Software Requirements Specification (SRS)

AI-powered Forex Trading Analysis Bot

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

Submission Date: December 27, 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 Bot**. 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.

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).

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

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

AI   Artificial Intelligence.

RP   Requirement Provider (domain expert for the project).

SRS   Software Requirements Specification.

UI   User Interface.

1.4 References

- IEEE Std 830-1984, IEEE Guide to Software Requirements Specifications
- Forex Trading Analysis Bot Project Proposal

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.

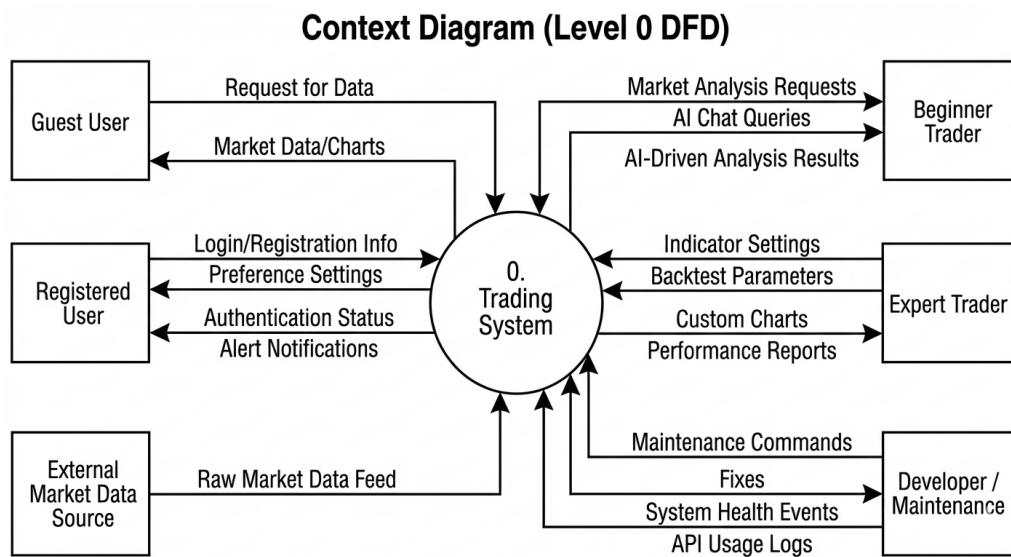# Context Diagram (Level-0 / Zero Diagram)

Figure 1: Level-0 Context Diagram of the Forex Trading Analysis System

2.2 Product Functions

Major functions of the system include:

- **F1: User Management**

  - Support guestmode users with default analysis.

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

- **F2: Data Collection**

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

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

- **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.

- **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.

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

Developer / Maintenance	Responsible for system maintenance	Technical expertise in software and database management	Requires access to logs, configuration settings, and performance monitoring

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: 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.

3.1.5 FR-5: 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.

3.1.6   FR-6: 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).

Processing:

- Check user type at runtime.

- Restrict or enable features as per the user's type.

Outputs:

- Appropriate level of feature access.

3.1.7   FR-7: 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.

3.1.8 FR-8: 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.

3.1.9 FR-9: 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.

3.1.10 FR-10: 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.

3.1.11 FR-11: 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.

3.1.12 FR-12: 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.

3.1.13 FR-13: 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.

3.1.14 FR-14: 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.

3.1.15 FR-15: 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.

3.1.16 FR-16: 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.

3.1.17 FR-17: 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.

3.1.18   FR-18: 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.

3.1.19   FR-19: 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.

3.1.20 FR-20: 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.

3.1.21 FR-21: Backtesting

Introduction: The system shall allow testing of indicator strategies against historical data.

Inputs:

- Selected historical date range.
- User-defined indicator strategy.

Processing:

- Apply the strategy calculations on historical price data.
- Compare strategy signals with actual market outcomes.
- Generate performance metrics (e.g., win rate).

Outputs:

- Backtesting report with performance metrics.
- Graphical comparison of strategy performance.

3.1.22 FR-22: Alert System

Introduction: The system shall notify 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.

Processing:

- Display system health and performance metrics.
- Track API usage and error rates.
- Provide basic user account management.

Outputs:

- Admin dashboard with monitoring visuals.
- System audit and troubleshooting logs.

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

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)

Non-functional requirements provide measurable criteria.

ID	Requirement
NFR-1 (Performance)	The system shall serve a standard analysis request (precomputed indicators + AI inference) and render the results on the dashboard within **2 seconds** under expected prototype load.

NFR-2 (Availability)	The system shall be available to users during typical working hours (weekdays) with reasonable uptime; prototype uptime target is $> 95\%$ during demo periods.
NFR-3 (Reliability)	The system shall gracefully degrade when an external data API fails, showing cached results and a clear message indicating stale data.
NFR-4 (Security)	The system shall never store raw user credentials for external services; any credentials shall be stored only using secure secrets management if needed.
NFR-5 (Usability)	The user interface shall be intuitive for traders with basic knowledge; initial user testing should yield an average SUS-like subjective usability score above a defined baseline (to be collected during evaluation).
NFR-6 (Explainability)	For every AI suggestion, the system shall present at least the top three contributing features and an automatically generated natural-language explanation.
NFR-7 (Traceability)	All functional requirements shall have unique IDs and be traceable to use cases and test cases.

3.3 External Interface Requirements

3.3.1 User Interface

The system shall provide a web-based dashboard for:

- Selecting currency pairs and timeframes.

- Viewing chart(s) with indicator overlays.

- Displaying AI signals and explanations.

- Managing user-defined indicators and exporting reports.

3.3.2 Software Interfaces

- The system shall call a Forex price API to fetch historical and live price data (REST/HTTPS).

- The system shall call an economic events API (REST/HTTPS) to retrieve scheduled events.

- The system shall expose an internal API (for integration/testing) to request analysis programmatically.

3.3.3 Hardware Interfaces

No specialized hardware is required for prototype operation beyond a standard server/VM for the backend and user device for the frontend.

3.4 Traceability

Each requirement ID in Sections above will be referenced in test plans and use case mappings in the project repository. Example mapping:

- **UseCase: Analyze Market** → FR-1, FR-4, FR-5, FR-6, FR-7, FR-9.

- **UseCase: Apply Custom Indicator** → FR-8, FR-9, FR-10.
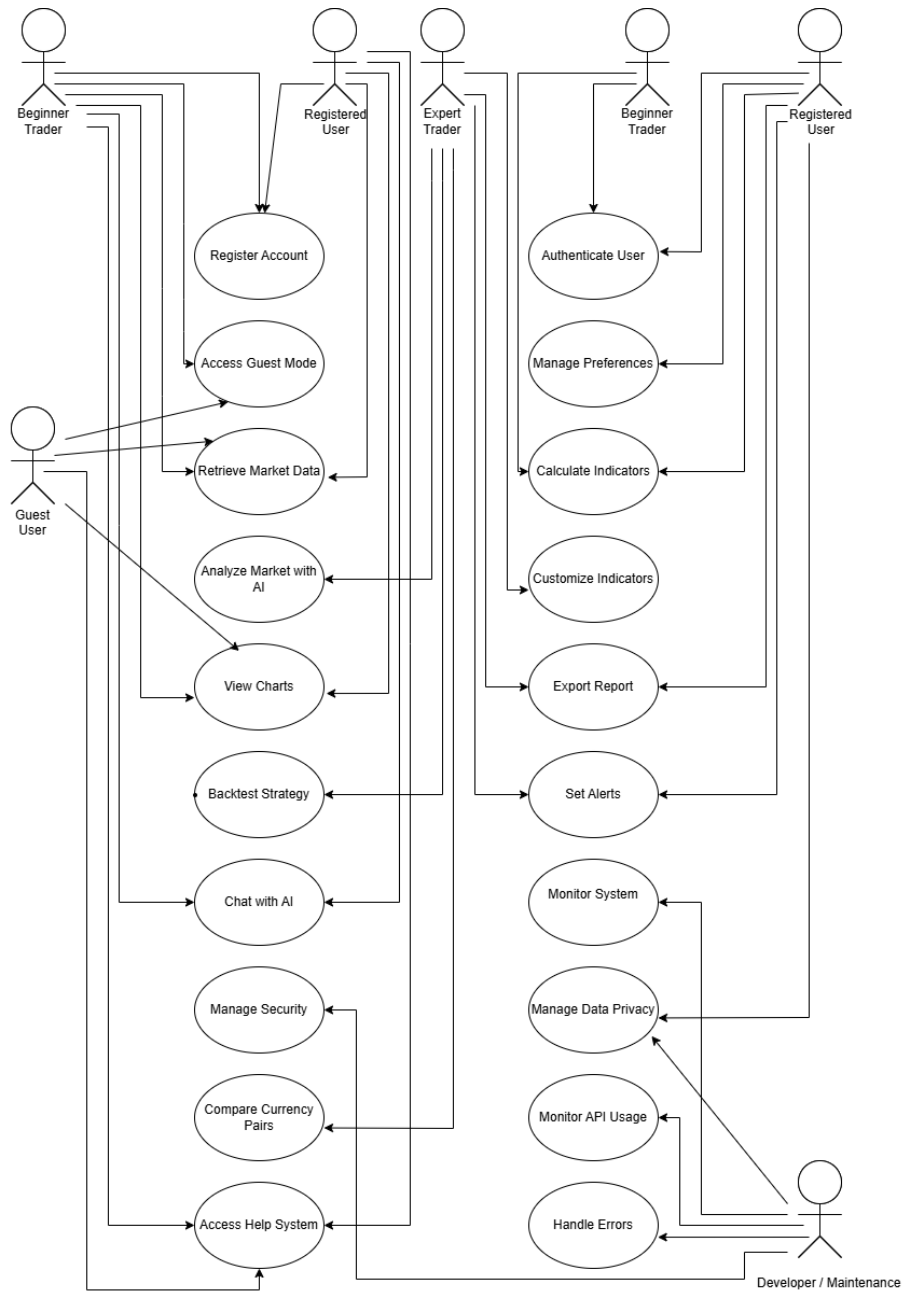
System Architecture Diagram

Figure 2: Overall System Architecture of AI-powered Forex Trading Analysis Bot

4 Appendices

4.1 Appendix A: Use Case Descriptions (brief)

Use Case: Analyze Market

Use Case: Analyze Market Primary actor: Trader. Precondition: Currency pair and timeframe selected. Postcondition: AI directional suggestion and explanation displayed; user may export a report.

Use Case: Generate AI Signal Primary actor: Trader / system (automated). Precondition: Valid feature vector available. Postcondition: Directional signal + confidence and explanation produced.

Use Case: Apply Custom Indicators Primary actor: Trader. Precondition: UI supports custom indicator definitions. Postcondition: Custom indicator computed; results compared against AI signals.