DELTA – Střední škola informatiky a ekonomie, s.r.o.
Ke Kamenci 151, PARDUBICE

MATURITNÍ PROJEKT
Systém pro sledování kryptoměn

Příjmení, jméno: Mervart Lukáš

Třída: 4.A

Studijní obor: Informační technologie 18-20-M/01

Školní rok: 2022/2023

Zadání maturitního projektu z informatických předmětů

Jméno a příjmení: *Lukáš Mervart*

Pro školní rok: *2022/2023*

Třída: *4. A*

Obor: *Informační technologie 18-20-M/01*

Téma práce: *Systém pro sledování kryptoměn*

Vedoucí práce: *Vlaďka Janů*

Způsob zpracování, cíle práce, pokyny k obsahu a rozsahu práce:

Cílem práce je vytvořit plně funkční webovou aplikaci, která bude sloužit k sledování kryptoměn. Uživatel se nejprve zaregistruje a poté může přidávat, upravovat nebo odebírat kryptoměny ze svého portfolia.

Aplikace bude obsahovat detaily měn např. pohlédnutí hodnot do minulosti, různé grafy a podrobné informace o kryptoměnách, také analyzovat jaký je uživatelův profit nebo jestli prodělal v rámci různých faktorů. Do výpočtu bude zahrnuta aktuální cena kryptoměny i kurz koruny vůči fiat měně.

Aplikace bude napsána v jazyce Typescript/Javascript s frameworkem React a technologií Node.js

Stručný časový harmonogram (s daty a konkretizovanými úkoly):

- Konec září – Návrh databáze a designu aplikace
- Říjen – Vývoj aplikace a její propojení s databází
- Konec listopadu – Prototyp funkční aplikace, vytvořen výpočet profitu
- Prosinec – Dokončování hlavních částí aplikace
- Leden – Začátek psaní dokumentace, dolaďování, oprava případných chyb
- Únor – Nasazení aplikace na web, dočištění aplikace, dodělávání dokumentace

Prohlášení

Prohlašuji, že jsem svou práci vypracoval samostatně a použil jsem pouze prameny a literaturu uvedené v seznamu bibliografických záznamů.

V Pardubicích dne _____ _____

Lukáš Mervart

Poděkování

Děkuji vedoucí mého projektu Bc. Vlaďce Janů za vedení mého projektu, rady, nápady a hlavně trpělivost. Také bych chtěl poděkovat kamarádům a rodině za podporu.

Anotace

Tento projekt má za cíl vytvořit plně funkční a jednoduchý systém pro sledování kryptoměn. Uživatelé budou mít možnost přidávat, odebírat a upravovat si kryptoměny a sledovat profit/lose. Webová aplikace cílí na lidi, kteří hledají jednoduchý způsob na sledování vlastněných kryptoměn.

Klíčová slova

React, webová aplikace, Next.JS, Typescript, Firebase, Kryptoměny

Annotation

This project aims to create a fully functional and simple cryptocurrency tracking system. Users will be able to add, remove and edit cryptocurrencies and track profit/lose. The web application targets people who are looking for a simple way to track the cryptocurrencies they own.

Keywords

React, web application, Next.JS, Typescript, Firebase, Cryptocurrencies

Obsah

1. Úvod 1
2. Technologie 1
2.1 React 1
2.2 Next.js 2
2.3 Node.js 2
2.4 TypeScript 2
2.5 HTML 3
2.6 CSS 3
2.7 Firebase 4
2.7.1 Firestore 4
2.7.2 Firebase Authentication 4
2.8 Material UI 5
2.9 Axios 5
2.10 CoinGecko API 5
2.11 Lightweight Charts 6
3. Jiná řešení 6
3.1 CoinStats.app 6
3.2 Delta.app 6
4. Architektura 7
4.1 Node server 7
4.1.1 Next.js aplikace 7
4.1.1.1 Komponenty 8
4.2 Firebase backend 9
4.2.1 Ukládání dat 9
4.2.2 Správa uživatelů 10
5. Webová aplikace 10
5.1 Registrace 11
5.2 Přihlášení 12
5.3 Hlavní stránka 13
5.3.1 Formulář na přidávání kryptoměn 13
5.3.2 Odhlášení uživatele 14

5.3.3 Seznam přidaných kryptoměn 14
5.3.3.1 Celkový profit/lose 15
5.3.4 Detail kryptoměny 16
5.3.4.1 Profit/lose 18
5.4 Responzivita webu 19
6. Závěr 19
Reference 20
Reference obrázků 21
Seznam obrázků 22

1. Úvod

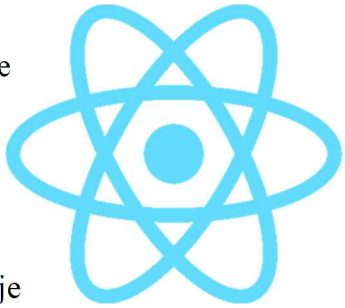
Existuje spoustu portfolií ohledně kryptoměn, jelikož poslední roky jsou velice populární. Jsou kryptoměnové burzy, portfolia, oboje naráz, v podstatě všechno, co nás napadne o kryptoměnách už někdo vytvořil. Také si hodně lidí dělá vlastní „portfolio“ například v Excelu, avšak to je za mě hodně zbytečné práce. U hodně systémů pro sledování kryptoměn je pouze možnost si to propojit s nějakou naší kryptoměnovou peněženkou (Binance, Metamask, Coinbase). Pro někoho to může být výhoda, ale pro někoho ne. Pokud si například někdo chce vyzkoušet kolik by vydělal/prodělal, pokud by nakoupil nějakou kryptoměnu v nějaký čas apod., nebo nechtěl propojovat svoji krypto peněženku.

Právě kvůli tomuto jsem se rozhodl vytvořit tuto aplikaci. Cíleno je na uživatele, kteří si chtějí jednoduše přidat kryptoměny, aniž by si museli s touto aplikací propojovat svojí krypto peněženku nebo je vůbec reálně vlastnit.

2. Technologie

V této kategorii se podíváme na hlavní a nejdůležitější technologie této webové aplikace – framework, jazyky, databáze.

2.1 React



Obrázek 1 React logo [18]

React je open-source JavaScript knihovna pro tvorbu uživatelských rozhraní, kterou vytvořila společnost Facebook, dnes Meta. Jedná se o jednu z nejpopulárnějších knihoven pro tvorbu webových aplikací v průmyslu. Základem knihovny React jsou takzvané komponenty, které mohou být znovupoužity a jednoduše kombinovány do větších aplikací. Virtuální DOM, který React využívá, minimalizuje náklady na aktualizaci rozhraní a optimalizuje výkon aplikace.

Poskytuje mnoho dalších funkcí a nástrojů pro práci s komponenty, včetně lifecycle metod, které umožňují reagovat na různé události v průběhu životního cyklu komponenty.

React umožňuje psát kód v JSX, což je rozšíření syntaxe JavaScriptu, které umožňuje vkládat HTML podobné kódy přímo do kódu JavaScriptu. React podporuje komunikaci mezi komponentami pomocí props a callback funkcí. Stavové komponenty umožňují udržovat a měnit stav aplikace na základě interakce uživatele nebo jiných podnětů. [1] [2]

2.2 Next.js

Next.js je moderní open-source framework pro vývoj webových aplikací a statických webů založený na Reactu. Byl vyvinut společností Vercel a nabízí řadu funkcí, které usnadňují vývoj a optimalizaci React aplikací. Mezi hlavní vlastnosti Next.js patří server-side rendering, statické generování stránek a automatická optimalizace kódu.

Díky těmto funkcím Next.js umožňuje vytvářet rychlejší a výkonnější aplikace. Navíc Next.js podporuje funkci Incremental Static Regeneration, což umožňuje efektivně aktualizovat statický obsah bez nutnosti celého rebuildu.

Díky svým výhodám a flexibilitě získává Next.js stále více popularity a je vhodný pro různé typy projektů, od malých webů až po rozsáhlé aplikace. [3]

Obrázek 2 Next.js logo [19]

2.3 Node.js

Node.js se používá k vytvoření robustního a škálovatelného back-endu. Node.js je také prostředí umožňující spouštět JavaScript kód mimo webový prohlížeč. Také umožňuje jednoduché integrace s externími službami, například s API pro získání aktuálních cen kryptoměn nebo s databází pro ukládání uživatelských dat.

Díky své asynchronní povaze poskytuje Node.js vysokou propustnost a výkon, což je ideální pro zpracování velkého množství požadavků. To zvyšuje rychlost a efektivitu vaší webové aplikace, což vede k lepšímu uživatelskému zážitku. [4]

Obrázek 3 Node.JS logo [20]

2.4 TypeScript

TypeScript je open-source programovací jazyk vyvinutý společností Microsoft, který je nadstavbou JavaScriptu a přidává do něj statické typování. Cílem TypeScriptu je zlepšit udržitelnost, čitelnost, přehlednost a případné škálování kódu, odhalování chyb při kompilaci, díky čemuž snižuje pravděpodobnost chyb při běhu aplikace.

TypeScript je kompatibilní se stávajícím JavaScriptem, což usnadňuje jeho začlenění do existujících projektů.

Obrázek 4 TypeScript logo [21]

Díky svému přísnému typovacímu systému značně usnadňuje vývoj a ladění aplikací. TypeScript se stává stále populárnějším a populárnějším jazykem, zejména pro vývoj velkých a komplexních aplikací. [5]

2.5 HTML

HTML je základním stavebním kamenem webových stránek a aplikací. Tento jazyk byl vytvořen v roce 1990 a od té doby prošel několika verzemi, aktuální je HTML5. Slouží k popisu struktury webových stránek a je zodpovědný za jejich zobrazení v internetových prohlížečích. Používají se zde různé tagy, avšak v této aplikaci je většina z nich nahrazena rovnou nastylovanými MUI tagy. [6]

Obrázek 5 HTML logo [22]

2.6 CSS

CSS je jazyk používaný pro vizuální formátování a stylování webových stránek. Byl vytvořen v roce 1996 a od té doby prošel několika verzemi, aktuální verze je CSS3. CSS umožňuje vývojářům oddělit obsah (psaný v HTML) od jeho vizuální prezentace, což zjednodušuje údržbu a zlepšuje přístupnost webových stránek.

CSS používá selektory pro cílení konkrétních HTML prvků a aplikuje na ně styly, které se definují pomocí vlastností a hodnot. Například barva textu, velikost písma, pozadí, ohraničení či rozložení prvků na stránce.

Obrázek 6 CSS logo [23]

V této aplikaci je většina CSS nahrazena knihovnou MUI, avšak zde je CSS stejně použito na dodatečnou úpravu už předem nastylovaných komponent. [7]

2.7 Firebase

Firebase je komplexní platforma pro vývoj webových a mobilních aplikací, vyvinutá společností Google. Poskytuje celou řadu nástrojů a služeb, které usnadňují vývoj, nasazení a škálování aplikací. Mezi hlavní služby Firebase se řadí Firestore a Firebase Authentication, které jsou široce využívány pro vytváření moderních aplikací s bohatými funkcemi a jednoduchou správou a také obsaženy v tomto projektu.

Firebase, Firestore a Firebase Authentication společně poskytují ideální a bezpečný základ pro moderní aplikace. Díky těmto nástrojům je možné rychle vytvářet a nasazovat aplikace bez nutnosti vytvářet a spravovat vlastní infrastrukturu nebo se starat o základní bezpečnost. [8] [10]

Obrázek 7 Firebase logo [24]

2.7.1 Firestore

Firestore je cloudová NoSQL databáze. Umožňuje snadné ukládání, načítání a synchronizaci dat mezi klienty a serverem v reálném čase. Firestore je navržen tak, aby byl snadno škálovatelný a nabízel vysokou dostupnost a konzistenci dat. Jeho flexibilní datový model umožňuje ukládat data ve formě dokumentů, které jsou organizovány do kolekcí. Díky tomu je Firestore vhodný pro široké spektrum aplikací, od jednoduchých webových stránek až po rozsáhlé webové aplikace. [9] [10]

2.7.2 Firebase Authentication

Firebase Authentication je služba, která usnadňuje správu a autentizaci uživatelů v aplikacích. Podporuje mnoho metod přihlašování, včetně sociálních sítí (Google, Facebook, Twitter), e-mailu a hesla, telefonního čísla nebo anonymního přihlášení. Firebase Authentication je navržen tak, aby byl bezpečný, škálovatelný a snadno integrovatelný do aplikací. [8] [10]

2.8 Material UI

Material-UI, zkráceně MUI, vyvinutá společností Google je populární React knihovna komponent, která implementuje designový jazyk. MUI usnadňuje vývoj moderních a responzivních webových aplikací tím, že poskytuje předdefinované komponenty a styly pro různé UI prvky, jako jsou tlačítka, formuláře, navigace nebo dialogy.

Obrázek 8 MUI logo [25]

Použitím Material-UI v projektu získáte vysokou úroveň uživatelského zážitku a profesionální vzhled, což přispěje k úspěchu vaší webové aplikace.

Díky MUI můžete rychle a snadno navrhnout a implementovat uživatelské rozhraní, které je responzivní, přizpůsobivé a vizuálně pěkné. MUI také podporuje přizpůsobení barev, typografie a dalších vizuálních prvků, což umožňuje jednoduše upravit vzhled aplikace dle vašich preferencí. [11]

2.9 Axios

Axios je jednoduchá knihovna pro webové aplikace a node.js. Slouží k provádění HTTP požadavků, umožňuje jednoduše získávat data z API a provádět CRUD [1]operace. [12]

2.10 CoinGecko API

Jako API se používá free verze CoinGecko API, která mi přišla jako nejideálnější volba pro tento projekt. Tato API má spoustu možností na dotazování, jak pro aktuální, tak pro historická data. Data poskytuje ve formátu JSON a pro dotazování ve free verzi není potřeba API klíč. Jediná nevýhoda je poměrně málo requestů za minutu, pro jednoho uživatele to stačí, ale pokud by aplikaci využívalo více uživatelů naráz, mohl by s tím nastat problém. [13]

[1] CREATE, REMOVE, UPDATE DELETE

2.11 Lightweight Charts

Lightweight Charts je vysoce výkonná, lehká a flexibilní client-side knihovna pro vytváření interaktivních finančních grafů. Tato knihovna je vyvíjena společností TradingView a je speciálně vytvořena pro finanční grafy a navržena tak, aby minimalizovala spotřebu prostředků a maximalizovala rychlost zobrazení. [14]

3. Jiná řešení

Existuje spoustu aplikací na trackování kryptoměn, s různými funkcemi a postaveny na různých technologiích. Zde si je trošku projdeme, abychom měli přehled o jiných řešeních.

3.1 CoinStats.app

Jako první se podíváme na aplikaci CoinStats, která mi sloužila jako dobrá inspirace pro moji vlastní interpretaci aplikace pro sledování kryptoměn.

CoinStats je jedna z nejpopulárnějších aplikací pro sledování kryptoměn, která umožňuje uživatelům sledovat tržní ceny, portfolia a novinky z krypto světa.

Technologie bohužel nejsou zveřejněny, ale předpokládá se, že využívá React s Node.js a JavaScript. [15]

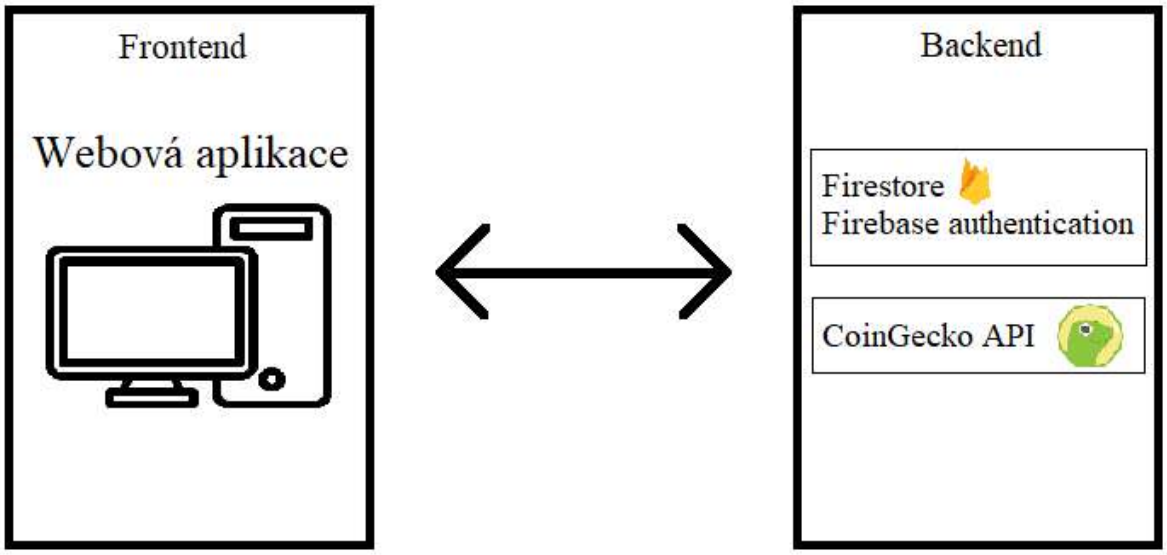
3.2 Delta.app

Delta.app je aplikace pro správu kryptoměnového portfolia, ale i například akcií, která poskytuje uživatelům přehled o jejich investicích do kryptoměn a sledování tržních cen. Aplikace se zaměřuje na vizuální vzhled, přizpůsobivost a bezpečnost a využívá moderní technologie a principy k poskytnutí prvotřídního zážitku z používání.

Technologie bohužel nejsou zveřejněny, ale předpokládá se, že využívá Ruby a JavaScript. [16]

4. Architektura

Architekturu aplikace tvoří tři základní prvky. Next.js – webová aplikace, Node server, CoinGecko API a Firebase – backend.



Obrázek 9 Architektura

4.1 Node server

Node.js server nám umožňuje spouštět JavaScript kód na straně serveru. To znamená, že namísto toho, aby byl JavaScript spouštěn pouze v prohlížeči na straně klienta, může být spouštěn také na serveru. To umožňuje vytvářet webové aplikace s jedním jazykem pro obě části (klient i server).

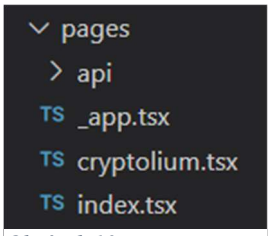
Běží zde naše next.js aplikace a jsou zde také nainstalované rozšiřující balíčky pro naší aplikaci. [4]

4.1.1 Next.js aplikace

Next.js je nadstavbou Reactu a v našem projektu poskytuje dodatečné prvky pro usnadnění práce s Reactem. Například přesně danou strukturu souborů. Dva hlavní soubory jsou „/pages“ a „/components“.

Ve složce „/pages“ se nachází všechny stránky, na které se můžeme v aplikaci dostat, výjimkou jsou akorát stránky, které mají na začátku názvu podtržítko. Tato složka je vidět na obrázku č. (10). [3]

Jako první je tsx (kombinace TypeScriptu a JavaScript XML) soubor „cryptolium“, ve kterém se nachází hlavní část aplikace – přidávání kryptoměn, seznam koupených kryptoměn a jejich profit/lose.



Obrázek 10 pages

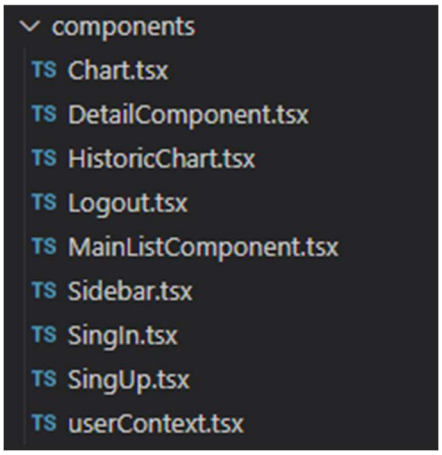
Další je „index“, což je vlastně první stránka, která se načte po zadáni url adresy. Nachází se zde formulář pro registraci uživatele.

„Singin“, stránka na přihlašování, na kterou se dá dostat přes stránku index.

4.1.1.1 Komponenty

Ve složce „/components“ se nachází takzvané komponenty, což jsou vlastně základní stavební kameny všech React aplikací. Tyto komponenty nám umožňují rozdělit kód do více souborů a také se komponenty se dají opakovaně využívat. Díky tomu je kód v „/pages“ kratší, čistější a přehlednější. Tato složka je vidět na obrázku č. (11).

Komponenta „userContext“, která se nikde nevykresluje, ale slouží spíše jako API, pro manipulaci s uživatelem a jeho daty, přihlášení, registrace, odhlašování.



Obrázek 11 komponenty

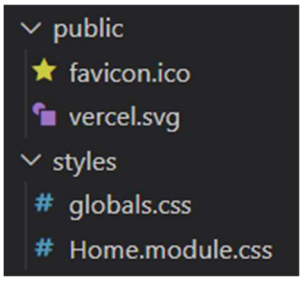
Komponenta „SingUp“ se nachází na stránce „index“ a slouží k registraci uživatele.

Komponenta „SingIn“ se nachází na stránce „Singin“, slouží k přihlašování uživatele.

Komponenta „Sidebar“ se nachází se na stránce „cryptolium“. Je v ní logika přidávání kryptoměn do seznamu a do databáze. Dále se v ní nachází i „Logout“, který zajišťuje odhlašování uživatele.

Komponenta „MainListComponent“ se nachází na stránce „cryptolium“. Je to hlavní část aplikace, je v ní seznam kryptoměn a dá se přes ní dostat na komponentu „DetailComponent“, kde jsou všechny detaily o určité kryptoměny a komponenty „Chart“ a „HistoricChart“, což jsou grafy o cenovém vývoji kryptoměn.

Dále jsou zde například složky „/styles“ pro stylování a „/public“ například pro obrázky, ikony a veřejně dostupné věci. Ty vidíme na obrázku č. (12).



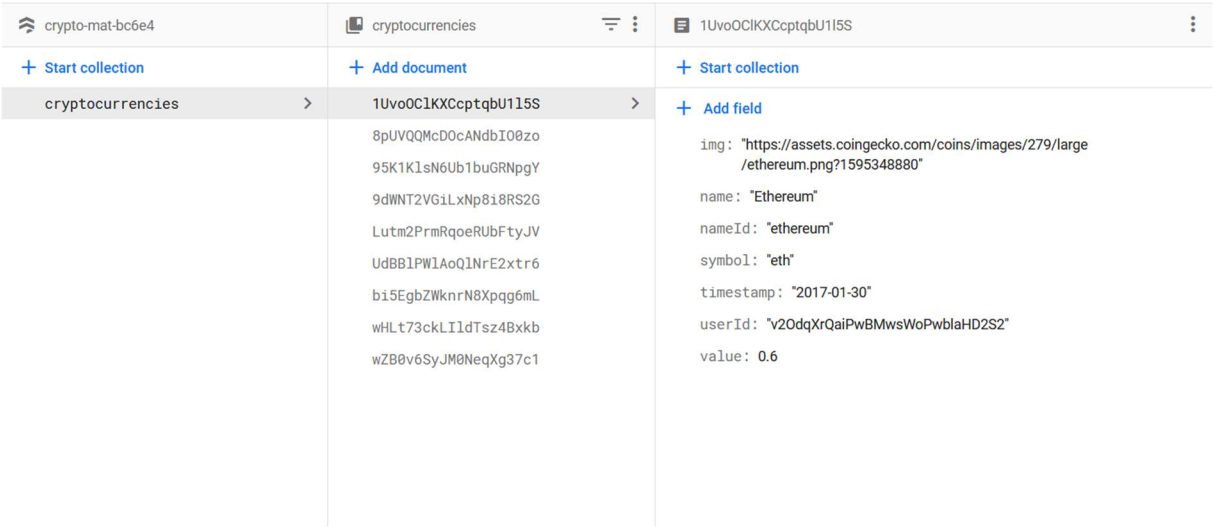
Obrázek 12 public & styles

4.2 Firebase backend

Jak zde už bylo zmíněno, pro backend v této aplikaci využíváme Firebase. Ta komunikuje s klientem díky svému vlastnímu SDK, který umožňuje snadno integrovat Firebase služby do aplikace. [10]

4.2.1 Ukládání dat

Jako databázi používáme Firestore od Firebase, což je NoSQL databáze. Poskytuje synchronizaci v reálném čase a offline podporu, což znamená, pokud se odpojíme od internetu, data se uloží na lokální úložiště a jakmile budeme zase online, data se synchronizují. [9]



Obrázek 13 Firestore struktura

Zde na obrázku č.(13) vidíme strukturu naší databáze. Základem jsou kolekce, které se potom dělí na dokumenty a v ní různá pole s hodnotami, ve kterých se dá zvolit typ, například string, number, boolean, array. Zde používáme datové typy pouze string a number, jelikož více není potřeba. Určitě by pro timestamp, šel použít poskytovaný typ timestamp, ale při jeho použití musí být udán čas i v sekundách, což pro toto řešení není ideální.

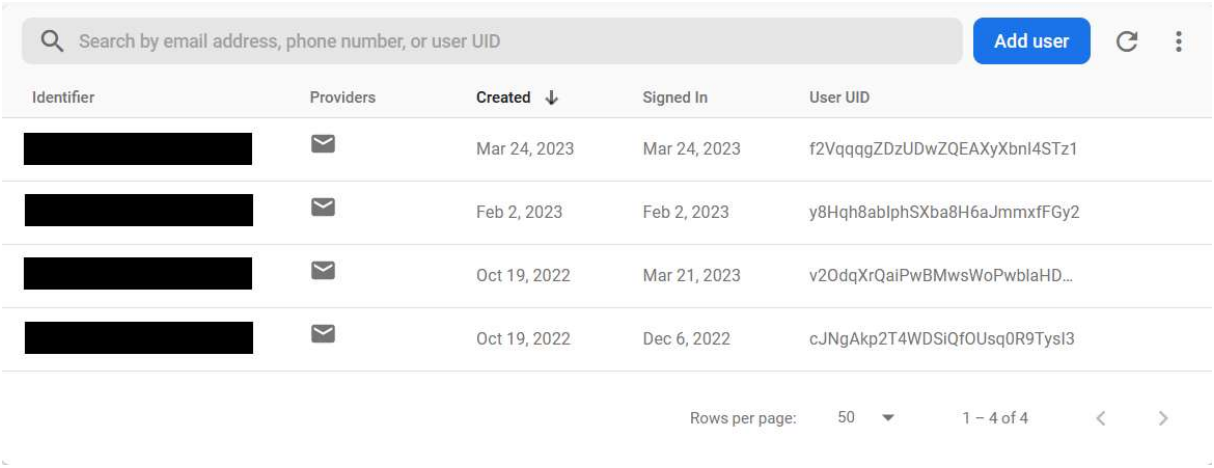
V naší databázi více než jednu kolekci nepotřebujeme, proto tu máme pouze kolekci „cryptocurrencies“. Dále jsou zde dokumenty, které vlastně reprezentují už různé uživatelsky přidané kryptoměny a jako jejich názvy jsou jejich UIDs, které automaticky vygeneruje Firebase při vytvoření dokumentu.

V každém dokumentu už jsou různé pole s informacemi o dané přidané kryptoměně. Jako první je obrázek (img) kryptoměny, dále 3 položky pro jméno (name), id kryptoměny (nameId) a symbol (symbol), poté časový údaj (timestamp) kdy byla kryptoměna zakoupena, userId abychom věděli, k jakému uživateli tato kryptoměna patří a kolik kryptoměny jsme koupili (value).

4.2.2 Správa uživatelů

Pro autentizaci uživatelů zde používáme Firebase Authentication, což je vcelku jednoduchý, ale kvalitní tool, který řeší přihlašování a registrování uživatelů. V této aplikaci využíváme login/registraci jen přes e-mail a heslo, ale dá se i použít například Google nebo Facebook účet. Tyto údaje se potom uloží do složky Users, kterou můžeme vidět na obrázku č. (14). Zde se nachází identifier, což je v našem případě email, kdy byl účet vytvořen, poslední přihlášeni a uživatelovo unikátní id.

Firebase authentication poskytuje při registraci pouze e-mail a heslo. Určitě by byla možnost přidat i například username, ale podle mého názoru to v takovémto typu aplikace není potřeba.

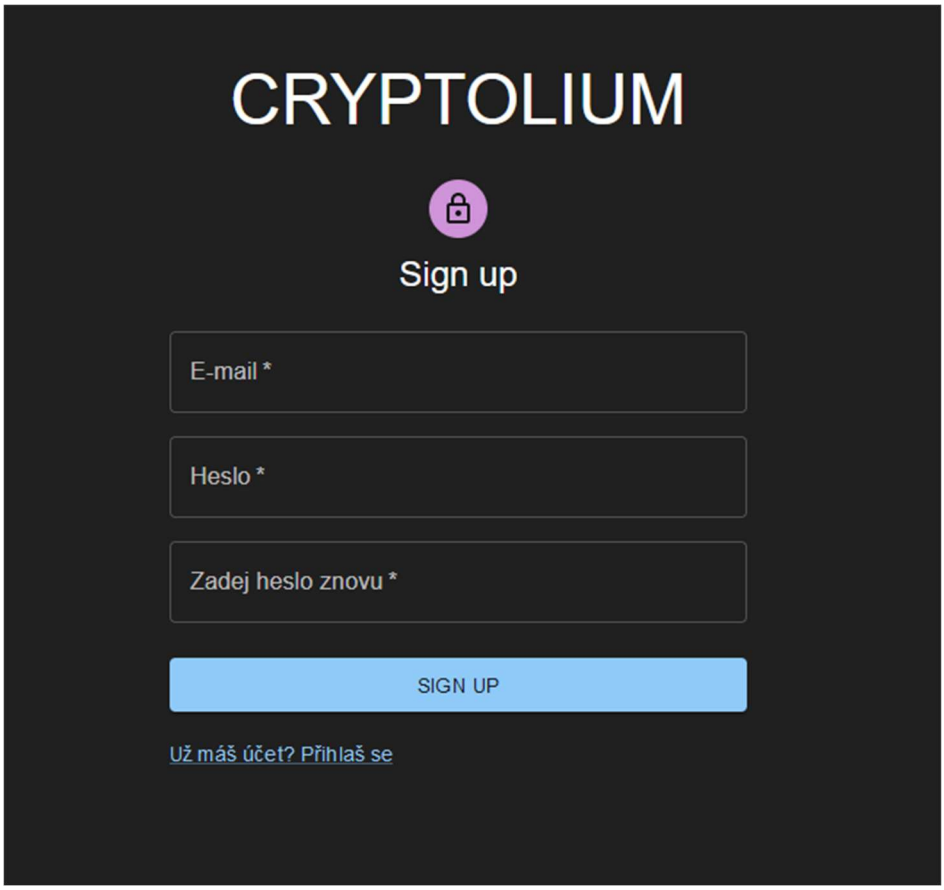


Obrázek 14 Firebase authentication

5. Webová aplikace

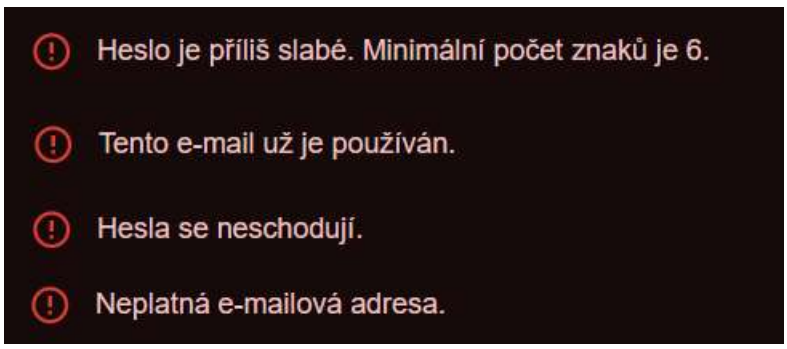
V této části si popíšeme veškeré funkce systému přístupné uživatelům.

5.1 Registrace



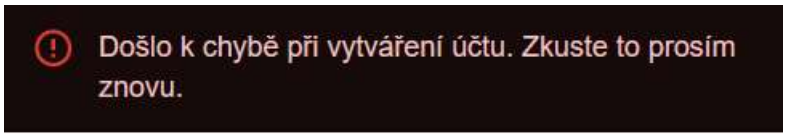
Obrázek 15 Stránka registrace

Registrace funguje přes již zmíněnou Firebase Authentication. UI vidíme na obrázku č. (15). Pro úspěšné fungování aplikace, musí být uživatel zaregistrován a přihlášen. Při registraci se vyplňuje e-mail, heslo a pro ověření heslo ještě jednou. Pro validní registraci, musí být správná forma e-mailu a heslo musí mít minimálně 6 znaků. Kód následně ověří, jestli už e-mail není zaregistrován, popřípadě napíše uživateli, že je e-mail už použit. U hesla ověří požadovanou délku 6 znaků, jestli se hesla shodují a pokud je nějaký problém, na uživatele vyskočí text s problémem, který můžeme vidět na obrázku č. (16).



Obrázek 16 Specifické error zprávy

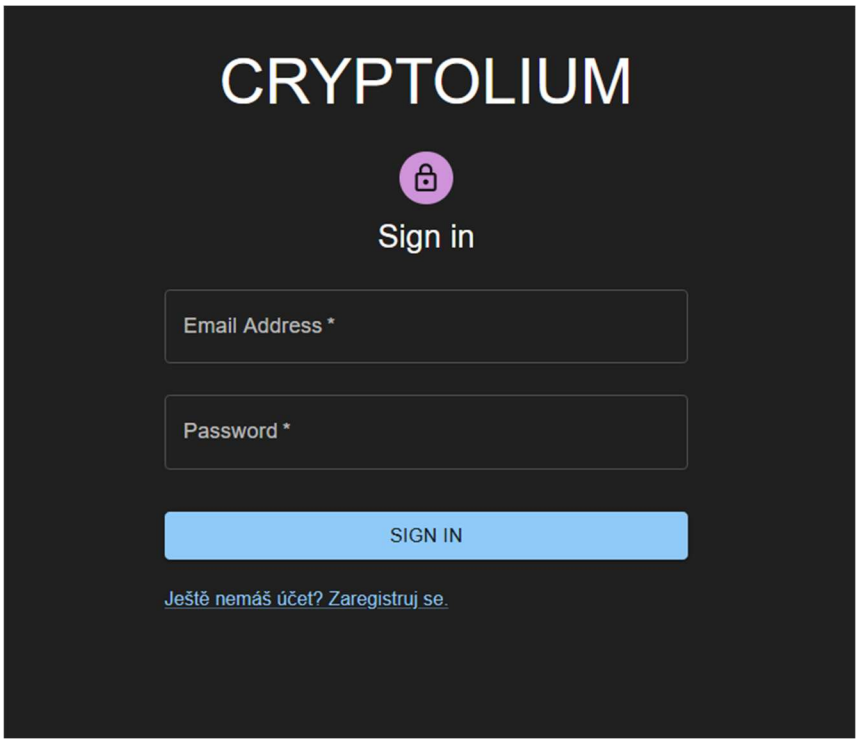
Pokud se objeví nějaký error, který není definován, ukáže zpráva jako na obrázku č. (17).



Obrázek 17 Výchozí error zpráva

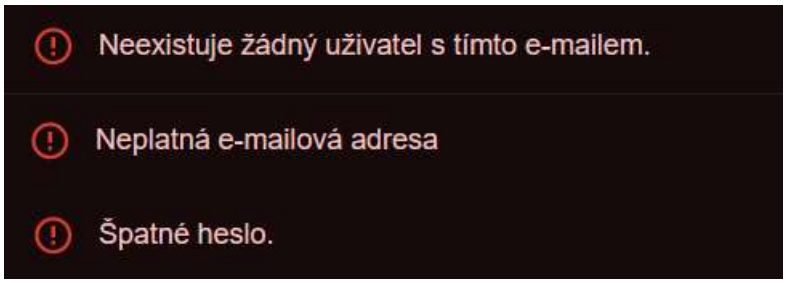
Po úspěšné registraci je uživatel přesměrován na hlavní stránku.

5.2 Přihlášení



Obrázek 18 Stránka přihlášení

Přihlašování funguje stejně jako registrace přes Firebase Authentication. UI vidíme na obrázku č. (18). Uživatel musí být vždy přihlášen, aby mu aplikace fungovala. Při přihlášení se zadává předem zaregistrované heslo a e-mail, které kód ověří, jestli má e-mail správnou formu, je už v databázi a jestli k němu sedí správné heslo, popřípadě vypíše, co je špatně. Chybové hlášky vidíme na obrázku č. (19).



Obrázek 19 Specifické error zprávy

Pokud se stane nějaký error, který není definován ukáže zpráva jako na obrázku č. (20).

Obrázek 20 Výchozí error zpráva

Po úspěšném přihlášení je uživatel přesměrován na hlavní stránku.

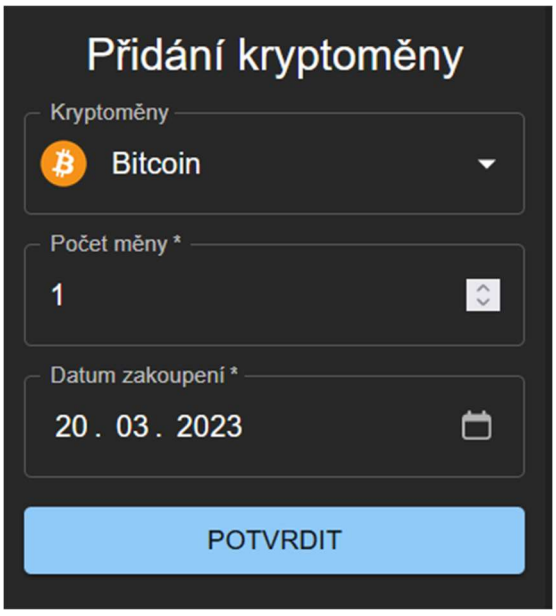
5.3 Hlavní stránka

Na úvodní straně najde uživatel všechny důležité věci. Přidání kryptoměn a seznam již vložených kryptoměn a po rozkliknutí podrobné informace k nim.

Barevné schéma bylo zvoleno tmavé (do šedočerné) hlavní barvy jsou použity #202020, #282828, #222. Schéma bylo zvoleno takové, jelikož obecně je mezi mladšími lidmi oblíbenější, včetně mě.

5.3.1 Formulář na přidávání kryptoměn

Nachází se na levé straně stránky, v případě malého rozměru okna/otevření na mobilu je v horní části stránky. UI vidíme na obrázku č. (21). V tomto formuláři jsou 4 interaktivní prvky, 3 z nich inputové a všechny z nich musí být vyplněny. První je výběr kryptoměny – MUI autocomplete, který bere data z Coingecko API, respektive její název a obrázek. Nachází se zde 200 nejpopulárnějších kryptoměn. Více kryptoměn není podle mého názoru potřeba, jelikož pro uživatele, na které touto aplikací mířím, nejsou malé a neznámé kryptoměny relevantní.



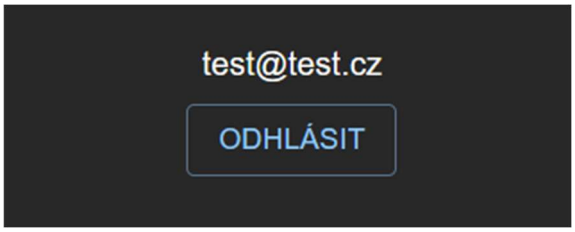
Obrázek 21 Formulář na přidávání kryptoměn

Dále počet měny, což je jednoduchý textfield stejně jako další inputy. Dá se zde vložit jakékoliv kladné číslo, až do nějakých 6 desetinných míst. Více desetinných míst nebylo podle mého názoru potřeba. Tento input je ošetřen, aby uživatel nemohl zadat negativní číslo.

Datum zakoupení, zde se vybírá den, kdy uživatel kryptoměnu nakoupil. Input je ošetřen, aby uživatel nemohl zadat budoucí den, nebo den kdy ještě kryptoměna nebyla zaznamenávána Coingecko API. Defaultní hodnota je zde dnešní datum.

Poslední je tlačítko potvrdit, které kontroluje, jestli jsou všechny data vyplněná a validní, následně spouští funkci a zapisuje data do databáze a současně se zobrazí v listu na stránce.

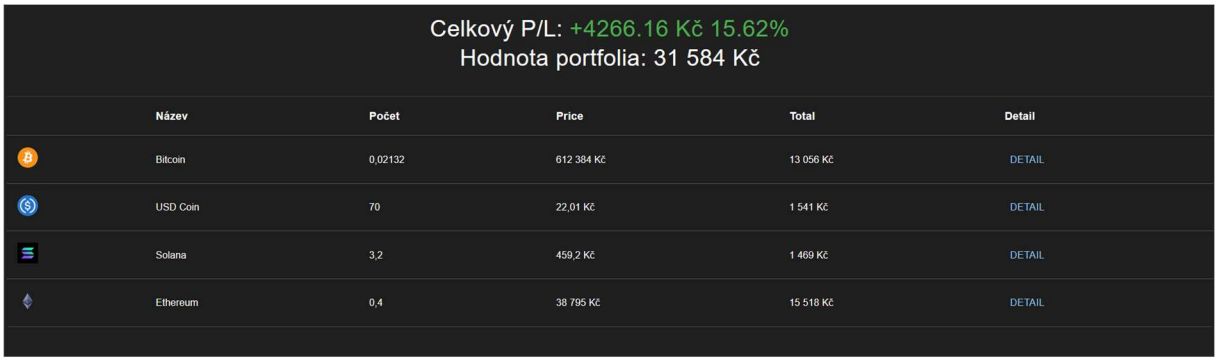
5.3.2 Odhlášení uživatele



Obrázek 22 Odhlašování

Kód se nachází v komponentě Logout a na stránce se nachází v levém horním rohu stránky, popřípadě při malém okně nebo otevření aplikace v mobilu se nachází v horní části stránky. Je zde e-mail uživatele, aby věděl, na jakém účtu je přihlášen a tlačítko odhlásit, které spustí funkci na odhlášení a zároveň uživatele přesměruje na login stránku. Vidíme na obrázku č. (22)

5.3.3 Seznam přidaných kryptoměn



Obrázek 23 Seznam kryptoměn

Na obrázku vidíme jednu z důležitějších částí aplikace – seznam přidaných kryptoměn. V horní části obrázku č. (23) je Celkový profit/lose, vypočtený z celkové dnešní hodnoty

všech přidaných kryptoměn mínus hodnota všech kryptoměn v den, ve kterém byli nakoupeny. Tyto data se všechny berou z CoinGecko API, jak historické zjištěné podle UNIX timestampu, tak aktuální. Data získané z API se cachují, aby byla aplikace výkonnější, méně náročná na network traffic a aby uživateli nedošel dovolený počet requestů do API, jelikož free verze API je má omezené.

UNIX timestamp je počet celkově uběhnutých sekund od 1. ledna 1970, hojně se využívá v IT obecně, protože je velice snadné s nimi pracovat, porovnávat nebo třídit, a navíc jsou velice jednoduché. [17]

Pod tím je hodnota portfolia, což jsou pouze sečtené hodnoty násobku aktuální ceny a počtu kryptoměn. Dále je už samotný list přidaných kryptoměn, kde se bere z databáze jméno, id kryptoměny a její počet. Pokud mají kryptoměny stejné jméno, tak se spojí jejich počet a vypíše se pouze v jednom řádku, i přesto že jsme nakoupili třeba 4x. Dále se podle její id vyfiltrují data z API a všechno se to vypíše tak, jak vidíte na obrázku.

5.3.3.1 Celkový profit/lose

Výpočet profitu nebo ztráty na hlavní stránce funguje jednoduše. Nejdříve se ke každé kryptoměně přidá přes API a filtrování aktuální cena. Následně se spustí javascript funkce „reduce“, kterou vidíme na obrázku č. (24). Ta slouží k procházení pole a sčítání určitých hodnot dohromady pomocí akumulátoru. Pokaždé co se projde jeden prvek v poli se přidá k akumulátoru (acc) násobek počtu a aktuální ceny. Takhle získáme sečtenou aktuální hodnotu všech uživatelem vlastněných kryptoměn. Funkce nám tedy vrátí jednu hodnotu.

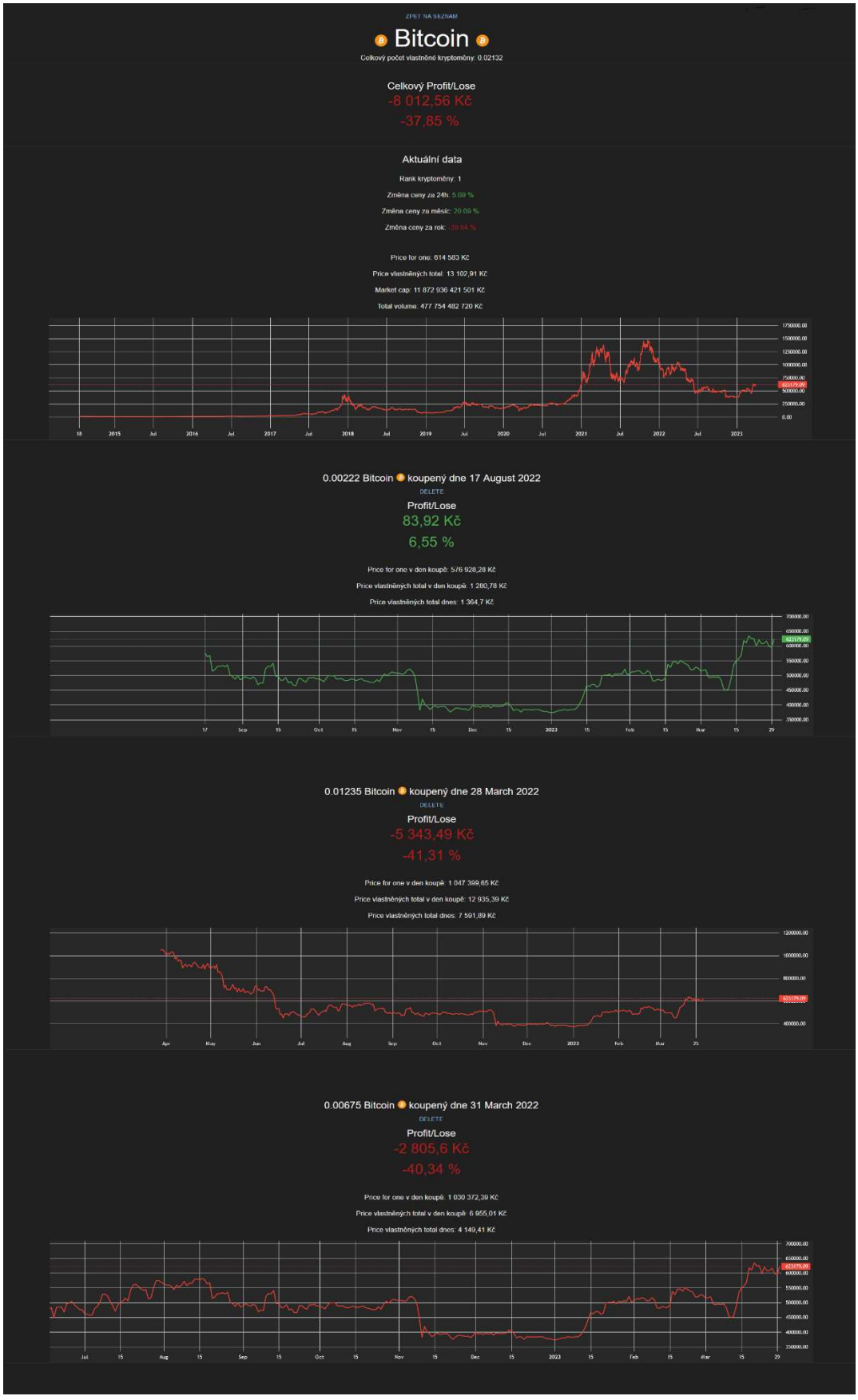
```
setCryptoSum(cryptoObj.reduce(
  (acc: number,
   crypto: { value: number; current_price: number; }) =>
   acc + crypto.value * crypto.current_price, 0)
);
```

Obrázek 24 Příklad kódu

Stejným způsobem se nastavuje i celková suma cen, za které jsme nakoupili. Zase se tu nejdřív přiřadí pro každou kryptoměnu cena v dobu nákupu podle UNIX timestampu, který zadal uživatel jako datum. Poté funkce „reduce“ projde pole a sečte násobky počtu a historické ceny.

Následně se už jen od celkové dnešní hodnoty portfolia odečte součet hodnot všech kryptoměn při koupi. Procenta se potom vypočítávají výdělek/prodělek děleno součtem hodnot všech kryptoměn při koupi.

5.3.4 Detail kryptoměny



Obrázek 25 Detail kryptoměny

Na obrázku č. (25) vidíme stránku po kliknutí na tlačítko DETAIL u jednotlivé kryptoměny, pro příklad byl zvolen Bitcoin. Úplně nahoře vidíme proklik zpátky na main page, název kryptoměny a počet, kolik té dané kryptoměny uživatel vlastní. Tyto prokliky fungují přes useState, který se předává mezi komponenty MainListComponent a DetailComponent.

Pod tím máme Profit/lose, v korunách a v procentech, jeho barva se mění podle toho, jestli jsme v plusu nebo v mínusu. Jeho výpočet je popsán zde.

Dále je položka Aktuální data, tu vidíme na obrázku č. (26). V její první části se nachází rank kryptoměny podle Market Capu, na obrázku můžeme vidět, že Bitcoin je první. Dále změny cen v procentech za určité časové období, v našem případě za 24 hodin, 1 měsíc a 1 rok.

Market cap – násobek momentální ceny a kolik kusů kryptoměny je v oběhu, zjednodušeně to je měřítko velikosti dané kryptoměny.

Total volume – celkové množství obchodovaných kryptoměn v daném časovém úseku, dá se z něho zjistit aktivita a likvidita trhu s touto kryptoměnou.

Obrázek 26 Aktuální data

Zde vidíme první graf, tento pokrývá celé období od kdy CoinGecko API kryptoměnu sleduje. Díky Lightweight charts je vytvoření těchto grafů poměrně jednoduché, pouze byli komponentě Chart předány data z API a graf je hotový. Linie grafu mění barvu podle toho, jestli jsme v profitu nebo ne. Linie X zobrazuje měsíce a linie Y cenu kryptoměny v Kč.

Dále je jeden nebo více itemů, podle toho, kolik jsme přidali kryptoměn. Jsou vytvořeny procházením pole „cryptos“ mapovací funkcí. Díky tomuto přístupu vidí uživatel informace pro každý jeho nákup kryptoměny zvlášť, s tím že předpokládáme že pokaždý byl nákup proveden v jiný den. Zde například na obrázku č. (27) můžeme vidět, že samotný nákup ze dne 17. srpna by byl výdělečný. Ale kvůli nakoupené 28. a 31. března je uživatel celkové v mínusu, protože oba dny byly prodělečné.

Obrázek 27 Kryptoměna koupená v určitý den

Každý item v tomto listu obsahuje jeho samostatný profit/lose, který zase mění barvu podle toho, jestli jsme v plusu nebo v mínusu. Jeho výpočet je popsán zde. Dále se zde nachází cena za jeden kus v den zakoupení, za kolik jsme nakoupili v den koupě a jaké je hodnota pouze tohoto nákupu dnes. Historická data jsou brána zase z API a díky nim a počtu kryptoměny získané z databáze, kterou zadal uživatel, je vypočtena celková cena v den koupě a dnes.

Graf se zase skládá pouze z dat získaných z API a v tomto případě graf začíná, až v den, kdy byla kryptoměna koupena. Například v nákupu z 17. srpna 2022 začíná graf v bodě 17. srpna 2022. Graf stejně jako předtím mění barvu linie podle toho, jestli jsme v profitu nebo ne, na pravé strany jsou ceny a v dolní části datumy.

V každém itemu se nachází tlačítko na odstranění této přidané měny, které po kliknutí vymaže kryptoměnu koupenou v danou dobu.

5.3.4.1 Profit/lose

V detailu kryptoměny jsou dva typy profit/lose výpočtu. První si popíšeme celkový výdělek ze všech. Jako první se vyfiltrují všechny kryptoměny z databáze s tím určitým id a přes funkci reduce se sečte jejich celkový počet. Ten je následně vynásoben aktuální cenou a od toho se odečte součet násobků počtu a ceny v době nákupu. Samozřejmě cena v době nákupu je předem nastavena. Tento profit/lose funguje stejně jako ten na hlavní stránce.

Jako druhý máme profit/lose pro každou jindy koupenou kryptoměnu zvlášť. Funguje v podstatě stejně jako ty, které jsme si popsali předtím. Násobek historické ceny a počtu se odečte od aktuální ceny a počtu. Při procentech se pouze číslo předtím vypočtené vydělí násobkem počtu a historické ceny.

5.4 Responzivita webu

Web je navržen pro responzivitu jak pro mobilní telefony, tak i pro tablety. V tomto případě se sidebar posune do horní části stránky, stejně jako odhlášení uživatele. U listu kryptoměn se pří malé velikosti objeví slider, díky kterému se v něm můžeme pohybovat. Tento přístup jsem zvolil proto, aby to bylo uživatelsky přívětivější – vše se dá lehce přečíst, není to chaotické a stále návrhově podobné.

6. Závěr

Cílem projektu bylo vytvořeni systému pro sledování kryptoměn a splňuje skoro vše, co bylo zadáno. Jediná věc, co by se dala vytknout je, že není konverze mezi fiat měnou a korunou. CoinGecko poskytuje už jako fiat měnu právě korunu a o konverzi se starají oni a jejich API.

V aplikaci jsou všechny důležité features, přihlašovaní, registrace, přidávání, odebírání a detail kryptoměn, zobrazení všech grafů, historické údaje a různé profity či ztráty, jak k celému portfoliu, tak pouze k jednotlivým kryptoměnám nebo pouze jednotlivým nákupům.

Určitě se dá vytknout grafické zpracování, ale nemám mnoho zkušeností s grafickým návrhem webu. Aplikace je tudíž graficky udělána velice minimalisticky.

Práce na projektu mě bavila, i přesto že ze začátku jsem si s technologiemi moc nerozuměl. Díky řešení spousty problémů jsem se naučil mnoho nových věcí a získal hodně nových znalostí. Projekt bych ještě rád lehce upravil, hlavně po grafické stránce. Aplikace je otestována a měla by být plně funkční a připravena na používání.

Reference

[1] React (JavaScript library). In: *Wikipedia: the free encyclopedia* [online]. San Francisco (CA): Wikimedia Foundation, 2001- [cit. 2023-03-29]. Dostupné z: https://en.wikipedia.org/wiki/React_(JavaScript_library)

[2] *Quick Start – React* [online]. [cit. 2023-03-29]. Dostupné z: https://react.dev/learn

[3] *What is Next.js | Learn Next.js* [online]. [cit. 2023-03-29]. Dostupné z: https://nextjs.org/learn/foundations/about-nextjs

[4] *Úvod do Node.js* [online]. [cit. 2023-03-29]. Dostupné z: https://www.itnetwork.cz/javascript/nodejs/uvod-do-nodejs

[5] *Úvod do TypeScryptu* [online]. [cit. 2023-03-29]. Dostupné z: https://www.itnetwork.cz/javascript/typescript/uvod-do-typescriptu

[6] Hypertext Markup Language. In: *Wikipedia: the free encyclopedia* [online]. San Francisco (CA): Wikimedia Foundation, 2001- [cit. 2023-03-29]. Dostupné z: https://cs.wikipedia.org/wiki/Hypertext_Markup_Language

[7] Kaskádové styly. In: *Wikipedia: the free encyclopedia* [online]. San Francisco (CA): Wikimedia Foundation, 2001- [cit. 2023-03-29]. Dostupné z: https://cs.wikipedia.org/wiki/Kask%C3%A1dov%C3%A9_styly

[8] What is Google Firebase?. In: *Wikipedia: the free encyclopedia* [online]. San Francisco (CA): Wikimedia Foundation, 2001- [cit. 2023-03-29]. Dostupné z: https://cs.wikipedia.org/wiki/Kask%C3%A1dov%C3%A9_styly

[9] *Firestore documentation* [online]. [cit. 2023-03-29]. Dostupné z: https://cloud.google.com/firestore/docs

[10] *Firebase – Introduction* [online]. [cit. 2023-03-29]. Dostupné z: https://www.geeksforgeeks.org/firebase-introduction/

[11] *Overview – Material UI* [online]. [cit. 2023-03-29]. Dostupné z: https://mui.com/material-ui/getting-started/overview/

[12] *Axios React* [online]. [cit. 2023-03-29]. Dostupné z: https://www.freecodecamp.org/news/axios-react-how-to-make-get-post-and-delete-api-requests/

[13] *Introduction – CoinGecko API Guide* [online]. [cit. 2023-03-29]. Dostupné z: https://apiguide.coingecko.com/getting-started/introduction

[14] *Getting started | Lightweight Charts* [online]. [cit. 2023-03-29]. Dostupné z: https://tradingview.github.io/lightweight-charts/docs

[15] *What is CoinStats App?* [online]. [cit. 2023-03-29]. Dostupné z: https://help.coinstats.app/en/articles/1537032-what-is-coinstats-app

[16] *Delta App Review* [online]. [cit. 2023-03-29]. Dostupné z: https://thecollegeinvestor.com/38209/delta-app-review/

[17] *Unix Time Stamp* [online]. [cit. 2023-03-29]. Dostupné z: https://www.unixtimestamp.com/

Reference obrázků

[18] *React icon* [online]. In: . [cit. 2023-03-29]. Dostupné z: https://commons.wikimedia.org/wiki/File:React-icon.svg

[19] *Next.js icon* [online]. In: . [cit. 2023-03-29]. Dostupné z: https://commons.wikimedia.org/wiki/File:Nextjs-logo.svg

[20] *Node.js icon* [online]. In: . [cit. 2023-03-29]. Dostupné z: https://cs.m.wikipedia.org/wiki/Soubor:Node.js_logo.svg

[21] *Typescript icon* [online]. In: . [cit. 2023-03-29]. Dostupné z: https://cs.m.wikipedia.org/wiki/Soubor:Typescript_logo_2020.svg

[22] *HTML5 icon* [online]. In: . [cit. 2023-03-29]. Dostupné z: https://cs.m.wikipedia.org/wiki/Soubor:HTML5_logo_and_wordmark.svg

[23] *CSS3 icon* [online]. In: . [cit. 2023-03-29]. Dostupné z: https://commons.wikimedia.org/wiki/File:CSS3_logo_and_wordmark.svg

[24] *Firebase icon* [online]. In: . [cit. 2023-03-29]. Dostupné z: https://commons.wikimedia.org/wiki/File:Firebase_Logo.svg

[25] *MUI* [online]. In: . [cit. 2023-03-29]. Dostupné z: https://mui.com/

Seznam obrázků

Obrázek 1 React logo [18] 1
Obrázek 2 Next.js logo [19] 2
Obrázek 3 Node.JS logo [20] 2
Obrázek 4 TypeScript logo [21] 2
Obrázek 5 HTML logo [22] 3
Obrázek 6 CSS logo [23] 3
Obrázek 7 Firebase logo [24] 4
Obrázek 8 MUI logo [25] 5
Obrázek 9 Architektura 7
Obrázek 10 pages 8
Obrázek 11 komponenty 8
Obrázek 12 public & styles 8
Obrázek 13 Firestore struktura 9
Obrázek 14 Firebase authentication 10
Obrázek 15 Stránka registrace 11
Obrázek 16 Specifické error zprávy 11
Obrázek 17 Výchozí error zpráva 12
Obrázek 18 Stránka přihlášení 12
Obrázek 19 Specifické error zprávy 13
Obrázek 20 Výchozí error zpráva 13
Obrázek 21 Formulář na přidávání kryptoměn 13
Obrázek 22 Odhlašování 14
Obrázek 23 Seznam kryptoměn 14
Obrázek 24 Příklad kódu 15
Obrázek 25 Detail kryptoměny 16
Obrázek 26 Aktuální data 17
Obrázek 27 Kryptoměna koupená v určitý den 18