

# PatronusCash

Hackathon ITBA 2022



# Información del Grupo

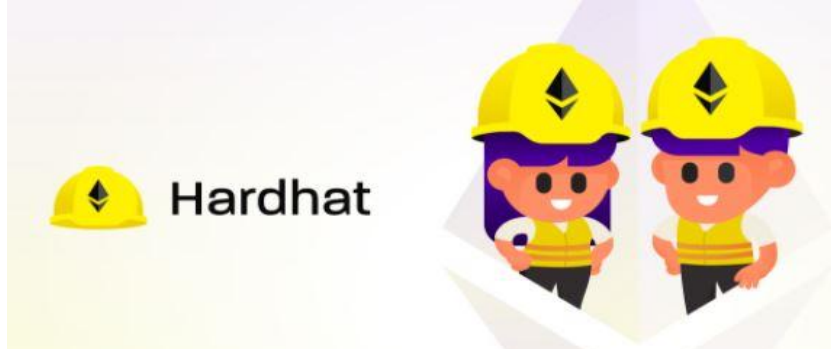
**Nombre:** Ethcaloneta

**Integrantes:** Juan Pose

**Modalidad:** Presencial

**Categoría:** Ciberseguridad y Privacidad

# Herramientas utilizadas



# Problema

Hoy en día existen infinidad de **scams**, **phishings**, y más actores que buscan **personas vulnerables** dentro de redes blockchain, como **Ethereum**.

## Propuesta: PatronusCash

Con el fin de **reducir la superficie de ataque** de actores maliciosos, y evitar que diversas personas continúen interactuando con direcciones ya verificadas como **scams, phishings, exploits**, y más, surge la idea de un contrato inteligente llamado **PatronusCash**.

## ¿En qué consiste?

```
contract PatronusCash is Ownable, ReentrancyGuard{  
    mapping(address => bool) public CriminalAddresses;
```

**PatronusCash** es un smart contract que propone una lista pública de direcciones ya comprobadas como maliciosas. Esta lista, o lista negra mejor dicho, será consultada para advertir al usuario sobre una posible estafa, por ejemplo, impidiendo que el mismo finalice dicha transacción y pierda su dinero.

# ¿Cómo funciona?

- 1) **Diego desea enviar Ether a una dirección.**
- 2) **En lugar de una transacción directa, Diego envía una transacción a PatronusCash, indicando la dirección a la que realmente quiere enviar dinero, junto con el monto.**
- 3) **PatronusCash verifica que la dirección destino no se encuentre en su lista negra. Si no lo está, el contrato continúa la transacción y envía el dinero correspondiente a la dirección indicada. Caso contrario, revierte la transacción y devuelve el dinero al propietario, indicándole con un mensaje que la dirección destinataria es maliciosa.**

# My Hardhat Token (MHT)

Welcome **0xf39fd6e51aad88f6f4ce6ab8827279cfff92266**, you have **999999 MHT**.

Error sending transaction: Address criminal detectada!



## Transfer

Amount of MHT

100

Recipient address

0x3C44CdDdB6a900fa2b585dd299e03d12FA4293BC

Transfer



¡Gracias por escuchar!

¿Preguntas?

