

Міністерство освіти і науки України
Івано-Франківський національний технічний університет нафти і газу
Факультет інформаційних технологій
Кафедра комп'ютерних систем і мереж

Бояновський Олександр Вікторович

УДК 004.4

БАКАЛАВРСЬКА РОБОТА

Розробка iOS-додатку моніторингу фінансових активів із динамічною візуалізацією даних

(назва роботи)

Комп'ютерна інженерія

(назва освітньої програми)

123 - Комп'ютерна інженерія

(шифр і назва спеціальності)

Робота містить результати власних досліджень, використання ідей, результатів і текстів інших авторів мають посилання на відповідне джерело:

Здобувач освітнього ступеня Бояновський О.В.
(підпис, ініціали та прізвище здобувача)

Науковий керівник Лазорів А. М., к.т.н., доцент
(підпис, прізвище, ім'я, по батькові, науковий ступінь, вчене звання керівника)

Допущено до захисту
Завідувач кафедри КСМ

д.т.н., проф. _____ С.І. Мельничук
(посада) (підпис) (дата) (ініціали та прізвище)

Івано-Франківськ – 2025 рік

Івано-Франківський національний технічний університет нафти і газу

(повне найменування закладу вищої освіти)

Факультет інформаційних технологій

Кафедра комп'ютерних систем і мереж

Освітній рівень бакалавр

Спеціальність 123 – Комп'ютерна інженерія

(шифр і назва)

ЗАТВЕРДЖУЮ

Завідувач кафедри **КСМ**

(С.І. Мельничук)

«__» _____ 2025 року

З А В Д А Н Н Я

НА БАКАЛАВРСЬКУ РОБОТУ СТУДЕНТОВІ

Бояновському Олександру Віторовичу

(прізвище, ім'я, по батькові)

1. Тема роботи Розробка iOS-додатку моніторингу фінансових активів із динамічною візуалізацією даних.

керівник роботи Ларозів А.М., к.т.н., доцент.

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

затверджені наказом вищого навчального закладу від 05.05.2025 № 275/7

2. Термін подання студентом роботи 12 червня 2025 р

3. Вихідні дані до роботи Матеріали і результати отримані під час проходження переддипломної практики, методичні вказівки, технічна література.

4. Зміст розрахунково-пояснювальної записки (перелік питань, які потрібно розробити) 1. Аналіз аналогічних рішень та загальна характеристика додатку. 2. Розробка розробка ios-додатку моніторингу фінансових активів. 3. Перевірка працездатності розробленого ios-додатку моніторингу фінансових активів.

5. Перелік графічного матеріалу (з точним зазначенням обов'язкових креслень):

-

Нормоконтроль	Лазорів А.М.	
Антиплагіат	Ширмовська Н.Г	

6. Дата видачі завдання 24 лютого 2025 р.

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів роботи	Термін виконання етапів роботи	Примітка
1	<i>Аналіз аналогічних рішень та загальна характеристика додатку</i>	<i>Лютий, 2025</i>	
2	<i>Розробка розробка ios-додатку моніторингу фінансових активів</i>	<i>Квітень, 2025</i>	
3	<i>Перевірка працездатності розробленого ios-додатку моніторингу фінансових активів</i>	<i>Травень, 2025</i>	
4	<i>Оформлення пояснювальної записки</i>	<i>Червень, 2025</i>	

Студент _____ Бояновський О.В.
(підпис) (прізвище та ініціали)

Керівник роботи _____ Лазорів А.М.
(підпис) (прізвище та ініціали)

АНОТАЦІЇ

Метою даної бакалаврської роботи є розробка iOS-додатку для моніторингу фінансових активів з динамічною візуалізацією інформації про портфоліо та ринкову ситуацію.

У першому розділі було проведено аналіз популярних мобільних додатків у сфері моніторингу криптовалютних активів, зокрема CoinGecko та CoinMarketCap. Здійснено їхнє порівняння за функціональністю, перевагами та недоліками, що дозволило визначити ключові вимоги до власного рішення.

У другому розділі було реалізовано безпосередню розробку додатку із використанням SwiftUI, Combine та Core Data. Побудовано архітектуру за шаблоном MVVM+C, реалізовано інтеграцію з API CoinGecko, створено механізм збереження та оновлення даних, а також локалізацію інтерфейсу. Також увагу приділено реалізації візуалізації даних у вигляді графіків.

У третьому розділі виконано Перевірка функціональних можливостей додатку, перевірено стабільність оновлення даних, правильність розрахунку прибутковості портфоліо та адаптивність інтерфейсу.

За підсумками виконаної роботи підтверджено працездатність та ефективність розробленого програмного продукту.

Ключові слова: iOS-додаток, криптовалюта, SwiftUI, Core Data, CoinGecko API, MVVM+C, мобільна розробка, локалізація, офлайн-режим, візуалізація даних, портфоліо активів, UI/UX, калькулятор валют.

ANNOTATIONS

The aim of this bachelor's thesis is to develop an iOS application for monitoring financial assets with dynamic visualization of portfolio and market data.

The first chapter provides an analysis of popular mobile applications in the field of cryptocurrency asset tracking, specifically CoinGecko and CoinMarketCap. Their functionality, advantages, and disadvantages are compared, which allowed for the identification of key requirements for the custom solution.

The second chapter focuses on the development of the application using SwiftUI, Combine, and Core Data. The architecture is built following the MVVM pattern, integration with the CoinGecko API is implemented, and a mechanism for data storage and updates is developed. The interface is localized, also attention is given to the implementation of data visualization in the form of charts and diagrams.

The third chapter includes functional testing of the application, verification of data update stability, accuracy of portfolio profitability calculations, and adaptability of the interface.

The results confirm the functionality and efficiency of the developed software product.

Keywords: iOS application, cryptocurrency, SwiftUI, Core Data, CoinGecko API, MVVM+C, mobile development, localization, offline mode, data visualization, asset portfolio, currency calculator, and UI/UX

ЗМІСТ

ПЕРЕЛІК ОСНОВНИХ ПОЗНАЧЕНЬ І СКОРОЧЕНЬ.....	5
ВСТУП.....	6
1 АНАЛІЗ АНАЛОГІЧНИХ РІШЕНЬ ТА ЗАГАЛЬНА ХАРАКТЕРИСТИКА ДОДАТКУ.....	8
1.1 Визначення інформаційної системи та інформаційного процесу.....	8
1.2 Аналіз науково-технічної та нормативної документації.....	9
1.3 Аналіз аналогічних рішень	..Помилка! Закладку не визначено.
1.4 Загальна характеристика розробленого додатку13
1.5 Постановка задачі.....	14
2 РОЗРОБКА iOS-ДОДАТКУ МОНІТОРИНГУ ФІНАНСОВИХ АКТИВІВ16
2.1 Розробка структури додатку	16
2.2 Розробка UML діаграми для додатку	19
2.4 Розробка компонентів додатку.....	21
2.4.1 Розробка Market екрану.....	21
2.4.2 Розробка Portfolio екрану.....	24
2.4.3 Розробка Search екрану.....	27
2.4.4 Розробка Calculator екрану.....	29
2.4.5 Розробка Settings екрану.....	33
2.4.6 Розробка Details екрану.....	36
2.5 Налаштування бази даних та офлайн режиму.....	40
3 ПЕРЕВІРКА ПРАЦЕЗДАТНОСТІ РОЗРОБЛЕНОГО iOS-ДОДАТКУ МОНІТОРИНГУ ФІНАНСОВИХ АКТИВІВ.....	43
3.1 Перевірка роботи Market сторінки та переходу на Details екран.....	43

3.2 Здійснення додавання валюти в портфель.....	51
3.3 Перевірка пошуку та калькуляції криптовалют.....	57
3.3 Перевірка налаштувань додатку.....	57
ВИСНОВКИ	62
ПЕРЕЛІК ПОСИЛАНЬ НА ДЖЕРЕЛА	64
ДОДАТКИ	
БІБЛІОГРАФІЧНА ДОВІДКА.....	

ПЕРЕЛІК ОСНОВНИХ ПОЗНАЧЕНЬ І СКОРОЧЕНЬ

API (*Application Programming Interface*) – інтерфейс прикладного програмування;

CoinGecko API – публічний інтерфейс доступу до ринкових даних про криптовалюту;

Combine – фреймворк Apple для реактивного програмування;

Core Data – фреймворк Apple для збереження локальних даних у додатках;

HTTP (*Hypertext Transfer Protocol*) – протокол передачі гіпертексту;

HTTPS (*Hypertext Transfer Protocol Secure*) – захищений протокол передачі гіпертексту;

iOS – мобільна операційна система компанії Apple;

JSON (*JavaScript Object Notation*) – формат обміну даними;

MVVM (*Model-View-ViewModel*) – архітектурний шаблон проектування програмного забезпечення;

SDK (*Software Development Kit*) – набір засобів розробки програмного забезпечення;

SwiftUI – декларативна мова інтерфейсу користувача для платформ Apple;

UI (*User Interface*) – інтерфейс користувача;

UML (*Unified Modeling Language*) – уніфікована мова моделювання систем;

UX (*User Experience*) – досвід користувача;

ВСТУП

В сучасному світі спостерігається значне зростання зацікавленості у використанні мобільних додатків для аналізу фінансових ринків, особливо у сфері криптовалют. Ринок цифрових валют активно розвивається та залучає дедалі більше користувачів, що потребують зручних та ефективних інструментів для моніторингу фінансових активів, аналізу їх цінних змін, а також управління власними портфелями.

Особливою актуальністю відзначаються мобільні застосунки, які дозволяють користувачам оперативно отримувати інформацію, проводити аналітичні розрахунки та стежити за динамікою ринку. З огляду на це, розробка сучасних та зручних мобільних додатків стає пріоритетним напрямом як для індивідуальних користувачів, так і для професійних інвесторів.

Метою даної бакалаврської роботи є розробка мобільного додатку для відстеження та аналізу криптовалютного ринку, що забезпечує користувачам простий та ефективний доступ до необхідної фінансової інформації, аналітичних інструментів та можливість зручно управляти власним портфелем криптовалютних активів.

Об'єктом дослідження виступає мобільний застосунок для аналізу криптовалютного ринку.

Предметом дослідження є процес проектування та реалізації мобільного додатку, що включає взаємодію з API, обробку даних, розробку інтерфейсу користувача та інтеграцію сучасних технологій для ефективного вирішення задач управління криптовалютами активами.

Практичне значення одержаних результатів полягає в створенні зручного та функціонального мобільного додатку, що дозволяє користувачам оперативно відслідковувати зміни на криптовалютному ринку, здійснювати аналітичні розрахунки та приймати обґрунтовані фінансові рішення.

Структура дипломної роботи складається зі вступу, двох основних

розділів, висновків, списку використаних джерел із 45 найменувань. Робота містить 39 рисунків та 2 таблиць. Загальний обсяг роботи складає 67 сторінок машинописного тексту.

1 АНАЛІЗ АНАЛОГІЧНИХ РІШЕНЬ ТА ЗАГАЛЬНА ХАРАКТЕРИСТИКА ДОДАТКУ

1.1 Визначення інформаційної системи та інформаційного процесу

У межах даної дипломної роботи досліджується та розробляється мобільна інформаційна система для моніторингу фінансових активів. Ця система реалізована у вигляді iOS-додатку, який забезпечує збір, зберігання, обробку, фільтрацію, візуалізацію та надання фінансової інформації користувачу в динамічній формі.

Інформаційним процесом, що реалізується системою, є:

- отримання ринкових фінансових даних через відкритий API (CoinGecko);
- обробка, кешування та відображення статистичних показників;
- управління портфелем користувача (введення кількості активів, обчислення прибутковості тощо);
- динамічна візуалізація змін (графіки, порівняння, сортування).

Система також реалізує збереження локальної копії інформації через Core Data, що дозволяє працювати з додатком в офлайн-режимі.

Інформативні ознаки (параметри):

- Назва криптовалюти;
- Поточна ціна;
- Зміна за останні 24 години;
- Ринкова капіталізація;
- Об'єм торгів;
- Кількість активів у портфелі;
- Загальна вартість портфелю;
- Динаміка ціни (у вигляді графіка);
- Мова, тема, стартовий екран.

1.2 Аналіз науково-технічної та нормативної документації

Розробка інформаційної системи для фінансового моніторингу вимагає дотримання ряду технічних та нормативних стандартів.

Технічна документація:

- Apple Human Interface Guidelines — визначає вимоги до UX/UI для мобільних застосунків [44];
- CoinGecko API Documentation — опис REST API, структура JSON-відповідей, правила використання [25].

Нормативна база:

- Регламент роботи з відкритими API без збереження персональних даних.
- Стандарти збереження локальних даних (Core Data) відповідно до вимог Apple Security & Privacy [20-24].

Оскільки додаток не здійснює платіжні операції та не оперує персональними даними, обмеження, пов'язані з фінансовим регулюванням, на нього не поширюються.

1.3 Аналіз аналогічних рішень

CoinGecko

CoinGecko — одна з провідних платформ у сфері криптовалютної аналітики, яка пропонує широкі можливості для моніторингу активів (рис.1.1). Користувачі можуть відстежувати ціни, графіки, об'єми торгів, ринкову капіталізацію та змінювати налаштування портфеля. Доступна підтримка сповіщень та багатомовність. Результати проведеного аналізу відображено в таблиці 1.1.

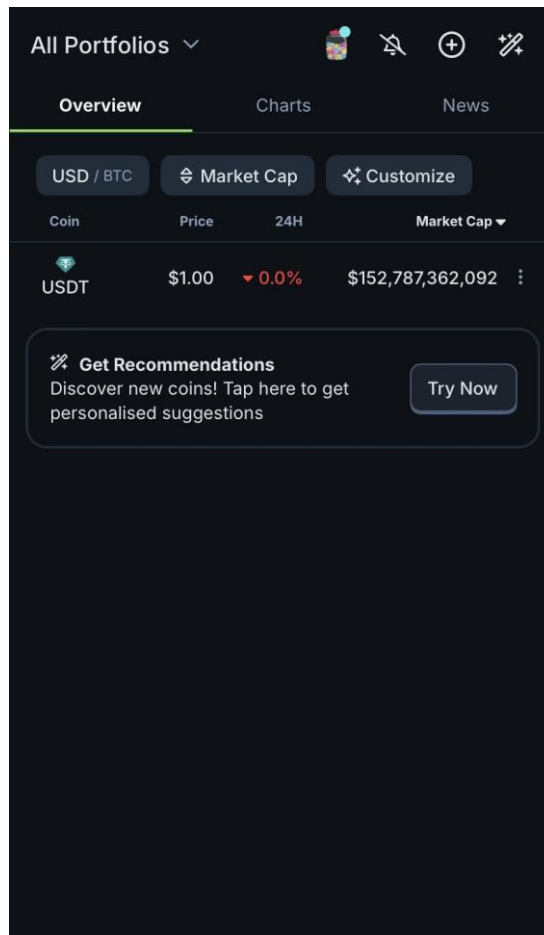


Рисунок 1.1 – Інтерфейс сторінки Портфоліо в додатку CoinGecko

Основний функціонал:

- Відстеження понад 10 000 криптовалют з деталізованою інформацією;
- Дані щодо ринкової капіталізації, обсягів торгівлі, ціни, історичних змін;
- Візуалізація змін курсу у вигляді графіків за годину, день, тиждень, місяць тощо;
- Можливість створення персоналізованого списку монет і власного портфеля, з налаштуванням сповіщень.

Таблиця 1.1 – Переваги та недоліки додатку CoinGecko

Переваги	Недоліки
Широке охоплення активів і джерел даних	Перевантажений інтерфейс, який може бути складним для новачків
Багатомовна підтримка та активна спільнота	Обмежені можливості кастомізації окремих елементів інтерфейсу та портфеля
Часте оновлення інформації у реальному часі	

CoinMarketCap

CoinMarketCap — ще одна популярна платформа, що спеціалізується на крипторинковій статистиці та аналітиці (рис.1.2). Особливістю є інтеграція з популярними біржами та деталізована рейтингова система [26]. Результати проведеного аналізу відображено в таблиці 1.2.

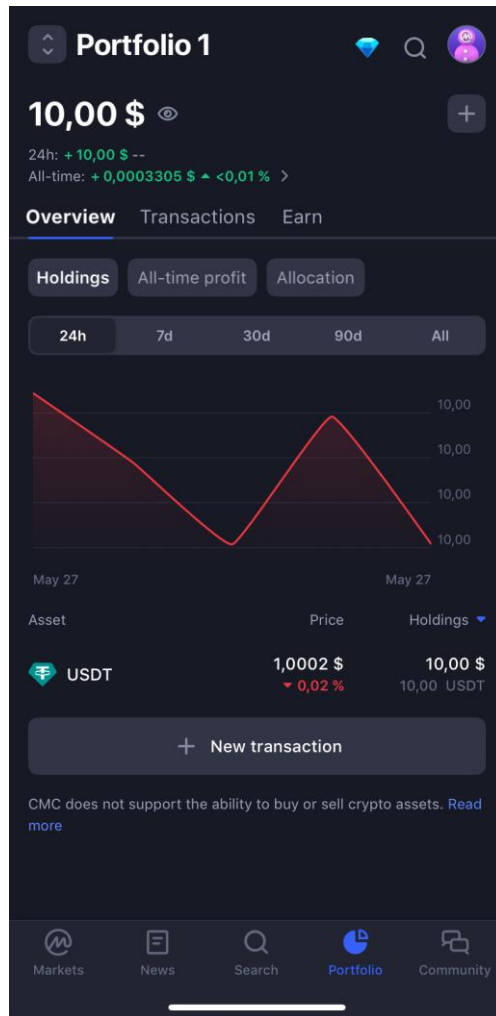


Рисунок 1.2 – Інтерфейс сторінки Порт CoinMarketCap

Основний функціонал:

- Моніторинг цін, ринкової капіталізації, рейтингу криптовалют;
- Створення власного портфеля, відстеження змін, встановлення сповіщень про ціни;
- Інтеграція новин і аналітичних матеріалів прямо в інтерфейсі додатку. Синхронізація з популярними криптобіржами.

Таблиця 1.2 – Переваги та недоліки додатку CoinMarketCap

Переваги	Недоліки
Висока точність рейтингових показників	Затримки з оновленням інформації в окремих випадках

Зручна навігація та підключення до біржових акаунтів	Інтерфейс не має достатньої гнучкості для досвідчених користувачів
--	--

Результати аналізу свідчать, що попри значні функціональні можливості та широку аудиторію, існуючі рішення мають ряд недоліків, які знижують їх зручність та універсальність у використанні.

1.4 Загальна характеристика розробленого додатку

На основі проведеного аналізу аналогічних рішень було сформульовано ключові вимоги до нового продукту, метою якого є поєднання сильних сторін провідних мобільних додатків у сфері криптовалютного моніторингу при усуненні їхніх основних недоліків.

Розроблений додаток є сучасним iOS-застосунком, орієнтованим на моніторинг фінансових активів з акцентом на швидкодію, візуальну аналітику та персоналізацію.

Основні характеристики додатку: зручний та інтуїтивно зрозумілий інтерфейс, побудований відповідно до рекомендацій Apple Human Interface Guidelines [45] із використанням SwiftUI [1-5], що забезпечує адаптивність та простоту у взаємодії. Оновлення даних у режимі реального часу, система взаємодії з публічним API CoinGecko, оновлюючи курси криптовалют, ринкові дані та аналітику з мінімальною затримкою. Візуалізація даних, для кожного активу реалізовано графіки зміни вартості за останні 7 днів, динамічну статистику та ключові фінансові показники. Персоналізоване портфоліо, де користувач має змогу додавати у власне портфоліо будь-які монети, вказувати їх кількість, а система автоматично обчислює загальну вартість і зміну прибутковості. Оптимізованість для новачків і досвідчених користувачів [5-11]. Завдяки простому UI та можливості налаштувати стартовий екран, тему, мову інтерфейсу — додаток є однаково комфортним

для користувачів різного рівня підготовки.

1.5 Постановка задачі

Проаналізувавши функціональні можливості мобільних додатків-конкурентів, зокрема CoinGecko та CoinMarketCap, було прийнято рішення про доцільність створення власного iOS-додатку для моніторингу фінансових активів з простим, зрозумілим інтерфейсом і розширеними можливостями роботи з портфоліо.

Сучасний користувач подібних сервісів потребує швидкого доступу до інформації про вартість активів, зміни на ринку та власного портфеля. Більшість існуючих рішень або перевантажені даними, або мають складну навігацію, що ускладнює роботу з додатком, особливо для початківців.

Рішенням цих проблем є розробка мобільного додатку під операційну систему iOS, який дозволяє миттєво отримувати актуальні дані з фінансових API, зберігати інформацію локально для офлайн-доступу, візуалізувати структуру портфоліо та динаміку змін цін у вигляді графіків, та підтримувати багатомовність.

Для реалізації додатку необхідно спроектувати структуру архітектури додатку (MVVM+C) [41], налагодити взаємодію з CoinGecko API для отримання даних, забезпечити збереження даних у Core Data для офлайн-доступу [13-16], а також створити зручний інтерфейс. Впровадити візуалізацію даних, додати підтримку багатомовності (локалізація на кілька мов), налаштувати механізм оновлення даних у реальному часі, забезпечити простоту у використанні для користувача будь-якого рівня досвіду.

Висновки до розділу

У межах першого розділу було здійснено детальний аналіз популярних мобільних додатків для моніторингу фінансових активів — CoinGecko та

CoinMarketCap. У результаті аналізу визначено їхні переваги, недоліки та функціональні можливості, що дозволило сформулювати вимоги до власного рішення.

Також було обґрунтовано доцільність створення iOS-додатку з фокусом на простоту інтерфейсу, швидкість завантаження даних, можливість створення портфоліо та підтримку багатомовності.

На основі проведеного дослідження сформовано постановку задачі для реалізації iOS-додатку моніторингу фінансових активів із динамічною візуалізацією даних, що забезпечить користувачам зручний доступ до актуальної ринкової інформації.

2 РОЗРОБКА iOS-ДОДАТКУ МОНІТОРИНГУ ФІНАНСОВИХ АКТИВІВ

2.1 Розробка структури додатку

Архітектура додатку побудована за принципом MVVM+C (Model – View – ViewModel + Coordinator) [40-23], що дозволяє чітко розділити відповідальність між окремими компонентами, забезпечити масштабованість та зручну підтримку коду.

Model відповідає за структури даних, отриманих з API, збереження у базі даних (Core Data) та взаємодію з іншими джерелами інформації. View реалізовано з використанням SwiftUI. Вона відповідає за відображення списків активів, динамічних графіків, статистики та деталізованої інформації про кожен актив. ViewModel виконує роль посередника між Model і View. Вона містить бізнес-логіку, обробляє дані та передає їх у View. Реактивність забезпечується за допомогою фреймворку Combine [12, 40]. Coordinator виступає в ролі стартового модуля, що ініціалізує усі процеси в додатку.

Після запуску додатку система звертається до файлу з атрибутом @main, що визначає початкову точку входу. Цей файл ініціалізує головний Coordinator, який, у свою чергу, створює та відображає rootView — головне кореневе представлення інтерфейсу користувача.

```
@main
struct CoinListApp: App {

    @StateObject private var coordinator = AppCoordinator()
    @StateObject private var settings = SettingsManager.shared

    var body: some Scene {
        WindowGroup {
            rootView()
                .environmentObject(coordinator)
        }
    }
}
```

```

        .environmentObject(settings)
        .preferredColorScheme(settings.isDarkMode ? .dark : .light)
        .environment(\.locale, Locale(identifier: settings.languageCode))
    }
}
}

```

RootView умовно поділяється на дві початкові фази: екран завантаження (LaunchView) та основний екран додатку (HomeView).

На основі булевої змінної showLaunchView, Coordinator керує тим, який екран має бути відображений. Якщо showLaunchView == true, користувач бачить екран завантаження. У цей час додаток надсилає HTTPS-запити до API CoinGecko, отримує JSON-відповіді та перетворює їх у модельовані дані, які зберігаються локально за допомогою Core Data [17-20]. Після завершення синхронізації showLaunchView змінює значення на false, і система автоматично переходить до відображення HomeView.

Навігація

Основна навігація реалізована через компонент NavigationView, що є частиною iOS SDK [45]. Внизу екрана розташована навігаційна панель, яка містить п'ять вкладок: Market, Portfolio, Search, Calculator, Settings (рис.2.1). Кожна з них веде на окремий екран. При переході до певного екрану ініціалізується відповідна ViewModel.

З цього моменту починає працювати архітектурний підхід MVVM:

- ViewModel отримує дані з моделі (з бази або API) і обробляє їх.
- ViewModel зберігає дані в оперативній пам'яті та виконує бізнес-логіку (обчислення, сортування, фільтрацію тощо).
- View лише відображає дані, не знаючи, як вони були отримані або змінені.

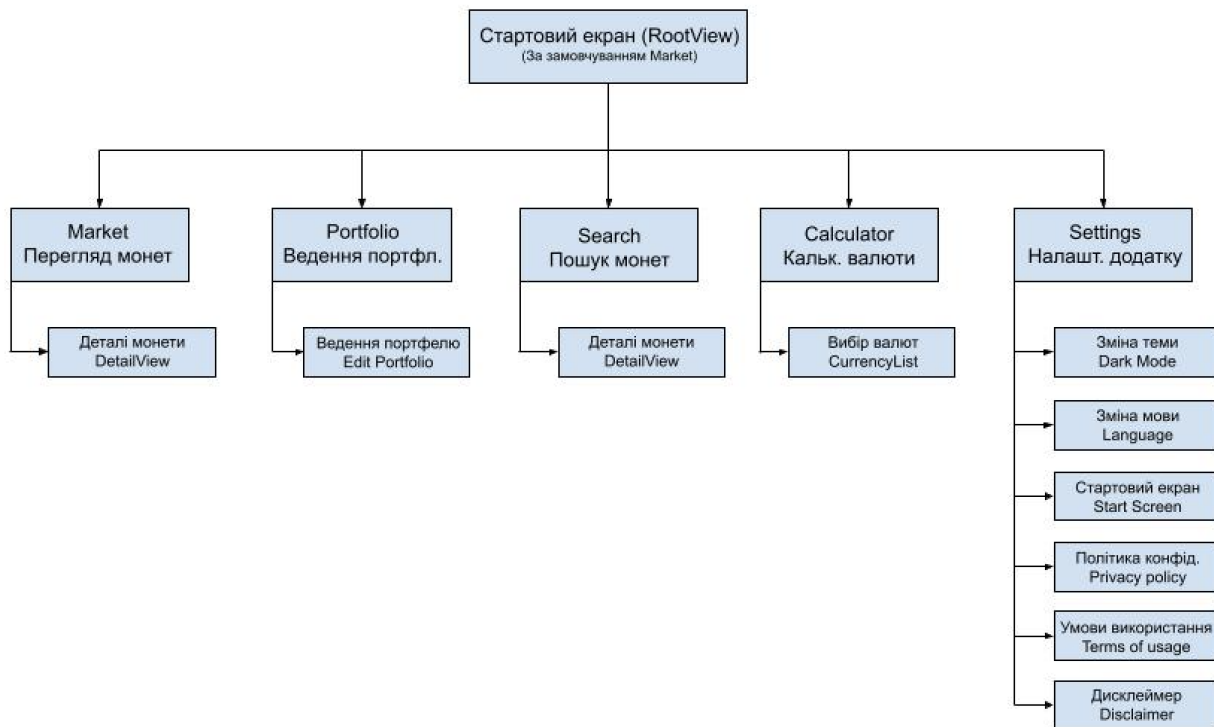


Рисунок 2.1 – Структурна схема навігації додатку

Такий підхід забезпечує розділення обов'язків. View — відповідає за інтерфейс та візуальну частину. ViewModel — за логіку роботи з даними. Model — за збереження та структуру інформації (рис.2.2).

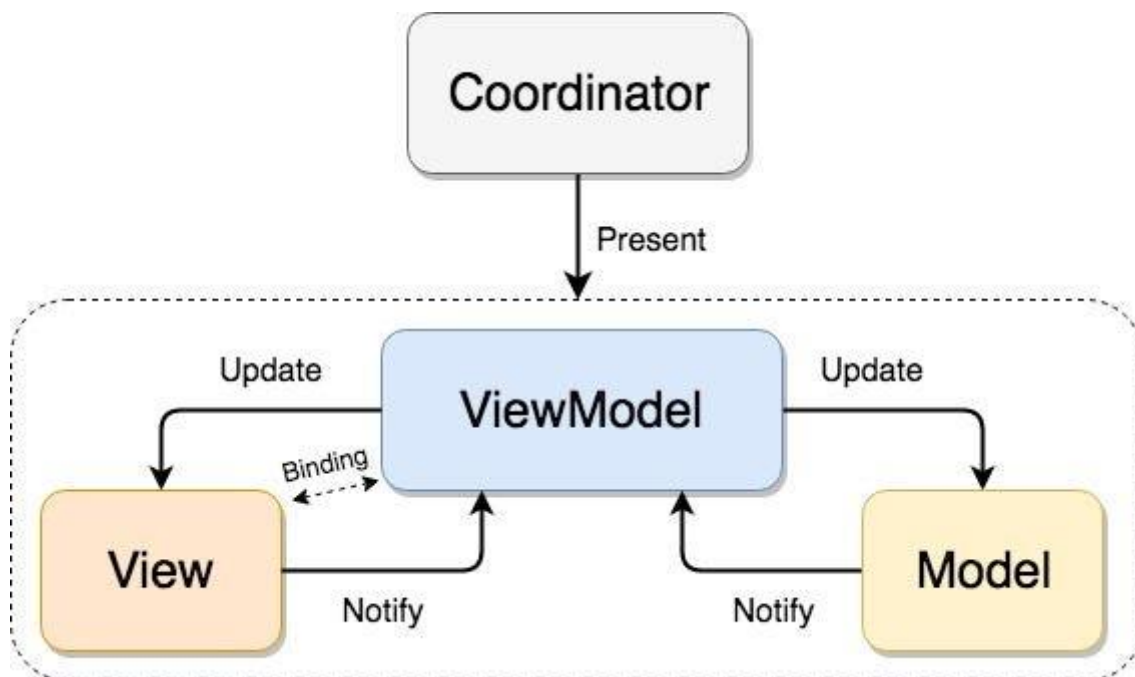


Рисунок 2.2 – Схема архітектури MVVM+C

Підсумок

Coordinator відповідає за ініціалізацію, запуск додатку та завантаження даних. RootView — головне представлення, що визначає логіку відображення екрану завантаження або основного інтерфейсу, NavigationView — забезпечує зручну навігацію між розділами додатку, ViewModel — управляє станом і логікою, пов'язаною з даними кожного окремого екрану, а Model — зберігає дані та виконує взаємодію з API та Core Data.

Реалізація архітектури MVVM+C у дозволила досягти чіткої структури, високої продуктивності та зручності у підтримці та масштабуванні проєкту.

2.2 Розробка UML діаграми для додатку

У процесі проєктування мобільного додатку важливим етапом є побудова UML-діаграм, які відображають взаємозв'язки між елементами системи, основні компоненти архітектури та сценарії взаємодії користувача з додатком (рис.2.3).

Для додатку було розроблено діаграму, яка демонструє основні екрани, переходи між ними та структуру навігації.

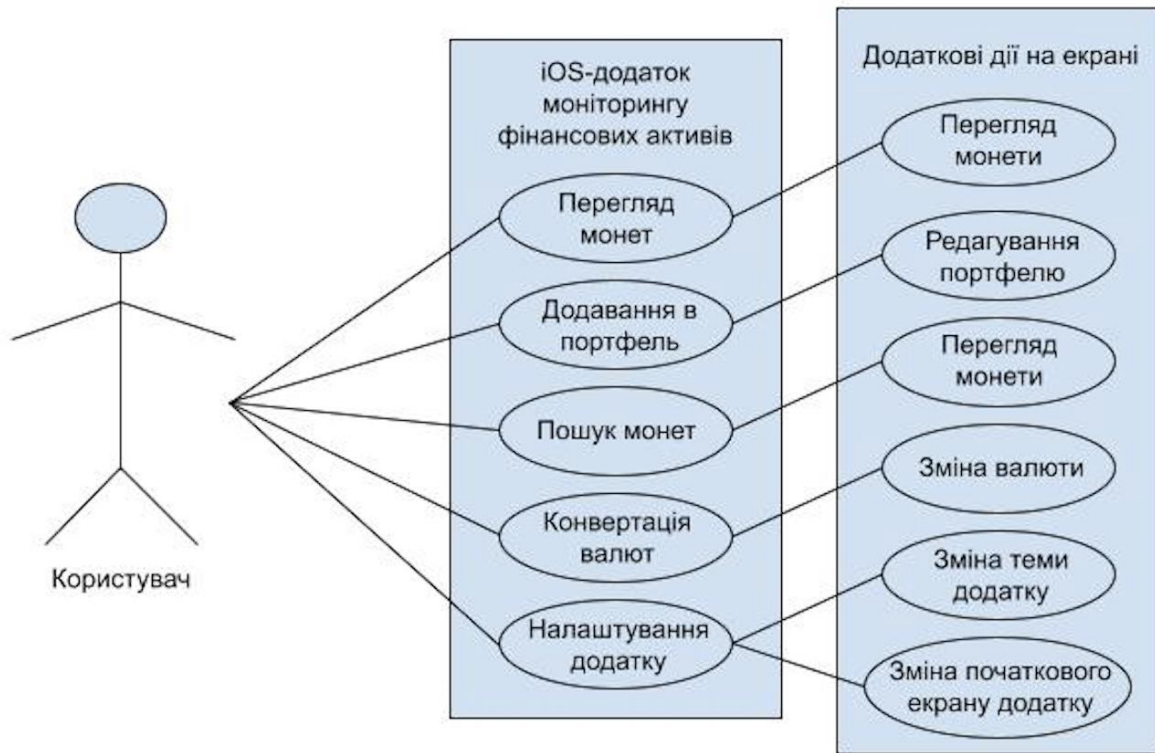


Рисунок 2.3 – UML-діаграма додатку

У додатку реалізовано просту та інтуїтивну модель взаємодії користувача. При першому запуску користувач автоматично потрапляє на головний екран (рис.2.1). З технічної точки зору це дозволяє зосередитися на основному функціоналі без затримок на початковому етапі.

Основні компоненти інтерфейсу:

- Головний екран (HomeView): відображає загальний огляд ринку криптовалют, динаміку змін та основну статистику;
- Панель навігації внизу екрана забезпечує швидкий перехід між п'ятьма основними розділами: Market, Portfolio, Search, Calculator, Settings;
- Додаткові переходи реалізовані через кнопки, що ведуть до детальних переглядів монет або додавання нових активів у портфолію;
- Взаємодія між екранами реалізована через механізм NavigationLink у SwiftUI, а логіка переходів координується за допомогою координатора, який відслідковує стан додатку та відповідає за запуск відповідного сценарію.

Таким чином, побудована UML-схема демонструє чітку архітектуру навігації в додатку, що відповідає принципам зручності та логічної структурованості сучасних iOS-застосунків.

2.4 Розробка компонентів додатку

2.4.1 Розробка Market екрану

Головний екран додатку — Market Screen — є ключовим інтерфейсом для ознайомлення користувача з актуальним станом криптовалютного ринку. Його інтерфейс умовно поділено на три частини:

- Верхній навігаційний бар (Navigation Bar) [7];
- Блок загальної ринкової статистики;
- Список монет з таблицею даних.

Навігаційна панель

У верхній частині екрану розміщено навігаційний бар, в якому є логотипу, назви додатку та дві кнопок навігації: Portfolio і Search (рис.2.4).

Кожна з кнопок дозволяє перейти до однойменного екрану додатку. При натисканні виконується оновлення значення активного екрану, що передає сигнал для Coordinator, який ініціює відображення нової в'ю. Механізм реалізовано за допомогою спостереження за змінною selectedScreen, що пов'язана з NavigationView.



Рисунок 2.4 – Навігаційна панель на екрані Market

Ринкова статистика

Другий компонент — інформаційний блок зі статистикою ринку — відображає агреговані дані про загальна ринкова капіталізація, добовий

торговий обіг і домінування основних криптовалют.

ViewModel отримує цю інформацію з локальної бази даних, яка була попередньо оновлена під час синхронізації з сервером. На основі отриманих даних формується інформативний блок, який дозволяє користувачу побачити загальну ситуацію на ринку (рис.2.5).

```
private func mapMarketData(_ data: MarketDataModel?) -> [StatisticModel] {
    guard let data else { return [] }
    let marketCap = StatisticModel(title: "Global Market Cap", value: data.marketCap,
percentage: data.marketCapChangePercentage24HUsd)
    let volume = StatisticModel(title: "24h Volume", value: data.volume)
    let btcDominance = StatisticModel(title: "BTC Dominance", value:
data.btcDominance)

    return [marketCap, volume, btcDominance]
}
```

Global Market Cap	24h Volume	BTC Dominance
\$3,453.62 Tr	\$135.26 Bn	60.0%
▼ -1.1%		

Рисунок 2.5 – Інформаційний блок зі статистикою ринку

Таблиця монет

Основним функціональним елементом екрану є таблиця з переліком криптовалют (рис.2.6). Він містить такі стовпці:

- Ранг (№);
- Назву монети;
- Поточну ціну;
- Зміну ціни за останні 24 години;
- Ринкову капіталізацію.














#	Coin	Price	24H	Market Cap
1	 BTC	\$108,953	↑0.85%	\$2,164,869,120,000
2	 ETH	\$2,547.43	↑0.80%	\$307,526,673,000
3	 USDT	\$1	↑0.01%	\$152,779,164,000
4	 XRP	\$2.33	↑0.20%	\$137,024,189,000
5	 BNB	\$669.78	↑0.06%	\$97,713,260,300
6	 SOL	\$174.55	↓0.90%	\$90,769,715,400
7	 USDC	\$0.999793	↑0.00%	\$61,701,835,000
8	 DOGE	\$0.223469	↓0.57%	\$33,383,066,100
9	 ADA	\$0.75801	↑1.70%	\$27,342,984,300
10	 TRX	\$0.271349	↑0.33%	\$25,744,900,700
11	 STETH	\$2,549.01	↑0.75%	\$22,941,833,400
12	 WBTC	\$108,862	↑0.82%	\$14,023,018,800
13	 LIVDE	\$39.07	↑10.98%	\$13,045,245,100

Рисунок 2.6 – Інформаційний блок зі статистикою ринку

Для зручності користувача реалізовано механізм сортування за кількома колонками (ціна, зміна за 24 год, капіталізація). Натискання на назву відповідної колонки автоматично оновлює вигляд таблиці відповідно до обраного критерію.

```
private func sortCoins(by option: SortOption, coins: inout [CoinModel]) {
    switch option {
    case .perDay:
        coins.sort { ($0.priceChangePercentage24H ?? 0) < ($1.priceChangePercentage24H ?? 0) }
```

```

        case .perDayReversed:
            coins.sort { ($0.priceChangePercentage24H ?? 0) >
($1.priceChangePercentage24H ?? 0) }
        case .price:
            coins.sort { $0.currentPrice > $1.currentPrice }
        case .priceReversed:
            coins.sort { $0.currentPrice < $1.currentPrice }
        case .marketCap:
            coins.sort { ($0.marketCap ?? 0) > ($1.marketCap ?? 0) }
        case .marketCapReversed:
            coins.sort { ($0.marketCap ?? 0) < ($1.marketCap ?? 0) }
        case .none, .holdings, .holdingsReversed:
            break
    }
}

```

При виборі певної монети, користувача перенаправляє на деталізований екран монети, де відображено розширену інформацію про вибраний актив. Повний вигляд сторінки наведено у додатку А.

2.4.2 Розробка Portfolio екрану

Сторінка Portfolio дозволяє користувачу відстежувати власні активи, додавати нові монети до портфеля, переглядати їхній ріст та загальну капіталізацію.

Інтерфейс екрана складається з п'яти компонентів:

- Верхня панель навігації з кнопкою додаванняж
- Блок зі статистикою портфоліо;
- Список монет, доданих користувачем;
- Інформаційна підказка (якщо портфоліо порожнє);
- Екран додавання монети.

Верхня панель навігації

У навігаційному барі розміщено кнопку з іконкою “+”, яка відкриває інтерфейс для додавання нової монети до портфоліо (рис.2.7). Цей елемент завжди доступний незалежно від стану портфеля, що сприяє швидкому доступу до відповідної функції.



Рисунок 2.7 – Верхня панель навігації з кнопкою додавання на екрані Portfolio

Підказка та порожній стан

У випадку, якщо користувач ще не додав жодного активу, відображається підказка з пропозицією почати наповнення портфелю. У цей момент блок статистики відображає нульові значення, а таблиця монет не показується (рис.2.8). Це реалізовано через перевірку кількості активів у локальній базі даних.

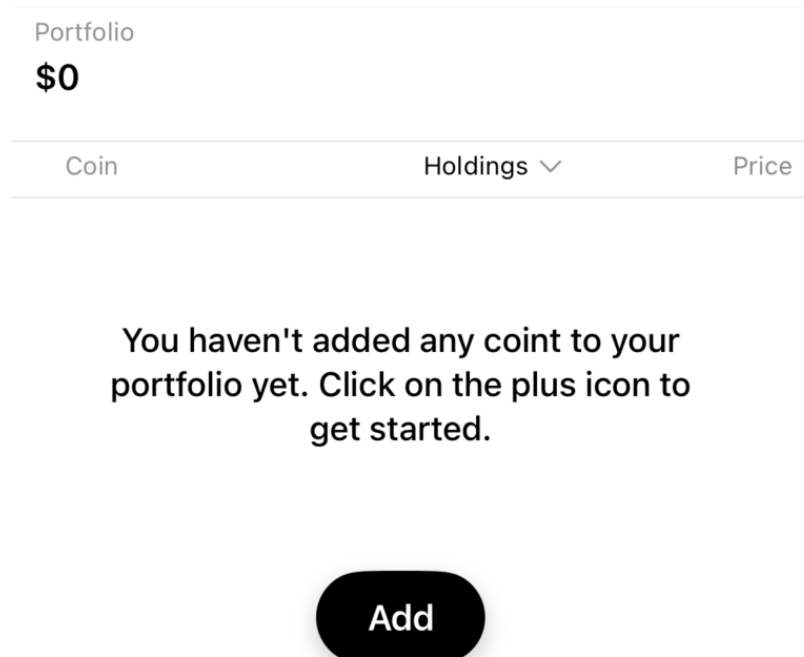


Рисунок 2.8 – Підказка з пропозицією почати наповнення портфелю

Додавання монет

Після натискання на кнопку “+” відкривається екран додавання активу (реалізований як pop-up). Користувач може обрати монету зі списку та вказати її кількість. Для забезпечення зручності UX/UI, екран додавання і екран портфоліо використовують спільну ViewModel, що дає змогу одразу записати обрані монети до бази даних [21-24], автоматично оновити вміст таблиці активів та перерахувати статистику.

```
private func saveButtonTapped() {
    guard let coin = selectedCoin, let amount = Double(quantityText) else { return }

    viewModel.updatePortfolio(coin: coin, amount: amount)
}

func updatePortfolio(coin: CoinModel, amount: Double) {
    if let entity = savedEntities.first(where: { $0.coinID == coin.id }) {
        if amount > 0 {
            update(entity: entity, amount: amount)
        } else {
            delete(entity: entity)
        }
    } else {
        add(coin, amount: amount)
    }
}
```

Список монет та статистика

Після додавання хоча б однієї монети підказка зникає, а на її місці з’являється таблиця з монетами у портфелі (рис.2.9). Таблиця містить коротку інформацію по кожному активу, а у верхній частині екрана відображається загальна статистика портфеля: сукупна вартість в \$ (долар),

ріст або падіння за останні 24 години та кількість активів.

Coin	Holdings	Price
ETH	10 189,72 \$	2 547,43 \$ ↑ 0.80%
DOGE	27,710156 \$	0,223469 \$ ↓ 0.57%
USDC	11,997516 \$	0,999793 \$ ↑ 0.00%

Рисунок 2.9 – Список активів в портфолію користувача з статистикою по ньому

Ці дані оновлюються в реальному часі на основі поточних ринкових цін, що отримуються з серверу та зберігаються у базі. Повний вигляд сторінки наведено у додатку Б.

2.4.3 Розробка Search екрану

Екран Search відповідає за швидкий доступ до інформації про будь-яку монету, доступну в базі додатку. Його реалізація передбачає зручність для користувача без необхідності виконання зайвих дій.

Інтерфейс Search екрану складається з двох частин: поле пошуку та список результатів пошуку.

Поле пошуку

Поле пошуку розташоване у верхній частині екрана та реалізоване з урахуванням стандартів Apple [44-45] щодо пошукових компонентів. Користувач вводить текст, і вже під час введення (у режимі реального часу) список результатів динамічно оновлюється. Це покращує взаємодії з

інтерфейсом та усуває потребу в додаткових діях.

Для обробки запиту реалізовано фільтр, який виконує пошук за повною або частковою назвою монети, ігнорує пробіли, відступи та інші зайві символи, що забезпечує нечутливість до регістру з обробку неповних запитів.

```
private func filterCoins(coins: [CoinModel], by query: String) -> [CoinModel] {
    let trimmed = query.trimmingCharacters(in: .whitespacesAndNewlines)

    guard !trimmed.isEmpty else { return coins }

    let lower = trimmed.lowercased()

    return coins.filter {
        $0.name.lowercased().contains(lower)
        || $0.symbol.lowercased().contains(lower)
        || $0.id.lowercased().contains(lower)
    }
}
```

Список результатів

У відповідь на пошуковий запит користувач бачить оновлений список монет, кожна з яких містить додаткову інформацію:

- назва монети;
- поточна ринкова ціна;
- відсоткова зміна ціни за останні 24 години.

Це дозволяє не лише знайти монету, а й швидко оцінити її ринкову ситуацію без потреби відкривати додатковий екран.

```
if viewModel.filteredCoins.isEmpty && !viewModel.searchText.isEmpty {
    Spacer()
    EmptyStateView(
        imageName: "bookmark.slash",
        title: "No matching results",
```

```
        subtitle: "Try searching for a different term and check your spelling"
    )
    Spacer()
} else {
    SearchCoinListView(coins: viewModel.filteredCoins)
}
```

У випадку, якщо жодна монета не відповідає введеному запиту, на екрані відображається інформаційне повідомлення, яке повідомляє користувача про відсутність результатів. Також може бути запропоновано уточнити запит або перевірити правильність написання. Повний вигляд сторінки наведено у додатку В.

2.4.4 Розробка Calculator екрану

Екран Calculator виконує функцію конвертації введеної користувачем суми у вибраній валюті у відповідну кількість різних криптовалют. Це дає можливість користувачу швидко оцінити купівельну спроможність у контексті ринкових цін.

Інтерфейс складається з трьох елементів: поле введення суми, вибір валюти і список криптовалют з розрахованою кількістю.

Ініціалізація та бізнес логіка

Після запуску екрана відбувається ініціалізація CalculatorViewModel, яка звертається до локальної бази даних по актуальний список монет, отримує поточні курси валют (як криптовалют, так і фіатних) та зберігає ці дані в оперативній пам'яті для подальших обчислень.

Потім ці дані використовуються для оновлення списку на екрані відповідно до введеної суми та обраної валюти.

Поле введення суми

Поле введення підтримує лише цифрові символи, що реалізовано через обмеження на тип введення. Це дозволяє уникнути введення літер або спеціальних символів, які можуть спричинити помилки під час розрахунку (рис.2.10).

```
TextField("0", text: $viewModel.amountText)  
  .keyboardType(.decimalPad)  
  .focused($isFocused)  
  .font(.title2.weight(.semibold))  
  .padding(.vertical, 12)  
  .padding(.leading, 16)
```

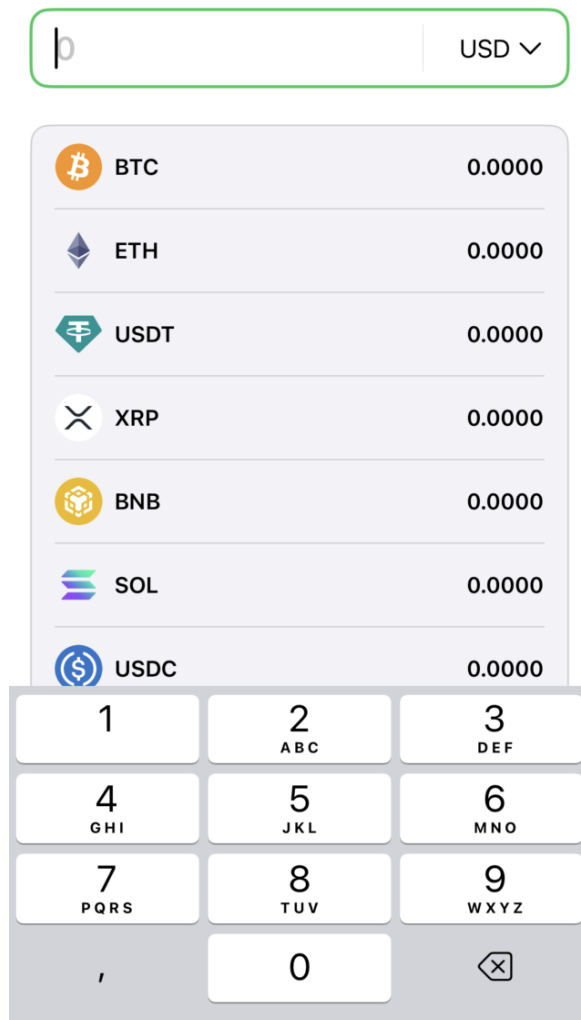


Рисунок 2.10 – Вигляд роботи поля пошуку з цифровою клавіатурою

Поруч із полем розміщена кнопка з поточно обраною валютою.

Натискання на неї відкриває окремий екран, де відображено перелік найпопулярніших фіатних валют і криптовалют, серед яких користувач може обрати бажану (рис.2.11).

Для зручності реалізовано пошук за назвою валюти, що дозволяє швидко знайти, наприклад, “гривню” чи “євро”.

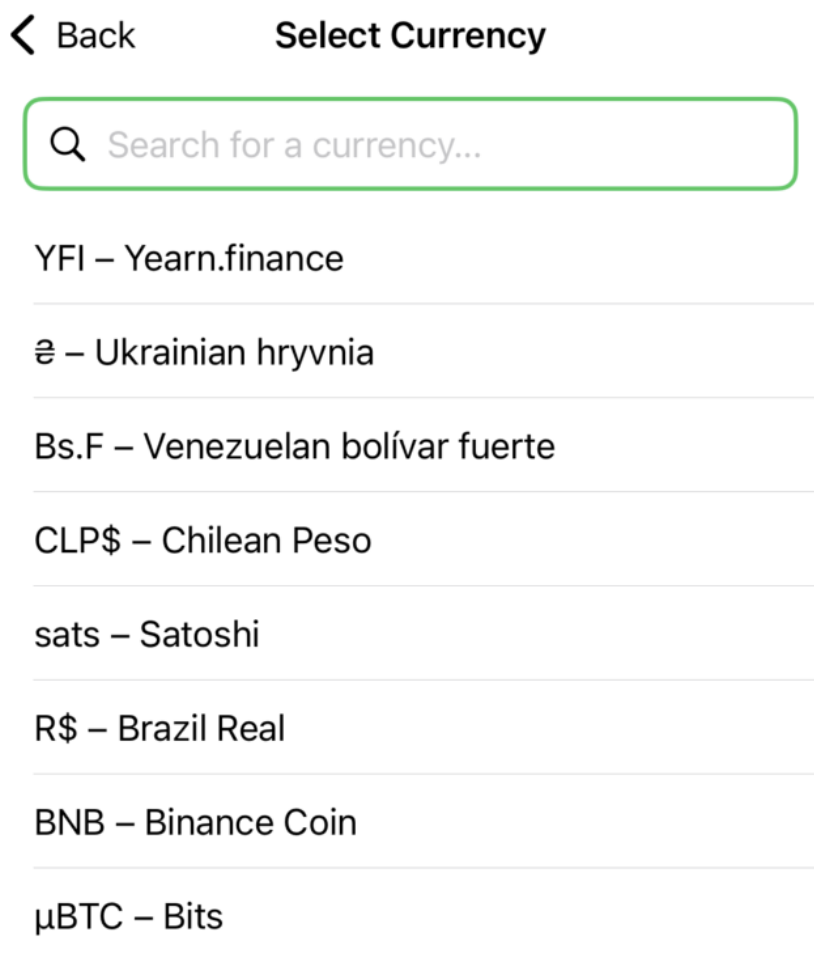


Рисунок 2.11 – Екран вибору валюти з пошуком

Список монет з конвертацією

Після введення суми калькулятор автоматично обчислює та відображає кількість кожної криптовалюти, яку користувач міг би отримати за цю суму (рис.2.12). Дані оновлюються в режимі реального часу, без необхідності додаткових дій.








37500		USD ▾
	BTC	0.3442
	ETH	14.7207
	USDT	37,500.0000
	XRP	16,094.4206
	BNB	55.9885
	SOL	214.8382
	USDC	37,507.7641

Рисунок 2.12 – Список монет з конвертованою кількістю

Для виконання обчислень використовується окремий конвертаційний модуль, який визначає відношення між поточною вартістю криптовалюти та введеною сумою в обраній валюті.

```

func quantity(for coin: CoinModel) -> Double {
    let sanitized = amountText.replacingOccurrences(of: ",", with: ".")
    guard
        let fiatAmount = Double(sanitized),
        let rate = fetchRate(code: selectedFiat),
        let usdRate = fetchRate(code: "usd"),
        coin.currentPrice > 0
    else { return 0 }

    var priceInFiat = Double()

```

```

if selectedFiat == "usd" {
    priceInFiat = coin.currentPrice * rate
} else {
    priceInFiat = coin.currentPrice * (rate / usdRate)
}

return fiatAmount / priceInFiat
}

```

Завдяки даній логіці взаємодія користувача з калькуляцією значно спрощується, оскільки список оновлюється динамічно. Повний вигляд сторінки наведено у додатку Г.

2.4.5 Розробка Settings екрану

Екран Settings забезпечує можливість гнучкого налаштування додатку відповідно до уподобань користувача. Його функціонал реалізовано через взаємодію з менеджером налаштувань (SettingsManager), що дозволяє в лайві оновлювати UI елементи.

Інтерфейс SettingsView має з дві секції: Налаштування (Preferences) і Інше (Other).

Секція “Налаштування”

У цій частині користувач має доступ до трьох параметрів: Темна тема, Мова інтерфейсу та Початковий екран (рис.2.13).

Темна тема (Dark Mode) [35-37]. Дозволяє перемикатися між світлим та темним видами інтерфейсу. Реалізовано з використанням кастомних кольорів, доданих у конфігурацію середовища. Для цього було створено розширення до UIColor з властивість theme, що посилається на доданий набір кольорів. Зміна теми відбувається шляхом оновлення preferredColorScheme на .dark або .light, з оновленням усіх дочірніх View [38-39].

Мова інтерфейсу (Language) [28-31]. Реалізована через механізм локалізації з використанням файлу Localizable.strings. У цьому файлі зберігаються всі текстові значення. Після додавання нової мови всі ключі дублюються, і для них задаються переклади [32-34]. Додаток підтримує англійську, українську, німецьку, французьку та іспанську мови.

Початковий екран (Start Screen) [42]. Користувач може обрати, який із чотирьох основних екранів буде відкриватися при запуску додатку. Обране значення зберігається в UserDefaults за допомогою SettingsManager. Під час ініціалізації AppCoordinator зчитує це значення та передає його у rootView. Це дозволяє відобразити потрібний стартовий екран.

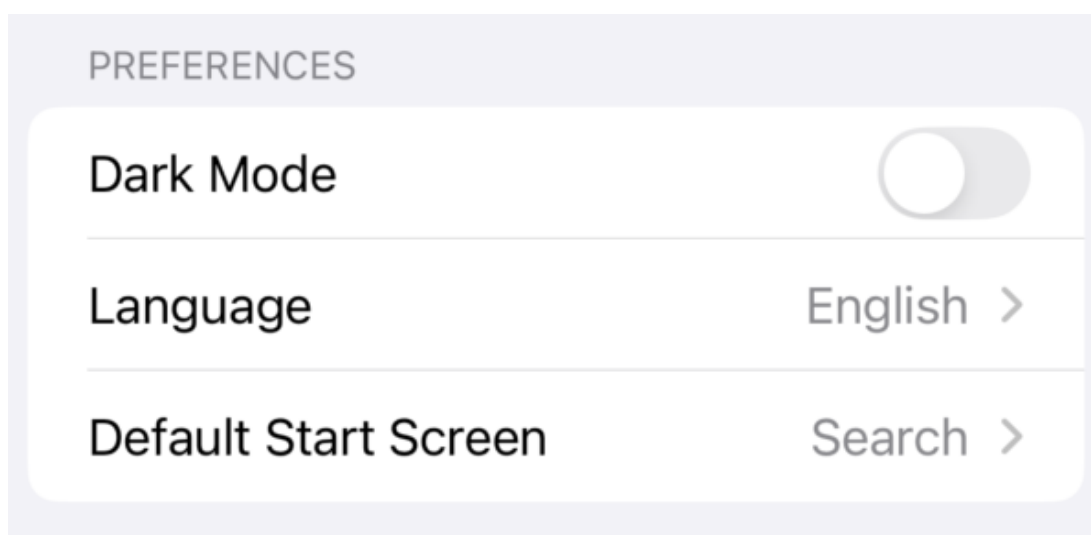


Рисунок 2.13 – Секція “Налаштування” з параметрами: Dark Mode, Language та Default Start Screen

Секція “Інше”

У секції “Інше” розміщені інформаційні блоки Політика конфіденційності, Умови надання послуг та Дисклеймер (рис.2.14).

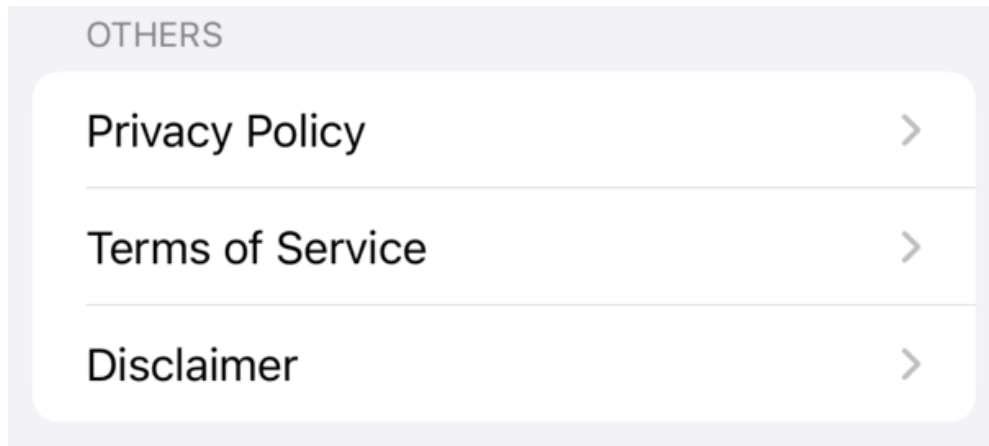


Рисунок 2.14 – Секція “Інше” з параметрами: Privacy Policy, Terms of Service та Disclaimer

Кожен пункт веде на окрему сторінку, яка завантажується з HTML-файлу. Для кожного з документів існує локалізована версія. Відображення зроблено за допомогою створеного `SettingInfoWebView`, яке працює на базі `WKWebView`. Це в'ю приймає `enum SettingInfoType`, який визначає: `localizedTitle` — заголовок, перекладений відповідно до вибраної мови і `fileName` — назву відповідного HTML-файлу..

```
struct SettingInfoWebView: View {
    let infoType: SettingInfoType
    var body: some View {
        if let url = localizedHTMLURL() {
            WebView(url: url)
                .navigationTitle(infoType.localizedTitle)
                .navigationBarTitleDisplayMode(.inline)
        } else {
            Text("Unable to load \((infoType.localizedTitle).")
                .foregroundColor(.red)
        }
    }
}

private func localizedHTMLURL() -> URL? {
    let languageCode = SettingsManager.shared.languageCode
```

```

        if let path = Bundle.main.path(forResource: infoType.fileName, ofType: "html",
inDirectory: nil, forLocalization: languageCode) {
            return URL(fileURLWithPath: path)
        }

        return Bundle.main.url(forResource: infoType.fileName, withExtension: "html")
    }
}

```

Завдяки цьому `SettingInfoWebView` є універсальним компонентом, що може використовуватися для усіх трьох інформаційних документах. Якщо користувач змінює мову в налаштуваннях, ця зміна працює також для відкритого документу. Повний вигляд сторінки наведено у додатку Г.

2.4.6 Розробка Details екрану

Екран `Details` надає користувачу розширену інформацію про обрану валюту. Цей екран відкривається при натисканні на один з елементів у таблицях на сторінках `Market` або `Search`. Він створений як окрема в'ю, що з'являється в режимі `push`-навігації з правого боку екрану.

Структура екрана

`Details Screen` складається з декількох компонентів, які надають юзеру всю інформацію про дану валюту. Заголовок (`Title Section`) У верхній частині розміщено: кнопку “Назад” для повернення на попередній екран, логотип криптовалюти і повну назву та аббревіатуру (рис.2.15).

Рисунок 2.15 – Верхня панель з кнопкою повернення та інформацією про монету на екрані Details

Графік зміни ціни (Chart View). Один з основних елементів сторінки — графік зміни ціни криптовалюти за останні 7 днів (рис.2.16). Він зроблений з використанням лінійної діаграми, яка показує динаміку росту та спадання криптовалюти. Крім самої лінії, на графіку відображено:

- максимальну ціну;
- мінімальну ціну;
- середнє значення за період.

Графік оновлюється в реальному часі на основі API-даних, що робить візуалізацію актуальною.

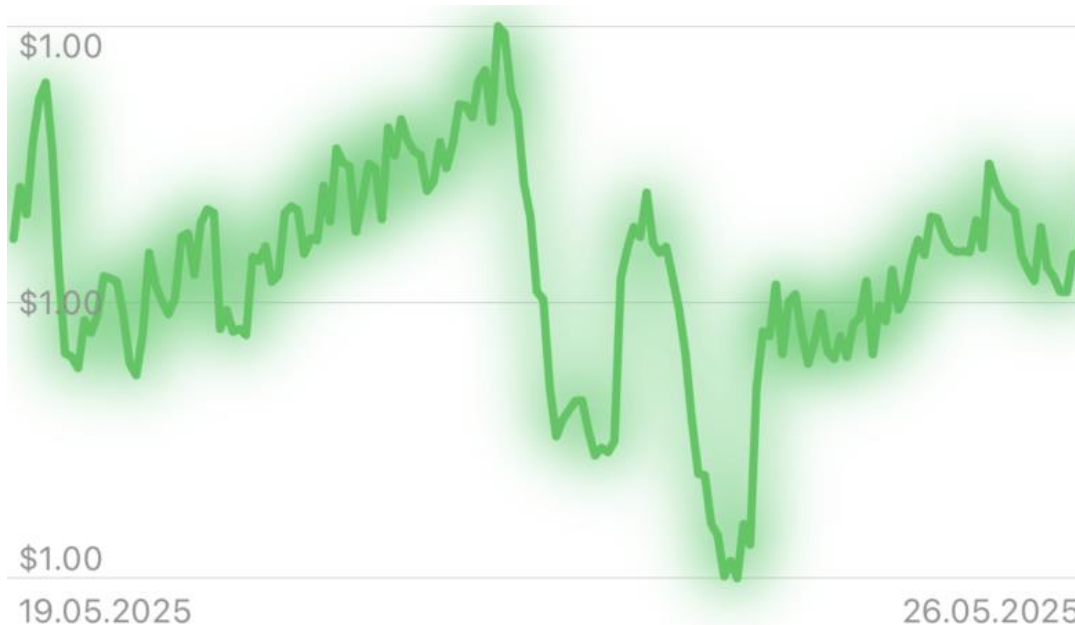


Рисунок 2.16 – Графік ціни на ринку з ціновими межами на екрані Details

Основна інформація (Overview Section). Ця секція містить короткий опис криптовалюти, її мету, походження та місце на ринку (рис.2.17). Також

включено базову статистику: ринкова капіталізація, добовий обсяг торгів і загальна пропозиція монет.

Overview

Tether (USDT) is a cryptocurrency with a value meant to mirror the value of the U.S. dollar. The idea was to create a stable crypt...

[Read more...](#)

Current Price

\$1

▲ 0.0%

Market Cap

\$152.78 Bn

▲ 0.0%

Rank

3

Volume

\$48.82 Bn

Рисунок 2.17 – Секція Overview з основна інформація про валюту

Додаткова інформація (Additional Info). У цьому блоці наведено менш поширені, але корисні метрики: дата створення, хеш-функція (для блокчейнів), категорія або сектор використання, чи підтримується монета на основних біржах (рис.2.18).

Additional Details

High	Low
\$1	\$1
24h Price change	Market Cap
\$1	\$152.78 Bn
▲ 0.0%	▲ 0.0%
Block time	Hashing algorithm
n/a	n/a

Рисунок 2.18 – Секція Additional Info з додатковою інформацією про валюту

Корисні посилання (Links Section). Внизу екрану містяться активні посилання на зовнішні ресурси, що стосуються криптовалюти: офіційний вебсайт проєкту, сторінка спільноти [25] або форуму та документація або whitepaper, якщо доступна (рис.2.19).

Links

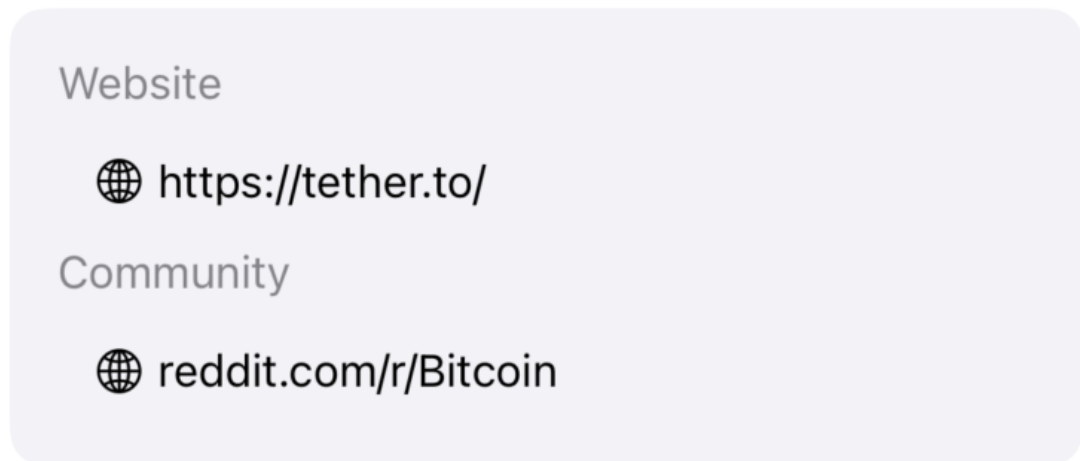


Рисунок 2.19 – Секція Links Info з посилання на зовнішні ресурси

Вся ці дані є новими, бо отримуються при відкритті Details сторінки, а не збереженими в базу. Тому користувач може проводити роботу базуючись на актуальній інформації. Повний вигляд сторінки наведено у додатку Д.

2.5 Налаштування бази даних та офлайн режиму

З огляду на специфіку додатку — а саме необхідність постійного доступу до фінансових даних — було прийнято рішення реалізувати офлайн-режим, який забезпечує доступ до інформації навіть за відсутності інтернет-з'єднання. Для цього у проекті використовується вбудована технологія Apple — Core Data [15-20].

Усі дані в додатку оновлюються при запуску або коли користувач виконує pull-to-refresh.

```
func reloadData() {
    dataSync.syncCoinsData()
        .receive(on: DispatchQueue.main)
        .sink { [weak self] in self?.fetchRawCoins() }
        .store(in: &dataSyncCancellables)
```

```
dataSync.syncMarketData()
    .receive(on: DispatchQueue.main)
    .sink { [weak self] in self?.fetchStatistics() }
    .store(in: &dataSyncCancellables)
}
```

Процес виглядає наступним чином:

- Coordinator ініціалізує виклик API;
- отримані JSON-дані десеріалізуються у відповідні data models;
- дані з моделей трансформуються у відповідні Core Data entities.

Всі зміни записуються в локальну базу, а ViewModel отримує сповіщення про оновлення.

Висновки до розділу

У другому розділі дипломної роботи було детально розглянуто процес розробки iOS-додатку для моніторингу фінансових активів — починаючи з вибору архітектурного підходу, закінчуючи реалізацією ключових екранів та налаштуванням бази даних.

Було обґрунтовано доцільність використання архітектури MVVM+C [40-42], яка спрощує перевірку та масштабування проєкту. Особливу увагу приділено налаштуванню координатора, що відповідає за керування потоком екранів і стартовою логікою додатку.

У межах підрозділів здійснено опис кожного з екранів:

- Market — для перегляду ринку криптовалют з динамічною таблицею та можливістю сортування;
- Portfolio — для управління власними активами, з підтримкою додавання монет і статистикою росту;
- Search — для швидкого пошуку криптовалют у реальному часі;
- Calculator — для конвертації сум у криптовалюту;
- Settings — для персоналізації додатку (тема, мова, стартовий екран,

доступ до політики конфіденційності тощо);

– Details — для детального ознайомлення з інформацією про монету, включно з графіком ціни.

Окремо розглянуто оновлення локальної бази даних через Core Data, що дозволяє реалізувати повноцінну підтримку офлайн-режиму та відображати користувачу найновіші дані.

3 ПЕРЕВІРКА ПРАЦЕЗДАТНОСТІ РОЗРОБЛЕНОГО iOS-ДОДАТКУ МОНІТОРИНГУ ФІНАНСОВИХ АКТИВІВ

3.1 Перевірка роботи Market сторінки та переходу на екран Details

З метою підтвердження правильності екрану Market, було проведено перевірку даної сторінки, а також переходу на Details при натисканні на один з елементів таблиці.

Перевірка поведінки Market екрану

Після запуску додатку користувачу відображається головний екран із поточним списком криптовалют та загальною ринковою статистикою. Було виконано наступні дії для перевірки його працездатності: скролінг списку монет вверх/вниз (рис.3.1).

Очікуваний результат: список коректно прокручується, відображаючи додаткові елементи без збоїв.














#	Coin	Price	24H	Market Cap ▼
25	 LTC	\$95.59	↓ 0.07%	\$7,253,290,190
26	 WETH	\$2,554.39	↑ 1.34%	\$7,219,907,030
27	 DOT	\$4.51	↓ 0.63%	\$6,853,105,980
28	 USDS	\$0.99972	↓ 0.00%	\$6,650,320,050
29	 BGB	\$5.38	↓ 3.42%	\$6,274,930,800
30	 WEETH	\$2,719.44	↑ 0.95%	\$6,182,046,620
31	 BSC-USD	\$1	↓ 0.30%	\$5,985,949,690
32	 PEPE	\$0.00001359	↓ 1.12%	\$5,716,655,080
33	 PI	\$0.782337	↑ 0.25%	\$5,666,682,550
34	 USDE	\$1.002	↑ 0.21%	\$5,219,693,850
35	 WBT	\$32.3	↑ 1.61%	\$4,654,908,110
36	 CBBTC	\$109,108	↑ 1.13%	\$4,625,193,220
37	 AAVE	\$268.16	↑ 2.68%	\$4,050,088,500

Рисунок 3.1 – Перевірка скролінгу списку монет вверх/вниз

Оновлення списку за допомогою Pull-to-Refresh (рис.3.2). Очікуваний результат: оновлення списку монет та статистики, завантаження нових даних з API, відображення останніх цін і змін.

The screenshot shows the Coinlist mobile application interface. At the top, there is a Bitcoin logo and the text 'coinlist'. To the right are icons for a star and a search magnifying glass. Below this, three summary statistics are displayed: 'Global Market Cap' at \$3,453.62 Tr (with a -1.1% change), '24h Volume' at \$135.26 Bn, and 'BTC Dominance' at 60.0%. A table below lists 11 cryptocurrencies with columns for rank, coin name, price, 24-hour change, and market cap. A pull-to-refresh spinner is visible above the table. At the bottom, a navigation bar contains icons for Market, Portfolio, Search, Calculator, and Settings.

#	Coin	Price	24H	Market Cap
1	BTC	\$108,953	↑ 0.85%	\$2,164,869,120,000
2	ETH	\$2,547.43	↑ 0.80%	\$307,526,673,000
3	USDT	\$1	↑ 0.01%	\$152,779,164,000
4	XRP	\$2.33	↑ 0.20%	\$137,024,189,000
5	BNB	\$669.78	↑ 0.06%	\$97,713,260,300
6	SOL	\$174.55	↓ 0.90%	\$90,769,715,400
7	USDC	\$0.999793	↑ 0.00%	\$61,701,835,000
8	DOGE	\$0.223469	↓ 0.57%	\$33,383,066,100
9	ADA	\$0.75801	↑ 1.70%	\$27,342,984,300
10	TRX	\$0.271349	↑ 0.33%	\$25,744,900,700
11	STETH	\$2,549.01	↑ 0.75%	\$22,941,833,400

Рисунок 3.2 – Оновлення списку за допомогою Pull-to-Refresh

Натискання на одну з криптовалют у списку. Очікуваний результат: відкривається екран Details (рис.3.3), який містить розширену інформацію про вибрану монету.

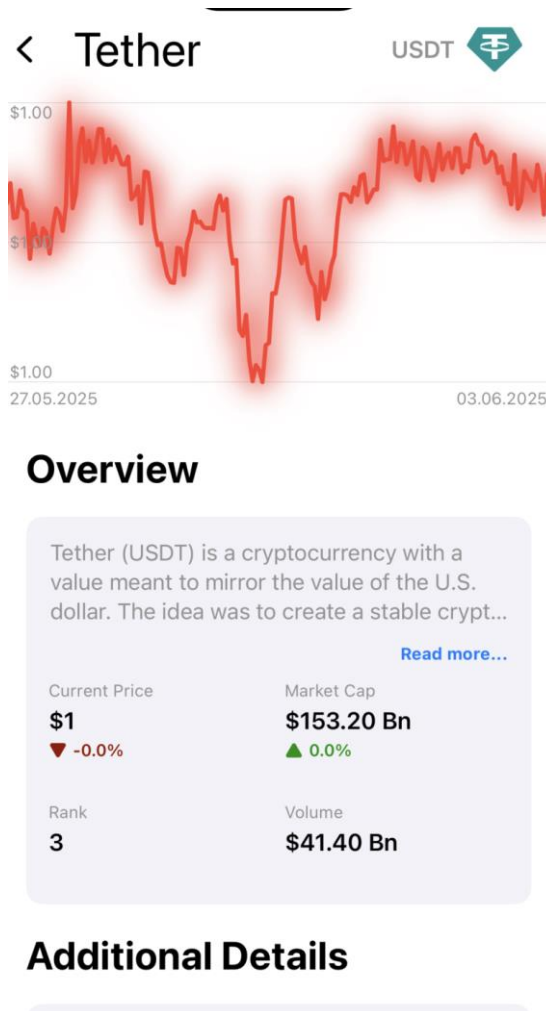


Рисунок 3.3 – Оновлення списку за допомогою Pull-to-Refresh

Сортування таблиці монет за критеріями (ціна, зміна, капіталізація). Очікуваний результат: після натискання на відповідний заголовок колонки (рис.3.4), таблиця автоматично оновлюється згідно обраного сортування (рис.3.5).














#	Coin	Price	24H ∨	Market Cap
125	 CAKE	\$2.35	↓ 0.02%	\$756,965,191
80	 QNT	\$94.92	↓ 0.95%	\$1,380,198,790
250	 FRAX	\$3.09	↓ 4.45%	\$279,446,735
173	 KAITO	\$1.91	↓ 2.35%	\$461,555,990
82	 VIRTUAL	\$2.03	↑ 1.78%	\$1,331,654,520
52	 TKX	\$32.72	↓ 4.55%	\$2,617,820,560
85	 INJ	\$13	↓ 2.38%	\$1,267,964,660
73	 JUP	\$0.555872	↑ 1.38%	\$1,610,572,420
189	 EIGEN	\$1.41	↑ 1.14%	\$427,824,254
232	 SAFE	\$0.546558	↑ 5.60%	\$317,012,782
37	 AAVE	\$268.16	↑ 2.68%	\$4,050,088,500
201	 ETHFI	\$1.23	↑ 0.51%	\$395,560,131
123	 LDO	\$0.873718	↑ 0.85%	\$782,187,438

Рисунок 3.4 – Сортування таблиці за ростом 24г

#	Coin	Price ^	24H	Market Cap
248	BABYDOGE	\$0.000000001681	↑ 0.07%	\$280,440,783
187	NFT	\$0.000000433972	↓ 0.01%	\$428,593,548
132	BTT	\$0.000000721484	↓ 0.47%	\$711,447,038
170	MOG	\$0.00000012	↑ 2.24%	\$467,962,918
223	CHEEMS	\$0.000000164	↑ 2.67%	\$333,142,043
32	PEPE	\$0.000001359	↓ 1.12%	\$5,716,655,080
19	SHIB	\$0.000001445	↑ 0.76%	\$8,492,646,110
74	BONK	\$0.000002067	↓ 2.54%	\$1,595,151,520
184	XEC	\$0.000002192	↓ 1.31%	\$435,532,337
225	LUNC	\$0.000006109	↓ 0.04%	\$333,665,388
101	FLOKI	\$0.000009705	↓ 1.24%	\$937,764,890
249	TOSHI	\$0.00066536	↓ 0.73%	\$279,688,475
210	MFW	\$0.00406793	↓ 1.75%	\$361,601,118

Рисунок 3.5 – Сортування таблиці за сортування за зростанням ціни

Дані рисунки демонструють, що сортування за колонками відбувається успішно. Всі три варіанти працюють як в зростанні, так і в спаданні.

Перевірка функціоналу Details екрану

Після переходу на Details екран було виконано перевірку кожного з його компонентів:

Графік зміни ціни (ChartView) (рис.3.6). Очікуваний результат:

- графік коректно відображає зміну вартості монети за останні 7 дні
- присутні маркери максимальної, мінімальної
- середньої ціни

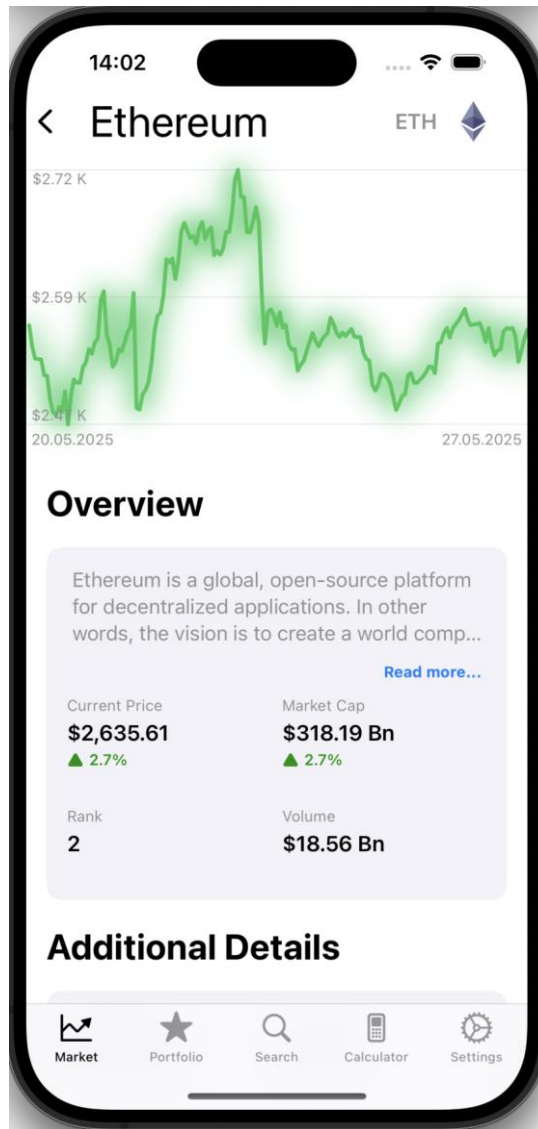


Рисунок 3.6 – Перевірка роботи екрану Details з графіком та додатковими секціями

Секція “Overview”. Очікуваний результат: текст опису криптовалюти відображається у скороченому вигляді, з можливістю розгортання повного тексту (рис.3.7). Статистика виводиться у правильному форматі.

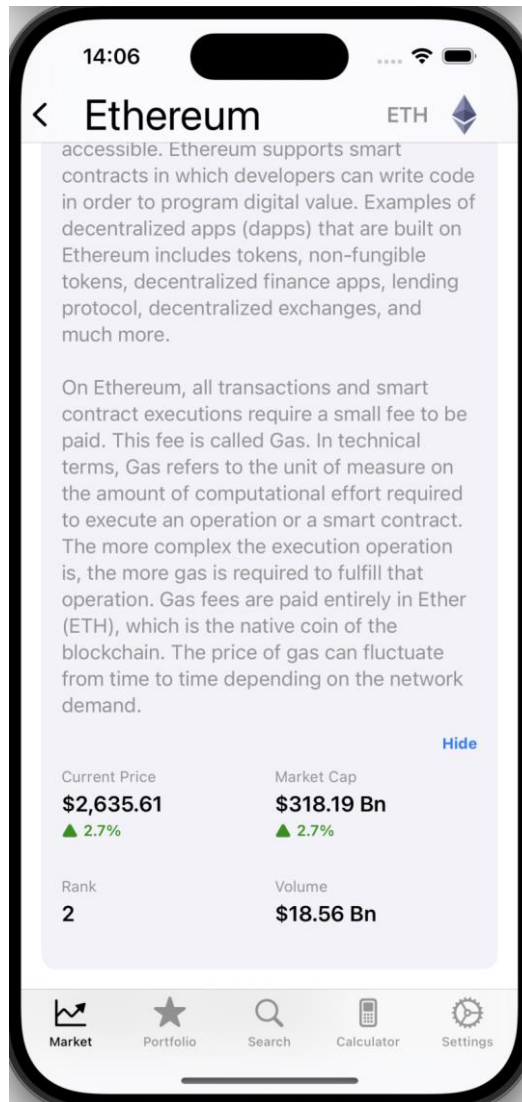


Рисунок 3.7 – Перевірка секції Overview з описом валюти

Секція “Links” (рис.3.8). Очікуваний результат: виводяться активні посилання з назвами.

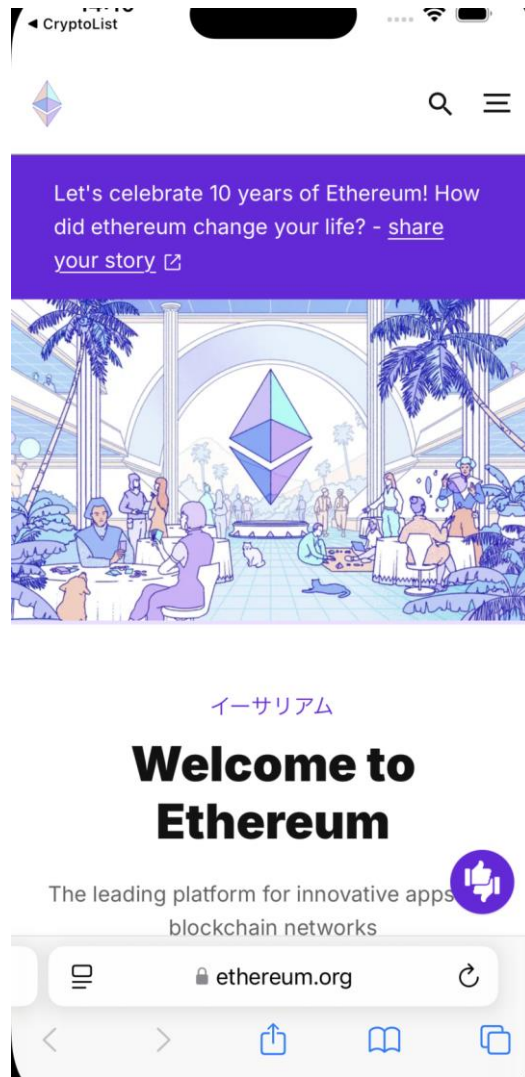


Рисунок 3.8 – Перевірка роботи посилань в секції Links

При натисканні відкривається відповідна сторінка в браузері або у WebView всередині додатку.

3.2 Здійснення додавання валюти в портфель

Наступним етапом перевірки працездатності додатку є Перевірка функціоналу додавання монет до особистого портфеля користувача через екран Portfolio.

Початковий стан

При переході на екран Portfolio, якщо користувач ще не додав жодної криптовалюти, на екрані відображається інформаційне повідомлення з

пропозицією додати актив до портфеля. Повідомлення супроводжується кнопкою “Add” (або іконкою “+” у навігаційній панелі). Очікуваний результат: повідомлення видно лише тоді, коли список портфеля порожній. Кнопка доступна для взаємодії.

Вікно редагування портфеля (Edit Portfolio)

Після натискання кнопки “Add”, з’являється нове модальне вікно — Edit Portfolio (рис.3.9). Його призначення — вибір монети та введення відповідних даних для збереження в базу.

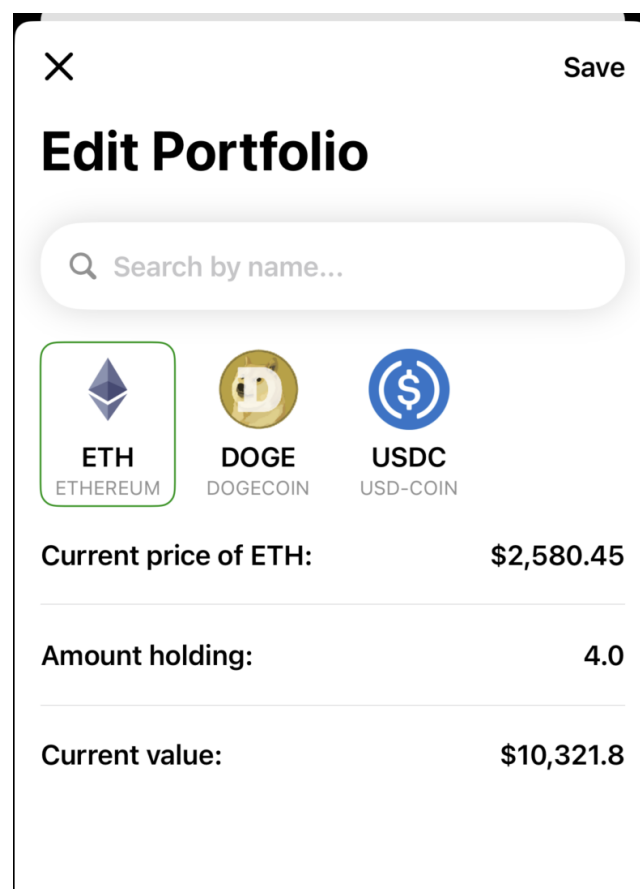


Рисунок 3.9 – Перевірка роботи екрану Edit Portfolio з полями для введення.

У вікні повинне бути поле пошуку монет, реалізоване як горизонтальний список з результатами. Після вибору монети з’являються три поля для введення:

- Current Price — поточна ціна активу (відображається автоматично);
- Amount Holdings — кількість, якою володіє користувач;
- Current Value — розрахована вартість активу на основі попередніх даних.

Також при натисканні на “X” юзера перенаправить назад на сторінку Портфоліо і незбережені зміни будуть видалені.

Збереження монети

Після заповнення всіх полів користувач натискає кнопку “Save”. Очікуваний результат: якщо всі поля заповнено коректно, у верхньому правому куті з’являється іконка “checkmark” (рис.3.10), яка підтверджує успішне збереження.

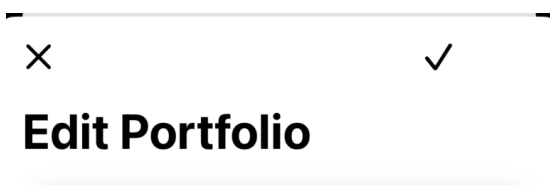


Рисунок 3.10 – Результат після успішного збереження валюти на сторінці Edit portfolio

Дані монети записуються в локальну базу через CoreDataService. ViewModel оновлює стан портфеля та повідомляє екран Portfolio про зміну.

3.3 Перевірка пошуку та калькуляції криптовалют

Однією з важливих функціональних складових додатку є можливість пошуку криптовалют і калькуляція вартості активів у різних валютах. Було проведено перевірка Search Calculator екранів, що дозволило оцінити стабільність і точність їх роботи.

Перевірка функціоналу пошуку полягала в поетапному огляді екрана Search.

Етап 1: Пошук за частковою назвою (рис.3.11). Дія: користувач вводить кілька символів, які можуть знаходитись у різних частинах назви криптовалюти (на початку, в середині або в кінці). Очікуваний результат: динамічно оновлений список монет, що відповідають пошуковому запиту.

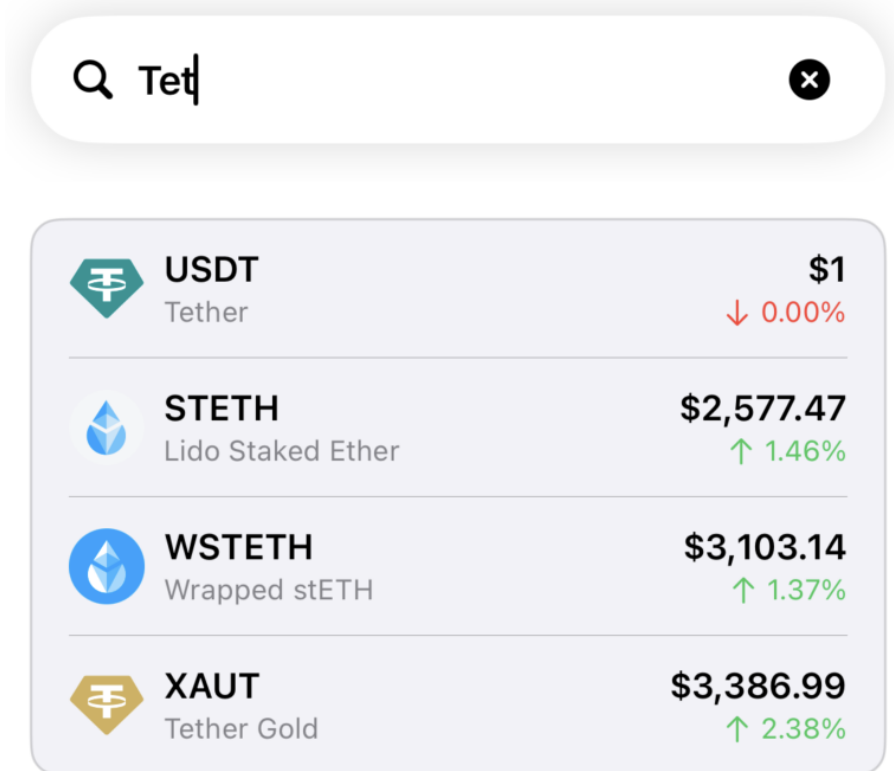
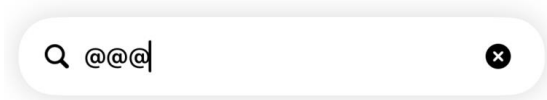


Рисунок 3.11 – Пошук за частковою назвою на сторінці Search

Етап 2: Введення некоректного або неіснуючого значення. Дія: користувач вводить спеціальні символи (наприклад, “#”, “@@”) або слово, якого немає у базі. Очікуваний результат: відображення інформаційного повідомлення про відсутність результатів з відповідною рекомендацією перевірити запит (рис.3.12).



No matching results

Try searching for a different term and check
your spelling

Рисунок 3.12 – Повідомлення про відсутність результатів на екрані
Search

Перевірка функціоналу калькуляції (екран Calculator). Калькулятор дозволяє розрахувати кількість монет, яку користувач може отримати за введену суму у вибраній валюті. Перевірка відбувалося на двох рівнях: основний екран калькулятора та екран вибору валюти.

Етап 1: Основний екран калькулятора. Дія: користувач вводить числове значення у поле суми (рис.3.13). Очікуваний результат: список криптовалют динамічно оновлюється відповідно до розрахованої кількості монет (на основі поточного курсу). Додатково перевірено: поле не приймає літери або спеціальні символи, лише цифрові значення.









	23	USD ▾
 BTC		0.0002
 ETH		0.0089
 USDT		23.0000
 XRP		10.4545
 BNB		0.0346
 SOL		0.1472
 USDC		23.0047
 DOGE		117.9608

Рисунок 3.13 – Перевірка калькуляції на екрані Calculator

Етап 2: Екран вибору валюти. Дія: користувач натискає на кнопку з назвою поточної валюти, відкриваючи новий екран з переліком фіатних та криптовалют (рис.3.14). Очікуваний результат: коректне відображення списку валют; можливість здійснити пошук за назвою валюти (наприклад, “hryvnia”).

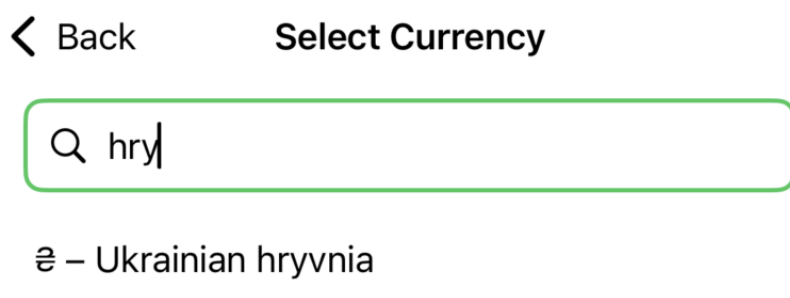


Рисунок 3.14 – Перевірка пошуку по списку валют

Вибір нової валюти автоматично оновлює розрахунки на основному екрані калькулятора.

3.4 Перевірка налаштувань додатку

З метою перевірки коректності роботи екрана налаштувань Settings, було проведено перевірку кожного параметрів. Критерієм успішності тесту була відповідність фактичного результату очікуваному, з урахуванням миттєвих змін інтерфейсу, збереження параметрів, а також підтримки локалізації.

Перевірка темної теми (Dark Mode). Послідовність дій: Користувач відкриває Settings, після чого Активує/деактивує перемикач Dark Mode. Очікуваний результат: Увесь інтерфейс додатку змінює кольорову схему відповідно до обраного режиму (світлий або темний), змінюються кольори фону, тексту та інші візуальні елементи згідно з кастомними налаштуваннями кольорів (рис.3.15). Зміни застосовуються до всіх дочірніх View без необхідності перезапуску додатку.



Рисунок 3.15 – Перевірка вмикання темної теми додатку

Перевірка зміни мови (Language). Послідовність дій: користувач змінює мову інтерфейсу у параметрі Language, далі повертається на головний екран або будь-який інший розділ. Очікуваний результат: усі елементи інтерфейсу, які використовують Localizable.strings, відображаються у відповідному перекладі (рис.3.16). Власні назви не змінюються.



Рисунок 3.16 – Перевірка зміни мови (на іспанську)

Перевірка вибору стартового екрана (Start Screen). Послідовність дій:

- користувач вибирає один із варіантів у полі Start Screen (наприклад, Market, Portfolio тощо);
- Закриває додаток;
- Повторно відкриває додаток.

Очікуваний результат: Після запуску додаток відкривається саме на тому екрані, який було обрано як стартовий, значення зчитується з UserDefaults (рис.3.17) і передається до RootView під час ініціалізації.

√ Root	Dictionary	◇	(4 items)
isDarkMode	Boolean	◇	NO
defaultCurrency	String	◇	¥
languageCode	String	◇	es
startScreen	String	◇	Search

Рисунок 3.17 – Перевірка запису нового значення для startScreen

Перевірка інформаційних сторінок (Privacy Policy, Terms of Service, Disclaimer). Послідовність дій:

- Користувач відкриває один з документів у секції “Інше”;
- Вибирає іншу мову в налаштуваннях;
- Повторно відкриває той самий документ.

Очікуваний результат: Відображається HTML-файл з відповідним перекладом, заголовок сторінки та її вміст відповідають обраній мові та весь текст відображається повністю, без обрізань чи технічних помилок (рис.3.18).

← Back **Política de privacidad**

Última actualización: 09.05.2025

Gracias por usar CoinList (“nosotros”, “nuestro”). Tu privacidad es importante para nosotros. Esta Política de Privacidad explica cómo recopilamos, usamos y protegemos tu información cuando utilizas nuestra aplicación móvil.

1. Descripción general de la aplicación

CoinList es una aplicación de análisis del mercado de criptomonedas. Permite a los usuarios seguir los precios de las monedas, gestionar su portafolio personal de criptomonedas, calcular valores entre monedas y ver información dinámica sobre distintos activos digitales.

2. Límite de edad y público objetivo

La aplicación está dirigida a usuarios de 12 años en adelante. Está diseñada para un

Рисунок 3.18 – Перевірка WebView з Політикою конфіденційності в

Якщо користувач нажме на кнопку “Back”, його поверне на Settings сторінку, де він зможе далі продовжити роботу.

Висновки до розділу

У межах третього розділу було проведено комплексна перевірка основних функціональних модулів iOS-додатку моніторингу фінансових активів. Кожен екран та компонент додатку перевірявся на очікувану поведінку та стабільність роботи.

Зокрема, було перевірено:

– Market екран — на коректність відображення списку криптовалют, оновлення даних через pull-to-refresh, сортування монет та навігацію до екрану Details;

– Details екран — на відображення графіка ціни за останні 7 днів, опису криптовалюти, додаткової інформації та активних зовнішніх посилань;

– Portfolio екран — на працездатність механізму додавання монет, збереження даних, оновлення інтерфейсу та перерахунку статистики;

– Search екран — на пошук по базі криптовалют, а також обробку невалідних чи порожніх запитів;

– Calculator екран — на точність розрахунків, обмеження вводу, зміну валюти та оновлення кількості монет;

– екран налаштувань — на зміну теми (Dark Mode), мови інтерфейсу, вибір стартового екрана та відображення інформаційних документів у обраній локалізації.

Результати перевірки засвідчили стабільність додатку, узгодженість логіки роботи компонентів та реалізацію користувацьких сценаріїв. Усі функції працюють згідно з технічними вимогами, а інтерфейс залишається зручним, адаптивним та локалізованим.

ВИСНОВКИ

У межах виконання бакалаврської роботи було реалізовано iOS-додаток для моніторингу фінансових активів. Під час розробки було вирішено низку практичних, архітектурних та дизайнерських завдань, що дозволили створити сучасний, інтуїтивно зрозумілий та технічно стабільний мобільний застосунок.

У процесі роботи проведено аналіз аналогічних рішень на ринку (CoinGecko, CoinMarketCap), на основі якого визначено функціональні потреби та переваги додатку. Обрано оптимальну архітектуру проєкту та реалізовано шість основних екранів: Market, Portfolio, Search, Calculator, Settings та Details — кожен з яких має продуманий функціонал і взаємодію. Застосовано технології SwiftUI для побудови UI, Combine для реактивного оновлення даних. Забезпечено офлайн-режим роботи, локалізацію інтерфейсу шістьма мовами та гнучкі налаштування інтерфейсу (тема, стартовий екран, мова тощо). Проведено функціональну перевірку кожного компонента, що підтвердило працездатність додатку та відповідність функціональним вимогам.

У результаті виконаної роботи створено стабільний iOS-додаток, який дозволяє користувачам відстежувати стан криптовалютного ринку, керувати власним портфелем та отримувати аналітичну інформацію у зручному й візуально привабливому форматі. Реалізований застосунок є прикладом ефективного поєднання сучасних технологій, хороших практик розробки та орієнтації на зручність кінцевого користувача.

ПЕРЕЛІК ПОСИЛАНЬ НА ДЖЕРЕЛА

1. SwiftUI – офіційна документація: веб-сайт. URL: <https://developer.apple.com/documentation/swiftui> (дата звернення: 24.02.2025)
2. Apple Developer Documentation: SwiftUI Guide URL: <https://developer.apple.com/documentation/swiftui/> (дата звернення: 04.06.2025)
3. Apple Developer Documentation: SwiftUI Tutorials URL: <https://developer.apple.com/tutorials/swiftui/> (дата звернення: 04.06.2025)
4. Apple Developer Documentation: SwiftUI Essentials URL: <https://developer.apple.com/documentation/swiftui/> (дата звернення: 04.06.2025)
5. Sadun E., Gallaher T. SwiftUI for Absolute Beginners (2-nd ed.). URL: <https://link.springer.com/book/10.1007/978-1-4842-5895-1> (дата звернення 04.06.2025 р.)
6. Reda A., Carstens D. Pro SwiftUI. URL: <https://link.springer.com/book/10.1007/978-1-4842-6006-0> (дата звернення 04.06.2025 р.)
7. Allsop D. SwiftUI Animations By Tutorials. URL: <https://www.raywenderlich.com/books/swiftui-animations-by-tutorials> (дата звернення 04.06.2025 р.)
8. Ostrander K., da Prada F. SwiftUI Cookbook. URL: <https://www.packtpub.com/product/swiftui-cookbook/9781803234453> (дата звернення 04.06.2025 р.)
9. Apple Developer Documentation: SwiftUI Overview URL: <https://developer.apple.com/documentation/swiftui/> (дата звернення: 04.06.2025)
10. Apple Developer Documentation: SwiftUI URL: <https://developer.apple.com/documentation/swiftui/> (дата звернення: 04.06.2025)
11. Raywenderlich.com – SwiftUI tutorials: веб-сайт. URL: <https://www.raywenderlich.com> (дата звернення: 24.02.2025)
12. NSHipster – Combine framework overview: веб-сайт. URL: <https://nshipster.com/combine> (дата звернення: 24.02.2025)

13. Apple Developer Documentation: Core Data Guide URL: <https://developer.apple.com/documentation/coredata/> (дата звернення: 04.06.2025)
14. Feiler L. Core Data: Updated for Swift URL: <https://www.oreilly.com/library/view/core-data-updated/9780135262125> (дата звернення 04.06.2025 р.)
15. Apple Developer Documentation: Core Data Essentials URL: <https://developer.apple.com/documentation/coredata/> (дата звернення: 04.06.2025)
16. Core Data – офіційна документація Apple: веб-сайт. URL: <https://developer.apple.com/documentation/coredata> (дата звернення: 24.02.2025)
17. Apple Developer Documentation: Core Data URL: <https://developer.apple.com/documentation/coredata/> (дата звернення: 04.06.2025)
18. Apple Developer Documentation: Core Data Programming Guide URL: <https://developer.apple.com/documentation/coredata/> (дата звернення: 04.06.2025)
19. Apple Developer Documentation: Security URL: <https://developer.apple.com/documentation/security/> (дата звернення: 04.06.2025)
20. Apple Developer Documentation: Security Best Practices URL: <https://developer.apple.com/documentation/security/> (дата звернення: 04.06.2025)
21. Gregoire M. Pro iOS Security and Forensics. URL: <https://link.springer.com/book/10.1007/978-1-4842-5621-4> (дата звернення 04.06.2025 р.)
22. Apple Developer Documentation: Security Overview URL: <https://developer.apple.com/documentation/security/> (дата звернення: 04.06.2025)
23. Apple Developer Documentation: Security Framework URL: <https://developer.apple.com/documentation/security/> (дата звернення: 04.06.2025)
24. Apple Developer Documentation: Security Essentials URL: <https://developer.apple.com/documentation/security/> (дата звернення: 04.06.2025)
25. CoinGecko API Documentation: веб-сайт. URL: <https://www.coingecko.com/en/api/documentation> (дата звернення: 24.02.2025)
26. CoinMarketCap – аналітична платформа криптовалют: веб-сайт. URL: <https://coinmarketcap.com> (дата звернення: 24.02.2025)

27. Localization in iOS – офіційна документація Apple: веб-сайт. URL: <https://developer.apple.com/documentation/foundation/internationalization> (дата звернення: 24.02.2025)
28. Apple Developer Documentation: Localization URL: <https://developer.apple.com/localization/> (дата звернення: 04.06.2025)
29. Apple Developer Documentation: Localization in Xcode URL: <https://developer.apple.com/documentation/xcode/localization> (дата звернення: 04.06.2025)
30. Apple Developer Documentation: Localization Overview URL: <https://developer.apple.com/localization/> (дата звернення: 04.06.2025)
31. Apple Developer Documentation: Localization Guide URL: <https://developer.apple.com/localization/> (дата звернення: 04.06.2025)
32. Apple Developer Documentation: Localization Essentials URL: <https://developer.apple.com/localization/> (дата звернення: 04.06.2025)
33. Apple Developer Documentation: Localization Techniques URL: <https://developer.apple.com/localization/> (дата звернення: 04.06.2025)
34. Apple Developer Documentation: Localization Best Practices URL: <https://developer.apple.com/localization/> (дата звернення: 04.06.2025)
35. Apple Developer Documentation: Supporting Dark Mode in Your Interface URL: <https://developer.apple.com/documentation/uikit/supporting-dark-mode-in-your-interface> (дата звернення: 04.06.2025)
36. Apple Developer Documentation: Dark Mode Support URL: <https://developer.apple.com/documentation/uikit/supporting-dark-mode-in-your-interface> (дата звернення: 04.06.2025)
37. Apple Developer Documentation: Dark Mode Implementation URL: <https://developer.apple.com/documentation/uikit/supporting-dark-mode-in-your-interface> (дата звернення: 04.06.2025)
38. Apple Developer Documentation: Dark Mode Guidelines URL: <https://developer.apple.com/documentation/uikit/supporting-dark-mode-in-your-interface> (дата звернення: 04.06.2025)

39. Apple Developer Documentation: Dark Mode Essentials URL: <https://developer.apple.com/documentation/uikit/supporting-dark-mode-in-your-interface> (дата звернення: 04.06.2025)
40. Medium – статті про iOS-архітектуру, Combine, SwiftUI: веб-сайт. URL: <https://medium.com> (дата звернення: 24.02.2025)
41. GitHub – приклади реалізації архітектури MVVM+C: веб-сайт. URL: <https://github.com> (дата звернення: 24.02.2025)
42. Tsui K. Modern iOS Architecture: Patterns and Best Practices. URL: <https://www.packtpub.com/product/modern-ios-architecture/9781801813223> (дата звернення (дата звернення: 04.06.2025 р)
43. Ben-David L. Designing Crypto Apps for Humans. URL: <https://www.wiley.com/en-us/Designing+Crypto+Apps+for+Humans-p-9781119876543> (дата звернення 04.06.2025 р.)
44. Human Interface Guidelines – дизайн-система Apple: веб-сайт. URL: <https://developer.apple.com/design/human-interface-guidelines> (дата звернення: 24.02.2025)
45. Pione D. Designing Data-Intensive iOS Apps. URL: <https://www.oreilly.com/library/view/designing-data-intensive-ios/9781492092902>
(дата звернення 04.06.2025 р.)

Додатки

Рисунок А.1 – Вигляд екрану Market

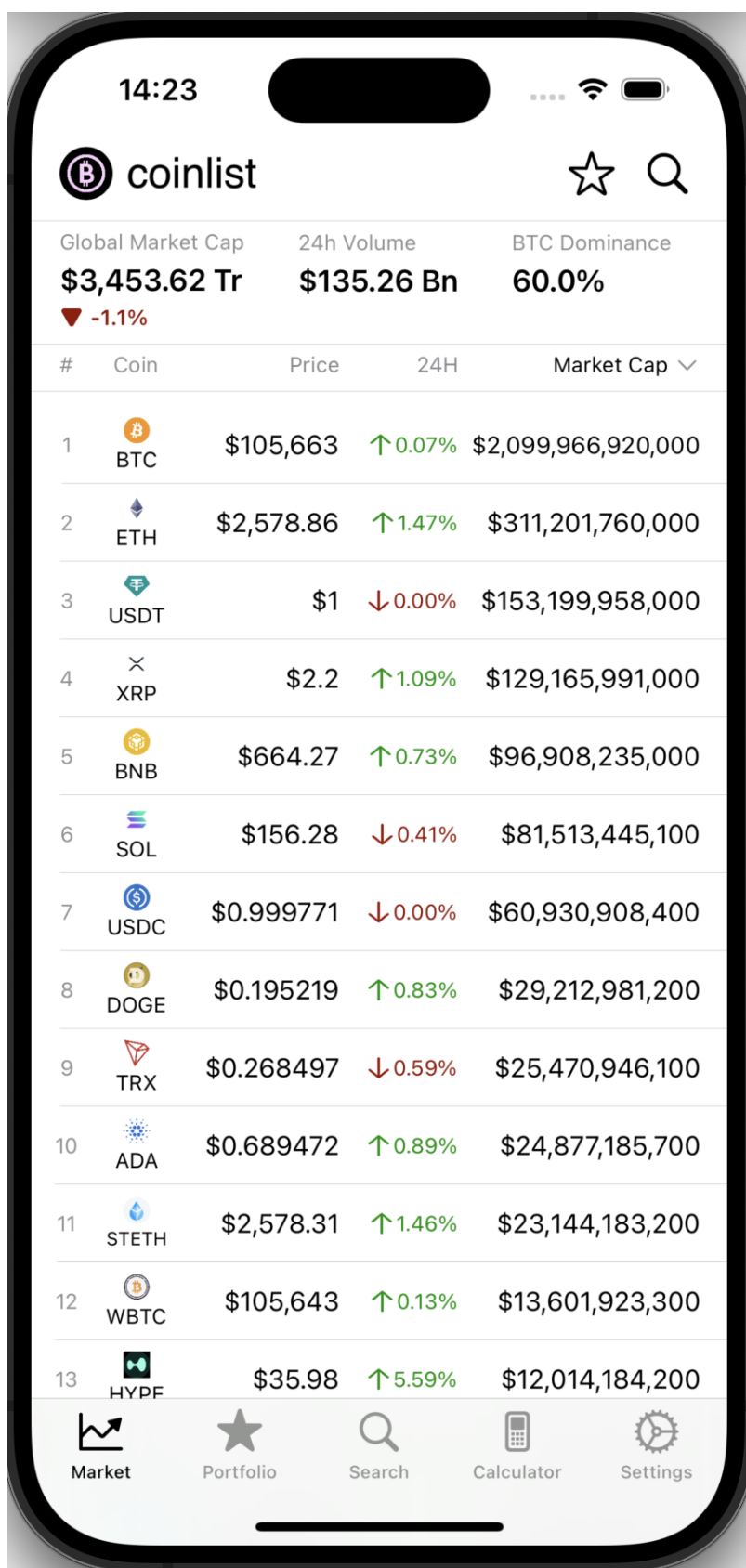


Рисунок А.2 – Вигляд екрану Market в темній темі з локалізацією

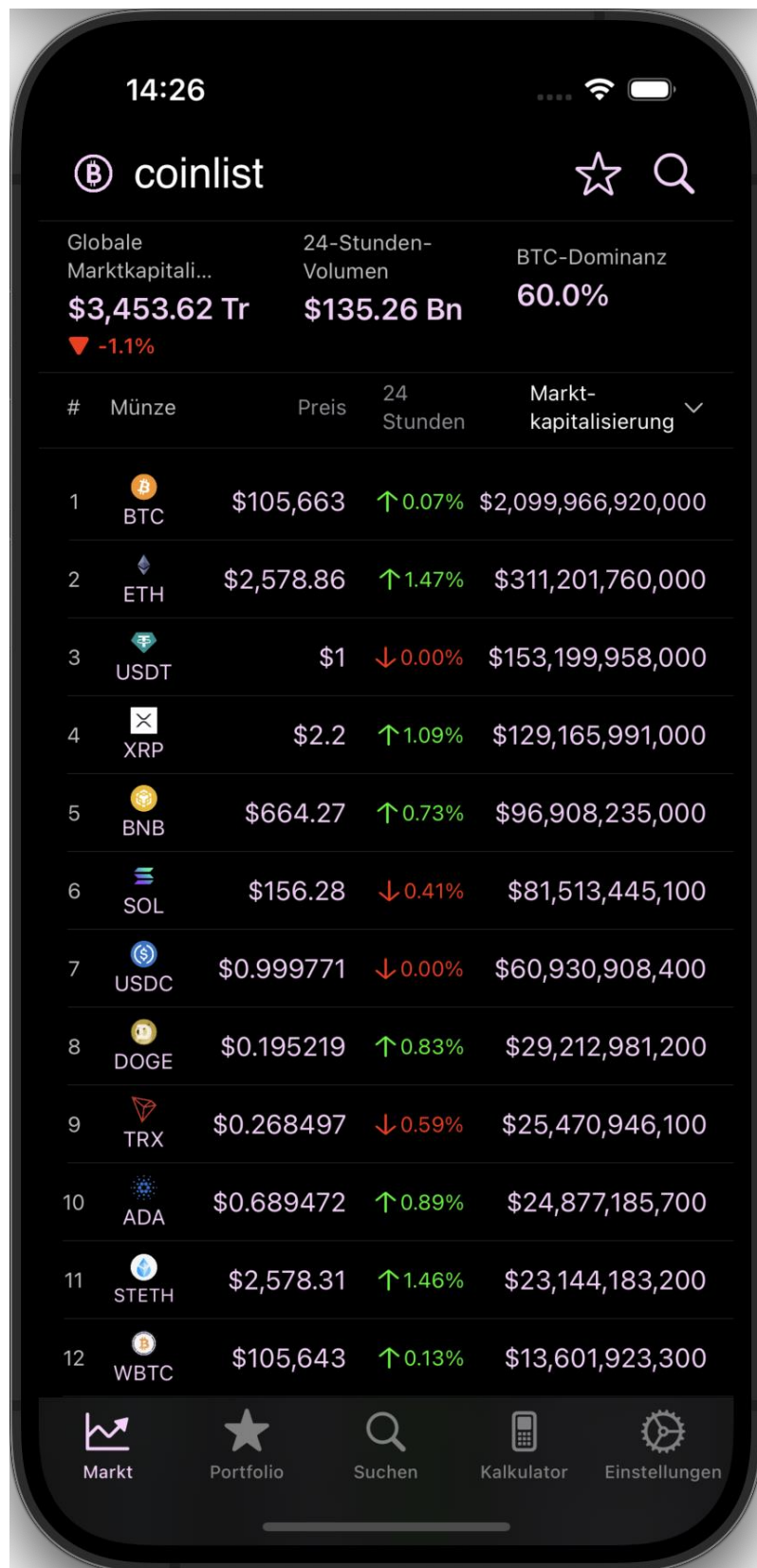


Рисунок Б.1 – Вигляд екрану Portfolio

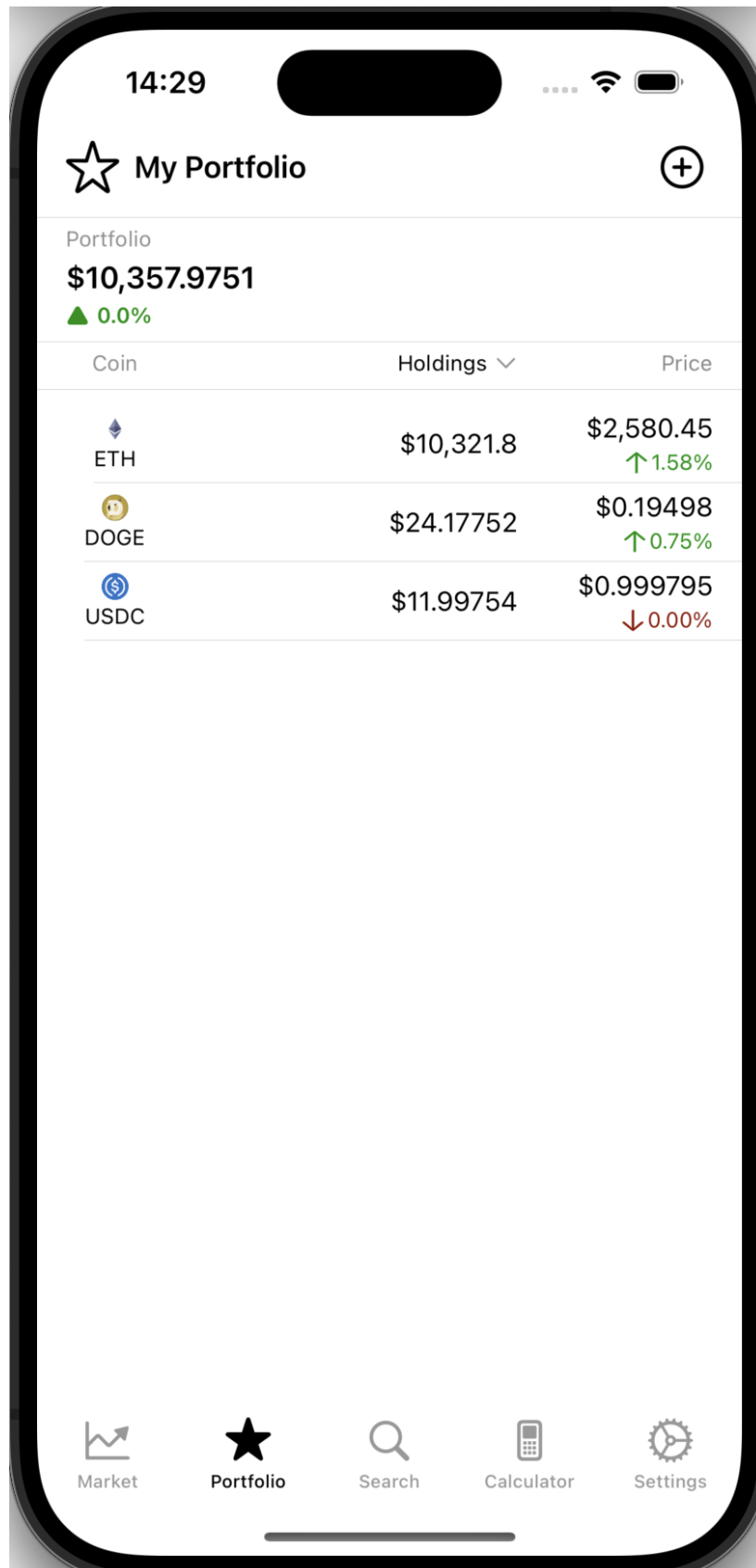


Рисунок Б.2 – Вигляд екрану Portfolio в темній темі з локалізацією

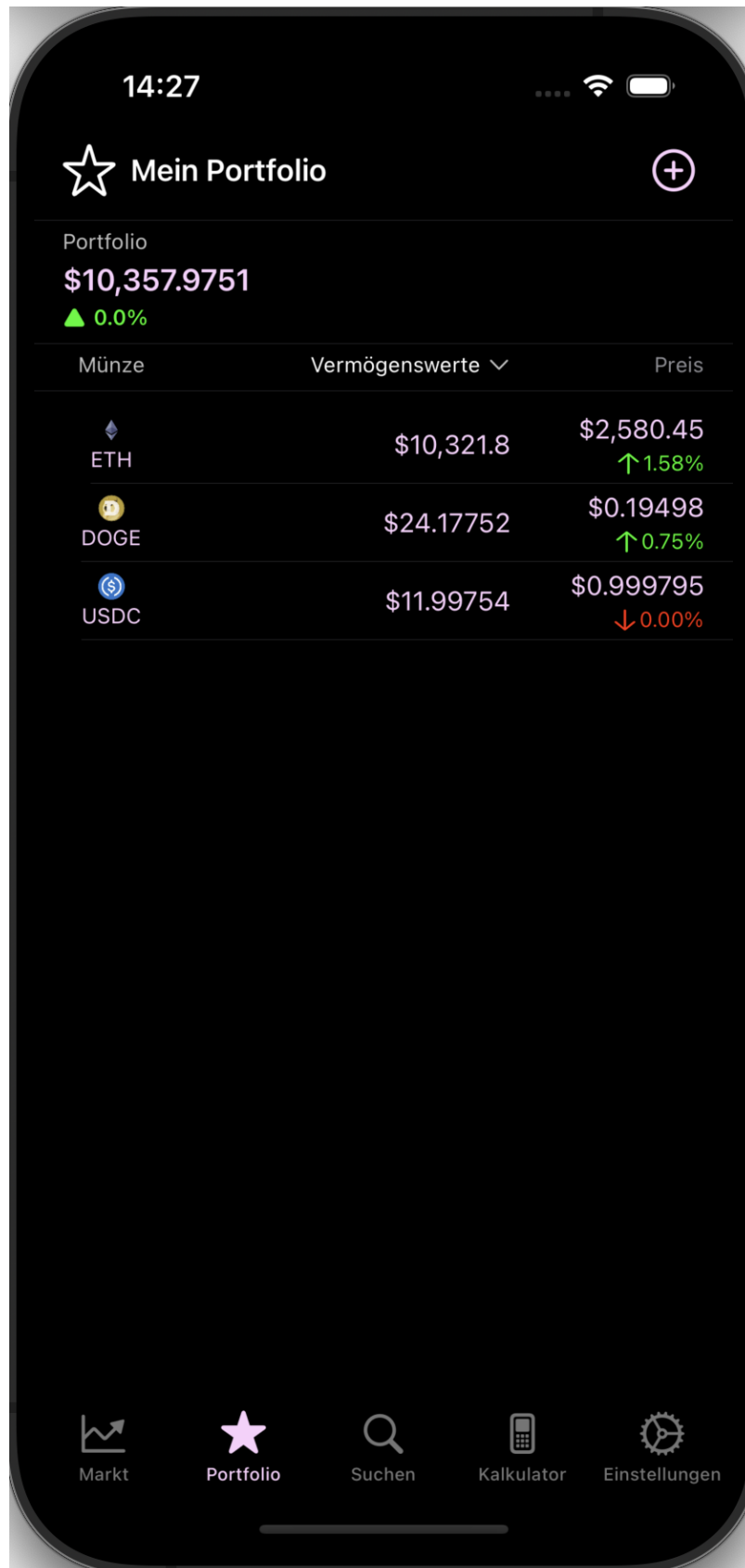


Рисунок Б.3 – Вигляд вікна Edit Portfolio

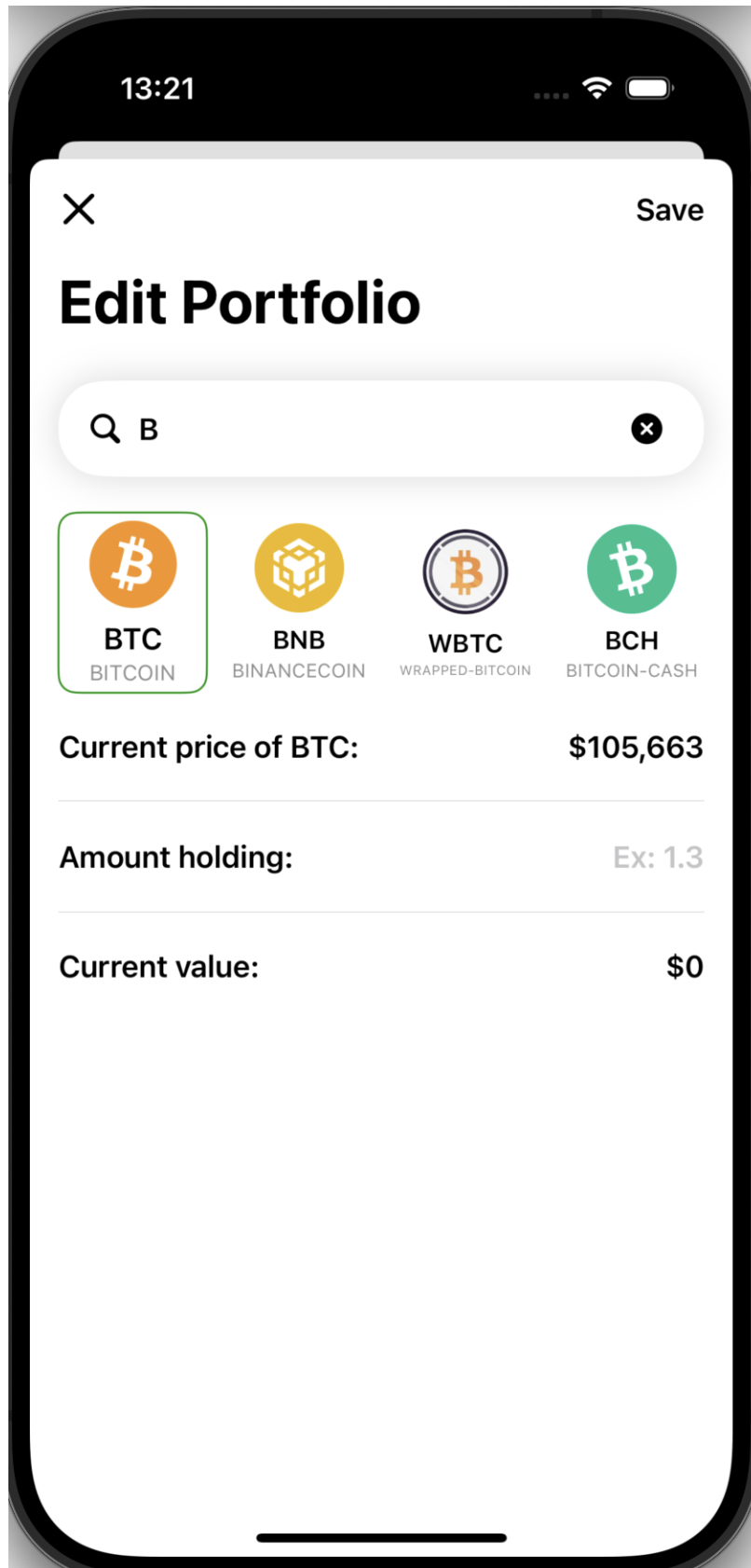


Рисунок Б.4 – Вигляд вікна Edit Portfolio в темній темі з локалізацією

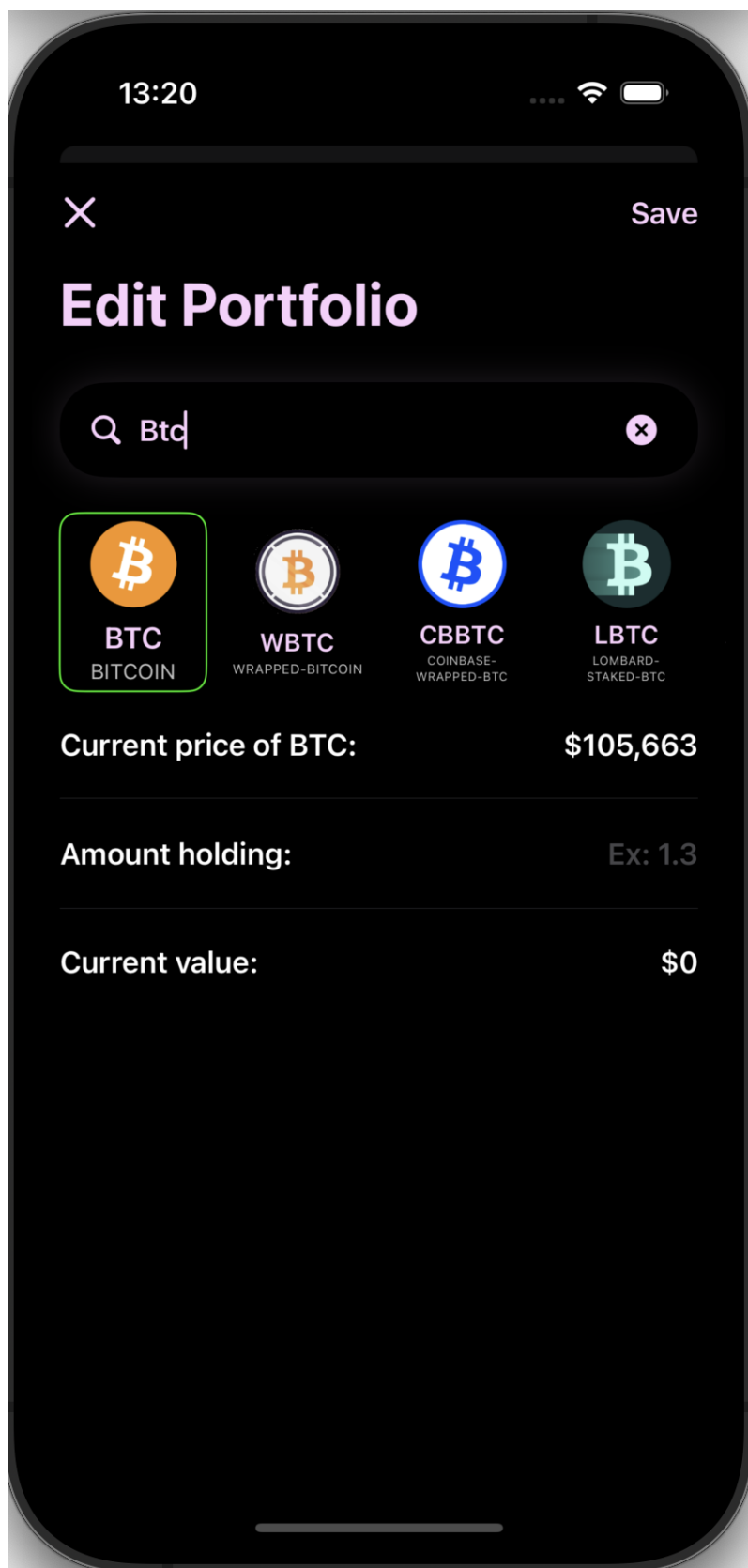


Рисунок В.1 – Вигляд екрану Search

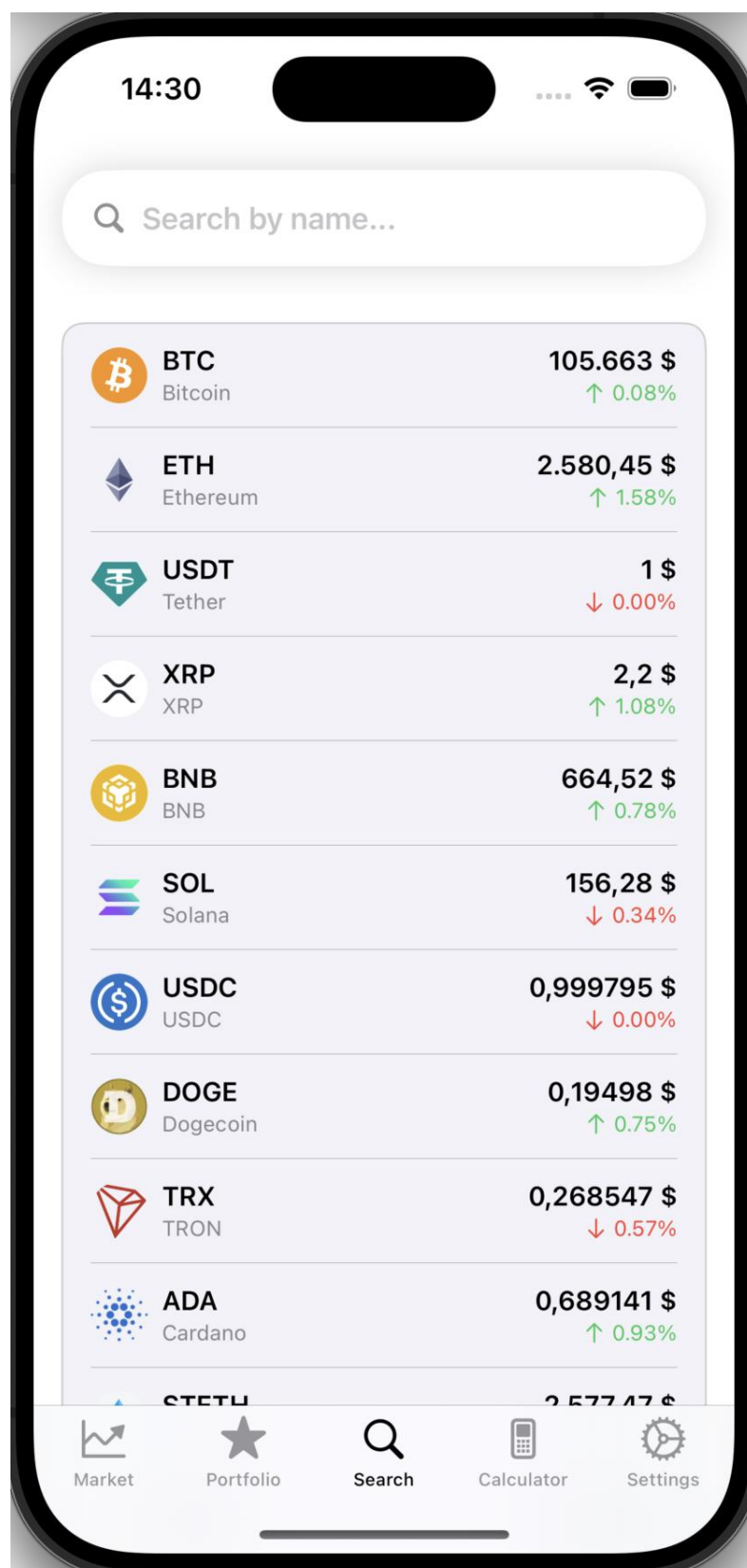













Рисунок В.2 – Вигляд екрану Search в темній темі з локалізацією

14:27



Q Nach Namen suchen...

	BTC Bitcoin	105.663 \$ ↑ 0.08%
	ETH Ethereum	2.580,45 \$ ↑ 1.58%
	USDT Tether	1 \$ ↓ 0.00%
	XRP XRP	2,2 \$ ↑ 1.08%
	BNB BNB	664,52 \$ ↑ 0.78%
	SOL Solana	156,28 \$ ↓ 0.34%
	USDC USDC	0,999795 \$ ↓ 0.00%
	DOGE Dogecoin	0,19498 \$ ↑ 0.75%
	TRX TRON	0,268547 \$ ↓ 0.57%
	ADA Cardano	0,689141 \$ ↑ 0.93%
	STETU	2 577,17 \$



Markt



Portfolio



Suchen



Kalkulator



Einstellungen

Рисунок Г.1 – Вигляд екрану Calculator

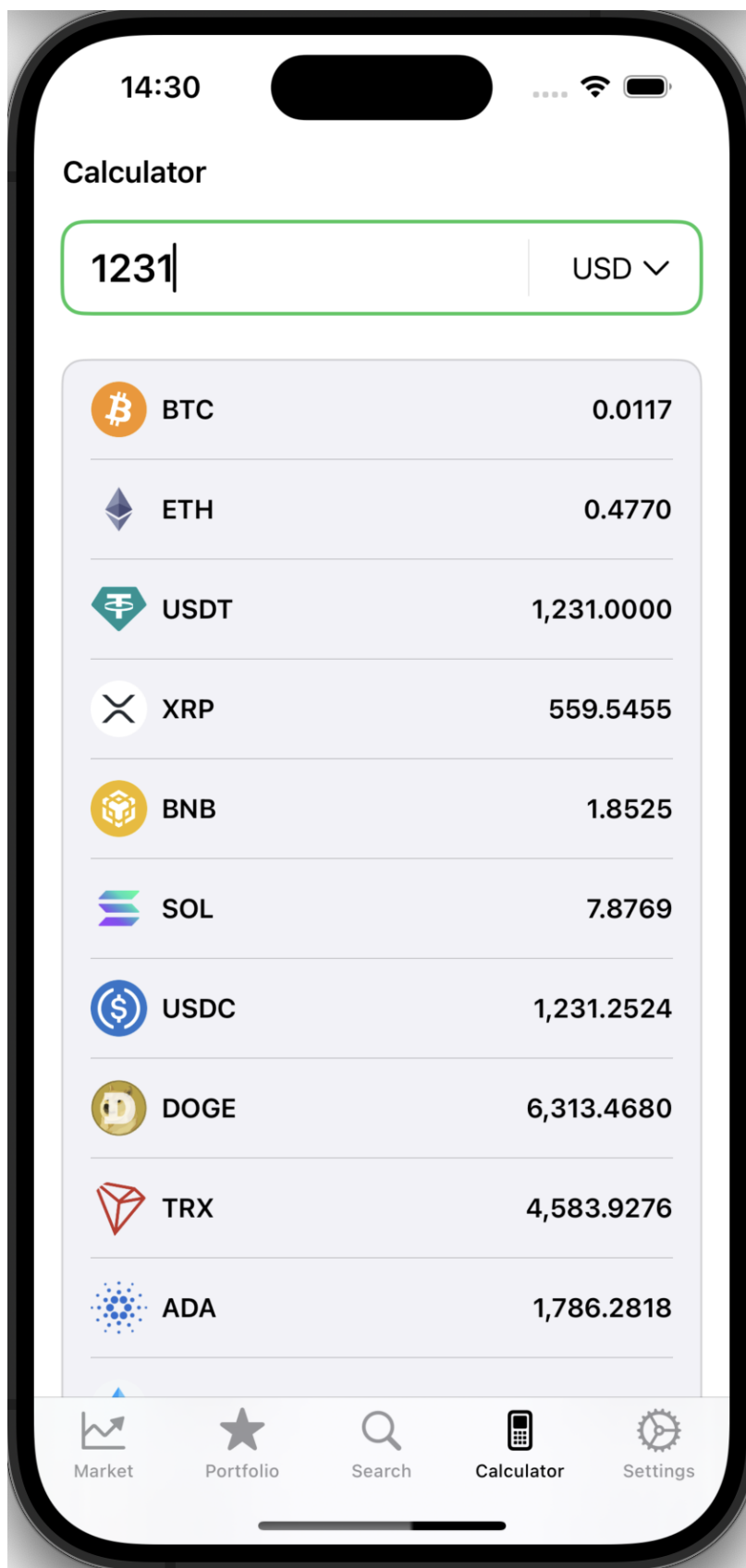


Рисунок Г.2 – Вигляд екрану Calculator в темній темі з локалізацією

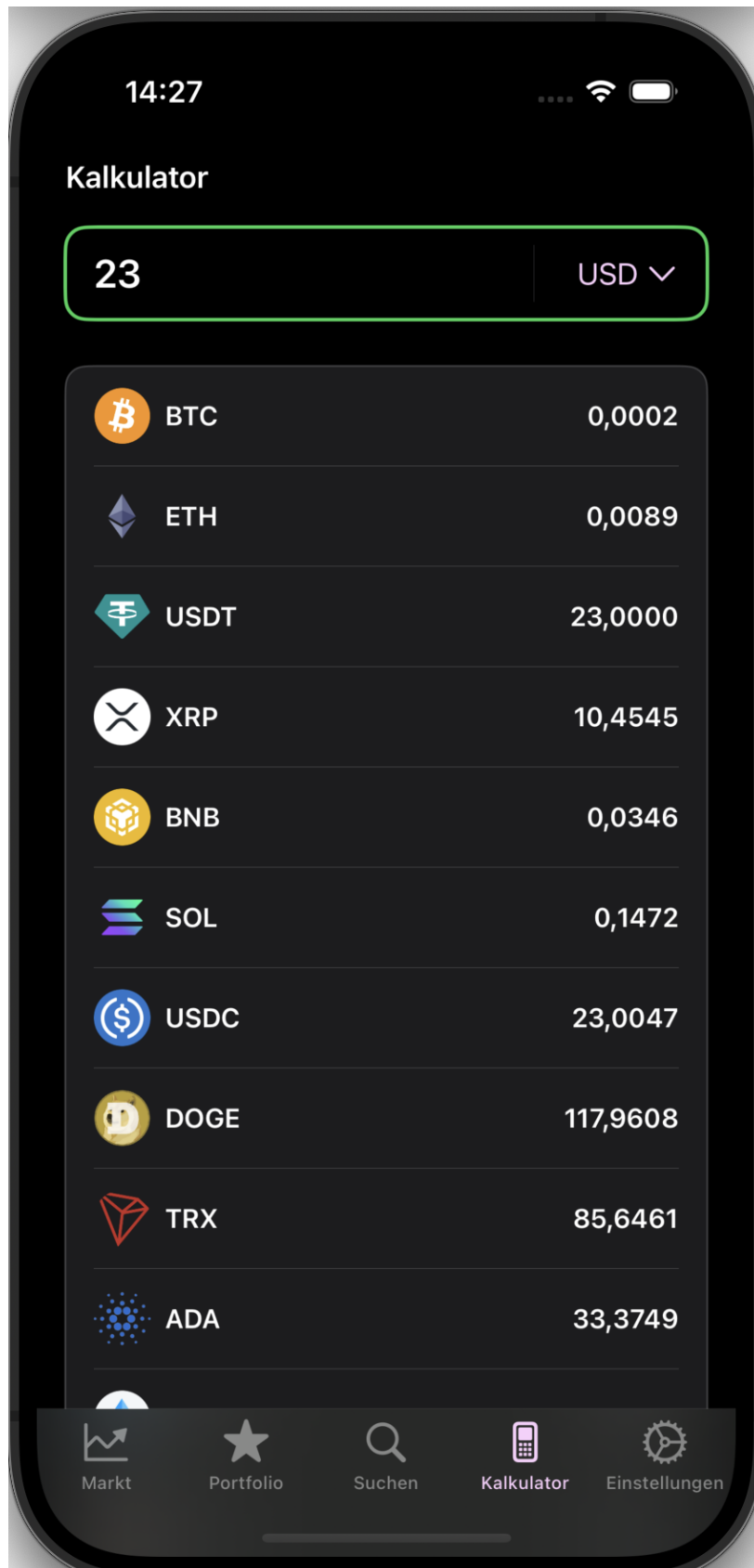


Рисунок Г.3 – Вигляд сторінки вибору валюти

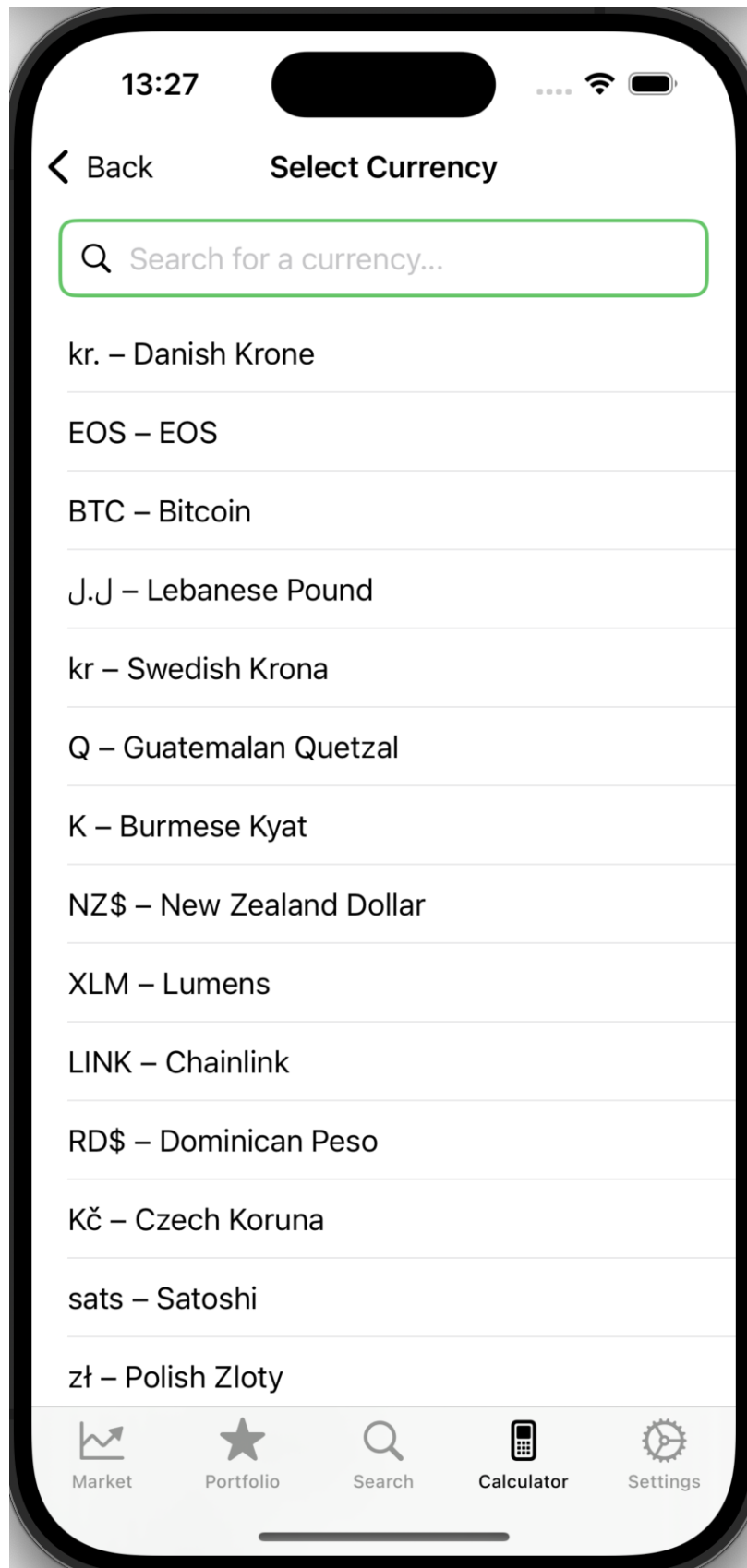


Рисунок Г.4 – Вигляд сторінки вибору валюти в темній темі з локалізацією

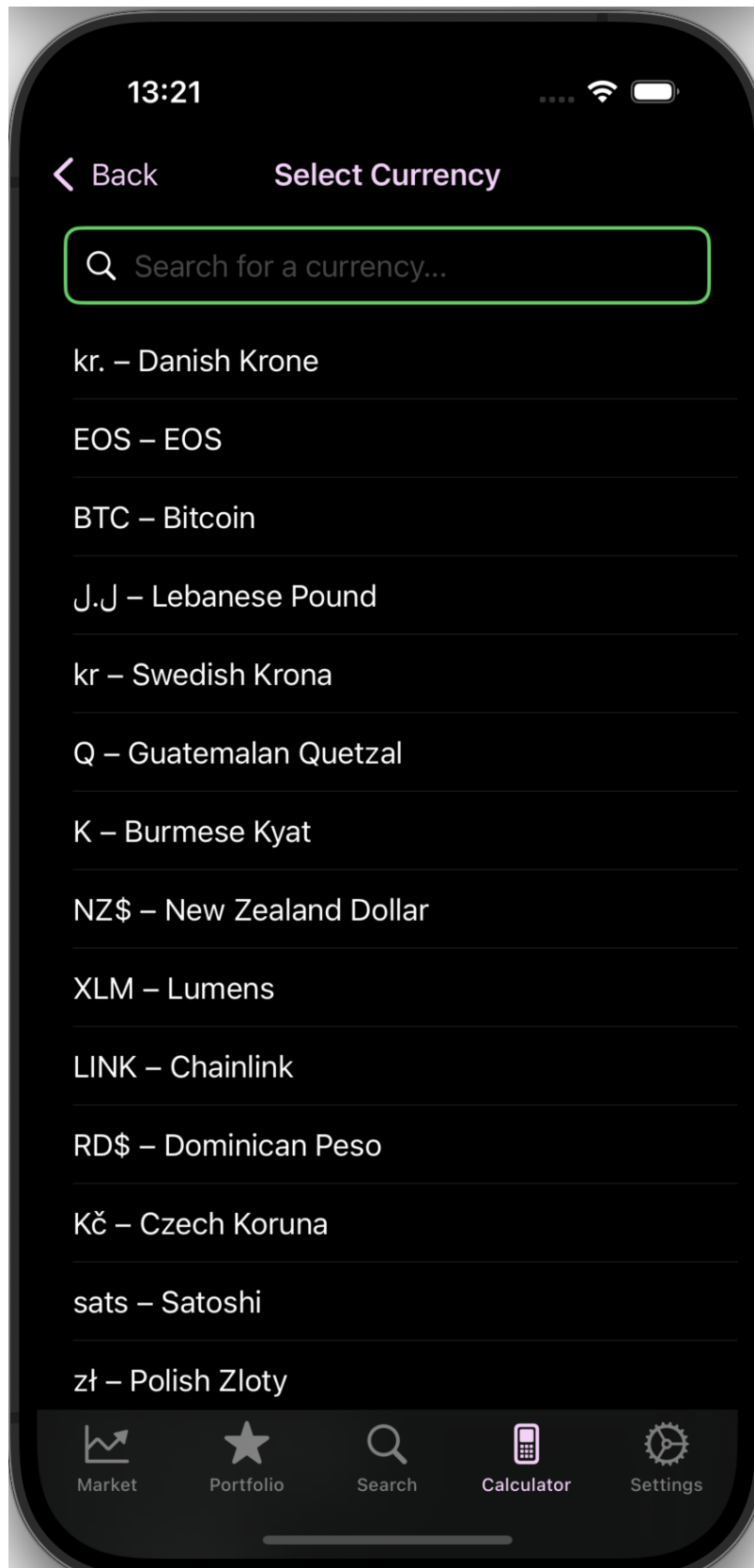


Рисунок Г.1 – Вигляд екрану Settings

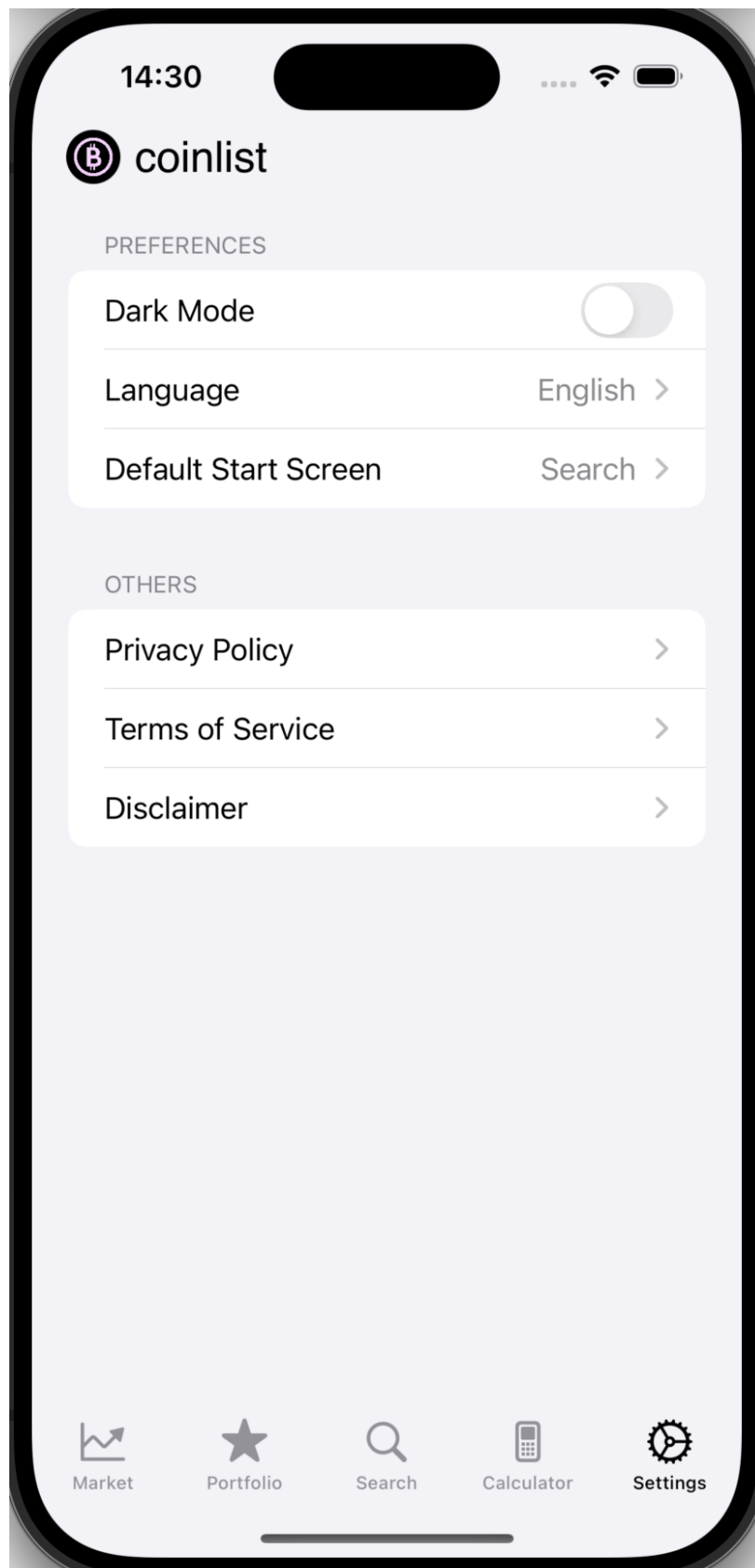


Рисунок Г.2 – Вигляд екрану Settings в темній темі з локалізацією

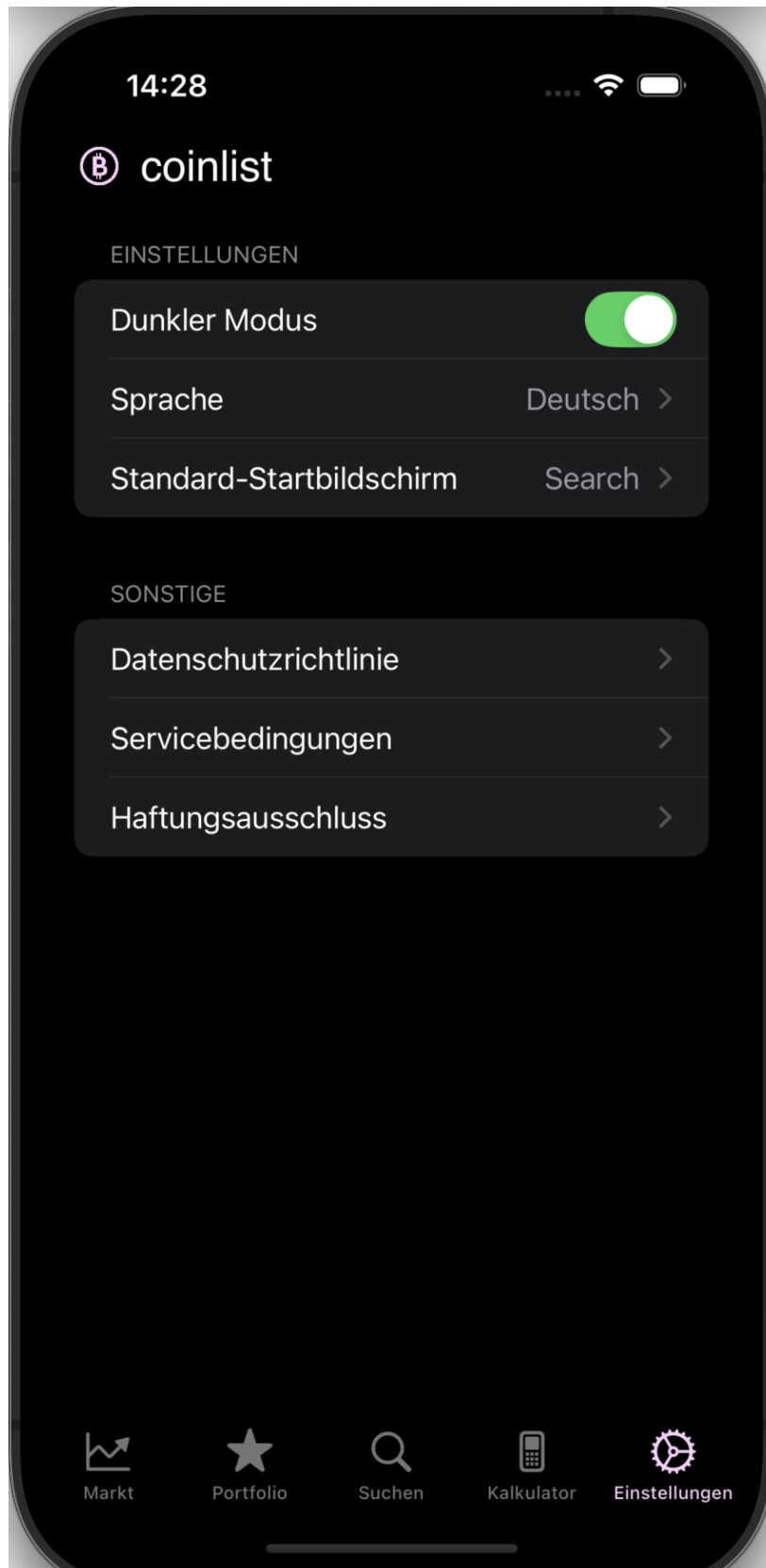


Рисунок Г.3 – Вигляд вікна Privacy policy

13:22



[← Back](#)

Privacy Policy

Last updated: 09.05.2025

Thank you for using CoinList ("we", "us", or "our"). Your privacy is important to us. This Privacy Policy explains how we collect, use, and protect information when you use our mobile application.

1. Application Overview

CoinList is a cryptocurrency market analytics application. It allows users to track coin prices, manage their personal crypto portfolio, calculate value across currencies, and view dynamic information related to various cryptocurrencies.

2. Age Limit and Target Audience

The app is intended for users aged 12 and above. It is designed for general audiences and is not directed specifically at children under 16. There are no financial transactions or real-money operations within the app.



Market



Portfolio



Search



Calculator



Settings

Рисунок Д.1 – Вигляд екрану Details

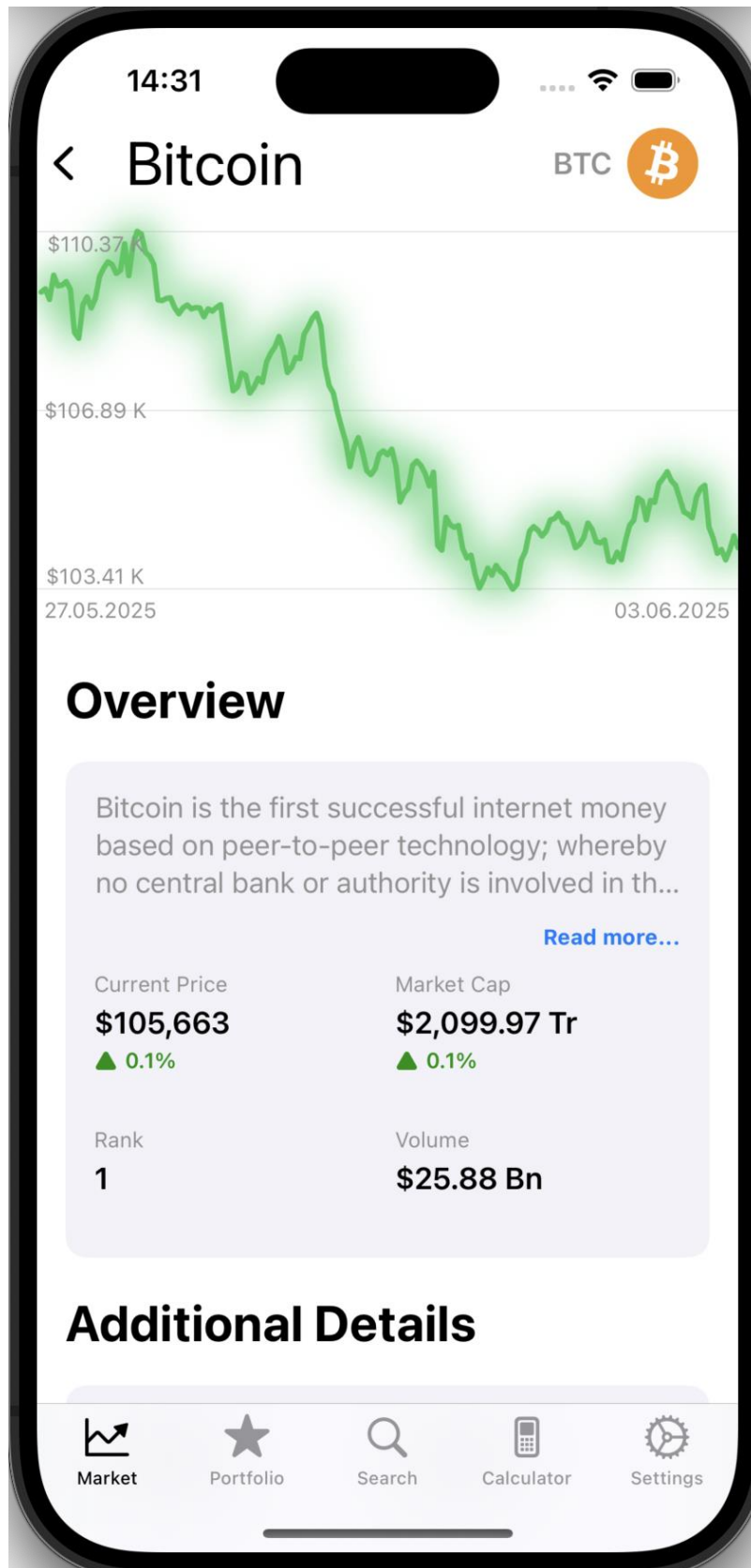
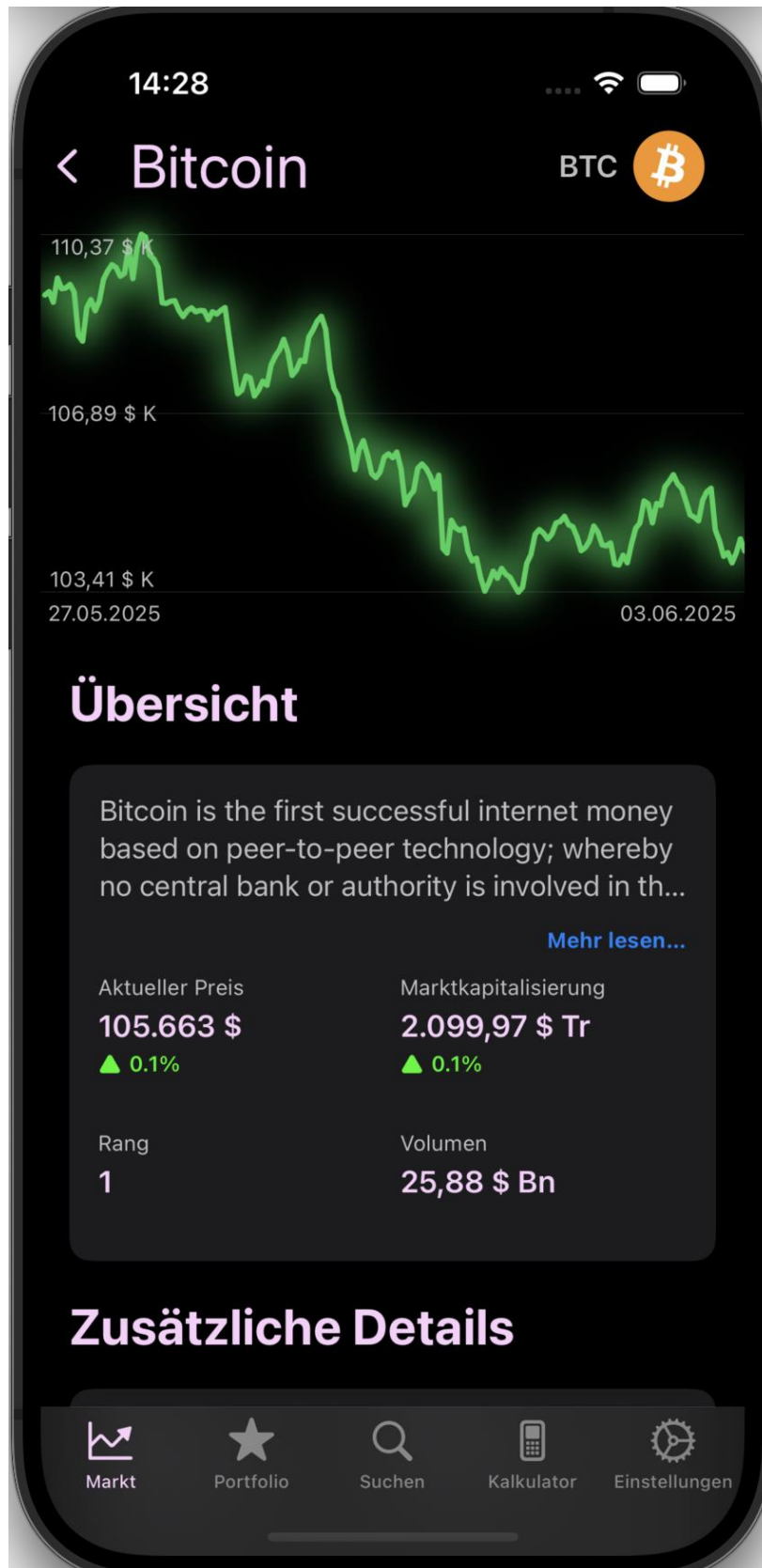


Рисунок Д.2 – Вигляд екрану Details в темній темі з локалізацією



БІБЛІОГРАФІЧНА ДОВІДКА

Тема бакалаврської роботи: **Розробка iOS-додатку моніторингу фінансових активів із динамічною візуалізацією даних.**

Обсяг пояснювальної записки 67 аркуші.

2 таблиць;
39 рисунків;
17 додатків.

Дата завершення роботи: *12 червня 2025 року.*

Підпис студента-дипломника

Бояновський О.В.