## **STATE UNIVERSITRY OF ZANZIBAR**

## **SCHOOL OF BUSSINESS**



**STUDENT NAME: SHARIFA KHAMIS FAKI**

**REGISTATION NUMBER: BITA/4/20/004/****TZ**

**COURSE CODE AND NAME: WT 822 ADVANCED WEB**

**LECTURER NAME: MASOUD .H. MMANGA**

TABLE OF CONCTENTS

CHAPTER ONE

[**1.** **Introduction** 3](#_Toc102708265)

[**2.** **Relational** 3](#_Toc102708266)

[**3.** **General objectives** 3](#_Toc102708267)

[**i.** **Main Objective** 3](#_Toc102708268)

[a) Specific Objectives 3](#_Toc102708269)

[**ii.** **Challenge of current existing system** 4](#_Toc102708270)

[**iii.** **Solution to current business operations** 4](#_Toc102708271)

[**CHAPTER TWO:** 5](#_Toc102708272)

[**4.** **Deliverables** 5](#_Toc102708273)

[**CHAPTER THREE** 6](#_Toc102708274)

[**5.** **Functional Requirements** 6](#_Toc102708275)

[**6.** **Security Requirements** 6](#_Toc102708276)

[**CHAPTER FOUR** 7](#_Toc102708277)

[**7.** **Agile for software development life cycle** 7](#_Toc102708278)

[**iv.** **Advantages** 7](#_Toc102708279)

[**v.** **Disadvantages** 7](#_Toc102708280)

[**8.** **System architecture** 7](#_Toc102708281)

[**9.** **Conceptual design of the system** 8](#_Toc102708282)

[**10.** **User Interface** 8](#_Toc102708283)

[**11.** **Database Design** 8](#_Toc102708284)

[a) **Conceptual database design** 8](#_Toc102708285)

[**12.** **Description of Relationships (Data Dictionary)** 11](#_Toc102708286)

[**13.** **Development technology (languages).** 16](#_Toc102708288)

# 

# **CHAPTER ONE:**

## **Introduction**

Online Ticket Booking System (OTBS) refers to a system that is able to process tickets for all the events in Airplane Zanzibar. Airplane Zanzibar has been losing many customers due to a poor and booking system for tickets.

The citizens and foreign nationals have to look for agents and pay cash in specific areas around the city to purchase the tickets before any event which results to long queues and time wastages.

## **Relational**

The reason leading to create this project are:

• The many people in Zanzibar and other foreign have to the long queues and overcrowding of the theatre entrances by clients who come to purchase tickets to attend events in Airplane.

• It taken great time to find a ticket for transport which peoples have need, because some people haven obtain challenge on the get ticket

## **General objectives**

Due to the Long queues and overcrowding of the theatre entrances by clients who come to purchase tickets to attend events in Airplane, which leads to some guest missing to attend the available event in time. There for the researcher will come up with a system that will enable ticket booking easy.

### **Main Objective**

The main objective is to develop a secure online ticket booking system to improve and enable booking of tickets easier to avoid long queues.

### Specific Objectives

To identify requirements for the Online Ticket Booking System

To design the system that will make online booking of tickets easier and fast.

To test, validate and implement the system.

### **Challenge of current existing system**

Zanzibar has its own ticketing system that is done manually by the authorized agents who are poorly distributed all over the city and in some area’s they are not able to cater for the long queues of guest.

For those who wish to book in advance in order to attend a major entertainment event in the Zanzibar, one has to go to the official booking offices, stand in line and then pay in order to obtain a ticket.

The tickets have to be printed which usually results in potential theft, loss, forgery, illegal selling of tickets and poor record management of tickets sold. This always leads to low revenue income to event sponsors.

### **Solution to current business operations**

The Zanzibar faces a problem of tickets being expensive to manufacture and distribute.

The Researcher has decided to come up with an online ticket booking system that will reduce the long queues and time wastage which discourages the guests to attend certain events that take place in the theatre.

# 

# **CHAPTER TWO:**

## **Deliverables**

* Online Ticket Booking Systems:

Online ticket booking system is an information system that uses Internet web technologies to deliver information and services, to users or other information systems/ applications. It is a software system whose main purpose is to publish and maintain data by using hypertext-based principles.

* Online booking/Reservation:

Electronic ticket or E-ticket is an example of such a class of e-service-tickets give evidence to their holders to have permission enter a place of entertainment, use a means of transportation, or have access to some internet services. E-tickets give evidence to their holders to have permission to enter a place of entertainment, use means of transportation or have access to some internet services.

# **CHAPTER THREE**

## **Functional Requirements**

These are requirements necessary for the system to function efficiently and effectively and they include the following:

|  |  |
| --- | --- |
| No | Functional Requirements |
| 1 | The system should enable customers to register as members before they order event's ticket. |
| 2 | The online Ticket booking System should also enable customers to view ticket availability, ticket cost, events schedules and after booking successfully, an email can be sent to them. |
| 3 | The system should show venue, the start and end times of events, and ticket number. |
| 4 | The system should allow the administrator to upload new information so that the member can check the availability of events. |
| 5 | The system should also allow the agent to view tickets booked by a member. |
| 6 | The system should be able to store the records for all members who have booked tickets. |
| 7 | Booking confirmation should be sent to user to the specified contact details |
| 8 | System should consider time zone synchronization when accepting bookings from different time zones |
| 9 | Search results should enable users to find the most recent and relevant booking options |

|  |  |
| --- | --- |
| No | Non-functional requirement |
| 1 | System should visually confirm as well as send booking confirmation to the user's contact |
| 2 | User should be helped appropriately to fill in the mandatory fields, in case of invalid input |
| 3 | System should accept payments via different payment methods, like PayPal, wallets, cards, vouchers, etc |
| 4 | Search results should populate within acceptable time limits |
| 5 | Use of catch and encryption to avoid bots from booking tickets |
| 6 | **Mandatory service level requirements to be fulfilled by a pricing engine** |
| 7 | **Mandatory service level requirements to be fulfilled by a channel.** |
| 8 | **Mandatory service level requirements to be fulfilled by a fare provider / carrier** |
| 9 | As the response of our system, I want the response time less than 0.5 seconds so that it gives the convenience to customers |
| 10 | As the response of our system, I want the available watch time 24 hours so that customers can access and see information of movie all the time |

## **6 Security Requirements**

Before members can book a ticket, they have to create an account with the system with a unique username. Until then they can only access limited sections of the system such as viewing available events.

The system particularly does not accept blank fields or non\_valid input. The system has a unique login section for the admin who can view and carry out a series of activities. Users can logout at any time they fill like since there are provisions for such on any authorized page.

# **CHAPTER FOUR**

## **Agile for software development life cycle**

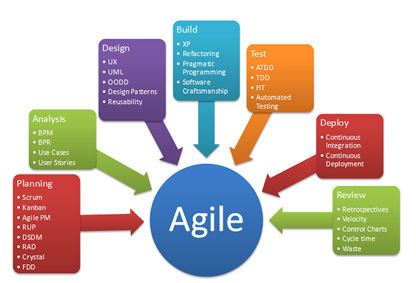
The software development life cycle is Agile due to the following reasons that relate on the project.

### **Advantages**

* Responding to change: according to the project agile development focused on quick responses to change and continue development.
* Individuals and interaction: It enable to use this agile because self-organization and motivation are important as interaction.

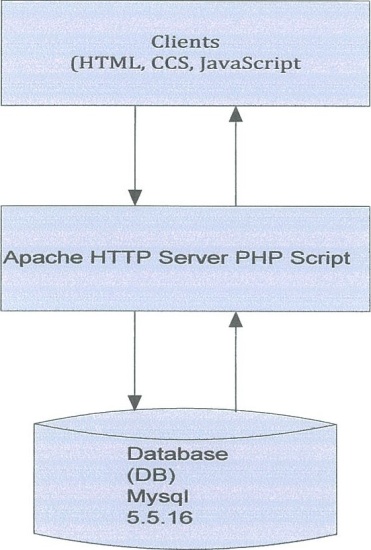
### **Disadvantages**

* Sometimes the agile methodology can take more time and commitment due to testers, customer and developer must constantly interact with each other.
* The starting of the software development life cycle in a level of collaboration can be difficult to maintain.



**7 System architecture**

The system architecture used is a three-tier architecture because the client can access the data from the, server and the database due to the project involve android application that access the data by using network.

Figure 4.1 describes the architectural design of the developed online ticket booking system showing how the system users interact with the graphical user interface and how the interface goes ahead to interact with the database.

## 

## **Conceptual design of the system**

Conceptual design is the first phase of a design where drawings are the primary factor focus. A good conceptual design comprised of the creation of an idea, the exploration of the intentions of an idea and the representation of an idea in form of a plan.

## **User Interface**

This component mainly allows the user to interact with the system. It provides the following services;

1. Authentication of members. The system allows the members to login into the system so as to gain access to the system services.
2. Registration and management of users. Through submission of user profiles to the system, the user can be managed and registered.
3. Data entry forms. The system provides the administrator with data entry forms through which they add booking events and venues into the backend of the system.
4. Reports Display. The system provides the agents with detailed report showing the amount of tickets that have been booked.

## **Database Design**

### **Conceptual database design**

This is the design of a model of the information used in the system independent of all physical consideration. This involves identification of the entities, relationships and attributes.

#### **Entities and their attributes in the System.**

The entity types that were identified were; event, member, agent and administrator entities.

Event is the area or major objects under a search Member is the information consumer

Agent acts as a representative and an authorized user of the online ticket booking system and helps clients to acquire tickets at various venues. Administrator maintains the data in the database.

a. Table 4.3 shows member, agent and administrator entities and their attributes

|  |  |  |
| --- | --- | --- |
| Member | Agent | Administrator |
| member id (PK) | agent id(PK) | administrator id(PK) |
| user id(FK) | user id(FK) | user id(FK) |
| Fname | Fname | Fname |
| Lname | Lname | Lname |
| Password | Password | Password |
| Gender | Gender | Gender |
| Email | Email | Email |
| Addr | Addr | Addr |
| Usertype | Phone | Phone |
|  | Usertype | Usertype |

Table 4.3: Entities and their attributes

#### **Entities and Multiplicities**

The entities were then structured into binary entity relationship diagrams as shown in figure 4.4 below.



1..1

Assumption: A member orders ONE to MANY tickets and a ticket can be ordered by ONLY ONE member

Hosts 1..1

|  |  |  |  |
| --- | --- | --- | --- |
| Event |  |  | Venue |

Assumption: An event can be hosted by ONE venue and a venue can host ONE to many events

Confirms

|  |  |  |
| --- | --- | --- |
| Agent |  | Ticket |
|  | |

Assumption An agent confirms ONE to MANY tickets and a ticket can be confirmed by ONE to MANY agents



1..1

Assumption: An administrator updates ONE to MANY venues and a venue can be updated by ONE administrator



1..1

Updates

Assumption: An administrator updates ONE to MANY event and an event can be updated by ONE administrator



1..1

Assumption: An administrator can add ONE to MANY agents and an agent can be added by one administrator

Figure 4.4: Binary entity relationship diagram

The binary entity diagrams were assembled into an enhanced entity relationship diagram as shown in figure 4.5

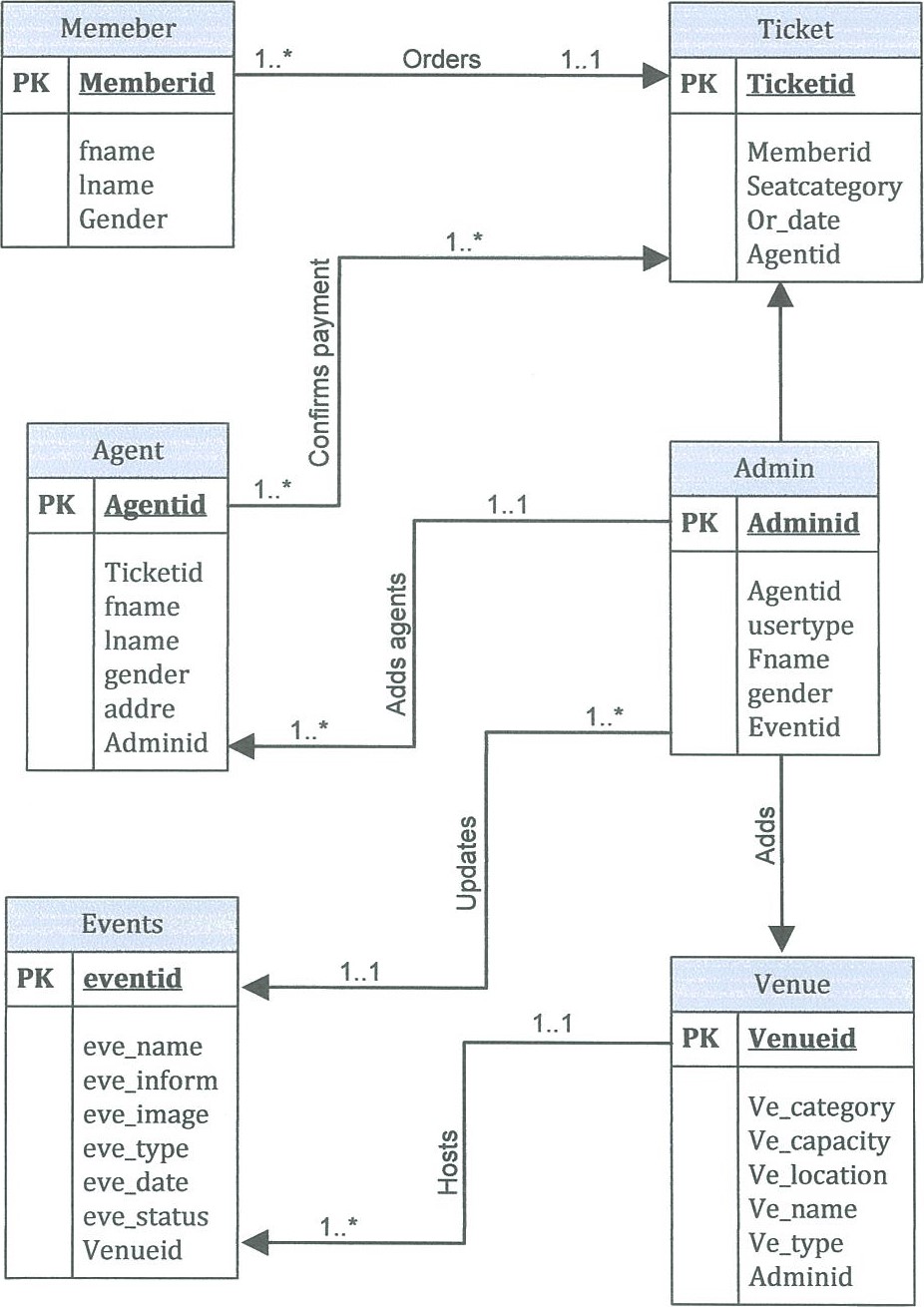


Figure 4.5: Enhanced Entity Relationship Diagram

## **Description of Relationships (Data Dictionary)**

The data dictionary shows a centralized repository of information about data upon various processes such as ticket booking, venue categories.

Table 4.4 describes the administrators, data types description of field name and the outstanding details

|  |  |  |  |
| --- | --- | --- | --- |
| Field Name | Data Type | Description | Outstanding detail |
| Adminid | Int(20) | Admin Id | Not null, PK |
| Userid | Varchar(50) | User Id | Not null, FK |
| Fname | Varchar(50) | Admin's fname | Not null |
| Lname | Varchar(50) | Admin's sname | Not null |
| Password | Varchar(50) | Admin ' spassword | Not null |
| Gender | Varchar( 10) | Admin' sgender | Not null |
| Email | Varchar(50) | Admin' semail | Not null |
| Addr | Varchar(50) | Admin's address | Not null |
| Phone | Varchar(15) | Admin 's phone number | Not null |
| Usertype | Varchar( 10) | Admin's user type | Not null |

Table 4.4: Administrator

Agent's table 4.5 describing agent, data type, the description of the field names and the outstanding details of the agent

|  |  |  |  |
| --- | --- | --- | --- |
| Field Name | Data Type | Description | Outstanding detail |
| Agent id | Int(20) | Agent's id | Not null, PK |
| Userid | Varchar(20) | User's id | Not null |
| Fname | Varchar(50) | Agent's first name | Not null |
| Lname | Varchar(50) | Agent's sir name | Not null |
| Password | Varchar(50) | Agent' s password | Not null |
| Gender | Varchar(10) | Agent's gender | Not null |
| Email | Varchar(50) | Agent's email | Not null |
| Adder | Varchar(50) | Agent's address | Not null |
| Phone | Varchar( 15) | Agent' s phone number | Not null |
| Usertype | Varchar( 10) | User type | Not null |

Table 4.5: Agent

Event table 4.6: describing event, data type, the description of the event and the outstanding details of the event.

|  |  |  |  |
| --- | --- | --- | --- |
| Field Name | Data Type | Description | Outstanding detail |
| Eve id | Int(20) | Event's id | Not null, PK |
| eve name | Varchar(50) | Event's name | Not null |
| eve venue | Varchar(50) | Event's venue | Not null |
| eve\_category | Varchar(50) | Event's category | Not null |
| eve image | Varchar(20) | Event's image | Not null |
| eve info | Varchar(500) | Event's information | Not null |
| eve date | Varchar(50) | Event's date | Not null |
| eve time | Varchar(50) | Event's time | Not null |
| eve status | Varchar(50) | Event's status | Not null |
| Venue id | Int(20) | Venue id | Not null, FK |

Table 4.6: Event

The table 4.7 shows member name, data type, description and the details of the associated member.

|  |  |  |  |
| --- | --- | --- | --- |
| Field Name | Data Type | Description | Outstanding detail |
| Memberid | Int(20) | Member's id | Not null, PK |
| Userid | Varchar(20) | User's id | Not null, FK |
| Fname | Varchar(50) | Member's first name | Not null |
| Lname | Varchar(50) | Member's sir name | Not null |
| Password | Varchar(50) | Member's password | Not null |
| Gender | Varchar( 10) | Member's gender | Not null |
| Email | Varchar(50) | Member's email | Not null |
| Addr | Varchar(50) | Member's address | Not null |
| Phone | Varchar(15) | Member's phone number | Not null |
| Usertype | Varchar( 10) | User's type | Not null |

Table 4.7: Member

Venue table 4.8 showing the fields, data type, description and the outstanding details.

|  |  |  |  |
| --- | --- | --- | --- |
| Field Name | Data Type | Description | Outstanding detail |
| Venueid | Int(20) | Venue's id | Not null, PK |
| ve name | Varchar(50) | Venue's name | Not null |
| ve\_capacity | Varchar(50) | Venue's capacity | Not null |
| ve location | Varchar(50) | Venue's location | Not null |
| ve-phone | Varchar(50) | Venue's phone number | Not null |

Table 4.8: Venue

Below is a ticket table 4.9 showing field names, data types description of the field names and the outstanding details.

|  |  |  |  |
| --- | --- | --- | --- |
| Field Name | Data Type | Description | Outstanding detail |
| Orderid | Int(20) | Order id | Not null, PK |
| Or fname | Varchar(50) | Order first name | Not null, |
| Or sname | Varchar(50) | Order sir name | Not null |
| Or-phone | Varchar(50) | Order phone | Not null |
| Or email | Varchar(50) | Order email | Not null |
| Or-date | Varchar(10) | Order date | Not null |
| Seat cat | Varchar(50) | Seat category | Not null |
| or status | Varchar(50) | Order status | Not null |
| Memberid | Int(20) | Member's id | Not null, FK |
| Agentid | Int(20) | Agent's id | Not null, FK |
| Eventid | Int(20) | Event's id | Not null, FK |

Table 4.9: Ticket

Activity flow of the system

**Data Flow Diagram**

Figure 4.3 shows the flow of information for user's log in into the system, up to the administrator updating the database and viewing of generated reports

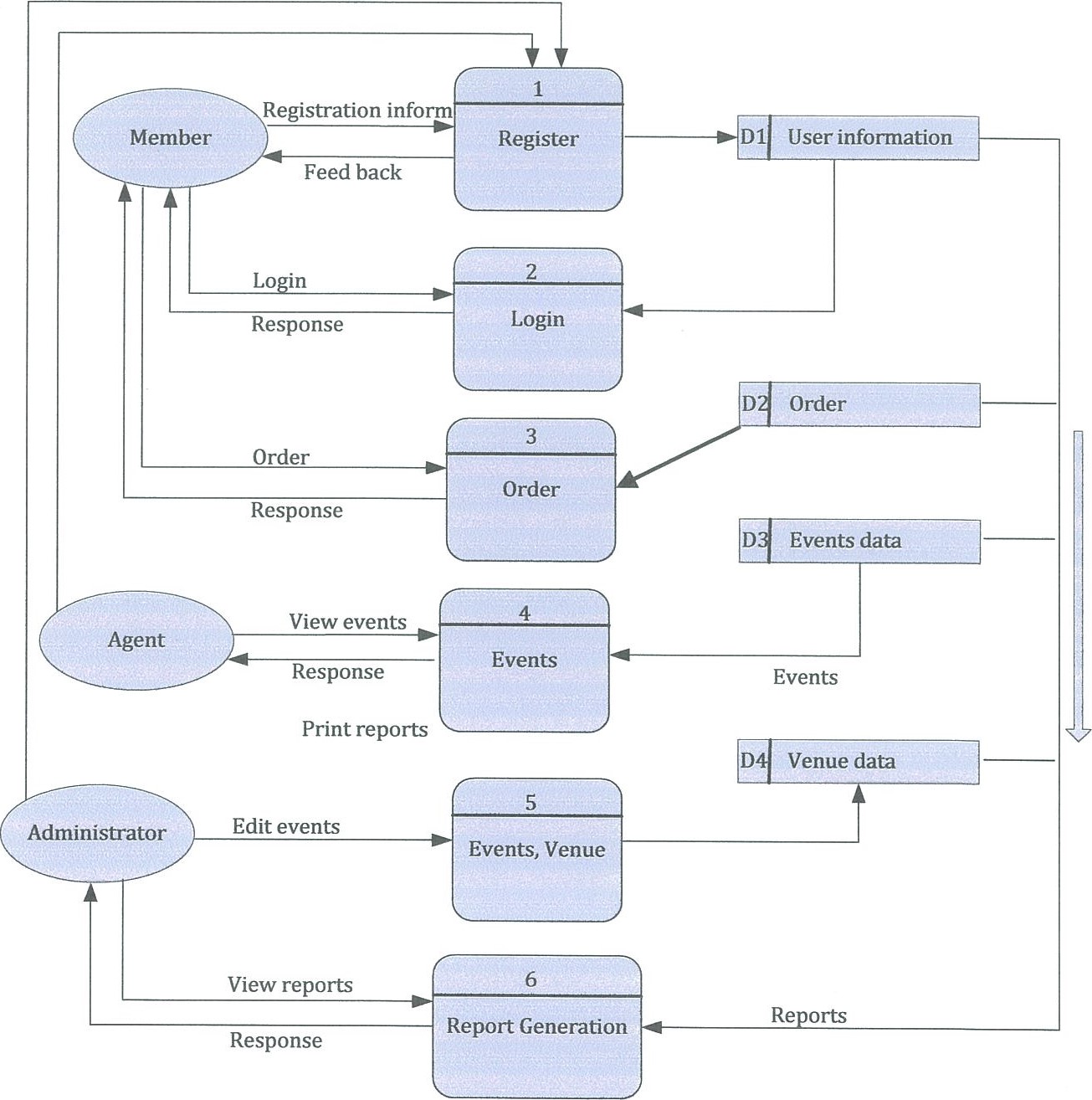
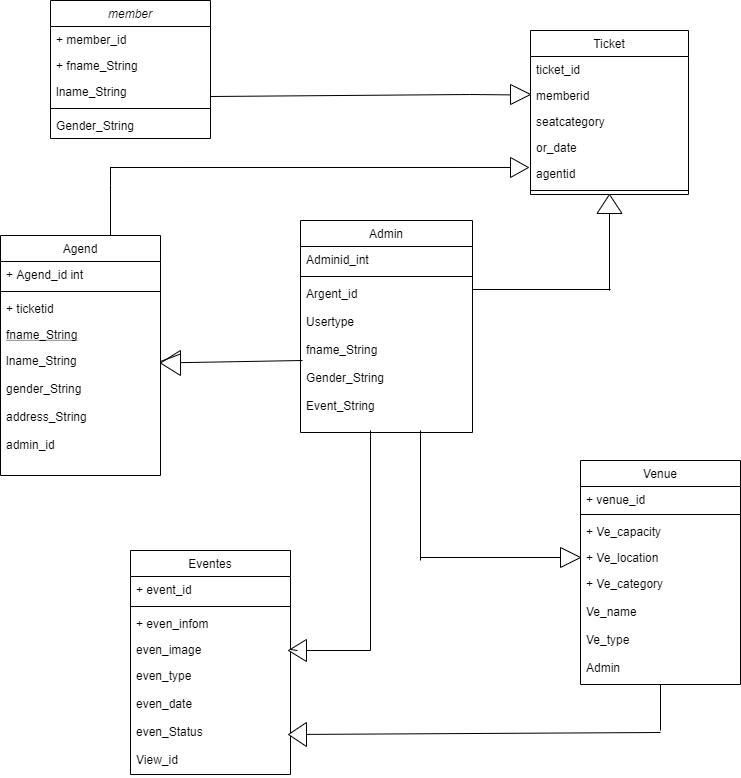


Figure 4.3 : Shows the flow of information for user's log in into the system

## **Class Diagram Requirement**



## **Development technology (languages).**

Angular for use frame, html, mysql for database, Spring boot