|  |
| --- |
| **RESTAURANT MANAGEMENT SYSTEM** |

**ABSTRACT**

This project presents the design and implementation of a comprehensive Restaurant Management System (RMS) utilizing Database Management techniques. The system aims to streamline various operations within a restaurant, including order management, inventory tracking, employee scheduling, and customer relationship management.

The Restaurant Management System is designed to provide efficient and user-friendly functionalities for both restaurant staff and customers. It incorporates a robust database schema to store and manage diverse data sets, including menu items, customer orders, chef details, and inventory information.The system is designed using a relational database and object-oriented programming principles in Java.

**INTRODUCTION**

Restaurant management systems are intended to assist with the tasks associated with the day-to-day management of a restaurant or similar business.It helps restaurant owners manage their business more efficiently. The system can handle everything from menu creation to inventory management, table reservations, employee scheduling and payment processing. The project involves designing and implementing a database that stores all the necessary information about the restaurant. Such as menu items, prices, customer orders.

**FUNCTIONAL REQUIREMENTS**

* **MENU:**

Designing a menu for a restaurant management system involves listing the various food and beverage items available for customers to order. Here's a simple example of a restaurant menu

**Menu id:** A unique identifier for each menu item, typically assigned sequentially or using a specific coding system.

**Name:** The name or title of the menu item, providing a clear description of what it is.

**Description:** Additional information about the menu item, such as ingredients, cooking method, portion size, or any special instructions.

**Price:** The price of the menu item, indicating its cost to the customer.

* **TABLE:**

In a restaurant management system, the "Table" entity represents the physical tables available in the restaurant.

**Table id:** A unique identifier for each table.

**Capacity:** The number of seats available at the table.

* **ORDER:**

**Order id:** A unique identifier for each order.

**Table id:** (table\_id) REFERENCES table (table\_id)

**Menu id:** (menu\_id) REFERENCES menu (menu\_id)

**Quantity:** The "quantity" attribute denotes the numerical count of a specific item that is being ordered within a single order transaction in the restaurant.It represents the quantity of individual units or portions of an item that the customer requests.

**Status:** The "status" attribute indicates the current stage or state of an order within the restaurant.It provides valuable information regarding the progress of the order, allowing staff to efficiently manage order fulfillment and communicate effectively with customers.

* **RESERVATION:**

The "Reservation" entity represents the booking made by a customer to secure a table at the restaurant for a specific date and time.

**Reservation id:** A unique identifier for each reservation.

**Table id:** (table\_id) REFERENCES table (table\_id)

**Date:** The date of the reservation.

**Time:** The time of the reservation.

* **PAYMENT:**

The "Payment" entity represents the financial transaction between the customer and the restaurant for the goods and services provided.

**Payment id:** A unique identifier for each payment transaction.

**Payment date:** The date when the payment was made.

* **CHEF:**

**Chef id:** A unique identifier for each chef.

**Name:** The name of the chef.

**Speciality:** The area of culinary expertise or specialization of the chef (e.g., Italian cuisine, pastry chef).

* **INGREDIENT:**

**Ingredient id:** A unique identifier for each ingredient.

**Name:** The name of the ingredient.

**Quantity:**The current quantity of the ingredient available in the restaurant's inventory.

* **RECIPE:**

The "Recipe" entity represents the specific instructions and ingredients required to prepare a dish or menu item.

**Recipe id:** A unique identifier for each recipe.

**Name:** The name of the dish or menu item.

* **REVIEW:**

The "Review" entity represents feedback provided by customers about their dining experience.

**Review id**:A unique identifier for each review.

**Rating:**A numerical rating (e.g., on a scale of 1 to 5) indicating the overall satisfaction level.

**Comment:** Additional comments or feedback provided by the customer.

**Date:** The date when the review was submitted.

* **INVENTORY:**

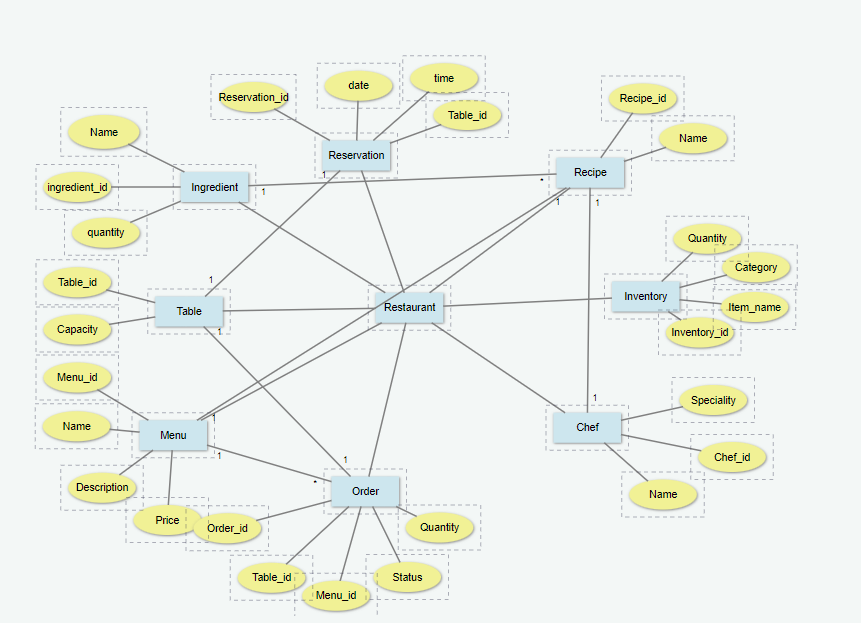
The "Inventory" entity represents the stock of ingredients, supplies, and other items used in the operation of the restaurant.

**Inventory id:**A unique identifier for each inventory item.

**Ingredient id:** The "ingredient ID" attribute in the inventory table of a restaurant management system refers to a unique identifier assigned to each ingredient tracked in the system.

**Quantity:** The "quantity" attribute indicates the current stock level or amount of a particular ingredient stored in the restaurant's inventory. It provides essential information about the availability of ingredients, enabling effective inventory management and ensuring that adequate stock is maintained to fulfill menu requirements.

**E-R DIAGRAM OF RESTAURANT MANAGEMENT SYSTEM**



**QUERIES TO CREATE TABLES IN THE DATABASE:**

* **Menu**

CREATE TABLE menu (

menu\_id INT PRIMARY KEY,

Name VARCHAR (45) NOT NULL,

Description TEXT NOT NULL,

Price DECIMAL (10,2) NOT NULL

);

* **Table**

CREATE TABLE table (

table\_id INT PRIMARY KEY,

Capacity VARCHAR (45) NOT NULL

);

* **Order**

CREATE TABLE order (

order\_id INT PRIMARY KEY,

menu\_id INT ,

table\_id INT ,

quantity INT ,

Status VARCHAR (45),

FOREIGN KEY (menu\_id) REFERENCES menu (menu\_id)

FOREIGN KEY (table\_id) REFERENCES table (table\_id)

);

* **Reservation**

CREATE TABLE reservation (

Reservation\_id INT PRIMARY KEY,

table\_id INT (10) NOT NULL,

Date DATE,

Time VARCHAR (45)

FOREIGN KEY (table\_id) REFERENCES table (table\_id)

);

* **Payment**

CREATE TABLE payment (

Payment\_id INT PRIMARY KEY,

amount INT NOT NULL,

Payment\_date DATETIME

);

* **Chef**

CREATE TABLE chef (

chef\_id INT PRIMARY KEY,

Name VARCHAR(45) NOT NULL,

Speciality VARCHAR(45) NOT NULL

);

* **Ingredient**

CREATE TABLE restaurantdb.ingredient (

ingredient\_id INT PRIMARY KEY,

name VARCHAR (45),

quantity VARCHAR (45)

);

* **Recipe**

CREATE TABLE recipe (

recipe\_id INT PRIMARY KEY,

Name VARCHAR (45) NOT NULL

);

* **Review**

CREATE TABLE review (

review\_id INT PRIMARY KEY,

Rating FLOAT NOT NULL,

Comment VARCHAR (45) NOT NULL,

Date DATETIME NOT NULL

);

* **Inventory**

CREATE TABLE inventory (

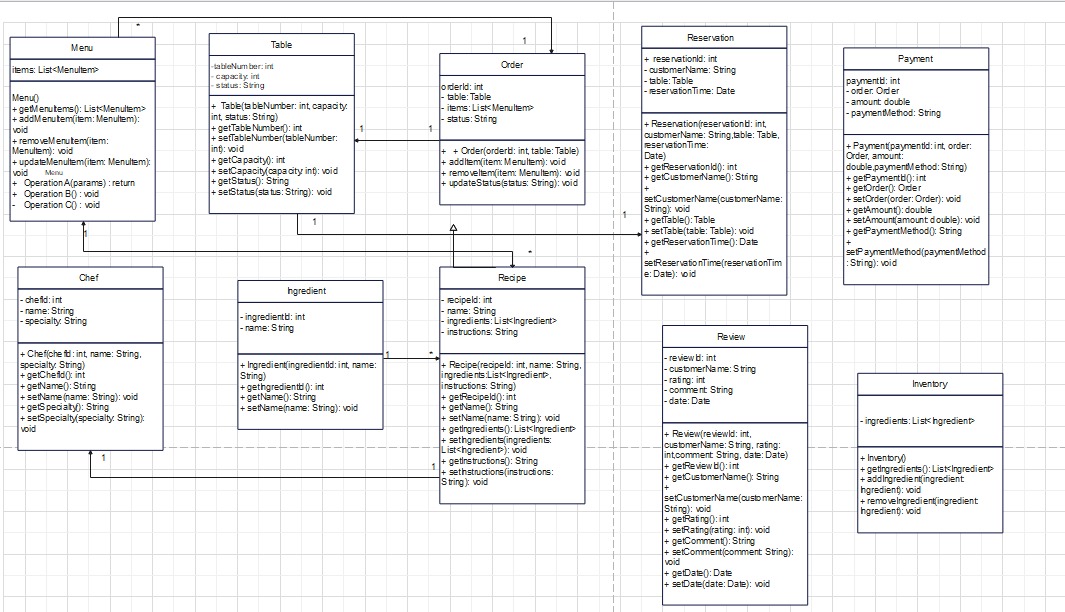
inventory\_id INT PRIMARY KEY,

ingredient\_id INT NOT NULL,

quantity INT NOT NULL

);

**UML DIAGRAM**

****

**ALL CLASSES JAVA FILE CODES**

* **Ingredient.java**

Package restaurant;

public class Ingredient {

private int ingredientID;

private String name;

private int quantity;

public Ingredient(int ingredientID, String name, int quantity) {

this.ingredientID = ingredientID;

this.name = name;

this.quantity = quantity;

}

public int getIngredientID() {

return ingredientID;

}

public void setIngredientID(int ingredientID) {

this.ingredientID = ingredientID;

}

public String getName() {

return name;

}

public void setName(String name) {

this.name = name;

}

public int getQuantity() {

return quantity;

}

public void setQuantity(int quantity) {

this.quantity = quantity;

}

}

* **Reservation.java**

Package restaurant;

import java.time.LocalDate;

import java.time.LocalTime;

public class Reservation {

private int reservationID;

private int tableID;

private LocalDate date;

private LocalTime time;

public Reservation(int reservationID, int tableID, LocalDate date, LocalTime time) {

this.reservationID = reservationID;

this.tableID = tableID;

this.date = date;

this.time = time;

}

public int getReservationID() {

return reservationID;

}

public void setReservationID(int reservationID) {

this.reservationID = reservationID;

}

public int getTableID() {

return tableID;

}

public void setTableID(int tableID) {

this.tableID = tableID;

}

public LocalDate getDate() {

return date;

}

public void setDate(LocalDate date) {

this.date = date;

}

public LocalTime getTime() {

return time;

}

public void setTime(LocalTime time) {

this.time = time;

}}

* **Table.java**

Package restaurant;

public class Table {

private int tableID;

private int capacity;

public Table (int tableID, int capacity) {

this.tableID = tableID;

this.capacity = capacity;

}

public int getTableID() {

return tableID;

}

public void setTableID(int tableID) {

this.tableID = tableID;

}

public int getCapacity() {

return capacity;

}

public void setCapacity(int capacity) {

this.capacity = capacity;

}}

* **Menu.java**

Package restaurant;

public class MenuItem {

private int menuID;

private String name;

private String description;

private double price;

public MenuItem(int menuID, String name, String description, double price) {

this.menuID = menuID;

this.name = name;

this.description = description;

this.price = price;

}

public int getMenuID() {

return menuID;

}

public void setMenuID(int menuID) {

this.menuID = menuID;

}

public String getName() {

return name;

}

public void setName(String name) {

this.name = name;

}

public String getDescription() {

return description;

}

public void setDescription(String description) {

this.description = description;

}

public double getPrice() {

return price;

}

public void setPrice(double price) {

this.price = price;

}

}

* **Order.java**

Package restaurant;

public class Order {

private int orderID;

private int tableID;

private int menuID;

private int quantity;

private String status;

public Order(int orderID, int tableID, int menuID, int quantity, String status) {

this.orderID = orderID;

this.tableID = tableID;

this.menuID = menuID;

this.quantity = quantity;

this.status = status;

}

public int getOrderID() {

return orderID;

}

public void setOrderID(int orderID) {

this.orderID = orderID;

}

public int getTableID() {

return tableID;

}

public void setTableID(int tableID) {

this.tableID = tableID;

}

public int getMenuID() {

return menuID;

}

public void setMenuID(int menuID) {

this.menuID = menuID;

}

public int getQuantity() {

return quantity;

}

public void setQuantity(int quantity) {

this.quantity = quantity;

}

public String getStatus() {

return status;

}

public void setStatus(String status) {

this.status = status;

}

}

* **Chef.java**

Package restaurant;

public class Chef {

private int chefID;

private String name;

private String specialty;

public Chef(int chefID, String name, String specialty) {

this.chefID = chefID;

this.name = name;

this.specialty = specialty;

}

public int getChefID() {

return chefID;

}

public void setChefID(int chefID) {

this.chefID = chefID;

}

public String getName() {

return name;

}

public void setName(String name) {

this.name = name;

}

public String getSpecialty() {

return specialty;

}

public void setSpecialty(String specialty) {

this.specialty = specialty;

}

}

* **Recipe.java**

Package restaurant;

public class Recipe {

private int recipeID;

private String name;

public Recipe(int recipeID, String name) {

this.recipeID = recipeID;

this.name = name;

}

public int getRecipeID() {

return recipeID;

}

public void setRecipeID(int recipeID) {

this.recipeID = recipeID;

}

public String getName() {

return name;

}

public void setName(String name) {

this.name = name;

}

}

* **Review.java**

Package restaurant;

import java.time.LocalDate;

public class Review {

private int reviewID;

private int rating;

private String comment;

private LocalDate date;

public Review(int reviewID, int rating, String comment, LocalDate date) {

this.reviewID = reviewID;

this.rating = rating;

this.comment = comment;

this.date = date;

}

public int getReviewID() {

return reviewID;

}

public void setReviewID(int reviewID) {

this.reviewID = reviewID;

}

public int getRating() {

return rating;

}

public void setRating(int rating) {

this.rating = rating;

}

public String getComment() {

return comment;

}

public void setComment(String comment) {

this.comment = comment;

}

public LocalDate getDate() {

return date;

}

public void setDate(LocalDate date) {

this.date = date;

}

}

}

* **Inventory.java**

Package restaurant;

import java.time.LocalDate;

public class Inventory {

private int inventoryID;

private int ingredientID;

private int quantity;

private LocalDate lastUpdated;

public Inventory(int inventoryID, int ingredientID, int quantity, LocalDate lastUpdated) {

this.inventoryID = inventoryID;

this.ingredientID = ingredientID;

this.quantity = quantity;

this.lastUpdated = lastUpdated;

}

public int getInventoryID() {

return inventoryID;

}

public void setInventoryID(int inventoryID) {

this.inventoryID = inventoryID;

}

public int getIngredientID() {

return ingredientID;

}

public void setIngredientID(int ingredientID) {

this.ingredientID = ingredientID;

}

public int getQuantity() {

return quantity;

}

public void setQuantity(int quantity) {

this.quantity = quantity;

}

public LocalDate getLastUpdated() {

return lastUpdated;

}

public void setLastUpdated(LocalDate lastUpdated) {

this.lastUpdated = lastUpdated;

}

}

* **Payment.java**

Package restaurant;

import java.time.LocalDate;

public class Payment {

private int paymentID;

private int orderID;

private double amount;

private LocalDate paymentDate;

public Payment(int paymentID, int orderID, double amount, LocalDate paymentDate) {

this.paymentID = paymentID;

this.orderID = orderID;

this.amount = amount;

this.paymentDate = paymentDate;

}

public int getPaymentID() {

return paymentID;

}

public void setPaymentID(int paymentID) {

this.paymentID = paymentID;

}

public int getOrderID() {

return orderID;

}

public void setOrderID(int orderID) {

this.orderID = orderID;

}

public double getAmount() {

return amount;

}

public void setAmount(double amount) {

this.amount = amount;

}

public LocalDate getPaymentDate() {

return paymentDate;

}

public void setPaymentDate(LocalDate paymentDate) {

this.paymentDate = paymentDate;

}

}