Сучасні представлення характеристик функціональних вимог практично повністю не мають формалізованих описів. Тому використання існуючих методів аналізу вимог стикається з проблемами формального підтвердження чи спростування наявності у функціональної вимоги відповідної характеристики. Особливо сильно дані проблеми виявляються під час аналізу функціональних вимог на непротиріччя.

Для усунення цих проблем поставлено задачу розробки формалізованих описів ситуацій повної та часткової невідповідності знання-орієнтованих описів функціональних вимог та вдосконалення на основі отриманих описів існуючого методу аналізу функціональних вимог інформаційної системи, яка створюється чи модифікується, на непротиріччя. В якості формального опису функціональної вимоги до інформаційної системи, що аналізується, використаний знання-орієнтований опис, який базується на фреймах. Під час вирішення цієї задачі були розроблені формалізовані описи ситуацій повного протиріччя та часткової невідповідності описів фреймів функціональних вимог, які аналізуються. Показано, що ситуація повного протиріччя є частковим випадком ситуації часткової невідповідності описів фреймів, які аналізуються. Даний результат було покладено до основи вдосконаленого методу аналізу окремих фреймів описів функціональних вимог на непротиріччя. Вдосконалений метод дозволяє не тільки виявляти ситуації повного протиріччя та часткової невідповідності фреймів, які аналізуються, але й кількісно оцінювати ступінь виявленої невідповідності.

Апробація початкового та вдосконаленого методів проводилася під час аналізу функціональних вимог задачі планування індивідуальної діяльності науково-педагогічних працівників закладу вищої освіти. Було розглянуто три типові ситуації, які можуть виникнути під час аналізу функціональних вимог на непротиріччя. Показано, що вдосконалений метод дозволяє знайти більшу кількість помилок в описах функціональних вимог

Ключові слова: функціональна вимога, знання-орієнтований опис, аналіз вимог, непротиріччя, фрейм, інформаційна система

UDC 658.11.05.06
DOI: 10.15587/1729-4061.2020.205518

IMPROVING A METHOD TO ANALYZE THE REQUIREMENTS FOR AN INFORMATION SYSTEM FOR CONSISTENCY

N. Vasiltcova
PhD, Associate Professor*
E-mail: nataliia.vasyltsova@nure.ua

I. Panforova
PhD, Associate Professor*
E-mail: iryna.panforova@nure.ua

O. Neumyvakina
PhD, Lead Engineer
Department of Educational and Research**
E-mail: olga.neumyvakina@nure.ua

*Department of Information Control System**
**Kharkiv National University of Radio Electronics
Nauky ave., 14, Kharkiv, Ukraine, 61166

Received date 10.04.2020
Accepted date 10.06.2020
Published date 30.06.2020

1. Introduction

One of the main problems of modern IT-project management is to reduce the non-productive cost of creating an IT-product. The most effective solutions to this problem are those that are made in the course of initiation and planning of IT-projects [1].

The main description of an IT-product at these IT-project stages is the description of the requirements for this product. At the same time, any IT-product is presented as a system – a combination of interacting elements organized to achieve one or more set goals [2]. This representation makes it possible to divide the requirements for an IT-product into two groups:

a) functional requirements that answer the question: "What exactly should a system do?";

b) non-functional requirements that answer the question: "How exactly should a system meet a specific functional requirement?"

Therefore, it will be possible to reduce the cost of creating an IT-product if one presents this IT-product as a system of consistent functional requirements. To do this, it is necessary to complete a certain list of works, which in article [2] are consolidated into two main processes: the process of determining the needs and requirements of the stakeholder and the process of determining systemic requirements. Both processes share one common feature: first, the works on a collection of separate requirements are done and then the works on the analysis of a whole set of requirements are carried out. At the same time, the main characteristics of such analysis in [2] imply that the requirements are necessary, freely implemented, consistent, have content, complete, reflect specificity, can be performed, traceable, verifiable, and have boundaries.

However, the use of the considered characteristics of analyzed requirements makes it very difficult to automate the works on the analysis of requirements. The fact is that most of these characteristics do not have any formal description. Consequently, the existence of these characteristics in analyzed requirements cannot be proved or refuted by the standard methods of data analysis, structured or unstructured texts. In particular, this difficulty is associated with such a characteristic of requirements for an IT product as consistency. At the same time, analysis of a complete set of functional requirements for consistency is very important in the creation of such IT-products as information systems (IS). Following the definition of "a system", given in [2], it is possible to represent an IS as a combination of interacting functions, and this interaction is carried out by transmitting and receiving structured data arrays. Identification of contradictory functional requirements for the IS at the early stages of an IT project will allow reducing the cost to implement an IT project to create that IS. Such costs include, first and foremost, the time spent to detect contradictory functions of the IS, elimination of detected contradictions, and redesigning the functional structure, database, and IS software. That is why research into the area of automation of works on the analysis of functional requirements for the IS and, in particular, consistency analysis, should be considered relevant.

2. Literature review and problem statement

The basic methods for the analysis of requirements were formulated in the late 20th and early 21st centuries. Examples of these methods are available in [3]. However, the effectiveness of these methods in practice is insufficient. As shown in [4], errors in determining the requirements for an IS have a direct impact on the effectiveness of the representation of these IS. It is indicated in [4] that the main reason for these errors is the gap between perceptions of the requirement generated by a customer and an analyst. Existing methods are focused on analyzing descriptions of requirements formulated from the point of view of an analyst only, and cannot detect such errors. It should be noted that the organization of close interaction between a customer and an analyst in the course of analysis of requirements in accordance with the provisions of modern Agile-methods does not make it possible to improve the effectiveness of this analysis. As shown in paper [5], nowadays, the application of the Agile-methods of IT-project management can have a certain positive effect only in the course of works on the detection of functional requirements. However, during the planning and execution of other works, related to the requirements, the effect of the application of Agile-methods is not proved [5].

That is why modern research into the models and methods for the analysis of requirements for an IS, including functional requirements, is being carried out in two main directions. One of these areas involves research and creation of the methods for requirements analysis based on the interaction of IT-project participants. Thus, the requirement analysis method that helps to bridge the communication gap between a customer and a developer is offered in paper [4]. The method for analyzing requirements for software development, proposed in [6], is based on the joint participation of representatives of IT-project stakeholders. However, the results presented in [7] show that such methods can be improved by structuring the perception of the system created by the participants of an IT-project as a formal conceptual mental model. This model is a visual embodiment of the beliefs and expectations of a user of the created system. It should be admitted that the advantage of this model, as shown in [7], is the improvement as a result of its application of correctness, completeness, and predictability of scenarios of meeting these requirements. However, this model is also an additional source of possible errors that arise from the gap between the perceptions of the requirement formed by a customer and an analyst. In addition, the creation of such a model requires additional time consumption, which leads to a highly undesirable increase in the duration of an IT-project.

Another way of the formal description of the behavior of users is the application of the mathematical apparatus of the category theory, which is the basis of the special declarative language proposed in article [8]. The possibility of automatic semantic analysis of the scenarios of meeting requirements should be considered the advantage of this approach. However, this approach is not without serious shortcomings. The main drawback is the orientation of the approach to the formal description of behavior, considered in [8], to solve the problem of analysis of only the publications of separate requirements for a particular IT-project. This approach does not enable the future application of accumulated experience in subsequent IT-projects for a similar purpose.

The use for requirement analysis of the methods of modern cybernetics, based on the presentation of the process of setting requirements for data as a feedback management system, is considered in paper [9]. The main advantage of this approach is the ability to automate the works on the collection of requirements for the operated IS, provided that the descriptions of such IS services and their behavior are strongly formalized. However, the application of this approach in the practice of operational management is associated with a whole range of problems. In particular, there is an unclear set of variables that should accurately describe:

a) operated serviced of the IS;

b) requests for a change in operated IS services;

c) requests for new versions of the IS services, which are formulated based on requests for a change in operated IS services;

d) problems of IS operation management in general.

Another area of research involves the development of methods for identifying and analyzing requirements based on formal descriptions of these requirements. Thus, it is suggested in paper [10] to analyze software behavior based on its probabilistic descriptions. The advantage of this method is the possibility to make a quantitative analysis of requirements based on probabilistic modeling of mobile software behavior. However, this and similar methods require the accumulation of quite a large data array for statistical studies of possible hypotheses of patterns. At the same time, the creation of new, modernization, or developing existing ISs inevitably leads to new requirements arising from the development of business processes, automated enterprises, or organizations. Such new requirements may not always be described based on previously known patterns or hypotheses about the IS software behavior. That is why this version of the IS requirement analysis has not become quite common yet.

The most common way of the formal description of the IS requirements is to recognize knowledge-oriented descriptions. In this case, much attention is paid to improving the existing models and requirement analysis methods [11]. In particular, paper [12] considered the variant of comparative analysis of the elements of a system based on their descriptions in the form of the Use Case, and Activity diagrams, and data flow diagrams. The transformation of publications of stakeholders' requirements into an executable model of the system based on the Activity and State UML diagrams was considered in [13]. The procedural model for the specification of functional requirements for software is proposed in article [14]. Paper [15] deals with the model of innovative service design, the use of which in the course of the work with requirements involves a combination of formal knowledge-oriented methods and informal methods of the theory of inventive problem-solving. In general, the models and methods discussed in papers [12–15] are particular cases of the same approach. This approach involves separation and subsequent analysis of knowledge-oriented descriptions of requirements for an IS, based on the publication of these requirements in the form of existing visual models. The main advantage of the analysis methods based on this approach is the ability to quantify the characteristics of analyzed requirements. This assessment makes it possible to improve the objectivity of the values of requirement characteristics formed by an analyst. Another, equally important advantage is the use of the knowledge-oriented models as formal descriptions of the analyzed requirements. Such models make it possible to ensure multiple uses of descriptions of separate requirements, including those in various IT-projects.

However, this approach, and, therefore, the models and methods based on it and considered in [12–15] have a series of shortcomings. These shortcomings greatly limit the use of these methods. These shortcomings, first of all, include:

a) almost complete lack of research into formal descriptions of the characteristics of analyzed requirements;

b) linking of formal descriptions of requirements and methods for their analysis to specific visual representations of these requirements, which limits the scope of use of these methods of analysis.

Based on the analysis of the features, merits, and shortcomings of the considered research into the methods for requirements analysis, the following findings can be presented:

a) majority of the methods for the analysis of functional requirements, regardless of the direction of research, are based on some formalized descriptions;

b) one of the main approaches to the formalization of a description of requirements for the IS, stakeholders as the sources of these requirements, as well as the behavior of IT-project participants in the course of working with requirements, is knowledge-oriented models;

c) knowledge-oriented descriptions of requirements are most often implemented in the form of visual diagrams or mathematical models;

d) the use of knowledge-oriented descriptions to analyze requirements for the IS and their separate characteristics are still aimed at the improvement of the known methods for analysis.

These findings determine the need for research to improve existing methods for analyzing functional requirements for the IS in terms of their meeting the desired characteristics. In this case, it is necessary to take into consideration that a set of formulated requirements for an IS is the main source of information for the works on synthesis and description of the architecture of the created system [2]. That is why the most promising studies in this area are those that focus on the improvement of the methods for the analysis of functional requirements for an IS regarding its consistency. Such methods should make it possible to identify the cases of both complete and partial contradiction in the descriptions of separate requirements at the minimal participation in the analysis process of the IT-company employee who performs this analysis.

3. The aim and objectives of the study

The aim of this study is to improve a method to analyze functional requirements for a created or modified IS for consistency. As a result of the proposed improvement, the method should identify the situations of a complete and partial contradiction of descriptions of separate functional requirements for an IS. This makes it possible to formalize the performance of works on the analysis of requirements and to implement the proposed methods in the form of the elements of technology of requirements formation and analysis.

To achieve the aim, the following tasks should be solved:

– to develop formalized descriptions of contradictory and non-compliant formulated functional requirements for the IS;

– to improve a method to analyze separate frames of descriptions of functional requirements for an IS for consistency in order to identify conflicting and inconsistent functional requirements for an IS;

– to check the opportunity of the implementation of assessment of a degree of inconsistency of the frames of functional requirements to detect the situations of complete contradiction and inconsistency.

4. Results of the development of the formalized descriptions of contradictory and inconsistent functional requirements

In paper [16], it was proposed to perform a consistency analysis of functional requirements for the IS based on the knowledge-oriented formal descriptions of such requirements. This description in the general case takes the following form [17]:

$$K_i^f = \left\{ D_{fr}^i, D_{if}^i, D_{fr\_rel}^i \right\}, \tag{1}$$

where K_i^f is the designation of the knowledge-oriented description of the i-th formulated functional requirement, $K_i^f \in K^f$; K^f is the set of functional requirements; D_{fr}^i is the set of the frames describing the i-th functional requirement, $\left\langle d_n^{im}, \left\langle d_{el\_fr}^{imn}, d_{el\_fr\_t}^{imn} \right\rangle \right\rangle \in D_{fr}^i$, m=1, 2, 3, ..., u, n=1, 2, 3, ..., x; u are the number of frames in the description of the i-th requirement; x is the number of elements in the description of the m-th frame of the i-th requirement; d_n^{im} is the description of the name of the m-th frame of the i-th requirement; $d_{el\_fr}^{imn}$ is the description of the n-th element of the m-th frame of the i-th requirement; $d_{el\_fr\_t}^{imn}$ is the description of the type of the n-th element of the m-th frame of the i-th requirement; D_{if}^i is the set of interfaces describing the i-th functional requirement; $\left\langle d_g^{im}, \left\langle d_{el\_if}^{imn}, d_{el\_if\_t}^{imn} \right\rangle \right\rangle \in D_{if}^i$, m=1, 2, 3, ..., v, n=1, 2, 3, ..., y; v are the number of interfaces in the description of the i-th requirement; y is the number of elements in the description of the m-th interface of the i-th requirement; d_g^{im} is the description of the name of the m-th interface of the i-th requirement; $d_{el\_if}^{imn}$ is the description of the name of the n-th element of the m-th interface of the i-th requirement; $d_{el\_if\_t}^{imn}$ is the description of the type of the n-th element of the m-th interface of the i-th requirement; $D_{fr\_rel}^i$ is the set of relations describing the i-th functional requirement, $\left\langle d_{fr\_rel}^{im}, \left\langle d_{el\_fr\_rel}^{imn}, d_{el\_fr\_rel\_t}^{imn} \right\rangle \right\rangle \in D_{fr\_rel}^i$, m=1, 2, 3, ..., w, n=1, 2, 3, ..., z; w are the number of relations in the description of the i-th requirement; z is the number of elements in the description of the m-th relation of the i-th requirement; $d_{fr\_rel\_n}^{im}$ is the description of the name of the m-th connection between the interfaces and/or frames of the i-th requirement; $d_{el\_fr\_rel}^{imn}$ is the description of the names of the n-th element of the m-th relation of the i-th requirement; $d_{el\_fr\_rel\_t}^{imn}$ is the description of the type of the n-th element of the m-th relation of the i-th requirement.

Model (1) makes it possible to consider the representation of the functional requirement for the IS element as a fragment of the network of frames and interfaces of these frames. Then two or more functional requirements, for which at least one of the following situations is satisfied, will be considered contradictory:

a) frames or interfaces with the same or similar name contain two non-intersecting sets of elements;

b) there are different relations between two or more pairs of frames or between a frame and an interface with the same or similar descriptions.

However, this interpretation of contradiction is too narrow. Thus, it does not take into consideration the following possibility: frames or interfaces with the same or similar name contain sets of elements that intersect each other in a minimum number of elements.

Therefore, instead of the concept of "contradiction", we will introduce a broader concept of "inconsistency" as a comparative characteristic of the elements of descriptions of the requirements for an IS. Inconsistency occurs between the elements of descriptions of two or more frames or interfaces with the same or similar names. At the same time, sets of elements of descriptions of inconsistent frames or interfaces as a result of intersection form a new set, the number of elements of which is below the permissible limit. This situation can be represented as follows:

$$\left|\left\langle d_{el\_fr}^{ian}, d_{el\_fr\_t}^{ian}\right\rangle \cap \left\langle d_{el\_fr}^{jbn}, d_{el\_fr\_t}^{jbn}\right\rangle\right| \le \min\left(fr^{ia}, fr^{jb}\right), \quad (2)$$

when the condition is met

$$\left(d_n^{ia} \subseteq d_n^{jb}\right) \vee \left(d_n^{ia} \supseteq d_n^{jb}\right). \quad (3)$$

Here, the magnitude $\min\left(fr^{ia}, fr^{jb}\right)$ designates minimally permissible limit of coincidence of descriptions of two frames fr^{ia} and fr^{jb}, $fr^{ia} \in K_i^f$, $fr^{jb} \in K_j^f$.

Then it is suggested that those frames or interfaces should be termed contradictory, the names of which coincide completely, and the result of the intersection of the sets of their elements will be an empty set. The model of a full contradiction of frames fr^{ia} and fr^{jb} in analog to model (2) and condition (3) can be represented as follows:

$$\left|\left\langle d_{el\_fr}^{ian}, d_{el\_fr\_t}^{ian}\right\rangle \cap \left\langle d_{el\_fr}^{jbn}, d_{el\_fr\_t}^{jbn}\right\rangle\right| = 0, \quad (4)$$

when the condition is met

$$d_n^{ia} = d_n^{jb}. \quad (5)$$

Expressions (4) and (5) are the particular cases of expressions (2) and (3). This makes it possible to improve the method for consistency analysis of separate frames of knowledge-oriented descriptions of functional requirements for the IS K_i^f, proposed in [16].

5. Results of improvement of the method to analyze separate frames of the knowledge-oriented descriptions of the functional requirements for consistency

The method for consistency analysis of separate frames of knowledge-oriented descriptions of functional requirements for the IS K_i^f, proposed in [16], involved the implementation of the following stages:

Stage 1. Choose frame $fr^{ia} \in K_i^f$, $K_i^f \in K^f$, which was not considered before.

Stage 2. Choose $fr^{jb} \in K_j^f$, $K_j^f \in K^f$, which was not considered before.

Stage 3. If the condition is satisfied

$$\left(d_n^{ia} = d_n^{jb}\right) \wedge \left(\left(\left\{\left\langle \begin{matrix} d_{el\_fr}^{ian}, \\ d_{el\_fr\_t}^{ian} \end{matrix} \right\rangle\right\} \setminus \left\{\left\langle \begin{matrix} d_{el\_fr}^{jbn}, \\ d_{el\_fr\_t}^{jbn} \end{matrix} \right\rangle\right\}\right) \ne \varnothing\right), \quad (6)$$

admit the existence of a contradiction between the *i*-th and the *j*-th functional requirements in the descriptions of frames fr^{ia} and fr^{jb}, then proceed to Stage 5.

Stage 4. If the condition is satisfied

$$\left(\left(d_n^{ia} \subseteq d_n^{jb}\right) \vee \left(d_n^{ia} \supseteq d_n^{jb}\right)\right) \wedge$$
$$\wedge\left(\left(\left\{\left\langle d_{el\_fr}^{ian}, d_{el\_fr\_t}^{ian}\right\rangle\right\} \cap \left\{\left\langle d_{el\_fr}^{jbn}, d_{el\_fr\_t}^{jbn}\right\rangle\right\}\right) = \varnothing\right), \quad (7)$$

admit the existence of a contradiction between the *i*-th and the *j*-th functional requirements in the descriptions of frames fr^{ia} and fr^{jb}.

Stage 5. Exclude frame fr^{jb} from further consideration. If not all frames $fr^{jb} \in K_j^f$, were considered, proceed to Stage 2.

Stage 6. Exclude the representation K_j^f from further consideration. If not all representations of the set $\left\{K_j^f\right\}$, were considered, choose representation $K_j^f \in \left\{K_j^f\right\}$, which was not considered before, and proceed to Stage 2.

Stage 7. Exclude frame fr^{ia} from further consideration. If not all frames $fr^{ia} \in K_i^f$, were considered, proceed to Stage 1, otherwise, complete the method application.

This method makes it possible to establish the fact that the descriptions of two different frames do not coincide. However, it does not make it possible to quantify the inconsistency degree of such descriptions. That is why it is proposed to improve this method by applying the proposed formalized descriptions of situations of inconsistency and contradiction of descriptions of separate frames.

The improved method for analysis of separate frames of knowledge-oriented descriptions of the functional requirements for the IS K_i^f for consistency has the following stages.

Stage 1. Select frame $fr^{ia} \in K_i^f, K_i^f \in K^f$, which was not considered before.

Stage 2. Select $fr^{jb} \in K_j^f$, $K_j^f \in K^f$, which was not considered before.

Stage 3. Estimate the degree of inconsistency of frames fr^{ia} and fr^{jb} by performing the following steps.

Step 3. 1. Check if condition (3) is satisfied. If the condition is not satisfied, proceed to Stage 4.

Step 3. 2. Calculate capacities of sets of elements of frames fr^{ia} and fr^{jb}.

Step 3. 3. Select the minimal value from the results of the performance of Step 3.2 and assign it to a variable $\min(fr^{ia}, fr^{jb})$.

Step 3. 4. Form the array of results of the *res* check, the number of elements in which is $\left\lceil \frac{\min\left(fr^{ia}, fr^{jb}\right)}{2} \right\rceil + 1.$

Step 3. 5. Determine the value of the variable k=0.

Step 3. 6. Check if the inequality is satisfied.

$$\left|\left\langle d_{el\_fr}^{ian}, d_{el\_fr\_t}^{ian}\right\rangle \cap \left\langle d_{el\_fr}^{jbn}, d_{el\_fr\_t}^{jbn}\right\rangle\right| \le k. \quad (8)$$

If inequality (8) is satisfied, determine the value of the element *res*(*k*)=1. Otherwise, determine the value of the element *res*(*k*)=0.

Step 3. 7. Increase the value of the variable k by unity. If $k \le \left\lceil \frac{\min\left(fr^{ia}, fr^{jb}\right)}{2} \right\rceil$, proceed to Step 3. 6. Otherwise, complete Stage 3.

Stage 4. Exclude frame fr^{jb} from further consideration. If not all frames $fr^{jb} \in K_j^f$, were considered, proceed to Stage 2.

Stage 5. Exclude representation K_j^f from further consideration. If not all the representations of the set $\left\{K_j^f\right\}$, were considered, select representation $K_j^f \in \left\{K_j^f\right\}$, that was not considered before, and proceed to Stage 2.

Stage 6. Exclude frame fr^{ia} from further consideration If not all frames $fr^{ia} \in K_i^f$, were considered, proceed to Stage 1.

Stage 7. For the identified cases of the inconsistency of frames fr^{ia} and fr^{jb}, publish the values of arrays *res* and finish the method application.

Expression $\left\lceil \frac{\min\left(fr^{ia}, fr^{jb}\right)}{2} \right\rceil$ means rounding off the number $\frac{\min\left(fr^{ia}, fr^{jb}\right)}{2}$ to the nearest integer towards the large side (the limit of number $\frac{\min\left(fr^{ia}, fr^{jb}\right)}{2}$ (by Kenneth Anderson).

The application of the improved method will make it possible for each case of full or partial coincidence of the names of frames fr^{ia} and fr^{jb} to obtain the tables with the estimates of the degree of non-conformity of the descriptions of elements of these frames.

6. Checking the ability to implement an estimation of the degree of inconsistency of frames of functional requirements for detection of situations of complete contradiction and inconsistency

Experimental testing of the improved method was carried out during the analysis for consistency of functional requirements that were put forward to the problem "Formation and keeping the individual plan of a scientific and teaching employee of the department". The purpose of the automated solution of this problem is to reduce the time spent on drawing up a rational version of the individual plan of works of a scientific and teaching employee of a university department for the upcoming academic year. This problem is solved before the beginning of the academic year after the scientific and teaching staff of the department receive a staffing plan and individual educational load. In addition, the solution of this problem can be carried out in the event of a change in a staffing plan, individual educational load, or a list of works of a department that need to be done.

In the course of collecting the functional requirements for the problem "Formation and keeping the individual plan of a scientific and teaching employee of the department", the model of publication of a functional requirement as a process that converts input data flows into output data flows. This model was described in paper [18]. The names of these processes, input and output data flows are given in Table 1.

In accordance with the requirements of the RD 50-34.698-90 standard [19] for drawing up the document "Description of problem setting", each flow of output and input data was presented as a separate frame. The elements of these frames are separate attributes, described at the conceptual level. This representation of the functional requirements makes it possible to apply the proposed improved method for analysis of functional requirements for consistency, formulated in Table 1.

Consider the use of the improved method for the following cases:

a) the names of the comparable frames are exactly the same – the input data flow frame 4.1 and the input data flow frame 6.2, as well as the output data flow frame 6.1 and the input data flow frame 10.5 from Table 1;

b) the name of one of the comparable frames is a subset of the name of another of the comparable frames – the output data flow frame 7.1 and the input data flow frame 9.2 from Table 1.

Descriptions of the content of these frames, performed by an analyst during the collection and primary publication of functional requirements, are given in Table 2.

The focus is on the progress of Stage 3 and Stage 7 of the improved method. First, let us consider the execution of these stages to test the consistency of input data flow frame 4. 1 and input data flow frame 6.2.

During the implementation of Stage 3.1, to formalize the name of the frames executed in the domain language, here and thereafter it is proposed to use the combination of the text pre-processing techniques – stemming and stop-word removal. Porter's stemmer was used for stemming in this case. Prepositions were considered as stop-words. The result of processing the names of input data flow frame 4.1 and input data flow frame 6.2. is given in Table 3.

The obtained results show that condition (3) for these frames is met – the names of the frames are completely identical.

As a result of the implementation of Steps 3.2–3.4, the capacity of the set of elements of input data flow frame 4.1 in Table 2 is accepted equal to 7. The capacity of a set of elements of input data flow frame 6.2 is accepted equal to 7. Therefore, for the selected frames, the value of variable $\min\left(fr^{ia}, fr^{jb}\right)=7$. The length of the array of the check results *res* will be 5.

Results of implementation of Steps 3.5–3.7 for the values of variable *k* from 0 to 4 are given in Table 4.

The values of the elements of the results check array, given in Table 4, make it possible to conclude during the implementation of Stage 7: input data flow frame 4.1 and input data flow frame 6.2 do not contradict each other, and the descriptions of their elements correspond to each other more than by 50 %.

Next, let us consider the implementation of Stage 3 and Stage 7 of the improved method for testing the consistency of output data flow frame 6.1 and input data flow frame 10.5.

The result of processing the names of output data flow frame 6.1 and input data flow frame 10.5 are given in Table 5.

The obtained results show that condition (3) for these frames is fulfilled – the names of the frames coincide completely.

As a result of completion Steps 3.2–3.4, the capacity of the set of elements of output data flow frame 6.1 according to Table 2 is equal to 10. The capacity of the set of elements of input data flow frame 10.5 is accepted equal to 10. Therefore, for the selected frames, the value of variable $\min\left(fr^{ia}, fr^{jb}\right)=10$. The length of the array of checking results *res* will be 6.

Table 1

Description of functions, input and output data flows of the functional problem "Individual plan of a scientific and teaching employee of a department"

Description of the function		Description of the input data flow		Description of the output data flow	
No. of entry	Title	No. of entry	Title	No. of entry	Title
1	Conversion of section «Educational work»	1.1	Teacher's educational load for the academic year	1.1	Information from the individual plan section «Educational work»
2	Formation and keeping regulatory-reference information on the key KPIs	2.1	Information about the key KPI of the department	2.1	Information about the key KPI of the department
3	Formation of a list of types of work recommended being done	3.1	Types of work recommended being done	3.1	Types of work recommended being done
4	Formation of the section «Scientific work»	4.1	Information about a teacher	4.1	Information from the individual plan section «Scientific work»
		4.2	Information about the kinds of works that are planned to be done		
		4.3	Hours left		
		4.4	Kinds of work recommended to be done		
		4.5	Information from the section of the individual plan «Scientific work»		
5	Formation of a list of positions and long-term responsibilities	5.1	Information about positions long-term responsibilities	5.1	Information from the section of the individual plan «List of positions and long-term responsibilities
		5.2	Information from the section of the individual plan «List of positions and long-term responsibilities»		
6	Formation of section «Organizational and educational work»	6.1	Hours left	6.1	Information from section of the individual plan «Organizational and educational work»
		6.2	Information about a teacher		
		6.3	Information about the kinds of work that are planned to be done		
		6.4	Kinds of work recommended to be done		
		6.5	Information from section of the individual plan «Organizational and educational work»		
7	Formation of the section «Methodical work»	7.1	Hours left	7.1	Information from a section of the individual plan «Methodical work»
		7.2	Information about a teacher		
		7.3	Information about the kinds of work planned to be done		
		7.4	Kind of works recommended to be done		
		7.5	Information from a section of the individual plan «Methodical work»		
8	Formation of the teacher's KPI and a part of the department's KPI	8.1	Information from the section of the individual plan «Scientific work»	8.1	Information about the teacher's KPI and a part of the department's KPI
9	Formation of a summary table for the academic year	9.1	Information about the number of hours from the section of the individual plan «Educational work»	9.1	Information about the number of hours for the plan sections
		9.2	Information about the number of hours from section of the individual plan «Methodical work»		
		9.3	Information about the number of hours from the section of the individual plan «Scientific work»		
		9.4	Information about the number of hours from section «Organizational and educational work»		
		9.5	Information about a teacher		
10	Formation of output document «Individual plan»	10.1	Information about a teacher	10.1	Individual plan
		10.2	Information from the section of the individual plan «Educational work»		
		10.3	Information from the section of the individual plan «Methodical work»		
		10.4	Information from the section of the individual plan «Scientific work»		
		10.5	Information from the section of the individual plan «Organizational and educational work»		

Table 2

Descriptions of the content of the analyzed frames

Frame description		Description of frame elements			
No. of entry	Title	No. of entry	Title	Type of information	Length
4.1	Information about a teacher	4.1.1	Employee identifier	Numeric	4
		4.1.2	Surname	Symbolic	30
		4.1.3	Name	Symbolic	15
		4.1.4	Patronymic	Symbolic	30
		4.1.5	Department	Symbolic	7
		4.1.6	Current position	Symbolic	60
		4.1.7	Share of load	Symbolic	4
6.2	Information about a teacher	6.2.1	Employee identifier	Numeric	4
		6.2.2	Surname	Symbolic	30
		6.2.3	Name	Symbolic	15
		6.2.4	Patronymic	Symbolic	30
		6.2.5	Department	Symbolic	7
		6.2.6	Current position	Symbolic	60
		6.2.7	Share of load	Symbolic	4
6.1	Information from section of the individual plan «Organizational and educational work»	6.1.1	Academic year	Date	9
		6.1.2	No. by order	Numeric	2
		6.1.3	Content	Symbolic	255
		6.1.4	Number of hours	Numeric	3
		6.1.5	Final result	Symbolic	30
		6.1.6	Deadline	Symbolic	15
		6.1.7	Mark about completion	Symbolic	4
		6.1.8	Total hours	Numeric	3
		6.1.9	Teacher	Symbolic	25
		6.1.10	Head of department	Symbolic	25
10.5	Information from section of the individual plan «Organizational and educational work»	10.5.1	Academic year	Date	9
		10.5.2	No. by order	Numeric	2
		10.5.3	Content	Symbolic	255
		10.5.4	Number of hours	Numeric	3
		10.5.5	Final result	Symbolic	30
		10.5.6	Deadline	Symbolic	15
		10.5.7	Mark about completion	Symbolic	4
		10.5.8	Total hours	Numeric	3
		10.5.9	Teacher	Symbolic	25
		10.5.10	Head of the department	Symbolic	25
7.1	Information from section of the individual plan «Methodical work»	7.1.1	Academic year	Date	9
		7.1.2	No. by order	Numeric	2
		7.1.3	Content	Symbolic	255
		7.1.4	Number of hours	Numeric	3
		7.1.5	Final result	Symbolic	30
		7.1.6	Deadline	Symbolic	15
		7.1.7	Mark about completion	Symbolic	4
		7.1.8	Total hours	Numeric	3
		7.1.9	Teacher	Symbolic	25
		7.1.10	Head of department	Symbolic	25
9.2	Information about the number of hours from the section of the individual plan «Methodical work»	9.2.1	Total hours	Numeric	3

Table 3

Results of pre-processing of the names of analyzed frames for input data flow frame 4.1 and input data flow frame 6.2

Stage of pre-processing	Input data flow frame 4.1	Input data flow frame 6.2
Initial frame name	Information about a teacher	Information about a teacher
Results of Porter's stemmer application	Informat about a teacher	Informat about a teacher
Results of stop-words deletion	Informat teacher	Informat teacher

The results of the implementation of Steps 3.5–3.7 for the values of variable *k* from 0 to 5 are given in Table 6.

Table 4

Results of detection of situations of contradiction and inconsistencies for input data flow frame 4.1 and input data flow frame 6.2

Value of variable *k*	The capacity of the intersection of sets of frame elements	Value of elements of an array of checking results *res (k)*
0	7	0
1	7	0
2	7	0
3	7	0
4	7	0

Table 5

Results from pre-processing of the names of analyzed frames for output data flow frame 6.1 and input data flow frame 10.5

Pre-processing stage	Output data flow frame 6.1	Input data flow frame 10.4
Initial name of the frame	Information from the section of the individual plan «Organizational and educational work»	Information from the section of the individual plan «Organizational and educational work»
Results of application of Porter's stemmer	Informat from the section of the individu plan «Organiz and educ work»	Informat from the section of the individu plan «Organiz and educ work»
Results of deleting stop-words	Informat section individu plan «Organiz educ work»	Informat section individu plan «Organiz educ work»

Table 6

Results of identification of contradiction and inconsistency situations for output data flow frame 6.1 and output data flow frame 10.5

Value of variable *k*	The capacity of the intersection of sets of elements of frames	Value of the element of the array of checking results *res*(*k*)
0	10	0
1	10	0
2	10	0
3	10	0
4	10	0
5	10	0

The values of the elements of the array of result checking, given in Table 6, make it possible to conclude during the implementation of Stage 7: output data flow frame 6.1 and input data flow frame 10.5 do not contradict each other, and the descriptions of their elements correspond to each at least by 50 %.

Next, let us consider the implementation of Stage 3 and Stage 7 of the improved method for testing the inconsistency of output data flow frame 7.1 and input data flow frame 9.2.

The result of processing the names of output data flow frame 7.1 and input data flow frame 9.2 are given in Table 7.

The obtained results show that condition (3) for these frames is fulfilled – the names of the frame of output data flow frame 7.1 is the subset of the name of input data flow frame 9.2.

Table 7

Results of pre-processing of the names of analyzed frames for output data flow frame 7.1 and input data flow frame 9.2

Pre-processing stage	Output data flow frame 7.1	Input data flow frame 9.2
Initial name of the frame	Information from the section of the individual plan «Methodical work»	Information about the number of hours from the section of the individual plan «Methodical work»
Results of application of Porter's stemmer	Informat from the section of the individu plan «Method work»	Informat about the number of hour from the section of the individu plan «Method work»
Results of deleting stop-words	Informat section individu plan «Method work»	Informat number hour section individu plan «Method work»

As a result of the completion of Steps 3.2–3.4, the capacity of the set of elements of output data flow frame 7.1 according to Table 2 is accepted equal to 10. The capacity of the set of elements of input data flow rate 9.2 is accepted equal to 1. Therefore, for the selected frames, the value of variable $\min\left(fr^{ia}, fr^{jb}\right)=1.$ The length of the array of checking results *res* will be 2.

The results of the implementation of Steps 3.5–3.7 for the values of variable *k* from 0 to 1 are given in Table 8.

Table 8

Results of identification of situations of contradiction and inconsistency for output data flow frame 7.1 and input data flow frame 9.2

Value of variable *k*	The capacity of the intersection of frame elements sets	Value of the element of the array of checking results *res*(*k*)
0	1	0
1	1	1

The values of the elements of the array of result checking, given in Table 8, make it possible to conclude during the implementation of Stage 7: output data flow frame 7.1 and input data flow frame 9.2 do not correspond to each other completely.

The cause of the detected inconsistency should be considered the minimal attributive description of input data flow frame 9.2. It is only one attribute, which should be recognized as an error of the analyst who performed collection and publication of functional requirements. The refined description of the content of input data flow frame 9.2 will take the form, given in Table 9.

Then, for input data flow frame 9.2 and output data flow frame 7.1, the results of the implementation of Stage 3 of the improved method will take the form given in Table 10.

The values of the elements of the array of result checking, given in Table 10, make it possible to conclude during the implementation of Stage 7: output data flow frame 7.1 and input data flow frame 9.2 correspond to each other at least by 50 %.

Now consider the application of the method for analysis of the same pairs of frames, proposed in [16]. In this case, special attention should be paid to the implementation of Stages 3 and 4 of this method.

Table 9

Refined description of the content of input data flow frame 9.2

Frame description		Description of elements of the frame			
No. of entry	Title	No. of entry	Title	Information type	Length
9.2	Information about the number of hours from the individual plan section «Methodical work»	9.2.1	Academic year	Data	9
		7.1.8	Total hours	Numeric	3
		7.1.9	Teacher	Symbolic	25
		7.1.10	Head of the department	Symbolic	25

Table 10

Results of identification of situations of contradiction and inconsistency for output data flow frame 7.1 and input data flow frame 9.2

Value of variable k	The capacity of the intersection of frame elements sets	Value of the element of the array of checking results $res(k)$
0	4	0
1	4	0
2	4	0

For the input data flow frame 4.1 and input data flow frame 6.2, in the course of implementation of Stage 3 of the method, proposed in [16], it was found that:

a) the names of analyzed frames coincide (Table 3), therefore, the left part of condition (6) is satisfied;

b) the attributes of analyzed frames coincide (Table 2), therefore, the right part of condition (6) is not satisfied.

Therefore, condition (6) for the analyzed frames is not satisfied and frames at this stage are recognized as consistent at this stage.

In the course of implementation of Stage 4 of the method, proposed in [16], it was established for the given frames that:

a) the names of the analyzed frames coincide (Table 3), therefore, the left part of condition (7) is satisfied;

b) the attributes of analyzed frames coincide (Table 2), therefore, the right part of condition (7) is not satisfied.

Therefore, condition (7) for the analyzed frames is not met and the frames at this stage are considered consistent.

By the results of the application of the method proposed in [16], input data flow frame 4.1 and input data flow frame 6.2, were found to be consistent. This is in line with the results of the application of the improved method.

For output data flow frame 6.1 and input data flow frame 10.5, during the implementation of Stage 3 of the method proposed in [16], it was found that:

a) the names of analyzed frames coincide (Table 5), therefore, the left part of condition (6) is satisfied;

b) the attributes of analyzed frames coincide (Table 2), therefore, the right part of condition (6) is not satisfied.

Therefore, condition (6) for analyzed frames is not met and the frames are recognized as consistent at this stage.

In the course of Stage 4 of the method proposed in [16], it was found for these frames that:

a) the names of the analyzed frames coincide (Table 5), therefore, the left part of condition (7) is satisfied;

b) the attributes of analyzed frames coincide (Table 2), therefore, the right part of condition (7) is not satisfied.

Therefore, condition (7) for the analyzed frames is not met and the frames are considered consistent at this stage.

According to the results of the application of the method, proposed in [16], output data flow frame 6.1 and input data flow frame 10.5 method are considered consistent. This is in line with the results of the application of the improved method.

For output data flow frame 7.1 and input data flow frame 9.2, during the implementation of Stage 3 of the method, proposed in the [16], it was found:

a) the names of analyzed frames do not coincide (Table 7), therefore, the left part of condition (6) is not satisfied;

b) 9 attributes of the analyzed output data flow frame 7.1 are not contained in the description of input data flow frame 9.2 (Table 2), therefore, the right part of condition (6) is not satisfied.

Therefore, condition (6) for analyzed frames is not satisfied and the frames at this stage are recognized as consistent.

In the course of implementation of Stage 4 of the method proposed in [16], it was established for these frames:

a) the name of output data flow frame 7.1 is a subset of the name of input data flow frame 9.2 (Table 7), therefore, the left part of condition (7) is satisfied;

b) the result of the intersection of descriptions of sets of attributes of analyzed frames contain one element (Table 2), therefore, the right part of condition (7) is not satisfied

Therefore, condition (7) for analyzed frames is not satisfied and the frames are considered consistent at this stage.

Based on the results of the application of the method, proposed in [16], output data flow frame 7.1 and input data flow frame 9.2, are considered consistent. This result is correct because these frames do not really contradict each other. However, this result is inaccurate because, as shown above, the descriptions of these frames do not fully correspond to each other. This inaccuracy subsequently could lead to errors in the course of development of the database, the SQL-queries, and the applied software of the analyzed problem.

The possibility to detect not only a situation of complete contradiction but also a situation of the partial inconsistency of descriptions of the contents of separate frames is the main advantage of the improved method compared to the method proposed in paper [16]. Another important advantage of the improved method is the quantitative estimate of the degree of inconsistency of descriptions of the contents of separate frames. As shown in Tables 4, 6, and 10, the results of the implementation of steps 3.5–3.7 of the improved method are equal to 0. This means that in compared frames there are the same descriptions of attributes, the number of which is more than 50 % (according to the value of the variable $\left\lceil \frac{\min\left(fr^{ia}, fr^{jb}\right)}{2} \right\rceil$) of the minimum number of attributes descriptions in analyzed frames. An example of detection of the situation, where the minimum coincidence degree is less than 50 %, is given in Table 8. This threshold for the coincidence of attribute descriptions was chosen for practical reasons (frame descriptions may vary partially due to the peculiarities of domains and scenarios of performance of IS functions). At the same time, an insignificant modification of Step 3.4 of the improved method will make it possible to check the facts of the coincidence/mismatch of all descriptions of the attributes of analyzed frames. In this case, it becomes possible to estimate the degree of matching frames from 0 to 100 % due to some increase in the time spent on the implementation of this method. Such estimation allows the subsequent formal solution to the problem of deciding on performing further works to clarify the descriptions of separate functional requirements.

However, these merits also determine the main drawbacks of the improved method. In particular, the identification of the situations of the partial inconsistency of descriptions of the contents of individual frames significantly increases the sensitivity of the improved method to the completeness of the description of separate frames. As shown above, it is advisable to have at least four or five attributes in the description of each separate frame. Such sensitivity requires more time to identify detailed descriptions of functional requirements.

Another drawback of the improved method is that it is necessary to admit that there is a slight increase in the time spent on its implementation compared to the method proposed in paper [16]. However, this drawback is not significant, especially in the case of implementation of this method as an element of information technology for the formation and analysis of functional requirements for the IS.

7. Discussion of the obtained theoretical and practical results

The use of the method proposed in paper [16] allowed the identification of only the situation of the inconsistency of descriptions of separate frames but did not enable quantification of the degree of inconsistency of these descriptions. Therefore, this method was improved by developing formalized descriptions (2) to (5) of the situations of contradiction and inconsistency between two frames from the descriptions of different requirements. Analysis of these descriptions led to the conclusion that the situation of contraction between the descriptions of two frames is a particular case of the situation of the inconsistency of these descriptions. As a result of its improvement, the method makes it possible not only to identify situations of inconsistency or contradiction of descriptions of analyzed frames but also to quantify the degree of identified inconsistency.

The improved method was tested during the analysis of functional requirements of the problem of automation of planning the individual activities of the academic and teaching staff of a higher educational institution. As a result of testing, it was found that the improved method is more accurate than the method for analysis proposed in paper [16]. This became possible due to using condition (8) for analysis of frames, which makes it possible to detect not only a contradiction but also a partial inconsistency between the descriptions of analyzed frames. It should be noted, however, that the method proposed in paper [16] and the improved method are approximately the same in terms of the computational complexity of the algorithms implementing these methods.

The main feature that makes the improved method difficult to use is its sensitivity to the completeness of the description of analyzed frames. The result of this feature is additional works to correct the descriptions of separate functional requirements and, as a result, an increase in the time consumption for initiation and planning of an IT-project for the IS creation. However, such costs are offset by a reduction in the time spent on the implementation of an IT-project of the IS creation due to the fact that the project plan does not contain any additional works to correct the contradictions and inconsistencies between the descriptions of separate functions of this IS.

Another feature of the application of the improved method is the establishment of a minimum compliance degree of not less than 50 % of the total number of attributes of analyzed frames. This magnitude was chosen based on practical considerations about the possible divergence of frames describing the same element of a domain. At the same time, the improved method can be easily adjusted to verify the compliance degrees up to 100 %. However, such correction will lead to some increase in the time spent on the implementation of the algorithm realizing the method. It should be admitted that solving the problem about the appropriate magnitude of the degree of conformity of descriptions of separate frames requires long-term practical tests, including the ones involving the IT-projects that were completed.

8. Conclusions

1. The formalized descriptions of situations of complete contradiction and inconsistency of separate frames of formulated functional requirements for the IS were developed. Based on these descriptions, it was concluded that a situation of the complete contradiction of descriptions of analyzed frames is a particular case of a situation of the inconsistency of these descriptions. This conclusion is the basis for the improvement of the earlier developed method for analyzing the descriptions of functional requirements to the IS consistency.

2. The previously proposed method of analyzing descriptions of functional requirements for the IS consistency was improved. This method enabled the identification of description only of the situation of the contradiction of descriptions of analyzed frames. The essence of the improvement is the transition from two particular conditions for checking the descriptions of analyzed frames for consistency to one common condition of verification of the descriptions of these frames for incompliance with each other. The obtained improved method makes it possible to identify not only situations of complete contradiction, but also those of partial inconsistency of descriptions of analyzed frames.

3. We tested the possibility of implementing the estimation of the degree of inconsistency of frames of functional requirements to detect the situations of complete contradiction and inconsistency during analysis for consistency of separate functional requirements for the problem "Formation and keeping of the individual plan of a scientific and teaching employee of the department". For comparison, the frame pairs, analyzed during testing, were also analyzed using the previously developed method for analysis of the descriptions of functional requirements for the IS consistency. The results of testing suggest that the improved method detects more errors in the descriptions of functional requirements. Thus, the implementation and application of the improved method enable the reduction of the time it takes to implement an IT-project of the IS creation. This reduction occurs due to the exclusion from the project-plan of additional works related to correcting the errors in the information and software elements of the IS, caused by the errors in the descriptions of functional requirements for this IS.

References

1. A Guide to the Project Management Body of Knowledge (PMBOK® Guide) (2013). Newton Square: Project Management Institute, Inc., 586.
2. GOST R 57193-2016. Systems and software engineering. System life cycle processes (ISO/IEC/IEEE 15288:2015, NEQ) (2016). Moscow: Standartinform, 98.
3. Maguire, M., Bevan, N. (2002). User Requirements Analysis. Usability, 133–148. doi: https://doi.org/10.1007/978-0-387-35610-5_9
4. Stowell, F., Cooray, S. (2017). The Appreciative System, Learning And Its Impact Upon Is Design. Communications of the Association for Information Systems, 40, 93–119. doi: https://doi.org/10.17705/1cais.04006
5. Ferreira Martins, H., Carvalho de Oliveira Junior, A., Dias Canedo, E., Dias Kosloski, R. A., Ávila Paldês, R., Costa Oliveira, E. (2019). Design Thinking: Challenges for Software Requirements Elicitation. Information, 10 (12), 371. doi: https://doi.org/10.3390/info10120371
6. Ali, N., Lai, R. (2016). A method of requirements elicitation and analysis for Global Software Development. Journal of Software: Evolution and Process, 29 (4), e1830. doi: https://doi.org/10.1002/smr.1830
7. Beimel, D., Kedmi-Shahar, E. (2018). Improving the identification of functional system requirements when novice analysts create use case diagrams: the benefits of applying conceptual mental models. Requirements Engineering, 24 (4), 483–502. doi: https://doi.org/10.1007/s00766-018-0296-z
8. Asteasuain, F., Braberman, V. (2016). Declaratively building behavior by means of scenario clauses. Requirements Engineering, 22 (2), 239–274. doi: https://doi.org/10.1007/s00766-015-0242-2
9. Liu, L., Zhou, Q., Liu, J., Cao, Z. (2017). Requirements cybernetics: Elicitation based on user behavioral data. Journal of Systems and Software, 124, 187–194. doi: https://doi.org/10.1016/j.jss.2015.12.030
10. Yu, Y.-J., Liu, C. (2017). Little Model in Big Data: An Algebraic Approach to Analysing Abstract Software Behaviours. Ruan Jian Xue Bao/Journal of Software, 28 (6), 1488–1497. doi: http://doi.org/10.13328/j.cnki.jos.005229
11. Ambreen, T., Ikram, N., Usman, M., Niazi, M. (2018). Empirical research in requirements engineering: trends and opportunities. Requirements Engineering, 23 (1), 63–95. doi: https://doi.org/10.1007/s00766-016-0258-2
12. Kaiya, H., Adachi, K., Chubachi, Y. (2019). Requirements Exploration by Comparing and Combining Models of Different Information Systems. Knowledge-Based Software Engineering: 2018, 64–74. doi: https://doi.org/10.1007/978-3-319-97679-2_7
13. Kim, S.-K., Myers, T., Wendland, M.-F., Lindsay, P. A. (2012). Execution of natural language requirements using State Machines synthesised from Behavior Trees. Journal of Systems and Software, 85 (11), 2652–2664. doi: https://doi.org/10.1016/j.jss.2012.06.013
14. Cornejo-Aparicio, V., Flores-Silva, S., Bedregal-Alpaca, N., Tupacyupanqui-Jaen, D. (2020). Procedural model for the specification of functional requirements in software construction. RISTI - Revista Iberica de Sistemas e Tecnologias de Informacao, 2020 (E26), 571–586.
15. Lee, C.-H., Chen, C.-H., Trappey, A. J. C. (2019). A structural service innovation approach for designing smart product service systems: Case study of smart beauty service. Advanced Engineering Informatics, 40, 154–167. doi: https://doi.org/10.1016/j.aei.2019.04.006
16. Ievlanov, M., Vasiltcova, N., Panforova, I. (2018). Development of methods for the analysis of functional requirements to an information system for consistency and illogicality. Eastern-European Journal of Enterprise Technologies, 1 (2 (91)), 4–11. doi: https://doi.org/10.15587/1729-4061.2018.121849
17. Ievlanov, M. (2015). Methods of presenting formulated requirements to the information system at the level of knowledge. Eastern-European Journal of Enterprise Technologies, 4 (3 (76)), 4–11. doi: https://doi.org/10.15587/1729-4061.2015.47535
18. Ievlanov, M., Vasiltsova, N., Panfiorova, I. (2014). Web-service registry model improvements. Visnyk Kremenchutskoho natsionalnoho universytetu imeni Mykhaila Ostrohradskoho, 5, 34–41.
19. RD 50-34.698-90. Avtomatizirovannye sistemy. Trebovaniya k soderzhaniyu dokumentov (2002). Moscow: IPK Izdatel'stvo standartov, 25.
20. Katsov, I. (2019). Mashinnoe obuchenie dlya biznesa i marketinga. Sankt-Peterburg: Piter, 512.