

Галицький фаховий коледж імені В'ячеслава Чорновола
відділення комп'ютерних та видавничих технологій
циклова комісія інформатики та комп'ютерних дисциплін

ДОПУСТИТИ ДО ЗАХИСТУ

Завідувач відділенням
комп'ютерних та видавничих
технологій

Чубей О.О. / _____ /

підпис

«__» _____ 2022 р.

ПОЯСНЮВАЛЬНА ЗАПИСКА

до дипломного проєкту
освітньо-кваліфікаційного рівня «молодший спеціаліст»
зі спеціальності 122 «Комп'ютерні науки»
на тему : «Сайт моніторингу ринку криптовалют»

Студент групи КН-41 Кравчун О.М.

(підпис)

Керівник проєкту Чубей О.О.

(підпис)

Консультанти:

з техніко-економічного
обґрунтування

Меленчук Л.І.

(підпис)

нормоконтролер

Сиротюк Н.С.

(підпис)

Галицький фаховий коледж імені В'ячеслава Чорновола
відділення комп'ютерних та видавничих технологій
циклова комісія інформатики та комп'ютерних дисциплін

ЗАТВЕРДЖУЮ

Завідувач відділенням
комп'ютерних та видавничих
технологій

Чубей О.О. / _____ /
підпис

« ____ » _____ 2021 р.

ЗАВДАННЯ
на дипломне проєктування
на здобуття освітньо-кваліфікаційного рівня «молодший спеціаліст»
студенту

_____ (прізвище, ім'я та по-батькові студента)

1. Тема проєкту

затверджена наказом по коледжу від “ ____ ” _____ 202_ р., № ____

2. Термін здачі студентом завершеного проєкту “ ____ ” _____ 202_ р

3. Вихідні дані до проєкту _____

4. Перелік питань, які повинні бути розроблені в проєкті: _____

а) основна частина _____

б) техніко-економічного обґрунтування _____

5. Перелік графічного матеріалу _____

6. Консультанти проєкту: _____

Розділ	Консультанти	Підпис, дата	
		Завдання видано	Завдання прийнято
з техніко-економічного обґрунтування	<div></div> <div>(вчена ступень, звання П.І.Б.</div> <div></div> <div>консультанта)</div>		

КАЛЕНДАРНИЙ ПЛАН дипломного проєктування

№ п/п	Найменування етапу	Терміни	
		початку	завершення
1.	Вибір теми, ознайомлення з вимогами до дипломного проєктування.	29.09.21 р.	01.10.21 р.
2.	Огляд типових рішень та написання відповідного розділу ПЗ	06.12.21 р.	26.01.22 р.
3.	Дослідження технологій реалізації та написання відповідного розділу ПЗ	26.01.22 р.	14.02.22 р.
4.	Розробка функціональних вимог до проєкту та робота над структурою програмного продукту. Написання відповідного розділу ПЗ	14.02.22 р.	02.03.22 р.
5.	Встановлення на налаштування середовища реалізації та написання відповідного розділу ПЗ	02.03.22 р.	16.03.22 р.
6.	Проеєтування програмного засобу (функціоналу, інтерфейсу, бази даних продукту) та написання відповідного розділу ПЗ	16.03.22 р.	17.04.22 р.
7.	Реалізація та налаштування програмного засобу та написання відповідного розділу ПЗ	17.04.22 р.	03.05.22 р.
8.	Доопрацювання модулів	05.05.22 р.	18.05.22 р.
9.	Тестування на налагодження програмного продукту та написання відповідного розділу ПЗ	18.05.22 р.	01.06.22 р.
10.	Опрацювання економічного розділу дипломного проєкту та оформлення спеціального розділу	20.05.22 р.	05.06.22 р.
11.	Робота над оформленням пояснювальної записки	05.06.22 р.	12.06.22 р.
12.	Попередній захист дипломного проєкту, доопрацювання	15.06.22 р.	15.06.22р
13.	Підготовка до захисту дипломного проєкту	16.06.22 р.	24.06.22 р.
14.	Захист дипломного проєкту	25.06.22 р.	26.06.22 р.

7. Дата видачі завдання ” ____ ” _____ 2021 р.

Керівник _____ / _____

Завдання прийняв до виконання _____ / _____

Реферат

Дипломний проєкт. Тема: «Сайт моніторингу ринку криптовалют». 64 сторінки, 46 рисунків, 1 таблиця, 10 джерел, 1 додаток.

Основною метою дипломного проєкту є створення веб-сервісу для моніторингу найпопулярніших криптовалют та місць де їх можна безпечно обміняти. Перед створенням слід проаналізувати вже існуючі рішення та використати всі їх переваги.

Дипломний проєкт було реалізовано в середовищі розробки WebStorm яке було розроблено саме для роботи з веб-технологіями. Застосовано сучасні мови інструменти веб розробника, а саме ReactJS (JavaScript бібліотека), для написання клієнтської частини та NodeJS (JavaScript платформа) для написання серверної частини, а також базу даних MongoDB для збереження потрібних даних. Також використані інші дрібні бібліотеки.

Готовий проєкт повинен допомагати стежити за ринком криптовают і обмінників де їх можна вигідно та безпечно продати, а також регулярно оновлення новин для того щоб користувачі мали всі необхідні дані для подальших прогнозів на ринку.

REACTJS JS, NODE JS, EXPRESS JS, MONGOOSE, MONGODB, JAVASCRIPT, HTML, CSS, AXIOS.

Abstract

Diploma project. Topic: " Cryptocurrency market monitoring site". 64 pages, 46 figures, 1 table, 10 sources, 1 appendix.

The main goal of the graduation project is to create a web service for monitoring the most popular cryptocurrencies and places where they can be safely exchanged. Before creating, you should analyze existing solutions and use all their advantages.

The graduation project was implemented in the WebStorm development environment, designed specifically for working with web technologies. Modern languages were used for web developer tools, namely ReactJS (JavaScript library) for writing the client part and NodeJS (JavaScript platform) for writing the server part, as well as the MongoDB database for saving the necessary data. Other smaller libraries are also used.

The finished project should help to monitor the market of cryptocurrencies and exchangers where they can be sold profitably and safely, as well as regular news updates so that users have all the necessary data for further forecasts on the market.

REACTJS JS, NODE JS, EXPRESS JS, MONGOOSE, MONGODB, JAVASCRIPT, HTML, CSS, AXIOS.

ЗМІСТ

Вступ.....	6
1 Аналіз предметної області та постановка завдань.....	7
1.1 Обґрунтування доцільності створення веб-сервісу.....	7
1.2 Огляди існуючих рішень.....	8
1.3 Постановка завдання	15
2 Проєктування web-сервісу	16
2.1. Інструментарій для реалізації проєкту	16
2.1.1 Середовище розробки	16
2.1.2 Клієнтська сторона.....	18
2.1.3 Серверна сторона.....	21
2.1.4 Бази даних	25
2.2 Проєктування структури веб-сервісу.....	26
2.3 Проєктування користувацького інтерфейсу сайту.....	28
3 Реалізація та тестування веб-сервісу.....	30
3.1 Вибір інструментів реалізації	30
3.2 Створення клієнтської частини	33
3.3 Створення бази даних.....	40
3.4 Створення серверної частини	41
3.5 Тестування функціоналу сайту.....	43
4 Техніко-економічне обґрунтування	49
4.1 Аналіз ринку.....	49
4.2 Розрахунок витрат на проєктування	49
4.3 Обґрунтування необхідності розробки.....	51
Висновки	51
Перелік джерел посилання	54
Додатки.....	55

					ДП.КН.22.468.01.000		
Змн.	Арк.	№ докум.	Підпис	Дата	САЙТ МОНІТОИНГУ РИНКУ КРИПТОВАЛЮТ		
Розроб.		Кравчун О.М.					
Перевір.		Чубей О.О.					
Реценз.		Кузик В.М.					
Н. Контр.		Сиротюк Н.С.					
Затверд.		Чубей О.О.					
					Літ.	Арк.	Акрушів
						5	63
					ГФК.ВКВТ.ЦКІКД КН-41		

ВСТУП

Сучасний світ дуже швидко розвивається, постійно впроваджуються нові технології, особливо швидко розвиваються інформаційні технології. Які на сьогоднішній день дуже сильно вплили на наше повсякденне життя. Однією з найбільш розповсюджених та використовуваних зараз технологій є інтернет. Саме через мобільний банкінг ми проводимо найбільше розрахунків, але грошова система також не стоїть на місці. Розробниками зі всього світу стали запроваджуватися крипто-гроші, це призвело до створення сайтів криптообмінників. Велика кількість обмінників в свою чергу породила сайти для їх перегляду. На таких сайтах можна відстежувати самі криптовалюти, а головне це те де їх можна безпечно купити або обміняти.

Зараз недостатньо зробити просто хороший сайт, потрібне щось особливе та якісне, тим більше в сфері, яка постійно розвивається та стає все більш популярною.

Основним завданням дипломного проєкту є розв'язання поставлених завдань за допомогою знань та навичок здобутих в процесі навчання. А також є головним критерієм для оцінювання студента.

Основною метою дипломного проєкту є створення веб-сервісу для моніторингу найпопулярніших криптовалют та місць де їх можна безпечно обміняти. Перед створенням слід проаналізувати вже існуючі рішення та використати всі їх переваги.

					ДП.КН.22.468.01.000	Арк.
						6
Змн.	Арк.	№ докум.	Підпис	Дата		

1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ ТА ПОСТАНОВКА ЗАВДАНЬ

1.1 Обґрунтування доцільності створення веб-сервісу

Створення веб-сервісу – це комплекс заходів, що поєднують у собі розробку дизайну, інформаційне наповнення, застосування веб та маркетингових технологій, спрямованих на задоволення потреб відвідувачів та власників майбутнього сайту.

З розвитком інтернету у нашому житті з'явилося багато нових речей та можливостей. Одна з таких речей це криптовалюта. Першою криптовалютою був біткойн, на фоні його популяризації з'явилося багато нових валют, зараз їх налічується близько 19000, а світова капіталізація становить 2,14 трильйони доларів.

З виникненням такої кількості криптовалют з'явилася потреба в сервісі де нею можна торгувати та моніторити зміни на ринку. Тут на допомогу прийшли криптобіржі.

Що таке валютні біржі, цінні папери та товари – відомо багатьом, а хтось навіть займався поглибленим вивченням. 21 століття є епохою інформаційних технологій і віртуального світу, тому виникнення бірж криптовалют, що здійснюють торгівлю криптогрошима – явище цілком очікуване.

Криптобіржа – це електронний майданчик, де торгують криптовалютою. Як і на біржах традиційних валют, завдання учасників – отримати прибуток, тобто придбати дешевше, а продати дорожче. Хоча, багато експертів сходяться на думці, що це не що інше, як чергова "фінансова бульбашка", яка через якийсь час просто лусне. Проте такий підхід зовсім не вірний. Маючи достатній досвід, певні специфічні знання та володіючи інформацією, можна працювати так само ефективно, як і фондовому ринку.

Але перепродаж крипто це не єдиний спосіб її застосування. Вона чудово підходить для переказу та конвертації великих сум грошей між країнами. Основна перевага полягає в швидкості операції та малій комісії. Багато держав уже

					ДП.КН.22.468.01.000	Арк.
						7
Змн.	Арк.	№ докум.	Підпис	Дата		

легалізувати криптовалюту серед них і Україна, окрім того за крипту можна купити все більше товарів та послуг.

На фоні всіх цих подій з'явилося багато бірж та валют шахраїв і повстало питання, як відрізнити де і що можна купувати, а де ні. На допомогу прийшли ресурси, які збирають інформацію з усіх джерел та об'єднують її. За допомогою таких сервісів можна відслідковувати курси різних валют та популярність і надійність тієї чи іншої біржі. Також на таких ресурсах можна ділитися новинами та дізнаватися щось нове, вести портфоліо угод на біржах.

1.2 Огляди існуючих рішень

В інтернеті є багато подібних сервісів, і це дає змогу проаналізувати їх перед створенням власного сервісу.

Розглянемо три таких сервіси: CoinMarketCap, CryptoCompare та CoinCheckup.

CoinMarketCap – найбільш популярний та авторитетний ресурс, на якому є можливість моніторити валюту, біржі, NFT та вести звітність своєї діяльності на ринку (рисунок 1.1). Ресурс є безплатним та призначений для поверхової оцінки. Як тільки виходить якась валюта, то дані про неї одразу ж заносяться на сайт.

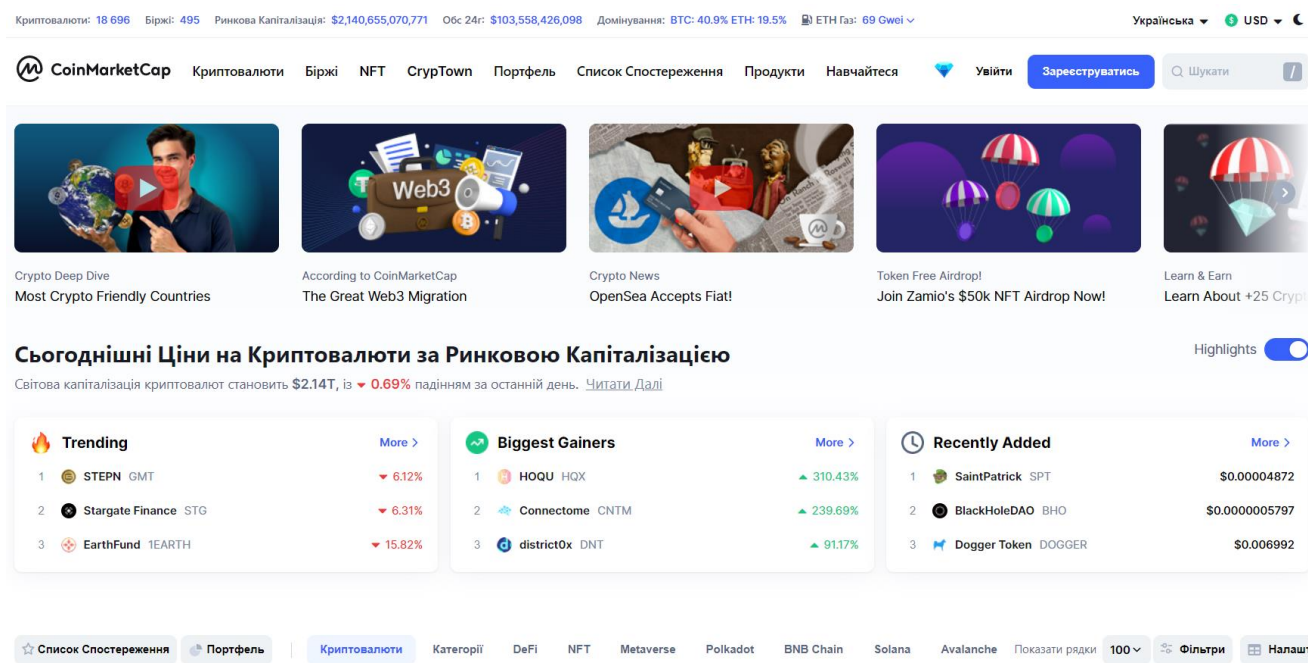


Рисунок 1.1 – Головна сторінка CoinMarketCap

					ДП.КН.22.468.01.000	Арк.
						8
Змн.	Арк.	№ докум.	Підпис	Дата		

Слідкувати за змінами на ринку криптовалют можна за допомогою зручної таблиці. На ній подана ціна, зміни в курсі, ринкова капіталізація, обсяг торгівлі за 24 години, кількість валюти в обігу та графік (рисунок 1.2). Інформація оновлюється кожні 5 хвилин. Дана таблиця має хороший та простий дизайн, зрозумілий з першого погляду, що є плюсом для новачків. Це дозволяє значно зменшити поріг входу та полегшити пошук потрібної криптовалюти.

<div> <div>Список Спостереження</div> <div>Портфель</div> <div>Криптовалюти</div> <div>Категорії</div> <div>DeFi</div> <div>NFT</div> <div>Metaverse</div> <div>Polkadot</div> <div>BNB Chain</div> <div>Solana</div> <div>Avalanche</div> <div>Показати рядки 100</div> <div>Фільтри</div> <div>Налашт.</div> </div>									
#	Ім'я	Ціна	24h %	7d %	Ринкова Капіталізація	Обсяг(24h)	Циркуляційний Запас	Останні 7 днів	
1	Bitcoin BTC Купити	\$45,929.35	▼1.17%	▼3.49%	\$877,152,224,338	\$30,709,580,121 665,289 BTC	19,002,531 BTC		⋮
2	Ethereum ETH Купити	\$3,451.61	▼1.41%	▲1.94%	\$417,082,245,314	\$16,982,407,256 4,895,867 ETH	120,240,864 ETH		⋮
3	Tether USDT	\$1.00	▼0.01%	▼0.02%	\$82,348,231,228	\$75,491,685,034 75,476,741,592 USDT	82,331,930,545 USDT		⋮
4	BNB BNB Купити	\$443.07	▼1.12%	▲0.82%	\$73,490,998,059	\$2,201,232,073 4,945,644 BNB	165,116,761 BNB		⋮
5	USD Coin USDC	\$0.9996	▼0.04%	▼0.02%	\$51,606,090,322	\$3,686,643,972 3,687,176,556 USDC	51,613,545,493 USDC		⋮
6	Solana SOL Купити	\$131.71	▼3.34%	▲16.54%	\$43,153,388,113	\$2,345,186,173 17,694,753 SOL	325,598,261 SOL		⋮
7	Cardano ADA	\$1.23	▲4.41%	▼0.08%	\$41,642,492,005	\$1,888,438,090 1,530,025,306 ADA	33,739,028,516 ADA		⋮
8	XRP XRP	\$0.8266	▼1.47%	▼6.65%	\$39,945,417,656	\$1,692,185,938 2,039,435,639 XRP	48,135,209,660 XRP		⋮

Рисунок 1.2 – Таблиця моніторингу криптовалют

Якщо натиснути на потрібну валюту то користувача буде переадресовано на сторінку з детальною інформацією про неї (рисунок 1.3). Тут присутні посилання на офіційні сторінки розробників, більш детальна інформація про монету та зручний графік, який бере початок від створення валюти, що дозволяє детально ознайомитися з обраною криптовалютою.

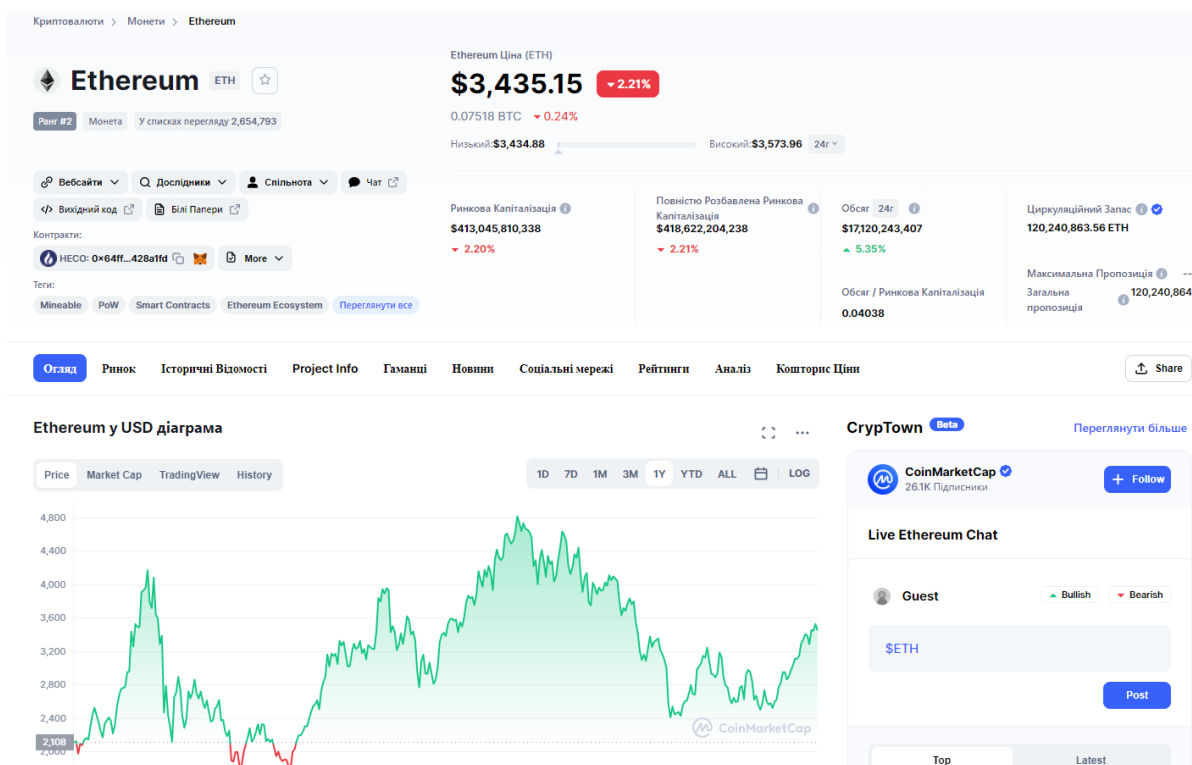


Рисунок 1.3 – Сторінка з детальною інформацією

Також на сайті доступний перегляд криптобірж, (рисунок 1.4) де зазначений їх грошовий обіг, кількість візитів, рейтинг, кількість валют та інші дані. Так з цією таблицею можна легко знайти криптобіржі з високим рівнем довіри.

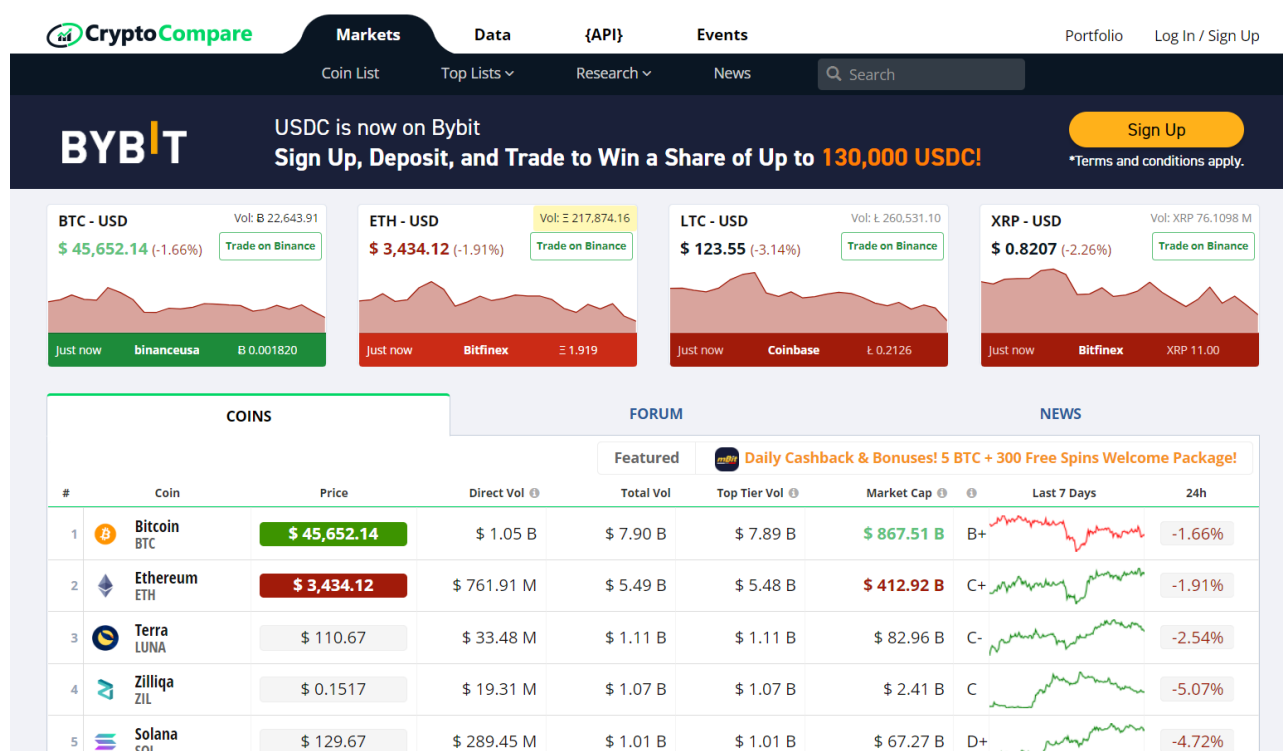
#	Name	Exchange Score	Volume(24h)	Avg. Liquidity	Weekly Visits	# Markets	# Coins	Fiat Supported	Volume Graph (7d)
1	Binance	9.9	\$19,015,864,357 ▲ 38.99%	837	22,060,223	1673	395	AED, ARS, AUD and +43 more	
2	FTX	8.3	\$2,350,616,522 ▲ 26.71%	735	4,718,412	420	284	USD, EUR, GBP and +7 more	
3	Coinbase Exchange	8.1	\$2,190,995,640 ▲ 18.6%	708	2,191,082	519	179	USD, EUR, GBP	
4	Kraken	7.8	\$678,780,957 ▲ 18.66%	751	1,661,405	556	174	USD, EUR, GBP and +4 more	
5	KuCoin	7.4	\$1,567,089,115 ▲ 21.98%	562	2,554,651	1318	710	USD, AED, ARS and +45 more	
6	Gate.io	7.1	\$1,848,799,723 ▲ 30.27%	477	3,363,357	2390	1418	KRW, EUR	
7	Gemini	7.1	\$123,485,219 ▲ 15.16%	685	426,379	121	98	USD, GBP, EUR and +4 more	
8	Huobi Global	7.0	\$2,025,538,369 ▲ 29.23%	517	964,449	1114	562	ALL, AUD, BRL and +47 more	

Рисунок 1.4 – Таблиця моніторингу криптобірж

Плюси даного ресурсу:

- простий інтерфейс;
- інформативні графіки;
- багато мов на вибір;
- конвертація в різні валюти

CryptoCompare – справжній портал (в сенсі масштабності та різноплановості) про криптовалюти (рисуюнок 1.5).



Рисуюнок 1.5 – Головна сторінка CryptoCompare

CryptoCompare не обмежується графіками та статистикою, публікує статті, топи та навчальні матеріали, збирає новини, підтримує форум та дозволяє відстежувати особистий портфель інвестицій. На сайті є розділи для трейдерів, майнерів, інвесторів та звичайних користувачів. Тут є відгуки та рейтинги криптобірж, гаманців, асиків, відеокарт та ICO-проектів.

Так виглядає портфель інвестицій (рисуюнок 1.6)

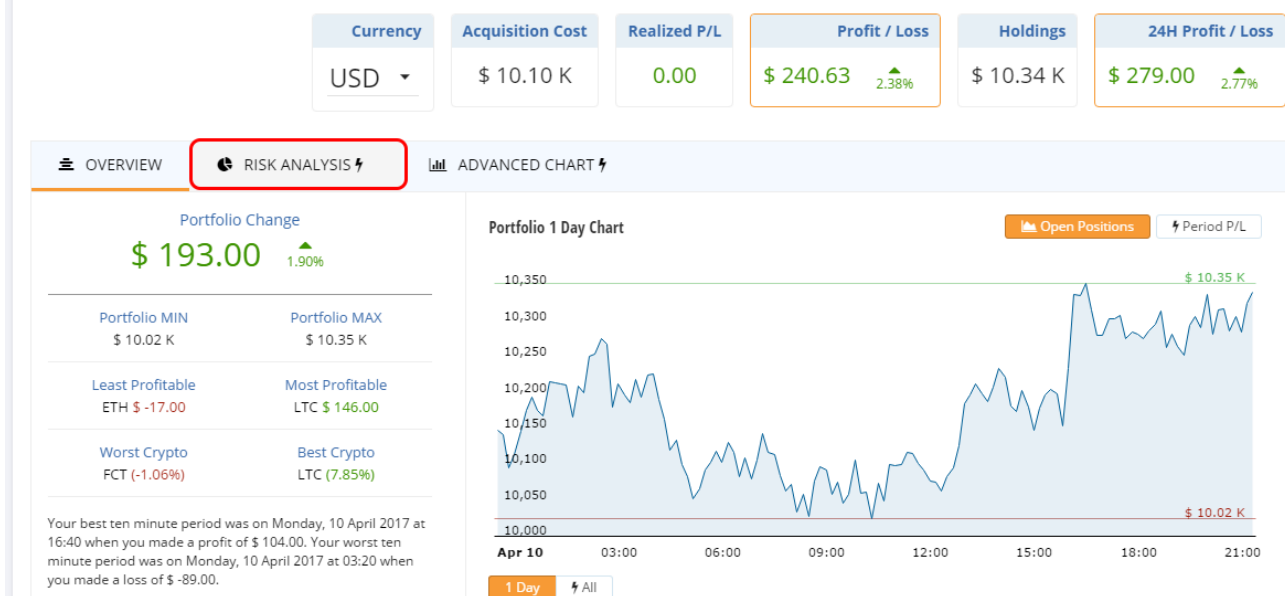


Рисунок 1.6 – Портфель інвестицій

Відвідувач може створити профіль, підписатися на криптовалюти або компанії, що цікавлять, відстежувати угоди на різних торгових майданчиках, вивчати історичні відомості та аналізувати вплив соцмереж на ринок. Дані про ціни на сайті змінюються у реальному часі.

На рисунку 1.7 зображено сторінку з новинами в даній сфері.

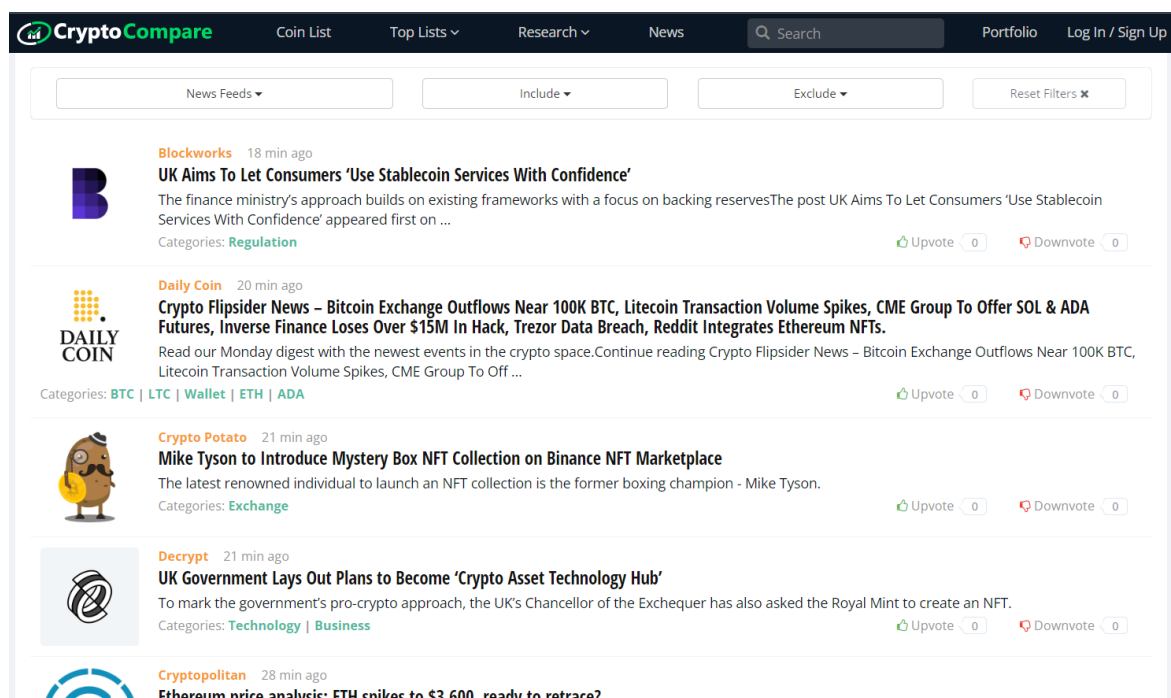


Рисунок 1.7 – Сторінка з новинами

					ДП.КН.22.468.01.000	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		12

Плюси CryptoCompare:

- розділ з новинами;
- розділ з навчальними матеріалами;
- зручний портфель.

З мінусів можна відзначити відсутність вибору мови.

CoinCheckup позиціонує себе як дослідницьку платформу для ринку криптовалют (рисунок 1.8). За його простим інтерфейсом ховаються цікаві можливості. Наприклад, прогноз цін на найближчий рік, зведення про доходність кожної монети з моменту її запуску та рейтинг криптовалют на основі купи об'єктивних даних: активності розробників, якості бренду та продукту, спілкування в соцмережах та інших особливостей конкретного проєкту.

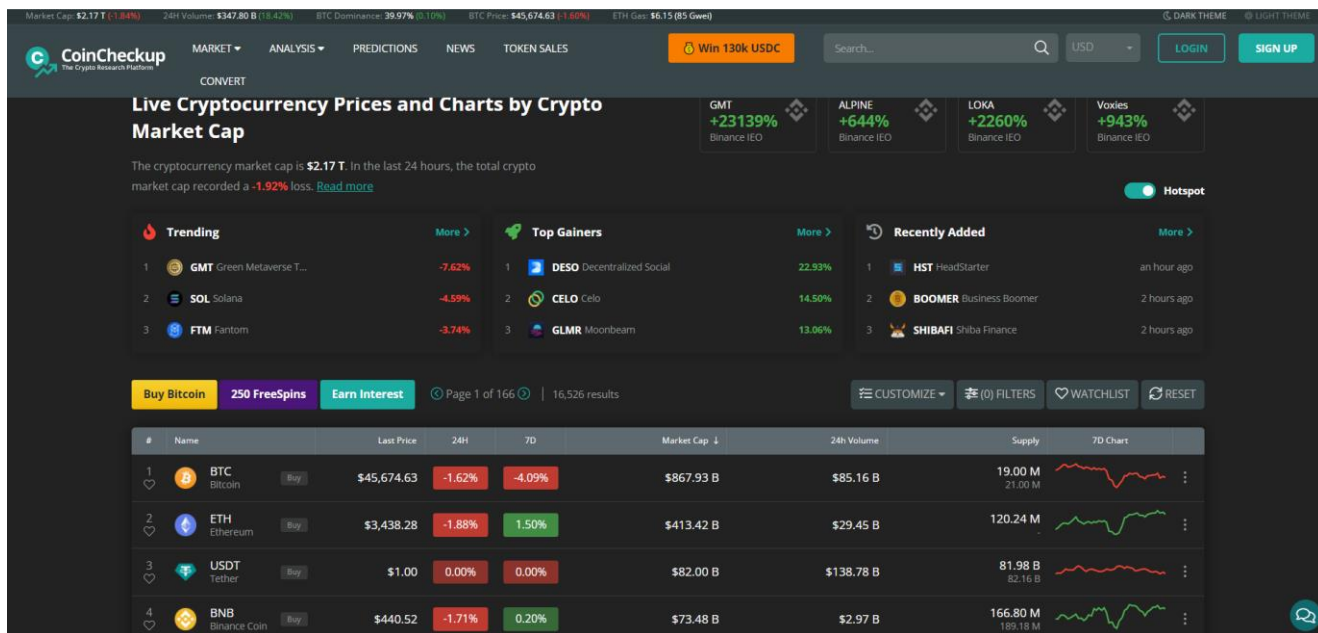


Рисунок 1.8 – Головна сторінка CoinCheckup

За заявами творців, CoinCheckup з'явився для того, щоб зробити ринок криптовалют прозорішим. Розробники займалися інвестиціями, не знайшли відповідного ресурсу та створили свій з фундаментальним аналізом та прогностичними алгоритмами.

З цікавих особливостей на сайті є функція передбачення курсу (рисунок 1.9)

					ДП.КН.22.468.01.000	Арк.
						13
Змн.	Арк.	№ докум.	Підпис	Дата		

#	Name	Last Price	2% BTC Growth	10% BTC Growth	50% BTC Growth	Internet	Mobile	Facebook	Google	4% WW MO
1	BTC Bitcoin	\$45,695.20	\$46,362.70	\$48,951.83	\$61,897.45	\$59,018.61	\$58,520.31	\$101,410.52	\$71,352.63	\$5,647.62
2	ETH Ethereum	\$3,440.21	\$3,491.99	\$3,687.00	\$4,662.05	\$4,445.22	\$4,407.69	\$7,638.14	\$5,374.21	\$425.37
3	USDT Tether	\$1.00	\$1.01	\$1.07	\$1.35	\$1.29	\$1.28	\$2.22	\$1.56	\$0.1236
4	BNB Binance Coin	\$440.22	\$446.44	\$471.37	\$596.03	\$568.31	\$563.51	\$976.52	\$687.08	\$54.38
5	USDC USD Coin	\$1.00	\$1.01	\$1.07	\$1.35	\$1.29	\$1.28	\$2.22	\$1.56	\$0.1235
6	SOL Solana	\$130.14	\$132.20	\$139.59	\$176.50	\$168.29	\$166.87	\$289.17	\$203.46	\$16.10
7	LUNA Terra	\$111.03	\$112.27	\$118.54	\$149.88	\$142.91	\$141.71	\$245.56	\$172.78	\$13.68
8	ADA Cardano	\$1.22	\$1.24	\$1.31	\$1.66	\$1.58	\$1.56	\$2.71	\$1.91	\$0.1510
9	XRP XRP	\$0.8227	\$0.8335	\$0.8801	\$1.11	\$1.06	\$1.05	\$1.82	\$1.28	\$0.1015
10	AVAX Avalanche	\$92.71	\$94.29	\$99.55	\$125.88	\$120.03	\$119.01	\$206.24	\$145.11	\$11.49
11	DOT Polkadot	\$22.14	\$22.48	\$23.74	\$30.01	\$28.62	\$28.37	\$49.17	\$34.60	\$2.10

Рисунок 1.9 – Функція передбачення курсу

Також на сайті присутній зручний конвертер криптовалют та просто валют (рисунок 1.10). Дозволяє з легкістю та по актуальному курсі конвертувати валюти.

Convert Ethereum/Ukrainian Hryvnia

1 ETH to UAH = 101,009 Ukrainian Hryvnia

AMOUNT

1

FROM

Ethereum ETH

TO

Ukrainian Hryvnia UAH

1 Ethereum =
101,009 Ukrainian Hryvnia

1 ETH = 101,009 UAH
1 UAH = 0.000010 ETH

For best rate, exchange your ETH to UAH on Binance.

Exchange ETH/UAH

Buy ETH

Ethereum to Ukrainian Hryvnia conversion. Mid-market ETH/UAH exchange rate was last updated on Apr 4, 2022, 17:51 UTC.

ETH to UAH Chart

LIVE

1 ETH = 101,009 UAH

Apr 4, 2022, 17:51 UTC

Рисунок 1.10 – Конвертер криптовалют

Плюси ресурсу:

- прогнозування курсу;
- приємний інтерфейс;
- конвертер валют.

					ДП.КН.22.468.01.000	Арк.
						14
Змн.	Арк.	№ докум.	Підпис	Дата		

З мінусів також відсутня зміна мови.

1.3 Постановка завдання

Потрібно проаналізувати наведені вище приклади, визначити всі плюси та мінуси, і на основі цього розробити свій аналог.

Метою дипломної роботи є створення сайту для отримання інформації з крипто-сфери, моніторинг графіків у реальному часі. Сайт повинен бути зрозумілим для новачків у цій сфері.

На сайті повинен бути реалізований такий функціонал:

- зручний інтерфейс;
- реєстрація;
- авторизація;
- вивід інформації в реальному часі;
- додавання нових постів;
- реалізований особовий кабінет;
- перегляд з будь-якого девайсу;
- інформація сайту повинна бути добре захищеною.

Також загальними критеріями для хорошого сайту є наповненість інформацією та її актуальність, швидка та безперебійна робота та хороша структура.

					ДП.КН.22.468.01.000	Арк.
						15
Змн.	Арк.	№ докум.	Підпис	Дата		

2 ПРОЄКТУВАННЯ WEB-SERVICE

2.1. Інструментарій для реалізації проєкту

2.1.1 Середовище розробки

Навколишній світ стає все більш пронизаний інформаційними технологіями. Автоматизується все, що може автоматизуватися, важко знайти підприємство на якому немає спеціалізованого програмного забезпечення. Через великий попит постійно розробляються нові рішення. І так саме як підприємству потрібне спеціалізоване ПЗ, так воно потрібно і його розробнику. Перші середовища розробки були створені для роботи через консоль, надалі середовища постійно вдосконалювалися для більш зручного і швидкого написання коду.

Середовище розробки – це сукупність програм, функціонал яких забезпечує розробку системи.

Зазвичай середовище розробки складається з таких елементів:

- редактор;
- компілятор;
- інтерпретатор;
- засоби автоматизації;
- налагоджувач.

Зараз на вибір є багато різних середовищ, здебільшого вони створені для вирішення певних завдань, проте є і для багатоцільові. Також багато середовищ можна налаштувати під свою потребу за допомогою плагінів. Плагінами можна розширити функціонал, пришвидшити та полегшити написання коду і налаштовувати оформлення робочого середовища. Проте жодне середовище не є універсальним, іноді для створення проєкту необхідно різні середовища.

Для написання веб-сайту достатньо і одного середовища. Зараз веб-технології стрімко розвиваються, а разом з ними розвиваються і ІСР. Зараз популярними є такі середовища WebStorm, Visual Studio Code, Sublime Text.

					ДП.КН.22.468.01.000	Арк.
						16
Змн.	Арк.	№ докум.	Підпис	Дата		

WebStorm – інтегроване середовище розробки для HTML, CSS, JavaScript та пов’язаними з ними технологіями (рисунок 2.1).

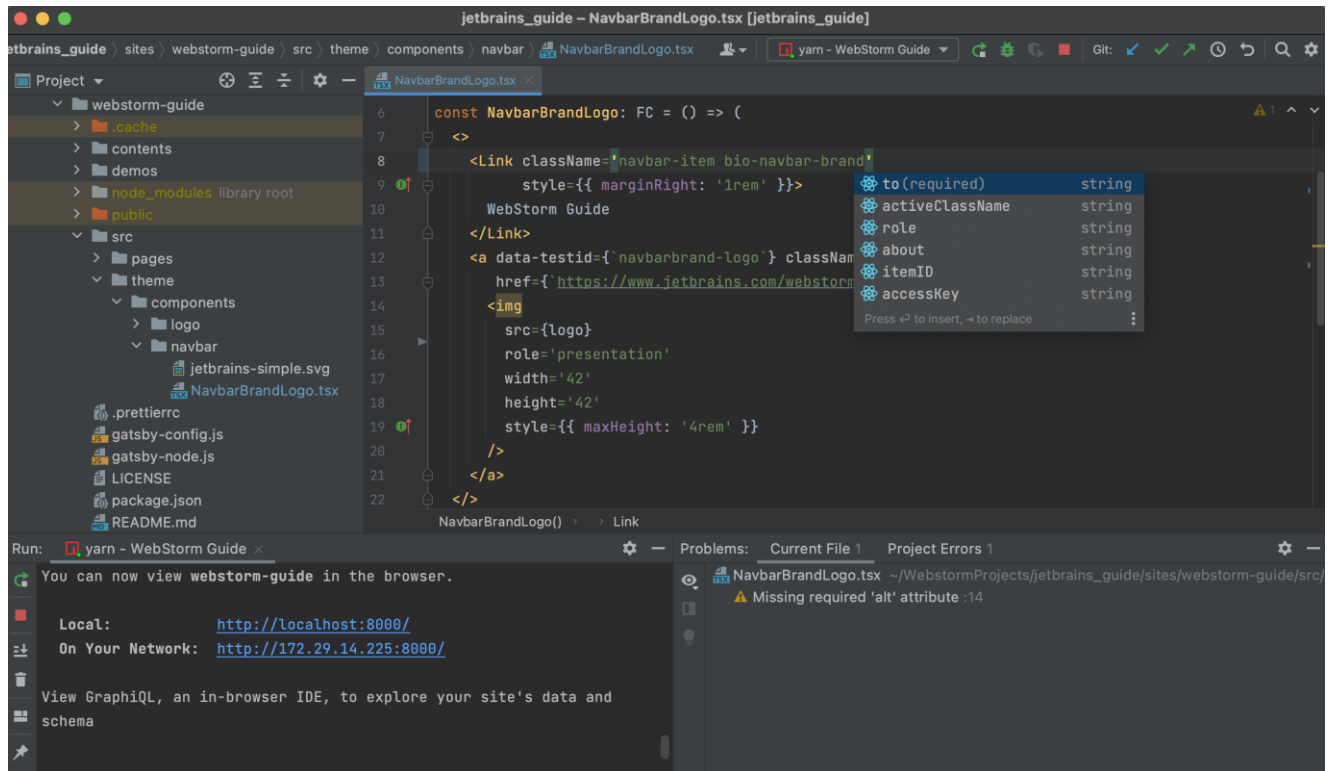


Рисунок 2.1 – Інтерфейс ІСР "WebStorm"

Ключові функції WebStorm:

- готова підтримка JavaScript, TypeScript, React, Vue, Angular, Node.js, HTML, таблиць стилів та інших;
- інтелектуальний редактор з автодоповненням коду та виявленням помилок;
- вбудовані інструменти розробника, які дозволяють налагоджувати та тестувати клієнтські програми та програми Node.js, а також працювати з контролем версій, інструментами збирання, терміналом та HTTP-клієнтом;
- інструменти для ефективної командної роботи;
- можливість налаштувати робоче середовище;
- фірма "JetBrains" має програму підтримки студентів.

					ДП.КН.22.468.01.000	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		17

Visual Studio Code – безкоштовний кросплатформовий редактор орієнтований на веб-розробку. Функції Visual Studio Code дозволяють користувачеві змінювати редактор відповідно до своїх потреб (рисунок 2.2).

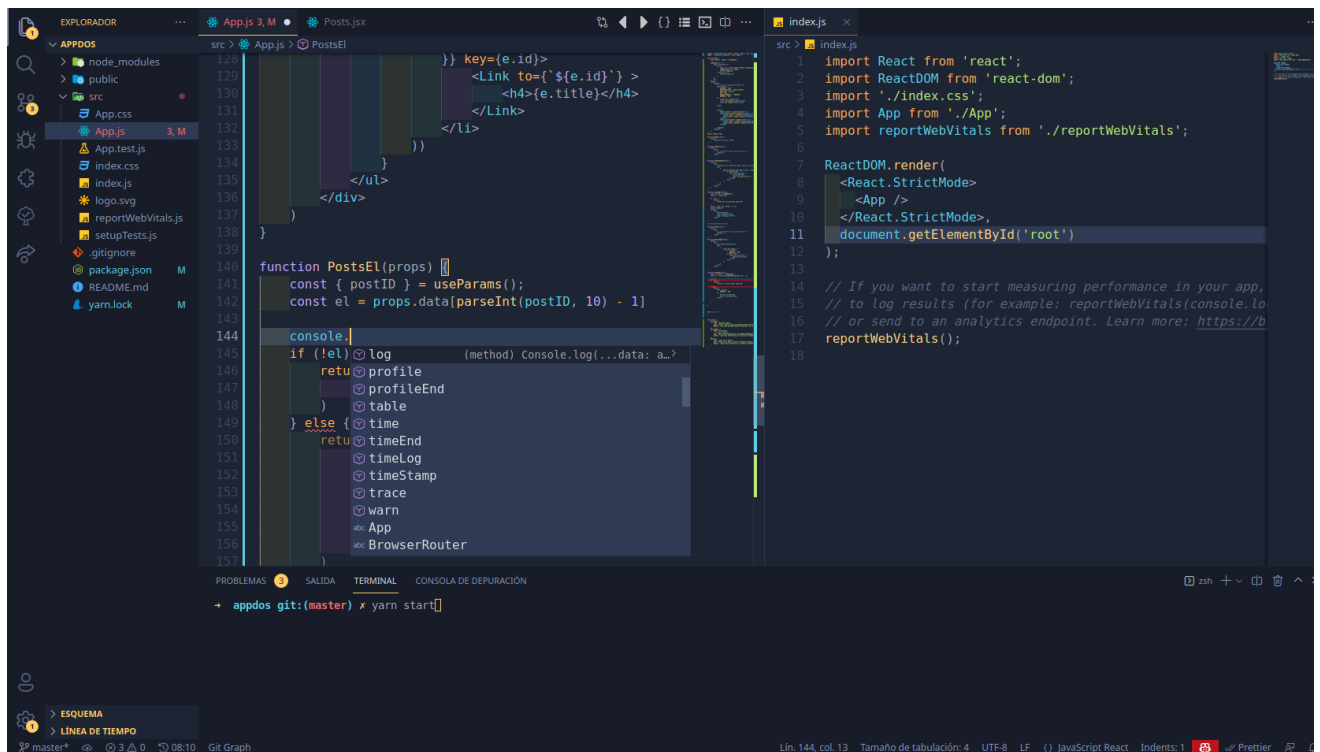


Рисунок 2.2 – Інтерфейс VS Code

Основні переваги VS Code:

- кросплатформний;
- підтримує багато мов програмування;
- налагодження;
- вбудована інтеграція Git;
- кастомізація;
- велика кількість плагінів;
- постійні оновлення;
- велика спільнота.

2.1.2 Клієнтська сторона

Веб додатки почали стрімко розвиватися в кінці 90-х. Одним з основних критеріїв додатку є його клієнтська сторона. Усі сучасні програми оснащені

					ДП.КН.22.468.01.000	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		18

графічним інтерфейсом, який полегшує взаємодію людини з комп'ютером. Інтерфейс являє собою сукупність засобів обробки інформації, якомога краще пристосованих для комфорту та зручності користувача.

В свою чергу веб-інтерфейс призначений для взаємодії з сервісом за допомогою веб-браузера. На початку свого розвитку веб-інтерфейси були дуже примітивними та будувалися одним лише HTML про якісь красиві сайти тоді навіть і не йшлося. По суті HTML являє собою звичайні будівельні блоки нічим не примітні та не красиві [1]. Проте на початку цього в повній мірі вистачало, але згодом постало питання створення більш красивих сторінок, які б були простіші для сприйняття користувачу. І тоді було розроблено мову стилів CSS, вона значною мірою покращила тодішні сайти.

CSS мова стилів спеціально розроблена для створення веб-інтерфейсів. З приходом CSS прийшов і новий блочний стиль верстки на заміну табличному [2]. Це значно полегшило верстку та зробило її різноманітнішою. Навіть до тепер деякі інтерфейси розробляють одними лише HTML та CSS.

Для полегшення та пришвидшення верстки сайту було створено фреймворк "Bootstrap". Bootstrap – набір інструментів для створення сайтів та веб-застосунків [3]. Дозволяє використовувати HTML та CSS – шаблони кнопок, форм, міток, блоків навігації та інших елементів інтерфейсу, включно з JS – скрипти.

Також для полегшення написання стилів було створено скриптову метамову SASS [4]. Ця мова значно полегшує написання коду дозволяє створювати змінні, міксини та цикли.

Та цих технологій замало для створення веб-сайту. Сучасні веб-сайти базуються на взаємодії людиною. Для надання інтерактивності було створено мову JavaScript

JavaScript отримала свою назву внаслідок великої популярності мови програмування Java, проте нічого спільного ці мови не мають [5]. З приходом JS розробники отримали змогу додавати на сайт інтерактивний інтерфейс. Тепер сайти могли видозмінюватися в процесі їх використання: додавалися нові HTML-

					ДП.КН.22.468.01.000	Арк.
						19
Змн.	Арк.	№ докум.	Підпис	Дата		

блоки, мінявся колір, елемент змінювався при наведенні або кліку по ньому мишкою.

Раніше вся клієнтська логіка ґрунтувалася на використанні бібліотеки JQuery, яка дозволяє працювати з DOM, анімацією на сторінці та робити AJAX запити.

JQuery – перша JS – бібліотека, яка отримала популярність та вивела JavaScript на новий рівень. З її створенням набули популярності односторінкові сайти. Широко використовувалася до 2013 року, але не витримала конкуренції нових бібліотек та фреймворків.

DOM – це структура HTML – сторінки. Робота з DOM – це пошук, додавання, зміна, переміщення та видалення HTML-тегів.

AJAX (asynchronous javascript and XML) – це загальна назва для технологій, які дозволяють робити асинхронні запити до сервера та обмінюватися даними. Оскільки клієнтська та серверна частини веб-програми написані різними мовами програмування, то для обміну інформацією необхідно перетворювати структури даних, в яких вона зберігається, в JSON-формат.

JSON (JavaScript Object Notation) – формат для обміну даними між клієнтом та сервером. Він являє собою простий рядок, який може бути використаний у будь-якій мові програмування.

За допомогою маніпуляцій з DOM можна повністю керувати вмістом сторінок. За допомогою AJAX можна обмінюватися даними між клієнтом та сервером. А з цими технологіями можна створити Single page application. Але при створенні складного додатка код фронтенду заснованого на JQuery швидко стає заплутаним і важко підтримується.

Та на заміну старій та не зрозумілій JQuery прийшло багато нових бібліотек та фреймворків. Найпопулярнішими зараз є AngularJS, Vue.js, React, та інші. Саме вони дозволяють писати односторінкові інтерфейси. Ці технології пропонують багато готових рішень зібраних за час існування JavaScript. Вони працюють за

					ДП.КН.22.468.01.000	Арк.
						20
Змн.	Арк.	№ докум.	Підпис	Дата		

допомогою платформи Node.js, що надає можливість використовувати JS на сервері.

Vue.js – це фреймворк для створення інтерфейсів користувача. Його ядро насамперед вирішує завдання рівня представлення, що спрощує інтеграцію з іншими бібліотеками та існуючими проєктами. З іншого боку, Vue повністю підходить і для створення складних односторінкових програм, якщо використовувати його спільно з сучасними інструментами та додатковими бібліотеками.

React – відкрита JS бібліотека призначення для створення веб-інтерфейсу, покликана вирішувати оновлення вмісту при односторінковій [6] розробці. Розробляється компанією та підтримується "Meta" з 2013 року. Наразі використовується багатьма компаніями, а саме: "Netflix", "Yahoo", "Sony" та інші.

Single page application (односторінковий інтерфейс) – веб застосунок, який вміщується на одній сторінці, основна перевага такого підходу максимально зменшення затрат часу та трафіку для загрузки сторінок. До приходу цієї технології кожна сторінка сайту завантажувалася окремо, при повільному інтернеті це призводило до тривалих загрузок. На відміну від багато сторінкового інтерфейсу цей завантажує лише одну сторінку з подальшим завантаженням окремих елементів сторінки. Тепер при переході на умовно нову сторінку з сервера завантажується тільки та інформація, що відрізняється від попередньої, непотрібно заново будувати цілу сторінку, а лише окремі дрібні елементи. Також це дало змогу виводити нові дані, які надійшли на сервер без перезавантаження сторінки. Прикладом є звичайний месенджер, JS надсилає запит на сервер та повертає результатом лише частину HTML – блоку, той, який слугує новим повідомленням. Тепер не потрібно оновлювати сторінку в очікуванні на повідомлення.

2.1.3 Серверна сторона

Більшість великих веб-сайтів використовують програмування серверної частини, щоб динамічно відображати різні дані при необхідності, в основному,

					ДП.КН.22.468.01.000	Арк.
						21
Змн.	Арк.	№ докум.	Підпис	Дата		

взяті з бази даних, що розташовується на сервері і надсилаються клієнту для відображення через деякий код (наприклад, HTML і JavaScript).

Можливо, найбільша користь програмування серверної частини полягає в тому, що воно дозволяє формувати контент веб-сайту під конкретного користувача. Динамічні сайти можуть виділяти контент, який актуальніший залежно від переваг і звичок користувача. Це також може спростити використання сайтів, за рахунок збереження особистих переваг та інформації, наприклад повторного використання збережених даних кредитної картки для оптимізації наступних платежів.

Це також дає можливість взаємодіяти з користувачем сайту, надсилаючи повідомлення та оновлення електронною поштою або іншими каналами. Всі ці можливості дозволяють глибше взаємодіяти з користувачами.

Веб-браузери взаємодіють із веб-серверами за допомогою гіпертекстового транспортного протоколу (HTTP). Коли ви натискаєте посилання на веб-сторінці, заповнюєте форму або запускаєте пошук, HTTP-запит відправляється з вашого браузера на цільовий сервер.

Запит включає URL, що визначає ресурс, метод, що визначає необхідну дію (отримати, видалити або опублікувати дані) і може включати додаткову інформацію, закодовану в параметрах URL (поле-значення, відправлені як рядок запиту), як POST запит (дані, надіслані методом HTTP POST) або в файли cookie.

Веб-сервери очікують повідомлень з клієнтськими запитами, обробляють їх після прибуття та відповідають веб-браузеру за допомогою відповідного HTTP повідомлення. Відповідь містить рядок стану, що показує, був запит успішним чи ні.

Схема нижче показує базову архітектуру веб-сервера для статичного сайту. Коли користувач хоче перейти на сторінку, браузер відправляє HTTP-запит "GET" із зазначенням його URL-адреси.

Сервер витягує запитаний документ зі своєї файлової системи та повертає HTTP-відповідь, що містить документ та успішний статус. Якщо файл не може бути вилучений з будь-якої причини, повертається статус помилки (рисунок 2.3).

					ДП.КН.22.468.01.000	Арк.
						22
Змн.	Арк.	№ докум.	Підпис	Дата		

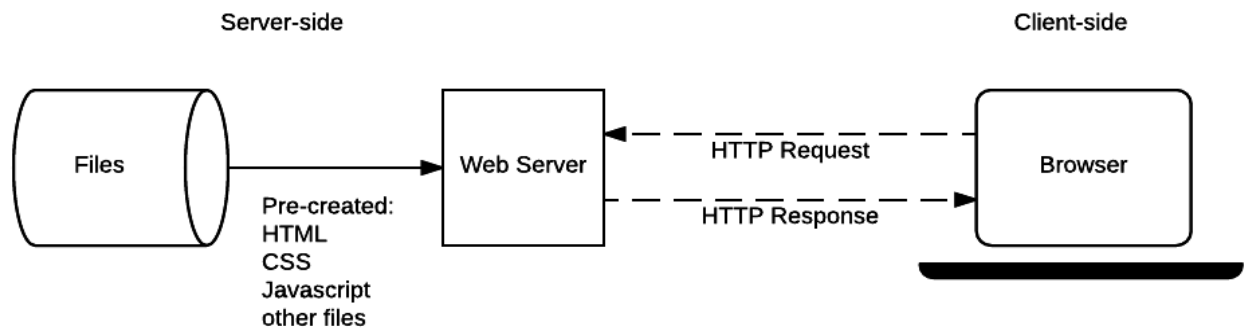


Рисунок 2.3 – Архітектура серверу для статичного веб-сайту

Динамічний веб-сайт – це той, де частина вмісту відповіді генерується динамічно лише за необхідності. На динамічному веб-сайті HTML-сторінки зазвичай створюються шляхом вставки даних із бази даних у заповнювачі в HTML-шаблони.

Динамічний сайт може повертати різні дані URL-адреси на основі інформації, наданої користувачем або збереженими налаштуваннями, і може виконувати інші операції, як частина повернення відповіді.

Більшість коду для підтримки динамічного веб-сайту має виконуватися на сервері. Створення цього коду відоме як "програмування серверної частини" або "програмування бекенду".

Схема нижче вказує просту архітектуру динамічного веб-сайту. Як і на попередній схемі, браузері відправляють HTTP-запити на сервер, потім сервер обробляє запити та повертає відповідні HTTP-відповіді (рисунок 2.4).

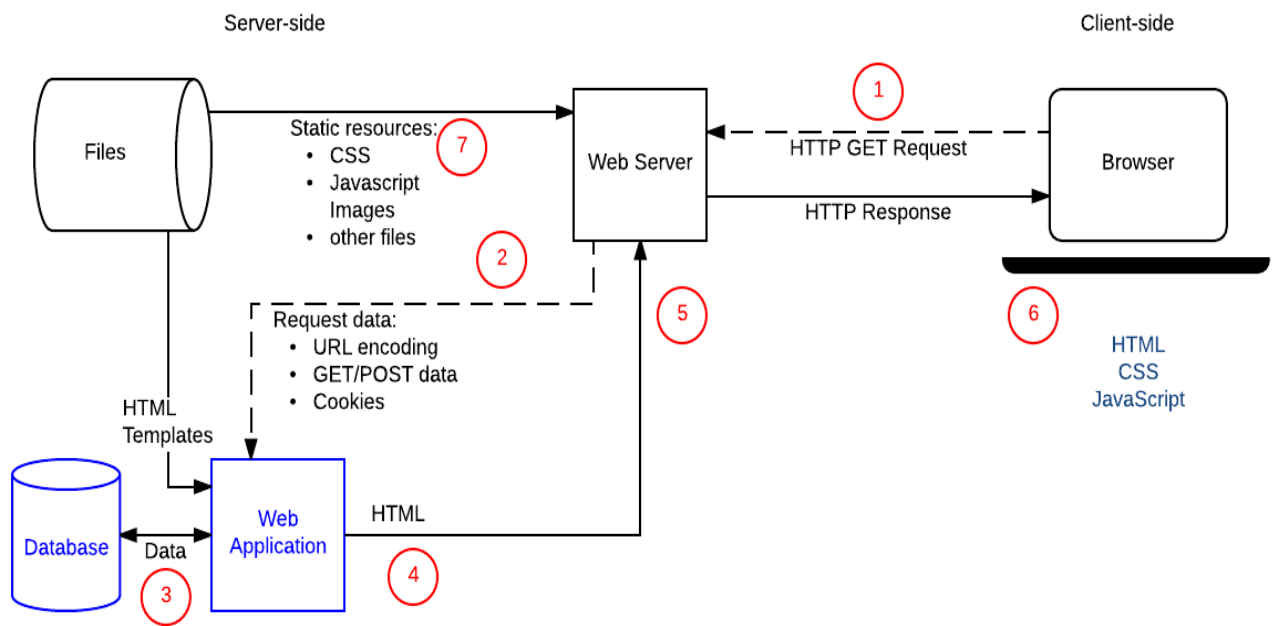


Рисунок 2.4 – Архітектура серверу для динамічного веб-сайту

Код серверної частини може бути написаний на будь-якій кількості мов програмування, приклади популярних мов серверної частини включають PHP, Python, Ruby, C# і NodeJS (JavaScript). Код серверної частини має повний доступ до операційної системи сервера, і розробник може вибрати, яку мову програмування він хотів би використати.

Розробники зазвичай пишуть свій код, використовуючи веб-фреймворки. Веб-фреймворки – це набори функцій, об'єктів, правил та інших конструкцій коду, призначених для вирішення загальних проблем, прискорення розробки та спрощення різних типів завдань, що стоять у конкретній галузі.

PHP на даний момент є найпопулярнішою серверною мовою. Її популярність зумовлена тим, що багато років поспіль для написання серверної частини сайту здебільшого використовувалася саме PHP. Проте останніми роками у нових проєктах її використання падає. Це пов'язано з досить складним синтаксисом та виходом і адаптацією простіших мов програмування.

Python – за останні кілька років його популярність різко зросла, і він навіть перевершив Java. З розвитком машинного навчання, аналізу даних і веб-додатків багато розробників стали частіше використовувати цю мову програмування, оскільки вона має безліч корисних бібліотек, простий синтаксис і мобільність.

NodeJS (JavaScript) – це мова, популярність якої за останні п'ять років зростає швидше ніж у інших мов, при тому, що С# і PHP втрачають позиції. [7] Поширеність JavaScript, якщо навіть не говорити виключно про Node, йде вгору.

Як можна пояснити те, що JavaScript у ролі серверної мови, був настільки швидко і широко прийнятий спільнотою розробників? Простіше кажучи, Node пережив стадію, в якій сприймався як якась гра, і увійшов у фазу стабільності та зрілості. Навколо нього сформувалося сильне співтовариство, розмір якого неухильно зростає. Екосистема Node також варта згадки, тому що, наприклад, менеджер пакетів Node, npm зараз представлений найбільшим реєстром ПЗ в інтернеті.

2.1.4 Бази даних

Бази даних для сайтів використовуються з метою зберігання різної інформації, і спрощено є деяким набором взаємопов'язаних таблиць [8]. Розміри таблиць у БД різні, їх кількість довільна. Саме в базах даних зберігається на сервері необхідна для роботи сайту інформація, наприклад, інформація про клієнтів, каталог товарів, статистичні дані тощо.

Найпопулярнішими БД в сфері веб є MySQL, PostgreSQL, MongoDB та інші.

MySQL – це система управління реляційними базами даних із відкритим вихідним кодом з моделлю клієнт-сервер. СУРБД – це програмне забезпечення, для управління реляційними базами даних.

- багатопоточність, підтримка кількох одночасних запитів;
- Оптимізація зв'язків із приєднанням багатьох даних за один прохід;
- Записи фіксованої та змінної довжини;
- Гнучка система привілеїв та паролів;
- Гнучка підтримка форматів чисел, рядків змінної довжини та міток часу.

MongoDB – система управління базами даних, яка працює з документоорієнтованою моделлю даних. На відміну від реляційних СУБД, MongoDB не потребує таблиці, схеми або окремої мови запитів. Інформація зберігається у вигляді документів чи колекцій [9].

					ДП.КН.22.468.01.000	Арк.
						25
Змн.	Арк.	№ докум.	Підпис	Дата		

Розробники позиціонують продукт як проміжну ланку між класичними СУБД та NoSQL. MongoDB не використовує схеми, як це роблять реляційні бази даних, що підвищує продуктивність усієї системи.

Основними перевагами є:

- відсутність схеми, ця база даних базується на колекціях різних документів. Кількість полів, зміст та розмір цих документів може відрізнятися. Тобто, різні сутності не повинні бути ідентичними за структурою;
- зрозуміла структура кожного об'єкта;
- легко масштабується;
- для зберігання даних, що використовуються в даний момент, використовується внутрішня пам'ять, що дозволяє отримати швидкий доступ;
- дані зберігаються у вигляді документів JSON;
- mongoDB підтримує динамічні запити документів (document-based query);
- немає необхідності мапінгу об'єктів програми в об'єкти БД.

Ще одна технологія, з якою ми постійно стикаємося, яка присутня у веб-додатках і програмному забезпеченні, так і на рівні процесора в наших комп'ютерах і смартфонах.

Cache – це концепція в розробці, коли дані, що часто використовуються, замість того, щоб їх щоразу діставати з БД, обчислювати або готувати іншим способом, зберігаються в швидко доступному місці.

У всіх стандартних браузерях включено кешування статичних файлів. Завдяки цьому, відкриваючи сайт не вперше все завантажується помітно швидше. Мінус для розробника в тому, що з увімкненим кешем не завжди відразу видно зміни, зроблені в коді.

2.2 Проєктування структури веб-сервісу

Структура сайту буде складатися з трьох основних елементів:

- header (навігація, кнопки та пошук);
- main (основна частина сайту з динамічним наповненням);

					ДП.КН.22.468.01.000	Арк.
						26
Змн.	Арк.	№ докум.	Підпис	Дата		

– footer (посилання на соцмережі, інформація про організацію та ін.).

Хедер та футер будуть відображатися на кожній сторінці та матимуть статичну інформацію. Мейн буде заповнюватися динамічно, залежно від сторінки на якій перебуває користувач, та від типу користувача (гість, зареєстрований, адміністратор).

Головна або "Криптовалюти" сторінка сайту міститиме маленьку панель новин, та таблицю з інформацією про криптовалюти.

Сторінка "Криптовалюти" міститиме таблицю з інформацією про криптовалюти.

Сторінка "Новини" міститиме список з новинами. Натиснувши на заголовок новини, користувач потрапить на сторінку з детальним описом новини.

Сторінка "Профіль" доступна авторизованим користувачам буде наповнена певним функціоналом в залежності від типу користувача.

Також форми реєстрації та авторизації, виклик яких доступний з будь-якої сторінки сайту. Авторизованим користувачам буде відкриватися новий функціонал сайту.

На рисунку 2.5 зображено діаграму використання сайту зі сторони користувача та адміністратора.

					ДП.КН.22.468.01.000	Арк.
						27
Змн.	Арк.	№ докум.	Підпис	Дата		

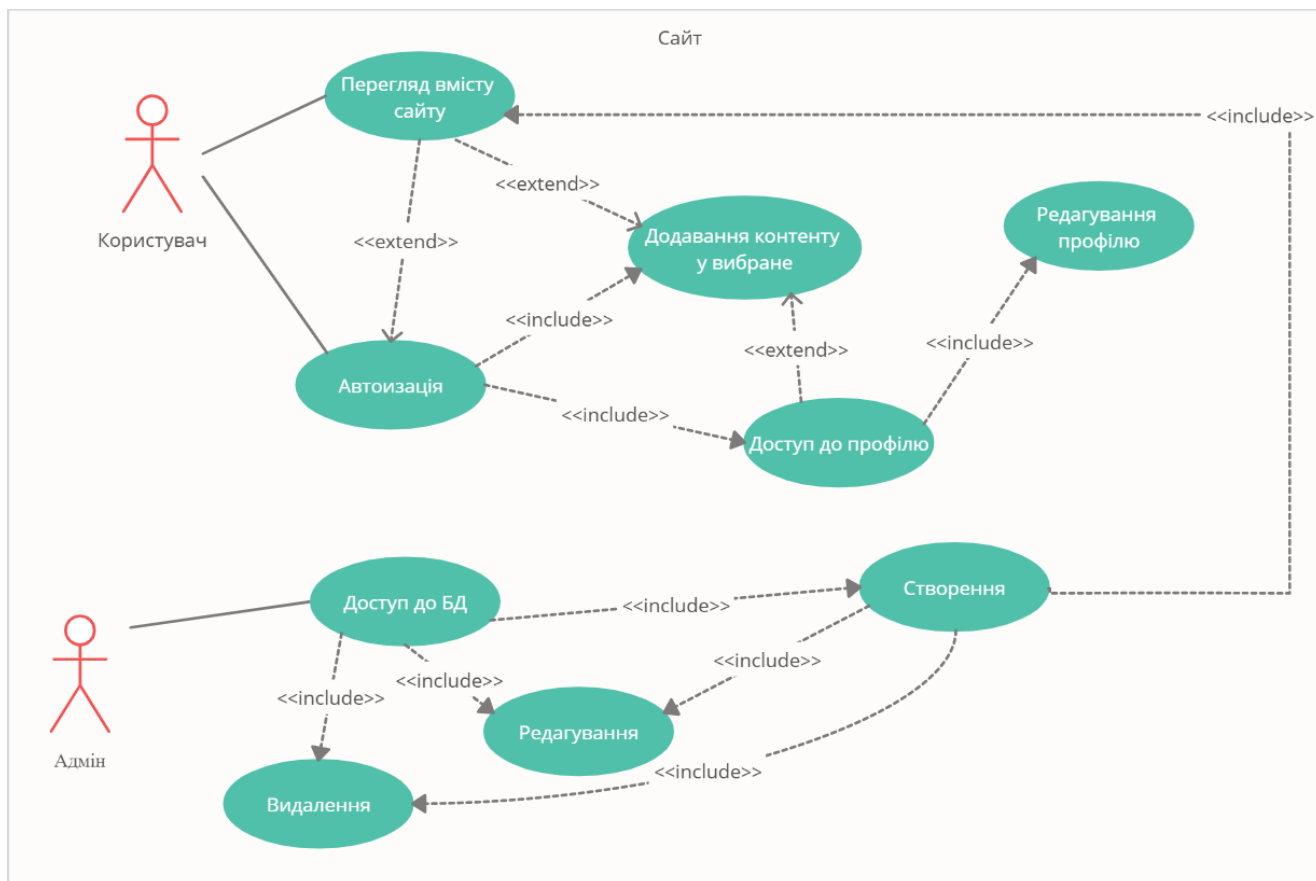


Рисунок 2.5 – Діаграма прецедентів

2.3 Проектування користувацького інтерфейсу сайту

Інтерфейс – це набір необхідних інструментів, для зручного керування застосунком.

Зручний інтерфейс є невід’ємною складовою будь-якого комп’ютерного додатку, сайти також не є виключенням.

Першочерговим завданням інтерфейсу є надання користувачеві усіх необхідних інструментів для навігації та керування сайтом. Проте не менш важливим критерієм веб-інтерфейсу є його лаконічність, практичність та інтуїтивність. Це особливо актуально для сайтів загального використання, де користувачу потрібно лише ознайомитися з інформацією. Головну сторінку слід формувати таким чином щоб вона містила якомога більше інформації, що може зацікавити потенційного гостя сайту.

Header		
Logo	navLink1 navLink2...	<input type="button" value="Login"/> <input type="button" value="Sign up"/> <input type="text" value="Search _____"/> <input type="button" value="Search"/>
Main	<div>Basic information</div>	
Footer		
information about the site		

Рисунок 2.6 – Макет сторінки сайту

На рисунку 2.6 зображено приблизний дизайн сторінки сайту. Усі навігаційні елементи винесено у верхню панель, для зручного доступу. Основний контент сторінки відображатиметься по середині сторінки з невеликими відступами по боках. У нижній частині сайту буде відображатися додаткова інформація.

3 РЕАЛІЗАЦІЯ ТА ТЕСТУВАННЯ ВЕБ-СЕРВІСУ

3.1 Вибір інструментів реалізації

Вище було описано про доступні інструменти для розробки, а тепер слід обрати їх для реалізації власного сайту.

Для розробки проєкту використовувалося середовище WebStorm.

Клієнтську частину було реалізовано за допомогою HTM, CSS(SASS) та JavaScript.

Оскільки одним лише JS хорошого сайту не напишеш, слід обрати якийсь фреймворк або бібліотеку [10]. Було обрано бібліотеку React.

React – це потужна сучасна бібліотека, яка розробляється та підтримується компанією Meta. З кожним роком все більше набирає популярності, а з цим і росту затребуваність в розробниках. Саме тому була вибрана ця бібліотека.

Основними перевагами є:

- велика спільнота розробників;
- доступні навчальні матеріали;
- велика кількість допоміжних інструментів;
- регулярні оновлення;
- визнання великими компаніями;
- універсальність та швидкість роботи;
- легкість сприйняття.

Також під час розробки було застосовано такі допоміжні бібліотеки:

- axios – бібліотека, яка дозволяє робити http запити з клієнтської частини на сервер;
- redux – бібліотека для управління станами;
- react-router-dom – бібліотека для реалізації навігації по сайту;
- інші дрібні інструменти.

Серверна частина реалізована платформою Node.js

					ДП.КН.22.468.01.000	Арк.
						30
Змн.	Арк.	№ докум.	Підпис	Дата		

Node.js не тільки здійснив революцію в серверній розробці, але завдяки йому зроблено внесок і в продуктивність клієнтських додатків, оскільки до розвитку V8 були залучені серйозні сили. Крім того, він відіграє помітну роль у розширенні всієї екосистеми JavaScript і у вдосконаленні сучасних JS-фреймворків, таких як Angular, React або Vue.

Перевагами Node є:

- оптимальна швидкість роботи програм;
- можливість спільного та повторного використання;
- безліч безкоштовних інструментів;
- кросплатформність;
- використання синтаксису JS;
- можливість застосовувати одну мову на клієнті та сервері;
- асинхронна робота.

Для пришвидшення та полегшення розробки застосовано такі допоміжні інструменти:

- JWT(JSON Web Token) – для отримання токена аутентифікації;
- mongoose – об'єктно-документний відображувач, для зв'язку між сервером та базою даних;
- express-validator – бібліотека для полегшення валідації;
- bcryptjs – інструмент для шифрування паролів;
- express – фреймворк для полегшення розробки на Node.js;
- cors – пакет для Node.js налаштування CORS(Cross-Origin Resource Sharing);
- nodemon – утиліта для автоматичного перезапуску серверу після змін в коді.

Базою даних слугуватиме документо-орієнтована система керування базами даних MongoDB.

Кілька причин чому саме MongoDB:

					ДП.КН.22.468.01.000	Арк.
						31
Змн.	Арк.	№ докум.	Підпис	Дата		

– гнучкість. MongoDB використовує документи, які можуть містити вкладені документи у складних ієрархіях, що робить його виразним та гнучким. MongoDB може відображати об'єкти з будь-якої мови програмування, забезпечуючи простоту впровадження та обслуговування.

– гнучка модель запиту. Користувач може вибірково індексувати деякі частини кожного документа або запиту на основі регулярних виразів, діапазонів або значень атрибутів і мати стільки властивостей кожного об'єкту, скільки необхідно для прикладного рівня.

– власна агрегація. Вбудована агрегація дозволяє користувачам вилучати та перетворювати дані з бази даних. Дані можна завантажити в новий формат, або експортувати в інші джерела даних.

– модель без схеми. Програми набувають повноважень та відповідальності за інтерпретацію різних властивостей документів колекції.

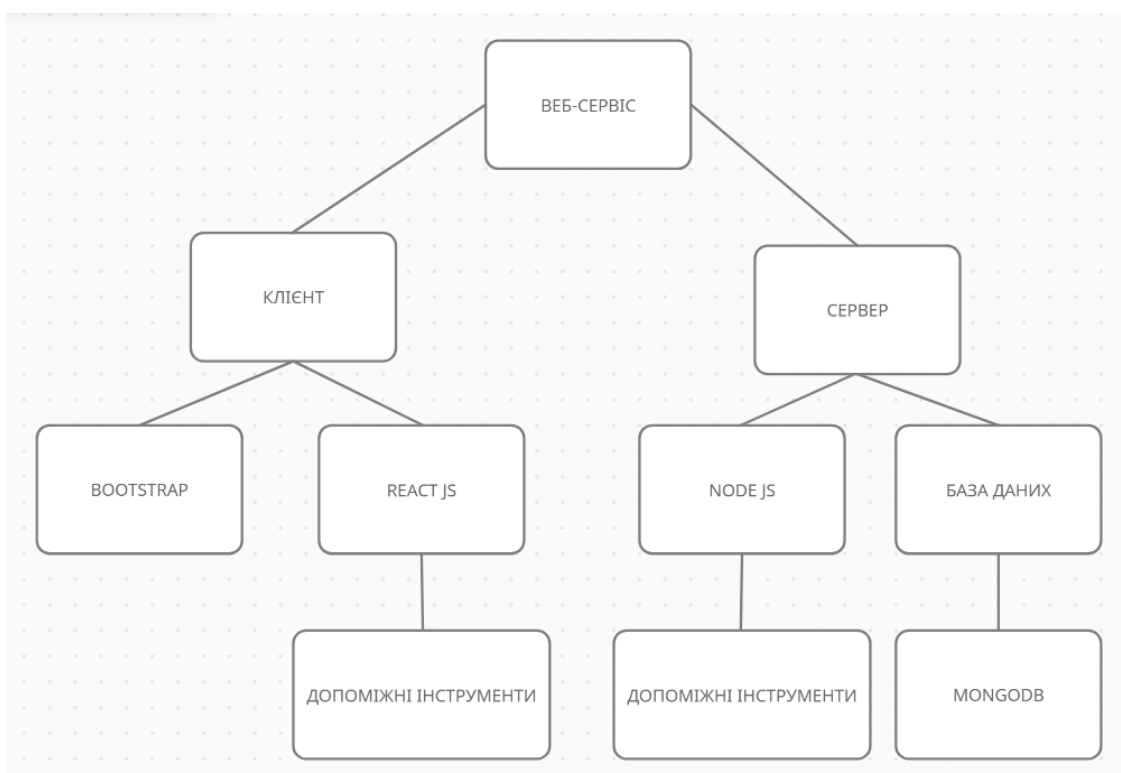


Рисунок 4.1 – Використаний інструментарій

На рисунку 4.1 подано графічно схему обраних інструментів. Після обрання інструментів можна переходити до розробки.

3.2 Створення клієнтської частини

Робота над дипломним проектом розпочинається з установки Node.js. Без установки цього середовища робота над проектом за допомогою React неможлива.

Для встановлення Node.js слід перейти на їхній офіційний сайт(nodejs.org) та вибрати версію для скачування (рисунок 4.2).

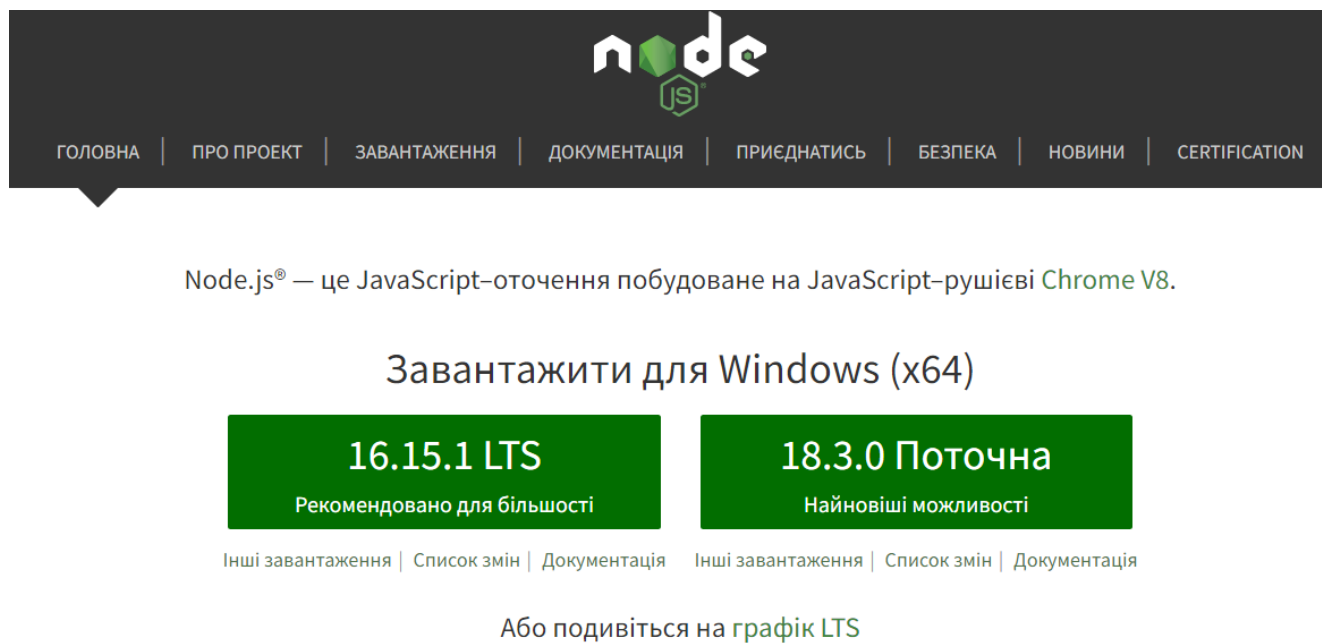


Рисунок 4.2 – Сайт для завантаження Node.js

Було обрано версію LTS по тій причині, що вона є більш стабільною та під неї налаштовано більше допоміжних інструментів.

Тепер можна ініціалізувати React проект. Це відбувається за допомогою команди "npx create-react-app", яку слід ввести в терміналі свого робочого середовища.

Після кількох хвилин очікування в проекті буде встановлено пакет з усіма необхідними компонентами (рисунок 3.3).

					ДП.КН.22.468.01.000	Арк.
						33
Змн.	Арк.	№ докум.	Підпис	Дата		

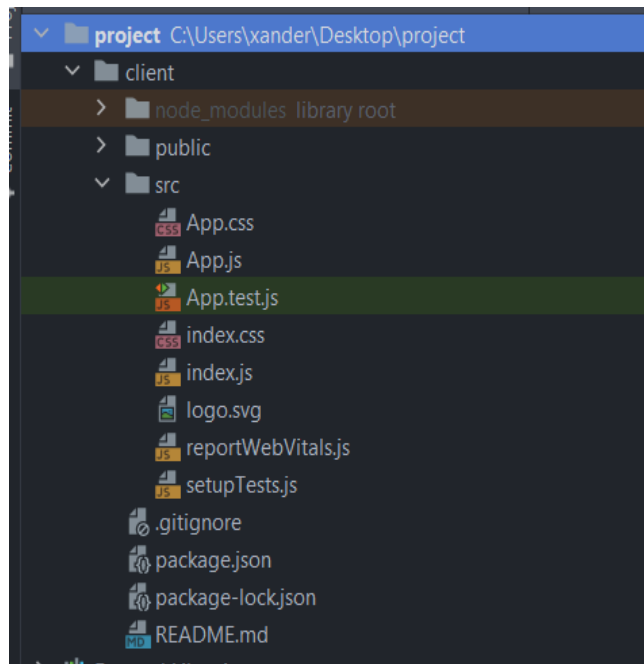


Рисунок 3.3 – Структура React проєкту

Для запуску проєкту у терміналі слід перейти до папки "client" (`cd client`) та запустити проєкт командою "`npm start`". Після виконаних інструкцій запуститься локальний сервер, а в браузері відкриється стандартний сайт проєкту (рисунок 3.4).

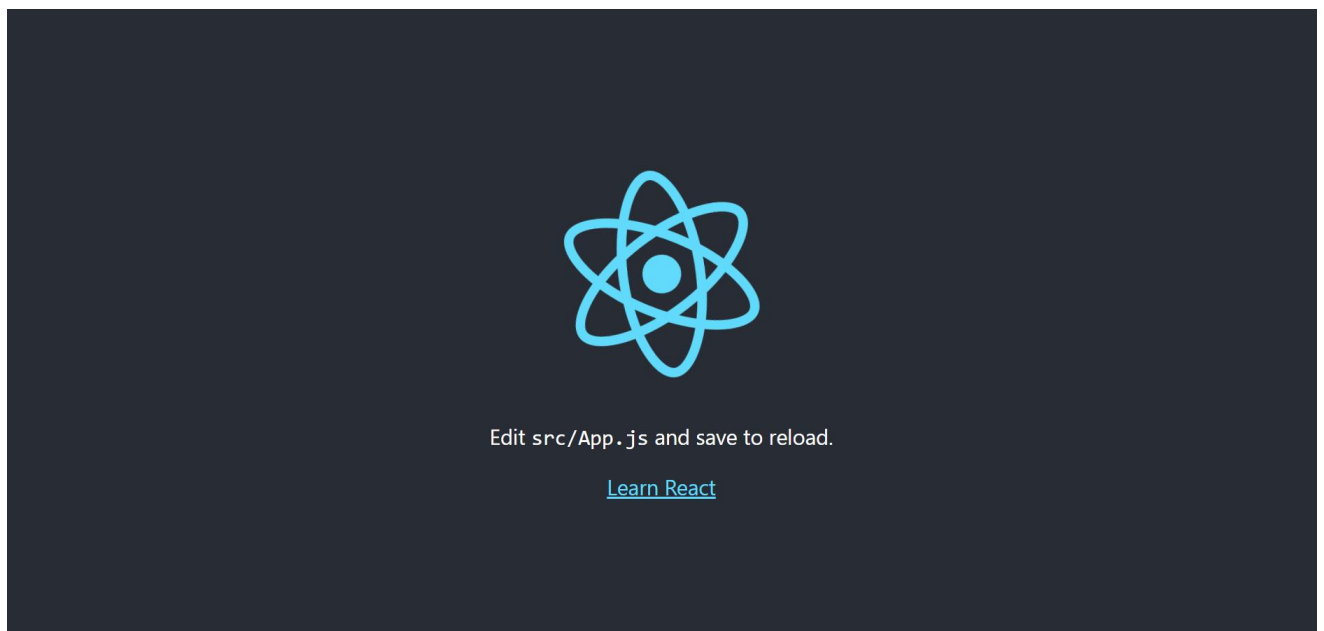


Рисунок 3.4 – Стандартний макет від React

Після чого видаляємо усі лишні файли та створюємо папку "components" – де відповідно будуть зберігатися компоненти до сайту (рисунок 3.5).

					ДП.КН.22.468.01.000	Арк.
						34
Змн.	Арк.	№ докум.	Підпис	Дата		

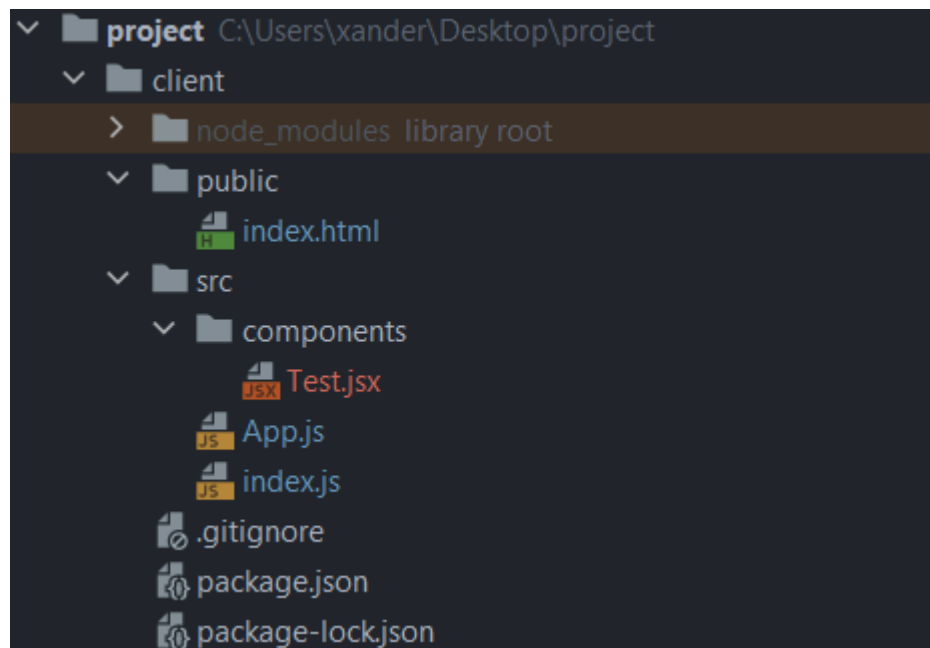


Рисунок 3.5 – Структура налаштованого проєкту

Заздалегідь варто встановити усі допоміжні бібліотеки, щоб потім не забути та не наробити помилок, в термінал вводимо відповідні команди (npm і axios, npm і redux, npm і react-router-dom). При використанні бібліотек їх слід імпортовувати в файл де вони використовуються.

На рисунку 3.6 зображено вигляд головної сторінки

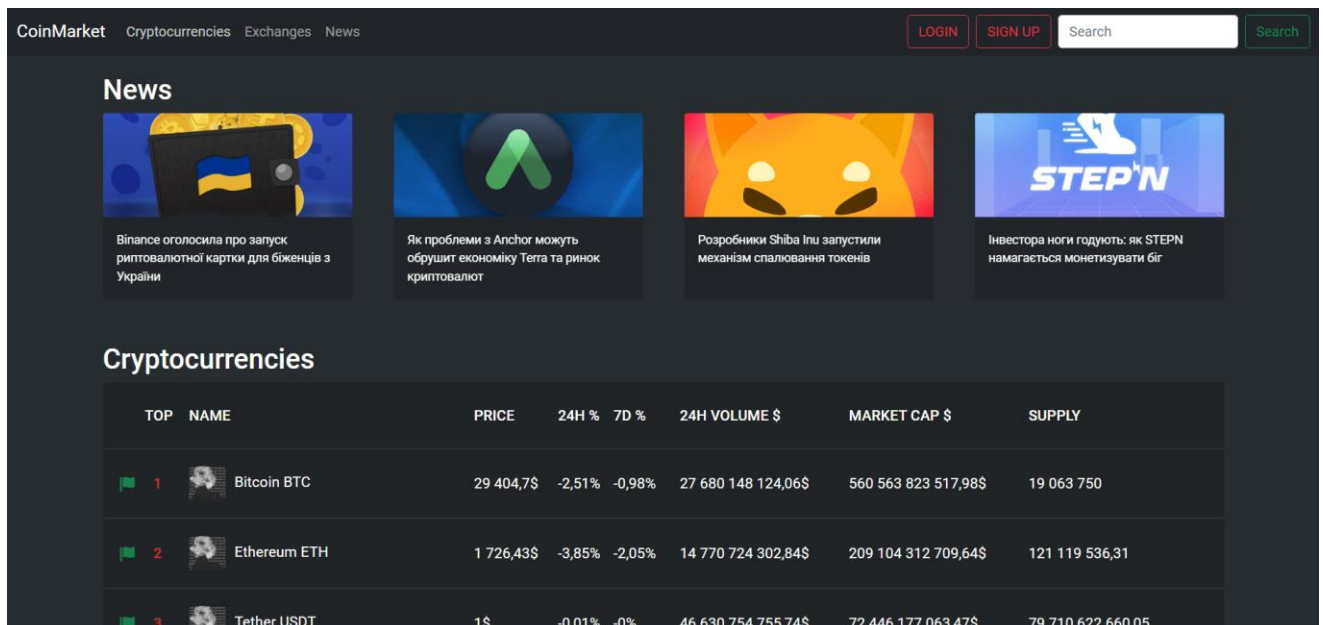
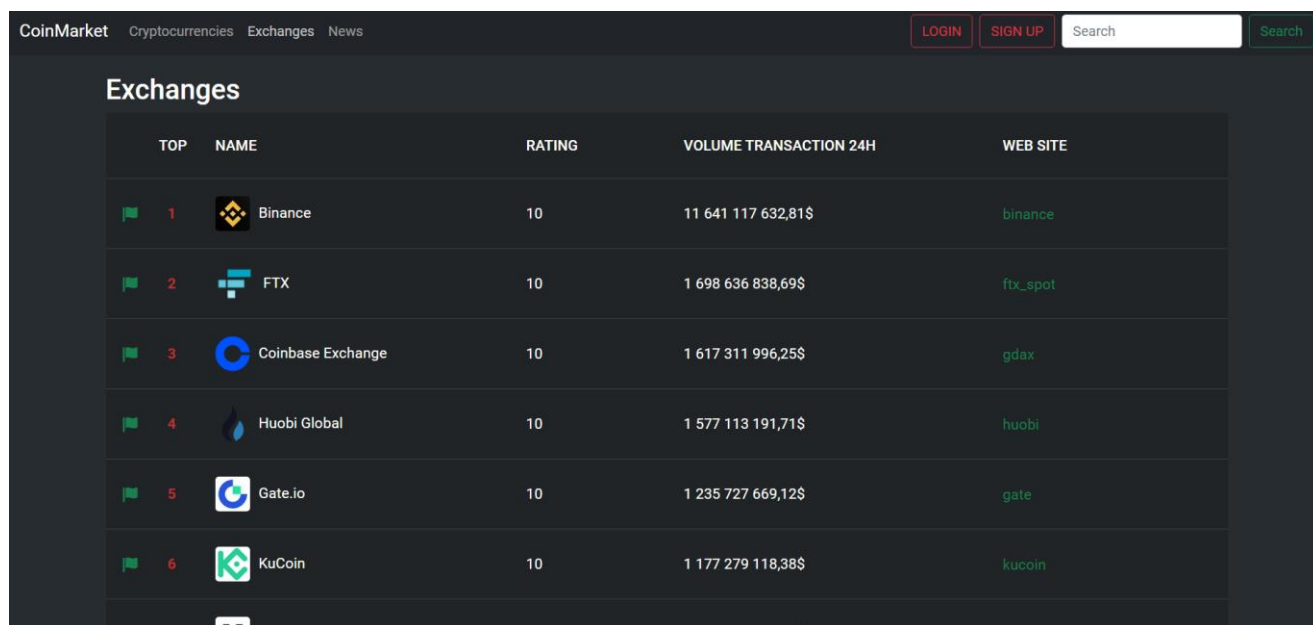


Рисунок 3.6 – Головна сторінка сайту

На головній сторінці будь-якого сайту розміщена найважливіша інформація. Це робиться з метою зацікавити користувача, що завітав на сайт.

На головній сторінці даного сайту розміщено навігаційну панель, декілька новин і таблицю з актуальною інформацією про криптовалюти. Дані в таблицю передаються по API Coin Market Cap та виводяться в порядку популярності.

Далі йде сторінка з криптобіржами (рисунок 3.7).



CoinMarket					Cryptocurrencies	Exchanges	News	LOGIN	SIGN UP	Search	Search
Exchanges											
TOP	NAME	RATING	VOLUME TRANSACTION 24H	WEB SITE							
1	Binance	10	11 641 117 632,81\$	binance							
2	FTX	10	1 698 636 838,69\$	ftx_spot							
3	Coinbase Exchange	10	1 617 311 996,25\$	gdax							
4	Huobi Global	10	1 577 113 191,71\$	huobi							
5	Gate.io	10	1 235 727 669,12\$	gate							
6	KuCoin	10	1 177 279 118,38\$	kucoin							

Рисунок 3.6 – Сторінка " Exchanges"

Тут виведена таблиця з найпопулярнішими крипто обмінниками. В таблиці показано місце в топі, назва, рейтинг, щоденний грошовий обіг та посилання на сам обмінник. Дані було взято з API сайту CoinGecko.

Далі йде сторінка з новинами (рисунок 3.7). На цій сторінці відображаються карточки з новинами у вигляді списку. На карточні присутній час створення, фото, заголовок та короткий опис.

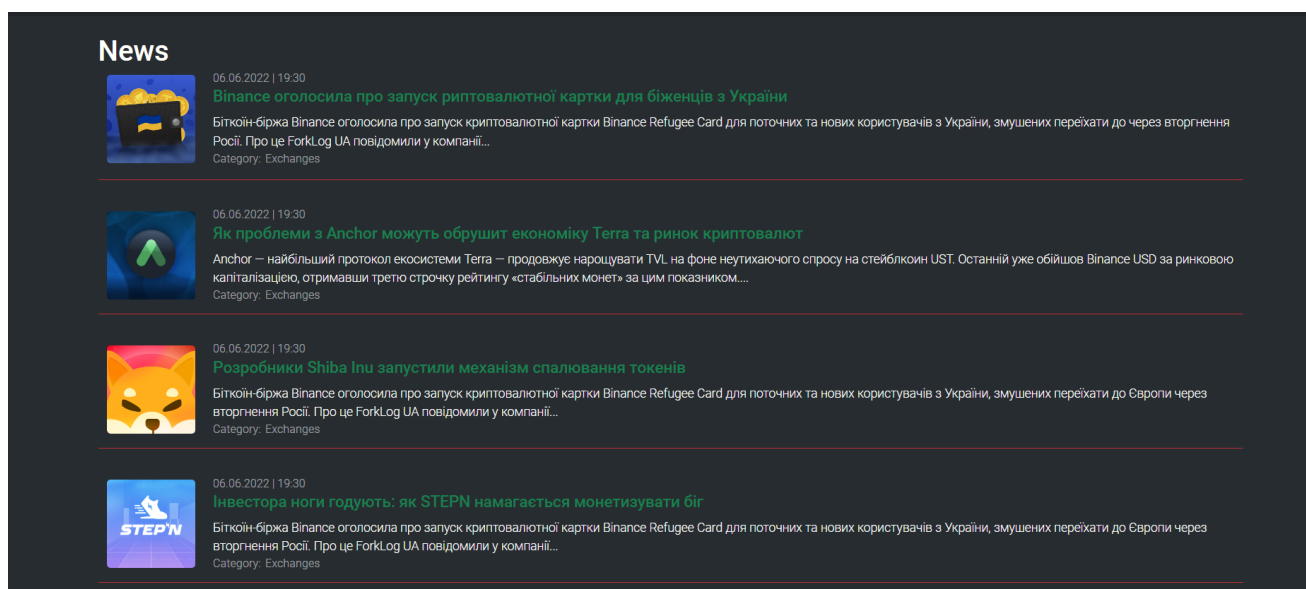


Рисунок 3.7 – Сторінка " News"

Далі сторінка з детальним описом новини (рисунок 3.8).

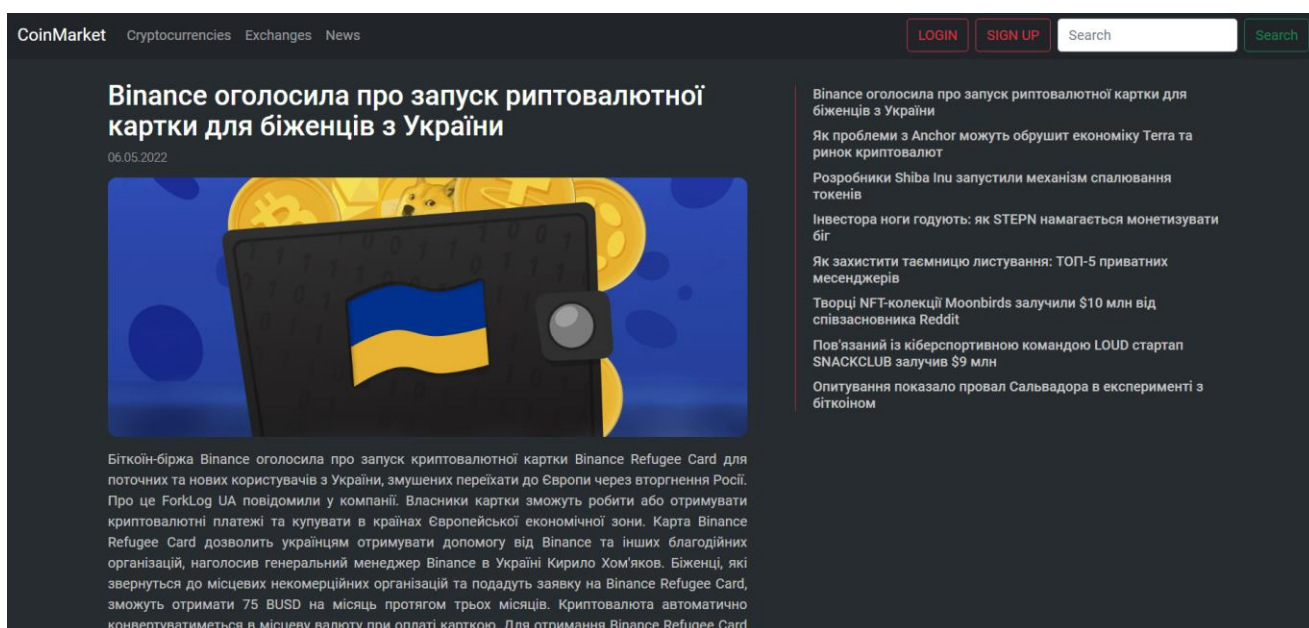


Рисунок 3.8 – Сторінка " News-page"

На сторінці відображено заголовок, фото та детальний опис новини. З правого боку є колонка з заголовками останніх новин, клацнувши по котрому відкриється така ж сама сторінка, але з даними про іншу новину.

На цьому всі сторінки доступні не авторизованому користувачу закінчуються, щоб отримати доступ до іншого контенту слід зареєструватися та

					ДП.КН.22.468.01.000	Арк.
						37
Змн.	Арк.	№ докум.	Підпис	Дата		

авторизуватися. Форма реєстрації (рисунок 3.9) та форма авторизації (рисунок 3.10).

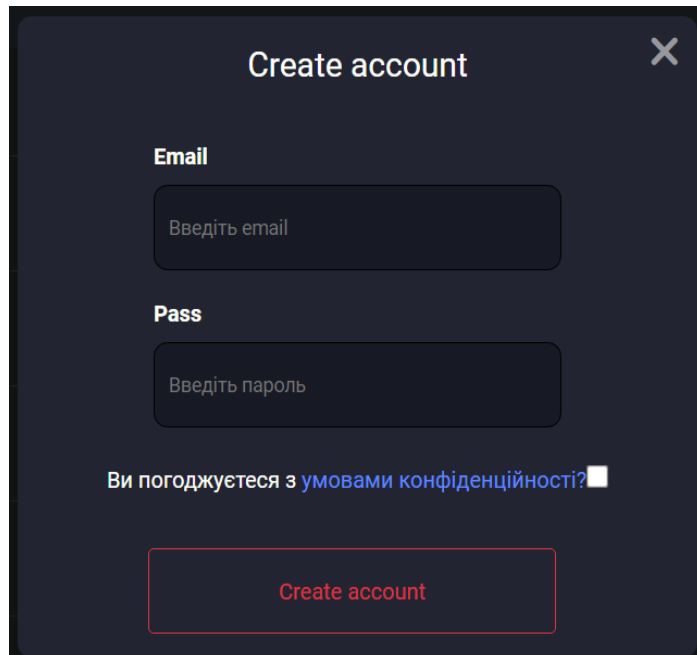
A dark-themed modal window titled "Create account" with a close button (X) in the top right corner. It contains two input fields: "Email" with placeholder text "Введіть email" and "Pass" with placeholder text "Введіть пароль". Below the fields is a checkbox labeled "Ви погоджуєтесь з умовами конфіденційності?". At the bottom is a red-outlined button labeled "Create account".

Рисунок 3.9 – Форма реєстрації

Форма реєстрації відправляє дані на сервер, а він в свою чергу в базу даних, там вони зберігаються та використовуються при авторизації.

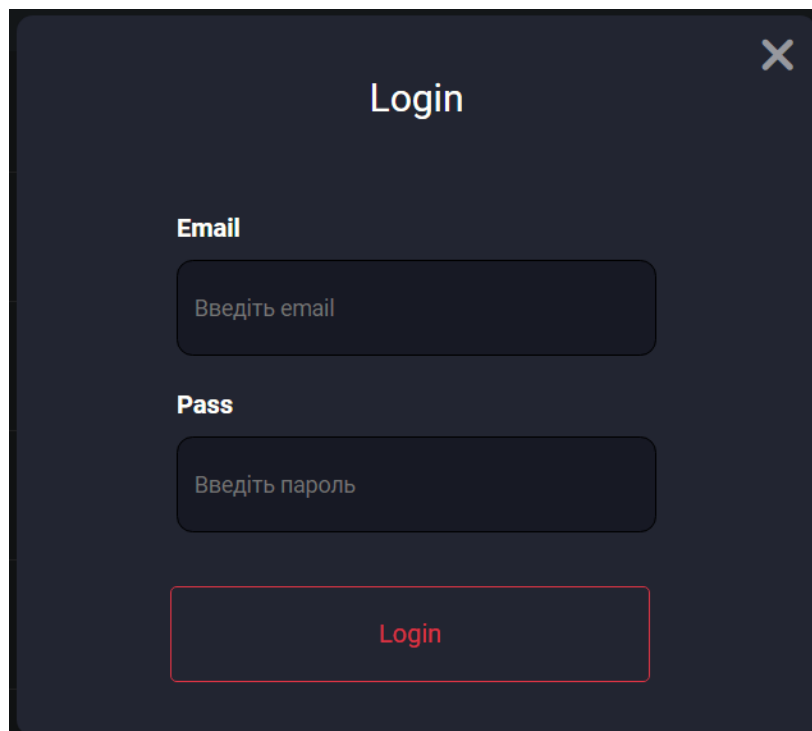
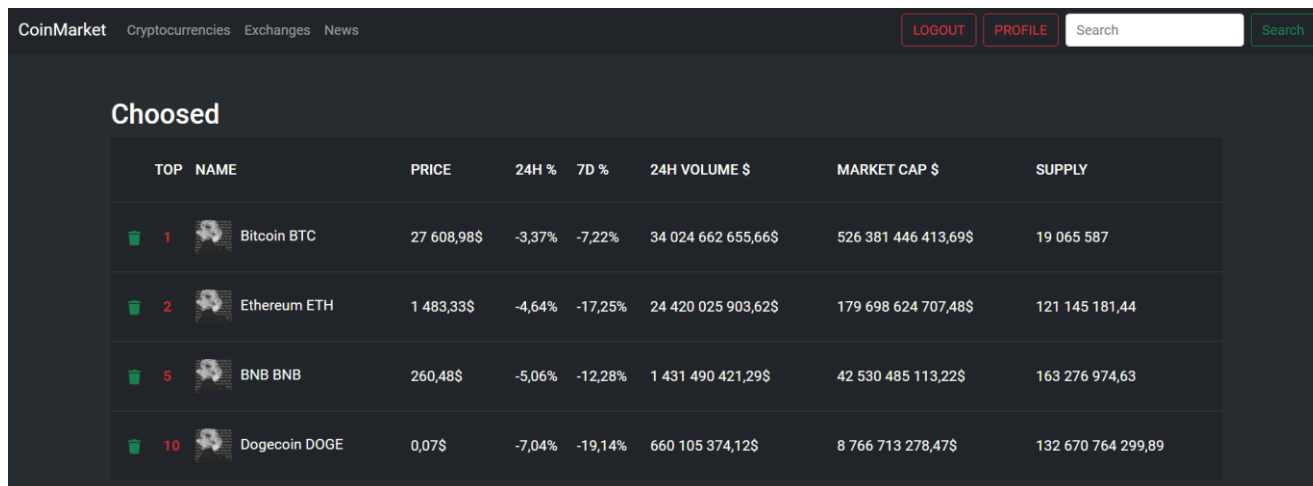
A dark-themed modal window titled "Login" with a close button (X) in the top right corner. It contains two input fields: "Email" with placeholder text "Введіть email" and "Pass" with placeholder text "Введіть пароль". At the bottom is a red-outlined button labeled "Login".

Рисунок 3.10 – Форма авторизації

					ДП.КН.22.468.01.000	Арк.
						38
Змн.	Арк.	№ докум.	Підпис	Дата		

На формах присутні по два поля для пошти та пароля, а також кнопка для відправки запитів до бази даних. Форми викликаються відповідними кнопками, що розташовані в навігаційній панелі.

Після авторизації користувачу надається доступ до профіля (рисунок 3.11).



CoinMarket Cryptocurrencies Exchanges News LOGOUT PROFILE Search

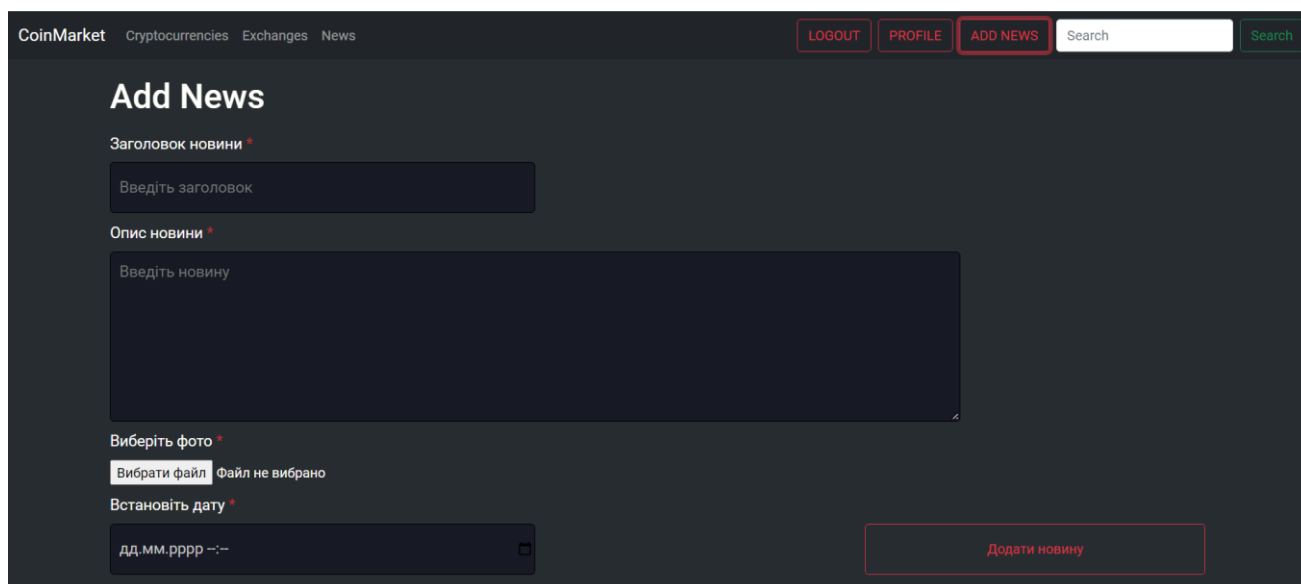
Chooosed

TOP	NAME	PRICE	24H %	7D %	24H VOLUME \$	MARKET CAP \$	SUPPLY
1	Bitcoin BTC	27 608,98\$	-3,37%	-7,22%	34 024 662 655,66\$	526 381 446 413,69\$	19 065 587
2	Ethereum ETH	1 483,33\$	-4,64%	-17,25%	24 420 025 903,62\$	179 698 624 707,48\$	121 145 181,44
5	BNB BNB	260,48\$	-5,06%	-12,28%	1 431 490 421,29\$	42 530 485 113,22\$	163 276 974,63
10	Dogecoin DOGE	0,07\$	-7,04%	-19,14%	660 105 374,12\$	8 766 713 278,47\$	132 670 764 299,89

Рисунок 3.11 – Профіль

У профілі можна переглядати вибрану криптовалюту. Щоб додати її в профіль, на сторінці "Exchanges" слід натиснути на прапорець і вибрана монета збережеться в профілі, доки користувач її не видалить. Усі ці функції доступні лише авторизованим користувачам.

Адміністратору ж доступна сторінка для публікування новин (рисунок 3.12).



CoinMarket Cryptocurrencies Exchanges News LOGOUT PROFILE ADD NEWS Search

Add News

Заголовок новини *

Введіть заголовок

Опис новини *

Введіть новину

Виберіть фото *

Вибрати файл Файл не вибрано

Встановіть дату *

дд.мм.рррр --

Додати новину

Рисунок 3.12 – Сторінка адміністратора

Тут адміністратор може постити новини. Є заголовок, опис, фото та дата публікації. Після натискання кнопки дані з форми вносяться до бази даних.

Також адміністратор може видаляти уже опубліковані новини, перейшовши на вкладку з зі списком новин (рисунок 3.13).

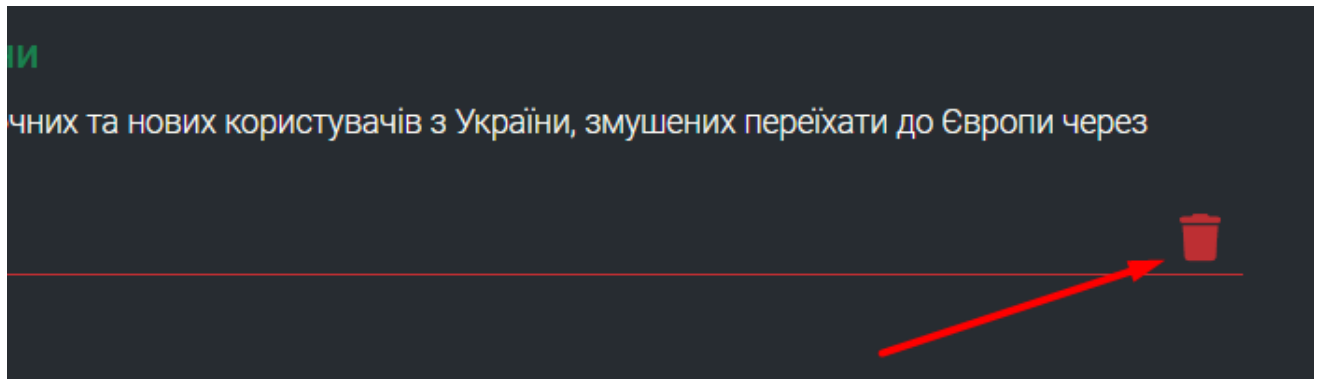


Рисунок 3.13 – Видалення новини

3.3 Створення бази даних

Далі слід створити базу даних для збереження в ній інформації з сайту. Для цього слід перейти на сайт "MondoDB", авторизуватися та створити проєкт (рисунок 1.14).

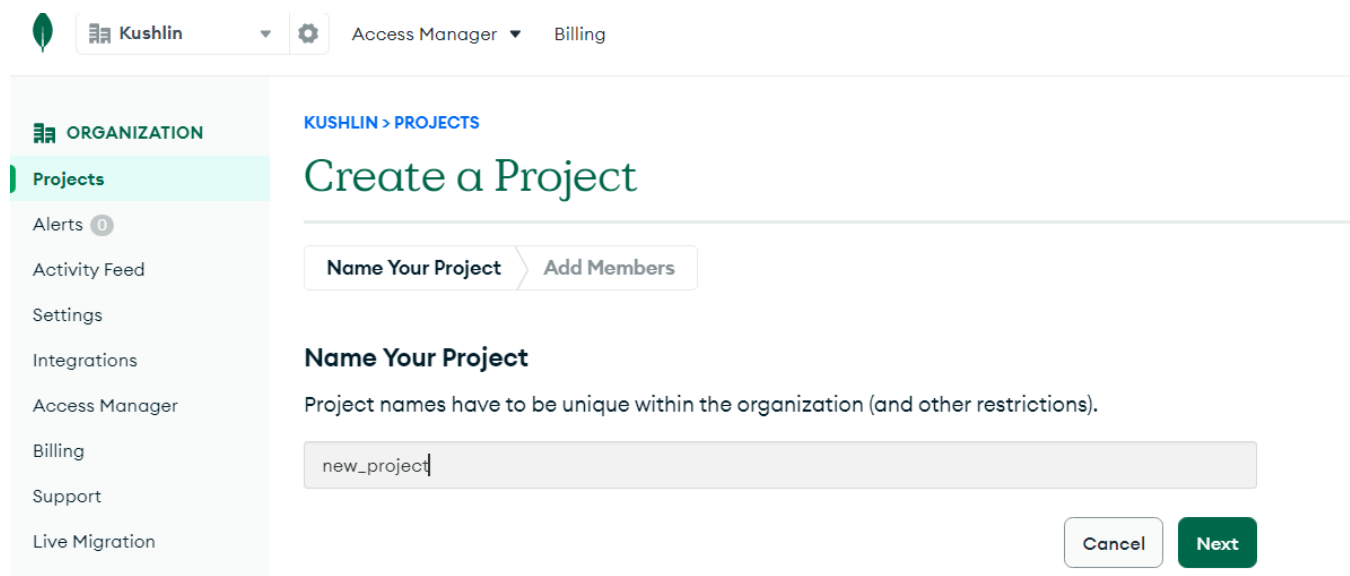


Рисунок 3.14 – Створення проєкту в MondoDB

В проєкті створюємо та розгортаємо хмарну базу даних, після чого вибираємо сервер ближче до місця проживання (рисунок 3.15).

					ДП.КН.22.468.01.000	Арк.
						40
Змн.	Арк.	№ докум.	Підпис	Дата		

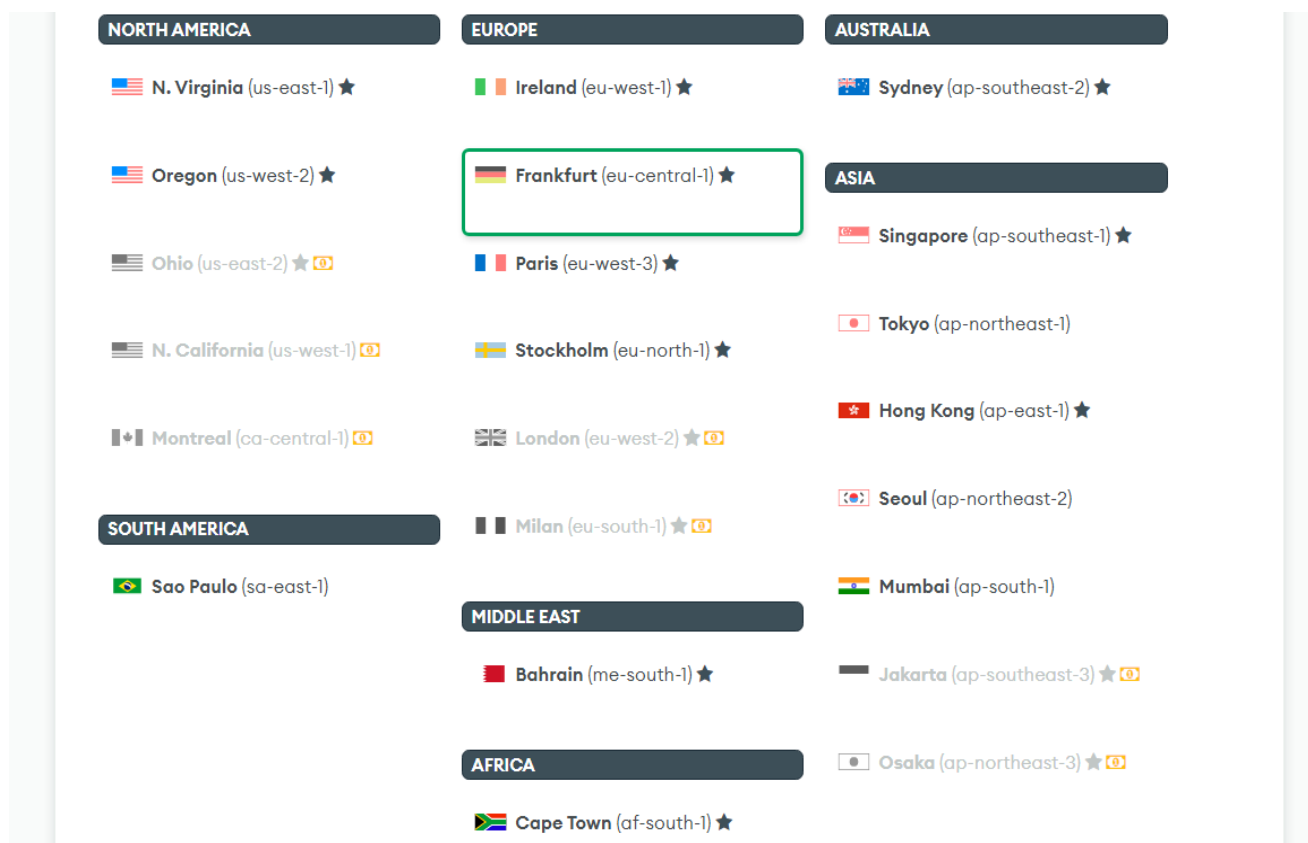


Рисунок 3.15 – Вибір сервера

Через деякий час БД захоститься, і після того слід додати користувача та переходити до підключення в свій проєкт.

3.4 Створення серверної частини

В нашому проєкті створюємо папку з назвою "server" та ініціалізуємо її командою "npm init -y". Після цього в папці буде створено файл "package.json", він призначений для налаштування проєкту.

Тепер слід встановити допоміжні бібліотеки та інструменти, робиться це командою "npm i (назва бібліотеки)". На рисунку 3.16 можна побачити усі використані інструменти.

```

"dependencies": {
  "axios": "^0.27.2",
  "bcryptjs": "^2.4.3",
  "body-parser": "^1.20.0",
  "cors": "^2.8.5",
  "express": "^4.18.1",
  "express-validator": "^6.14.1",
  "jsonwebtoken": "^8.5.1",
  "mongoose": "^5.13.14",
  "nodemon": "^2.0.16"
}

```

Рисунок 3.16 – Бібліотеки які були використані в проєкті

Далі проводимо дрібні налаштування та приєднуємося до раніше створеної бази даних. Для цього переходимо до БД та натискаємо "Connect" (рисунок 3.17).

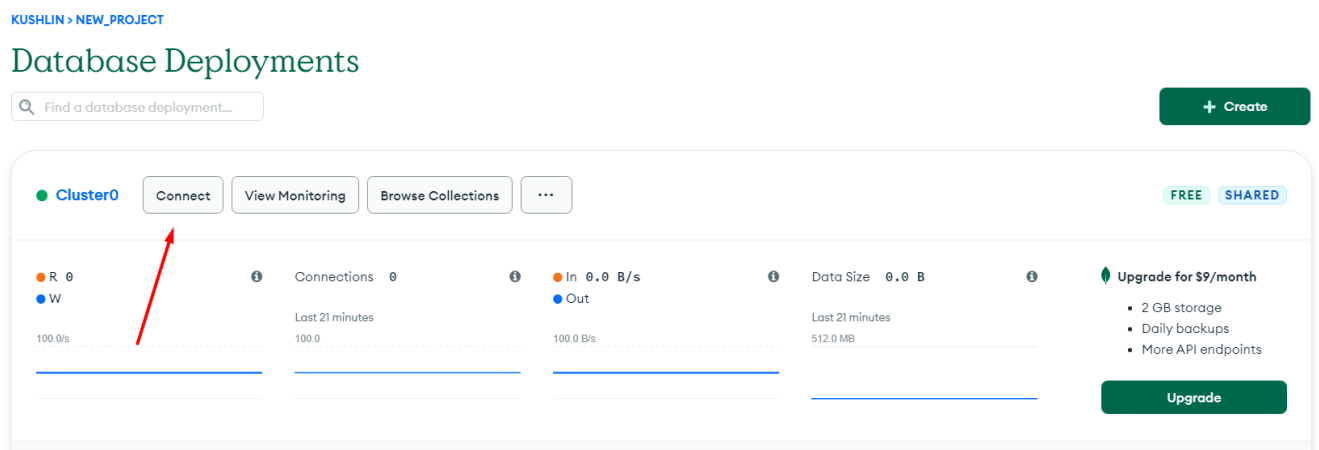


Рисунок 3.17 – Підключення до БД

Проводимо потрібні налаштування та отримуємо посилання на БД, яку потрібно вставити в серверну частину проєкту (рисунок 3.18) та прописати пароль і назву БД у відведеному місці.

```

mongodb+srv://admin:<password>@cluster0.mhvc6kt.mongodb.net/?
retryWrites=true&w=majority

```

Рисунок 3.18 – Згенероване посилання для підключення БД

Запуск серверу виконуємо командою "npm start". Якщо все добре, то в терміналі вилізе сповіщення про вдалий запуск серверу.

					ДП.КН.22.468.01.000	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		42

Тепер можна переходити до написання логіки сайту.

3.5 Тестування функціоналу сайту

Що ж перейдемо до тестування, усі основні функції сайту доступні для авторизованих користувачів, тому для початку перевіримо реєстрацію. Для реєстрації слід натиснути кнопку "SIGN UP", що знаходиться на навігаційній панелі. У відкритому модальному вікні прописуємо свою пошту, придумуємо пароль та натискаємо створити акаунт (рисунок 3.19) після чого дані відправляються на сервер, пошта як є, а пароль зберігається захешованим.

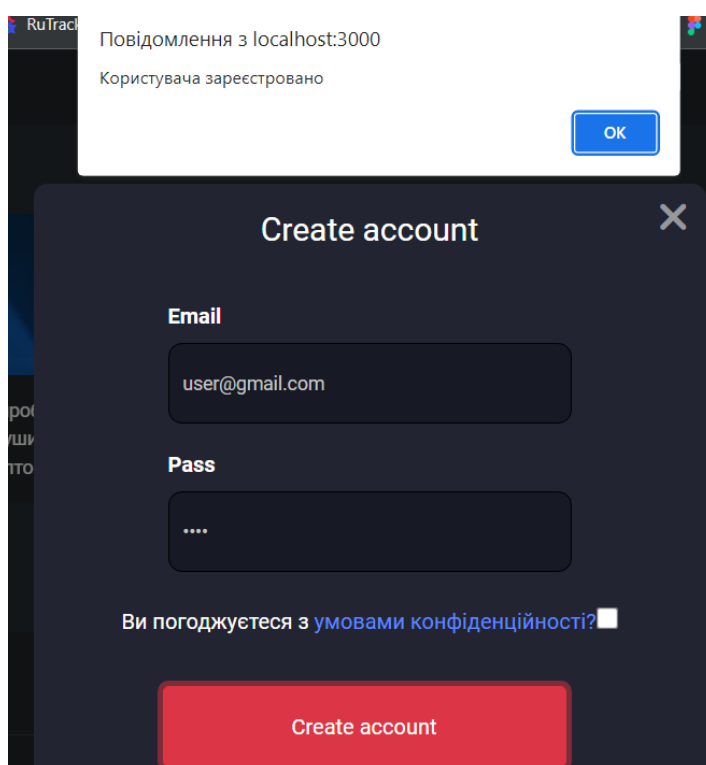


Рисунок 3.19 – Сповіщення про створення акаунту

Тепер переходимо до форми авторизації та вводимо дані щойно створеного користувача (рисунок 3.20).

					ДП.КН.22.468.01.000	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		43

A dark-themed login modal window titled "Login" with a close button (X) in the top right corner. It contains two input fields: "Email" with the text "user@gmail.com" and "Pass" with masked characters "...". Below the fields is a red-outlined button labeled "Login".

Рисунок 3.20 – Вхід в акаунт

Якщо авторизація пройшла успішно, то кнопки входу та реєстрації заміняться (рисунок 3.21).



Рисунок 3.21 – Кнопки якщо користувач авторизувався

Тепер перейдемо в профіль (рисунок 3.22).

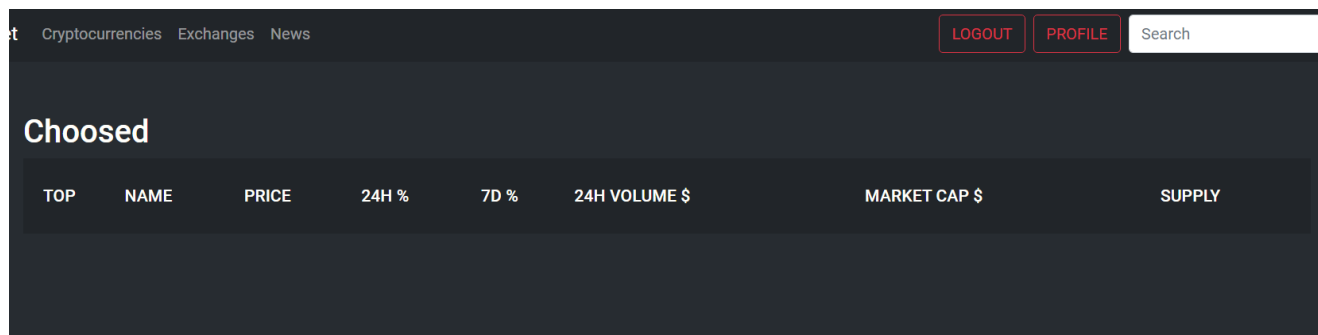


Рисунок 3.22 – Профіль користувача

Як бачимо поки, що тут пусто, для того, щоб внести сюди інформацію слід перейти на головну сторінку та вибрати декілька криптовалют (рисунок 3.23).

					ДП.КН.22.468.01.000	Арк.
						44
Змн.	Арк.	№ докум.	Підпис	Дата		

Cryptocurrencies							
TOP	NAME	PRICE	24H %	7D %	24H VOLUME \$	MARKET CAP \$	SUPPLY
1	Bitcoin BTC	28 122,53\$	-1,45%	-6,48%	31 338 487 198,45\$	536 174 359 175,83\$	19 065 650
2	Ethereum ETH	1 520,61\$	-1,07%	-16,39%	22 894 042 255,71\$	184 216 362 392,05\$	121 146 206,75
3	Tether USDT	1\$	-0,01%	-0,03%	55 152 958 855,83\$	72 423 623 882,81\$	79 710 622 658,24
4	USD Coin USDC	1\$	0,03%	0,02%	5 831 564 636,09\$	53 963 760 390,08\$	53 939 541 406,21
5	BNB BNB	267,49\$	-1,24%	-10,71%	1 347 877 167,02\$	43 674 313 605,38\$	163 276 974,63
6	Cardano ADA	0,53\$	-4,65%	-7%	1 501 322 054,93\$	18 027 711 714,54\$	34 277 702 081,6

Рисунок 3.22 – Тестування додавання валют в профіль

Тепер переходимо знову до профілю та перевіряємо (рисунок 3.23).

Chooosed							
TOP	NAME	PRICE	24H %	7D %	24H VOLUME \$	MARKET CAP \$	SUPPLY
1	Bitcoin BTC	28 121,02\$	-1,31%	-6,48%	31 324 311 463,83\$	536 145 474 327,17\$	19 065 650
2	Ethereum ETH	1 520,08\$	-1,05%	-16,53%	22 870 556 997,08\$	184 151 955 136,11\$	121 146 206,75
6	Cardano ADA	0,53\$	-4,5%	-6,91%	1 499 959 414,99\$	18 032 868 411,5\$	34 277 702 081,6

Рисунок 3.23 – Валюти в профілі

Перевіримо функцію видалення (рисунок 3.24), для цього слід натиснути кнопку для видалення.

Chooosed							
TOP	NAME	PRICE	24H %	7D %	24H VOLUME \$	MARKET CAP \$	SUPPLY
1	Bitcoin BTC	28 132,14\$	-1,26%	-6,43%	31 348 344 158,27\$	536 357 552 552,25\$	19 065 650
6	Cardano ADA	0,53\$	-4,28%	-6,81%	1 501 494 284,92\$	18 044 812 421,17\$	34 277 702 081,6

Рисунок 3.24 – Результат видалення

					ДП.КН.22.468.01.000		Арк.
							45
Змн.	Арк.	№ докум.	Підпис	Дата			

Як бачимо видалення пройшло успішно. Можна переходити до тестування функції адміністратора. Для цього виходимо з акаунту та знову логінимось, але цього разу під акаунтом адміністратора.

Адміністратор має ті ж самі можливості, що й користувач, плюс додавання та видалення новин. Натискаємо кнопку "ADD NEWS" (рисунок 3.25) та переходимо на форму написання новин.

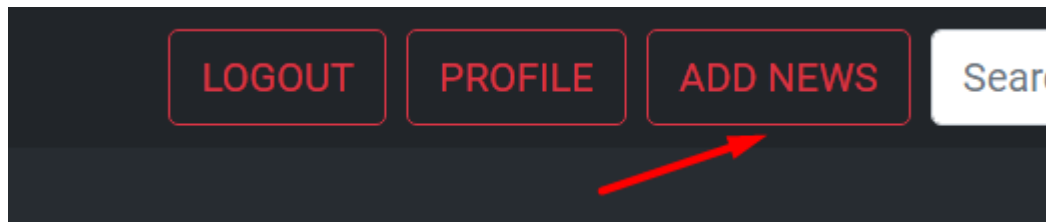


Рисунок 3.26 – Додаткова кнопка у адміністратора

Заповнюємо форму та відправляємо дані на сервер (рисунок 3.26).

Рисунок 3.26 – Результат відправки

Тепер перейдемо на сторінку з новинами та впевнимися, що новина додалася (рисунок 3.27).

					ДП.КН.22.468.01.000	Арк.
						46
Змн.	Арк.	№ докум.	Підпис	Дата		

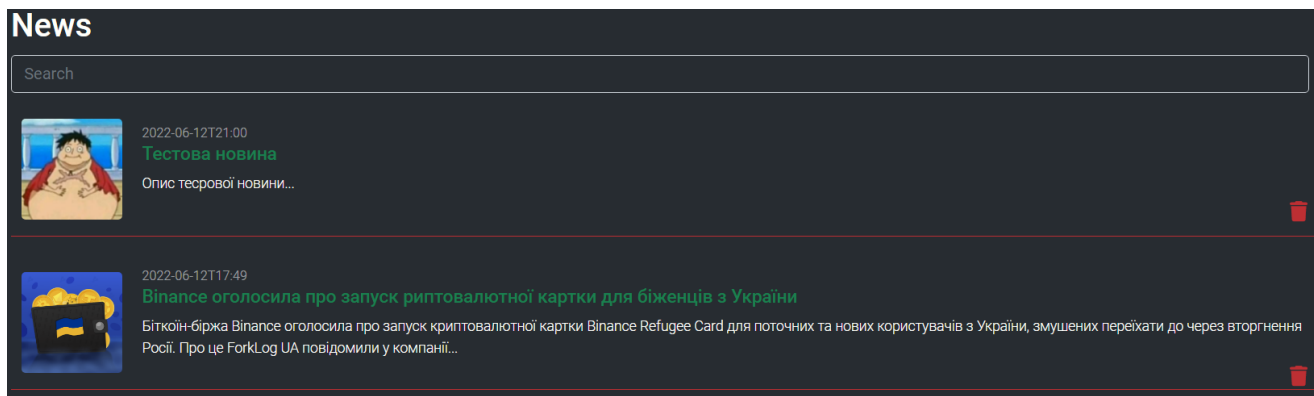


Рисунок 3.27 – Додана новина

Тепер натиснемо на видалення та поглянемо на результат (рисунок 3.28).

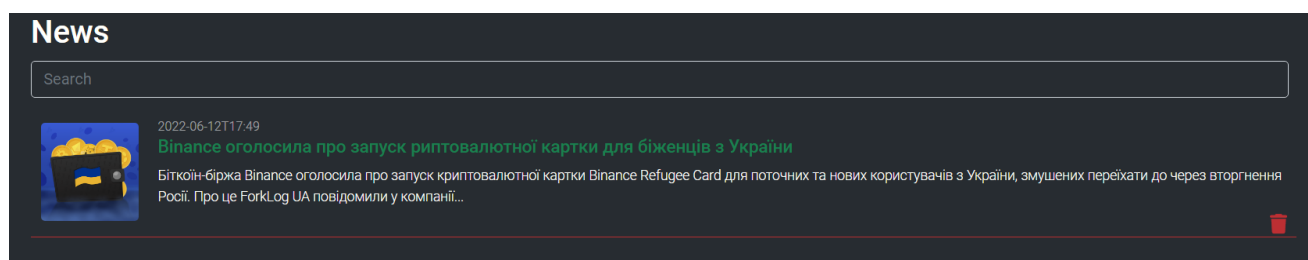


Рисунок 3.28 – Результат видалення

Також на сайті присутня система пошуку. Перейдемо на головну сторінку та проведемо пошук (рисунок 3.29).

Cryptocurrencies							
bitcoin							
TOP	NAME	PRICE	24H %	7D %	24H VOLUME \$	MARKET CAP \$	SUPPLY
1	Bitcoin BTC	27 672,24\$	-3,54%	-7,68%	31 181 087 064,89\$	527 590 254 840,76\$	19 065 687
12	Wrapped Bitcoin WBTC	27 648,51\$	-3,58%	-7,81%	441 624 774,98\$	7 560 521 790,15\$	273 451,35
26	Bitcoin Cash BCH	152,95\$	-5,06%	-15,97%	1 944 067 398,1\$	2 919 619 015,18\$	19 088 706,25
50	Bitcoin SV BSV	50,15\$	-4,66%	-11,53%	120 734 022,26\$	957 180 738,62\$	19 084 882,89
96	Bitcoin Gold BTG	18,21\$	-5,86%	-17,23%	8 125 645,9\$	318 917 415,01\$	17 513 923,59

Рисунок 3.29 – Результат пошуку

					ДП.КН.22.468.01.000	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		47

Як можна побачити пошук добре відпрацьовує, система пошуку реалізована на кожній сторінці де вона є потрібною. І оскільки сайт побудований за технологією "SPA (single page application)" при переході між сторінками перезавантаження не відбувається, принцип такої роботи показано на рисунку 3.30.

SPA Lifecycle

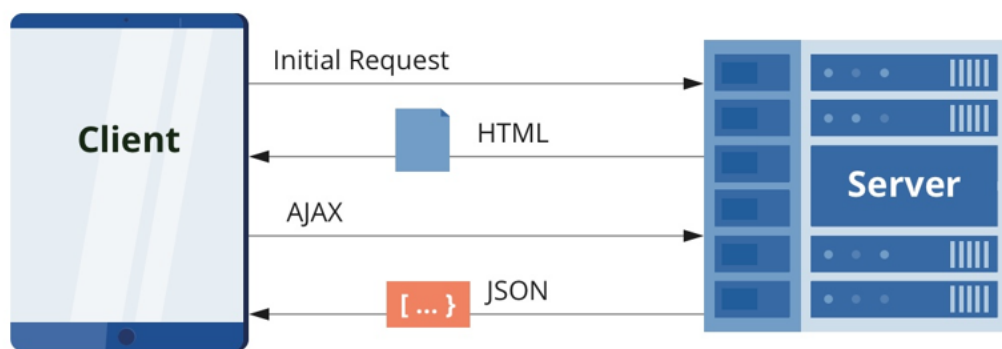


Рисунок 3.30 – Принцип роботи технології SPA

4 ТЕХНІКО-ЕКОНОМІЧНЕ ОБҐРУНТУВАННЯ

4.1 Аналіз ринку

Проаналізувавши найпопулярніші подібні сервіси можна зробити висновки, що найбільш затребуваним функціоналом є можливість відстежувати криптовалюту та криптобіржі, а також переглядати актуальні новини у цій сфері. Адже криптовалюти не сама стабільна річ на землі, іноді їх курс залежить від висловлювань про них якихось відомих людей, тут і стане в пригоді сторінка з новинами. Також поширеною є функція "вибране", вона дозволяє зберігати в профілі монети, які цікавлять конкретного користувача.

Сучасний ринок потребує швидких сервісів, які не перевантажені лишньою інформацією, а також не менш важливим є зовнішній вигляд сайту: його кольорова схема, розміщення на ньому інформації, можливість використання на телефоні.

Потенційними користувачами сайту є як звичайні користувачі, що просто цікавляться криптовалютою так і користувачі з досвідом трейдингу, для них особливо важливим є вчасне та постійне оновлення інформації.

Реалізований й ж сервіс надає користувачам наступний функціонал:

- доступ до перегляду інформації;
- реєстрація;
- авторизація;
- функція вибраного в профілі.

Для ведення сайту потрібен працівник, котрий буде шукати та вносити новини на сайт, а також потрібен той хто слідкує за роботоспроможністю сайту.

4.2 Розрахунок витрат на проектування

Мінімальна заробітна плата в Україні становить 6500 гривень в місяць, тому зарплата працівників не може бути меншою.

По даних з сайту "Work ua" середня заробітна плата Front-end розробника становить 30000 гривень.

					ДП.КН.22.468.01.000	Арк.
						49
Змн.	Арк.	№ докум.	Підпис	Дата		

За даними цього ж сайту середня зарплата за посадою «Backend розробник» в Україні є 45500 гривень.

Середня зарплата за посадою "Full stack developer" складає 48000 гривень

Середня зарплата тестувальника становить 22000 гривень.

Середня зарплата веб-дизайнера становить 25000 гривень.

Проаналізувавши ринок, на біржі фрілансерів було найнято 3 співробітника (веб-дизайнер, full stack розробник, тестувальник).

Консультуючись з замовником за тиждень роботи дизайнером був зроблений макет сторінки.

Веб-розробник працював над проєктом близько одного місяця, його задачею була реалізація макету, який надав дизайнер, написання серверної частини та хостинг сайту.

Тестувальник включався на певних етапах роботи. Його основним завданням була перевірка веб-сайту на помилки. При знаходженні помилки про неї повідомляли розробника, а він в свою чергу займався її виправленням та доопрацюванням проєкту.

Зарплата розробників даного веб сервісу подана в таблиці 4.1

Таблиця 4.1 – заробітна плата працівників

№	Посада	Зарплата (грн)	Кількість співробітників	Загалом(грн)
1	Веб-дизайнер	10000	1	10000
2	Full stack розробник	40000	1	40000
3	Тестувальник	15000	1	15000
Всього				65000

Заробітна плата нараховувалась згідно з аналізом ринку та виконаною роботою.

					ДП.КН.22.468.01.000	Арк.
						50
Змн.	Арк.	№ докум.	Підпис	Дата		

У випадку якщо робітників найняв підприємець або підприємство податки на дохід оплачуються ними, 18% – ПДФО (податок на доходи фізичних осіб) та 1.5% – військовий збір.

$\text{ПДФО} = (18\% \text{ з } 10000) = 1800 \text{ грн.}$

$\text{Військовий збір} = (1.5\% \text{ з } 10000) = 150 \text{ грн.}$

Разом 1950 грн.

$\text{ПДФО} = (18\% \text{ з } 40000) = 7200 \text{ грн.}$

$\text{Військовий збір} = (1.5\% \text{ з } 40000) = 600 \text{ грн.}$

Разом 7800 грн.

$\text{ПДФО} = (18\% \text{ з } 15000) = 2700 \text{ грн.}$

$\text{Військовий збір} = (1.5\% \text{ з } 15000) = 225 \text{ грн.}$

Разом 2925 грн.

Всього податків 12675.

Таким чином підприємство повинне сплатити 77675 грн (65000 зарплата + 12675 податків).

У випадку, якщо замовником є фізична особа податок платить фрілансер, 18% – ПДФО (податок на доходи фізичних осіб) та 1.5% – військовий збір.

Дизайнер сплатить 1950 грн податків, а його дохід складе 8050 грн (10000 - 1950).

Програміст сплатить 7800 грн податків, а його дохід складе 32200 грн (40000 - 7800).

Тестувальник сплатить 2925 грн податків, а його дохід складе 12075 грн (15000 - 2925).

4.3 Обґрунтування необхідності розробки

Хоча криптовалюти з'явилися не так давно, вони доволі швидко розвиваються, та все більше людей дізнаються про них. Майже кожен чув слово криптовалюта або біткоїн, хоча і не зовсім знають, що це таке. Однією з причин створення сайту якраз і є дати змогу зацікавленим ознайомитися з цими технологіями.

					ДП.КН.22.468.01.000	Арк.
						51
Змн.	Арк.	№ докум.	Підпис	Дата		

Основним функціоналом є показ криптобірж відсортованих по рейтингу всіх користувачів з метою запобігання обману нових користувачів котрі не знають де можна безпечно, швидко та вигідно торгувати.

Таким чином, веб-сервіс заохочує нових людей до входу у крипто сферу, та значно зменшує поріг входу для новачків. Для більш досвідчених у цій сфері людей, сайт надає актуальні новини з крипто сфери, які є досить важливими для прогнозування подальшого розвитку сфери. Прикладом є заява Китаю про заборону транзакцій пов'язаних з криптовалютами. У зв'язку з цією новиною курс за один день впав на 6 відсотків. Отож, знання подібних новин допомагають захистити свій капітал.

					ДП.КН.22.468.01.000	Арк.
						52
Змн.	Арк.	№ докум.	Підпис	Дата		

ВИСНОВКИ

В результаті роботи, ми отримали повністю готовий до використання веб-сервіс для моніторингу криптовалют. З допомогою якого можна легко дізнатися актуальний курс валюти, які зараз в тренді, популярні і рейтингові криптобіржі та новини, які стосуються крипто сфери.

Сайт створено за допомогою сучасних інструментів та технологій веб-розробки:

- Postman;
- HTML;
- CSS;
- React.js;
- Node.js;
- MongoDB.

Ці інструменти чудово підходять для використання початківцями, оскільки по них є багато навчальних матеріалів, а самі мови легкі в вивченні тим хто має хоч якийсь досвід в програмуванні.

Дизайн сайту виконано згідно з початковим планом та дотриманням усіх основних критеріїв щодо розміщення контенту та хорошої кольорової схеми.

Під час виконання завдання було отримано цінний досвід в сфері веб-програмування та досить детально вивчено крипто індустрію. В проєкті були застосовані знання здобуті протягом навчання та освоєно нові технології в процесі виконання.

В подальшому планується поступова модернізація веб-сервісу(розширення прав адміністратора, збільшення функціоналу для користувача, поліпшення інтерфейсу).

					ДП.КН.22.468.01.000	Арк.
						53
Змн.	Арк.	№ докум.	Підпис	Дата		

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Посібник для початківців. HTML: веб-сайт URL: <https://developer.mozilla.org/ru/docs/Web/HTML> (дата звернення: 05.05.2022).
2. Документація та самовчителі. CSS: веб-сайт URL: <https://developer.mozilla.org/ru/docs/Web/CSS> (дата звернення: 05.05.2022).
3. Introduction. Bootstrap 5: веб-сайт URL: <https://getbootstrap.com/docs/5.0/getting-started/introduction/> (дата звернення: 20.05.2022).
4. Documentation. Sass: веб-сайт URL: <https://sass-lang.com/documentation/> (дата звернення: 25.05.2022).
5. Навчальні матеріали. JavaScript: веб-сайт URL: <https://developer.mozilla.org/ru/docs/Web/JavaScript> (дата звернення: 14.05.2022).
6. Початок роботи. React JS: веб-сайт URL: <https://uk.reactjs.org/docs/getting-started.html> (дата звернення: 20.05.2022).
7. Node.js v17.9.1 documentation. Node.js: веб-сайт URL: <https://nodejs.org/docs/latest-v17.x/api/> (дата звернення: 15.05.2022).
8. Основи баз даних І.О. Завадський. Основи баз даних: веб-сайт URL: http://www.svpu-profi.lg.ua/pdf/library/osn_baz_danih2011.pdf (дата звернення: 21.05.2022).
9. Welcome to the MongoDB Documentation. MongoDB: веб-сайт URL: <https://www.mongodb.com/docs/> (дата звернення: 16.05.2022).
10. Основи веб-розробки. Веб-розробка: веб-сайт URL: <https://sites.google.com/view/inf10/дистанційне-навчання/brainbasket-foundation/основи-веб-розробки> (дата звернення: 01.05.2022).

ДОДАТКИ

Додаток А

Лістинг програмного коду

Файл user.js функція запису на авторизацію. Код:

```
export const login = (username, password, setActive) => {

  return async dispatch => {
    try {

      const response = await
        axios.post(`https://kravchuncoin.herokuapp.com/auth/login`, {
          username,
          password
        })

      dispatch(setUser(response.data.user))
      localStorage.setItem('token', response.data.token)
      console.log(response.data)
      localStorage.setItem('id', response.data.user._id)
      console.log(response.data.user._id)
      localStorage.setItem('coinId',
        JSON.stringify(response.data.user.selectedCoins))

      console.log(JSON.stringify(response.data.user.selectedCoins))
      localStorage.setItem('role',
        JSON.stringify(response.data.user.roles))
      console.log(JSON.stringify(response.data.user.roles))
      setActive(false)
    } catch (e) {
```

					ДП.КН.22.468.01.000		
Змн.	Арк.	№ докум.	Підпис	Дата	САЙТ МОНІТОИНГУ РИНКУ КРИПТОВАЛЮТ		
Розроб.	Кравчун О.М.						
Перевір.	Чубей О.О.						
Реценз.	Кузик В.М.						
Н. Контр.	Сиротюк Н.С.						
Затверд.	Чубей О.О.						
					Лім.	Арк.	Акрушіє
						5	63
					ГФК.ВКВТ.ЦКІКД КН-41		


```

console.log(e)
      alert(e.response.data.message)

    }
  }
}

```

Файл coinData.js. Код:

```

import axios from "axios";
import {setCoin, setExchange} from
"../reducers/cryptocurrencyReducer";

export const parseCoins = () => {

  return async dispatch => {

    try {
      const response = await
axios.get(`https://kravchuncoin.herokuapp.com/crypto/getcoins`)

      dispatch(setCoin(response.data))
    } catch (e) {

      console.log(e)
      alert(e)

    }
  }
}

export const parseExchange = () => {

  return async dispatch => {
    try {

```

					ДП.КН.22.468.01.000	Арк.
						56
Змн.	Арк.	№ докум.	Підпис	Дата		

```

        const response = await
axios.get(`https://kravchuncoin.herokuapp.com/crypto/getexchange`)
        dispatch(setExchange(response.data))
    } catch (e) {
        console.log(e)
        alert(e)
    }
}
}
}

```

Файл authController.js. Код:

```

const User = require('./models/User')
const News = require('./models/News')
const Role = require('./models/Role')
const bcrypt = require('bcryptjs');
const jwt = require('jsonwebtoken');
const {validationResult} = require('express-validator')
const {secret} = require("./config")

const generateAccessToken = (id, roles) => {
    const payload = {
        id,
        roles
    }
    return jwt.sign(payload, secret, {expiresIn: "24h"})
}

class authController {
    async registration(req, res) {
        console.log(req.body)
        try {

```

					ДП.КН.22.468.01.000	Арк.
						57
Змн.	Арк.	№ докум.	Підпис	Дата		

```

const errors = validationResult(req)
if (!errors.isEmpty()) {
    return res.status(400).json({message: "Помилка при
реєстрації", errors})
}
const {username, password} = req.body;
const candidate = await User.findOne({username})
if (candidate) {
    return res.status(400).json({message: "Ім'я вже
заняте"})
}
const hashPassword = bcrypt.hashSync(password, 7);
const userRole = await Role.findOne({value: "USER"})
const user = new User({username, password: hashPassword,
roles: [userRole.value]})
await user.save()
return res.json({message: "Користувача зареєстровано"})
} catch (e) {
    console.log(e)
    res.status(400).json({message: 'Registration error'})
}
}

async login(req, res) {
    try {
        const {username, password} = req.body
        const user = await User.findOne({username})
        if (!user) {
            return res.status(400).json({message: `Користувача
${username} не існує`})
        }
        const validPassword = bcrypt.compareSync(password,
user.password)
        if (!validPassword) {
            return res.status(400).json({message: `Неправильний

```

					ДП.КН.22.468.01.000	Арк.
						58
Змн.	Арк.	№ докум.	Підпис	Дата		

```

    пароль`}))
        }
        const token = generateAccessToken(user._id, user.roles)
        return res.json({token, user})
    } catch (e) {
        console.log(e)
        res.status(400).json({message: 'Login error'})
    }
}

async getUsers(req, res) {
    try {
        console.log(req)
        const users = await User.find()
        res.json(users)
    } catch (e) {
        console.log(e)
    }
}

async getNews(req, res) {
    debugger
    try {

        const news = await News.find()
        res.json(news)
    } catch (e) {
        console.log(e)
    }
}

async addNews(req, res) {
    debugger
    try {
        const {title, description, date, img

```

					ДП.КН.22.468.01.000	Арк.
						59
Змн.	Арк.	№ докум.	Підпис	Дата		

```

        } = req.body;
        const news = new News({title, description, date, img})
        await news.save()

        res.json({message: 'Додано'})
        alert("csacs")

    } catch (e) {
        debugger
        console.log(e)
    }

}

async addToCoins(req, res) {
    try {
        const {id} = req.params;
        const {userId} = req.body
        const user = await User.findById(userId)
        await user.addToCoins(id)
        console.log(req.user)

        // res.json({message: 'Додано'})

    } catch (e) {
        console.log(e)
    }

}

async delCoins(req, res) {
    try {
        const {id} = req.params;

```

					ДП.КН.22.468.01.000	Арк.
						60
Змн.	Арк.	№ докум.	Підпис	Дата		

```

        const {userId} = req.body
        const user = await User.findById(userId)
        await user.delToCoins(id)
        console.log(req.user)
        // res.json({message: 'Видалено'})

    } catch (e) {
        console.log(e)
    }
}

async delNews(req, res) {
    try {
        const {id} = req.params;
        await News.findByIdAndDelete(id)
        // res.json({res})
        res.json({message: 'Видалено'})

    } catch (e) {
        console.log(e)
    }
}

async getCoinsToProfile(req, res) {
    try {
        const {id} = req.params;
        const {userId} = req.body
        const user = await User.findById(userId)
        await user.addToCoins(id)
        console.log(req.user)
    }
}

```

					ДП.КН.22.468.01.000	Арк.
						61
Змн.	Арк.	№ докум.	Підпис	Дата		

```

    } catch (e) {
        console.log(e)
    }

}

async getNew(req, res) {
    try {
        const {id} = req.params;
        const post = await News.findById(id)
        // res.json({res})

        res.json(post)
    } catch (e) {
        console.log(e)
    }
}
}

```

```
module.exports = new authController()
```

Файл cryptoController.js. Код:

```

const axios = require('axios')

class cryptoController {
    async getCoin(req, res) {
        let response = null;
        new Promise(async (resolve, reject) => {
            try {
                response = await axios.get('https://pro-
api.coinmarketcap.com/v1/cryptocurrency/listings/latest', {
                    headers: {
                        'X-CMC_PRO_API_KEY': 'KEY',
                    },
                },
            )
        })
    }
}

```

					ДП.КН.22.468.01.000	Арк.
						62
Змн.	Арк.	№ докум.	Підпис	Дата		

```

        });
    } catch(e) {
        console.log(e)

        return res.json({message: e.message})
    }
    if (response) {
        const json = response.data;
        resolve(json);
        res.json(json.data);
    }
    });
}

async getExchange(req, res) {
    let response = null;
    new Promise(async (resolve, reject) => {
        try {
            response = await
axios.get('https://api.coingecko.com/api/v3/exchanges');
        } catch(e) {
            console.log(e)

            return res.json({e})
        }
        if (response) {
            const json = response;
            resolve(json);
            res.json(json.data)
        }
    });
}

module.exports = new cryptoController()

```

					ДП.КН.22.468.01.000	Арк.
						63
Змн.	Арк.	№ докум.	Підпис	Дата		