

Fusion ERP

Software Requirements Specification

for

SA-2-MESS MANAGEMENT

Faculty Mentor:

Dr. Vijaypal Singh Rathor

Team Mentor:

Gaurang Bhutani - [21BCS085](#)

Team Members:

Chandrashekhar Gouda - [21BCS064](#)

Manish Kumar - [21BCS128](#)

Manoj Kumar - [21BCS129](#)

Mohammad Adil - [21BCS133](#)

Navneet - [21BCS143](#)

Fusion ERP

1. Introduction

1.1 Introduction about the Fusion – A brief Description

FusionIIIT stands as a testament to the seamless integration and automation of diverse functions within PDPM Indian Institute of Information Technology, Design and Manufacturing, Jabalpur. Crafted with precision using Flutter and a client-optimized programming language named 'Dart', this initiative is a student-driven endeavor designed to elevate the institute's operational landscape. Encompassing everything from efficient administration management to academic prowess and miscellaneous departmental tasks, FusionIIIT is a holistic solution that harmonizes the intricacies of campus life.

Imagine it as a digital wizard that takes care of everything, from organizing the administrative stuff to making academics smoother and making mess management easy. It's not just limited to the usual tasks; FusionIIIT jumps into various departments and sections, making sure every corner of campus life runs smoothly.

In the admin side, it handles the complicated paperwork and processes. For academics, it brings a digital touch, making learning and managing courses easier. But it doesn't stop there; FusionIIIT is like a friendly companion for all the different parts of the campus, making sure everything works well.

In simpler terms, FusionIIIT is not just a tool – it's a helpful friend, making life at PDPM IIITDM Jabalpur more organized and enjoyable for everyone.

1.2 Purpose of the module

This document outlines the requirements for the "Central Mess Management System" integrated into the Institute's ERP. The primary goal of our software is to streamline and automate the operations of the Central Mess, offering a sophisticated platform for seamless interaction between users and administrators. It will elaborate on the system's purpose, features, interfaces, functionalities, operational constraints, and the nature of interactions within the system.

1.3 Scope of the module

The module is developed for the convenience of registered students, the Mess Caretaker, the Mess Warden, and Students at PDPM IIITDM Jabalpur. It's an Android application dedicated to the efficient management of the institute's mess.

Students can leverage the app to sign in, browse the menu, apply for leave, order special request meals or , submit feedback, and view announcements.

The app's administrative capabilities include tracking student bills, refreshing the menu, monitoring stock and budget details, overseeing guest entries, and addressing student requests.

This module is designed to simplify mess operations, enable remote access to its services, and foster better communication among stakeholders. A relational database will handle data management, storing details such as menu items, bills, student IDs, consumption data, and feedback.

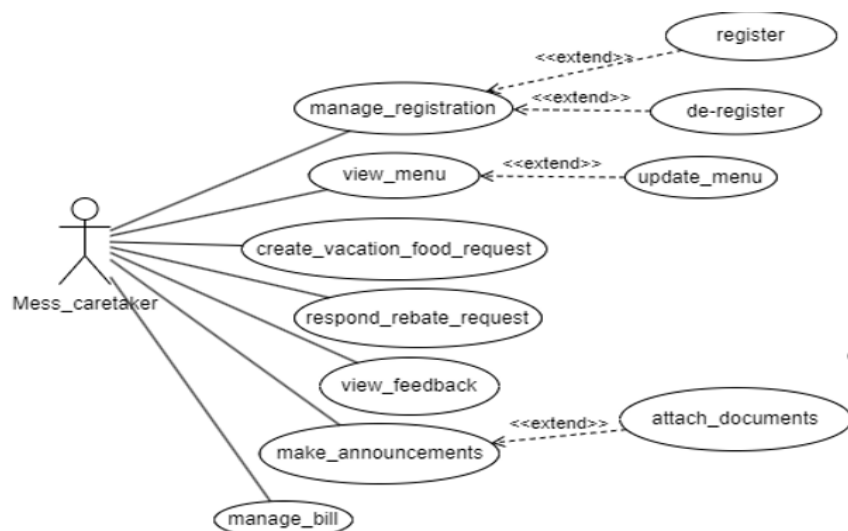
2. User/Actor Description(characteristics)

i) Mess_caretaker:

The Mess_caretaker is responsible for the overall management of the mess.

Functionalities:

- Manages registration and de-registration of students.
- Views and updates the menu.
- Creates vacation food requests and responds to rebate requests.
- Views feedback and makes announcements.
- Manages bills.



ii) Mess_warden:

The Mess_warden oversees the operations of the mess and has access to various reports and feedback.

Functionalities:

- Views the menu, feedback, registrations, and announcements.
- Make announcements.
- Searches or generates reports.



iii) Student:

The Student is the user of the mess services and has various options for interaction with the system.

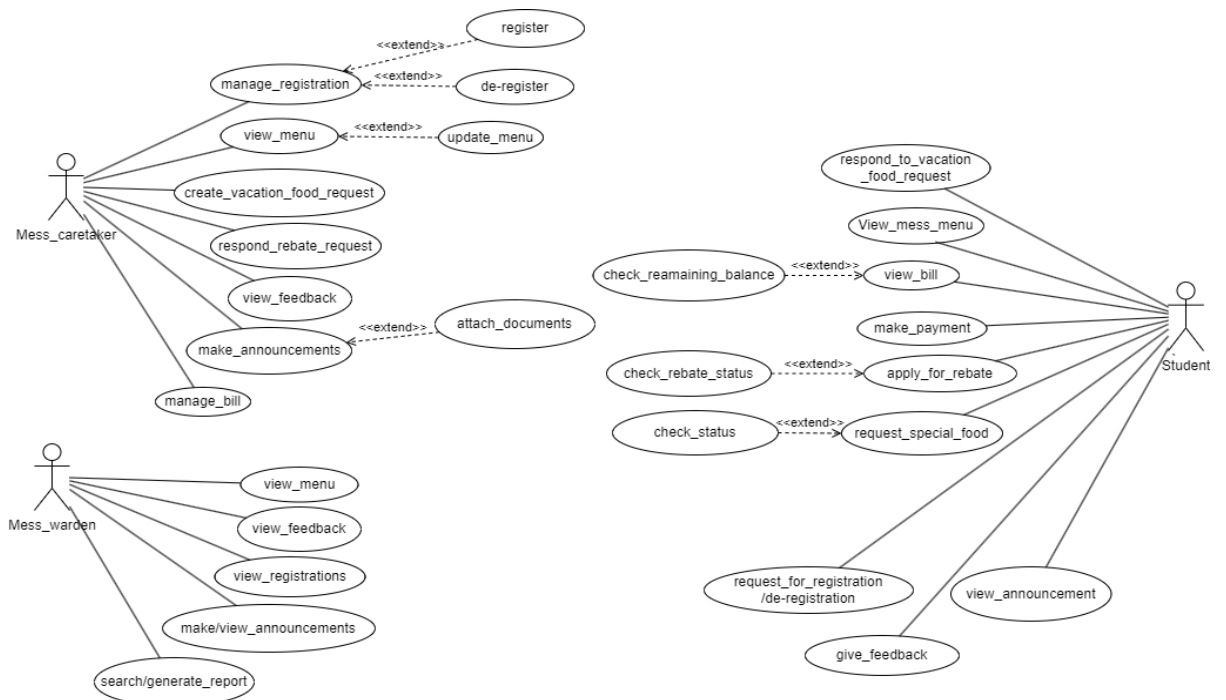
Functionalities:

- Registers, de-registers, checks remaining balance, views bill, makes payment, applies for rebate, checks rebate status, requests special food and checks its status.
- Responds to vacation food requests.
- Views mess menu and announcements.
- Requests for registration/deregistration and gives feedback.



3. Functional Requirements

3.1 Use case Diagram



3.2 Use case description:

- Use case description for Student is listed below:

Use case #1

| | | |
|------------------------------|--|--|
| UC ID | UC#1 | |
| Use case Name | view_menu | |
| Description | The " view_menu " use case allows the Student to view the mess menu for both Mess-1 and Mess-2 through the Fusion portal. | |
| Actor | Student | |
| Precondition | The actor should be logged in into the system. | |
| Main Flow | 1 | The actor navigates to the "Mess Menu" section. |
| | 2 | The system displays the option to select the Mess for which the menu is to be displayed. |
| | 3 | The actor selects a mess to view the menu. |
| | 4 | The actor can view or download the menu of the selected mess. |
| Post conditions | NIL | |
| Alternate Flow | NIL | |
| Sub Flow | NIL | |
| Global Alternate Flow | GA1 | The actor returns to the main screen. |

Use case #2

| | | |
|-----------------------|--|---|
| UC ID | UC#2 | |
| Use case Name | view_announcement | |
| Description | The "view_announcement" use case allows the Student to view the announcements through the Fusion portal. | |
| Actor | Student | |
| Precondition | The actor should be logged in into the system. | |
| Main Flow | 1 | The actor navigates to the "Announcements" section. |
| | 2 | The actor views the announcements sorted date wise. |
| Post conditions | NIL | |
| Alternate Flow | NIL | |
| Sub Flow | NIL | |
| Global Alternate Flow | GA1 | The actor returns to the main screen. |

Use case #3

| | | | |
|-----------------|---|---|---|
| UC ID | UC#3 | | |
| Use case Name | request_special_food | | |
| Description | The "request_special_food" use case allows the Student to request special foods on different occasions through the Fusion portal. | | |
| Actor | Student | | |
| Precondition | The actor should be logged in into the system. | | |
| Main Flow | 1 | The actor navigates to the "Request Special Food" section. | |
| | 2 | The actor selects the “Request Special Food” option to request. | |
| | 3 | The actor fills the form and submits. | |
| | 4 | The actor can check the status of previous requests. [A1]. | |
| Post conditions | NIL | | |
| Alternate Flow | A1 | 1 | The actor navigates to the “Request History” option. |
| | | 2 | The system displays the list of previous requests and their statuses. |

| | | |
|------------------------------|-----|---------------------------------------|
| Sub Flow | NIL | |
| Global Alternate Flow | GA1 | The actor returns to the main screen. |

Use case #4

| | | |
|------------------------------|--|---|
| UC ID | UC#4 | |
| Use case Name | give_feedback | |
| Description | The student can give feedback for the services provided by the mess and also view previous feedback. | |
| Actor | Student | |
| Precondition | The student must be logged in. | |
| Main Flow | 1 | The student clicks on the 'Feedback' button. |
| | 2 | The student fills in the feedback form and submits it. |
| | 3 | The student can view the previous feedback. |
| Post conditions | The feedback details are added in the database and are notified to the mess warden and mess caretaker. | |
| Alternate Flow | NIL | |
| Sub Flow | NIL | |
| Global Alternate Flow | GA1 | The actor returns to the main screen abandoning any intermediate operation. |

Use case #5

| | | |
|----------------------|---|--|
| UC ID | UC#5 | |
| Use case Name | request_registration/deregistration | |
| Description | The "request_registration/deregistration" use case allows the Student to request register/deregister to mess through the Fusion portal. | |
| Actor | Student | |
| Precondition | The actor should be logged in into the system. | |
| Main Flow | 1 | The actor navigates to the "Register/Deregister" section. |
| | 2 | The system displays to select register and make payment [UC#8] if they are deregistered and deregister if they are registered. |
| | 3 | The actor selects register to register in mess and deregister to deregister in central mess. |
| | 4 | The actor views their status. |
| Post | The register/deregister request will be sent to the mess caretaker. | |

| | | |
|------------------------------|-----|---------------------------------------|
| conditions | | |
| Alternate Flow | NIL | |
| Sub Flow | NIL | |
| Global Alternate Flow | GA1 | The actor returns to the main screen. |

Use case #6

| | | |
|------------------------------|--|---|
| UC ID | UC#6 | |
| Use case Name | view_bill | |
| Description | The student gets to view the bill for the month and also the positive remaining balance. | |
| Actor | Student | |
| Precondition | The actor must be logged in. | |
| Main Flow | 1 | The actor clicks on the “Mess Bill”. |
| | 2 | A list is displayed on the screen. |
| | 3 | The actor can view their monthly bill. |
| | 4 | The bill details for the month and positive balance (if any) are displayed. |
| Post conditions | NIL | |
| Alternate Flow | NIL | |
| Sub Flow | NIL | |
| Global Alternate Flow | GA1 | The actor returns to the main screen. |

Use case #7

| | | |
|----------------------|--|---|
| UC ID | UC#7 | |
| Use case Name | apply_for_rebate | |
| Description | The student can fill a leave application form stating the duration and purpose of leave which will be approved/rejected by the mess caretaker. | |
| Actor | Student | |
| Precondition | The actor must be logged in. | |
| Main Flow | 1 | The student clicks on the ‘Apply for Rebate’ button |
| | 2 | A leave application form is displayed. |

| | | | |
|-----------------------|---|--|---|
| | 3 | The actor fills in the necessary details and submits it. | |
| | 4 | The actor can view the status of previous rebates. [A1] | |
| Post conditions | The leave request details are added in the database and are notified to the mess caretaker. | | |
| Alternate Flow | A | 1 | The actor navigates to the “Rebate History” section. |
| | 1 | 2 | The system displays the previous rebate requests along with the status. |
| Sub Flow | NIL | | |
| Global Alternate Flow | G A 1 | The actor returns to the main screen. | |

Use case #8

| | | |
|-----------------------|--|---|
| UC ID | UC#8 | |
| Use case Name | make_payment | |
| Description | The student can make payment on the dashboard. | |
| Actor | Student | |
| Precondition | The actor must be logged in. | |
| Main Flow | 1 | The actor clicks on “Make Payment”. |
| | 2 | The system gets redirected to the Payment Gateway Portal. |
| Post conditions | The changes made should get reflected on the system. | |
| Alternate Flow | NIL | |
| Sub Flow | NIL | |
| Global Alternate Flow | GA 1 | The actor returns to the main screen. |

Use case #9

| | | | |
|---------------|--|---|--|
| UC ID | UC#9 | | |
| Use case Name | respond_to_vacation_food_request | | |
| Description | The "respond_to_vacation_food_request" use case to respond to the vacation food request through the Fusion portal in vacation. | | |
| Actor | Student | | |
| Precondition | The actor should be logged in into the system. | | |
| Main Flow | 1 | The actor responds based on their availability. | |
| | 2 | The actor will be registered in the mess during vacation after a positive response. | |
| Post | NIL | | |

| | | |
|------------------------------|-----|---------------------------------------|
| conditions | | |
| Alternate Flow | NIL | |
| Sub Flow | NIL | |
| Global Alternate Flow | GA1 | The actor returns to the main screen. |

- Use case description for Mess Caretaker is listed below:

Use case #10

| | | |
|------------------------------|--|---|
| UC ID | UC#10 | |
| Use case Name | respond_rebate_request | |
| Description | Mess Caretaker handles the rebate requests. | |
| Actor | Mess Caretaker | |
| Precondition | The actor must be logged in. | |
| Main Flow | 1 | The caretaker clicks on the “Rebate Requests”. |
| | 2 | A list of students is displayed who had applied for rebate, along with approve or deny options. |
| Post conditions | The rebate requests of the students will be approved or denied by the caretaker. | |
| Alternate Flow | NIL | |
| Sub Flow | NIL | |
| Global Alternate Flow | GA 1 | The actor returns to the main screen. |

Use case #11

| | | |
|----------------------|---------------------------------------|---|
| UC ID | UC#11 | |
| Use case Name | make_announcements | |
| Description | Mess Caretaker makes an announcement. | |
| Actor | Mess Caretaker | |
| Precondition | The actor must be logged in. | |
| Main Flow | 1 | The caretaker clicks on the “Make Announcements”. |
| | 2 | Fill the form displayed and click ‘make announcement’.. |

| | | |
|------------------------------|---|---------------------------------------|
| Post conditions | The new announcements are added to the database and are notified to the students and the Mess Warden. | |
| Alternate Flow | NIL | |
| Sub Flow | NIL | |
| Global Alternate Flow | GA1 | The actor returns to the main screen. |

| | | |
|------------------------------|---|---|
| UC ID | UC#12 | |
| Use case Name | manage_bill | |
| Description | Mess Caretaker manages the bill. | |
| Actor | Mess Caretaker | |
| Precondition | The actor must be logged in. | |
| Main Flow | 1 | The actor clicks on the 'Manage Bill' button. |
| | 2 | The system displays the list of registered students along with their corresponding mess bill. |
| | 3. | The actor can send the corresponding mess bill to respective students. |
| Post conditions | The changes made are reflected on the system. | |
| Alternate Flow | NIL | |
| Sub Flow | NIL | |
| Global Alternate Flow | GA1 | The actor returns to the main screen. |

Use Case#12

Use case #13

| | | |
|------------------------|--|--|
| UC ID | UC#13 | |
| Use case Name | manage_registration | |
| Description | The user gets to login and avail the facilities of the web based mess management system. | |
| Actor | Mess Caretaker | |
| Precondition | The actor must be logged in | |
| Main Flow | 1 | The actor clicks the `Registrations` button. |
| | 2 | The system displays the list of requests by students to get registered/ deregistered students. |
| | 3 | The actor can accept/reject the requests.[A1] |
| Post conditions | Session starts for the logged in user. | |
| Alternate Flow | A1 | The student is prompted to Enter the login details again. (mismatch credentials) |
| | A2 | The user returns to initial screen (cancel) |
| Sub Flow | NIL | |

| | | |
|------------------------------|-----|---------------------------------------|
| Global Alternate Flow | GA1 | The actor returns to the main screen. |
|------------------------------|-----|---------------------------------------|

Use case #14

| | | |
|------------------------------|--|---|
| UC ID | UC#14 | |
| Use case Name | create_vacation_food_request | |
| Description | Mess Caretaker create for food request in vacation | |
| Actor | Mess Caretaker | |
| Precondition | The actor must be logged in. | |
| Main Flow | 1 | The actor clicks on the 'Create Food Request' button. |
| Post conditions | The students respond to vacation food requests by the actor according to their availability. | |
| Alternate Flow | | NIL |
| Sub Flow | | NIL |
| Global Alternate Flow | GA 1 | The actor returns to the main screen. |

Use case #15

| | | |
|------------------------------|---|---|
| UC ID | UC#15 | |
| Use case Name | view_feedback | |
| Description | The "View Feedback" use case allows the Mess Caretaker to view the feedback by the students for both Mess-1 and Mess-2 through the Fusion portal. | |
| Actor | Mess Caretaker | |
| Precondition | The actor should be logged in into the system. | |
| Main Flow | 1 | The actor navigates to the "View Feedback" section. |
| | 2 | The system displays the list of all the feedback. |
| | 3 | The actor can select any feedback and review it. |
| Post conditions | NIL | |
| Alternate Flow | NIL | |
| Sub Flow | NIL | |
| Global Alternate Flow | GA1 | The actor returns to the main screen. |

Use case #16

| | |
|--------------|--------------|
| UC ID | UC#16 |
|--------------|--------------|

| | | |
|------------------------------|--|---|
| Use case Name | view_menu | |
| Description | The "View Menu" use case allows the Mess Caretaker to view the mess menu for both Mess-1 and Mess-2 through the Fusion portal. | |
| Actor | Mess Caretaker | |
| Precondition | The actor should be logged in into the system. | |
| Main Flow | 1 | The actor navigates to the "Mess Menu" section. |
| | 2 | The system displays the option to select either the Mess for which the menu is to be displayed. |
| | 3 | The actor selects a mess to view the menu. |
| | 4 | The actor views or downloads the menu of the selected mess. |
| | 5 | The actor can also update the menu. [A1] |
| Post conditions | NIL | |
| Alternate Flow | A 1 | 1 The actor navigates to the "Update Menu" option. |
| | | 2 The system displays a form for updating the menu. |
| | | 3 The actor fills the form and submits it to update the menu. |
| Sub Flow | NIL | |
| Global Alternate Flow | G A 1 | The actor returns to the main screen. |

- Use case description for Mess Warden is listed below:

Use case #17

| | | |
|------------------------|--|---|
| UC ID | UC#17 | |
| Use case Name | view_menu | |
| Description | The " view_menu " use case allows the Mess Warden to view the mess menu for both Mess-1 and Mess-2 through the Fusion portal. | |
| Actor | Mess Warden | |
| Precondition | The actor should be logged in into the system. | |
| Main Flow | 1 | The actor navigates to the "Menu Menu" section. |
| | 2 | The system displays the option to select either the Mess for which the menu is to be displayed. |
| | 3 | The actor selects a mess to view the menu. |
| | 4 | The actor views or downloads the menu of the selected mess. |
| Post conditions | NIL | |
| Alternate | NIL | |

| | | |
|------------------------------|------|---------------------------------------|
| Flow | | |
| Sub Flow | NIL | |
| Global Alternate Flow | GA 1 | The actor returns to the main screen. |

Use case #18

| | | |
|------------------------------|---|--|
| UC ID | UC#18 | |
| Use case Name | view_feedback | |
| Description | The " view_feedback " use case allows the Mess Warden to view the feedback by the students for both Mess-1 and Mess-2 through the Fusion portal. | |
| Actor | Mess Warden | |
| Precondition | The actor should be logged in into the system. | |
| Main Flow | 1 | The actor navigates to the "Feedbacks" section. |
| | 2 | The system displays the list of all the feedbacks. |
| | 3 | The actor can select any feedback and review it. |
| Post conditions | NIL | |
| Alternate Flow | NIL | |
| Sub Flow | NIL | |
| Global Alternate Flow | GA1 | The actor returns to the main screen. |

Use case #19

| | | |
|----------------------|--|--|
| UC ID | UC#19 | |
| Use case Name | view_registrations | |
| Description | The " view_registrations " use case allows the Mess Warden to view the registrations done by the students of all the programs for both Mess-1 and Mess-2 through the Fusion portal. | |
| Actor | Mess Warden | |
| Precondition | The actor should be logged in into the system. | |
| Main Flow | 1 | The actor navigates to the "Registrations" section. |
| | 2 | The system displays the option to select mess and program to view registrations. |
| | 3 | The actor can select any mess and program to view the registration. |
| | 4 | The system displays the total registrations in a filtered manner. |

| | | |
|------------------------------|------|---------------------------------------|
| Post conditions | NIL | |
| Alternate Flow | NIL | |
| Sub Flow | NIL | |
| Global Alternate Flow | GA 1 | The actor returns to the main screen. |

Use case #20

| | | |
|------------------------------|---|---|
| UC ID | UC#20 | |
| Use case Name | make/view_announcements | |
| Description | The " make/view_announcements " use case allows the Mess Warden to view or create announcements through the Fusion portal. | |
| Actor | Mess Warden | |
| Precondition | The actor should be logged in into the system. | |
| Main Flow | 1 | The actor navigates to the "Announcements" section. |
| | 2 | The system displays the option to view and make announcements. |
| | 3 | The actor can view all the announcements created till that point of time. |
| | 4 | The actor can also create a new announcement. |
| Post conditions | A new announcement will get added to the list of announcements if the warden chooses to create a new announcement | |
| Alternate Flow | NIL | |
| Sub Flow | NIL | |
| Global Alternate Flow | GA1 | The actor returns to the main screen. |

Use case #21

| | | |
|----------------------|---|--|
| UC ID | UC#21 | |
| Use case Name | search/generate_report | |
| Description | The " search/generate_report " use case allows the Mess Warden to view the final report of the month of mess inventory and other expenses. | |

| | | |
|------------------------------|--|---|
| Actor | Mess Warden | |
| Precondition | The actor should be logged in into the system. | |
| Main Flow | 1 | The actor navigates to the "Generate Report" section. |
| | 2 | The system displays the option to select a mess and a program to view registration, expenses. |
| | 3 | The actor can select any mess and program to view the report of expenses , student's registration etc . |
| | 4 | The system displays the report in a filtered manner. |
| Post conditions | NIL | |
| Alternate Flow | NIL | |
| Sub Flow | NIL | |
| Global Alternate Flow | GA1 | The actor returns to the main screen. |

3.3 Other Functional Requirements

- 3.3.1.** This module will make use of the communication module for sending notifications and alerts to various actors involved in the module suitably for registration, deregistration, any change in mess menu or any other important notice etc.
- 3.3.2.** The system will implement role-based dashboards for each actor, displaying relevant information based on the user's role and ensure customizability and responsiveness for various devices.
- 3.3.3.** The Super admin of Fusion should be able to assign roles for mess caretaker, mess warden, mess committee(students).
- 3.3.4.** Changes in the mess food price may be possible from time to time and the decision of the institute authorities.

3.4 Other Constraints

3.4.1 User Interfaces

- The user interface must align with the color scheme and dashboard design of FUSION IIIT.
- Inter-module navigation should be smooth and intuitive.
- All functionalities should be user-friendly, requiring no specific training for module usage.

3.4.2 Tech Stack Used

- **Flutter:** A UI software development toolkit developed by Google for creating natively compiled applications across various platforms.
- **Dart:** A client-optimized programming language used to write Flutter applications.
- **Git and GitHub:** Git is a version control system used to manage and track changes in the project codebase, while GitHub is a web-based hosting service that provides a platform for collaborative development using Git.
- **Visual Studio Code:** A lightweight and versatile code editor commonly used for developing Flutter applications, which offers extensive Flutter and Dart extensions for enhanced development capabilities.
- **(Optional)Android Studio:** An integrated development environment (IDE) provided by Google for developing Android applications. It also supports Flutter development with built-in Flutter and Dart plugins.
- **Flutter Packages:** Various open-source libraries and packages available from the Flutter community, which provide additional functionality and features to enhance the development process.

3.4.3 Business Rules (if any)

- If the remaining positive balance is less than the monthly bill, the system will automatically deregister the student for the next month.
- If a student's balance becomes negative, the system will automatically deregister the student.

4. Non-Functional Requirements

4.1 Performance:

The system should respond quickly to user interactions related to mess booking, menu viewing, attendance tracking, and feedback submission.

Response times for inventory updates, notifications, and student bill processing should be optimized.

4.2 Scalability:

The system should be capable of handling a large number of concurrent users during peak times, such as meal hours ,mess advanced fee payment time.,vacation time.

Performance should be tested and evaluated under increasing loads to ensure smooth operation during high-traffic periods.

4.3 Availability:

The central mess system should be available for use by students, Mess Caretaker, Mess Warden, and members of the Mess Committee 99.9% of the time.

Any scheduled maintenance or downtime should be communicated in advance to users.

4.4 Security:

Data related to student bills, student personal information, attendance records, menu updates, and feedback should be kept confidential and secure.

Role-based access control should be implemented to ensure that only authorized users can perform actions such as updating the menu, processing bills, or accessing sensitive data.

5. Module dependencies with other fusion modules

5.1 UI Level

- Integration into the Fusion portal, with a dedicated section for Mess Management.
- User-specific dashboards and interfaces tailored to each actor's role.

5.2 DB Level Dependencies

- Shared data related to students, mess menus, feedback, and financial transactions.
- Careful consideration of data ownership and access rights.

5.3 Module Level Dependencies

- Dependencies on the Notifications module for sending timely notifications.
- Interactions with the already existing Payment API for all Mess related transactions.