WildLifeGuardian Soulbound Token Documentation

The WildLifeGuardianToken contract is a unique ERC721 token contract designed for Wildlife conservationists which we refer to as wildlife Guardians. This contract enables the creation, ownership, and management of non-fungible tokens (NFTs) associated with the Wildlife Guardian initiative. These NFTs can be claimed by members through a secure Merkle proof process. This simply means a person can only claim the NFT if and only when they pass certain requirements, Additionally, the contract allows for the safe minting and burning of tokens, as well as the addition of new Merkle root hashes for token whitelisting. The contract has various functions to manage the token minting, claiming, and burning processes, along with the necessary conditions for ERC721 compliance.

Overview

The WildLifeGuardianToken contract includes several key components:

ERC721 Implementation: This contract extends the functionality of the ERC721 and ERC721URIStorage contracts, providing the basic structure for creating and managing NFTs.

Ownable Contract: The contract includes the Ownable contract, enabling the specification of the contract owner and associated permissions.

MerkleProof Library: The contract utilizes the MerkleProof library from OpenZeppelin, allowing for secure verification of Merkle proofs during the token claiming process.

Contract Details

State Variables

_tokenIdCounter: A private variable used for tracking the number of tokens minted.

rootHash: A public bytes32 variable representing the Merkle root hash for token whitelisting.

tokenUri: A string variable that holds the token URI for the new tokens.

DAO: A variable representing the GofundmeDAO contract.

claimed: A mapping that keeps track of claimed tokens.

addressToIds: A mapping that stores the token IDs corresponding to specific addresses.

daoMembers: A mapping that stores information about DAO memberships.

Functions

constructor: Initializes the contract with a provided owner address and token URI.

safeMint: Safely mints new tokens and assigns them to specified addresses.

claimToken: Allows the claiming of a token for an address using a Merkle proof.

addRootHash: Adds a new Merkle root hash for token whitelist ting.

burn: Burns a token by its ID, restricted to the DAO contract.

showIds: Retrieves the token ID associated with a specific member's address.

transferFrom: Overrides the transferFrom function to disable transfers.

tokenURI: Overrides the tokenURI function to return the URI associated with a specific token ID.

supportsInterface: Overrides the supportsInterface function to check the supported interfaces.

Error Handling

InvalidAddress: Throws an error if the provided address is invalid.

AlreadyClaimed: Throws an error if the token has already been claimed.

NotWhitelisted: Throws an error if the address is not whitelisted.

Usage

This contract provides a secure and efficient way to manage NFTs for the Wildlife Guardian initiative. It ensures that only eligible members can claim tokens using a Merkle proof and allows the DAO to manage token issuance and burning. The contract is designed to enforce ownership restrictions and prevent unauthorized transfers. Additionally, the contract enables the storage and retrieval of membership information for DAO members.