A Project Report

*On*

**“Hall Booking Management System’’**

*By*

**Sahil Bipin Dhariya :-(Roll no: - 307011)**

**Pranav Anant Tambat:-(Roll no: - 307013)**

**Kushal Jayprakash Agarwal (Roll no: - 307010)**

*Under the guidance of*

# Dr. Mrs. K.S.Thakre



**Department of Information Technology**

**Sinhgad College of Engineering**

**SAVITRIBAI PHULE PUNE UNIVERSITY**

**2019-2020**

**CERTIFICATE**



Sinhgad Technical Education Society,



Department of Information Technology



Sinhgad College of Engineering,

Pune

-

41



Date:



This is to certify that,

**Sahil Bipin Dhariya :-(Roll no: - 307011)**

**Pranav Anant Tambat:-(Roll no: - 307013)**

**Kushal Jayprakash Agarwal (Roll no: - 307010)**

of class T.E. IT have successfully completed their project work on “**Hall Booking Management System**’’ at SINHGAD COLLEGE OF ENGINEERING in the partial fulfillment of the Graduate Degree course in T. E at the Department of Information Technology, in the academic Year 2019-2020. Semester –V as prescribed by the Savitribai Phule Pune University.

**Prof. Dr.Mrs. K.S.Thakre Prof. G.R. Pathak**

Guide Head of the Department

(Department of Information Technology) (Department of Information Technology)

**Acknowledgement**

I feel great pleasure in expressing my deepest sense of gratitude and sincere thanks to my guide **Prof. Dr. Mrs. K.S.Thakre** for their valuable guidance during the Project work, without which it would have been very difficult task. I have no words to express my sincere thanks for valuable guidance, extreme assistance and cooperation extended to all the **Staff Members** of my Department.

This acknowledgement would be incomplete without expressing my special thanks to **Prof. G. R. Pathak** , Head of the Department (Information Technology) for their support during the work.

I would also like to extend my heartfelt gratitude to my **Principal, Dr. S. D. Lokhande** who provided a lot of valuable support, mostly being behind the veils of college bureaucracy.

Last but not least I would like to thanks all the Teaching, Non- Teaching staff members of my Department, my parents and my colleagues those who helped me directly or indirectly for completing of this Project successfully.

**Contents**

1. **TITLE OF PROJECT:**

**“HALL BOOKING MANAGEMENT SYSTEM”**

1. **ABSTRACT**
2. **INTRODUCTION**
3. **SCOPE**
4. **SPECIFIC REQUIREMENTS**

* ***HARDWARE REQUIREMENTS***
* ***SOFTWARE REQUIREMENTS***

1. **THEORY OF SOFTWARE USED**

* ***JAVA (JDK)***
* ***MySQL***

1. **DATABASE FORMAT**
2. **ER DIAGRAM**
3. **OUTPUT SCREEN(GUI)**
4. **SAMPLE CODE**
5. **CONCLUSION**
6. **REFERENCE**

**ABSTRACT**

Nowadays, everyone relies on technology for every single thing. So, we have designed a Hall Booking Management System for helping Customer to find a details about Hall in their city according to his/her preferences just by sitting at one place just by going through the ample of database provided on the site. The purpose of this application is to help a person in finding and booking of nearby Auditory Hall by just one click.

This is a real time Java based application where it consists of information of various halls in respective cities in every aspect and every user can enter their Personal details, E-mail details and do their SIGN UP for searching hall and their further processes for booking it. They can have stored details of booked hall and see their data with re-Login. The system will generate the booked hall details and their information after every data entry and booking respective halls. The user can insert and update their booking date so easily so that no two users have access to a single hall at the same date and this validation will avoid chaos.

The Admin of the site is intended to manage all the New and working hall details in various cities of an organization. Admin can do various updating, deletion or insertion of hall details by considering various users’ needs and market signals that provides necessary information. Hence, this is the efficient solution to manage hall details in an appropriate manner. The main aim of this Hall Management System is to tack all details of halls of an organization and provide the outstanding service to the customers that will effective reduce their loads about traditional booking system.

**INTRODUCTION**

**1.1 Problem Definition**

**Need for system:**

This project is aimed at developing a user friendly, aesthetic “Hall Booking Management System”. The theory of planned behavior was applied to understand the emerging phenomenons of large numbers of users are facing various problem of booking the hall or the events by the traditional way.

Application of this theory, which uses hall details like hall name having particular cost in each city provided with administrative contact details in

different aspects and manage their data into database server , is provide the necessary information for booking halls for the events purpose.

Hall Information Systems are software solutions that allow easy managing, sharing, updating and accessing information regarding a hall. A user can see the nearby halls of their location and negotiate anything with the site admin. User can choose the respective hall according to his/her need, budget, and importance of event. This database

will be the associate with MySQL schemas so more efficiency is added. Project is the

real time example of security, privacy and the co-relational example of the user and the admin. This will be satisfactory as per the user’s needs.

## **SCOPE**

Hall Booking Management System is a database-software wherein all the information regarding the Hall can be stored and retrieved. A user with the knowledge of how to use software, follow the services steps can access and use the database information. The administrator will have access to all the information presents in the database and can do various data manipulating operations, whereas the user will have privileges to view his/her information after login account on the site through sign up. The user interface will have the necessary input options to guide them through the process of inserting and updating their information during sign up and if any wrong input is given by the user then the java validation help them for the appropriate inputs. Data will be inputted using data entry screens (user friendly interfaces), which will have all the steps for proper insertion. The administrator will have all the privileges to add or update any data in the system regarding of the hall. User after viewing the records of available halls, can do booking and also have the facility for the cancellation of the hall. Output will be in the form of an informatory table showing the past booking information of the booked hall(s) for the same user. The final Report of all booked halls will be provided to the administrator.

Hall Booking Management System will hold the following information about the hall: ---

• Hall Number with its name.

• Hall administrator contact number.

• Hall cost according to city and space.

• Hall type as an AC or Non-AC.

• Hall address in each city.

## **SPECIFIC REQUIRMENTS**

The system analysis contains a planning and design phases where a logical design of system is developed and to work accordingly a plan is established. Also the requirements of system are identified and the operating environment is identified.

1. **HARDWARE REQUIREMENTS**

**->** 4 GB RAM

**->** Intel core i5

**->** 1GB Memory Space

-> Windows 8 or 10 OS

1. **SOFTWARE REQUIREMENTS**

**->** Java

-> MySQL

-> Intellij IDE

-> JavaFX

-> JFoenix

**SOFTWARE USED**

**> Java**

Java is a programming language originally developed by James Gosling at Sun Microsystems (which is now a subsidiary of Oracle Corporation) and released in 1995 as a core component of Sun Microsystems’ Java platform. The language derives much of its syntax from C and C++ but has a simpler object model and fewer low-level facilities. Java applications are typically compiled to byte code (class file) that can run on any Java Virtual Machine (JVM) regardless of computer architecture. Java is a general-purpose, concurrent, class-based, object-oriented language that is specifically designed to have as few

implementation dependencies as possible. It is intended to let application developers "write once, run any- where". The java is independent to platform so it’s important. Java is currently one of the most popular programming languages in use, and is widely used from application software to web

applications.

James Gosling, Mike Sheridan, and Patrick Naught on initiated the Java language project in June 1991. Java was originally designed for interactive television, but it was too advanced for the digital cable television industry at the time. The language was initially called Oak after an oak tree that stood outside Gosling's office; it went by the name Green later, and was later renamed Java, from a list of random words. Gosling aimed to implement a virtual machine and a language that had a familiar C/C++

of notation.

Java is an object-oriented programming language developed by Sun Microsystems in 1990s. Since then, Java has gained enormous popularity as a computer language. Java was chosen as the programming language for network computers. It is a universal front end for enterprise database. Sun Microsystems states that, “Java is a simple, object-oriented, distributed, secure, architecture, robust, multi-threaded and dynamic language. The program can be written once, and run anywhere”. One of the most significant advantages of Java is that, it has the ability to move easily from one computer to another. It also has the ability to run the same program on many different operating systems. With such exemplary benefits, Java is a hot favorite among techies and software professional sit allows you to create modular programs and reusable codes.

**Java Features**

1. **Simple, Small and familiar:**

Java is a simple and small language. The Syntax of java is just like C++, so it is very easy to learn. It is simple because it I) does not use header files ii) eliminated the use of pointer iii) operator overloading and virtual base classes are eliminated.

1. **Object oriented:**

Java is a pure Object oriented. Everything in java is object. All programs and data reside inside objects and classes

1. **Distributed:**

Java has networking facilities. so java can create application on network.

1. **Robust:**

Java gives importance to memory management by using the technique called Garbage Collection and Exception handling.

1. **Secure:**

Since java is used on internet, security is an important issue. A security code is asked before a java code is interpreted on internet.

1. **Platform independent:**

Java compiler generates a platform independent code called byte code.

1. **Portable:**

The Byte code generated by java can be used on any machine. So it can be portable.

1. **Compiled and Interpreted:**

Generally, computer languages are either complied or interpreted. But java combines both compiler and interpreted.

1. **High performance:**

The use of byte code makes the performance high. the speed is also high with comparing c, c++.

1. **Multithreading and interactive:**

Multithreading means handling more than one job at a time. Java supports Multithreading.

1. **Dynamic and extensible:**

Java is a dynamic language. So it is capable of linking dynamic new classes, methods and objects. Java supports functions written in C and C++ also. These functions are called native methods. During Run-Time Native methods can be linked dynamically.

**> MySQL**

**MySQL** “My S-Q-L", officially, but also called “My Sequel" is the world’s most widely used open-source relational database management system (RDBMS). It is named after co-founder Michael Wideness’s daughter. The SQL phrase stands for Structured Query Language. MySQL is a popular choice of database for use in web applications, and is a central component of the widely used LAMP open source web application software stack. LAMP is an acronym for "Linux, Apache, MySQL, Perl/PHP/Python." Free-software-open source projects that require a full-featured database management system often use MySQL.

MySQL is also used in many high-profile, large-scale websites, including Wikipedia, Google (though not for searches), Facebook, Twitter, Flickr, and YouTube. MySQL is the most popular and widely used relational database management system that provides multi-user access to number of databases. MySQL is now owned by Oracle and uses Sequential Query Language to manage database. Its source is available under GNU license and propriety agreements. MySQL is most popular among PHP developers and used for websites, web applications and online services.

**Features of MySQL:**

 Because of its unique storage engine architecture MySQL performance is very high.

 Supports large number of embedded applications which makes MySQL very flexible.

 Use of Triggers, Stored procedures and views which allows the developer to give a higher productivity.

 Allows transactions to be rolled back, commit and crash recovery

 Embedded database library

 Full-text indexing and searching

 Updatable views

 Cursors

 Triggers

 Cross-platform support

**Limitation of MySQL:**

 Like other SQL databases, MySQL does not currently comply with the full SQL standard for some of the implemented functionality, including foreign key references when using some storage engines other than the default of InnoDB

 No triggers can be defined on views.

 MySQL, like most other transactional relational databases, is strongly limited by hard disk performance. This is especially true in terms of write latency

**DATABASE FORMAT**

**mysql> desc admin;**

**+----------+----------+------+-----+---------+-------+**

**| Field | Type | Null | Key | Default | Extra |**

**+----------+----------+------+-----+---------+-------+**

**| username | char(30) | YES | | NULL | |**

**| password | char(30) | YES | | NULL | |**

**+----------+----------+------+-----+---------+-------+**

**2 rows in set (0.00 sec)**

**mysql> desc hall;**

**+------------+----------+------+-----+---------+-------+**

**| Field | Type | Null | Key | Default | Extra |**

**+------------+----------+------+-----+---------+-------+**

**| hall\_no | char(30) | NO | PRI | | |**

**| hall\_name | char(30) | YES | | NULL | |**

**| address | char(30) | YES | | NULL | |**

**| cost | char(30) | YES | | NULL | |**

**| city | char(30) | YES | | NULL | |**

**| contact\_no | char(30) | YES | | NULL | |**

**| type | char(30) | YES | | NULL | |**

**+------------+----------+------+-----+---------+-------+**

**7 rows in set (0.02 sec)**

**mysql> desc history;**

**+----------+----------+------+-----+---------+-------+**

**| Field | Type | Null | Key | Default | Extra |**

**+----------+----------+------+-----+---------+-------+**

**| hall\_no | char(30) | YES | | NULL | |**

**| date | char(30) | YES | | NULL | |**

**| username | char(30) | YES | | NULL | |**

**+----------+----------+------+-----+---------+-------+**

**3 rows in set (0.00 sec)**

**mysql> desc user;**

**+-----------+----------+------+-----+---------+-------+**

**| Field | Type | Null | Key | Default | Extra |**

**+-----------+----------+------+-----+---------+-------+**

**| username | char(30) | NO | PRI | | |**

**| firstname | char(30) | YES | | NULL | |**

**| lastname | char(30) | YES | | NULL | |**

**| password | char(30) | YES | | NULL | |**

**| date | char(30) | YES | | NULL | |**

**| address | char(30) | YES | | NULL | |**

**+-----------+----------+------+-----+---------+-------+**

**6 rows in set (0.00 sec)**

**mysql> desc status;**

**+----------+----------+------+-----+---------+-------+**

**| Field | Type | Null | Key | Default | Extra |**

**+----------+----------+------+-----+---------+-------+**

**| hall\_no | char(30) | YES | MUL | NULL | |**

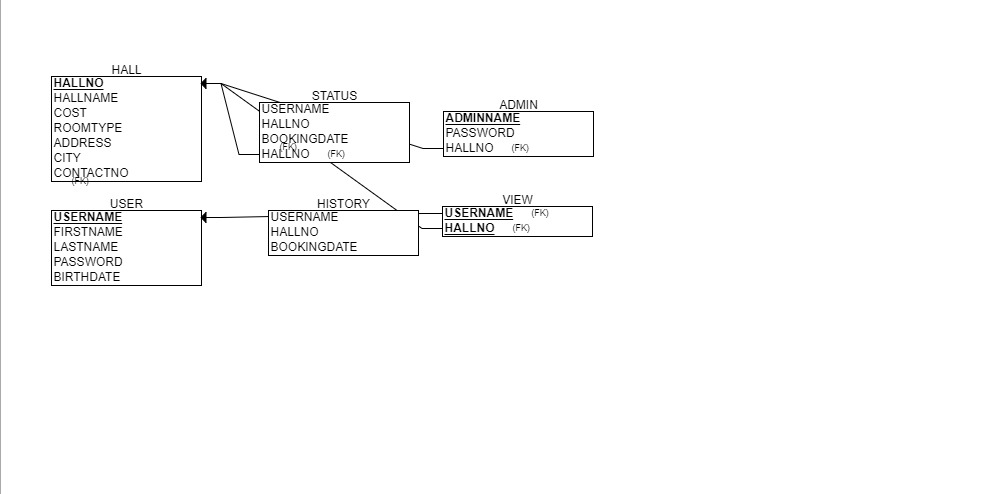
**| date | char(30) | YES | | NULL | |**

**| username | char(30) | YES | MUL | NULL | |**

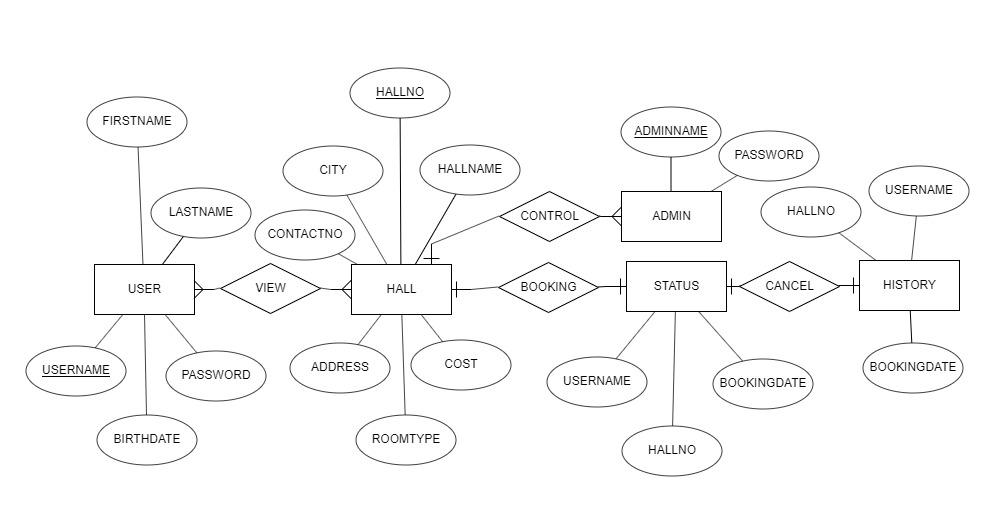
**+----------+----------+------+-----+---------+-------+**

**3 rows in set (0.00 sec)**

**SCHEMA DIAGRAM**

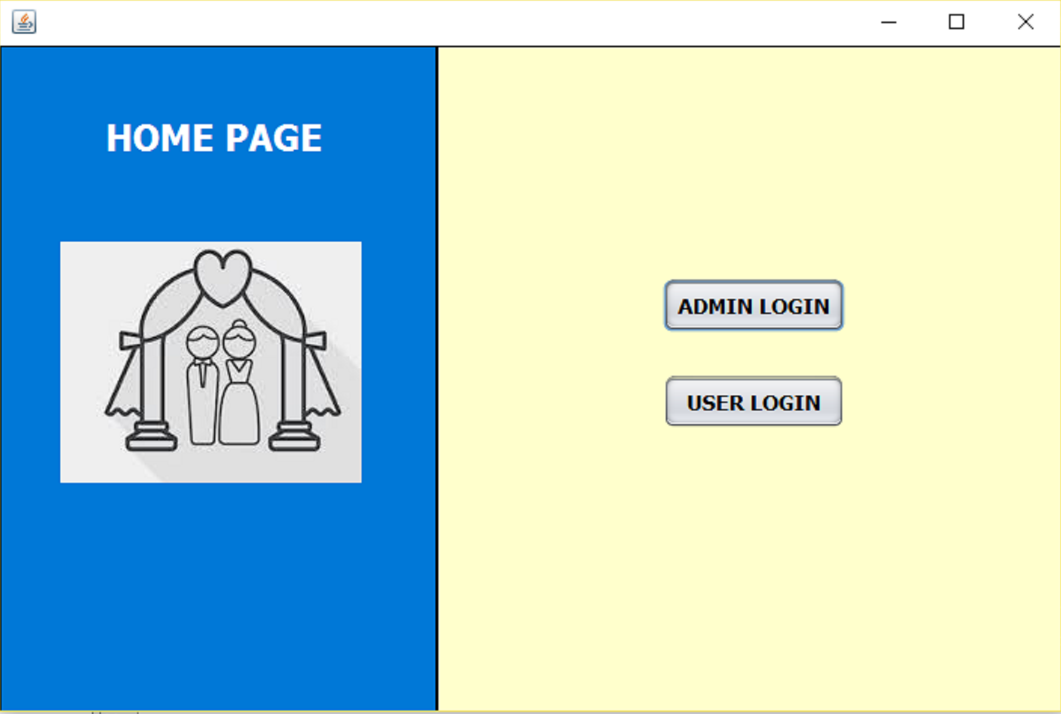
****

**ER DIAGRAM**

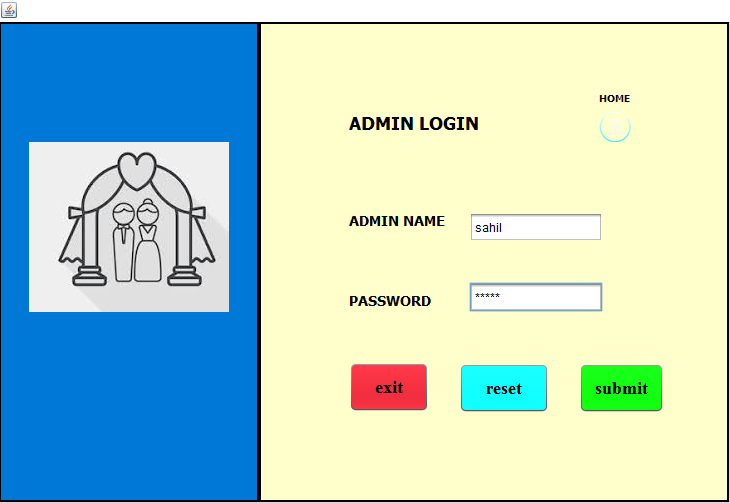


**SCREEN OUTPUT**

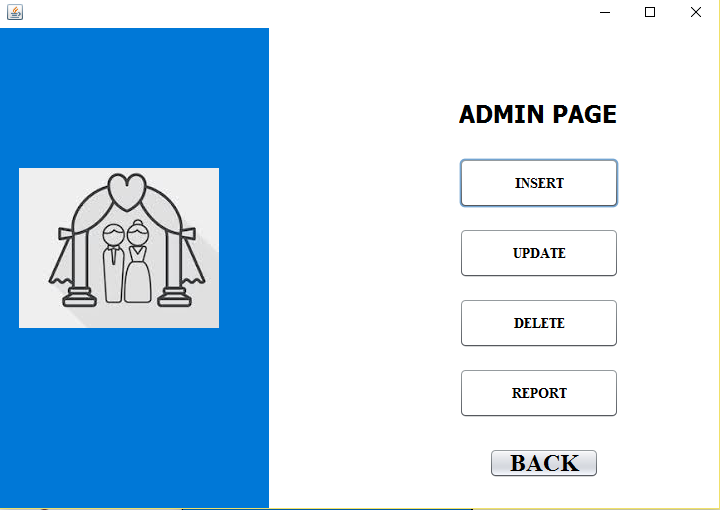
**1. Home Page:**

****

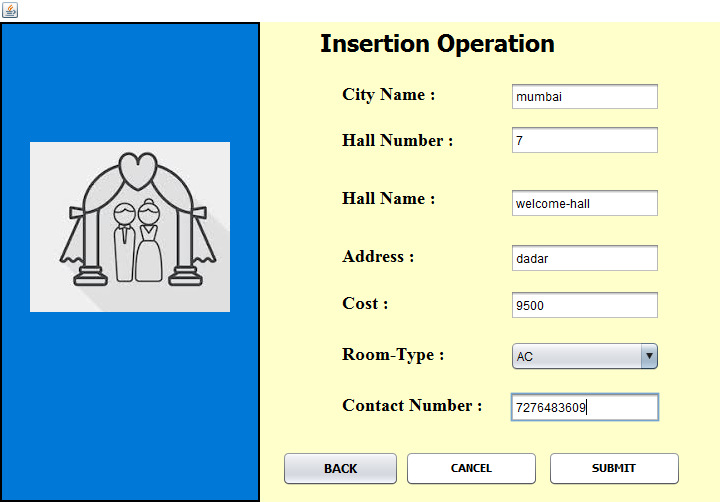
**2. Admin Login:**

****

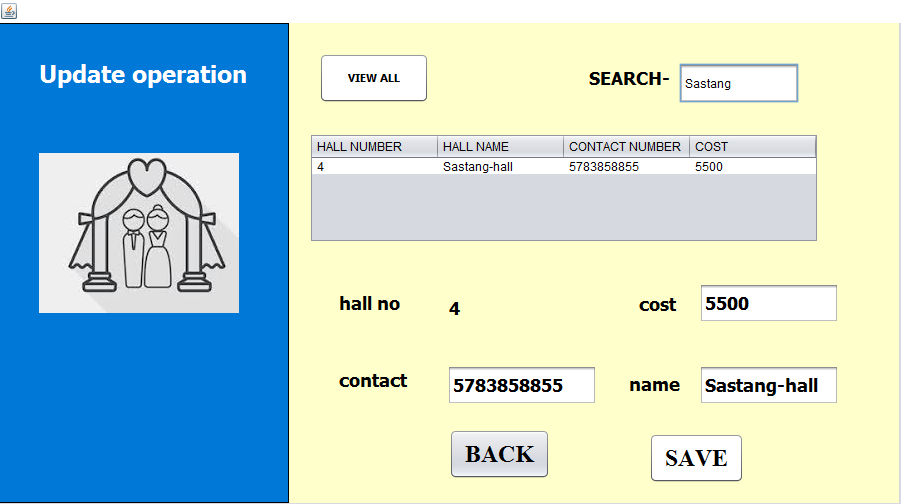
**3. Admin Page:**

****

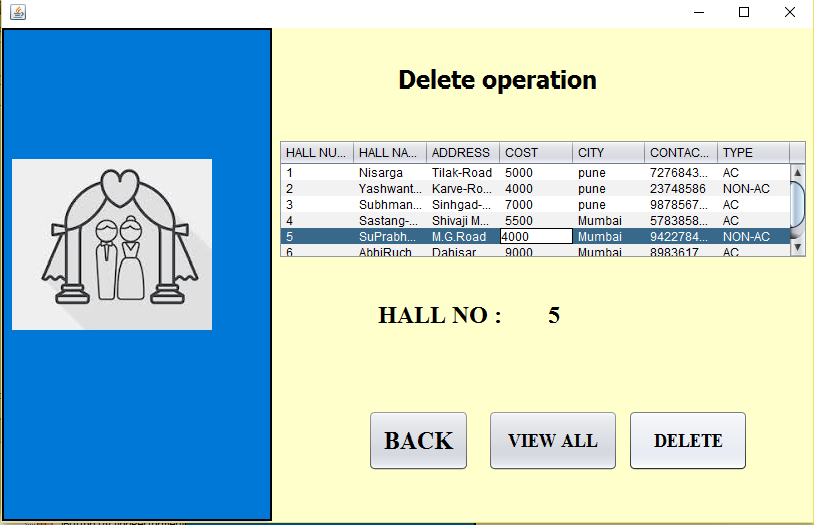
**4. Insert Page**:



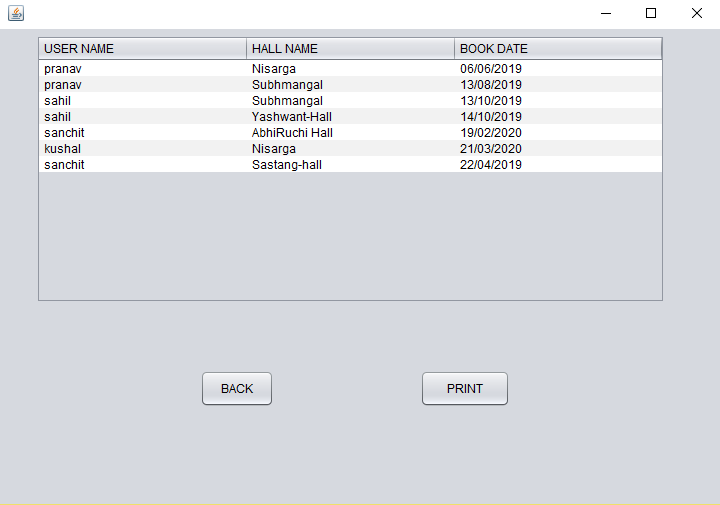
**5. Update Page:**



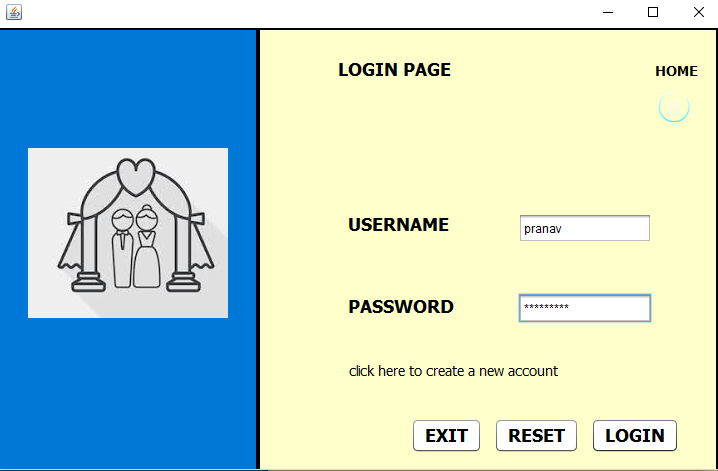
**6. Delete Page:**

****

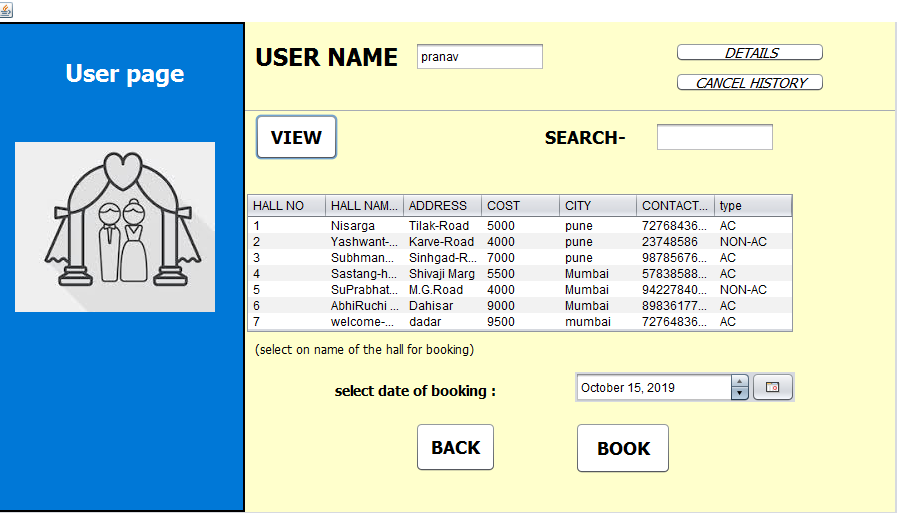
**7. Booking Record:**

****

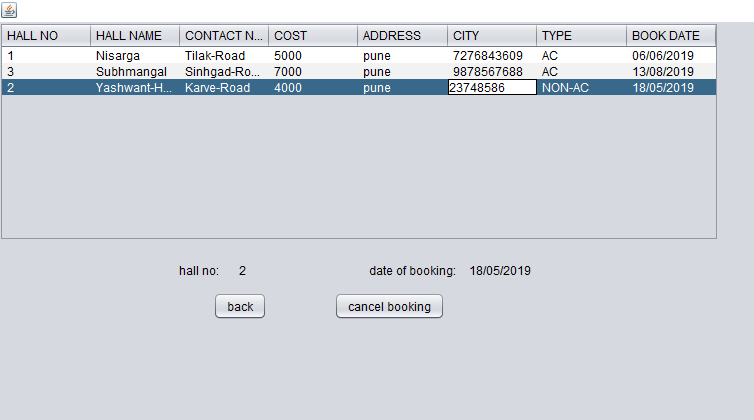
**8. User Login:**

****

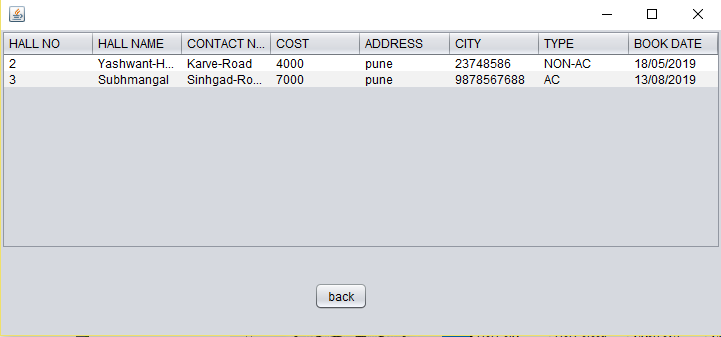
**9. User Page:**

****

**10. User Past Bookings:**

****

**11. User Cancel History:**

****

**Sample Code**

//Login sample Code

package login\_sysytem;

import javax.swing.\*;

import java.awt.\*;

import java.sql.\*;

public class Login extends javax.swing.JFrame {

public Login() {

initComponents();

}

@SuppressWarnings("unchecked")

// <editor-fold defaultstate="collapsed" desc="Generated Code">

private void initComponents() {

private void submitActionPerformed(java.awt.event.ActionEvent evt) {

// TODO add your handling code here:

String usr=username.getText();

String pass=password.getText();

try{

Class.forName("com.mysql.jdbc.Driver");

Connection con=DriverManager.getConnection(

"jdbc:mysql://localhost:3306/db","root","pranav");

PreparedStatement stmt=con.prepareStatement("select \* from user where username = ? and password = ?");

stmt.setString(1, usr);

stmt.setString(2, pass);

ResultSet rs=stmt.executeQuery();

boolean alreadyExists = false;

while(rs.next()){

alreadyExists = true;

}

if(alreadyExists == false){

JOptionPane.showMessageDialog(null, "Account Does not Exist ");

}else{

this.setVisible(false);

new User\_page(usr).setVisible(true);

}

con.close();

}catch(Exception e){

System.out.println(e);

}

}

private void jButton4ActionPerformed(java.awt.event.ActionEvent evt) {

// TODO add your handling code here:

username.setText(null);

password.setText(null);

}

private void jButton5ActionPerformed(java.awt.event.ActionEvent evt) {

// TODO add your handling code here:

System.exit(0);

}

private void jLabel5MouseClicked(java.awt.event.MouseEvent evt) {

// TODO add your handling code here:

Home h = new Home();

h.setVisible(true);

}

private void jLabel3MouseClicked(java.awt.event.MouseEvent evt) {

// TODO add your handling code here:

Register r=new Register();

r.setVisible(true);

}

private void usernameActionPerformed(java.awt.event.ActionEvent evt) {

// TODO add your handling code here:

}

private void passwordActionPerformed(java.awt.event.ActionEvent evt) {

// TODO add your handling code here:

}

public static void main(String args[]) {

try {

for (javax.swing.UIManager.LookAndFeelInfo info : javax.swing.UIManager.getInstalledLookAndFeels()) {

if ("Nimbus".equals(info.getName())) {

javax.swing.UIManager.setLookAndFeel(info.getClassName());

break;

}

}

} catch (ClassNotFoundException ex) {

java.awt.EventQueue.invokeLater(new Runnable() {

public void run() {

new Login().setVisible(true);

}

});

**CONCLUSION**

In today’s ever expanding and demanding world there is always a need as well as scope to enhance the capability of individual with the help of technology. In today’s world, where computers are used extensively so as to speed up various processes, this project of ours tries to comfort problems faced in booking hall for event purpose by traditional methods and to build ease of retrieving data of nearby halls through our application. Using this site, users can find information of any hall in particular city. This site will reduce the manual work that is instead paper work, you just need to register on this site which will provide ample of hall profiles that are registered to system by admin.

Due to this project, we were able to learn many things in IntelliJ IDE, as well as MySQL which is used as backend for database. This project also gave us the experience of working in group and securing nice co-ordination among all of the members.

**REFERENCES**

***Online References***

1. www.mysql.com
2. www.stackoverflow.com
3. www.geeksforgeeks.com
4. www.oracle.com(Java software download)

***Books Reference***

1. Database System Concepts

By - Silberschatz, Korth and Sudarshan

1. Head First Java

By – Kathy Sierra and Bert Bates

1. A Programmer’s Guide to Java SE 8 (OCA)

By – Khalid A. Mughal and Rolf W Rasmussen