

E-Commerce Application

An e-commerce application will be developed that primarily allows users to buy and sell goods with each other (most similar to Craigslist and Amazon). All users have their own respective inventory to post/sell items with a unique title/ProductID with price and quantity attributes for prospective buyers to view and place orders. Placing an order will automatically pull/update the item(s) from the seller's inventory and move it to two separate lists: (1) order list in the buyer's account and (2) sold list in the seller's account. More details will be provided as the application is near release.

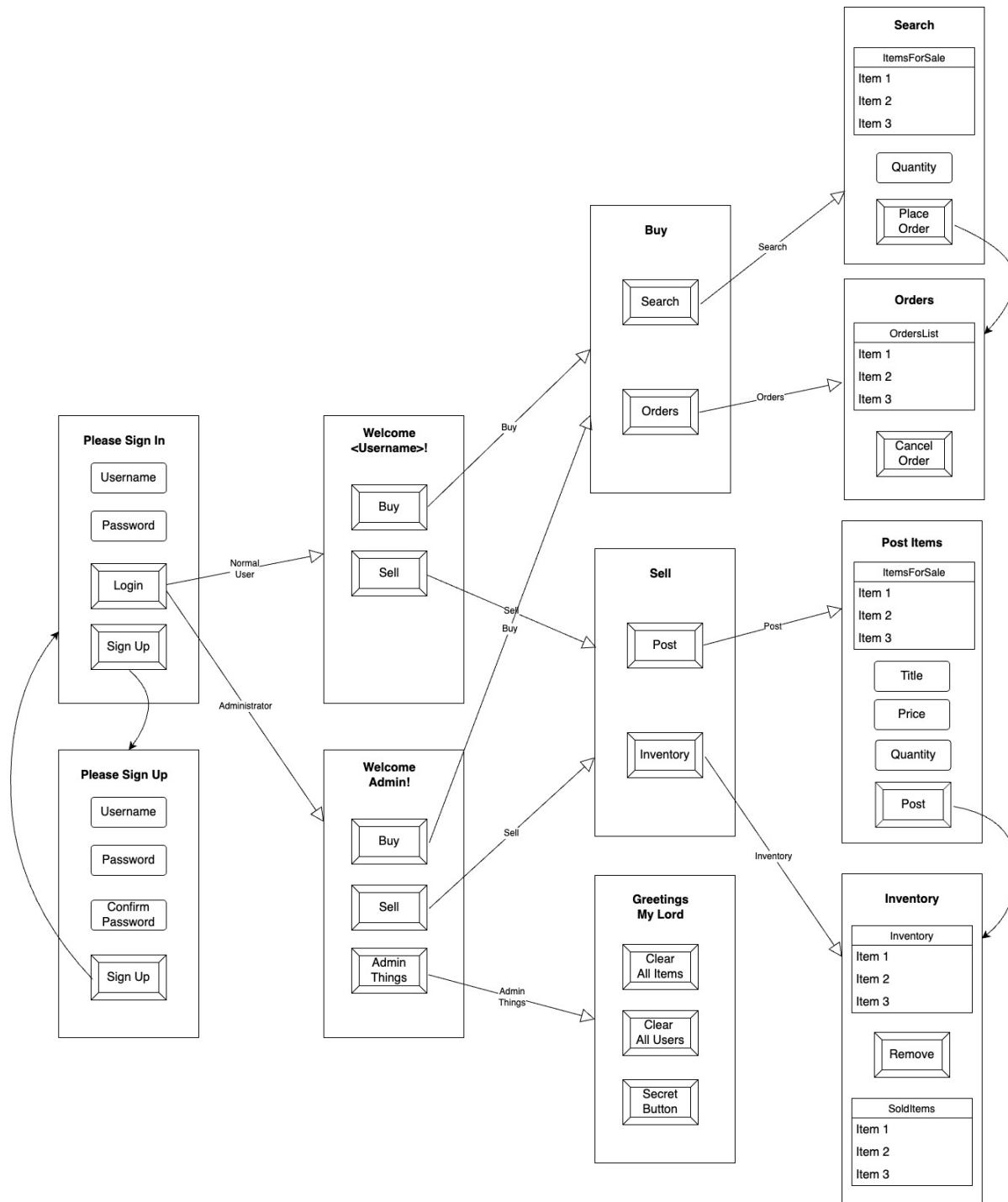
Github: <https://github.com/hapopal/Project2>

Table of contents

Initial Layout	2
Use Case Model	3
Entity Relationship Diagram (ERD)	4
Use Case 01: Predefined Users	5
Use Case 02: Persistence	6
Use Case 03: Create Account	7
Use Case 04: Delete Account	8
Use Case 05: Bad Username from Creating Account	9
Use Case 06: Password Mismatch from Creating Account	10
Use Case 07: Invalid Username Login	11
Use Case 08: Modify Inventory as Seller	12
Use Case 09: Cancel Order as Buyer	13
Use Case 10: Place Same Order that was Cancelled	14
Use Case 11: Clear All Items	15
Use Case 12: Clear all Non-Admin Accounts	16

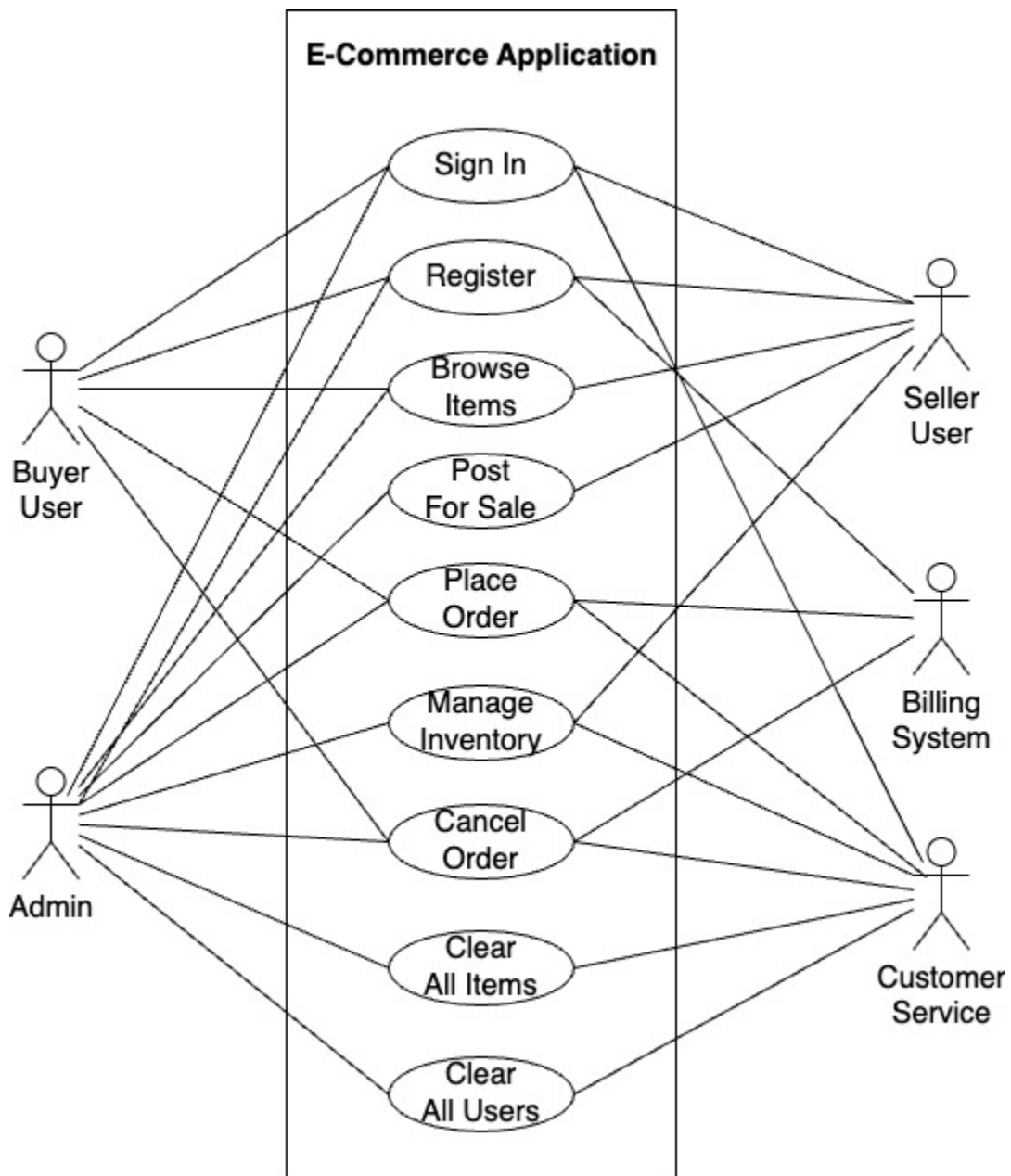
Initial Layout

Include a layout similar to the one shown below. This can be created using screenshots from Android Studio, using [Draw.io](https://draw.io), or even sketched out on paper (or a tablet if you are fortunate enough to have one).



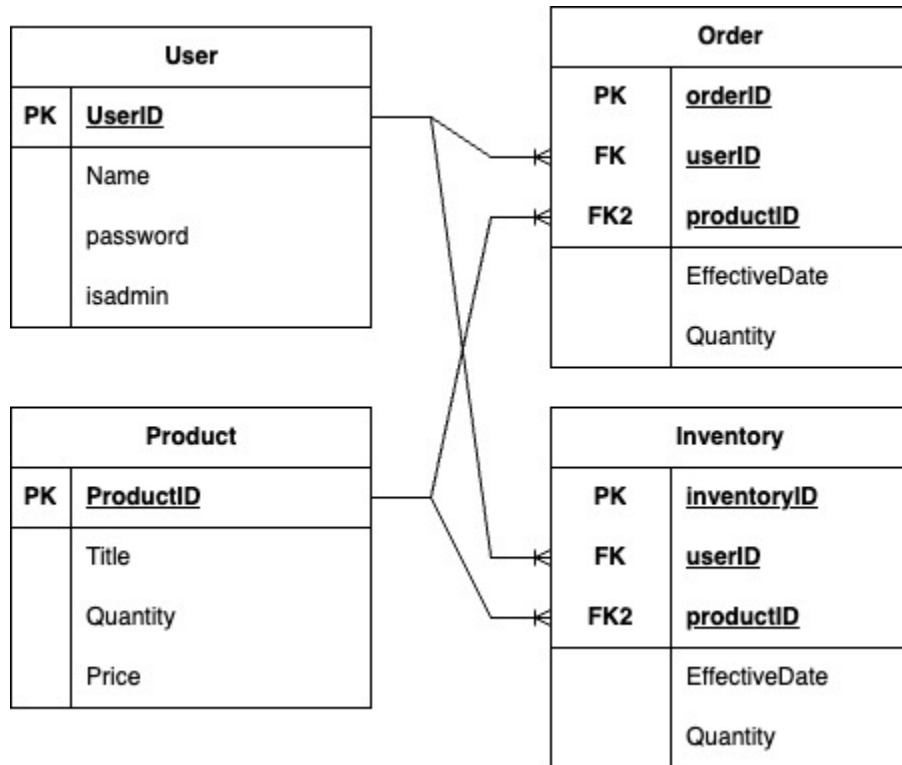
Use Case Model

Insert your actor diagrams here. It should show the users and their roles and the use cases they complete. This does not need to be 100% comprehensive but I'd like to see at least 2 actors with three actions each.



Entity Relationship Diagram (ERD)

The assignment MUST use a database and must have at least three tables



Use Case 01: Predefined Users

1. Force quit the application¹
2. Login as testuser1
3. Display the username 'testuser1'
4. Logout
5. Login as admin2
6. Display the username 'admin2'
7. Clear all items
8. Clear all non-admin users

This use case passes if all of these conditions are met. It fails otherwise.

¹ How to force quit an application in Android:

<https://www.digitaltrends.com/mobile/how-to-force-close-apps-android/>

Use Case 02: Persistence

1. Add an item to the database
2. Force quit the application²
3. Show the item added in step 1 is still in the database
4. Change an item in the database
5. Force quit the application
6. Show the item modifications from step 4 have been saved

² How to force quit an application in Android:

<https://www.digitaltrends.com/mobile/how-to-force-close-apps-android/>

Use Case 03: Create Account

1. User opens app and clicks the "Create Account" button
2. App displays the create account screen
3. User fills user information (username, password)
4. User clicks "Create Account" button
5. User is returned to the landing page
6. A Toast message is displayed indicating the successful creation of an account
7. Force quit application
8. Login via new username
9. App displays welcome user page
10. Use case ends.

Use Case 04: Delete Account

1. Open app to landing page and click Login
2. User enters admin2 credentials and clicks sign in
3. App displays admin page for "admin2"
4. User clicks "Manage Accounts" to view Account page
5. App displays all accounts (except admin)
6. User selects/clicks account to delete
7. App displays selected account's information with "Confirm Delete Account" button
8. User reviews account information and clicks "Confirm Delete Account"
9. A Toast message is displayed indicating the successful deletion of an account
10. Logout of admin2
11. Force quit application
12. User attempts to login with deleted account credentials
13. App displays invalid account message
14. Use case ends

Use Case 05: Bad Username from Creating Account

Starting at 3 from Use Case 03 {Create Account}

3. User enters a username that is already in use
4. App displays message "Username already in use, try a different one"
5. User enters the same username again
6. App displays message "Timeout buddy!" and returns to landing page
7. App displays message "Too many bad attempts"
8. Use case ends

Use Case 06: Password Mismatch from Creating Account

Starting at 3 from Use Case 03 {Create Account}

3. User enters a valid username, password, and nonmatching confirm password
4. App displays message "Passwords do not match, try again"
5. User enters a valid username, password, and nonmatching confirm password again
6. App displays message "C'mon buddy, match your passwords"
7. User enters a valid username, password, and nonmatching confirm password again again
8. App displays "Timeout buddy!" message and returns to landing page
9. A Toast message is displayed "Try creating an account again"
10. Use case ends

Use Case 07: Invalid Username Login

1. Open App and click Login
2. User enters credentials with nonexistent username to login
3. App displays message "This username doesn't exist, try again"
4. User again enters the same credentials with nonexistent username
5. App displays message "C'mon buddy, maybe create an account" and returns to landing page
6. A Toast message is displayed "Either create an account or login as a different user"
7. Use case ends

Use Case 08: Modify Inventory as Seller

1. Open app to landing page and click "Login"
2. User enters testuser1 credentials and clicks "Sign In"
3. App displays Welcome User page for "testuser1"
4. User selects/clicks "Sell" button
5. App displays Sell page
6. User selects/clicks "Inventory" button
7. App displays Inventory page
8. User selects items to remove, enter quantity, and clicks "Remove"
9. Toast message is displayed "<#> of item <XXX> was removed from Inventory"
10. User returns to Sell page and clicks "Post" button
11. App displays Post Items page
12. User notes difference in ItemsForSale list displayed
13. Use case ends

Use Case 09: Cancel Order as Buyer

Starting at 4 from Use Case 08 {Modify Inventory as Seller}

4. User selects/clicks “Buy” button
5. App displays Buy page
6. User selects/clicks “Orders” button
7. Apps displays Orders page
8. User selects order to cancel and clicks “Cancel Order”
9. Toast message is displayed “Order <#> of item <XXX> is canceled”
10. User returns to Buy page and clicks “Search” button
11. Apps displays Search page
12. User notes difference in ItemsForSale list displayed
13. Use case ends

Use Case 10: Place Same Order that was Cancelled

Starting at 13 from Use Case 09 {Cancel Order as Buyer}

13. User selects same item, enters same quantity, and clicks "Place Order"
14. Toast message is displayed "Item <#> is ordered, your Order Number is <#>"
15. App directs user to Order page
16. User confirms order is correct on OrdersList with new unique Order Number
17. Use case ends

Use Case 11: Clear All Items

1. Open app to landing page and click "Login"
2. User enters admin2 credentials and clicks "Sign In"
3. App displays Admin page for "admin2"
4. User clicks "Admin Stuff" button
5. App displays Admin Privileges page
6. User clicks "Delete All Items"
7. App displays message "Are you sure you want to delete all items?"
8. User selects/confirm/clicks "Yes" button
9. A Toast message is displayed "All items have successfully been deleted"
10. User logs out as admin2
11. Force quit application
12. Open app to landing page and click Login
13. User enters testuser1 credentials and clicks "Sign In"
14. User clicks "Buy" button
15. App displays Buy page
16. User notes the lack of items for sale
17. Use case ends

Use Case 12: Clear all Non-Admin Accounts

Starting at 6 from Use Case 11 {Clear All Items}

6. User clicks "Delete All Accounts"
7. App displays message "Are you sure you want to delete all non-admin accounts?"
8. User selects/confirm/clicks "Yes" button
9. A Toast message is displayed "All non-admin accounts have successfully been deleted"
10. User logs out as admin2
11. Force quit application
12. Open app to landing page and click Login
13. User enters testuser1 credentials and clicks "Sign In"
14. App displays message "This username doesn't exist, try again"
15. User enters admin2 credentials and clicks "Sign In"
16. App displays Admin page for "admin2"
17. User clicks "Buy" button
18. App displays Buy page
19. User notes the lack of non-admin items for sale
20. Use case ends