Software Requirements Specification

Version 1.0
<<Annotated Version>>

October 28, 2014

Mess Management System

Abhishek Mohta
Sanath Shetty
V.Madhumita Malvika
Yash Bardapurkar
Bhaargav S.
Vikrant Chauhan
Swapnil Mengade

Submitted in partial fulfillment
Of the requirements of
F213 Object Oriented Programming

Table of Contents

Introduction

1.1 Purpose

The purpose of this document is to present a detailed description of the Mess Management System. It will explain the purpose and features of the system, the interfaces of the system, what the system will do, the constraints under which it must operate and how the system will react to external stimuli. This document is intended for the developers of the system and will be proposed to the Mess Facility in Bits Pilani for its approval.

1.2. Scope of Project

This software system will be a ERP system for the manager of the mess. This system will be designed to maximize the manager's productivity by providing tools to assist in automating the process of managing the students, employees, and stock, which would otherwise have to be performed manually or in other inefficient mediums. By maximizing the editor's work efficiency and production the system will meet the manager's needs while remaining easy to understand and use. Another purpose would be to reduce the wastage of food by having a new leave system for students and also handling the night canteen in a more organised manner with billing to the student's SWD account.

More specifically, this system is designed to allow the manager to students and his employees when it comes to the menu, responsibilities of the employee and also handling the leave applications. The software will facilitate communication between mess manager, mess employees, and the students via E-Mail, SMS and the online portal. Students can request for the particular day's menu by SMS. The system also contains a relational database containing a list of students, employees, and mess stock details.

1.3. Glossary

Term	Definition
Student	Member of the academic institute
Menu	Food to be served
Database	Collection of all the information monitored by this system.
Field	A cell within a form.
Employee Database	The existing employee database.
Student Database	The existing student database to keep track of them.
Employee	Personnel working in the mess.
Software Requirements Specification	A document that completely describes all of the functions of proposed system and the constraints under which it must operate. For example, this document.

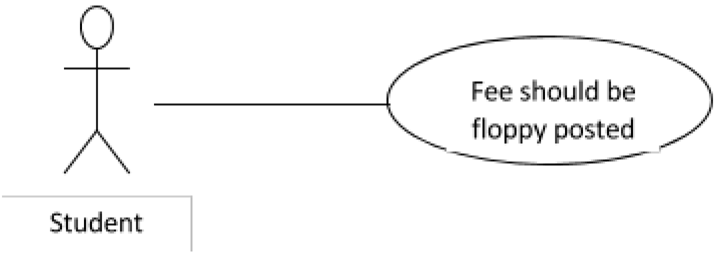
2.0 Overall Description

Functional Requirements Specification

2.1.1 Student Use Case

Use case: Pay fees by floppy posting

Diagram:



Brief Description

The student pays his/her fees on selecting the mess for the particular week

Initial Step-By-Step Description

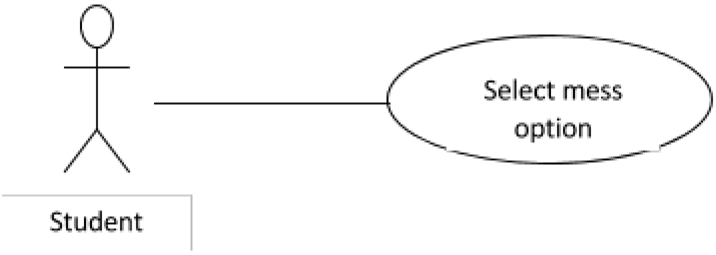
Before this use case can be initiated, the student should have already been registered and have had the minimum amount in their SWD account.

1. The student has a predetermined amount in his SWD account before the deadline.
2. The amount is deducted from his SWD account and transferred to the mess account.

2.1.2

Use case: Select mess option

Diagram:



Brief Description

The student fills his/her mess option on the SWD portal before a new week begins, to register for a mess.

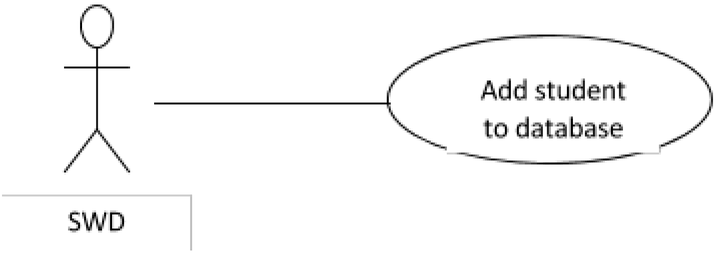
Initial Step-By-Step Description

1. The student logs into his SWD account on the SWD portal.
2. The student fills his/her mess preference.
3. The student is then registered for the selected mess for that week.

2.1.3 SWD Use Case

Use case: Add student to database

Diagram:



Brief Description

SWD adds information of a new student in the student database

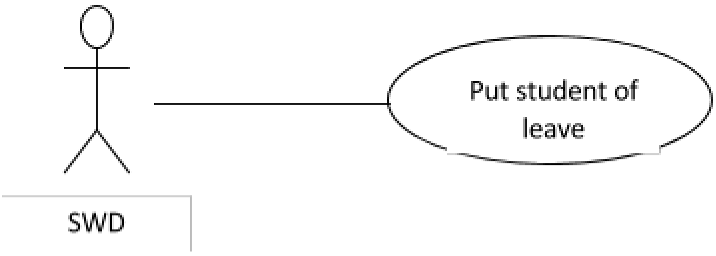
Initial Step-By-Step Description

Before this use case can be initiated, the student has paid his/her fees.

1. SWD deducts the fees from the bank account of the student.
2. SWD adds the student's personal information to the student database.

Use case: Put student on leave

Diagram:



Brief Description

SWD marks the student as 'on leave' in the student database.

Initial Step-By-Step Description

Before this use case can be initiated, the student should have applied for leave.

1. The student logs into his SWD account on the SWD portal.
2. The student applies for leave
3. SWD approves or rejects the students' application for leave.

2.3 User Characteristics

The Student is expected to be Internet literate and be able to use the SWD portal.

Also, the student is expected to have a bank account.

SWD is expected to be able to collect fees from the students' bank account, and also to maintain an accurate student database.

The student database is expected to have accurate student information.

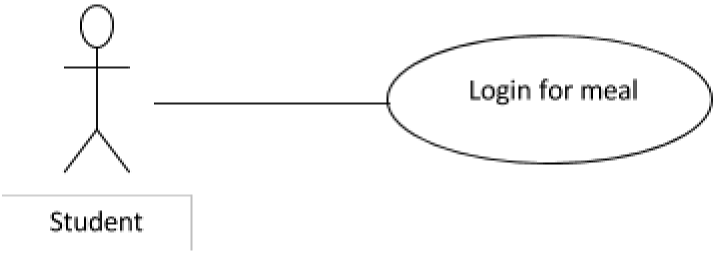
Mess

Functional Requirements Specification

2.2.1 Student Use Case

Use case: Login for meal

Diagram:



Brief Description

The student scans his/her ID card at the mess counter and eats the meal.

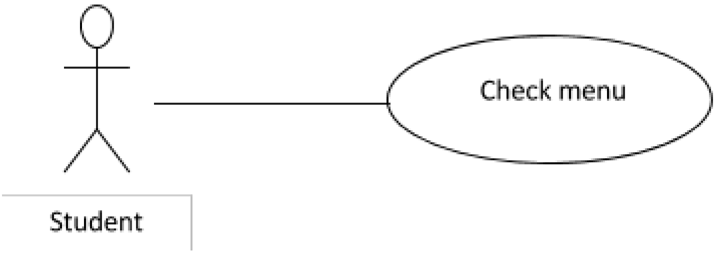
Initial Step-By-Step Description

Before this use case can be initiated, the student should have filled his/her mess option on SWD portal.

1. The student shows his/her ID card at the mess counter.
2. The employee scans the ID card.
3. The student database verifies that the student is registered for that mess, and has not eaten that meal.
4. If the student is not registered for that mess, the student is asked to pay for that meal.
5. The student eats his/her meal.

Use case: Check menu

Diagram:



Brief Description

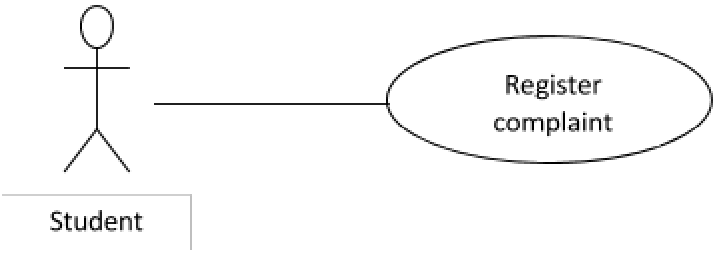
The Student check the daily menu.

Initial Step-By-Step Description

1. The student enquires about the daily menu.
2. (Optional) The student receives the menu via SMS.

Use case: Register complaint

Diagram:



Brief Description

The student registers his/her complaint in the complaint register.

Initial Step-By-Step Description

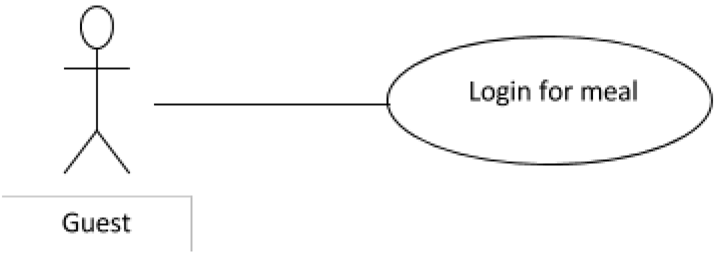
Before this use case can be initiated, the student should be registered for that mess.

1. The student enters a complaint in the complaints register
2. The Manager receives the complaints.

2.2.2 Guest Use Case

Use case: Login for meal

Diagram:



Brief Description

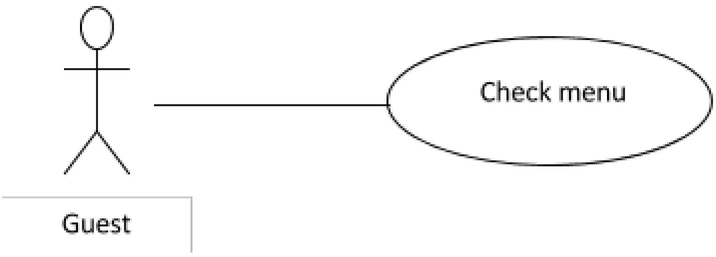
The Guest pays for the meal at the mess counter and eats the meal.

Initial Step-By-Step Description

1. The Guest pays money at the mess counter to the employee.
2. The Employee makes an entry of the guest in the Guest Database.
3. The Guest eats his/her meal.

Use case: Check menu

Diagram:



Brief Description

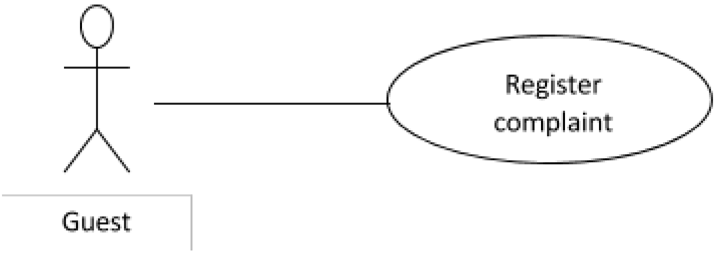
The Guest check the daily menu.

Initial Step-By-Step Description

1. The Guest enquires about the daily menu.
2. (Optional) The Guest receives the menu via SMS.

Use case: Register complaint

Diagram:



Brief Description

The Guest registers his/her complaint in the complaint register.

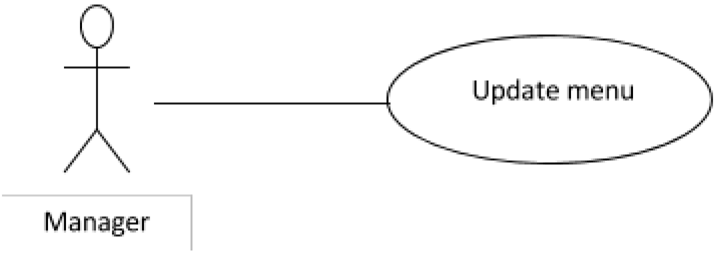
Initial Step-By-Step Description

1. The Guest enters a complaint in the complaints register
2. The Manager receives the complaints.

2.2.3 Manager Use Case

Use case: Update menu

Diagram:



Brief Description

The Manager makes changes in the weekly menu.

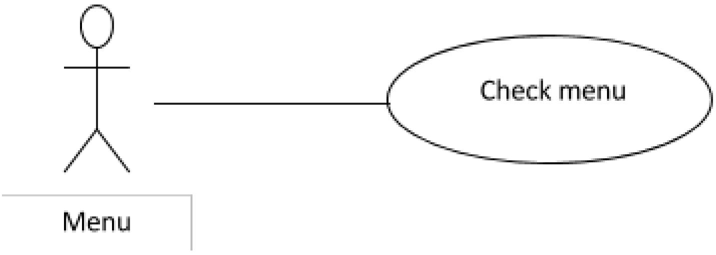
Initial Step-By-Step Description

1. Manager receives complaints and feedback.
2. The Manager changes the menu, as per student complaints or feedback.

2.2.3 Menu Use Case

Use case: Check menu

Diagram:



Brief Description

The Menu receives an enquiry about the day’s menu and sends information.

Initial Step-By-Step Description

1. The student or guest enquires about the menu.
2. The menu sends back the information by appropriate channel..

2.3 User Characteristics

The Student is expected to be Internet literate and be able to use the SWD portal. Also, the student is expected to have a bank account.

SWD is expected to be able to collect fees from the students’ bank account, and also to maintain an accurate student database.

The student database is expected to have accurate student information.

2.0. Overall Description

2.1 System environment

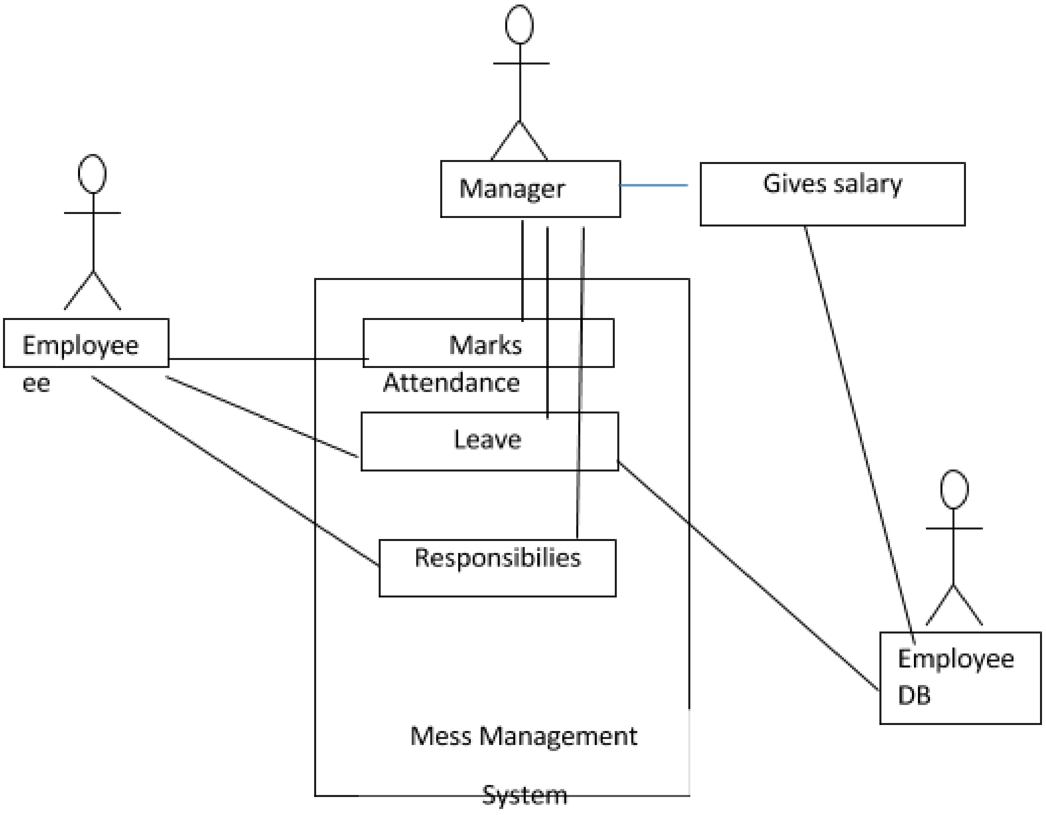


Figure 1 - System Environment

The Mess Management System has three active actors including one database system.

The Employee, Manager and the Employee database are the 3 actors.Employees are assigned various tasks and they coordinate and report to Manager. The Manager supervises the employees and is involved with everything to do with them.

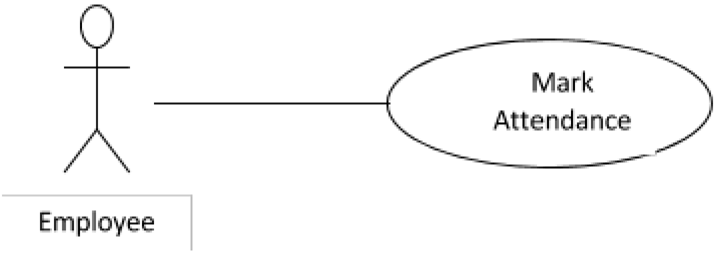
2.2 Functional Requirements Specification

This section outlines the use cases for each of the active readers separately. The reader, the author and the reviewer have only one use case apiece while the editor is main actor in this system.

2.2.1 Employee Use Case

Use case: Mark attendance

Diagram:



Brief Description

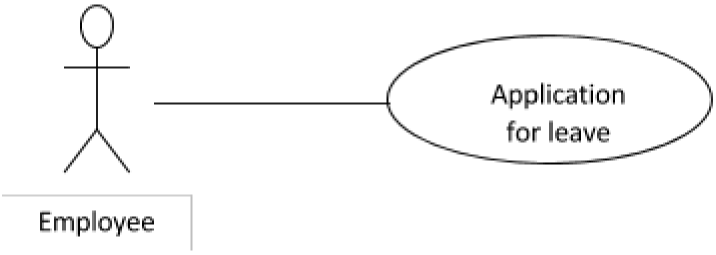
The Employee has to mark his attendance everyday while working in the mess.

Initial Step-By-Step Description

3. The Employee opens the employee portal to sign his attendance.
4. He uses his ID to login.
5. Once he is logged in, the employee enters additional details like in-time when he starts work and out-time when he leaves the job.

Use case: Apply for leave

Diagram:



Brief Description

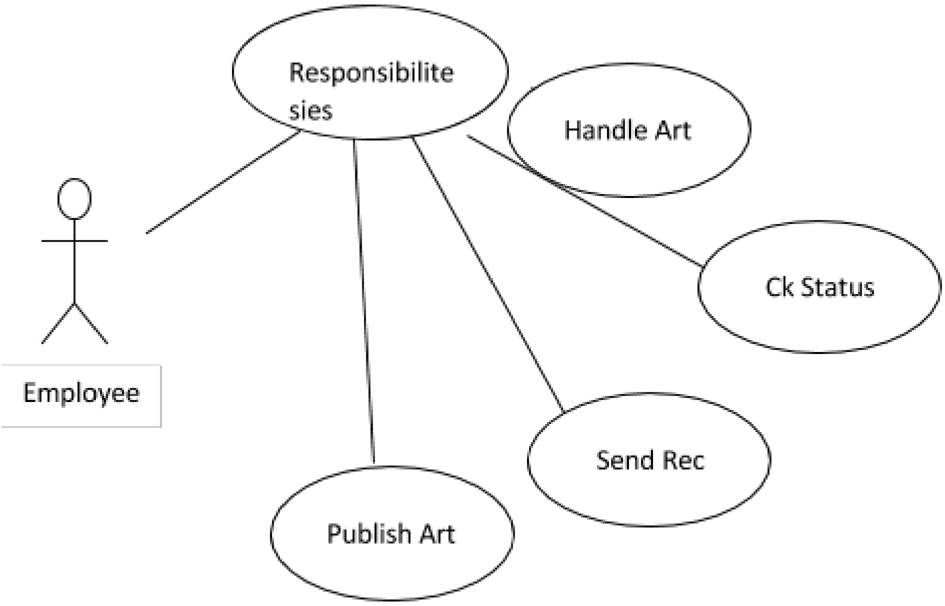
The Employee has to follow a procedure to apply for leave .

Initial Step-By-Step Description

1. The employee has to apply for leave two to three days prior to the day.
2. He opens his portal, and with his ID makes a request to get leave.
3. The permission has to come from the manager.
4. Once the leave is approved, the employee can take off from work for that particular day, when he has applied for leave.

Use case: Responsibilities

Diagram:



Brief Description

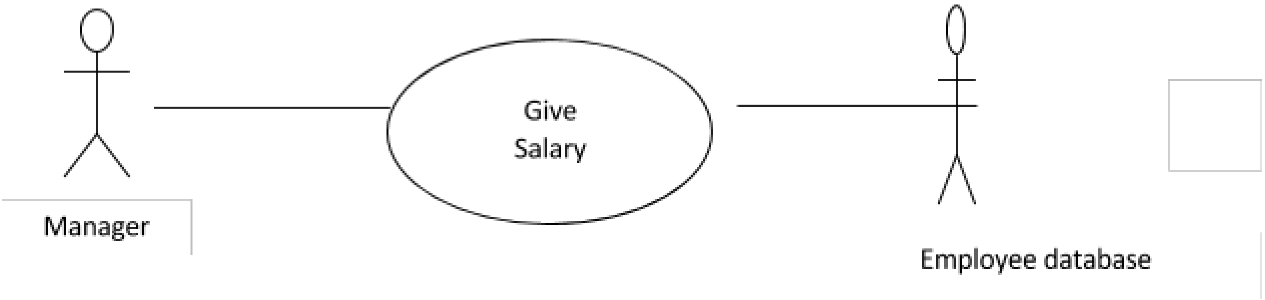
These are the responsibilities, given to the employees working in the mess. The responsibilities given to them keep varying.

Initial Step-By-Step Description

1. The employee is assigned tasks by the manager on a weekly basis.
2. The main tasks performed by the employees include cooking, cleaning, serving and the clerk job at the entrance.
3. Cooking involves cooking the 3 meals of the day in the kitchen, helping the manager prepare the menu and place order for items needed to the supplier.
4. Cleaning includes cleaning the mess at regular intervals.
5. The clerk is responsible to enter the students details on to the mess portal for every meal.
6. The student signs in at the entrance with his Id, the ID card is scanned by the clerk.

Use case: Giving Salary

Diagram:



Brief Description

The Manager credits all the employees with their salary .

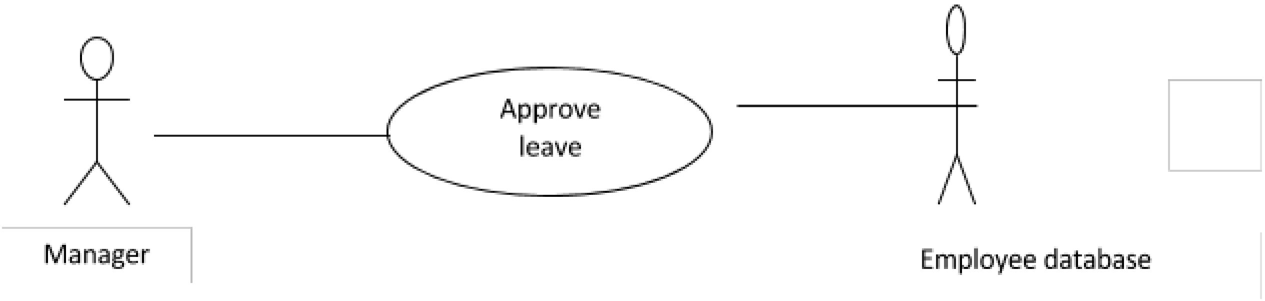
Initial Step-By-Step Description

1. The manager gives salary to all the employees based on the work done by them.

2. He keeps a note of the work they do, hours per day and the number of days the employee has taken off before concluding his salary.
3. He gives salary to the employees on a monthly basis.
4. All this information is fed into the employee database.

Use case: Approve Leave

Diagram:



Brief Description

The Manager gives permission to the employees to take off .

Initial Step-By-Step Description

1. The manager opens the portal and signs in.
2. He looks at the requests for holidays by various employees.
3. Based on the validity of the reasons and necessity/urgency, the manager approves or rejects the requests.
4. All this information is fed into the employee database.

3.2.1 Use Case: Paying his/her fees through floppy posting

Use Case Name	Paying his/her fees through floppy posting
XRef	
Trigger	The student selects the mess for that particular week.
Precondition	Before this use case can be initiated, the student should have already been registered and have had the minimum amount in their SWD account.
Basic Path	1. The student has a predetermined amount in his SWD account before the deadline. 2. The amount is deducted from his SWD account and transferred to the mess account.
Alternative Paths	None
Postcondition	The student has paid his/her fees through floppy posting.
Exception Paths	None
Other	None

3.2.2 Use Case: Select Mess Option

Use Case Name	Select Mess Option
XRef	
Trigger	When user accesses the mess option page in SWD portal.
Precondition	Student must register before the week begins.
Basic Path	4. The student logs into his SWD account on the SWD portal. 5. The student fills his/her mess preference. 6. The student is then registered for the selected mess for that week.
Alternative Paths	None

Postcondition	Student has selected his/her mess option.
Exception Paths	None
Other	None

3.2.3 Use Case: Add student to database.

Use Case Name	Add student to database.
XRef	
Trigger	When a new student is registered.
Precondition	The student should have paid his/her fees.
Basic Path	1. SWD deducts the fees from the bank account of the student. 2. SWD adds the student's personal information to the student database.
Alternative Paths	None
Postcondition	The student has been added to the database.
Exception Paths	None
Other	None

3.2.4 Use Case: Put student on leave.

Use Case Name	Put student on leave.
XRef	
Trigger	When student's leave is approved.
Precondition	None

Basic Path	1. The student logs into his SWD account on the SWD portal. 2. The student applies for leave 3. SWD approves or rejects the students' application for leave.
Alternative Paths	None
Postcondition	SWD marks the student as 'on leave' in the student database.
Exception Paths	None
Other	None

3.2.5 Use case: Login for a meal.

Use Case Name	Login for a meal.
XRef	
Trigger	When student scans his/her ID card at the mess counter or when a guest shows up at the counter.
Precondition	None
Basic Path	1. The student shows his/her ID card at the mess counter. 2. The employee scans the ID card. 3. The student database verifies that the student is registere for that mess, and has not eaten that meal. 4. If the student is not registered for that mess, the student i asked to pay for that meal. 5. In case of a guest, he/she is asked to pay for the meal directly. 6. Then the mess clerk enters the guest details in the guest database.
Alternative Paths	None
Postcondition	Student/Guest eats his/her meal.
Exception Paths	None
Other	None

3.2.6 Use Case: Check Menu

Use Case Name	Check Menu
XRef	
Trigger	When student/guest checks the daily menu.
Precondition	None
Basic Path	The student checks the daily menu option.
Alternative Paths	The student receives the menu via SMS or email.
Postcondition	The mess menu is seen by then student/guest.
Exception Paths	None
Other	None

3.2.7 Use Case: Register Complaints

Use Case Name	Register Complaints
XRef	
Trigger	Student/Guest access the complaint register.
Precondition	None
Basic Path	1. The Guest enters a complaint in the complaints register 2. The Manager receives the complaints.
Alternative Paths	None
Postcondition	The Student/Guest registers his/her complaint in the complaint register.
Exception Paths	None

Other	None

3.2.8 Use Case: Update menu

Use Case Name	Update menu
XRef	
Trigger	When manager chooses to update the menu.
Precondition	None
Basic Path	1. Manager receives complaints and feedback. 2. The Manager changes the menu, as per student complain or feedback.
Alternative Paths	None
Postcondition	The Manager makes changes in the weekly menu.
Exception Paths	None
Other	None

3.2.9 Use Case: Mark attendance

Use Case Name	Mark attendance
XRef	
Trigger	Employee enters or leaves work.
Precondition	None
Basic Path	1. The Employee opens the employee portal to sign his attendance. 2. He uses his ID to login.

	3. Once he logs in, the employee enters additional details like in-time when he starts work and out-time when he leaves the job.
Alternative Paths	None
Postcondition	The Employee marks his attendance everyday while working in the mess.
Exception Paths	None
Other	None

3.2.10 Use Case: Apply for leave

Use Case Name	Apply for leave
XRef	
Trigger	When employee wants to take some days off.
Precondition	None
Basic Path	1. The employee has to apply for leave two to three days prior to the day. 2. He opens his portal, and with his ID makes a request to g leave. 3. The permission has to come from the manager. 4. Once the leave is approved, the employee can take off from work for that particular day, when he has applied fo leave.
Alternative Paths	None
Postcondition	The employee gets his request approved.
Exception Paths	None
Other	None

3.2.11 Responsibilities

Use Case Name	Responsibilities
XRef	
Trigger	The manager assigns responsibilities to the employees,at the starting of every week.
Precondition	None
Basic Path	1. The employee is assigned tasks by the manager on a weekly basis. 2. The main tasks performed by the employees include cooking, cleaning, serving and the clerk job at the entran 3. Cooking involves cooking the 3 meals of the day in the kitchen, helping the manager prepare the menu and place order for items needed to the supplier. 4. Cleaning includes cleaning the mess at regular intervals. 5. The clerk is responsible to enter the students details on t the mess portal for every meal. 6. The student signs in at the entrance with his Id, the ID ca is scanned by the clerk.
Alternative Paths	None
Postcondition	The responsibilities are assigned to the employees by the manager.
Exception Paths	None
Other	None

3.2.12 Giving Salary

Use Case Name	Giving Salary
XRef	
Trigger	
Precondition	None
Basic Path	1. The manager gives salary to all the employees based on t work done by them.

	2. He keeps a note of the work they do, hours per day and the number of days the employee has taken off before concluding his salary. 3. He gives salary to the employees on a monthly basis. 4. All this information is fed into the employee database.
Alternative Paths	None
Postcondition	The Manager credits all the employees with their salary .
Exception Paths	None
Other	None

3.2.13 Approve Leave

Use Case Name	Approve Leave
XRef	
Trigger	When manager approves the leave applied by the employees.
Precondition	None
Basic Path	5. The manager opens the portal and signs in. 6. He looks at the requests for holidays by various employees. 7. Based on the validity of the reasons and necessity/urgenc the manager approves or rejects the requests. 8. All this information is fed into the employee database.
Alternative Paths	None
Postcondition	The Manager gives permission to the employees to take off .
Exception Paths	None
Other	None