CRC cards

Master (entity class)

Responsibilities:

- Stores all users corresponding to the username of the user
- Stores all products corresponding to the product ID
- Stores the list of products corresponding to the tag word that is associated with that list of products.

Collaborators:

User
Product

User (entity class)

Responsibilities:

- Attributes of user: username, listPosts, shoppingCart, currentSearches, listfollowing, feed, isVerified, password (to be implemented later)
- constructor (sets username to username, password to password - implementation later and everything else to empty arraylists)
- constructor
- Getter and setter for username, listPosts, shoppingCart, currentSearches, listfollowing, feed, isVerified

Collaborators:

- Post
- Product

Product (entity class)

Responsibilities:

- Constructor
    - Create the product
    - Holds information about the product
- instance attributes: name, price (float that will be added to the price of other products in the cart of the user), category, size(None, or list of sizes)
- Attribute: basically the description of the product
- See which user leaves how many stars, associated with the comment they leave for the product.
- Getters and setters
    - Price, dimensions, colours etc.

Collaborators:

None

Post (entity class)

Responsibilities:

- Constructor
    - Create the post
-
- instance attributes: product: type Product, tag words for product,, userMadePost: User, description
- getPost_Topic: return the product that it represents
- getTags return ArrayList<String> of tags for the post
- getDescription return the description of the post

Collaborators:

- Product
- User

masterManager (use case)

Responsibilities:

- Get a user from the master from a given username
- Accesses users when the user inputs correct username
- Save the username of a user as keys and the profile of the user as values
- Product getter based on product ID and product list getter based on tag word.
- User dictionary setter(add users to master)
- Product dictionary setter(add products to master)
- Product tag map which maps tags to the products.
- Product tag map getter to get the list of products associated with a tag.
- Add product to the Master product data base and add the product to dictionary associated with the tag.
-

Collaborators:

- Master

userManager(use case)

Responsibilities:

- (public) Methods:
    - createUser (creates a new user)
    - changeUsername (changes Username of the user)
    - addToShoppingCartSearches (adds a product to the user’s shopping cart based on index of Product in user’s current Searches list)
    - addToShoppingCartFeed (adds a product to user’s shopping cart associated with post at index entered which is index of post in the user’s feed)
    - emptyShoppingCart (removes all products from a user’s shopping cart)
    - addToPostList (adds a new post to the user’s list of posts)
    - addFollowing (adds a new following (type User) to another User’s list of following)
    - verifyUser (changes user to being verified, isVerified = true)
    - getFeedList (gets feed of the user)

Collaborators:

- User (entity class)
- masterManager (use case)
- Post

postManager(use case)

Responsibilities:

- CreatePost method to take information from the user and create a product using product Manager
- Adds the post to the post list for user via user manager
- Adds product to the tag map using the given tag via the master manager

Collaborators:

- productManager
- Post
- Usermanager
- masterManager

productManager(use case)

Responsibilities:

- Create product directly from the information and adds in to the master via the masterManager
- Increase the quantity of a product
- Decrease quantity of product
- Check if the product ID is in in the database

Collaborators:

- Product
- MasterManager

browseController (Controller)

Responsibilities:

- Search the feed of the user in searchFeed
- Output the feed to the user by accessing this user’ followers’ selling items via the user manager
- Calls cart controller to prompt if they wish to buy anything in their feed

Collaborators:

- userManager
- cartController

cartController (controller)

Responsibilities:

- addToCartSearch: Takes in user input to add desired product to the user’s cart from the search option
- addToCartFeed: Takes in user input to add desired product to the user’s cart from the user feed.

Collaborators:

- userManager

createPostController (controller)

Responsibilities:

- postCreator, takes in the user and gets input from that user about the options for the post that they want to create.
- Relays this information to the postManager via createPost method from postManager

Collaborators:

- postManager

createUserController (controller)

Responsibilities:

- Prompt the user to input their new username which is called userSignUp
- userCreater takes input from the user and creates an new User entity via UserManager

Collaborators:

- userManager

loginOptionsController (Controller)

Requirements:

- userInput method asks if the user wants to sign in or sign up
- If they select sign in,
  - It prompts the user to sign in via sign in controller
  - Runs OptionController to continue with options to browse, search, etc.
- If they select sign up
  - If directs therm to createUserController to create the User

Collaborators

- createUserController
- SignInController
- OptionController

searchController (controller)

Responsibilities:

- Takes in user input of a tag that they want to search and it searched the master for any product that matches the tag via the masterManager and it adds to this user’s currentSearches attribute via cart Controller

Collaborators:

- masterManager (use case)
- cartController

signInController (controller)

Responsibilities:

- UserSignIn method takes input of the username and checks if it is a valid username via masterManager that has been created already and if it is, it lets you sign in and if it does not, it prompts the user to input valid username

Collaborators:

- masterManager

userOptionsController (controller)

Responsibilities:

- Prompt the user to input search, create post, Browse, or buy from cart
- From this information, it runs methods in specific controllers or user manager to perform what they asked

Collaborators:

- userManager
- Search (controller)
- Browse (controller)
- createPost (controller)

SystemInOut (command line interface)

Responsibilities:

- getInput: gets a users input
- sendOutput: prints a string to the user

Collaborators:

Collaborate with all controllers