Shared Shopping Portal
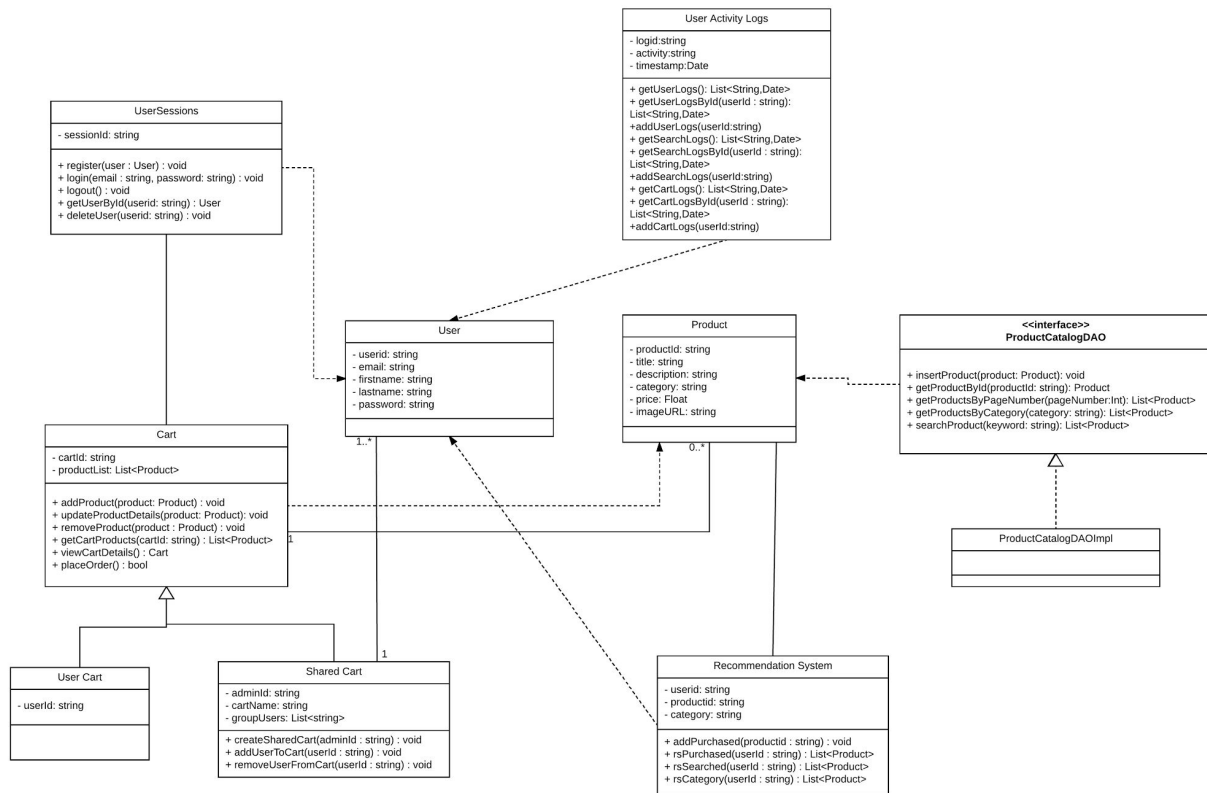
Team Starks
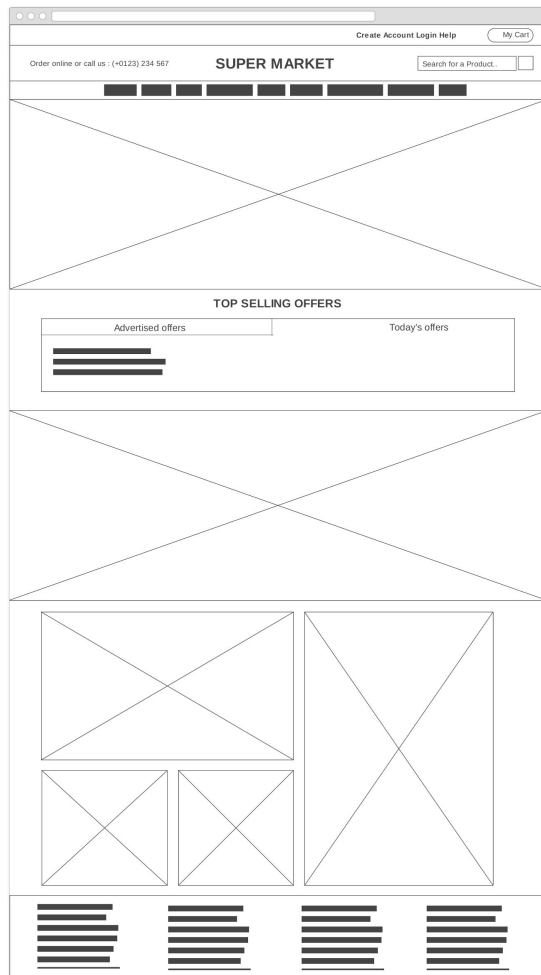## CMPE281

Anuj Chaudhari
Purvesh Kothari
Isha Jain
Ritika Chadha
Shantagouda Kuntoji

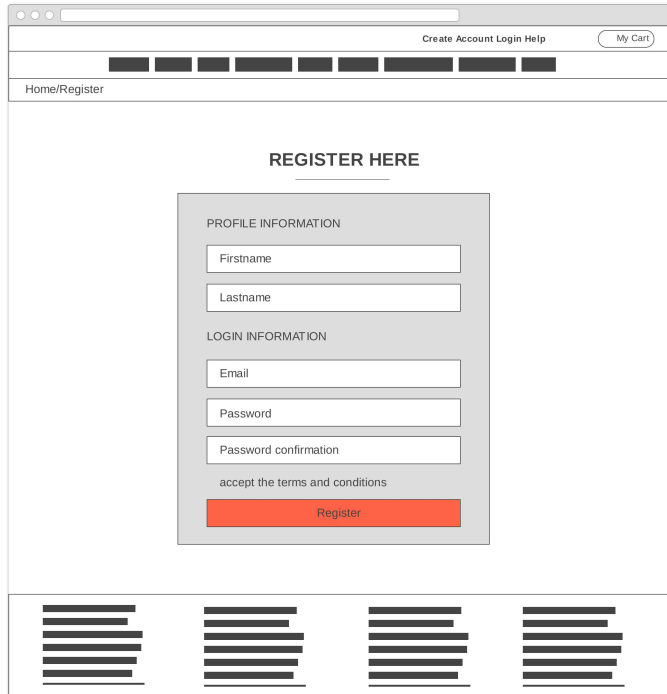# Class Diagram

UserSessions

- sessionId: string

+ register(user : User) : void
+ login(email : string, password: string) : void
+ logout() : void
+ getUserById(userid: string) : User
+ deleteUser(userid: string) : void

User Activity Logs

- logid:string
- activity:string
- timestamp:Date

+ getUserLogs(): List<String,Date>
+ getUserLogsById(userId : string): List<String,Date>
+addUserLogs(userId:string)
+ getSearchLogs(): List<String,Date>
+ getSearchLogsById(userId : string): List<String,Date>
+addSearchLogs(userId:string)
+ getCartLogs(): List<String,Date>
+ getCartLogsById(userId : string): List<String,Date>
+addCartLogs(userId:string)

User

- userid: string
- email: string
- firstname: string
- lastname: string
- password: string

1..*

Product

- productId: string
- title: string
- description: string
- category: string
- price: Float
- imageURL: string

0..*

<<interface>>
ProductCatalogDAO

+ insertProduct(product: Product): void
+ getProductById(productId: string): Product
+ getProductsByPageNumber(pageNumber:Int): List<Product>
+ getProductsByCategory(category: string): List<Product>
+ searchProduct(keyword: string): List<Product>

ProductCatalogDAOImpl

Cart

- cartId: string
- productList: List<Product>

+ addProduct(product: Product) : void
+ updateProductDetails(product: Product): void
+ removeProduct(product : Product) : void
+ getCartProducts(cartId: string) : List<Product>
+ viewCartDetails() : Cart
+ placeOrder() : bool

1

User Cart

- userId: string

Shared Cart

- adminId: string
- cartName: string
- groupUsers: List<string>

+ createSharedCart(adminId : string) : void
+ addUserToCart(userId : string) : void
+ removeUserFromCart(userId : string) : void

1

Recommendation System

- userid: string
- productid: string
- category: string

+ addPurchased(productid : string) : void
+ rsPurchased(userId : string) : List<Product>
+ rsSearched(userId : string) : List<Product>
+ rsCategory(userId : string) : List<Product>

Wireframes

Homepage

Registration and Login

Products, Recommendations and Activity logs
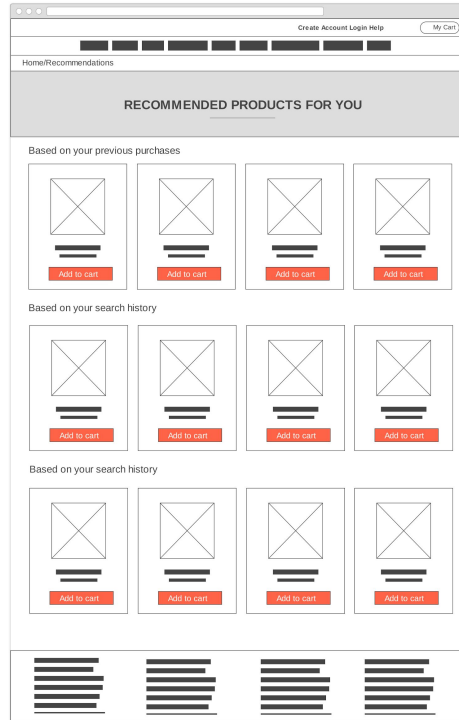
Shopping cart

Activity Diagram

Browse

Browse products

Search

Search product

Not found

Found

View item

Add to personal cart/ shared cart

View cart

Update cart

Checkout

Check Shopping cart
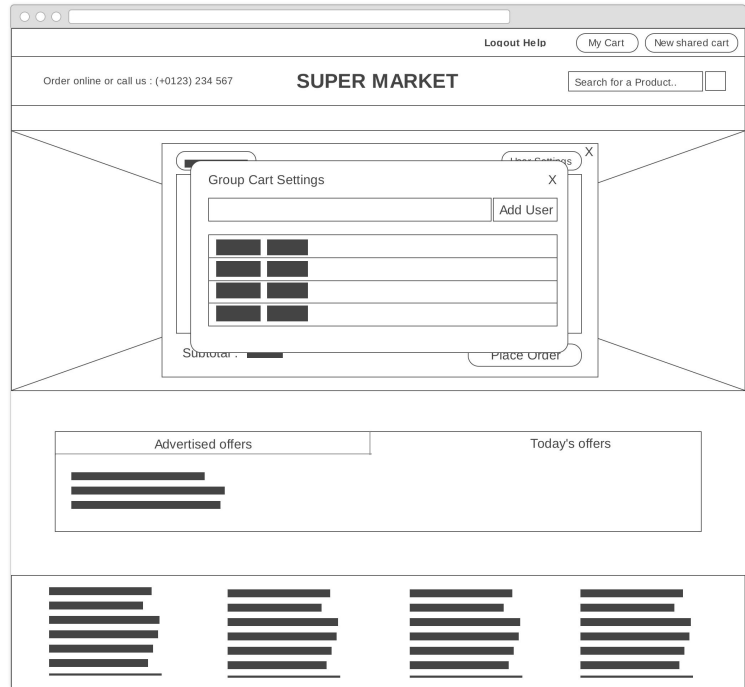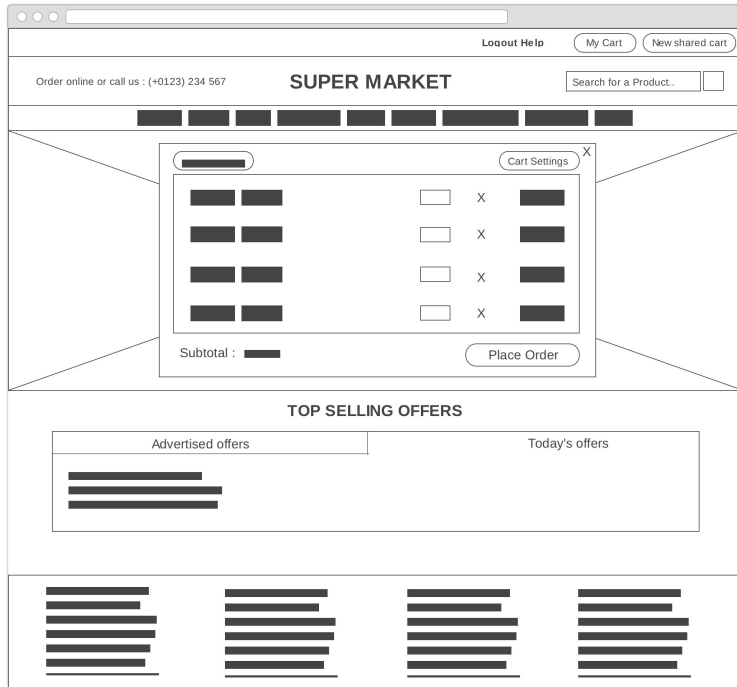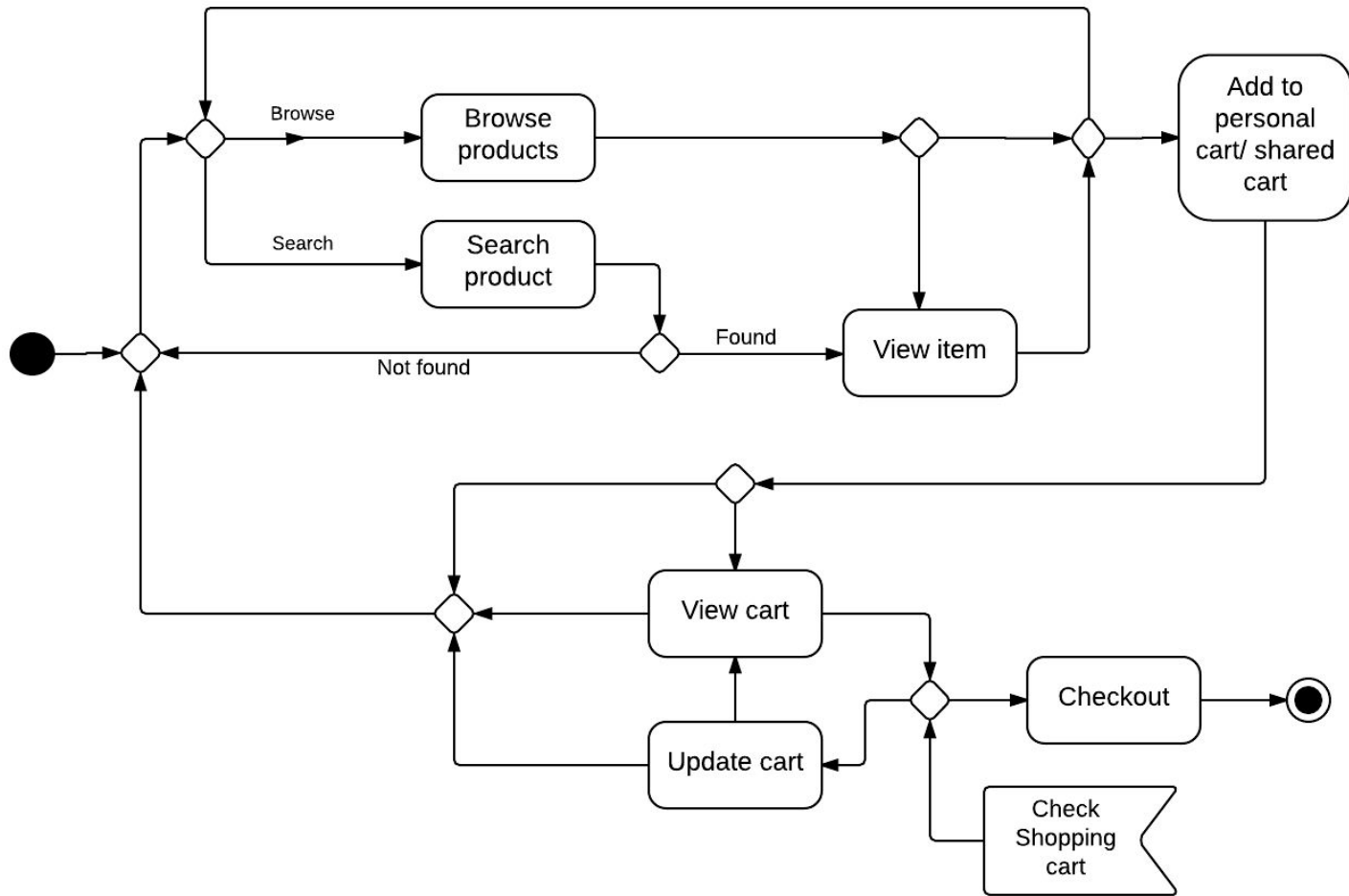
Sequence Diagram

Online Shopping System Sequence Diagram

Actors and objects:
- User
- Online Shopping System
- Customer
- Product Catalog
- Recommendation System
- Activity Logs
- Shopping cart
- Databases
  - User login DB
  - Product Catalog DB
  - User Activity DB
  - Shopping Cart DB
  - Recommendation DB

Messages:
- Register()
- RegisterUser(emailid,firstname,lastname, password)
- Login()
- Verify()
- Validate_Account(emailid,password)
- ReturnLoginStatus(true/false)
- LoginStatus(false)
- GetProducts()
- SearchProductCatalogDb(productid/pagenumber/category)
- ReturnProducts(product)
- GetUserDetails()
- GetRecmmendedProducts(userid)
- ShowRecommendedProducts(products)
- RecommendProducts
- Displayproducts()
- CreateSharedCart(adminid,cartName,groupmembers)
- PlaceOrder()
- AddUsersToCart(userid)
- OrderPlaced(orderid)
- ViewCartDetails()
- UpdateActivityLogs()
- StoreActivityDetails(userid)
- GetActivityLogs(userid)
- GetActivityLogs(Userid)
- Message
- ShowActivityLogs()
- ReturnActivityLogs()
- Logout()

Use Case Specification

LOGIN USE CASE SPECS

Use Case 1. Register to Super Market

Description: Registration to the application in order to place orders

Primary Actors: Customer
Compatible Web Browser
Super Market application
Preconditions: Active Internet Connection
Connection to the Web Application Servers
Connection to the User Sessions Servers
Connection to the User Sessions Database Servers
Basic Flow of events:
1. Supermarket customer enters the required URL into the compatible browser.
2. Customer clicks on the tab 'Create Account' for the registration process to initiate.
3. Customer enters her/his details in the form.
4. Customer needs to enter a valid 'First Name' and 'Last Name'.
5. Customer has the option of subscribing to the newsletter which she/he can opt to choose.
6. Customer enters a valid Email address.
7. Customer enters a password in the 'Password' and 'Password Confirmation' fields which should match.
8. Customer then checks the box to accept all the terms and conditions.
9. Customer then clicks on the 'Register' button to sign up.
Alternate flows:
1. The user does not enter one of the fields.
   -> The application throws an error after Register is clicked stating 'There was a problem adding the information to the database'.
2. Customer does not enter matching passwords.
   -> The application throws a validation error

Use Case 2. Login to Super Market

Brief Description: Login to the application to view your profile

Primary Actors: Customer
Compatible Web Browser
Super Market application

Preconditions: Active Internet Connection
Connection to the Web Application Servers
Connection to the User Sessions Servers
Connection to the User Sessions Database Servers

Basic Flow of events:
1. Customer enters required fields to sign up.
2. Customer enters email and password to login successfully.
3. Customer then clicks on the 'Login' button.
4. Customer is validated.
5. Customer is directed to his/her dashboard.

Alternate flows:
1. Customer enters incorrect email address.
    -> Customer is thrown and error message stating 'There was a problem finding the user'.
2. Customer enters incorrect password.
    -> Customer is thrown a 401 HTTP error.

PRODUCT CATALOG SPECS

Use Case 3. Search in the Super Market

Description: Searches for a particular product

Primary Actors: Customer
                     Compatible Web Browser
                     Super Market application

Preconditions: Active Internet Connection
                     Connection to the Web Application Servers
                     Connection to the Product Catalog Servers
                     Connection to the Product Catalog Database Servers

Basic Flow of events:
1. Customer enters the keyword to be searched for in the Search Bar on the right and clicks on the magnifier for the application to display the search results.
2. Customer is directed to the Search Results page.
3. This page provides the customer with a list of products in the order of the most relevant on the basis of the search keyword entered by the customer.

Alternate flows:
1. Customer enters no keyword and clicks on the magnifier.
   -> Search textbox requires a keyword to be entered for the search to be performed. No action takes place.
2. Customer enters an invalid search keyword.
   -> The search results page shows no valid output for the customer.

Use Case 4. Views Items in the Catalog

Description: A display of items on the web page for the customer in the form of a page by page entries

Primary Actors: Customer
                     Compatible Web Browser
                     Super Market application

Preconditions: Active Internet Connection
                    Connection to the Web Application Servers
                    Connection to the Product Catalog Servers
                    Connection to the Product Catalog Database Servers

Basic Flow of events:
1. Customer can opt to choose from the different categories of products available from the navigation bar.
2. Customer clicks on one of the categories and is further provided with another set of options to pick from.
3. Customer clicks on one of the options.
4. Customer is directed to another page with the items in that particular category picked from the Nav bar.
5. Customer picks from the drop down the page number in order to view the items on that page.

Alternate flows:

1. Customer is not connected to the Product Catalog Servers.
   -> The web page renders the same set of items for different page numbers as well.
2. Customer is not connected to the Product Catalog Database Servers.
   -> The web page renders the same set of items for different page numbers as well.

SHOPPING CART SPECS

Use Case 5. Place order through a User Cart

Description: To place an order of an item in the existing user cart of the customer

Primary Actors: Customer
                  Compatible Web Browser
                  Super Market application

Preconditions: Active Internet Connection
                  Connection to the Web Application Servers
                  Connection to the Shopping Cart Servers
                  Connection to the Shopping Cart Database Servers

Basic Flow of events:
1. Click on one of the categories present on the home page of the Super Market Application.
2. Choose a particular type of items to choose from.
3. Click on the item and click on 'Add to Cart' button placed below the item.
4. Click on the 'My Cart' icon on the header of the web page.
5. The customer can view the items placed in the cart.
6. Click on 'Place Order' to place the order for the products in the user cart.
7.

Alternate flows:
1. Customer does not have any item in the cart.
   ->Customer cannot place an order as the cart is empty and the button for placing an order is disabled.

Use Case 6. Place order through a Shared Cart

Description: To place an order of an item in the existing shared cart of the customer

Primary Actors: Customer
                Compatible Web Browser
                Super Market application

Preconditions: Active Internet Connection
                Connection to the Web Application Servers
                Connection to the Shopping Cart Servers
                Connection to the Shopping Cart Database Servers

Basic Flow of events:
1. Customer clicks on the 'New Shared Cart' button on the top of the page.
2. Customer then enters the name of the cart.
3. Customer searches for the product to be added into the cart.
4. User clicks on Add to Cart button.
5. User can view his cart on the Cart Settings dialog box.
6. User selects the cart among the various user and shared carts.
7. User then clicks on 'Place Order' place an order.

Alternate flows:
1. Customer does not have any item in the cart.
      -> Customer cannot place an order as the cart is empty and the button for placing an order is disabled.
2. Customer is not an admin and wants to place an order.
      -> Customer cannot place an order as the cart is empty and the button for placing an order is disabled.

RECOMMENDATION SPECS

Use Case 7. View Search-based Recommendations

Description: To view recommendations suited to the customer based on the search history logs.

Primary Actors: Customer
Compatible Web Browser
Super Market application

Preconditions: Active Internet Connection
Connection to the Web Application Servers
Connection to the Recommendations Servers
Connection to the Recommendations Database Servers

Basic Flow of events:
1.   Click on the Recommendations tab in order to view the recommendations.
2.   The user can view the recommendations of all the products based on search history.

Alternate flows:
1.   Customer does not search for any products.
->No recommendations would be shown on the basis of searched products.
2. Customer is not connected to the Recommendations Server
->No recommendations would be shown.

Use Case 8. View Purchase-based Recommendations

Description: To view recommendations suited to the customer based on purchases

Primary Actors: Customer
  Compatible Web Browser
  Super Market application

Preconditions: Active Internet Connection
  Connection to the Web Application Servers
  Connection to the Recommendations Servers
  Connection to the Recommendations Database Servers

Basic Flow of events:
1. Click on the Recommendations tab in order to view the recommendations or click on the recommended pro
2. The user can view the recommendations of all the products based on the previous purchases and purchased category.

Alternate flows:
1. Customer does not purchase any products.
   -> No recommendations would be shown on the basis of products.
2. Customer is not connected to the Recommendations Server
   -> No recommendations would be shown.

ACTIVITY LOGS SPECS

Use Case 9. View Search Logs

Description: To view user logs based on the search activity.

Primary Actors: Customer
                      Compatible Web Browser
                      Super Market application

Preconditions: Active Internet Connection
                      Connection to the Web Application Servers
                      Connection to the User Activity Logs Servers
                      Connection to the User Activity Logs Database Servers

Basic Flow of events:
1. Customer goes to the Search Bar and enters the keyword to be searched.
2. Customer clicks on the magnifier.
3. Customer can view the item he/she searched for previously.
4. Customer then clicks on the My Activity tab on the top of the page.
5. Customer selects the 'Search Logs' to be viewed.

Alternate Flows:
1. Customer has not searched for any product.
    -> Search Logs tab would show no keywords that were searched for.
2. Customer is not connected to the Activity Logs Server
    -> No logs would be shown.
3. Customer has not signed in.
    -> Error in showing logs.

Use Case 10. View Cart Logs

Description: To view user logs based on the cart activities.

Primary Actors: Customer
　　　　　　　　Compatible Web Browser
　　　　　　　　Super Market application

Preconditions: Active Internet Connection
　　　　　　　　Connection to the Web Application Servers
　　　　　　　　Connection to the User Activity Logs Servers
　　　　　　　　Connection to the User Activity Logs Database Servers

Basic Flow of events:
1. Customer goes to the Shared Cart.
2. Customer deletes an item from the list of products in his/her cart.
3. Customer then clicks on the My Activity tab on the top of the page.
4. Customer selects the 'Cart Logs' to be viewed.
5. Customer can see an entry with a timestamp stating that a product was deleted from the cart.

Alternate Flows:
1. Customer has not added any product.
   -> Cart Logs tab would show no such log in the cart logs.
2. Customer has not created a cart.
   -> No logs related to that particular log would be shown.
3. Customer has not signed in.
   -> Error in showing logs.

User Story

User Story: Modification of the shopping cart

In order to let multiple users update a shopping cart
As an **ecommerce website user**
I want to grant the access of my shopping cart to other users of the system

Scenario#1: Another user should be able to add product to my shopping cart.
Given I am logged into my account
And I have created a shared shopping cart
When I enter the email address of the other user who has already an account
to share my shopping cart
Then he should be able to add his choice of products to my shopping cart

Scenario#2: Authorized shopping cart user should be able to remove the products from my shopping cart.
Given the authorized user is logged into his account
And the authorized user has already added one or more products in the shared shopping cart
And I, the cart owner hasn't checked out the order yet
When the authorized user clicks on the button to remove the item
Then that particular item should be removed from my shared shopping cart

Scenario#3: Only the authorized user should be able to remove/update items and its quantities in the shared shopping cart.
Given the authorized user has added one or more products in the shared shopping cart
And I as the cart owner am able to see the updated shopping cart
When I try to remove or modify the items or quantity added by the authorized user
Then a message should be displayed mentioning that this operation is prohibited

Scenario#4: User without creating account should not be able to access the shared cart
Given I am logged into my account
And I have created a shared shopping cart
When I enter the email address with which no account has been created in the system
Then a message should be displayed mentioning that a user with given id doesn't exist

Scenario#5: Modification to the shared shopping cart is logged
Given I add or remove the items from the shopping cart
And authorized user add or remove the item from the shopping cart
When site admin chooses the activity logs section page
Then all the activities related to the cart is shown to the admin

Scenario#6: Modification of the shared shopping cart is restricted after checkout
Given an authorized user has added the items in the cart
And I have checked out the cart to complete the order
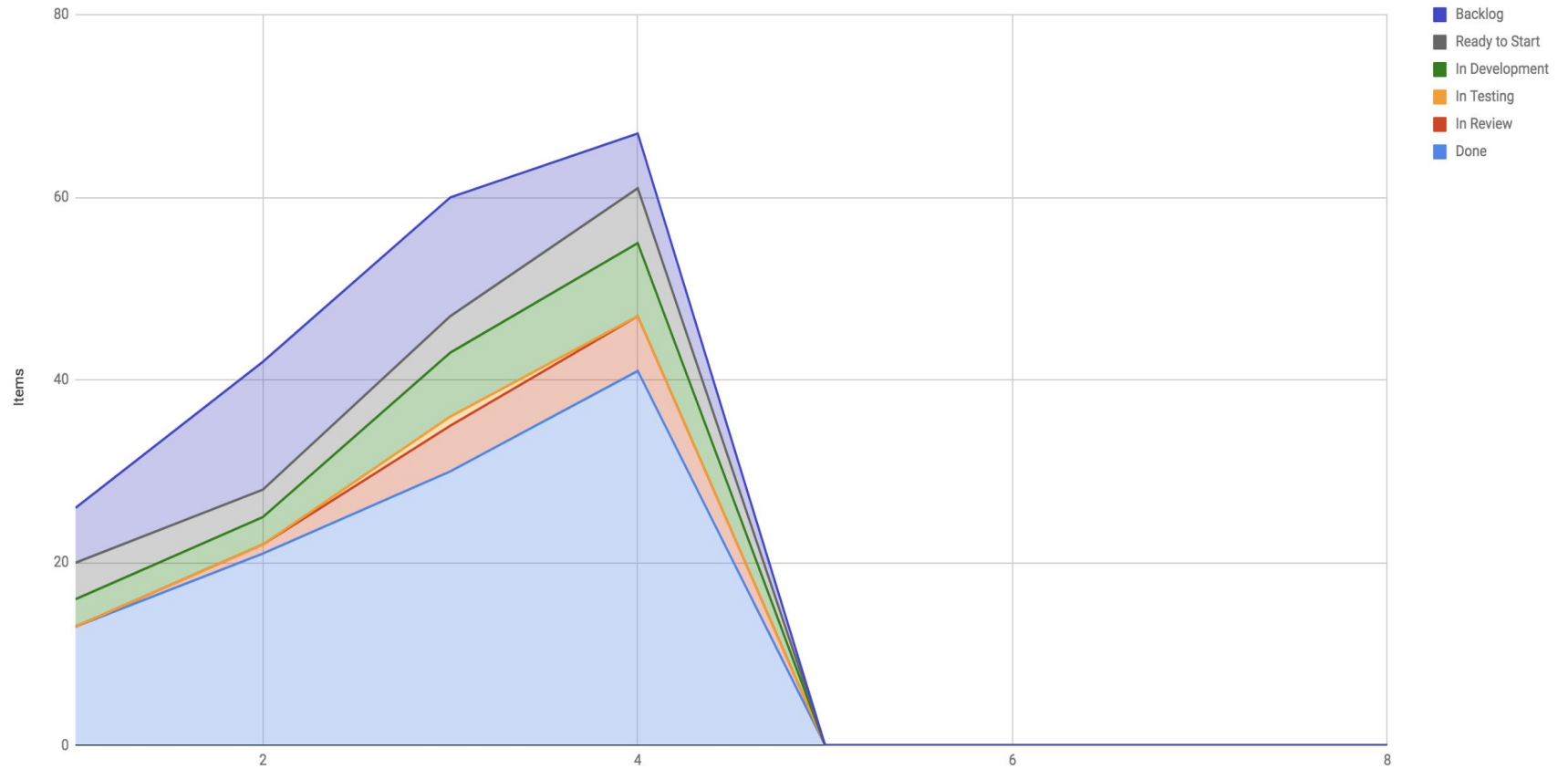**When** authorized user tries update the order or quantity
Then a message should be displayed mentioning that order is already processed
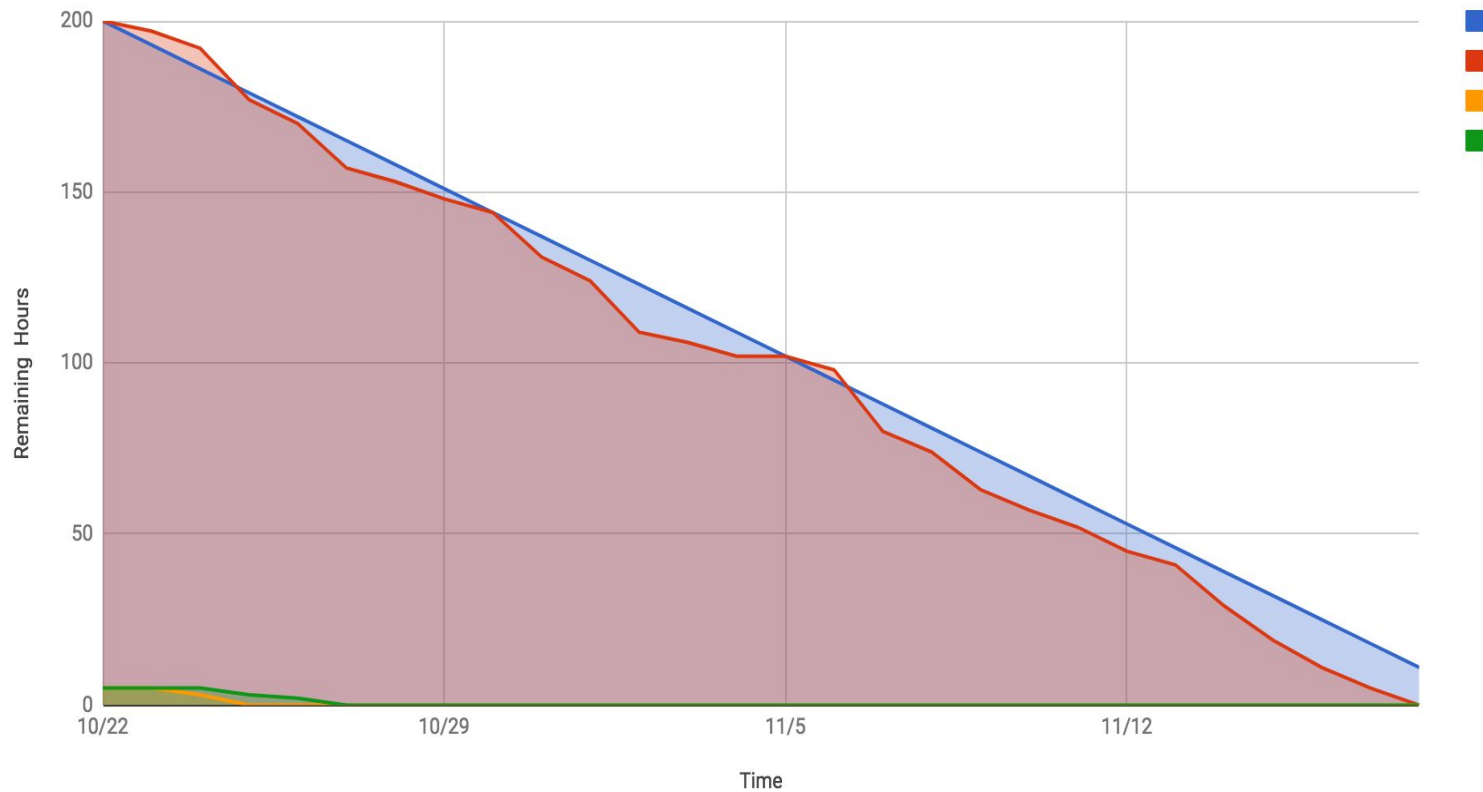
Final Task Board

- https://github.com/nguyensjsu/cmpe281-starks/projects/1

Cumulative Flow Diagram

Cumulative Flow Diagram

Legend:
- Backlog
- Ready to Start
- In Development
- In Testing
- In Review
- Done

Scrum Burndown Chart

Scrum Burndown Chart

Team's Project Retrospective

What went well?

- Due to proper planning and management our project got completed on time.
- Because of the appropriate documentation, integration of all the modules in the project was smooth.
- Frequent and timely commits helped to keep the project bug free.
- Team meetings were replete with new ideas and views.
- Team had figured out functional workflow well in advance, this helped to work well with each other.

What didn't go well?

- Insufficient research led to some last minute changes.
- Because of the last minute changes, a considerable waste was generated.
- Database clustering and partition tolerance was reserved for last week of project. We realised that this could have been done bit earlier.
- Long time was spent in understanding of the requirement for the partition tolerance of the database resulting in hurried implementation.

What have we learned?

- To work as a team and divide tasks efficiently.
- To consider feedback from fellow teammates in a positive light.
- To respect team members and listen to new ideas and views.
- To be clear about what we want to accomplish before we sat down at the computer.
- It is important to explore all the possible use cases of the system before implementing the functionalities.

Thank You