Samantha High, Team Lead, Deadline Enforcer, and Co-Developer

Darius Alexander, Creativity Officer, Submission Proofreader, and Co-Developer

Project Group 1 - Deft Database Designers

Project Title: Moogle Marketplace Sales & Inventory Database

## Executive Summary

In the beginning of this project we were overly ambitious. We knew we wanted to design an effective database for a fictitious convenience store serving the gaming community, specifically those who attend gaming conventions. Knowing this we started out this project designing a database with 8 entities and 2 intersection tables facilitating the M:M relationships between entities. With each step we made slight adjustments to our database based on the wonderful reviews from our peers and TAs. The most notable changes came from the feedback about the number of entities we were working with. It was pointed out that some of our entities could be folded into other existing entities as attributes. We took that feedback and pared down the entities by folding 2 existing entities into entities they were more dependent on and entirely removing another entity that was now erroneous. Our database finally ended up with a total of 7 tables comprising 5 entities and 2 intersection tables facilitating the M:M relationships between entities. We also received other feedback suggesting that we eliminate all of our intersection tables but those were needed to facilitate the M:M relationships so we decided against following that feedback. Some of our peers also caught some syntactical errors that we ended up correcting and other peers and TAs suggested that we update our ERD and schema to match what was discussed in a new discussion post so we made sure to make those adjustments as well. Other feedback we received was more about the aesthetics of our website as we had no styling in the beginning. Now our website has some color to match the Moogle theme of our fictitious convenience store. Overall we received some good feedback at each step and were able to successfully make necessary changes.

# Project Outline, Database Outline, ERD Schema, Sample Data, and UI Screenshots

Overview

Moogle Marketplace is a new store serving gaming enthusiasts who enjoy attending conventions. Their name is inspired by the square enix final fantasy gaming franchise and they would like to attract gaming enthusiasts as their customer base especially around gaming convention season. Moogle Marketplace sells a variety of convenience and gaming related items, such as food, hygiene products, games, and books. Each site handles 1,000 transactions each day, makes approximately $4 million annually and serves approximately 400,000 customers per year, and our system can handle over 5 million transactions per day. Moogle Marketplace wants a company-wide inventory and accounting system to track its various *locations, products* sold, *employees*, *customers,* and *orders*. We are designing a database-driven website to record this data and store the relevant information in the aforementioned tables: *locations, products* sold*, employees*, *customers, orders, orderDetails* intersection table, and *inventoryDetails* intersection table*.*

Database Outline

Below is a bulleted list of the entities in our database and their corresponding attributes and relationships.

- **locations:** The purpose of this entity is to keep record of all of Moogle Marketplace's sites/stores/locations along with relevant details about those locations. Darius is responsible for implementing.
    - locationID: int, not NULL, primary key
    - locationStreet: varchar, not NULL
    - locationState: varchar, not NULL
    - locationCity: varchar, not NULL
    - locationZip: varchar, not NULL

      Relationship 1: M:M relationship between *locations* and *products* is implemented with the facilitation of the *inventoryDetails* intersection table which contains both the productID and locationID as a foreign keys, where these relationships are 1:M.

      Relationship 2: 1:M relationship between *locations* and *employees* is implemented with the locationID as a foreign key inside of *employees*.

- **products:** The purpose of this entity is to record the products sold by Moogle Marketplace and relevant details about those products. Darius is responsible for implementing.
    - productID: int, auto_increment, not NULL, primary key
    - productName: varchar not NULL
    - productPrice: decimal(14,2) not NULL
    - productCost: decimal(14,2) not NULL
    - productType: varchar not NULL

      Relationship 1: M:M relationship between *products* and *locations* is implemented with the facilitation of the *inventoryDetails* intersection table which contains both the productID and locationID as a foreign keys, where these relationships are 1:M.

      Relationship 2: M:M relationship between *products* and *orders* is implemented with the facilitation of the *orderDetails* intersection table which contains both the productID and orderID as a foreign keys, where these relationships are 1:M.

- **employees:** The purpose of this entity is to record the details of Moogle Marketplace's employees. Required attributes are their unique ID, first and last name, email, phone number, and address. Samantha is responsible for implementing.
    - employeeID: int, auto_increment, not NULL, primary key
    - locationID: int, foreign key
    - firstName: varchar, not NULL
    - lastName: varchar, not NULL
    - employeeEmail: varchar, not NULL
    - employeePhone: bigint, unique, not NULL

      Relationship 1: M:1 relationship between *employees* and *locations* is implemented with the locationID as a foreign key inside of *employees*.

      Relationship 2: 1:M relationship between *employees* and *orders* is implemented with the employeeID as a foreign key inside of *orders*.

- **customers:** The purpose of this entity is to record the details about Moogle Marketplace's customers. Each customer has a unique ID which is the primary key, a name, email address, phone number, and loyalty card number. Samantha is responsible for implementing.
    - customerID: int, auto_increment, not NULL, primary key
    - firstName: varchar, not NULL
    - lastName: varchar, not NULL
    - customerEmail: varchar
    - customerPhone: bigint

      Relationship: 1:M relationship between *customers* and *orders* is implemented with the customerID as a foreign key inside of *orders*.

- **orders:** The purpose of this entity is to record the details of the sales made at Moogle Marketplace. Identifies the customer, employee, and whether or not the order was shipped (0 would indicate no and 1 would indicate yes). Each row is unique. Samantha is responsible for implementing.
    - orderID: int, auto_increment, not NULL, primary key
    - customerID: int, not NULL, foreign key
    - employeeID: int, not NULL, foreign key
    - shipped: tinyint, not NULL

      Relationship 1: M:1 relationship between *orders* and *customers* is implemented with the customerID as a foreign key inside of *orders*.

      Relationship 2: M:1 relationship between *orders* and *employees* is implemented with the employeeID as a foreign key inside of *orders*.

      Relationship 3: M:M relationship between *orders* and *products* is implemented with the facilitation of the *orderDetails* intersection table which contains both the productID and orderID as a foreign keys, where these relationships are 1:M.

- **orderDetails:** Intersection table for the M:M relationship between *orders* and *products.* Also contains the quantity of each product ordered. Samantha is responsible for implementing.
    - orderDetailID: int, auto_increment, not NULL, primary key
    - orderID: int, not NULL, foreign key
    - productID: int, not NULL, foreign key
    - quantity: int, not NULL

      Relationship 1: M:1 relationship between *orderDetails* and *orders* is implemented with the orderID as a foreign key inside of *orderDetails*. Facilitates the M:M relationship between *orders* and *products.*

      Relationship 2: M:1 relationship between *orderDetails* and *products* is implemented with the productID as a foreign key inside of *orderDetails*. Facilitates the M:M relationship between *orders* and *products*.

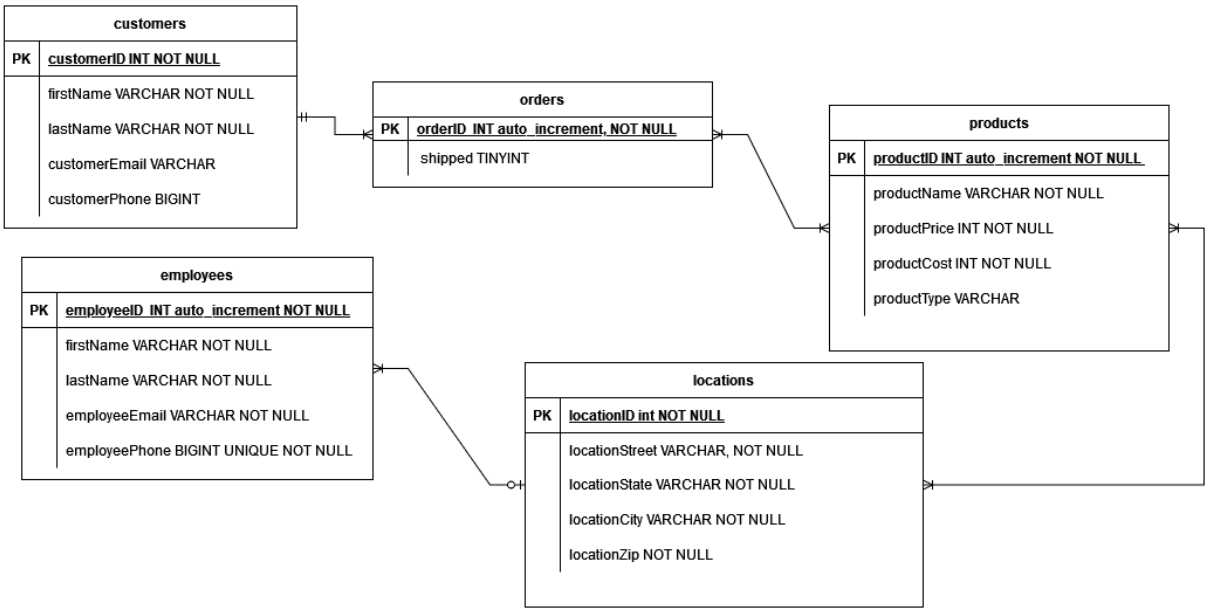
- **inventoryDetails:** Intersection table for the M:M relationship between *locations* and *products.* Also contains the quantity of each product. Darius is responsible for implementing.
    - inventoryDetailID: int,  auto_increment, not NULL, primary key
    - locationID: int, not NULL, foreign key
    - productID: int, not NULL, foreign key
    - quantity: int, not NULL

      Relationship 1: M:1 relationship between *inventoryDetails* and *locations* is implemented with the locationID as a foreign key inside of *inventoryDetails.* Facilitates the M:M relationship between *locations* and *products.*

      Relationship 2: M:1 relationship between *inventoryDetails* and *products* is implemented with the productID as a foreign key inside of *inventoryDetails.* Facilitates the M:M relationship between *locations* and *products.*

Entity-Relationship Diagram

ERD updated to be a more minimalist representation of the entity relationship with only the relevant entities and attributes and without the intersection tables as those will now be represented in the schema along with all of the other attributes. ERD created in Draw.io.



Schema

Schema updated with a consolidation of two previous entities into the attributes of other entities. Intersection tables present to facilitate the M:M relationships.


Example Data

Below is the example data for the entity *locations*.

| Location ID | Street | City | State | Zip |
|---|---|---|---|---|
| 1 | 123 Main | Seattle | WA | 98101 |
| 2 | 45 First | Vancouver | WA | 98663 |
| 3 | 111 Third | San Diego | CA | 92101 |
| 4 | 56 Main | Los Angeles | CA | 90043 |
| 5 | 700 Second | Dallas | TX | 75214 |

Below is the example data for the entity *products*.

| Product ID | Product Name | Product Price | Product Cost | Product Type |
|---|---|---|---|---|
| 1 | Obscurio | 49.99 | 39.99 | Game |
| 2 | Toothpaste | 4.99 | 3.99 | Hygiene |
| 3 | Red Bull | 2.49 | 1.25 | Drink |
| 4 | Food | 4.99 | 2.25 | Sandwich |
| 5 | Book | 14.99 | 9.99 | Game Guide |

Below is the example data for the entity *employees*. The locationID foreign key will be shown to the user in a more user friendly manner as seen in the data table below. All of the people populating this table were created by us and do not represent real people.

| Employee ID | Location | First Name | Last Name | Email | Phone |
|---|---|---|---|---|---|
| 1 | Seattle WA | Harvey | Smith | hsmith@moogle.com | 5551234567 |
| 2 | Seattle WA | Sue | Jones | sjones@moogle.com | 5551001000 |
| 3 | Seattle WA | Tom | Meer | tmeer@moogle.com | 5559995555 |
| 4 | Vancouver WA | Heidi | Schmidt | hschmidt@moogle.com | 5553219999 |
| 5 | Vancouver WA | Camila | Valentine | cvalentine@moogle.com | 5557891234 |

Below is the example data for the entity *customers*. All of the people populating this table were created by us and do not represent real people.

| Customer ID | First Name | Last Name | Email | Phone |
|---|---|---|---|---|
| 1 | Ali | Turner | alikat@randemail.com | 5555007777 |
| 2 | Gabriel | Knight | gabeknight@randemail.com | 5565656666 |
| 3 | Diya | Bolton | boltd@randemail.com | 5568005050 |
| 4 | Fatima | Brewer | fbrew@randemail.com | 5557841111 |
| 5 | Charles | Porter | charlie1@randemail.com | 5567001234 |

Below is the example data for the entity *orders*. Both the customerID and employeeID foreign keys will be shown to the user in a more user friendly manner as seen in the data table below.

| Order ID | Customer | Employee | Shipped |
|---|---|---|---|
| 1 | Charles Porter | Sue Jones | Yes |
| 2 | Fatima Brewer | Sue Jones | Yes |
| 3 | Diya Bolton | Heidi Schmidt | Yes |
| 4 | Gabriel Knight | Camila Valentine | Yes |
| 5 | Ali Turner | Harvey Smith | No |

Below is the example data for the intersection table, *orderDetails*, for the M:M relationship between *orders* and *products.* The productID foreign key will be shown to the user in a more user friendly manner as seen in the data table below. The orderID foreign key will remain a number as that is the most logical way to look up an order number.

| Order Detail ID | Order ID Number | Product | Quantity |
|---|---|---|---|
| 1 | 1 | Red Bull | 2 |
| 2 | 1 | Sandwich | 4 |
| 3 | 1 | Game Guide | 3 |
| 4 | 1 | Generic Toothpaste | 1 |
| 5 | 1 | Obscurio | 5 |

Below is the example data for the intersection table, *inventoryDetails*, for the M:M relationship between *inventories* and *products.* Both the locationID and productID foreign keys will be shown to the user in a more user friendly manner as seen in the data table below.

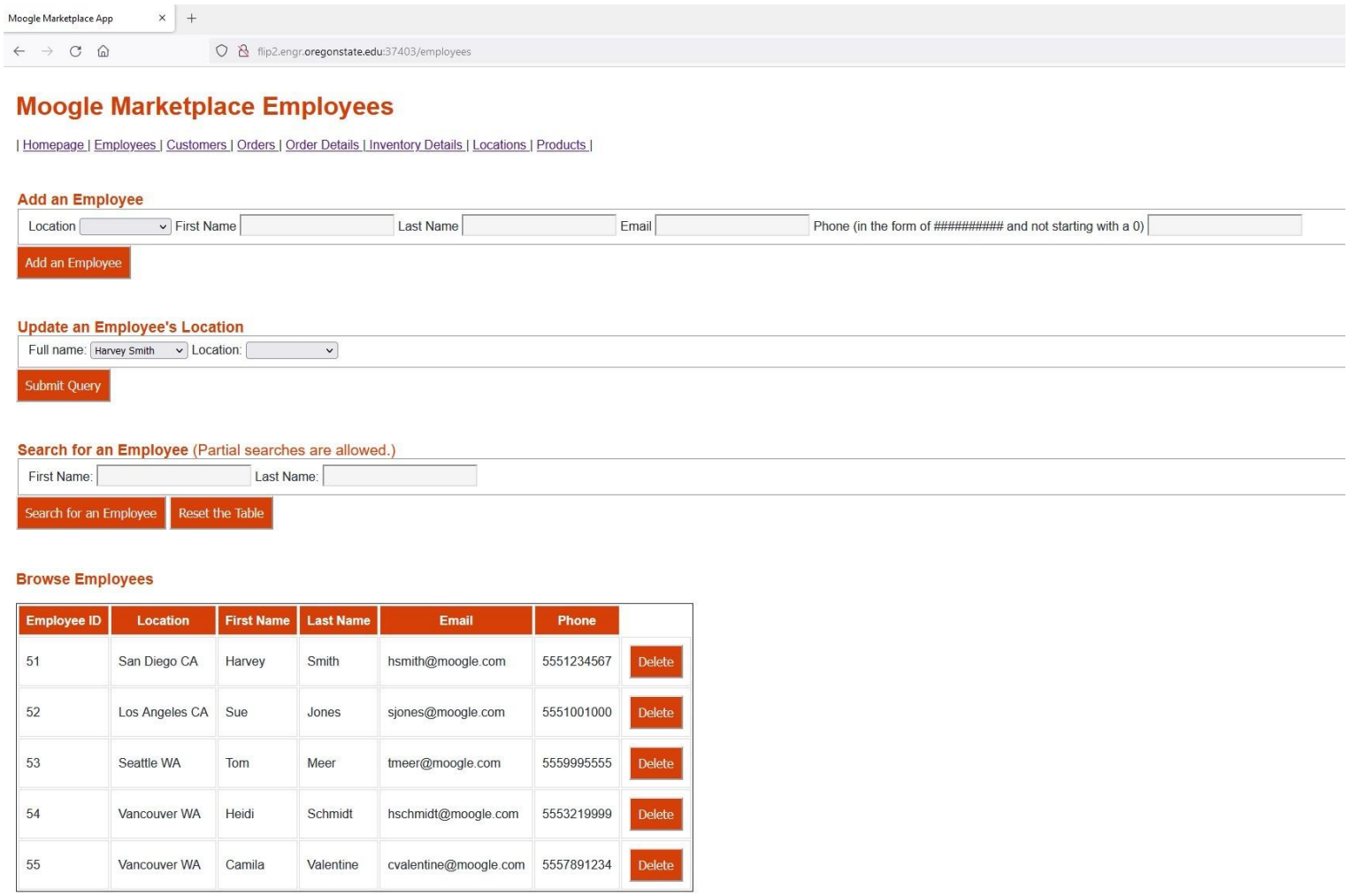
| Inventory Detail ID | Location | Product | Quantity |
|---|---|---|---|
| 1 | Seattle WA | Obscurio | 15000 |
| 2 | Seattle WA | Generic Toothpaste | 35000 |
| 3 | Seattle WA | Red Bull | 10000 |
| 4 | Seattle WA | Sandwich | 20000 |
| 5 | Seattle WA | Game Guide | 50000 |

UI Screenshots:

All screenshots were taken on August 2, 2022 at 10:30am and show what was present in the tables at that time.

**Index Page**
Here you will view and use the links to all other pages.

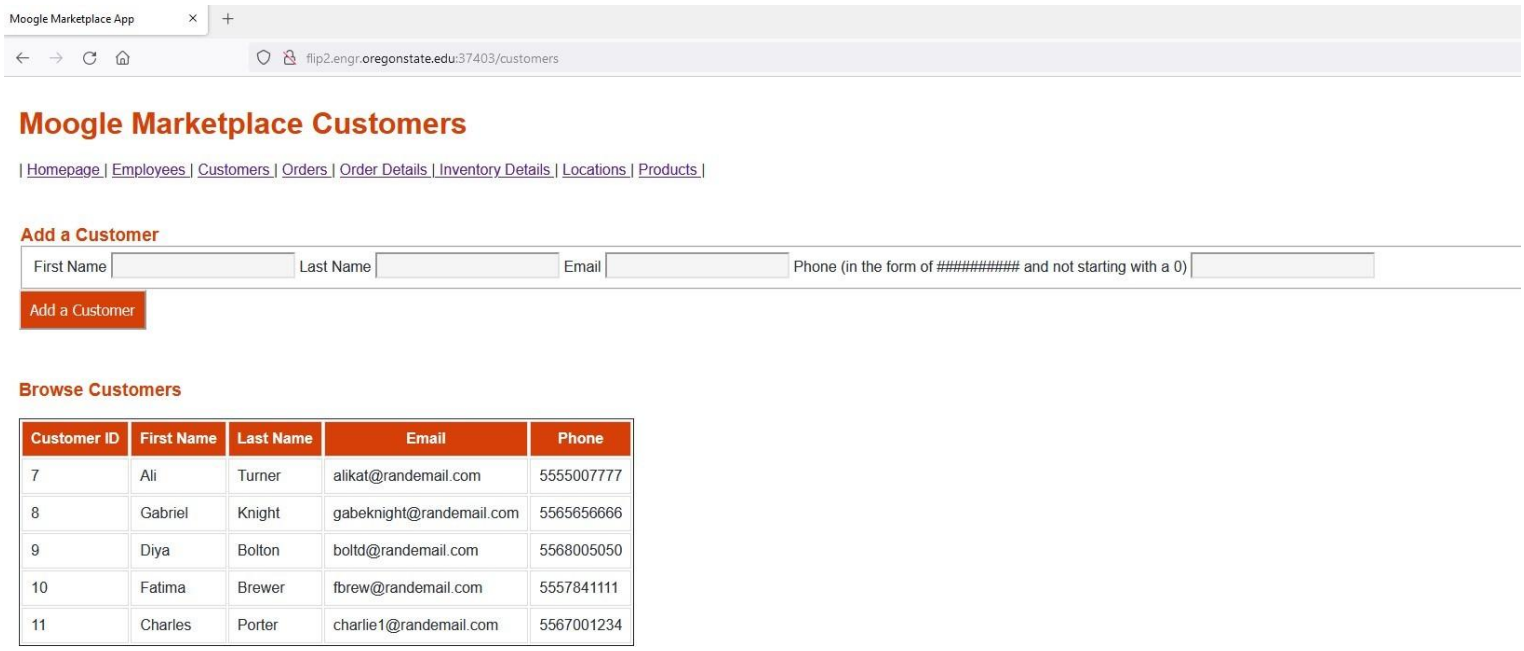

**CREATE/READ/UPDATE/DELETE Employees Page**
Here you can view the list of employees, add a new employee, update an existing employee, delete an existing employee, and search for an existing employee.




| Employee ID | Location | First Name | Last Name | Email | Phone | |
|---|---|---|---|---|---|---|
| 51 | San Diego CA | Harvey | Smith | hsmith@moogle.com | 5551234567 | Delete |
| 52 | Los Angeles CA | Sue | Jones | sjones@moogle.com | 5551001000 | Delete |
| 53 | Seattle WA | Tom | Meer | tmeer@moogle.com | 5559995555 | Delete |
| 54 | Vancouver WA | Heidi | Schmidt | hschmidt@moogle.com | 5553219999 | Delete |
| 55 | Vancouver WA | Camila | Valentine | cvalentine@moogle.com | 5557891234 | Delete |

**CREATE/READ Customers Page**

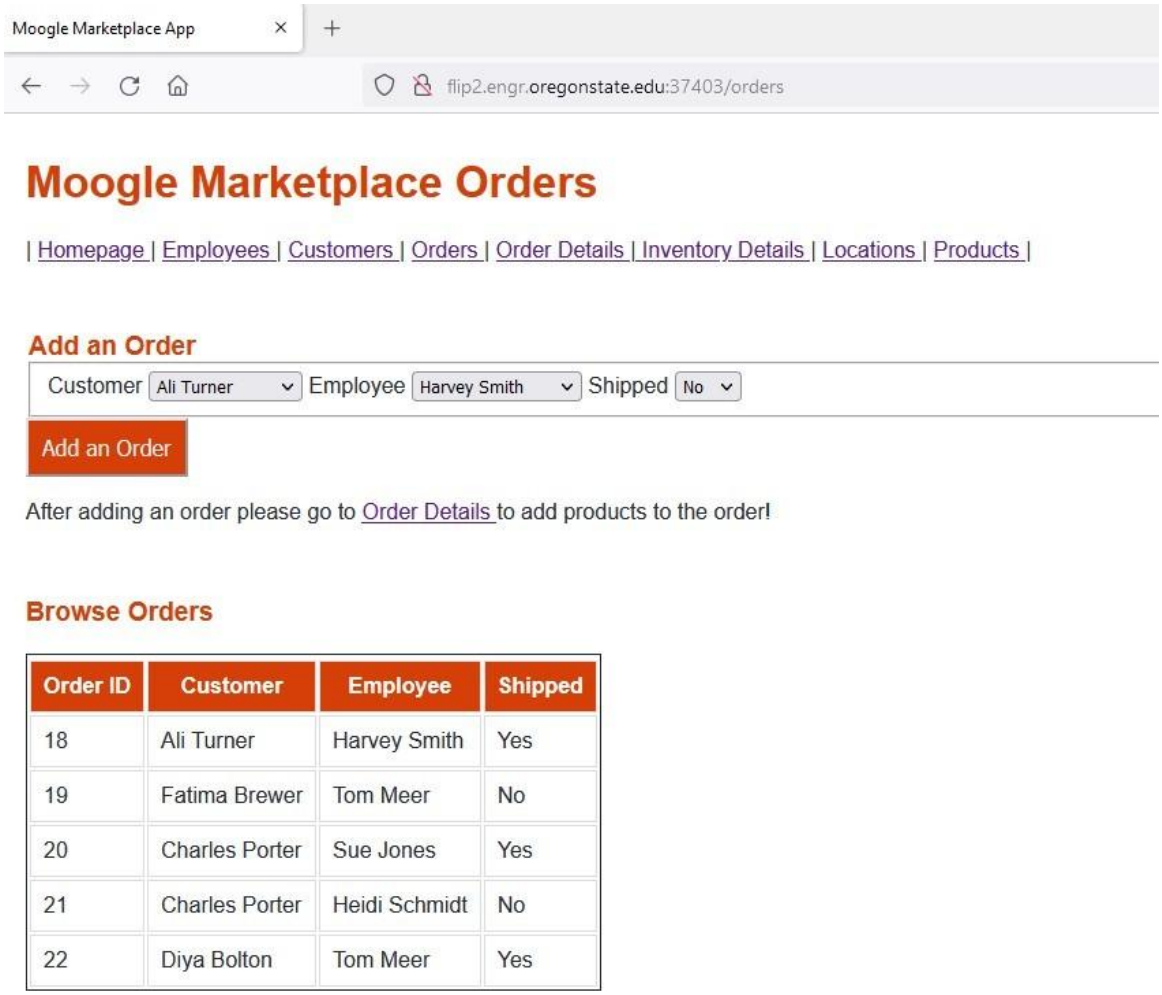
Here you can view the list of customers and add a new customer.




| Customer ID | First Name | Last Name | Email | Phone |
|---|---|---|---|---|
| 7 | Ali | Turner | alikat@randemail.com | 5555007777 |
| 8 | Gabriel | Knight | gabeknight@randemail.com | 5565656666 |
| 9 | Diya | Bolton | boltd@randemail.com | 5568005050 |
| 10 | Fatima | Brewer | fbrew@randemail.com | 5557841111 |
| 11 | Charles | Porter | charlie1@randemail.com | 5567001234 |

**CREATE/READ Orders Page**

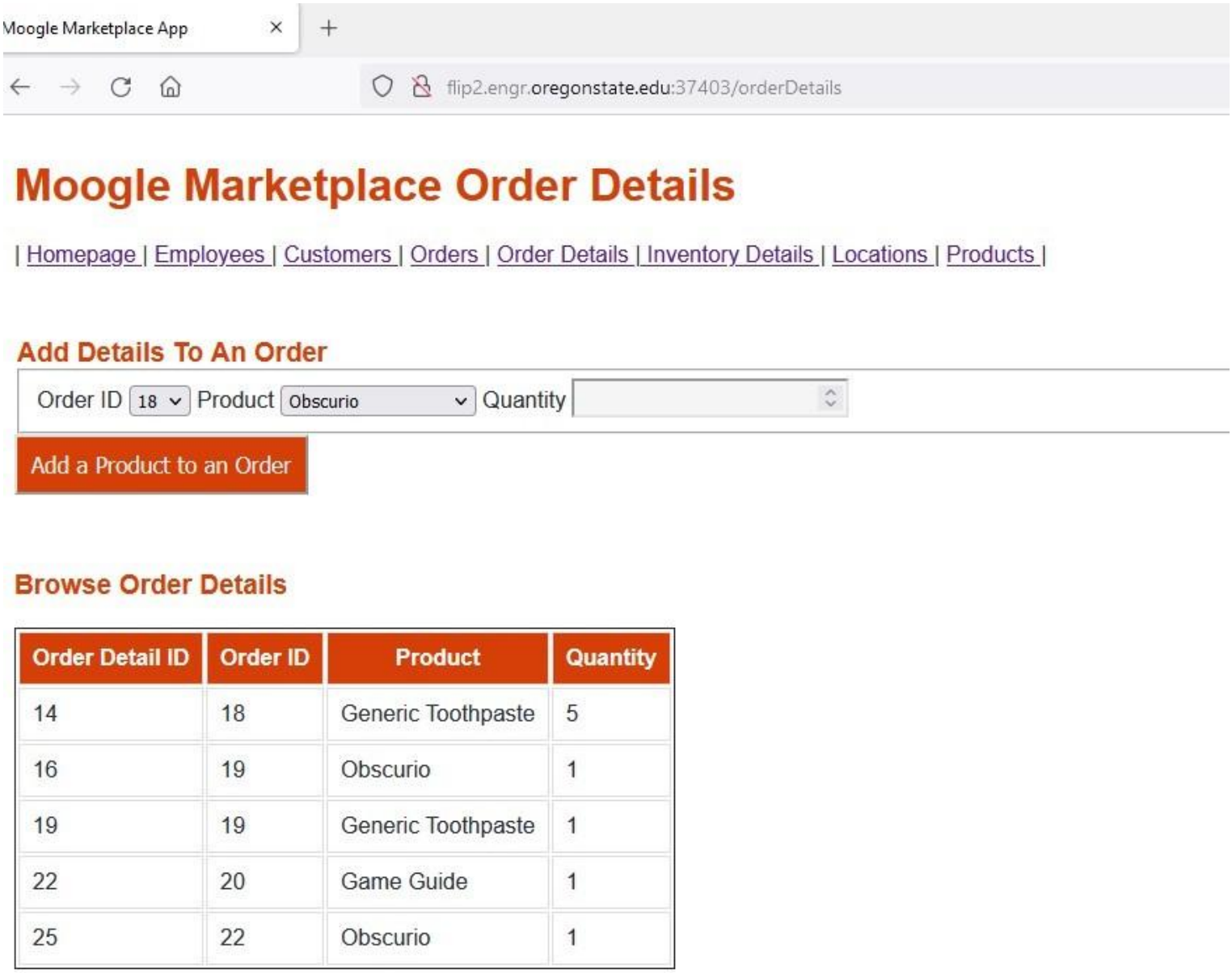
Here you can view the list of orders and add a new order.




| Order ID | Customer | Employee | Shipped |
|---|---|---|---|
| 18 | Ali Turner | Harvey Smith | Yes |
| 19 | Fatima Brewer | Tom Meer | No |
| 20 | Charles Porter | Sue Jones | Yes |
| 21 | Charles Porter | Heidi Schmidt | No |
| 22 | Diya Bolton | Tom Meer | Yes |

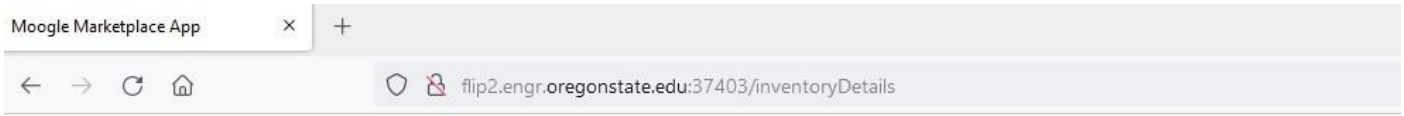
**CREATE/READ Order Details Page**

This page represents an intersection table facilitating the M:M relationship between orders and products. Orders are identified by their order number and products are identified by their name. Here you can view the list of order details and add a new product to an existing order.



**CREATE/READ Inventory Details Page**

This page represents an intersection table facilitating the M:M relationship between locations and products. Locations are identified by their address and products are identified by their name. Here you can view the list of inventory details and add products to an inventory.




# Moogle Marketplace Inventory Details



## Add To Inventory

Location [Seattle WA] Product [Obscurio] Quantity [ ]

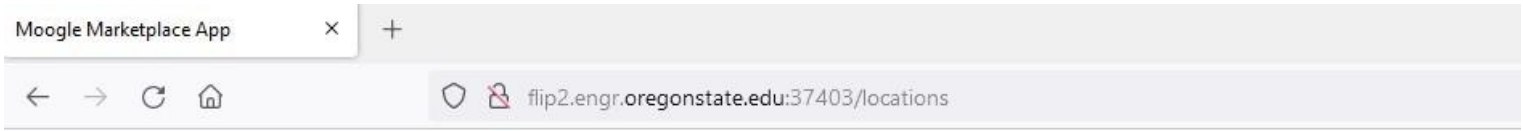
Add to Inventory

## Browse Inventory Details

| Inventory Detail ID | Location | Product | Quantity |
|---|---|---|---|
| 23 | 111 Third, San Diego, CA 92101 | Game Guide | 13 |
| 28 | 111 Third, San Diego, CA 92101 | Obscurio | 25 |
| 29 | 111 Third, San Diego, CA 92101 | Generic Toothpaste | 40000 |
| 30 | 111 Third, San Diego, CA 92101 | Sandwich | 1000 |
| 31 | 111 Third, San Diego, CA 92101 | Kettle Chips | 500 |
| 32 | 111 Third, San Diego, CA 92101 | Cola | 55555 |
| 33 | 123 Main, Seattle, WA 98101 | Obscurio | 35 |
| 34 | 123 Main, Seattle, WA 98101 | Generic Toothpaste | 25333 |
| 35 | 123 Main, Seattle, WA 98101 | Game Guide | 55 |
| 36 | 123 Main, Seattle, WA 98101 | Red Bull | 63475 |
| 37 | 123 Main, Seattle, WA 98101 | Cola | 2354 |
| 38 | 123 Main, Seattle, WA 98101 | Sandwich | 431 |
| 39 | 123 Main, Seattle, WA 98101 | Kettle Chips | 7866 |
| 40 | 456 First, Vancouver, WA 98663 | Obscurio | 10 |
| 41 | 123 Main, Seattle, WA 98101 | Sandwich | 500 |
| 42 | 123 Main, Seattle, WA 98101 | Cola | 563 |
| 43 | 700 Second, Dallas, TX 75214 | Obscurio | 5 |
| 44 | 123 Main, Seattle, WA 98101 | Game Guide | 6 |
| 45 | 123 Main, Seattle, WA 98101 | Generic Toothpaste | 10000 |

**CREATE/READ Locations Page**

Here you can view a list of locations and add a new location.




# Moogle Marketplace Location Details



## Add a Location

Location Street

Location City

Location State

Location Zip

Add Location | Cancel

## Browse Locations

| Location ID | Location Street | Location State | Location City | Location Zip |
|---|---|---|---|---|
| 1 | 123 Main | WA | Seattle | 98101 |
| 2 | 456 First | WA | Vancouver | 98663 |
| 3 | 111 Third | CA | San Diego | 92101 |
| 4 | 56 Main | CA | Los Angeles | 90043 |
| 5 | 700 Second | TX | Dallas | 75214 |

**CREATE/READ/UPDATE/DELETE Products Page**

Here you can view the list of products, add a new product, update an existing product, delete an existing product, and search for an existing product. Deleting an existing product will delete that product from the inventory details intersection table, which removes this product from the M:M relationship. The only aspect of a product that should be updated is its price and that is reflected here.