Forex Analytics

Introduction

1.1 Purpose

The purpose of this project is to build an AOS (Automated Trading System) for broker xtb trading.

1.2 Scope of the project

The purpose of Forex Analytics is to create an app that trades on the Forex stock exchange.

Forex is forex trading and is also the largest and most liquid financial market in the world. On the server side, the application will collect data and communicate with the Xtb server. From these collected data, the app will analyze currencies and trades, and will continue to open or close a shop based on the algorithm that the system will contain. Above all, we hope to provide a comfortable user experience and increase user profits on the stock market.

 Overall description

 2.1 PRODUCT PERSPECTIVES

Forex analytics stores the following information

User description

Email and login password.

Description of the logged-in user

Account number, Email and password to log in to xtb. The account number is generated and sent to the user's e-mail when registering on xstation5.xtb. This app data stores only with the user's permission to sign up for the application.

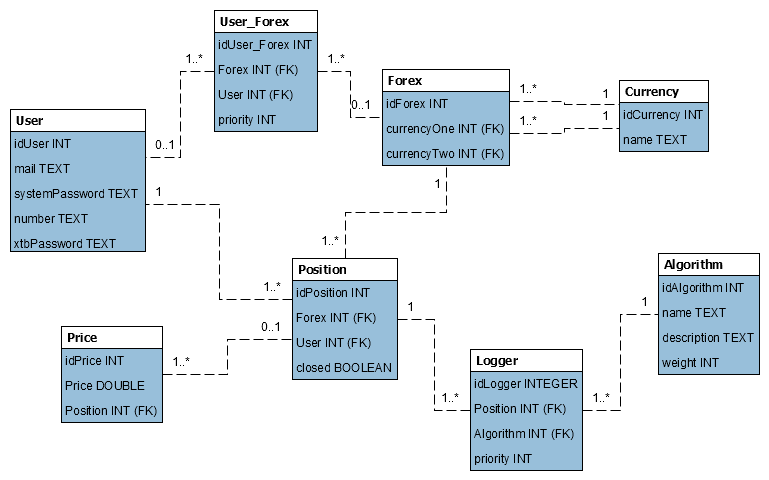
Business Description

Store status, trade opening price, Forex trade. Store status indicates whether the store is open or closed.

History of shops

Forex Price When Opening Trade Up to Close Trade, Opening and Closing Time, Forex Trading Type, Trade Status.

2.2 ER Model (UML Class Diagram)



2.3 USER CLASS and CHARACTERISTICS

Logged in user

Users should be able to choose the currency in which they will trade. Based on the selected currency, the app will analyze and then begin trading. Open trades that have a different currency will close until the application evaluates it as appropriate. The customer will be able to see the status of his account and its ongoing changes.

The user will be able to log in to the system. Registering the user will be the administrator. Upon registration, the user will have the choice of whether the system can store logins on the xtb server. If a user accepts a user's application automatically logs on to the xtb server (if the data is already stored in the database), if it rejects, the user will have to fill in the logins to login to the xtb server.

Unlogged user

An unlogged user will be able to sign in.

2.4 OPERATING ENVIRONMENT

The operating environment for Forex analytics is listed below.

• Client / Server / AOS system

• SQL database

• Platform: Java, Spring, React

3. System Functions

3.1 Description

Forex analytics keeps user information, Forex type. Furthermore, the system stores Forex and the price of the individual Forex traded or traded.

3.2 Functional Requirements

Other features of the system include:

The client / server system

The term client / server refers primarily to architecture or logical division of responsibility, the client's application (also known as front-end), and the server is DBMS (also known as back-end). All data is on the server side of the system. Some data (e.g., account status, whether the user is earning or earning) are being sent to the client.

4. External interface requirements

4.1 User Interface

• Front-end software: React

• Back-end software: Java Spring

• AOS software: Java

4.2 Hardware interface

• Browser that supports HTML, CSS, and Javascript

4.3 Software interface

• We chose the Windows and Linux operating system.

• We chose the Postgres SQL database to store currency, currency, store, user, store status and more.

• We chose Java, HTML, CSS, and Javascript to implement the project

4.4 Communication interface

This project is designed for Mozilla Firefox and Chrome browsers. We use simple forms to sign in, register, and choose the type of currency to trade with.

5. System constraints

The system is not designed to open and close a shop at customer's discretion. The customer will not be able to start an app in the app, leave it open as it sees fit, or end it when it wants. These functions perform applications for users based on algorithms.