

# Creación de un Contrato Inteligente para un Sistema de Votación en Ethereum

Samuel López Iglesias

Octubre 2023

## 1 Introducción

Este documento detalla el proceso completo de creación de un contrato inteligente para un sistema de votación en una elección, utilizando la plataforma Ethereum y el lenguaje de programación Solidity. El sistema permitirá a los usuarios votar por candidatos y contar los votos de forma segura y transparente en la blockchain.

## 2 Análisis y Definición del Escenario

### 2.1 Caso de Uso

El caso de uso implica la implementación de un sistema de votación donde los usuarios pueden votar por candidatos en una elección.

### 2.2 Funcionalidades Requeridas

- Permitir a los usuarios votar por candidatos.
- Contar los votos de manera segura y transparente en la blockchain de Ethereum.

## 3 Diseño del Contrato Inteligente

### 3.1 Contenido del Contrato Inteligente

El contrato inteligente está diseñado con las siguientes estructuras y funciones:

#### 3.1.1 Estructuras

- **Candidato:** Representa un candidato con un ID, nombre y número de votos.

### 3.1.2 Funciones

- `agregarCandidato(string memory _nombre)`: Permite agregar un nuevo candidato al sistema.
- `votar(uint _candidatoId)`: Permite a los usuarios votar por un candidato específico.
- `obtenerTotalCandidatos()`: Retorna el total de candidatos en el sistema.
- `obtenerCandidato(uint _candidatoId)`: Retorna los detalles de un candidato por su ID.
- `obtenerGanadorVotacion()`: Retorna los detalles del candidato con más votos recibidos.

## 4 Implementación

El contrato inteligente fue implementado en Solidity utilizando Remix IDE, una herramienta de desarrollo de Ethereum en línea. El código del contrato fue programado siguiendo las estructuras y funciones definidas en la sección de Diseño del Contrato Inteligente.

## 5 Pruebas

Las pruebas del contrato inteligente se realizaron en Remix IDE, ejecutando las funciones del contrato para verificar que los votos se registren correctamente. También se realizaron pruebas unitarias utilizando frameworks como Truffle para validar la funcionalidad del contrato en diversas situaciones.

## 6 Conclusiones

El contrato inteligente para el sistema de votación ha sido creado con éxito y probado para garantizar su funcionamiento adecuado. Se ha demostrado cómo utilizar Solidity y Remix IDE para implementar un contrato inteligente básico en Ethereum. Es importante tener en cuenta que los contratos inteligentes reales pueden requerir medidas de seguridad adicionales y una lógica de negocio más compleja.