

**Міністерство освіти і науки України
Одеський національний технологічний університет
Вінницький національний технічний університет
Інститут комп'ютерних систем і технологій
"Індустрія 4.0" ім.П.Н.Платонова**

**II Всеукраїнська науково-технічна конференція
молодих вчених, аспірантів та студентів**

**«КОМП'ЮТЕРНІ ІГРИ ТА МУЛЬТИМЕДІА ЯК
ІННОВАЦІЙНИЙ ПІДХІД ДО КОМУНІКАЦІЇ»**

Матеріали конференції



Одеса

29-30 вересня 2022 р.

Комп'ютерні ігри та мультимедіа як інноваційний підхід до комунікації / Матеріали II Всеукраїнської науково-технічної конференції молодих вчених, аспірантів та студентів. Одеса, 29-30 вересня 2022 р. - Одеса, Видавництво ОНТУ, 2022 р. – 178 с.

Збірник включає матеріали доповідей учасників конференції, які об'єднані за тематичними напрямками конференції.

ОРГАНІЗАЦІЙНИЙ КОМІТЕТ

Голова - Богдан Єгоров, президент ОНТУ

Заступники голови:

Наталя Поварова, проректор з наукової роботи, ОНТУ,

Сергій Котлик, директор навчально-наукового інституту Комп'ютерних систем і технологій «Індустрія 4.0» ім. П.Н. Платонова, ОНТУ,

Сергій Шестопапов, декан факультету Комп'ютерної інженерії, програмування і кіберзахисту, ОНТУ

Члени комітету:

Олексій Ізвалов, регіональний координатор Global Game Jam в Східній Європі, ЕТІ ім.Ельворті,

Сергій Артеменко, зав.каф. Комп'ютерної інженерії, ОНТУ,

Михайло Кисленко, Unity Developer, DAL'S Games,

Олександр Романюк, зав.каф. Програмного забезпечення, ВНТУ,

Ольга Чолишкіна, директор Інституту комп'ютерно-інформаційних технологій і дизайну, МАУП,

Олександр Терьшин, Unity 3d developer, BlueGoji,

Валерій Плотніков, зав.каф. Інформаційних технологій і кібербезпеки, ОНТУ,

Павло Івасюк, Senior Snapchat JS Developer, BeVisioned,

Петро Горват, зав.каф. Комп'ютерних систем і мереж, ДВНЗ "Ужгородський національний університет".

Матеріали подано українською та англійською мовами.

Редактор збірника Котлик С.В.

**СПИСОК
організацій, представники яких взяли участь у роботі конференції**

Turan University, Almaty, Republic of Kazakhstan
University of food technologies, Plovdiv, Bulgaria
V.N. Karazin Kharkiv National University
Відокремлений структурний підрозділ "Фаховий коледж промислової автоматичної та інформаційних технологій ОНТУ"
Відокремлений структурний підрозділ «Одеський технічний фаховий коледж ОНТУ»
Вінницький національний технічний університет
Волинський національний університет імені Лесі Українки
ДВНЗ «Приазовський державний технічний університет»
Державний торговельно-економічний університет
Донецький національний медичний університет
Донецький національний університет імені Василя Стуса
Економіко-технологічний інститут імені Роберта Ельворті
Запорізький національний університет
Київський національний економічний університет імені Вадима Гетьмана
Київський національний університет технологій та дизайну
Книжкова палата України ім. Івана Федорова
Мелітопольський державний педагогічний університет ім. Богдана Хмельницького
Науково-дослідний інститут інтелектуальної власності Національної академії правових наук України
Національна академія сухопутних військ імені гетьмана П. Сагайдачного
Національний авіаційний університет
Національний лісотехнічний університет України
Національний технічний університет «Харківський політехнічний інститут»
Національний університет «Львівська політехніка»
Національний університет «Полтавська політехніка імені Юрія Кондратюка»
Національний університет харчових технологій
Одеська національна морська академія
Одеський національний технологічний університет
Одеський національний університет імені І. І. Мечникова
Первомайська гімназія №2 Первомайської міської ради Миколаївської обл.
Українська академія друкарства
Хмельницький національний університет
Центральноукраїнський інститут розвитку людини Відкритого міжнародного університету розвитку людини «Україна»

гейміфікації в інформаційних системах управління персоналом. (Вінницький національний технічний університет)	
Кудряшова А. В. Аналіз факторів впливу на рівень читацького попиту. (Українська академія друкарства)	70
Пилюченко Д.В., Бевзо Ф.О. Free-to-pay in free-to-play або дорогий безплатний геймінг. (Науково-дослідний інститут інтелектуальної власності Національної академії правових наук України)	73
Стогул В.М., Болтач С.В. Аналіз бізнес-моделей різних підходів монетизацій безкоштовних ігор. (Одеський національний технологічний університет)	76
Розділ 4. Технології (віртуальна реальність, доповнена реальність, інтернет речей, пристрої, що носяться, штучний інтелект, машинне навчання)	79
Viktoria Boichuk. Analysis of embedded software for professional nail decoration. (Ukrainian Academy of Printing)	79
Fedossov Y.V., Belov A.M., Ismailova R.T. Video game development with Unity. (Turan University, Almaty, Republic of Kazakhstan)	81
Kopp A.M., Shynkarenko D.V. Smart contract code generation based on natural language business rules for cryptocurrency tokens creation. (National Technical University «Kharkiv Polytechnic Institute»)	83
Mamyrova A.K., Makulbekov T.N. Optimization of test scenario for software autotest systems. (Turan University, Almaty, Republic of Kazakhstan)	86
Mamyrova A.K., Tokmashov D.S. Development of mobile application "Gostestnik". (Turan University, Almaty, Republic of Kazakhstan)	87
Moldakalykova B., Bimoldina Zh., Askarbek A. Python as an Android application programming tool. (Turan University, Almaty, Republic of Kazakhstan)	90
Turabayev A.T., Ismailova R.T. Development of a website to promote the services of the company IE «TAT». (Turan University, Almaty, Republic of Kazakhstan)	92
V.Voedilo. Spatial modeling and research of machine park components of operational printing. (Ukrainian Academy of Printing)	95
Азархов О.Ю., Сілі І.І. IoT фетальний пульсометр на базі ESP32. (ДВНЗ «Приазовський державний технічний університет»)	98
Alekseienkova D.S. Machine learning in game development. (V. N. Karazin Kharkiv National University)	100
Астахов В.І., Болтач С.В. Порівняльний аналіз використання доповненої та віртуальної реальності в сфері розробки ігор. (Одеський національний технологічний університет)	101
Буруков О.В., Жуковецька С.Л. Характерні механіки комп'ютерних ігор жанру «Slasher». (Одеський національний технологічний університет)	104
Варіс І.О., Саврасов Я.К. Використання віртуальної реальності в менеджменті персоналу. (Київський національний економічний університет)	105

SMART CONTRACT CODE GENERATION BASED ON NATURAL LANGUAGE BUSINESS RULES FOR CRYPTOCURRENCY TOKENS CREATION

KOPP A.M. (kopp93@gmail.com), SHYNKARENKO D.V. (shinkarenko.cmds@gmail.com)
National Technical University «Kharkiv Polytechnic Institute»

Abstract. Nowadays, blockchain and smart contracts are emerging fields used by fintech and technology-oriented companies to create their cryptocurrency tokens for voting rights, rewards, tokenized assets, and other utilities. However, blockchain development is a complex technology area that requires specific skills, while attracting people and organizations that do not have such skills. Hence, this paper is focused on simplifying the process of developing software components for decentralized systems by building smart contracts based on business rules formulated in natural language.

Problem statement. Information systems require natural language processing techniques because an unprecedented volume of data is currently kept in text format, making it difficult to extract knowledge and further present it. Unquestionably, one of the most significant user interfaces of the future is natural language. Even now, the services that are provided to users lag far behind those that information systems, particularly decentralized ones, are capable of providing. Technology that enables effective communication between the user and the computer is the answer to this issue. The most recent method for resolving this issue is smart contracts [1].

The following are the key benefits of smart contracts. Security: the smart contract is encrypted and kept on several devices thanks to the usage of blockchain technology, which protects against loss or illegal change. Cheapness and quickness: the majority of processes are automated, and the majority of intermediates are removed. Standardization: there are numerous smart contract solutions available now, and users can select the best one for a given task. Smart contracts, however, have several drawbacks: the smart contract is stored in the blockchain, which means that it cannot be updated to erase faults. However, because the contract is a code developed by humans, it may have errors, which is crucial when working with cryptocurrencies [1].

Fig. 1 depicts the market for blockchain technology as it has grown [2].



Figure 1 – The volume of the global blockchain technology market from 2016 to 2021 (in millions of US dollars) [2]

Today, the Ethereum blockchain core is most frequently used as the foundation for decentralized networks. Ethereum's public test networks use various Proof-of-Authority (PoA) and Proof-of-Stake (PoS) consensus models, while the public Ethereum network uses Proof-of-Work (PoW) consensus. Such technologies are the most advanced for building blockchains with all types

of consensus and fully developed, multipurpose smart contracts [3]. They also meet the toughest security requirements and have been tested in dozens of actual functioning networks.

Ethereum makes use of smart contracts that are typically created in the Solidity programming language and the Ethereum Virtual Machine (EVM). With a substantial quantity of pre-written code and development templates, EVM has long since established itself as the industry standard for virtual machines. The security of EVM contracts is now quite high [4] since the EVM contract code is in charge of enormous sums of cryptocurrency and any breach discovered results in a major community and media response.

As it turned out, people are significantly more interested in learning how to write smart contracts without having any prior knowledge of blockchain development technology than they are, however, in learning the programming language Solidity, which is frequently used for that purpose. Fig. 2 depicts the interest graph of online users.



Figure 2 – “Create token” (blue) and “Learn solidity” trend graphs (red) for Google searches

Hence, the generation of the smart contract based on business rules using natural language processing (NLP) is studied in this paper. Given that ERC20 [5] is the most widely used smart contract standard, its developers – financial technology companies and other tech-focused businesses – can design their cryptocurrency tokens with a particular utility in consideration, such as voting rights, rewards, smart property, tokenized assets, etc. Therefore, the generation of ERC20 smart contracts suitable for the EVM blockchain platform is considered in this study.

Study aim and tasks. The object of work is the process of creating smart contracts based on business rules using NLP. The subject of work includes software components for building smart contracts based on business rules using natural language processing. The purpose of the work is to simplify the process of developing software components for decentralized systems by building smart contracts based on business rules formulated in natural language. Hence, we need to determine core functional and non-functional requirements for software components for building smart contracts based on business rules using NLP, and model use cases of the future software solution.

Research results. In this work, it is necessary to develop software components that will allow the generation of smart contracts under the requirements expressed in natural language using NLP techniques. Therefore, the following functional requirements (FR) were formulated:

- FR1. A user who does not have an account yet should be able to sign up.
- FR2. A user who has an account should be able to sign in to the system.
- FR3. A user authorized in the system should be able to enter business rules in English.
- FR4. Business rules entered by a user should be processed using NLP techniques.
- FR5. Based on the entered business rules, the Solidity smart contract should be generated.
- FR6. A user should be able to deploy a smart contract to the one of EVM-based blockchains.
- FR7. A user should be able to interact with the deployed smart contract.
- FR8. A user should be able to switch between EVM-compatible blockchain networks.
- FR9. User data, business rules, and generated smart contracts should be stored in the database.

Also, the following non-functional requirements (NFR) were formulated:

NFR1. The developed software should have a clear user interface that will allow anyone to launch it and use it without special training.

NFR2. The developed software should retain its functionality under the invalid entered data and inform the user about occurring errors.

NFR3. Uninterrupted operation of the developed software should be ensured.

The use case diagram created using the Unified Modeling Language (UML), is shown in Fig. 3.

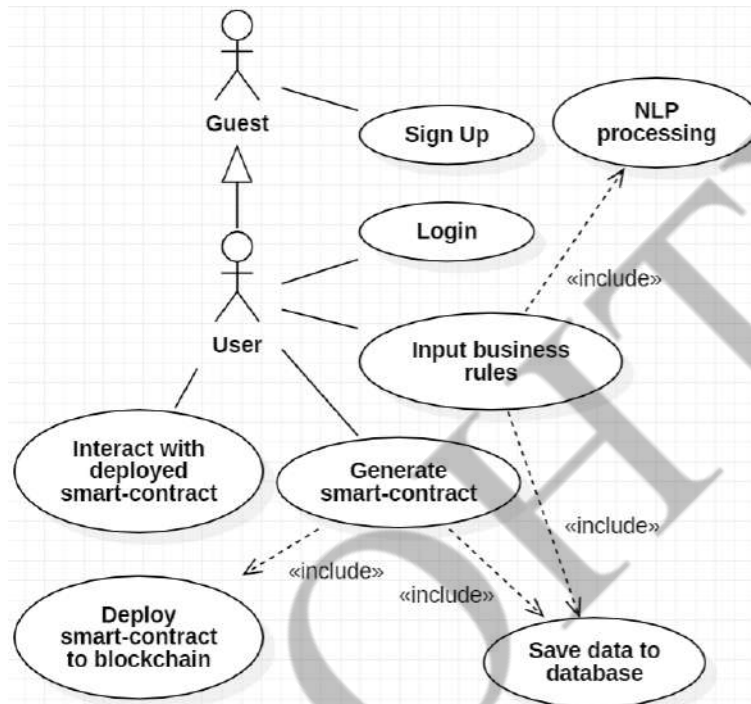


Figure 3 – The UML use case diagram

Hence, the future work includes the following tasks that should be solved:

- analyze and improve business processes related to the construction of smart contracts;
- select the type of software components architecture and design its structure;
- develop software components for building smart contracts based on business rules using NLP;

- assess performance and efficiency of developed software components;
- assess the quality of produced smart contract code.

Conclusion. In this paper, we analyzed existing software solutions for building smart contracts based on business rules using natural language processing. We have determined the main functional and non-functional requirements of software components for building smart contracts based on business rules using NLP techniques. The use case diagram that models future user interaction with the software is created. Therefore, in the future, it is necessary to use natural language analysis to determine key terms based on the use of appropriate NLP approaches. It is necessary to develop an algorithm for generating the source code of a smart contract in the Solidity language based on the defined key terms. It is also necessary to demonstrate the functionality of the developed software on a sample set of business rules and to analyze the obtained results.

References.

- [1] S. Nakamoto, “Bitcoin: A peer-to-peer electronic cash system”, in *Decentralized Business Review*, 2008, p. 21260.
- [2] M. Möser et al., “An empirical analysis of traceability in the monero blockchain”, in *arXiv preprint*, arXiv:1704.04299, 2017.

[3] M. C. Suciu, C. Năsulea, D. Năsulea, “Is Blockchain a New Creative Industry?”, in *2nd International Conference on Economics and Social Sciences*, 2019, pp. 5–12.

[4] Y. Hirai, “Defining the ethereum virtual machine for interactive theorem provers”, in *International Conference on Financial Cryptography and Data Security*. Springer, Cham, 2017, pp. 520–535.

[5] “ERC20.” [Online] Available: <https://docs.openzeppelin.com/contracts/4.x/erc20>

UDC 004.912

OPTIMIZATION OF TEST SCENARIO FOR SOFTWARE AUTOTEST SYSTEMS

MAMYROVA A.K., MAKULBEKOV T.N.

(a.mamyrova@turan-edu.kz, 21220933@turan-edu.kz)

Turan University, Almaty, Republic of Kazakhstan

The article discusses the types of system testing and the effective use of scenario in automated testing systems through optimization.

Keywords: testing; autotest

Self-test software optimization plays an important role in improving the quality of business management, improves quality indicators, optimally plans and distributes company resources, improves the efficiency of self-test software, which together leads to the provision of high-quality and convenient product / service for the client, thereby increasing the growth of loyal customers.

There are different approaches to optimization, ranging from building block diagrams, identifying strengths and weaknesses, designating tasks and limits (temporary and financial), analyzing information resources, and ending with process automation.

Reducing the time spent on collecting and processing systematic tasks is a small plus of automation. Automation also allows you to integrate business processes into a coherent environment. Thanks to automation, the management team has the opportunity to set tasks for employees and control the quality of their implementation, create transparent reporting, monitor the importance and interconnection of chain processes. This process allows you to objectively analyze the business process as a whole and provide timely feedback on negative deviations.

At the present stage, there are several categories of automation areas: business process management; task planning and internal communication; work with human resources/personnel; communication with distributors and sales agents; attraction and communication with consumers, advertising analytics. Depending on the category and goals of automation, developers provide services with certain functionality.

Automated testing framework

The testing framework[1] is a user guide for test engineers to support the efficient execution of scenario tests. For each type of testing, a test scenario, principles and rules for its implementation should be defined.

There are various frameworks depending on the purpose of testing. Figure-1 shows the step-by-step scenario for creating or checking the status of a client. This script is written in the JUnit framework on Java 8.

**II Всеукраїнська науково-технічна конференція
молодих вчених, аспірантів та студентів**

**«КОМП'ЮТЕРНІ ІГРИ ТА МУЛЬТИМЕДІА ЯК
ІННОВАЦІЙНИЙ ПІДХІД ДО КОМУНІКАЦІЇ»**

Одеса

29-30 вересня 2022 р.

Збірник включає доповіді учасників конференції. Тези доповідей публікуються у вигляді, в якому вони були подані авторами.

Відповідальність за зміст і форму подачі матеріалу несуть автори статей.

Редакційна колегія: Котлик С.В., Шестопапов С.В.,
Корнієнко Ю.К.

Комп'ютерний набір і верстка: Соколова О.П.

Відповідальний за випуск: Котлик С.В.