрозуміле графічне відображення статистичних даних для зручного сприйняття користувачем.

Інтеграція з Open Dota API: Взаємодіяти з відкритим API Open Dota для отримання актуальних даних про гравців та матчі.

Оптимізація продуктивності та безпеки: Забезпечити оптимальну продуктивність та безпеку застосунку, використовуючи сучасні технології розробки та дотримуючись найкращих практик програмування для платформи Android.

2.2 Визначення функціональних вимог до Android застосунку

Визначення функціональних вимог є ключовим етапом у процесі розробки застосунку "DOTA Tracker". Ці вимоги конкретизують, які функції та можливості має мати додаток і що він повинен забезпечувати користувачам. Вони встановлюють чітку спрямованість розробки та дозволяють уникнути непорозумінь між розробниками та вимогами користувачів.

- Перегляд статистики гравців: Користувач може ввести ідентифікатор ("SteamID") гравця та переглянути детальну статистику про його гру у "Dota 2", таку як загальна кількість перемог, програшів, середня кількість кіл тощо.
- Перегляд історії матчів: Користувач може переглядати історію останніх матчів гравця, включаючи деталі кожного матчу, такі як результат, тривалість, кількість убивств та смертей.
- Графічне відображення статистики: Забезпечення інтуїтивного та зрозумілого графічного інтерфейсу для зручного сприйняття користувачем статистичних даних.
- Збереження налаштувань користувача: Застосунок повинен забезпечувати можливість збереження налаштувань користувача.
- Інтеграція з Open Dota API: Взаємодія з відкритим API Open Dota для отримання актуальних даних про гравців та матчі.
- Оптимізація продуктивності та безпеки: Забезпечення оптимальної продуктивності та безпеки застосунку, використовуючи сучасні технології розробки та дотримуючись найкращих практик програмування для платформи Android.

Відповідно до цих функціональних вимог, розробники зможуть створити повнофункціональний та корисний застосунок для любителів гри "Dota 2", який забезпечить їм зручний та доступний спосіб відстежування статистики гравців та матчів.

2.3 Формування користувацьких історій Android застосунку

Як користувач, я хочу мати можливість переглядати статистику гравців у грі "Dota 2", щоб отримати інформацію про їхні досягнення та результати гри.

Як користувач, я хочу мати можливість переглядати історію останніх матчів обраного гравця, щоб оцінити його гральну активність та успішність.

Як користувач, я хочу здійснювати пошук матчів, щоб знайти цікаву інформацію.

Як користувач, я хочу мати можливість зберігати свої налаштування, щоб персоналізувати додаток під свої потреби.

Як користувач, я хочу бачити графічне відображення статистики, щоб зручно аналізувати та порівнювати дані.

Як користувач, я хочу мати можливість взаємодії зі статистикою через інтуїтивний та зрозумілий інтерфейс, щоб використання застосунку було максимально зручним і простим.

2.4 Визначення нефункціональних вимог до Android застосунку

Після визначення функціональних вимог, необхідно сформулювати нефункціональні вимоги, які висуваються до Android застосунку.

NFR1 Пристрій – будь-який не досить старий Android пристрій.

NFR2 Інтернет-підключення – застосунок вимагає наявності активного Інтернет-підключення для використання.

2.5 Висновки до другого розділу

В другому розділі курсової роботи було проведено детальне проектування Android застосунку "DOTA Tracker" на базі сучасних технологій та архітектурних принципів для забезпечення стабільності, ефективності та відмінного користувацького досвіду.

Під час проектування визначено, що мета застосунку полягає у наданні користувачам можливості перегляду статистики гравців гри "Dota 2" за їх SteamID. Потенційною аудиторією є як фанати самої гри, так і ті, хто цікавиться статистикою гравців. Основні функції включають можливість перегляду загальної статистики гравця та статистики окремих ігор, а також використання спеціального загальнодоступного API для отримання цієї інформації.

На основі цього були встановлені основні функціональні та нефункціональні вимоги до застосунку. Серед них були вимоги до Інтернет-підключення та наявність Android-пристрою.

В цілому, проектування застосунку "DOTA Tracker" було спрямоване на створення зручного та функціонального інструменту для перегляду статистики гравців гри "Dota 2" з урахуванням потреб користувачів та сучасних стандартів розробки Android-додатків.

3 ПРОГРАМНА РЕАЛІЗАЦІЯ ANDROID ЗАСТОСУНКУ ЗІ СТАТИСТИКОЮ

3.1 Керування вихідним кодом Android застосунку

Використання системи контролю версій є дуже важливим для будь-якого програмного проекту, в тому числі і для Android застосунку. Використання системи контролю версій дозволяє зберігати історію всіх змін, які були внесені до веб-ресурсу.

Це дозволяє в разі потреби повернутися до попередніх проекту та відновити попередній стан, який був на момент попередньої версії.

Також, система контролю версій дозволяє контролювати версії Android застосунку, що значно полегшує роботу з проектом та дозволяє підтримувати його в актуальному стані.

Основні метрики репозиторію Android застосунку (https://github.com/Vo1maR/DOTA_TRACKER) наведені в таблиці 3.1.

Таблиця 3.1 – Метрики керування програмним кодом додатку

Android застосунок	Кількість комітів	Кількість pull-реквестів	Кількість гілок у репозиторії
	22	3	3

3.2 Інструкція користувача Android застосунку

Для забезпечення успішного користувацького досвіду використання розробленого Android застосунку необхідно скласти інструкцію користувача, яка включає в себе знімки екранів та пояснювальний текст до кожного знімка. Це допоможе користувачам легко зорієнтуватись у функціоналі Android застосунку та ефективно використовувати його.

Перше, що ми бачимо під час переходу в Android застосунку - це його початковий екран (рис. 3.1). Ми можемо побачити тут логотип гри «Dota 2», поле для введення steam id гравця, кнопку пошуку за введеним id та, нижче, «улюблені аккаунти», частина простіру екрану де з'являються аккаунти за якими ви слідкуєте.

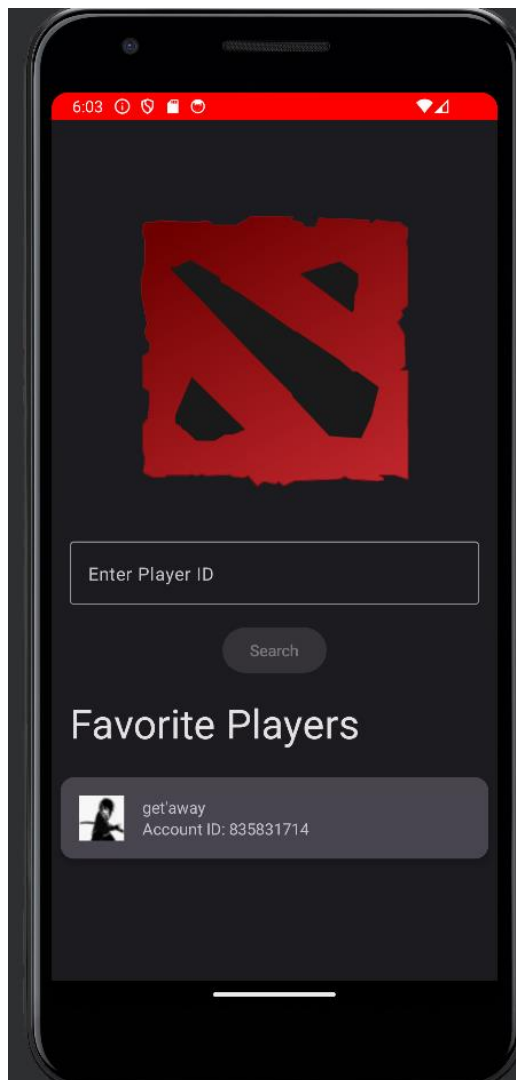


Рисунок 3.1 – Початковий екран.

Щоб перейти в наступний екран, потрібно буде ввести правильне id гравця або вибрати з тих що слідкуєте, перейдемо до наступного екрану – екрану профіля гравця. В цьому екрані (рис 3.2) можна побачити: зображення профілю, ім'я аккаунту, айді аккаунту, країну гравця, ранг у грі «Dota 2», де перша цифра: 1 – рекрут, 2 – страж, 3 – лицар, 4 – герой, 5 – легенда, 6 – властелин, 7 – божество, 8 – титан, а друга цифра означає кількість зірок на відповідному рангу. Також тут

наявне поле з кількістю перемог та поразок за останні 20 ігор. Наявні кнопки «Додати в/Прибрати з слідкуючих аккаунтів» та «Подивитись останні ігри». При натисканні першої кнопки цей аккаунт відповідно буде прибрано або додано зі слідкуючих. При натисканні на другу кнопку відбудеться перехід на наступний екран.

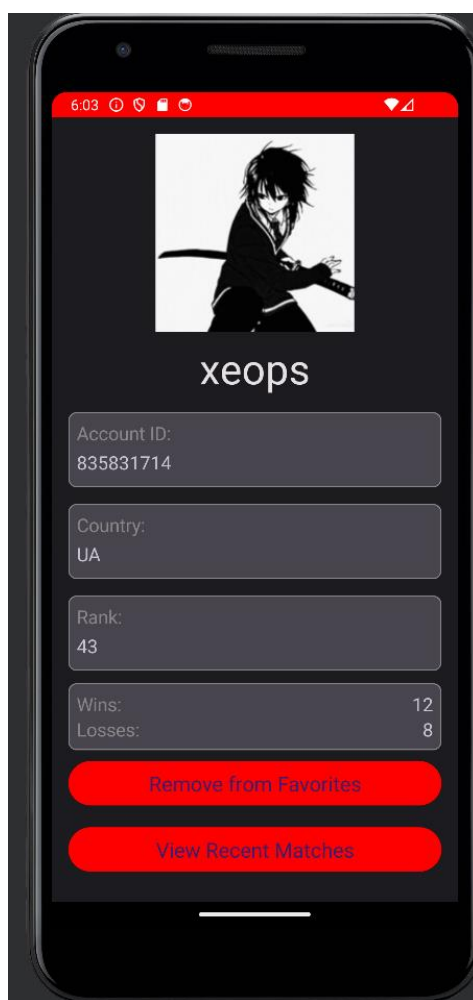


Рисунок 3.2 – Екран профіля.

Після того як ми вибрали кнопку «Подивитись останні ігри» нас перекидує на наступний екран – екран останніх ігор (рис. 3.3). Тут можна побачити статистику за останні ігри відповідного гравця, а саме: героя, статистику (kills/death/assists) та результат (Win/Lose). Цей екран можна гортати. При натисканні на будь-який з останніх матчів він буде підсвічений та відбудеться перехід на наступний екран.

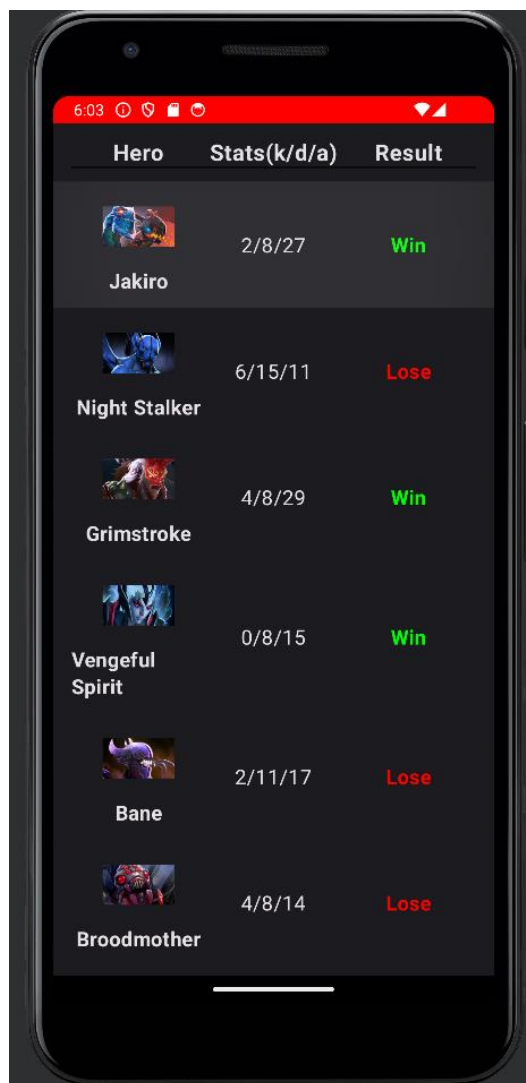


Рисунок 3.3 – Екран останніх ігор.

Переходимо до наступного екрану – екрану подробиць матчу (рис. 3.4). Тут можна побачити id гри, команду що здобула перемогу, загальний час цього матчу. Нижче є «таблиця» з ім'ям гравця, ім'ям героя, та статистика у вигляді k/d/a (kills/death/assists), нижче всі гравці розподілені за командами – Radiant та Dire з відповідними кольорами, також є зображення героя окрім його назви.

З цього екрану можна натиснути на якийсь профіль гравця у цьому матчі та здійсниться перехід на екран профілю.

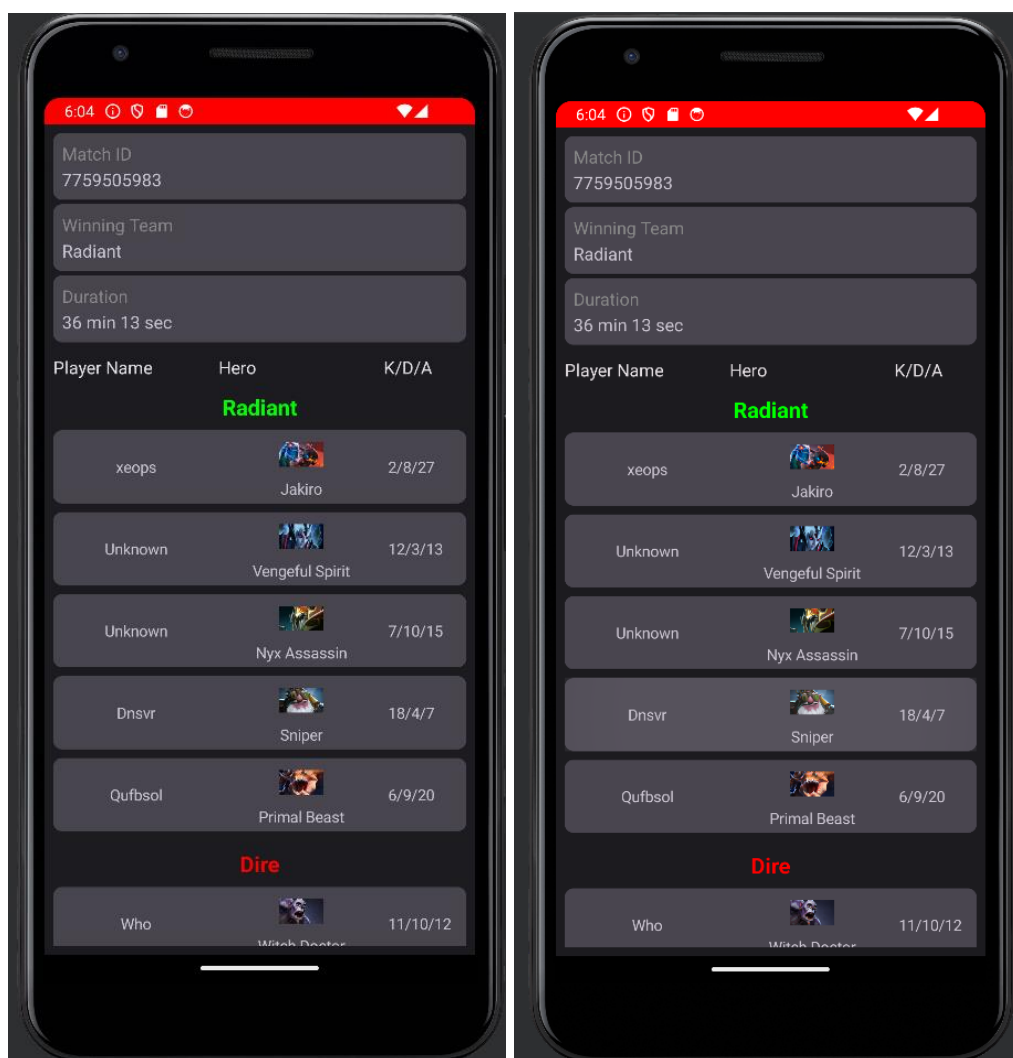


Рисунок 3.4 – Екран подробиць матчу до і після натискання по профілю.

Натиснімо наприклад на гравця з ім'ям «Dnsvr»(рис. 3.4).

Здійніться перехід на наступний екран – екран профілю гравця «Dnsvr»(рис. 3.5).

Тут як і раніше в екрані профілю «xeops» маємо ті ж самі поля, але з різною інформацією, також бачимо його зображення профілю. Можемо його додати до «Favorites»

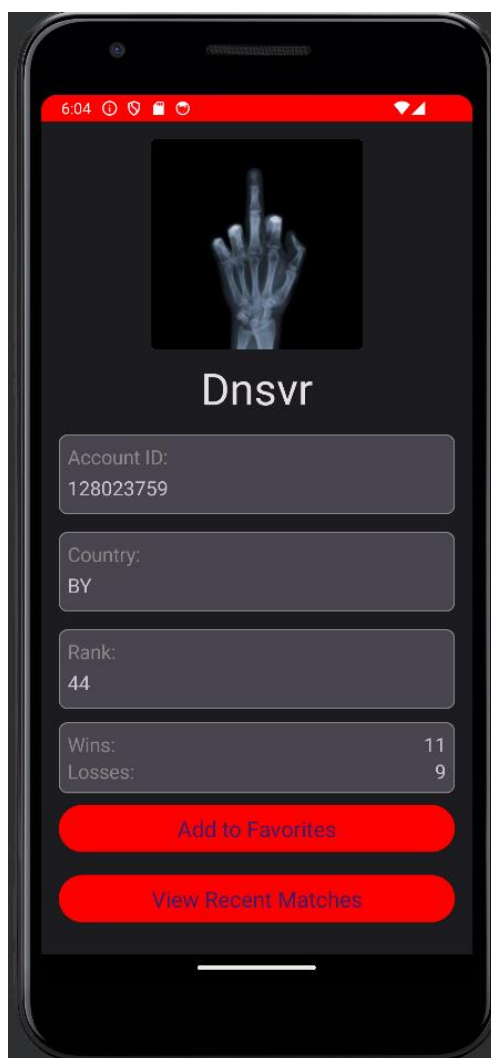


Рисунок 3.5 – Екран профілю гравця з «нікнеймом» Dnsvr.

3.3 Вихідний код Android застосунку.

Вихідний код програмних класів розробленого Android застосунку наведений на Github репозиторії даної роботи.

3.4 Висновки до третього розділу

В даному розділі курсової роботи було проведено програмну реалізацію Android застосунку «DOTA Tracker» на мові програмування Kotlin. Для того, щоб провести програмну реалізацію, був використаний проект Android застосунку, який був розроблений в другому розділі даної роботи.

Перш за все було розглянуто питання використання системи контролю версій для упорядкування розробки Android застосунку. Були визначені основні репозиторії проекту та надані основні показники метрик цих репозиторіїв.

По-друге, була розроблена інструкція користувача у вигляді множини знімків екрану та пояснювального тексту, який визначає функціонал додатку у даному вікна та можливі дії користувача. Крім того, наданий вихідний код розробленого Android застосунку.

ВИСНОВКИ

У даній курсовій роботі був розроблений Android застосунок «DOTA Tracker» для перегляду статистики гравців гри «Dota 2» за їх SteamID. Процес розробки включав детальне проектування, визначення функціональних та нефункціональних вимог, а також імплементацію застосунку з використанням сучасних технологій.

У першому розділі була проведена аналітична робота, де було детально вивчено сферу ігрової статистики та потреби користувачів у доступі до неї. Також було проведено огляд схожих застосунків та їх можливостей, що дозволило чітко визначити потреби аудиторії та основні функції, які має виконувати розроблений застосунок.

У другому розділі було проведено проектування застосунку. Була визначена його мета, аудиторія та основні можливості. На основі цього були встановлені функціональні та нефункціональні вимоги, що дозволяють краще зрозуміти взаємодію користувача з застосунком.

У третьому розділі було проведено програмну реалізацію Android застосунку «DOTA Tracker» на мові програмування Kotlin. Для того, щоб провести програмну реалізацію, був використаний проект Android застосунку, який був розроблений в другому розділі даної роботи.

Далі було проведено програмування застосунку. Застосовано сучасні технології, такі як Jetpack Compose для створення користувацького інтерфейсу та Retrofit для взаємодії з API. Розроблено функціонал для отримання та відображення статистики гравців з використанням інформації з Open Dota API тощо.

Загалом, розроблений застосунок відповідає поставленим вимогам та меті проекту, надаючи користувачам зручний та ефективний спосіб перегляду статистики гравців гри «Dota 2». Процес розробки був успішним завдяки використанню сучасних технологій та докладному аналізу потреб користувачів.

ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Retrofit. URL: <https://square.github.io/retrofit/> (дата звернення 15.04.2024).
2. Jetpack Compose Documentation. URL: <https://developer.android.com/jetpack/compose/documentation> (дата звернення 17.04.2024).
3. Official Android Developers Documentation. URL: <https://developer.android.com/docs> (дата звернення 26.04.2024).
4. ChatGPT. URL: <https://openai.com/blog/chatgpt> (дата звернення 27.04.2024).
5. YouTube Channels. URL: <https://www.youtube.com/> (дата звернення 28.04.2024).