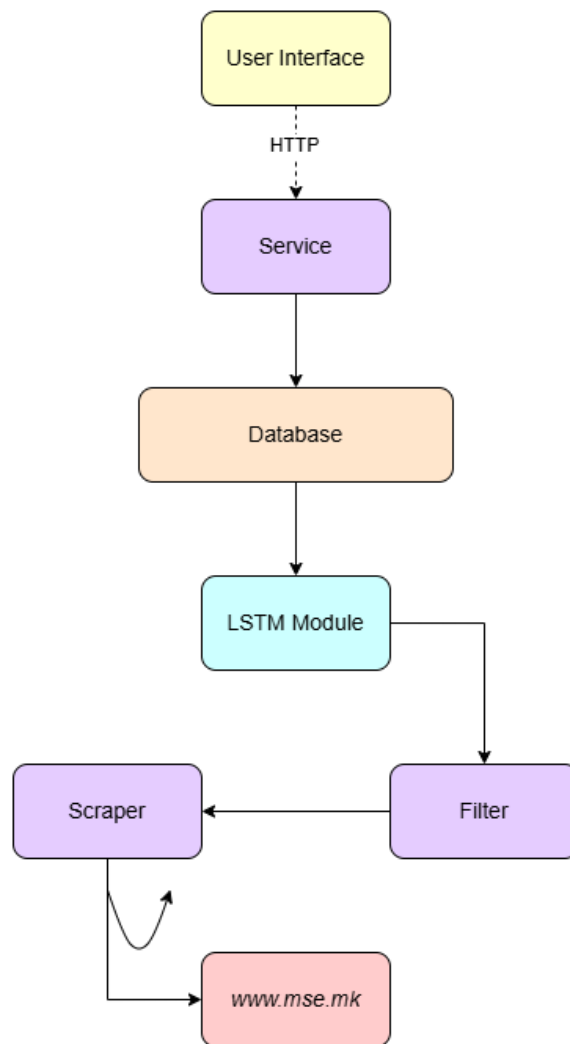


Execution Architecture

Key Components, Data Flow and Communication Protocols

The execution architecture illustrated in the provided diagram describes the runtime interaction between components of the stock market analysis application. The architecture is designed to ensure a clear flow of data from its source to the end-user, leveraging modular components for data scraping, filtering, processing, storage, and analysis.



Key Components

1. User Interface (UI):

- Role: Serves as the entry point for user interaction.

- Interaction: Users send requests and receive responses through HTTP protocols.
- Functionality: Enables users to view processed data, perform queries, and analyze stock market trends.

2. Service Layer:

- Role: Acts as a mediator between the User Interface and the backend components.
- Interaction: Receives HTTP requests from the User Interface and processes them to interact with the database and other backend modules.
- Functionality: Handles routing, authentication (if required), and business logic.

3. Database:

- Role: Central storage for all historical and updated stock data.
- Interaction: The Service, Scraper, Filter, and LSTM Module interact with the database for data retrieval and storage.
- Functionality: Maintains consistency and integrity of stock data, ensuring it is formatted and ready for use by other components.

4. Scraper:

- Role: Automates the extraction of stock market data from the Macedonian Stock Exchange website (www.mse.mk).
- Interaction:
 - Communicates directly with the source website to fetch raw data.
 - Passes the data to the Filter for preprocessing.
- Functionality: Ensures continuous and real-time scraping of data, adhering to the predefined rules to capture relevant information.

5. Filter:

- Role: Processes and cleans the raw data provided by the Scraper.
- Interaction:
 - Accepts raw data from the Scraper.

- Outputs cleaned and formatted data to the database.
- Functionality:
 - Removes unnecessary information.
 - Ensures consistency in data formatting (e.g., date and price formatting).
 - Prepares data for further analysis.

6. LSTM Module:

- Role: Performs predictive analysis on stock data using a Long Short-Term Memory (LSTM) neural network.
- Interaction:
 - Retrieves historical data from the database.
 - Processes predictions and outputs results for user queries via the Service.
- Functionality:
 - Provides insights and trends based on historical stock data.
 - Generates predictions to help users make informed decisions.

7. Source Website (www.mse.mk):

- Role: The origin of the raw stock market data.
- Interaction: The Scraper retrieves data from this source.
- Functionality: Supplies real-time and historical data for all issuers on the Macedonian Stock Exchange.

Data Flow

1. Data Retrieval:

- The Scraper extracts raw stock data from www.mse.mk.
- Scraped data is passed to the Filter for preprocessing.

2. Data Transformation:

- The Filter cleanses and formats the data, ensuring it meets the application's standards.
- Filtered data is stored in the Database.

3. Data Processing:

- The LSTM Module retrieves data from the Database for predictive analysis.
- Predictions are generated and sent back to the Database or directly to the Service Layer.

4. User Interaction:

- The Service Layer processes HTTP requests from the User Interface, interacting with the Database and LSTM Module as needed.
- The User Interface displays the processed and predicted data to the user.

Communication Protocols

- HTTP: Used for communication between the User Interface and Service Layer.
- Internal Calls: Backend components (Scraper, Filter, Database, LSTM Module) communicate using internal APIs or direct function calls.