

Desarrollo Blockchain Ethereum con Solidity

Módulo 6 - Prueba de contratos

Prueba de contratos

Testing de código

- Nos permite detectar errores en etapas más tempranas de desarrollo y de forma más rápida.
- Facilita los cambios y la integración de componentes.
- Puede funcionar como documentación del código y ejemplos de uso.
- Reduce el tiempo/costo de mantenimiento del proyecto.



Testing de contratos

- Todo **contrato** inteligente desarrollado con Solidity para Ethereum debe de ser **probado**.
- Entre las múltiples ventajas se encuentra la detección temprana de potenciales **incidencias**.
- No importa cuántos contratos haya, deben testearse.
- Puede haber tantos **test y archivos** de test como se **desea**.
- La correcta realización de los test permitirá por ejemplo evitar los casos conocidos de fallas públicas como KingOfEther, TheDao, etc.



Testing de contratos

Testear en JS o en Solidity

- **Truffle** nos va a permitir desarrollar nuestros tests tanto en **Javascript** como en **Solidity**.
- Truffle ha integrado en su solución de testing en **Solidity** una librería de assertions muy similar a la que utilizan Mocha y Chai.
- Es recomendable leer la documentación de testing con **Mocha** en <https://mochajs.org/>
- La aplicación generada en la clase anterior (Metacoin) por defecto incluye algunos test redactado en Solidity y en Javascript.



Testing de contratos

Todo comienza con el unbox de **truffle** para generar una pequeña **dApp...**

```
truffle unbox webpack
```

que genera una serie de **directorios**.



SOLIDITY



- App
 - javascripts
 - stylesheets
 - index.html
- Contracts
 - MetaCoin.sol
 - Migrations.Sol
 - ConvertLib.Sol
- Test
 - metacoin.js
 - TestMetacoin.sol

Testing de contratos con Truffle

- Los archivos de test deben almacenarse en la carpeta **"test"**.
- Se ejecutan por medio del comando **"truffle test"**.
- Cada prueba se ejecuta en un entorno propio, no se comparten estados entre distintas pruebas.
- La ejecución Truffle-Ganache para pruebas tiene una performance con respecto a otras herramientas similares.
- Se puede ver un rastro de lo que sucedió en la prueba por medio del comando **"truffle test --stacktrace"**.
- Existe un comando **"truffle test --stacktrace-extra"** que muestra más información que funciona con versiones de Solidity mayores a 0.6.3



¡Muchas gracias!

¡Sigamos trabajando!