

# Implementation of Fest Management System and Event Handling

A PROJECT REPORT

*Submitted*

**Om Mule – 19BCE0439**

**Siddharth Chatterjee – 19BCE2249**

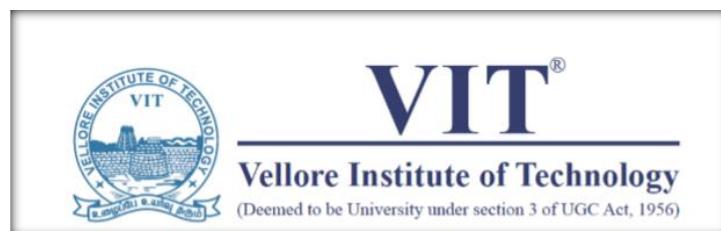
**Ishan Jogalekar – 19BCE2250**

**Apurv Shewale - 19BCE0167**

**CSE2004 -- Database Management Systems**

**Project Guide: Pradeepa M**

**B Tech CSE --- Scope**



**Fall Semester 2020-21**

## **Abstract:**

- When colleges conduct their technical and cultural fests, they usually contain a lot of events which are required to be managed efficiently for the fest to be a success. Now there are might be many different ways to use and implement a particular DBMS model but in this project, we propose the use of Distributed Database Management System (DDBMS) to deal with registration of the events and the transactions involved.
- DDBMS involves storing data over multiple servers or database applications and consists of loosely coupled repositories of data. It works as a single database system, even though the database hardware might be run by systems in different locations.
- DDBMSs have a high degree of distribution transparency (Network Transparency, Replication Transparency and Fragmentation Transparency) and this will make it much simpler for the using client to access the database in case of our project.
- For a simple integrated schema of all data, as in our case, Event management; DDBMS proves very advantageous even with distributing the requirement analysis to suit 2 Entity-Relationship Diagrams.
- In Review 1 we lay out the Problem Statement and the Objective of our project and consequently the Literature Review with research on 5 Reference Papers as well as Journals. We then conclude with a brief requirement analysis and an overview of the proposed ER diagram.
- In Review 2, we start with an elaboration on requirement analysis and construct 2 separate ER diagrams (DDBMS approach), follow it up with screenshots of queries used in creation of Database, schema and relational diagrams.
- In Review 3, we delve deep into how we have connected our database tables to both the frontend as well as backend and how the creation of tables are saved and the events are registered from the client point of view.

## Table of Contents

	Page #
<b>1. Introduction</b>	<b>4</b>
<b>2. Literature Review</b>	<b>4</b>
<b>3. Problem Statement</b>	<b>5</b>
<b>4. Objectives</b>	<b>5</b>
<b>5. Software Requirements</b>	<b>5</b>
<b>6. Implementation</b>	
a. Requirement Analysis	5
b. ER Diagram and Relational Schema	7
c. ERD Illustration Model using ERD Assistant Software	8
d. Schema and Relational Diagram	9
e. Generalized Interface	10
f. Normalization	11
<b>7. Table Creation in Oracle 10g</b>	
a. Table Club	20
b. Table Details	20
c. Table Card	21
d. Table Event	21
e. Table Student	22
f. Table student_transaction_history	22
g. Table registered_for	23
h. Table payment	23
i. Table Venue	23
<b>8. Frontend Interface</b>	<b>26</b>
<b>9. Backend Code for Webpage in Java</b>	<b>37</b>
<b>10. Java Application</b>	<b>59</b>
<b>11. Final Conclusion</b>	<b>66</b>
<b>12. References</b>	<b>67</b>
<b>13. Appendices</b>	
a. HTML Frontend Code for Webpage	68
b. Java Application Code	91

## **Introduction:**

1. During events and fests that are conducted in colleges, there is a huge challenge faced by the various clubs for the management of all the information of the many students who take part in various events.
2. In order to solve this problem, we have come up with the solution of **DDBMS (Distributed Database Management Systems)**.
3. **DDBMS** basically helps to fetch data from various locations and access it all at one location; this helps us to maintain the data properly and also access it whenever required with ease.

## **Literature Review:**

1. In the aforementioned reference papers and journals from leading organisations, we can notice that the DBMS provides users and programmers with a systematic way to create, retrieve, update and delete data in a database.
2. Our research work based on literature review establishes an important theoretical setting to the study of event management and the various details like Title, Author, Publication Date, Problem Statements, Objectives, Results and limitations.
3. There is also a section of proposed methodology of literature review under each Reference Paper and Journal
4. The Reference Papers and Journals usually have applications containing the database which has the details of the participants, their name, and the events they are willing to participate, their registration ID, and event details like day, venue, time, etc. Instead of using paperwork, participant can simply use this mobile application to register.

## **Problem Statement:**

1. To implement a DDBMS (Distributed Database Management System) for organizing the College fest and other events more efficiently.
2. Also properly use the database for handling the registration of the events and also with the transaction processes thus maintaining the proper organized information about the same.

## **Objectives:**

1. Cab booking apps like Ola and Uber are known to store their ride details and payment details on two different databases. The redundant data like ride details is usually stored on cost-effective databases and payment details are stored on premium servers
2. This way they are able to cut down their spending and also are able to manage their data in an effective and untangling way.
3. Making use of the idea we will try to use **heterogenous DDBMS** by storing the **event information in one database** and **transactional details in another database**.

## **Software Requirements:**

- 1. Oracle 10g database**
- 2. JAVA jdk 1.8 version, Java Swing Library**
- 3. Ojdbc library for database connections , Ojdbc 8 version**
- 4. ER Assistant Software**
- 5. Visual Paradigm Software**
- 6. Java IDE (Netbeans)**

- 7. Web browser (Google Chrome)**
- 8. IntelliJ IDEA**
- 9. Atom Text Editor (Any other may do)**
- 10. HTML references from web**
- 11. Bootstrap**

## **Implementation:**

### **Requirement Analysis:**

#### **1. Club Entity:**

It will give the information about clubs which are going to conduct event , so Clubs attribute has primary key (Club name) and 1 or many relation with Event table

1. Name and Description attribute
2. Faculty Co-Ordinator attribute
- 3.Contact information of club

As club co-Ordinator info can be given in club description so this attribute is not compulsory to fill. So, Club entity is connected to Event entity via relationship entity “Organized by”. And it will have one or many relationships with “Organized by” relationship entity. And “Organized by” relationship has zero or one relationship with Event table

#### **2. Venue Entity:**

- 1. Building attribute**
- 2. Room no. attribute:**

It will contain information of venue of events. Where Building and Room no. have primary key with Venue entity.

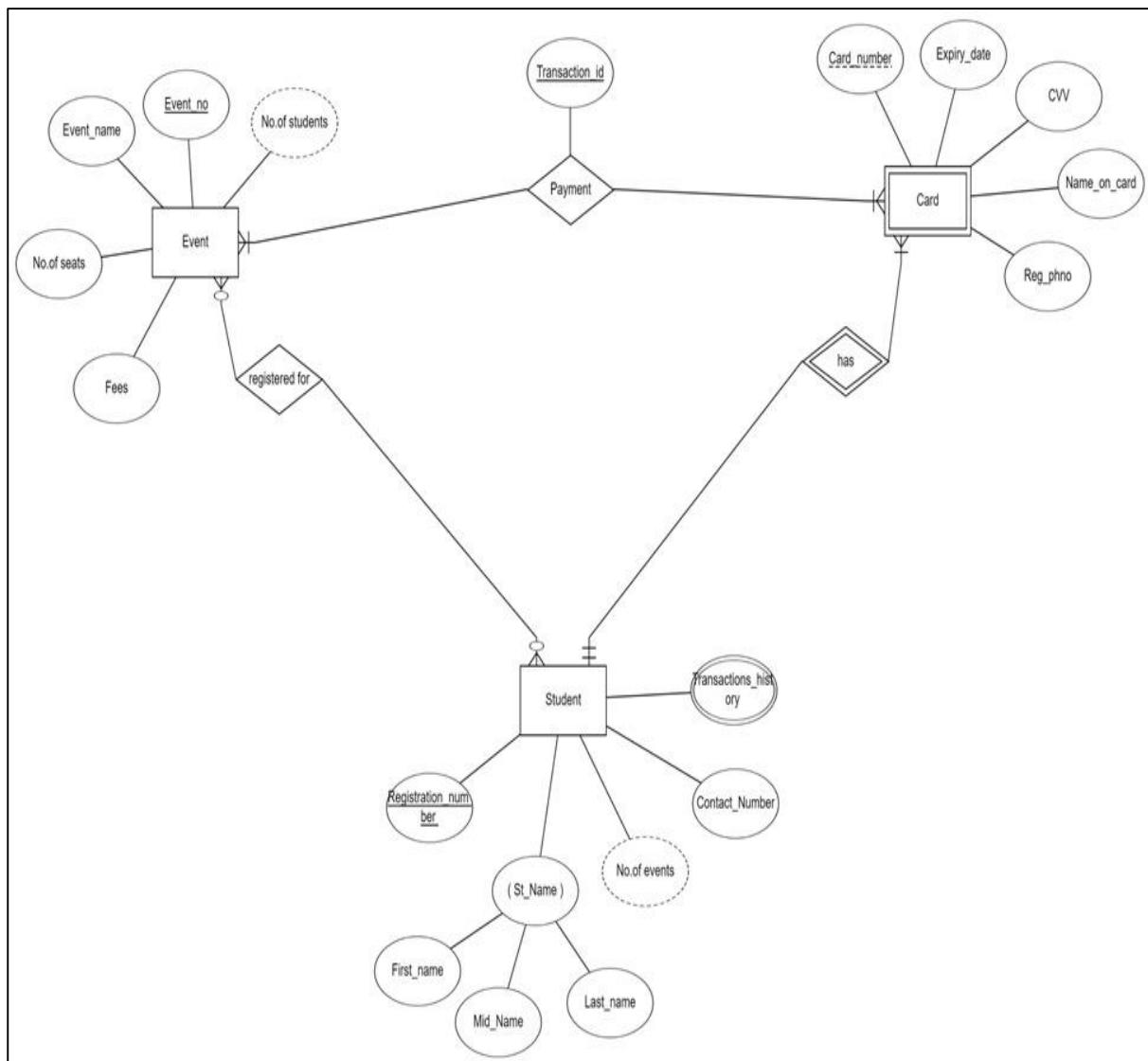
- 3. No of events attribute:**

This is derived attribute of Venue because information can be calculated from Venue attribute and Events information.

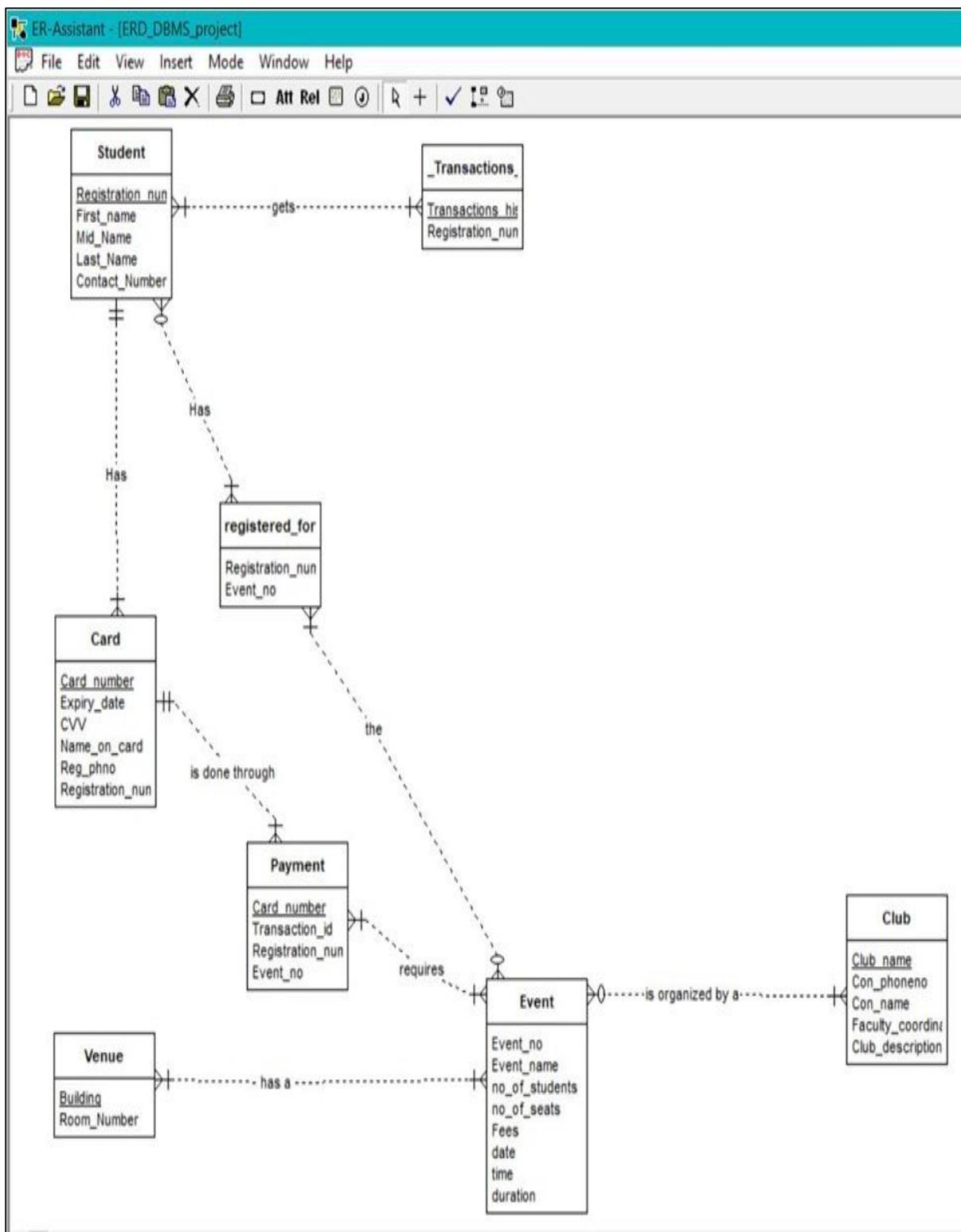
So, this is Venue entity has one or many relationships with “Has a” relationship- entity. And “Has a” relationship has one and only one relation with Event entity.

Here Club and Venue are important entities of Event table database.

## **Er Diagram & Relation Schema:**

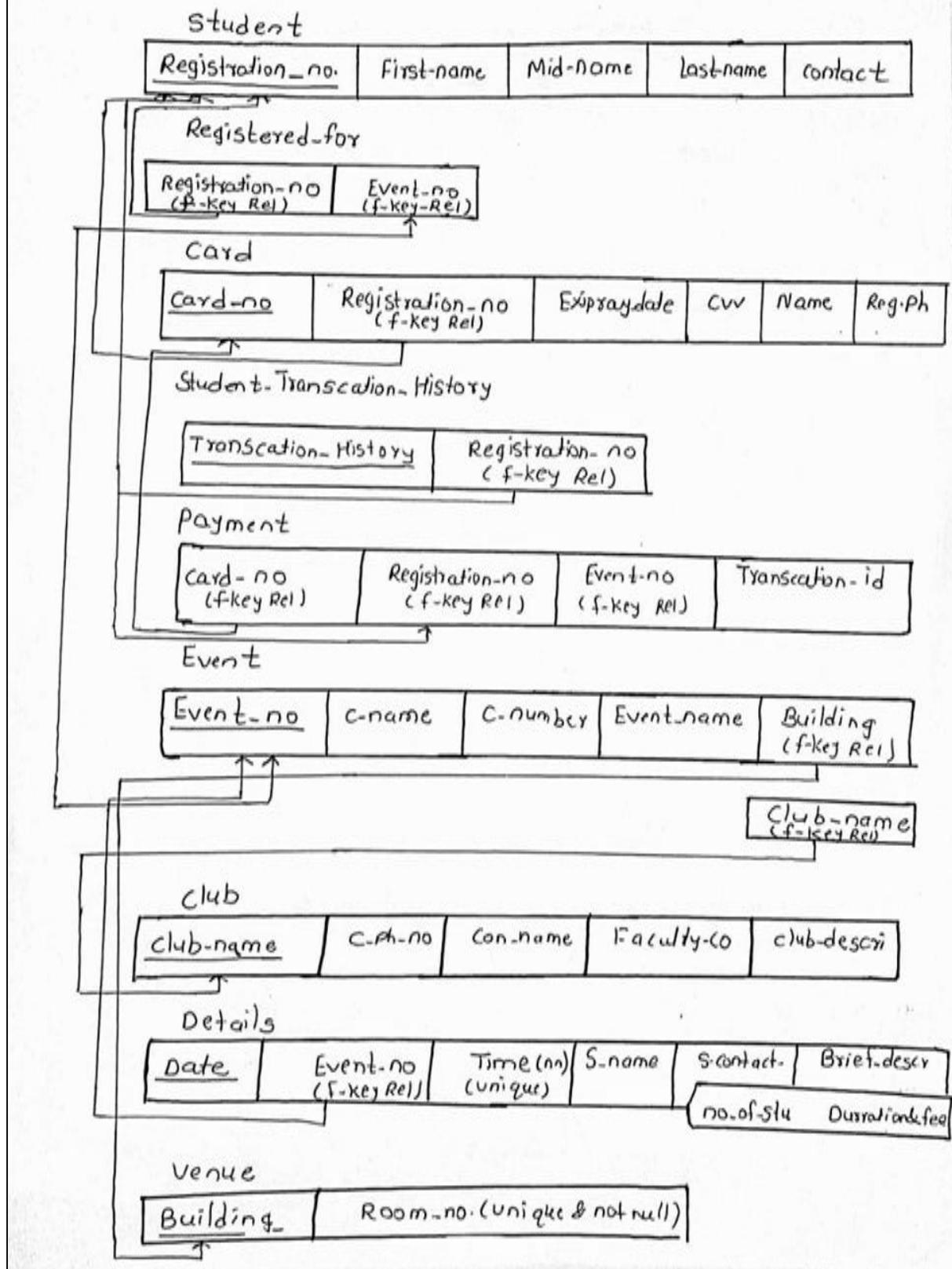


# ERD's Illustration Model using ER-Assistant Software

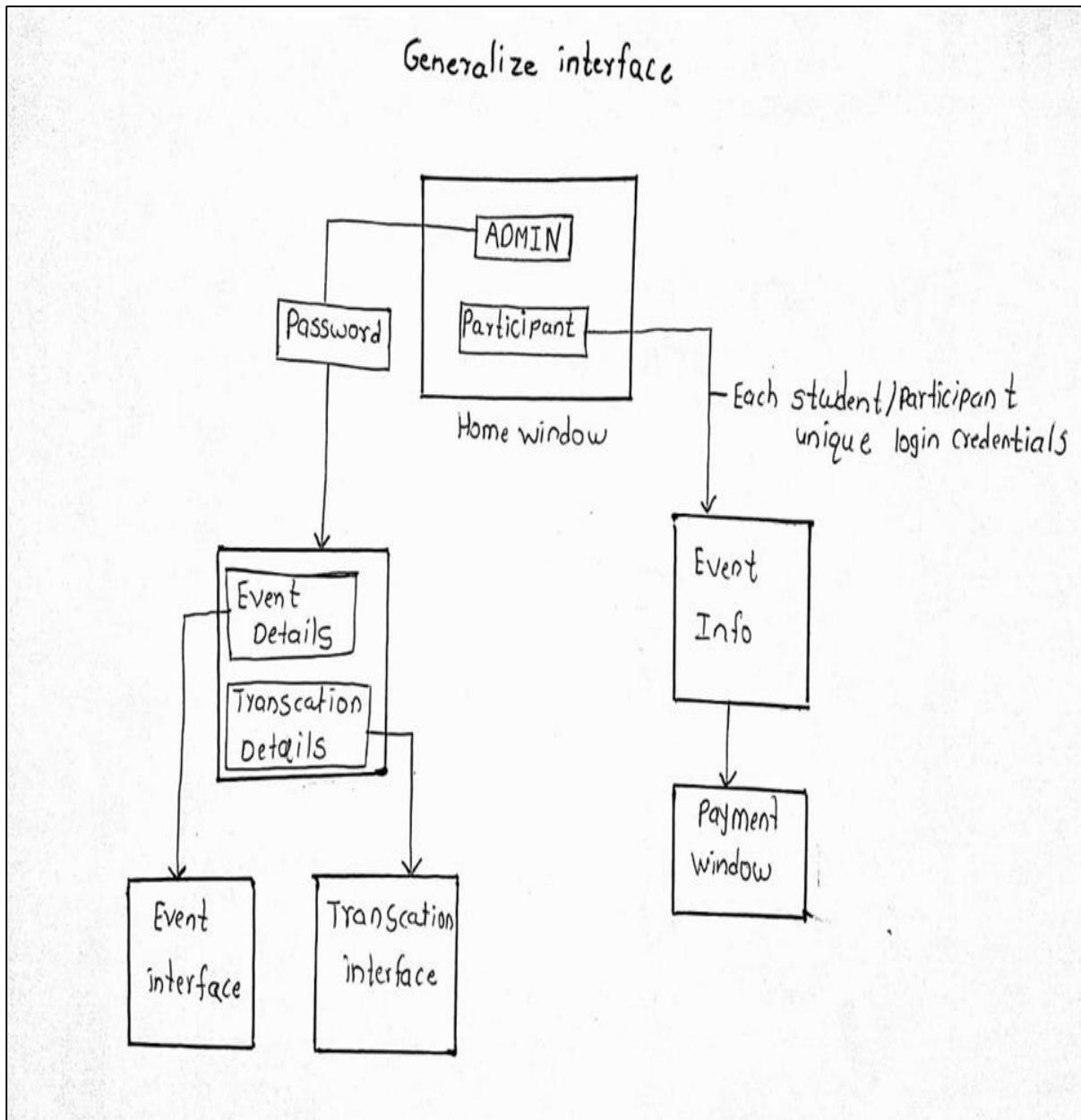


## Schema and Relational Diagram

Relational diagram :-



## Generalized Interface



## Normalization:

STAG	(2)	DATE	PAGE
<u>Functional dependencies :-</u>			
1) Student table :-			
<u>Registration_no</u> <u>First_name</u> <u>Mid_name</u> <u>Last_name</u> <u>Contact</u>			
In student table Registration_no is primary key and it is functionally dependent on First_name of Student. Every Student has its own unique Registration_no and contact. ∴ Contact is functionally dependent on First-name so,			
$\text{Registration\_no} \rightarrow \text{First\_name}$ $\text{First\_name} \rightarrow \text{Contact}$ ∴ Registration_no → contact also (Transitive candidate key, CK {Reg_no, First_name} property)			
2) Registered_for :-			
Registered-for table contain Registration_no and Event_no attributes. Where both are independent. So, there is no functional dependencies in table Registered for.			
<u>Registration-no</u> (f-key-Rel) <u>Event_no</u> (f-key-Rel)			
3) Card :-			
<u>Card_no</u> <u>Reg_no</u> <u>Expiry-date</u> <u>CVV</u> <u>Name</u> <u>Reg.Ph</u> (f-key-Rel)			
In card table, there is card details as entries in table. Card-no is primary key and it is functionally dependent to Expiry-date, CVV, Name & Reg.Ph.			

So,

$\text{card\_no} \rightarrow \text{Expiry\_date}$

$\text{card\_no} \rightarrow \text{CVV}$

$\text{card\_no} \rightarrow \text{Name}$

$\text{card\_no} \rightarrow \text{Reg\_Ph}$

$\therefore \text{card\_no} \rightarrow \text{Expiry\_date}, \text{CVV}, \text{Name}, \text{Reg\_Ph}$

$\therefore \text{candidate key } CK \{ \text{card\_no} \}$

4) Student Transaction History :-

Transcation\_History       $\downarrow$       Reg-no

(F-key-Rel)

Transcation\_History and Reg-no are two attributes of student\_Transcation\_History.

Transcation\_History is primary key so it is unique and not null for every Registration no and Reg no is f. key related to Reg no of Student table Transcation\_History is functionally dependent to Reg-no.

$\therefore \text{Transcation_History} \rightarrow \text{Reg\_no}$

$\therefore \text{candidate Key , CK } \{ \text{Transcation_History} \}$

5) Payment :-

card-no       $\downarrow$       Registration-no       $\downarrow$       Event no       $\downarrow$       Transcation id  
 $(Rcd)$                            $(Rel)$                            $(Rel)$

In Payment table Transcation\_id is primary key and functionally related to Event no and card-no is related to Registration-no, where all other attributes are foreign keys.

So,

Transcation\_id → Event\_no

card - no → Registration no

candidate key. ck { Transaction\_id , card\_no }

6) Event :-

Event\_no    C\_name.    C-number    Event\_name    Building    C\_name.

In Event table Event-no is primary key related to Event-name, Building and Cl-name and C-name is functionally dependent to C-number.

Building and cl-name both are foreign key.

so,

Event\_no → Event\_name, Building, cl\_name

C-name  $\rightarrow$  C-number

Candidate key, CK f Event\_no, C-name }

7) Club :-

Club-name C\_Phone Con<sup>↓</sup>name Faculty-co clubdesc

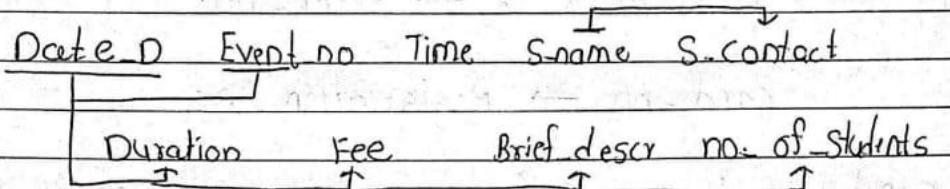
In club table club-name is functionally related to con-name and club-desc, where club-name is primary key and prime attribute. con-name is also functionally relate to c.phno.

so club-name → con-name, club-desc

con name → c-ph no

∴ candidate key, ck f Club-name, can name?

8) Details :-



In Details table Date\_D is primary key, so it is unique and not null.

Event\_no is dependent of Duration, Fee, Brief\_descr and no. of students and S-name is functionally dependent of S-contact.

So,

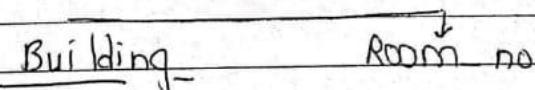
Event\_no → Duration, Fee, Brief-descr, no-of-students.

S-name → S-contact

Here, Date\_D, Event\_no, Duration, Fee, no-of-students and Brief-description all are prime attributes candidate key,

CK {Event-no, S-name}

9) Venue :-



In venue table Building is primary key and Room\_no is unique and not null, so both are prime attributes. Building is functionally dependent on Room\_no.

So,

Building → Room\_no

Candidate key,

CK {Building}

Normal form :-

1<sup>st</sup> Normal form :-

In 1<sup>st</sup> NF table should not contain any multivalued attribute.

so, to keep all tables in 1 NF, we applied unique and not null constraint on all attributes.

Here, all tables are in 1 NF form so, we can proceed to check 2 NF of tables.

2<sup>nd</sup> Normal form :-

conditions for 2 NF are tables must be in 1 NF and all the non-prime attributes should be fully functionally dependent on candidate key.

1) Student table :-

Registration no → First name

First name → contact

Candidate key { Registration no, First name }

Prime attributes { Registration no, First name, contact }

There is no partial dependency in this table so, this table is in 2 NF.

2) Registered for table :-

This table does not have any functional dependencies relations between attributes Registration no and Event no.

so, they are independent attributes. so, this table satisfies all conditions of 2 NF and 3 NF also BCNF.

3) card table :-

card-no → Expiry\_date, CVV, Name, Reg\_ph

Candidate key, { card-no }

Prime attributes, { card-no, Expiry\_date, CVV, Name, Reg\_ph }

Here, All non-prime attributes fully dependent on  
candidate key. So this table is in 2NF.

4) Student Transaction History:-

Transaction\_History → Reg\_no

Candidate key, { Transaction\_History }

prime attributes { Transaction\_History, Reg\_no }

So, there is no data redundancies in this table.  
This table is in 2NF.

5) Payment table :-

Transaction\_id → Event\_no

Card-no → Registration\_no

Candidate key, { Transaction\_id, Card\_no }

Prime attributes { Transaction\_id, Card\_no, Event\_no, Registration\_no }

There is no partial dependencies in this table.  
so this table is in 2NF.

6) Event table :-

$\text{Event-no} \rightarrow \text{Event-name, Building, cl-name}$   
 $\text{C-name} \rightarrow \text{C-number}$

Candidate Key {Event-no, C-name}

Prime attributes {Event-no, Event-name, C-name, Building, cl-name}

Here there is a partial dependency between,  
 $\text{C-name} \rightarrow \text{C-number}$

So, we split it between two tables, where Event-no will be a common attribute

Event 1 :-

$\text{Event-no} \rightarrow \text{Event-name, Building, cl-name}$

Event 2 :-

$\text{Event-no} \rightarrow \text{C-name}$

$\text{Event-no} \rightarrow \text{C-number}$

$\text{C-name} \rightarrow \text{C-number}$

Now, Event table is in 2 NF.

7) Club table :-

$\text{club-name} \rightarrow \text{con-name, club-desc}$

$\text{con-name} \rightarrow \text{C-phno}$

Candidate key {club-name, con-name}

Prime attributes {club-name, con-name, club-desc}

So, there is no partial dependencies in this table  
Club table satisfies conditions of 2 NF.

8) Details table :-

Event\_no → Duration, Fee, Brief-descr, no\_of\_student  
S-name → S-contact

Candidate key {Event\_no, S-name}

There is no partial dependency.  
Hence Details table is in 2NF.

9) Venue :-

Building → Room\_no

Candidate key {Building}

Prime attribute {Building, Room\_no}

Hence, there is no partial dependency in this table.  
So, this table is in 2NF.

• 3<sup>rd</sup> Normal form :-

1) Student table :-

There is transitive dependencies in  
Student table.

Registration no  $\rightarrow$  First-name

First-name  $\rightarrow$  contact

Candidate key { Registration-no, First-name }

Prime attributes { Registration-no, First-name, Contact }

$\therefore$  we split tables,  
Student 1 table :-

Registration no  $\rightarrow$  First-name

student 1 , (Registration-no, First-name, Mid-name, Last name)

student 2 table :-

First-name  $\rightarrow$  contact

Candidate key { First-name }

prime attributes { First-name }

student 2 , (First name, contact)

BCNF:-

For BCNF , if  $X \rightarrow Y$  then Y should have  
trivial dependencies.

Hence, All table are in BCNF form.

## Table Creation In Oracle 10g:

```
SQL> CREATE TABLE Club
  2  (
  3    Club_name varchar2(30) NOT NULL,
  4    Con_phoneno char(10) NOT NULL,
  5    Con_name varchar2(30) NOT NULL,
  6    Faculty_coordinator varchar2(30) NOT NULL,
  7    Club_description varchar2(80)NOT NULL,
  8    constraint TC_CN_pk PRIMARY KEY (Club_name)
  9  );

Table created.

SQL> desc Club;
Name                           Null?    Type
-----                         -----
CLUB_NAME                      NOT NULL VARCHAR2(30)
CON_PHONENO                     NOT NULL CHAR(10)
CON_NAME                        NOT NULL VARCHAR2(30)
FACULTY_COORDINATOR             NOT NULL VARCHAR2(30)
CLUB_DESCRIPTION                NOT NULL VARCHAR2(80)

SQL>
```

```
SQL> CREATE TABLE Details
  2  (
  3    D_Date Date NOT NULL ,
  4    Time varchar2(20) NOT NULL,
  5    S_name varchar2(30) NOT NULL,
  6    S_Contact char(10) NOT NULL,
  7    Brief_description varchar2(80) NOT NULL,
  8    No_of_seats INT NOT NULL,
  9    Duration varchar2(30) NOT NULL,
 10    Fees INT NOT NULL,
 11    Event_no INT NOT NULL,
 12    constraint TD_UN UNIQUE (Time)
 13  );

Table created.

SQL> desc details;
Name                           Null?    Type
-----                         -----
D_DATE                         NOT NULL DATE
TIME                           NOT NULL VARCHAR2(20)
S_NAME                          NOT NULL VARCHAR2(30)
S_CONTACT                       NOT NULL CHAR(10)
BRIEF_DESCRIPTION              NOT NULL VARCHAR2(80)
NO_OF_SEATS                     NOT NULL NUMBER(38)
DURATION                        NOT NULL VARCHAR2(30)
FEES                            NOT NULL NUMBER(38)
EVENT_NO                        NOT NULL NUMBER(38)

SQL>
```

```

SQL> CREATE TABLE Card
  2  (
  3    Card_number INT NOT NULL,
  4    Expiry_date INT NOT NULL,
  5    CVV INT NOT NULL,
  6    Name_on_card varchar2(20) NOT NULL,
  7    Reg_phno char(10) NOT NULL,
  8    Registration_number varchar2(20) NOT NULL,
  9    constraint TC_pk PRIMARY KEY (Card_number)
 10  );
Table created.

SQL> desc Card;
Name                           Null?    Type
-----                         -----
CARD_NUMBER                    NOT NULL NUMBER(38)
EXPIRY_DATE                   NOT NULL NUMBER(38)
CVV                           NOT NULL NUMBER(38)
NAME_ON_CARD                  NOT NULL VARCHAR2(20)
REG_PHNO                      NOT NULL CHAR(10)
REGISTRATION_NUMBER           NOT NULL VARCHAR2(20)

SQL>

```

```

SQL> CREATE TABLE Event
  2  (
  3    Event_no INT NOT NULL,
  4    C_name varchar2(20) NOT NULL,
  5    C_contact char(10) NOT NULL,
  6    Event_name varchar2(30) NOT NULL,
  7    Building varchar2(20) NOT NULL,
  8    Club_name varchar2(30) NOT NULL,
  9    constraint TableEv_pk PRIMARY KEY (Event_no)
 10  );
Table created.

SQL> desc event;
Name                           Null?    Type
-----                         -----
EVENT_NO                       NOT NULL NUMBER(38)
C_NAME                         NOT NULL VARCHAR2(20)
C_CONTACT                      NOT NULL CHAR(10)
EVENT_NAME                     NOT NULL VARCHAR2(30)
BUILDING                       NOT NULL VARCHAR2(20)
CLUB_NAME                      NOT NULL VARCHAR2(30)

SQL>

```

```
SQL> CREATE TABLE Student
  2  (
  3    Registration_number varchar2(20) NOT NULL,
  4    First_name varchar2(30) NOT NULL,
  5    Mid_Name varchar2(30) NOT NULL,
  6    Last_name varchar2(30) NOT NULL,
  7    Contact_Number char(10) NOT NULL,
  8    constraint SR_pk PRIMARY KEY (Registration_number)
  9  );
```

Table created.

```
SQL> desc student;
```

Name	Null?	Type
REGISTRATION_NUMBER	NOT NULL	VARCHAR2(20)
FIRST_NAME	NOT NULL	VARCHAR2(30)
MID_NAME	NOT NULL	VARCHAR2(30)
LAST_NAME	NOT NULL	VARCHAR2(30)
CONTACT_NUMBER	NOT NULL	CHAR(10)

```
SQL>
```

```
SQL> CREATE TABLE Student_Transactions_history
```

```
 2  (
  3    Transactions_history varchar2(20) NOT NULL,
  4    Registration_number varchar2(20) NOT NULL,
  5    constraint TH_pk PRIMARY KEY (Transactions_history)
  6  );
```

Table created.

```
SQL> desc Student_Transactions_history;
```

Name	Null?	Type
TRANSACTIONS_HISTORY	NOT NULL	VARCHAR2(20)
REGISTRATION_NUMBER	NOT NULL	VARCHAR2(20)

```
SQL>
```

```
SQL> CREATE TABLE registered_for
  2  (
  3    Registration_number varchar2(20) NOT NULL,
  4    Event_no INT NOT NULL);
```

Table created.

```
SQL> desc registered_for;
Name                           Null?    Type
-----                         -----
REGISTRATION_NUMBER            NOT NULL VARCHAR2(20)
EVENT_NO                       NOT NULL NUMBER(38)
```

SQL>

```
SQL> CREATE TABLE Payment
  2  (
  3    Transaction_id INT NOT NULL,
  4    Card_number INT NOT NULL,
  5    Registration_number varchar2(20) NOT NULL,
  6    Event_no INT NOT NULL,
  7    constraint TPCN_pk PRIMARY KEY (Card_number));
```

Table created.

```
SQL> desc Payment;
Name                           Null?    Type
-----                         -----
TRANSACTION_ID                NOT NULL NUMBER(38)
CARD_NUMBER                   NOT NULL NUMBER(38)
REGISTRATION_NUMBER           NOT NULL VARCHAR2(20)
EVENT_NO                      NOT NULL NUMBER(38)
```

SQL>

```
SQL> CREATE TABLE Venue
  2  (
  3    Building varchar(20) NOT NULL,
  4    Room_Number INT NOT NULL,
  5    constraint TV_pk PRIMARY KEY (Building),
  6    constraint TV_RN_UN UNIQUE (Room_Number)
  7  );
```

Table created.

```
SQL> desc Venue;
Name                           Null?    Type
-----                         -----
BUILDING                      NOT NULL VARCHAR2(20)
ROOM_NUMBER                   NOT NULL NUMBER(38)
```

SQL>

```
SQL> alter table Card
  2 add constraint Card_RN_Rel FOREIGN KEY (Registration_number)
  3 REFERENCES Student(Registration_number);
```

Table altered.

```
SQL>
SQL> alter table Student_Transactions_history
  2 add constraint STHRN_rel FOREIGN KEY (Registration_number)
  3 REFERENCES Student(Registration_number);
```

Table altered.

```
SQL> alter table Event
  2 add constraint TEB_Rel FOREIGN KEY (Building)
  3 REFERENCES Venue(Building);
```

Table altered.

```
SQL>
SQL> alter table Event
  2 add constraint TECN FOREIGN KEY (Club_name)
  3 REFERENCES Club(Club_name);
```

Table altered.

```
SQL> alter table Details
  2 add constraint DeE_Rel FOREIGN KEY (Event_no)
  3 REFERENCES Event(Event_no);
```

Table altered.

```
SQL> alter table registered_for
  2 add constraint RFRN_Rel FOREIGN KEY (Registration_number)
  3 REFERENCES Student(Registration_number);
```

Table altered.

```
SQL>
SQL>
SQL> alter table registered_for
  2 add constraint RFEN_Rel FOREIGN KEY (Event_no)
  3 REFERENCES Event(Event_no);
```

Table altered.

```
SQL>
SQL> alter table Payment
  2 add constraint PCN_Rel FOREIGN KEY (Card_number)
  3 REFERENCES Card(Card_number);

Table altered.

SQL> alter table Payment
  2 add constraint PRN_Rel FOREIGN KEY (Registration_number)
  3 REFERENCES student(Registration_number);

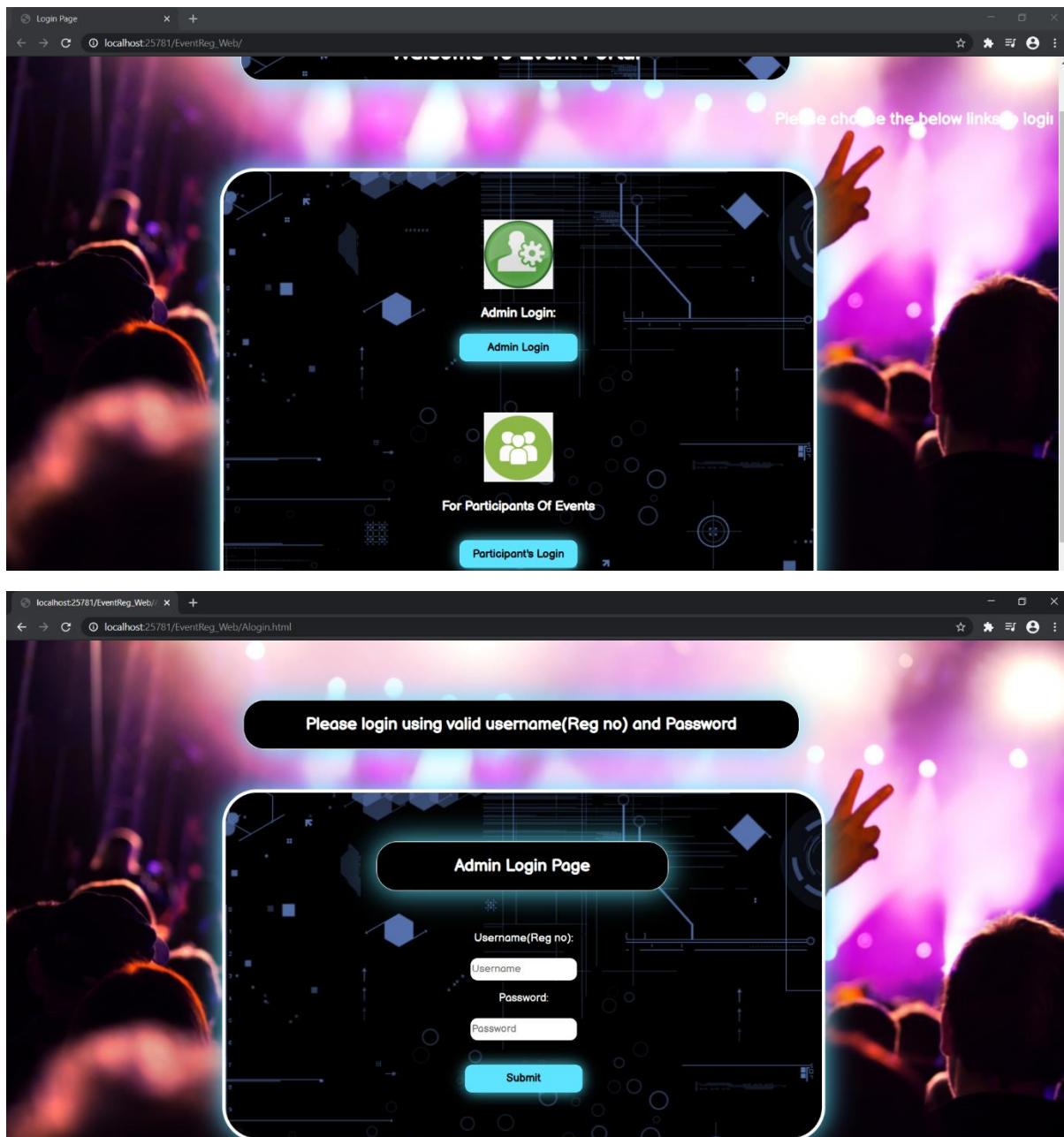
Table altered.

SQL>
SQL> alter table Payment
  2 add constraint PEN_Rel FOREIGN KEY (Event_no)
  3 REFERENCES Event(Event_no);

Table altered.

SQL>
```

## Frontend Interface:



**!! Please Enter Valid Username & Password !!**

Please login using valid username(Reg no) and Password

**Admin Login Page**

Username(Reg no):

Password:

Submit

**!! Welcome Admin !!**

Logout

For Event Details

Event Details

For Transaction Details

Transaction Details

**Event Management**

First Enter Club details then venue and then add Event details

1. Create Event Link [Create Event](#)
2. View Event Link [View Event](#)
3. Create Club Link [Create Club](#)
4. View Club Link [View Club](#)
5. Create Venue [Create Venue](#)

**Create Event**

Event Number:

Event Name:

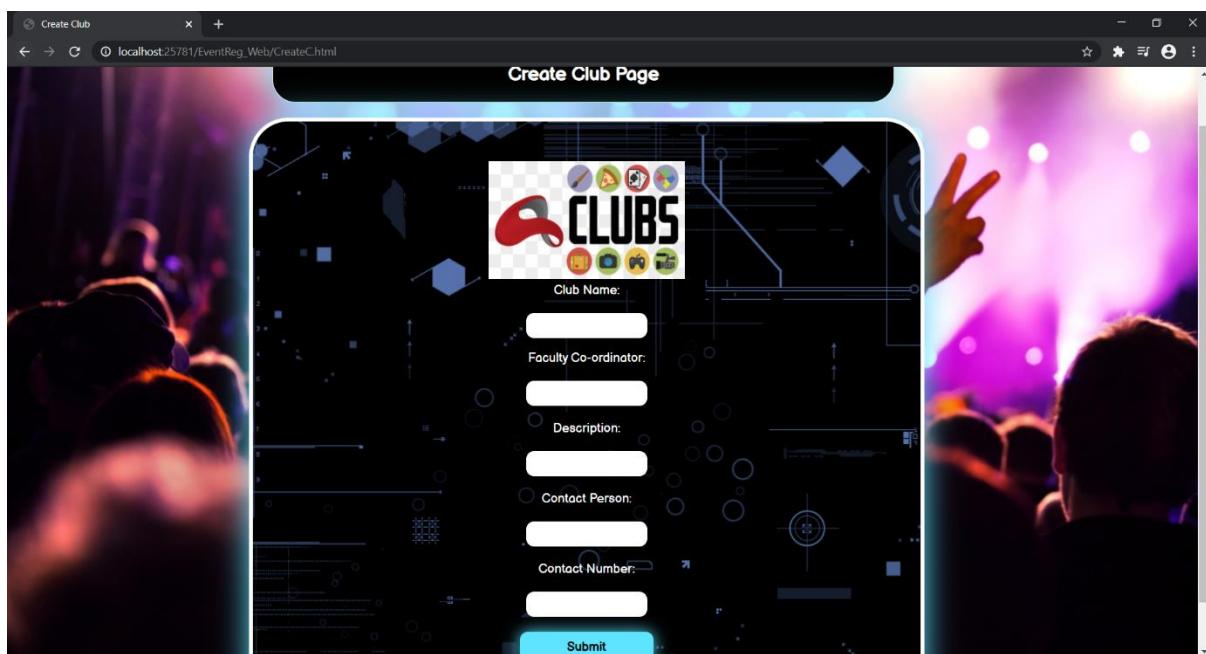
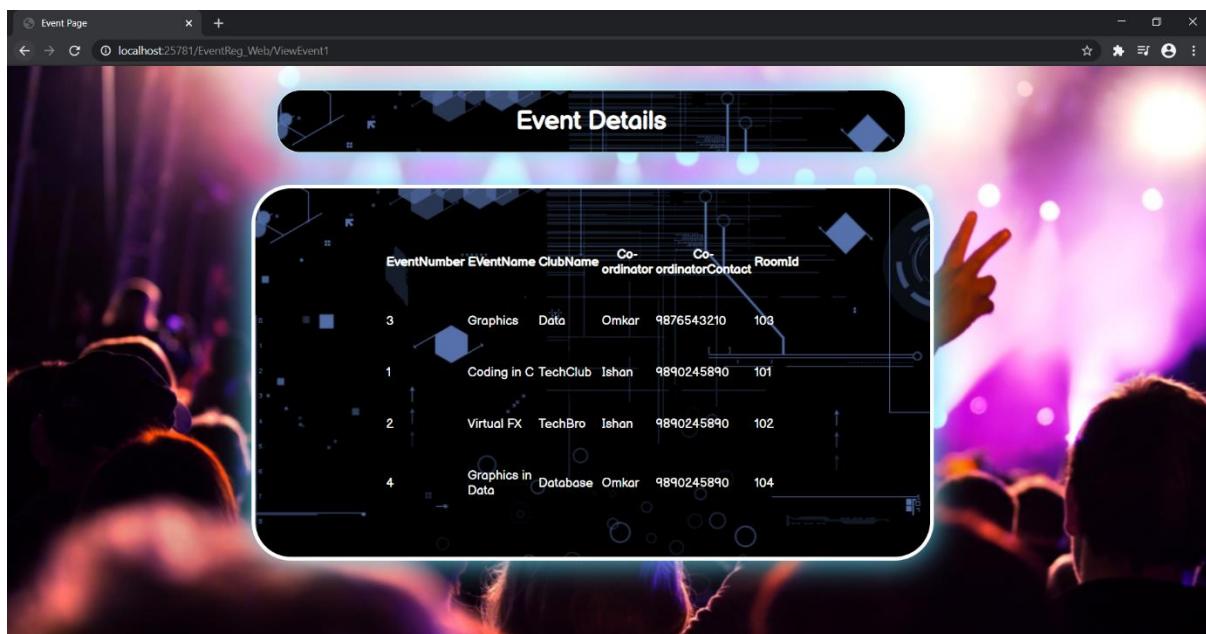
Club Name:

Co-ordinator Name:

Co-ordinator Number:

Room Number:

**Submit**



The image displays two screenshots of a web application interface, "Venue Details" and "Create Venue", overlaid on a blurred background of a concert crowd.

**Venue Details**

This screenshot shows a table of venue details:

RoomId	Building	RoomNo
102	SJT	202
103	SJT	203
104	SJT	204
101	SJT	201

**Create Venue**

This screenshot shows a form for creating a new venue:

**Create Venue**

Form fields:

- Room Id:
- Building:
- Room No:
- Submit** button

**Club Details**

ClubName	Faculty	Description	ContactPerson	ContactNumber
Data	Pradeepa Technical M Mam	Club	Ishan	9890425802
Database	Pradeepa Technical M Mam	Club	Arjun	9890245802
TechClub	Pradeepa Technical M Mam	Club	Om	9421034070
TechBro	Janai T	Technical Club	Ishan	9876543210

**Add Events details**

!! Welcome All Participants !!

Please Add Event Details.

Event Number:

Date:

Time:

Fees:

Duration:

Description:

The image displays two screenshots of a web application interface, set against a vibrant, blurred background of a concert or event with silhouettes of people and stage lights.

**Participant's Login Page**

This screenshot shows a login interface with a dark, futuristic design. It features a central rounded rectangle containing text and buttons. At the top, it says "Participant's Login Page". Below that, it asks "Create account if you are new user?" and provides a "Participant SignUp" button. In the center, it says "Already a user! Please login" and has a "Participant's Login" button. The background of the interface is black with abstract geometric shapes like hexagons and diamonds in blue and white.

**Transactions Page**

This screenshot shows a transaction details page. At the top, it says "Transaction Details". Below that, it displays a table with three columns: "Event No", "Registration No", and "Payment Date". The data shown is:

Event No	Registration No	Payment Date
2	19BCE002	12 June 2019
2	19BCE001	12 June 2019

**Event Details**

EventNumber	EventDate	EventTime	Fees	Duration	Description	SpeakerName	SpeakerNo	Seats
1	12 Jun 2019	10 AM	150	2 Hr	Coding in C	Mr Rovat	9327621603	150
4	12 Jun 2019	11 AM	120	2 Hr	Graphics	Mr Chandra	9327621603	145

**Participant Login**

Please login using valid username(Reg no) and Password

Participant Login Page

Username should be registration number

Username(Reg no):

Password:

Confirm Password:

Submit

The image displays two screenshots of a web application interface, both set against a vibrant, futuristic background featuring silhouettes of people and glowing purple lights.

**Event Details Page:**

This page shows a table of event details:

EventNumber	EventName	ClubName	Co-ordinator	Co-ordinatorContact	RoomId
3	Graphics Data	Omkar	9876543210		103
1	Coding in C	TechClub	Ishan	9890245890	101
2	Virtual FX	TechBro	Ishan	9890245890	102
4	Graphics in Data	Database	Omkar	9890245890	104

**Profile Page:**

This page contains fields for profile information:

- Registration No:
- Name:
- Contact Number:
- Submit

**!! Welcome All Participants !!**

Logout

First Complete your profile

1. Complete Your Profile

Complete Profile

2. View Events

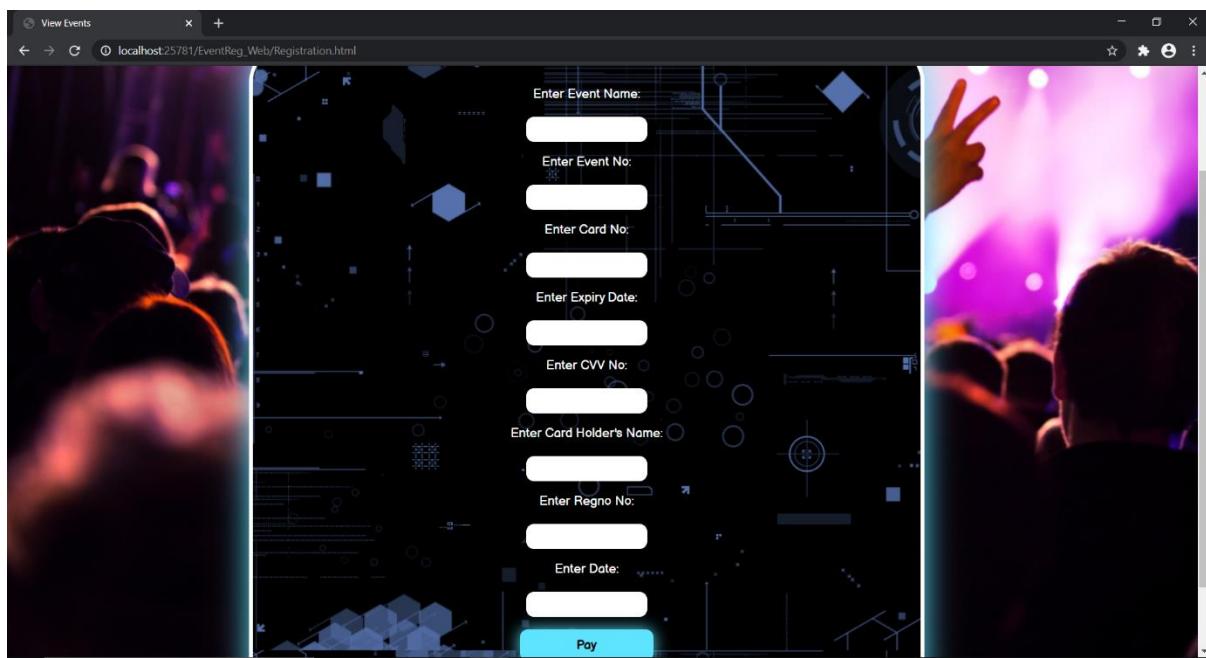
View Events

3. Register For Event

Register For Event

**Event Details**

EventNumber	EventDate	EventTime	Fees	Duration	Description	SpeakerName	SpeakerNo	Seats
1	12 Jun 2019	10 AM	150	2 Hr	Coding in C	Mr Ravat	9327621603	150
4	12 Jun 2019	11 AM	120	2 Hr	Graphics	Mr Chandra	9327621603	145



## Backend Codes Of Webpage In Java (Servlet)

- **AddEvent.java :-**

```
import java.io.*;
import javax.servlet.*;
import javax.servlet.http.*;
import java.sql.*;
public class AddEvent extends HttpServlet{
    public void doPost(HttpServletRequest request, HttpServletResponse response) throws
IOException, ServletException{
        response.setContentType("text/html");
        PrintWriter out=response.getWriter();
        String a1=request.getParameter("EventNo");
        String a2=request.getParameter("EventName");
        String a3=request.getParameter("ClubName");
        String a4=request.getParameter("coordinatorName");
        String a5=request.getParameter("CoordinatorNo");
        String a6=request.getParameter("RoomNo");
        try{
            Class.forName("oracle.jdbc.driver.OracleDriver");
            String conURL = "jdbc:oracle:thin:@localhost:1521:xe";
            String user = "Eventweb";
            String pass = "e1234";
            Connection con = DriverManager.getConnection(conURL, user, pass);
            Statement stmt=con.createStatement();
            String Qs="insert into Event
values('"+a1+"','"+a2+"','"+a3+"','"+a4+"','"+a5+"','"+a6+"') ";
            ResultSet rs=stmt.executeQuery(Qs);
```

```

RequestDispatcher rd=request.getRequestDispatcher("CreateE.html");
rd.include(request, response);

out.println("<br><center><h3> Event Added</h3></center>");
con.close();
}catch(Exception exe){System.out.println("Exception caught");}
}
}

```

- **AddEventDetails.java :-**

```

import java.io.*;
import javax.servlet.*;
import javax.servlet.http.*;
import java.sql.*;
public class AddEventDetails extends HttpServlet{
    public void doPost(HttpServletRequest request, HttpServletResponse response) throws
    IOException, ServletException{
        response.setContentType("text/html");

        PrintWriter out=response.getWriter();
        String a1=request.getParameter("EventNo");
        String a2=request.getParameter("EDate");
        String a3=request.getParameter("ETime");
        String a4=request.getParameter("EFees");
        String a5=request.getParameter("EDuration");
        String a6=request.getParameter("EDescription");
        String a7=request.getParameter("SpeakerName");
        String a8=request.getParameter("SpeakerNo");
        String a9=request.getParameter("Seats");

```

```

try{
    Class.forName("oracle.jdbc.driver.OracleDriver");
    String conURL = "jdbc:oracle:thin:@localhost:1521:xe";
    String user = "Eventweb";
    String pass = "e1234";
    Connection con = DriverManager.getConnection(conURL, user, pass);
    Statement stmt=con.createStatement();

    String Qs="insert into event_details
values(""+a1+"','"+a2+"','"+a3+"','"+a4+"','"+a5+"','"+a6+"','"+a7+"','"+a8+"','"+a9+"') ";
    ResultSet rs=stmt.executeQuery(Qs);

    RequestDispatcher rd=request.getRequestDispatcher("AddEventDetails.html");
    rd.include(request, response);

    out.println("<br><center><h3> Event Details Added</h3></center>");
}catch(Exception exe){System.out.println("Exception caught");}
}

```

- **CreateClub.java :-**

```

import java.io.*;
import javax.servlet.*;
import javax.servlet.http.*;
import java.sql.*;
public class CreateClub extends HttpServlet{
    public void doPost(HttpServletRequest request, HttpServletResponse response) throws
IOException, ServletException{
        response.setContentType("text/html");
        PrintWriter out=response.getWriter();

```

```

String a1=request.getParameter("ClubName");
String a2=request.getParameter("Faculty");
String a3=request.getParameter("CDescription");
String a4=request.getParameter("Ccontactp");
String a5=request.getParameter("ContactNo");

try{
    Class.forName("oracle.jdbc.driver.OracleDriver");
    String conURL = "jdbc:oracle:thin:@localhost:1521:xe";
    String user = "Eventweb";
    String pass = "e1234";
    Connection con = DriverManager.getConnection(conURL, user, pass);
    Statement stmt=con.createStatement();

    String Qs="insert into Club values('"+a1+"','"+a2+"','"+a3+"','"+a4+"','"+a5+"') ";
    ResultSet rs=stmt.executeQuery(Qs);

    RequestDispatcher rd=request.getRequestDispatcher("CreateC.html");
    rd.include(request, response);

    out.println("<br><center><h3> Club Added</h3></center>");
}catch(Exception exe){System.out.println("Exception caught");}
}
}

```

- **CreateVenue.java :-**

```

import java.io.*;
import javax.servlet.*;
import javax.servlet.http.*;
import java.sql.*;

```

```

public class CreateVenue extends HttpServlet{
    public void doPost(HttpServletRequest request, HttpServletResponse response) throws
IOException, ServletException{
        response.setContentType("text/html");

        PrintWriter out=response.getWriter();
        String a1=request.getParameter("RoomId");
        String a2=request.getParameter("Building");
        String a3=request.getParameter("Room No");

        try{
            Class.forName("oracle.jdbc.driver.OracleDriver");
            String conURL = "jdbc:oracle:thin:@localhost:1521:xe";
            String user = "Eventweb";
            String pass = "e1234";
            Connection con = DriverManager.getConnection(conURL, user, pass);
            Statement stmt=con.createStatement();

            String Qs="insert into venue values('"+a1+"','"+a2+"','"+a3+"') ";
            ResultSet rs=stmt.executeQuery(Qs);

            RequestDispatcher rd=request.getRequestDispatcher("CreateV.html");
            rd.include(request, response);

            out.println("<br><center><h3> Venue Added</h3></center>");
            con.close();
        }catch(Exception exe){System.out.println("Exception caught");}
    }
}

```

- **LoginDao.java :-**

```
import java.sql.*;  
  
public class LoginDao {  
    public static boolean validate(String User_name, String Pass_word){  
        boolean status=false;  
        try{  
            Class.forName("oracle.jdbc.driver.OracleDriver");  
            Connection con=DriverManager.getConnection(  
                "jdbc:oracle:thin:@localhost:1521:xe", "plogin", "p1234");  
  
            PreparedStatement ps=con.prepareStatement(  
                "select * from plogindeatils where User_name=? and Pass_word=?");  
            ps.setString(1,User_name);  
            ps.setString(2,Pass_word);  
  
            ResultSet rs=ps.executeQuery();  
            status=rs.next();  
  
        }catch(Exception e){System.out.println(e);}  
        return status;  
    } }
```

- **Profile.java :-**

```
import java.io.*;  
import javax.servlet.*;  
import javax.servlet.http.*;  
import java.sql.*;
```

```

public class Profile extends HttpServlet{
    public void doPost(HttpServletRequest request, HttpServletResponse response) throws
IOException, ServletException{

        response.setContentType("text/html");

        PrintWriter out=response.getWriter();
        String a1=request.getParameter("Reg");
        String a2=request.getParameter("Name");
        String a3=request.getParameter("ContactNo");
        try{
            Class.forName("oracle.jdbc.driver.OracleDriver");
            String conURL = "jdbc:oracle:thin:@localhost:1521:xe";
            String user = "Transweb";
            String pass = "t1234";
            Connection con = DriverManager.getConnection(conURL, user, pass);
            Statement stmt=con.createStatement();

            String Qs="insert into Student values('"+a1+"','"+a2+"','"+a3+"') ";
            ResultSet rs=stmt.executeQuery(Qs);

            RequestDispatcher rd=request.getRequestDispatcher("Profile.html");
            rd.include(request, response);

            out.println("<br><center><h3> Profile Added</h3></center>");
        }catch(Exception exe){System.out.println("Exception caught");}
    }
}

```

- **Register.java :-**

```
import java.io.*;
import javax.servlet.*;
import javax.servlet.http.*;
import java.sql.*;
public class Register extends HttpServlet{
    public void doPost(HttpServletRequest request, HttpServletResponse response) throws
IOException, ServletException{
        response.setContentType("text/html");
        PrintWriter out=response.getWriter();
        String a1=request.getParameter("cardno");
        String a2=request.getParameter("edate");
        String a3=request.getParameter("cvv");
        String a4=request.getParameter("cname");
        String a5=request.getParameter("reg");
        String a6=request.getParameter("enum");
        String a7=request.getParameter("ename");
        String a8=request.getParameter("date");
        try{
            Class.forName("oracle.jdbc.driver.OracleDriver");
            String conURL = "jdbc:oracle:thin:@localhost:1521:xe";
            String user = "Transweb";
```

```

String pass = "t1234";
Connection con = DriverManager.getConnection(conURL, user, pass);
Statement stmt=con.createStatement();

String Qs="insert into card values(""+a1+"','"+a2+"','"+a3+"','"+a4+"','"+a5+"') ";
ResultSet rs=stmt.executeQuery(Qs);

String Qt="insert into transaction_history values(""+a6+"','"+a5+"','"+a8+"') ";
ResultSet rp=stmt.executeQuery(Qt);

con.commit();
con.close();

RequestDispatcher rd=request.getRequestDispatcher("Registration.html");
rd.include(request, response);

out.println("<center><h2>Registered For Event: "+a7);
out.println("<br>");
out.println("<center><h2>Paid succefully </h2></center>");
out.println("<br><center><a href=\"ParticipantEvent.html\">Go back</a></center>");
}catch(Exception exe){System.out.println("Exception caught");}

}

```

- **StoreP.java :-**

```
import java.io.*;
import javax.servlet.*;
import javax.servlet.http.*;
import java.sql.*;
public class StoreP extends HttpServlet{
    public void doPost(HttpServletRequest request, HttpServletResponse response) throws
IOException, ServletException{
        response.setContentType("text/html");
        PrintWriter out=response.getWriter();
        String up=request.getParameter("Pusername");
        String pp=request.getParameter("Ppassword");
        String cp=request.getParameter("Cpassword");
        String a1=up;
        String a2=pp;
        String a3=cp;
        if(a2.equals(a3)){
            try{
                Class.forName("oracle.jdbc.driver.OracleDriver");
                String conURL = "jdbc:oracle:thin:@localhost:1521:xe";
                String user = "plogin";
                String pass = "p1234";
                Connection con = DriverManager.getConnection(conURL, user, pass);
                Statement stmt=con.createStatement();
            }
        }
    }
}
```

```

String Qs="insert into plogindeatils values('"+a1+"','"+a2+"') ";
ResultSet rs=stmt.executeQuery(Qs);

RequestDispatcher rd=request.getRequestDispatcher("Plogin.html");
rd.forward(request, response);
}catch(Exception exe){System.out.println("Exception caught");}

}

else
{
    out.println("<center><h1>!! Please Enter Password And Confirm Password Same
    !!</h1><center>");
    RequestDispatcher rd=request.getRequestDispatcher("Psignup.html");
    rd.include(request, response);
}

}

}

```

- **TraH.java :-**

```

import java.io.*;
import javax.servlet.*;
import javax.servlet.http.*;
import java.sql.*;
public class TraH extends HttpServlet{
    public void doGet(HttpServletRequest request, HttpServletResponse response) throws
    IOException, ServletException{

```

```

response.setContentType("text/html");
PrintWriter out = response.getWriter();
out.println("<!DOCTYPE html>");
out.println("<html>");
out.println("<head>");
out.println("<title> Transactions Page</title>");
out.println("<link rel=\"stylesheet\" href=\"total.css\">");
out.println("<link href=\"https://fonts.googleapis.com/css2?family=Balsamiq+Sans&display=swap\" rel=\"stylesheet\">");
out.println("</head>");
out.println("<body>");

try{
    Class.forName("oracle.jdbc.driver.OracleDriver");
    String conURL = "jdbc:oracle:thin:@localhost:1521:xe";
    String user = "Transweb";
    String pass = "t1234";
    Connection con = DriverManager.getConnection(conURL, user, pass);

    Statement stmt = con.createStatement();
    ResultSet rs = stmt.executeQuery("select * from transaction_history");

    out.println("<center><h1> Transaction Details </h1> </center> ");

    out.println("<div>");

    out.println("<center>");
    out.println("<table border=1 width=50% height=50%>");


```

```

        out.println("<tr><th>Event No </th><th> Registration No</th><th>Payment
Date</th>");

        while (rs.next())
        {

            String en = rs.getString("enum");
            String re = rs.getString("regno");
            String pd = rs.getString("hdate");

            out.println("<tr><td>" + en + "</td><td>" + re + "</td><td>" + pd
+ "</td></tr>");
        }

        con.commit();
        con.close();
        out.println("</table>");
        out.println("</h3></center>");
        out.println("</div>");
        out.print("</body>");
        out.print("</html>");

    }

    catch(Exception exe){System.out.println("Exception caught");}
}

}

```

- **VaPa.java :-**

```
import java.io.IOException;
import java.io.PrintWriter;

import javax.servlet.RequestDispatcher;
import javax.servlet.ServletException;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;

public class VaPa extends HttpServlet {
    public void doPost(HttpServletRequest request, HttpServletResponse response)
        throws ServletException, IOException {
        response.setContentType("text/html");
        PrintWriter out = response.getWriter();

        String n=request.getParameter("Pausername");
        String p=request.getParameter("Papassword");

        if(LoginDao.validate(n, p)){
            RequestDispatcher rd=request.getRequestDispatcher("ParticipantEvent.html");
            rd.forward(request,response);
        }
        else{
            out.print("Sorry username or password incorrect");
            RequestDispatcher rd=request.getRequestDispatcher("Plogin.html");
            rd.include(request,response);
        }
    }
}
```

```
    out.close();
}
}
```

- **ValidateAd.java :-**

```
import java.io.*;
import javax.servlet.*;
import javax.servlet.http.*;
public class ValidateAd extends HttpServlet{
    public void doPost(HttpServletRequest request, HttpServletResponse response) throws
IOException, ServletException{
        PrintWriter out=response.getWriter();
        String un=request.getParameter("Ausername");
        String pw=request.getParameter("Apassword");
        response.setContentType("text/html");

        String u1="19BCE2250";
        String p1="1234";

        String u2="19BCE2249";
        String p2="5678";

        String u3="19BCE0439";
        String p3="ABCD";

        String u4="19BCE0167";
        String p4="abcd";

        if(pw.equals(p1)&&(un.equals(u1))){
            RequestDispatcher rd=request.getRequestDispatcher("AdminEvent.html");

```

```

rd.forward(request, response);
}

else if(pw.equals(p2)&&un.equals(u2)){
    RequestDispatcher rd=request.getRequestDispatcher("AdminEvent.html");
    rd.forward(request, response);
}

else if(pw.equals(p3)&&un.equals(u3)){
    RequestDispatcher rd=request.getRequestDispatcher("AdminEvent.html");
    rd.forward(request, response);
}

else if(pw.equals(p4)&&un.equals(u4)){
    RequestDispatcher rd=request.getRequestDispatcher("AdminEvent.html");
    rd.forward(request, response);
}

}

else
{
    out.println("<center><h1>!! Please Enter Valid Username & Password
    !!</h1><center>");
    RequestDispatcher rd=request.getRequestDispatcher("Alogin.html");
    rd.include(request, response);
}

}

```

- **ViewClub1.java :-**

```
import java.io.*;
```

```

import javax.servlet.*;
import javax.servlet.http.*;
import java.sql.*;
public class ViewClub1 extends HttpServlet{
    public void doGet(HttpServletRequest request, HttpServletResponse response) throws
IOException, ServletException{
        response.setContentType("text/html");
        PrintWriter out = response.getWriter();
        out.println("<!DOCTYPE html>");
        out.println("<html>");
        out.println("<head>");
        out.println("<title>Club Details Page</title>");
        out.println("<link rel=\"stylesheet\" href=\"total.css\">");
        out.println("<link href=\"https://fonts.googleapis.com/css2?family=Balsamiq+Sans&display=swap\" rel=\"stylesheet\">");
        out.println("</head>");
        out.println("<body>");

        try{
            Class.forName("oracle.jdbc.driver.OracleDriver");
            String conURL = "jdbc:oracle:thin:@localhost:1521:xe";
            String user = "Eventweb";
            String pass = "e1234";
            Connection con = DriverManager.getConnection(conURL, user, pass);

            Statement stmt = con.createStatement();
            ResultSet rs = stmt.executeQuery("select * from Club");

            out.println("<center><h1> Club Details </h1> </center> ");

```

```

        out.println("<br><br>");

        out.println("<div>");
        out.println("<center>");

        out.println("<table border=1 width=50% height=50%>");

out.println("<tr><th>ClubName</th><th>Faculty</th><th>Description</th><th>ContactPer
son</th><th>ContactNumber</th>");

        while (rs.next())
        {

            String cn = rs.getString("cname");
            String fc = rs.getString("fc");
            String desc = rs.getString("bdesc");
            String conta=rs.getString("contper");
            String conn=rs.getString("connnum");
            out.println("<tr><td>" + cn + "</td><td>" + fc + "</td><td>" + desc +
" </td><td>" + conta + "</td><td>" + conn + "</td></tr>");

        }

        con.commit();
        con.close();
        out.println("</table>");
        out.println("</center>");
        out.println("</div>");
        out.print("</body>");
        out.print("</html>");

    }
}

```

```
        catch(Exception exe){System.out.println("Exception caught");}
    }
}
```

- **ViewEvent1.java :-**

```
import java.io.*;
import javax.servlet.*;
import javax.servlet.http.*;
import java.sql.*;
public class ViewEvent1 extends HttpServlet{
    public void doGet(HttpServletRequest request, HttpServletResponse response) throws
IOException, ServletException{
        response.setContentType("text/html");
        PrintWriter out = response.getWriter();
        out.println("<html>");
        out.println("<head>");
        out.println("<title>Event Page</title>");
        out.println("<link rel=\"stylesheet\" href=\"total.css\">");
        out.println("<link
href=\"https://fonts.googleapis.com/css2?family=Balsamiq+Sans&display=swap\""
rel=\"stylesheet\">");
        out.println("</head>");
        out.println("<body>");
        try{
            Class.forName("oracle.jdbc.driver.OracleDriver");
            String conURL = "jdbc:oracle:thin:@localhost:1521:xe";
            String user = "Eventweb";
            String pass = "e1234";
            Connection con = DriverManager.getConnection(conURL, user, pass);
            response.setContentType("text/html");
            Statement stmt1 = con.createStatement();
            ResultSet rp = stmt1.executeQuery("select * from Event");
        }
    }
}
```

```

        out.println("<center><h1>Event Details</h1></center>");
        out.println("<br>");
        out.println("<div>");

        out.println("<center>");
        out.println("<table border=1 width=50% height=50%>");

out.println("<tr><th>EventNumber</th><th>EVentName</th><th>ClubName</th><th>Co-
ordinator</th><th>Co-ordinatorContact</th><th>RoomId</th>");

        while (rp.next())
        {
            String n = rp.getString("enum");
            String nm = rp.getString("ename");
            String cn = rp.getString("cname");
            String co = rp.getString("coord");
            String cono = rp.getString("coordnum");
            String roomid=rp.getString("roomid");

            out.println("<tr><td>" + n + "</td><td>" + nm + "</td><td>" + cn +
"("</td><td>" + co + "</td><td>" + cono + "</td><td>" + roomid + "</td></tr>");

        }
        con.commit();
        con.close();
        out.println("</table>");
        out.println("</center>");
        out.print("</body>");
        out.print("</html>");

    }

catch(Exception exe){System.out.println("Exception caught");}
}

```

```
}
```

## ViewEventD.java :-

```
import java.io.*;
import javax.servlet.*;
import javax.servlet.http.*;
import java.sql.*;
public class ViewEventD extends HttpServlet{
    public void doGet(HttpServletRequest request, HttpServletResponse response) throws
IOException, ServletException{
        response.setContentType("text/html");
        PrintWriter out = response.getWriter();

        out.println("<html>");
        out.println("<head>");
        out.println("<title>Event Details Page</title>");
        out.println("<link rel=\"stylesheet\" href=\"total.css\">");
        out.println("<link
href=\"https://fonts.googleapis.com/css2?family=Balsamiq+Sans&display=swap\""
rel=\"stylesheet\">");
        out.println("</head>");
        out.println("<body>");

        try{
            Class.forName("oracle.jdbc.driver.OracleDriver");
            String conURL = "jdbc:oracle:thin:@localhost:1521:xe";
            String user = "Eventweb";
            String pass = "e1234";
            Connection con = DriverManager.getConnection(conURL, user, pass);
            response.setContentType("text/html");
            Statement stmt = con.createStatement();
            ResultSet rs = stmt.executeQuery("select * from event_details");
            out.println("<center><h1> Event Details </h1> </center> ");
        }
    }
}
```

```

        out.println("<br><br>");

        out.println("<div>");
        out.println("<center>");

        out.println("<table border=1 width=50% height=50%>");

out.println("<tr><th>EventNumber</th><th>EventDate</th><th>EventTime</th><th>Fees</th><th>Duration</th><th>Description</th><th>SpeakerName</th><th>SpeakerNo</th><th>Seats</th>");

        while (rs.next())
        {
            String n = rs.getString("enum");
            String da = rs.getString("edate");
            String ti = rs.getString("time");
            String fe = rs.getString("fees");
            String dur=rs.getString("dur");
            String des=rs.getString("descr");
            String sn=rs.getString("sname");
            String snu = rs.getString("snum");
            String sea=rs.getString("seats");

            out.println("<tr><td>" + n + "</td><td>" + da + "</td><td>" + ti +
" </td><td>" + fe + "</td><td>" + dur + "</td><td>" + des + "</td><td>" + sn + "</td><td>" + snu + "</td><td>" + sea + "</td></tr>");

        }

        con.commit();
        con.close();
        out.println("</table>");
        out.println("</center>");
        out.print("</body>");
        out.print("</html>");

    }

    catch(Exception exe){System.out.println("Exception caught");}
}

```

```
}
```

- **ViewVenue1.java :-**

```
import java.io.*;  
import javax.servlet.*;  
import javax.servlet.http.*;  
import java.sql.*;  
  
public class ViewVenue1 extends HttpServlet{  
  
    public void doGet(HttpServletRequest request, HttpServletResponse response) throws  
IOException, ServletException{  
  
        response.setContentType("text/html");  
        PrintWriter out = response.getWriter();  
  
        out.println("<!DOCTYPE html>");  
        out.println("<html>");  
        out.println("<head>");  
        out.println("<title>Venue Page</title>");  
        out.println("<link rel=\"stylesheet\" href=\"total.css\">");  
        out.println("<link  
href=\"https://fonts.googleapis.com/css2?family=Balsamiq+Sans&display=swap\"  
rel=\"stylesheet\">");  
        out.println("</head>");  
        out.println("<body>");  
        try{  
            Class.forName("oracle.jdbc.driver.OracleDriver");  
            String conURL = "jdbc:oracle:thin:@localhost:1521:xe";  
            String user = "Eventweb";  
            String pass = "e1234";  
            Connection con = DriverManager.getConnection(conURL, user, pass);  
            Statement stmt=con.createStatement();  
            response.setContentType("text/html");
```

```

ResultSet rp = stmt.executeQuery("select * from Venue");
out.println("<center><h1>Venue Details</h1></center>");
out.println("<br><br>");
out.println("<div>");
out.println("<center>");
out.println("<table border=1 width=50% height=50%>");
out.println("<tr><th>RoomId</th><th>Building</th><th>RoomNo</th>");
while (rp.next())
{
    String roomid=rp.getString("roomid");
    String bd = rp.getString("building");
    String roomno= rp.getString("roomno");
    out.println("<tr><td>" + roomid + "</td><td>" + bd+"</td>
<td>" + roomno + "</td></tr> ");
}
out.println("</table>");

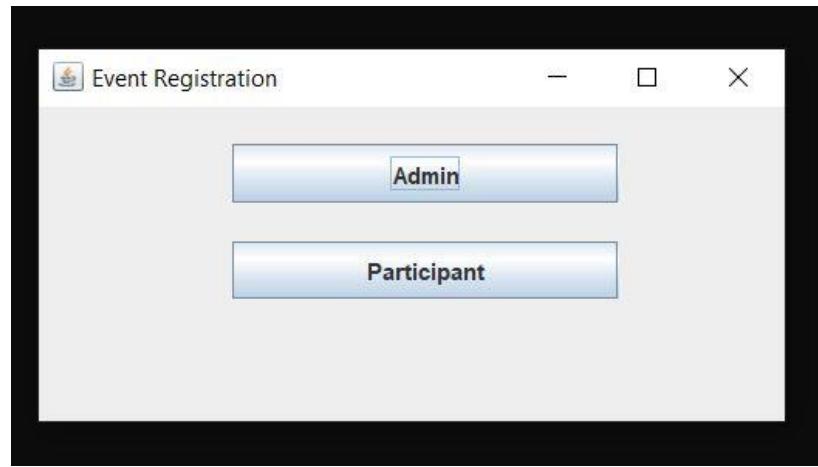
out.println("</body>");
out.println("</html>");

con.close();
}

catch(Exception exe){System.out.println("Exception caught");}
}
}

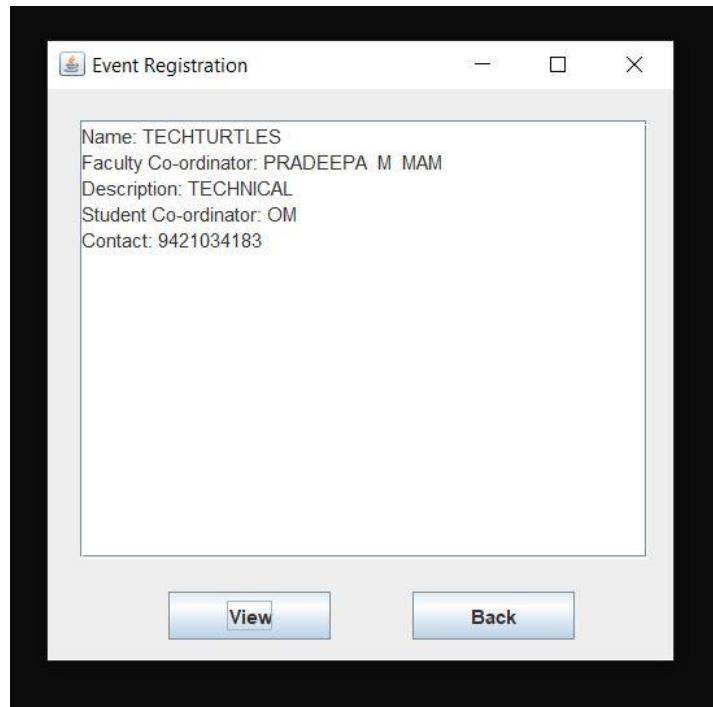
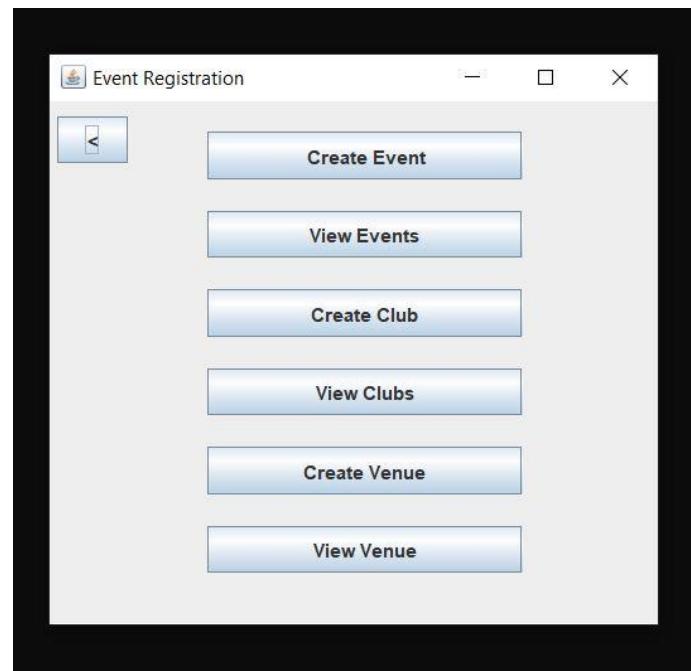
```

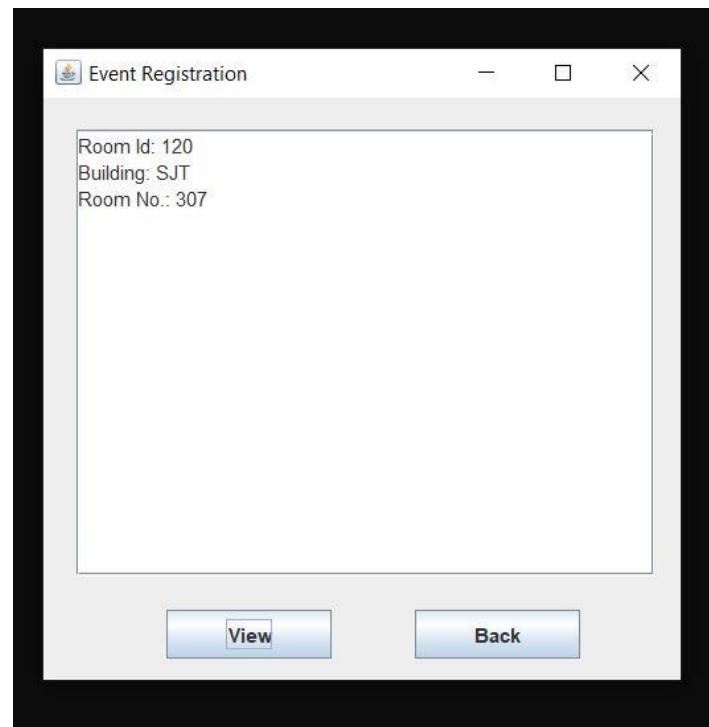
## Java Application Interface :



The screenshot shows a Java Swing application window titled "Event Registration". The content area contains five horizontal rows of form fields. Each row consists of a label on the left followed by a text input field on the right. The labels are: "Name", "Faculty Coord.", "Description", "Contact Person", and "Contact Number". At the bottom of the form are two blue-highlighted rectangular buttons: "Submit" on the left and "Back" on the right.

Name	<input type="text"/>
Faculty Coord.	<input type="text"/>
Description	<input type="text"/>
Contact Person	<input type="text"/>
Contact Number	<input type="text"/>





The screenshot shows a window titled "Event Registration". It contains three input fields labeled "Room Id", "Building", and "Room No.", each with an associated text input box. At the bottom of the window are two buttons: "Submit" on the left and "Back" on the right.

 Event Registration

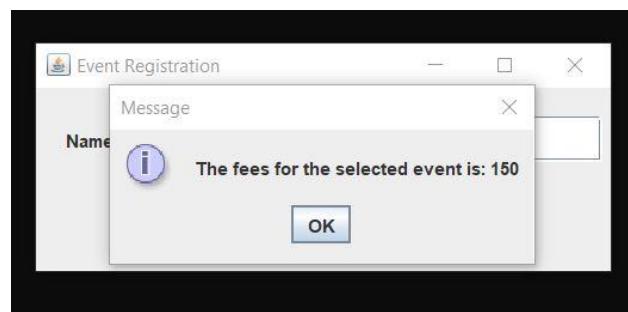
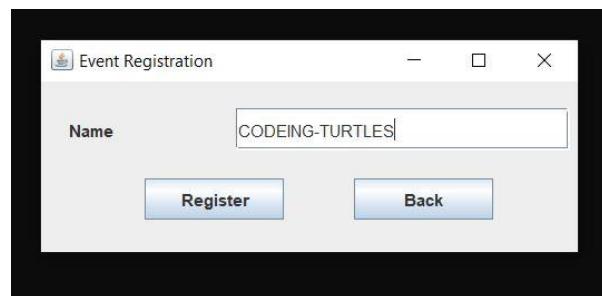
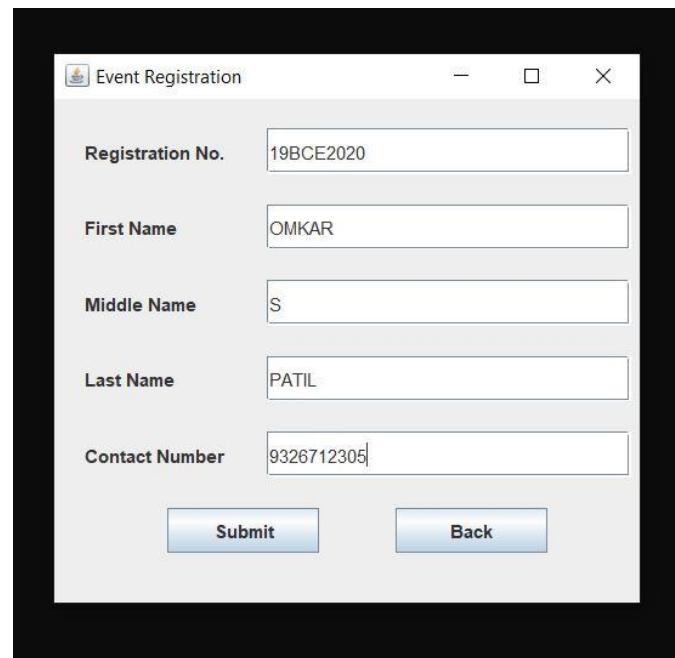
<b>Event No.</b>	101
<b>Date</b>	2019-10-12
<b>Time</b>	10 AM
<b>Fees</b>	150
<b>Duration</b>	2 HR
<b>Description</b>	CODING IN C++
<b>Speaker Name</b>	MR. RAJ C
<b>Speaker Number</b>	9890245801

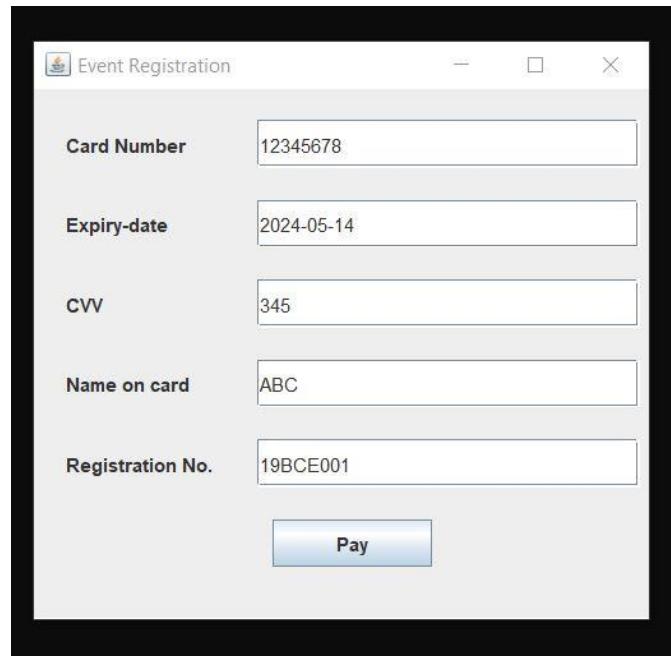
**Submit**      **Back**

 Event Registration

No.: 101
Name: CODEING-TURTLES
Club Name: TECHTURTLES
Coordinator Name: ISHAN
Co-ordinator Number: 9421043707
Room ID: 120
Date: 2020-10-12 00:00:00.0
Time: 10 AM
Fees: 150
Duration: 5 HR
Description: CODING IN C++
Speaker Name: MR. RAG C
Speaker Number: 9890245801
Seats: 120

**View**      **Back**





## **Results and Discussion [Final Conclusion]**

The event management database and its consequent interrelation with the frontend and backend are shown in the previous slides.

From the requirement analysis we get a good idea of what to and what not to include in the ERD.

Simultaneously we decide the data type for each attribute of all such entities and use normal forms to make it functional and redundancy-free.

The frontend and backend as well as the relational schema make our DBMS model using Distributed DBMS a workable prototype ready to be deployed in the modern world.

## **REFERANCES**

- An Overview of Distributed Databases Parul Tomar and Megha2
- Sciencedirect.com/topics/computer-science/distributed-database-management-system
- Fundamentals of Distributed Database System ,Sagar Bajpayee, SwapnilShukla
- IOSR Journal of Computer Engineering (IOSR-JCE) e-ISSN: 2278-0661,p-ISSN: 2278-8727 PP 15-18 [www.iosrjournals.org](http://www.iosrjournals.org).
- International Journal of Trend in Research and Development, Volume2(5), ISSN 239333 [www.ijtrd.com](http://www.ijtrd.com)
- IJCSMS International Journal of Computer Science and ManagementStudies, Vol. 11, Issue 02, Aug 2011 ISSN (Online): 2231-5268[www.ijcsmms.com](http://www.ijcsmms.com)

## **APPENDICES**

### **A**

#### **HTML Frontend Codes for Webpage**

- **AddEventDetails :**

```
<!DOCTYPE html>

<!--

To change this license header, choose License Headers in Project Properties.
To change this template file, choose Tools | Templates
and open the template in the editor.

-->

<html><br>
<head>
  <link rel="stylesheet" href="total.css">

  <link
  href="https://fonts.googleapis.com/css2?family=Balsamiq+Sans&display=swap"
  " rel="stylesheet">

  <title> Add Events details </title>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <h2 id="common" style="text-align:center">!! Welcome All Participants
  !! </h1>
</head>
```

```
<marquee scrollamount="15" width="100%" direction="left"
height="100px">

    <h2> Please Add Event Details.</h2>

</marquee>

<body>
<div>

    <center>

<form method="post" action="AddEventDetails">

    Event Number:<br><br>
    <input type="text" id="text" name="EventNo"><br>

    Date:<br><br>
    <input type="text" id="text" name="EDate"><br><br>

    Time:<br><br>
    <input type="text" id="text" name="ETime"><br><br>

    Fees:<br><br>
    <input type="text" id="text" name="EFees"><br><br>

    Duration:<br><br>
    <input type="text" id="text" name="EDuration"><br><br>

    Description:<br><br>
    <input type="text" id="text" name="EDescription"><br><br>
```

```

Speaker Name:<br><br>
<input type="text" id="text" name="SpeakerName"><br><br>

Speaker Number:<br><br>
<input type="text" id="text" name="SpeakerNo"><br><br>

Seats:<br><br>
<input type="text" id="text" name="Seats"><br><br>

<input type="submit" id="sub" value="Submit">
</form>
</center>
</body>
</html>

```

## • AdminEvent :

```

<!DOCTYPE html>
<html><br>
<head>
  <link rel="stylesheet" href="total.css">

  <link href="https://fonts.googleapis.com/css2?family=Balsamiq+Sans&display=swap"
rel="stylesheet">
  <title>Admin's Page For Event Management </title>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">

```

```
<h2 id="common" style="text-align:center">!! Welcome Admin !! </h1>
</head>
<body>
<span style="float:right">
<a href="Alogin.html"><button>Logout</button></a>
</span><br><br><p></p> <br>
<div>

<center>

</center>
<br>
<center><h2> For Event Details </h2>
<br>
<a href="EventDetails.html"><button>Event Details</button></a>
</center>
<br><br><br>
<center>

</center>
<br>
<center>
<h3> For Transaction Details </h3>
<br>
<a href="TransactionDetails.html"><button>Transaction Details</button></a>
</center>
</body>
</html>
```

- **Alogin :**

```
<!-- ADMIN LOGIN- -->

<!DOCTYPE html>
<!--
To change this license header, choose License Headers in Project Properties.
To change this template file, choose Tools | Templates
and open the template in the editor.
-->
<html>
<head>
<br> <br> <br>
<meta charset="UTF-8">
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<h2 style="text-align:center" id="common">Please login using valid username(Reg no)
and Password </h2>
<link rel="stylesheet" href="total.css">

<link href="https://fonts.googleapis.com/css2?family=Balsamiq+Sans&display=swap"
rel="stylesheet">
</head>
<body>
<br> <br>
<div >
<center><h2 id="common"> Admin Login Page </h2></center>
<br><br>

<center>
<form method="post" action="ValidateAd">
Username(Reg no):<br><br>
```

```

<input type="text" id="text" name="Ausername" placeholder="Username"><br><p>
Password:<br><br>

<input type="password" id="text" name="Apassword"
placeholder="Password"><br><br><p>

<input type="submit" id="sub" value="Submit">
</form>
</center>
</div>
</body>
</html>

```

## • CreateC :

```

<!-- Create club : -->
<!DOCTYPE html>

<html><br>
<head>
<link rel="stylesheet" href="total.css">

<link href="https://fonts.googleapis.com/css2?family=Balsamiq+Sans&display=swap"
rel="stylesheet">

<title>Create Club </title>
<meta charset="UTF-8">
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<h2 id="common" style="text-align:center"> Create Club Page </h2>
</head>
<body>
<div>
<center>

```

```


</center>

<center>
<form method="post" action="CreateClub">

Club Name:<br><br>
<input type="text" id="text" name="ClubName"><br><p></p>

Faculty Co-ordinator:<br><br>
<input id="text" type="text" name="Faculty"><br><br>

Description:<br><br>
<input id="text" type="text" name="CDescription"><br><br>

Contact Person:<br><br>
<input id="text" type="text" name="Ccontactp"><br><br>

Contact Number:<br><br>
<input id="text" type="text" name="ContactNo"><br><br>

<input type="submit" id="sub" value="Submit">
</form>
</center>
</div>
</body>
</html>

```

## • CreateE :

```
<!-- Create event:- -->
```

```
.<!DOCTYPE html>
<html>
  <head>
    <link rel="stylesheet" href="total.css">

    <link href="https://fonts.googleapis.com/css2?family=Balsamiq+Sans&display=swap"
rel="stylesheet">

    <title>Create Event</title>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <h2 id="common" style="text-align:center"> Create Event </h2>

  </head>
  <body>
    <div>
      <center>

<form method="post" action="AddEvent">

  Event Number:<br><br>
  <input id="text" type="text" name="EventNo"><br><br>

  Event Name:<br><br>
  <input id="text" type="text" name="EventName"><br><br>

  Club Name:<br><br>
  <input id="text" type="text" name="ClubName"><br><br>

  Co-ordinator Name:<br><br>
  <input id="text" type="text" name="coordinatorName"><br><br>

  Co-ordinator Number:<br><br>
  <input id="text" type="text" name="CoordinatorNo"><br><br>
```

```

Room Number:<br><br>
<input id="text" type="text" name="RoomNo"><br><br><p></p>
<input type="submit" id="sub" value="Submit">

</form>
</center> </div>
</body>
</html>

```

## • CreateV :

```

<!-- create venue:- -->
<!DOCTYPE html>
<html><br>
<head>
    <link rel="stylesheet" href="total.css">

    <link href="https://fonts.googleapis.com/css2?family=Balsamiq+Sans&display=swap"
rel="stylesheet">

    <title> Create Venue </title><p></p>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <h2 style="text-align:center" id="common"> Create Venue </h2>
</head>
<body>
    <div>
        <center>
            
        </center>
        <center>
            <form method="post" action="CreateVenue">

```

```

<h2>Room Id:</h2>
<input id="text" type="text" name="RoomId"><br>

<h2>Building:</h2>
<input id="text" type="text" name="Building"><br>

<h2>Room No:</h2>
<input id="text" type="text" name="Room No"><br><br><p></p>

<input type="submit" id="sub" value="Submit">
</form>
</center></div>
</body>
</html>

```

## • EventDetails :

```

<!DOCTYPE html>
<html><br>
<head>
    <title> Event Details </title>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <h1 style="text-align:center">Event Management </h1>
    <link rel="stylesheet" href="total.css">

    <link href="https://fonts.googleapis.com/css2?family=Balsamiq+Sans&display=swap"
    rel="stylesheet">
</head>
<body>
<center>

```

<center> <h2 style="color: black"> First Enter Club details then venue and then add Event details </h2></center>

<h2 style="color: black"> 1. Create Event Link </h2>

<a href="CreateE.html"><button type="button" id="button">Create Event</button></a>

<br><br>

<h2 style="color: black"> 2. View Event Link </h2>

<a href="ViewEvent1"><button type="button" id="button">View Event</button></a>

<br><br>

<h2> 3. Create Club Link </h2>

<a href="CreateC.html"><button type="button" id="button">Create Club</button></a>

<br><br>

<h2> 4. View Club Link </h2>

<a href="ViewClub1"><button type="button" id="button">View Club</button></a>

<br><br>

<h2> 5. Create Venue </h2>

<a href="CreateV.html"><button type="button" id="button">Create Venue</button></a>

<br><br>

<h2> 6. View Venue </h2>

<a href="ViewVenue1"><button type="button" id="button">View Venue</button></a>

<br><br>

<br><br>

<center>

```



</center>

<br>

<center>

<h2> 7. Event Details</h2>

<a href="AddEventDetails.html"><button>Add Event Details </button></a>

<br><br>

<h2>8. View Event Details</h2>

<a href="ViewEventD"><button>View Event Details</button></a>

</center>

</center>

</body>

</html>

```

## • Index :

```

<!-- Login Page -->

<!DOCTYPE html>

<html >

<head>

<title>Login Page</title>

<meta charset="UTF-8">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<h1 style="text-align:center">Welcome To Event Portal </h1>

<link rel="stylesheet" href="total.css">

```

```
<link href="https://fonts.googleapis.com/css2?family=Balsamiq+Sans&display=swap"
rel="stylesheet">

</head>

<marquee scrollamount="10" width="100%" direction="left" height="100px">

<h2> Please choose the below links to login to your account.</h2>

</marquee>

<body>

<div>

<br>

<center>



</center>

<br><br><br>

<center>

<h3>

Admin Login: </h3>

<p>

<a href="Alogin.html"><button> Admin Login </button> </a>

</h4></center>

<br><br><br>

<center>



</center>

<center>

<h3> For Participants Of Events </h3>

<br>

<a href="Parlogin.html"><button>Participant's Login</button></a>

</center></div>

</body>

</html>
```

## • Parlogin :

```
<!DOCTYPE html>
<!--
To change this license header, choose License Headers in Project Properties.
To change this template file, choose Tools | Templates
and open the template in the editor.
-->
<html><br>
<head>
<link rel="stylesheet" href="total.css">

<link href="https://fonts.googleapis.com/css2?family=Balsamiq+Sans&display=swap"
rel="stylesheet">

<title>Participant's Login Page</title>
<meta charset="UTF-8">
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<h2 style="text-align:center" id="common"> Participant's Login
Page</h2><br><p></p><p></p>
</head>
<body>
<div >

<center>
<h3> Create account if you are new user </h3> <br>
<a href="Psignup.html"><button>Participant SignUp </button></a>
</center>

<br><br><br>
<center> <h3>Already a user ! Please login </h3>
<a href="Plogin.html"><button>Participant's Login</button></a>
```

```
</center>
```

```
</div>
```

```
</body>
```

```
</html>
```

## • ParticipantEvent :

```
<!DOCTYPE html>
```

```
<!--
```

To change this license header, choose License Headers in Project Properties.

To change this template file, choose Tools | Templates

and open the template in the editor.

```
-->
```

```
<html><br>
```

```
<head>
```

```
  <link rel="stylesheet" href="total.css">
```

```
  <link href="https://fonts.googleapis.com/css2?family=Balsamiq+Sans&display=swap" rel="stylesheet">
```

```
  <title>Page For Event Registration </title>
```

```
  <meta charset="UTF-8">
```

```
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
```

```
  <h2 id="common" style="text-align:center">!! Welcome All Participants !! </h1>
```

```
</head>
```

```
<body>
```

```
  <span style="float:right">
```

```
    <a href="Plogin.html"><button>Logout</button></a>
```

```
  </span><br><br><p></p> <br>
```

```

<br>
<br>
<center>
    <h3 style="color: black"> First Complete your profile </h3>
    <h3 style="color: black"> 1.Complete Your Profile </h3>
    <a href="Profile.html"><button type="button" id="button"> Complete Profile
</button></a>
    <br><br>
    <br><br>
    <h3> 2.View Events </h3>
    <a href="ViewEvents.html"> <button type="button" id="button">View Events
</button></a>
    <br><br>
    <br><br>
    <h3> 3. Register For Event</h3>
    <a href="Registration.html"> <button type="button" id="button">Register For Event
</button></a>
    <br><br>
</center>
</body>
</html>

```

## • Plogin :

```

<!DOCTYPE html>
<!--
To change this license header, choose License Headers in Project Properties.
To change this template file, choose Tools | Templates
and open the template in the editor.

```

```

-->

<html>

  <head>
    <link rel="stylesheet" href="total.css">

    <link href="https://fonts.googleapis.com/css2?family=Balsamiq+Sans&display=swap"
    rel="stylesheet">

    <title>Participant Login </title>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0"><br>
    <h2 id="common" style="text-align:center">Please login using valid username(Reg no)
    and Password </h2>

  </head><br><br><br><br>

  <body>

  <}>

  <div >

    <center><h2> Participant's Login Page </h2></center>
    <br><br>

    <center>
      <form method="post" action="VaPa">
        Username(Reg no):<br><br>
        <input id="text" style="color: Black" type="text" name="Pusername"><br><p></p>
        Password:<br><br>
        <input id="text" type="password" name="Papassword"><br><br><br>
        <input type="submit" id="sub" value="Submit">
      </form>
    </center>
  </div >

```

```
</center>  
</div>  
</body>  
</html>
```

## • Profile :

```
<!DOCTYPE html>
```

```
<!--
```

To change this license header, choose License Headers in Project Properties.

To change this template file, choose Tools | Templates  
and open the template in the editor.

```
-->
```

```
<html><br>  
<head>  
<link rel="stylesheet" href="total.css">
```

```
<link href="https://fonts.googleapis.com/css2?family=Balsamiq+Sans&display=swap"  
rel="stylesheet">
```

```
<title> Profile </title>  
<meta charset="UTF-8">  
<meta name="viewport" content="width=device-width, initial-scale=1.0">  
<h2 id="common" style="text-align:center"> Profile Page </h2>
```

```
</head>
```

```
<body>
```

```
<div>
```

```
<center>
```

```
<form method="post" action="Profile">
```

```
Registration No:<br><br>
```

```
<input type="text" id="text" name="Reg"><br><p></p>
```

```
Name:<br><br>
```

```

<input id="text" type="text" name="Name"><br><br>

Contact Number:<br><br>
<input id="text" type="text" name="ContactNo"><br><br>

<input type="submit" id="sub" value="Submit">
</form>

<br><br>
</center>
</div>
</body>
</html>

```

## • Psignup :

<!-- Participant sign up:- -->

<!DOCTYPE html>

<!--

To change this license header, choose License Headers in Project Properties.

To change this template file, choose Tools | Templates

and open the template in the editor.

-->

<html>

<head>

<link rel="stylesheet" href="total.css">

<link href="https://fonts.googleapis.com/css2?family=Balsamiq+Sans&display=swap" rel="stylesheet">

<title>Participant Login </title>

```

<meta charset="UTF-8">
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<h2 id="common" style="text-align:center">Please login using valid username(Reg no)
and Password </h2>

</head>
<body>
<div>

<center><h3> Participant Login Page </h3>
<h4> Username should be registration number </h4>
</center>

<center>
<form method="post" action="StoreP">
Username(Reg no):<br><br>
<input id="text" type="text" name="Pusername"><br><p></p>
Password:<br><br>
<input id="text" type="password" name="Ppassword"><br><br>
Confirm Password:<br><br>
<input id="text" type="password" name="Cpassword"><br><br>
<br><br>
<input type="submit" id="sub" value="Submit">
</form>
</center>  </div>
</body>
</html>

```

## • Registration :

```

<!DOCTYPE html>
<!--
To change this license header, choose License Headers in Project Properties.

```

To change this template file, choose Tools | Templates  
and open the template in the editor.

-->

```
<html><br>
<head>
  <link rel="stylesheet" href="total.css">

  <link href="https://fonts.googleapis.com/css2?family=Balsamiq+Sans&display=swap"
rel="stylesheet">

  <title>View Events </title>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <h2 id="common" style="text-align:center">!! Welcome All Participants !! </h1>
</head>
<body>
  <div>
    <center>
<form method="post" action="Register">
  Enter Event Name:<br><br>
  <input type="text" id="text" name="ename"><br><p></p>
  Enter Event No:<br><br>
  <input type="text" id="text" name="enum"><br><p></p>
  Enter Card No:<br><br>
  <input type="text" id="text" name="cardno"><br><p></p>

  Enter Expiry Date:<br><br>
  <input type="text" id="text" name="edate"><br><p></p>

  Enter CVV No:<br><br>
  <input type="password" id="text" name="cvv"><br><p></p>
  Enter Card Holder's Name:<br><br>
```

```

<input type="text" id="text" name="cname"><br><p></p>
Enter Regno No:<br><br>
<input type="text" id="text" name="reg"><br><p></p>
Enter Date:<br><br>
<input type="text" id="text" name="date"><br><p></p>

<input type="submit" id="sub" value="Pay">
</form>

<br><br>
</center>
</div>
</body>
</html>

```

## • TransactionDetails :

```

<!-- Transaction details :- -->

<!DOCTYPE html>
<html><br>
<head>
<link rel="stylesheet" href="total.css">

<link href="https://fonts.googleapis.com/css2?family=Balsamiq+Sans&display=swap"
rel="stylesheet">

<title>Transaction Details</title>
<meta charset="UTF-8">
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<h2 id="common" style="text-align:center"> Transaction Details: </h1>
</head>

```

```

<marquee scrollamount="15" width="100%" direction="left" height="100px">
    <h2 style="color:black"> View Transaction Details.</h2>
</marquee>
<body><div class="">

    <center>
        

        <h2> View Transaction Details </h2>
        <a href="TraH"><button type="button" id="button">View All Transaction Details
        </button></a>
        <br><br>
    </center>
    <br>   </div>
</body>
</html>

```

## • ViewEvents :

```

<!DOCTYPE html>
<!--
To change this license header, choose License Headers in Project Properties.
To change this template file, choose Tools | Templates
and open the template in the editor.
-->
<html><br>
<head>
    <link rel="stylesheet" href="total.css">

    <link href="https://fonts.googleapis.com/css2?family=Balsamiq+Sans&display=swap"
    rel="stylesheet">
<title>View Events </title>

```

```

<meta charset="UTF-8">
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<h2 id="common" style="text-align:center">!! Welcome All Participants !! </h1>
</head>
<body>
<center>
<h2 style="color: black"> 1. View Event Link </h2>
<a href="ViewEvent1"><button type="button" id="button">View Event</button></a>
<br><br>
<h2 style="color: black">2. View Event Details</h2>
<a href="ViewEventD"><button>View Event Details</button> </a>
</center>
</body>
</html>

```

## **B**

## **Java Application Codes**

- **Adminframe.java :**

```

package Adminframe;

import Edetailsframe.*;
import Mainframe.*;
import Transactionviewframe.Transactionviewframe;

import java.util.*;
import java.io.*;
import javax.swing.*;
import java.awt.*;
import java.awt.event.*;

```

```

public class Adminframe extends JFrame{
    Container c;
    JButton b1,b2,b3;
    public Adminframe(){
        c=getContentPane();
        c.setLayout(null);

        b1=new JButton("Event Details");
        b2=new JButton("Transaction Details");
        b3=new JButton("<");
        c.add(b1);
        c.add(b2);
        c.add(b3);
        b1.setBounds(100,20,200,30);
        b2.setBounds(100,70,200,30);
        b3.setBounds(5,10,45,30);

        ActionListener a1=(ae)->{
            Mainframe a=new Mainframe();
            this.dispose();
        };
        b3.addActionListener(a1);

        ActionListener a2=(ae)->{
            Edetailsframe a=new Edetailsframe();
            this.dispose();
        };
        b1.addActionListener(a2);

        ActionListener a3=(ae)->{
            Transactionviewframe a=new Transactionviewframe();
            this.dispose();
        };
        b2.addActionListener(a3);

        this.setSize(400,200);
        this.setTitle("Event Registration");
        this.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        this.setVisible(true);
        this.setLocationRelativeTo(null);
    }// end of const.
}// end of Adminframe

```

- **Clubframe.java :-**

```
package Clubframe;

import Edetailsframe.*;

import java.util.*;
import java.io.*;
import java.sql.*;
import javax.swing.*;
import java.awt.*;
import java.awt.event.*;

public class Clubframe extends JFrame{
    Container c;

    JLabel name,fc,cd,con_name,con_no;
    JTextField name1,fc1,cd1,con_name1,con_no1;
    JButton b1,b2;

    public Clubframe(){
        c=getContentPane();
        c.setLayout(null);

        name=new JLabel("Name");
        fc=new JLabel("Faculty Coord.");
        cd=new JLabel("Description");
        con_name=new JLabel("Contact Person");
        con_no=new JLabel("Contact Number");

        name1=new JTextField(20);
        fc1=new JTextField(20);
        cd1=new JTextField(200);
        con_name1=new JTextField(20);
        con_no1=new JTextField(10);

        b1=new JButton("Submit");
        b2=new JButton("Back");

        c.add(b1);
        c.add(b2);
        c.add(name);
        c.add(fc);
        c.add(cd);
        c.add(con_name);
        c.add(con_no);
```

```

c.add(name1);
c.add(fc1);
c.add(cd1);
c.add(con_name1);
c.add(con_no1);

name.setBounds(20,20,120,30);
fc.setBounds(20,70,120,30);
cd.setBounds(20,120,120,30);
con_name.setBounds(20,170,120,30);
con_no.setBounds(20,220,120,30);

name1.setBounds(140,20,240,30);
fc1.setBounds(140,70,240,30);
cd1.setBounds(140,120,240,30);
con_name1.setBounds(140,170,240,30);
con_no1.setBounds(140,220,240,30);

b1.setBounds(75,270,100,30);
b2.setBounds(225,270,100,30);

ActionListener a1=(ae)->{
    Edetailsframe a=new Edetailsframe();
    this.dispose();
};
b2.addActionListener(a1);

ActionListener a2=(ae)->{
    Connection con=null;
    try{
        DriverManager.registerDriver(new oracle.jdbc.driver.OracleDriver());

con=DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:xe","Event","e
1234");
        String s1="insert into club values(?,?,?,?,?)";
        PreparedStatement p1=con.prepareStatement(s1);
        String cnam=name1.getText();
        String f=fc1.getText();
        String c1=cd1.getText();
        String con_nam=con_name1.getText();
        double con_n=Double.parseDouble(con_no1.getText());
        p1.setString(1,cnam);
        p1.setString(2,f);
        p1.setString(3,c1);
        p1.setString(4,con_nam);
    }
}

```

```

        p1.setDouble(5,con_n);
        int r1=p1.executeUpdate();
        JOptionPane.showMessageDialog(c,r1+" rows inserted.");
        p1.close();
    }
    catch(SQLException e){
        JOptionPane.showMessageDialog(c,"Issues: "+e);
    }
    catch(NumberFormatException e){
        JOptionPane.showMessageDialog(c,"Issues: "+e);
    }
    finally{
        try{
            con.close();
        }
        catch(SQLException e){
            JOptionPane.showMessageDialog(c,"Issues: "+e);
        }
    }
};

b1.addActionListener(a2);

this.setSize(400,370);
this.setTitle("Event Registration");
this.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
this.setVisible(true);
this.setLocationRelativeTo(null);
}// end of const.
}// end of Clubframe

```

- **ClubViewFrame.java :-**

```

package Clubviewframe;

import Edetailsframe.*;

import java.util.*;
import java.io.*;
import javax.swing.*;
import java.awt.*;
import java.awt.event.*;
import java.sql.*;

```

```
public class Clubviewframe extends JFrame{
    Container c;

    JTextArea textArea;
    JScrollPane scroll;
    JButton b1,b2;

    public Clubviewframe(){
        c=getContentPane();
        c.setLayout(null);

        textArea = new JTextArea();
        scroll = new JScrollPane(textArea);
        b1=new JButton("Back");
        b2=new JButton("View");

        c.add(scroll);
        c.add(b1);
        c.add(b2);

        scroll.setBounds(20,20,350,270);
        b1.setBounds(225,310,100,30);
        b2.setBounds(75,310,100,30);

        ActionListener a1=(ae)->{
            Edetailsframe a=new Edetailsframe();
            this.dispose();
        };
        b1.addActionListener(a1);

        ActionListener a2=(ae)->{
            Connection con=null;
            String z="";
            try{
                DriverManager.registerDriver(new
oracle.jdbc.driver.OracleDriver());
```

```

con=DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:xe
","Event","e1234");
    Statement s1=con.createStatement();
    String s2="select * from club";
    ResultSet rs=s1.executeQuery(s2);
    while(rs.next()){
        String a=rs.getString(1);
        String b=rs.getString(2);
        String c=rs.getString(3);
        String d=rs.getString(4);
        String e=rs.getString(5);
        z+="Name: "+a+"\nFaculty Co-ordinator:
"+b+"\nDescription: "+c+"\nStudent Co-ordinator: "+d+"\nContact:
"+e+"\n\n";
    }
    textArea.setText(z);
    rs.close();
    s1.close();
}
catch(SQLException e){
    System.out.println("Issues: "+e);
}
catch(NumberFormatException e){
    System.out.println("Issues: "+e);
}
finally{
    try{
        con.close();
    }
    catch(SQLException e){
        System.out.println("Issues: "+e);
    }
}
};

b2.addActionListener(a2);

this.setSize(400,390);
this.setTitle("Event Registration");

```

```
this.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
this.setVisible(true);
this.setLocationRelativeTo(null);
}// end of const.
}// end of ClubViewframe
```

- **EdetailsFrame.java :**

```
package Edetailsframe;

import Adminframe.*;
import Clubviewframe.*;
import Clubframe.*;
import Eventframe.*;
import Eventviewframe.*;
import Venueframe.Venueframe;
import Venueviewframe.Venueviewframe;

import java.util.*;
import java.io.*;
import javax.swing.*;
import java.awt.*;
import java.awt.event.*;

public class Edetailsframe extends JFrame{

    Container c;
    JButton b1,b2,b3,b4,b5,b6,b7,b8;

    public Edetailsframe(){
        c=getContentPane();
        c.setLayout(null);

        b1=new JButton("Create Event");
        b2=new JButton("View Events");
        //b3=new JButton("Update Event");
        b4=new JButton("<");


```

```

b5=new JButton("Create Club");
b6=new JButton("View Clubs");
b7=new JButton("Create Venue");
b8=new JButton("View Venue");

c.add(b1);
c.add(b2);
//c.add(b3);
c.add(b4);
c.add(b5);
c.add(b6);
c.add(b7);
c.add(b8);

b1.setBounds(100,20,200,30);
b2.setBounds(100,70,200,30);
//b3.setBounds(100,120,200,30);
b5.setBounds(100,120,200,30);
b6.setBounds(100,170,200,30);
b7.setBounds(100,220,200,30);
b8.setBounds(100,270,200,30);
b4.setBounds(5,10,45,30);

ActionListener a1=(ae)->{
    Adminframe a=new Adminframe();
    this.dispose();
};
b4.addActionListener(a1);

ActionListener a2=(ae)->{
    Clubframe a=new Clubframe();
    this.dispose();
};
b5.addActionListener(a2);

ActionListener a3=(ae)->{
    Eventframe a=new Eventframe();
    this.dispose();
};
b1.addActionListener(a3);

```

```

    ActionListener a4=(ae)->{
        Venueframe a=new Venueframe();
        this.dispose();
    };
    b7.addActionListener(a4);

    ActionListener a5=(ae)->{
        Eventviewframe a=new Eventviewframe();
        this.dispose();
    };
    b2.addActionListener(a5);

    ActionListener a6=(ae)->{
        Clubviewframe a=new Clubviewframe();
        this.dispose();
    };
    b6.addActionListener(a6);

    ActionListener a7=(ae)->{
        Venueviewframe a=new Venueviewframe();
        this.dispose();
    };
    b8.addActionListener(a7);

this.setSize(400,370);
this.setTitle("Event Registration");
this.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
this.setVisible(true);
this.setLocationRelativeTo(null);
}// end of const.
}// end of Edetailsframe

```

- **EventDetailsFrame.java :**

```

package Eventdetailsframe;

import Eventframe.*;

```

```
import java.util.*;
import java.io.*;

import javax.swing.*;
import java.awt.*;
import java.awt.event.*;
import java.sql.*;
import java.text.SimpleDateFormat;
import java.text.ParseException;

public class Eventdetailsframe extends JFrame{
    Container c;

    JLabel no,date,time,fees,dur,des,s_name,s_con;
    JTextField no1,date1,time1,fees1,dur1,des1,s_name1,s_con1;
    JButton b1,b2;

    public Eventdetailsframe(){
        c=getContentPane();
        c.setLayout(null);

        b1=new JButton("Submit");
        b2=new JButton("Back");
    }
}
```

```
no=new JLabel("Event No.");
date=new JLabel("Date");
time=new JLabel("Time");
fees=new JLabel("Fees");
dur=new JLabel("Duration");
des=new JLabel("Description");
s_name=new JLabel("Speaker Name");
s_con=new JLabel("Speaker Number");

no1=new JTextField(4);
date1=new JTextField("yyyy-MM-dd");
time1=new JTextField(8);
fees1=new JTextField(5);
dur1=new JTextField(8);
des1=new JTextField(200);
s_name1=new JTextField(20);
s_con1=new JTextField(10);

c.add(b1);
c.add(b2);
c.add(no);
c.add(date);
c.add(time);
c.add(fees);
c.add(dur);
c.add(des);
```

```
c.add(s_name);
c.add(s_con);
c.add(no1);
c.add(date1);
c.add(time1);
c.add(fees1);
c.add(dur1);
c.add(des1);
c.add(s_name1);
c.add(s_con1);

no.setBounds(20,20,120,30);
date.setBounds(20,70,120,30);
time.setBounds(20,120,120,30);
fees.setBounds(20,170,120,30);
dur.setBounds(20,220,120,30);
des.setBounds(20,270,120,30);
s_name.setBounds(20,320,120,30);
s_con.setBounds(20,370,120,30);

no1.setBounds(140,20,240,30);
date1.setBounds(140,70,240,30);
time1.setBounds(140,120,240,30);
fees1.setBounds(140,170,240,30);
dur1.setBounds(140,220,240,30);
des1.setBounds(140,270,240,30);
s_name1.setBounds(140,320,240,30);
```

```
s_con1.setBounds(140,370,240,30);
```

```
b1.setBounds(75,420,100,30);
```

```
b2.setBounds(225,420,100,30);
```

```
ActionListener a1=(ae)->{
```

```
    Eventframe a=new Eventframe();
```

```
    this.dispose();
```

```
};
```

```
b2.addActionListener(a1);
```

```
date1.addFocusListener(new FocusListener() {
```

```
    public void focusGained(FocusEvent e) {
```

```
        date1.setText("");
```

```
}
```

```
    public void focusLost(FocusEvent e) {
```

```
        // nothing
```

```
}
```

```
});
```

```
ActionListener a2=(ae)->{
```

```
    Connection con=null;
```

```
    try{
```

```
        DriverManager.registerDriver(new oracle.jdbc.driver.OracleDriver());
```

```
        con=DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:xe","Eve  
nt","e1234");
```

```
String s1="insert into event_details values(?,?,?,?,?,?,?,?,120)";

PreparedStatement p1=con.prepareStatement(s1);

int q=Integer.parseInt(no1.getText());

SimpleDateFormat formatter = new SimpleDateFormat("yyyy-MM-
dd");

java.util.Date date=null;

try {

    date = formatter.parse(date1.getText());

} catch (ParseException e){

    JOptionPane.showMessageDialog(c,"Issues: "+e);

}

java.sql.Date sqlDate = new java.sql.Date(date.getTime());

String e=time1.getText();

int r=Integer.parseInt(fees1.getText());

String t=dur1.getText();

String y=des1.getText();

String i=s_name1.getText();

double o=Double.parseDouble(s_con1.getText());

p1.setInt(1,q);

p1.setDate(2,sqlDate);

p1.setString(3,e);

p1.setInt(4,r);

p1.setString(5,t);

p1.setString(6,y);

p1.setString(7,i);

p1.setDouble(8,o);

int r1=p1.executeUpdate();

JOptionPane.showMessageDialog(c,r1+" rows inserted.");
```

```

        p1.close();
    }

    catch(SQLException e){
        JOptionPane.showMessageDialog(c,"Issues: "+e);
    }

    catch(NumberFormatException e){
        JOptionPane.showMessageDialog(c,"Issues: "+e);
    }

    finally{
        try{
            con.close();
        }
        catch(SQLException e){
            JOptionPane.showMessageDialog(c,"Issues: "+e);
        }
    }

};

b1.addActionListener(a2);

this.setSize(400,510);

this.setTitle("Event Registration");

this.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);

this.setVisible(true);

this.setLocationRelativeTo(null);

}// end of const.

}// end of Eventdetailsframe

```

- **EventFrame.java :**

```
package Eventframe;

import Edetailsframe.*;
import Eventdetailsframe.*;

import java.util.*;
import java.io.*;
import javax.swing.*;
import java.awt.*;
import java.awt.event.*;
import java.sql.*;
import java.text.ParseException;

public class Eventframe extends JFrame{
    Container c;

    JLabel no,name,cn,cor_name,cor_no,rn;
    JTextField no1,name1,cn1,cor_name1,cor_no1,rn1;
    JButton b1,b2,b3;

    public Eventframe(){
        c=getContentPane();
        c.setLayout(null);

        no=new JLabel("Number");

```

```
name=new JLabel("Name");
cn=new JLabel("Club Name");
cor_name=new JLabel("Co-ordinator");
cor_no=new JLabel("Co-ordinator Number");
rn=new JLabel("Room No.");

no1=new JTextField(4);
name1=new JTextField(20);
cn1=new JTextField(20);
cor_name1=new JTextField(20);
cor_no1=new JTextField(10);
rn1=new JTextField(4);

b1=new JButton("Event Details");
b2=new JButton("Submit");
b3=new JButton("Back");

c.add(no);
c.add(name);
c.add(cn);
c.add(cor_name);
c.add(cor_no);
c.add(rn);
c.add(no1);
c.add(name1);
c.add(cn1);
c.add(cor_name1);
```

```
c.add(cor_no1);
c.add(rn1);

c.add(b1);
c.add(b2);
c.add(b3);

no.setBounds(20,20,120,30);
name.setBounds(20,70,120,30);
cn.setBounds(20,120,120,30);
cor_name.setBounds(20,170,120,30);
cor_no.setBounds(20,220,120,30);
rn.setBounds(20,270,120,30);

no1.setBounds(140,20,240,30);
name1.setBounds(140,70,240,30);
cn1.setBounds(140,120,240,30);
cor_name1.setBounds(140,170,240,30);
cor_no1.setBounds(140,220,240,30);
rn1.setBounds(140,270,240,30);

b1.setBounds(25,320,100,30);
b2.setBounds(150,320,100,30);
b3.setBounds(275,320,100,30);

ActionListener a1=(ae)->
    Edetailsframe a=new Edetailsframe();
```

```

        this.dispose();
    };
    b3.addActionListener(a1);

    ActionListener a3=(ae)->{
        Eventdetailsframe a=new Eventdetailsframe();
        this.dispose();
    };
    b1.addActionListener(a3);

    ActionListener a2=(ae)->{
        Connection con=null;
        try{
            DriverManager.registerDriver(new oracle.jdbc.driver.OracleDriver());
            con=DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:xe","Event","e1234");
            String s1="insert into event values(?,?,?,?,?,?)";
            PreparedStatement p1=con.prepareStatement(s1);
            int q=Integer.parseInt(no1.getText());
            String w=name1.getText();
            String e=cn1.getText();
            String t=cor_name1.getText();
            double y=Double.parseDouble(cor_no1.getText());
            int u=Integer.parseInt(rn1.getText());
            p1.setInt(1,q);
            p1.setString(2,w);

```

```

        p1.setString(3,e);
        p1.setString(4,t);
        p1.setDouble(5,y);
        p1.setInt(6,u);

        int r1=p1.executeUpdate();
        JOptionPane.showMessageDialog(c,r1+" rows inserted.");
        p1.close();
    }

    catch(SQLException e){
        JOptionPane.showMessageDialog(c,"Issues: "+e);
    }

    catch(NumberFormatException e){
        JOptionPane.showMessageDialog(c,"Issues: "+e);
    }

    finally{
        try{
            con.close();
        }
        catch(SQLException e){
            JOptionPane.showMessageDialog(c,"Issues: "+e);
        }
    }
};

b2.addActionListener(a2);

this.setSize(400,420);

```

```
    this.setTitle("Event Registration");
    this.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
    this.setVisible(true);
    this.setLocationRelativeTo(null);
} // end of const.
}// end of Eventframe
```

- **EventViewFrame.java :**

```
package Eventviewframe;

import Edetailsframe.*;
import java.util.*;
import java.io.*;
import javax.swing.*;
import java.awt.*;
import java.awt.event.*;
import java.sql.*;

public class Eventviewframe extends JFrame{
    Container c;

    JTextArea textArea;
    JScrollPane scroll;
    JButton b1,b2;

    public Eventviewframe(){
        c=getContentPane();
        c.setLayout(null);
```

```

textArea = new JTextArea();
scroll = new JScrollPane(textArea);
b1=new JButton("Back");
b2=new JButton("View");

c.add(scroll);
c.add(b1);
c.add(b2);

scroll.setBounds(20,20,350,270);
b1.setBounds(225,310,100,30);
b2.setBounds(75,310,100,30);

ActionListener a1=(ae)->{
    Edetailsframe a=new Edetailsframe();
    this.dispose();
};

b1.addActionListener(a1);

ActionListener a2=(ae)->{
    Connection con=null;
    String z="";
    try{
        DriverManager.registerDriver(new oracle.jdbc.driver.OracleDriver());
        con=DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:xe","Event","e1234");
        Statement s1=con.createStatement();
        String s2="select * from event natural join event_details";
        ResultSet rs=s1.executeQuery(s2);
        while(rs.next()){

```

```

        int a=rs.getInt(1);
        String b=rs.getString(2);
        String c=rs.getString(3);
        String d=rs.getString(4);
        String e=rs.getString(5);
        int f=rs.getInt(6);
        String g=rs.getString(7);
        String h=rs.getString(8);
        int i=rs.getInt(9);
        String j=rs.getString(10);
        String k=rs.getString(11);
        String l=rs.getString(12);
        String m=rs.getString(13);
        int n=rs.getInt(14);

        z+="No.: "+a+"\nName: "+b+"\nClub Name: "+c+"\nCoordinator Name:
"+d+"\nCo-ordinator Number: "+e+"\nRoom ID: "+f+"\nDate"+g+"\nTime: "+h+"\nFees:
"+i+"\nDuration: "+j+"\nDescription: "+k+"\nSpeaker Name: "+l+"\nSpeaker Number:
"+m+"\nSeats: "+n+"\n\n";
    }

    textArea.setText(z);
    rs.close();
    s1.close();
}

catch(SQLException e){
    System.out.println("Issues: "+e);
}

catch(NumberFormatException e){
    System.out.println("Issues: "+e);
}

finally{
    try{

```

```

        con.close();
    }

    catch(SQLException e){
        System.out.println("Issues: "+e);
    }
}

};

b2.addActionListener(a2);

this.setSize(400,390);
this.setTitle("Event Registration");
this.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
this.setVisible(true);
this.setLocationRelativeTo(null);
}// end of const.

}// end of EventViewframe

```

- **MainFrame.java :**

```

package Mainframe;

import Adminframe.*;
import Participantframe.*;

import java.util.*;
import java.io.*;
import javax.swing.*;
import java.awt.*;
import java.awt.event.*;

```

```

public class Mainframe extends JFrame{

```

```

Container c;
JButton b1,b2;

public Mainframe(){
    c=getContentPane();
    c.setLayout(null);

    b1=new JButton("Admin");
    b2=new JButton("Participant");
    c.add(b1);
    c.add(b2);
    b1.setBounds(100,20,200,30);
    b2.setBounds(100,70,200,30);

    ActionListener a1=(ae)->{
        Adminframe a=new Adminframe();
        this.dispose();
    };
    b1.addActionListener(a1);

    ActionListener a2=(ae)->{
        Participantframe a=new Participantframe();
        this.dispose();
    };
    b2.addActionListener(a2);

    this.setSize(400,200);
    this.setTitle("Event Registration");
    this.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
    this.setVisible(true);
    this.setLocationRelativeTo(null);
}// end of const.

public static void main(String... args){
    Mainframe m=new Mainframe();

}// end of main
}// end of class mainframe

```

- **ParticipantFrame.java :**

```
package Participantframe;

import Mainframe.*;
import Payment1.*;
import Peventviewframe.Peventviewframe;
import Studentframe.Studentframe;

import java.util.*;
import java.io.*;
import javax.swing.*;
import java.awt.*;
import java.awt.event.*;

public class Participantframe extends JFrame{
    Container c;
    JButton b1,b2,b3,b4;
    public Participantframe(){
        c=getContentPane();
        c.setLayout(null);

        b1=new JButton("View Events");
        b2=new JButton("Payment");
        b4=new JButton("Student");
        b3=new JButton("<");
        c.add(b1);
        c.add(b2);
        c.add(b3);
        c.add(b4);
        b1.setBounds(100,20,200,30);
        b2.setBounds(100,70,200,30);
        b4.setBounds(100,120,200,30);
        b3.setBounds(5,10,45,30);

        ActionListener a1=(ae)->{
            Mainframe a=new Mainframe();
            this.dispose();
        };
    }
}
```

```

b3.addActionListener(a1);

ActionListener a2=(ae)->{
    Studentframe a=new Studentframe();
    this.dispose();
};

b4.addActionListener(a2);

ActionListener a3=(ae)->{
    Payment1 a=new Payment1();
    this.dispose();
};

b2.addActionListener(a3);

ActionListener a4=(ae)->{
    Peventviewframe a=new Peventviewframe();
    this.dispose();
};

b1.addActionListener(a4);

this.setSize(400,210);
this.setTitle("Event Registration");
this.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
this.setVisible(true);
this.setLocationRelativeTo(null);
}// end of const.
}// end of Participantframe

```

- **Payment1.java :**

```

package Payment1;

import Participantframe.Participantframe;
import Payment2.Payment2;

import java.util.*;
import java.io.*;

```

```
import javax.swing.*;
import java.awt.*;
import java.awt.event.*;
import java.sql.*;
import java.text.SimpleDateFormat;
import java.util.Date;

public class Payment1 extends JFrame{
    Container c;

    JLabel name;
    JTextField name1;
    JButton b1,b2;

    public Payment1(){
        c=getContentPane();
        c.setLayout(null);

        name=new JLabel("Name");
        name1=new JTextField(20);

        b1=new JButton("Register");
        b2=new JButton("Back");

        c.add(b1);
        c.add(b2);
        c.add(name);
        c.add(name1);

        name.setBounds(20,20,120,30);
        name1.setBounds(140,20,240,30);
        b1.setBounds(75,70,100,30);
        b2.setBounds(225,70,100,30);

        ActionListener a1=(ae)->{
            Participantframe a=new Participantframe();
            this.dispose();
        };
        b2.addActionListener(a1);
```

```

ActionListener a2=(ae)->{
    String w=name1.getText();
    Connection con=null;
    try{
        DriverManager.registerDriver(new
oracle.jdbc.driver.OracleDriver());

con=DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:xe
","Event","e1234");

String s2="select enum from event where ename=?";
PreparedStatement s1=con.prepareStatement(s2);
s1.setString(1,w);
ResultSet rs=s1.executeQuery();
int a=0;
while( rs.next() ) {
    a = rs.getInt(1);
    System.out.println(a);
}

s2="select seats from event_details where enum = ?";
s1=con.prepareStatement(s2);
s1.setInt(1,a);
ResultSet rs1=s1.executeQuery();
int b=0;
while( rs1.next() ) {
    b = rs1.getInt(1);
    System.out.println(b);
}
b--;

s2="update event_details set seats=? where enum=?";
s1 = con.prepareStatement(s2);
s1.setInt(1,b);
s1.setInt(2,a);
s1.executeUpdate();

```

```

s2="select fees from event_details where enum=?";
PreparedStatement stmt1=con.prepareStatement(s2);
stmt1.setInt(1,a);
rs=stmt1.executeQuery();
int d=0;
while( rs.next() ) {
    d = rs.getInt(1);
    System.out.println(d);
}
//-----
//-----

Connection con1=null;
try{
    DriverManager.registerDriver(new
oracle.jdbc.driver.OracleDriver());

con1=DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:x
e","Trans","t1234");
    String s3="insert into transaction_history values(?, ?, ?)";
    PreparedStatement s4=con1.prepareStatement(s3);
    s4.setInt(1,a);
    SimpleDateFormat formatter = new
SimpleDateFormat("yyyy-MM-dd");
    java.util.Date date = new java.util.Date();
    java.sql.Date now=new java.sql.Date(date.getTime());
    s4.setString(2,"Reg No");
    s4.setDate(3,now);
    s4.executeQuery();
    s4.close();
}
catch(SQLException e){
    System.out.println("Issues: "+e);
}
catch(NumberFormatException e){
    System.out.println("Issues: "+e);
}
finally{
    try{

```

```

        con1.close();
    }
    catch(SQLException e){
        System.out.println("Issues: "+e);
    }
}

//-----
JOptionPane.showMessageDialog(c,"The fees for the selected
event is: "+d);
Payment2 q=new Payment2();
this.dispose();
rs.close();

stmt1.close();
s1.close();
}
catch(SQLException e){
    JOptionPane.showMessageDialog(c,"Issues: "+e);
}
catch(NumberFormatException e){
    JOptionPane.showMessageDialog(c,"Issues: "+e);
}
finally{
    try{
        con.close();
    }
    catch(SQLException e){
        JOptionPane.showMessageDialog(c,"Issues: "+e);
    }
}
};

b1.addActionListener(a2);

this.setSize(400,160);
this.setTitle("Event Registration");
this.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
this.setVisible(true);
this.setLocationRelativeTo(null);
}// end of const.

```

```
}// end of Payment1
```

- **Payment2.java :**

```
package Payment2;

import Participantframe.Participantframe;

import java.util.*;
import java.io.*;
import javax.swing.*;
import java.awt.*;
import java.awt.event.*;
import java.sql.*;
import java.text.SimpleDateFormat;
import java.text.ParseException;

public class Payment2 extends JFrame{
    Container c;

    JLabel cn,ed,cvv,name,rn;
    JTextField cn1,ed1,cvv1,name1,rn1;
    JButton b1;

    public Payment2(){
        c=ContentPane();
        c.setLayout(null);
```

```
cn=new JLabel("Card Number");
ed=new JLabel("Expiry-date");
cvv=new JLabel("CVV");
name=new JLabel("Name on card");
rn=new JLabel("Registration No.");

cn1=new JTextField(16);
ed1=new JTextField("yyyy-MM-dd");
cvv1=new JTextField(3);
name1=new JTextField(20);
rn1=new JTextField(9);

b1=new JButton("Pay");

c.add(b1);
c.add(cn);
c.add(ed);
c.add(cvv);
c.add(name);
c.add(rn);
c.add(cn1);
c.add(ed1);
c.add(cvv1);
c.add(name1);
c.add(rn1);

cn.setBounds(20,20,120,30);
```

```

ed.setBounds(20,70,120,30);
cvv.setBounds(20,120,120,30);
name.setBounds(20,170,120,30);
rn.setBounds(20,220,120,30);

cn1.setBounds(140,20,240,30);
ed1.setBounds(140,70,240,30);
cvv1.setBounds(140,120,240,30);
name1.setBounds(140,170,240,30);
rn1.setBounds(140,220,240,30);

b1.setBounds(150,270,100,30);

ed1.addFocusListener(new FocusListener() {
    public void focusGained(FocusEvent e) {
        ed1.setText("");
    }

    public void focusLost(FocusEvent e) {
        // nothing
    }
});

ActionListener a1=(ae)->{
    Connection con=null;
    try{
        DriverManager.registerDriver(new oracle.jdbc.driver.OracleDriver());

```

```

con=DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:xe","Trans","t1234");

String s1="insert into card values(?,?,?,?,?)";

PreparedStatement p1=con.prepareStatement(s1);

double q=Double.parseDouble(cn1.getText());

SimpleDateFormat formatter = new SimpleDateFormat("yyyy-MM-
dd");

java.util.Date date=null;

try {

    date = formatter.parse(ed1.getText());

} catch (ParseException e){

    JOptionPane.showMessageDialog(c,"Issues: "+e);

}

java.sql.Date sqlDate = new java.sql.Date(date.getTime());

int r=Integer.parseInt(cvv1.getText());

String t=name1.getText();

String y=rn1.getText();

p1.setDouble(1,q);

p1.setDate(2,sqlDate);

p1.setInt(3,r);

p1.setString(4,t);

p1.setString(5,y);

int r1=p1.executeUpdate();

JOptionPane.showMessageDialog(c,r1+" rows inserted.");

JOptionPane.showMessageDialog(c,"Payment Successful!");

this.dispose();

```

```

s1="update transaction_history set regno=? where regno=?";
p1=con.prepareStatement(s1);
p1.setString(1,y);
p1.setString(2,"18BCE0149");
p1.executeUpdate();

p1.close();
Participantframe a=new Participantframe();
}

catch(SQLException e){
JOptionPane.showMessageDialog(c,"Issues: "+e);
}

catch(NumberFormatException e){
JOptionPane.showMessageDialog(c,"Issues: "+e);
}

finally{
try{
con.close();
}

catch(SQLException e){
JOptionPane.showMessageDialog(c,"Issues: "+e);
}

}
}

};


```

```
b1.addActionListener(a1);

this.setSize(400,370);
this.setTitle("Event Registration");
this.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
this.setVisible(true);
this.setLocationRelativeTo(null);
}// end of const.

}// end of Payment2
```

- **Peventviewframe.java :**

```
package Peventviewframe;

import Participantframe.Participantframe;

import java.util.*;
import java.io.*;
import javax.swing.*;
import java.awt.*;
import java.awt.event.*;
import java.sql.*;

public class Peventviewframe extends JFrame{
    Container c;

    JTextArea textArea;
    JScrollPane scroll;
```

```
JButton b1,b2;

public Peventviewframe(){
    c= getContentPane();
    c.setLayout(null);

    textArea = new JTextArea();
    scroll = new JScrollPane(textArea);
    b1=new JButton("Back");
    b2=new JButton("View");

    c.add(scroll);
    c.add(b1);
    c.add(b2);

    scroll.setBounds(20,20,350,270);
    b1.setBounds(225,310,100,30);
    b2.setBounds(75,310,100,30);

    ActionListener a1=(ae)->{
        Participantframe a=new Participantframe();
        this.dispose();
    };
    b1.addActionListener(a1);

    ActionListener a2=(ae)->{
        Connection con=null;
```

```

String z="";
try{
    DriverManager.registerDriver(new oracle.jdbc.driver.OracleDriver());

con=DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:xe","Event","e1234");
Statement s1=con.createStatement();
String s2="select * from event natural join event_details";
ResultSet rs=s1.executeQuery(s2);
while(rs.next()){
    int a=rs.getInt(1);
    String b=rs.getString(2);
    String c=rs.getString(3);
    String d=rs.getString(4);
    String e=rs.getString(5);
    int f=rs.getInt(6);
    String g=rs.getString(7);
    String h=rs.getString(8);
    int i=rs.getInt(9);
    String j=rs.getString(10);
    String k=rs.getString(11);
    String l=rs.getString(12);
    String m=rs.getString(13);
    int n=rs.getInt(14);

    z+="No.: "+a+"\nName: "+b+"\nClub Name: "+c+"\nCoordinator
Name: "+d+"\nCo-ordinator Number: "+e+"\nRoom ID:
"+f+"\nDate"+g+"\nTime: "+h+"\nFees: "+i+"\nDuration: "+j+"\nDescription:
"+k+"\nSpeaker Name: "+l+"\nSpeaker Number: "+m+"\nSeats: "+n+"\n\n";
}
}

```

```
        }

        textArea.setText(z);

        rs.close();

        s1.close();

    }

    catch(SQLException e){

        System.out.println("Issues: "+e);

    }

    catch(NumberFormatException e){

        System.out.println("Issues: "+e);

    }

    finally{

        try{

            con.close();

        }

        catch(SQLException e){

            System.out.println("Issues: "+e);

        }

    }

};

b2.addActionListener(a2);

this.setSize(400,390);

this.setTitle("Event Registration");

this.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
```

```
    this.setVisible(true);
    this.setLocationRelativeTo(null);
}// end of const.
}// end of EventViewframe
```

- **StudentFrame.java :**

```
package Studentframe;
```

```
import Participantframe.Participantframe;
```

```
import java.util.*;
import java.io.*;
import javax.swing.*;
import java.awt.*;
import java.awt.event.*;
import java.sql.*;
```

```
public class Studentframe extends JFrame{
```

```
    Container c;
```

```
    JLabel reg,f_name,m_name,l_name,con_no;
    JTextField reg1,f_name1,m_name1,l_name1,con_no1;
    JButton b1,b2;
```

```
    public Studentframe(){
        c=getContentPane();
```

```
c.setLayout(null);

reg=new JLabel("Registration No.");
f_name=new JLabel("First Name");
m_name=new JLabel("Middle Name");
l_name=new JLabel("Last Name");
con_no=new JLabel("Contact Number");

reg1=new JTextField(9);
f_name1=new JTextField(10);
m_name1=new JTextField(10);
l_name1=new JTextField(10);
con_no1=new JTextField(10);

b1=new JButton("Submit");
b2=new JButton("Back");

c.add(b1);
c.add(b2);
c.add(reg);
c.add(f_name);
c.add(m_name);
c.add(l_name);
c.add(con_no);

c.add(reg1);
```

```
c.add(f_name1);
c.add(m_name1);
c.add(l_name1);
c.add(con_no1);

reg.setBounds(20,20,120,30);
f_name.setBounds(20,70,120,30);
m_name.setBounds(20,120,120,30);
l_name.setBounds(20,170,120,30);
con_no.setBounds(20,220,120,30);

reg1.setBounds(140,20,240,30);
f_name1.setBounds(140,70,240,30);
m_name1.setBounds(140,120,240,30);
l_name1.setBounds(140,170,240,30);
con_no1.setBounds(140,220,240,30);

b1.setBounds(75,270,100,30);
b2.setBounds(225,270,100,30);

ActionListener a1=(ae)->
{
    Participantframe a=new Participantframe();
    this.dispose();
};

b2.addActionListener(a1);

ActionListener a2=(ae)->
```

```
Connection con=null;
try{
    DriverManager.registerDriver(new oracle.jdbc.driver.OracleDriver());

    con=DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:xe","Trans","t1234");

    String s1="insert into student values(?,?,?,?,?)";
    PreparedStatement p1=con.prepareStatement(s1);
    String q=reg1.getText();
    String w=f_name1.getText();
    String e=m_name1.getText();
    String t=l_name1.getText();
    double y=Double.parseDouble(con_no1.getText());

    p1.setString(1,q);
    p1.setString(2,w);
    p1.setString(3,e);
    p1.setString(4,t);
    p1.setDouble(5,y);

    int r1=p1.executeUpdate();
    JOptionPane.showMessageDialog(c,r1+" rows inserted.");
    p1.close();
}
catch(SQLException e){
    JOptionPane.showMessageDialog(c,"Issues: "+e);
}
```

```

        catch(NumberFormatException e){
            JOptionPane.showMessageDialog(c,"Issues: "+e);
        }
    finally{
        try{
            con.close();
        }
        catch(SQLException e){
            JOptionPane.showMessageDialog(c,"Issues: "+e);
        }
    }
};

b1.addActionListener(a2);

this.setSize(400,370);
this.setTitle("Event Registration");
this.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
this.setVisible(true);
this.setLocationRelativeTo(null);
}// end of const.

}// end of Clubframe

```

- **TransactionViewFrame.java :**

```

package Transactionviewframe;

import Adminframe.Adminframe;

```

```
import java.util.*;
import java.io.*;
import javax.swing.*;
import java.awt.*;
import java.awt.event.*;
import java.sql.*;

public class Transactionviewframe extends JFrame{
    Container c;

    JTextArea textArea;
    JScrollPane scroll;
    JButton b1,b2;

    public Transactionviewframe(){
        c=getContentPane();
        c.setLayout(null);

        textArea = new JTextArea();
        scroll = new JScrollPane(textArea);
        b1=new JButton("Back");
        b2=new JButton("View");

        c.add(scroll);
        c.add(b1);
        c.add(b2);
    }
}
```

```
scroll.setBounds(20,20,350,270);
b1.setBounds(225,310,100,30);
b2.setBounds(75,310,100,30);
```

```
ActionListener a1=(ae)->{
    Adminframe a=new Adminframe();
    this.dispose();
};

b1.addActionListener(a1);
```

```
ActionListener a2=(ae)->{
    Connection con=null;
    String z="";
    try{
        DriverManager.registerDriver(new oracle.jdbc.driver.OracleDriver());

        con=DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:xe","Trans","t1234");

        Statement s1=con.createStatement();
        String s2="select * from transaction_history";
        ResultSet rs=s1.executeQuery(s2);
        while(rs.next()){
            int a=rs.getInt(1);
            String b=rs.getString(2);
            String c=rs.getString(3);
            z+="Event No.: "+a+"\nStudent's Reg: "+b+"\nDate: "+c+"\n\n";
        }
    }
}
```

```
        textArea.setText(z);

        rs.close();
        s1.close();
    }

    catch(SQLException e){
        System.out.println("Issues: "+e);
    }

    catch(NumberFormatException e){
        System.out.println("Issues: "+e);
    }

    finally{
        try{
            con.close();
        }
        catch(SQLException e){
            System.out.println("Issues: "+e);
        }
    }
};

b2.addActionListener(a2);

this.setSize(400,390);
this.setTitle("Event Registration");
this.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
this.setVisible(true);
this.setLocationRelativeTo(null);
```

```
 } // end of const.  
 } // end of EventViewframe
```

- **VenueFrame.java :**

```
 package Venueframe;  
  
 import Edetailsframe.Edetailsframe;  
  
 import java.util.*;  
 import java.io.*;  
 import javax.swing.*;  
 import java.sql.*;  
 import java.awt.*;  
 import java.awt.event.*;  
  
  
  
  
 public class Venueframe extends JFrame{  
     Container c;  
  
     JLabel rno,bldg,room;  
     JTextField rno1,bldg1,room1;  
     JButton b1,b2;  
  
  
     public Venueframe(){  
         c=getContentPane();  
         c.setLayout(null);
```

```
rno=new JLabel("Room Id");
bldg=new JLabel("Building");
room=new JLabel("Room No.");

rno1=new JTextField(4);
bldg1=new JTextField(20);
room1=new JTextField(4);

b1=new JButton("Submit");
b2=new JButton("Back");

c.add(b1);
c.add(b2);
c.add(rno);
c.add(bldg);
c.add(room);
c.add(rno1);
c.add(bldg1);
c.add(room1);

rno.setBounds(20,20,120,30);
bldg.setBounds(20,70,120,30);
room.setBounds(20,120,120,30);

rno1.setBounds(140,20,240,30);
bldg1.setBounds(140,70,240,30);
```

```
room1.setBounds(140,120,240,30);

b1.setBounds(75,170,100,30);
b2.setBounds(225,170,100,30);

ActionListener a1=(ae)->{
    Edetailsframe a=new Edetailsframe();
    this.dispose();
};

b2.addActionListener(a1);

ActionListener a2=(ae)->{
    Connection con=null;
    try{
        DriverManager.registerDriver(new oracle.jdbc.driver.OracleDriver());

        con=DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:xe","Event","e1234");

        String s1="insert into venue values(?, ?, ?)";
        PreparedStatement p1=con.prepareStatement(s1);
        int rn=Integer.parseInt(rno1.getText());
        String bld=bldg1.getText();
        int roo=Integer.parseInt(room1.getText());
        p1.setInt(1,rn);
        p1.setString(2,bld);
        p1.setInt(3,roo);
        int r1=p1.executeUpdate();
        JOptionPane.showMessageDialog(c,r1+" rows inserted.");
    }
}
```

```

        p1.close();
    }

    catch(SQLException e){
        JOptionPane.showMessageDialog(c,"Issues: "+e);
    }

    catch(NumberFormatException e){
        JOptionPane.showMessageDialog(c,"Issues: "+e);
    }

    finally{
        try{
            con.close();
        }
        catch(SQLException e){
            JOptionPane.showMessageDialog(c,"Issues: "+e);
        }
    }

};

b1.addActionListener(a2);

this.setSize(400,250);
this.setTitle("Event Registration");
this.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
this.setVisible(true);
this.setLocationRelativeTo(null);

}// end of const.

}// end of Venueframe

```

- **VenueViewFrame.java :**

```
package Venueviewframe;

import Edetailsframe.Edetailsframe;

import java.util.*;
import java.io.*;
import javax.swing.*;
import java.awt.*;
import java.awt.event.*;
import java.sql.*;

public class Venueviewframe extends JFrame{
    Container c;

    JTextArea textArea;
    JScrollPane scroll;
    JButton b1,b2;

    public Venueviewframe(){
        c=getContentPane();
        c.setLayout(null);

        textArea = new JTextArea();
        scroll = new JScrollPane(textArea);
        b1=new JButton("Back");
    }
}
```

```
b2=new JButton("View");

c.add(scroll);
c.add(b1);
c.add(b2);

scroll.setBounds(20,20,350,270);
b1.setBounds(225,310,100,30);
b2.setBounds(75,310,100,30);

ActionListener a1=(ae)->{
    Edetailsframe a=new Edetailsframe();
    this.dispose();
};

b1.addActionListener(a1);

ActionListener a2=(ae)->{
    Connection con=null;
    String z="";
    try{
        DriverManager.registerDriver(new oracle.jdbc.driver.OracleDriver());

        con=DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:xe","Event","e1234");
        Statement s1=con.createStatement();
        String s2="select * from venue";
        ResultSet rs=s1.executeQuery(s2);
        while(rs.next()){


```

```

        int a=rs.getInt(1);

        String b=rs.getString(2);

        int c=rs.getInt(3);

        z+="Room Id: "+a+"\nBuilding: "+b+"\nRoom No.: "+c+"\n\n\n";

    }

    textArea.setText(z);

    rs.close();

    s1.close();

}

catch(SQLException e){

    System.out.println("Issues: "+e);

}

catch(NumberFormatException e){

    System.out.println("Issues: "+e);

}

finally{

    try{

        con.close();

    }

    catch(SQLException e){

        System.out.println("Issues: "+e);

    }

}

};

b2.addActionListener(a2);

this.setSize(400,390);

```

```
this.setTitle("Event Registration");
this.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
this.setVisible(true);
this.setLocationRelativeTo(null);
}// end of const.
}// end ofVenueViewframe
```