

# CSE 424

# INTRODUCTION TO BLOCKCHAIN

## PROJECT 2

### Contents

Challenges in the Real Estate Transaction Process .....	2
Necessity for a Solution.....	3
Blockchain Based Solution Proposal for Simplifying Property Buying and Selling Processes .....	3
Blockchain Solution .....	4
Advantages of Blockchain.....	4
Blockchain Application Examples .....	4
Summary of Blockchain Solution.....	4
Ethereum Blockchain on the Sepolia Test Network.....	5
1. Smart Contract and DApps Development .....	5
2. Large User and Developer Community .....	5
3. Extensive Infrastructure Support.....	5
4. Integration and Partnerships .....	5
5. Security and Transparency .....	5
Compatibility with Project Requirements .....	5
Smart Contract Overview .....	6
Contract Structure and Functions .....	6
Key Functions .....	6
Implementing and Deploying the Smart Contract.....	7
1. Writing the Contract in Remix.....	7
2. Compiling the Contract .....	8
3. Deploying the Contract.....	8
4. Integrating Contract Address and ABI into the Web Interface .....	9
5. Adding and Approving Real Estate Listings in the Interface.....	11
6. Bidding on Real Estate Listings .....	13
7. Conclusion .....	14

# Real Estate Transactions with Blockchain

In an era marked by significant technological progress, the real estate sector remains encumbered by outdated and inefficient practices. Traditional methods of property transactions are not only time-consuming and complex but are also characterized by substantial financial burdens due to intermediaries and administrative overheads. These conventional processes often lack transparency, leading to potential fraud and mistrust among parties.

Blockchain technology presents a groundbreaking opportunity to overhaul these archaic systems. Its inherent properties of decentralization, immutability, and transparency are perfectly suited to address the core deficiencies of the real estate market. By integrating smart contracts into real estate transactions, blockchain offers a way to conduct these exchanges directly, securely, and efficiently, without the need for intermediaries.

This project explores the application of blockchain technology to reform real estate transactions. We will analyze the current challenges faced by the industry, examine the limitations of traditional centralized systems, and propose a smart contract solution tailored to streamline and secure property exchanges. Our proposed framework not only aims to simplify the transaction process but also enhances security and trust, potentially transforming the landscape of real estate operations.

## Challenges in the Real Estate Transaction Process

The real estate transaction process is plagued by several critical issues that make property trading overly cumbersome and less efficient. These problems arise from the inherent complexities and the reliance on outdated systems, and they significantly impact all parties involved. Here are the key challenges:

- 1. High Transaction Costs:** Real estate transactions typically involve multiple intermediaries, including agents, lawyers, and banks. Each adds layers of fees that significantly increase the overall cost of transactions.
- 2. Lengthy Procedures:** The process of buying or selling property involves numerous steps, from initial listing to final closing. Each stage can be time-consuming due to the need for manual intervention and verification, often taking weeks or even months to complete.
- 3. Lack of Transparency:** Traditional real estate transactions lack a transparent process, making it difficult for parties to access real-time information. This opacity can lead to unequal negotiations, misinformation, and potential fraud.
- 4. Paper-Based Systems:** Many real estate transactions still rely on paper-based documentation, which is susceptible to loss, damage, and forgery. This reliance also contributes to the slow processing times and difficulties in data verification.
- 5. Regulatory Compliance:** Ensuring compliance with local and international real estate laws can be complex and requires significant effort from all parties to avoid legal issues.

These problems are significant because they not only add to the cost and time of real estate transactions but also affect the accessibility and fairness of the real estate market. The inefficiencies and risks associated with traditional methods can deter potential investors and complicate the buying and selling process for ordinary consumers.

## Necessity for a Solution

Addressing these issues is crucial for several reasons. Firstly, reducing transaction costs and streamlining processes can make real estate investments more attractive and accessible to a broader range of people. Secondly, increasing transparency and security in transactions can build trust among participants and reduce the likelihood of disputes and fraud. Lastly, by modernizing the real estate transaction process, the market can adapt more rapidly to changes in the economy and demographics, ensuring it remains robust and competitive.

A blockchain-based solution, specifically through the use of smart contracts, promises to address these issues by automating many of the processes involved, reducing the need for intermediaries, and providing a secure, immutable record of all transactions. This not only speeds up transactions but also significantly lowers costs, increases transparency, and enhances the overall integrity of the real estate market.

## Blockchain Based Solution Proposal for Simplifying Property Buying and Selling Processes

Property buying and selling is a process that involves many complex legal procedures and a large number of intermediaries to advance the transaction, leading to a time-consuming and costly environment. As a result of this environment, delays, errors and costs are encountered. This paper will focus on how blockchain technology can be used to simplify and optimise the property buying and selling process. We will take advantage of the basic features of blockchain such as transparency, security and automation and present how this technology can contribute to today's property transactions.

The scope of property purchase and sale transactions is very wide. It is an important financial transaction involving individuals, organisations and communities throughout history. There are many stages in these transactions, mainly land registry registration, transfer of ownership, financing and legal procedures. At each of these stages, many financial and temporal losses are observed due to the involvement of different intermediaries and institutions. The complexity and delay caused by this situation continues every day. The main problems of the traditional property buying and selling process are as follows:

**Complexity:** The involvement of many institutions and individuals in the process leads to a large number of documents and stages. This leads to a complex process.

**Delays:** The aforementioned communication and approval processes between organisations and individuals cause delays.

**High Costs:** the cost created by each of these stages and a large number of processes is expensive.

**Security Risks:** There are security risks such as fraud and forgery.

# Blockchain Solution

With the benefits of blockchain technology, there is an opportunity to completely change the entire order in the property buying and selling process. Transparency, security and automation, which are the most contributing and important attributes of blockchain technology, allow this process to be simplified and optimised.

## Advantages of Blockchain

**Transparency:** Each transaction made on the system is recorded on the blockchain. In the transparent environment created by this, everyone can access this information. Thanks to this feature, individuals and institutions gain the ability to follow transactions.

**Security:** A cryptographic secure environment is provided with the decentralised structure, which is the most important structure of the blockchain. Fraud and fraud incidents are minimised.

**Automation:** Smart contracts can eliminate manual processes by automating transactions such as land registration, transfer of ownership and payment.

**Low Costs:** As mentioned in the previous paragraphs, financial contribution is provided by eliminating intermediaries and institutions.

## Blockchain Application Examples

There are many examples where the blockchain-based solution we describe in this topic has been successfully implemented today. For example, in Sweden, several real estate projects have moved from the trial phase into use, allowing transactions to be recorded and tracked using blockchain technology. However, it has been seen that the developments we discussed theoretically have also been achieved in practice.

**Property Records:** Land registry records can be securely stored and tracked on the blockchain. With the features it brings, the risks of forgery and fraud can almost end.

**Smart Contracts:** With the smart contract, which we will focus more on in this project, it is possible to automate property transfer, payment and other transactions. Thus, transactions become faster and less costly.

**Tokenisation:** By tokenising real estate assets, investors can own smaller pieces of real estate assets, resulting in speed and cost gains in transactions.

## Summary of Blockchain Solution

As can be seen, blockchain technology can simplify and optimise real estate buying and selling processes. The main features of this technology such as transparency, security and automation can completely solve today's problems. For example, it can reduce complexity, avoid delays, reduce costs and increase security. Theoretically, if these processes are applied today, they can end the life of some professions while providing speed, security and low cost to people.

Of course, this process cannot be integrated in a very short time, because in order to fully integrate Blockchain technology into the real estate sector, legal and technical infrastructure work needs to start now and be built over the years.

# Ethereum Blockchain on the Sepolia Test Network

For our project, we have selected the Ethereum blockchain, specifically utilizing the Sepolia test network. Ethereum is widely recognized as one of the most robust platforms for developing smart contracts and decentralized applications (DApps). This decision was influenced by several factors that align with the project's requirements.

## 1. Smart Contract and DApps Development

Ethereum offers a Turing-complete programming environment with its Solidity language, enabling the creation of complex smart contracts. This capability allows developers to implement a wide array of applications, such as automated financial transactions, voting systems, and decentralized organizational functionalities.

## 2. Large User and Developer Community

Ethereum boasts a substantial developer community and a broad user base. This provides easy access to resources, support, and user feedback for emerging projects. The ecosystem is also supported by a variety of development tools, libraries, and frameworks.

## 3. Extensive Infrastructure Support

Ethereum is supported by infrastructure providers like MetaMask, Infura, and Alchemy. These platforms facilitate blockchain connectivity, network status monitoring, and application development. Ethereum also benefits from exploratory tools like Etherscan, which allows for easy tracking of transactions and contracts.

## 4. Integration and Partnerships

Ethereum supports integrations with various financial services, payment systems, and other blockchain networks. Standards such as ERC-20 (token standard) and ERC-721 (non-fungible token standard) facilitate compatibility with a wide range of applications and services.

## 5. Security and Transparency

Ethereum's decentralized nature ensures the security and transparency of all transactions connected to the network. Every transaction is verified by miners and permanently recorded on the blockchain, providing protection against manipulation.

## Compatibility with Project Requirements

The choice of Ethereum, particularly the Sepolia test network, for this project is primarily due to:

**Decentralized Marketplace:** Ethereum enables secure, authority-free transactions such as listing real estate and making offers. Transaction rules and logistics are automatically managed via smart contracts.

**Transparency and Trust:** Transactions between sellers and buyers are transparently recorded on the Ethereum blockchain, ensuring full visibility and trust for all parties.

**Flexibility and Customization:** Ethereum provides the flexibility needed to implement specific business logic and tailor the project to meet unique requirements.

The use of the Sepolia test network specifically allows for testing and development in an environment similar to the Ethereum mainnet but without the costs associated with real transactions. This provides an excellent platform for development and testing phases, enabling thorough evaluation and adjustment before deployment on the mainnet.

In summary, Ethereum's technical capabilities, security features, and extensive adoption make it an ideal choice to meet the requirements of this project. Moreover, the Sepolia test network offers a practical, cost-effective environment for development and testing, ensuring that the project can be refined and optimized with minimal expenditure.

## Smart Contract Overview

The RealEstateMarket smart contract, written in Solidity for the Ethereum blockchain, facilitates a decentralized platform for real estate transactions. This contract manages the listing of properties, making offers, and the acceptance or rejection of these offers.

### Contract Structure and Functions

**Property Structure:** Each property is represented with a structure containing details such as the owner's address, a description of the property, its price, and whether it is listed for sale.

**Offer Structure:** Each offer includes the buyer's address, the amount offered, and a status indicating whether the offer has been accepted.

**Properties:** An array that stores all listed properties.

**Offers:** A mapping that associates each property ID with an array of offers made for that property.

### Key Functions

**addProperty(string memory \_description, uint \_price):** Allows users to list a new property, automatically setting it as listed.

**makeOffer(uint \_propertyId, uint \_offerAmount):** Used to make an offer on a specific property, requiring the sent value to match the offer amount.

**acceptOffer(uint \_propertyId, uint \_offerId):** Enables the property owner to accept an offer. Upon acceptance, the offer amount is transferred to the owner, ownership of the property is transferred to the buyer, and the property is unlisted.

**rejectOffer(uint \_propertyId, uint \_offerId):** Allows the property owner to reject an offer, whereupon the offered amount is refunded to the buyer.

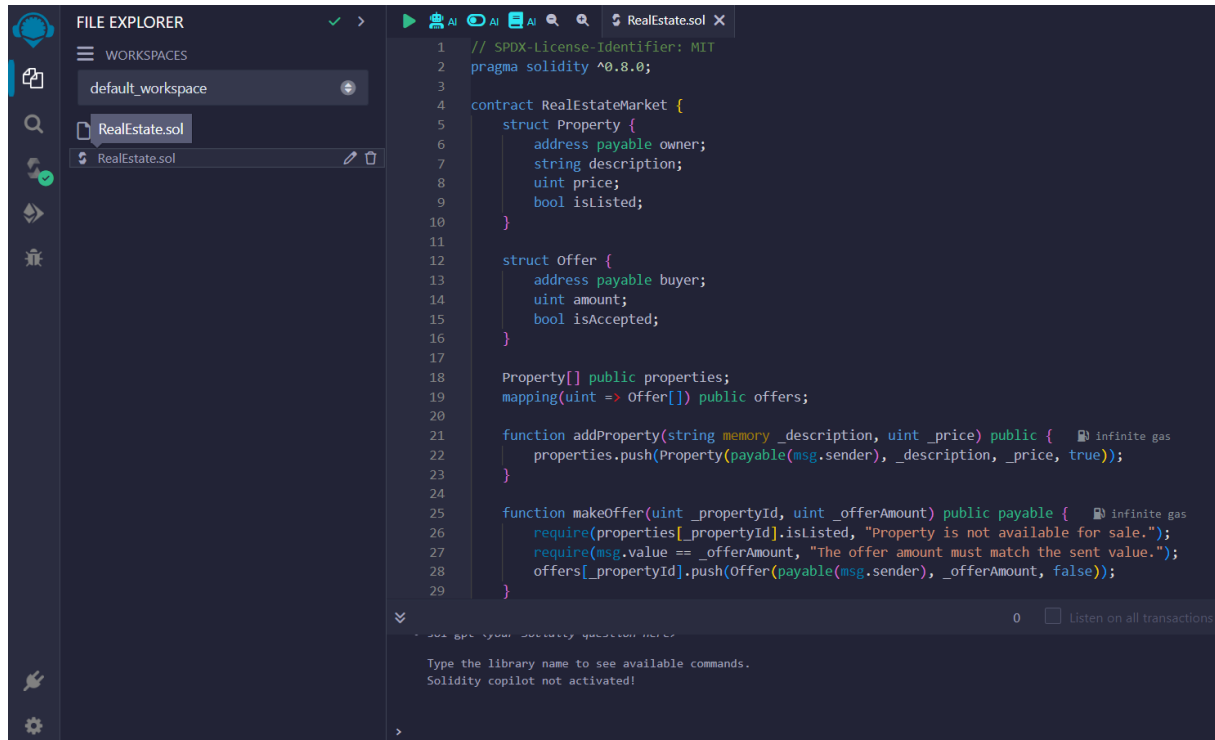
**getPropertiesCount():** Returns the number of properties listed on the platform.

**getOffers(uint \_propertyId):** Retrieves all offers made on a specific property.

**getOffersCount(uint \_propertyId):** Returns the total number of offers made on a particular property.

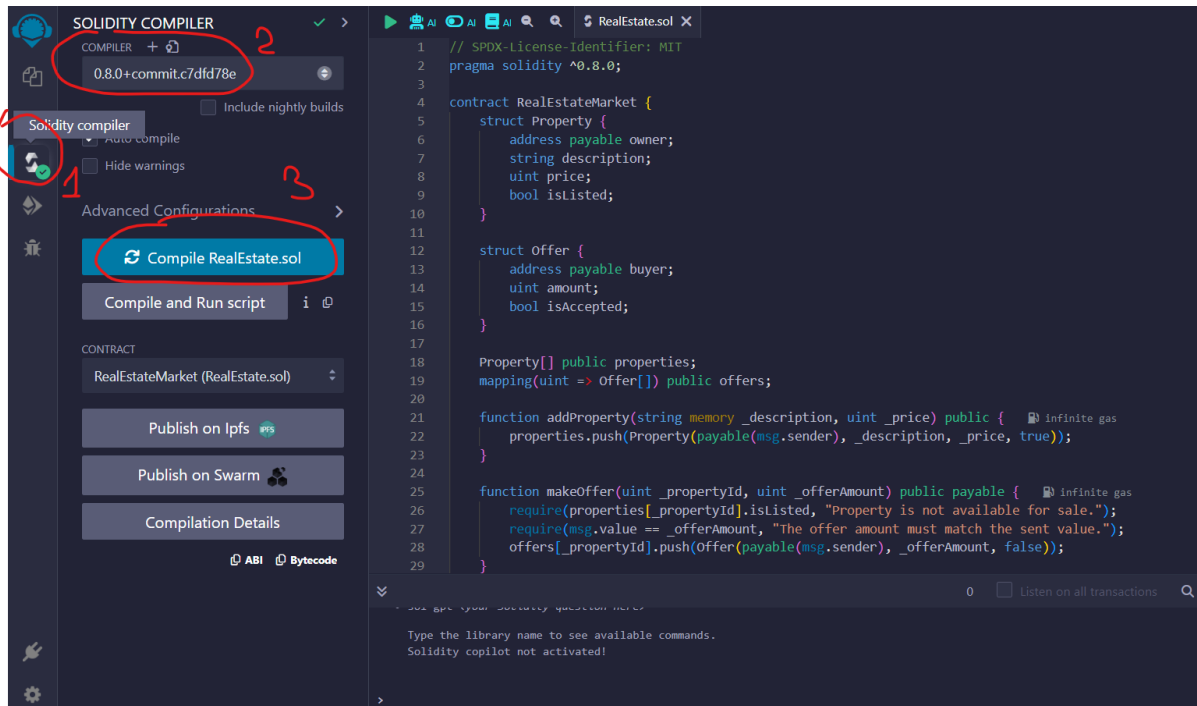
# Implementing and Deploying the Smart Contract

## 1. Writing the Contract in Remix



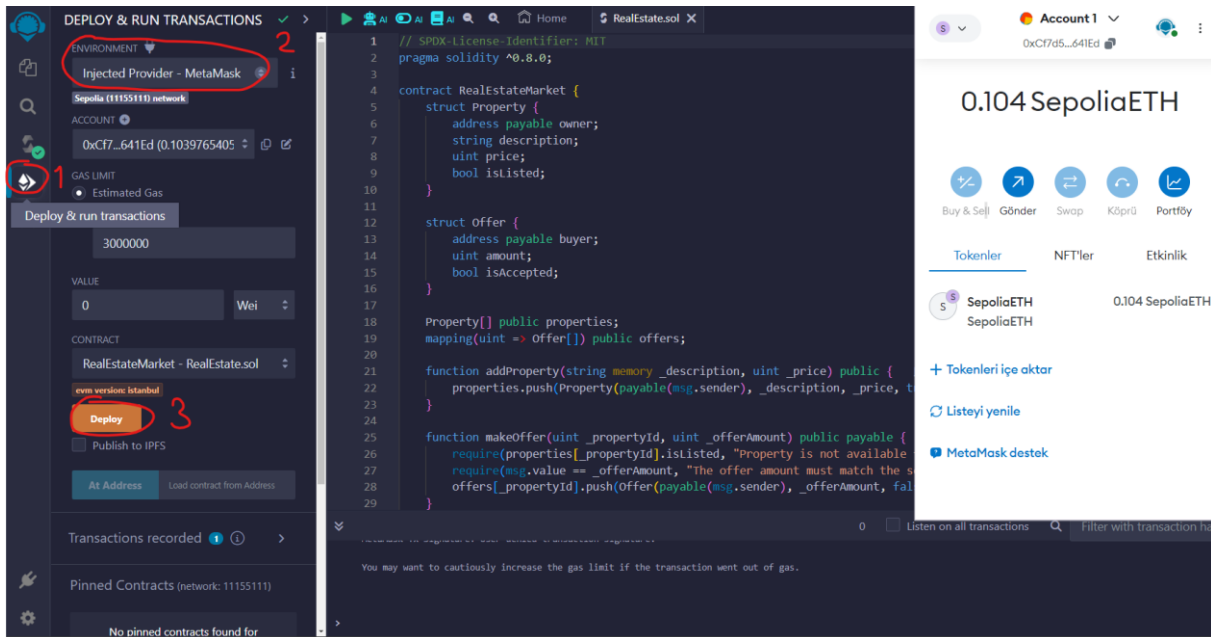
We start by saving our solidity code to the remix IDE. In remix, Contract information, contract functions, contract compile and contract deployment operations are performed. Switching between tabs is done from the leftmost section. The 3 important tabs are the 1st, 3rd and 4th tabs. The first tab contains the part seen on the screen. encoding processes and file layout are displayed here. Compiling is done in the 3rd tab. In the 4th tab, the contract's deployment and transaction operations are performed.

## 2. Compiling the Contract



After the smart contract is written, click on the area shown in the first box of the tabs on the left side of the screen and the compile screen shown in the picture should be displayed. Then, the compiler version should be selected in the field shown in the 2nd box, according to the solidity version used in the first line of the code. as the last step. The compile process is completed with the button shown in the 3rd box.

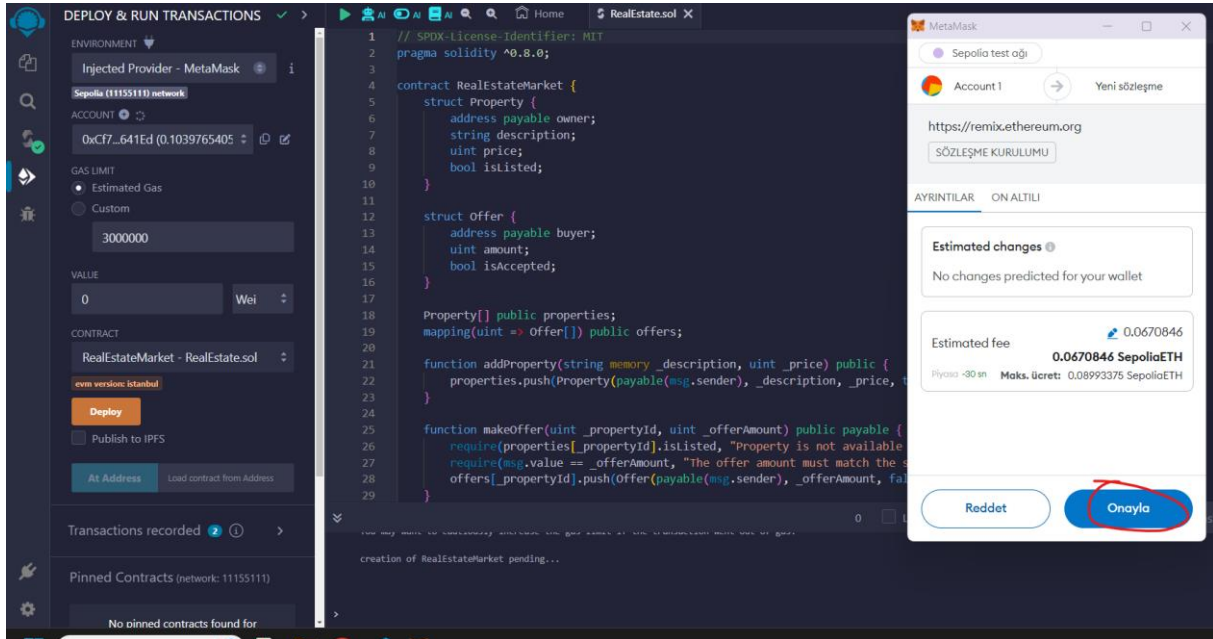
## 3. Deploying the Contract



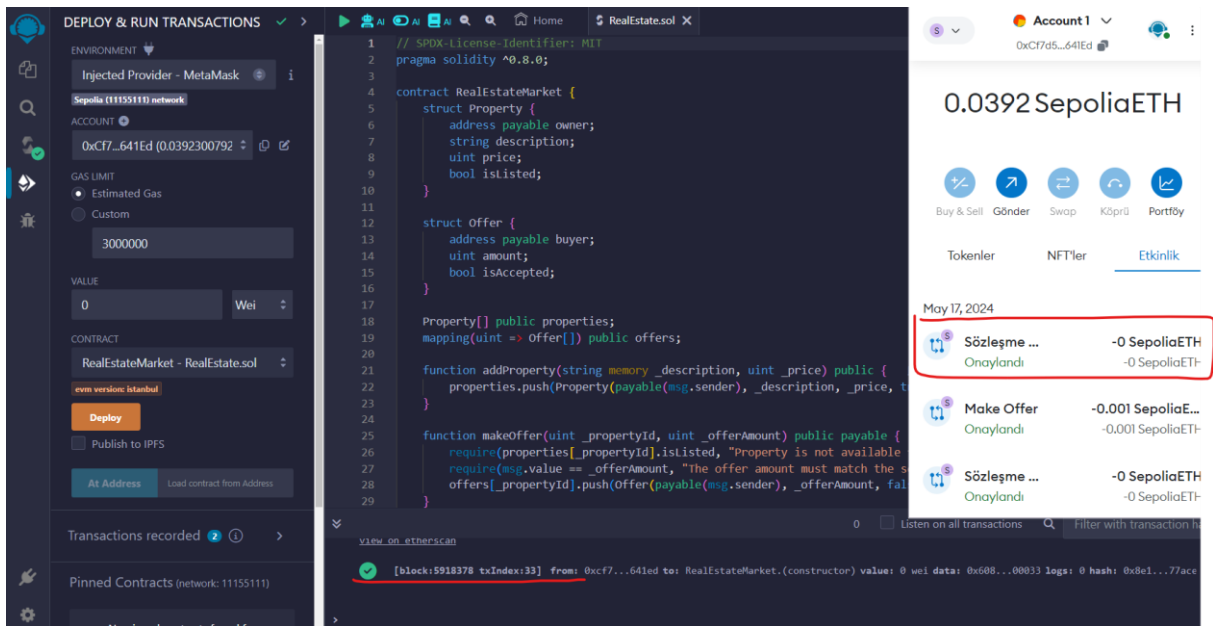
After our solidity code is compiled, it should be directed to this page from the tab you see as the 1st box on the screen in order to deploy the contract. Since the metamask wallet is used as seen on the right, the environment section should be selected as the metamask



environment as seen in the 2nd box. Then, by pressing the deploy button, a confirmation notification will be sent to metamask.



The transaction takes place shortly after the notification is approved.

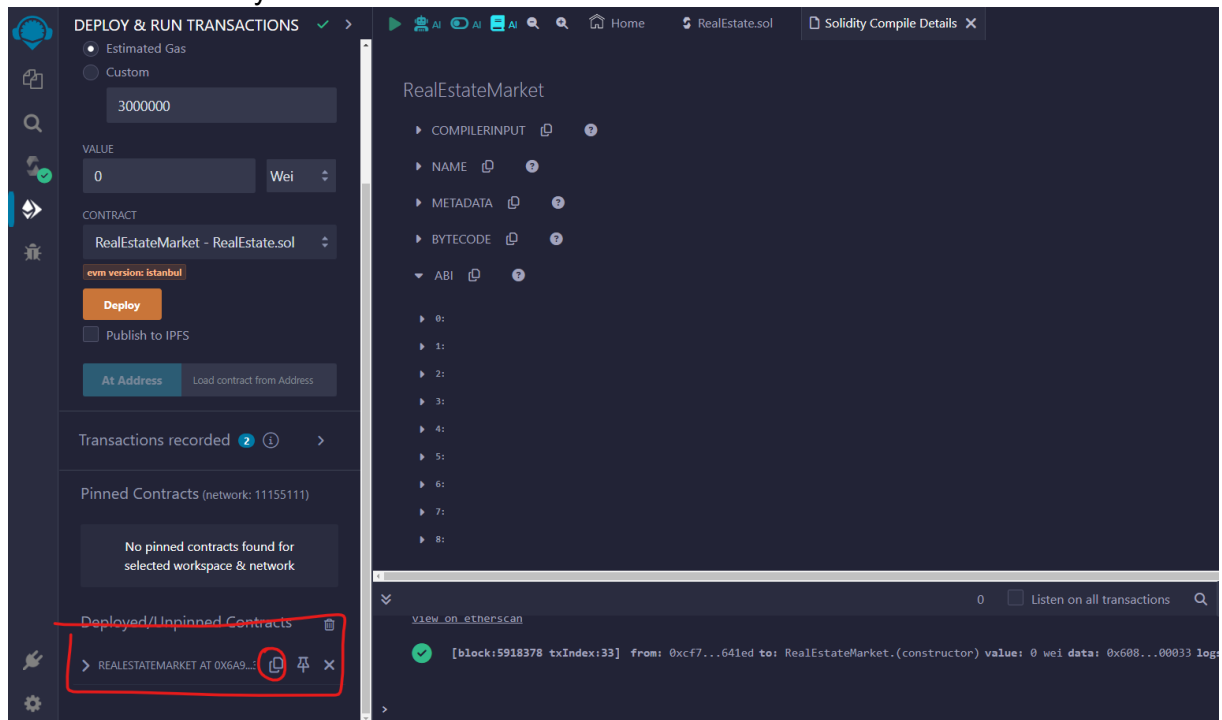


The status of the transaction appears in metamask history. The contract has been successfully deployed.

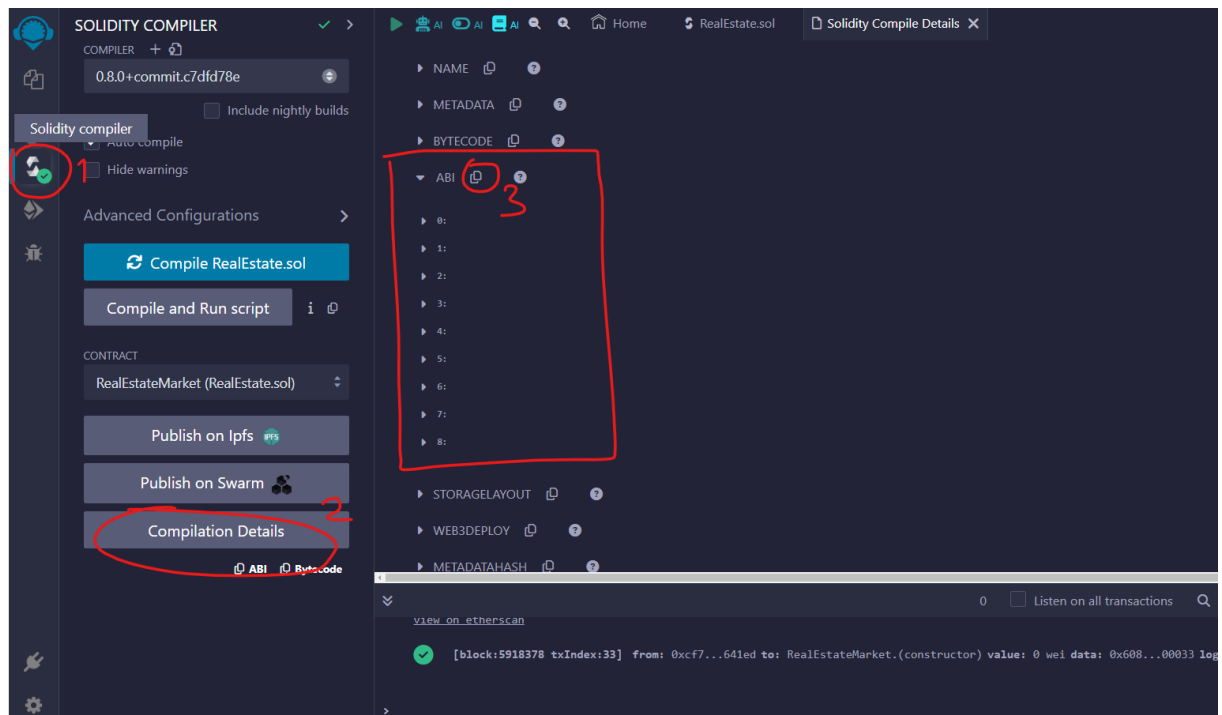
## 4. Integrating Contract Address and ABI into the Web Interface

To enable web interface interaction with a deployed smart contract, you must integrate the contract's Address and ABI into your JavaScript code. The contract address links your

application to the contract on the blockchain, while the ABI details the methods and structures needed to interact with the contract, facilitating seamless communication and transactions directly from the web interface.



After the contract is deployed, the contract address can be seen in the box shown at the bottom of the screen. Contract address information can be copied to the clipboard by pressing the button shown on the screen.



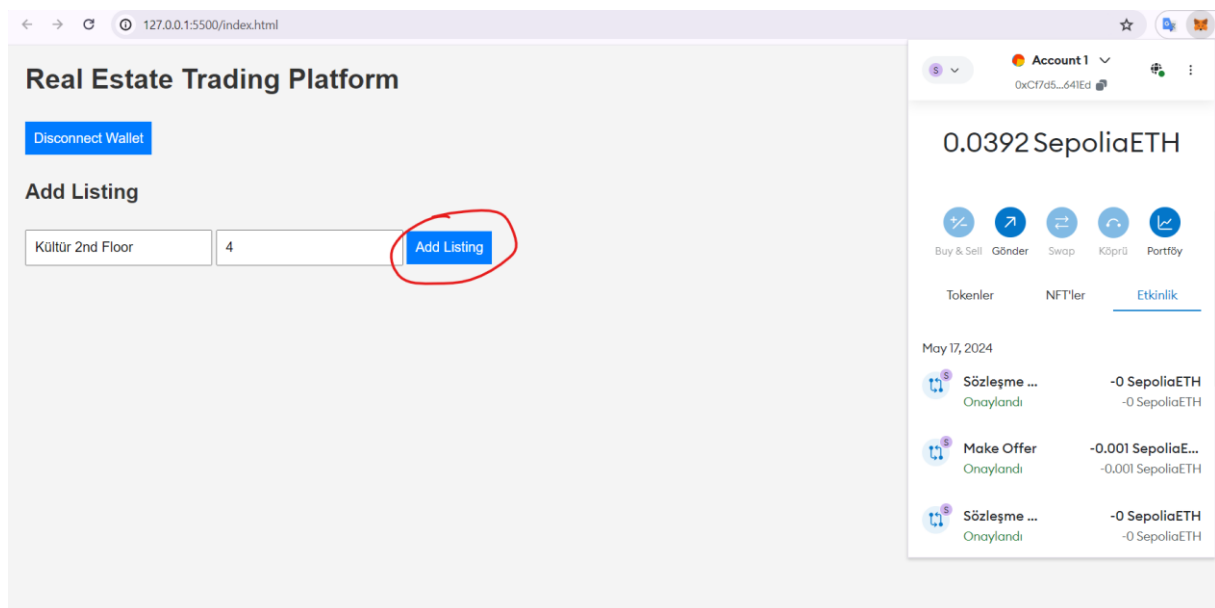
For ABI, go to the Solidity Compiler tab number 1, click on Compilation Details section number 2 and get the ABI in json format from section number 3.

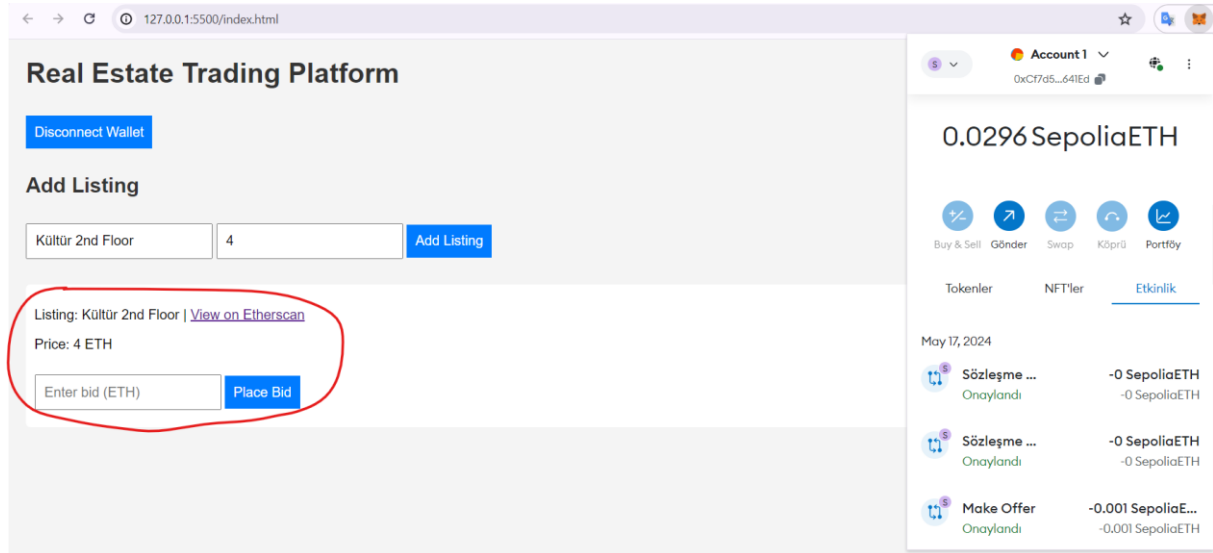
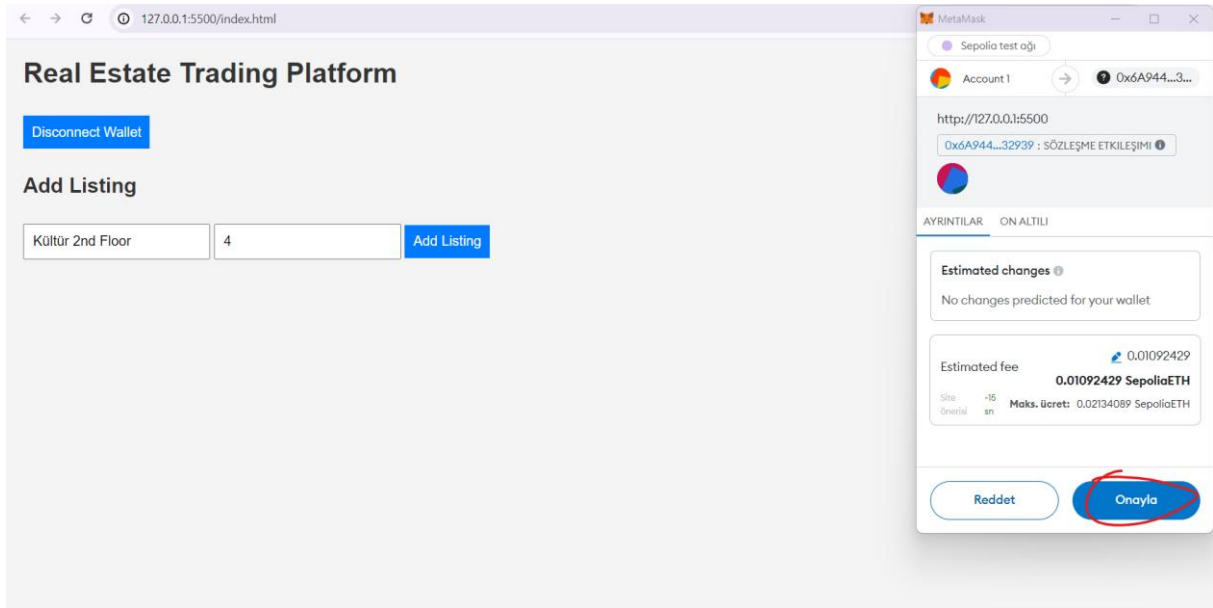
```
index.html # styles.css JS app.js X
JS app.js > submitProperty

1  const web3 = new Web3(window.ethereum);
2
3  let contract;
4  const contractAddress = '0x6A94452D67d706f96AC8c56078b0a0046Fc32939';
5  const contractABI = [
6      {
7          "inputs": [
8              {
9                  "internalType": "uint256",
10                 "name": "_propertyId",
11                 "type": "uint256"
12             },
13             {
```

The copied information is added to the js code as seen.

## 5. Adding and Approving Real Estate Listings in the Interface





When adding a real estate listing, the listing name and price information is entered and the Add Listing button is clicked, followed by a confirmation notification to Metamask. Once approved, the real estate advert is added successfully.

## 6. Bidding on Real Estate Listings

**Real Estate Trading Platform**

Disconnect Wallet

**Add Listing**

Kültür Duplex 89 Add Listing

Listing: Kültür Duplex | [View on Etherscan](#)

Price: 89 ETH

0.001 Place Bid

Listing: Kültür 2nd Floor | [View on Etherscan](#)

Price: 4 ETH

Enter bid (ETH) Place Bid

**MetaMask**

Sepolia test ağı

Account 1 0x6A944...3...

0.001 SepoliaETH

AYRINTILAR ON ALTILI

Ağ meşgul. Gaz fiyatları yüksektir ve tahminler daha az doğrudur.

Estimated changes

You send -0.001 SepoliaETH \$2.94

Estimated fee 0.0101946 SepoliaETH

Site -15 sn Maks. ücret: 0.01996375 SepoliaETH

Reddet Onayla

**Real Estate Trading Platform**

Disconnect Wallet

**Add Listing**

Kültür Duplex 89 Add Listing

Listing: Kültür Duplex | [View on Etherscan](#)

Price: 89 ETH

0.001 Place Bid

Offer by 0xC7d5F4544737f7729178A045794aFf15Df641Ed: 0.001 ETH

Accept Reject

Listing: Kültür 2nd Floor | [View on Etherscan](#)

Price: 4 ETH

**MetaMask**

Account 1 0xC7d5...641Ed

0.0141 SepoliaETH

Buy & Sell Gönder Swap Kİsprü Portföy

Tokenler NFT'ler Etkinlik


May 17, 2024

Make Offer -0.001 SepoliaETH Onaylandı

Sözleşme ... -0 SepoliaETH Onaylandı

Sözleşme ... -0 SepoliaETH Onaylandı

At the bottom of the ad, there is a button to enter the bid price and make an offer. After entering the bid, the bid button is clicked and a confirmation notification is sent to Metamask. Once approved, the bid is placed and the wallet address of the bidder appears below the ad. The owner of the ad can accept or reject the ad offer by pressing the accept and reject buttons.


**Contract** 0x6A94452D67d706f96AC8c56078b0a0046Fc32939

Source Code

**Overview**

ETH BALANCE

0.001 ETH


**More Info**

CONTRACT

0xCf7d5F

The bid amount in the ad is sent to the contract address. If the owner of the real estate ad accepts it, it is transferred from the contract address to the ad owner; if the owner rejects it, it is sent back to the bidder.

## 7. Conclusion


**Address** 0xCf7d5F4544737f7729176A045794aFf15Df641Ed

**Overview**

ETH BALANCE

0.014090326563588132 ETH

**More Info**

LAST TXN SENT

0xe23a88df1bf... from 44 secs ago

FIRST TXN SENT

0xca0d95e6cb... from 5 hrs ago

**Multichain Info**

N/A

Transactions

Token Transfers (ERC-20)

Latest 25 from a total of 35 transactions
 

Download P

Transaction Hash	Method	Block	Age	From	To	Value
0xe23a88df1bf...	Make Offer	5918423	44 secs ago	0xCf7d5F45...15Df641Ed	0x6A94452D...46Fc32939	0.001 ETH
0xa29b3d2845...	0xf54e1aef	5918419	1 min ago	0xCf7d5F45...15Df641Ed	0x6A94452D...46Fc32939	0 ETH
0xe8d14bca41...	0xf54e1aef	5918410	3 mins ago	0xCf7d5F45...15Df641Ed	0x6A94452D...46Fc32939	0 ETH
0x657c994da3...	0x60806040	5918378	9 mins ago	0xCf7d5F45...15Df641Ed	Contract Creation	0 ETH

Transactions can be viewed at sepolia etherscan.

<https://sepolia.etherscan.io/address/0xcf7d5f4544737f7729176a045794aff15df641ed>

From bottom to top, the contract was created, the first real estate ad was made, the second real estate ad was made, the real estate bid was made, and there are transaction histories.

Contract **0x6A94452D67d706f96AC8c56078b0a0046Fc32939**

Source Code

**Overview**  
ETH BALANCE  
0.001 ETH

**More Info**  
CONTRACT CREATOR  
0xCf7d5F45...15Df641Ed at txn 0x657c994da3...

**Multichain Info**  
N/A

Transactions Token Transfers (ERC-20) **Contract** Events

**Code** Read Contract Write Contract

✓ **Contract Source Code Verified** (Exact Match)

Contract Name:	<b>RealEstateMarket</b>	Optimization Enabled:	<b>No</b> with 200 runs
Compiler Version	<b>v0.8.0+commit.c7dfd78e</b>	Other Settings:	<b>default</b> evmVersion

**Contract Source Code** (Solidity)

```
1- /**
2-  *Submitted for verification at Etherscan.io on 2024-05-17
3-  */
4-
5- // SPDX-License-Identifier: MIT
6- // Created by Anil, Emre, Alim
7- pragma solidity ^0.8.0;
8-
9- contract RealEstateMarket {
10-     struct Property {
11-         address payable owner;
```

The contract can be viewed at sepolia etherscan.

<https://sepolia.etherscan.io/address/0x6a94452d67d706f96ac8c56078b0a0046fc32939#code>