

Software Requirements Specification

Event Management System

Submitted in partial fulfillment
Of the requirements of
CS 223 Software Engineering

This work is based upon the submissions of the course Software Engineering (CS223).

The students who submitted this team projects were :

- Aakash Hasija, UG201310001
- Aman Singh, UG201310004
- Amit Jain UG201310005
- Shubham Saxena UG201310033

Version History

Version 1.0

As on 16th January, 2016, the first version of the Software Requirement Document was made. This document consists of the Introduction, Overall description, User Characteristics, Non-Functional requirements and Requirement Specifications

Version 2.0

As on 27th January, 2016, the second version of the Software Requirement Document was made. This document consisted of the details of Version 1 with an addition of the UML Diagrams, which included Use Case Diagrams, Class Diagrams, Sequence Diagrams and Activity Diagrams.

Version 3.0

As on 12th April, 2016, the third version of the Software Requirement Document was made. The document consisted of all the details of Version 2.0 with minor changes in the Use Case Diagrams and Class Diagram.

Table of Contents

Table of Contents.	i
List of Figures.	ii
1.0. Introduction.	1
1.1. Purpose.	1
1.2. Scope of Project	1
1.3 Constraints.	1
1.4 Assumptions and Dependencies.	2
1.3. Glossary.	2
1.4. References.	1
1.5. Overview of Document	1
2.0. Overall Description.	2
2.1 System Environment	2
2.2 Functional Requirements Specification.	2
2.2.1 Use case 1.	2
2.3 User Characteristics.	2
2.4 Non-Functional Requirements.	2
3.0. Requirements Specification.	3
3.1 Functional Requirements.	3
3.1.1 Signup	
3.1.2 Login	
3.1.3 Registering for An Event	
3.1.4 Add an event into Event manager	
3.1.5 Sending Automated event Notification mail	
3.1.6 Automated schedule generation	
3.1.7 View Database of an event	
3.1.8 Delete User	
3.1.9 View event Details	
3.3 Detailed Non-Functional Requirements.	3
3.4 Logical Structure of the Data.	3
4.0 Supporting information.	3
4.1 Table of contents and index.	3
4.2 Appendixes.	3

List of Figures

No table of figures entries found

1.0. Introduction

1.1. Purpose

The purpose of this document is to provide all the features, functions and constraints of our software **Event Management System**.

1.2. Scope of Project

This software system will be designed to create a dedicated application to manage and organize events. This will provide two different interfaces – manager interface for creating and organizing events and user interface for registering for events. The vision of this software is to improve efficiency and reduce cost and time associated with planning and management of the Festival.

1.3 Constraints

1. One person can only participate in one at a time.
2. Restriction on the number of participant in an event.
3. At one time only one event can take place at a particular venue.
4. A person cannot be event coordinator of two or more events at same instance.
5. A person cannot take part in more than 8 events in the whole festival.

1.4 Assumptions and Dependencies

1. Festival is of four days.
2. An Event will not have a duration of less than an hour and more than a day.
3. A day in the festival will start at 9:00 am and end at midnight.
4. Our software will not manage informal events.
5. The Payment will be made physically.

1.3. Glossary

Term	Definition
Manager	Festival Coordinator
User	Participant

1.4. References

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

1.5. Overview of Document

The rest of the document is designed in the following way:

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

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

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

The fourth chapter, *Supporting information*, of this document contains information required for validating the document and proper referrals to the above information contained.

2.0. Overall Description

2.1 System Environment

Python 2.7 and any version Django 1.7+

2.2 Functional Requirements Specification

2.2.1 Use case 1

Use case:

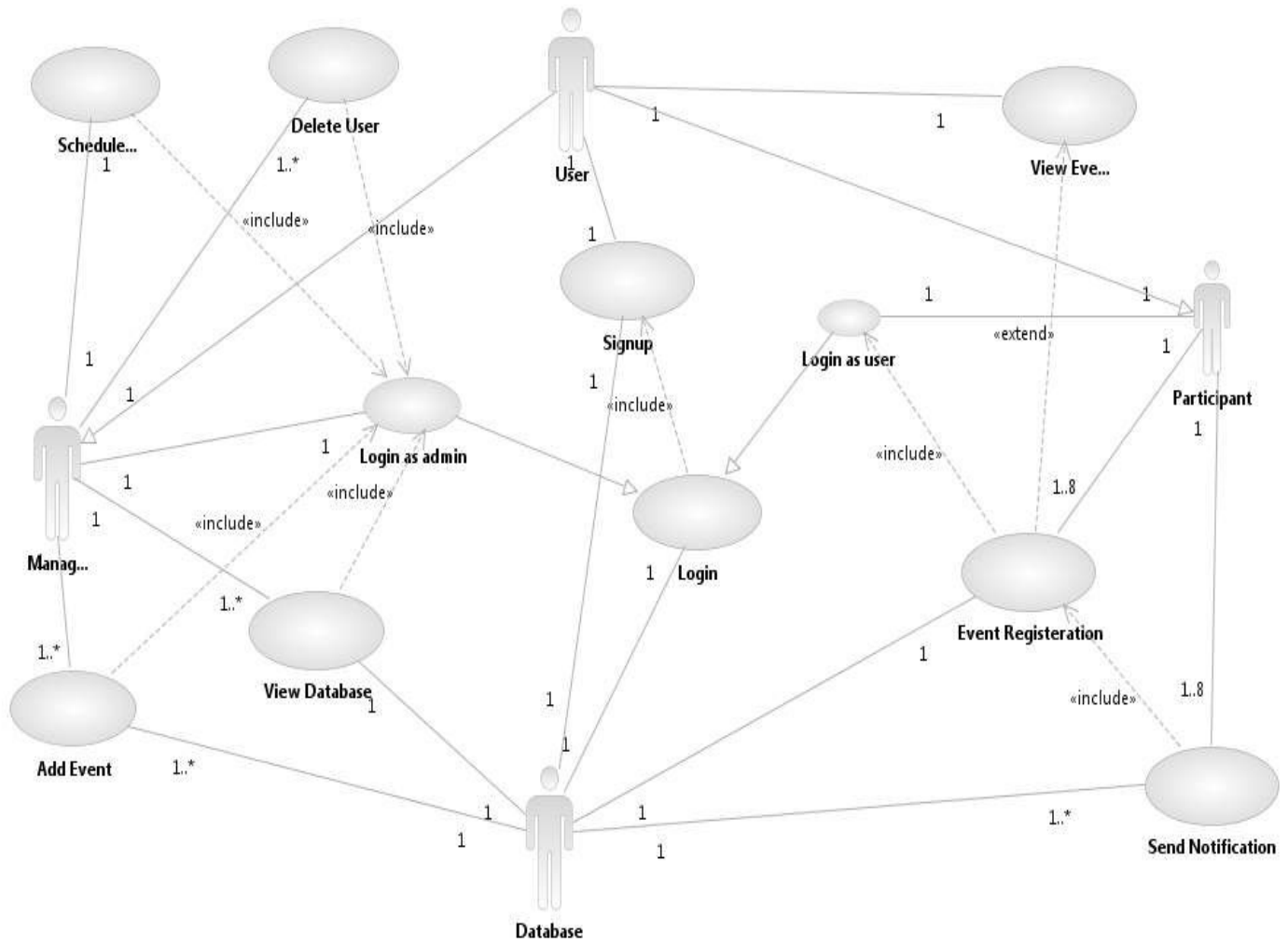
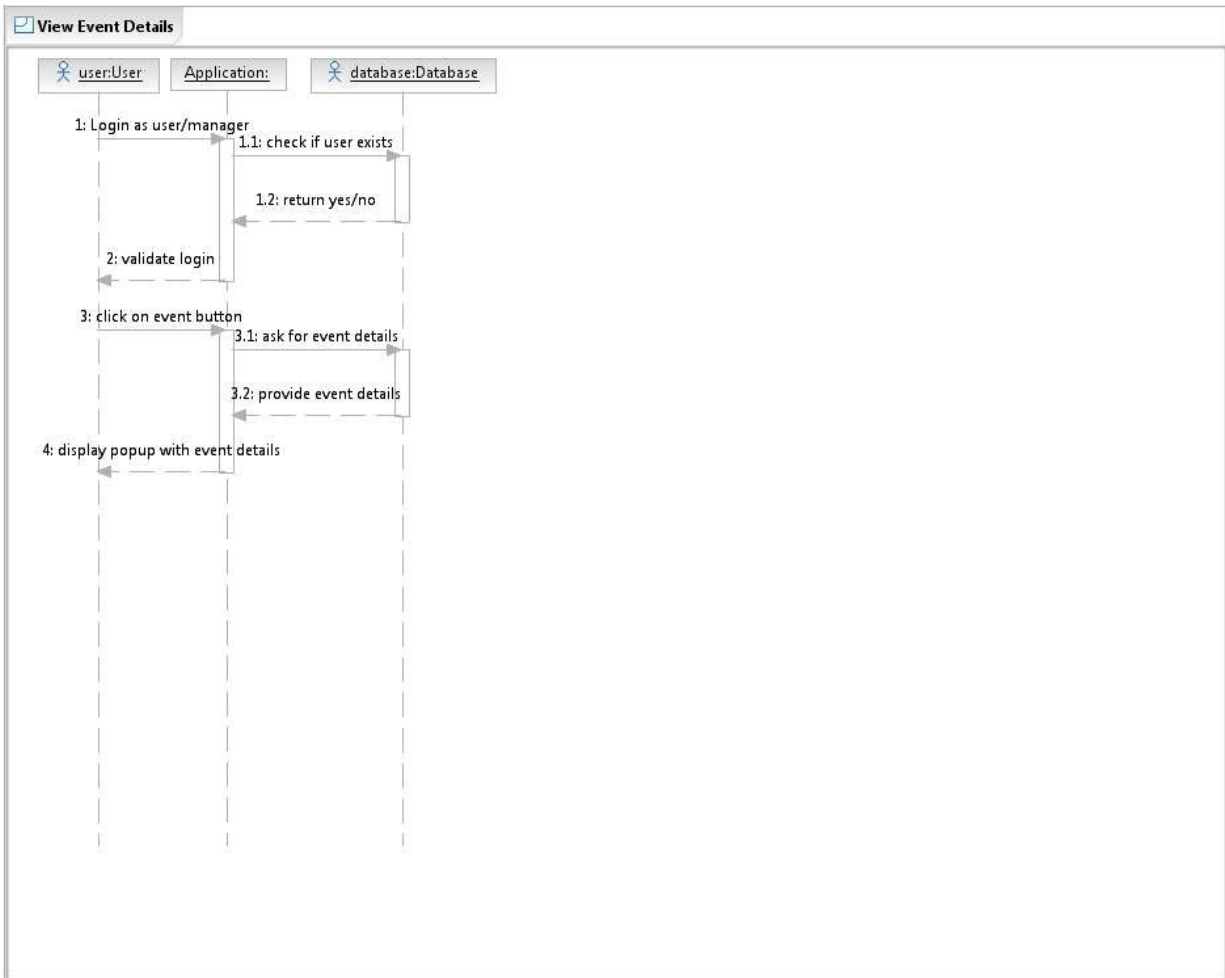


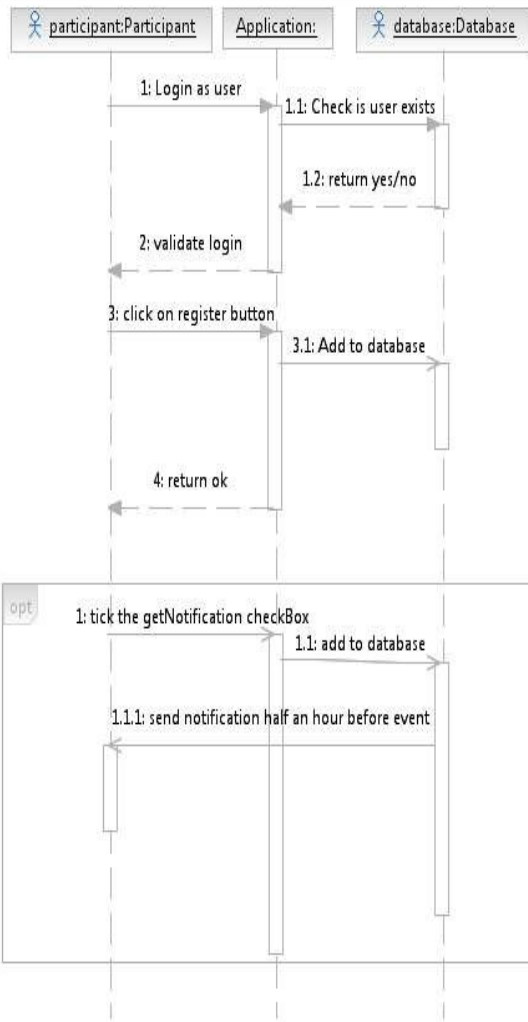
Diagram:

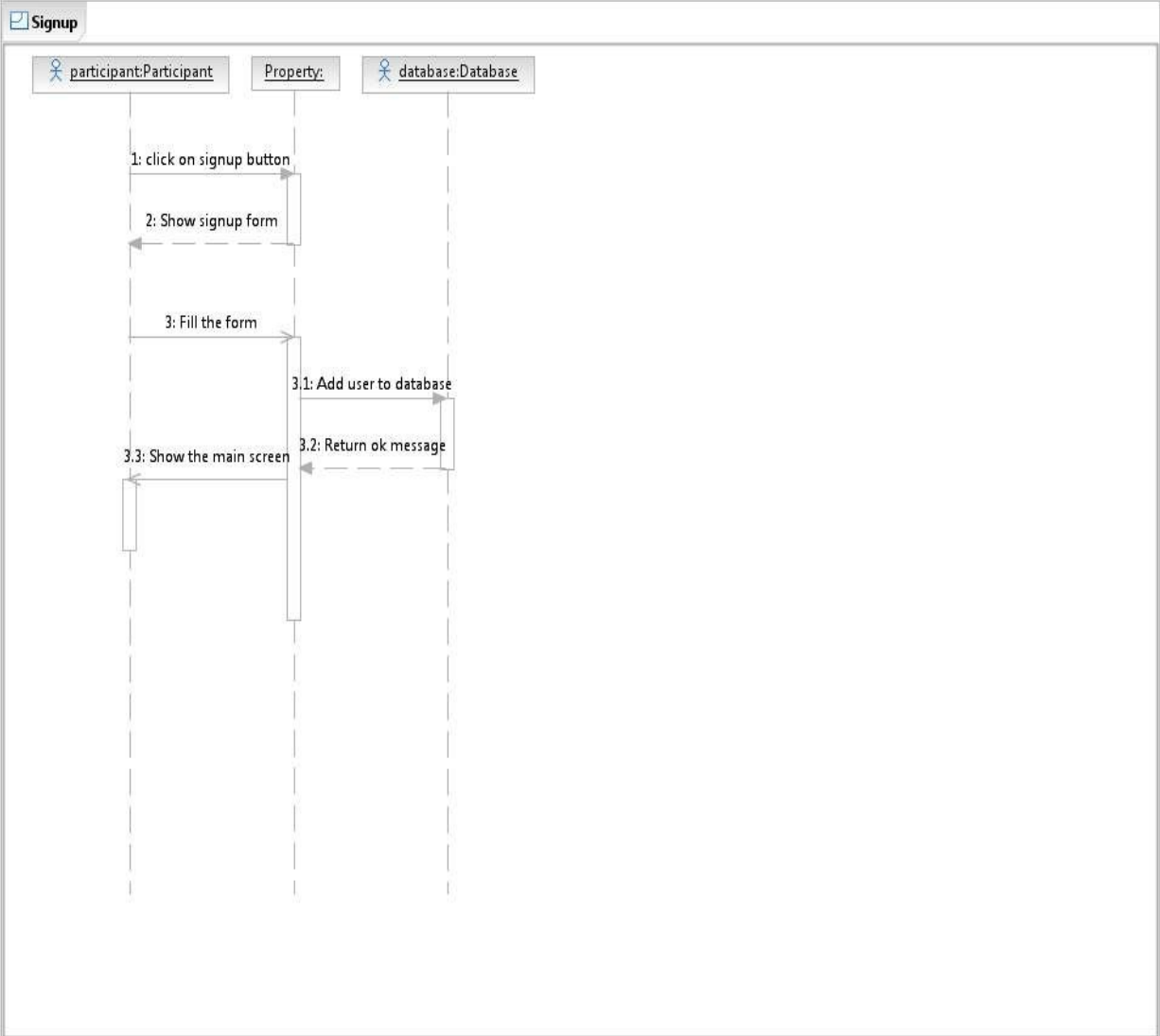
-

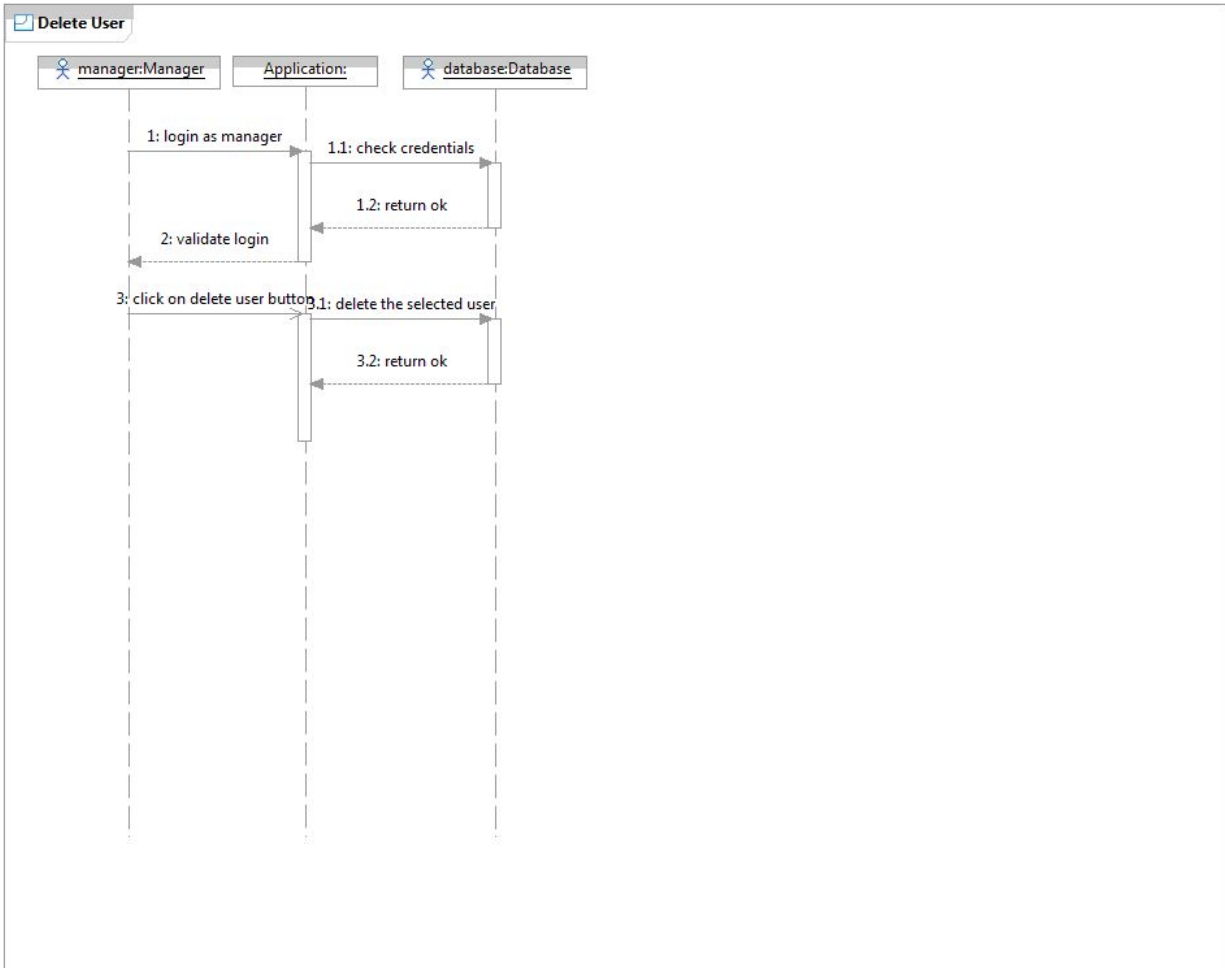


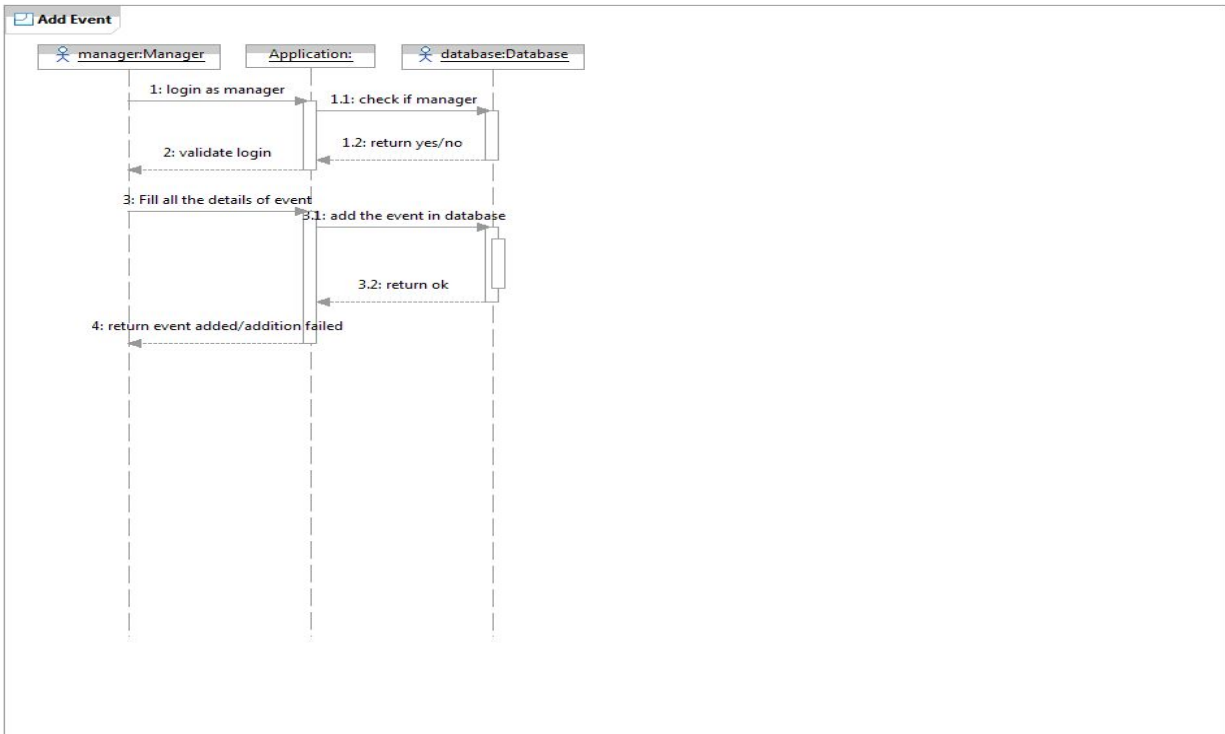


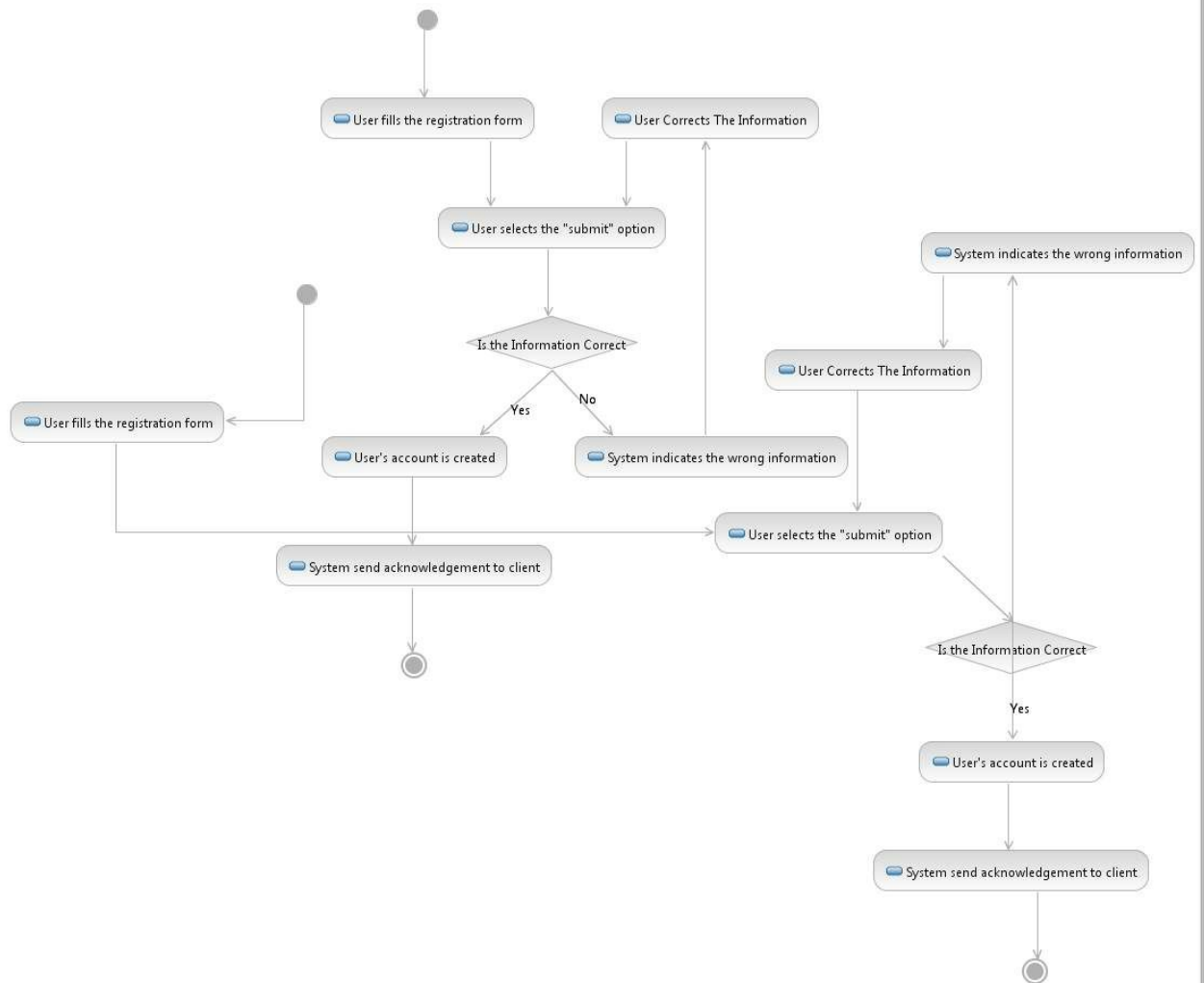
Event Registration



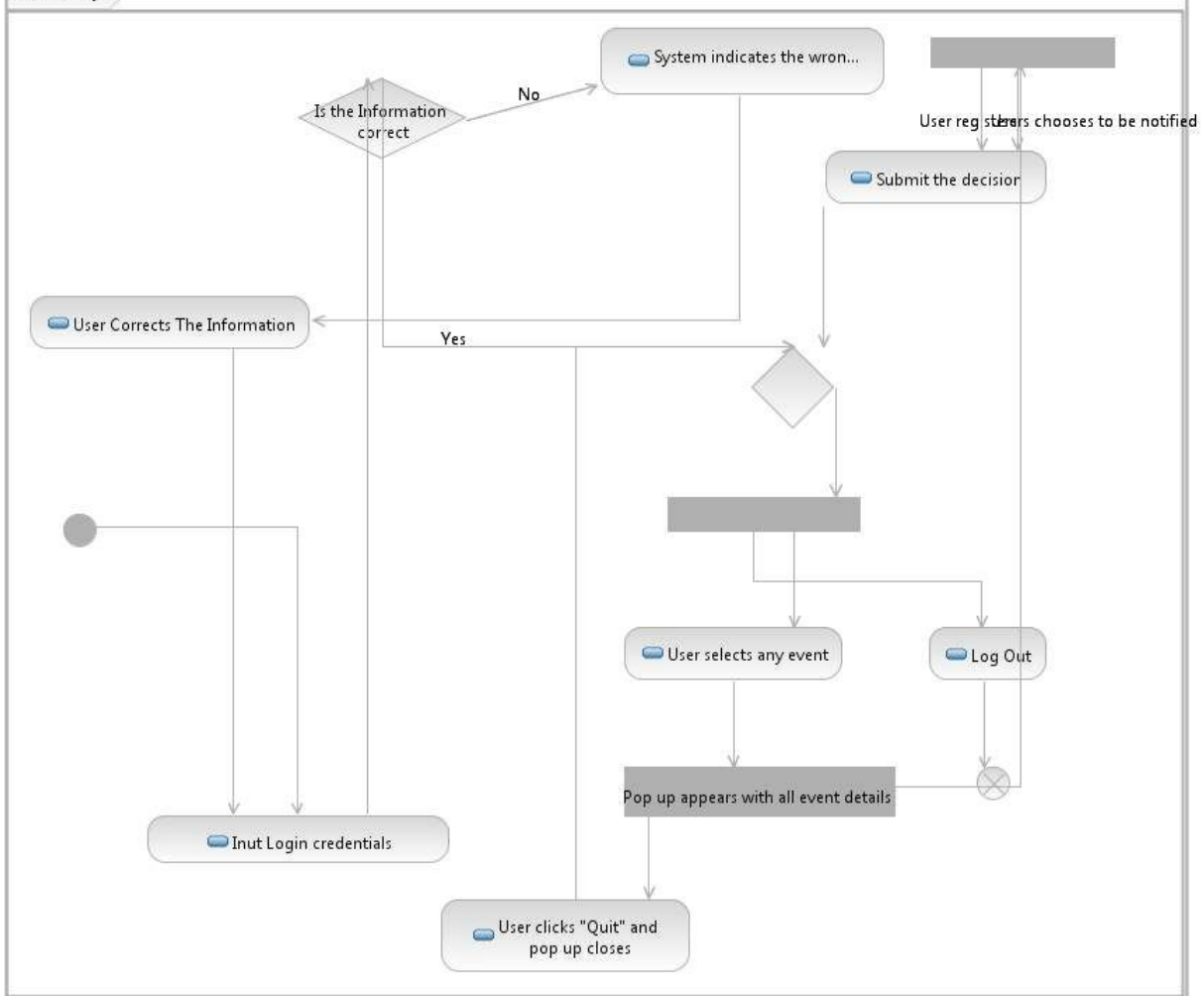




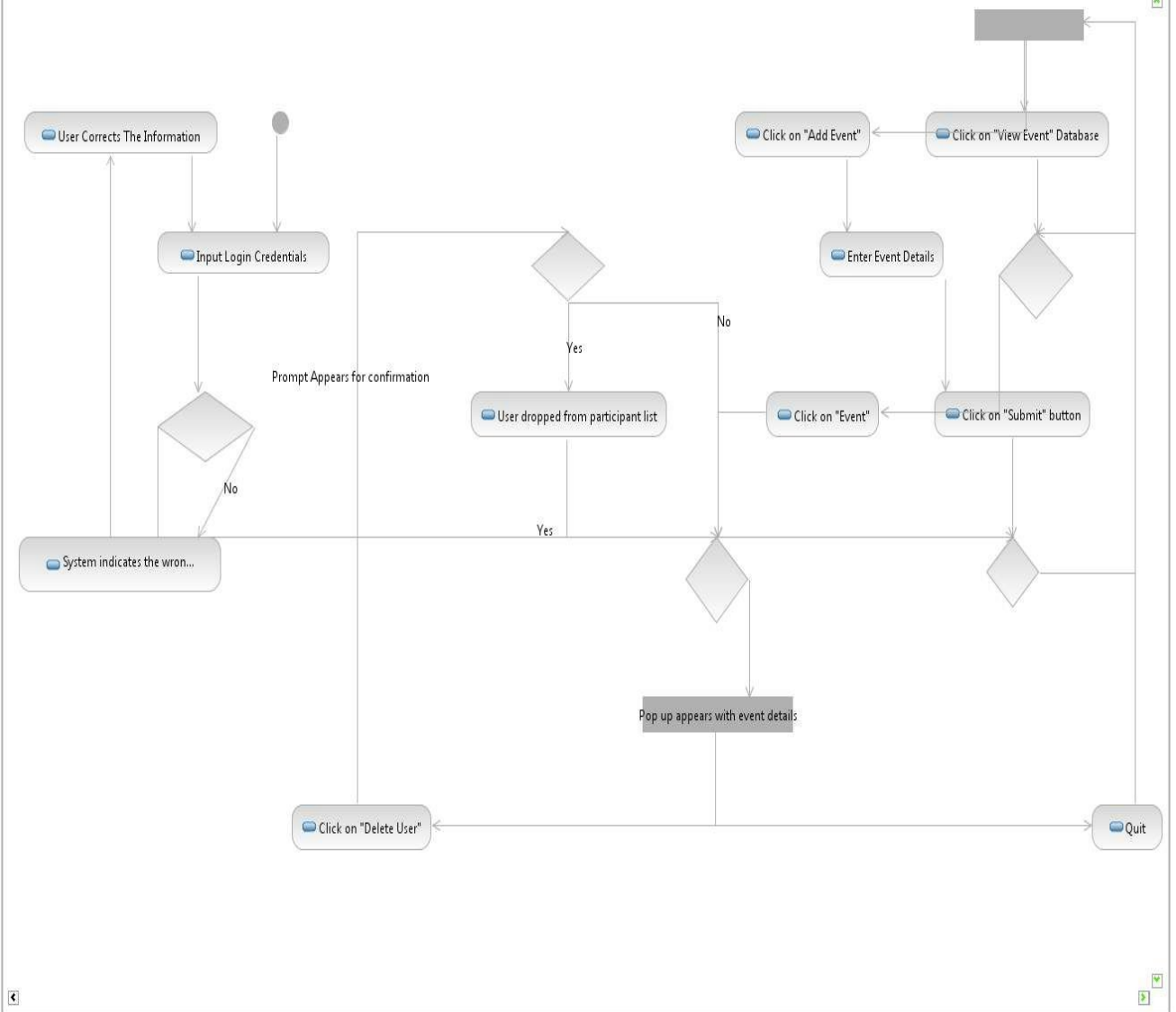




Activity2



Activity4



Brief Description

Initial Step-By-Step Description

2.3 User Characteristics

The user of this software system requires the following skills to use this software

1. Basic computer operating skills.
2. Knowledge of English Language
3. Access to Internet.

2.4 Non-Functional Requirements

The event manager will be on an online server with high speed internet connectivity. The hardware can be any general purpose computing machine. The event manager uses django (python) on the backend with sqldb as the database. The user with django installed on their pc's could use this software.

3.0. Requirements Specification

3.1 Functional Requirements

3.1.1 Signup

Use Case Name	Adding new user
----------------------	-----------------

Trigger	“Add event” button
Precondition	User should possess a valid Email Account
Alternative Paths	No Alternative path.
Basic Path	<ol style="list-style-type: none"> 1. Go to login page 2. Click “Signup” Button 3. Fill the required details in Signup form 4. Click “Done” Button
Postcondition	Only one user account on one Email Account
Exception Paths	Giving incorrect or invalid credentials
Other	-----

3.1.2 Login

Use Case Name	Login
Trigger	“Login” button
Precondition	User should have already signed up
Alternative Paths	No Alternative path.
Basic Path	<ol style="list-style-type: none"> 1. Go to login page 2. Enter the login credentials. 3. Click “Login” button.
Postcondition	User will be redirected to the home page of the software.
Exception Paths	Giving incorrect or invalid credentials

Other	-----
--------------	-------

3.1.3 Registering for An Event

Use Case Name	Register for an event
Trigger	Checking the “Register” checkbox in Event PopUp Screen
Precondition	Should be logged in as a user
Basic Path	<ol style="list-style-type: none"> 1. Login as the user 2. Click on the Button with the name of the event you are registered for. 3. A Popup with the complete event description opens. 4. Checking the register Checkbox in the popup.
Alternative Paths	No Alternative path.
Postcondition	User will be shown registered for that event and registration amount will be added to the total amount to be paid by the user.
Exception Paths	<ol style="list-style-type: none"> 1. User is already registered for an event in the same time slot. 2. User is already registered for 8 events.
Other	-----

3.1.4 Add An Event Into The Event Manager

Use Case Name	Add an Event Into The Event Manager.
Trigger	Clicking the “New Event” button
Precondition	User should be the Admin user only, i.e Manager.
Basic Path	<ol style="list-style-type: none"> 1. Login as the Admin user. 2. Click the “New Event” button. 3. Recheck the new schedule for discrepancies.
Alternative Paths	No alternative path since only manager is allowed.
Postcondition	An event will be added in the schedule reschedules by Maximum Bipartite Matching.
Exception Paths	Event is not added and the schedule remains as it is
Other	-----

3.1.5 Sending Automated Event Notification mail

Use Case Name	Sending Event Notification through mail
Trigger	Selecting the CheckBox in the event Popup
Precondition	Should be registered for that event
Basic Path	<ol style="list-style-type: none"> 1. Register for the festival 2. Login as user 3. Register for the desired event 4. Tick the CheckBox
Alternative Paths	Login as the manager

Postcondition	A mail notification will be sent to the user or manager
Exception Paths	User is not registered for the event
Other	-----

3.1.6 Automated Schedule Generation

Use Case Name	Automated Schedule Generation
Trigger	“Add event” button
Precondition	Entering the event details and clicking on “Add Event” Button
Basic Path	<ol style="list-style-type: none"> 1. Manager login 2. Click on “New event” Button 3. Enter event details 4. Click on “Add event” button
Postcondition	A Possible Schedule will be created
Exception Paths	If slot allocation is not possible with the given three preferences of that event.
Other	We will use Maximum Bipartite Matching algorithm to allocate preferred slots.

3.1.7 View Database of an event

Use Case Name	Viewing database of an event by the managers
----------------------	--

Trigger	Pressing the required event button by the Manager
Precondition	Manager should have admin privilege.
Basic Path	<ol style="list-style-type: none"> 1. Create a new Manager from the database(Only for first time) 2. Login on the admin page. 3. Click on the button with the required event name.
Alternative Paths	None
Postcondition	Manager will be able to access the complete event database.
Exception Paths	Login attempt from account with non admin privilege.
Other	-----

3.1.8 Delete User

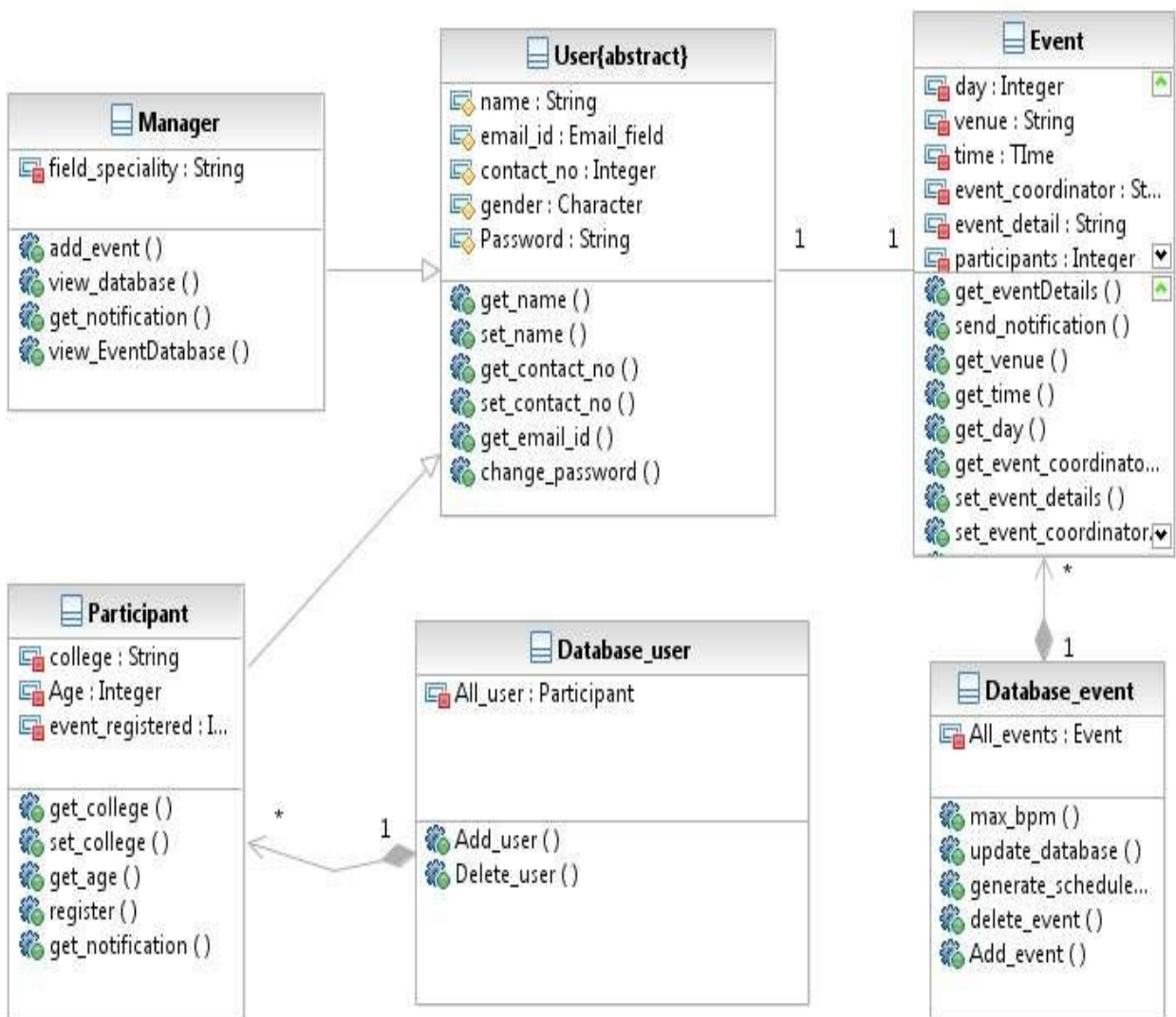
Use Case Name	Delete User
Trigger	“Delete” button
Precondition	The login should be as the manager or admin
Alternative Paths	No Alternative path.
Basic Path	<ol style="list-style-type: none"> 1. Login. 2. Click on the respective event. 3. Click on “View Details” button in front of the event. 4. The details will be displayed on a popup.

Postcondition	The screen will return to its previous state.
Exception Paths	Login attempt from User account will not provide this functionality.
Other	-----

3.1.9 View Event Details

Use Case Name	View Event Details
Trigger	“View Details” button
Precondition	The login should be as the manager or user
Alternative Paths	No Alternative path.
Basic Path	Login as Manager. Click on user database. Click on cross in front of the user to be deleted. The user will be removed.
Postcondition	The screen will return to its previous state.
Exception Paths	-----
Other	-----

Class Diagram:



3.3 Detailed Non-Functional Requirements

3.4 *Logical Structure of the Data*

<< Keep this blank for the time being>>

4.0 Supporting information

4.1 Table of contents and index

4.2 Appendixes