Functional Requirements

Data Retrieval

FR1.1: The system shall automatically download historical daily stock data for all issuers from the Macedonian Stock Exchange.

FR1.2: The system shall support downloading data covering at least the last 10 years.

FR1.3: Users shall be able to specify the date range for stock data retrieval through the command line interface.

FR1.4: The system shall support retries up to 3 times in case of failed downloads due to network issues.

Data Processing

FR2.1: The system shall filter out unnecessary information, retaining only essential stock attributes such as date, closing price, and volume.

FR2.2: The system shall transform raw stock data into a standardized format (e.g., CSV, JSON) for analysis.

FR2.3: The system shall handle missing or corrupted data by either filling, flagging, or removing such records based on user configuration.

Data Storage

FR3.1: The system shall store processed stock data in a PostgreSQL database.

FR3.2: The database shall be indexed by issuer name and date to allow fast retrieval.

FR3.3: The system shall ensure data integrity during storage by validating input formats before saving.

User Interaction

FR4.1: The system shall provide a Command Line Interface (CLI) for initiating data download, processing, and reporting tasks.

FR4.2: The system shall display real-time progress and status messages during data processing.

FR4.3: The system shall allow users to export processed data into CSV or JSON formats for external analysis.

Data Export

FR5.1: Users shall be able to export the filtered and processed data for external analysis tools.

FR5.2: The system shall generate logs for all data processing activities to aid in troubleshooting.

Non-Functional Requirements

Performance

NFR1.1: The system shall process historical stock data for all issuers within a maximum of 1 hour for datasets up to 10 million records.

NFR1.2: The system shall respond to user commands within 2 seconds.

NFR1.3: The system shall support concurrent data processing tasks without significant performance degradation.

Scalability

NFR2.1: The system shall handle an increasing volume of data by scaling the data processing pipeline.

NFR2.2: The system architecture shall support adding new data sources without major changes to the codebase.

Reliability

NFR3.1: The system shall achieve a data accuracy rate of 99.9% for all processed stock data.

NFR3.2: The system shall implement automatic retries for failed API requests to ensure data availability.

Security

NFR4.1: All data transmitted between the application and external APIs shall be encrypted using HTTPS.

NFR4.2: The system shall implement role-based access control to restrict unauthorized access to sensitive financial data.

Usability

NFR5.1: The Command Line Interface (CLI) shall be user-friendly and provide clear instructions for all available commands.

NFR5.2: The system shall provide detailed error messages and troubleshooting tips for common issues.

Maintainability

NFR6.1: The system shall use modular code design, following the Pipe and Filter architecture pattern, to facilitate easy updates and maintenance.

NFR6.2: The code shall be documented thoroughly, with inline comments and external documentation for key modules.

Portability

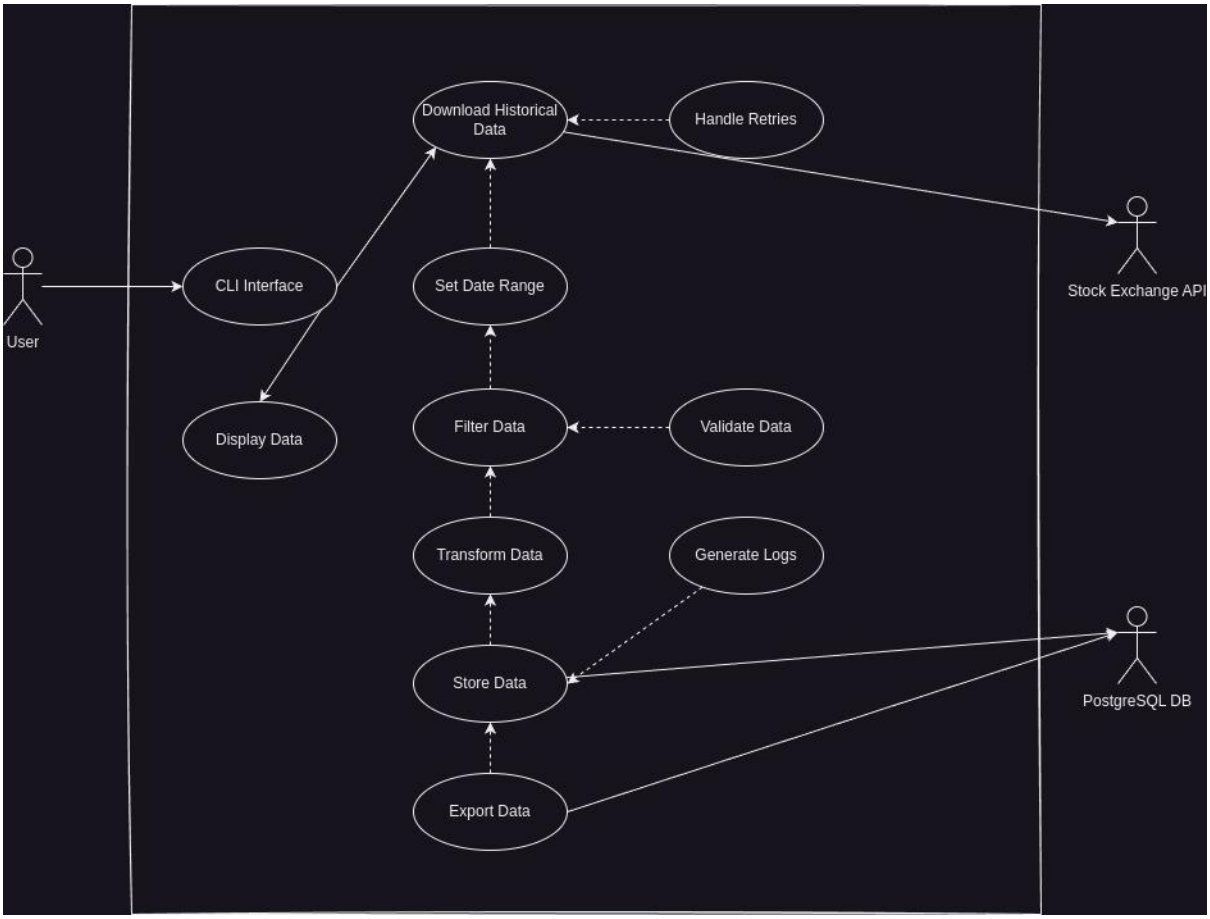
NFR7.1: The application shall be cross-platform compatible, supporting Windows, macOS, and Linux environments.

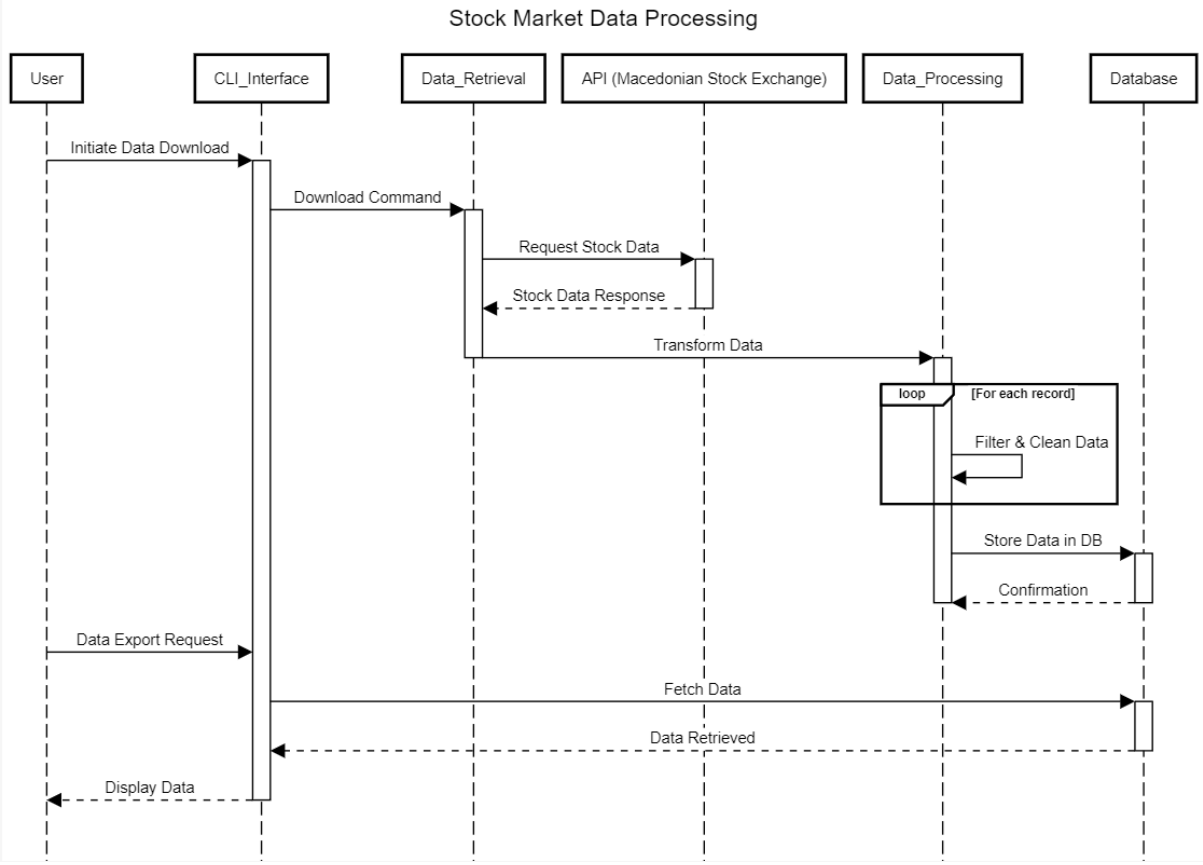
NFR7.2: The system shall require only Python and PostgreSQL as dependencies, minimizing external library requirements.

Compliance

NFR8.1: The system shall comply with data privacy regulations, ensuring that all stored financial data is protected.

NFR8.2: The application shall provide audit logs for all data processing activities to meet compliance requirements.




Stock Market Data Processing
User
CLI_Interface
Data_Retrieval
API (Macedonian Stock Exchange)
Data_Processing
Database
Initiate Data Download
Download Command
Request Stock Data
Stock Data Response
Transform Data
loop
[For each record]
Filter & Clean Data
Store Data in DB
Confirmation
Data Export Request
Fetch Data
Data Retrieved
Display Data