Dining Hall Management System

Software Requirement Specifications

Project by

Mohith S | Dev Kathuria | Jackson Jose

Table of Contents

1. Introduction

The Dining Hall Management System version 1.0 is a complete package for managing the functioning of a dining hall in any institution. This software is designed keeping in mind everyone (at all hierarchy) who would be involved to make sure the right people get the required access/control over the Dining Hall.

1.1 Purpose

The purpose of this software is to automate maximum work of managing the Dining Hall, give better and useful insights to the authorities (like the current number of orders, change/add food items, enable/disable ala-carté), ease the process of ordering for the students. With a user-friendly interface, the entire process for every person involved will be made simple and convenient.

1.2 Document conventions

DH	Dining Hall
FB	Firebase
DHMS	Dining Hall Management System
QR code	Quick Response Code

1.3 Intended Audience

Our Product is targeted towards any vendor which provides mess services in environments such as schools, colleges, universities or offices, where the customers dine regularly.

1.4 Contact information

- Mohith S. (Roll No: 1710110219)
    - Specialization: iOS App Designing - Development - Testing
- Dev Kathuria (Roll No: 1710110103)
    - Specialization: Web App Designing - Development - Testing
- Jackson Jose(Roll No: 1710110152)
    - Specialization: Android App Designing - Development - Testing

2. The Overall Description

2.1 Product Perspective

- This product consists of four different sub-systems working together in coherency to run. An iOS app, an android app, an admin management software and a software to check receipts.
- This product helps the vendors' customers to order food from wherever they are, without having the need to be physically present.
- This system requires mobile devices in the food counter.
- This system uses cloud database, hence will require only minimal space in the devices running our product.
- The vendor’s customers must have an android or iOS device to send requests from their devices.

2.2 Product Functions

The major functions DHMS performs are described as follows:

1. WebApp
    a. Login Functionality for various employees(Admin/ Chef/ Bakery/ Rollmaal)
    b. Displays orders for rollmaal and Bakery Counter to ease and fasten the process of completing the order in a systematic way
    c. Admin functionality includes setting the mess menu for the day and displaying various insights related to the orders placed
    d. Ala-Carte Functionality includes
        i. Setting approximate time for the orders to get ready
        ii. Setting order is ready

iii. Create a Batch of various items from different Ala-Carte orders to allow chefs to make the dishes in an efficient and fast manner

2. Mobile App
    a. Ordering system to add items from the menu to cart.
    b. Method to pay and a cash wallet, in which you can add money.
    c. A mechanism to check previous purchases.
    d. A QR system to describe the order.
    e. A method to let customers know when their order is ready.
    f. Feedback system.

2.3 User Classes and Characteristics

There are different classes of users who will be interacting with the system. The intended users of the software are as follows :

Students: Students are the prime consumer in any educational institution. The Android and iOS applications are designed and developed for them. The application just allows them to place orders, gives them order history (for better insights of spendings).

Vendor (Admin): The admins (at the dining hall) will have full access to all the functions. They control the daily menu, the availability of any item at any given point,

2.4 Constraints

1. Only one transaction related to bakery/rollmaal items needs to be processed at a time to keep the quantity available updated on the main database with correct values.
2. Cross platform availability(Specifically Android and iOS)
3. The number of invalid pin entries must not exceed 3. After that, the user will have to login again through his/her gmail account

2.5 Assumptions and Dependencies

The requirements stated in the SRS could be affected by the following factors

1. One major dependency that the project might face is the changes that needs to he incorporated with the changes in the Dining Hall vendor policies regarding different services. As the policies changes the system needs to be updated with the same immediately. A delay in doing the same will result in tremendous loss to the bank. So this should be changed as and when required by the developer.
2. At this stage, no quantitative measures are imposed on the software in terms of speed and memory although it is implied that all functions will be optimized with respect to speed and memory.

3. External Interface Requirements

3.1 User interfaces

The interface provided to the user should be very user-friendly one and it should provide an interactive help for each of the service listed along with modern UI/UX features.
The interface provided will be tab driven and the following screens would be provided:

- Login Screen is provided in the beginning to login with college provided email account which corresponds to their mess id as well
- Order Now: It shows the items(along with their prices) available to order, it also shows the menu for buffet and allows user to add the desired quantity of any item in their cart
- Receipt: It contains the items ordered(paid), it can be directly shown at the item counter to receive the items, It also dynamically updates itself if more order(s) are placed
- Cart: It contains the item(s) selected along with their prices and the sum total of all of them, payment is done on this screen.
  It also notifies the user in case of insufficient balance.
- Menu: Displays the buffet menu for the day
- Passbook: It shows the history of orders placed, user information and the monthly balance which remains

WebApp

- Login Screen: Depending weather it is the bakery id/rollmaal id/chef id or the admin id, different access privileges will be provided.

- AlaCarte/Rollmaal Screen: Displays the orders placed in chronological order and a seperate table which shows item/quantity pairs for current order.
- Ala-Carte: It shows ala-carte orders in chronological order and creates batches to be cooked. Tracking time is inserted manually.
- Admin: Displays data insights from orders and option to change today's menu.

3.2 Hardware interfaces

- A mobile devices is required to access the wallet and order item. This is for the customer of the vendor.
- Every stall requires a mobile device with a camera to scan the receipt consisting of the QR code.

3.3 Software interfaces

- An android or iOS app with the customers of the vendor to order food and check menu etc.
- A WebApp with the admin of the mess, to set the day's menu, the gather insights, to check incoming orders.
- A WebApp which sorts food items into batches for the kitchen staff the plan their working in an efficient way.
- An android app which scans the receipt and informs the stall staff which items are ordered.

4. System Features

4.1 Secure Transaction Implementation

The application involves transactions that has to be verified at various levels before confirming the order. The orders are placed in a 'queue' of pending transactions. Once it is verified that the student has enough balance in his account and the availability of the food item, the confirmed orders are placed in another 'queue' of confirmed transactions.

4.2 QR code receipt

To make sure that the digital receipt can't be forged, we use QR codes within which the order ID, Date and Time of the order are embedded. Once this gets scanned at the counter, the QR code gets removed from the database and will no longer be valid.

4.3 Security Measure (Biometric Validation)

Before a transaction is sent processes, the mobile app verifies the user, and uses biometric identification (if available natively on the device), else uses a pin. This is done to ensure that the transaction does not occur in an unauthorised way.
After 5 failed attempts, the account automatically logs out, and will have to be authenticated with google sign in.