# Project Overview and My Role

## Project Description:

The food ordering web application allows users to browse menus, place orders, and track delivery statuses in real-time. The platform aims to streamline the food ordering process for both customers and restaurant owners. A special feature we included is the ability for customers to customize the spiciness, cheese level, sauce level, and amount of meat in their orders. Once the food is ordered, the user can share the auto-generated posters through Twitter.

Additionally, the administrator/cafe owner can view the orders, change the order status, add menu items from the application, and view the users.

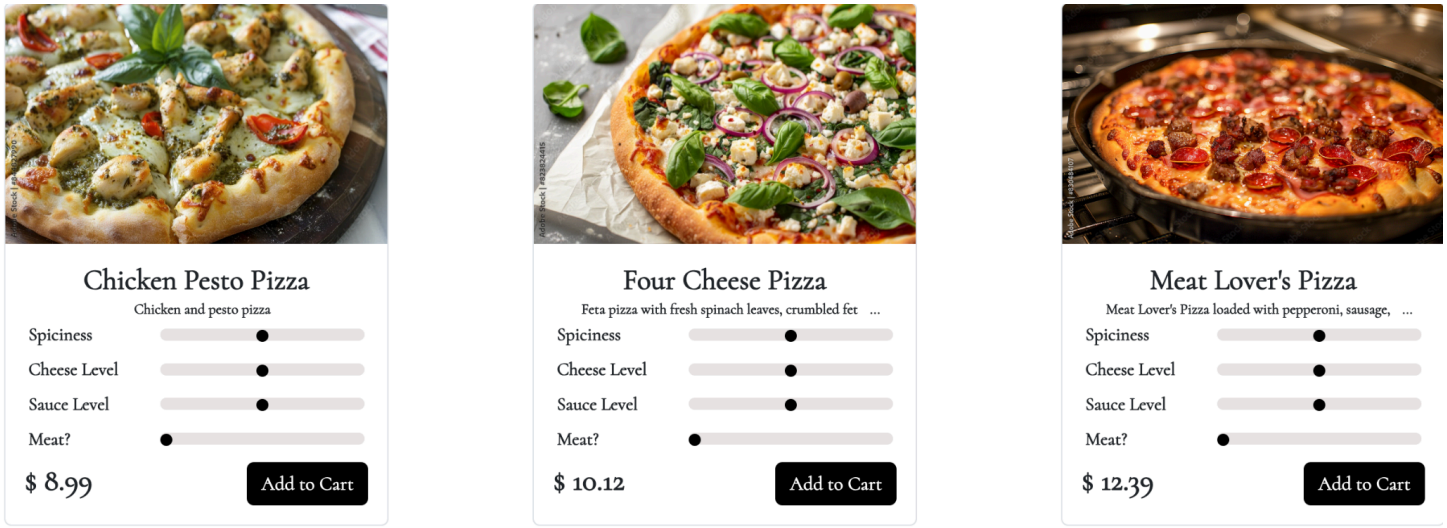


*Fig: Menu Card Example*

## Video Demonstration:

You can view a video demonstration of the project here.

## GitHub Repository:

You can review the project's source code on my GitHub repository here.

## Technology Stack:

**Cloud Server**: AWS EC2 t2.micro instance
**Operating System and Version Number**: Ubuntu Server 22.04 LTS (HVM)

**Database and Version Number**: MySQL 8.0.33
**Web Server**: Nginx 1.24.0
**Server-side Language**: NodeJS
**Web Application Framework**: Express
**Front-end technology**: React
**Additional technologies:**

- IDE - Visual Studio Code
- API - Twitter(X) API

# My Role:

## Team Lead for a Team of 5:

- **Team Leadership:**
  - Guided a diverse team, leveraging individual strengths to achieve project goals.
  - Managed challenges effectively, especially those arising at critical moments, developing skills in responsibility and working under pressure.
- **Communication and Collaboration:**
  - Successfully kept the team aligned and focused amid distractions, ensuring a cohesive workflow and optimal productivity.

## Backend Integration:

- Entire backend was developed by me.
- Developed RESTful APIs using Node.js and Express to handle user authentication, order processing, and real-time updates.
- Ensured seamless communication between the frontend and backend.
- Integrated Twitter API.

## Frontend Development:

- Implemented animations and enhanced the visual appeal of the application.

## Database Management:

- Set up and managed the MySQL database, ensuring data consistency and security.
- Designed schemas and optimized queries for efficient data retrieval.
- Database connection credentials:
  - **Database URL:** pizzacraze-database-1.c740k6c82945.us-east-2.rds.amazonaws.com
  - **Username:** admin
  - **Password:** mypassword
  - **Port:** 3306

### AWS Deployment:

- Deployed the application on AWS EC2, configuring the server environment and ensuring high availability.

## Challenges and Solutions:

- **Twitter API:**
  - **Challenge:** It was my first time using an API, and it took some time to integrate the API in the backend. Additionally, the free API version would only work for a single account and didn't include a login portal, requiring the security keys to be hardcoded.
  - **Solution:** I thoroughly studied the API documentation and sought help from online communities to understand the integration process. I implemented secure practices for handling the API keys within the constraints of the free version.
- **AWS EC2 Backend Deployment:**
  - **Challenge:** Running the backend and frontend on two different ports and connecting them via Nginx took considerable time and learning through trial and error. Ports were occasionally occupied, necessitating code adjustments. Furthermore, I had to take down the server eventually due to high AWS costs.
  - **Solution:** Learned to configure Nginx to route traffic correctly between the frontend and backend. Managed port conflicts by systematically releasing occupied ports and reconfiguring the application as needed. Explored cost-effective deployment alternatives for future use.
- **Git Conflict Resolution:**
  - **Challenge:** Managing and resolving git conflicts among team members.
  - **Solution:** We established a protocol where team members regularly pulled and updated their code before starting work and communicated updates when pushing code. Assigning a dedicated person to review pull requests proved highly effective in preventing conflicts.
- **Managing the Team:**
  - **Challenge:** Managing the team and ensuring the work was done effectively was a significant challenge, especially under tight deadlines.
  - **Solution:** I maintained clear and consistent communication with team members, held regular check-ins to monitor progress, and provided support and guidance to overcome any obstacles. This approach helped in keeping the team aligned and productive.

## Learning Outcomes:

Throughout the development of the food ordering web application, I gained valuable insights and skills in several key areas:

- **API Integration:**
    - Learned how to effectively integrate and manage third-party APIs, specifically the Twitter API. This experience enhanced my understanding of API documentation and secure handling of API keys.
- **Cloud Deployment:**
    - Acquired practical knowledge of deploying applications on AWS EC2. I learned how to configure servers, manage ports, and use Nginx for connecting frontend and backend services. This experience taught me the importance of documentation and systematic troubleshooting.
- **Team Management:**
    - Developed leadership skills by managing a diverse team. I learned how to leverage individual strengths, maintain team alignment, and handle challenges under pressure. This experience improved my ability to communicate effectively and ensure optimal productivity within a team.
- **Full-Stack Development:**
    - Gained comprehensive experience in full-stack development, including setting up and managing databases (MySQL), developing backend services (Node.js and Express), and enhancing frontend applications (React). This holistic understanding allows me to contribute effectively to all aspects of web application development.

# Best Practices:

## Offload Processing to the Backend:

- Put most of the load in the backend so that the server will handle intensive processing tasks, ensuring better performance and efficiency on the client side.

## Know Your Teammates:

- Understand the strengths and skills of your team members to leverage their abilities effectively. Assign tasks that align with their strengths for optimal team performance.

## Follow Git Feature Branching:

- Using git feature branching is highly beneficial. It helps in organizing work, managing code changes, and reducing conflicts by isolating development work into dedicated branches.