Final Project Report: SD_Foods_csc322

Team Members: Shreyosee Chowdhury, Sanskriti Khadka, Damilola Babajide, Sri Suvetha Meenaa Subramanian

Introduction:

Our project, SD_Foods, is an AI-powered online ordering and delivery system for restaurants that simulates all elements of a smart restaurant. The platform supports three main user groups, each with unique capabilities and workflows: customers (registered and VIP), visitors, and employees (chefs, delivery people, and managers).

To ensure easy installation and consistency, the system was built using widely supported packages and platforms.The backend was created using Python (Flask), with MySQL serving as the local relational database for users, menu items, orders, complaints, bids, and the knowledge base. The backend has role-based logic for chefs, delivery drivers, managers, and customers. For customer support chat and knowledge-base fallback reasoning, we added an AI module that utilized locally hosted LLMs via Ollama Transformers. The frontend is built using React + Vite (JavaScript), providing an interactive UI with menu images, orders, rating pages, warnings display, and the AI chat interface.
The system can be installed on Windows, macOS, and Linux. Running the application only requires installing Python 3.8 or above, Node.js, and pulling our GitHub repository. After installing dependencies (pip install -r requirements.txt and npm install), the backend and frontend can be launched from scratch without additional configuration.

2. 1-to-1 Correspondence to Required System Features

User Types:

Features requirement	**Status**
at least two chefs who independently decide the menus and make the dish;	Finished
Two delivery people	Finished
Manager role with HR and complaint handling	Finished
Customer (register and VIP)	**Finished**
Visitor browsing menus	Finished

Restaurant features:

Features	status
Menu browsing with pictures	Finished
Customers ordering food	Finished
Rating system/ feedback	Finished
Personalized recommendations.	Partially done
Discussion topics on chefs/dishes/delivery	Unable to do it

Reputation and complaint management

Feature	Status
Customers can file complaints about chefs/delivery/customers	Finished
Delivery people can complain about customers	Finished
Manager resolves disputes, assigns warnings	Finished
Automatic consequences (deregistration, demotion, firing)	Partially done

Finance Manangement

Feature	Status
Customers deposit money	Finished
Reject order if insufficient funds + add warning	Finished

Human resource:

Features	Status
Blacklist kicked-out customers	Finished
Chef/delivery demotion & bonus logic	Partially Done
Delivery bidding system	Finished

AI- based customer service

Features	Status
Local knowledge base responses	Finished
Fallback to LLM when KB has no answer	Finished
Rating of KB answers + flagging bad content	Finished

3. Contribution of team members and work division:

Our project was developed collaboratively by the 4 members with responsibilities divided based on the project requirements.

- **Frontend Team (2 members):**

  Responsible for designing and implementing the user interface using React and Vite.

- **Backend Team (2 members):**

  Responsible for developing the Flask backend and implementing Rest API routes.

- **Shared Responsibilities (All 4 members):**

  Database design and implementation

  AI Chatbot and knowledge base and creative feature.