Konkala Rithvik (2020A8PS0517H)

Visweswar Sirish Parupudi (2020AAPS0330H)

Gautham Gutta (2020AAPS2204H)

Gokul Pradeep (2018B5A70785H)

THIS FILE WILL GIVE A BRIEF BUT EXHAUSTIVE DESCRIPTION OF THE IMPLEMENTED CODE.

IMPORTANT NOTE :- Please run the following code on remix.ethereum.org online IDE.

1) SOLIDITY :

- Solidity is an object-oriented programming language created specifically by the Ethereum Network team for constructing and designing smart contracts on Blockchain platforms.
- It's used to create smart contracts that implement business logic and generate a chain of transaction records in the blockchain system.
- In our current problem statement, we have to draft a smart contract between a buyer and a merchant (here abc.com or xyz.com).
- We have to make it such that the buyer gets the product they ordered and only after buyer's acknowledgement the seller gets the money and buyer gets the goods.
- We deploy a smartcontract for this scenario with the following logic.

2) STEPS OF TRANSACTION :

STEP 1

- The seller (xyz.com) has to draft the smartcontract for a product.
- Then seller pays a security deposit into the smartcontract which is double the specified cost of the product they desire to sell
- This is to ensure that the seller has an incentive to actually send the promised product to the buyer, in case things go wrong the security deposit is lost and used to make up for buyer's loss.

STEP 2

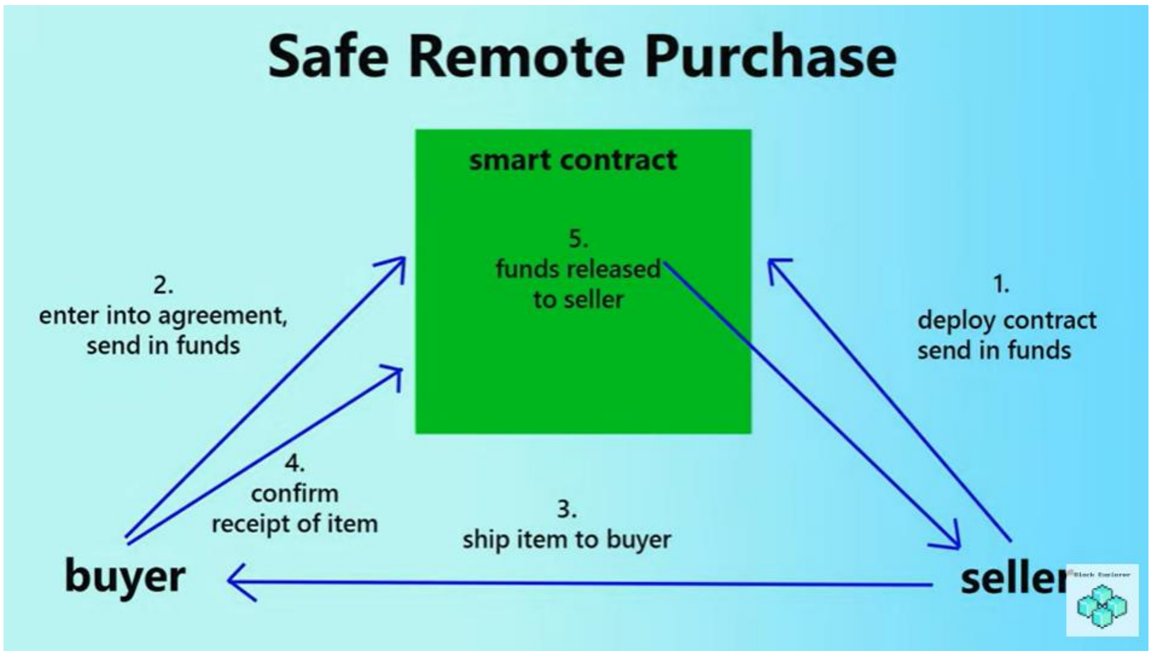
- Involves the buyer confirming that he wants the product by paying a security deposit (in this case also twice the value of price of requested goods)

- The buyer security deposit and seller security deposits are refunded after the transaction.
- The refunds are the incentives for the buyer and seller to honestly report the status of delivery of goods.

  STEP 3
- now the seller ships the promised goods to the buyer.

  STEP 4
- Buyer now confirms the product on contract and start the process for his refund of security deposit minus the cost of good.
- So now the buyer has the goods and half of his security deposit back since the other half is used up as payment for goods automatically by the contract.

  STEP 5
- Now the seller initiates the payment function for themselves (this function can only be initiated after confirmation from buyer is received).
- Seller now gets his security deposit back (twice the value of goods) plus half the security deposit from the buyer (compensation for the goods he sent).
- Hence the net transaction of the value of goods is handled by automation and is completely fair for both the involved parties.
- Below is a diagram representing the steps.


Safe Remote Purchase
smart contract
5. funds released to seller
2. enter into agreement, send in funds
1. deploy contract send in funds
4. confirm receipt of item
3. ship item to buyer
buyer
seller

3) Variables description :

- Seller - is an address type variable that stores the value of the seller (the person who drafts the contract, in this case xyz.com)
- Buyer - is an address type variable that stores the value of the buyer (The person who requests goods from the seller)
- Cost – is an uint type variable that stores the cost of the goods that are exchanged in the transaction between buyer and seller
- State – is enum (user-defined data types) and has the 4 states of transaction which are
- Created - contract enters this state when seller deploys the contract.
- Locked - contract enters the state when buyer puts in a value equal to twice the cost.
- Release - contract enters this state when buyer confirms that he received the product.
- Inactive - contract enters this state when seller is refunded and given the extra paid by buyer.

4) Functions description :

- Cancel() – can be called by the seller if he doesn't wish to sell the product any longer, but it can only be triggered before a buyer locks in.
- confirmPurchase() – can be called only by buyer, buyer has to send a message with a value of twice the cost of product only then will he be locked into the contract with the seller
- confirmReceived() – can be called only by the buyer. This function is used as a confirmation on buyers end that he received the desired product. As soon as buyer presses this he is refunded his half of deposit and the other half is sent as payment for the goods to the seller.
- function refundSeller() – can be called by seller only after contract enters release state i.e confirmation by buyer that goods have arrived is received. This function gives back the seller his security deposit back and additionally gives him half the deposit of buyer as payment for the goods received.
- The code also has few MODIFIERS which are code snippets used in function as condition checks.

- onlyBuyer() – used to check if the person accessing the function of contract is the buyer. Gives clearance only with buyer.
- onlySeller() - used to check if the person accessing the function of contract is the seller. Gives clearance only with seller.
- inState(State state_) – checks whether the current state of the contract is the same as the state specified in the state_ argument.
- The MODIFIERS have custom errors which can be evoked with the use of REVERT.
- error OnlyBuyer(); /// Only the buyer can call this function.
- error OnlySeller(); /// Only the seller can call this function.
- error InvalidState(); /// The function cannot be called at the current state.
- error CostNotEven(); /// The provided Cost has to be even.

5) Events description :

Event is an inheritable member of a contract. An event is emitted, it stores the arguments passed in transaction logs. These logs are stored on blockchain and are accessible using address of the contract till the contract is present on the blockchain. An event generated is not accessible from within contracts, not even the one which have created and emitted them.

- Aborted() – emitted when contract is cancelled by seller before buyer has chance to lock in money
- PurchaseConfirmed() – emitted after buyer has locked in the contract by paying the security deposit for the goods.
- ItemReceived() – emitted after buyer acknowledges that they have received the goods.
- SellerRefunded() – emitted after the seller has received his security deposit and the payment from buyer.