

Trading Simulation

Inhalt

Description 2

Use Cases 3

 Use Case-Diagramm 3

 Use Case-Description 3

CRC-Cards 7

Class Diagramm 9

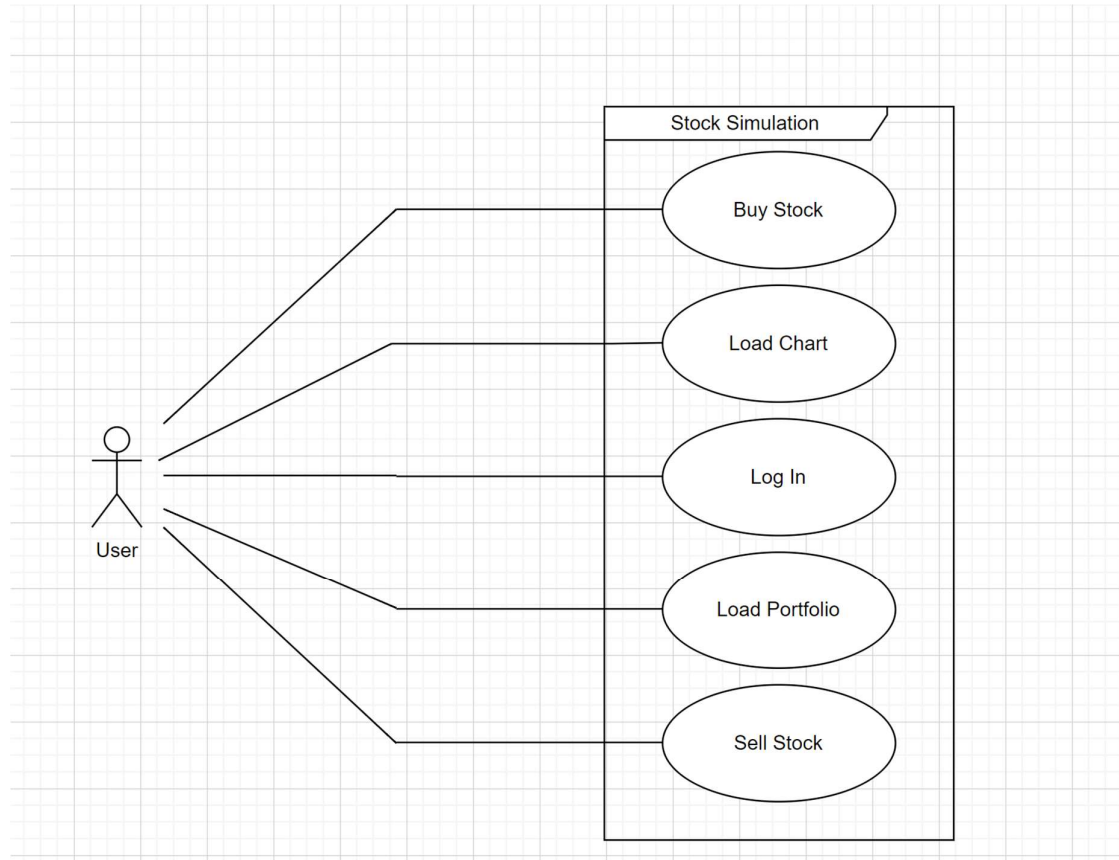
Domain Model 10

Description

Our program is intended to be a stock trading trainer/simulator. For the user it is possible to log in and have a personal portfolio. Since this project does not use a database, the portfolio gets saved into a file after a transaction was executed or the program got closed. The user can buy and sell stocks in the stock market and the stocks will appear in the user profile page. Also, there is an option to search for a specific stock and look at its stats and a chart. Next to the stock and user page there will be a page to display all stocks and the base information like price etc. The goal of this simulator is to gain as much money as possible.

Use Cases

Use Case-Diagramm



Use Case-Description

Use Case	Login
Use Case ID	1
Description	The user can log into his account
Precondition	User has an account
Actors (Primary)	user
Procedure: <ol style="list-style-type: none"> 1. Enter username and password into text fields 2. Click "Login" 	
Postcondition:	User has access to his account
Alternative flows: Wrong credentials	

Use Case	Wrong credentials
Use Case ID	1.1
Description	The user tries to log into an account with wrong credentials
Precondition	none
Actors (Primary)	user
Procedure: <ol style="list-style-type: none"> 1. Enter username and password into text fields 2. Click "Login" 3. User gets informed that his login data was wrong 	
Postcondition:	none

Use Case	Load Chart
Use Case ID	2
Description	The user loads a chart from a stock
Precondition	The user is authenticated, the stock is available in the system
Actors (Primary)	user
Procedure: <ol style="list-style-type: none"> 1. Search for a stock 2. The chart of the stock should be displayed 	
Postcondition:	none

Use Case	Buy a stock
Use Case ID	3
Description	The user can buy a specific stock
Precondition	The user is authenticated
Actors (Primary)	user
Procedure: <ol style="list-style-type: none"> 3. Load chart of desired stock 4. Click "Buy Stock" 5. Money is withdrawn from user 6. Share gets transferred to user's portfolio 	
Postcondition:	User owns shares

Alternative flows: Not enough money

Use Case	Not enough money
Use Case ID	3.1
Description	The user buys a specific stock
Precondition	The user is authenticated
Actors (Primary)	user
Procedure: <ol style="list-style-type: none"> 1. Alternative flow starts after step 2 in the Main flow 2. Money cannot be withdrawn because user hasn't enough money 3. Transaction stops 	
Postcondition:	none

Use Case	Sell a stock
Use Case ID	4
Description	The user sells a specific stock
Precondition	The user is authenticated, owns shares of stock to sell
Actors (Primary)	user
Procedure: <ol style="list-style-type: none"> 1. Load chart of desired stock 2. Click "Sell Stock" 3. Share gets withdrawn from user 4. Money gets transferred to user's portfolio 	
Postcondition:	none
Alternative flows: none	

Use Case	Load Portfolio
Use Case ID	5
Description	The user retrieves his own portfolio to see his shares and his balance
Precondition	The user is authenticated

Actors (Primary)	user
Procedure: <ol style="list-style-type: none"> 1. Load Portfolio 2. The balance and owned share of the user are displayed 	
Postcondition:	none
Alternative flows: none	

CRC-Cards

Class: Stock	Collaborators
<hr/> Responsibilities - Blueprint of Stock	

Class: Portfolio	Collaborators Share
<hr/> Responsibilities - Holds stocks	

Class: Display	Collaborators UserDisplay StockOverviewDisplay StockDisplay Searchbar
<hr/> Responsibilities - UI	

Class: Filehandler	Collaborators
<hr/> Responsibilities - read and write to file	

Class: StockDisplay	Collaborators Stock
<hr/> Responsibilities - Display a specific stock	

Class: StockOverviewDisplay <hr/> Responsibilities - Displays a list of stocks	Collaborators Stock Searchbar
--	--

Class: UserDisplay <hr/> Responsibilities - Displays the user and his portfolio	Collaborators User Portfolio
---	---

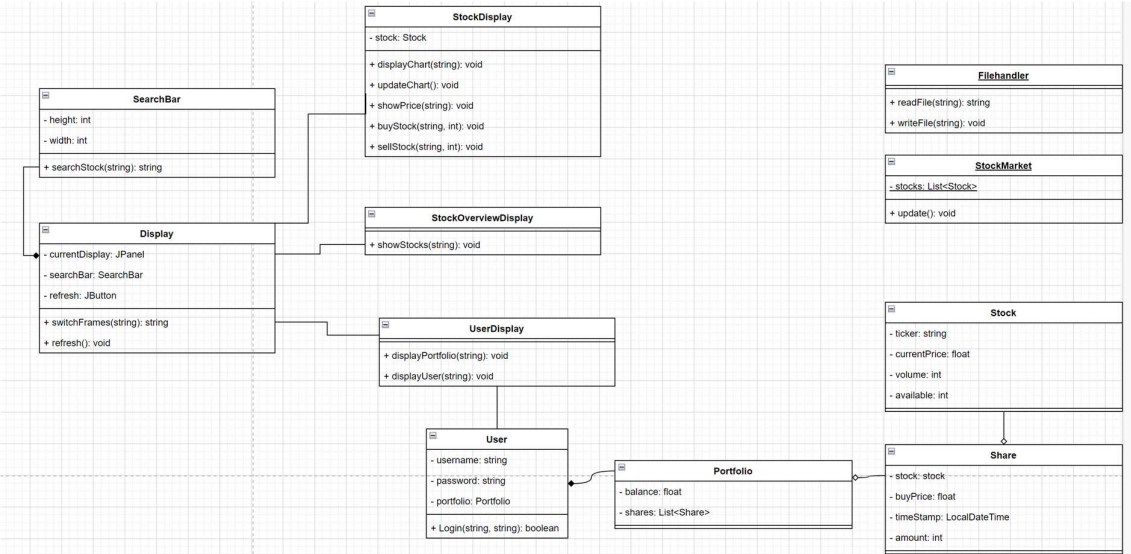
Class: Searchbar <hr/> Responsibilities - Option to search for a stock	Collaborators Stock
--	-------------------------------

Class: User <hr/> Responsibilities - User credentials	Collaborators Portfolio
---	-----------------------------------

Class: <hr/> Responsibilities - holds stock, buying price and amount of held stock	Share Collaborators Stock
--	---

Class: <hr/> Responsibilities - holds all stocks - updates stocks	StockMarket Collaborators
--	--

Class Diagramm



Domain Model

