



Programación con Herramientas Modernas

Trabajo práctico 2021

Dominio y precondiciones

El dominio sigue siendo el mismo de la entrega 0. En esta oportunidad vamos a trabajar completamente fuera de la aplicación.

Entregables

Persistencia con Blockchain

En este caso vamos a modelar parte de nuestro dominio en una Blockchain Ethereum: un Smart contract que nos permita validar y manejar las preguntas respondidas.

El mismo debe tener persistidas las preguntas. Las mismas tienen un id autoincremental numérico, saben qué usuario es el autor, el texto de la pregunta y las posibles respuestas.

Así también se deben persistir las respuestas de los usuarios que contendrán como mínimo el texto elegido y el puntaje obtenido por responder.

El smart contract tendrá un dueño y una serie de diferentes estados que permiten alterar el funcionamiento del mismo. A saber:

- **ACTIVO** toda la funcionalidad habilitada.
- **LECTURA** sólo se pueden consultar preguntas existentes.
- **RESPONDER** sólo se permite consultar y responder preguntas, pero no crear nuevas.
- **BOOTSTRAP** sólo se permite crear nuevas preguntas.

Debemos acceder a las siguientes funcionalidades :

- Se debe poder obtener una pregunta a partir de su id.
- Se debe poder agregar una nueva pregunta indicando su autor (el ingresante), el texto y las opciones.

- Se debe poder responder una pregunta, indicando su id, el texto de la respuesta y el puntaje obtenido (0 para una respuesta incorrecta).
- Un usuario debe poder obtener el promedio aproximado de los puntajes de todas las respuestas que **él** hizo. Esto debe al menos contar con precisión entera (no es necesario calcular con decimales).

Así mismo se cuenta con las siguientes restricciones:

- No se debe permitir responder una pregunta propia.
- Solo el creador puede cambiar el estado del smart contract

BONUS:

El punto consiste en implementar una restricción adicional que no se debe permitir responder preguntas creadas hace más de 5 minutos.

El entregable deberá contar con los tests unitarios que validen el comportamiento del modelo. Pueden utilizar el framework de testeo mocha.

Tecnologías utilizadas

Para el desarrollo de la Blockchain vamos a necesitar:

- [Truffle](#) una herramienta que nos permite compilar y hacer deploy de Smart contract.
- [Solidity](#) que es el lenguaje que utilizaremos para desarrollar el smart contract.
- [Ganache](#) que es una herramienta que nos permite levantar una blockchain personal y testearla.