**Pizza Ordering System**

**Software Requirements Specification**

**Version 1**

**Team Number: Group 2**

**Project Manager: Aaron Jones**

**Team Members: Angel Candela-Serratos, Blake Tallman, Richard Londola, Shannon Herman, Ethan Ines**

**Revisions**

| **Version** | **Primary**  **Author(s)** | **Description of Version** | **Date Completed** |
| --- | --- | --- | --- |
| 1 | Aaron Jones | Initial revision | 10/20 |
|  |  |  |  |

**Review History**

| **Reviewer** | **Version Reviewed** | **Date** |
| --- | --- | --- |
|  |  |  |
|  |  |  |

**Table of Contents (version 1)**

1. Introduction
   1. Project Objectives
   2. Project Scope
   3. Project Overview
2. Project Description
   1. Project Features / Functions
   2. User Stories
   3. Use Cases
   4. Project Assumptions and Dependencies
3. Project Collaboration and Documentation
4. Project Management
5. Requirements Specification
   1. Business Requirements
   2. User Requirements
   3. Functional Requirements
   4. Non-Functional Requirements
6. **Introduction**
   1. Project Objectives

The primary goal of the Pizza Ordering System is to provide an efficient, user-friendly platform for customers to order pizza online. The system aims to streamline the process of pizza customization, order tracking, and payment while integrating with the restaurant's kitchen management and delivery operations.

* 1. Project Scope

This project includes the development of a web-based application that allows users to customize pizzas, view available options, place orders, and make payments online. It will also allow the restaurant to manage orders in real-time.

* 1. Project Overview

The Pizza Ordering System will enable users to:

* Customize pizzas by selecting sizes, crust types, and toppings.
* Place and track their orders in real-time.
* Make payments directly within the application.

1. **Project Description**
   1. Project Features / Functions
2. **Pizza Customization** – Users can select pizza size, crust type, and toppings.
3. **Shopping Cart** - Users can review their order within the cart before placing the order.
4. **Real-time Order Tracking** – Customers can track the progress of their order from preparation to delivery.
5. **Secure Online Payment** – The user will have an option to directly pay within the app, otherwise cash payment can be accepted upon delivery.
   1. User Stories
      1. As a customer, I want to customize my pizza so that I can choose my favorite toppings.
      2. As a customer, I want to review my order before paying so I don’t order the wrong thing.
      3. As a customer, I want to track my order so that I know when it will arrive.
      4. As a restaurant employee, I want to receive real-time orders so that I can start preparing pizzas promptly.
   2. Use Case
      1. Use Case 1: Pizza Customization
         * Actors: Customer
         * Precondition: The customer is logged into the system.
         * Description: The customer selects a pizza, chooses a size, crust type, and toppings, and adds it to the cart.
         * Postcondition: The customized pizza is added to the shopping cart.
      2. Use Case 2: Shopping Cart
         * Actors: Customer
         * Precondition: The customer has a pizza and topping selected and is ready to pay.
         * Description: The customer clicks a “Review Cart” button.
         * Postcondition: The cart correctly displays the items the customer has selected.
      3. Use Case 3: Order Tracking
         * Actors: Customer
         * Precondition: The customer has placed an order.
         * Description: The customer tracks their order status in real-time.
         * Postcondition: The customer sees updates on the order’s preparation, dispatch, and delivery status.
      4. Use Case 4: Payment Processing
         * Actors: Customer
         * Precondition: The customer has finalized the order and is ready to pay.
         * Description: The customer selects a payment method and completes the transaction.
         * Postcondition: The order is confirmed, and payment is processed securely.
   3. Project Assumptions and Dependencies
      * + The restaurant's kitchen must have a system in place to receive and manage orders.
        + Internet access is required for customers to use the application.
        + The application depends on third-party payment gateways for secure transactions.
6. **Project Collaboration and Documentation**

The team will use GitHub for version control and Microsoft Teams for communication. Documentation will be maintained with the GitHub repository, and tasks will be discussed and distributed in Teams.

1. **Project Management**

The project will follow an Agile development process, using Scrum methodology. Sprints will be 2 weeks long, with weekly stand-up meetings, sprint planning, and retrospectives.

1. **Requirements Specification**
   1. **Business Requirements**

| Requirement ID | Requirement Description | MOSCOW |
| --- | --- | --- |
| BR1 | The system must provide a way for customers to place and track orders online. | M |
| BR2 | The system must offer pizza customization options for customers. | M |
| BR3 | The system should provide detailed order analytics for the restaurant management. | S |

* 1. **User Requirements**

| Requirement ID | Requirement Description | MOSCOW |
| --- | --- | --- |
| UR1 | Users must be able to customize their pizza and view the price dynamically. | M |
| UR2 | Users should be able to review their order before paying. | S |
| UR3 | Users should be able to track their order status in real-time. | S |
| UR4 | Users must be able to securely pay within the application. | M |

* 1. **Functional Requirements**

| Requirement ID | Requirement Description | MOSCOW |
| --- | --- | --- |
| FR1 | The system must allow users to customize pizzas by selecting various options. | M |
| FR2 | The system should display a shopping cart with the items the user has selected. | S |
| FR3 | The system should display real-time order tracking to the user. | S |
| FR4 | The system must complete transactions after verifying payment methods. | M |

* 1. **Non-Functional Requirements**

| Requirement ID | Requirement Description | MOSCOW |
| --- | --- | --- |
| NFR1 | The system must have a user-friendly interface that allows users to order pizza easily. | M |
| NFR2 | The system could handle up to 500 concurrent users without performance degradation. | C |