Department of Software Engineering SWE 431: Thesis/Project

Project Documentation

Project On: Mess Bazar Management System

Supervised By

Mr. Khalid Been Badruzzaman Biplob

Lecturer

Department Of Software Engineering

Daffodil International University

Submitted By

Nazmul Haque

ID: 151-35-980

Department Of Software Engineering

A project submitted in partial fulfillment of the requirement for the degree of Bachelor of Science in Software Engineering Fall-2019

DECLARATION

We hereby declare that, this project has been done by us under the supervision of **Mr. Khalid Been Badruzzaman Biplob, Lecturer Department of Software Engineering, Daffodil International University**. We also declare that neither this project nor any part of this project has been submitted elsewhere for award of any degree or diploma.

Nazmul

Nazmul Haque

ID: 151-35-980

Department of Software Engineering

Daffodil International University

Certified by:

Mr. Khalid Been Badruzzaman Biplob Lecturer

Department of Software Engineering

Faculty of Science & Information Technology

Daffodil International University

Acknowledgement

Lastly, I am very glad that I have successfully able to complete my project on time. I appreciate all the helpers for helping out along the way of this development. I thank all of them cordially for their helpful attitude.

I am very pleased to our supervisor **Mr. Khalid Been Badruzzaman Biplob** Lecturer, Department of Software Engineering, Daffodil International University. Deep Knowledge & keen interest of our supervisor in the field of "**Mess Bazar Management System**" to carry out this project. His endless patience, scholarly guidance, continual encouragement, constant and energetic supervision, constructive criticism, valuable advice, reading many inferior draft and correcting them at all stage have made it possible to complete this project.

I wish to express our sincere thanks to **Pro. Dr. Touhid Bhuiyan**, Professor & Head of Software Engineering department for his kind help to finish our project and also to other faculty member and the SWE department of Daffodil International University. Finally, we must acknowledge with due respect the constant support and patients of our parents.

TABLE OF CONTENTS

CHAPTER 1

INTRODUCTION

The purpose of this application is to automatize the mess functionality and provide both the user and the admin a smart platform to interact with each other. To use this facility, the user has to register and with the account credentials he can login the account. User can use the functionalities of the mess such as accessing his account information, applying for leave and checking the menu. On the other hand the admin can log on and check the inventory details this application aims at providing ease to both the users and the admin. The mass guys communicate is other and width there connectivity. This application store mass people information to helping our government.

1.1 Objective

- This System truly helps who live in Mass
- Save record every days expense.
- Perfectly calculate cost for per meal
- Manage Bazar among mass guys.
- Save deposit record for mass guys
- Will reduce the computation pressure for the mass manager

1.2 Features

Admin

- Create mass member
- Manage daily meal bazar
- Manage Daily expense
- Monitoring every day meal of mass member
- Update mass member meal quantity
- Prepare monthly cost for all mass member

General user/ Mass member

- Daily meal quantity entry into the system
- View per month cost to pay the mass manager
- View meal rate per month
- View mass expense sheet

CHAPTER 2

SOFTWARE REQUIREMENT SPECIFICATION

2.1 Stakeholders

- Mass People
- Mass Owner
- Mass Member
- Bachler People
- Student who live in Mass

2.2 Functional Requirements

The functional requirement of the system is given blew:

2.2.1 Admin Registration:

Admin registration: Fr-01	**Admin registration**
Description	Admin registration page is only for those user who want to create mass community
Stakeholder	Admin

Table 2.2.1 Admin Registration

2.2.2 Member Registration:

Member registration: Fr-02	**Member registration**
Description	Member registration page is only for those user who want to join mass under mass manager.
Stakeholder	Member

Table 2.2.2 Member Registration

2.2.3 Admin/Member Login:

Admin/Member Login: Fr-03	**Admin/Member Login**
Description	Admin/Member Login page is only for those users who want to login in this system using register login credential

Stakeholder	Admin, Member

Table 2.2.3 Admin/Member Login

2.2.4 Expense:

Expense: Fr-03	**Expense**
Description	Admin can manage expense
Stakeholder	Admin

Table 2.2.4 Expense

2.2.5 Meal:

Meal: Fr-04	**Meal**
Description	Admin and member can update regular meal quantity.
Stakeholder	Admin, Member

Table 2.2.5 Meal

2.3 Data Requirements:

For running any system there are needed different types of data. Like that my system needs the following data.

- Meal Quantity
- Expense

CHAPTER 3

SYSTEM ANALYSIS

3.1 Class Diagram

Mess Bazar Management System application develop to following this diagram. All class define with properties, functions and construction to follow proper naming convention.

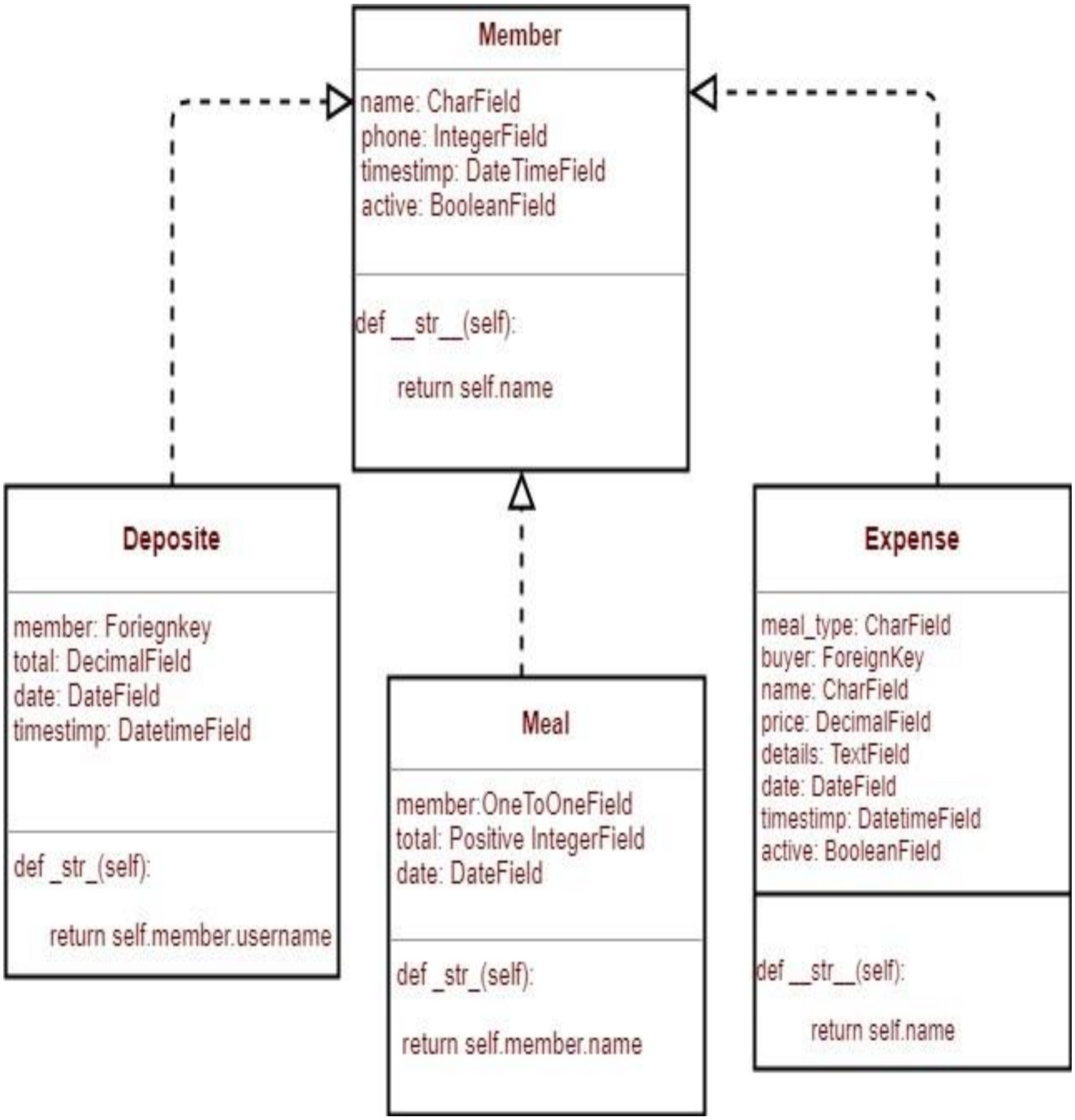


Fig 3.1.1

3.2 Use Case Diagram

This diagram indicate the Admin and member functionality and set of actions and services.

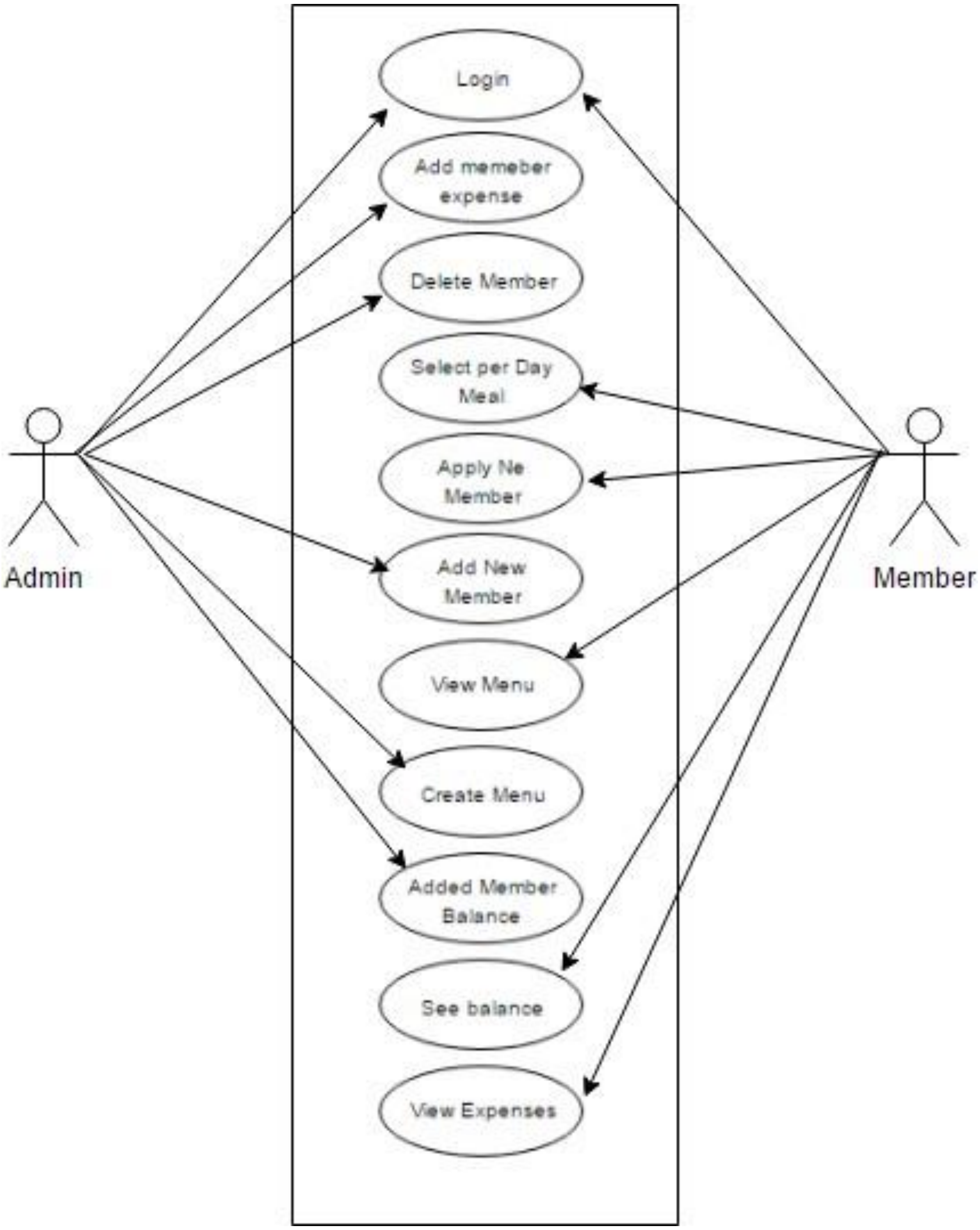


Fig 3.2.1 Use case Diagram

3.3 Use Case Description

3.3.1 Login

Use Case Name	**Login**
Actors	Admin, Member
Preconditions	Every actors have login credential.
Post Condition	Access Main features according to user role
Primary Scenario	Hit on application domain and provide valid data.

Table 3.3.1 Login

3.3.2 Add new Member

Use Case Name	**Add New Member**
Actors	Admin
Preconditions	Must be authorize and navigate the Add New Member page and input valid data.
Post Condition	New member successfully insert
Primary Scenario	Navigate the Add New Member page and fill up the require data and click save button.

Table 3.3.2 Add new Member

3.3.3 Add Member Balance/Deposit

Use Case Name	**Add New Member**
Actors	Admin
Preconditions	Must be authorize and navigate the Deposit page and input valid data.
Post Condition	Member deposit successfully insert
Primary Scenario	Navigate the Deposit page and fill up the require data and click save button.

Table 3.3.3 Add Member Balance/Deposit

3.3.4 Expense

Use Case Name	**Expense**
Actors	Admin
Preconditions	Must be authorize and navigate the Expense page and input valid data.
Post Condition	Successfully expense insert
Primary Scenario	Navigate the Expense page and fill up the require data and click save button.

Table 3.3.4 Expense

3.3.5 Meal

Use Case Name	**Meal**
Actors	Admin, member
Preconditions	Must be authorize and navigate the Meal Page and input valid data.
Post Condition	Successfully Meal insert
Primary Scenario	Navigate the Meal page and fill up the require data and click save button.

Table 3.3.5 Meal

3.4 ER Diagram

Mess Bazar Management System Application database design following this ER diagram.

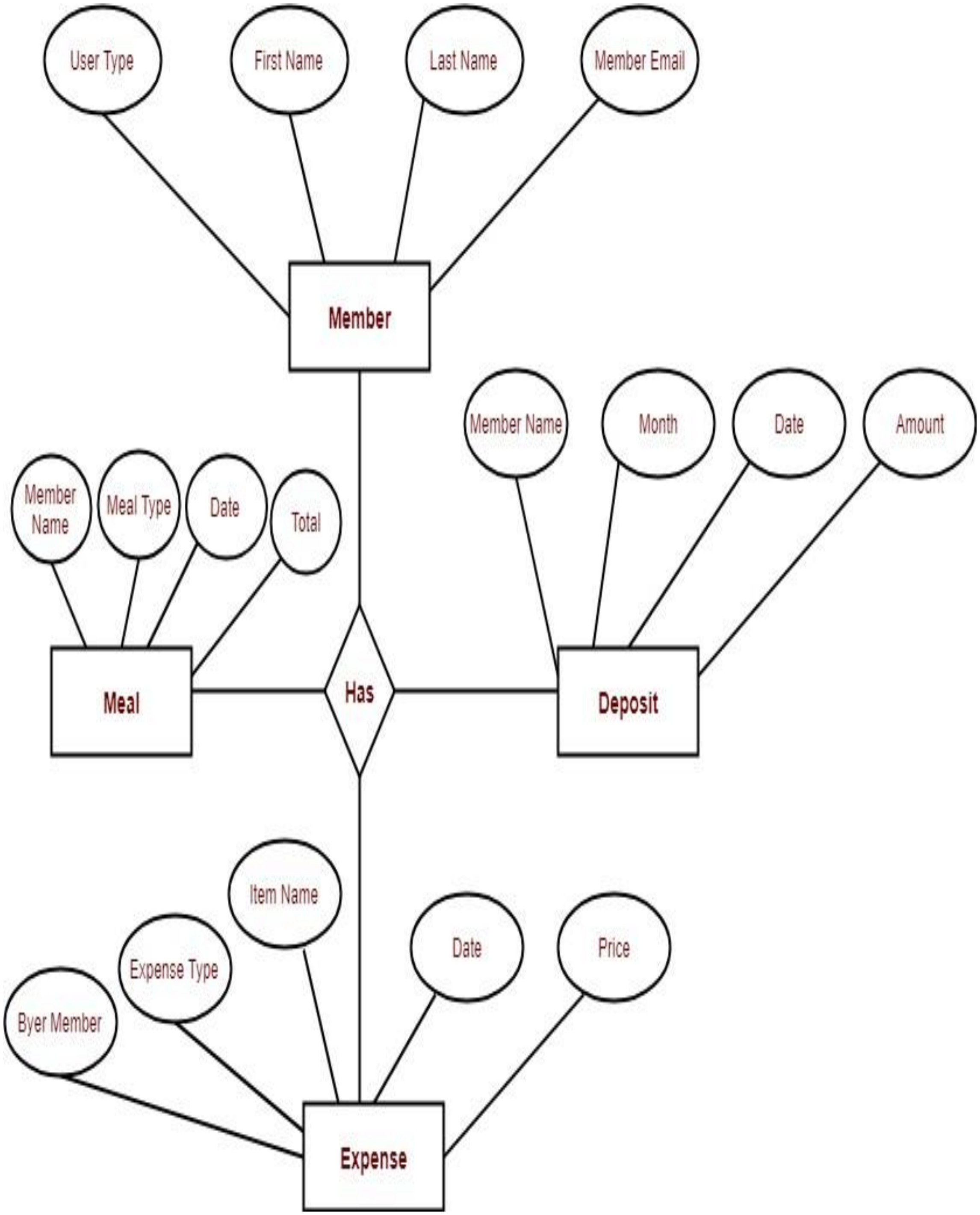


Fig 3.4.1 ER Diagram

3.5 Activity Diagram

Admin Login

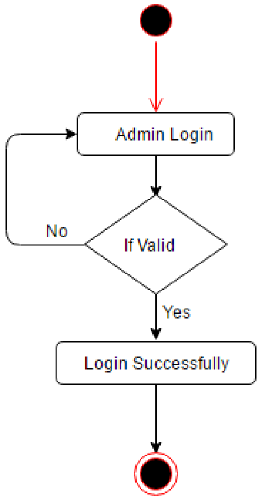


Fig 3.5.1 Admin Login

3.5.2 Admin Add New Member

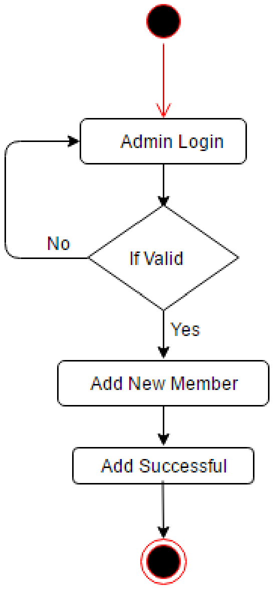


Fig 3.5.2 Admin Add New Member

3.5.3 Admin Add Member Balance

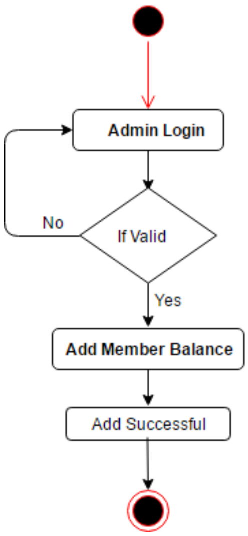


Fig 3.5.3 Admin Add Member Balance

3.5.4 Admin Add Member

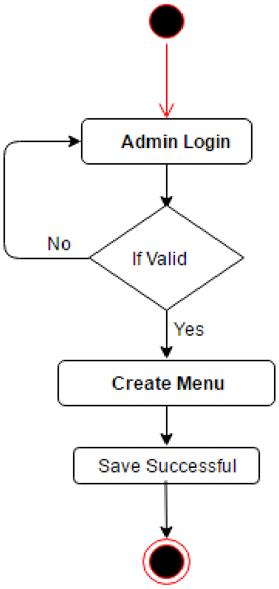


Fig 3.5.4 Admin Add Member

3.5.5 Member Login

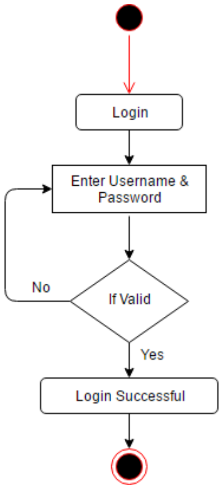


Fig 3.5.5 Member Login

3.5.6 See Balance

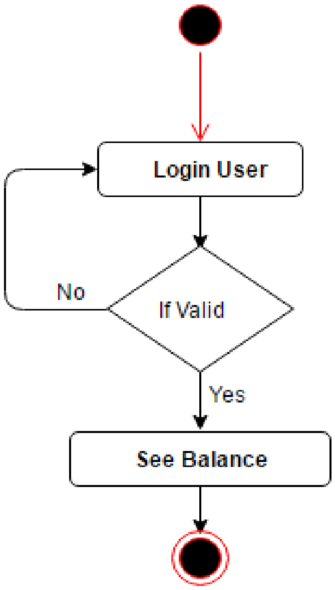


Fig 3.5.6 See Balance

3.5.7 See Expense

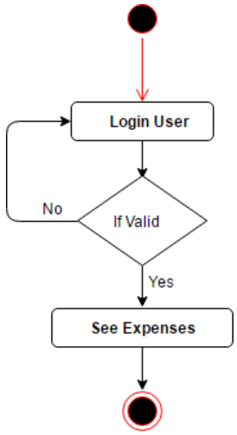


Fig 3.5.7 See Expense

3.5.8 Select Meal

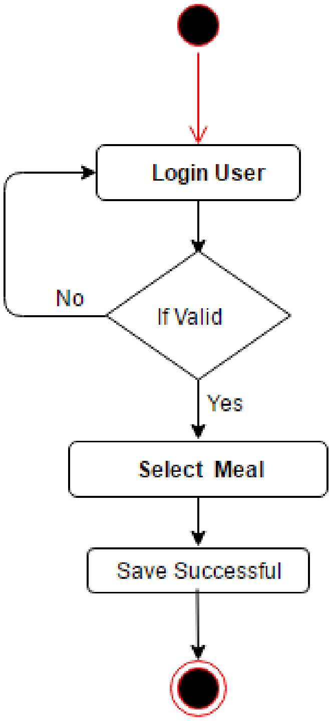


Fig 3.5.8 Select Meal

3.5.9 View Menu

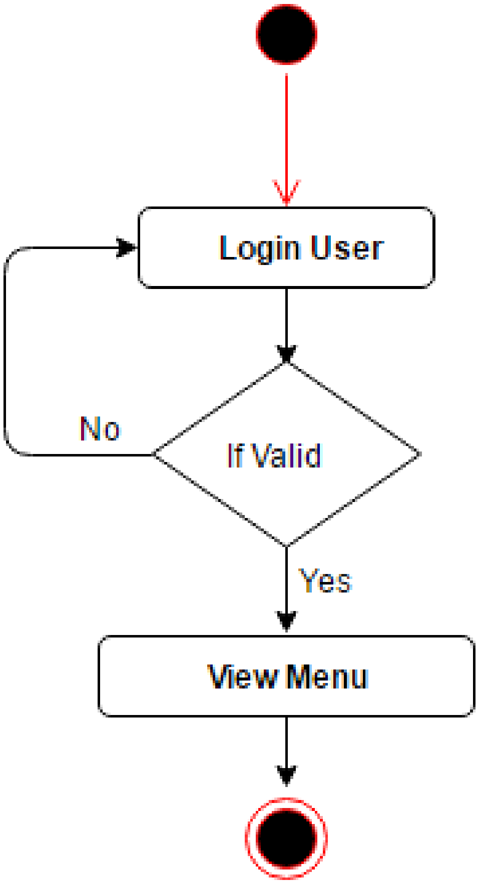


Fig 3.5.9 View Menu

3.6 Sequence Diagram

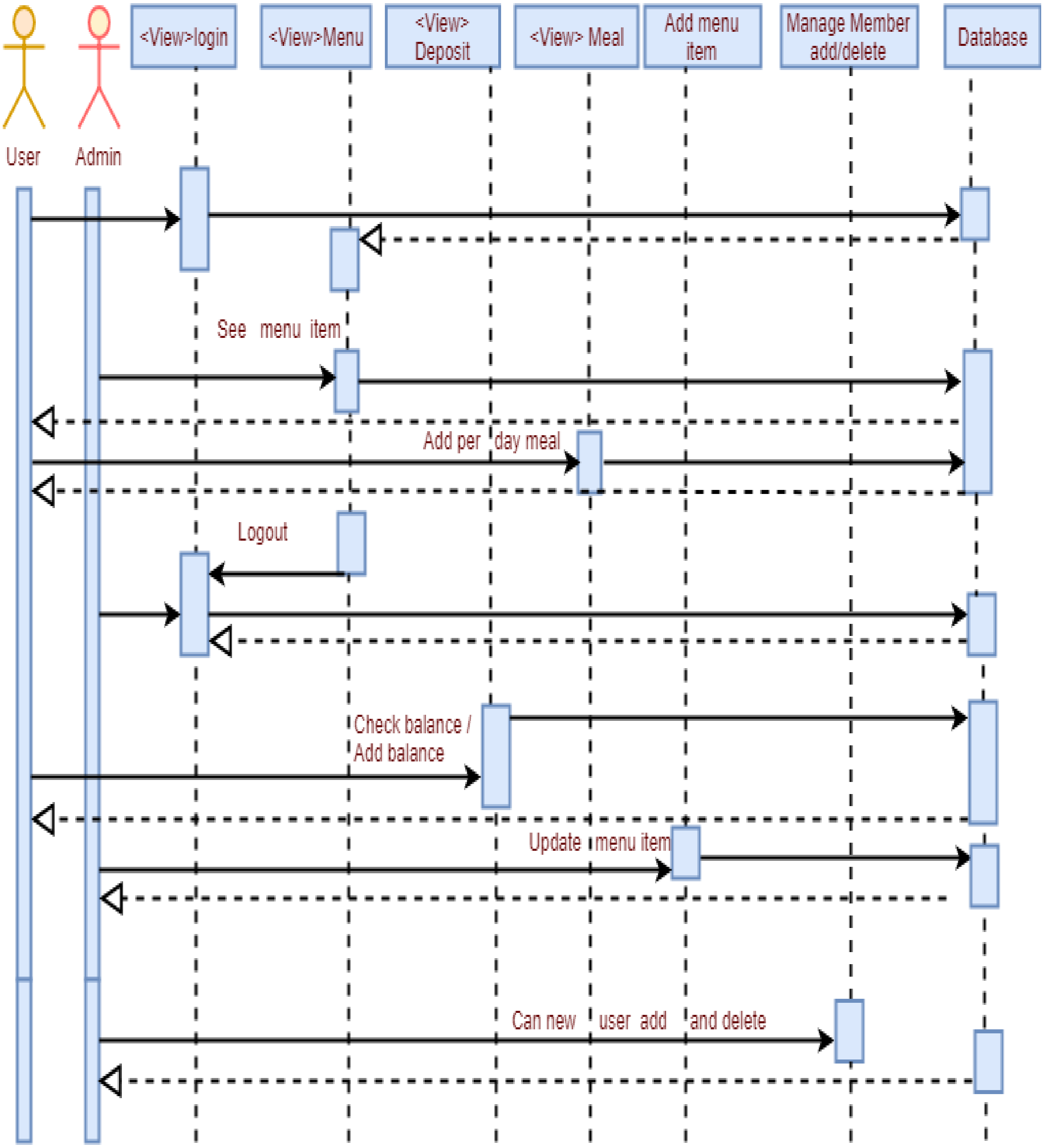


Fig 3.6.1

CHAPTER 4

SYSTEM DESIGN SPECIFICATION

4.1 Gang Chart

Task/Date	Start Date	End Date	Status	Mar	Apr	Aug	Sep	Oct	Nov
Proposal	07-03-2019	23-03-2019	Compete	🟩					
Requirements	24-03-2019	02-04-2019	Compete		🟧				
Design	03-08-2019	10-09-2019	Compete			🟫	🟫		
Implementation	18-09-2019	06-10-2019	Compete				🟥	🟥	
Testing	11-10-2019	15-11-2019	Compete					🟫	
Documentation	16-11-2019	30-11-2019	Compete						🟦

Table 4.1.1 Gang Chart

4.2 Plan

Milestone, it is a time frame of project. That will define the project task. Project milestones are as follows:

Task no.	Task Name	Start Date	End Date	Duration
01	Planning	01-01-19	12-01-19	12 days
02	Requirements Gathering & Analysis	13-01-19	24-01-19	12 days
03	System Design	25-09-19	06-10-19	12 days
04	Development	07-10-19	01-11-19	26 days
05	Testing	02-11-19	09-11-19	08 days
06	Implementation	10-11-19	15-11-19	06 days
07	Release	16-11-19	21-11-19	06 days
Total				**82 days**

Table 4.2.1 Plan

4.3 System Implementation Details

Client Side

- Html
- CSS
- Bootstrap

- JavaScript

Server Side

- Programming Language - Python
- Framework – Django

Database

- SqlLite3

IDE

- Pycharm, Atom)
- Local/Web Server(Powershell, cmd),

Source Control

- Git hub

4.4 Architecture used MTC (Model, Template and Controller) Description of Folder Structure:

Model part folder Structure:

Login, Registration

App name: account

Folder: Templete/registration/forms.py

All post model

App name: messApp

Folder: Templates/model.py

Template part folder Structure:

Folder name: account/template

Login.html, Registration.html

messApp>template/messapp/balance.html/bazar.html/dashboard.html/dinner.html/expense_list.html/lunch.html/meal_list.html/member.html/members_details.html

Controller part folder Structure:

Login, registration controller

App name: account

Folder: template/views.py/url.py

App name: messApp

Folder: template/messapp/views.py/url.py

CHAPTER 5

SYSTEM TESTING

5.1 Testing Features:

For verifying the working process of the features of any system, Testing is very necessary. We can detect the proper performance of the features by testing properly.

5.2 Testing strategies:

- Quality test
- Measure test

5.3 Test Approach:

In test approach there are two types of techniques:

- **Proactive** - An approach in which the test design process is initiated as early as possible in order to find and fix the defects before the build is created.
- **Reactive** - An approach in which the testing is not started until after design and coding are completed.

5.4 Pass/Fail criteria

- When inserted data is ok then it is passed
- When inserted data is not ok then it is failed.

5.5 Testing Environment

- Testing IDE: Microsoft Visual Studio 2017
- Browser: Google chrome
- Server: Microsoft SQL Server 2017
- Operating System: Windows 10
- Possessor: Core I 3

5.6 Test Case

Test case is very important for justifying the activity of any system. Here is given my system's test cases.

5.6.1 Test case for Admin registration

Test case ID: 01	**Module name: Admin**
Sub Module: Admin registration	Test design by: NazmulHaque
Test priority(low/medium/high): high	Test design date: 02-11-2019
Text title: registration with valid information	Test executed by: NazmulHaque
Description: test the system's on registration page	Text executed date:02-11-2019

Table 5.6.1 Test case for Admin registration

Preconditions: The user navigate to registration page and input the required filled. And click on the Register button.

Step	Test step	Test data	Expected result	Actual result	Pass/Fail
1	Navigate to admin registration page	Click on Registration button.	Data inserted successfully	Data inserted	pass
2	Name	Nazmul			
3	Email	nazmul1661@gmail.com			
4	Password	N15135980			

Table 5.6.2

5.6.3 Test case for Member registration

Test case ID: 02	**Module name: Admin**
Sub Module: Admin registration	Test design by: Nazmul Haque
Test priority(low/medium/high): high	Test design date: 02-11-2019
Text title: registration with valid information	Test executed by: Nazmul Haque
Description: test the system's on registration page.	Text executed date:02-11-2019

Table 5.6.3 Test case for Member registration

Preconditions: The user navigate to registration page and input the required filled. And click on the Register button.

Step	Test step	Test data	Expected result	Actual result	Pass/Fail
1	Navigate to Member registration page	Click on Registration button.	Data inserted successfully	Data inserted	pass
2	Name	Rizu			
3	Username	RiZuoo7			
4	Password	R@123456			

Table 5.6.4 Test case for Member registration

5.6.5 Test case for Meal

Test case ID: 03	**Module name: Admin**
Sub Module: Admin registration	Test design by: Nazmul Haque
Test priority(low/medium/high): high	Test design date: 02-11-2018
Text title: registration with valid information	Test executed by: Nazmul Haque
Description: test the system's on registration page	Text executed date:02-11-2019

Table 5.6.5 Test for Meal

Preconditions: The user navigate to Meal Add page and input the required filled. And click on the Register button.

Step	Test step	Test data	Expected result	Actual result	Pass/Fail
1	Navigate to Meal Add page	Click on Add button.	Data inserted successfully	Data inserted	pass
2	Breakfast	6			

	Quantity				
3	Lunch Quantity	6			
4	Dinner Quantity	8			
	Date	28-11-2019			

Table 5.6.6

5.6.7 Test case for Expense

Test case ID: 04	Module name: Adminr
Sub Module: Admin registration	Test design by: Nazmul Haque
Test priority(low/medium/high): high	Test design date: 02-11-2018
Text title: registration with valid information	Test executed by: Nazmul Haque
Description: test the system's on registration page	Text executed date:02-11-2019

Table 5.6.7 Test case for Expense

Preconditions: The user navigate to Expense page and input the required filled. And click on the Register button.

Step	Test step	Test data	Expected result	Actual result	Pass/Fail
1	Navigate to Expense page	Click on Add button.	Data inserted successfully	Data inserted	pass
2	Item	Banana, Bread			
3	Amount	200 taka			
4	Date	28-11-2019			

Table 5.6.8

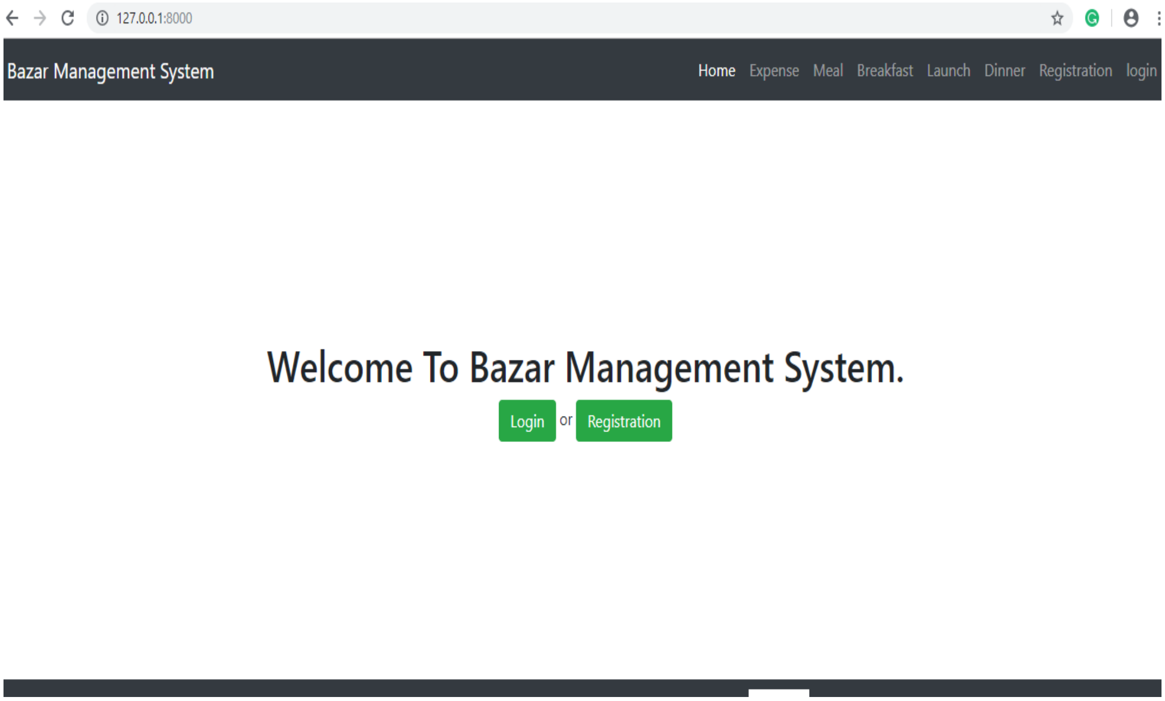
CHAPTER 6

USER MANUAL

User Interface Design Screenshot

6.1 Home Page

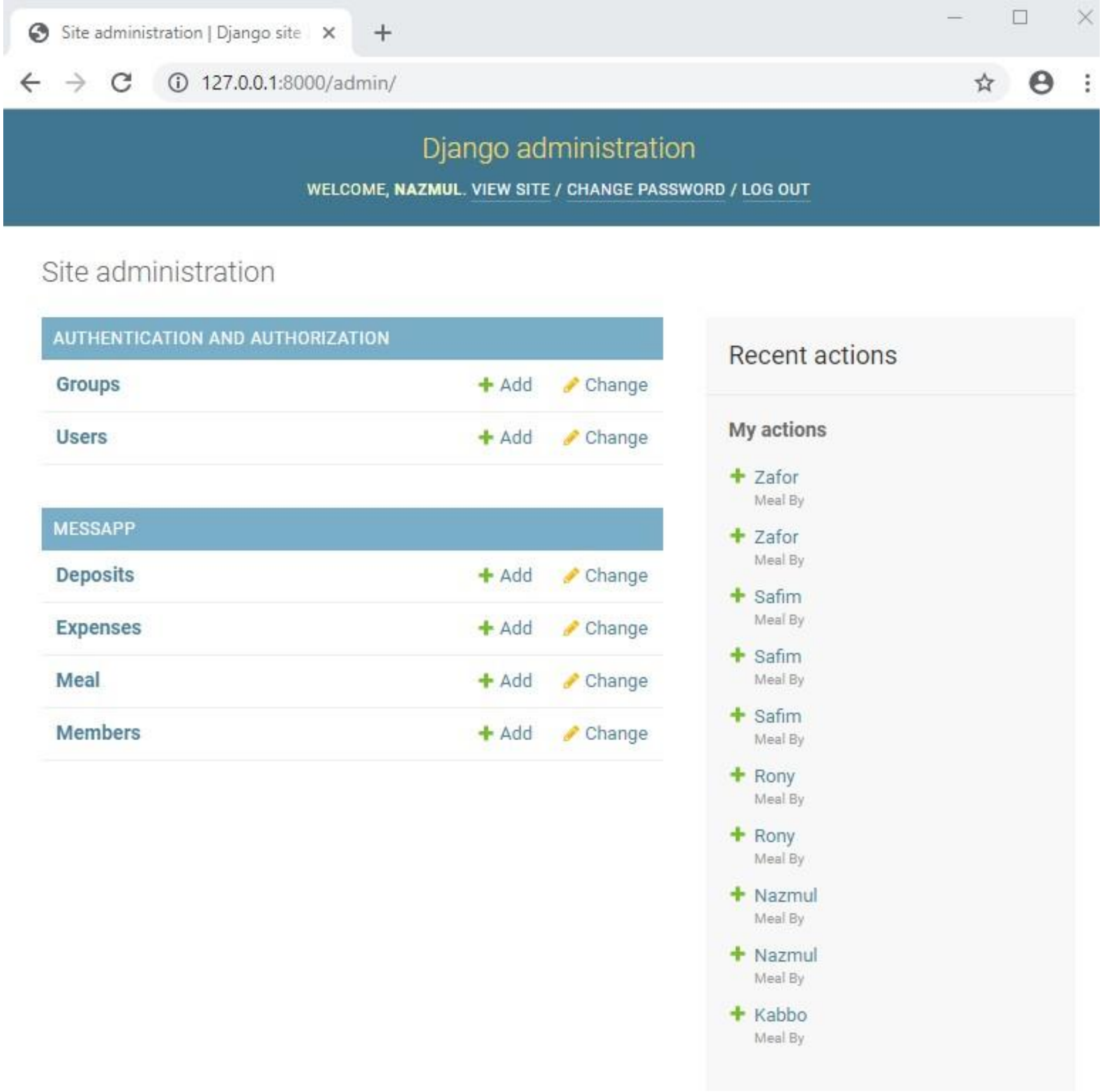
When run the application then visualize this UI. Here admin navigate to login and registration page.



Welcome To Bazar Management System.

6.2 Admin Dashboard

Admin access this UI to very authentication and authorization. Here admin view the all mass daily meal.



Site administration | Django site

127.0.0.1:8000/admin/

Django administration

WELCOME, NAZMUL. VIEW SITE / CHANGE PASSWORD / LOG OUT

Site administration

AUTHENTICATION AND AUTHORIZATION		
Groups	+ Add	Change
Users	+ Add	Change

MESSAPP		
Deposits	+ Add	Change
Expenses	+ Add	Change
Meal	+ Add	Change
Members	+ Add	Change

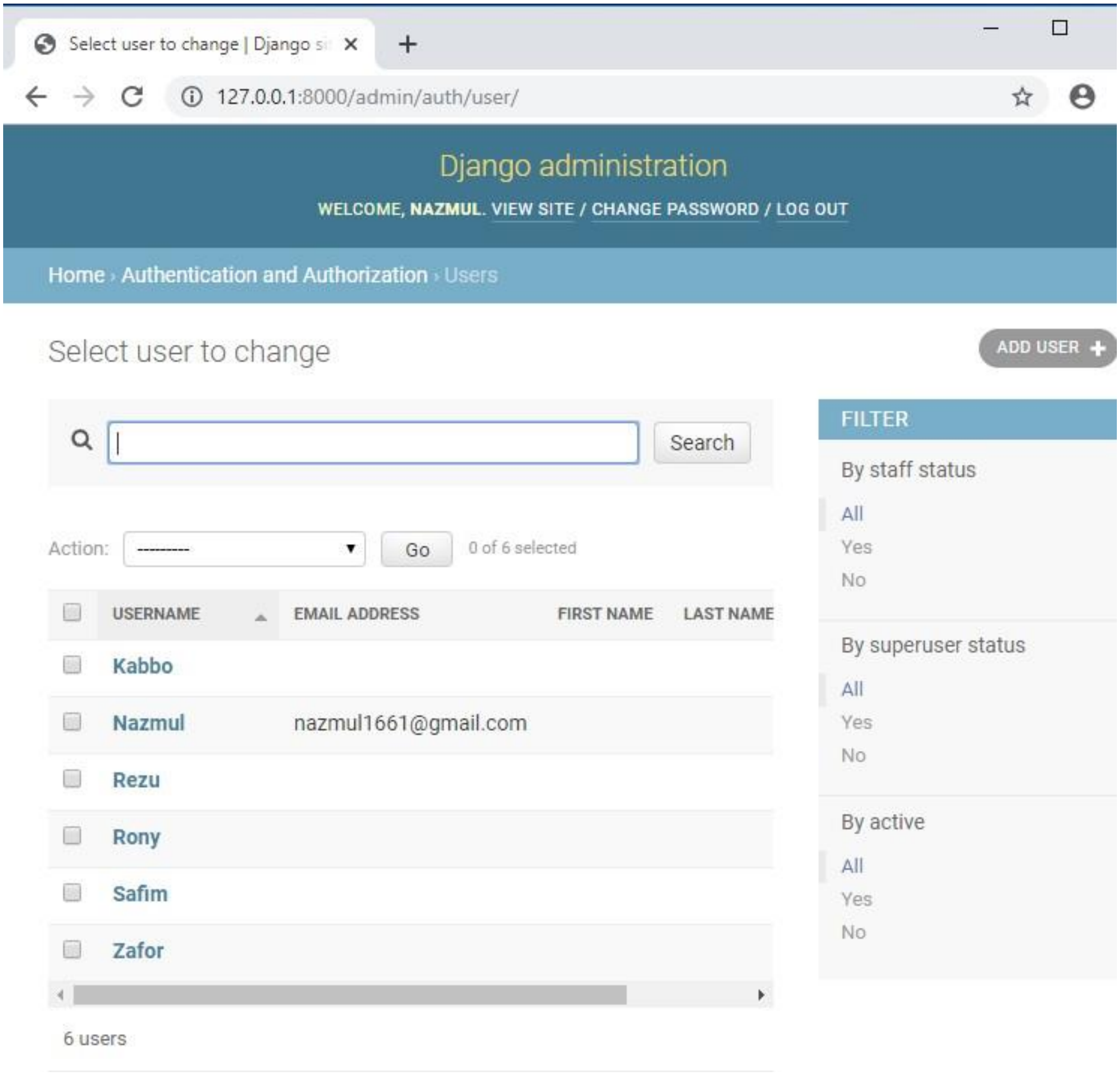
Recent actions

My actions

+ Zafor
Meal By
+ Zafor
Meal By
+ Safim
Meal By
+ Safim
Meal By
+ Safim
Meal By
+ Rony
Meal By
+ Rony
Meal By
+ Nazmul
Meal By
+ Nazmul
Meal By
+ Kabbo
Meal By

6.3 View Member List

Admin see the all mass member at a glance from this UI. Here admin manage the mass member active status.



Select user to change | Django s
127.0.0.1:8000/admin/auth/user/
Django administration
WELCOME, NAZMUL. VIEW SITE / CHANGE PASSWORD / LOG OUT
Home › Authentication and Authorization › Users
ADD USER
Search
Action:

Go
0 of 6 selected
FILTER
By staff status
All
Yes
No
By superuser status
All
Yes
No
By active
All
Yes
No

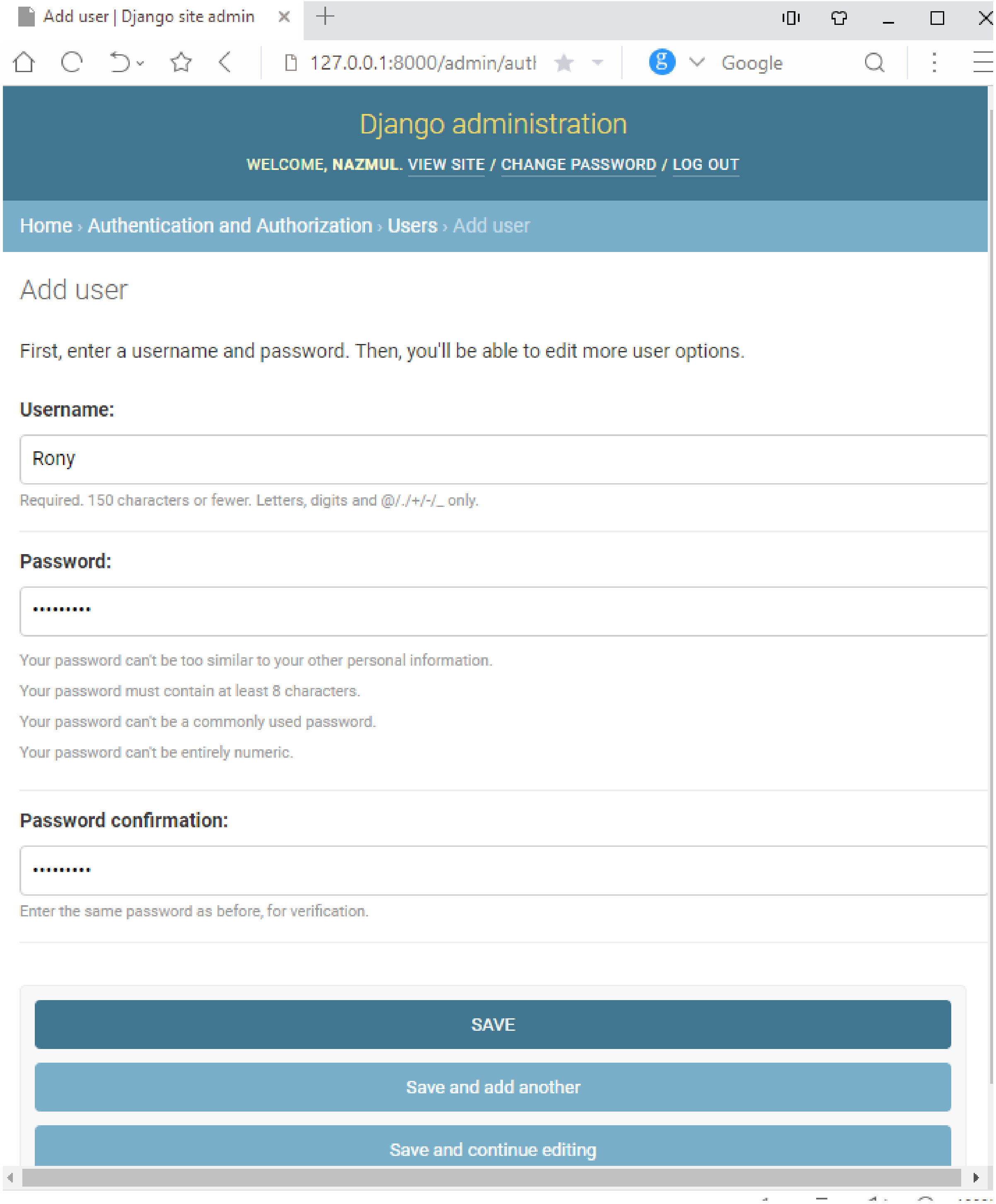
Select user to change

☐	USERNAME	EMAIL ADDRESS	FIRST NAME	LAST NAME
☐	Kabbo			
☐	Nazmul	nazmul1661@gmail.com		
☐	Rezu			
☐	Rony			
☐	Safim			
☐	Zafor			

6 users

6.4 Add New Member

Admin can add new member to navigate this UI. Here admin provide valid input data and click on save then new mass member entry in this system successfully.



Add user | Django site admin
127.0.0.1:8000/admin/autl
Google
Django administration
WELCOME, NAZMUL. VIEW SITE / CHANGE PASSWORD / LOG OUT
Home › Authentication and Authorization › Users › Add user

Add user

First, enter a username and password. Then, you'll be able to edit more user options.

Username: Rony

Required. 150 characters or fewer. Letters, digits and @/./+/-/_ only.

Password: •••••••••

Your password can't be too similar to your other personal information.

Your password must contain at least 8 characters.

Your password can't be a commonly used password.

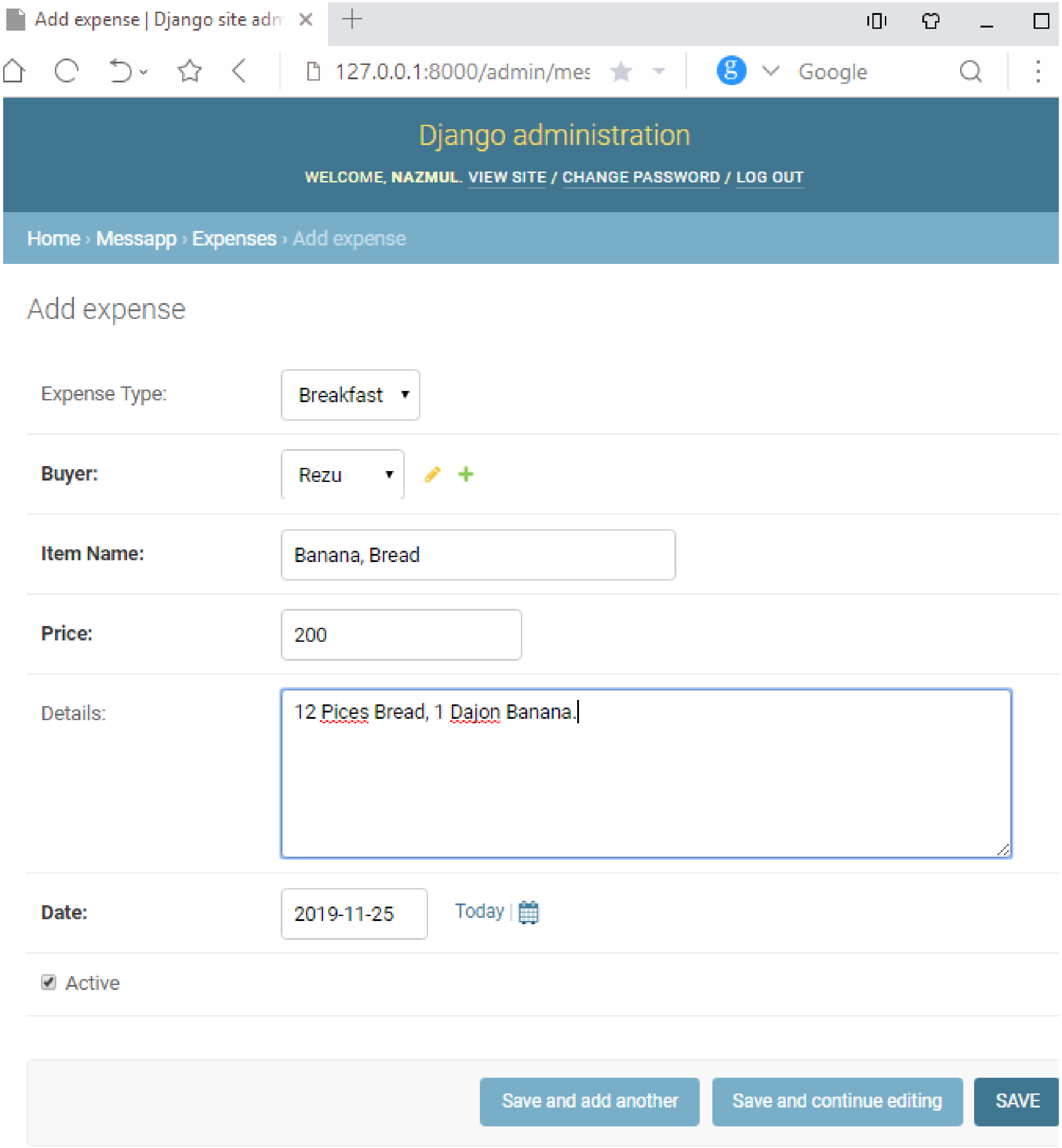
Your password can't be entirely numeric.

Password confirmation: •••••••••

Enter the same password as before, for verification.

6.5 Meal Bazar

Admin can assign every day's bazar among mess members. This work partly done using this UI. Must be given valid input data.


Add expense | Django site adm
127.0.0.1:8000/admin/mes
Google
Django administration
WELCOME, NAZMUL. VIEW SITE / CHANGE PASSWORD / LOG OUT
Home › Messapp › Expenses › Add expense

Add expense

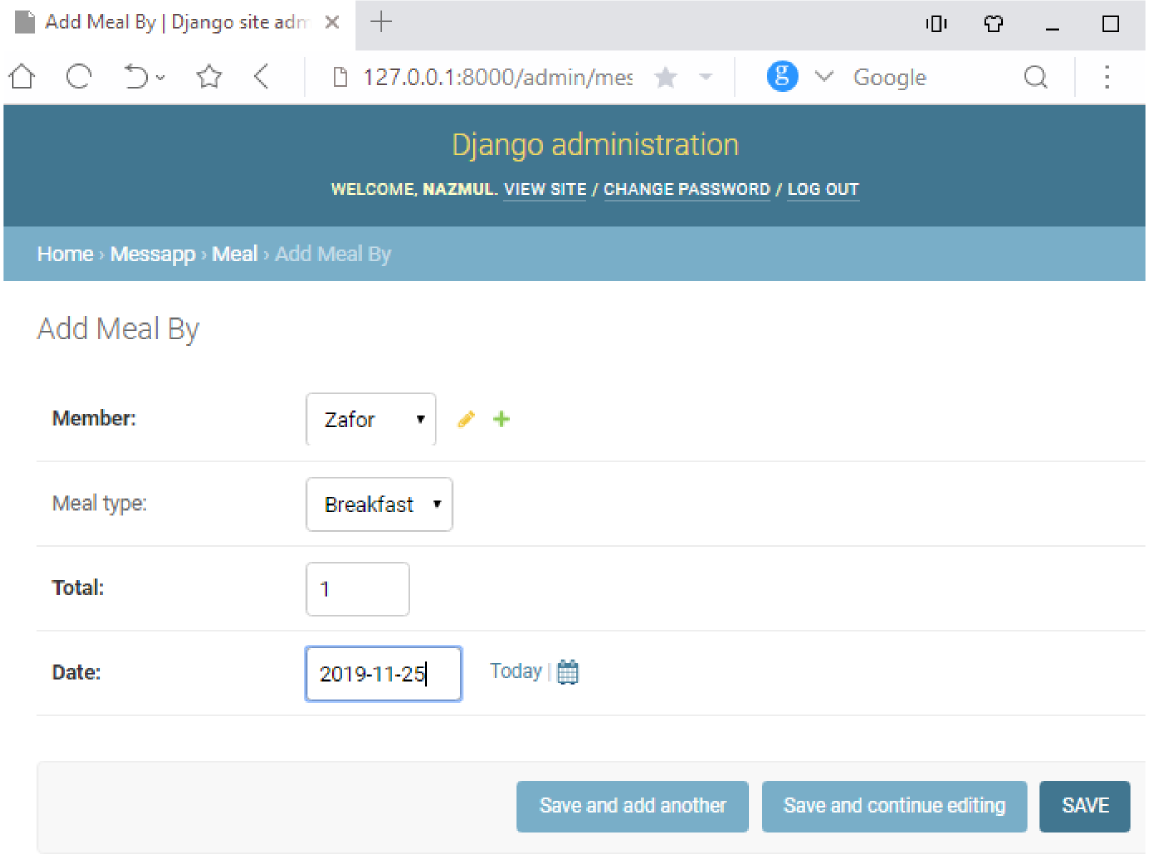
Expense Type:	Breakfast
Buyer:	Rezu
Item Name:	Banana, Bread
Price:	200
Details:	12 Pices Bread, 1 Dajon Banana.
Date:	2019-11-25 Today

☑ Active

Save and add another
Save and continue editing
SAVE

6.6 Meal Entry

Admin and member must be update regular meal in this application. The system has to know how many meal will be cooked today. To use this UI you perfectly done this job.



Django administration

WELCOME, NAZMUL. VIEW SITE / CHANGE PASSWORD / LOG OUT

Home › Messapp › Meal › Add Meal By

Add Meal By

Member:	Zafor
Meal type:	Breakfast
Total:	1
Date:	2019-11-25

CHAPTER 7

PROJECT SUMMARY

7.1 Critical Evolution:

- Knowing information about current all electric currency
- Security issue
- Gathering Requirement issue

7.2 Limitations:

As my system is on international perspective but there is a limitation for users to access at a time.

7.3 Obstacles & Achievements:

- Requirement collecting was an obstacle.
- I have learned many things about Machine learning language, UI/UX design, SqlLite3 query and so on.

7.4 Future Scope:

In future this project have a commercial develop. This project help the bachelors and communicate them like as social media. In near future this project will be developed more to search for rent a house easily. Here gather all kind of information for mass people.

7.5 References

- ➢ https://www.w3schools.com
- ➢ https://python101.pythonlibrary.org
- ➢ https://www.tutorialspoint.com/python/index.htm

CONCLUSION

Mess Bazar Management System is one kind of Human Resource Management System. It will hold everything of any Hostel/Mess Management Count Down. It will help you to keep your Management system in History. People can use this system if he/ she wants to make his life easier when he/she stays with any large Hostel/Mess management system.