BSCS-F22-001

03-134192-018 MIAN RAFEH AHMAD

03-134192-015 HAFIZA MAHAM AZIZ

Helping TaskForce

In partial fulfilment of the requirements for the degree of

Bachelor of Science in Computer Science

Supervisor: Munazza Sher

Department of Computer Sciences

Bahria University, Lahore Campus

June 2023

© Bahria University, 2023

C e r t i f i c a t e

We accept the work contained in the report titled

"Helping TaskForce"

written by

MIAN RAFEH AHMAD

HAFIZA MAHAM AZIZ

as a confirmation to the required standard for the partial fulfilment of the degree of Bachelor of Science in Computer Science.

Approved by:

Supervisor: Munazza Sher

_____ (Signature)

June 20th, 2023

DECLARATION

We hereby declare that this project report is based on our original work except for citations and quotations which have been duly acknowledged. We also declare that it has not been previously and concurrently submitted for any other degree or award at Bahria University or other institutions.

Enrolment	Name	Signature
03-134192-018	MIAN RAFEH AHMAD	
03-134192-015	HAFIZA MAHAM AZIZ	

Date : June 20th, 2023

Specially dedicated to
my beloved grandmother, grandfather, mother, and father
(MIAN RAFEH AHMAD)
my beloved grandmother, grandfather, mother, and father
(HAFIZA MAHAM AZIZ)

ACKNOWLEDGEMENTS

We would like to thank everyone who had contributed to the successful completion of this project. We would like to express our gratitude to my research supervisor, Miss Munazza Sher for her invaluable advice, guidance, and her enormous patience throughout the development of the research.

In addition, we would also like to express our gratitude to our loving parent and friends who had helped and given use encouragement.

MIAN RAFEH AHMAD
HAFIZA MAHAM AZIZ

Helping TaskForce

ABSTRACT

The Helping TaskForce aims to systematize the Community Service Program process and track students (volunteers) working hours with the supervision of different Community Service Program organizers institutes. The currently applied system involves the students (volunteers) registering for the upcoming activities using methods like Google Forms, Microsoft Forms, etc, which will then be used by the activities organizers institute to manage the students (volunteers). This method is tiresome and time-consuming for students, organizers, and the study institute to manage students' Community Service Program working hours.

Establishing and developing the web portal, Helping TaskForce, the process is now modernized by joining the day-to-day services availability list sign-up from one system. This system now saves much time for all the members involved in the Community Service Program, i.e., the Student Institute, Student (volunteers), and Activity Organizer Institutes. The Helping TaskForce Project is implemented in Feature Driven Development Methodology. The database section to store the volunteer's data is handled with My Structured Query Language.

In conclusion, the Helping TaskForce is a system that revolutionize the Community Service Program process, making it more efficient and effective. With the implementation of the web portal, the system now saves time and effort of stakeholders, ultimately making a significant difference in the lives of those who need it most.

TABLE OF CONTENTS

DECLARATION **ii**
ACKNOWLEDGEMENTS **iv**
ABSTRACT **v**
TABLE OF CONTENTS **vi**
LIST OF TABLES **ix**
LIST OF FIGURES **xi**

CHAPTERS

1 INTRODUCTION **1**
1.1 Background 1
1.2 Problem Statements 2
1.3 Aims and Objectives 3
1.4 Scope of Project 3

2 SOFTWARE REQUIREMENT SPECIFICATION (SRS) **5**
2.1 Product Perspectives 5
2.2 User Classes and Characteristics 5
2.2.1 Volunteer Panel 5
2.2.2 Admin Panel 6
2.2.3 Community Service Organizer Panel 6
2.2.4 University Panel 6
2.3 Operating Environment 7
2.4 Design and Implementation Constraints 7
2.5 Assumptions and Dependencies 7

	2.6	Software Environment	8
	2.7	Hardware Environment	8
	2.8	Safety Requirements	8
	2.9	System Requirement Chart	9
3	**DESIGN AND METHODOLOGY**		**11**
	3.1	Use Case Diagram	11
	3.2	Sequence Diagram	22
		3.2.1 Admin	23
		3.2.2 Volunteer	27
		3.2.3 University	33
		3.2.4 Foundation	37
	3.3	Entity-Relation Diagram	44
	3.4	Collaboration Diagram	45
		3.4.1 Admin	45
		3.4.2 Volunteer	49
		3.4.3 Foundation	54
		3.4.4 University	57
4	**DATA AND IMPLMENTATION**		**60**
	4.1	Development Model	60
		4.1.1 Planning and designing	60
		4.1.2 Implementation	60
		4.1.3 Testing	61
		4.1.4 Documentation	61
		4.1.5 Maintenance	61
	4.2	Test Cases	61
	4.3	Technical Aspect of Helping TaskForce	67
		4.3.1 Hardware	67
		4.3.2 Front-end	68
		4.3.3 Back-end	68
5	**USER MANUAL**		**69**

5.1 Helping TaskForce 69
5.1.1 Landing 69
5.1.2 Admin 73
5.1.3 Foundation 76
5.1.4 University 78
5.1.5 Volunteer 81

6 CONCLUSION AND RECOMMENDATIONS 84
6.1 Conclusion 84
6.2 Recommendation 84

REFERENCES 85

APPENDICES 86

LIST OF TABLES

TABLE	TITLE	PAGE
	Table 2.1: System Requirement Chart	9
	Table 3.1: Login UC-1	13
	Table 3.2: Sign-Up UC-2	14
	Table 3.3: Forgot Password UC3	14
	Table 3.4: View / Update Profile UC4	15
	Table 3.5: Manage Users UC5	15
	Table 3.6: View All Activities UC6	16
	Table 3.7: Register Activities UC7	16
	Table 3.8: Generate CV UC8	17
	Table 3.9: Review to activities UC9	17
	Table 3.10: Track Volunteer Hours (CSP) UC10	18
	Table 3.11: Manage Students UC11	18
	Table 3.12: Manage Activities UC12	19
	Table 3.13: Manage Volunteer Hours UC13	19
	Table 3.14: View Volunteer Timeline UC14	20
	Table 3.15: Manage Foundations UC15	20
	Table 3.16: Manage Universities UC16	21
	Table 3.17: View Volunteers UC17	21
	Table 3.18: Receive Reminder Notification UC18	22

Table 4.1: Login with Valid Credentials – Test Case 62

Table 4.2: Login with Invalid Credentials – Test Case 62

Table 4.3: Add User with Unique Email – Test Case 63

Table 4.4: Generate and Print CV – Test Case 63

Table 4.5: Add User with Duplicate Email – Test Case 64

Table 4.6: Receive Reminder Notification Email – Test Case 64

Table 4.7: View Volunteer Timeline – Test Case 65

Table 4.8: Manage Volunteer Hours – Test Case 65

Table 4.9: Track Volunteer Hours (CSP) – Test Case 66

Table 4.10: Register Activity – Test Case 67

LIST OF FIGURES

FIGURE	TITLE	PAGE
	Figure 3.1: Admin – University Use Case	11
	Figure 3.2: Admin – Foundation Use Case	12
	Figure 3.3: Admin – Volunteer Use Case	12
	Figure 3.4: Use Case Diagram – 4	13
	Figure 3.5: Login	23
	Figure 3.6: View All Activities	23
	Figure 3.7: View Users	24
	Figure 3.8: Create User	24
	Figure 3.9: Update User	25
	Figure 3.10: Delete User	25
	Figure 3.11: View All Activities	26
	Figure 3.12: View Account	26
	Figure 3.13: Edit Account	27
	Figure 3.14: Login	27
	Figure 3.15: Register	28
	Figure 3.16: View All Activities	28
	Figure 3.17: View Recommended Activities	29
	Figure 3.18: View Registered Activities	29
	Figure 3.19: Register Activity	30

Figure 3.20: Get Review to Activity 30

Figure 3.21: View Timeline 31

Figure 3.22: Generate CV 31

Figure 3.23: Edit Account 32

Figure 3.24: View Account 32

Figure 3.25: View All Students 33

Figure 3.26: Login 33

Figure 3.27: Register 34

Figure 3.28: Manage Students 34

Figure 3.29: Edit University Profile 35

Figure 3.30: View Timeline 35

Figure 3.31: Edit Account 36

Figure 3.32: View Account 36

Figure 3.33: Login 37

Figure 3.34: Register 37

Figure 3.35: View All Activities 38

Figure 3.36: New Activity 38

Figure 3.37: Update Activity 39

Figure 3.38: Delete Activity 39

Figure 3.39: Manage Attendance 40

Figure 3.40: View Timeline 40

Figure 3.41: Edit Foundation Profile 41

Figure 3.42: View Account 41

Figure 3.43: Edit Account 42

Figure 3.44: Get Users 42

Figure 3.45: Create User 43

Figure 3.46: Update User 43

Figure 3.47: Entity-Relation Diagram 44

Figure 3.48: Login 45

Figure 3.49: View Volunteers 45

Figure 3.50: View Institutions 46

Figure 3.51: View Foundation 46

Figure 3.52: View Profile 47

Figure 3.53: Edit Profile 47

Figure 3.54: View Users 48

Figure 3.55: Add User 48

Figure 3.56: Login 49

Figure 3.57: Sign Up 49

Figure 3.58: View Activities 50

Figure 3.59: View Recommended Activities 50

Figure 3.60: View Registered Activities 51

Figure 3.61: View Timeline 51

Figure 3.62: Generate CV 52

Figure 3.63: Register Activity 52

Figure 3.64: Edit Profile 53

Figure 3.65: View Profile 53

Figure 3.66: Sign Up 54

Figure 3.67: View Activities 54

Figure 3.68: Add Activity 55

Figure 3.69: Manage Presence 55

Figure 3.70: View Profile 56

Figure 3.71: Edit Profile 56

Figure 3.72: Login 57

Figure 3.73: Sign Up 57

Figure 3.74: View Volunteer Timeline 58

Figure 3.75: Update Volunteer Status 58

Figure 3.76: Edit Profile 59

Figure 3.77: View Profile 59

Figure 5.1: Home Page 69

Figure 5.2: Volunteers 70

Figure 5.3: Activities 70

Figure 5.4: Login Page 71

Figure 5.5: Volunteer Registration Page 71

Figure 5.6: University Registration Page 72

Figure 5.7: Community Service Organization Registration Page 72

Figure 5.8: Manage Institute 73

Figure 5.9: Manage Foundation 73

Figure 5.10: Volunteers 74

Figure 5.11: Activities 74

Figure 5.12: Add User 75

Figure 5.13: View / Update Profile 75

Figure 5.14: Activities 76

Figure 5.15: Add Activity 76

Figure 5.16: Activities 77

Figure 5.17: Manage Volunteer Presence 77

Figure 5.18: Manage Students 78

Figure 5.19: Verify Volunteer 78

Figure 5.20: Manage University Users 79

Figure 5.21: Add New User 79

Figure 5.22: View / Update University Profile 80

Figure 5.23: View / Update Profile 80

Figure 5.24: View Activities 81

Figure 5.25: View Recommended Activities 81

Figure 5.26: Timeline 82

Figure 5.27: Generate CV 82

Figure 5.28: Registered Activities 83

Figure 5.29: Review Activity 83

CHAPTER 1

INTRODUCTION

1.1 Background

The landscape of community service was characterized by the limitations and challenges faced in effectively addressing various social and community needs. These challenges included the lack of resources and manpower, the lack of coordination and communication between organizations and volunteers, limited knowledge about community needs, and the difficulty in effectively demonstrating the impact of community service efforts.

Despite these challenges, community service efforts were carried out, and many organizations and volunteers made a positive impact on community. However, the limitations of physical efforts and resources, as well as the lack of coordination and communication, made it difficult for community service efforts to reach their full potential.

A community service program is an integral part of any sector seeking to thrive in the current global climate to strive. In universities, students are encouraged to aid the needy with day-to-day operations, to work as volunteerism. Such volunteerism work has been done on a time-to-time basis and the students (volunteers) pick the day on which they wish to work. The process of serving the needy is done by students (volunteers). The process of serving the needy people in universities has been done under the Community Support Centre with different activities organizers. Different Community Service Centre such as Al-Khidmat [1], Akhuwat [2], UNDP Pakistan [3], Giving-way [4] are working to serve the needy people to support the humanity.

The process to convey the task to students (volunteers) is very tedious because the process is handled by sending message to different groups that were very time-consuming, the effect of this is that the volunteers are not coming up in large numbers to gather more volunteers, to serve the humanity, the Helping TaskForce system is required. The helping TaskForce system is very helpful for the students, universities, and the activities organizers to maintain the records of different activities as well as the record of the volunteers. By leveraging technology, this system now helps to make community service efforts more efficient, effective, and accessible.

1.2 Problem Statements

The process of spreading the information about tasks to volunteers is very tedious because the process is handled by sending messages to different groups that were very time-consuming, the effect of this is that the students (volunteers) are not coming up in large numbers. Community service organizations struggled to coordinate and collaborate with each other, resulting in duplicated efforts and the inefficient use of resources. The lack of a centralized platform for information sharing made it difficult for organizations to raise awareness about community needs and attract volunteers.

This issue is particularly acute for different stakeholders in these activities including volunteers, student institutes, and activities organizers. It takes a bunch of time to search on the internet to find different activities.

The Helping TaskForce system is now very helpful for the students, educational institutes, and the activities organizers to maintain the record of the activities of volunteers. By using this system, the student (volunteer) institute now monitor the record of the activities that are done by their student. Different activities organizer institutes will join this system, so that they add different activities to the system to gather more volunteers. Further, many universities require students to complete a certain number of CSP hours as part of their degree program, by Helping

TaskForce platform tracking these hours can help them to ensure that students are meeting these requirements, so they add these hours to their degree as CSP Hours.

1.3 Aims and Objectives

The objectives of this project are shown as following:

i) To connect the different activities organizers institutes with students (volunteers) and institutes (universities, colleges, etc).
ii) To verify the authentic CSP hours for the student for their degree completion.
iii) To maintain the CSP hours of the student for their degree completion.
iv) To visualize the timeline of a volunteer's progress
v) To recommend various activities to students to register to serve the community.
vi) To send the activity reminder notification (email) to volunteers
vii) To generate the verified CV of each volunteer

1.4 Scope of Project

The process of spreading the information about tasks to volunteers is very tedious because the process is handled by sending messages to different groups that were very time-consuming, this effect that the volunteers are not coming up in large numbers. This issue is particularly acute for different stakeholders in these activities. It takes a bunch of time to search on the internet to find different activities organizers and institutes and register with third-party platforms like Google Forms, Microsoft Forms, etc.

The Helping TaskForce system is very helpful for the students, student institutes, and the activities organizers to maintain the record of the activities of volunteers. By using this system, the volunteer institute, monitor the record of the activities of their student. Moreover, the institute will add their student working hours to the degree as community serving hours. In this project, different activities organizer institutes are registered with the system, so that the organizers add different activities to this system for gathering volunteers. With this system, the community serving hours of the volunteers maintained authentically. Each volunteer will receive a rating of their effort. Moreover, the verified CV of each volunteer will be generated. An activity reminder notification (email) will be sent to the volunteers before the upcoming activities.

CHAPTER 2

SOFTWARE REQUIREMENT SPECIFICATION (SRS)

2.1 Product Perspectives

Helping TaskForce is being developed to allow volunteers to connect with a variety of activities. Helping TaskForce is user-friendly and easy to navigate, with clear and concise information about community service activities. It will serve as a platform for community service activity organizers to manage and promote community service activities. Tracking of volunteer hours will ensure that students are fulfilling their degree requirements for community service programs. It is also important to track the hours to promote community engagement among students and to make a positive impact in their communities. Helping TaskForce is compatible with a wide range of devices and browsers to ensure accessibility for all users.

2.2 User Classes and Characteristics

The system is intended for the following users:

2.2.1 Volunteer Panel

1. Signup
2. Login
3. Edit Profile
4. View All Activities

5. View Recommended Activities

6. View Registered Activities
7. View Activity Details
8. Register Activity
9. View Timeline
10. Download CV
11. Give Review to Activities

2.2.2 Admin Panel

1. Login.
2. Manage Administrators
3. View / Update Profile
4. View Activities
5. View Volunteers
6. View Volunteer Timeline
7. Manage Institutes (Universities)
8. Manage Community Service Organizations

2.2.3 Community Service Organizer Panel

1. Signup
2. Login
3. View / Update Profile
4. Manage Users
5. Manage Activities
6. Give Review to Volunteers
7. Ensure Volunteer Presence
8. View Activities Volunteer Timeline
9. Track Volunteer Hours

2.2.4 University Panel

1. Signup

2. Login
3. View / Update Profile
4. Manage Students Records
5. Update University Record
6. Manage Users
7. Track Volunteer Hours
8. View Volunteer Timeline

2.3 Operating Environment

Helping TaskForce is a web-based platform, any user with an internet connection will be able to use this platform using a browser. Simple search engine can be used to search website.

2.4 Design and Implementation Constraints

The primary constraints of the Helping TaskForce are the user's security and safety. Other constraint is that all the user information and data is stored in a mySQL database, there are chances of hacking the database if there are weak security rules. In short, Helping TaskForce has constraints in terms of security, safety, complexity, and reliability however the constraints can be managed.

2.5 Assumptions and Dependencies

An Updated version of web browser is required to visit our application, probably Chrome or Firefox. Another assumption is that users must have internet connection in their device or desktop, so that status of availability of the application is ensue. Down Internet Connection effect on communication between application and users.

2.6 Software Environment

The following are the software that are needed to use Application:

- Browser: Google Chrome, Mozilla Firefox, Opera, Safari

2.7 Hardware Environment

There is not specific Hardware required to access this product, as the website does not directly interact with the hardware. All devices with internet browsing capability with JavaScript and HTML5 are supported. Extended support to other devices may be implemented in future release of this product. Minimum hardware recommended to access this application are as below.

Computer

A personal computer with either Windows, Linux, or Mac OS installed. At least a browser application installed to view the application is required.

Internet Access Device

Stable internet access is recommended. IP/domain of the system must be allowed in firewall rules if applicable.

2.8 Safety Requirements

To Provide the users with best experience we bring time to time updates in our web application to prevent any bugs and fixes and to provide our users with best user experience.

2.9 System Requirement Chart

Following is the system requirement chart table.

Table 2.1: System Requirement Chart

ID	Priority	Type	Source	Used in USE CASE	Description
1	High	Functional	End User's	U1	User can login after making account
2	High	Functional	End User's	U2	User can sign up if they don't have account
3	High	Functional	End User's	U3	User can reset their password
4	High	Functional	End User's	U4	User can View / Update Account Information
5	Medium	Functional	End User's	U5	Admin / University/ Organizers can Manage their Users
6	High	Functional	End User's	U6	Volunteer can view all Activities
7	High	Functional	End User's	U7	Volunteer can register Activities
8	High	Functional	End User's	U8	Volunteer can generate their CV
9	High	Functional	End User's	U9	Volunteer can give review to activity
10	High	Functional	End User's	U10	University can track the Volunteer Hours
11	Medium	Functional	End User's	U11	University can manage their students
12	High	Functional	End User's	U12	Organizers can manage Activities
13	High	Functional	End User's	U13	Organizers manage Volunteers in terms of attendance
14	High	Functional	End User's	U14	Admin, University and Activity Organizer can

					View Volunteer Timeline
15	High	Functional	End User’s	U15	Administrator can Manage Activities Organizers
16	High	Functional	End User’s	U16	Administrator can Manage Universities
17	Medium	Functional	End User’s	U17	Admin can view Volunteers
18	High	Functional	End User’s	U18	Volunteer can receive notification (email) 1 day before activity

CHAPTER 3

DESIGN AND METHODOLOGY

3.1 Use Case Diagram

Use case Diagrams represent the functionality of the system from a user's point of view. Use cases are used during requirements elicitation and analysis to represent the functionality of the system. Actors are external entities that interact with the system. Example of actors include users like administrator, volunteers, foundations, and universities, etc or another system like central database.

This diagram shows how we can use Helping TaskForce, and the specific actions which are performed by which actors, as shown in figure.

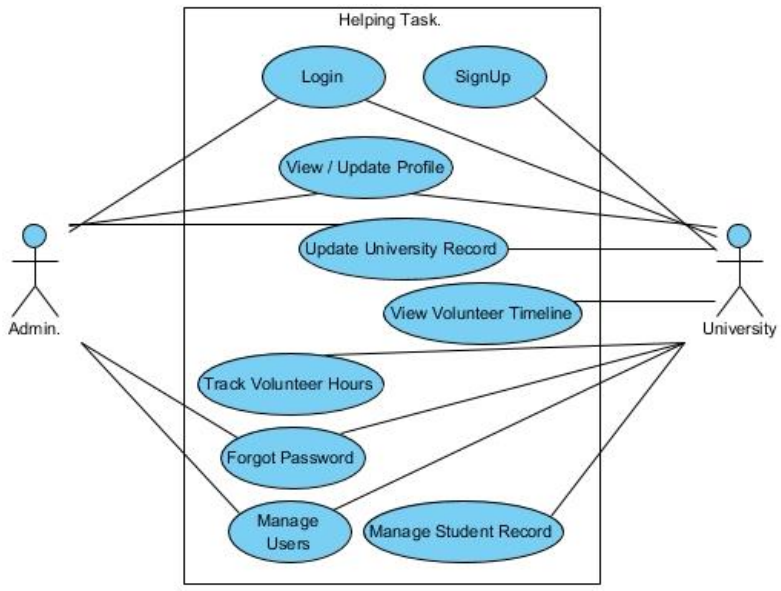

Figure 3.1: Admin – University Use Case

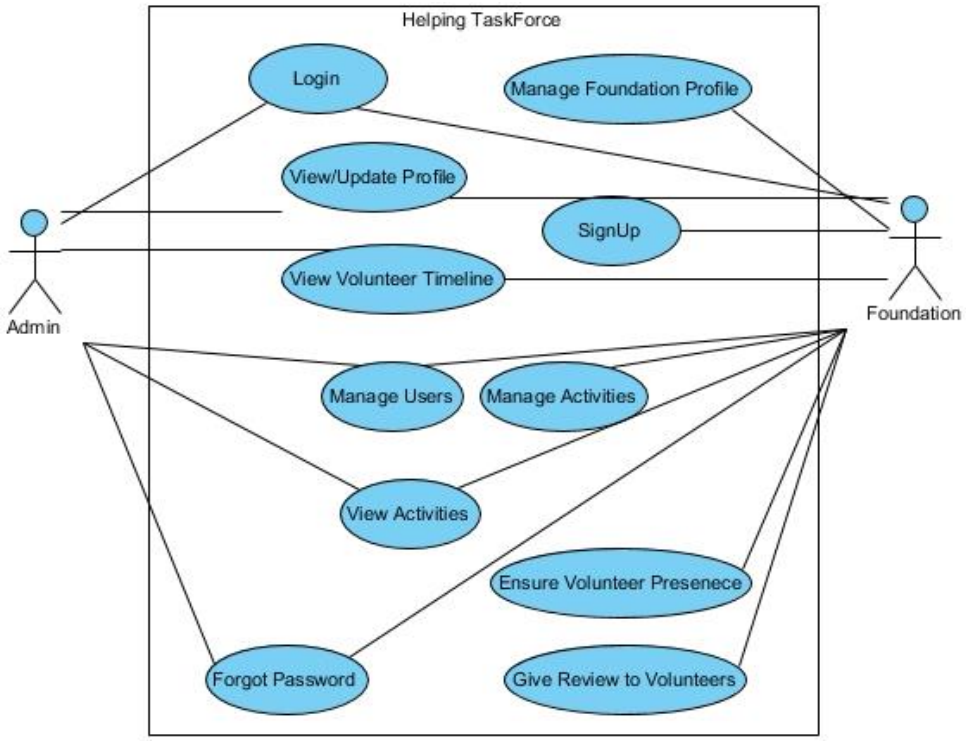

Figure 3.2: Admin – Foundation Use Case

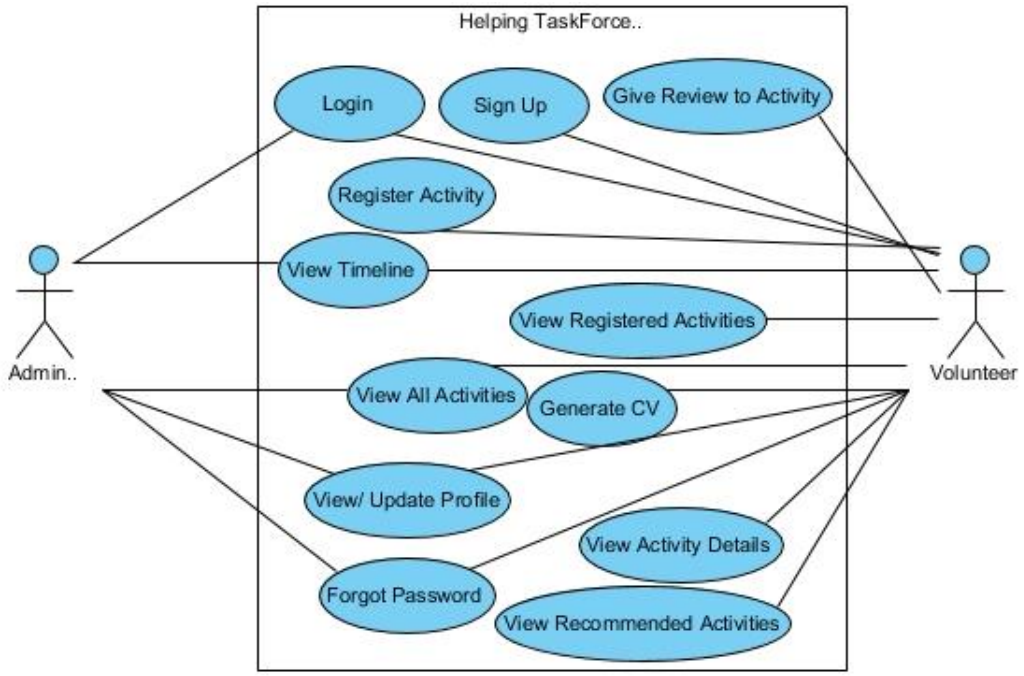

Figure 3.3: Admin – Volunteer Use Case

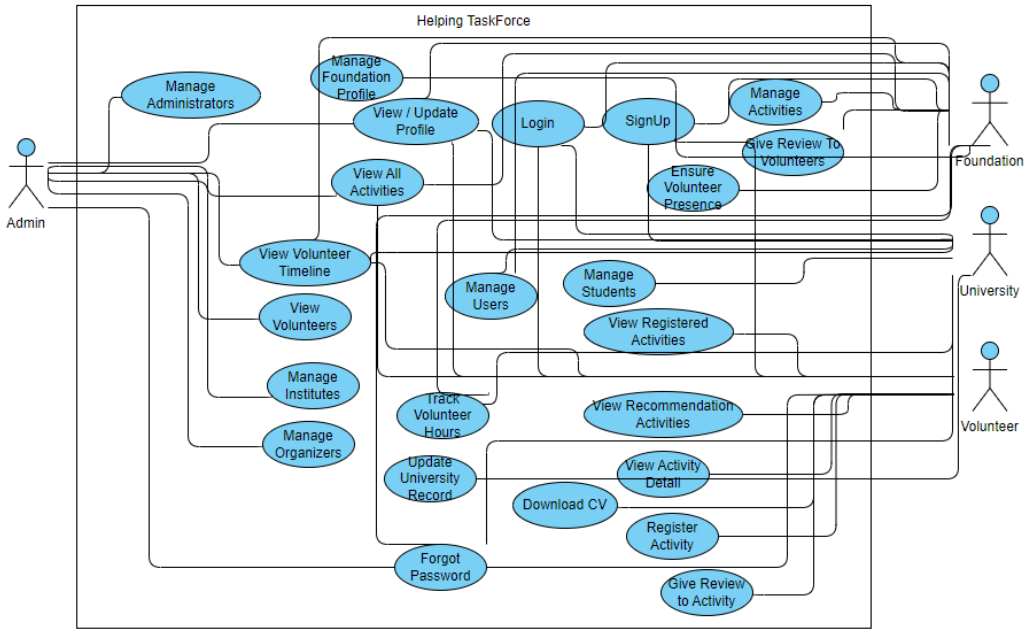

Figure 3.4: Use Case Diagram – 4

Table 3.1: Login UC-1

Use Case ID	UC1
Use Case Name	Login
Primary Actor	Admin, Foundations, Universities, Volunteers
Secondary Actor	No
Description	It will let User to sign in into their accounts to get access over functionalities of webapp.
Pre-Condition	Users must have internet connectivity and access to browser from where our webapp can be accessed.
Scenarios	User must provide email and password. After enters the credentials user click/enter to sign-in. System verifies the credential by the user.
Post-Condition	If Credential is verified, then System will let the user into their respective dashboard. If Credential is not verified, then System will redirect back to login page with invalid credential.
Priority	High
Exception	If user provide wrong credential, system shows the message
Channel to Actor	Via web client (browser)

Table 3.2: Sign-Up UC-2

Use Case ID	UC2
Use Case Name	Sign-Up
Primary Actor	Foundations, Universities, Volunteers
Secondary Actor	No
Description	It will let User to sign up or register to the system to access over functionalities of webapp.
Pre-Condition	Users must have internet connectivity and access to browser from where our webapp can be accessed. User don't have an account before.
Scenarios	User must provide personal details and their linked organization data if any. After enters the data user click/enter to register. System validates the data and register the user.
Post-Condition	If the data is validated correctly, the system will show the message. If the data in not validate, the system will show the error message.
Priority	High
Exception	If user enter any wrong details, the system will show a message.
Channel to Actor	Via web client (browser)

Table 3.3: Forgot Password UC3

Use Case ID	UC3
Use Case Name	Forgot Password
Primary Actor	Admin, Foundations, Universities, Volunteers
Secondary Actor	No
Description	It would change the user password if they forgot.
Pre-Condition	Users must have internet connectivity and access to browser from where our webapp can be accessed and user must be registered before.
Scenarios	Users need to click on forgot-password link from login screen to go to forgot password screen. Users enter the email. Clicks on 'request password'. System verifies the email and send the reset link to the provided email.
Post-Condition	If user email is correct, new reset link send to the user email. If user email is incorrect, message will show to the user,
Priority	High
Exception	If user provide wrong email, System shows the message.
Channel to Actor	Via web client (browser)

Table 3.4: View / Update Profile UC4

Use Case ID	UC4
Use Case Name	View / Update Profile
Primary Actor	Admin, Foundations, Universities, Volunteers
Secondary Actor	No
Description	It would let the user to see and update the personal data
Pre-Condition	Users must have internet connectivity and access to browser from where our webapp can be accessed and user must be login.
Scenarios	Users enter the login credentials, If the credential is correct, user move to their access area, and then click on the setting icon to go to the Account page to edit or view details. If the credential is incorrect, the system shows a message to the user.
Post-Condition	User successfully login to the system, and redirect to their access area, user must click on the setting icon from the sidebar to go to account page to view, or update the details
Priority	High
Exception	If user enter wrong credentials, System shows a message.
Channel to Actor	Via web client (browser)

Table 3.5: Manage Users UC5

Use Case ID	UC5
Use Case Name	Manage Users
Primary Actor	Admin, Foundations, Universities
Secondary Actor	No
Description	Admin, Foundation, Universities main user can perform register, update, and remove functionality on admins, foundations, universities user respectively.
Pre-Condition	Users must have internet connectivity and access to browser from where our webapp can be accessed and must be login.
Scenarios	Admin, Foundation, Universities main user need to login. Admin, Foundation, Universities main user needs to move to the Manage Users tab, click on the add user button to fill up the form to create the user, or edit to change the data. Main User can also remove the accounts.
Post-Condition	After performing certain functionalities admins receive the message.
Priority	High
Exception	If Admin, Foundation, University main user do not use unique email, their respective users will not create and message the shown to them
Channel to Actor	Via web client (browser)

Table 3.6: View All Activities UC6

Use Case ID	UC6
Use Case Name	View All Activities
Primary Actor	Volunteer
Secondary Actor	No
Description	It let the user to see all the activities that are available in the system.
Pre-Condition	Users must have internet connectivity and access to browser from where our webapp can be accessed and must be login.
Scenarios	Volunteers go to the activities tab after login to view all the activities that are posted by different community service organizations.
Post-Condition	User successfully login and must click on the activities tab to view the activities.
Priority	High
Exception	No
Channel to Actor	Via web client (browser)

Table 3.7: Register Activities UC7

Use Case ID	UC7
Use Case Name	Register Activities
Primary Actor	Volunteer
Secondary Actor	No
Description	It let the user to register the activity of their choice from the list that are available in the system.
Pre-Condition	Users must have internet connectivity and access to browser from where our webapp can be accessed and must be login.
Scenarios	Volunteers go to the activities tab after login to view all the activities, and then click on the view button after it go the activity detail page it must click on the register button to register the activity.
Post-Condition	User successfully login and must click on the activities tab to view the activities, then the view button of the activity that he wants to register and click on the register button to register.
Priority	High
Exception	No Exception
Channel to Actor	Via web client (browser)

Table 3.8: Generate CV UC8

Use Case ID	UC8
Use Case Name	Generate CV
Primary Actor	Volunteer
Secondary Actor	No
Description	It let the user to generate or print the CV.
Pre-Condition	Users must have internet connectivity and access to browser from where our webapp can be accessed and must be login.
Scenarios	Volunteers must go to the Generate CV tab, to generate CV. After the page completely load at the end of the page the print button is displayed to print the CV.
Post-Condition	User successfully login and must click on the generate CV tab, to generate the CV. To print the CV, volunteer must go to the bottom of the page and click on the print button.
Priority	High
Exception	No Exception
Channel to Actor	Via web client (browser)

Table 3.9: Review to activities UC9

Use Case ID	UC9
Use Case Name	Review to activity
Primary Actor	Volunteer
Secondary Actor	No
Description	It let the user to give review the activity that they registered.
Pre-Condition	Users must have internet connectivity and access to browser from where our webapp can be accessed and must be login.
Scenarios	Volunteers go to the registered activities tab after login to view all the registered activities, and then click on the edit button against the activity and give the review to this activity and save it.
Post-Condition	User successfully login and must click on the registered activities tab to view the registered activities, then the click on edits button of the activity that he wants to review, write the review, and click on the submit button.
Priority	High
Exception	No
Channel to Actor	Via web client (browser)

Table 3.10: Track Volunteer Hours (CSP) UC10

Use Case ID	UC10
Use Case Name	Track Volunteer Hours (CSP)
Primary Actor	University
Secondary Actor	No
Description	It let the user to view the track volunteer (student) hours.
Pre-Condition	Users must have internet connectivity and access to browser from where our webapp can be accessed and must be login.
Scenarios	University user after login go to the student tab, search the student, and take the CSP hours into the consideration.
Post-Condition	User successfully login and must click on the student tab to view the students, after that to view the specific record, it must write some necessary detail in the search field.
Priority	High
Exception	No
Channel to Actor	Via web client (browser)

Table 3.11: Manage Students UC11

Use Case ID	UC11
Use Case Name	Manage Students
Primary Actor	University
Secondary Actor	No
Description	It let the user to manage the students in term to verify that the student (volunteer) belongs to their institute.
Pre-Condition	Users must have internet connectivity and access to browser from where our webapp can be accessed and must be login.
Scenarios	University user after login go to the student tab, search the student, and click on the edit button to verify the student (volunteer).
Post-Condition	User successfully login and must click on the student tab to view the students, and to verify it must click on the edit button to change the verification status.
Priority	Medium
Exception	No
Channel to Actor	Via web client (browser)

Table 3.12: Manage Activities UC12

Use Case ID	UC12
Use Case Name	Manage Activities
Primary Actor	Foundation (Community Service Organizer)
Secondary Actor	No
Description	It let the user to manage the activities that are posted by them.
Pre-Condition	Users must have internet connectivity and access to browser from where our webapp can be accessed and must be login.
Scenarios	Foundation has to login to access the activities page, after login success, it goes to activities tab from where it can add, edit, delete the activities.
Post-Condition	Foundation user successful login, then it goes to activities tab, and perform the add, edit, or delete operation.
Priority	High
Exception	No
Channel to Actor	Via web client (browser)

Table 3.13: Manage Volunteer Hours UC13

Use Case ID	UC13
Use Case Name	Manage Volunteers Hours
Primary Actor	Foundation (Community Service Organizer)
Secondary Actor	No
Description	It let the user to manage the volunteer hour against the activity. When the user marks the volunteer as present the volunteer CSP hours is assigned to respective volunteer.
Pre-Condition	Users must have internet connectivity and access to browser from where our webapp can be accessed and must be login.
Scenarios	After the user login, to mark the volunteer hour, he/she goes to activities tab and click on the select box against the activity the submenu shows clicks on the manage attendance. When the user marks the volunteer as present the volunteer CSP hours is assigned to respective volunteer.
Post-Condition	User successfully login and must click on the activities tab, after page open, it clicks on the select box in the activity row and click on the manage attendance in submenu to mark attendance.
Priority	High
Exception	No
Channel to Actor	Via web client (browser)

Table 3.14: View Volunteer Timeline UC14

Use Case ID	UC14
Use Case Name	View Volunteer Timeline
Primary Actor	Admin, Volunteer, Foundation, University
Secondary Actor	No
Description	It let the user to view the volunteer timeline.
Pre-Condition	Users must have internet connectivity and access to browser from where our webapp can be accessed and must be login.
Scenarios	After the user login, for admin they must go to volunteer tab and click on the view icon to view, for volunteer it must click on the Timeline to view, for foundation it must click on the manage activities then manage volunteer and click on view button, for university go to student tab and click on the view icon against the student.
Post-Condition	User successfully login and must click on the respective button against each access area to view volunteer timeline.
Priority	High
Exception	No
Channel to Actor	Via web client (browser)

Table 3.15: Manage Foundations UC15

Use Case ID	UC15
Use Case Name	Manage Foundations
Primary Actor	Admin
Secondary Actor	Foundations
Description	It let the user to manage the foundations.
Pre-Condition	Users must have internet connectivity and access to browser from where our webapp can be accessed and must be login.
Scenarios	After the admin login, and must click on the foundations tab, and perform the add, edit, delete operation on the foundations.
Post-Condition	User successfully login and must click on the foundations and perform the operation like add, edit, delete.
Priority	High
Exception	No
Channel to Actor	Via web client (browser)

Table 3.16: Manage Universities UC16

Use Case ID	UC16
Use Case Name	Manage Universities
Primary Actor	Admin
Secondary Actor	Universities
Description	It let the user to manage the universities.
Pre-Condition	Users must have internet connectivity and access to browser from where our webapp can be accessed and must be login.
Scenarios	After the admin login, and must click on the universities tab, and perform the add, edit, delete operation on the universities.
Post-Condition	User successfully login and must click on the universities and perform the operation like add, edit, delete.
Priority	High
Exception	No
Channel to Actor	Via web client (browser)

Table 3.17: View Volunteers UC17

Use Case ID	UC17
Use Case Name	View Volunteers
Primary Actor	Admin
Secondary Actor	No
Description	It let the user to view all the volunteers that are added in the system.
Pre-Condition	Users must have internet connectivity and access to browser from where our webapp can be accessed and must be login.
Scenarios	After the admin login, it must click on the volunteer tab to view all the volunteers that are added in the system.
Post-Condition	User successfully login and must click on the volunteer tab to view volunteers
Priority	Medium
Exception	No
Channel to Actor	Via web client (browser)

Table 3.18: Receive Reminder Notification UC18

Use Case ID	UC18
Use Case Name	Receive Reminder Notification
Primary Actor	System
Secondary Actor	Volunteer
Description	It let the system to send the reminder notification (email) to the volunteers automatically.
Pre-Condition	System must be active all the time and have proper functionality to do job.
Scenarios	System setup successfully after specific time it on setup, the system sends the email to all volunteers whose registered activities is tomorrow. System does this automatically if the CRON Job setup correctly in the system.
Post-Condition	Volunteer received email
Priority	High
Exception	No
Channel to Actor	Via web client (browser)

3.2 Sequence Diagram

A Sequence diagram is a visual representation that illustrates the interactions and flow of messages between objects or components in a system. It captures the sequence of method invocations and the order in which they occur, providing valuable insights into the system's dynamic behaviour. By displaying the exchange of messages and the lifelines of objects involved, Sequence diagrams aid in understanding the collaboration and coordination among different elements of a system. This makes the Sequence diagram a very useful tool to visualize the runtime behaviour of a system, enabling effective communication and analysis during the design and implementation phases.

The following are the sequence diagrams depicting the interactions and responsibilities assigned to each role in our project's system.

3.2.1 Admin

3.2.1.1 Login

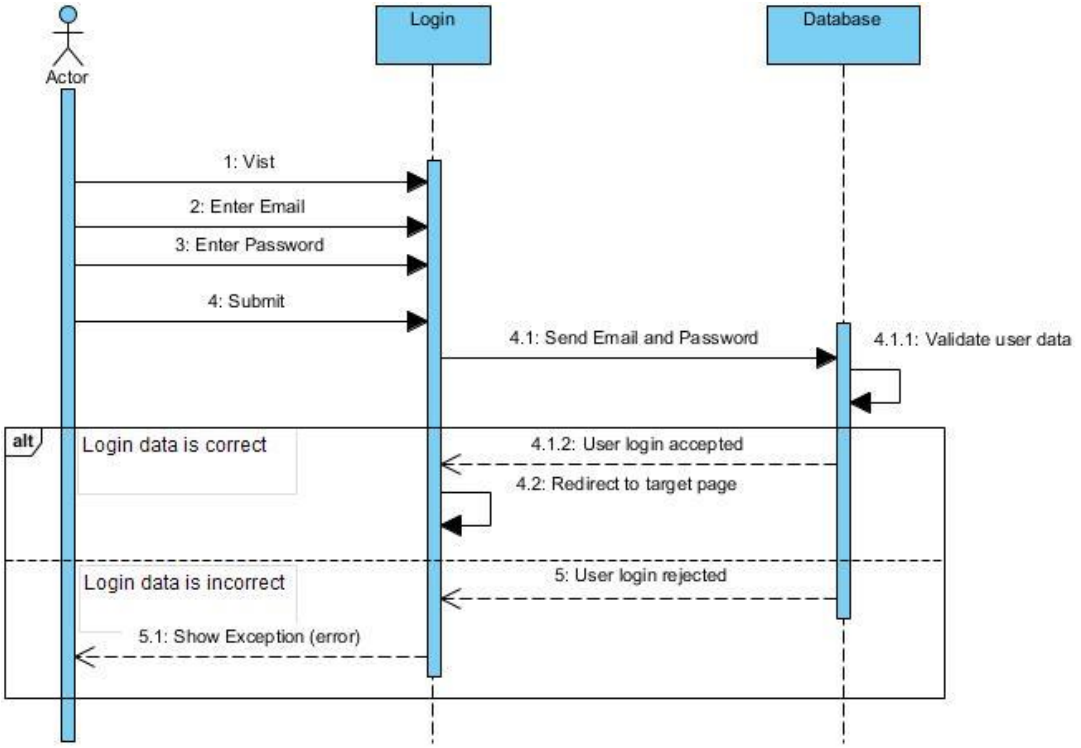

Figure 3.5: Login

3.2.1.2 View All Activities

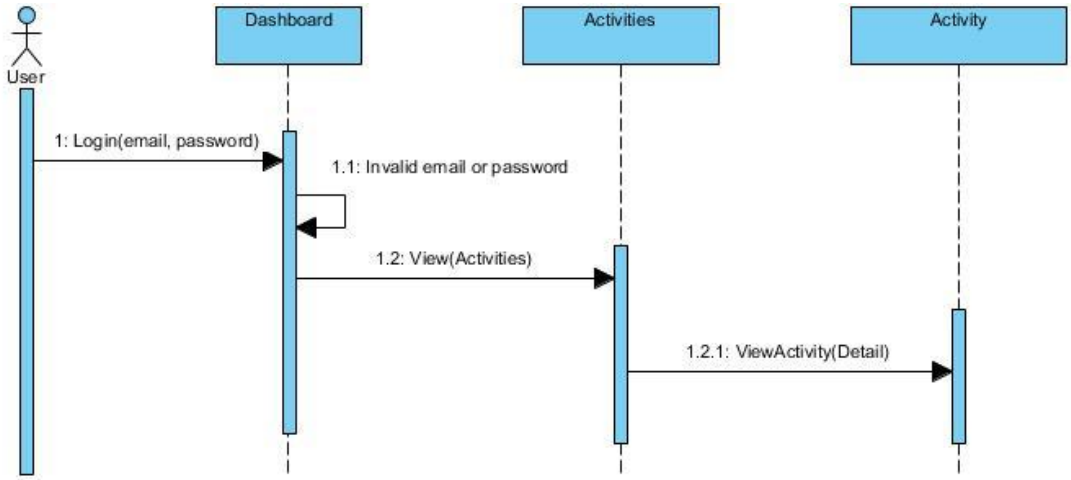

Figure 3.6: View All Activities

3.2.1.3 View Users

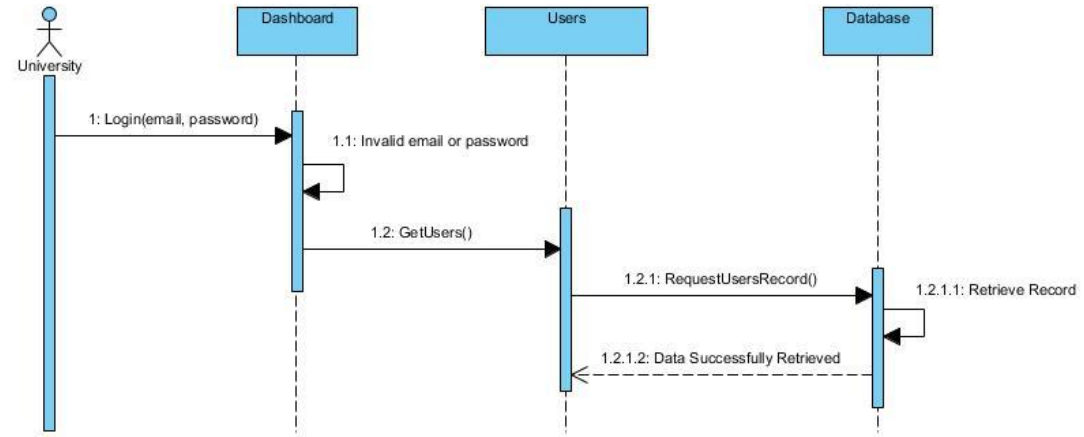

Figure 3.7: View Users

3.2.1.4 Create User

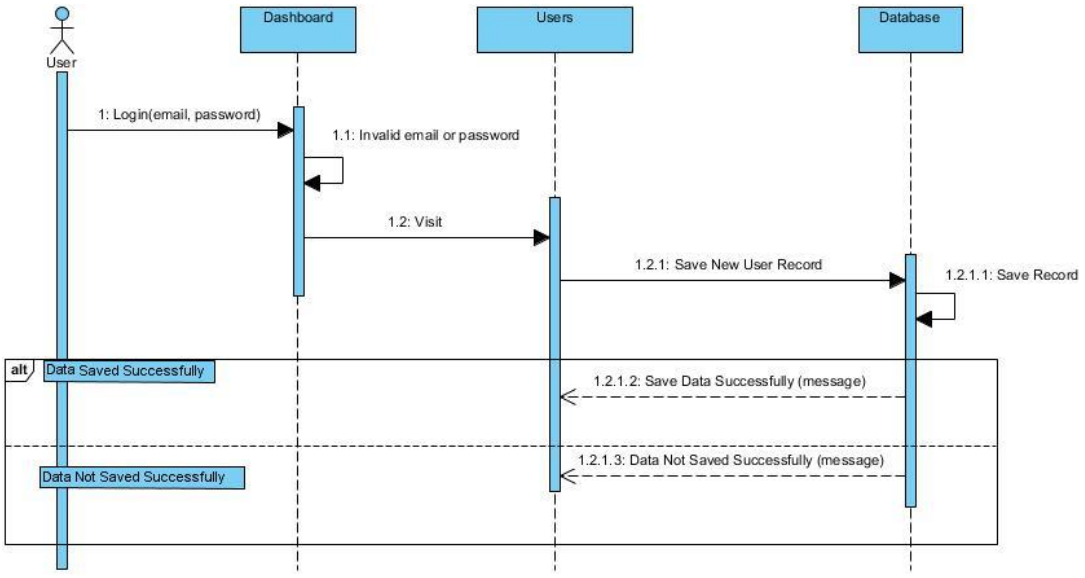

Figure 3.8: Create User

3.2.1.5 Update User

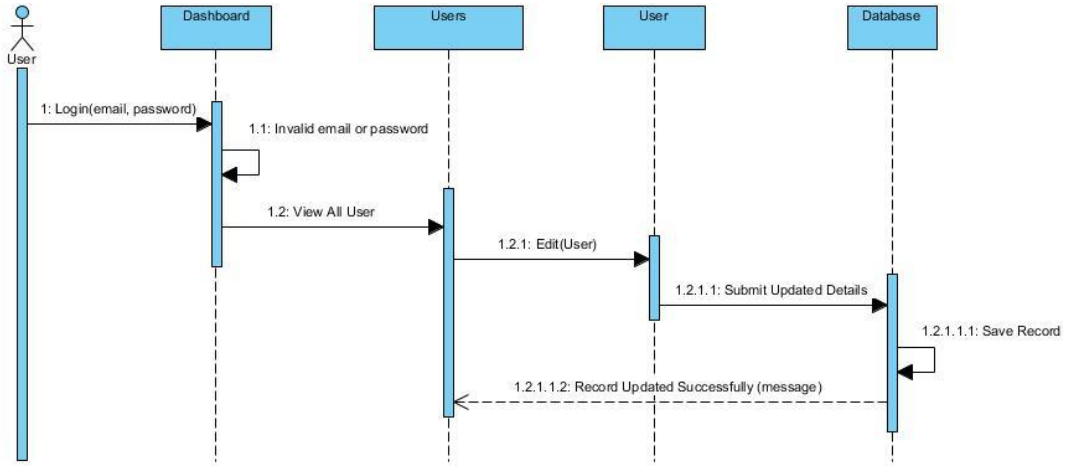

Figure 3.9: Update User

3.2.1.6 Delete User

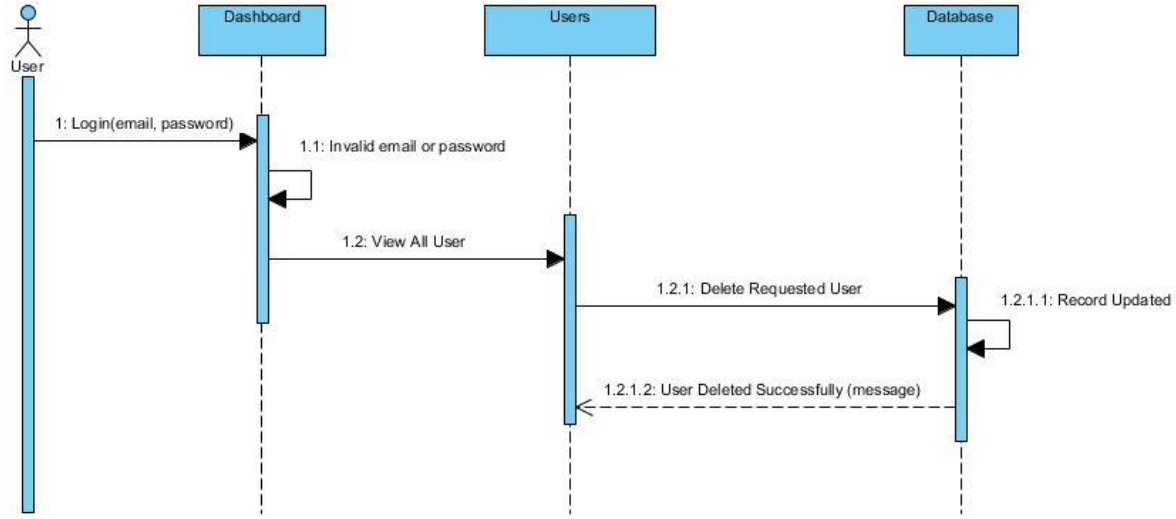

Figure 3.10: Delete User

3.2.1.7 View All Activities

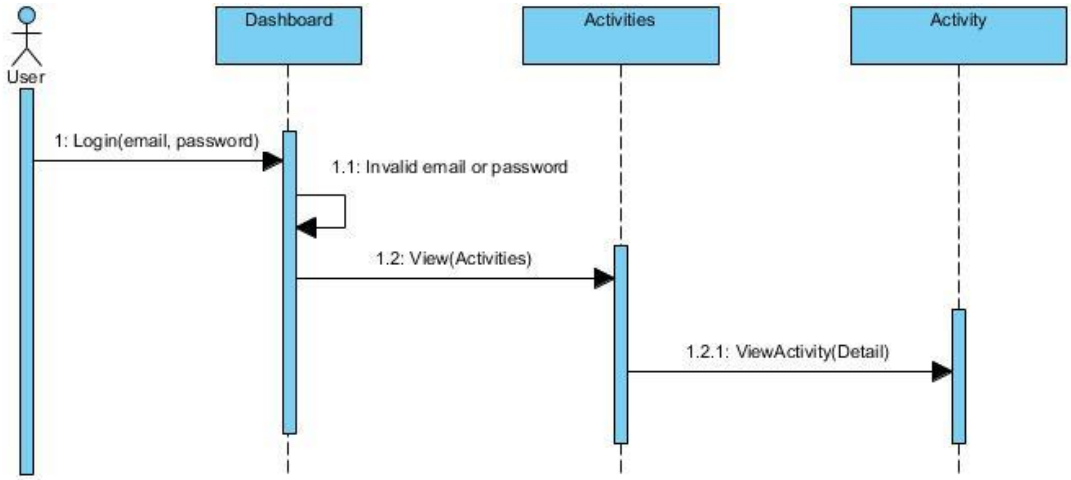

Figure 3.11: View All Activities

3.2.1.8 View Account

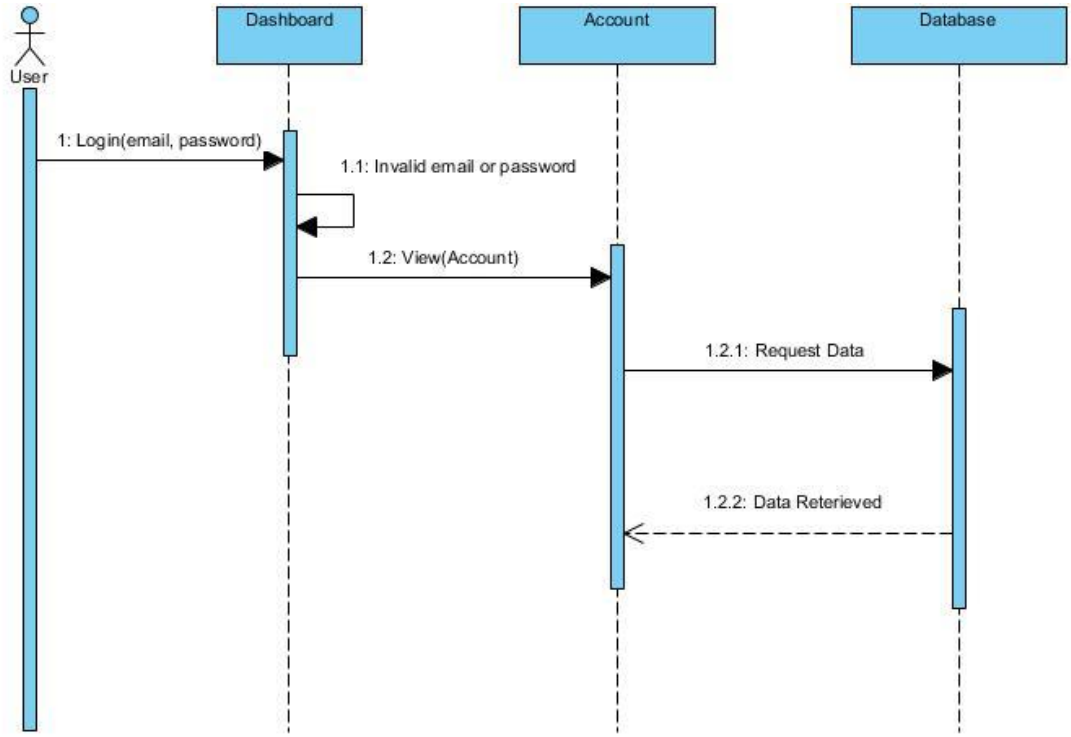

Figure 3.12: View Account

3.2.1.9 Edit Account

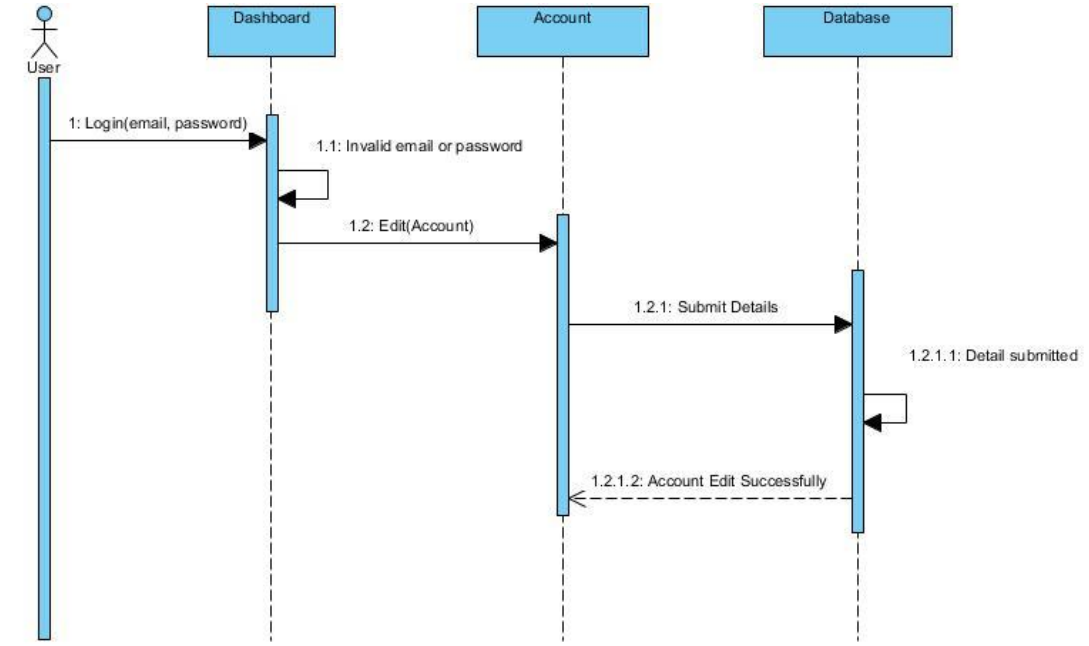

Figure 3.13: Edit Account

3.2.2 Volunteer

3.2.2.1 Login

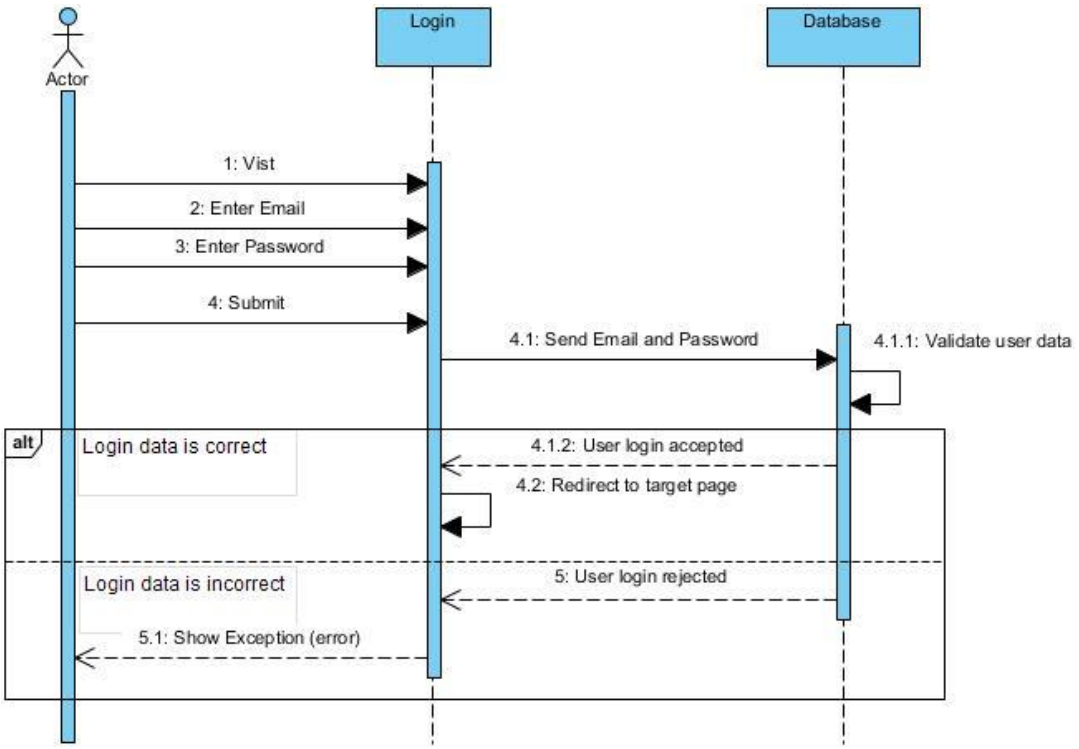

Figure 3.14: Login

3.2.2.2 Register

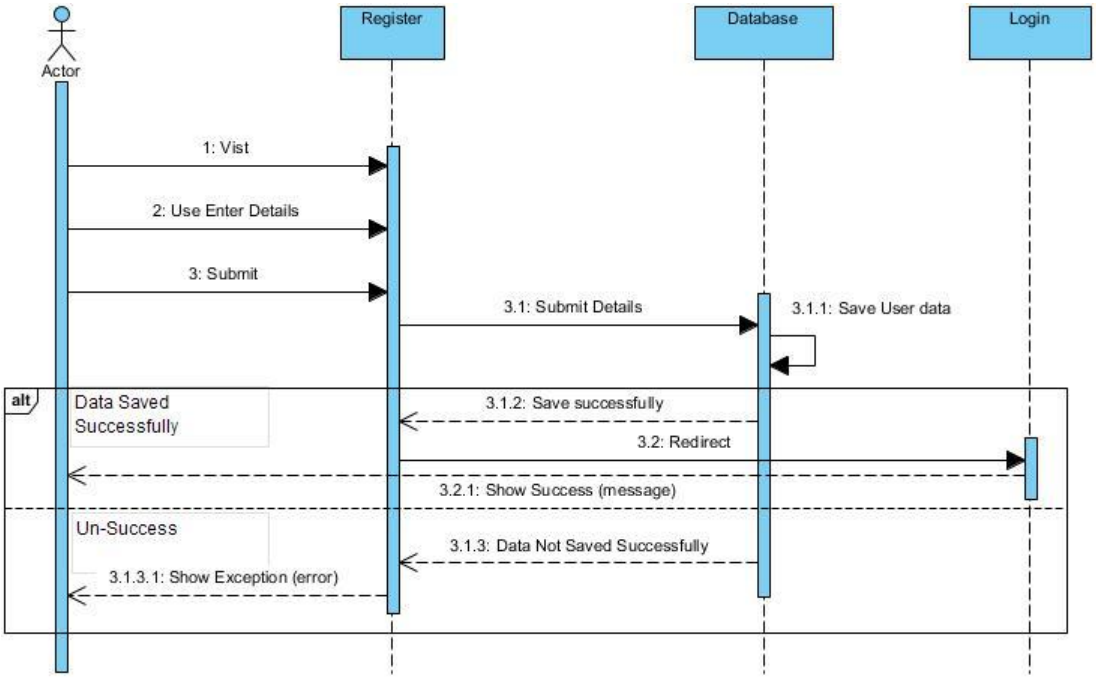

Figure 3.15: Register

3.2.2.3 View All Activities

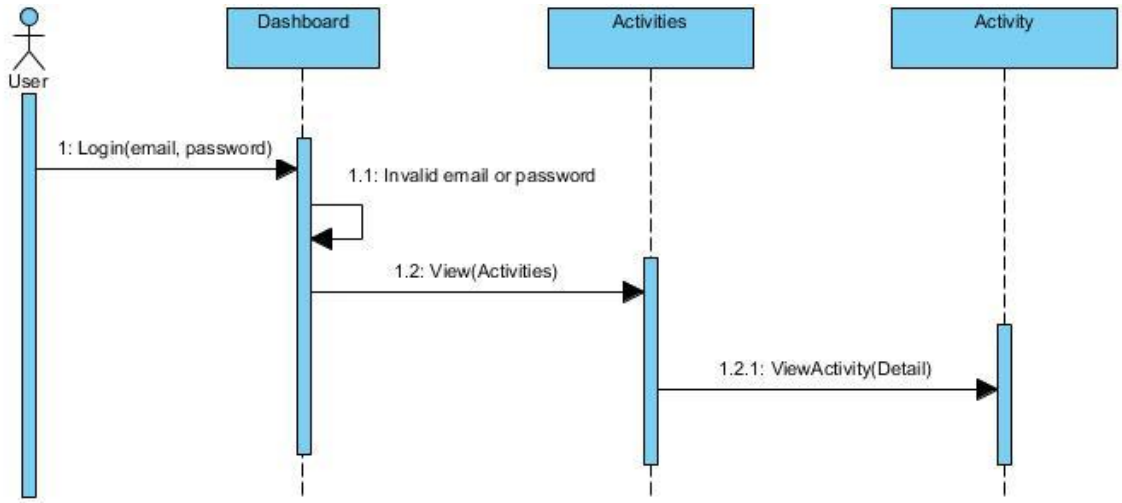

Figure 3.16: View All Activities

3.2.2.4 View Recommended Activities

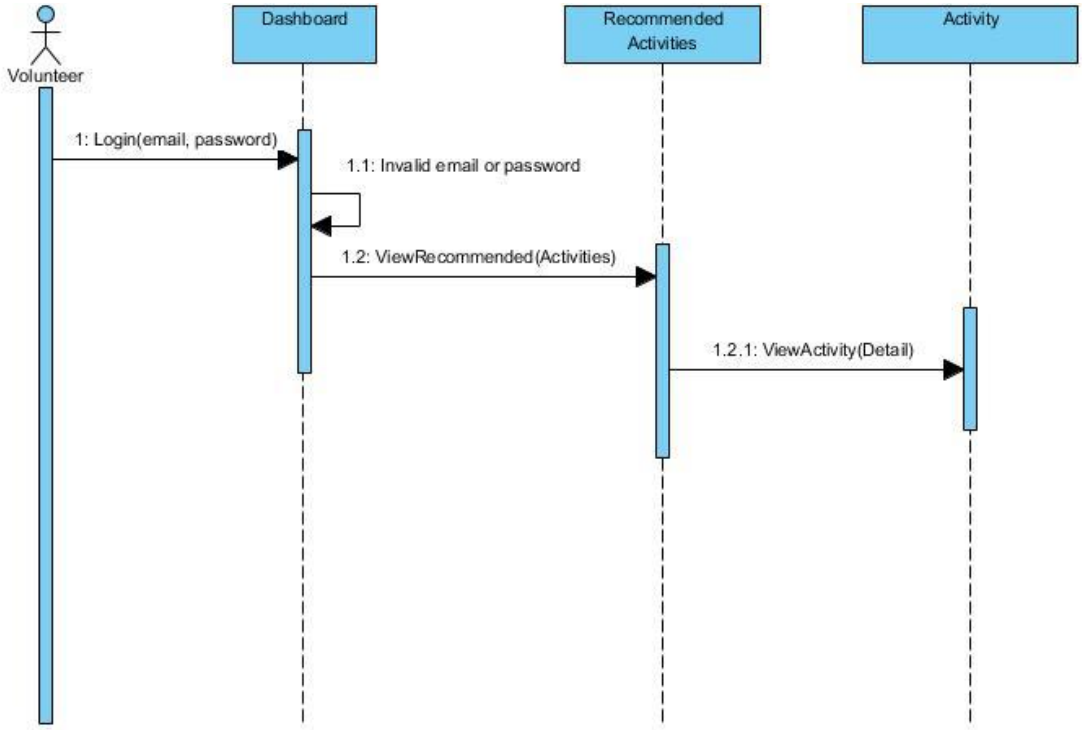

Figure 3.17: View Recommended Activities

3.2.2.5 View Registered Activities

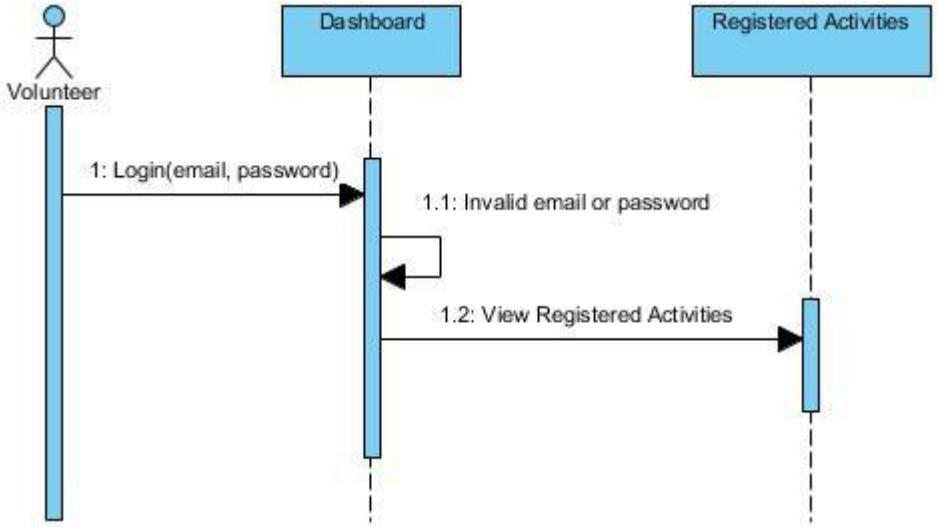

Figure 3.18: View Registered Activities

3.2.2.6 Register Activity

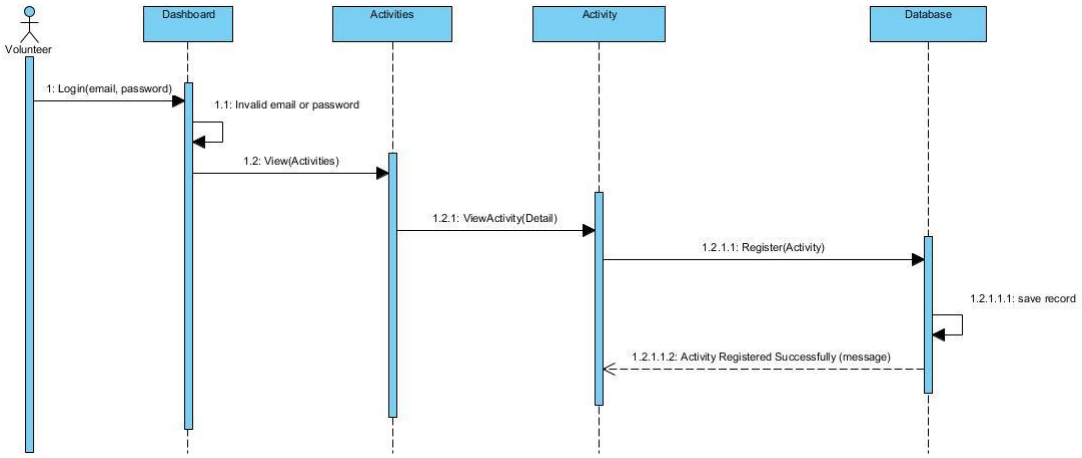

Figure 3.19: Register Activity

3.2.2.7 Get Review to Activity

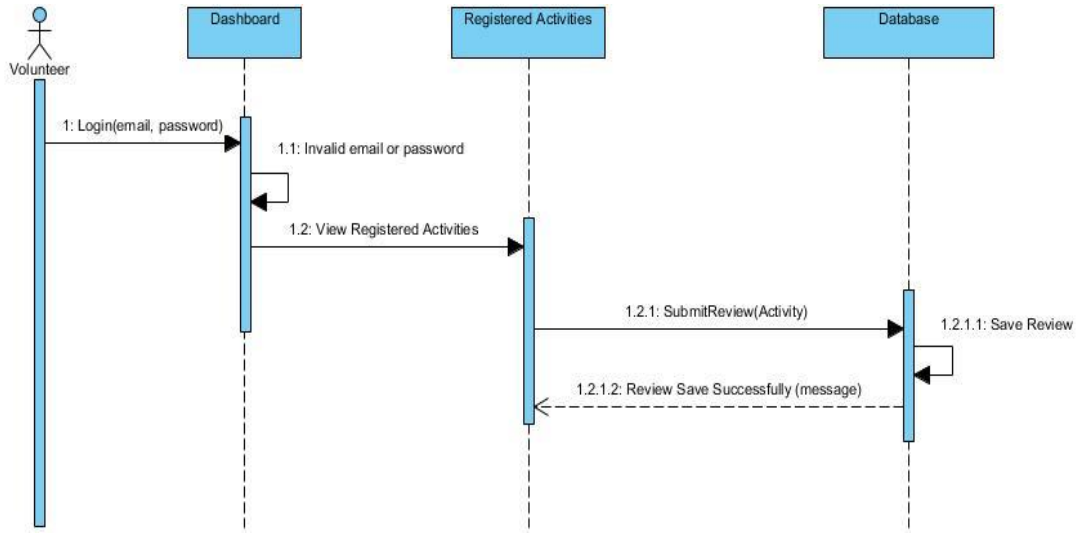

Figure 3.20: Get Review to Activity

3.2.2.8 View Timeline

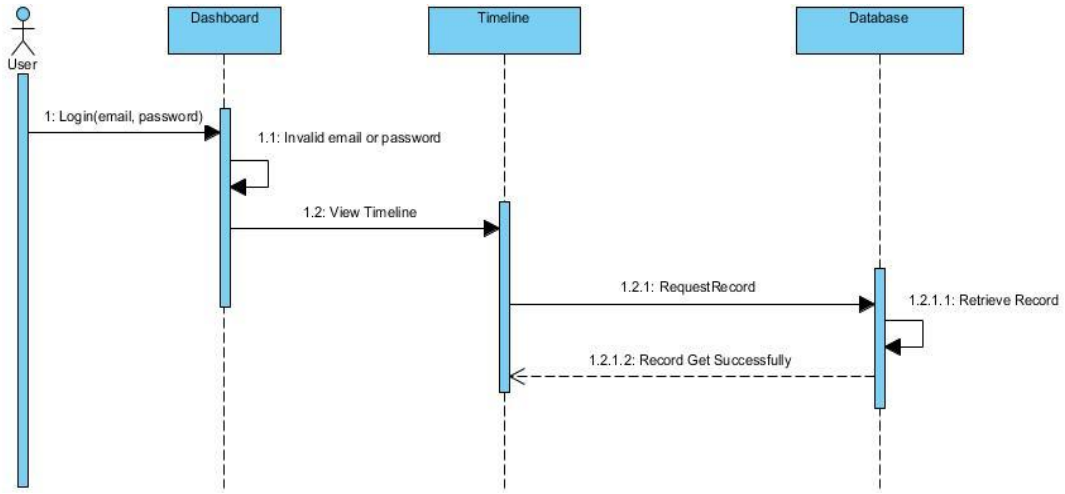

Figure 3.21: View Timeline

3.2.2.9 Generate CV

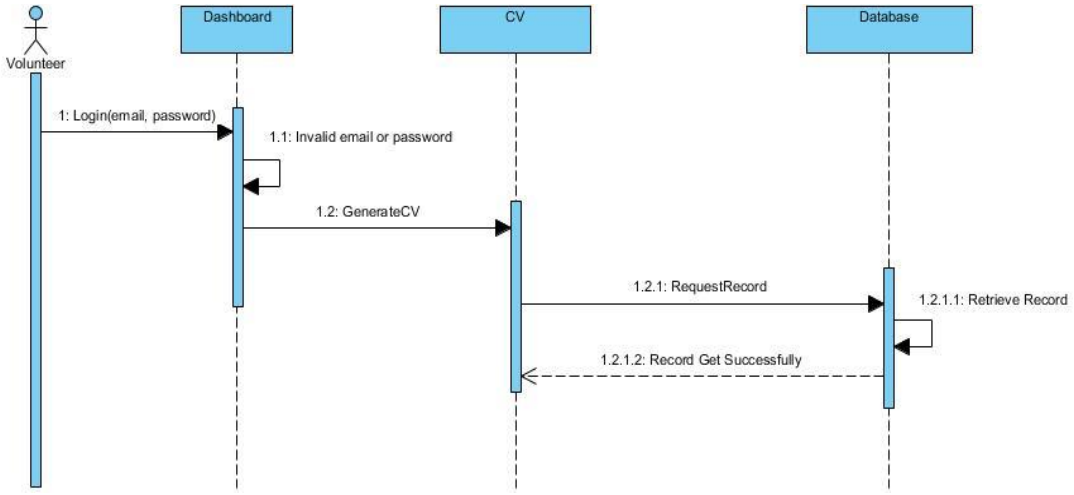

Figure 3.22: Generate CV

3.2.2.10 Edit Account

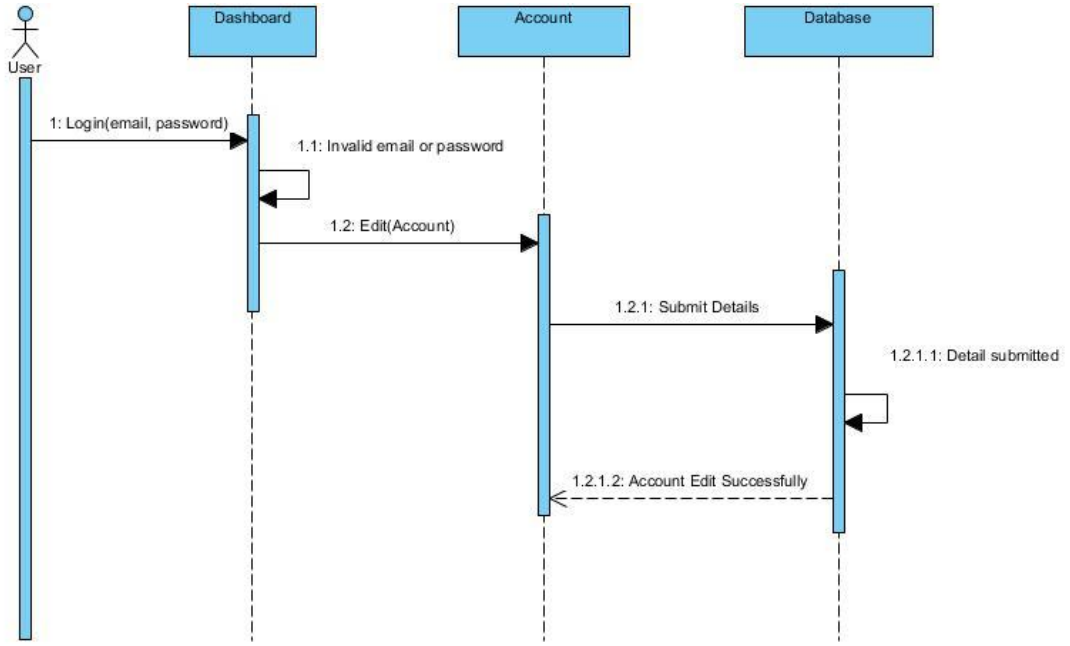

Figure 3.23: Edit Account

3.2.2.11 View Account

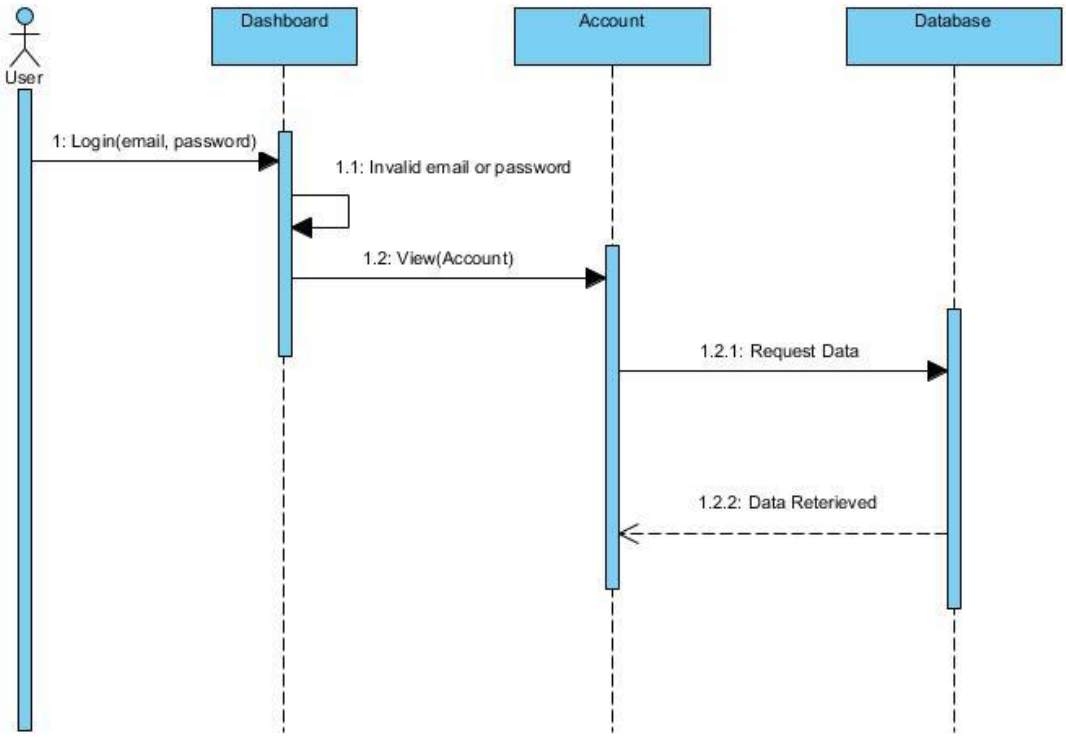

Figure 3.24: View Account

3.2.3 University

3.2.3.1 View All Students

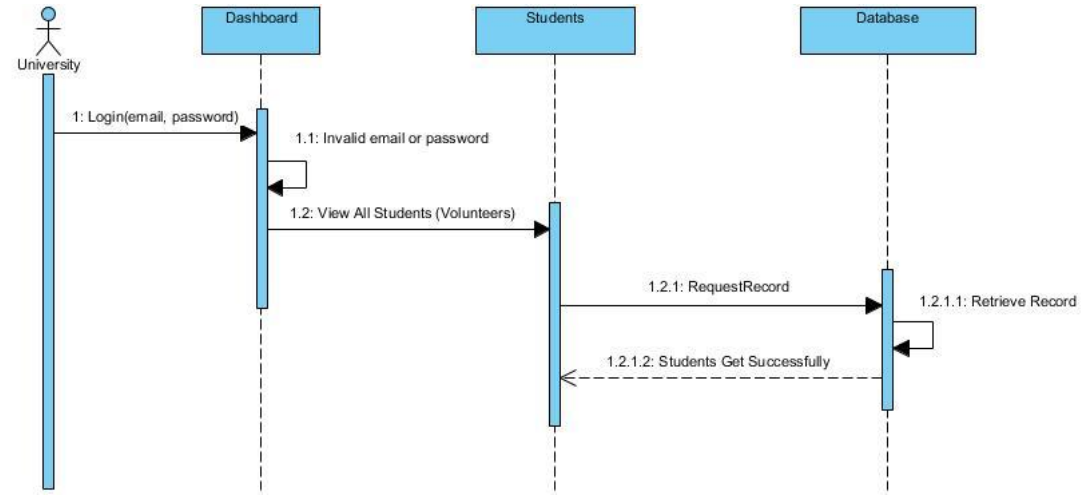

Figure 3.25: View All Students

3.2.3.2 Login

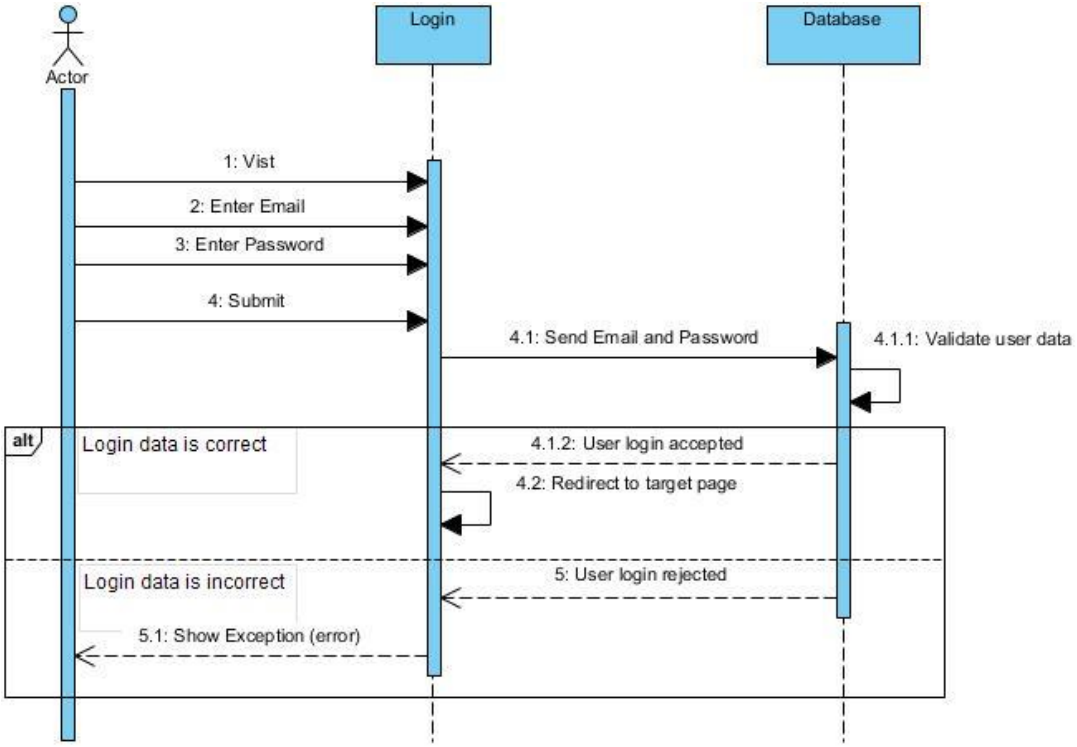

Figure 3.26: Login

3.2.3.3 Register

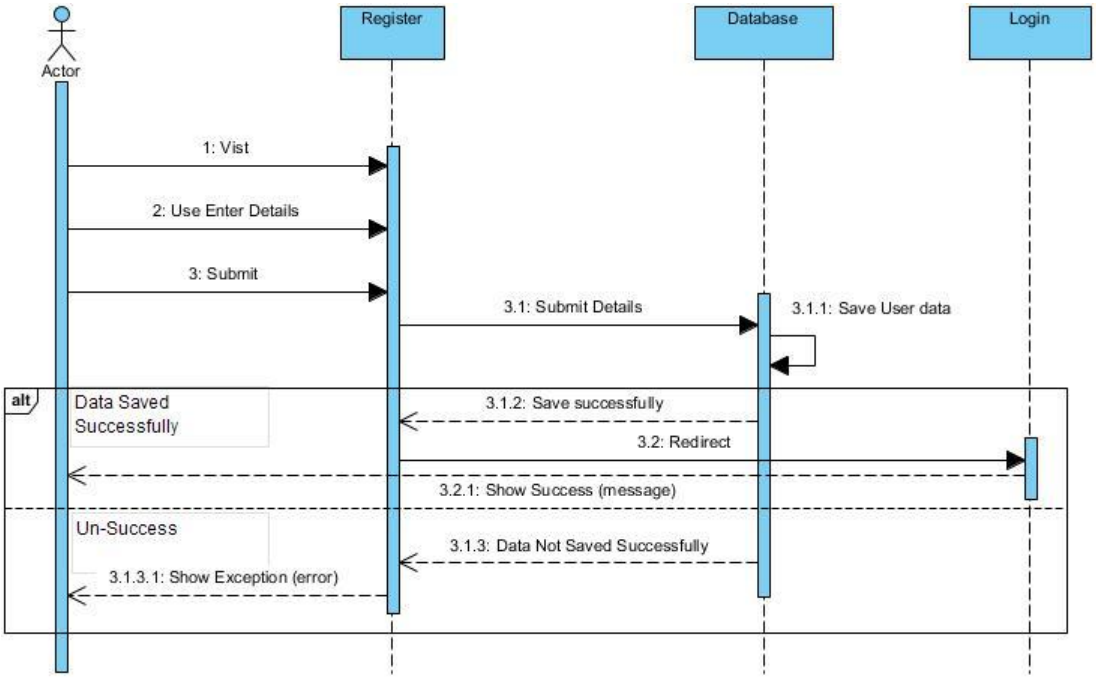

Figure 3.27: Register

3.2.3.4 Manage Student

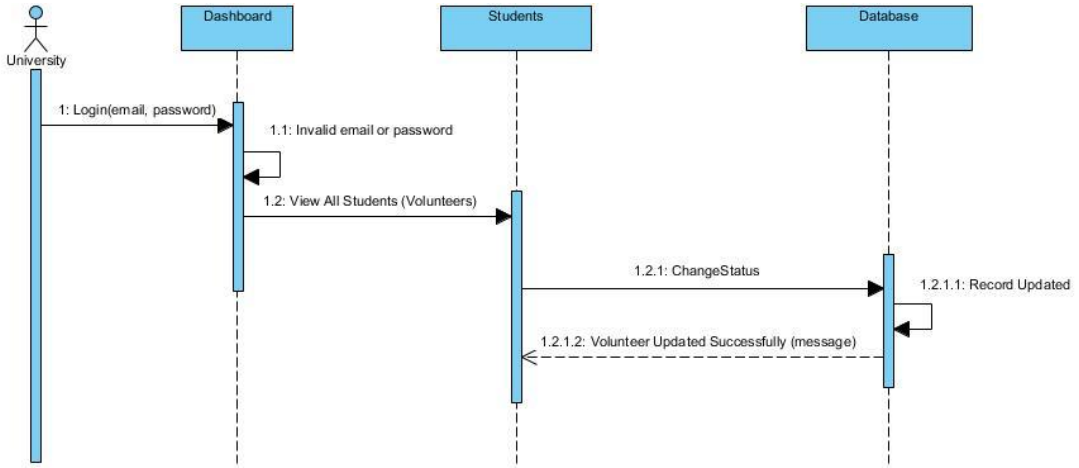

Figure 3.28: Manage Students

3.2.3.5 Edit University Profile

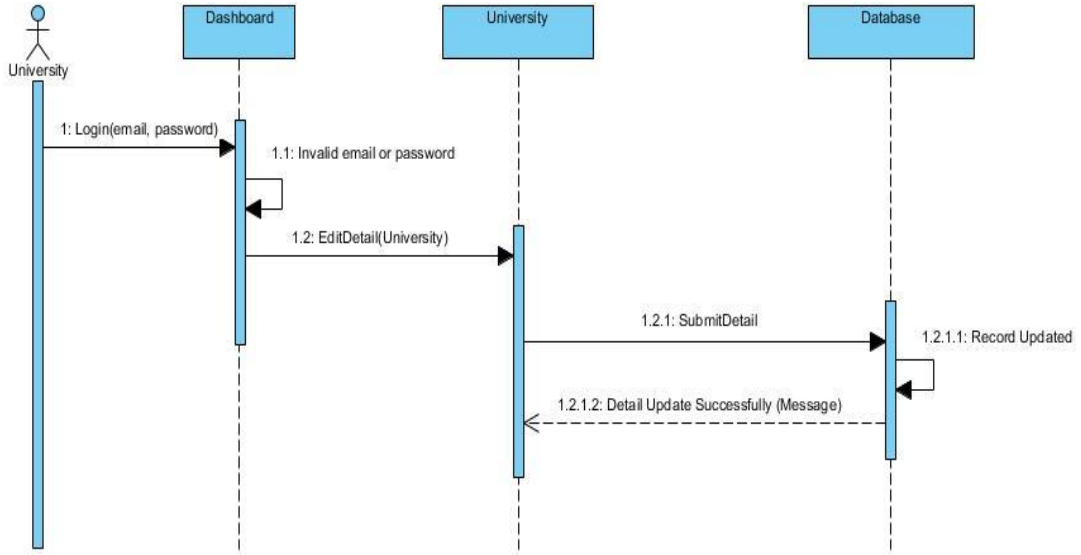

Figure 3.29: Edit University Profile

3.2.3.6 View Timeline

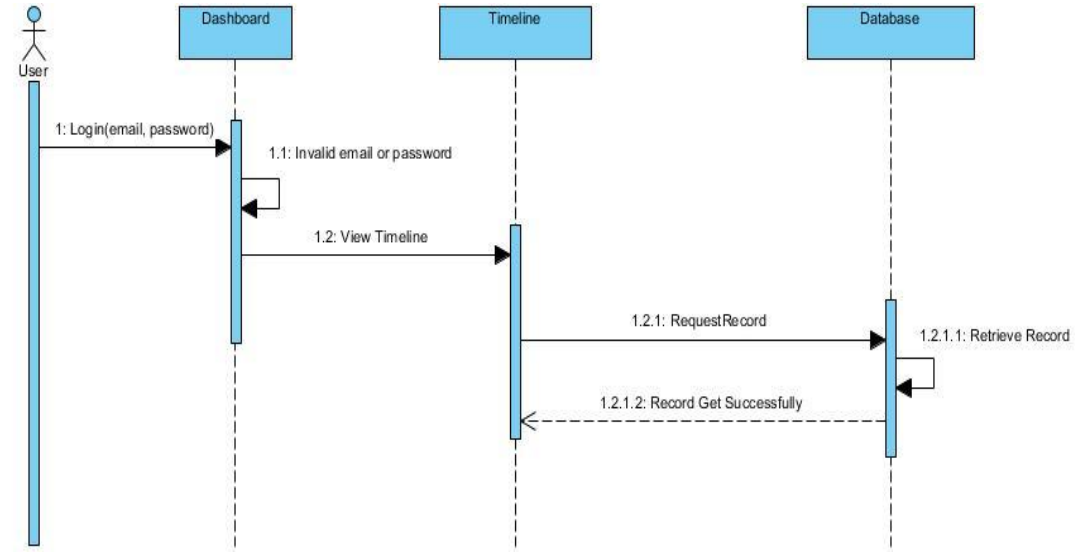

Figure 3.30: View Timeline

3.2.3.7 Edit Account

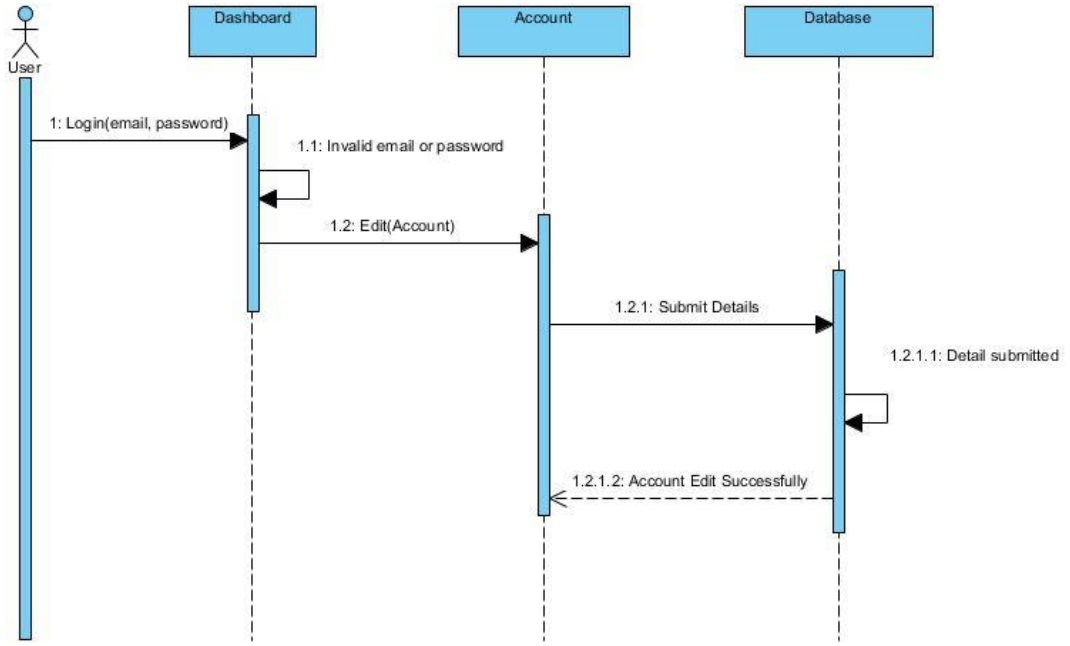

Figure 3.31: Edit Account

3.2.3.8 View Account

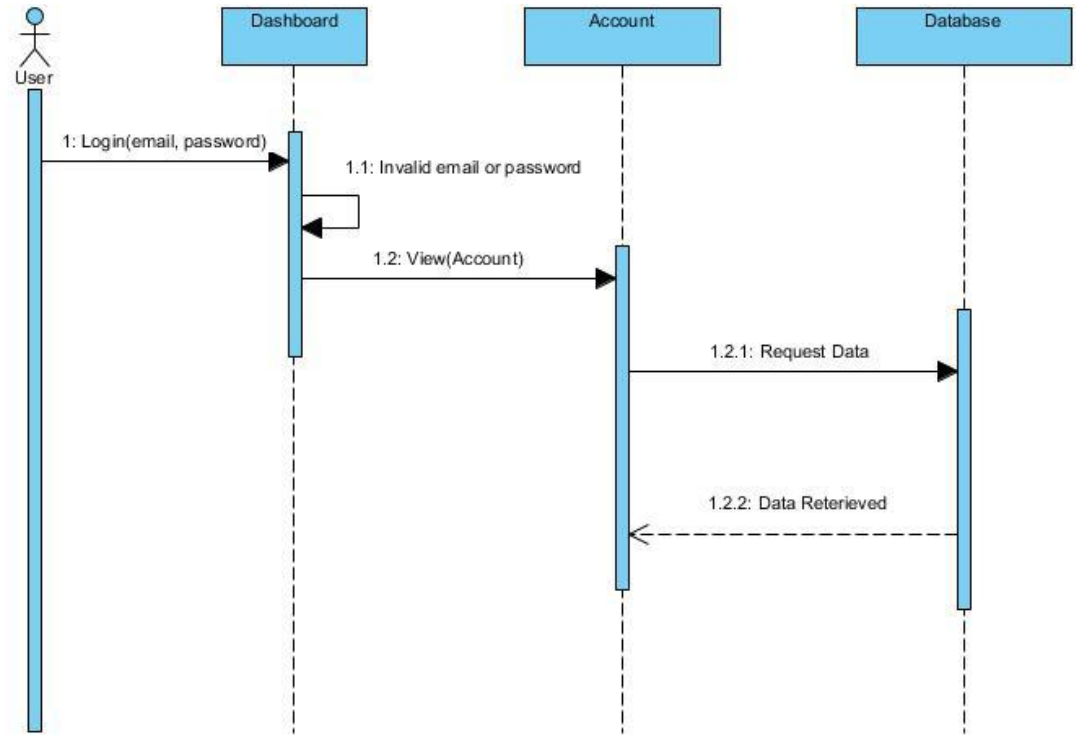

Figure 3.32: View Account

3.2.4 Foundation

3.2.4.1 Login

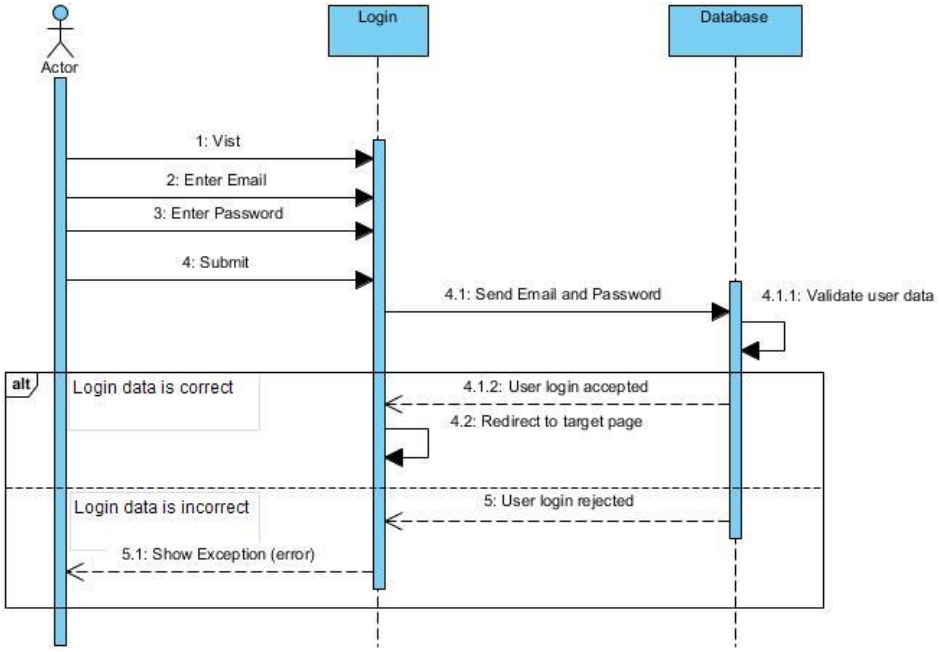

Figure 3.33: Login

3.2.4.2 Register

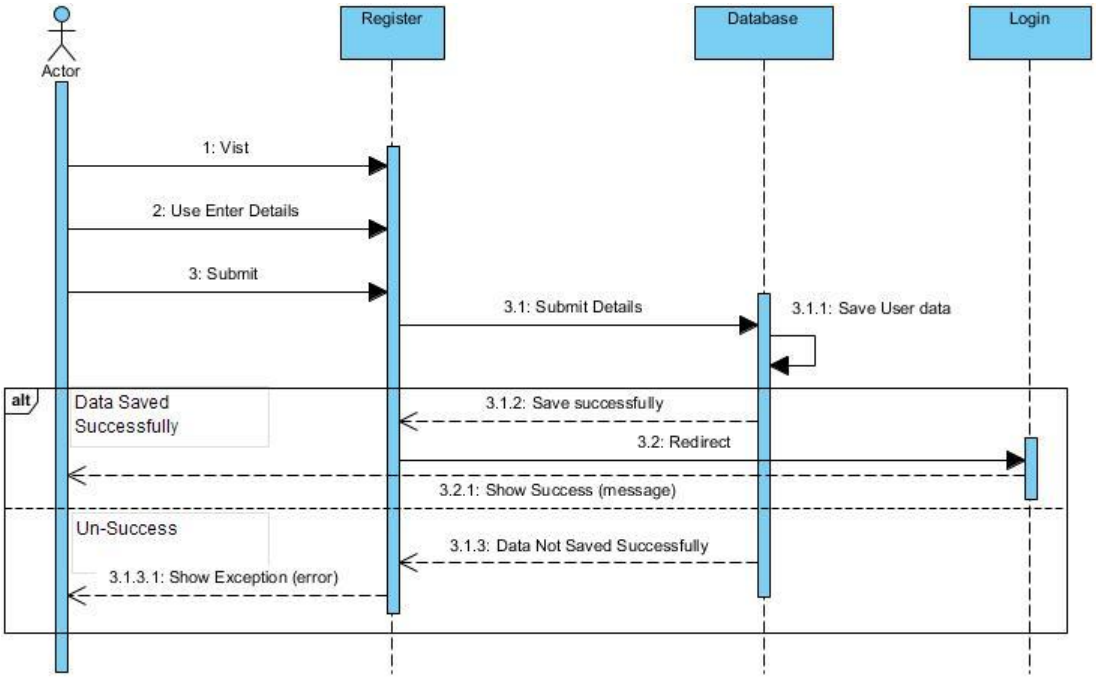

Figure 3.34: Register

3.2.4.3 View All Activities

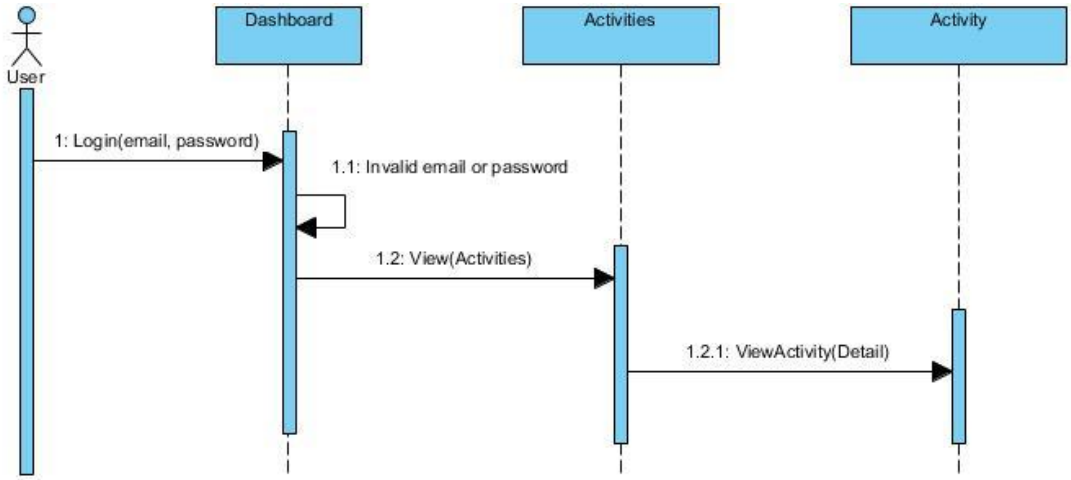

Figure 3.35: View All Activities

3.2.4.4 New Activity

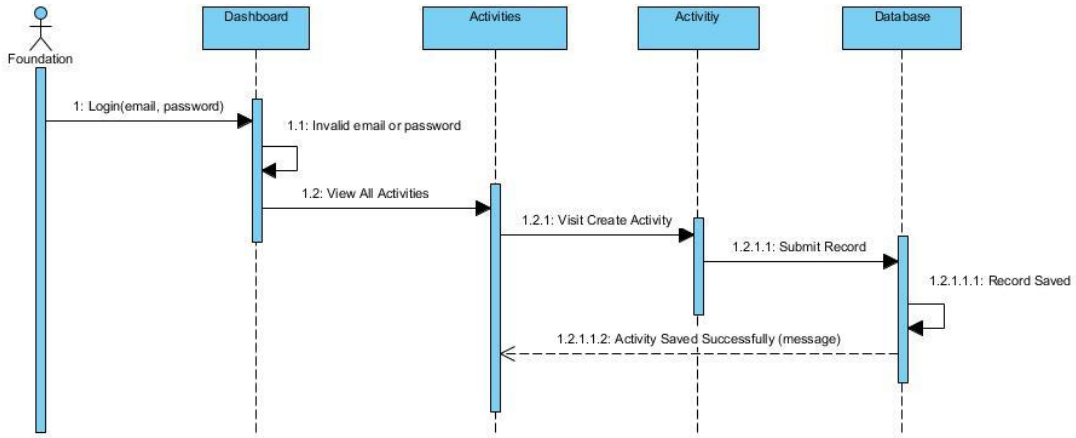

Figure 3.36: New Activity

3.2.4.5 Update Activity

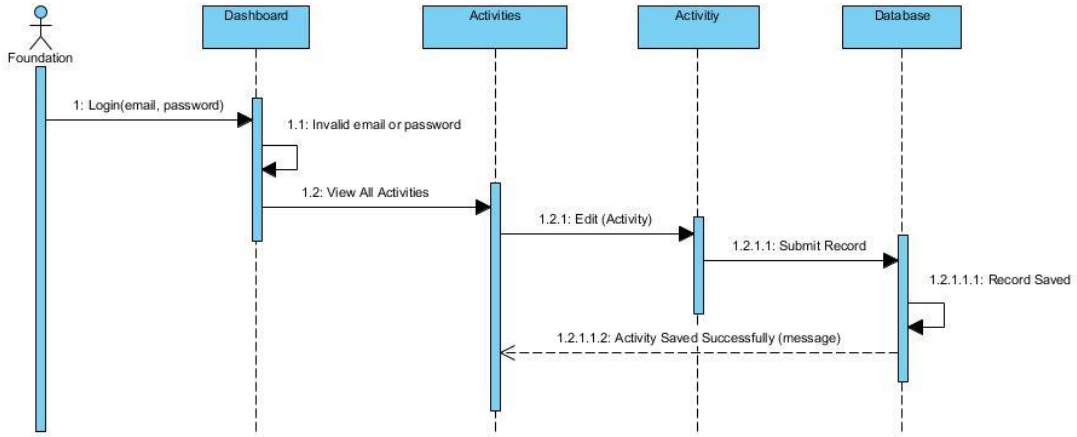

Figure 3.37: Update Activity

3.2.4.6 Delete Activity

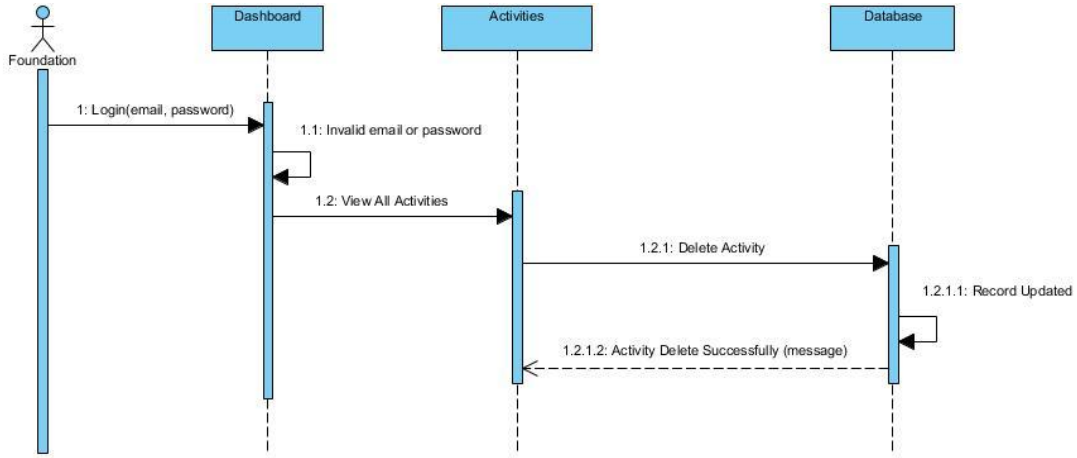

Figure 3.38: Delete Activity

3.2.4.7 Manage Attendance

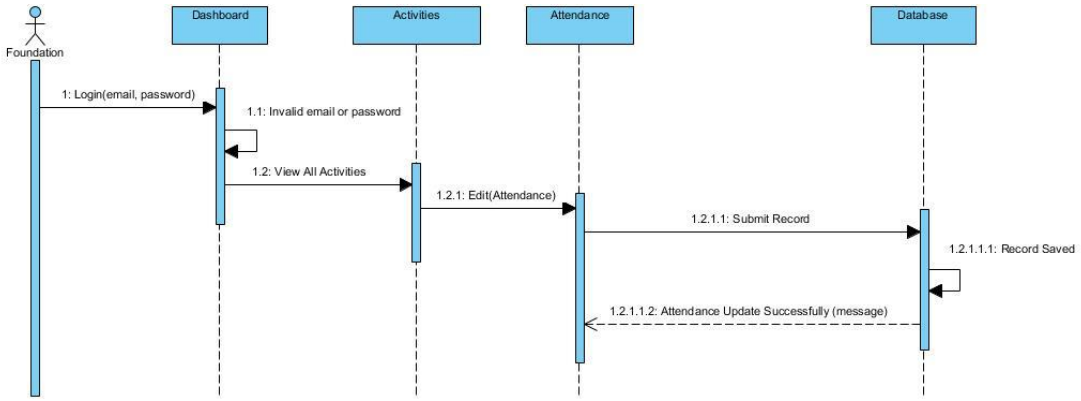

Figure 3.39: Manage Attendance

3.2.4.8 View Timeline

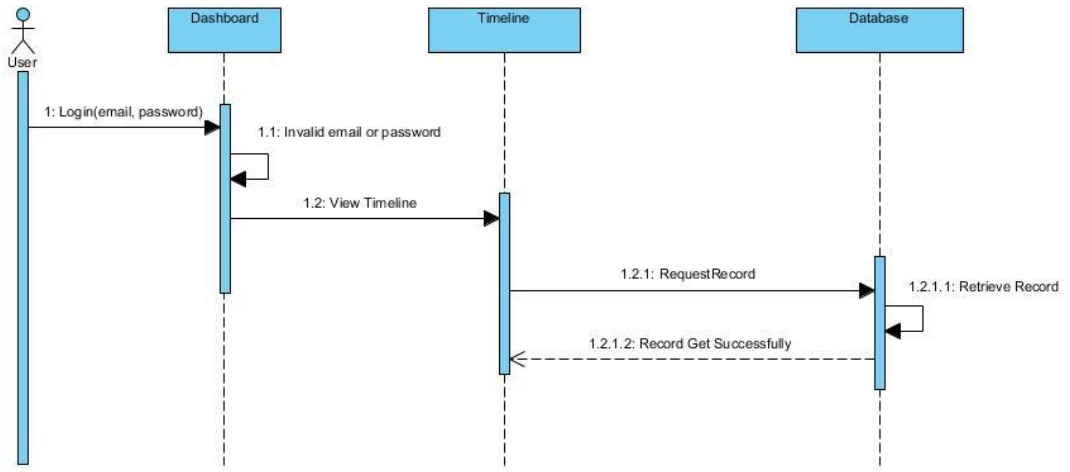

Figure 3.40: View Timeline

3.2.4.9 Edit Foundation Profile

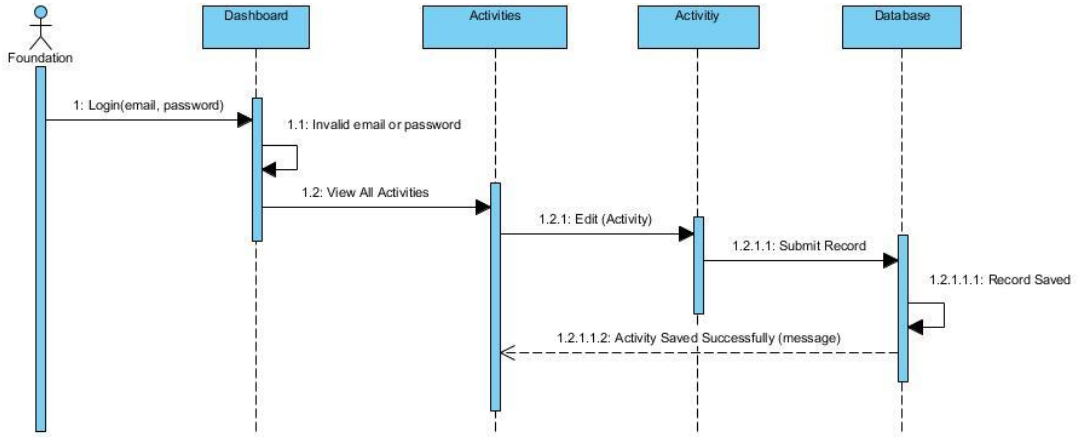

Figure 3.41: Edit Foundation Profile

3.2.4.10 View Account

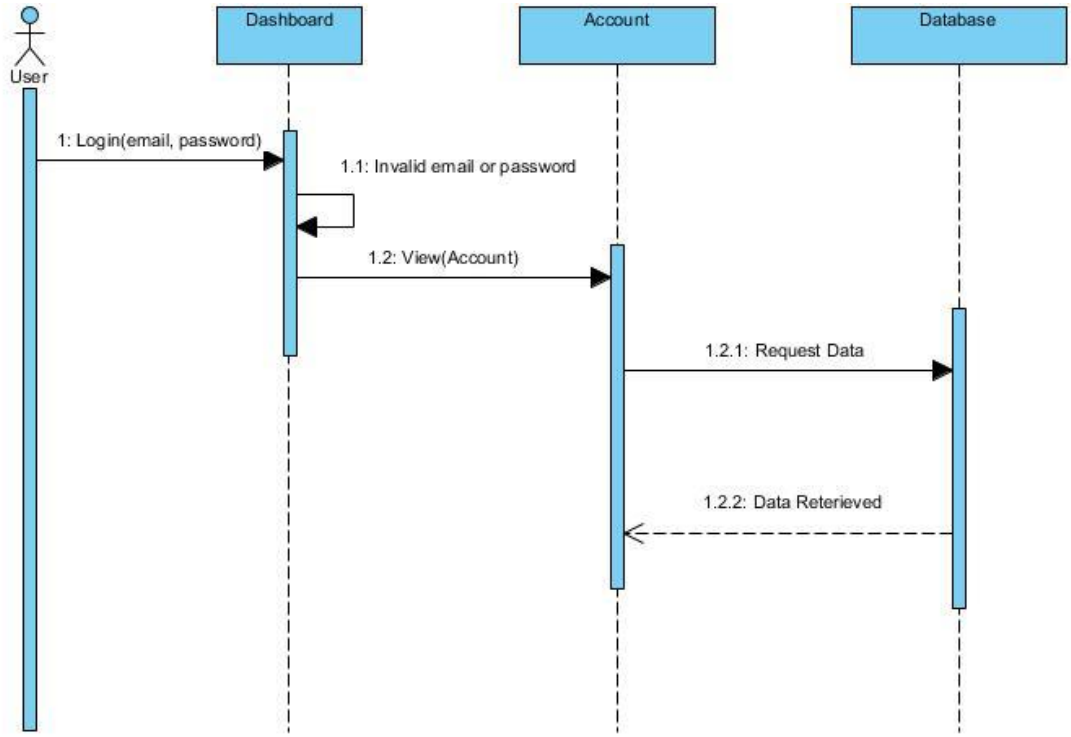

Figure 3.42: View Account

3.2.4.11 Edit Account

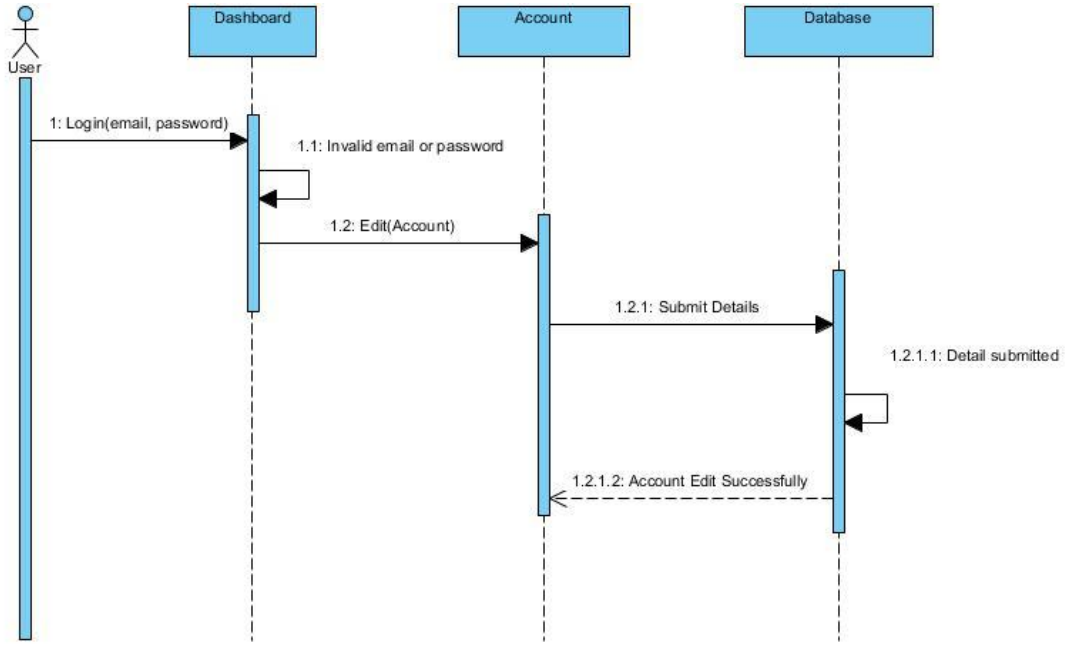

Figure 3.43: Edit Account

3.2.4.12 Get Users

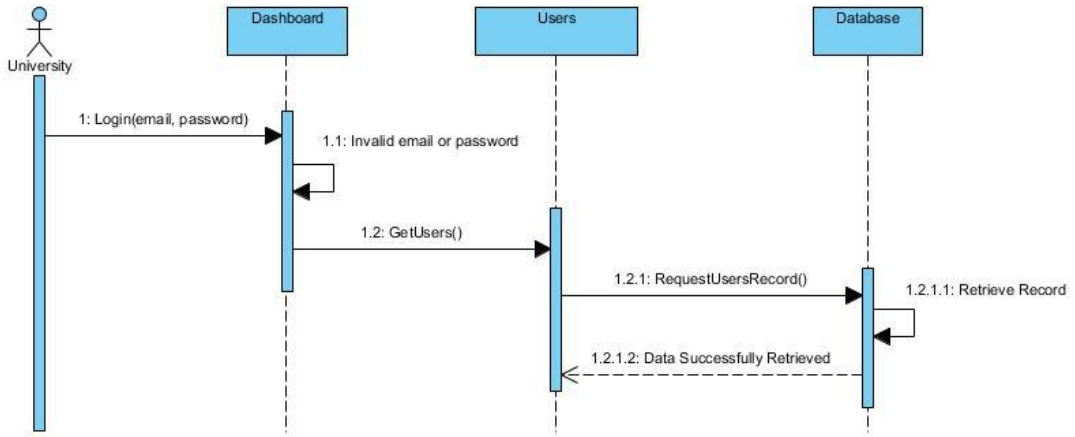

Figure 3.44: Get Users

3.2.4.13 Create User

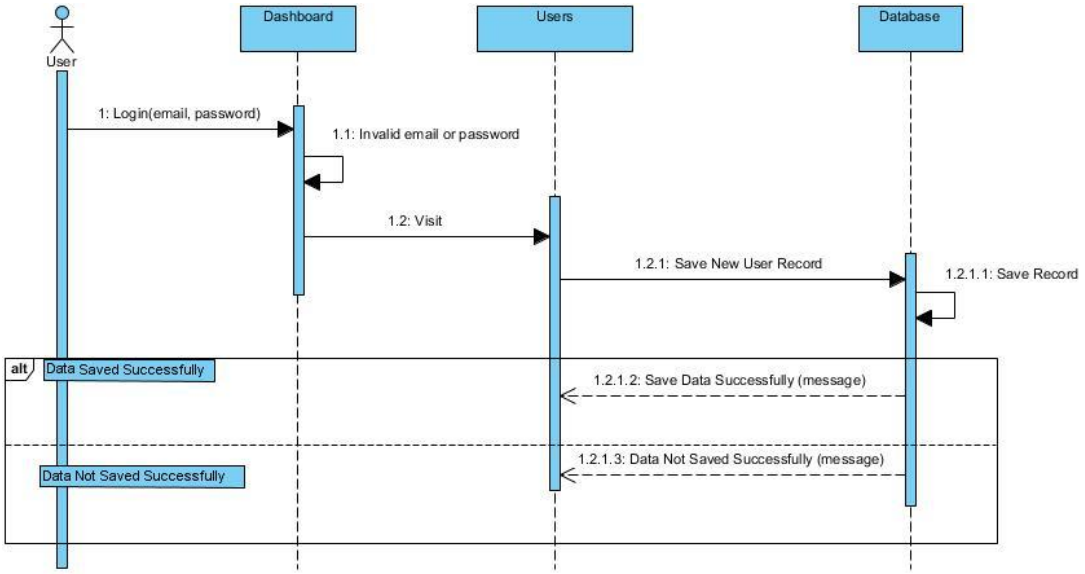

Figure 3.45: Create User

3.2.4.14 Update User

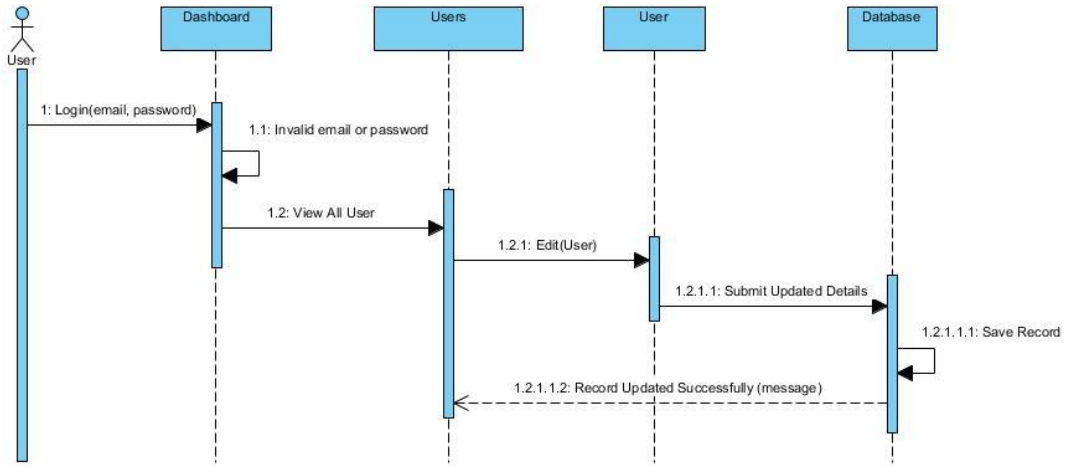

Figure 3.46: Update User

3.3 Entity-Relation Diagram

An Entity-Relationship Diagram (ERD) is a visual representation of the data structure and relationships within a project or system. It illustrates the entities (objects or concepts) involved in the project, their attributes (properties or characteristics), and the relationships between these entities.

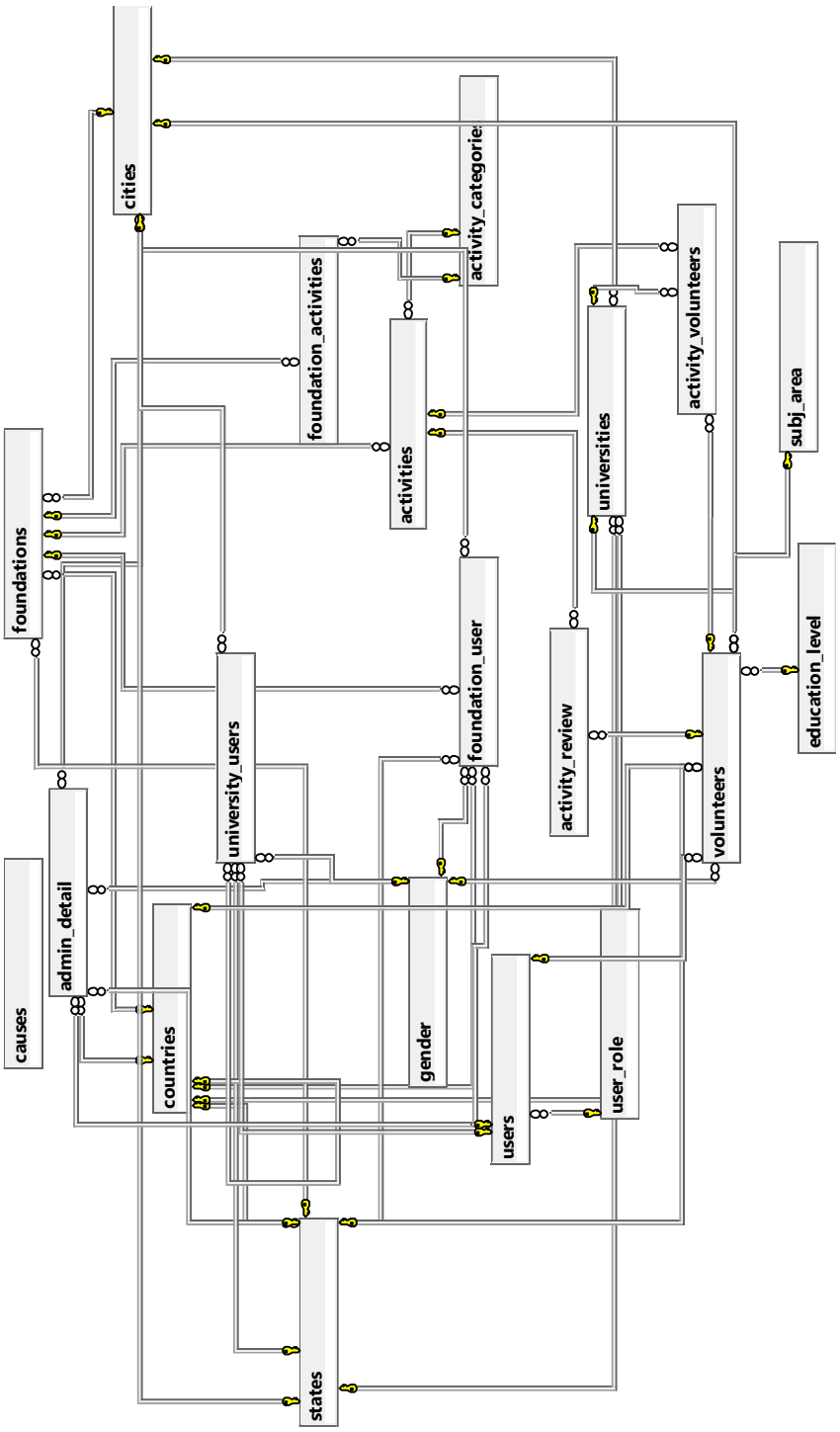

Figure 3.47: Entity-Relation Diagram

3.4 Collaboration Diagram

A collaboration diagram visually represents the interaction patterns of objects, illustrating their interconnectedness through links and the exchange of messages. This diagram helps in understanding the dynamic flow of interactions within a system. It shows the objects participating in the interaction by their links to each other and the messages that they send to each other.

3.4.1 Admin

3.4.1.1 Login

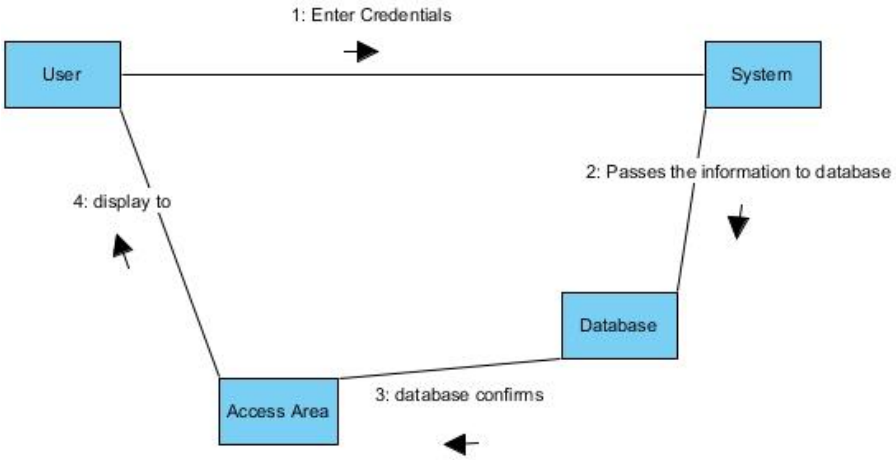

Figure 3.48: Login

3.4.1.2 View Volunteers

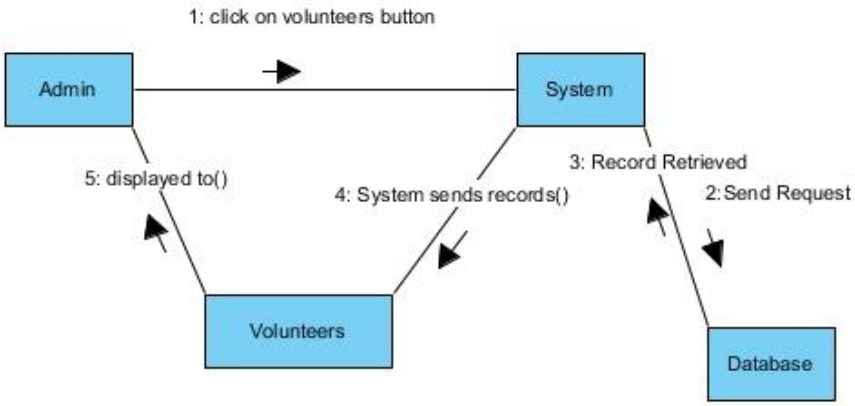

Figure 3.49: View Volunteers

3.4.1.3 View Institution

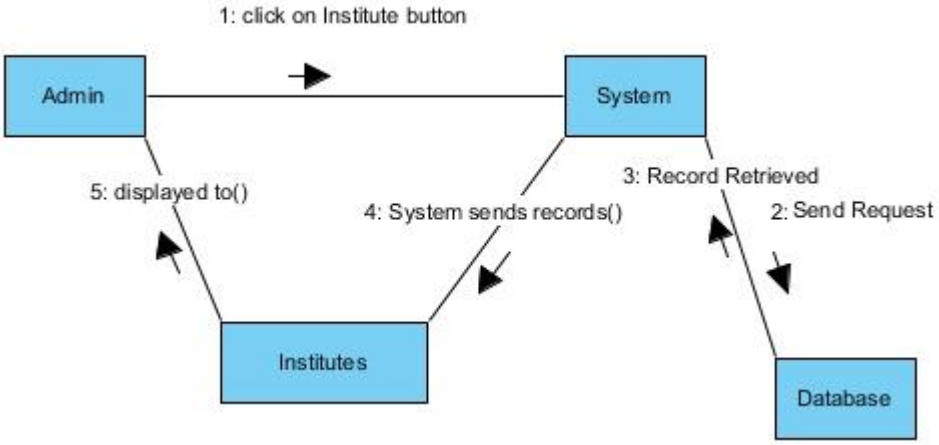

Figure 3.50: View Institutions

3.4.1.4 View Foundation

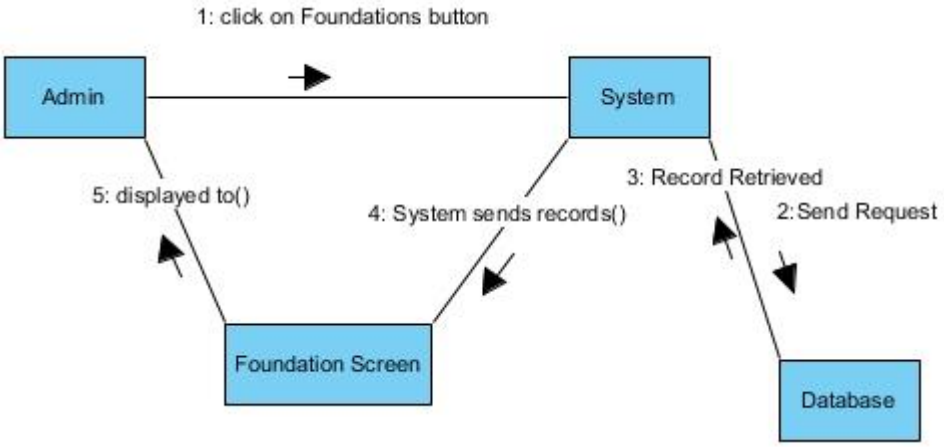

Figure 3.51: View Foundation

3.4.1.5 View Profile

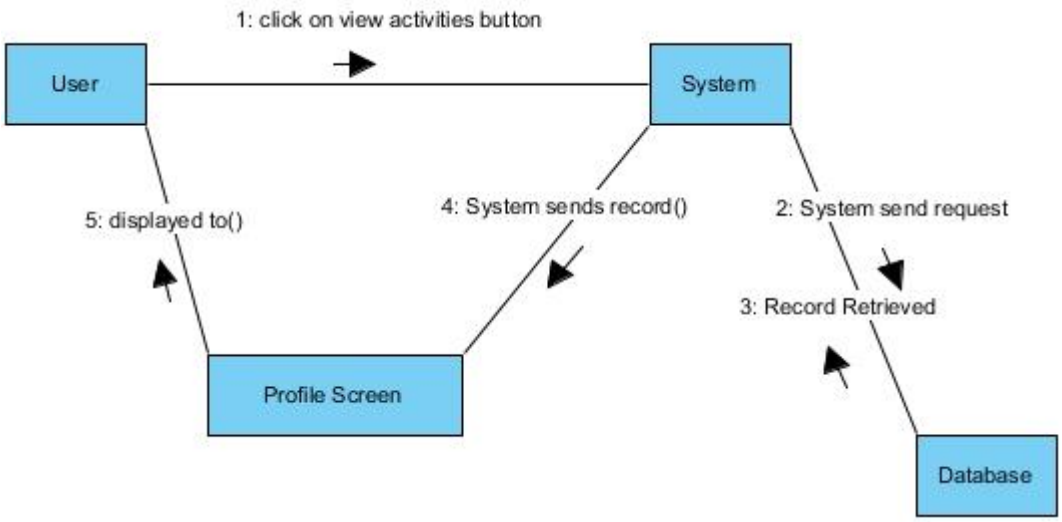

Figure 3.52: View Profile

3.4.1.6 Edit Profile

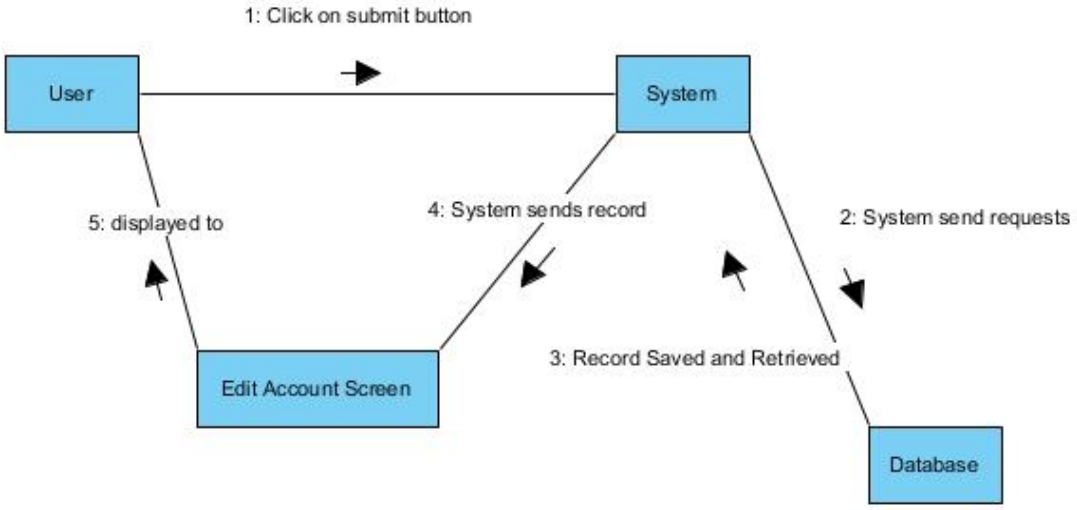

Figure 3.53: Edit Profile

3.4.1.7 View Users

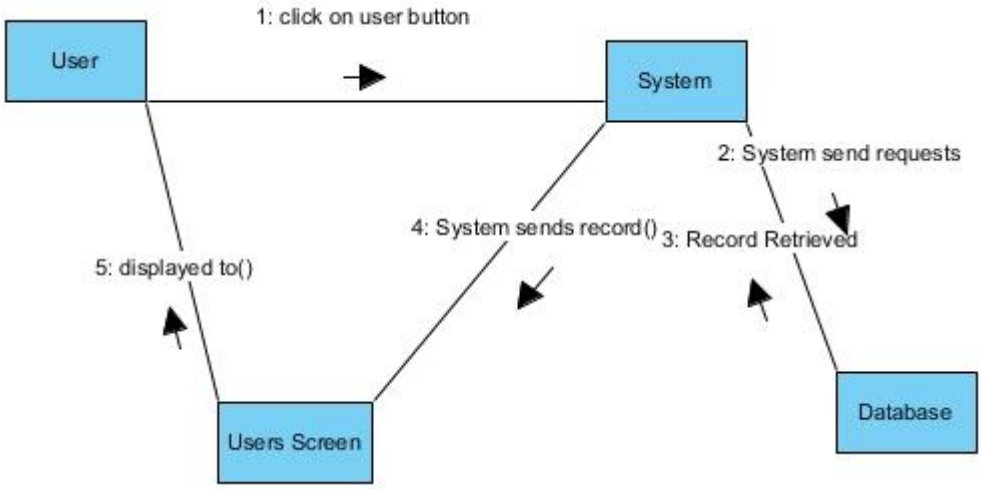

Figure 3.54: View Users

3.4.1.8 Add User

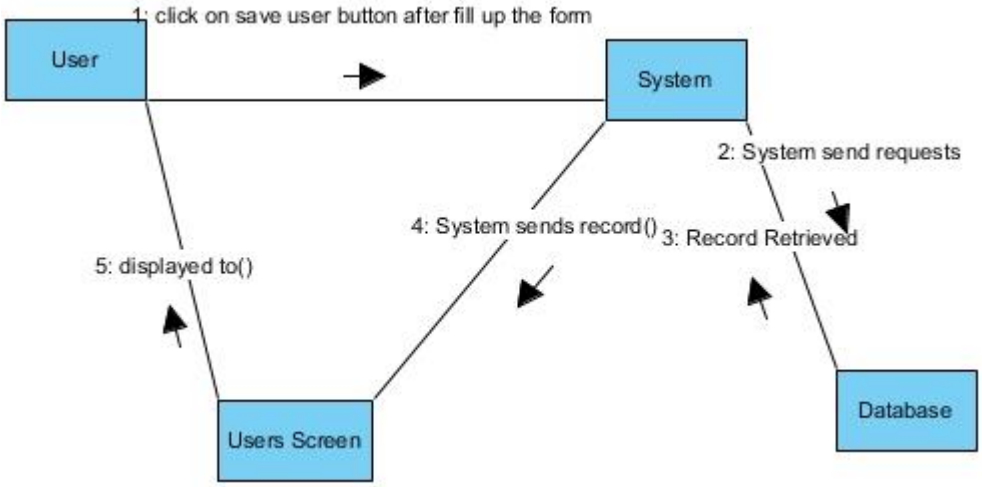

Figure 3.55: Add User

3.4.2 Volunteer

3.4.2.1 Login

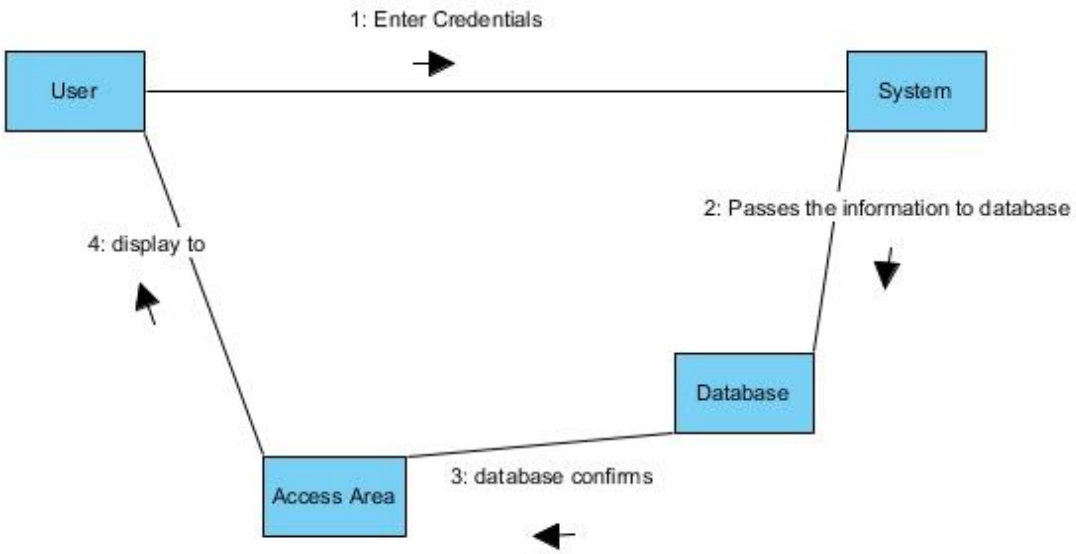

Figure 3.56: Login

3.4.2.2 Sign Up

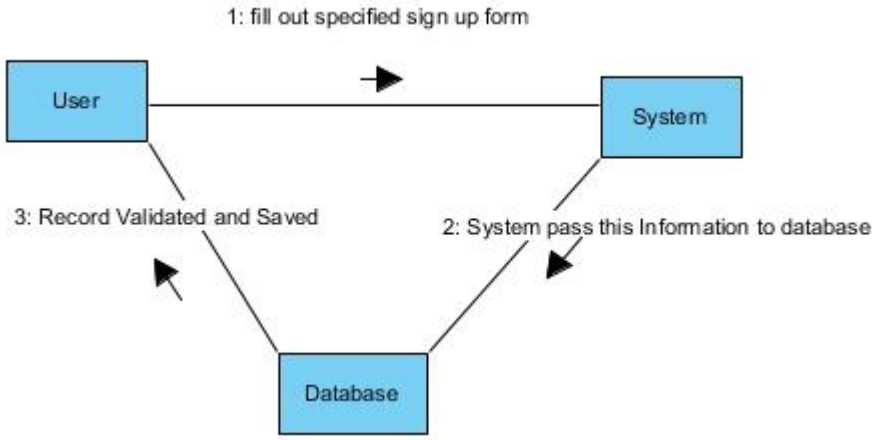

Figure 3.57: Sign Up

3.4.2.3 View Activities

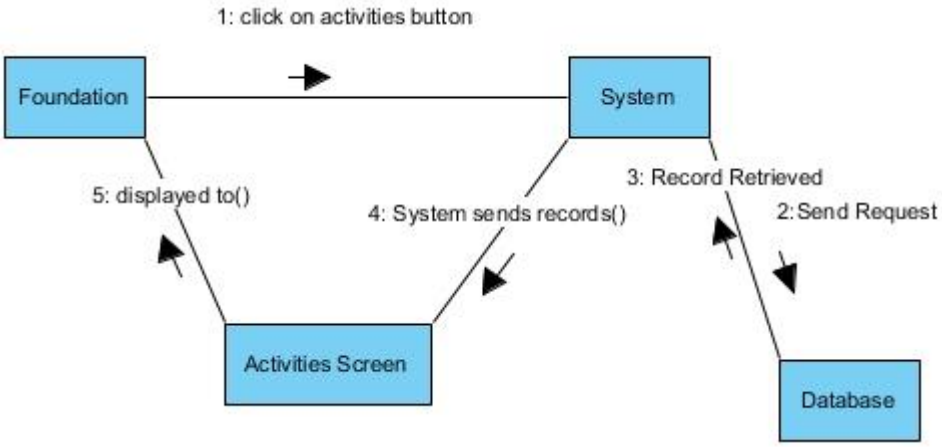

Figure 3.58: View Activities

3.4.2.4 View Recommended Activities

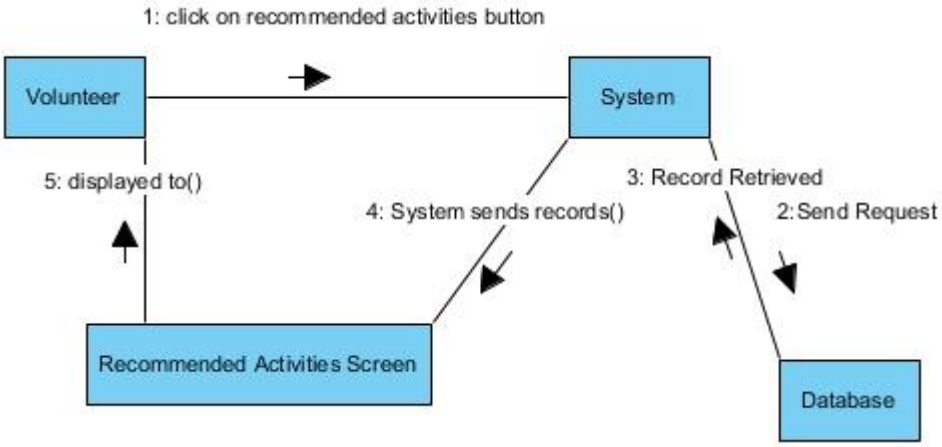

Figure 3.59: View Recommended Activities

3.4.2.5 View Registered Activities

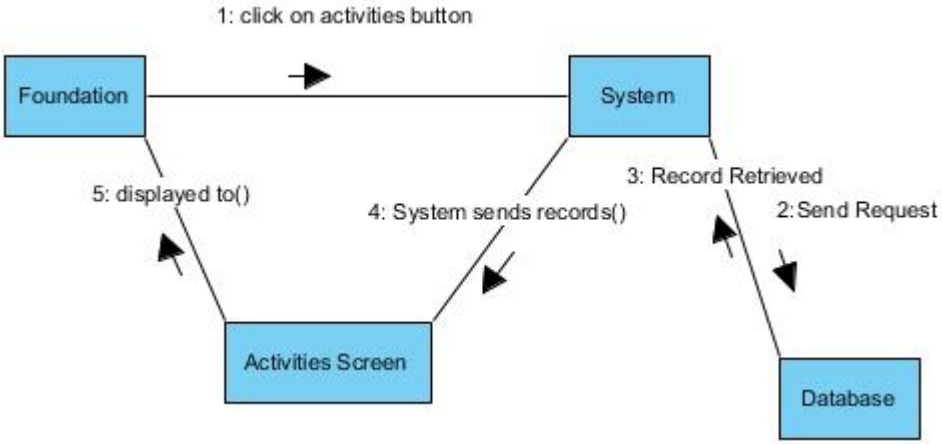

Figure 3.60: View Registered Activities

3.4.2.6 View Timeline

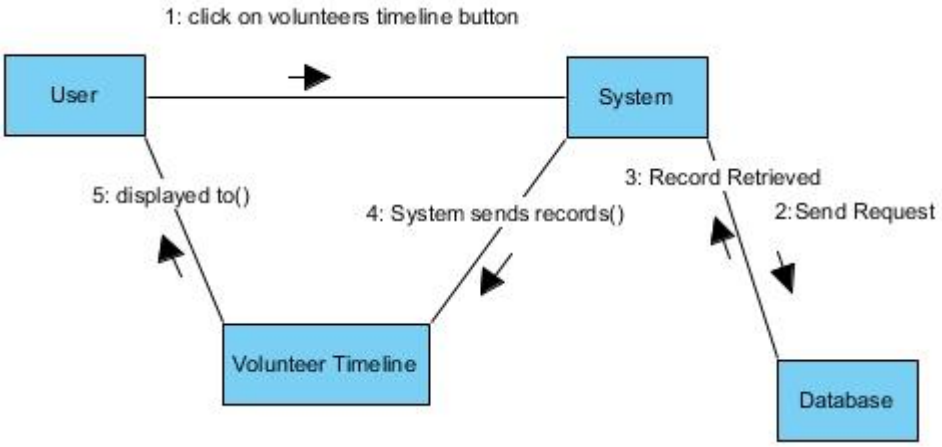

Figure 3.61: View Timeline

3.4.2.7 Generate CV

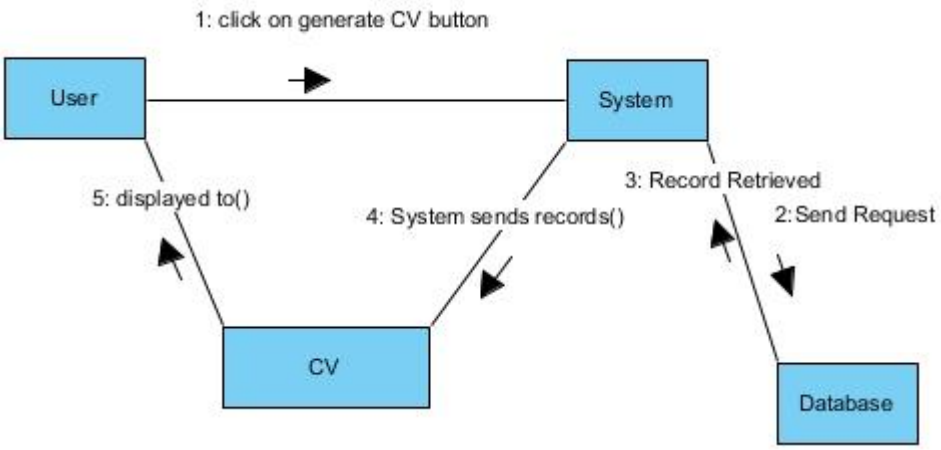

Figure 3.62: Generate CV

3.4.2.8 Register Activity

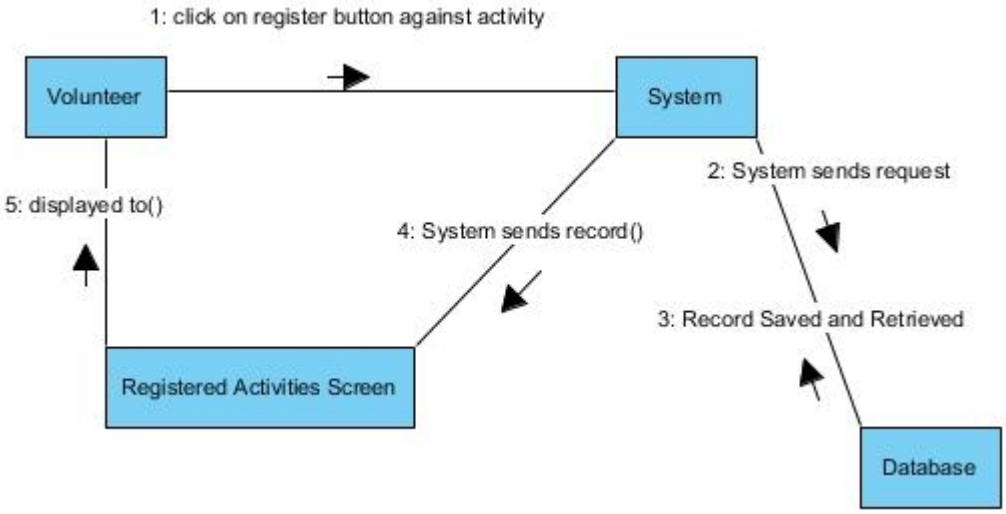

Figure 3.63: Register Activity

3.4.2.9 Edit Profile

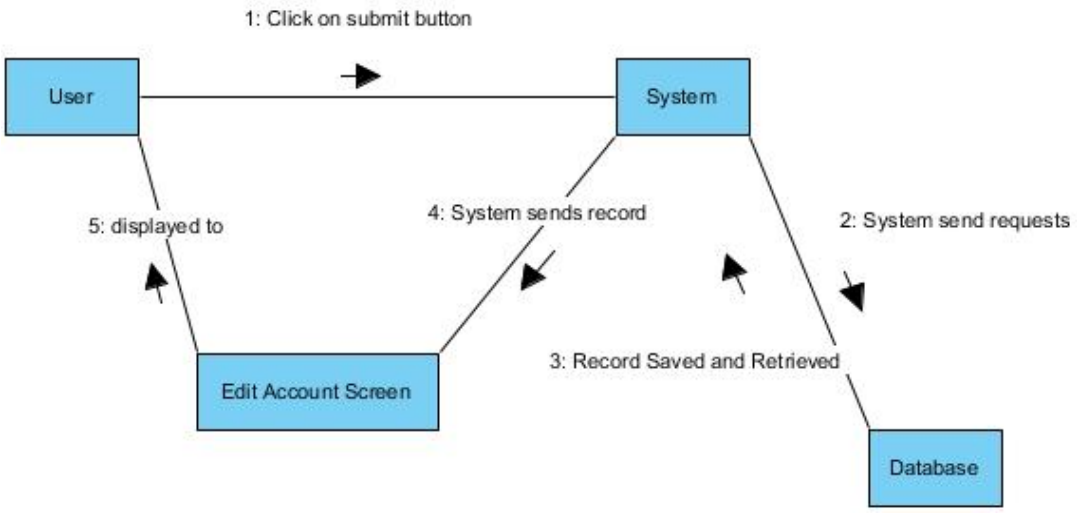

Figure 3.64: Edit Profile

3.4.2.10 View Profile

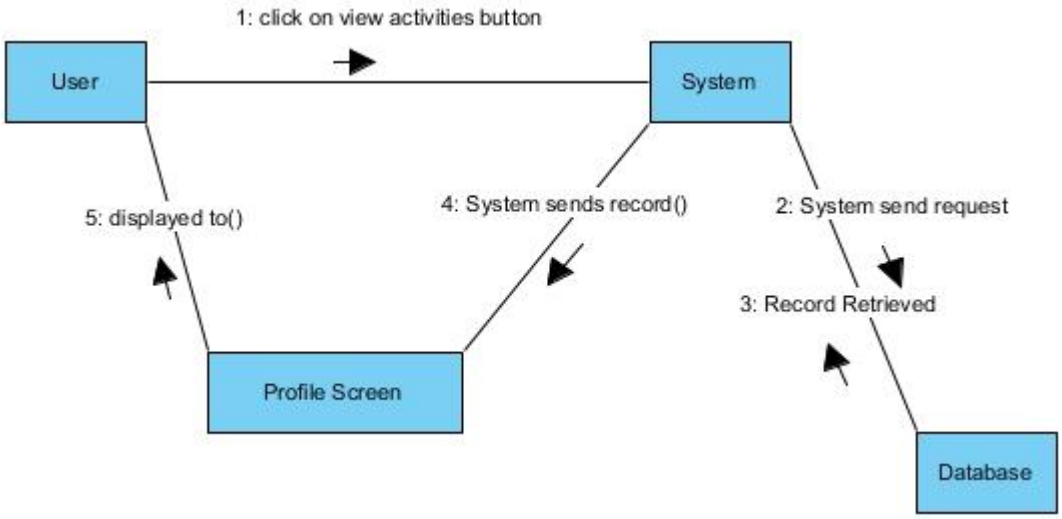

Figure 3.65: View Profile

3.4.3 Foundation

3.4.3.1 Sign Up

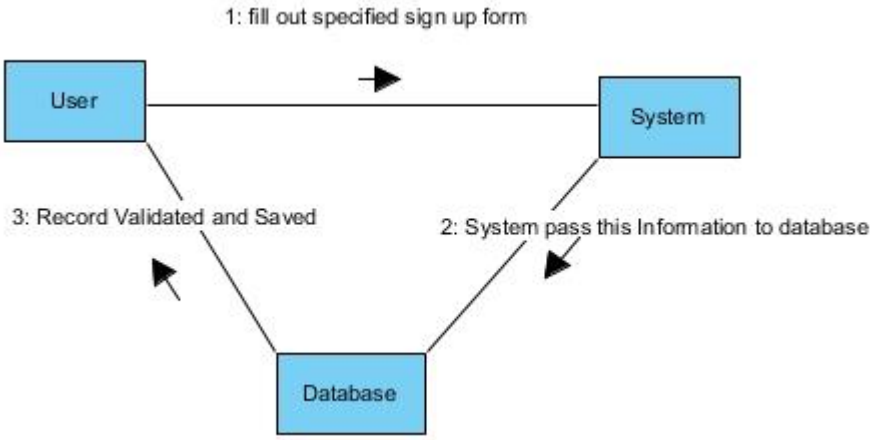

Figure 3.66: Sign Up

3.4.3.2 View Activities

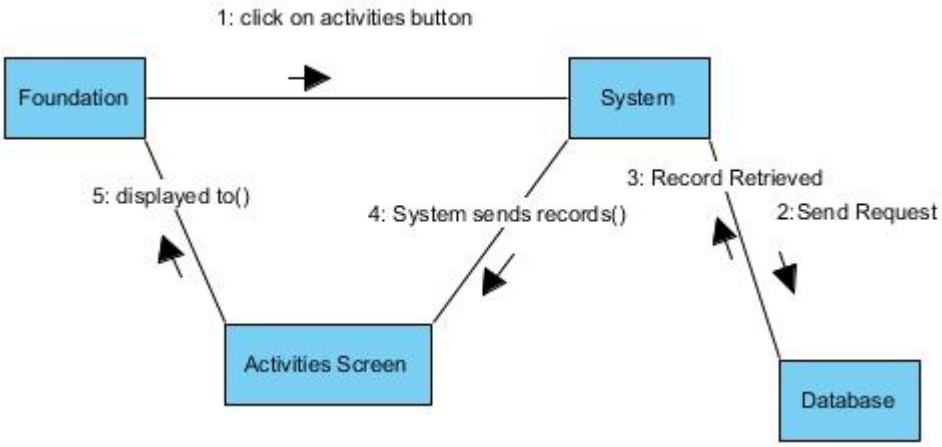

Figure 3.67: View Activities

3.4.3.3 Add Activity

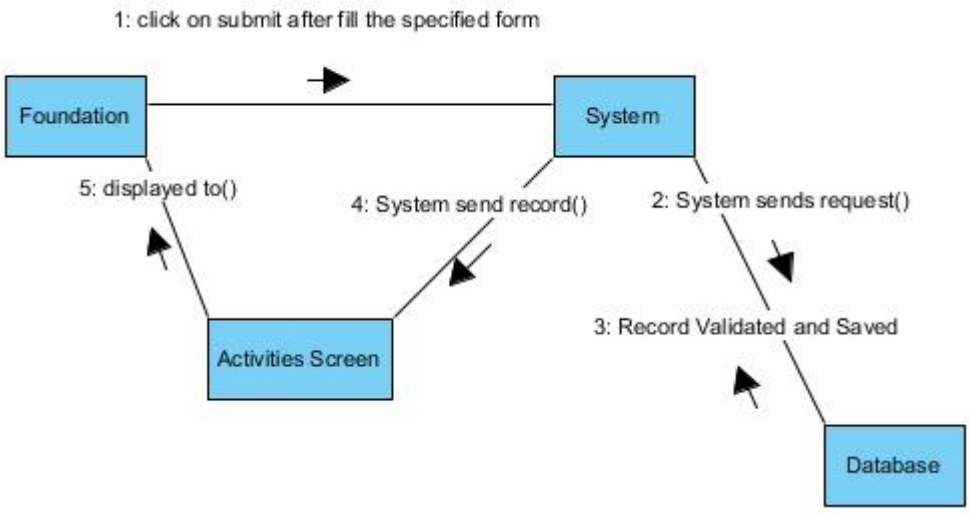

Figure 3.68: Add Activity

3.4.3.4 Manage Presence

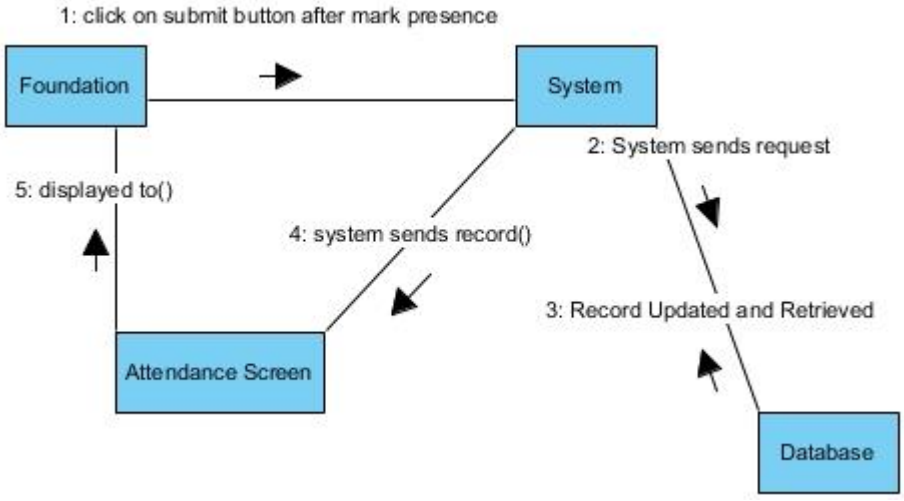

Figure 3.69: Manage Presence

3.4.3.5 View Profile

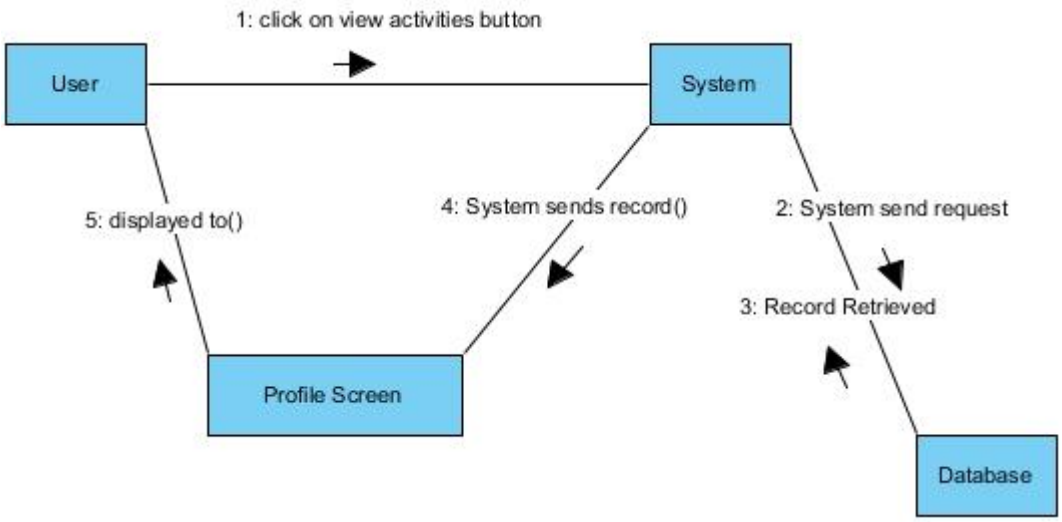

Figure 3.70: View Profile

3.4.3.6 Edit Profile

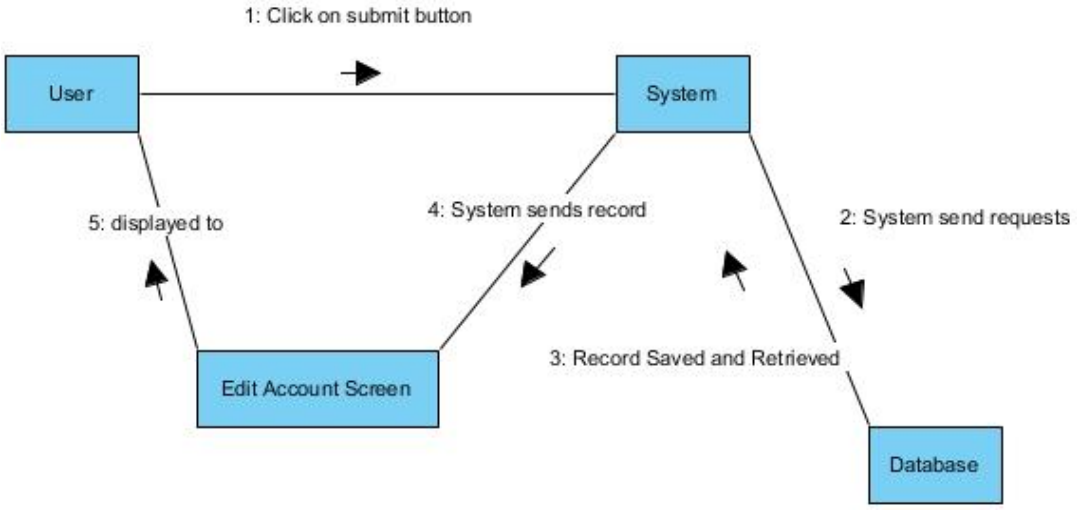

Figure 3.71: Edit Profile

3.4.4 University

3.4.4.1 Login

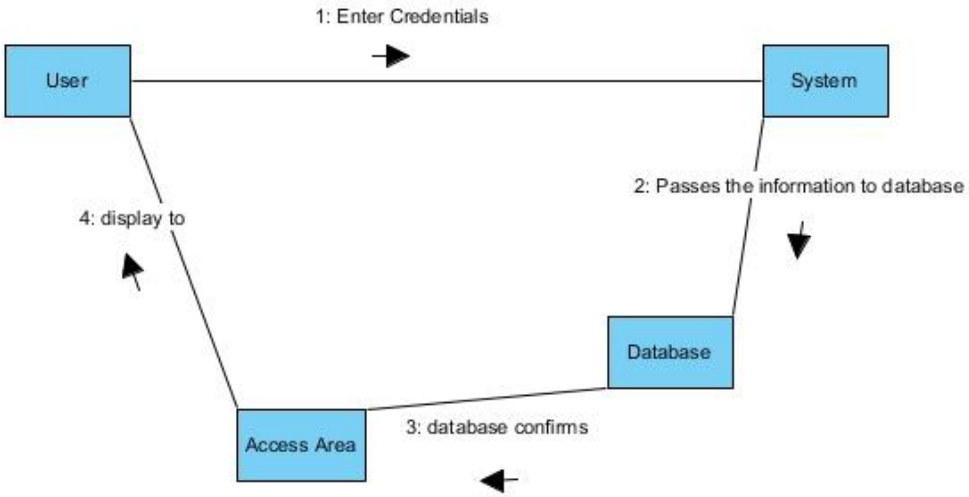

Figure 3.72: Login

3.4.4.2 Sign Up

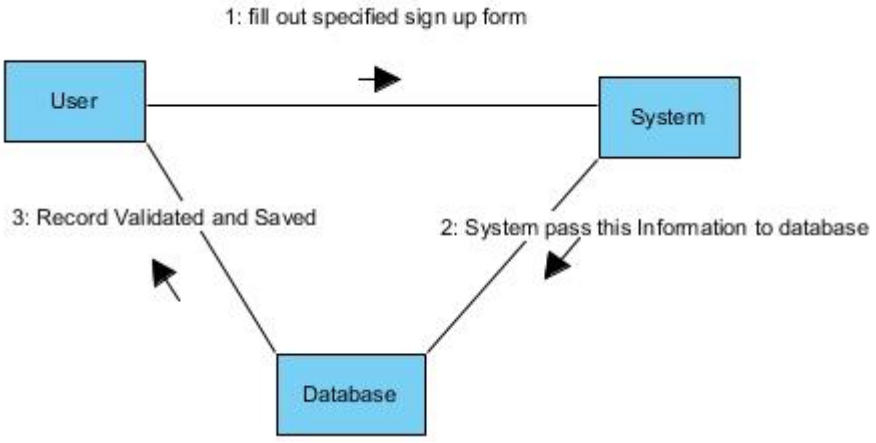

Figure 3.73: Sign Up

3.4.4.3 View Volunteer Timeline

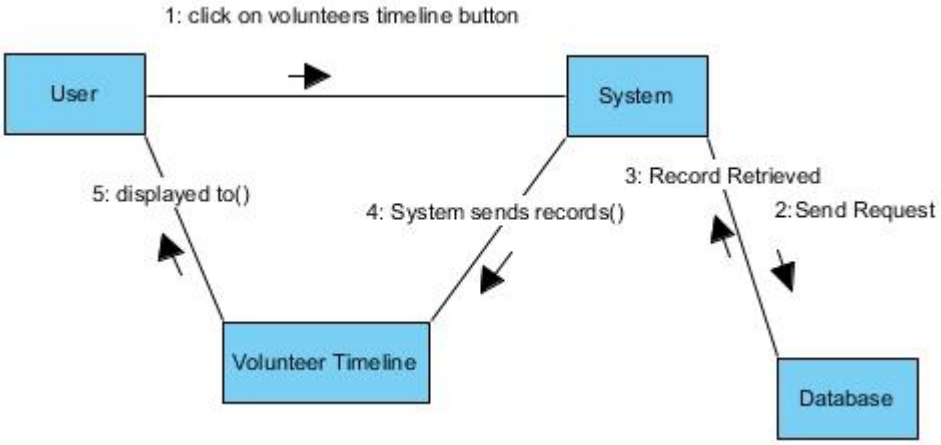

Figure 3.74: View Volunteer Timeline

3.4.4.4 Update Volunteer Status

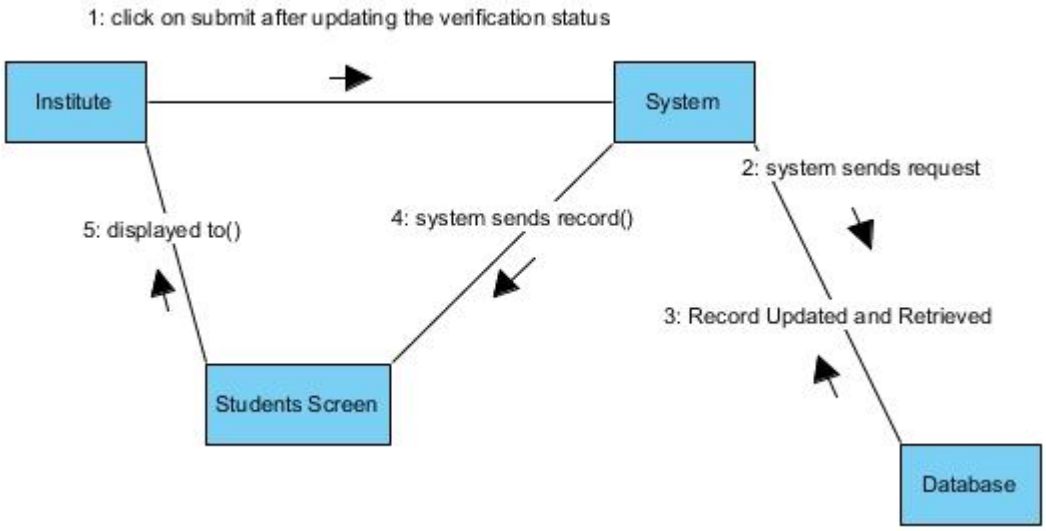

Figure 3.75: Update Volunteer Status

3.4.4.5 Edit Profile

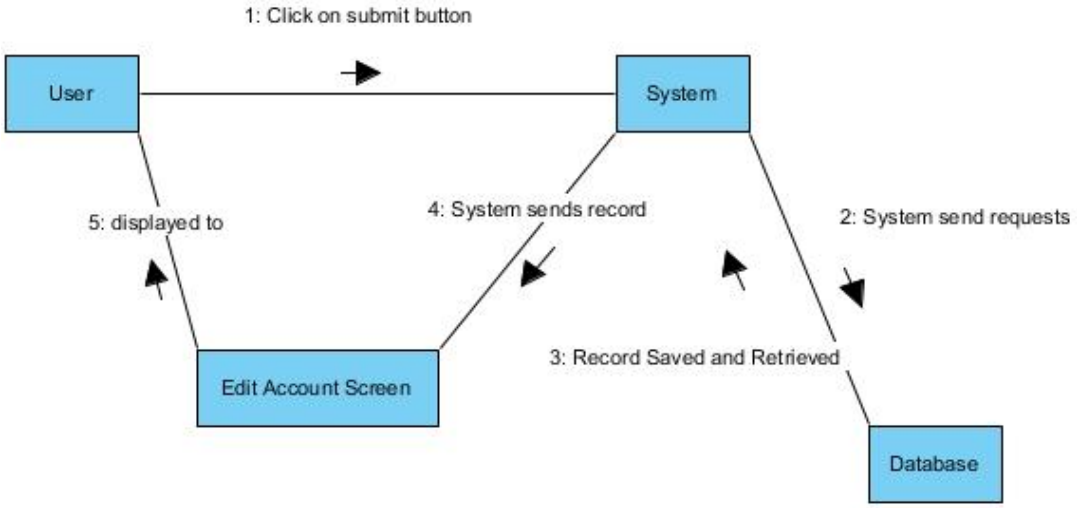

Figure 3.76: Edit Profile

3.4.4.6 View Profile

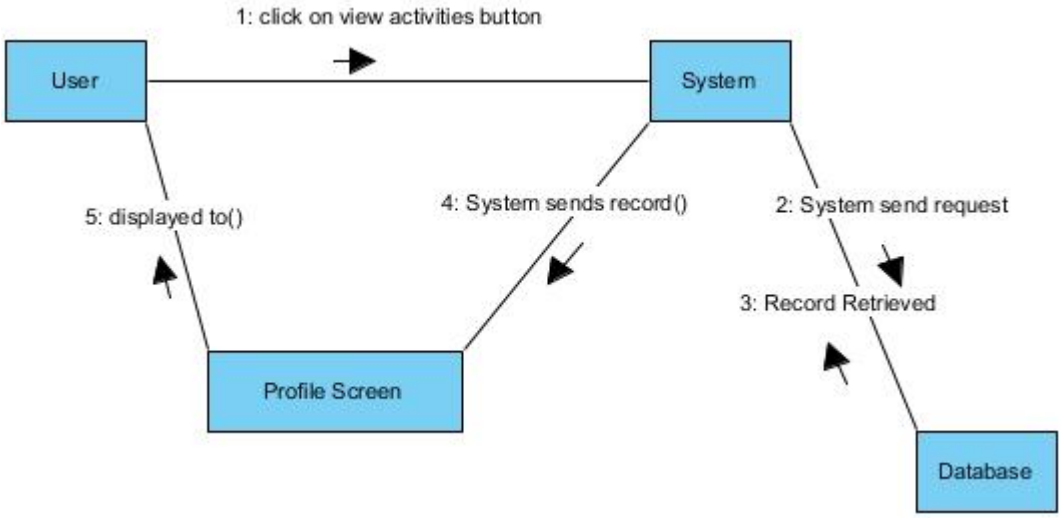

Figure 3.77: View Profile

CHAPTER 4

DATA AND IMPLMENTATION

4.1 Development Model

The software development process typically encompasses several distinct phases, including planning, design, implementation, testing, documentation, and maintenance. Each of these phases is followed in the development of the Helping TaskForce system, ensuring a comprehensive approach to its implementation.

4.1.1 Planning and designing

This initial stage of software development marks the beginning of the process. It involves gathering requirements and conducting analyses by engaging various stakeholders associated with the system. The Helping TaskForce development team ensures clear understanding and documentation of all the necessary elements needed for the system.

4.1.2 Implementation

During this phase, the system structure is translated into an executable program for the Helping TaskForce system. CodeIgniter, a PHP programming language framework, is utilized to achieve this translation. JavaScript and CSS are also employed as programming languages to aid in the development of the Helping TaskForce system.

The chosen database management system for this project is MySQL. The program itself will be written in a specific programming language. Additionally, the system incorporates a Reminder Notification (Email) feature, which is implemented using CRON Jobs to execute the designated function at predetermined times.

4.1.3 Testing

This phase entails executing the system and assessing potential defects, followed by corrective actions. The system is tested using methods such as Unit testing and End-to-End testing, which evaluate various components, including the system's interface, data storage, and the execution of stakeholders' activities. These testing approaches ensure thorough examination of the Helping TaskForce system's functionalities and identify any issues that need to be addressed.

4.1.4 Documentation

In this phase, all activities during the development process are carefully documented and recorded. This documentation serves as a valuable reference for potential upgrades, updates, and maintenance of the system. It ensures that relevant information is preserved for future use, facilitating easier troubleshooting, and referencing when necessary.

4.1.5 Maintenance

The product can be tailored to fulfil the specific needs of the user and be deployed for use. Any newly identified issues can be resolved, and any missed requirements can be incorporated into the system.

4.2 Test Cases

Following are the test case that are used to test the Helping TaskForce system’s functionalities.

Table 4.1: Login with Valid Credentials – Test Case

Test Case ID	TC1
Test Case Name	Login with Valid Credentials
Test Steps	1. Open the web client (browser). 2. Enter the URL of the web application. 3. Verify that the login page is displayed. 4. Enter a valid email address in the email field. 5. Enter a valid password in the password field. 6. Click on the "Sign In" button. 7. Verify that the system verifies the provided credentials. 8. Verify that the user is redirected to their respective dashboard. 9. Verify that the dashboard page is displayed. 10. Verify that the user has successfully logged in.
Expected Result:	• The system should successfully verify the provided credentials. • The user should be redirected to their respective dashboard. • The dashboard page should be displayed. • The user should be successfully logged in.

Table 4.2: Login with Invalid Credentials – Test Case

Test Case ID	TC2
Test Case Name	Login with Invalid Credentials
Test Steps	1. Open the web client (browser). 2. Enter the URL of the web application. 3. Verify that the login page is displayed. 4. Enter an invalid email address or username in the email field. 5. Enter an invalid password in the password field. 6. Click on the "Sign In" button. 7. Verify that the system verifies the provided credentials. 8. Verify that the system displays an error message indicating invalid credentials. 9. Verify that the user is redirected back to the login page.
Expected Result:	• The system should display an error message indicating invalid credentials. • The user should be redirected back to the login page.

Table 4.3: Add User with Unique Email – Test Case

Test Case ID	TC3
Test Case Name	Add User with Unique Email
Test Steps	1. Login to the web application as an Admin, Foundation, or University main user. 2. Navigate to the "Manage Users" tab. 3. Click on the "Add User" button. 4. Fill in the required fields in the user creation form, including a unique email address. 5. Click on the "Create User" button. 6. Verify that the system successfully creates the user. 7. Verify that a success message is displayed to the Admin, Foundation, or University main user.
Expected Result:	• The system should successfully create the user with the provided details. • A success message should be displayed to the Admin, Foundation, or University main user.

Table 4.4: Generate and Print CV – Test Case

Test Case ID	TC4
Test Case Name	Generate and Print CV
Test Steps	1. Login to the web application as a Volunteer. 2. Navigate to the "Generate CV" tab. 3. Verify that the Generate CV page is displayed. 4. Fill in the required fields and provide necessary information to generate the CV. 5. Click on the "Generate" button. 6. Verify that the system generates the CV based on the provided information. 7. Scroll down to the bottom of the page. 8. Verify that the print button is displayed. 9. Click on the print button. 10. Verify that the system initiates the print dialog for the CV.
Expected Result:	• The system should successfully generate the CV based on the provided information. • The print button should be displayed at the bottom of the page. • Clicking on the print button should initiate the print dialog for the CV.

Table 4.5: Add User with Duplicate Email – Test Case

Test Case ID	TC5
Test Case Name	Add User with Duplicate Email
Test Steps	1. Login to the web application as an Admin, Foundation, or University main user. 2. Navigate to the "Manage Users" tab. 3. Click on the "Add User" button. 4. Fill in the required fields in the user creation form, including an email address that already exists in the system. 5. Click on the "Create User" button. 6. Verify that the system displays an error message indicating a duplicate email. 7. Verify that the user is not created.
Expected Result:	• The system should display an error message indicating a duplicate email. • The user should not be created.

Table 4.6: Receive Reminder Notification Email – Test Case

Test Case ID	TC6
Test Case Name	Receive Reminder Notification Email
Test Steps	1. Ensure that the system is active and running. 2. Ensure that the CRON Job for sending reminder notifications is properly set up in the system. 3. Verify that there are registered activities for volunteers scheduled for tomorrow. 4. Wait for the specified time when the system is set to send reminder notifications. 5. Check the email inbox of the volunteers who have registered activities for tomorrow. 6. Verify that the volunteers receive the reminder notification email. 7. Open the email and verify that it contains the necessary details about the scheduled activity.
Expected Result:	• The system should automatically send reminder notification emails to the volunteers whose registered activities are scheduled for tomorrow. • The volunteers should receive the reminder notification email. • The email should contain relevant details about the scheduled activity.

Table 4.7: View Volunteer Timeline – Test Case

Test Case ID	TC7
Test Case Name	View Volunteer Timeline
Test Steps	1. Login to the web application as an Admin, Volunteer, Foundation, or University user. 2. Based on the user role: a. For Admin: Navigate to the "Volunteer" tab and locate the volunteer for whom the timeline needs to be viewed. Click on the "View" icon next to the volunteer. b. For Volunteer: Click on the "Timeline" button or link. c. For Foundation: Navigate to the "Manage Activities" tab, then click on "Manage Volunteer" and locate the volunteer for whom the timeline needs to be viewed. Click on the "View" button next to the volunteer. d. For University: Go to the "Student" tab and locate the student volunteer. Click on the "View" icon next to the student. 3. Verify that the volunteer's timeline is displayed, showing their activities, events, or any relevant information. 4. Review the timeline and ensure that it contains accurate and up-to-date information. 5. Close the volunteer timeline view.
Expected Result:	• The system should successfully display the volunteer's timeline. • The timeline should contain accurate and up-to-date information related to the volunteer's activities.

Table 4.8: Manage Volunteer Hours – Test Case

Test Case ID	TC8
Test Case Name	Manage Volunteer Hours
Test Steps	1. Login to the web application as a Foundation. 2. Navigate to the "Activities" tab. 3. Locate the activity for which volunteer hours need to be managed. 4. Click on the select box or dropdown menu next to the activity. 5. From the submenu, click on the "Manage Attendance" option.

	6. Verify that the attendance management page for the selected activity is displayed. 7. Locate the volunteer(s) in the attendance list. 8. Mark the volunteer(s) as present or record their attendance based on the provided options. 9. Verify that the volunteer CSP hours are assigned to the respective volunteer(s). 10. Optionally, repeat steps 4-9 for other activities if necessary.
Expected Result:	• The system should successfully display the attendance management page for the selected activity. • The user should be able to mark the volunteer(s) as present or record their attendance. • The volunteer CSP hours should be assigned to the respective volunteer(s) based on their attendance

Table 4.9: Track Volunteer Hours (CSP) – Test Case

Test Case ID	TC9
Test Case Name	Track Volunteer Hours (CSP)
Test Steps	1. Login to the web application as a university user. 2. Navigate to the "Student" tab. 3. Verify that the student list page is displayed. 4. Identify the specific student for whom the volunteer hours (CSP) need to be tracked. 5. Use the search field to locate the student in the student list. 6. Verify that the student's record is displayed, showing their relevant details. 7. Take note of the CSP hours tracked for the student. 8. Optionally, review additional details about the student's activities or performance if available. 9. Close the student's record view.
Expected Result:	• The system should successfully display the student list page. • The student's record should be displayed, showing their relevant details. • The tracked volunteer hours (CSP) should be accurately displayed for the student. • Additional details about the student's activities or performance, if available, should be accessible.

Table 4.10: Register Activity – Test Case

Test Case ID	TC10
Test Case Name	Register Activity
Test Steps	1. Login to the web application as a Volunteer. 2. Navigate to the "Activities" tab. 3. Verify that the activities list page is displayed. 4. Review the available activities and locate the activity that the volunteer wants to register for. 5. Click on the "View" button next to the desired activity. 6. Verify that the activity detail page is displayed. 7. Click on the "Register" button to register for the activity. 8. Verify that the system successfully registers the volunteer for the selected activity. 9. Optionally, verify that the volunteer's registration details (date, time, activity, etc.) are accurately recorded. 10. Close the activity detail page.
Expected Result:	• The system should successfully display the activities list page. • The volunteer should be able to locate the desired activity and view its details. • Clicking on the "Register" button should register the volunteer for the selected activity. • The system should accurately record the volunteer's registration details.

4.3 Technical Aspect of Helping TaskForce

As an embedded system, the Helping TaskForce requires the integration of a web server, a database management system, and a web application. The system comprises a front-end component and a back-end component, which can be further classified into the following categories:

4.3.1 Hardware

Hardware contains necessary hardware such as Personal Computer (PC), Android or iOS mobiles.

4.3.2 Front-end

Front-end involves HTML, CSS, Bootstrap 5.0, JavaScript.

4.3.3 Back-end

Back-end involves CodeIgniter, MySQL, and PHP 8.1.

CHAPTER 5

USER MANUAL

5.1 Helping TaskForce

5.1.1 Landing

Landing pages are web specifically designed and optimized pages. They serve as entry points for users who arrive at your website through various channels.

5.1.1.1 Home

When the user opens the system, it saw the home screen where some quick links are given as shown in below figure.

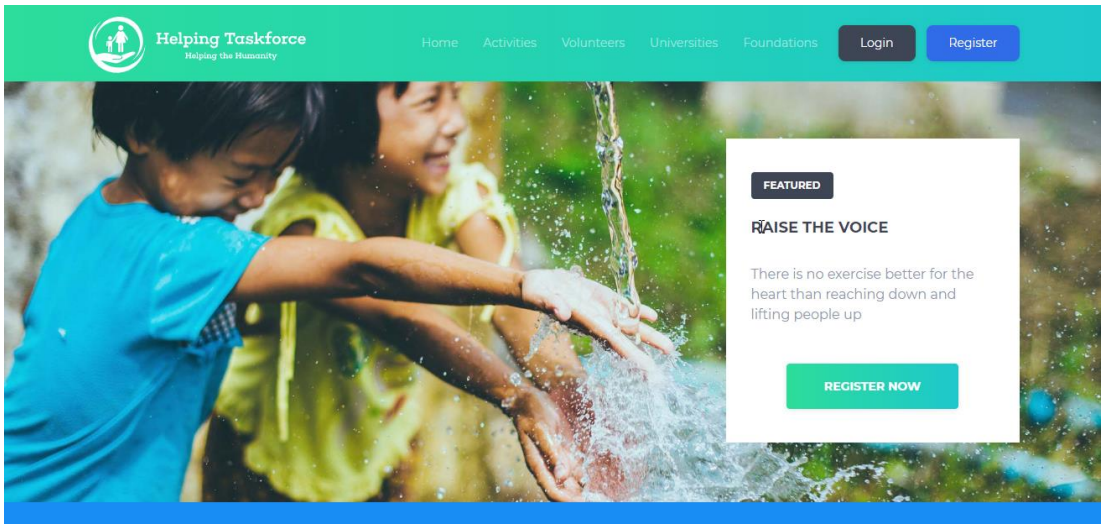

Figure 5.1: Home Page

5.1.1.2 Volunteer

Click on the Volunteers link in the navigation bar to view all the volunteers that was engaged with the system. Click on the View Detail button to view the volunteer timeline that contains there all the done activities.

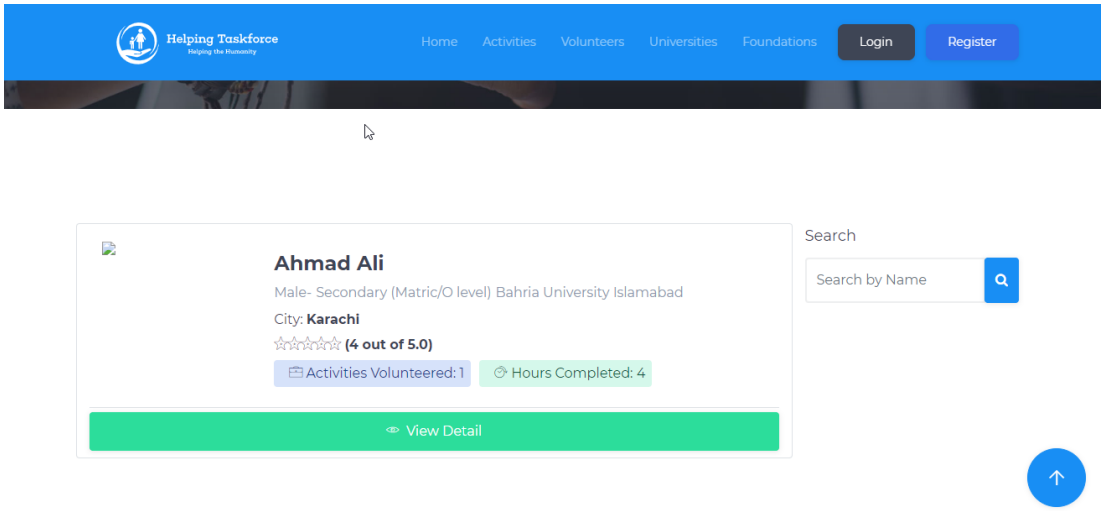

Figure 5.2: Volunteers

5.1.1.3 Activities

Click on the Activities link in the navigation bar to view all uploaded activities in the system.

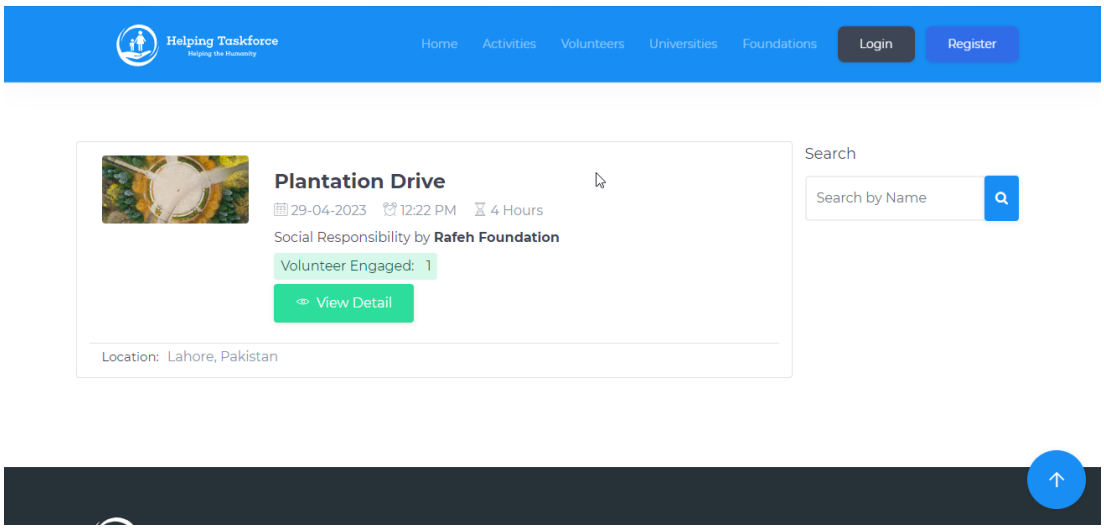

Figure 5.3: Activities

5.1.1.4 Login

Click on the Login button at the top, the login page loaded. Now, fill up the credentials and click Login. When the credentials are verified, it redirects user to their access area.

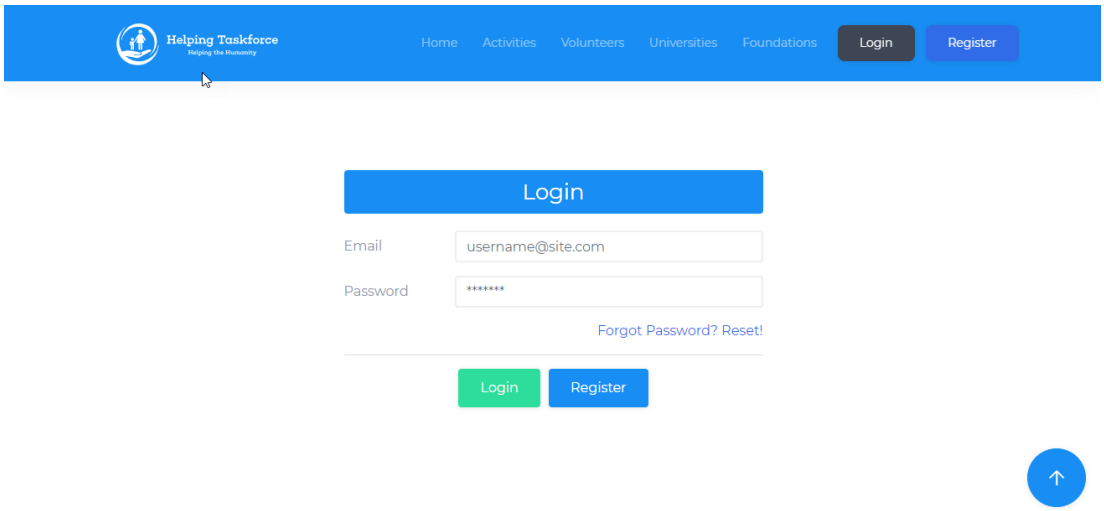

Figure 5.4: Login Page

5.1.1.5 Register as Volunteer

Click on the register button at the top, to register yourself as volunteer. Volunteer is only registered if their university was already registered in this system. Choose the University and fill up other details and click on Register. Volunteer can get an email on their provided email for activating the account.

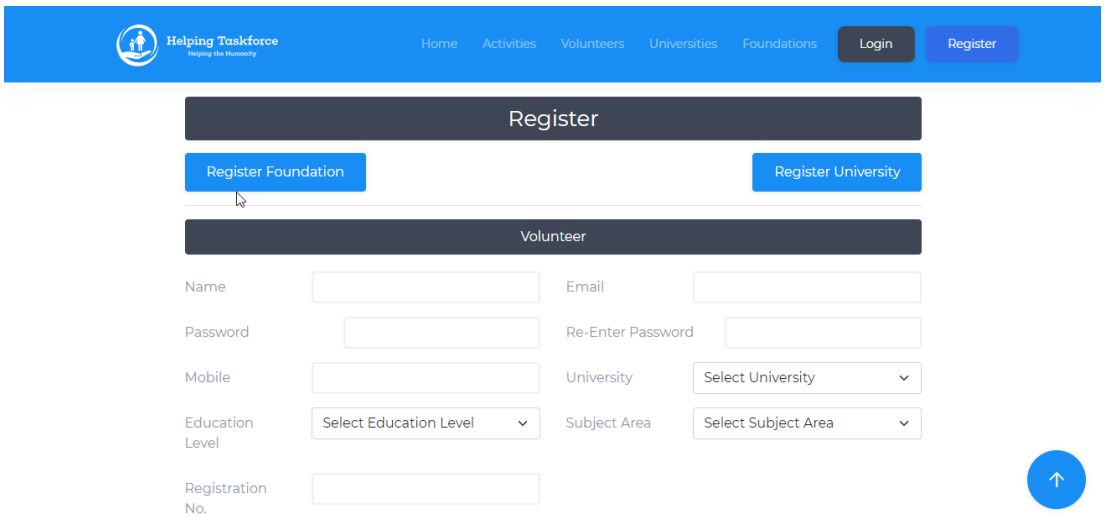

Figure 5.5: Volunteer Registration Page

5.1.1.6 Register as University

Click on the register button, the volunteer registration page open, now click on the Register University button to register as university in this system, so the university student can engage with this system. University user can receive the activation email after click on the register button to activate the account.

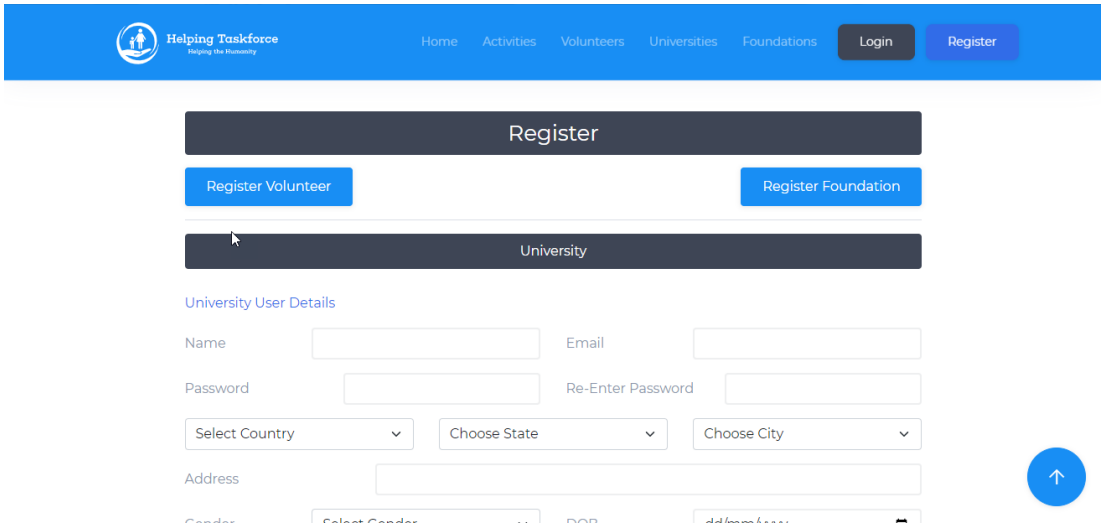

Figure 5.6: University Registration Page

5.1.1.7 Foundation

Click on the register button, the volunteer registration page open, now click on the Register Foundation button to register as community service organization in this system. User will receive an activation email to activate their account.

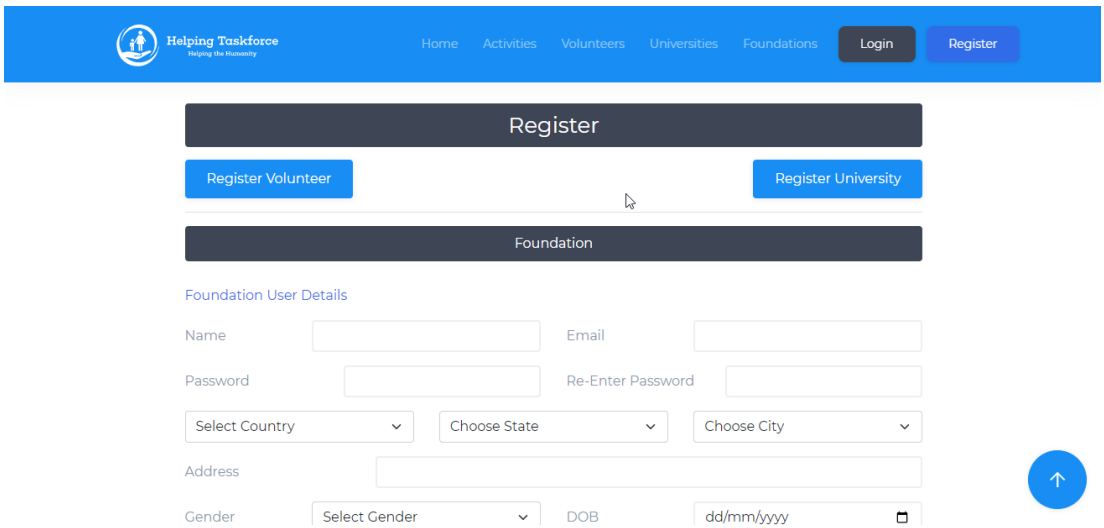

Figure 5.7: Community Service Organization Registration Page

5.1.2 Admin

5.1.2.1 Manage Institute

Click on the Add New Institute button to add the new University, College, etc in this system by filling up the desired information. Admin can update the university profile data or delete the institute from the system.

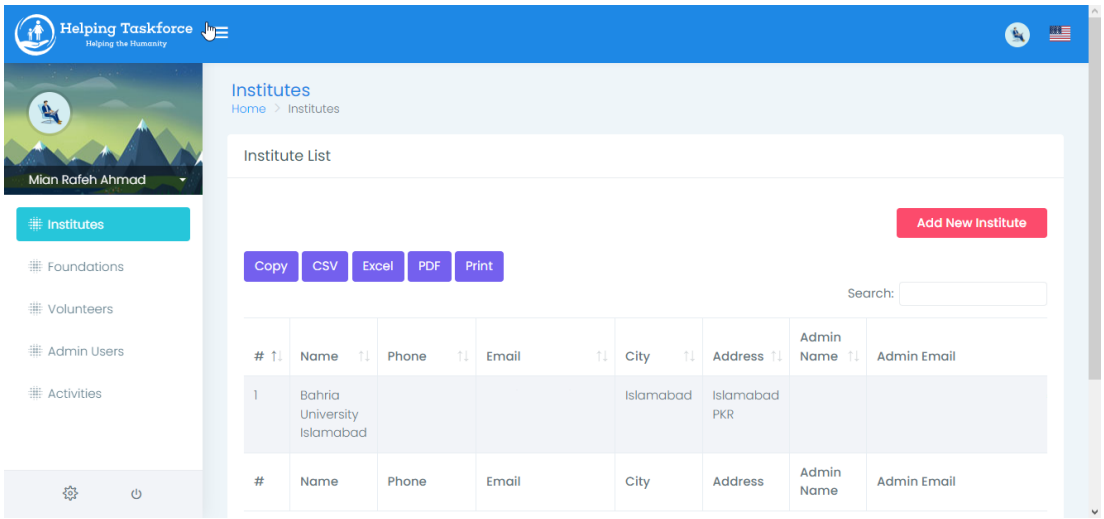

Figure 5.8: Manage Institute

5.1.2.2 Manage Foundation

Click on the Add New Foundation button to add the new Foundation, Community Service Organization, etc in this system by filling up the desired information. Admin can update the Foundation profile data or delete the institute from the system.

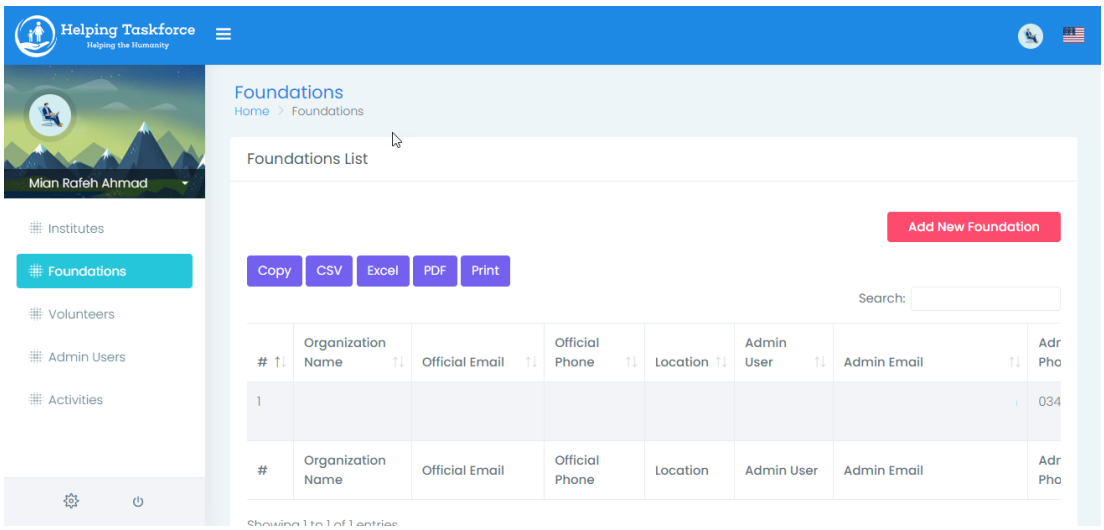

Figure 5.9: Manage Foundation

5.1.2.3 View Volunteers

By click on the Volunteers button in the sidebar, the admin user can see all the volunteers that was registered in this system.

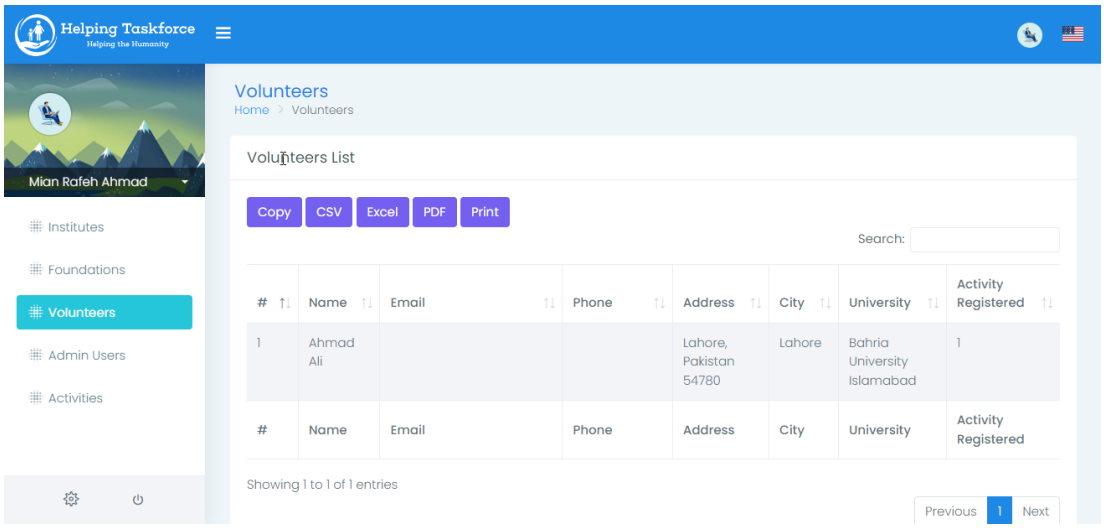

Figure 5.10: Volunteers

5.1.2.4 View Activities

By click on the Activities button in the sidebar, the admin user can see all the activities that was posted in this system.

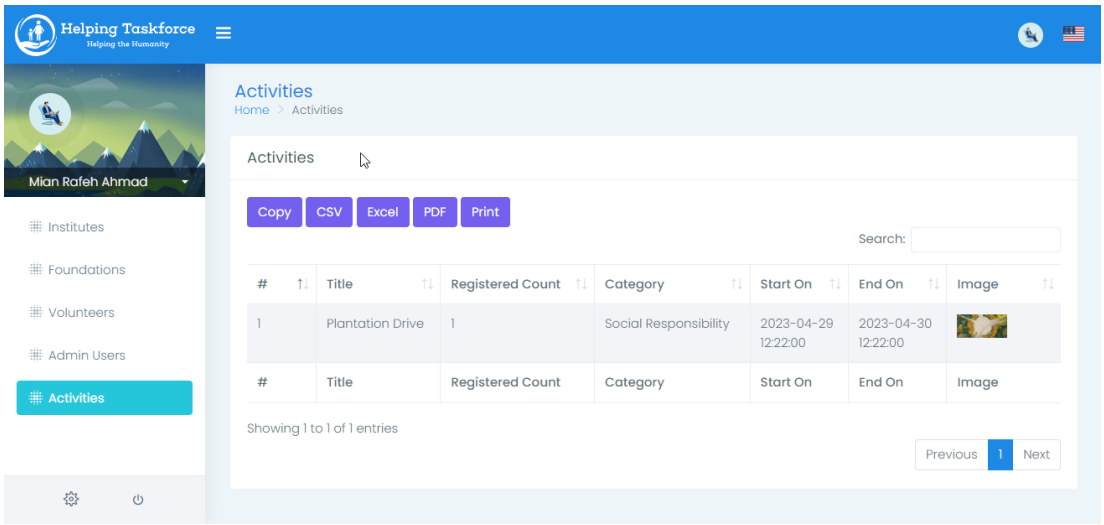

Figure 5.11: Activities

5.1.2.5 Add New User

Click on the Admin Users, to view all the admins of the platform, To add the new admin, click on the Add New User, the below page will be loaded. Now fill up the information and submit the form. User will receive an email on the provided email for account activation.

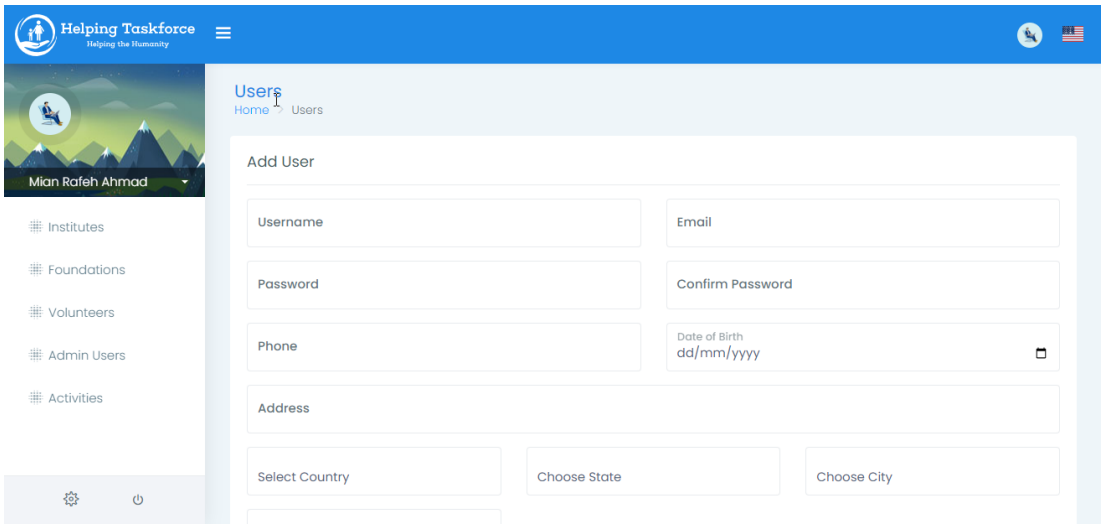

Figure 5.12: Add User

5.1.2.6 View/Update Profile

Click on the setting button, from the bottom of the sidebar, to view or update his/her profile information.

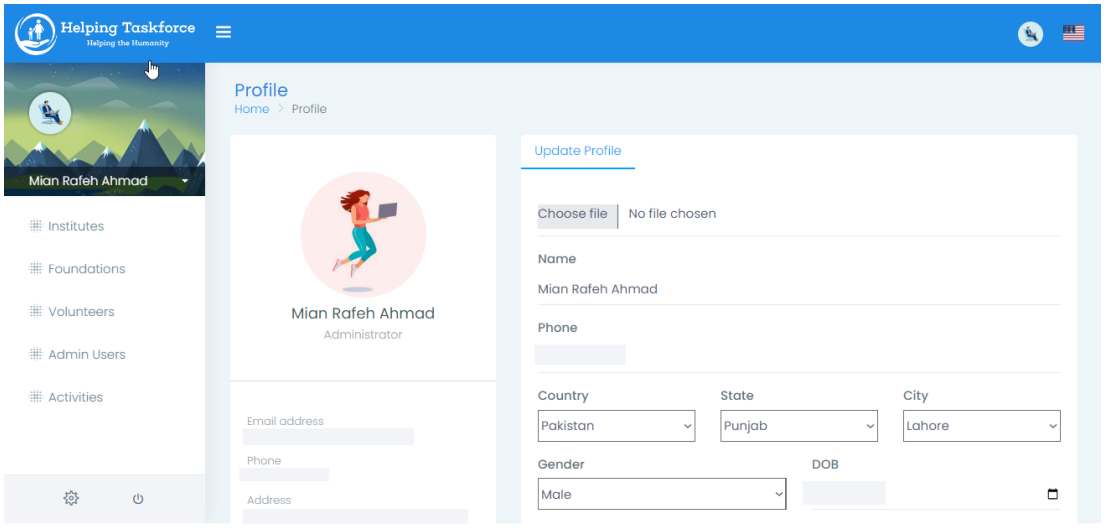

Figure 5.13: View / Update Profile

5.1.3 Foundation

5.1.3.1 Activities

Click on the Activities to button, to load all the activities that was posted by the Foundation of the user.

To add new activity, click on the Add Activity button.

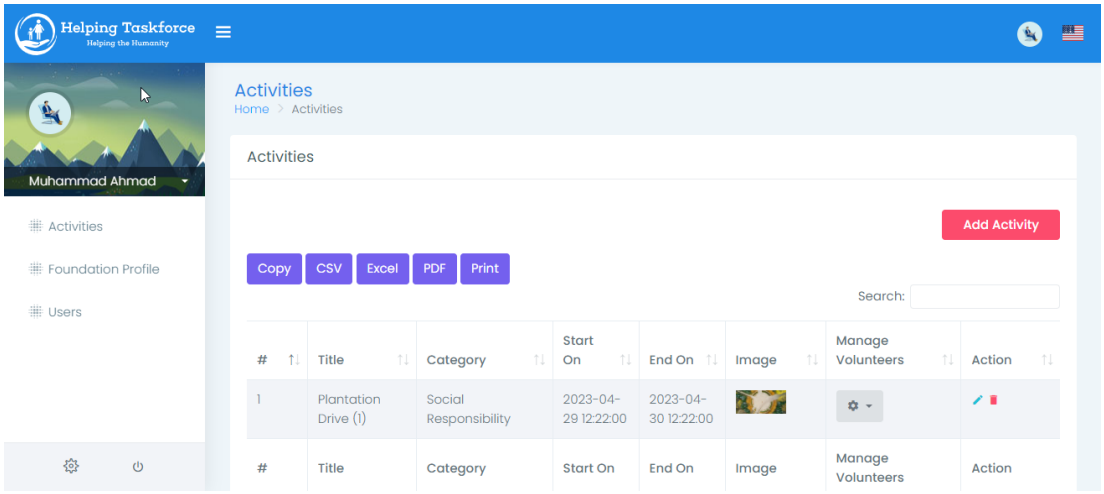

Figure 5.14: Activities

5.1.3.2 Add New Activity

By clicking on the Add Activity button, the below page will be loaded. Fill up the activity detail and click on submit. Activity will be posted.

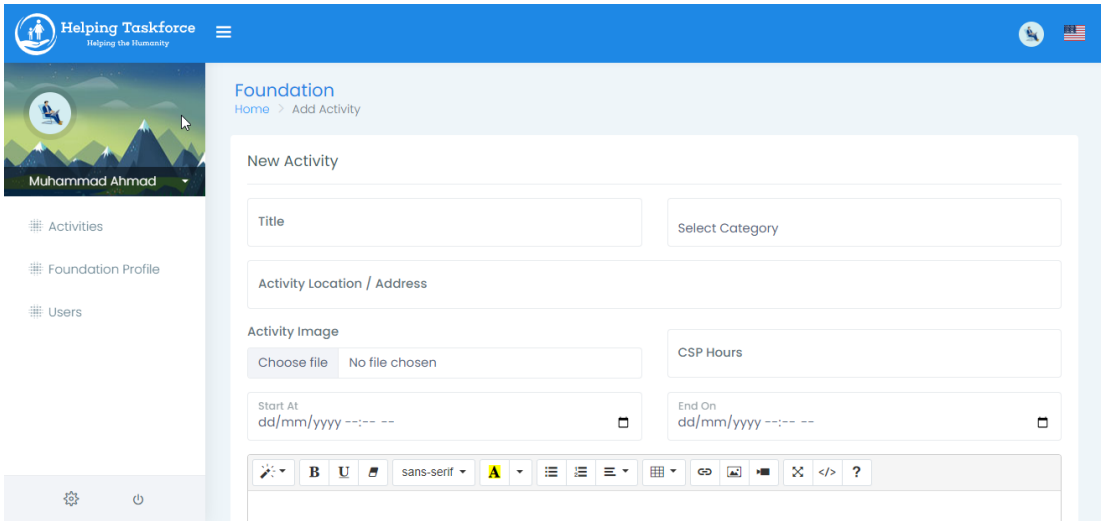

Figure 5.15: Add Activity

5.1.3.3 Manage Volunteer Presence

Click on the Manage Volunteers button, against the activity, the menu will appear. Now click on the Manage Attendance button to load the volunteer that register this activity, and to manage their presence.

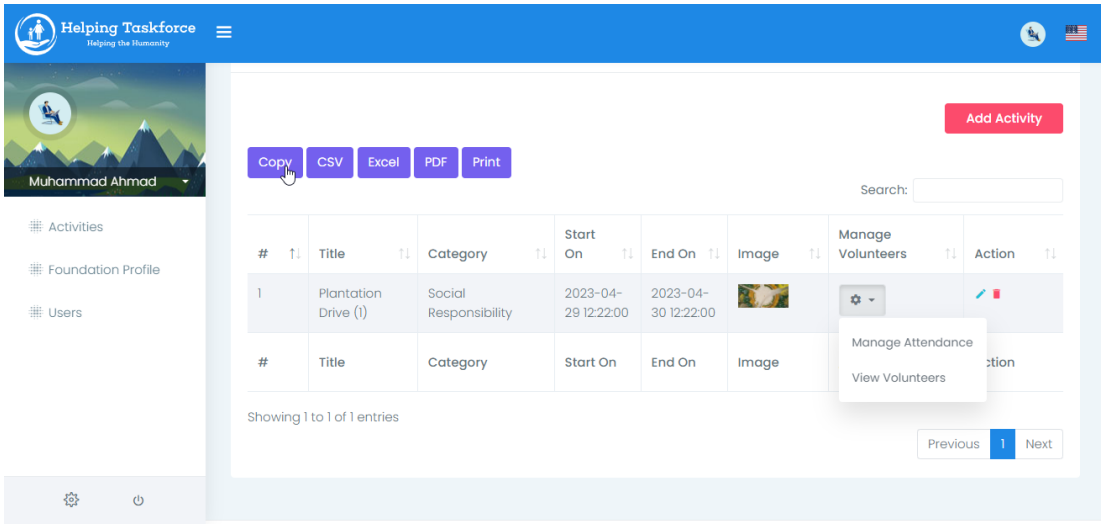

Figure 5.16: Activities

Mark the volunteer presence and give rating to the volunteers, and then click on the save button to save the record.

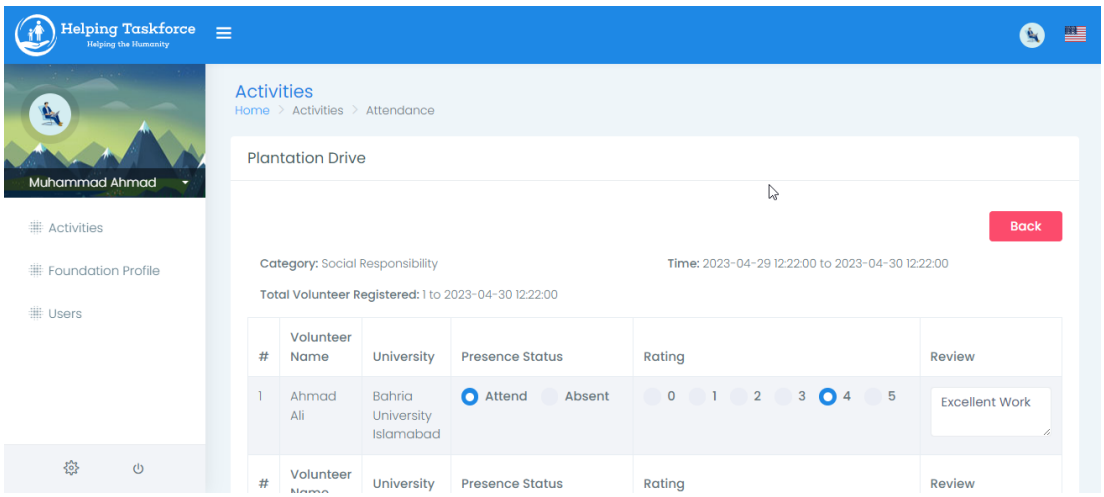

Figure 5.17: Manage Volunteer Presence

5.1.4 University

5.1.4.1 Manage Students

Click on the Students button from the sidebar menu to load the students of the University whose user was logged in. University user can see the CSP hours and rating from this screen.

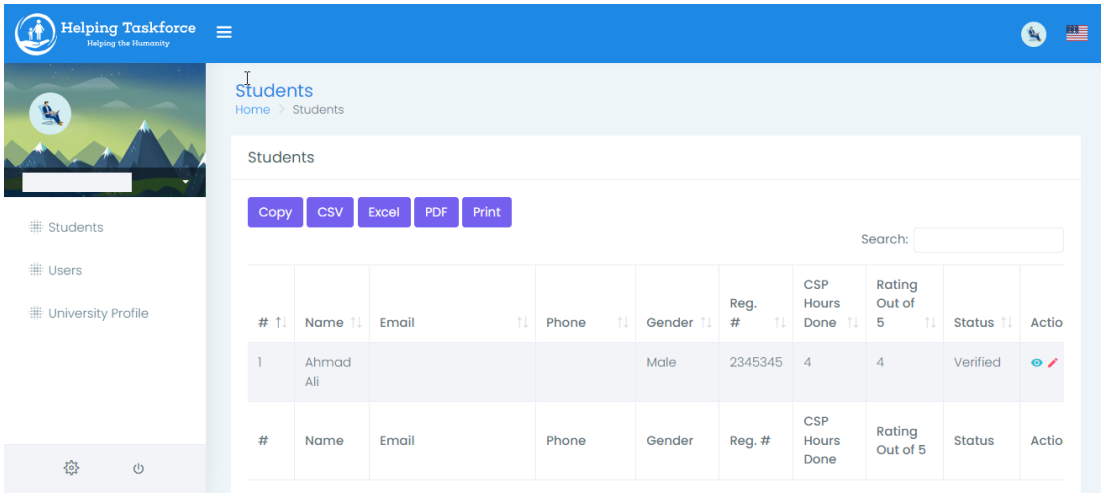

Figure 5.18: Manage Students

To verify the student volunteer that they belong to this university click on the edit button against the student. The pop-up model will show, choose verified if the students below to your organization and save the record.

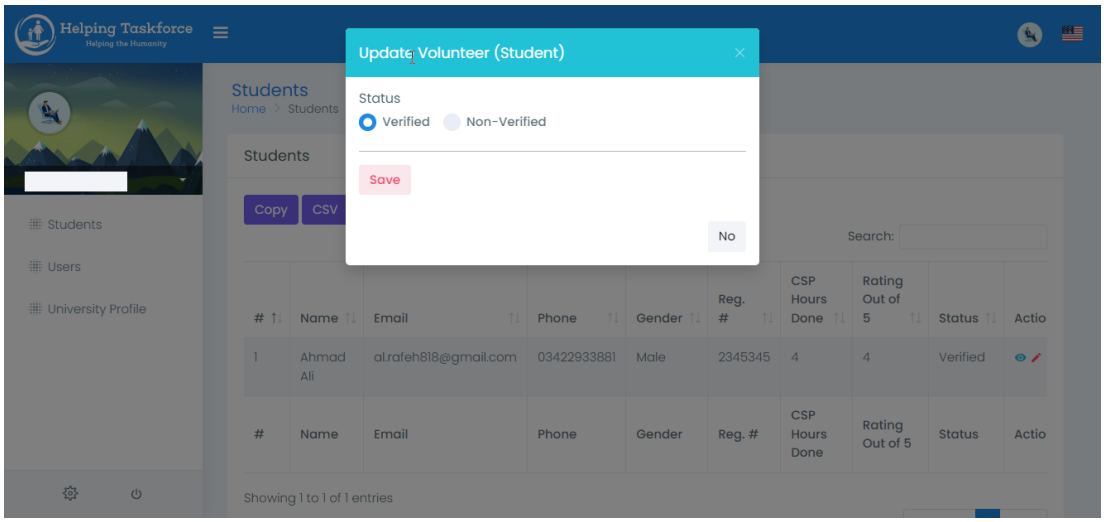

Figure 5.19: Verify Volunteer

5.1.4.2 Manage Users

University main user can add more user to the profile, but these users have the limited access to the account. They can only Manage the students.

To add the New User, click on the Add New User button and the page will be loaded. Fill up the desired information about the user and submit. User will receive the email on the provided email to activate the account.

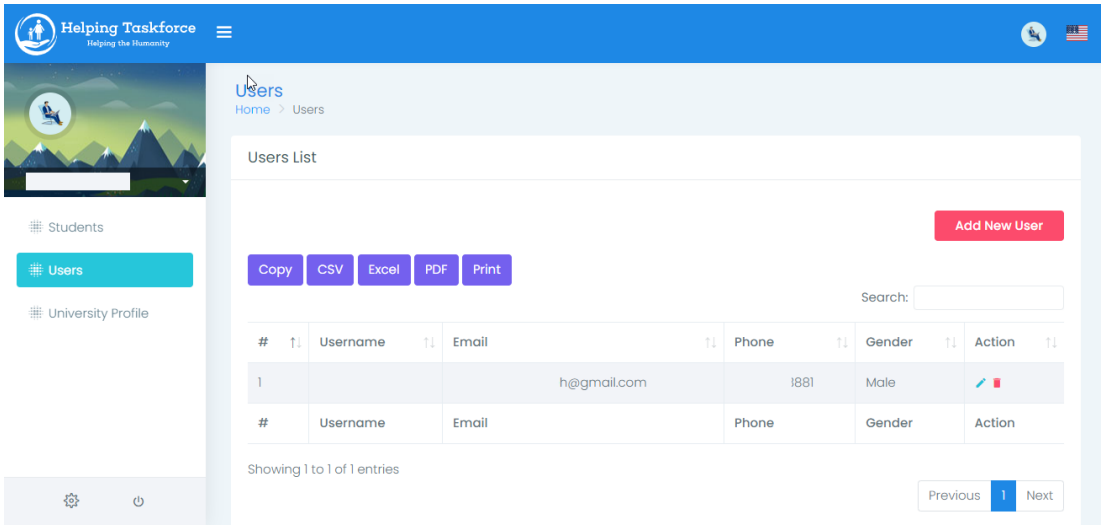

Figure 5.20: Manage University Users

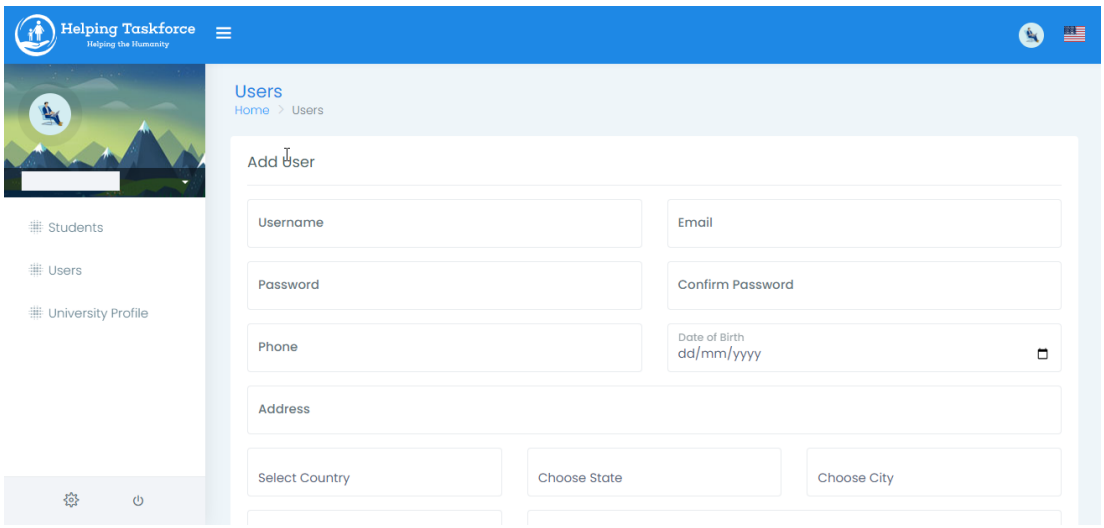

Figure 5.21: Add New User

5.1.4.3 University Profile

To update the University details, click on the University Profile button from the sidebar, and update the details.

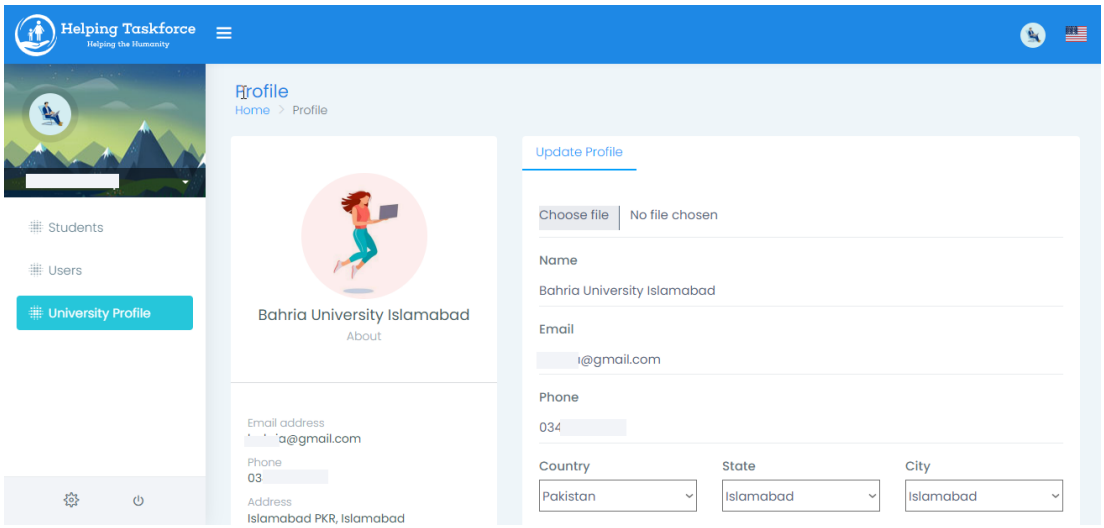

Figure 5.22: View / Update University Profile

5.1.4.4 View / Update Profile

To update your profile information, click on the setting button from the sidebar and update the details.

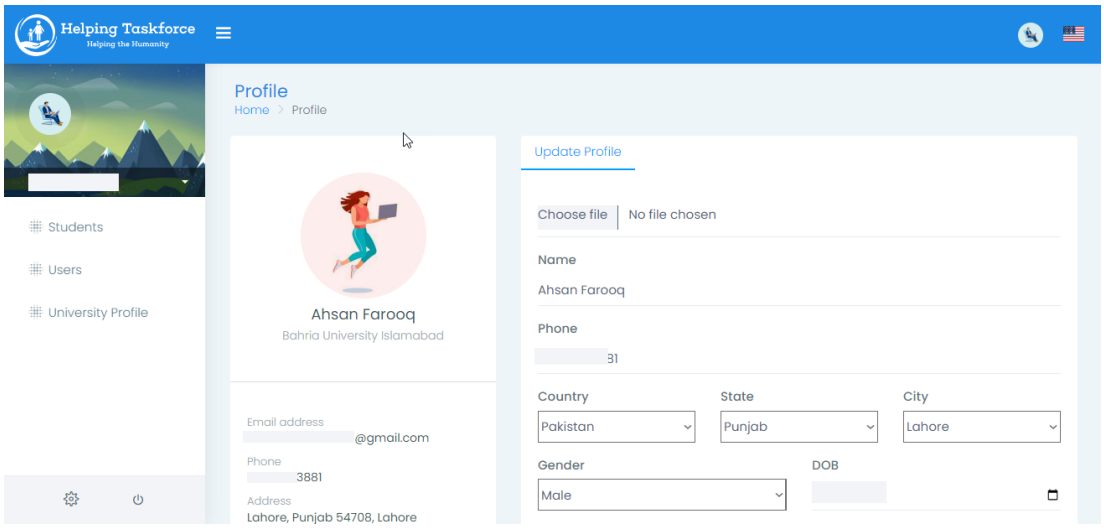

Figure 5.23: View / Update Profile

5.1.5 Volunteer

5.1.5.1 View Activities

To View all the activities, click on the activities button from the sidebar the activities will be loaded.

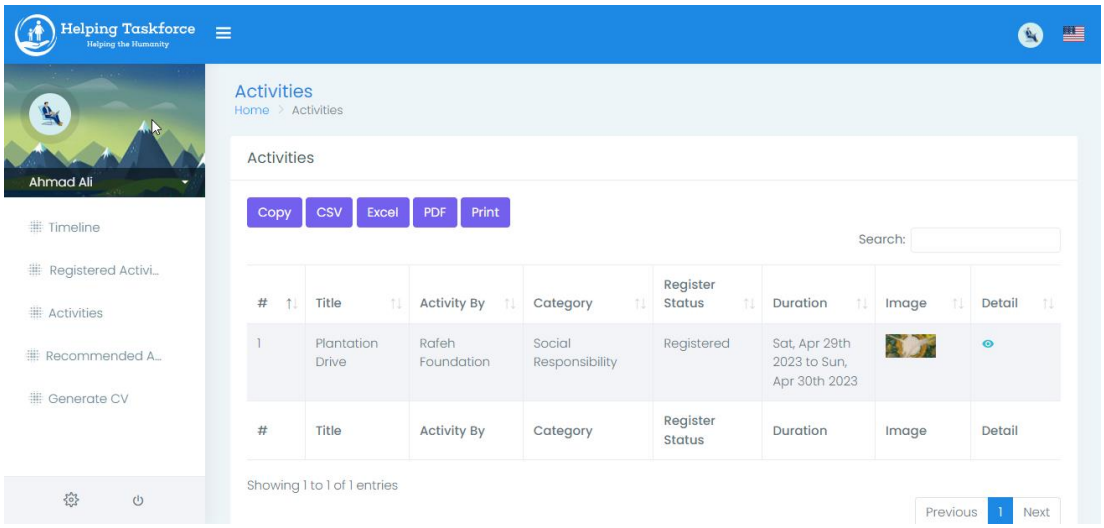

Figure 5.24: View Activities

5.1.5.2 View Recommended Activities

To View all the recommended activities, click on the activities button from the sidebar the activities will be loaded.

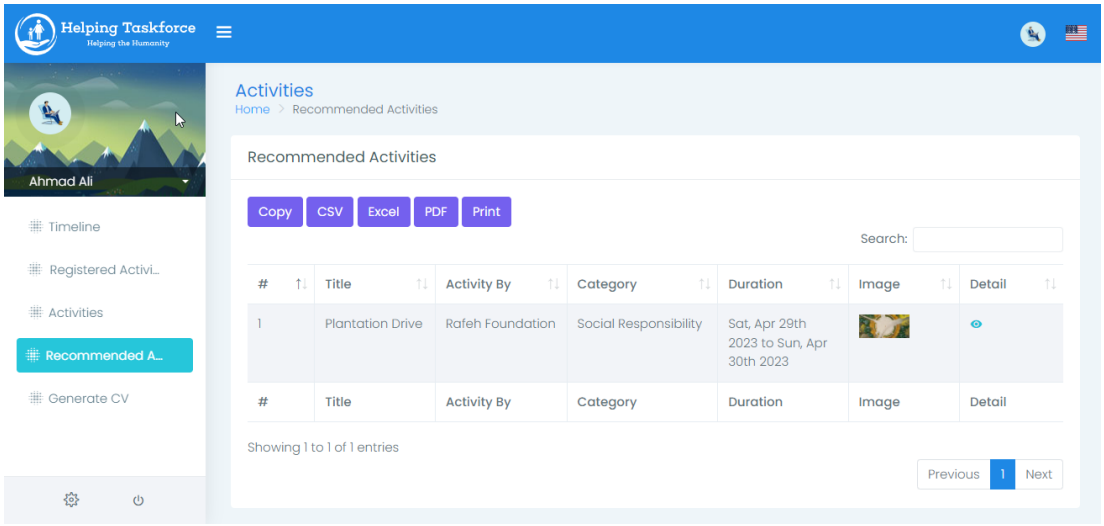

Figure 5.25: View Recommended Activities

5.1.5.3 Timeline

To view your timeline, click on the timeline button from the sidebar, the timeline of all the done past activities will be loaded.

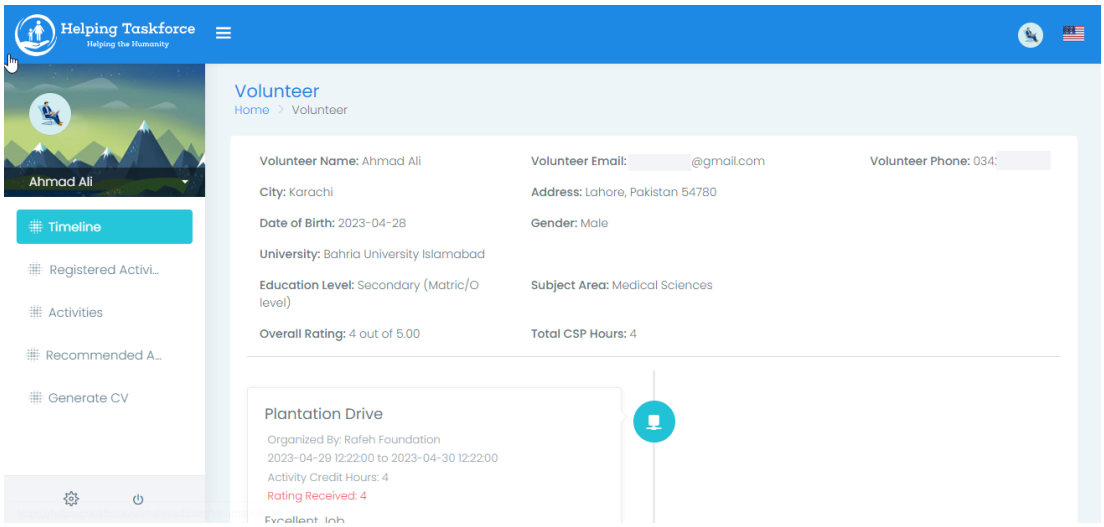

Figure 5.26: Timeline

5.1.5.4 Generate CV

To print or download the CV, click on the Generate CV button from the sidebar menu, the CV layout will be loaded, scroll to the bottom, and click on the print button the download the CV or print the CV.

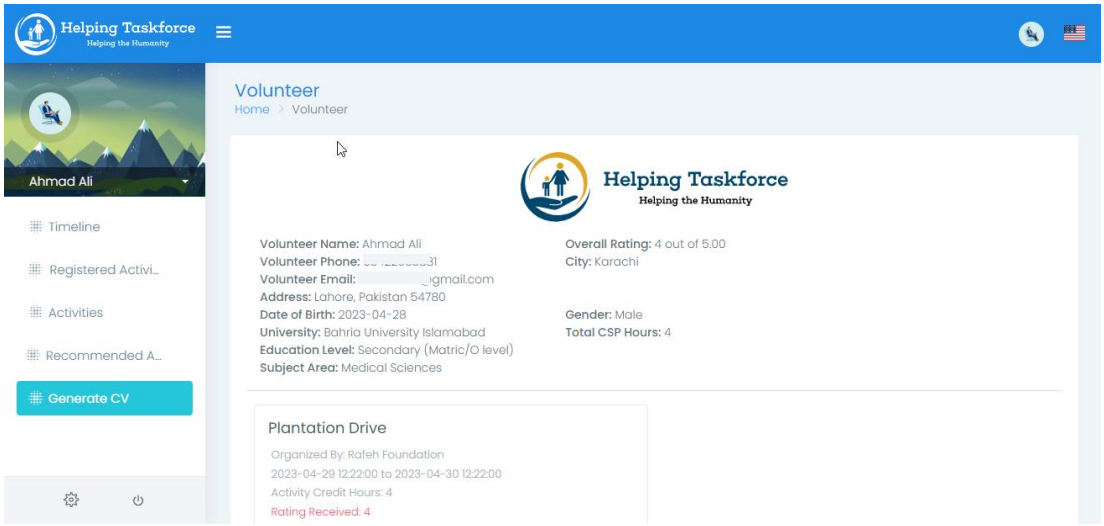

Figure 5.27: Generate CV

5.1.5.5 View Registered Activities

To view the registered activities, that you registered in the past, click on the Registered Activities button, the registered activities will be loaded.

To give rating and review to the activity click the edit button against the activity, the pop-up model open, write the review and save.

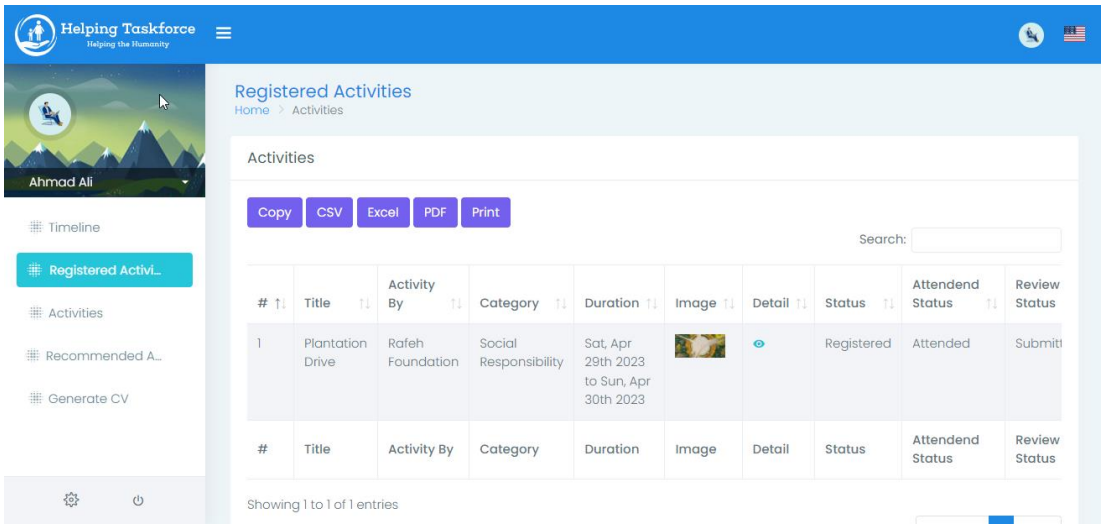

Figure 5.28: Registered Activities

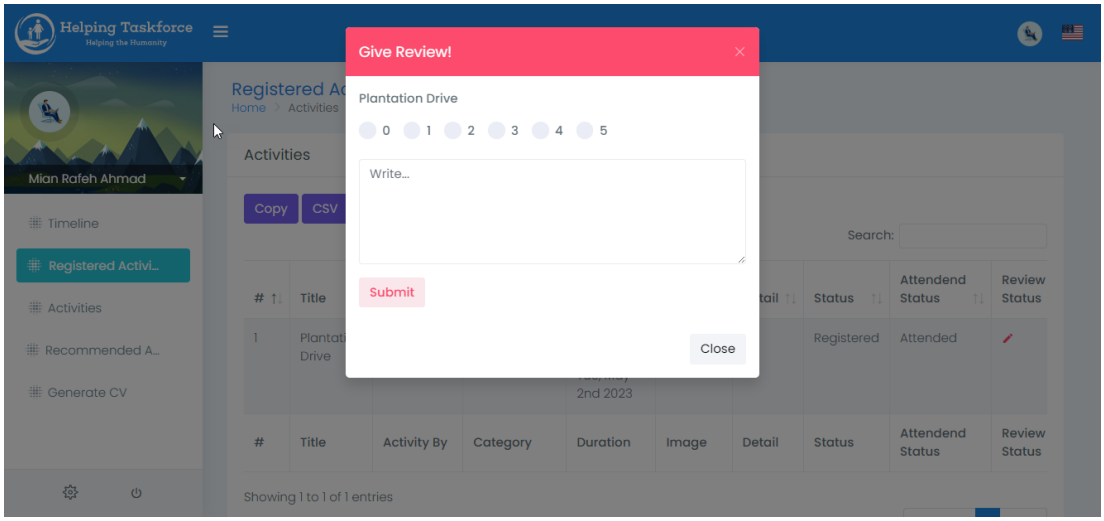

Figure 5.29: Review Activity

CHAPTER 6

CONCLUSION AND RECOMMENDATIONS

6.1 Conclusion

The main purpose of this system is to provide a platform to different stakeholders i.e., volunteers, community service organizers (foundations), Universities to maintain the CSP hours of the volunteers without any problem. There will be no need to make announcement about different volunteerism activities through Chat groups or used different forms for registration purposes. This way the volunteer linked to the universities can visit website and register themselves to participate on different activities and gathered there CSP hours easily. This also minimize the load of the universities to manage the students CSP hours manually.

6.2 Recommendation

Regarding this project, we would like to recommend that a lot of more work can be done on this project which may include extending database, increasing productivity, a mobile application can also be designed for Android or iOS. The users should use a 3G or 4G facility on their devices otherwise they could not access the system or to avoid any interruption due to poor internet connectivity.

REFERENCES

Electronic Sources from Internet:

[1] Al-Khidmat, “Al-Khidmat Foundation Pakistan – Leading Charity Foundation”. On-Line, Available at: https://alkhidmat.org/.

[2] Akhuwat, “Akhuwat | NGO for Zakat, Sadaqah, Islamic Microfinance”. On-Line, Available at: https://akhuwat.org.pk/.

[3] UNDP, “Pakistan | United Nations Development Programme”. On-Line, Available at: https://www.undp.org/pakistan.

[4] GivingWay, “GivingWay”. On-Line, Available at: https://www.instagram.com/givingway/?hl=en.

[5] CodeIgniter, “Database Configuration”. On-Line, Available at: https://codeigniter.com/userguide3/database/configuration.html.

[6] Cloundinary, “Upload SDKS with the Upload API”. On-Line, Available at: https://cloudinary.com/documentation/image_upload_api_reference.

[7] Hostinger Tutorials, “CRON Job”. On-Line, Available at: https://www.hostinger.com/tutorials/cron-job.’

[8] SendGrid, “SendGrid – Documentation”. On-Line, Available at: https://docs.sendgrid.com/.

APPENDICES

APPENDIX A: Volunteer CV

Helping Taskforce
Helping the Humanity

Volunteer Name: Mian Rafeh Ahmad
Overall Rating: 5 out of 5.00
Volunteer Phone: 03422933881
City: Lahore
Volunteer Email: 03-134192-018@student.bahria.edu.pk
Address: 34-E Marghazar Colony Multan Road, Lahore
Date of Birth: 1999-09-03
Gender: Male
University: Bahria University
Total CSP Hours: 30
Education Level: Bachelors/ BS (16 years)
Subject Area: Computer Science & Information Technology

Fundraising and Donation
Organized By: Alkhidmat Foundation Pakistan
2023-05-12 12:42:00 to 2023-05-20 12:42:00
Activity Credit Hours: 6
Rating Received: 5
--

Plantation Drive
Organized By: Alkhidmat Foundation Pakistan
2023-05-12 12:42:00 to 2023-05-20 12:42:00
Activity Credit Hours: 6
Rating Received: 5
--

Cloths Drive
Organized By: Alkhidmat Foundation Pakistan
2023-05-12 12:42:00 to 2023-05-20 12:42:00
Activity Credit Hours: 6
Rating Received: 5
--

Raise the Fund
Organized By: Alkhidmat Foundation Pakistan
2023-05-12 12:42:00 to 2023-05-20 12:42:00
Activity Credit Hours: 6
Rating Received: 5
--

Food for All
Organized By: Alkhidmat Foundation Pakistan
2023-05-12 12:42:00 to 2023-05-20 12:42:00
Activity Credit Hours: 6
Rating Received: 5
--

Figure APPENDIX-A: Volunteer CV

BSCS-F22-001

ORIGINALITY REPORT

14%	8%	1%	12%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	Submitted to Higher Education Commission Pakistan Student Paper	2%
2	www.coursehero.com Internet Source	2%
3	Submitted to Edith Cowan University Student Paper	1%
4	dspace.daffodilvarsity.edu.bd:8080 Internet Source	1%
5	Submitted to CEPT University Student Paper	1%
6	Submitted to Universiti Tunku Abdul Rahman Student Paper	1%
7	Submitted to British University in Egypt Student Paper	<1%
8	Submitted to Informatics Education Limited Student Paper	<1%
9	gecgudlavalleru.ac.in Internet Source	<1%

10	Submitted to University of Houston, Downtown Student Paper	<1 %
11	www.bluephare.com Internet Source	<1 %
12	Submitted to University of Essex Student Paper	<1 %
13	bphanikrishna.files.wordpress.com Internet Source	<1 %
14	Submitted to The British College Student Paper	<1 %
15	bnrhvprod-ssb.desu.edu Internet Source	<1 %
16	Submitted to Texas A&M University, College Station Student Paper	<1 %
17	docplayer.net Internet Source	<1 %
18	Submitted to RMIT University Student Paper	<1 %
19	Submitted to Taylor's Education Group Student Paper	<1 %
20	Submitted to Eastern Mediterranean University Student Paper	<1 %

21	Submitted to University of Hertfordshire Student Paper	<1 %
22	Submitted to CSU, San Jose State University Student Paper	<1 %
23	Submitted to University of Wales Institute, Cardiff Student Paper	<1 %
24	Submitted to HELP UNIVERSITY Student Paper	<1 %
25	Submitted to National School of Business Management NSBM, Sri Lanka Student Paper	<1 %
26	zombiedoc.com Internet Source	<1 %
27	Submitted to University of Teesside Student Paper	<1 %
28	edgefolio.com Internet Source	<1 %
29	Submitted to Gulf College Oman Student Paper	<1 %
30	Submitted to University of Greenwich Student Paper	<1 %
31	coronet.iicm.edu Internet Source	<1 %

32	www.cram.com Internet Source	<1 %
33	unbscholar.lib.unb.ca Internet Source	<1 %
34	Sibanjan Das. "Data Science Using Oracle Data Miner and Oracle R Enterprise", Springer Nature, 2016 Publication	<1 %
35	dokumen.tips Internet Source	<1 %
36	ng6.toulouse.inra.fr Internet Source	<1 %
37	pdfcoffee.com Internet Source	<1 %
38	www.etravel.slb.com Internet Source	<1 %
39	www.slideshare.net Internet Source	<1 %

Exclude quotes	On	Exclude matches	Off
Exclude bibliography	On		