**CS5200 – Project**

**Husky Eats – A Food & Grocery Delivery Application**

**Group Name:** Anand BabuBShahabuddinAVijayakumaranP

**Group Members:** Afrah Fathima Shahabuddin, Balaji Sundar Anand Babu, Pradeep Kumar Vijayakumaran

**Project Description**

Husky Eats is a community-driven platform created to improve university staff and students' eating and grocery buying experiences. By providing this lively community with a wide variety of nearby restaurants and supermarkets, we hope to promote the discovery of both delectable foods and necessary daily necessities. Enjoying meals and stocking up on groceries is made easier by the platform's ability to let users share their eating experiences, write reviews, and find new possibilities. Husky Eats wants to establish itself as the premier destination for those seeking to socialise with their peers and savour delicious meals.

This application will support CRUD (Create, Read, Update, Delete) operations for managing the customer login data, cart, order and delivery information, allowing for customers to login / sign up, order and receive orders with OTP security and delivery agents to login and assign themselves orders to deliver.

**Rationale for choosing this Domain:**

The university community's increasing demand for easy access to shopping and dining options is the reason behind the selection of this domain. Since faculties and students frequently have hectic schedules, a platform that combines groceries and meal services improves their overall experience. Furthermore, encouraging user connections through common dining experiences encourages local food culture and community involvement. In the end, Husky Eats promotes research and discovery of a variety of culinary products while attending to actual demands inside the university.

**Technical Specifications for the Project:**

* **Database Management System:** MySQL for structured and relational storage.
* **Database Management Interface:** MySQL Workbench
* **Backend Framework:** Flask, integration and control.
* **Frontend Stack:** HTML, CSS, Bootstrap for creating an interactive and decorated user interface.
* **Repository Management: GitHub, utilized for version control and collaborative coding**

**README:**

This application connects to a MySQL database and allows users to create accounts in the Husky Eats application, add items to their cart, place orders, and manage their order history, including assignment of delivery and delivery status

**How to Install:**

For MySQL, follow the instructions provided in the MySQL Installation Guide on setting up MySQL on your system.

**https://northeastern.instructure.com/courses/192325/pages/do-install-mysql-relationaldatabase?module\_item\_id=10552701**

For Python, visit the Official Python website and download the latest version suitable for your operating system. Follow the instructions provided to complete the setup.

**https://www.python.org/downloads/**

After Installing Python, do the following:

* Open Command Prompt or Terminal.
* To install flask run: pip install flask --user. To verify installation, run: pip show flask
* If you encounter permission issues, use sudo on macOS/Linux: sudo pip install flask

**Project Structure:**

Inside the project submission, in the Web\_Files folder, it consists of the following Python files:

* **main.py:** This file is responsible for running the main application and handling user interactions, including setting up the web server, routing requests, and managing sessions for user login, account creation, and order placement.
* **sql\_queries.py:** This file contains the implementation of the SQL query logic for interacting with the MySQL database. It defines functions for querying and modifying data, such as retrieving user information, managing cart items, checking stock, placing orders, and updating inventory.

The SQL/dump folder contains 'husky\_eats.sql' which is the database dump of the 'husky\_eats' database, and the Web\_Files contains the necessary .html files to facilitate the frontend app (GUI).

**How to Run**

* Navigate to the directory where the project files are located
* Before running the application, note that images for food and grocery items are stored in the database as medium BLOBs. Since they cannot be dumped directly from the database, we have provided two Python scripts that will insert them back into the database
* In Web\_Files/images, open insert\_img.py, open the script and set the path for the **menu\_images** folder as per your local machine
* In Web\_Files/images, open insert\_img\_groc.py, open the script and set the path for the **grocery\_images** folder as per your local machine
* Open Command Prompt or Terminal
* Run the insert\_img.py using Python with the command: python insert\_img.py
* Run the insert\_img\_proc.py using Python with the command: python insert\_img\_proc.py
* Finally, Run the application using Python with the command: python main.py and do ctrl + click on the IP address that displays on the terminal and give ctrl + c to close the application
* **Note:** At a time, you can run the application and login both as a customer and delivery agent in 2 different tabs.

**Additional Notes:**

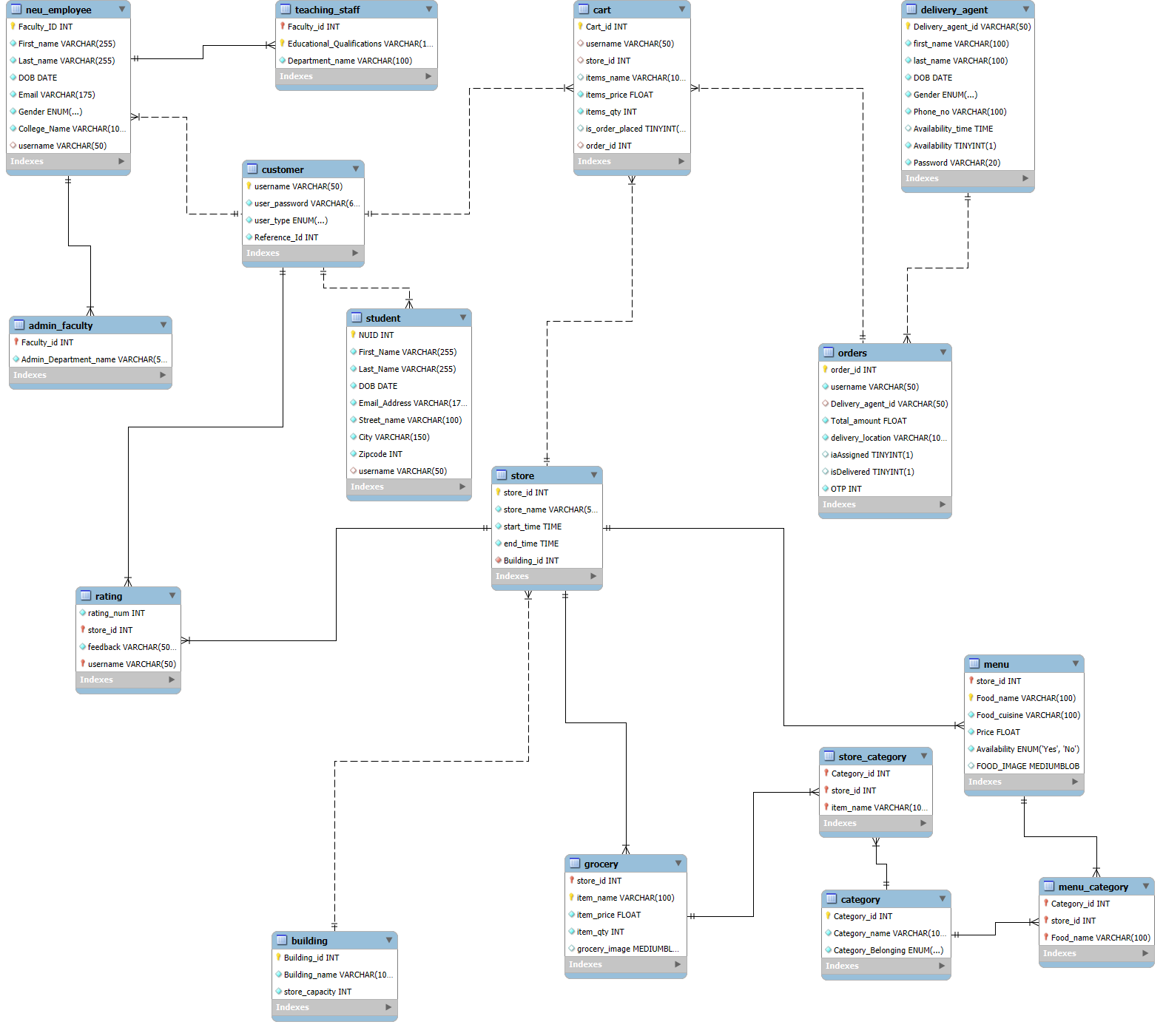
* Ensure your MySQL server is running and accessible
* Ensure you have Python and the necessary libraries installed
* Verify that you have the correct username and password
* Make sure your MySQL user has the necessary permissions to access the 'husky\_eats' database
* Update the database configuration in sql\_queries.py as needed if your setup is different (host, database etc.)

**Conceptual Design:**

**A diagram of a computer flowchart

Description automatically generated**

**Logical Design:**

****

**How the User Interacts with the Application:**

* The Husky Eats web application is designed to provide a seamless and engaging experience for university students and faculty as they explore local dining and grocery options.
* Users can easily create accounts or log in to access personalized features that enhance their interactions with the platform.
* Upon logging in, users are greeted with a graphical user interface that allows them to browse a variety of food and grocery items. They can quickly search for specific products or explore different categories, making it easy to find what they need.
* Each store listing provides detailed information and user ratings, helping users make informed choices
* As users compile their selections, they can manage their shopping cart with ease, reviewing items and adjusting before proceeding to place the order.
* The ordering process is streamlined, ensuring that users can confirm their choices and enter delivery information with minimal effort.
* In addition, users can track the status of their orders in real-time. They will be notified whether their order is being processed, out for delivery, or has been delivered successfully.
* After receiving their orders, users can rate the stores from which they bought items, thus allowing the app too promote community involvement by enabling users to share their dining experiences and perspectives with others.
* Users feel more connected to one another because of this contact, which supports the university's thriving culinary culture.
* The Husky Eats web application is designed to provide a seamless and engaging experience for both university students and faculty, as well as delivery agents, as they interact with the platform. Upon logging in, delivery agents can access personalized features that help them efficiently manage and fulfill orders.
* Upon logging in, delivery agents are greeted with a graphical user interface that allows them to view a list of available orders.
* They can quickly browse through different orders, see details such as delivery locations and customer information, and choose the ones they wish to deliver, making it easy to select the best options for them
* Overall, the Husky Eats web application is designed to be user-friendly and responsive, enabling students and faculty to easily access food and grocery options while enhancing their overall experience within the local community.

**Activity Diagram:**

A black screen with white rectangles

Description automatically generated

**List of Procedures and Functions in MySQL:**

**Procedures:**

CustomerSignup

get\_category

get\_grocery

get\_grocery\_category

get\_menu\_category

get\_top\_menu

update\_cart

**Functions:**

get\_delivery\_name

get\_email

get\_name

**Lessons Learned:**

**Technical Expertise Gained**

**Database Design and Management:** Gained experience in designing a robust relational database schema that supports various entities such as customers, orders, carts, restaurants, and delivery agents. This included establishing the relationships between these entities and ensuring the database could handle CRUD operations effectively.

**Security and Authentication:** Implemented OTP (One Time Password) authentication for order verification. This provided deeper insights into secure user authentication techniques.

**Frontend - Backend Integration:** Worked on integrating the frontend interface with the backend database to handle user interactions such as signing up, logging in, placing orders, and assigning deliveries.

**Insights, Time Management Insights, Data Domain Insights**

**Time Management:** The most significant challenge was ensuring that all aspects of the project, including database design and frontend work, were completed within the timeline. Some features required more time for integration and testing than anticipated. Future project planning could benefit from allocating additional time for troubleshooting.

**Data Domain Insights:** A key insight was understanding the importance of normalizing the database schema to prevent data redundancy and ensure efficient queries.

**Realized or Contemplated Alternative Design / Approaches to the Project**

**Authentication:** Initially, we considered only using traditional username password authentication, but the decision to implement OTP for order acceptance was driven by the need for added security, especially for handling customer orders.

**Order Tracking:** For order tracking, we initially thought of using a simple status column in the orders table. However, a more efficient approach could have involved using a system that pushes order updates in real-time directly to the customer's device, offering a more interactive and engaging experience.

**Future Work:**

**Planned Uses of the Database:**

**Expanding the Food Options:** The current platform focuses on providing a set of predefined restaurants. Future work can include adding more diverse and dynamic restaurant options, possibly sourced through partnerships with local eateries, which would require updates to the database schema to accommodate more detailed restaurant and menu information.

**Analytics and Reporting:** The database will be leveraged to create detailed reports on user behavior, popular restaurants, peak ordering times, and more, helping businesses to make informed decisions on menu updates, promotional activities, and customer engagement strategies.

**Potential Areas for Added Functionality:**

**User Profile Enhancements:** Allow users to create and customize profiles with preferences for types of food, favorite restaurants, dietary restrictions, etc. This could help personalize the experience and improve customer satisfaction.

**Loyalty Program:** Introducing a loyalty program for frequent users, where customers earn points for every order they place (i.e.) like the Huskies Rewards App.

**Order Prediction and Recommendation System:** Implement a machine learning model that predicts users' preferred food items based on past orders, location, and seasonality. This could lead to a more personalized experience and potentially higher conversion rates.

**Delivery Fee Calculation Based on Distance:** A dynamic pricing model for delivery fees based on the distance between the customer and the restaurant could be added, requiring real-time geolocation data and algorithms to calculate delivery charges.

**Conclusion:**

Husky Eats effectively enhances the dining and grocery shopping experience for university staff and students by providing easy access to local restaurants and supermarkets. The platform promotes community engagement through shared reviews and discovery of new food options, while ensuring a smooth user experience with efficient management of orders, carts, and delivery information. As Husky Eats grows, it has the potential to become an integral part of campus life, offering convenience and fostering connections within the university community.