

## Blockchain

### • Descripción

En este proyecto se implementa un sistema Blockchain en Python. En la primera fase, se establece el back-end, programando la lógica interna para crear bloques, gestionar la cadena de bloques y manejar transacciones. Se diseñan mecanismos para calcular y gestionar hashes de bloques, evitando la suplantación de la cadena. Posteriormente se implementa un servicio web accesible mediante peticiones HTTP, desplegándolo en nodos de la red. Esto facilita la formación de una red sincronizada que mantiene la integridad de la información asociada a bloques y transacciones.

### • Desarrollo

Este proyecto consta de los siguientes elementos de desarrollo de software:

- 752 líneas de código.
- 3 módulos.
- 23 funciones.
- 3 clases.
- 1 fichero de texto.
- Librerías:
  - time, json, hashlib, uuid, platform, flask, argparse, threading, requests

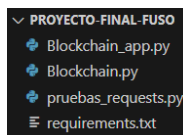


Figura 1. Estructura de módulos

### • Resultados

El proyecto detalla el proceso de desarrollo de blockchain, desde la implementación de la lógica interna del blockchain hasta la creación de una aplicación web descentralizada.

A través de la aplicación podemos añadir una transacción mediante una petición de HTTP de tipo POST.

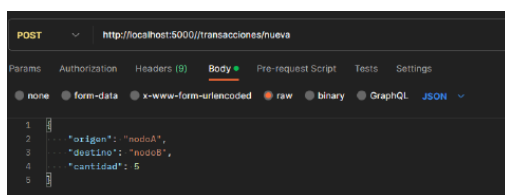


Figura 2. Método de añadir una transacción

También podemos solicitar al nodo minar un bloque mediante una petición HTTP de tipo GET o comprobar la cadena del blockchain que tenemos hasta el momento. La aplicación web descentralizada permite además que otros nodos se registren en ella.

Podemos ejecutar la aplicación en el puerto 5000 y lanzar otra instancia en el puerto 5001. En el primer puerto añadimos una transacción, minamos un bloque y comprobamos la instancia del puerto 5001.

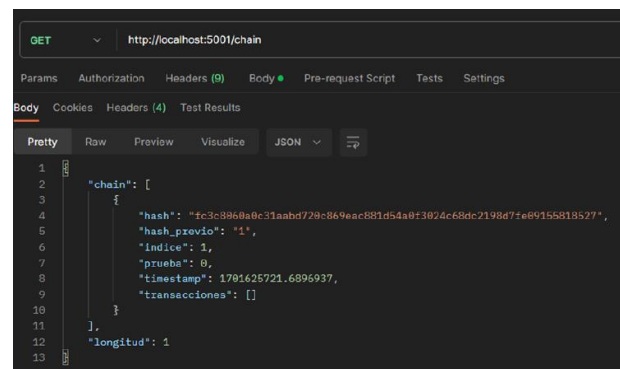


Figura 3. Resultado obtenido al comprobar la cadena

Registramos el nodo en la aplicación desplegada en el puerto 5000 y al hacer una petición GET a la aplicación desplegada en el puerto 5001 obtenemos la misma cadena del puerto 5000.

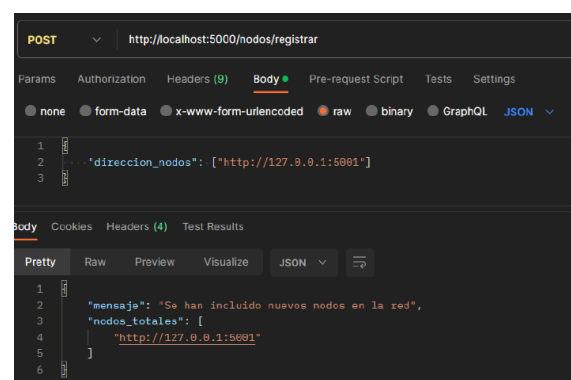


Figura 4. Resultado obtenido al registrar la cadena

En este trabajo también implementamos el protocolo ICMP para comprobar que el host de destino tiene conexión y realizamos modificaciones para que hubiese un nodo en el entorno Windows (host) y otro en una de las máquinas virtuales de Ubuntu (guest).