



SAPIENTIA  
ERDÉLYI MAGYAR  
TUDOMÁNYEGYETEM  
Marosvásárhelyi Kar

## Cryptocurrency Tracker-Portfolio

Szoftver Rendszerek Tervezése projekt

**Készítette:**

Pakulár Róbert  
Păcurar Alex

**Témavezető:**  
Szántó Zoltán

# Contents

<b>1 Bevezető</b>	<b>2</b>
<b>2 Projekt célja</b>	<b>2</b>
<b>3 Követelmény specifikációk</b>	<b>3</b>
3.1 Felhasználói követelmények . . . . .	3
3.2 Rendszerkövetelmények . . . . .	4
3.2.1 Funkcionális követelmények . . . . .	4
3.2.2 Nem-funkcionális követelmények . . . . .	4
<b>4 Tervezés</b>	<b>5</b>
4.1 Architektúra . . . . .	5
4.2 Frontend modul . . . . .	6
4.3 UI terv . . . . .	7
4.3.1 Fő képernyők . . . . .	7
4.4 Backend modul . . . . .	8
4.5 Adatbázis terv . . . . .	8
4.6 Sequence diagramok . . . . .	9
4.6.1 Login sequence . . . . .	9
4.6.2 Portfolio sequence . . . . .	10
4.6.3 Search sequence . . . . .	10
4.7 Verziókövetés és projekt menedzszerelése . . . . .	11
4.7.1 Jira Tábla – Projekt Állapotkövetés . . . . .	11
4.7.2 GitHub Verziókezelés – Projektfejlesztés . . . . .	11
<b>5 Alkalmazás működése</b>	<b>12</b>
5.1 UI megvalósítás . . . . .	12
5.1.1 Főoldal . . . . .	12
5.1.2 Login / Register . . . . .	12
5.1.3 Coin lista és részletek . . . . .	13
5.1.4 Portfolio kezelés . . . . .	14
5.1.5 Watchlist . . . . .	15
5.1.6 Keresés . . . . .	16
5.1.7 Beállítások . . . . .	16
<b>6 Összegzés</b>	<b>16</b>
<b>7 Ábrák jegyzéke</b>	<b>17</b>
Ábrák jegyzéke . . . . .	17

# 1 Bevezető

A kriptovaluták világa rendkívül dinamikus, gyorsan változó és információ-intenzív környezet, ahol a felhasználóknak naprakész adatokra, átlátható felületekre és megbízható eszközökre van szükségük a tudatos pénzügyi döntésekhez. A különböző forrásokból származó piaci információk követése, az árfolyamok összehasonlítása, a portfólió teljesítményének nyomon követése és a saját figyelőlista kezelése gyakran időigényes, szétszórt és hibalehetőségekkel teli folyamat. Különösen kihívást jelent azok számára, akik több pénznemben gondolkodnak (USD, EUR, HUF, RON), vagy akik kezdeteként szeretnének tiszta, érthető képet kapni a kriptopiacról.

A **Cryptocurrency Tracker & Portfolio** alkalmazás célja, hogy mindezt egyszerűvé, gyorssá és megbízhatóvá tegye. Az alkalmazás egy egységes, könnyen kezelhető webes felületen gyűjti össze a legfontosabb információkat: valós idejű és közel valós idejű piaci adatok megjelenítése, árfolyamok konvertálása több pénznemben, testreszabható figyelőlista (watchlist), valamint portfólió kezelés, ahol a felhasználók rögzíthetik a saját befektetéseiket és követhetik azok értékének alakulását. A rendszer a CoinMarketCap és CoinGecko adatforrásait használja a megbízható és részletes kriptovaluta információkhöz, miközben egy biztonsági mentésként szolgáló "DataDummy" tárolóban a fontosabb coinok adatai lokálisan is elérhetők, hogy külső API kiesés esetén se szakadjon meg a működés.

Kiemelt célunk egy stabil, gyors és letisztult dashboard kialakítása, amely egy helyen foglalja össze a legfontosabb információkat: aktuális árfolyamok, piacmozgások, 24 órás változások, piaci kapitalizáció, rangsorok, valamint a felhasználó saját portfóliójának és figyelőlistájának állapota.

# 2 Projekt célja

A projekt célja egy megbízható, átlátható és hatékony webes rendszer megvalósítása, amely a kriptovaluta-piac követését és a személyes befektetések kezelését egyetlen, jól használható felületen biztosítja.

- **Valós idejű kriptovaluta lista és statisztikák megjelenítése**

A rendszer naprakész piaci adatokat mutat (ár, 24h változás, piaci kapitalizáció, rang, forgalom), gyors frissítési ciklussal és kliens-szerver oldali cache-eléssel a felesleges API-hívások csökkentésére. Hálózati hiba esetén DataDummy mentés szolgál tartalékként, hogy az alapinformációk továbbra is elérhetők legyenek.

- **Személyes watchlist és portfólió funkciók**

A felhasználó hitelesítés után saját watchlistet és portfóliót kezelhet. Az adatok felhasználónként szeparáltan tárolódnak; a hozzáférés JWT-vel védett, a jelszavak bcrypt-tel hashelt formában kerülnek tárolásra. A portfólió támogatja a mennyiségek, átlagárak és hozamok követését, míg a watchlist gyors áttekintést ad a kiemelt coinokról.

- **Több pénznem konzisztens kezelése és megjelenítése**

A rendszer több deviza (pl. USD, EUR, HUF, RON) egységes kezelését biztosítja. Az átváltás egységes konverziós szabályokkal történik, az adatok bázisdevizában

(USD) kerülnek tárolásra, a felhasználói felületen pedig választott pénznem sz-erint, konzisztens kerekítéssel és formázással jelennek meg. A pénznem választás perzisztálódik.

- **Stabil és reszponzív felhasználói felület**

A cél egy gyors, reszponzív dashboard minimális vizuális villogással. Skeleton és shimmer betöltők, inkrementális frissítések, animált értékváltások és lista-virtualizáció csökkentik a vibrálást és javítják az észlelt teljesítményt.

Ezen célok összehangolt megvalósítása egy olyan eszközt eredményez, amely kezdőknek és haladóknak egyaránt megbízható alapot nyújt a kriptopiaci döntésekhez.

## 3 Követelmény specifikációk

### 3.1 Felhasználói követelmények.

**Áttekintés:** A felhasználó egy könnyen használható dashboardon böngészni a kriptovalutákat, részleteket néz meg, keres, és bejelentkezés után saját watchlistet és portfóliót kezel. (Lásd 3.1.1)

#### Nyilvános funkciók:

- Kriptovaluta lista megtekintése (ár, 24h változás, market cap, rang, forgalom)
- Keresés név/szimbólum alapján, javaslatokkal
- Coin részletek oldal: kibővített adatok

#### Bejelentkezés után:

- Watchlist kezelés: hozzáadás/eltávolítás, gyors áttekintés
- Portfólió kezelés: coinok rögzítése, mennyiségek, vételi ár, aktuális érték és hozam
- Beállítások: megjelenítési pénznem (USD, EUR, HUF, RON), UI preferenciák

#### Biztonság és adatszeparáció:

- Felhasználói adatok elkülönített tárolása (JWT-alapú hozzáférés, bcrypt jelszavak)
- Kliens oldali preferenciák LocalStorage-ban (nem érzékeny adatok)

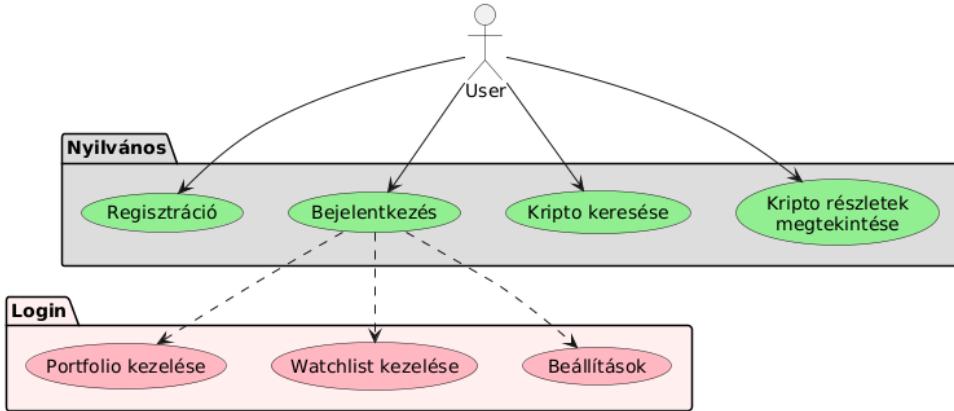


Figure 3.1.1: Use Case diagram

## 3.2 Rendszerkövetelmények

### 3.2.1 Funkcionális követelmények

- **Kriptovaluta lista**

A rendszer lekéri és megjeleníti az aktuális árakat, 24h változást, market cap-et, forgalmat és rangot. A listában meg lehet jeleníteni 10/25/50 elemet az ablak alján.

- **Keresés**

Keresés a listában név/szimbólum alapján;

- **Coin részletek**

Egy coin részletes adatainak megjelenítése, képek és grafikonnal ábrázolt árfolyamokkal.

- **Bejelentkezés és regisztráció**

Email+jelszó, jelszó minimum 6 karakter, JWT token visszaadása, lastLogin mentése.

- **Watchlist kezelés**

Bejelentkezés után coin hozzáadása/eltávolítása a watchlistből; adatok felhasználónként tárolva.

- **Portfólió kezelés**

Bejelentkezés után coinok mentése a portfólióba (mennyiség, vételi ár), aktuális érték és hozam számítása.

- **Beállítások**

Pénznem választása és mentése; UI preferenciák megválasztása (Dark/Light mode).

### 3.2.2 Nem-funkcionális követelmények

- **Teljesítmény**

Lista frissítése 5–6s intervallumonként.

- **Megbízhatóság**

API kiesés esetén DataDummy backupból alapadatok elérhetőek. Cache TTL-ek kontrolláltak (coins: 10s; sparkline: 1h; map: 24h).

- **Biztonság**  
Jelszó bcrypt-tel hashelve; JWT auth, role kezelés („user”, „admin”). CORS és input validáció minden endpointon.
- **Használhatóság**  
Reszponzív UI, minimális vizuális villogás. Skeleton/shimmer, konzisztens pénznem formázás.
- **Skálázhatóság**  
Horizontális skálázás backendnél; cache csökkenti API terhelést.
- **Karbantarthatóság**  
Tiszta komponensstruktúra, logikus route-ok.
- **Adatintegritás**  
Felhasználónkénti adatszeparáció; validáció a portfólió/beállítások mentésénél.

## 4 Tervezés

### 4.1 Architektúra

**Áttekintés:** A rendszer a modern **MERN stack**-re épül, azaz MongoDB, Express.js, React és Node.js technológiákat használ. A szoftver három fő rétegből áll: frontend (React), backend (Node.js + Express) és adatbázis (MongoDB). Külső adatforrások: CoinMarketCap (piaci adatok) és CoinGecko (sparkline). (Lásd 4.1.1)

#### Komponensek:

- **Frontend:** Pages és Components (Dashboard, Portfolio, Login/Register, Coin Details, Settings), Context (Settings), Utils (portfolio, watchlist).
- **Backend:** REST API (auth, coins, search, sparkline), Auth modul (JWT + bcrypt), Cache (in-memory Map TTL).
- **Adatbázis:** users kollekció; DataDummy backup a kriptovaluta adatokhoz.

#### Kapcsolatok:

- React → Express: HTTP/REST hívások
- Express → CoinMarketCap/CoinGecko: HTTPS lekérések
- Express → MongoDB: Mongoose
- React → LocalStorage: kliens oldali persistencia (beállítások, portfolio/watchlist)

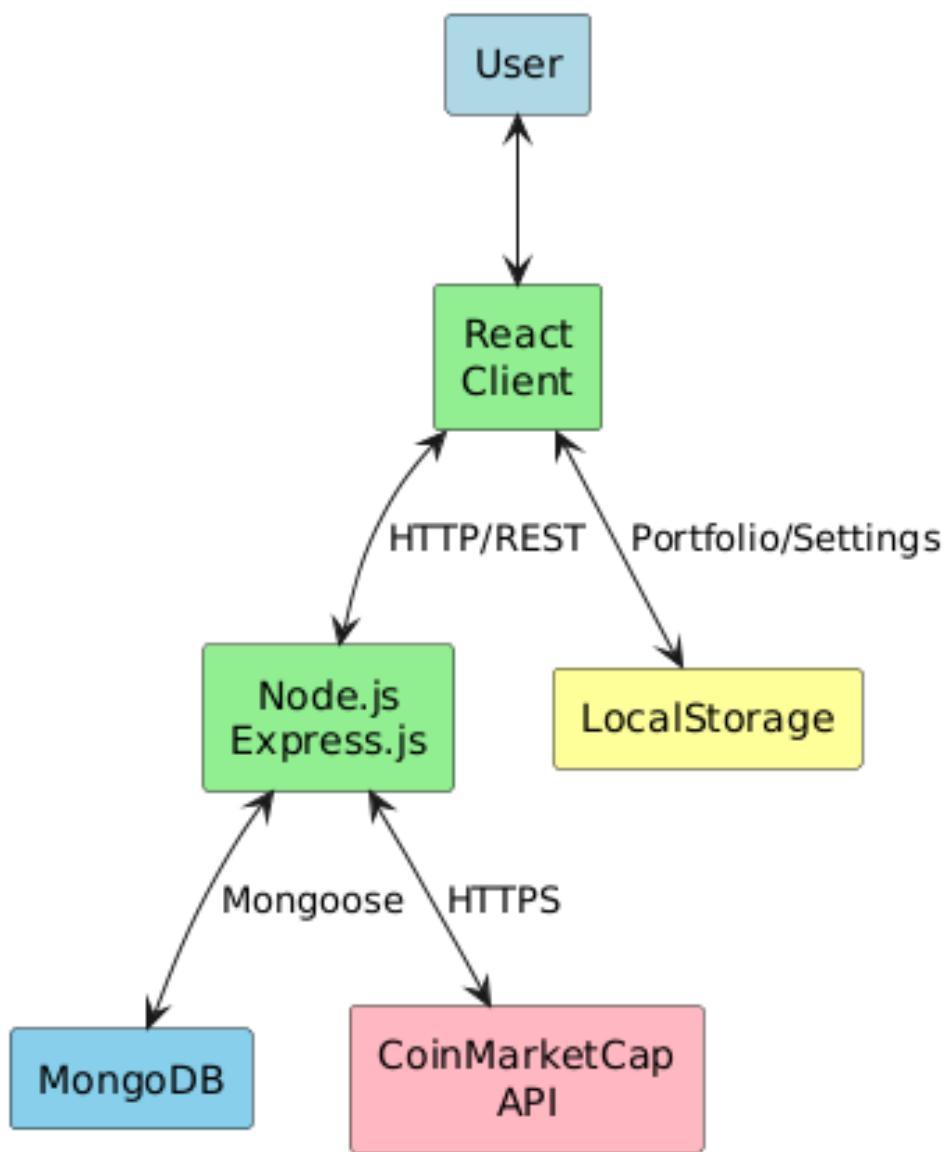


Figure 4.1.1: Komponens diagram

## 4.2 Frontend modul

Pages:

- **Dashboard:** Kriptók listája, statisztikák, keresés, automatikus frissítés 5–6s jitterrel.
- **Portfolio / PortfolioPage:** Felhasználói portfólió áttekintése és kezelése.
- **Login / Register:** Hitelesítés, token mentés, hibakezelés.
- **CoinDetails:** Részletes coin információk, sparkline megjelenítés (CoinGecko).
- **Settings:** Pénznem választás (USD/EUR/HUF/RON), UI preferenciák.

## Components:

- **Navbar:** Navigáció és autentikációs állapot kezelése.
- **CryptoTable:** Táblázatos lista, rendezés, „add to portfolio” interakció.
- **StatCard:** Összesített mutatók megjelenítése.
- **PortfolioSummary:** Portfólió aggregált adatok.

## Context / Utils:

- **SettingsContext:** Globális pénznem beállítás.
- **utils/portfolio.js:** get/set/add/remove/isInPortfolio funkciók LocalStorage-ban.
- **utils/watchlist.js:** Watchlist CRUD kliens oldalon.

## 4.3 UI terv

**Célok:** Tiszta, reszponzív felület; minimális vizuális villogás frissítések során; konzisztens tipográfia és színvilág.

### Fő irányelvezek:

- **Layout:** Rácsalapú elrendezés, sticky navbar, tartalomkártyák.
- **Állapotkezelés:** Toast értesítések hibákhoz/sikerhez; skeleton/shimmer betöltés.
- **Frissítés:** Inkrementális adatmódosítás (nem teljes re-render), listákban kulcsok stabilan.
- **Pénznem megjelenítés:** Locale-aware formázás, konzisztens kerekítés, rövidített kompakt megjelenítés nagy számoknál.

**Stílus:** Sötét/Világos téma támogatás, kontrasztos elemek, hozzáférhetőség.

### 4.3.1 Fő képernyők

#### Fő képernyők:

- **Dashboard:** Lista + statisztikák + kereső + gyorselemek.
- **Portfolio:** Felhasználói eszközök, összérték, profit/loss, gyors műveletek.
- **Watchlist:** Kedvenc coinok gyors áttekintése, hozzáadás/eltávolítás.
- **CoinDetails:** Részletes adatok, sparkline, meta információk.
- **Auth (Login/Register):** Egyszerű form, visszajelzések, átirányítás.
- **Settings:** Valuta és vizuális beállítások.

## 4.4 Backend modul

Routes / Controllers:

- **POST /api/auth/register:** Validáció, user létrehozás, JWT generálás.
- **POST /api/auth/login:** Bcrypt jelszó összehasonlítás, lastLogin frissítés, JWT visszaadása.
- **GET /api/coins:** CoinMarketCap listings lekérése, adat transzformálás, cache (10s TTL).
- **GET /api/coins/:id:** CoinMarketCap info + quotes transform, cache (60s TTL).
- **GET /api/search:** CMC map cache (24h TTL), filter/sort, top 20 találat.
- **GET /api/coins/:id/sparkline:** CoinGecko történeti árak, cache (1h TTL).
- **GET /api/health:** Port és DB státusz ellenőrzés, ok visszaadása.

Services:

- **Cache:** In-memory Map {ts, ttl, data}, egyszerű TTL érvényesítés.
- **Upstream fetch:** Hibakezelés és válasz ellenőrzés.

Models:

- **User:** Mongoose schema (username, email, password [bcrypt], role, isActive, lastLogin, timestamps).

Adatfolyam:

- Frontend kérés → Backend cache ellenőrzés → API hívás (ha kell) → adat transzformálás → válasz a kliensnek.

## 4.5 Adatbázis terv

MongoDB séma (Lásd 4.5.1):

- **User kollekció:**
  - Mezők: username, email, password (bcrypt), role, isActive, lastLogin, createdAt, updatedAt
  - Indexek: email, username
- **DataDummy kollekció:**
  - Fontos kripták mentett adatai: id, name, symbol, current\_price, market\_cap, market\_cap\_rank, total\_volume, price\_change\_percentage\_24h, image, timestamps
  - API-kiesés esetére szolgáló tartalék

## Adatvédelmi irányelvez:

- Bcrypt jelszó hashing
- JWT tokenek kliensen tárolva; érzékeny adat nem kerül LocalStorage-ba

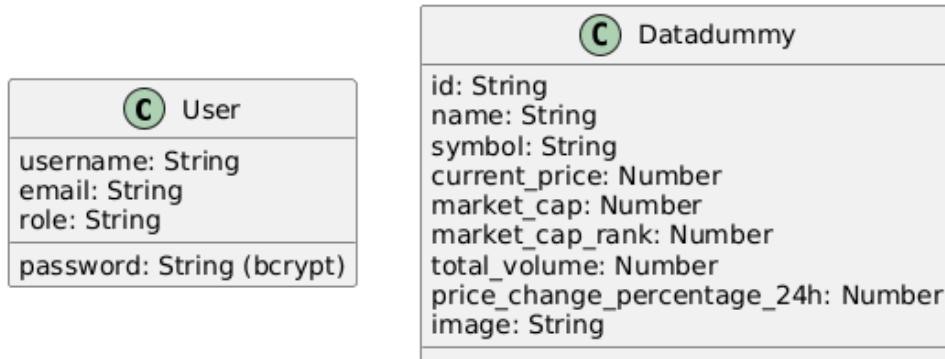


Figure 4.5.1: User diagram

## 4.6 Sequence diagramok

### 4.6.1 Login sequence

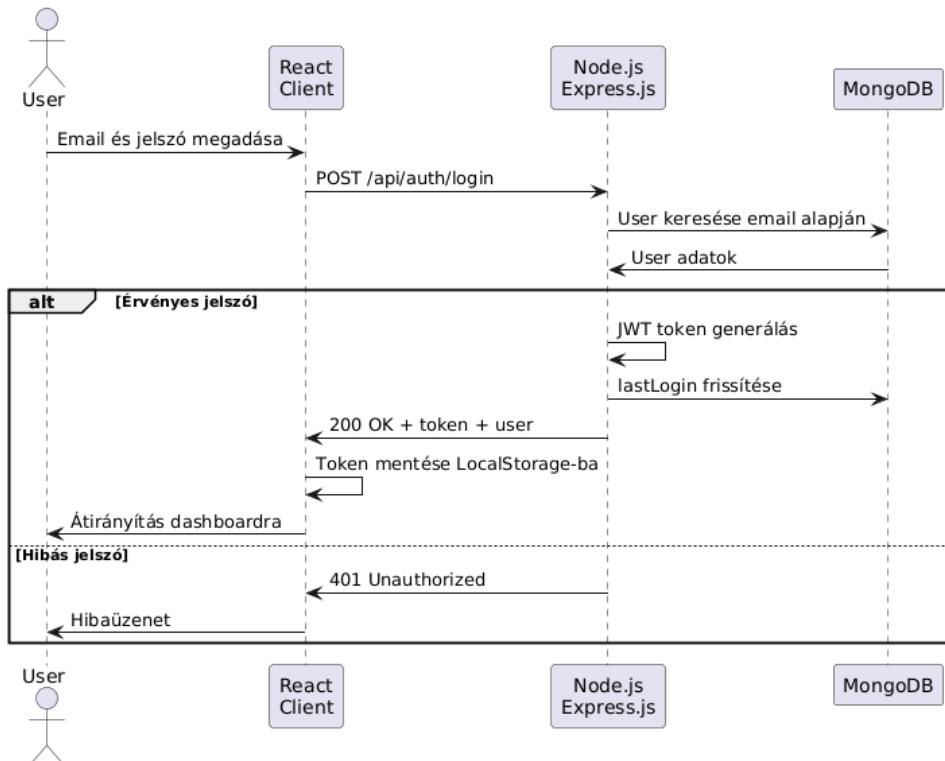


Figure 4.6.1: Login sequence diagram

#### 4.6.2 Portfolio sequence

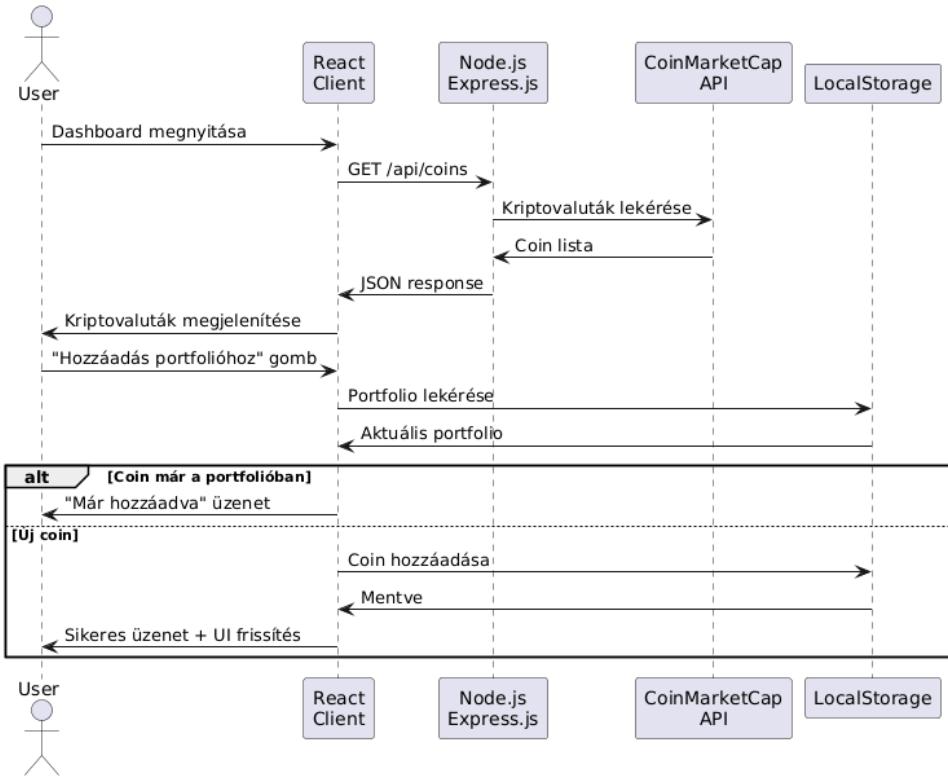


Figure 4.6.2: Portfolio sequence diagram

#### 4.6.3 Search sequence

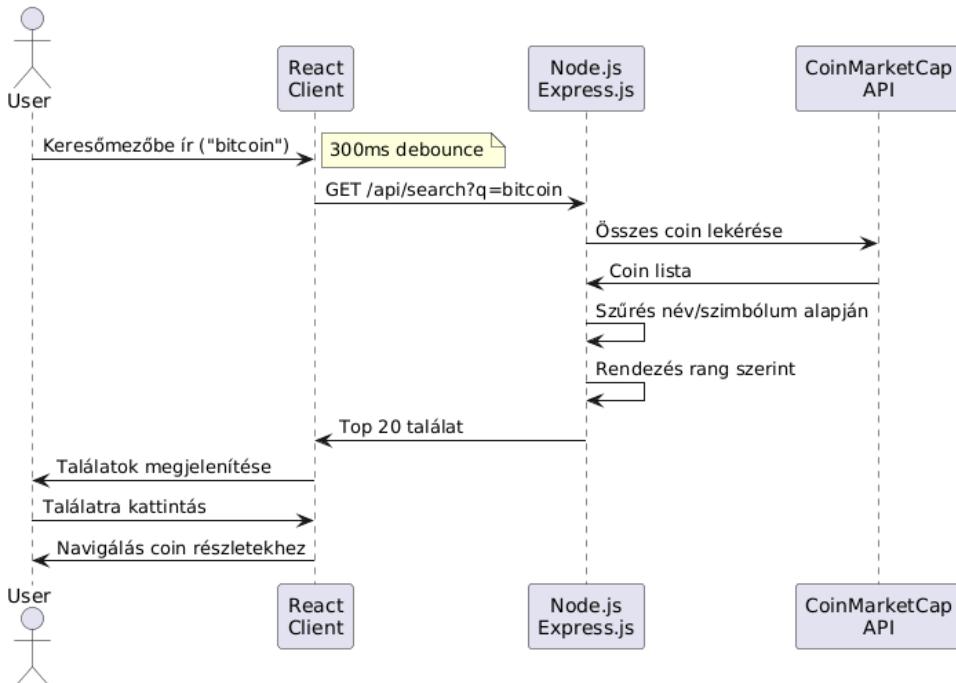


Figure 4.6.3: Search sequence diagram

## 4.7 Verziókövetés és projekt menedzselése

A **Cryptocurrency Tracker & Portfolio** fejlesztését a GitHub és egy Jira Board segítségével menedzselünk. A GitHub biztosította a projekt csapatmunkáját és verziókövetését, míg a Jira Board a feladatok és problémák vizualizálását és nyomon követését segítette. (Lásd 4.7.1)

### 4.7.1 Jira Tábla – Projekt Állapotkövetés

A projekt állapotának és a feladatok előrehaladásának átlátható követésére Jira Board-ot használtunk. Az oszlopok segítségével pontosan láthattuk, mely feladatokon dolgozunk, melyek állnak készen, és melyek várak megoldásra.

#### Kanban oszlopok:

- Backlog (Ötletek):** minden új ötletet, funkciót ide vettünk fel.
- To Do (Tennivalók):** A megvalósításra váró, prioritást kapott feladatok ide kerültek.
- In Progress (Folyamatban):** A jelenleg folyamatban lévő munkák, az egyes taskokhoz hozzárendelt felelőssel.
- Done (Kész):** A befejezett, tesztelt és a fő branch-be integrált feladatok.

#### Jira előnyei:

- Átláthatóság:** A feladatok állapota könnyen nyomon követhető, mindenki pontosan látta, ki dolgozik melyik taskon.
- Prioritáskezelés:** Könnyű volt a feladatok priorizálása (kritikus hibák, új funkciók).

### 4.7.2 GitHub Verziókezelés – Projektfejlesztés

A GitHub-ot a kód verziókövetésére és a csapatmunkára használtuk. Lehetővé tette a változtatások nyomon követését, a párhuzamos fejlesztést és a kód integritásának biztosítását.

Work	Assignee	Reporter	Priority	Status	Resolution	Created	Updated	Due date
GRV-22 Updated docker setup for backend and env s...	pakular.robert	pakular.robert	= Medium	Done	Done	Nov 19, 2025, 5:07 PM	Nov 19, 2025, 5:07 PM	None
GRV-21 Asset Details Page	pakular.robert	pakular.robert	= Medium	Done	Done	Nov 05, 2025, 4:57 PM	Nov 25, 2025, 5:25 PM	None
GRV-20 Saving user data in database	Pacurar Alex	Pacurar Alex	= Medium	Done	Done	Nov 05, 2025, 4:38 PM	Nov 11, 2025, 5:22 PM	None
GRV-19 Login/Register page implementation	pakular.robert	pakular.robert	= Medium	Done	Done	Oct 29, 2025, 4:51PM	Nov 05, 2025, 4:38 PM	None
GRV-17 Data dummy	Pacurar Alex	Pacurar Alex	= Medium	Done	Done	Oct 27, 2025, 6:02 PM	Nov 11, 2025, 5:21 PM	None
GRV-16 MERN Stack structure	Pacurar Alex	Pacurar Alex	= Medium	Done	Done	Oct 27, 2025, 6:00 PM	Oct 27, 2025, 6:01 PM	None
GRV-15 Base server setup	Pacurar Alex	Pacurar Alex	= Medium	Done	Done	Oct 21, 2025, 5:15 PM	Oct 21, 2025, 8:12 PM	None
GRV-14 Portfolio performance	Unassigned	Pacurar Alex	= Medium	Unresolved	Unresolved	Oct 21, 2025, 5:14 PM	Oct 21, 2025, 5:14 PM	None
GRV-13 Export portfolio as pdf		pakular.robert	= Medium	Unresolved	Unresolved	Oct 21, 2025, 5:14 PM	Nov 19, 2025, 4:48 PM	None
GRV-11 Multi-currency support	Pacurar Alex	Pacurar Alex	= Medium	Done	Done	Oct 21, 2025, 5:14 PM	Nov 16, 2025, 7:50 PM	None
GRV-10 Documentation	Pacurar Alex	Pacurar Alex	= Medium	Unresolved	Unresolved	Oct 21, 2025, 5:13 PM	Dec 02, 2025, 3:40 PM	None
GRV-8 Portfolio dashboard UI	pakular.robert	Pacurar Alex	= Medium	Done	Done	Oct 21, 2025, 5:07 PM	Nov 19, 2025, 4:46 PM	None
GRV-6 Implement search and filter	Pacurar Alex	Pacurar Alex	= Medium	Done	Done	Oct 21, 2025, 5:06 PM	Nov 20, 2025, 10:38 AM	None
GRV-5 Display live cryptocurrency data	Pacurar Alex	Pacurar Alex	= Medium	Done	Done	Oct 21, 2025, 5:06 PM	Nov 16, 2025, 7:50 PM	None
GRV-4 Navigation	Pacurar Alex	Pacurar Alex	= Medium	Done	Done	Oct 21, 2025, 5:05 PM	Nov 16, 2025, 7:50 PM	None
GRV-3 Base structure	Pacurar Alex	Pacurar Alex	= Medium	Done	Done	Oct 21, 2025, 5:04 PM	Oct 21, 2025, 8:12 PM	None
GRV-2 Database setup	Pacurar Alex	Pacurar Alex	= Medium	Done	Done	Oct 21, 2025, 5:03 PM	Nov 11, 2025, 5:29 PM	None
GRV-1 Docker setup	Pacurar Alex	Pacurar Alex	= Medium	Done	Done	Oct 21, 2025, 5:01 PM	Oct 27, 2025, 6:29 PM	None

Figure 4.7.1: Jira board issuek

## 5 Alkalmazás működése

### 5.1 UI megvalósítás

#### 5.1.1 Főoldal

Kiírja az elérhető coinokat, azoknak a részletes adatait és fent egy összesítő sor is van ahol az összes coin értéke látszik, a BTC dominálási szint,s hogy hány coin van betöltve

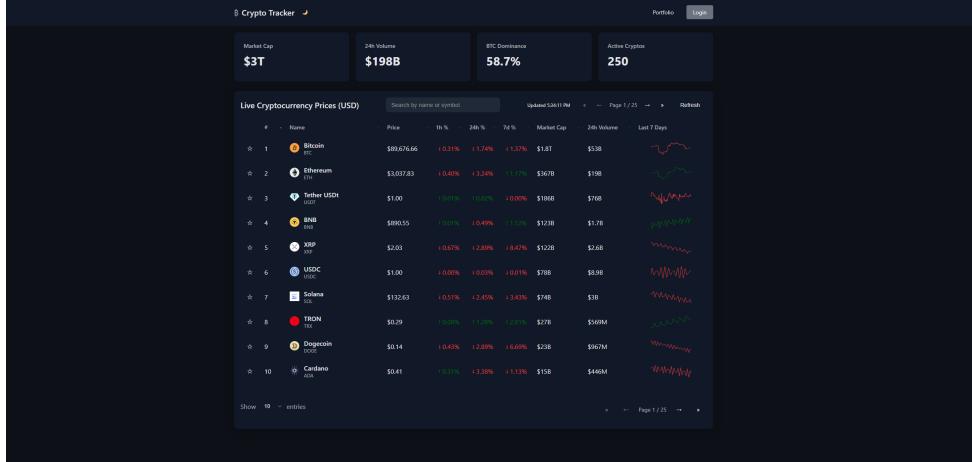


Figure 5.1.1: Főoldal

#### 5.1.2 Login / Register

Bejelentkező, illettve regisztrációs oldal. Az adatmezők helyes kitöltése után bejelentkezik a fiókba/regisztrál egy fiókot.

The figure shows a screenshot of the Crypto Tracker login page. The page has a dark background with white text. It features a "Login" heading, two input fields for "Email" and "Password", and two buttons at the bottom: "Login" and "Register". Below the "Login" button is a "Sign in with Google" button. The overall design is clean and modern.

Figure 5.1.2: Login Page

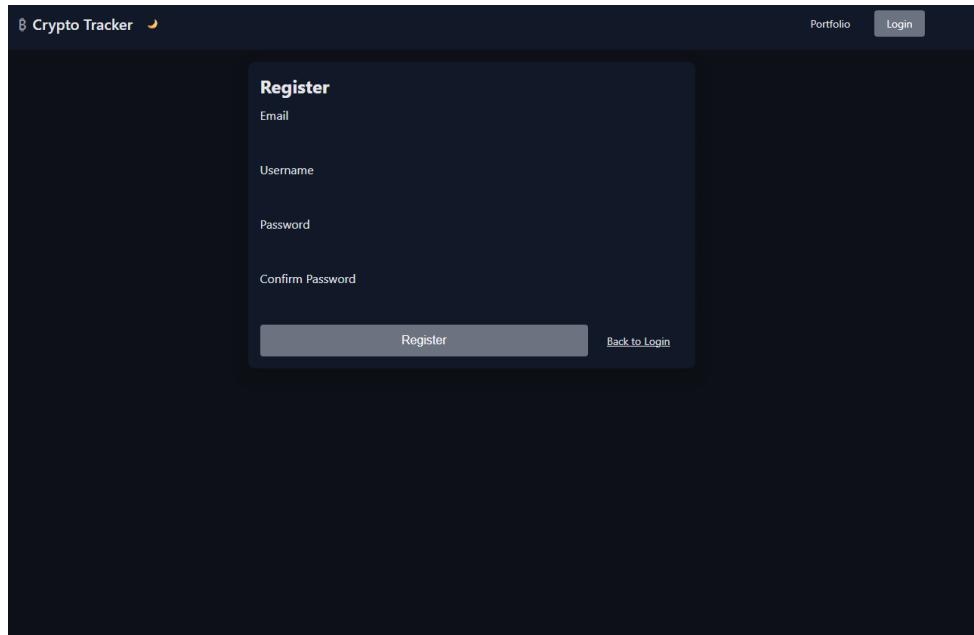


Figure 5.1.3: Register Page

### 5.1.3 Coin lista és részletek

Egy adott coinról részletes leírást ad, az árfolyam szerint egy grafikont jelenít meg. Ha még az adott coin nincs a felhasználó portfóliójában akkor van lehetőség innen is hozzáadni azt.

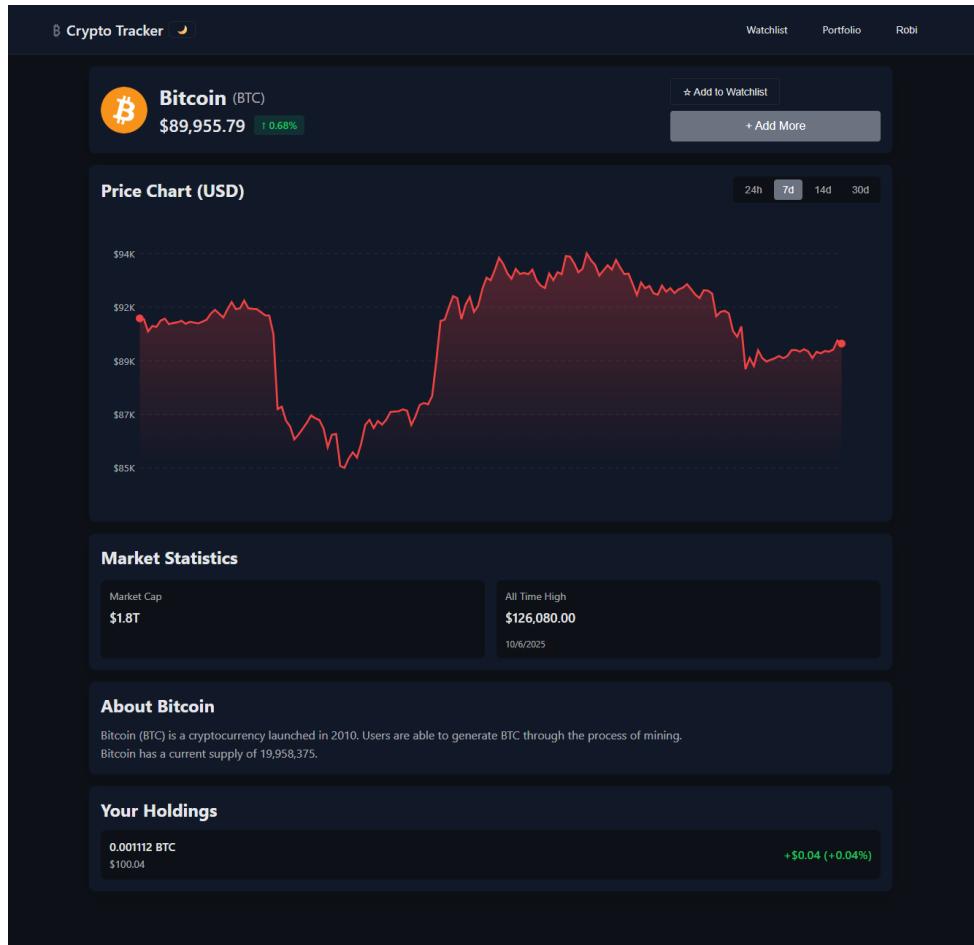


Figure 5.1.4: Coin Details Page

#### 5.1.4 Portfolio kezelés

Listázza a portfólióban szereplő coinokat, azoknak az adatait, s a megvásárlási árat. Az "+ Add" gomb lenyomásával pedig a felhasználó hozzá tudja adni a kívánt coint a portfóliójához.

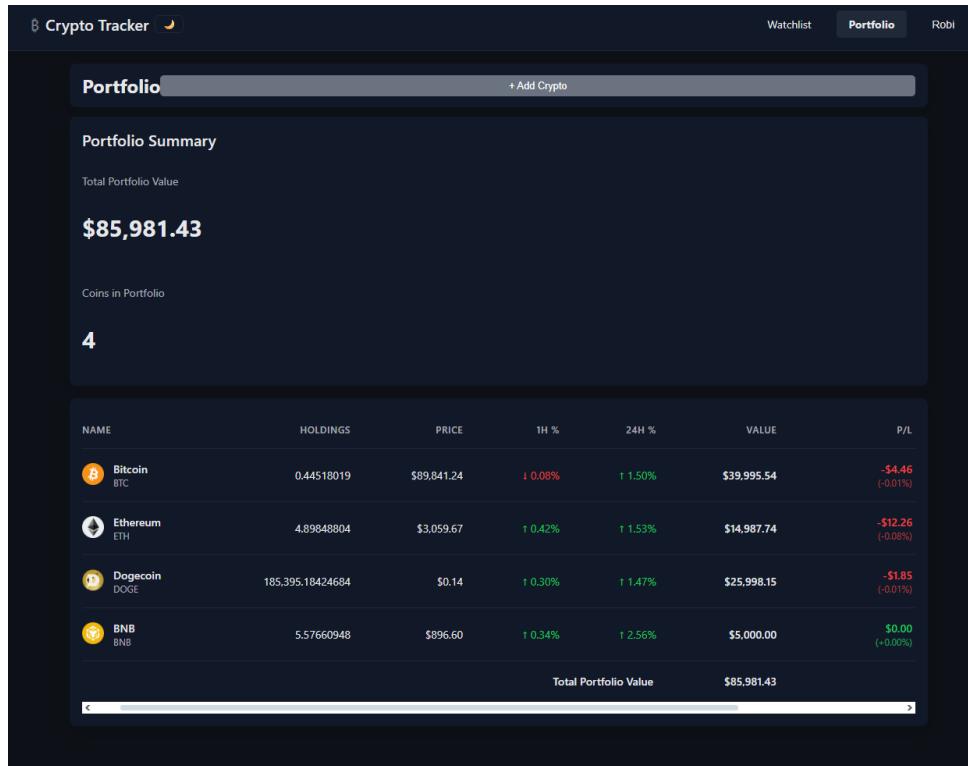


Figure 5.1.5: Portfolio Page

### 5.1.5 Watchlist

Listázza azokat a coinokat és azoknak adatait amiket a felhasználó "becsillagozí", tehát a watchlistjére rak. Csak bejelentkezett felhasználó képes használni ezt a funkciót.

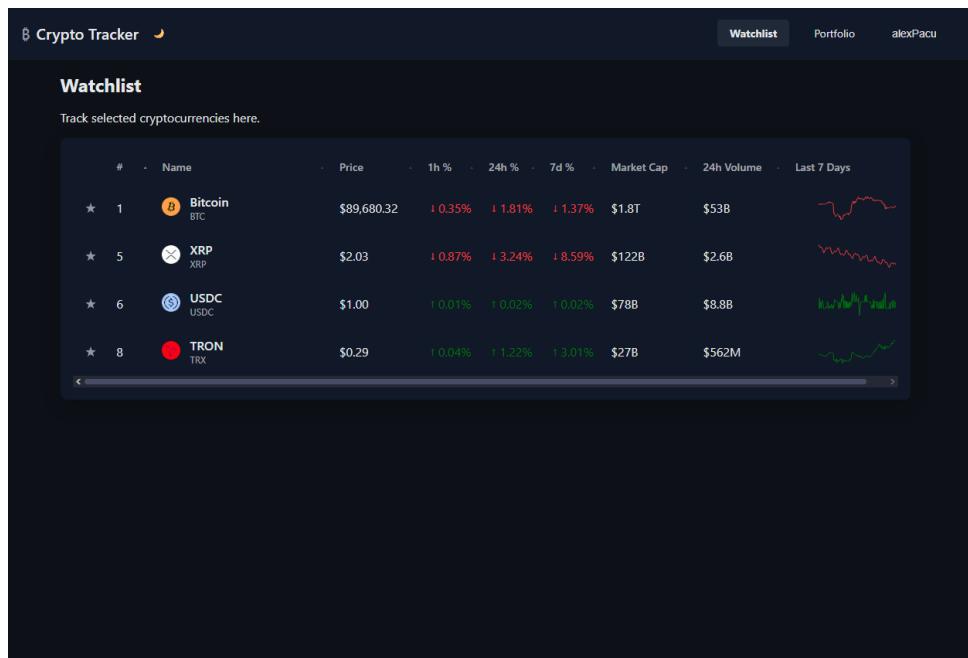


Figure 5.1.6: Watchlist Page

### 5.1.6 Keresés

Keresőmezőbe beiírt coin név vagy rövidítésre rákeres, s listázza a találatokat. Egy coint kiválasztva megnyitja a keresett coin részletes leírását s adatait.

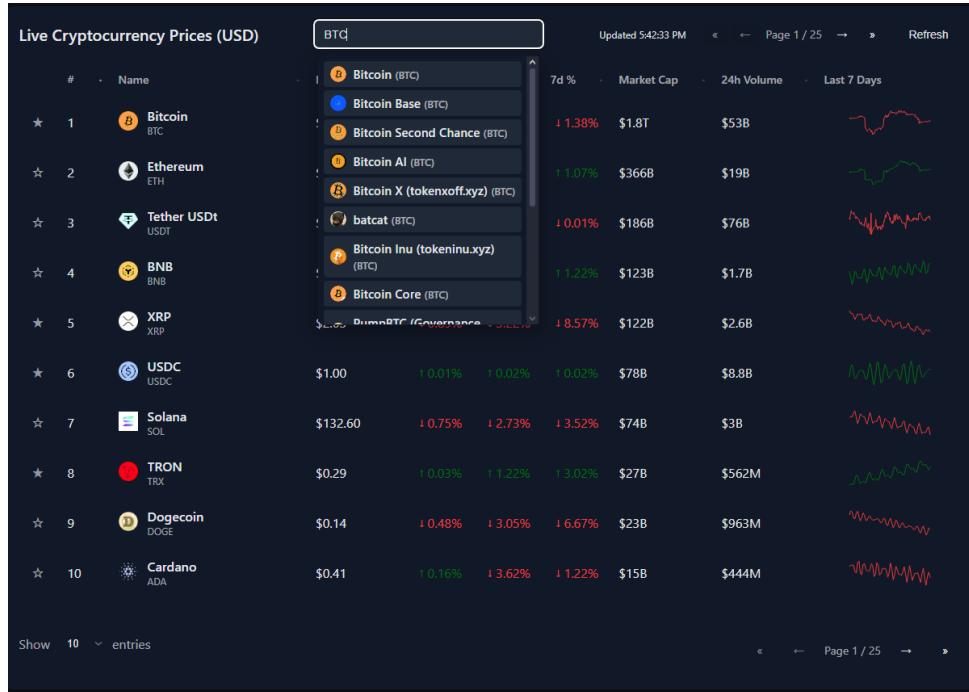


Figure 5.1.7: Search dropdown

### 5.1.7 Beállítások

Beállítási menü, belehet állítani a preferált valutát (USD, EUR, HUF, RON), illettve témát lehet állítani: világos/sötét.

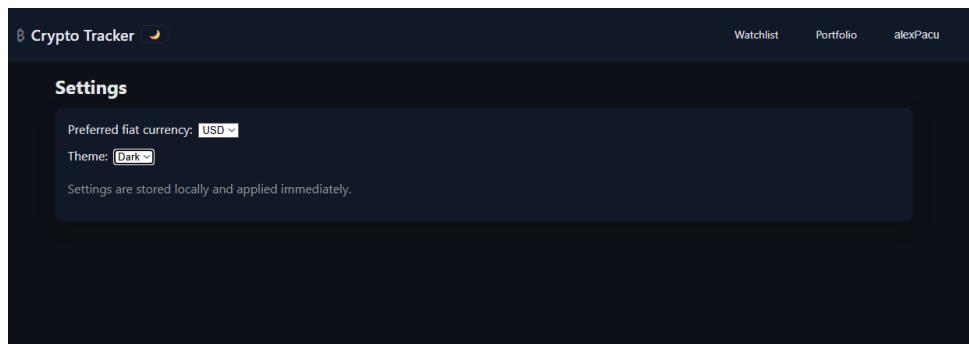


Figure 5.1.8: Settings Page

## 6 Összegzés

A **Cryptocurrency Tracker & Portfolio** alkalmazás azokat a kihívásokat célozza meg, amelyekkel a felhasználók nap mint nap szembesülnek: szétszórt információforrások, gyorsan változó árfolyamok, több pénznem kezelése, valamint a saját portfólió és figyelőlista (watchlist) áttekintése. A rendszer ezeket központosítja, és egységes, jól megtervezett

felületen jeleníti meg.

### Fő funkciók és előnyök:

- **Valós idejű és közel valós idejű piaci adatok:** Az alkalmazás megjeleníti az aktuális árakat, 24 órás változásokat, piaci kapitalizációt, rangot és forgalmat, több pénznemben (USD, EUR, HUF, RON), konzisztens formázással és kerekítéssel.
- **Watchlist és portfólió kezelés:** Bejelentkezés után a felhasználók személyre szabott figyelőlistát tarthatnak fenn és portfóliójukat menedzselhetik. A rendszer támogatja a mennyiségek, vételi árak és hozamok (profit/loss) követését, miközben fejlett vizuális visszajelzéseket ad.
- **Keresés és felfedezés:** Gyors, debounce-olt kereső segítségével egyszerűen megtalálhatók a kívánt coinok név vagy szimbólum alapján. Az eredmények intelligens rendezést és szűrést kapnak a jobb relevancia érdekében.
- **Stabil, reszponzív UI:** A dashboard minimalizálja a vizuális villogást az inkrementális frissítésekkel és skeleton/shimmer betöltőkkel, így az észlelt teljesítmény magas marad. A felület mobilon és asztali környezetben is konzisztens élményt nyújt.
- **Biztonság és adatszeparáció:** A jelszavak bcrypt-tel kerülnek hash-elésre, a hozzáférés JWT tokenekkel történik, a felhasználói adatok elkülönítve tárolódnak. A nem érzékeny preferenciák (pl. pénznem) kliens oldalon, LocalStorage-ban perzisztálnak.

### Jövőbeli fejlesztési lehetőségek:

- Valós idejű frissítések (WebSocket) a még gyorsabb adatmegjelenítéshez.
- Portfolio analitika (átlagár, allokáció, historikus teljesítmény).
- Értesítések (árfolyam alert, napi összefoglaló).
- Bővített pénznem- és nyelvi támogatás.
- Mobil alkalmazás (React Native) a teljes omnichannel élményért.

Összességében a **Cryptocurrency Tracker & Portfolio** egy modern, megbízható és hatékony eszköz a kriptovaluta piac követésére és a személyes befektetések kezelésére. Célja, hogy csökkentse az információs zajt, növelje az átláthatóságot és támogassa a felhasználókat abban, hogy magabiztosan, adatokra támaszkodva hozzanak döntéseket – egyetlen, könnyen használható dashboardon.

## 7 Ábrák jegyzéke

### List of Figures

3.1.1 Use Case diagram . . . . .	4
4.1.1 Komponens diagram . . . . .	6

4.5.1 User diagram . . . . .	9
4.6.1 Login sequence diagram . . . . .	9
4.6.2 Portfolio sequence diagram . . . . .	10
4.6.3 Search sequence diagram . . . . .	10
4.7.1 Jira board issuek . . . . .	11
5.1.1 Főoldal . . . . .	12
5.1.2 Login Page . . . . .	12
5.1.3 Register Page . . . . .	13
5.1.4 Coin Details Page . . . . .	14
5.1.5 Portfolio Page . . . . .	15
5.1.6 Watchlist Page . . . . .	15
5.1.7 Search dropdown . . . . .	16
5.1.8 Settings Page . . . . .	16