**OFICIO No. 043/DGA/UNISTMO/2021**
**Asunto:** *Gastos De publicación, Solicitud de Liberación.*
Sto. Domingo Tehuantepec, Oax., 10 de junio de 2021

**Dr. Isaías Elizarraraz Alcaraz**
Director de Fortalecimiento Institucional
**ATN:**
**Lic. Sergio Pascual Conde Maldonado**
Jefatura de Desarrollo y Operación

Por este conducto le envío un cordial saludo, al tiempo que remito la **Solicitud de Liberación** correspondiente al *apoyo para Gastos de Publicación* autorizado al **PTC RICARDO CARREÑO AGUILERA**, adscrito a la Universidad del Istmo. Cabe señalar que el apoyo fue autorizado mediante oficio No. 511-6/2020-10232, de fecha 16 de diciembre de 2020.

| *Nombre del PTC* | *Revista / ISSN* | *Título del artículo* | *Costo (M.N.)* |
|---|---|---|---|
| *Dr. Ricardo Carreño Aguilera* | Fractals / 0218-348X | INTERNET OF THING EXPERT SYSTEM FOR SMART CITIES USING THE BLOCKCHAIN TECHOLOGY | $25,000.00 |

Se adjunta solicitud de liberación por parte del PTC en comento, Informe final (impacto académico logrado), copia del artículo publicado; y desglose financiero de recursos que emite la Universidad del Istmo. Los documentos en PDF se enviaron a los correos dfi.degesui@nube.sep.gob.mx, sconde@nube.sep.gob.mx y graciela.hernandez @nube.sep.gob.mx.

Garantizando la transparencia en el ejercicio de los recursos, agradezco la atención prestada al presente, y aprovecho la ocasión para agradecer los apoyos que nos brinda el Programa en mejora de la educación de nuestra región, nuestro estado y nuestro país.

A T E N T A M E N T E

*Voluntas totum potest*
*Guiraa zandaane guendaracala'dxi'*

**L.C.E. Claudia Hernández Cela**
**Jefa del Departamento de Gestión Académica**

UNIVERSIDAD DEL ISTMO
PROGRAMA PARA EL DESARROLLO PROFESIONAL DOCENTE, PARA EL TIPO SUPERIOR
PRODEP
REPRESENTANTE INSTITUCIONAL

C.f.p .-Dr. Modesto Seara Vázquez.- Rector de la Universidad del Istmo.- Para su conocimiento.
.-Dra. Cora Silvia Bonilla Carreón.- Vice-rectora Académica.- Universidad del Istmo.- Mismo fin.
.-Dr. Ricardo Carreño Aguilera.- PTC beneficiado.- Universidad del Istmo.- para seguimiento.
.- Archivo
*CHC/

# UNIVERSIDAD DEL ISTMO

## PROGRAMA PARA EL DESARROLLO PROFESIONAL DOCENTE, PARA EL TIPO SUPERIOR 2020

## OFICIO APROBACION DSA 511-6/2020-10232 16'DIC'2020

## GASTOS DE PUBLICACIÓN

**Beneficiario :** **RICARDO CARREÑO AGUILERA**

| Fecha | No. Ch. / T.E. | No. Comprobante | Beneficiario del Pago | R.F.C. | Concepto del Gasto | IMPORTE FACTURA (Ejercido) | GASTOS DE PUBLICACIÓN |
|---|---|---|---|---|---|---|---|
| | | | **Monto Aprobado** | | | | **$ 25,000.00** |
| 11/02/2021 | 64040 | 29012/DGA/2021 | RICARDO CARREÑO AGUILERA | *** | PAGO DE GASTOS DE PUBLICACIÓN DEL ARTICULO "INTERNET OF THINGS EXPERT SYSTEM FOR SMART CITIES USING THE BLOCKCHAIN TECNOLOGY " REVISTA FRACTALS. | **$ 25,000.00** | $ - |
| | | | | | **Total Gastos** | **$ 25,000.00** | **$ -** |

Cd. Ixtepec, Oax. a 15 de mayo de 2021

**L.C. Yazmín Carolina López Morales**
**Jefa del Departamento de Recursos Financieros**

**M.A. Oscar Cortés Olivares**
**Vice-Rector de Administración**

Sto. Domingo Tehuantepec Oaxaca a 20 de mayo de 2021

Asunto: Gastos de publicación,
Solicitud de Carta de Liberación.

**Dr. Isaías Elizarraraz Alcaraz**
Director de Fortalecimiento Institucional

**AT´N:**
**Lic.**

**Sergio Pascual Conde Maldonado**
Jefatura de Desarrollo y Operación

Sirva el presente para enviarle un cordial saludo y mi agradecimiento por el apoyo recibido en el proyecto Gastos de Publicación. Así mismo, aprovecho la ocasión para solicitarle de la manera más respetuosa la **Carta de Liberación** correspondiente al apoyo recibido en mérito del Programa, autorizado en el oficio No. 511-6/2020-10232 de fecha 16 diciembre del 2020. Es importante comentar que el artículo titulado "INTERNET OF THINGS EXPERT SYSTEM FOR SMART CITIES USING THE BLOCKCHAIN TECHNOLOGY", se publicó en la revista "fractals journal", con ISSN 0218-348X, en el volumen 29 (1) de fecha 29 enero del 2021.

Se adjunta al presente, copia del artículo publicado en revista indexada, así como, informe final que contiene el impacto académico logrado con el apoyo recibido.

Sin más por el momento quedo de usted.

A T E N T A M E N T E.

*"Voluntas totum potest"*
*Guirá' zanda ne guendaracala'dxi'*

**Atentamente**

Dr. Ricardo Carreño Aguilera
Profesor Investigador Titular A
Universidad del Istmo

C.c.p. –Dra. Cora Silvia Bonilla Carreón- Vice-rectora Académica. - Universidad del Istmo. -Para su conocimiento.
.-L.C.E. Claudia Hernández Cela.- Representante Institucional ante el Programa.- Universidad del Istmo.-Mismo fin.
.-Archivo

# Universidad del Istmo

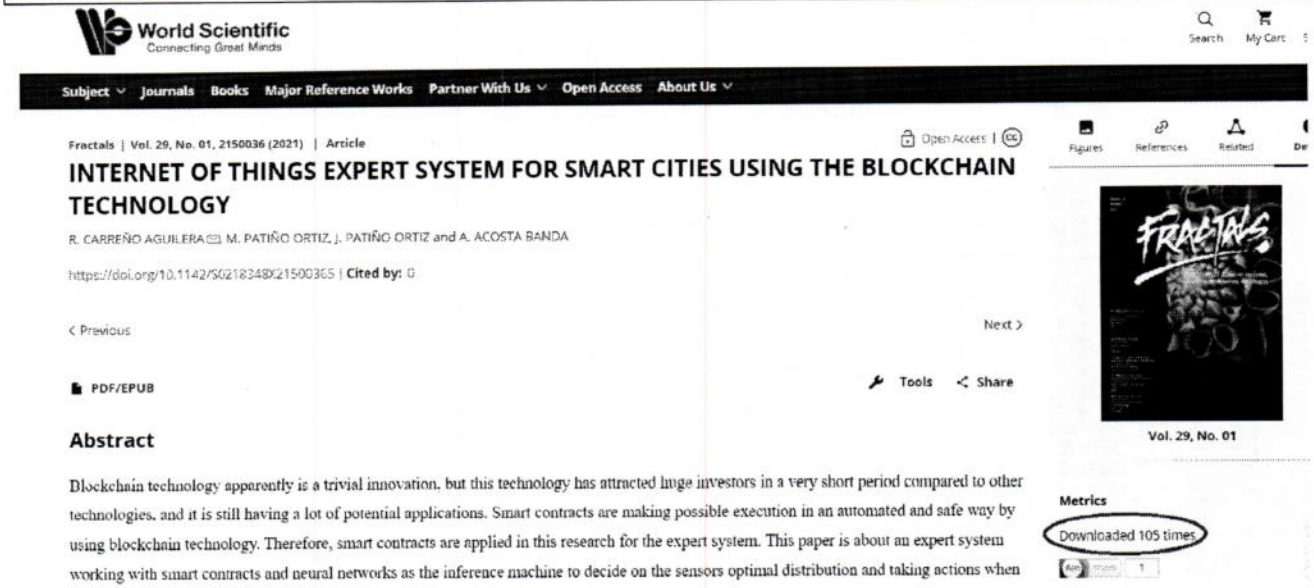
**INFORME FINAL**

**GASTOS DE PUBLICACIÓN**

| | | | |
|---|---|---|---|
| **IES de adscripción:** | **Universidad del Istmo** | **Título de la publicación:** | INTERNET OF THINGS EXPERT SYSTEM FOR SMART CITIES USING THE BLOCKCHAIN TECHNOLOGY |
| **Nombre del PTC:** | **Dr. Ricardo Carreño Aguilera** | **Nombre de la Revista:** | Fractals Journal.... |
| **ID del PTC:** | **172804** | **ISSN:** | 0218-348X |
| **Número de oficio de autorización:** | **511-6/2020-10232** | **Link: (opcional)** | https://www.worldscientific.com/doi/pdf/10.1142/S0218348X21500365 |
| **Fecha de autorización:** | **16 diciembre del 2020** | **Monto aprobado** | $25,000 |

***Impacto académico logrado con el apoyo recibido:***

Ejemplo: El artículo publicado hasta el momento del presente informe ha tenido 105 descargas.



World Scientific
Connecting Great Minds

Search My Cart

Subject Journals Books Major Reference Works Partner With Us Open Access About Us

Fractals | Vol. 29, No. 01, 2150036 (2021) | Article

Open Access

Figures References Related

**INTERNET OF THINGS EXPERT SYSTEM FOR SMART CITIES USING THE BLOCKCHAIN TECHNOLOGY**

R. CARREÑO AGUILERA, M. PATIÑO ORTIZ, J. PATIÑO ORTIZ and A. ACOSTA BANDA

https://doi.org/10.1142/S0218348X21500365 | Cited by: 0

< Previous Next >

PDF/EPUB Tools Share

**Abstract**

Blockchain technology apparently is a trivial innovation, but this technology has attracted huge investors in a very short period compared to other technologies, and it is still having a lot of potential applications. Smart contracts are making possible execution in an automated and safe way by using blockchain technology. Therefore, smart contracts are applied in this research for the expert system. This paper is about an expert system working with smart contracts and neural networks as the inference machine to decide on the sensors optimal distribution and taking actions when



**Dr. Ricardo Carreño Aguilera**

**L.C.E. Claudia Hernández Cela**
**Representante Institucional ante PRODEP**

OPEN ACCESS
Fractals, Vol. 29, No. 1 (2021) 2150036 (10 pages)

DOI: 10.1142/S0218348X21500365

World Scientific
www.worldscientific.com



# INTERNET OF THINGS EXPERT SYSTEM FOR SMART CITIES USING THE BLOCKCHAIN TECHNOLOGY

R. CARREÑO AGUILERA,*,‡ M. PATIÑO ORTIZ,†
J. PATIÑO ORTIZ† and A. ACOSTA BANDA†
*Universidad del Istmo, Cd. Universitaria S/N
70760, Tehuantepec, Oaxaca, México
†Instituto Politécnico Nacional, SEPI-ESIME ZACATENCO
07738, México City, México
‡ricardo.carreno.a@hotmail.com

Received February 20, 2020
Accepted August 17, 2020
Published January 29, 2021

## Abstract

Blockchain technology apparently is a trivial innovation, but this technology has attracted huge investors in a very short period compared to other technologies, and it is still having a lot of potential applications. Smart contracts are making possible execution in an automated and safe way by using blockchain technology. Therefore, smart contracts are applied in this research for the expert system. This paper is about an expert system working with smart contracts and neural networks as the inference machine to decide on the sensors optimal distribution and taking actions when sensor readings are out of range: control lights, activating fire alarms, temperature alarms, etc. for all spaces (parks, schools, hospitals, etc.) in a smart city based on the needs, and likes of the expert system user. This expert system works using a blockchain

‡Corresponding author.

structure on the EOSIO ecosystem with all data gathered by the sensors being saved in cloud online making internet of things environment and essential data saved in a blockchain node.

*Keywords*: Blockchain; Expert System; Smart Contract; IoT (Internet of Things).



## 1. INTRODUCTION

Blockchain technology[1,2] started its success with the creation and launch of Bitcoin in 2009,[3,4] although initially it was not intended to be useful other than for validating payment transactions,[5–10] that is why Bitcoin did not implement additional features rather than those that had to do with its only functionality that was secure transfer. However, while time goes by and with the importance that Bitcoin has developed throughout the world, people realized that the advantages in security, integrity, and decentralization of blockchain could be exploited for other usages such as smart contracts, initially used on the Ethereum platform.[11,12]

Although this innovative technology is relatively new compared to others with the same impact as the Internet or personal computers, it has caused disruptive changes in the way of producing technologies and managing finances, as well as in multiple other applications.[13–20] In essence, blockchain is an incorruptible digital book that records all kinds of transactions, whether financial or anything else that has value. This book is protected with cryptographic algorithms and is copied to each node that makes up the network, allowing its management to be decentralized.

What makes blockchain to be so important is that it eliminates intermediaries for any type of operation that we want to perform through a computer, allowing us to become independent of agencies or authorities in which we had to trust. What happens is that the transactions are validated by consensus automatically by the network users, and once validated, each transaction is protected by cryptography and linked to the previous transaction through the blockchain.

One of the most important blockchain-based projects due to its magnitude is EOSIO[21,22] as it goes beyond all its predecessors, emulating most of the computer and software attributes designed for the development and execution of decentralized applications in a scalable way. Although there is still a long way to go, EOSIO already allows to build and deploy these types of applications on its platform, therefore, we decided to use this platform.

The main purpose of this study is to develop an expert system[23–26] able to do two smart actions: (i) recommends the sensors distribution in spaces, either in smart cities or domotic houses (temperature, humidity, lighting sensors, etc.); (ii) take smart actions when sensors readings are out of range either control light intensity, turn on/off lights, activate alarms for fire sensors, etc. Data are stored and managed on a data-base with semantic searches[27] on a web page, to achieve the benefits of the concept of the internet of things,[28–30] and sensors coordinates are saved in the EOSIO platform in a EOS node to promote the use and mass adoption of this expert designer system. This expert system is intelligent since, being assisted by a neural network and smart contracts,[31–44] it learns how to make better recommendations as being used. The neural network consists of a Deep neural network of 72 layers using Kalman filters and the smart contract using the EOSIO platform, the deep learning performance is not shown in this paper since it deserves a deep analysis that it is not the core of this paper.

This study is also useful to analyze, through its use by diverse handlers on the web, the capacity of the EOSIO platform as a driver of applications in general. An *Ad hoc* application was developed for this purpose in this study.

It should be noted that one of the attractive points of this work is not only the expert system itself but in addition to that, it is the blockchain technology with the one it was implemented, letting the system to be decentralized and to promote the massive adoption of this expert system and thus take advantage of the popularity that this new technology possesses by itself.

## 2. METHODOLOGY

There are different methodologies to carry out the development of software. In this work, we described the incremental evolutionary development technique used; Incremental methodology refers to building the product in increments, one piece at a time.[45] The process is shown by a diagram in

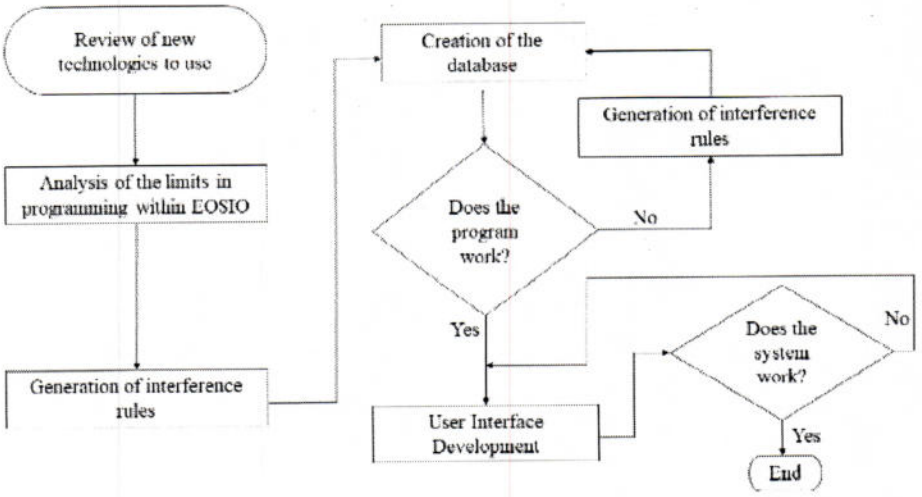



**Fig. 1** Incremental methodology for the development of the expert system.

Fig. 1 and the activities are described as follows:

(1) A general review of the new technologies to be used, in this case, is the entire ecosystem. Since EOSIO blockchain is in constant development, it is very important to keep updated to avoid incompatibility with the different versions.
(2) Download and installation of EOSIO software and development tools (CDT), as well as the detailed review of documentation and testing to know the limits of programming on this platform, both in performance and in size and type of programs that can be run.
(3) Software development starting with the creation of the smart contract with the rules that provide the operation of the expert system, duly specified so that they can interact with the outside once the contract is deployed in the blockchain.
(4) Appropriate table fields and types definition for the expert system database to store data and keep record of all the queries made by the users and its annex to the smart contract for its deployment in blockchain to corroborate its correct operation, if what was expected is what is gotten then go to step 5, if not return to step 3.
(5) Development of the user interface considering two main aspects, the first: is to build an interface that allows users to interact with the application easily, defining the output formats that the application will have, and; the second point will be to connect the web browser with the EOSIO blockchain so that the latter can respond to the different requests made from the application.
(6) Once the previous steps have been completed, the functionality of the application is verified, and if there are no details to solve, the work can be finished.

## 3. EXECUTION

Different and diverse tests were performed before starting the development of this expert system and blockchain. These isolated tests were performed to obtain information on what type of functionalities could be obtained from the blockchain. Some examples of this are the creation of actions that can interact with the blockchain outside, the user's validation and the creation of tables for storing information within the chain itself.

### 3.1. Creation and Deployment of the Smart Contract

For the smart contract creation described in step 3, a class was created in C++ following the specifications so that it can be recognized as a smart contract when compiling it with eosio-cpp; which is the compiler provided by the development tools, the code template is shown as follows:

```
#include <eosiolib/eosio.hpp>
  using namespace eosio;
   class name_class : public contract {
    public:
    using contract::contract;
     [[eosio::action]]
```

```
void function(argument type) {
// Actions for the Smart contract}};
EOSIO_DISPATCH(name_contract, (action)).
```

The following important points must be highlighted from the previous template (the Linux command lines of the EOSIO ecosystem are presented in quotes and bold letters).

(1) The **"eosiolib/eosio.hpp"** must include the library for the proper use of the EOSIO blockchain functions.
(2) It is important to inherit the content from the **"contract"** class to use its properties for the smart contracts.
(3) The **"[[eosio :: action]]"** label must be included for each function that will interact from the blockchain outside, these functions are known as actions.
(4) The **"EOSIO_DISPATCH (contract_name, (action))"** directive must be included so that the compiler can generate the files required by the EOSIO blockchain.

From this point on, the generation of the rules is like their implementation in any other object-oriented programming language. Once the rules have been generated, the structure that will store the facts that will be useful to the expert system is created and the table that will store the queries made by the users is created. This table is a structure that allows storing data in the chain of EOSIO blocks, the data storage structure is shown in the following code:

```
struct [[eosio::table]] table_name {
  uint64_t key;
   data type;
    data type;
     data type;
     data type;
  uint64_t primary_key() const {return key;}
};
typedef eosio::multi_index<"nickname"_n, table_name> type_name;
```

Some important points must be highlighted from the data storage code in the blockchain:

(1) The **"[[eosio:: table]]"** label should be placed to specify the compiler that this structure will be used to store data in the blockchain.
(2) A data type **"uint64_t"**, must be specified as a key to identify each new set of data entered.
(3) The **"primary_key ()"** function must be specified to retrieve each set of elements in the blockchain.
(4) The new type of data must be defined with the specification **"eosio :: multi_index"**.

Once the program is finished, it is compiled using the following instruction: **"eosio-cpp-on_contract_name.wasm contract_name.cpp – abigen"**. After obtaining the files generated by the compiler, for its deployment it was necessary to carry out the following steps:

(1) Create a wallet: **"cleos wallet create –to-console"**.
(2) Unlock the wallet: **"cleos wallet unlock"**.
(3) Create keys for the wallet: **"cleos wallet create_key"**.
(4) Import the main EOS key: **"cleos wallet import"**.
(5) Create a test account: **"cleos create account eosio account_name KEY_PUBLIC"**.
(6) Create an account to host the smart contract: **"cleos create account eosio NAME_CONTRATO KEY_PUBLICA -p eosio @ active"**.
(7) Finally deploy the smart contract: **"cleos set contract contract_name DIRECTORY_CONTRATES/contract_name -p account_name @active"**.

### 3.2. Creation of the User Interface and its Connection with the EOSIO Blockchain

After the corresponding tests to corroborate the correct operation and when no failures were found, the user interface was developed, it was defined as a web application. Therefore, the NodeJS framework was used for the development of the backend of the developed website and the eosj libraries were used to communicate to the web browser with the EOSIO blockchain, and that only by pressing a button from the web browser the data was sent from the application to the blockchain, making this process transparent to the user. The instructions for the use of eosjs within the backend can be observed with the creation of the user interface code and its connection with the EOSIO blockchain that is shown as follows:

```
const {Api, JsonRpc, RpcError} = require('eosjs');
const JsSignatureProvider = require('eosjs/dist/
    eosjs-jssig');
```

```
const fetch = require('node-fetch');
const {TextEncoder, TextDecoder} =
  require('util');
const defaultPrivateKey = active_key
const signatureProvider = new
JsSignatureProvider([defaultPrivateKey]);
const rpc = new JsonRpc('http://127.0.0.1:8888',
{fetch});
```

From the previous code the following is highlighted:

(1) Eosjs requirement in; require (**'eosjs'**).
(2) Definition of the activator key, to access the blockchain.
(3) Specification of the endpoint in which the queries are made.

The transparency of the interaction between the web browser and the EOSIO blockchain is achieved thanks to the fact that functions defined by the programmers of the blockchain are found in the eosjs library. Allowing its use, without needing to define them. The following code shows the template for the execution of actions/transactions in the blockchain from the browser:

```
(async () => {
 const result = await api.transact({
  actions: [{
   account: 'contract_account_name',// 2
    name: 'action_to_run', // 3
     authorization: [{
      actor: 'user_name',
         permission: 'active', // specifies
           that permission is required
          }],
         data: { // 4
                 name: data,
             name: data,
           name: data
              },
             }]
           }, {
     blocksBehind: 3,
       expireSeconds: 30,
     });
console.dir(result); // the results are printed
})();
```

From the previous code the following points are important:

(1) Specification of the asynchronous **"transact"** function from the eosjs api in **"await api.transact"**.
(2) Specification of the account name.
(3) Specification of the action to be executed.
(4) Specification of the data that the required action will use.

Despite the important parts that were listed in the previous point, the other elements must be kept as described, since they all prove to be essential for the execution of any type of smart contract. Once the backend was finished, the frontend development was done using html and css, to make it visually attractive.

## 4. RESULTS

The execution of a local node can be seen in Fig. 2. The results are shown by the terminal and in these, the actions performed on the blockchain as well as other relevant data can be observed. This fact is important since this node is where all the smart test contracts and the final expert system contract proposed in this investigation are assembled.

Figures 3a–3c show the results of the smart contract registration within the blockchain, which is the engine of the expert system and the neural network proposed. The results are displayed in a self-generated text file. To reach this result, the correct compilation of the smart contract code must be done first.

As mentioned above, the proposed expert system can be used through a web browser. In Fig. 4, the main view of the application's user interface is shown with all the fields that must be filled in to perform a query.

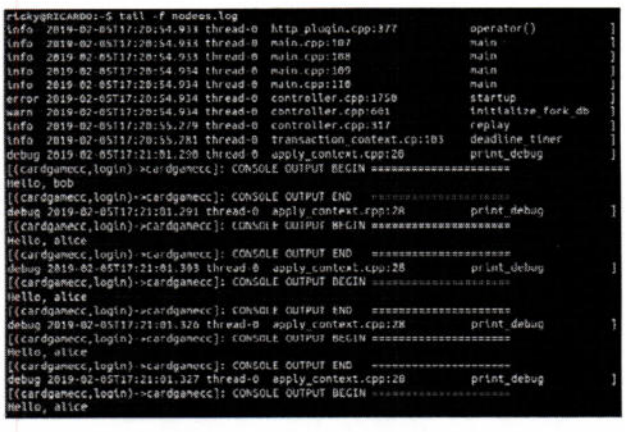



**Fig. 2** EOSIO blockchain running locally.

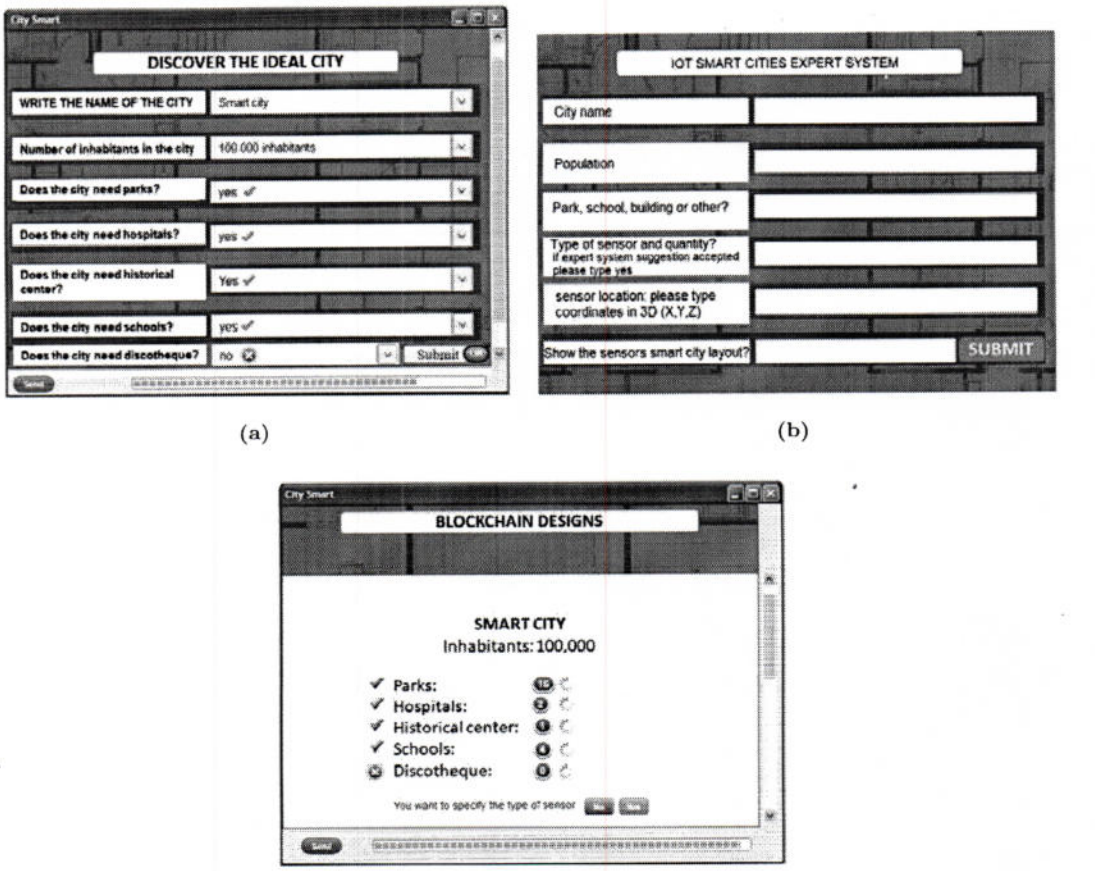
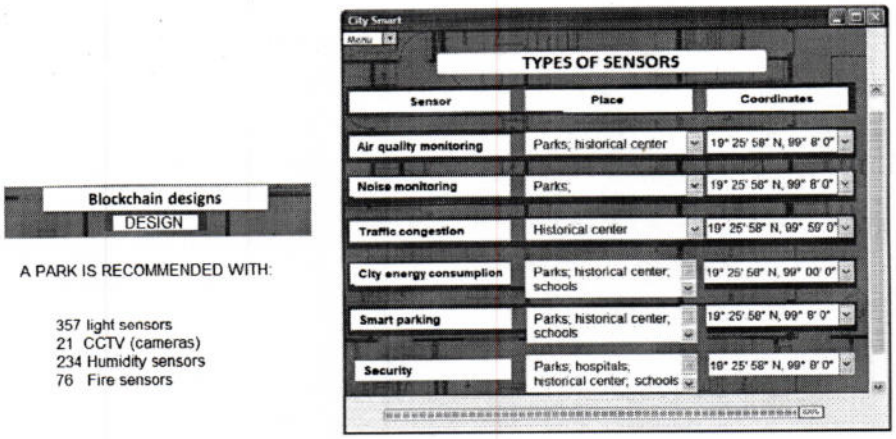



**Fig. 3** **(a)** Registration of the smart contract in the blockchain. **(b)** Registration of the smart contract in the blockchain. **(c)** Registration of the smart contract in the blockchain.

```
• cleos set contract si.exp /home/ricky/contracts/si.exp -p si.exp@active
• Reading WASM from /home/ricky/contracts/si.exp/si.exp.wasm...
• Publishing contract...
• executed transaction: db13f144bca89720c78d748e32a322facbf9f5da5e4451d6d6bf85fec4b3530e
  1448 bytes  697 us
• #         eosio <= eosio::setcode               {"account":"si.exp","vmtype":0,"vmversion":
  0,"code":"0061736d0100000001390b60027f7e006000017f60027f7f...
• #         eosio <= eosio::setabi                {"account":"si.exp","abi":
  "0e656f73696f3a3a6162692f312e3000010268690001047573657046e616d650100000000...
• warn  2018-12-09T00:18:52.518 thread-0  main.cpp:487                  print_resuwarning:
  transaction executed locally, but may not be confirmed by the network yet
```

**Fig. 4** The main view of the application's user interface, the query is made here.

The expert system saves the coordinates data of the sensor distribution of the smart city in an EOS node with blockchain technology. These coordinates can be retrieved by the expert system at any time with a query done by the expert system user, but the purpose of the expert system is not only to optimize and propose the distribution of the sensors but also to optimize the usage of the sensors for the smart city, for example: save power when lights are not required and regulate the intensity, set alarms on when fire sensors are activated, set temperature alarms on when temperature sensors are activated, etc. The expert system is smart since it is using smart contracts on the EOS platform to take smart decisions about the best choice of sensors for each space (parks, schools, buildings, etc.) and to active control if the case, alarms and actions when sensors are out of a regular range, regarding the expert system databases: there is a database in a EOS node saving the coordinates of the sensors




(a) (b)

**Fig. 5** **(a)** Expert system query result showing a proposal for each space (a park in this case) **(b)** Expert system query result showing sensor distributions proposal with coordinates.

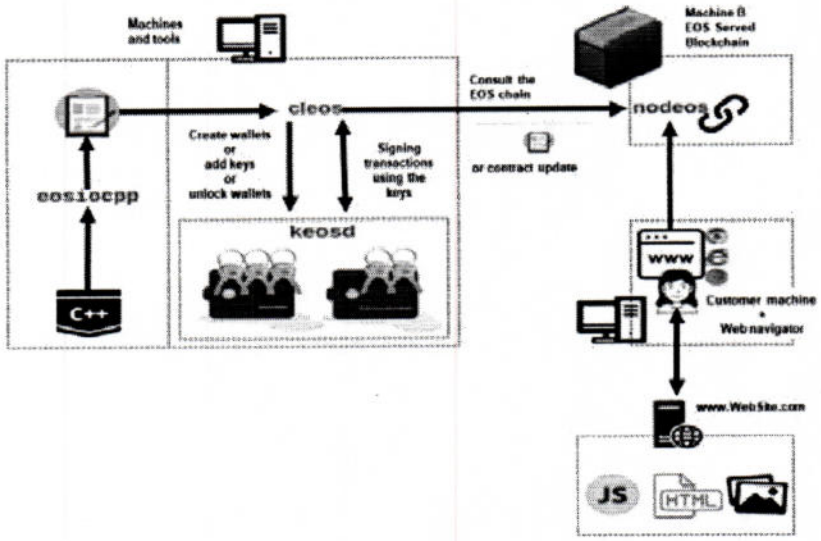



**Fig. 6** Operation scheme of decentralized applications.

and another database in the internet cloud saving the general information of the users and the expert system. When coming to the 5g internet technology, sensors will become common in usage by all people. Therefore, smart cities as well as smart contracts can be a reality.

Figures 5a and 5b show the second view of the application interface where you can see the results of a query made. Both views are linked since the first leads to the second. In Fig. 5a, the proposed quantity of sensors for a park is shown. Also, the proposed quantity of sensors for all the spaces of the smart city can be shown. In Fig. 5b, the 3D location for all sensors of all spaces (park, schools, hospitals, etc.) can be shown. This can be useful to locate sensors or install sensors. The expert system is a shell (a prototype) doing the proposals with a minimum learning based on an initial database, but more learning takes place when the users introduce more information, this is the basic concept as human beings learn since birth but knowledge increases along the time.

Finally, all the actions and components that are performed when using the proposed application are exemplified by a scheme in Fig. 6, such a process must be followed in all EOSIO based platform applications, since one of the key points is security. Therefore, certain guidelines must be followed for the proper development of such applications.

## 5. CONCLUSIONS

In this study, two relevant topics were addressed: (i) the development of an expert system for house specifications design that can be executed in a decentralized way through a smart contract and (ii) this, through a blockchain platform, both fields (expert systems and blockchain) still have much to explore. However, with their current capabilities, they are already revolutionizing the way people do things. The functionality of a node of the EOSIO platform was tested, achieving satisfactory results with the minimum RAM requirements (8 GB). The correct deployment of a smart contract on the EOSIO platform was also achieved, proving that the contract may have functionality errors, but if eosio-cpp compiles it correctly, it will not mark an error when deployed. User interface views are functional to demonstrate that a viable product of an expert system can be created and executed in the EOSIO blockchain. In addition, a simple interaction with the user was achieved, since the user at no time perceives to be using blockchain technology, proving with this that these applications can turn out to be simple but powerful.

The general objective of creating an expert system that could be executed in a blockchain was achieved, being able to demonstrate that the EOSIO platform is already functional and that it offers performance comparable to the centralized services provided by large corporations such as Amazon, Microsoft, and Google. These results could motivate more developers to start using this technology, achieving a benefit from both parties as the user obtains security and functionality at low cost and the platform grows more with each new user.

## 6. DISCUSSIONS

The development and the results obtained from the design and creation of the expert system for the recommendation of distribution of spaces and house sensors, with the main core of a smart contract deployed in the EOSIO blockchain proposed in this study could be the basis of much more powerful and complete applications.

An expert system capable of giving recommendations for house specifications has not been explored yet, or at least none of which implementation is explained in the literature. Despite this, its implementation is relatively simple since it works with the knowledge base and rules like any other expert systems. However, what makes a difference is its implementation with the EOSIO blockchain since it has not been massively explored, what was done was the most practical and useful application for its users. Besides, it could demonstrate the scope of the EOSIO platform.

As already stated, the proposed expert system has no direct precedents. However, it can be compared with other works that could be related

(1) In the paper "The Design and Implementation of an Intelligent Apparel Recommend Expert System",[46] a design scheme of a garment recommendation system is proposed to improve the customer's shopping experience. This expert system is well detailed and complete; however, it is developed under traditional technologies that are not related to blockchain technology.
(2) In the paper "Expert Systems Application In Manufacturing", the application of expert systems in the manufacturing process is discussed. Although more than one expert system is mentioned, it does not focus on its implementation, focusing only on its use, besides blockchain technology is not mentioned or used.[47]
(3) The study of the use of expert systems in agricultural production in China is addressed in "The Research of Agricultural Expert System based on IOT". The use of the internet of things for data collection using a mobile device for the interface in an expert system is explained. However, it does not have a relation to house design, and it does not refer to the use of blockchain for its implementation.[48]
(4) In the "An IoT Expert System Shell in BlockChain Technology with ELM as Inference Engine", an expert system for the creation of designs, using blockchain technology is proposed. This is the most comparable work to which described in this document. It also proposes the creation of an ICO on the Ethereum platform for the financing of the project.[49]

## REFERENCES

1. E. Morgen, Blockchain world — Do you need a blockchain? This chart will tell you if the technology can solve your problem, *IEEE Spectrum* **54**(10) (2017) 38–60.
2. J. J. Xu, Are blockchain immune to all malicious attacks? *Financial Innov.* **2**(25) (2016) 1–7.
3. S. Nakamoto, Bitcoin contract: A peer to peer electronic cash system, www.bitcoin contract.org, 1998.

4. J. Donier, A million metaorder analysis of market impact on the bitcoin, Market microstructure and liquidity, *World Scientific Publishing Company* **1**(2) (2015) 1–29.
5. S. Haber and W. S. Stornetta, Secure names for bit-strings, In *Proc. 4th ACM Conf. Computer and Communications Security*, 1997, pp. 28–35.
6. R. C. Merkle, Protocols for public key cryptosystems, In *Proc.* 1980 *Symp. Security and Privacy*, IEEE Computer Society, 1980, pp. 122–133.
7. S. Haber and W. S. Stornetta, How to time-stamp a digital document, *J. Cryptol.* **3**(2) (1991) 99–111.
8. D. Bayer, S. Haber and W. S. Stornetta, Improving the efficiency and reliability of digital time-stamping, In sequences II: Methods in communication, *Security and Computer Science*, 1993, pp. 329–334.
9. Z. Zhang *et al.*, Meteorological satellite data distributed storage method based on hash algorithm, *Artif. Intell. Sci. Technol.* **1** (2017) 49–53.
10. X. Lin, Research on information encryption and authentication in big data environment, *Des. Manuf. Mechatron.* **1** (2017) 714–719.
11. G. Wood, Ethereum: A secure descentralised generalised transaction ledger Byzantium version, http://ethereum.github.io/yellowpaper/paper.pdf, 2019.
12. J. A. F. Castellanos, D. Coll-Mayor and J. A. Notholt, Cryptocurrency as guarantees of origin: Simulating a green certificate market with the Ethereum blockchain, *IEEE Int. Conf. Smart Energy Grid Engineering* (*SEGE*), 2017, pp. 367–372.
13. S. Pradip, C. Mu-Yen and P. Jong, A software defined fog node based distributed blockchain cloud architecture for IoT, *IEEE Access* **99** (2017) 115–124.
14. J. Sun *et al.*, Blockchain based sharing services: What blockchain technology can contribute to smart cities, *Financial Innov.* **2**(26) (2016) 1–7.
15. J. L. Zhao, Overview of business innovation and research opportunities in blockchain and introduction to the special issue, *Financial Innovation* **3**(9) (2017) 3–9.
16. T. Aste, P. Tasca and T. D. Matteo, Blockchain technologies: The foreseeable impact on society and industry, *Computer* **9** (2017) 18–28.
17. M. E. Peck and D. Wagman, Energy trading for fun and profit buy your neighbor's rooftop solar power or sell your own-it'll all be on a blockchain, *IEEE Spectrum* **54**(10) (2017) 56–61.
18. L. Cheng and Z. Liang-Jie, A blockchain based new secure multi-layer network model for internet of things, *IEEE International Congress on Internet of Things* (*ICIOT*), 2017, pp. 33–41.
19. J. Sidhu, Syscoin: A peer-to-peer electronic cash system with blockchain-based services for e-business, *26th Int. Conf. Computer Communication and Networks* (*ICCCN*), 2017, pp. 1–6.
20. H. Kim and M. Laskowski, A perspective on blockchain smart contracts: Reducing uncertainty and complexity in value exchange, *26th Int. Conf. Computer Communication and Networks* (*ICCCN*), 2017, pp. 1–6.
21. Blockone, EOS.IO Technical White Paper v2, https://github.com/EOSIO/Documentation/blob/master/TechnicalWhitePaper.md, 2018.
22. Blockone, developers, https://developers.eos.io/eosio-home/docs.
23. W. Gu *et al.*, Expert system for ice hockey game prediction: Data mining with human judgment, *Int. J. Inform. Technol. Decis. Making* **15**(4) (2016) 763–789.
24. C. A. Magni *et al.*, An alternative approach to firms evaluation: Expert systems and fuzzy logic, *Int. J. Inform. Technol. Decis. Making* **5**(1) (2016) 195–225.
25. N. J. Pizzi *et al.*, Expert system approach to assessments of bleeding predispositions in tonsillectomy/adenoidectomy patients, *Adv. Artif. Intell.* **27** (1990) 67–83.
26. Agrawal, Prateek, Madaan, Vishu, Kumar and Vikas, Fuzzy rule-based medical expert system to identify the disorders of eyes, ENT and liver, *Int. J. Adv. Intell. Paradigm.* **7** (2015) 352–367.
27. M. Kopel, Libraries in the semantic web era, *Qualitative and Quantitative Methods in Libraries* (2010) 115–122.
28. A. Dattakumar and R. S. Sharma, Smart cities and knowledge societies: Correlation, causation or distinct? 2016 *IEEE Int. Conf. Management of Innovation and Technology* (ICMIT), 2016, pp. 193–197.
29. M. Samaniego and R. Deters, Internet of smart things — IoST: Using blockchain and CLIPS to make things autonomous, *IEEE Int. Conf. Cognitive Computing* (*ICCC*), 2017, pp. 9–16.
30. F. K. Flores, Internet of things: Managing wireless sensor network with rest API for smart homes, *Theory and Practice of Computation, Manila* (2016), pp. 132–142.
31. W. Yu *et al.*, Modeling of gasoline blending via discrete time neural networks, *Neural Networks Procedure* (2004), p. 6.
32. W. Yu *et al.*, Neural networks for the optimization of crude oil blending, *Int. J. Neural Syst.* **15**(5) (2005) 377–389.
33. W. Yu and X. Li, Automated nonlinear system modeling with multiple fuzzy neural networks and kernel smoothing, *Int. J. Neural Syst.* **20**(5) (2010) 429–435.

34. S. Lu *et al.*, Nonlinear autoregressive and nonlinear autoregressive moving average model parameter estimation by minimizing hypersurface distance, *IEEE Trans. Signal Process.* **51**(12) (2003) 3020–3026.
35. L. Sheng and J. Hwan, A new algorithm for linear and nonlinear ARMA model parameter estimation using affine geometry, *IEEE Trans. Biomed. Eng.* **48**(10) (2001) 1116–1124.
36. J. J. Carrasco, ELM regularized method for classification problems, *Int. J. Artif. Intell. Tools* **25**(1) (2016) 7–14.
37. G. B. Huang, What are extreme learning machines? Filling the gap between Frank Rosenblatt's dream and John von Neumann's puzzle, *Cogn. Comput.* **7**(3) (2015) 263–278.
38. Y. Li, Extreme learning machine for determining signed efficiency measure from data, International Journal of Uncertainty, *Fuzziness Knowl.-Based Syst.* **21**(2) (2013) 131–142.
39. R. Carreño *et al.*, Comparative analysis on nonlinear models for ron gasoline blending using neural networks, *Fractals* **25**(6) (2017) 1–8.
40. L. Yu *et al.*, Prediction-based multi-objective optimization for oil purchasing and distribution with the NSGA-II algorithm, *Int. J. Inform. Technol. Decis. Making* **15**(2) (2016) 423–451.
41. S. Rionero, Lyapunov functionals for the coincidence between the first and the second lyapunov stability methods, *New Trends in Fluid and Solid Models*, 2009, pp. 113–121.
42. H. T. Huynh *et al.*, An improvement of extreme learning machine for compact single hidden-layer feedforward neural networks, *Int. J. Neural Syst.* **18**(5) (2008) 433–441.
43. Y. Huan *et al.*, Nonlinear manifold learning for early warnings in financial markets, *Eur. J. Oper. Res.* **258** (2017) 692–702.
44. R. Carreño *et al.*, A nonlinear acoustic transducer model with variable structure control, *Fractals* **25**(2) (2017) 1–10.
45. C. Jones, *Software Methodologies: A Quantitative Guide* (Taylor & Francis Group, 2018).
46. A. H. Dong, D. Shan, Z. Ruan, L.Y. Zhou and F. Zuo, The design and implementation of an intelligent apparel recommend expert system, *Mathematical Problems in Engineering* **2013** (2013) 1–8.
47. P. Som, R. Chitturi and A. J. G. Babu, Expert Systems Application In Manufacturing. Proceedings of SPIE — *The International Society for Optical Engineering* **786** (1987) 474–479, 10.1117/12.940659.
48. B. Fu, The Research of Agricultural Expert System Based on IOT (2015).
49. R. Carreño *et al.*, An IoT expert system shell in block-chain technology with ELM as inference engine, *IJITDM* **18**(1) (2019) 87–104.