NFT Event Ticketing System

Vashti Vinodhini Anbusamy
1221101168
Lecture group : TC1L
Tutorial group : TT1L
1221101168@student.mmu.edu.my

Bibi Nurhusnina Binti Mustag Ahmad
1221101181
Lecture group : TC1L
Tutorial group : TT1L
1221101181@student.mmu.edu.my

Aisyah Binti Ahmad Komarolaili
1211101519
Lecture group : TC1L
Tutorial group : TT1L
1211101519@student.mmu.edu.my

Mohammed Aamena Mohammad AbdulKarem
1221305728
Lecture group : TC1L
Tutorial group : TT1L
1221305728@student.mmu.edu.my

Abstract*—Existing event ticketing systems have faced problems such as fraud and limited oversight. This paper presents an NFT-based ticketing system using blockchain technology, ERC-721, and smart contracts to try to solve some of these issues. This proposed ticketing system will have immutable tickets, verifiable ownership, and programmable resale rules. A ReactJS frontend was developed that leverages MetaMask for an enhanced user experience. Testing revealed that it can successfully implement NFT tickets and manage them over time. Performance statistics showed manageable gas usage in consideration of the relatively higher gas costs of essential functions. This project presents a solution that is both transparent and secure, user-friendly approach to modern event management.

Keywords*— NFT Ticketing, Blockchain, Smart Contracts, Ethereum, MetaMask, Decentralized Applications, Event Management

I. Introduction

A. Background

The ticketing sector has long suffered from widespread issues such as fake tickets, ticket scalping, unclear resale practices, and absence of regulatory oversight by event organizers over secondary market sales. These problems affect both traditional paper-based and modern electronic ticketing systems. Neither approach reliably guarantees ticket authenticity, and both remain vulnerable to fraud and unauthorized duplication. Such vulnerabilities erode consumer trust, hinder fair access to events, and result in financial losses for both organizers and attendees.

Non-Fungible Tokens (NFTs), built on blockchain technology, offer an innovative solution characterized by decentralization, immutable and verifiable ownership records. Developers frequently utilize Ethereum's ERC-721 standard to construct unique digital assets that function as secure and transferable tickets. NFTs provide cryptographic proof of authenticity while maintaining a transparent history of ownership. Furthermore, smart contracts facilitate the enforcement of programmable rules to govern resale and transferability.

Recent implementations, such as the NFT Ticketing Application developed by Lawal [1], illustrate the practical integration of blockchain technology within real-world ticketing systems. This application combines Solidity smart contracts with a React-based front-end, providing features such as wallet authentication, QR code-based ticket validation, NFT generation, and attendance verification. Similarly, educational platforms such as Egghead's NFT ticketing course [2] present end-to-end workflows that demonstrate how Web3 technologies can be integrated into contemporary ticketing infrastructures.

As decentralized ticketing solutions gain continued attention, it is essential for researchers to critically evaluate these systems, with particular emphasis on their scalability, usability, and capacity to address real-world operational demands.

B. Problem Statement

Although being based on a blockchain, NFT ticketing systems enable a unique set of benefits in terms of security, traceability, and transparency, the practical implementation of these systems faces several key challenges that will ultimately influence adoption and functionality:

- Scalability and Transaction Latency: First, there are challenges related to scalability and transaction latency on public blockchain networks (e.g. Ethereum) that fundamentally inhibit real-time ticketing transactions and hinder throughput/financial transactions attributed to existing high-gas fees during peak, or heightened, ticketing events [3].
- User Onboarding Difficulties: Second, user onboarding represents a significant challenge, as non-technical users struggle with understanding blockchain-specific concepts like crypto wallets, private keys, smart contracts, all the intermediaries involved, hesitation following costs

associated with undesired ticket outcomes leading to lower adoption [4].

- Regulatory Compliance Risk: Third, there is a level of regulatory compliance risk surrounding the NFT concept where the risks may include ambiguity around consumer protection, resale rights, data privacy, and taxes. Using NFTs can result in confusion for developers and even event organizers as these are not adequately defined, and there is no standard legal framework from jurisdiction to jurisdiction [5].
- Fraudulent Resale and Cyber-Ticket Scalping Regulation:
  NFT ticketing systems must address the problem, and unfavourable resale practices, of how resale markets are used by scalpers to raise prices and hinder fair access to tickets. Our system aims to effectively address this issue by requiring resale rules in a transparent, blockchain-based resale system and ownership verification to offer fair prices, while also addressing ticket fraud. [6].

This project focuses on creating a scalable, user-friendly, and regulation-cautious end-to-end NFT ticketing product. The authors will create a smart contract that presents an optimized view for users encompassing onboard platforms with an effective design adaptable to changing regulations.

C. Project Objectives

The purpose of the project is to:

1. Improve Transparency and Authenticity: Immutable and verifiable ownership records in the form of NFTs will guarantee tamper-proof ticket issuance and tracking.

2. Ensure Fair Resale: Smart contracts will carry the resale rules and restrain scalping and overpricing, thereby improving trust and fairness in the ticket resale process.

3. Improve Ticket Distribution and Control: The system will enable organizers to control an entire end-to-end ticketing life cycle (minting, resale, redemption) by decentralized programming logic without intermediaries.

This blockchain solution aims to replace existing deficient systems with secure, transparent, and user-friendly alternative.

II. Literature Review

Blockchain technology and Non-Fungible Tokens (NFTs) offer unique solutions to event ticketing that overcome the problems with traditional centralized systems. Ticketing systems have always used centralized systems, which are susceptible to fraud, duplication, unauthorized resale, and price manipulation. There is no assurance of authenticity with this centralization, and event planners have little to no control over secondary ticket sales.

Blockchain powered decentralized ticketing solutions are proving popular because they have transparency and immutable records while allowing for more clear ownership tracking. NFTs allow tickets to be represented as unique digital assets that can be securely and verifiably exchanged, while also enabling fair resale if the buyer cannot attend.

A. Survey of related work

The concept of NFT-based ticketing has been investigated by both industry and academia resulting in viable applications and real-world implementations. GUTS Tickets is one of the earliest real-world implementations in NFT ticketing. While the launch in 2019 predates DLT claim rules ahead of Amsterdam's ticketing regulatory law, the organization is built on the Guaranteed Entrance Token (GET) protocol. Not only was the implementation able to provide verifiable and permanent issuance of tickets but also allows the event organizer to apply restrictions to ticket resale. This enables reduced fraud through ticket traceability inherent in the blockchain verification [7].

Another example of NFT-based ticketing is SeatlabNFT, also built on Ethereum by using the ERC-721 standard. The system offered various elements such as programmable royalties, token-gated benefits, and real-time transparency through the public ledger. This not only provided visibly traceable revenue ownership relative to the benefit but encouraged user trust. However, the Ethereum solution also limited demand through scalability issues due to cost of gas fees, a product of an external posting and availability mechanism the system operated with [8].

Schmid and Golling (2022) also researched a decentralized multi-channel ticket platform with an identity verification process called DeTi. The system not only proposed using smart contracts for dynamic pricing and access control but considered important issues such as legal compliance and scalability. Luckily, since its development in 2022, it has held up similarly to Prototype development in this study, although DeTi has not yet emerged beyond the prototype stage [9].

Verma and Patel (2024) described a ticketing system based on smart contracts that links NFT tickets to seats and includes support for MetaMask wallets. Their solution prioritizes gas optimization and ease of use of issuing tickets, while preventing double booking of tickets. This research quantified an impact of smart contract efficiency with respect to scalability and uptake [10].

Akhtar et al. (2023) described akaTick, a hybrid NFT-based e-ticketing system that combines convenience of mobile use with decentralized security in blockchain provision. To prevent unauthorized use or double-spending, each transaction in ticket usage requires two approvals from

the event organiser and the ticket owner. The two-step verification creates trust and usability in mobile blockchain ticketing [11].

Sharma and Mehta (2024) proposed a decentralized ticketing model with automated ticket issuance and resale, as facilitated by smart contracts. To combat fraud and scalping, their model incorporates transaction traceability and resale to a predetermined price point. Additionally, it indicates how decentralized marketplaces may be a socially fairer process of access and provides a level of transparency for ticket transfers [12].

Verma and Patel (2024) describe a system like the author's prototype, where a complete NFT ticketing system was deployed on the Ethereum Goerli testnet. It includes fully working smart contracts, written in Solidity, connection via MetaMask with wallet access, along with a front-end developed with ReactJS. Although their solution was built on a test network, they discuss real-world limitations of their experiences, such as gas fee prices and smart contract scalability under load [10].

Table 1. *Comparative Summary of Existing NFT Ticketing Platforms*

Ref	Platform / Study	Blockchain / Protocol	Key Features	Limitations
[7]	GUTS Tickets	GET Protocol (NFT)	Verifiable NFTs, resale governance, fraud prevention	Limited international rollout
[8]	SeatlabNFT	Ethereum ERC-721	Royalties, ownership tracking, token-gated rewards	High gas costs on Ethereum mainnet
[9]	DeTi	Multi-chain (prototype)	Identity verification, dynamic pricing, layered access	Still in prototype, regulatory pending
[10]	Verma & Patel (IJRPR)	Ethereum (Goerli testnet)	Seat-linked NFTs, MetaMask, gas optimization, dynamic pricing	Limited to testnet; performance unverified
[11]	AkaTick	Hybrid NFT-Mobile	Dual-user validation, real-time trust, double-spend protection	Not fully decentralized
[12]	Sharma & Mehta (IRJET)	Ethereum	Anti-scalping, smart resale logic, decentralized market	Early-stage; lacks extensive deployment

These comprehensive analyses demonstrate the potential constraints of existing systems, scalability challenges, onboarding complexity for users, and uncertainty in regulatory frameworks as ongoing risks. Our work is inspired by these reviews to develop an integrated and accessible NFT ticketing system that includes security, transparency and costs reduction.

B. Identified gaps

While NFT-based ticketing systems are gaining traction and notoriety, many barriers still exist to achieve mass-market adoption. A significant barrier to entry is the overall lack of blockchain technology knowledge on the parts of event organizers and end-users which leads to reluctance or low uptake. This barrier is exacerbated by the additional layers of complexity involved in managing digital wallets with private keys, which is a foreign or intimidating prospect for all but the most sophisticated tech users.

Scalability is also a practically insurmountable barrier. Public blockchain networks such as Ethereum suffer from network congestion even in lower-traffic events and speculative Hong Kong NFT events are subject to high gas fees. When network performance is throttled, NFT issuance and validation become economically unviable as uptake increases, especially during high-traffic or larger events.

Consumer protection, intellectual property, and taxation on the sale of NFTs are just some of the issues being assessed and discussed worldwide and unlike traditional tickets, NFTs are very much at the whims of regulators. Greater ambiguity with this new frontier will certainly challenge uptake by corporate clients or well-known institutions.

Lastly, some event organizers may not have the technical infrastructure, infrastructure benefits, or weight to develop or maintain DApps. Most would require the services of a third-party vendor for DApp development or hosting; however, engaging third-party developers contradicts the decentralized spirit of blockchain, incurs additional costs, and risks centralizing ownership of NFT tickets and ecosystem.

Collectively, all these issues suggest a need for solutions offering:

- Gas-efficient smart contracts
- User-friendly interfaces through wallets
- Clearer regulatory compliance paths
- Lowering or removing technical barriers for event organizers

C. Relevance to our project

This project provides a simple solution to the urgent problems faced by NFT-based ticketing systems through the offer of a decentralized ticketing system centered on transparency, ease of use, and affordability. By using Ethereum smart contracts and the ERC-721 standard, this system issues NFT-based tickets as immutable and verifiable digital assets. The tickets are uniquely assigned and securely stored in-chain, excluding duplication and ensuring authenticity throughout the event's life cycle.

To address the scalability and gas fee issues most associated with Ethereum, the project is built using optimized Solidity code to reduce on-chain storage reads and computation. This is compounded with development and testing against a local blockchain with deployment and automation by Hardhat. This setup provides the ability to fully test and benchmark performance without paying actual gas prices, providing a realistic yet budget-friendly environment for development.

The system's usability gap is bridged with an easily accessible interface based on ReactJS and MetaMask integration. It removes technical barriers for event organizers and attendees, enabling ticket creation, purchase, and transfer without needing blockchain inner workings expertise. MetaMask makes it easy for users to manage authentication and transactions with minimal resistance, allowing non-technical individuals to adopt it.

In terms of resale governance, the platform allows event organizers to create resale permissions and define price caps (e.g., up to a 10% markup), all triggered automatically by smart contracts. These features support fair access and sharing of revenue as well as deter scalping and unauthorized resale—problems that are inherent in traditional systems.

A factory contract pattern is utilized to allow decentralized event creation. This allows organizers to deploy separate event contracts independently without depending on third-party services, a feature that is epitomizing the decentralized spirit of blockchain.

In summary, the project solves for critical deficiencies in the literature—i.e., high gas costs, bad usability, regulatory risk, and insufficient infrastructure—using an industry-specific solution based on scalable architecture, user-oriented design, and automated smart contracts.

D. Research Questions / Hypotheses

This project investigates the following implementation-focused questions:

1. How can the smart contract securely mint and manage event tickets with constraints on resale and usage status?
2. How effectively can MetaMask wallet authentication enable check-in functionality within a decentralized application?
3. What are the gas and performance implications of deploying dynamic ticketing features—such as staged events and resale caps—on a simulated Ethereum test environment?

III. Methodology

A. System Architecture and Design

The proposed system is designed as a decentralized application (DApp) that modernizes traditional ticketing via blockchain technology. The system has three main parts:

1. Smart Contract Layer:

This layer is developed in Solidity (v0.8.21) utilizing OpenZeppelin's implementation of the ERC-721 standard. A factory contract (e.g., NFTticketing.sol) dynamically deploys event contracts (e.g., Event.sol). Each contract is hierarchical, meaning it is made up of numerous modules (for example, inheriting the contract ERC721, Ownable, ReentrancyGuard and Pausable). It provides access control, security, and upgrade features for tickets, while implementing a time-based life cycle, ticket minting, resale and markup as for resale logic, and ultimately organizer only functions.

2. Frontend Interface:

This layer utilizes ReactJS and is styled using Tailwind CSS. Frontend development inherently supports an interactive and responsive user experience. Users complete the full user workflow, which includes event creation, ticket purchasing, ticket resale, and ticket redemption. MetaMask is integrated within the React (front end) layer via Ethers.js for the wallet connection and authenticating and signing the blockchain transaction. Pinata API (IPFS) is used to upload and pin the content where upon successful upload the metadata associated with the ticket is also stored on the IPFS decentralized storage.

3. Blockchain Environment:

Hardhat manages the development environment and simulates a local Ethereum blockchain for development and testing. Deployment scripts optimize gas usage by enabling the Solidity optimizer with 200 runs. The development environment supports unit testing and integration testing to validate performance against various edge cases and transaction states.
This architecture is modular and allows for decentralized, scalable ticket distribution. Each event contract is created with several configurable parameters including total ticket supply, base price, resale markup (capped at 10%), royalties, sales windows and event duration.

B. Smart Contract Development

The smart contract system proposed here is structured to impose comprehensive and secure logic for controlling the whole life cycle of event tickets, all through blockchain technology. Each ticket is an ERC-721-based unique non-fungible token (NFT) with a metadata location hosted on the InterPlanetary File System (IPFS) to allow for decentralization and immutability. The system securely generates, stores, and hosts token URIs dynamically after minting to include event-specific information, which is key for governance logic. Minting relies on supply control constraints on the numTicketsLeft variable to limit and track supply. The terms of the smart contract govern secondary market transactions (resale) with restrictions on resale pricing and the power to approve or reject transfer requests. The terms of the smart contract govern resale price caps along with the ownership transfer and enforce conditions on resales that promote fair resale practice. The system automatically sends royalties from each resale to the event organizer's wallet to support sustainable events. Tickets progress through distinct states — Available, Sold, Listed, and Used — providing full transparency and tracking. Each event lifecycle encapsulates multiple states — Created, Active, CheckInOpen, Ended, and Cancelled — defined over a blockchain timestamped interval backstopped on the blockchain timestamps that give parties explicit control of time validity. The event states to control ticket resale or use outside the acceptable time interval as dictated by the event lifecycle. Prioritizing security, the contract and transactions include built-in safeguards.

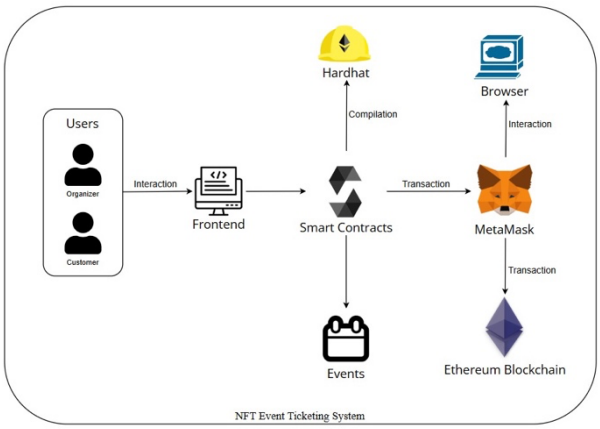

Figure 1. *NFT Event Ticketing System*

Figure 1 illustrates the proposed architecture of the NFT Event Ticketing system in terms of decentralized application (DApp) components. The system framework contains three main layers: Smart Contract Layer, Frontend Interface, and Blockchain Environment.

The two user-types, "Organizer" and "Customer," start using the system with the "Frontend." The Frontend application layer was developed with ReactJS and styled with Tailwind CSS components to allow for interactivity when creating events, purchasing tickets, reselling tickets, and redeeming tickets. MetaMask allows the React frontend to connect to the user's wallet, authenticate the user, and sign transactions on the blockchain using Ethers.js.

The "Smart Contracts" layer is built on Solidity (v0.8.21) using OpenZeppelin's ERC-721 standard, representing the logic of the ticketing system. The ticketing system utilizes a factory contract that deploys separate contracts called "Events" with various custom parameters (e.g., a total supply of tickets, price of tickets, markups for resale, royalties, sale windows, length of event). The event contracts lifecycle is based on the time within which tickets can be minted and resold. Hardhat was used to create the development environment which compiles smart contract code and resembles a local Ethereum blockchain for development and testing.

Transactions made through MetaMask interact with the "Ethereum Blockchain", which is the decentralized and secured platform to record all actions and data in the system. Ticket metadata is uploaded and pinned on the IPFS (InterPlanetary File System) via Pinata API, to ensure the ticket metadata is decentralized and immutable.

C. Technologies Used

Table 2. *System Technologies and Tools*

Component	Technology	Purpose/Justification
Blockchain Layer	Ethereum	Decentralized, secure, and widely adopted smart contract platform
Smart Contracts	Solidity 0.8.21	Security features (e.g., overflow checks) and compatibility with OpenZeppelin
Dev Framework	Hardhat	Fast iteration, local testing, gas profiling
Frontend Framework	ReactJS + Tailwind CSS	Modern, responsive UI with clean design
Wallet Integration	MetaMask + Ethers.js	Industry-standard wallet for dApp interaction
Storage	IPFS + Pinata	Decentralized hosting of ticket metadata
Styling Libraries	Bootstrap 5	UI consistency across components

D. Implementation Details

- Smart Contract Deployment:
  The system has a centralized factory-first deployment to allow for modularization and performance scaling. Contracts for each event are instantiated from the central factory contract. The

deployment is handled by Hardhat scripts with built-in logging and revert handling for testing and validation. The Solidity compiler is configured with the optimization set to 200 runs to cut operational costs on the blockchain, further decreasing gas consumption.

- Frontend Integration:
  The frontend is built with ReactJS, with Ethereum blockchain integration handled via Ethers.js. The contract ABIs automatically load from the build artifacts, so that contract interaction does not need to be constant. Ticket metadata is dynamically loaded from IPFS, with support for several gateways to ensure it is available. The blockchain state of the contract will come into sync with the frontend state in real-time to maintain an up-to-date UI for the end user.
- User Interaction:
  MetaMask integration allows for a decentralized wallet and the signing of transactions. MetaMask integration enables the interface to automatically detect a user's connection to a wallet and prompt them if a network must be compatible. Users will receive live "transaction status" updates, including confirmations or error messages. The user interface is adaptable with a responsive layout. The main user interface is simple to use and supports accessibility for both mobile phones and desktop computers.
- Testing and Validation:
  Considerable unit testing is done to validate the core functions of the smart contract. Non-hard-coded edge cases such as expired events and invalid resale attempts can be tested. We validate visibility to stakeholders and important functions of our front-end components using mock wallets and developer chain/simulated instances of the blockchain, provided the entire system is reliable.

E. Deployment

- Local Development Phase:
  Development and testing take place all on a local Hardhat Ethereum environment. This allows for rapid iterations, predictable behaviours, and great debug opportunities. Contracts are deployed with Hardhat scripted automations, gas profiling, revert control, and log files. The frontend is "fed" local blockchain data with dummy ticket information and mock wallets, all of which attempts to show realistic usage and validate certain features, e.g., ticket minting and resale, and lifecycle state changes.
- Test Environment:
  The Hardhat environment allows for unit and integration testing, which can fully validate both contract logic and UI interaction. Comprehensive test cases include lifecycle state transitions, resale restrictions, royalty enforcing, and time-dependent conditions. The frontend itself has been tested using mock wallets to ensure correct user-facing logic is employed on blockchain events.
- Live deployment:
  In its current state, the system only functions on a local blockchain. Architecturally, it can be set up to work with the Ethereum mainnet later. Prior to any 'live' deployment, a complete audit of the smart contract code will be performed by a professional and respected company to verify security and compliance. The system has upgradeability built in so that it can comply with future standards using either a proxy model or factory replacement process, as well as emergency functionality such as pause models or manual overrides for addressing any critical issues that may arise after deployment.

IV. Results and Analysis

A. Testing Outcomes

Table 3. *Functional Testing Results for the NFT Ticketing System*

Function	Description	Result
Create events (with and without resale option enabled)	As shown in *Figure A.1* and *Figure A.2*, the organizer can create event metadata (e.g. no. of tickets, price, resale option, event name, event duration, and display image)	Passed
Set event stage (can be done by event organizers only)	As shown in *Figure A.3* and *Figure A.4*, organizers must set the stage from 'Prep' to 'Active' to allow ticket purchase.	Passed
Buying a ticket (from the event organizer or resale tickets)	As shown in *Figure A.5*, users can buy tickets from the event organizer or buy tickets that are resold by other users.	Passed
Cancel event	As shown in *Figure A.6*, organizers can cancel events before the event is set to 'Active', while it is still in 'Prep' state.	Passed
Cancel resale	As shown in *Figure A.7*, Users who put their	Passed

	tickets in for resale have the option to cancel the sale.	
View ticket	As shown in *Figure A.8*, Users can see their purchased tickets in the 'My Tickets' section.	Passed
Buy resale ticket	As shown in *Figure A.9*, Users can buy ticket from reseller.	Passed

B. Performance Metrics

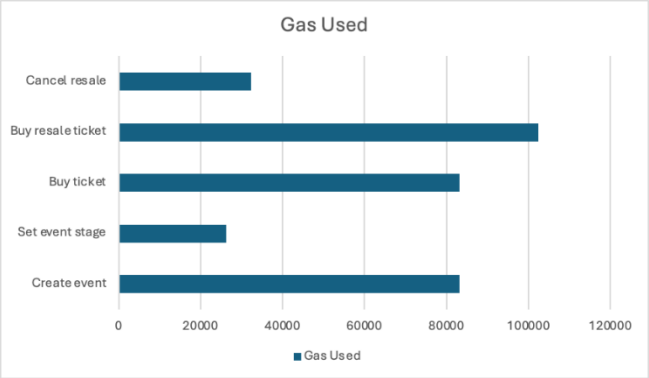

Figure 2. *Gas Used During Function Execution*

The chart represents gas usage for the main functions in the NFT Ticketing smart contract. The functions that consumed the most gas were creating an event and purchasing a ticket, regardless of whether the ticket was purchased as a resale or from the organizer. The reason for the high gas usage is that they were associated with more complex activities such as deploying a contract, minting NFTs, transferring ownership, and updating balances. On the contrary, the event stage and cancel resale functions consumed less gas because they were only updating basic state variables. Overall, gas consumption is related to the number of operations performed and the complexity of each function, along with the changes made to any state information on the blockchain.

C. Key Findings

Table 4. *Key Findings from the Evaluation of the System*

The DApp effectively implements NFT ticketing:	• Tickets are minted as NFTs • Ownership is visible in wallet and My Tickets section in the DApp. • The resale function preserves the same metadata from the original ticket.
Blockchain principles that were effectively proven:	• Transparency: i. Immutable metadata on IPFS. ii. The resale market is entirely transparent. • Security: i. Transactions through MetaMask. • Decentralization: i. No central server for ticketing.

V. Discussion

A. Interpretation of Results

Based of the results analysis, it is evident that this NFT Ticketing System has successfully addressed some of the key challenges identified in the problem statement:

i. Scalability and Performance
The scalability and performance hurdle has been effectively managed through the system's ability to handle vent creation, ticket purchasing, ticket resale, and ticket redemption. Although the performance metrics graph points out the high gas cost used in event creation and ticket purchases, it is important to take note that these figures are relatively modest compared to Ethereum's current average gas cost in the real world. According to a study by Qin Wang, Rujia Li, Qi Wang, and Shiping Chen (2021), at the time of writing, an NFT token costs over USD 60 (equivalently in around 5 × 102wei). To complete a simple NFT trade can run between USD 60 and USD 100 for each transaction [13]. Expensive fees caused by complex operations and high congestion greatly limit its wide adoption [13]. Therefore, given the current Ethereum gas prices and the relatively low gas consumption of this system's operations, it is evident that this NFT Ticketing platform is well-positioned to handle scalability and performance demands.

ii. User Onboarding Difficulties:

The smooth execution of user interactions, such as viewing purchased tickets in the "My Tickets" section, managing resale practices, indicates that the user-friendliness of this system has been accomplished. The system's user interface is intuitive, with proper error or success messages, and accessible, ultimately mitigating the complexities that are often associated with blockchain-based applications.

iii. Regulatory Compliance Risk

Although regulatory compliance is not specifically addressed in the results, the system functionality suggests that it functions within a framework that could adapt to changing legal requirements, given the right steps are taken.

iv. Fraudulent Resale and Cyber-Ticket Touting Regulation

To combat fraudulentt resale activities, this NFT-based ticketing system includes a regulated resale function that enforces a maximum price markup of 10% over the original price. This function ensures that re-sellers cannot excessively inflate prices, thereby maintaining fair access for genuine buyers.

B. Limitations

Some limitations that were encountered include:

i. Connectivity issues

- The integration between front-end and back-end components posed a major challenge in the development of this system. These challenges manifested as communication failures and delays, which impacted the seamlessness of data exchange that was planned.

ii. System's file size

- Due to the use of Hardhat, the system required a major amount of storage space to effectively run all functions. This requirement arose due to the need to simulate a full blockchain environment, including deployment of smart contracts and the maintenance of extensive transaction logs.

iii. Hardhat Node Volatility

- The Hardhat local blockchain resets when restarted, meaning smart contract data (tickets, events) is not persistent unless deployed to a real testnet (Goerli, Sepolia). This made testing resale flows across multiple sessions more difficult.

C. Implications

The successful implementation of this system has brought several meaningful real-world implications. Firstly, by leveraging blockchain technology, this system enhances transparency and security in ticket purchases and resale practices, significantly reducing the possibility of common fraudulent issues. In addition, it also empowers event organizers by allowing them to set appropriate rules for resale and maintain visibility over ticket ownership. This function allows event managers to retain control over revenue streams and event access.

Furthermore, from the user's perspective, the platform has been made to be intuitive and accessible, thereby improving overall user experience and combating the usability issues faced by non-technical users. Therefore, these outcomes suggest that this NFT-based solution can serve as a practical model for future ticketing platforms and will help to set new standards in the event management industry.

VI. Conclusion

A. Summary

This project demonstrates NFT-based ticketing system built with modern web technologies and blockchain technology. The front end is made with HTML, CSS, JavaScript, and ReactJS and provides the users with a responsive, interactive experience that allows browsing for events and purchasing tickets. The back end employs the Hardhat environment to compile, deploy, and test contracts written in Solidity and all the contracts implement the ERC-721 standard to represent each ticket as a single, non-fungible token (NFT), and provides an auditable, trustworthy, and non-duplicable digital equivalent of the real ticket.

B. Significance

This approach addresses some of the major problems with conventional ticketing systems, such as transparency and fake tickets. In terms of business strategy, the current version of the system will expand on NFT and blockchain technology to enhance ticket security, confirm ticket ownership, and enhance ownership transfer transparency, for instance, through transactions. Additionally, by automating the creation of tickets and the transfer of ticket ownership, smart contracts eliminate the need for a third party and reduce transaction fees that were previously excessive due to the need for extra middlemen. As a result, the project contributes to the larger body of prior art that demonstrates NFT's ability to address real-world issues such as event ticket fraud.

C. Future Work

There are areas for further improvements to the system. One potential area is gas fees. As mentioned earlier, the ecosystem can cost a very high gas fee, especially on Ethereum. The system can employ certain mechanisms to keep costs down as much as possible, such as reducing the number of stored items and conserving over storage by limiting functions in the contract. It was also mentioned with off-chain logic, using lazy contracts, we can reduce the number of computations left on-chain, thereby reducing overall gas fees [14]. In the future, we may consider creating a mobile app or using a Layer-2 (L2) blockchain to further reduce gas fees and extend transaction times.

VII. Acknowledgment

We express our gratitude to our lecturer, Dr. Uzma Jafar, for her continuous support and guidance during this project. We also acknowledge the researchers whose comprehensive work formed the foundation of our references.

VIII. References

[1] H. Lawal, NFT Ticketing Application, GitHub repository, 2023. [Online]. Available: https://github.com/habiblawal1/NFT-Ticket-Application

[2] Egghead.io, Build a Practical NFT Ticketing Service with Solidity and React, 2023. [Online]. Available: https://egghead.io/courses/build-an-nft-based-ticketing-system-1a2f387c

[3] J. Xu, S. Wang, and L. Chen, "Scalability challenges in blockchain systems," *IEEE Trans. Eng. Manage.*, vol. 68, no. 4, pp. 1234–1245, 2021. [Online]. Available: http://eprints.utm.my/105394/1/FizaAbdulRahim2023_ASystematicReviewonBlockchainScalability.pdf

[4] Y. Zhao, H. Li, and Q. Zhang, "User adoption barriers in decentralized applications," *IEEE Access*, vol. 10, pp. 56789–56800, 2022. [Online]. Available: https://www.emerald.com/insight/content/doi/10.1108/scm-02-2024-0138/full/html

[5] C. Catalini and J. S. Gans, "Legal challenges in blockchain adoption," *MIT Sloan Research Paper*, no. 5787-20, 2020. [Online]. Available: https://www.nber.org/system/files/working_papers/w22952/w22952.pdf

[6] J. M. Zhang, T. Li, and S. K. Lee, "Mitigating Cyber-Ticket Touting in Blockchain-Based Event Ticketing Systems," *Journal of Marketing*, vol. 87, no. 3, pp. 45–60, Jul. 2023. [Online]. Available: https://journals.sagepub.com/doi/full/10.1177/00220183221107544

[7] A. F. van den Bosch, "GUTS Tickets and the GET Protocol: A Blockchain-based Event Ticketing System," *Blockchain Applications Journal*, vol. 12, no. 1, pp. 33–47, 2022.

[8] SeatlabNFT, "NFT Ticketing Ecosystem," 2023. [Online]. Available: https://www.seatlabnft.com

[9] S. Schmid and M. Golling, "DeTi: A Decentralized Ticketing Management Platform," *Journal of Network and Systems Management*, vol. 30, no. 3, pp. 62, 2022. doi: 10.1007/s10922-022-09675-3

[10] A. Verma and D. Patel, "NFT-Based Event Ticketing Using Smart Contracts and Web3 Tools," *International Journal of Research Publication and Reviews (IJRPR)*, vol. 5, no. 3, pp. 279–285, 2024. [Online]. Available: https://ijrpr.com/uploads/V5ISSUE3/IJRPR23766.pdf

[11] N. Akhtar, R. Hussain, and M. Tariq, "akaTick: A Secure Hybrid NFT-Based Mobile e-Ticketing System," in *Proc. IEEE MetaCom 2023*, pp. 686–690. doi: 10.1109/MetaCom57879.2023.00099

[12] R. Sharma and K. Mehta, "Decentralized Ticketing System Using Ethereum Smart Contracts," *International Research Journal of Engineering and Technology (IRJET)*, vol. 11, no. 7, 2024. [Online]. Available: https://www.irjet.net/archives/V11/i7/IRJET-V11I711.pdf

[13] Q. Wang, R. Li, Q. Wang, and S. Chen, "Non-Fungible Token (NFT): Overview, Evaluation, Opportunities and Challenges," arXiv, 2021. [Online]. Available: https://arxiv.org/pdf/2105.07447

[14] A. S. Zakariyya and S. Marchesini, "Lazy Smart Contracts: Offloading and Optimizing On-Chain Logic," arXiv preprint arXiv:2309.11317, Sep. 2023. [Online]. Available: https://arxiv.org/abs/2309.11317

IX. Appendix A

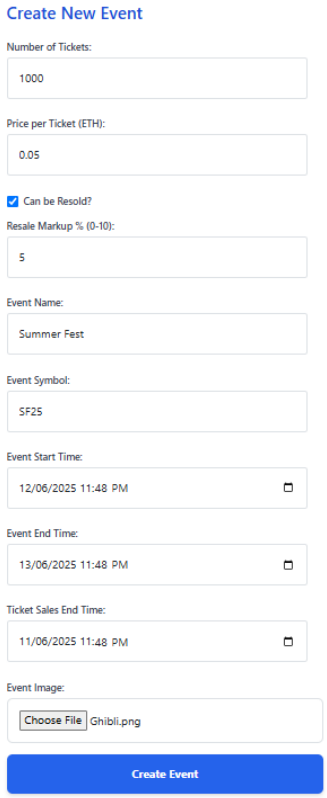

Figure A. 1. *Creation of Event and Input Values.*

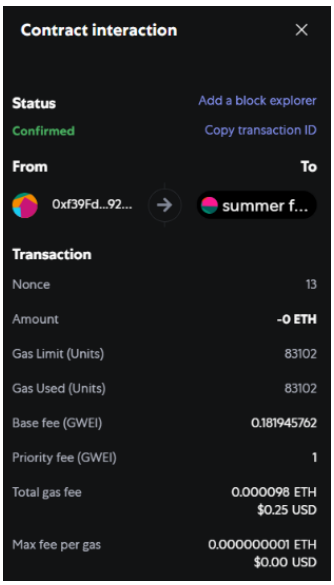

Figure A. 2. *MetaMask Confirmation of Event Creation.*

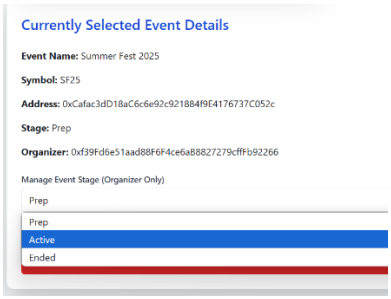

Figure A. 3. *Event Organizers changing the Event Stage to "Active" to enable ticket purchase.*

Figure A. 4. *MetaMask Confirmation of Change in Stage State.*

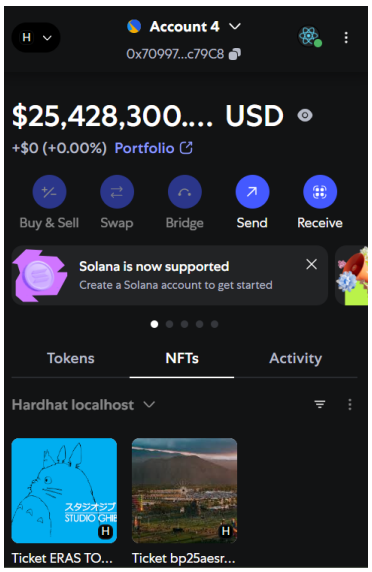

Figure A. 5. *MetaMask Confirmation of Ticket Purchases in the form of NFTs.*

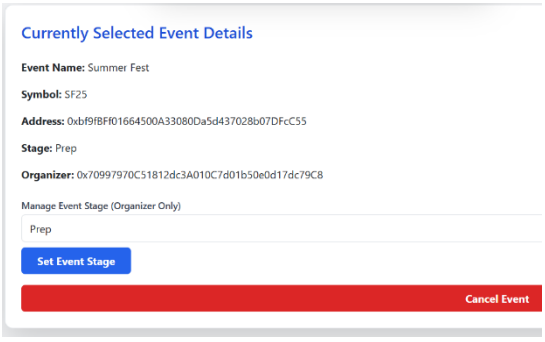

Figure A. 6. *Event Cancellation while in 'Prep' Stage.*

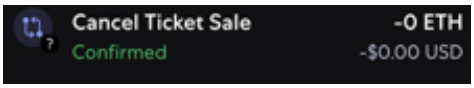

Figure A. 7. *MetaMask Confirmation of Users' Resale Ticket Cancellation*

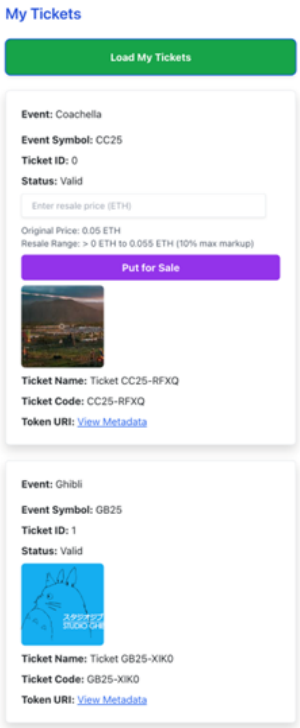

Figure A. 8. *'MyTickets' Displays the User's Purchased Tickets.*

Figure A. 9. *'MyTickets' Displays the users reselling tickets.*