**Proiect PCBE**

**Bursa**

**Timisoara Stock Exchange**

Echipa:

* Danci Ionut-Cosmin
* Dragoescu Cristian-Florin
* Dumitrescu Bogdan-Doru

Cuprins :

1. Descriere
2. UML
3. Tools
4. Link repository
5. Probleme concurenta
6. Functionalitati nou-incluse
7. Link-uri pentru commit-uri

8. Explicatii si observatii

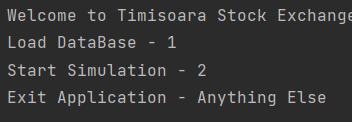
1. Descriere aplicatie

Am stabilit de comun acord arhitectura proiectului :

- un sistem distribuit (server si clienti):

in partea de server ne ocupam de thread-uri si logica aplicatiei; iar in partea de client exista 2 clase (buyer si seller) unde se realizeaza comunicarea pe socket cu serverul.

La executia programului se vor afisa mai multe optiuni intr-un meniu din terminal.



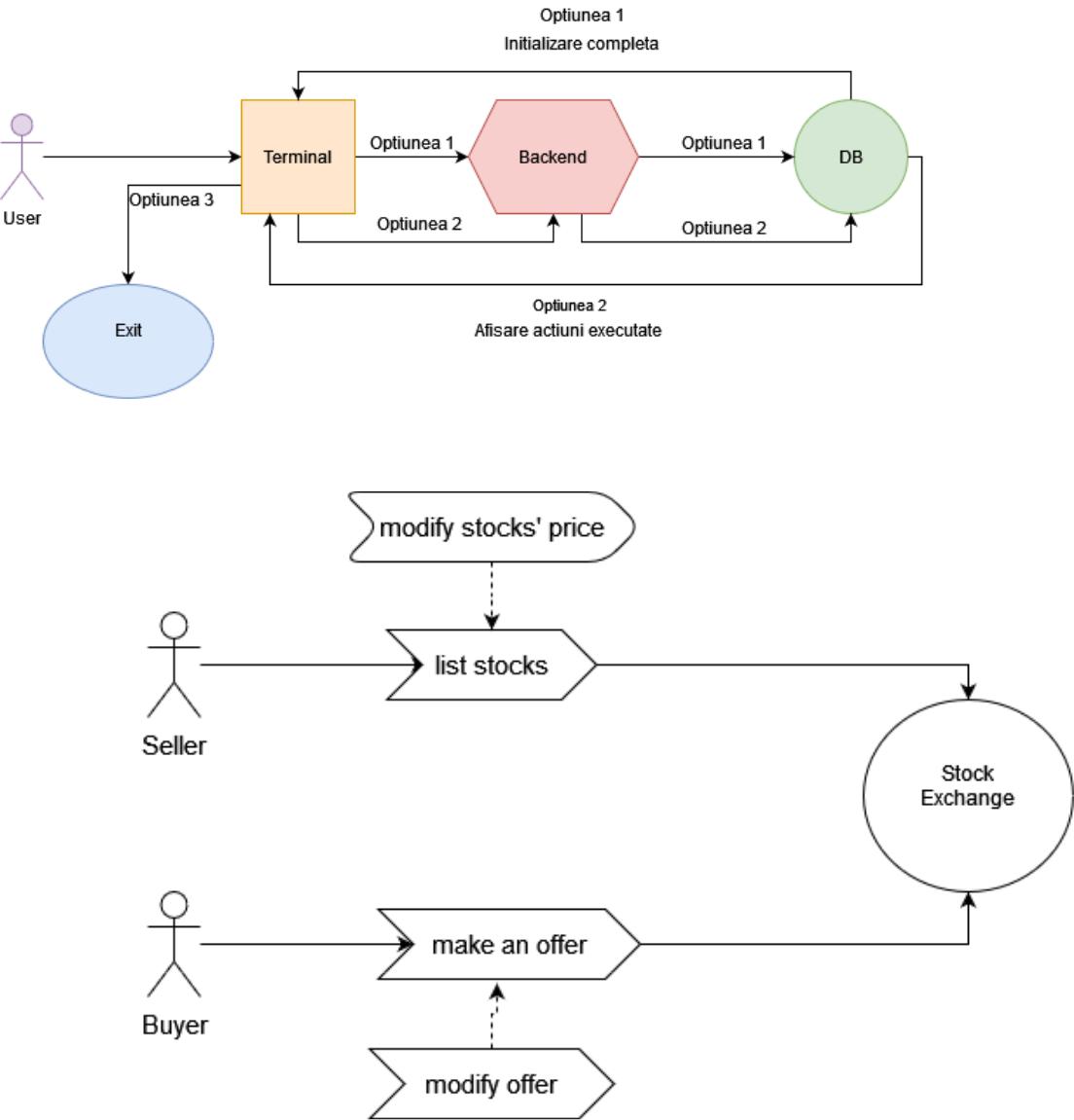
Prima optiune din meniu este cea de initializare a campurilor unor obiecte. In baza de date vor exista anumite actiuni, iar la initializare se vor aloca un anumit numar de actiuni la fiecare seller.

A doua optiune este cea de simulare. Aceasta va genera evenimentele si thread-urile. Dupa fiecare tranzactie efectuata, baza de date va fi initializata. Aici se vor putea observa situatiile de concurenta in terminal in forma de log-uri(reprezentand diverse informatii despre buyer, seller si tranzactii). Daca obiectele nu sunt initializate folosind prima optiune, aceasta optiune nu poate fi executata.

A treia optiune este cea de afisare a istoricului tranzactiilor.

A patra optiune este cea de oprire a programului.

2. UML



3. Tools

Tehnologii: Java, Maven

Baza de date aleasa: SQLite

Specific algorithms used: citim in buffered reader

Tools used for development: Intelij, Git, GitHub, RabbitMQ

1. Link repository

https://github.com/cristiandrgg/Proiect\_PCBE.git

5. Probleme concurenta

In momentul in care se realizeaza o tranzactie, istoricul trebuie pastrat, si pana ce se finalizeaza, toate celelalte actiuni o sa fie blocate (sa nu se poata face alte oferte).

O alta problema de concurenta apare la modificarea preturilor actiunii si a sumei ofertei. Daca sumele nu sunt egale, pretul ofertei scade, iar pretul actiunii creste. Exista riscul ca acestea sa nu se intersecteze niciodata. Pentru a rezolva aceasta problema, modificarea trebuie facuta doar la actiune sau doar la oferta.

O problema de sincronizare apare atunci cand mai multi cumparatori liciteaza pentru aceasi actiune la preturi diferite. Acest mecanism blocheaza executia thread-ului respectiv pana in momentul in care se ajunge la pretul dorit de vanzator.

1. Functionalitati nou-incluse

* BuyerProcess
  1. run() method
* SellerProcess
  1. run() method
* Server
* Seller
  + Constructor
  + Setters and Getters
* Buyer
  1. Constructor
  2. Setters and Getters
* Bid
  1. Constructor
  2. Setters and Getters
* Stock
  1. Constructor

1. Setters and Getters

* SqliteDB
  + getBuyers()

Text

Description automatically generated

* + getSellers()
  + getStocks()

Text

Description automatically generated

* + getBids()
  + updateStock()
  + updateBid()

Text

Description automatically generated

Text

Description automatically generated

* + simulation

Text

Description automatically generated

* meniu options and database loading

Text

Description automatically generated

* event message sender function

Text

Description automatically generated

* show history function

Text

Description automatically generated

* event message consumer function

Text

Description automatically generated

* simulation for buyer/seller

Text

Description automatically generated

Text

Description automatically generated

* exchange function concurrency

Text

Description automatically generated

* sql querry example from exchange function

7. Link-uri pentru commit-uri

|  |  |
| --- | --- |
| **Autor** | **Descriere & Link** |
| DumitrescuBogdanDoru (Dumitrescu Bogdan-Doru) | [M-1: Milestone1 documentation added](https://github.com/cristiandrgg/Proiect_PCBE/commit/3174aa69968f049fe3b5853a12193d1c51527dbf) |
| DumitrescuBogdanDoru (Dumitrescu Bogdan-Doru) | [M-2: empty project created](https://github.com/cristiandrgg/Proiect_PCBE/commit/8556b5203e096cc8348b5e898173531392f14efb) |
| EvoShady (Danci Ionut-Cosmin) | [Moved .gitignore to root](https://github.com/cristiandrgg/Proiect_PCBE/commit/4063960c32eda6c7de4b06c73def266991915bde) |
| cristiandrgg (Dragoescu Cristian-Florin) | [M-2: added Bid](https://github.com/cristiandrgg/Proiect_PCBE/commit/8b248d69413c56ebed65c0aa883ba9a0a61cc0f8) |
| cristiandrgg (Dragoescu Cristian-Florin) | [M-2: added Seller](https://github.com/cristiandrgg/Proiect_PCBE/commit/177e11e943a06cb6fd40cccdf59fd0fa5836b9f2) |
| cristiandrgg (Dragoescu Cristian-Florin) | [M-2: added Stock](https://github.com/cristiandrgg/Proiect_PCBE/commit/c480ba0ec20e16ff6929e21ab8aca38b2d77d75b) |
| DumitrescuBogdanDoru (Dumitrescu Bogdan-Doru) | [M-2: added SellerProcess](https://github.com/cristiandrgg/Proiect_PCBE/commit/c74fd21c6b71a39322fb54b1d42fa68f271599d8) |
| DumitrescuBogdanDoru (Dumitrescu Bogdan-Doru) | [M-2: added SellerThread](https://github.com/cristiandrgg/Proiect_PCBE/commit/dd9fae02a343e27b6f9cce5b469d65772e0269b4) |
| EvoShady (Danci Ionut-Cosmin) | [M-2: Added Buyer Model](https://github.com/cristiandrgg/Proiect_PCBE/commit/c7d583d4e8bf88363b09387a33943d1183207f46) |
| EvoShady (Danci Ionut-Cosmin) | [M-2: Added Buyer Thread](https://github.com/cristiandrgg/Proiect_PCBE/commit/bf3188edee3cda11798edd31312067c83b4d0bf3) |
| DumitrescuBogdanDoru (Dumitrescu Bogdan-Doru) | [M-2: modified Server](https://github.com/cristiandrgg/Proiect_PCBE/commit/e361b6537f7c10fd09e247604f81c2287a5f4976) |
| DumitrescuBogdanDoru (Dumitrescu Bogdan-Doru) | [M-3: connected to a db](https://github.com/cristiandrgg/Proiect_PCBE/commit/bde002398c6a2a014200d954f5b0aa1fff920fe0) |
| DumitrescuBogdanDoru (Dumitrescu Bogdan-Doru) | [M-3: modified stock, bid, server](https://github.com/cristiandrgg/Proiect_PCBE/commit/2cd49265be8f8d270f36014931b8b014c20d5717) |
| EvoShady (Danci Ionut-Cosmin) | [M-3: Removed BuyerThread, SellerThread. Added run method in BuyerProcess, SellerProcess, updated Server](https://github.com/cristiandrgg/Proiect_PCBE/commit/ce8057fabc23b4d104d9d466c33c8b4609b73370) |
| DumitrescuBogdanDoru (Dumitrescu Bogdan-Doru) | [M-3: add methods for updating stock and bid price](https://github.com/cristiandrgg/Proiect_PCBE/commit/92dd1bab51a73cfe1a89dc61fee2dd274e77ab22) |
| cristiandrgg (Dragoescu Cristian-Florin) | [M-4: added menu and other functionalities](https://github.com/cristiandrgg/Proiect_PCBE/commit/7535e85d0e6c38c33eb6dec66ea922a4c911cf9f) |
| EvoShady (Danci Ionut-Cosmin) | [M-4: Fixed Server Child Thread List](https://github.com/cristiandrgg/Proiect_PCBE/commit/89a4ff2c387e441cb811883e8da9122056a77859) |
| DumitrescuBogdanDoru (Dumitrescu Bogdan-Doru) | [M-4: buyers' bid list bug fixed](https://github.com/cristiandrgg/Proiect_PCBE/commit/953f47cd1cd473f83fe95dbe1e3e3914fd3d0bab) |
| EvoShady (Danci Ionut-Cosmin) | [M-4: Updated update method for BuyerProcess and SellerProcess](https://github.com/cristiandrgg/Proiect_PCBE/commit/33d88d31a04e0ea2d84a8a4abc9cb6ed37c1d934) |
| DumitrescuBogdanDoru (Dumitrescu Bogdan-Doru) | [M-4: exchange method added](https://github.com/cristiandrgg/Proiect_PCBE/commit/a893d35ebb3f42814c741c94a1259a132a48d850) |
| cristiandrgg (Dragoescu Cristian-Florin) | [M-4: added mutex & semaphore](https://github.com/cristiandrgg/Proiect_PCBE/commit/22a1c3fb7e82d4a990faaec0c6a0b8acbe668f1e) |
| EvoShady (Danci Ionut-Cosmin) | [M-4: Added RabbitMQ and Consumer Test Class](https://github.com/cristiandrgg/Proiect_PCBE/commit/7d15eb981bd02d388ece4f452448bab22c6dea7b) |
| DumitrescuBogdanDoru (Dumitrescu Bogdan-Doru) | [M-4: added event driven messages](https://github.com/cristiandrgg/Proiect_PCBE/commit/42f3766c6e1dca0d9df8aa2e32d403cf4aa76a80) |
| EvoShady (Danci Ionut-Cosmin) | [M-4: Added Semaphore for self-concurrency in exchange](https://github.com/cristiandrgg/Proiect_PCBE/commit/3d513688080a7e6450b3e62a2ed125b757033739) |
| DumitrescuBogdanDoru (Dumitrescu Bogdan-Doru) | [M-4: added History](https://github.com/cristiandrgg/Proiect_PCBE/commit/79263afde33bcdfafda03343a72c62840db1f2fc) |
| DumitrescuBogdanDoru (Dumitrescu Bogdan-Doru) | [M-4: modified db](https://github.com/cristiandrgg/Proiect_PCBE/commit/6c9efcd1925a40e2a289d29f9ddd52fccbddffab) |

1. Explicatii si observatii

Câteva observații relative la mecanismul synchronized:

* + O metodă sincronizată apelată pe două obiecte diferite, în două thread-uri diferite, nu va sincroniza cele două thread-uri, pentru că blocajele se vor obține independent pe două obiecte diferite.
  + Blocarea se face totdeauna pe obiectul fizic și nu pe referințe la acesta.
  + Metodele sincronizate se scriu, în general, în afara claselor thread.

Mecanisme de sincronizare a thread-urilor:

Thread-urile sunt folosite în programare pentru a eficientiza execuția programelor executând porțiuni distincte de cod in paralel prin intermediul aceluiași proces. Uneori aceste porțiuni de cod care formează corpul thread-urilor nu sunt complet independente și în anumite momente ale execuției se poate întâmplă că un thread să trebuiască să aștepte execuția unor instrucțiuni din alt thread pentru a-și putea execută propriile instrucțiuni.

Tehnica prin care un thread așteaptă execuția altor thread-uri înainte de a-și continuă propria execuție se numește sincronizare. În java avem următoarele mecanisme de sincronizare ale thread-urior:

* + Mecanismul Synchronize
  + Metodele Wait, Notify si Notify All

