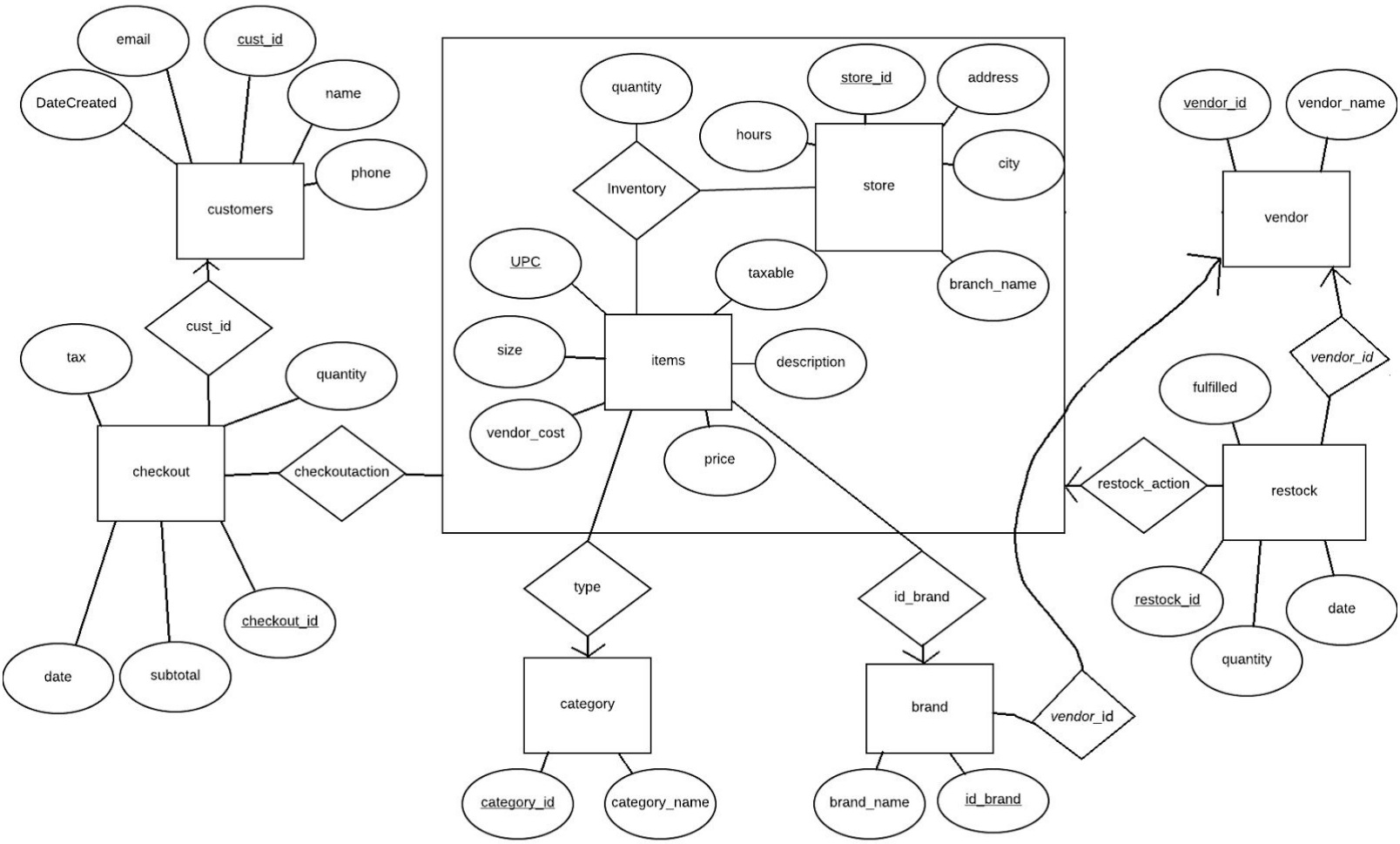
CS 3413 - Group 2
Term Project: Database and Application Design & Implementation

Members: Dean, Fabio, Taha, Zach

Content

E-R Diagram



Additions to the E-R Diagram

n/a

Relational Schema

brand(id_brand, vendor, brand_name)
category(category_id, category_name)
checkout(checkout_id, tax, subtotal, date, cust_id)
checkoutaction(checkout_id, UPC, quantity)
customers(cust_id, name, phone, email, DateCreated)
inventory(UPC, store_id, quantity)
items(description, brand_id, vendor_cost, price, type, taxable, size, UPC)
restock(restock_id, date, quantity, UPC, store_id, vendor_id, fulfilled)
store(store_id, branch_name, address, hours, city)
vendor(id_brand, vendor_id, brand_name)

SQL Files

a. Group2_DDL.sql

b. Group2_INSERTS.sql

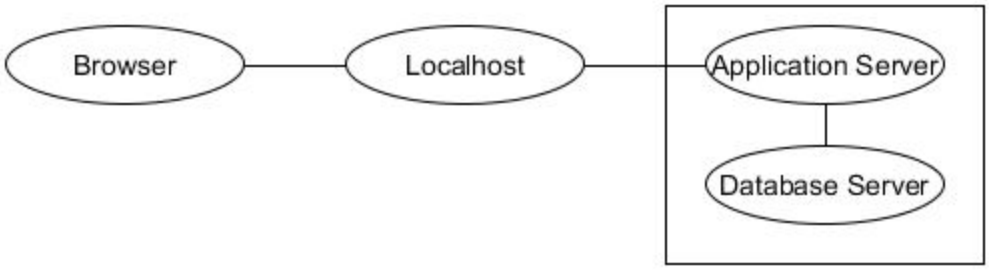
Application Assumptions

Our application is a PHP web application that serves simple html pages and uses PHP for the application logic and the database access. We used Bootstrap 4 and custom css to style the user interface. Our database is hosted as an RDS database instance on AWS in order to have a central database instance that is accessible from everywhere.

We used XAMPP as a convenient way to run the project locally. XAMPP provides us with the option to run an Apache web server combined with an MySQL module, which is all we need to run the project locally. In order to set up the project from scratch, copy the “db-group” folder right in the xampp/htdocs folder (default: C:\xampp\htdocs). When both the Apache and MySQL

instance are running in XAMPP, the application is reachable under the URL "http://localhost/db-group/htdocs/", an internet connection is required to access the bootstrap UI elements.

Application Architecture Diagram



Implemented Features

a. Customer Section

1. Browse Products

Customers can browse all available products. The shop displays both product details and availability.

2. Add to Shopping Cart

Customers can add shopping items to their cart.

3. Manage Shopping Cart

Customers can change the amount of the items in their shopping cart, delete entries and proceed to the checkout.

4. Checkout

Will populate the checkoutaction table with all products and quantities, with a reference to the checkout table which tracks the total price, tax and order ID. This will also reduce the inventory in store 1 by default.

b. Manager Section

1. View low stock inventory

Managers can view any stock that is below a certain threshold

2. Restock request

When viewing low stock inventory managers have an option to request restock which will send a request to the vendor page

3. View Inventory

Managers can view inventories of individual stores

4. CRUD operations for relevant tables

CRUD operations have been implemented for tables to allow managers to edit, add and delete information about stores, vendors, products, brands and categories

c. Vendor Section

1. View restock requests

Vendors are able to check restock requests that have been issued by managers

2. Fulfill restock requests

Vendors are able to fulfill restock requests which update the inventory table

Not implemented yet/ Future Work

- Shipping/arrival dates working with restock fulfillment
- Checkout action should get customer id from login credentials and populate the table with the current date
- Customers should be able to choose which store they want to purchase from and view the available inventory for that store
- Login for different account types (customer, manager, vendor)