Software Requirements Specification

for

Hostel Mess Management System

Prepared by

Prasun Kumar (1801126)
Prateek Jain (1801127)
Shreyansh Singh Tomar (1801169)

Indian Institute of Information Technology, Guwahati

26 January 2021

Contents

Chapter 1

Introduction

1.1 Purpose

The purpose of developing this system is to provide a robust, highly reliable, feature-rich and user-centric centralised management system for hostel messes. The software would be a mobile as well as web based application that would be available to the residents of a hostel or similar establishment. This software would move much of the workflow to a digital platform.

1.2 Product Scope

It will automate the work like maintaining records of mess dues and payment history, coupon booking, lodging complaint and providing feedback for both residents and caterers. The system would be flexible enough to easily adapt to any hostel mess and provide services to its users.

1.3 Current System

The way the mess currently operates introduces lots of friction for the caterers to provide the services as well as the residents to avail those services, some of which are listed below:

- Currently, there is no way for the caterers to estimate the number of students who will attend the mess at any point of time. This leads to overestimation or underestimation of the food preparation which in turn leads to food wastage or scarcity.
- Residents tend to skip some meals. However, they still have to pay for the entire month of service.
- There is no standardized procedure for handling complaints and feedback. Students find it difficult to lodge complaints. The mess manager is not easily approachable. Changes take long time to take effect.

1.4 Benefits of proposed system

The students and visitors will be benefited from the app as follows:

- Coupon based system would allow students to pay for only those meals that they eat.
- With the centralized complaints system, it will be easy for them to lodge complaints and offer suggestions. The app will help them track the status of their complaints.

The caterers will be benefited from the system as follows:

- The app will provide the estimate of the students who will attend the mess making it is easier for them to estimate the quantity of raw materials required. This will help overcome situations of food wastage and scarcity.
- Feedback and complaints on the extended services can be easily monitored.
- Do away with manual system of maintaining the paperwork.
- Helps in advance planning.

Chapter 2

Overall Description

2.1 Product Description

A hostel mess is a vital part of an institute's infrastructure. It provides the people living far from their homes with the basic necessity of life. With such parts of a system, which are vital and essential for daily working, comes the responsibility of making it work as smoothly as possible.

There are quite a few problems in a mess that could be solved or made simpler using a specialised management software. The software could deal with multiple modes of operation such as Food Coupons as well as pay-per-month services to students. Different charges and bills can be issued to different people in such a software using required authentication features which would generally create a problem in implementing without a software on a short-staffed or busy day.

Lodging complaints, daily and anytime accessible bills as well as keeping track of past payments and dues could be simplified significantly using such a software minimising large and repetitive paperwork. Furthermore, the data generated daily could be analysed and inferred to get a prediction of food needed, minimising wastage. This would be helpful to the caterer as well as to our society in general which faces with wastage along with scarcity of food.

2.2 Product Features

- Sign Up and Login
- *Plans:*
  1. Coupon Based
  2. Pay per Month
- Payment Interface.
- *Tracking:*
  1. Payment History
  2. Coupons Left
  3. Days attended
- Lodge Complaints/ Provide feedback.
- Pause/ Resume Pay Per Month service.
- Weekly Reports and Analytics to Caterers.
- Complaint Handling for Caterers.
- Hassle Free Paperless work for Mess Management.

2.3 User Classes and Characteristics

User Class	Intended uses
Students	Ability to buy coupons and pay bills
Visitors	Buy coupons
Admin	Generate bills, collect payments and complaints

2.4 Constraints

- An employee of the mess would need to be present during food serving to validate coupons using the app by scanning the codes.
- The size of the media uploaded during complaint registration would need to be capped to a limit to minimize server costs.

2.5 Assumptions

- Every consumer in the mess would carry a mobile phone with them.
- The establishment has good internet connectivity which would be required by the mess employee during coupon or pass validation.
- The price of a meal using a coupon is greater than the price by using the pay-per-month service. This will ensure the caterer doesn't suffer any losses by students opting the coupon system.

Chapter 3

External Interface Requirements

3.1 User Interface Requirements

- A register page for users to create their accounts.
- A login page for users to submit their credentials and log into their accounts.
- A reset password page to be used when user forgets their password.
- A page for buying coupons available only if the user has opted for the coupon system
- A billing dashboard where users can see their due and past bills and make payments.
- A complaint lodging page to register complaints.
- A module on the admin side to validate codes of coupons.
- A complaint tracking page on admin side.

3.2 Software Interface Requirements

For development of the product, the following requirements have been identified

Software Type	**Component**
Deployment Server OS	Linux
Databases	PostgreSQL
Development tools	VSCode, PyCharm IDE, Android Studio
Design Tool	Figma

For using the product the following components are supported

Software Type	**Component**
Operating System	Cross Platform
Web Browser	Mozilla Firefox, Google Chrome

Chapter 4

Functional Requirements

4.1 Mess Users

- Users could register their accounts using a valid college email address or as a visitor.
- Users could opt in for coupon-based system or pay-per-month service for a month.
- Users can pay bills using the inbuilt payment interface.
- Users can book coupons from inside the app.
- Users can keep track of all their past coupons and paid bills.
- Users can lodge complaints or give suggestions regarding the food quality and cleanliness in the establishment.
- If opted for the pay-per-month service, users would be able to apply for a leave if they are leaving for more than five days.

4.2 Admin

- Caterers would be provided with comprehensive weekly report about the mess allowing to adjust their food preparation with actual consumption. This will minimize food wastage.
- The mess management would be able to handle complaints in a standardized and centralised manner.
- The mess management would be able to avoid the unnecessary paperwork that is inherent with manual operation of such a system.

Chapter 5

Non Functional Requirements

5.1 Flexibility

The software would be flexible enough to adapt to any institution with minimal changes to the code.

5.2 High Peak Performance

The system would be designed to handle bursty traffic which would peak at around 3 times a day. The system would be maintained in off hours to ensure smooth functioning during peak traffic.

5.3 Data Integrity

The records of transactions kept by the system would be stored reliably and would be backed up regularly to ensure that the data is reliable.

5.4 Reliability

The system architecture should be simple to ensure that the software is reliable and bug-free.

5.5 Security

The payment module would need to ensure secure payments

5.6 Easy to use interface

Since the product is intended for a widespread users, the user interface should be easy to use. This will ensure maximum adoption of the system.

Chapter 6

References

The following references were used during preparation of this document

- Sample SRS template provided in the course of Software Engineering.
- Pressman, Roger S. Software Engineering: A Practitioner's Approach, page-136. Eighth edition, McGraw-Hill Education, 2015.
- http://users.encs.concordia.ca/~eshihab/teaching/slides/srs_template_sep14.pdf