Kellam, Dominique N.

CSC 340 Fall, 2023

Pizza order System

Final Report

Table of Contents

[I.Introduction 2](#_Toc148601232)

[A.Problem Statement 2](#_Toc148601233)

[B.System Proposal 2](#_Toc148601234)

[II.System Description 2](#_Toc148601235)

[III.System Requirements 2](#_Toc148601236)

[A.Functional Requirements 2](#_Toc148601237)

[B.Non-Functional Requirements 2](#_Toc148601238)

[IV.Use Case Diagram 2](#_Toc148601239)

[V.Class Diagram 2](#_Toc148601240)

[VI.Sequence Diagrams 2](#_Toc148601241)

[VII.Activity Diagram 3](#_Toc148601242)

[VIII.State Diagrams 4](#_Toc148601243)

[IX.Database Design 4](#_Toc148601244)

[A.ER Diagram 4](#_Toc148601245)

[B.Table Schema 4](#_Toc148601246)

[X.Conclusion 4](#_Toc148601247)

# Introduction

The purpose of the online pizza ordering system is to provide Dom’s Pizza with a user-friendly and efficient platform for customers to order pizzas online. This system aims to streamline the pizza ordering process, enhance customer satisfaction, and boost the overall efficiency of Dom’s Pizza's operations.

## Problem Statement The Dom’s Pizza needs a system for their customers to order pizzas online.

## System Proposal We propose a software system to help the store with online orders.

# System Description

The system includes three subsystems: an online ordering subsystem, an order management subsystem, and a stock management subsystem. The online ordering subsystem allows customers to order pizzas online. The orders shall be kept in a database. The staffs of the store can then handle the orders with the order management subsystems. The status of an order should be kept in the database. The customers should be able to trace their orders online. The stock management subsystem is for managing the stock of raw materials and sauces. It will issue a warning message to remind the staffs when the stack of an item is low. The staffs will then find a supplier the store with the item.

# System Requirements

## Functional Requirements

1. The system shall allow a customer to create a member account.
   1. The customer shall press the “Create Account/Login” button.
   2. The system shall display a new member form.
   3. The customer shall enter their name, address, phone number, email as username password.   
      A screenshot of a computer

      Description automatically generated
   4. The customer shall press the Submit button.
   5. The system shall check if the email is already in the database.
      1. If the email exists, the system shall display error message to request the customer to enter a different email address. Go to 1.3.  
         A screenshot of a computer

         Description automatically generated
      2. If the email does not exist, save the data to the database. Display a confirmation message.   
         A screenshot of a computer

         Description automatically generated
   6. Go to main menu.
2. The system shall allow a customer/member to order pizzas.
   1. The customer shall press “Order Pizza” button.
   2. The system shall display a pizza form. A pizza on a table

      Description automatically generated with medium confidence
   3. The customer shall press Add to Cart button.
   4. The system shall add the item to the cart.
   5. The system shall display a confirmation message to the customer.
      1. A screenshot of a computer

         Description automatically generated
   6. The customer shall press the “Check Out” button.
   7. The system shall display the contents of the cart in a cart form. A screenshot of a computer

      Description automatically generated
   8. The customer shall enter their name on card, card number, card expiration date, zip code and CVV number.
   9. The customer shall press the “Submit” button.
   10. The system shall display a confirmation message to the customer with their order number.  
        A screenshot of a computer error

       Description automatically generated
   11. Go to main menu.
3. The system shall allow a customer to trace the status of their order.
   1. The customer shall press the “Check Order Status” button.
   2. The customer shall enter their order ID number.
   3. The customer shall click the “Search” button.
   4. The system shall display the Order Status form.  
      A screenshot of a computer

      Description automatically generated
   5. Go to main menu.
4. The system shall allow a member to edit their profile.
   1. The member shall press the “Edit Account” button.
   2. The system shall display an edit account form.  
      A screenshot of a computer

      Description automatically generated
   3. The system shall display and confirmation message.  
      A screenshot of a message

      Description automatically generated
   4. Go to main menu.
5. The system shall allow the member to view their order history.
   1. The member shall press the “Check Order History” button.
   2. The member shall enter their member ID number.
   3. The member shall press the “Search” button.
   4. The system shall display the Order History form. A screenshot of a search box

      Description automatically generated

## Non-Functional Requirements

1. The length of password for a member account should be at least 8 characters long.
2. The name, address, phone number and email address should not be null.

# Use Case Diagram

Five use cases have been identified for the system:

### Customer creates an account, then becomes a member.

### Customer orders a pizza(s).

### Customer views the status of their order.

### Customer/Member edit’s their profile.

### Customer/Member views their order history.

A black and white background with white text

Description automatically generated

# Class Diagram

1. Member subclass inherits Customer superclass.
2. Customer class has a many-to-many relationship with Pizza class.
3. Pizza class has a many-to-many relationship with Orders class.
4. Payment class has a 1-to-1 relationship with the Orders class.

A screenshot of a diagram

Description automatically generated

# Sequence Diagrams

### Sequence diagram for “Create Account” use case:



### Sequence diagram for “Order Status” use case:

A screenshot of a computer

Description automatically generated

### Sequence diagram for “Order Pizza” use case:

A screenshot of a computer

Description automatically generated

### Sequence diagram for “Order history” use case:

A screenshot of a computer

Description automatically generated

### Sequence diagram for “Edit Profile” use case:

A screenshot of a computer

Description automatically generated

# Activity Diagram

### Activity diagram for the checkEmail() function in the “Create Account” sequence diagram.

A screenshot of a computer

Description automatically generated

### Activity diagram for the addToCart() and placeOrder() functions in the “Order Pizza” sequence diagram.

A diagram of a data flow

Description automatically generated

### Activity diagram for the displayOrders() function in the “Order History” sequence diagram. A screenshot of a diagram Description automatically generated

### Activity diagram for the editProfile() function in the “Edit Profile” sequence diagram.

A diagram of a software project

Description automatically generated with medium confidence

### Activity diagram for the checkOrderStatus() function in the “Order Status” sequence diagram.

A screenshot of a computer

Description automatically generated

# State Diagrams

### State Diagram: Create AccountA diagram of a user account Description automatically generated

### State Diagram: Order PizzaA diagram of a cart Description automatically generated

### State Diagram: Check Order Status

A diagram with a red line and a blue line

Description automatically generated with medium confidence

### State Diagram: Check Order HistoryA diagram of a diagram Description automatically generated

### State Diagram: Edit ProfiA diagram of a user account Description automatically generated

# Database Design

## ER Diagram

Entity Relationship diagram to show the relationships between the database tables.

A screenshot of a diagram

Description automatically generated

## Table Schema

Table relational schema of the database tables with their primary keys and foreign keys:

A black screen with white text

Description automatically generated

# Conclusion

In conclusion, the online pizza ordering system for Dom’s Pizza is designed to provide customers with a seamless and convenient way to order their favorite pizzas while improving operational efficiency for the restaurant. It encompasses user account management, pizza menu, order processing, and administrative tools to create a comprehensive and efficient online ordering experience.