<Online Ticket System>

Requirements Specification and Analysis

<1.0>

<17.11.2016>

HASAN MURAT AKPOLAT

EMRE YILMAZ

ERTAN GÖKYILDIRIM

MELİS KILIÇ

Prepared for

SE301 Software Engineering



Table of Contents

[1. Introduction 1](#_Toc467433011)

[1.1. Purpose of the System 1](#_Toc467433012)

[1.2. Scope of the System 1](#_Toc467433013)

[1.3. Objectives and Success Criteria of the Project 2](#_Toc467433014)

[1.4. Definitions, Acronyms, and Abbreviations 2](#_Toc467433015)

[1.5. Overview 2](#_Toc467433016)

[2. Current System 3](#_Toc467433017)

[3. Proposed System 4](#_Toc467433018)

[3.1. Overview 4](#_Toc467433019)

[3.2. Functional Requirements 5](#_Toc467433020)

[3.3. Nonfunctional Requirements 5](#_Toc467433021)

[3.4. System Models 8](#_Toc467433022)

[Scenarios 8](#_Toc467433023)

[Use case model 17](#_Toc467433024)

[Object model 30](#_Toc467433025)

[Dynamic model 30](#_Toc467433026)

[4. Glossary 43](#_Toc467433027)

[5. References 44](#_Toc467433028)

ONLINE TICKET SYSTEM

# Introduction

The following section provides an overview of the derived Requirement Analysis Document for the subject “Online Ticket System”. First of all, the aim of the analysis is presented and written as understandable for client, users and professions. Purpose, scope of the system, objectives and success criteria are specified below. This document also includes functional, non-functional requirements, system models, glossary and references.

## Purpose of the System

The main aim and objective of this system is to design and implement web based ticket system for show listings.

This application should allow users to:

1) View event listings;

2) Select events;

3) View event description;

4) Check availability;

5) Buy Tickets

Also, purpose of the system includes providing some functionalities to admin so that they can manage their events.

## Scope of the System

They system provides service to users,clients and admin. Thus, there are some common functionalities for both users, clients and admin, also some different functionalities for the both too.

In system for users; the users can list events, search event types like as concert, cinema and theatre. They can register to system and after they are done registering, users can list all events, and they can search event types and select an event. Before selecting events, users can determine date and region that they want.Users can buy tickets by using their credit cards.

In system for clients; the client can add events to system. Admin can update these events. In addition, clients can add time periods, seats, price, region and place to this event. Clients can add events which are only connected with their jobs in database.

In system for admin; admin can add new place, update this place. And also admin can add region into the system and admin can update this region. Also Admin can add event types into the system. Finally admin can cancel the event.

Online Ticket System provides to users ,clients and admin is; all users and clients can register to the system, and All of them(users ,clients and admin) can login the system. Also users , clients and admin can change their passwords.

## Objectives and Success Criteria of the Project

The main objective of our Online Ticket System is to provide another way for the users to buy online available tickets. It is an automatic system.

After inserting the data to database, Admin controls the event requests. And then clients need not todue with the order receive through the system. In fact, there is similar system on the Internet, but there is no refund method found in the existing system.

Online Ticket System is basically aimed to provide the user the complete information of the event , according to which the user can book the tickets and refund facility provides more flexibility to the system

The objectives of Online Ticket System are;

1. To provide a anytime anyplace service for the users
2. To minimize the number of staff at the ticket box
3. To promote the event on the internet
4. To increase the profit

Also Online Ticket System project gives secure ticket sytem, by providing safety means the user can complete the task with reliable information. And safety storage must be into the system. Because in the system users pay money with his/her credit card to buy ticket. Therefore safety is very important.

## Definitions, Acronyms, and Abbreviations

RAD: Requirements Analysis Document

PHP: Hypertext Preprocessor

SQL: My Structured Query Language

CSS: Cascading Style Sheets

## Overview

Requirements Specification and Analysis Document of Online Ticket System includes Current System section, Proposed System section, Overview of Online Survey section, Functional Requirements section, Nonfunctional Requirements section, System Models section, Object Model section, Dynamic Model section and Glossary.

In Online Ticket System section of our RAD documentation, we talked about functions and features of current Online Ticket System. We mentioned disadvantages of current system such as too long response time and unnecessary functions, and also we explained that the current system does not work easily.

**In Proposed System part**, we talked about our new Online Ticket System’s advantages and modifications. For example, we explained that our new Online Ticket System will be more basic interface and safety than current other available ticket systems.

**In Overview section**, we described overview of features, functions and details of Online Ticket System. In addition, we explained functions from takers and creators perspective

**In Functional Requirements part**, we talked about functions and features of Online Ticket System. And also, we described functions of Online Ticket System from users’, clients’, and admin’s perspective.

**In Nonfunctional Requirements section**, we mentioned nonfunctional parts of our new system such as usability, performance, reliability or availability and so on. We described all of nonfunctional requirements of Online Ticket System.

**In System Models part**, we described scenarios and use cases of Online Ticket System. We defined scenarios, actors, and use cases’ flow events and so on in this part.

**In Object Model section**, we explained class diagrams of our system and we defined relationships between classes of Online Ticket System.

**In Dynamic Model part**, we mentioned sequence diagram that is to say we explained methods and functions and their operations with actors of Online Ticket System.

**In Glossary section**, we described all things of Online Ticket System, it can call Online Ticket System dictionary. We explained all words in RAD documentation.

# Current System

Online Ticket System is a web-based online ticket system, it was developed for usage purpose of the all people to buy ticket online.

**The Users’ Mode**, the users can list events, search event types. They can register to system and after their registration request are approved, Users can show available events , and they can search event types and select an event. If the users do not register to the system, users can not select events and list of all events. Before selecting events, Users should login into system. After that Users can determine date and region that they want. Customers can buy tickets by using their credit cards. Customers can update their passwords registered on the system.

**The Client Mode**, Clients can add events to system. Client can update these events. In addition, Clients can add region to events. Also they can add seats, price, time period and place to this event.

**The Admin Mode**, Admin can add new place, region, and event type into the system. Also Admin checks the registration request and approve or cancel that requests. In addition Admin can modify the event information and cancelation of events. Also Admin can update the places which are found into database.

The main important feature is safety credit card payment in Online Ticket System. Users can make payments with 3-D secure payment technology.

# Proposed System

The Online Ticket System is developed with the aim to overcome the drawbacks of existing system. The proposed system has got many advantages. People from different parts of the world can register very easily. This system is more personalized. It is maze in such a manner that all the new users can understand all the options in it very easily. It is made in a quick and easy referential manner. This system helps the user to go through the rates quoted by different travel agencies and select the convenient rate that is suitable for him. Access to all important matters are not always locked and can be opened easily at the time of urgency. The advantages of proposed system are that security is maintained in the new system. Securities for all important data are maintained confidentially. As it is easily understandable and user friendly, quick entries can be made in this system system

## Overview

Online Ticket System is web-based ticket system for all people/clients. Users can use Online Ticket System to buy a ticket online. Users can buy their event tickets easily and quick through this system. In addition, Users register to system by using Online Ticket System. Client uses Online Ticket System to offered some events to users. Online Ticket System has common and different functionalities for users. Members of the system are users, clients and admin. They have some common and different functions. For example, All of the members can login on same screen. Users and clients can register to the system. Also Users, Clients and Admin can change their passwords. These are common for both of them. However, Clients can add events. And also Clients can add categories, seats, price, time period and region to any event. In Online Ticket System, Users can list events, search event types They can register to system and after they are done registering, Users can list all events and they can search event types and select an event. Before selecting events, Users can determine date and region that they want. Also Users can buy tickets by using their credit cards. Online Ticket System allows control and manage web server like as a bank. The bank responses positive or negative result to buy ticket for users. After the payment process, the system gives a message for payment. (Payment is successful, No Limit…. etc.)

Online Ticket System will be more understandable and friendly for the users. For example, users can buy a ticket by using basic interfaces. And also, response time of the system will not be long, so the users can use Online Ticket System more efficiently .For example buying a ticket time will be faster than the other ticket systems.

## Functional Requirements

Online Ticket System is a web based ticket platform for all members.

In Users side, Online Ticket System has several functions. For instance, users can list all events, search event types, search regions, selects events, select categories and seats by using Online Ticket System. Also Users can change password by using Online Ticket System.

In Client Side, Online Ticket System has different functions. For example, Client can add events, seats, price, region, time period and place to event. Clients can update region, event types, time period, place and event by using Online Ticket System.

In Admin side, Admin can add places, update these places. Also admin can add region into the system. Also Admin can add event types into the system. The main important thing in Online Ticket System is admin’s permission about client functions.Using Admin’s permission, Clients can add event which is only permitted category.Another Admin function is “Modifying Event” function.In this function,Admin can update the information about events.Finally, Admin can control the registration on the system. Admin checks the registration status of Clients, If everything is okey,Admin approves the account and Client can login into the system. User registration does not require the admin permission.

The main and important function of Online Ticket System is buying online ticket with their credit card. Users use Online Ticket System to buy tickets with their credit cards.

## Nonfunctional Requirements

A non-functional requirement is a requirement that specifies criteria that can be used to judge the operation of a system, rather than specific functions.

**3.3.1 Usability**

|  |
| --- |
| **Function Description** |
| The user is able to buy an event ticket with just few clicks. |
| Online payment session uses the high secure 3D payment function. |
| System warns users about the credit-card payment by e-mail. |
| User cannot sign up the system without e-mail, password and phone number informations. |

**3.3.2 Reliability**

|  |
| --- |
| **Function Description** |
| The user password length must be between 8 digits with integer and string together. |
| A system crash should not seen with any reservation lost. |
| It cannot possible to register two times with the same email/username. |

**3.3.3 Performance**

|  |
| --- |
| **Function Description** |
| System must send ticket update to clients . |

**3.3.4 Supportability**

|  |
| --- |
| **Function Description** |
| User can show the complete reservations on the “My Reservations” page. |
| Clients can log on the system with admin’s registration permission |
| If there is no available seat in selected event, The system should inform the users |
| If the User or Client forget his/her password, system should send an email to user including a temporary password to refresh the password. |
| The system sends an e-mail after the approved client registration . |

**3.3.5 Implementation**

* The Database is implemented in MySql WorkBench 6.3 CE.
* The Programming is done with Net Beans.

**3.3.6 Interface**

Online Ticket System use interfaces to provide accesses to our system. In Online Ticket System,using interfaces are;

\* Register Page

\*Login Page

\*Change Password Page

\* User Home Page

\*Search Event Page

\*MyReservations Page

\* Payment Page

\*Client Home Page

\*Client Registration Form

\*Admin Page

\*Check Registration Page

**3.3.7 Packaging**

The system must be available on both of desktop and mobile devices and it performs properly on every browser.

**3.3.8 Legal**

* For users the system is free to use.
* For Clients, the system must be licensed.

## System Models

This section includes scenarios, use cases, sequence diagrams and state machine diagrams of these four functions above mentioned. Also, the whole class diagram and use case diagram is shown below.

### Scenarios

A scenario is an instance of a use case.

**Scenario 1:**

|  |
| --- |
| *Scenario name:* Update Profile |
| *Participant actor instances:* Sasha: User |
| *Flow of events:* 1. Sasha sends a request to Online Ticket System to update his/her personal information which are his/her identification, and address, credit card Information.  2. Online Ticket System offers the “Personal Information Form” by using the “User” table. Then, displays it on the screen so that the user can update fields he/she wishes to.  3. Sasha edits the fields he/she wants, then he/she submits the form by using the update button to be updated.  4. Online Ticket System updates the information into the “User” table. Finally, it displays the “Acknowledgement Notice” to the user. |

**Scenario 2:**

|  |
| --- |
| *Scenario name:* Select Category |
| *Participant actor instances:* Sasha: User |
| *Flow of events:*  1.Sasha logs into the Online Ticket System.  2.Sasha select region on the event page.  3.Sasha select the category filter for events.  4.And Sasha can show events of selected category in selected region |

**Scenario 3:**

|  |
| --- |
| *Scenario name:* Select Ticket Quantity |
| *Participant actor instances:* Sasha: User |
| *Flow of events:*  1. Sasha logs into the Online Ticket System.  2. Sasha selects region on the event page.  3. Sasha selects the category filter for events.  4. And Sasha selects “SelectTicketQuantity” button for the quantity of ticket |

**Scenario 4:**

|  |
| --- |
| *Scenario name:* Select Region |
| *Participant actor instances:* Sasha: User |
| *Flow of events:* 1. Sasha opens the Online Ticket System web site.  2. Sasha lists all regions  3. Sasha selects any region what she/he wants. |

**Scenario 5:**

|  |
| --- |
| *Scenario name:* Choose Seat |
| *Participant actor instances:* Sasha: User |
| *Flow of events:*  1.Sasha logs into the Online Ticket System.  2.Sasha selects region on the event page.  3.Sasha selects the category filter for events.  4.And Sasha selects “SelectTicketQuantity” button for the quantity of ticket  5.If there is available seat for selected ticket quantity, Sasha can select his/her seat on system. |

**Scenario 6:**

|  |
| --- |
| *Scenario name:* Register |
| *Participant actor instances:* Sasha: User |
| *Flow of events:* 1. Sasha opens Online Ticket System web site.  2. Sasha clicks register button.  3.After Sasha writes the information about account  4.Sasha clicks the “Register” button and session completes |

**Scenario 7:**

|  |
| --- |
| *Scenario name:* Add Seats to Event |
| *Participant actor instances:* Danny :Client |
| *Flow of events:* 1. Client logs into the Online Ticket System.  2. Client selects an event and adds seats to this event.  3. Client adds seats into this categories and events. |

**Scenario 8:**

|  |
| --- |
| *Scenario name:* Add Price to Event |
| *Participant actor instances:* Danny :Client |
| *Flow of events:* 1. Danny logs into the Online Ticket System.  2. Danny selects an event and adds categories to this event.  3. Danny adds price into this event. |

**Scenario 9:**

|  |
| --- |
| *Scenario name:* Add Time Period of Event |
| *Participant actor instances:* Danny :Client |
| *Flow of events:*  1. Danny logs into the Online Ticket System.  2. Danny selects an event and clicks on “AddTimePeriod” button.  3. Danny adds valid information about time period of event. |

**Scenario 10:**

|  |
| --- |
| *Scenario name:* Add Place to Event |
| *Participant actor instances:* Danny :Client |
| *Flow of events:* 1. Danny logs into the “Add Event” button.  2. Danny selects an event.  3. Danny adds place to this event. |

**Scenario 11:**

|  |
| --- |
| *Scenario name:* Add Region to Event |
| *Participant actor instances:* Danny :Client |
| *Flow of events:* 1. Danny logs into the “Add Event” button.  2. Danny selects an event.  3. Danny adds region where event is done to this event. |

**Scenario 12:**

|  |
| --- |
| *Scenario name:* Add Type of Event |
| *Participant actor instances:* Jesse:Admin |
| *Flow of events:*  1.Jesse logs into the “Add Event” button.  2. Jesse clicks AddTypeOfEvent Button and then the admin adds new types into the system.  3. Online Ticket System receives this request and displays form that new type is added. |

**Scenario 13:**

|  |
| --- |
| *Scenario name:* Update Place |
| *Participant actor instances:* Jesse:Admin |
| *Flow of events:*  1. Jesse logs into the Online Ticket System.  2. Jesse selects a place and then updates this places’s information. |

**Scenario 14:**

|  |
| --- |
| *Scenario name:* Add Region |
| *Participant actor instances:* Jesse:Admin |
| *Flow of events:*  1. Jesse logs into the Online Ticket System.  2. Jesse clicks “AddRegion” button and adds a region into the system.  3. Online Ticket System receives this request and displays form that new region is added. |

**Scenario 15:**

|  |
| --- |
| *Scenario name:* Remove Event |
| *Participant actor instances:* Jesse:Admin |
| *Flow of events:*  1. Jesse logs into the Online Ticket System.  2. Jesse Jesse selects event which is She wants to remove .  3.Jesse clicks on “RemoveEvent” button |

**Scenario 16:**

|  |
| --- |
| *Scenario name:* Modify Event |
| *Participant actor instances:* Jesse:Admin |
| *Flow of events:*  1. Jesse logs into the Online Ticket System.  2. Jesse selects event which is he/she want to modify.  3.Jesse clicks on “ModifyEvent” button |

**Scenario 17:**

|  |
| --- |
| *Scenario name:* Confirm Registration |
| *Participant actor instances:* Jesse:Admin |
| *Flow of events:*  1. Jesse logs into the Online Ticket System.  2. Jesse clicks on “Registration” page.  3.Jesse shows the registration requests on the screen  4.Jesse can approve or disapprove the requests  5. Jesse clicks on “Registration update” button and updates registration status of accounts. |

**Scenario 18:**

|  |
| --- |
| *Scenario name:* Add Event |
| *Participant actor instances:* Danny : Client |
| *Flow of events:* 1. Danny wants to add an event to the system. Danny first logs in to Online Ticket System.    2. Danny chooses add event button on the screen. Then, Danny chooses one region which system offers.  3. Then Danny chooses place, event type and date for this event. Danny adds categories and event’s price to the system.  4. After Danny adds all this information to the system, Danny enters “Finish” button.  5. Online Ticket System receives this event and saves it into the “Event” table which contains information of events.  6. Danny receives an acknowledgement indicates that the process has been completed successfully. |

**Scenario 19:**

|  |
| --- |
| *Scenario name:* ShowEvents |
| *Participant actor instances:* Sasha : User |
| *Flow of events:* 1. Sasha wants to go to concert. Therefore Sasha opens the Online Ticket System web site. Sasha first logs in to Online Ticket System.  2. Sasha selects a region.  3. Sasha selects date and place what he/she wants.  4. Sasha selects event type.  5. Online Ticket System offers the events to her. All filtered event list seen on the screen.  6. Online Ticket System receives the request, then responds by offering list. |

**Scenario 20:**

|  |
| --- |
| *Scenario name:* BuyTicket |
| *Participant actor instances:* Sasha:User |
| *Flow of events:* 1. After Sasha lists all events which filtered by Sasha, Sasha clicks any event.  2. Sasha enters “Buy Ticket” button.  3. System wants to Sasha to select which how much ticket.  4. System also wants to Sasha to select seat.  5. Sasha select ticket quantity, seat and enters continue button.  6. System wants to Sasha to write her credit card information.  7. Sasha writes all information about her card. Then Sasha enters continue button.  8. Sasha receives an acknowledgement indicates that the process has been completed successfully. |

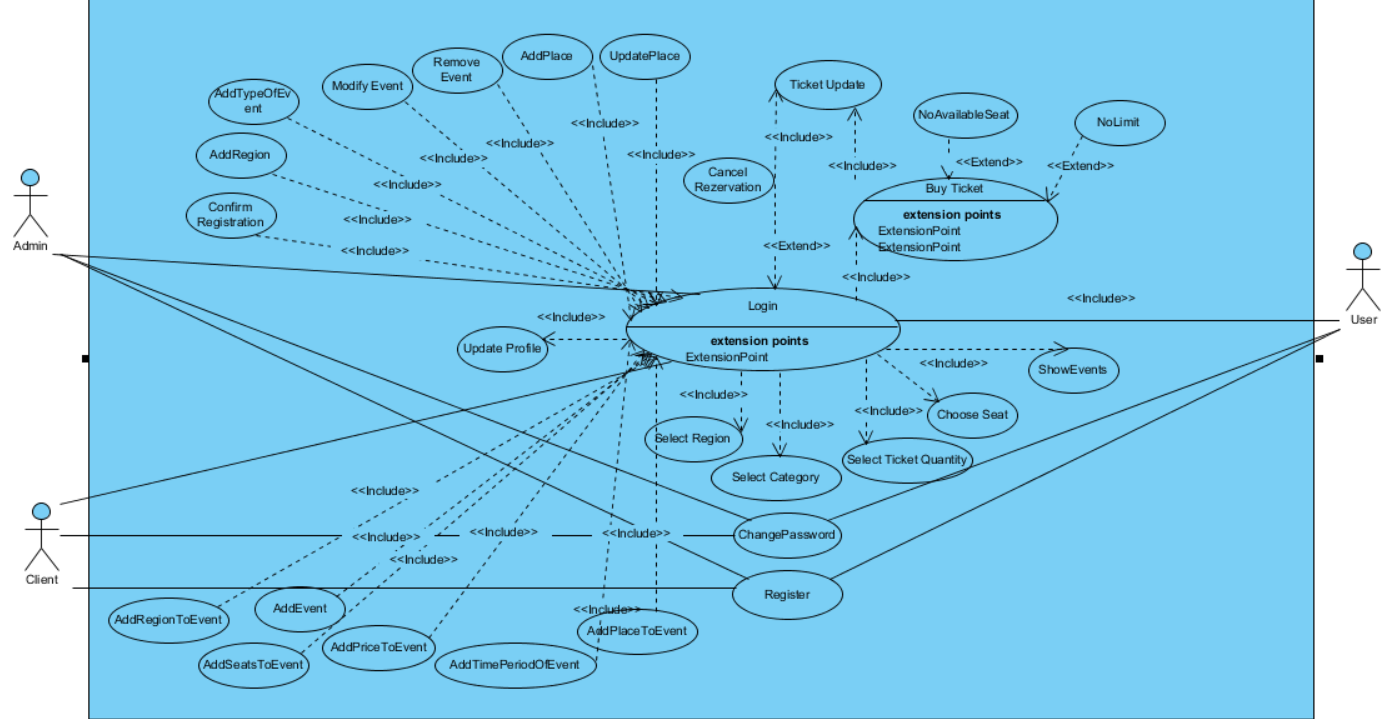
**Scenario 21:**

|  |
| --- |
| *Use case name:* ChangePassword |
| *Participant actor instances:* Sasha:User |
| *Flow of events:* 1. Sasha thinks his/her password is not safe enough so he thinks about changing it.  2. Sasha enters to Online Ticket System and logins with his informations.  3. Online Ticket System checks her password and user id in User Table.  4. Sasha selects Change Password option.  5. Online Ticket System fetches request and displays a form with four fields, which are “current password”, “new password”, “confirm new password” and “Change Password Button”.  6. Sasha types her current password and types her new password, Sasha confirms new password by typing again. Then submits the form through the submit option.  7. Online Ticket System receives request and changes Sasha’s password field on User Table.  8. Sasha views “Password has been changed' message on the screen. |

**Scenario 22:**

|  |  |  |  |
| --- | --- | --- | --- |
| |  | | --- | | *Use case name:* AddPlace | | *Participant actor instances:* Jesse : Admin | | *Flow of events:* 1. Jesse hears that there is a place in Marmara Region. This place is not in system now.  2. Jesse talks to this place’s boss to add this place into the system.  3. The boss says “You can add my place on your ticket system” and Jesse logs into the system.  4. Jesse chooses a region and then adds this place to this region. | |

### Use case model



**Use Case 1:**

|  |
| --- |
| *Use case name:* Login |
| *Participant actors:* User, OR Admin, OR Client |
| *Flow of events:* 1. The user first enters to Online Ticket System system.    2. Online Ticket System presents the “Login Form” to the user.  3. The user enters his/her username into username text field on the screen, also enters his/her password into password text field on the screen. Lastly, the user sends a request to Online Ticket System by using login button on the screen to be logged in.  4. Online Ticket System checks the username ,password and user type from the “User” table so that Online Ticket System allows the user to login. |
| *Entry Condition:* The user enters to login screen. |
| *Exit Condition:* The user is logged in to Online Ticket System |
| *Quality Requirements:* The user cannot login cause of wrong username or password |

**Use Case 2:**

|  |
| --- |
| *Use case name:* UpdateProfile |
| *Participant actors:* Customer ,OR Admin,OR Orginizer |
| *Flow of events:* 1. The user sends a request to Online Ticket System to update his/her personal information which are his/her identification, and address, credit card Information.  2. Online Ticket System offers the “Personal Information Form” by using the “User” table. Then, displays it on the screen so that the user can update fields he/she wishes to.  3. The user edits the fields he/she wants, then he/she submits the form by using the update button to be updated.  4. Online Ticket System updates the information into the “User” table. Finally, it displays the “Acknowledgement Notice” to the user. |
| *Entry Condition:* The user is logged into Online Ticket System.  The user chooses Update Information option on the screen. |
| *Exit Condition:* Information of the user has been updated and the user has received an acknowledgement message. |
| *Quality Requirements* : The user has received an explanation indicating why form could not updated. |

**Use Case 3:**

|  |
| --- |
| *Use case name:* AddEvent |
| *Participant actors:* Client |
| *Flow of events:* 1. The Client logs in Online Ticket System.  2. Client enters “Add Event” button.  3. Client select place ,region, date, categories, seats and writes prices for categories. After that Client enters “Add” button.  4. Online Ticket System receives these and add this event to DB. Then, displays message that is “Your event is added” on the screen so that the user can add event. |
| *Entry Condition:* The Client is logged into Online Ticket System.  The Client chooses add event option on the Client Home Page. |
| *Exit Condition:* The Client has seen message on the screen. |
| *Quality Requirements:* The Client has received an explanation indicating why the event could not be added. |

**Use Case 4:**

|  |
| --- |
| *Use case name:* ShowEvents |
| *Participant actors:* User |
| *Flow of events:* 1. The User sends a request to Online Ticket System to see all events on the screen.  2. Online Ticket System displays all events on the screen. |
| *Entry Condition:* The User selects Show Events option on the screen. |
| *Exit Condition:* The User shows all of the available events on screen |
| *Quality Requirements:* The User has received an explanation indicating why the list could not be showed. |

**Use Case 5:**

|  |
| --- |
| *Use case name:* SelectCategory |
| *Participant actors:* User |
| *Flow of events:*  1.User logs into the Online Ticket System.  2.User select region on the event page.  3.User select the category filter for events.  4.And User can show events of selected category in selected region |
| *Entry Condition:* The user is logged into Online Ticket System.  The user chooses “Filter Events” button on the screen. |
| *Exit Condition:* The user has viewed selected category |
| *Quality Requirements:* The user has received an explanation indicating why the category could not be selected. |

**Use Case 6:**

|  |
| --- |
| *Use case name:* SelectTicketQuantity |
| *Participant actors:* User |
| *Flow of events:*  1.User logs into the Online Ticket System.  2.User select region on the event page.  3.User select the category filter for events.  4.And User selects “SelectTicketQuantity” button for the quantity of ticket |
| *Entry Condition:* The user is logged into Online Ticket System.  The user chooses “SelectTicketQuantity” button on the screen. |
| *Exit Condition:* The user has viewed Ticket Quantity OR, |
| *Quality Requirements:* The user has received an explanation indicating why the ticket quantity could not be selected. |

**Use Case 7:**

|  |
| --- |
| *Use case name:* ChooseSeat |
| *Participant actors:* User |
| *Flow of events:*  1.User logs into the Online Ticket System.  2.User select region on the event page.  3.User select the category filter for events.  4.And User selects “SelectTicketQuantity” button for the quantity of ticket  5.If there is available seat for selected ticket quantity, User can select his/her seat on system. |
| *Entry Condition:* The user is logged into Online Ticket System.  The user chooses “ChooseSeat” button on the screen. |
| *Exit Condition:* The user has viewed Seat Screen |
| *Quality Requirements:* The user has received an explanation indicating why the ticket seat could not be displayed. |

**Use Case 8:**

|  |
| --- |
| *Use case name:* SelectRegion |
| *Participant actors:* User |
| *Flow of events:* 1. The user opens the Online Ticket System web site.  2. User lists all regions  3.User selects any region what she/he wants. |
| *Entry Condition:* The user chooses “ListRegion” button. |
| *Exit Condition:* The user has seen all regions’ list and selected one of these regions |
| *Quality Requirements:* The user has received an explanation indicating why the regions could not be selected or listed. |

**Use Case 9:**

|  |
| --- |
| *Use case name:* BuyTicket |
| *Participant actors:* User |
| *Flow of events:* 1. User logs into the Online Ticket System.  2. User selects a region and then views the categories .  3.User select category and event what he/she wants  4.User want selects quantity and time period of event.  5. After user select the all attributes about this event,He/She clicks the “Buy Ticket” button.  5. If there is an available seat, The user can continue the payment page  6. Online Ticket System wants to enter his/her credit card information.  7.If Bank is approves the payment, Buy ticket session complete. |
| *Entry Condition:* The customer is logged into Online Ticket System.  The customer selects an event and then selects “Buy Ticket” button.. |
| *Exit Condition:* The customer sees Your ticket is added |
| *Quality Requirements:* The User has received an explanation indicating why the ticket could not be bought. System extends like timeout, NoAvailableSeat or NoLimit in credit card. |

**Use Case 10:**

|  |
| --- |
| *Use case name:* ChangePassword (Low Priority) |
| *Participant actors:* User, OR Client |
| *Flow of events:* 1. The member sends a request to Online Ticket System to change his/her current password.  2. Online Ticket System receives the request and it responses by presenting the “Password Change Form” to the user on the screen.  3. The user enters his/her current password into password text field on the screen. Also, enters new password into new password text field on the screen and the same new password into confirm new password text field to match the two so that the new password will be constructed unambiguously. Lastly, the user sends a request to Online Ticket System through the send option on the screen to change his/her current password.  4. Online Ticket System receives the request from the logged user and checks current password to be matched with the password which is stored in the “User” table. It also checks the new password and confirm new password to be matched each other. |
| *Entry Condition:* The user is logged into Online Ticket System.  The user chooses Change Password option on the screen. |
| *Exit Condition:* The user’s password has been changed, and the user has been received an acknowledgement message. OR, |
| *Quality Requirements:* The user has received an error message which informs the user that why the request could not be completed. |

**Use Case 11:**

|  |
| --- |
| *Use case name:* Register |
| *Participant actors:* User ,OR Client |
| *Flow of events:* 1. The visitor opens Online Ticket System web site.  2. The visitor clicks register button.  3. System offers some questions about member type.( User ,OR Client)  4.After that visitor writes the information about account  5.Visitor clicks the “Register” button and session completes |
| *Entry Condition:* The user opens Online Ticket System website. |
| *Exit Condition:* The Visitor sees a message that “You have registered into system” |
| *Quality Requirements:* The Visitor has received an explanation indicating why the registration could not be processed. |

**Use Case 12:**

|  |
| --- |
| *Use case name:* AddSeatsToEvent |
| *Participant actors:* Client |
| *Flow of events:* 1. Client logs into the Online Ticket System.  2. Client selects an event and adds seats to this event.  3. Client adds seats into this categories and events. |
| *Entry Condition:* The Client should be logged into Online Ticket System.  The Client chooses an event and categories. |
| *Exit Condition:* The Client sees a message that “Seats are added” OR, |
| *Quality Requirements:* The Client has received an explanation indicating why the seat could not be added. |

**Use Case 13:**

|  |
| --- |
| *Use case name:* AddPriceToEvent |
| *Participant actors:* Client |
| *Flow of events:* 1. Client logs into the Online Ticket System.  2. Client selects an event and adds categories to this event.  3. Client adds price into this event. |
| *Entry Condition:* The Client should be logged into Online Ticket System.  The Client chooses an event and categories. |
| *Exit Condition:* The Client sees a message that “Price is added” OR, |
| *Quality Requirements:* The Client has received an explanation indicating why the price could not be added. |

**Use Case 14:**

|  |
| --- |
| *Use case name: AddTimePeriodofEvent* |
| *Participant actors:* Client |
| *Flow of events:*  1. Client logs into the Online Ticket System.  2. Client select an event and clicks on “AddTimePeriod” button.  3.Client adds valid information about time period of event. |
| *Entry Condition:* The Client should be logged into Online Ticket System.  The Client selects “AddTimePeriod” button. |
| *Exit Condition:* The Client shows the event information page on screen,session is complete |
| *Quality Requirements:* The Client has received an explanation indicating why the time period could not be added. |

**Use Case 15:**

|  |
| --- |
| *Use case name:* AddPlaceToEvent |
| *Participant actors:* Client |
| *Flow of events:* 1. Client logs into the “Add Event” button.  2. Client select an event.  3. Client adds place to this event. |
| *Entry Condition:* The Client is logged into “Add Event” button.  The Client chooses an event on the screen. |
| *Exit Condition:* The Client sees a message that “Place is added” OR, |
| *Quality Requirements:* The Client has received an explanation indicating why the place could not be added. |

**Use Case 16:**

|  |
| --- |
| *Use case name:* AddRegionToEvent |
| *Participant actors:* Client |
| *Flow of events:* 1. Client logs into the “Add Event” button.  2. Client select an event.  3. Client adds region where event is done to this event. |
| *Entry Condition:* The Client is logged into “Add Event” button.  The Client chooses an event on the screen. |
| *Exit Condition:* The Client sees a message that “Region is added” OR, |
| *Quality Requirements:* The Client has received an explanation indicating why the region could not be added. |

**Use Case 17:**

|  |
| --- |
| *Use case name:* AddTypeOfEvent |
| *Participant actors:* Admin |
| *Flow of events:*  1. Admin logs into the “Add Event” button.  2. Admin clicks AddTypeOfEvent Button and then the admin adds new types into the system.  3. Online Ticket System receives this request and displays form that new type is added. |
| *Entry Condition:* The Admin should be logged into Online Ticket System.  The Admin clicks AddTypeOfEvent Button. |
| *Exit Condition:* The admin sees a message that “Event is added” |
| *Quality Requirements:* The admin has received an explanation indicating why the eventType could not be added |

**Use Case 18:**

|  |
| --- |
| *Use case name:* AddPlace |
| *Participant actors:* Admin |
| *Flow of events:*  1. Admin logs into the Online Ticket System.  2. Admin selects “AddPlace” button and adds a place into the system.  3. Online Ticket System receives this request and displays form that new place is added. |
| *Entry Condition:* The Admin should be logged into Online Ticket System.  The Admin clicks “AddPlace” button. |
| *Exit Condition:* The admin sees a message that “Place is added” |
| *Quality Requirements:* The admin has received an explanation indicating why the place could not be added. |

**Use Case 19:**

|  |
| --- |
| *Use case name:* UpdatePlace |
| *Participant actors:* Admin |
| *Flow of events:*  1. Admin logs into the Online Ticket System.  2. Admin selects a place and then updates this places’s information. |
| *Entry Condition:* The Admin should be logged into Online Ticket System.  The Admin clicks “UpdatePlace” button. |
| *Exit Condition:* The admin sees a message that “Place is updated” |
| *Quality Requirements:* The admin has received an explanation indicating why the place could not be updated. |

**Use Case 20:**

|  |
| --- |
| *Use case name:* AddRegion |
| *Participant actors:* Admin |
| *Flow of events:*  1. Admin logs into the Online Ticket System.  2. Admin clicks “AddRegion” button and adds a region into the system.  3. Online Ticket System receives this request and displays form that new region is added. |
| *Entry Condition:* The Admin should be logged into Online Ticket System.  The Admin clicks “AddRegion” button. |
| *Exit Condition:* The admin sees a message that “Region is added” OR, |
| *Quality Requirements:* The admin has received an explanation indicating why the region could not be added. |

**Use Case 21:**

|  |
| --- |
| *Use case name:* RemoveEvent |
| *Participant actors:* Admin |
| *Flow of events:*  1. Admin logs into the Online Ticket System.  2. Admin select event which is he/she want to remove .  3.Admin clicks on “RemoveEvent” button |
| *Entry Condition:* The Admin should be logged into Online Ticket System.  Admin clicks “RemoveEvent” button. |
| *Exit Condition:* The admin sees a message that “Event is removed” OR, |
| *Quality Requirements:* The admin has received an explanation indicating why the event could not remove. |

**Use Case 22:**

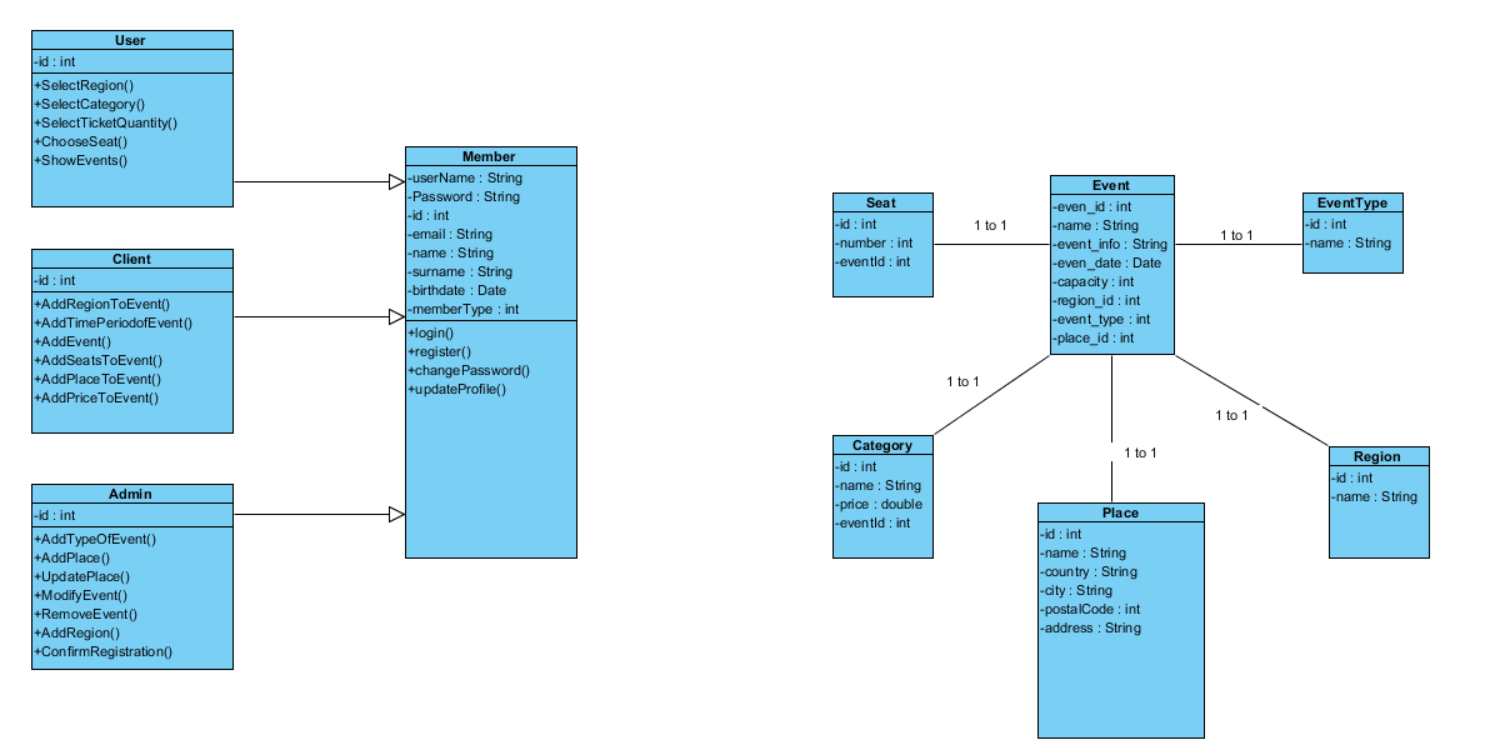
|  |
| --- |
| *Use case name:* ModifyEvent |
| *Participant actors:* Admin |
| *Flow of events:*  1. Admin logs into the Online Ticket System.  2. Admin select event which is he/she want to modify.  3.Admin clicks on “ModifyEvent” button |
| *Entry Condition:* The Admin should be logged into Online Ticket System.  Admin clicks “ModifyEvent” button. |
| *Exit Condition:* The admin sees a message that “Event is modified” OR, |
| *Quality Requirements:* The admin has received an explanation indicating why the event could not modified. |

**Use Case 23:**

|  |
| --- |
| *Use case name:* ConfirmRegistration |
| *Participant actors:* Admin |
| *Flow of events:*  1. Admin logs into the Online Ticket System.  2. Admin clicks on “Registration” page.  3.Admin shows the registration requests on the screen  4.Admin can approve or disapprove the requests  5.Admin clicks on “Registration update” button and updates registration status of accounts. |
| *Entry Condition:* The Admin should be logged into Online Ticket System.  Admin clicks “Registration” page. |
| *Exit Condition:* The admin sees a message that “Registration status updated” OR, |
| *Quality Requirements:* The admin has received an explanation indicating why the “ConfirmRegistration” could not be updated. |

### Object model

The analysis object model, depicted with UML class diagrams, includes classes, attributes, and operations. The analysis object model is a visual dictionary of the main concepts visible to the user.



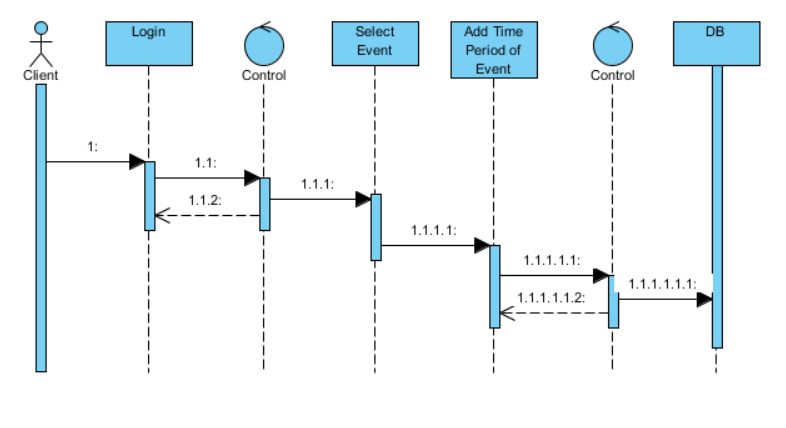
### Dynamic model

The dynamic model is depicted with sequence diagrams and with state machines. Sequence diagrams represent the interactions among a set of objects during a single use case. State machines represent the behavior of a single object (or a group of very tightly coupled objects). The dynamic model serves to assign responsibilities to individual classes and, in the process, to identify new classes, associations, and attributes to be added to the analysis object model.

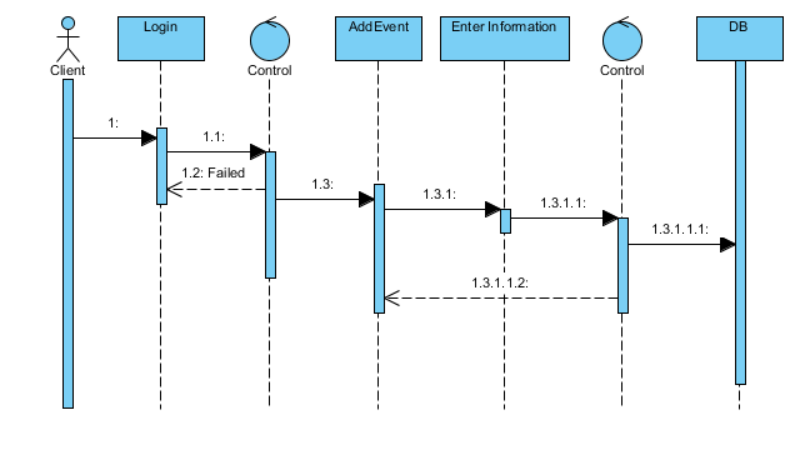
When working with either the analysis object model or the dynamic model, it is essential to remember that these models **represent user-level concepts, not actual software classes or components.**

**SEQUENCE DIAGRAMS**

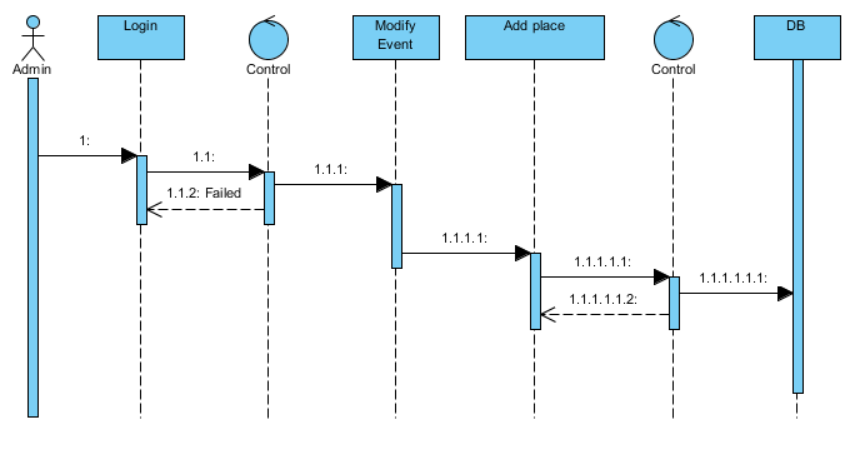
**1-) AddTimePeriodofEvent**



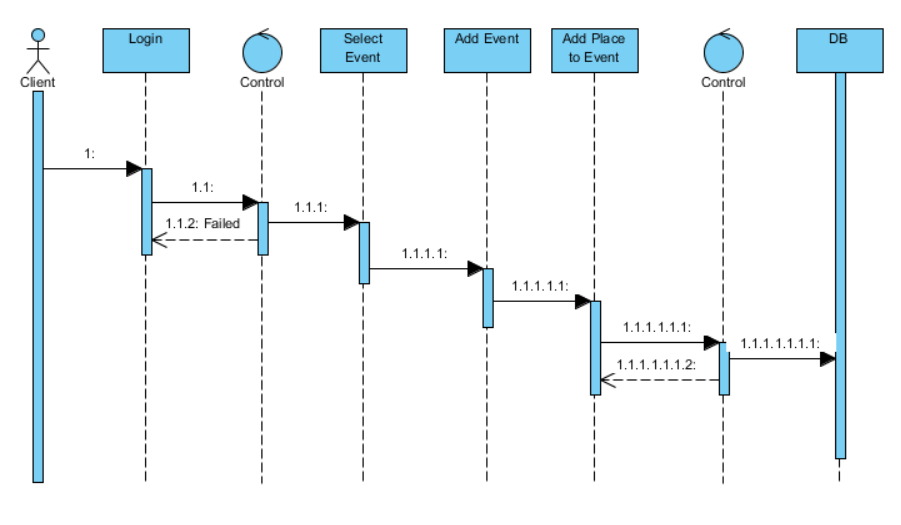
**2-) AddEvent**



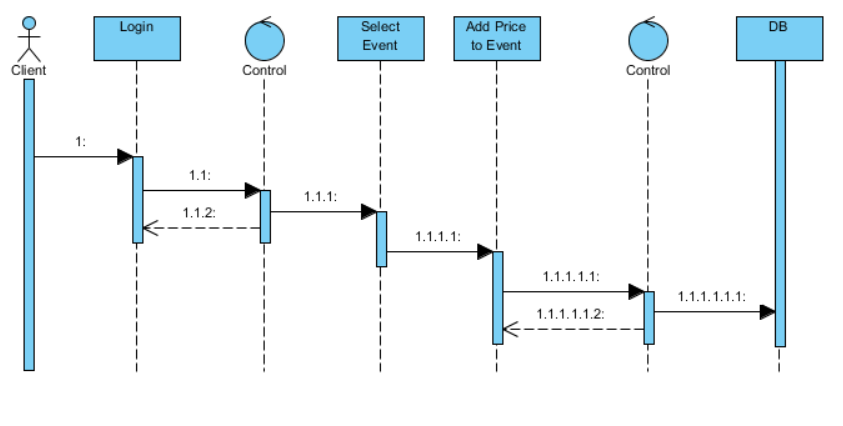
**3-) AddPlace**



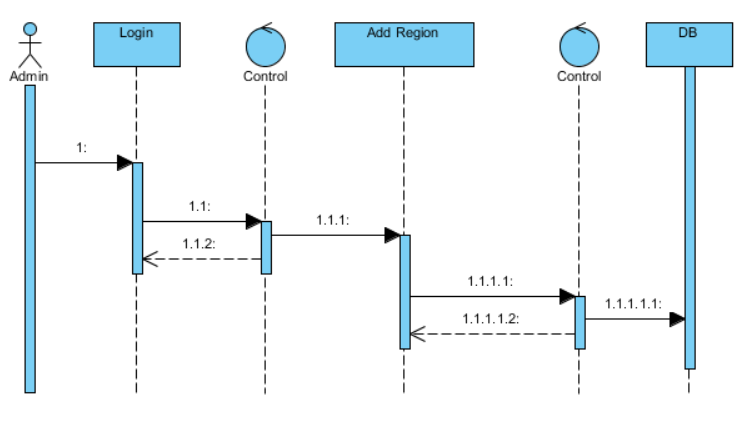
**4-) AddPlacetoEvent**



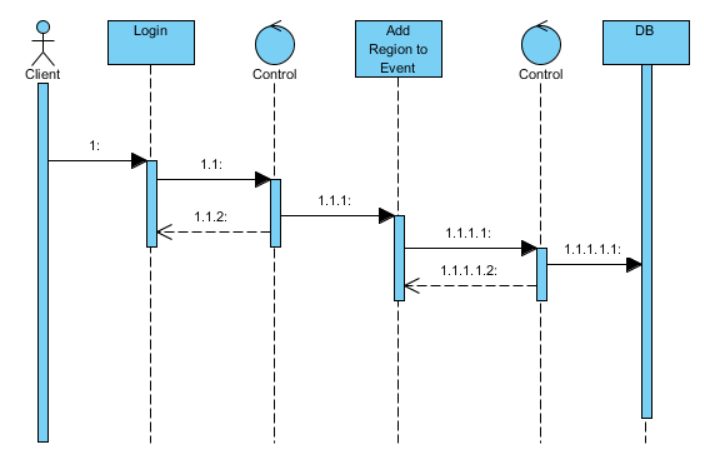
**5-)AddPriceToEvent**



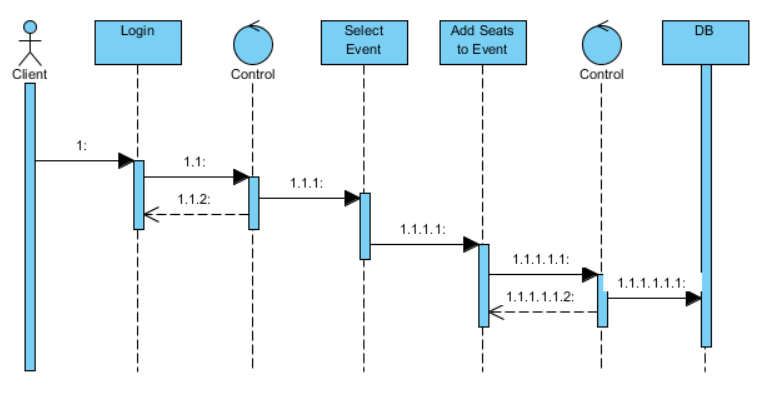
**6-)AddRegion**



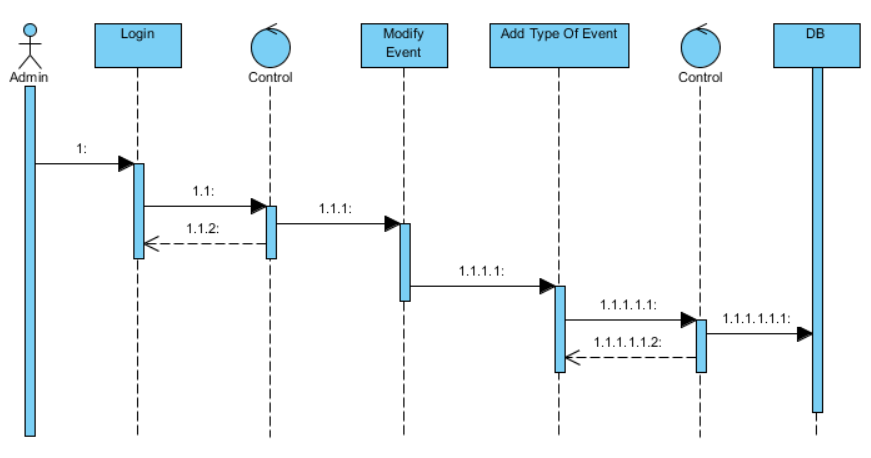
**7-) AddRegionToEvent**



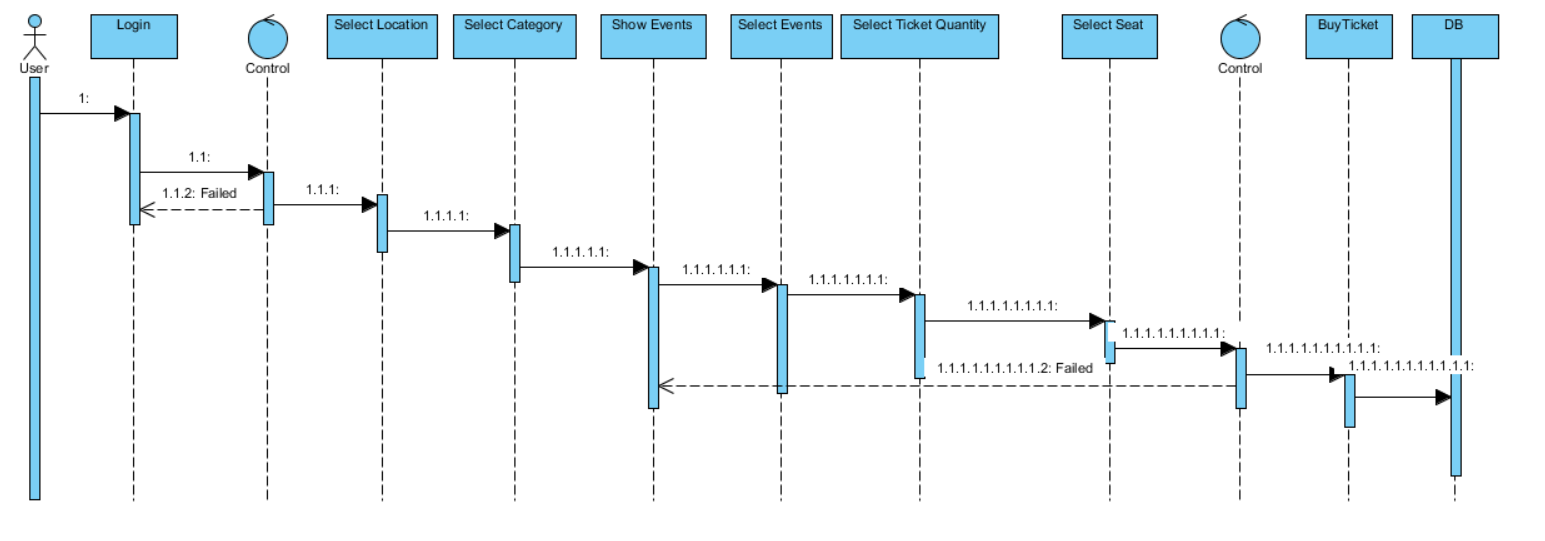
**8-)AddSeatsToEvent**



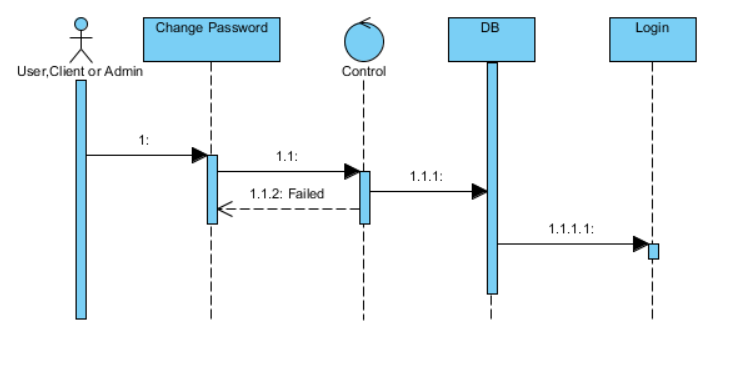
**9-)AddTypeOfEvent**



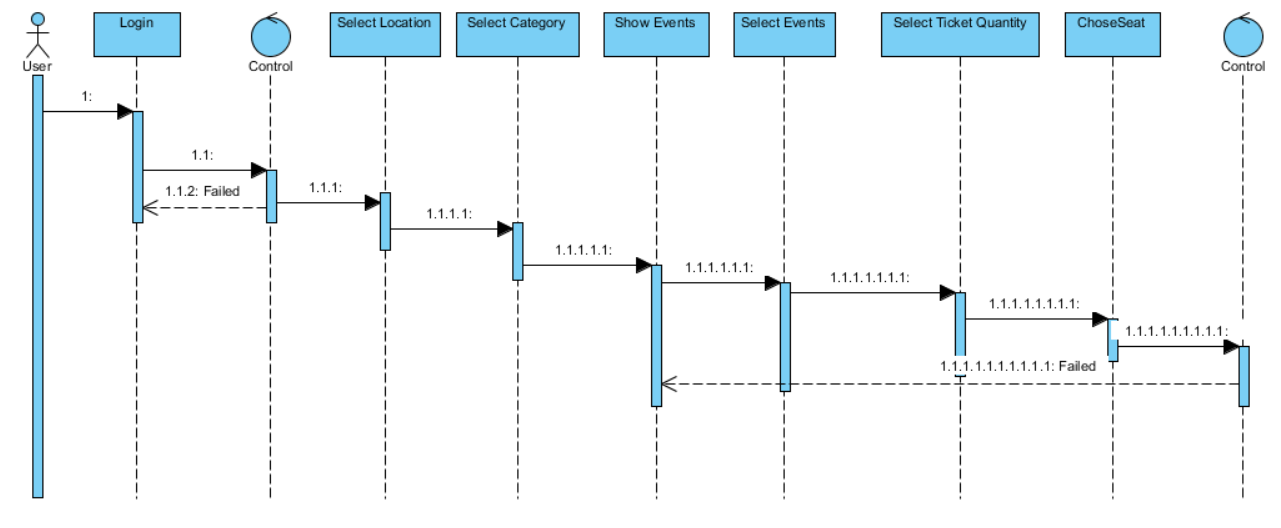
**10-) BuyTicket**



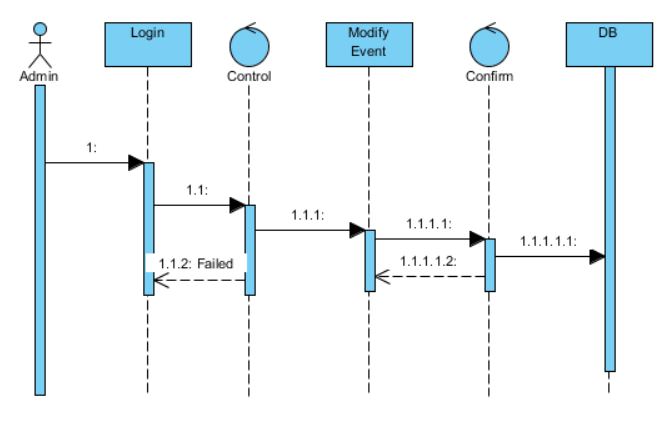
**11-) ChangePassword**



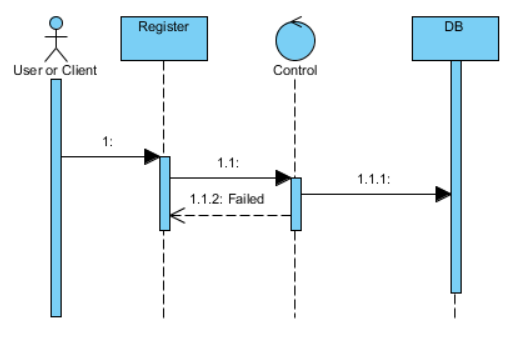
**12-) ChooseSeat**



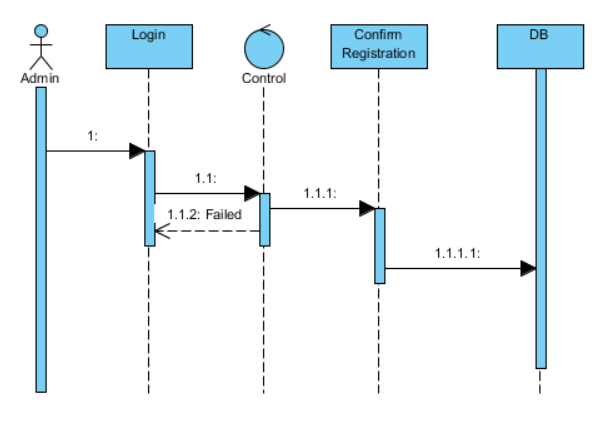
**13-) ModifyEvent**



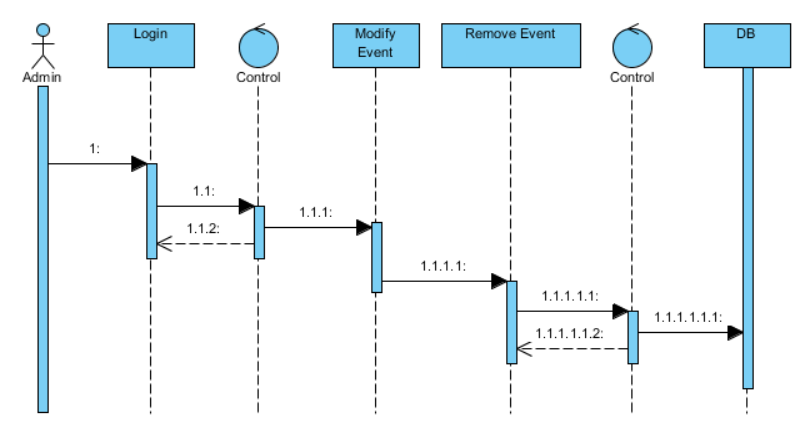
**14-)Register**



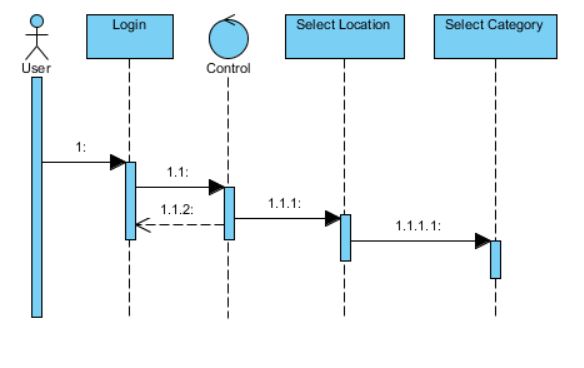
**15-) ConfirmRegistration**



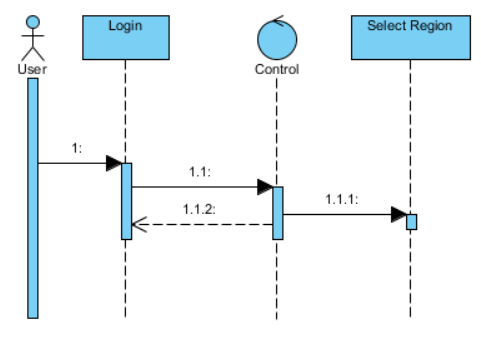
**16-) RemoveEvent**



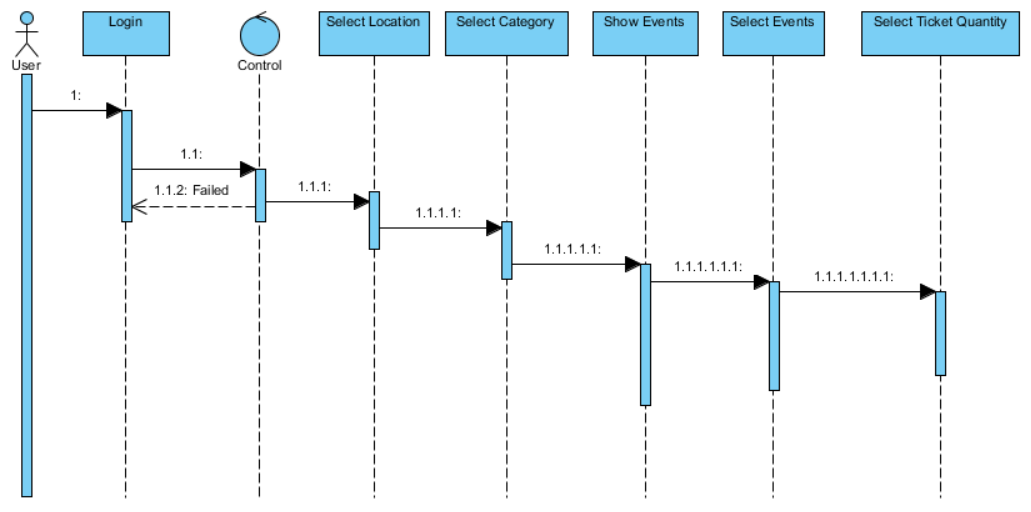
**17-)SelectCategory**



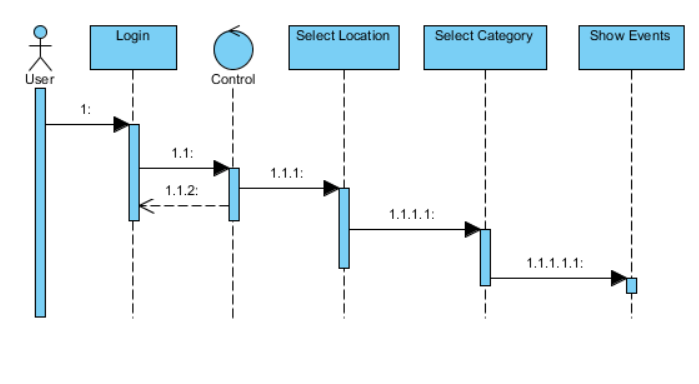
**18-) SelectRegion**



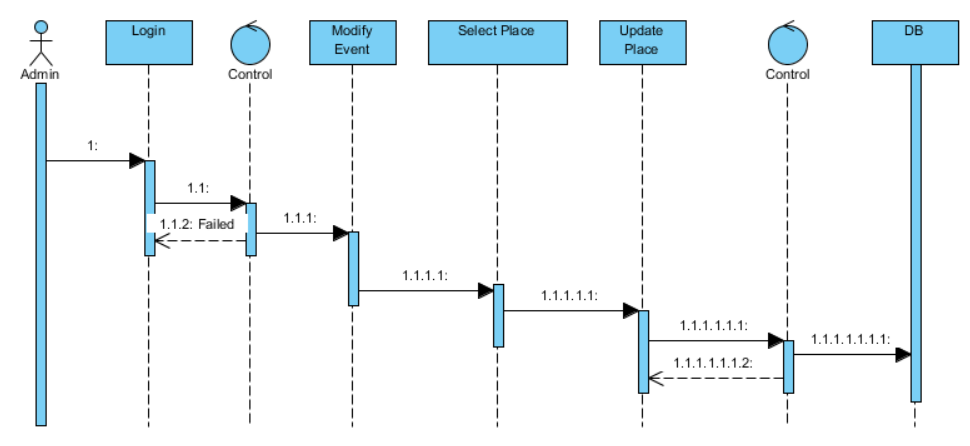
**19-)SelectTicketQuantity**



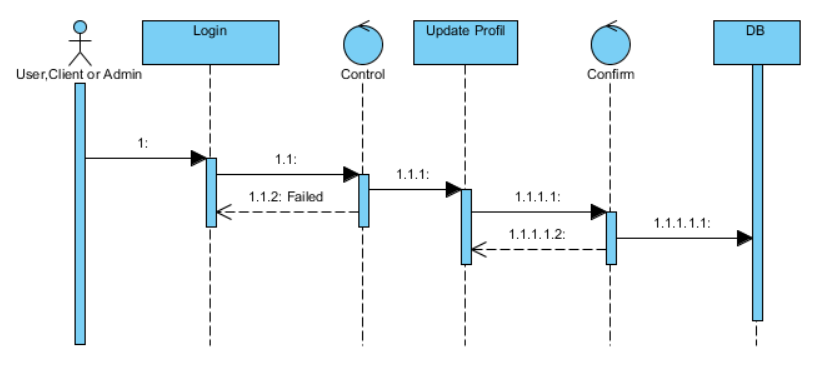
**20-)ShowEvents**



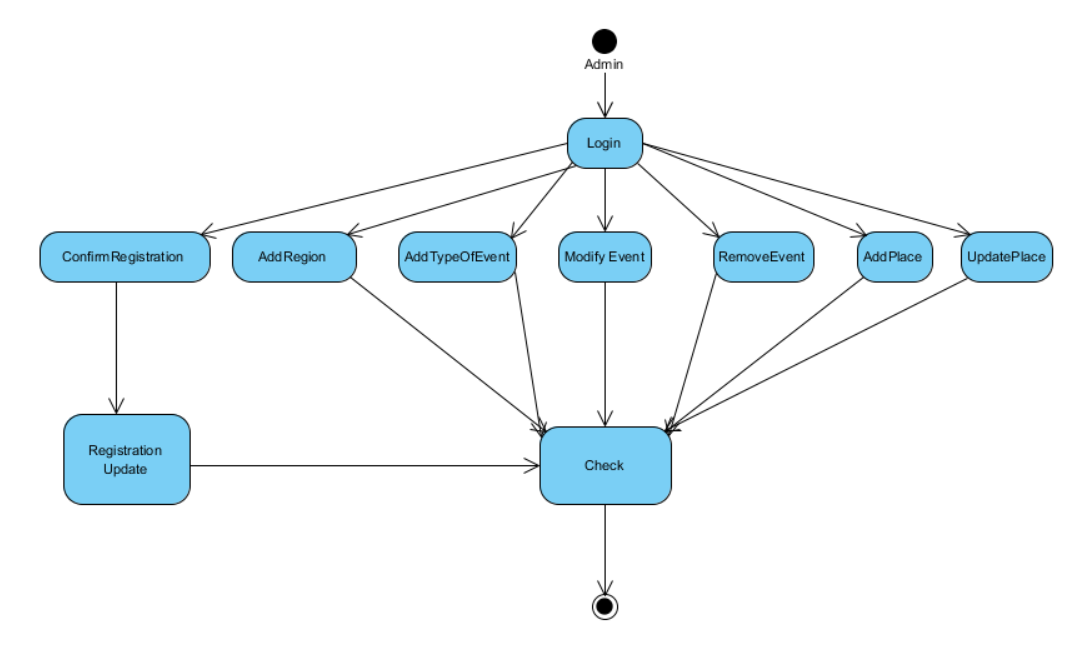
**21-) UpdatePlace**



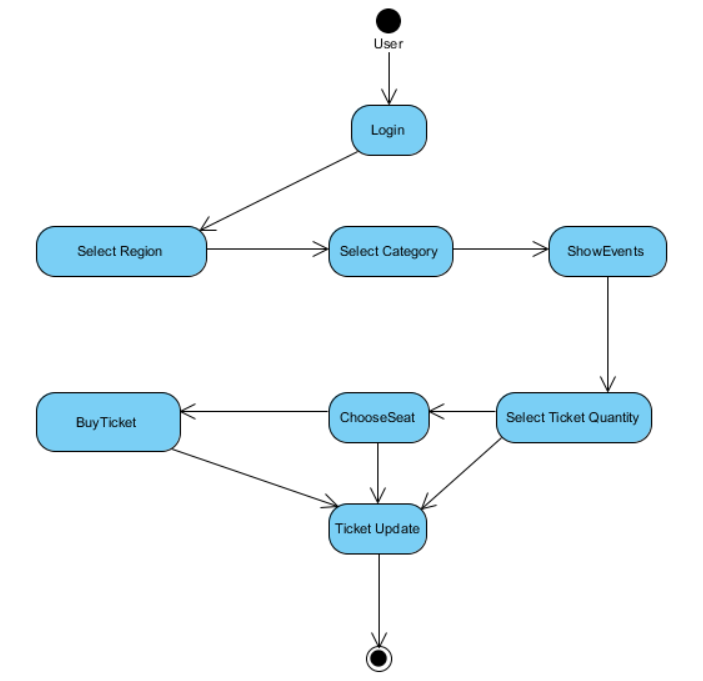
**22-) UpdateProfile**



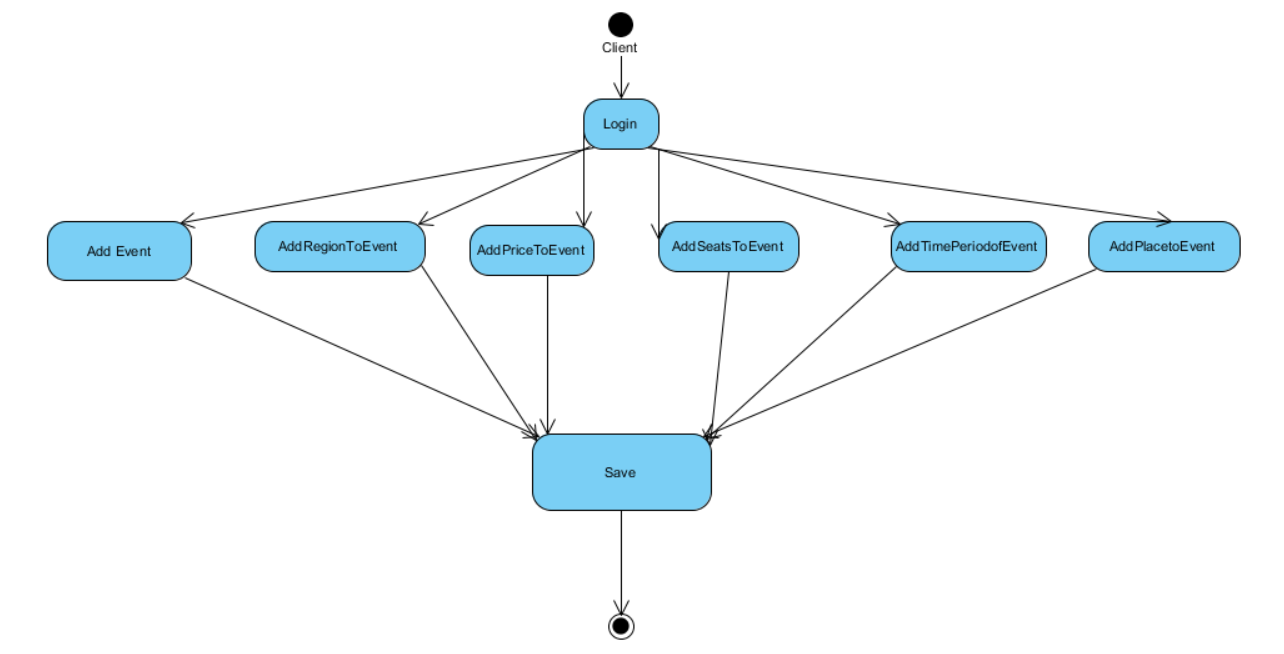
**Admin Activity Diagram**



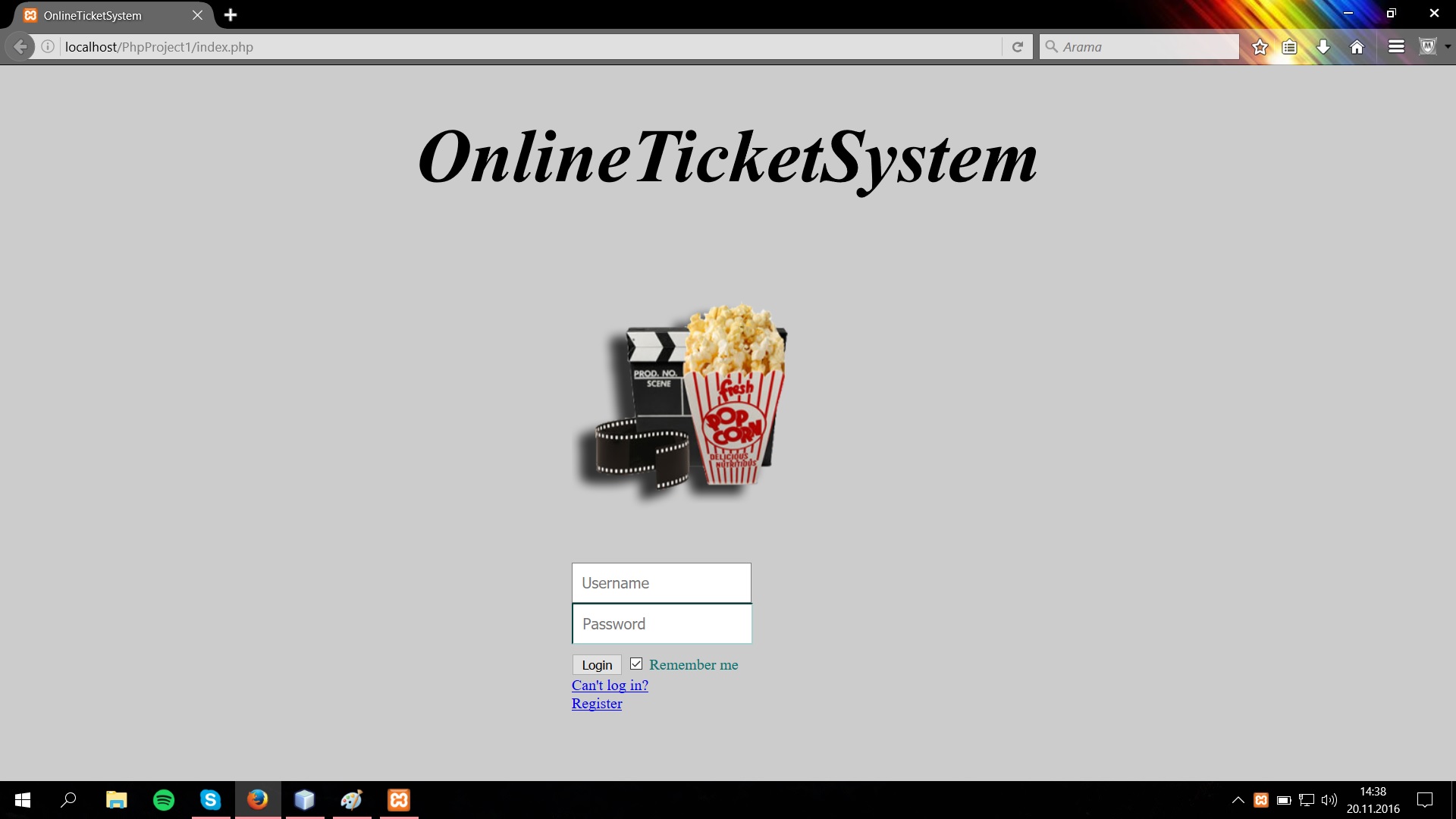
**User Activity Diagram**



**Client Activity Diagram**



**UserInterface**



# Glossary

*Member table:*Contains username, password, id, e-mail, name, surname and birthdate, member type.

*Region table:* Contains the region id and name.

*EventType table:* Contains the type id and name.

*Event:* Contains the event id, event name, informations, event date, capacity for event,region id, event type id and place id.

*Seat table:* Contains seat id, seat number and event id.

*Place table*: Contains address id, address name, user id, country, city, postalCode.

*Category table:* Contains category id, category name, price for event and event id.

*User Table*: User id

*Client Table*: Client id

*Admin Table*: Admin id

*Login Form:* Contains two text fields, one for username and one for password. The form also contains a login button to provide, the functionality of submitting the form, to the user.

*Profile Form:* Contains the user’s personal information and text fields which are used to display the personal information of the user and are also used to edit/update some personal information. The form also contains a button named “update” so that the user can submit the form.

*Password Change Form:* Contains three text fields namely “password”, “new password”, “confirm new password” and a button named “send” so that the user can submit the form.

# References

[1]http://www.cs.fsu.edu/~lacher/courses/COP3331/rad.html