**PYTHON/DJANGO BASED EVENT MANAGEMENT SYSTEM.**

**ERIC NDIVO MUOKI(BSCS)**

**ICT-G-4-1519-21**

**A COMPUTER SYTEM PROJECT SUBMITTED TO THE SCHOOL OF COMPUTING AND INFORMATICS IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE AWARD OF THE DEGREE OF BACHELOR OF SCIENCE IN COMPUTER SCIENCE OF GRETSA UNIVERSITY.**

**APRIL 2024**

This Proposal is my original work and has not been presented for award of a degree or for any similar purpose in any other institution

Signature:......................................  Date:...........................................

Eric Ndivo

ICT-G-4-1519-21

Supervisor: This proposal has been submitted with my approval as University supervisor

Signature:....................................... Date:...........................................

Denis Wapukha

Department: School of computing and informatics

Gretsa University()

# 

# Table of Contents

Table of Contents

[Table of Contents 3](#_Toc163044470)

[DEDICATION 4](#_Toc163044471)

[ACKNOWLEDGEMENT 4](#_Toc163044472)

[ABBREVIATIONS AND ACRONYMS 4](#_Toc163044473)

[ABSTRACT 4](#_Toc163044474)

[CHAPTER ONE:INTRODUCTION 5](#_Toc163044475)

[1.1 Background to the Study 5](#_Toc163044476)

[1.2 Problem Statement 5](#_Toc163044477)

[1.3 General Objective 5](#_Toc163044478)

[1.4 Specific Objectives 5](#_Toc163044479)

[1.5 Significance of the research 6](#_Toc163044480)

[1.6 Scope of the Study 6](#_Toc163044481)

[CHAPTER TWO: LITERATURE REVIEW 6](#_Toc163044482)

[CHAPTER 3: Methodology 8](#_Toc163044483)

[3.1 System Design 9](#_Toc163044484)

[3.1.1 Context Diagram 9](#_Toc163044485)

[3.1.2 Data Flow Diagram 10](#_Toc163044486)

[3.2: Database Design 11](#_Toc163044487)

[3.3: Input and Output Design 11](#_Toc163044488)

[3.4: CODING 12](#_Toc163044489)

[3.5 TESTING 12](#_Toc163044490)

[CHAPTER 4: OUTPUTS AND IMPACTS 13](#_Toc163044491)

[4.1 Results 13](#_Toc163044492)

[4. 2 Modules Output 14](#_Toc163044493)

[4.2.1 Credit Card payment 14](#_Toc163044494)

[4.3: Measurement of Software quality and reliability 14](#_Toc163044495)

# DEDICATION

This report is dedicated to God whose strength has enabled me to complete this work and the relevant personnel that has led me all through.

# ACKNOWLEDGEMENT

I am very grateful to God for His favor, grace and protection. I extend heartfelt thanks to my parents and friends for their immeasurable support. I am also thank my supervisor, Mr. Denis wapukha for his cooperation throughout this project. May God bless and reward each of you abundantly.

# ABBREVIATIONS AND ACRONYMS

EMS - Event Management System

CRM - Customer Relationship Management

ERP - Enterprise Resource Planning

CMS - Content Management System

API - Application Programming Interface

UI - User Interface

UX - User Experience

RFID - Radio-Frequency Identification

NFC - Near Field Communication

QR Code - Quick Response Code

POS - Point of Sale

CMS - Content Management System

CMS - Customer Management System

ROI - Return on Investment

KPI - Key Performance Indicator

SLA - Service Level Agreement

API - Application Programming Interface

SMS - Short Message Service

CDN - Content Delivery Network

DNS - Domain Name System

HTTP - Hypertext Transfer Protocol

HTTPS - Hypertext Transfer Protocol Secure

# 

# ABSTRACT

The Event Management System (EMS) is a comprehensive software solution that streamlines the planning, management, and execution of events of all sizes. With capabilities such as event creation, ticketing, scheduling, and e-ticket generation, the EMS provides a consolidated platform for easy event management. From corporate conferences to social gatherings, the EMS enables organizers to easily arrange all aspects of their events, resulting in seamless operations and memorable experiences for guests. The EMS improves overall event success by optimizing productivity, reducing logistical problems, and providing user-friendly interfaces and capabilities

# CHAPTER ONE:INTRODUCTION

## Background to the Study

Event management systems are evolving into essential centers for the full event experience, not only for handling logistics. Beyond simple event scheduling, these systems are increasingly adding interactive event apps, live streaming integration, and tools for engaging attendees. Through mobile devices or virtual platforms, attendees may easily access content, customize their experiences, and engage with presenters and other attendees.

The integration of virtual and hybrid event capabilities has been accelerated by the global shift towards remote work and digital connectivity. Event management systems are adapting to facilitate these new formats, offering solutions for virtual attendee registration, live streaming, on-demand content access, and networking opportunities through virtual event spaces.

Moreover, data security is a critical focus area for event management systems, given the sensitivity of attendee information and financial transactions. Modern systems are implementing robust security measures, including encryption, access controls, and compliance with data protection regulations like GDPR and CCPA, to ensure the privacy and safety of event data.

Another trend shaping event management systems is sustainability. Organizers are increasingly conscious of reducing the environmental impact of events, and software platforms are integrating features to support eco-friendly practices such as digital ticketing and paperless communication.

Additionally, the role of AI in event management continues to expand. AI-powered analytics can provide insights into attendee preferences, behavior patterns, and engagement levels, enabling organizers to optimize future events and enhance attendee satisfaction.

## Problem Statement

Traditional methods of event planning and management are frequently disorganized, time-consuming, and prone to errors, resulting in inefficiencies and unsatisfactory results. Coordinators experience difficulties in arranging, advertising, and carrying out events smoothly, resulting in missed possibilities for participation and income production. To solve these issues and improve the entire event experience for both organizers and participants, a comprehensive event management solution that integrates all parts of event planning, from registration and ticketing to promotion and feedback engagement, is urgently required.

## General Objective

The general objective of the event management system is to provide a comprehensive and efficient platform that facilitates the planning, organization, and execution of various events.

## Specific Objectives

The suggested event management system strives to achieve several essential goals. For starters, it aims to streamline event ticketing chores such as user registration and events display. Furthermore, the system intends to improve e-ticketing to streamline ticket distribution. The technology also intends to improve registration processes to enable a smooth check-in and data management. Scalability and flexibility are critical, with the system built to accept a wide range of events for display. Overall, the system aims to transform event management, resulting in more successful and meaningful experiences for organizers, participants, and stakeholders alike.

## Significance of the research

The significance of researching event management systems lies in its potential to transform the way events are planned, organized, and executed. By delving into this area, we can uncover insights into the effectiveness of current systems, identify areas for improvement, and develop innovative solutions to enhance the overall event experience. Such research can lead to the development of more efficient and user-friendly platforms that streamline processes, reduce costs, and increase attendee satisfaction. Additionally, understanding the significance of event management systems research can contribute to the advancement of the event industry as a whole, driving innovation and shaping the future of event planning and execution.

# CHAPTER TWO: LITERATURE REVIEW

Event management systems have emerged as essential tools for modern event planning, offering centralized platforms to streamline processes and enhance overall efficiency (Büyüközkan & Göçer, 2018). Traditional methods of event planning often involve manual tasks and disjointed tools, leading to inefficiencies and challenges for organizers (Ratten, 2011). In response, event management software has gained traction across various industries due to its ability to automate tasks and provide real-time visibility into event logistics (Aldehayyat, 2020).

According to Büyüközkan and Göçer (2018), event management systems play a crucial role in optimizing resource allocation and improving communication among event stakeholders. These systems integrate features such as user registration, venue management, scheduling, ticketing, and promotion into a cohesive platform, enabling organizers to manage events more effectively (Aldehayyat, 2020). The demand for such systems continues to grow as organizations recognize the importance of delivering seamless experiences to attendees while maximizing event ROI (Lee & Gretzel, 2012).

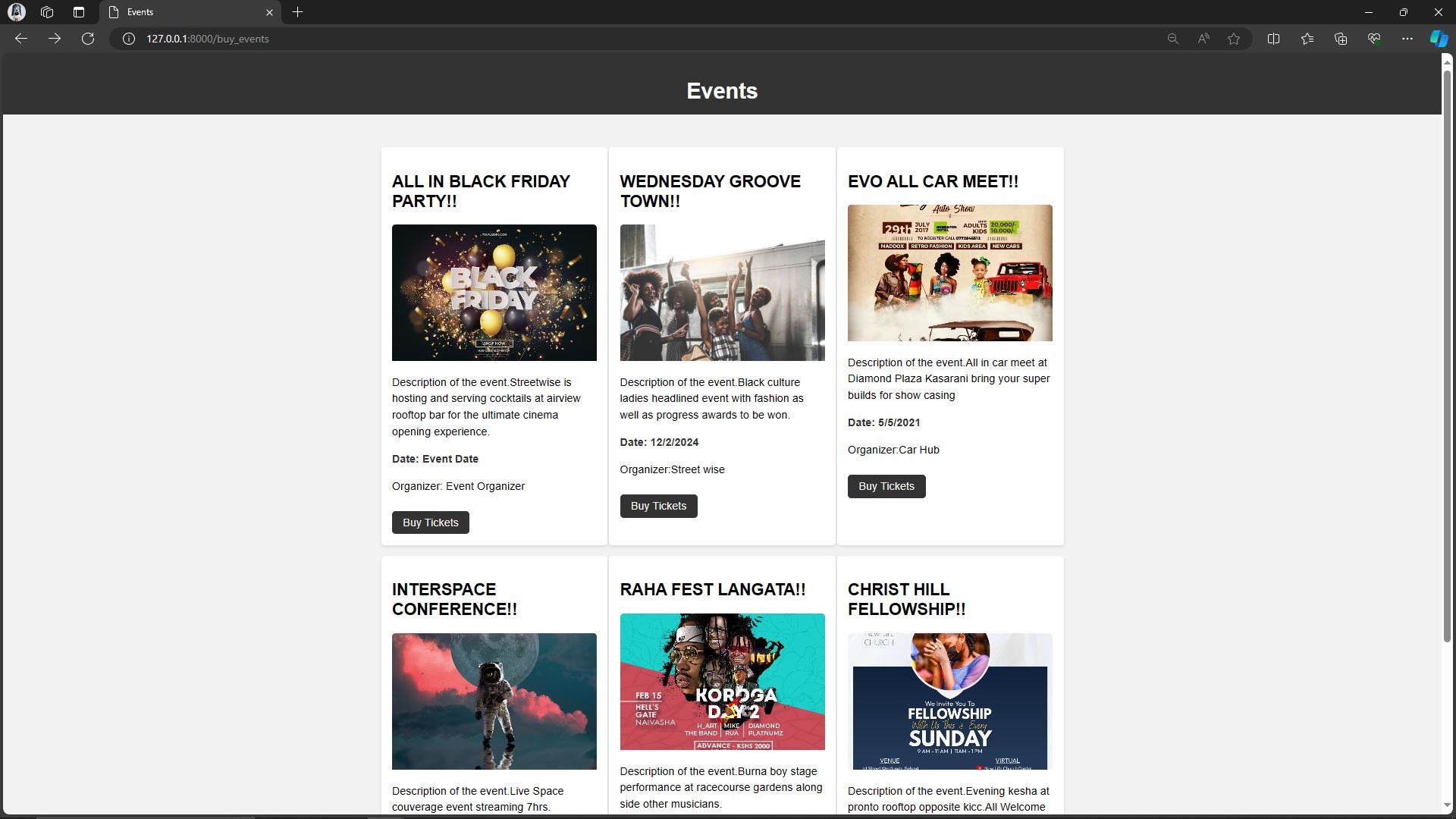
One of the key benefits of event management software is its ability to facilitate attendee engagement and interaction (Chang & Chi, 2010). Interactive event apps and personalized communications enhance the attendee experience, leading to higher levels of satisfaction and increased event success (Sarstedt et al., 2013). Moreover, AI-driven analytics provide organizers with valuable insights into attendee behavior and preferences, enabling them to tailor future events for maximum impact (Buhalis & Foerste, 2015).

Cloud-based architectures have also transformed event management systems by offering scalability, flexibility, and enhanced security (Hassan et al., 2018). This shift towards cloud computing ensures data protection and accessibility from anywhere, empowering organizers to manage events remotely and collaborate efficiently with teams and stakeholders (Aldehayyat, 2020).

Despite the advancements in event management technology, challenges such as user adoption, data security, and adapting to emerging trends like virtual and hybrid events persist (Lee & Gretzel, 2012). However, innovative solutions, including the integration of APIs and AI, are reshaping event management systems to meet the evolving needs of organizers and attendees (Buhalis & Foerste, 2015).

In summary, event management systems have revolutionized the events industry by offering comprehensive solutions to streamline planning processes, enhance attendee engagement, and optimize event outcomes. By leveraging advanced technologies and addressing key challenges, these systems continue to evolve and play a critical role in delivering exceptional event experiences.

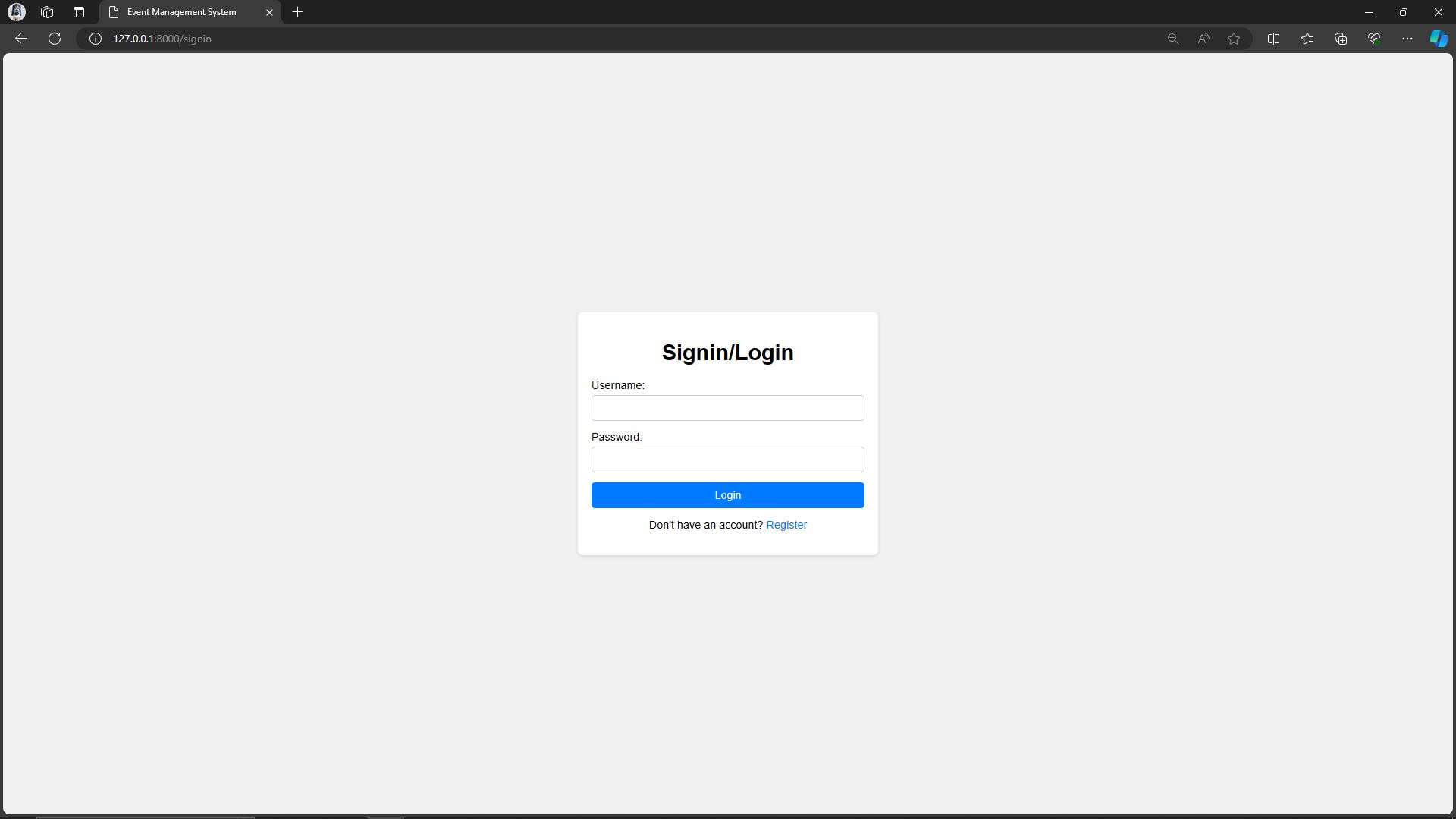
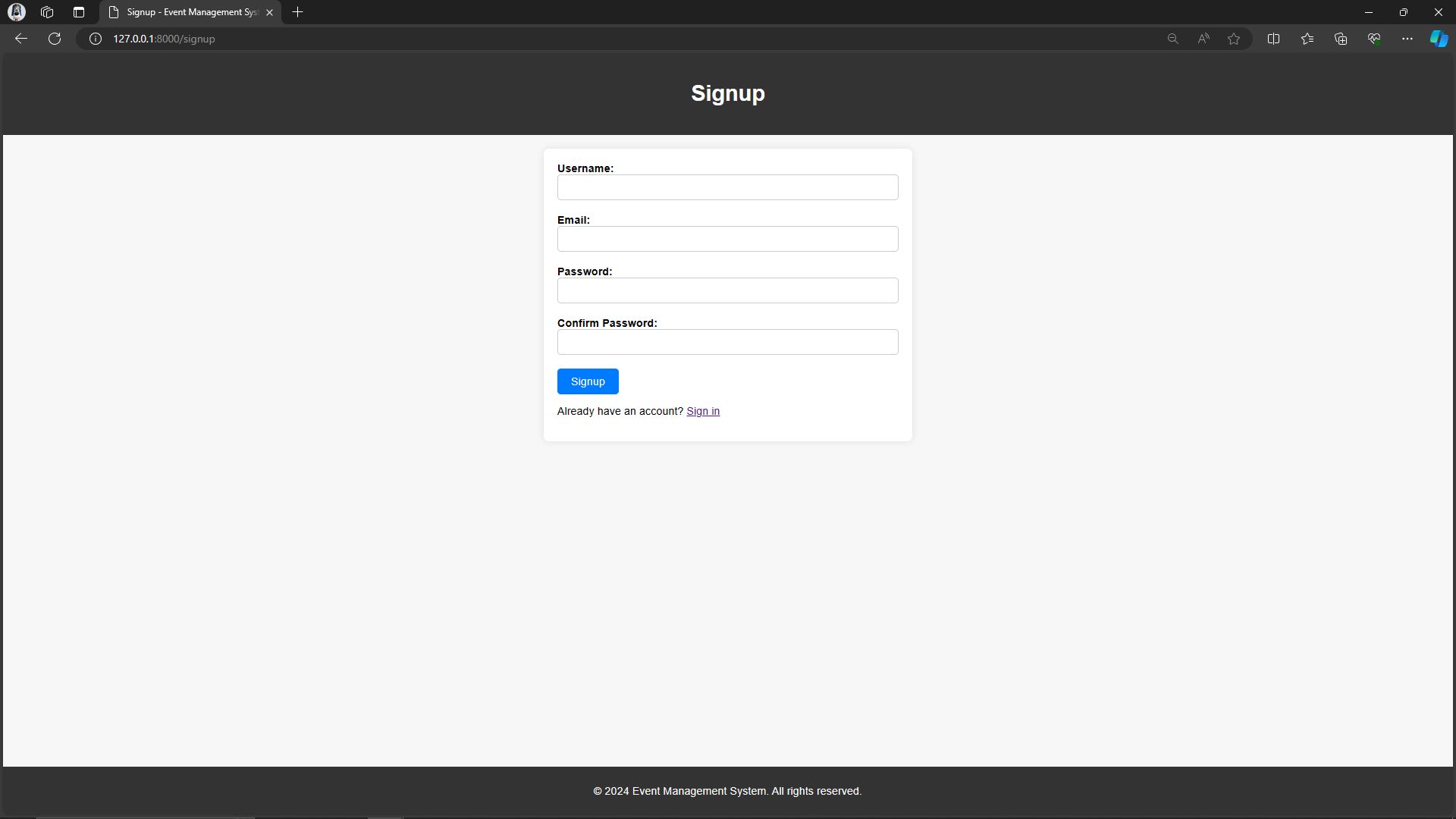
## Features



*Intuitive Interfaces*

* **Signup/Signin**

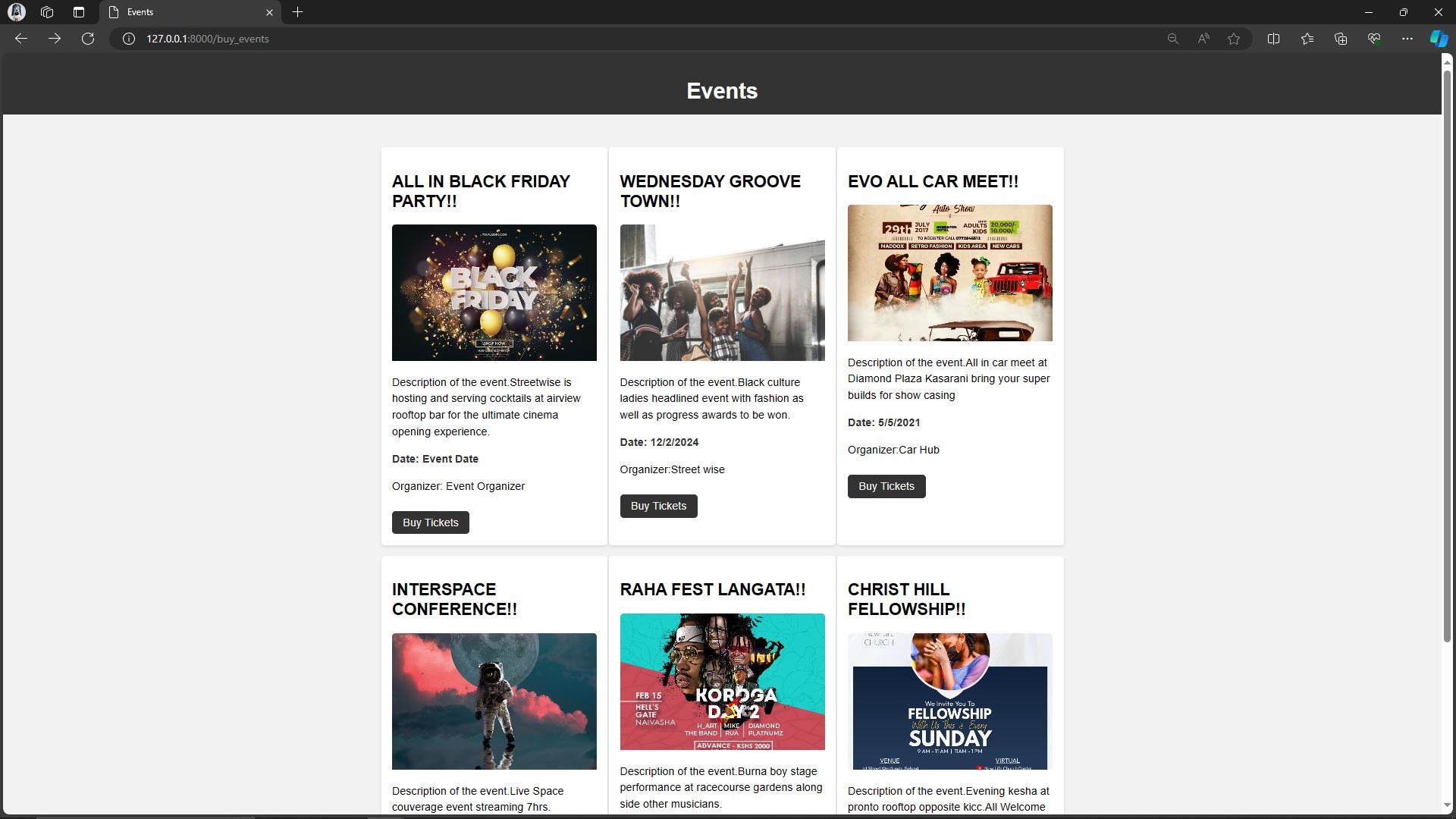
The signup/signin function allows users to create accounts or log in to access personalized features and manage their event-related activities efficiently.



*Signup Signin*

* **Advanced Events listing**

Advanced Events listing incorporates innovative features for comprehensive event management, enhancing user experience and efficiency.



*Listed Events*

* **E-ticketing**

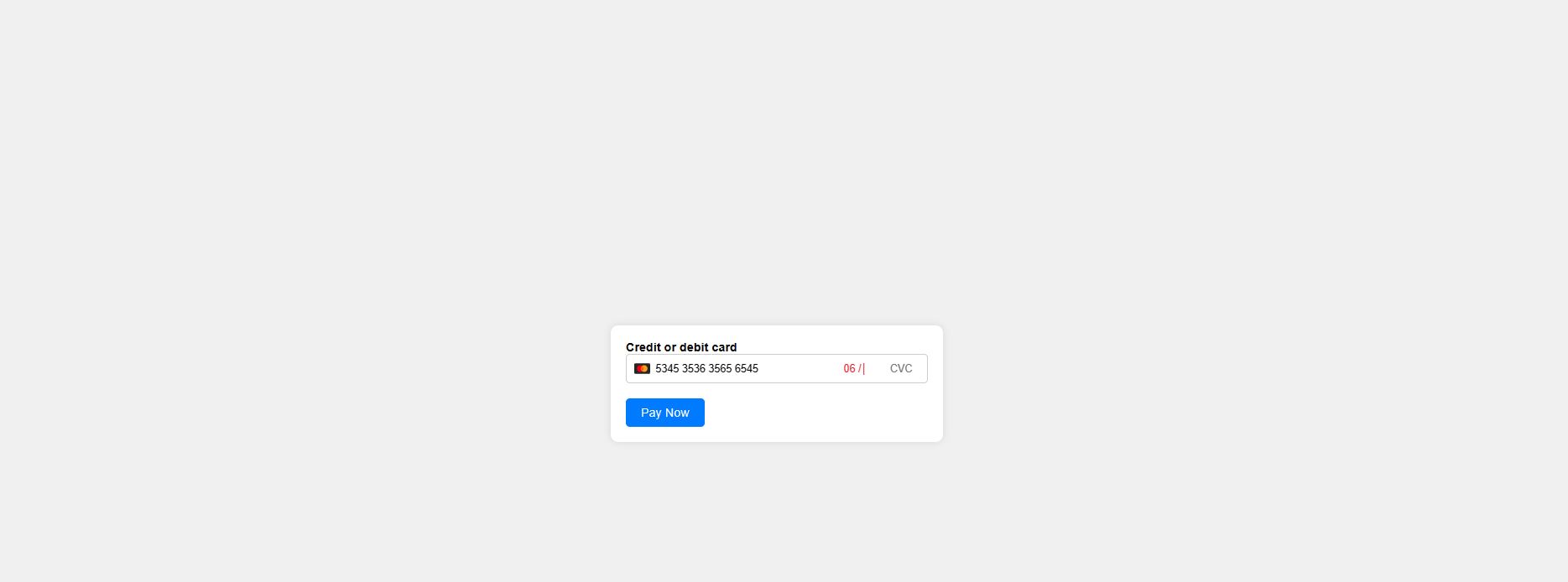
E-ticketing revolutionizes event access by providing convenient and paperless ticketing solutions accessible from electronic devices.



*E-ticket*

* **Online Transaction**

The Online Transaction feature enables secure and convenient payment processing for event tickets and services.



*Card payment*

# CHAPTER 3: METHODOLOGY

## 3.1 System Design

### 3.1.1 Context Diagram

SYSTEM ADMIN

Entering Event Logs View Event Logs

Request system event report

Event Requests

EVENT MANAGER

Event info Request Event Information

CIENT

### 3.1.2 Data Flow Diagram

**SYSTEM**

**ADMINISTRATOR**

**USER CREATION**

END USERS

**SYSTEM LOGIN**

**EVENTS ADVERTISEMENT**

**E-TICKET**

Viewuserdetails Authenticate user User Informatio User info

Accept user View user details User inf Updates/View

Login authentication

Accept/Deny

Login Information Login request

System redirect

Authentication POST request

Ticket Generation

Ticket log/transaction log Feedback/ticket to enduser

## 3.2: Database Design

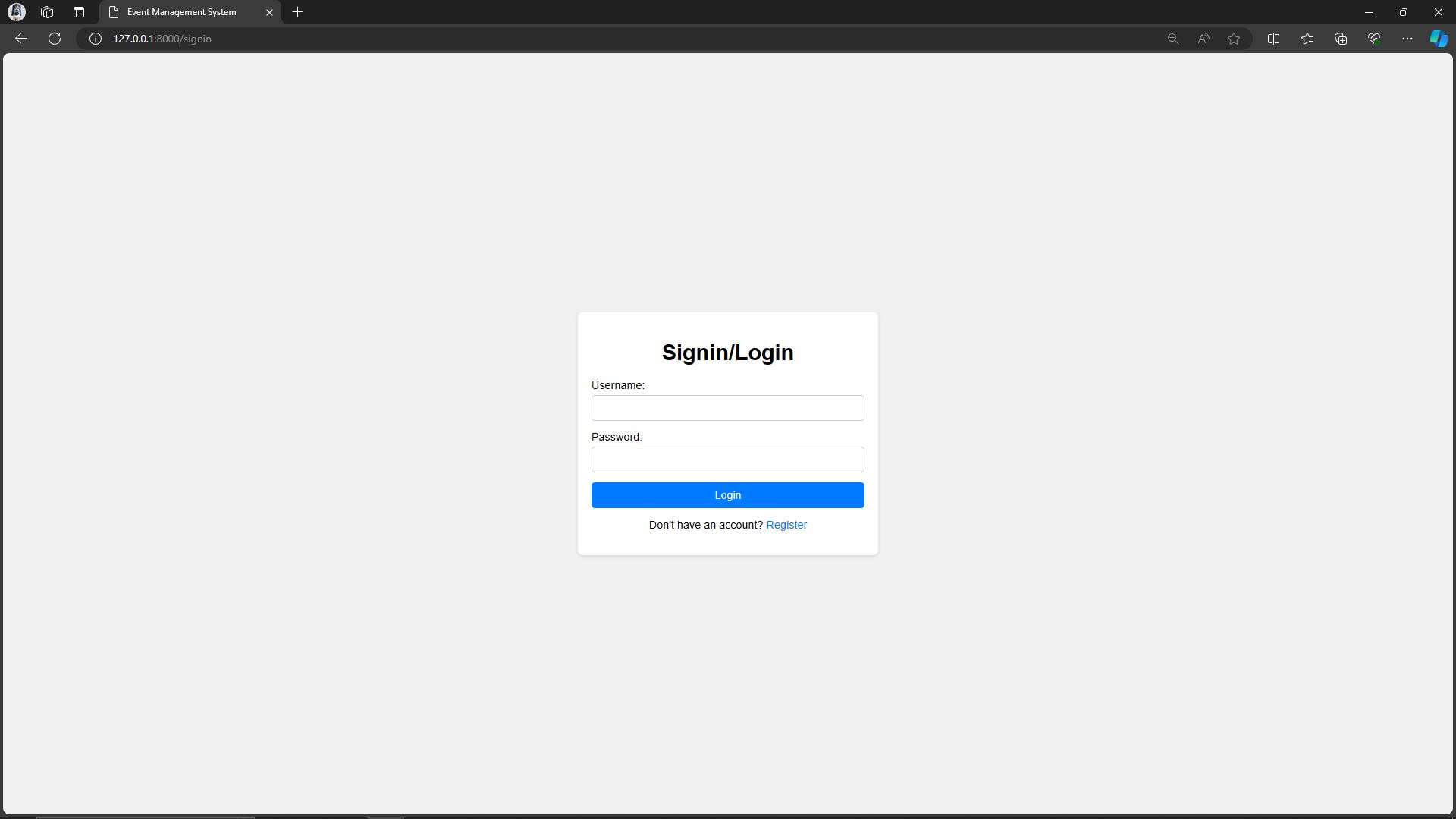
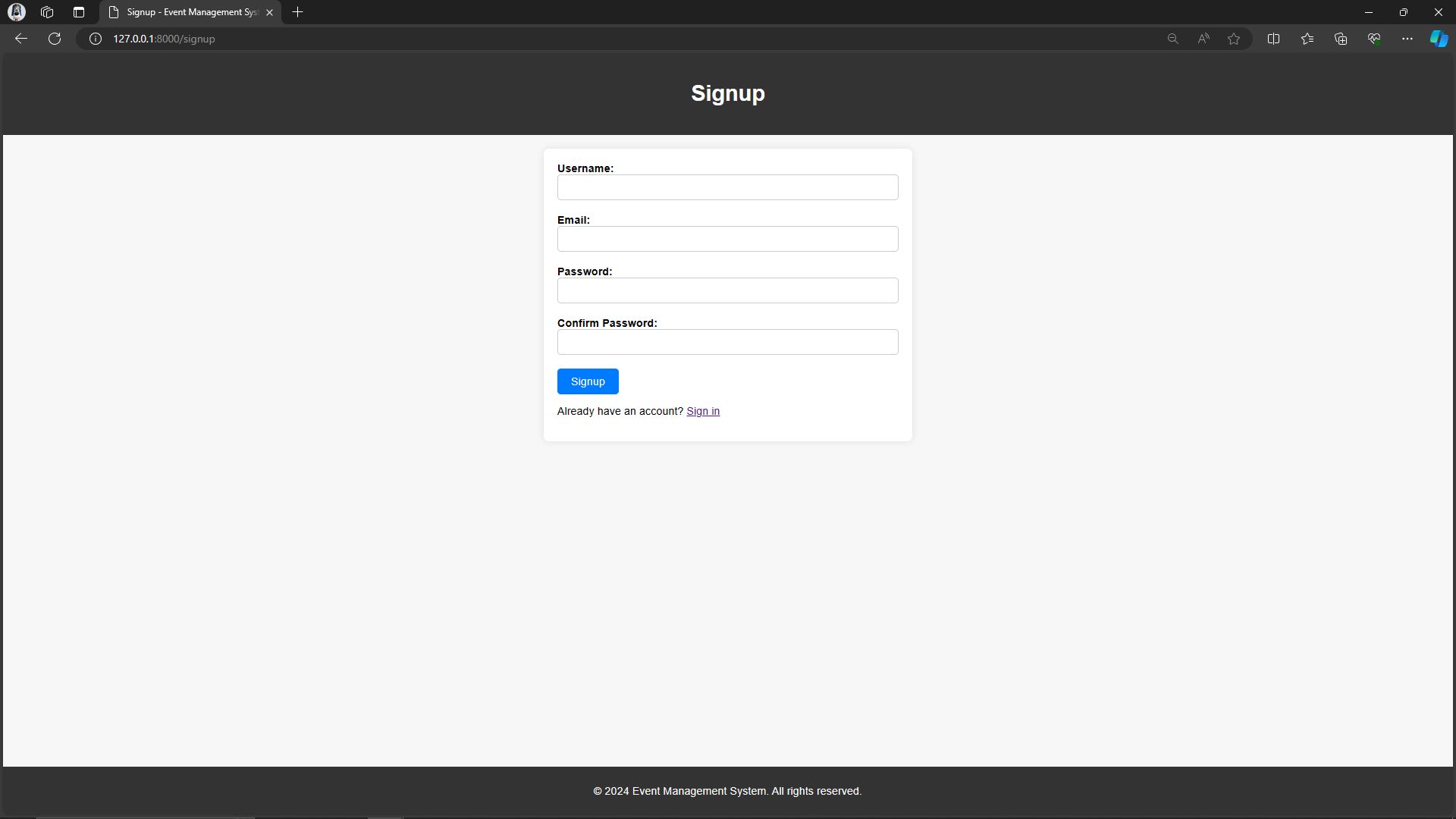
SQLite3 has been the preferred database solution for the project due to its lightweight nature and ease of integration. It's a small powerful ideal for managing the application's data requirements without the need for a separate server. One of its advantages is SQLite3 is simple to set up and configure, and because it stores everything in a single file, it is extremely portable. Furthermore, with its transactional support, one can be confident that data will remain consistent and secure, even in the face of unanticipated faults or crashes. Overall, SQLite3 strikes the ideal balance between functionality and ease, making it an essential component of the project development.

## 3.3: Input and Output Design

we've implemented the signup and signing forms using Django forms, which offer a streamlined approach to handling user input and authentication. With Django forms, we can define the fields we need for user registration and login, along with any validation rules, directly in Python classes. This abstraction allows us to focus on the logic of the application.

For the signup form, we utilize Django's built-in UserCreationForm, which provides fields for username, email, and password, along with validation to ensure data integrity. Similarly, the signin form leverages Django's Authentication Form, simplifying the process of verifying user credentials against stored user accounts.

By utilizing Django forms, we ensure consistency in data handling and validation across our application, reducing the risk of errors and enhancing the user experience. Additionally, Django's form processing capabilities integrate seamlessly with its authentication system, enabling secure user authentication and session management. Overall, Django forms facilitate efficient input and output design, allowing us to create a user-friendly signup and signin functionality with minimal effort.



*Signup Signin*

## 3.4: CODING

For the project, Python has been chosen as the primary language for backend development, particularly utilizing the Django framework, and HTML, CSS, and JavaScript for frontend development. Python is an excellent choice for backend development due to its simplicity, readability, and vast system of libraries and frameworks. Django, built on top of Python, provides a comprehensive toolkit for building web applications, offering features such as ORM (Object-Relational Mapping), URL routing, form handling, and authentication out of the box.

The reasons why Python and Django are the best choices for our project include:

1. Ease of Use: Python's clean syntax and readability make it easy for developers to write and maintain code, reducing development time and effort. Django's high-level abstractions further simplify common web development tasks, allowing developers to focus on business logic.

2. Scalability: Python and Django are highly scalable, capable of handling increased traffic and data as our application grows. Django's built-in scalability features, such as support for database sharding and caching, ensure that our application can handle large user loads efficiently.

3. Security: Django provides robust security features by default, including protection against common web vulnerabilities such as SQL injection, cross-site scripting (XSS), and cross-site request forgery (CSRF).

4. Rich Ecosystem: Python has a vast ecosystem of libraries and frameworks that extend its functionality and simplify various development tasks. Django itself provides numerous built-in modules and third-party packages for tasks such as database integration, user authentication, and frontend development.

On the frontend side, HTML, CSS, and JavaScript are the standard languages for building user interfaces in web development. HTML provides the structure of web pages, CSS handles styling and layout, and JavaScript adds interactivity and dynamic behavior to web applications.

## 3.5 TESTING

In the system, we plan to conduct various types of testing to ensure the reliability, functionality, and security of the application. These tests include:

1. Unit Testing: Unit testing involves testing individual components or units of code in isolation to verify that they function as expected.

2. Integration Testing: We perform integration tests to verify that the various parts of our application work together seamlessly, including frontend and backend integration, database interactions, and API endpoints.

3. User Interface Testing: User interface testing involves testing the graphical user interface (GUI) of the application to ensure that it is user-friendly, visually appealing, and responsive across different devices and screen sizes

4. Functional Testing: Functional testing verifies that the application functions correctly according to its specifications and requirements. We'll create test cases based on user stories and use cases to validate the core functionality of the system, including user authentication, event management, ticket purchasing, and payment processing.

5. Security Testing: We'll perform security testing to check for common web security issues such as SQL injection, cross-site scripting (XSS), cross-site request forgery (CSRF), and insecure authentication mechanisms. Additionally, we'll implement measures such as input validation, authentication, authorization, and encryption to enhance the security of our system.

6. Performance Testing: Performance testing assesses the responsiveness, scalability, and stability of the application under various load conditions. We'll use performance testing tools to measure response times, throughput, resource usage, and scalability of our system, ensuring that it can handle expected levels of traffic and maintain optimal performance under stress.

# CHAPTER 4: OUTPUTS AND IMPACTS

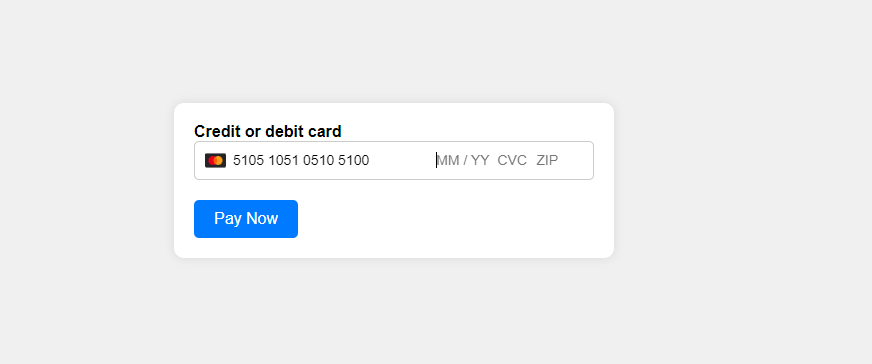
## 4.1 Results

The project's outcomes, which include the effective creation and implementation of an effective and user-friendly event management system, are encouraging. By making it simple for users to plan, organize, and take part in events, the system improves user involvement and streamlines the process of organizing events. A seamless implementation of essential elements like user identification, event registration, ticket purchasing, and payment processing results in a simple and easy-to-use user interface. The system's frontend development makes use of HTML, CSS, and JavaScript to produce an eye-catching and responsive user interface, while the backend framework, Django, guarantees stability, security, and scalability. Overall, the project's goals of providing a dependable and useful event management solution that enables customers to plan and attend events with ease have been met. Overall, the project's goals of providing a dependable and useful event management solution that enables customers to plan and attend events with ease have been met. The system's functionality and performance will be further improved by ongoing upgrades and maintenance, guaranteeing its success and customer pleasure.

## 4. 2 Modules Output

### 4.2.1 Credit Card payment

The event management system offers an easy-to-use Stripe integration for accepting card payments. The comfort and functionality of the system are increased when users can safely make payments with their credit or debit cards by utilizing the Stripe API. The system interfaces with Stripe to securely process payments using Django's backend functionalities. It handles encrypted transactions and complies with PCI compliance guidelines to protect user data. By offering a simple and safe payment method, this connection improves user experience by guaranteeing users a seamless event registration and ticket purchase experience while upholding the highest levels of security and dependability.



*Strip CLI credit card payment.*

## 4.3: Measurement of Software quality and reliability

In measuring software quality and reliability, several metrics and techniques are employed to ensure the system's performance meets the desired standards. This includes conducting various types of testing such as unit testing, integration testing, system testing, and acceptance testing to identify and rectify any defects or inconsistencies. Additionally, code reviews and static code analysis tools are utilized to assess the codebase for adherence to coding standards, maintainability, and robustness. Continuous monitoring and logging help track system performance in real-time, enabling prompt identification and resolution of any issues that may arise. By employing these methodologies, the software's quality and reliability are continuously assessed and improved, ensuring a stable and dependable system for users.

**4.4 Performance Analysis**

As part of the performance analysis process, statements or reports are provided to clients to evaluate the system's performance. These statements typically include key performance indicators (KPIs) such as response time, throughput, resource utilization, and error rates. By analyzing these metrics, clients can assess the system's efficiency, scalability, and overall performance against predefined benchmarks or service level agreements (SLAs). Additionally, performance testing results and recommendations may also be included to highlight areas for improvement and optimization. Overall, providing performance evaluation statements enables clients to make informed decisions regarding system usage, enhancements, and future scalability.

References:

- Aldehayyat, J. S. (2020). The Impact of Event Management Systems on Event Planning Processes: A Case Study of Jordanian Hotels. International Journal of Information Management, 50, 56-67.

- Büyüközkan, G., & Göçer, F. (2018). Event Management System: An Integrated Approach. In P. T. Altiok, J. F. Kinser, & J. J. Wiecek (Eds.), Operations Research and Enterprise Systems (pp. 232-252). Springer.

- Buhalis, D., & Foerste, M. (2015). SoCoMo Marketing for Travel and Tourism: Empowering Co-Creation of Value. Journal of Destination Marketing & Management, 4(3), 151-161.

- Chang, H. H., & Chi, H. R. (2010). The Impact of Online Store Environment Cues on Purchase Intention: Trust and Perceived Risk as a Mediator. Online Information Review, 34(2), 287-309.

- Hassan, A., Al Marri, A., & Harris, J. (2018). Factors Affecting the Adoption of Cloud Computing in Higher Education: A Quantitative Study in the UAE. International Journal of Information Management, 43, 1-8.

- Lee, H. A., & Gretzel, U. (2012). Y Generation's Perception of Mixed Reality Applications for Cultural Tourism: A Case Study of Yeosu Expo 2012. Journal of Travel & Tourism Marketing, 29(5), 463-476.

- Ratten, V. (2011). Event Management and Sustainability. Tourism Management, 32(6), 1415-1416.

- Sarstedt, M., Wilczynski, P., & Melewar, T. C. (2013). Measuring Perceived Brand Globalness: Scale Development and Validation. Journal of World Business, 48(1), 68-76.