

---

# **Software Requirements Specification**

**for**

**Blockchain-Enabled Secure Ticketing System: Advancing  
Transparency in Digital Transactions**

**Version 1.0 approved**

**Prepared by**

**RAJAN R (110820104037)**

**RAVINDRAN S B (110820104038)**

**Jaya Engineering College**

**14-03-2024**

## TABLE OF CONTENTS

<b>1.</b>	<b>INTRODUCTION</b>	<b>3</b>
1.1	Purpose	3
1.2	Scope	3
1.3	Overview	4
1.4	Reference Material	4
1.5	Definitions and Acronyms	5
<b>2.</b>	<b>SYSTEM OVERVIEW</b>	<b>6</b>
<b>3.</b>	<b>SYSTEM ARCHITECTURE</b>	<b>7</b>
3.1	Architectural Design	7
3.2	Decomposition Description	9
3.3	Design Rationale	20
<b>4.</b>	<b>DATA DESIGN</b>	<b>22</b>
4.1	Data Description	22
4.2	Data Dictionary	23
<b>5.</b>	<b>COMPONENT DESIGN</b>	<b>25</b>
<b>6.</b>	<b>HUMAN INTERFACE DESIGN</b>	<b>28</b>
6.1	Overview of User Interface	28
6.2	Screen Images	32
6.3	Screen Objects and Actions	36

## **INTRODUCTION**

### **Purpose**

This Software Design Document (SDD) outlines the architecture and system design of the Blockchain-Enabled Secure Ticketing System. The document serves as a comprehensive guide for software developers, providing essential information for the implementation of a secure and transparent ticketing platform.

The intended audience includes software developers, project supervisors, stakeholders, and potential users of the ticketing system.

The Blockchain-Enabled Secure Ticketing System is designed to revolutionize the ticketing industry by leveraging blockchain technology to ensure security, transparency, and fairness in ticket transactions. The system aims to eliminate issues such as ticket fraud, scalping, and unfair resale practices, providing users with a trustworthy and efficient ticketing experience.

Key features of the system include decentralized ticket issuance, non-fungible token (NFT) integration for ticket authenticity, smart contract automation for transparent transaction handling, and anti-scalping measures to promote fairness. Additionally, the system offers user-friendly interfaces for both administrators and ticket buyers, real-time updates on ticket availability, and secure payment processing.

The system also prioritizes user privacy and data security, with encryption techniques employed to safeguard sensitive information. Integration with existing ticketing platforms and infrastructure ensures seamless adoption and interoperability.

### **1.1 Scope**

The Blockchain-Enabled Secure Ticketing System is to modernize and enhance the ticketing industry by providing a secure, transparent, and user-friendly platform accessible across devices. Its objectives include developing a responsive ticketing platform, implementing blockchain technology for security and transparency, integrating non-fungible token (NFT) technology for ticket authentication, and improving communication and accessibility features. The system aims to streamline ticketing processes, improve security, increase transparency, and enhance user satisfaction for event organizers and ticket buyers.

## 1.2 Overview

This document serves as a comprehensive Software Design Description (SDD) for our project, the Blockchain-Enabled Secure Ticketing System. It outlines the architecture, data design, component design, and human interface design of the system. The document is organized as follows:

1. Introduction: Provides an overview of the purpose, scope, and objectives of the SDD.
2. System Overview: Briefly outlines the key functionalities and features of the ticketing system.
3. System Architecture: Describes the architectural design, decomposition, and rationale behind the system's structure.
4. Data Design: Presents the data description and data dictionary, detailing the structure and organization of data within the system.
5. Component Design: Details the design of individual software components, including their functionality and interactions.
6. Human Interface Design: Discusses the user interface design, including screen images and objects, to ensure a user-friendly experience.
7. Requirements Matrix: Summarizes the system requirements in matrix form, facilitating traceability and validation.
8. Appendices: Includes additional supplementary information and documentation as needed.

This document serves as a guide for software developers, project supervisors, and stakeholders involved in the development and implementation of the Blockchain-Enabled Secure Ticketing System.

## 1.3 Reference Material

- D. Zalar, R. Ušpalytė-Vitkūnienė, D. Rebolj and M. Lep, "A methodological framework for measuring the level of convenience of transport ticketing systems".

- J. Ding and L. Yu, "Analysis on the Construction of Online Marketing Platform for Performing Arts Ticketing under the Background of Big Data", International Conference on Cloud Computing and Information Science, vol. 750, 2019.

## **1.4 Definitions and Acronyms**

### **Definitions:**

1. Non-Fungible Token (NFT): A unique digital asset stored on a blockchain representing ownership or proof of authenticity of a specific item, such as a ticket or artwork, and cannot be replicated or exchanged on an equal basis.
2. Smart Contract: A self-executing contract with terms directly written into code. It automatically executes and enforces the terms of the agreement when predefined conditions are met, without the need for intermediaries.
3. Wallet: A digital tool or application used to store, manage, and interact with cryptocurrencies and digital assets securely. It typically includes functionalities for sending, receiving, and monitoring the balance of assets.
4. Consensus Mechanism: A protocol or set of rules used to achieve agreement among nodes in a distributed network regarding the validity of transactions and the state of the ledger. Examples include Proof of Work (PoW), Proof of Stake (PoS), and Practical Byzantine Fault Tolerance (PBFT).
5. Immutable Record: A record or data entry that cannot be altered or deleted once it has been added to the blockchain, ensuring data integrity and transparency
6. Encryption: The process of encoding information in such a way that only authorized parties can access it.

### **Acronyms/Abbreviations:**

1. NFT: Non-Fungible Token
2. IPFS : Interplanetary File System
3. AES: Advanced Encryption Standard
4. PoW: Proof of Work
5. PoS: Proof of Stake
6. QR Code: Quick Response Code

## 2. SYSTEM OVERVIEW

The Blockchain-Enabled Secure Ticketing System is a comprehensive digital platform designed to revolutionize the ticketing industry by providing a secure, transparent, and user-friendly ticketing experience for event organizers and ticket buyers alike. It serves as a centralized platform accessible to event organizers, ticket buyers, and other stakeholders involved in the ticketing process.

**Functionality:** The system provides a range of functionalities to support the entire ticketing lifecycle.

**Ticket Management:** Event organizers can create and manage tickets, set pricing, and define ticket availability.

**Ticket Purchase:** Ticket buyers can securely purchase tickets through the platform using various payment methods.

**Authentication:** The system ensures secure authentication and authorization for both event organizers and ticket buyers.

**Non-Fungible Tokens (NFTs):** NFTs are utilized to represent unique tickets, ensuring authenticity and preventing fraud.

**Smart Contracts:** Smart contracts automate ticket transactions, ensuring transparency and fairness.

**Anti-Scalping Measures:** The system implements measures to prevent ticket scalping and ensure fair distribution of tickets.

**Context :** The Blockchain-Enabled Secure Ticketing System operates within the context of the ticketing industry, aiming to address common challenges such as ticket fraud, scalping, and unfair resale practices. It caters to the needs of event organizers, ticket buyers, and other stakeholders, providing them with a modern and efficient platform to manage ticketing processes effectively.

**Design :** The system is designed with a user-centric approach, focusing on usability, accessibility, and security. It features a responsive web design to ensure accessibility across devices, and incorporates robust security measures such as encryption for sensitive ticketing details and transactions. The user interface is intuitive and easy to navigate, with clear delineation of functionalities and features.

**Background Information:** The need for the Blockchain-Enabled Secure Ticketing System arose from the challenges faced in traditional ticketing processes, including ticket fraud, scalping, and lack of transparency. By leveraging blockchain technology

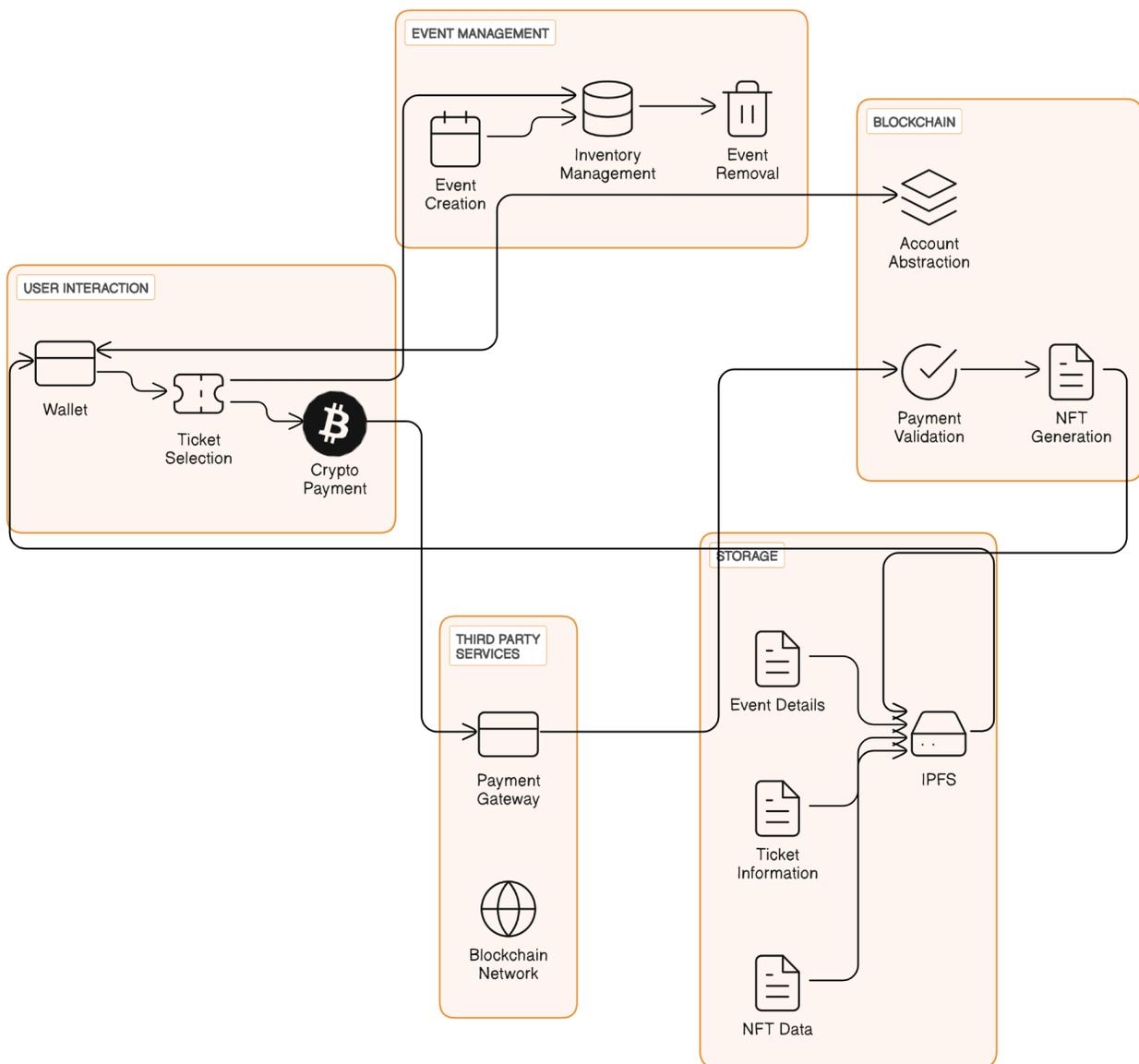
and modern design principles, the system aims to address these challenges and improve the efficiency and transparency of ticketing processes for all stakeholders involved.

### 3. SYSTEM ARCHITECTURE

#### 3.1 Architectural Design

Certainly! Let's delve into the architectural design of the Blockchain Enabled Secure Ticketing system

**Blockchain Enabled Secure Ticketing System Architecture**



The image depicts the following components and processes:

**User interaction :**

In the User interaction process, users begin by securely connecting their digital wallets to the platform, ensuring a safe and reliable transaction environment. Once connected, users navigate through the available events and select the one they wish to attend. Following event selection, users proceed to securely initiate and complete the payment using cryptocurrencies such as Bitcoin or Ethereum. Throughout this process, the system ensures the security and integrity of the payment transaction, providing users with a seamless and trustworthy ticket booking experience.

**Event management :**

In event management, organizers begin by creating an event and setting details like date, time, and ticket types. Users then browse and select events they're interested in, view event details, and choose tickets. Organizers monitor ticket sales and event logistics through a dashboard, sending updates to attendees. After the event, the system updates ticket statuses and removes event details. Users interact with the system through user-friendly interfaces and receive notifications for updates.

**Third-party services :**

Our platform seamlessly integrates with leading third-party wallet providers such as Metamask, Rainbow, and Trust Wallet to ensure secure and convenient transactions for our users. These wallet providers offer robust encryption and security measures, allowing users to securely store and manage their digital assets. Through our integration, users can easily connect their wallets to our platform, enabling them to initiate and manage cryptocurrency transactions effortlessly. This integration enhances the overall user experience by providing a familiar and intuitive interface for managing digital assets and engaging with our platform's features, ultimately ensuring a seamless and secure transaction process.

**Blockchain :**

Ticketing system employs Blockchain technology for fundamental functionalities through account abstraction, users interact securely with the system, safeguarding their private keys and enhancing overall security measures. Smart contracts facilitate payment validation, automating transactions securely and removing intermediaries, thereby mitigating fraud risks. Moreover, the creation of Non-Fungible Tokens (NFTs) on the Blockchain guarantees tamper-proof event ticket representation,

improving authenticity and ownership verification. Furthermore, Blockchain's decentralized and immutable characteristics ensure data security and privacy by encrypting user data and transaction details. This integration enhances trust and transparency in ticketing and event management processes, offering organizers and attendees a reliable, transparent, and secure solution that addresses challenges associated with traditional systems, ensuring a seamless experience for all stakeholders.

### **IPFS Storage:**

Utilizing IPFS storage enhances essential functionalities within the architectural diagram. IPFS provides decentralized and distributed storage, ensuring data availability and resilience against single points of failure. By leveraging IPFS, users can securely store and access event-related data, including ticket information and transaction details, with enhanced reliability and accessibility. Additionally, IPFS enhances data security and privacy by encrypting stored data and ensuring tamper-proof integrity through its immutable nature. This integration bolsters trust and transparency in ticketing and event management processes, offering organizers and attendees a reliable, transparent, and secure solution while mitigating risks associated with centralized storage systems.

## **3.2 Decomposition Description**

The Blockchain-Enabled Secure Ticketing includes Major three decomposition Subsystems they are:

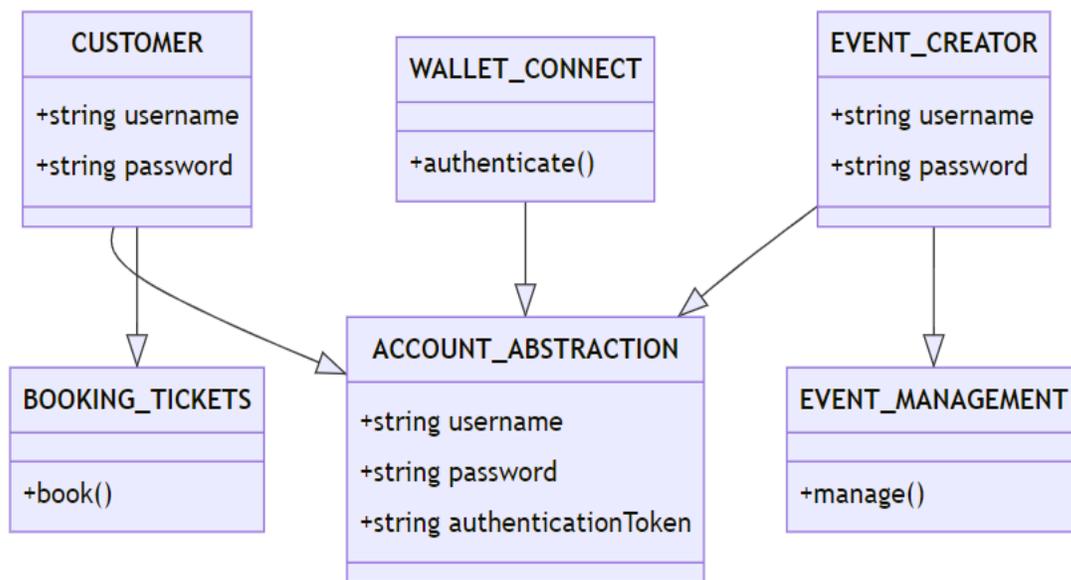
- User Authentication and Access Management
- Event Management and Ticketing
- Payment Processing and Transaction Management

### **3.2.1 User Authentication and Access Management**

The User Authentication and Access Management subsystem serves as the cornerstone of the platform, ensuring secure access for both customers booking tickets and event creators managing event details. It encompasses various functionalities to guarantee robust authentication and user privacy. Account creation enables users to securely register by providing essential details, while password management allows for effective password security and management. Additionally, the integration of

Wallet Connect and Account Abstraction ensures secure interactions with blockchain-powered features without compromising private keys. Once authenticated, customers can seamlessly browse and book tickets for events, while event creators have access to a dashboard to manage event details such as date, time, venue, and ticket types. This subsystem ensures a smooth and intuitive experience for all users, prioritizing security and privacy throughout the authentication and access process.

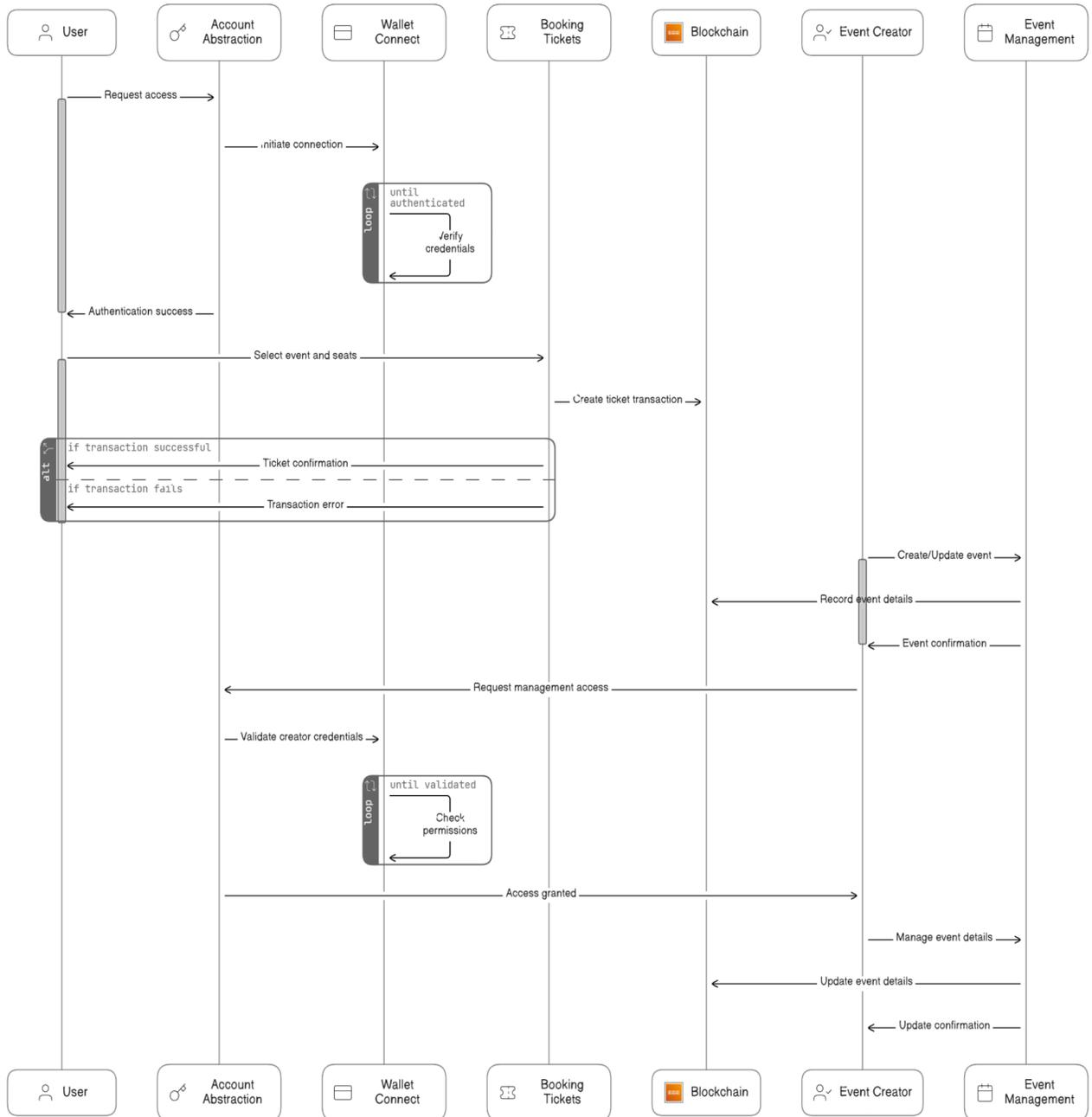
### Class Diagram :



**Description:** In this module, The User Authentication and Access Management subsystem is a cornerstone of our Blockchain-Enabled Secure Ticketing System, ensuring the integrity and security of user interactions. Consisting of the UserAuthentication and AccessManagement classes, it provides robust verification of user credentials and meticulous control over access permissions. UserAuthentication employs methods like authenticateUser() to validate user credentials, while AccessManagement governs access rights through functions like grantAccess() and revokeAccess(). This subsystem is pivotal in upholding user trust and confidence in the platform's security measures. With meticulous implementation and continuous refinement, it serves as a bulwark against unauthorized access attempts and data

breaches, reinforcing the platform's reliability and safeguarding sensitive user information.

### Sequence Diagram:



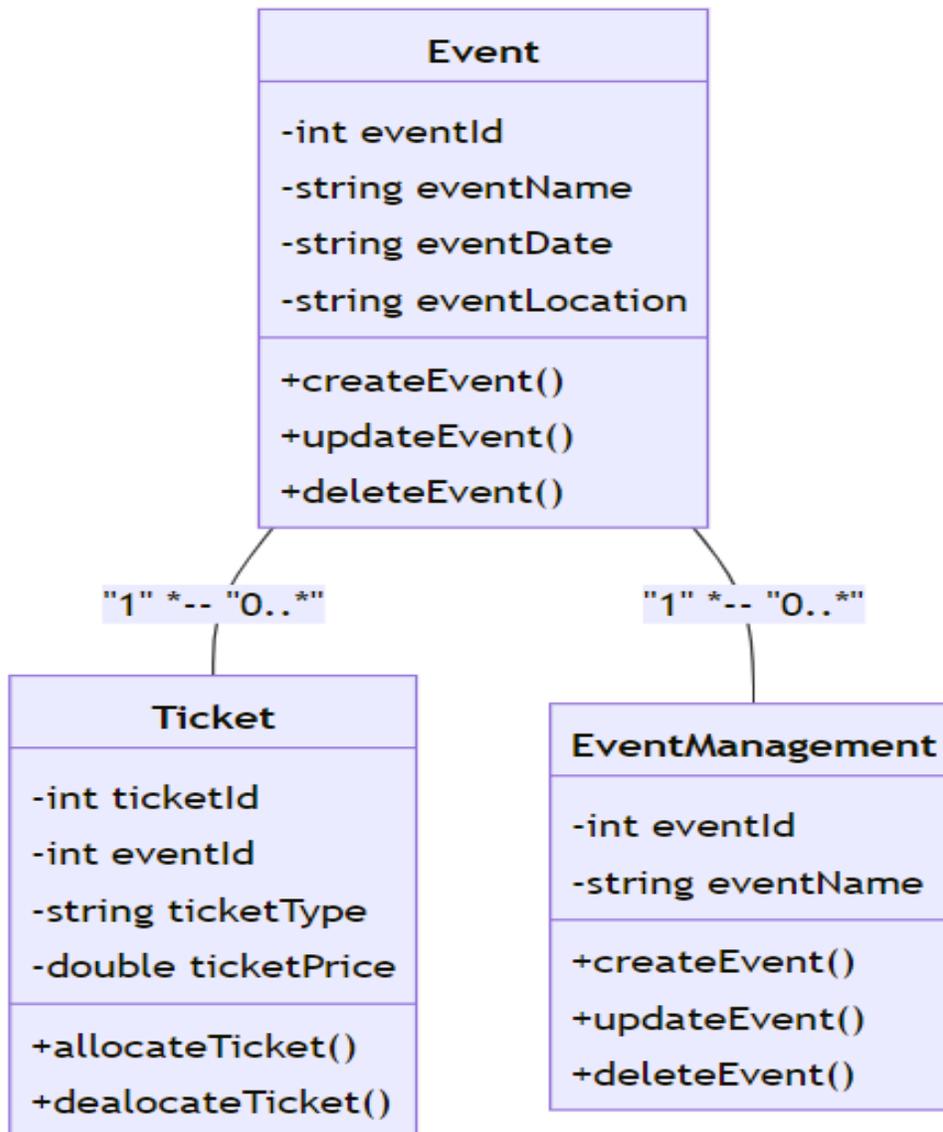
**Description :** The Sequence Diagram for the User Authentication and Access Management subsystem in our Blockchain-Enabled Secure Ticketing System delineates the orderly progression of steps involved in verifying user credentials and managing access permissions. It commences with a user login attempt, initiating the authentication process within the UserAuthentication class through the authenticateUser() method. Following successful authentication, the UserAuthentication class proceeds to authorize the user's access privileges using the authorizeUser() method. This step involves validating the user's role and permissions to determine the level of access they should be granted within the system.

Subsequently, the AccessManagement class evaluates the authorization request and assigns the appropriate access rights to the user using the grantAccess() method, enabling them to interact with the system's features as per their authorized permissions. In cases where access permissions require modification or revocation, the AccessManagement class facilitates these changes through the revokeAccess() method, ensuring alignment with the user's current role and responsibilities. Overall, the Sequence Diagram offers a concise visualization of the user authentication and access management process, emphasizing the seamless interaction between the UserAuthentication and AccessManagement classes to uphold secure and authorized user interactions within the system.

### **3.2.2 Event Management and Ticketing subsystem :**

In this Module, The Event Management and Ticketing subsystem is a critical component responsible for overseeing the entire lifecycle of events and managing ticketing operations. This subsystem encompasses functionalities related to event creation, ticket allocation, and event management, ensuring seamless coordination and execution of events within the system. Key features include event creation, where event organizers can input event details such as date, time, venue, and ticket types, facilitating the setup of new events within the system. Additionally, the subsystem manages ticket allocation, ensuring that tickets are allocated appropriately based on user requests and availability. It also facilitates event management operations, allowing event organizers to modify event details, manage ticket inventory, and track event attendance. Overall, the Event Management and Ticketing subsystem plays a pivotal role in orchestrating events and ticketing operations, providing a comprehensive solution for event organizers and attendees within the Blockchain-Enabled Secure Ticketing System.

## Class Diagram:



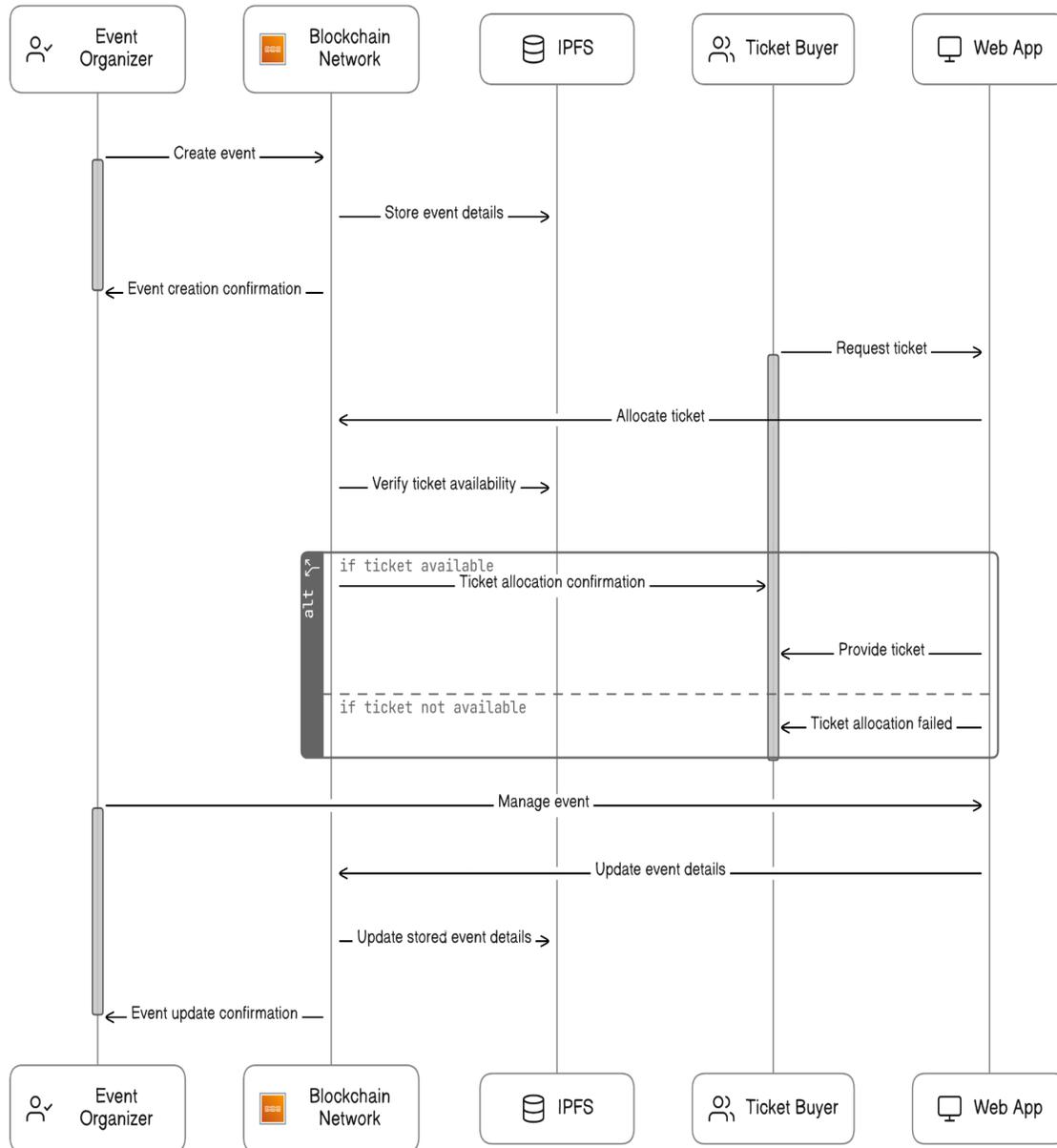
**Description:** In this module, The Event Management and Ticketing subsystem is a vital component of our Blockchain-Enabled Secure Ticketing System, overseeing event lifecycle and ticketing operations. It includes functionalities like event creation, ticket allocation, and event management, ensuring smooth coordination and execution

of events. Event creation allows organizers to input event details such as date, time, venue, and ticket types, facilitating seamless setup of new events.

Moreover, the subsystem manages ticket allocation to ensure appropriate distribution based on user requests and availability. It also supports event management tasks, enabling organizers to modify event details, manage ticket inventory, and monitor event attendance. In essence, the Event Management and Ticketing subsystem plays a central role in orchestrating events and ticketing operations, providing a comprehensive solution for organizers and attendees alike.

In summary, the Event Management and Ticketing subsystem of our Blockchain-Enabled Secure Ticketing System facilitates the entire event lifecycle, from creation to management. It streamlines event setup, ensures fair ticket allocation, and empowers organizers to efficiently oversee event operations. This subsystem is integral to the system's functionality, delivering a seamless experience for both organizers and attendees throughout the event management process.

## Sequence Diagram:



**Description:** The Event Management and Ticketing subsystem within our Blockchain-Enabled Secure Ticketing System follows a structured sequence to streamline event organization and ticket sales. It commences with event creation,

where organizers input vital details like event date, time, venue, and ticket types. This initiates the generation of event records, providing a central repository for managing event-related information efficiently.

Subsequently, the ticket allocation process ensures equitable distribution of tickets based on user requests and availability. This involves meticulous management of ticket inventory, assigning tickets to specific events, and updating availability in real-time to accurately reflect stock levels. By maintaining a synchronized flow of information, the system ensures fair ticket distribution and prevents overselling.

Finally, the event management phase empowers organizers to oversee various aspects of event planning and execution. They can modify event details, manage ticket inventory to prevent overselling, track event attendance, and address any last-minute changes or updates seamlessly. This structured approach enhances overall user satisfaction and contributes to the success of events hosted within our Blockchain-Enabled Secure Ticketing System.

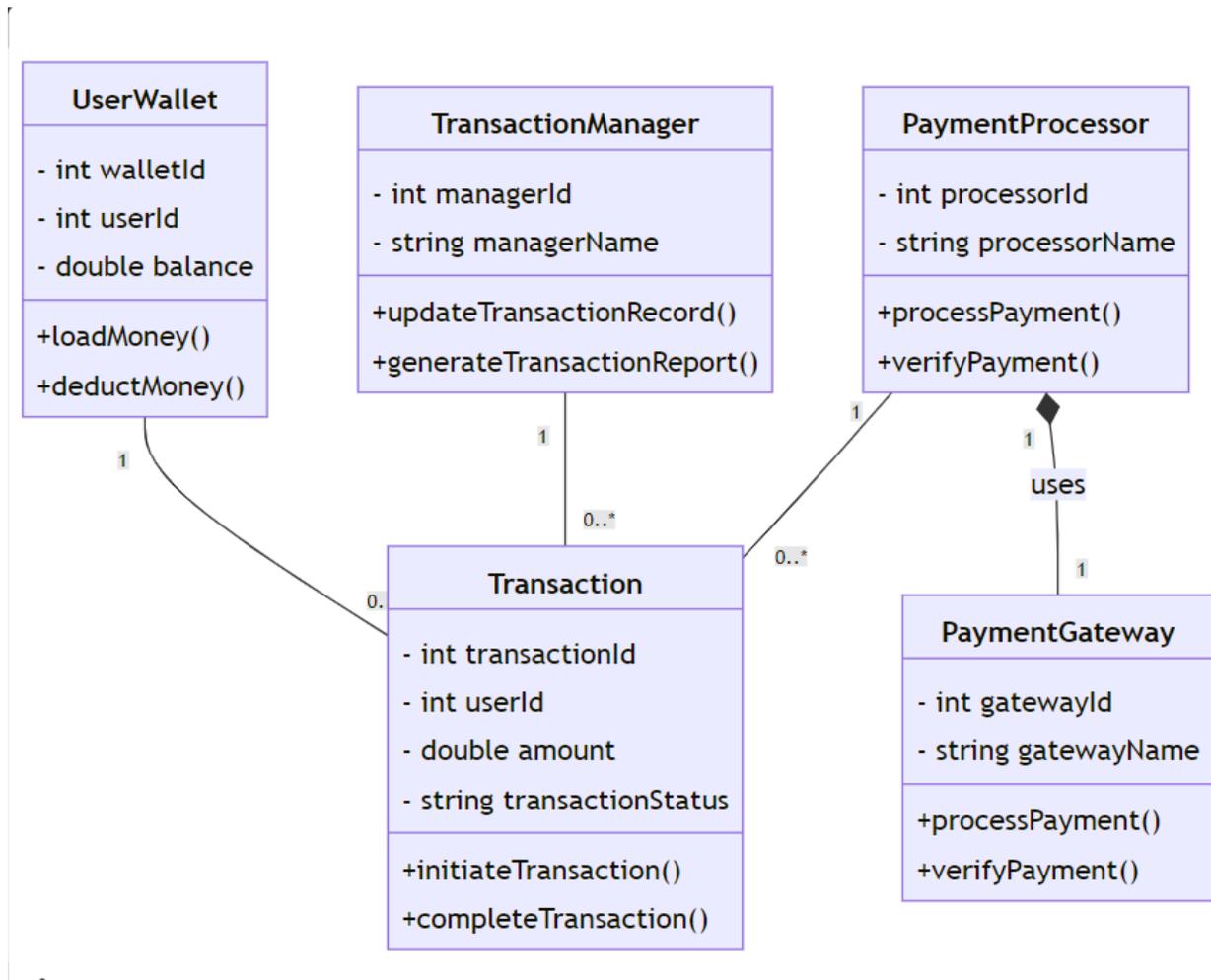
### **3.2.3 Payment Processing and Transaction Management:**

The Payment Processing and Transaction Management module is a pivotal component of our Blockchain-Enabled Secure Ticketing System, designed to facilitate seamless and secure transactions for ticket purchases. At its core, this module is responsible for overseeing the processing of payments made by users when purchasing event tickets. It integrates with various payment gateways or blockchain networks to handle transactions using a wide range of payment methods, including credit/debit cards, cryptocurrencies, and digital wallets. By providing users with multiple payment options and ensuring efficient transaction processing, this module enhances the overall user experience and accessibility of the ticketing platform.

In addition to payment processing, the Transaction Management aspect of this module plays a crucial role in maintaining accurate records of all financial transactions within the system. It systematically records transaction details such as unique transaction IDs, payment amounts, timestamps, and associated ticket purchases. This meticulous transaction management ensures transparency and traceability of financial transactions, enabling users and administrators to track and reconcile payment records effortlessly. By maintaining comprehensive transaction logs, the module promotes accountability and transparency in financial operations, contributing to the overall integrity of the ticketing platform.

Furthermore, the Payment Processing and Transaction Management module prioritizes the implementation of robust security measures to safeguard users' sensitive financial information and protect against potential security threats. Utilizing advanced encryption techniques, secure communication protocols, and adherence to industry-standard security practices, the module ensures the confidentiality and integrity of payment transactions. By maintaining a secure environment for financial transactions, this module instills trust and confidence in users, fostering a positive and secure ticket purchasing experience. Overall, the Payment Processing and Transaction Management module serves as a cornerstone of our ticketing system, enabling secure, efficient, and transparent transactions while upholding the highest standards of security and integrity.

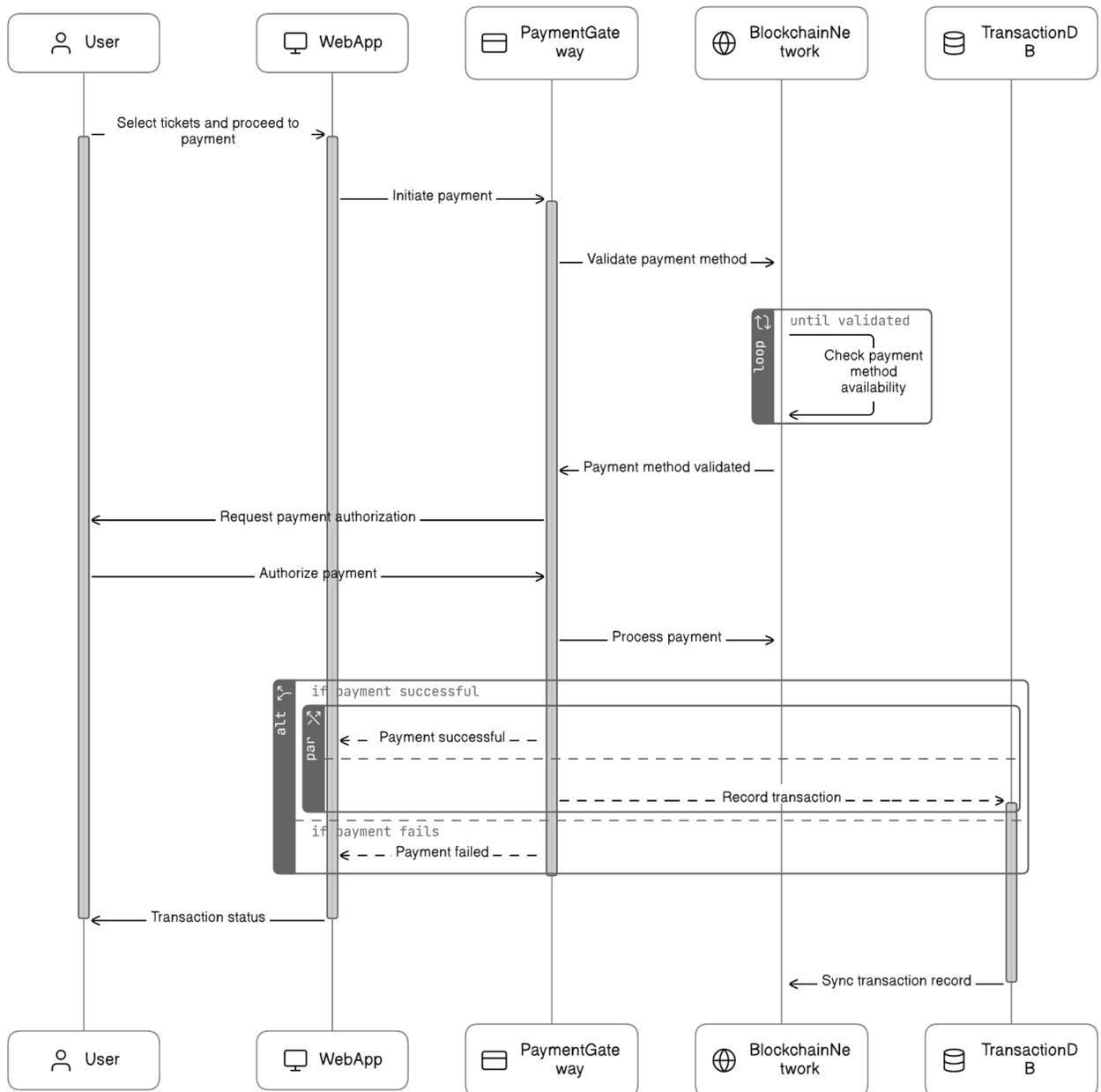
### Class Diagram:



**Description:** The Payment Processing and Transaction Management module within our Blockchain-Enabled Secure Ticketing System is an essential component tasked with overseeing all financial transactions linked to ticket purchases. Situated at the core of the system, it efficiently manages the processing of user payments across multiple channels. By integrating with third-party payment gateways or blockchain networks, the module enables transactions using various methods like credit/debit cards, cryptocurrencies, and digital wallets. It ensures the accuracy and reliability of these transactions, validating payments securely and facilitating the transfer of funds between users and event organizers.

Furthermore, the Transaction Management functionality of this module diligently records and administers all financial transactions occurring within the system. It maintains an extensive database containing transaction details such as unique transaction IDs, payment amounts, timestamps, and associated ticket purchases. This meticulous transaction management system ensures transparency and accountability in financial operations, allowing users and administrators to effectively track, reconcile, and audit payment records. Through its focus on security and efficiency, the Payment Processing and Transaction Management module enhances the overall user experience and inspires confidence in the integrity of financial transactions within the ticketing platform.

**Sequence Diagram:**



**Description:** The sequence diagram captures the intricate flow of interactions among key system components and actors involved in the ticket purchase and event attendance process. It effectively portrays the seamless orchestration of events from the user's initial selection of events and ticket quantities to the generation of unique NFTs representing the purchased tickets. The diagram also highlights the critical role of the payment gateway in securely processing user payments and confirming transaction completion, ensuring a smooth and reliable payment experience for users. Furthermore, it emphasizes the pivotal function of NFTs in providing users with

digital proofs of ticket ownership and enabling streamlined access to events, ultimately enhancing the overall efficiency and security of the ticketing system. Through its concise depiction of these interactions, the sequence diagram offers a comprehensive understanding of the system's operational dynamics and its ability to facilitate seamless ticket purchasing and event attendance experiences for users..

### **3.3 Design Rationale**

The architectural design for our Blockchain-Enabled Secure Ticketing System encompasses several high-level subsystems, each tailored to specific functionalities essential for a seamless ticketing experience. Here's an overview of the major subsystems identified, along with their designated roles and responsibilities:

**User Management Subsystem :** This subsystem orchestrates user registration, authentication, and login processes, ensuring secure access to the platform. It employs mechanisms like OTP generation for authentication and retrieves user details from a centralized database during login.

**Event Management Subsystem:** Responsible for overseeing the lifecycle of events within the system, including event creation, ticket allocation, and event management functionalities. It empowers event organizers with the ability to create and manage events, set pricing, and define ticket availability.

**Ticketing Subsystem:** Facilitates the ticket purchasing process for users, enabling secure transactions through various payment methods. It ensures a smooth and reliable transaction process for ticket buyers while integrating anti-scalping measures to prevent fraudulent activities.

**NFT Generation Subsystem:** Utilizes Non-Fungible Tokens (NFTs) to represent unique tickets, ensuring authenticity and preventing fraud in ticket transactions. It employs smart contracts to automate ticket transactions, fostering transparency and fairness.

**Communication Subsystem:** Enables effective communication between event

organizers and ticket buyers, providing real-time updates and notifications regarding event details and ticket availability. It includes features such as chat support and email notifications to facilitate seamless communication.

**Data Management Subsystem:** Acts as the repository for storing all user details, event information, and transaction logs securely. It ensures data consistency, integrity, and accessibility across the system, enabling efficient retrieval of information by other subsystems.

**Integration with Third-Party Services:** Integrates with third-party services like payment gateways and blockchain networks to enhance the functionality and user experience of the system. It ensures seamless transaction processing and interoperability with external platforms.

The architectural design of our Blockchain-Enabled Secure Ticketing System was meticulously crafted based on several critical factors:

**Scalability:** The chosen architecture allows for seamless scalability to accommodate a growing user base and increasing ticketing volumes.

**Security:** Incorporating features like secure authentication and encrypted data storage ensures the security and integrity of user information and transactions.

**Accessibility:** Providing options for standard and NFT-based tickets ensures accessibility and inclusivity for all users, regardless of their preferences.

**Integration:** Seamless integration with third-party services enhances the overall functionality and user experience of the system.

Alternative architectures, such as monolithic or serverless architectures, were considered but deemed unsuitable due to concerns about scalability, security, and customization limitations. The selected architecture strikes a balance between scalability, security, accessibility, and integration, ensuring a comprehensive and robust solution for secure ticketing.

## **DATA DESIGN**

## **Data Description**

Each user's data is stored to manage interactions within the system. This includes a unique User ID, Username for authentication, securely hashed Password, Email for communication, Phone Number for contact, User Role (e.g., event organizer, ticket buyer), and Registration Date. This information is stored in the Blockchain named 'Users'.

Details related to events and tickets are stored to facilitate event management and ticketing processes. This includes an Event ID, Event Name, Date and Time of the event, Venue Location, Ticket Types (e.g., general admission, VIP), Ticket Prices, and Ticket Availability Status. Additionally, each ticket is associated with a unique Ticket ID, Ticket Type, Event ID, Buyer ID, Purchase Date, and Ticket Status (e.g., purchased, reserved). This data is stored in the Blockchain named 'Events' and 'Tickets'.

NFT-related data is stored to manage the creation and distribution of NFT tickets. This includes an NFT ID, Ticket ID, Wallet Address of the ticket owner, NFT Metadata (e.g., event details, ticket type), and Creation Date. This information is stored in an IPFS Storage.

Information regarding financial transactions related to ticket purchases is logged. Each transaction is associated with a Transaction ID, Buyer ID, Seller ID (event organizer), Transaction Amount, Payment Method, Transaction Status, and Timestamp. This data is stored in Blockchain.

Communication logs between users, such as event organizers and ticket buyers, are recorded for reference and auditing purposes. This includes a Log ID, Sender ID, Receiver ID, Message Content, Communication Type (e.g., chat, email), and Timestamp.

Information related to the removal of events after 48 hours from their end is logged. This includes an Event ID, Event Removal Date, and Reason for Removal. This data is stored in the Blockchain named 'EventRemoval'.

Data related to the storage of event details, ticket information, and NFTs on the InterPlanetary File System (IPFS) is maintained. This includes IPFS File Hashes, File Sizes, and Timestamps. This data is stored in IPFS Storage.

These effectively organize and manage the essential information required for user management, event handling, ticketing processes, NFT management, financial transactions, communication logging, event removal, and IPFS storage within the Blockchain-Enabled Secure Ticketing System.

## **Data Dictionary**

**Users:**

UserID: int

Username: varchar

Password: varchar

Email: varchar

Phone: varchar

UserRole: varchar

RegistrationDate: datetime

**Events:**

EventID: integer

EventName: string

EventDateTime: datetime

VenueLocation: string

TicketTypes: string

TicketPrices: float

TicketAvailabilityStatus: string

**Tickets:**

TicketID: integer

TicketType: string

EventID: integer

BuyerID: integer

PurchaseDate: datetime

TicketStatus: string

**NFTs:**

NFTID: integer

TicketID: integer

WalletAddress: string

NFTMetadata: string

**Transactions:**

TransactionID: integer

BuyerID: integer

SellerID: integer

TransactionAmount: float

PaymentMethod: string

TransactionStatus: string

Timestamp: datetime

**CommunicationLogs:**

LogID: integer

SenderID: integer

ReceiverID: integer

MessageContent: text

CommunicationType: string

Timestamp: datetime

**EventRemoval:**

EventID: integer

EventRemovalDate: datetime

RemovalReason: string

**IPFSStorage:**

FileHash: string

FileSize: integer

Timestamp: datetime

These entities store crucial information about users, events, tickets, NFTs, transactions, communication logs, event removals, and IPFS storage within the system. Each entity has its own set of attributes defining its properties, contributing to the overall functionality and data management of the system.

## **Pseudo Code for User Authentication and Access Management:**

```
function userRegistration(username, password, email, phone):
```

```
  if username, password, email, and phone are not empty:
```

```
    if username is unique:
```

```
      createUser(username, password, email, phone)
```

```
      return true
```

```
    else:
```

```
      return false
```

```
  else:
```

```
    return false
```

```
function userLogin(username, password):
```

```
  if username and password are not empty:
```

```
    user = getUserByUsernameAndPassword(username, password)
```

```
    if user is not null:
```

```
      setSessionUser(user)
```

```
      return true
```

```
    else:
```

```
      return false
```

```
  else:
```

```
    return false
```

```
function createEvent(eventName, eventDate, venueLocation, ticketTypes,  
ticketPrices):
```

```
  if userIsLoggedIn() and userIsEventOrganizer():
```

```
    event = createNewEvent(eventName, eventDate, venueLocation)
```

```
    if event is not null:
```

```
      for each ticketType, price in ticketTypes, ticketPrices:
```

```
        createTicketForEvent(event, ticketType, price)
```

```
      return true
```

```
  return false
```

```
function purchaseTickets(eventID, ticketType, quantity, paymentMethod):
```

```
  if userIsLoggedIn():
```

```
    event = getEventByID(eventID)
```

```
    if event is not null and ticketsAreAvailable(event, ticketType, quantity):
```

```
      totalPrice = calculateTotalPrice(event, ticketType, quantity)
```

```

    if processPayment(totalPrice, paymentMethod):
        for i in range(quantity):
            issueNFTTicket(eventID, ticketType, getCurrentUser())
        return true
    return false

```

### **Pseudo Code For Event Management and Ticketing:**

```

function createEvent(eventName, eventDate, venueLocation, ticketTypes,
ticketPrices):

```

```

    if userIsLoggedIn() and userIsEventOrganizer():
        event = createNewEvent(eventName, eventDate, venueLocation)
        if event is not null:
            for each ticketType, price in ticketTypes, ticketPrices:
                createTicketForEvent(event, ticketType, price)
            return true
    return false

```

```

function viewEventDetails(eventID):

```

```

    if eventExists(eventID):
        event = getEventByID(eventID)
        return eventDetails(event)
    else:
        return null

```

```

function purchaseTickets(eventID, ticketType, quantity, paymentMethod):

```

```

    if userIsLoggedIn():
        event = getEventByID(eventID)
        if event is not null and ticketsAreAvailable(event, ticketType, quantity):
            totalPrice = calculateTotalPrice(event, ticketType, quantity)
            if processPayment(totalPrice, paymentMethod):
                for i in range(quantity):
                    issueNFTTicket(eventID, ticketType, getCurrentUser())
                return true
    return false

```

```

function viewPurchasedTickets():

```

```

    if userIsLoggedIn():
        tickets = getPurchasedTicketsForUser(getCurrentUser())
    return tickets

```

```
return null
```

### **Pseudo Code For Payment Processing and Transaction Management:**

```
function processPayment(totalPrice, paymentMethod):
```

```
    if paymentMethod == "credit_card":
```

```
        if validateCreditCard():
```

```
            deductAmountFromCreditCard(totalPrice)
```

```
            recordTransaction(totalPrice, "credit_card")
```

```
            return true
```

```
    elif paymentMethod == "crypto":
```

```
        if validateCryptoWallet():
```

```
            deductAmountFromCryptoWallet(totalPrice)
```

```
            recordTransaction(totalPrice, "crypto")
```

```
            return true
```

```
    elif paymentMethod == "paypal":
```

```
        if validatePaypalAccount():
```

```
            deductAmountFromPaypalAccount(totalPrice)
```

```
            recordTransaction(totalPrice, "paypal")
```

```
            return true
```

```
    return false
```

```
function recordTransaction(totalPrice, paymentMethod):
```

```
    transactionID = generateTransactionID()
```

```
    payerID = getCurrentUser()
```

```
    payeeID = getEventOrganizerID()
```

```
    transactionStatus = "completed"
```

```
saveTransactionDetails(transactionID, payerID, payeeID, totalPrice,  
paymentMethod, transactionStatus)
```

```
function viewTransactionHistory():
```

```
    if userIsLoggedIn():
```

```
        transactions = getTransactionsForUser(getCurrentUser())
```

```
        return transactions
```

```
    return null
```

## HUMAN INTERFACE DESIGN

### Overview of User Interface

#### **Title:**

The “Title” field is where event organizers set the stage. It’s akin to a captivating book cover—short, impactful, and enticing.

A well-crafted title should encapsulate the event’s essence, leaving potential attendees curious and eager to learn more.

#### **Description:**

The “Description” section is your canvas for storytelling. Here, event organizers weave a narrative that entices, informs, and excites.

Include event highlights, speakers, activities, and any unique selling points. Paint a vivid picture that resonates with your target audience.

#### **Date:**

The “Date” field is the heartbeat of your event. It’s more than just a calendar entry; it’s an invitation to a moment in time.

Specify the exact date, start and end times, and time zone. Clarity here ensures attendees don’t miss out.

## **Location:**

The “Location” section is your event’s GPS. Attendees need clear directions to join the journey.

Provide the full address, including any room numbers or specific instructions. Consider adding a map link for convenience.

## **Total Number of Seats:**

The “Total Number of Seats” field sets expectations. It’s like reserving a table at a restaurant.

Be transparent about seating availability. Whether it’s an intimate gathering or a grand conference, clarity matters.

## **Upload Banner Image:**

The “Upload Banner Image” is your visual billboard. It’s the first thing attendees see.

Choose an eye-catching image that reflects the event’s vibe—a concert stage, a serene landscape, or a futuristic design.

## **Upload Images of Event:**

The “Upload Images of Event” section adds depth. Think of it as a photo album capturing memories.

Include past event photos, artist profiles, or sneak peeks. Visual storytelling enhances engagement.

## **Notification:**

The notification bell icon is the event whisperer. It alerts users about updates, ticket availability, or interactions within the app.

Attendees stay informed without missing out on important event-related news.

## **Trending Events:**

The “TRENDING EVENTS” section showcases the hottest happenings. Each event listing includes:

An enticing image: Visual allure matters—it’s the event’s first impression.

Title: A succinct glimpse into what the event offers.

Date: Attendees need to know when to mark their calendars.

Location: Where the magic unfolds.

BUY button: The gateway to securing a spot at these sought-after events.

### **View all:**

The “View all” link beckons curious souls. Clicking it opens the treasure trove of additional trending events.

Attendees can explore beyond the initial glimpses, discovering hidden gems.

### **Image:**

The “IMAGE” placeholder is the canvas for event organizers. Here, they upload captivating visuals—a concert stage, a serene sunset, or an artist in action.

A compelling banner image sets the mood and sparks interest.

### **Authentication:**

The “AUTHENTICATION” button ensures security and trust. It’s the digital bouncer at the event entrance.

Users may need to authenticate their identity or ownership of the NFT ticket before accessing the event.

It’s a gatekeeper for exclusive experiences.

### **NFT Ticket:**

The “NFT TICKET” section is the heart of the interface. It visually represents the non-fungible token (NFT) ticket.

Each NFT is unique, verifiable, and tied to a specific event. It's a digital collectible granting access.

Attendees can proudly showcase their ownership and rarity.

### **About:**

The “About” tab provides context. It's the event's backstage pass, revealing what lies beyond the curtain.

Event organizers share detailed information: event purpose, speakers, agenda, and any special perks.

Attendees get a comprehensive overview before committing.

### **Event description:**

Within the “About” section, the “EVENT DESCRIPTION” area dives deeper.

It's the event's story—crafted with care. Attendees learn why this event matters, what to expect, and why they should be excited.

Clarity here ensures informed decisions.

### **Location:**

The “LOCATION” icon or button is a compass for attendees. It likely allows users to set their preferred location or discover nearby events.

Geolocation features enhance personalized event recommendations.

Attendees can find events relevant to their physical context.

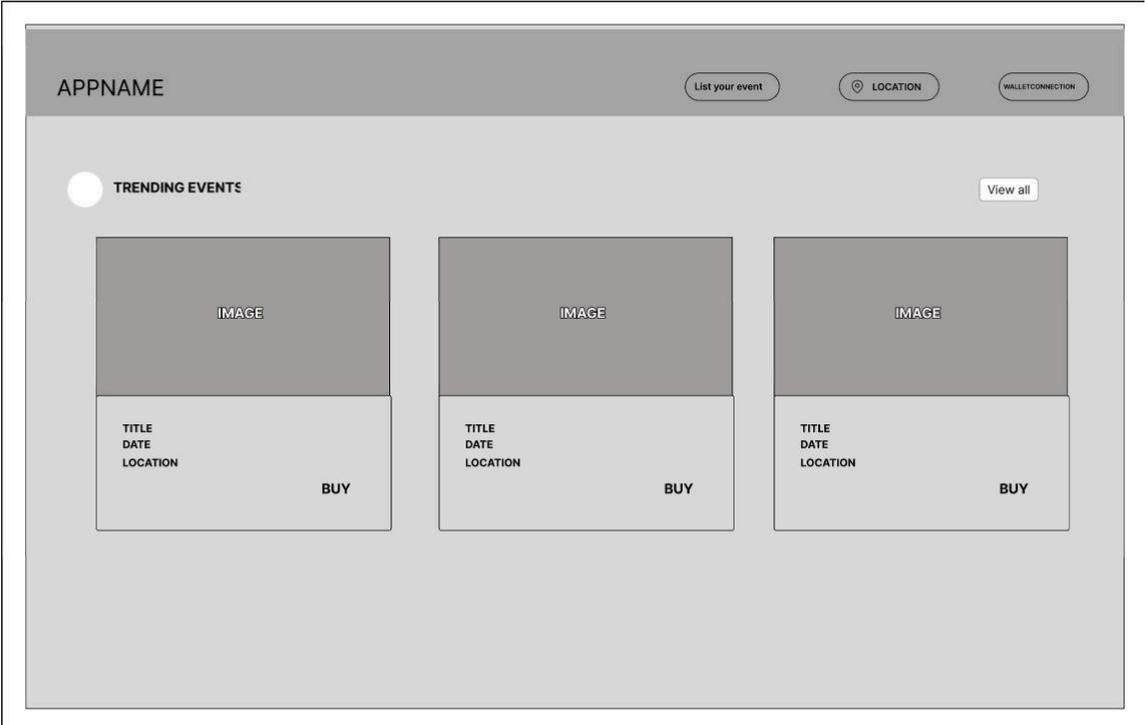
### **Event has been created:**

The “Event has been created” notification bar provides immediate feedback. It's the green light after a successful event setup.

Users feel assured that their event is live and ready for attendees.

Clarity here ensures confidence in the process.

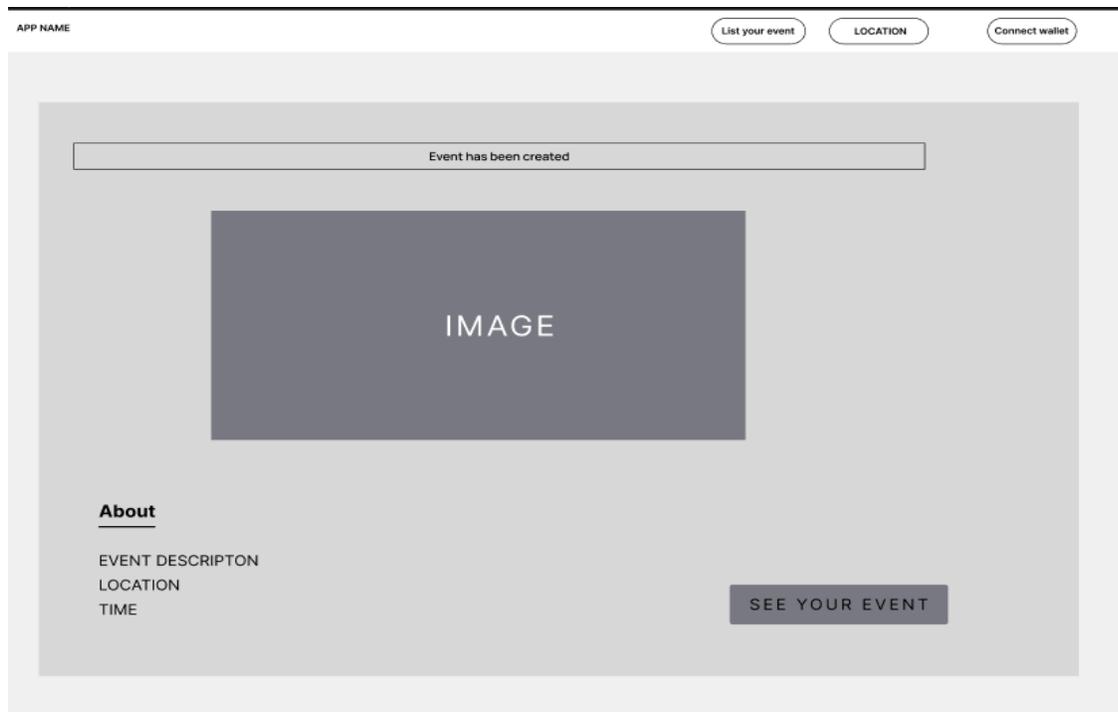
### Screen Images



In the realm of Web 3.0, the “APP NAME” serves as the digital marquee, proudly displaying the platform’s identity and instilling a sense of trust and recognition. The “List your event” button stands as a beacon for event organizers, inviting them to share their unique experiences and enrich the platform’s tapestry with diverse events. The “LOCATION” feature acts as a digital compass, guiding attendees to events that resonate with their locale, while the “Connect wallet” button is the key to unlocking the event’s treasures, ensuring secure and verified access. Upon successful event creation, the “Event has been created” notification affirms the fruition of one’s efforts, providing a sense of accomplishment and readiness for the upcoming gathering. The “IMAGE” placeholder is the visual storyteller, allowing organizers to set the stage with images that capture the essence of the event, enticing attendees with a preview of the experience. Lastly, the “About” section, with its detailed “EVENT DESCRIPTION,” “LOCATION,” and “TIME,” offers a narrative of the event, painting a picture of what attendees can anticipate, grounding the event in reality, and setting the stage for an immersive experience.



In the innovative landscape of Web 3.0, the “APPNAME” serves as a beacon of recognition, anchoring the user’s journey with a sense of trust. The “List your event” button stands as a testament to empowerment, inviting organizers to weave their narratives into the platform’s fabric. The “LOCATION” feature acts as a digital compass, guiding attendees through a personalized event discovery process. The “AUTHENTICATION” mechanism fortifies the gateway, ensuring secure and exclusive access to events. At the core lies the “NFT TICKET”, a unique emblem of participation and ownership, symbolizing the attendee’s entry ~into~ a curated experience. The “About” tab unfurls the narrative tapestry of the event, offering a backstage glimpse into the purpose, speakers, and exclusive perks. Within this section, the “EVENT DESCRIPTION” delves into the story’s heart, painting a vivid picture of the event’s essence and inviting attendees to immerse themselves in the unfolding spectacle



1. In the dynamic world of Web 3.0, the “APP NAME” section is the digital handshake, welcoming users and establishing the platform’s identity. The “List your event” button serves as a portal for organizers to share their events, fostering a rich tapestry of experiences. The “LOCATION” feature acts as a navigator, guiding attendees to events that resonate with their locale, while the “Connect wallet” function is the key to seamless and secure event access. The “Event has been created” notification provides essential feedback, assuring organizers of their event’s readiness. The “IMAGE” placeholder is a visual prelude, allowing organizers to set the stage with images that capture the essence of the event, enticing attendees with a preview of the experience. Lastly, the “About” section, with its detailed “EVENT DESCRIPTION,” “LOCATION,” and “TIME,” offers a narrative of the event, painting a picture of what attendees can anticipate, grounding the event in reality, and setting the stage for an immersive experience.

## Screen Objects and Actions

A discussion of screen objects and actions associated with those objects.

Screen Images	Screen Objects	Actions
Homepage	Logo	Click Logo: Return to the homepage
	Navigation Bar	Click Icons: Navigate to different sections of the application
	Book Event	This button likely leads users to a booking events page .
	Create Event	This button likely leads users to a Create events page
Event Page	Connect wallet	Users can connect their digital wallet by clicking the button
	Event banner	Users can learn more about an event and purchase tickets by clicking on an event card.
Booking Page	Details card	Users can learn more about an event .
	Buy button	Users can purchase tickets by clicking this button .

Event Creation Page	Create Creation Form	The event creation form lets you set up your event. Fill in the details like name, description, category, and upload an image. Decide on the ticket price and quantity. You can even offer different ticket tiers. Finally, hit "Publish Event" to make it live for ticket sales, or save it as a draft for later.
Tickets Page	NFT Tickets	Users can able to view all their NFT tickets on this page .

