**Overview**

The Escrow Contract is a smart contract built on the Ethereum blockchain using Solidity. It facilitates agreements between two users for the sale of tangible items, with payment in Ether (ETH) or any ERC-20 token. The contract includes a basic dispute resolution mechanism involving an arbitrator.

**Features**

* Resistance to theft of funds from the contract.
* Third-party settlement of disputes through an arbitrator.
* Support for any token in the ERC-20 standard.

**Getting Started**

**2.1 Prerequisites**

* Ethereum wallet or a development environment like Remix, metamask.
* Ethereum account with enough Ether for deployment and testing in remix/hardhat/truffle.
* ERC-20 token contract address if using a specific token.

**2.2 Deploying the Contract**

* Copy the EscrowContract.sol code into your preferred Solidity development environment.
* Compile the contract.
* Deploy the contract to the Ethereum blockchain.

**3. Usage**

**3.1 Creating an Agreement**

* Use the createAgreement function to create a new agreement specifying the seller, arbitrator, payment amount, and token.

**3.2 Making a Payment**

* Buyers can make payments using the pay function, supporting both Ether and ERC-20 token payments.

**3.3 Delivering the Item**

* Sellers can mark the item as delivered using the deliverItem function.

**3.4 Raising a Dispute**

* Either party can raise a dispute using the raiseDispute function.

**3.5 Resolving Dispute**

* The arbitrator can resolve a dispute using the resolveDispute function.

**3.6 Releasing Payment**

* Sellers can release payments using the releasePayment function after delivering the item.

**4. Technical Details**

**4.1 Contract Structure**

* The contract is structured with functions for agreement creation, payment, dispute resolution, and payment release.

**4.2 Enumerations**

* The State enumeration represents different states of an agreement.

**4.3 Structs**

* The Agreement struct stores details about each agreement.

**4.4 Modifiers**

* Modifiers are used to restrict access to specific functions (onlyParties, onlyArbitrator).

**4.5 Events**

* Events (AgreementCreated, PaymentReleased, ItemDelivered, DisputeRaised, DisputeResolved) provide transparency on contract actions.

**5. Testing**

**5.1 Running Automated Tests**

* Use testing frameworks Hardhat to run automated tests.

**7. Conclusion**

* This document provides an overview of the Escrow Contract.