

Restaurant Management System

About:

This project aims to develop a Restaurant Management System (RMS) using JAVA programming. The current system at the quaint restaurant is likely inefficient, lacking features to streamline operations. This can lead to issues like slow order processing, inventory mismanagement, and inaccurate record keeping. So, we create an RMS that can handle menu items, orders, and inventory efficiently. It allows administrators to manage menu items by inserting, deleting, or updating their details. Customers can browse the menu, place orders, and avail discounts if applicable. Orders are assigned unique identifiers and stored for future reference. Payment processing and inventory management functionalities are also included. Additionally, restaurant staff can view and manage placed orders for efficient order fulfillment and transaction recording.

Features:

The Restaurant Management System is designed to streamline the operations of food ordering and payment processes. The system includes the following key features:

Customer Features

- **Place Order:** Customers can view the menu, select food items, specify quantities, and place orders.
- **Discounts:**
 - Regular customers pay the full price.
 - Students receive a 10% discount on the total bill.
- **Order Summary:** Customers receive a detailed summary of their order, including food items, quantities, and prices.

Staff Features

- **Login System:** Staff members must log in with valid username and password.
- **Manage Food Items:**
 - **Add Food:** Add new food items to the menu with a unique ID, name, and price.
 - **Update Food:** Modify existing food item details.
 - **Delete Food:** Remove food items from the menu.
 - **Display Food Items:** View the current list of food items.
- **Confirm Payment:** Staff confirm that customers have paid their bills to complete the order process.

Ways of Income:

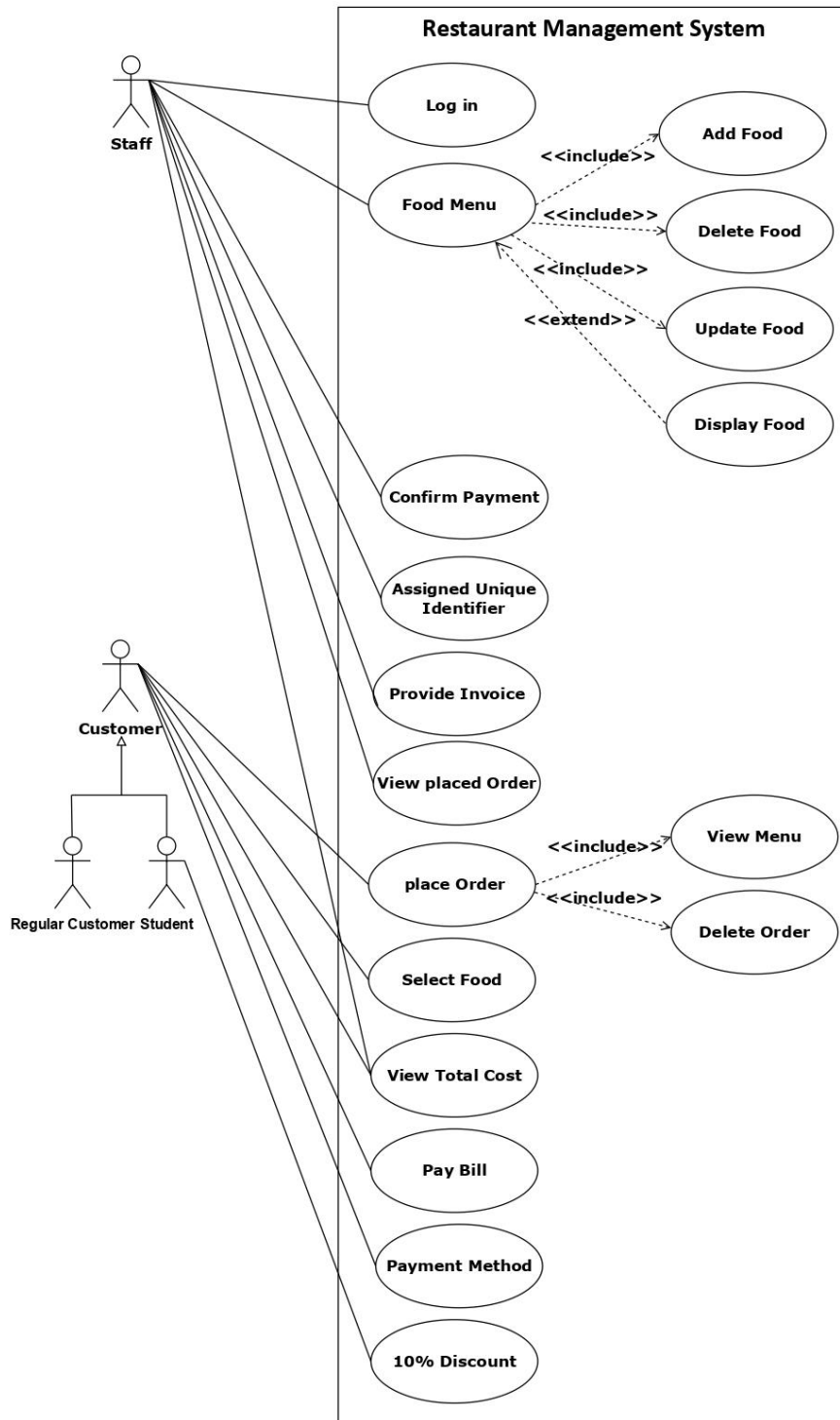
The system facilitates multiple revenue streams by leveraging the food ordering process and customer interaction. The key income sources include:

- **Direct Sales:** Revenue from food sales, including orders placed by regular customers and students.
- **Dynamic Pricing:** Prices of food items can be adjusted through the Manage Food Items feature to reflect demand, promotions, or seasonal changes.
- **Membership Benefits:** Regular and student customers could be encouraged to join membership programs with loyalty rewards for repeat orders.
- **Advertisement Partnerships:** Display ads or promotional offers in the menu interface to generate additional income through collaborations with third-party businesses.

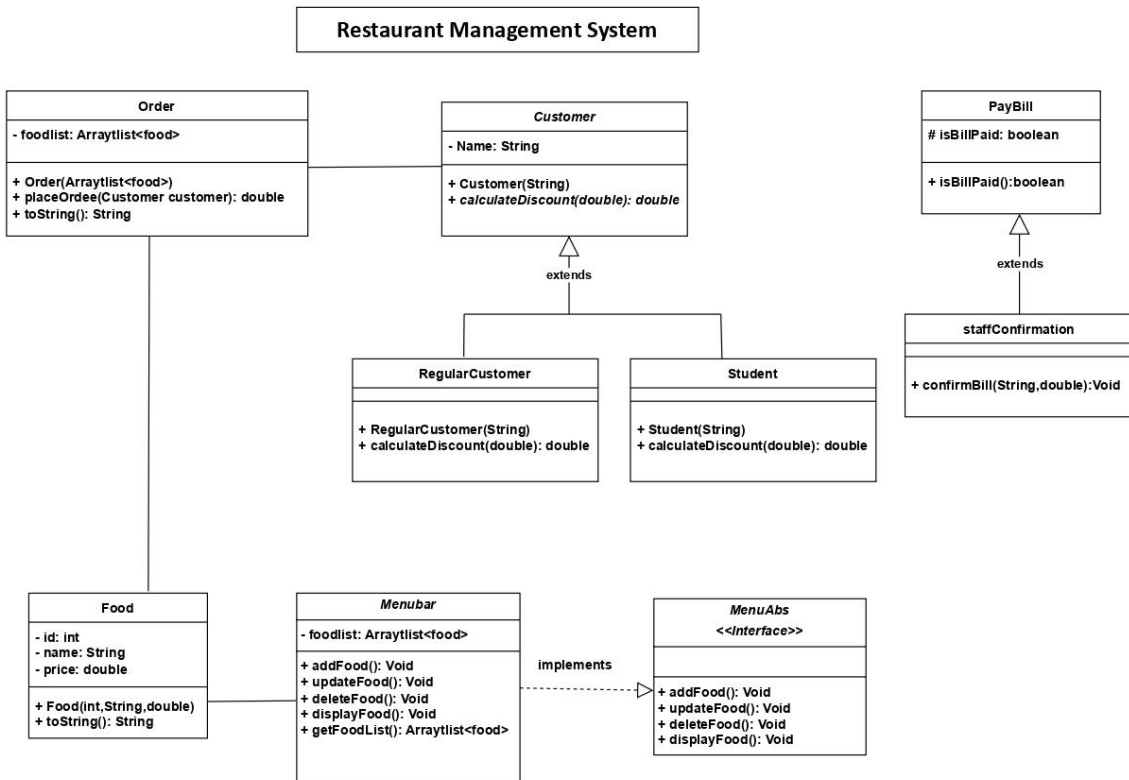
Stakeholders:

- **Customers:**
 - Regular Customers: Individuals who place orders and pay the listed prices.
 - Students: Customers eligible for a 10% discount on their orders.
- **Staff:** Administrators who manage food items and confirm payments.
- **Restaurant Owners:** Oversee the overall system operations and generate reports for business insights.
- **Third-Party Partners:** Companies that collaborate for advertisements or promotions.
- **System Developers:** Responsible for maintaining and updating the system based on business needs

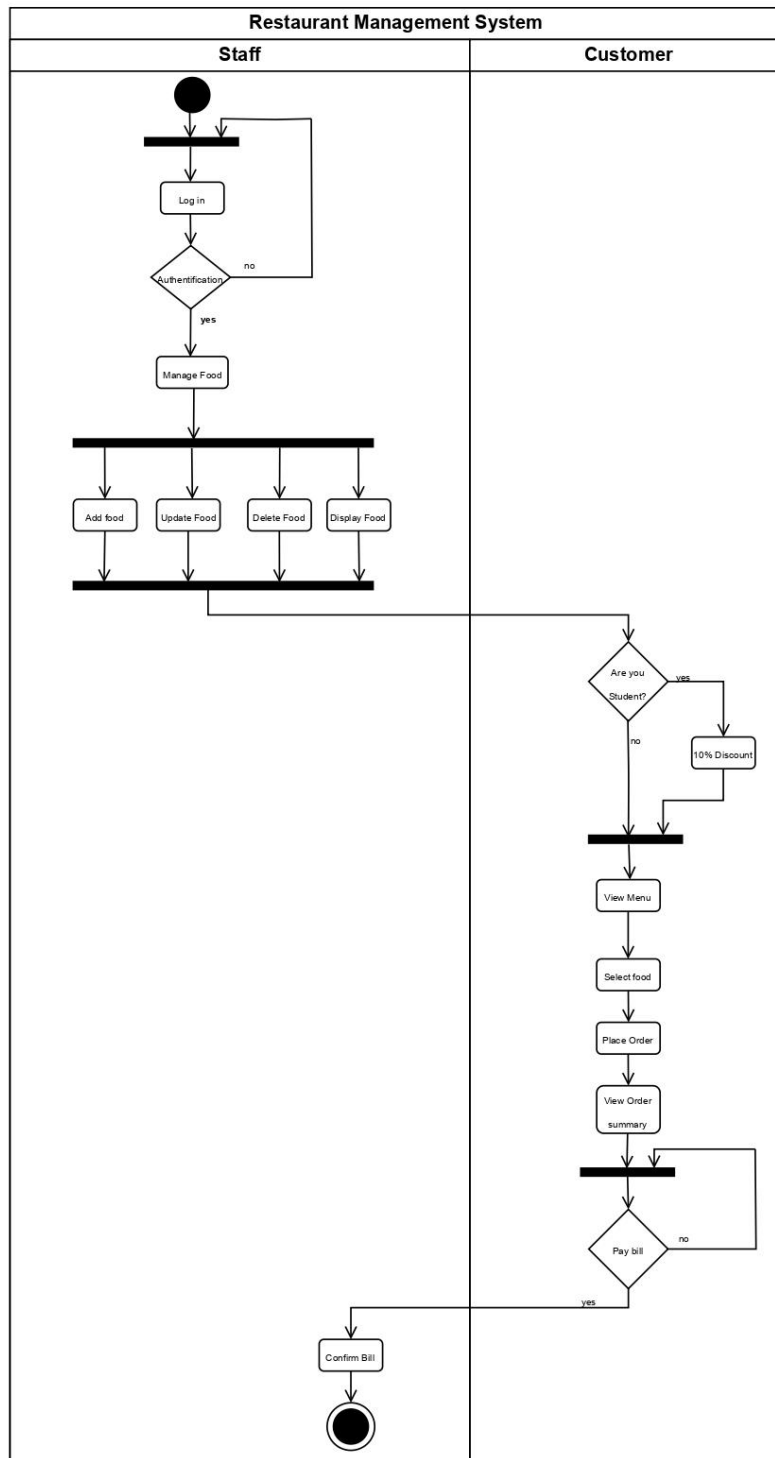
Use Case Diagram:



UML Diagram:



Activity Diagram:



Github Link: <https://github.com/Sayem666/Simple-Restaurant-Management-System/tree/main>

Future enhancement:

To address the limitations and further improve the system, the following enhancements are proposed:

Role-based Access Control: Implement a more robust role-based access control system where different users (staff, customer) have clearly defined and separate permissions.

Order Management Improvements: Allow customers to modify or cancel their orders. Introduce more detailed order statuses (e.g., pending, preparing, served, paid)

Inventory Management: Implement inventory tracking for food items to ensure items are in stock before allowing orders to be place.

Implementing these enhancements will greatly improve the functionality, security, and user experience of the Restaurant Management System.