Smart Contracts for Multi-agent Interaction of Regional Innovation Subjects

Leyla Gamidullaeva[1], Alexey Finogeev[1(✉)], Sergey Vasin[1], Anton Finogeev[1], and Sergey Schevchenko[2]

[1] Penza State University, 40, Krasnaya str, Penza, Russia
gamidullaeva@gmail.com, alexeyfinogeev@gmail.com, pspu-met@mail.ru, fanton3@yandex.ru
[2] National Technical University «Kharkiv Polytechnic Institute», 2, Kyrpychova str, Kharkiv, Ukraine
s.v.shevchenko55@gmail.com

Abstract. The main obstacle to effective interaction between innovation agents is high innovation transaction costs. The development of innovation requires the continuous interaction of participants at all stages of the innovation process, from idea to commercialization. The article discusses the creation of a safe and reliable way to support such interaction in regional innovation systems based on blockchain technology and smart contracts. This approach is recommended to exclude unfair and fraudulent actions on the part of participants. Another feature is the transfer of third-party functions to a smart contract to ensure safe communication. The smart contract will allow, on the one hand, to realize trustful and reliable relationships between the project participants themselves, and, on the other hand, between participants and stakeholders. The article discusses the possibilities of the Ethereum blockchain platform, with the help of which the main components of a smart contract were synthesized for concluding contracts for creating and introducing innovations, transferring intellectual property rights, using licenses, etc. The basis of the smart contract is a distributed registry of transactions and a database with descriptions of innovative objects.

Keywords: Smart contract · Blockchain · Ethereum platform · Big data · Cyber-social system · Innovation system · Data mining

1 Introduction

The regional innovation systems existing nowadays are uncoordinated due to insufficient interaction between them. The main barrier impeding efficient interaction of innovation activity subjects (agents) in Russia is high transaction costs. It is necessary to simulate interactions of regional activity participants to have a qualitative and quantitative understanding of the role of relationships between environment's components, their impact on the development of a concrete region. One of the ways to develop innovation is the growth of interactions between the participants of the regional innovation system in the process of creating an idea before being implemented in a real commercial project. There are certain problems of innovative enterprises

A. G. Kravets et al. (Eds.): CIT&DS 2019, CCIS 1083, pp. 420–434, 2019.
https://doi.org/10.1007/978-3-030-29743-5_34

associated with the growth of instability and uncertainty of relations and relations between them. This causes an increase in transaction costs and, consequently, an increase in the cost of creating and promoting innovation. Innovation costs are not associated with resource changes. They have a transactional nature, which is determined by the interactions of the participants to enter into mutually beneficial contacts. Such transaction costs will be considered the main barrier to the innovative development of enterprises, since they actually reduce the owners' interest in the introduction and use of innovations [1, 2].

It is known that the process of innovation creation is a result of interactions between innovation activity agents, It depends both on transactions within a company and on established relations of an economic agent with the institutional environment. The institutional environment creates conditions for cooperation of economic agents with other innovation activity participants in terms of searching for information and other resources, experience sharing. It determines the behavior of economic agents by shaping their behavior sets, formal institutional structures that are crucial when making decisions on creation and implementation of innovations. According to the neoinstitutional approach the innovation system is a complex open structure providing interaction of economic agents, formal and informal regional institutions.

Innovation process participants should interact freely to gain experience of joint projects aimed at development and commercialization of innovations. This requires efficient coordinated relations within the "subject-project-medium" continuum. Regarding the intensification of interactions of a subject, a project and a medium the key process is self-organization providing a decrease of transaction costs, which is based on promoting the emergent effect unavailable to some innovation activity participants. Minimization of transaction costs means determination of their optimal level, which is reasonably necessary for economic system functioning [3]. Consequently, the innovation system management should be focused on development and realization of a strategy identifying socioeconomic structures with investment appeal that can be called innovation attractors. Such structures most precisely correspond to main paths of regional development and take into account a current situation and participants' capacities.

At present, various cyber-social technologies are being successfully developed, which can also be used to support the interaction of participants in the innovative regional system. An example of the use of such technologies is the creation of a social network for communication among the participants of the innovation environment. The network allows you to develop a community of people interested in creating and promoting innovation. These include universities and research centers, industrial enterprises, government agencies, ordinary citizens. The use of cyber-social technologies will allow turning the innovative system of a region into an adaptive intellectual environment due to synergistic effect [4].

The cyber-social system may be considered as an intellectual system focused on lowering the barriers to implementation of innovations (transaction costs of innovation activity subjects' interaction) by engaging a larger amount of participants in the innovation process and ensuring their intensive interaction. Its synthesis requires a mechanism that will enable different agents of innovation interaction having common development goals to create new knowledge and exchange it in a safe intelligent network.

2 Blockchain, Smart Contracts and Decentralized Applications for Multi-agent Interaction

The implementation of secure transactions during the exchange of information between participants in economic interaction is a necessary condition for supporting transactional processes, including in innovative systems. The objective is to reduce the influence of interested parties in concluding contracts and fulfilling contractual obligations, as well as their possible interference in the interaction processes between economic agents. One of the mechanisms for secure interaction is distributed registry technology (blockchain), the creation and implementation of smart contract algorithms.

The main problem of the safe use of e-commerce technology is to ensure trust relationships between users, online trading platforms and financial structures. Here is widely used method of attracting a third trusted party to confirm the guarantees of the transaction. But this method can not fully guarantee the reliability and security of transactions. At this stage of development, the blockchain technology or a distributed registry with hashing algorithms can replace the certification mechanism using a third party to completely eliminate the possibility of fraudulent activities on the part of the parties to the transaction. In fact, the functions of a third party are transferred to some intellectual agent, f namely, the smart contract algorithm [5].

As is known, the blockchain technology involves the synthesis of a sequential chain of blocks, which is called a distributed registry, according to a given algorithm. The registry stores information about participants in transactions, transactions and objects of the transaction, in our case, participants in innovative interactions and innovations [6]. In this case, copies of the blocks are placed on the computing devices of the participants themselves, where they are processed [7, 8]. The technology was created to create and conduct operations with cryptocurrencies and was first implemented in the Bitcoin system. As it turned out, it can be used to organize and support any information interactions. All data in the system is stored on user nodes as a distributed database, and not in a centralized repository. At the same time, only one part of information (block) or several copies of blocks is contained on one node. The principle of decentralized storage does minimizes system vulnerabilities to information attacks. The information blocks themselves are encrypted using hash functions (Fig. 1).

As is known, the blockchain technology involves the synthesis of a sequential chain of blocks, which is called a distributed registry, according to a given algorithm. The registry stores information about participants in transactions, transactions and objects of the transaction, in our case, participants in innovative interactions and innovations [6]. The copies of the blocks are placed on the computing devices themselves. An important feature of the distributed registry is the fact that each participant in the transaction actually has information about the actions of other parties by copying and distributing blocks with transactions throughout the network. This ensures transparency of transactions and serves as a confirmation of trust relationships between all subjects of innovation activity. The decentralized repository stores all the data to ensure innovative interactions. Examples of such data are:

- personal data of participants,
- information about legal entities,

– intellectual property rights and description of protected objects,
– digital copies of various documents (contracts, invoices, invoices, etc.),
– records of financial transactions (copy payment orders, cash checks), etc.

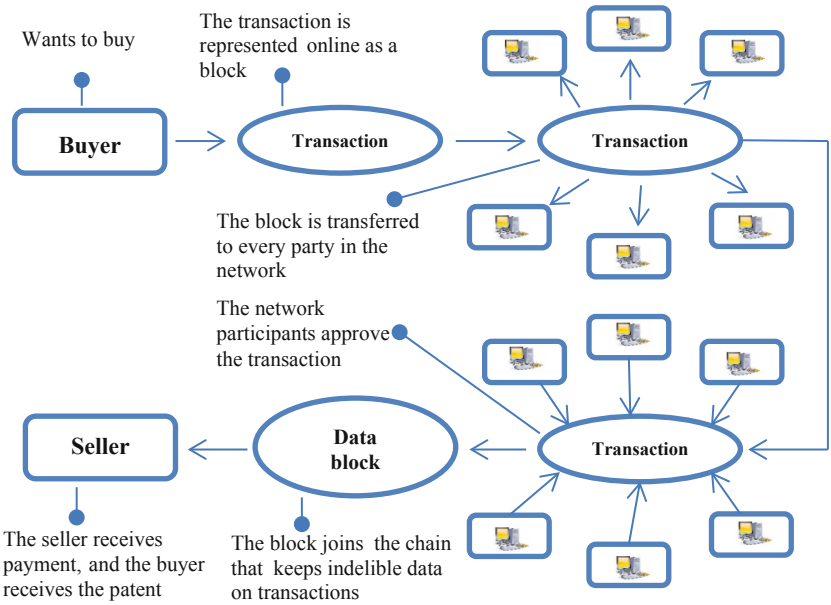

Fig. 1. A transaction scheme using the distributed register

In the information space, authenticity and rights to virtual objects are most often realized through the use of digital certificates, which are issued by a third trusted party. The blockchain technology allows you to opt out of a third party, for example, to confirm the right of authors and intellectual property owners to innovative objects. In addition, the technology will help to control the life cycle of innovations from their creation to implementation and obtaining a commercial result [9]. As an example, we give scientific publications. In many countries, scientists who work in universities and other educational institutions exhibit the greatest publication activity. They are not always participants in economic interactions and often are not associated with the actual production process. However, they are the main sources of innovation. Enterprises, in turn, cannot use their research in economic activity. The author of innovations must register the intellectual property right and then he can transfer it to a company that is interested in putting the innovation into practice or in its refinement and further transfer to the next participant in the innovation process. Thus, a chain of innovative transactions arises, where at each step it is necessary to guarantee the security and reliability of the transfer of intellectual property rights. At the same time, it is required to provide information on the completed operations of all previous owners up to the primary author's source. To implement such a chain of interactions, a distributed

registry mechanism is needed. The blockchain technology will allow you to follow the process of creating, modifying, transferring and using innovations, as well as help in obtaining and distributing fair remuneration between all owners and authors. An example of the use of such an approach is the creation of a distributed register of created objects of art, with assigning unique hash identifiers to authors and their works and then tracking the transfer of ownership in the process of selling or donating [10].

An example of blockchain application in intellectual property rights management in Russia is the implementation of IPChain on HyperLedger Fabric allowing to work with different information channels within a single register and determining the transaction approval policy for each of them [11]. Hyperledger Fabric is a project of the consortium led by IBM embracing top IT companies, such as Intel, Oracle, Cisco, Digital Assets, etc. The main advantage is an adaptive algorithm designed to achieve concord between trusted nodes by means of a mechanism that performs decentralized registration of transactions in a set number of equal nodes and, in case the authenticity of results is proved, confirms a transaction. The infrastructure of IPChain includes a bound register of intellectual property objects and a transaction register with the said objects, transaction registry nodes, transaction fixing nodes, network administration nodes and trust certificate issuing nodes.

In 1994 Szabo [12] proposed the conception of smart contracts, which became possible to realize only in 2008, when blockchain occurred. As a smart contract, this concept defines some algorithm by which it is possible to implement the contracting process and ensure the safe and secure fulfillment of contractual obligations based on records in the form of blockchains. [13]. This type of contracts fits any transactions. It guarantees money transfer or execution of other actions as soon as all parties have completed contract obligations. When parties conclude a smart contract, it is similar to transferring of cryptocurrency blockchain funds. After that the contract comes into effect. In order to have contract's obligations automatically complied with there is required a special environment enabling automatic execution of all contract's clauses. Thus, a smart contract can exist only within such environment, where the program code executing the contract's algorithm has access to its objects. Therefore, all relationships between parties within the contract should be mathematically formalized and feature a clean execution logic. According to transaction conditions, the smart contract's algorithm tracks accomplishments or breaches of its clauses and makes a corresponding decision automatically to ensure authenticity of contract obligations.

To ensure interaction between interested parties, the algorithm should describe the following objects:

- information about the parties to the transaction, which can accept, change or withdraw from the terms of the contract, using electronic digital signatures to identify themselves,
- the conditions and subject of the contract for innovative objects and objects of intellectual property, information about which is placed in a distributed registry,
- description of the procedures and operations of the contract execution in the form of a formalized algorithmic description that can be programmed in the blockchain environment.

The operation of the smart contract algorithm is based on the following technologies and platforms:

- digital signature and certification technologies using asymmetric encryption cryptographic algorithms,
- blockchain tools platforms like Codius, Ethereum, Counterparty,
- distributed database technologies for decentralized storage of transaction information.

It should be noted, the use of blockchain platforms for the development of decentralized applications that are similar to smart contracts, but are not only intended to provide contractual and financial obligations [14]. Such applications can be used to provide any informational interactions, are not limited to the parties to the transaction, and are used in various fields, for example, in online games.

All platforms can be conditionally divided into global and private ones. Global platforms make it possible for users and application developers to use an open blockchain network, which represents a network of peer-to-peer nodes, containing a transaction log replicated on many nodes.

The tools with which you can implement a smart contract include platforms Aeternity, Cardano, Ethereum, Hyperledger Fabric [15].

The Aeternity blockchain platform is based on the Lightning Network payment protocol [18, 19]. The protocol works with block chains to conduct fast transactions between nodes. However, he himself solves the problem of scaling and the developed algorithms do not affect the system performance. Computational load with the increase in the number of the transaction is transferred to the logical level, where transactions between participants are implemented in separate logical channels. The entire block chain is not involved, but is used as a distributed database to control financial transactions and as an arbitrator in case of disagreement to resolve disputes.

The Cardano system [21] was designed to transfer the cost of cryptocurrency with the scaling property. It represents the third generation of blockchain platforms. For the synthesis of a chain of blocks, the Haskell programming language is used. The main feature of the system is the logical separation of computational layers into the main layer for working with ADA cryptocurrency and the layer of synthesis and functioning of smart contracts. Here too, a special mechanism for reaching agreement through evidence of the share of ownership of an information object (cryptocurrency, intellectual property, etc.) is implemented. For example, for cryptocurrency, the number of units of a user's own cryptocurrency determines the probability of synthesizing its block in a chain. Thus, the owners of cryptocurrency can control all operations with cryptocurrency in the network.

The Ethereum system not only works with cryptocurrency, but is also intended for the synthesis and implementation of smart contracts, as well as for the development and implementation of decentralized applications [16]. Moreover, the platform can use the network for operations of any level of complexity through synthesized decentralized applications. In fact, the platform is a network virtual machine for conducting secure operations. The ability to create and use a smart contract is implemented by advanced tools and the presence of an embedded Solidity programming language [17], which allows developing smart contracts for various ownership conditions, with many

transaction formats and state transition functions. Smart contracts allow you to register transactions with assets in a distributed database, and security is ensured by using hashes in blockchain chains. Calculations hash of block sums is implemented on the computational tools of the users themselves.

The project Hyperledger Fabric is implemented with the support of IBM and JP Morgan and is designed to synthesize an open distributed registry for universal use [20]. The system allows you to develop decentralized applications with the possibility of multilayer blockchain configuration. As well as the Aeternity platform, the system implements custom transaction channels with increased data transfer speed, security and reliability.

3 Choosing a Blockchain Platform for Regional Innovation Systems

In the process of working on the project, a comparative analysis of these systems was made and tools were chosen for synthesizing smart contracts and supporting secure interactions between the participants of the innovation system based on the blockchain platform Ethereum. The platform contains many tools for solving specific tasks: CPP-Ethereum, Solidity, Remix, Webpack, Geth, Web3.js, Parity, etc.

The smart contract algorithm is programmed in the Solidity language as a decentralized application. This is an object-oriented language similar to javascript. Synthesis of the contract is performed in the Remix cloud environment. The environment implements the process of debugging the application by running the generated code directly in the browser. The code is broadcast and executed on the Ethereum virtual machine on distributed computing nodes.

As already mentioned, the system has implemented a way to prove that the Proof-of-Work has been completed in order to achieve consensus. The authenticity of completed transactions is confirmed by the computing power of network nodes. The disadvantage of the approach is that the probability of synthesizing a new unit directly depends on the power of the network node. Therefore, to eliminate this dependence, the Proof-of-Stake method is used, in which the probability of block formation is proportional to the share that the cryptocurrency calculation units belonging to the participant constitute of their total number.

The CPP-Ethereum, Geth and Parity tools are used to connect new network nodes to the circuit. They are loaded as clients on user nodes and are responsible for implementing the Ethereum protocol. Work with the blockchain can be performed via a website using special browsers or standard browsers with the installation of MetaMask and Mist extensions to execute programs and send commands. The Web3.js library provides operation with network nodes via the Remote Procedure Call (RPC) protocol via the HTTP protocol.

Another tool is the Truffle framework. It supports contract management, their placement on the blockchain and migration. To access the blockchain, network nodes are used that interact with each other via the Ethereum protocol.

4 Results

In the process of working on the synthesis of a smart contract algorithm to describe and support secure interactions of participants in a regional innovation system, components were developed for performing transactions. The main system requirements are:

- ensuring the conclusion of contracts,
- support the fulfillment of contractual obligations in the creation and transfer of innovative solutions,
- ensuring compliance with the terms of license agreements,
- implementation of the safe transfer of intellectual property rights,
- monitoring the use of intellectual property rights and licenses in innovation activities
- ensuring the transparency of transactions and the immutability of data on innovations and participants.

The main component is a distributed transaction database (registry) with digital descriptions of innovations and participants in interactions. The registry is a virtual chronological notary. The database describes the main transactions. Examples of transactions are: (a) granting access rights to the innovation passport; assessment of (b) innovation objects; (c) examination of innovations; (d) registration of intellectual property rights; (e) registration of innovation; (f) fixing the transfer of property rights; (g) financial transactions; (h) payment of remuneration; (i) intellectual property rights dispute, etc.

Smart contracts perform various tasks to support interaction and information exchange between users of the innovation system. These tasks include:

- Identification of participants and innovative assets;
- Synthesis of digital copies of innovative assets
- Confirmation of authenticity of documents;
- Verification and validation of intellectual property rights;
- Use of cryptocurrency for mutual settlements between participants;
- Raising funds to support the development, promotion and implementation of innovations;
- Maintaining a decentralized exchange of innovations, etc.

Let us consider the blockchain system performance by the example of issuance of a smart contract for transfer of rights on intellectual activity results (IAR) between innovation participants.

The process of working with a distributed register of innovations begins with the creation of a contract for each result of innovation activity (IAR). The contract is synthesized using a client web application. IAR information is entered into the blockchain and is a smart contract. According to this algorithm, only its owner of the

IAR can transfer the rights to the IAR into ownership to another user. The owner is specified in the "Current owner" variable. The owner pays a commission for placing information about the IAR in the registry. After the innovation description is placed in the registry, the IAR is assigned a unique hash identifier by calculating the hash function. To visualize the information on the IAR, the identifier is translated into a QR code, which is placed on the web page of the website with the name and a brief description of the information. This is how a showcase of innovative solutions is formed available for exchange or sale (Fig. 2).

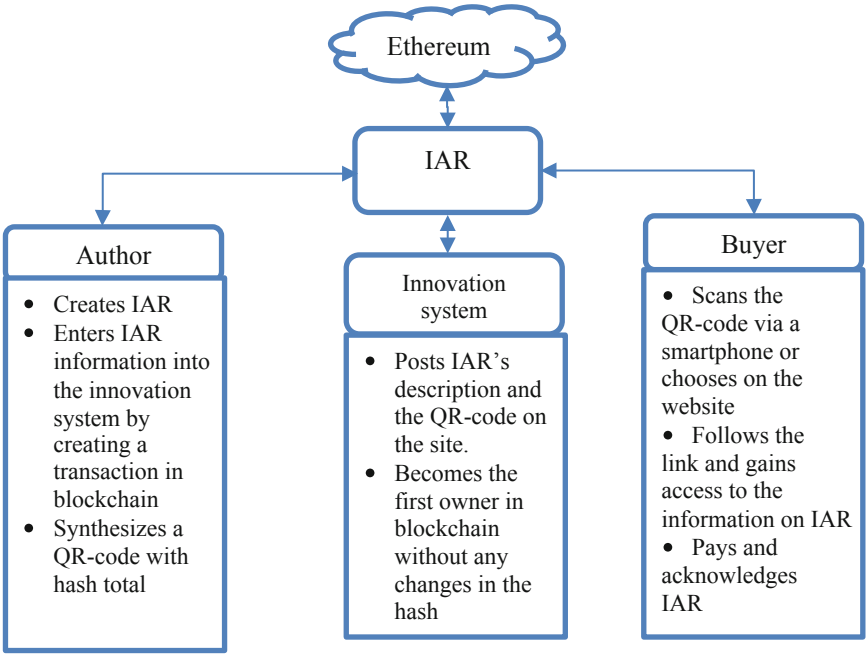

Fig. 2. A scheme of smart contract functioning for IAR

The new owner can access the detailed description of the innovation through the installed mobile application by scanning the QR code. Traditional access via a browser and search box is also supported. Available on request information about the IAR includes: description of information with drawings and diagrams, information about the author, information on intellectual property rights (patent, certificate of registration), date of creation and registration of the IAR in the registry, information on all owners of the IAR, etc.

For registration of the process of selling IAR and the transfer of intellectual property rights, a functionality has been implemented that establishes a new owner. When a IAR is transferred to the ownership of a new owner, the procedure for confirming ownership is implemented, and information about the new owner is added to the blockchain, after which the information block is synthesized and the new hash sum is calculated as an identifier of innovation. To confirm ownership, the new owner must also pay a reward to the previous owner and a commission for the transaction made.

The smart contract is synthesized using the tools of Git, VisualStudio Code, Ganache, Node.js v6 + LTS, truffle and web3.js libraries. For web applications, the lite-server web server is used. Compilation and debugging of smart contracts is implemented in the IDE Remix software.

The compiled and debugged smart contract is transferred to the blockchain system using Ethereum client Ganache. To access and work with the contract, you must add the Metamask extension to the standard browser.

Thus, the process of interaction of subjects of innovation in the region is recorded in the form of a smart contract, and all transactions between participants are recorded in the blockchain.

To complete the functions it is necessary to create a Product.sol smart contract in the Remix environment. The smart contract must include an IAR's creation function, a function of IAR's transfer to a new owner, a property rights confirmation function. To create IAR there is developed an IARItem function with such parameters as IAR's name, author, model (description), price, date, next owner's address.

Having synthesized IAR's description in blockchain the only one, who is authorized to transfer IAR to a new owner, is the one specified in _currentOwner. The function of IAR ownership transfer by a current owner is performed for an input variable _nextOwner with the next owner's address:

```
function setNextOwner(address
nextOwner) public returns(bool set) {
    if (_currentOwner != msg.sender) {
       return false;
    }
    _nextOwner = nextOwner;
    return true;
}
```

Ownership confirmation for the next IAR owner is performed through the function confirmOwnership, which features the new owner's name at the input. The IAR owners' array _owners is appended by a variable _nextOwner, and this address is assigned to a variable _currentOwner. Until the new owner remains unknown,

the variable _nextOwner has a zero address. The name of the new owner is added to the owners' array. In order for buyers to check authenticity of IAR the function of full IAR's owners list acquisition is provided for:

```
function confirmOwnership(string customer)
public returns(bool confirmed) {
    if (_nextOwner != msg.sender) {
        return false;
    }
    _owners.push(_nextOwner);
    _currentOwner = _nextOwner;
    _nextOwner = address(0);
    _customer.push(customer);
    return true;
}
```

Following the synthesis of main functions, the contract should be compiled using Truffle and its performance should be checked in the IDE Remix testing network. In the course of compilation a JSON file is created with Application Binary Interface (ABI) of the smart contract, which is used for the contract placement in blockchain.

The operation with smart contracts in the blockchain system requires a web interface based on the bootstrap library that includes the following pages: create.html – IAR creation, confirm.html – IAR receipt confirmation, etc. The work with the web interface and the interaction with blockchain are carried out by means of the web3.js library. The work with the smart contract requires an event service function. An example of a concrete event is the creation of an IAR's digital copy in blockchain. The creation algorithm can be launched by pressing the corresponding button on the web page. It includes the following steps:

- IAR Contract synthesis,
- Contract deployment in blockchain (specification of a contract owner's address, an IAR data line and an amount of gas required for the deployment. The term "gas" defines units of payment for completion of common tasks in blockchain. For example, an addition of numbers costs 3 gas, a multiplication – 5 gas. A transition of IAR ownership rights to another user demands a certain amount of gas. Using Metatask users set prices and amounts of gas),
- If a contract is successfully added in blockchain, it launches the smart contract's function ProductItem with parameters from input fields,
- A QR-code is generated by the qrcode.js library to identify IAR.

Below is the function of IAR's digital copy creation:

```
$(#makeIAR)'.click(function() {
   var IARContract = web3.eth.contract(abi);
   var IAR = IARContract.new(
   {
      from: web3.eth.accounts[0],
      data: '
0x6080604052348015610010576000 80fd5b50611253806... ',
      gas: '4700000'
   }, function (e, contract){
      console.log(e, contract);
      if (typeof contract.address !== 'undefined') {
         console.log('Contract mined! address: ' +
contract.address + ' transac-tionHash: ' +
contract.transactionHash);
         addr=contract.address;
         var code="http://localhost:3000/check.html#"+addr;
con-tract.IARItem($('#nameIAR').val(),$('#manufacturer').val
(),$('#modelIAR').val(),$('#priceIAR').val(),String($('#create
Date').val()),$('#nextOwner').val(),function(e,contract){
            $('#bigText').qrcode(code);
            console.log(e,contract);
      });
      }
   })
```

The function of IAR ownership reception is initiated by the corresponding button on the web interface. The IAR sale function operates with the new owner's address or in the absence of the latter with the address «0x0000000000000000000000000000000000000000».

The execution of transactions in functions cost gas with exception of the IAR checking function, which is launched when jumping to the page/check.html with the address of a contract to be checked and gives information on IAR as a result of script execution:

```
$('location='http://localhost:3000/app/view/check.html').click(func
tion(){
   var checkIAR = web3.eth.contract(abi);
   var check = checkIAR.at(codeIAR);
   check._nameProduct(function(error, result){
        if(!error)
           {
              $('#nameProduct').text(result);
           }
        else
           console.error(error);
     });
     check._model(function(error, result){
        if(!error)
           {
              $('#model').text(result);
           }
        else
           console.error(error);
     });
     check._color(function(error, result){
        if(!error)
           {
              $('#color').text(result);
           }
        else
           console.error(error);
     });
     check._createDate(function(error, result){
        if(!error)
           {
              $('#createDate').text(result);
           }
        else
           console.error(error);
     });
     check.getOwnersCount(function(error, result){
        if(!error)
           {
              var countOwner=result;
              $('#ownersCount').text(result);
              for (var i = 0; i < countOwner; i++) {
                 check.getOwners(i,function(error, result){
                    if(!error)
                       {
                          $('<div class=flex-container row'><span
id=owner'+i+'">'+result+'</span></div><br>').appendTo($('#ow
ners'));
                       }
                    else
                       console.error(error);
                 });
              }
           }
        else
           console.error(error);
     });
 });
```

Similarly, smart contracts are created to accomplish any operations on informational interaction of regional innovation system's participants. Having smart contracts created and deployed in the blockchain system it is necessary to run automatic testing implemented by the Truffle framework tools. The testing targets are individual functions of each smart contract, as well as its functioning in general and its web interface.

5 Conclusion

The article describes the process of a component of a smart contract for its use as a system to support the secure interaction of participants in innovative regional systems. In the course of the system development the following tasks were completed:

- adjustment of the smart contract environment based on the Ethereum platform, installation and testing of the required tools and creation of the blockhain testing network to work with smart contracts,
- development of the web interface to work with smart contracts,
- development of scripts running basic functions for a number of smart contracts. In particular, the intellectual property rights transfer contract,
- development of event processing scripts for users to work with smart contracts through the web interface, etc.

The technology of smart contracts on the blockchain platform represents a new mechanism for managing regional innovation development. This allows you to combine the efforts of scientific researchers and business to create and implement innovative solutions based on safe and transparent interaction within the framework of cooperation efforts to promote, commercialize innovative projects, reduce transaction costs of participants and investors.

Acknowledgments. The reported study was funded by RFBR according to the projects: № 18-010-00204-a, 18-07-00975-a, 19-013-00409-a.

References

1. Gamidullaeva, L.A.: About formation of innovation management system in Russia. Econ. Revival Russia **4**(50), 74–84 (2016)
2. Vasin, S.M., Gamidullaeva, L.A.: Development of Russian innovation system management concept. Innovations **5**(223), 34–40 (2017)
3. Gamidullaeva, L.A., Tolstykh, T.O.: Transaction costs, institutions and regional innovation development: the case of Russia. In: Proceedings of the 30th International Business Information Management Association Conference (IBIMA), Vision 2020: Sustainable Economic development, Innovation Management, and Global Growth, Madrid Spain, pp. 2121–2135, 8–9 November 2017
4. Finogeev, A.G.: Simulation of systems-synergistic processes in information environments, p. 223. Penza State University, Penza (2004)

5. Blockchain in Russia (2018). http://www.tadviser.ru/index.php/Статья:Блокчейн_в_России#cite_note-7. Accessed 17 April 2019
6. Swan, M.: Blockchain: Blueprint for a New Economy, 152 p. O'Reilly Media, Inc., Sebastopol (2015). ISBN 978-1-4919-2047-3
7. Franco, P.: The Blockchain. In: Understanding Bitcoin: Cryptography, Engineering and Economics, 288 p. Wiley, Hoboken (2014). ISBN 978-1-119-01916-9
8. Antonopoulos, A.M.: The Blockchain. In: Mastering Bitcoin. O'Reilly Media, Inc., Sebastopol (2014). ISBN 978-1-4493-7404-4
9. Matveev S.: Blockchain will provide a fair reward for scientists. https://4science.ru/articles/Sergei-Matveev-blokchein-obespechit-uchenim-spravedlivoe-voznagrazhdenie. Accessed on 17 April 2019
10. Official site of Ascribe company https://www.ascribe.io/. Accessed on 17 April 2019
11. In Russia may appear a blocking analogue of eBay in the field of intellectual property management. http://forklog.com/v-rossii-mozhet-poyavitsya-blokchejn-analog-ebay-v-sfere-upravleniya-intellektualnymi-pravami/. Accessed on 17 April 2019
12. Szabo, N.: Smart contracts: formalizing and securing relationships on public networks. First Monday **2**, 9 (1997). http://firstmonday.org/ojs/index.php/fm/article/view/548/469
13. Smart Contracts, Explained. Partnership Material. https://cointelegraph.com/explained/smart-contracts-explained. Accessed on 17 April 2019
14. Decentralized platforms for smart contracts: challenges and solutions. https://forklog.com/detsentralizovannye-platformy-dlya-smart-kontraktov-vyzovy-i-resheniya/. Accessed on 17 April 2019
15. Blockchain platforms. http://smart-contracts.ru/platforms.html. Accessed on 17 April 2019
16. Blockchain app platform. https://www.ethereum.org/. Accessed on 17 April 2019
17. Solidity is a contract-oriented, high-level language for implementing smart contracts. http://solidity.readthedocs.io/en/v0.4.24/
18. Aeternity blockchain. https://aeternity.com/. Accessed on 17 April 2019
19. Poon, J., Dryja, T.: The Bitcoin Lightning Network: scalable off-chain instant payments (2016). http://lightning.network/lightning-network-paper.pdf
20. Hyperledger Fabric is a platform for distributed ledger solutions. http://hyperledger-fabric.readthedocs.io/en/release-1.1/. Accessed on 17 April 2019
21. Cardano is a decentralised public blockchain and cryptocurrency project and is fully open source. https://www.cardano.org/en/home/. Accessed on 17 April 2019