THE EVENT MANAGER

Submitted in partial fulfillment of the requirements of

PG Diploma in Advanced Computing

By

Vishal Kadam	**230950320036**
Kiran Kambale	**230950320037**
Dnyaneshwar Kangude	**230950320038**
Kapil Hadoltikar	**230950320039**
Kartik Shelke	**230950320040**
Keshav Sangi	**230950320041**
Komal Kawthe	**230950320042**

Guide:

Mr. Vamsikrishna Kotapati

Centre for Development of Advanced Computing

Hyderabad.

Sept 2023

Table of Contents

Acronyms

Acronym	Definition
SRS	Software Requirement Specification
EPWA	Event Manager Web Application
QFD	Quality Function Deployment
ER	Entity Relationship
CRC	Class Responsibility Collaboration
DFD	Data Flow Diagram

Chapter 1: Introduction

1.1 Purpose

This document is the Software Requirements Specification (SRS) for the Event Manager Web Application(EMS). It contains detailed functional, non-functional, and support requirements and establishes a requirements baseline for development of the system. The requirements contained in the SRS are independent, uniquely numbered, and organized by topic. The SRS serves as the official means of communicating user requirements to the developer and provides a common reference point for both the developer team and stakeholder community. The SRS will evolve over time as users and developers work together to validate, clarify and expand its contents.

1.2 Intended Audience

This SRS is intended for several audiences, including the customer, as well as the project manager, designers, developer and tester.

- The customer will use this SRS to verify that the developer team has created a product that is acceptable to the customer.
- The project manager of the developer team will use this SRS to plan milestones and a delivery date, and ensure that the developing team is on track during development of the system.
- The designer will use this SRS as a basis for creating the system's design. The designer will continually refer back to this SRS to ensure that the system they are designing will fulfill the customer's needs.
- The developer will use this SRS as a basis for developing the system's functionality. The developer will link the requirements defined in this SRS to the software they create to ensure that they have created software that will fulfill all of the customer's documented requirements.
- The tester will use this SRS to derive test plans and test cases for each documented requirement.

1.3 Scope

This document will address a few key themes in terms of the Event Manager Web Application– A web based authenticated interface of creating new events and a notification system for running and upcoming events. We intend to consider only Client and faculty members of Center of Development Mumbai

1.4 Rational

- Event Manager is the application of project management to the creation and development of large or small scale events such as Festivals, Weddings, Birthdays, Conferences, Ceremonies, Formal parties and Concerts. The last few years have seen a rapid growth in the event management industry. Considering the existing system problems related to event management we are developing a Web application for event management. This application will be accessible in Web Browser. Application will mainly focus on Weddings, Birthday parties, Functions and Social events. These details will be firstly verified by the administrator to fulfill the security protocols and then it will generate the complete descriptive bill. In this paper we present a Web application to make it easier for a layman to plan an event in a hassle-free manner. This application will assist him/her in planning a successful and fun event.

Chapter 2: Scenario-Based Model

2.1 Introduction

In this model the system is described from the user's point of view. As this is the first model, it serves as input for creation of other modeling elements.

2.2 Use Case Scenario

Level – 0	Level – 1	Level – 2	Actors
Event Manager Web Application	Authentication	Sign Up	Clients
		Verification	Admin
		Sign In	Admin, Event Manager, Clients
		Sign Out	Admin, Event Manager, Clients
		Change Password	Admin, Event Manager, Clients
		Create User	Admin
		Remove User	Admin
	Event	Create event	Admin, Event Manager, Clients
		Update event	Admin, Event Manager, Clients
		Sub event	Admin, Event Manager, Clients
	Template	Create template	Admin
		Update template	Admin
		Use Template	Admin, Event Manager, Clients
		Sub event of template	Admin, Event Manager, Clients
	User activity		Admin, Event Manager, Clients

2.3 Use Case Description

We will elaborate use case scenario to use case diagram, description, activity diagram swim-lane diagram. Here is the use case diagram of level-0 for Event Manager Web Application

This is the elaborated form of level-0 for Event Manager Web Application.

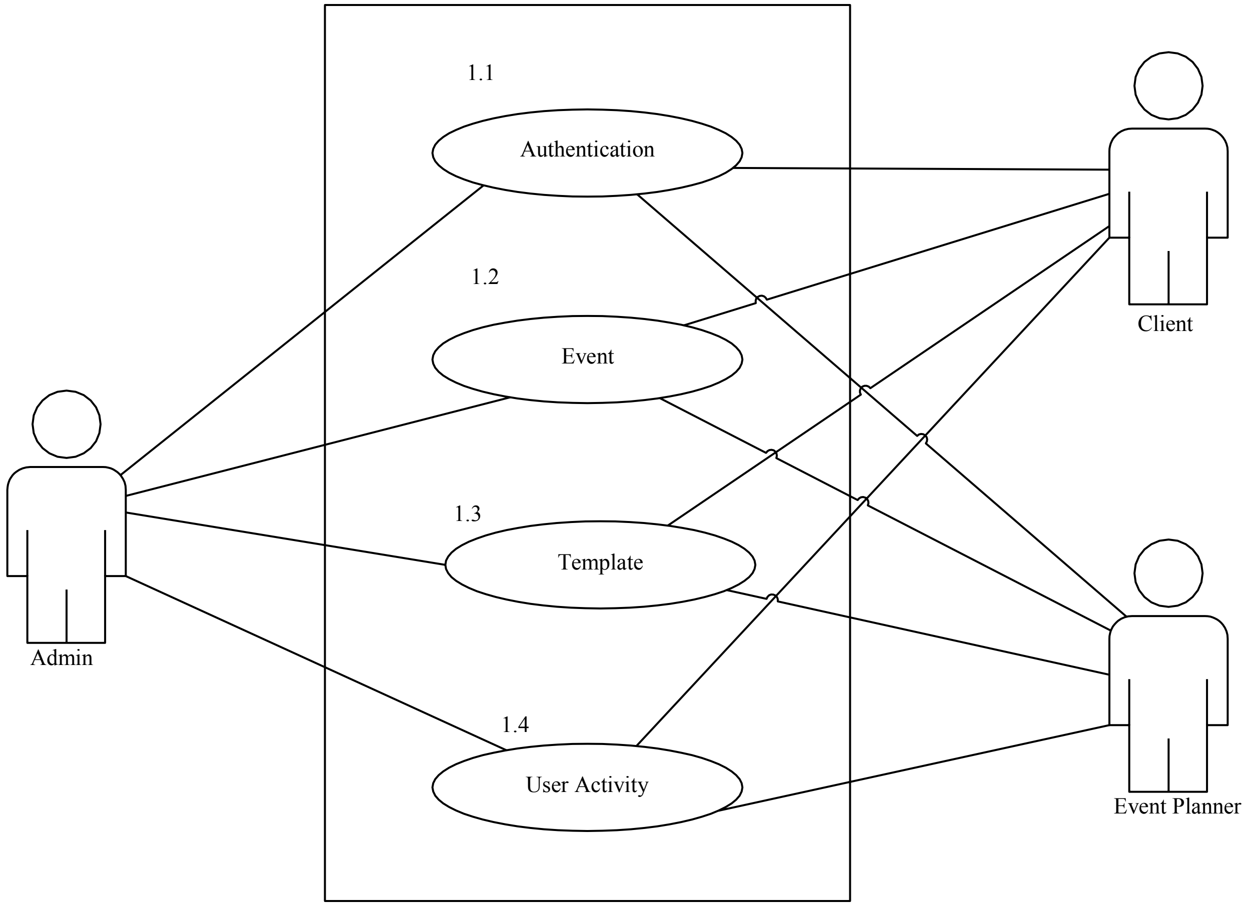

Figure 2:Use Case Diagram of EPWA (Level-1)

This the elaborated form of level-0 for Event Manager Web Application

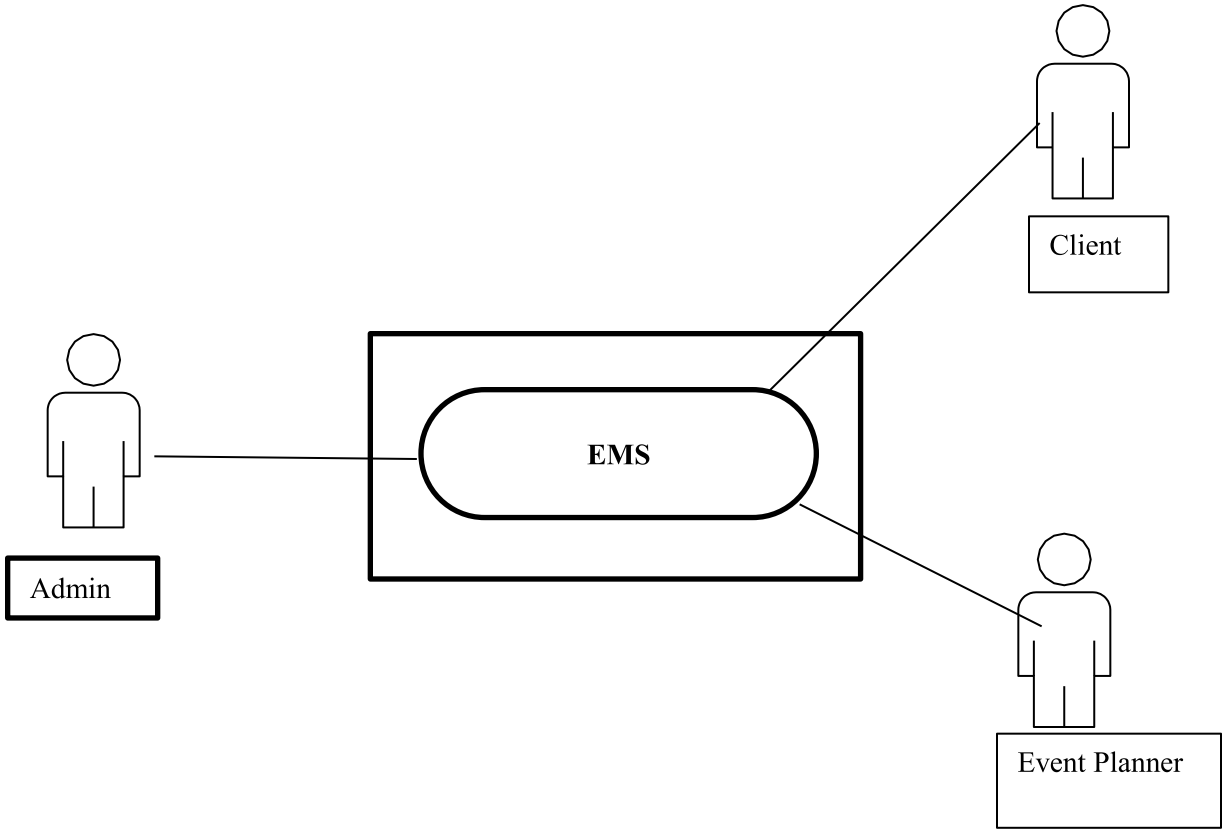

Figure 2:Use Case Diagram of EPWA

2.3.1Authentication

2.3.1.1 Sign up

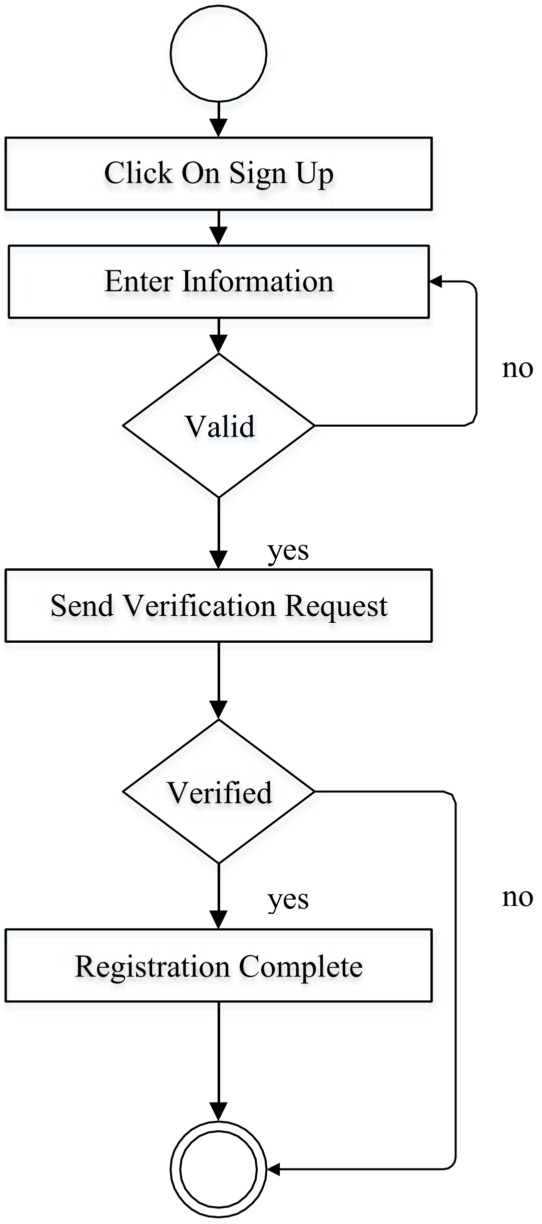

Figure 3: Activity Diagram – Level 1.1.1 (Sign Up)

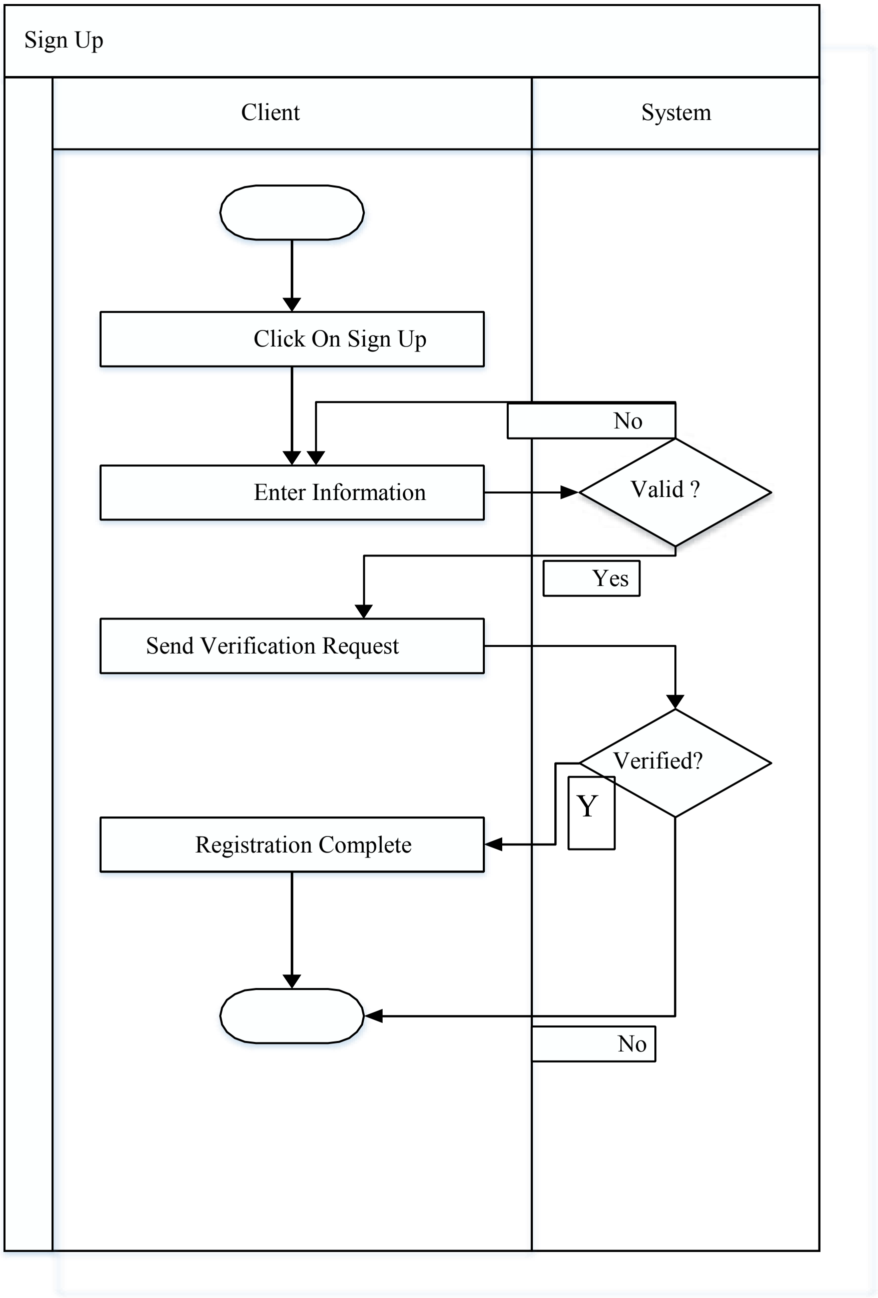

Figure 4: Swim Lane Diagram – Level 1.1.1 (Sign Up)

2.3.1.2 Sign in

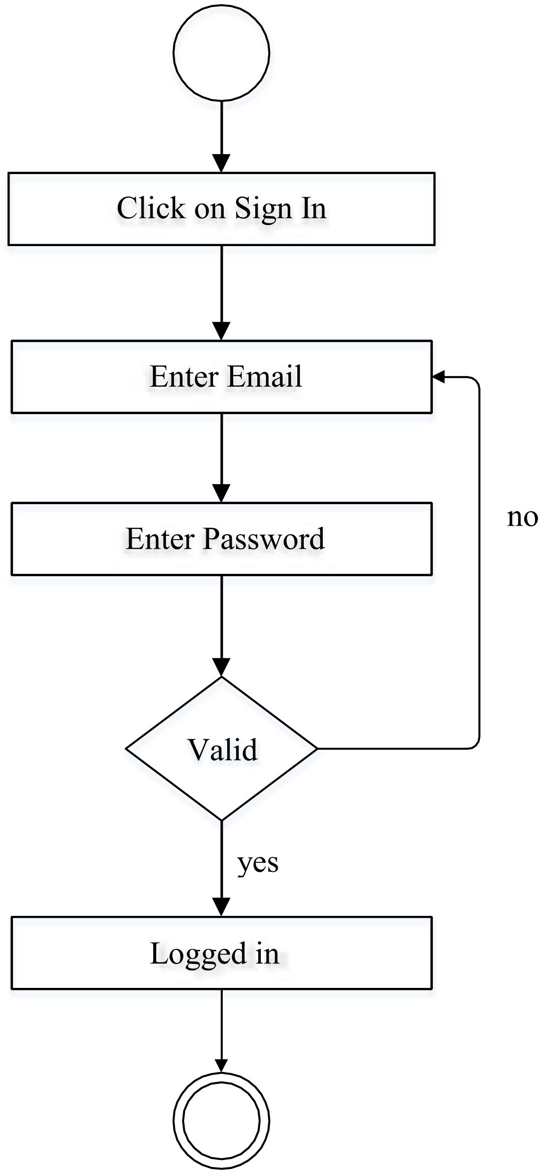

Figure 5 : Activity Diagram – Sign in

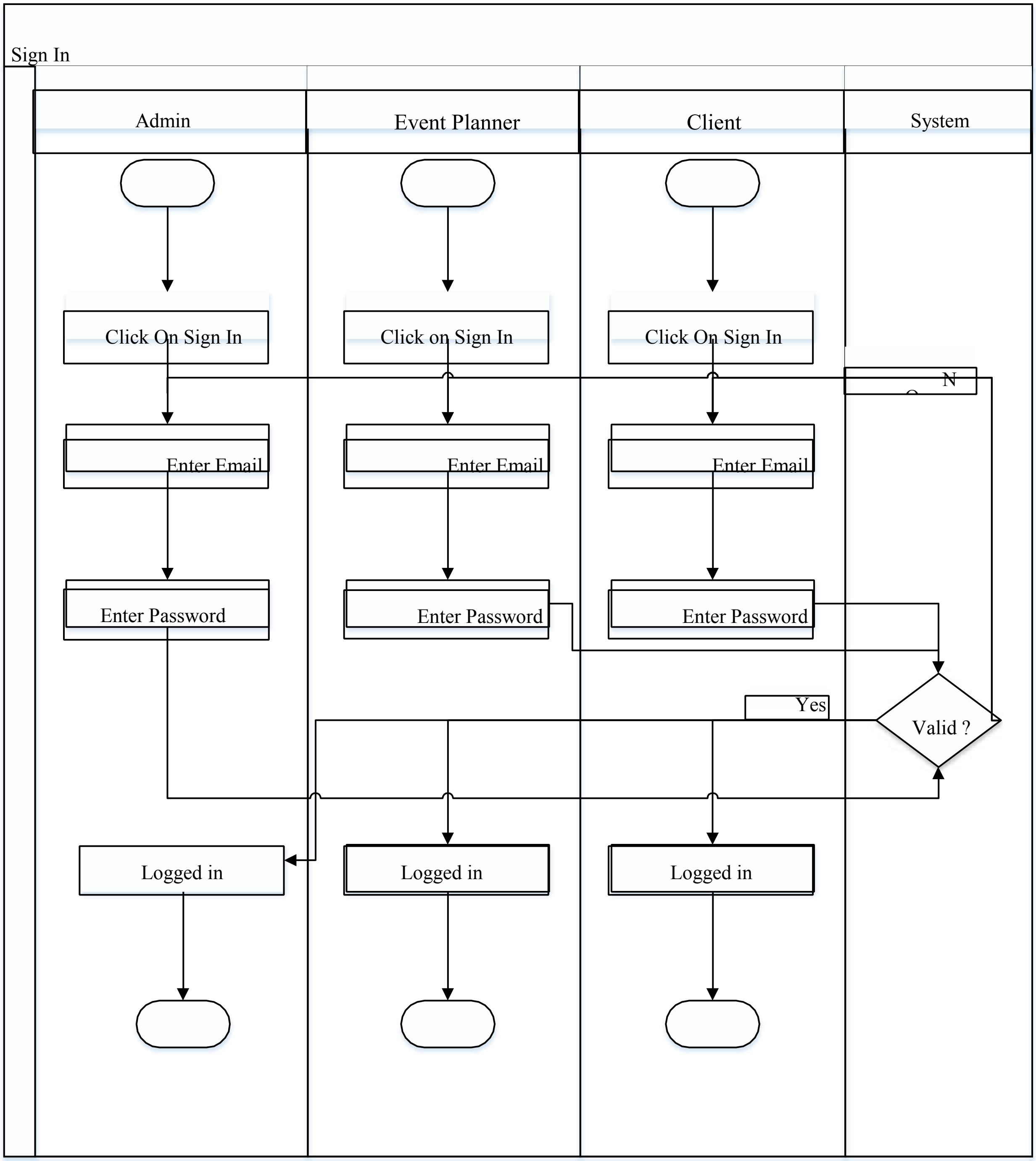

Figure 6: Swim Lane Diagram – Sign in

2.3.1.3 Sign out

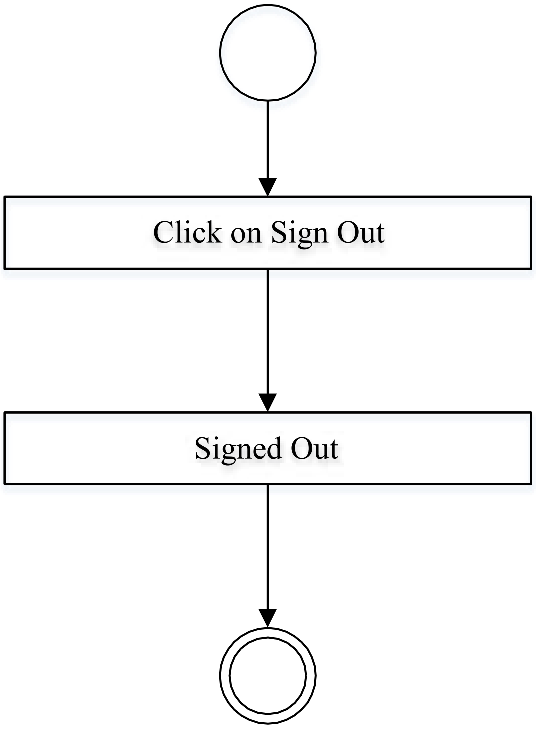

Figure 7: Activity Diagram – Sign out

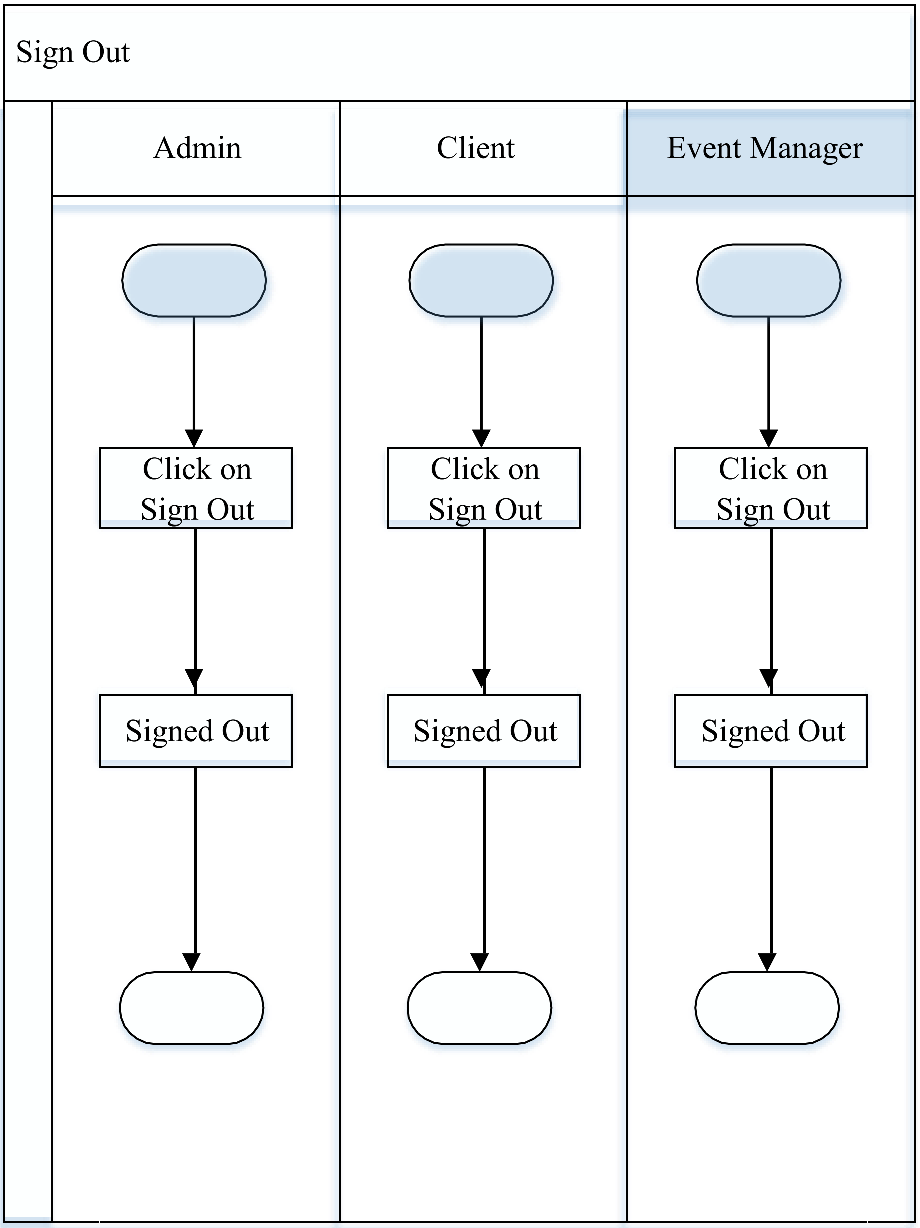

Figure 8: Swim Lane Diagram – Sign out

2.3.1.4 Change password

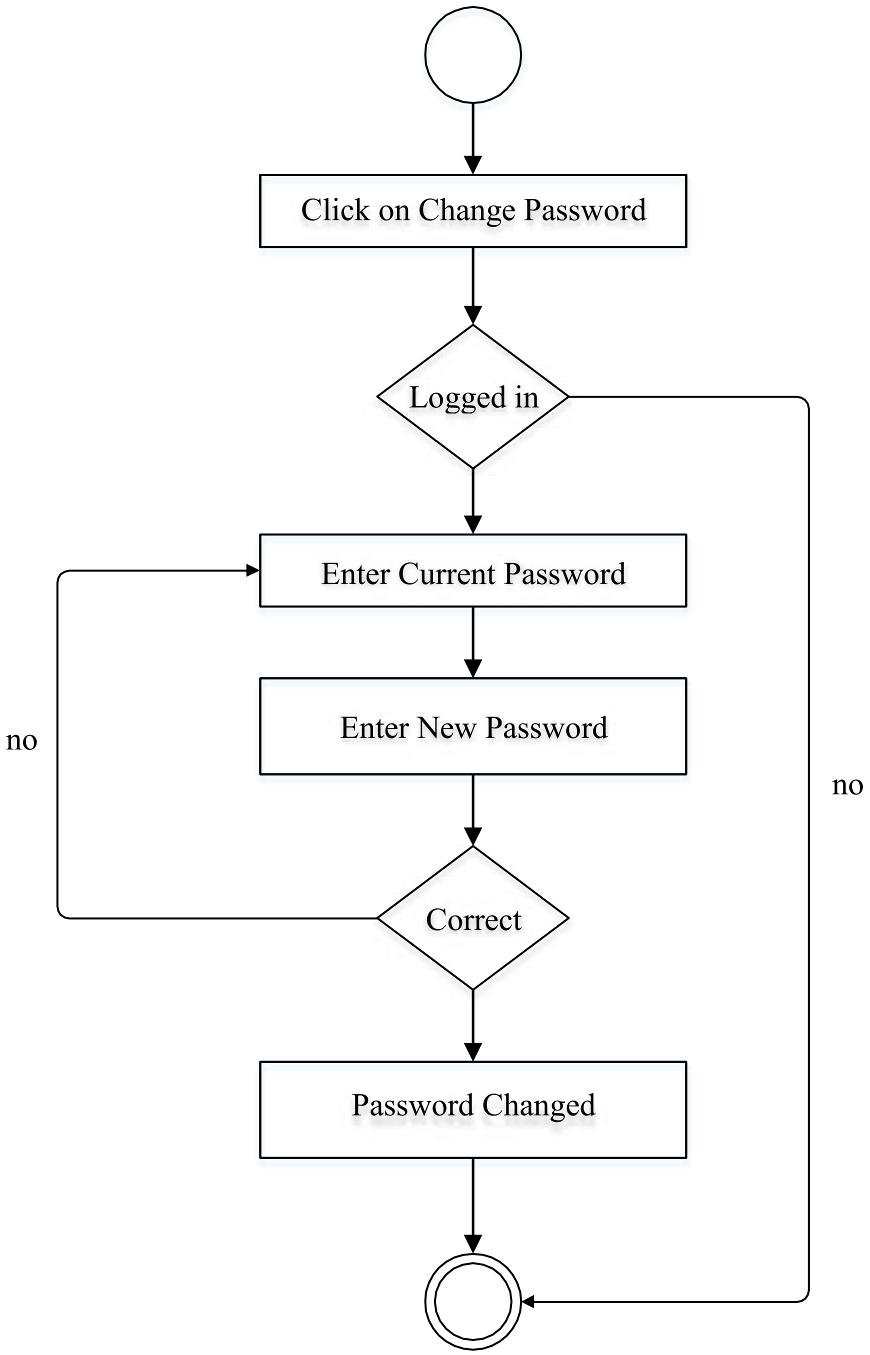

Figure 9: Activity Diagram – Change Password

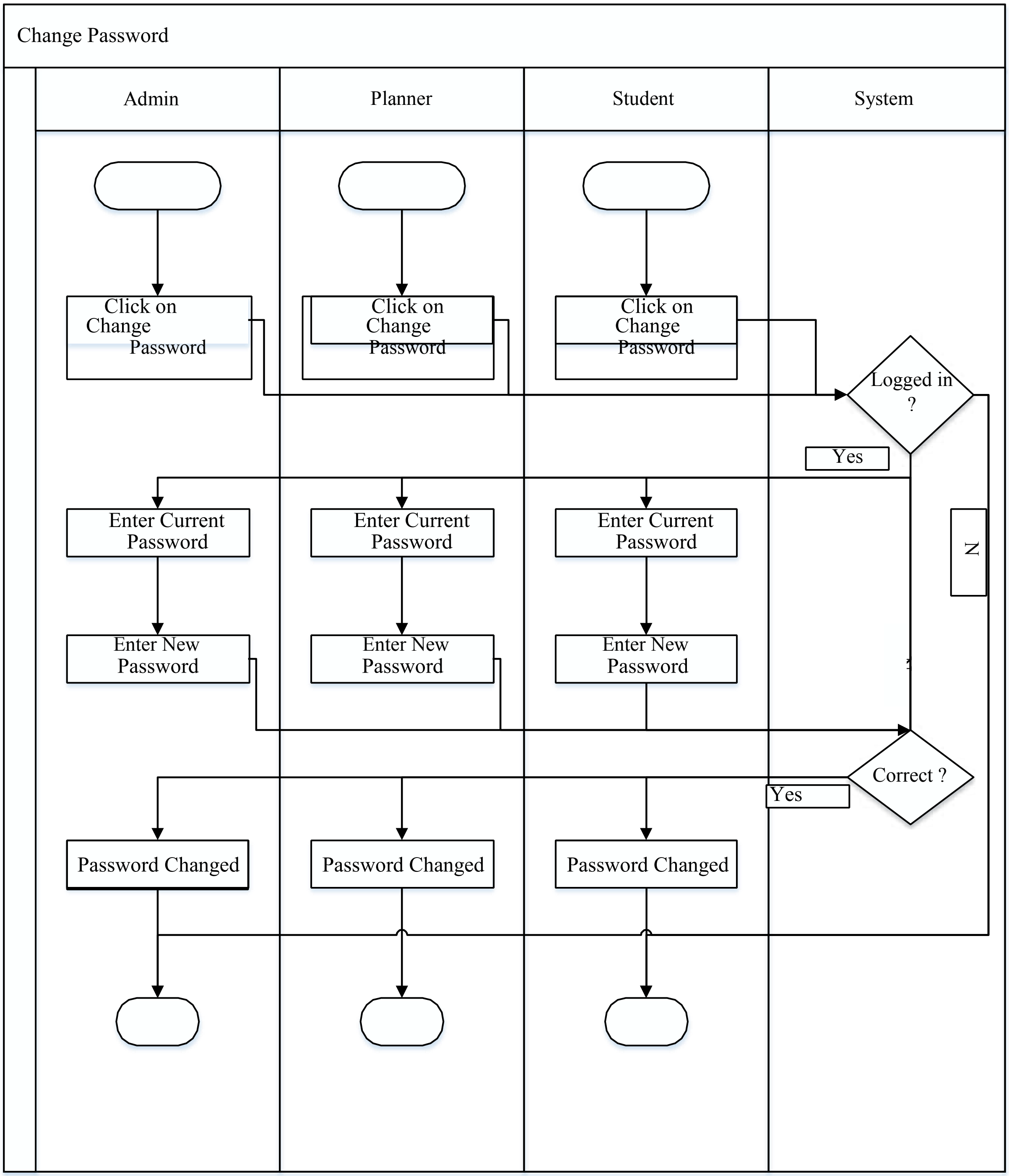

Figure 10 :Swim Lane Diagram – Change Password

2.3.2 Event

2.3.2.1 Create Event

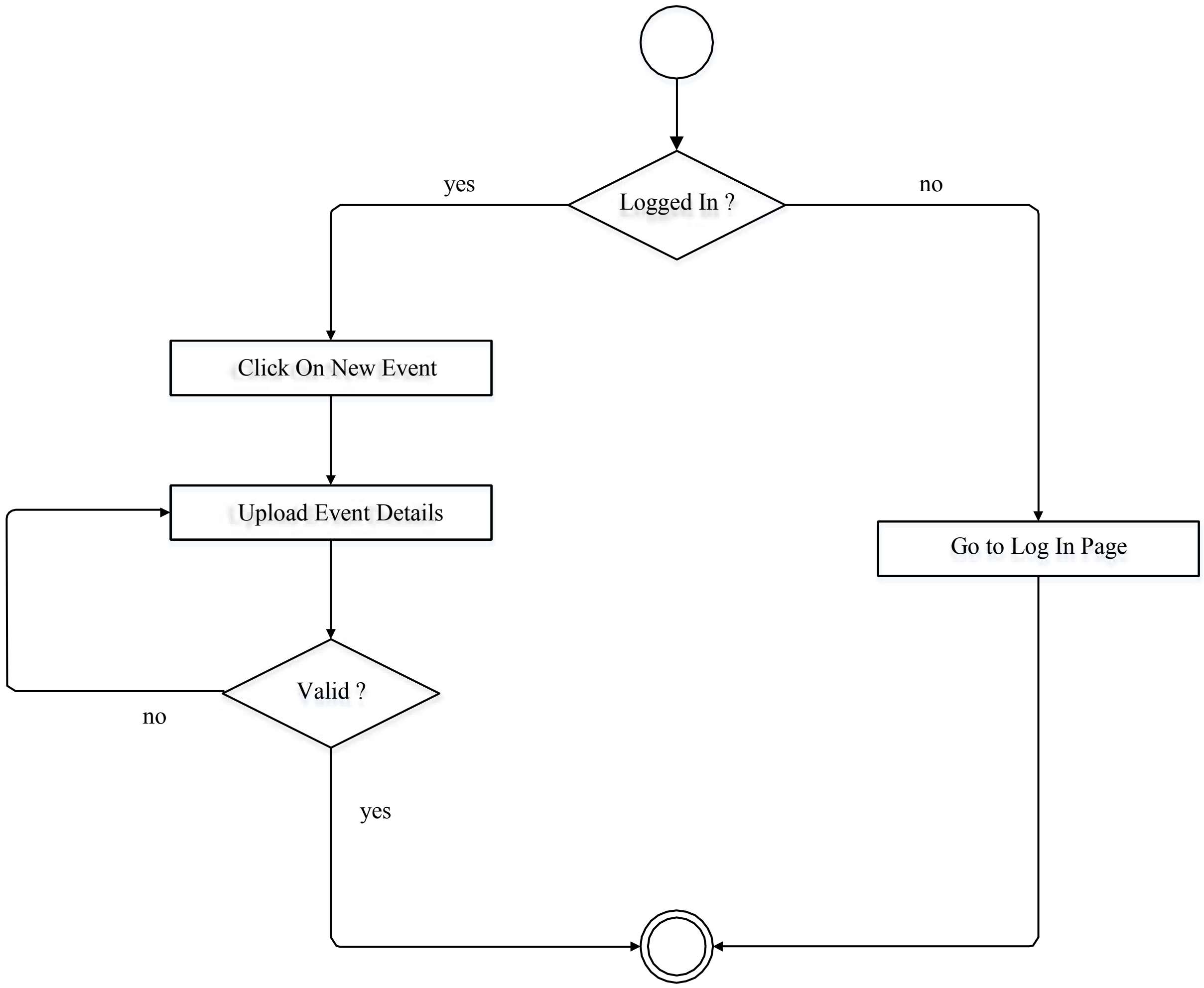

Figure 11 : Activity Diagram – Create Event

2.3.2.2 Create Client

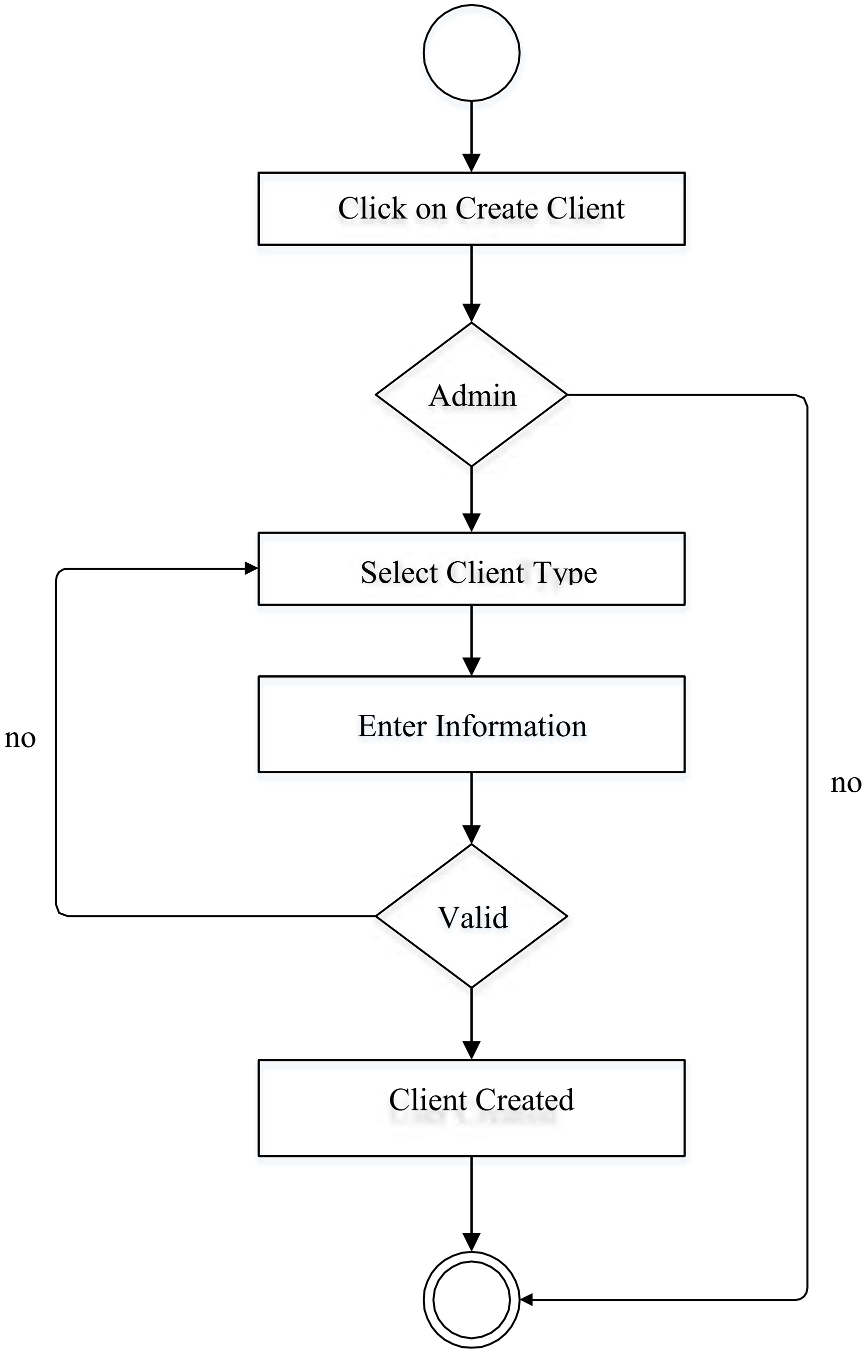

Figure 12 : Activity Diagram – Create Client

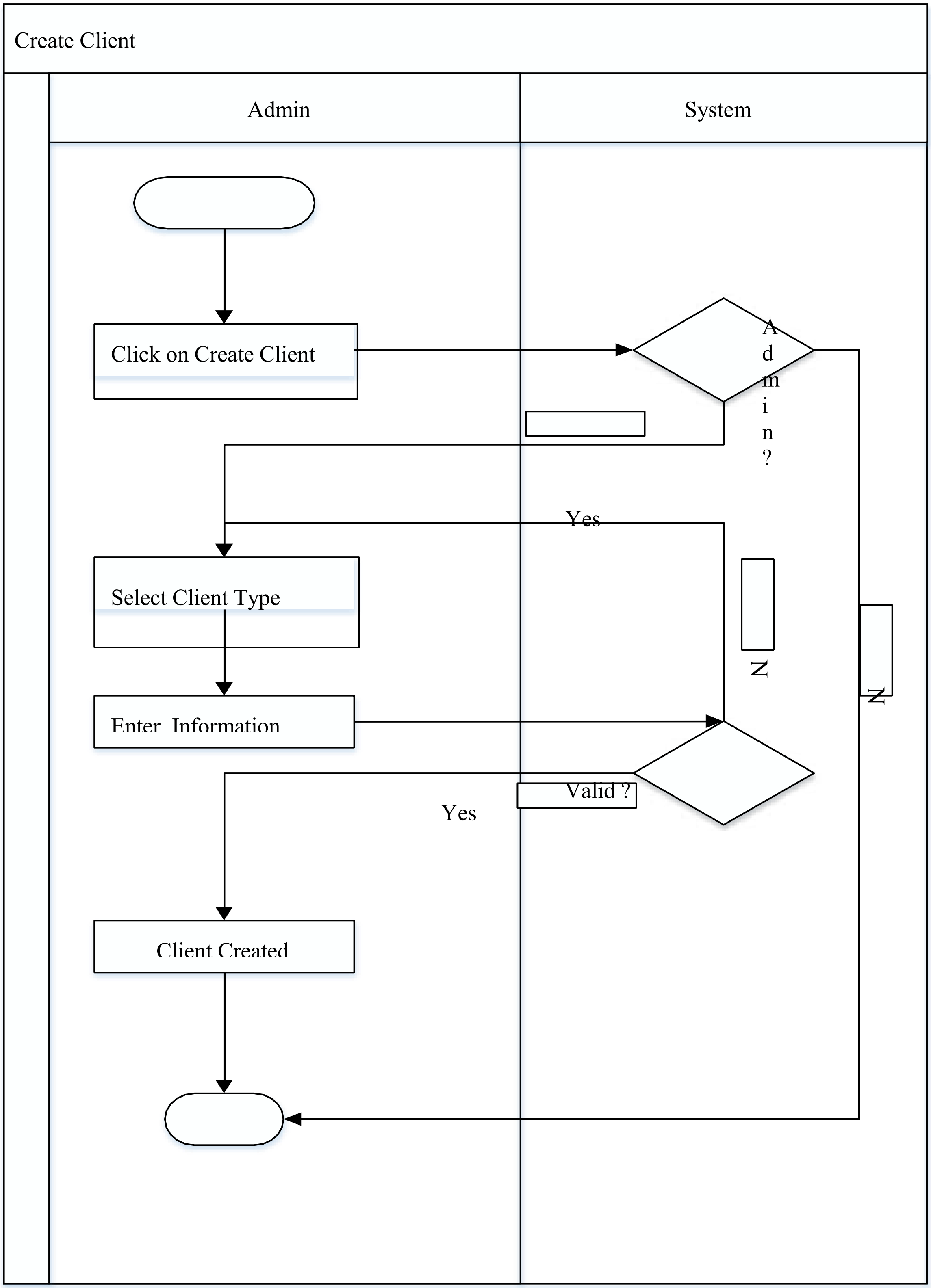

Figure 13 : Swim Lane Diagram – Create Client

2.3.2.3 Create Event

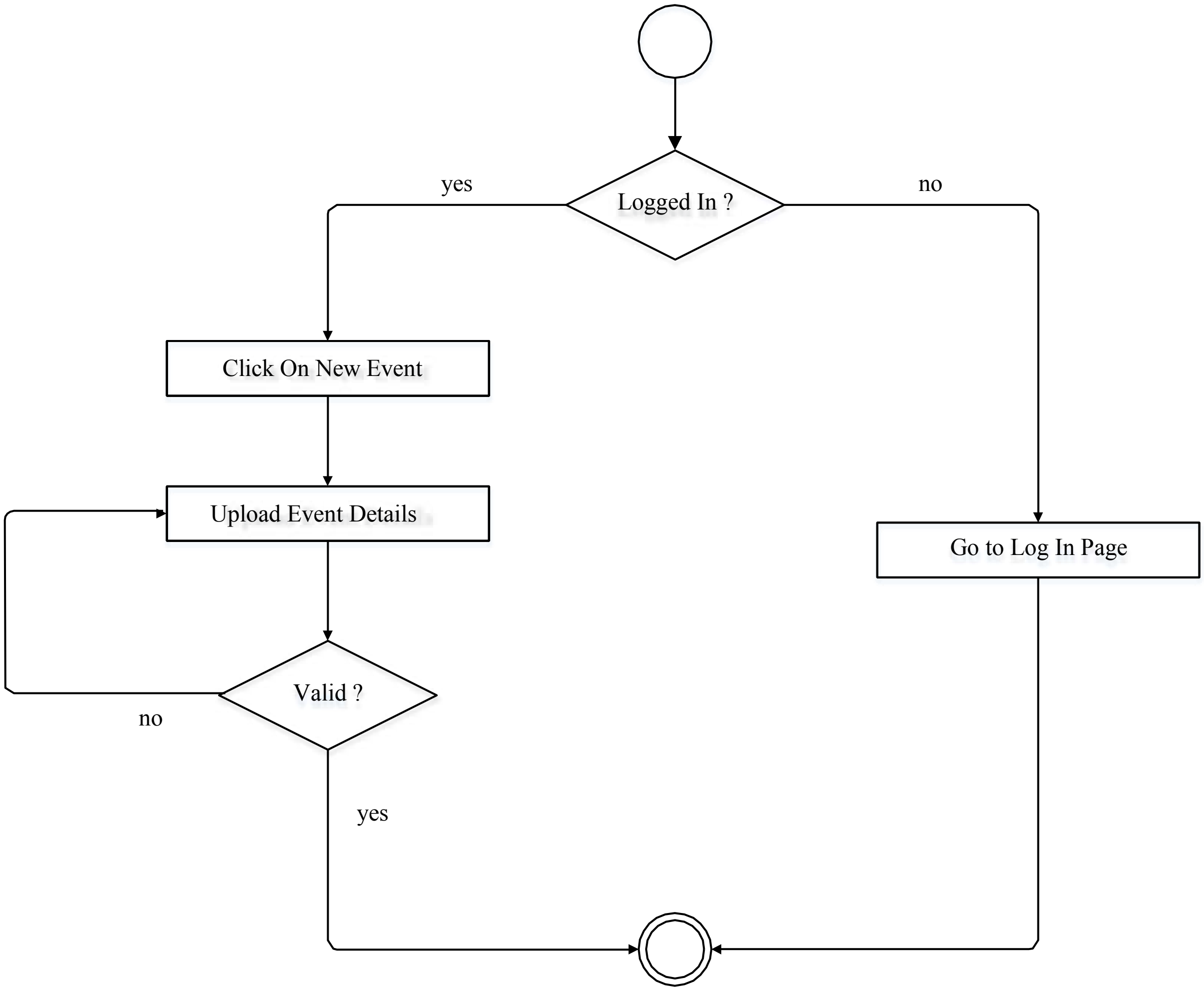

Figure 14: Activity Diagram – Create Event

2.3.2.3 Update Event

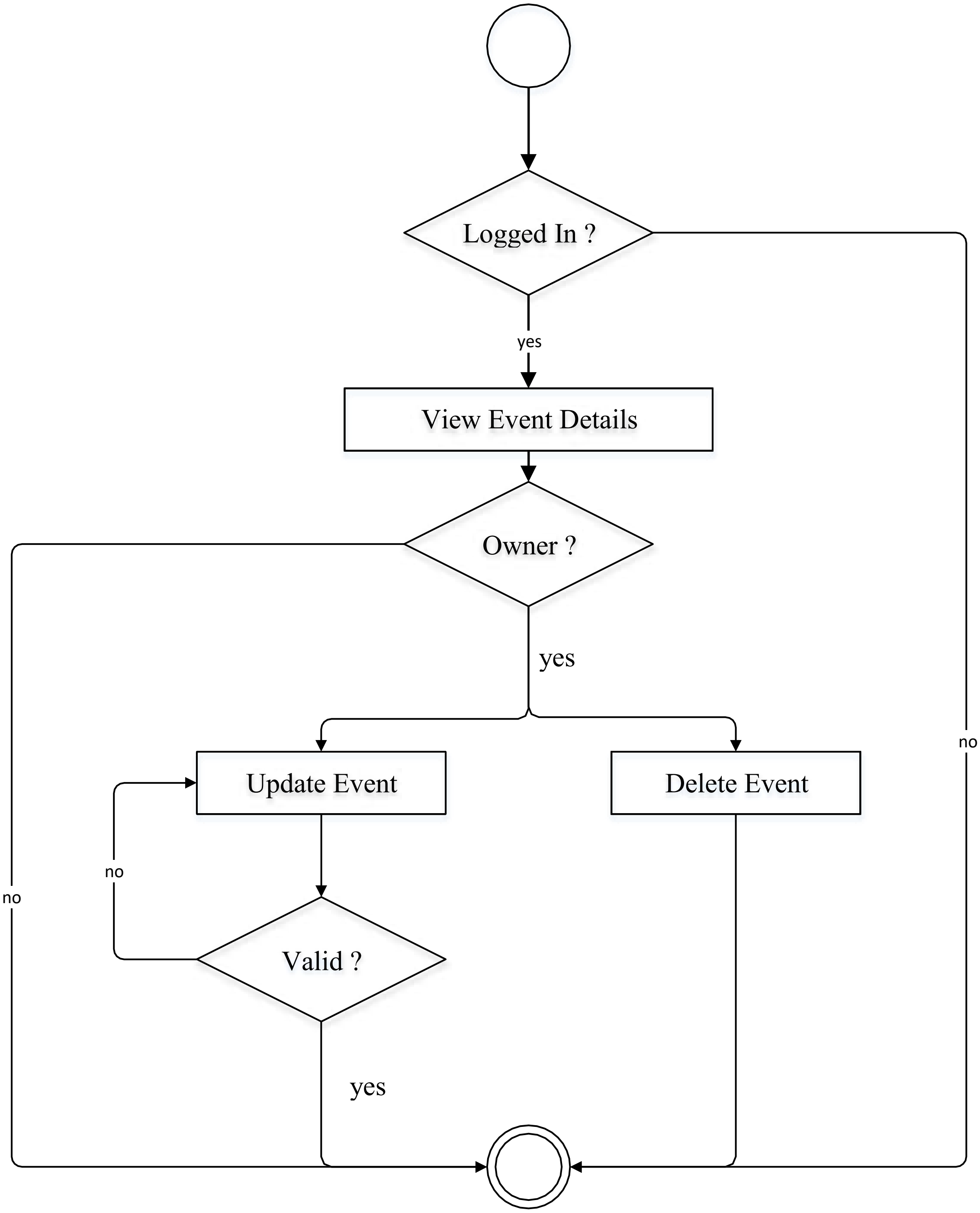

Figure 15: Activity Diagram – Update Event

2.3.3 Sub event

add and delete Sub event

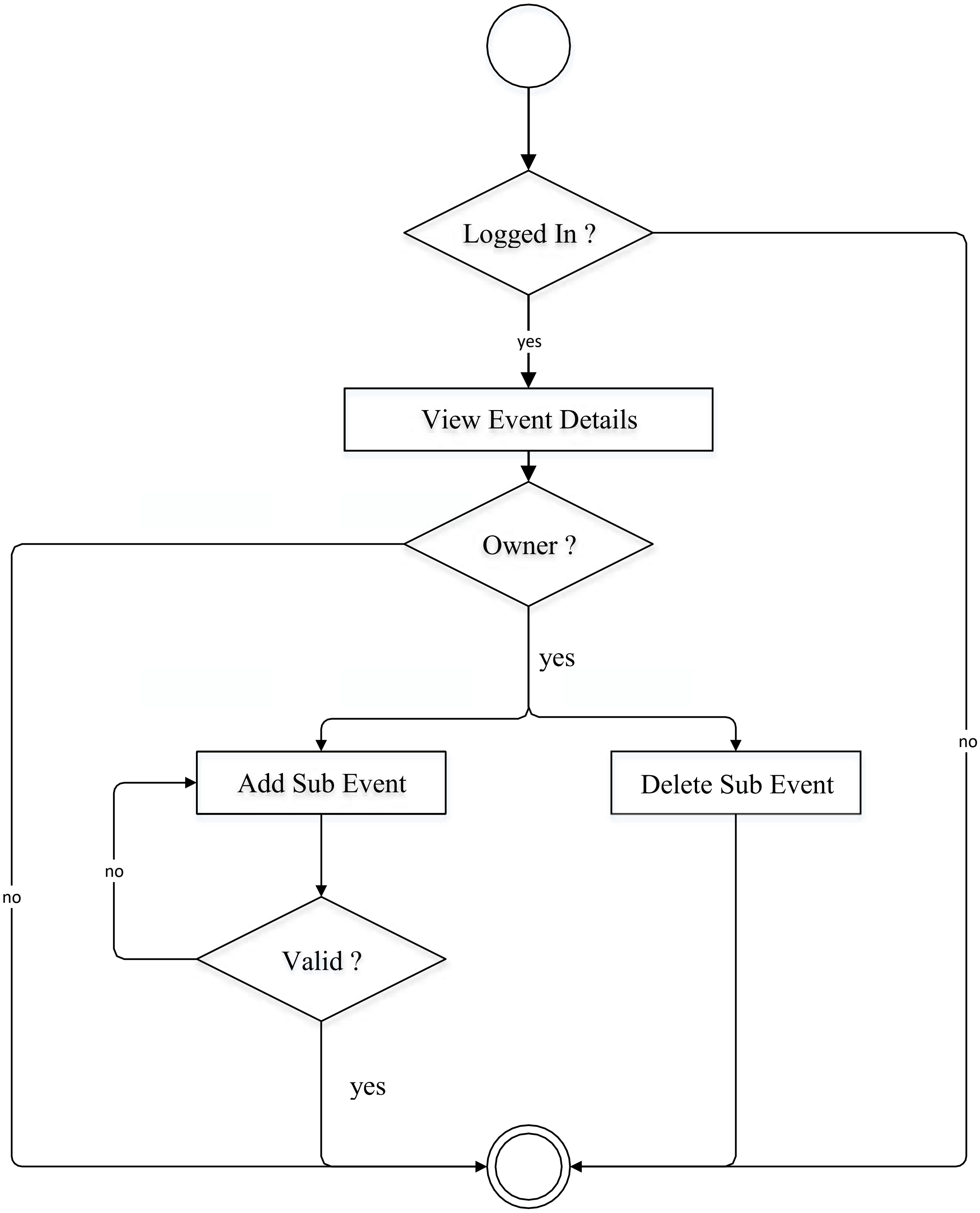

Figure 16 : Activity Diagram – Sub Event

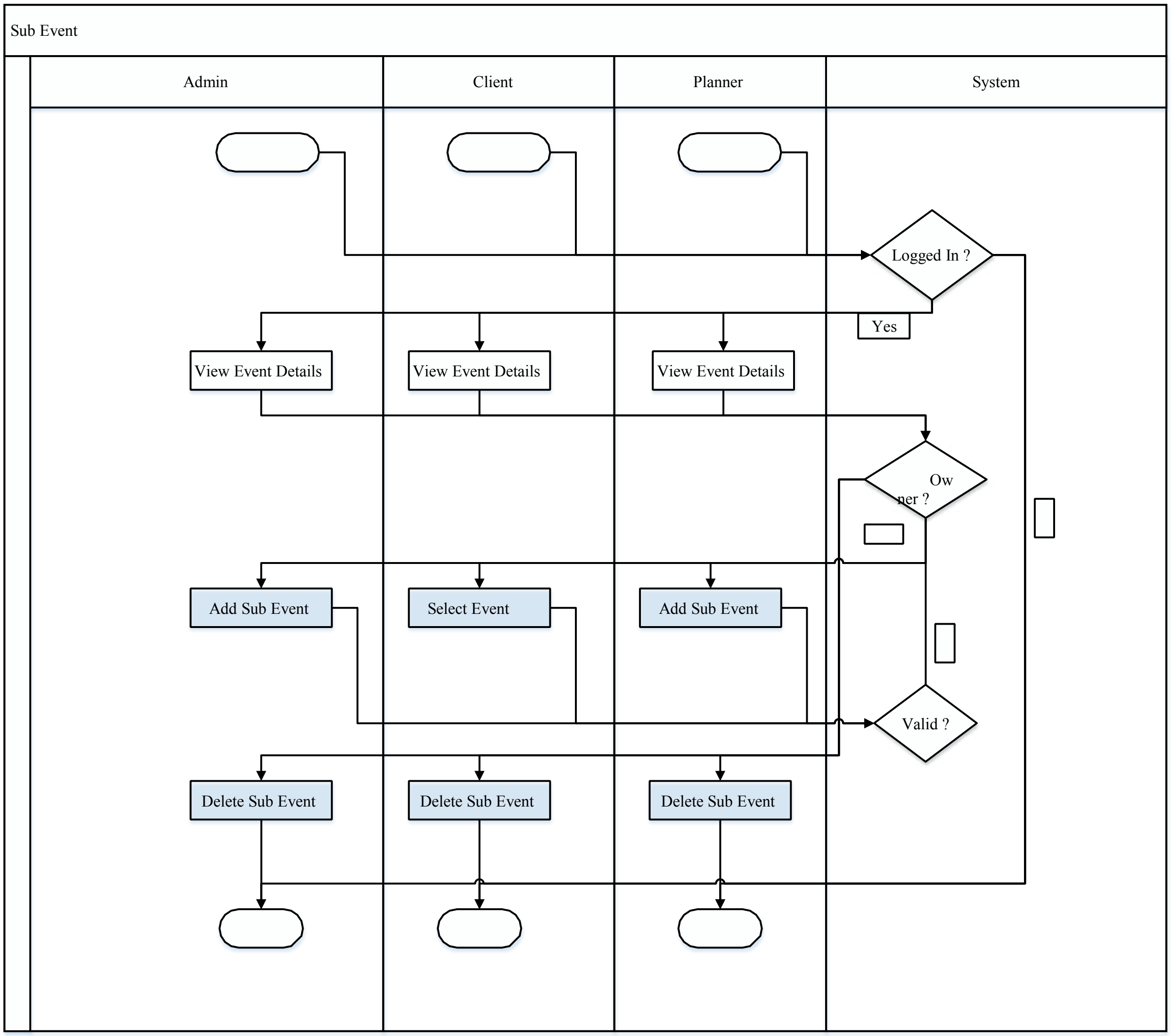

Figure 17 : Swim Lane Diagram – Sub Event

2.3.4 Template

Create Template

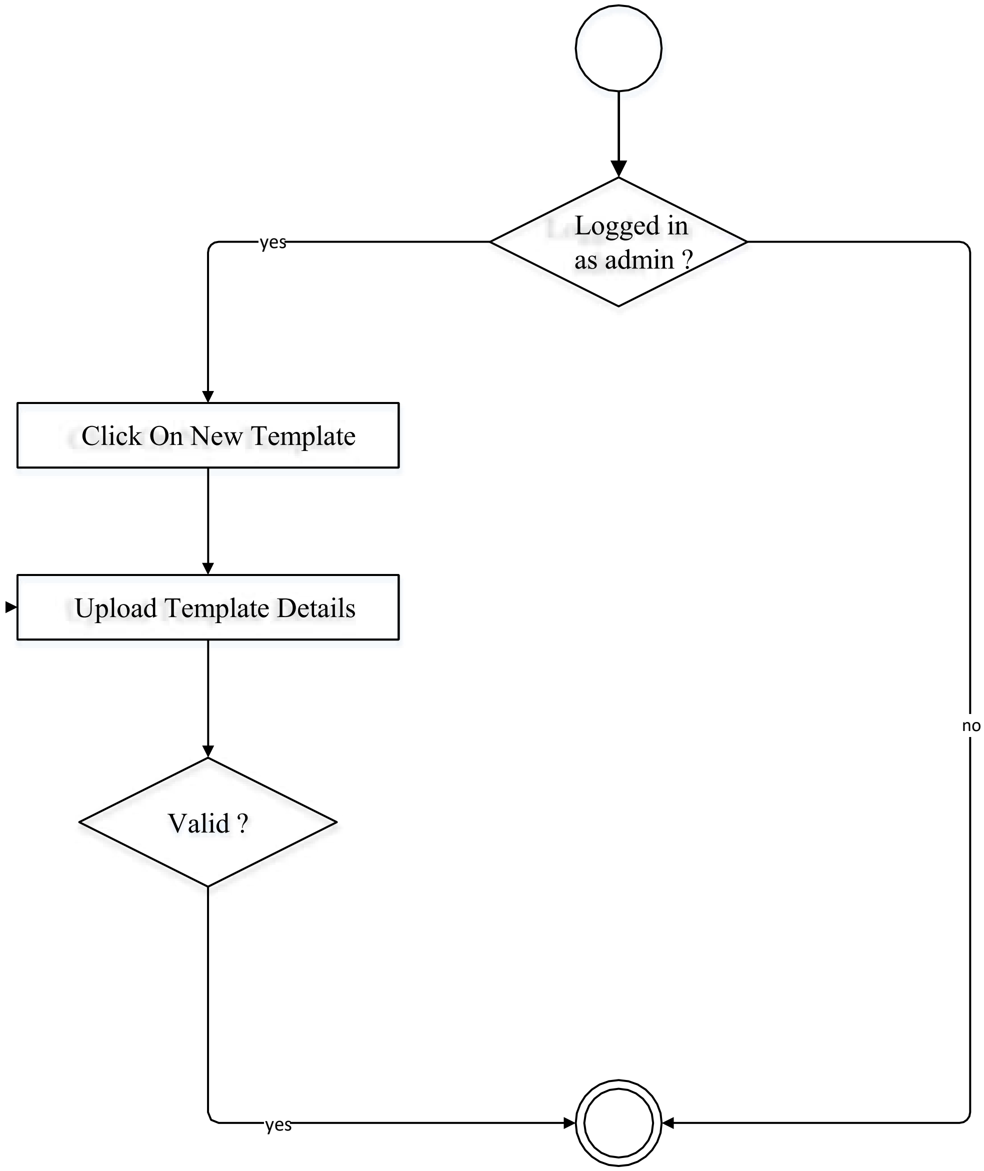

Figure 18 : Activity Diagram – Create Template

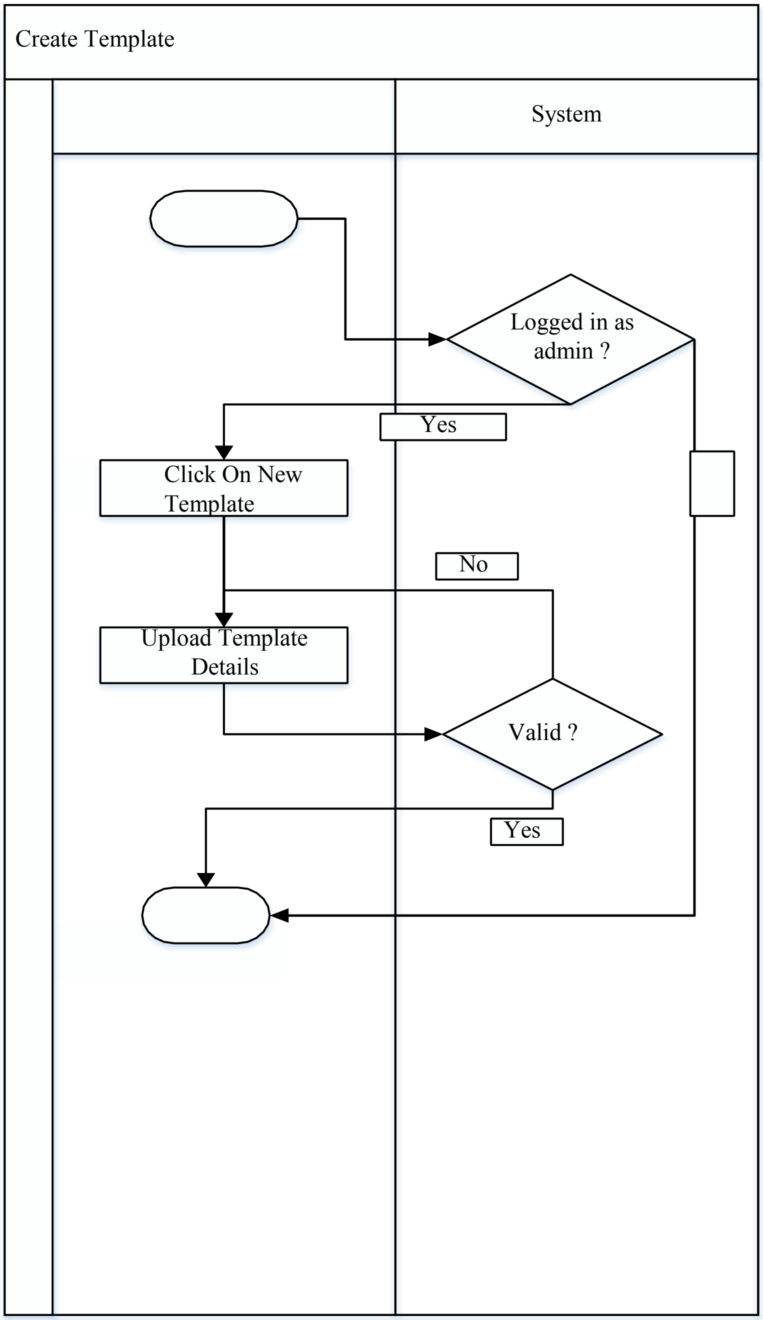

Figure 19: Swim Lane Diagram – Create Template

Chapter 4: Data Model

4.1 Introduction

If software requirements include the need to create, extend, or interface with a database or if complex data structures must be constructed and manipulated, the software team may choose to create a data model as part of overall requirements modeling.

4.2 Data Object Selection

A data object is a representation of information which has different properties or attributes that must be understood by software. Here is the table of potential data objects. Some attributes were not present in the scenario, but I included them as they are necessary attributes in this system.

Table 4.1: Data Object Selection

Noun	Attributes	Description	Remarks
Client	Name, ClientID, email, password, list (event)	Potential Data Object	Accepted
Name		An Attribute of Client, Event Manager, Admin	Rejected
Email		An Attribute of Client, Event Manager, Admin	Rejected
Password		An Attribute of Client, Event Manager, Admin	Rejected
ClientID		An Attribute of Client	Rejected
Admin	Name, designation, email, password, list (event url)	Potential Data Object	Accepted
Event Manager	Name, designation, email, password, list (event)	Potential Data Object	Accepted
Designation		An Attribute of Event Manager and Admin	Rejected

Event	Event name, description, list (sub-event), start date, duration, notification	Potential Data Object	Accepted

Client	
Attributes	Types
ClientID	Int (15)
Name	Varchar (30)
Email	Varchar (30)
Password	Varchar (30)

Admin	
Attributes	Types
Name	Varchar (30)
Designation	Varchar (30)
Email	Varchar (30)
Password	Varchar (30)

Event Manager	
Attributes	Types
Name	Varchar (30)
Designation	Varchar (30)
Email Password	Varchar (30)

Event	
Attributes	Types
Name	Varchar (30)
description	Varchar (500)
duration	Int (10)
start date	Date
notification	Boolean
list< Sub event>	

Template	
Attributes	Types
Template	Varchar (30)
Name	Varchar (30)
duration	Int (10)
list<Sub event of template>	

Sub event of template	
Attributes	Types
Name	Varchar (30)
Start	Int (10)
Duration	Int (10)

4.3 Relationship Between Data Objects

Here we have shown pair wise relation between two entities.

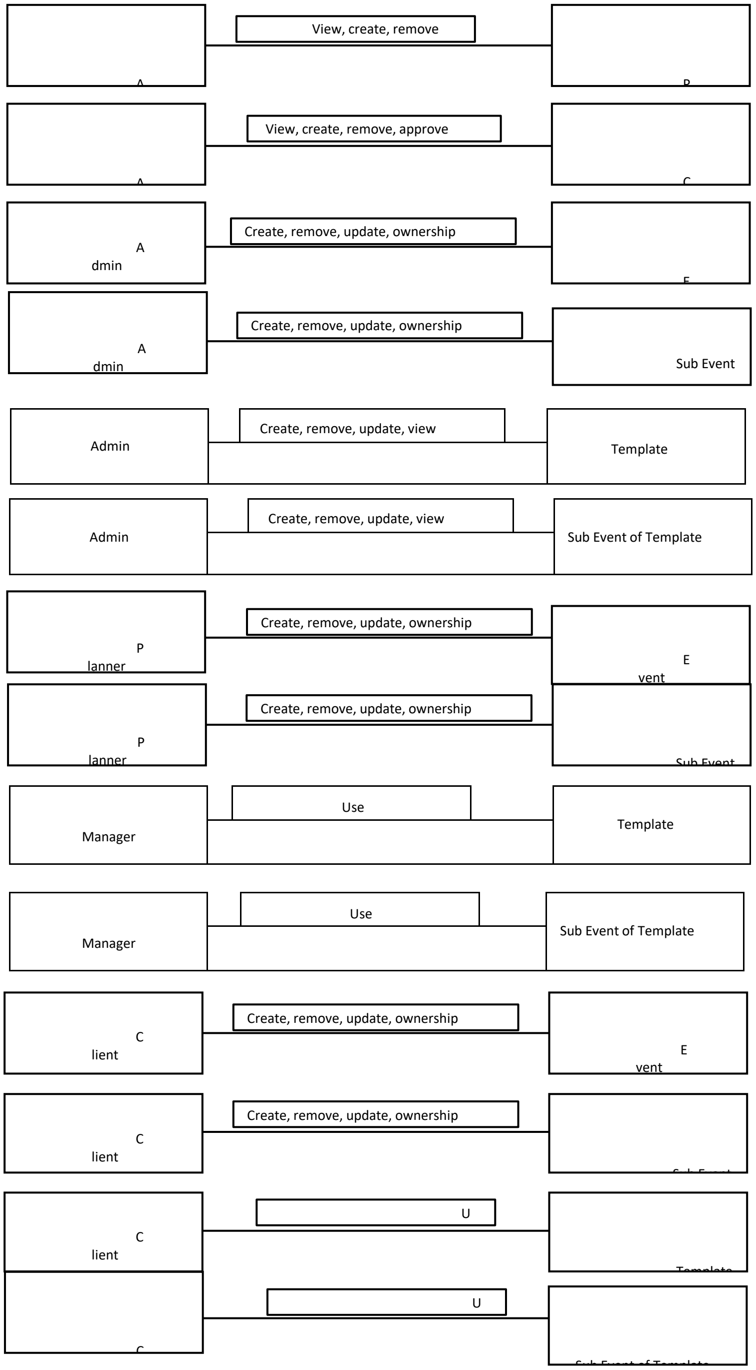
View, create, remove
View, create, remove, approve
A
dmin
Create, remove, update, ownership
A
dmin
Create, remove, update, ownership
Sub Event
Admin
Create, remove, update, view
Template
Admin
Create, remove, update, view
Sub Event of Template
P
lanner
Create, remove, update, ownership
E
vent
P
lanner
Create, remove, update, ownership
Manager
Use
Template
Manager
Use
Sub Event of Template
C
lient
Create, remove, update, ownership
E
vent
C
lient
Create, remove, update, ownership
C
lient
U
U

Figure 21 : Data Objects

4.4 Entity-Relationship(ER) Diagram

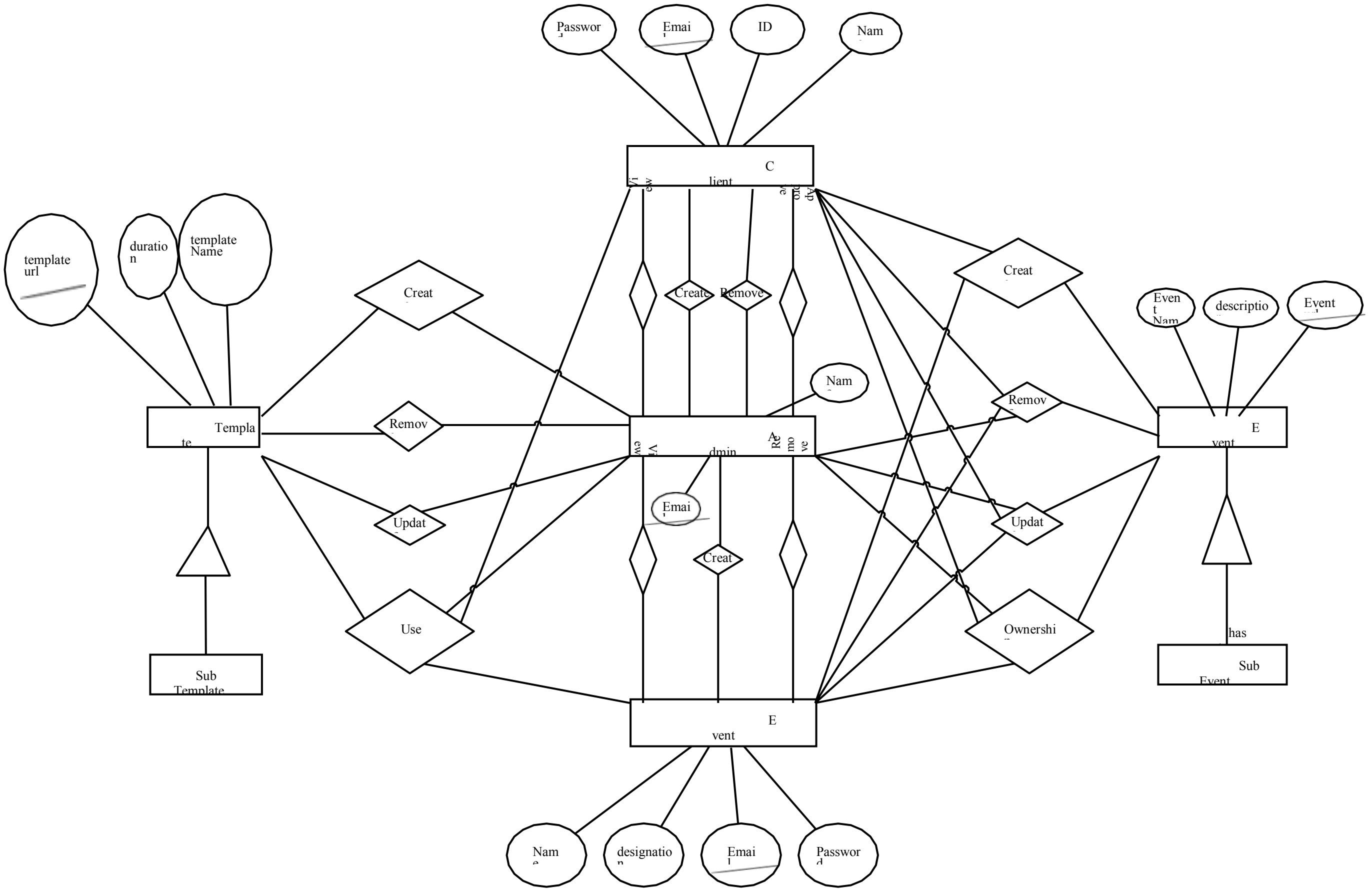

Figure 22 : ER Diagram

Chapter 5: Class Based Modeling

5.1 Introduction

Class based modeling is designed to demonstrate the whole software on the view or perspective of object-oriented concept. In this model what the objects are and what their responsibilities will be, how they will interact with each other is defined very clearly.

Event	Event name, description, start date, duration, list<sub Event>
Sub Event	Sub Event name, description, start, duration, Notification
Template	Template name, duration, list<sub Event of Template>
Sub Event of Template	Sub Event of Template name, duration, start
Database	All attributes

Event	
Attributes	Methods

Event name
Description
Start date
Duration
	Insert()
	Delete()
	Update()
	View()

Responsibilities	Collaborative Class
Viewing	Clients, Event Manager, Admin, Event, Template

5.2 CRC Diagram- **Class-Responsibility Collaborator**

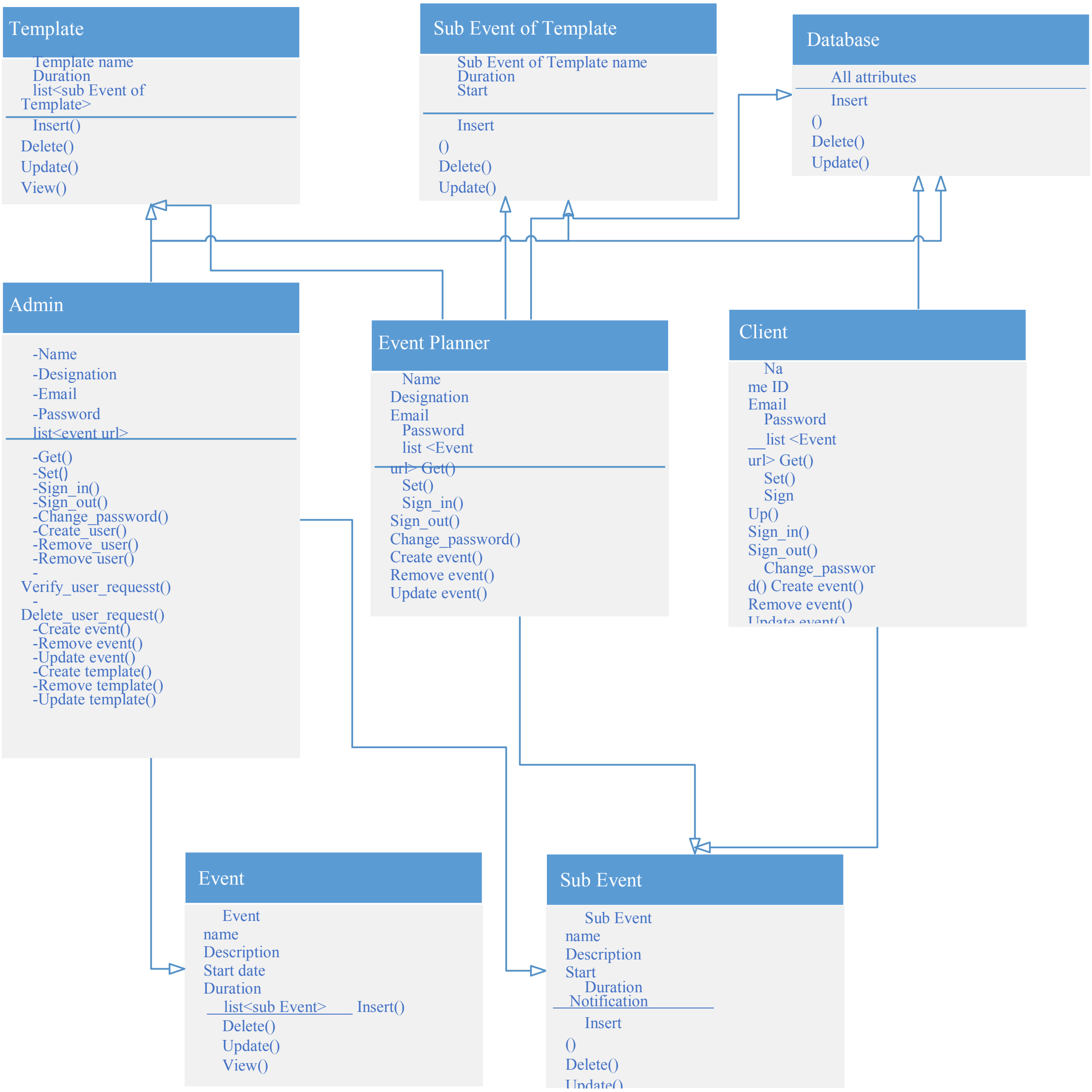

Fig 23 : CRC diagram

Chapter 6: Flow Oriented Model

6.1 Introduction

Although data flow-oriented modeling is perceived as an outdated technique by some software engineers, it continues to be one of the most widely used requirements analysis notations in use today.

6.2 Data Flow Diagram (DFD)

The Data Flow Diagram (DFD) takes an input-process-output view of a system. Data objects flow into the software, are transformed by processing elements and resultant data objects flow out of the software. Data objects are represented by labeled arrows and transformations are represented by circles.

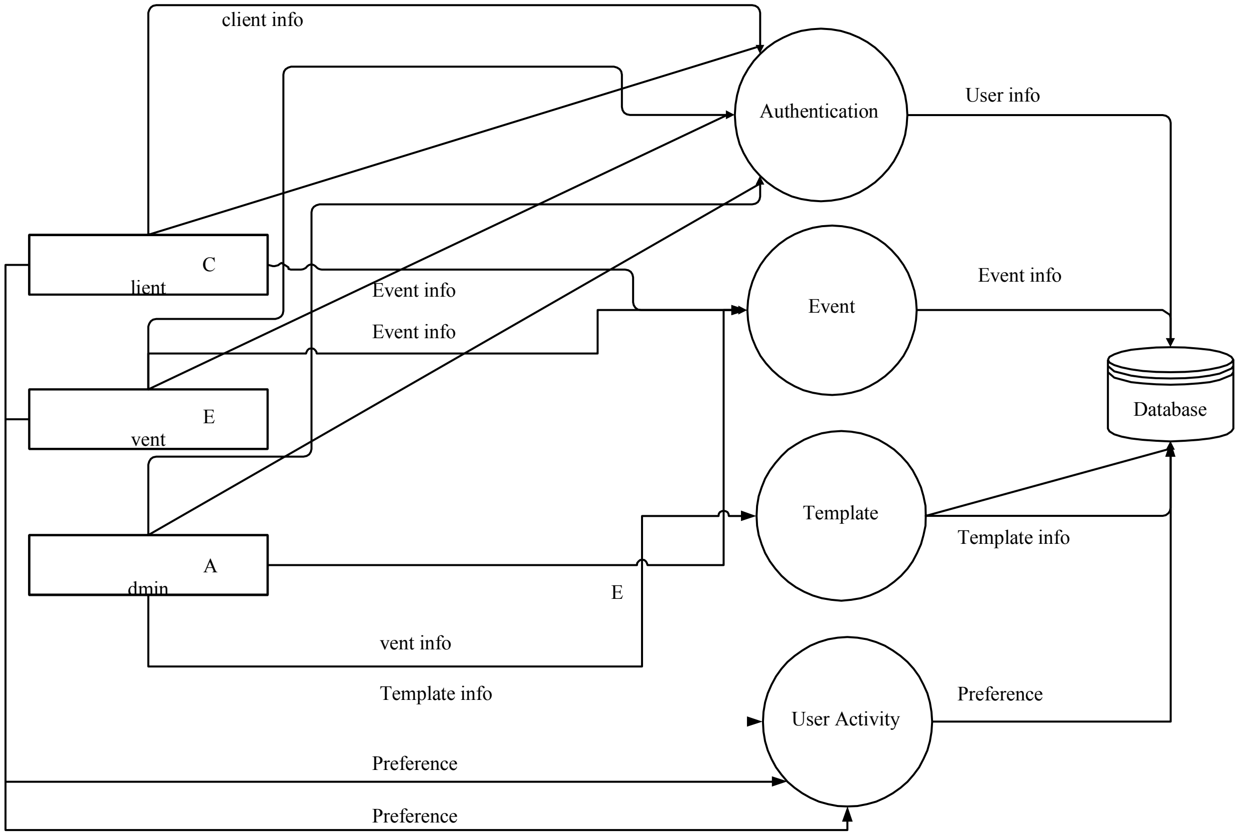

Figure 24 : DFD (Level- 1)

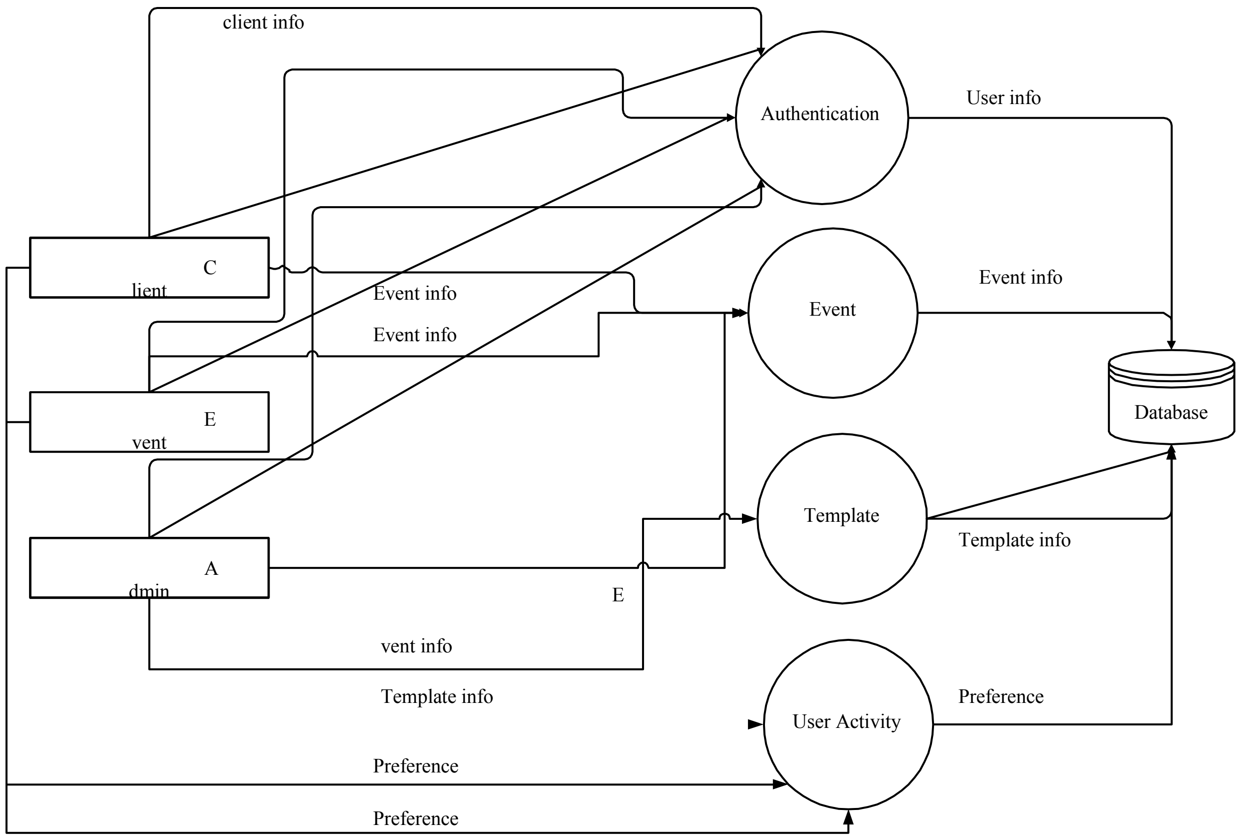

Figure 25 : DFD