Application Development

**Ahamed Musthafa Careem**

1020325  (British College of Applied Studies)

[LO1 5](#_Toc106660882)

[LO2 5](#_Toc106660883)

[Tools used to develop the Software 5](#_Toc106660884)

[IntelliJ Idea Ultimate 5](#_Toc106660885)

[Why I used IntelliJ? 5](#_Toc106660886)

[SQLite 7](#_Toc106660887)

[Why I used SQLite? 7](#_Toc106660888)

[DB Browser 7](#_Toc106660889)

[Why I used DB Browser? 7](#_Toc106660890)

[Scene Builder 8](#_Toc106660891)

[Why I used Scene Builder? 8](#_Toc106660892)

[FXML 9](#_Toc106660893)

[Why I choose to go with FXML instead of typical JavaFx or JavaSwing? 9](#_Toc106660894)

[Similar Technologies 9](#_Toc106660895)

[Other IDEs instead of IntelliJ Idea Ultimate 9](#_Toc106660896)

[Pros 10](#_Toc106660897)

[Cons 10](#_Toc106660898)

[Other database instead of SQLite 10](#_Toc106660899)

[Pros 10](#_Toc106660900)

[Cons 11](#_Toc106660901)

[Other GUI builders instead of SceneBuilder 11](#_Toc106660902)

[Software Methodologies 11](#_Toc106660903)

[Agile Software Method 11](#_Toc106660904)

[Advantages of Agile 11](#_Toc106660905)

[Disadvantages of Agile 11](#_Toc106660906)

[Waterfall Model 11](#_Toc106660907)

[Advantages of Waterfall Model 12](#_Toc106660908)

[Disadvantages of Waterfall Model 12](#_Toc106660909)

[Prototype Method 12](#_Toc106660910)

[Advantages of Prototype Method 12](#_Toc106660911)

[Disadvantages of Prototype Method 12](#_Toc106660912)

[Spiral Model 12](#_Toc106660913)

[Advantages of Spiral Model 12](#_Toc106660914)

[Disadvantages of Spiral Model 12](#_Toc106660915)

[Method I choose 13](#_Toc106660916)

[V Model 13](#_Toc106660917)

[Advantages of Scrum Development Method 13](#_Toc106660918)

[Disadvantages of Scrum Development Method 13](#_Toc106660919)

[Why I choose V Model? 13](#_Toc106660920)

[LO3 14](#_Toc106660921)

[Pitch Template 14](#_Toc106660922)

[Developing the Software 14](#_Toc106660923)

[Code Snippets 17](#_Toc106660924)

[Userpackage 17](#_Toc106660925)

[Farmer 17](#_Toc106660926)

[Miller 17](#_Toc106660927)

[Farming Agency 18](#_Toc106660928)

[Agriculture Agency 18](#_Toc106660929)

[User 19](#_Toc106660930)

[Auction 21](#_Toc106660931)

[Bid 23](#_Toc106660932)

[Buyer 28](#_Toc106660933)

[DatabaseConnection 28](#_Toc106660934)

[Main 41](#_Toc106660935)

[Product 42](#_Toc106660936)

[Seller 45](#_Toc106660937)

[Resources Package 46](#_Toc106660938)

[auctionMenu 46](#_Toc106660939)

[login 46](#_Toc106660940)

[registration 47](#_Toc106660941)

[sessionSelection 48](#_Toc106660942)

[Controllers Package 49](#_Toc106660943)

[AuctionMenuController 49](#_Toc106660944)

[LoginController 50](#_Toc106660945)

[RegistrationController 50](#_Toc106660946)

[SessionSelectionController 50](#_Toc106660947)

[Image Files included 50](#_Toc106660948)

[Admin 53](#_Toc106660949)

[Logging in 53](#_Toc106660950)

[Register Users 54](#_Toc106660951)

[Edit a User 58](#_Toc106660952)

[Load user details 59](#_Toc106660953)

[Deleting a user 59](#_Toc106660954)

[Create auction session 60](#_Toc106660955)

[Delete an auction session 60](#_Toc106660956)

[Logging out 61](#_Toc106660957)

[Farmer 61](#_Toc106660958)

[Logging in to the system 61](#_Toc106660959)

[Entering the auction session 62](#_Toc106660960)

[Selling a product 63](#_Toc106660961)

[Logging out 65](#_Toc106660962)

[Miller 65](#_Toc106660963)

[Logging into the system 65](#_Toc106660964)

[Entering the auction session 65](#_Toc106660965)

[Selling a Product 66](#_Toc106660966)

[Placing a bid 67](#_Toc106660967)

[Logging out 67](#_Toc106660968)

[Small Farming Coperate Agency 67](#_Toc106660969)

[Logging in 67](#_Toc106660970)

[Entering the auction session 68](#_Toc106660971)

[Selling a product 69](#_Toc106660972)

[Placing a bid 69](#_Toc106660973)

[Logging out 73](#_Toc106660974)

[Agriculture Coperative Agency 74](#_Toc106660975)

[Logging in 74](#_Toc106660976)

[Entering the auction session 74](#_Toc106660977)

[Selling a product 75](#_Toc106660978)

[Placing a bid 76](#_Toc106660979)

[Logging out 77](#_Toc106660980)

[LO4 78](#_Toc106660981)

[Test Case 78](#_Toc106660982)

[Deadline Schedule 78](#_Toc106660983)

[Quality Assurance 79](#_Toc106660984)

[SDLC Activities 79](#_Toc106660985)

[Changes Made in the System 79](#_Toc106660986)

[Reviewing Software Tools 79](#_Toc106660987)

[Project Planning and Overseeing 80](#_Toc106660988)

[Product Assessments 80](#_Toc106660989)

[Process Assessments 80](#_Toc106660990)

[Performance of the application against the Problem Definition Statement and initial requirements. 80](#_Toc106660991)

[Problem Statement 80](#_Toc106660992)

[Problem 80](#_Toc106660993)

[Current Performance Issues 80](#_Toc106660994)

[Target Performance 81](#_Toc106660995)

[How to resolve the problem 81](#_Toc106660996)

[Requirements 81](#_Toc106660997)

[Performance of the Application 81](#_Toc106660998)

[Assumptions when building the system 82](#_Toc106660999)

[Functional Requirements 82](#_Toc106661000)

[Admin 82](#_Toc106661001)

[Farmers 82](#_Toc106661002)

[Millers 82](#_Toc106661003)

[Small Farming Agency 82](#_Toc106661004)

[Agriculture Agency 83](#_Toc106661005)

[Strengths and Weakness of my Application 83](#_Toc106661006)

[Strengths 83](#_Toc106661007)

[Weakness 83](#_Toc106661008)

[How to improve my Software 83](#_Toc106661009)

[References 84](#_Toc106661010)

# LO1

Attached separately as **“LO1 Agriculture Auction SDD”**

# LO2

## Tools used to develop the Software

### IntelliJ Idea Ultimate

I used IntelliJ Idea Ultimate as my IDE to develop my code which was mainly written on Java, Javafx/swing and SQL. There are many other IDEs which can be used. But I used IntelliJ because

#### Why I used IntelliJ?

* Intellisense

IntelliJ IDEA understands about my project and suggests various functionalities and auto completes on various situations which makes coding faster and easier

* Scene Builder

Ability to run fxml files seamlessly due to its support for scenebuilder

* Latest User Interface

Unlike typical IDEs such as Netbeans and Eclipse, IntelliJ has a very modern clean look with dark mode support.

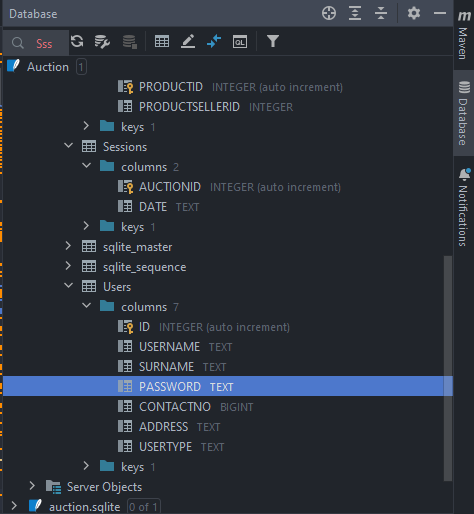
* Extensive plugins support

I use the following plougins in Intellij

* + WakaTime:- To track the coding time and update my self
  + Rainbow Brackets:- Makes easier to identify which brackets are opened and closes
  + Scala:- Coding Assitance highlighting and refractoring
  + SonarLint:- Fixes bugs and finds vulnerabilities
* Ability to run apps from inbuilt

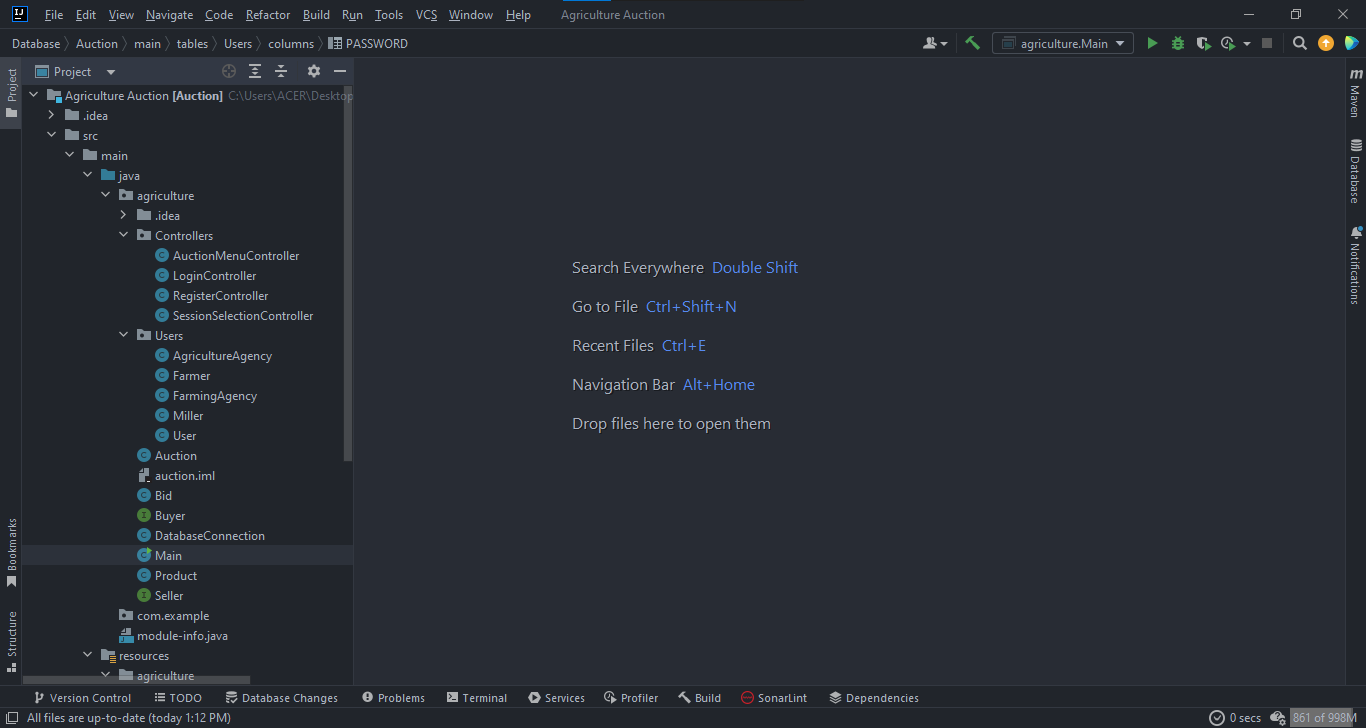
You can run your apps easily and quickly compared to other IDEs

* Integrated Database

Since I am using the Ultimate version, you can view the database there it self which makes easy to work on it. A sample picture is shown below 

* Others

Tons of awesome features are inbuilt which overall increases coding speed and efficiency. It also consists a distraction free focus mode and many other extra features.



### SQLite

I have used SQLite for my database management system (RDBMS). It supports all the major types of Operating systems and there are precompiled binaries.

#### Why I used SQLite?

* Fast and better performance

The read and write speed from SQLite database is extremely faster than others. It only loads the data which is needed, rather than reading the entire file and hold it in memory.

* Lightweight

This database is very lightweight so when it comes to creating my system this will be advantages if you have a low end pc

* Supportability

It can be integrated with many third party tools.

The one which I use is called “DB Browser”.

### DB Browser

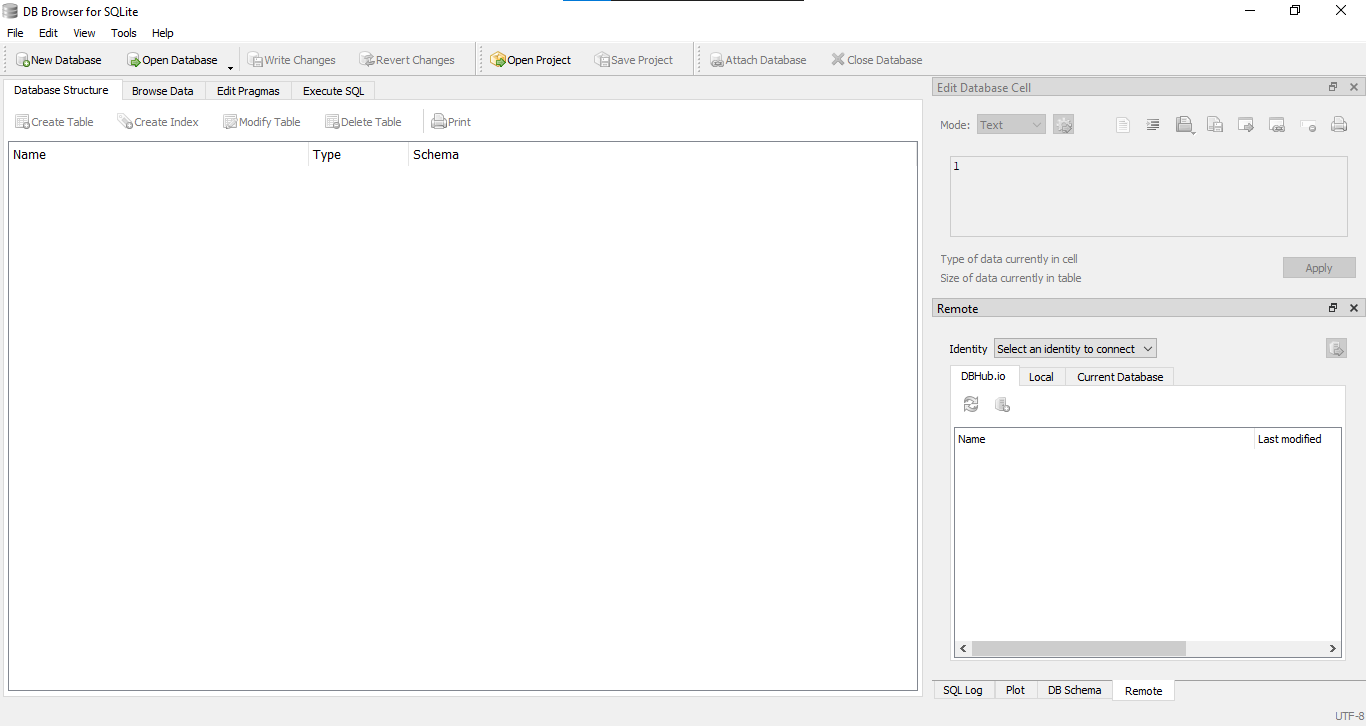
DB Browser is an open source tool made for creating, editing and designing SQLite database files. It uses a general spreadsheet-like interface which makes everything simple.

#### Why I used DB Browser?

This is a very light weight software

Can be used by the End user and the developer because everything is simple  
I can simply do functions such as

* Create database files
* Create, define, edit, delete tables
* Create and delete indexes
* Import and export record as plain text
* Ability to get or export data from CSV type files
* Easily add SQL queries



### Scene Builder

To build the layout of the application with GUI, JavaFX provides a design tool called the JavaFX Scene Builder. You just need to drag and drop UI components to a JavaFX Content pane, and the tool generates the FXML code that can be used in an IDE such as IntelliJ, NetBeans etc.

#### Why I used Scene Builder?

* User friendly interface

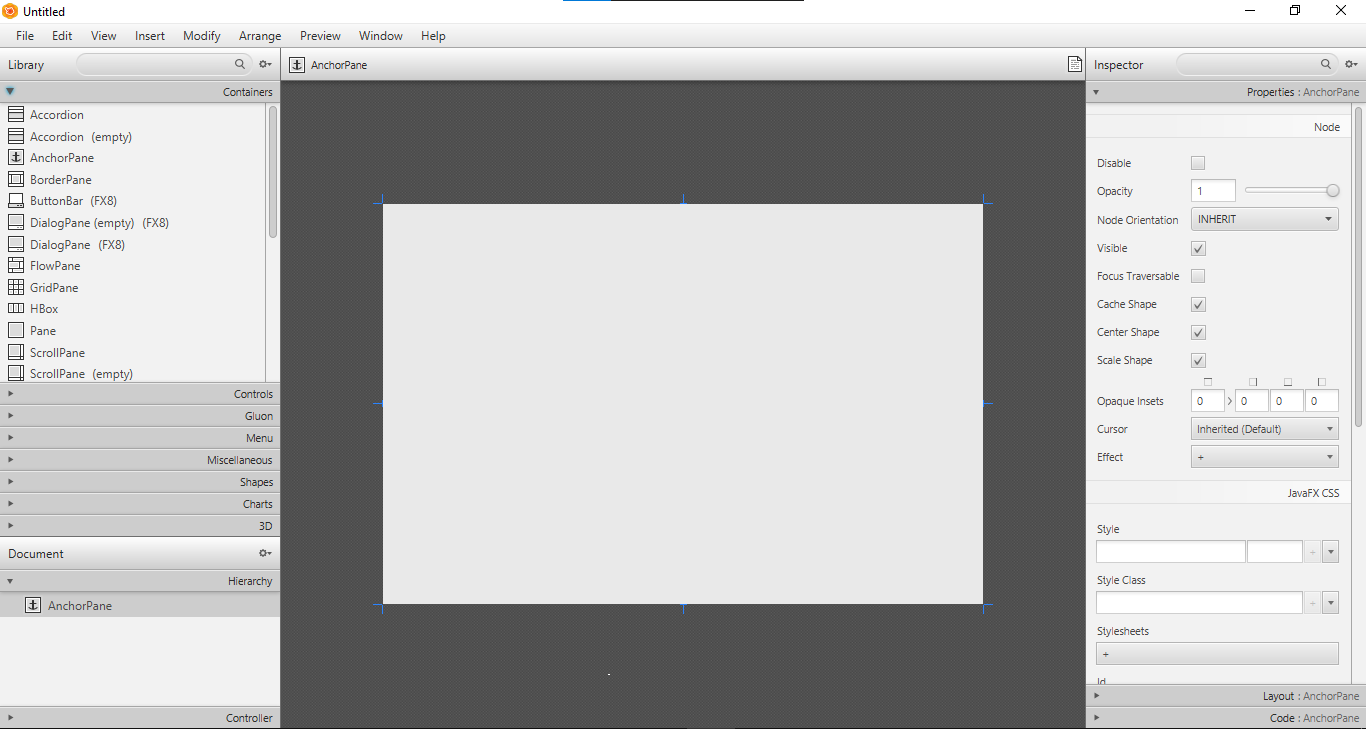
The easy UI helped me to access and understand the functions and create an interface effortlessly

* Preview Option

The preview option helped me to look and choose the colors so I can see how will my interface look like when I run it

* Configure with any IDE

Most IDEs supports scene builder. It was easy to do in IntelliJ and I was able to make changes directly from my IDE .



### FXML

FXML is an XML-based markup language which allows developers to create a user interface in a JavaFX application separately from implementing the application logic.

#### Why I choose to go with FXML instead of typical JavaFx or JavaSwing?

* Easy to add features and reconfigure the UI
* Shows a clear separation of the GUI from the GUI and the controller which makes easier to update in the future
* Easily able to integrate with Scene Builder
* It can be edited real time and the changes in the GUI will show there itself
* Can use with any JVM (Java Virtual Machine)
* The approach and the codings looks more professional and standardized

## Similar Technologies

### Other IDEs instead of IntelliJ Idea Ultimate

There are many IDEs suitable for Java such as Eclipse,Oracle ,Jdeveloper,Atom and Apache Netbeans.

Let us discuss more on Apache Netbeans since it is one of the best competitors.

Apache NetBeans is an IDE which is a development environment, application framework and a tooling platform. Both supports Java development.

#### Pros

IntelliJ is better than Netbeans because they provide more fucntions and tools

IntelliJ IDEA has better Themes than Netbeans which make pleasant for the eyes and ability to work easier

IntelliJ IDEA has a very large plugin library and able to directly download where in Netbeans you can only download a few and import from marketplace separately

Netbeans is not memory efficient like IntelliJ IDEA

#### Cons

Eventhough IntelliJ also has a free community version, netbeans is completely free

IntelliJ IDEA is mostly Java focused but Netbeans can be equally used for other languages

### Other database instead of SQLite

There are many alternatives for our database management system.  
Most popular ones are MySQL, MongoDB, MariaDB, Microsoft SQLServer etc.

Let us compare with MySQL

MySQL Community Edition is an open source database and is one of the most popular DBMS today. Unlike MySQL, SQLite is serverless and self contained.

Data Type Support

SQLite:-

Blob Integer Null Text Real

MySQL:-

Tinyint Smallint Mediumint Int Bigint Double Float

Real Decimal Double precision Numeric Timestamp Date

Datetime Char

(Duggal, 2021)

#### Pros

* SQLite is lighter than MySQL
* SQLite is easy and understandable than MySQL
* SQLite is easy to setup and configure than MySQL

#### Cons

* MySQL has a password configuration hence making it more secure than SQLite
* MySQL supports multiple users unlike SQLite

### Other GUI builders instead of SceneBuilder

The top alternatives for scene bulder are JavaFX, SWT from Eclipse, CustomStage, Gluon and etc.

Let us compare it with JavaFx and see why I choose SceneBuilder instead of developing it on SceneBuilder directly.

* JavaFx can be found in a typical IDE itself but SceneBuilder is a separate software and also it supports few IDEs like IntelliJ IDEA.
* Scene builder creates .fxml files unlike JavaFx
* Scenebuilder has more features and functions than JavaFx

## Software Methodologies

SDLC known as Software Development Life Cycle models are various design, development and testing processes followed by industry and professionals.

Choosing the correct type of software development methodology is really important and it depends on goals, software type, team etc. Various software development methodologies has been introduced in the past let us go through a few.

1. Agile Model
2. Waterfall Model
3. Prototype Model
4. Spiral Model
5. V Model

### Agile Software Method

Agile method is a combination of an incremental and iterative approach. Main principle is if you fail early you can fix minor issues before they grow into large issues.

#### Advantages of Agile

* Allows software to release on iteration
* Ability to find and fix bugs in the early stage
* Reduces time due to less documentation

#### Disadvantages of Agile

* Relies on real time communication
* Clients and end users are unsure about requirements
* Huge time commitment since each feature should be completed fully to get the approval

### Waterfall Model

In waterfall, the project team completes one phase at a time and uses the previous information to move forward step by step.

#### Advantages of Waterfall Model

* Linear nature makes easier to understand
* Good for small scale projects
* Clear Milestones are defined

#### Disadvantages of Waterfall Model

* After requirements are gathered there is no formal way to change it
* It can be slow and costly

### Prototype Method

Prototype method is a sdlc process where a prototype is created before developing an actual project

It is created according to the customer requirement and to get the feedback from them. The process is carried on until they are satisfied.

#### Advantages of Prototype Method

* Any missing feature can be easily taken care of
* Able to make sure the system is perfect
* Easy to understand requirements and less confusions since the customer is involved

#### Disadvantages of Prototype Method

* More time might get consumed if the requirements keep changing
* Customer can change the requirement each an every time and make the code complex

### Spiral Model

In Spiral Model the software passes through 4 phases until the project is finished which is planning, risk analysis, development and evaluation. This is very suitable for developers who are building a highly customizable project. In Spiral we predict potential risks and then the best way to avoid or minimize them.

#### Advantages of Spiral Model

* Ability to make changes in the later stage of development
* Good risk management
* Ability to get customer feedback

#### Disadvantages of Spiral Model

* Only practical for large projects
* Risk of missing the deadlines and not meeting budgets due to high time and labor cost
* More documentation required

(Singh, 2022)

(team, n.d.)

(michaelpage, n.d.)

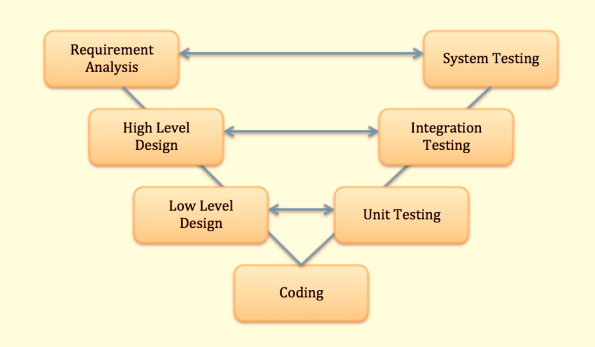
(softwaretestinghelp, 2015)

## Method I choose

### V Model

In V model the testing is happened at the end of the project and happens on every stage of the development and starts the next stage when the previous stage is completed similar to the waterfall model. The software tester has to verify whether the requirements of the development are met or not.

For every phase in this model, there is a corresponding phase. Look at the image below



Picture Credits:- (guru99, n.d.)

The left side of the model is Software Development Life Cycle – SDLC

The right side of the model is Software Test Life Cycle – STLC

#### Advantages of Scrum Development Method

* Higher chances of success
* Avoid bugs and issues
* Fits for smaller projects
* Saves lot of time since planning and designing is done for testing before coding takes place

#### Disadvantages of Scrum Development Method

* Requirements and test documents need to update if changes are made during development
* Cannot create a prototype early

### Why I choose V Model?

* Simple and easy to follow
* Prevent bugs and issues carrying out in the next stages
* This is suitable for me since the requirements are defined clearly
* The agriculture agency is confident on my skills so no need to produce any prototypes
* Since the planning and test designing are done prior the success rate is high
* Able to track process easily

# LO3

## Pitch Template

Has attached separately as **“AD Software Pitch Presentation”**

## Developing the Software

To develop the software I have used

* IntelliJ IDEA Ultimate as my IDE
* DB Browser to store my database
* Scene Builder to create my interface

I have used Java , Javafx and SQLite to develop my software.

The system has been divided into 3 packages named Controllers, Users and Resources.

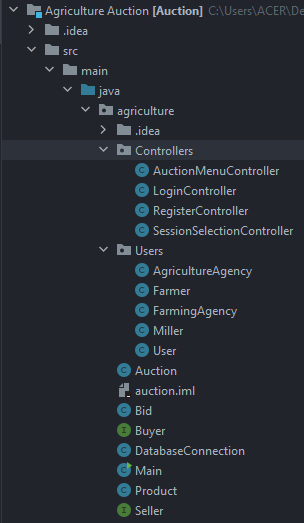
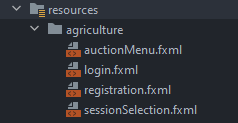
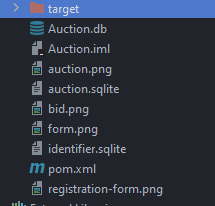
The Users package consists all the classes in our system. They are:-

Agriculture Agency, Farmer, Farming Agency, Miller, User, Auction, Bid, Buyer, DatabaseConnection, Main, Product and Seller.

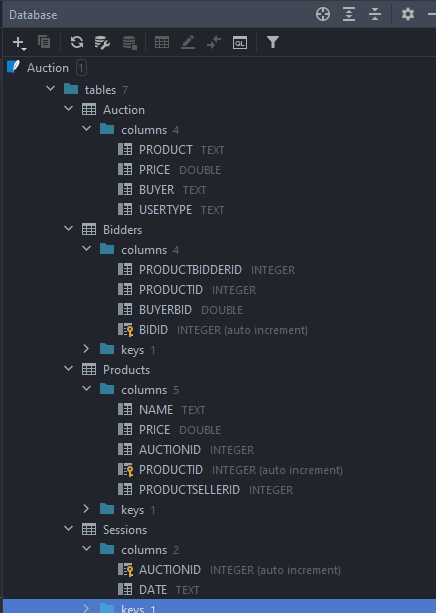
The resources package consists all of my fxml files which is the JavaFx files.

The Controllers package consists all the files which are linked to the corresponding fxml file and act on its behavior.

The Auction database and the necessary images has been imported to the system.



These are my database tables



Scenario and Key points to consider:-

* Functions:-
* Farmers can sell
* Millers can buy from Farmers and sell to Farming Agency
* Farming Agency can buy from Millers and sell to Agriculture Agency
* An Agricultural Agency can purchase the items from a farmer.
* Since Users are not much educated and unaware about the technology the admin has the privilege to Register new users, edit and delete them.
* If the User needs to change their password they need to follow the procedure which will be discussed in this report.

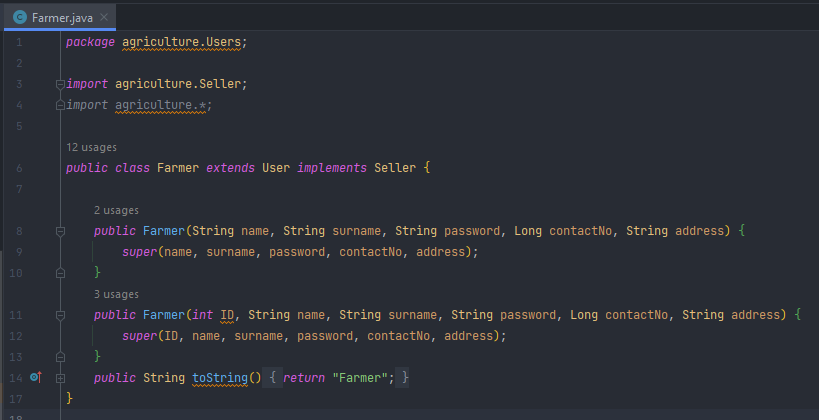
## Code Snippets

Let us go through few code blocks in each pieces

### Userpackage

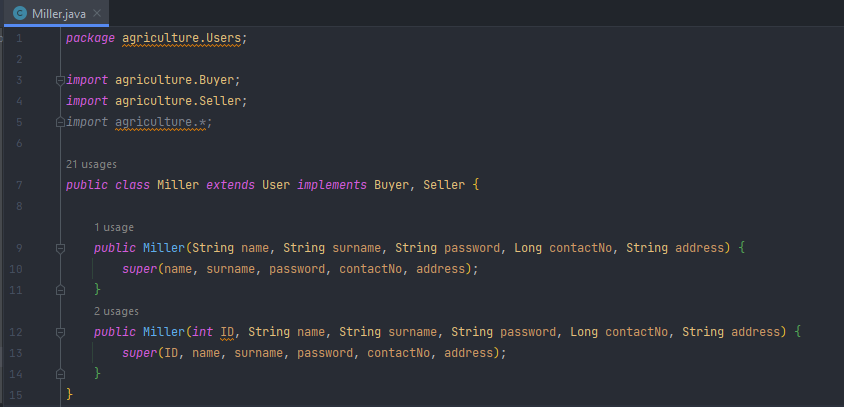
#### Farmer

We have overridden this method in Users. We also have called the field names in the Farmer table in our database. We also store the Farmers data in our database



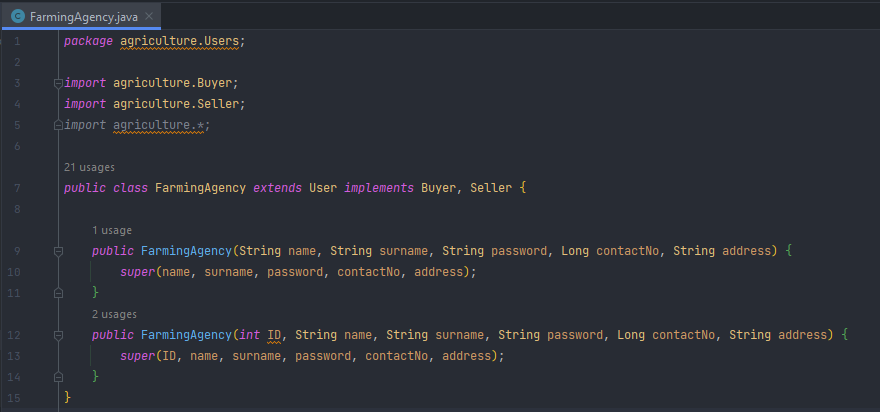
#### Miller

The Miller user class is a sub class of Buyer and Seller class. All the relevant database tables and columns are called in here.



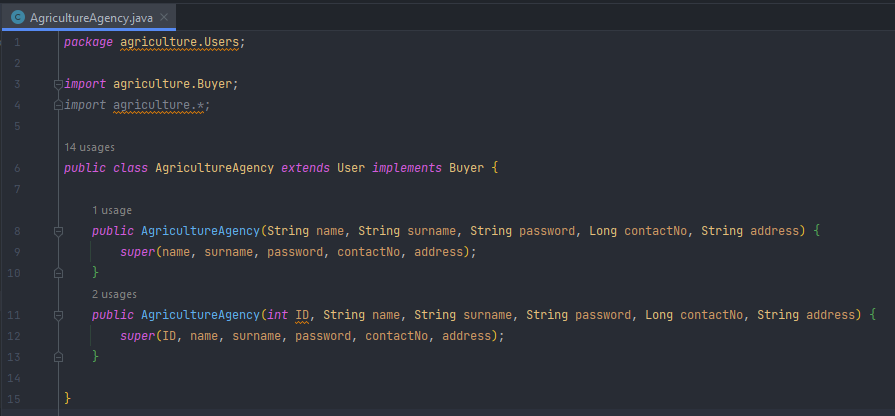
#### Farming Agency

The Farming Agency user class is a sub class of Buyer and Seller class. All the relevant database tables and columns are called in here.



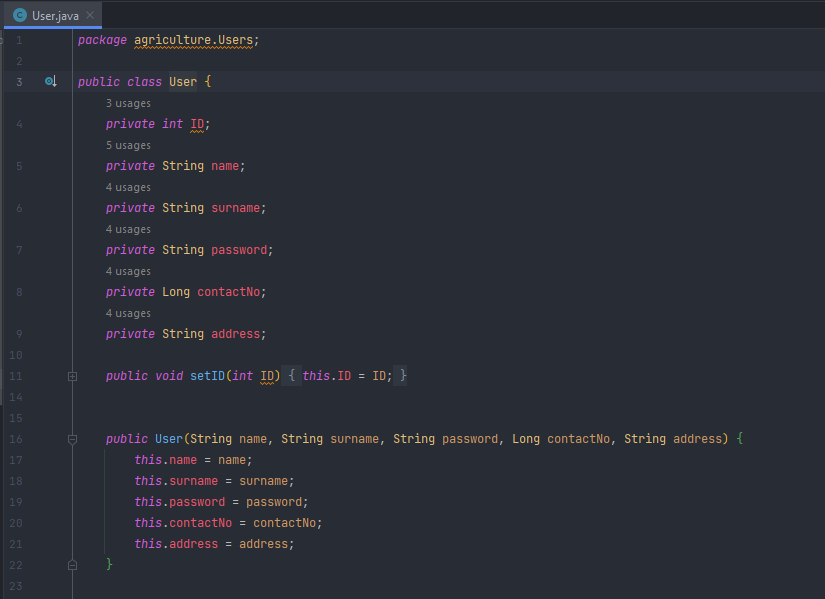
#### Agriculture Agency

The Agriculture Agency user class is a sub class of Buyer class. All the relevant database tables and columns are called in here.

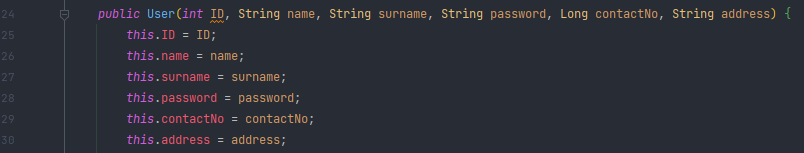


#### User

As you saw before Farmer, Miller, Farming Agency and AgricultureAgency are sub classes of the User class. Hence we can call User as a parent class.



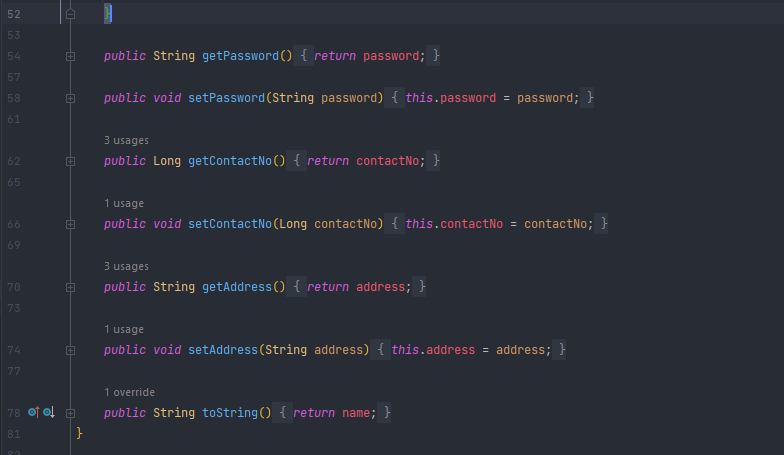
The user class displays the data from the database



As you can see the public getter and setter class retrieves and updates the value of a variable outside the encapsulating class

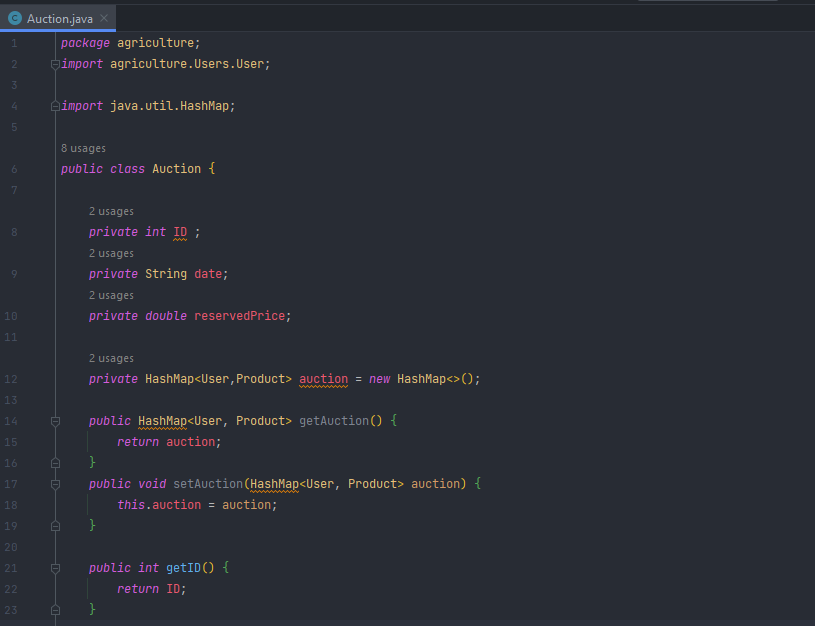


In line 78 we can see this overrides the method in user and in the object which we saw in the 14th line in Farmer class.

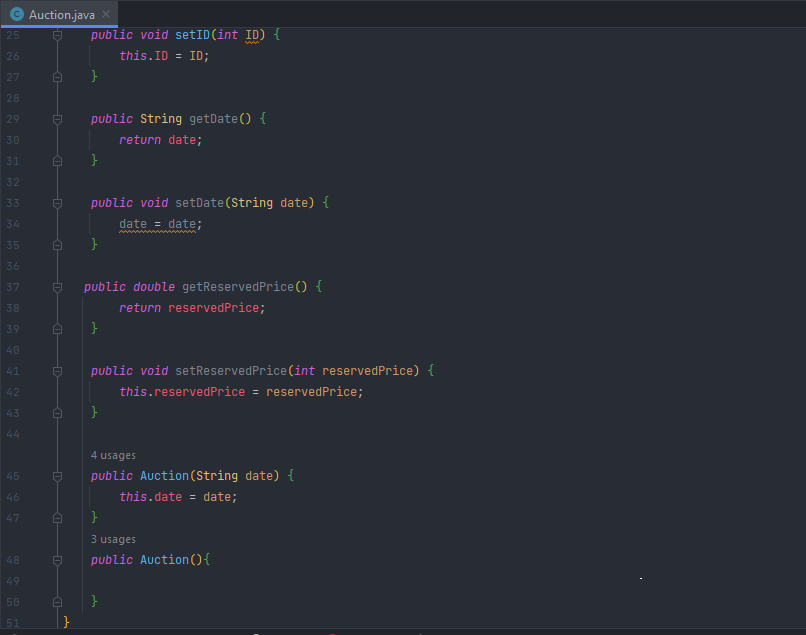


#### Auction

Auction class handles the data which is stored in the Auction session. Java Hashmap stores elements in key/value pairs. Where, keys are unique identifiers used to associate each value on a map.

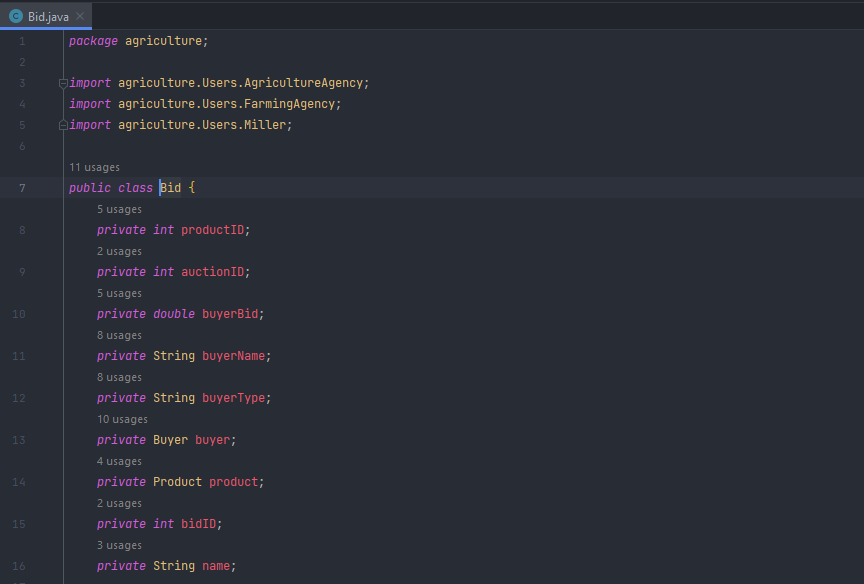


I have used various functions to declare certain commands.

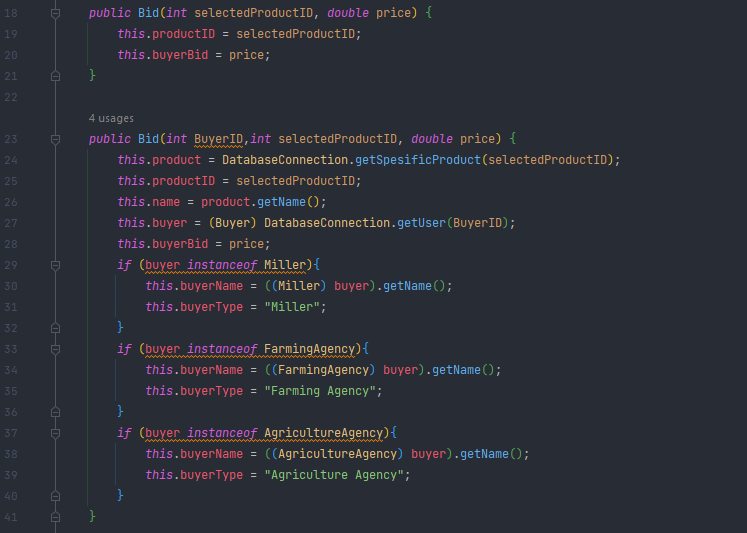


#### Bid

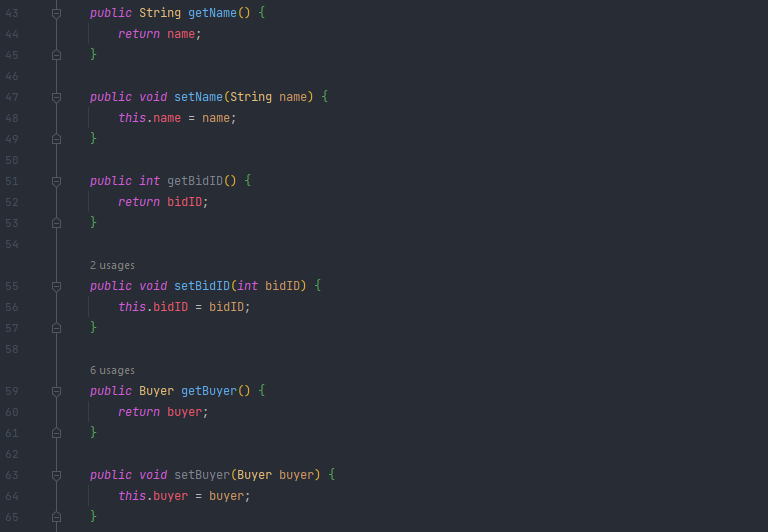
The Bid class contains certain data from Millers, farming agencies and agriculture agency.



I have used various set of rules to declare according to certain types of users.



We are calling the objects from the constructor/method below.

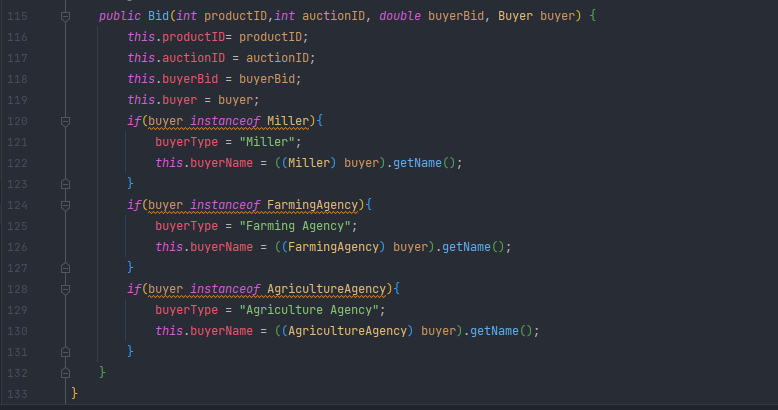


This shows what to display on corresponding data input



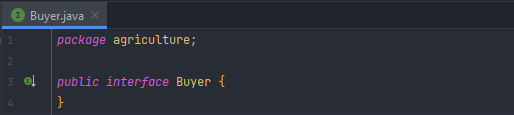


We use more constructors to update the set of rules.



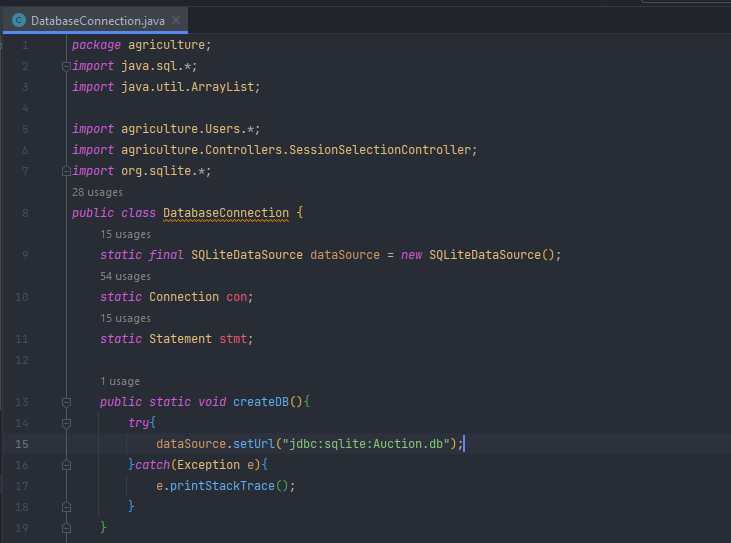
#### Buyer

Buyer classes are implemented in Miller, FarmingAgency and AgricultureAgency. It is not in Farmer class because farmers cannot purchase anything.

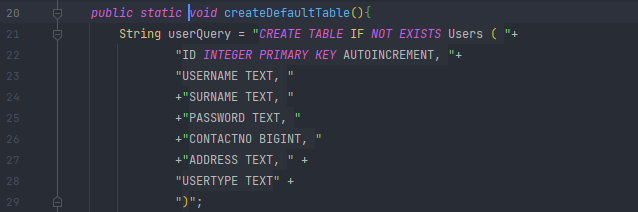


#### DatabaseConnection

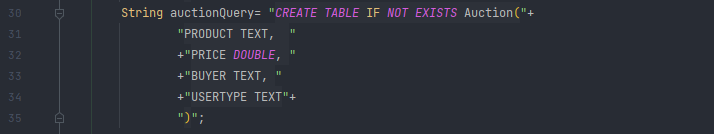
This class connects and manages our database connection. In my scenario it is SQLite.



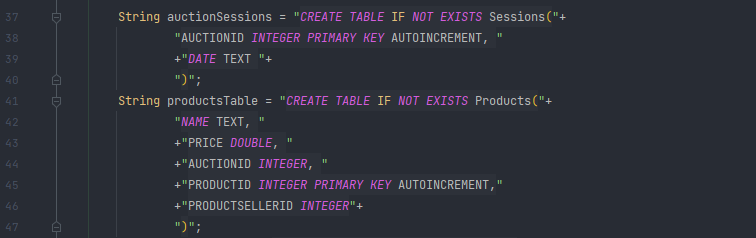
The below coding creates various types of tables with columns to store data. All the data types are declared within the table to minimize errors.  
The User table is created below



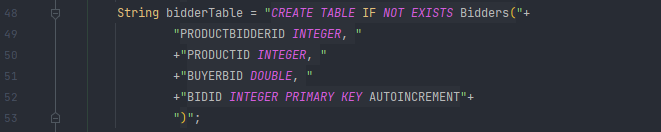
Auction Table is created in here



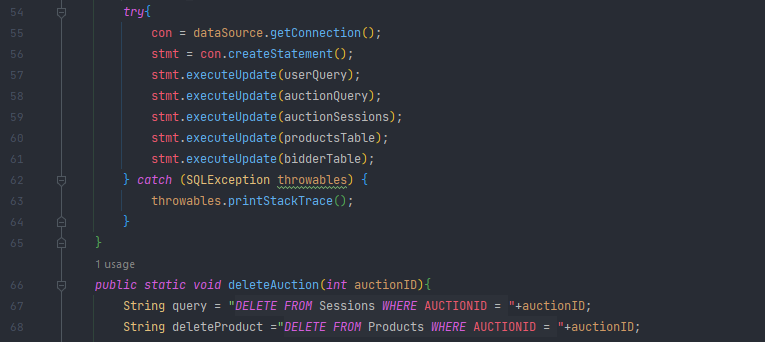
The session and Products table is created



The Bidders table is created in here



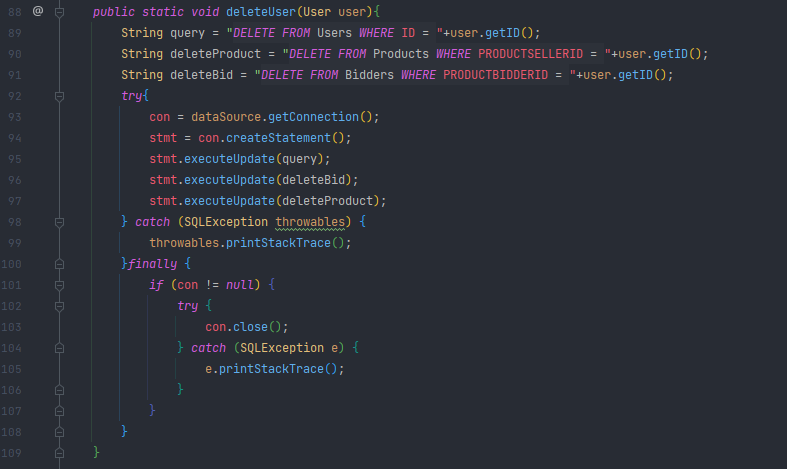
We are using try catch to give out an exception



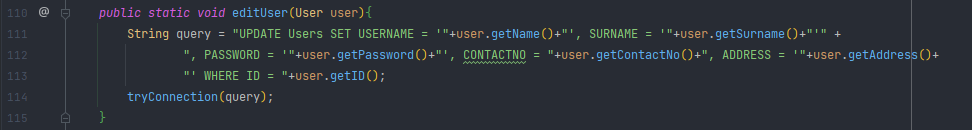
Stmt:- The Statement interface provides methods to execute queries with the database.



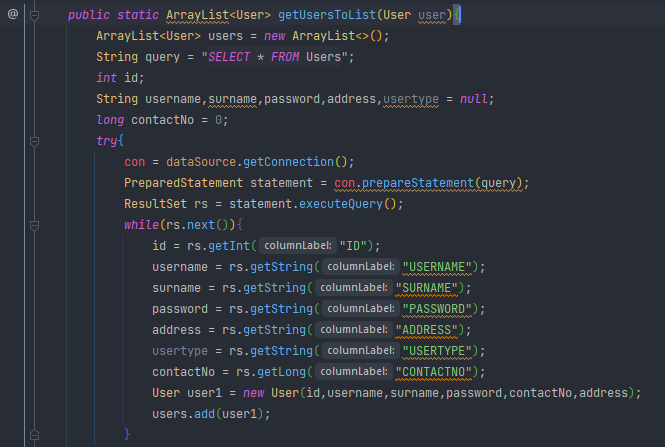
The below coding consists the functions of deleting a user

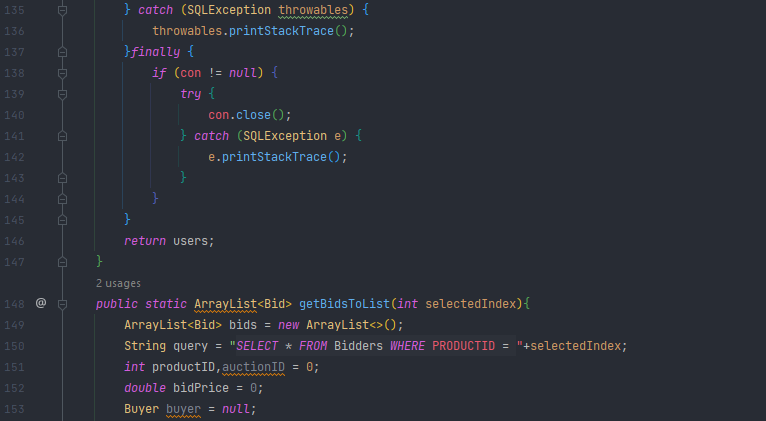


This is to update the users information once it is deleted



ArrayList in java is used to store elements





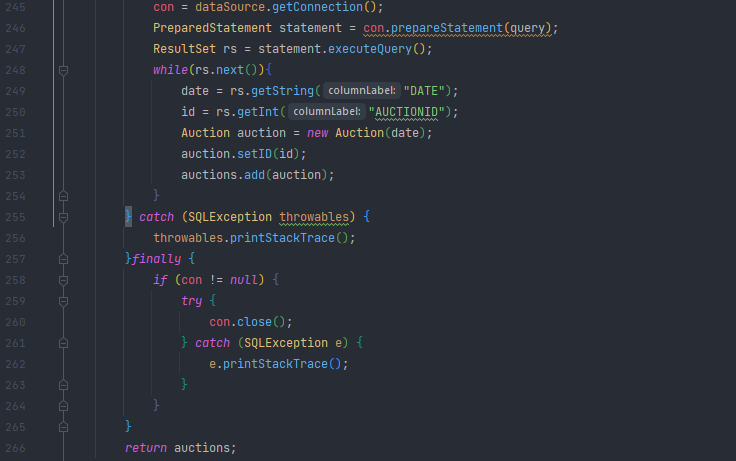
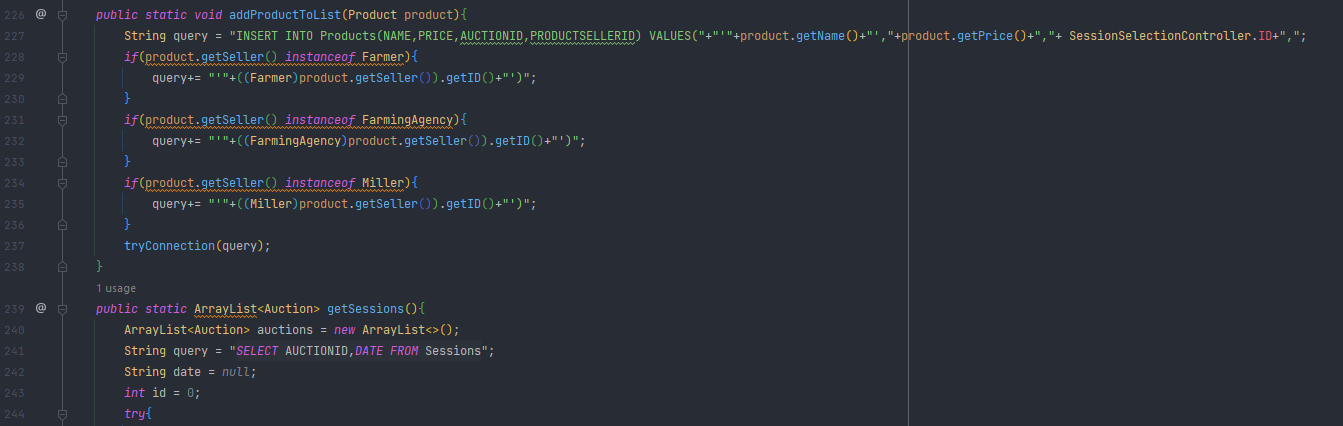
We are using stacks in Java and it updates the information based on the user input.

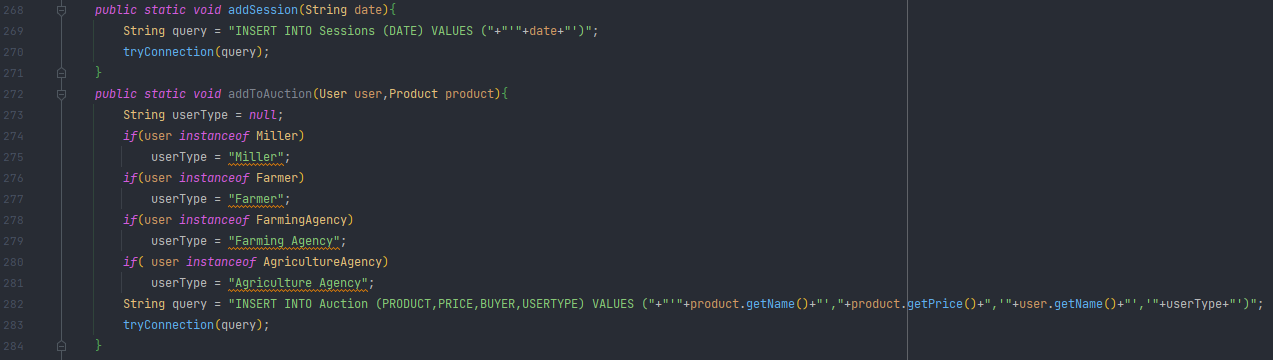


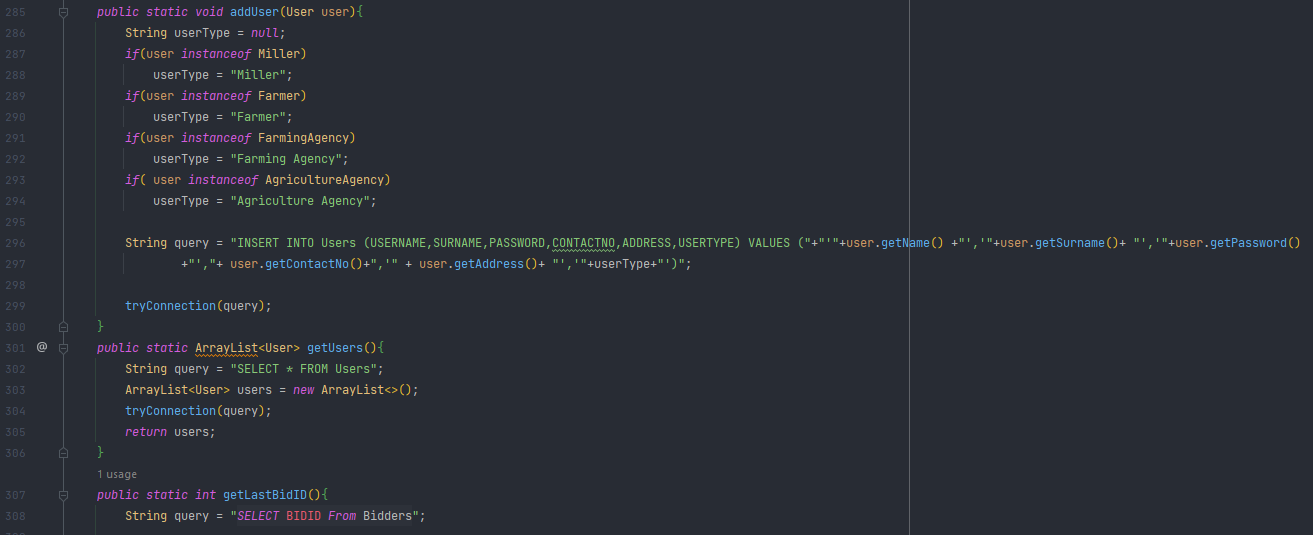


The stack trace listing provides a way to follow the called stack to the place in the method where the exception occurs.









As similar as before in this stage we connect to the database and display the necessary data



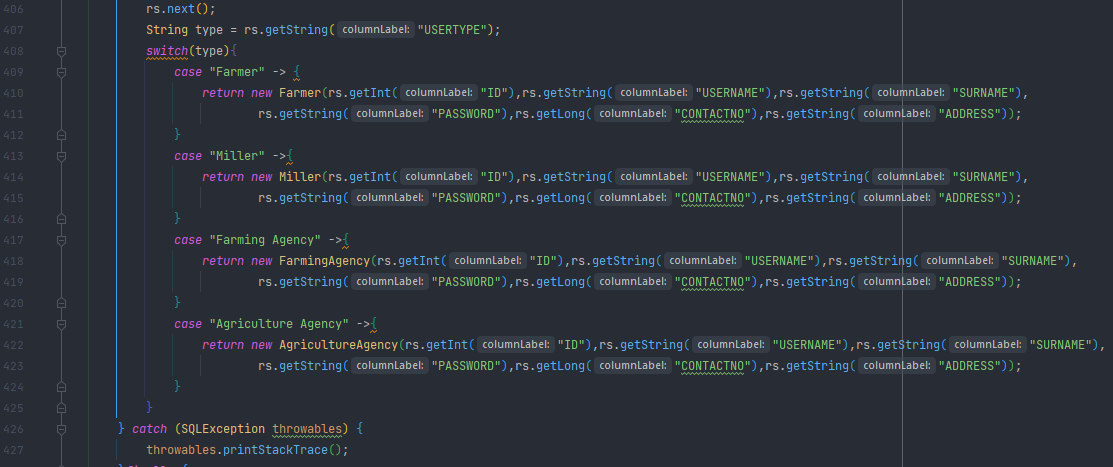


This is to display the relevant page when the user enters his username and password





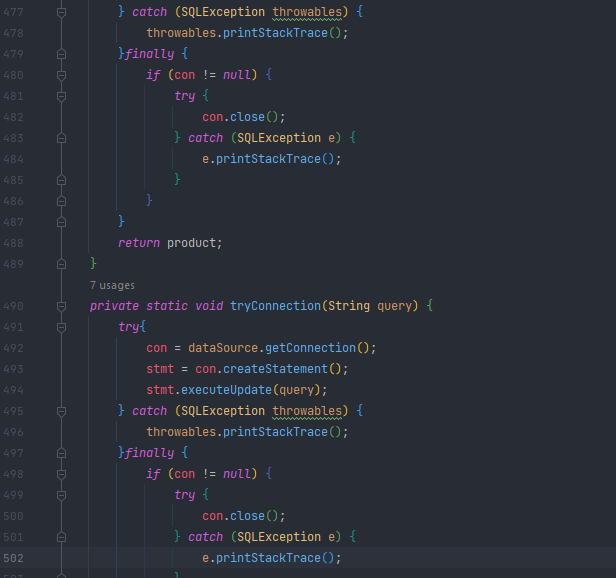
In this case we are using the switch function to execute one statement from multiple condition



“Finally” is used to close the code no matter if they are executed or not

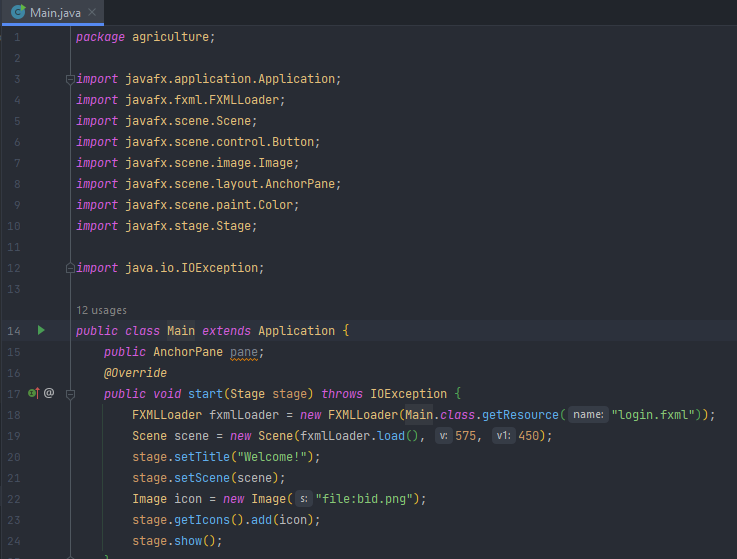






#### Main

The Main class is the class which we run the file. The starting point for JVM to start execution of a Java program where without a Main class, the JVM wont run.



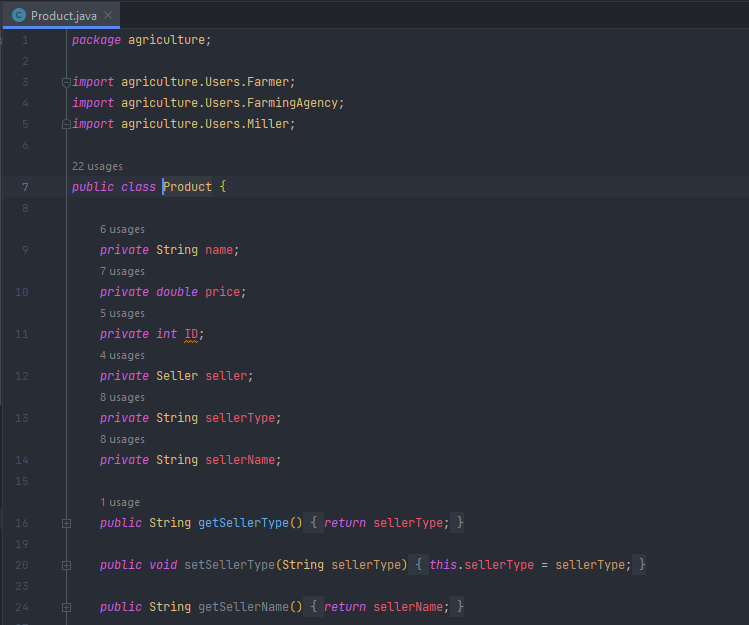
We are using the override functionality to our fxml file to execute them together.



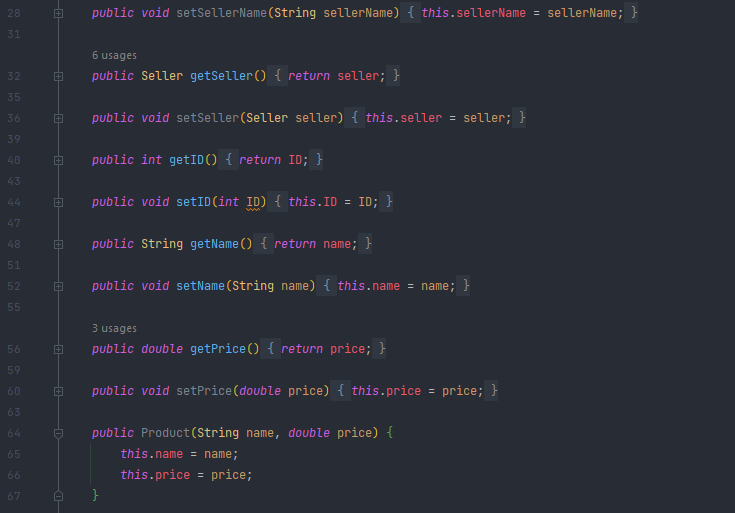
The stage in the Main class is the container which hosts the JavaFx visuals.

#### Product

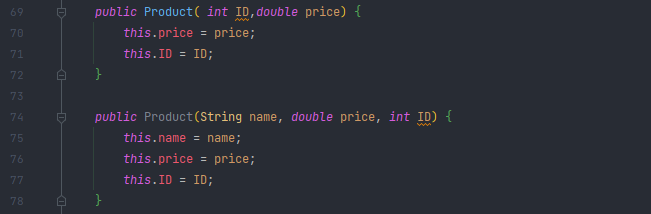
This class contains information such as product type and it is linked to the auction class



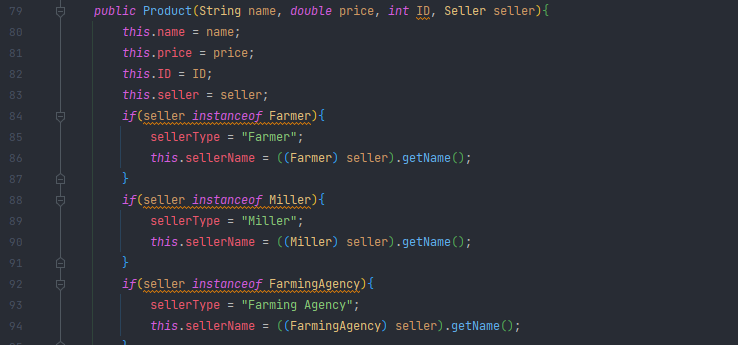
We have called all the necessary data from the necessary classes.



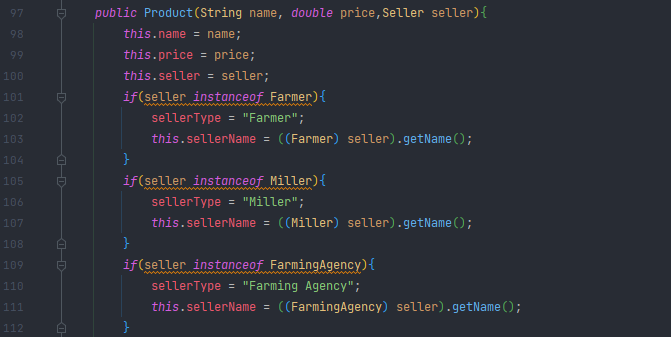
We have created the objects in here to make our process easier.



Then we declare variables for our objects

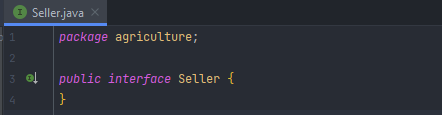


We set conditions to display relevant information for relevant users



#### Seller

The seller functions are used in classes such as Farmers, Millers, Small Farming Agencies. AgricultureAgency does not use this because he cannot sell.



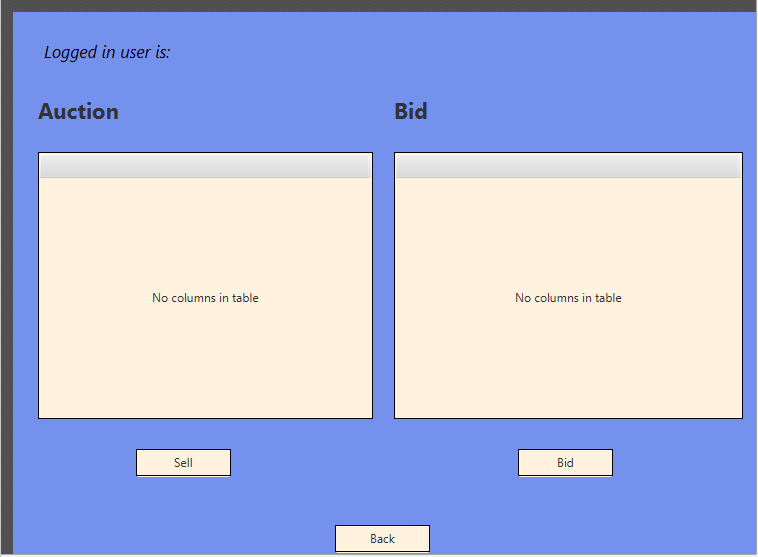
### Resources Package

The resource package includes all the fxml files we created from the scene builder in our applications. FXML is a file format of JavaFx which is used to create various GUI.

These classes are controlled by our controller classes.

Our GUI looks like below:-

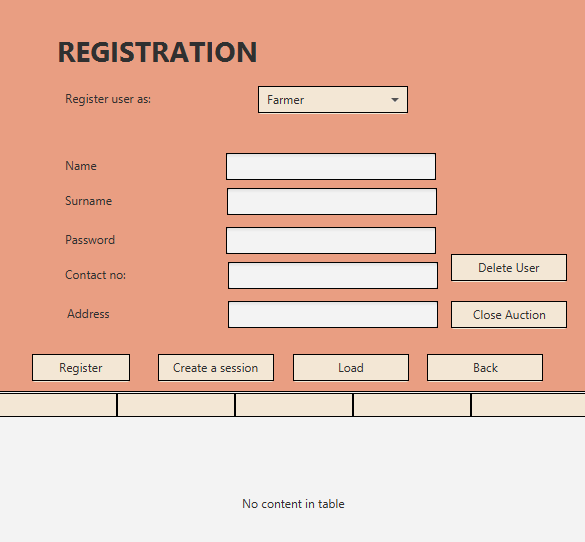
#### auctionMenu



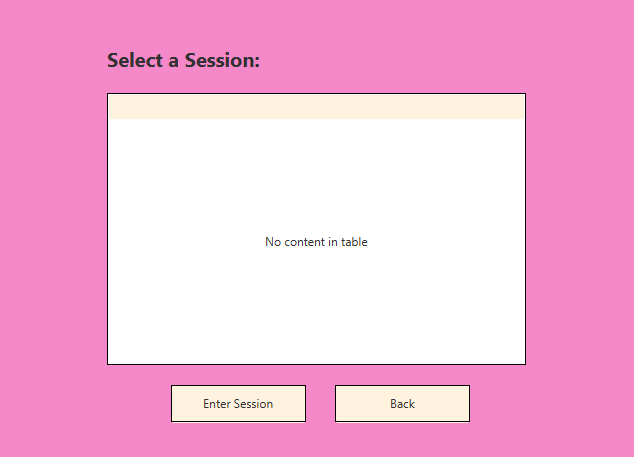
#### login



#### registration



#### sessionSelection



### Controllers Package

The Controllers package sets various functions for the fxml files which is in the resource folder.

#### AuctionMenuController

In this class we have linked the auctionMenu.fxml and this class to collaborate together. We also have imported the necessary database data and information to display and also configured what happens if we click certain buttons and what are the functions.

Main Functions includes:-

* Placing a bid
* Selling a product

#### LoginController

We have connected this class to the Login.fxml class. Here not only we have linked our actions but also we have made sure if the password is correct then return to the system or if any inputs are wrong, we display an error message.

Main Functions includes:-

* Logging into the system
* Displaying error messages if the inputs are wrong

#### RegistrationController

This class woks with registraton.fxml file. This controls all the admin functions

Main Functions includes:-

* Register a user
* Edit a user
* Delete a user
* Load Details
* Create and auction session
* Delete an auction session

#### SessionSelectionController

This class is connected with the sessionSelection.fxml javafx file. This class consists and configures about the sessions created by the admin.

Main Functions includes:-

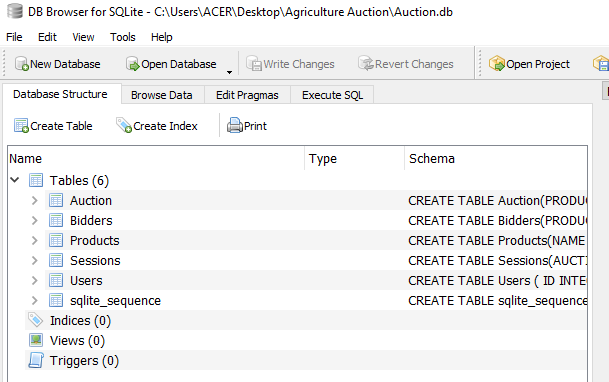
* Showing all the current auction sessions
* Ability to enter an auction session

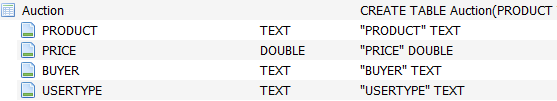
### Image Files included

* auction.png
* bid.png
* form.png
* resgistration-form.png

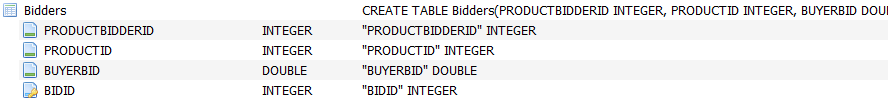
Database

This is our auction database

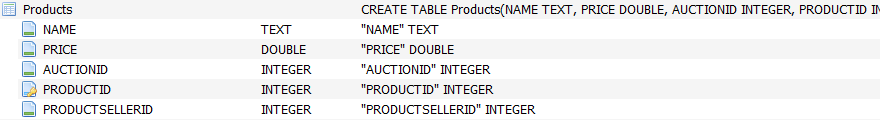


The Auction table includes

Bidders table includes



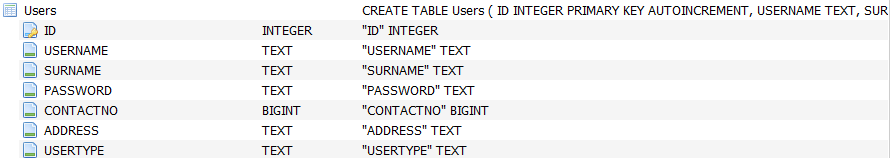
Products table includes



Sessions table includes



Users table includes

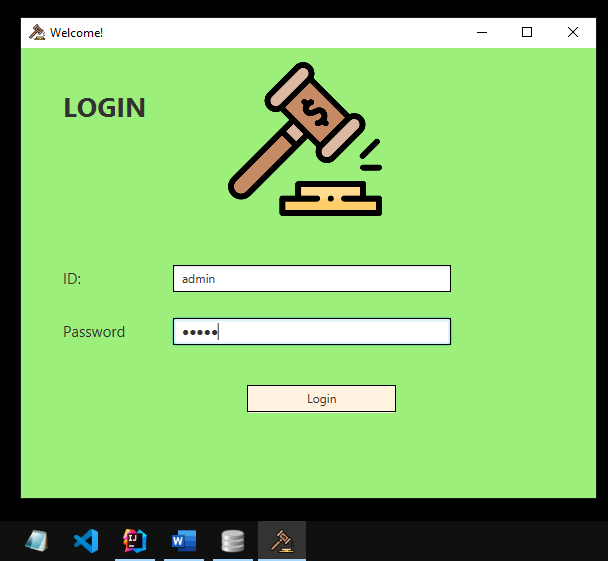


Running the Application.

Once we run our Main class file, our application will start to run.

You will be getting a screen as Follows

### Admin



The application icon is “bid.png” which we added for the aesthetics.

Then the admin Window will popup. In other words it will call the registration class.

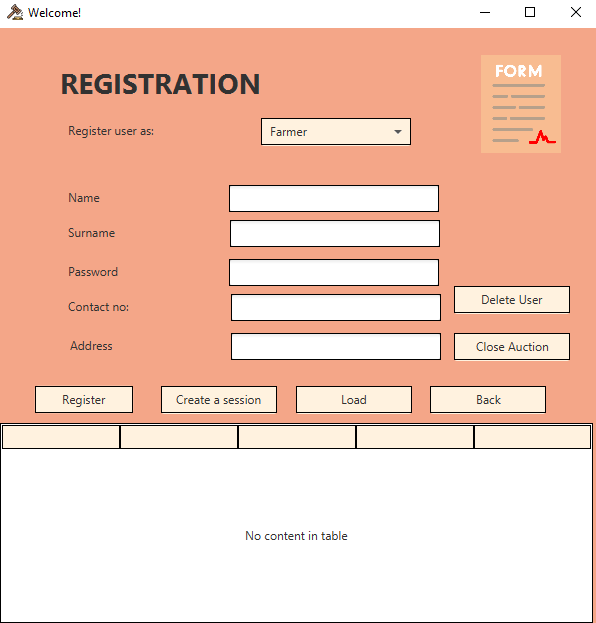
Let us see the functions which the admin can do.

#### Logging in

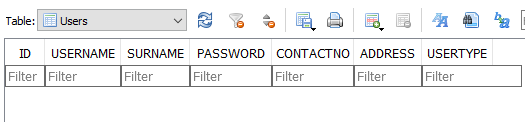
I will be entering my admin credentials as.

ID:- admin

Password:- admin



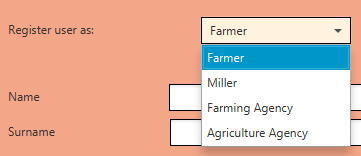
The table displays “No content in table” because currently we don’t have any registered users.



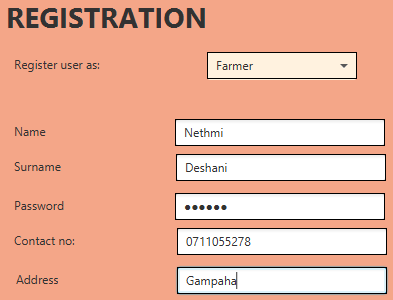
#### Register Users

As we saw the registering process in details, now let us add a few users to demonstrate the system

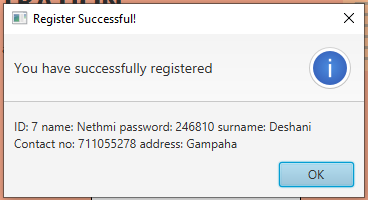
When creating a user you can select the user type from the drop down menu

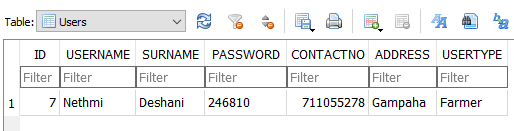


Let us now add our first user.  
As per the example in our case let us say this as a farmer and lets click the register button

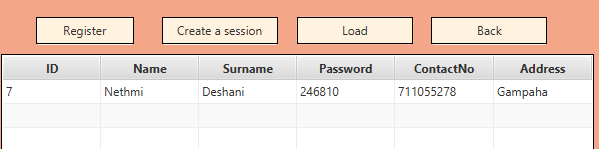


After you click the register button if everything goes well then you will get the following message

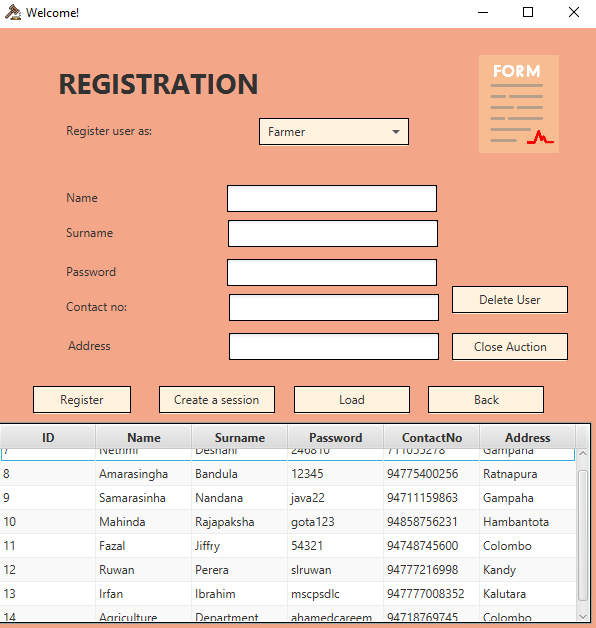


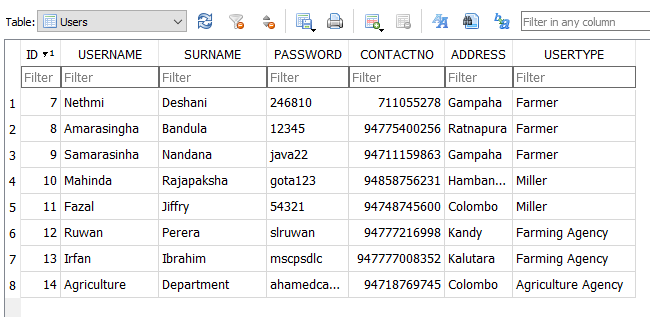
And if you refresh the database we can see that the database has been updated 

And once we click the load button the details will be shown to the admin



Let us add a few more users to diverse our database and explain the functions further





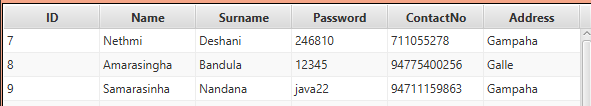
#### Edit a User

As an example let us say that we need to change the address of the user ID 8.



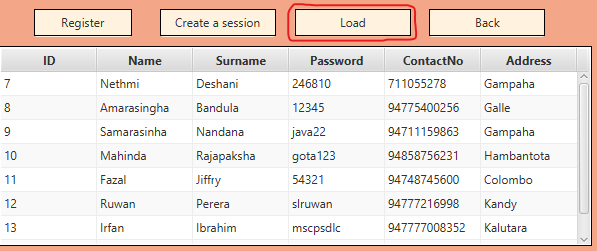
To edit a user, the admin can directly edit them by clicking the user in the table and then clicking the area you want to change the details



And after you edit it you just need to click enter and it will get updated 

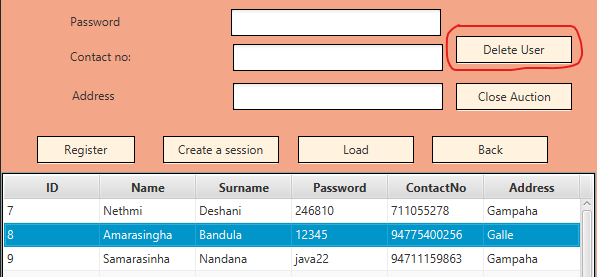
#### Load user details

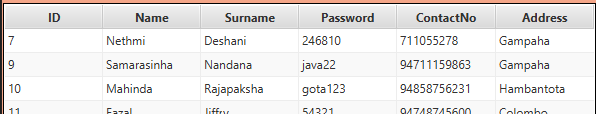
As we saw previously if the admin clicks the Load button all the necessary data in the database will be displayed



#### Deleting a user

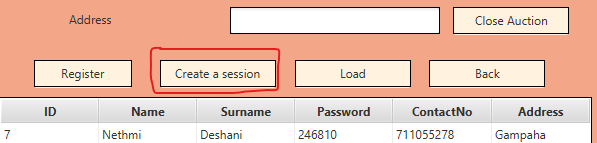
Let us say that the admin needs to delete the User ID 8. Then what he needs to simply do is just click the user and click the Delete User button



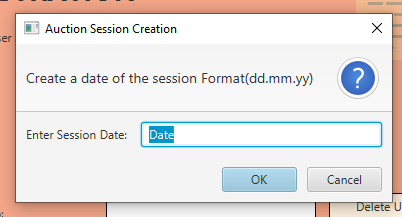
Then you will notice that the User ID 8 has gone

#### Create auction session

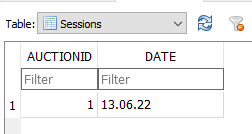
The admin can create a session for the auction where the users can sell or place bid on the products.  
It can be done by simply clicking the “Create a session” button



Then you will get a pop up session asking for a date. So let us simply add it

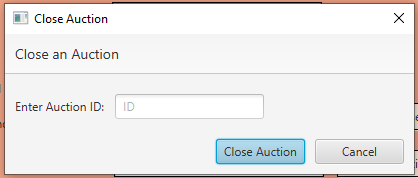


After we add a session our database will get updated



#### Delete an auction session

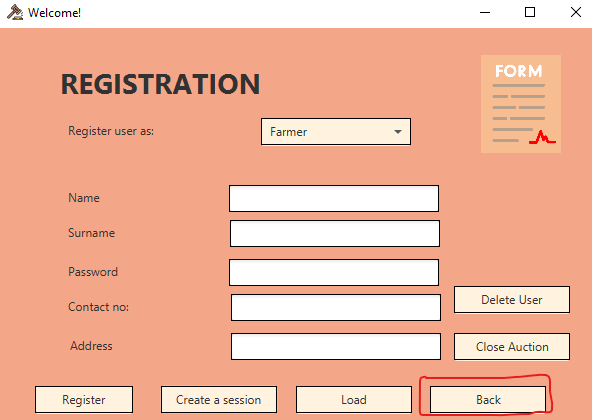
As same as deleting a session the admin can close an auction session once the trading has been completed for the day. To do that you need to click the “Close Auction” button.

The you will get a pop up windows as follows

To delete a session you need to then enter the auction ID. In our case it is “1”. After entering the auction will be removed from our database

#### Logging out

And if we click back the admin will logout from the system and go back to the login page

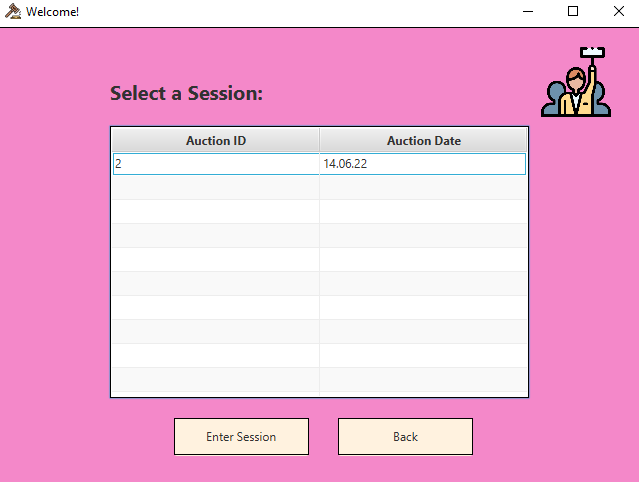


### Farmer

#### Logging in to the system

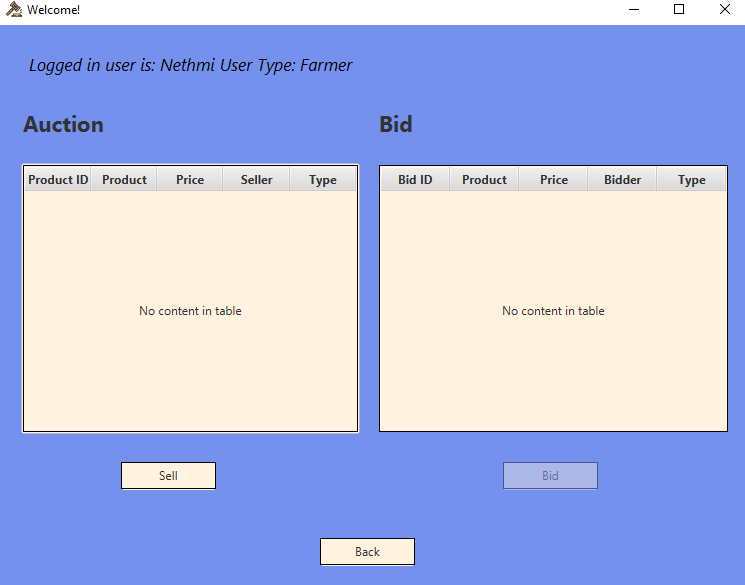
Now let us log into the system as a farmer

Let us enter ID :- 7 Password:- 246810

Then the Farmer will be greeted to the auction page where he/she can enter a session by clicking it

#### Entering the auction session

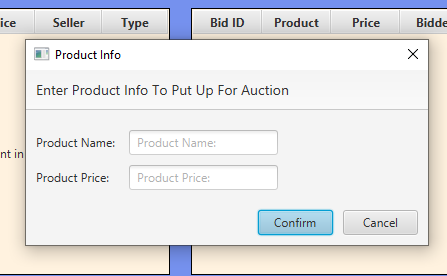
Once the farmer clicks “Enter Session” he will be taken to the following page  
The page will display the Users name as well as the User Type



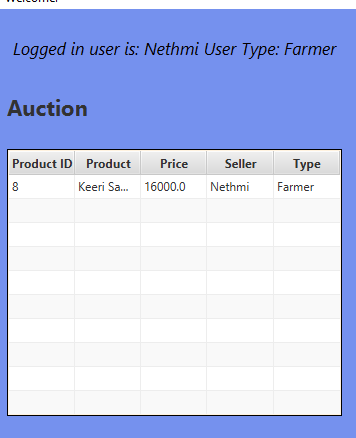
The Bid option is greyed out because a Farmer can only sell.

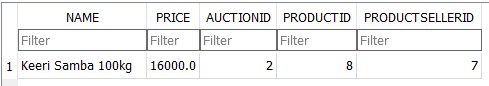
#### Selling a product

To sell a product the farmer needs to click the “Sell” button.  
Then he will get a pop up window asking to enter the details



Once we add details and click confirm, the details will be posted to the system and the database will be updated



You also can see that our database has been updated

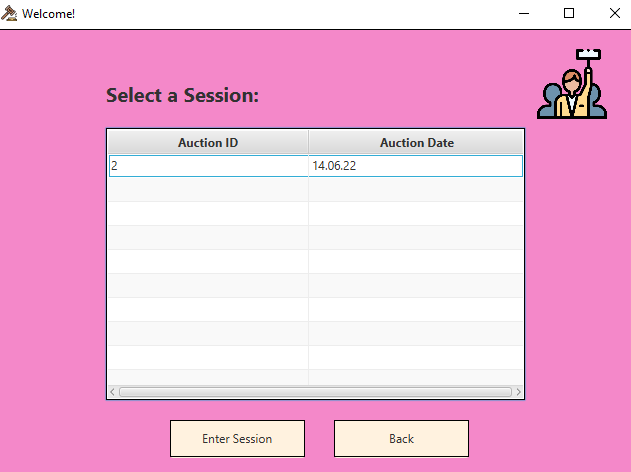
#### Logging out

The farmer can also logout from the system after clicking the “Back” button

### Miller

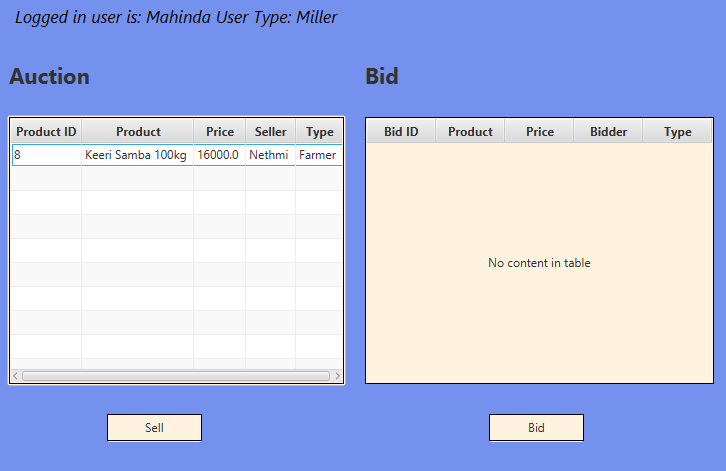
#### Logging into the system

Let us log into the system.  
As an example let us enter as ID:- 10 Password:- gota123

Then the Miller will be greeted to the welcome page

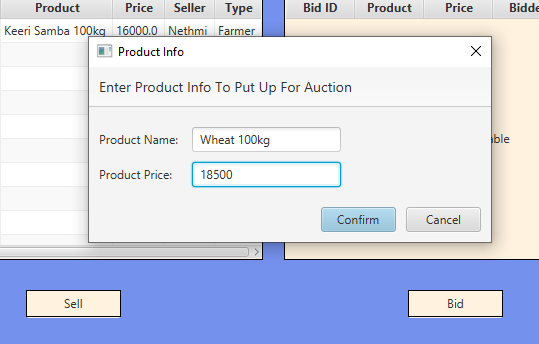
#### Entering the auction session

The user can simply select the relevant auction ID and then click “Enter Session” button to enter the auction session.



The Miller can sell a product or else place a bid on a product where a farmer has auctioned

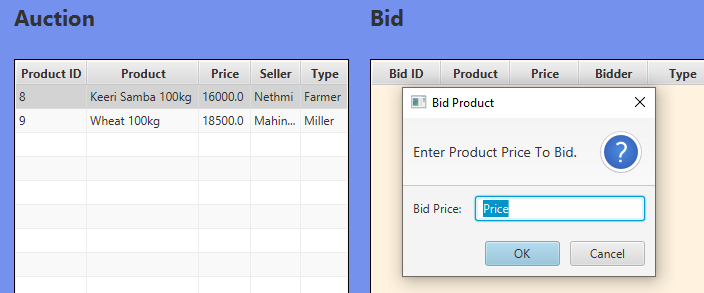
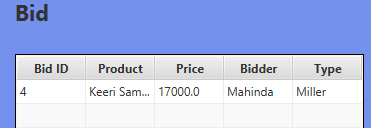
#### Selling a Product

Once the miller clicks the “Sell” button it will ask for the details.

Once you enter the details then you need to click confirm. And then our system will be updated

#### Placing a bid

Miller can buy from a farmer. So let us test this theory.

To bid to a product, you just need to select the product to bid and click the ”Bid” button. Once you enter the value the table and the database will get updated

#### Logging out

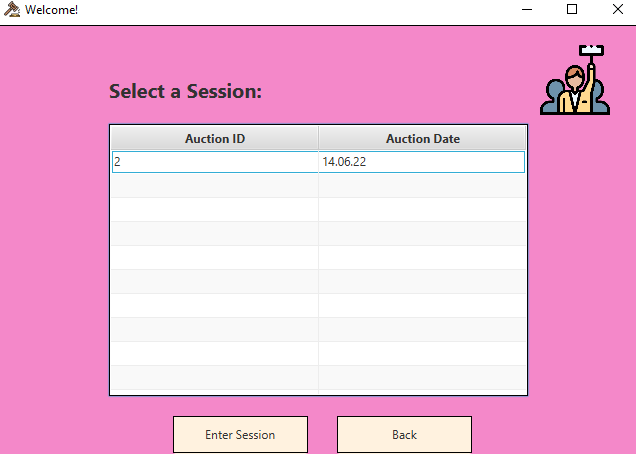
The Miller can log out the system by simply clicking the “Back” button.

### Small Farming Coperate Agency

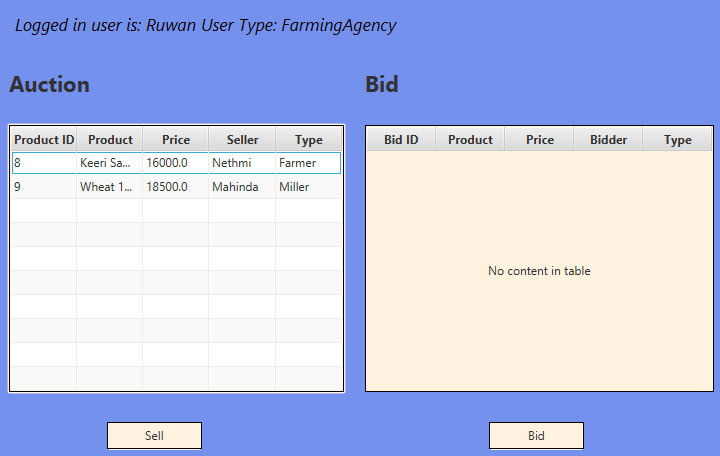
#### Logging in

Let us enter to the login page as the below mentioned example:-

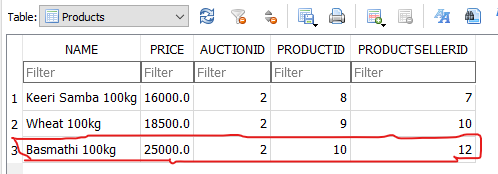
ID:- 12 Password:- slruwan

Then he will be taken to the welcome page where he can enter the auction 

#### Entering the auction session

To enter the auction session as we discussed previously , same as others the farming agency should click the auction and click “Enter Session”.  
You will see the interface like this 

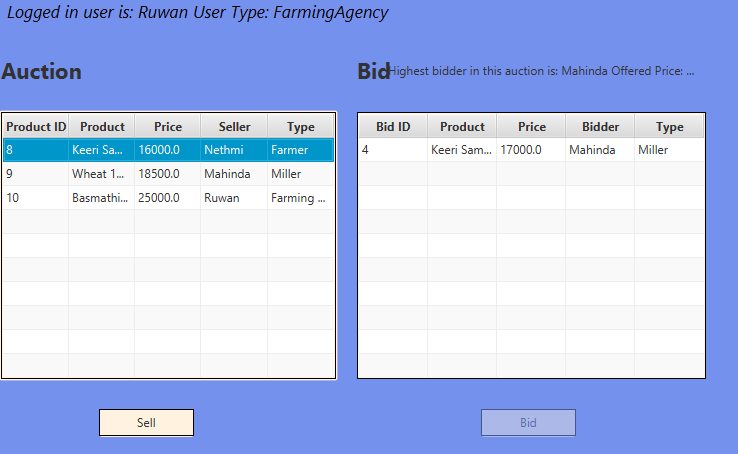
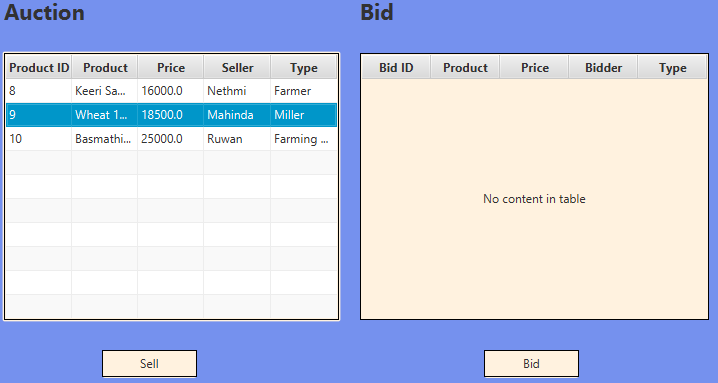
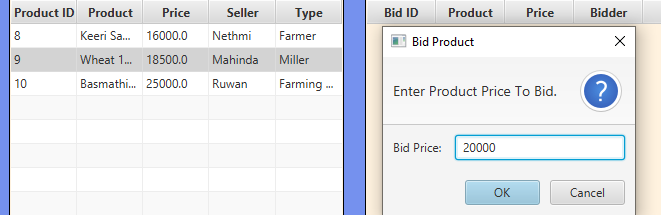
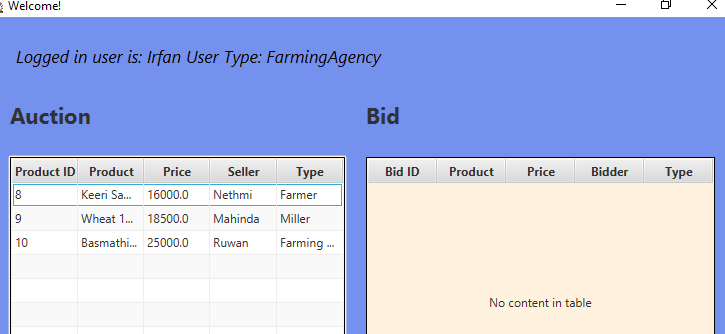
#### Selling a product

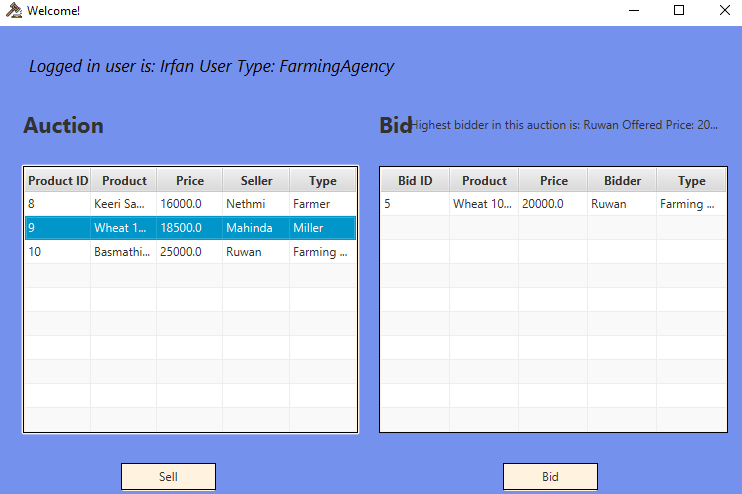
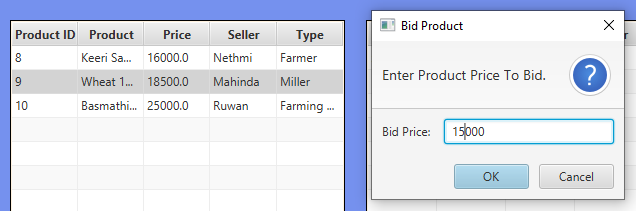
If the Farming Agency needs to sell he can click the “Sell” button and enter the details and then click “Confirm”. The details will be entered to our system and our database will get updated. 

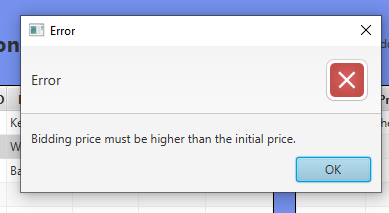
#### Placing a bid

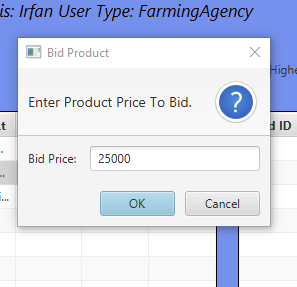
There is a rule when it comes to placcing bid as a SF Agency.

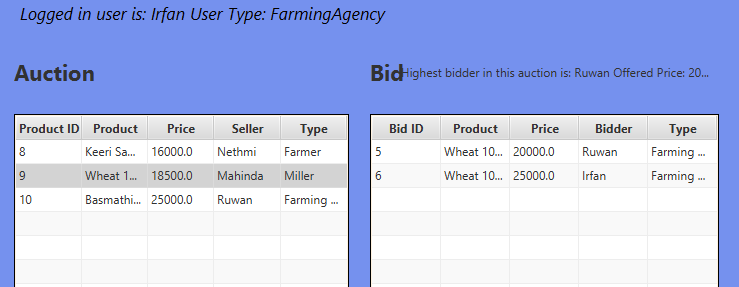
You can only place bid from Millers and not from Farmers. Let us test this out

  
As you can see in the above pic if we click a farmer the Bid button will grey out and block the ability to place bid into the system.  
But when he clicks on a Miller, the bid button will work And let us enter a value   
Now to check how the system works, let us login as another Miller and place the bid to see.   
The window will show as below  


Now if we click the Miller then the database will show the current bid value and we can place the bid which should be higher than the previous value.   
Simply just click the Miller and place a bid more than the current price.  
Let us test by entering a lower value   
If the SFA enter a lower value, you will see something as below



The system wont get data from a lower value.   


And then you can see that our value will be updated  


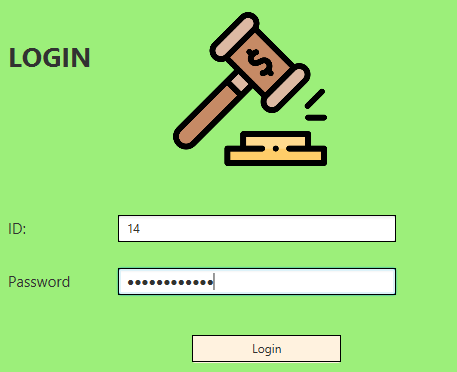
#### Logging out

As same as the others the user can logout from the system by clicking the back button

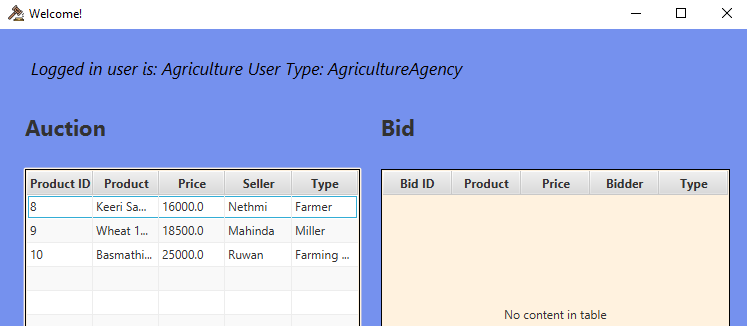
### Agriculture Coperative Agency

As we know the Agriculture Cooperative Agency is the final user of this system. Since there are many agriculture cooperative agency branches among sri lanka we have decided to implement them too in the system..

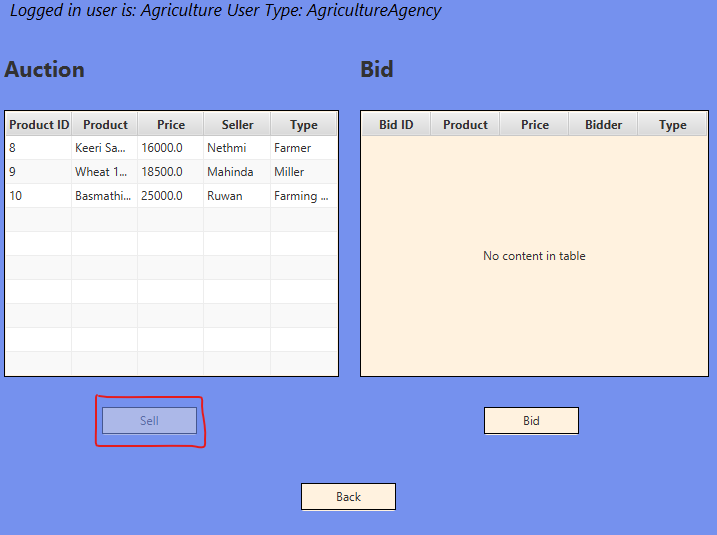
#### Logging in

Lets log into the system by providing necessary credentials

#### Entering the auction session

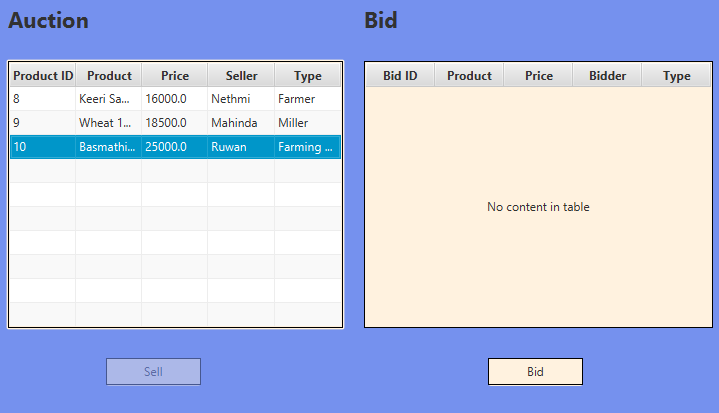
As same as the others the agency can log into a session by selecting the auction and clicking the “Enter Session” button.  
Then you will be able to see the auction page 

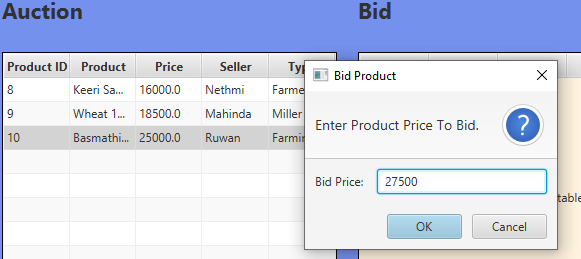
#### Selling a product

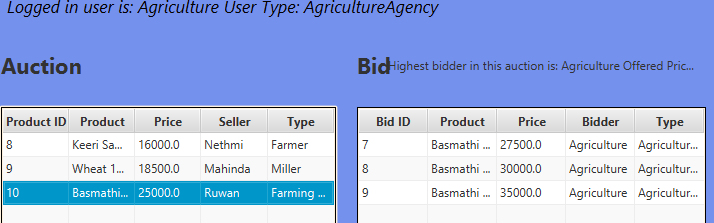
Since the agricultural agency cannot sell products the Sell button is greyed out 

#### Placing a bid

When it comes to bidding on a product we can only place bids for the products auctioned by Small Farming Agencies



Agricultural agency can then place the bids 

We can also place multiple bids as a user 

Once everything is finalized the admin can later close the auction.

#### Logging out

And by clicking the Back button you will be logged out from the system and taken back to the homepage.

Now you will have the complete understanding of this application system.

Insights, Improvements and Ideas

# LO4

## Test Case

A test case are certain actions took to verify a function or a feature. There test case has specific variables which uses to determine a software production software functioning software are you.

Few Test cases completed

Checking the results on entering the ID and password.

Testing responses got from the participants

Checking white the buttons are working or not

Testing techniques followed in my system.

* Boundary Value Analysis (BVA):- Defining testing boundries for a specified range of values

I tried this to test out the highest bid on the auction session panel

* Equivalence Partition (EP):- Dividing into equal groups and testing which has the same behaviour

I tried using this when I test the datatypes of users

* State Transition Technique:- Used when application bheaviour changes from one state to another when perform an action

Tried this when I entered the auction where the options included such as open in the same window or create a separate window

* Error Guessing Technique:- Guessing the errors which may arise while doing manual testing

### Deadline Schedule

|  |  |
| --- | --- |
| **Date** | **Subject** |
| May 20th | Project Background Gather Software Information |
| June 1st | Submit Draft design Get feedback from client Showcase them different options |
| June 5th | Submit detailed design Getting the approval |
| June 10th | Present the working software |
| June 15th | Asking for changes and fixing them |
| June 18th | Final Approval and Light edits |
| June 20th | Presenting the Final Version |

# Quality Assurance

Purpose

The purpose of this Software Quality Assurance (SQA) Plan is to establish the goals, processes, and responsibilities required to implement effective quality assurance functions for the agriculture auction project.

### SDLC Activities

1. Project Planning and Oversight
2. Software Development Environment
3. System Requirements Analysis
4. System Design
5. Software Requirements Analysis
6. Software Design
7. Software Implementation and Unit Testing
8. Unit Integration and Testing
9. System Qualification Testing
10. Software Use Preparation

#### Changes Made in the System

* Change of colors of the windows
* Client said to add the phone number feature so I had to add them
* Ability to group user types and data together

#### Reviewing Software Tools

* Making sure the necessary tools are already there
* Each tools are reviewed and monitored to make sure they are on a perfect working stage to minimize interruptions in future

#### Project Planning and Overseeing

* A software development document is prepared and documented with all the plans. This helps the customer to understand about the software.

#### Product Assessments

The following are typical product assessments conducted by me to get the best out of this software.

* Peer Review packages
* Document Reviews
* Test results

### Process Assessments

The following are typical process assessments taken by me.

• Project Planning

• Project Monitoring and Control

• System Reviews

• Peer Reviews

• Requirements Management

• Software Configuration Management and Configuration Audits

• Test Management (Verification & Validation)

• Software Problem Reporting and Corrective Action

## Performance of the application against the Problem Definition Statement and initial requirements.

### Problem Statement

#### Problem

Currently the agricultural department does the auction system in the manual process. There are so many problems and costs happening due to this process.

#### Current Performance Issues

* The process is done manually:- This can lead into willing and unwillingly human error which can affect the overall auction system
* Market manipulation:- Can manipulate the market by adding false information and prices
* Low satisfaction due to unfair wealth distribution (Some users might know the bidders and some will miss them which unsatisfies them)
* Heavy workload:- There should be more than 1 employee to do the necessary tasks since doing by an individual is impossible and time consuming.
* High Cost :- Cost is incurred for papers, ink and time. They should also unnecessarily pay salaries for unwanted employees.
* Security Issues :- The data can be stolen or destroyed which can affect the records.
* Portfolio Misplacement:- The documents and information of the users can be misplaced hence it will take extra time to reregister them and check the records.

#### Target Performance

* The process is done manually:- Everything should be automated to save time
* Market manipulation:- All the auction details should be shown to everyone.
* Low satisfaction due to unfair wealth distribution :- There should be a system where the bidder to know the highest bid value.
* Heavy workload:- Since everything is automated only a trusted employee to manage the admin account is sufficient
* High Cost :- Low cost compared to utilizing other resources and paying salaries for more employees
* Security Issues :- All the accounts should be encrypted. The auction system as well as the users individually.
* Portfolio Misplacement:- The registered users data should be stored in a safe place. If there is a backup function it would be more ideal.

#### How to resolve the problem

* Create a software application to run the auction system suitable for the agriculture agency
* The system should be encrypted
* It should satisfy all the previous mentioned requirements where it eliminates the current problems

#### Requirements

* A suitable computer device to run the application
* A monitor or a tv to display and view the system
* A keyboard & Mouse or a touch screen to enter the data to the system
* A database software such as MySQL or SQLite to be installed to store the data offline.

### Performance of the Application

The Application performs well without any issues. It covers all the requirements and competes with the expectations.

### Assumptions when building the system

* Since the average user doesn’t know much about technology there is no forgot password option. To change the password of the user it should be done from the admins end who is a very trusted employee
* Made the system minimal as possible since there is no high end devices in Sri Lankan government office

### Functional Requirements

#### Admin

* Log into the system
* Register new users
* Delete registered users
* Edit users details
* Load and view user details
* Create an auction session
* Close an auction session
* Export or backup data to another device by exporting the data from the database

#### Farmers

* Log into the system
* Join auction session
* Veiw auction details and the sellers and bidders details along with various products
* Sell products to Millers
* Log out from the system

#### Millers

* Log into the system
* Join auction session
* View auction details and the sellers and bidders details along with various products
* Placing bid on farmers products
* Selling items to Small Farming Agency
* Log out from the system

#### Small Farming Agency

* Log into the system
* Join auction session
* View auction details and the sellers and bidders details along with various products
* Placing bid on millers products
* Selling items to Agriculture Agency
* Log out from the system

#### Agriculture Agency

* Log into the system
* Join auction session
* View auction details and the sellers and bidders details along with various products
* Placing bid on small farming agency products
* Log out from the system

## Strengths and Weakness of my Application

### Strengths

* User-friendliness :- Userfriendly system makes the non technical influenced or people with zero knowledge to interact and work with the system
* Fast :- Ability to Insert, delete and edit data very fast
* Efficient :- The software is easy to maintain and update in the future since I have used java OOP concepts
* Cheap :- It is cheap and economical to maintain. Can also run in a low end machine
* Multithreading:- Since I developed the application using Java, java language supports multithreading and it can work very fast on devices with a good CPU. We can run more than one thread at a time.

### Weakness

* Storage:- Since the application stores data and information offline, we will need a large storage space to store the gathered data
* GUI:- The graphical user interface is minimal and doesn’t attract a lot.
* Memory:- Since the application is created using Java it will need memory/ RAM to run. The JVM performs various backend functions that decrease the speed of the program.
* Complex to update:- Since the software is divided into to many class such as controllers and users it might be difficult to add new features in future.

Problems I identified in my Software

* Lack of Security in database and backup
* It would be good if we can pay through software itself
* Many farmers and millers doesn’t know English Language

## How to improve my Software

* Using APIs to make the data storage more online based and secure
* Including Payment gateways to pay through the system it self
* Adding Sinhala and Tamil languages since most of the Sri Lankans doesn’t understand English

# References

Duggal, A., 2021. *hevodata.* [Online]   
Available at: https://hevodata.com/learn/sqlite-vs-mysql/

guru99, n.d. *guru99.* [Online]   
Available at: https://www.guru99.com/v-model-software-testing.html

michaelpage, n.d. *michaelpage.* [Online]   
Available at: https://www.michaelpage.com.au/advice/career-advice/productivity-and-performance/top-7-sdlc-methodologies

Singh, V., 2022. *hackr.io.* [Online]   
Available at: https://hackr.io/blog/sdlc-methodologies

softwaretestinghelp, 2015. *softwaretestinghelp.* [Online]   
Available at: https://www.softwaretestinghelp.com/sdlc-methodologies/

team, s. e., n.d. *synopsys.* [Online]   
Available at: https://www.synopsys.com/blogs/software-security/top-4-software-development-methodologies/