Restaurant Management System

DEVELOPED BY:
Shreya Bharti (IIT2021067)
Ekeshwar Aditya (IIT2021052)

ABSTRACT:

The aim of this project is to create a Restaurant Management Database (RMD) is an online application for restaurant management. This system wakes to provide service facilities to restaurants and to the customer. The services which are provided are food ordering, reservation of the table by the customer through the system online, menu information management, and report.

Work Plan for Restaurant Management System is as follow:

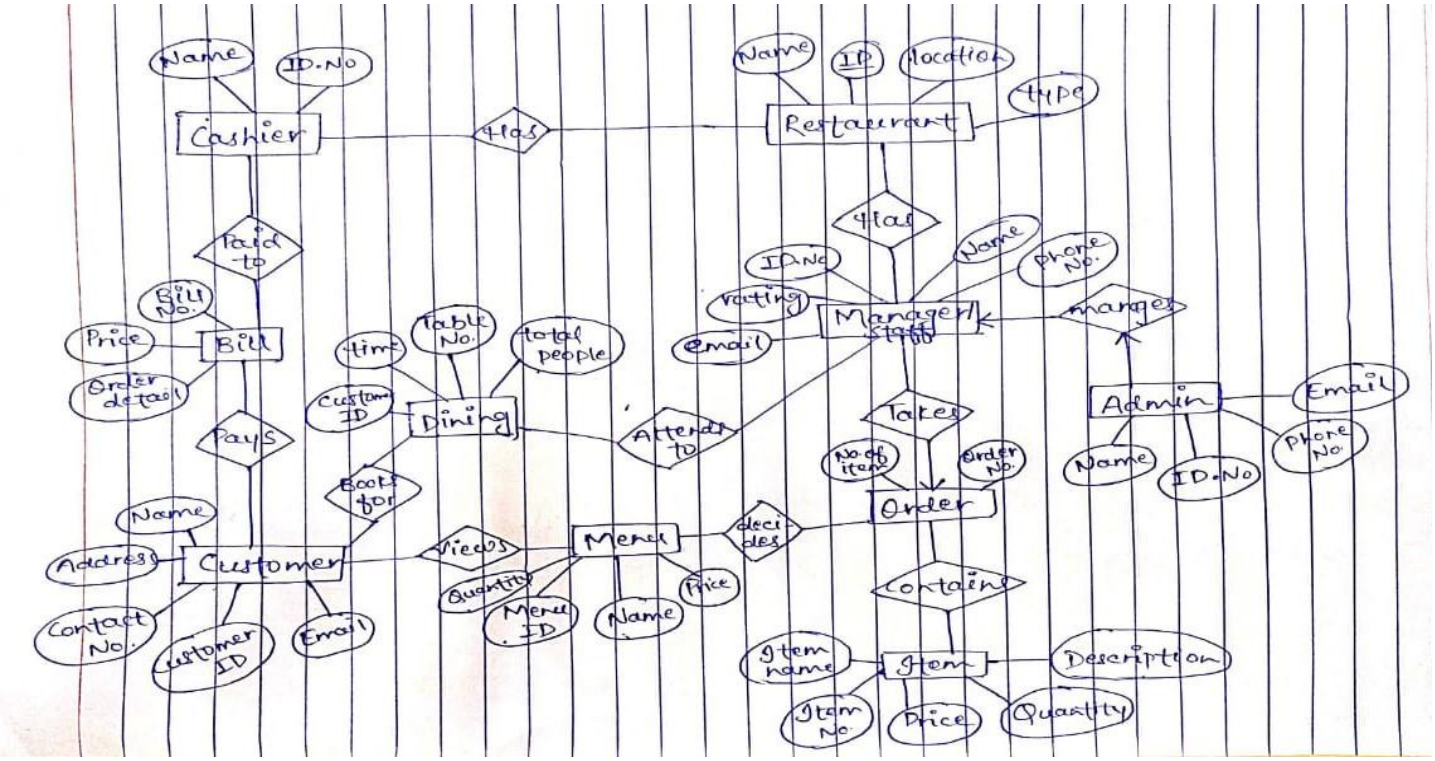
STEP 1.
PLANNING AND REQUIREMENTS:

- Define the scope and requirements of the system
- Conduct research on similar systems in the market

● Gather requirements for functional components of the project like management of menu, reservation, order, payment, inventory and staffs.
● Design the UI/UX for the frontend and database schema for the backend
● Finalize the technology stack for the project.

STEP 2.
DATABASE DESIGN:

ER diagram for the restaurant management system



STEP 3.
FRONTEND DEVELOPMENT PHASE:

● Set up the development environment
● Develop the user interface of the restaurant management system using HTML, CSS, and JavaScript

- Create the frontend logic using a frontend framework such as React
- Develop responsive design for various screen sizes and devices
- Integrate any third-party libraries or APIs as needed
- Test the frontend code and fix any issues

STEP 4.
BACKENED DEVELOPMENT PHASE:

- Set up the development environment for the backend
- Develop the backend of the restaurant management system using a server-side language(PHP) and client-side language javascript.
- Create the database schema and implement data storage using a database management system such as MySQL or Oracle database server.
- Implement the necessary APIs and services for the frontend to interact with the backend
- Implement security measures such as authentication and authorization
- Test the backend code and fix any issues

STEP 5.
INTEGRATION AND TESTING PHASE:

- Integrate the frontend and backend components
- Conduct integration testing to ensure that all components work together seamlessly
- Conduct user acceptance testing to ensure that the system meets the requirements and is user-friendly
- Fix any bugs or issues identified during testing phase.

CONCLUSION:

The purpose of the restaurant management system is to improve worker efficiency and to maximize profit margin of restaurant owners by providing better service. Providing prompt response to

customers through use of a System and data collection by the Main Dispatcher will allow this to happen. The System is not designed to replace the existing ordering systems which are at many restaurants but to complement it.