COMP3700 Assignment 3

Tripp Isbell
cai0004@auburn.edu

For the first version of the store mangement system, we want to start with the following user stories:

- As a user, I want to add a new product into the system.
- As a user, I want to add a new customer into the system.
- As a user, I want to record a purchase from a customer into the system.

1. Write a common use case for each user story. Sketch the screens the system should display in each use case.

   Use Case: add a product into the system
   Actors: employees
   Goals: update database to include new product
   Preconditions: interface is functional and connected to underlying database
   Postconditions: The product database is updated with the item
   Steps:

   - the user navigates to a display of the product database
   - they click an add button and a prompt appears
   - they enter the information about the product (name, price)
   - they click a button and the database is updated with the new item

   Use case: add a customer into the system
   Actors: employees
   Goals: update database to include new customer
   Related use cases: adding a product (above) **Preconditions, steps, postconditions:** Same as above just replace "product" with "customer" (also they'd be stored in separate databases).

   Use case: record a transaction
   Actors: employees
   Goals: update transaction records with new transaction (customer, product, price paid)
   Related use case: similar to product and customer cases
   Preconditions: interface is set up and connected to transaction database, ideally the process is also set up to be somewhat automated
   Postconditions: the database is updated with the transaction (ideally the system automatically breaks the purchase of multiple products into atomic purchases of each individual product)
   Steps:

   - the employee locates the product(s) to be purchased in the database (this can be done automatically via barcode etc.)
   - the employee uses the customers name (or email, username, phone number, etc) to locate the customer in the database
   - the employee reviews the summary info of the transaction to verify, and then confirms

2. Draw the entity-relationship diagram for this system. We assume the minimal requirement with two entities: products and customers, and one relationship "a customer purchases a product".

3. Design the database logically, i.e., write the relations, attributes, and define keys.
4. Design the database physically using SQL, i.e., write SQL code to create the tables for those relations.
5. Insert data into the tables, with at least 5 products, 5 customers, and 10 purchases.