

Group symbol: < >

Team: <team>

Project title: <Trading Bot>

Team members (*filled by PM, Team Leader*):

No	Name	Surname	Student ID	Role
1	Bohdan	Kyryliuk	267855	<i>PM, Team Leader</i>
2	Serhii	Ohurtsov	251530	<i>Team member</i>
3	Sergiy	Vergun	251203	<i>Team member</i>
4	Ilgin	Sogut	282416	<i>Team member</i>

1. Elaboration of application concept (F1)

1.1. Project (business) goals

Main Goal: Develop an efficient and user-friendly cryptocurrency trading bot that uses machine learning for predictive trading.

- Improve trade efficiency using machine learning models.
- Provide a seamless user interface for easy navigation and trading.
- Secure all transactions and user data.
- Allow detailed reporting and analysis for users.

1.2. Identification of project's internal and external Stakeholders

Identify the role of each **internal** stakeholder in the project, along with a textual description.

Symbol	Name	Role	Description

I1	Serhii	Documentation Specialist	Manages documentation, reports, and user manuals.
I2	Sergiy	Frontend Developer	Responsible for the design and maintenance of the user interface.
I3	Bohdan	Backend Developer	Handles the creation, optimization, and security of backend processes.
I4	Ilgın	ML Engineer	Leads the integration and tuning of machine learning models for predictive trading.

Identify the role of each **external** stakeholder in the project, along with a textual description.

Symbol	Name	Role	Description
E1	Users	End Users	Individuals who will use the trading bot for cryptocurrency trading to execute buy or sell orders.
E2	Investors	Financiers	Individuals or organizations that have funded or invested in the project to ensure its successful execution.
E3	Regulatory Bodies	Regulators	Authorities that oversee trading platforms' operations and ensure they adhere to laws and regulations.

1.3. Domain description

Phenomena and Abstracted Concepts:

Price Fluctuation: Cryptocurrency prices fluctuate constantly, influenced by a myriad of factors from global events to market sentiment.

Trade Execution: Trades (buying/selling of cryptocurrencies) occur when certain criteria are met.

Strategy Analysis: The bot constantly analyzes market data to determine the optimal trading strategy.

User Interactions: Users can log in, set trading parameters, start or stop the bot, and monitor performance.

Notifications: Users receive notifications on significant events like successful trades, large price drops, or bot errors.

Entities:

- **User:** An individual who has an account and uses the bot for trading.

Attributes: user_id, username, password, email, wallet_balance

Operations: login, logout, update_profile, change_password

- **TradingBot**: The automated system that makes trades on behalf of a user.

Attributes: bot_id, strategy, is_active, start_time, end_time

Operations: start_bot, stop_bot, set_strategy, get_performance_metrics

- **Trade**: Represents an action of buying or selling cryptocurrency.

Attributes: trade_id, trade_timestamp, trade_type, trade_amount, trade_price

Operations: execute_trade, cancel_trade

- **CurrencyPair**: Represents a trading pair, e.g., BTC/ETH.

Attributes: pair_id, base_currency, quote_currency, current_price

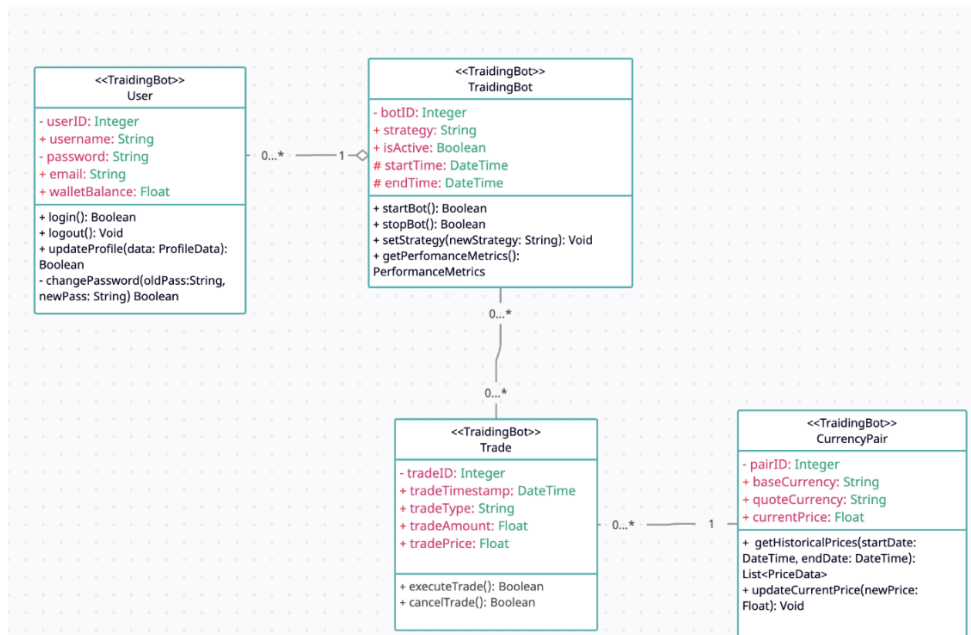
Operations: get_historical_prices, update_current_price

Relationships:

1. **User to TradingBot**: One-to-Many. A user can have multiple trading bots but each bot belongs to one user.
2. **TradingBot to Trade**: One-to-Many. A bot can execute many trades but each trade is made by one bot.
3. **Trade to CurrencyPair**: Many-to-One. Multiple trades can happen for a single currency pair, but each trade is made for one currency pair.

Events:

1. **Bot Activation**: When a user starts their trading bot.
2. **Bot Deactivation**: When a user stops their trading bot.
3. **Trade Execution**: When a bot executes a buy or sell order.
4. **Strategy Update**: When a bot's trading strategy is changed.
5. **Balance Alert**: When a user's wallet balance goes above or below a certain threshold.
6. **Error Notification**: When the bot encounters an error or issue.



1.4. Project schedule (Gantt chart)

Time Interval	Task	Assigned To (Symbol)	Status	10.10	17.10	24.10	31.10	7.11	14.11	21.11	28.11	5.12	12.12	19.12	26.12	2.1	9.1	16.1	23.1	30.1
10.10-17.10	Requirement Gathering and Analysis	I1, I2, I3, I4	Done																	
10.10-24.10	Design Frontend Mockups	I2 (Sergiy)	Ongoing																	
17.10-31.10	Backend Architecture Design	I3 (Bohdan)	Ongoing																	
24.10-07.11	Neural Network Initial Concept & Research	I4 (Ilgin)	Ongoing																	
24.10-10.11	Frontend Development (Phase 1)	I2 (Sergiy)	Pending																	
31.10-14.11	Backend API Development	I3 (Bohdan)	Pending																	
07.11-21.11	Neural Network Initialization, Training	I4 (Ilgin)	Pending																	
14.11-28.11	Integration of Frontend & Backend	I2 (Sergiy), I3 (Bohdan)	Pending																	
21.11-05.12	Initial Testing & Debugging	I2, I3, I4	Pending																	
05.12-19.12	AI integration with the app	I4 (Ilgin), I3 (Bohdan)	Pending																	
12.12-19.12	Documentation Drafting (Weekly Reports)	I1 (Serhii)	Pending																	
19.12-02.01	User Acceptance Testing	I2 (Sergiy)	Pending																	
02.01-16.01	Finalization & Optimization	I2, I3, I4	Pending																	
09.01-16.01	Documentation Finalization	I1 (Serhii)	Pending																	
16.01-23.01	Final Review & Deployment	I2, I3, I4	Pending																	
23.01-30.01	Post-Deployment Monitoring & Support	I3 (Bohdan), I4 (Ilgin)	Pending																	
Weekly	Weekly Reports	I1 (Serhii)	Ongoing																	

1.5. Identification of existing or alternative solutions

1)Competitor: 3Commas

Main Features:

- Portfolio tracking and management
- Analytics of trading performance
- Mobile app with instant notifications

Advantages:

- User-friendly interface
- High success rate in trade predictions
- Diverse cryptocurrency portfolio

Disadvantages:

- No mobile support
- No manual trading mode
- Limited customization options

2)Competitor: Coinrule

Main Features:

- Any Coin Scanner
- AI-Based Trading Bot Marketplace
- Beginner-Friendly UI
- TradingView Integration

Advantages:

- Allows users to define their own trading logic
- Educational Resources
- Leverage Trading

Disadvantages:

- Pricing(30-450\$)
- No mobile app

3)Competitor: NAGA

Main Features:

- Machine learning-based insights
- Social trading (follow top traders)

- Risk management tools

Advantages:

- Deposit via bank, debit and credit cards
- Users can copy trades from successful traders
- Up to 1,000x leveraged trading.

Disadvantages:

- Requires a high minimum deposit
- No mobile app

1.6. Project context

Application Context:

- **Purpose:** The primary purpose of the bot is to automate cryptocurrency trading based on specific algorithms and machine learning models to maximize profit.
- **User Demographics:** Targeted at both novice and professional cryptocurrency traders.
- **Environment:** Designed to operate 24/7, given the non-stop nature of the cryptocurrency markets.

Technological Context:

- **Platform:** Cloud-based solution ensuring high availability and scalability.
- **Integration:** Capable of integrating with multiple cryptocurrency exchanges via their APIs.
- **Security:** Implementing state-of-the-art encryption and security protocols to safeguard user data and assets.

Organisational Context:

- **Development Team:** A cross-functional team including frontend, backend, machine learning specialists, and documentation experts.
- **Stakeholders:** Internal stakeholders like developers, management, and finance teams. External stakeholders include end-users, regulatory bodies, and potential investors.
- **Project Management:** Agile development methodology with weekly sprints and regular reporting.

Legal Context:

- **Regulation:** As cryptocurrency regulations can vary by region, the bot must adhere to the specific legal framework of each target market.
- **Data Protection:** Comply with global data protection regulations, such as GDPR, to ensure user data privacy.
- **Licensing:** Ensure the proper licensing of all third-party software and libraries used in the bot's development.

Important Aspects for Product Development:

1. **Scalability:** The ability to handle an increasing number of users and trades without compromising performance.

2. **Security:** Safeguarding user assets and data should be a top priority given the financial nature of the application.
3. **Regulatory Compliance:** Keeping abreast of the evolving legal landscape for cryptocurrencies and ensuring the bot remains compliant.
4. **User Experience:** Ensuring the bot is user-friendly and caters to both novice and experienced traders.
5. **Integration Flexibility:** The capability to integrate with new exchanges or platforms as the crypto landscape evolves.

1.7. Technologies used in the project

1. Technology: Spring boot

Description: Spring Boot is a Java-based framework designed for the development of robust and scalable web applications and microservices.

Justification: Selected for its streamlined configuration and convention-over-configuration approach, Spring Boot empowers developers to quickly create production-ready applications with minimal setup, reducing development time and effort.

Key Responsibilities: Backend development; serving as the foundational framework for building the core logic and business functionality of various software applications, from web services to full-fledged web applications.

Link: <https://spring.io/>

2. Technology: Flutter

Description: Flutter is an open-source framework developed by Google for building natively compiled applications for mobile, web, and desktop from a single codebase.

Justification: Chosen for its fast development cycle, expressive UI components, and cross-platform capabilities, Flutter enables developers to create visually stunning and performant applications while maintaining code reusability across different platforms.

Key Responsibilities: Multi-platform app development; powering the creation of immersive and visually appealing user interfaces and experiences for mobile devices, web browsers, and desktop applications.

Link: [Flutter - Build apps for any screen](#)

3. Technology: Python

Description: Python is a versatile and widely-used programming language renowned for its prominence in the field of machine learning and data science.

Justification: Selected for its extensive libraries and frameworks, such as TensorFlow, PyTorch, and scikit-learn, Python provides a rich ecosystem for machine learning, making it the go-to choice for developing, training, and deploying machine learning models.

Key Responsibilities: Machine learning development; serving as the primary language for data preprocessing, model training, evaluation, and deployment, Python plays a crucial role in developing intelligent applications and predictive analytics.4. Technology: MongoDB

Link: [Welcome to Python.org](https://www.python.org/)

4. Technology: PostgreSQL

PostgreSQL is a powerful, open-source relational database management system (RDBMS) known for its robustness, extensibility, and advanced data management capabilities.

Justification: Chosen for its reliability and scalability, PostgreSQL is the preferred database solution for a wide range of applications, from small-scale projects to enterprise-level systems. It offers features such as ACID compliance, support for complex data types, and a thriving community, making it an ideal choice for data-centric applications.

Key Responsibilities: Database management; serving as the backend data store for various applications, PostgreSQL efficiently stores, retrieves, and manages structured data, ensuring data integrity and security

Link: <https://www.postgresql.org/>

5. Technology: Docker

Description: A platform used to develop, ship, and run applications inside containers.

Justification: Provides consistency across multiple development and production environments, ensuring the app runs the same everywhere.

Key Responsibilities: Containerization; encapsulating the application and its dependencies into a 'container' to ensure consistency across environments.

Link: [Official Docker Documentation](https://docs.docker.com/)

6. Technology: WebSocket

Description: A protocol providing full-duplex communication channels over a single TCP connection.

Justification: Allows for real-time data transfer, which is crucial for a trading bot that needs instantaneous market updates.

Key Responsibilities: Real-time communication; providing live trading data updates and notifications to users.

Link: [WebSocket Explanation](https://www.websocket.org/)

