

Bootcamp para Desarrolladores de Smart Contracts

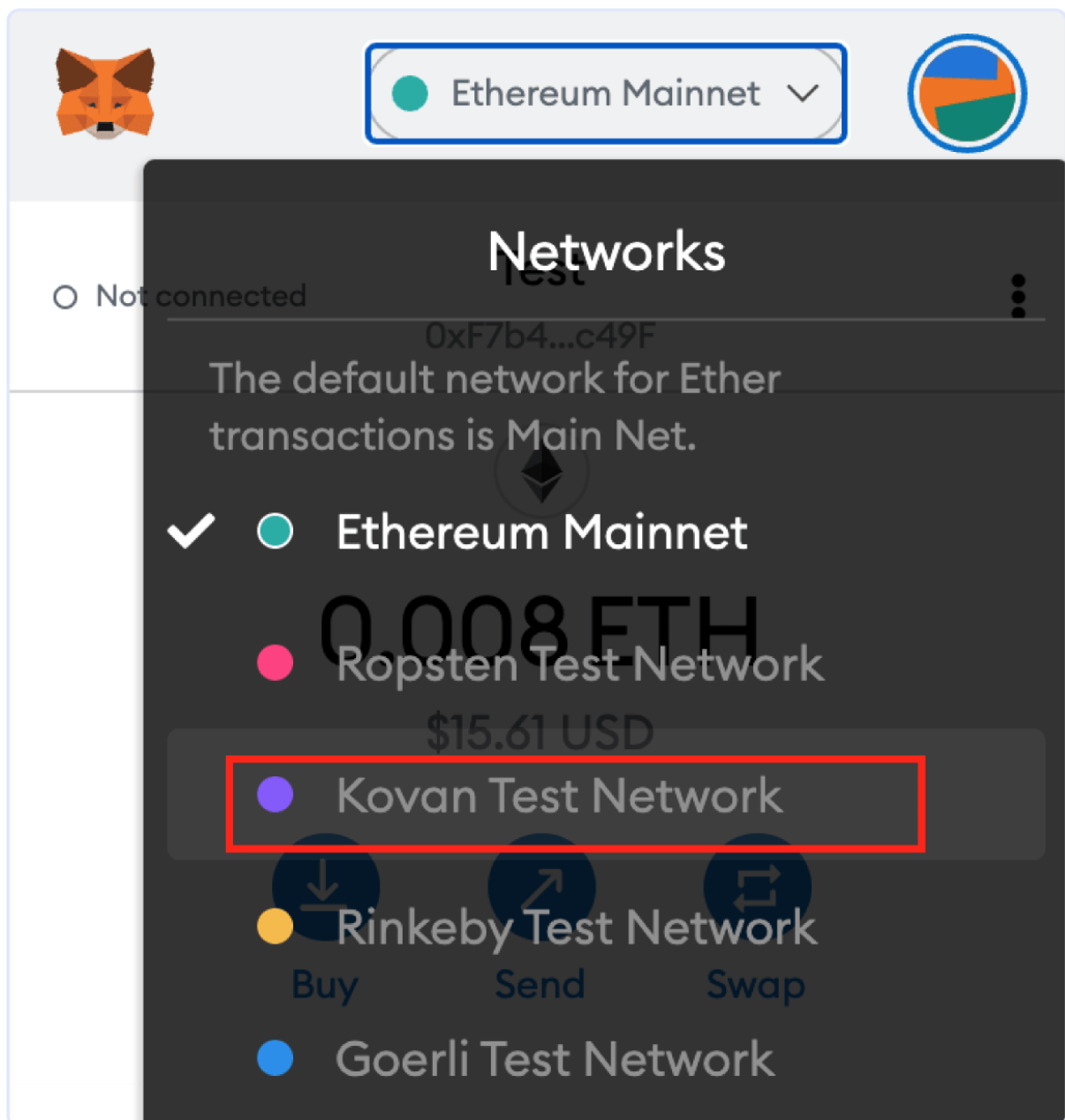
Instrucciones de Instalación: Brownie Track

Instalación del Software

Por favor, complete los siguientes pasos para instalar el software necesario para el Developer Bootcamp. Puede omitir los pasos para el software que ya tiene instalado si no hay una versión mínima indicada; de lo contrario, asegúrese de que su versión es igual o superior a la versión mínima indicada del software. Si su versión es inferior a la versión mínima indicada, actualice a la última versión según las instrucciones que se indican a continuación.

METAMASK

1. Descarga e instala MetaMask, una extensión del navegador que utilizarás como cartera para interactuar con tus contratos inteligentes y la red Ethereum. MetaMask requiere el navegador Google Chrome o Brave, y también tiene soporte limitado para Firefox y Opera. Recomendamos Chrome o Brave.
2. Abre tu extensión de navegador MetaMask (busca el símbolo del zorro en tus extensiones en la parte superior derecha del navegador), y configura tu cuenta.
3. Una vez que tu cuenta esté configurada, cambia la red Ethereum a la que MetaMask está conectada, seleccionando el menú desplegable en la parte superior, y cambiándolo a "Kovan Test Network".



NODE.JS

Versión mínima requerida : 12.0.0

Versión Mínima de Windows: 10

Versión Mínima de macOS: 10

Puedes omitir este paso si ya tienes una instalación de Node.js 12.0 o superior en funcionamiento. Para comprobarlo, puedes abrir una nueva Terminal (u otra CLI de tu elección), escribe el comando que aparece a continuación y pulsa enter. Para los usuarios de Windows, el Símbolo del sistema se puede encontrar pulsando el botón de inicio de Windows y buscando "Símbolo del sistema". Para los usuarios de Mac, la terminal se puede encontrar en Aplicaciones.

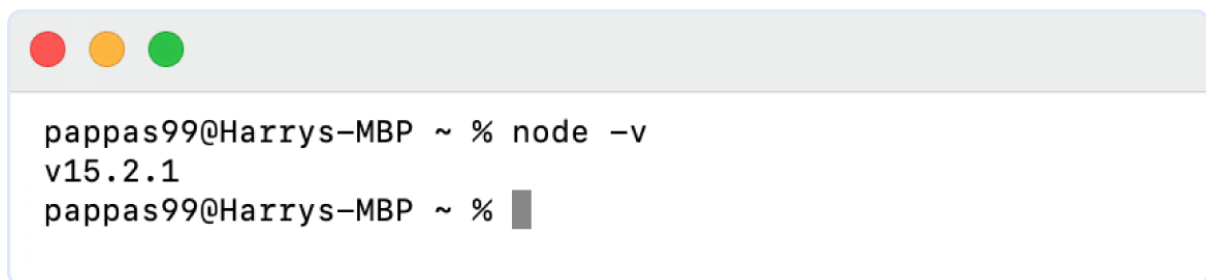
```
node -v
```

Si obtienes una versión reportada (según la captura de pantalla más abajo), ya tienes Node.js instalado, y puedes saltarte esta sección si cumple con los requisitos mínimos. De lo contrario, si obtienes un error, significa que no tienes Node.js instalado, y debes continuar estos pasos para descargarlo e instalarlo. Si tienes una versión de Node.js instalada, pero es inferior a la 12, sigue [estas instrucciones](#) para actualizarlo.

Si no tiene instalado Node.js, descargue e instale la última versión del entorno de ejecución de JavaScript [Node.js and package manager NPM](#). Este paso es necesario tanto para las sesiones de JavaScript como de Python del bootcamp. Acepte todas las respuestas predeterminadas para las preguntas.

Una vez que haya completado la instalación, abra una nueva Terminal (u otra CLI de su elección), escriba el comando que aparece a continuación para asegurarse de que la instalación se ha realizado correctamente. Para los usuarios de Windows, el símbolo del sistema se puede encontrar pulsando el botón de inicio de Windows y buscando "símbolo del sistema". Para los usuarios de Mac, el terminal se puede encontrar en Aplicaciones.

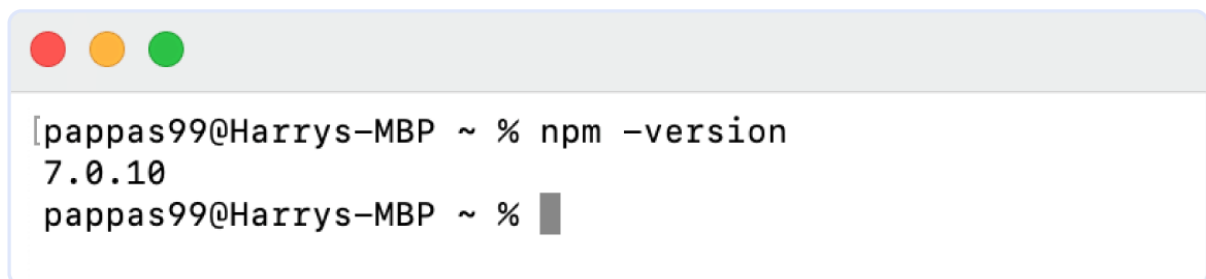
```
node -v
```



```
pappas99@Harrys-MBP ~ % node -v
v15.2.1
pappas99@Harrys-MBP ~ %
```

Una vez que hayas verificado tu instalación de Node.js, verifica también tu instalación de NPM introduciendo el siguiente comando:

```
npm -version
```



```
[pappas99@Harrys-MBP ~ % npm -version
7.0.10
pappas99@Harrys-MBP ~ %
```

GIT

Descarga e instala el sistema de control de versiones Git. Lo necesitaremos cuando instalemos algunos paquetes, y también para revisar tu código en un repositorio cuando hayas terminado. Acepta todos los valores por defecto durante la instalación. Si utilizas macOS y se te pide que instales "herramientas de desarrollo de línea de comandos", acepta e instálalas.

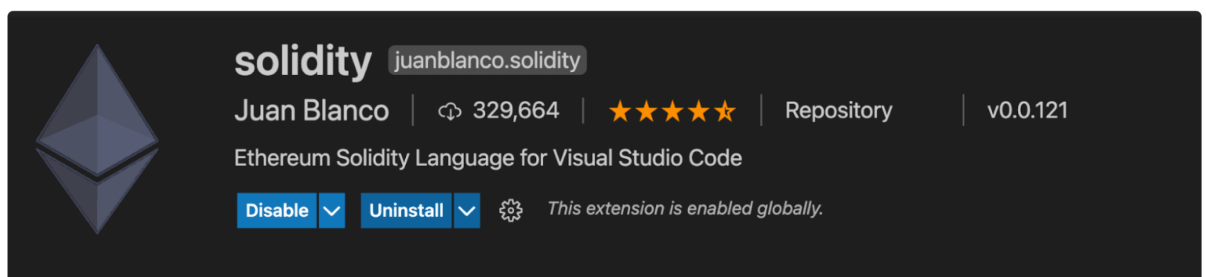
Para probar git una vez instalado, abre un nuevo Terminal (u otro CLI de tu elección), y escribe el siguiente comando:

```
git --version
```

VISUAL STUDIO CODE

Necesitarás un editor de texto/archivos para los días 2 y 3 del bootcamp. Recomendamos Visual Studio Code, uno de los editores gratuitos de código abierto más populares para el desarrollo de contratos inteligentes. El uso de Visual Studio Code le facilitará el seguimiento de las capturas de pantalla de los ejercicios, y el terminal integrado hace que sea más fácil cambiar entre la edición de archivos y la ejecución de comandos. Si tienes tu propio editor con el que te sientes cómodo, puedes saltarte este paso.

1. Vaya a <https://code.visualstudio.com/>, descargue e instale la última versión de Visual Studio Code para su sistema operativo.
2. Abra Visual Studio Code. En el menú superior, seleccione Code (File si utiliza Windows) > Preferences > Extensions
3. Busque la palabra 'solidity', luego seleccione e instale la extensión Solidity. Esto le permitirá resaltar la sintaxis cuando desarrolle sus contratos inteligentes



4. Habilita el terminal seleccionando View -> Terminal en el menú superior (o el atajo del teclado Ctrl + ñ). Asegúrese de que puedes ejecutar los comandos de Node.js en tu ventana de terminal ejecutando el siguiente comando:

```
node -v
```

```
PROBLEMS    OUTPUT    DEBUG CONSOLE    TERMINAL

pappas99@Harrys-MBP chainlink-hardhat-box-1 % node -v
v15.2.1
pappas99@Harrys-MBP chainlink-hardhat-box-1 % █
```

Si estás usando Windows y obtienes un error, asegúrate de que estás usando tu terminal CMD

- En VS Code, pulse CTRL + SHIFT + P
- Buscar 'Terminal Select Default Profile'
- Seleccione la opción 'Command Prompt' o 'Símbolo de Sistema'
- Reinicie VS Code e intente ejecutar el comando de nuevo en el terminal

PYTHON

Descargue e instale la última versión de [Python](#) para su sistema operativo.

Una vez que haya completado la instalación, abra una nueva Terminal (u otra CLI de su elección), y escriba el siguiente comando para asegurarse de que la instalación se ha realizado correctamente:

```
python --version
python3 --version
```

```
pappas99@Harrys-MBP ~ % python3 --version
Python 3.6.8
pappas99@Harrys-MBP ~ % █
```

CONFIGURACIÓN DE PIPX

Durante los días 2 y 3 del bootcamp, trabajarás con el Entorno de Desarrollo Brownie. Intentaremos configurarlo con antelación para ahorrar tiempo en la instalación durante el bootcamp. Pipx es una herramienta que te ayuda a instalar y ejecutar aplicaciones de usuario final escritas en Python. Es más o menos similar a brew de macOS, npx de JavaScript, y apt de Linux.

1. Una vez instalado python, abre una nueva Terminal (u otra CLI de tu elección), y escribe el siguiente comando para instalar pipx. Si estás usando Windows, asegúrate de ejecutar el Símbolo del sistema como administrador.

Sugerencia para la solución de problemas: Si este comando no funciona al

principio, pruebe con python en lugar de python3

```
python3 -m pip install --user pipx
```

```
python3 -m pipx ensurepath
```

Si tiene éxito aquí, puede pasar a “Instalar Brownie”

Solución de problemas para usuarios de MacOS

Si está utilizando macOS y obtiene un error similar a este: *‘invalid active developer path (/Library/Developer/CommandLineTools), missing xcrun at: /Library/Developer/CommandLineTools/usr/bin/xcrun’*, ejecute el siguiente comando para descargar e instalar las herramientas de línea de comandos de apple developer, luego cierre (escriba exit) y vuelva a abrir su ventana de terminal e intente el comando pipx de nuevo

```
xcode-select --install
```

Solución de problemas para los usuarios de Windows

Si obtiene un error con algo como:

```
error: Microsoft Visual C++ 14.0 is required. Get it with  
"Microsoft Visual C++ Build Tools"
```

Tendrás que hacer lo siguiente:

1. Descarga e instala las [Build Tools for Visual Studio](#), y luego instala las `C++ build tools`. Una vez instalado, ejecuta el siguiente comando en tu terminal. [Aquí hay un video tutorial que muestra cómo instalar las herramientas de compilación.](#)

```
pip install --upgrade setuptools
```

INSTALAR BROWNIE

1. Cierra tu terminal (teclea `exit`, luego presiona enter), vuelve a abrir una nueva terminal, e introduce el siguiente comando:

```
pipx install eth-brownie
```

Deberías ver un output que se parece aproximadamente a:

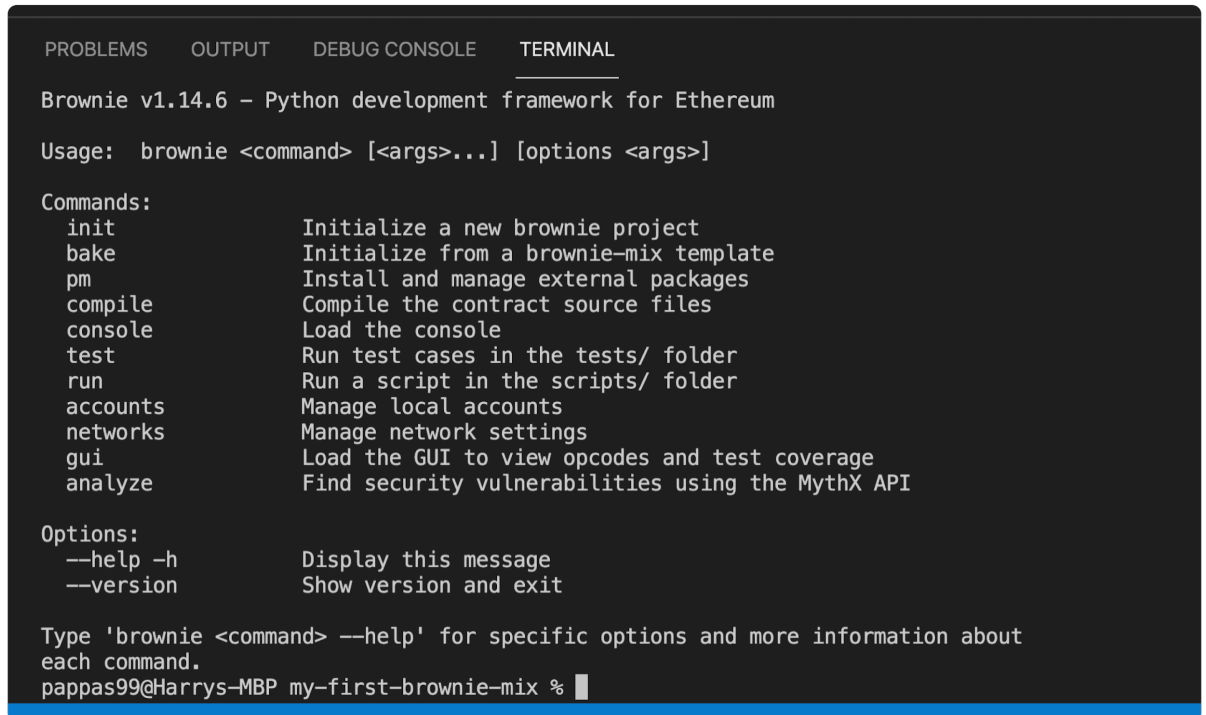
```
patrick@iMac: [~] $ pipx install eth-brownie  
installed package eth-brownie 1.14.6, Python 3.8.0  
These apps are now globally available  
- brownie  
done! ✨ ✨ ✨  
patrick@iMac: [~] $
```

Si todavía te da errores, puedes intentar instalar eth-brownie usando pip en lugar de pipx, de lo contrario, si has instalado con éxito eth-brownie, salta al siguiente paso

```
pip install eth-brownie
```

2. Una vez completada la instalación, escribe `brownie` para verificar que se ha instalado con éxito. Deberías ver una salida similar a la captura de pantalla de abajo.

```
brownie
```



```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL

Brownie v1.14.6 - Python development framework for Ethereum

Usage: brownie <command> [<args>...] [options <args>]

Commands:
  init           Initialize a new brownie project
  bake           Initialize from a brownie-mix template
  pm             Install and manage external packages
  compile        Compile the contract source files
  console        Load the console
  test           Run test cases in the tests/ folder
  run            Run a script in the scripts/ folder
  accounts       Manage local accounts
  networks       Manage network settings
  gui            Load the GUI to view opcodes and test coverage
  analyze        Find security vulnerabilities using the MythX API

Options:
  --help -h      Display this message
  --version      Show version and exit

Type 'brownie <command> --help' for specific options and more information about
each command.
pappas99@Harrys-MBP my-first-brownie-mix %
```

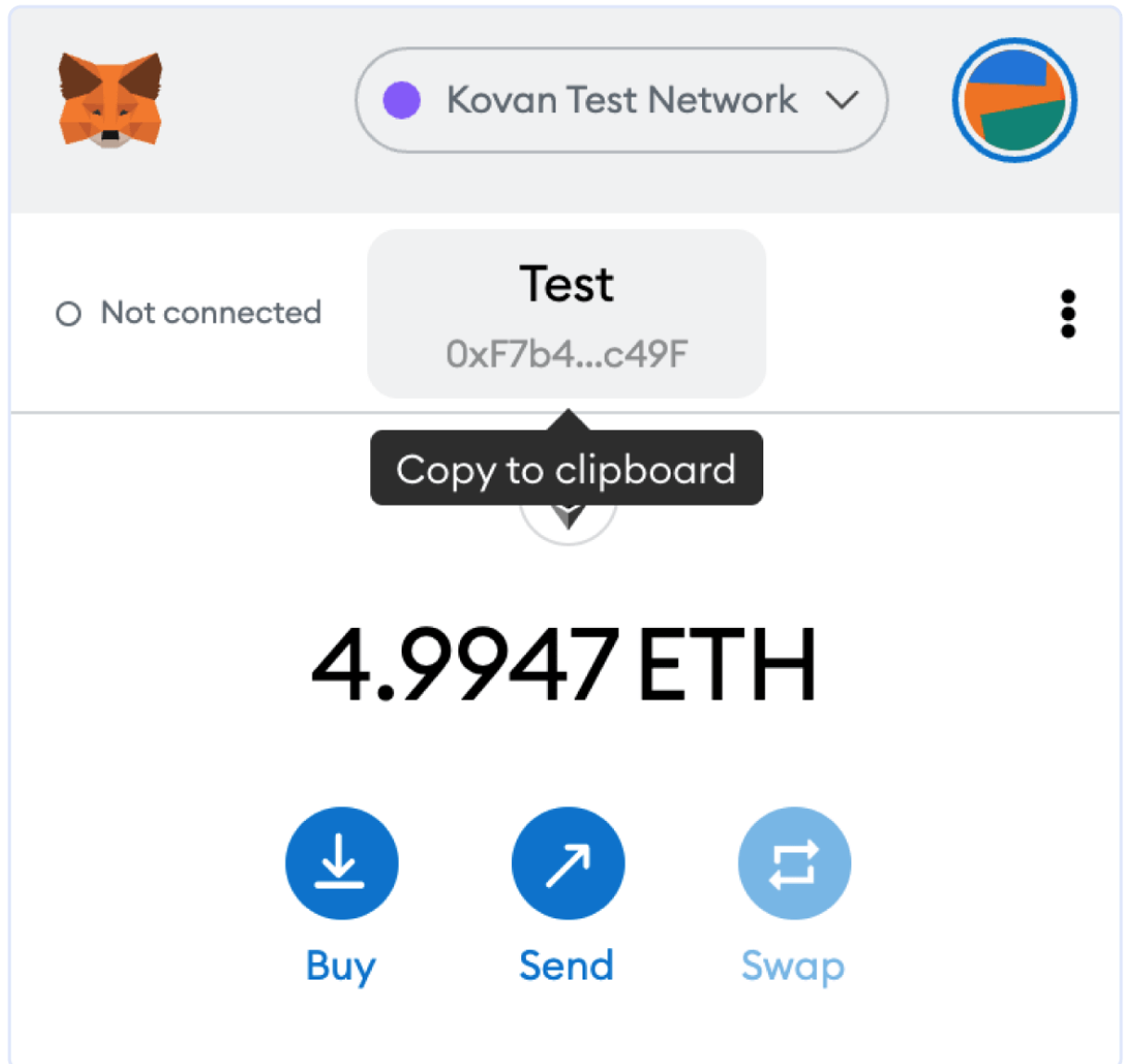
Obtener fondos para tu Cuenta de Metamask

Nota: A los participantes que ya tienen y utilizan MetaMask, les sugerimos encarecidamente que no utilicen ninguna cuenta que tenga fondos en Mainnet. En caso de que accidentalmente registres un código que contenga tu clave privada en un repositorio público, alguien puede utilizarlo para robar tus fondos en Mainnet. Para el bootcamp, utiliza una [dirección limpia recién generada](#) que nunca haya sido utilizada antes en Mainnet.

Si ya has instalado MetaMask, y nos has proporcionado una dirección ETH limpia en el formulario de inscripción del Developer Bootcamp, entonces debes comprobar tu cuenta para ver si ya has recibido algunos ETH de testnet y LINK (tu saldo de ETH será > 0). Si lo has hecho, puedes saltarte este paso. Si has recibido algo de ETH, pero no tienes LINK en tus activos en MetaMask, sigue los pasos 3 y 4 de abajo

para añadir el activo LINK a tu cartera de MetaMask, y entonces deberías poder ver si también has recibido algún LINK o no.

1. Abre MetaMask, mueve el cursor sobre el nombre de tu cuenta y haz clic en él para copiar la dirección de la cartera en el portapapeles



2. Dirígete a <https://linkfaucet.protofire.io/kovan>, pega la dirección de tu wallet y pulsa el botón 'Send me 0.1 Test ETH button'. El faucet te enviará algunos ETH de prueba de Kovan, que deberían aparecer en tu cartera de MetaMask unos segundos después en la sección de Activos.

Chainlink Faucet

Your Wallet address:

KOVAN



0x0BA3e9178C1459DcF70C64C17740fBF81425063f



I'm not a robot



reCAPTCHA
Privacy - Terms

Send me 10 Test LINK

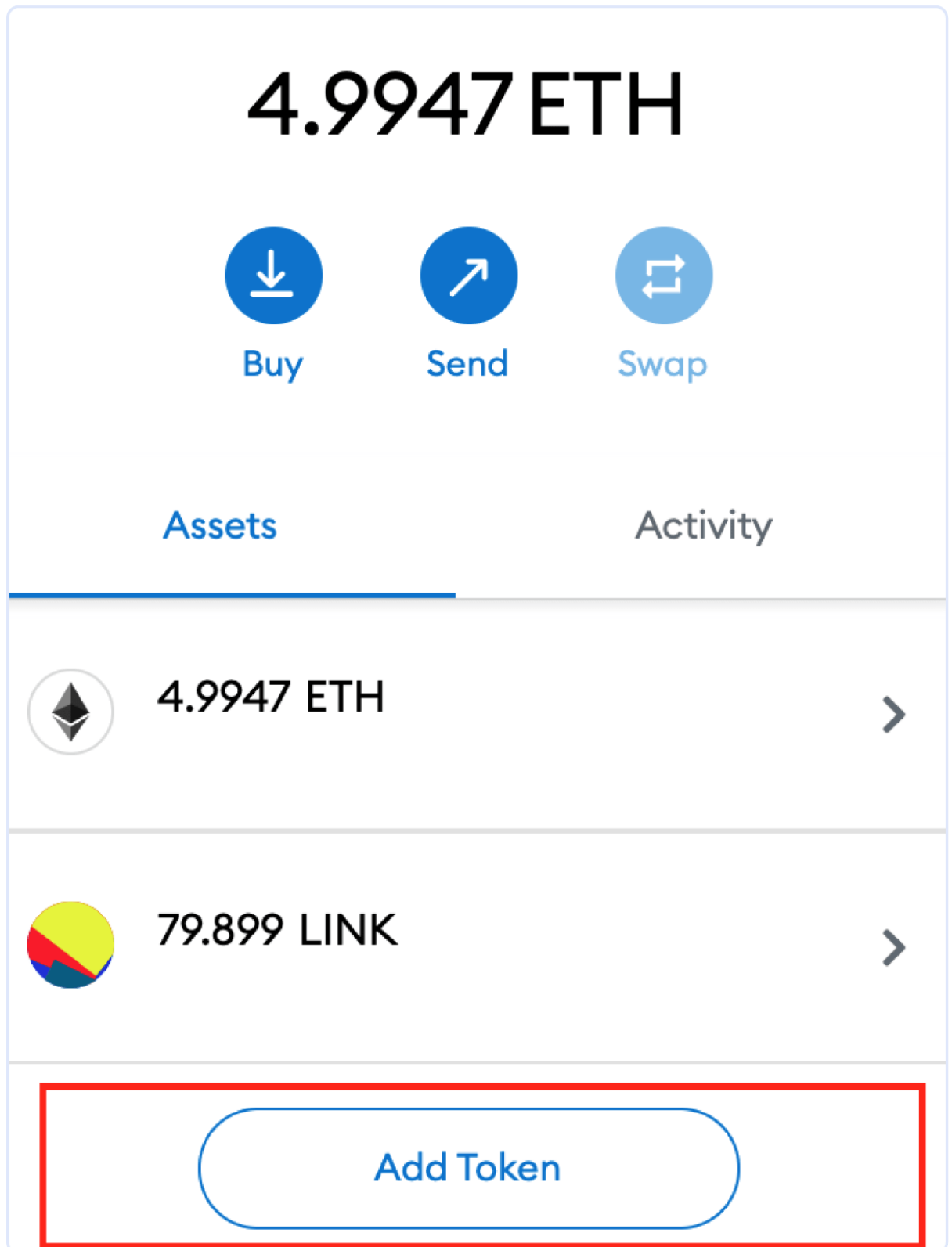


Send me 0.1 Test ETH



Nota: Si el sitio web de Faucet está caído, por favor pega tu dirección ETH en el canal **#bootcamp-español** del [Discord de Chainlink](#), y te enviaremos algo de ETH.

3. Comprueba si tienes el token LINK añadido a la sección "Activos" de tu wallet MetaMask.



4. Si no tienes el token LINK en tus Activos (con 0 o más LINK), pulsa el botón 'Add Token' en la parte inferior, luego introduce la dirección del contrato del token LINK para la red de Kovan

0xa36085F69e2889c224210F603D836748e7dC0088, luego pulsa Siguiente, luego Agregar Token.

5. Vé de vuelta al sitio web del faucet, completa el reCAPTCHA de nuevo, y luego presiona el botón 'Send me 10 Test LINK'. Deberías ver 10 LINK aparecer en tu cuenta de MetaMask unos segundos después en la sección de Activos.

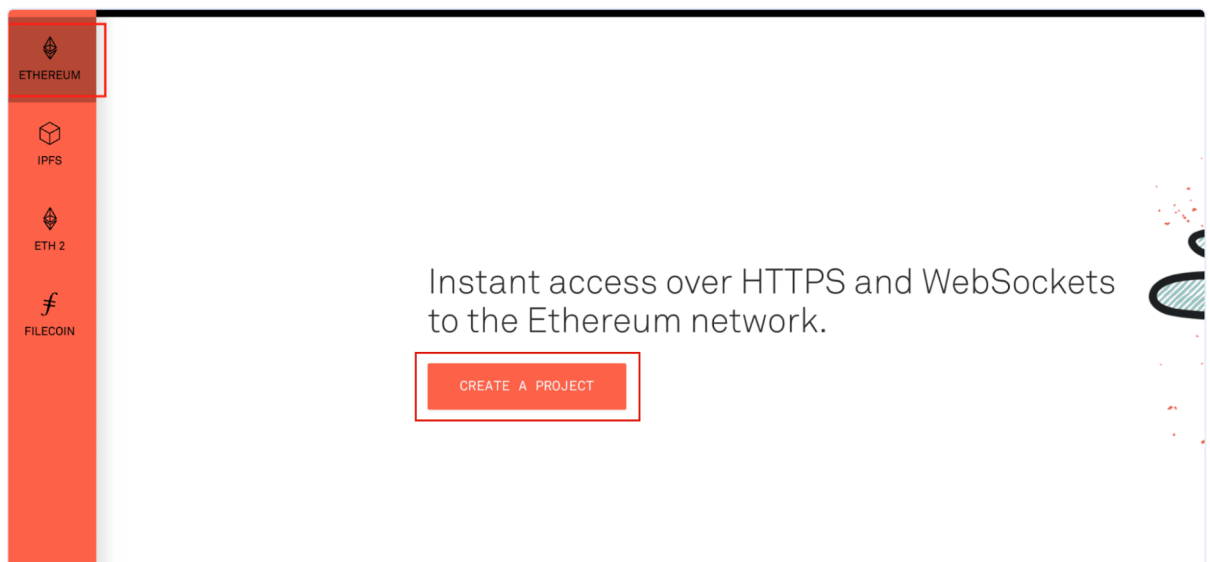
Regístrate en Infura

Para los días 2 y 3 del bootcamp, utilizaremos los Chainlink Starter Kits. Ambos kits requieren claves de acceso de Infura. Puedes obtener claves de prueba gratuitas.

INFURA

Infura proporciona una puerta de acceso a la red Ethereum sin tener que ejecutar su propio nodo Ethereum. Complete los siguientes pasos para obtener su clave gratuita de Infura.

1. Dirígete a <https://infura.io/product/ethereum>, presiona en Sign Up, y rellena tus datos. Una vez que hayas verificado tu dirección de correo electrónico, deberías poder iniciar sesión en el sitio.
2. Una vez iniciada la sesión, selecciona el botón Ethereum en el menú la izquierda, y selecciona 'Create a Project', llámalo 'Developer Bootcamp' y presiona Create.



3. Una vez que estés en tu proyecto, en la configuración del proyecto, cambia el desplegable 'ENDPOINTS' a 'Kovan', copia y toma nota de tu endpoint HTTPS endpoint (guárdalo en algún sitio). Lo necesitarás en los días 2 y 3 del bootcamp.

KEYS

PROJECT ID

PROJECT SECRET ⓘ

ENDPOINTS

Kovan

▼

https://kovan.infura.io/v3/

wss://kovan.infura.io/ws/v3/

¡Felicitaciones, ya estás preparado para el Smart Contract Developer Bootcamp!