Project Report On

“Event Organizing Web Application”

Submitted in partial fulfillment for the award of

Post Graduate Diploma in Advanced Computing

from

C-DAC ACTS (Pune)

Guided by

Mr. Doppa Srinivas

Presented By

Shubham Bare – 240340120198

Amruta Vaidya- 240340120023

Shital Shelke – 240340120182

Soham Bugad- 240340120053

Pawan Jenekar- 240340120127

Centre of Development of Advanced Computing (C-DAC), Pune

CERTIFICATE

TO WHOMSOEVER IT MAY CONCERN

This is to certify that,

Shubham Bare – 240340120198

Amruta Vaidya- 240340120023

Shital Shelke – 240340120182

Soham Bugad- 240340120053

Pawan Jenekar- 240340120127

have successfully completed their project titled

"Event Organizing Web Application"

Under the Guidance of Mr. Doppa Srinivas

Project Guide **HOD ACTS**

ACKNOWLEDGEMENT

This project **"Event Organizing Web Application"** was a great learning experience for us and we are submitting this work to Advanced Computing Training School (CDAC ACTS).

We all are very glad to mention the name of **Mr. Doppa Srinivas** for his valuable guidance to work on this project. His guidance and support helped us to overcome various obstacles and intricacies during the course of project work.

Our most heartfelt thank goes to Mrs **Swati Salunkhe** (Course Coordinator, PG-DAC) who gave all the required support and kind coordination to provide all the necessities like required hardware, internet facility and extra Lab hours to complete the project and throughout the course up to the last day here in C-DAC ACTS, Pune.

Shubham Bare - 240340120198
Amruta Vaidya - 240340120023
Shital Shelke - 240340120182
Soham Bugad - 240340120053
Pawan Jenekar - 240340120127

TABLE OF CONTENTS

1. Introduction

Developed an Event Organizing Web Application that facilitates seamless event management for admins, customers, and vendors. The front end is built using ReactJS, with Redux for state management, Axios for API requests, and TailwindCSS for styling. The back end is powered by Spring Boot, with secure authentication and authorization implemented using Spring Security with JWT tokens, and MySQL is used as the Database Management System. To ensure the reliability of the application, we utilized JUnit for unit testing and Postman for integration testing. This project exemplifies a full-stack development approach, delivering a secure, scalable, and user-friendly platform.

The "Event Organizing Web Application" project is a comprehensive Spring Boot-based backend solution designed to efficiently manage event-related processes, including user authentication, event creation, booking management, and payment processing. Utilizing Java technologies, this system provides secure and scalable functionalities through well-defined components such as controllers, services, repositories, and DTOs, each serving a specific role in handling requests, business logic, and data persistence. By implementing robust security measures with JWT, centralizing exception handling, and incorporating data validation and utility functions, the project ensures a reliable and user-friendly platform for organizing and managing events effectively. This architecture not only supports seamless integration and scalability but also guarantees secure, efficient management of user interactions and event operations.

2.Software/Hardware Requirement

Server:

Processor: Intel Core i5 or equivalent AMD processor.

RAM: Minimum 8GB RAM.

Storage: SSD storage for improved performance.

Network: Ethernet or Wi-Fi connectivity.

Operating System: Windows 10 and above version preferred for server deployment.

Client Devices:

Processor: Dual-core processor or higher.

RAM: Minimum 4GB RAM.

Storage: Sufficient storage for caching and local data.

Network: Ethernet or Wi-Fi connectivity.

Browser: Compatible with latest versions of popular browsers like Google Chrome, Mozilla Firefox, and Safari.

3. Tools and Technologies used

- SpringBoot
- SpringDataJPA
- RESTful Web
- Node JS
- SpringWeb
- MYSQL
- JWT
- Git
- Spring Security
- React JS
- Tailwind CSS
- Axios
- Paypal Payment Integration
- JUnit Testing

1. Spring Boot: Utilized to develop the backend of the application, providing a robust framework for building Java-based web applications with ease.

2. Spring Data JPA: Implemented for data access, allowing seamless interaction with the MySQL database to store and retrieve sports data efficiently.

3. RESTful Web Services: In the context of an e-commerce web application like Book Charm, RESTful web services play a crucial role in facilitating communication between the frontend and backend components. These services adhere to the principles of Representational State Transfer (REST), which emphasizes a stateless, standardized approach for building web services

4. Node JS: Employed for web scraping, enabling the application to extract live scores andmatch details from various sports websites and APIs.

5. Spring Web: Used for handling web requests and responses, managing controllers, and serving static resources to the frontend.

6. Aiven cloud database MySQL: Chosen as the relational database management system to store data.

7. JWT (JSON Web Tokens): Implemented for secure user authentication and authorization, ensuring that only authenticated users can access into account and buy books and seller can add books .

8. Axios: In the context of a web application like Web Organizing Event Application, Axios is likely used asa client-side HTTP library. Axios simplifies the process of making asynchronous HTTP requests from the frontend (React.js) to the backend (Node.js/Express.js). It isinstrumental in fetching data from the server, handling API calls, and facilitating smooth communication between the frontend and backend components, ensuring efficient data retrieval and seamless user interactions in the e-commerce application

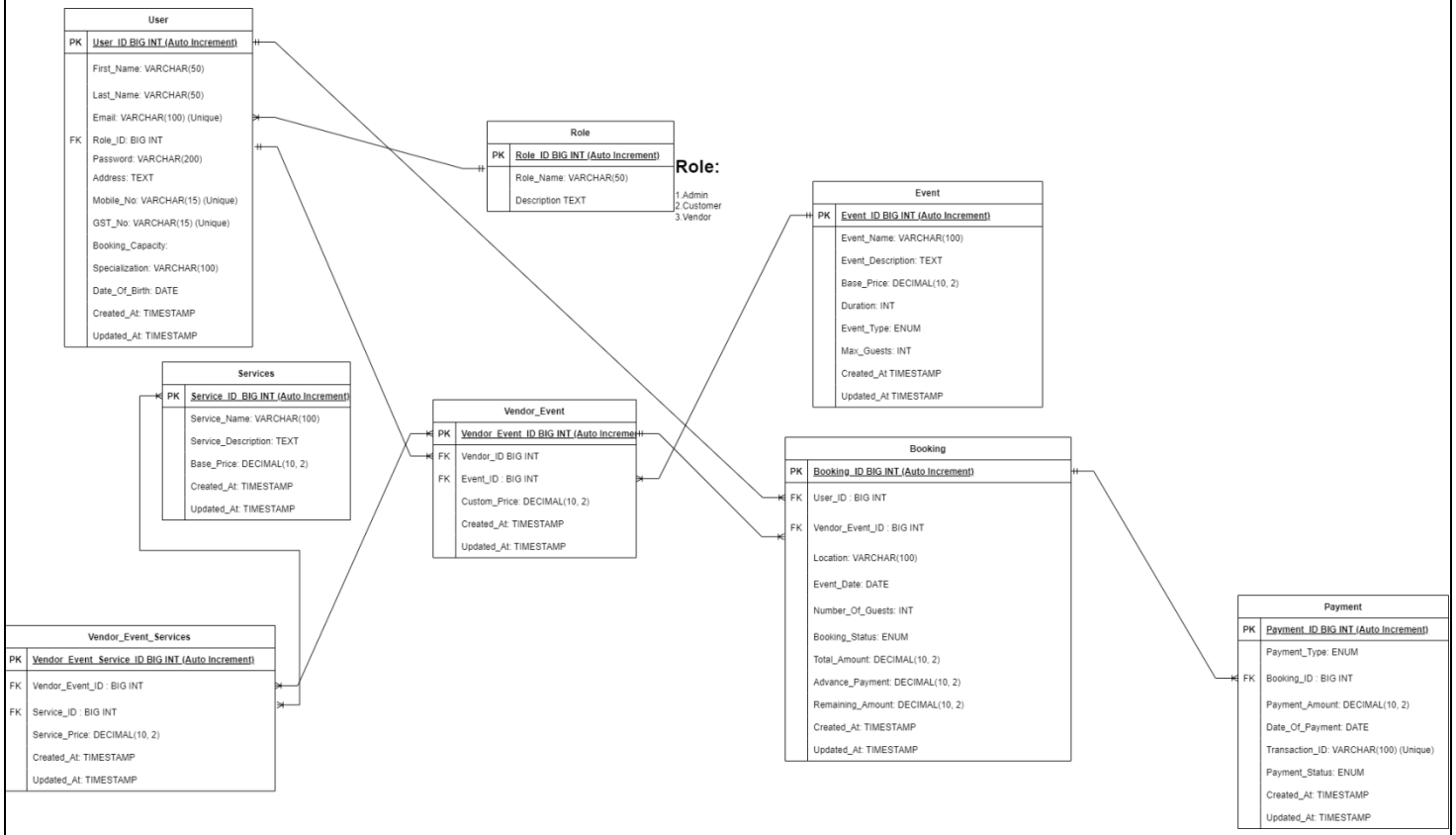
9. React: Employed to build the frontend of the application, offering a component-based architecture for creating dynamic and interactive user interfaces.

10. Tailwind CSS: Leveraged to enhance the visual appeal and user experience of the application by incorporating Tailwind CSS

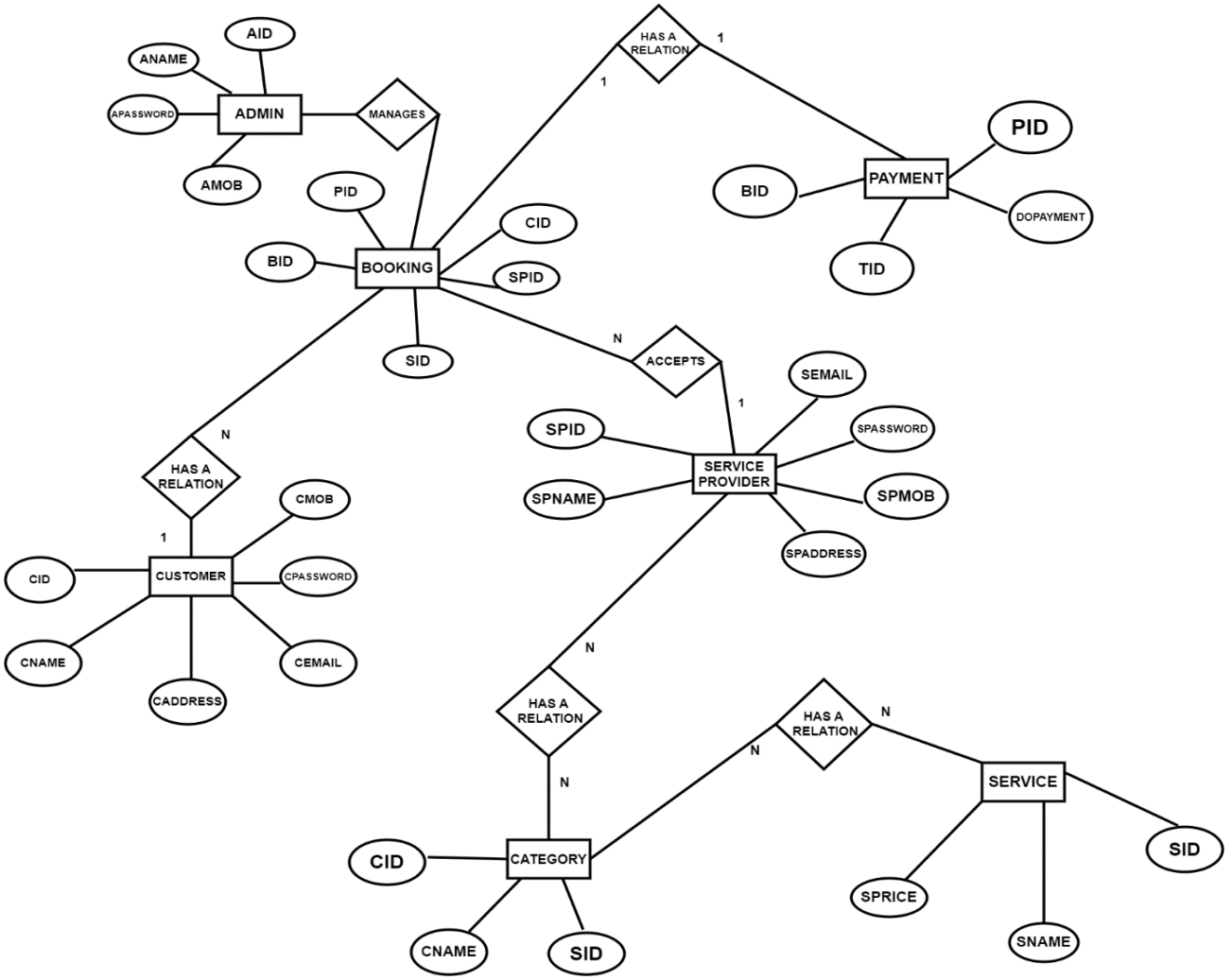
11. Git: Implemented as a version control system to track changes in the source code, enabling collaboration among developers, and facilitating code management and deployment workflows.

12. Paypal Payment Integration: Paypal is integrated as the payment gateway, enabling secure and streamlined online transactions. Paypal provides a developer-friendly API, allowing seamlessintegration for processing payments, managing subscriptions, and ensuring areliable end-to-end payment experience for users in the e-commerce web application

4. Project Database Diagram



5.Project E-R(Entity relationship) Diagram



6.Table Structure

Booking table:

Field	Type	Null	Key	Default	Extra
booking_id	bigint	NO	PRI	NULL	auto_increment
advance_payment	decimal(19,2)	NO		NULL	
booking_status	varchar(255)	NO		NULL	
created_at	timestamp	YES		CURRENT_TIMESTAMP	DEFAULT_GENERATED
event_date	date	NO		NULL	
location	varchar(255)	NO		NULL	
number_of_guests	int	NO		NULL	
remaining_amount	decimal(19,2)	NO		NULL	
total_amount	decimal(19,2)	NO		NULL	
updated_at	timestamp	YES		CURRENT_TIMESTAMP	DEFAULT_GENERATED on update CURRENT_TIMESTAMP
user_id	bigint	NO	MUL	NULL	
vendor_event_id	bigint	NO	MUL	NULL	

Events table:

Field	Type	Null	Key	Default	Extra
event_id	bigint	NO	PRI	NULL	auto_increment
base_price	decimal(19,2)	NO		NULL	
created_at	timestamp	YES		CURRENT_TIMESTAMP	DEFAULT_GENERATED
duration	int	NO		NULL	
event_description	varchar(255)	YES		NULL	
event_name	varchar(255)	NO		NULL	
event_type	varchar(255)	NO		NULL	
max_guests	int	NO		NULL	
updated_at	timestamp	YES		CURRENT_TIMESTAMP	DEFAULT_GENERATED on update CURRENT_TIMESTAMP

Payments Table:

Field	Type	Null	Key	Default	Extra
payment_id	int	NO	PRI	NULL	auto_increment
created_at	timestamp	YES		CURRENT_TIMESTAMP	DEFAULT_GENERATED
date_of_payment	date	NO		NULL	
payment_amount	decimal(19,2)	NO		NULL	
payment_status	varchar(255)	NO		NULL	
payment_type	varchar(255)	NO		NULL	
transaction_id	varchar(255)	NO	UNI	NULL	
updated_at	timestamp	YES		CURRENT_TIMESTAMP	DEFAULT_GENERATED on update CURRENT_TIMESTAMP
booking_id	bigint	NO	MUL	NULL	

Roles Table:

Field	Type	Null	Key	Default	Extra
role_id	bigint	NO	PRI	NULL	auto_increment
description	varchar(255)	YES		NULL	
role_name	varchar(255)	YES	UNI	NULL	

Services Table:

Field	Type	Null	Key	Default	Extra
service_id	bigint	NO	PRI	NULL	auto_increment
base_price	decimal(19,2)	NO		NULL	
created_at	timestamp	YES		CURRENT_TIMESTAMP	DEFAULT_GENERATED
service_description	varchar(255)	YES		NULL	
service_name	varchar(255)	NO		NULL	
updated_at	timestamp	YES		CURRENT_TIMESTAMP	DEFAULT_GENERATED on update CURRENT_TIMESTAMP

User Table:

Field	Type	Null	Key	Default	Extra
user_id	bigint	NO	PRI	NULL	auto_increment
address	varchar(255)	YES		NULL	
booking_capacity	int	YES		NULL	
created_at	timestamp	YES		CURRENT_TIMESTAMP	DEFAULT_GENERATED
date_of_birth	date	YES		NULL	
email	varchar(255)	YES	UNI	NULL	
first_name	varchar(255)	YES		NULL	
gst_no	varchar(255)	YES	UNI	NULL	
last_name	varchar(255)	YES		NULL	
mobile_no	varchar(255)	YES	UNI	NULL	
password	varchar(255)	YES		NULL	
specialization	varchar(255)	YES		NULL	
updated_at	timestamp	YES		CURRENT_TIMESTAMP	DEFAULT_GENERATED on update CURRENT_TIMESTAMP
role_id	bigint	NO	MUL	NULL	

Vendor Event Services Table:

Field	Type	Null	Key	Default	Extra
vendor_event_service_id	bigint	NO	PRI	NULL	auto_increment
created_at	timestamp	YES		CURRENT_TIMESTAMP	DEFAULT_GENERATED
service_price	decimal(19,2)	NO		NULL	
updated_at	timestamp	YES		CURRENT_TIMESTAMP	DEFAULT_GENERATED on update CURRENT_TIMESTAMP
service_id	bigint	NO	MUL	NULL	
vendor_event_id	bigint	NO	MUL	NULL	

Vendor Events Table:

Field	Type	Null	Key	Default	Extra
vendor_event_id	bigint	NO	PRI	NULL	auto_increment
created_at	timestamp	YES		CURRENT_TIMESTAMP	DEFAULT_GENERATED
custom_price	decimal(19,2)	NO		NULL	
updated_at	timestamp	YES		CURRENT_TIMESTAMP	DEFAULT_GENERATED on update CURRENT_TIMESTAMP
event_id	bigint	NO	MUL	NULL	
vendor_id	bigint	NO	MUL	NULL	

7. Advantages

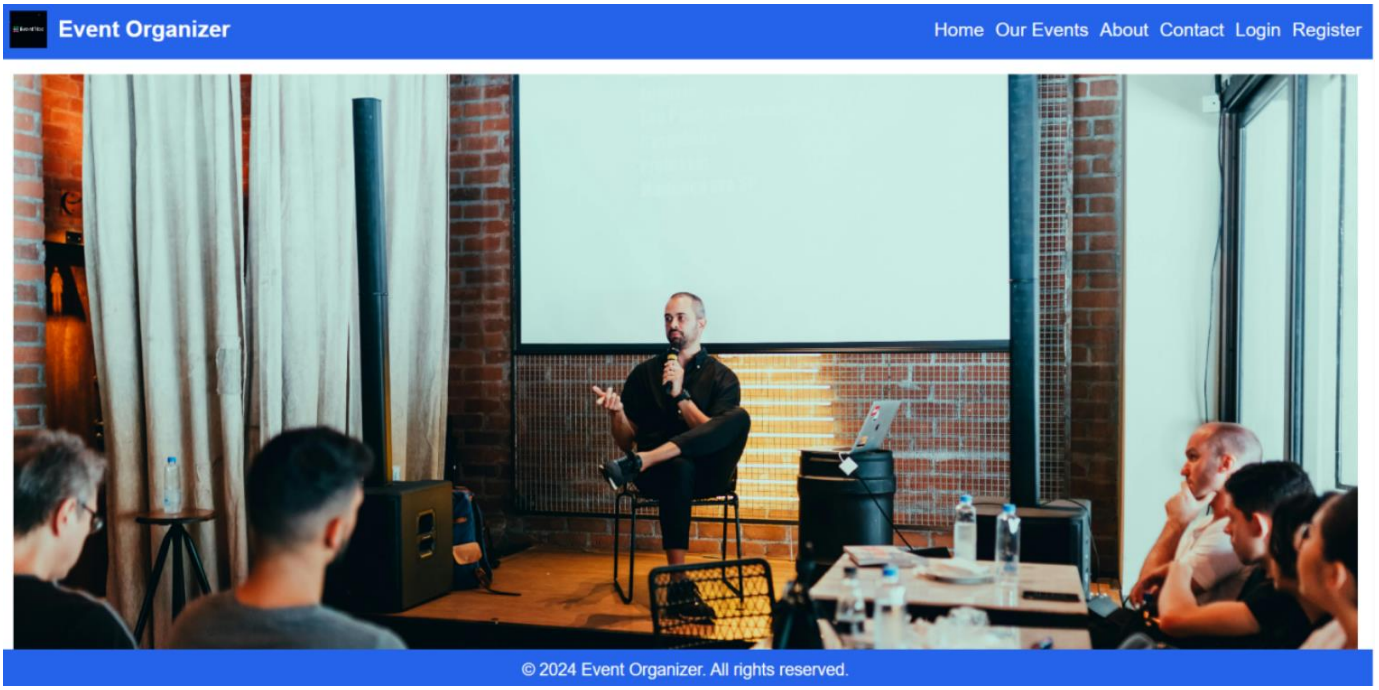
- Use of MySQL RDBMS
  - Data Storage and Management:MySQL stores and manages structured data in tables consisting of rows and columns, which are organized into databases.
  - SQL Queries:Supports SQL (Structured Query Language) for querying and manipulating data, including SELECT, INSERT, UPDATE, and DELETE operations.
  - Data Integrity:Enforces data integrity through constraints like PRIMARY KEY, FOREIGN KEY, UNIQUE, and NOT NULL.
  - Transaction Management: Supports transactions with commands like BEGIN, COMMIT, and ROLLBACK, ensuring ACID (Atomicity, Consistency, Isolation, Durability) properties.
  - Scalability:Can handle large databases and supports vertical and horizontal scaling to accommodate growing amounts of data.
  - Performance Optimization:Offers indexing, query caching, and optimization tools for enhancing the speed and performance of database operations.
  - User Management and Security:Provides user authentication, authorization, and role-based access control to secure data from unauthorized access.
  - Replication:Supports data replication across multiple servers, enabling data redundancy, load balancing, and failover capabilities.
  - Backup and Recovery:Includes tools for data backup and recovery, helping to prevent data loss and ensure continuity.
  - Open Source and Community Support:MySQL is open-source, with a large community offering extensive support, documentation, and plugins.
  - Cross-Platform Compatibility:MySQL runs on various operating systems, including Windows, Linux, and macOS, ensuring flexibility in deployment.
  - Integration with Other Technologies:Easily integrates with web technologies (e.g., PHP, Node.js), frameworks (e.g., Laravel, Django), and other databases.

- Data Migration:Supports data migration and import/export through tools like mysqldump and LOAD DATA.
- Use Cases:Commonly used in web development, content management systems (CMS), e-commerce platforms, and enterprise applications for managing user data, transactions, and content.
- Cost-Effective:Offers a cost-effective solution for database management, particularly in its community edition, which is free to use.

- Use of JWT for authorization
  - Stateless Authentication: JWT allows for stateless authentication, meaning server-side sessions or database lookups for authentication are not required, resulting in reduced server load and improved scalability.
  - Enhanced Security: JWTs are digitally signed, ensuring data integrity and preventing tampering or unauthorized access to user data. Additionally, since JWTs do not store sensitive information, they mitigate the risk of data exposure in case of a breach.
  - Cross-Domain Compatibility: JWTs can be easily transmitted over HTTP headers or URLs, making them suitable for use in cross-domain communication and enabling seamless integration with various frontend and backend technologies.

8. Screenshots

Fig-1: Home Page



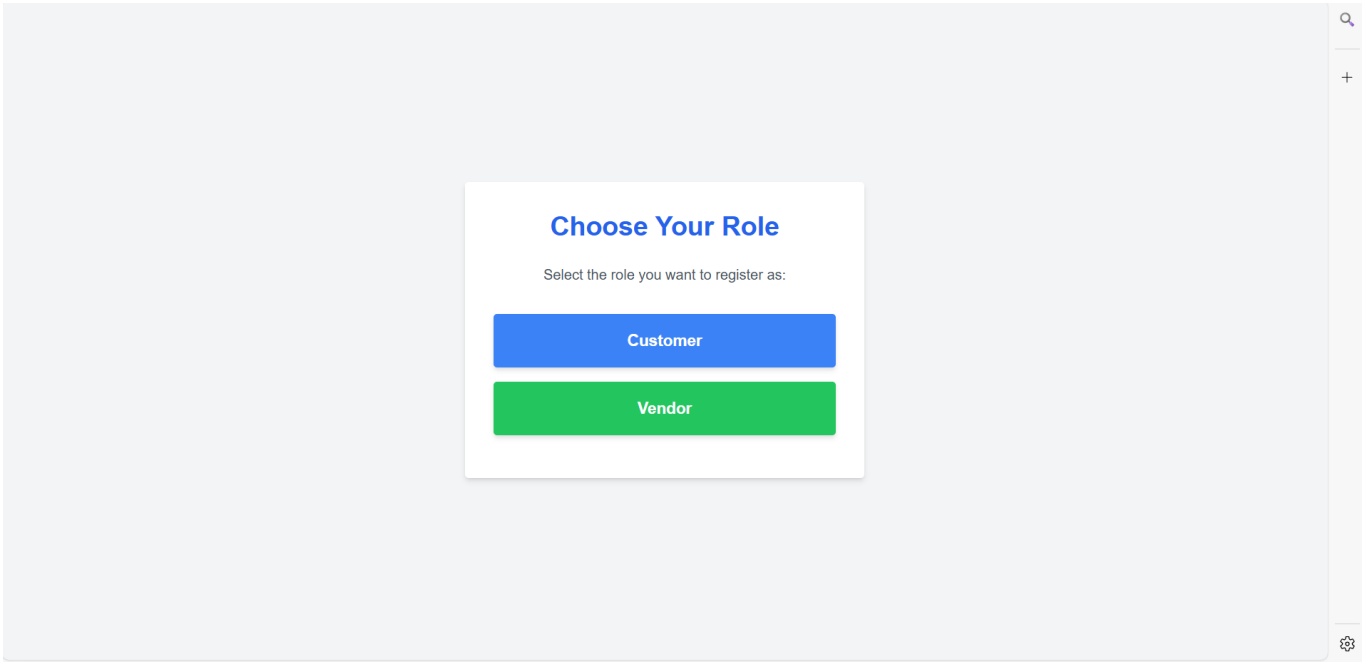


Fig2 – Role Selection page

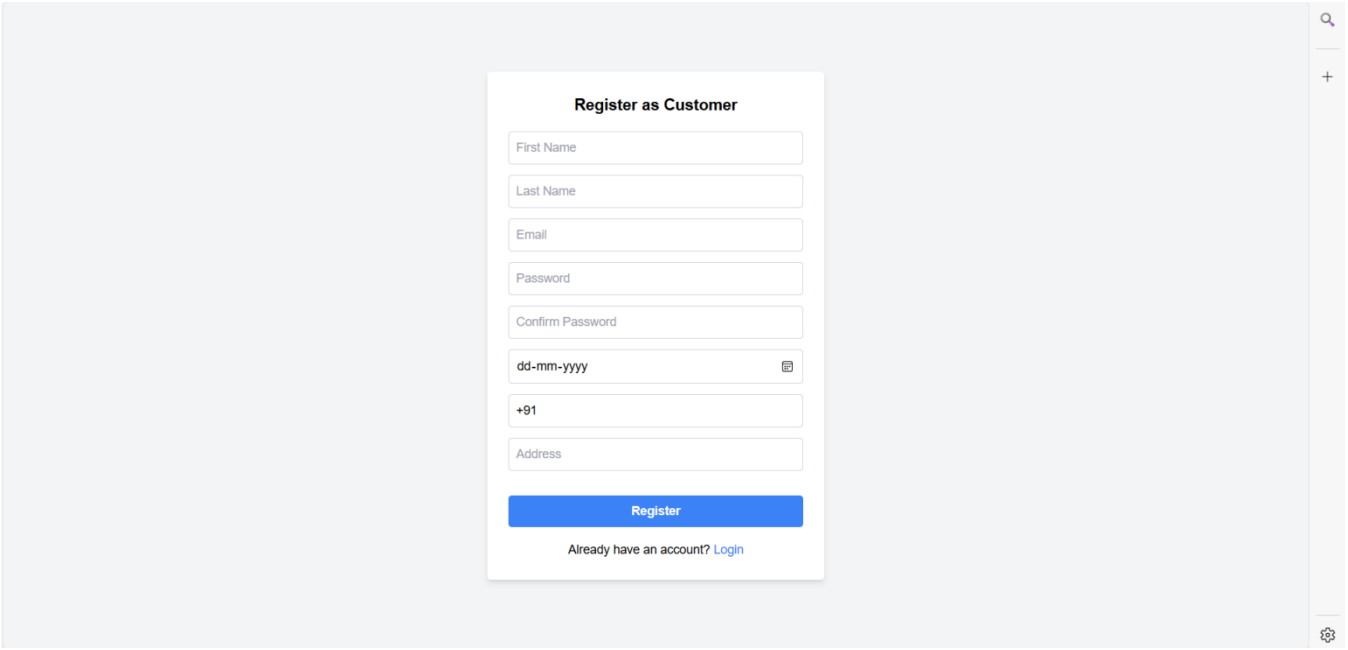


Fig3 – Customer Registration page

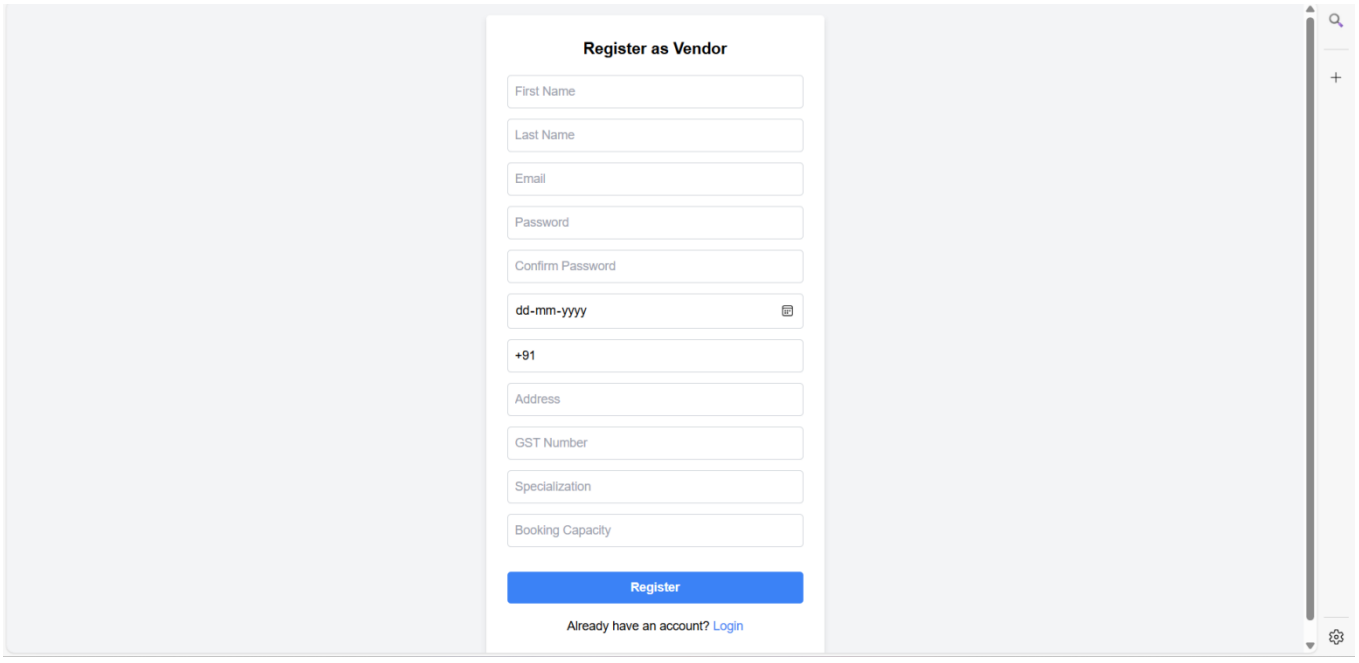


Fig 4 –Vendor Registration page

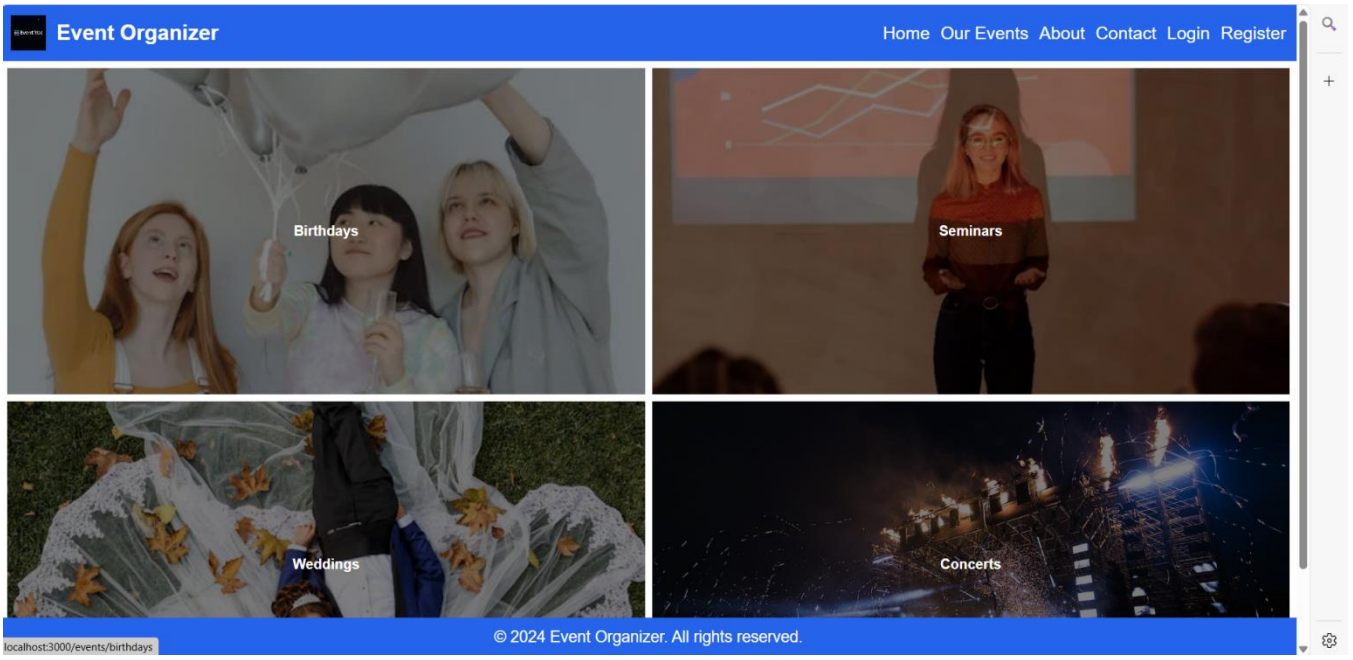


Fig 5 –Our Events page

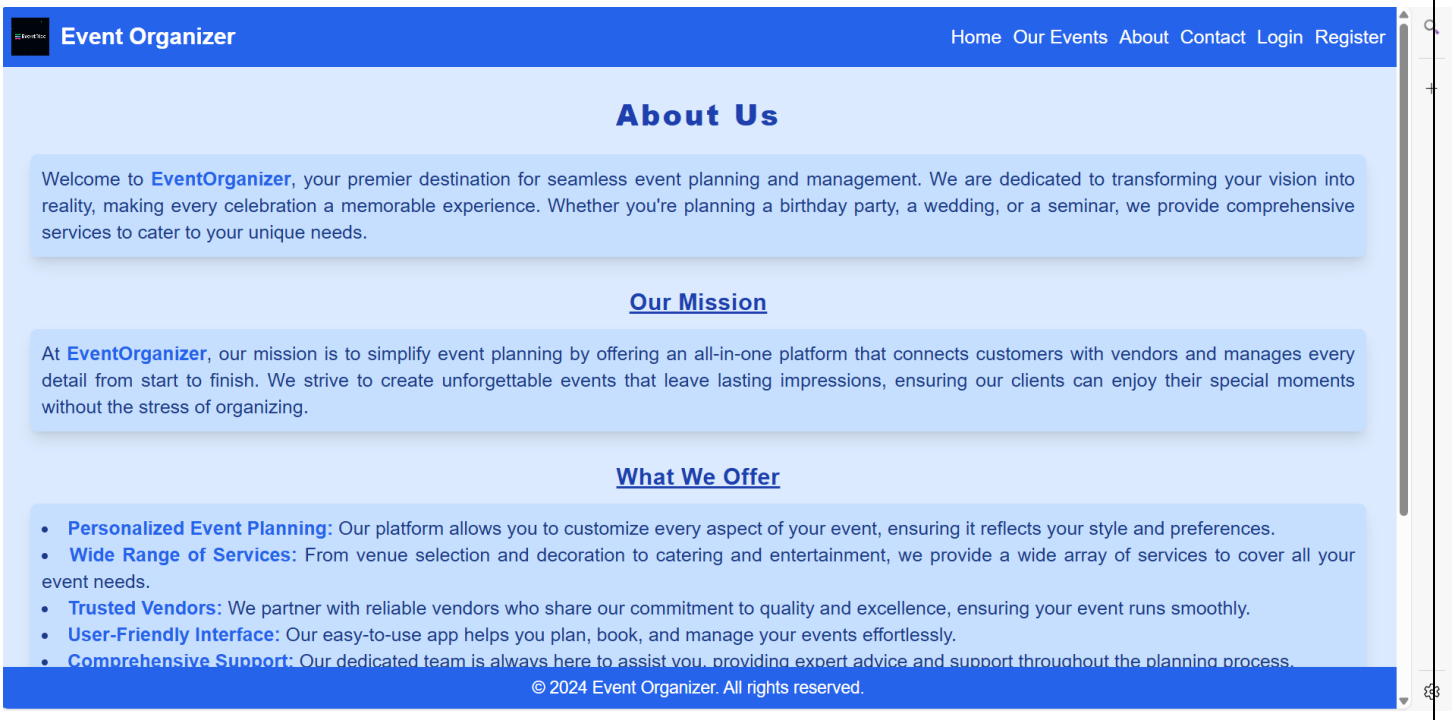


Fig 6 –About Us Page

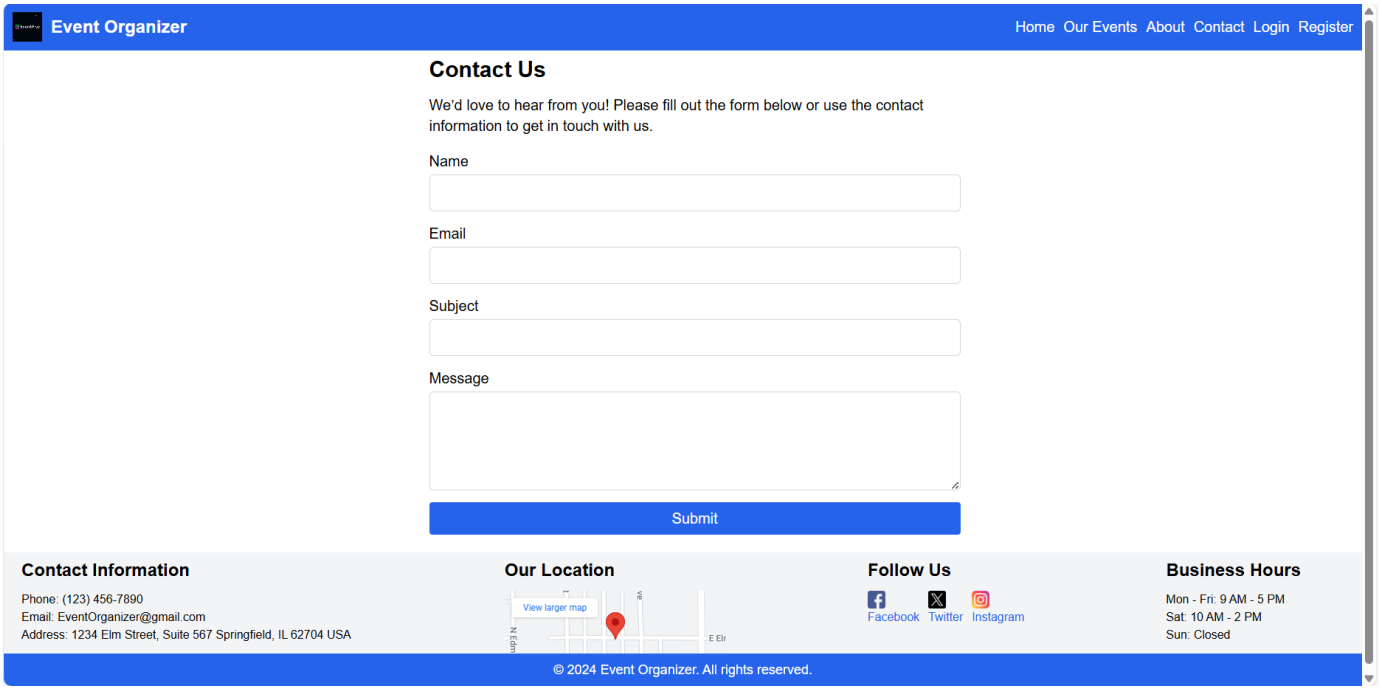


Fig 7 –Contact Us Page

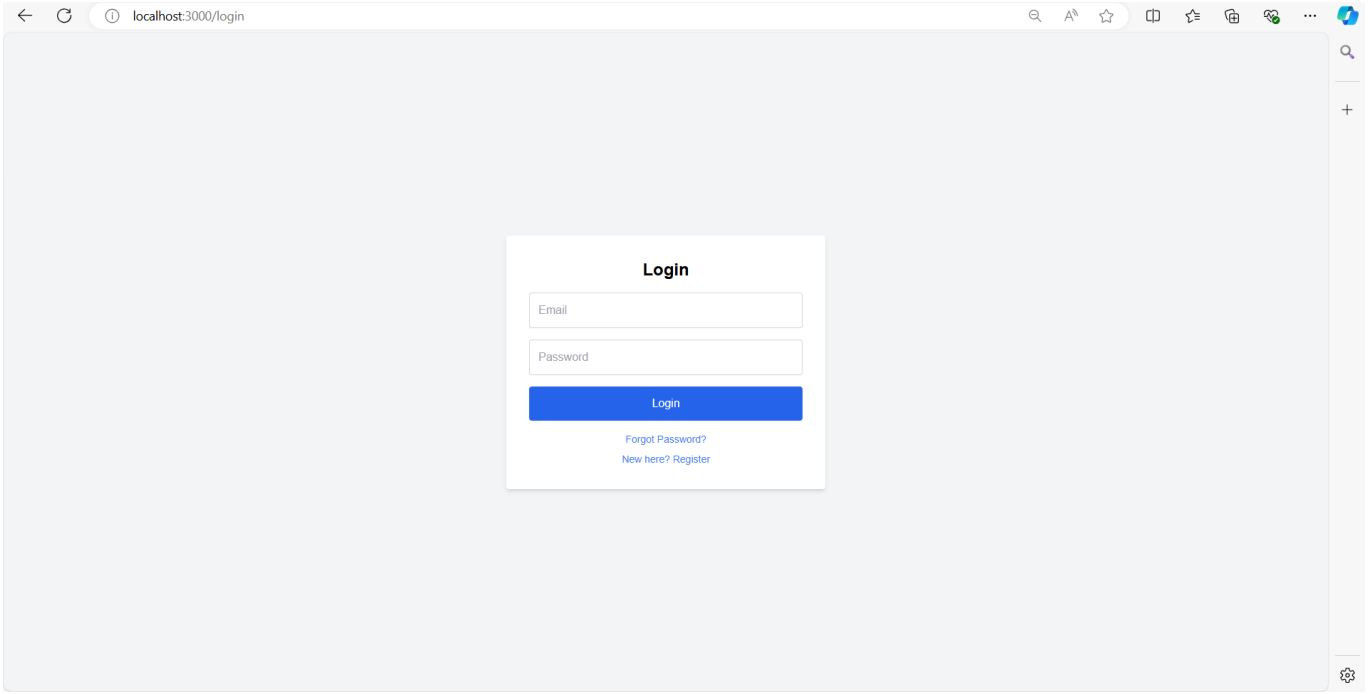


Fig 8 –Login Page

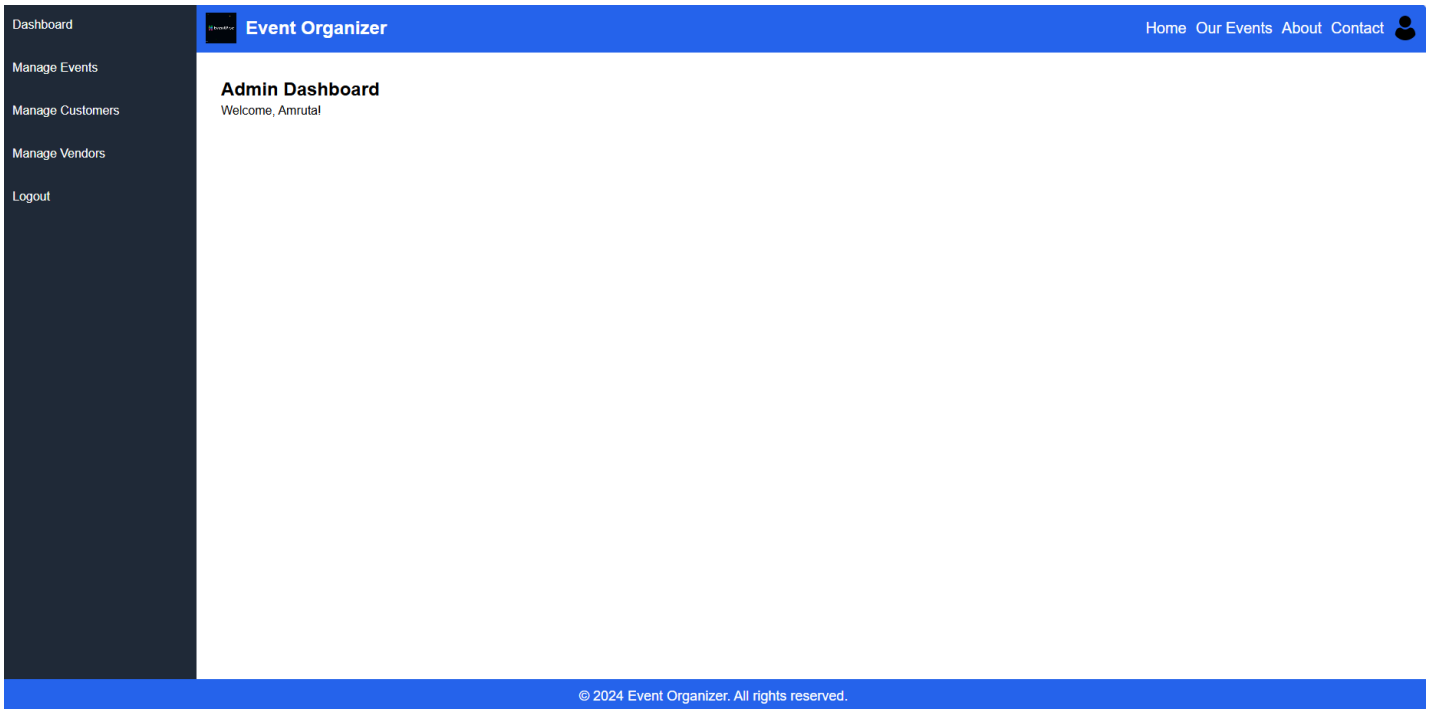


Fig 9 –AdminDashboard Page

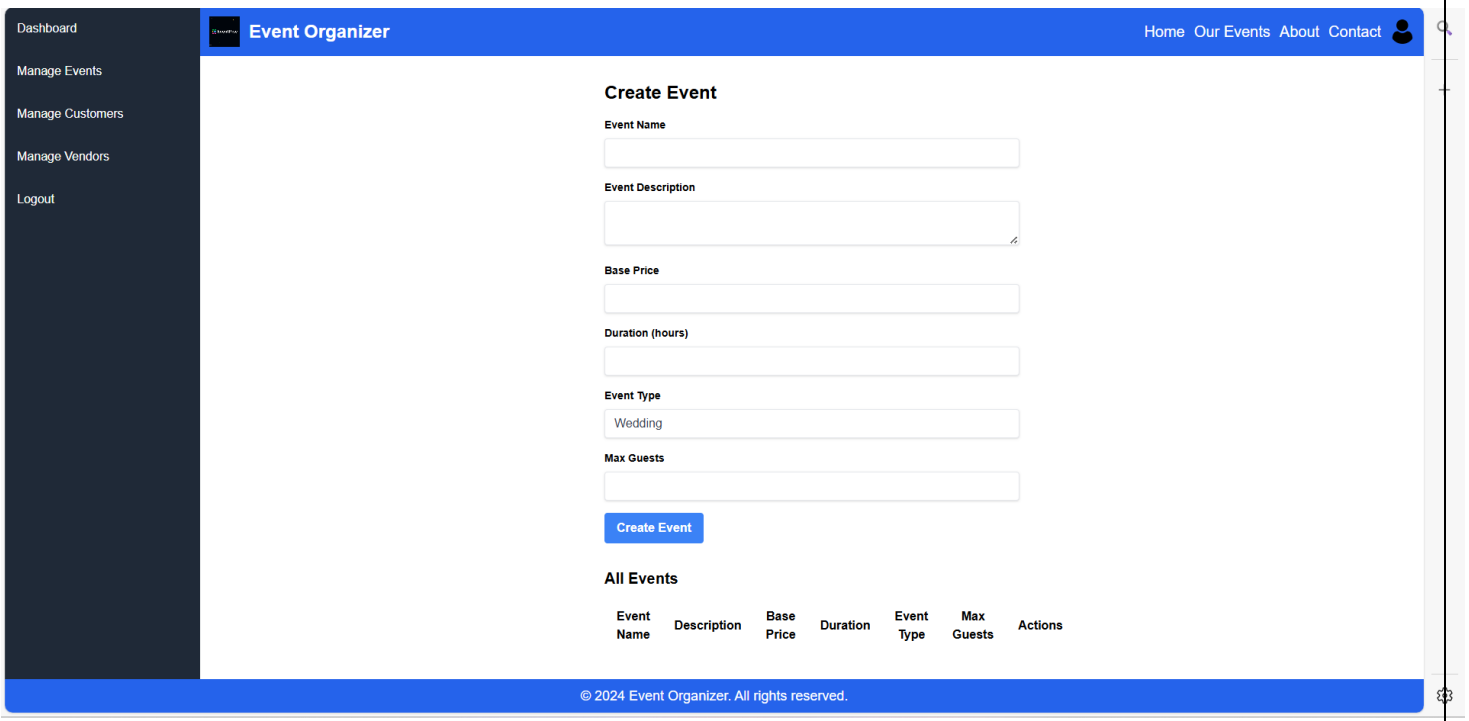


Fig 10 –Manage Events Page

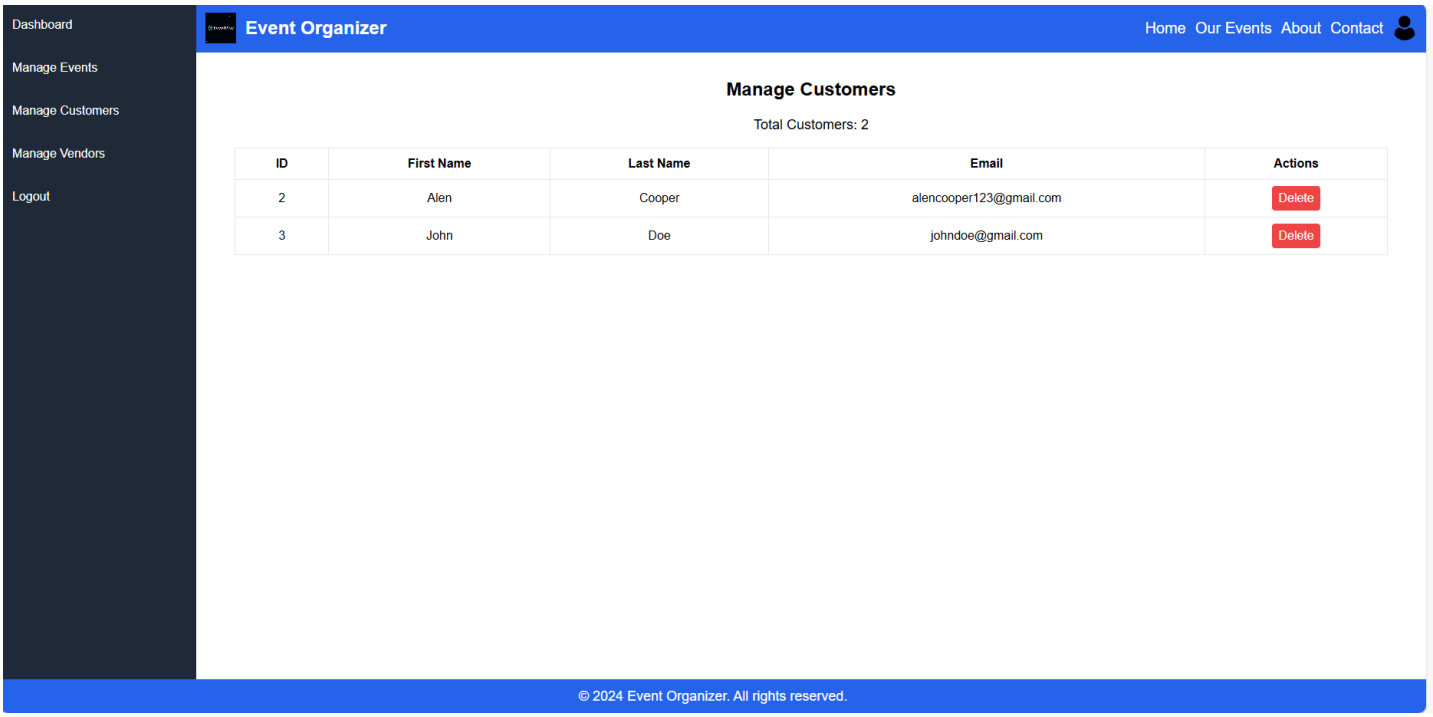


Fig 11 –Manage Customers Page

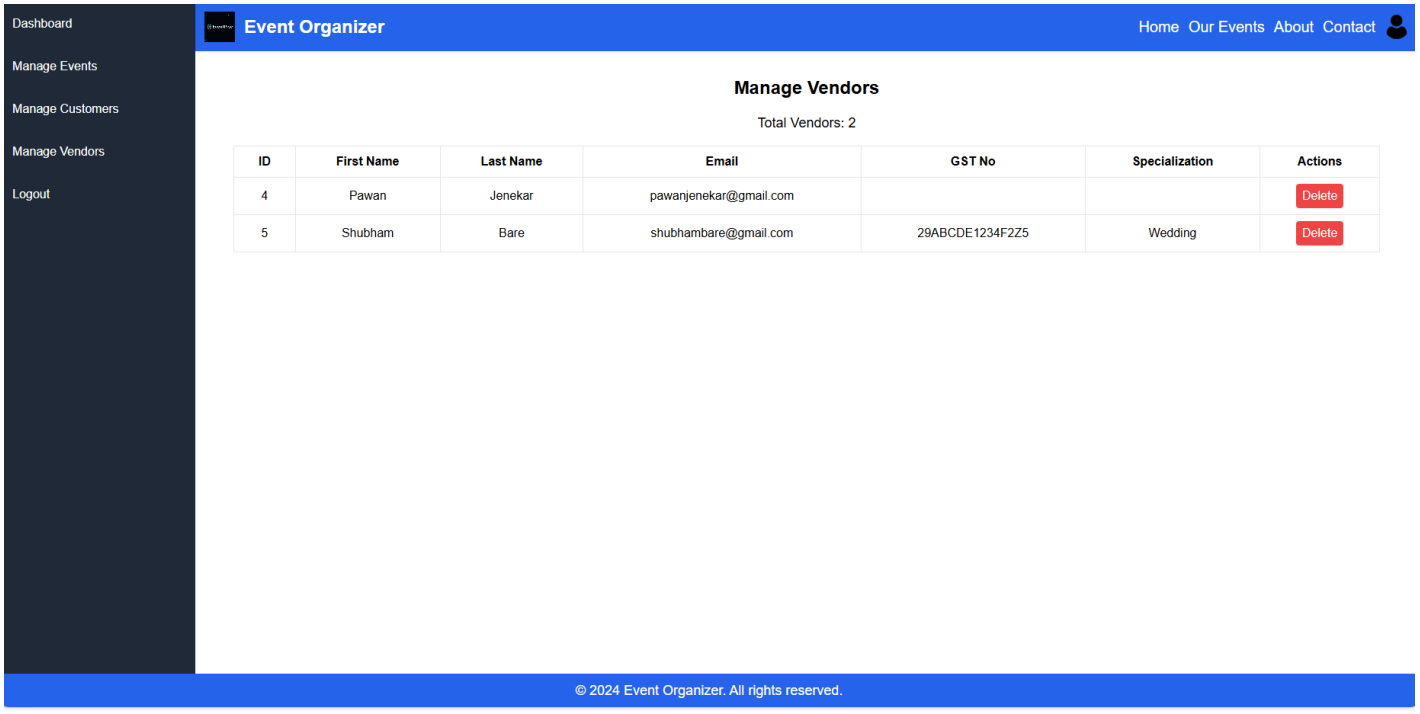


Fig 12 –Manage Vendors Page

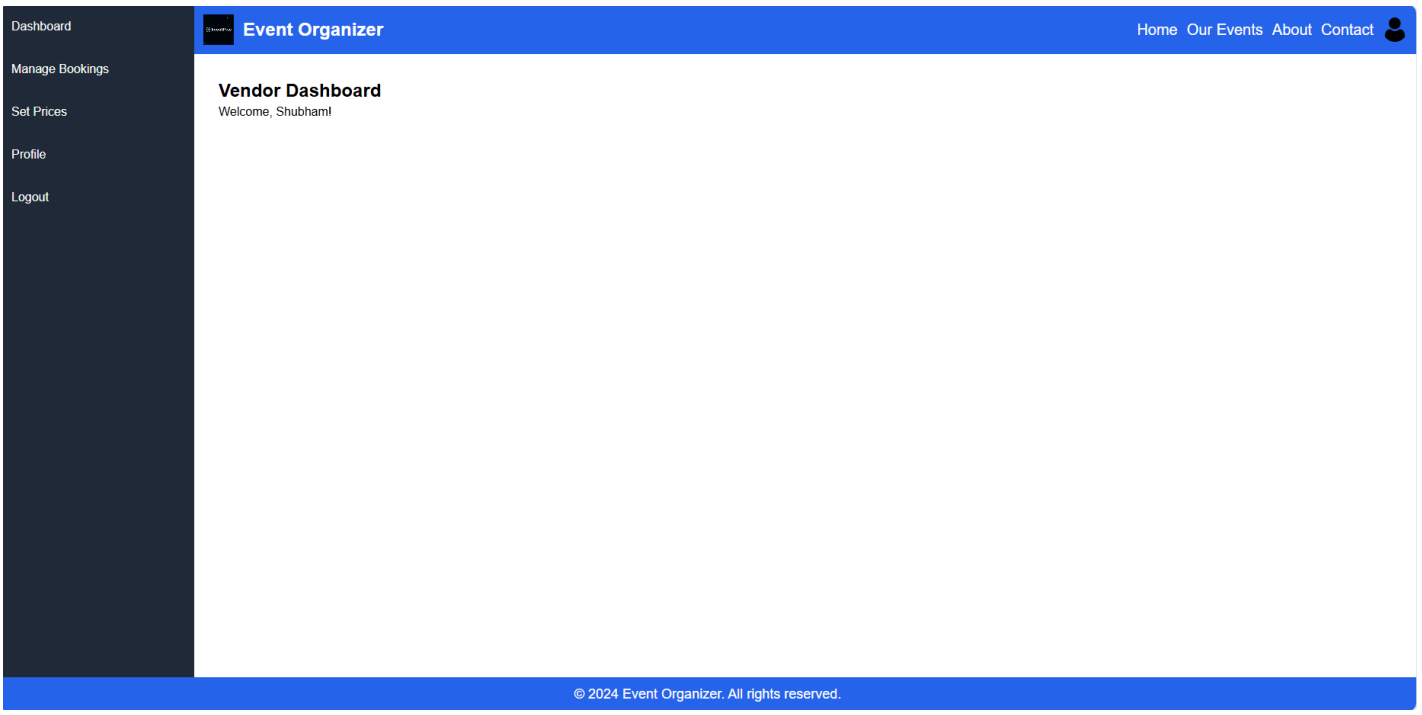


Fig 13 –Vendors Dashboard Page

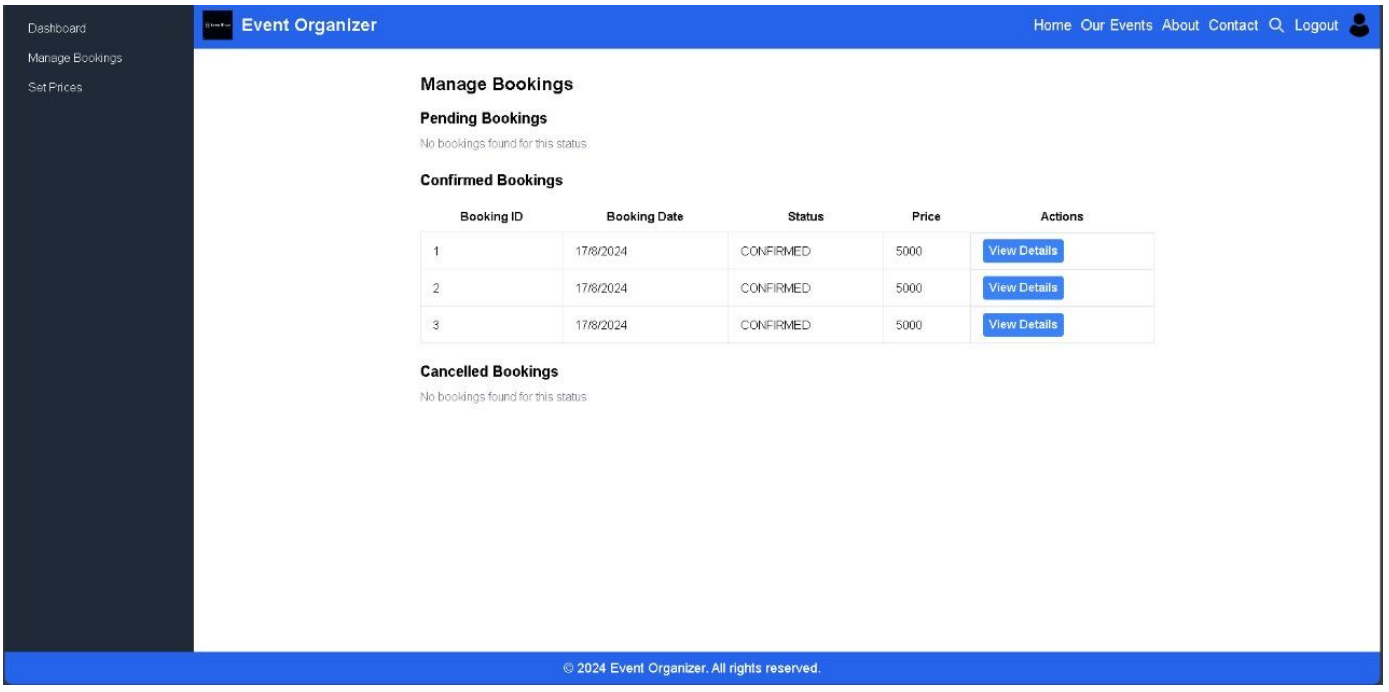


Fig 14 –Manage Bookings Page

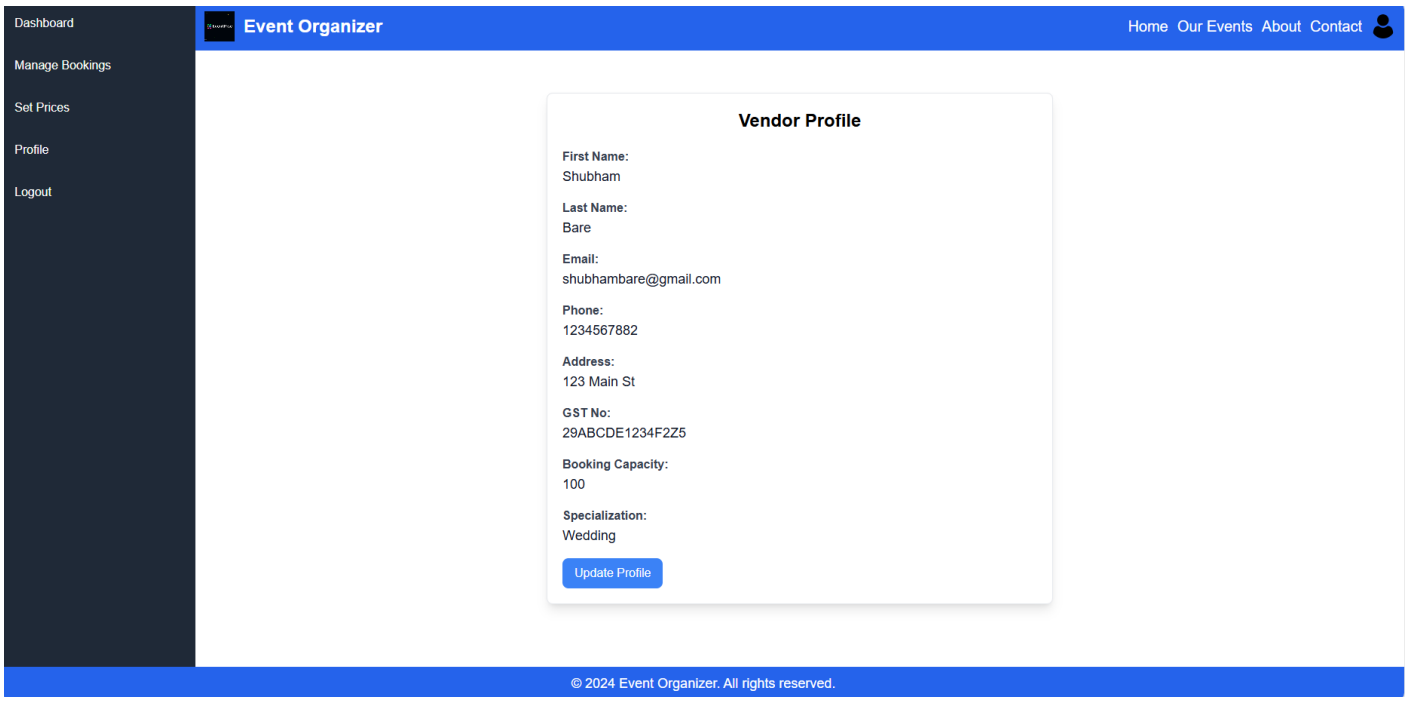


Fig 15 – Vendor Profile Page

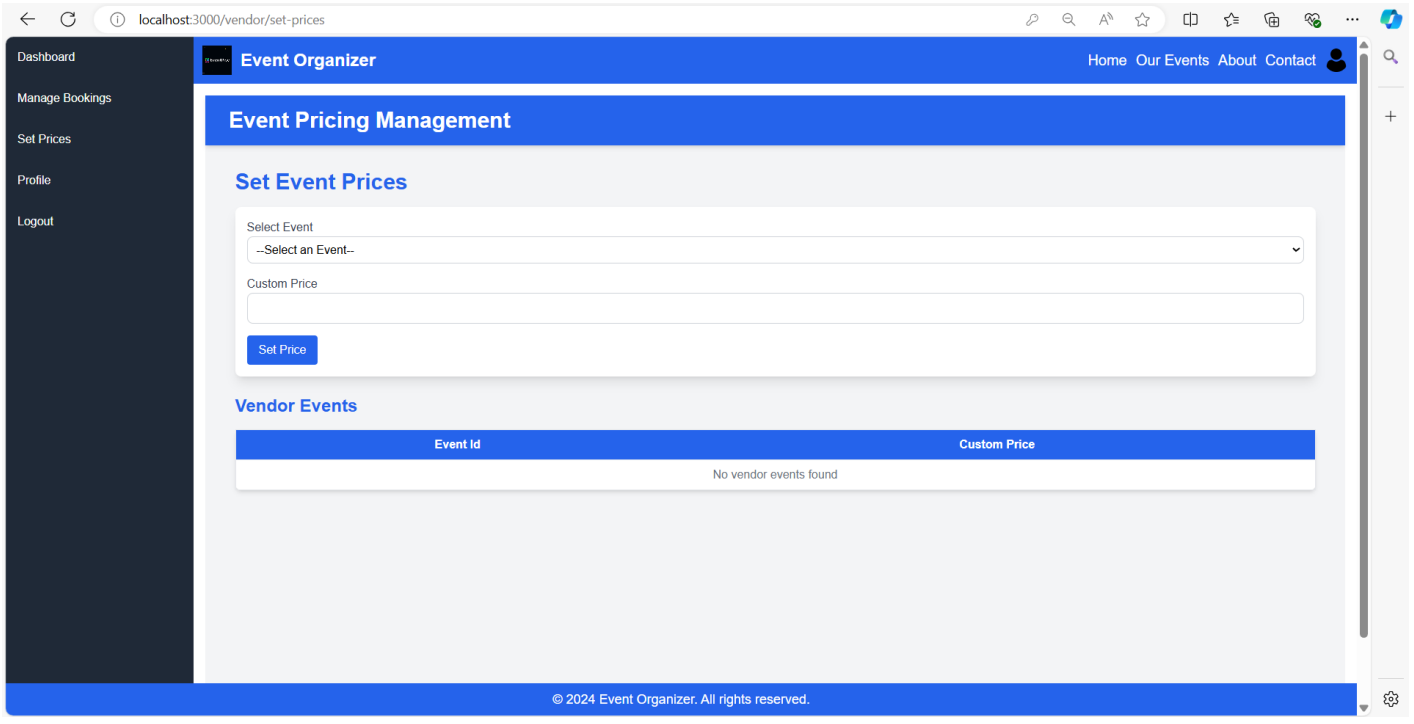


Fig 16 – Set Event Prices Page

9. Conclusion

In conclusion, the "Event Organizing Web Application" project successfully demonstrates the integration of modern web technologies to create a robust, user-friendly platform that streamlines event management for admins, customers, and vendors. By leveraging ReactJS for a dynamic front end, coupled with Redux for efficient state management and Axios for seamless API communication, the application provides a responsive and intuitive user experience. The Spring Boot-powered backend ensures a secure and scalable environment, with JWT-based authentication and authorization safeguarding user data and interactions. The use of MySQL as the database solution underpins the application's reliability and performance, while rigorous testing with JUnit and Postman reinforces the system's robustness. This project exemplifies a full-stack development approach, effectively balancing security, performance, and usability to deliver a comprehensive solution for event management.

10. References

1. https://spring.io/projects/spring-boot
2. https://spring.io/projects/spring-data-jpa
3. https://restfulapi.net/
4. https://www.mysql.com/
5. https://spring.io/projects/spring-web
6. https://reactjs.org/
7. https://nodejs.org/