

TradersGem: Simple, Functional, Valuable.

Luiz Medeiros

Pedro Miranda

Analysis Phase:

Description

The application includes the ability to follow stocks. Moreover, it allows the user to create watch lists. Here, the user is able to add notes to the stocks and pose it in a priority list. There is also the possibility to see the graphs of the stocks.

With the graphs, you have the chance to see multiple types of curves and graphing styles based on the functions and mathematical models available (ex. MFI, RSI, EMA, etc...).

Furthermore, the app allows the user to create portfolios that offers the options to track the buys, sells and short sells. There is also the ability to track the profits, losses and percentage gains/losses.

Through such features, the user is able to keep track of his/her investments as well make notes, see current market analysis, and make real time decisions on whether to buy or sell a particular holding.

Now, the application will also provide the ability to simulate the stocks based on certain parameters. For example, if you buy and sell for a certain price, what is your profit. Based on current market numbers, is this a buy, hold or sell.

This simulation will be applied through financial mathematical models and algorithms such as the ones mentioned above.

Screening for stock based on given parameters is also a possibility with this application.

Hence, the overall goal of the application is to be able to keep a trader fully informed and educated regarding his current position and probable position on any one current or possible trade. It is to be a tool that will provide valuable information for the trader on the go, so that an opportunity will not be missed and profits may continue to flourish.

Use Cases:

Actors: Represent

- 1) Log in the app
 1. User clicks on the application
 2. Application launches and inquires for log in ID and password

3. User enters the information in the fields provided
4. Application checks user's credibility
5. Verification succeeds, users is presented with the home screen displaying the main portfolio.
- 2) Screen for Stocks:
 1. User succeeds in User case #1
 2. ...
- 3) Add Portfolio:
- 4) Add Watch List:
- 5) Add note to stock:
- 6) Add stocks to watch list:
- 7) Add user:
- 8) Perform simulation on a stock:
- 9) Add graph analysis to graph:
- 10) Create simulation model:

Functional Specifications:

The application will provide the users the ability to login to their personal accounts and monitor their portfolios. TradersGem's target will be an Android application using java for the backend programming, XML for the GUI design and the Activity class for the connection between the UI and java interface. This is similar to the swing interface. The project will also include a SQL Lite database which will be used to save user account information such as username and password along with sub categories such linked to both. These sub categories are comprised of the apps functionalities such as the results of simulations, watch lists, portfolio, notes and graphs.

The log in will also facilitate an eventual integration with a web platform, such as a larger and more powerful version of the app running on the web.

Design Phase:

CRCs

The following is the CRCs achieved for the project:

StocksDB		Portfolio	
Contain Stocks	WatchList	Owned Stocks	StockList
	Stock		Stock
	UserAccount		UserAccount
	Portfolio		
TGGraphView		StockMarketStock	
Show Graph Data	Stock	Stock random hitory	StocksDB
	Simulator	random values	Stock
Stock		TempDB	
stock name	Graph	Hold temporary info	Graph
stock financial information	Simulator		Simulator
stock history	Database		
stock simulation results	Note		Note
simulation result fields			
Simulator		UserAccountExcs	
Preset Analysis	Stock	manage exceptions	LoginSystem
Custom analysis	Graph		UserAccount
SimGUI	Database		
	DataRetriever		
AlarmSystem		StockListAdapter	
Warning	Stock	Warning	Stock
	UserAccount		UserAccount
Home		ItemClickMenu	
Holds main Screen	LoginSystem	Hold options	Graph
	UserAccount		AddNote
	Addstock		Sim
	RemoveStock		Stock
	Statistic		
NewUser		RemoveStock	
User Information	UserAccountDB	Remove stock from DB	StocksDB
Password	UserAccount		Stocks
			View

Login System	
Manage Login	UserAccount
	DataBase
WatchList	
current Stocks	StockList
	UserAccount
UserAccountsDB	
User Information	UserAccount
	LoginSystem
Note	
User Strings	Stock
	TempDB
Screener	
	DataRetriever
Filter stocks	Stock
	UserAccount
ListOfStocksController	
List View	Stock

UserAccount	
hold user name	LoginSystem
password	DataBase
database index	StockList
Stock choices	WatchList

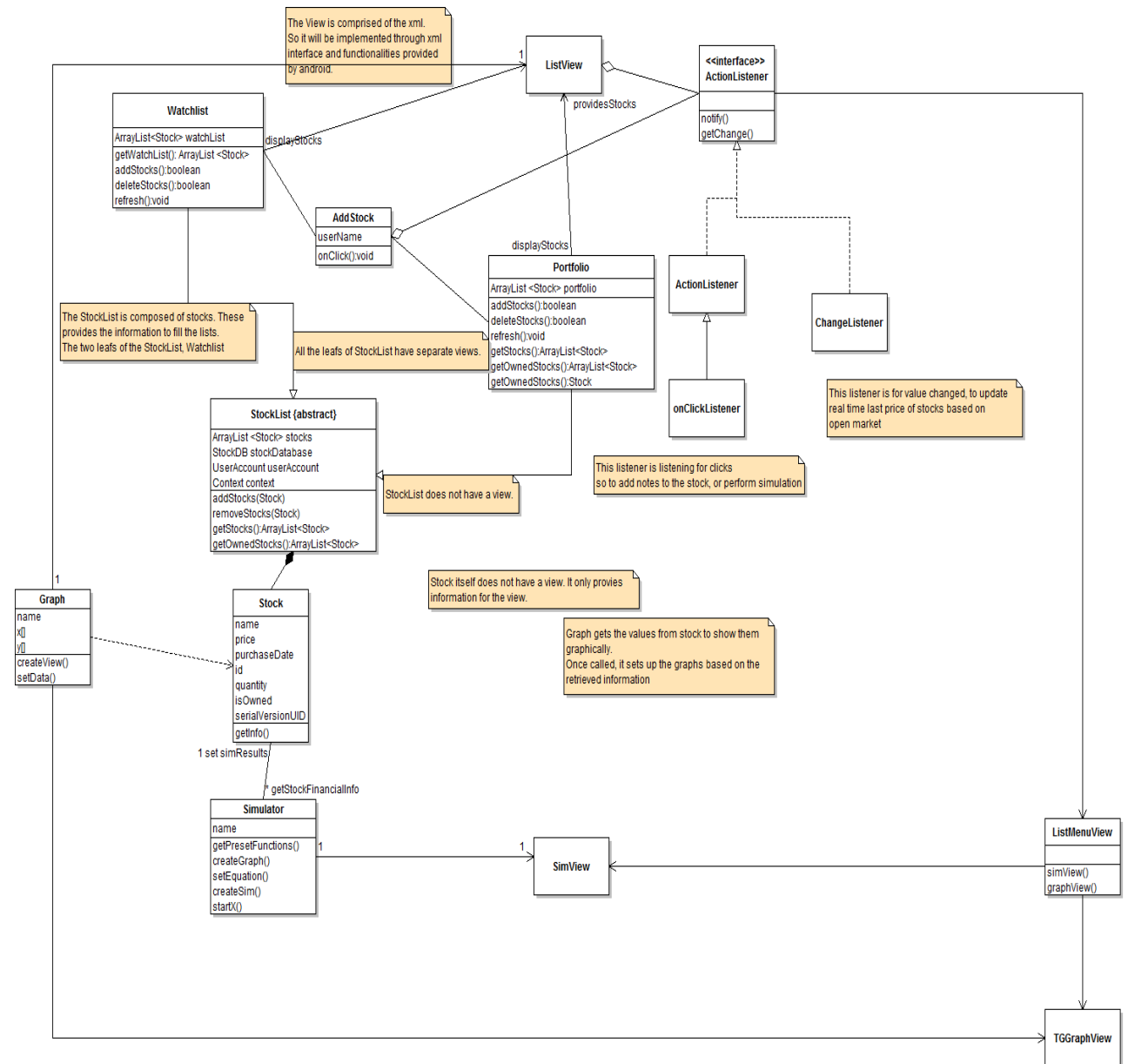
Below is a set of relationship put into tables between the classes and their respective patterns:

UserAccount,LoginSystem, DataBase	Strategy
UserAccount, WatchList,StockList,Stock DataRetriever, Database	Composite
Stock,Simulator,Graph	MVC + Strategy Pattern
AlarmSystem,Note,Stock	Observer pattern

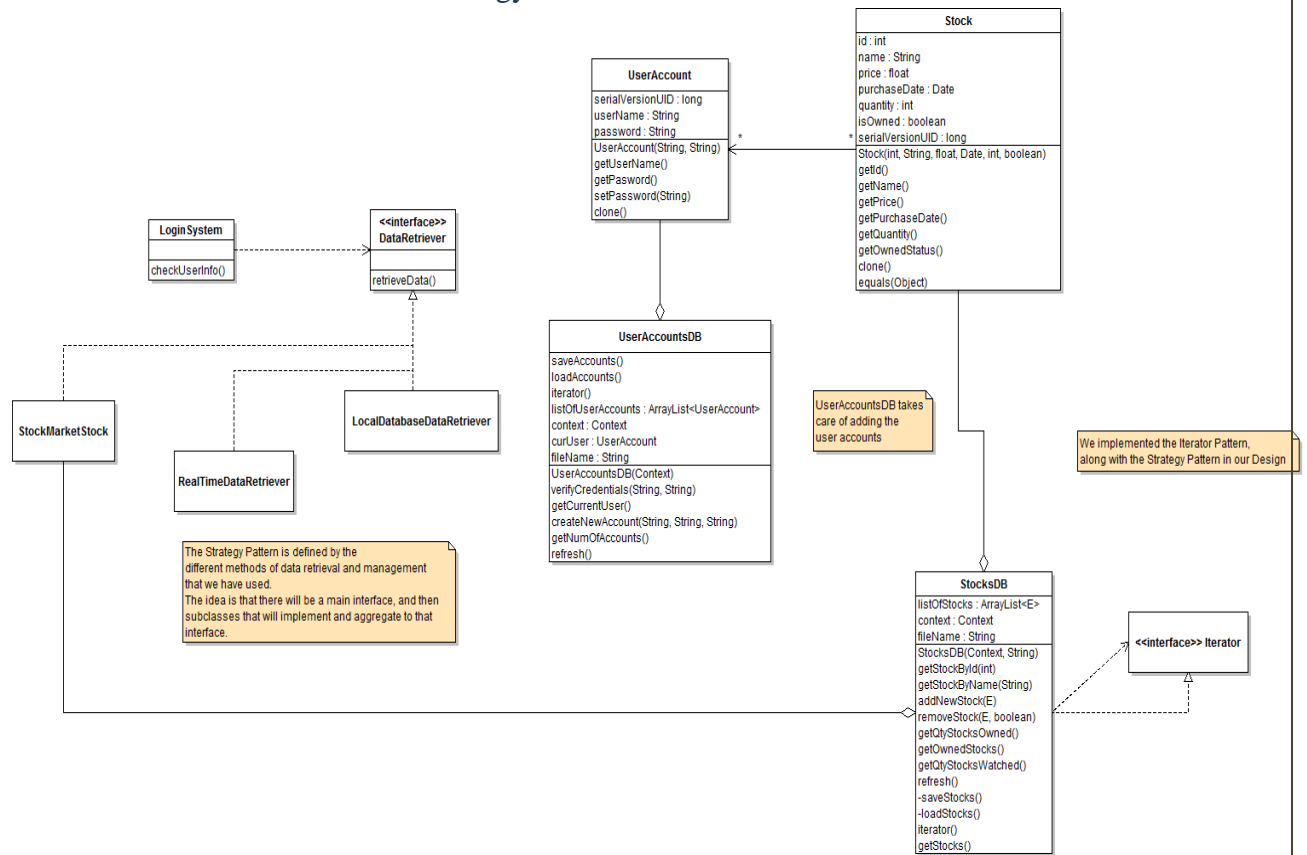
Class Diagrams

The following is the Class Diagrams achieved for the project:

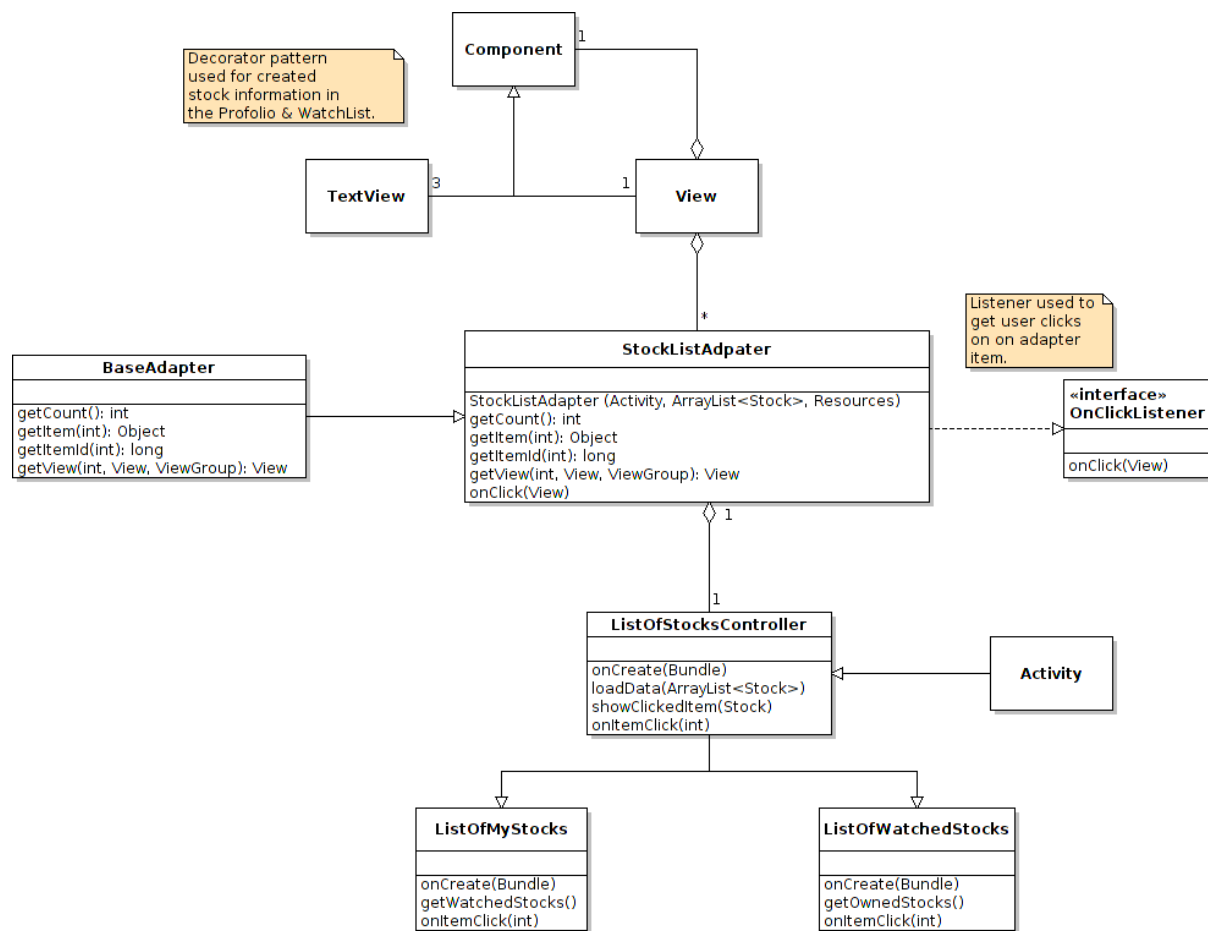
The Class Diagram below implements the interactions between the classes Simulator, Graph and Stock. The graph implements the MVC pattern as well as the Observer Pattern.



The Class Diagram below implements the interactions between the Database, Stock and Account classes involved in the Strategy Pattern and Iterator Pattern.

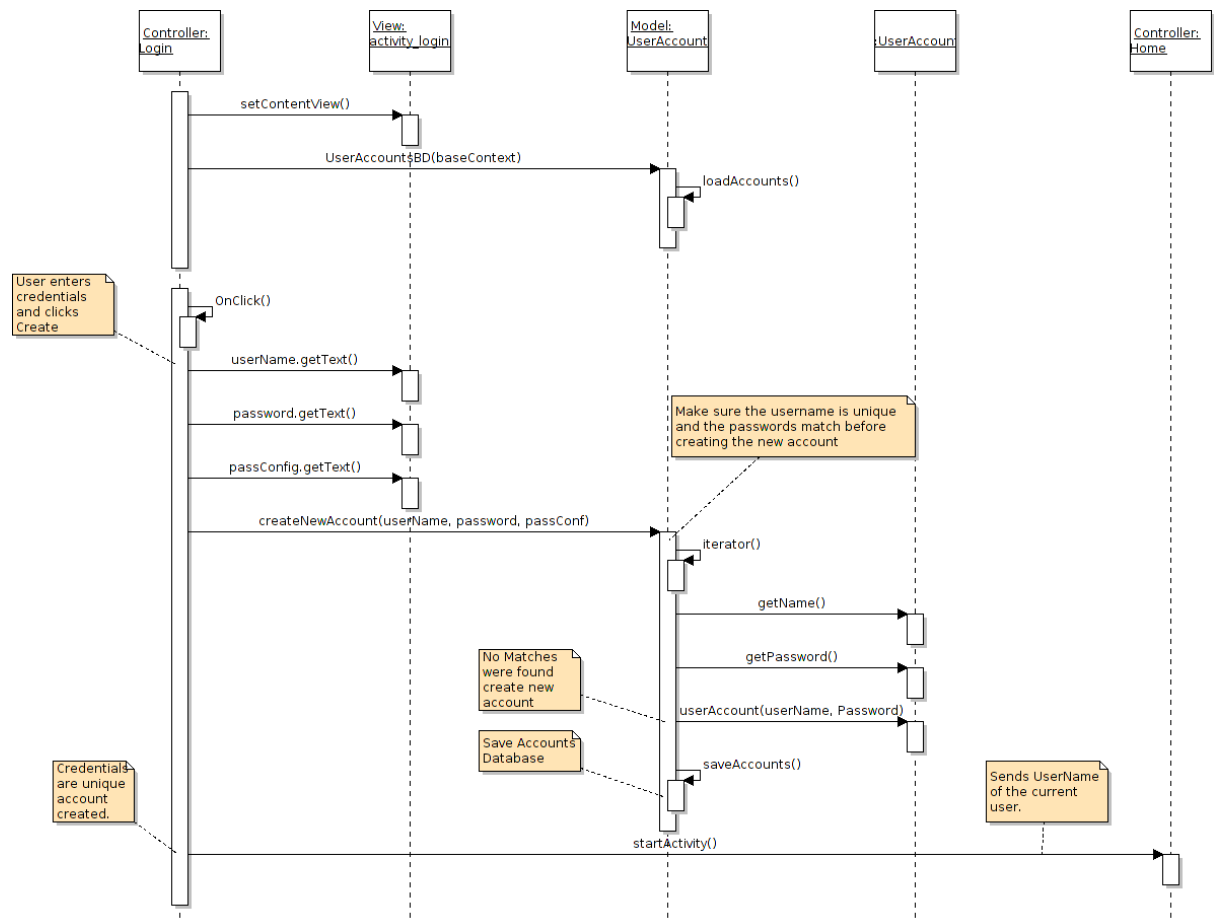


The Class Diagram below implements the interactions between the Watch list and Portfolio classes involved in the Decorator Pattern and Observer Pattern.

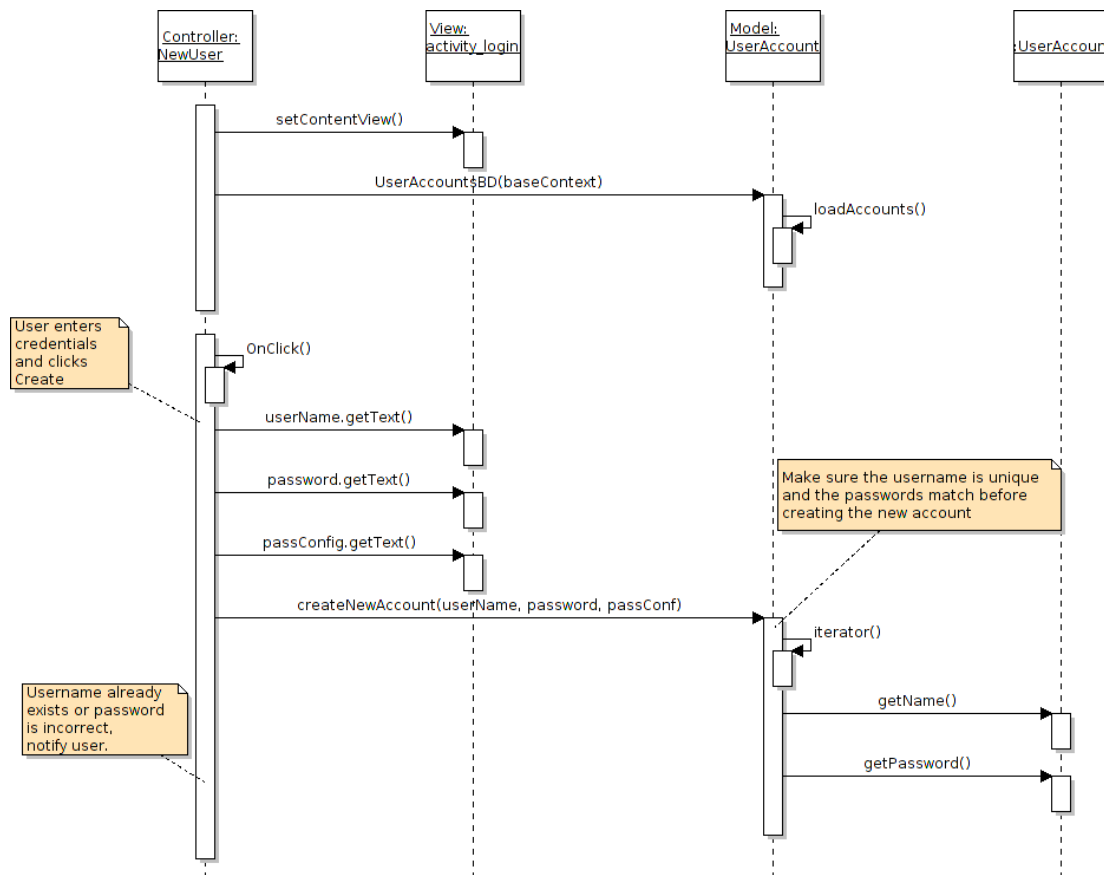


Sequence Diagrams

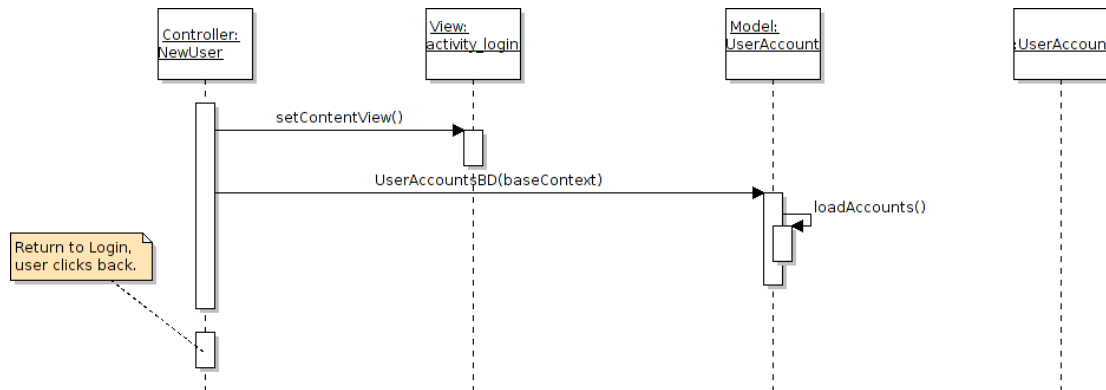
Sequence Diagram: Create New User



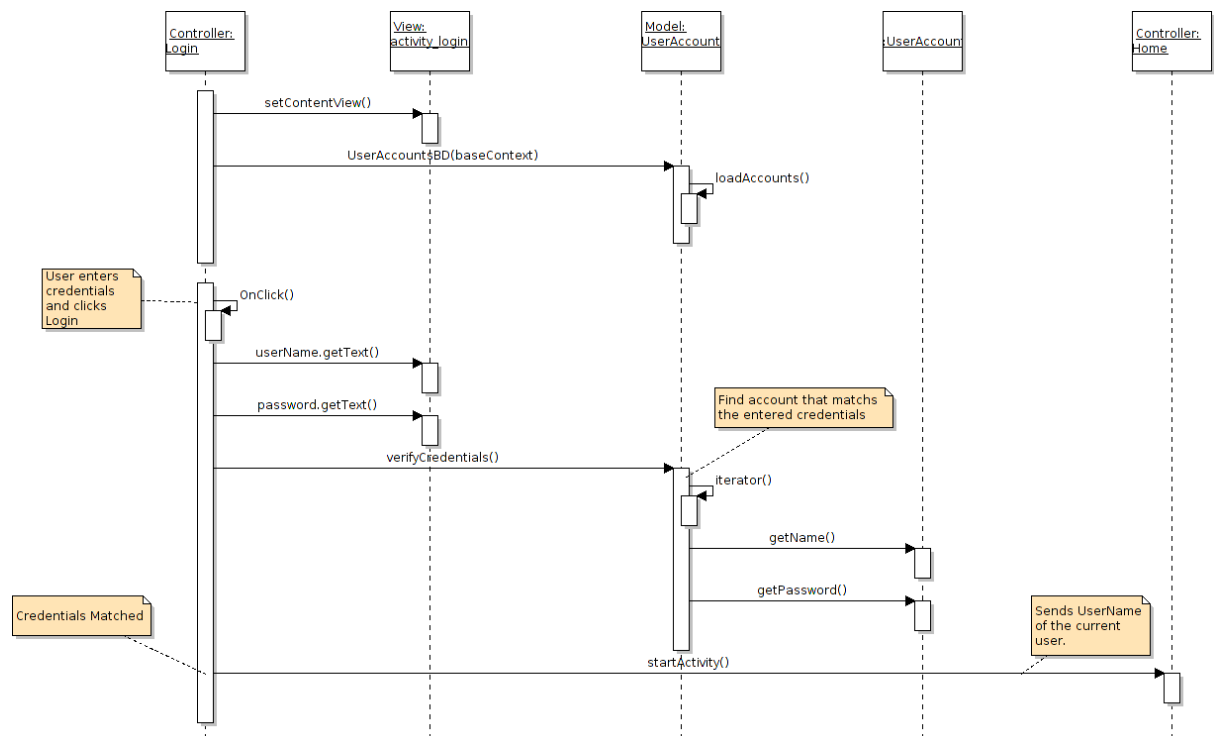
Sequence Diagram: Create New User Variation I: User Already Exists



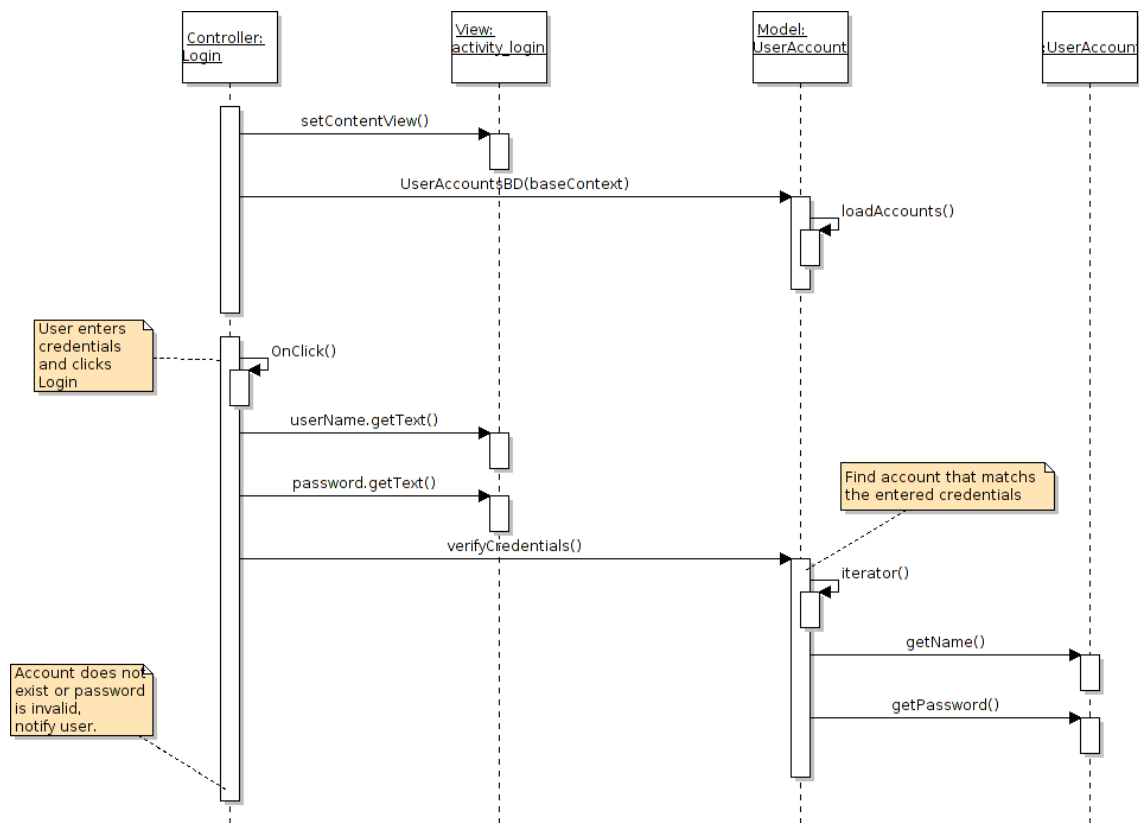
Sequence Diagram: Create New User Variation I: User Already Exists



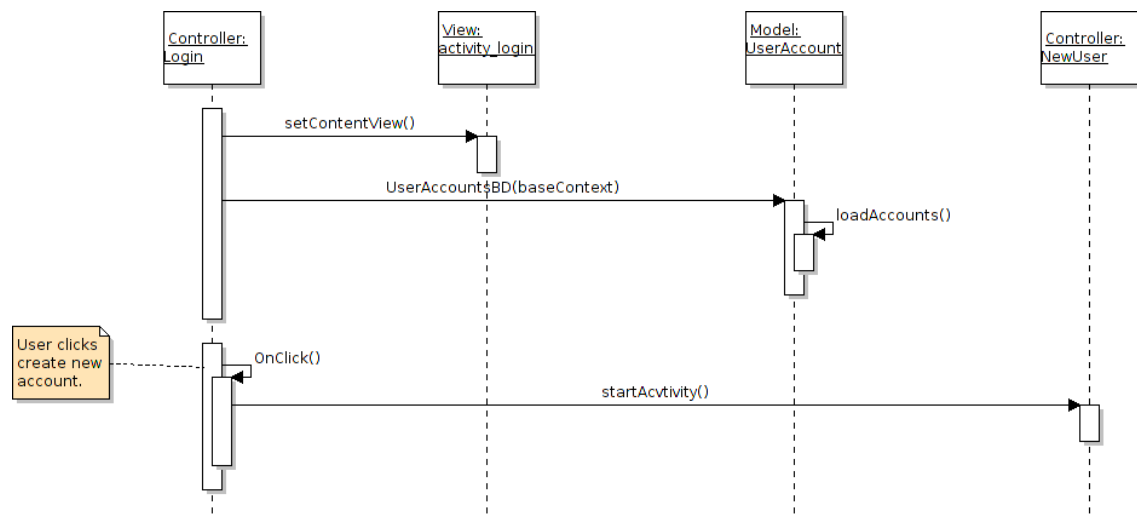
Sequence Diagram: User Login



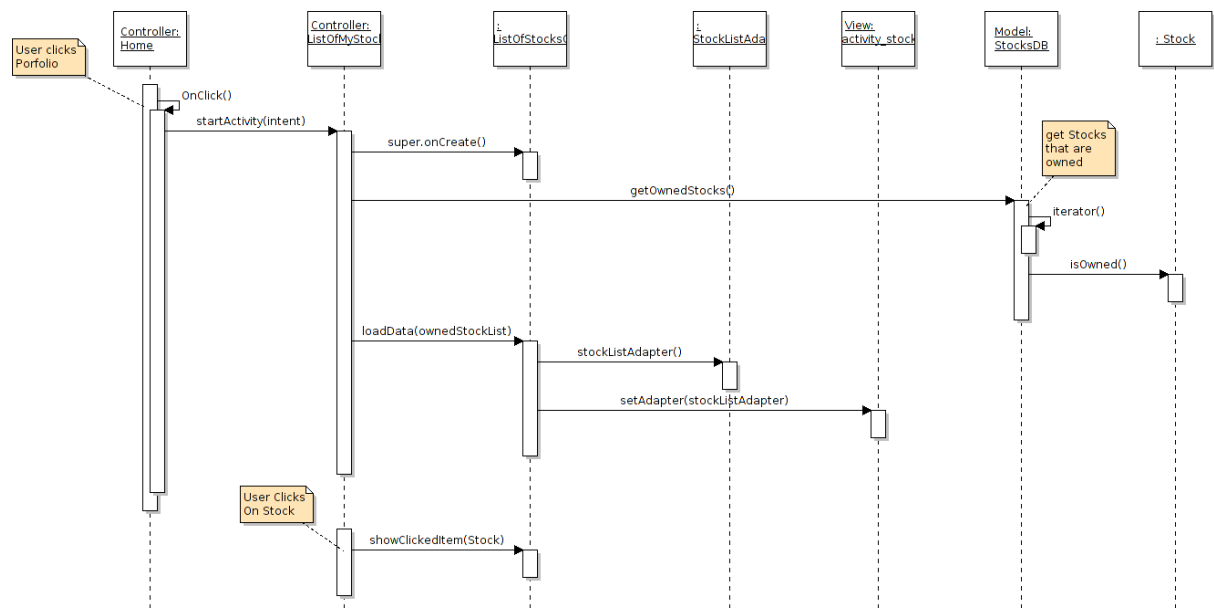
Sequence Diagram: User Login Variation I: Invalid Login



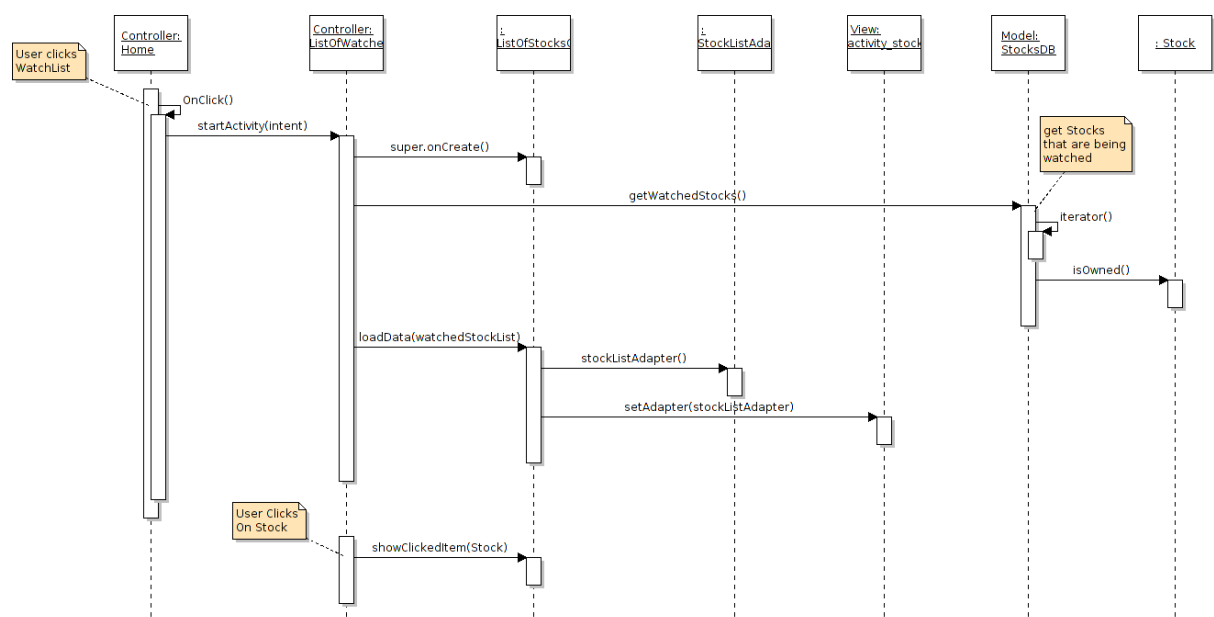
Sequence Diagram: User Login Variation II: New User



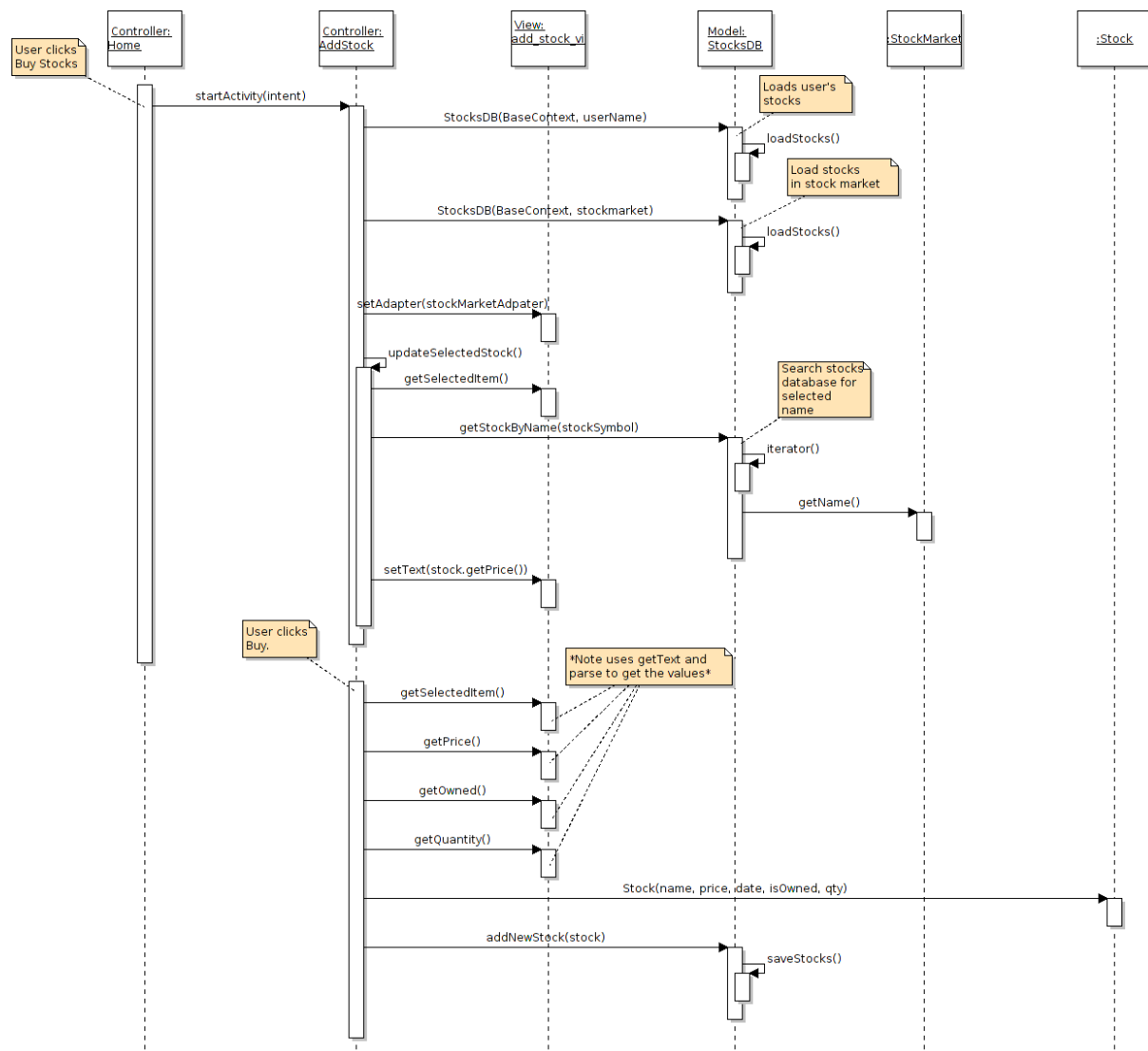
Sequence Diagram: Portfolio



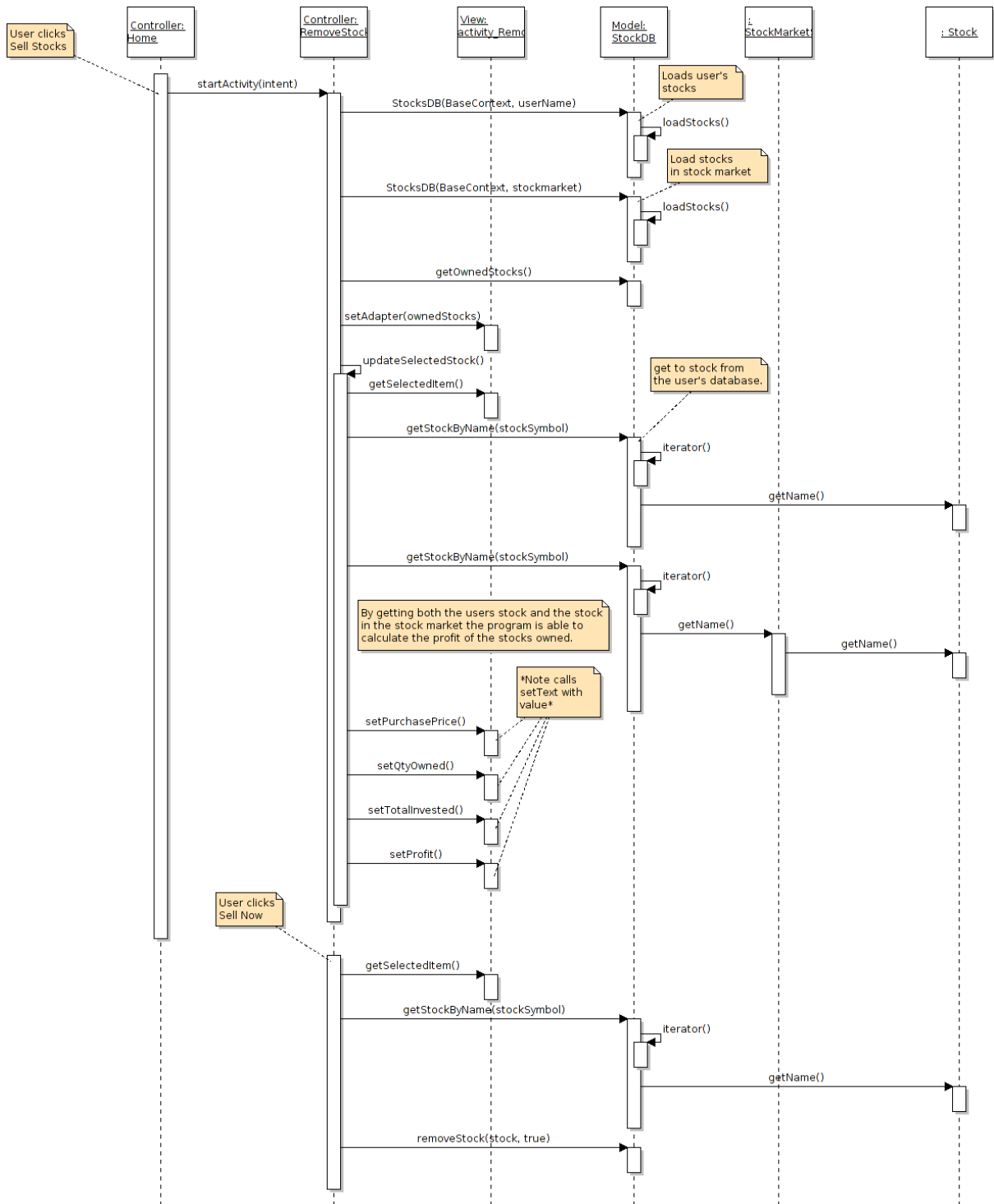
Sequence Diagram: Watched List



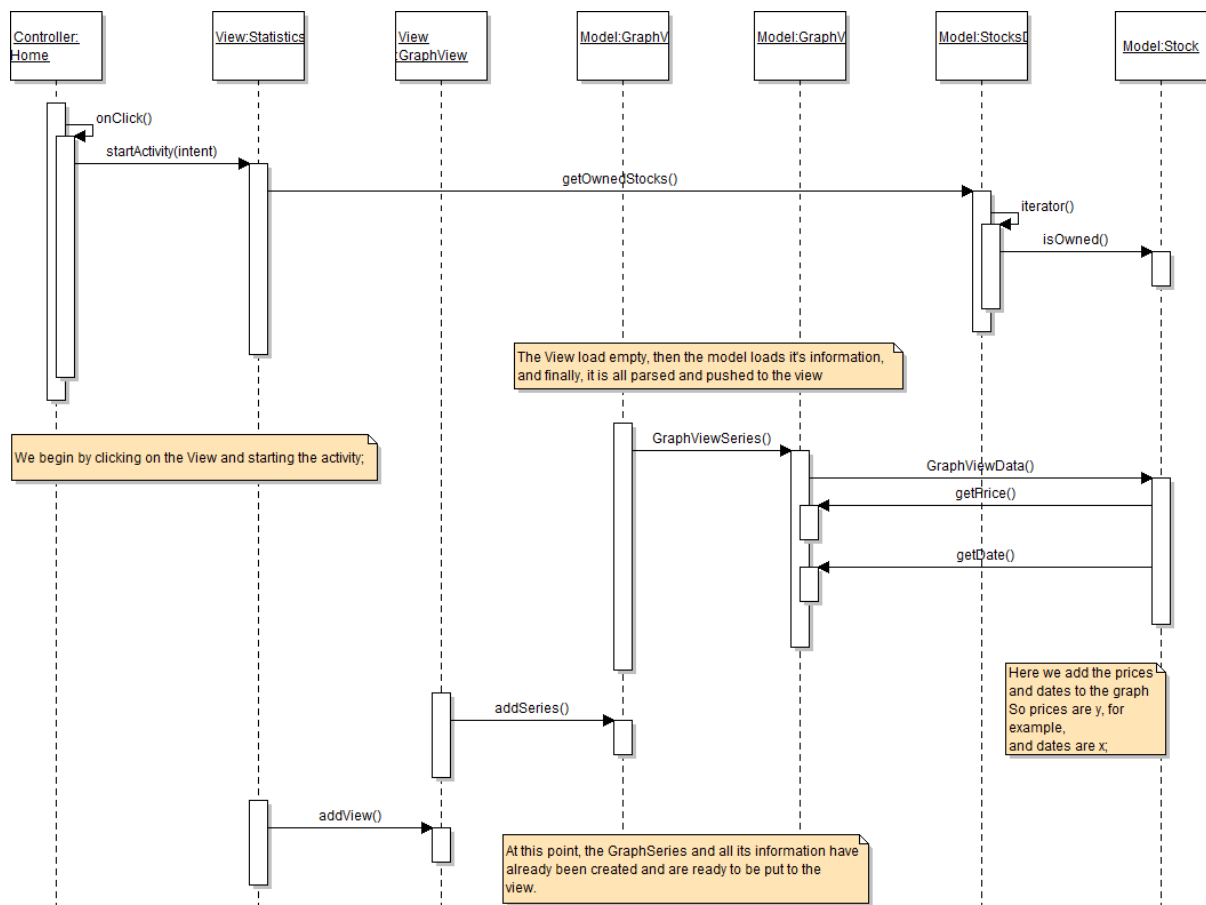
Sequence Diagram: Buy Stocks



Sequence Diagram: Sell Stocks



Sequence Diagram: Statistics



State Diagram

