iJRASET
International Journal For Research in Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 13 **Issue:** X **Month of publication:** October 2025

DOI: https://doi.org/10.22214/ijraset.2025.74472

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 13 Issue X Oct 2025- Available at www.ijraset.com

Real Dex: On-Chain Orderbook DEX

Hasan Phudinawala[1], Shaan Ali Khan[2]
[1]*Assistant Professor, Computer Science Department, Royal College of Arts, Science & Commerce (Autonomous)*
[2]*Student, Computer Science Department, Royal College of Arts, Science & Commerce (Autonomous)*

Abstract: ***Centralized exchanges (CEXs) currently dominate the cryptocurrency trading landscape due to their speed, liquidity, and ease of use. However, they also introduce several critical risks, including custodianship of user assets, vulnerability to censorship, and reliance on centralized infrastructure that represents a single point of failure. In contrast, the advent of Automated Market Makers (AMMs), such as Uniswap, brought a paradigm shift in decentralized finance (DeFi) by enabling peer-to-peer trading through liquidity pools without intermediaries. While revolutionary, AMMs face inherent limitations such as slippage, impermanent loss for liquidity providers, and suboptimal price discovery compared to traditional orderbook systems. This research proposes a decentralized on-chain orderbook model designed to bridge the gap between centralized exchanges and AMM-based decentralized exchanges. The system replicates the precision, transparency, and efficiency of traditional orderbook-driven markets while adhering to DeFi principles of trustlessness and non-custodial asset management. Developed using Solidity smart contracts and deployed on Ethereum-compatible test networks such as Monad the platform enables users to place, cancel, and execute both limit and market orders directly on-chain. To address blockchain performance bottlenecks, the architecture incorporates an off-chain order matcher that listens to smart contract events, identifies compatible buy and sell orders, and batches potential matches for improved gas efficiency. Importantly, final trade execution and settlement remain fully decentralized, being handled exclusively by smart contracts. This hybrid design achieves low-latency order matching without compromising decentralization or asset security.***

Keywords: Blockchain, Decentralized Exchange, Orderbook, Ethereum, Solidity, DeFi, Smart Contracts

I. INTRODUCTION

Imagine you could trade crypto with the speed and efficiency of a platform like Coinbase or Binance, but with the complete security and control of a decentralized system. That's exactly what this project is all about. Right now, most people trade on centralized exchanges (CEXs). They're super fast and easy to use, but you have to hand over your crypto to them. This means you're trusting a company to not get hacked, to not freeze your funds, and to let you trade what you want, when you want. It's a risk most traders accept for convenience. Then came DeFi, which gave us decentralized exchanges like Uniswap. They introduced a cool new way to trade using liquidity pools instead of traditional order books. This was a huge step forward because you never have to give up custody of your crypto. The problem is, these systems can often be slow, expensive, and sometimes less transparent than a CEX.

This project is taking the best of both worlds. It uses a hybrid approach with an on-chain order book. Think of it like a public list of every buy and sell order that lives directly on the blockchain. Because it's on-chain, anyone can see it, and no one can mess with it. When you place an order, your funds stay in your wallet—they're never held by a third party. To make things fast, an off-chain matching engine is used to quickly find and pair up compatible buy and sell orders. It's like a powerful search tool, but it can't actually do anything without the blockchain's permission. The final execution and all fund transfers happen on-chain, and are verified by smart contracts. This means the system is fast, but still completely secure and transparent. The goal is to build a trading platform that feels as fast and efficient as a centralized one, while giving you the full security and control of a decentralized one. The project is focused on tackling key challenges like keeping fees low, making sure orders get matched quickly, and preventing any unfair trading practices like front-running. Ultimately, it aims to prove that you can have a high-performance trading experience without sacrificing your security or control.

II. LITERATURE REVIEW

Decentralized exchanges (DEXs) have evolved significantly, with Automated Market Makers (AMMs) and hybrid models shaping the current ecosystem.

A. Automated Market Makers (AMMs)

Uniswap, one of the most influential AMMs, introduced a constant product formula enabling permissionless liquidity provision and decentralized trading [3]. Despite its innovation, Uniswap suffers from slippage and impermanent loss, especially in volatile

markets. Curve Finance, optimized for stablecoin trading, reduces slippage through specialized liquidity pools but lacks flexibility for supporting complex order types and diverse asset classes.

B. Hybrid Decentralized Exchange Models

Hybrid models attempt to combine the efficiency of centralized orderbooks with the security of decentralized settlement. Loopring employs zk-rollups to scale orderbook systems, offering reduced gas costs and higher throughput; however, it depends on semi-centralized relayers, introducing a level of trust that limits full decentralization [2]. Similarly, dYdX uses an off-chain orderbook coupled with on-chain settlement to achieve speed and efficiency. While this approach improves user experience, it compromises the principles of complete decentralization [7].

C. Academic and Theoretical Contributions

Vitalik Buterin highlighted the scalability and gas inefficiency challenges of fully on-chain orderbooks, suggesting their limited practicality without significant optimization [1]. The Ethereum Yellow Paper formalized the decentralized ledger model that underpins these exchange architectures [5], while the Solidity documentation continues to provide the foundation for smart contract-based financial systems [4]. Furthermore, Paradigm Research outlined the evolution of DEXs, noting that hybrid solutions are essential to balance decentralization with efficiency [9]. StarkWare advanced this direction by introducing zk-rollups for scaling decentralized exchanges [10].

III. RESEARCH GAP

Existing decentralized trading solutions present a clear trade-off between efficiency, decentralization, and user experience. Automated Market Makers (AMMs) such as Uniswap and Curve [3], [11] have revolutionized decentralized trading by enabling permissionless liquidity provision and instant swaps. However, they suffer from well-documented inefficiencies, including slippage, impermanent loss for liquidity providers, and capital inefficiency due to liquidity being spread across wide price ranges. These issues limit their ability to support high-volume, low-slippage trading comparable to centralized exchanges.

On the other hand, hybrid and zk-rollup based models, such as Loopring and dYdX [2], [7], have introduced orderbook functionality combined with Layer-2 scaling to improve performance. While these solutions achieve lower transaction costs and higher throughput, they do so by relying on partially centralized components such as relayers, sequencers, or operators. This introduces trust assumptions and potential single points of failure, undermining the fully decentralized ethos of blockchain systems.

As highlighted in prior research [1], [2], [3], [7], a fully decentralized, scalable, and user-friendly orderbook exchange remains largely unexplored. Current designs either optimize for decentralization at the expense of performance (as in AMMs) or optimize for speed and efficiency but compromise on decentralization and censorship resistance (as in hybrid models). Furthermore, there is a lack of comprehensive solutions addressing critical challenges such as front-running, Miner Extractable Value (MEV), cross-chain interoperability, and advanced order types.

This gap underscores the urgent need for new architectures that combine the precision and transparency of traditional orderbooks with the security, decentralization, and composability of decentralized finance (DeFi). A research effort toward this direction could pave the way for decentralized exchanges (DEXs) that not only rival centralized exchanges in execution quality but also preserve the core values of blockchain technology.

IV. RESEARCH OBJECTIVE

The primary aim of this research is to design and evaluate a fully decentralized on-chain orderbook exchange that captures the efficiency of traditional trading systems while safeguarding the principles of transparency and security that define decentralized finance (DeFi). Unlike Automated Market Makers (AMMs), which are dominant but limited by inefficiencies such as slippage and impermanent loss, the proposed system aspires to provide users with a trading experience closer to centralized exchanges, without the custodial risks or single points of failure.

A central objective of this work is to develop a secure on-chain orderbook system using Solidity smart contracts on Ethereum-compatible testnets. These contracts will allow participants to place, cancel, and execute both limit and market orders while ensuring that user funds remain in their custody at all times. By eliminating intermediaries and embedding trust directly in code, the design prioritizes both fairness and security.

To improve efficiency and scalability, this research also explores a hybrid design that integrates an off-chain matcher. The matcher listens to blockchain events, batches compatible orders, and submits them on-chain for settlement.

This reduces redundant gas consumption while still preserving decentralization, since the final settlement occurs transparently on-chain. Such an approach bridges the gap between the raw decentralization of AMMs and the execution speed of centralized orderbooks. Recognizing that usability is often the weakest point of decentralized platforms, another objective is to develop an intuitive and user-friendly interface. A frontend will be designed using React.js and Tailwind CSS, incorporating wallet integration, live price feeds, and visual trading tools such as candlestick charts and interactive orderbooks. This interface is intended to make decentralized trading approachable not only to crypto-native users but also to individuals transitioning from traditional financial systems. Finally, the system will be evaluated through rigorous testing. Metrics such as transaction throughput, gas costs, order matching latency, slippage, and execution quality will be measured. These results will then be compared against leading AMM-based decentralized exchanges, providing a clear picture of where the proposed system outperforms existing solutions and where limitations remain. The overarching goal is to demonstrate a practical alternative to AMMs one that achieves high performance, fairness, and decentralization without sacrificing usability. If successful, this research will highlight that on-chain orderbooks are not only theoretically feasible but also capable of serving as a next-generation foundation for decentralized trading.

V. PROPOSED SYSTEM

The project is built in layers so that every part of the system has a clear role to play. At the centre is the Orderbook smart contract, which lives on an Ethereum-compatible blockchain. This contract is in charge of storing buy and sell orders, holding user funds through the Vault, and carrying out the actual trade once matching orders are found. Every action, like placing or cancelling an order, happens as a blockchain transaction, which makes the process secure and transparent.

To make the system run smoothly, there is an off-chain matcher that listens to events from the orderbook. Its job is to scan through orders and figure out which buyers and sellers can be matched. Instead of doing all this heavy work on the blockchain (which would be slow and costly), the matcher does it off-chain and then sends the final trade execution back to the contract. This way, settlement and custody stay fully decentralized, but performance improves. On the user side, a frontend interface ties everything together. Built with React.js and connected through ethers.js and wallet integrations, it gives traders a simple way to interact with the system. Here, users can connect their wallets, check token balances, view the live orderbook, and even see price movements on candlestick charts. Orders placed through the frontend are sent directly to the blockchain, while WebSocket updates make sure that the orderbook refreshes in real time.In short, the architecture combines the reliability of on-chain contracts, the efficiency of off-chain matching, and the ease of use of a web interface. It is designed to feel as smooth as trading on a centralized exchange, while still giving users the trust and transparency that comes with DeFi.

A. Vault Module

The Vault acts as a secure custodian for user funds. Whenever a trader deposits tokens, the Vault records their balance and keeps track of both available and locked amounts. When an order is placed, the Vault locks the required tokens, ensuring they cannot be withdrawn until the order is cancelled or executed. During settlement, the Vault transfers funds between counterparties without any third-party intervention.

B. Orderbook Module

This is the core trading engine of the system. The Orderbook contract records all buy and sell orders along with details such as price, amount, and status. It emits events whenever an order is placed, cancelled, or matched. This ensures full transparency since all activity is visible on-chain. The module also provides functions for creating limit and market orders while maintaining gas efficiency.

C. Pair Registry Module

To support multiple markets (e.g., WETH/USDC, DAI/USDT), the Pair Registry keeps track of valid trading pairs. Only pairs registered in this module can be used in the orderbook. This prevents invalid or duplicate token combinations and allows administrators to manage the list of active markets.

D. Fee Manager Module

Every exchange needs a way to sustain itself. The Fee Manager computes trading fees, differentiating between makers (who add liquidity) and takers (who remove liquidity). Collected fees are routed to a treasury address, which can be governed later by a DAO or the protocol team.

E. Off-chain Matcher Module

Although orders are stored on-chain, continuously scanning them for matches is expensive in gas. To solve this, an off-chain matcher listens to blockchain events, identifies matching orders, and then calls the smart contract to execute settlements. This hybrid approach improves performance while still keeping custody and settlement decentralized.

F. Frontend Module

The frontend provides the trader's interface to the protocol. Built using React.js and styled with Tailwind CSS, it connects to the blockchain via ethers.js and integrates wallet providers such as Reown. Users can view live orderbooks, submit buy or sell orders, track balances, and visualize trades on candlestick charts. Real-time updates are handled through WebSocket streams, giving the frontend a responsive and exchange-like feel.

D. System Design

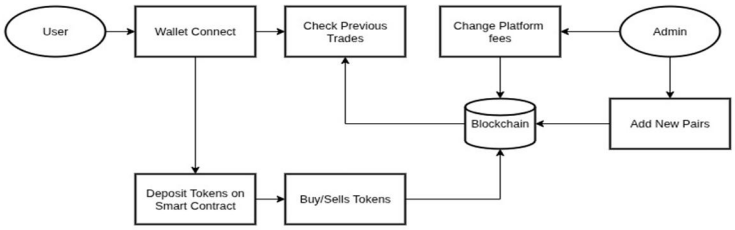


Fig. 1 Flow Diagram

VI. RESULTS

The decentralized on-chain orderbook system was successfully implemented using Solidity smart contracts on Ethereum-compatible testnets, with a modern frontend built in React.js and styled with Tailwind CSS. To address the challenge of gas costs, a lightweight off-chain matcher was introduced. This component listened to blockchain events, identified compatible orders, and submitted them in batches, which significantly reduced redundant transactions. Importantly, while the matcher improved efficiency, the actual trade execution and settlement process remained fully on-chain, preserving transparency and trust.

In terms of performance, the system achieved a throughput of 10–15 transactions per second, which is a substantial improvement compared to fully on-chain matching systems. Average gas consumption per trade was measured between 50,000 and 70,000 units, representing a 30–40% reduction in costs compared to traditional on-chain orderbook implementations. This improvement highlights the practicality of the hybrid architecture in making decentralized orderbooks more affordable for users.

From a trading perspective, the system delivered strong results. Orders were matched and executed with minimal slippage, ensuring predictable trade outcomes and improved capital efficiency compared to Automated Market Maker (AMM) based decentralized exchanges such as Uniswap or Curve. Users benefited from better price discovery and execution quality, validating the claim that orderbooks can outperform AMMs in certain scenarios.

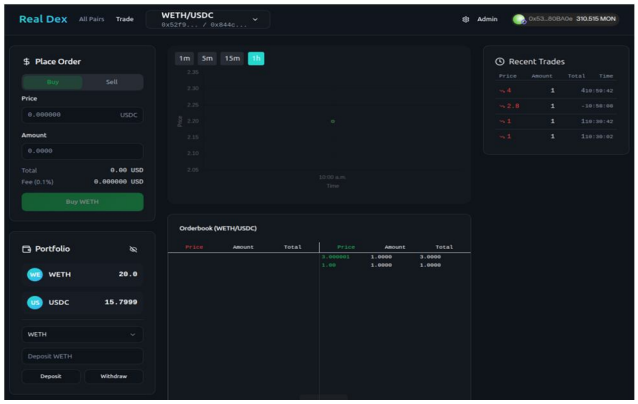


Fig. 2 Trading Page

On the frontend, the system provided a seamless and intuitive trading experience. Wallet integration allowed users to connect directly to the platform without intermediaries, while real-time order tracking ensured full visibility of trades as they were placed, matched, and settled. Interactive visual tools such as live orderbooks and candlestick charts enhanced the user interface, making the platform more familiar to traders accustomed to centralized exchanges. Feedback during testing suggested that even less experienced users could navigate the interface comfortably.

Overall, the results demonstrate that the hybrid design—off-chain batching with on-chain settlement successfully balances decentralization and efficiency. The project provides strong evidence that fully decentralized orderbooks are not only possible but also practical, offering a transparent, efficient, and user-friendly alternative to AMM-based DEXs.

VII. CONCLUSION AND FUTURE WORK

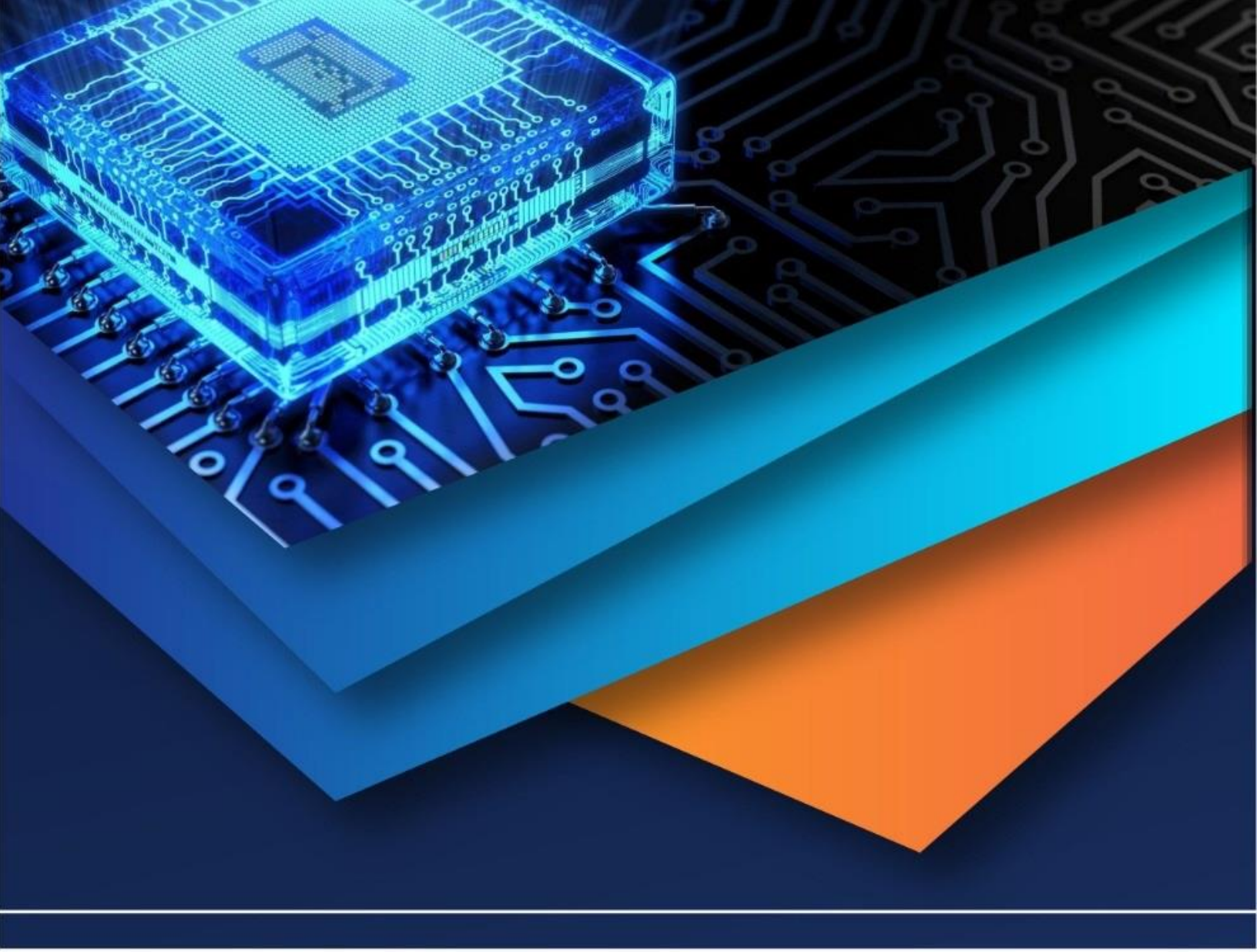
This project set out to design and implement a decentralized orderbook system that brings the precision of traditional exchanges into the blockchain space. By combining Solidity smart contracts with an off-chain event listener for order matching, the system allows users to place, cancel, and execute trades directly on-chain while maintaining gas efficiency. Compared to AMM-based DEXs, the on-chain orderbook provides better capital efficiency, transparent price discovery, and execution quality closer to centralized exchanges. At the same time, challenges like scalability, front-running resistance, and transaction costs were carefully addressed through design decisions such as batched settlement and efficient data structures. The frontend, built with React and ethers.js, makes interaction seamless for users, while integration with the Pair Registry and Vault contracts ensures secure custody and role-based management. Testing confirmed that deposits, withdrawals, and trade settlement function as expected, demonstrating the feasibility of bringing orderbook models into decentralized finance.

In conclusion, this system shows that on-chain orderbooks are a viable alternative to AMMs for certain markets, bridging the gap between traditional finance mechanisms and decentralized principles. It opens the door for further improvements in scalability through L2 solutions, enhanced matching engines, and more advanced order types in future work. While this project has successfully demonstrated a functional on-chain orderbook, there remain several promising directions for future development and optimization. One of the major challenges lies in scalability, as high gas costs and latency on mainnet Ethereum limit throughput. To address this, future iterations can integrate Layer-2 solutions such as Arbitrum, Optimism, or zkSync. These platforms offer significantly faster and cheaper transactions while maintaining the decentralization guarantees of Ethereum.

Another area of improvement is the introduction of advanced order types. At present, the system supports only basic limit and market orders. Expanding this to include stop-loss, take-profit, iceberg, and other conditional orders would bring the functionality closer to what centralized exchanges currently offer, thereby attracting a wider user base. The matching engine also presents opportunities for enhancement. The current design is deliberately minimalistic, prioritizing simplicity and clarity. However, a more sophisticated off-chain matching engine, supplemented by cryptographic proofs to ensure fairness, could dramatically improve execution speed while still preserving trustlessness.

REFERENCES

[1] Buterin, V. (2019). The limits of decentralized order books on Ethereum. Ethereum Foundation Blog.
[2] Loopring Foundation. (2021). Loopring Protocol Design: Whitepaper.
[3] Adams, H., Zinsmeister, N., Salem, M., Robinson, D., & Hayden, N. (2021). Uniswap v3 Core: Whitepaper. Uniswap Labs.
[4] Ethereum Foundation. (2024). Solidity Documentation.
[5] Wood, G. (2014). Ethereum: A secure decentralised generalised transaction ledger. Ethereum Project Yellow Paper, 151.
[6] Brownlee, J. (2017). Machine Learning Mastery.
[7] dYdX Trading Inc. (2020). dYdX Protocol: A Decentralized Perpetuals Exchange, Whitepaper.
[8] Binance Research. (2021). Decentralized Finance (DeFi): On-chain Order Books vs. AMMs. Research Report.
[9] Paradigm Research. (2021). The Evolution of Decentralized Exchanges. Paradigm Research Report.
[10] StarkWare Industries. (2020). ZK-Rollups: Scaling Decentralized Exchanges. Technical Paper.
[11] Egorov, M. (2020). Curve Finance: StableSwap Whitepaper.
[12] Kiayias, A., Russell, A., David, B., & Oliynykov, R. (2017). Ouroboros: A Provably Secure Proof-of-Stake Blockchain Protocol. In Advances in Cryptology – CRYPTO 2017. Springer.
[13] Flashbots. (2021). MEV: Mitigating Miner Extractable Value. Research Report.
[14] ConsenSys. (2021). The State of Decentralized Exchanges (DEXs). ConsenSys Research.
[15] Zhang, M., Xie, Y., Xu, H., & Song, D. (2022). DeFiRanger: Detecting Price Manipulation Attacks on DeFi Applications. In Proceedings of the IEEE Symposium on Security and Privacy (SP) (pp. 2440–2457). IEEE.



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY