INSTITUTE FOR ADVANCED COMPUTING AND SOFTWARE DEVELOPMENT AKURDI, PUNE

Documentation On

Event Booking System

PG-DAC SEPT 2022

Submitted By:

Group No: 58

Roll No.	**Name:**
229131	**Anshul Gupta**
229234	**Vinit Shrivastava**

Mr. Rohit Puranik
Centre Coordinator

Mrs. Manjary Deshpande
Project Guide

ABSTRACT

The event booking system documentation outlines the functionality, usage, and technical specifications of a software system designed to streamline the process of event management. This system allows users to create, manage, and book events, as well as handle all associated tasks such as scheduling, payment processing, and attendee management. The documentation includes a detailed description of the system architecture, user interfaces, and database schema, as well as instructions for installation, configuration, and customization.

The documentation also provides a comprehensive guide to the system's APIs, which enable integration with external systems such as payment gateways and email marketing services.

ACKNOWLEDGEMENT

I take this occasion to thank God, almighty for blessing us with his grace and taking our endeavor to a successful culmination. I extend my sincere and heartfelt thanks to our esteemed guide, **Mrs. Manjary Deshpande** for providing me with the right guidance and advice at the crucial juncture sand for showing me the right way. I extend my sincere thanks to our respected **Centre Co-Ordinator Mr. Rohit Puranik**, for allowing us to use the facilities available. I would like to thank the other faculty members also, at this occasion. Last but not the least, I would like to thank my friends and family for the support and encouragement they have given me during the course of our work.

Anshul Gupta (229131)

Vinit Shrivastava (229234)

Table of Contents

LIST OF FIGURES

INTRODUCTION

Welcome to the documentation for the event booking system, a powerful software solution designed to simplify event management. In today's fast-paced world, event organizers are constantly searching for ways to improve efficiency and reduce the workload associated with organizing events. The event booking system offers a comprehensive set of tools that enables organizers to manage all aspects of event planning and booking from a single, centralized platform.

This documentation is designed to provide you with a thorough understanding of the event booking system and its capabilities. We will walk you through the features and functionality of the system, including creating and managing events, handling attendee registration and payment processing, and generating reports. Additionally, we will provide technical details regarding the system architecture, APIs, and database schema, as well as instructions for installation, configuration, and customization.

Features: -

1. Events Available- Birthday, Baby Shower, Anniversary
2. Search for vendors and events easily ,
3. Cart feature
4. Date and time of event will be notified by the system
5. The admin can add/delete Suppliers and delivery boys.
6. Allows the customers to maintain cart.

1.1 PROJECT OBJECTIVE

The objective of the project is to make an online platform to book events by a customer listed by a vendor. In order to build such an application complete web support, need to be provided. A complete and efficient web application which can provide the online shoppingexperience is the basic objective of the project. The web application can be implemented inthe form of an android application with web view.

1.2 PROJECT OVERVIEW

The event booking system is a software solution designed to simplify and automate event management processes. It provides a comprehensive set of tools for event organizers to create and manage events, handle attendee registration and payment processing, and generate reports.
The system is built with a user-friendly interface and can be customized to meet the specific needs of different types of events, from small local gatherings to large-scale conferences.

The primary objective of the event booking system is to provide event organizers with a user-friendly and efficient platform for managing all aspects of event planning and booking. The system is designed to automate and streamline the event management process, from creating and promoting events to handling attendee registration and payment processing.

The event booking system is suitable for various types of events, including conferences, seminars, workshops, festivals, and more. The documentation provides a detailed guide on the system's functionality, technical specifications, installation, configuration, and customization, helping users to utilize the system effectively and achieve their event management objectives.

1.3 PROJECT SCOPE

The event booking system is not intended to handle the logistics of the event itself, such as venue management, catering, or transportation. It is solely designed to provide event organizers with a comprehensive set of tools to manage the event booking process and enhance the overall attendee experience.

1.4 STUDY OF THE SYSTEM

1.4.1 MODULES:

The system after careful analysis has been identified to be presented with the following modules and roles.

The modules involved are:

- Administrator
- Vendor
- Customer

1.4.1.1 Administrator:

The administrator is the super user of this application. Only admin have access into this admin page. Admin may be the owner of the shop. The administrator has all the information about the users and about all products.

This module is divided into different sub modules.

1. Manage Vendor

2. Manage Customer
3. Manage Events
4. Approve Users

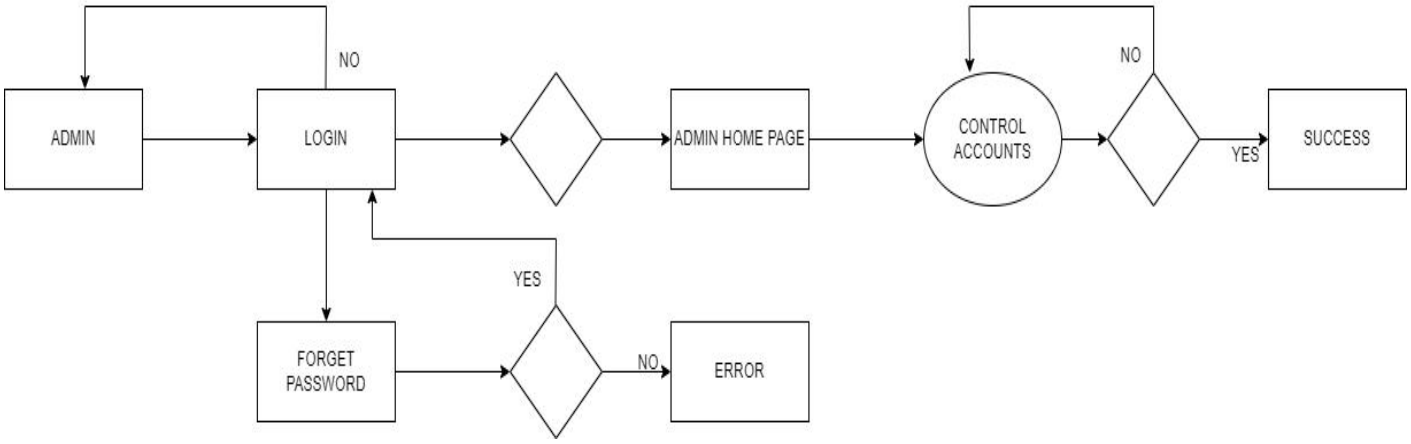


Figure 1 Admin Activity Diagram

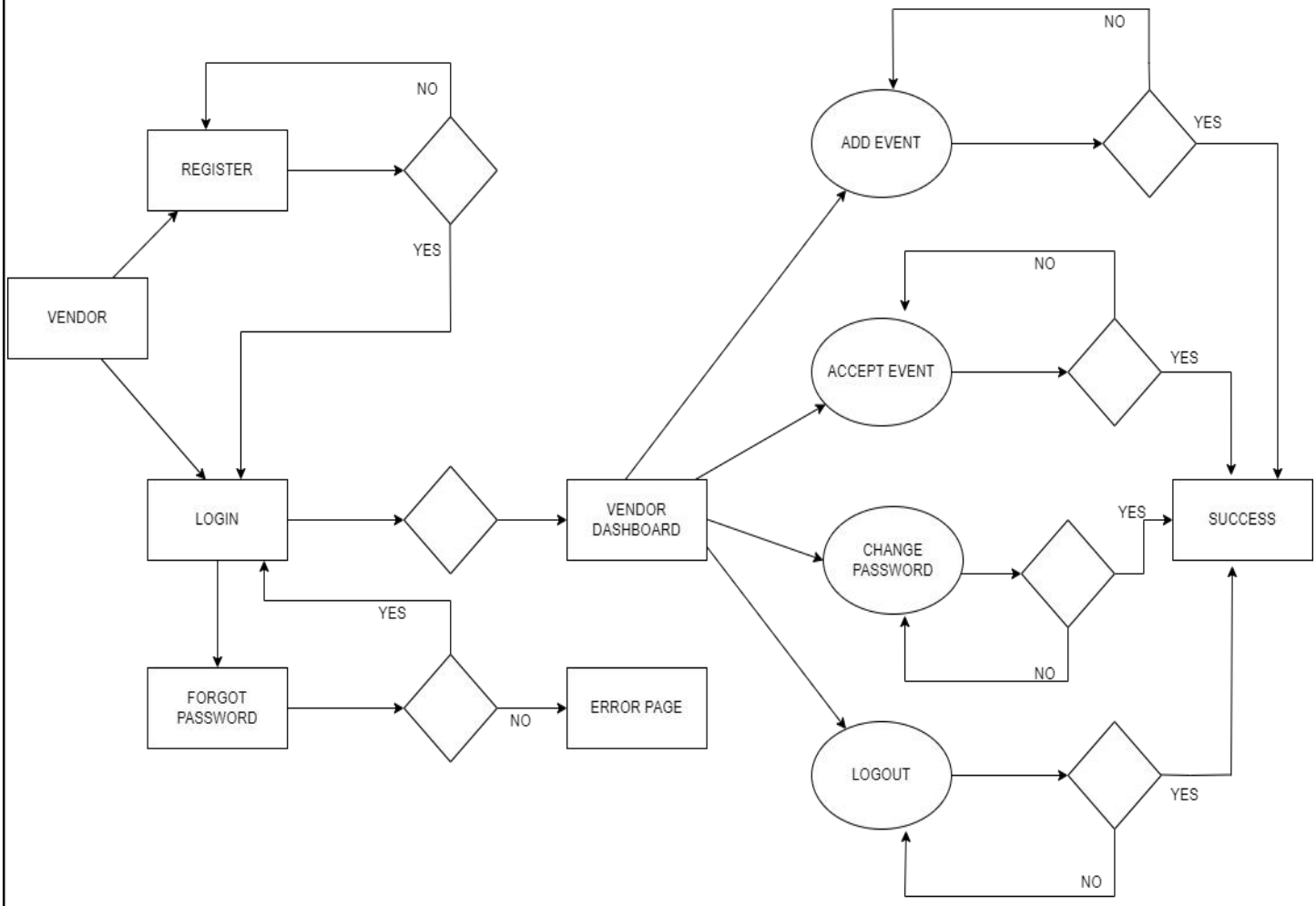


Figure 2 Vendor Activity Diagram

- **Edit Event**

  Vendor can edit his added event.

- **Set Discounts**

  Vendor can set discounts to the events.

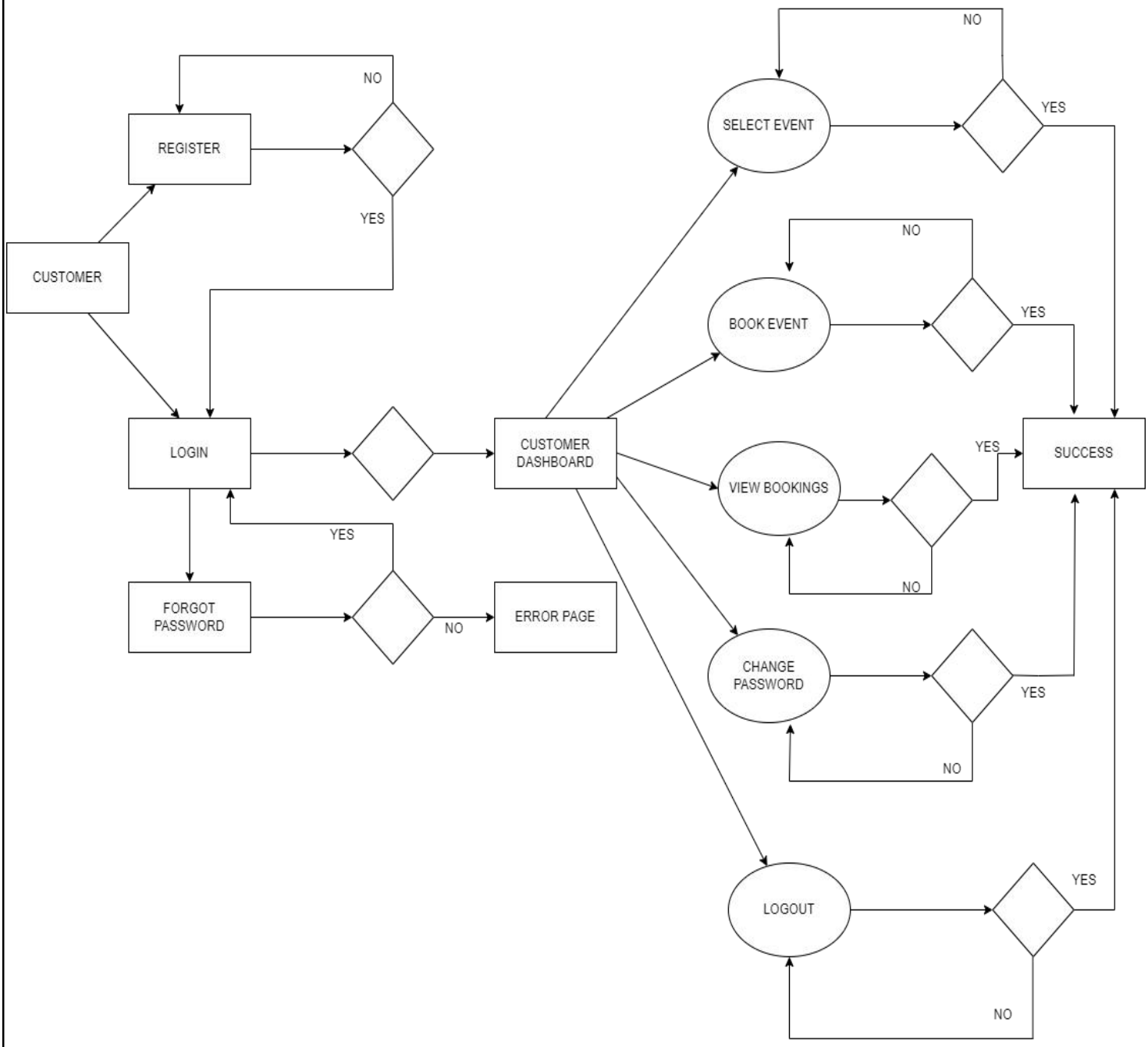


Figure 3 Customer Activity Diagram

- **Customer sign in, sign out, create account**

  This feature is provided to customer so he can sign in, sign out and create account for new customer.

- **Search Event**

  Customer can search the event as per his wish.

Book Event

Customers can book event from his cart by doing payment. .

SYSTEM ANALYSIS

System analysis is the process of gathering and interpreting facts, diagnosing problems, and using the information to recommend improvements on the system. System analysis is a problem-solving activity that requires intensive communication between the system users and system developers.

System analysis or study is an important phase of any system development process. The system is viewed as a whole, the inputs are identified, and the system is subjected to close study to identify the problem areas. The solutions are given as a proposal. The proposal is reviewed on user request and suitable changes are made. This loop ends as soon as the user is satisfied with the proposal.

2.1 EXISTING SYSTEM

The current system for shopping is to visit the shop manually and from the available product choose the item customer want and buying the item by payment of the price of the item.

- ✓ It is less user-friendly.
- ✓ Limited vendors available. .
- ✓ It is difficult to get vendor for small scale event.
- ✓ It is a time-consuming process
- ✓ Not in reach of distant users.

2.2 PROPOSED SYSTEM

The new system is totally micro event based system. A new system provides features like budget friendly small event organisers, user profiles for both customer and vendor. It is the most convenient software application for maintaining a contact between the customer and a small vendor.

2.3 SYSTEM REQUIREMENT SPECIFICATION

2.3.1 GENERAL DESCRIPTION

Product Description:

It is a platform, designing especially for small scale event organizers and customers. The Event Booking System portal provides complete functionality for listing various vendors along with their catalogue

Problem Statement:

The small event organizers had to broadcast about their business on their own as they don't have a public platform to register or advertise themselves. In the existing system customers with small budget for events have to either pay more to book an event organizer or they have to organize on their own.

2.3.2 SYSTEM OBJECTIVES

- To provide a Web application for online shopping of products in an existing shop.
- To provide an online shopping web site for the same shop.

2.3.3 SYSTEM REQUIREMENTS

2.3.3.1 NON-FUNCTIONAL REQUIREMENTS

i. EFFICIENCY REQUIREMENT

When an online shopping cart android application implemented customer can purchase product inan efficient manner.

ii. RELIABILITY REQUIREMENT

The system should provide a reliable environment to both customers and owner. All orders shouldbe reaching at the admin without any errors.

iii. USABILITY REQUIREMENT

The Web application is designed for user friendly environment and ease of use.

iv. IMPLEMENTATION REQUIREMENT

Implementation of the system using React in front end with Spring Boot as back end and it will be used for database connectivity. And the database part is developed by MySQL. Responsive web designing is used for making the website compatible for any type of screen.

v. DELIVERY REQUIREMENT

The whole system is expected to be delivered in four months of time with a weekly Evaluation bythe project guide.

.

2.3.3.2 FUNCTIONAL REQUIREMENTS

USER

- **USER LOGIN**

Description of features

This feature used by the user to login into system. A user must login with his username and password to the system after registration. If they are invalid, the user not allowed to enter the system.

Functional Requirement

- Username and password will be provided after user registration is confirmed.
- Password should be hidden from others while typing it in the field

- **REGISTER NEW**

USER Description of feature

A new user will have to register in the system by providing essential details in order to view the products in the system. The admin must accept new user by unblocking him.

Functional Requirement

- System must be able to verify and validate information.

MODERATOR

Description of features

A moderator is considered as a staff who can manage orders for the time being. As a future update moderator may give facility to add and manage his own products. Moderators can reduce the workload of admin. Now moderator has all the privilege of an admin having except managing other moderators. He can manage users and manage products. He can also check the orders and edit his profile.

Functional Requirement

- The system must identify the login of a moderator.

ADMIN

- **MANAGE USER**

Description of features

The administrator can add user, delete user, view user and block user.

- **MANAGE MODERATOR**

Description of features

The administrator can add moderator, delete moderator, block moderator and search for a moderator.

- **MANAGE EVENTS**

Description of features

The administrator can add events, delete, and view events.

- **MANAGE ORDER**

Description of features

The administrator can view orders and delete orders.

Functional Requirements:

- The system must identify the login of the admin.
- Admin account should be secured so that only owner of the shop can access that account.

MODERATOR

Description of features

A moderator is considered as a staff who can manage orders for the time being. As a future update moderator may give facility to add and manage his own products. Moderators can reduce the workload of admin. Now moderator has all the privilege of an admin having except managing other moderators. He can manage users and manage products. He can also check the orders and edit his profile.

Functional Requirement

- The system must identify the login of a moderator.

SYSTEM DESIGN

System design is the solution for the creation of a new system. This phase focuses on the detailed implementation of the feasible system. Its emphasis on translating design. Specifications to performance specification. System design has two phases of development.

- Logical Design
- Physical Design

During logical design phase the analyst describes inputs (sources), outputs(destinations), databases (data sores) and procedures (data flows) all in a format that meets the user requirements. The analyst also specifies the needs of the user at a level that virtually determines the information flow in and out of the system and the data resources. Here the logical design is done through data flow diagrams and database design. The physical design is followed by physical design or coding. Physical design produces the working system by defining the design specifications, which specify exactly what the candidate system must do. The programmers write the necessary programs that accept input from the user, perform necessary processing on accepted data and produce the required report on a hard copy or display it on the screen.

3.1 INPUT AND OUTPUT DESIGN

3.1.1 INPUT DESIGN:

Input design is the link that ties the information system into the world of its users. The input design involves determining the inputs, validating the data, minimizing the data entry and provides a multi-user facility. Inaccurate inputs are the most common cause of errors in data processing. Errors entered by the data entry operators can be controlled by input design. The user-originated inputs are converted to a computer-based format in the input design. Input data are collected and organized into groups of similar data. Once identified, the appropriate input media are selected for processing. All the input data are validated and if any data violates any conditions, the user is warned by a message. If the data satisfies all the conditions, it is transferred to the appropriate tables in the database. In this project the student details are to be entered at the time of registration. A page is designed for this purpose which is user friendly and easy to use. The design is done such that users get appropriate messages when exceptions occur.

3.1.2 OUTPUT DESIGN:

Computer output is the most important and direct source of information to the user. Output design

is a very important phase since the output needs to be in an efficient manner. Efficient and intelligible output design improves the system relationship with the user and helps in decision making. Allowing the user to view the sample screen is important because the user is the ultimate judge of the quality of output. The output module of this system is the selected notifications.

DATABASE DESIGN

3.2 DATABASE

Databases are the storehouses of data used in the software systems. The data is stored in tables inside the database. Several tables are created for the manipulation of the data for the system. Two essential settings for a database are

- Primary key - the field that is unique for all the record occurrences
- Foreign key - the field used to set relation between tables

Normalization is a technique to avoid redundancy in the tables.

3.3 SYSTEM TOOLS

The various system tools that have been used in developing both the front end and the back end of the project are being discussed in this chapter.

3.3.1 FRONT END:

React is a library which is developed by Facebook are utilized to implement the frontend. React (also known as React.js or ReactJS) is a free and open-source front-end JavaScript library for building user interfaces or UI components. It is maintained by Facebook and a community of individual developers and companies. React can be used as a base in the development of single page or mobile applications. However, React is only concerned with state management and rendering that state to the DOM, so creating React applications usually requires the use of additional libraries for routing, as well as certain client-side functionality.

3.3.2 BACKEND:

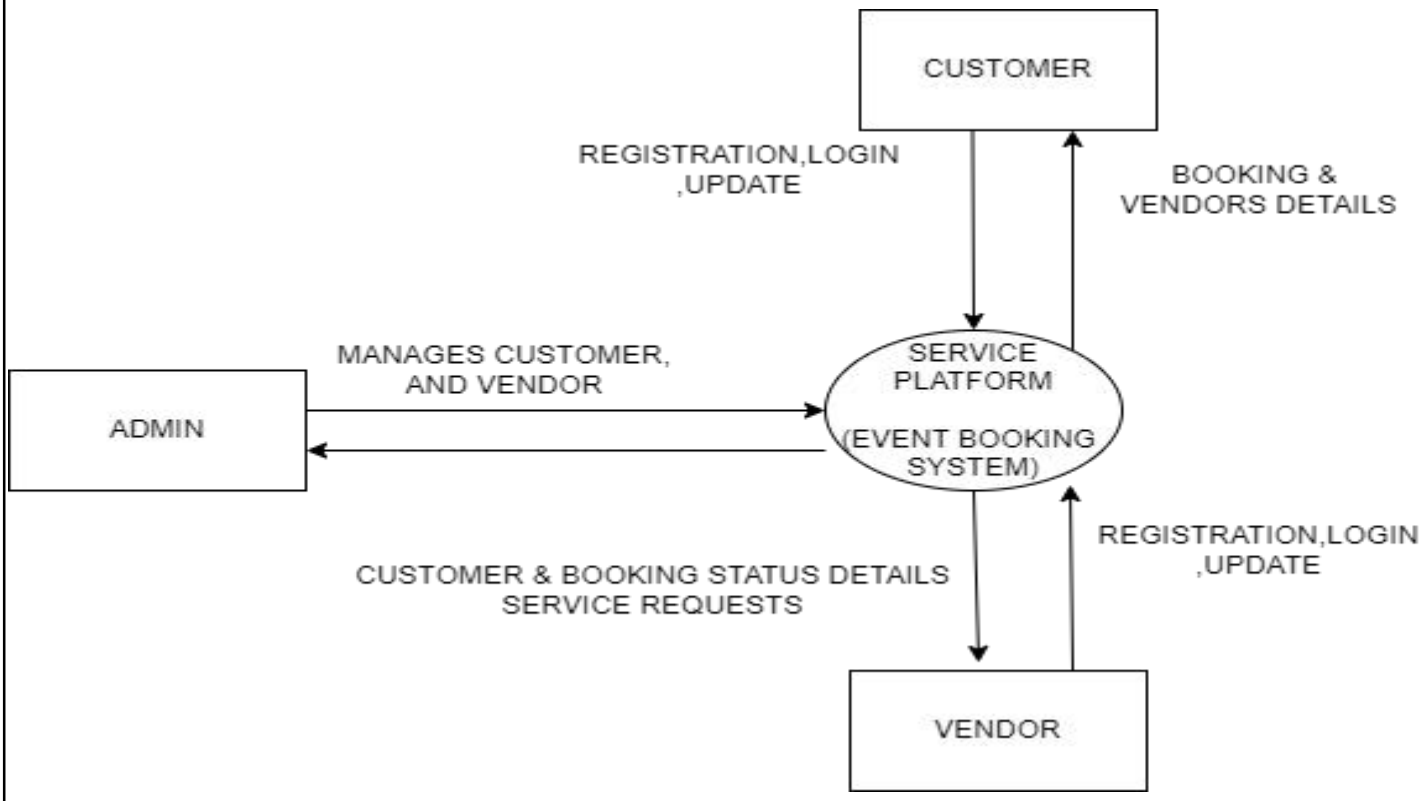
The back end is implemented using MySQL which is used to design databases.

MySQL:

MySQL is the world's second most widely used open-source relational database management system (RDBMS). The SQL phrase stands for Structured Query Language. An application software called Navicert was used to design the tables in MySQL.

Spring-Boot:

This is used to connect MYSQL and fetch data from database and store the data in database. The Spring Framework is an application framework and inversion of control container for the Java platform. The framework's core features can be used by any Java application, but there are extensions for building web applications on top of the Java EE (Enterprise Edition) platform. Although the framework does not impose any specific programming model, it has become popular in the Java community as an addition to the Enterprise JavaBeans (EJB) model. The Spring Framework is Open-source Framework.



Zero Level DFD

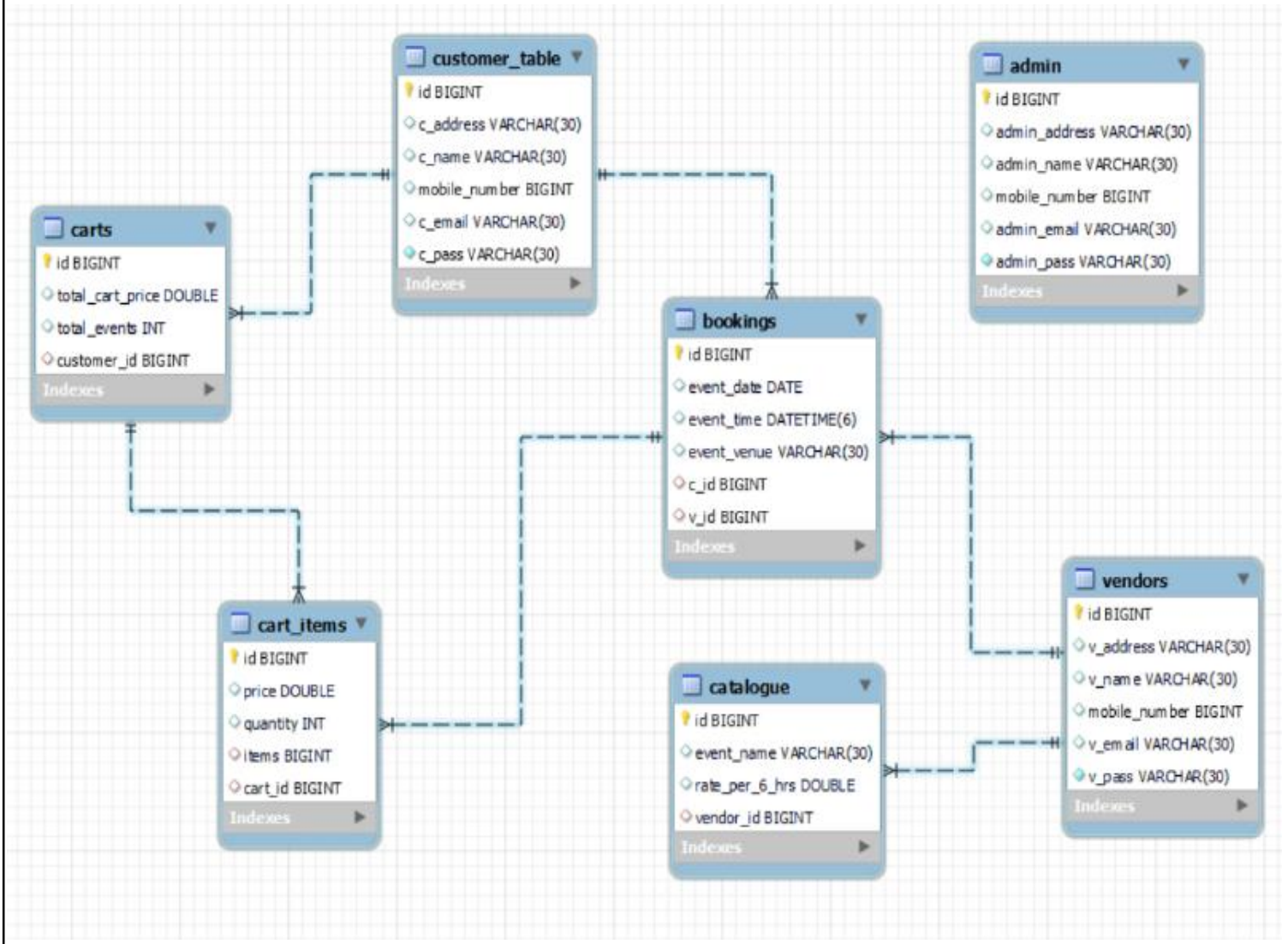


Figure 7 E-R Diagram

TABLE STRUCTURE:

TABLES:

Tables_in_bookmyevent
admin
bookings
cart_items
carts
catalogue
customer_table
vendors

Admin:

Field	Type	Null	Key	Default	Extra
id	bigint	NO	PRI	NULL	auto_increment
admin_address	varchar(30)	YES		NULL	
admin_name	varchar(30)	YES		NULL	
mobile_number	bigint	YES		NULL	
admin_email	varchar(30)	YES	UNI	NULL	
admin_pass	varchar(30)	NO		NULL	

Customer:

```
mysql> desc customer_table;
+---------------+--------------+------+-----+---------+----------------+
| Field         | Type         | Null | Key | Default | Extra          |
+---------------+--------------+------+-----+---------+----------------+
| id            | bigint       | NO   | PRI | NULL    | auto_increment |
| c_address     | varchar(30)  | YES  |     | NULL    |                |
| c_name        | varchar(30)  | YES  |     | NULL    |                |
| mobile_number | bigint       | YES  |     | NULL    |                |
| c_email       | varchar(30)  | YES  | UNI | NULL    |                |
| c_pass        | varchar(300) | NO   |     | NULL    |                |
+---------------+--------------+------+-----+---------+----------------+
6 rows in set (0.00 sec)
```

Vendor:

```
mysql> desc vendors;
+---------------+--------------+------+-----+---------+----------------+
| Field         | Type         | Null | Key | Default | Extra          |
+---------------+--------------+------+-----+---------+----------------+
| id            | bigint       | NO   | PRI | NULL    | auto_increment |
| v_address     | varchar(30)  | YES  |     | NULL    |                |
| v_name        | varchar(30)  | YES  |     | NULL    |                |
| mobile_number | bigint       | YES  |     | NULL    |                |
| v_email       | varchar(30)  | YES  | UNI | NULL    |                |
| v_pass        | varchar(300) | NO   |     | NULL    |                |
+---------------+--------------+------+-----+---------+----------------+
6 rows in set (0.00 sec)
```

Catelogue

```
mysql> desc catalogue;
+---------------+-------------+------+-----+---------+----------------+
| Field         | Type        | Null | Key | Default | Extra          |
+---------------+-------------+------+-----+---------+----------------+
| id            | bigint      | NO   | PRI | NULL    | auto_increment |
| event_name    | varchar(30) | YES  | UNI | NULL    |                |
| cat_image     | varchar(300)| YES  |     | NULL    |                |
| rate_per_6_hrs| double      | YES  |     | NULL    |                |
| vendor_id     | bigint      | NO   | MUL | NULL    |                |
+---------------+-------------+------+-----+---------+----------------+
5 rows in set (0.00 sec)
```

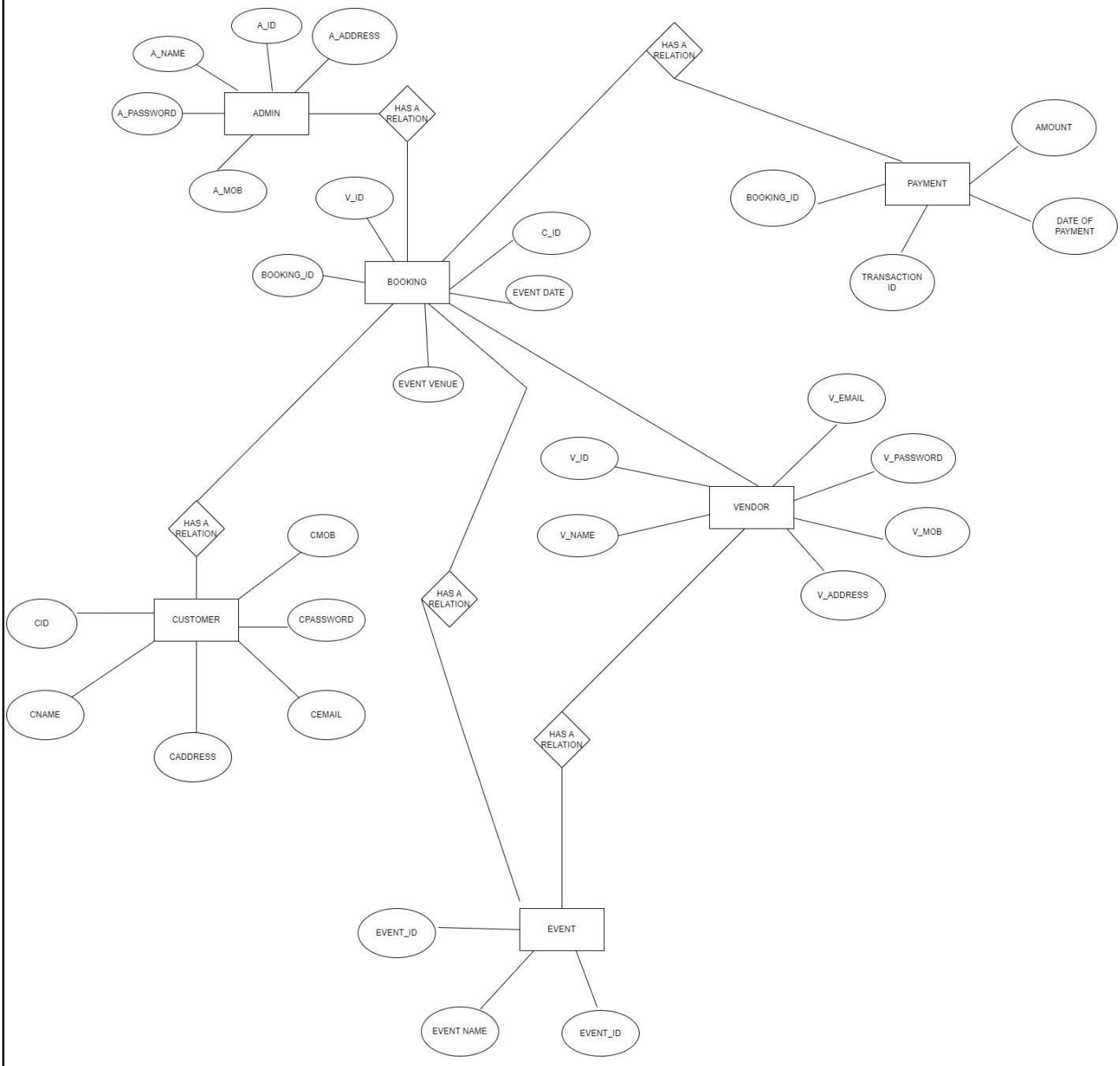
Booking

```
mysql> desc booking;
+-------------+--------------+------+-----+---------+----------------+
| Field       | Type         | Null | Key | Default | Extra          |
+-------------+--------------+------+-----+---------+----------------+
| id          | bigint       | NO   | PRI | NULL    | auto_increment |
| dobooking   | datetime(6)  | YES  |     | NULL    |                |
| event_date  | date         | YES  |     | NULL    |                |
| status      | varchar(255) | YES  |     | NULL    |                |
| event_venue | varchar(30)  | YES  |     | NULL    |                |
| catalog_id  | bigint       | YES  | MUL | NULL    |                |
| c_id        | bigint       | YES  | MUL | NULL    |                |
| payment_id  | bigint       | YES  | MUL | NULL    |                |
| v_id        | bigint       | YES  | MUL | NULL    |                |
+-------------+--------------+------+-----+---------+----------------+
```

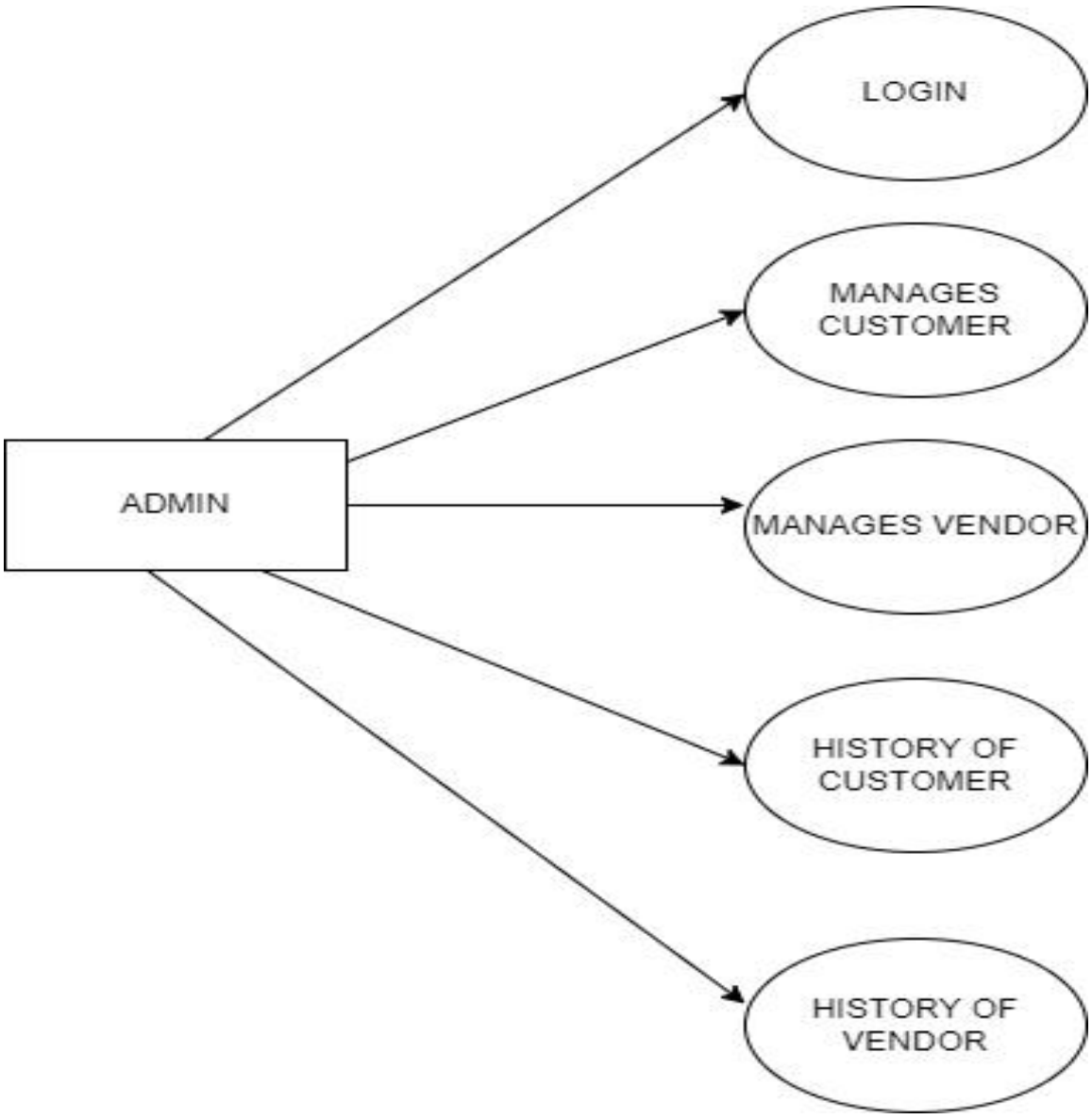
Cart Items:

Field	Type	Null	Key	Default	Extra
id	bigint	NO	PRI	NULL	auto_increment
price	double	YES		NULL	
quantity	int	YES		NULL	
items	bigint	YES	MUL	NULL	
cart_id	bigint	YES	MUL	NULL	

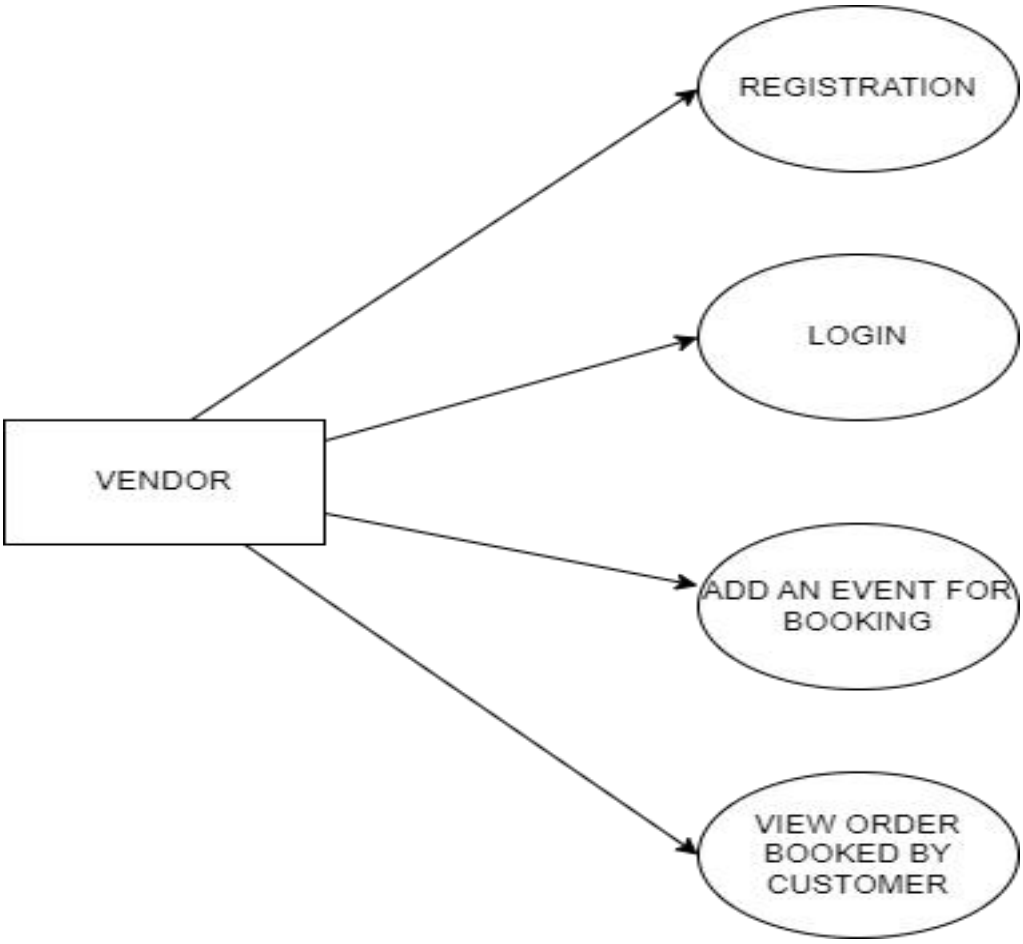
Project Diagrams:



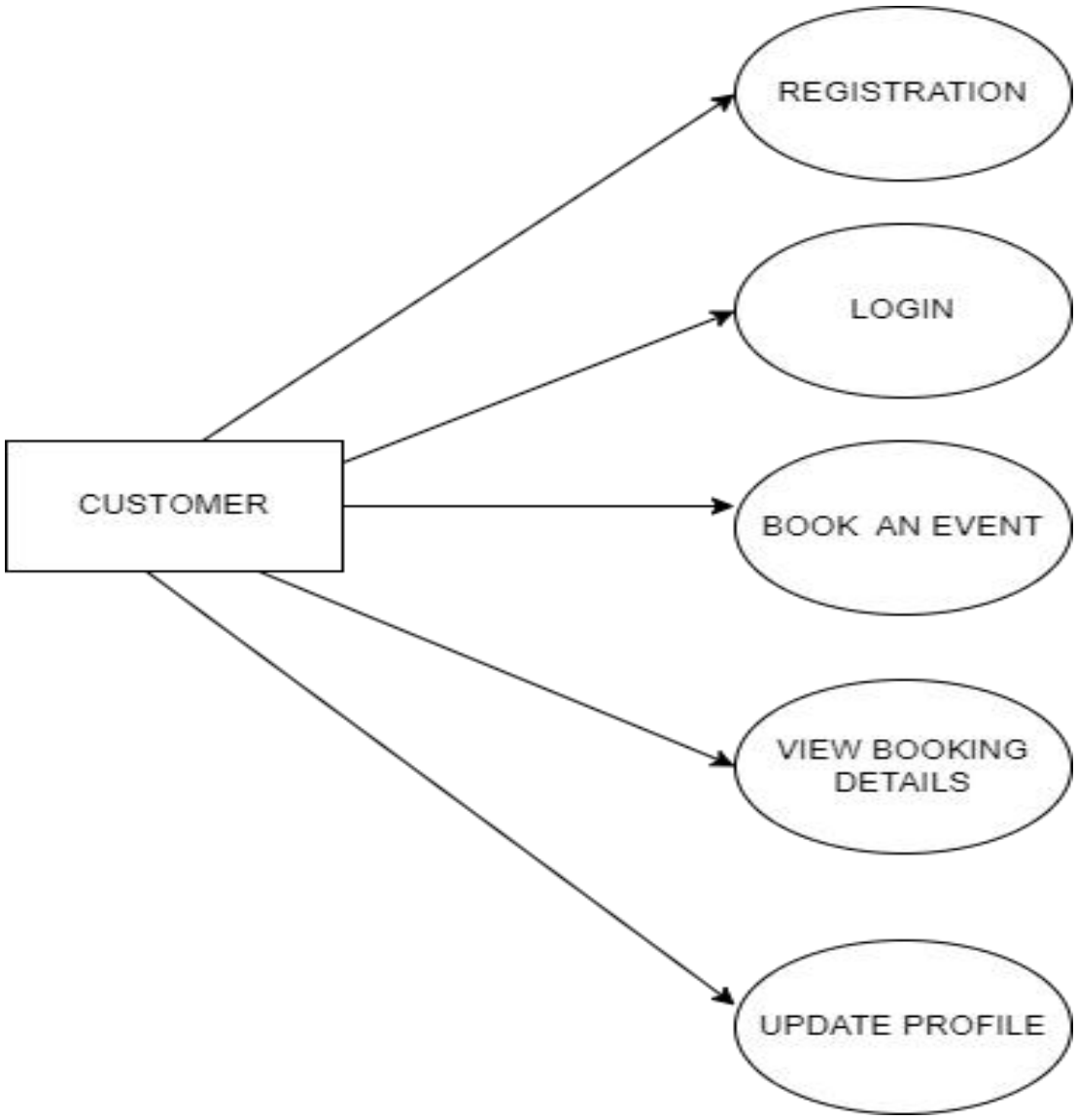
E-R DIAGRAM



ADMIN USECASE DIAGRAM



VENDOR USECASE DIAGRAM



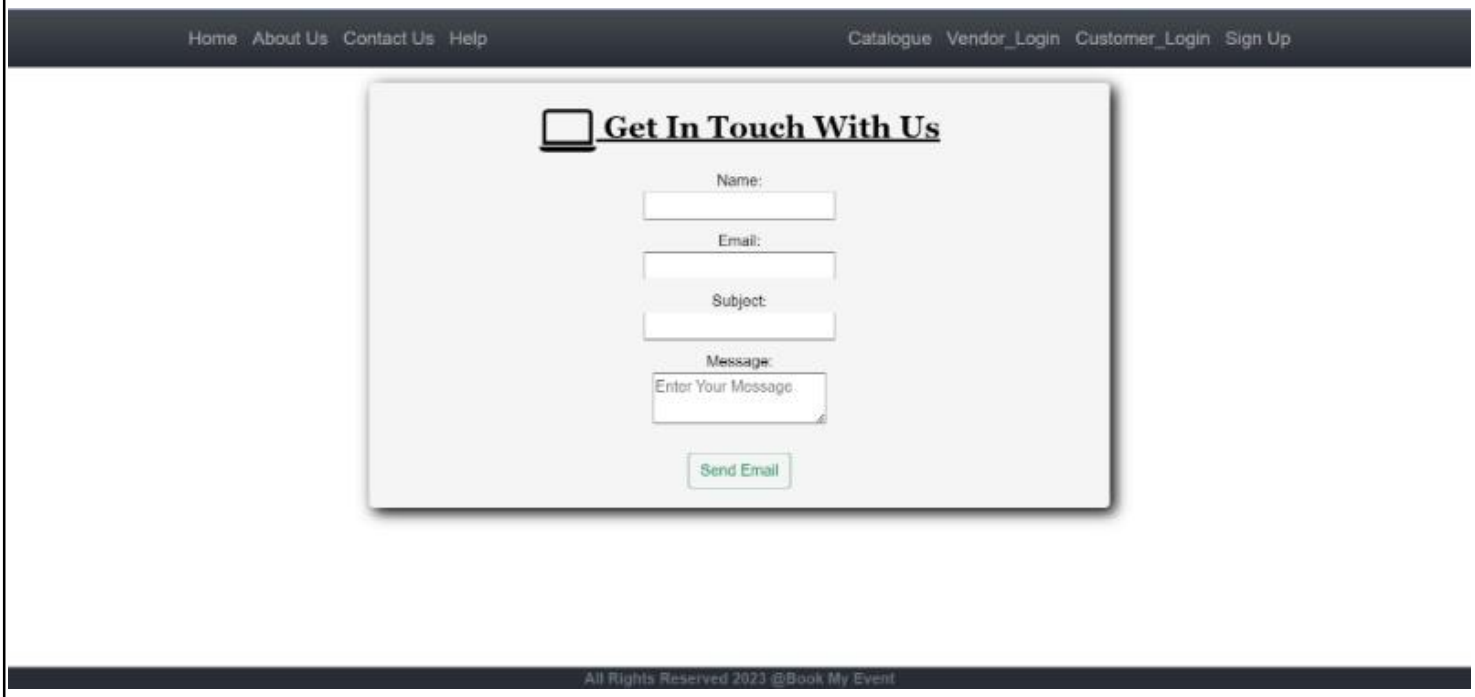
CUSTOMER USECASE DIAGRAM

UI SCREENSHORTS:

and ensure the success of the event.

Technologically Protected

- Our technology is based on SpringBoot which has a platform-independent build and that helps deploying Web-based enterprise applications online swiftly in no time.
- React as a frontend is a free and open-source front-end JavaScript library for building user interfaces or UI components.



Vendor

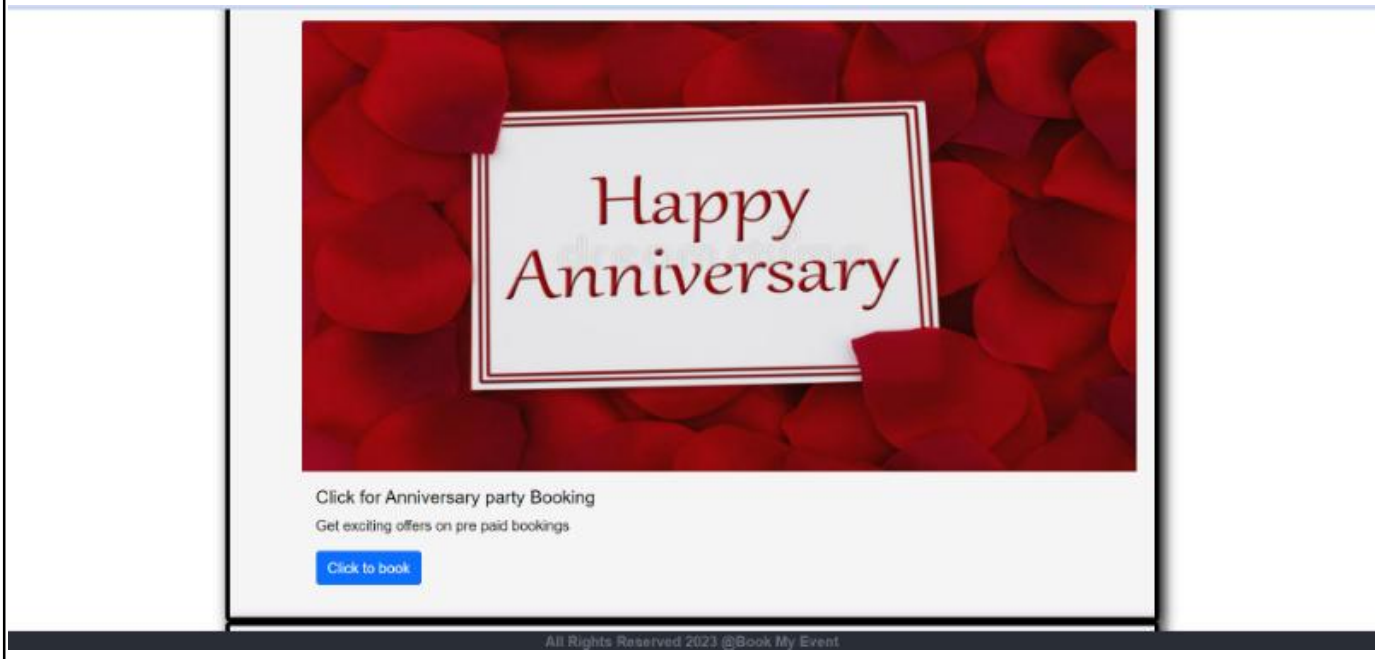
The Customer can login and Start his search for budget friendly events after filling the form.

Contact Us

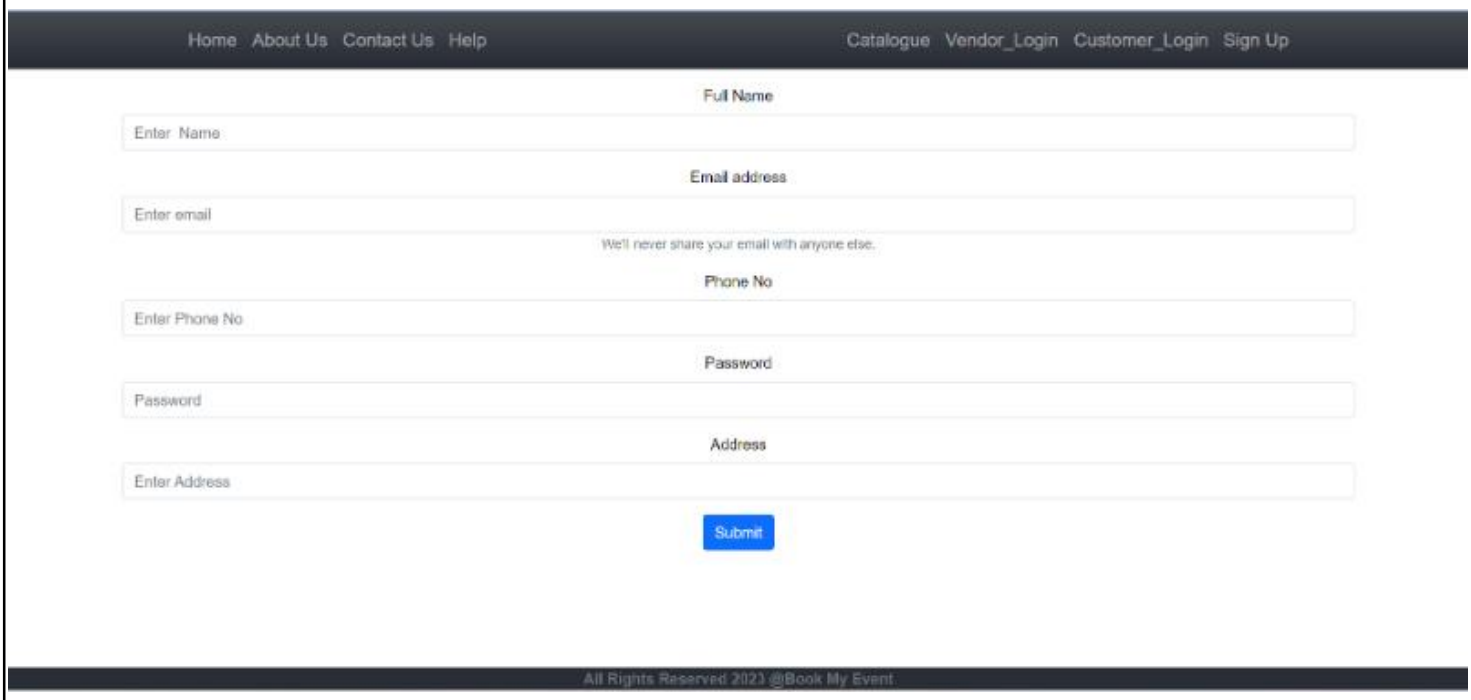
Click for Baby showers Booking

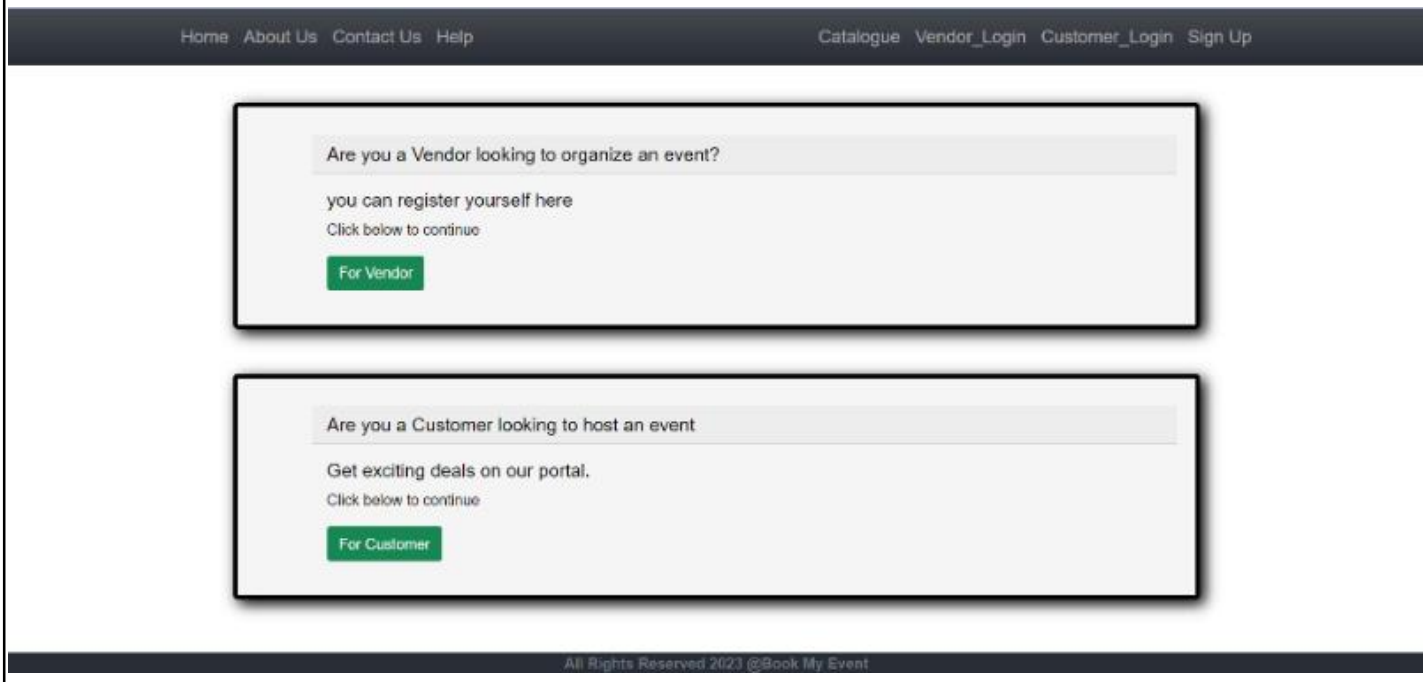
Get exciting offers on pre paid bookings

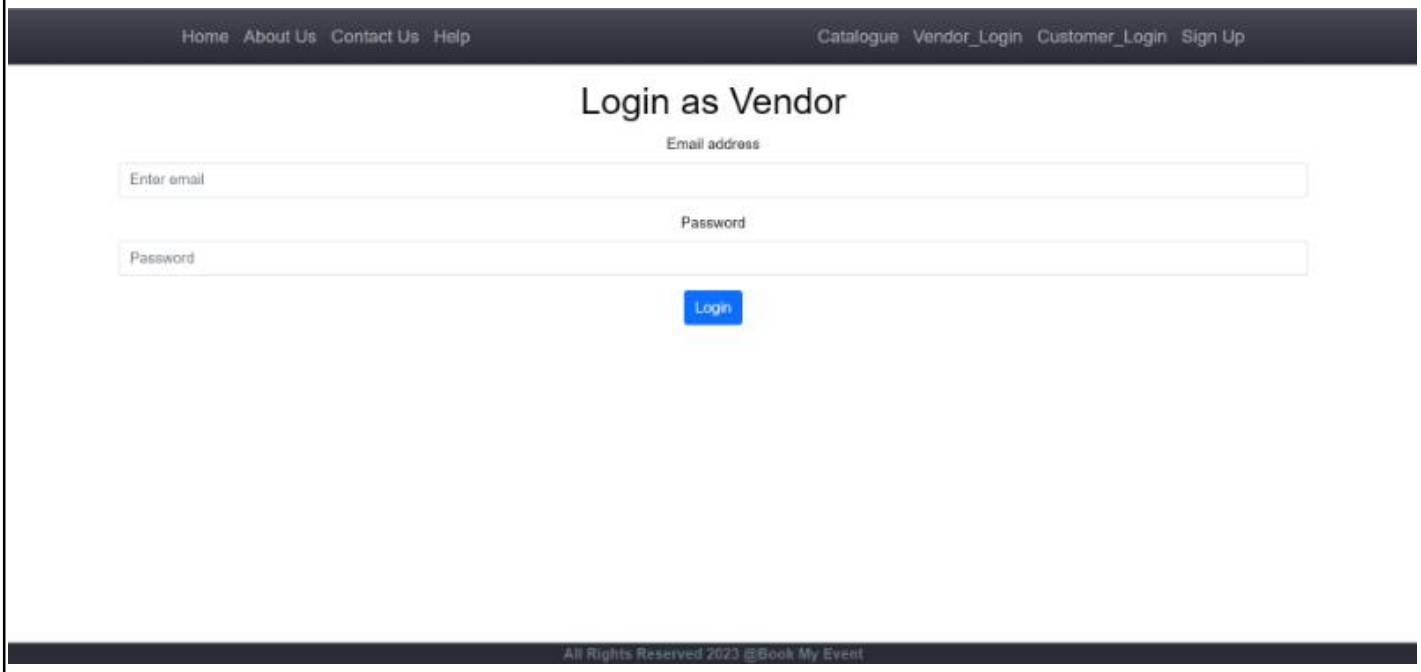
Click to book


Happy Anniversary
Click for Anniversary party Booking
Get exciting offers on pre paid bookings
Click to book
All Rights Reserved 2023 @Book My Event


Home About Us Contact Us Help
Catalogue Vendor_Login Customer_Login Sign Up
Happy Birthday
Click for Birthday Booking
Get exciting offers on pre paid bookings
All Rights Reserved 2023 @Book My Event


Home About Us Contact Us Help
Catalogue Vendor_Login Customer_Login Sign Up
Full Name
Enter Name
Email address
Enter email
We'll never share your email with anyone else.
Phone No
Enter Phone No
Password
Password
Address
Enter Address
Submit
All Rights Reserved 2023 @Book My Event


Home About Us Contact Us Help
Catalogue Vendor_Login Customer_Login Sign Up
Are you a Vendor looking to organize an event?
you can register yourself here
Click below to continue
For Vendor
Are you a Customer looking to host an event
Get exciting deals on our portal.
Click below to continue
For Customer
All Rights Reserved 2023 @Book My Event


Home About Us Contact Us Help
Catalogue Vendor_Login Customer_Login Sign Up
Login as Vendor
Email address
Enter email
Password
Password
Login
All Rights Reserved 2023 @Book My Event

CONCLUSION

The project entitled **Event Booing System** was completed successfully.

The system has been developed with much care and free of errors and at the same time it is efficient and less time consuming. The purpose of this project was to develop a web application and an android application for purchasing items from a shop.
This project helped us in gaining valuable information and practical knowledge on several topics like designing web pages using React.js, usage of responsive templates, designing of android applications, and management of database using MySQL. The entire system is secured. Also, the project helped us understanding about the development phases of a project and software development life cycle. We learned how to test different features of a project.
This project has given us great satisfaction in having designed an application which can be implemented to any nearby shops or branded shops selling various kinds of products by simple modifications.

There is a scope for further development in our project to a great extent. A number of features can be added to this system in future like providing moderator more control over products so that each moderator can maintain their own products. Another feature we wished to implement was providing classes for customers so that different offers can be given to each class. System may keep track of history of purchases of each customer and provide suggestions based on their history. These features could have implemented unless the time did not limit us.

REFERENCES

[1] JavaScript Enlightenment, Cody Lindley-First Edition, based on JavaScript 1.5, ECMA-262, Edition

[2] Mc Graw Hill's, Java: The complete reference 7thEdition, Herbert Scheldt

[3] Complete CSS Guide, Maxine Sherrin and John Allsopp-O'ReillyMedia; September 2012

ONLINE REFERENCE

[1] www.Google.com
[2] www.w3school.com
[3] www.javatpoint.com