Restaurant Management System

Sahithi Chigurupati, Jyothika Jalla

Northeastern University

[chigurupati.sah@northeastern.edu](mailto:chigurupati.sah@northeastern.edu), [jalla.j@northeastern.edu](mailto:minsky@mit.edu)

Abstract

The primary goal of the restaurant database management system is to streamline and automate the intricate day-to-day operations of a restaurant, enhancing operational efficiency, customer satisfaction, and decision-making processes. This comprehensive system empowers customers with the convenience of exploring the restaurant's menu after a seamless login, allowing them to peruse a wide array of tantalizing food options, each accompanied by its corresponding price. Customers can effortlessly make their selections from the menu and add items to their orders, simplifying the ordering process. Moreover, the system extends a hospitality hand to patrons by enabling them to reserve tables at their desired dining time or join a waiting list for reservations when all tables are occupied. On the kitchen front, the system equips chefs with a real-time queue of incoming orders, enabling them to efficiently manage the preparation process. As the orders are expertly crafted, chefs can promptly update the order status to ready, facilitating swift and accurate delivery to eagerly awaiting customers. Furthermore, restaurant managers are provided with valuable insights and data-driven decision-making capabilities as they can access statistics encompassing weekly sales.

Database Design

A diagram of a computer program

Description automatically generated

Figure 1: Entity-Relationship Diagram.

Key Design Decisions

In designing the database, our paramount goal was to create a robust and normalized schema that not only captures the intricate relationships among various entities but also mitigates data redundancy and ensures maintainability.

Entities

The selection of entities is crucial in defining the core components of the database. The following entities have been identified to encapsulate the essential aspects of restaurant operations.

• Customer: Captures customer details.

• Food\_Item: Represents items available on the menu.

• Course: Type of course the food item belongs to.

• Cuisine: Type of cuisine the food item belongs to.

• Diet: Type of diet the food item belongs to.

• Order: Manages customer orders.

• Reservation: Facilitates table reservations.

• Waiting\_List: Adds customers to waiting list

• Chef: Stores chef information.

• Manager: Manages managerial data.

• Feedback: Stores customer’s experiences.

Relationships

Establishing relationships between entities is pivotal for modeling the interactions within the system.

• A One-to-Many relationship was established between Customer and Order.

• A One-to-Many relationship was established between Customer and Order.

• One-to-Many relationships were employed between Food\_Item and Cuisine, Food\_Item and Course, Food\_Item and Diet.

• A One-to-Many relationship was established between Order and Chef.

• One-to-Many relationships were employed between Customer and Reservation, Customer and Waiting\_List.

• A One-to-Many relationship was established between Customer and Feedback.

• A One-to-Many relationship was established between Customer and Order.

• A One-to-Many relationship was established between Customer and Order.

• A Many-to-Many relationship was established between Order and Food\_Item for placing orders.

Normalization

Normalization is pivotal in refining the database structure, eliminating data anomalies, and ensuring data integrity. Our design adheres to the principles of normalization, specifically applying 1NF, 2NF, and 3NF. This systematic approach optimizes data storage, reduces redundancy, and enhances the overall efficiency of the database.

Data Collection

For the menu items in our restaurant database, we utilized the publicly available dataset provided by Bappy Malik on Kaggle [1]. This dataset, titled "Delicious Cuisines Dataset," includes information about various menu items, including image URLs, contributing to the richness of our restaurant database. The Food\_Item, Course, Cuisine and Diet tables have been created from this dataset. We also curated custom datasets for other tables, incorporating customer, order, reservation, waiting\_list, feedback, chef, and manager information.

Application Description

The Restaurant Database Management System application offers a user-centric, feature-rich experience for customers, chefs, and managers. From streamlined order processing to real-time kitchen coordination and comprehensive managerial insights, the application encapsulates the diverse needs of a modern restaurant environment. The amalgamation of robust backend technologies and an intuitive frontend design results in an efficient and user-friendly platform that significantly contributes to the overall operational efficiency of the restaurant.

Main Features

• User Authentication: Customers, chefs, and managers should have separate login capabilities. New customers are welcome to sign up to access our services.

• Menu Management: Display the restaurant's menu with items, with their respective category, course, and prices. Allow customers to browse and select items.

• Order Management: Customers can create and manage orders, adding items to their cart. The cart shows the final cost of all items.

• Table Reservations: Customers can reserve tables for a specific date and time. A waiting list is also available for managers to add customers to the list.

• Chef Queue: Chefs can view and manage the queue of orders. Orders are initially created in the "Pending" state. Chefs can update them as "In Progress" or "Completed" when prepared.

• Manager Statistics: Managers can access data on weekly sales and other business metrics to track the sales of each food item. These statistics have been plotted using matplotlib library.

• Feedback: Customers can provide feedback which is displayed in the home page of the application.

The front-end is developed using a combination of HTML5, CSS, and Bootstrap. This ensures a responsive and visually appealing user interface across different devices. The application relies on a Flask backend to manage the flow of data, ensuring seamless communication between the front-end and the MySQL database. Pymysql is employed to interact with the database, enabling efficient storage and retrieval of data. The data retrieval is done using stored procedures, functions and triggers created in the database.

Libraries and Tools

• Programming Language: Python.

• Front-End Development: HTML5, CSS, Bootstrap.

• Back-End Development: Flask.

• Tools & libraries: Pandas, Pymysql, Matplotlib, MS Excel.

• Database Systems: MySQL.

Conclusion

In conclusion, the development and implementation of the Restaurant Database Management System marks a significant step towards modernizing and optimizing day-to-day operations within a restaurant setting. We gained valuable insights into database design and normalization. The project also helped us develop proficiency in data processing and cleaning using Python. This project gave us an opportunity to successfully implement a Full-stack application with diverse technologies.

As with any project, there are areas for future enhancement. A payment gateway could be incorporated to enable customers to make direct payments for food orders within the application. Looking ahead, the integration of machine learning techniques for predictive analysis, such as demand forecasting, presents an exciting avenue for future development. By leveraging historical data, the system could provide valuable insights into potential trends, enabling better resource planning and inventory management. This predictive capability would further enhance the application's value in the dynamic landscape of restaurant management.

In reflection, the Restaurant Database Management System project has been a valuable learning experience, and the collaborative effort of the team has culminated in a robust and feature-rich system. For future DS 5110 students, we recommend starting early, fostering effective team collaboration, and continually testing and refining the application to meet evolving requirements. The project has provided a solid foundation, and with additional time, further refinements and expansions could position the system as a cutting-edge solution for restaurant management in the digital age.

References

Dataset

[1] Kaggle. 2021. Delicious Cuisines Dataset (Including Image URL). https://www.kaggle.com/datasets/bappymalik/delicious-cuisines dataset-including-image-url. Accessed: 2023-11-28.

Acknowledgments

This project was made possible by the collaboration and efforts of the entire team. We acknowledge the support and guidance received throughout the development process. We would also like to thank Professor Roi Yehoshua and the teaching assistants of the DS 5110 course for their invaluable support and constructive feedback on our project milestones.