Restaurant Ordering System

Francis Quang

912679019

VtecFrancis

Checkpoint #	Date Submitted
Checkpoint 1	2/20/2024
Checkpoint 2	3/5/2024

Table of Contents

Project Description

- **The motivation**
  - Our restaurant ordering system's motivation is enhancing customer pleasure by taking eating experiences to new levels of convenience and customization. Diners value efficiency and convenience of service just as much as meal quality in today's fast-paced society. The system is made to simplify all aspects of restaurant operations, including food service and table reservations, so that every encounter is smooth, satisfying, and customized to each customer's preferences. With the ability for customers to pre-order their meals, the system combines sophisticated reservation features with the goal of reducing wait times, accurately matching kitchen operations to customer demand, and creating a smooth connection between customer expectations and the restaurant's service delivery. The goal of this simultaneous emphasis on customer-centric features and operational efficiency is to create a devoted clientele, encourage repeat business, and gain a competitive advantage in the fast-paced restaurant industry.
- **High-level Description:**
  - The proposed restaurant ordering system is a creative innovation designed to completely change how restaurants interact with their customers. The opportunity to reserve a table and the choice to pre-order food for dine-in service are the system's two primary features. customers can reserve a table with their choice of date and time guaranteeing they will have a seat waiting for them. customers can choose their meals in advance by a digital menu with food descriptions and photos. The pre-order options will cut down on wait times and guarantee that every dish is served on time. This system intends to speed up operations for the restaurant by giving real-time insights into reservation scheduling and pre-order details. Better staffing levels, efficient kitchen operations, and efficient table management are made possible by this, guaranteeing that resources are employed effectively and that customers enjoy attentive service.

- **High level description of unique features:**
  - **Order ahead of time:** This feature is the same as a pick up however the key difference will be for large parties and can order ahead of time so there will be less wait times and accurate orders in a timely fashion.
  - **Customer loyalty:** The system will offer personalized recommendations to customers based on their previous orders and preferences. customers can be members of our loyalty system making new customers into regulars. They can enter their phone number or email and sign up to earn points for discounts on menu items restaurants offer.
- **At least two existing products that would benefit from using our database**
  - **Lunchbox restaurant management tool:** The main use of lunchbox will benefit customers who are able to order from their table to the kitchen using a digital menu accessed from their devices. customers will use our pre-order feature. customers can also provide special instructions or modifications to menu items. For the restaurant's benefit, they can manage orders, change menu items, and manage guests all in real-time. With so much control this will be a powerful tool that will benefit in our database. This management tool pairs with major point of sale software such as Toast.
  - **Toast point of sale & loyalty program:** Toast offers a software for points that restaurants can customize the promotions. customers will earn points upon ordering from the restaurant and will reward free promotions on birthdays. The loyalty software can only be used with toast's POS. Toast also comes with contactless payments and online ordering.
  - **Opentable:** Opentable will be a great benefit to our database with their easy to use platform that features guest profiles, floor planning, timeline view, shift planning to optimize operations and real-time alerts such as payments by customers.

Functional Database Requirements

1. **Customer**
    1.1. Customers can register with many accounts
    1.2. Customers shall have exactly one first name
    1.3. Customers shall have exactly one last name
    1.4. Customers shall have at least one phone number
    1.5. Customers shall have at least one email
    1.6. Customers shall be able to update their personal information
    1.7. Customers shall place at least one order at a time
    1.8. Customers shall be able to rate at least one food item
    1.9. Customers shall be able to review at least one food item
    1.10. Customers shall be able to rate at least one restaurant
    1.11. Customers shall be able to review at least one restaurant
    1.12. Customers shall be able to repeat at least one previous order
    1.13. Customers shall save at least one menu item
    1.14. Customers shall be able to pre-order at least one food for dine-in
    1.15. Customers shall be able to pre-order at least one food for pickup
    1.16. Customers shall have at least one payment method
    1.17. Customers shall have at most one pay now option
    1.18. Customers shall have at most one pay in person option
    1.19. Customers shall be able reserve exactly one table with a date, time, and number of guests
    1.20. Customers shall be able to manage exactly one reserved table
    1.21. Customers shall have at least one accessibility requests
    1.22. Customers must reserve exactly one table to place a pre-order for dine-in
2. **User**
    2.1. A user shall have exactly one first name
    2.2. A user can have at most one middle name
    2.3. A user shall have exactly one last name
    2.4. A user shall have exactly one email
    2.5. A user shall create a password

3. Servers

3.1. Servers shall receive at least one new orders

3.2. Servers shall manage at least one new changes to existing orders

3.3. Servers shall update at least one status of orders

3.4. Servers shall enter at least one customer orders

3.5. Servers shall manage at least one table reservation

3.6. Servers shall assign customers to at least one table

3.7. Servers shall manage at least one waiting lists

4. Chef

4.1. Chefs shall receive at least one new order

4.2. Chefs shall updates at least one existing orders

4.3. Chefs shall update at least one status orders

5. Account

5.1. An account shall be created with exactly one user

5.2. An account can store at least one payment option

5.3. An account can be logged simultaneously on at least one device

5.4. An account can sign up for exactly one loyalty program

6. Admin

6.1. Admins shall manage at least one table reservation

6.2. Admins shall manage at least one user accounts

6.3. Admins shall set role permission to at least one accounts

7. Orders

7.1. An order shall contain at least one menu item

7.2. An order shall allow at least one special instructions for customers

7.3. An order shall be linked to exactly one customer’s account

7.4. An order shall have at least one updated status

8. Loyalty program

8.1. Loyal customers shall redeem at least one point for discounts

8.2. Loyal customers shall receive exactly one free menu item on their birthday

8.3. Loyal customers shall redeem multiple points via email

8.4. Loyal customers shall redeem multiple points via phone number

9. Menu

9.1. Menu shall create at least one special menu item for special occasions

9.2. Menu shall allow staff add at least one menu categories

9.3. Menu allow staff to delete at least one menu categories

9.4. Menu shall staff to update at least one menu categories

10. Menu item

10.1. Menu items shall provide exactly one name of item

10.2. Menu items shall provide exactly one detailed description

10.3. Menu items shall provide exactly one price for each food item

11. Reservation

11.1. Booking shall have exactly one customer's first name

11.2. Booking shall have at most one middle name

11.3. Booking shall have exactly one last name

11.4. Booking shall have exactly one customer's email

11.5. Booking shall have exactly one customer's phone number

11.6. Booking shall have at least one time of reservation

11.7. Booking shall have at least one date of reservation

11.8. Booking shall have multiple updates of reservations

11.9. Booking shall have at least one number of guests in the reservation

11.10. Booking shall have customers exactly one number in line if tables are full

12. Tables

12.1. Tables shall allow customers to make reservations at most one date

12.2. Table shall allow customers to make reservations at most one time

12.3. Tables shall allow customer to make reservation with at least one guest

12.4. Tables shall allow staffs to manage tables

12.5. Tables shall assign at most one waitlist number to customers if reservations are fully booked

13. Payment transaction

13.1. Payments transactions shall support at least one payment methods

13.2. Payment transaction shall provide receipts for customers that completed a transaction

13.3. Payment transaction shall apply at most one promotion offers to loyal customers

13.4. Payment transaction shall generate at least one total amount after closing

13.5. Payment transaction shall generate at least one total tip amount after closing

13.6. Payment transaction shall record at least one item ordered with price

13.7. Payment transaction shall allow servers to issue at least one refund

13.8. Payment transaction shall allow servers to adjust at least one charge

13.9. Payment transaction shall record at least one charge adjustments

14. Recipe

14.1. Recipe system shall allow staff to manage at least one recipes

14.2. Recipes shall be associated with exactly one inventory item

14.3. Recipe system shall calculate at least one cost of ingredients

15. Inventory

15.1. Inventory shall record multiple supply cost

15.2. Inventory shall record at least one delivery date

15.3. Inventory shall record at least one time of delivery

16. Feedback

16.1. Customers shall be able to rate multiple restaurants

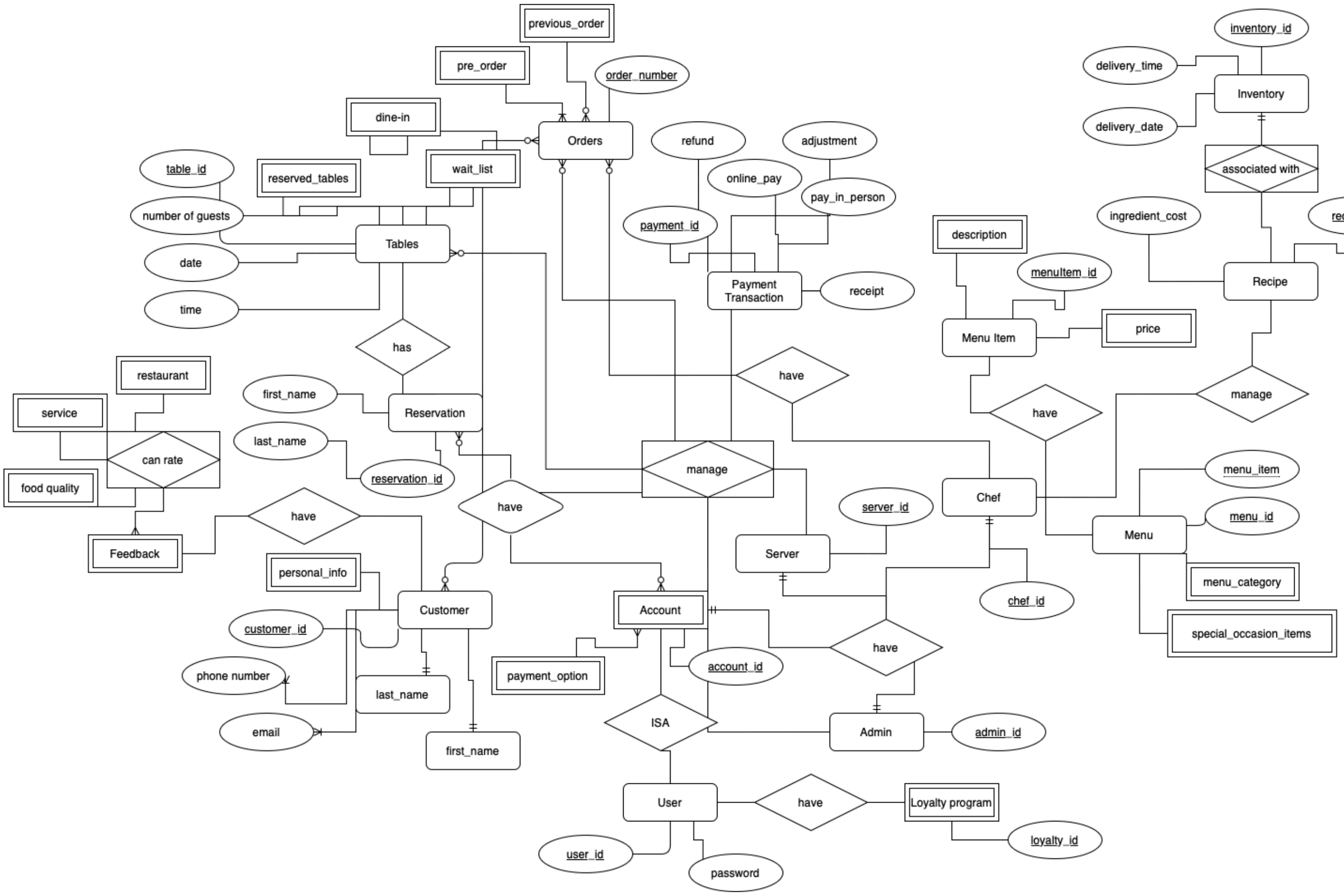
16.2. Customers shall be able to rate multiple services

16.3. Customers shall be able to rate multiple food qualities

Non-functional Database Requirements

1. **Security**
    - 1.1. Servers, admins, and chefs shall login with a unique username and password to access employee systems
    - 1.2. Customers shall login with a unique username and password to access table reservations and ordering from the menu
    - 1.3. Users must have a minimum length of 8 characters that include alphabets and numbers
    - 1.4. Sensitive data shall be encrypted into the database system
    - 1.5. Payments shall have a secure transaction
    - 1.6. Payment transactions shall be recorded
2. **Usability**
    - 2.1. The database system shall be easy to use for customers to navigate to menus
    - 2.2. The database systems shall be easy for servers and chefs to navigate to their current position and schedule
    - 2.3. The database system shall have multilingual support
3. **Performance**
    - 3.1. The database system shall have page menu loading times under 2 seconds
    - 3.2. The database system shall receive order submissions under 1 second
    - 3.3. The database system shall support high number of users simultaneously
4. **Reliability**
    - 4.1. Payments shall have a reliable transaction
    - 4.2. The database system shall be available during open hours
    - 4.3. The database system shall ensure menu prices and order information
    - 4.4. The database system shall conduct regular backups
    - 4.5. The database system shall alert admins to handle errors
    - 4.6. The database system shall apply updates without disrupting the service
5. **Privacy**
    - 5.1. Privacy of customers shall collect only necessary data for ordering and managing customer’s accounts
    - 5.2. The system shall state users the type of data that will be collected and the reason for collecting data

Entity Relationship Diagram (ERD)



Entity Description

1. **Customer (Strong)**
    1.1. customer_id: key, numeric
    1.2. first_name: single-value, alphanumeric
    1.3. last_name: single-value, alphanumeric
    1.4. phone_numbers: multivalue, alphanumeric
    1.5. emails: multivalue, alphanumeric
    1.6. personal_information: composite, alphanumeric
    1.7. payment_methods: multivalue, alphanumeric
    1.8. pay_now_option: single-value, boolean
    1.9. pay_in_person_option: single-value, boolean
    1.10. accessibility_requests: multivalue, text
2. **User (Strong)**
    2.1. user_id: key, numeric
    2.2. first_name: single-value, alphanumeric
    2.3. middle_name: single-value, alphanumeric
    2.4. last_name: single-value, alphanumeric
    2.5. email: single-value, alphanumeric
    2.6. password: single-value, text
3. **Server (Strong)**
    3.1. server_id: key, numeric
    3.2. orders_managed: multivalue, numeric
    3.3. tables_managed: multivalue, numeric
    3.4. waiting_lists_managed: multivalue, text
4. **Chef (Strong)**
    4.1. chef_id: key, numeric
    4.2. orders_received: multivalue, numeric
5. **Account (Weak)**
    5.1. account_id: key, numeric
    5.2. user_id: foreign key, numeric
    5.3. payment_options: multivalue, alphanumeric
    5.4. loyalty_program_status: single-value, boolean

6. **Admin (Strong)**
    6.1. admin_id: key, numeric
    6.2. user_accounts_managed: multivalue, numeric
    6.3. role_permissions_set: multivalue, text
7. **Order (Strong)**
    7.1. order_id: key, numeric
    7.2. menu_items: multivalue, numeric
    7.3. special_instructions: multivalue, text
    7.4. customer_account_id: foreign key, numeric
    7.5. status_updates: multivalue, text
8. **Loyalty Program (Weak)**
    8.1. loyalty_id: key, numeric
    8.2. account_id: foreign key, numeric
    8.3. points_redeemable: multivalue, numeric
    8.4. birthday_offer: single-value, boolean
9. **Menu (Strong)**
    9.1. menu_id: key, numeric
    9.2. special_items: multivalue, numeric
    9.3. categories: multivalue, text
10. **MenuItem (Strong)**
    10.1. menu_item_id: key, numeric
    10.2. name: single-value, alphanumeric
    10.3. description: single-value, text
    10.4. price: single-value, numeric
11. **Reservation (Strong)**
    11.1. reservation_id: key, numeric
    11.2. customer_id: foreign key, numeric
    11.3. time: single-value, timestamp
    11.4. date: single-value, date
    11.5. number_of_guests: single-value, numeric
    11.6. status_updates: multivalue, text
12. **Table (Strong)**
    12.1. table_id: key, numeric

12.2. customer_id: foreign key, numeric

12.3. reservation_date: single-value, date

12.4. reservation_time: single-value, timestamp

12.5. guests: single-value, numeric

12.6. status: single-value, text

13. Payment Transaction (Strong)

13.1. transaction_id: key, numeric

13.2. payment_methods: multivalue, alphanumeric

13.3. receipt_issued: single-value, boolean

13.4. discounts_applied: multivalue, numeric

13.5. promotion_offers_applied: multivalue, numeric

13.6. total_amount: single-value, numeric

13.7. total_tip_amount: single-value, numeric

13.8. items_ordered_with_price: composite, alphanumeric

14. Recipe (Strong)

14.1. recipe_id: key, numeric

14.2. inventory_items: multivalue, numeric

14.3. cost_of_ingredients: single-value, numeric

15. Inventory (Strong)

15.1. inventory_id: key, numeric

15.2. delivery_dates: multivalue, date

15.3. delivery_time: numeric

16. Feedback (Weak)

16.1. feedback_id: key, numeric

16.2. customer_id: foreign key, numeric

16.3. restaurant_ratings: multivalue, numeric

16.4. service_ratings: multivalue, numeric

16.5. food_quality_ratings: multivalue, numeric