IJCRT.ORG ISSN : 2320-2882

INTERNATIONAL JOURNAL OF CREATIVE RESEARCH THOUGHTS (IJCRT)

An International Open Access, Peer-reviewed, Refereed Journal

Decentralized Nft Marketplace Using Ethereum And Ipfs With Ai-Powered Price Prediction

Prathmesh Hirnaik[1], Ashwin Rajendran[2], Deb Pattanayak[3], Dhanush Hegde[4]
[1]Student, SIES Graduate School of Technology, Nerul, Maharashtra, India
[2]Student, SIES Graduate School of Technology, Nerul, Maharashtra, India
[3]Student, SIES Graduate School of Technology, Nerul, Maharashtra, India
[4]Student, SIES Graduate School of Technology, Nerul, Maharashtra, India

Abstract: This project introduces a decentralized NFT marketplace on the Ethereum blockchain, with IPFS integration for secure and decentralized metadata storage. It enables transparent trading of unique digital assets like digital art and collectibles via ERC-721 smart contracts, removing the need for intermediaries. A key innovation is the integration of an AI-based price prediction system using the RandomForestRegressor algorithm. Trained on historical NFT data, the model considers factors like collection traits, creator details, rarity, market trends, and time-based metrics to provide price estimates. The platform includes a Next.js frontend, interacts with the blockchain through ethers.js and Web3 wallets, and features a backend for AI inference and business logic. Model evaluation shows strong performance ($R^2 \approx 0.84$, RMSE ≈ 0.32 ETH), suggesting that machine learning can support users in making informed NFT pricing decisions. Overall, the system combines decentralization, secure asset handling, and intelligent analytics to improve NFT marketplace efficiency and user experience.

Index Terms - Decentralized NFT marketplace, AI Price Prediction, Ethereum blockchain, IPFS, Smart contracts, RandomForestRegressor, Machine Learning, ERC-721, NFT Valuation, Decentralized Application (DApp).

I. Introduction

1.1 Background: The Rise of NFTs

Non-Fungible Tokens (NFTs) have revolutionized digital ownership by allowing the tokenization of unique assets like art, music, and virtual collectibles on blockchain networks—primarily Ethereum. Unlike cryptocurrencies, NFTs are non-interchangeable and verifiable on-chain, enabling creators to monetize their work and ensure provenance without traditional intermediaries. The rapid growth of the NFT market reflects increasing demand for digital asset ownership.

1.2 Challenges in the NFT Market

Despite their popularity, NFTs come with several challenges:

- **Price Volatility**: NFT values fluctuate due to speculation and fast-changing trends.
- **Information Asymmetry**: Access to key valuation factors (rarity, creator reputation, historical sales) is uneven.
- **Complex Valuation**: Pricing is influenced by subjective and dynamic factors like artistic value, trait rarity, and market sentiment.

- **Scalability and Gas Fees**: Ethereum's congestion can lead to high transaction costs and delays.
- **Centralized Metadata**: Many platforms store NFT data on centralized servers, risking data loss or censorship.

1.3 Proposed Solution and Contributions

This paper presents a decentralized NFT marketplace that integrates blockchain, decentralized storage, and AI-based analytics to tackle these issues. Our main contributions are:

1. **Decentralized Framework**: Marketplace built using Ethereum ERC-721 smart contracts and IPFS for distributed metadata storage.
2. **AI Price Estimation**: Integration of a RandomForestRegressor model to predict NFT prices using historical data, creator info, rarity, and market trends.
3. **User-Friendly Interface**: A frontend developed with Next.js and Web3 wallet support (e.g., MetaMask) for easy access and trading.
4. **Model Evaluation**: Performance of the AI model assessed through R^2 and RMSE metrics, showcasing its potential in supporting informed decision-making.

2. Literature Review

This section summarizes prior work relevant to NFT marketplaces and the use of AI in price prediction, forming the basis for our proposed system.

2.1 NFT Marketplace Architectures

NFT platforms have evolved from centralized models to decentralized ones for improved trust and data integrity.

- **Storage Models**: Centralized metadata storage poses risks of loss or censorship. As shown in works like *Artcart* [3], using IPFS enhances data security and permanence through decentralized storage. Our platform adopts this model to ensure resilience.
- **Smart Contract Standards**: Standards like ERC-721 (for unique NFTs) and ERC-1155 (for batchable tokens) are widely used, with OpenZeppelin offering secure implementations. We use ERC-721 and base our smart contracts on these libraries for security and compatibility.
- **User Experience**: Frontends built with frameworks like Next.js and Tailwind CSS, as seen in [1], improve usability. Wallet integration via MetaMask enhances accessibility. Our marketplace mirrors this approach using React (Next.js) and web3modal. Additionally, multi-chain support, explored in [2], is recognized as a future enhancement to reduce costs and improve scalability.

2.2 NFT Valuation and Price Influences

Understanding NFT pricing dynamics is critical for both platforms and AI models.

- **Rarity & Provenance**: As Chohan [7] notes, scarcity and verified ownership drive NFT value—core principles built into ERC-721.
- **Market Dependencies**: Ante [6] links NFT pricing to broader crypto markets, suggesting price movements can reflect external financial trends. Our model can be extended to account for such influences.
- **Trait-Based Valuation**: Rarity and the number of traits significantly impact prices. While aesthetics are subjective, features like trait_rarity and num_traits help quantify this for AI-based models.

2.3 AI in Financial and NFT Price Prediction

AI has a long-standing role in price prediction, with growing application in the NFT domain.

- **Algorithms**: Techniques like linear regression, SVM, and ensemble models are widely used. We chose Random Forest for its robustness and suitability for non-linear, tabular data without complex tuning.

- **Feature Engineering**: Predictive accuracy depends on well-designed features. Our model combines on-chain data (e.g., transaction history), metadata (e.g., traits), and market activity (likes, bids), along with time-based features.

- **Emerging NFT-Specific AI Models**: Though still developing, early tools and research are beginning to explore NFT valuation using AI. Our contribution is a fully integrated, real-world implementation within a working NFT marketplace.

3. System Architecture and Design

System Architecture

The proposed system is built as a decentralized application (DApp) with a modular architecture for enhanced security, usability, and scalability. Figure 3.1 presents an updated architectural diagram, including the AI component and data flow across various modules.

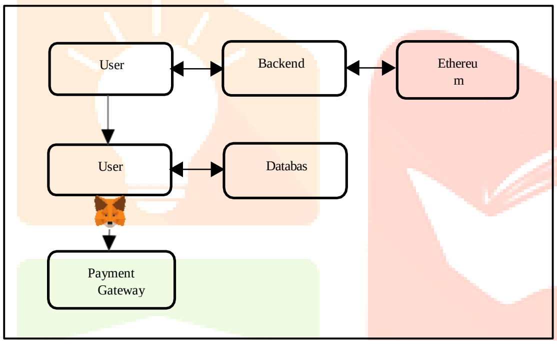

Figure 3.1 Proposed System

3.1 System Components

The **frontend** is developed using Next.js for optimized performance through server-side rendering. It provides users with an intuitive interface to connect their wallets (e.g., MetaMask via web3modal), browse and filter NFTs, view detailed metadata and AI-predicted prices, mint new tokens, list NFTs for sale, and manage their collections. Libraries like framer-motion, react-dropzone, and react-icons are used for enhanced user experience. The frontend interacts with both the backend server and the Ethereum blockchain using ethers.js for executing transactions.

The **backend server**, likely built using Node.js and Express.js, acts as a middleware that serves the frontend, handles API requests, manages user sessions, fetches marketplace data, and communicates with the AI price prediction module. It may also interact with an off-chain database to cache or retrieve NFT metadata, listing info, and analytics data.

The **AI price prediction module**, implemented in Python using scikit-learn, uses a trained RandomForestRegressor model to estimate NFT prices. It loads saved encoders and the model (joblib files), preprocesses input features (e.g., rarity, traits), and returns a predicted price in ETH when called by the backend.

The **Ethereum blockchain** serves as the decentralized trust layer, recording all NFT ownership and transactions. Smart contracts, written in Solidity and deployed using Hardhat, handle core operations like minting (mintNFT), listing (listNFT), and purchasing (buyNFT) of NFTs. These contracts are based on OpenZeppelin's secure ERC-721 implementation and may include additional logic for auctions or royalties.

IPFS (InterPlanetary File System) is used to store NFT metadata and media files in a decentralized way. During minting, metadata is converted into a JSON file, uploaded to IPFS (using services like Pinata), and the resulting CID is stored on-chain as the tokenURI.

User wallets, such as MetaMask, manage private keys and allow users to interact with the blockchain. Integrated via web3modal, these wallets enable users to approve transactions and store their crypto assets securely.

An **off-chain database** (e.g., MongoDB or PostgreSQL) stores non-critical data like user preferences, cached listings, and transaction history. This data supports frontend rendering, analytics, and may also assist in feature generation for the AI model.

3.2 Example Data Flow: Price Prediction

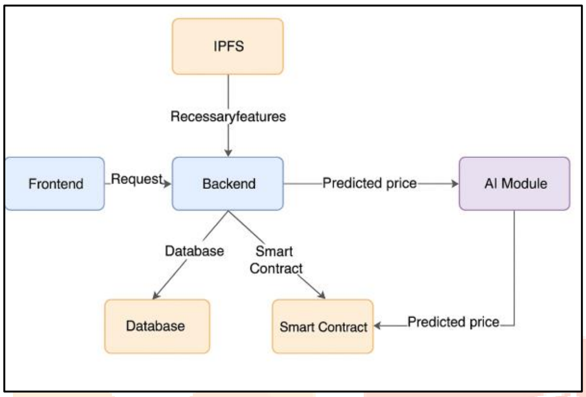

Figure 3.2 Architecture Diagram

When a user opens an NFT detail page, the frontend sends a request to the backend with the NFT's ID. The backend collects relevant features by fetching metadata from IPFS, retrieving historical data from the database, and optionally querying the smart contract for real-time information like current owner or listing status. These features are formatted and sent to the AI module, which processes the input using its encoders and pre-trained model to generate a price prediction. This predicted value is then returned to the backend and displayed on the frontend alongside the NFT's other details. Figure 3.2 illustrates this end-to-end data flow and interaction between components during the prediction process.

4. Methodology and Implementation Details

4.1Marketplace Implementation

The NFT marketplace was developed using a blend of blockchain technologies, web frameworks, and decentralized storage systems. At its core is the NFTMarketplace.sol smart contract written in Solidity (e.g., version ^0.8.0), which follows the ERC-721 standard and likely uses OpenZeppelin libraries for token and access control functionalities. The contract supports minting (mintNFT), listing (listNFT), and buying NFTs (buyNFT), handling ownership transfers and payments securely. Development and deployment were managed using the Hardhat framework, with tools for testing and compiling.
The frontend was created using Next.js for fast performance and server-side rendering. The ethers.js library enables blockchain interactions, while web3modal simplifies wallet connections (e.g., MetaMask). State is managed using React features or libraries like Zustand or Redux, and the UI is enhanced with libraries like react-dropzone, react-icons, and framer-motion.
For decentralized storage, the platform integrates IPFS. When minting an NFT, media files are uploaded to IPFS, followed by metadata JSON creation (e.g., name, image CID, traits), which is then also uploaded. The resulting metadata CID becomes the tokenURI. During retrieval, the frontend fetches this URI from the smart contract and retrieves the metadata using an IPFS gateway.

4.2 AI Price Predictor Implementation

The price prediction system was developed in Python using the scikit-learn library. The model is trained on synthetic NFT sales data, allowing controlled experimentation with various features like rarity scores, number of traits, likes, bids, collection floor prices, and time-based attributes. The target variable is the NFT's sale price in ETH.

Preprocessing involved encoding categorical features (e.g., collection, creator, blockchain) using LabelEncoder, calculating time-related features like days_since_first_sale, and managing missing data. A RandomForestRegressor model was chosen for its robustness, ability to handle non-linear relationships, and good performance on tabular data. Key parameters like n_estimators=150 and random_state=42 were set, although no extensive hyperparameter tuning was performed.

The dataset was split 80-20 for training and testing, with performance evaluated using R^2 and RMSE metrics. The model and encoders were serialized using joblib for reuse in real-time prediction without retraining. The architecture, shown illustrates how, upon viewing an NFT, the frontend requests metadata and historical data from the backend, which is processed by the AI module to return a predicted price alongside the listing.

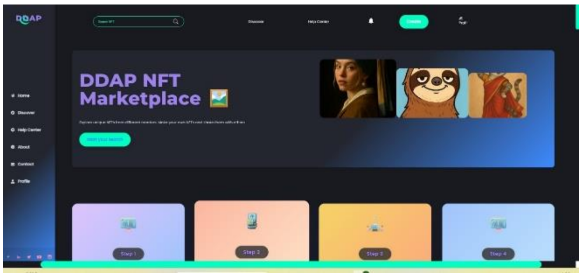

Figure 4.1 Homepage

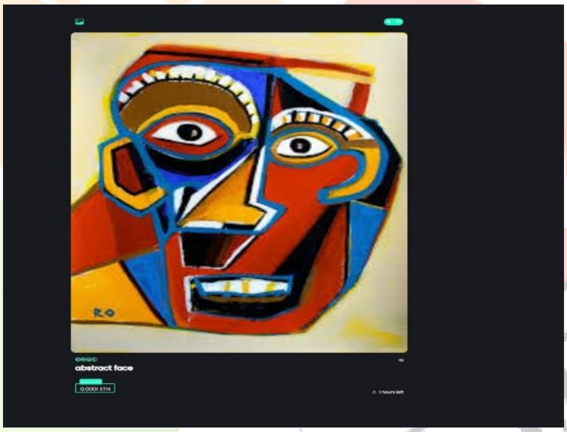

Figure 4.2 NFT

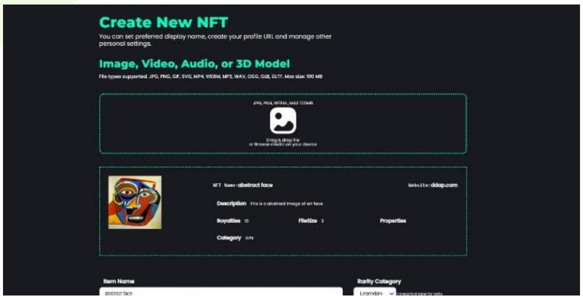

Figure 4.3 Creation Of NFT

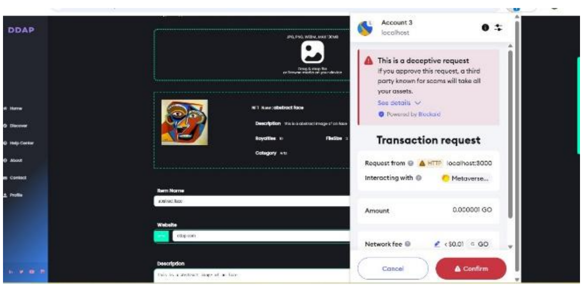

Figure 4.4 Selling Of NFT

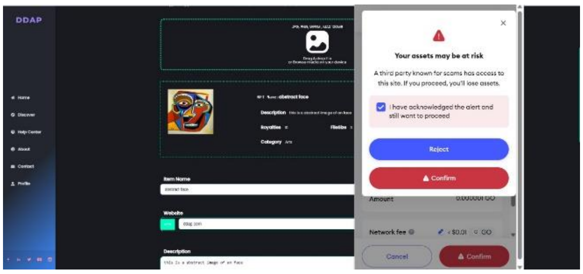

Figure 4.5 Confirmation OF Transaction

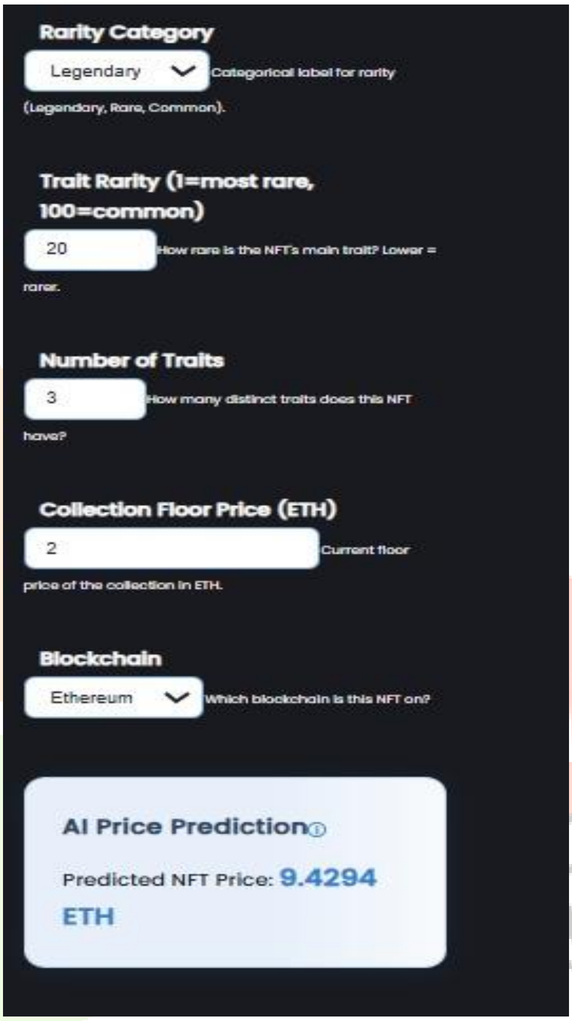

Figure 4.6: AI Based NFT price Predictor

The figures illustrate the key user interface components and features of the implemented NFT marketplace and AI price predictor. Figure 4.1 displays the homepage of the decentralized application (DDAP NFT Marketplace), offering a clean, user-friendly entry point for users to browse or create NFTs. Figure 4.2 showcases an example of an NFT listed on the platform, giving a visual representation of the digital asset along with its details. Figure 4.3 depicts the interface used to create or mint a new NFT, allowing users to upload media files and provide essential metadata. Figure 4.4 presents the selling interface, where users can list their NFTs for sale, specifying the price and confirming ownership. Figure 4.5 shows the confirmation page that appears when a transaction is about to be completed, ensuring clarity before the transfer of ownership and funds. Finally, Figure 4.6 illustrates the AI-based NFT price predictor, where users input attributes such as rarity, number of traits, and floor price to receive an estimated market value of the NFT, powered by the trained Random Forest Regressor model.

5. Results and Evaluation

This section outlines the evaluation of both the decentralized NFT marketplace functionality and the performance of the integrated AI price prediction model.

Marketplace Functionality Evaluation:
Qualitative testing confirmed that the primary features of the decentralized marketplace worked as expected. Users were able to mint, list, and purchase NFTs through smart contract interactions, with successful ownership transfers recorded on the blockchain and verified via testnet explorers like Etherscan. During the minting process, metadata and image files were uploaded to IPFS and could be retrieved through their corresponding CIDs, ensuring that the content was stored independently of any centralized server. The platform's user interface was generally smooth and user-friendly, particularly for individuals with experience in Web3. It allowed seamless wallet connections, navigation through NFT listings, and interaction with transaction flows. Integration with MetaMask via Web3Modal provided a secure and dependable method for users to approve and execute transactions. Performance-wise, the frontend was responsive and off-chain data retrieval was efficient; however, blockchain operations were affected by gas fees and confirmation delays, which are inherent limitations of Ethereum's current infrastructure.

AI Price Predictor Performance Evaluation:
The RandomForestRegressor model was trained and tested using a synthetic NFT sales dataset. Evaluation on a 20% hold-out test set yielded an R^2 score of 0.8421 and a root mean squared error (RMSE) of 0.3175 ETH. These results indicate that the model was able to explain around 84% of the variance in NFT prices, suggesting a strong correlation between the selected features and sale outcomes. A scatter plot comparing predicted and actual prices showed most predictions clustering around the y = x line, particularly in the lower price range (under 20 ETH). However, higher-priced NFTs showed greater deviations and some outliers were evident, which is typical when predicting extreme values. A feature importance chart indicated that variables such as collection floor price, trait rarity, and time since first sale played a significant role in model predictions. Market engagement metrics like the number of bids or likes also contributed, while some features such as creator identity or textual descriptions were less impactful due to encoding limitations.

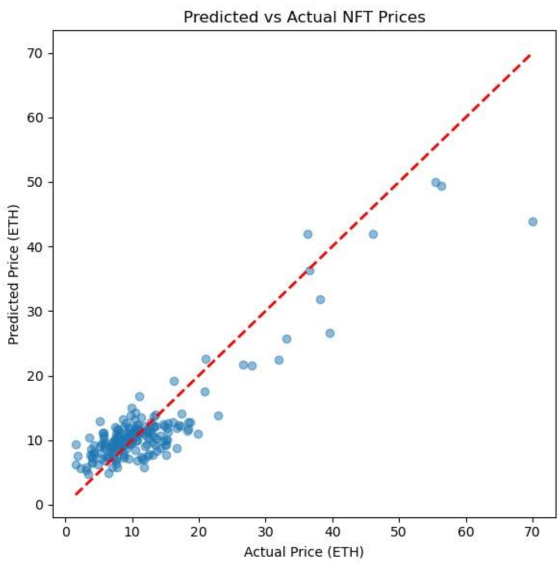

Figure 5.1: Predicted vs. Actual NFT Prices (in ETH)

The scatter plot above shows the Predicted vs. Actual NFT Prices as part of the AI-based price prediction module integrated into the Ethereum-based NFT marketplace. Each point represents an NFT in the test dataset, with the actual price on the X-axis and the predicted price on the Y-axis. The red dashed line represents the ideal case where predicted prices exactly match actual prices (y = x). The clustering of points near the line, especially in the lower price range (< 20 ETH), indicates that the model performs well in predicting common NFT prices. However, as prices increase, prediction errors become more noticeable, with a few outliers deviating significantly. This is typical in regression models, where predicting extreme values is more challenging. Overall, the plot supports that the model provides reasonably accurate estimates, reinforcing the potential of AI integration in blockchain-based NFT platforms.

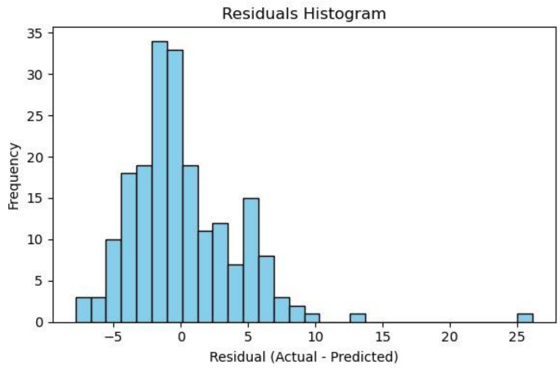

Figure 5.2: Residuals Histogram of NFT Price Predictions

This histogram displays the distribution of residuals (actual price minus predicted price) for the test dataset. Most residuals are centered around zero, indicating that the model makes fairly accurate predictions. However, a slight rightward skew and a few large outliers suggest occasional overestimation or underestimation, particularly for higher-priced NFTs.

6. Future Enhancements and Scope

1. Real-World Data Integration:
To improve model reliability, real-time NFT sales data should be collected from major marketplaces (like OpenSea or Magic Eden) using APIs or data indexers such as The Graph. This also includes creating a data pipeline for regular updates and implementing cleaning processes to handle real-world blockchain data noise.

2. Improved AI Modeling:
Future models can explore advanced algorithms like XGBoost, LightGBM, or neural networks. Feature engineering should include social sentiment, network behavior, rarity combinations, and market volatility. For better transparency, explainability tools like SHAP can clarify predictions.

3. Scalability & Cost Optimization:
Adopting Ethereum Layer 2 solutions (e.g., Polygon, Optimism) will help reduce gas fees and improve transaction speed. Off-chain computation can also reduce on-chain load and costs.

7. Conclusion

This project presents the successful development of a decentralized NFT marketplace built on Ethereum and IPFS, with the added innovation of an AI-based price prediction model using RandomForestRegressor. The platform enables secure minting, listing, and trading of ERC-721 tokens, with AI-generated price estimates enhancing user decision-making. Evaluation on synthetic data showed promising results ($R^2 \approx 0.84$, RMSE $\approx$ 0.32 ETH), indicating the model's ability to capture meaningful pricing patterns. However, limitations include reliance on synthetic data and Ethereum's scalability and cost issues.

Future improvements will focus on real-world data integration, advanced modeling, continuous learning, Layer 2 adoption, and expanded marketplace features. Overall, this work highlights the strong potential of combining AI with blockchain to advance NFT trading ecosystems.

II. Acknowledgment

We would like to express our sincere gratitude to Dr. Rizwana Shaikh , Professor, SIES Graduate School Of Technology, for their invaluable guidance, continuous support, and encouragement throughout the course of this project. Their expertise and insights were crucial to the development and completion of this work. We also thank the SIES Graduate School Of Technology for providing the necessary infrastructure and resources that enabled us to carry out our research effectively.

Lastly, we extend our thanks to our peers, friends, and family for their support and motivation during this endeavor.

References

[1] Shriyankshi Gupta, Chaitanya Kashyap, Akshat Tewari, "NFT Social Marketplace", June 2022.
[2] Prof. Rakhi Bhardwaj, Tanmay Kadam, Shubham Waghule. " Nft Marketplace with Digital Currency Exchange", February 2023.
[3] Prof. Himanshi Agrawal, Abhay Bodhe, Ananta Sontakke, Aniket Shahane, Rushi Bihade. "Artcart: NFT Marketplace", International Journal of Research Publication and Reviews, Vol 3, no 11, pp 2646-2653 November 2022.
[4] D. Yaga, P. Mell, N. Roby and K. Scarfone, "Blockchain Technology Overview", arXiv: Cryptography and Security, 2018.
[5] D. Vujičić, D. Jagodic and S. Ranđić, "Blockchain Technology, Bitcoin and Ethereum: A Brief Overview", 17th International Symposium Infoteh- Jahorina (INFOTEH), 2018. Available: 10.1109/infoteh.2018.8345547.
[6] L. Ante, "The non-fungible token (NFT) market and its relationship with Bitcoin and Ethereum", SSRN Electronic Journal, 2021. Available: 10.2139/ssrn.3861106.
[7] U. Chohan, "Non-Fungible Tokens: Blockchains, Scarcity, and Value", SSRN Electronic Journal, 2021. Available: 10.2139/ssrn.3822743.
[8] A. Thilagaraj and J. Davis. "Non-Fungible Token (NFT) – The Game Changer in The Digital Art World", Ciencia Y Sociedad, vol. 51, pp. 190-194, 2021.
(Consider adding references specific to: RandomForestRegressor, scikit-learn, IPFS, ethers.js, web3modal, Next.js, NFT price prediction studies, Layer 2 solutions like Polygon/Optimism)