

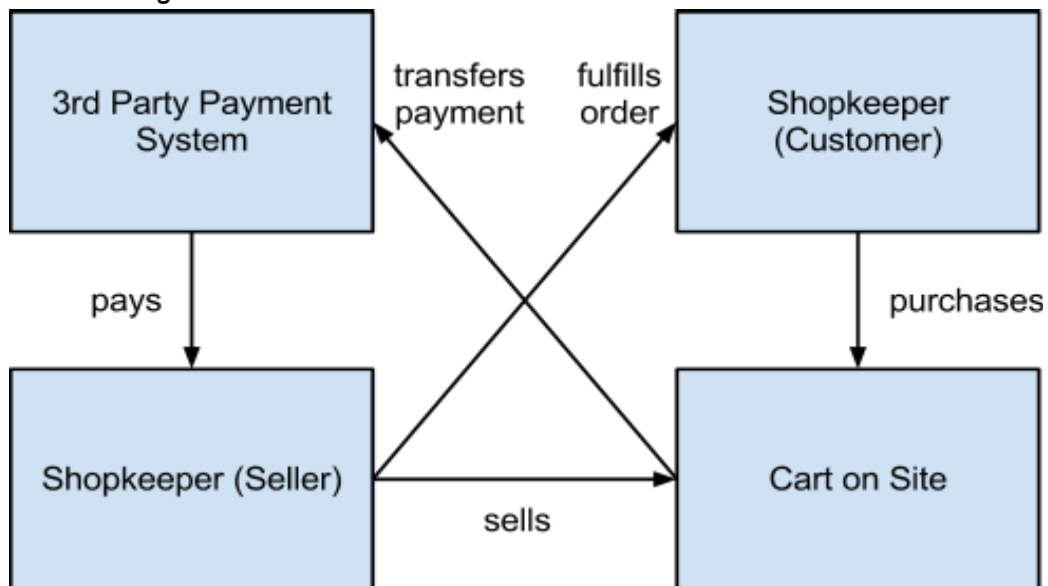
## Problem Analysis PS2

### Overview

#### *Purpose and goals*

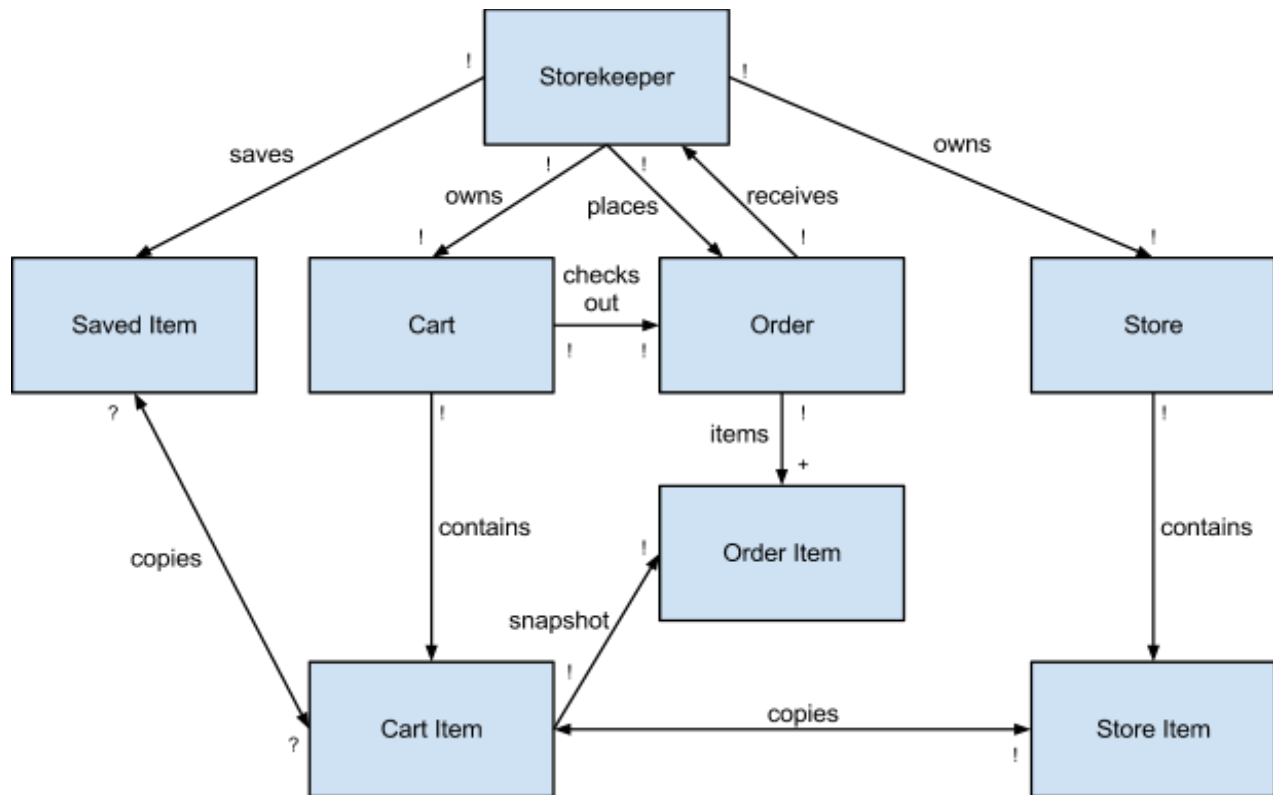
The goal of this product is to create an intuitive interface for customers and sellers to interact. This product will allow customers to easily browse and shop for products from multiple vendors. It will also be easy to shopkeepers to set up and connect with customers. Existing solutions aren't geared towards a community of customers and storekeepers, so this product will lower the barrier separating the two parties.

#### *Context diagram*



### Domain

#### *Object model*



### Event model

Storekeeper Login

Storekeeper Logout

Storekeeper Add Item

Storekeeper Edit Item: edit the price or quantity of an item

Storekeeper View Orders: view both orders placed in your store and placed by you

Storekeeper Fulfill Order: view items of customer order, press fulfill

Storekeeper Add to Cart: add item from particular store

Storekeeper Checkout: can only be placed if items are in cart, and items available

Storekeeper Save Item: save item from cart for later

ShoppingCart ::= (StorekeeperStart) (Shopper)+ (StorekeeperEnd)+

StorekeeperStart ::= Login+ (AddItem EditItem)\* Logout\*

Shopper ::= Login\* (AddtoCart\* SaveItem\* | Checkout)\* Logout\*

StorekeeperEnd ::= Login+ View Orders FulfillOrder Logout\*

### Behavior

#### Feature descriptions

- Shopping cart: shoppers can add items from any store into one cart and proceed to checkout normally
- Saved List: shoppers can save items from their cart for later, and proceed with purchase

- Shopkeeper Homepage: can easily view the status of incoming and outgoing orders, fulfill orders, edit current merchandise, and add new items
- Shopper Homepage: show all merchandise from all shopkeepers organized by stores
- Checkout: redirect to third party vendor for payment processing, confirm which items were available for purchase

### *Security concerns*

The most sensitive part of this project is the payment procedure. In this case, all payments will be handled by a third party (think PayPal), which eliminates this risk.

At all sensitive pages, there are checks to ensure that the user is both logged in and the “correct user,” i.e. the owner of the store if editing store items or the owner of the cart if proceeding to checkout.

### *Operations*

- Add to Cart
  - Pre: Item exists, Logged in
  - Frame: Item exists
  - Post: Item in Cart
- Checkout
  - Pre: Item in user cart, Item has positive quantity, Logged in
  - Frame: Item has positive quantity
  - Post: Item not in Cart
- Fulfill Order
  - Pre: Item has pending status, Logged in
  - Post: Item has fulfilled status, Order status may also change
- Create Item
  - Pre: Store exists, Logged in
  - Post: Item in Store
- Create Store
  - Pre: Logged in
  - Post: Store exists
- Modify Item
  - Pre: Logged in, Store exists, Item in Store
  - Post: Item data updated
- View Recent Orders
  - Logged in
- View Cart
  - Logged in
- Save Items
  - Pre: Logged in, Item in Cart
  - Post: Item is Saved, Item not in Cart
- Add Back to Cart
  - Pre: Logged in, Item in Saved

- Post: Item in Cart, Item not in Saved

### User interface

The diagram shows a user interface layout. At the top right, there is a header with the text "home | login". In the center of the page, there is a rectangular box labeled "LOGO". Below the logo, there are two rounded rectangular buttons side-by-side, labeled "Shop" and "Sell".

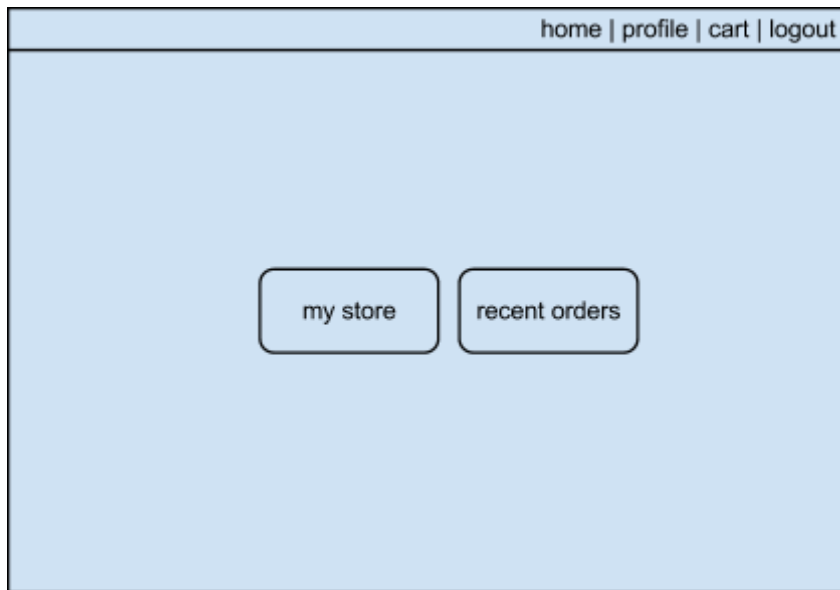
### Homepage

Shop => Shopper Landing Page (After login or signup)

Sell => Storekeeper Landing Page (After login or signup)

The diagram shows a login page layout. It features two stacked rectangular input fields. The top field is labeled "username" and the bottom field is labeled "password". Below these fields is a rounded rectangular button labeled "Login".

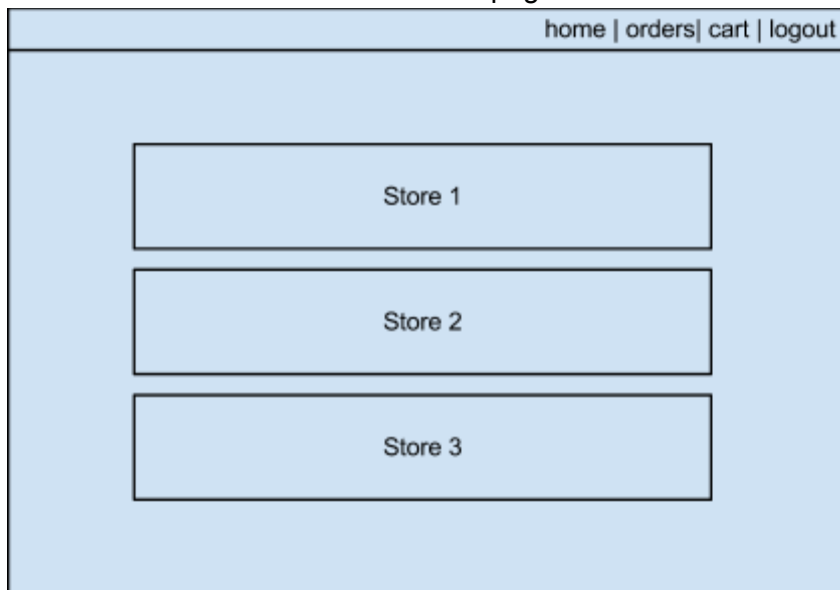
Login page (redirects to home)



Storekeeper landing page

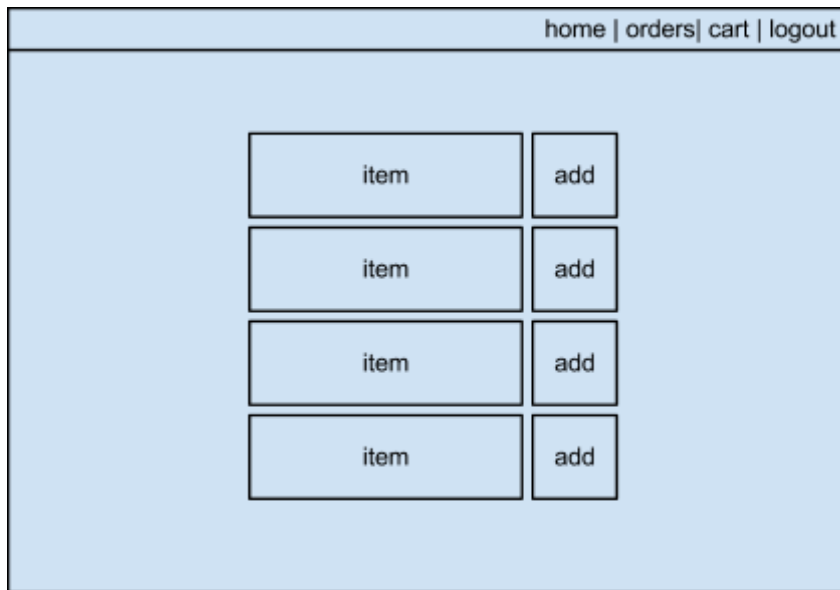
My store leads to page similar to shopping index page, but with edit buttons

Recent orders leads to recent orders page



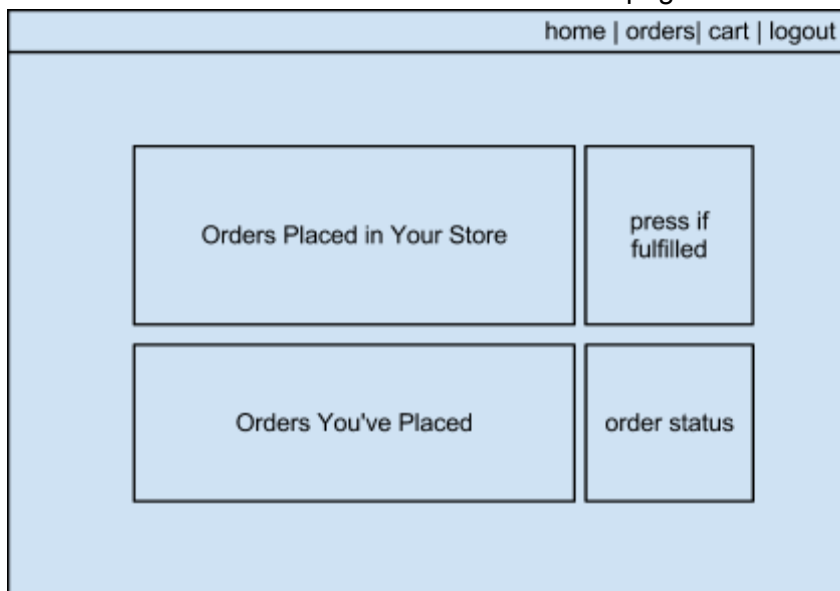
Shopper landing page

Any button in the middle screen => Shopping index page with appropriate filters

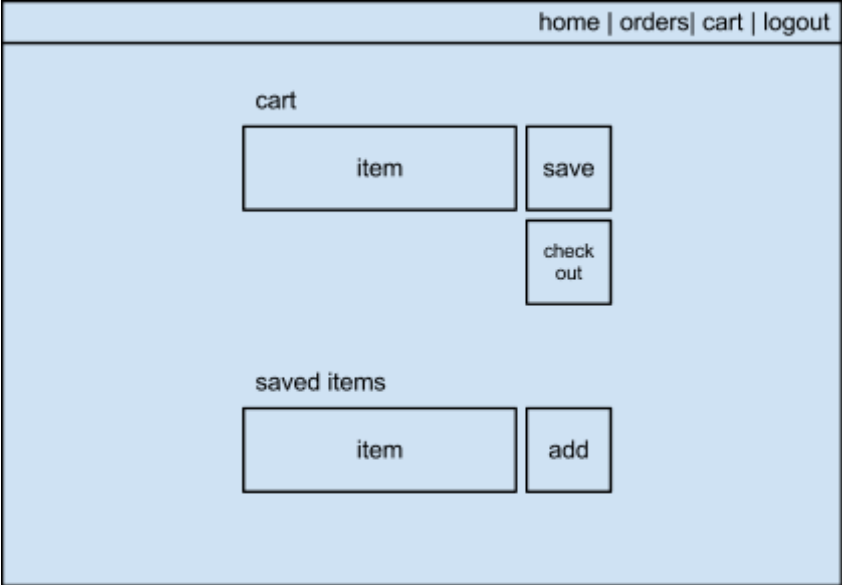


Shopping index page

Add button will add item to cart and lead to cart page



Recent orders: displays both orders you've placed and received, with the ability to fulfilled those orders placed in your store and view the status of those you've placed in other stores



Cart page: allows user to toggle items between cart and saved list, also allows for checkout