



International Science Group

ISG-KONF.COM

|

**INTERNATIONAL SCIENTIFIC AND PRACTICAL
CONFERENCE "PROBLEMS OF MODERN SCIENCE AND
PRACTICE"**

Boston, USA

September 21 – 24, 2021

ISBN 978-1-63972-061-3

DOI 10.46299/ISG.2021.II.I

PROBLEMS OF MODERN SCIENCE AND PRACTICE

Abstracts of I International Scientific and Practical Conference

Boston, USA
September 21 – 24, 2021

Library of Congress Cataloging-in-Publication Data

UDC 01.1

The I International Science Conference «Problems of modern science and practice»,
September 21 – 24, 2021, Boston, USA. 487 p.

ISBN - 978-1-63972-061-3

DOI - 10.46299/ISG.2021.II.I

EDITORIAL BOARD

<u>Pluzhnik Elena</u>	Professor of the Department of Criminal Law and Criminology Odessa State University of Internal Affairs Candidate of Law, Associate Professor
<u>Liubchych Anna</u>	Scientific and Research Institute of Providing Legal Framework for the Innovative Development National Academy of Law Sciences of Ukraine, Kharkiv, Ukraine, Scientific secretary of Institute
<u>Liudmyla Polyvana</u>	Department of Accounting and Auditing Kharkiv National Technical University of Agriculture named after Petr Vasilenko, Ukraine
<u>Mushenyk Iryna</u>	Candidate of Economic Sciences, Associate Professor of Mathematical Disciplines, Informatics and Modeling. Podolsk State Agrarian Technical University
<u>Oleksandra Kovalevska</u>	Dnipropetrovsk State University of Internal Affairs Dnipro, Ukraine
<u>Prudka Liudmyla</u>	Odessa State University of Internal Affairs, Associate Professor of Criminology and Psychology Department
<u>Slabkyi Hennadii</u>	Doctor of Medical Sciences, Head of the Department of Health Sciences, Uzhhorod National University.
<u>Marchenko Dmytro</u>	Ph.D. in Machine Friction and Wear (Tribology), Associate Professor of Department of Tractors and Agricultural Machines, Maintenance and Servicing, Lecturer, Deputy dean on academic affairs of Engineering and Energy Faculty of Mykolayiv National Agrarian University (MNAU), Mykolayiv, Ukraine
<u>Harchenko Roman</u>	Candidate of Technical Sciences, specialty 05.22.20 - operation and repair of vehicles.
<u>Belei Svitlana</u>	Ph.D. (Economics), specialty: 08.00.04 "Economics and management of enterprises (by type of economic activity)"

91.	Tvoroshenko I., Maksimenko H. TO THE QUESTION OF ANALYSIS OF EXISTING MECHANISMS OF WEB APPLICATION TESTING	403
92.	Tvoroshenko I., Bielinskyi Y. ON THE FEATURES OF METHODS OF PROCESSING AND RECOGNITION OF HANDWRITTEN TEXT	410
93.	Tvoroshenko I., Kuznetsov M. ABOUT THE ROLE OF TESTING IN PROCESS OF MOBILE APPLICATION DEVELOPMENT	416
94.	Voskoboinick V., Turick V., Voskobiinyk A., Voskoboinyk O. VORTICITY FIELD INSIDE CROSS-STREAMLINED SEMI-CYLINDRICAL TRENCH	422
95.	Yevsieiev V., Demska N. STUDY OF THE STRUCTURE OF CYBER-PHYSICAL PRODUCTION SYSTEMS IN INDUSTRY 4.0	431
96.	Бандура В. АНАЛІЗ НОВІТНІХ РОЗРОБОК ВИГОТОВЛЕННЯ ПРОДУКТІВ ФУНКЦІОНАЛЬНОГО ПРИЗНАЧЕННЯ НА ОСНОВІ ПЕЧІНКОВОГО ПАШТЕТУ	435
97.	Білюк І.С., Савченко О.В., Шарейко Д.Ю., Гаврилов С.О., Фоменко А.М. БЛОК ЖИВЛЕННЯ СИЛОВИХ ЛАНЦЮГІВ ПРОМИСЛОВОГО УСТАТКУВАННЯ	439
98.	Дудников В.С. АВТОПОДАТЧИК ЗАГОТОВОК ДЛЯ ДЕРЕВООБРАБАТЮВАЮЩИХ СТАНКОВ	442
99.	Корчак М.М. МАТЕМАТИЧНА МОДЕЛЬ ТЕХНОЛОГІЧНОГО СПОСОБУ ОБРОБІТКУ ҐРУНТУ	452
100.	Лях К.С., Егорова О.А., Алексеев Г.В. СОВЕРШЕНСТВОВАНИЕ ТЕХНОЛОГИИ ПОЛУЧЕНИЯ ПЕКТИНА ПРИ ИНТЕНСИФИКАЦИИ ПРОЦЕССА ЭКСТРАКЦИИ РАСТИТЕЛЬНОГО СЫРЬЯ	458

ABOUT THE ROLE OF TESTING IN PROCESS OF MOBILE APPLICATION DEVELOPMENT

Tvoroshenko Iryna,

Ph.D., Associate Professor
Kharkiv National University of Radio Electronics

Kuznetsov Maksym,

Master in informatics
Kharkiv National University of Radio Electronics

Due to the development and innovations of the information technologies in the various sides of human life, new challenges arise for software developers [1-3].

Software products become multicomponent and require a specialized approach for their development and testing. Today, the most popular is the creation of modern mobile devices with much functionality for its users [4-8].

In this case, mobile application development means creating the technical architecture of the application and its design, creating a web portal for data management and the corresponding API, testing the application and web portal, eliminating detected functional errors, as well as technical support and other tasks provided by the attached specification

Each software must be tested before it can be used [9, 10]. Due to the rapid development of IT technologies, software quality assurance specialists regularly face the question of streamlining the testing process, which takes into account the entire process of rational allocation of resources, defines mechanisms and methods of testing, and establishes the sequence of all actions within the software life cycle, as well as a reasonable description of the start stage and completion criteria of testing.

If the successful launch of a software product is a higher priority for the customer than high development speed and cost savings, testing should be an integral part of each phase of the software development lifecycle. Given the complexity of modern programs, it is difficult to imagine a situation where the development process goes without any testing. However, from time to time you may encounter a situation where software developers fail due to the fact that the testing is included in the overall development process only as one of the individual stages of the overall process, separate from the overall cycle [11].

Based on the features of the modern software, such as complex architecture, neglect of the testing during the development lifecycle almost always has a negative impact on the quality of security, performance, or functionality of the final product. The question of how a development company can ensure the best quality of a developed product is quite natural.

It should be noted that the testing conducted at all stages of development can significantly improve the quality, reliability and performance of the system. During the

testing, the team of testers makes sure that the software product performs all the documented functions properly and does not do what it should not do [12-15].

To ensure the high quality of the final product, it is critical to include the testing in the software development lifecycle. Of particular importance is the introduction of the testing in the early stages of work on the project, as this approach can significantly reduce the cost of eliminating the identified errors [16-19].

The main advantages of implementing the testing in the software life cycle:

- The testing introduced in the early stages of development significantly reduces the cost of error correction;

- Given the peculiarities of the modern market, only high-quality products are competitive. Thus, when investing resources in software development, you need to make every effort to ensure that the final product will be able to compete with existing solutions. Software testing throughout the development cycle is one of the key factors in ensuring the proper quality of the end result;

- The environment used in software development is different from the one in which the final product will be used. Testing thus allows you to test the software in real-world conditions.

Depending on the chosen software development methodology, the full life cycle may consist of a different number of stages. To simplify, consider the four main stages of development specific to any project:

- Creation of requirements to the project;

- Analysis and design;

- Development;

- Pre-closing and deployment.

The main task of formation of the project's requirements stage is to collect business requirements for the final product. In general, such requirements are as follows: it is determined who will use the application, access to which data is needed and how they will be used [20].

This step tests the requirements. The main goal is to identify errors and inconsistencies in the business logic of software at an early stage of the life cycle. Once the complete documentation has been compiled, the team of testers can evaluate it according to the following criteria:

- Completeness;

- Redundancy;

- Unambiguity;

- Lack of contradiction;

- Ranking;

- Verifiability.

Therefore, careful testing of the documentation allows to detect errors in the initial stages, which leads to a reduction in the cost of their correction, and hence the total cost of the development. Quality documentation reduces the complexity of the project and reduces the overall development time. Unambiguous and complete business requirements allow the development team to better assess the scope of work and work out the terms of reference. Due to a careful detailing and reduction of the risk component of the project, the total cost of development is reduced.

At the stage of analysis and design, the documentation is compiled at the preliminary stage of the software life cycle. Based on it, a layout of a mobile application is created, as well as the architecture of the future software product is designed.

Creating and testing a prototype helps to assess the quality of the future software product and its commercial prospects. Studying the prototype at the initial stage of the project development allows you to make the necessary changes in accordance with the objectives. By testing the software product at the prototype stage, you can save time and reduce costs, as the project will be carefully worked out according to the specification before the development team begins to write the code. The team of testers pays special attention to the detection of logical errors in the prototype, which can lead to the failure of the entire system in the event of their migration to the next stages of development. In addition, the prototype is compared with similar software products, which offers options for its improvement. Testing with the prototype allows you to calculate the approximate cost at each stage of product development and choose the most effective development methodology [21].

The usability audit of the prototype allows you to assess the usability of the future software product. The team of testers carefully examines the prototype provided by the customer and, if necessary, offers recommendations for its improvement. As a result, it is possible to increase the usability of the software by the end user. Such testing usually takes place in several steps:

- Testing of a rough schematic prototype at the initial stage of design;
- Testing of a prototype of average detailing;
- Testing of the exact finished prototype of the approved design.

One of the main advantages of the usability audit at an early stage of development is a significant reduction in the cost of error correction compared to later stages.

At the development stage, the source code of the future software product is written. Depending on the development methodology, there could be, for example, the following steps:

– Component testing allows you to thoroughly check each component of the software (for example, object, module, class) and make sure that it works. Such testing verifies the functionality that is added during the software development. Each component is tested in isolation, in an artificial environment. A list of approved testing requirements is used as a basis for conducting modular testing. The team of testers creates a list of test cases describing the relevant steps and expected results. There is also a list of software usage scenarios that describe the sequence of user actions and the expected response of the system to them. Component testing is performed during the development of each individual module of the system. Thus, in case of errors, you will need to change the design for only the specific module under test, and not the system as a whole. This type of testing allows you to identify deficiencies in the specification or in architecture of the program, as well as to assess the performance of individual parts of the product at each stage of development;

– Automation of the testing is introduced in cases where the tester is dealing with a large project, which leads to a very large number of inspections. The team of testers creates test cases for the program and a checklist of necessary checks, on the basis of which functional tests are created. These tests are run daily or automatically at certain

times of the day, such as at night. At the same time the report on the passed tests is automatically generated. The testing team supports already written tests, as well as creates new ones as needed. Errors detected during this testing would be much more difficult to detect manually. It speeds up the process and quality of complex computational tests, such as checking complex formulas. The accuracy and reliability of tests increases.

The final stage of the software development life cycle usually consists of two stages: beta-deployment and final deployment. The pre-deployment phase is necessary so that the test team has the opportunity to catch bugs in the product before it is released to the market. The test results can be used by the development team to make the latest adjustments before the final deployment of the product. At this stage, the team of testers checks the correctness of the correction of the errors detected in the previous stages, as well as tests the resistance of software to high loads and hacking [3, 10].

Therefore, considering all the features of testing mobile applications, you can determine the following algorithm:

1. Planning testing at the formal level.
2. Determining the required types of testing.
3. Preparation of a test data set and development of test scenarios.
4. Performing manual and automated testing.
5. Ease of use testing and beta testing.
6. Performance testing.
7. Pre-final testing and security testing.
8. Mobile device testing.

Thus, testing of applications on mobile devices generally corresponds to the general principles of testing, but there are a number of features which are characteristic of testing of mobile applications.

To understand the features of testing applications on mobile devices, it is necessary to consider the factors that distinguish a mobile application from a computer, namely: specific operating systems for mobile platforms, various configurations of components, functionality of such devices.

The tools for automation of the testing are not fully implemented to this day for the following reasons:

1. Wrong attitude to testing, as many company executives believe that a developer can write a program that does not contain errors.
2. High cost of the testing automation tools.
3. The desire to save on qualified personnel (the salary of a specialist in testing automation is higher than that of a regular tester).
4. Time limits (testing automation requires significant time and is recommended only if the project is at least medium-term).
5. Unsuccessful experience in the use of such tools and the expectation of an immediate effect from their implementation.

It should be noted that one of the main problems of automated testing is its complexity: even though it eliminates many operations and speeds up the execution of tests, large resources can be spent on updating the tests themselves. Changing the

interface of the program requires rewriting all the tests that are associated with the updated forms, with a large number of tests involving significant resources.

The current dynamics of finding modern solutions to the problem of testing automation is extremely useful for other areas of software development, as well as for the end user, as more and more sophisticated and at the same time universal tools are constantly created to check the quality and perfection of the program code.

References:

1. Daradkeh Y.I., Tvoroshenko I., Gorokhovatskyi V., Latiff L.A., and Ahmad N. (2021) Development of Effective Methods for Structural Image Recognition Using the Principles of Data Granulation and Apparatus of Fuzzy Logic, *IEEE Access*, 9, pp. 13417-13428.
2. Kobylin O., Gorokhovatskyi V., Tvoroshenko I., and Peredrii O. (2020) The application of non-parametric statistics methods in image classifiers based on structural description components, *Telecommunications and Radio Engineering*, 79(10), pp. 855-863.
3. Yousef Ibrahim Daradkeh, and Iryna Tvoroshenko (2020) Application of an Improved Formal Model of the Hybrid Development of Ontologies in Complex Information Systems, *Applied Sciences*, 10(19). p. 6777.
4. Asaad Ma. Babker, Abd Elgadir A. Altoum, Irina Tvoroshenko, and Vyacheslav Lyashenko (2019) Information Technologies of the Processing of the Spaces of the States of a Complex Biophysical Object in the Intellectual Medical System HEALTH, *International Journal of Advanced Trends in Computer Science and Engineering*, 8(6), pp. 3221-3227.
5. Gorokhovatskyi V.O., Tvoroshenko I.S., and Vlasenko N.V. (2020) Using fuzzy clustering in structural methods of image classification, *Telecommunications and Radio Engineering*, 79(9), pp. 781-791.
6. Tvoroshenko I.S., and Gorokhovatsky V.O. (2020) Effective tuning of membership function parameters in fuzzy systems based on multi-valued interval logic, *Telecommunications and Radio Engineering*, 79(2), pp. 149-163.
7. Tvoroshenko I.S., and Kramarenko O.O. (2019) Software determination of the optimal route by geoinformation technologies, *Radio Electronics Computer Science Control*, 3, pp. 131-142.
8. Kucherenko Y.I., Filatov V.A., Tvoroshenko I.S., and Baidan R.N. (2005). Intellectual technologies in decision-making technological complexes based on fuzzy interval logic. *East European Journal of Advanced Technologies*, 2, 92-96.
9. Tvoroshenko I.S., and Gorokhovatsky V.O. (2019) Modification of the branch and bound method to determine the extremes of membership functions in fuzzy intelligent systems, *Telecommunications and Radio Engineering*, 78(20), pp. 1857-1868.
10. Gorokhovatskyi V., and Tvoroshenko I. (2020) Image Classification Based on the Kohonen Network and the Data Space Modification, *In CEUR Workshop Proceedings: Computer Modeling and Intelligent Systems (CMIS-2020)*, 2608, pp. 1013-1026.

11. Matarneh Rami, Tvoroshenko Irina, and Lyashenko Vyacheslav (2019) Improving Fuzzy Network Models For the Analysis of Dynamic Interacting Processes in the State Space, *International Journal of Recent Technology and Engineering*, 8(4), pp. 1687-1693.
12. Irina Tvoroshenko, M. Ayaz Ahmad, Syed Khalid Mustafa, Vyacheslav Lyashenko, and Adel R. Alharbi (2020) Modification of Models Intensive Development Ontologies by Fuzzy Logic, *International Journal of Emerging Trends in Engineering Research*, 8(3), pp. 939-944.
13. Кучеренко Є.І., Творошенко І.С. (2011) Оперативне оцінювання простору станів складних розподілених об'єктів з використанням нечіткої інтервальної логіки. *Штучний інтелект*, 3, С. 382-387.
14. Кучеренко Е.И., Корниловский А.В., Творошенко И.С. (2010) О методах настройки функций принадлежности в нечетких системах. *Системы управления, навигации и связи*, 1(13), С. 94-98.
15. Кучеренко Е.И., Творошенко И.С. (2003) Процессы принятия решений в сложных системах на основе нечетких интервальных представлений. *Вісник Національного технічного університету «ХПІ». Тематичний випуск: Системний аналіз, управління та інформаційні технології*. Х.: НТУ «ХПІ», 1(7), С. 79-86.
16. Gorokhovatskyi V., Rusakova N., and Tvoroshenko I. (2020) The application of image analysis methods and predicate logic in applied problems of magnetic monitoring, *Telecommunications and Radio Engineering*, 79(20), pp. 1801-1811.
17. Гороховатський В.О., Творошенко І.С. (2021) Методи інтелектуального аналізу та оброблення даних: навч. посібник. Харків: ХНУРЕ, 92 с.
18. Творошенко І.С. (2021) Технології прийняття рішень в інформаційних системах: навч. посібник. Харків: ХНУРЕ, 120 с.
19. Кобилін О.А., Творошенко І.С. (2021) Методи цифрової обробки зображень: навч. посібник. Харків: ХНУРЕ, 124 с.
20. Flah P. (2015) *Machine learning. The science and art of building algorithms that extract knowledge from data*, Moscow, Russia: DMK Press, 400 p., (in Russian).
21. Nong Ye. (2013) *Data Mining: Theories, Algorithms, and Examples*, Florida, USA: CRC Press, 349 p.