**VISVESVARAYA TECHNOLOGICAL UNIVERSITY BELAGAVI, KARNATAKA**

*A WTA Mini Project Report*

***(Sixth Semester)***

*on*

**Concerts and Shows Reservation Platform**

*Submitted in the partial fulfillment for the requirements for the conferment of degree of*

**BACHELOR OF ENGINEERING**

in

**INFORMATION SCIENCE AND ENGINEERING**

By

**Mr. Monish S** USN: 1BY20IS088

**Mr. Murali Manohara Hegde A S** USN: 1BY20IS091

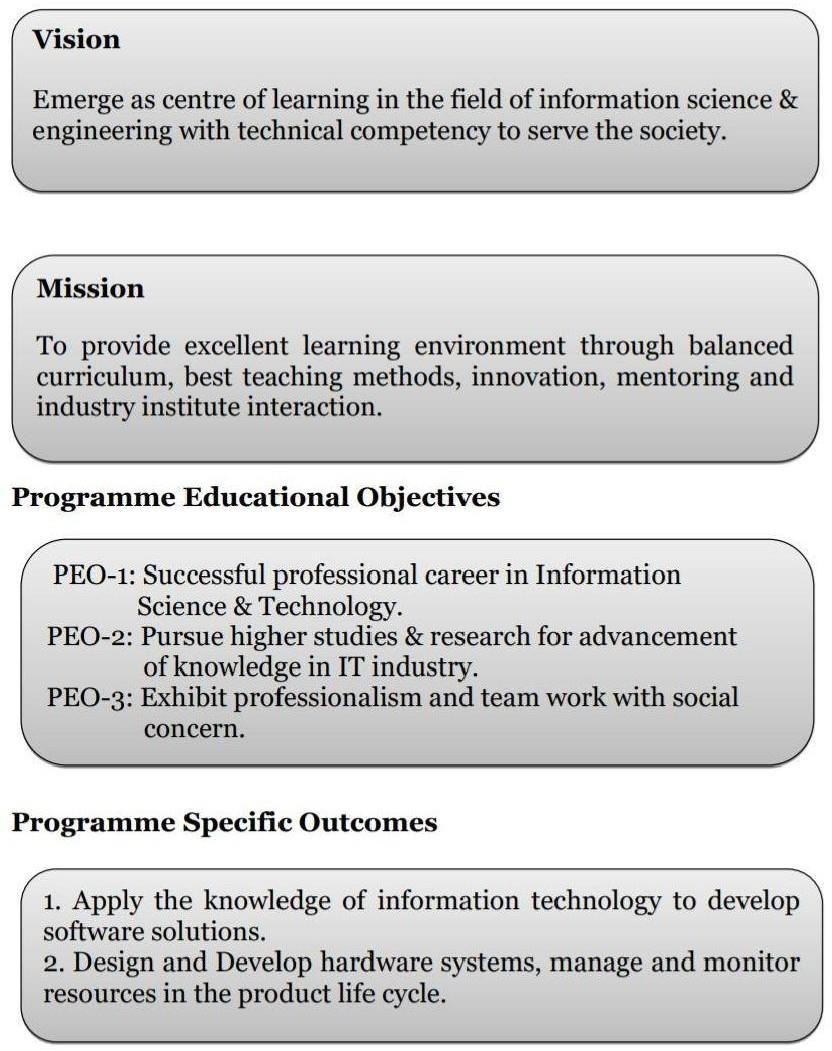
Under the guidance of

**Dr. Mohan B A** Associate Professor Dept. of ISE



**2022-2023**

# Department Vision & Mission



**BMS INSTITUTE OF TECHNOLOGY & MANAGEMENT**

#### YELAHANKA, BENGALURU-560064

**DEPARTMENT OF INFORMATION SCIENCE & ENGINEERING**



**CERTIFICATE**

This is to certify that the Mini Project (Fifth Semester) entitled **“Concerts and Shows Reservation Platform”** is a bonafide work carried out by **Mr. Monish S (1BY20IS088),** and **Mr. Murali Manohara Hegde A S(1BY20IS091)** in partial fulfillment for the award of **Bachelor of Engineering Degree in Information Science and Engineering** of the Visvesvaraya Technological University, Belagavi during the year 2022-2023. It is certified that all corrections/suggestions indicated for internal assessment have been incorporated in this report. The mini project report has been approved as it satisfies the academic requirements with respect to mini project work for the B.E Degree.

**Signature of the Guide Signature of the HOD**

Dr. Mohan B A

**Examiner**

### Name Signature

1.

### 2.

# ACKNOWLEDGEMENT

We are happy to present this Mini Project after completing it successfully. This Mini Project would not have been possible without the guidance, assistance and suggestions of many individuals. We would like to express our deep sense of gratitude and indebtedness to each and every one who has helped us make this Mini Project a success.

We are grateful to **Dr. Mohan Babu G N,** Principal, BMS Institute of Technology & Management for his constant encouragement and support.

We heartily thank the Head of the Department, Information Science and Engineering, for her constant encouragement and inspiration in taking up this project.

We gracefully thank our guide**, Dr. Mohan B A** Associate Professor, Dept. of Information Science and Engineering, for her encouragement and advice throughout the course of this project work.

Nevertheless, we express our gratitude towards our family and friends for the encouragement and support which helped us to finish this project successfully.

By,

**Monish S**

**Murali Manohara Hegde A S.**

# DECLARATION

We, hereby declare that the Mini Project titled “CONCERTS AND SHOWS RESERVATION PLATFORM” is a record of original Mini Project work undertaken for the award of the degree of Bachelor of Engineering in Information Science and Engineering of the Visvesvaraya Technological University, Belagavi during the year 2022-23. We have completed this Mini Project work under the guidance of **Dr. Mohan B A,** Associate professor, Dept. of ISE.

We also declare that this Mini Project report has not been submitted for the award of any degree, diploma, fellowship or other title anywhere else.

**Student Photos:**

**Monish S Murali Manohara Hegde A S**

# ABSTRACT

This project is a Concert and show reservation portal that is implemented on a website. This project gives feature of remotely creating, removing, information retrieval, allotting tasks to different participants]in an event. This project is capable of providing all the important access to the admin and all the people related to a particular event. It gives Hosts of Shows and Concerts access to see participants and guest list. Also, he/she will be able to create or delete a show. This project will reduce paperwork and manpower hence creating a hassle-free way of managing an event. Every even requirement will be traceable. One would be able to collect feedback from people and improve according to that feedback. Volunteers can be invited, and duties can be assigned to them, also revenue collection can be monitored through this platform. The purpose of creating this project is to digitalize the processes involved in an event, also it is easy to access so one can access it anytime and anywhere and perform a manage an event. It provides most of the basic functionality required by any event manager to smoothly run the event. This is to satisfy the needs of event manager as well as participants of event. They can enter their data and register for the event. This data is then sent to administrator which in turn can contact the participants.

# TABLE OF CONTENTS

|  |  |  |
| --- | --- | --- |
| Chapter No. | CHAPTER NAME | PAGE NO. |
|  | **Acknowledgement** | **i** |
|  | **Declaration** | **ii** |
|  | **Abstract** | **iii** |
| 1 | **Introduction**   * 1. Outline   2. Motivation and Scope   3. Problem Statement | **1 - 2**  1  2  2 |
| 2 | **Requirement Specification**   * 1. Functional Requirements   2. Non-Functional requirement   3. Domain Constraints | **3 – 4**  3  3  4 |
| 3 | **System/Requirement Analysis**   * 1. Overall System Design   2. Admin Module   3. User Module | **5 - 6**  5  6  6 |
| 4 | **System Design**   * 1. Entity Relationship Diagram   2. Schema Diagram | **7 – 10**  7  9 |

|  |  |  |  |
| --- | --- | --- | --- |
| 5 | **Implementation**   * 1. Description of Database tool   2. Description of implementation (Frontend | | **11 - 13**  11  d) 12 |
| 6 | **Testing**   * 1. Component Testing   2. System Testing | | **14 – 15**  14  15 |
| 7 | **Interpretation of Results** | | **16 – 19** |
|  | | **Conclusion** | **20** |
|  | | **References** | **21** |

|  |  |  |
| --- | --- | --- |
|  |  |  |
|  | |  |
|  | |  |

**LIST OF FIGURES**

|  |  |
| --- | --- |
| Figure No. | TITLE |
| 3.1 | Overall System Design |
| 3.2 | Admin Module |
| 3.3 | User Module |
| 4.1  4.2 - 4.6 | ER Diagram  Entity sets |
| 4.7 | Schema Diagram |
| 7.1 – 7.9 | Interpretation of Results |
|  |  |
|  |  |

**LIST OF TABLES**

|  |  |
| --- | --- |
| Table No. | TITLE |
| 6.1 | Login-Page Test |
| 6.2 | Events-Page Test |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |

# CHAPTER 1: INTRODUCTION

#### Outline:

Database is an organized collection of data. The data is typically organized to model aspects of reality in a way that supports processes requiring information. A DBMS makes it possible for end users to create, read, update and delete data in a database. The DBMS essentially serves as an interface between the database and end users or application programs, ensuring that data is consistently organized and remains easily accessible.

The DBMS manages three important things: the data, the database engine that allows data to be accessed, locked and modified and the database schema, which defines the database’s logical structure. These three foundational elements help provide concurrency, security, data integrity and uniform administration procedures. The DBMS can offer both logical and physical data independence.

Concerts and Shows Reservation Platform is a process of organizing a professional and focused event, for a particular target audience. It involves visualizing concepts, planning, budgeting, organizing and executing events such as wedding, musical concerts, corporate seminars, exhibitions, birthday celebrations, theme parties, etc. Event Management is a multi-million-dollar industry, growing rapidly, with events hosted regularly. Surprisingly, there is no formalized research conducted to access the growth of this industry. The industry includes fields such as the events, exhibitions, conferences and seminars as well as live music and sporting events. On the profession side, event management is a glamorous and exciting profession that demands a lot of hard work and dynamism. The logistics side of the industry is paid less than the sales/sponsorship side, though some may say that these are two different industries. Concerts and Shows Reservation is the application of project management to the creation and development of large-scale events. The process of planning and coordinating the event is usually referred toas event planning and which can include budgeting, scheduling, site selection, acquiring necessary permits, coordinating transportation and parking, arranging for speakers or entertainers, arranging decor, event security, catering, coordinating with third party vendors, and emergency plans. The events industry now includes events of all sizes from the Olympics down to business breakfast meetings. Many industries, charitable organizations, and interest groups hold events in order to market themselves, build business relationships, raise money, or celebrate achievement.

#### Motivation and Scope:

* + - To deploy a website which keeps track of all Concerts and Shows happening at a Venue. Thus, creating a transparent and a user-friendly website for Concerts and Shows Reservation Platform.
    - To reduce the redundancy of existing reservation systems.
    - The main objective of the project is to design and develop a user-friendly efficient computerized Concerts and Shows Reservation Platform.
    - Computerization can be helpful as a means of saving time & money.
    - Secured data storage for authority end.
    - To provide a platform which allows users to host and manage shows and concerts, and venue owners to host different events.
    - Ease of organizing, managing and hosting either shows or concerts.
    - To allow for users to reserve their presence to any of the shows or concerts happening at a Venue.

#### Problem Statement:

The objective of this application is to develop a system that effectively manages all the data related to the various events that take place in an organization. The purpose is to maintain a centralized database of all the event related information. The goal is to support various functions and processes necessary to manage the data efficiently.

This existing system is not providing secure registration and profile management of all the users properly. This system is not providing on-line help. This system doesn’t provide tracking of user’s activities and their progress. This manual system gives us very less security for saving data and some data may be lost due to mismanagement. This system is not providing event management through internet. This system is not providing proper events information. The system is giving manual information through the event management executer.

# CHAPTER 2: REQUIREMENT SPECIFICATIONS

#### Functional Requirement:

* + - **R1: Registration**

Description: To enter into this site user has to register himself first. Requirements of registration are first name, last name, user name, email-id, password, confirm password etc.

Input: User Details

Output: Filled Registration Details.

Processing User details are checked with database. Password constraint is checked as per validation.

#### R2: User Interface

Description: The System provides interface to the system.

Output: Display all Events & Venues

#### R3: Select the event

Description: The user can select the event and also select payment method. Input: Main event, Sub event, Enrollment number, Add team member.

Output: Event selected Successfully, also see all detail and delete also. Processing The system will add selected data into database.

#### R4: Admin panel

Description: The Admin can add manager, main event, sub-event also. Input: main event, sub-event, manager.

Output: Add successfully in database.

Processing: The system will add selected data into database.

#### Non-Functional Requirement:

**Performance**:

* + - Response time of the System should be less than 3 second most of the time.
    - Response time refers to the waiting time while the system accesses, queries and retrieves the information from the databases.

#### Reliability:

* + - It shall be available 24 hours a day, 7 days a week.
    - It shall always provide an accurate listing of the available venues and ongoing events.

#### Integrity:

* + - Only the system administrator has the right to change system parameters, such as deleting unwanted/spam Details concerned to users/organizations and event details.
    - Users need to be authenticated before having access to any data.

A Convenient dashboard, user-friendly UI, separation of working pages for user’s convenience, separate dashboards for Users and Admin is a must.

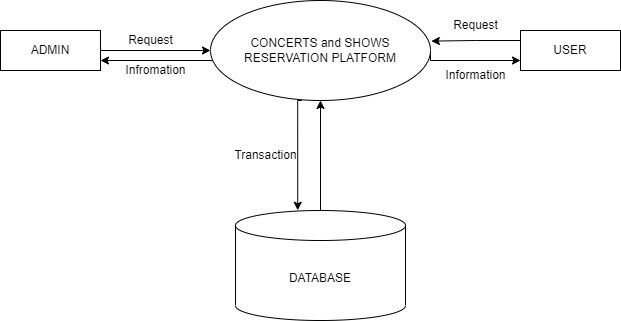
#### Domain Constraints:

Domain constraints are defined as the valid set of values of an attribute. In this project, we have used various domain constraints such as primary keys, foreign keys and restriction on the type of data stored in the table. The tables use the VARCHAR Data type to store strings and text values. INTEGER data type is used to store respective entity. Integrity constraints are also managed in this project. The Primary Key used is unique and does not repeat. The Foreign Key used is derived from an existing table and is used for a valid attribute to make connections to the tables and run the queries easily. Normalization, which is defined as the process of storing data in a database, was also used. Normalization up to 3 Normal Form was used so as to reduce redundancy.

* + - **Regulatory policies**: It is mandatory that no text box must contain insufficient data.
    - **Hardware limitations:** There must be a 64 MB on board memory.
    - **Control functions**: The software is user-friendly and displays appropriate error messages.
    - **Parallel operations**: It supports many users simultaneously.
    - **Safety/security considerations**: The application always exits normally.
    - **Software Requirement**: OS- Windows/Mac, Browser- Chrome/Mozilla Firefox
    - **Hardware Requirement:** Processor- 32, Memory- 4GB RAM

# CHAPTER 3: SYSTEM/REQUIREMENT ANALYSIS

#### Overall System Design:



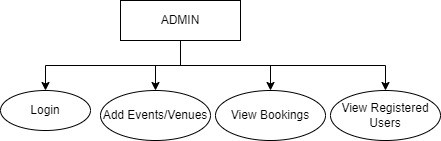
**Fig. 3.1:** System Design

The first step in the project was having a discussion about what functionalities we want to provide with our work. After getting a mutual understanding of what the end product might look like, we worked on the database design Fig 3.1. Three of us made our own database schema, compared them, and put together a final database schema that we all could agree upon, with parts of all our works to make what we deemed to be the most functionally accurate database.

The next step was frontend development (using HTML + CSS) and database creation (using MySQL), both of which were done simultaneously. The frontend was made with regular interaction with the backend-in-charge, to make sure it properly reflects the backend and at the same time is user-friendly.

After these were done, PHP was used to put the project together, some final touches were added, and was then hosted onto the internet via a local host using the XAMPP.

#### Admin Module:



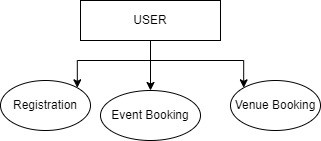
Updat

**Fig. 3.2:** Admin Module Design

As shown in Fig 3.2, Admin upon Login is redirected to the dashboard where the admin can view all the events, venues and the bookings made. Admin also can manage the users or audience for a particular event.

#### User Module:

#### As detailed out in Fig 3.3, a user can register himself/herself for an event. Request for booking an Event at a Venue. User also can book for a venue and the validation is done by the admin. User in our website can view all the ongoing/upcoming events in the front page and also have a look the venues depicted in the venues page.

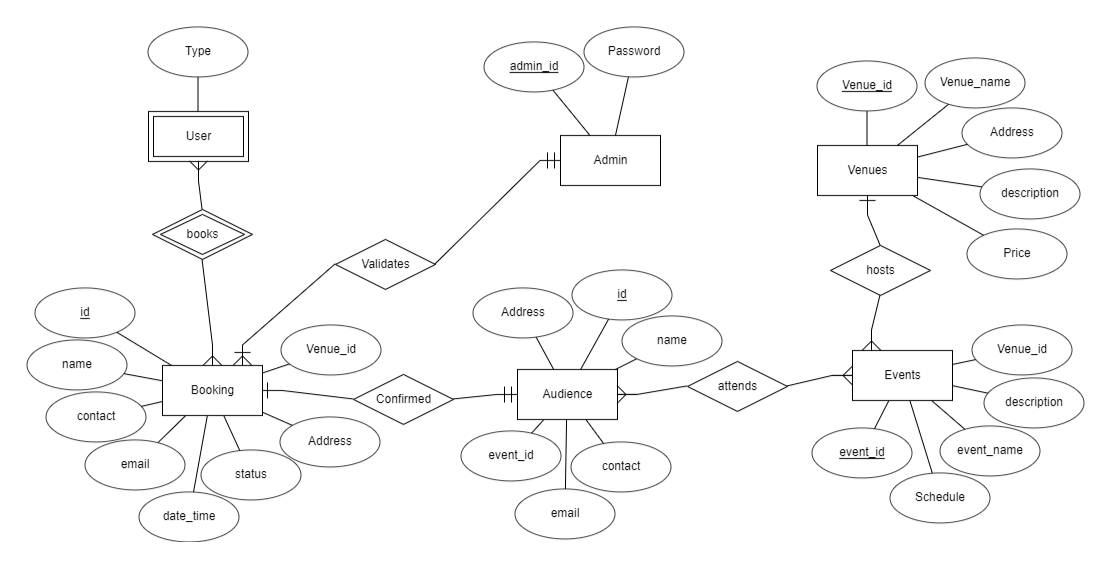


**Fig. 3.3:** User Module Design

# CHAPTER 4: SYSTEM DESIGN

## Entity Relationship Diagram:

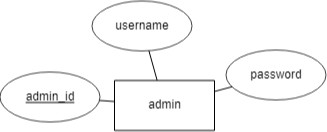
## ER Diagram stands for Entity Relationship diagram, also known as ERD is a diagram that displays the relationship of entity sets(objects) stored in a database. In this project database, the ER Diagram is depicted in Fig 4.1.



**Fig. 4.1:** Entity Relationship Diagram

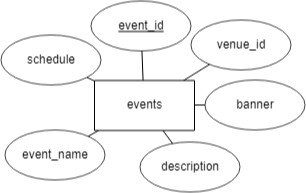
## Entity Sets:

### Admin



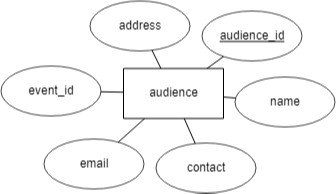
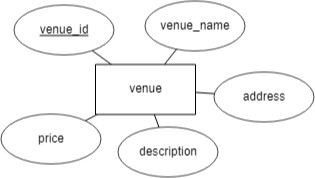
**Fig. 4.2:** Admin Entity Set

### Events  Booking



**Fig. 4.3:** Events Entity Set **Fig. 4.4:** Booking Entity Set

* Audience • Venue



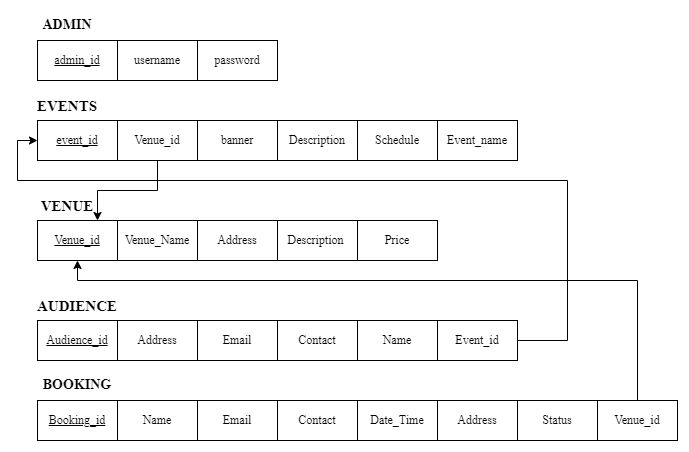
**Fig. 4.5:** Audience Entity Set **Fig. 4.6:** Venue Entity Set

As shown, there are 5 Entity Sets in our database. Admin Entity Fig 4.2 to store the admin details, Events Fig 4.3 to store list of events happening and respective venues, Booking Fig 4.4 to keep track of user bookings made, Audience Fig 4.5 to store details of all audience for an event and generate a report, Venue Fig 4.6 to store the venue details such as price and address of the venue.

#### Schema Diagram:

A database Schema defines how data is organized with a relational database; this is inclusive of logical constraints such as table names, fields, data types and the relationship between these entities. The Schema developed for the project is defined in Fig 4.7.

IMPLEMENTATION



**Fig. 4.7:** Schema Diagram

# CHAPTER 5: IMPLEMENTATION

#### Description of Database Tool (Backend)

The Database used was MySQL, MySQL is an open-source relational database management system (RDBMS) that works with an operating system to implement a relational database in a computer's storage system, manages users, allows for network access and facilitates testing database integrity and creation of backups. It is most noted for its quick processing, proven reliability, ease and flexibility of use. It is a stable, reliable and powerful solution with advanced features like: Data Security, High Performance, complete workflow control, flexibility of open source.

The MySQL Database Server is very fast, reliable, scalable, and easy to use. If that is what you are looking for, you should give it a try. MySQL Server can run comfortably on a desktop or laptop, alongside your other applications, web servers, and so on, requiring little or no attention.

The MySQL Database Software is a client/server system that consists of a multithreaded SQL server that supports different back ends, several different client programs and libraries, administrative tools, and a wide range of application programming interfaces (APIs). It also provides MySQL Server as an embedded multithreaded library that you can link into your application to get a smaller, faster, easier-to-manage standalone product.

The MySQL Database Server is very fast, reliable, scalable, and easy to use. If that is what you are looking for, you should give it a try. MySQL Server can run comfortably on a desktop or laptop, alongside your other applications, web servers, and so on, requiring little or no attention.

PHP is a server-side scripting language designed primarily for web development but also used as a general-purpose programming language. Originally created by Rasmus Lerdorf in 1994, the PHP reference implementation is now produced by The PHP Development Team. PHP originally stood for personal home page but it now stands for the recursive acronym PHP: Hypertext Preprocessor.

PHP code may be embedded into HTML or HTML5 code, or it can be used in combination

with various web template systems, web content management systems and web frameworks. PHP code is usually processed by a PHP interpreter implemented as a module in the web server or as a Common Gateway Interface (CGI) executable. The web server combines the results of the interpreted and executed PHP code, which may be any type of data, including images, with the generated web page. PHP code may also be executed with a command-line interface (CLI) and can be used to implement standalone graphical applications.

#### Description of Implementation (Frontend)

For front-end development, Visual Studio Code was used. Visual Studio Code is a free source-code editor made by Microsoft for Windows, Linux and macOS. Features include support for debugging, syntax highlighting, intelligent code completion, snippets, code refactoring, and embedded Git.

Visual Studio Code is a lightweight but powerful source code editor which runs on your desktop and is available for Windows, macOS and Linux. It comes with built-in support for JavaScript, TypeScript and Node.js and has a rich ecosystem of extensions for other languages (such as C++, C#, Java, Python, PHP, Go) and runtimes (such as .NET and Unity).

#### HTML5:

#### Hypertext Markup Language revision 5 (HTML5) is markup language for the structure and presentation of World Wide Web contents. HTML5 supports the traditional HTML and XHTML style syntax and other new features in its markup, New APIs, XHTML and error handling.

#### CSS:

Cascading Style Sheets (CSS) is a style sheet language used for describing the presentation of a document written in a markup language such as HTML. CSS is a cornerstone technology of the World Wide Web, alongside HTML and JAVASCRIPT.

#### JAVASCRIPT:

JavaScript is a programming language that started off simply as a mechanism to add logic and interactivity to an otherwise static Netscape browser.

#### MySQL:

MySQL is an open-source relational database management system (RDBMS). Its name is a combination of "My", the name of co-founder Michael Widenius' daughter, and "SQL", the abbreviation for Structured Query Language. The MySQL development project has made its source code available under the terms of the GNU General Public License, as well as under a variety of proprietary agreements. MySQL was owned and sponsored by a single for-profit firm, the Swedish company MySQL AB, now owned by Oracle Corporation.

# CHAPTER 6: TESTING

#### Component Tests:

Component testing is undertaken when a module has been created and has successfully reviewed.

Each component of the software was tested individually from the Login Page Table 6.1 for Admin and the Events/Venues Page Table 6.2. The Login Test was Authenticated for an admin. The Events Page Test consisted of adding new events/Venues and verifying whether the details of the events and venues were displayed.

**Table 6.1**: Login-Page Test

|  |  |
| --- | --- |
| Sl No. of test case: | 1 |
| Name of test: | Login test |
| Sample Input: | Click on login that runs admin.php |
| Expected output: | login GUI should display |
| Actual output: | login GUI displayed as expected |
| Remarks: | Test Successful |

**Table 6.2:** Events-Page Test

|  |  |
| --- | --- |
| Sl. No. of test case: | 2 |
| Name of test: | Check Event page test case |
| Sample Input: | Clicking the Check Event page that runs  Events.php |
| Expected output: | All Event details are viewed. |
| Actual output: | All Event details are viewed. |
| Remarks: | Test case successful |

#### System Test:

The whole system testing was done to evaluate the efficient working of software. All the bugs that were found were sorted out.

The Project went through two levels of testing:

#### : Unit Testing:

Unit Testing is a type of software testing where individual units or components of a software are tested. The purpose is to validate that each unit of the software code performs as expected. Unit Testing is done during the development (coding phase) of an application by the developers.

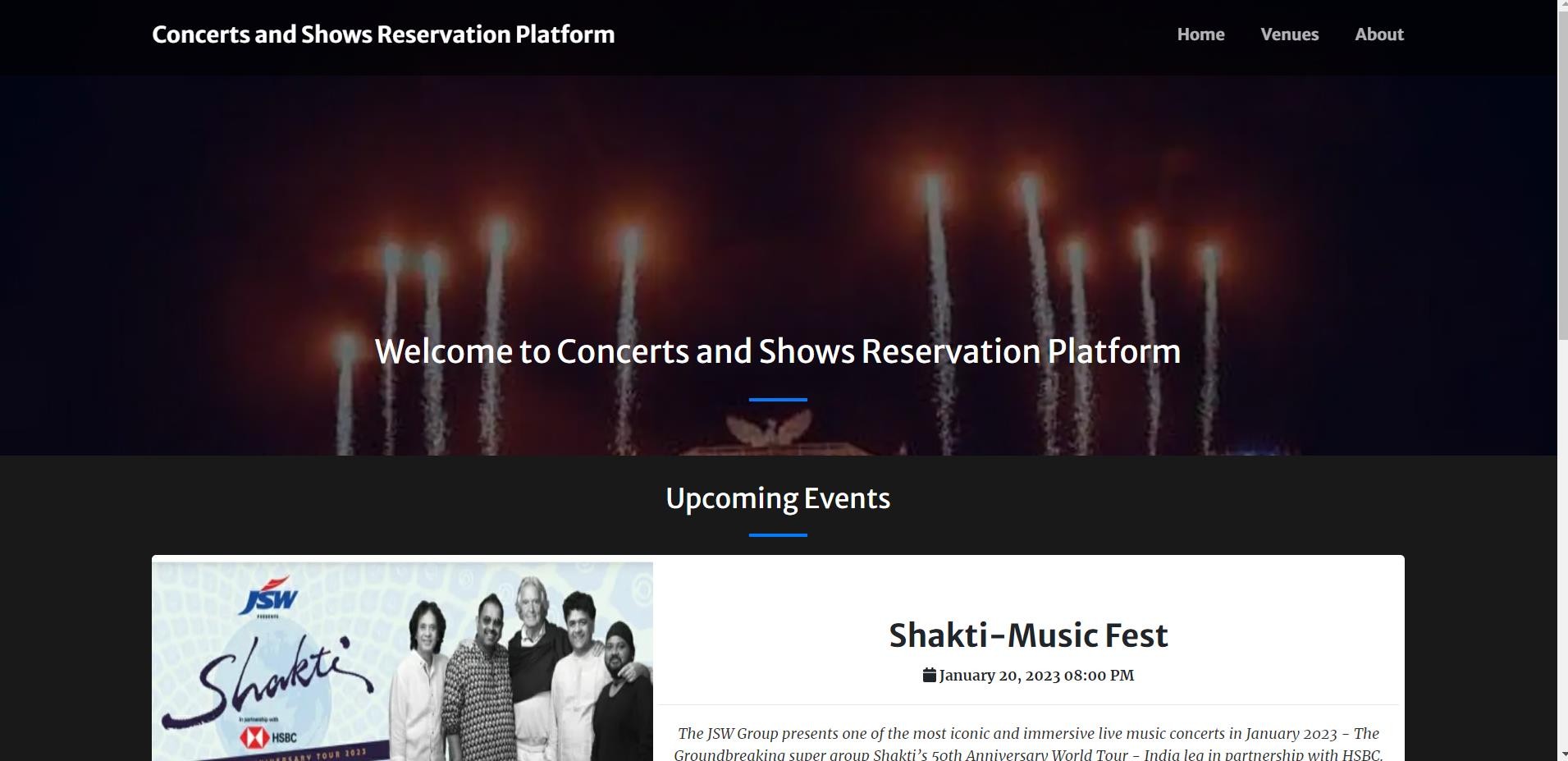
In the website the Events.php and Venues.php were the two main components and were tested individually.

#### :Integration Testing:

Integration Testing is defined as a type of testing where software modules are integrated logically and tested as a group. A typical software project consists of multiple software modules, coded by different programmers. The purpose of this level of testing is to expose defects in the interaction between these software modules when they are integrated.

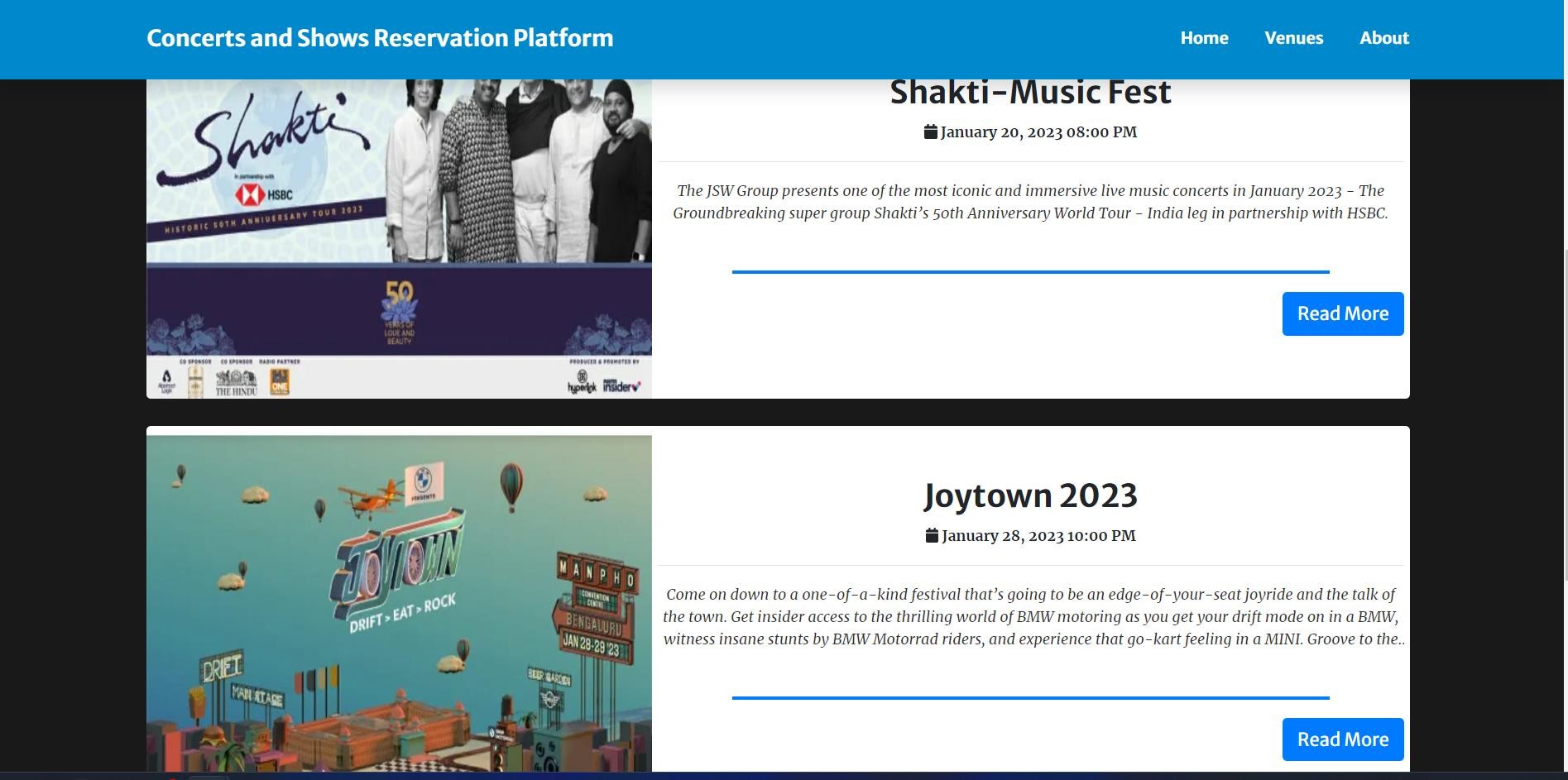
The whole page was integrated and checked against dynamic changes to the website, i.e. the events and venues requested by user were validated and changes were made apparent in the respective front-end pages of the website.

# CHAPTER 7: INTERPRETATION OF RESULTS



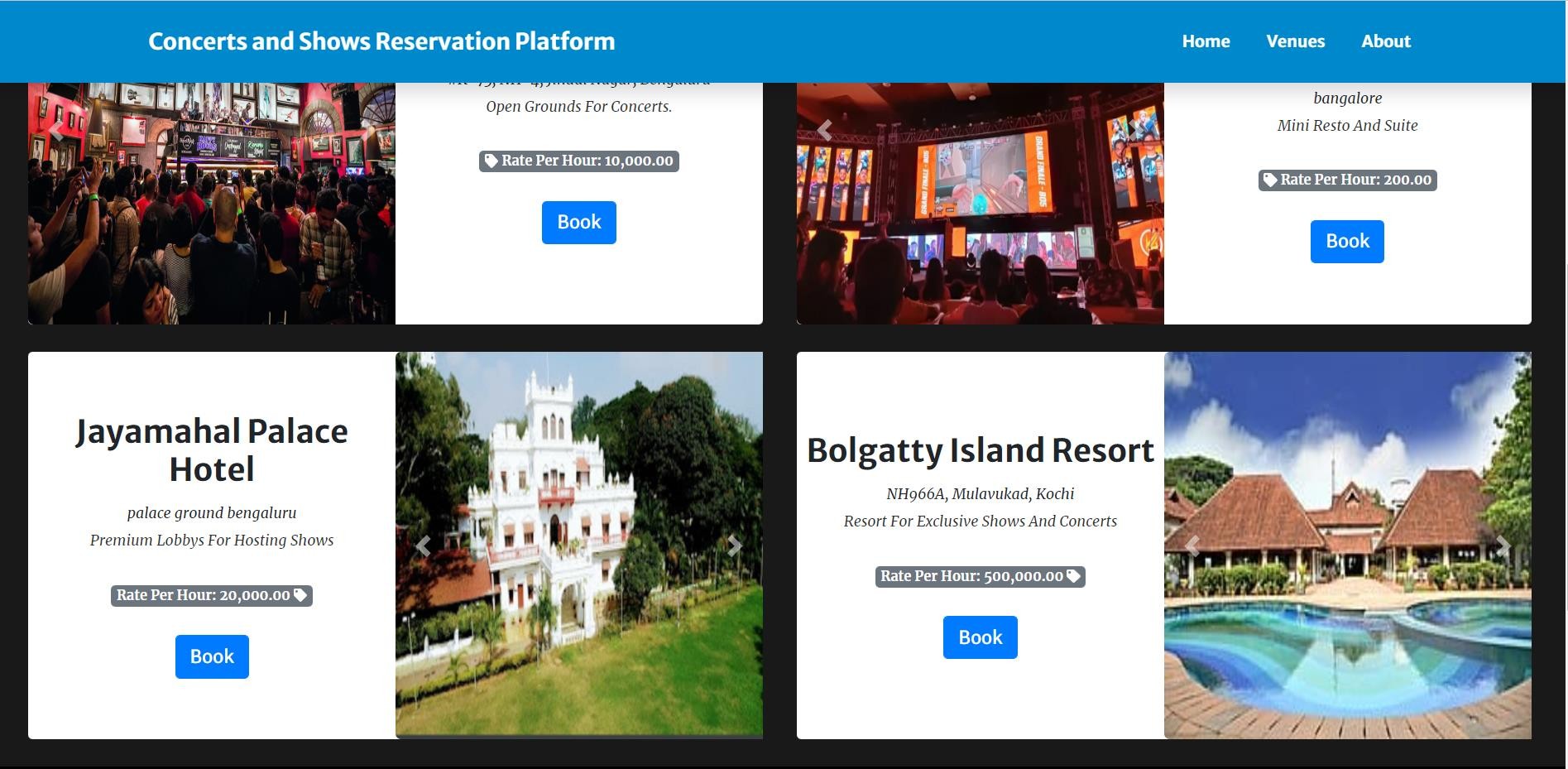
**Fig. 7.1 :** Home Page of Website

The home page of the website which shows a navigation to other parts of the website as shown in Fig 7.1.



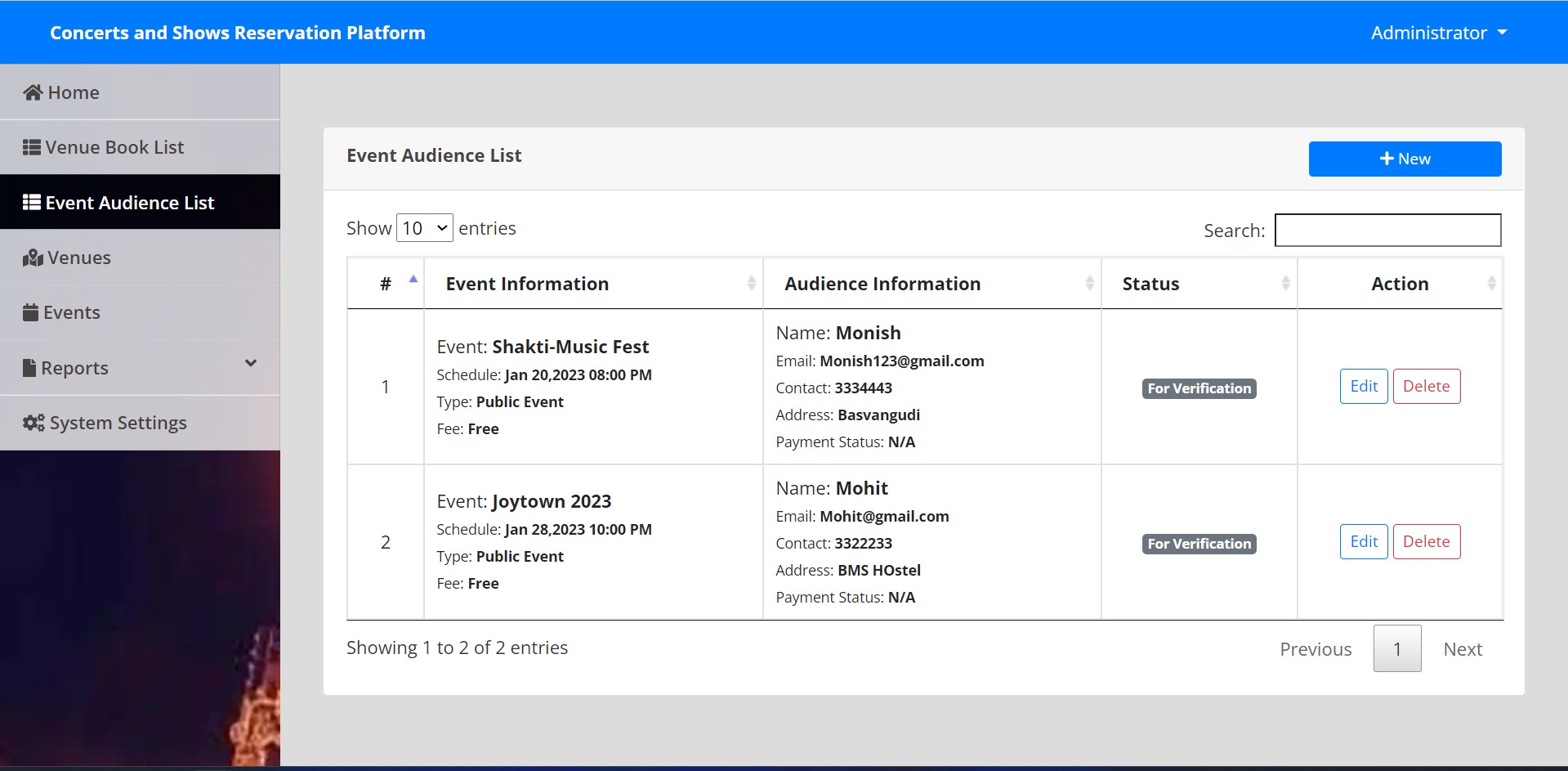
**Fig 7.2 :** Upcoming Shows and Concerts

All the upcoming events are displayed in later section of the home page Fig 7.2. Users can know the dates of the shows or concerts and Venues of Events.



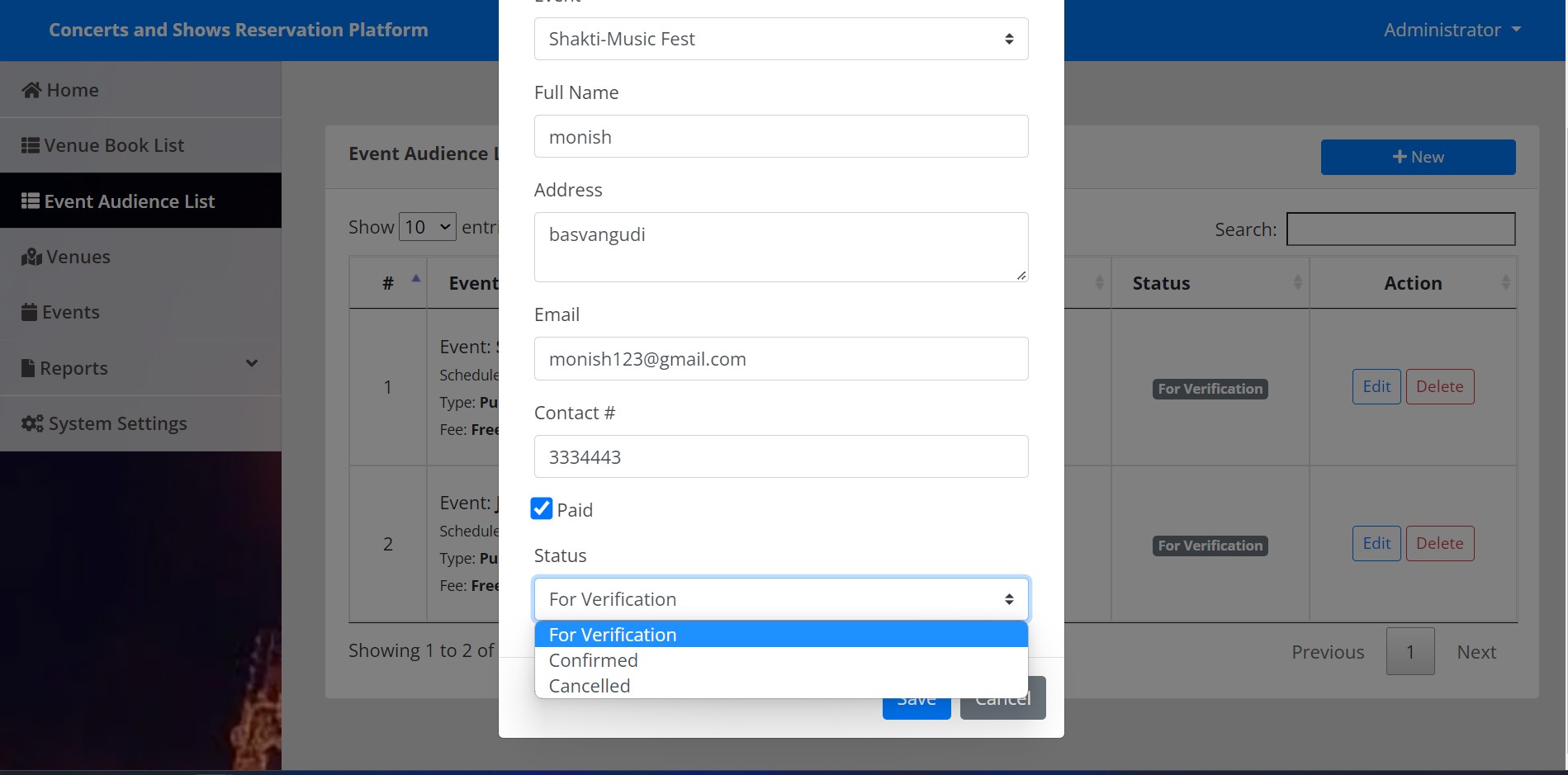
**Fig 7.3 :** List of all Venues

All the registered Venues are displayed in the Venues page Fig 7.3, Users can see the price of accommodation at a venue and book a venue.



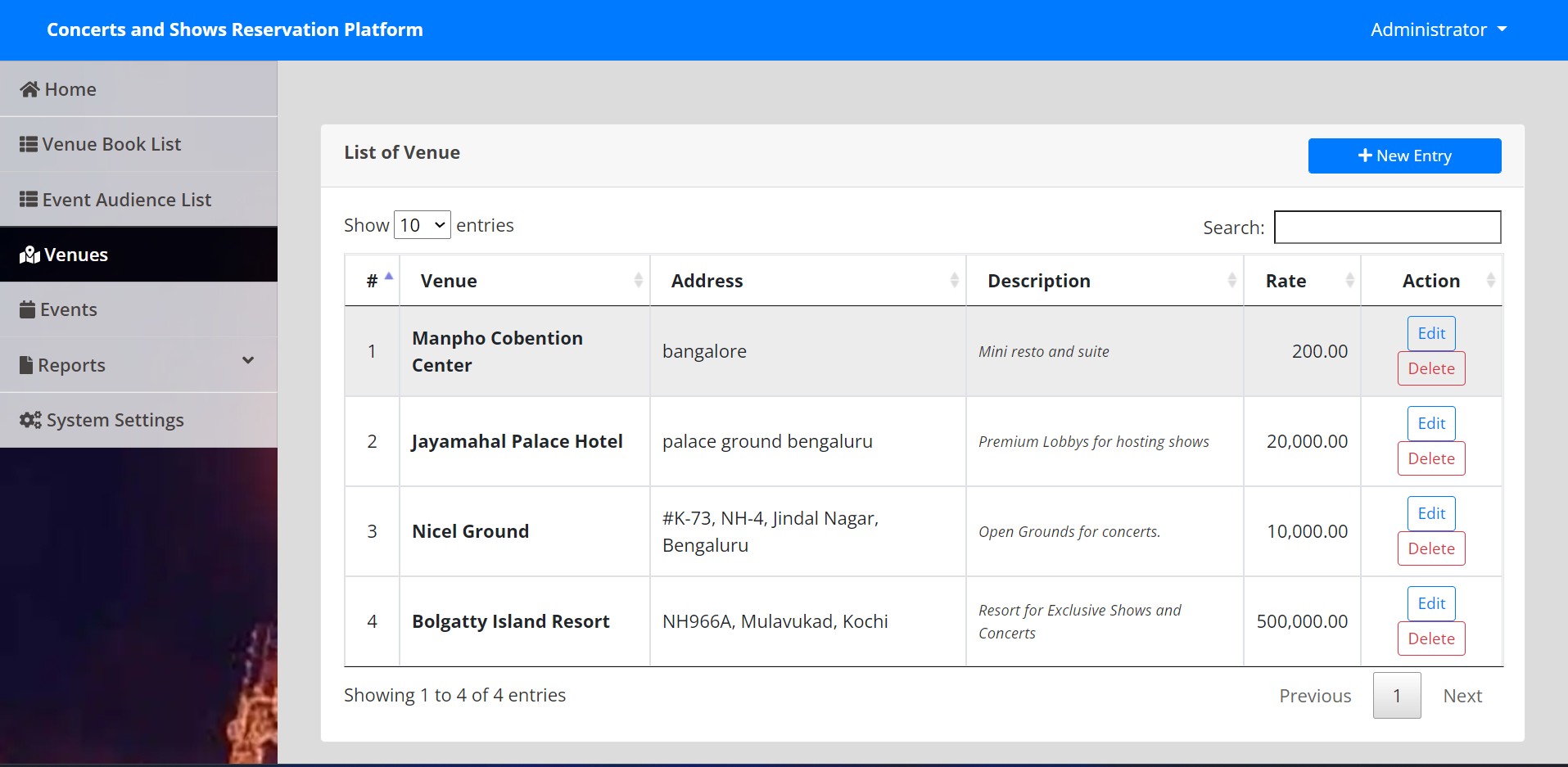
**Fig 7.4 :** View of Audience Reservations

The Admin can manage the audiences of a concert or a show in this tab depicted in Fig 7.4.



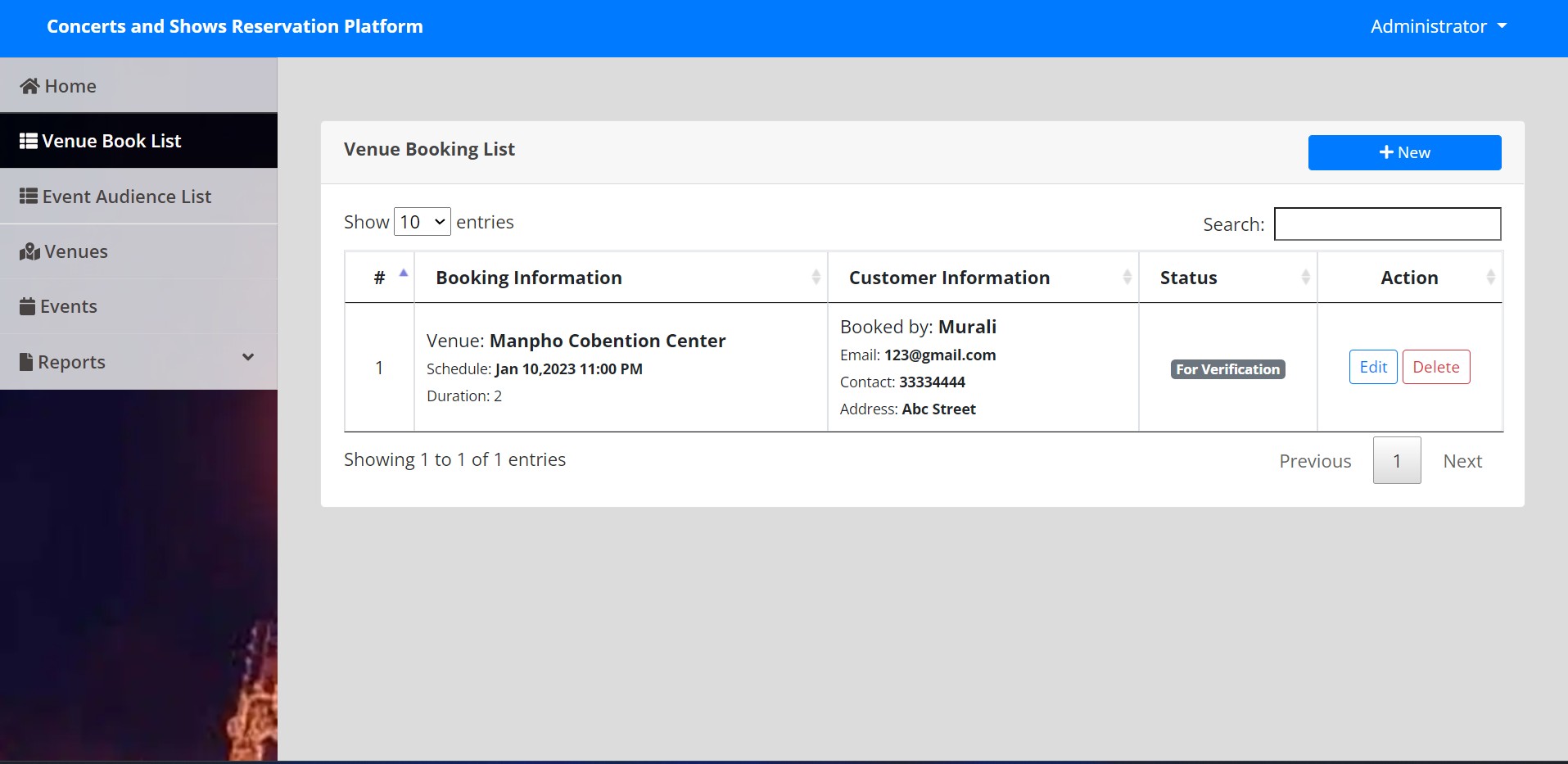
**Fig 7.5 :** Approval of Reservation by Admin

Admin can Verify Fig 7.5 whether the user is eligible to attend the event by verifying the details of the users.



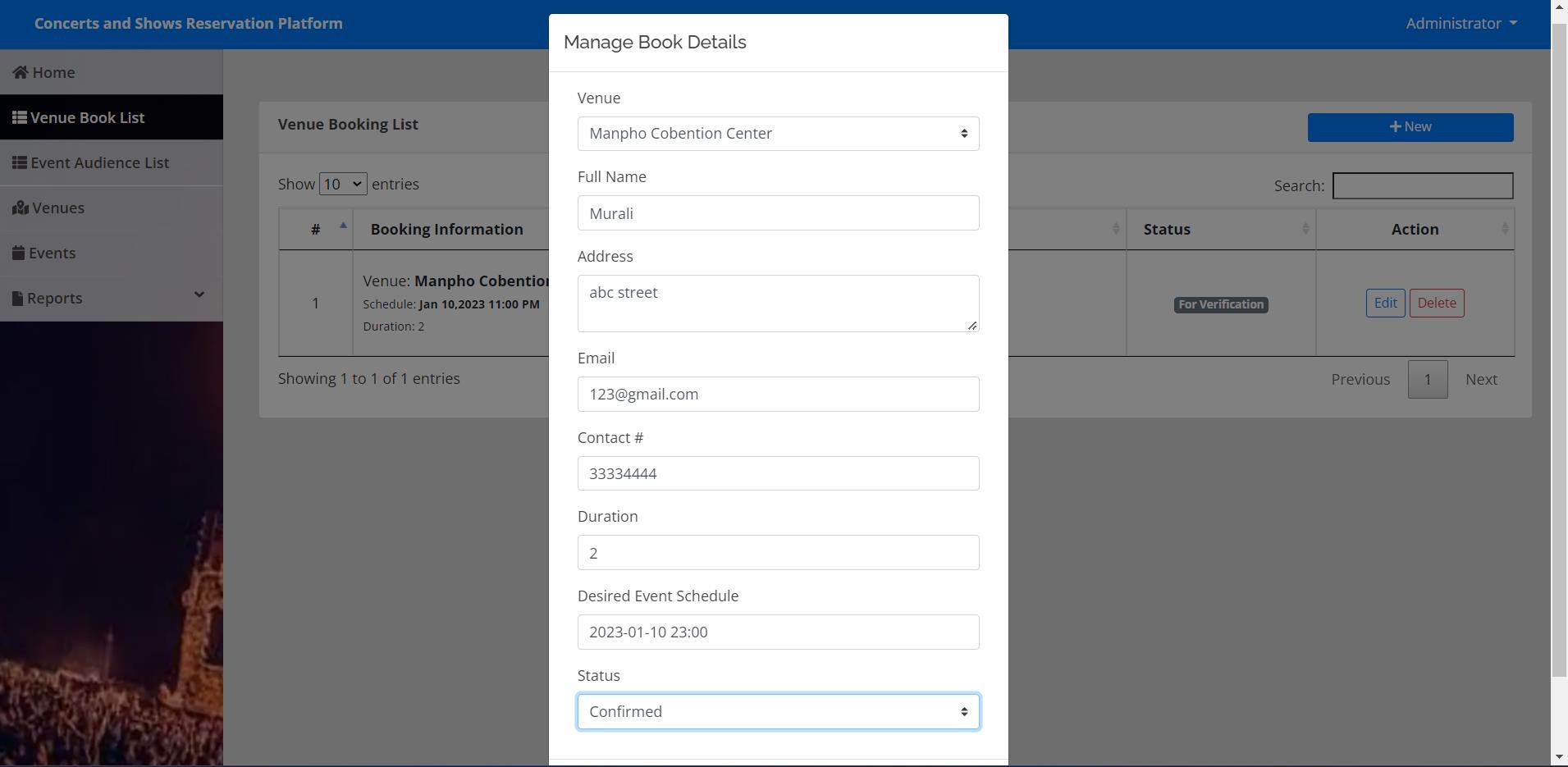
**Fig 7.6 :** Reservations of Venues

Venue Owners can request the admin to add their venues for hosting concerts and shows, the admin can confirm the venue list and add the venue to the list of venues Fig 7.6.



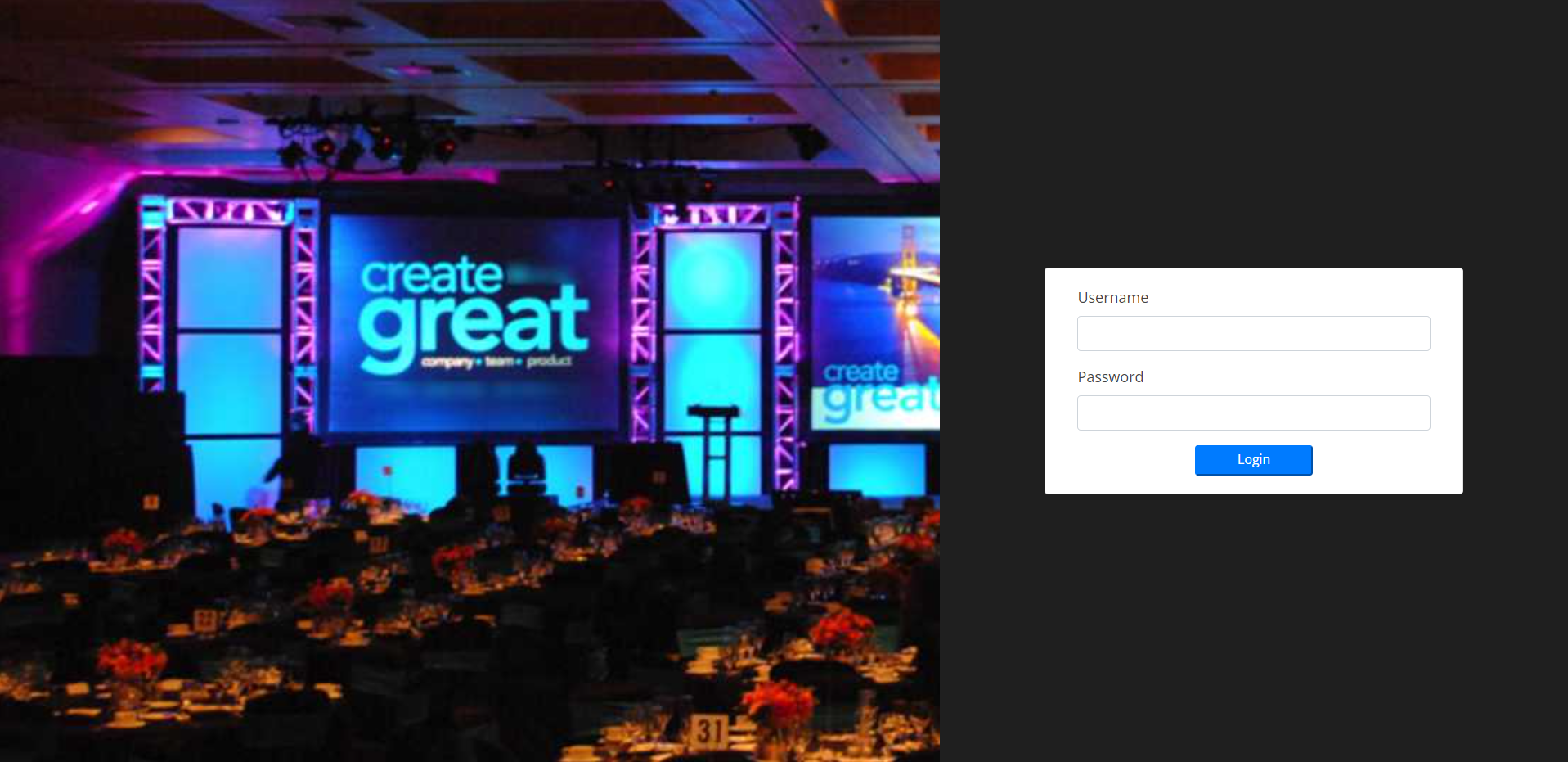
**Fig 7.7 :** View of Reservation for venues

Admin can View the reservation for venues in the dashboard Fig 7.7.



**Fig 7.8 :** Approval of Venue by Admin

Admin can approve Fig 7.8 venues and the venue will be displayed at the Venue Page Fig 7.3.



**Fig 7.9:** Admin Login Panel

Admin can Login Fig 7.9 into the dashboard to manage the Shows, Concerts and the users or Audiences.

# CONCLUSION

Thus, we have successfully implemented concerts and show reservation database management which helps us in centralizing the data used for managing the tasks such as reservation booking, events and venue booking, we have successfully implemented various functionalities of MySQL and created the fully functional database management system for concerts and shows bookings.

Event Management System is user friendly and cost-effective system, it is customized with activities related to event management life cycle.

It provides a new edge to management industry. The Concerts and Show Reservation was successfully designed and is tested for accuracy and quality. During this project we have accomplished all the objectives and this project meets the needs of the organization. The developed will be used in searching, retrieving, and generating information for the concerned requests.

**GOALS ACHIEVED**

* Reduced entry work.
* Easy retrieval of information.
* Reduced errors due to human intervention.
* User friendly screens to enter the data.
* Portable and flexible for further enhancement
* Web enabled.
* Fast finding of information request

# REFERENCES

1. Fundamentals of Database Systems, Ramez Elmasri and Shamkant B. Navathe, 7th Edition, 2017, Pearson.
2. Database management systems, Ramakrishnan, and Gehrke, 3rd Edition, 2014, McGraw Hill.
3. Learning PHP, MySQL & JavaScript: A Step-by-Step Guide to Creating Dynamic Websites, Robin Nixon, 6th Edition.
4. PHP and MySQL Web Development, Luke Welling, Laura Thomsan, 5th Edition.

# 

