SRS

MESS MANAGEMENT SYSTEM

NARENDER SINGH YADAV 2016UCP1474

LALIT KUMAR 2016UCP1572

BHARAT PAL 2016UCP1650

Table of Contents

1.0. 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 what kind of system interactions take place.

1.2. Scope of Project

This software system will be a mess management system which can be used by the students who eat in the mess and one for the mess administrator. The student software allows students to login and choose the monthly mess option, pay fees, view the daily mess menu , view mess balance,view food consumption statistics, inform the mess authorities when he/she won't be eating in the mess and submit feedback and complaint. The admin software will allow the mess administrator to make changes to the daily mess menu, update student mess balance and add student ,upload food consumption statistics and can read feedback and complaint. The system will utilize a relational database for

handling all the data such as menu items, prices, student IDs, consumption statistics, student feedback and complaint, etc.

1.3. Glossary

Term	Definition
Database	Collection of all the information monitored by this system.
Mess Admin	The administrator in charge of the mess - updates the databases, add students and update their mess balance etc.
Software Requirements Specification	A document that completely describes all of the functions of a proposed system and the constraints under which it must operate. For example, this document.
Student	A student avails the mess facilities.
Menu	A collection of all the food items which are available in the mess.

1.4. References

IEEE. *IEEE Std. 830-1998 IEEE Recommended Practice for Software Requirements Specifications.* IEEE Computer Society, 1998.

1.5. Overview of Document

The next chapter, the Overall Description section, of this document gives an overview of the functionality of the product. It describes the informal requirements and is used to establish a context for the technical requirements specification in the next chapter.

The third chapter, Requirements Specification section, of this document is written primarily for the developers and describes in technical terms the details of the functionality of the product.

Both sections of the document describe the same software product in its entirety, but are intended for different audiences and thus use different language.

2.0.Overall Description

2.1 System Environment

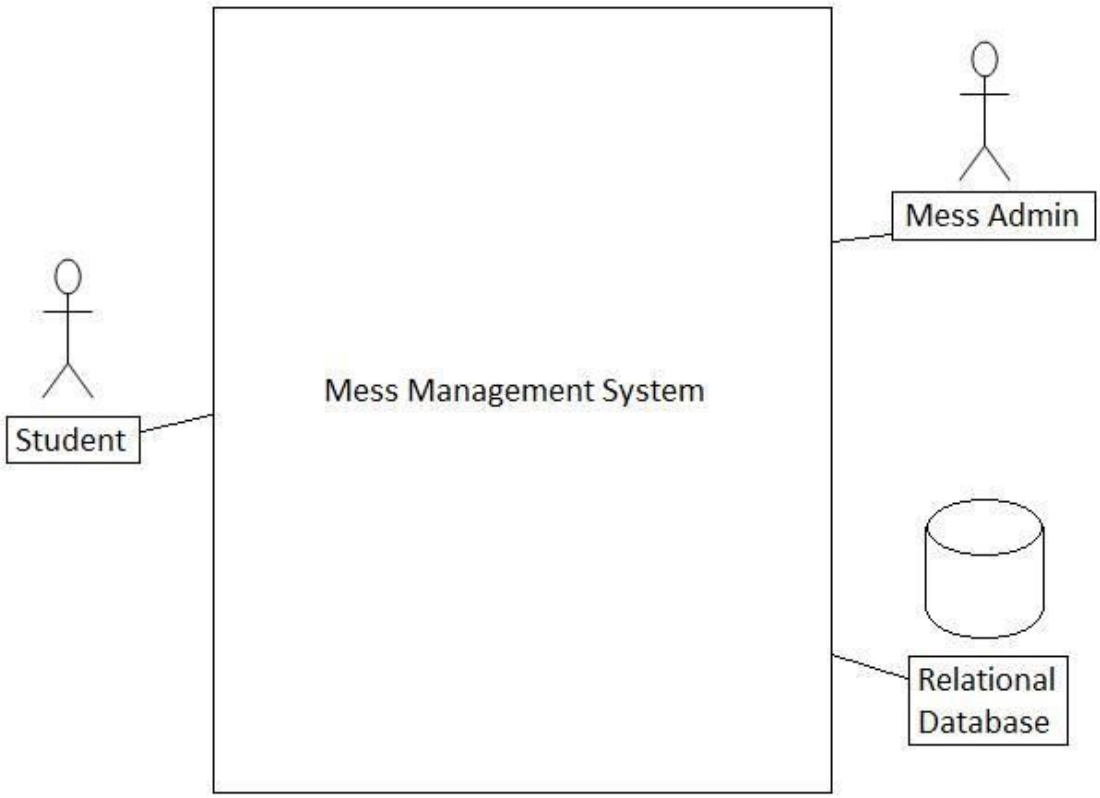


Figure 1 - System Environment

The Mess Management System has two active actors and one relational database.The Students use a portal to which they can login with their ID and password and avail all the facilities available. The Mess Admin has access to a different interface on his/her end, with a login and password. A relational database is used to store all the data such as feedback and complaint, consumption statistics, list of registered students, menus with items and prices, etc.

2.2 Functional Requirements Specification

This section outlines the use cases for each of the actors separately. The Students have a different set of use cases (which are related to availing the mess facilities) than the Mess Admin (which are related to managing the various aspects of the mess such as mess balance, add students and their fees, student mess option, menu updates, etc.).

2.2.1 Use Cases

A brief description of all the Student Use Cases:

1) **Login** – The student logs into the system so that he can carry on with other options like see mess balance and mess menu. The student has to enter his ID number and password which will be then compared with the database entries to validate his login credentials.

2) **Reset Password** – The student reset the current account password. To do this he has to click on forgot password then enter the student id then automatic mail with new password will be sent to student.

3) **View Menu** – The student gets to view the menu for the week. He can also choose to view a particular menu like Monday's lunch menu or the Night mess menu. The query from the student is sent to the database where the data is read and is sent back to the student to be displayed on the screen.
4) **Logout** – The student logs out of the system so that nobody else can modify his mess option or book orders from his account.
5) **Profile-** The student can view his/her profile updated during first time login in the software.
6) **History and Balance-** The student can see his coupon history as well as balance amount in his account.
7) **Generate coupon-** The student can generate daily coupons once a day.
8) **Feedback-** The student can give feedback to mess admin.
9) **Announcement-** The student will get announcement made by mess admin through email.

A brief description of the Mess Admin Use Cases:

1) **Login** – The administrator has to login to the system so that he can carry on with the other activities such as updating the mess menu, approving special orders, viewing and uploading statistics, viewing

NC orders etc. The data entered will be matched with the fields in the database to validate the admin's credentials.

2) **Change Profile** – The administrator can update profile of any student by searching it and making appropriate changes in it.
3) **Update Menu** – The administrator can update the menu for any meal for any day of the week or the might mess menu by selecting new food items to add to the menu or by removing unpopular food items. The changes after the updation are saved in the database so that the students can see the updated menu.
4) **View Feedback** – The mess admin can view the feedback submitted by the students. The database will be queried for feedback and complaints and the admin can check all these forms and take the required action on them. He can also update the feedback status as read or the complaint status as per its current state.
5) **Announcement-** The mess admin can make announcements which will sent to each student through email.
6) **Add Student/Balance-** The mess admin can add new students to login_details and can also add balance to any student.
7) **Logout –** The mess admin can log out safely while he/she is not using the software in order to avoid any misuse.

2.3 User Characteristics

The Students and Mess Admin are expected to have a basic working knowledge of a computer in order to use this software. The User Interface will be quite intuitive, so any advanced knowledge will not be necessary.

2.4 Non-Functional Requirements-The physical machine to be used in the mess needs to have internet access in order to connect to the database.. Students need internet access on their devices as well, since all the data will be stored on the database which the software will need to connect to.

3.0.Requirements Specification

3.1 External Interface Requirements

3.11 User Interfaces : The Interface will be in the form of software and android . It is designed to be functional and minimal in its styling. All options will be displayed in a menu based format. JAVA , PHP and FXML will be used to setup the page layout and add minimal styling to make the interface user friendly.

3.12 Hardware Interfaces : A web server will be required so that the students and the mess admin can connect to it to exchange information. The server have a database to store all the data entries. The Server will have to have a high speed 1 Gigabit ethernet connection to the college's local network.

3.13 Software Interfaces : The server will be hosted using Apache Tomcat Web server (Version 8.0.14). It will also have a MySQL relational database. The main backend processing will be done using PHP and Java Server Pages (JSP) including connecting to and accessing the database and processing requests.

3.14 Communications Interfaces : The main communication protocol will be Hypertext Transfer Protocol (HTTP). This will be used to transfer information back and forth from the client to the server. HTTP GET and POST will be used to send the information.

3.2 Functional Requirements

3.2.1 Student Login

Use Case Name	Login
Trigger	The Student clicks on the login button on the login page
Precondition	The Student has entered his login details on the login page
Basic Path	1. The student navigates to the login page 2. The student enters the username and password 3. The student clicks the login button 4. If the form data is empty, system shows a prompt for login details 5. If data is not empty, it is sent to the server

	6. The server compares the login data with the password stored in the database 7. If login credentials are verified, the student is logged in 8. If not, the student is prompted to enter the login details again
Postcondition	The Student is logged in and is taken to his account page
Exception	The Student may terminate the login at any time.
Paths	

3.2.2 Change Student Password

Use Case Name	Change Password
Trigger	The student clicks on the change password button on his account page
Precondition	The student has logged in and is on his account page
Basic Path	1. The student enters the previous password and the new desired password. 2. The data is sent to the server if it is not empty 3. The server validates the user's password and then updates his password in the database
Alternative Paths	The student may click on the forgot password link on the login page to have his password changed after having it emailed to him
Postcondition	The user's password is updated in the database
Exception	The attempt may be abandoned at any time.
Paths	

3.2.3 Choose Mess option

Use Case Name	Choose Mess option
Trigger	The user clicks the choose mess option button on his account page

Precondition	The user has logged in and the date is a valid date for selecting mess option (since it can only be done in a
	particular time of the month)
Basic Path	1. The student is given two options in the form of radio button to choose from 2. Once the user has selected an option, he clicks the submit button where the server receives this data 3. The server updates his mess choice in the database and also updates that he has chosen an option so that
	he cannot change it again
Alternative	If the student clicks submit without selecting an option,
Paths	the system will prompt for an option
Postcondition	The student's mess choice for the following month is
	updated in the database
Exception	The student may abandon the operation at any time.
Paths	
Other	Date and time validity needs to be checked. Also if no option is selected in that time period, the system will
	automatically assign a random choice

3.2.4 View Menu

Use Case Name	View Menu
Trigger	The student clicks the view menu button
Precondition	The student has clicked the view menu button on the
	home page of the mess management system
Basic Path	1. The student can see the mess schedule.

3.2.9 Logout

Use Case Name	Logout
Trigger	The Student clicks the logout button
Precondition	The Student has already logged in
Basic Path	1. The student is redirected to the login page

	2. The server ends the student's current session and	
	logs him out of the system	
Alternative	None.	
Paths		
Postcondition	The Student is logged out of the system	
Exception	If the student has logged in from multiple devices, there	
Paths	might be a conflict	
Other	None	

3.2.10 Mess admin login

Use Case Name	Login	
Trigger	The Admin clicks on the login button on the admin	
	login page	
Precondition	The admin has entered his login details on the login	
	page	
Basic Path	1. The admin navigates to the login page 2. The admin enters the username and password 3. The admin clicks the login button 4. If the form data is empty, system shows a prompt for login detail 5. If data is not empty, it is sent to the server 6. The server compares the login data with the password stored in the database 7. If login credentials are verified, the student is logged in 8. If not, the admin is prompted to enter the login	
		details again
Postcondition	The admin is logged in and is taken to his account page	
Exception	The admin may terminate the login at any time.	

3.2.12

Use Case Name	Update Menu
Trigger	The admin clicks the update menu button on his
	account page
Precondition	The admin has logged into his account
Basic Path	1. The admin gets a screen of the current menu 2. He can then create new items which will be added to the database 3. Using these food items, the admin can make changes to the men 4. The changes made will be sent to the server 5.The server reflects these changes by updating the
	menu database
Alternative	The admin can also choose to edit the Night mess menu
Paths	in the same way
Postcondition	The updated menu is saved in the database
Exception	The admin may abandon the operation at any time in
Paths	which case no changes will be made.
Other	Menu consists of the menu items and their details

3.2.16 Logout

Use Case Name	Logout
Trigger	The admin clicks the logout button
Precondition	The admin has already logged in
Basic Path	1. The admin is redirected to the login page 2. The server ends the admin's current session and
	logs him out of the system
Alternative	None.
Paths	
Postcondition	The admin is logged out of the system
Exception	None

Paths	
Other	None

3.3 Detailed Non-Functional Requirements

3.3.1 Logical Structure of the Data

Data Entries in the Database

LOGIN DETAILS

Data Item	Type	Description	
student_id	Text	Student id	
Password	Text	Password of student	
Balance	Int	Mess balance of student	

Student Data Entity

Data Item	**Type**	**Description**	
Name	Text	Name of Student	
ID	Text	id	
Hostel_no	Text	Hostel number	
Mobile_no	Text	student's mobile number	
Dept_name	Text	Department name	

Admin Data Entity

Data Item	Type	Description
User_name	Text	Username of the Admin
Password	Text	Password of Admin

Feedback Data Entity

Data Item	Type	Description
Content	Text	Feedback content
Date	Text	Date

Hostel

Data Item	Type	Description
hostel_no	Text	Hostel number
hostel_name	Text	Hostel name

Department

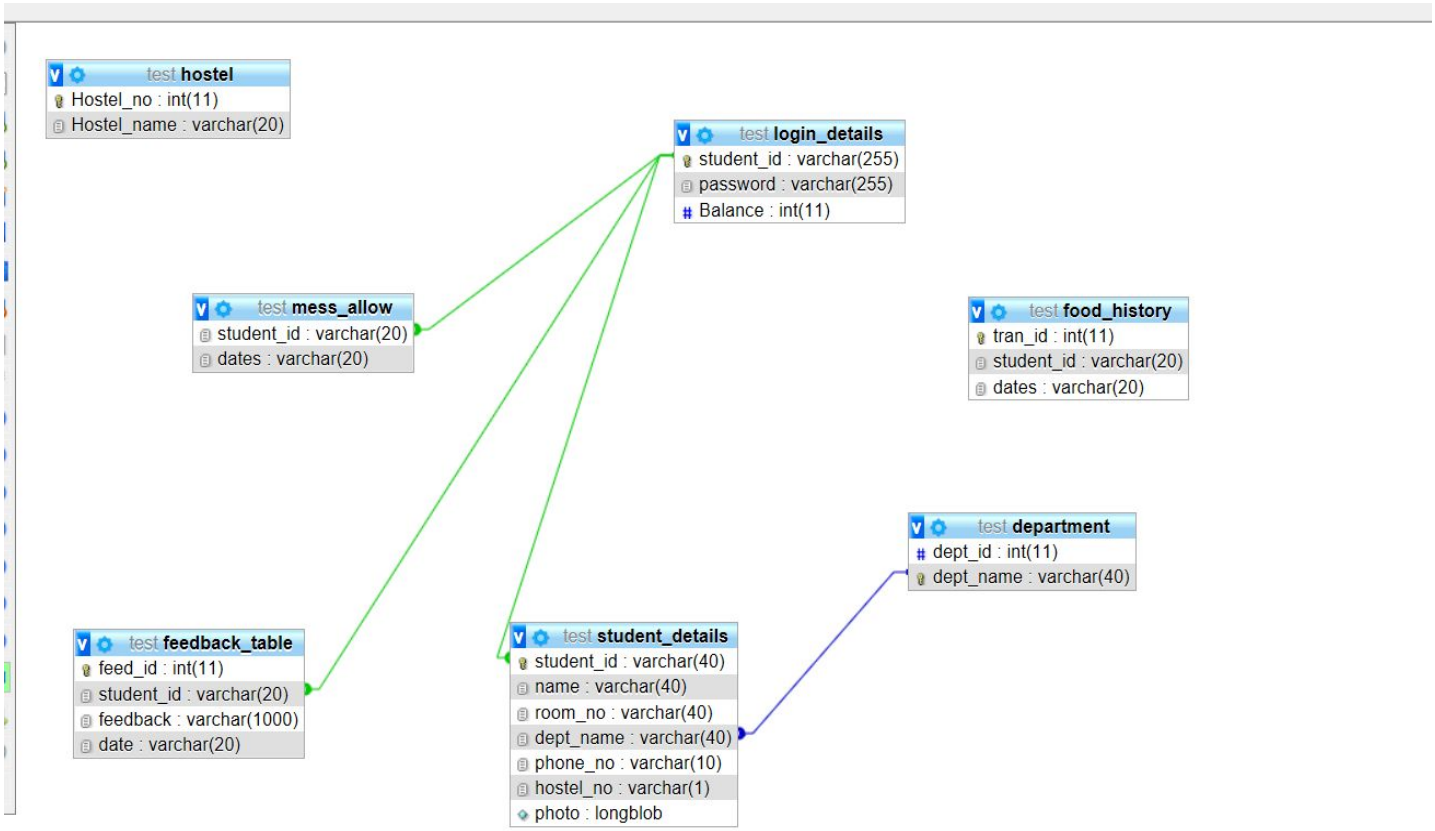
Data Item	Type	Description
dept_id	Text	Department id
dept_name	Text	Department name

Mess allow

Data Item	Type	Description
student_id	Text	Student id
Date	Text	Date

History

Data Item	Type	Description
student_id	Text	Student id
Date	Text	Date



3.3.2 **Security**

The server on which the Mess Management resides will have its own security to prevent unauthorized write/delete access. There is no restriction on read access. In case a password is forgotten, a new one will be emailed to that user's email ID. In addition, the passwords will be MD5 hashed for security.