

**МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ**  
**Чорноморський національний університет імені Петра Могили**  
**Факультет комп'ютерних наук**  
**Кафедра інтелектуальних інформаційних систем**

ДОПУЩЕНО ДО ЗАХИСТУ

В. о. завідувача кафедри інтелектуальних  
інформаційних систем

\_\_\_\_\_ Євген СІДЕНКО

« \_\_\_\_ » \_\_\_\_\_ 20\_\_ р.

**КВАЛІФІКАЦІЙНА РОБОТА**  
**НА ЗДОБУТТЯ ОСВІТНЬОГО СТУПЕНЯ МАГІСТРА**  
**ІНТЕЛЕКТУАЛЬНА СИСТЕМА ПРОГНОЗУВАННЯ**  
**ВАРТОСТІ КРИПТОВАЛЮТ**

Спеціальність 122 Комп'ютерні науки  
Освітня програма «Інтелектуальні інформаційні системи»

*Здобувач*

\_\_\_\_\_ Павло ЗАЧИНЯЄВ  
« \_\_\_\_ » \_\_\_\_\_ 20\_\_ р.

*Керівник* д-р техн. наук, професор

\_\_\_\_\_ Ірина КАЛІНІНА  
« \_\_\_\_ » \_\_\_\_\_ 20\_\_ р.

Чорноморський національний університет імені Петра Могили  
(повне найменування закладу вищої освіти)

|                     |                                      |
|---------------------|--------------------------------------|
| Факультет           | Комп'ютерних наук                    |
| Кафедра             | Інтелектуальних інформаційних систем |
| Рівень вищої освіти | Другий (магістерський)               |
| Освітній ступень    | Магістр                              |
| Спеціальність       | 122 Комп'ютерні науки                |
| Освітня програма    | Інтелектуальні інформаційні системи  |

ЗАТВЕРДЖУЮ

В. о. завідувача кафедри інтелектуальних  
інформаційних систем

\_\_\_\_\_ Євген СІДЕНКО

«\_\_\_\_\_» \_\_\_\_\_ 20\_\_ р.

**ЗАВДАННЯ**  
**на кваліфікаційну роботу здобувача**

**Зачиняєв Павло Дмитрович**

(прізвище, ім'я, по батькові здобувача)

1. Тема кваліфікаційної роботи: Інтелектуальна система прогнозування вартості криптовалют.

Керівник роботи: Калініна Ірина Олександрівна, професор кафедри ІС, д-р техн. наук, доцент.  
(прізвище, ім'я, по батькові, посада, науковий ступінь, вчене звання)

Затверджена наказом ЧНУ ім. Петра Могили від «07» липня 2025 р. № 184.

2. Строк представлення кваліфікаційної роботи «16» грудня 2025 р.

3. Очікуваний результат роботи та початкові дані, якщо такі потрібні: предметна сфера криптовалютних ринків, методи аналізу та моделі прогнозування з використанням нейронних мереж, а також набір даних із динамікою курсу криптовалют.

4. Перелік питань, що підлягають розробці: аналіз предметної сфери ринку криптовалют та дослідження теоретичних засад виявлення тенденцій і

прогнозування руху цін, обґрунтування вибору технологій і інструментальних засобів для створення системи аналізу та прогнозування, а також розробку, програмну реалізацію та дослідження якості прогнозних моделей у межах побудованої системи.

5. Перелік графічних матеріалів: презентація, рисунки, таблиці

**Керівник роботи**

\_\_\_\_\_  
(Особистий підпис)

Ірина КАЛІНІНА  
(Власне ім'я ПРІЗВИЩЕ)

**Здобувач**

\_\_\_\_\_  
(Особистий підпис)

Павло ЗАЧИНЯЄВ  
(Власне ім'я ПРІЗВИЩЕ)

Дата видачі завдання «28» червня 2025 р.

## КАЛЕНДАРНИЙ ПЛАН кваліфікаційної роботи

Тема: Інтелектуальна система прогнозування вартості криптовалют

| №  | Найменування роботи                                                                                                                                         | Початок    | Закінчення | Примітки |
|----|-------------------------------------------------------------------------------------------------------------------------------------------------------------|------------|------------|----------|
| 1  | Визначення керівника і теми. Подання заяви на затвердження теми                                                                                             | 07.07.2025 | 07.07.2025 | Виконано |
| 2  | Отримання завдання на виконання                                                                                                                             | 25.08.2025 | 27.08.2025 | Виконано |
| 3  | Розроблення та погодження календарного плану                                                                                                                | 16.09.2025 | 26.09.2025 | Виконано |
| 4  | Проведення огляду літератури та сучасних досліджень. Аналіз предметної області, ринку криптовалют, методів прогнозування та інструментів технічного аналізу | 11.10.2025 | 03.11.2025 | Виконано |
| 5  | Вибір методології, технологій та інструментальних засобів                                                                                                   | 05.11.2025 | 10.11.2025 | Виконано |
| 6  | Проектування системи, створення дизайну та програмна реалізація                                                                                             | 11.11.2025 | 16.11.2025 | Виконано |
| 7  | Підготовка та оформлення розділів фахової частини                                                                                                           | 17.11.2025 | 23.11.2025 | Виконано |
| 8  | Узагальнення результатів дослідження та формування висновків                                                                                                | 24.11.2025 | 25.11.2025 | Виконано |
| 9  | Обговорення результатів з науковим керівником. Проведення попереднього захисту                                                                              | 26.11.2025 | 26.11.2025 | Виконано |
| 10 | Унесення коригувань за результатами попереднього захисту                                                                                                    | 26.11.2025 | 06.12.2025 | Виконано |
| 12 | Другий попередній захист                                                                                                                                    | 09.12.2025 | 09.12.2025 | Виконано |
| 12 | Остаточне оформлення пояснювальної записки та підготовка презентації                                                                                        | 10.12.2025 | 11.12.2025 | Виконано |
| 13 | Подання рецензенту та рецензування                                                                                                                          | 12.12.2025 | 13.12.2025 | Виконано |
| 14 | Подання КР, її електронної копії та інших документів (відгуку, рецензії) до захисту                                                                         | 14.12.2025 | 16.12.2025 | Виконано |
| 15 | Захист КР перед ЕК                                                                                                                                          | 22.12.2025 | 22.12.2025 | Виконано |

Керівник роботи

\_\_\_\_\_  
(Особистий підпис)

Ірина КАЛІНІНА

\_\_\_\_\_  
(Власне ім'я ПРІЗВИЩЕ)

Здобувач

\_\_\_\_\_  
(Особистий підпис)

Павло ЗАЧИНЯЄВ

\_\_\_\_\_  
(Власне ім'я ПРІЗВИЩЕ)

Дата складання календарного плану

«08» \_\_липня\_\_ 2025 р.

## АНОТАЦІЯ

до кваліфікаційної роботи  
здобувача групи 601м ЧНУ ім. Петра Могили

**Зачиняєва Павла Дмитровича**

на тему: **“ІНТЕЛЕКТУАЛЬНА СИСТЕМА ПРОГНОЗУВАННЯ ВАРТОСТІ  
КРИПТОВАЛЮТ”**

Кваліфікаційна робота присвячена розробленню та програмній реалізації інтелектуальної системи аналізу й прогнозування ринку криптовалют. Актуальність дослідження зумовлена високою волатильністю криптовалютної індустрії та потребою у надійних інструментах для оцінювання ринкових процесів, оскільки виявлення та аналіз цінових тенденцій є ключовими для формування ефективної інвестиційної стратегії.

**Об’єкт роботи** – процес аналізу та прогнозування ринку криптовалют.

**Предмет роботи** – програмні засоби та методи інтелектуального аналізу даних для аналізу й прогнозування ринку криптовалют.

**Мета роботи** – підвищення ефективності планування інвестицій на ринку криптовалют шляхом створення системи виявлення ринкових тенденцій з використанням методів технічного аналізу та інструментів інтелектуального аналізу даних.

Фахова частина складається зі вступу, трьох розділів, висновків та додатків. У першому розділі розглянуто теоретичні засади аналізу криптовалютного ринку. У другому – обґрунтовано вибір методів, технологій і програмних засобів для створення системи. У третьому – подано проєктування та реалізацію інтелектуальної системи аналізу тенденцій криптовалютного ринку.

Кваліфікаційна робота містить 91 сторінок (без додатків), 62 рисунків, 2 таблиць, 38 джерел та 5 додатків.

**Ключові слова:** криптовалюти, тенденції цін на криптовалюту, індекси криптовалют, технічний аналіз, індикатори технічного аналізу, нейромережеві моделі, прогнозування.

## **ABSTRACT**

to the qualification work by the student of the group 601m of Petro Mohyla Black Sea  
National University

**Zachyniaiev Pavlo**

### **“INTELLIGENT SYSTEM FOR CRYPTOCURRENCY PRICE FORECASTING”**

The qualification work is devoted to the development and software implementation of an intelligent system for analyzing and forecasting the cryptocurrency market. The relevance of the study is driven by the high volatility of the cryptocurrency industry and the need for reliable tools to evaluate market processes, as identifying and analyzing price trends is essential for forming an effective investment strategy.

**The object of the study** is the process of analyzing and forecasting the cryptocurrency market.

**The subject of the study** is software tools and methods of intelligent data analysis for the analysis and forecasting of the cryptocurrency market.

**The aim of the work** is to increase the effectiveness of investment planning in the cryptocurrency market by creating a system for detecting market trends using technical analysis methods and intelligent data analysis tools.

The professional part consists of an introduction, three chapters, conclusions, and appendices. The first chapter examines the theoretical foundations of cryptocurrency market analysis. The second chapter justifies the choice of methods, technologies, and software tools for building the system. The third chapter presents the design and implementation of the intelligent system for analyzing cryptocurrency market trends.

The qualification work contains 91 pages (excluding appendices), 62 figures, 2 tables, 38 references, and 5 appendices.

**Keywords:** cryptocurrencies, cryptocurrency price trends, cryptocurrency indices, technical analysis, technical indicators, neural network models, forecasting.

## ЗМІСТ

|                                                                                                           |    |
|-----------------------------------------------------------------------------------------------------------|----|
| СКРОРОЧЕННЯ ТА УМОВНІ ПОКАЗНИКИ.....                                                                      | 4  |
| ВСТУП.....                                                                                                | 5  |
| 1 ТЕОРЕТИЧНІ ЗАСАДИ АНАЛІЗУ РИНКУ КРИПТОВАЛЮТ .....                                                       | 7  |
| 1.1 Сфера діяльності та особливості ринку криптовалют .....                                               | 7  |
| 1.2 Криптовалютні індекси як показники ринкової динаміки.....                                             | 11 |
| 1.3 Методології та підходи до аналізу ринку криптовалют .....                                             | 14 |
| 1.4 Індикатори та методи виявлення ринкових трендів .....                                                 | 16 |
| 1.5 Огляд платформ і сервісів для технічного аналізу криптовалют.....                                     | 23 |
| 1.6 Постановка задачі.....                                                                                | 24 |
| Висновки до розділу 1 .....                                                                               | 26 |
| 2 ТЕХНОЛОГІЇ ТА ЗАСОБИ РОЗРОБКИ ІНТЕЛЕКТУАЛЬНОЇ СИСТЕМИ<br>ПРОГНОЗУВАННЯ ВАРТОСТІ КРИПТОВАЛЮТ .....       | 28 |
| 2.1 Середовище розробки програмного забезпечення .....                                                    | 28 |
| 2.2 Мова програмування Python та її можливості .....                                                      | 29 |
| 2.3 Фреймворк Streamlit для розробки веб-додатків .....                                                   | 31 |
| 2.4 Бібліотеки для роботи з даними криптовалют .....                                                      | 32 |
| 2.5 Використання API для отримання криптовалютних даних.....                                              | 34 |
| 2.6 Онлайн-сервіси для моніторингу криптовалютних індексів.....                                           | 36 |
| 2.7 Типи нейронних мереж та їх застосування у часових рядах.....                                          | 38 |
| 2.8 Формування структури інтелектуальної системи.....                                                     | 42 |
| Висновки до розділу 2 .....                                                                               | 45 |
| 3 ПРОЄКТУВАННЯ І ПРОГРАМНА РЕАЛІЗАЦІЯ ІНТЕЛЕКТУАЛЬНОЇ<br>СИСТЕМИ ПРОГНОЗУВАННЯ ВАРТОСТІ КРИПТОВАЛЮТ ..... | 47 |
| 3.1 Основні етапи створення інтелектуальної системи прогнозування вартості<br>криптовалют.....            | 47 |
| 3.2 Попередня обробка часових рядів (препроцесінг даних) .....                                            | 49 |
| 3.3 Програмна реалізація системи прогнозування .....                                                      | 52 |
| 3.4 Збірка та налаштування програмного комплексу .....                                                    | 65 |
| 3.5 Інтерфейс системи та результати аналізу й прогнозування .....                                         | 67 |

|                                                         |     |
|---------------------------------------------------------|-----|
| Висновки до розділу 3 .....                             | 89  |
| ВИСНОВКИ.....                                           | 91  |
| ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ.....                           | 94  |
| ДОДАТОК А Головна сторінка “Home” .....                 | 98  |
| ДОДАТОК Б Сторінка “Details” .....                      | 100 |
| ДОДАТОК В Сторінка “Index” .....                        | 103 |
| ДОДАТОК Г Сторінка “Analysis” (Головні функції) .....   | 105 |
| ДОДАТОК Д Сторінка “Prediction” (Головні функції) ..... | 106 |



## **СКОРОЧЕННЯ ТА УМОВНІ ПОКАЗНИКИ**

ТА – технічний аналіз

ФА – фундаментальний аналіз

API – Application Programming Interface

BDI – Bitcoin Dominance Index

AMCI – Altcoin Market Cap Index

CFGFI – Crypto Fear and Greed Index

TCMC – Total Cryptocurrency Market Cap

EMA – Exponential Moving Average

WMA – Weighted Moving Average

MA – Moving Average

MAE – Mean Absolute Error

MAPE – Mean Absolute Percentage Error

RMSE – Root Mean Squared Error

RSI – Relative Strength Index

MACD – Moving Average Convergence Divergence

SMA – Simple Moving Average

BB – Bollinger Bands

DL – Deep Learning

UI – User Interface

LSTM – Long Short-Term Memory

GRU – Gated Recurrent Unit

BiLSTM – Bidirectional Long Short-Term Memory

CNN – Convolutional Neural Network

## ВСТУП

**Актуальність.** Сучасний ринок цифрових валют є однією з найдинамічніших і найінноваційніших сфер світової економіки. Проте інвестиційна діяльність у криптовалютному секторі супроводжується значними ризиками через високу волатильність курсів, спекулятивний характер торгів та відсутність належного правового регулювання у більшості країн. У зв'язку з цим актуальним завданням стає розроблення інтелектуальних систем, здатних здійснювати прогнозування вартості криптовалют на основі комплексного аналізу історичних даних та ринкових тенденцій.

Перспективним напрямом підвищення ефективності інвестиційного планування є використання інтелектуальних алгоритмів обробки часових рядів, які дозволяють виявляти закономірності у зміні цін криптовалют та формувати прогнози їх подальшого розвитку. Вивчення ринку цифрових активів здійснюється за допомогою криптовалютних індексів, що відображають динаміку різних активів і допомагають визначати періоди високої та низької волатильності. Це забезпечує можливість ухвалення більш обґрунтованих рішень щодо купівлі, продажу чи утримання активів.

Разом із тим, криптоіндекси мають низку недоліків – вони можуть не враховувати нові ринкові тренди, піддаватися впливу маніпуляцій або спотворювати дані через штучно завищені обсяги торгів. Тому для більш точної оцінки динаміки та прогнозування цін доцільно застосовувати методи інтелектуального аналізу даних, зокрема машинне навчання та нейронні мережі.

Методи технічного аналізу, у поєднанні з сучасними підходами штучного інтелекту, дають змогу ефективно аналізувати часові ряди криптовалют, виявляти закономірності та формувати прогнози їх вартості. Використання таких методів дозволяє автоматизувати процес виявлення трендів, оптимізувати інвестиційні стратегії та зменшити ризики, пов'язані з людським фактором.

Таким чином, **мета роботи** полягає у підвищенні ефективності процесу прогнозування вартості криптовалют шляхом розроблення інтелектуальної системи, яка поєднує методи технічного та інтелектуального аналізу для виявлення ринкових тенденцій і побудови прогнозів цін цифрових активів.

Відповідно до поставленої мети було сформульовано **завдання**:

- здійснити аналіз предметної сфери ринку криптовалют та дослідити теоретичні засади виявлення тенденцій і прогнозування руху цін;
- обґрунтувати вибір технологій та інструментальних засобів розробки системи аналізу тенденцій ринку криптовалют;
- розробити, здійснити програмну реалізацію системи аналізу та прогнозування ринку криптовалют та дослідити якість прогнозної моделі.

**Об’єкт роботи** – процес аналізу та прогнозування ринку криптовалют.

**Предмет роботи** – програмні засоби та методи інтелектуального аналізу даних для аналізу й прогнозування ринку криптовалют.

**Методологічною основою** є загальнонаукові, статистичні та інтелектуальні методи аналізу даних, зокрема обробка часових рядів і алгоритми машинного навчання. Вони дозволили всебічно дослідити ринок криптовалют і підходи до прогнозування їх вартості.

**Практична цінність** роботи полягає у створенні інтелектуальної системи, яка забезпечує автоматизований аналіз і прогнозування вартості криптовалют, що сприяє підвищенню ефективності інвестиційних рішень.

**Структура кваліфікаційної роботи.** Робота складається зі вступу, трьох розділів, висновків, списку джерел та додатків і побудована відповідно до мети та завдань дослідження.

## **1 ТЕОРЕТИЧНІ ЗАСАДИ АНАЛІЗУ РИНКУ КРИПТОВАЛЮТ**

### **1.1 Сфера діяльності та особливості ринку криптовалют**

Упродовж останнього десятиліття ринок криптовалют зазнав стрімкого розвитку, поступово інтегруючись у різні сфери економічної діяльності – фінанси, торгівлю, інвестиції та інформаційні технології. Така інтеграція зумовлює необхідність глибокого вивчення природи криптовалют як соціально-економічного та фінансового явища. Криптовалюти, що забезпечують однорангові транзакції без участі державних або банківських інституцій, стали однією з найвизначніших фінансових інновацій сучасності. Їх дослідження важливе не лише з погляду технологічного прогресу, а й через можливі ризики, виклики та нові можливості, які вони створюють у фінансовій сфері [1].

Криптовалюта – це цифровий актив, що виконує функцію платіжного засобу, створений на основі алгоритмів шифрування, які забезпечують надійність і захист транзакцій. Аналіз сучасного стану криптовалютного ринку, його структурних особливостей, ризиків та перспектив розвитку є ключовим для розуміння впливу цих активів на монетарну політику, фінансову стабільність і економічну безпеку держав. Криптовалюта розглядається як фінансова інновація нового покоління, яка здатна стати ефективною альтернативою традиційним грошам.

Основні криптовалюти, такі як Bitcoin, Ethereum, Bnb, Solana, Cardano та інші, класифікуються за рівнем ринкової капіталізації, а також за регіональним поширенням (Європа, Америка, Азійсько-Тихоокеанський регіон, Близький Схід) [2]. Розвиток криптовалютної індустрії супроводжується розширенням інвестиційних можливостей, зростанням інтересу з боку бізнесу та приватних осіб, а також активізацією дискусій щодо необхідності її державного регулювання.

Станом на сьогодні існують тисячі криптовалют, кожна з яких має власне призначення. Серед основних напрямів їх створення можна виділити такі:

- забезпечення альтернативи готівковим грошам;
- підтримка дешевих і швидких платіжних систем;

- розвиток однорангової торгівлі шляхом створення токенів для спрощення доступу до товарів і послуг;
- забезпечення роботи платформ та протоколів.

Поява нових криптовалют відбувається постійно, тому цілі їх розробки та застосування залишаються відкритими й динамічними.

Базовою технологією функціонування більшості криптовалют є блокчейн – розподілена децентралізована база даних, що гарантує прозорість, надійність і незмінність інформації (рис. 1.1). Криптовалюти дедалі більше впливають на ключові елементи грошово-кредитної та фінансової систем, що викликає підвищену увагу з боку регуляторів у всьому світі. Висока волатильність, складність технічних механізмів і можливість використання криптовалют у тіньових операціях створюють потребу у формуванні чіткої нормативно-правової бази їх функціонування.

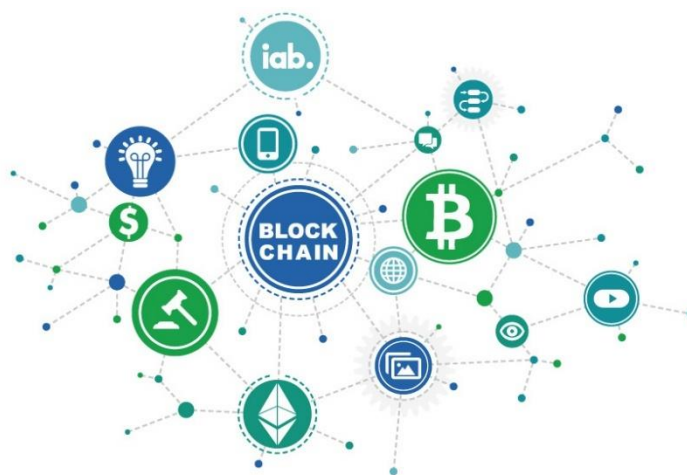


Рисунок 1.1 – Блокчейн [2]

Отже, криптовалюта є новим видом цифрових активів, що поєднує у собі технологічні та економічні характеристики. Сам термін утворено від двох складових: «крипто» – від слова «криптографія» та «валюта» – форма вартості чи грошей. Таким чином, криптовалюта – це цифрові гроші, створені на основі

криптографічних принципів, що забезпечують безпеку, достовірність і контроль над обігом [3].

Усі криптовалютні активи поділяються за функціональними ознаками. Першим і найвідомішим є Bitcoin, який має унікальний статус як перша криптовалюта у світі. Решта криптовалютних проєктів належить до таких основних категорій: стейблкоїни, альткоїни, токени, мем-монети, DeFi (децентралізовані фінансові сервіси) та NFT (незамінні токени) (рис. 1.2).

| # | Coin                                                                                           | Price                                                                                         | 1h     | 24h    | 7d      | 24h Volume        | Market Cap          | Last 7 Days                                                                           |
|---|------------------------------------------------------------------------------------------------|-----------------------------------------------------------------------------------------------|--------|--------|---------|-------------------|---------------------|---------------------------------------------------------------------------------------|
| 1 |  Bitcoin BTC  |  \$121,399   | ▼ 1.4% | ▼ 0.9% | ▲ 1.7%  | \$62,156,539,689  | \$2,419,711,492,590 |    |
| 2 |  Ethereum ETH |  \$4,325.29  | ▼ 1.3% | ▼ 3.3% | ▼ 1.7%  | \$41,599,370,975  | \$521,018,840,477   |    |
| 3 |  Tether USDT  |  \$1.00      | ▲ 0.0% | ▲ 0.0% | ▼ 0.0%  | \$122,707,079,271 | \$178,271,947,216   |    |
| 4 |  BNB BNB     |  \$1,236.46 | ▼ 2.5% | ▼ 5.5% | ▲ 16.5% | \$6,778,011,202   | \$171,853,328,116   |   |
| 5 |  XRP XRP    |  \$2.80    | ▼ 0.8% | ▼ 2.4% | ▼ 6.2%  | \$5,002,666,808   | \$167,557,675,559   |  |
| 6 |  Solana SOL |  \$220.59  | ▼ 1.8% | ▼ 0.5% | ▼ 2.8%  | \$8,105,063,842   | \$120,352,111,034   |  |
| 7 |  USDC USDC  |  \$1.00    | ▲ 0.0% | ▲ 0.0% | ▼ 0.0%  | \$12,592,483,398  | \$75,409,985,717    |  |

Рисунок 1.2 – Криптовалютні активи [18]

Стейблкоїни – це криптовалюти, вартість яких прив’язана до стабільних активів. Вони бувають забезпечені фіатними валютами (USDT, USDC), криптовалютами (DAI) або підтримуються алгоритмами (FRAX, RAI). Їхня головна мета – мінімізувати волатильність ринку.

Альткоїни – усі криптовалюти, створені після Bitcoin. Вони часто мають власні блокчейни або модифіковані версії існуючих, пропонуючи швидші транзакції, енергоефективні алгоритми чи підтримку смартконтрактів.

Токени – це цифрові активи, що випускаються на базі вже існуючих блокчейнів, найчастіше Ethereum. Вони виконують функції внутрішніх платіжних одиниць або інвестиційних інструментів у межах певних проєктів і використовуються в ICO або DeFi-платформах.

Мем-монети спочатку створювалися як жарт, але завдяки підтримці спільнот і впливу соціальних мереж набули популярності.

DeFi – це екосистема застосунків, що дозволяє проводити фінансові операції без посередників, використовуючи смартконтракти. Вони забезпечують кредитування, обмін і дохідність у криптовалюті.

NFT – це унікальні цифрові активи, які підтверджують право власності на цифрові об'єкти, такі як зображення, музика чи віртуальна нерухомість.

Таким чином, ринок криптовалют є складною екосистемою, що поєднує технологічні інновації та фінансові інструменти, створюючи основу для нової цифрової економіки.

Криптовалюти можна купувати, продавати й обмінювати на різних типах платформ – від централізованих бірж до децентралізованих рішень і навіть фізичних банкоматів. Вибір платформи залежить від потреб користувача, рівня безпеки, зручності, ліквідності та регулятивних вимог.

Централізовані біржі (CEX) дозволяють здійснювати операції з криптовалютами за допомогою фіатних грошей або інших монет. Вони пропонують високу ліквідність, широкий вибір торгових пар і зручний інтерфейс. Однак користувачі залежать від централізованого управління, що створює ризик хакерських атак, замороження коштів або втручання регуляторів (рис. 1.3).

Децентралізовані біржі (DEX) – функціонують без посередників. Торгівля відбувається напряму між користувачами через смартконтракти, що забезпечує більшу приватність і контроль над активами. Водночас DEX мають нижчу ліквідність і складніший інтерфейс, що може ускладнювати використання новачками (рис. 1.3).

ОТС-платформи (Over-the-Counter) застосовуються для великих угод поза біржами. Вони забезпечують конфіденційність і мінімізують вплив на ринкову ціну, але часто супроводжуються високими комісіями та потребують довіри між сторонами.

Пірингові (P2P) платформи, дозволяють користувачам напряду домовлятися про купівлю чи продаж криптовалюти, використовуючи зручні способи оплати. Це забезпечує гнучкість, проте збільшує ризики шахрайства або невиконання угод.

Окрему нішу займають криптовалютні банкомати (АТМ), які дозволяють обмінювати готівку на цифрові активи. Вони зручні у використанні, проте мають високі комісії та обмежену кількість доступних валют.

| Crypto Exchanges |          |  |
|------------------|----------|--|
| #                | Exchange |  |
| 1                | Binance  |  |
| 2                | MEXC     |  |
| 3                | Gate     |  |
| 4                | Bybit    |  |
| 5                | Bitget   |  |

| Crypto Exchanges |                       |                 | Decentralized Exchanges |  |
|------------------|-----------------------|-----------------|-------------------------|--|
| #                | Exchange              | 24h Volume      |                         |  |
| 1                | PancakeSwap V3 (BSC)  | \$4,691,977,938 |                         |  |
| 2                | Wombat Exchange (BNB) | \$1,509,882,684 |                         |  |
| 3                | Uniswap V3 (BSC)      | \$993,899,978   |                         |  |
| 4                | Uniswap V3 (Ethereum) | \$865,357,084   |                         |  |
| 5                | PancakeSwap (v2)      | \$843,596,355   |                         |  |

Рисунок 1.3 – Найкращі CEX та DEX [18]

Таким чином, сучасна криптовалютна інфраструктура охоплює широкий спектр платформ, що дає змогу користувачам обирати оптимальний варіант залежно від рівня досвіду, обсягу операцій і бажаного балансу між безпекою, швидкістю та приватністю.

## 1.2 Криптовалютні індекси як показники ринкової динаміки

Індекси криптовалют відіграють важливу роль у сучасній криптоекономіці, слугуючи ключовим інструментом для аналізу, оцінювання та відстеження динаміки цифрових активів. Вони дозволяють інвесторам, трейдерам і аналітикам



отримувати узагальнену інформацію про стан ринку, визначати тренди та порівнювати ефективність різних криптовалют між собою [8].

Криптовалютні індекси є статистичними показниками, що відображають зміну вартості певного набору цифрових активів – так званої «корзини криптовалют». Їх розрахунок базується на агрегованих даних із різних бірж і торгових платформ, що дозволяє формувати єдиний уніфікований показник ринкової активності. Індекси можуть бути зважені за різними критеріями – ринковою капіталізацією, обсягом торгів або часткою активів у загальному обсязі ринку. Це робить їх зручним аналітичним інструментом як для довгострокових інвесторів, так і для спекулятивних стратегій.

Залежно від структури та призначення, індекси криптовалют можуть відображати динаміку всього ринку, окремих його сегментів або навіть емоційних настроїв інвесторів. Найбільш поширені серед них, наведено нижче.

*Bitcoin Dominance Index (BDI)* – показує частку ринкової капіталізації Bitcoin у загальній вартості всіх криптовалют. Індекс дозволяє оцінити домінування Bitcoin та ступінь впливу альткоїнів на ринок.

*Altcoin Market Cap Index (AMCI)* – відображає частку ринкової капіталізації всіх альткоїнів (усіх криптовалют, крім Bitcoin) у структурі загального ринку. Його динаміка свідчить про зміну інтересу інвесторів до альтернативних активів.

*Total Cryptocurrency Market Cap Index (TCMC)* – демонструє сукупну капіталізацію ринку криптовалют, забезпечуючи узагальнений огляд його масштабів і тенденцій розвитку.

*Crypto Fear and Greed Index (CFGFI)* – оцінює емоційний стан ринку на основі показників волатильності, обсягів торгів, соціальної активності та домінування Bitcoin. Цей індекс є популярним інструментом для виявлення моментів паніки або надмірного оптимізму (рис. 1.4).

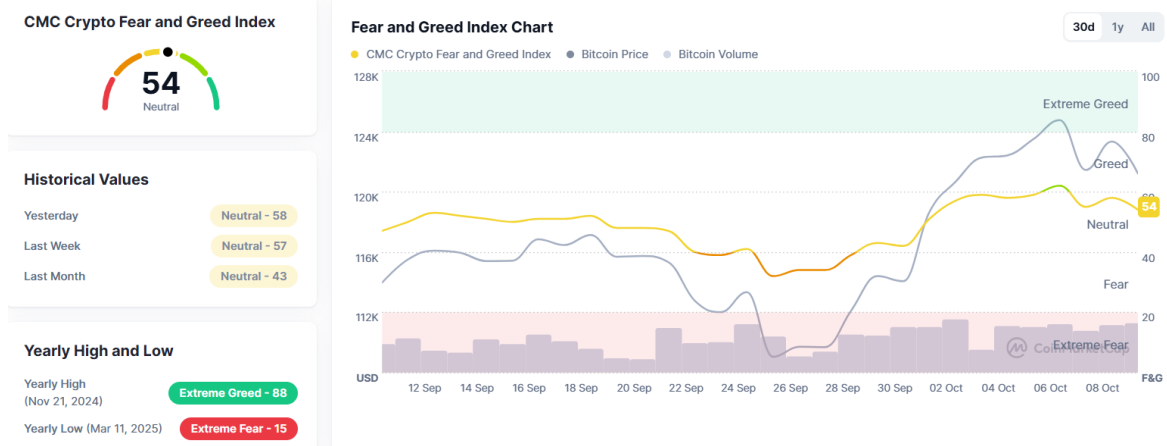


Рисунок 1.4 – Crypto Fear and Greed Index (CFGI) [18]

У сучасній фінансовій екосистемі індекси криптовалют стали невід’ємним елементом інфраструктури цифрових ринків, який забезпечує прозорість, стабільність та аналітичну підтримку їх функціонування. Завдяки індексам учасники ринку отримують можливість комплексно оцінювати динаміку цифрових активів, аналізувати тенденції та приймати обґрунтовані інвестиційні рішення.

Індекси криптовалют використовуються для виявлення ринкових закономірностей, таких як циклічність, сезонність та волатильність, що сприяє більш глибокому розумінню процесів, які відбуваються у цифровій фінансовій сфері. Вони також дозволяють визначати періоди підвищеної активності чи стабільності, що є корисним для стратегічного планування торгівлі та управління ризиками.

Водночас слід зазначити, що індекси криптовалют, як і будь-які інші фінансові інструменти, мають певні обмеження. Зокрема, показники, сформовані на основі історичних даних, можуть не враховувати поточних ринкових змін, появи нових технологічних трендів або фундаментальних трансформацій у криптоіндустрії. Крім того, на точність індексів можуть впливати маніпуляції з обсягами торгів, непрозорість окремих біржових майданчиків чи концентрація капіталу у великих проєктах.

### 1.3 Методології та підходи до аналізу ринку криптовалют

Аналіз тенденцій ринку криптовалют є складним та багатограним процесом, оскільки цей сектор характеризується високою волатильністю, динамічним розвитком технологій та залежністю від зовнішніх факторів. Для ефективного аналізу такого середовища інвестори та аналітики використовують комплекс різних підходів, які допомагають оцінити як поточний стан ринку, так і можливі напрями його подальшого розвитку. Серед основних методів виділяють фундаментальний, технічний, соціальний та сентимент-аналіз. Кожен із них має власну специфіку, переваги та обмеження, що робить їх доцільними для використання в певних умовах або у поєднанні один з одним.

*Фундаментальний аналіз (ФА)* спрямований на визначення внутрішньої вартості криптоактиву шляхом вивчення внутрішніх та зовнішніх факторів, що впливають на його ціну та потенціал розвитку. Головна мета ФА полягає у визначенні, чи є певний актив переоціненим або недооціненим, що дозволяє інвесторам приймати обґрунтовані стратегічні рішення щодо інвестування чи виходу з позицій.

Традиційно фундаментальний аналіз у класичних фінансах базується на бізнес-показниках, таких як прибутковість, балансові дані або коефіцієнти рентабельності. У сфері криптовалют до нього додаються технологічні та мережеві показники, що враховують специфіку блокчейн-проектів.

Основні метрики фундаментального аналізу криптовалют включають:

- технологічні характеристики (архітектура блокчейну, механізм консенсусу, швидкість транзакцій, масштабованість, рівень безпеки тощо);
- активність мережі (кількість транзакцій, унікальних адрес, активність гаманців);
- економічну модель (механізми емісії, винагороди, стимулювання учасників, токеноміка);

- фінансові показники (ринкова капіталізація, обсяг торгів, динаміка ціни активу);
- команду та партнерства (досвід засновників, підтримка інституційних інвесторів, стратегічні альянси).

Фундаментальний аналіз надає глибше розуміння потенціалу проєкту у довгостроковій перспективі, однак він менш ефективний для короткострокових рішень через повільну реакцію на швидкі зміни ринку.

*Технічний аналіз (ТА)* – це метод, що базується на дослідженні історичних даних (цін, обсягів торгів, часових рядів), з метою виявлення закономірностей та побудови прогнозів майбутніх цінових рухів. Він виходить з припущення, що ринкові тенденції повторюються, а історична поведінка ціни містить інформацію про її можливий подальший напрям.

Основні етапи технічного аналізу включають:

- збір даних – отримання історичних котирувань активу за різні періоди;
- побудову графіків – візуалізацію динаміки цін у вигляді свічкових чи лінійних графіків;
- використання індикаторів – застосування статистичних методів для пошуку сигналів купівлі або продажу;
- виявлення патернів – аналіз фігур графіків (трикутники, “голова і плечі”, подвійна вершина тощо);
- формування прогнозу – інтерпретація результатів і прийняття інвестиційних рішень.

Технічний аналіз є найпоширенішим інструментом серед криптотрейдерів, оскільки дозволяє швидко реагувати на зміни ринку. У даному дослідженні саме цей підхід обрано основним методом аналізу, оскільки він забезпечує можливість кількісної оцінки ринкової динаміки та побудови моделей прогнозування на основі часових рядів.

*Соціальний аналіз* фокусується на впливі соціальних та комунікативних факторів на розвиток і сприйняття криптовалют. У цьому контексті досліджується,

як громадська думка, поведінка користувачів, активність у соціальних мережах та реакція спільнот формують попит і довіру до певних активів.

До ключових напрямів соціального аналізу належать:

- моніторинг соціальних мереж (Twitter, Reddit, Telegram, X тощо) для виявлення трендів і настроїв;
- аналіз реакції на регуляторні зміни, технічні оновлення, партнерства;
- оцінка соціально-економічного контексту, у якому функціонує крипториниок.

Такий підхід дозволяє глибше зрозуміти поведінкові аспекти інвестування, а також передбачати потенційні сплески інтересу або паніки на ринку.

*Сентимент-аналіз* – це метод дослідження емоційної складової криптовалютного ринку, який вивчає ставлення інвесторів і трейдерів до певного активу. Його мета полягає у виявленні домінуючих настроїв – страху, жадібності чи нейтральності – на основі аналізу повідомлень у ЗМІ, соціальних мережах, новинних порталах і торгових сигналів.

Завдяки цьому аналізу можна визначити, чи схилиються учасники ринку до купівлі або продажу активів, що допомагає прогнозувати короткострокові рухи цін. Наприклад, переважання негативних оцінок зазвичай сигналізує про можливе зниження цін, тоді як позитивні настрої можуть свідчити про наближення фази зростання.

#### **1.4 Індикатори та методи виявлення ринкових трендів**

Для успішного проведення операцій з криптоактивами необхідно мати ефективні торгові стратегії, засновані на об'єктивному аналізі ринку. Одним із найважливіших інструментів такого аналізу є ринкові індикатори, які дають змогу виявляти тенденції, закономірності та формувати обґрунтовані рішення щодо купівлі або продажу активів.

Індикатори надають трейдерам важливу інформацію про поточні ринкові тренди, напрямки руху цін та їхню силу. В основі роботи індикаторів лежать

математичні та статистичні розрахунки, що дозволяють виявити ринкові моделі, тенденції та потенційні точки розвороту цін. Використання таких інструментів допомагає трейдерам визначати оптимальні моменти для відкриття або закриття позицій, зменшуючи ризик збитків і підвищуючи рівень прибутковості.

Ринок криптовалют характеризується високою волатильністю та непередбачуваністю, тому покладатися лише на інтуїцію під час ухвалення рішень – ризиковано. Індикатори дають змогу проводити об'єктивний аналіз на основі даних, що підвищує точність прогнозів. Для більшої надійності трейдери зазвичай комбінують кілька індикаторів, щоб підтвердити отримані сигнали, уникнути помилкових висновків і зміцнити довіру до виявленого тренду [13].

У практиці трейдингу використовують кілька основних груп торгових індикаторів, наведено нижче.

*Індикатори тренду* – визначають напрямок ринкового руху. До них належать лінії тренду та ковзні середні (MA).

*Індикатори моментуму* – оцінюють силу поточного тренду та допомагають виявити можливі розвороти цін. Найпоширеніші приклади – індекс відносної сили (RSI) та MACD (конвергенція/дивергенція ковзних середніх).

*Індикатори волатильності* – показують рівень коливань цін на ринку, допомагаючи оцінити ризики. Класичний приклад – смуги Боллінджера.

*Індикатори обсягу* – відображають кількість укладених угод та підтверджують силу тренду. Найвідоміший серед них – індикатор балансового обсягу (On-Balance Volume).

*Осцилятори* – коливаються між двома граничними рівнями, сигналізуючи про перекупленість чи перепроданість активу. Прикладом є середній індекс спрямованості (ADX).

Отже, індикатори виступають невід'ємною складовою технічного аналізу, надаючи трейдерам інструменти для прийняття рішень, що ґрунтуються на даних, а не емоціях.

Індикатор ковзної середньої використовується для згладжування короткострокових коливань і виявлення основних тенденцій на ринку (рис. 1.5). Він обчислює середнє значення цін або інших показників за певний період часу, що дозволяє визначати напрямок тренду, рівні підтримки та опору, а також знаходити сигнали для відкриття чи закриття позицій. МА є базою для інших типів середніх, зокрема експоненціальної (ЕМА) та зваженої (WMA), які точніше реагують на зміни ринку.

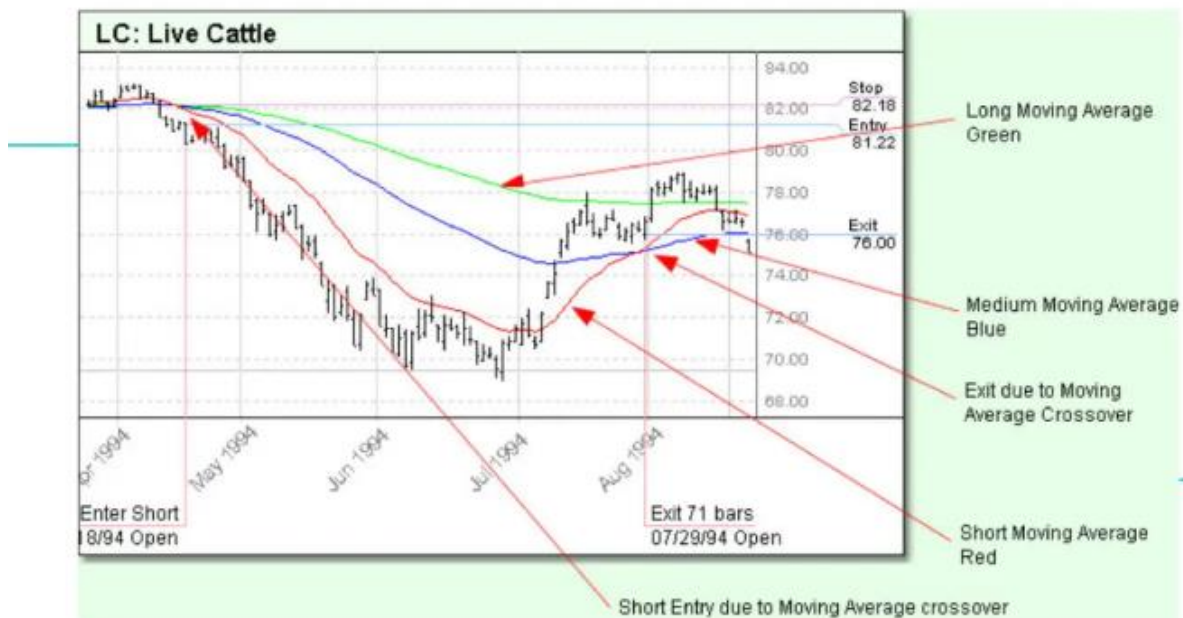


Рисунок 1.5 – Індикатор ковзних середніх (МА) [31]

Формула для розрахунку простого ковзного середнього SMA (англ. Simple Moving Average) виглядає наступним чином:

$$SMA = \frac{\sum_{i=1}^n A_i}{n}, \quad (1.1)$$

де  $A_i$  – ціна активу в період  $i$ ;

$n$  – кількість періодів, по яким робиться розрахунок – порядок ковзної середньої.

Формула для розрахунку експоненційної ковзної середньої ЕМА надає більше значення останнім цінам та враховує усі ціни попереднього періоду:

$$EMA_i = EMA_{i-1} + k \cdot (A_i - EMA_{i-1}) \quad (1.2)$$

де  $A_i$  – ціна активу на поточний момент часу;

$EMA_{i-1}$  – значення експоненційного ковзного середнього на попередній момент часу;

$k$  – експоненційно згладжуючий фактор.

Індикатор *MACD (Moving Average Convergence Divergence)* використовується для визначення напрямку та сили ринкового тренду. Він базується на співвідношенні двох експоненціальних ковзних середніх (ЕМА) – короткого (12) та довгого (26) періоду. Різниця між ними формує лінію MACD, а згладжена ЕМА з періодом 9 створює сигнальну лінію.

Формула MACD визначається наступним чином:

$$MACD = EMA_s(A) - EMA_l(A) \quad (1.3)$$

де  $EMA_s$  – короткострокова експоненційна ковзна середня;

$EMA_l$  – довгострокова експоненційна ковзна середня.

По замовчуванню обираються  $s = 12$ ,  $l = 26$ . Далі отримана лінія періоду згладжується третьою експоненційною ковзною середньою  $EMA_d$ , з періодом 9 і з'являється так звана сигнальна лінія *MACD(Signal)*.

Коли лінія MACD перетинає сигнальну зверху вниз – це вказує на можливий спад, а перетин знизу вгору сигналізує про початок зростання. Відстань між лініями відображає силу тренду, а гістограма MACD показує різницю між ними, наочно демонструючи його імпульс (рис. 1.6).





Рисунок 1.6 – Індикатор конвергенції/дивергенції ковзних середніх (MACD): лінії MACD перетинаються одна з одною [31]

Окрім класичних сигналів перетину, важливе значення має дивергенція між MACD та ціною. Якщо під час зростання ціни осцилятор формує нижчі максимуми – очікується розворот вниз (сигнал до продажу). Якщо ціна знижується, але MACD формує вищі мінімуми – це сигнал до покупки (рис. 1.7).



Рисунок 1.7 – Індикатор конвергенції/дивергенції ковзних середніх (MACD): дивергенція – ціна рухається угору, а гістограма вниз [31]

MACD часто поєднують з іншими індикаторами, зокрема ковзною середньою (MA) або індексом відносної сили (RSI), щоб підтвердити тренд і підвищити точність торгових рішень.

Індикатор RSI (*Relative Strength Index*) – індекс відносної сили, що відображає ступінь перекупленості або перепроданості активу в певному часовому інтервалі (рис. 1.8). Його значення коливається від 0 до 100: рівень вище 70 свідчить про перекупленість, нижче 30 – про перепроданість. Індикатор допомагає визначати оптимальні моменти входу чи виходу з ринку, однак під час сильних трендів RSI може залишатися у цих зонах тривалий час.



Рисунок 1.8 – Індекс відносної сили (RSI) [31]

Розрахунок RSI базується на співвідношенні середніх приростів і втрат ціни протягом певного періоду:

$$RSI = 100 - \frac{100}{1 + RS}, \quad (1.4)$$

де  $RS$  – відношення середнього приросту ціни до середнього падіння ціни протягом  $n$  періодів.

RSI доцільно використовувати разом з іншими індикаторами, наприклад MACD або ковзними середніми, щоб підтвердити сигнали ринку. Він також

ефективний для виявлення дивергенцій між рухом ціни та самим індикатором, що може вказувати на розворот тренду.

Під час низької волатильності або різких коливань RSI іноді подає хибні сигнали, тому для підвищення точності аналізу важливо враховувати кілька таймфреймів і поєднувати його з іншими методами технічного аналізу.

*Індикатор Bollinger Bands (BB)* – це один із найпоширеніших інструментів технічного аналізу, що використовується для оцінки волатильності ринку та виявлення можливих сигналів купівлі чи продажу (рис. 1.9).



Рисунок 1.9 – Індикатор лінії Боллінджера (BB) [31]

Коли ціна наближається або перетинає верхню межу, це може сигналізувати про перекупленість ринку і потенційний момент для продажу. Навпаки, досягнення або перетин нижньої межі свідчить про перепроданість і можливість купівлі активу.

Для побудови смуг Боллінджера спочатку розраховується проста ковзна середня (SMA) та стандартне відхилення (SD) ціни за певний період  $n$ . На основі цих параметрів визначаються три лінії індикатора:

$$ML = n\text{-денне SMA}, \quad (1.5)$$

$$UB = n\text{-денне SMA} + (n\text{-денне SD} \times D), \quad (1.6)$$

$$LB = n\text{-денне SMA} - (n\text{-денне SD} \times D), \quad (1.7)$$

де  $D$  – коефіцієнт відхилення;

$SD$  – стандартне відхилення,  $n$  – період розрахунку.

Перевага смуг Боллінджера полягає в їхній динамічності – вони постійно відображають поточний рівень волатильності. Завдяки цьому індикатор дозволяє своєчасно оцінювати зміни ринку та знаходити точки входу або виходу з позицій. Водночас він може давати хибні сигнали під час низької активності ринку або різких коливань, тому його рекомендують поєднувати з іншими індикаторами, зокрема RSI чи MACD, для підвищення точності аналізу.

### 1.5 Огляд платформ і сервісів для технічного аналізу криптовалют

Сьогодні на ринку існує велика кількість сервісів для технічного аналізу криптовалют, які надають різноманітні інструменти для дослідження тенденцій ринку та прогнозування цінових рухів. Частина з них доступна безкоштовно, тоді як більш просунуті функції, включно з розширеними індикаторами та аналітичними інструментами, часто потребують платної підписки.

*TradingView* – один із найпоширеніших сервісів для аналізу фінансових ринків, включаючи криптовалюту. Платформа пропонує широкий вибір графіків, технічних індикаторів та аналітичних інструментів, а також можливість обговорювати ідеї з іншими трейдерами;

*CoinMarketCap* – популярний ресурс для отримання інформації про курси криптовалют, ринкові капіталізації, обсяги торгів та інші статистичні дані. Він також надає базові інструменти для аналізу ринку;

*Nansen* – платформа для глибокого аналізу блокчейн-даних, що дозволяє відстежувати активність великих інвесторів, «розумних» гаманців та інституційних учасників ринку. Nansen допомагає трейдерам отримувати прозору інформацію про рух капіталу та приймати обґрунтовані рішення;

*Dune Analytics* – сервіс, який надає можливість створювати кастомізовані дашборди та аналізувати дані з блокчейнів у реальному часі. Платформа популярна серед аналітиків та дослідників DeFi, оскільки дозволяє отримувати точні дані про транзакції, активність користувачів та тенденції на ринку.

Попри популярність цих платформ, вони мають певні обмеження: високу вартість підписки, обмежені можливості безкоштовних версій та недостатню гнучкість під специфічні потреби трейдера.

Тому створення власного проєкту у сфері технічного аналізу криптовалют може бути виправданим кроком. Власна платформа дозволяє налаштовувати інструменти під конкретні потреби користувачів, інтегрувати їх із іншими сервісами та надавати унікальні функції, недоступні на існуючих платформах. Такий підхід може забезпечити конкурентну перевагу та залучити аудиторію, яка шукає індивідуальні рішення для аналізу ринку.

## 1.6 Постановка задачі

Побудова інтелектуальної системи для аналізу та прогнозування тенденцій ринку криптовалют є актуальним завданням у зв'язку з високою волатильністю, динамічністю та непередбачуваністю криптовалютного середовища. Ефективне прогнозування змін цінних тенденцій за допомогою сучасних методів штучного інтелекту, зокрема нейронних мереж, дає можливість підвищити точність аналітичних оцінок та оптимізувати інвестиційні стратегії.

**Об'єкт роботи** – процес аналізу та прогнозування ринку криптовалют.

**Предмет роботи** – програмні засоби, інтелектуальні методи аналізу та моделі прогнозування ринку криптовалют із використанням нейронних мереж.

**Мета роботи** – підвищення ефективності планування інвестиційних рішень на ринку криптовалют шляхом розробки інтелектуальної системи аналізу та прогнозування тенденцій із застосуванням методів технічного аналізу та нейронних мереж.

Для досягнення поставленої мети необхідно виконати такі завдання:

- здійснити аналіз предметної області ринку криптовалют та дослідити теоретичні основи прогнозування руху цін і виявлення тенденцій;
- обґрунтувати вибір технологій, інструментальних засобів і архітектури системи аналізу ринку криптовалют;
- дослідити можливості використання нейронних мереж (зокрема рекурентних моделей, таких як LSTM або GRU) для прогнозування часових рядів цін криптовалют;
- розробити програмну реалізацію інтелектуальної системи, що поєднує методи технічного аналізу, машинного навчання та нейронних мереж;
- провести експериментальну перевірку моделі та оцінити її точність, стабільність і ефективність прогнозування.

Розроблювана інтелектуальна система повинна мати такі функціональні можливості:

- збір даних – імпорт фінансових даних з крипто-сервісів через API;
- очищення даних – виявлення пропусків, аномалій, дублікатів та коректне формування наборів даних;
- аналіз історичних даних – врахування показників ціни, обсягу торгів, волатильності та інших індикаторів;
- застосування методів аналізу – технічного, фундаментального та статистичного аналізу ринку;
- прогнозування – формування короткострокових і довгострокових прогнозів цін криптовалют на основі навчання нейронних мереж;
- візуалізація результатів – відображення динаміки цін, тенденцій та прогнозів за допомогою інтерактивних графіків і діаграм.

Таким чином, розроблена інтелектуальна система забезпечить автоматизований аналіз і прогнозування вартості криптовалют.

## Висновки до розділу 1

У цьому розділі магістерської роботи розглянуто теоретичні засади аналізу та прогнозування ринку криптовалют, який сьогодні є однією з найдинамічніших і водночас найризикованіших складових світової фінансової системи. Ринок криптовалют відзначається високою волатильністю, швидкими коливаннями цін і залежністю від як економічних, так і соціально-психологічних факторів. У зв'язку з цим ефективно його дослідження потребує комплексного підходу, який поєднує статистичні, фундаментальні, технічні та соціальні методи аналізу. Застосування таких підходів дозволяє не лише оцінювати поточні тенденції, а й виявляти глибинні закономірності поведінки ринку, формувати достовірні прогнози та мінімізувати інвестиційні ризики.

Проаналізовано широкий спектр інструментів і програмних середовищ, що застосовуються для дослідження криптовалютного ринку, зокрема Python (з бібліотеками `pandas`, `numpy`, `scipy`, `matplotlib`, `pytorch`), а також спеціалізовані сервіси `CoinGecko` та `CoinMarketCap`. Встановлено, що більшість існуючих рішень націлені переважно на аналітичну обробку даних, але не забезпечують повної автоматизації процесу прогнозування. Це створює певні труднощі для користувачів без технічної підготовки та підкреслює потребу у створенні зручних, інтуїтивно зрозумілих програмних систем, які б поєднували простоту використання з потужними інструментами аналітики та прогнозування.

Особливу увагу приділено аналізу криптовалютних індексів, що є важливими індикаторами ринкової динаміки. До таких індексів належать показники ринкової капіталізації, обсягів торгів, рівня волатильності, домінування біткоїна, а також індекси настроїв інвесторів (наприклад, `Fear and Greed Index`). Їх інтеграція у моделі аналізу дозволяє підвищити точність прогнозів, врахувати макроекономічні й поведінкові чинники, а також забезпечити більш об'єктивну оцінку стану ринку.

Проведений огляд методологій продемонстрував доцільність поєднання класичних методів технічного аналізу (`MA`, `RSI`, `MACD`, `Bollinger Bands`) із

сучасними технологіями штучного інтелекту. Зокрема, застосування нейронних мереж і машинного навчання дає змогу моделювати складні нелінійні залежності між ринковими параметрами, виявляти приховані закономірності та здійснювати довгострокові прогнози з вищою точністю. Нейромережеві моделі можуть навчатися на історичних даних, адаптуватися до змін у ринкових умовах і ефективно реагувати на нетипові коливання криптовалютних курсів.

Отже, аналіз теоретичних засад ринку криптовалют дозволив визначити ключові напрями подальших досліджень та формування концепції автоматизованої системи прогнозування. Така система має забезпечувати повний цикл роботи з даними – від збору, очищення та попередньої обробки до побудови прогнозних моделей і візуалізації результатів у зручному для користувача форматі. Її впровадження сприятиме підвищенню ефективності аналітичних процесів, полегшенню ухвалення рішень щодо інвестування, а також зменшенню впливу людського фактора на результати аналізу.



## **2 ТЕХНОЛОГІЇ ТА ЗАСОБИ РОЗРОБКИ ІНТЕЛЕКТУАЛЬНОЇ СИСТЕМИ ПРОГНОЗУВАННЯ ВАРТОСТІ КРИПТОВАЛЮТ**

### **2.1 Середовище розробки програмного забезпечення**

Visual Studio Code (VS Code) – це сучасне та потужне середовище розробки від Microsoft, що працює на операційних системах Windows, Linux та macOS. Воно поєднує легкість текстового редактора з функціональністю повноцінної IDE, що робить його універсальним інструментом для створення застосунків на різних мовах програмування та платформах.

VS Code забезпечує програмістів усім необхідним для ефективної роботи: відлагоджувач, інтеграцію з Git, підсвічування синтаксису, IntelliSense, автодоповнення коду, рефакторинг та керування проєктами. Це дозволяє легко писати, тестувати та налагоджувати програмне забезпечення будь-якої складності.

Редактор має широкі можливості персоналізації – зміну тем оформлення, гарячих клавіш, розширень та файлів конфігурації, що робить його максимально гнучким. Попри те, що готові збірки розповсюджуються під пропрієтарною ліцензією, вихідний код VS Code відкритий і доступний на GitHub під ліцензією MIT.

Історично, Visual Studio Code було представлено 29 квітня 2015 року на конференції Microsoft Build, а вже 18 листопада того ж року код став відкритим. Після виходу з бета-тестування 14 квітня 2016 року, VS Code швидко здобув популярність серед розробників завдяки простоті, швидкодії та активній спільноті, яка постійно створює нові розширення.

Однією з переваг VS Code є підтримка великої кількості мов програмування, серед яких Python, JavaScript, TypeScript, C++, Java тощо. Крім того, середовище має вбудовані інструменти для роботи з Docker та Kubernetes, що робить його ідеальним для розробки мікросервісних і хмарних застосунків.

Завдяки поєднанню потужності, зручності та активного розвитку VS Code сьогодні є одним із найпопулярніших і найгнучкіших середовищ розробки програмного забезпечення.

## **2.2 Мова програмування Python та її можливості**

Python – це інтерпретована мова програмування високого рівня з об'єктно-орієнтованою архітектурою та динамічною типізацією, створена Гвідо ван Россумом у 1990 році. Завдяки простому синтаксису, гнучкості та великій кількості готових бібліотек, Python став одним із найпопулярніших інструментів у сучасній розробці програмного забезпечення.

Мова підтримує модулі та пакети, що сприяє структурованому підходу до програмування та повторному використанню коду. Вона також реалізує кілька парадигм програмування – об'єктно-орієнтовану, процедурну, функціональну та аспектно-орієнтовану, що робить Python універсальним засобом для розв'язання завдань різного типу.

Основні переваги Python:

- простий і зрозумілий синтаксис, де використання відступів робить код читабельним та логічно структурованим;
- кросплатформеність, що забезпечує роботу програм на різних операційних системах;
- велика стандартна бібліотека, яка містить готові модулі для роботи з файлами, мережею, базами даних та графічним інтерфейсом;
- можливість інтерактивної роботи, що полегшує експерименти та тестування коду;
- інтегроване середовище розробки IDLE, створене на самій мові Python;
- підтримка математичних обчислень, у тому числі з комплексними числами;
- відкритий вихідний код, що забезпечує вільне розповсюдження та модифікацію.

Python пропонує ефективні високорівневі структури даних та зручний підхід до об'єктно-орієнтованого програмування, що сприяє швидкій розробці прикладних програм і скриптів. Його інтерпретатор та стандартна бібліотека доступні безкоштовно на офіційному сайті мови, а також можуть бути розширені модулями, написаними на C або C++.

Мова є повністю кросплатформенною: вона підтримується на Windows, Linux, macOS, а також на мобільних та вбудованих платформах, включаючи Android, iOS та навіть мейнфрейми. Python використовує динамічну типізацію, коли тип змінної визначається під час виконання, та підтримує різноманітні структури даних – списки, кортежі, словники, множини тощо.

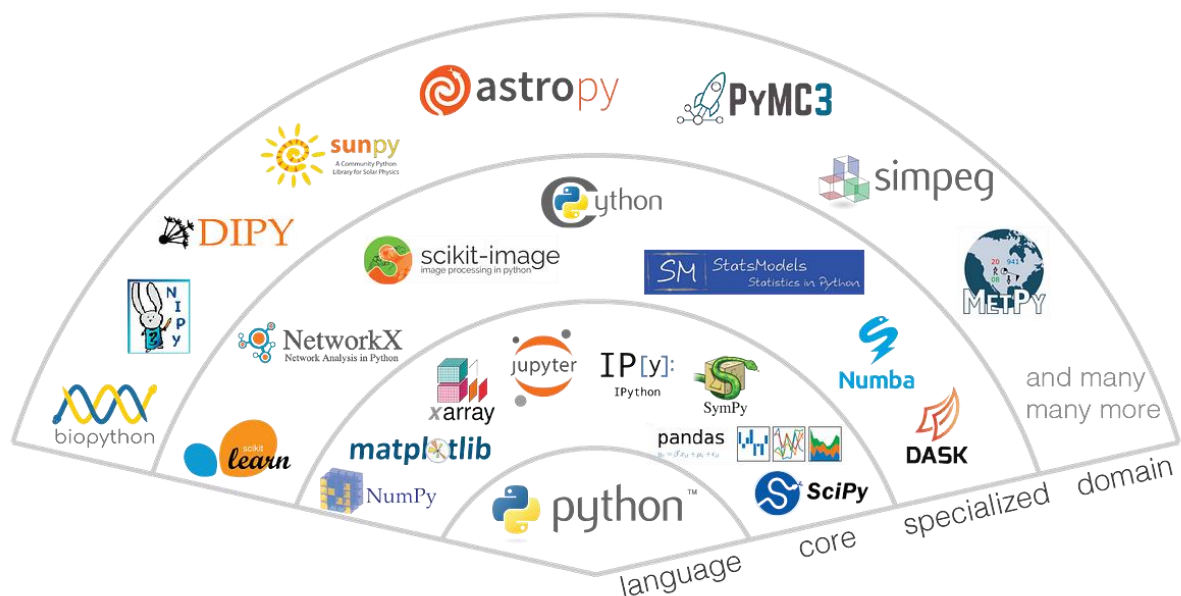


Рисунок 2.1 – Екосистема Python [19]

Python – це не лише мова програмування, а й потужне середовище для аналітики та наукових обчислень. Його екосистема (рис. 2.1) включає такі бібліотеки, як NumPy, Pandas, Matplotlib та Seaborn, що дозволяють ефективно працювати з великими наборами даних, виконувати статистичний аналіз, візуалізацію та моделювання.

Крім того, Python має розвинуті інструменти для роботи з мережевими даними та API-бібліотеки для криптобірж дають змогу отримувати, очищати й обробляти дані з відкритих джерел. Завдяки цьому Python активно використовується у дослідженнях ринку криптовалют, де необхідна обробка великих обсягів інформації та прогнозування динаміки цін.

Також мова має широкі можливості для машинного навчання та штучного інтелекту. За допомогою бібліотек TensorFlow, Keras, Scikit-learn та PyTorch можна створювати моделі прогнозування, розпізнавання та класифікації даних, що робить Python провідним інструментом у сфері інтелектуального аналізу даних.

Завдяки потужній спільноті розробників та постійним оновленням, Python продовжує розвиватися, залишаючись одним із найзручніших та найефективніших інструментів для створення інноваційних програмних продуктів, зокрема у сфері аналізу та прогнозування тенденцій ринку криптовалют.

### **2.3 Фреймворк Streamlit для розробки веб-додатків**

Розробка інтелектуальної системи для прогнозування цін криптовалют потребує зручного інструменту для візуалізації результатів аналізу та взаємодії з користувачем у режимі реального часу. Одним із найефективніших рішень для цього є фреймворк Streamlit – сучасний інструмент для створення веб-додатків на основі Python [21].

Streamlit дозволяє розробникам швидко перетворювати аналітичні або дослідницькі скрипти на повноцінні інтерактивні веб-застосунки, не вдаючись до традиційних технологій фронтенд-розробки. Це робить його особливо корисним для побудови систем аналізу даних, машинного навчання та прогнозування.

Основні особливості Streamlit [22]:

- простота використання. Однією з ключових особливостей фреймворку є його інтуїтивний синтаксис. Для створення веб-додатків достатньо базових знань Python – без необхідності використовувати HTML, CSS чи JavaScript;

- інтерактивність. Streamlit забезпечує миттєву реакцію інтерфейсу на зміну параметрів користувачем. Це дозволяє створювати динамічні аналітичні панелі, що оновлюються в реальному часі;
- широкий набір компонентів. Фреймворк має вбудовані елементи управління, такі як слайдери, поля вводу, кнопки, чекбокси, графіки та таблиці, що полегшують створення інтерфейсів без додаткового кодування;
- автоматичне оновлення. Зміни у вихідному коді застосунку миттєво відображаються у браузері без потреби перезавантаження – це спрощує налагодження та прискорює розробку;
- інтеграція з іншими бібліотеками. Streamlit легко поєднується з Pandas, NumPy, Plotly, Scikit-learn, PyTorch та іншими бібліотеками Python, що дає змогу інтегрувати моделі машинного навчання, статистичні обчислення та графічну візуалізацію у єдиному застосунку;
- відкритий вихідний код і спільнота. Streamlit є проєктом із відкритим кодом, який активно розвивається. Широка спільнота розробників постійно створює нові компоненти та допомагає у вирішенні технічних питань.

Завдяки цим можливостям Streamlit став одним із найпопулярніших інструментів для швидкої розробки прототипів інтелектуальних систем, аналітичних панелей та науково-дослідних інтерфейсів. У контексті аналізу ринку криптовалют використання цього фреймворку дозволяє ефективно поєднати візуалізацію, прогнозування та взаємодію з користувачем, забезпечуючи високу швидкість розробки, гнучкість і наочність результатів [23].

## **2.4 Бібліотеки для роботи з даними криптовалют**

Застосування різних бібліотек Python є ключовим елементом у забезпеченні ефективного аналізу, обробки та прогнозування ринкових цін з високою точністю і швидкістю [24]. У розробці системи аналізу тенденцій ринку криптовалют особливе значення мають такі інструменти, як Pandas, NumPy, Plotly, Streamlit, scikit-learn, PyTorch та Requests (рис. 2.2).

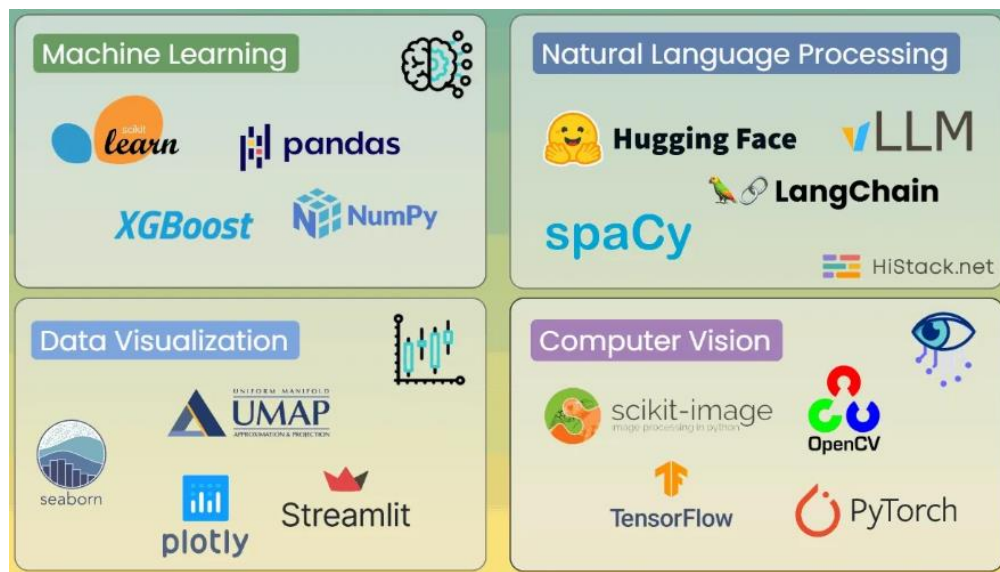


Рисунок 2.2 – Python-бібліотеки для роботи з даними [24]

Застосування цих бібліотек дозволяє виконувати повний цикл аналізу – від отримання даних до побудови інтерактивних візуалізацій та прогнозування цінових змін на криптовалютному ринку. Це сприяє підвищенню точності аналітичних висновків, автоматизації обчислень і покращенню зручності користувацької взаємодії.

Функціональні можливості основних бібліотек, що використовуються в дослідженні, наведено нижче.

*Pandas* – бібліотека для обробки та аналізу табличних даних у Python. Вона надає гнучкі структури даних (DataFrame), що дозволяють легко завантажувати, об'єднувати, фільтрувати, трансформувати та агрегувати великі обсяги інформації. Також Pandas спрощує роботу з відсутніми значеннями та часовими рядами [25].

*NumPy* – основна бібліотека для наукових обчислень. Вона забезпечує підтримку багатовимірних масивів і функцій для виконання математичних операцій, що значно підвищує швидкість обробки даних та ефективність аналітичних розрахунків.

*Plotly* – потужний інструмент для створення інтерактивних графіків і візуалізації даних. Бібліотека підтримує різноманітні типи графіків – від лінійних і

стовпчастих до теплових карт і 3D-візуалізацій. Її інтерактивні можливості дозволяють користувачам динамічно досліджувати дані у веб-додатках [26].

*Streamlit* – фреймворк для швидкої розробки інтерактивних веб-додатків на Python. Він дозволяє створювати повнофункціональні аналітичні інтерфейси для візуалізації даних, налаштування параметрів моделі та відображення результатів у режимі реального часу без необхідності використання HTML або JavaScript.

*scikit-learn* – бібліотека машинного навчання, що забезпечує широкий набір алгоритмів для масштабування, класифікації, регресії та оцінки якості моделей. У даному проєкті вона використовується для нормалізації даних (через *MinMaxScaler*) і розрахунку метрик точності прогнозу (RMSE, MAE тощо).

*PyTorch* – фреймворк для побудови та навчання нейронних мереж. Його використання дозволяє реалізувати сучасні архітектури, такі як LSTM або GRU, що ефективно моделюють часові залежності та прогнозують ринкові тренди з урахуванням історичних даних.

*Requests* – зручна бібліотека для виконання HTTP-запитів. Вона використовується для отримання актуальних даних із веб-ресурсів або API криптовалютних бірж, що забезпечує оновлення інформації в реальному часі [28].

Таким чином, поєднання зазначених бібліотек створює потужну інфраструктуру для побудови інтелектуальних систем аналізу ринку криптовалют, забезпечуючи точність прогнозів, зручність інтерактивної взаємодії та гнучкість масштабування.

Крім того, важливу роль у сучасних аналітичних рішеннях відіграє використання API, що забезпечує доступ до зовнішніх джерел даних. У криптовалютній сфері це дозволяє отримувати оперативну інформацію про ринкові показники, курси активів та новини, інтегруючи їх у єдине середовище аналізу.

## 2.5 Використання API для отримання криптовалютних даних

API (Application Programming Interface) – це набір правил, протоколів і методів, що забезпечують взаємодію між різними програмними системами.

Завдяки API розробники можуть використовувати готові сервіси для створення більш складних та функціональних додатків [31].

З точки зору дослідження ринку криптовалют, API виступає інструментом для автоматичного отримання актуальних даних про вартість криптовалют, обсяги торгів, ринкову капіталізацію та інші ключові показники з різних бірж. Наприклад, CoinGecko API дозволяє завантажувати історичні та поточні ціни, а також статистику ринку у форматі, зручному для подальшої аналітики. Це дає змогу швидко виявляти тенденції, що особливо важливо для трейдерів, аналітиків і дослідників крипторинку.

При розробці програм, що використовують API, важливо забезпечити простоту, надійність та стабільність взаємодії із зовнішнім сервісом. Це дозволяє ефективно інтегрувати систему аналізу тенденцій у веб-додаток, як у випадку зі Streamlit-проектом, і гарантує безперебійну роботу навіть за умов динамічних змін ринку.

Вибір типу API залежить від завдань і вимог системи. Для збору даних із криптовалютних бірж найчастіше застосовується RESTful API, який базується на стандартних HTTP-запитах і повертає дані у форматі JSON. Такий підхід, який використовує CoinGecko API (рис. 2.3), є простим, масштабованим і зручним для інтеграції у Python-додатки.

Використання API має як переваги, так і обмеження. Серед основних переваг можна виділити:

- швидкість розробки: готові API значно скорочують час створення програмних рішень;
- економію ресурсів: немає потреби створювати власні джерела даних – можна використовувати наявні;
- розширення функціональності: за допомогою API можна додати нові можливості без суттєвих змін у коді.

Водночас існують і певні недоліки:



- обмеження запитів: сервіси часто мають ліміти на кількість звернень або обсяг даних;
- залежність від зовнішніх ресурсів: якщо CoinGecko API змінить структуру відповіді або буде недоступний, це вплине на роботу програми;
- питання безпеки: необхідно забезпечити захист даних і безпечне зберігання ключів доступу (якщо вони використовуються).

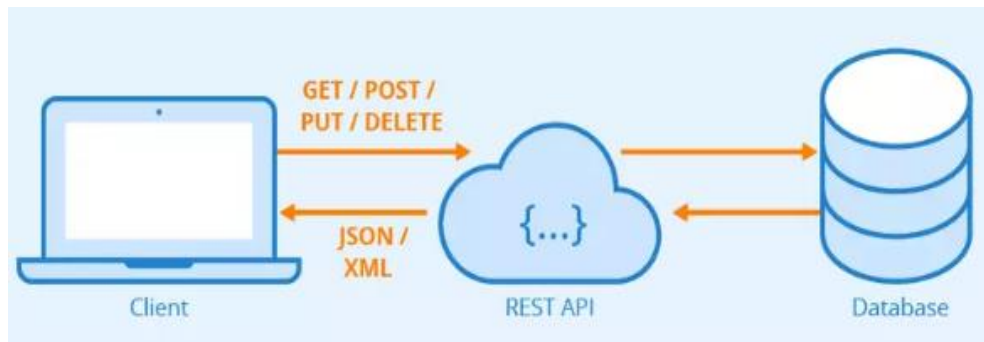


Рисунок 2.3 – Архітектура REST API [33]

Загалом, попри певні обмеження, API є ключовим інструментом сучасної розробки програмних систем. Його використання дозволяє прискорити процес створення, покращити функціональність і забезпечити інтеграцію з різними джерелами даних.

У випадку з CoinGecko API це відкриває можливості для побудови інтелектуальних систем прогнозування, які в режимі реального часу аналізують динаміку ринку та допомагають приймати обґрунтовані рішення.

## 2.6 Онлайн-сервіси для моніторингу криптовалютних індексів

Для ефективної роботи з криптовалютами важливо постійно відстежувати зміни ринку. Онлайн-сервіси моніторингу допомагають користувачам отримувати актуальні дані, аналізувати динаміку цін і приймати обґрунтовані інвестиційні рішення [18].

Такі сервіси, відомі як крипто-трекери або чекери, автоматично збирають інформацію з різних бірж, дозволяючи порівнювати ціни, обсяги торгів і капіталізацію. Вони однаково корисні як для новачків, так і для досвідчених трейдерів.

Основні критерії вибору крипто-трекера:

- надійність і репутація. Обирайте сервіси з перевіреними джерелами даних, API від відомих бірж і позитивними відгуками користувачів;
- підтримка різних валют і бірж. Краще, якщо сервіс охоплює широкий спектр криптовалют, а не лише Bitcoin чи Ethereum;
- оновлення в реальному часі. Мінімальна затримка дозволяє швидко реагувати на зміни ринку;
- зручний інтерфейс. Простота навігації допомагає швидко отримувати потрібну інформацію;
- гнучкі налаштування. Можливість створювати власну панель моніторингу для обраних активів.

Сьогодні існує велика кількість платформ і додатків для моніторингу криптовалютних індексів, які відрізняються функціональністю, дизайном та рівнем аналітики. Правильно підібраний сервіс дозволяє не лише спостерігати за цінами, але й аналізувати ринкові тренди, обсяги торгів, зміни капіталізації та навіть соціальні сигнали, що впливають на динаміку курсу.

Більшість сучасних трекерів забезпечують інтеграцію з біржами через API, що гарантує оперативність отримання даних і високу точність відображення цін. Крім цього, деякі сервіси мають можливість налаштування сповіщень про зміну ціни або обсягу торгів, що особливо корисно для активних трейдерів.

Нижче наведено найбільш популярні та інформативні сервіси, які широко застосовуються в дослідженнях і інвестиційній аналітиці.

*CoinMarketCap* – один із найвідоміших і найавторитетніших ресурсів, який відображає дані про ціни, ринкову капіталізацію, обсяги торгів, зміни за 24 години

та тиждень. Сервіс підтримує тисячі криптовалют і пропонує мобільний додаток із розширеною аналітикою;

*CoinGecko* – альтернатива з поглибленими аналітичними можливостями. Окрім стандартних показників, CoinGecko відстежує активність розробників, спільноти та ліквідність бірж, що робить його цінним інструментом для фундаментального аналізу;

*CryptoCompare* – сервіс, який поєднує аналітику ринку з освітніми матеріалами. Він надає дані про ціни, історію торгів, добові обсяги та дозволяє створювати власний криптопортфель для моніторингу прибутковості. Завдяки детальним звітам і оглядам, CryptoCompare є корисним інструментом для довгострокових інвесторів.

Використання таких платформ значно спрощує аналіз ринку, підвищує ефективність прийняття рішень і мінімізує ризики. Надійний крипто-трекер – це не лише джерело інформації, а й стратегічний інструмент для формування обґрунтованої інвестиційної стратегії.

## **2.7 Типи нейронних мереж та їх застосування у часових рядах**

Сучасні методи аналізу часових рядів дедалі частіше базуються на використанні глибоких нейронних мереж, здатних моделювати складні нелінійні залежності між спостереженнями. На відміну від класичних статистичних моделей (ARIMA, ETS, SARIMA тощо), нейронні мережі ефективно враховують довгострокові взаємозв'язки, нелінійності та шуми в даних, що робить їх особливо корисними для прогнозування високоволатильних процесів, таких як ціни криптовалют.

Загальна схема класифікації методів глибокого навчання представлена на рисунку 2.4. Червоним кольором виділено назви архітектур нейронних мереж, що використовуються в даній роботі, а саме LSTM, BiLSTM, GRU та гібридна модель CNN-LSTM. Вони належать до категорій *дискримінативних* (LSTM, BiLSTM, GRU) та *гібридних* (CNN+LSTM) методів глибокого навчання.

Застосування цих моделей зумовлене їхньою здатністю ефективно аналізувати часові ряди, виявляючи як короткострокові закономірності (через CNN), так і довгострокові залежності (через LSTM-типи мереж), що робить їх придатними для прогнозування динаміки криптовалютних цін.

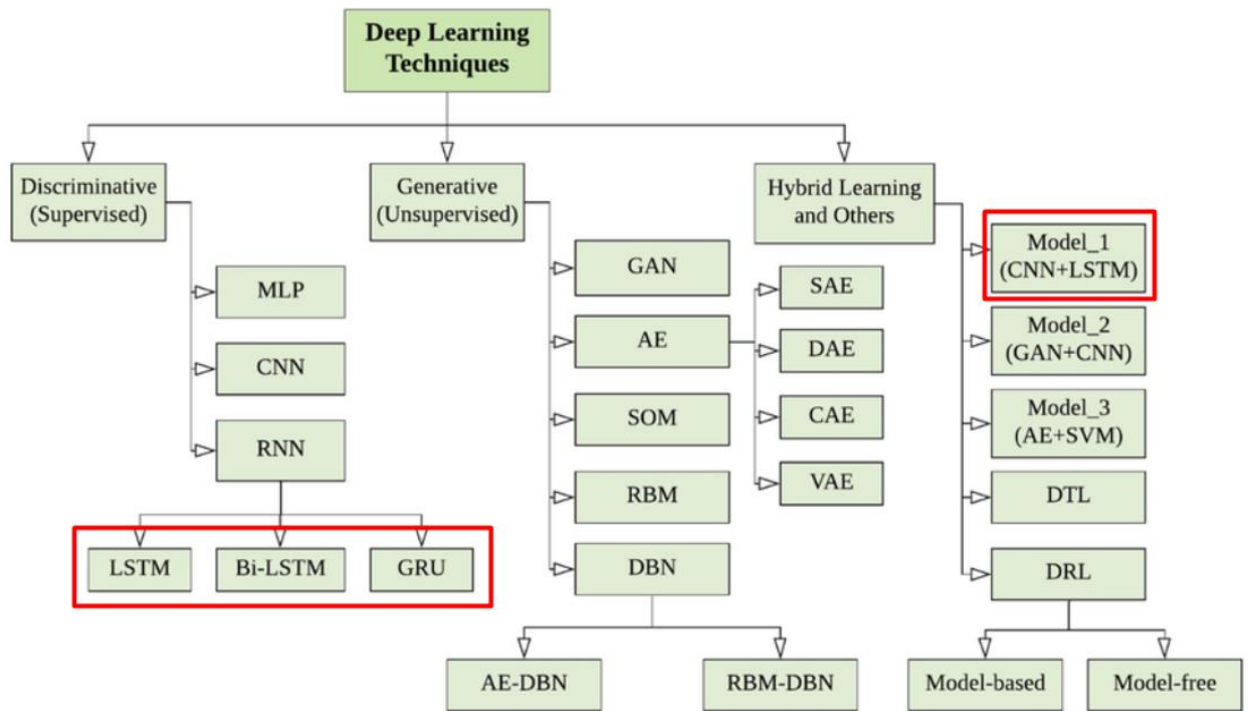


Рисунок 2.4 – Схема класифікації методів глибокого навчання [32]

**LSTM** – це вдосконалений тип рекурентної нейронної мережі (RNN), створений для подолання проблеми “згасання градієнта”, характерної для звичайних RNN. Архітектура LSTM включає спеціальні елементи – комірки пам’яті, що дозволяють мережі зберігати інформацію про попередні стани протягом тривалого часу (рис. 2.8).

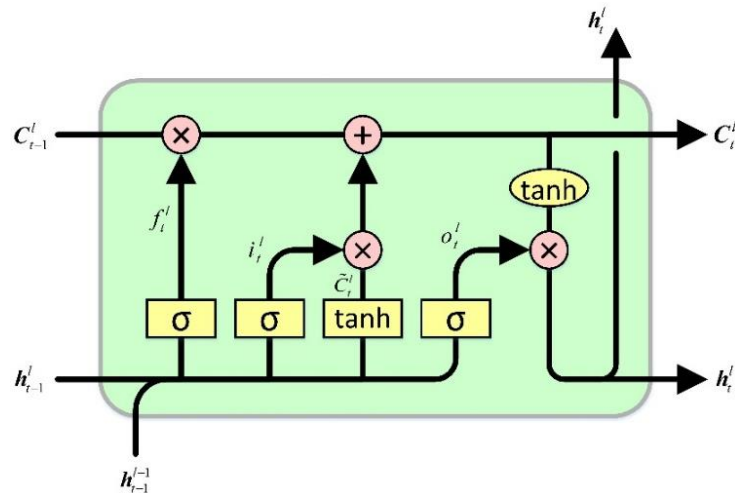


Рисунок 2.5 – Схема архітектури нейромережі LSTM [32]

Завдяки цьому LSTM ефективно виявляє довгострокові залежності у часових рядах, що робить її особливо корисною для прогнозування цін фінансових активів, де поточні значення залежать від попередніх тенденцій.

У даному проєкті модель LSTM використовується для прогнозування цін криптовалют на основі попередніх значень із заданим вікном спостережень.

**GRU** є спрощеною модифікацією LSTM, що поєднує функції “вхідних” і “вихідних” вентилів у єдиний оновлювальний вентиль. Така архітектура зменшує кількість параметрів і прискорює навчання без істотної втрати точності (рис. 2.9).

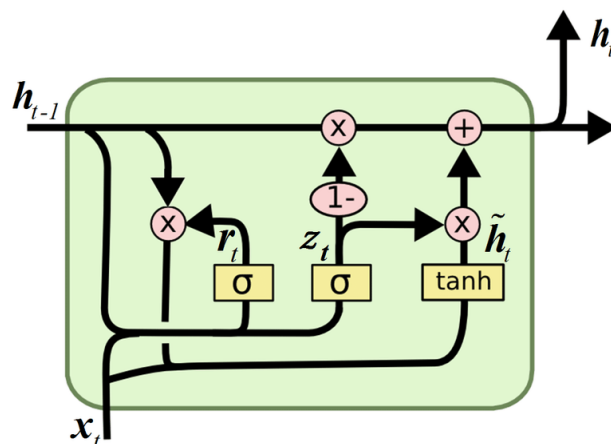


Рисунок 2.6 – Схема архітектури нейромережі GRU [32]

GRU показує себе ефективною при обробці великих обсягів даних і в ситуаціях, коли часовий контекст не потребує надто глибокої пам'яті. У застосуванні до часових рядів криптовалют це дозволяє отримати точні прогнози при меншій кількості обчислень.

**Двонапрямлена LSTM (BiLSTM)** розширює стандартну LSTM, обробляючи послідовність у двох напрямках – прямому та зворотному. Такий підхід дає змогу враховувати як попередні, так і наступні контексти під час навчання (рис. 2.10).

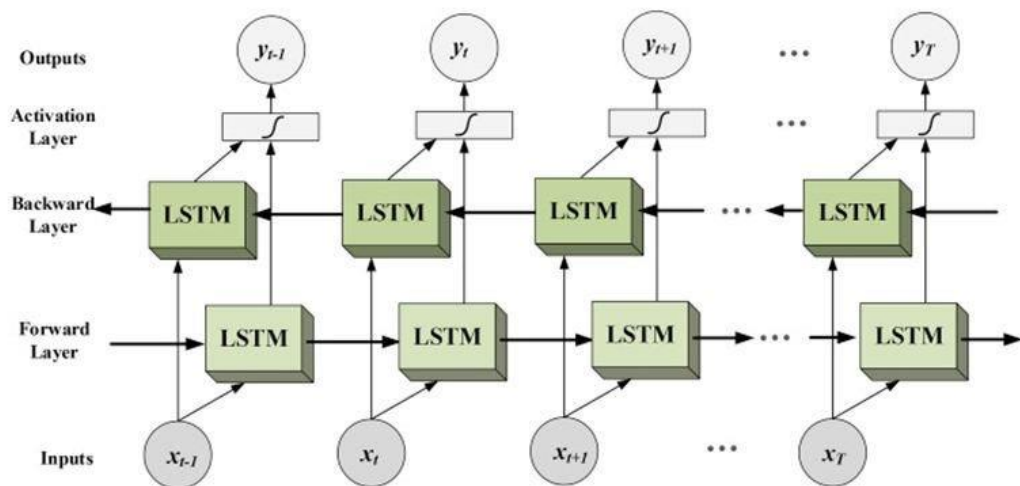


Рисунок 2.7 – Схема архітектури нейромережі BiLSTM [32]

Хоча в задачах прогнозування майбутніх значень зазвичай не використовують “майбутні” дані, двонапрямлена структура може підвищити точність моделі за рахунок кращого узагальнення часових закономірностей під час тренування.

Моделі BiLSTM демонструє високу здатність виявляти складні патерни динаміки цін і часто забезпечує нижчу помилку RMSE порівняно зі звичайною LSTM.

**Комбінована модель CNN-LSTM** поєднує сильні сторони двох архітектур: згорткової нейронної мережі (CNN) для виділення локальних ознак (короткострокових трендів) і LSTM для аналізу часових залежностей.

На першому етапі CNN виконує згортку над вхідними послідовностями, виділяючи локальні шаблони змін ціни. Потім LSTM аналізує динаміку цих шаблонів у часі (рис. 2.11).

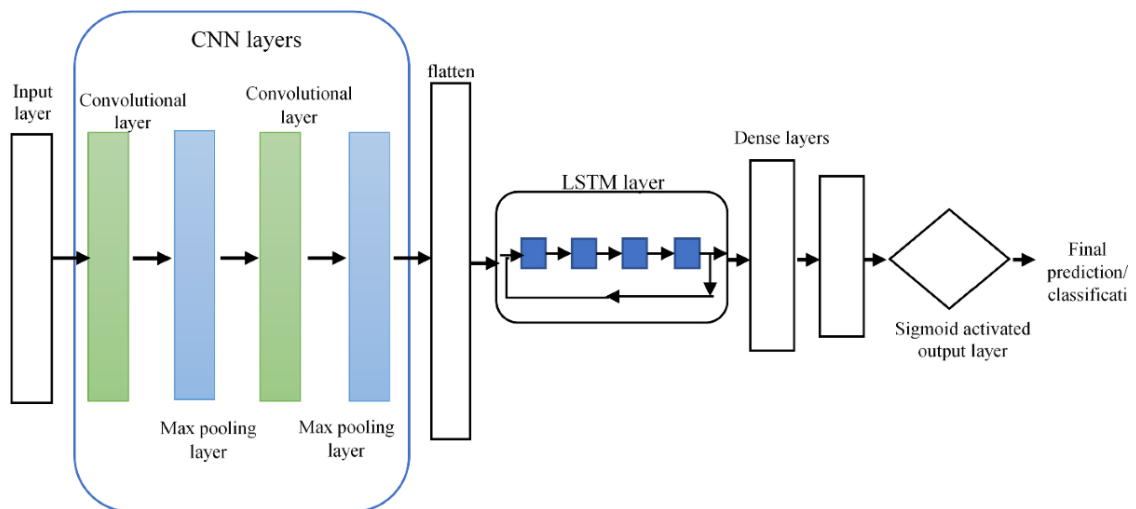


Рисунок 2.8 – Схема архітектури нейромережі CNN-LSTM [32]

Таке поєднання особливо ефективно для даних з високою частотою оновлення, як котирування криптовалют, де короткострокові коливання мають суттєвий вплив на загальну тенденцію.

У даному проєкті моделі використовуються для автоматизованого прогнозування цін криптовалют, а результати оцінюються за основними метриками – RMSE (Root Mean Square Error), MAE (Mean Absolute Error) та MAPE (Mean Absolute Percentage Error). Це дозволяє оцінити точність передбачень і вибрати оптимальну архітектуру для конкретного типу даних.

## 2.8 Формування структури інтелектуальної системи

Основна мета системи полягає в зборі, комплексному аналізі даних криптовалютного ринку та генерації точних прогнозів щодо майбутньої вартості

активів. Робота системи будується на взаємодії кількох ключових компонентів, починаючи від інтерфейсу користувача та закінчуючи підсистемою оцінки якості результатів.

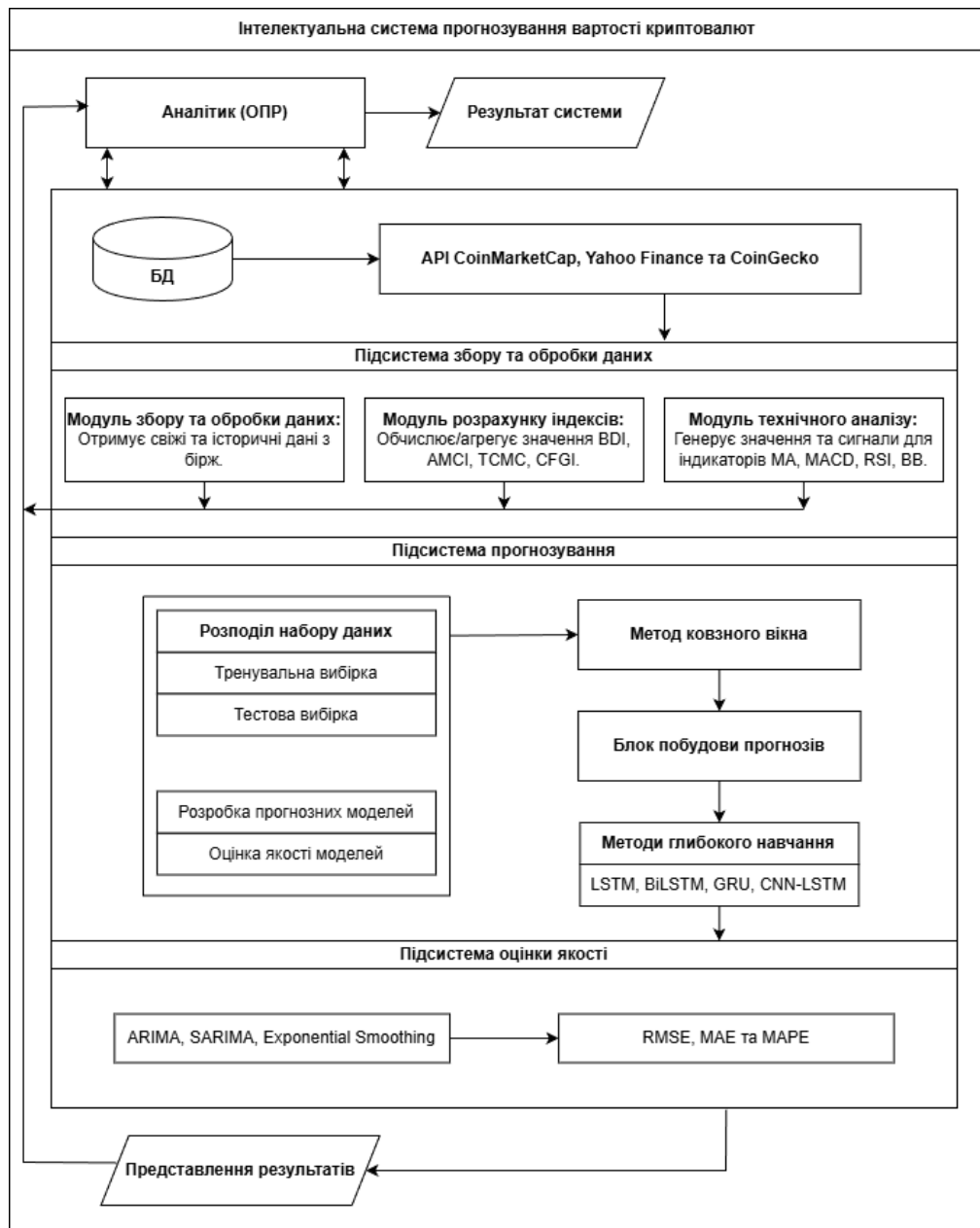


Рисунок 2.9 – Структура інтелектуальної системи

Інтерфейс користувача слугує точкою входу в систему. Користувач взаємодіє з веб-інтерфейсом, який надає доступ до всіх функцій, а всі запити та відображення результатів відбуваються через нього. Для зберігання всієї необхідної інформації,



включаючи історичні дані, розраховані індекси та результати прогнозування, використовується підсистема бази даних.

Серцевиною всієї системи є Інтелектуальна система, яка складається з чотирьох спеціалізованих модулів, наведено нижче.

1. Модуль збору та обробки даних відповідає за отримання свіжих та історичних даних з криптобірж. Ця інформація є первинною сировиною для всіх подальших аналітичних процесів.

2. Модуль розрахунку індексів проводить агрегацію даних та обчислює значення ключових ринкових індикаторів, таких як BDI, AMCI, TCMC та CFGI. Це дозволяє оцінити загальний стан ринку.

3. Модуль технічного аналізу генерує значення та торгові сигнали для класичних технічних індикаторів, зокрема ковзних середніх (MA), MACD, RSI та смуг Боллінджера (BB). Ці інструменти допомагають виявити поточні тенденції та моменти розвороту ринку.

4. Модуль прогнозування цін є найскладнішим аналітичним блоком. Він відповідає за безпосереднє створення прогнозів на основі методів глибокого навчання.

Робота модуля прогнозування будується на Підсистемі прогнозування. Для підготовки даних до навчання моделей використовується метод ковзного вікна. Безпосередньо побудова прогнозних моделей здійснюється за допомогою архітектур глибокого навчання, спеціалізованих для роботи з послідовностями, таких як LSTM, BiLSTM, GRU та їх гібридна версія CNN-LSTM. Процес розробки цих моделей включає розподіл наявних даних на тренувальну та тестову вибірки, розробку самих моделей та оцінку їх якості.

Для перевірки ефективності системи функціонує Підсистема оцінки якості. Вона аналізує точність побудованих прогнозів, використовуючи стандартні метрики регресійного аналізу: RMSE, MAE та MAPE. Результати цієї оцінки, як і кінцеві прогнози, презентуються користувачеві через інтерфейс, замикаючи таким

чином весь цикл роботи системи – від збору даних до отримання інтелектуального результату.

## **Висновки до розділу 2**

У цьому розділі було розглянуто сучасні технології, інструменти та підходи, використані при розробці інтелектуальної системи прогнозування вартості криптовалют. Проведено обґрунтування вибору середовища розробки, мови програмування, фреймворків та бібліотек, які забезпечують надійність, масштабованість і гнучкість програмного продукту. Використання оптимального технологічного стеку дає змогу поєднати аналітичну потужність алгоритмів обробки даних із зручністю користувацької взаємодії у вебсередовищі.

Середовище Visual Studio Code обрано як універсальну платформу для розробки, що підтримує інтеграцію з різними мовами, бібліотеками та фреймворками. Його гнучкість, розширюваність і підтримка систем контролю версій роблять його ефективним інструментом для створення інтерактивних інтелектуальних систем аналізу даних.

Основною мовою програмування системи є Python, що забезпечує широкий спектр можливостей для роботи з даними, побудови моделей машинного навчання та реалізації алгоритмів глибокого навчання. Використання бібліотек pandas, numpy, matplotlib, plotly, pytorch дозволяє ефективно здійснювати попередню обробку даних, проводити аналітичні розрахунки, створювати візуалізації та будувати нейромережеві архітектури для прогнозування цін криптовалют.

Для розробки інтерактивного інтерфейсу застосовано фреймворк Streamlit, який забезпечує швидке створення вебдодатків без необхідності ручного написання HTML або CSS. Завдяки Streamlit користувачі отримують можливість зручно взаємодіяти з даними, переглядати графіки, таблиці та результати прогнозів у режимі реального часу. Це суттєво підвищує інтуїтивність та наочність аналітичного процесу.

Важливою складовою системи є інтеграція API для отримання актуальних даних із криптовалютних бірж і сервісів, таких як CoinGecko та CoinMarketCap. Через бібліотеку Requests реалізовано автоматизований обмін інформацією між клієнтом і сервером, що забезпечує своєчасне оновлення показників та стабільну роботу системи у динамічному середовищі ринку.

Окрему роль у функціонуванні системи відіграє підсистема прогнозування, побудована на базі нейронних мереж типу LSTM, BiLSTM, GRU та CNN-LSTM, які спеціалізуються на аналізі часових рядів. Застосування методу ковзного вікна дозволяє ефективно підготувати дані до навчання, а використання регресійних метрик RMSE, MAE та MAPE дає змогу об'єктивно оцінити точність моделей.

Серцевиною розробленої системи є чотири взаємопов'язані модулі:

- модуль збору та обробки даних – відповідає за отримання і попередню підготовку інформації з криптобірж;
- модуль розрахунку індексів – обчислює ключові показники ринку (BDI, AMCI, TCMC, CFGI), що відображають його поточний стан;
- модуль технічного аналізу – генерує індикатори, такі як MA, RSI, MACD і Bollinger Bands, для виявлення трендів та моментів зміни напрямку ринку;
- модуль прогнозування цін – використовує нейронні мережі для створення прогнозів руху вартості криптовалют.

Таким чином, розроблена архітектура системи охоплює повний цикл аналітичної обробки даних – від збору, фільтрації та аналізу до формування прогнозів і візуалізації результатів у зручному інтерфейсі. Використання сучасних інструментів Python і Streamlit забезпечує швидкодію, масштабованість та доступність системи як для фахівців у галузі фінансових технологій, так і для користувачів без глибокої технічної підготовки.

## **3 ПРОЄКТУВАННЯ І ПРОГРАМНА РЕАЛІЗАЦІЯ ІНТЕЛЕКТУАЛЬНОЇ СИСТЕМИ ПРОГНОЗУВАННЯ ВАРТОСТІ КРИПТОВАЛЮТ**

### **3.1 Основні етапи створення інтелектуальної системи прогнозування вартості криптовалют**

Основна логіка роботи системи базується на функціонуванні Інтелектуальної системи, яка є центральним компонентом програмного комплексу. Вона складається з чотирьох взаємопов'язаних модулів, що забезпечують повний цикл – від збору даних до формування прогнозів і оцінки їхньої точності.

Після запуску вебзастосунку користувач потрапляє на домашню сторінку, звідки може перейти до однієї з чотирьох основних сторінок: «Details», «Index», «Analysis» або «Prediction», для яких створено відповідну діаграму класів (рис. 3.1).

На сторінці «Details» користувач може обрати конкретну криптовалюту для перегляду. Система автоматично звертається до API криптобірж для отримання актуальних та історичних даних. Після цього відображається тривимірний історичний графік цін та 24-годинний свічковий графік, що дозволяють користувачу отримати базову інформацію про вибрану валюту та її короткострокову динаміку.

На сторінці «Index» представлено модуль розрахунку ринкових індексів, який агрегує отримані дані та формує значення ключових показників, таких як BDI, AMCI, TCMC та CFGI. Ці показники дозволяють користувачу оцінити загальний стан ринку криптовалют, його структуру та поточні тенденції.

Сторінка «Analysis» реалізує модуль технічного аналізу, який розраховує та візуалізує класичні технічні індикатори – ковзні середні (MA), індикатор конвергенції/дивергенції ковзних середніх (MACD), індекс відносної сили (RSI) та смуги Боллінджера (BB). Ці інструменти допомагають користувачу ідентифікувати тренди, визначити точки розвороту ринку та оцінити поточну ринкову ситуацію.

На сторінці «Prediction» користувач може скористатися модулем прогнозування, який є найскладнішим аналітичним блоком системи. Для

підготовки даних використовується метод ковзного вікна, що дозволяє формувати часові вибірки для навчання моделей. Прогнозування здійснюється за допомогою архітектур глибокого навчання, спеціалізованих для роботи з часовими рядами, таких як LSTM, BiLSTM, GRU та CNN-LSTM. Система генерує лінійний графік історичних цін разом із прогнозом, що дозволяє користувачу оцінити можливі сценарії розвитку ринку.

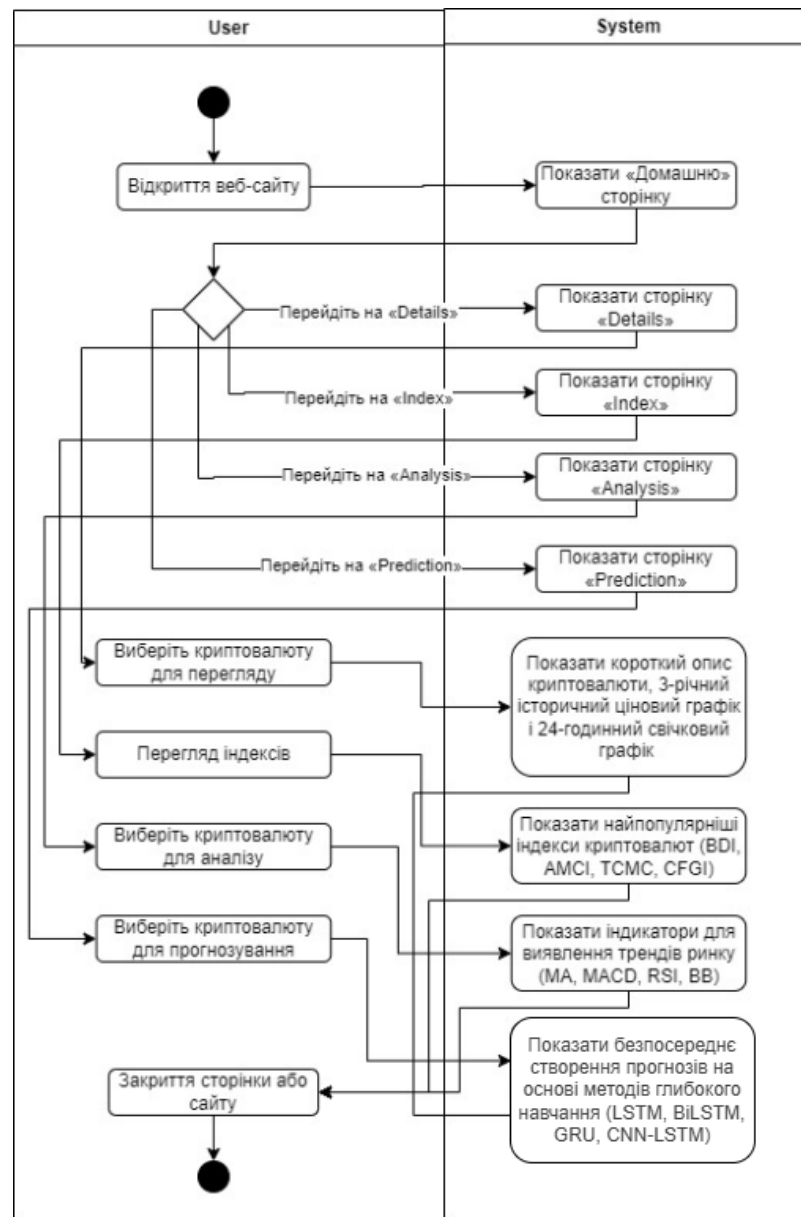


Рисунок 3.1 – Діаграма станів та переходів

Для перевірки ефективності прогнозів функціонує підсистема оцінки якості, яка використовує стандартні метрики регресійного аналізу: RMSE (Root Mean Squared Error), MAE (Mean Absolute Error) та MAPE (Mean Absolute Percentage Error). Отримані результати відображаються у вебінтерфейсі та дають можливість порівняти точність різних моделей.

Таким чином, система реалізує повний цикл роботи – від збору даних до побудови прогнозів і оцінки їхньої точності. Це забезпечує користувачу комплексний інструмент для глибокого аналізу ринку криптовалют, виявлення тенденцій і прийняття обґрунтованих інвестиційних рішень.

### **3.2 Попередня обробка часових рядів (препроцесінг даних)**

Перед безпосередньою роботою з даними важливо провести їх попередню обробку (препроцесінг). Цей етап забезпечує підвищення якості моделей машинного навчання, стабільність навчання та точність прогнозів. Препроцесінг включає очищення даних, нормалізацію, формування навчальних вибірок, а також обробку часових рядів для подальшого використання в моделях глибокого навчання.

*Очищення даних* включає кілька основних методів. Один із них – це обробка пропусків. Замість видалення рядків із пропусками іноді застосовують методи їх заповнення розрахунковими значеннями, що дозволяє зберегти обсяг даних та їхню структуру.

Наявність відсутніх значень може призвести до спотворення статистичних показників, зниження якості навчання моделі та появи помилкових прогнозів.

Процес очищення починається з перевірки даних на повноту та цілісність. Така перевірка дозволяє переконатися, що всі записи завантажені коректно й не містять порожніх значень. У разі виявлення пропусків застосовується метод `dropna()`, який автоматично видаляє рядки з відсутніми даними (NaN) (рис. 3.2).

|   | A   | B   | C   |   | A | B   | C   |
|---|-----|-----|-----|---|---|-----|-----|
| 0 | NaN | NaN | NaN | → |   |     |     |
| 1 | 1.0 | 4.0 | 4.0 |   | 1 | 1.0 | 4.0 |
| 2 | NaN | 8.0 | 2.0 |   |   |     |     |
| 3 | 4.0 | NaN | 3.0 |   | 5 | 1.0 | 1.0 |
| 4 | NaN | 8.0 | NaN |   |   |     |     |
| 5 | 1.0 | 1.0 | 5.0 |   |   |     |     |

Рисунок 3.2 – Обробка пропусків [36]

Застосування цього підходу забезпечує, що модель машинного навчання навчатиметься лише на повних і достовірних даних. Це, у свою чергу, підвищує стабільність навчання, точність прогнозів і достовірність отриманих результатів.

*Нормалізація даних* – це фундаментальний процес попередньої обробки, який полягає у приведенні числових даних до спільного масштабу. Її головна мета – усунути проблеми, пов'язані з різними діапазонами та одиницями вимірювання ознак, при цьому зберігаючи відносні відмінності між значеннями (рис. 3.3).

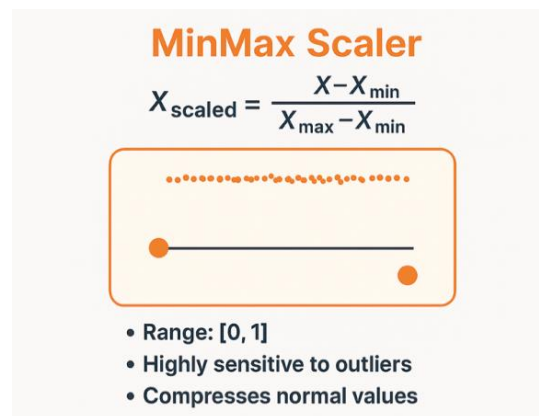


Рисунок 3.3 – Нормалізація (Min-Max Scaling) [35]

Метод Min-Max Scaling є одним з найпопулярніших способів нормалізації, який явно перетворює значення ознак у заданий діапазон, найчастіше [0, 1]. Його головна перевага полягає в тому, що він усуває проблеми, пов'язані з великими чисельними розбіжностями між ознаками.

*Розбиття на тренувальний, валідаційний та тестовий набори* є фундаментальною процедурою в машинному навчанні. Її мета – надійно оцінити

здатність моделі узагальнювати знання на нових, невидимих даних. Для цього вихідний набір даних випадковим чином розділяється на три незалежні частини, кожна з яких виконує свою унікальну роль (рис. 3.4).

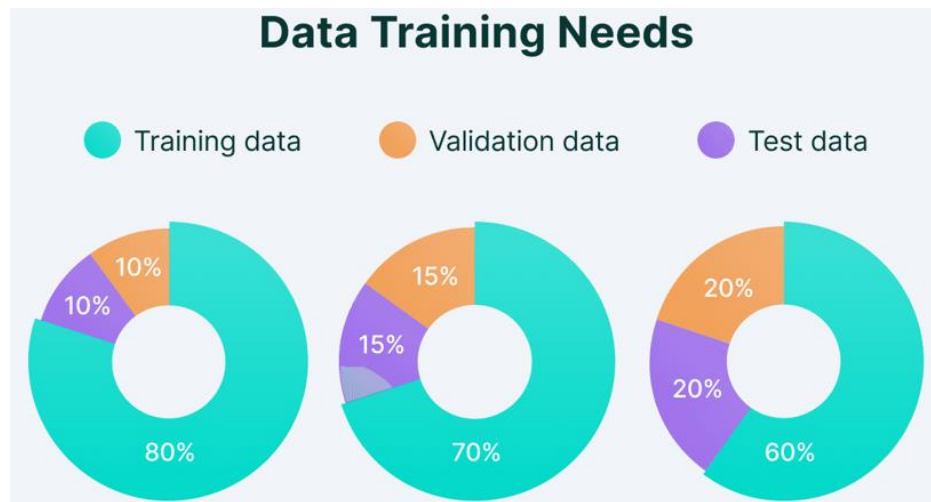


Рисунок 3.4 – Розбиття на train/test [36]

Тренувальний набір – це найбільша частина даних, яка використовується для безпосереднього навчання моделі. На цих даних алгоритм вивчає закономірності, підганяє свої внутрішні параметри (ваги) і мінімізує функцію втрат.

Валідаційний набір – використовується для проміжної перевірки якості моделі під час навчання. Він допомагає вибирати найкращі гіперпараметри, виявляти перенавчання і прийняти рішення про зупинення навчання.

На відміну від нього, тестовий набір залишається повністю ізольованим і використовується лише один раз для фінальної, об'єктивної оцінки якості вже повністю навченої та налаштованої моделі. Цей останній крок імітує реальну експлуатацію моделі і дає змогу зрозуміти, наскільки добре вона працюватиме на абсолютно нових даних.

*Метод ковзного вікна* є ключовою технікою в машинному навчанні для роботи з послідовними даними, такими як часові ряди. Його суть полягає в тому, що довга послідовність даних розбивається на менші, послідовні відрізки (вікна)



однакової довжини. Кожне таке вікно слугує окремим прикладом для навчання моделі (рис. 3.5).

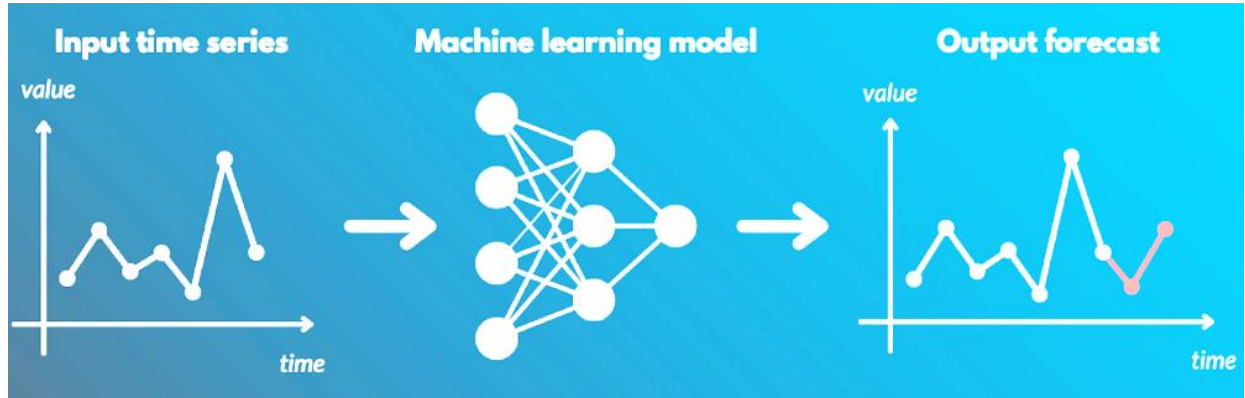


Рисунок 3.5 – Створення вікон (sliding window) [36]

Цей метод ефективно перетворює задачу прогнозування на основі послідовності на звичайну задачу навчання з учителем.

### 3.3 Програмна реалізація системи прогнозування

Програмна реалізація інтелектуальної системи прогнозування вартості криптовалют була виконана у середовищі розробки Visual Studio Code з використанням мови програмування Python, яка є однією з найпопулярніших для створення аналітичних і наукових застосунків. Для забезпечення ефективності роботи системи було застосовано низку спеціалізованих бібліотек.

Структура програмного комплексу побудована таким чином, щоб забезпечити логічний поділ функціональних компонентів і спростити подальше розширення системи. Вона складається з трьох основних файлів та однієї папки під назвою «pages», у якій розміщено чотири додаткові файли, що відповідають за реалізацію окремих сторінок застосунку (рис. 3.7).

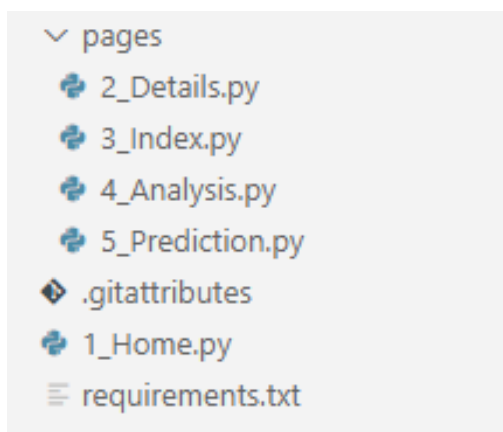


Рисунок 3.7 – Загальна структура файлів програми

Головним елементом системи є файл Home, який виконує роль центральної сторінки та забезпечує навігацію між основними модулями: Details, Index, Analysis і Prediction. Саме через головну сторінку користувач отримує доступ до аналітичних функцій, перегляду поточних ринкових індикаторів і формування прогнозів.

Зокрема, на сторінці «Details» відображається опис вибраної криптовалюти та її графіки; на «Index» – розраховані ринкові індекси; на «Analysis» – технічні індикатори для аналітичного огляду; а на «Prediction» – результати прогнозування вартості криптовалют, сформовані за допомогою нейронних мереж, спеціально адаптованих для роботи з часовими рядами.

Основні функції та їх призначення описано у таблиці 3.1.

Таблиця 3.1 – Назва функцій та їх призначення

| Назва функції    | Призначення                                                                           |
|------------------|---------------------------------------------------------------------------------------|
| fetch_data       | Отримує поточну ціну криптовалюти з API CoinMarketCap.                                |
| make_candlestick | Створює свічковий графік на основі 1-денних даних із інтервалом 5 хвилин              |
| make_line_price  | Створює лінійний графік, який показує ціну криптовалюти протягом останніх трьох років |

Закінчення таблиці 3.1

|                           |                                                                                                            |
|---------------------------|------------------------------------------------------------------------------------------------------------|
| get_info                  | Отримує розширений опис криптовалюти з даних                                                               |
| calculate_rsi             | Обчислює значення індексу відносної сили (RSI) для виявлення перекупленості/перепроданості ринку           |
| calculate_macd            | Обчислює значення індикатора конвергенції/дивергенції ковзних середніх (MACD)                              |
| calculate_bollinger_bands | Обчислює значення ліній Боллінджера (верхньої, середньої та нижньої)                                       |
| calculate_ma              | Обчислює значення ковзної середньої (МА)                                                                   |
| preprocess_data           | Здійснює очищення, нормалізацію та розділення даних на тренувальний і тестовий набори для нейронної мережі |
| create_sequences          | Формує навчальні послідовності з часових рядів для підготовки вхідних даних                                |
| train_model               | Навчає обрану нейронну мережу, оновлюючи її ваги на кожній епосі                                           |
| calculate_indicators      | Обчислює SMA, MA20 та лінії Боллінджера для оцінки ринкового тренду                                        |

Функція `st.set_page_config()` використовується для налаштування основних параметрів сторінки вебзастосунку Streamlit, зокрема заголовка, іконки, початкового стану бічної панелі та способу розташування елементів на екрані (рис. 3.8) (додаток А). Завдяки цій функції можна створити узгоджений зовнішній вигляд усіх сторінок застосунку. У даному випадку вона задає заголовок сторінки “Home”, автоматично розгортає бічну панель та вирівнює всі елементи по центру, забезпечуючи зручність і естетичність інтерфейсу.

```
st.set_page_config(
    page_title="Home",
    page_icon="🏠",
    initial_sidebar_state="expanded",
    layout='centered'
)
```

Рисунок 3.8 – Функція st.set\_page\_config

Функція st.markdown() фреймворку Streamlit дає змогу відображати текст із використанням Markdown та HTML-розмітки (рис. 3.9). Це дозволяє розробнику не лише формувати текст, а й створювати стилізовані заголовки, додавати кольори, фони, шрифти або навіть вставляти елементи CSS для покращення візуального оформлення інтерфейсу (додаток А). Таким чином, st.markdown() є гнучким інструментом для персоналізації зовнішнього вигляду застосунку.

```
st.markdown(
    """
    <div style='background-color: #464e5f; padding: 20px; border-radius: 50px; width: 100%;'>
      <h1 style='color: white; text-align: center;'>Інтелектуальна система аналізу тенденцій ринку криптовалют</h1>
    </div>
    """,
    unsafe_allow_html=True
)

st.markdown(
    """
    <style>
    .stApp {
      background-image: url('https://img.freepik.com/premium-vector/business-candle-stick-graph-chart-stock-market-investmen
      background-size: cover;
      background-repeat: no-repeat;
    }
    </style>
    """,
    unsafe_allow_html=True
)
```

Рисунок 3.9 – Функція st.markdown

Функція st.sidebar.expander() створює розгортальний елемент (expander) у бічній панелі користувацького інтерфейсу застосунку Streamlit (додаток А). Такий елемент дозволяє компактно розміщувати інформацію, яка за потреби може бути розкрита користувачем. У цьому випадку розгортальник має назву “Джерело даних”, що забезпечує зручну навігацію в інтерфейсі.

Метод `.write()` використовується для виведення вмісту всередині розгортальника – зокрема, для відображення гіперпосилання на офіційний вебсайт CoinMarketCap із підписом “Джерело даних” (рис. 3.10).

```
sources = st.sidebar.expander('Джерело даних')
sources.write(''[CoinMarketCap](https://coinmarketcap.com/uk/)'')
```

Рисунок 3.10 – Функція `st.sidebar.expander`

Функція `init` виконує ініціалізацію об’єкта класу, приймаючи як вхідні параметри `name` (назва активу) та `symbol` (біржовий символ). У межах конструктора реалізовано завантаження фінансових даних вибраного активу за допомогою бібліотеки `yfinance` (додаток Б). Зокрема, відбувається отримання вартості активу у форматі USD за останній день, а також історичних даних за останні три роки з денним інтервалом. Отримані дані надалі використовуються для аналізу та побудови графіків динаміки ціни (рис. 3.11).

```
class Crypto:
    def __init__(self, name, symbol):
        self.__name = name
        self.__symbol = symbol
        self.ticker = symbol + "-USD"
        self.initialize = yf.Ticker(self.ticker)
        self.data_1d = self.initialize.history(period="1d", interval="5m")
        self.data_3y = self.initialize.history(period="3y", interval="1d")
```

Рисунок 3.11 – Функція `init`

Функція `fetch_data`, яка використовується на сторінці «Details», призначена для отримання остання ціни криптовалюти в доларах США за допомогою API CoinMarketCap. Символ криптовалюти для запиту зберігається у полі `__symbol` класу (рис. 3.12).

Для отримання даних функція формує URL-адресу запиту, додаючи параметр `symbol`, що відповідає конкретній криптовалюти (додаток Б). Далі виконується GET-запит, до якого додаються необхідні заголовки, включно з ключем API та форматом прийому даних. Після отримання відповіді у форматі JSON, функція обробляє її та повертає поточну ціну криптовалюти в доларах США.

Цей підхід забезпечує автоматичне отримання актуальної інформації про ринок криптовалют для подальшого аналізу та візуалізації.

```
def fetch_data(self):  
    url = f"https://pro-api.coinmarketcap.com/v1/cryptocurrency/quotes/latest"  
    parameters = {  
        'symbol': self.__symbol  
    }  
    headers = {  
        'Accepts': 'application/json',  
        'X-CMC_PRO_API_KEY': '95fc1a6d-625d-4aae-9ecb-af3e92d8a300',  
    }  
    response = requests.get(url, headers=headers, params=parameters)  
    data = response.json()  
    return data['data'][self.__symbol]['quote']['USD']['price']
```

Рисунок 3.12 – Функція `fetch_data` (Details)

Функція `make_candlestick` призначена для побудови свічкового графіка (candlestick chart), який відображає динаміку цін фінансового активу за останній день (додаток Б). Для реалізації візуалізації використовується бібліотека `plotly.graph_objects`, що забезпечує інтерактивність та високу якість графічного відображення даних.

У межах функції створюється об'єкт графіка `fig`, до якого за допомогою методу `add_trace()` додається свічковий елемент, побудований на основі показників відкриття (open), максимуму (high), мінімуму (low) та закриття (close) цін активу. Такий підхід дає змогу наочно відобразити коливання вартості активу протягом обраного періоду та визначити тенденції його зміни (рис. 3.13).

```
def make_candlestick(self):
    fig = go.Figure()

    fig.add_trace(go.Candlestick(x=self.data_1d.index,
                                open=self.data_1d['Open'],
                                high=self.data_1d['High'],
                                low=self.data_1d['Low'],
                                close=self.data_1d['Close'], name = 'market data'))

    fig.update_layout(
        title=self.__name + ' - 24-годинний свічковий графік',
        yaxis_title=self.__name + ' Price (in US Dollars)'
    )

    fig.update_xaxes(
        rangelslider_visible=True,
        rangeselector=dict(
            buttons=list([
                dict(count=15, label="15m", step="minute", stepmode="backward"),
                dict(count=45, label="45m", step="minute", stepmode="backward"),
                dict(count=1, label="HTD", step="hour", stepmode="todate"),
                dict(count=6, label="6h", step="hour", stepmode="backward"),
                dict(step="all")
            ])
        )
    )

    cs = fig.data[0]

    cs.increasing.fillcolor = '#3D9970'
    cs.increasing.line.color = '#3D9970'
    cs.decreasing.fillcolor = '#FF4136'
    cs.decreasing.line.color = '#FF4136'

    return fig
```

Рисунок 3.13 – Функція make\_candlestick

Функція make\_line\_price здійснює підготовку та візуалізацію історичних даних за останні три роки. Спочатку дані перетворюються у DataFrame із скиданням індексу для подальшої обробки (додаток Б). Числові колонки – відкриття (open), максимум (high), мінімум (low), закриття (close) – конвертуються у тип float64 для забезпечення коректності обчислень. На основі цих даних будується лінійний графік, де по осі x відкладається дата, а по осі y – максимальна ціна активу. Додатково встановлюється заголовок графіка та підписи осей, що покращує наочність та зрозумілість візуалізації (рис. 3.14).

```
def make_line_price(self):
    df = self.data_3y.reset_index()
    for i in ['Open', 'High', 'Close', 'Low']:
        df[i] = df[i].astype('float64')

    fig = go.Figure([go.Scatter(x=df['Date'], y=df['High'])])

    fig.update_layout(
        title=self.__name + ' - Діаграма за 3 роки',
        yaxis_title=self.__name + ' Price (in US Dollars)'
    )

    fig.update_xaxes(
        rangelslider_visible=True,
        rangeslector=dict(
            buttons=list([
                dict(count=1, label="1mo", step="month", stepmode="backward"),
                dict(count=6, label="6mo", step="month", stepmode="backward"),
                dict(count=1, label="YTD", step="year", stepmode="todate"),
                dict(count=1, label="1y", step="year", stepmode="backward"),
                dict(step="all")
            ])
        )
    )

    return fig
```

Рисунок 3.14 – Функція make\_line\_price

Функція fetch\_data для сторінки «Index» відповідає за отримання даних щодо домінування біткоїна, домінування альткоїнів та загального ринкового капіталу з веб-ресурсу CoinMarketCap через HTTP-запити (додаток Б). Спершу виконується запит до СМС API з використанням переданих URL та заголовків. Після отримання відповіді у форматі JSON функція витягує значення домінування біткоїна, домінування альткоїнів та загальний ринковий індекс.

Далі відбувається запит до іншого веб-ресурсу для отримання індексу Fear and Greed (CFGI), також із розшифровкою відповіді у форматі JSON. Після успішного отримання всіх необхідних даних функція повертає їх у вигляді кортежу: домінування біткоїна, домінування альткоїнів, загальний ринковий індекс та індекс Fear and Greed (рис. 3.15).



```
def fetch_data(self):
    response_cmc = requests.get(self.cmc_url, headers=self.headers_cmc)
    data_cmc = response_cmc.json()
    bitcoin_dominance = data_cmc['data']['btc_dominance']
    altcoin_dominance = data_cmc['data']['eth_dominance']
    market_cap_index = data_cmc['data']['quote']['USD']['total_market_cap']

    response_fng = requests.get(self.fng_url)
    data_fng = response_fng.json()
    fng_index = int(data_fng['data'][0]['value'])

    return bitcoin_dominance, altcoin_dominance, market_cap_index, fng_index
```

Рисунок 3.15 – Функція fetch\_data (Index)

Функція calculate\_rsi обчислює індикатор відносної сили (RSI) для заданих даних фінансового інструменту, за замовчуванням використовуючи період у 14 днів (рис. 3.16). RSI дозволяє оцінити стан перекупленості або перепроданості ринку, аналізуючи коливання цін протягом визначеного періоду (додаток Г). Спершу функція визначає зміну ціни (delta), після чого розділяє її на позитивні (gain) та негативні (loss) зміни, обчислюючи їх середні значення за заданий період.

```
def calculate_rsi(data, window=14):
    delta = data["Close"].diff()
    gain = (delta.where(delta > 0, 0)).rolling(window=window).mean()
    loss = (-delta.where(delta < 0, 0)).rolling(window=window).mean()
    rs = gain / loss
    rsi = 100 - (100 / (1 + rs))
    return rsi
```

Рисунок 3.16 – Функція calculate\_rsi

Функція calculate\_macd обчислює індикатор конвергенції-дивергенції рухомих середніх (MACD), що є трендовим інструментом для визначення сили, напрямку, імпульсу та тривалості цінового тренду (рис. 3.17). Для цього застосовуються дві експоненціально зважені середні (ЕМА): короткострокова (short\_window, за замовчуванням 12 днів) та довгострокова (long\_window, за

замовчуванням 26 днів). MACD розраховується як різниця між цими ЕМА. Додатково обчислюється сигнальна лінія, яка є ЕМА від MACD за період 9 днів. Функція повертає обидва значення – MACD та сигнальну лінію – для подальшого використання у технічному аналізі (додаток Г).

```
def calculate_macd(data, short_window=12, long_window=26):  
    short_ema = data['Close'].ewm(span=short_window, min_periods=1, adjust=False).mean()  
    long_ema = data['Close'].ewm(span=long_window, min_periods=1, adjust=False).mean()  
    macd = short_ema - long_ema  
    signal_line = macd.ewm(span=9, min_periods=1, adjust=False).mean()  
    return macd, signal_line
```

Рисунок 3.17 – Функція calculate\_macd

Функція calculate\_bollinger\_bands обчислює Боллінджеріві смуги для заданих даних фінансового інструменту, за замовчуванням використовуючи період у 20 днів (додаток Г). Боллінджеріві смуги формуються на основі середнього ковзного значення (rolling\_mean) та двох стандартних відхилень (rolling\_std) від цього середнього, що утворюють верхню (upper\_band) та нижню (lower\_band) смуги. Вони дозволяють оцінити, наскільки високо або низько знаходиться ціна активу відносно попередніх торгових періодів, що допомагає виявляти потенційні точки входу та виходу з ринку (рис. 3.18).

Функція calculate\_ma обчислює просте ковзне середнє (MA) для переданих даних фінансового інструменту за певний період, який за замовчуванням становить 50 днів (рис. 3.19). Ковзне середнє згладжує коливання цін, що дозволяє легше визначати напрямок тренду (додаток Г). Воно обчислюється як середнє значення ціни закриття активу за вибраний період і повертає серію значень, яка може бути використана для технічного аналізу, допомагаючи трейдерам приймати обґрунтовані рішення на основі історичних даних.

```
def calculate_bollinger_bands(data, window=20):
    rolling_mean = data['Close'].rolling(window=window).mean()
    rolling_std = data['Close'].rolling(window=window).std()
    upper_band = rolling_mean + (rolling_std * 2)
    lower_band = rolling_mean - (rolling_std * 2)
    return rolling_mean, upper_band, lower_band
```

Рисунок 3.18 – Функція calculate\_bollinger\_bands

```
def calculate_ma(data, window=50):
    return data['Close'].rolling(window=window).mean()

for crypto in selected:
    crypto_data = data['data'][crypto]
    market_cap = crypto_data['quote']['USD']['market_cap']
    marketcap_list.append(market_cap)

    ticker = yf.Ticker(crypto + '-USD')
    hist = ticker.history(period='5y', interval='1d')
    hist['RSI'] = calculate_rsi(hist)
    hist['50MA'] = calculate_ma(hist, window=50)
    hist['200MA'] = calculate_ma(hist, window=200)
    hist['MACD'], hist['Signal Line'] = calculate_macd(hist)
    hist['Rolling Mean'], hist['Upper Band'], hist['Lower Band'] = calculate_bollinger_bands(hist)
    hist['Name'] = crypto
    dataframe_list.append(hist)
```

Рисунок 3.19 – Функція calculate\_ma

Функція preprocess\_data, здійснює попередню обробку даних, очищає їх від пропусків, нормалізує значення у діапазоні [0,1] та розділяє на тренувальний (train\_data) і тестовий (test\_data) набори у співвідношенні 80/20. Повертає також об'єкт scaler, що дозволяє відновлювати нормалізовані дані до їхніх реальних значень для інтерпретації результатів прогнозування (рис. 3.20).

Функція create\_sequences, формує послідовності для навчання рекурентних моделей на основі часових рядів, створює набори даних довжиною seq\_length (за замовчуванням 60 точок), що містять попередні значення ціни (X) та відповідне наступне значення (y). Це дозволяє моделі навчатися прогнозувати майбутні ціни на основі історичних даних (рис. 3.21).

```
def preprocess_data(data):  
    if data is None or data.empty:  
        return None, None, None  
    data = data.dropna()  
  
    scaler = MinMaxScaler(feature_range=(0, 1))  
    scaled_data = scaler.fit_transform(data[['price']].values)  
    train_size = int(len(scaled_data) * 0.8)  
    train_data = scaled_data[:train_size]  
    test_data = scaled_data[train_size:]  
    return train_data, test_data, scaler
```

Рисунок 3.20 – Функція preprocess\_data

```
def create_sequences(data, seq_length=60):  
    x, y = [], []  
    for i in range(seq_length, len(data)):  
        x.append(data[i-seq_length:i, 0])  
        y.append(data[i, 0])  
    return np.array(x), np.array(y)
```

Рисунок 3.21 – Функція create\_sequences

Функція train\_model відповідає за повний цикл навчання обраної архітектури нейронної мережі (LSTM, GRU, BiLSTM або комбінованої CNN+LSTM) на попередньо підготовлених вхідних послідовностях. В якості оптимізатора використовується алгоритм Adam, який забезпечує ефективне оновлення ваг моделі, а для оцінки якості прогнозування застосовується функція втрат MSELoss (Mean Squared Error Loss), що мінімізує середньоквадратичну похибку між прогнозованими та фактичними значеннями.

Процес навчання організований у вигляді ітерацій по заданій кількості епох, на кожній з яких відбувається повний forward pass для отримання прогнозів, розрахунок функції втрат, backpropagation для обчислення градієнтів та оновлення параметрів мережі. Для моніторингу прогресу навчання реалізовано періодичний вивід значень втрат - на початку тренування, кожні 10 епох та після завершення

останньої епохи. Після успішного завершення всього циклу тренування функція повертає повністю навчену модель, готову до використання для прогнозування на нових даних (рис. 3.22).

```
def train_model(X_train, y_train, hidden_size, num_layers, learning_rate, num_epochs, model_type="LSTM"):
    if model_type == "LSTM":
        model = LSTMModel(input_size=1, hidden_size=hidden_size, num_layers=num_layers)
    elif model_type == "GRU":
        model = GRUModel(input_size=1, hidden_size=hidden_size, num_layers=num_layers)
    elif model_type == "BiLSTM":
        model = BiLSTMModel(input_size=1, hidden_size=hidden_size, num_layers=num_layers)
    elif model_type == "CNN+LSTM":
        model = CNNLSTMModel(input_size=1, hidden_size=hidden_size, num_layers=num_layers)
    else:
        st.error("Невідома модель")
        return None

    criterion = nn.MSELoss()
    optimizer = torch.optim.Adam(model.parameters(), lr=learning_rate)

    for epoch in range(num_epochs):
        model.train()
        optimizer.zero_grad()
        output = model(X_train)
        loss = criterion(output, y_train)
        loss.backward()
        optimizer.step()
        if epoch == 0 or (epoch + 1) % 10 == 0 or epoch == num_epochs - 1:
            st.write(f"[Epoch {epoch+1}] Loss: {loss.item():.4f}")
    return model
```

Рисунок 3.22 – Функція train\_model

Функція calculate\_indicators реалізує комплексний підхід до технічного аналізу фінансових даних шляхом розрахунку ключових статистичних показників та трендових індикаторів. Вона обчислює низку ковзних середніх різних періодів: SMA50 (50-періодне просте ковзне середнє) для визначення середньострокового тренду, SMA200 (200-періодне ковзне середнє) для аналізу довгострокової тенденції ринку, та MA20 (20-періодне ковзне середнє), яке використовується як основа для побудови інших індикаторів (рис. 3.23).

```
def calculate_indicators(data):  
    data['SMA50'] = data['price'].rolling(window=50).mean()  
    data['SMA200'] = data['price'].rolling(window=200).mean()  
    data['MA20'] = data['price'].rolling(window=20).mean()  
    data['stddev'] = data['price'].rolling(window=20).std()  
    data['UpperBand'] = data['MA20'] + (data['stddev'] * 2)  
    data['LowerBand'] = data['MA20'] - (data['stddev'] * 2)  
    return data
```

Рисунок 3.23 – Функція calculate\_indicators

На основі цих розрахунків функція формує повноцінну систему смуг Боллінджера, де stddev представляє стандартне відхилення ціни за 20 періодів, що характеризує мінливість ринку. Ця система індикаторів дозволяє не лише оцінювати напрямок ринкового тренду, але й ідентифікувати ключові рівні підтримки та опору, виявляти періоди надмірної купівлі чи продажу, а також генерувати потенційні торгові сигнали на основі пробою меж каналу або відскоків від його границь.

### 3.4 Збірка та налаштування програмного комплексу

Вебзастосунок інтелектуальної системи прогнозування вартості криптовалют складається з кількох взаємопов'язаних сторінок, створених із використанням фреймворку Streamlit. Кожна сторінка виконує окрему функцію, пов'язану з аналізом, візуалізацією або прогнозуванням ринку цифрових активів. Така багаторівнева структура дозволяє користувачам отримувати доступ до широкого спектра інформації – від поточних ринкових цін і історичних даних до аналітичних показників і прогнозів, сформованих нейронними мережами.

Streamlit забезпечує створення інтерактивних веб-сторінок із зручним і зрозумілим інтерфейсом. Це дозволяє користувачам без спеціальної технічної підготовки взаємодіяти з аналітичними інструментами системи, змінювати параметри аналізу та миттєво спостерігати результати у вигляді графіків, діаграм і таблиць. Автоматичне оновлення візуалізацій у реальному часі забезпечує наочність аналізу та дозволяє швидко оцінювати поточні ринкові тенденції.

Код вебзастосунку структуровано модульно, що сприяє гнучкому розширенню функціоналу. Кожна сторінка реалізована як окремий файл або модуль (рис. 3.24), що спрощує підтримку, тестування та оновлення системи. Такий підхід дозволяє швидко додавати нові розділи, інтегрувати додаткові API або алгоритми прогнозування, зберігаючи при цьому цілісність структури проєкту.

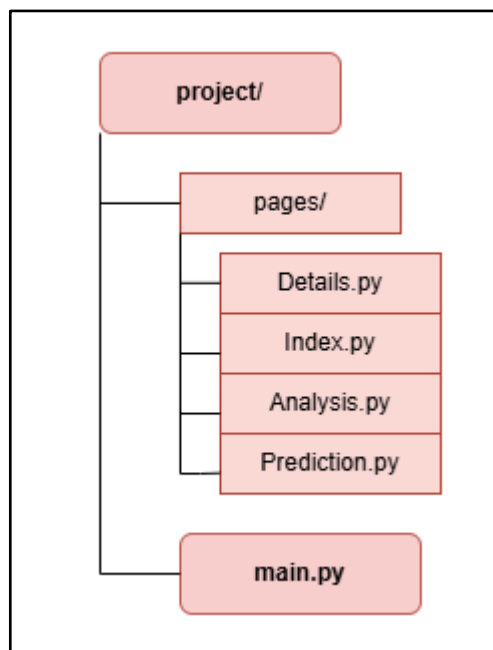


Рисунок 3.24 – Модульна структура

Особливістю застосунку є наявність модуля прогнозування, у якому реалізовано нейронні мережі глибокого навчання, зокрема LSTM, BiLSTM, GRU та CNN-LSTM, оптимізовані для аналізу часових рядів. Ці архітектури здатні розпізнавати складні закономірності у динаміці криптовалютних цін, враховуючи як короткострокові коливання, так і довгострокові тренди. Завдяки використанню методу ковзного вікна для підготовки даних система формує навчальні вибірки, що дозволяє нейронним мережам здійснювати точніші прогнози майбутніх цінових рухів.

Дотримання стандартних принципів програмування та детальне документування коду забезпечують його зрозумілість, підтримуваність і

2025 р.

масштабованість. Це створює умови для подальшого розвитку системи, інтеграції нових моделей машинного навчання або вдосконалених методів оцінки ризиків.

Таким чином, вебзастосунок поєднує зручність інтерфейсу Streamlit із аналітичною потужністю нейронних мереж, формуючи комплексну платформу для дослідження, аналізу та прогнозування стану ринку криптовалют.

### **3.5 Інтерфейс системи та результати аналізу й прогнозування**

Початкова сторінка розробленої інтелектуальної системи цін криптовалют зображена на рисунку 3.25. Вона одразу демонструє основні можливості застосунку, який забезпечує комплексний аналіз і прогнозування тенденцій криптовалютного ринку.

Зліва розміщена бічна панель із вибором потрібної сторінки для дослідження ринку.

Сторінки застосунку для аналізу ринку:

- сторінка (Details): відображає графіки динаміки цін на різних криптовалютних ринках;
- сторінка (Index): показує основні ринкові індекси, що відображають загальну динаміку крипторинку;
- сторінка (Analysis): містить технічні індикатори, які допомагають визначати тенденції та сигнали для купівлі чи продажу криптовалют;
- сторінка (Prediction): забезпечує прогнозування подальших змін цін на основі моделей глибокого навчання.

На кожному інтерфейсі система автоматизовано здійснює побудову графіків, індексів та індикаторів тенденцій ринку криптовалют, що забезпечує інтерактивний та динамічний аналіз.

Для збору, оновлення та прогнозування даних використовуються API CoinMarketCap, API Yahoo Finance та API CoinGecko, причому CoinGecko застосовується для прогнозування цін за допомогою нейронних мереж, які аналізують часові ряди та виявляють нелінійні закономірності ринку.



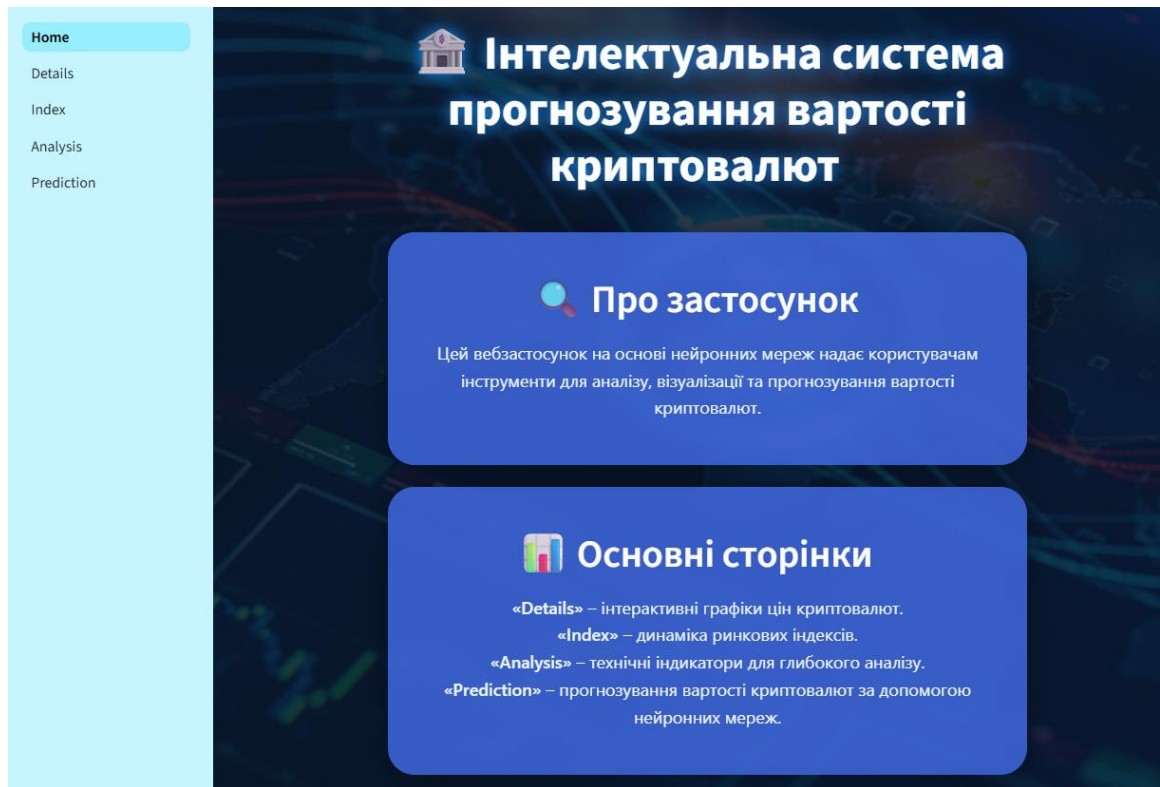


Рисунок 3.25 – Початкова сторінка системи

Для роботи з API CoinMarketCap програма формує URL-адресу запиту, встановлює параметри та ключ API, після чого надсилає GET-запит на сервер. Отримані у форматі JSON дані обробляються, і з них витягується необхідна інформація, зокрема поточна ціна криптовалюти (рис. 3.26).

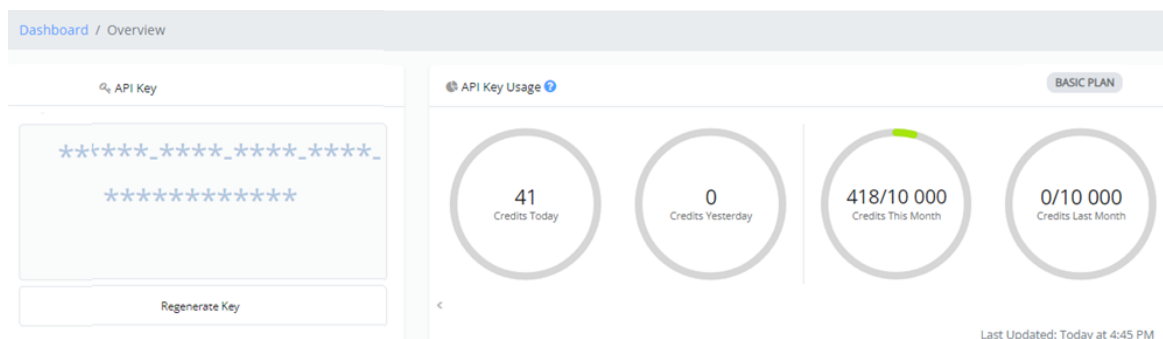


Рисунок 3.26 – API CoinMarketCap

Для API Yahoo Finance використовується бібліотека `yfinance`, яка забезпечує отримання історичних даних про ціни криптовалют за останні три роки та короткострокових даних за останні 24 години з інтервалом 5 хвилин.

API `CoinGecko` інтегровано для автоматичного отримання оновлених ринкових даних і навчання нейронної мережі, яка прогнозує можливі зміни вартості криптовалют на основі часових рядів та поточних ринкових трендів.

Після отримання даних на сторінці «Details» користувач може обрати конкретну криптовалюту для перегляду (рис. 3.27). Система відображає трирічний історичний ціновий графік та 24-годинний свічковий графік, що дозволяє отримати як загальну картину, так і короткострокову динаміку вибраного активу.

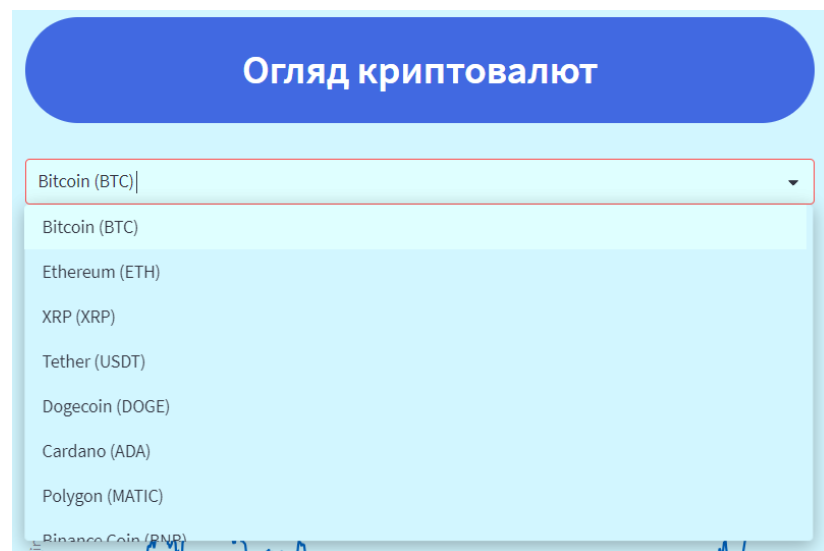


Рисунок 3.27 – Вибір конкретної криптовалюти

Трирічний історичний ціновий графік обраної криптовалюти (рис. 3.28) відображає її довгострокову динаміку вартості, дозволяючи користувачу оцінити загальні тенденції ринку, виявити ключові етапи зростання чи спаду та визначити стійкість активу до ринкових коливань. Завдяки такому візуальному аналізу можна ідентифікувати патерни руху ціни та зробити попередні висновки щодо потенційної стабільності або ризиковості криптовалюти.

24-годинний свічковий графік обраної криптовалюти (рис. 3.29) демонструє короткострокові коливання ціни протягом останньої доби, використовуючи дискретні свічки, що відображають ціну відкриття, максимум, мінімум та закриття кожного інтервалу часу. Такий формат подання дозволяє детально аналізувати інтенсивність торгівлі, рівень волатильності та можливі торгові сигнали.

На основі цього графіка користувач може виявляти короткострокові тенденції, визначати точки входу та виходу з ринку.



Рисунок 3.28 – Трирічний історичний ціновий графік



Рисунок 3.29 – 24-годинний свічковий графік

На сторінці «Index» представлено різноманітні криптовалютні індекси, що відображають загальний стан та тенденції ринку. Система аналізу автоматично завантажує та оновлює дані з відкритих джерел, надаючи користувачу можливість відстежувати ключові показники ринкової активності.

Серед доступних індексів – BDI (Bitcoin Dominance Index), AMCI (Altcoin Market Capitalization Index), TCMC (Total Cryptocurrency Market Capitalization) та CFGI (Crypto Fear and Greed Index). Їхній комплексний аналіз дозволяє сформувати цілісне уявлення про стан ринку, баланс між Bitcoin та альткоїнами, а також емоційний настрій учасників ринку, що є важливим чинником при прийнятті інвестиційних рішень.

Індекс загальної ринкової капіталізації (TCMC) відображає сумарну вартість усіх криптовалют, показуючи загальний обсяг ринку та його зміни з часом (рис. 3.30). Таким чином, він служить агрегованим індикатором ризиків та інвестиційної привабливості галузі в цілому.

Індекс домінування Bitcoin (BDI) характеризує відносний вплив Bitcoin на ринок криптовалют, обчислюючи частку його ринкової капіталізації у відсотках від загальної капіталізації всіх цифрових активів (рис. 3.31). Аналіз динаміки цього показника дозволяє оцінити структурні зрушення ринку та визначити періоди посилення або ослаблення позицій Bitcoin відносно інших криптовалют.



Рисунок 3.30 – Total Cryptocurrency Market Cap Index (TCMC)

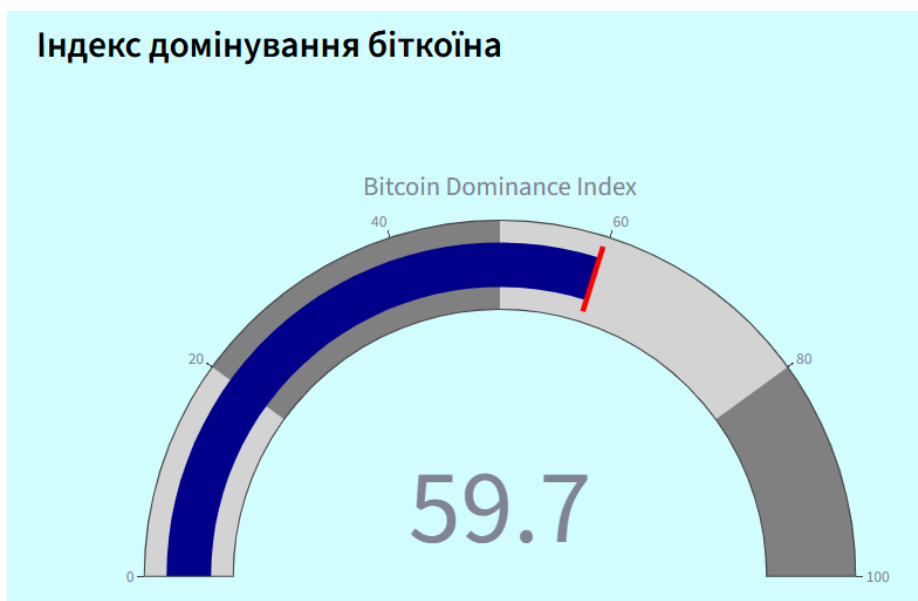


Рисунок 3.31 – Bitcoin Dominance Index (BDI)

Індекс ринкової капіталізації альткоїнів (AMCI) вимірює частку ринкової капіталізації всіх альткоїнів відносно загальної ринкової капіталізації криптовалютного ринку (рис. 3.32). Цей показник дає змогу оцінити ступінь диверсифікації ринку та виявити періоди зростання інтересу до альтернативних цифрових активів.

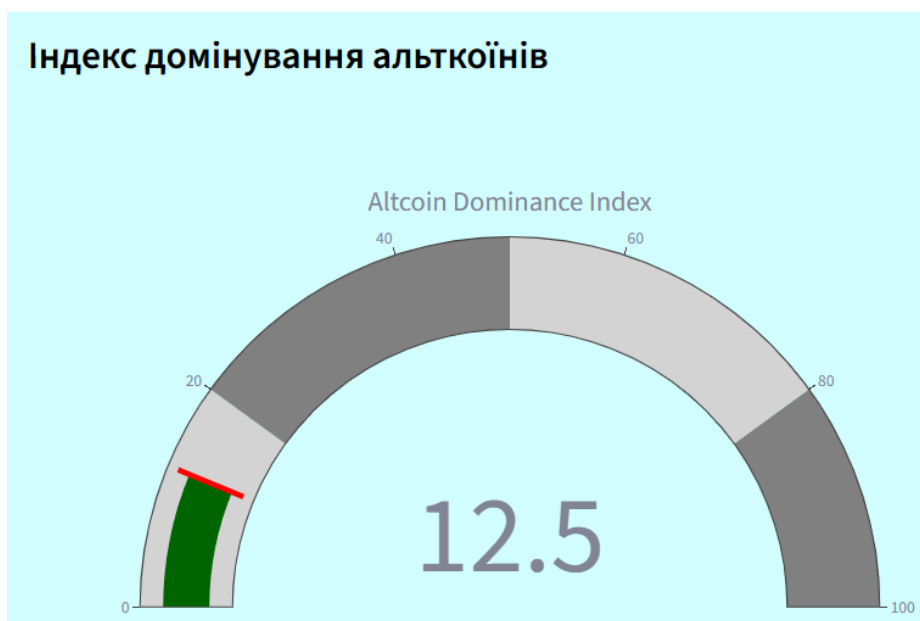


Рисунок 3.32 – Altcoin Market Cap Index (AMCI)

Індекс страху та жадібності криптовалют (CFGI) характеризує емоційний стан ринку, відображаючи співвідношення між панічними настроями (“страх”) та спекулятивною активністю (“жадібність”) серед інвесторів (рис. 3.33). Цей індекс є важливим індикатором психологічних факторів, що впливають на прийняття інвестиційних рішень.

Узагальнюючи, використання сукупності ринкових індексів – BDI, AMCI, TCMC та CFGI – дає змогу сформувати комплексне уявлення про поточний стан криптовалютного ринку, його структуру, емоційний фон та ступінь домінування окремих активів.



Рисунок 3.33 – Crypto Fear and Greed Index (CFGI)

На сторінці «Analysis» користувач має можливість обрати одну або кілька криптовалют для порівняння їхньої ціни та ринкової капіталізації. Такий підхід дає змогу здійснити попередній аналітичний огляд перед застосуванням технічних індикаторів для виявлення тенденцій розвитку криптовалютного ринку.

Функціонал порівняння цін реалізований через інтерактивні графіки, створені за допомогою бібліотеки Plotly (рис. 3.34). Користувач може обрати кілька криптовалют із переліку найпопулярніших і порівняти їх історичні дані, що дозволяє простежити динаміку цін закриття, періоди зростання та спаду, а також рівень волатильності активів.

Завдяки цьому інструменту користувач може не лише оцінити порівняльну поведінку ринкових активів, а й отримати базу для подальшого застосування технічних індикаторів і побудови моделей прогнозування на основі виявлених закономірностей.



Рисунок 3.34 – Порівняння цін

Для отримання даних щодо ринкової капіталізації з метою порівняльного аналізу система інтегрується з API платформи CoinMarketCap, за допомогою якого автоматично отримуються актуальні дані щодо капіталізації обраних криптовалют. Отримана інформація візуалізується у вигляді стовпчикової діаграми (рис. 3.35), що забезпечує наочне відображення ринкової ваги, фінансової сили та рівня популярності кожного цифрового активу.



Рисунок 3.35 – Порівняння ринкової капіталізації

Після попереднього порівняння динаміки цін і ринкової капіталізації користувач може перейти до застосування технічних індикаторів, що дають змогу здійснити глибший аналітичний огляд та визначити поточні ринкові тренди.

Серед індикаторів використовуються: RSI (Relative Strength Index), MACD (Moving Average Convergence Divergence), ковзні середні (MA) та лінії Боллінджера (Bollinger Bands). Їх поєднане використання дозволяє оцінити імпульс, силу та напрямок ринкового руху, а також виявити потенційні сигнали для входу чи виходу з ринку.

Індекс відносної сили (RSI) розраховується на основі історичних даних про ціни закриття криптовалют і відображає ступінь перекупленості або перепроданості активу. На рисунку 3.36, наведено приклад графіка RSI для криптовалюти ADA.

RSI є осцилятором моментуму, що коливається в діапазоні від 0 до 100 і використовується для оцінювання швидкості та амплітуди змін ціни.

Традиційно прийнято вважати:



- $RSI > 70$  – зона перекупленості, що може свідчити про ймовірну корекцію ціни вниз;
- $RSI < 30$  – зона недокупленості, яка може сигналізувати про недооціненість активу та потенційну можливість для купівлі.

Таким чином, аналіз RSI допомагає користувачу своєчасно визначати зміни ринкових настроїв і прогнозувати можливі точки розвороту тренду, підвищуючи точність прийняття рішень у рамках інтелектуальної системи аналізу криптовалютного ринку.

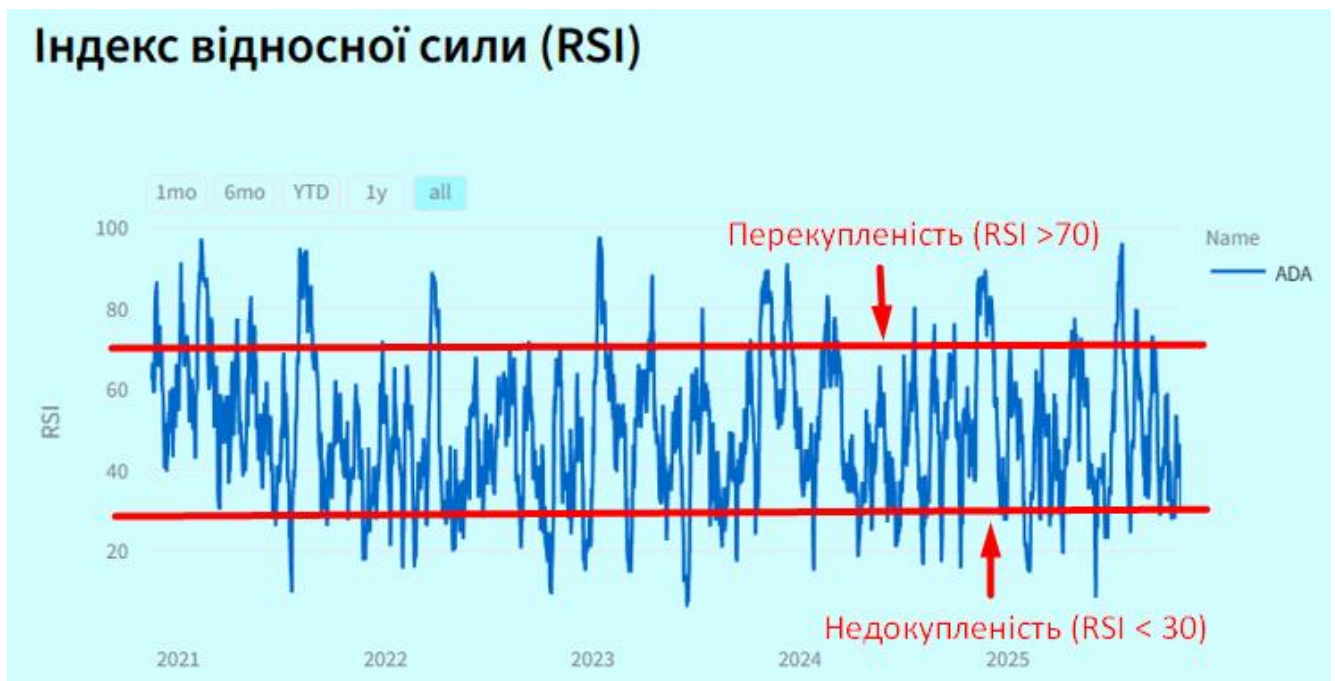


Рисунок 3.36 – Індекс відносної сили (RSI)

Індикатор MACD є одним із найпоширеніших інструментів технічного аналізу, який дозволяє визначати силу, напрямок і тривалість тренду.

MACD обчислюється як різниця між 12-денною та 26-денною експоненціальними ковзними середніми (EMA), а сигнальна лінія являє собою 9-денну EMA самого індикатора MACD.

На рисунку 3.37 наведено приклад графіка MACD для криптовалюти ADA за період із січня 2021 року по листопад 2025 року. Протягом цього проміжку

спостерігалися декілька значних коливань індикатора, що відображають зміну ринкових трендів та моментуму.

Зокрема, помітне зростання MACD у середині 2021 року могло слугувати сигналом для входу в позицію, тоді як різке падіння наприкінці 2022 року вказувало на можливість зниження ціни та доцільність фіксації прибутку або виходу з ринку.



Рисунок 3.37 – Індикатор конвергенції/дивергенції ковзних середніх (MACD)

На рисунку 3.38 зображено взаємодію лінії MACD та сигнальної лінії, що дає змогу визначати ключові торгові сигнали:

- сигнал купівлі – виникає, коли лінія MACD перетинає сигнальну лінію знизу вгору, що свідчить про посилення висхідного тренду. У цьому випадку короткострокова ЕМА (12-денна) перевищує довгострокову (26-денну);
- сигнал продажу – формується, коли лінія MACD перетинає сигнальну лінію зверху вниз, що зазвичай передбачає початок спадного руху ціни;
- сигнал зміни тренду – спостерігається, коли обидві лінії перетинають нульову вісь. Перетин нульової лінії знизу вгору зазвичай вказує на початок висхідного тренду, а перетин зверху вниз – на формування низхідного.

Таким чином, індикатор MACD забезпечує надійний інструмент для визначення моментів входу та виходу з ринку, а його аналіз у поєднанні з іншими технічними показниками дозволяє підвищити точність прогнозування динаміки цін у межах інтелектуальної системи.



Рисунок 3.38 – Виявлення трендів за допомогою індикатора MACD

Ковзні середні (МА) застосовуються для згладжування цінових коливань та визначення загального напрямку руху ринку. У застосунку використано 50-денну та 200-денну МА, де 50-денна показує середнє значення ціни активу за останні 50 днів, а 200-денна – за останні 200 днів.

На рисунку 3.39 представлено динаміку ковзних середніх для криптовалюти “ADA” за період з 2020 по 2025 рік.

Основні спостереження:

- початкове зростання: у 2020–2021 роках вартість ADA різко підвищилась, перевищивши позначку 2, що може свідчити про активне зростання інтересу інвесторів та позитивні очікування на ринку;
- зниження активності: після піку 2021 року спостерігалися коливання з поступовим спадом, що могло бути ознакою зменшення попиту та загальної невизначеності серед трейдерів;

– консолідація: наприкінці 2024 року МА показує стабілізацію цін нижче рівня 1, що вказує на період консолідації після значної волатильності.

Таким чином, аналіз МА дозволяє інвесторам ефективніше виявляти середньо та довгострокові тренди, своєчасно реагуючи на зміни ринкової динаміки.



Рисунок 3.39 – Індикатор ковзних середніх (МА)

Індикатор ліній Боллінджера (BB) є одним із ключових інструментів технічного аналізу, який використовується для оцінки волатильності ринку та визначення моментів перекупленості або перепроданості активу. Центральна лінія зазвичай є простою ковзною середньою (SMA), тоді як верхня та нижня лінії відображають стандартні відхилення від неї, що дозволяє виявляти межі коливань ціни.

На рисунку 3.40 наведено приклад застосування індикатора Боллінджера для активу ADA у період з 2020 по 2025 рік. З графіка видно, що періоди широких смуг вказують на високу волатильність, тоді як звуження смуг – на стабілізацію ринку, яка часто передує різкому ціновому руху.

Взаємодія ціни ADA із лініями BB демонструє типову поведінку активу:

– досягнення або перевищення верхньої лінії свідчить про можливу перекупленість, що може бути сигналом до продажу;

- зниження нижче нижньої лінії вказує на перепроданість, що може створювати можливість для покупки;
- звуження смуг часто передує сильному трендовому руху, сигналізуючи про зміну фази ринку.



Рисунок 3.40 – Індикатор лінії Боллінджера (BB)

У цілому, аналіз ліній Боллінджера для ADA показує, що актив протягом 2020–2025 років переживав значні коливання волатильності. Для трейдерів у 2025 році такий індикатор залишається цінним інструментом для виявлення потенційних точок входу й виходу з ринку, а також для прогнозування змін у динаміці цін криптовалют.

На сторінці «Prediction» користувач має змогу здійснювати прогнозування майбутньої вартості криптовалют із використанням моделей глибокого навчання. Інтерфейс реалізований у середовищі Streamlit, що забезпечує інтерактивну взаємодію з даними та візуалізацію результатів у режимі реального часу.

Після запуску сторінки користувач обирає криптовалюту зі списку (Bitcoin, Ethereum, Litecoin, XRP, Cardano, Polkadot) та вказує кількість днів, за які потрібно

завантажити історичні дані. Дані автоматично отримуються з API CoinGecko у форматі часових рядів, що містять часову мітку та значення ціни (рис. 3.41).



Рисунок 3.41 – Візуалізація початкових функцій на сторінці прогнозування

Для попереднього аналізу даних у застосунку передбачено можливість відображення ковзних середніх (SMA 50, SMA 200) та ліній Боллінджера (Bollinger Bands), які дають змогу оцінити волатильність ринку та виявити потенційні зони перекупленості або перепроданості активу. На графіку зображено реальні дані цін обраної криптовалюти за вказаний період із можливістю масштабування та перегляду динаміки в різних часових інтервалах (рис. 3.42).



Рисунок 3.42 – Візуалізація історичних даних з індикаторами SMA та BB

Для прогнозування цін у застосунку реалізовано чотири архітектури нейронних мереж:

- LSTM (Long Short-Term Memory) – базова рекурентна модель, здатна враховувати довгострокові залежності у часових рядах;
- GRU (Gated Recurrent Unit) – спрощена версія LSTM, що забезпечує швидшу збіжність при збереженні точності;
- BiLSTM (Bidirectional LSTM) – двонаправлена модель, яка враховує як минулі, так і майбутні залежності в послідовності;
- CNN+LSTM – комбінована модель, що поєднує згорткові шари (для виявлення локальних патернів) і рекурентні шари (для моделювання часових залежностей).

Користувач може налаштовувати параметри навчання, зокрема: розмір прихованого шару (hidden size), кількість шарів (layers), швидкість навчання (learning rate), кількість епох (epochs) та довжину вхідного вікна (seq\_length).

У бічній панелі застосунку (sidebar) передбачено інтерфейс для налаштування параметрів навчання нейронної мережі. Користувач може



самостійно обрати тип моделі (LSTM, GRU, BiLSTM або CNN+LSTM) із випадального списку та задати індивідуальні параметри для оптимізації процесу прогнозування (рис. 3.43).

Зокрема, можна регулювати такі параметри:

- розмір прихованого шару (Hidden size) – визначає кількість нейронів у прихованих шарах моделі, що впливає на її здатність запам'ятовувати та узагальнювати часові залежності;
- кількість шарів (Layers) – задає глибину нейронної мережі, тобто кількість послідовних рівнів обробки даних;
- швидкість навчання (Learning rate) – контролює, з яким кроком оновлюються ваги під час оптимізації, впливаючи на швидкість та стабільність збіжності;
- кількість епох (Epochs) – визначає, скільки разів модель проходить повний навчальний набір даних, поступово зменшуючи похибку прогнозу;
- довжину вхідного вікна (Sequence length) – кількість попередніх спостережень, які враховуються під час формування прогнозу.

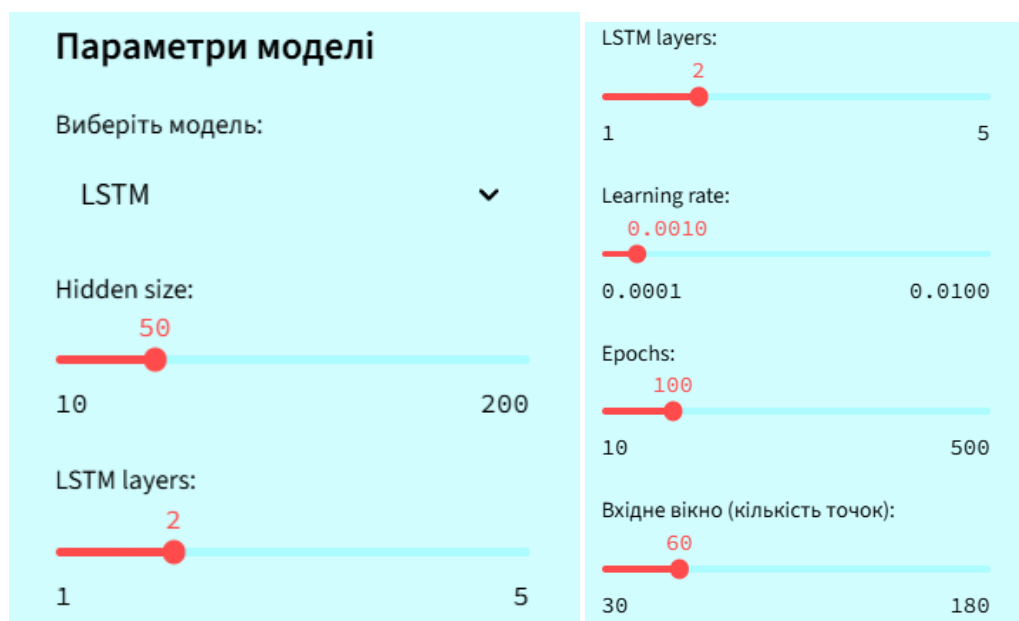


Рисунок 3.43 – Налаштування параметрів моделі



Після запуску навчання дані нормалізуються в діапазоні  $[0,1]$ , розділяються на навчальну та тестову множини у співвідношенні 80/20. Навчання проводиться на навчальній множині, а якість моделі оцінюється на тестовій. Втрата (Loss) виводиться на екран для контролю процесу збіжності моделі.

Після завершення навчання нейронна мережа переходить до етапу прогнозування, де на основі виявлених закономірностей у часових рядах формує очікувану динаміку цін обраної криптовалюти.

Отримані результати відображаються у вигляді інтерактивного лінійного графіка, який дозволяє наочно порівняти фактичні та передбачені значення.

Синя лінія демонструє реальні історичні дані – як змінювалася вартість криптовалюти в минулому.

Помаранчева лінія показує прогнозовану тенденцію на обраний період (1, 7, 14 або 28 днів), що дає змогу оцінити можливий напрямок руху ціни в майбутньому.

Візуалізація результатів моделі наведена на рисунку 3.44.



Рисунок 3.44 – Візуалізація результатів прогнозування (BTC)

Такий підхід дозволяє не лише спостерігати за минулою поведінкою активу, а й отримувати аналітичне бачення подальших коливань ринку.

Для оцінювання точності прогнозу в застосунку реалізовано розрахунок основних метрик, розрахунок яких наведено нижче.

*Mean Absolute Error/Mean Absolute Derivation (MAE/MAD)* – середнє абсолютне відхилення прогнозованих значень від фактичних:

$$MAE = MAD = \frac{1}{n} \sum_t^n |y_t - \hat{y}_t|, \quad (3.1)$$

де  $\hat{y}_t$  та  $y_t$  – прогнозоване та фактичне значення рівнів ряду.

Вважається, що показник MAE/MAD повинен бути меншим за 10% середнього значення фінансового часового ряду. Менше значення MAE свідчить про вищу точність прогнозу;

*Root Mean Squared Error (RMSE)* – середньоквадратична похибка, що характеризує загальну величину відхилень прогнозу від реальних даних і чутлива до наявності великих помилок:

$$RMSE = \frac{1}{n} \sum_t^n (y_t - \hat{y}_t)^2, \quad (3.2)$$

де  $\hat{y}_t$  та  $y_t$  – прогнозоване та фактичне значення рівнів ряду.

При налаштуванні параметрів нейронної мережі або порівнянні кількох моделей оптимальною вважається та, для якої значення RMSE є мінімальним;

*Mean Absolute Percentage Error (MAPE)* – середня абсолютна похибка у відсотках, що показує відносне відхилення прогнозу від фактичних значень:

$$MAPE = \frac{100}{n} \sum_{t=1}^n \left| \frac{y_t - \hat{y}_t}{y_t} \right|, \quad (3.3)$$

де  $\hat{y}_t$  та  $y_t$  – прогнозоване та фактичне значення рівнів ряду.

Інтерпретація значень MAPE:

– менше 10% – прогноз має дуже високу точність;

- 10–20% – хороша точність;
- 20–50% – задовільна точність;
- понад 50% – прогноз вважається неточним.

Після побудови графіка прогнозу використовується метод, який визначає частину фактичних даних, що відповідає обраному періоду прогнозування. Для оцінювання точності результатів моделі застосовуються наведені вище метрики MAE, RMSE та MAPE (рис. 3.45), обчислені згідно з відповідними формулами.

| Порівняння моделей |                |          |          |          |
|--------------------|----------------|----------|----------|----------|
|                    | Модель         | RMSE     | MAE      | MAPE (%) |
| 0                  | <b>LSTM</b>    | 1,343.24 | 1,027.95 | 1.01     |
| 1                  | ARIMA          | 4,178.13 | 3,275.29 | 3.24     |
| 2                  | SARIMA         | 4,125.21 | 3,212.26 | 3.18     |
| 3                  | Exp. Smoothing | 3,551.41 | 2,703.59 | 2.68     |

|   | Модель         | RMSE     | MAE      | MAPE (%) |
|---|----------------|----------|----------|----------|
| 0 | <b>GRU</b>     | 858.33   | 621.46   | 0.61     |
| 1 | ARIMA          | 4,178.13 | 3,275.29 | 3.24     |
| 2 | SARIMA         | 4,125.21 | 3,212.26 | 3.18     |
| 3 | Exp. Smoothing | 3,551.41 | 2,703.59 | 2.68     |

|   | Модель         | RMSE     | MAE      | MAPE (%) |
|---|----------------|----------|----------|----------|
| 0 | <b>BiLSTM</b>  | 1,128.79 | 845.70   | 0.83     |
| 1 | ARIMA          | 4,178.13 | 3,275.29 | 3.24     |
| 2 | SARIMA         | 4,125.21 | 3,212.26 | 3.18     |
| 3 | Exp. Smoothing | 3,551.41 | 2,703.59 | 2.68     |

|   | Модель          | RMSE     | MAE      | MAPE (%) |
|---|-----------------|----------|----------|----------|
| 0 | <b>CNN+LSTM</b> | 1,281.24 | 933.89   | 0.92     |
| 1 | ARIMA           | 4,178.13 | 3,275.29 | 3.24     |
| 2 | SARIMA          | 4,125.21 | 3,212.26 | 3.18     |
| 3 | Exp. Smoothing  | 3,551.41 | 2,703.59 | 2.68     |

Рисунок 3.45 – Оцінка якості прогнозу для моделей (LSTM, GRU, BiLSTM та CNN+LSTM) (BTC)

Отримані значення метрик MAE, RMSE та MAPE, наведені в таблиці 3.2, свідчать про задовільну якість прогнозування цін на Bitcoin за допомогою різних архітектур нейронних мереж. Крім того, для забезпечення комплексної оцінки ефективності прогнозування виконано порівняння результатів нейронних моделей із класичними статистичними підходами, зокрема ARIMA, SARIMA та Exponential Smoothing.

У таблиці 3.2 наведено результати порівняння якості прогнозування для різних моделей, горизонтів прогнозу та режимів вибірки даних. Зважаючи на високу волатильність криптовалютного ринку, показники MAE, RMSE та MAPE можуть істотно варіюватися.

Таблиця 3.2 – Порівняння метрик оцінки якості прогнозування за певний проміжок часу (для Bitcoin)

| Криптовалюта Bitcoin (BTC)                                     | Метрики якості прогнозів |          |      |
|----------------------------------------------------------------|--------------------------|----------|------|
| Hidden size: 50; Layers: 2; Learning rate: 0.001; Epochs: 200; |                          |          |      |
| 30 днів / 60 точок / 4 точки кожні 6 годин                     | MAE                      | RMSE     | MAPE |
| LSTM                                                           | 1,343.24                 | 1,027.95 | 1.01 |
| GRU                                                            | 858.33                   | 621.46   | 0.61 |
| BiLSTM                                                         | 1,128.79                 | 845.70   | 0.83 |
| CNN+LSTM                                                       | 1,281.24                 | 933.89   | 0.92 |
| ARIMA                                                          | 4,178.13                 | 3,275.29 | 3.24 |
| SARIMA                                                         | 4,125.21                 | 3,212.26 | 3.18 |
| Exp. Smoothing                                                 | 3,551.41                 | 2,703.59 | 2.68 |
| 60 днів / 80 точок / 2 точки кожні 12 годин                    | MAE                      | RMSE     | MAPE |
| LSTM                                                           | 2,875.89                 | 2,593.01 | 2.54 |
| GRU                                                            | 2,700.12                 | 2,450.33 | 2.42 |
| BiLSTM                                                         | 2,654.60                 | 2,395.30 | 2.35 |
| CNN+LSTM                                                       | 3,811.38                 | 3,485.09 | 3.42 |
| ARIMA                                                          | 4,160.81                 | 3,652.18 | 3.59 |
| SARIMA                                                         | 4,198.55                 | 3,693.91 | 3.63 |
| Exp. Smoothing                                                 | 3,589.36                 | 3,076.69 | 3.03 |

Аналіз отриманих результатів дозволяє виявити закономірності у поведінці моделей та визначити, які архітектури демонструють найкращу узгодженість між точністю прогнозу та стабільністю результатів.

Модель ARIMA застосовується для аналізу часових рядів, відображаючи взаємозв'язки між попередніми спостереженнями та дозволяючи моделювати тренди й циклічні коливання. Її розширення – SARIMA – додатково враховує сезонні компоненти, що робить її ефективною для даних із періодичними змінами. Метод експоненційного згладжування (Exponential Smoothing) надає більш плавні прогнози, поступово зменшуючи вагу старіших спостережень і знижуючи вплив короткострокових коливань.

Аналіз результатів дозволяє визначити наступні закономірності:

Розгляд результатів дозволяє виділити кілька ключових закономірностей. Для короткого та детального горизонту спостереження (30 днів / 4 точки на добу) спостерігається значно нижчий рівень похибки. Модель GRU демонструє найкращі результати серед нейронних мереж: MAE  $\approx$  858 доларів, RMSE  $\approx$  621 долар, MAPE  $\approx$  0,61%. LSTM та BiLSTM також показують високу точність, що підтверджує ефективність цих архітектур при короткостроковому прогнозуванні з високою частотою вибірки.

Класичні методи статистичного прогнозування (ARIMA, SARIMA, Exponential Smoothing) виявляються менш точними для короткого горизонту, із значно вищими значеннями похибок, що вказує на їх обмежену здатність відстежувати швидкі коливання цін.

Для довшого горизонту спостереження (60 днів / 2 точки на добу) похибки прогнозу закономірно зростають. MAE нейронних моделей підвищується до 2,654–2,875 доларів, RMSE – до 2,395–2,593 доларів, а MAPE – до 2,35–2,54%. У цьому сценарії BiLSTM виявилася найбільш стійкою, демонструючи найнижчі показники серед тестованих архітектур. Класичні моделі, особливо ARIMA та SARIMA, показали значно вищі значення похибок, що свідчить про їх обмежену здатність прогнозувати у довгостроковому періоді з рідшою вибіркою.

Загалом, для Bitcoin, зважаючи на середньодобові коливання ціни у кілька тисяч доларів, можна вважати прийнятними такі діапазони похибок:

- $MAE = 858\text{--}2,875$  доларів;
- $RMSE = 621\text{--}2,593$  доларів;
- $MAPE = 0,61\text{--}2,54\%$ .

Це підтверджує адекватність застосованих моделей для коротко- та середньострокового прогнозування, причому вибір оптимальної архітектури напряму залежить від обраного горизонту та частоти вибірки даних.

### **Висновки до розділу 3**

У даному розділі представлено процес проектування та програмної реалізації інтелектуальної системи прогнозування вартості криптовалют, що поєднує інструменти технічного аналізу, статистичного моделювання та методи глибинного навчання.

Розроблена система забезпечує автоматизований збір і обробку ринкових даних із відкритих джерел, зокрема API CoinMarketCap, Yahoo Finance та CoinGecko, що дозволяє формувати часові ряди цін і капіталізації провідних криптовалют. На основі цих даних реалізовано побудову графіків, індексів і технічних індикаторів, таких як RSI, MACD, ковзні середні (SMA) та лінії Боллінджера, що дає змогу користувачеві проводити детальний аналіз динаміки ринку.

Важливою складовою системи є модуль прогнозування, побудований на основі нейронних мереж глибинного навчання (LSTM, GRU, BiLSTM, CNN+LSTM). Ці архітектури дозволяють моделювати складні нелінійні залежності у часових рядах і виявляти довгострокові закономірності зміни цін. Завдяки попередній нормалізації, налаштуванню параметрів навчання та оцінюванню моделі за метриками MAE, RMSE та MAPE, система забезпечує високий рівень точності прогнозів як для коротко, так і для середньострокового періодів.

Окрім нейронних моделей, у системі реалізовано модулі класичного статистичного прогнозування – ARIMA, SARIMA та Exponential Smoothing, які забезпечують базове порівняння точності прогнозів. Модель ARIMA дозволяє моделювати тренди та циклічні коливання часових рядів, тоді як SARIMA додатково враховує сезонні компоненти. Метод експоненційного згладжування формує більш плавні прогнози, зменшуючи вплив короткострокових коливань. Використання цих підходів дає змогу оцінити ефективність нейронних моделей у контексті традиційних методів аналізу.

Інтеграція Streamlit дала змогу створити зручний користувацький інтерфейс із можливістю інтерактивного відображення історичних даних, результатів навчання моделі та прогнозованих значень у вигляді динамічних графіків.

Таким чином, розроблена інтелектуальна система дозволяє не лише проводити глибокий аналіз ринку криптовалют, але й прогнозувати подальші зміни їхньої вартості з використанням сучасних методів штучного інтелекту. Це сприяє підвищенню точності аналітичних оцінок, формуванню більш обґрунтованих інвестиційних рішень та мінімізації ризиків у високоволатильному середовищі крипторинку.

## ВИСНОВКИ

Ринок криптовалют посідає важливе місце у сучасній фінансовій екосистемі, виступаючи середовищем для обміну, торгівлі та інвестування цифрових активів. Його динамічний розвиток і висока волатильність зумовлюють потребу у застосуванні ефективних методів аналізу та прогнозування, які дають змогу виявляти ринкові закономірності, оцінювати ризики та приймати обґрунтовані інвестиційні рішення.

У ході роботи здійснено детальний аналіз сучасних програмних засобів і методологій, що використовуються для прогнозування цінових змін на криптовалютному ринку. Визначено основні групи інструментів, зокрема статистичні пакети (Python з бібліотеками pandas, numpy, scipy, matplotlib, pytorch), а також спеціалізовані сервіси CoinGecko та CoinMarketCap.

Попри широкий аналітичний потенціал, використання цих інструментів вимагає значного рівня технічної підготовки, розгортання середовищ і налаштування моделей, що ускладнює їх застосування для звичайних користувачів. Це обґрунтовує потребу у створенні інтелектуальної системи з інтуїтивно зрозумілим інтерфейсом, яка автоматизує всі етапи аналізу та прогнозування тенденцій ринку криптовалют.

Аналіз існуючих підходів засвідчив, що найбільш поширеними є три типи аналітичних методів: технічний, фундаментальний та соціальний аналіз.

Технічний аналіз базується на використанні математичних моделей і показників, таких як проста рухома середня (MA), індекс відносної сили (RSI), ковзна середня конвергенції/дивергенції (MACD) та смуги Боллінджера (Bollinger Bands).

Фундаментальний аналіз враховує економічні, фінансові та макроекономічні чинники, що формують ринкову вартість цифрових активів.

Соціальний аналіз, у свою чергу, спирається на аналіз даних із соціальних мереж, форумів і новинних джерел, що дає змогу відстежувати настрої інвесторів і



ринкові очікування. Поєднання цих методів забезпечує багатовимірний підхід до прогнозування, підвищуючи точність оцінок і надійність результатів.

Особливе значення для комплексного аналізу мають криптовалютні індекси, які узагальнюють динаміку ринку. До них належать показники ринкової капіталізації, торговельних обсягів, рівня волатильності, домінування біткоїна та індекси інвесторських настроїв, зокрема Crypto Fear & Greed Index (CFGI). Врахування таких показників у прогностичних моделях дає змогу формувати більш об'єктивні висновки щодо майбутніх трендів.

Для реалізації інтелектуальної системи аналізу та прогнозування тенденцій ринку криптовалют було обґрунтовано вибір середовища розробки Visual Studio Code та мови програмування Python, яка забезпечує широкий набір інструментів для аналітики й машинного навчання.

Інтерфейс системи розроблено з використанням фреймворку Streamlit, що гарантує інтерактивність і простоту у використанні.

Для візуалізації результатів застосовано Plotly Express, для збирання й опрацювання даних – Pandas, а обмін інформацією з API здійснюється за допомогою бібліотеки Requests.

Розроблена інтелектуальна система автоматично збирає ринкові дані з відкритих джерел (CoinMarketCap, Yahoo Finance), будує графіки коротко- та довгострокової динаміки, обчислює ключові індекси (BDI, AMCI, TCMC, CFGI) та технічні індикатори (RSI, MACD, MA, Bollinger Bands).

Крім того, у системі реалізовано нейронні мережі глибокого навчання типів LSTM, GRU, BiLSTM та CNN+LSTM, що дозволяють моделювати складні нелінійні взаємозв'язки та формувати прогнози майбутніх цін криптовалют. Для оцінювання точності прогнозів використовуються метрики MAE, RMSE і MAPE, які забезпечують об'єктивну перевірку якості моделей.

Також, у системі реалізовано модулі класичного статистичного прогнозування – ARIMA, SARIMA та Exponential Smoothing, які забезпечують базове порівняння точності прогнозів. Модель ARIMA дозволяє моделювати

тренди та циклічні коливання часових рядів, тоді як SARIMA додатково враховує сезонні компоненти. Метод експоненційного згладжування формує більш плавні прогнози, зменшуючи вплив короткострокових коливань. Використання цих підходів дає змогу оцінити ефективність нейронних моделей у контексті традиційних методів аналізу.

Розроблена система забезпечує користувачеві можливість здійснювати всебічний аналіз криптовалютного ринку, візуалізувати ключові тренди, формувати прогнози та приймати обґрунтовані інвестиційні рішення в умовах високої волатильності.

Виконані у магістерській роботі завдання реалізовані повністю. Надалі система може бути вдосконалена шляхом розширення функціоналу – додавання нових індикаторів, інтеграції додаткових джерел даних, застосування більш складних алгоритмів машинного навчання та оптимізації інтерфейсу для підвищення рівня автоматизації аналітичних процесів.

## ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Серажим В. В., Михайленко А. О. Аналіз волатильності цін на криптовалюту та її вплив на глобальні фінансові ринки / Актуальні проблеми економіки. Львів : ЛНУ ім. І. Франка, 2023. Т. 4, №2. С. 33–41.
2. Вінья П., Кейсі М. Епоха криптовалют. Як блокчейн та біткоїн змінюють світовий економічний порядок. 2018. 432 с.
3. Які бувають криптовалют і чим вони відрізняються : вебсайт. URL: <https://inventure.com.ua/uk/analytics/articles/yaki-buvayut-kriptovalyuti-i-chim-voni-vidriznyayutsya> (дата звернення: 12.10.2025).
4. Дідик А. М., Нечай О. Р. Порівняльний аналіз становлення та розвитку криптовалют у світі: уроки для України / Вісник СНУ. Луганськ : СНУ ім. В. Даля, 2024. №1. С. 58–67.
5. Casey M.J. The Age of Cryptocurrency: How Bitcoin and the Blockchain Are Challenging the Global Economic Order / M.J. Casey – London: St. Martin's Press, 2015. – 368 p.
6. Приходько Д. Криптовалюта. Навчальний посібник по роботі з цифровими активами. 2016. №7. С. 72-85.
7. Галушка Є. О., Пакон О. Д. Сутність криптовалют та перспективи їх розвитку. Молодий вчений. 2019. № 4, (44). Квітень. С. 634–638.
8. Що таке індекси криптовалют? : вебсайт. URL: <https://blog.trusteeglobal.com/ru/indeks-kriptovalyut-chto-eto-i-top-10-vazhnyh-kriptoindekov-2/> (дата звернення: 14.10.2025).
9. Долгіх А.О., Байбуз О.Г. Аналіз методів, моделей та програмних засобів прогнозування часових рядів. Відкриті інформаційні та комп'ютерні інтегровані технології. 2018. №79. С. 74-85.
10. Прогнозування та аналіз часових рядів : методичні вказівки / укл. Юрченко М. Є. Чернігів : ЧНТУ, 2018. 78 с.

11. Технічний аналіз криптовалютного ринку : вебсайт. URL: <https://www.binance.com/ru/blog/markets/как-проводит-технический-анализ-криптовалютного-рынка-1507239240565689107> (дата звернення: 15.10.2025).
12. Основи та методи технічного аналізу : вебсайт. URL: <https://articles.opexflow.com/uk/analysis-methods-and-tools/osnovy-i-metody-texnicheskogo-trejdinga.htm> (дата звернення: 16.10.2025).
13. Індикатори, що використовуються в технічному аналізі : вебсайт. URL: <https://academy.binance.com/uk/articles/5-essential-indicators-used-in-technical-analysis> (дата звернення: 18.10.2025).
14. Найкращі інструменти для початківців криптографів : вебсайт. URL: <https://cryptomus.com/uk/blog/best-crypto-market-checker-tools-for-tracking-and-analyzing-cryptocurrency-markets> (дата звернення: 19.10.2025).
15. Графічні моделі перелому тенденції : вебсайт. URL: [https://stud.com.ua/22256/finansii/grafichniy\\_analiz\\_tsinnih\\_paperv](https://stud.com.ua/22256/finansii/grafichniy_analiz_tsinnih_paperv) (дата звернення: 22.10.2025).
16. Harvey A. C. Time Series Models. 2nd Edition. Cambridge : The MIT Press, 1993. 308 p.
17. Visual Studio Code - Code Editing. Visual Studio Code : вебсайт. URL: <https://code.visualstudio.com/> (дата звернення: 28.10.2025).
18. Today's Cryptocurrency Prices by Market Cap. Coinmarketcap : вебсайт. URL: <https://coinmarketcap.com> (дата звернення: 29.10.2025).
19. Підручник з Python – Документація : вебсайт. URL: <https://docs.python.org/uk/3/tutorial/index.html> (дата звернення: 01.11.2025).
20. Структури даних в Python : вебсайт. URL: <https://programming.in.ua/programming/python/219-python-structure-of-data.html> (дата звернення: 02.11.2025).
21. Streamlit : вебсайт. URL: <https://streamlit.io> (дата звернення: 03.11.2025).
22. Python Tutorial – Streamlit : вебсайт. URL: <https://www.datacamp.com> (дата звернення: 03.11.2025).

23. Створення веб-програм для візуалізації даних за допомогою Streamlit : вебсайт. URL: <https://dev-gang.ru/article/sozdanie-vebprilozhenija-dlja-vizualizacii-dannyh-s-ispolzovaniem-pythonstreamlit-ruik5c1774/> (дата звернення: 05.11.2025).
24. Найпопулярніші бібліотеки Python : вебсайт. URL: <https://foxminded.ua/ru/biblioteki-python/> (дата звернення: 06.11.2025).
25. Pandas – Python Data Analysis Library: website. URL: <https://pandas.pydata.org> (дата звернення: 07.11.2025).
26. Plotly Open Source Graphing Library for Python : website. URL: <https://vitalflux.com/different-types-of-time-series-forecasting-models/> (дата звернення: 07.11.2025).
27. Yfinance Library – A Complete Guide : website. URL: <https://matplotlib.org> (дата звернення: 07.11.2025).
28. Python's Requests Library : website. URL: <https://realpython.com/python-requests/> (дата звернення: 08.11.2025).
29. Forecasting at Scale : website. URL: <https://peerj.com/preprints/3190.pdf> (дата звернення: 09.11.2025).
30. Основи API : вебсайт. URL: [https://cloud.itstep.org/blog\\_3/what-is-an-api-why-is-it-used-by-programmers-and-the-basics-of-working-with-it](https://cloud.itstep.org/blog_3/what-is-an-api-why-is-it-used-by-programmers-and-the-basics-of-working-with-it) (дата звернення: 09.11.2025).
31. Tradingview: website. URL: <https://www.tradingview.com/chart> (дата звернення: 09.11.2025).
32. RNN & LSTM & BI-LSTM & GRU: website. URL: <https://medium.com/@antoniosmalak14/rnn-lstm-bi-lstm-gru-cb5869660691> (дата звернення: 11.11.2025).
33. Андрусенко Ю.О. Аналіз основних моделей прогнозування часових рядів. Збірник наукових праць Харківського національного університету Повітряних сил. 2020. № 3(65). С. 91-96.

34. Different types of Time-series Forecasting Models. Data Analytics : website. URL: <https://vitalflux.com/different-types-of-time-series-forecasting-models/> (дата звернення: 11.11.2025).

35. The Data Scientist's Guide to Scaling: Standard, MinMax & Robust Methods : website. URL: <https://medium.com/@amitkharche14/the-data-scientists-guide-to-scaling-standard-minmax-robust-methods-25818b30c881> (дата звернення: 12.11.2025).

36. Guide to Time Series Data Pre-processing Methods : website. URL: <https://vitalflux.com/different-types-of-time-series-forecasting-models/> (дата звернення: 13.11.2025).

37. The Complete Guide to Time Series Models. Built-in. Data Science : website. URL: <https://builtin.com/data-science/time-series-model> (дата звернення: 14.11.2025).

38. Woodward W. A., Sadler B. P., Robertson S. Time Series for Data Science: Analysis and Forecasting. Boca Raton : Chapman and Hall/CRC, 2022. 506 p.

## ДОДАТОК А

### Головна сторінка “Home”

```
import streamlit as st

st.set_page_config(
    page_title="Home",
    page_icon="🏠",
    layout="centered",
    initial_sidebar_state="expanded"
)

# Фонове зображення та базовий стиль
st.markdown("""
<style>
    .stApp {
        background-image: linear-gradient(rgba(10, 25, 47, 0.85), rgba(10, 25, 47, 0.85)),
            url('https://psm7.com/wp-content/uploads/2022/12/FUz_joTXwAAwdN3.jpg');
        background-size: cover;
        background-position: center;
        background-attachment: fixed;
        color: white;
    }

    h1, h2, h3, p {
        font-family: 'Segoe UI', sans-serif;
    }

    .main-title {
        text-align: center;
        font-size: 38px;
        color: #ffffff;
        text-shadow: 0 0 15px #1E90FF;
        animation: fadeIn 2s ease-in-out;
    }

    @keyframes fadeIn {
        from {opacity: 0;}
        to {opacity: 1;}
    }

    .info-box {
        background: rgba(65, 105, 225, 0.85);
        border-radius: 25px;
        padding: 25px;
        margin: 20px auto;
        width: 85%;
        box-shadow: 0 0 25px rgba(0,0,0,0.4);
        backdrop-filter: blur(5px);
        transition: transform 0.3s ease;
    }

    .info-box:hover {
        transform: scale(1.03);
    }

    .section-title {
        color: #E0FFFF;
    }
""")
```

```

font-size: 24px;
text-align: center;
font-weight: 600;
}

.section-text {
color: #F0F8FF;
font-size: 18px;
line-height: 1.6;
text-align: center;
}

.emoji {
font-size: 26px;
}
</style>
"""', unsafe_allow_html=True)

# Заголовок
st.markdown("<h1 class='main-title'> Інтелектуальна система прогнозування вартості криптовалют</h1>",
unsafe_allow_html=True)

# Опис
st.markdown("""
<div class='info-box'>
<h2 class='section-title'> Про застосунок</h2>
<p class='section-text'>
Цей вебзастосунок на основі нейронних мереж надає користувачам інструменти для
аналізу, візуалізації та прогнозування вартості криптовалют.
</p>
</div>

<div class='info-box'>
<h2 class='section-title'> Основні сторінки</h2>
<p class='section-text'>
<b>«Details»</b> – інтерактивні графіки цін криптовалют.<br>
<b>«Index»</b> – динаміка ринкових індексів.<br>
<b>«Analysis»</b> – технічні індикатори для глибокого аналізу.<br>
<b>«Prediction»</b> – прогнозування вартості криптовалют за допомогою нейронних мереж.
</p>
</div>

<div class='info-box'>
<h2 class='section-title'> Мета</h2>
<p class='section-text'>
Система допомагає інвесторам та аналітикам ухвалювати зважені рішення,
використовуючи штучний інтелект і глибоке навчання для виявлення прихованих закономірностей у
фінансових даних.
</p>
</div>
"""', unsafe_allow_html=True)

```



## ДОДАТОК Б

### Сторінка “Details”

```
import streamlit as st
import plotly.graph_objs as go
import yfinance as yf
import requests

st.set_page_config(
    page_title="Details",
    page_icon="🔍",
    initial_sidebar_state="expanded"
)

sources = st.sidebar.expander('Джерело даних')
sources.write("[CoinMarketCap](https://coinmarketcap.com/uk/)")

class Crypto:
    def __init__(self, name, symbol):
        self.__name = name
        self.__symbol = symbol
        self.ticker = symbol + "-USD"
        self.initialize = yf.Ticker(self.ticker)
        self.data_1d = self.initialize.history(period="1d", interval="5m")
        self.data_3y = self.initialize.history(period="3y", interval="1d")

    def fetch_data(self):
        url = f"https://pro-api.coinmarketcap.com/v1/cryptocurrency/quotes/latest"
        parameters = {
            'symbol': self.__symbol
        }
        headers = {
            'Accepts': 'application/json',
            'X-CMC_PRO_API_KEY': '95fc1a6d-625d-4aae-9ecb-af3e92d8a300',
        }
        response = requests.get(url, headers=headers, params=parameters)
        data = response.json()
        return data['data'][self.__symbol]['quote']['USD']['price']

    def make_candlestick(self):
        fig = go.Figure()

        fig.add_trace(go.Candlestick(x=self.data_1d.index,
            open=self.data_1d['Open'],
            high=self.data_1d['High'],
            low=self.data_1d['Low'],
            close=self.data_1d['Close'], name = 'market data'))

        fig.update_layout(
            title=self.__name + ' - 24-годинний свічковий графік',
            yaxis_title=self.__name + ' Price (in US Dollars)'
        )

        fig.update_xaxes(
            rangelslider_visible=True,
            rangeselector=dict(
                buttons=list([
                    dict(count=15, label="15m", step="minute", stepmode="backward"),
```

```

        dict(count=45, label="45m", step="minute", stepmode="backward"),
        dict(count=1, label="HTD", step="hour", stepmode="todate"),
        dict(count=6, label="6h", step="hour", stepmode="backward"),
        dict(step="all")
    ])
)
)

cs = fig.data[0]

cs.increasing.fillcolor = '#3D9970'
cs.increasing.line.color = '#3D9970'
cs.decreasing.fillcolor = '#FF4136'
cs.decreasing.line.color = '#FF4136'

return fig

def make_line_price(self):
    df = self.data_3y.reset_index()
    for i in ['Open', 'High', 'Close', 'Low']:
        df[i] = df[i].astype('float64')

    fig = go.Figure([go.Scatter(x=df['Date'], y=df['High'])])

    fig.update_layout(
        title=self.__name + ' - Діаграма за 3 роки',
        yaxis_title=self.__name + ' Price (in US Dollars)'
    )

    fig.update_xaxes(
        rangeflider_visible=True,
        rangeselector=dict(
            buttons=list([
                dict(count=1, label="1mo", step="month", stepmode="backward"),
                dict(count=6, label="6mo", step="month", stepmode="backward"),
                dict(count=1, label="YTD", step="year", stepmode="todate"),
                dict(count=1, label="1y", step="year", stepmode="backward"),
                dict(step="all")
            ])
        )
    )

    return fig

def get_info(self):
    info = self.initialize.info
    description = info.get('longBusinessSummary')
    return description

st.markdown(
    """
    <div style='background-color: #4169E1; padding: 10px; border-radius: 100px; width: 100%;>
        <h2 style='color: white; text-align: center;'>Огляд криптовалют</h2>
    </div>
    """
    ,
    unsafe_allow_html=True
)

crypto_name = ["Bitcoin (BTC)", "Ethereum (ETH)", "XRP (XRP)", "Tether (USDT)", "Dogecoin (DOGE)", "Cardano (ADA)", "Polygon (MATIC)", "Binance Coin (BNB)", "USD Coin (USDC)", "Binance USD (BUSD)"]

```

```
selected = st.selectbox("-", crypto_name)

words = selected.split(" ")
crypto_name = " ".join(words[:-1])
crypto_symbol = words[-1].strip("(")

crypto_details = Crypto(crypto_name, crypto_symbol)
crypto_price = crypto_details.fetch_data()
crypto_candlestick = crypto_details.make_candlestick()
crypto_info = crypto_details.get_info()
crypto_line = crypto_details.make_line_price()

st.subheader(f"{crypto_name} ({crypto_symbol})")

st.write(f"Current Price: ${crypto_price}")
st.plotly_chart(crypto_line)
st.plotly_chart(crypto_candlestick)
```

## ДОДАТОК В

### Сторінка “Index”

```
import streamlit as st
import plotly.graph_objs as go
import requests

st.set_page_config(
    page_title="Index",
    page_icon="🔍",
    initial_sidebar_state="expanded"
)
sources = st.sidebar.expander('Джерело даних')
sources.write("[CoinMarketCap](https://coinmarketcap.com/uk/)")

class CryptoIndices:
    def __init__(self):
        self.cmc_url = "https://pro-api.coinmarketcap.com/v1/global-metrics/quotes/latest"
        self.fng_url = "https://api.alternative.me/fng/"
        self.headers_cmc = {
            'Accepts': 'application/json',
            'X-CMC_PRO_API_KEY': '95fc1a6d-625d-4aae-9ecb-af3e92d8a300',
        }

    def fetch_data(self):
        response_cmc = requests.get(self.cmc_url, headers=self.headers_cmc)
        data_cmc = response_cmc.json()
        bitcoin_dominance = data_cmc['data']['btc_dominance']
        altcoin_dominance = data_cmc['data']['eth_dominance']
        market_cap_index = data_cmc['data']['quote']['USD']['total_market_cap']

        response_fng = requests.get(self.fng_url)
        data_fng = response_fng.json()
        fng_index = int(data_fng['data'][0]['value'])

        return bitcoin_dominance, altcoin_dominance, market_cap_index, fng_index

crypto_indices = CryptoIndices()

bitcoin_dominance, altcoin_dominance, market_cap_index, fng_index = crypto_indices.fetch_data()
st.markdown(
    """
    <div style='background-color: #4169E1; padding: 10px; border-radius: 100px; width: 100%;>
        <h2 style='color: white; text-align: center;*>Індекси криптовалюти</h2>
    </div>
    """,
    unsafe_allow_html=True
)

st.subheader("Індекс ринкової капіталізації криптовалюти")
fig_market_cap_index = go.Figure(go.Indicator(
    mode="number",
    value=market_cap_index,
    title={'text': "Cryptocurrency Market Cap Index"}))
st.plotly_chart(fig_market_cap_index)

st.subheader("Індекс домінування біткоїна")
fig_bitcoin_dominance = go.Figure(go.Indicator(
```

2025 р.

```

mode="gauge+number",
value=bitcoin_dominance,
domain={'x': [0, 1], 'y': [0, 1]},
title={'text': "Bitcoin Dominance Index"},
gauge={'axis': {'range': [None, 100]},
      'bar': {'color': "darkblue"},
      'steps': [
        {'range': [0, 20], 'color': "lightgray"},
        {'range': [20, 50], 'color': "gray"},
        {'range': [50, 80], 'color': "lightgray"},
        {'range': [80, 100], 'color': "gray"}],
      'threshold': {'line': {'color': "red", 'width': 4}, 'thickness': 0.75, 'value': bitcoin_dominance}}))
st.plotly_chart(fig_bitcoin_dominance)

st.subheader("Індекс домінування альткоїнів")
fig_altcoin_dominance = go.Figure(go.Indicator(
  mode="gauge+number",
  value=altcoin_dominance,
  domain={'x': [0, 1], 'y': [0, 1]},
  title={'text': "Altcoin Dominance Index"},
  gauge={'axis': {'range': [None, 100]},
        'bar': {'color': "darkgreen"},
        'steps': [
          {'range': [0, 20], 'color': "lightgray"},
          {'range': [20, 50], 'color': "gray"},
          {'range': [50, 80], 'color': "lightgray"},
          {'range': [80, 100], 'color': "gray"}],
        'threshold': {'line': {'color': "red", 'width': 4}, 'thickness': 0.75, 'value': altcoin_dominance}}))
st.plotly_chart(fig_altcoin_dominance)

st.subheader("Індекс криптострахи та жадібності")
fig_fng_index = go.Figure(go.Indicator(
  mode="number",
  value=fng_index,
  title={'text': "Crypto Fear and Greed Index"},
  number={'suffix': ' / 100', 'valueformat': '.0f'},
  domain={'x': [0, 1], 'y': [0, 1]}
))
st.plotly_chart(fig_fng_index)

```

## ДОДАТОК Г

### Сторінка “Analysis” (Головні функції)

```
def calculate_rsi(data, window=14):
    delta = data["Close"].diff()
    gain = (delta.where(delta > 0, 0)).rolling(window=window).mean()
    loss = (-delta.where(delta < 0, 0)).rolling(window=window).mean()
    rs = gain / loss
    rsi = 100 - (100 / (1 + rs))
    return rsi

def calculate_macd(data, short_window=12, long_window=26):
    short_ema = data['Close'].ewm(span=short_window, min_periods=1, adjust=False).mean()
    long_ema = data['Close'].ewm(span=long_window, min_periods=1, adjust=False).mean()
    macd = short_ema - long_ema
    signal_line = macd.ewm(span=9, min_periods=1, adjust=False).mean()
    return macd, signal_line

def calculate_bollinger_bands(data, window=20):
    rolling_mean = data['Close'].rolling(window=window).mean()
    rolling_std = data['Close'].rolling(window=window).std()
    upper_band = rolling_mean + (rolling_std * 2)
    lower_band = rolling_mean - (rolling_std * 2)
    return rolling_mean, upper_band, lower_band

def calculate_ma(data, window=50):
    return data['Close'].rolling(window=window).mean()

for crypto in selected:
    crypto_data = data['data'][crypto]
    market_cap = crypto_data['quote']['USD']['market_cap']
    marketcap_list.append(market_cap)

    ticker = yf.Ticker(crypto + '-USD')
    hist = ticker.history(period='5y', interval='1d')
    hist['RSI'] = calculate_rsi(hist)
    hist['50MA'] = calculate_ma(hist, window=50)
    hist['200MA'] = calculate_ma(hist, window=200)
    hist['MACD'], hist['Signal Line'] = calculate_macd(hist)
    hist['Rolling Mean'], hist['Upper Band'], hist['Lower Band'] = calculate_bollinger_bands(hist)
    hist['Name'] = crypto
    dataframe_list.append(hist)
```

## ДОДАТОК Д

### Сторінка “Prediction” (Головні функції)

```
@st.cache_data(show_spinner=False)
def load_data(crypto_symbol, days):
    url = f'https://api.coingecko.com/api/v3/coins/{crypto_symbol}/market_chart?vs_currency=usd&days={days}'
    try:
        response = requests.get(url, timeout=10)
        response.raise_for_status()
        data = response.json()
        if 'prices' not in data:
            return None
        prices = data['prices']
        df = pd.DataFrame(prices, columns=['timestamp', 'price'])
        df['timestamp'] = pd.to_datetime(df['timestamp'], unit='ms')
        df.set_index('timestamp', inplace=True)
        return df
    except requests.exceptions.RequestException:
        return None

def preprocess_data(data):
    if data is None or data.empty:
        return None, None, None, None, None
    data = data.dropna().copy()
    prices = data['price'].values.reshape(-1, 1)
    scaler = MinMaxScaler(feature_range=(0, 1))
    scaled_data = scaler.fit_transform(prices)
    train_size = int(len(scaled_data) * 0.8)
    train_data = scaled_data[:train_size]
    test_data = scaled_data[train_size:]
    return train_data, test_data, scaler, train_size, prices

def create_sequences(data, seq_length=60):
    X, y = [], []
    for i in range(seq_length, len(data)):
        X.append(data[i-seq_length:i, 0])
        y.append(data[i, 0])
    return np.array(X), np.array(y)

class LSTMModel(nn.Module):
    def __init__(self, input_size, hidden_size, num_layers):
        super(LSTMModel, self).__init__()
        self.lstm = nn.LSTM(input_size, hidden_size, num_layers, batch_first=True)
        self.fc = nn.Linear(hidden_size, 1)
    def forward(self, x):
        out, _ = self.lstm(x)
        return self.fc(out[:, -1, :])

class GRUModel(nn.Module):
    def __init__(self, input_size, hidden_size, num_layers):
        super(GRUModel, self).__init__()
        self.gru = nn.GRU(input_size, hidden_size, num_layers, batch_first=True)
        self.fc = nn.Linear(hidden_size, 1)
    def forward(self, x):
        out, _ = self.gru(x)
        return self.fc(out[:, -1, :])

class BiLSTMModel(nn.Module):
    def __init__(self, input_size, hidden_size, num_layers):
        super(BiLSTMModel, self).__init__()
        self.lstm = nn.LSTM(input_size, hidden_size, num_layers, batch_first=True, bidirectional=True)
        self.fc = nn.Linear(hidden_size * 2, 1)
    def forward(self, x):
```

```

        out, _ = self.lstm(x)
        return self.fc(out[:, -1, :])
class CNLSTMMModel(nn.Module):
    def __init__(self, input_size, hidden_size, num_layers):
        super(CNLSTMMModel, self).__init__()
        self.conv1 = nn.Conv1d(in_channels=1, out_channels=32, kernel_size=3)
        self.lstm = nn.LSTM(32, hidden_size, num_layers, batch_first=True)
        self.fc = nn.Linear(hidden_size, 1)
    def forward(self, x):
        x = x.permute(0, 2, 1)
        x = torch.relu(self.conv1(x))
        x = x.permute(0, 2, 1)
        out, _ = self.lstm(x)
        return self.fc(out[:, -1, :])
def train_model(X_train, y_train, hidden_size, num_layers, learning_rate, num_epochs, model_type="LSTM"):
    if model_type == "LSTM":
        model = LSTMModel(1, hidden_size, num_layers)
    elif model_type == "GRU":
        model = GRUModel(1, hidden_size, num_layers)
    elif model_type == "BiLSTM":
        model = BiLSTMMModel(1, hidden_size, num_layers)
    elif model_type == "CNN+LSTM":
        model = CNLSTMMModel(1, hidden_size, num_layers)
    else:
        return None
    criterion = nn.MSELoss()
    optimizer = torch.optim.Adam(model.parameters(), lr=learning_rate)
    for epoch in range(num_epochs):
        model.train()
        optimizer.zero_grad()
        output = model(X_train)
        loss = criterion(output, y_train)
        loss.backward()
        optimizer.step()
    return model
def classical_models_predict(full_prices, train_size_idx, test_len, future_steps):
    train_prices = full_prices[:train_size_idx].flatten()
    test_prices = full_prices[train_size_idx:train_size_idx + test_len].flatten()
    try:
        arima_model = ARIMA(train_prices, order=(2, 1, 2)).fit()
        arima_future = arima_model.forecast(steps=future_steps)
    except:
        arima_future = np.zeros(future_steps)
    try:
        sarima_model = SARIMAX(train_prices, order=(1,1,1), seasonal_order=(0,0,0,0),
                                enforce_stationarity=False, enforce_invertibility=False).fit()
        sarima_future = sarima_model.forecast(steps=future_steps)
    except:
        sarima_future = np.zeros(future_steps)
    try:
        exp_model = ExponentialSmoothing(train_prices, trend="add").fit()
        exp_future = exp_model.forecast(future_steps)
    except:
        exp_future = np.zeros(future_steps)
    future_preds = {
        "ARIMA": arima_future,
        "SARIMA": sarima_future,
        "Exp. Smoothing": exp_future
    }
    return future_preds

```