**Comp 730 + 830 Final Software Project Update 4**

Team 1: Sean Burwen, Greeshma Srinidhi

**Past Project Progress:**

In the last update, the cart page was created with most of its intended functionality in place. This week, the cart has been improved in a fundamental way. This improvement was the addition of a basic observer called SimpleObserver.java which can notify the cart window when listings are removed from the cart. This triggers a method in Cart.java which refreshes the screen to only include the remaining carted listings visually and recalculates the total. In addition to the cart improvements, the checkout window was created and completed as well. When the user clicks the “Check Out” button on the cart, a dialog box using a JFrame called CheckOut.java is opened. This screen prompts the user to fill out their payment details and displays a list of what’s currently in the cart along with the total price for the payment. When the payment is submitted, another popup message is displayed which informs the user that their payment has been completed for the total price. When an item is purchased, the quantity of that item is reduced.

Worked on implementing user authentication and created GUI for the login screen and user should be able to get login screen prompted when we click on login button and hardcoded the username and password in code to check the functionality.

**Sean’s Activities:**

* Created SimpleObserver.java
  + Created SimpleObserver object in Browser.java and passed it through to each class which needs to use it (Cart.java, Listing.java, CreateListing.java, Database.java).
  + Implemented notifyRemovedFromCart() method.
* Updated Cart.java
  + Implemented removedFromCart() method to update the screen and total when items are removed.
  + Implemented functionality to “Check Out” button which calls doCheckOut().
  + Implemented doCheckOut() method which handles the checkout process and is called by the checkout button. This method opens CheckOut.java.
* Updated Browser.java
  + Implemented reinitialize\_listings() method to update the screen each time the user returns to it.
* Created CheckOut.java
  + Created visual design for the check out window, which has a similar functional purpose and design to CreateListing.java.
  + Contains text fields for the user to provide their credit card number, expiration date, security code, and full name.
  + Takes the listings\_array as part of the constructor and creates a JList from it which shows the user what items are in the cart at the time of checkout.
  + The “Complete Purchase” button makes it look like a card has been charged (it cannot take real world credit card payments) and shows a message to the user that the purchase has been completed for the total dollar amount.
    - All carted listings are removed from the cart, the quantity of each is reduced, and the checkout window is then closed.

**Greeshma’s Activities:**

* Database
* Created a new table named UserAuth in Store.db
* Inserted data into the new table.
* Database.java
* Written a functionality to add and remove the username.
* Written a functionality to check Username is present in database or not.
* Written a functionality to check Password is present in database or not.
* Written a functionality to add new Username to the database.
* Written a functionality to add new Password to the database.
* Browser.java
* Written functionality for User Authentication.
* Written functionality for adding a new button edit profile.
* Written functionality to check Username and Password is present in database or not.
* Written functionality for user to sign up if the username is not present in database.
* Enhanced GUI for showing a pop-up box to the user whether to sign up or not.

**Next Steps:**

1. Create a process to generate some sort of itemized receipt.
   1. This may be a text file or a simple dialog window which could be saved or printed somehow.
2. Work on extra features from the 100% achievement goal.
   1. Add a filter or search functionality to Browser.java.
   2. Create an admin account which can edit or delete listings.
   3. Add more information fields to each account (phone number, email, etc.).

**Difficulties:**

One of the biggest difficulties of this week was experienced when trying to implement the simple observer. Getting the functionality to flow the intended way was a struggle but it is now working after some research. Another difficulty of this week was in trying to show the carted items in the JList in CheckOut.java and having it look right. This was also solved after some trial and error.

Faced difficulties while creating a new table in SQLite and while implementing user authentication in the browser involved integrating with the database to validate usernames and passwords. Handling authentication errors and implementing functionality to allow users to sign up if their username is not present in the database requires coordinating with the database operations. One more challenging thing was to ensure that the button behaves correctly and is visually appealing.