Practical No:02

Aim: To create an SRS for the AI Stock Analyzer project.

Project Title: AI Stock Analyzer
Date:
Prepared by: Atharva Bodade & Ashish Ingle

1. INTRODUCTION

1.1 Purpose

This document details the software requirements for the AI Stock Analyzer, an intelligent platform designed to assist investors and traders in analyzing stock market data using artificial intelligence and machine learning techniques. The primary audience includes developers, testers, product managers, financial analysts, and end-users such as traders and investment advisors.

1.2 Scope

The AI Stock Analyzer aims to automate and enhance stock market analysis, prediction, and reporting. It will provide users with actionable insights by leveraging real-time and historical financial data, predictive modeling, and intuitive visualization.

Objectives:

- Deliver accurate, AI-powered stock trends and price predictions.
- Enable robust portfolio and stock screening tools.
- Assist in risk management and informed investment decisions.
- Support real-time and scheduled alerting for critical events.

Benefits:

- Lowers research and decision-making time for investors.
- Reduces human error and emotional bias in stock selection.
- Democratizes advanced analytics for non-expert users.

1.3 Definitions, Acronyms, and Abbreviations

- **AI:** Artificial Intelligence

- **ML:** Machine Learning
- **UI:** User Interface
- **API:** Application Programming Interface
- **ROI:** Return on Investment

1.4 References

- IEEE SRS Standard (IEEE 830-1998)
- Yahoo Finance API documentation
- Alpha Vantage API documentation
- Internal Financial Informatics Policy (2025)

1.5 Overview

This SRS follows a schema introducing the system, its requirements, user classes, technical stack, constraints, and essential system diagrams.

2. OVERALL DESCRIPTION

2.1 Product Perspective

The AI Stock Analyzer is a web-based analytical tool, operable as a standalone platform or integrated with online brokerages and financial data providers via APIs for enhanced capabilities.

2.2 Product Functions

- Smart stock search and symbol screening.
- Historical market data aggregation.
- Predictive analytics using advanced AI/ML models.
- Visualization dashboards (price trends, volatility, portfolio analytics).
- Customizable notification/alert system.
- Report generation and export functionality.

2.3 User Classes and Characteristics

- **Investor/Trader:** Access forecasts, screen stocks, set up alerts, and visualize portfolio performance.

- **Administrator:** Manage users, oversee models, maintain system health, and audit system activity.

- **Analyst:** Conduct deeper analytics, customize prediction parameters, and export results.

- **Guest:** Limited access to market overview and sample visualizations.

2.4 Operating Environment

- **Hardware:** Modern personal computer, tablet, or smartphone.

- **Software:** Latest web browsers (Chrome, Edge, Firefox, Safari).

- **Network:** Stable internet access required.

- **Database:** PostgreSQL/MongoDB.

- **Backend:** Python (Flask/FastAPI), Node.js.

- **Frontend:** React or Angular with D3.js for visualization.

2.5 Design and Implementation Constraints

- Must be built with open-source, scalable technologies.

- All sensitive data must be encrypted in storage and transit.

- Compliance with relevant financial data privacy standards (e.g., GDPR).

- Dependency on third-party API/data provider availability.

2.6 User Documentation

- Step-by-step user manual (PDF/Online).

- Quick-start guides and in-app walkthroughs.

- FAQ and customer support documentation.

- Glossary of financial and AI/ML terms.

2.7 Assumptions and Dependencies

- Users have at least basic knowledge of financial markets.
- External APIs will provide timely and accurate data.
- There is institutional support for ongoing hosting and maintenance.

3. SPECIFIC REQUIREMENTS

3.1 Functional Requirements

- **FR1:** System will register users via email or single sign-on.
- **FR2:** Authenticate users with hashed credentials.
- **FR3:** Allow search and selection of stocks by symbol or company name.
- **FR4:** Aggregate and display real-time and historical stock data.
- **FR5:** Apply AI models to forecast price trends and recommend buy/sell/hold actions.
- **FR6:** Enable users to set up alerts for price or trend thresholds.
- **FR7:** Visualize predictions and historical data in interactive charts.
- **FR8:** Permit report generation/export (PDF/CSV).
- **FR9:** Allow admin to manage AI models, users, and logs.
- **FR10:** Integrate with external APIs for market and news data.

3.2 Non-functional Requirements

- **Performance:** Average interface response time under 2 seconds.
- **Security:** Enforce encryption of data, RBAC for admin tools, audit trails.
- **Usability:** Modern, responsive UI optimized for both desktop and mobile.
- **Reliability:** ≥99% uptime and robust error handling.
- **Scalability:** Should handle thousands of simultaneous users and data requests.
- **Maintainability:** Modular, well-documented codebase to simplify updates and debugging.

3.3 Interface Requirements

User Interfaces:

- Login/registration page
- Personalized dashboard
- Stock analysis workspace
- Interactive prediction graphs
- Alerts management
- System administration panel

Hardware Interfaces:

- No special hardware required; optional integration with biometric access (future scope).

Software Interfaces:

- Support for PostgreSQL/MongoDB database.
- Integration with third-party stock data APIs.
- Alerting via email, SMS, or in-app notifications.

Communication Interfaces:

- Secure HTTP/HTTPS protocols.
- RESTful API endpoints for system and third-party integrations.

4. APPENDICES

4.1 Diagrams

- **System Use Case Diagram:** Illustrates users, admins, and their primary system interactions (e.g., analyze, receive alerts, manage users).

- **Entity-Relationship (ER) Diagram:** Depicts entities such as User, Stock, Prediction, Alert, and their interrelations.

- **Data Flow Diagram (DFD):** Maps major processes (user input, data fetch, ML prediction, alerting), data stores, and external systems

- **Sample Model Architecture/Workflow:** Shows the end-to-end data pipeline: real-time data acquisition → AI/ML preprocessing → predictive modeling → visualization/output.

Conclusion:
This SRS provides a comprehensive blueprint for the clear and systematic development of the AI Stock Analyzer platform. Each section details requirements, constraints, and technical context to ensure successful, secure, and user-focused delivery of the software.