**Title**: Addition of EIP-4337 to HermesNFT

**Author**: Eduardo Marques

**Created**: 27 February 2024

**Introduction:**

The HermesNFT contract is a smart contract based on the ERC721 standard that allows for the creation and management of non-fungible tokens (NFTs) with dynamic properties and loyalty rewards. In order to enhance user experience and reduce the friction associated with minting NFTs, we propose the integration of EIP-4337 functionality into the HermesNFT contract.

**Overview of EIP-4337:**

EIP-4337, also known as "Account Abstraction for Native Meta Transactions," introduces a mechanism for allowing contract functions to be executed by accounts other than the direct caller, thereby abstracting away the need for users to interact directly with the Ethereum blockchain. By implementing EIP-4337, users can perform actions such as minting NFTs without needing to have an Ethereum wallet or manage gas fees.

**Benefits of EIP-4337 Integration:**

* + Enhanced User Experience: Users can mint NFTs without the need for an Ethereum wallet or handling gas fees, simplifying the process and making it more accessible.
  + Reduced Friction: By abstracting away the complexities of blockchain interaction, EIP-4337 reduces the friction associated with minting NFTs, leading to increased adoption and usage.
  + Scalability: The account abstraction feature enables off-chain entities to interact with the blockchain on behalf of users, potentially improving scalability by offloading some processing burden from the Ethereum network.

**Contract Architecture:**

The HermesNFT contract interacts with the account abstraction contracts through a set of functions designed to facilitate NFT minting and payment processing. The following functions are key components of the contract architecture:

* + paymentAccepted: Checks if the payment for minting the NFT is accepted. This function validates the payment details and returns a boolean value indicating whether the payment is accepted or not.
  + exchangeCurrencyToETH: Exchanges the specified amount of currency to ETH using a third-party exchange service like Binance. This function submits a request to the exchange service, receives the equivalent amount of ETH, and returns it to the caller.
  + mint: Mints a new NFT and assigns it to the specified address after processing the payment. This function first checks if the payment is accepted, then exchanges the currency to ETH, and finally mints the NFT.

The following changes will be made:

Add:

/// @dev Function to check if the payment is accepted.

/// @param \_currencyAmount The amount of currency sent by the sender.

function paymentAccepted(uint256 \_currencyAmount) internal returns (bool) {

// Step 1: Validate the payment details received from the third-party payment processor like Binance

// Step 2: Check if the payment amount meets the required criteria

// Step 3: Verify if the payment is successfully processed and confirmed by the payment processor

// Step 4: Return true if the payment is accepted, otherwise return false

return true;

} else {

return false;

}

}

/// @dev Function to exchange the specified amount of currency to ETH using a third-party exchange service like Binance.

/// @param \_currencyAmount The amount of currency to exchange.

/// @return The equivalent amount of ETH.

function exchangeCurrencyToETH(uint256 \_currencyAmount) internal returns (uint256) {

// Step 1: Connect to the Binance API

// Step 2: Submit a request to exchange the specified currency amount to ETH

// Step 3: Receive the equivalent amount of ETH from Binance

// Step 4: Return the received ETH amount

return ethAmount;

}

Change funtion to mint:

/// @dev Function to mint a new NFT and assign it to the specified address after processing the payment.

/// @param \_to The address to which the NFT will be minted.

/// @param \_currencyAmount The amount of currency to be exchanged to ETH for minting.

function mint(address \_to, uint256 \_currencyAmount) external returns (bool) {

// Step 1: Check if the payment is accepted

require(paymentAccepted(\_currencyAmount), "Payment not accepted");

// Step 2: Exchange the specified currency amount to ETH

if (paymentAccepted(\_currencyAmount)) {

uint256 ethAmount = exchangeCurrencyToETH(\_currencyAmount);

// Step 3: Mint the NFT to the specified address

\_mint(\_to, totalSupply() + 1);

return true;

} else {

return false; // Payment not accepted, return false

}

}

**Architectural Diagram**:

The following architectural diagram illustrates the interaction between the HermesNFT contract and the account abstraction contracts:

A blue rectangular sign with white text

Description automatically generated

**Conclusion**:

The integration of EIP-4337 functionality into the HermesNFT contract represents a significant advancement in NFT minting capabilities, offering a seamless and user-friendly experience for creators and collectors alike. By abstracting away the complexities of blockchain interaction and enabling off-chain entities to perform transactions on behalf of users, EIP-4337 enhances accessibility, usability, and scalability of the HermesNFT platform.