

Aim: To design a Class Diagram to represent the structural view of the system.

Class Diagram 1: Search Restaurant

Class 1: Filter

The Filter class represents a filtering mechanism for restaurant searches.

Attributes:

- FilterId
- FilterName
- PriceRange

Operations:

- ApplyFilter(): Applies the specified filter criteria.
- RemoveFilter(): Removes the applied filters.

Class 2: FoodCategory

The FoodCategory class represents different categories of food items.

Attributes:

- Category
- Name
- Information
- Attribute1
- PopularItems

Operations:

- getDescription(): Gets the description of the food category.
- getFoodItem(): Gets the list of food items in the category.
- SearchByCategory(): Searches for restaurants based on a specific food category.
- AddToCategory(): Adds a new food item to the category.
- ViewPopularItems(): Displays popular items in the category.
- SearchByInformation(): Searches for restaurants based on information.
- SearchByPopularity(): Searches for restaurants based on popularity.

Class 3: Review

The Review class represents user reviews for restaurants.

Attributes:

- Reviews

- Rating
- Feedback_form

Operations:

- Addreviews(): Adds a review to the restaurant.
- AddRating(): Adds a rating to the restaurant.

Class 4: SafetyGuidelines

The SafetyGuidelines class represents safety guidelines for restaurants.

Attributes:

- Rating
- Review
- FeedbackForm
- Hygiene

Operations:

- ViewGuidelines(): Displays safety guidelines.
- SubmitReview(): Allows users to submit reviews related to safety.

Class 5: SearchRestaurant

The SearchRestaurant class represents the main functionality for searching and filtering restaurants.

Attributes:

- RestuarantId
- RName
- Rating
- Website
- Certifications
- CuisineType
- OpeningHours
- ContactInformation
- Location

Operations:

- FilterByRating(): Filters restaurants based on rating.
- FilterByCuisine(): Filters restaurants based on cuisine type.
- SearchRestaurant(): Performs a general restaurant search.
- SearchByPrice(): Searches for restaurants based on price.
- SearchByRating(): Searches for restaurants based on rating.
- BookTable(): Allows users to book a table.
- FilterByLocation(): Filters restaurants based on location.

Class 6: User

The User class represents user profiles and actions.

Attributes:

- UserId
- UserName
- Password
- UserType
- OrderHistory

Operations:

- SearchRestaurent(): Searches for restaurants.
- ViewProfile(): Views the user's profile.
- SubmitFeedback(): Submits feedback for a restaurant.
- ModifyProfile(): Modifies user profile information.
- ViewRestaurant(): Views restaurant details.
- AddToFavourites(): Adds a restaurant to the user's favorites.
- RequestAccountDeletion(): Initiates the account deletion process.

Class 7: FoodItemType

The FoodItemType class represents different types of food items.

Attributes:

- ItemId
- Fname
- categoryId
- Price
- Ingredients
- DietaryInformation
- ServingSize
- NutrientValue

Operations:

- ViewNutrientValue(): Views the nutrient value of the food item.
- ViewProteinvalue(): Views the protein value of the food item. ●
- GetTotalAmt(): Gets the total amount for the food item.
- ViewIngrediants(): Views the ingredients of the food item.
- ViewCalories(): Views the calories of the food item.
- SetServingSize(): Sets the serving size for the food item.

Class Diagram 2:Payment System:

Class 1: Booking

The Booking class represents the reservation and booking details for a restaurant.

Attributes:

- BookingId
- UserId
- PaymentOptions
- TotalAmount
- RestaurantID
- Status
- GuestCount

Operations:

- CalculateAmount(): Calculates the total amount for the booking.
- UpdateBookingStatus(): Updates the status of the booking. ●
- ConfirmBooking(): Confirms the restaurant booking.
- ApplyPromoCode(): Applies a promotional code to the booking.
- ViewReservedTable(): Views the reserved table details.
- SplitPayment(): Allows for the split payment of the booking.
- CheckAvailability(): Checks the availability of tables.
- UpdateGuestCount(): Updates the guest count for the booking.

Class 2: User

The User class represents user profiles and actions related to bookings.

Attributes:

- Userid
- Name
- Password

- UserType
- BookingHistory

Operations:

- MakeBooking(): Initiates the process of making a new booking.
- ViewBooking(): Views the booking history for a user.
- CancelBooking(): Allows users to cancel an existing booking. ●
- RescheduleBooking(): Enables users to reschedule an existing booking.

Class 3: Delivery

The Delivery class represents details and operations related to food delivery.

Attributes:

- delieveryId
- delieveryDate
- DeliveryAddress
- delieveryStatus

Operations:

- scheduleDelivery(): Schedules the delivery of an order.
- updateStatus(): Updates the delivery status.
- getDeliveryStatus(): Retrieves the current delivery status.

Class 4: Payment

Summary:

The Payment class represents payment details and operations.

Attributes:

- PaymentId
- UserId
- Amount
- Method
- PaymentDate
- Payment_description

Operations:

- ProcessPayment(): Processes the payment for a booking.
- PrintReceipt(): Prints a receipt for the payment.
- AddPaymentMethod(): Adds a new payment method.
- DeletePaymentMethod(): Deletes an existing payment method.

Class 5: OrderItem

The OrderItem class represents details of items ordered as part of a booking.

Attributes:

- OrderId
- bookingId
- quantity
- TotalAmt

Operations:

- CalculateAmt(): Calculates the total amount for the ordered items.
- getQuantity(): Retrieves the quantity of items ordered.
- These class scenarios collectively provide a comprehensive representation of the booking, user interactions, delivery, payment, and ordered items within a restaurant management system.

Class Diagram 3: UserManagementSystem:

Class 1: Admin

The Admin class extends the Database class and represents administrators in the system.

Attributes:

- UserName
- Password
- Location

Operations:

- UpdatePortal(): Updates the admin portal.
- Operation2(): Placeholder operation.
- UpdateDatabase(): Updates the database.
- Permission(): Manages permissions for administrators.
- Rating(): Manages restaurant ratings.

Class 2: Backend

The Backend class represents the backend system with user registration status and location.

Attributes:

- UserName
- RegistrationStatus

- Location

Class 3: RestaurantGuide

The RestaurantGuide class extends the Database class and represents a restaurant guide with additional functionalities.

Attributes:

- UserName
- Password
- MobileNo
- Comission

Operations:

- OpeningActivity(): Manages the opening activities of a restaurant.
- CloseActivity(): Manages the closing activities of a restaurant.
- ManageReservations(): Handles restaurant reservations.
- ManageOnlineBookings(): Manages online bookings for a restaurant.
- GetComission(): Retrieves commission details.

Class 4: Report

The Report class represents reports with details related to employees.

Attributes:

- ReportId
- EmployeeId
- ReportDetails

Class 5: CityEmployee

The CityEmployee class extends the Database class and represents employees working in a city.

Attributes:

- UserName
- Password
- MobileNo
- Location

Operations:

- UpdateWebsite(): Updates the city website.
- UpdateCustomerDetails(): Updates customer details.
- ProvideInformation(): Provides information about the city.

- ReviewFeedback(): Reviews and manages feedback.

Class 6: Database

The Database class represents a database with information related to restaurant types, descriptions, and user details.

Attributes:

- RestaurantType
- RestaurantDescription
- UserDetails

Operations:

- AddRecord(): Adds a record to the database.
- ModifyRecord(): Modifies a record in the database. ●
- DeleteRecord(): Deletes a record from the database. ●
- AddRestaurant(): Adds restaurant information to the database.

Result: Class diagram has been designed and studied.