CMPSC 390 – Homework 2

Project Design & Responsibility Breakdown

Objective

In Lab 2, your team defined **what you are building**.
In this assignment, you will focus on **how the system will work** and **who is responsible for each part**.

You should already have from Lab 2:
• A defined project idea
• A list of proposed features
• A general direction for your application

Step 1: Divide Responsibilities (Group)

As a team, decide how work will be divided for **planning and design**.

Most teams will divide responsibilities into areas such as:
• Frontend planning
• Backend logic planning
• Database design

You may divide responsibilities however you prefer, but:
• Every team member must have a clear responsibility
• Responsibilities must be documented

Deliverable:
• A short list mapping **team members → responsibility areas**

Diego Perez → Frontend

Micheel & Anthony → Backend

Jason → Database

Step 2: Select Three Core Features (Group)

From your existing feature list (Lab 2), select **three features** to design in detail.

These should be:
• Central to your application
• Realistic to build during the semester

Deliverable:
• Names and brief descriptions of the three selected features:

- **User create account/Login**

Enables the user to access additional features, and save more information about the users preferences

- **Leaving a Review**

Allows user to verbalize their opinions on the person or establishment

- **Search for locations**

Allows users to find locations based on filters such as type or address.

Step 3: Frontend Design (Individual)

The team member(s) responsible for frontend planning must create wireframes for how users will interact with the system.
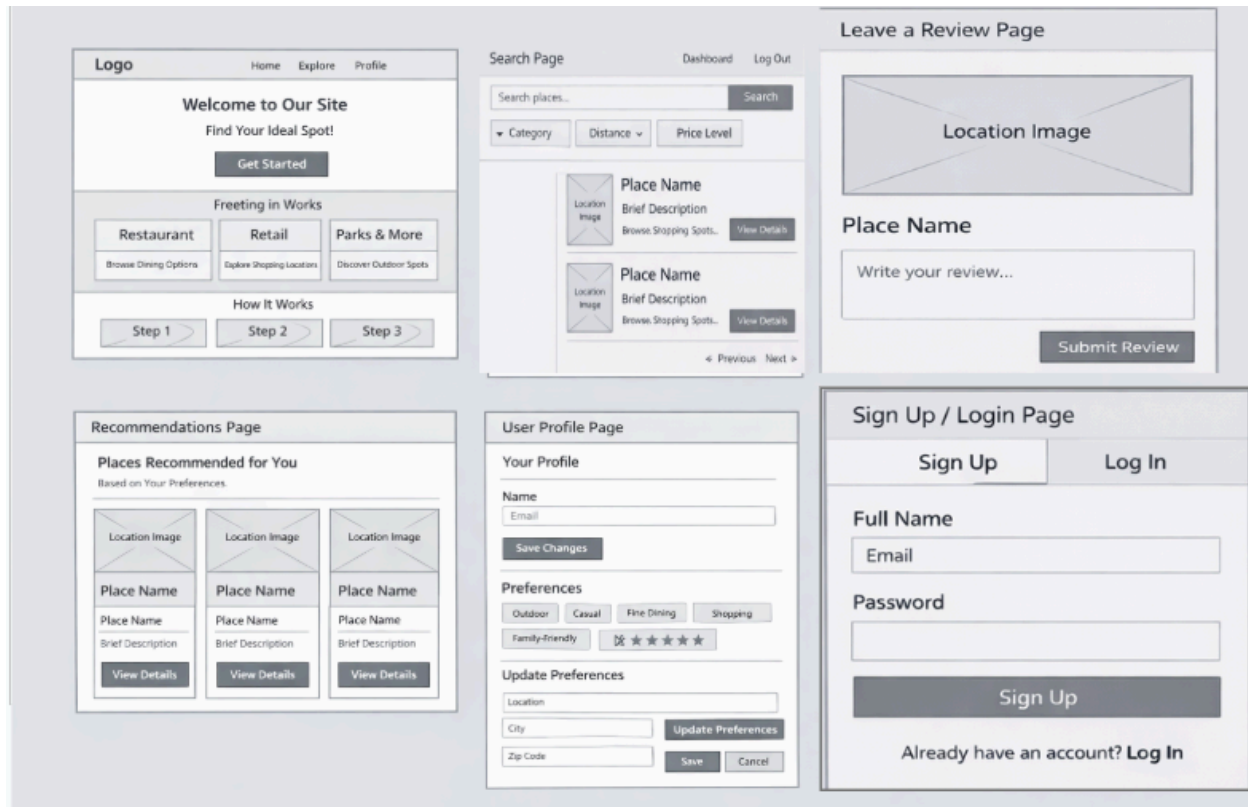
Requirements:
• At least **three wireframes**
• One must be the home or landing page
• Navigation between pages should be clear
• Visual consistency across screens

Deliverable:
• Wireframe diagrams
• Contributor name(s) listed:

Diego Perez

Step 4: Backend Logic Design (Individual)

The team member(s) responsible for backend planning must write **pseudocode** for the three selected features.

Each feature should describe:
• User action
• Backend logic
• Database read and or write
• Response returned to the user

Pseudocode should be:
• Language-agnostic
• Clear enough for another developer to implement

Deliverable:
• Pseudocode for three features
• Contributor name(s) listed: Micheel & Anthony

Step 5: Database Design (Individual)

The team member(s) responsible for data design must define how data will be stored to support the three selected features.

Choose one approach:

Relational Database

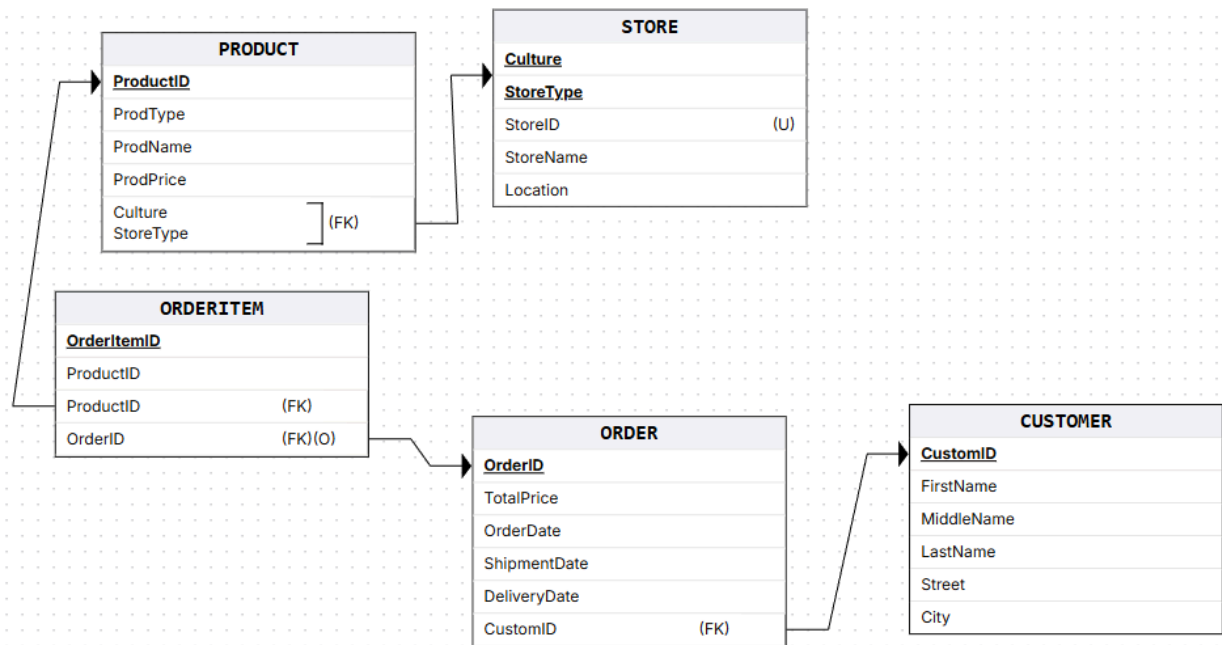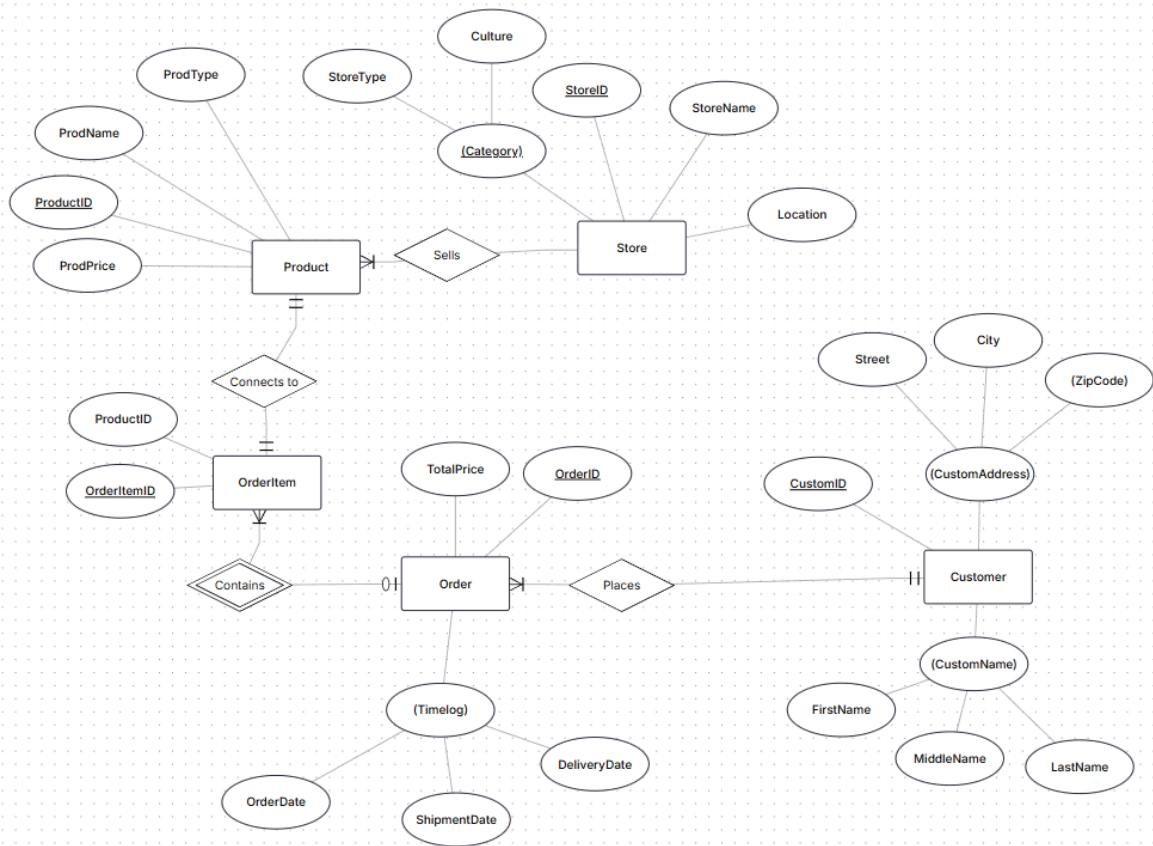• Tables and relationships
• ER diagram

NoSQL Database

• Document structure
• Mock data layout
• Diagram or structured explanation

Deliverable:
• Diagram and explanation
• Contributor name(s) listed:

Jason

Entity-Relationship Diagram

Conceptual Model (Top):

- ProdType
- StoreType
- Culture
- StoreID
- StoreName
- ProdName
- (Category)
- ProductID
- Location
- ProdPrice
- **Product** —< Sells >— **Store**
- Connects to
- ProductID
- TotalPrice
- OrderID
- Street
- City
- (ZipCode)
- OrderItemID
- **OrderItem**
- (CustomAddress)
- CustomID
- Contains
- **Order** — Places — **Customer**
- (Timelog)
- (CustomName)
- FirstName
- OrderDate
- ShipmentDate
- DeliveryDate
- MiddleName
- LastName

Logical Model (Bottom):

PRODUCT	
ProductID	
ProdType	
ProdName	
ProdPrice	
Culture	(FK)
StoreType	

STORE	
Culture	
StoreType	
StoreID	(U)
StoreName	
Location	

ORDERITEM	
OrderItemID	
ProductID	
ProductID	(FK)
OrderID	(FK)(O)

ORDER	
OrderID	
TotalPrice	
OrderDate	
ShipmentDate	
DeliveryDate	
CustomID	(FK)

CUSTOMER	
CustomID	
FirstName	
MiddleName	
LastName	
Street	
City	

Step 6: Tooling & Hosting Setup (Group)

As a team, ensure the following are set up:

• One GitHub repository with all members/instructor added
• One project board (Trello or equivalent)
• One communication method selected

Instructor access required:
• GitHub: @williamburbatt
• Trello: @williamburbatt2

In addition, identify **one potential hosting approach** for your project.
• Research only
• No deployment required

Deliverable:
• Short write-up describing the hosting option

We are still deciding on a hosting option but for now our potential choice could be one of the following: Google Firebase, Google Cloud, AWS, or Digital Ocean. We will decide which one soon based on accessibility, free-features and best suited for our web application.

Submission Requirements

One submission per team by the Project Manager.

Your GitHub repository must include:
• A documentation folder containing:
• Responsibility breakdown
• Frontend wireframes
• Backend pseudocode
• Database design
• Hosting notes

Submit in Canvas:
• Link to the team GitHub repository