

The background of the cover features a collage of two photographs. The top photograph shows the interior of a historic building with high ceilings, brick walls, and a wooden beam roof. The bottom photograph shows a modern library or study area with brick walls, string lights, bookshelves, and people working at tables.

MÁSTER

COMPUTACIÓN EN LA NUBE PARA ENTORNOS INDUSTRIALES

TRABAJO FINAL

2025-2026

Versión 1.0

Dr. Rafael Pastor Vargas
Dr. Agustín C. Caminero Herráez — Dr. Antonio Robles Gómez

MÁSTER UNIVERSITARIO EN INDUSTRIA
CONECTADA

Contenido

Introducción2

Ejercicio 1: Conocimiento del entorno de desarrollo.3

Ejercicio 2: Idea de investigación con Blockchain9

Introducción

En este documento se presenta el Trabajo Final (TF) de la asignatura "COMPUTACIÓN EN LA NUBE PARA ENTORNOS INDUSTRIALES", del "MÁSTER UNIVERSITARIO EN INDUSTRIA CONECTADA" de la UNED.

Este trabajo se realiza de forma individual. En las siguientes secciones se exponen los diferentes ejercicios que es necesario implementar para este trabajo.

Se proponen dos partes diferentes (ejercicio 1 y 2). El primer ejercicio consiste en familiarizarse con el IDE de desarrollo de contratos inteligentes REMIX (<https://remix.ethereum.org/>). El segundo ejercicio consiste en la implementación de un Smart Contract en el contexto de un trabajo de investigación.

El ejercicio 1 se valorará con un máximo de 2 puntos sobre 10, mientras que el ejercicio 2 valorará con un máximo de 8 puntos sobre 10.

La forma de evaluar el trabajo se hará en base a lo siguiente:

- Código realizado, listo para ser compilado/desplegado en el entorno de desarrollo. Se debe incluir no solo el código, sino también las explicaciones necesarias, imágenes, etc.
- Memoria explicativa. Esta memoria deberá añadir las explicaciones oportunas para cada ejercicio y aportar contenido adicional significativo (referencias, documentos adicionales, reflexiones, etc.).

Se valorará positivamente que la memoria contenga un apartado final en el que se explique la opinión del/la estudiante sobre este trabajo, los puntos fuertes y/o débiles, las recomendaciones para el futuro, así como una valoración general de este módulo.

Este material deberá incluirse en un fichero comprimido y enviarse a través del curso virtual dentro de los plazos establecidos para su entrega. El nombre de dicho fichero comprimido deberá tener la estructura *TF-ApellidosNombre.zip*, donde *Apellidos* y *Nombre* deben sustituirse por los valores correspondientes para el/la estudiante que realiza el envío (evitar el uso de acentos o símbolos).

A continuación, se detallan los ejercicios a completar.

Ejercicio 1: Conocimiento del entorno de desarrollo

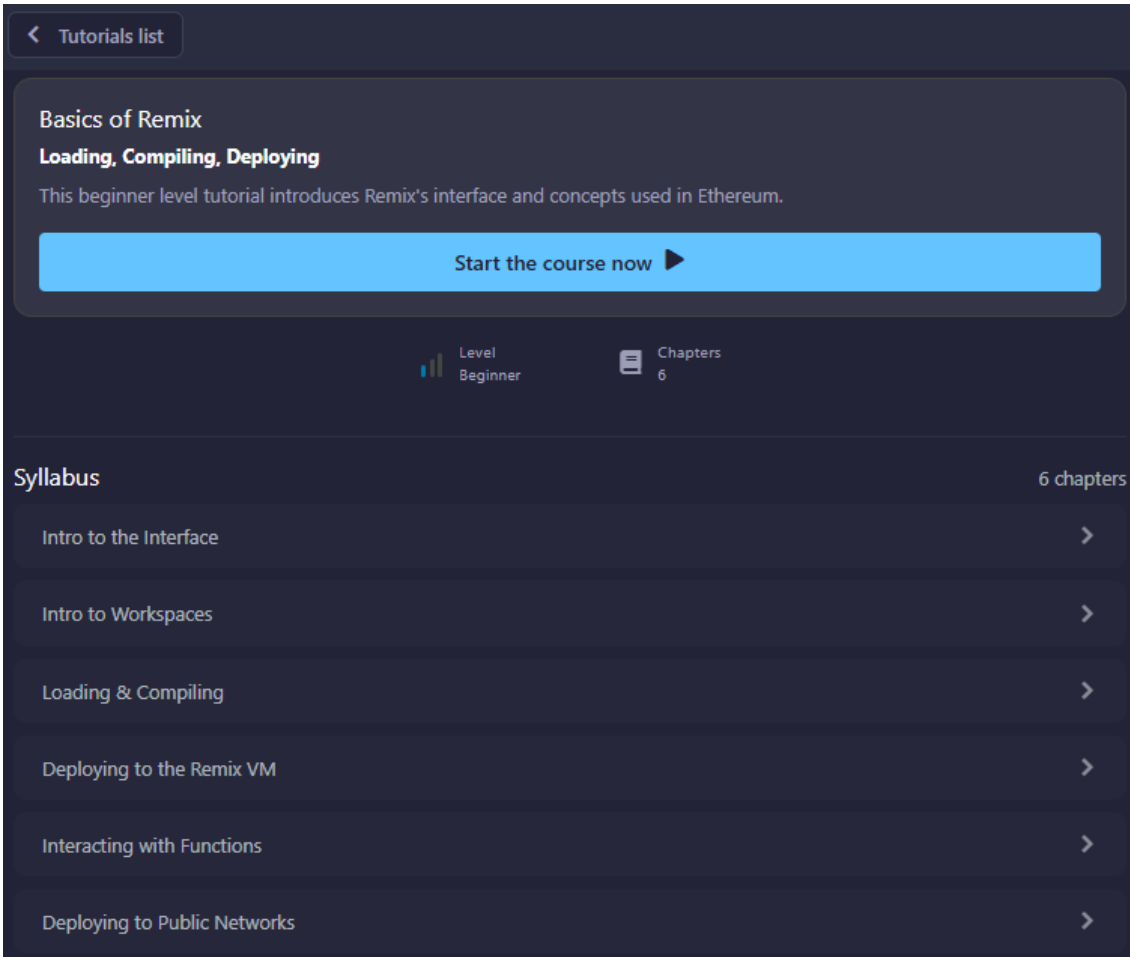
Este ejercicio pretende que el/la estudiante conozca las herramientas de desarrollo necesarias para implementar la idea de investigación sobre la tecnología Blockchain. Es recomendable leer primero la documentación básica del entorno:

<https://remix-ide.readthedocs.io/es/latest/>

El ejercicio consiste en desarrollar el tutorial básico de uso de la herramienta REMIX. Para ello se debe cargar la siguiente URL:

<https://remix.ethereum.org/?#activate=udapp,solidity,LearnEth>

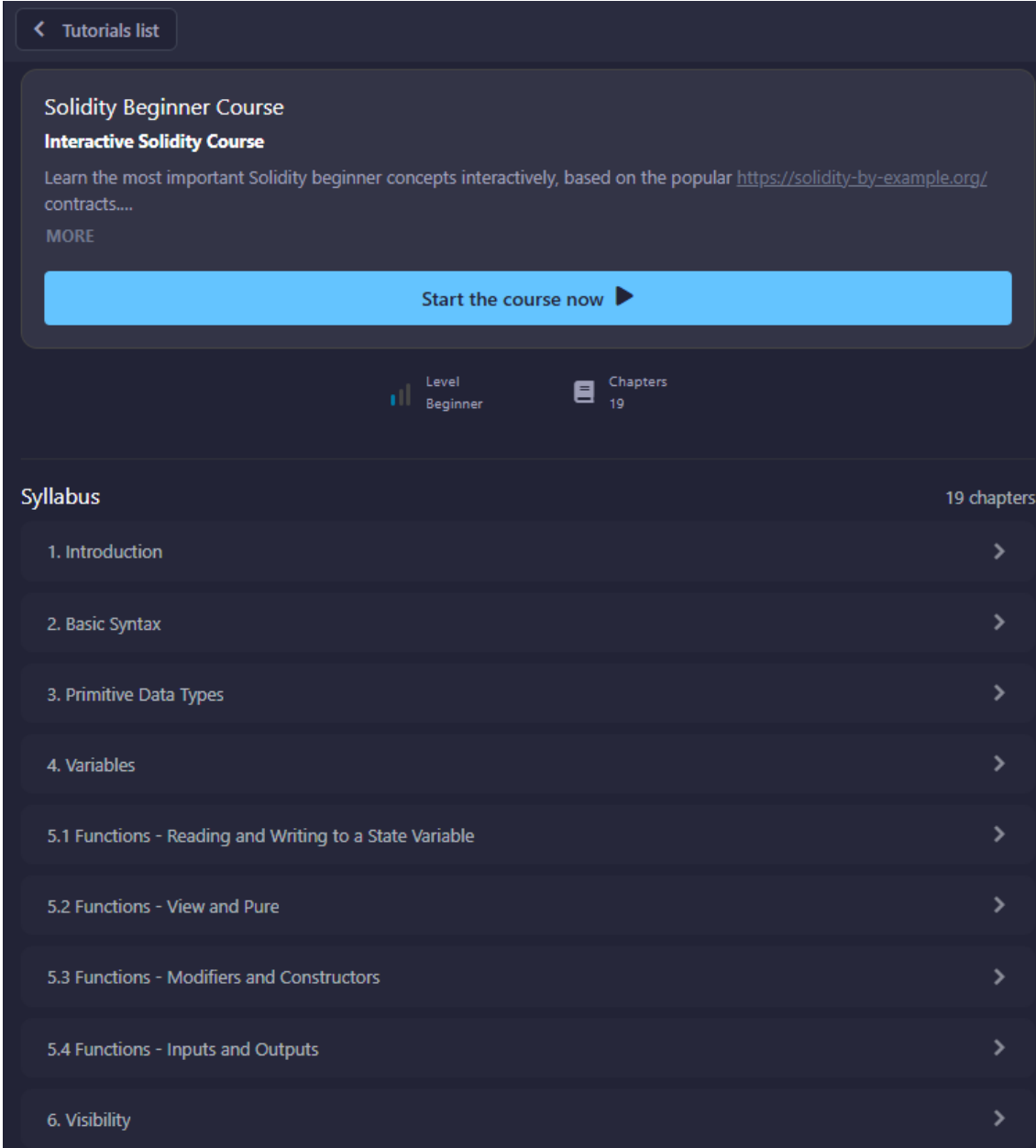
y seguir los pasos del tutorial básico:



The screenshot shows the 'Tutorials list' page in the Remix IDE. At the top, there is a back arrow and the text 'Tutorials list'. Below this, a section titled 'Basics of Remix' is highlighted. Underneath, the text 'Loading, Compiling, Deploying' is shown, followed by a description: 'This beginner level tutorial introduces Remix's interface and concepts used in Ethereum.' A large blue button with the text 'Start the course now' and a right-pointing arrow is prominently displayed. Below the button, there are two icons: a bar chart icon labeled 'Level Beginner' and a document icon labeled 'Chapters 6'. At the bottom, a 'Syllabus' section is listed with the title '6 chapters' on the right. The syllabus items are: 'Intro to the Interface', 'Intro to Workspaces', 'Loading & Compiling', 'Deploying to the Remix VM', 'Interacting with Functions', and 'Deploying to Public Networks'. Each item has a right-pointing arrow next to it.

Syllabus		6 chapters
Intro to the Interface	>	
Intro to Workspaces	>	
Loading & Compiling	>	
Deploying to the Remix VM	>	
Interacting with Functions	>	
Deploying to Public Networks	>	

Se recomienda hacer el segundo tutorial, para facilitar el desarrollo del Smart Contract requerido en la segunda parte y conocer los fundamentos del lenguaje Solidity.



[← Tutorials list](#)

Solidity Beginner Course

Interactive Solidity Course

Learn the most important Solidity beginner concepts interactively, based on the popular <https://solidity-by-example.org/contracts....>

MORE

[Start the course now ▶](#)

Level: Beginner

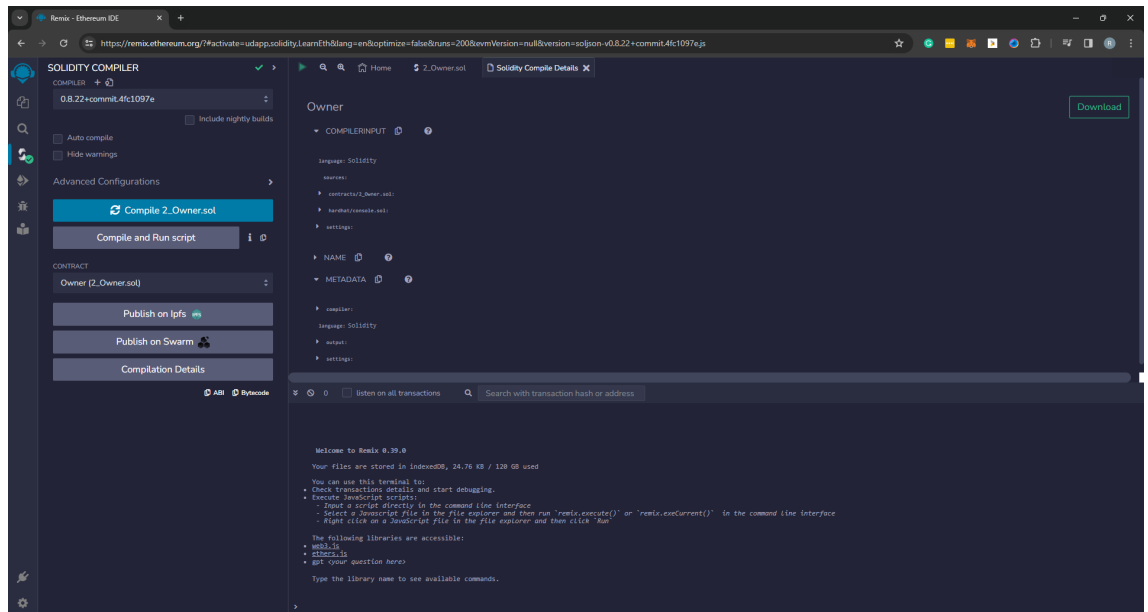
Chapters: 19

Syllabus 19 chapters

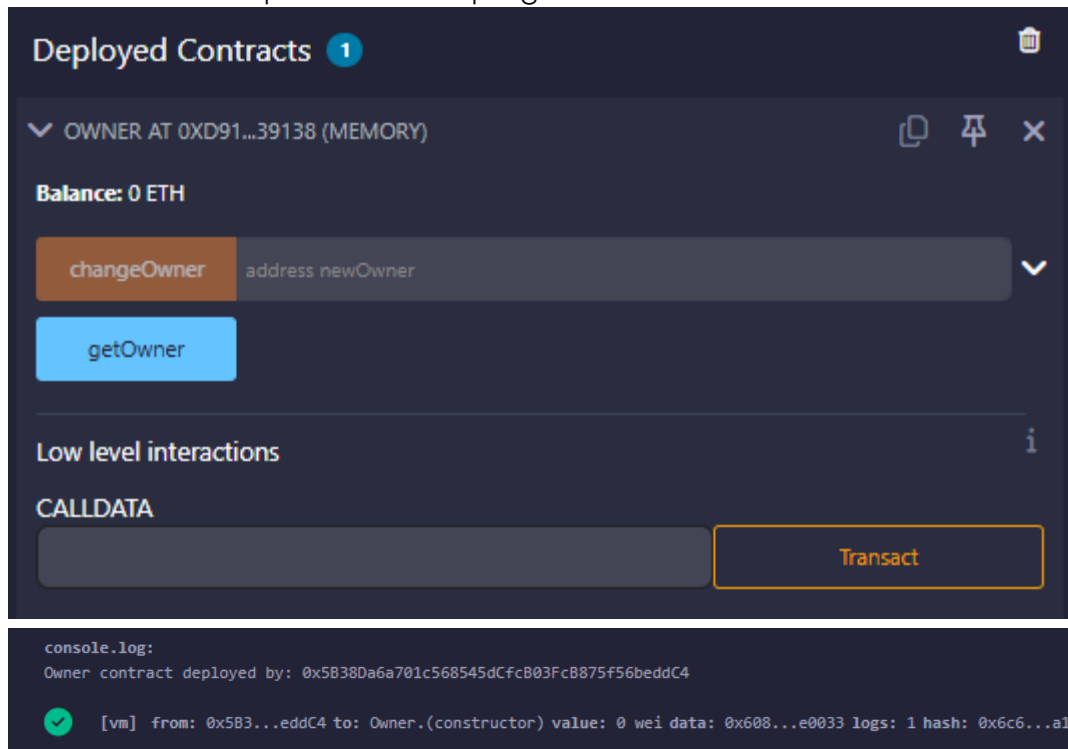
- 1. Introduction
- 2. Basic Syntax
- 3. Primitive Data Types
- 4. Variables
- 5.1 Functions - Reading and Writing to a State Variable
- 5.2 Functions - View and Pure
- 5.3 Functions - Modifiers and Constructors
- 5.4 Functions - Inputs and Outputs
- 6. Visibility

Como evidencias, se requiere que la memoria del ejercicio incluya, al menos, las siguientes (siguiendo los pasos del tutorial):

- 1) Pantallazo de la prueba de compilación del contrato referido en el tutorial.



2) Pantallazos con la prueba del despliegue del contrato.



3) Instalación de Metamask y verificación de la disponibilidad de Ether, necesarias para desplegar el contrato en una red de prueba. Se recomienda leer el siguiente enlace para comprender que es Metamask y que ofrece:

<https://www.binance.com/es/academy/articles/how-to-use-metamask>

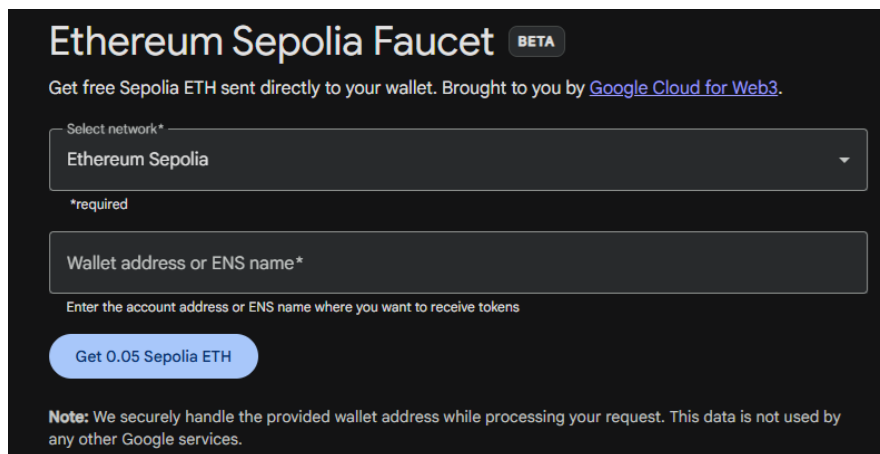
<https://www.learningheroes.com/blog/aprende-crypto/que-es-metamask-wallet-tu-cartera-crypto-segura>

<https://support.metamask.io/es/start/getting-started-with-metamask/>

Se va a emplear la red de prueba Sepolia de Ethereum para implementar el despliegue de contrato, por lo que es necesario que se use un “grifo” (faucet) de tokens para esta red (SepoliaETH). Este faucet se puede usar una vez al día (por cada red que está disponible):

<https://cloud.google.com/application/web3/faucet>

Seleccione la red Ethereum Sepolia e introduzca su número de wallet en la dirección:



Debe comprobar que se han transferido los tokens a su cuenta de metamask mediante la url de la transacción (que se debe añadir en el informe de esta parte):

[This is a Sepolia Testnet transaction only]

Transaction Hash:	0x960d9c210b9acbbcc1f33a4d4a1ba7a7be401b259f36adcbcabaa01aa1adaf95
Status:	Success
Block:	9601833 89 Block Confirmations
Timestamp:	18 mins ago (Nov-10-2025 07:32:12 PM UTC)
From:	
To:	
Value:	0.05 ETH
Transaction Fee:	0.000000023099979 ETH
Gas Price:	0.001099999 Gwei (0.000000000001099999 ETH)

- Url de la transacción del despliegue del contrato en la red de prueba (Sepolia). También se debe añadir un pantallazo de dicha transacción.

Implementación de contrato

Estado

Confirmado

Ver en el explorador de bloques

Copiar ID de transacción

De

Para

0xdEde6...27978

Contrato nuevo

Transacción

Nonce

0

Importe

-0 SepoliaETH

Límite De Gas (Unidades)

348359

Gas Usado (Unidades)

344458

Tarifa base (GWEI)

0.00000001

Tarifa de prioridad (GWEI)

1.5

Tarifa total de gas

0.000517 SepoliaETH

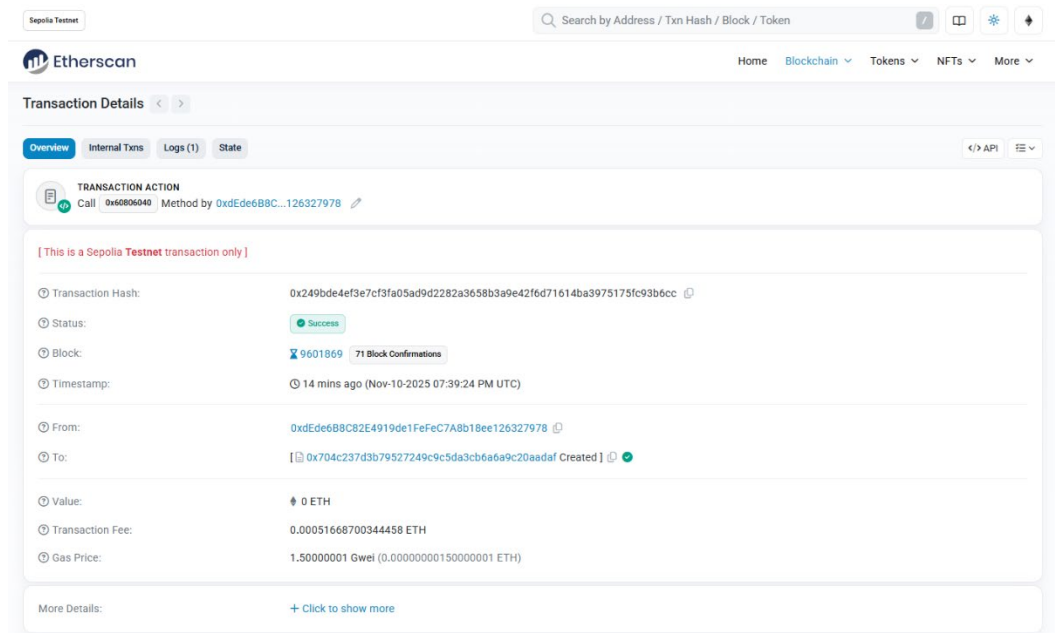
Tarifa máxima por gas

0.000000002 SepoliaETH

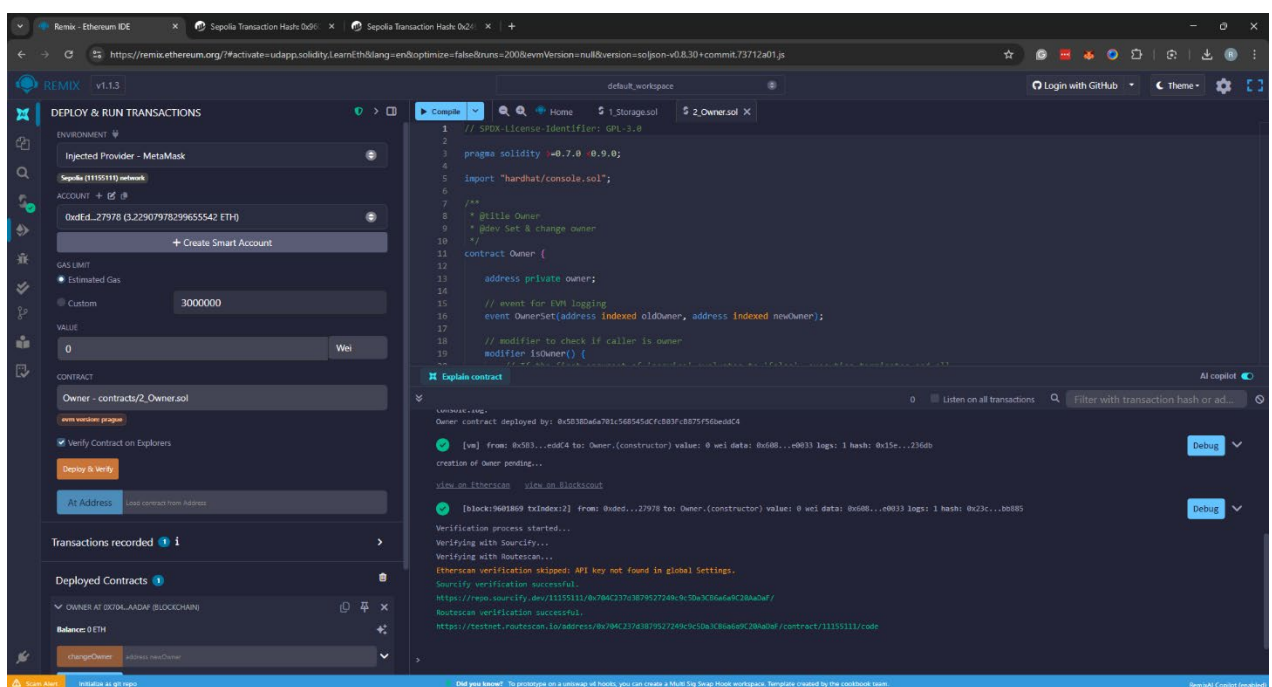
Total

0.00051669 SepoliaETH

<https://sepolia.etherscan.io/tx/0x249bde4ef3e7cf3fa05ad9d2282a3658b3a9e42f6d71614ba3975175fc93b6cc>



5) Pantallazo del contrato desplegado en la red de prueba de Sepolia.



Entregables del ejercicio:

- 1) Evidencias del tutorial (al menos las que se mostraron anteriormente).
- 2) Memoria del ejercicio 1: problemas identificados, impresiones personales y mejoras a la propuesta. Incluirá también las evidencias del punto 1) y se denominará memoria_ejercicio1 (docx/pdf).

Ejercicio 2: Idea de investigación con Blockchain

Desarrollar un contrato inteligente sobre una posible idea de investigación que permita implementarse con Blockchain. En la memoria del ejercicio 2 se debe plantear la idea a desarrollar, explicar por qué se justifica el uso de Blockchain y Smart Contract, y qué aporta de novedad, innovación o investigación en el área de aplicación.

A modo de ejemplo, y como referencia, se puede implementar la gestión de autoconsumo y producción de una pequeña planta fotovoltaica instalada en un hogar particular. El objetivo es almacenar esta información de manera segura en una cadena de Blockchain para ser usada después con fines analíticos y poder desarrollar algoritmos de tipo Machine Learning.

Se deben definir las funciones del contrato inteligente y describirlas en la memoria. En el caso de referencia, estas funciones serían las de almacenamiento de la producción de la planta, el consumo total de la instalación (autoconsumo + consumo de la red eléctrica) y el autoconsumo realizado en la casa. Para cada operación de almacenamiento se añadirán el valor en kW y las fechas de contabilización (inicio y final). Además de estas funciones, se deberán implementar las funciones de contabilidad de la información almacenada en la Blockchain: consumo total/autoconsumo/producción por días (últimos N días) y entre dos fechas determinadas.

Se debe desarrollar el Smart Contract en el entorno REMIX IDE, compilar y desplegar el contrato, y elaborar un plan de pruebas adecuado a la idea desarrollada. Se deberán aportar evidencias de la ejecución de este plan de pruebas (pantallazos y transacciones en la red blockchain de destino).

Entregables del ejercicio:

- 1) Fichero Solidity con el código del contrato
- 2) Pantallazos de REMIX IDE mostrando:
 - a. El despliegue del contrato.
 - b. Invocación de las operaciones del contrato, en especial aquellas que consuman "gas".
- 3) Memoria del ejercicio 2: idea de investigación, plan de pruebas, evidencias. La memoria debe incluir los pantallazos del punto 2) también, y se denominará memoria_ejercicio2 (docx/pdf).