Algo-trading market emulator HLD

For terminology and other relevant information please refer to the server HLD document.

<https://goo.gl/xUwXR2>

The project id delegated to three sub-projects(layers) containing the following:

**Presentation layer:**

This layer consists the GUI of the App.

In the Main Menu the user is able to file one of the following requests:

* Buy Commodities
* Sell Commodities
* Cancel a Buy/Sell Request
* Query Buy/Sell – provides information about a specific active request
* Query Market – provides information about a specific commodity
* User Query – provides information about the user – owned commodities, funds, requests
* All Market Query – shows information about **all** of the commodities
* User Requests Query – shows information about **all** the active requests of the user
* Market Statistics – shows graphs of minimum maximum and average prices of the commodities over a certain time period
* Generate PDF – generates a PDF file with the current user query data written in it

**Business layer:**

This layer mainly acts as the "intermediary" between the Presentation layer and the Data Access layer. It takes the requests the user file and sends them to the DAL for processing, and then returns an adequate message back to the user. This layer also holds the Autonomous Market Agent which consists an automatic agents which buys and sells commodities by permanent ground rules.

**Classes:**

Request Agent – where all the "mediation" takes place. Consists function which take user input and sends it to the DAL.

Autonomous Market Agent – where all the automat actions take place. This class connect with the server while the user is not using the GUI and knows to make 20 different actions accordingly to the rules we decided. The main we that we decide whether to buy oר sell a stock, is first calculate if in the last 20 deals from the last 5 days the stock in on ascent or descent, if on ascent then we will sell the stock with some conditions, and otherwise we will decide to buy the stock with some condition.

Stats Manager – in this class holds all the actions required for receiving the stats for the Market Statistics. It communicates with the SQL server.

**Data Access layer:**

The job of this layer is to communicate with the server. The layer process the requests which is coming from the user (the presentation layer), send it to the server with proper variables and then process the server response and sends it back to the presentation layer.

This layer also communicates with the SQL server.

**Classes:**

Buy Request\Sell Request\Cancel Request- these three classes are about the simple request of the user to buy, sell or cancel the last request.

Query Market Request\Query User Request\Query Sell Buy Request- this classes returns information about the current data of the market, the user request and the sell\buy request.

Query Sell Buy Request\Market Item Query\Market User Data- this classes present the required information as we get from the user.

SQL Manager – this class is responsible of creating a SQL connection to the SQL server, and allowing the BL to use this connection.

**Open Sources:**

Live Charts – a library used for generating various charts. In this project it is used to create the Market Statistics Charts.

I Text Sharp – a library used for generating a pdf file easily. In this project it is used to generate the User Report