###### Logo Black and White

PUCIT

Punjab University College of Information Technology

**Second Deliverable OF Smart Marital Contractor System (SMCS)**

**Version 1.0**

**Supervisor**

**Ma’am Amina Mirza**

Table of Contents

[1 Introduction 3](#_Toc148483291)

[1.1 Usecase descriptions 3](#_Toc148483292)

[1.2 Use Case Diagram (Refined) 4](#_Toc148483293)

[1.3 Domain Model 4](#_Toc148483293)

[1.4 Sequence Diagram 6](#_Toc148483295)

[1.4.1. Defining a Sequence diagram 6](#_Toc148483296)

[1.4.2. Basic Sequence Diagram Symbols and Notations 6](#_Toc148483297)

[1.4.3 Example 10](#_Toc148483298)

[1.4.4 Distributing Control Flow in Sequence Diagrams 10](#_Toc148483299)

[1.5 Collaboration Diagram 13](#_Toc148483300)

[1.5.1 Contents of Collaboration Diagrams 14](#_Toc148483301)

[1.5.2 Constructs of Collaboration Diagram: 14](#_Toc148483302)

[1.6 Operation Contracts 15](#_Toc148483303)

[1.7 Design Class Diagram 16](#_Toc148483304)

[1.7.1 Create Initial Design Classes 16](#_Toc148483305)

[1.7.2 Designing Boundary Classes 17](#_Toc148483306)

[1.7.3 Designing Entity Classes 17](#_Toc148483307)

[1.7.4 Designing Control Classes 17](#_Toc148483308)

[1.7.5 Identify Persistent Classes 18](#_Toc148483309)

[1.7.6 Define Class Visibility 19](#_Toc148483310)

[1.7.11 Design Class Relationships 23](#_Toc148483311)

# 

# 

## Group Members

|  |  |  |
| --- | --- | --- |
| **Name** | **Roll No** | **E-Mail** |
| Ali Ijaz | Mcsf19a040 | mcsf19a040@pucit.edu.pk |

## 1. Introduction

This project is to facilitate people with online booking of marriage halls.

The project is divided into two side Admin side (where hall owner can do Sign Up, Sign In, handle Inventory, handle booking request, handle sales).

Customer Side (View hall, Search hall, Booking Request, Select Package)

Second deliverable is all about the use-case modeling and software design. In the previous deliverable, analysis of the system is completed. So, I understand the current situation of the problem domain. Now I am ready to strive for a solution for the problem domain by using object-oriented approach. Following artifacts must be included in this deliverable.

1. Use case description

2. Use case diagram refined

3. Domain Model

4. Sequence Diagram

5. Collaboration Diagram

6. Operation Contracts

7. Design Class Diagram

8. Data Model

Now we discuss these artifacts one by one as follows:

## 1.1 Use case Description

While technically not part of UML, use case documents are closely related to UML use cases. A use case document is text that captures the detailed functionality of a use case. Description of all use case‘s is written down. Use case description typically contains the following parts:

#### Brief description

**Organizer Use Cases**

**Sign up:**

The Organizer will sign up on the website first.

**Login:**

Once he signed up, he will be able to login by entering his E-mail and password.

**Hall:**

Only the organizer can add, edit, update or delete the hall.

**Packages:**

The organizer will add package name, package type and person charges according to package type.

**Facilities:**

The organizer will add, update and delete the available facilities during to the function.

**Timing:**

The organizer will add, update and delete the available time slots of hall.

**Appointments:**

After seeing the customer’s request, the organizer will confirm with the customer and approve the request. He will then meet with the customer to agree on the terms and conditions.

**Order:**

Order contain record of advance payment and remaining amount and organizer may change order requirements and can discount on existing order.

**Transaction:**

The organizer will perform the transaction of remaining amount from the customer.

**Customer Use Cases**

**View Halls:**

The customer can view different types of hall.

**Search Halls:**

The customer can search the nearest hall.

**Send booking Request:**

The customer will select the hall according to his area and budget. He will fill out a form in which his name, phone number and email address is required.

**Customer review:**

Apart from the permanent user, if someone visits the website, he can also give an absolute review of the website.

#### Preconditions:

**Organizer use cases**

|  |  |
| --- | --- |
| Use Cases | Preconditions |
| Sign up | Email address |
| Login | Must signed up first |

**Customer use cases**

|  |  |
| --- | --- |
| **Use Cases** | **Preconditions** |
| **Booking Request** | **Name, phone number and email is required.** |

#### Basic flow

First the **user** will visit the website, then he will see the detail and then he will send a request to book the hall to the organizer. After seeing the customer’s request, the organizer will confirm with the customer and approve the request. He will then meet with the customer to agree on the terms and conditions. The organizer will perform the transaction of remaining amount from the customer.

#### Alternate flows

**Organizer use cases**

|  |  |
| --- | --- |
| **Use Case** | **Alternate flows** |
| **Sign up** | **Error message invalid E-mail** |
| **Login** | **Error message invalid user** |

**Customer use cases**

|  |  |
| --- | --- |
| **Use Case** | **Alternate flows** |
| **Booking Request** | **Error message missing field** |

#### Post conditions

**Organizer use cases**

|  |  |
| --- | --- |
| **Use cases** | **Post conditions** |
| **Sign up** | **Use valid E-mail format** |
| **Login** | **E-mail and password must match** |

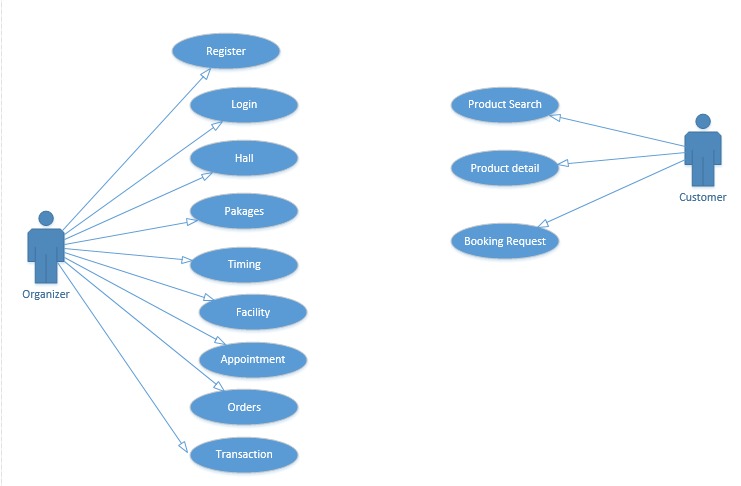
**Customer post conditions**

There is no post conditions for customer.

## 1.2 Use case Diagram (refined and updated)

Analysis level use case diagram is a refined High level use case diagram and is actually the explanation of high level use case diagram. In this diagram high level use cases are expanded in a way that exhibit how high level use cases will reach to their functionality. Two types of relationships are used in this diagram. Which are:

* Extend
* Include

****

## 1.3 Domain Model

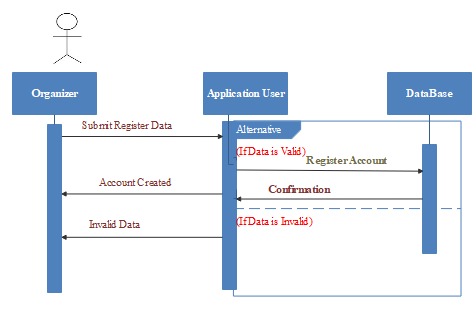
The following domain model is a conceptual model of the domain that incorporates both behavior and data of our project. A domain model is a formal representation of a knowledge domain to understand the project concepts, roles, datatypes, individuals, and rules and the logic.

## Domain Model

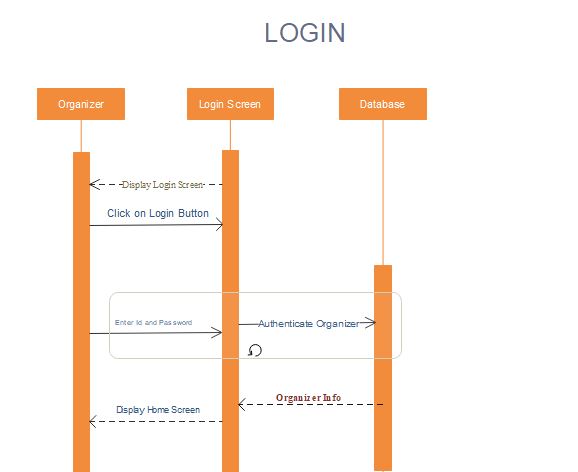
## 1.4 Sequence Diagram

A Sequence diagram depicts the sequence of actions that occur in a system. The invocation of methods in each object, and the order in which the invocation occurs is captured in a Sequence diagram. This makes the Sequence diagram a very useful tool to easily represent the dynamic behavior of a system.

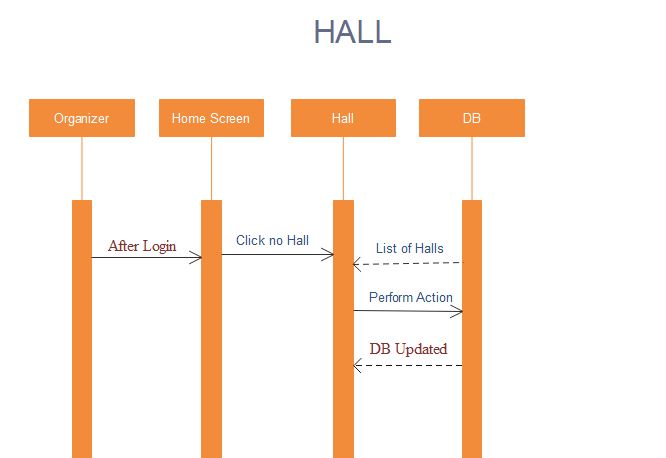
**Organizer Register:**

****

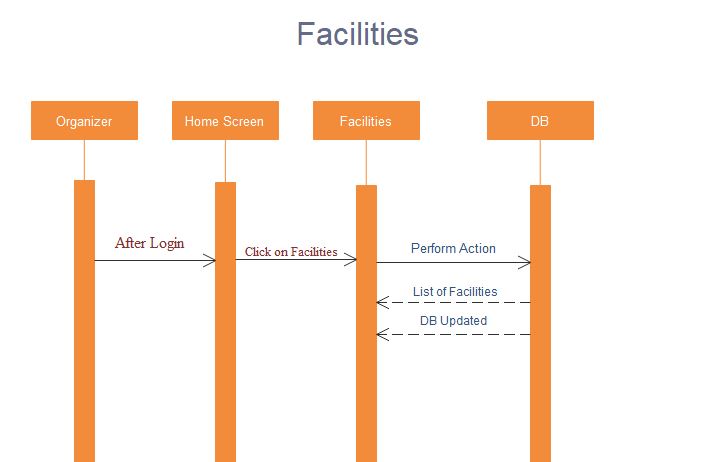
**Organizer Login:**

****

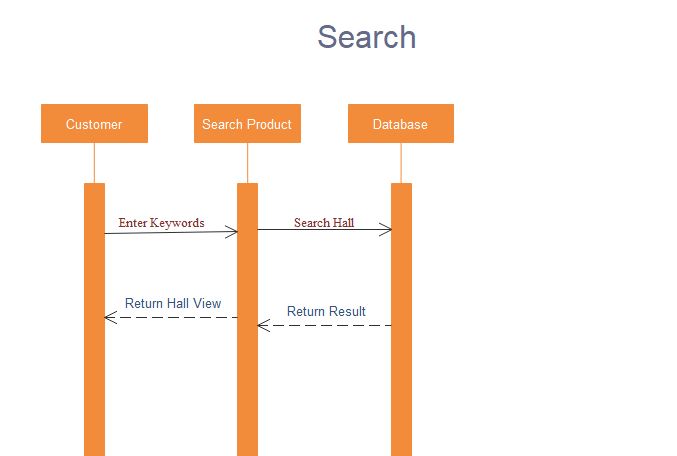
**Hall Update:**

****

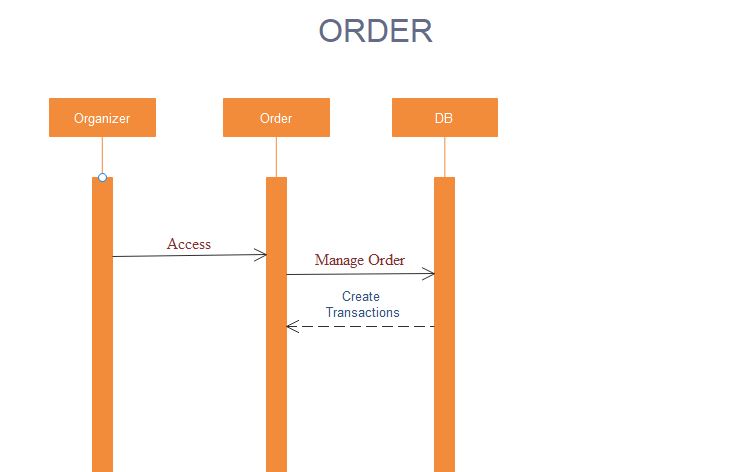
**Add Facilities:**

****

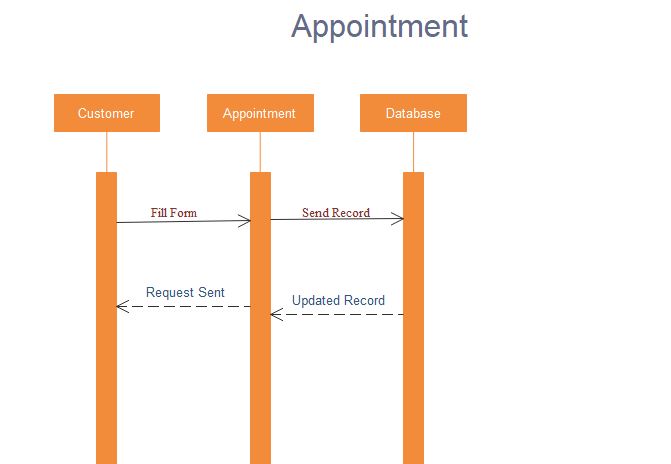
**Customer Search hall:**

****

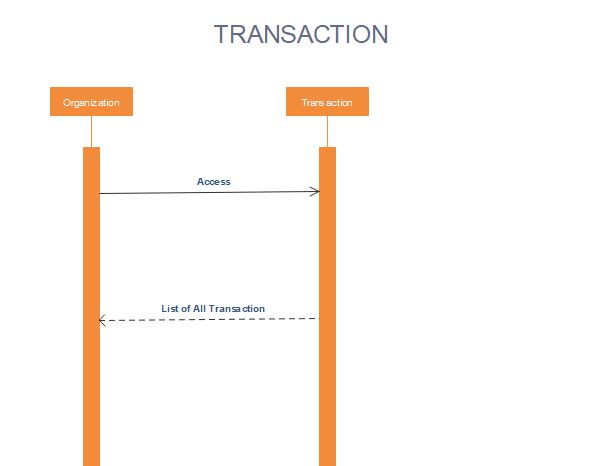
**Order Hall:**

****

**Appointment:**

****

**Transaction:**

****

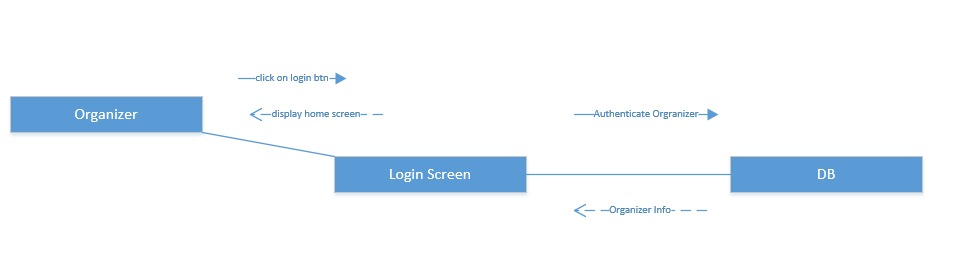
## 1.5 Collaboration Diagram

A collaboration diagram describes a pattern of interaction among objects; it shows the objects participating in the interaction by their links to each other and the messages that they send to each other. Collaboration diagrams known as Communication Diagram that are used to show how objects interact to perform the behavior of a particular use case, or a part of a use case. Along with sequence diagrams, collaboration is used by designers to define and clarify the roles of the objects that perform a particular flow of events of a use case. They are the primary source of information used to determining class responsibilities and interfaces

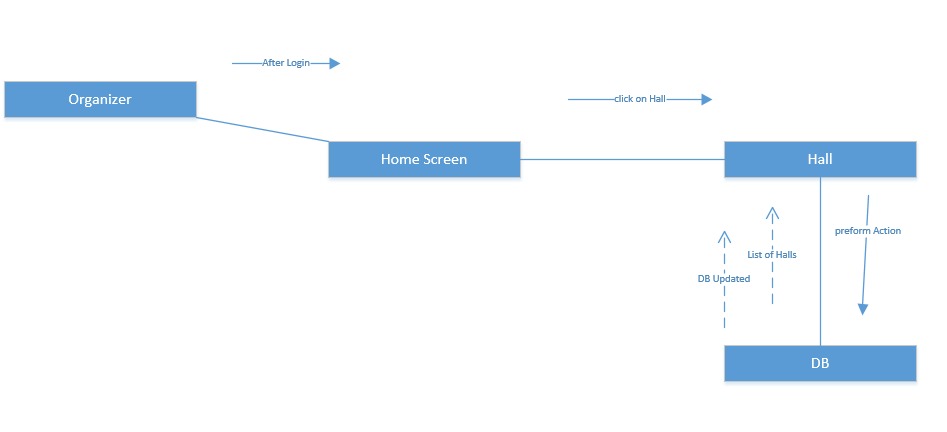
**Organizer Registration:**

## Registration

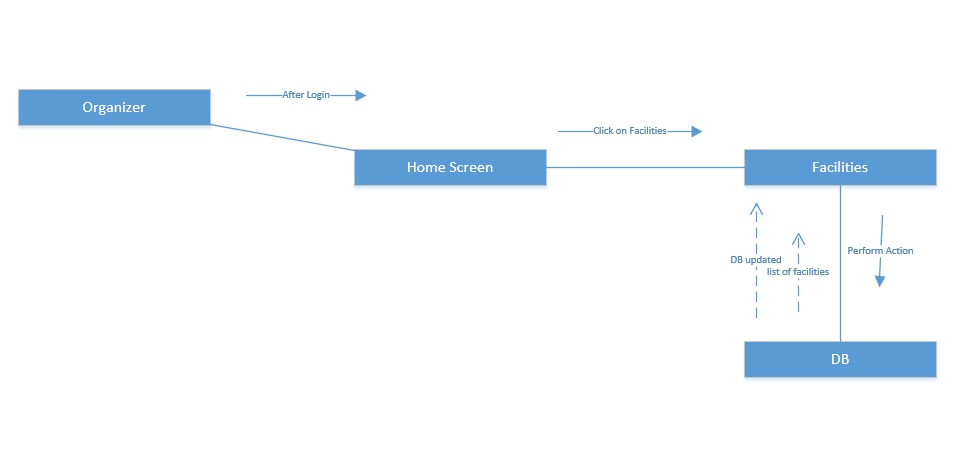
**Organizer Login:**

****

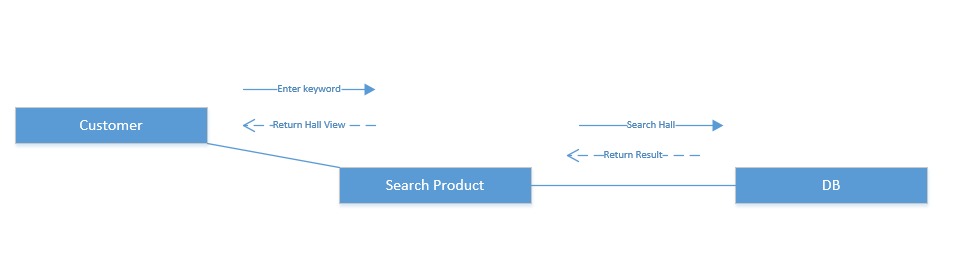
**Update Hall:**

****

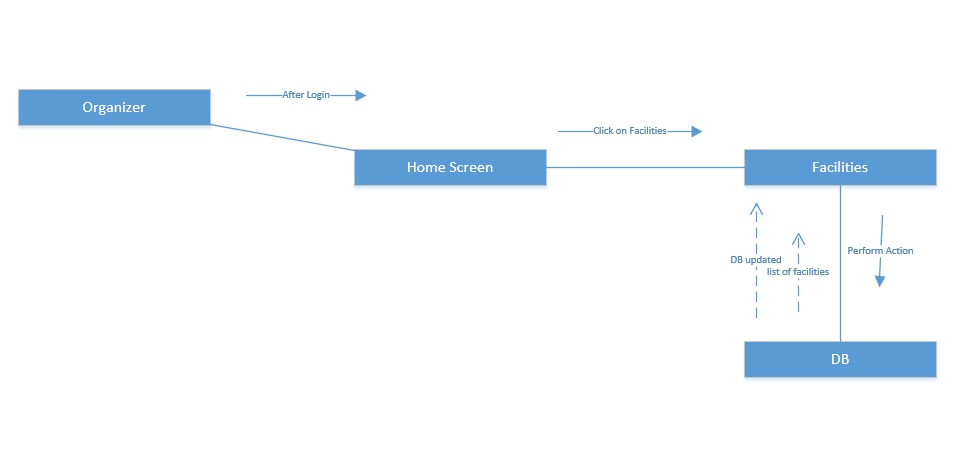
**Add Facilities:**

****

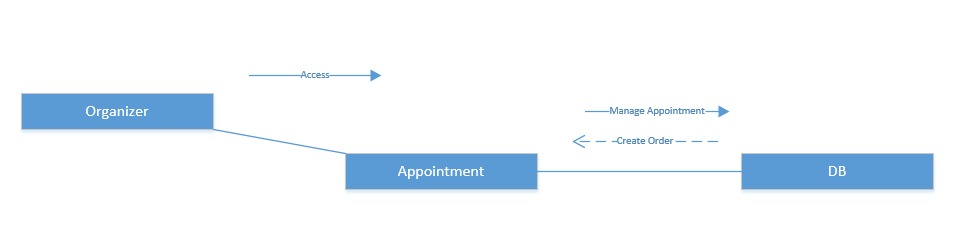
**Customer search hall:**

****

**Booking Request:**

****

**Appointment:**

****

## Transaction:

