

# **SDD Document (IEEE std 1016-2009)**

## **Group-1**

2022BCY0044 – Ansh Choudhary

## **Introduction**

### **1.1 Purpose of the Document**

The purpose of this SRS document is to provide requirements of the software system called CafeFlow+. which allows customers to buy and return plastic bottles and order food online from any cafe in college.

### **1.2 Scope**

- This software will handle plastic management at the cafes and food ordering and servicing and other payments for the cafe
- It will also provide the customer with a user-friendly interface for accessing services at the cafe.
- It will also provide a user-friendly interface for the shopkeepers to manage the services available at their cafe.

### **1.3 Definitions, acronyms, and abbreviations**

- SRS: Software Requirements Specification
- CafeFlow+: The name of the software system
- Customer/User: The primary user who will be able to place orders.
- Cafe Manager: The manager of the cafe and other staff who have permission to change the menu and manage plastic items and orders.
- Chef: The chef of the cafe who will prepare food and send notification for food preparation completion and confirm delivery.
- Administrator: The software managers who have super user access for the complete software.

# **System Overview**

## **2.1 System Context**

CafeFlow+ is a web and app-based application that allows customers to buy and return plastic bottles and order food online from any cafe in college. The system interacts with other external systems such as the college student database and payment gateway.

## **2.2 System Function**

CafeFlow+ provides the following functions:

- Search for available food items
- Ordering of food
- Plastic Item Purchasing
- Plastic Item Returning
- Payment processing
- Fine Generation

## **2.3 User Characteristics**

The CafeFlow+ is designed for two types of users: customers and cafe manager and chef. Customers are users who want to purchase food or any plastic item or return plastic item, while cafe managers are the users who confirm orders or accept plastic items physically or change menu. Also, chefs are the users who will prepare food and deliver it.

## **2.4 Constraints**

- The system must be able to handle a large number of concurrent
- users.

- The system must be secure to prevent unauthorized access to user information and payment details.
- The system must be scalable to handle an increasing number of orders and users.
- The system must be available 24/7 when the cafe is open except for maintenance downtime.

## **2.5 Assumptions and Dependencies**

- The college student database is reliable and provides up-to-date information on available students.
- The payment gateway is reliable and secure.
- The user, cafe owner and chef's internet connection are stable and meets the minimum requirements for accessing the system.

## **3.1 Overview**

This section describes the overall architecture of the Cafeflow+, including the hardware and software components, interfaces, and data flow between them.

## **3.2 Hardware Architecture**

The CafeFlow+ will run on an online server machine which will be connected to a local network. The server machine will have the necessary hardware resources, such as CPU, memory, and storage, to support the system's operations.

## **3.4 Data Architecture**

The bike rental system will use MongoDB to manage the system's data. The database will store user information, food orders, transaction history and cafe inventory.

### **3.5 Interface Architecture**

The CafeFlow+ will have three interfaces, one for users, one for cafe managers and one for the chef. The user interface will be a web and app-based application that allows users to buy and return plastic bottles and order food online from any cafe in college. The cafe manager interface will allow confirms orders or accept plastic item physically or change menu. The chef interface will allow him to see the sequence of ordered items and to send notification after preparing food and confirm for successful delivery.

### **3.6 Data Flow Diagrams**

The bike rental system's data flow diagrams will illustrate the data flow and interactions between the system's various components, including the database, cafe manager, chef, and user interfaces.

### **3.7 User Interface Design**

- The user interface design will be based on the principles of usability and user experience. The interface will be intuitive, easy to use, and visually appealing. It will provide users with all the necessary information they need to order food, purchase and return plastic items, view order history and issued plastic items.
- The cafe manager interface design will be based on the principles of usability and user experience. The interface will be intuitive, easy to use, and visually appealing. It will provide users with all the necessary information they need to confirm orders, view past orders and transaction history, take plastic items back, change menu.
- The chef interface design will be based on the principles of usability and user experience. The interface will be intuitive, easy to use, and visually appealing. It will provide users with all the necessary information they need to see a list of orders, send notifications for prepared items, and confirm delivery.

# Data Design

This section describes the data design of the CafeFlow+. It includes a description of the data used by the system, a data dictionary, and information about data storage, access, and manipulation.

## 4.1 Data Description

The bike rental system uses the following data:

- Customer/User : Name ,Contact , userid , Token Count , Wallet Amount, Payment History, Address, Gender, History
- Shopkeeper: Name ,Age, Gender, Customer Que, Tokens history, History , Transaction, Total Amount, Salary
- Kitchen Staff: Order que , Orders delivered, Chef assigned history
- Chef: Name , age, gender, experience, work que, Free, Salary history, Order delivered
- Order: Name, Price, Stock\_quantity, order\_avg\_rating
- Token: history,Number

## 4.2 Data Storage

The data for the café system will be stored in a MongoDB which is an easy to establish relational database. The database will provide data consistency, data integrity, and data security. The database will be hosted on a server with appropriate backup and recovery mechanisms.

**Data Access:** Data access to the CafeFlow+ will be provided through a app-based interface. The interface will allow authorized users to perform CRUD (Create, Read, Update, Delete) operations on the data. Access to the data will be controlled through authentication and authorization mechanisms.

## 4.3 Data Manipulation

Data manipulation in the café system will be performed using Mongo. The system will use stored procedures to ensure data consistency and data

integrity. The system will also implement appropriate error handling mechanisms to ensure invalid data is not entered.

## **Component design**

CafeFlow+ consists of following components:

### **5.1 Component Description**

#### **1.User Account Management Component:**

This component is responsible for managing user accounts for students and faculties. It handles user registration, assigns a unique ID to each user, tracks the count of plastic items owned by users, and enforces limits. It calculates and collects extra charges for plastic items, managing a common account for these transactions. The component also implements a fine system for users exceeding the allowed limit of plastic items. Additionally, it facilitates the return of plastic items at any cafe and processes refunds for the returned items.

#### **2.Menu and Ordering Component:**

This component handles the menu display for each cafe, allowing users to place orders and pay in advance. It incorporates acceptance or rejection functionality by the main shopkeeper, notifying cooking personnel for accepted orders.

#### **3.E-Wallet Component:**

The E-Wallet component is responsible for creating and managing e-wallets for each user.

#### 4.Notification System Component:

This component manages the notification system, sending notifications to users for order status updates. Notifications include order acceptance, rejection, and alerts when the food is ready for collection.

### **5.2 Component Functionality:**

#### 5.2.1. User Account Management Component:

- User Registration: Allows students and faculties to register and creates a unique ID for each user.
- Plastic Item Tracking: Manages the count of plastic items owned by each user and enforces limits.
- Extra Charge Collection: Calculates and collects extra charges for plastic items, depositing them into a common account.
- Fine Management: Implements a fine system for users exceeding the allowed limit of plastic items.
- Plastic Item Return: Provides a mechanism for users to return plastic items at any cafe.
- Refund Processing: Initiates the return of extra charges to the user's account upon successful plastic item return.

#### 5.2.2. Menu and Ordering Component:

- Menu Display: Shows the menu of each cafe to users.
- Order Placement: Allows users to select and order food items, paying in advance.

- Acceptance/Rejection: Main shopkeeper can accept or reject each ordered item.
- Notification System: Sends notifications to cooking personnel for accepted orders.
- Order Sequence and Timer: Cooking personnel view order sequence and timers for each item.
- Food Ready Notification: Allows cooking personnel to notify when the food is ready for collection.
- Order Confirmation: Verifies the order and completes the execution upon user confirmation.

#### 5.2.3. E-Wallet Component:

- E-Wallet Creation: Provides each user with an e-wallet upon registration.
- Top-Up Functionality: Allows users to top-up a certain amount into their e-wallets.
- Payment Processing: Enables users to make payments using the e-wallet for food orders.
- Limit Management: Sets and enforces limits for the e-wallet amount.
- Alternative Payment: Acts as an alternative payment option when the user's bank server is down.

#### 5.2.4. Notification System Component:

- Order Status Notifications: Sends notifications to users for order acceptance, rejection, and when the food is ready for collection.
- Role Management: Manages user roles and permissions for different functionalities.

### **5.3 Component Interfaces:**

#### 5.3.1. User Account Management Component Interfaces:



- Interface with Plastic Item Tracking: Manages the count of plastic items owned by users and enforces limits.
- Interface with Extra Charge Collection: Calculates and collects extra charges for plastic items, depositing them into a common account.
- Interface with Fine Management: Implements a fine system for users exceeding the allowed limit of plastic items.
- Interface with Plastic Item Return: Facilitates the return of plastic items at any cafe and processes refunds for returned items.

#### 5.3.2. Menu and Ordering Component Interfaces:

- Interface with Acceptance/Rejection Functionality: Allows main shopkeeper to accept or reject each ordered item.
- Interface with Notification System: Sends notifications to cooking personnel for accepted orders.
- Interface with Timer and Order Tracking: Provides information on the order sequence, timers, and order status for each food item.
- Interface with Order Confirmation: Verifies the order and completes the execution upon user confirmation.

#### 5.3.3. E-Wallet Component Interfaces:

- Interface with Top-Up Functionality: Allows users to top-up their e-wallets with a specified amount.
- Interface with Payment Processing: Enables users to make payments using their e-wallets for food orders.
- Interface with Limit Management: Enforces limits on the e-wallet amount.

#### 5.3.4. Notification System Component Interfaces:

- Interface with User Account Management: Sends notifications to users for order status updates.

- Interface with Menu and Ordering: Alerts users when the food is ready for collection.

## **INFRASTRUCTURE**

### **6.1 HARDWARE INFRASTRUCTURE**

Cafeflow+ will be hosted on a server in the college server room. The server will have the following specifications:

- Intel i7 12<sup>th</sup> gen
- Ram: 16 GB
- Storage: 1 TB
- OS: Linux

Apart from the server, the cafe staff will need to have the following hardware for smooth operation:

- 2 x computer with following specifications:
  - Intel i3 10<sup>th</sup> gen+
  - Ram: 4GB+
  - Storage: 256GB+
  - OS: Any OS with a browser

### **6.2 SOFTWARE INFRASTRUCTURE**

The software infrastructure for cafeflow+ will have the following components:

- OS: Linux
- Web server: Apache HTTP server
- Database management software: MongoDB
- Programming languages: Dart, JavaScript
- Docker

### **6.3 NETWORK INFRASTRUCTURE**

The cafe devices will be connected to the internet using already available routers on the campus. The devices will be connected to the server using a VPN to tunnel the network traffic and prevent any data sniffing and manipulation. The network uses the already available components inside the campus like routers, switches, Fortinet, etc.

### **6.4 SECURITY INFRASTRUCTURE**

To ensure data security, the following steps need to be taken:

- User Authentication: All the users trying to log into the system will have to authenticate themselves. Only authorized users will be allowed to access the application.
- Data encryption: All the important data will be encrypted using modern encryption standards like AES and RSA. Additionally, the data from cafe to servers will be encrypted using a VPN.
- Regular backups: All the transactions and other important data will be backed up to a backup server for additional safety.
- Firewall: The existing firewall can be modified specifically for the application to detect and prevent any malicious activities.