

1. INTRODUCTION

1.1 PURPOSE

The purpose of this document is to build a room booking system to manage booking of rooms and to ease the same.

1.2 DOCUMENT CONVENTIONS

This document uses the following conventions.

DB: Database

ER: Entity Relationship

1.3 INTENDED AUDIENCE AND READING SUGGESTIONS

This project is a prototype for the room booking management system and it is restricted within the college premises. This is being implemented under the guidance of college professors. This project is useful for the room booking management team and as well as to the faculty.

1.4 PROJECT SCOPE

The purpose of the room booking management system is to ease room booking and to create a convenient and easy-to-use application for faculty to book rooms, cancel booked rooms online without doing it manually . The system is based on a relational database with its room management and booking functions.

2.1 OVERALL DESCRIPTION:-

Our website will handle large amount of data related to user and admin login information, Available Bookings ,Expired Bookings, Canceled Bookings and so on.

Available Bookings:-It includes details related to rooms that are vacant and can be booked by the users.It includes Booking preference by Room Number or Time Slot .

Details also include Capacity of the room.

Expired Bookings:-It includes the information about all the bookings that are completed till date .

Cancelled Bookings:-It includes the information about bookings that are canceled by the user .

Booking Details:-It should include Room number , Purpose of Booking, Date on which Booking made , Date user need to have his/her preferred room and Time slot .

2.2 PRODUCT FUNCTIONING :-

User Functioning :-

Login to the website to make/cancel/view a room booking according to his/her preference.

- **Book a room:** User has to enter date of event, purpose, time slot or preferred room number or both. A list of available rooms with time slots is displayed. He can then use book now option and make a booking.
- **View previous bookings: (Booking history)**
 - **Active bookings:** Available for cancellation. User can cancel the booking he has made by selecting cancel option.
 - **Expired bookings:** Cannot be cancelled. Cancelled bookings and completed events are shown here.

Admin Functioning:-

Login to keep a track on the activities.

- **Available bookings:** Admin can view details of bookings which are still active.

- Expired bookings: Admin can view details of bookings which are either cancelled or whose events are completed.
- Book a room: Admin like a user, can book a room(interface same as user)
- Create a new user: Only the admin has the power of creating a new user.
- Edit user's profile: Admin can edit user's profile whenever requested by the user.

2.3 BRIEF OVERVIEW OF TECHNOLOGIES USED:-

FRONT END:- HTML ,JAVACRIPT(Active and Disable Tabs), And BOOTSTRAP

Back END:- PHP, AND SQL

2.4 DESIGN AND IMPLEMENTATION CONSTRAINTS:-

An sql event is planned every minute to send expired bookings from available bookings to expired bookings.

A Trigger is fired to delete expired Bookings from available bookings just after inserting into expired bookings (through event).

3. SYSTEM FEATURES

DESCRIPTION and PRIORITY

The room booking management system maintains information on Rooms, time slots, availability and cancellations. This project has a high priority because booking rooms uniquely for a particular time slot is difficult.

STIMULUS/RESPONSE SEQUENCES

Search for available rooms according to the timeslot and room capacity.

Displays a detailed list of available rooms from which one can book the room

Cancel booking for a previously booked room.

CLIENT/SERVER SYSTEM

The term client/server refers primarily to an architecture or logical division of responsibilities, the client is the application (also known as the front-end), and the server is the DBMS (also known as the back-end).

A client/server system is a distributed system in which,

Some sites are client sites and others are server sites.

All the data resides at the server sites.

All applications execute at the client sites.

4.External interface requirements

4.1 User interfaces

- **Front end software:** html, css and bootstrap,javascript(Active and disable tabs)
- **Back end:** php, sql.

4.2 Hardware interfaces:

- Windows
- A browser that supports css,html,php

4.3 Software interfaces

Following are the software used for room booking online management application

| Software used | Description |
|-------------------------|---|
| Operating system | We have chosen windows OS |
| Database | To handle records of booking, we chose mysql database |
| | |

4.4 Communication interfaces

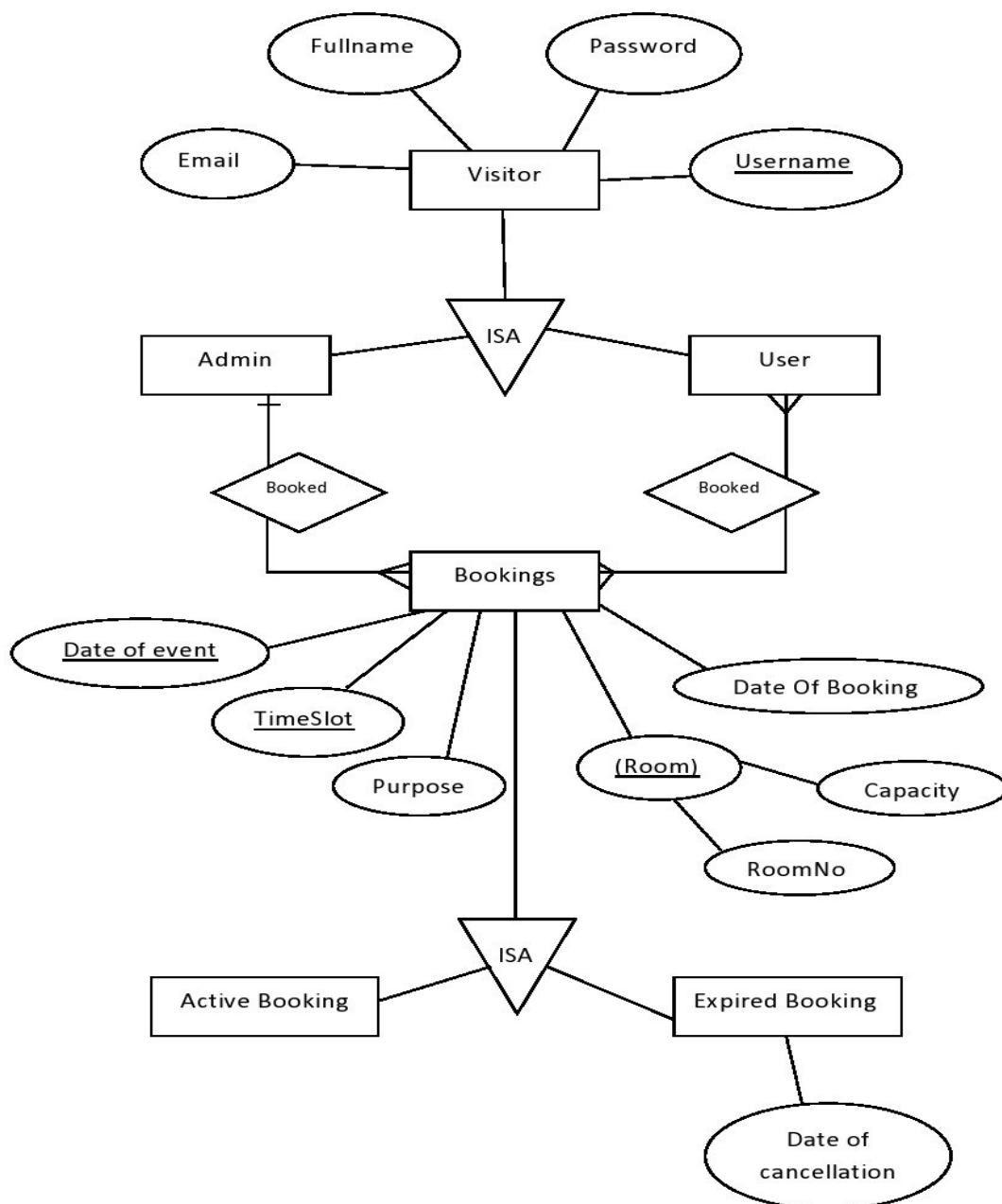
This project supports all types of web browsers. We are using electronic forms for room booking and user registration(solely for admin)

5.Nonfunctional requirements:

5.1 Performance requirements:

The steps involved to perform the implementation of room booking database are as listed below.

- **E-R DIAGRAM:** The E-R Diagram constitutes a technique for representing the logical structure of a database in a pictorial manner. This analysis is then used to organize data as a relation, normalizing relation and finally obtaining a relation database.



- **ENTITIES:** Which specify distinct real-world items in an application.
- **PROPERTIES/ATTRIBUTES:** Which specify properties of an entity and relationships.
- **RELATIONSHIPS:** Which connect entities and represent meaningful dependencies between them.
- Following are the entities and their attributes:

1. Admin: Details of admin people

- **Username:** Holds the username of the admin.
- **Password:** Password of admin to login into admin portal.
- **Full name:** Full name of the admin
- **Email:** Holds the email of admin

2. All rooms: Details of available rooms for booking.

- **Room no:** Room number
- **Capacity:** Capacity of room. Helps to choose correct room if strength is concerned factor.

3. Available bookings: Booking which are active at present.

- **Username:** Username of user who has made a booking.
- **Timeslot:** Timeslot associated with a particular booking.
- **Room no:** Room number associated with a particular booking and time slot.
- **Date of booking:** Stores the date when the booking was made.
- **Date of event:** Stores the date when the booking has to come into effect.
- **Purpose:** Purpose of booking room.

Storing this helps in disabling a room's availability for a particular time slot.

4. Expired bookings: Stores the details of a

- cancelled booking
- Expired booking (event completed).

- **Username:** Username of user who has made a booking.
- **Timeslot:** Timeslot associated with a particular booking.
- **Room no:** Room number associated with a particular booking and time slot.
- **Date of booking:** Stores the date when the booking was made.
- **Date of event:** Stores the date when the booking has to come into effect.
- **Purpose:** Purpose of booking room.
- **Date of cancellation:** Stores the date when a booking was cancelled. Stores null if a booking was expired.

5. Timeslot:

- **Time:** Stores all available timeslots for a booking.

6. Users: Details of users.

- **Username:** Holds the username of a user.
- **Password:** Password of user to login.
- **Full name:** Full name of the user.
- **Email:** Stores email ID of user to mail him as soon as a booking is made.

B) NORMALIZATION:

The basic objective of normalization is to reduce redundancy which means that information is to be stored only once. Storing information several times leads to wastage of storage space and increase in the total size of the data stored.

If a database is not properly designed it can give rise to modification anomalies. Modification anomalies arise when data is added to, changed or deleted from a database table. Similarly, in traditional databases as well as improperly designed relational databases, data redundancy can be a problem. These can be eliminated by normalizing a database.

Normalization is the process of breaking down a table into smaller tables. So that each table deals with a single theme. There are three

different kinds of modifications of anomalies and formulated the first, second and third normal forms (3NF) is considered sufficient for most practical purposes. It should be considered only after a thorough analysis and complete understanding of its implications.

5.2 SAFETY REQUIREMENTS

If there is extensive damage to a wide portion of the database due to catastrophic failure, such as a disk crash, the recovery method restores a past copy of the database that was backed up to archival storage (typically tape) and reconstructs a more current state by reapplying or redoing the operations of committed transactions from the backed up log, up to the time of failure.

5.3 SECURITY REQUIREMENTS

Security systems need database storage just like many other applications. However, the special requirements of the security market mean that vendors must choose their database partner carefully.

5.4 SOFTWARE QUALITY ATTRIBUTES

- **AVAILABILITY:** The required room should be available on the specified date and specified time to users on first come first serve basis eliminating clashes .
- **CORRECTNESS:** When a user books a room for a particular date and time slot ,then that room for that particular time slot is locked and set unavailable for booking for other users. Only the user who made a particular booking will be able to cancel it.
- **MAINTAINABILITY:** The administrators and flight in chargers should maintain correct schedules of flights.
- **USABILITY:** The flight schedules should satisfy a maximum number of customers needs.