System Requirements

03 : Autonomous Trading Bot

<team member names & ids>

Student ID	Name
23100197	Ahmed Tahir Shekhani
23100011	Suleman Mahmood
23100198	Ali Asghar
23100176	Syed Talal Hasan

Table of Contents

1. Introduction

A web application with an autonomous trading bot instance that will trade to generate profitable returns on stocks. The bot will be trained on the PSX data. The bot's decision will be based on the concepts of game theory, mathematical models, financial techniques, and especially artificial intelligence. The primary web app will allow the user to provide the bot's configuration, which includes, target return, risk appetite, and duration of the instance.

With recent advancements in deep learning frameworks and access to faster gpus, training complex models that can predict on time series data has opened new avenues to explore stock market trading. We plan on using models that that have a memory component in them, such as LSTM (Long Short Term Memory) to make predictions and trades on the stock market.

The overall objective for the application would be to achieve the return target provided by the analyst while configuring the bot and minimize loss according to the risk factor provided. The potential users of this application would be trade analysts or managers who will use the bot to run its instances according to their requirements.

Technical details:
The project's tech stack would be Next JS for web application UI, mainly front end, Flask for backend server, and Postgresql for persistent storage. The application would follow three-tier architecture with a repository pattern for the persistent layer, models, and command layer for modifying the state.

2. System Actors

Actor Name	Description
Analysts	Analysts will configure and start the execution of the bot, analyze it's statistics and assign investors to their bots.
Investors	Investors will use the bot for investment purposes and can view a summary report of the performance of the bot for the a specific range of time period (monthly, quarterly, half-yearly, yearly, or other combinations of days or months)

3. Functional Requirements

Requirements	
Sr#	**Requirement**
1	As an analyst, I want to be able to login with my credentials.
2	As an analyst, I want to be able to view recent highlights on my dashboard.
3	As an analyst, I want to be able to configure the parameters (such as balance, maximum drawdown, list of stocks) on which the bot will execute before it's executed.
4	As an analyst, I want to be able to set stopping parameters (target percentage return, target balance return, duration) of the bot.
5	As an analyst, I want to be able to initiate the execution of the bot.
6	As an analyst, I want to be able to forcefully stop the execution of the bot.
7	As an analyst, I want to be able to instantiate new instances of the bot based on different configurations.
8	As an analyst, I want to be able to register an investor who can then view their bot's performance.
9	As an analyst, I want to get login credentials for an investor after registering them.
10	As an analyst, I want to be able to assign one or more bots to an investor.
11	As an analyst, I want to view the performance of all my bots that are currently executing.
12	As an analyst, I want to view the entire history of all my bots that I ran.
13	As an analyst, I want to pick any bot in history and view its entire performance.
14	As an analyst, I want to view multiple graphs of a bot such as balance over time, number of trades over time.
15	As an analyst, I want to choose which ML models to use for a bot.
16	As an analyst, I want to choose a list of indicators to use for a bot.
17	As an analyst, I want to configure the parameters of my chosen indicators for the bot.
18	As an investor, I want to be able to login to my dashboard with the credentials provided by my analyst.
19	As an investor, I want to be able to see various graphs for my portfolio invested.
20	As an investor, I want to be able to see statistics of my bot's performance.

4. Non-functional Requirements / Quality Attributes

Sr#	Requirements
1	The system should complete execution of any request within 1 minute.
2	The system should entirely and accurately respond to requests at least 90% of the time it is invoked.
3	The time to get current stock prices from pakistan stock exchange should not be more than 5 seconds
4	The system should be available from Monday to Friday 9 am to 5 pm.
5	The bot should ping the trade instance every 1 minute to get the decision
6	The personal data of the investors and analysts will be stored in encrypted form
7	The trade profits, invested amounts, buying price, and selling will be in encrypted form as they are sensitive
8	Passwords should be salted and hashed before storage.
9	Multiple running bots will still ensure the individual performance of decision-making and request handling is catered within 60 seconds.
10	There must be at least 5 unique stock's pairs where the bots can trade

5. Security Requirements

Sr #	Security Risks	Potential Losses	Controls
1	Broken Access Control	- Unauthorized accessed users can use the amount from the main user's wallet for investment with high risk and might result in loss. - Access of API with missing access controls for Post, Put, and Delete result in loss of data for reporting and misconfiguration of the bot as unauthorized users can manipulate the saved configuration of the bot in the application.	- Allow users to access the records associated with their unique ID. - Log access failures and alert admin if failed multiple times. - Use stateless JWT with short duration
2	Cryptographic Failures	- Unauthorized users can steal the session cookie by intercepting requests. He can then replay this session cookie to access or modify the bot by hijacking the session. - Unauthorized users can gain access to the password	- Enforce TLS for all pages and require strong encryption. - Use complicated hash methods which should then further be salted so even if someone gains access to the password database, they can't make use of it.

		database through a file upload flaw and can gain access to unsalted hashes, which can be exposed with pre-calculated hashes from a rainbow table.	
3	Security Misconfiguration	- Unauthorized users may access the machine learning model and make the same predictions as the hedge fund manager, resulting in lower financial profits. - A misconfigured database of user credentials can lead to malicious users to gain access to the main application and its prediction model which can be misconfigured to return false predictions, resulting in huge losses for the hedge fund.	- Segmented Application with isolation of the model from the main components that requires special security access. - Remove unnecessary features and port access.

Sr#	Security Tool Name	Brief description (why the tool is suitable for your project)
1	Mend	- Can implement using CI/CD - It provides security over 70 CWE types including OWASP top 10 security issues. - It is easy to use, setup and administer - Since, our application is based on real time data management so fast scanning will be required for testing and for that mend is suitable as it is typically 10 times faster than other static security scanning tools.
2	Indusface WAS	- Uses globally accepted best practices. - Can be used for fully managed risk detection for the application.

6. Who Did What?

Name of the Team Member	Tasks done
Suleman Mahmood	Functional requirements (3), Actors (2)
Ahmed Tahir Shekhani	Security requirements (5), Introduction (1)
Talal	Security requirements (5), Introduction (1)
Ali	Non-functional requirements (4), Security requirements (5)

Note: Numbers in (x) means the section

7. Review checklist

Section Title	Reviewer Name(s)
Introduction	Ali Asghar
Actors	Ahmed Tahir Shekhani
Functional Requirements	Ali Asghar, Ahmed Tahir Shekhani
Non-functional requirements	Talal
Security Requirements	Suleman Mahmood