**Technical Specification for Currency Rate Bot**

**Overview**

This technical specification outlines the structure of a **Currency Rate Bot** built using Spring Boot. The bot interacts with the **Telegram API** to provide users with real-time currency exchange rates, allows them to subscribe to currency pair updates, and offers administrative features for managing users and their subscriptions. The bot is implemented using **TelegramLongPollingBot**, which handles all interactions with users via Telegram.

The bot has two roles:

1. **User**: Can retrieve exchange rates, subscribe to currency pairs, manage subscriptions, and view existing subscriptions.
2. **Admin**: Can view the list of users and their associated subscriptions.

The only non-Telegram interaction is a single **GET** endpoint that provides a JSON array of all supported currency pairs and their current exchange rates.

**Architecture Components**

* **Spring Boot**: Backend framework handling business logic and communication with the database.
* **MySQL**: Database for storing user information, roles, and subscriptions.
* **Telegram API**: The interface used for interacting with users via the Telegram bot.
* **Currency API**: External service providing real-time exchange rates.

**Endpoints**

**1. GET https://www.floatrates.com/daily/eur.json**

* **Method**: GET
* **Endpoint**: **https://www.floatrates.com/daily/eur.json**
* **Description**: Provides a list of all available currency pairs with their current exchange rates in JSON format. This is the only external endpoint.
* **Response Format**:

json

Копировать код

{

"aud": {

"code": "AUD",

"alphaCode": "AUD",

"numericCode": "036",

"name": "Australian Dollar",

"rate": 1.6126395478739,

"date": "Fri, 4 Oct 2024 23:59:00 GMT",

"inverseRate": 0.62010137436997

},

"cad": {

"code": "CAD",

"alphaCode": "CAD",

"numericCode": "124",

"name": "Canadian Dollar",

"rate": 1.4939108794939,

"date": "Fri, 4 Oct 2024 23:59:00 GMT",

"inverseRate": 0.66938397311811

},

...

]

* **Use Case**: This endpoint is designed for external systems or applications that require a list of currency rates. For instance, a frontend dashboard or another service could consume this data.
* **Authentication**: No authentication is required for this endpoint.

**Telegram API Interactions (TelegramLongPollingBot)**

All other interactions are managed through Telegram via the **TelegramLongPollingBot**. The bot listens to user commands and responds with appropriate actions, including querying exchange rates, subscribing, unsubscribing, and administrative tasks.

**User Functionality via Telegram Bot**

1. **Start Command**
   * **Command:** /start
   * **Description:** Initializes the bot interaction for the user.
   * **Example Response:**

*Добро пожаловать в бот курсов валют!  
Здесь Вы сможете:  
- узнать официальные курсы валют на сегодня;  
- подписаться на рассылку интересующих Вас курсов валют;  
- посмотреть свои подписки*

***Посмотреть курс***

***Подписаться***

***Показать мои подписки***

***Отписаться***

***Остановить бот***

1. **Set Base Currency**
   * **Command:** /rateMessage
   * **Description:** The bot asks the user to set base currency and sets base currency.
   * **Example Response:**

*Выберите БАЗОВУЮ валюту,  
либо введите код валюты самостоятельно:*

**EUR USD JPY GBP**

**AUD RUB KZT UAH**

1. **Set Required Currency**
   * **Command:** /setBaseCurrency
   * **Description:** The bot asks the user to set required currency and sets required currency.
   * **Example Response:**

*Выберите валюту,*

*либо введите самостоятельно код валюты,*

*курс которой желаете узнать:*

**EUR USD JPY GBP**

**AUD RUB KZT UAH**

1. **Get Rates**
   * **Command:** /setRequiredCurrency
   * **Description:** The bot shows actual currency rate of the given currency pair.
   * **Example Response:**

*Mon, 4 Nov 2024 07:55:05 GMT:*

*EUR : KZT*

*Курс: 53,.840*

*Обратный курс: 0,002*

1. **Subscribe to Currency Pair Updates**
   * **Command:** /subscribeMessage
   * **Description:** Bot asks to set currency pair for subscription.
   * **Example Response:**

*Введите валютную пару,  
на которую желаете подписаться  
в следующем формате:  
   
EUR:KZT*

1. **Successfully Subscribed to Currency Pair Updates**
   * **Command:** /subscribe
   * **Description:** Bot sends message with successfully subscribed currency pair.
   * **Example Response:**

*Вы успешно подписались на:  
   
EUR:KZT*

1. **Unsubscribe from Currency Pair Updates**
   * **Command:** /unsubscribeMessage
   * **Description:** Bot asks to set currency pair for unsubscription.
   * **Example Response:**

*Введите валютную пару,*

*от которой желаете отписаться*

*в следующем формате:*

*remove(EUR:KZT)*

1. **Successfully Unsubscribed from Currency Pair Updates**
   * **Command:** /unsubscribe
   * **Description:** Bot sends message with successfully subscribed currency pair.
   * **Example Response:**

*Вы успешно отписались от:  
   
EUR:KZT*

1. **View Subscriptions**
   * **Command:** /subscriptions
   * **Description:** Lists all the currency pairs the user is currently subscribed to.
   * **Example Response:**

*Вы подписаны на:*

*USD:EUR*

*GBP:USD*

1. **Stop Command**
   * **Command:** /stop
   * **Description:** Stops all interactions with the bot.
   * **Example Response:**

*Спасибо, что воспользовались нашим сервисом!*

*Не забудьте подписаться на рассылку интересующих вас курсов валют!*

**Admin Functionality via Telegram Bot**

**1. Send Administrative Information**

* **Command:** Automatically scheduled every day.
* **Description:** Sends a formatted message to all admin users, containing:
  + Names of all active subscriptions.
  + Total number of users.
  + Number of users with active subscriptions.

**Data Flow**

1. **Currency Data Fetching**:
   * The bot uses a **Currency API** to fetch real-time exchange rates. This is done asynchronously when a user requests the exchange rate or subscribes to updates.
2. **Telegram Interaction**:
   * User commands are processed by the TelegramLongPollingBot. Based on the command, the bot interacts with the services (CurrencyService, SubscriptionService, AdminService) to retrieve the necessary data and return a response.
3. **Database Operations**:
   * User information, subscriptions, and currency pairs are stored in a **MySQL** database. All user and admin commands involve querying or updating this database.
4. **Subscription Management**:
   * The bot sends daily updates to users who have subscribed to currency pairs using scheduled jobs, leveraging Spring's **@Scheduled** annotation to automate this process.

**Error Handling**

* **Invalid Currency Pairs**: If a user requests a currency pair that is not supported, the bot will respond with an error message.

**Security**

* **Token-based Authentication**: The bot uses a **Telegram Bot Token** for authentication with the Telegram API. This token is stored securely in the application configuration.

**Conclusion**

This bot integrates with the Telegram API to provide a seamless user experience for retrieving exchange rates and managing currency pair subscriptions. The entire interaction is handled through the Telegram chat interface, with one public GET endpoint for external currency rate queries. The bot also offers administrative tools for managing users and their subscriptions.