

# Design Analysis: AZCart, a Shopping Cart Plugin angelaz

## OVERVIEW

AZCart is a lightweight web application that simulates a shopping cart experience online. AZCart provides a similar service as Opencart, Shopify, Paypal, and Stripe, but it fulfills the shopkeepers' need for a more basic and minimal setup effort required product.

## FEATURES

When a shopper is browsing a website, they can

- 1) create account/log in/log out
- 2) add items to the shopping cart
- 3) remove items
- 4) edit quantity of an item
- 5) checkout items in the cart
- 6) retain the shopping cart across browser sessions
- 7) share shopping carts with other users

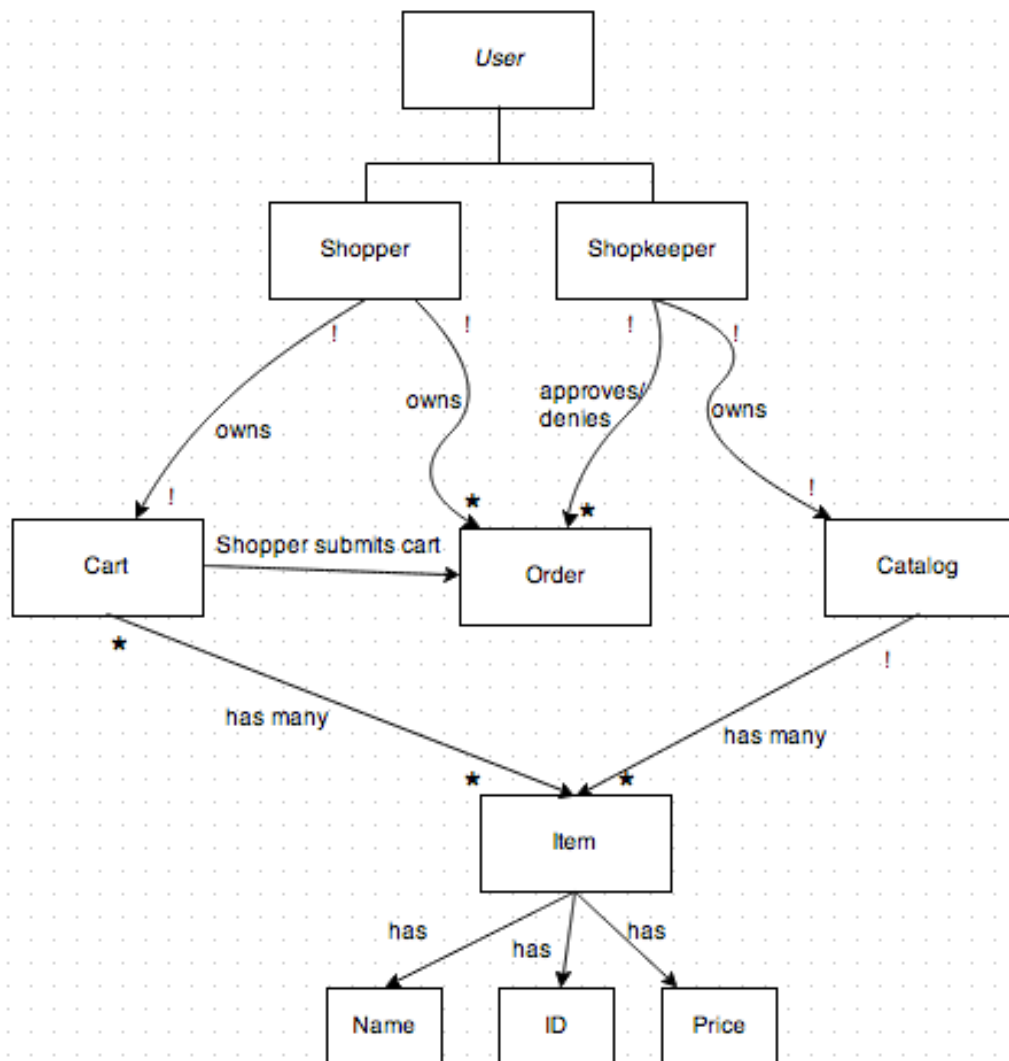
For a shopkeeper, AZCart will provide them a simple admin interface to:

- 1) add items to the catalog
- 2) edit a catalog item's title, price, and quantity
- 3) approve or decline orders submitted by a user

## CONCEPTS

AZCart models the following concepts:

- **User:** Represents a user on the site, could be either a Shopper or a Shopkeeper
- **Catalog:** Represents a catalog of items. Each Shopkeeper owns a Catalog and can add/remove/edit items in the Catalog.
- **Cart:** Represents a cart of items. Each Shopper owns a Cart and can add/remove/edit quantities of items in the cart.
- **Order:** Once the Shopper hits "Checkout," an Order is made from the Cart, which is available for the shopkeeper to approve or deny.
- **Item:** Represents an individual item in the Catalog. Each item can also belong to multiple Carts and Orders.



#### UPDATES (3/12/13):

- Shopper now owns multiple Carts, each with a "status" tag (-1 = denied, 0 = unsubmitted, 1 = submitted, 2 = approved).
- Order class is removed from the design
- The website holds a total of 1 Catalog, but may have multiple Shopkeepers/Sellers who share the catalog and have the same set of permissions.
- When an Item is added to a Cart, a CartItem is created with a foreign-key into the Cart. Each Cart holds multiple CartItems.

## UI

Home Page - Logged Out:

# Catalog Items

Name	Price	Add to Cart
Apple	10.0	Add to Cart
Banana	20.0	Add to Cart
Cactus	30.0	Add to Cart

[Sign up](#) or [log in](#)

Home Page - Seller:

# Catalog Items

Name	Price	Edit
Apple	10.0	Edit
Banana	20.0	Edit
Cactus	30.0	Edit

# Submitted Orders

Cart ID	Buyer Email	Show & Approve
4	buyer2@mit.edu	Show

# Approved Orders

Cart ID	Buyer Email	Show
1	buyer1@mit.edu	Show

# Denied Orders

Cart ID	Buyer Email	Show
3	buyer2@mit.edu	Show

[New Item](#)

Logged in as seller3@mit.edu, a Seller. [Log out](#)

Cart Page – Buyer:

# Items In Cart

**Buyer:** buyer2@mit.edu

**Status:** Unsubmitted

Name	Price	Quantity		
Apple	10.0	2	Edit	Delete

Continue Shopping

Place Order

## Challenges

Below is a list of potential challenges:

- Access control. Allowing users to create account, log in, and log out.
- Retaining the cart across browser sessions, specifically, from not logged-in to creating new account or logging into existing account.
- Cart -> Order. How to maximize efficiency. Deciding whether to copy all the items from one database to another or allow complex permission system to distinguish between Shoppers and Shopkeepers.
- Security concern. Hacking other people's account and stealing credit card number. How to make the account session secure.
- Scaling. What happens when there are lots of shopper and shopkeepers and items? Will the database query run in a reasonable speed?