P2.2: SeniorList

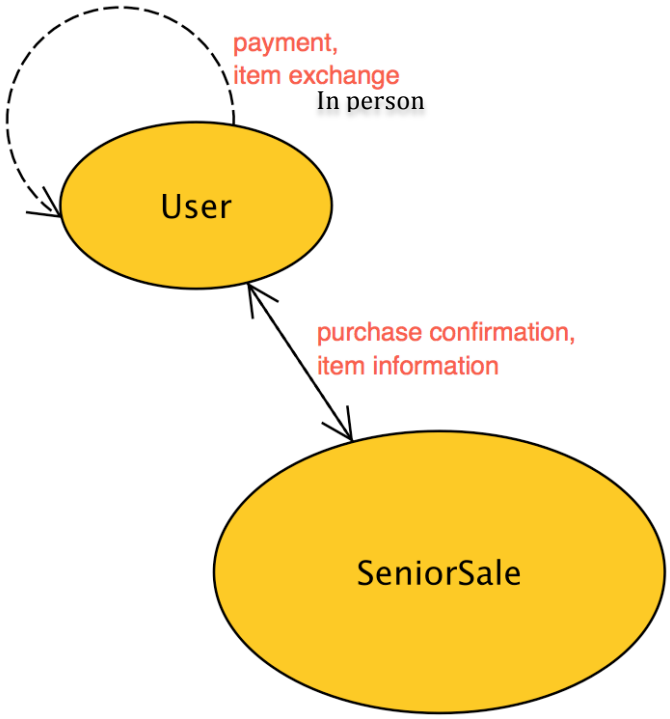
III. Problem Analysis

A. Overview

Purpose and goals:

At the end of the year students at MIT, specifically seniors try to find a way to sell all of the things they do not need or use in their room. The old fashioned way was, send out an email with a list of all the things they are selling, and get 20 emails back. And 10 more come in after the student sends out the "claimed" email. The goal of this application is to create an interface for students with MIT certificates to keep track of the items they are selling, and allow other students to buy them. This will require implementation of a shopping cart for site visitors that claims the items in a cart, until it expires in a few minutes, allowing other students to try and buy the items.

Context diagram



Notes:

- login for all users will eventually be through MIT certificates, but is currently using a username named *Kerberos*. This may allow for non students to purchase in the future
- item info (and eventually images) uploaded by user
- users exchange money and goods in person

Unusual requirements

None.

Key features

Users

- have a Room and a Bag
- Through the Room, can enter information and upload images for each item
- Through the Bag, can purchase items
- pay online to be implemented

Room

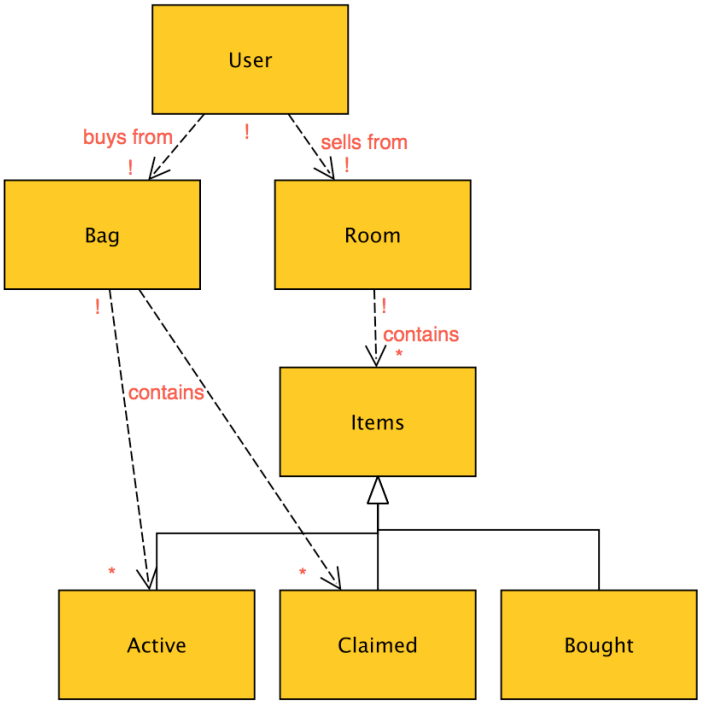
- each user has a room that is created when they register, even if they aren't planning to use the sell functionality of the site
- items are identified by rooms, rather than users

Bag

- each user has a bag that is created when they register, even if they aren't planning to use the buy functionality of the site
- when a user adds an item to his/her bag it becomes 'claimed', and unavailable for other users to see
- bags are saved through sessions, and items are only removed from them if a user removes it, or if the 'claimed' period expires (will be 24 hours, but is not implemented yet)
- when a user checks out his/her bag, the items in its status' are changed to 'bought', and they are moved to the user's purchased items list

B. Domain

Object model



Event model

User::= login (add_item | tag_item | upload_img | destroy_item)* logout
||
add_item (upload_img)* destroy_item
||
[login] (add_to_bag+ (checkout | save_bag)*) [logout]
||
login save_bag logout

Bag::= (add_item delete_item) (checkout | save_bag)

C. Behavior

Feature descriptions

Rooms
Users will add items, add descriptions, and upload images to their online store.

Bags
Users can also browse other items by owner or by tag, add them to their bag, and checkout the bag. Eventually there will be a time-out feature. Adding an item to a

bag will claim it for 24 hours, until which time it is unclaimed and able to be added to another students bag.

Security concerns

Requires an MIT certificate validation, and secure checkout procedures. Attackers should not be able to get information about users payment information. Currently this is implemented with usernames and passwords, completely unaffiliated with MIT Kerberos.

Operations

Events
signup
login/new_session
logout/destroy_session
add_item
edit_item
upload_image
delete_item
add_to_cart
delete_from_cart
checkout

Queries
view_items
(by owner, by tag)
view_item
item_is_claimed
item_is_bought

User interface