

TITLE: PIZZA ORDERING SYSTEM

1. Introduction

In the modern digital era, food delivery services are becoming increasingly popular, and customers expect a seamless online ordering experience. This project aims to develop a **cloud-based pizza ordering system** that enables customers to place orders, track order status, and manage their details efficiently. The system leverages a **relational database on the cloud** to ensure scalability, data consistency, and real-time updates.

This system will allow customers to:

- **Browse a menu** of available pizzas.
- **Select pizza(s) and place an order** with customized options.
- **Track order status** in real time.
- **Receive automatic order updates** as their order progresses.

2. Problem Statement

Traditional phone-based or in-person pizza ordering methods are inefficient and prone to errors. Customers may face:

- **Long wait times** for order processing.
- **Lack of real-time tracking** of order status.
- **Errors in order details** due to manual processing.

To solve these issues, this project will implement an **automated cloud-based ordering system** with real-time updates.

3. Objectives

The main objectives of this project are:

- **Implement a cloud-hosted database** to store customer, order, and menu details.
- **Develop an automated ordering system** where customers can place and track their orders.
- **Ensure real-time order status updates** using database triggers and events.
- **Optimize database performance** using indexing and normalization techniques.
- **Allow scalability** for multiple customers and orders simultaneously.

4. System Design & Architecture

4.1 System Overview

The **Pizza Ordering System** consists of three main components:

1. **Frontend (User Interface)** – Customers interact with the system via a user-friendly Java Swing GUI.
2. **Backend (Business Logic)** – Processes customer orders, manages database interactions, and handles order updates.
3. **Database (Cloud-Hosted MySQL)** – Stores all customer, menu, and order details.

4.2 Database Schema

The system consists of the following key tables:

- **Customers:** Stores customer details (name, phone, address, email).
- **Orders:** Tracks orders placed by customers.
- **Order Items:** Stores multiple pizzas for a single order.
- **Pizzas:** Stores menu items with size and price details.
- **Order Status Updates:** Logs order status changes.

Each table is **normalized up to 4NF** to eliminate redundancy and ensure data consistency.

4.3 Cloud Integration

- **The database is deployed on a cloud server** (e.g., AWS RDS, Google Cloud SQL).
- **Real-time order status updates** are triggered automatically in the cloud database using **MySQL Events**.

5. Features & Functionalities

5.1 User Features

Menu Display & Selection

- Customers can **view available pizzas**, including name, description, and price.
- Customers select pizzas and **specify quantities**.

Order Placement & Processing

- Customers enter their **contact details** and confirm orders.
- Orders are **assigned unique Order IDs** for tracking.

Order Status Tracking

- Orders move through **Pending → Preparing → Out for Delivery → Delivered**.
- A **MySQL Event runs every 30 seconds** to update statuses automatically.

Stored Procedures for Efficient Data Retrieval

- Customers can retrieve order details using their **Order ID**.

- The system provides **real-time order tracking**.

5.2 Database Functionalities

Automated Order Processing

- The database **automatically updates** the order status at fixed intervals.
- A **stored procedure** retrieves complete order details.

Indexing for Performance Optimization

- **Indexes on primary keys and foreign keys** speed up search queries.

Foreign Key Constraints

- Ensures referential integrity by using **ON DELETE CASCADE**.

6. Technologies Used

Frontend

- Java Swing (Graphical User Interface)

Backend

- Java (JDBC for database connectivity)

Database

- **MySQL (Cloud-Hosted)**
- Triggers, Events, and Stored Procedures

Cloud Deployment

- AWS RDS / Google Cloud SQL

7. Implementation plan:

PHASE 1	Design Database Schema & Cloud Setup, Implement Database & Write SQL Queries	Week 1
PHASE 2	Develop Backend Logic & API Connectivity, Build Frontend & Integrate with Database	Week 2
PHASE 3	Testing & Debugging, Deployment & Documentation	Week 3

8. Outcomes:

Menu selection output:

Menu button lets user to see pizza names, description and prices according to its size

Pizza Deluxe Ordering System

<

Place order output:

Place order button allows user to select single or multiple pizzas, size, quantity, add special instruction and total cost will be displayed accordingly

Pizza Deluxe Ordering System

Pizza Deluxe

Taste the Excellence

Menu Place Order Track Order

Select Pizza: BBQ Chicken

Size: Small

Quantity: 5

Total Price: \$70.00

Special Instructions:

Margherita – Small x1 (\$10.00)
BBQ Chicken – Small x5 (\$60.00)

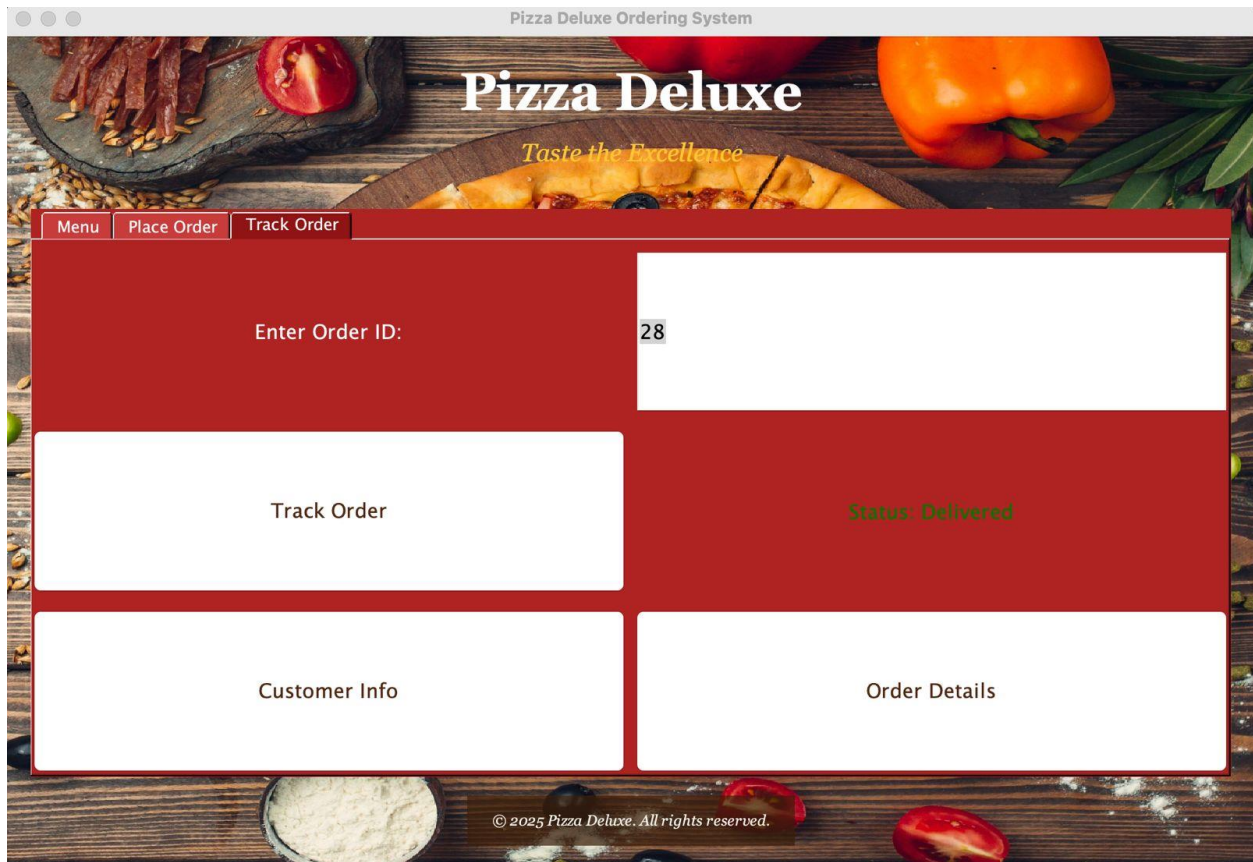
Add to Cart Place Order

Reset

© 2025 Pizza Deluxe. All rights reserved.

Track Order output:

Track order button allows the user to track order status using a unique order id. Customer info gets the customer information and order info gets the order information.



9. Conclusion:

This project aims to streamline the pizza ordering process using a **cloud-hosted MySQL database** with **automated status updates**. The system will reduce manual errors, improve efficiency, and provide a **better customer experience**.

This database-driven approach ensures:

Scalability – Supports large customer orders.

Automation – Orders update in real-time.

Accuracy – Eliminates manual errors in order tracking.

10. References:

- [MySQL Documentation](#)
- [AWS RDS / Google Cloud SQL Documentation](#)
- [Java Swing & JDBC Connectivity](#)