Cordial, Nicholas I.

  Fall Semester, 2023

FINAL REPORT

Contents

[I. Introduction 1](#_Toc149134889)

[A. Problem Statement 1](#_Toc149134890)

[B. Proposal 1](#_Toc149134891)

[II. System Description 1](#_Toc149134892)

[III. System Requirements 2](#_Toc149134893)

[A. Functional Requirements 2](#_Toc149134894)

[3. Non-functional Requirements 8](#_Toc149134895)

[IV. Use Case Diagram 9](#_Toc149134896)

[V. Class Diagram 10](#_Toc149134897)

[VI. Sequence Diagrams 10](#_Toc149134898)

[VII. Activity Diagrams 10](#_Toc149134899)

[VIII. State Diagram(s) 10](#_Toc149134900)

[IX. Database Design 10](#_Toc149134901)

[A. ER Diagram 10](#_Toc149134902)

[B. Table Schema 10](#_Toc149134903)

[X. Conclusion 10](#_Toc149134904)

# Introduction

## Problem Statement

The ZZZ Pizza needs a system for their customers to order pizza online.

## Proposal

We propose a software system to help the store with online purchases.

# 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 staff 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 staff when the stack of an item is low. The staff will then find a supplier the store with the item. Other considerations are welcome.

# System Requirements

## Functional Requirements

1. The system shall allow a customer to create a member account.
   1. The customer shall press the “Create Account” button.
   2. The system shall display a new member form.
   3. The customer shall enter his or her name, address, phone number, email as username, password, or username (if email is not used as username)

A screenshot of a application form

Description automatically generated

* 1. The customer shall press the Submit button.
  2. The system shall check if the email is already in the database.
     1. If the email is already in the database, the system shall display an error message to request the customer enter a different email.

A screenshot of a computer error

Description automatically generated

* + 1. If the email does not exist in the database, the system shall save the data to the database and display a confirmation message.

A screenshot of a computer

Description automatically generated

1.6 The system shall redirect the customer to the main menu if their account was made successfully.

1.7 Go to R2

1. The system shall allow a customer to order pizzas.
   1. The customer shall press the “Order Online” button.
   2. The system shall display an online order form with options the customer shall choose from.

A screenshot of a pizza menu

Description automatically generated

* 1. The customer shall press a button corresponding to the pizza they choose.
  2. The system shall update the customer’s cart with the pizza and price.
  3. The customer shall press either the “Back” or “Proceed to Checkout” button.
     1. If the customer has not chosen a pizza, they shall pick “Back” and be returned to the main menu.
     2. If the customer has chosen a pizza, they shall pick “Checkout” and the system will display a form where the customer can enter their payment information.

A screenshot of a checkout form

Description automatically generated

* 1. The customer shall press the “Submit” button.
     1. If the payment form is empty when the customer hits the submit button, the system shall display an error.

A screen shot of a computer

Description automatically generated

* + 1. If the payment form has been filled out when the customer hits the submit button, the system will display a confirmation message and return the customer to the main page.

A screenshot of a computer

Description automatically generated

* 1. Go to R3

1. The system shall allow a member to edit their account information.
   1. The member shall press the “Edit Account” button.
   2. The system shall display a member login screen where the member shall input their email and password.

A screenshot of a computer

Description automatically generated

* 1. The member shall press the login button.
  2. The system shall display a member account form with the member’s information.

A screenshot of a computer

Description automatically generated

* 1. The member shall press the “Edit” button.
  2. The system shall now allow the member to edit their information.

A screenshot of a computer screen

Description automatically generated

* 1. The member shall hit the “Save Changes” button.
  2. The system shall display a confirmation and update the database with the member’s new information.

A screenshot of a computer error

Description automatically generated

* 1. Go to R4

1. The system shall track the order of a customer/member.
   1. The customer shall press the “Check Order Status” button.
   2. The system shall display an order tracking form for the most recent order the system has handled.

A screenshot of a computer

Description automatically generated

* 1. Go to R5

1. The system shall allow a member to check their order history.
   1. The member shall press the “Check Order History” button.
   2. The system shall display a member login form to access the order history.

A screenshot of a computer

Description automatically generated

* 1. The customer shall enter their username and password.
     1. If the username and password is incorrect, the system shall display an error message.

A screenshot of a computer

Description automatically generated

* + 1. If the username and password are correct, the system shall display an order history form.

A screenshot of a pizza delivery

Description automatically generated

## Non-functional Requirements

1. The length of the password for the member account should be at least 8 characters long.
2. The system must be able to handle many customers placing orders at the same time.
3. The system must be able to handle customers creating accounts and updating a database with the customer’s information in an efficient manner.

# Use Case Diagram



Five use cases have been identified for the system:

* Create account will let a customer create a membership account for the store.
* Order pizza will allow customers to order a pizza online.
* Check order status will allow all customers to view the status of their order.
* Edit account will allow customers who have registered as members to edit their account information such as their email or password.
* Check order history will allow customers who have registered as members to view their order history.

# Class Diagram

A screenshot of a computer

Description automatically generated

*This is the class diagram for the system. It shows all the classes as well as their attributes and operations that are needed for the system to function properly.*

# Sequence Diagrams

A screenshot of a computer

Description automatically generated

*This is the sequence diagram for the Create Account use case*

A screenshot of a computer

Description automatically generated

*This is the sequence diagram for the Order Pizza use case.*

A screenshot of a computer

Description automatically generated

*This is the sequence diagram for the Edit Account use case.*

A screenshot of a diagram

Description automatically generated

*This is the sequence diagram for the Check Order Status use case.*

A diagram of a company

Description automatically generated with medium confidence

*This is the sequence diagram for the Check Order History use case.*

# Activity Diagrams



*This is the activity diagram for the Create Account use case*

*A diagram with text and words

Description automatically generated with medium confidence*

*This is the activity diagram for the Order Pizza use case.*

*A screenshot of a chat

Description automatically generated*

*This is the activity diagram for the Edit Account use case.*

*A screenshot of a computer

Description automatically generated*

*This is the activity diagram for the Check Order Status use case.*

*A screenshot of a diagram

Description automatically generated*

*This is the activity diagram for the Check Order History use case.*

# State Diagram(s)

A screenshot of a computer

Description automatically generated

*This is the state diagram for the Pizza Order System, with each use case and the different states for each case listed.*

# Database Design

## ER Diagram

A diagram of a company

Description automatically generated

*This is an ER Diagram displaying which tables will be included, the attributes associated with them, and the relationships between them all.*

## Table Schema

A screenshot of a computer

Description automatically generated

*This is the table schema for the database.*

# Conclusion

The system shall have five use cases consisting of create account, edit account, order pizza, check order status, and check order history. Above I have listed the functional requirements that shall fulfill these cases, as well as some additional non-functional requirements that will be beneficial to the system. Above I have also displayed the Sequence and Activity Diagrams to show how the system will perform operations, and the State Diagram that shows each state the system will have to pass through to operate. Finally, I have listed an ER Diagram and the Table Schema for the final database. Pizza yum 😊