ABSTRACT

DEVELOPMENT OF A PERSONALIZED
HOMECHEF PLATFORM

This paper explores the design and development of a web-based Home Chef Platform aimed at bridging the gap between home cooks and individuals residing in PGs, hostels, and rented accommodations, such as students and working professionals. With the increasing demand for personalized, affordable, and healthy meals, the platform enables users to connect with local cooks who can either cook at the user's place, offer takeaway meals from their own residence, or be hired for special events and gatherings.

The system investigates user preferences, meal personalization techniques, and recommendation algorithms to deliver a more tailored food experience. By integrating geolocation services and real-time availability, users are able to discover nearby cooks based on service type, cuisine, and ratings. The study also addresses key operational challenges including meal planning, ingredient sourcing, hygiene management, and dietary customization to support user health and satisfaction.

The platform architecture is designed with three distinct user roles: Admin, Cooks, and End Users, each with unique privileges and functionalities. Advanced features such as secure online payment integration, order tracking, and feedback systems enhance the overall usability and trust in the platform. Furthermore, experimental results showcase the effectiveness of machine learning-based meal recommendations in improving user engagement, with a notable increase in booking consistency and satisfaction rate

FRONTEND - HTML, CSS,JAVASCRIPT,PHP
BACKEND – PHP
DATABASE - MYSQL SERVER

1. Introduction

In today's fast-paced world, many students and working professionals staying in PGs and rented accommodations struggle to cook healthy meals due to time constraints and limited cooking resources. This project aims to develop a web-based food service platform that connects users with local cooks who offer three types of services:

- Cooking at the user's place
- Takeaway meals from the cook's place
- Booking for special events

This system simplifies meal access while providing employment opportunities to home-based or freelance cooks.

System Analysis

a. System Study

The system focuses on building a convenient digital solution for meal access, involving three major user types:

- Users (customers ordering food or booking cooks)
- Cooks (service providers)
- Admin (system manager)

It ensures smooth user interactions, order tracking, and secure payment integration.

b. Existing System

Currently, food delivery platforms like Zomato and Swiggy only offer food from restaurants. They lack personalized meal prep, cook-on-demand, or affordable homemade meal services. There is no platform exclusively connecting users to individual cooks who can offer custom, healthy options.

c. Proposed System

The proposed system introduces a dedicated platform where:

- Users can book nearby cooks, schedule services, and make payments online.
- Cooks can accept orders, manage availability, and list service types.
- Admin can monitor users, cooks, and platform activities.

Benefits:

- Homemade food accessibility
- Promotes healthy eating habits
- Provides livelihood to local cooks

d. Feasibility Study

Technical
Tools like PHP, MySQL, and JavaScript are readily available and manageable for mini project scale.

Operational
Easy to use interface for all user types. The platform can work on mobile or web browsers.

Economic
Minimal development cost using open-source tools. Optional hosting on free/cloud servers for demo.

e. Problem Description
- Difficulty in getting homemade or custom food
- Restaurant food is expensive or unhealthy
- Lack of flexible cook booking system
- No digital platform for freelance cooks

System Design

a. Design Concepts

The platform is based on:

- Three-tier architecture: Frontend – Backend – Database
- Role-based access control: Each role (Admin, Cook, User) has unique permissions
- Service Matching Logic: Users find nearby cooks based on location/service

b. Data Flow Diagram (DFD)

Level 0 DFD (Context Level)

Shows interactions between Users, Cooks, Admin, and the system.

```
[User] ---> [Food Service System] <--- [Cook]
                      ^
                      |
                  [Admin]
```
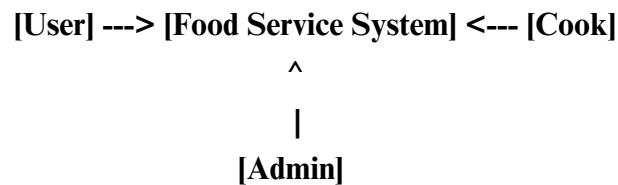
Level 1 DFD (can include in diagram or explain):

- User → Register/Login → Browse Cooks → Book/Pay
- Cook → Register/Login → Update Availability → Accept Orders
- Admin → Manage Accounts → Monitor Activities

c. Database / I-O Design

Database Tables (simplified):

- users: id, name, email, password, address
- cooks: id, name, email, services_offered, location
- bookings: id, user_id, cook_id, service_type, date, status
- payments: id, booking_id, amount, payment_status
- admin: id, username, password

Input/Output Examples:

- Input: User registration form, booking form
- Output: Booking confirmation, cook availability, admin reports

SYSTEM DESIGN

Data Flow Diagram (DFD)

A Data Flow Diagram (DFD) is a graphical representation of the "flow" of data through an information system, modelling its process aspects. A DFD is often used as a preliminary step to create an overview of the system, which can later be elaborated. DFDs can also be used for the visualization of data processing (structured design).
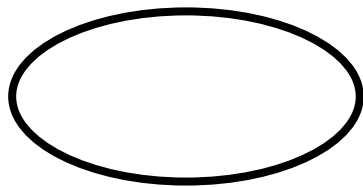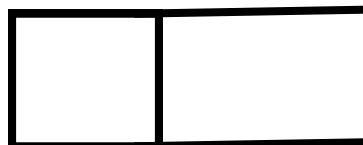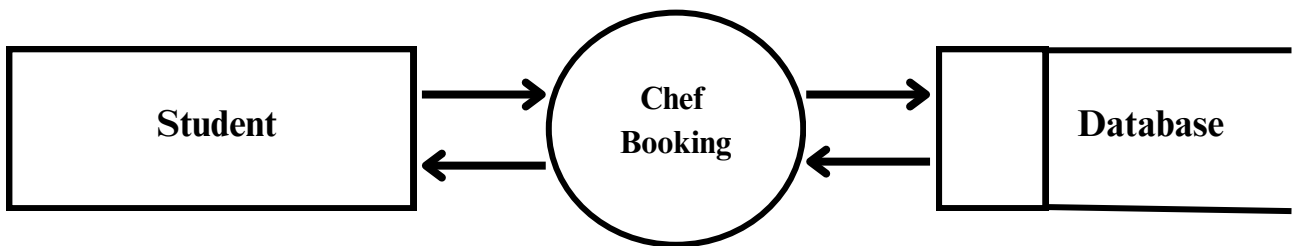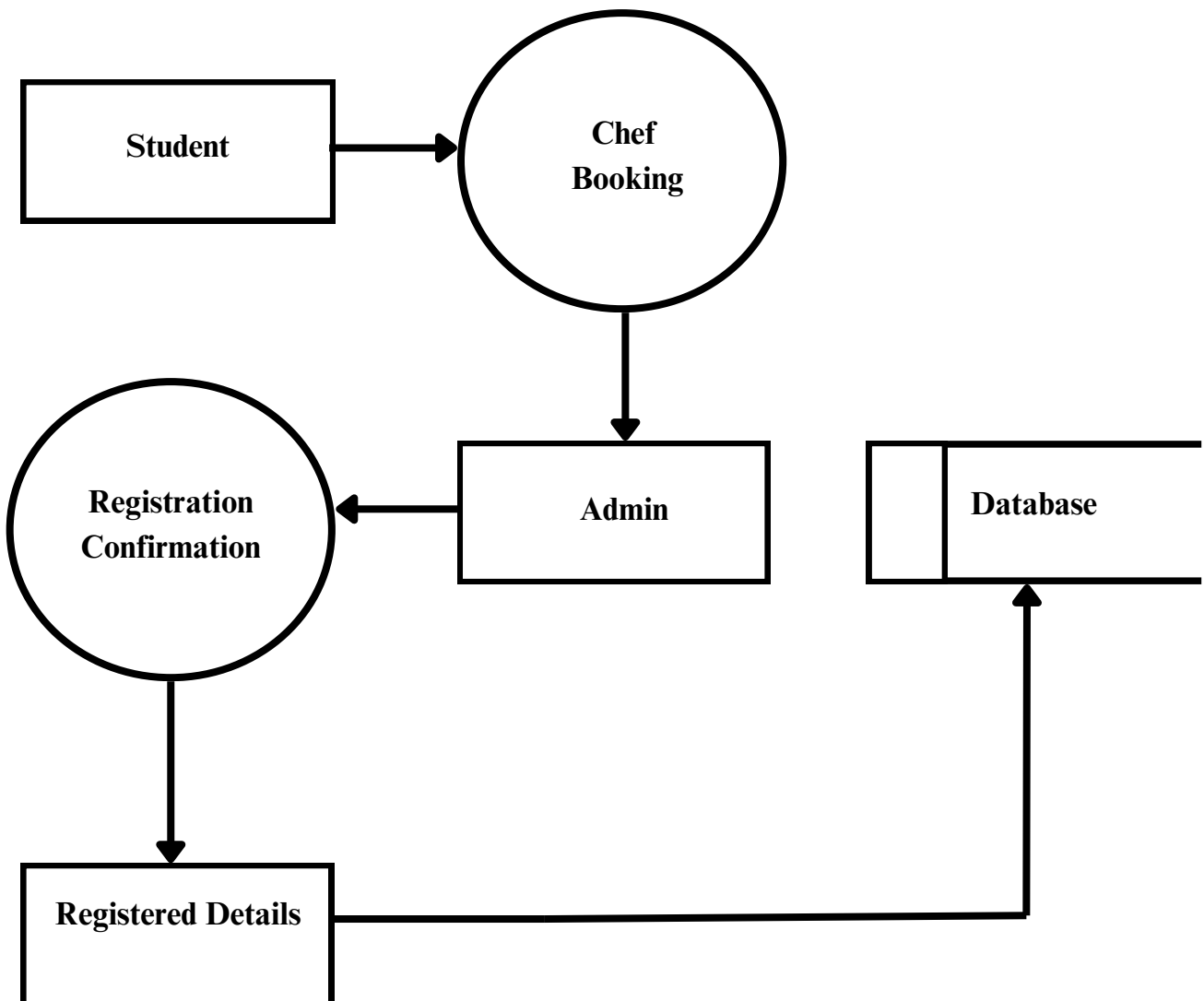
Function

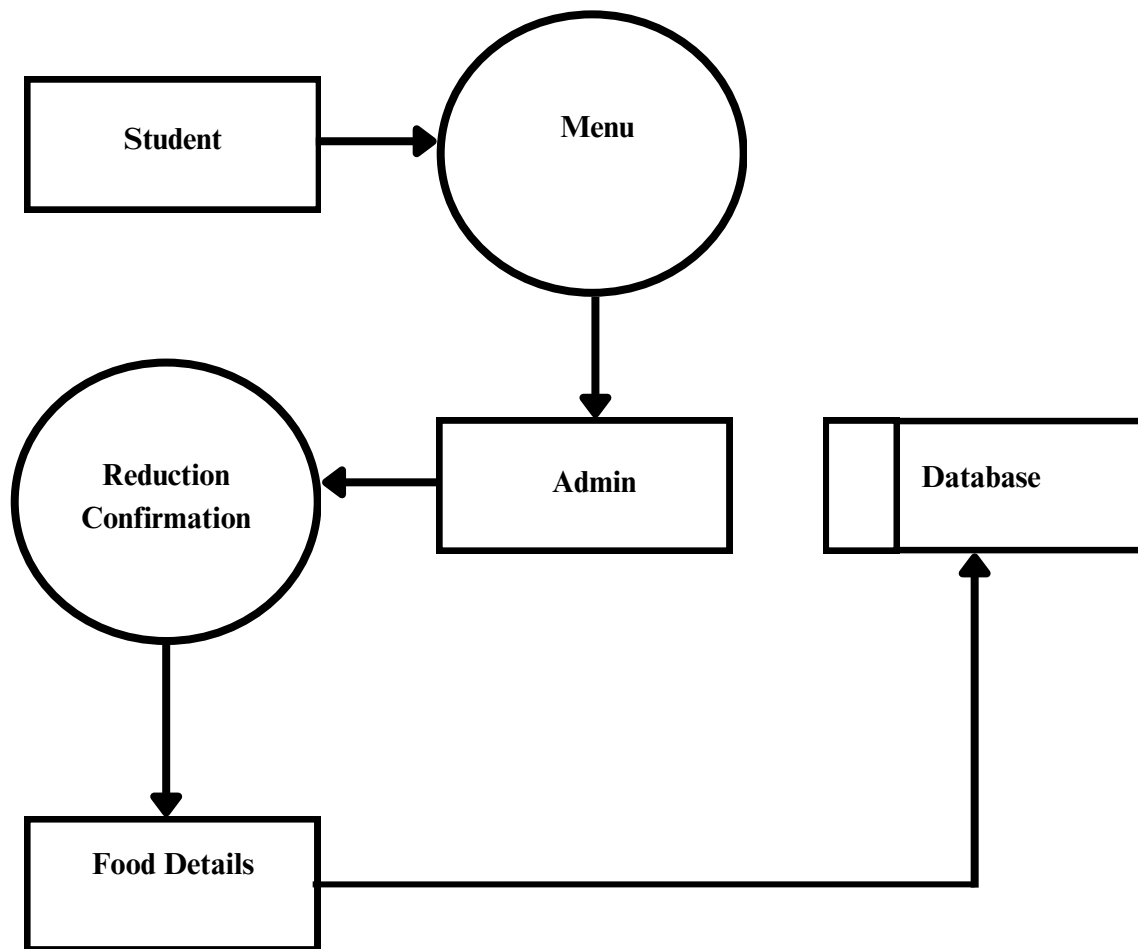Input / Output

Flow

Store the data

DATA FLOW DIAGRAM (DFD)

Level 0

Level 1

Level 2

```
┌──────────────┐          ╭──────────╮
│   Student    │ ───────▶ │   Menu   │
└──────────────┘          ╰────┬─────╯
                               │
                               ▼
  ╭──────────────╮        ┌──────────┐       ┌──┬──────────────┐
  │  Reduction   │ ◀───── │  Admin   │       │  │  Database    │
  │ Confirmation │        └──────────┘       └──┴──────────────┘
  ╰──────┬───────╯                                  ▲
         │                                          │
         ▼                                          │
  ┌──────────────┐                                  │
  │ Food Details │ ─────────────────────────────────┘
  └──────────────┘
```

Level 3

```
┌──────────────┐          ╭────────────────╮
│   Students   │ ───────▶ │   Reporting    │
└──────────────┘          │ Queries /      │
                          │ feedbacks      │
                          ╰───────┬────────╯
                                  │
                                  ▼
  ╭──────────────╮        ┌──────────┐       ┌──┬──────────────┐
  │  Analyzing   │ ◀───── │  Admin   │       │  │  Temporary   │
  │   Queries    │        └──────────┘       │  │  Database    │
  ╰──────┬───────╯                           └──┴──────────────┘
         │                                          ▲
         └──────────────────────────────────────────┘
```