Stock Trading Application

Created By	To be Reviewed By	Date Submitted	Changes Made	Version
Kalaivani KG	Trevor Doto	02/04/2023	Initial Draft	v0.1
Kalaivani KG	Trevor Doto		Stakeholder contact information, project scope, responsibilities, added web source to system overview, grammatical correction of functional requirement	v0.2

Table of Contents

1. Introduction

- **Background**

Our organization is seeking a new Stock Trading Application that replaces an old legacy trading platform.

We aim to have the first look at the application launched within the second quarter and will evaluate and implement the application by May 15th, 2023.

The targeted date for the full application is planned for Q1 of 2024 whereas the first look of the application (working prototype) with a basic requirement we're looking to satisfy, including traders' login, traders to buy/sell stocks, checking the cost of stocks and add it for different accounts they manage is planned for Mid-May 2023. The application should be easy to scale and flexible to add requirements in the future. The backend team takes the responsibility to develop and test the application.

- **Stakeholders**
  - **Project Manager**
    - Name : Trevor Doto
    - Responsibility: Responsible for holding all parties accountable to the project timeline
    - Contact no: +1 4379023489
    - Email id: trevor.doto@jarvis.ca
  - **Backend team**
    - Members: Ashwin Raj, Angel, Gopal, Emma Jane
    - Responsibility: Responsible for developing and testing the backend components. Once it is completed backend team would report back to the project manager.
    - Email id: Backend_team@jarvis.ca

- **Project Scope**

We want to implement an MVP for the following key components

- In Scope

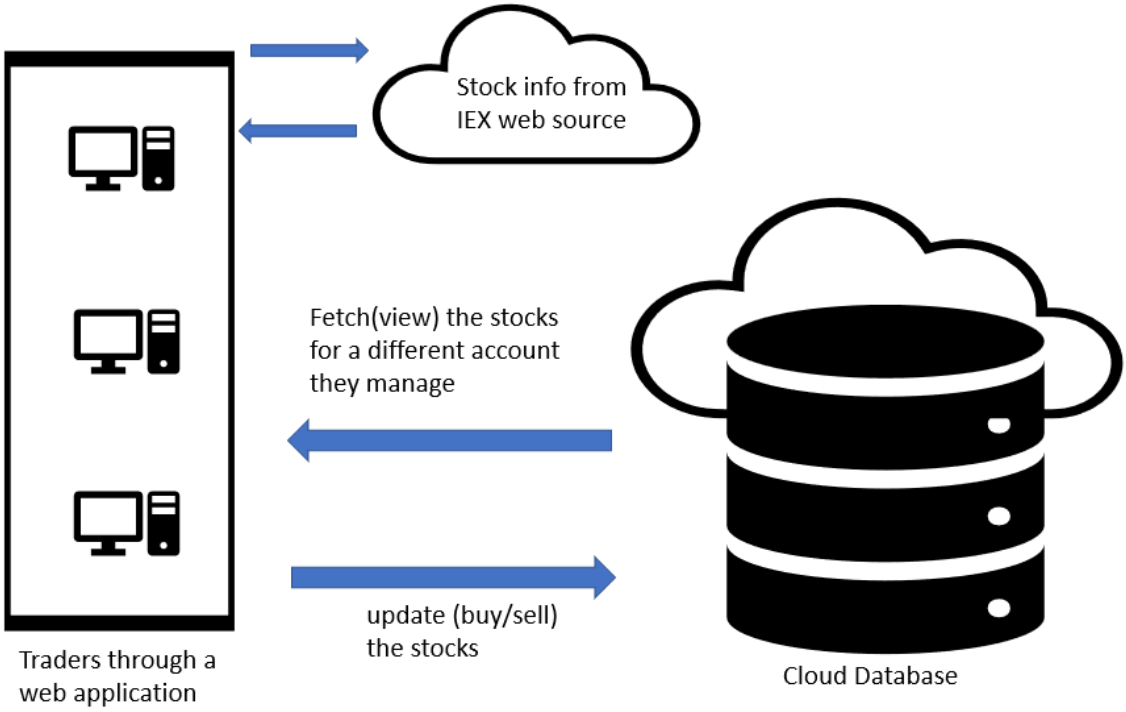
Create an application that has a login page with username and password requirements. Users(traders) should be able to pull out different accounts for their clients (investors). Traders will be allowed to pull-out quotes for stocks that are available. Traders have the option to buy and sell stocks.

- Out of scope
  - Application with sign-up option on the login page
  - Mobile application that can do project requirement

- Complex stock trading
- Dashboard creation, which comes under front-end development

- **Responsibilities:**
  - Create an application with login page that authorize user who has entry in the database
  - Display the available stocks and their quotes for the account that is selected.
  - The stock page should have the option to buy or sell stock
  - According to the requirement and the option selected the data should be updated in the database

2. System Overview



- Traders use web browsers to log in to the stock trading application
- On the successful login the user has the option to fetch stock quotes from the cloud (IEX web source) for the different accounts they manage for the investors they have
- Users also have the option to buy/sell stocks and update the same in the database

3. Functional Requirement

- Stock trading application will be authorize the users who has record in database

- With successful log-in, users will be able to access various accounts created for each investors
- They will be allowed to pull-out quotes for stocks that are available
- The user can see the stocks position for all the different accounts they hold.
- On selecting the stocks, user will be allowed to buy or sell the stocks they have in their account
- The cloud database will be updated with the stock information based on the transaction made

4. Non-Functional Requirement

- **Reliability**
  - The web application will be completely operational at least 99% of the time
  - The database will be completely operational at least 99.99% of the time.
- **Usability**
  - Login credentials are migrated from previous application. As user have an experience with legacy application, they were expected to use the application with minimum to no training
- **Performance**
  - User will able to buy and sell stocks in real time
  - There will be no major latency or delay as it is time sensitive
  - It will be accessible almost all the time
- **Security**
  - The website is only accessible to authorized traders
  - Database and transaction must be encrypted
- **Supportability**
  - The website will be viable from any standard web browser
- **Documentation and Help**
  - Manual must be documented and stored in corporate Notion/Confluence/Wiki site
- **Interfaces**
  - Web portal
  - English and French

5. Use Cases

Use Case Name	Traders authorized to web application
Basic Flow	1. Traders will get their credentials directly for authorization 2. The login user credentials will be verified with the cloud database during application authorization 3. Application will be accessible to any standard web browser

Use Case Name	Users check on quotes for stocks
Basic Flow	1. On successful login, the user will be allowed to view stocks 2. They will able to pull out quotes for all stocks they maintain for different accounts appropriate to their client 3. The data were pulled from the cloud database

Use Case Name	Users buy/sell stocks
Basic Flow	1. On successful login, the user will be allowed to view stocks 2. They will able to pull out quotes for all stocks they maintain for different accounts appropriate to their client 3. The data were pulled from the cloud database 4. The user will have the option to buy/sell stocks that they have pulled out. 5. According to the option selected the data will be updated in the database

6. Data Requirements/Modelling

a. Entities

Entity Name: Traders

Column Name	Data Type	Nullable	Primary Key	Foreign Key	Description
Trader_id	int	Not Null	Yes		
trader_name	varchar	Not Null			
email	varchar	Not Null			
password	varchar	Not Null			
Phone_no	int	Not Null			

Entity Name: Investors

Column Name	Data Type	Nullable	Primary Key	Foreign Key	Description
Investor_id	int	Not Null	yes		
Investor_name	varchar	Not Null			
Trader_id	int	Not Null		yes	Referring to traders(trader_id)
account_id	INT	Not Null		yes	referring to accounts(account_id)

Entity Name: Accounts

Column Name	Data Type	Nullable	Primary Key	Foreign Key	Description
Account_id	int	Not Null	yes		

account_name	varchar	Not Null			
Trader_id	int	Not Null		yes	Referring to trader(trader_id)

Entity Name: Quotes

Column Name	Data Type	Nullable	Primary Key	Foreign Key	Description
Stock_id	int	Not Null			
Account_id	int	Not Null		yes	Referring to accounts(account_id)
quotes	float	Not Null			

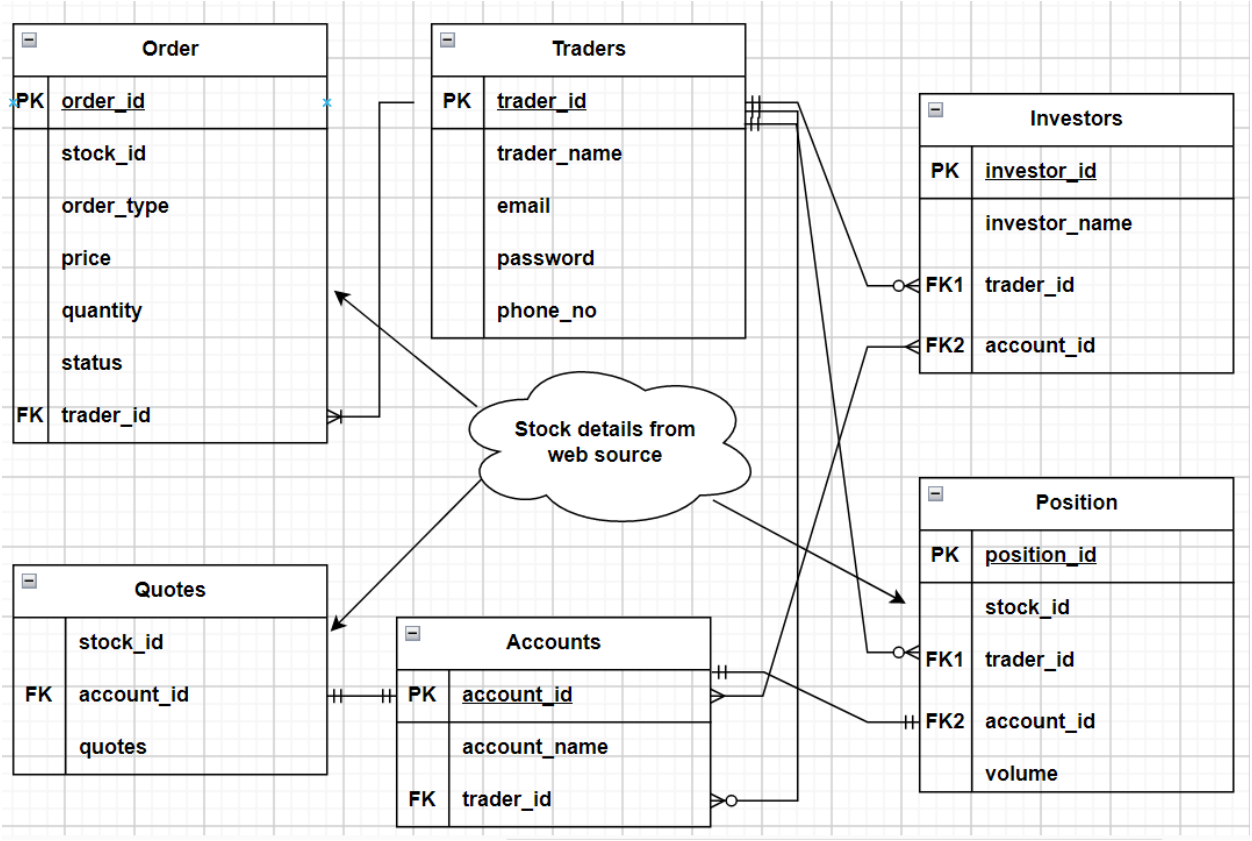
Entity Name: Position

Column Name	Data Type	Nullable	Primary Key	Foreign Key	Description
position_id	int	Not Null	Yes		
Stock_id	int	Not Null			
Account_id	int	Not Null		yes	referring to accounts(account_id)
Trader_id	int	Not Null		yes	referring to traders(trader_id)
Volume	float	Not Null			

Entity Name: Order

Column Name	Data Type	Nullable	Primary Key	Foreign Key	Description
order_id	int	Not Null	Yes		
Stock_id	Int	Not Null			
Order_type	varchar	Not Null			
price	float	Not Null			
quantity	Int	Not Null			
status	varchar	Not Null			
Trader_id	Int	Not Null		yes	Referring to traders(trader_id)

b. Entity Relationship



Entities:

1. Traders: The person who uses the app to trade stocks and manage the different accounts for investors.

2. Investors: The person who are the clients of traders.

3. Accounts: Number of accounts managed by traders for their client

4. Order: This entity has information on what stocks are bought and sold.

5. Position: The position table has information on who owns how much of stocks.

6. Quotes: It has quote information for the stocks.

Relationship:

1. Traders-investors: one-to-many relationship, where one trader can have multiple investors

2. Traders-order: one-to-many relationship, where one trader can have multiple orders

3. Investors-accounts: many-to-many relationship, multiple investors have multiple accounts

4. Trader-position: one-to-many relationship, one trader will have many accounts with a position
5. accounts-position: one-to-one relationship, one account will have one position
6. trader-accounts: one-to-many relationship, one trader will have many accounts
7. Quotes-accounts: one-to-one relationship, stock of one account will have one quote.
8. Orders-Quotes-position-web source: stock information is from a web source.