# 533.https://stackoverflow.com/questions/70383758/integrate-with-ethereum-smart-contract-using-other-person-signer-on-server-side

**T:**Integrate with Ethereum smart contract using other person Signer on server-side (NodeJS)

**Q:**I'm building a website for selling NFTs. The way I saw to let visitors pay for a NFT item is by integrating with the smart contract on client side in order to get Signer from their wallet (e.g. Metamask).  
  
My client code:  
  
export async function createItemToken(itemPrice: number): Promise<string> { const web3Modal = new Web3Modal(); const connection = await web3Modal.connect(); const provider = new ethers.providers.Web3Provider(connection); const signer = provider.getSigner(); const contract = new ethers.Contract( contractAddress, MyNFTContract.abi, signer, ); const formatedItemPrice = ethers.utils.parseUnits(String(itemPrice), 'ether'); const transaction = await contract.transferToken( {value: itemPrice}, ); const tx = await transaction.wait(); return tx.transactionHash;}  
  
WARN: THIS PARAGRAPH CONTAINS TAG: [CODE]   
  
I understand the access to visitor's wallet is only on the client side but it doesn't make sense to let the client side do the call to the smart contract. For example, the visitor can enter different price for the item.  
  
How to avoid it? is there a way to do this integration on the server side (NodeJS)?  
  
I couldn't find a way to get visitor's signer on the server so maybe my all flow is not correct...thanks for the help!

**C1:**named hyperlinks will highly appreciate your input about this

1 **Answer**

**A1:**the visitor can enter different price for the item  
  
Sure they can. But the smart contract should also validate whether the value is an expected price. If it's not, it should revert the transaction.  
  
WARN: THIS PARAGRAPH CONTAINS TAG: [CODE]   
  
Based on the provided code and questions, I'm assuming the price for token creation is only stored off-chain. You should use the contract as the source of truth for the price and validate it on-chain as well.  
  
pragma solidity ^0.8;contract MyNFTContract { function transferToken() external payable { require(msg.value == this.getPrice(), "Incorrect value"); // ... } function getPrice() external pure returns (uint256) { // can by also dynamic based on the token ID or any other on-chain param return 0.5 ether; }}  
  
WARN: THIS PARAGRAPH CONTAINS TAG: [CODE]

**C1:**Thanks for the answer! I'm building a marketplace and i'm trying to understand how sites like Opensea/Rarible doing it. The solution of writing to the smart contract the price every time someone add an item to sell it possible but it will cost him gas fee. When I play with Opensea/Rarible they didn't ask me to pay anything when I create a new item for sell (even the gas fee of the mint is on the buyer) - how do they do it?