Software Requirements Specification (SRS)

AI-powered Forex Trading Analysis System

Team: Saad Nawaz, Shahbaz Ali, Hafsa Gul
Course: CSC-225 — Software Engineering
Department of Computer Science, Namal University, Mianwali

Submission Date: December 28, 2025

Contents

1 Introduction

1.1 Purpose

The purpose of this Software Requirements Specification (SRS) is to provide a complete, unambiguous, and verifiable specification of the requirements for the **AI-powered Forex Trading Analysis System**. This document explains the system's functionalities and constraints and serves as a reference for students, developers, and evaluators to understand the system requirements and to guide development, testing, and evaluation.

1.2 Scope

The system is designed to support Forex traders by analyzing data across multiple timeframes and summarizing relevant fundamental (economic) events to produce AI-based directional suggestions with clear explanations. It is a decision-support prototype and does not execute trades automatically. The system focuses on integrating Forex price data and economic events, applying technical indicators, generating explainable analysis results, and presenting them through a web-based interface where users can customize indicators and export reports(CSV formts). Administrative functions for system monitoring and user management are also included.

1.3 Definitions, Acronyms, and Abbreviations

Forex Foreign Exchange market — global decentralized market for currency trading.

Currency Pair A quotation of two currencies (e.g., EUR/USD).

Timeframe Chart interval used for analysis (e.g., 4H, 1H, 15M).

OHLC Open, High, Low, Close — price data structure.

Technical Indicator Quantitative calculation based on price/time (e.g., RSI, MACD).

RSI Relative Strength Index — momentum oscillator.

MACD Moving Average Convergence Divergence — trend-following momentum indicator.

Fundamental Analysis Assessment of economic events, news and macro data.

AI Artificial Intelligence.

API Application Programming Interface.

JSON JavaScript Object Notation — data interchange format.

CSV Comma-Separated Values — file format.

PDF Portable Document Format.

UI User Interface.

UX User Experience.

HTTPS Hypertext Transfer Protocol Secure.

TLS	Transport Layer Security.
WebSocket	Computer communications protocol for real-time communication.
TTL	Time-to-Live — duration for cached data validity.
RP	Requirement Provider (domain expert for the project).
SRS	Software Requirements Specification.
FR	Functional Requirement.
NFR	Non-Functional Requirement.

1.4 References

- IEEE Std 830-1984, IEEE Guide to Software Requirements Specifications
- Forex Trading Analysis Bot Project Proposal
- General Data Protection Regulation (GDPR)-Key Principles

1.5 Overview

This SRS is organized per IEEE Std 830-1984 and contains:

- **General Description** — product perspective, functions, environment.
- **Specific Requirements** — functional and non-functional requirements, external interfaces, assumptions and constraints, and traceability pointers.
- **Appendices** — use case and context diagrams plus other supporting informative items.

2 General Description

2.1 Product Perspective

The system is a standalone web-based application that assists traders in decision-making. It collects price data and economic events from external APIs, performs technical indicator analysis across multiple timeframes, and provides directional trading suggestions. Results are displayed through interactive graphical charts along with textual explanations of the bot's analysis. Users can customize indicators and thresholds, and the system recalculates and updates the charts and suggestions in real time, enabling personalized analysis and comparison. The system supports both guest and registered users, providing default analysis for guests and personalized features for registered users. The system includes a limited administrative interface for system monitoring and maintenance.

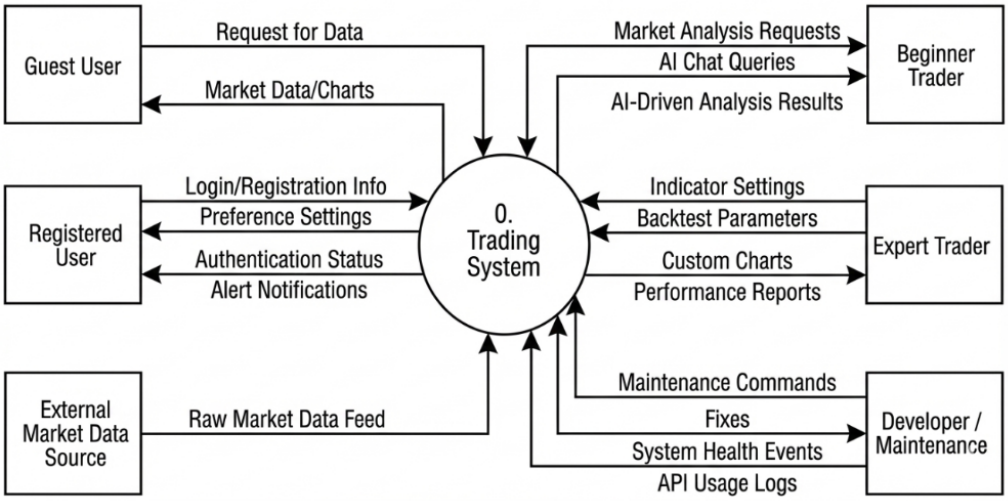

Figure 1: Context Diagram of the Forex Trading Analysis System

2.2 Product Functions

Major functions of the system include:

- **F1: User Management**

  - Support guest mode users with default analysis.

  - Support registered users with registration, login, logout, saved preferences, dashboards, and chat history.

  - Administrator access for system monitoring and user management.

  - Secure session management and data protection.

- **F2: Data Collection**

  - Collect historical and live Forex price data for selected currency pairs.

  - Retrieve economic events and fundamental market signals from external APIs.

  - Implement API fallback mechanisms and data validation.

- **F3: Indicator Calculation**

  - Compute technical indicators (e.g., RSI, MACD) across multiple timeframes.

  - Provide default calculations for guest users and customized calculations for registered users.

  - Allow registered users to customize indicators, parameters, and thresholds.

- **F4: AI Analysis**

  - Generate directional suggestions (UP, NEUTRAL, DOWN) with a confidence level for each timeframe.

  - Provide explanations showing how indicators influence AI suggestions.

- **F5: User Customization and Interaction**

  - Allow registered users to select indicators, adjust thresholds, and set preferred timeframes.

  - Recalculate results and update charts in real time based on user changes.

  - Enable users to query the bot for guidance or clarifications.

- **F6: Visualization and Reporting**

  - Display interactive charts with indicators, AI suggestions, and textual explanations.

  - Allow export of reports in PDF or CSV format.

  - Show default analysis for guests and personalized views for registered users.

  - Provide administrative dashboard for system monitoring and management.

  - Include tutorial and context-sensitive help system.

2.3 User Characteristics

User Type	**Description**	**Experience / Skills**	**Constraints**
Beginner Trader	New to trading, occasional user	Limited understanding of technical indicators and Forex concepts	Needs simple interface, default analysis, textual bot explanations, and guidance
Expert Trader	Experienced in trading	Familiar with technical indicators and multiple timeframes	Requires customization options, saved preferences, detailed charts, minimal guidance
Registered User	Logged-in user with saved preferences	Varies (beginner to expert)	System must store preferences and chat history
Guest User	Access without an account	Varies	Default analysis only, no data saving
System Developers	Manages system health, users, and configurations	Technical expertise in software and database management	Requires access to logs, configuration settings, and performance monitoring
Administrator	Manages system health, users, and configurations	Technical expertise in system monitoring and basic user support	Requires secure access, detailed logs, and system control interfaces

2.4 General Constraints

The design and operation of the Forex Trading Analysis System are subject to several high-level constraints that influence implementation choices:

- **External Data Dependency:** The system relies on third-party APIs to retrieve historical and live Forex prices and economic event data. API availability and rate limits may affect system performance and accuracy.
- **Data Storage:** All registered user data, including preferences, customized indicators, chat history, and exported reports, must be stored securely. Storage capacity and retrieval speed impose constraints on system scalability and responsiveness.
- **Real-Time Multi-User Calculations:** The system must support real-time recalculation of multiple technical indicators and AI-based analysis for multiple registered users simultaneously.
- **Data Security:** Registered user information, authentication credentials, and personalized settings must be protected according to best practices for web application security.

2.5 Assumptions and Dependencies

- At least one reliable Forex price API and one economic events API are available.
- It is assumed that data provided by third-party APIs is accurate, timely, and consistent.
- Users have basic familiarity with charting and trading terminology.
- Internet connectivity is present during normal operation.
- The backend server environment required to host the application and perform computations remains available and operational during system use.

3 Specific Requirements

Based on your requirements, here is the rewritten and consolidated list of Functional Requirements, with duplicates removed, economic calendar functions excluded (keeping only AI analysis integration), and organized in logical order:

3.1 Functional Requirements (FRs)

3.1.1 FR-1: User Registration

Introduction: The system shall allow a new user to create an account to enable personalized features and data storage.

Inputs:

- Username
- Email address

- Password
- Confirm password

Processing:

- Validate that all required fields are filled.
- Verify the format of the email address.
- Ensure that the password and confirm password match.
- Check that the email address is not already registered.
- Encrypt the password before storage.
- Store validated user information in the database.

Outputs:

- Confirmation message on successful registration.
- Error message if registration fails.

3.1.2 FR-2: User Login

Introduction: The system shall allow registered users to access the system using valid credentials.

Inputs:

- Registered email address
- Password

Processing:

- Verify that login fields are not empty.
- Authenticate user credentials against stored data.
- Create an active session upon successful authentication.

Outputs:

- Successful login message and system access.
- Error message for invalid credentials.

3.1.3 FR-3: Login Failure Handling

Introduction: The system shall manage incorrect login attempts to ensure security.

Inputs:

- Incorrect email or password.

Processing:

- Detect and log invalid login attempts.
- Display an appropriate error message.

- Allow the user to retry login.

Outputs:

- Error message indicating login failure.

3.1.4 FR-4: Account Deletion and Data Management

Introduction: The system shall allow users to manage their account data, including export and deletion.

Inputs:

- User request for data export or account deletion.
- User authentication confirmation.

Processing:

- Verify user identity through re-authentication.
- For export requests: compile all user data (preferences, chat history, saved analyses) into structured format.
- For deletion requests: permanently remove all user data from the database.
- Maintain audit logs of data management actions.
- Send confirmation to user upon completion.

Outputs:

- Exported data package (JSON/CSV format) for data portability.
- Confirmation of account/data deletion.
- Audit trail of data management actions.

3.1.5 FR-5: Guest Mode

Introduction: The system shall allow users to access the system without an account.

Inputs:

- User request to continue as guest.

Processing:

- Assign a temporary session to the guest user.
- Enable default indicators and settings.
- Disable personalized data saving features.

Outputs:

- Access to default analysis features.
- Notification of limited functionality.

3.1.6 FR-6: User Logout

Introduction: The system shall allow users to securely exit the system.

Inputs:

- Logout request.

Processing:

- Terminate the active user session.
- Clear all session-related data.

Outputs:

- Logout confirmation.
- Redirection to the home or login page.

3.1.7 FR-7: Access Control Based on User Type

Introduction: The system shall control feature access based on whether the user is a guest or a registered user.

Inputs:

- User session type (guest or registered).
- Requested feature or function.

Processing:

- Check user type and authentication status at runtime.
- Verify user permissions against the requested feature.
- Restrict or enable features as per the user's type.

Outputs:

- Appropriate level of feature access.
- Error message when guest users attempt to access restricted features.

3.1.8 FR-8: Forex Historical Data Retrieval

Introduction: This function retrieves historical price data for user-selected currency pairs and timeframes from external APIs.

Inputs:

- Currency pair
- Timeframe
- Start date and end date

Processing:

- Validate that all inputs are provided.

- Send an API request to the primary Forex data provider.
- Parse the returned OHLC, volume, and timestamp data.
- Verify data covers the requested date range and timeframe.
- If data is incomplete, attempt to retrieve from a secondary API.
- Store parsed data in temporary memory or cache for calculation.

Outputs:

- Complete dataset matching the request parameters.
- Error message if data is unavailable from all sources.

3.1.9 FR-9: Real-Time Data Fetching

Introduction: This function retrieves live market prices to update charts and AI calculations.

Inputs:

- Selected currency pair
- Selected timeframe

Processing:

- Verify the selected currency pair is supported.
- Send API request to retrieve the latest price/candle data.
- Validate timestamps and values for consistency.
- Handle API timeouts gracefully with configurable retries.
- Log API errors for diagnostics.
- Cache latest prices for immediate use by technical indicators.

Outputs:

- Latest market price data.
- Error message if data is unavailable.

3.1.10 FR-10: Economic Events Data for AI Analysis

Introduction: The system shall retrieve historical economic events data relevant to selected currency pairs from external APIs for use as AI model features.

Inputs:

- Selected currency pair(s)
- Historical date range
- Event type filter (e.g., central bank decisions, inflation data)

Processing:

- Send API requests to the configured economic data provider.
- Parse returned data (event name, date/time, impact level, actual values).
- Filter events relevant to the selected currency pair(s).
- Store formatted event data for AI model integration.
- Handle API errors gracefully.

Outputs:

- Structured dataset of historical economic events.
- Error message if data cannot be retrieved.

3.1.11 FR-11: API Response Validation

Introduction: Ensures data received from external APIs is complete, accurate, and usable.

Inputs:

- Raw data (JSON/CSV) from any external API.

Processing:

- Check for required fields (e.g., OHLC, timestamps).
- Detect and flag missing, null, or malformed data.
- Discard inconsistent entries.

Outputs:

- Validated, clean data structure.
- Error logs for invalid or missing data.

3.1.12 FR-12: API Fallback Handling

Introduction: Provides continuity of data retrieval if the primary API fails.

Inputs:

- Primary API status (success/failure)
- Secondary API configuration

Processing:

- If primary API fails, automatically attempt the secondary API.
- Merge or replace failed data with the fallback source.
- Notify the user if fallback data is being used.
- Log all failover attempts.

Outputs:

- Complete dataset from an available API source.

- User notification in case of fallback activation.
- System log entries.

3.1.13 FR-13: Technical Indicator Calculation

Introduction: The system shall compute selected technical indicators to support AI analysis and visualization.

Inputs:

- Validated price data (OHLC, volume)
- Selected currency pair and timeframe(s)
- User-selected indicators and parameters (or defaults for guests)

Processing:

- Retrieve the latest validated price data.
- Apply the corresponding calculation algorithm for each selected indicator.
- Process multiple timeframes simultaneously if requested.
- Handle missing data gracefully and flag unreliable results.
- Cache computed indicator values.

Outputs:

- Computed indicator values for the specified pair and timeframe(s).
- Warnings if calculations cannot be performed.

3.1.14 FR-14: AI-Based Market Analysis

Introduction: The system shall generate directional suggestions (UP, NEUTRAL, DOWN) for currency pairs.

Inputs:

- Computed technical indicator values.
- Historical economic event data (FR-9).
- User-selected timeframe(s).

Processing:

- Feed input features (indicators + economic events) into the AI model.
- Calculate a directional suggestion and a confidence level.
- Generate a textual explanation linking key indicators and events to the suggestion.

Outputs:

- Directional suggestion for each timeframe.
- Associated confidence score.

- Explanation text.

3.1.15 FR-15: Indicator Customization

Introduction: The system shall allow registered users to customize indicators, parameters, and thresholds.

Inputs:

- User-selected indicators.
- Custom parameter values (e.g., RSI period).
- Custom thresholds.

Processing:

- Store the user's custom indicator settings in the database.
- Apply customized calculations for charts and AI analysis.
- Validate parameter ranges.
- Update visualizations and calculations in real time.

Outputs:

- Confirmation of saved settings.
- Updated analysis results reflecting custom parameters.

3.1.16 FR-16: Charting and Visualization

Introduction: The system shall display computed indicators and AI suggestions in interactive charts.

Inputs:

- Computed indicator values.
- AI directional suggestions and confidence.
- Selected currency pair and timeframe.

Processing:

- Plot indicators as overlays on price charts.
- Visually highlight AI suggestions and confidence levels.
- Allow zooming, scrolling, and timeframe switching.
- Render appropriate views for guest and registered users.

Outputs:

- Interactive charts with indicators and AI insights.

3.1.17 FR-17: Report Generation and Export

Introduction: The system shall allow users to generate and export analysis reports.

Inputs:

- Selected currency pairs and timeframes.
- Associated indicators and AI suggestions.
- User-selected report format (PDF or CSV).

Processing:

- Collate analysis data into a structured report.
- Format the report according to the selected format.
- Validate data availability before export.

Outputs:

- Downloadable report file.
- Error message if export fails.

3.1.18 FR-18: User Preferences Management

Introduction: The system shall allow registered users to save, load, and manage settings.

Inputs:

- User-selected indicators, thresholds, and UI preferences.
- Request to save, load, or reset preferences.

Processing:

- Store user preferences securely in the database.
- Apply saved preferences upon user login or request.
- Allow resetting to default settings.

Outputs:

- Confirmation of saved preferences.
- Updated UI and analysis configuration.

3.1.19 FR-19: Data Caching

Introduction: The system shall cache API responses and computed values to improve performance.

Inputs:

- API response data.
- Computed indicator results.

Processing:

- Store frequently used data in cache with a Time-to-Live (TTL).
- Invalidate cache when new data arrives.
- Retrieve cached data for repeated queries before calling APIs.

Outputs:

- Faster data retrieval.
- Reduced number of external API calls.

3.1.20 FR-20: Historical Analysis Storage

Introduction: The system shall maintain a history of AI suggestions and analysis for review.

Inputs:

- Computed indicators and AI suggestions.
- Timestamp of analysis.

Processing:

- Store historical analysis results with timestamps.
- Allow registered users to query their past analyses.

Outputs:

- Accessible logs of historical analysis for registered users.

3.1.21 FR-21: Multi-Timeframe Analysis Aggregation

Introduction: The system shall aggregate AI results across timeframes to provide a consolidated market view.

Inputs:

- AI suggestions from multiple timeframes.

Processing:

- Compare and combine results from different timeframes.
- Resolve conflicts using weighted logic.
- Generate an overall market trend or bias.

Outputs:

- Consolidated market direction and confidence.
- Explanation showing contributing timeframes.

3.1.22 FR-22: Alert System

Introduction: The system shall inform users of specific market conditions.

Inputs:

- Real-time price and indicator data.
- User-defined alert conditions (price levels, indicator thresholds).

Processing:

- Monitor market data against user conditions in real-time.
- Trigger notifications when conditions are met.
- Support user preferences for alert types.

Outputs:

- Real-time alerts via the system UI.
- Log of triggered alerts.

3.1.23 FR-23: Interactive AI Chat

Introduction: The system shall provide a chat interface for users to query AI suggestions and request explanations.

Inputs:

- User queries in natural language.
- Current analysis context.

Processing:

- Process user queries to identify intent.
- Retrieve relevant market analysis data.
- Generate human-readable, context-aware explanations.
- Maintain chat history for registered users.

Outputs:

- Interactive, explanatory responses.
- Persistent chat history for registered users.

3.1.24 FR-24: Admin Dashboard

Introduction: The system shall provide administrative tools for monitoring and management.

Inputs:

- Admin login credentials.
- System logs and performance data.

- User account information.

Processing:

- Authenticate administrator using elevated privilege credentials.
- Display real-time system health and performance metrics (CPU, memory, API latency).
- Track API usage statistics and error rates across all users.
- Provide user account management capabilities (view users, deactivate accounts, reset passwords upon request).
- Generate system audit reports with timestamps and user activities.
- Monitor Forex data feed status and alert on disruptions.

Outputs:

- Admin dashboard with monitoring visuals and control.
- System audit and troubleshooting logs.
- API usage analytics and cost tracking reports.

3.1.25 FR-25: Security and Session Management

Introduction: The system shall manage session security and data protection.

Inputs:

- User login/session data.

Processing:

- Enforce secure session timeouts.
- Encrypt sensitive data and credentials.
- Implement secure authentication practices.

Outputs:

- Secure user sessions.
- Protected user data.

3.1.26 FR-26: Data Privacy Management

Introduction: The system shall provide tools for managing user data in compliance with privacy regulations.

Inputs:

- User data export or deletion requests.

Processing:

- Provide data export functionality (JSON/CSV).

- Implement complete data deletion upon user request.

Outputs:

- Exported data packages.
- Confirmation of data deletion.

3.1.27 FR-27: Multi-Currency Pair Comparison(optional)

Introduction: The system shall allow side-by-side comparison of multiple currency pairs.

Inputs:

- Selected currency pairs.
- Comparison parameters and timeframe.

Processing:

- Synchronize charts for the selected pairs.
- Compute and display correlation metrics.

Outputs:

- Multi-pair comparison interface.
- Correlation analysis summary.

3.1.28 FR-28: API Management and Monitoring

Introduction: The system shall monitor API usage and manage keys for performance and cost control.

Inputs:

- API keys and rate limits.
- API usage statistics.

Processing:

- Track API usage per user and system-wide.
- Enforce quotas and manage key rotation.
- Provide analytics on API performance and cost.

Outputs:

- API usage and cost dashboard.
- Quota management interface.

3.1.29 FR-29: Tutorial and Help System

Introduction: The system shall provide context-sensitive help and tutorials.

Inputs:

- User request for help.
- UI context (current page/feature).

Processing:

- Display interactive tutorials for new users.
- Show tooltips and detailed explanations for complex features/terms.

Outputs:

- On-demand help content.
- Step-by-step guides.

3.1.30 FR-30: Error Handling and User Notifications

Introduction: The system shall notify users of errors or important system events.

Inputs:

- System errors (API failures, calculation errors).
- Invalid user inputs.

Processing:

- Detect and log errors.
- Display user-friendly notifications describing the issue.
- Suggest corrective actions where possible.

Outputs:

- Informative error or warning messages.
- System logs for diagnostics.

3.2 Non-Functional Requirements (NFRs)

NFR-1: Performance

- The system shall respond to standard user interactions (login, dashboard load, chart updates) within **2 seconds** under normal load (up to 1,000 concurrent users).
- AI-based market analysis results shall be displayed within **5 seconds** of request submission.
- Historical data queries (up to 1 year) shall return within **10 seconds**.

NFR-2: Real-Time Updates

- Live Forex price data shall update on the user interface within **1–3 seconds** of receiving updates from external APIs.
- Real-time alerts shall be triggered within **5 seconds** of condition fulfillment.

NFR-3: Scalability

- The system shall support **10,000 concurrent users** performing standard operations without significant performance degradation.
- The architecture shall support horizontal scaling, allowing addition of server instances without service interruption.
- Database shall handle **1 TB** of historical data with efficient query performance.

NFR-4: Availability

- The system shall maintain **99% uptime** during operational hours (24/7).
- Recovery time for temporary failures shall not exceed **5 minutes** for automatic recoveries and **30 minutes** for manual interventions.
- Graceful degradation: In case of external API failures, the system shall continue functioning with cached data and display appropriate warnings.

NFR-5: Security

- All user passwords shall be stored using strong, salted hashing (bcrypt or equivalent).
- Sensitive data transmission shall use TLS 1.2 or higher.
- The system shall enforce session timeout after **15 minutes** of inactivity.
- Authentication failures shall be logged and monitored for brute-force attack patterns.
- API keys and credentials shall be stored encrypted in the database.

NFR-6: Usability

- The interface shall be intuitive for users with basic Forex knowledge.
- New users shall be able to perform core functions (view charts, run analysis) within **5 minutes** of first use.
- The system shall provide context-sensitive help and tooltips for all complex features.
- The interface shall be responsive and functional on desktop, tablet, and mobile devices.

NFR-7: Reliability

- System shall maintain data integrity during power failures or unexpected shutdowns.
- No single point of failure shall cause complete system unavailability.
- Data backups shall be performed daily with 30-day retention.

NFR-8: Maintainability

- The codebase shall follow modular architecture with clear separation of concerns.
- Average time to fix a critical bug shall be less than **48 hours**.

- Configuration changes shall not require system restart where possible.

NFR-9: Compliance and Standards

- All generated reports shall comply with standard CSV and PDF formats.
- User data management shall comply with GDPR principles (right to access, right to be forgotten).

3.3 External Interface Requirements

3.3.1 User Interfaces

The system shall provide a responsive web-based dashboard accessible via modern browsers including Chrome, Firefox, Safari, and Edge on desktop, tablet, and mobile devices.

Key Components:

- Login/Registration pages with secure authentication
- Main dashboard with real-time charts and AI suggestions
- Settings panel for indicator customization and preferences
- Admin dashboard for system monitoring and user management
- Report generation and export interface
- Consistent navigation with clear visual hierarchy
- Context-sensitive help and interactive tutorials
- Keyboard accessibility for all interactive elements
- Responsive design adapting to different screen sizes

3.3.2 Hardware Interfaces

- The backend server should have a minimum configuration of 8 CPU cores, 16 GB RAM, 200 GB SSD storage and network connectivity sufficient to support 10,000 concurrent users
- The system shall support smartphone, desktop, laptop, and tablet devices.
- Communication between client and server shall be over HTTPS/TLS. Real-time price updates shall use WebSocket protocol.

3.3.3 Software Interfaces

Operating Systems:

- Server: Linux (Ubuntu 20.04 or later)
- Client: Any OS with modern web browser support

Database:

- MySQL for storing user data, historical Forex data, and AI models
- Access via standard SQL queries and APIs

Third-party APIs:

- Forex price data through REST/JSON or WebSocket APIs
- AI model integration via internal REST APIs
- Authentication via OAuth 2.0 or API keys

Libraries and Frameworks:

- Frontend: React.js / Angular
- Backend: Node.js / Python Flask / Django
- Charts: Highcharts, D3.js, or Plotly

3.3.4 Communication Interfaces

Network Protocols:

- HTTPS/TLS for secure client-server communication
- WebSocket for real-time Forex price streaming

Data Formats:

- JSON for all API requests and responses
- CSV for user-exportable reports

Error Handling:

- API failures or network timeouts shall trigger graceful degradation, showing cached or partial data

3.4 Design Constraints

3.4.1 Standards Compliance

The system shall comply with the following standards:

- **Report Format:** All generated reports shall be in CSV or PDF format.
- **Data Protection:** User data handling shall follow GDPR principles including data portability and right to erasure.
- **Audit Tracing:** Critical operations such as login, AI analysis generation, and Forex data updates shall be logged with timestamps and user ID to support traceability and basic compliance.

- **Security Standards:** Sensitive data must be encrypted using AES-256 for storage and transmitted over TLS 1.2 or higher.
- All AI-generated trading suggestions must include an explainable rationale citing the indicators and data points used in the decision.

3.4.2 Hardware Limitations

The system shall operate within the following hardware constraints:

- **Server:** Minimum 8 CPU cores, 16 GB RAM, 200 GB SSD storage. Should support up to 10,000 concurrent users with acceptable performance.
- **Client Devices:** Desktop, laptop, tablet and standard input devices.
- **Network:** Minimum 5 Mbps bandwidth; tolerate latency up to 500 ms for real-time updates.
- **Instruction Set:** Backend must run on standard x86_64 processors.

3.4.3 Software and Architecture Constraints

- Backend: Python (Flask/Django) or Node.js; Frontend: React.js or Angular.
- Database: MySQL
- System shall follow modular architecture for easier maintenance and future scaling.
- AI computation shall be separated from frontend to reduce latency.
- External Forex APIs shall use REST or WebSocket protocols.

3.5 Software System Attributes

Reliability The system shall operate reliably under normal usage conditions. In case of temporary failures such as API timeouts or network issues, the system shall handle errors gracefully and continue functioning without data corruption.

Availability The system shall be available to users during normal operational hours with a minimum uptime of 99%. Planned maintenance activities shall be communicated to users in advance where possible.

Security The system shall protect user data by enforcing authentication, secure session management, and encrypted communication. Unauthorized access to user accounts and sensitive information shall be prevented.

Maintainability The system shall be designed using a modular architecture, allowing developers to update, debug, or enhance individual components without impacting the entire system.

Portability The system shall be accessible through standard web browsers and shall not require platform-specific installations, enabling use across different operating systems such as Windows, Linux, and macOS.

Usability The system shall be easy to learn and use for users with basic knowledge of Forex trading. New users shall be able to perform core functions such as viewing charts and analysis after minimal training.

4 Appendix A: Use Case Diagram

This appendix presents the use case diagram representing the interactions between users and the Forex analysis system.

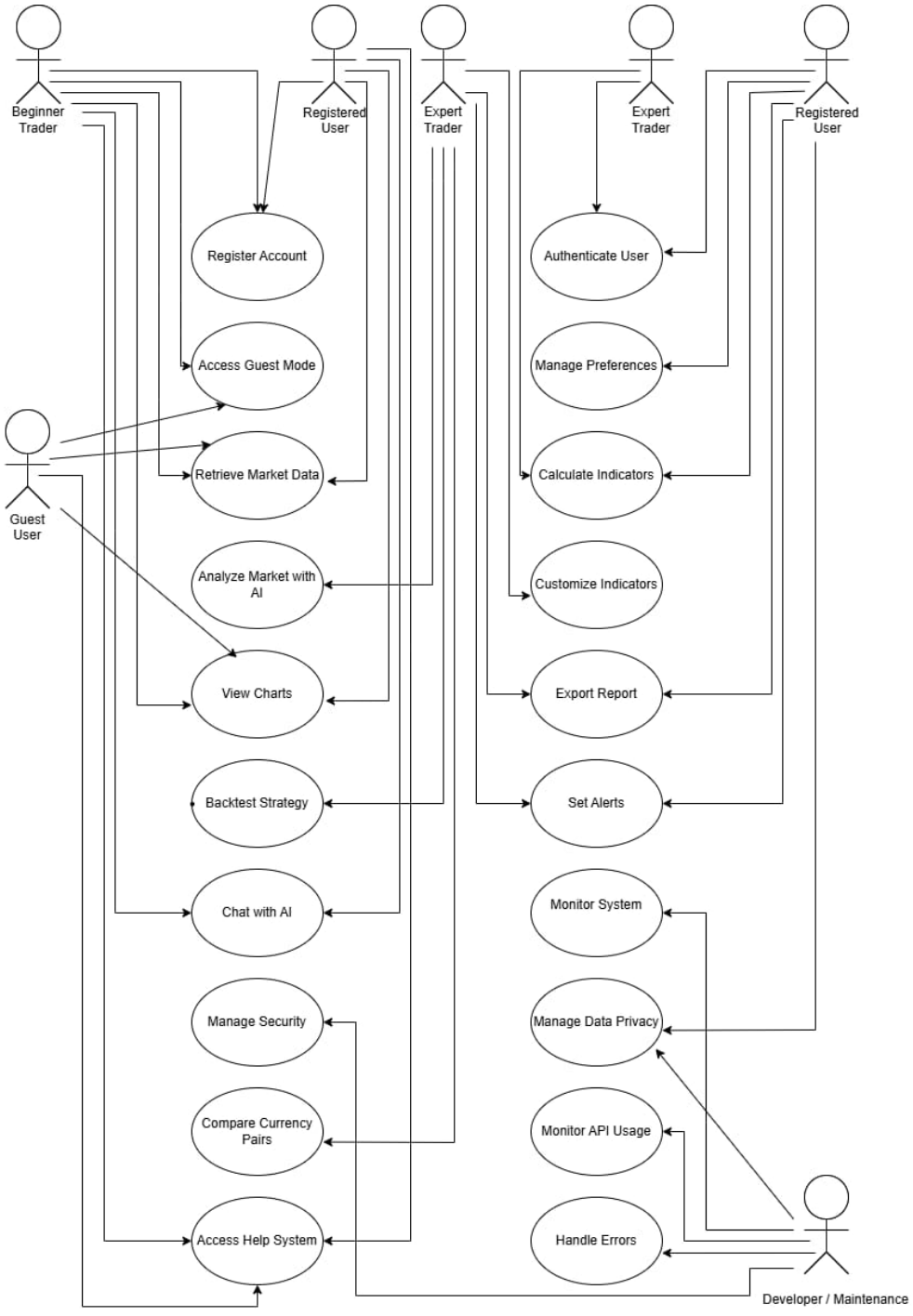

Figure 2: Use-Case Diagram of the Forex Trading Analysis System