

# **CHAPTER -1**

## **INTRODUCTION**

---

### **1.1 Need of the new system**

- In today's life, people have many options to enjoy weekends but with many options comes more trouble to choose places nearby.
- It is very time consuming and complex to search such events near surrounding.
- For this people have to gather information as such they don't know about the events occurring nearby.
- This web application will help people to overcome from such issues and search an immediate result of events occurring nearby.

### **1.2 Detailed problem definition**

- Here people can get the alternative to search for their choice of favorite events which attracts their interest.
- List of events available on our site will be fully authorized and safe as per today's condition.
- New system will add the better reviews for people who have already experienced our organized events.
- Even people will get a chance to Register online easily as per the quote says first comes first serve.

### **1.3 Viability of the system**

- In our website people can easily check or go through the events happening nearby on a click.
- It will be a user friendly website.

- No payment charges for any user.

#### **1.4 Presently Available Systems for the same**

- <https://insider.in/online-events-india>



Figure 1

- <https://www.whatshot.in/>



Figure 2

#### **1.5 Future Prospects**

- More categories will be added in upcoming time.
- More areas will covered in nearby future.
- Online payment methods will be included.

## **2.1 Requirement Analysis**

Aim of this website is to understand the exact requirements of the customer and to document them properly. It will also reduce the communication gap between developers and customers.

Types of users in our new system are:-

- Admin
- Client
- Event manager

### **Admin:**

- Admin directs the whole website.
- Looks after the website on regular basis and can make changes.
- They ensure security measures for the users.
- Troubleshoot issues and the outages.
- Incharge of all the panels provided.
- They generate the backups on daily basis.

### **Event manager:**

- Manages the event panel accordingly.
- Notifies about the upcoming events.
- Terms and conditions of the events.
- Posting about the regular events.

**Client:**

- They must register themselves to the websites.
- Can access areas and features of the website.
- Access to map for locating nearby events.
- Provides feedback system.

**2.2 Project Model**

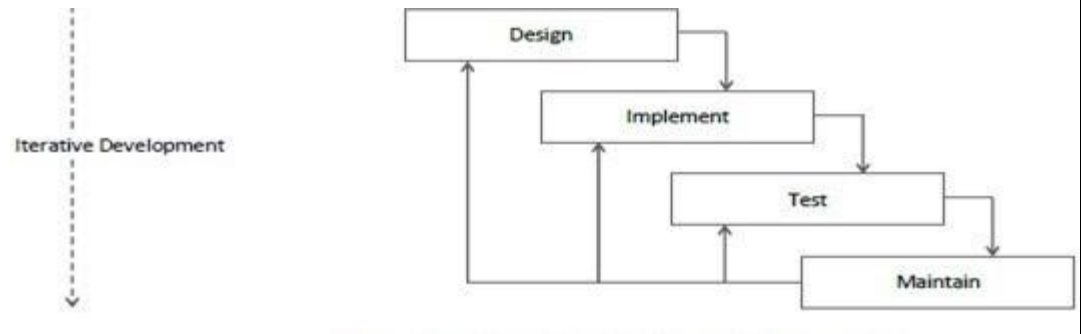
- **Iterative Waterfall Model**

The Iterative waterfall model provides feedback paths from every phase to its preceding phases.

In the Iterative model, iterative process starts with a simple implementation of a small set of the software requirements and iteratively enhances the evolving versions until the complete system is implemented and ready to be deployed.

An iterative life cycle model starts with specifying and implementing part of the software, which is then reviewed to identify further requirements.

- **Iterative Waterfall Model**



[Figure 1: Iterative Waterfall Model]

- **ADVANTAGES**

- The model is more flexible and less costly to change the scope and requirement.
- User gets a chance to experiment with partially developed software.
- This model helps finding exact user requirement.
- Feedback providing at each increment is useful for determining the better final product.

## 2.3 Schedule Representation

Generalized project scheduling tools and technique can be applied with little modification to software projects. Project evolution and review technique and critical paths method are two project scheduling method that can be applied to software development. Both techniques are driven by information already developed in earlier project planning activities.

[Table 1: Schedule Representation]

ACTIVITY	START DATE	FINISH DATE
Requirement Analysis	01-07-2021	15-08-2021
System Analysis	16-08-2021	22-09-2021
System Design	23-09-2021	
System Coding		
Testing and Integration		

## 2.4 Feasibility Study

### 1. Economical Feasibility:

The system being is economic with respect to client or software development company point of view. It will not take any extra charges or high rates from clients for registration but, for event organizer company charges would be taken.

### 2. Technical Feasibility:

This system will be technically feasible as it runs on PHP which is open source so one will not have to take/pay any licensing.

### **3. Legal feasibility:**

This system will be legally feasible as it does not have any functionality that is performed without any permission or illegally.

### **4. Environmental feasibility:**


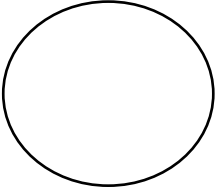
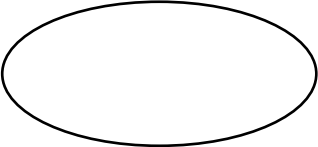


This system is environmentally feasible as it does not require any type of resources that harms nature or human as it runs on server.

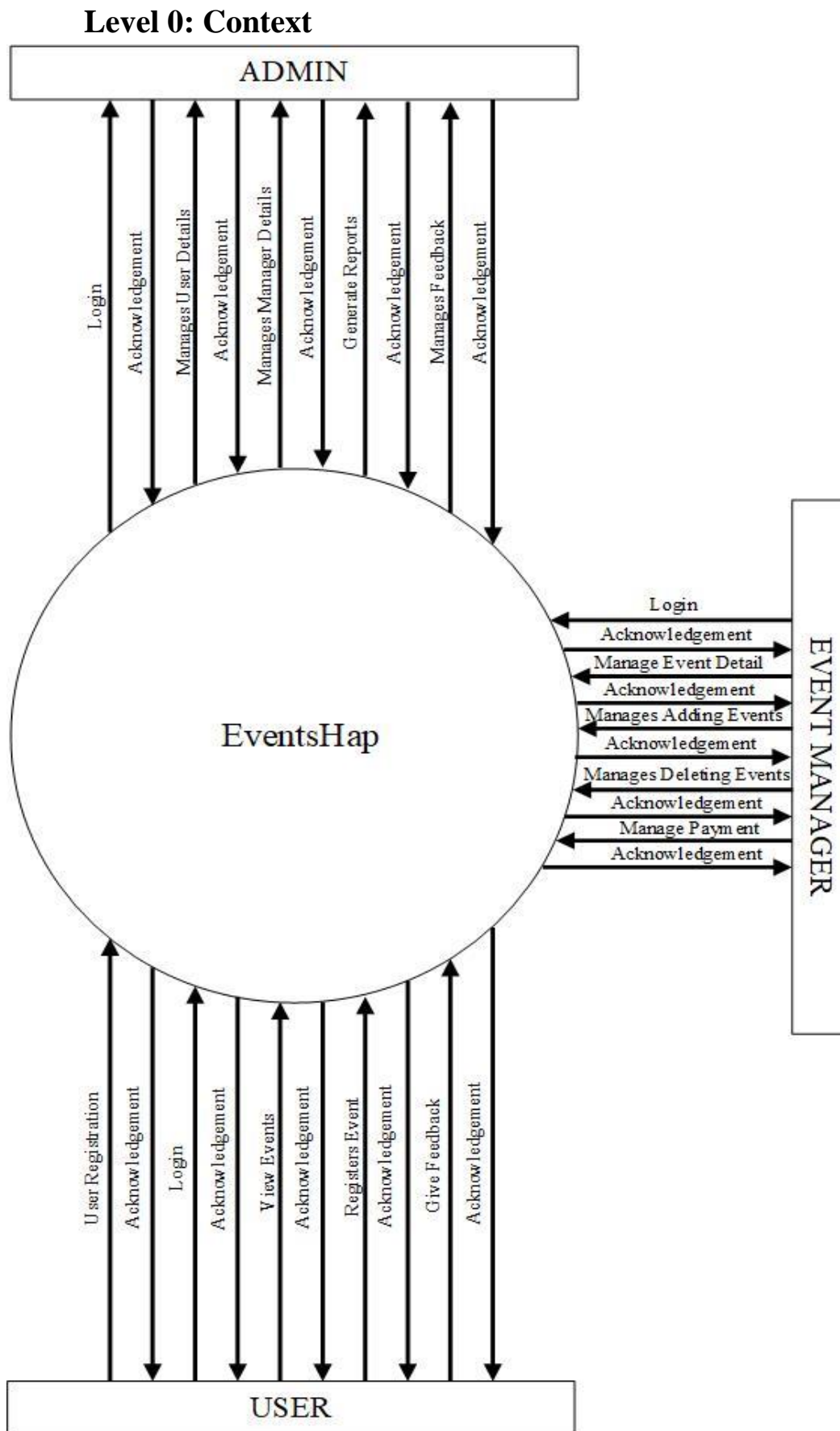
### **3.1 Data Flow Diagram**

- DFD (data flow diagram) is also known as bubble chart or data flow graph.
- DFD's are very useful in understanding the system and can be effectively used during analysis. It shows flow of data through a system visually. The DFD is a hierarchical graphical model of a system the different processing activities or functions that the system performs and the data interchange among these functions.
- It views a system as a function that transforms the inputs into desired output.
- Each function is considered as a process that consumes some input data and produces some output data.
- Function model can be represented using DFD.
- DFD graphically representing the functions, or processes, which capture, manipulate, store, and distribute data between a system and its environment and between components of a system.
- The visual representation makes it a good communication tool between User and System designer.
- Structure of DFD allows starting from a broad overview and expand it to a hierarchy of detailed diagrams.
- DFD has often been used due to the following reasons:
  1. Logical information flow of the system.
  2. Determination of physical system construction requirements.
  3. Simplicity of notation.
  4. Establishment of manual and automated systems requirements.



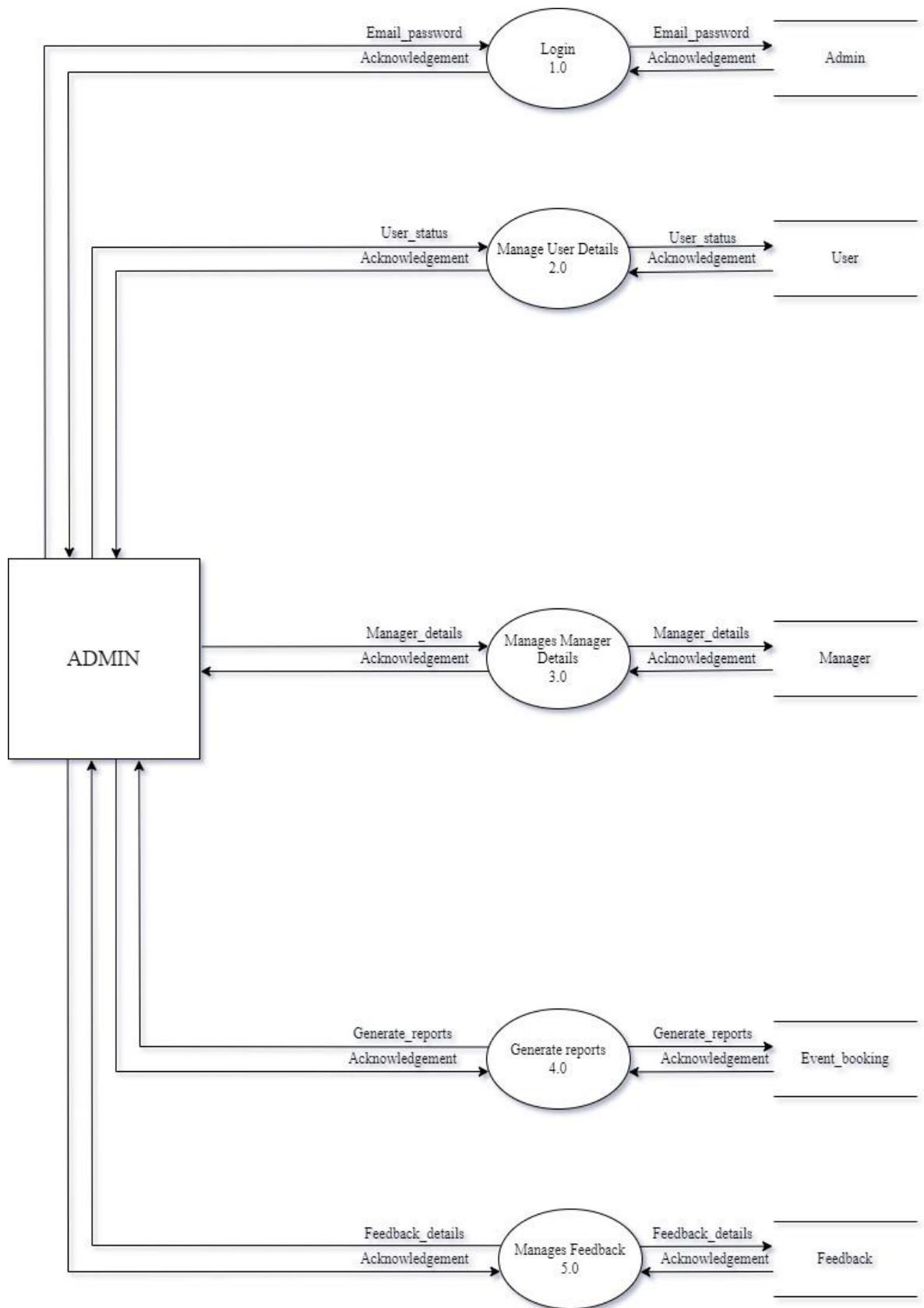
[Table 2: Data Flow Diagram Symbols]

Symbols	Description
	<b>Entity:</b> Entities are external to the system which interacts by inputting the data.
	<b>System:</b> It shows the system name.
	<b>Process:</b> It shows the part of the system that transforms into outputs.
	<b>Data Flow:</b> It passes the data from one part to another.
	<b>Data Store:</b> Data store is represented by two parallel lines. It is generally logical file or database.



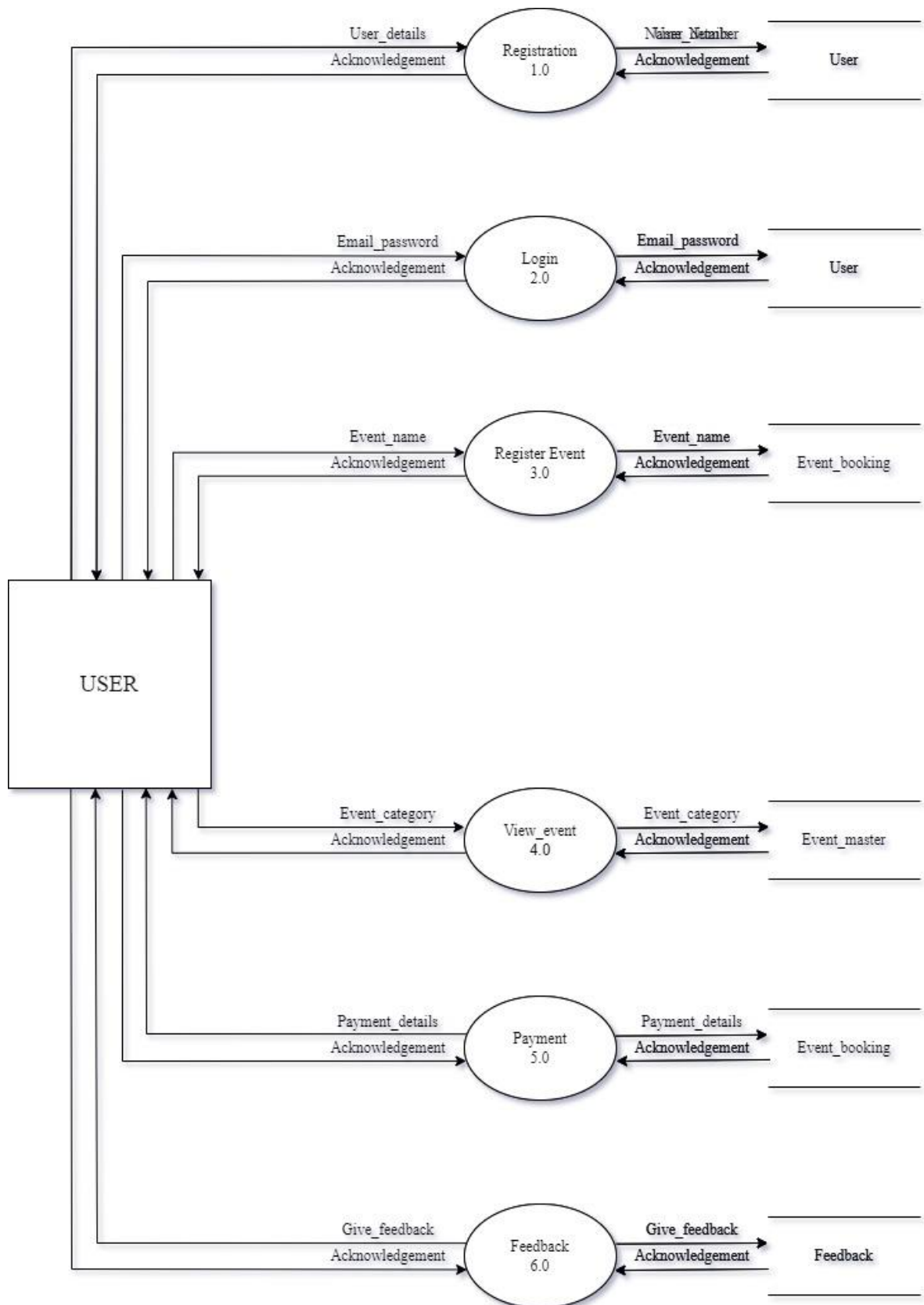
[Figure 2: CONTEXT LEVEL]

## Level 1: ADMIN



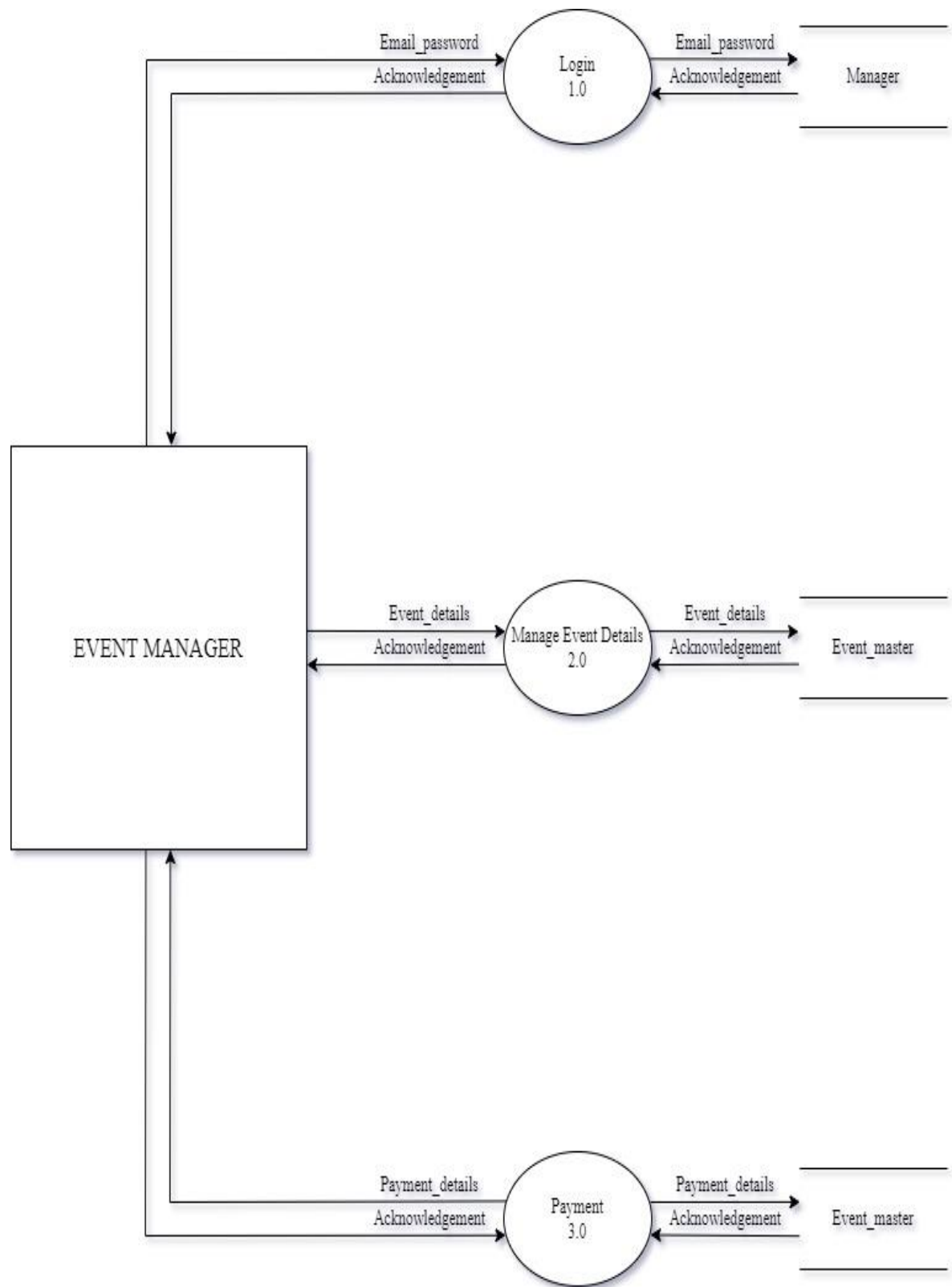
[Figure 3: DFD Level 1: Admin]

## Level 1: USER



[Figure 4: DFD Level 1: User]

## Level 1: EVENT MANAGER


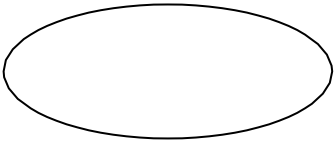
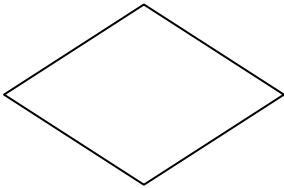
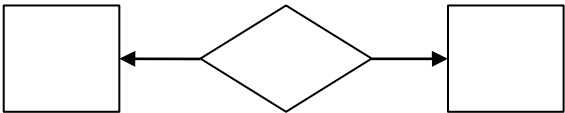


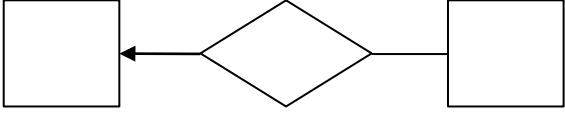
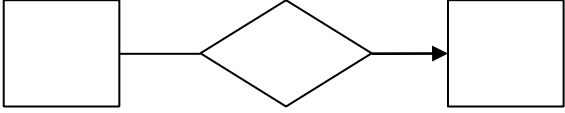
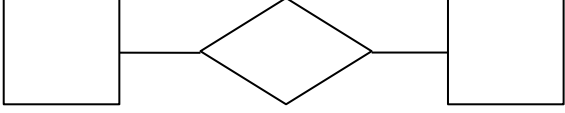
[Figure 5: DFD Level 1: Event Manager]

### 3.2 ER-Diagram

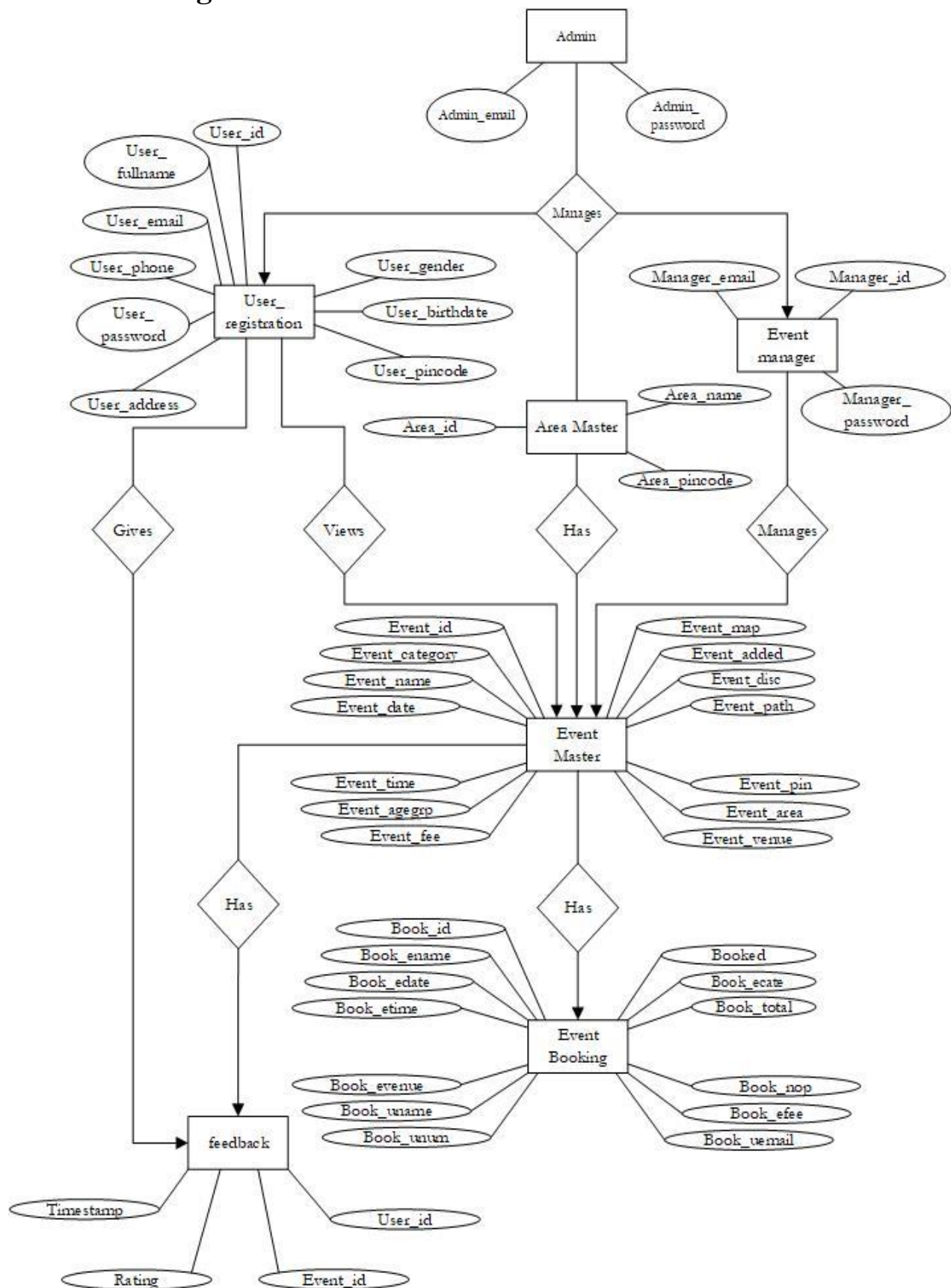
An Entity Relationship (ER) Diagram is a type of flowchart that illustrates how “entities” such as people, objects or concepts relate to each other within a system. ER Diagrams are most often used to design or debug relational databases in the fields of software engineering, business information systems, education and research. Also known as ERDs or ER Models, they use a defined set of symbols such as rectangles, diamonds, ovals and connecting lines to depict the interconnectedness of entities, relationships and their attributes. They mirror grammatical structure, with entities as nouns and relationships as verbs.

[Table 3: ER-Diagram Symbols]

Symbols	Description
	<b>Entity:</b> Data object is real world entity or thing. It is represented by a rectangle shape. An entity is an object or concept about which you want to store information.
	<b>Attributes:</b> An attribute is property of characteristic of an entity. It is represented by oval shape.
	<b>Relationship:</b> Entity are connected each other via relations. Generally, relationships in binary because there are two entities are related to each other.
	<b>Cardinality (One to One):</b> An instance of entity A can relate to one instances of entity B.

	<p><b>Cardinality (One to Many):</b> An instance of entity A can relate to one or many instances of B but we can only relate one instance of A.</p>
	<p><b>Cardinality (Many to One):</b> One or more instances of entity A can relate to one instances of B.</p>
	<p><b>Cardinality (Many to Many):</b> One or more instances of entity A can relate to one more instance of entity B.</p>

## ER-Diagram:





[Figure 6: ER Diagram]

## CHAPTER-4

### DATA DICTIONARY

---

#### 4.1 Database Dictionary

##### 1. Table Name: Admin

[Table 4: Admin]

SR.NO	FIELD NAME	DATATYPE (SIZE)	CONSTRAINT	DESCRIPTION
1	Admin_email	Varchar (100)	Primary key	Store admin's email
2	Admin_ password	Varchar (18)	Notnull	Store admin's password
3	Admin_name	Char(10)	Notnull	Store admin's name

## 2. Table Name: User Registration

[Table 5: User Registration]

SR.NO	FIELD NAME	DATATYPE (SIZE)	CONSTRAINT	DESCRIPTION
1	User_id	Int (11)	Primary key	Store user's Id
2	User_name	Char (50)	Not null	Store user's first name
3	User_contact	Bigint (10)	Unique key	Store user's phone number
4	User_email	Varchar (100)	Unique Key	Store user's email
5	User_password	Varchar (18)	Not null	Store user's password
6	User_cpassword	Varchar (18)	Not null	Store user's confirm password
7	User_address	Varchar (100)	Not null	Store user's address
8	User_pincode	Int (6)	Not null	Store user's pin code
9	User_birthdate	Varchar (8)	Not null	Store user's DOB
10	User_gender	Char (20)	Not null	Store user's gender

### 3. Table Name: Manager Login

[Table 6: Manager Login]

SR.NO	FIELD NAME	DATATYPE (SIZE)	CONSTRAINT	DESCRIPTION
1	Manager_id	Int (10)	Primary key	Store Manager's id
2	Manager_email	Varchar (100)	Not null	Store Manager's email
3	Manager_password	Varchar (20)	Not null	Store Manager's password

### 4. Table Name: Event Master

[Table 7: Event Master]

SR.NO	FIELD NAME	DATATYPE (SIZE)	CONSTRAINT	DESCRIPTION
1	Event_id	Int (5)	Primary key	Store Event's Id
2	Event_name	Varchar (50)	Not null	Store Event's Name
3	Event_category	char (50)	Not null	Store Event's category
4	Event_date	Date	Not null	Store Event's date
5	Event_venue	Varchar (200)	Not null	Store Event's venue
6	Event_pin	Int (6)	Not null	Store Event's

				pin code
7	Event_agegrp	Int (3)	Not null	Store Event's age group
8	Event_fee	Int (5)	Not null	Store Event's fee
9	Event_time	Varchar (100)	Not null	Store Event's time
10	Event_path	Varchar (100)	Not null	Store Event's Registration fees
11	Event_added	timestamp	Current_timestamp	Store Event Time When added
12	Event_map	Varchar(500)	Not null	Store Event's map
13	Events_desc	Varchar (200)	Not null	Store Event's description
14	Event_area	Char (10)	Not null	Store Event's Area

## 5. Table Name: Area

[Table 6: Area]

SR.NO	FIELD NAME	DATATYPE (SIZE)	CONSTRAINT	DESCRIPTION
1	Area_id	Int(9)	Primary Key	Store area id
2	Area_name	Varchar (20)	Not Null	Store area name
2	Area_pincode	Int (6)	Unique Key	Store area pin code

## 6. Table Name: Event Booking

[Table 7: Event Booking]

SR.NO	FIELD NAME	DATATYPE (SIZE)	CONSTRAINT	DESCRIPTION
1	Book_id	Int (8)	Primary Key	Store booking Id
2	Book_ename	Char (20)	Not null	Store Event name
3	Book_edate	Date	Not null	Store Event Date
4	Book_etime	Time	Not null	Store Event Time
5	Book_evenue	Varchar (200)	Not null	Store Event Venue
6	Book_uname	Char (30)	Not null	Store User name
7	Book_unum	Bigint (10)	Not null	Store User Number
8	Book_uemail	Varchar (100)	Not null	Store User Email
9	Book_efee	Bigint (10)	Not null	Store booking Fee
10	Book_nop	Int (3)	Not null	Store total Number of People
11	Book_total	Bigint (10)	Not null	Store booking Total fee
12	Book_ecate	Char (20)	Not null	Store Event category
13	Booked	Timestamp	Current_timestamp	Store Booking Time

## 7. Table Name: Feedback

[Table 8: Feedback]

SR.NO	FIELD NAME	DATATYPE (SIZE)	CONSTRAINT	DESCRIPTION
1	Feedback_id	Int (10)	Primary Key	Store feedback Id
2	Feedback_name	Char (50)	Not null	Store user's name
3	Event_email	Varchar (100)	Not null	Store event's email
4	Feedback	Varchar (200)	-	Store feedback
5	Feedback_time	timestamp	Not null	Store event's time

## **CHAPTER-5**

### **TECHNICAL SPECIFICATION**

---

#### **5.1 Hardware Specification:**

- **5.1.1 Ram:** 4GB
- **5.1.2 Hard drive Storage Needed:** 200GB
- **5.1.3 Other Hardware Requirements:** None

#### **5.2 Platform:**

- **5.2.1 Supported Operating System:** Windows XP and above, LINUX and MacOS is compatible.
- **5.2.2 Programmer Server:** Xampp Apache Server 8.0.10 64-bit.

#### **5.3 Framework:**

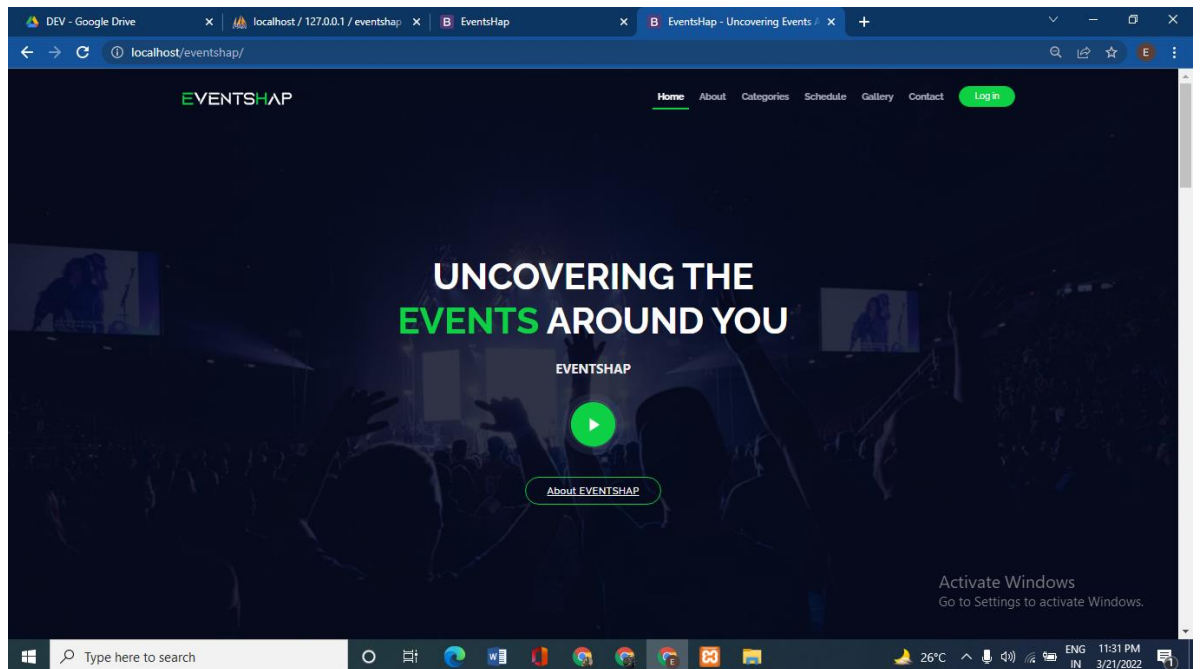
- **5.3.1 Mark-up Language:** HTML4 and HTML5.
- **5.3.2 Programming Language:** PHP 8.0.10v

#### **5.4 Technical Support:**

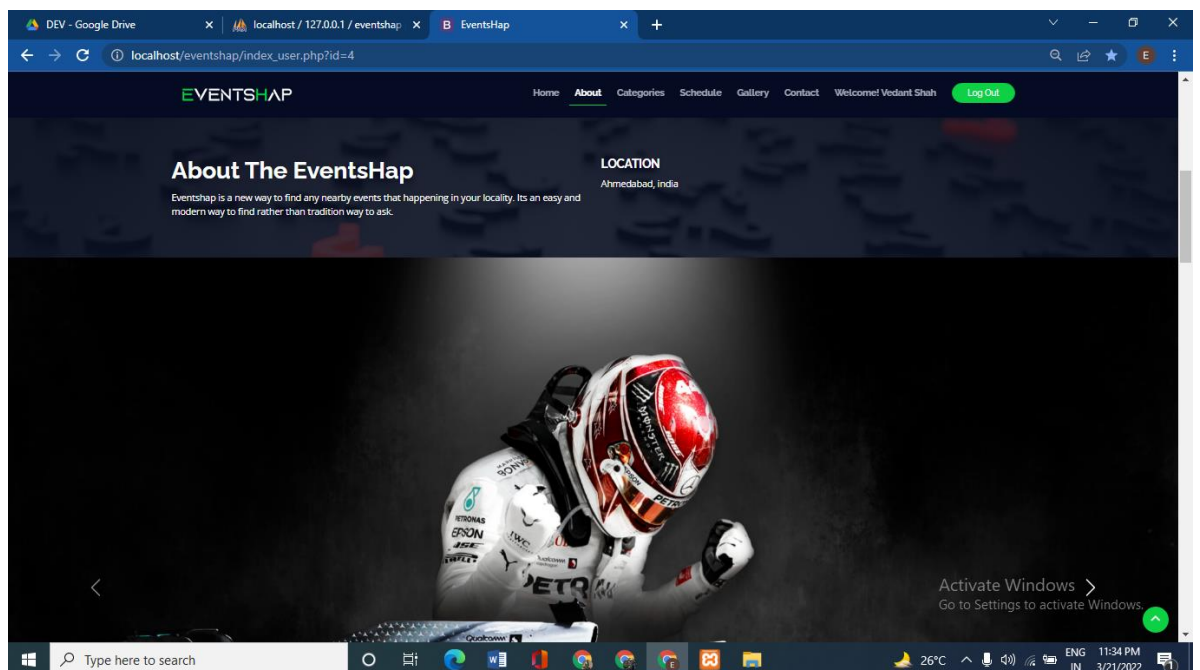
- **5.4.1 Front-End:** PHP 8.0.10v
- **5.4.2 Back-End:** MySQL 8.0.26
- **IDE Tools:** Sublime Text3 and Microsoft visual studio code.
- **UML Tools:** Microsoft Office Visio 2019.
- **SRS Tools:** Microsoft Word 2019

## 5.5 DESIGN LAYOUT

### User Side:

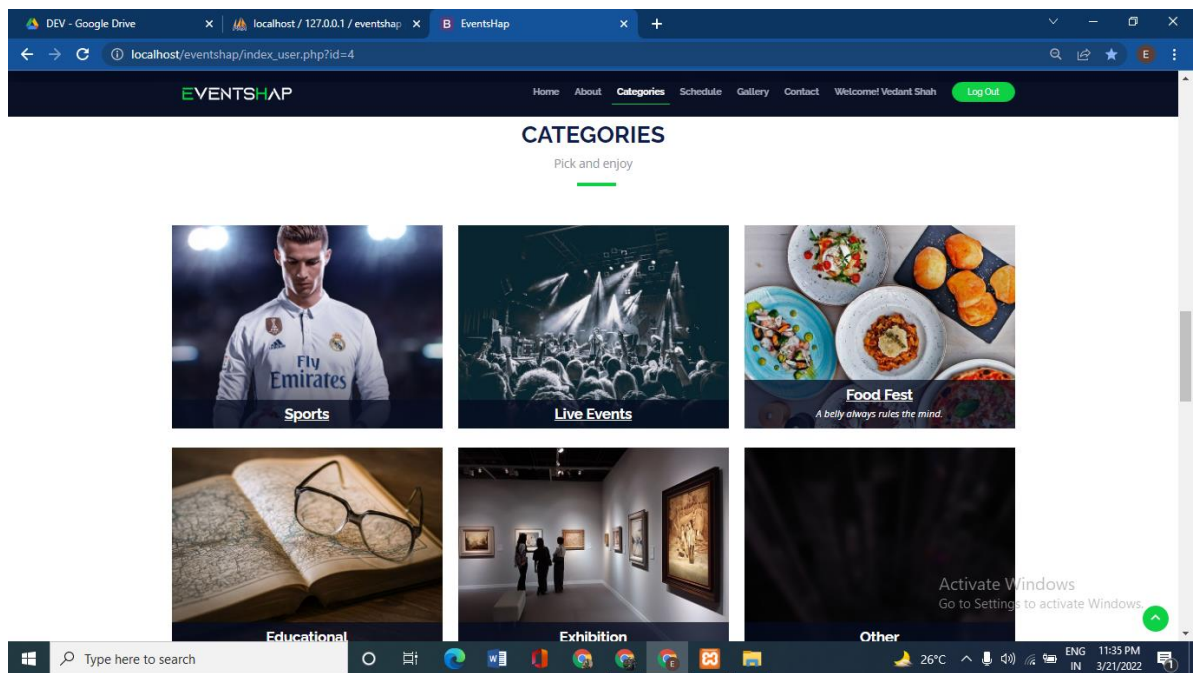


[Figure 8: Home Page]

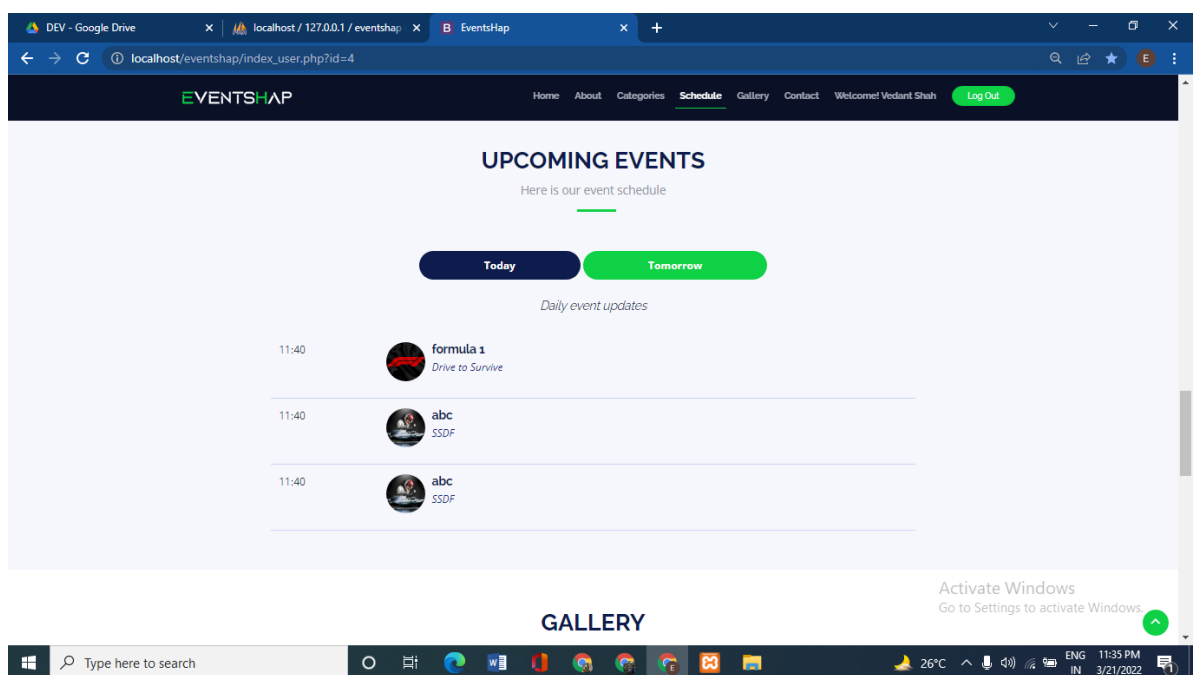


[Figure 9: Home Page]

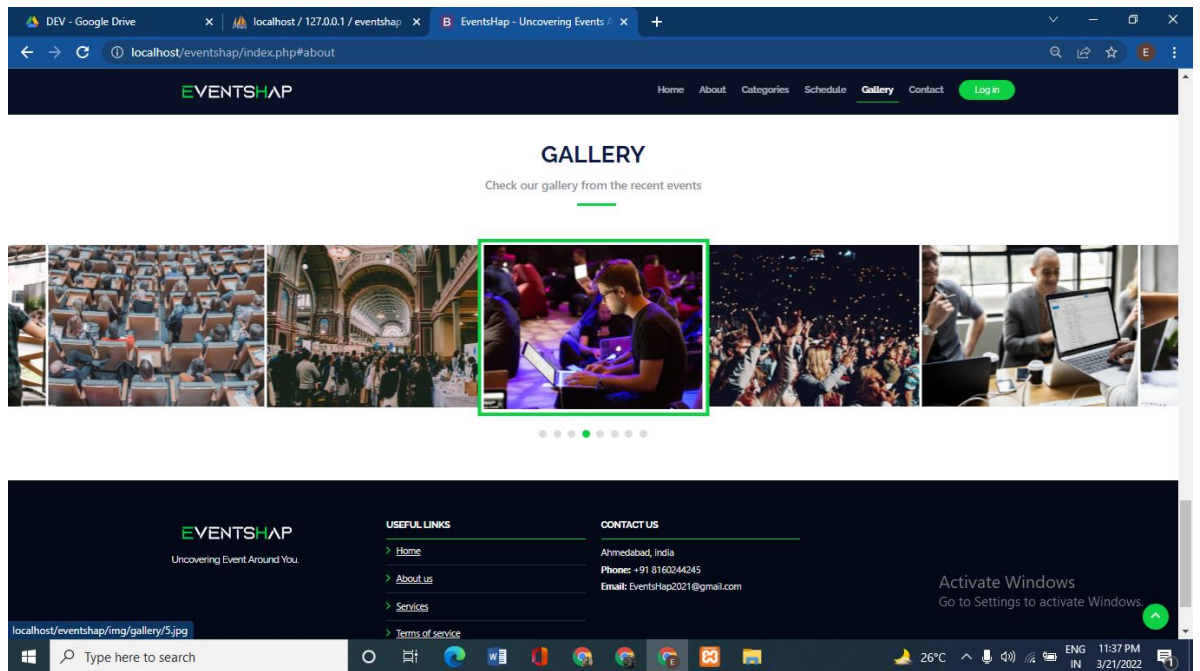




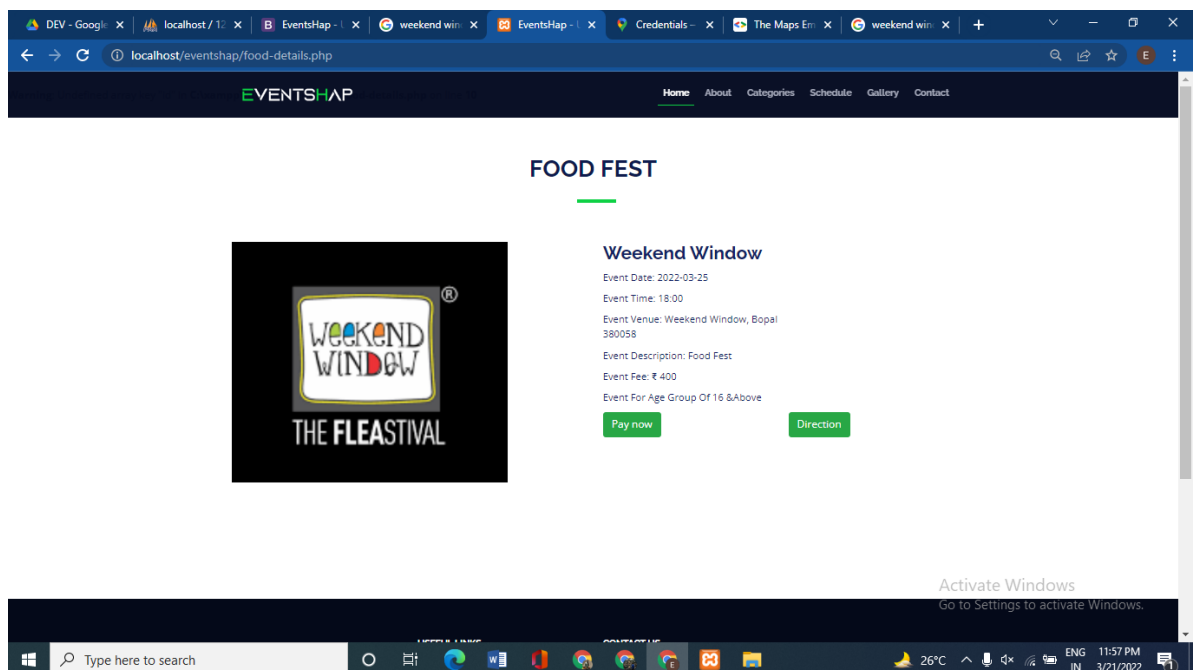
[Figure 10: Home Page]



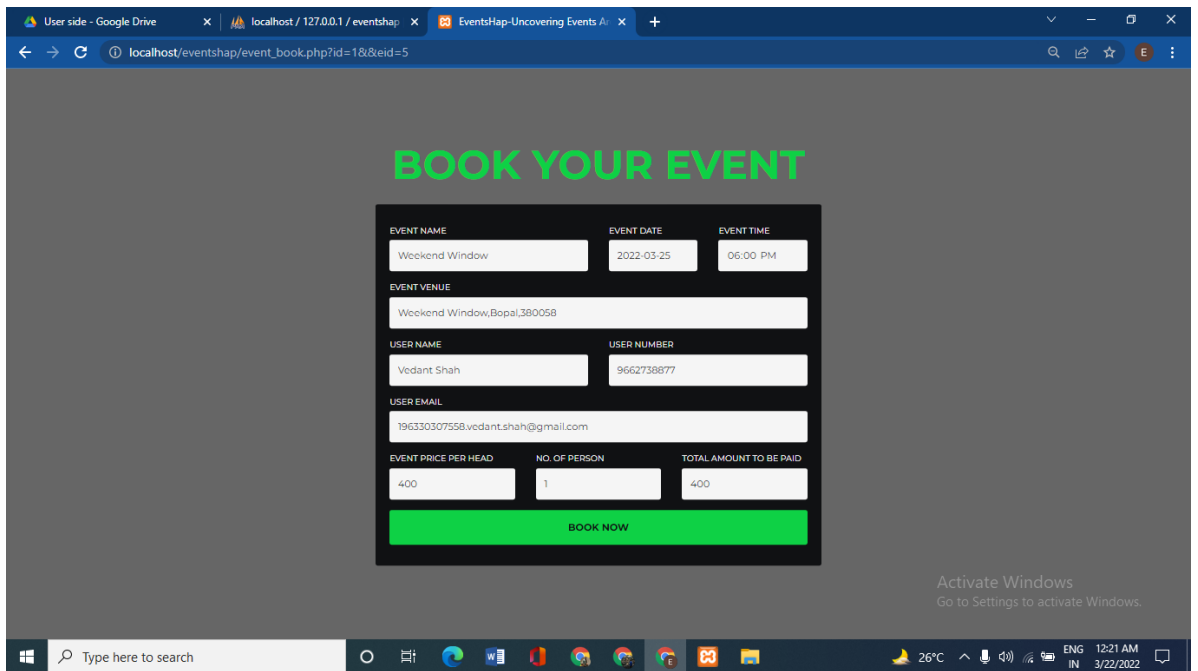
[Figure 11: Home Page]



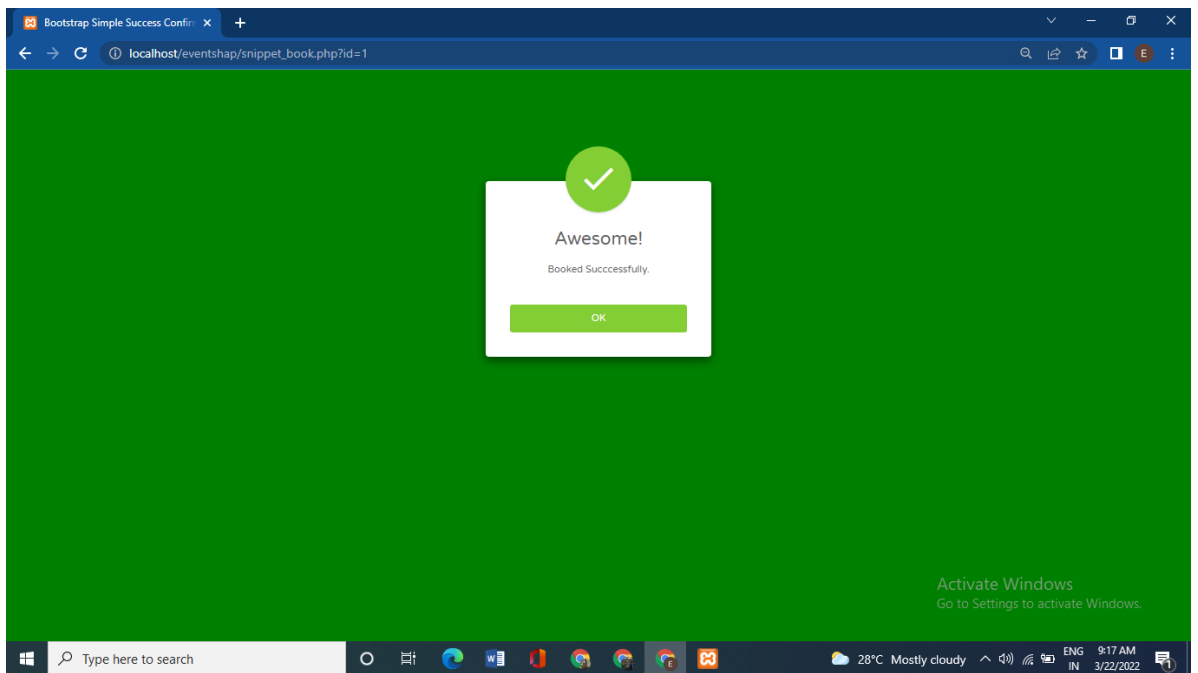
[Figure 12: Home Page]



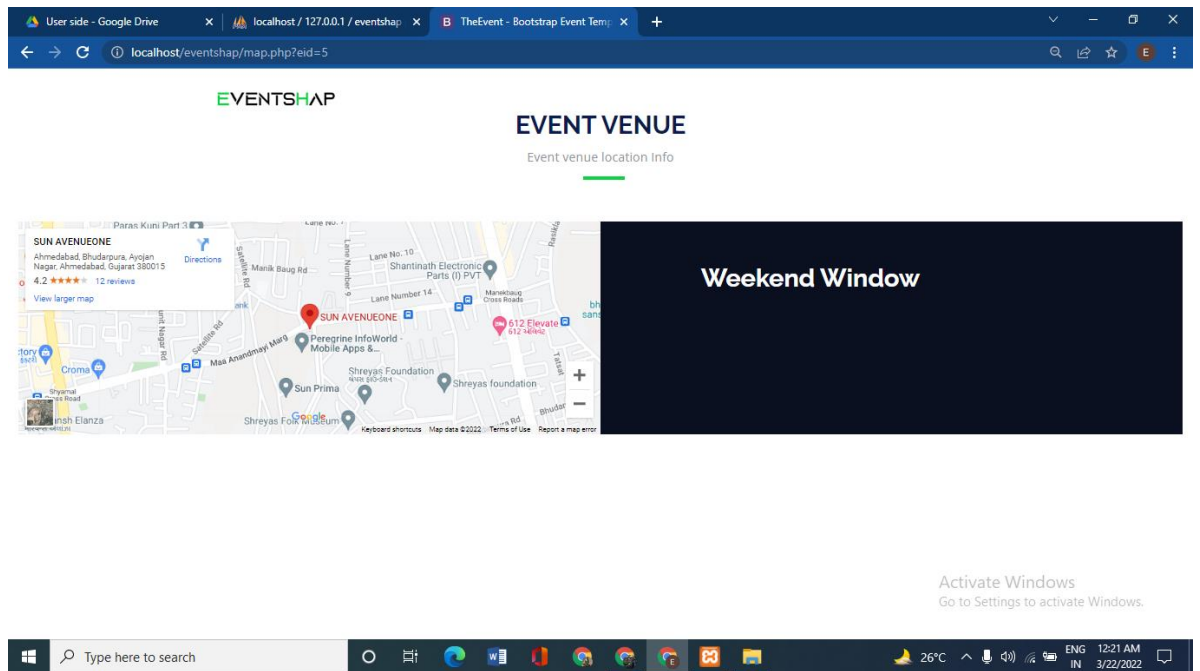
[Figure 13: Event Details Page]



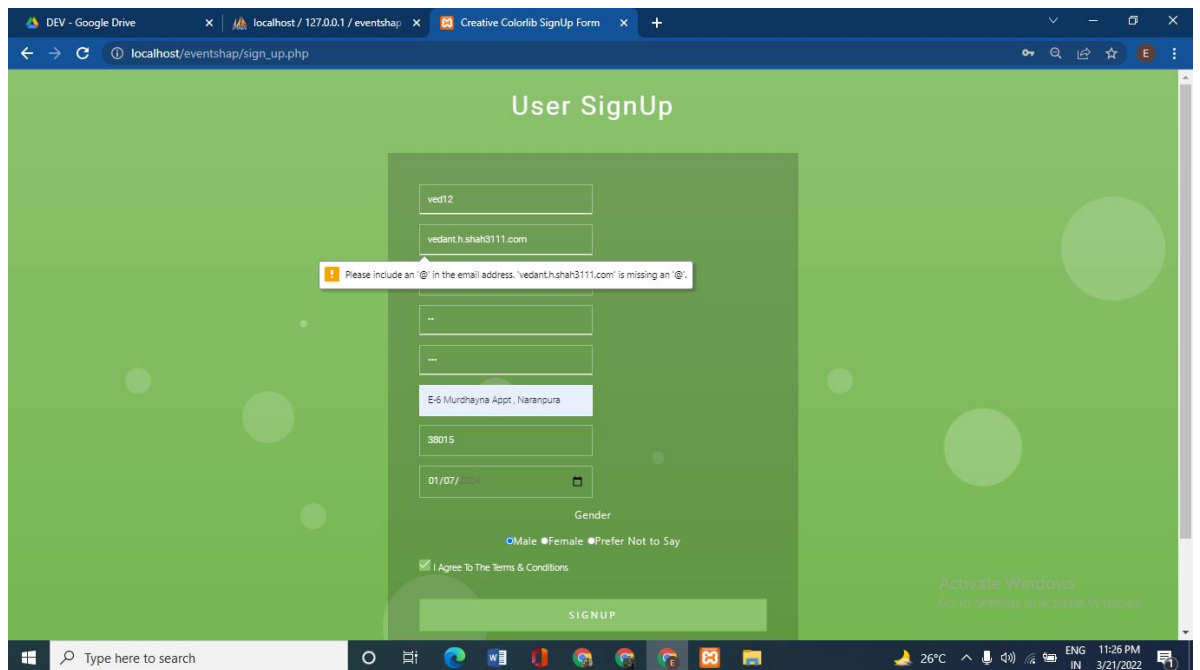
[Figure 14: Booking Page]



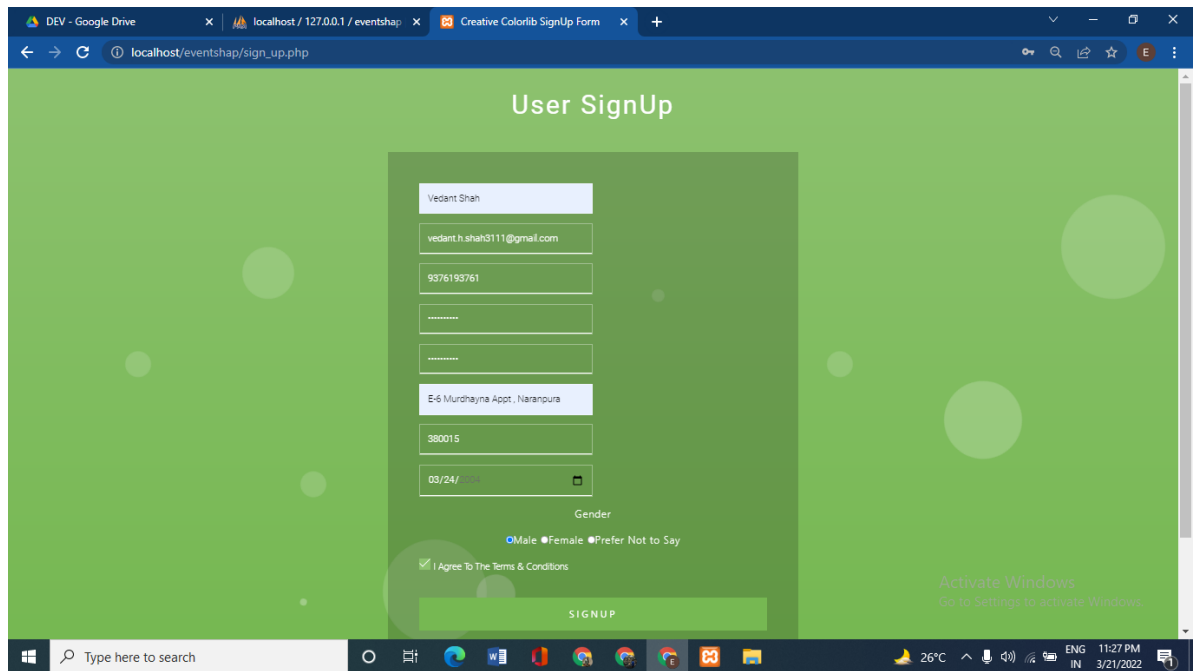
[Figure 15: Booking Successful Page]



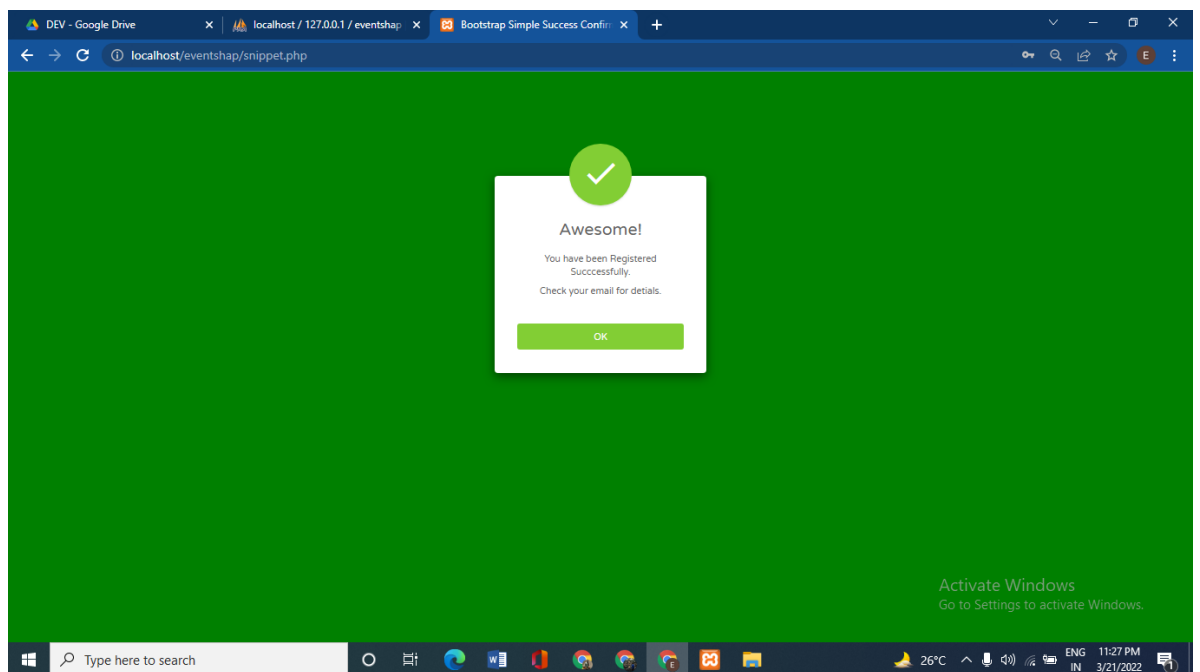
[Figure 16: Direction Page]



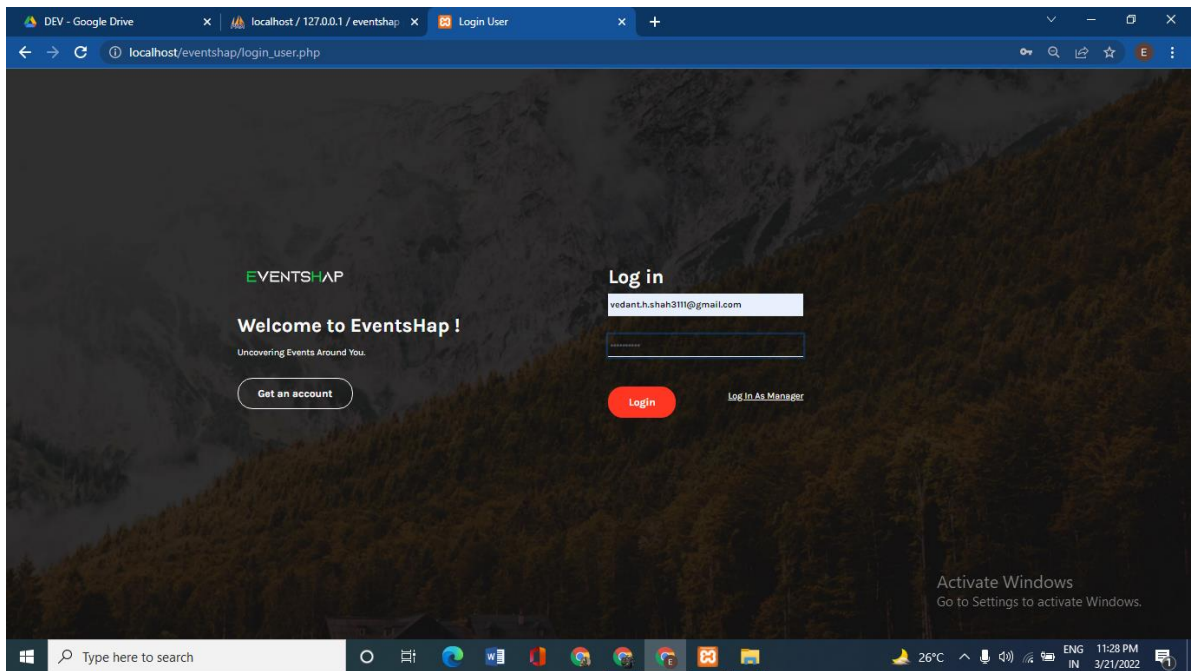
[Figure 17: User Registration Page Validation]



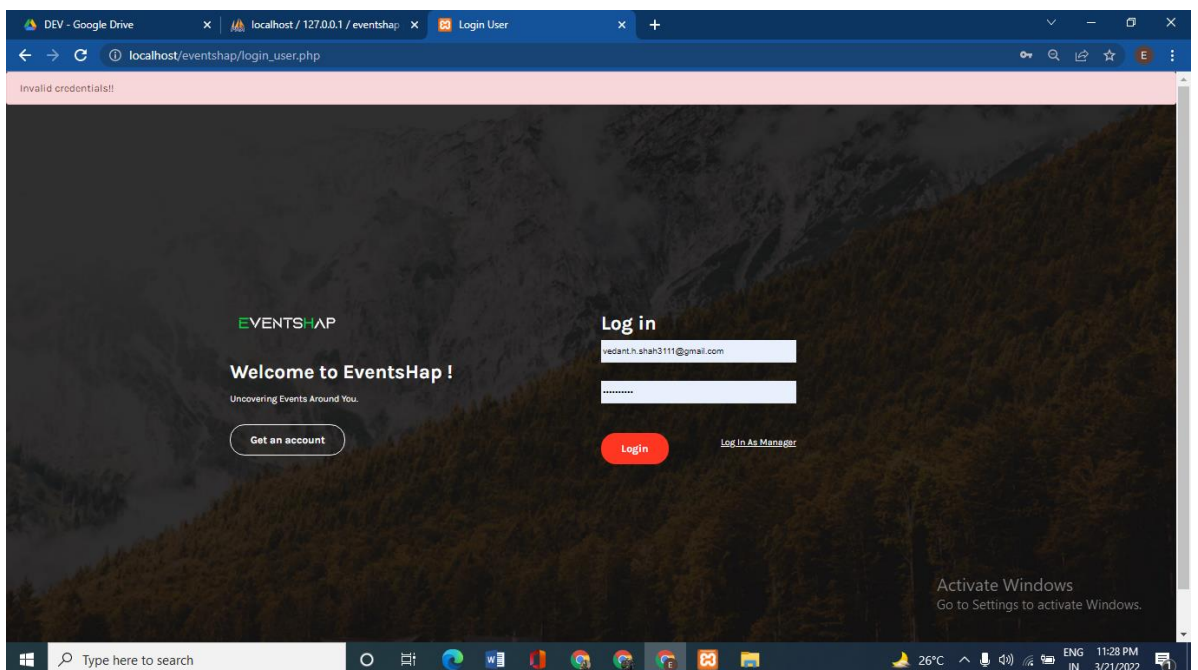
[Figure 18: User Registration Page]



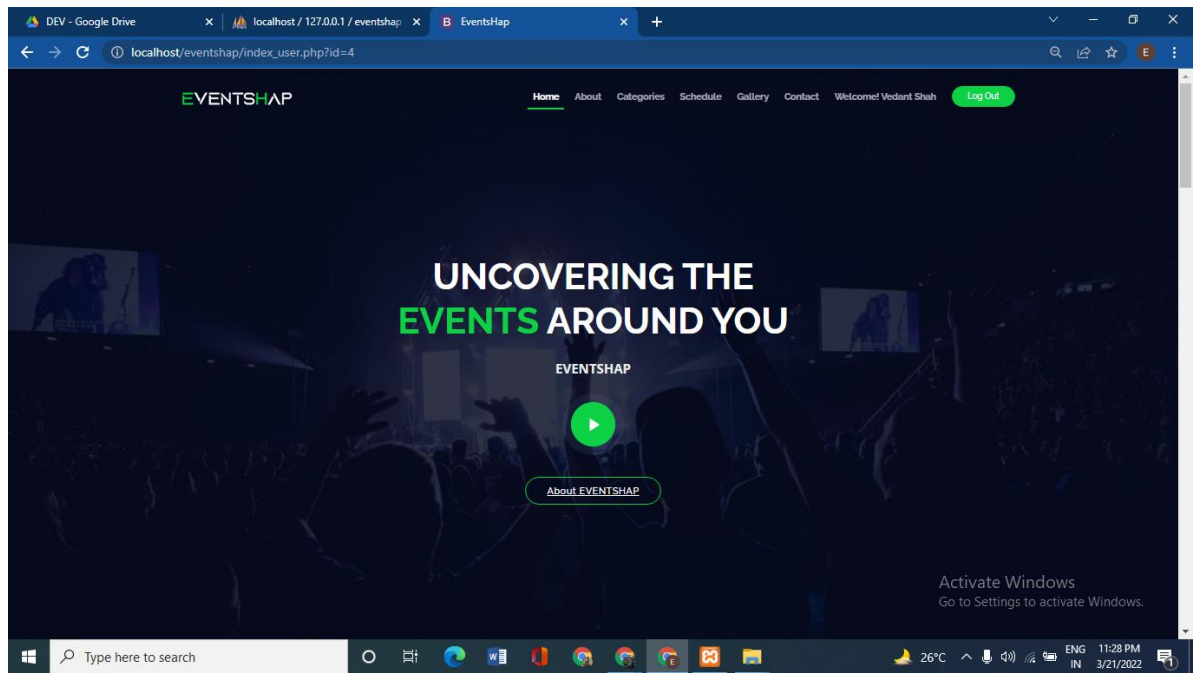
[Figure 19: Registration Confirm Page]



[Figure 20: Login Page]



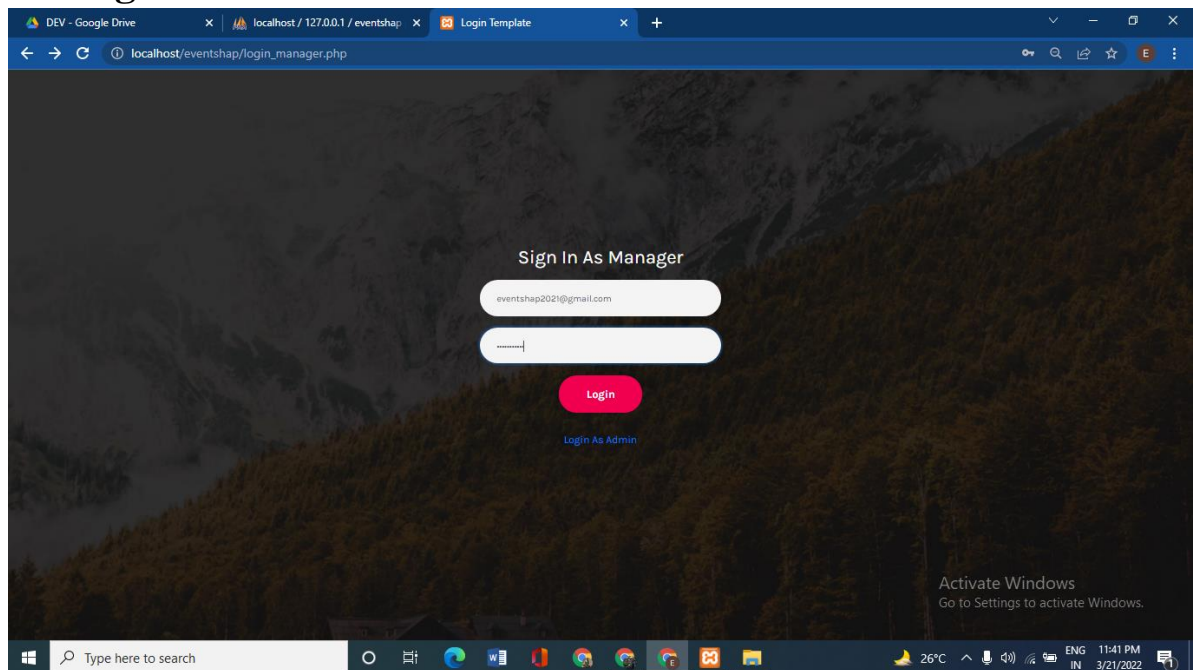
[Figure 21: Login Page]



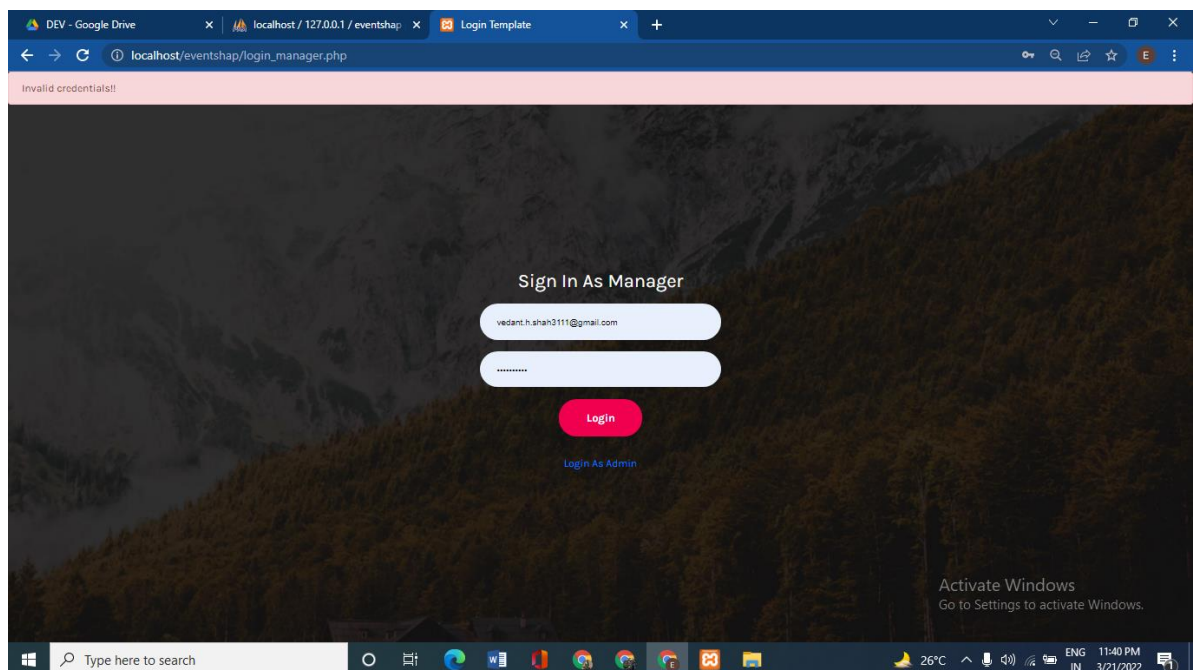
[Figure 22: User Home Page]



## Manager Side:

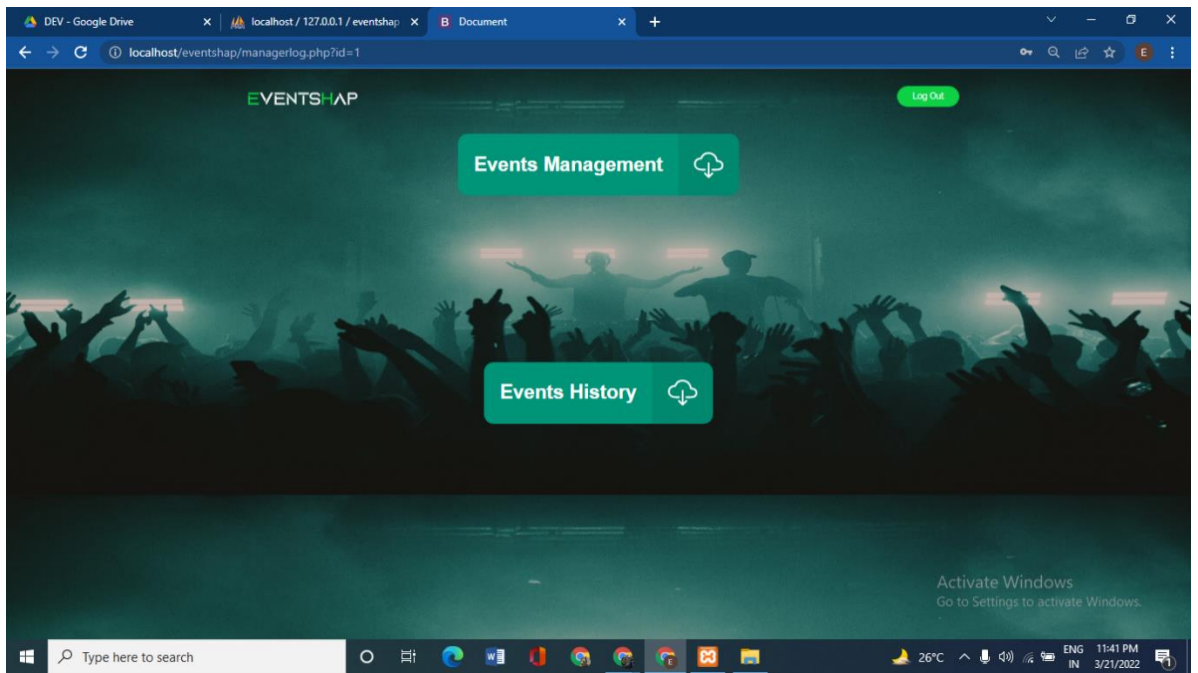


[Figure 23: Manager Login Page]

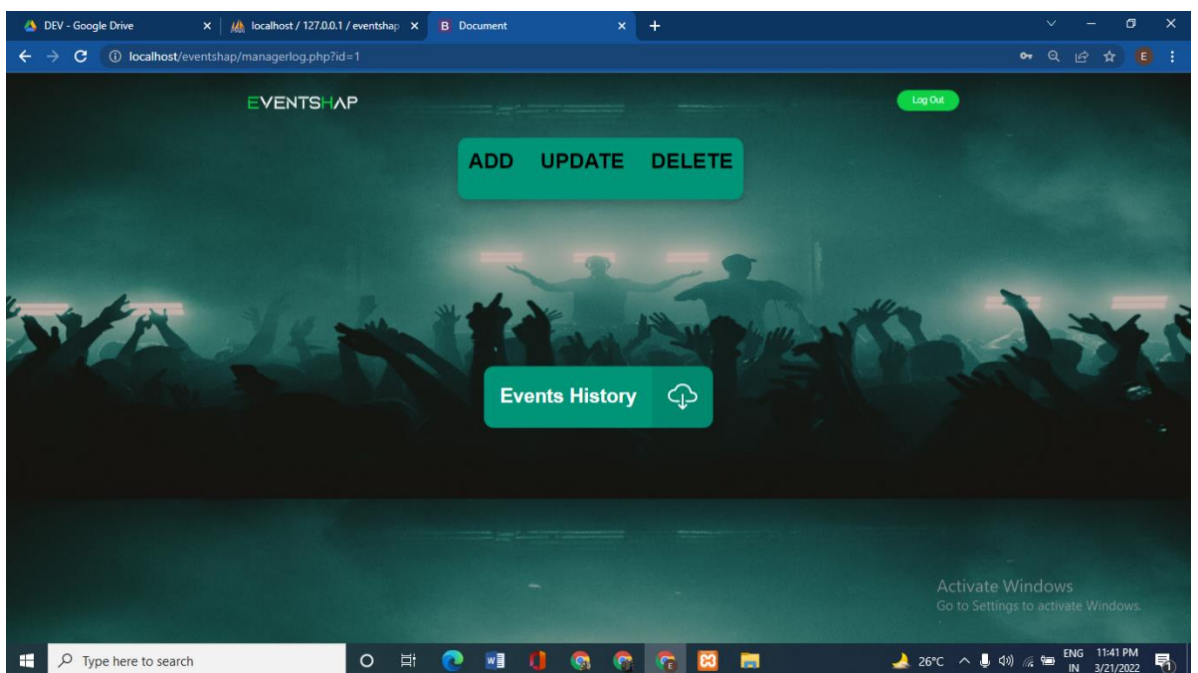


[Figure 24: Manager Login Page]

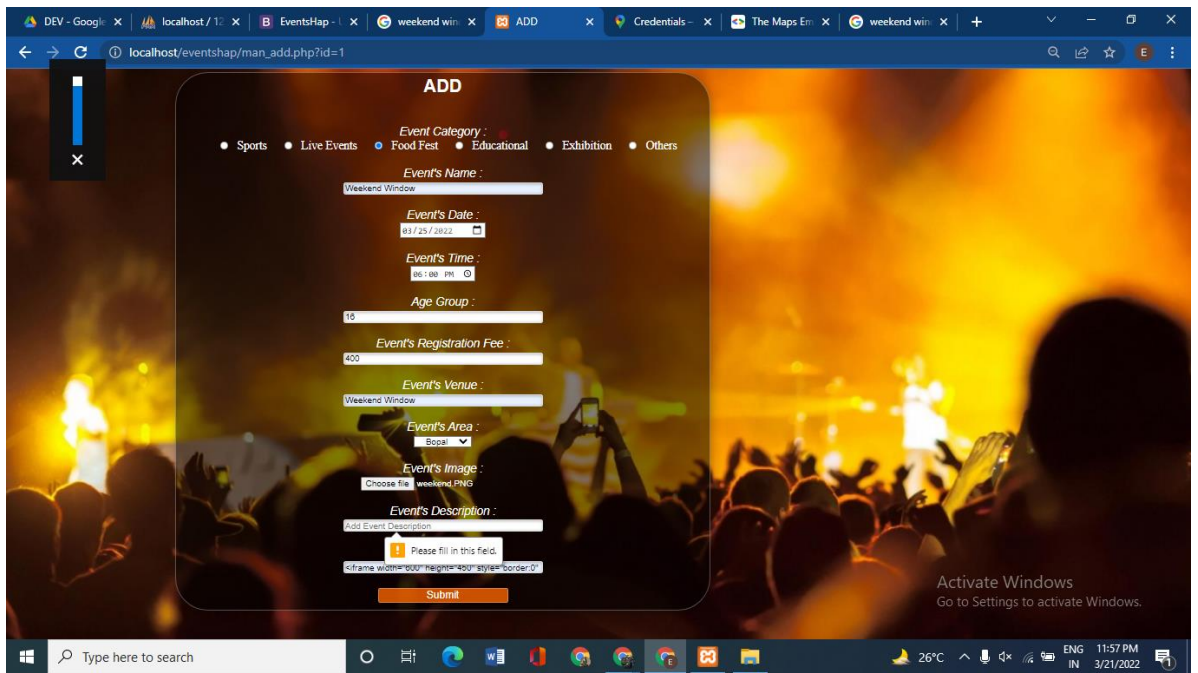




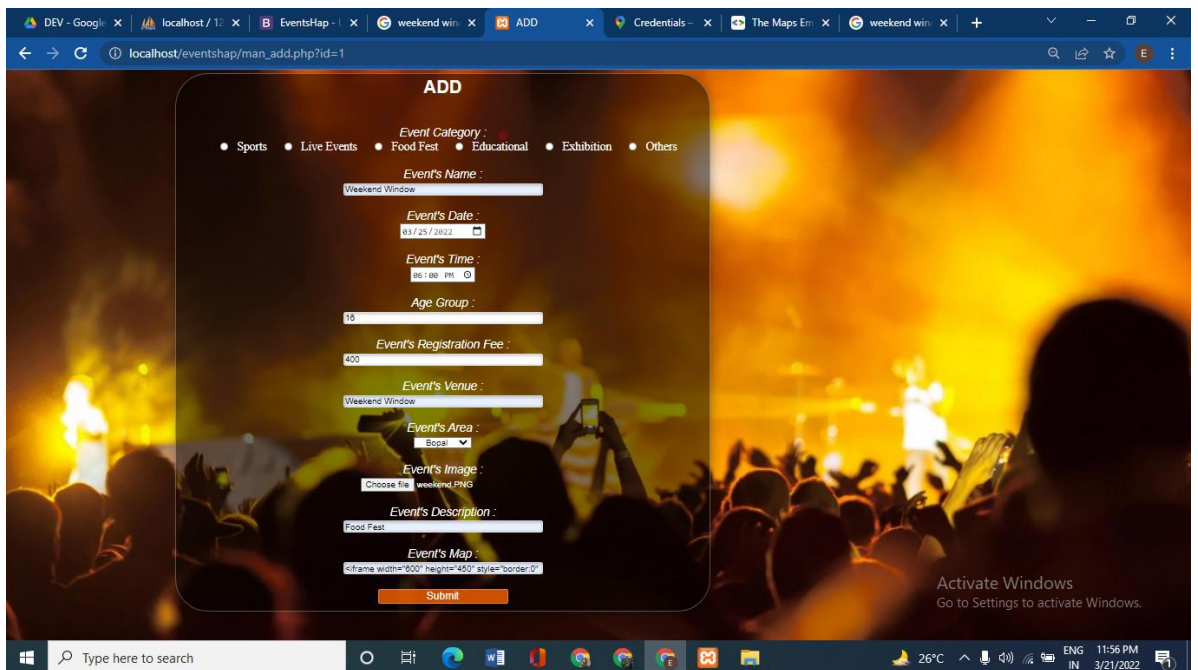
[Figure 25: Manager Home Page]



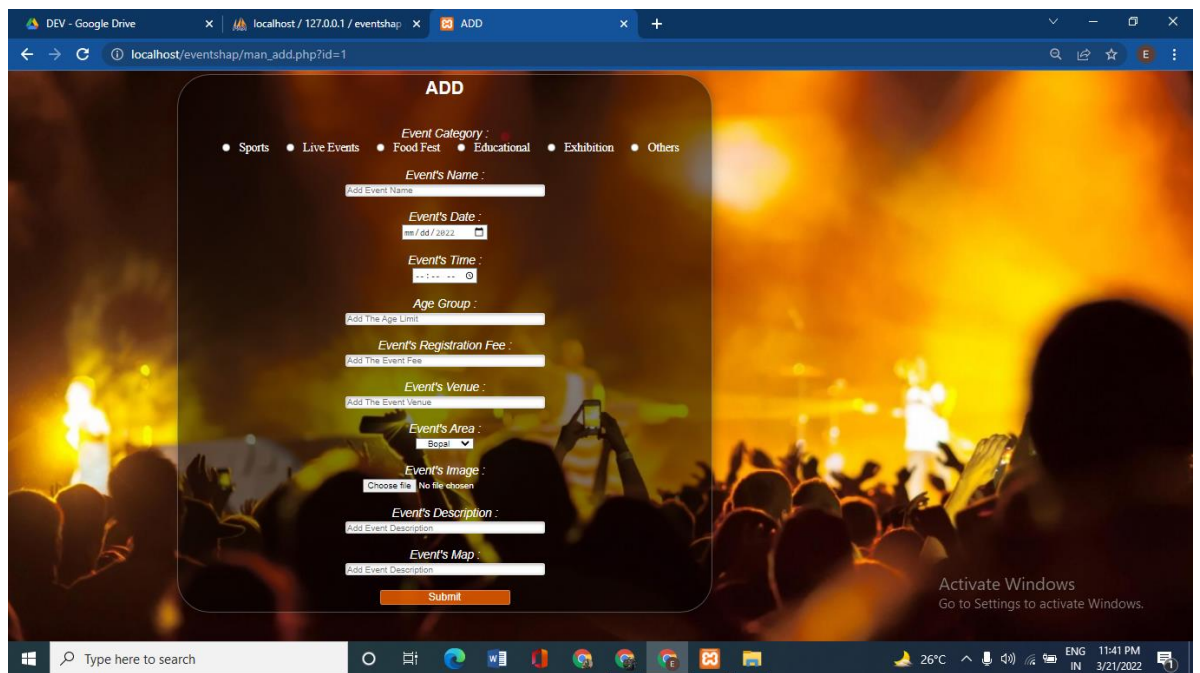
[Figure 26: Manager Option Page]



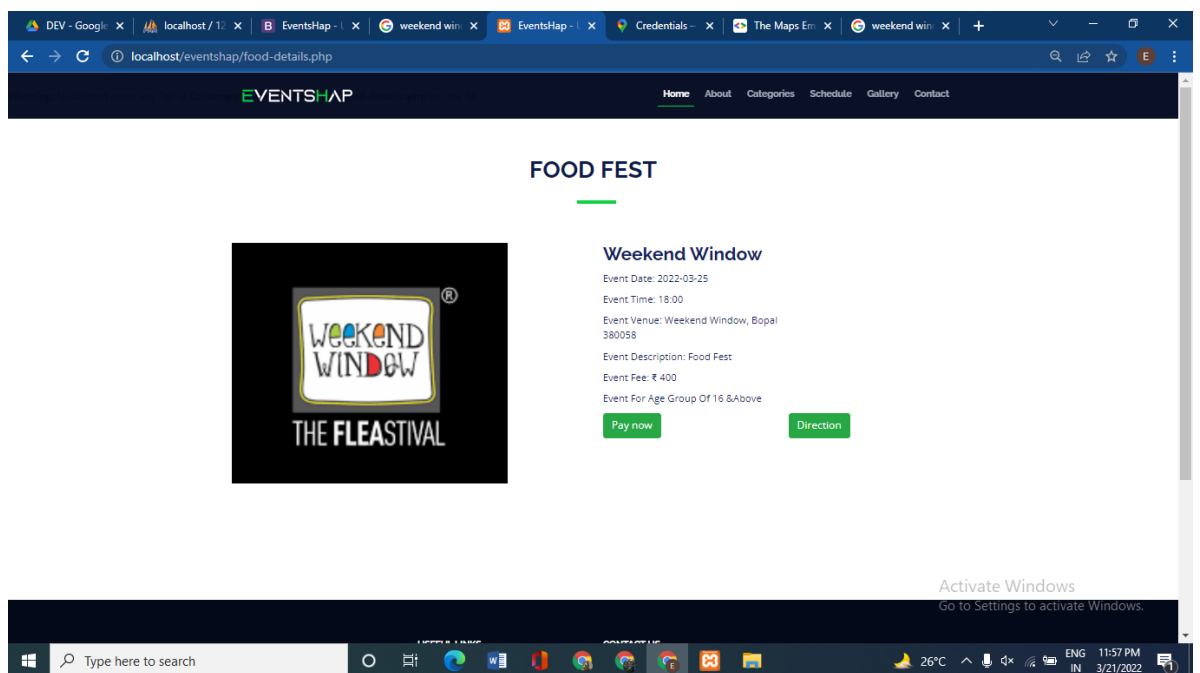
[Figure 27: Event Add Page Validation]



[Figure 28: Event Add Page Credential]



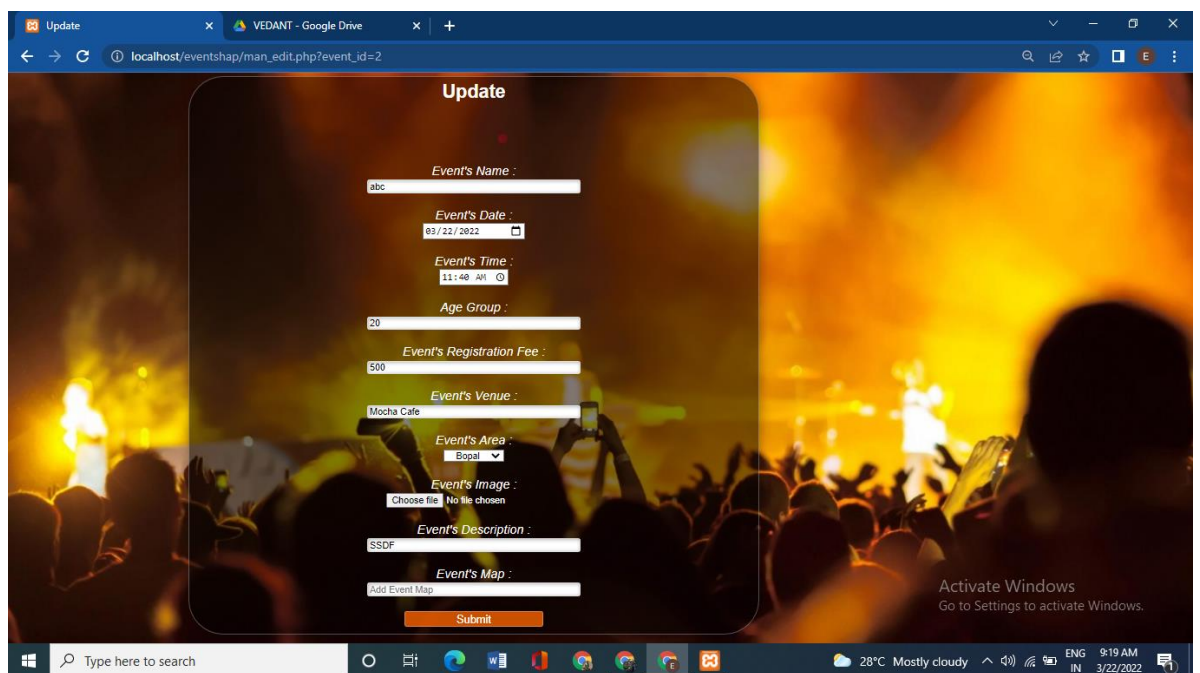
[Figure 29: Event Add Page]



[Figure 30: Event Details Page]

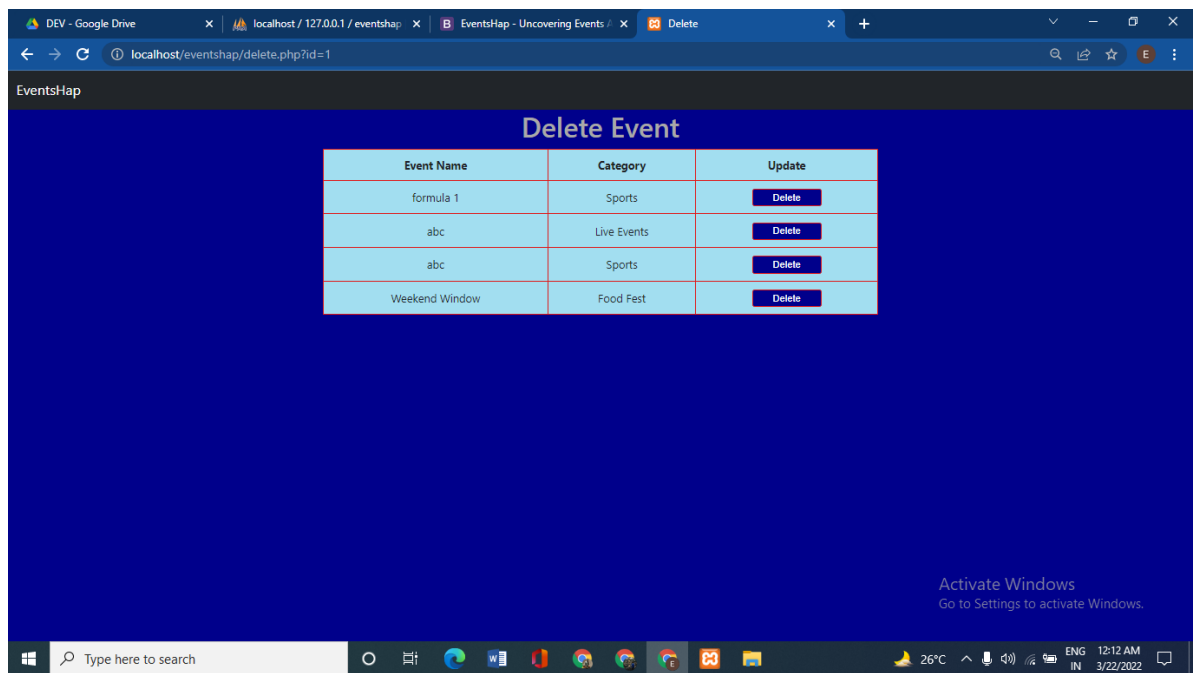


[Figure 31: Event Update Selection Page]

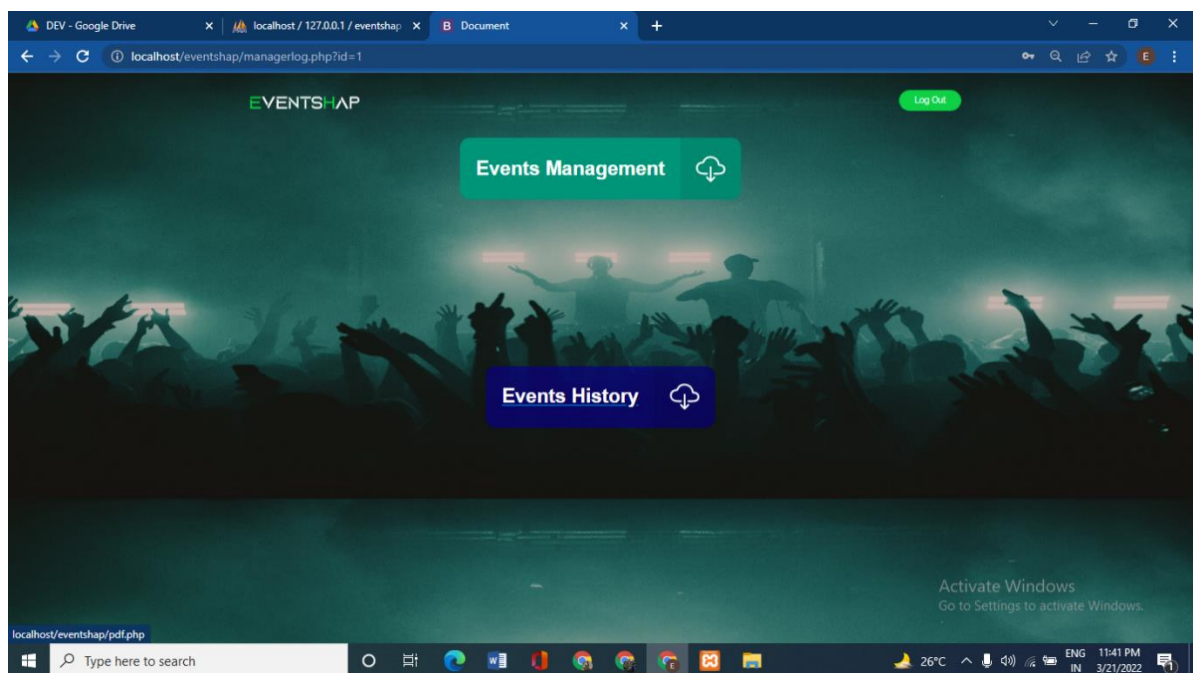


[Figure 32: Event Update Page]





[Figure 33: Event Delete Page]

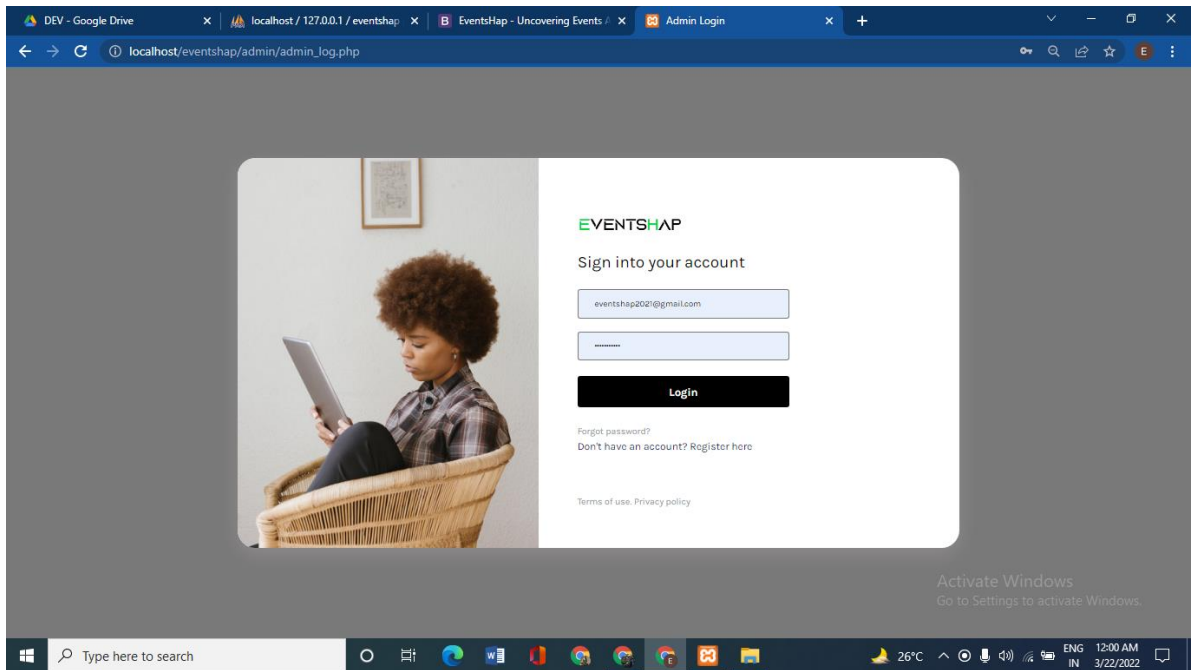


[Figure 34: Manager Home Event History Page]

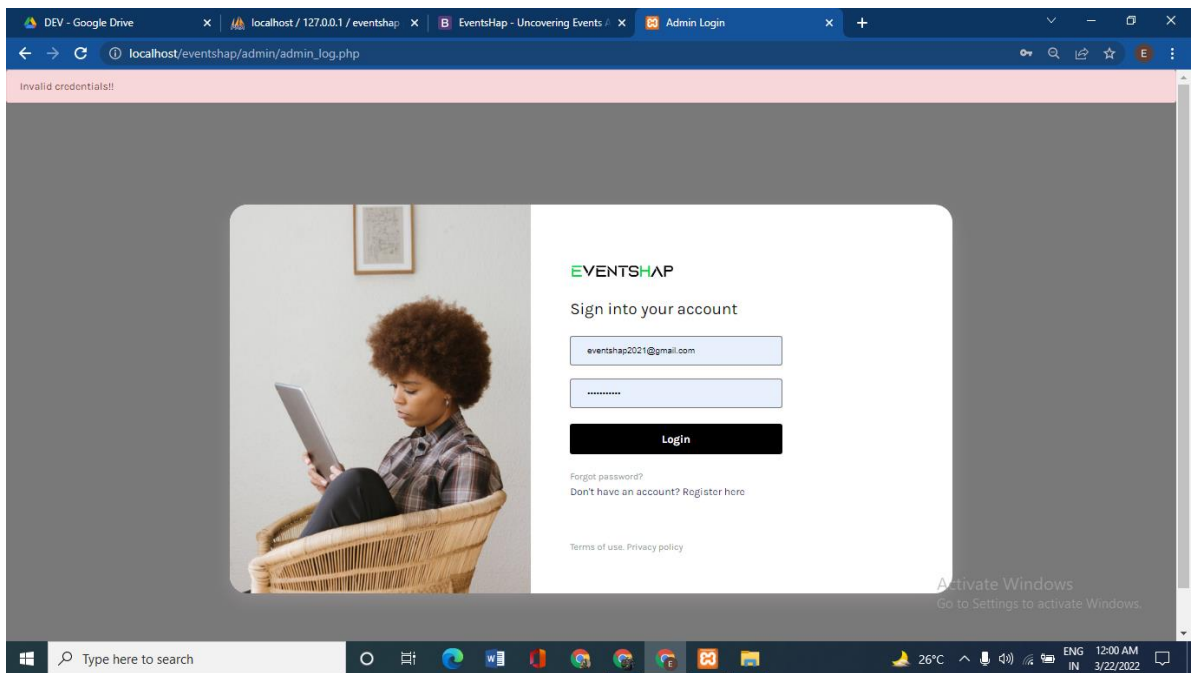


[Figure 35: Generated PDF]

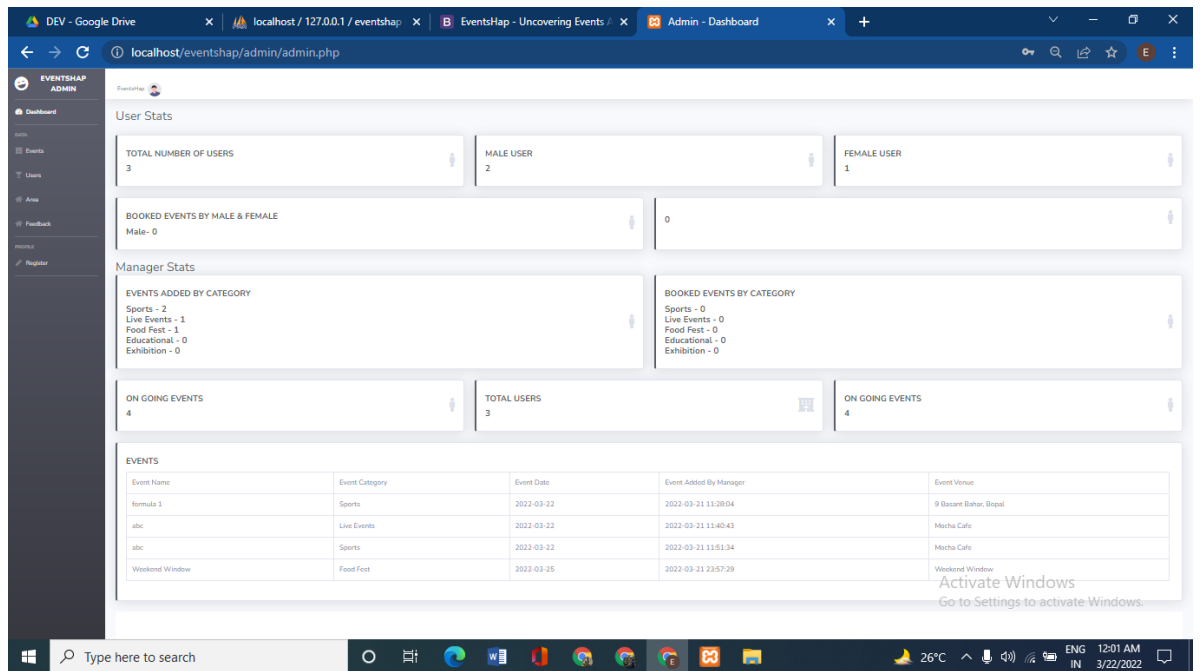
## Admin Side:



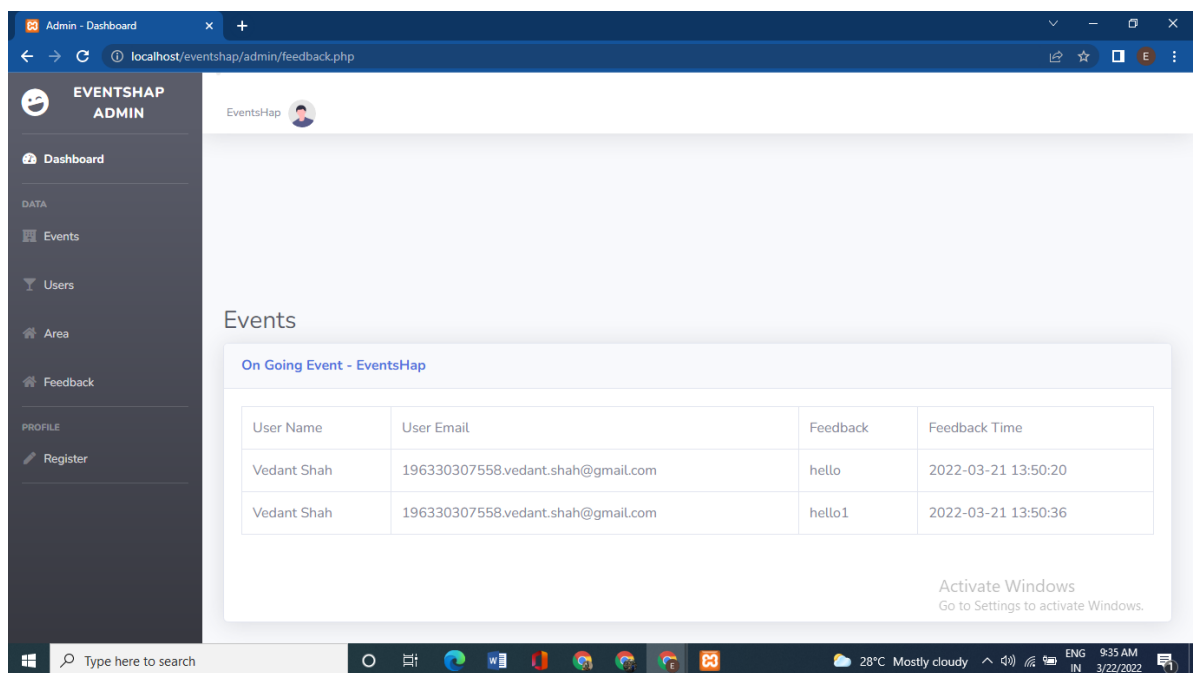
[Figure 36: Admin Login Page]



[Figure 37: Admin Login Page Validation]

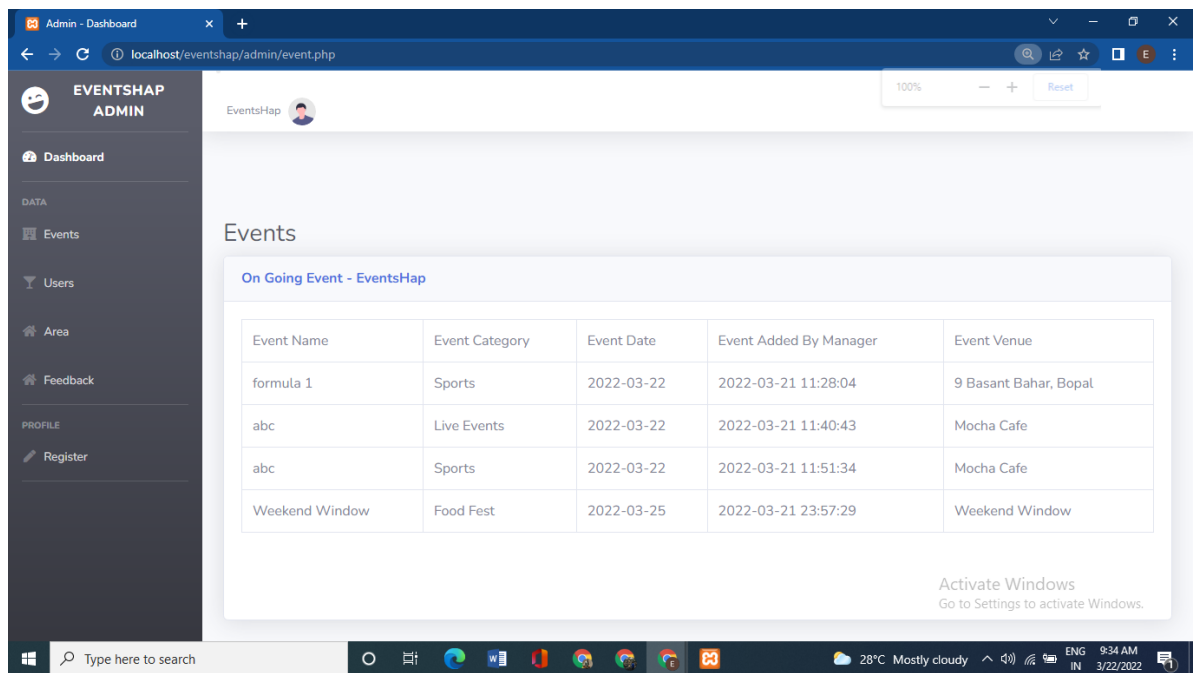


[Figure 38: Admin Dashboard]

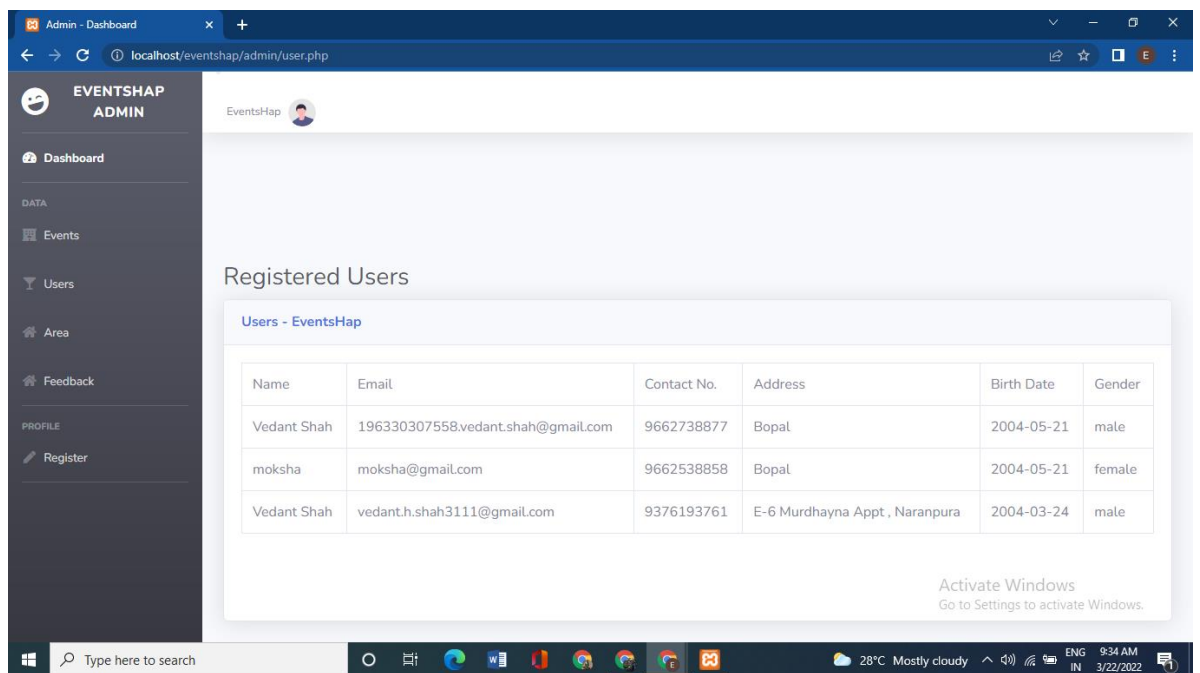


[Figure 39: Admin Event Page]

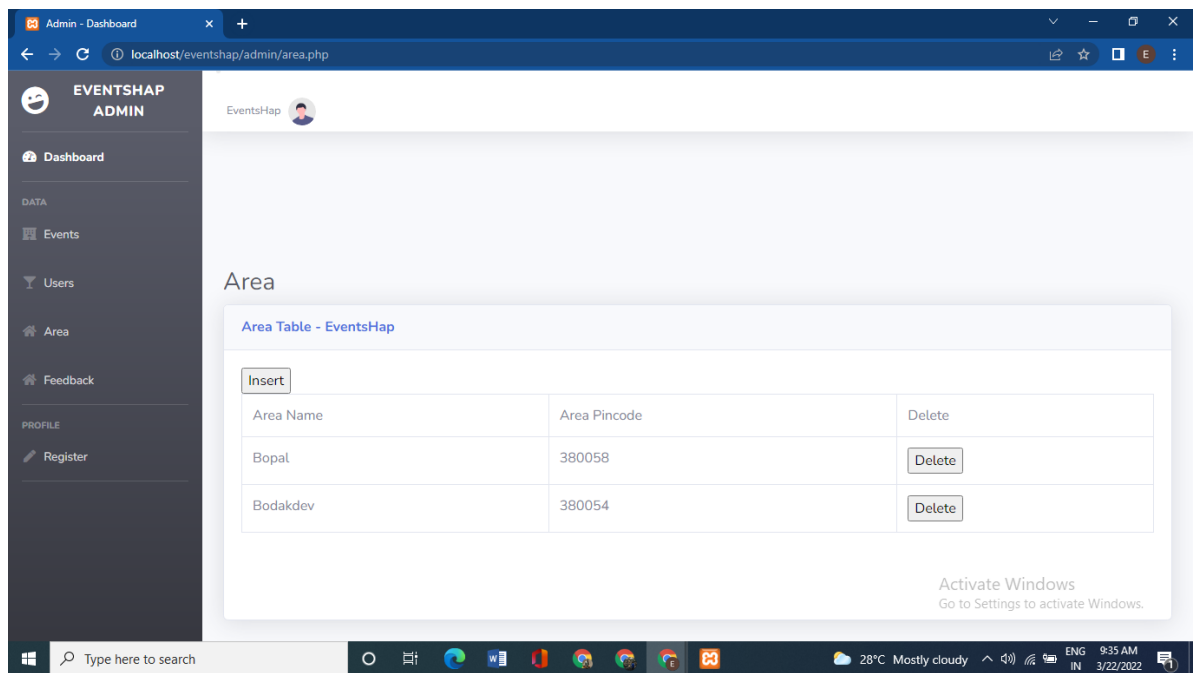




[Figure 40: Admin Event Page]



[Figure 41: Admin User Page]



[Figure 42: Admin Area Page]

#### 6.1 Testing Methods

##### 6.1.1 Black Box Testing

- Black Box Testing is also known as Behavioural Testing or Functional Testing. It is a technique of testing without having any knowledge of the internal working of the application.
- Black Box Testing treats the software as a “Black Box”- without any knowledge of internal working and it only examines the fundamental aspects of the system. This method of test can be applied to each and every level of software testing such as unit, integration, system and acceptance testing.



[Figure 8: Black box testing]

- This method attempts to find errors in the following cases:
  - Incorrect or missing functions
  - Interface Errors
  - Errors in structures or external database access
  - Behaviour or performance errors

- Initialization and termination errors

**Advantages:**

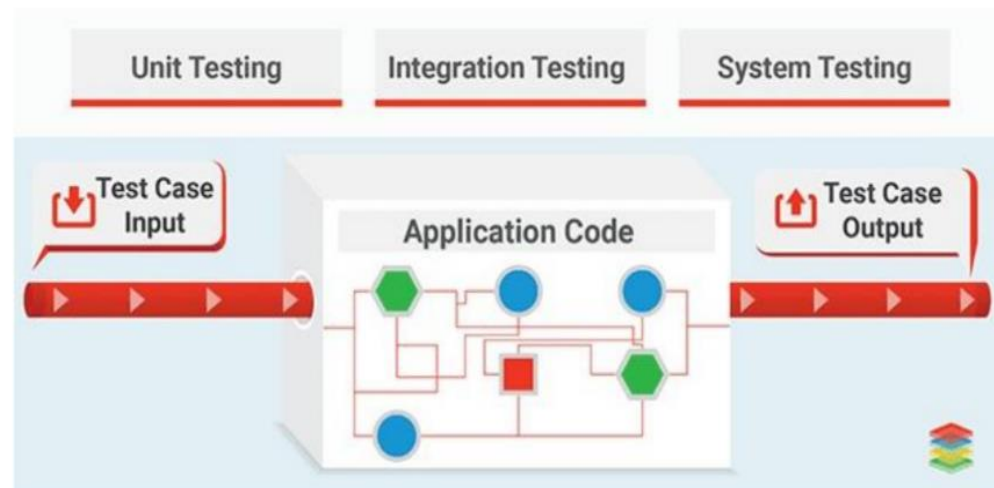
- Unbiased tests because the designer and tester work independently.
- Tester is free from any pressure of knowledge of specific programming languages to test the reliability and functionality of an application software.
- Test cases can be designed immediately after the completion of specifications.

**Disadvantages:**

- Testing every possible input stream is not possible because it is time-consuming and this would eventually leave many program paths untested.
- Test cases are extremely difficult to be designed without clear and concise specifications.
- Results might be overestimated at time.
- Cannot be used for testing complex segments of code.

### 6.1.2 White Box Testing

White Box Testing is a software testing method in which the internal structure/design/implementation of the item being tested is known to the tester. The tester chooses inputs to exercise paths through the code and determines the appropriate outputs. This method is named so because the software program, in the eyes of the tester, is like a white/transparent box; inside which one clearly sees.



[Figure 9: White box testing]

- The aim of this testing is to investigate the internal logic and structure of the code. That is why white box testing is also known as Structural Testing.
- Test Cases generated using White Box Testing can:
  - Guarantee that all independent paths within a module have been exercised at least once.
  - Exercise all decisions whether they are true or false.
  - Exercise external data structure of the program.

#### **Advantages:**

- Code optimization by revealing hidden errors.
- Transparency of the internal coding structure which is helpful in deriving the type of input data needed to test an application effectively.

- Covers all possible paths of a code thereby, empowering a software engineering team

to conduct thorough application testing.

- Enables programmer to introspect because developers can carefully describe any new

implementation.

- Gives engineering-based rules to stop testing an application.

### **Disadvantages:**

- Since tests can be very complex, highly skilled resources are required with a thorough

knowledge of programming and implementation.

- Test script maintenance can be a burden if the implementation changes too frequently.

- Necessity to create full range of inputs to test each path and condition make the white

box testing method time-consuming.

## **CONCLUSION**

It was a wonderful experience to work on this project. By doing so we have learnt a lot. This project has taken us into many phases of project development and gave us a real insight of how a website of particular event booking is managed online. This project will result in people being able to book the event of their choice and have a great experience.

# BIBLIOGRAPHY

## ▪ BOOK REFERENCES

[1] Jon duckett, “HTML and CSS: Design and build websites”,  
Wiley. [Edition 2].

[2] PHP and MySQL Web Development, Fifth Edition  
Authors: Luke Welling and Laura Thomsan  
Published by : Addison-Wesley Professional in 2016

## ▪ WEB REFERENCE

[1] W3school

Link: <https://www.w3schools.com/tags/default.asp>

[2] Java T point

Link: <https://www.javatpoint.com/html-tutorial>

[3] Tutorials Point

Link: <https://www.tutorialspoint.com/html/index.htm>

[4] W3resource

Link: <https://w3resource.com/mysql-exercises/>

[5] JQuery CDN

Link: <https://code.jquery.com/>