**Interacción con smart contract**

**Windows**

1) Iniciar Ganache.

2) Ir a la dirección de nuestro proyecto. En nuestro caso:

C:\Users\Javi-GISAI\Proyectos\WebDB

3) Una vez dentro, en la carpeta build\contracts, borramos los dos archivos json q encontramos: Gestion.json y Migrations.json. Esto lo hacemos para que a la hora de migrar el contrato al blockchain no nos dé problemas, y hemos de hacerlo **SIEMPRE** cada vez iniciemos todo este proceso.

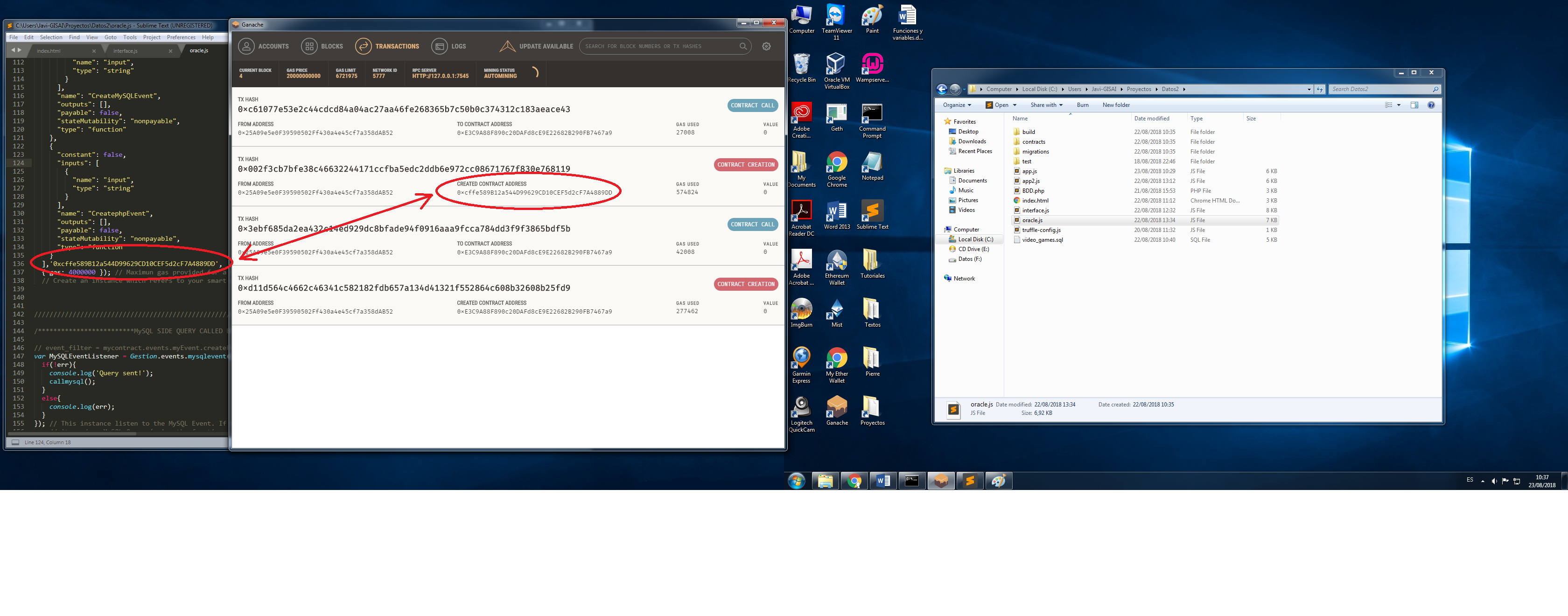
4) Abrimos un terminal, vamos a la dirección del proyecto y escribimos ***truffle compile***.

C:\Users\Javi-GISAI\Proyectos\WebDB>truffle compile

5) Una vez realizada la compilación, en esa misma ventana de terminal, escribimos ***truffle migrate.***

C:\Users\Javi-GISAI\Proyectos\WebDB>truffle migrate

6) Si todo ha salido bien, tendremos nuestro contrato en la blockchain. Antes de proseguir, hemos de comprobar que en el código de nuestros archivos *app.js* y *oracle.js* contienen la dirección correcta del contrato que acabamos de desplegar en la blockchain. Para ello, vamos a Ganache, y en la pestaña transacciones, comprobamos que, de las cuatro transacciones que tenemos actualmente, en la segunda, denominada *Contract Creation*, la dirección indicada como *Created contract address* coincide con la dirección de contrato que hemos incluido al declarar la variable Gestion en los archivos js antes mencionados. Más gráficamente:



7) Iniciamos Wampserver 64.

A la base de datos que tenemos ya creada podemos acceder tras iniciar Wamp mediante la dirección que pondremos en el navegador de Internet:

*localhost/phpmyadmin*

En nuestro caso, usamos como usuario *root* y el campo de contraseña lo dejamos vacío.

8) Desde la dirección del proyecto, iniciamos en tres terminales diferentes nuestros archivos js:

C:\Users\Javi-GISAI\Proyectos\WebDB> node oracle.js

C:\Users\Javi-GISAI\Proyectos\ WebDB > node app.js

C:\Users\Javi-GISAI\Proyectos\ WebDB > node interface.js

9) Para usar la aplicación web, en el navegador de Internet escribiremos la dirección siguiente para tener acceso a la página:

*localhost:1337*

Ejemplos de sentencias MySQL:

INSERT INTO video\_games (ID, name, owner, console, price, players, comments) VALUES (51, 'New Super Mario Bros', 'John', 'Wii', 4, 1, 'Incredible!')

SELECT name FROM video\_games WHERE console = 'Nintendo 64'

SELECT name FROM video\_games WHERE price >30

UPDATE video\_games SET owner='Kathy' WHERE ID=1

Podemos usar los **accesos directos y scripts** en la carpeta “**Scripts**” en el orden en que están señalados. Sencillamente tendríamos que editarlos y cambiar la localización de los archivos.

**Linux**

1) a 6) Seguiremos los mismos pasos que para Windows, pero teniendo en cuenta la nueva dirección donde se hallará nuestro proyecto. Recordemos que Linux es *case sensitive*.

7) En Linux emplearemos otro programa para acceder a nuestra base de datos, denominado Lampp. Para ejecutarlo, iremos a la dirección donde se ha instalado, en nuestro caso:

*/opt/lampp*

Una vez allí, escribiremos como superusuario:

*sudo ./lampp start*

Seguiremos pudiendo acceder a nuestra base de datos mediante la dirección en el navegador:

*localhost/myphpadmin*

8) y 9) Exactamente igual que en Windows, pero teniendo en cuenta el su equivalente funcionamiento en Linux.

Se han incluido una serie de **scripts** para hacer más sencillos algunos de los pasos. Habría que editarlos para adaptarlos a la nueva ruta. Primeramente habría que iniciar Ganache y luego ejecutar los scripts “*call*” y “*lamppstart*”. Tras esto, en la consola de comandos nos dirigimos a la carpeta donde se encuentra el proyecto y escribiremos:

*node oracle.js*

Y en otra consola, también dentro de dicha carpeta:

*node app.js*

Tras esto, podemos acceder desde el navegador de Internet tanto a la base de datos como a la página web con la que interactuaremos con la BS a través del smart contract.