Event Notification Application

By

Somia Khanam

Supervised by

Dr. Ghazanfer Farooq

Department of Computer Science

Quaid-i-Azam University

Islamabad

Session (2016-2018)

QUAID-I-AZAM UNIVERSITY
DEPARTMENT OF COMPUTER SCIENCE

Dated: October 26, 2018

FINAL APPROVAL

This is to certify that we have read the project report submitted by Ms. Somia Khanam and it is our judgment that this report is of sufficient standard to warrant its acceptance by the Quaid-i-Azam University, Islamabad for the degree of the Master of Science in Computer Science

COMMITTEE

1. External Examiner ______________________

Dr. Ayyaz Hussain
Associate Professor-FBAS
Department of Computer Science & Software Engineering
International Islamic University
Sector H-10
Islamabad

2. SUPERVISOR

Dr. Ghazanfar Farooq Siddique ______Siddiqi______
Assistant Professor
Deptt. of Computer Sciences
Quaid-i-Azam University
Islamabad.

3. INCHARGE

Dr. Onaiza Maqbool ______Onaiza______
Associate Professor
Deptt. of Computer Sciences
Quaid-i-Azam University
Islamabad

ACKNOWLEDGEMENT

In the name of ALLAH, the most Merciful and most Compassionate. First of all, All the praises, thanks and acknowledgments are for the creator Allah Almighty, who gave me strength and supported me to undertake and complete this task. Countless Darood upon the Holy Prophet Hazrat Muhammad (S.A.W), the supporter of humanity, the gem of mankind and source of knowledge for enlightening the world with the essence of faith in Allah.

I express my thanks and full-hearted gratitude to my respected supervisor ***Dr. Ghazanfer Farooq*** for his full guidance and entire support in completing this project successfully. I am thankful to him for the constant encouragement and continuous inspiration that he has given to me.

I am also very thankful to all my teachers and friends who have been always helping and encouraging me throughout this time period. I have no valuable words to express my thanks, but my heart is still full of the favors received from every person.

Finally I am also very thankful to my family members for their unconditional support and encouragement during my project work without which I would hardly be capable of producing this huge work.

Somia Khanam

2016-2018

Abstract

"Event Notification Application" is an android application which is helpful for the users. In this app users can easily find events of their interest. Event Notification App can keep a record of the participants. It will crawl data from the website and provide up-to-date data to the users. In our proposed system, users will get notifications about latest events from nearby places in Islamabad. This system will create a profile of the user. The user can invite his/her friend. For security purposes, it will allow the authenticated users only to use this application. The admin can delete user and start, stop crawling.

List of Tables

Table 2.3.1: Comparison of Different Applications 24

Table 2.2 UC-1: Login 27

Table 2.3 UC-2:Register user 28

Table 2.4 UC-3: Logout 29

Table 2.5 UC-4: invite friends 30

Table 2.6 UC-5: Select interest 31

Table 2.7 UC-6:Select Time 32

Table 2.8 UC-7:select Date 33

Table 2.9 UC-8: Search event 34

Table 2.10 UC-9: Register for event 35

Table 2.11 UC-10: Participate in event 36

Table 2.12 UC-11: Get invitation reminder 37

Table 2.13 UC-12 View Profile 38

Table 2.14 UC-13 Edit Profile 39

Table 2.15 UC-14: Upcoming events notifications 40

Table 2.16 UC-15: assign rating 41

Table 2.17 UC-16:View previous events 42

Table 2.18 UC-17 Delete User 43

Table 2.19 UC-18: Start crawling 44

Table 2.20 UC-19: Stop crawling 45

Table 2.21 UC-20: Set time interval 46

Table 5.22 TC-1: Login 81

Table 5.23 TC-3 Register User 82

Table 5.24 TC-3: Edit profile 83

Table 5.25 TC-4: Select Interest 84

Table 5.26 TC-5: Search event 85

Table 5.27 TC-6: Give rating 86

List of Figures

Figure 1.1 Implementation Time table (I) 18
Figure 1.2 Implementation Time table (II) 18
Figure 1.3 Implementation Gantt chart (I) 19
Figure 1.4 Implementation Gantt chart (II) 19
Figure 2.5 Use case diagram 26
Figure 2.6 ERD 49
Figure 2.7 SSD-1: Login 50
Figure 2.8 SSD-2: Register user 51
Figure 2.9 SSD-3: Logout 51
Figure 2.10 SSD-4: Invite Friends 52
Figure 2.11 SSD-5: Select Interest 52
Figure 2.12 SSD-6: Select Time 53
Figure 2.13 SSD-7: Select Date 53
Figure 2.14 SSD-8: Search Events 54
Figure 2.15 SSD-9: Register for event 54
Figure 2.16 SSD-10: Participate in event 55
Figure 2.17 SSD-11:Get Invitation Reminder 55
Figure 2.18 SSD-12: View Profile 56
Figure 2.19 SSD-13: Edit Profile 56
Figure 2.20 SSD-14: Upcoming Events notifications 57
Figure 2.21 SSD-15: Assign rating 57
Figure 2.22 SSD-16: View previous events 58
Figure 2.23 SSD-17: Delete user 58
Figure 2.24 SSD-18: start crawling 59
Figure 2.25 SSD-19: stop crawling 59
Figure 2.26 SSD-20: set time interval 60
Figure 2.27 Domain Model 61
Figure 3.28 System Architectural Diagram 64
Figure 3.29 Class diagram 65
Figure 3.30 SD-1: Login 66

Figure 3.31 SD-2: Sign Out 67
Figure 3.32 SD-3: Edit profile 67
Figure 3.33 SD-4: Assign rating 68
Figure 3.34 SD-5: Select Interest 68
Figure 3.35 SD-6: Search Events 69
Figure 3.36 SD-7: Register Event 69
Figure 4.37 screenshot of user login 73
Figure 4.38 screenshot of profile 74
Figure 4.39 Screenshot for Signup 74
Figure 4.40 screenshot of events detail 75
Figure 4.41 screenshot of assign rating 75
Figure 4.42 screenshot of applying filters 76
Figure 4.43 screenshot of Admin login 77
Figure 4.44 screenshot of Admin Panel 77

Table of Contents

ACKNOWLEDGEMENT i

Abstract ii

List of Tables iii

List of Figures iv

1.1 Introduction 13

1.1.1 Product Overview 13

1.1.2 Project Deliverables 13

1.2 Project Organization 13

1.2.1 Software Process Model 13

1.2.2 Roles and Responsibilities 14

1.2.3 Tools and Techniques 14

1.3 Project Management Plan 14

1.3.1 Tasks 14

1.3.2 Assignments 17

1.4 Task Schedule 18

1.4.1 Time Table 18

1.4.2 Gantt Chart 19

Chapter 2 20

Software Requirement Specification 20

2.1 Introduction 21

2.1.1 Problem Statement 21

2.1.2 Motivation 21

2.2 Product Overview 21

2.2.1 Project Scope 22

2.3 Related Work 22

2.3.1 Comparison of Event Manager Apps 24

2.4 Specific Requirement 25

2.4.1 External Interface Requirement 25

User Interface 25

Hardware Interface 25

Software Interface 25

2.5 Use Case Diagram 26

2.6 Use Case description 27

UC-1: Login 27

UC-2: Register User 28

UC-3: Logout 29

UC-4: Invite friends 30

UC-5: Select interests 31

UC-6: Select time 32

UC-7: Select Date 33

UC-8: Search Event 34

UC-9: Register for the event 35

UC-10: Participate in event 36

UC-11: Get invitation reminder 37

UC-12: View Profile 38

UC-13: Edit Profile 39

UC-14: Upcoming events Notification 40

UC-15: Assign rating 41

UC-16: View previous events 42

UC-17: Delete User 43

UC-18: Start Crawling 44

UC-19: Stop Crawling 45

UC-20: Set time interval 46

2.7 Software System Attributes 47

2.7.1 Reliability 47

2.7.2 Availability 47

2.7.3 Security 47

2.7.4 Portability 47

2.7.5 Maintainability 47

2.7.6 Performance 48

2.8 Database Requirement 49

2.8.1 ERD 49

2.9 System Sequence Diagram 50

SSD1: Login 50

SSD2: Register User 51

SSD3: Logout 51

SSD4: Invite friends 52

SSD5: Select Interest 52

SSD6: Select Time 53

SSD7: Select date 53

SSD8: Search Events 54

SSD9: Register for Event 54

SSD10: Participate in event 55

SSD11: Get Invitation Reminder 55

SSD12: View Profile 56

SSD13: Edit Profile 56

SSD14: Upcoming Events notification 57

SSD15: Assign Rating 57

SSD16: View Previous Events 58

SSD17: Delete User 58

SSD18: Start crawling 59

SSD19: Stop crawling 59

SSD20: Set time interval 60

2.10 Domain Model 61

Chapter 3 62

Software Design Description 62

3.1 Introduction 63

3.1.1 Design Overview 63

3.2 System Architectural Design 64

3.3 Class Diagram 65

3.4 Sequence Diagram 66

2.4.1 Sign in to system 66

2.4.2 Sign out from System 67

2.4.3 Edit user Profile 67

2.4.4 Assign Rating 68

2.4.5 Select interest 68

2.4.6 Search events 69

2.4.7 Register and Participate in Event 69

Chapter 4 70

Software Implementation Document 70
4.1 Introduction 71
4.2 Language Selection 71
4.3 Tools Selection 72
Android Studio 72
4.4 Resources 72
4.5 User Interface Design 73
Screenshot of Login 73
Screenshot of user profile 74
Screenshot of events details 75
Screenshot of assign rating to event 75
Screenshot of applying filters 76
Screenshot of Admin Login 77
Screenshot of Admin Panel 77
Chapter 5 78
Software Test Document 78
5.1 Introduction 79
1.1.1 System Overview 79
1.1.2 Test Approach 79
5.2 Test Plan 79
5.2.1 Features to be tested 79
5.2.2 Features not to be tested 80
5.2.3 Testing Tools and Environment 80
5.3 Test Cases 81
5.3.1 TC-1: Login 81

5.3.2 TC-2: Register User 82

5.3.3 TC-3: Edit Profile 83

5.3.4 TC-4: Select interest 84

5.3.5 TC-5: Search event 85

5.3.6 TC-6: Give rating 86

Chapter 6 87

Conclusion and Future Enhancement 87

6.1 Introduction 88

6.2 Summary 88

6.3 Conclusions 88

6.4 Future Enhancement 88

References 89

Chapter 1

Software Project Management Plan

1.1 Introduction

This chapter provides an overview of Project Event Notification. The Event notification application is an android application. This chapter describes milestones and deliverables of this project. It also describes tool and technique used to develop this system.

1.1.1 Product Overview

The Event Notification is an android based App which will facilitate users to find events in Islamabad. The user can easily find events of his/her interest. It will provide all upcoming events in Islamabad and user can show their interest to attend it. Furthermore, it will maintain a profile of each user. User invites his/her friend to the event. The user gets a notification about nearby events. It will ask the user for the rating of events he/she has attended. App first crawls data from different websites related to the events and then show them in the app according to the user's requirements.

1.1.2 Project Deliverables

- Software Project Management Plan (SPMP)
- Software Requirement Specifications (SRS)
- Software Design Description (SDD)
- Software Test Documentation (STD).

1.2 Project Organization

This section describes software process model, roles and responsibilities and tools and techniques.

1.2.1 Software Process Model

In this project incremental model will be used because

- Software will be generated quickly during the software life cycle.
- This model is less costly compared to others.
- It is generally easier to test and debug than other methods of software development because relatively smaller changes are made during each iteration.

1.2.2 Roles and Responsibilities

I am doing this project alone so all responsibilities are on me regarding this project.

1.2.3 Tools and Techniques

- Android studio
- Firebase Server
- MS Office
- Argo UML
- Genymotion (for testing)

1.3 Project Management Plan

This section describes the project tasks, milestones and deliverables of this project.

1.3.1 Tasks

The project has the following tasks.

1.3.1.1 Problem Understanding

Description

In this task project deliverables, milestones and resources needed will be described.

Resources Needed

Somia Khanam

Dependencies and Constraints

None

Risks and Contingencies

None

1.3.1.2 Software Project Management Plan

Description

In this task, the software approach and associated milestones will be included. It describes the functional and non-functional requirements of the proposed system. It also includes a set of use cases that describe user interaction that the software must provide.

Deliverables

Software Project Management Plan

Milestones

6-5-2018

Resources Needed

Somia Khanam

Dependencies and Constraints

Problem Understanding

Risks and Contingencies

None

1.3.1.3 Analysis and Requirement

Description

In this task analysis would be done and SRS will be included.

Deliverables

SRS

Milestones

29-5-2018

Resources Needed

Somia Khanam

Dependencies and Constraints

Software Project Management Plan

Risks and Contingencies

None

1.3.1.4 Develop System Design

Description

In this task of software, designer writers in order to give a software development team overall guidance to the architecture of the software project. It also describes the logic of design decisions taken.

Deliverables

Software design description (SDD)

Milestones

14-6-2018

Resources Needed

Somia Khanam

Dependencies and Constraints

Analysis and Requirement

Risks and Contingencies

None

1.3.1.5 Software Implementation

Description

In this task, we develop the plan how the project will be implemented.

Deliverables

Software Implementation Document (SID)

Milestones

4-8-2018

Resources Needed

Somia Khanam

Dependencies and Constraints

Develop System Design

Risks and Contingencies

None

1.3.1.6 Software Test Documentation

Description

In this task, plan and specifications to verify and validate the software and the results will be included

Deliverables

Software Test Document (STD)

Milestones

5-8-2018

Resources Needed

Somia Khanam

Dependencies and Constraints

Develop System Design

Risks and Contingencies

None

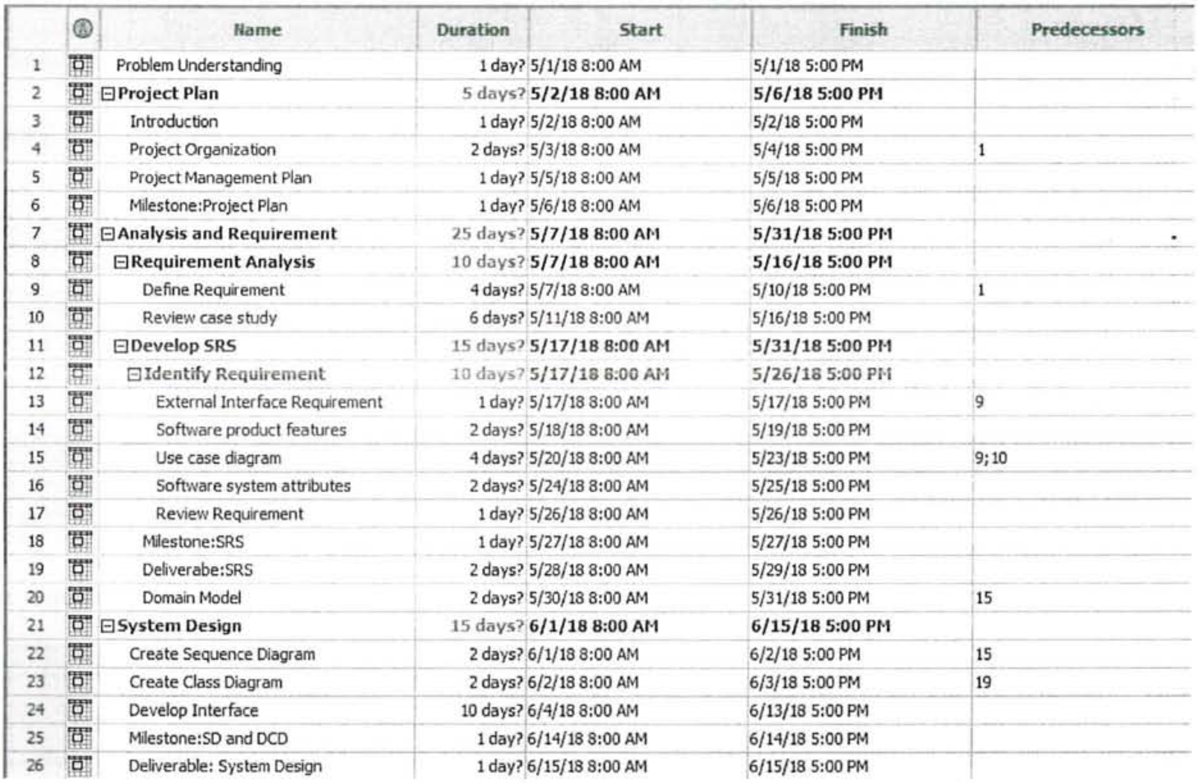
1.3.2 Assignments

It's a single person task. There is nothing assign to any other person. Everything will be done by me.

1.4 Task Schedule

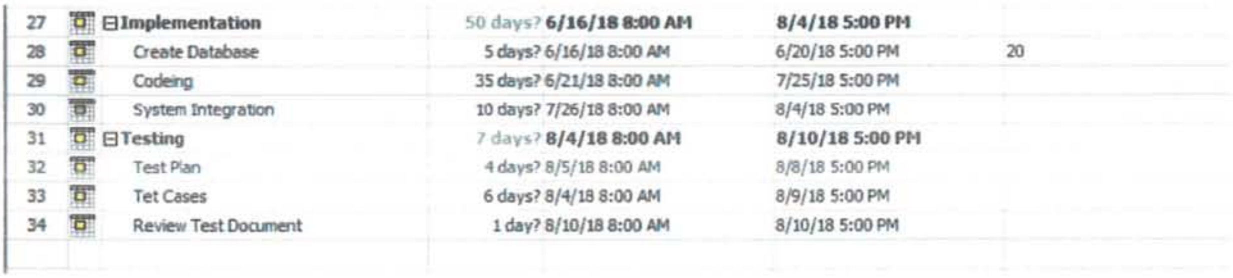
This section describes the timetable of the project as how the project tasks would be completed.

1.4.1 Time Table



		Name	Duration	Start	Finish	Predecessors
1		Problem Understanding	1 day?	5/1/18 8:00 AM	5/1/18 5:00 PM	
2		**⊟Project Plan**	5 days?	**5/2/18 8:00 AM**	**5/6/18 5:00 PM**	
3		Introduction	1 day?	5/2/18 8:00 AM	5/2/18 5:00 PM	
4		Project Organization	2 days?	5/3/18 8:00 AM	5/4/18 5:00 PM	1
5		Project Management Plan	1 day?	5/5/18 8:00 AM	5/5/18 5:00 PM	
6		Milestone:Project Plan	1 day?	5/6/18 8:00 AM	5/6/18 5:00 PM	
7		**⊟Analysis and Requirement**	25 days?	**5/7/18 8:00 AM**	**5/31/18 5:00 PM**	
8		**⊟Requirement Analysis**	10 days?	**5/7/18 8:00 AM**	**5/16/18 5:00 PM**	
9		Define Requirement	4 days?	5/7/18 8:00 AM	5/10/18 5:00 PM	1
10		Review case study	6 days?	5/11/18 8:00 AM	5/16/18 5:00 PM	
11		**⊟Develop SRS**	15 days?	**5/17/18 8:00 AM**	**5/31/18 5:00 PM**	
12		**⊟Identify Requirement**	10 days?	**5/17/18 8:00 AM**	**5/26/18 5:00 PM**	
13		External Interface Requirement	1 day?	5/17/18 8:00 AM	5/17/18 5:00 PM	9
14		Software product features	2 days?	5/18/18 8:00 AM	5/19/18 5:00 PM	
15		Use case diagram	4 days?	5/20/18 8:00 AM	5/23/18 5:00 PM	9;10
16		Software system attributes	2 days?	5/24/18 8:00 AM	5/25/18 5:00 PM	
17		Review Requirement	1 day?	5/26/18 8:00 AM	5/26/18 5:00 PM	
18		Milestone:SRS	1 day?	5/27/18 8:00 AM	5/27/18 5:00 PM	
19		Deliverabe:SRS	2 days?	5/28/18 8:00 AM	5/29/18 5:00 PM	
20		Domain Model	2 days?	5/30/18 8:00 AM	5/31/18 5:00 PM	15
21		**⊟System Design**	15 days?	**6/1/18 8:00 AM**	**6/15/18 5:00 PM**	
22		Create Sequence Diagram	2 days?	6/1/18 8:00 AM	6/2/18 5:00 PM	15
23		Create Class Diagram	2 days?	6/2/18 8:00 AM	6/3/18 5:00 PM	19
24		Develop Interface	10 days?	6/4/18 8:00 AM	6/13/18 5:00 PM	
25		Milestone:SD and DCD	1 day?	6/14/18 8:00 AM	6/14/18 5:00 PM	
26		Deliverable: System Design	1 day?	6/15/18 8:00 AM	6/15/18 5:00 PM	

Figure 1.1 Implementation Time table (I)



27		⊟Implementation	50 days?	6/16/18 8:00 AM	8/4/18 5:00 PM	
28		Create Database	5 days?	6/16/18 8:00 AM	6/20/18 5:00 PM	20
29		Codeing	35 days?	6/21/18 8:00 AM	7/25/18 5:00 PM	
30		System Integration	10 days?	7/26/18 8:00 AM	8/4/18 5:00 PM	
31		⊟Testing	7 days?	8/4/18 8:00 AM	8/10/18 5:00 PM	
32		Test Plan	4 days?	8/5/18 8:00 AM	8/8/18 5:00 PM	
33		Tet Cases	6 days?	8/4/18 8:00 AM	8/9/18 5:00 PM	
34		Review Test Document	1 day?	8/10/18 8:00 AM	8/10/18 5:00 PM	

Figure 1.2 Implementation Time table (II)

1.4.2 Gantt Chart

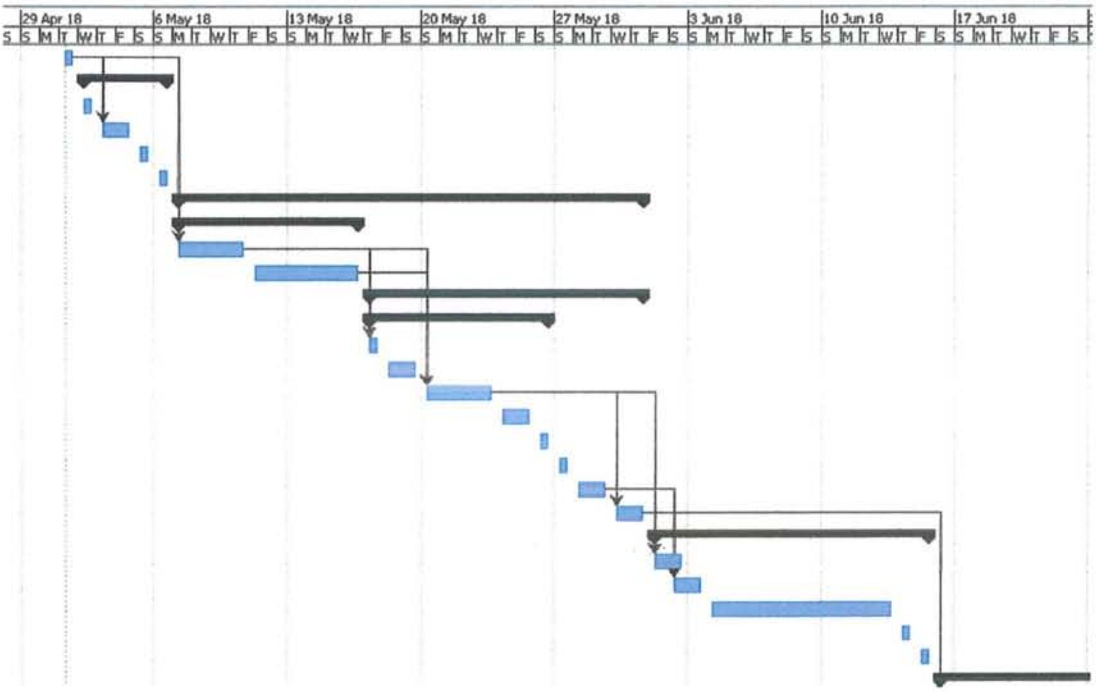


Figure 1.3 Implementation Gantt chart (I)

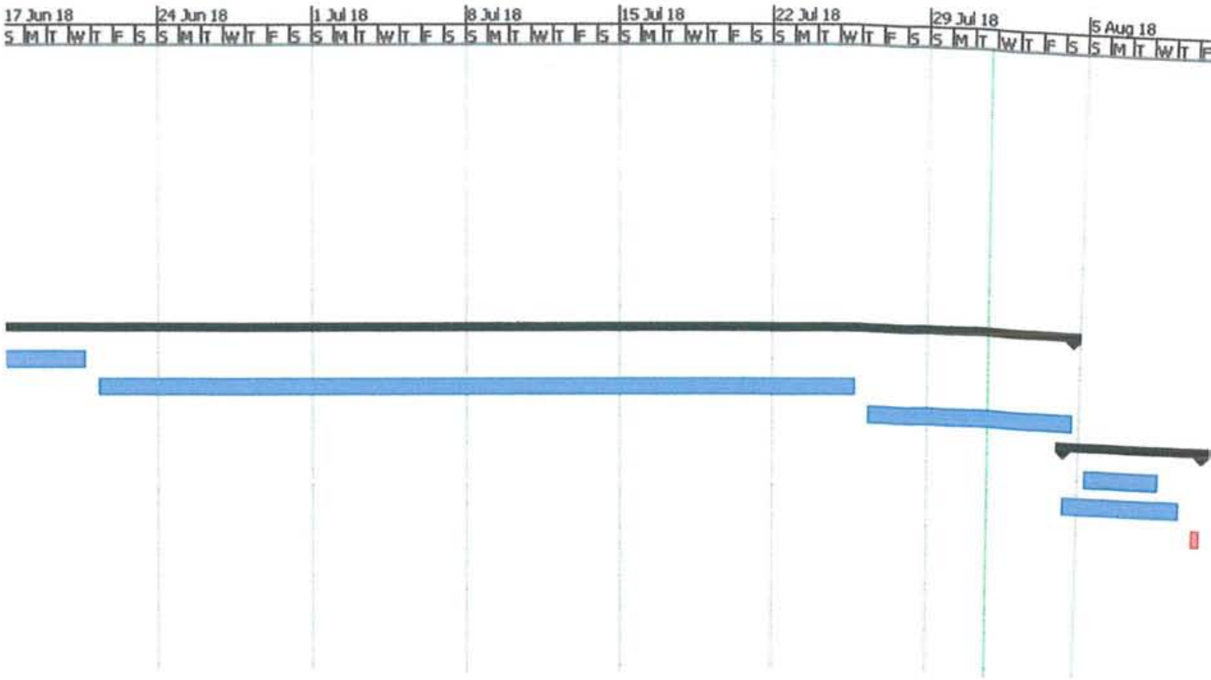


Figure 1.4 Implementation C

Chapter 2

Software Requirement Specification

2.1 Introduction

This chapter provides an overview of Software Requirement Specifications (SRS) for the subject of Event Notification App. The Event Notification App is android based system. This chapter includes a problem statement, motivation, related work, project overview, project scope and objectives. This chapter also describes functional and non-functional requirements of the proposed system.

2.1.1 Problem Statement

Finding a particular type of event is time-consuming. Sometimes users don't get an update about an event happening so they missed the event. Sometimes user can't attend an event due to time availability. Users have to search the specific place of event explicitly.

The purpose behind this project is to develop an event notification application. Users of this application can easily attend their events. They can get event reminder notification before happening. They can get upcoming events notification and users can show their interest to attend it. They can search event of their interest. They can get notify about nearby events.

2.1.2 Motivation

The motivation behind the proposed project is to facilitate users to get the news about latest events that have to be held in nearby places in Islamabad and provide them a platform where they can get information of all events from multiple sources.

2.2 Product Overview

The Event Notification App is an android based App which will facilitate users to find events in Islamabad. The user can easily find events of his/her interest. It will provide all upcoming events in Islamabad and user can register for the event to attend it. The user can search event and select their interest. User can get events details of specific hours and date. Furthermore, it will maintain a profile of each user and all information about user's event is stores there. User invites his/her friend to the event. The user gets notification about nearby events. The user gets alerts of their events before event happening. It will ask the user for the rating of events he/she has been attended. App first crawls data from different websites related to the events and then show them in the app according to the user's requirements. Admin can remove users from the system and set crawling time.

2.2.1 Project Scope

- Event Manager has 2 types of users
    - ✓ User
    - ✓ Admin
- App will first crawls data from the web.
- User can login.
- User can create their account.
- Users invite their friends to the event.
- User can view and update their profile.
- User can search for the events in Islamabad.
- User can also register for the events.
- User will get notifications about nearby events.
- User will get event reminder notification about an event he/she will be attending.
- Admin can start, stop and set time for crawling.
- Ranking about event.
- Admin can remove user from the system.

2.3 Related Work

Different apps related to event management are Eventbrite, DoStuff and All Events in city etc. provide different functionality to users. Here are some details related to existing apps. [1] [2]

- ❖ App
    - Eventbrite
- ❖ Features
    - Login (Facebook, Gmail).
    - Create user's profile.

- Search events.
- Gets user's interest.
- Register user for events.
- Create and host events.

❖ App

- Songkick

❖ Features

- User's Login (Facebook).
- Create user's profile.
- Only search for the concerts.
- Gets user's favorite artists.
- Give notification about nearby concerts of your favorite artist.

❖ App

- All Events in City

❖ Features

- Login (Facebook, Gmail).
- Create use's profile.
- Search events.
- Gets user's interest.
- Notification about nearby events.
- Create events.
- Follow organizer.
- Chatting with user and organizer.

❖ App

- DoStuff

❖ Features

- User's login (Facebook).
- Create user's profile.
- Create events.
- Search upcoming events by category and location (only 20 cities in North America).

Different apps like All Events in city, Eventbrite, DoStuff and SongKick etc. provide event management including features like login, register for the event, chat with other users and search nearby events.

2.3.1 Comparison of Event Manager Apps

App Name	Properties									
	Login	Create Profile	Create Events	Chatting	Invite Friends	Follow organizer	Search	Events Category	Search Events by location	Register for event
All Events in City	×	×	×	×	×	×	×	×	×	
EventBrite	×	×	×				×	×	×	×
DoStuff	×	×	×				×	×	×	
Songkick	×	×					×	×		
10Times	×	×		×	×		×	×	×	×

Table 2.3.1: Comparison of Different Applications

2.4 Specific Requirement

This section describes detail description of Interface requirement.

2.4.1 External Interface Requirement

User Interface

- The interface will be user-friendly.
- There will be two interfaces, one for users and other for the admin.

Hardware Interface

- Laptop(Computer)
- Mobile phone(Android)

Software Interface

- Android studio with API 23: Android 6.0 (Marshmallow).
- Windows operating system (for web)

2.5 Use Case Diagram

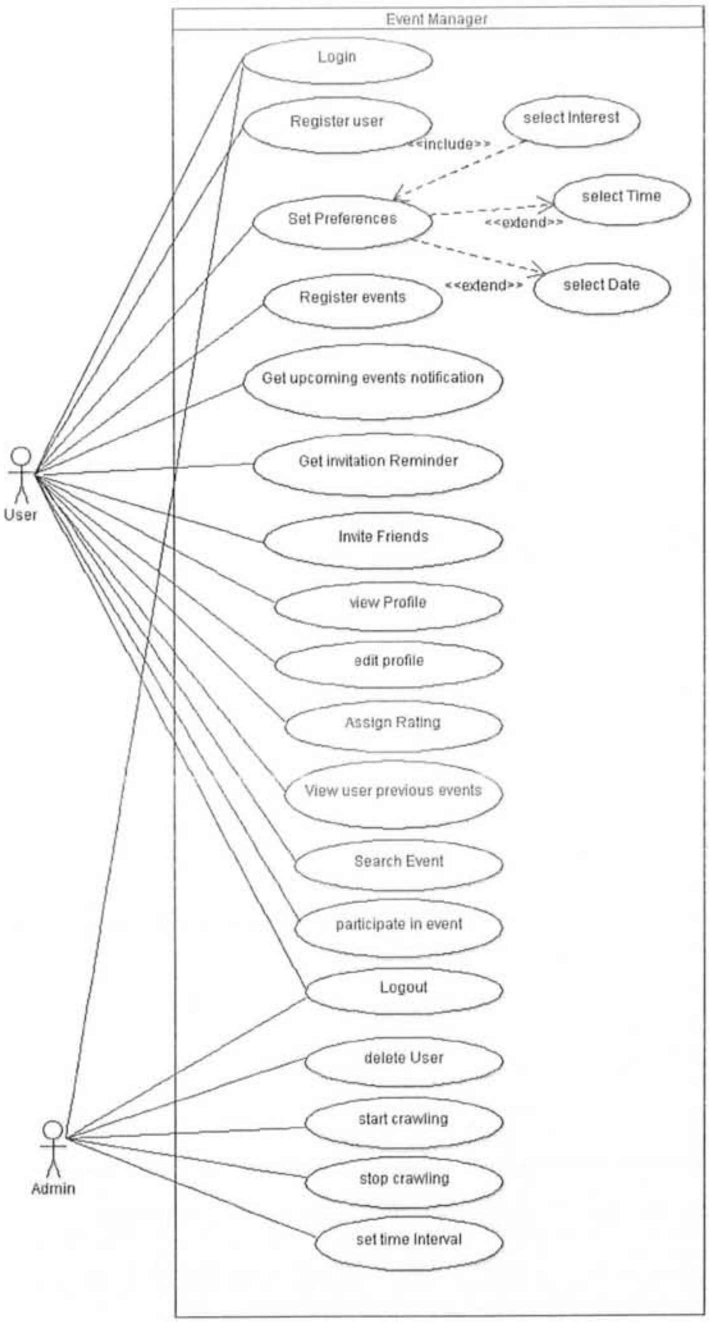


Figure 2.5 Use case diagram

2.6 Use Case description

UC-1: Login

Table 2.2 UC-1: Login

UC-1:Login	
Primary actor	User ,Admin
Stakeholders & Interests	**User:** will be able to login into the system. **Admin:** will be able to login into the system.
Pre-condition	Admin is already registered and has username and password.
Post-condition	Users will login successfully.
Main Success Scenario	1) User enters username and password. 2) User select radio button of user. 3) Admin enters username and password. 4) Admin select radio button of Admin. 5) Click login button. 6) System will show profile.
Alternative Scenario	1a) User enters wrong username or password. a. System will prompt user to enter correct username/password. 5a) User submits form without filing all required fields. a) System prompts user to fill all required fields. 6b) Server is down/No internet connection. a) Systemwaits until server works properly.
Special Requirements	Android phone with internet connection.
Technology	1. Internet connection. 2. Android phone. 3. Database server is connected.

UC-2: Register User

Table 2.3 UC-2:Register user

UC-2:Register user	
Primary actor	User
Stakeholders & Interests	**User:** will be able to logout account.
Post-condition	User has been registered successfully.
Main Success Scenario	1. User presses "create account" button. 2. User fills required fields. 3. User presses sign up button. 4. System checks validations and create user account.
Alternative Scenario	3a) User submits form without filing all required fields a) System prompts user to fill all required fields. b) System prompt user to enter valid entries. 4a) Server is down/no internet connection. 4b) System waits until server works properly.
Special Requirements	None.
Technology	1. Internet connection. 2. Android phone. 3. Database server is connected.
Frequency	Many times a day.

UC-3: Logout

Table 2.4 UC-3: Logout

UC-3:Logout	
Primary actor	User(User, Admin)
Stakeholders & Interests	**User:** and Admin will be able to logout account.
Pre-condition	User must be logged in.
Post-condition	Userhas been logged out successfully.
Main Success Scenario	5. User presses "Logout" button. 6. System logs the user out and invalidates the session.
Alternative Scenario	1a) Server is down/no internet connection. 1b) System waits until server works properly.
Special Requirements	None.
Technology	4. Internet connection. 5. Android phone. 6. Database server is connected.
Frequency	Many times a day.

UC-4: Invite friends

Table 2.5 UC-4: invite friends

UC-4:Invite friends	
Primary actor	User
Stakeholders & Interests	**User:** invites friend for the event to attend.
Pre-condition	User must be login.
Post-condition	User will send invitation to his/her friend successfully.
Main Success Scenario	1. User selects event. 2. User selects invite friend option from the menu. 3. App display form. 4. User enters friend's email address. 5. User presses invite button.
Alternative Scenario	2a) Server is down/no internet connection. 2b) System waits until server works properly. 4a) No internet connection. 4b) System waits until connection re-established.
Special Requirements	None.
Technology	1. Internet connection. 2. Android phone. 3. Database server is connected.
Frequency	Many times a day.

UC-5: Select interests

Table 2.6 UC-5: Select interest

UC-5:Select interest	
Primary actor	User
Stakeholders & Interests	**User:** selects his/her interest for events.
Post-condition	User will select and save interests successfully.
Main Success Scenario	1. User selects one or multiple interests from the menu. 2. User presses ok button to continue. 3. System will save user's interest in user's profile.
Alternative Scenario	2a) No internet connection. 2b) Configure an internet connection. 3a) Server is down. 3b) Wait until server workers properly.
Special Requirements	None.
Technology	1. Internet connection. 2. Android phone. 3. Database server is connected.
Frequency	At the time of logging into the app.

UC-6: Select time

Table 2.7 UC-6:Select Time

UC-6:Select Time	
Primary actor	User
Stakeholders & Interests	**User:** selects time for events.
Pre-condition	User must be login.
Post-condition	User will select time successfully.
Main Success Scenario	1. User open upcoming events. 2. User selects time. 3. System will show all events happening at that time.
Alternative Scenario	2a) No internet connection. 2b) Configure an internet connection. 3a) Server is down. 3b) Wait until server workers properly.
Special Requirements	None.
Technology	1. Internet connection. 2. Android phone. 3. Database server is connected.
Frequency	At the time of logging into the app.

UC-7: Select Date

Table 2.8 UC-7:select Date

UC-7:Select Date	
Primary actor	User
Stakeholders & Interests	**User:** selects date for events and will see upcoming events on particular date.
Pre-condition	User must be login.
Post-condition	User will select date successfully.
Main Success Scenario	1. User open upcoming events. 2. User selects time. 3. System will show all events happening on particular date.
Alternative Scenario	2a) No internet connection. 2b) Configure an internet connection. 3a) Server is down. 3b) Wait until server workers properly.
Special Requirements	None.
Technology	1. Internet connection. 2. Android phone. 3. Database server is connected.
Frequency	At the time of logging into the app.

UC-8: Search Event

Table 2.9 UC-8: Search event

UC-8:Search Event	
Primary actor	User
Stakeholders & Interests	**User:** search specific event according to the requirements.
Pre-condition	User must be logged in.
Post-condition	User searched required event successfully.
Main Success Scenario	1. User select search option from the menu. 2. User enters query. 3. User presses search button. 4. System will show searched events.
Alternative Scenario	3a) Server is down/no internet connection. 3b) Configure an internet connection. 3c) Wait until server workers properly.
Special Requirements	None.
Technology	1. Internet connection. 2. Android phone. 3. Database server is connected.
Frequency	Many times in a day.

UC-9: Register for the event

Table 2.10 UC-9: Register for event

UC-6:Register for the event	
Primary actor	User
Stakeholders & Interests	**User:** want to register for the particular event
Pre-condition	User must be logged in and searched an even to attend.
Post-condition	User register to attend event successfully.
Main Success Scenario	1. User selects an event. 2. User presses register event button. 3. System directs user to organizer's page for registration.
Alternative Scenario	5a) Server is down/no internet connection. a. Configure an internet connection. b. Wait until server workers properly.
Special Requirements	None.
Technology	1. Internet connection. 2. Android phone. 3. Database server is connected.
Frequency	Many times in a day.

UC-10: Participate in event

Table 2.11 UC-10: Participate in event

UC-10:Participate in event	
Primary actor	User
Stakeholders & Interests	**User:** will be able to choose participation in event (user is going or not).
Pre-condition	User must be logged in.
Post-condition	User chooses their participation successfully.
Main Success Scenario	1. User selects a particular event. 2. User views the event details. 3. User press going/not going button. 4. System store user's choice in profile. 5. System store event in user's registered event.
Alternative Scenario	1a) Server is down/no internet connection. 1b) System waits until server workers properly. 1c) Configure the internet connection.
Special Requirements	None.
Technology	1. Internet connection. 2. Android phone. 3. Database server is connected.
Frequency	Many times a day.

UC-11: Get invitation reminder

Table 2.12 UC-11: Get invitation reminder

UC-8:Get invitation reminder	
Primary actor	User
Stakeholders & Interests	**User:** will get notified about his/her today's event.
Pre-condition	User must be logged in and register for event to attend.
Post-condition	User get reminder for the event.
Main Success Scenario	1. User gets notification about the event happening for which he/she is registered. 2. User can view notification on notification bar.
Alternative Scenario	1a) Server is down/no internet connection. a) Configure an internet connection and wait until server workers properly. b) User gets notification.
Special Requirements	None.
Technology	1. Internet connection. 2. Android phone. 3. Database server is connected.
Frequency	Depends upon the need of a user.

UC-12: View Profile

Table 2.13 UC-12 View Profile

UC-12:View Profile	
Primary actor	User
Stakeholders & Interests	**User:** will be able to view profile details.
Pre-condition	User must be logged in.
Post-condition	User views the profile details successfully.
Main Success Scenario	1. User selects the view profile option from the menu. 2. User views the Profile details.
Alternative Scenario	1a) Server is down/no internet connection. 1b) System waits until server workers properly.
Special Requirements	None.
Technology	1. Internet connection. 2. Android phone. 3. Database connection.
Frequency	More frequently.

UC-13: Edit Profile

Table 2.14 UC-13 Edit Profile

UC-13:Edit Profile	
Primary actor	User
Stakeholders & Interests	**User:** will be able to edit profile information.
Pre-condition	User must be logged in.
Post-condition	User updates the profile successfully.
Main Success Scenario	1. User selects the edit profile option from the menu. 2. User modifies the profile information. 3. User presses "Save" button. 4. System will stores the modified account details into the database.
Alternative Scenario	2a) User dismisses the changes by pressing "Cancel" button. 3a) Server is down/no internet connection. 3b) System waits until server workers properly.
Special Requirements	None.
Technology	1. Internet connection. 2. Android phone. 3. Database server is connected.
Frequency	Less frequently.

UC-14: Upcoming events Notification

Table 2.15 UC-14: Upcoming events notifications

UC-14:Upcoming events Notification	
Primary actor	User
Stakeholders & Interests	**User:** will be able to get upcoming events notifications.
Pre-condition	User must be logged in.
Post-condition	User gets nearby upcoming events notifications successfully.
Main Success Scenario	1. User selects notification option from menu. 2. User views the notification details.
Alternative Scenario	a) Server is down/no internet connection. b) System waits until server workers properly. c) Configure the internet connection.
Special Requirements	None.
Technology	1. Internet connection. 2. Android phone. 3. Database server is connected.
Frequency	Many times a day.

UC-15: Assign rating

Table 2.16 UC-15: assign rating

UC-15: assign rating	
Primary actor	User
Stakeholders & Interests	**User:** will be able to rate events after attending.
Pre-condition	User must be logged in and register for the event.
Post-condition	User rate event by giving stars about particular event.
Main Success Scenario	1. User opens an event he/she has been attended. 2. User selects stars from the available rating bar. 3. System stores that data in the database. 4. System will rank events of that organizer on top according to rating.
Alternative Scenario	2a) User dismisses the rating bar and continues. 3a) Server is down/no internet connection. 3b) System waits until server workers properly. 3c) Configure the internet connection.
Special Requirements	Android phone with internet connection.
Technology	1. Internet connection. 2. Android phone. 3. Database server is connected.
Frequency	Less frequently.

UC-16: View previous events

Table 2.17 UC-16:View previous events

UC-16: View previous events	
Primary actor	User
Stakeholders & Interests	**User:** will be able to view previous events he/she has been attended.
Pre-condition	User must be logged in and register for event.
Post-condition	User views their events history successfully.
Main Success Scenario	1. User selects view profile from menu. 2. User selects their events. 3. System will show user's previous event.
Alternative Scenario	a) Server is down/no internet connection. b) System waits until server workers properly. c) Configure the internet connection.
Special Requirements	None.
Technology	1. Internet connection. 2. Android phone. 3. Database server is connected.
Frequency	Less frequently.

UC-17: Delete User

Table 2.18 UC-17 Delete User

UC-17: Delete User	
Primary actor	Admin
Stakeholders & Interests	**Admin:** will be able to Delete/Remove User from the system.
Pre-condition	Admin must be logged in into their account.
Post-condition	Admin remove user successfully.
Main Success Scenario	1. Admin open his profile. 2. Admin selects user. 3. Admin click on "Remove". 4. System deletes user information from the database.
Alternative Scenario	a) Server is down/no internet connection. b) System waits until server workers properly. c) Configure the internet connection.
Special Requirements	None.
Technology	1. Internet connection. 2. Android phone. 3. Database server is connected.
Frequency	Less frequently.

UC-18: Start Crawling

Table 2.19 UC-18: Start crawling

UC-18: Start crawling	
Primary actor	Admin
Stakeholders & Interests	**Admin:** will be able to start crawling though web service.
Pre-condition	Admin must be logged in into their account.
Post-condition	Admin start crawling successfully
Main Success Scenario	1. Admin open his profile. 2. Admin selects start button. 3. Web service starts crawling.
Alternative Scenario	a) Server is down/no internet connection. b) System waits until server workers properly. c) Configure the internet connection.
Special Requirements	None.
Technology	1. Internet connection. 2. Android phone.
Frequency	Less frequently.

UC-19: Stop Crawling

Table 2.20 UC-19: Stop crawling

UC-19: Stop crawling	
Primary actor	Admin
Stakeholders & Interests	**Admin:** will be able to stop crawling though web service.
Pre-condition	Admin must be logged in into their account.
Post-condition	Admin stop crawling successfully
Main Success Scenario	1. Admin open his profile. 2. Admin selects stop button. 3. Web service stops crawling.
Alternative Scenario	a) Server is down/no internet connection. b) System waits until server workers properly. c) Configure the internet connection.
Special Requirements	None.
Technology	1. Internet connection. 2. Android phone.
Frequency	Less frequently.

UC-20: Set time interval

Table 2.21 UC-20: Set time interval

UC-19: Set time interval	
Primary actor	Admin
Stakeholders & Interests	**Admin:** will be able to set time for crawling.
Pre-condition	Admin must be logged in into their account.
Post-condition	Admin setup time interval for crawling successfully
Main Success Scenario	1. Admin open his profile. 2. Admin selects 'setTime' button. 3. Web service crawl data according to that time.
Alternative Scenario	a) Server is down/no internet connection. b) System waits until server workers properly. c) Configure the internet connection.
Special Requirements	None.
Technology	1. Internet connection. 2. Android phone.
Frequency	Less frequently.

2.7 Software System Attributes

2.7.1 Reliability

- System is reliable.
- It will never crash whenever user is doing desired operation.

2.7.2 Availability

For availability it requires the following conditions:

- Internet connection
- Running server
- Database connectivity

Once it will in running condition then it will available 24 hours a day under above conditions.

2.7.3 Security

- Secure database will be used which required password before connection.
- User should be login before performing any tasks.
- Each type of user will have limited access to modules.

2.7.4 Portability

Our application will run on an android phone which is a portable device so user can use this app anywhere, at any time.

2.7.5 Maintainability

During the development period, all the code will be properly documented so that we can easily make changes and upgrade our application.

2.7.6 Performance

As Event Notification App is an android based application, so its performance could be affected by the internet speed. If there is no internet problem then it will display results of query from user sides quickly. Usually results will show in millisecond or maximum 5 seconds. New screen will come quickly when user perform any action

2.8 Database Requirement

Application crawls data about events from website, so we need to store huge amount of data. For that I chose Firebase.

Firebase stores and sync data with NoSQL cloud database. Data is synced across all clients in real time, and remains available when your app goes offline. The Firebase real time Database is a cloud-hosted database. Data is stored as JSON and synchronized in real time to every connected client. [3]

2.8.1 ERD

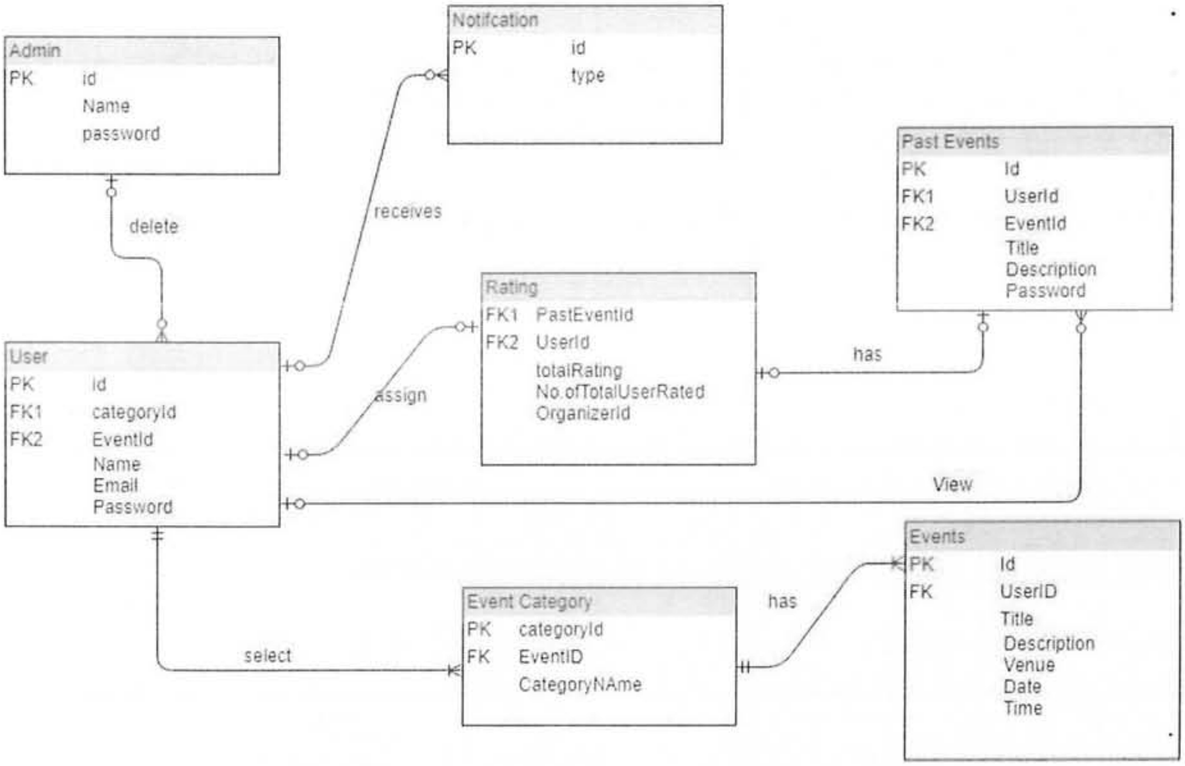


Figure 2.6 ERD

2.9 System Sequence Diagram

A system sequence diagram illustrates input and output events related to our system. System is treated as black box and the emphasis of the diagram is events that are generated by the system for a particular scenario of use-cases. [4]

Here I have given system sequence diagrams which we need to be tested.

SSD1: Login

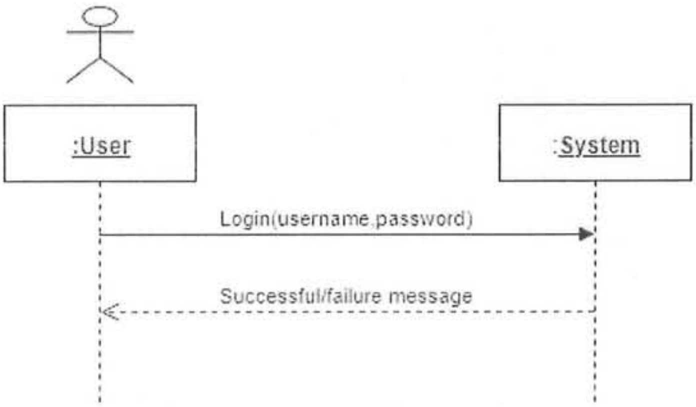


Figure 2.7 SSD-1: Login

SSD2: Register User

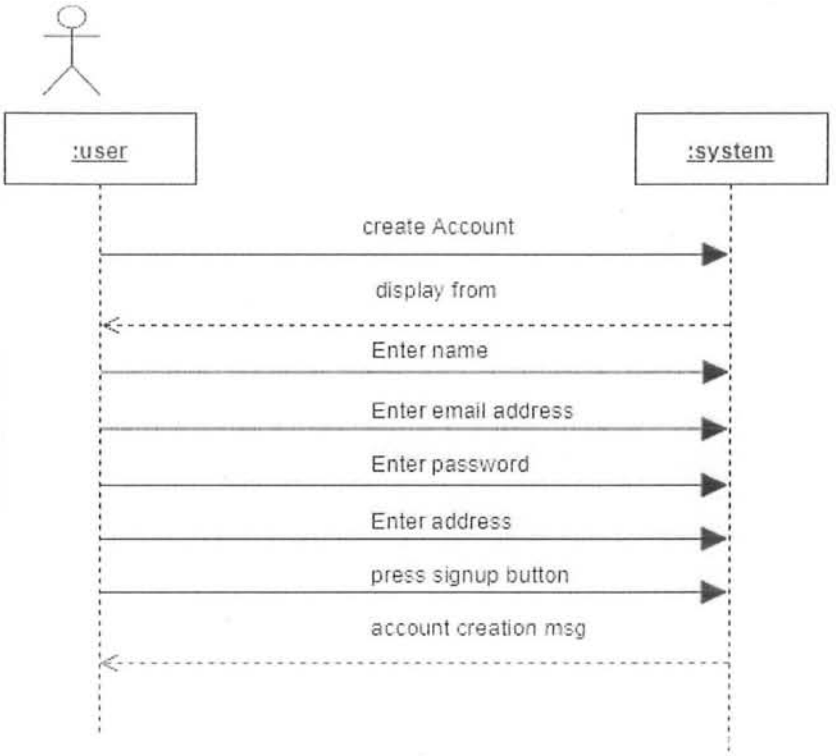


Figure 2.8 SSD-2: Register user

SSD3: Logout

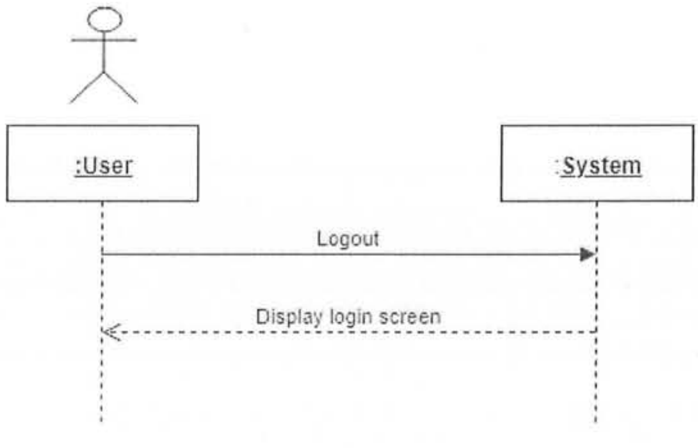


Figure 2.9 SSD-3: Logout

SSD4: Invite friends

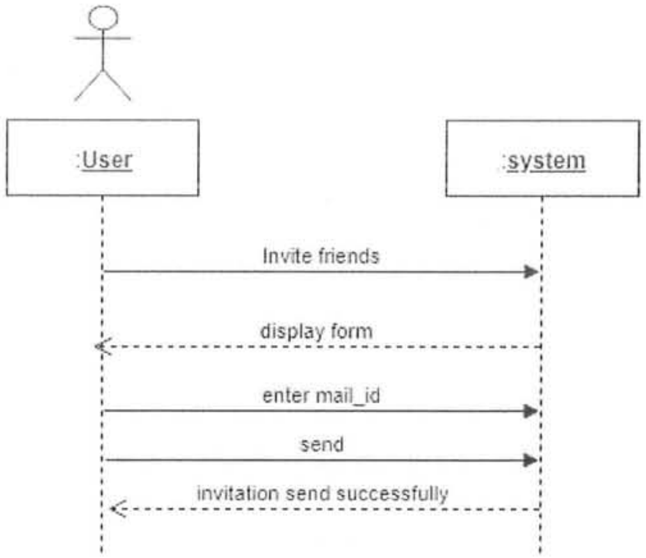


Figure 2.10 SSD-4: Invite Friends

SSD5: Select Interest

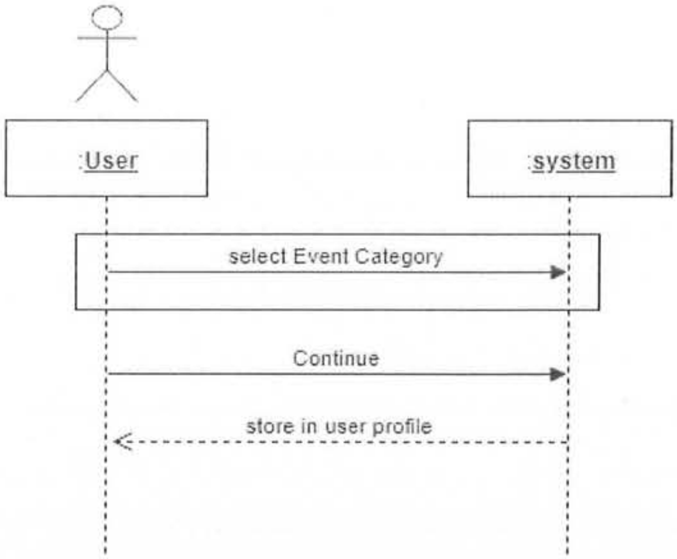


Figure 2.11 SSD-5: Select Interest

SSD6: Select Time

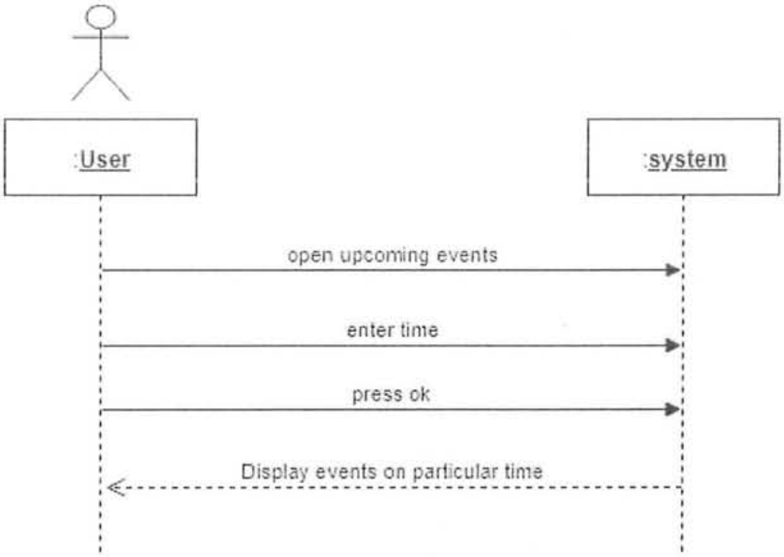


Figure 2.12 SSD-6: Select Time

SSD7: Select date

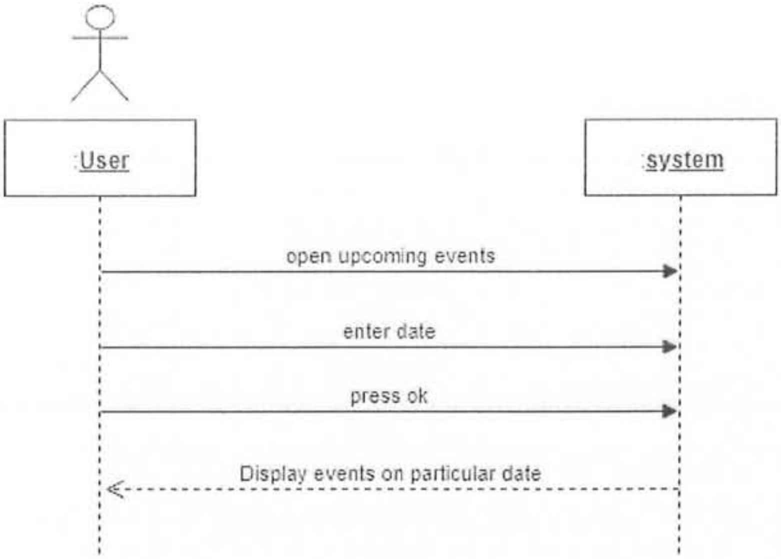


Figure 2.13 SSD-7: Select Date

SSD8: Search Events

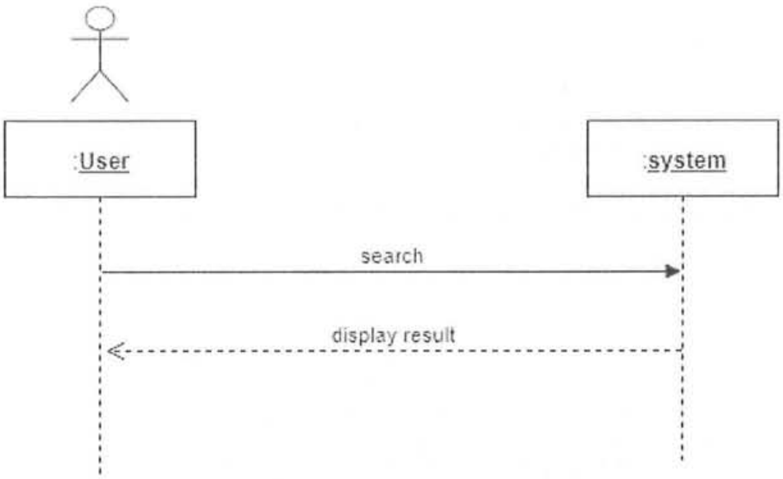


Figure 2.14 SSD-8: Search Events

SSD9: Register for Event

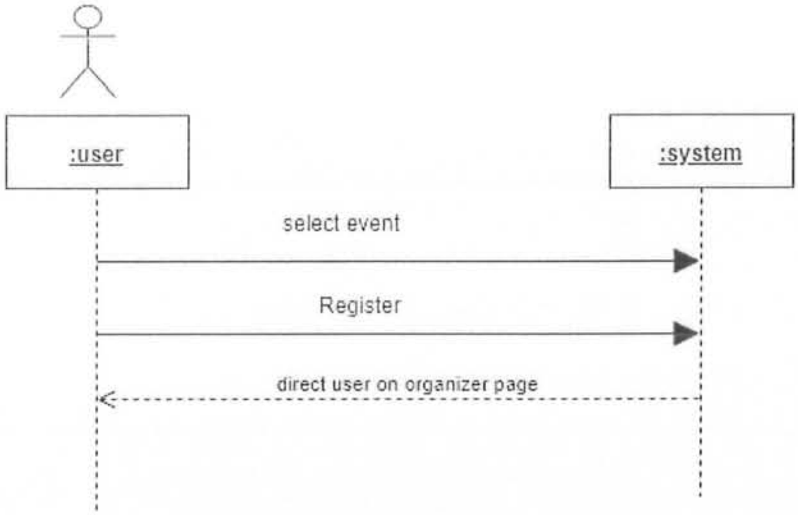


Figure 2.15 SSD-9: Register for event

SSD10: Participate in event

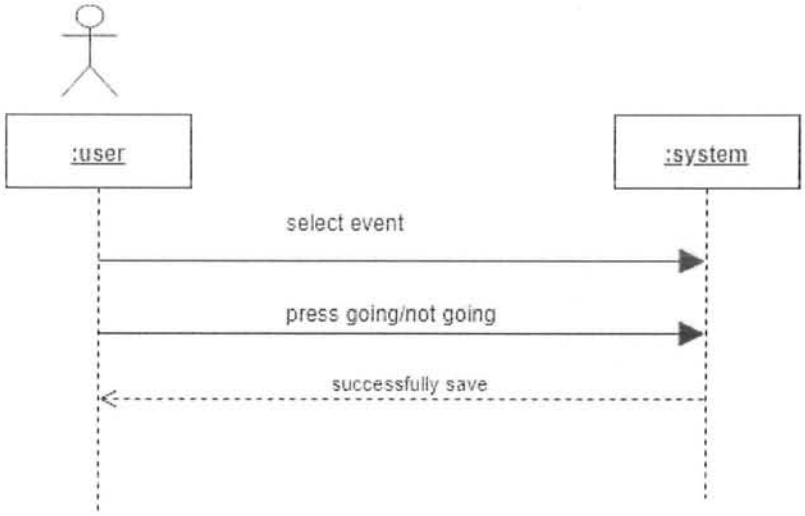


Figure 2.16 SSD-10: Participate in event

SSD11: Get Invitation Reminder

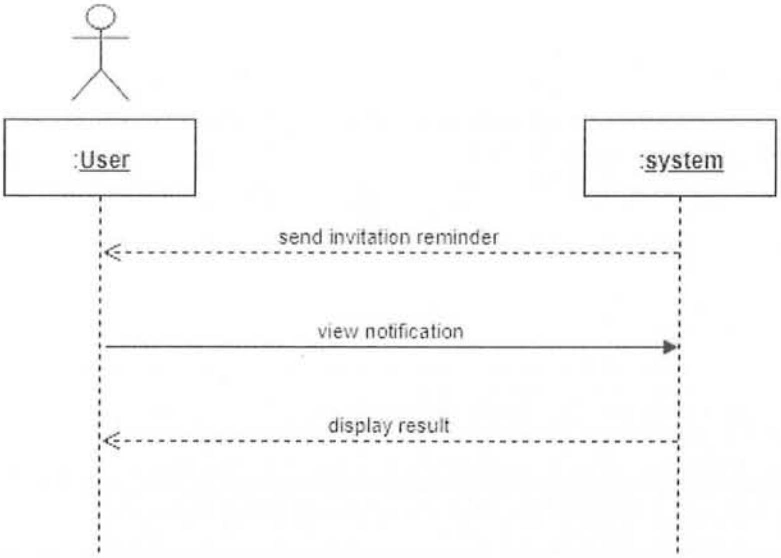


Figure 2.17 SSD-11:Get Invitation Reminder

SSD12: View Profile

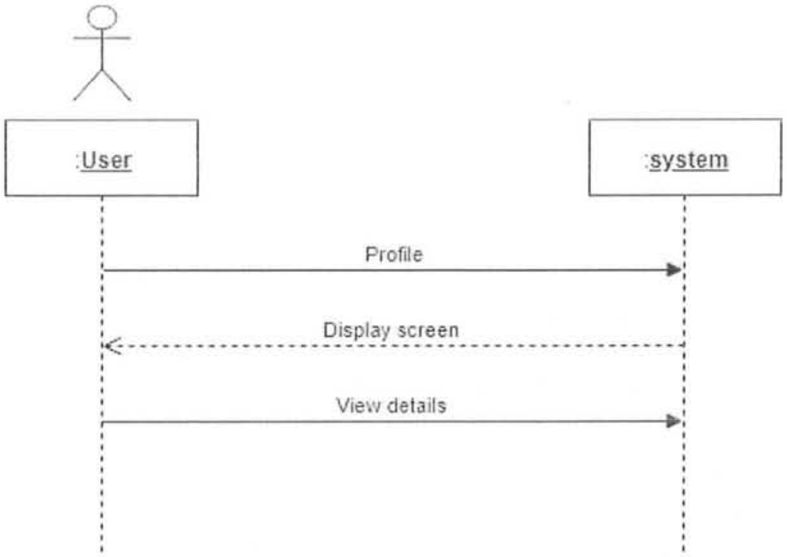


Figure 2.18 SSD-12: View Profile

SSD13: Edit Profile

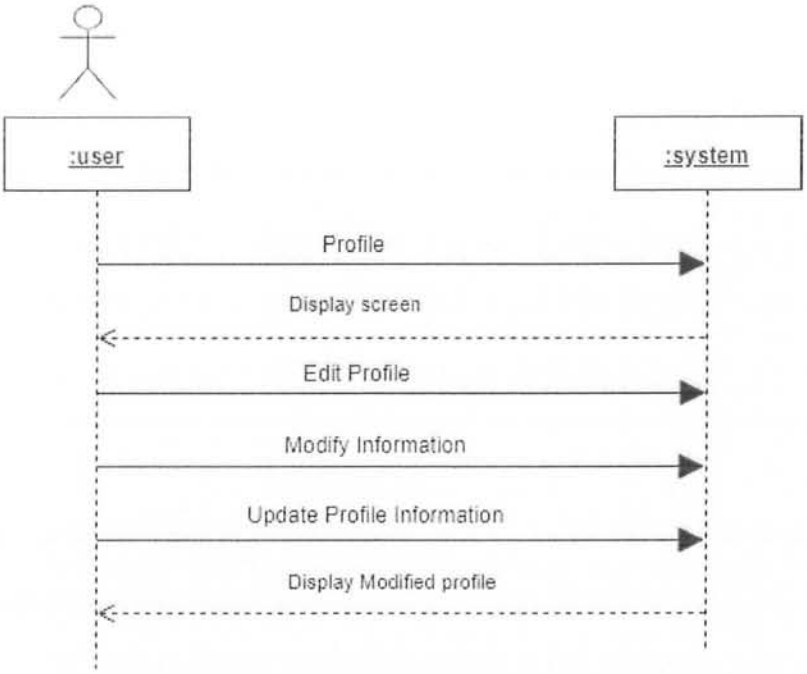


Figure 2.19 SSD-13: Edit Profile

SSD14: Upcoming Events notification

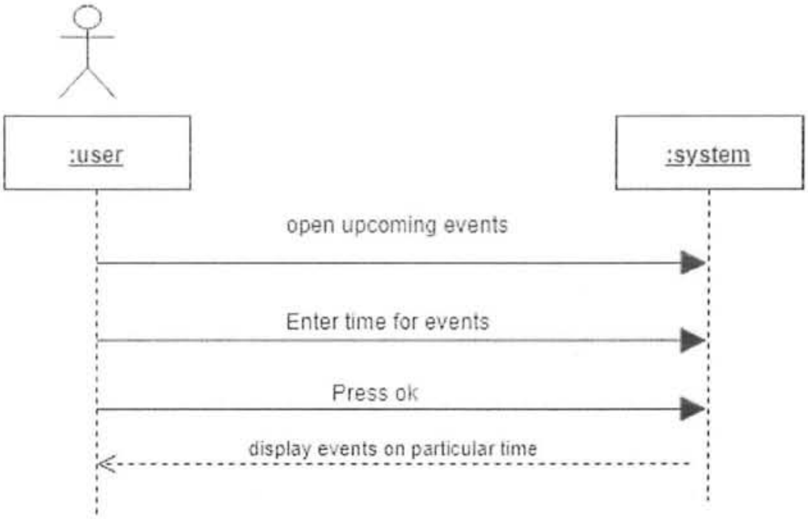


Figure 2.20 SSD-14: Upcoming Events notifications

SSD15: Assign Rating

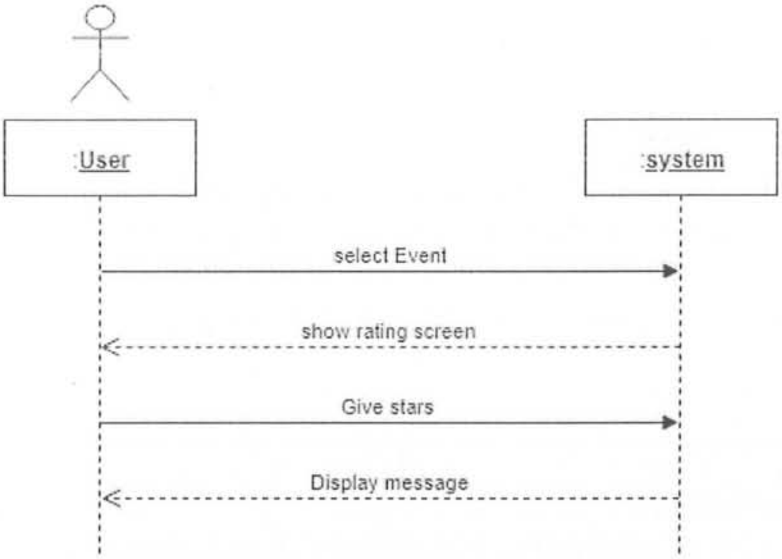


Figure 2.21 SSD-15: Assign rating

SSD16: View Previous Events

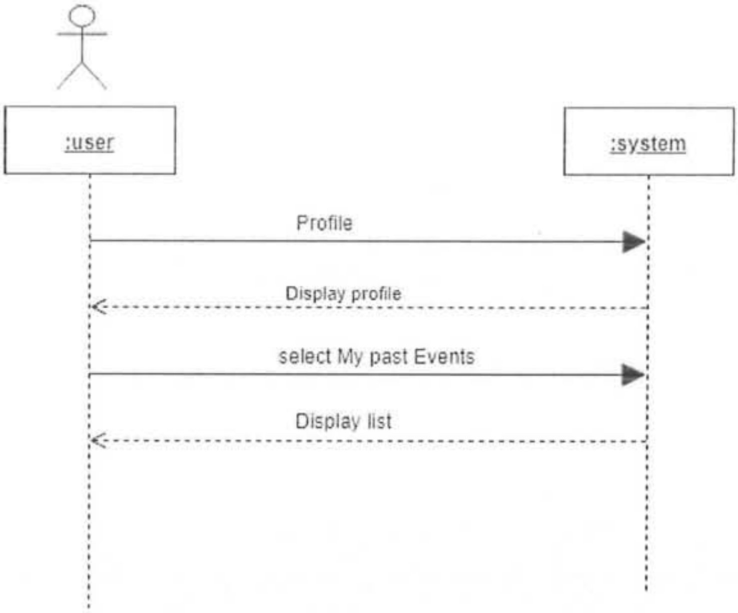


Figure 2.22 SSD-16: View previous events

SSD17: Delete User

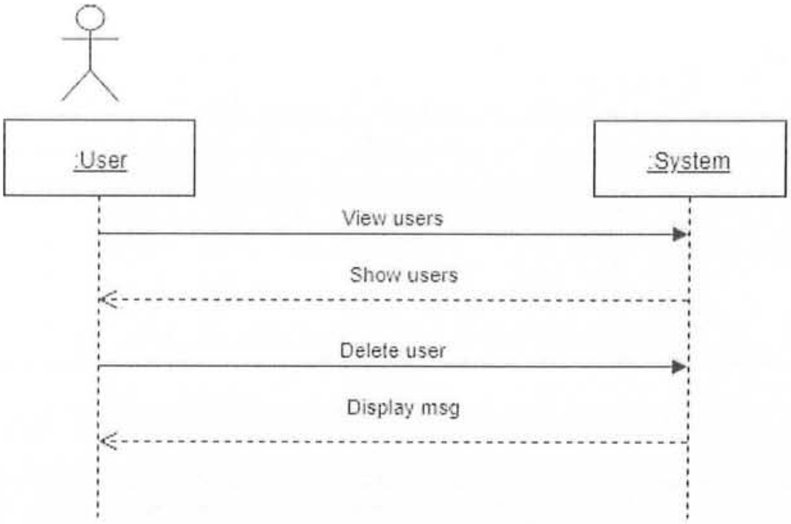


Figure 2.23 SSD-17: Delete user

SSD18: Start crawling

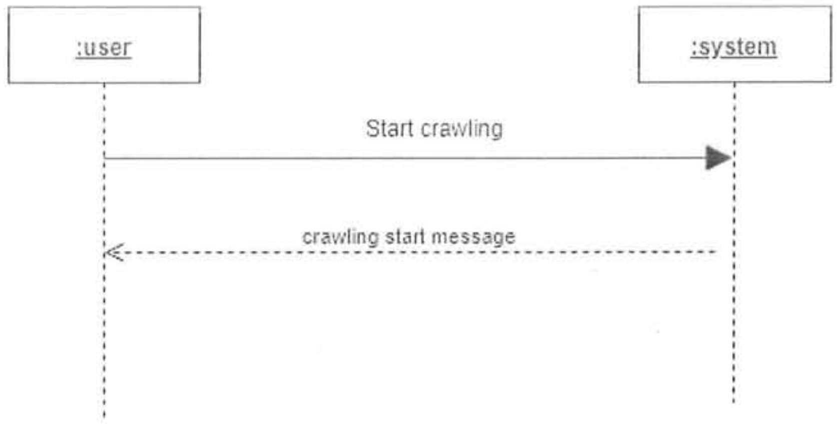


Figure 2.24 SSD-18: start crawling

SSD19: Stop crawling

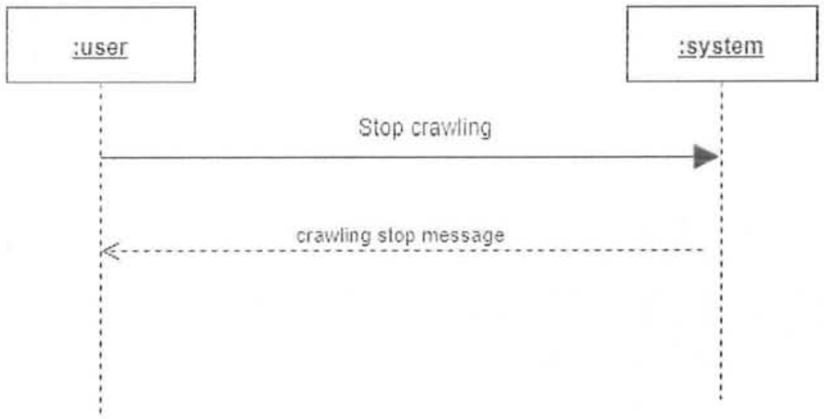


Figure 2.25 SSD-19: stop crawling

SSD20: Set time interval

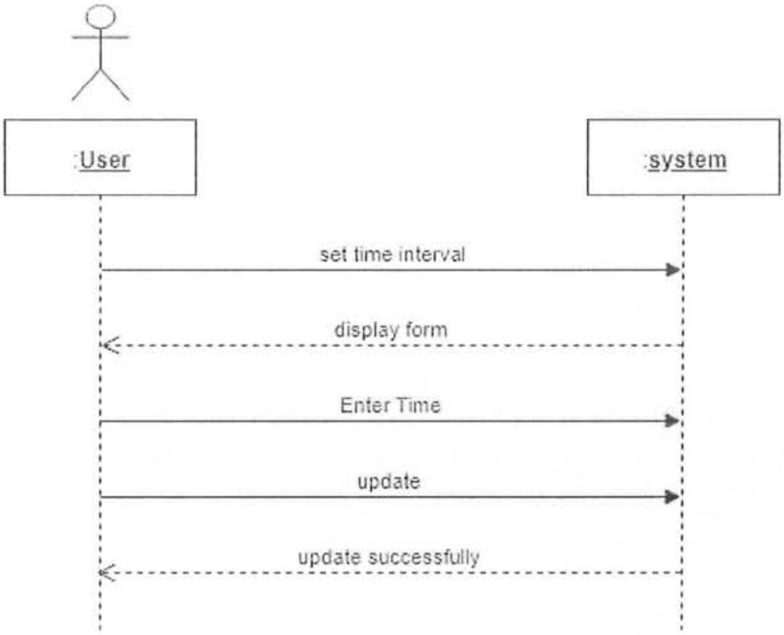


Figure 2.26 SSD-20: set time interval

2.10 Domain Model

A domain model is visual representation of real world objects in a domain of interest. Elements of domain model are domain object classes and the relationships between them. [4]

The domain model is created in order to represent the vocabulary and key concepts of problem domain. The domain model also identifies the relationships among all the entities within the scope of problem domain, and commonly identifies their attributes.

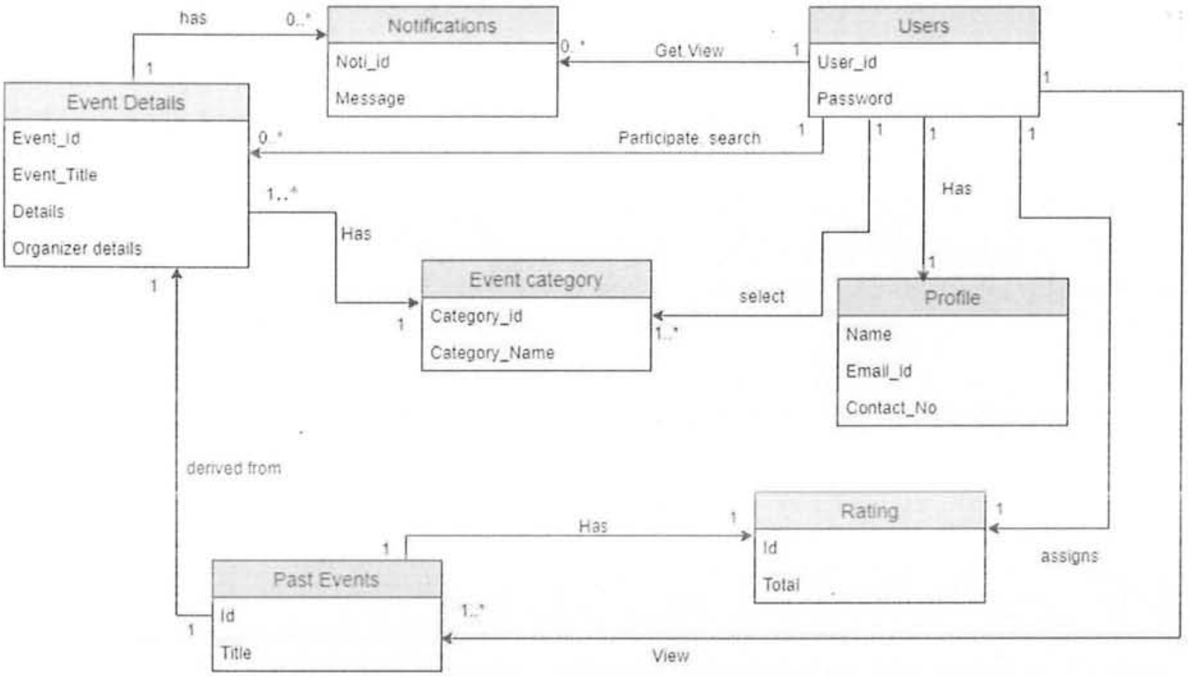


Figure 2.27 Domain Model

Chapter 3

Software Design Description

3.1 Introduction

Software design document describes the architecture and design of the system. Systems design is the process or art of defining the architecture, components, modules, interfaces, and data for a system to satisfy specified requirements.

3.1.1 Design Overview

This document describes the design details of event notification project. This section introduces and gives a brief overview of the design. It includes the following diagrams which will explain the design overview of the system.

- Architecture of system
- Sequence Diagrams
- Class Diagram

3.2 System Architectural Design

System architectural design represents the structure of the data and program components that are required to build a computer-based system. This section provides the process of defining a collection of hardware and software components and their interfaces to establish the framework for the development of a computer system.

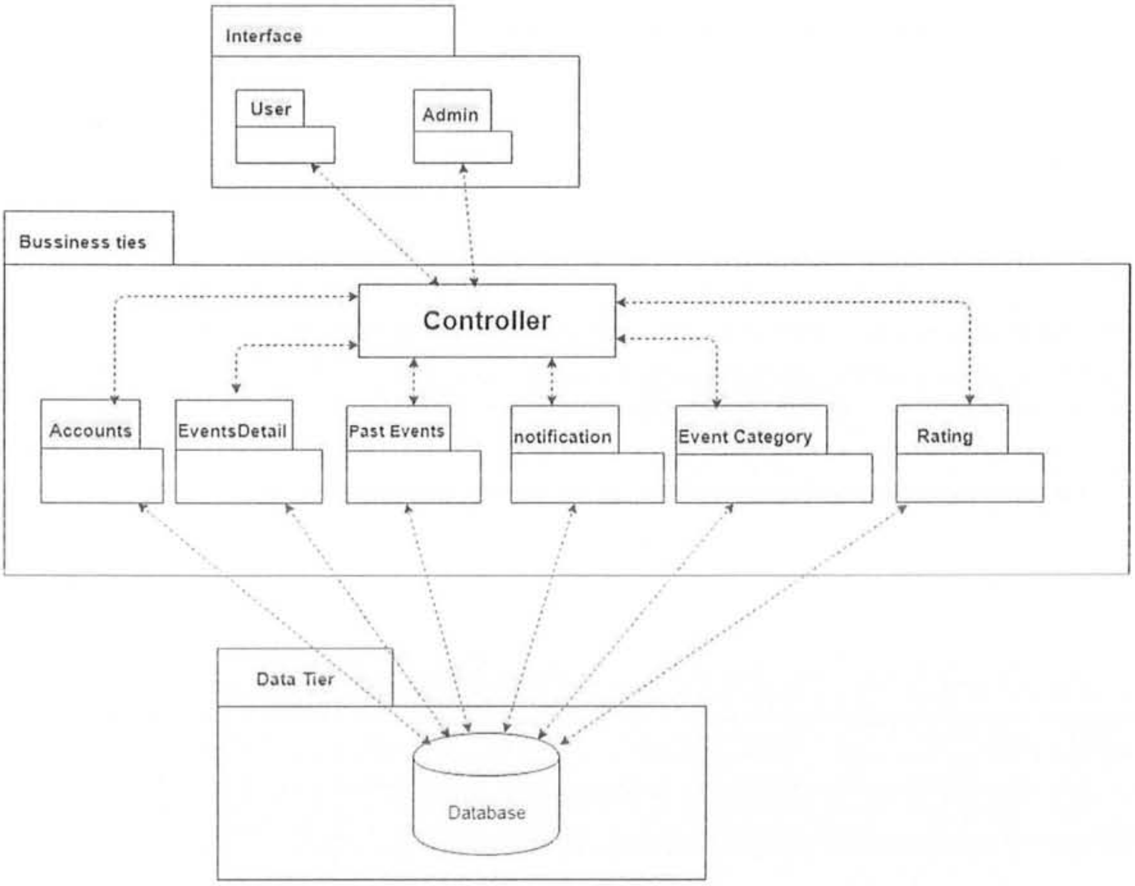


Figure 3.28 System Architectural Diagram

3.3 Class Diagram

A class diagram in the Unified Modeling Language (UML) is a type of static structure diagram that describes the structure of a system by showing the system's classes, their attributes, operations (or methods), and the relationships among objects. [4]

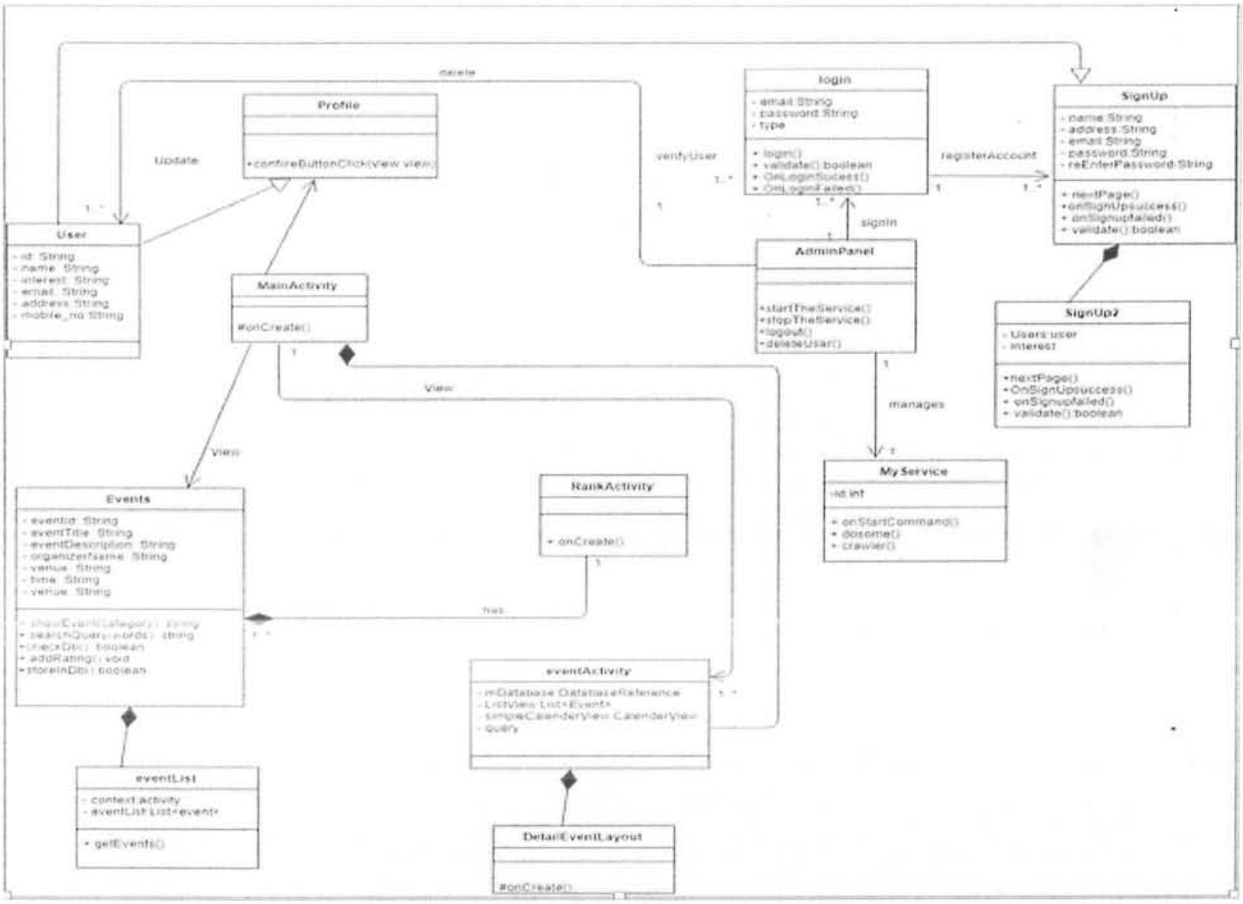


Figure 3.29 Class diagram

3.4 Sequence Diagram

A sequence diagram is an interaction diagram that shows how objects operate with one another and in what order. It is a construct of a message sequence chart. A sequence diagram shows object interactions arranged in time sequence. [5]

2.4.1 Sign in to system

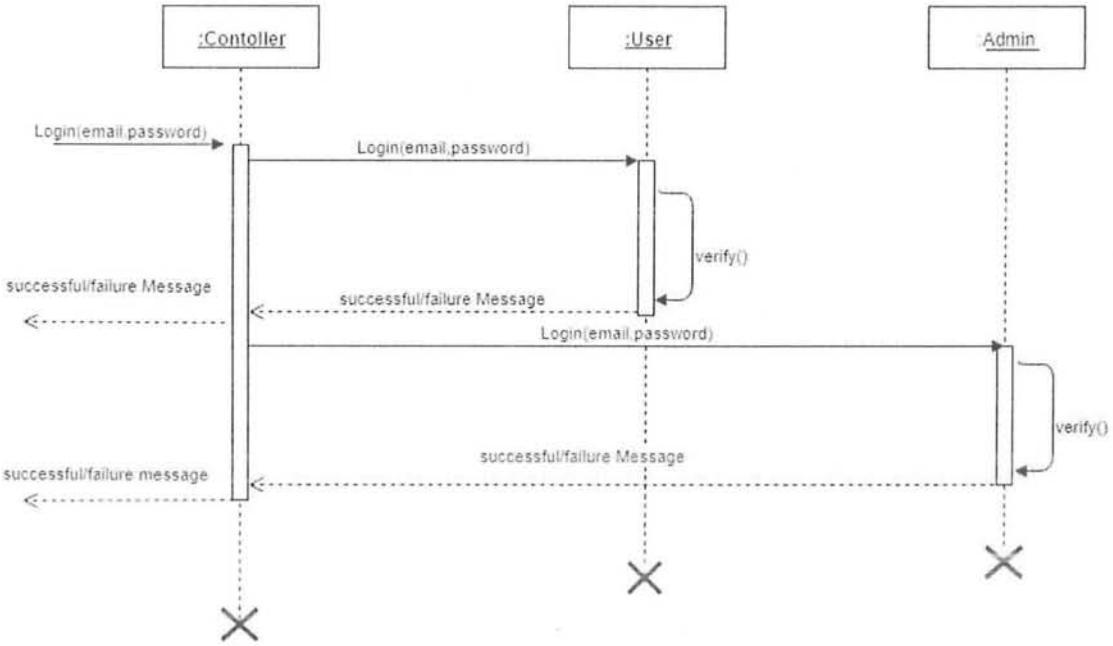

:Contoller
:User
:Admin
Login(email.password)
Login(email,password)
verify()
successful/failure Message
successful/failure Message
Login(email.password)
verify()
successful/failure Message
successful/failure message

Figure 3.30 SD-1: Login

2.4.2 Sign out from System

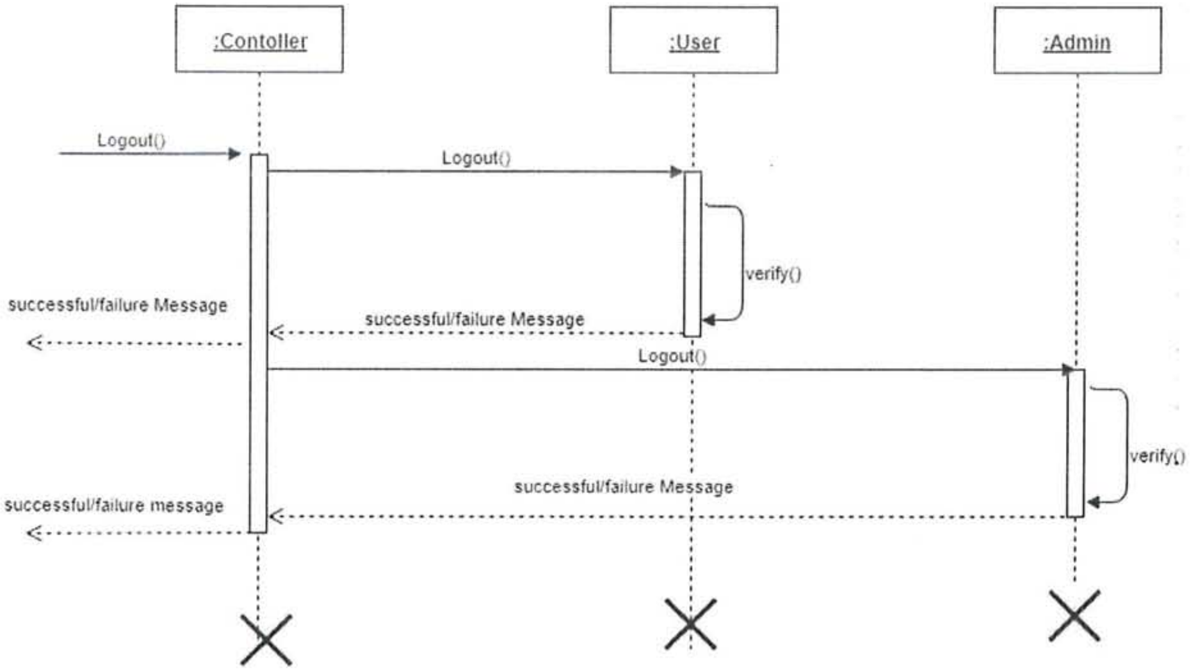

:Contoller
:User
:Admin
Logout()
Logout()
verify()
successful/failure Message
successful/failure Message
Logout()
verify()
successful/failure Message
successful/failure message

Figure 3.31 SD-2: Sign Out

2.4.3 Edit user Profile

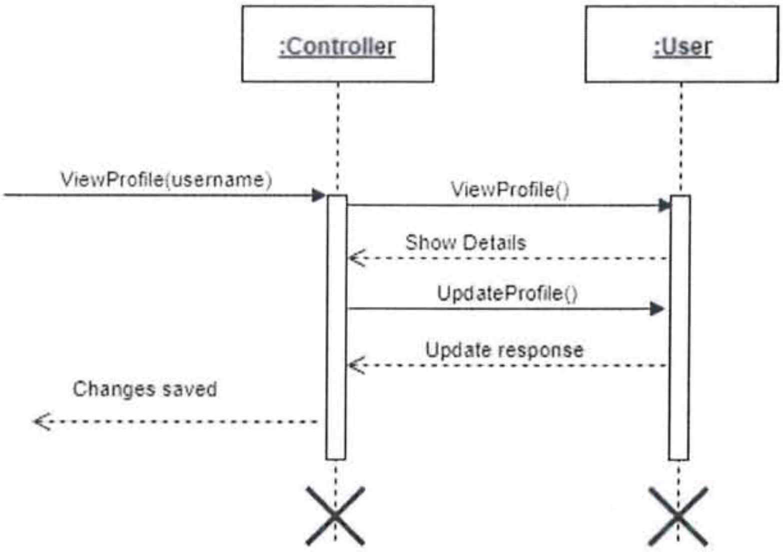


Figure 3.32 SD-3: Edit profile

2.4.4 Assign Rating

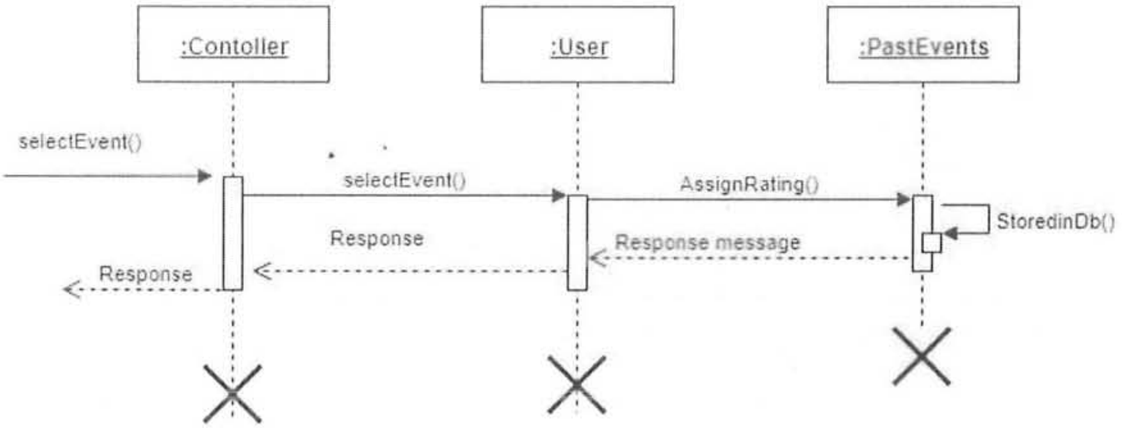


Figure 3.33 SD-4: Assign rating

2.4.5 Select interest

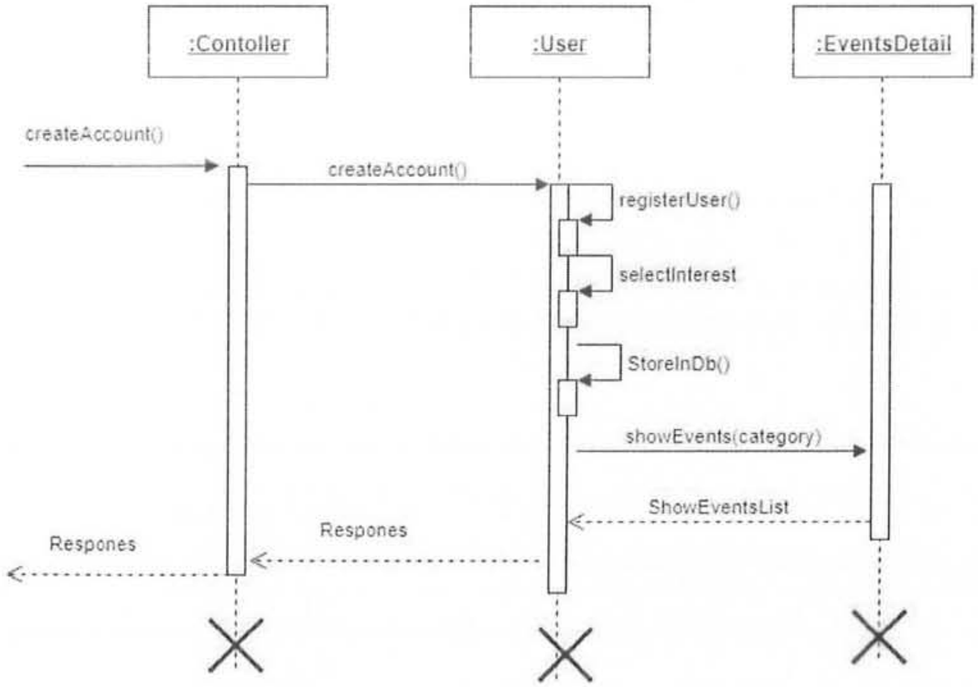


Figure 3.34 SD-5: Select Interest

2.4.6 Search events

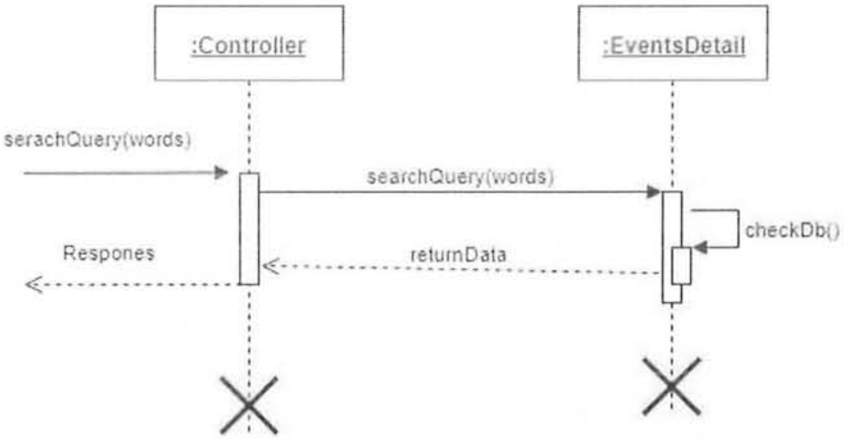


Figure 3.35 SD-6: Search Events

2.4.7 Register and Participate in Event

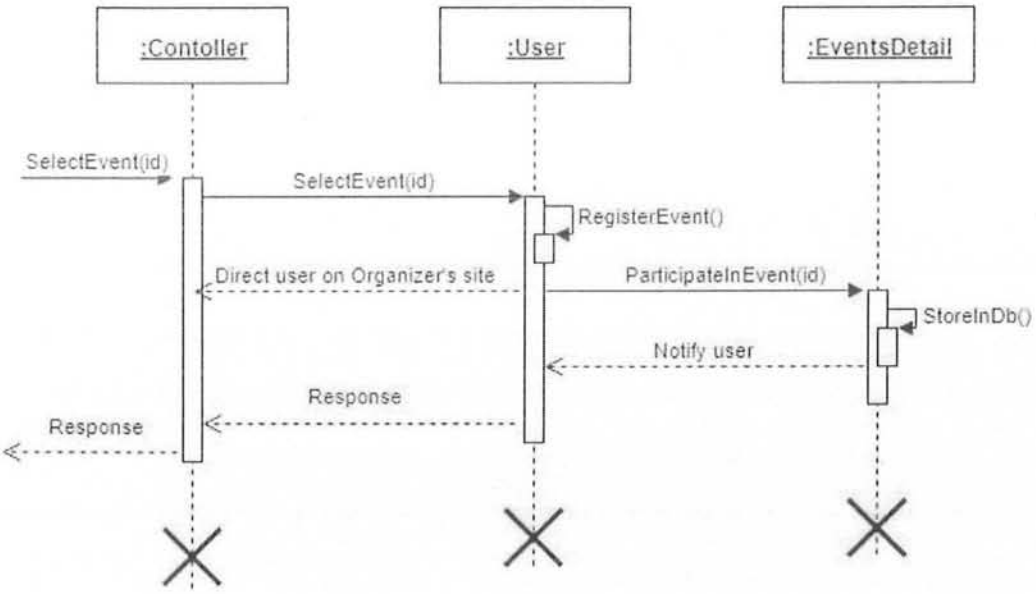


Figure 3.36 SD-7: Register Event

Chapter 4

Software Implementation Document

4.1 Introduction

This document describes the project implementation for developing the Event Notification app. The project implemented in java, Firebase and XML. Android studio is used as a tool for development.

4.2 Language Selection

The Java language is used for coding of application and XML is used for layout of application.

- JAVA

Before 2016, Java was the only official language to develop native android apps, but in October 2017, Google has adopted Kotlin as its 2nd official programming language.

- Firebase

Firebase is a mobile and web application development platform developed by Firebase, Inc. in 2011, then acquired by Google in 2014. Firebase stores and sync data with NoSQL cloud database. Data is synced across all clients in real-time, and remains available when your app goes offline. The Firebase Realtime Database is a cloud-hosted database. Data is stored as JSON and synchronized in real-time to every connected client. [3]

- XML

XML stands for Extensible Markup Language. XML is a markup language much like HTML used to describe data. XML tags are not predefined in XML. We must define our own Tags. Xml as itself is well readable both by human and machine. Also, it is scalable and simple to develop. In Android we use xml for designing our layouts because xml is lightweight language so it doesn't make our layout heavy.

4.3 Tools Selection

Android Studio

As Android Studio is the official Integrated Development Environment (IDE) for Android app development, based on IntelliJ IDEA. On top of IntelliJ's powerful code editor and developer tools, Android Studio offers even more features that enhance your productivity when building Android apps. It is developed by Google and JetBrains. Windows, Linux & macOs are its compatible operating systems.

4.4 Resources

Following resources are used in this project

- **ProjectLibre**

  Used to create Gantt chart.

- **Draw.io**

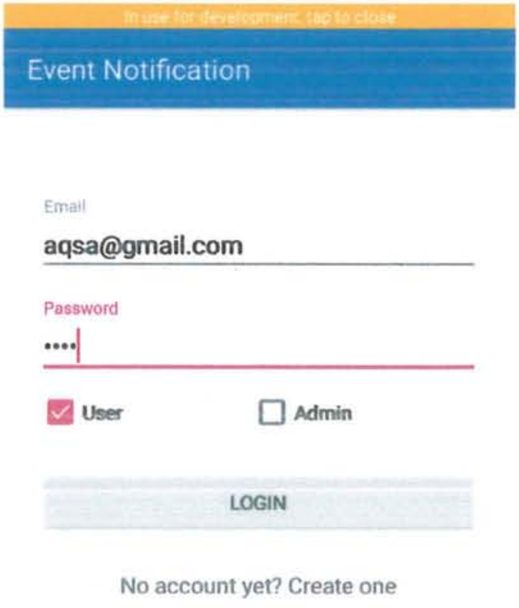
  It is free online diagram software for making flowcharts, process diagrams, organizational charts, UML, ER and network diagrams.

- **ArgoUML**

4.5 User Interface Design

User Interface design shows how the user will interact with the system to perform different task. The goal of user interface design is to make the user's interaction as simple and efficient as possible.

Screenshot of Login


In use for development, tap to close
Event Notification
LOGIN
No account yet? Create one

Email
aqsa@gmail.com

Password
••••

☑ User ☐ Admin

Figure 4.37 screenshot of user login

Screenshot of user profile

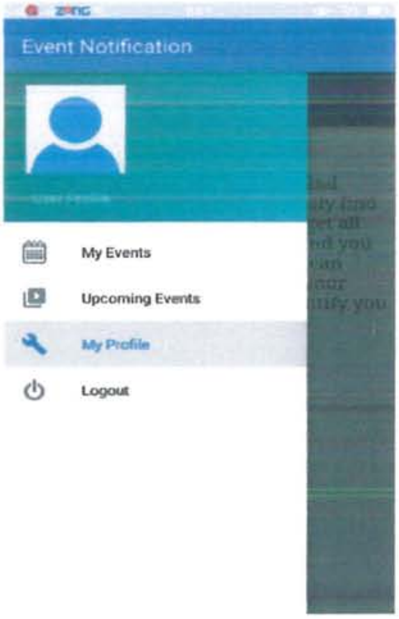
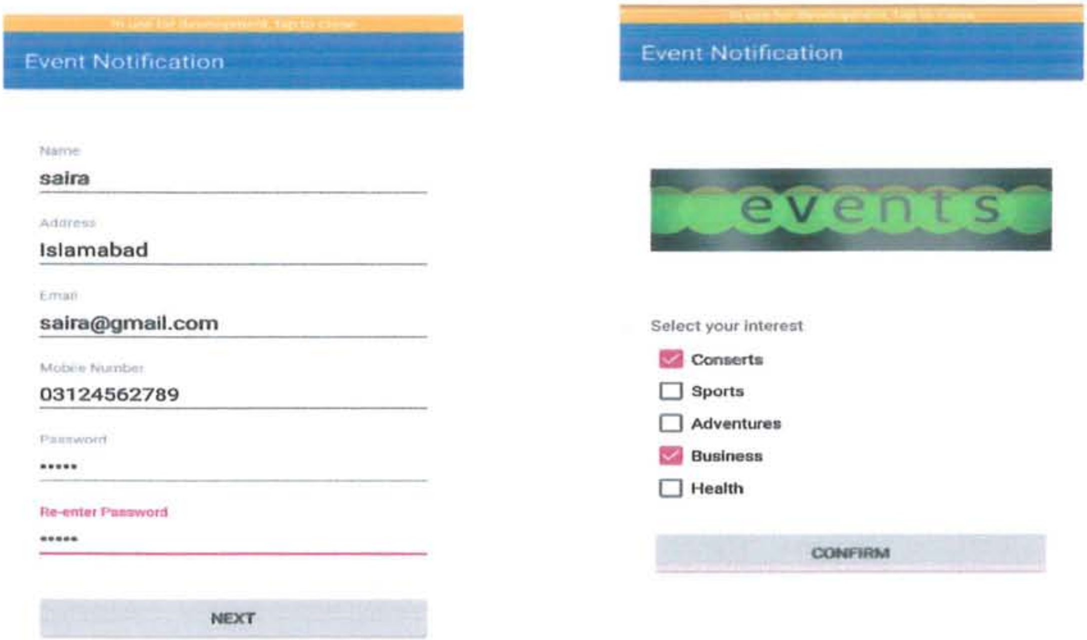


Figure 4.38 screenshot of profile

Screenshot of user Signup



Name
saira

Address
Islamabad

Email
saira@gmail.com

Mobile Number
03124562789

Password
*****

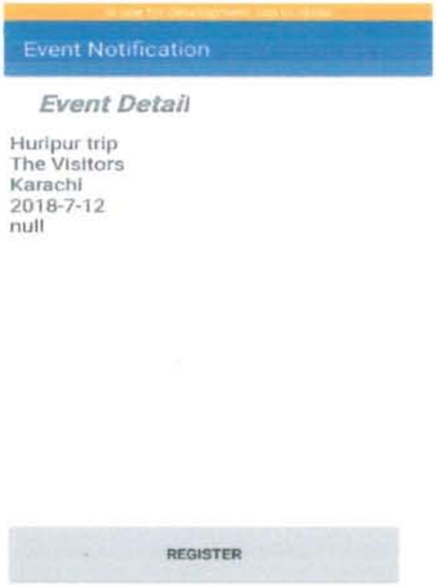
Re-enter Password
*****

Select your interest

☑ Conserts
□ Sports
□ Adventures
☑ Business
□ Health

Figure 4.39 Screenshot for Signup

Screenshot of events details

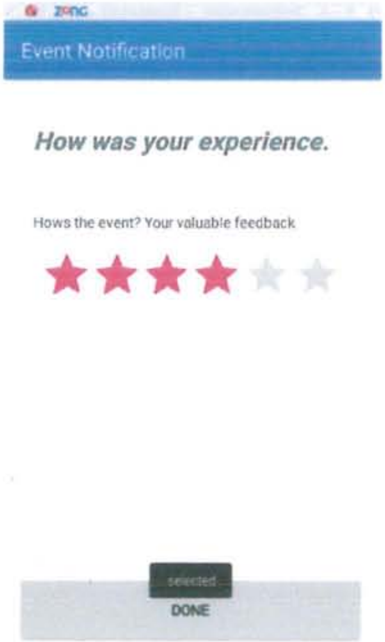


Event Detail

Huripur trip
The Visitors
Karachi
2018-7-12
null

Figure 4.40 screenshot of events detail

Screenshot of assign rating to event



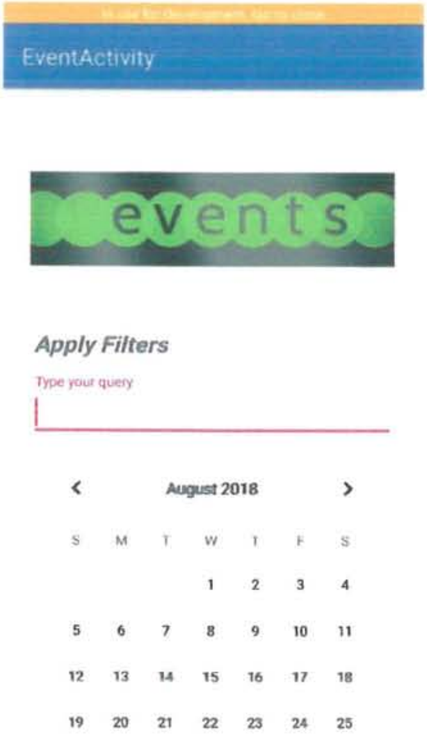
How was your experience.

Hows the event? Your valuable feedback

★★★★☆☆

Figure 4.41 screenshot of assign rating

Screenshot of applying filters

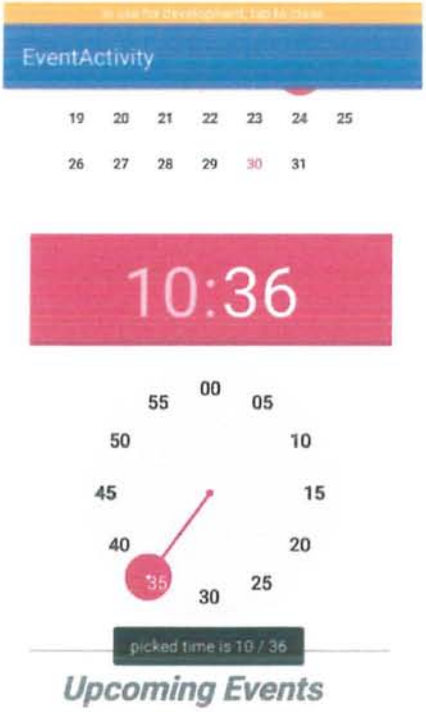


Apply Filters

Type your query

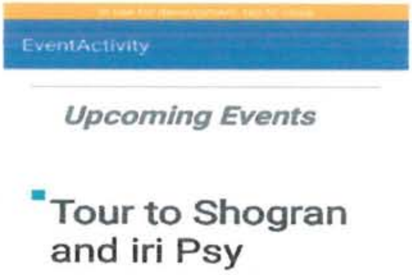
August 2018

S	M	T	W	T	F	S
			1	2	3	4
5	6	7	8	9	10	11
12	13	14	15	16	17	18
19	20	21	22	23	24	25



picked time is 10 / 36

Upcoming Events

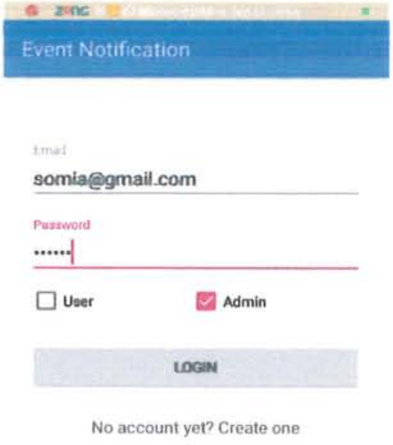


Upcoming Events

Tour to Shogran and iri Psy

Figure 4.42 screenshot of applying filters

Screenshot of Admin Login



Event Notification

Email

somia@gmail.com

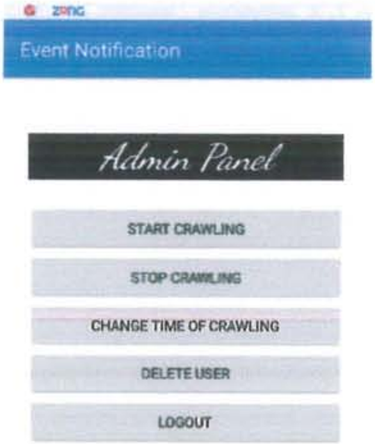
Password

••••••

□ User ☑ Admin

Figure 4.43 screenshot of Admin login

Screenshot of Admin Panel



Event Notification

Admin Panel

Figure 4.44 screenshot of Admin Panel

Chapter 5

Software Test Document

5.1 Introduction

Software testing contributes to improving the quality of the product. In current's testirng process, project requires planned and serialized documentation for testing and development. Proper documentation is the only key thing that can make it possible and makes testing more accurate in an organization.

This document describes the software test description of the Event Notification App software development project. It contains the description of test cases.

1.1.1 System Overview

This section, focusing on the structural aspects of testing, provides an overview of the system in terms of the components that are tested during the unit test.

1.1.2 Test Approach

A test approach is the test strategy implementation of a project, defines how testing would be carried out. It describes the sources (project documentation, electronic files/prototypes, etc.) for requirements that will be used to drive the test design.

I will use acceptance test plan. Acceptance testing is a test conducted to determine if the requirements of a specification are met.

5.2 Test Plan

A test plan outlines the strategy that will be used to test an application, the resources that will be used, the test environment in which testing will be performed, and the limitations of the testing.

5.2.1 Features to be tested

- Login
- Register user
- Edit Profile
- Select interest

- Search event
- Rating

5.2.2 Features not to be tested

- Logout
- Invite friends
- Invitation reminder
- Upcoming events notification
- View Profile
- Delete User
- Register for event
- Participate in event
- View past events

5.2.3 Testing Tools and Environment

- Laptop/Computer
- Mobile phone
- Internet connection

5.3 Test Cases

Test cases involve a set of steps, conditions, and inputs that can be used while performing testing tasks. The main intent of this activity is to ensure whether a software passes or fails in terms of its functionality and other aspects.

5.3.1 TC-1: Login

Table 5.22 TC-1: Login

Test ID	T001
Actor	User, Admin
Test description	Login to account.
Setup	Actor has account.
Instruction	A. 1. Actor selects Login. 2. Actor enters correct email address. 3. Actor enters correct password. 4. Actor clicks 'Login'. B. 1. Actor selects Login. 2. Actor enters correct email. 3. Actor enters incorrect password. 4. Actor clicks 'Login'. C. 1. Actor selects Login. 2. Actor enters incorrect email. 3. Actor enters correct password. 4. Actor clicks 'Login'.
Expected Result	A. Actor should be logged in. B. Prompts error message. C. Prompts error message.
Observed Result	A. As expected.

	B. As expected. C. As expected.
Verdict	A. Pass B. Pass C. Pass

5.3.2 TC-2: Register User

Table 5.23 TC-3 Register User

Test ID	T002
Actor	User
Test description	Account has been created successfully.
Setup	User opens the home page. Actor selects the create new account button
Instruction	A. 1. User enters email, name, password, phone #. 2. User enters Sign up button. B. 1. User enters email, name, password, phone#(digits <11) . 2. User enters Sign up button. C. 1. User enters unique email, name, password(less than 4 characters). 2. User enters sign up button.
Expected Result	A. User has successfully created an account. B. User get error message. C. User get error message.
Observed Result	A. As expected. B. As expected. C. As expected.

Verdict	A. Pass B. Pass C. Pass

5.3.3 TC-3: Edit Profile

Table 5.24 TC-3: Edit profile

Test ID	T003
Actor	User
Test description	User updates the account.
Setup	User login to account. A: User changes their Interest. B: User changes their name. C: User leave empty name field.
Instruction	A. 1. User selects 'Update Profile'. 2. User change event interest. 3. User clicks 'Update' button. B. 1. User selects 'Update Profile'. 2. User change name. 3. User clicks 'Update' button. C. 1. User selects 'Update Profile'. 2. User left empty the 'Name' field. 3. User clicks 'Update' button.
Expected Result	A. Profile should be updated. B. Profile should be updated. C. User should get error message.
Observed Result	A. As expected. B. As expected.

	C. As expected.
Verdict	A. Pass B. Pass C. Pass

5.3.4 TC-4: Select interest

Table 5.25 TC-4: Select Interest

Test ID	T004
Actor	User
Test description	User set their interest for event.
Setup	User logged in to the system.
Instruction	A. 1. User selects their interest from the list. 2. User presses Done button. B. 1. Use selects their interest. 2. User does not press Done button after selection.
Expected Result	A. User preference should be store in user's profile. B. User preference should not store.
Observed Result	A. As expected. B. As expected.
Verdict	A. Pass B. Pass

5.3.5 TC-5: Search event

Table 5.26 TC-5: Search event

Test ID	T005
Actor	User
Test description	User search required event.
Setup	Operate database connection 1. Create Connection User logged in to the system.
Instruction	A. 1. Press search option. 2. Enter "musical night" string. 3. Press 'OK' button. B. 4. Press search option. 5. Enter "" (null string). 6. Press 'OK' button.
Expected Result	A. Related result should be displayed. B. Blank screen should be displayed.
Observed Result	A. As expected. B. As expected.
Verdict	A. Pass B. Pass

5.3.6 TC-6: Give rating

Table 5.27 TC-6: Give rating

Test ID	T006
Actor	User
Test description	User rate events he has been attend.
Setup	Actor logged in to the system.
Instruction	A. 1. User select event from attend events. 2. User presses rate event. 3. User gives stars to event. 4. User press ok button. B. 1. User select event from attend events. 2. User presses rate event. 3. User gives stars to event. 4. User doesn't press ok and dismiss.
Expected Result	A. Event should be rate of that organizer. B. Event should not rate.
Observed Result	A. As expected. B. As expected.
Verdict	A. Pass B. Pass

Chapter 6

Conclusion and Future Enhancement

6.1 Introduction

This document describes the project summary, conclusion and future enhancements.

6.2 Summary

This is an application which helps people to know about their surrounding events. They can search events of their interest. By using this app user can get details of all nearby events in Islamabad like venue, date and time etc. And user can invite friends to attend event. This app also allows admin to remove user from the system. And admin can register event to attend by user.

6.3 Conclusions

This is very helpful and efficient application for every local user to find latest events in Islamabad. It will be an ease for organizers to handle events as events are handle on this platform. Admin will be able to manage user and crawling process.

6.4 Future Enhancement

In future there are some enhancements which can be done in this application. These enhancements will make the application more helpful and time saving.

- The most important enhancement which will extend its functionalities is that in future this application provides the details to user about all nearby hotels and restaurants so that user can easily go to attend faraway events and can stay there.
- We will introduce chatting system between users.
- As android has adopted Kotlin as a second official programming language after JAVA, so this could be develop in Kotlin as well.

References

[1] EventBrite, [Online]. Available: https://www.eventbrite.com/. [Accessed 2018].

[2] allevents, [Online]. Available: https://allevents.in/. [Accessed 2018].

[3] Firebase, [Online]. Available: https://firebase.google.com/docs/. [Accessed 2018].

[4] Applying UML and Patterns, An introduction to Object Oriented Analysis and Design and Iterative Development,Third Edition, Addison Wesley Professional, 2004.

[5] "Sequence Diagram," [Online]. Available: https://en.wikipedia.org/wiki/Sequence_diagram.

[6] Rogar.S.Pressman, Software Engineering-A Practitioner's Approach.7th Edition, McGraw Hill, 2010.