Розглянуто задачу розробки методів аналізу сформульованих функціональних вимог до інформаційної системи. Запропоновані методи дозволяють виявляти протиріччя між описами окремих фреймів, інтерфейсів та зв'язків у представленнях вимог на рівні знань та нелогічність описів функціональних вимог. Використання даних методів дозволяє скоротити час розробки інформаційної системи за рахунок виконання робіт з аналізу вимог паралельно роботам з формування вимог

Ключові слова: аналіз вимог, функціональні вимоги, фрейм, протиріччя, нелогічність, онтологічна точка

Рассмотрена задача разработки методов анализа сформулированных функциональных требований к информационной системе. Предложенные методы позволяют выявлять противоречия между описаниями отдельных фреймов, интерфейсов и связей в представлениях требований на уровне знаний и нелогичность описаний функциональных требований. Использование данных методов позволяет сократить время разработки информационной системы за счет выполнения работ по анализу требований параллельно работам по формированию требований

Ключевые слова: анализ требований, функциональные требования, фрейм, противоречие, нелогичность, онтологическая точка

UDC 044.03; 658.11.05.06
DOI: 10.15587/1729-4061.2018.121849

DEVELOPMENT OF METHODS FOR THE ANALYSIS OF FUNCTIONAL REQUIREMENTS TO AN INFORMATION SYSTEM FOR CONSISTENCY AND ILLOGICALITY

M. Ievlanov
Doctor of Technical Science, Associate Professor*
E-mail: maksym.ievlanov@nure.ua
N. Vasiltcova
PhD, Associate Professor*
E-mail: nataliia.vasyltsova@nure.ua
I. Panforova
PhD, Associate Professor*
E-mail: iryna.panforova@nure.ua
*Department of information control systems
Kharkiv National University of Radio Electronics
Nauky ave., 14, Kharkiv, Ukraine, 61166

1. Introduction

One of the most promising modern trends in the development of IT-sector is the reduction of unproductive expenditures for the initiation, planning, execution and control of IT-projects. Such a reduction is particularly important for IT-projects aimed at creating or developing complex IT-products. The examples of such products include information systems (IS), including control systems of enterprises and organizations.

One of the approaches to reducing the cost of IT-projects that create or develop IS aims to detect and eliminate maximally possible quantity of errors at the early stages of an IT-project [1]. In line with a given approach, possible errors should be identified in the course of analysis of descriptions of IS elements immediately after compiling the stakeholders' requirements to the IS being created or developed. In this case, it is recommended to conduct separate analysis of work directly in the course of development of right holders' requirements [2]. The purpose of these activities is, as a rule, to confirm the adequacy of the stated requirements to the subject area and IS to be created or developed.

However, most of the existing methods and techniques imply a non-automated execution of work when analyzing requirements to IS. This peculiarity makes it much harder to apply methods that analyze requirements to IT-projects aimed at creating and developing large-scale and complex IS. For such projects, there is a constant need to analyze the requirements put forward by a Consumer of IT-services, including changes to be made to the previously stated requirements. That is why IT-projects on the creation and development of IS inevitably face a situation when an analyst is unable to take into account all features of the influence of separate requirements to IS on each other. The result is a high risk of failure to detect errors due to the incorrect definition of certain requirements to IS and the synthesis of IS architecture description without agreeing on the descriptions of separate requirements to IS. The necessity to reduce a given risk allows us to consider the task on developing the requirements to IS, involving automated processing of publications of particular requirements, very important from the theoretical and applied points of view.

2. Literature review and problem statement

At present, among the most commonly used are the methods and techniques for requirement analysis based on the solutions, proposed in the late XX – early XXI century. These solutions are based on the following recommenda-

tions, compiled on the basis of practical experience in the implementation of a large number of successful IT-projects [2]:

a) it is recommended to analyze the entire range of identified requirements;

b) it is recommended that requirements analysis should include the identification of inconsistent, missed, incomplete, ambiguous, illogical, or unverifiable requirements, as well as defining of the priorities in meeting these requirements;

c) in the course of analysis, it is necessary to solve problems arising from the definition of requirements (those problems, related to requirements, which cannot be implemented, or those whose implementation is impractical).

The examples of basic methods and techniques for requirements analysis are described in [3]. However, the methods of requirements analysis, considered in [3], do not make it possible, as experience reveals, to detect and eliminate most errors, associated with the development of requirements to IS. It is now a well-known fact that erroneous definition of requirements to IS exerts a direct influence on the performance efficiency when releasing IS and its upgrades to the Consumer of IT-services [4].

To eliminate such an impact, studies are carried out in various fields. One of these trends implies the development and modification of methods for requirements analysis using maximally easy-to-implement tools. Thus, in [4], it is proposed to reduce the number of errors in requirements by devising a special method for requirements analysis that would help bridge the gap in communication between a customer and a developer. Paper [5] proposes a method for the detection and analysis of requirements to software development, based on joint participation of representatives of all stakeholders of an IT-project.

According to another, more common, trend in the research, the elimination of erroneous definition of requirements to IS should be implemented through the formalization of models and methods of analysis. For example, in [6], the process of establishing the requirements to data is formulated as a feedback control system with continuous optimization of users' behavior models. Such a representation makes it possible to apply the concepts of modern cybernetics in order to describe processes of requirements development and analysis. Paper [7] explores the possibilities to describe and model behavior of users of the system being developed applying the apparatus of the theory of categories, based on which, with the help of graphical methods, a special declarative language was created. Article [8] proposes an algebraic approach to the analysis of probabilistic behavior models for software.

Special attention should be paid to studies that consider the process of requirements analysis as a decision support process regarding the analyzed requirements. An example of using the METRO decision support platform to control the uncertainty of stated requirements to the product being developed was described in [9].

One of the most promising trends of research in the field of requirement engineering is the approach based on the application of ontologies, knowledge-oriented models and methods. The use of ontologies in requirement engineering is reviewed in [10]. It is noted specifically [10]:

a) there is empirical evidence of the advantages when using ontologies in requirements engineering for the purpose of reducing ambiguity, inconsistency and incompleteness of requirements;

b) the process of requirements engineering in most studies is considered only partially;

c) there is no a unified ontology-based style at present for modeling the requirements engineering processes;

d) most of the studies in this area are related only to functional requirements;

e) none of the ontologies of requirements engineering is commonly used at present in the community of experts in this field.

Similar conclusions were drawn in paper [11], which addressed the issues on data mining and knowledge management for developing the requirements specifications.

However, modern studies are not limited by theoretical aspects of the application of ontologies in requirements engineering. Several studies attempt to solve problems on using the knowledge-oriented models during development and application of specialized information technologies for requirements development and analysis. An example of such research is given in paper [12], which investigates application of the ontology-based tool for automated analysis of consistency, correctness, and completeness of requirements to an IT-product (3 Cs problems). Another example is given in article [13], which examines particular aspects in the analysis of requirements to multi-agent systems.

The analysis that we performed allows us to highlight the most promising trend of research in the field of analysis of requirements to IS. We consider such trend to be the development and verification of analysis methods, which are based on the knowledge-oriented models of requirements to IS.

3. The aim and objectives of the study

The aim of present research is to develop methods for the analysis of functional requirements to the created or modified information system. This would make it possible to formalize the work on requirements analysis, and to implement the proposed methods in the form of the elements of technology for requirements development and analysis.

To achieve the set goal, the following tasks must be solved:

– to specify the order of work aimed at analyzing the stated functional requirements to an information system;

– to develop methods for the analysis of functional requirements to IS for consistency;

– to develop methods for detecting duplicated, missed, and illogical functional requirements to IS.

4. Results of determining of the order of work aimed at analyzing the stated functional requirements to an information system

Organization of processes for determining rights holders' requirements and their analysis requires performing an analysis of functional requirements after executing essential work on the development of these requirements, [2]. However, the results of development of the service approach, models for the formal description of requirements at data levels, information and knowledge, as well as the methods for forming the representations of functional requirements, synthesis and selection of the description of rational architecture of the developed IS [14–18], make it possible to perform separate analysis of the requirements in parallel to the main work. Description of such an organization of work is given in the form of IDEF3 model in Fig. 1 [19].

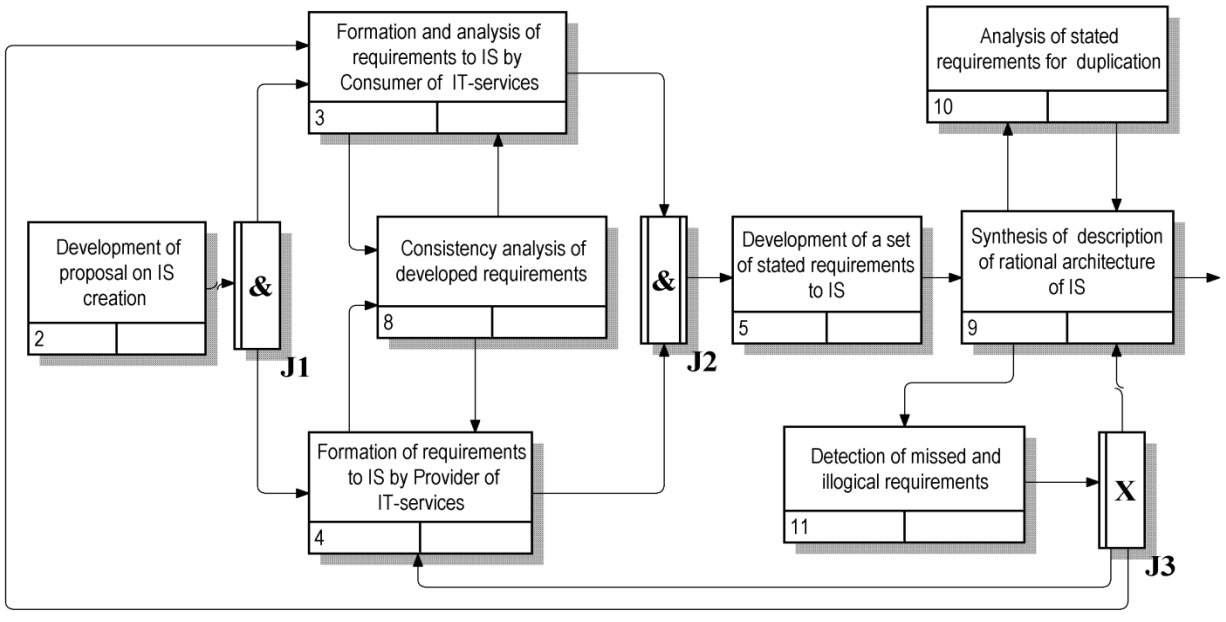

Fig. 1. IDEF3-model describes work on the development and analysis of functional requirements to the information system being developed in accordance with principles of service approach

According to the proposed IDEF 3 model of work organization, an analysis in the framework of proposed solutions should be carried out:

a) prior to the synthesis of description of rational architecture of the created IS for mutual consistency of separate functional requirements;

b) in the course of the synthesis of variants of description of rational architecture of the created IS for incompleteness of stated functional requirements by identifying representations of requirements that duplicate each other;

c) after selection of description of the rational architecture of the created IS by means of identifying missed or illogical functional requirements.

5. Results of the development of methods for the analysis of functional requirements to an information system for consistency

The basic prerequisite for consistency analysis of functional requirements is the use of a unified formal description of representation of requirements to IS at knowledge level. The structure and content of these representations is determined by the patterns of design of requirements to IS at knowledge level, considered in [14, 15], and can be represented by the model of the following form [16]:

$$K_i^f = \{< d_n^{ij}, \{< d_{el\_fr}^{ij}, d_{el\_fr\_t}^{ij} >\}, < d_g^{ij}, \{< d_{el\_if}^{ij}, d_{el\_if\_t}^{ij} >, \}>,$$
$$< d_{fr\_rel\_n}^{ij}, \{< d_{el\_fr\_rel}^{ij}, d_{el\_fr\_rel\_t}^{ij} >\}>\}, \quad (1)$$

where d_n^{ij} is the description of a frame's name; $d_{el\_fr}^{ij}$ is the description of a frame's element; $d_{el\_fr\_t}^{ij}$ is the description of the type of a frame's element; d_g^{ij} is the description of the interface's name; $d_{el\_if}^{ij}$ is the description of the interface's element; $d_{el\_if\_t}^{ij}$ is the description of the type of an interface's element; $d_{fr\_rel\_n}^{ij}$ is the description of relation between interfaces and/or frames; $d_{el\_fr\_rel}^{ij}$ is the description of the relation element; $d_{el\_fr\_rel\_t}^{ij}$ is the description of the type of the relation element.

Model (1) makes it possible to consider representation of functional requirements 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 true, will be considered inconsistent [19]:

a) frames or interfaces with the same or a similar name contain sets of elements that do not intersect;

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.

In this case, consistency analysis of functional requirements should not depend on which representations of requirements at the knowledge level are analyzed from the perspective of a Provider, from the perspective of a Consumer, or system-wide.

To identify the first situation, it is proposed to carry out consistency analysis of frames and frame interfaces of separate requirements. In the course of development of a separate *i*-th functional requirements, it is necessary to separate a set of representations of functional requirements at the knowledge level $\{K_j^f\}$, which were formed earlier. This set can be built by one of the following methods or by their combination:

a) from representations of previously stated versions of the *i*-th requirement (for the case of updating or development of IS or its separate functions);

b) from previously stated requirements to a created or updated IS (during IS creation or updating).

Then the method of consistency analysis of separate representation frames K_i^f involves the following stages.

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

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

Stage 3. If condition

$$(d_n^{ia} = d_n^{jb}) \wedge$$
$$\wedge((\{< d_{el\_fr}^{ia}, d_{el\_fr\_t}^{ia} >\} / \{< d_{el\_fr}^{jb}, d_{el\_fr\_t}^{jb} >\}) = \varnothing) \quad (2)$$

is met, acknowledge existence of inconsistency between the *i*-th and the *j*-th functional requirements in descriptions of frames fr^{ia} and fr^{jb}, After that proceed to Stage 5.

Stage 4. If condition

$$((d_n^{ia} \subseteq d_n^{jb}) \vee (d_n^{ia} \supseteq d_n^{jb})) \wedge$$
$$\wedge(\{< d_{el\_fr}^{ia}, d_{el\_fr\_t}^{ia} >\} \bigcap \{< d_{el\_fr}^{jb}, d_{el\_fr\_t}^{jb} >\}) = \varnothing) \qquad (3)$$

is met, acknowledge existence of inconsistency between the *i*-th and the *j*-th functional requirements in descriptions of frames fr^{ia} and fr^{jb}.

Then acknowledge inconsistency between the *i*-th and the *j*-th functional requirements in frames and descriptions

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

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

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

The method for inconsistency analysis of separate interfaces of representation K_i^f will be similar to the method of analysis of separate frames, discussed above.

To analyze the second possible situation, the method of consistency analysis of separate relations of representation K_i^f, involving the following stages, is proposed.

Stage 1. Select representation $K_j^f \in \{K_j^f\}$, not considered earlier.

Stage 2. Select relation $rel^{jm} \in K_j^f$, not considered earlier, as well as frames fr^{jc} and fr^{jd}, forming relation rel^{jm}.

Stage 3. Find in representation K_i^f frames fr^{ia} and fr^{ib}, for which condition is satisfied

$$(fr^{jc} = fr^{ia}) \wedge (fr^{jd} = fr^{ib}).$$

If there are no such frames, proceed to Stage 7.

Stage 4. Check existence of relation $rel^{ik} \in K_i^f$, determined in frames fr^{ia} and fr^{ib}. If this relation does not exist, acknowledge existence of inconsistency between the *i*-th and the *j*-th functional requirements in description of relation $rel^{jm} \in K_j^f$ and absence of similar description $rel^{ik} \in K_i^f$.

Stage 5. Verify if condition $rel^{jm} = rel^{ik}$ is satisfied. If the condition is not satisfied, acknowledge existence of inconsistency between the *i*-th and the *j*-th functional requirements in description of relations $rel^{jm} \in K_j^f$ and $rel^{ik} \in K_i^f$.

Stage 6. Exclude relation rel^{jm} from further consideration. If not all relations $rel^{jm} \in K_j^f$, were considered, proceed to Stage 2.

Stage 7. Exclude representation K_j^f from further consideration. If not all representation of set $\{K_j^f\}$, were considered, select representation $K_j^f \in \{K_j^f\}$, not considered earlier, and proceed to Stage 2. Otherwise, complete application of the method.

The results of application of these methods will be the lists of inconsistencies between separate functional requirements, identified during development of separate functional requirements to the created IS. Identification of these inconsistencies makes it possible to adjust publications of functional requirements in the course of formation, thus reducing the time for analysis of stated requirements.

6. Results of the development of methods for detecting the duplicated, missed, and illogical functional requirements to an information system

Incompleteness analysis of stated functional requirements and identification of duplicated requirements are performed as a result of application of method for the synthesis of architecture description variants of the created IS, described in [17]. This will make it possible not to separate incompleteness analysis in a particular work, requiring a special method [19].

To identify missing and illogical functional requirements, we will introduce the following definitions. Functional requirement to the IT-service that is not associated with any IT-service by data flows will be called illogical. One or more functional requirements that could eliminate the identified illogicality of a requirement will be called missing [19].

To detect illogical and missing requirements, it is proposed to use the term "ontological point", introduced in [18]. Ontological point is a separate branch of frame taxonomy, which exists in representation of the *i*-th functional requirement K_i^f and formally described by expression in the form of [18]:

$$OntPD =< FR_{OntPD} =$$
$$= (fr_i, ..., fr_k, ..., fr_j), C^{gen}, G_{OntPD} =$$
$$= (< fr_i, fr_{i+1}, C^{gen} >, ..., < fr_{k-1}, fr_k, C^{gen} >,$$
$$< fr_k, fr_{k+1}, C^{gen} >), ..., < fr_{j-1}, fr_j, C^{gen} >) >, \qquad (4)$$

when the condition is satisfied

$$\forall fr_k \in FR_{OntPD} \exists < fr_{k-1}, fr_k, C^{gen} > \bigcap$$
$$\bigcap < fr_k, fr_{k+1}, C^{gen} > \in G_{OntPD},\ i < k < j, \qquad (5)$$

where $OntPD$ is the formalized description of an ontological point; FR_{OntPD} is the subset of frames, forming ontological point $FR_{OntP} \subseteq FR$; G_{OntPD} is the set of representations, which set generalization relations between the frames, included in frame FR_{OntPD}; *i* is the identifier of the root frame of an ontological point; *j* is the identifier of the frame-list of an ontological point.

Then the *i*-th functional requirement can be considered illogical, if all its ontological points are present only in representation $K_i^{f_{IS}}$. To characterize illogicality degree of the *i*-th functional requirement, it is proposed to use indicator Irr_i, the value of which is determined from the formula [14]

$$Irr_i = \frac{\left|\{OntPD_{im}^{irr}\}\right|}{\left|\{OntPD_{im}\}\right|} \times 100\,\%, \qquad (6)$$

where $OntPD_{im}^{irr}$ is the description of the *m*-th ontological point, which exists only in representation $K_i^{f_{IS}}$; $OntPD_{im}$ is the description of the *m*-th ontological point, which exists in representation $K_i^{f_{IS}}$.

Then, to detect illogical functional requirements, it is proposed to use the method, involving the following stages

Stage 1. Choose representation $K_i^{f_{IS}}$, which was not considered earlier, from the set of representations $\{K_i^{f_{IS}}\}$, building description of the rational architecture of the created IS.

Stage 2. Separate the set of descriptions of ontological points $\{OntPD_{im}\}$ of representation $K_i^{f_{IS}}$. Accept the set of descriptions of illogical ontological points

$$\{OntPD_{im}^{irr}\} = \{OntPD_{im}\}.$$

Stage 3. Select representation $K_j^{f_{IS}}$, $j \neq i$, which was not considered earlier, from the set of representations, building a description of the rational architecture of the created IS. Accept.

$$\{K_j^{f_{IS}}\} = \{K_i^{f_{IS}}\} / K_i^{f_{IS}}.$$

Stage 4. Build the set of descriptions of ontological points $\{OntPD_{jm}\}$ of representation $K_j^{f_{IS}}$.

Stage 5. Accept the set of descriptions of illogical ontological points

$$\{OntPD_{im}^{irr}\} = \{OntPD_{im}^{irr}\} / \{OntPD_{jm}\}.$$

If $\{OntPD_{im}^{irr}\} = \varnothing$, proceed to Stage 8.

Stage 6. Exclude representation $K_j^{f_{IS}}$ from subsequent consideration. If $\{K_j^{f_{IS}}\} \neq \varnothing$, proceed to Stage 3.

Stage 7. Calculate the value of indicator Irr_i from formula (6) and acknowledge illogicality of the i-th functional requirement with separation of illogical ontological points.

Stage 8. Exclude representation $K_i^{f_{IS}}$ from subsequent consideration. If $\{K_i^{f_{IS}}\} \neq \varnothing$, proceed to Stage 1. Otherwise, complete application of the method.

The proposed method makes it possible to considerably simplify the procedure of detection of illogical functional requirements. It is offered to make this simplification by automation of operations on detection of illogical objects of a subject area of the studied requirement to IS-service. The objects, the data about which are not supposed to be used in any other IT-service, are illogical.

7. Discussion and verification of methods for the analysis of functional requirements to an information system

Verification of the developed methods was carried out in the course of pilot design of the functional module of labor safety (FM LS). The Consumer of IT-services put forward the following requirements to a given module [20]:

a) "to implement the function of keeping records of information about an enterprise and processes (operations), performed at a given enterprise" (functional requirement 1);

b) "to implement the function of keeping records of personnel data (data about the staff of an enterprise), which are minimally required for making management decisions to provide labor safety of an enterprise (functional requirement 2);

c) "to implement the function of compiling and keeping reference book of HPF, which act or can act in the course of execution of particular processes or works of an enterprise" (functional requirement 3);

d) "to implement the function of keeping record of results of observations of the actions of each HPF in the course of execution of processes or particular works of an enterprise" (functional requirement 4);

e) "to implement the function of the forecast of the HPF complex influence on the organism of an enterprise's employee, performing a separate process or work at an enterprise" (functional requirement 5).

To separate knowledge, these requirements were published in the form of data flow diagrams. These diagrams were the source data for application of the method of development of representation of functional requirements at the information level. The results of application of this method became the source information for implementation of the methods of development of functional requirements to FM at the knowledge level, considered in [16].

The result of application of the method of development of representations of functional requirements to FM LS at the knowledge level of a Consumer of IT- services is the fragments of frame networks $K_1^{f_U} - K_5^{f_U}$, describing separated requirements in accordance with the model (1), shown in Fig. 2–6 in the form of the diagrams of classes.

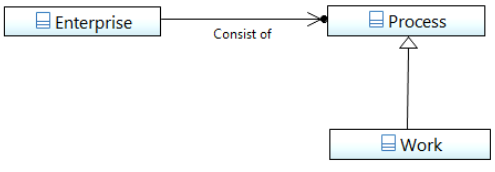

Fig. 2. Diagram of classes that describes representation of functional requirement 1 at the knowledge level of the Consumer of IT-services

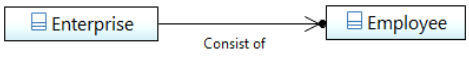

Fig. 3. Diagram of classes that describes representation of functional requirement 2 at the knowledge level of the Consumer of IT-services

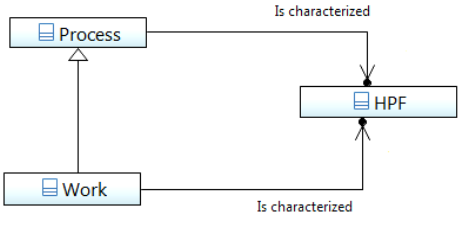

Fig. 4. Diagram of classes that describes representation of functional requirement 3 at the knowledge level of the Consumer of IT-services

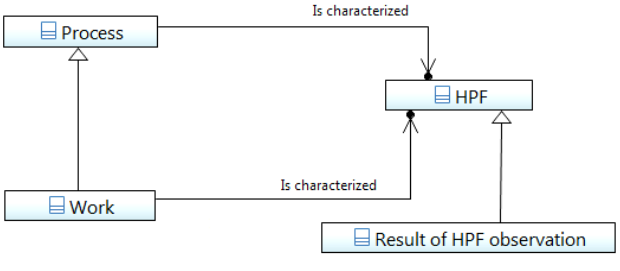

Fig. 5. Diagram of classes that describes representation of functional requirement 4 at the knowledge level of the Consumer of IT-services

Application of the method for consistency analysis of separate frames for representations $K_1^{f_U} - K_5^{f_U}$ shows no inconsistencies between frames of these representations, because:

a) frames with identical names use the same identifiers to indicate their presence in representations of various requirements at the knowledge level, which leads to failure to satisfy the condition (2);

b) for combinations of frames "Employee" and "Employee's state", "Employee" and "Forecast of a change in employee's state", "Employee's state" and "Forecast of a change in employee's state", "HPF" and "Result of HPF observation", the second part of condition (3) is not satisfied.

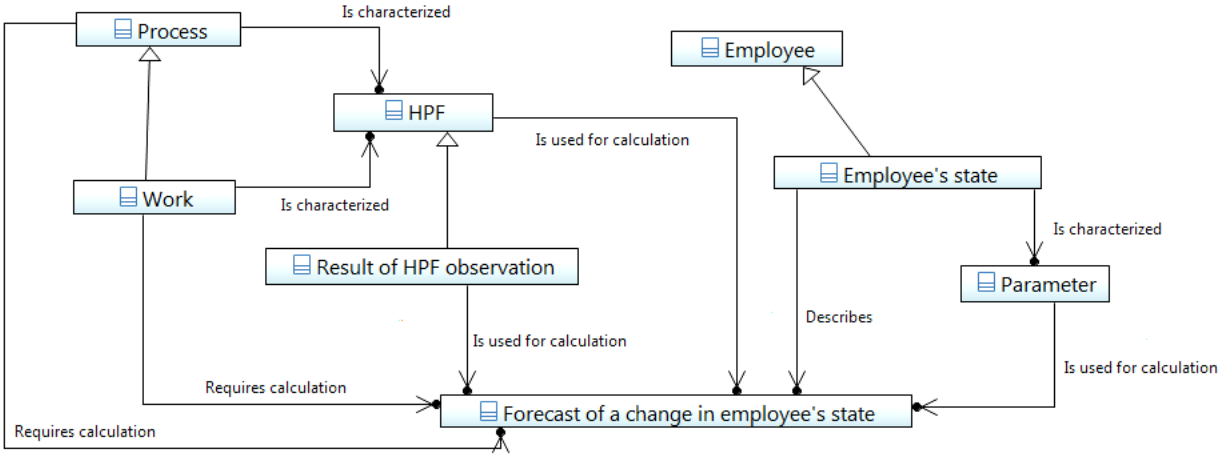

Fig. 6. Diagram of classes that describes representation of functional requirement 5 at the knowledge level of the Consumer of IT-services

All descriptions of relations with identical names (for example, relations "Is characterized" between the pair of frames "Process" and "HPF", as well as between the pair of frames "Work" and "HPF") do not coincide with each other. Therefore, application of the method of consistency analysis of separate relations of representations $K_1^{f_U} - K_5^{f_U}$ also indicates absence of inconsistencies.

The results of application of the methods for consistency analysis of representations of functional requirements $K_1^{f_U} - K_5^{f_U}$ makes it possible to conclude that representations of requirements at the knowledge level from the perspective of a Consumer are consistent and, therefore, allow formation of a unified information representation of a control object.

The progress of application of the method for detection of illogical functional requirements for FM LS for representations $K_1^{f_U} - K_5^{f_U}$ is given in Table 1.

Table 1

Results of application of the method for detection of illogical functional requirements for representations $K_1^{f_U} - K_5^{f_U}$

Number of analyzed requirement	Number of iteration of method	Number of ontological points of requirement	Number of illogical ontological points of requirement	Number of compared requirement	Number of coinciding ontological points	Value Irr of analyzed requirement, %
1	1	2	2	2	1	50
1	2	2	1	3	1	– (requirement is logical)
2	1	2	2	1	1	50
2	2	2	1	3	0	50
2	3	2	1	4	0	50
2	4	2	1	5	1	– (requirement is logical)
3	1	2	2	1	1	50
3	2	2	1	2	0	50
3	3	2	1	4	0	50
3	4	2	1	5	0	50
4	1	2	2	1	1	50
4	2	2	1	2	0	50
4	3	2	1	3	0	50
4	4	2	1	5	1	– (requirement is logical)
5	1	5	5	1	1	80
5	2	5	4	2	0	80
5	3	5	4	3	0	80
5	4	5	4	4	1	60

Based on the results of application of the method for detection of illogical functional requirements, it is possible to make the following conclusions.

Firstly, functional requirement 3 to FM LS is 50 % illogical. This illogicality is caused by the use of a simplified ontological point "HPF" for description of the term of subject area "Harmful industrial factor". In functional requirements 4 and 5, a detailed ontological point "HPF+Result of HPF observation" is used for the same purpose.

Secondly, functional requirement 5 to FM LS is 60 % illogical, since it uses ontological points "Employee + Employee's status", "Parameter", "Forecast of a change in employee's state".

To decrease illogicality of requirement 5 to FM LS, it was proposed to reuse a unified representation of the term of subject area "Employee" in the form, shown in Fig. 7. This unified representation is based on the existing project decision [14]. Therefore, the names of the frames in Fig. 7 are shown in the form, corresponding to the names of the elements of design solution.

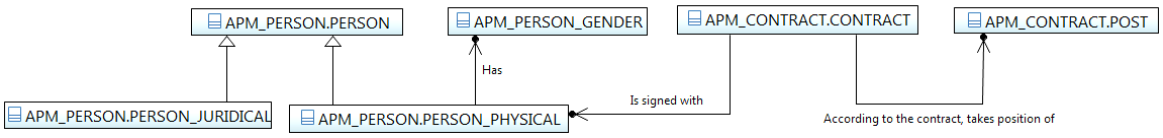

Fig. 7. Diagram of classes that describes the unified representation of term the "Employee" in subject area

The use of the unified representation for modification of descriptions of requirements 2 and 5 for FM LS, proposed in Fig. 7, made it possible to decrease the value of indicator Irr for requirements 2 and 5 for FM LS from 60 % to 37.5 %. The values of indicator Irr for the rest of the functional requirements to FM LS remained unchanged.

Results of verification of the developed methods for analysis show the possibility of decreasing inconsistency and illogicality of functional requirements to IS before synthesis of description of architecture of the created IS. The consequence of this is possibility to decrease the costs of integration the results of development of separate functional requirements within the created FM LS.

The verification results allow us to identify major shortcomings of the developed methods of analysis. The first of these shortcomings can be their orientation to presentation of functional requirements to IS at the knowledge level. If these representations were not created during development of functional requirements to IS, application of the proposed methods of analysis becomes very difficult. That is why one of the trends of subsequent research in this area can be consideration of possibility of application of the developed methods for analysis of representations of functional requirements for IS at the information level. These representations can be publications of requirements in the form of a text, structural and objective visual patterns (e. g., data flow diagrams or diagrams of usage variants).

Another considerable drawback of the developed methods should be considered the need to preserve representations of functional requirements to IS in the form of artifacts for subsequent usage. Thus, for example, for the methods of consistency analysis of separate frames and relations of representation K_i^f, storage of such representations is an obligatory precondition for implementation of these methods. The need to store separate artifacts significantly complicates not only specific applications of requirements development and analysis, but also general methodologies of IS design. Thus, for example, costs of practical implementation of modern architectural framework of design of 9.1 TOGAF systems [21, 22] are much higher than costs of practical implementation of another modern architectural framework of design of RM-ODP systems [23, 24]. This is due to the fact that in TOGAF 9.1, information and knowledge about the created system are stored in special semantic repository of architecture objects. In RM-ODP, information and knowledge about the created system are stored in the form of an interrelated totality of system's descriptions in five viewpoint languages (taking into account distribution transparencies). Relations of this language totality are determined by in a special agreement, establishing architectural semantics of open distributed data processing for different languages of formal specifications.

The desire to eliminate this drawback leads to the need to address a whole range of theoretical and applied problems, among which the following ones should be separated:

a) the problem of synthesis of description of architecture of the created system, based on texts, created with the use of architecture description languages;

b) comparative analysis of efficiency, quality, and cost of architecture synthesis of the created system, based on a set of artifacts (knowledge-oriented models), or a set of system's descriptions in architecture description languages;

c) optimization of the synthesized architecture of the created system.

Solving these problems is complicated by the fact that the problems of formal description of the system's architecture, as well as the processes of synthesis and architecture, is far from the final solution. Currently, the main types of architectures of information systems are distinguished. However, the problems of substantiated selection of an optimal architecture for the created system and transformation of the architecture of the operated system due to changes in requirements to this system are far from final resolution.

8. Conclusions

1. The order of work was specified aimed at analyzing the stated functional requirements to the information system within the framework of the processes of requirements development and analysis. It was proposed to arrange performing certain types of requirements analysis in parallel to the main work on the development of right holders' requirements and requirements to IS. The IDEF 3 model was constructed for the totality of work on the development and analysis of functional requirements to IS. The proposed parallel organization of work makes it possible to reduce time cost for analysis of the stated requirements.

2. Methods for consistency analysis of separate frames and relations of representation of a functional requirement to IS at the knowledge level were developed. These methods make it possible to detect inconsistencies, caused by using different sets of attributes and procedures for description of frames or interfaces with the same names. These methods also enable us to detect existence of different relations between similar frames or frames and interfaces. The developed methods make it possible to automate the works on analysis of stated functional requirements, which offers an opportunity to detect inconsistencies between such requirements in the process of their statement.

3. The method for illogicality analysis of the stated functional requirements to IS was developed. The essence of the method is in quantitative assessment of illogicality degree based on frequency of occurrence of separate branches of frame taxonomy in representations of functional requirements to IS at the knowledge level. Application of this method allows us to automate works on detection of illogical functional requirements. In future, the proposed quantitative assessment of a degree of logicality of a functional requirement (6) can be considered as a basis for determining of the system effect from implementation of IS system in general and its particular functions.

References

1. Project Management Body of Knowledge. 5-th ed. Newton Square: Project Management Institute, Inc., 2013. 586 p.
2. GOST ISO/MEK 15288-2005. System engineering. System life cycle processes. Moscow: Standartinform, 2006. 57 p.
3. Maguire M., Bevan N. User Requirements Analysis // Usability. 2002. P. 133–148. doi: 10.1007/978-0-387-35610-5_9
4. Stowell F., Cooray S. The Appreciative System, Learning And Its Impact Upon Is Design // Communications of the Association for Information Systems. 2017. Vol. 40. P. 93–119. doi: 10.17705/1cais.04006
5. Ali N., Lai R. A method of requirements elicitation and analysis for Global Software Development // Journal of Software: Evolution and Process. 2016. Vol. 29, Issue 4. P. e1830. doi: 10.1002/smr.1830
6. Requirements cybernetics: Elicitation based on user behavioral data / Liu L., Zhou Q., Liu J., Cao Z. // Journal of Systems and Software. 2017. Vol. 124. P. 187–194. doi: 10.1016/j.jss.2015.12.030
7. Asteasuain F., Braberman V. Declaratively building behavior by means of scenario clauses // Requirements Engineering. 2016. Vol. 22, Issue 2. P. 239–274. doi: 10.1007/s00766-015-0242-2
8. Yu Y.-J., Liu C. Little Model in Big Data: An Algebraic Approach to Analysing Abstract Software Behaviours // Ruan Jian Xue Bao/ Journal of Software. 2017. Vol. 28, Issue 6. P. 1488–1497.
9. The Value of Exact Analysis in Requirements Selection / Li L., Harman M., Wu F., Zhang Y. // IEEE Transactions on Software Engineering. 2017. Vol. 43, Issue 6. P. 580–596. doi: 10.1109/tse.2016.2615100
10. Applications of ontologies in requirements engineering: a systematic review of the literature / Dermeval D., Vilela J., Bittencourt I. I., Castro J., Isotani S., Brito P., Silva A. // Requirements Engineering. 2015. Vol. 21, Issue 4. P. 405–437. doi: 10.1007/s00766-015-0222-6
11. Serna M. E., Bachiller S. O., Serna A. A. Knowledge meaning and management in requirements engineering // International Journal of Information Management. 2017. Vol. 37, Issue 3. P. 155–161. doi: 10.1016/j.ijinfomgt.2017.01.005
12. Nguyen T. H., Grundy J. C., Almorsy M. Ontology-based automated support for goal–use case model analysis // Software Quality Journal. 2015. Vol. 24, Issue 3. P. 635–673. doi: 10.1007/s11219-015-9281-7
13. Supporting agent oriented requirement analysis with ontologies / Lopez-Lorca A. A., Beydoun G., Valencia-Garcia R., Martinez-Bejar R. // International Journal of Human-Computer Studies. 2016. Vol. 87. P. 20–37. doi: 10.1016/j.ijhcs.2015.10.007
14. Levykin V. M., Ievlanov M. V., Kernosov M. A. Pattern planning of requirements to the informative systems: design and application: monograph. Kharkiv: The «Kompaniya «Smit LTD», 2014. 320 p.
15. Levykin V., Ievlanov M., Neumyvakina O. Developing the models of patterns in the design of requirements to an information system at the knowledge level // Eastern-European Journal of Enterprise Technologies. 2017. Vol. 5, Issue 2 (89). P. 19–26. doi: 10.15587/1729-4061.2017.110586
16. Ievlanov M. Methods of presenting formulated requirements to the information system at the level of knowledge // Eastern-European Journal of Enterprise Technologies. 2015. Vol. 4, Issue 3 (76). P. 4–11. doi: 10.15587/1729-4061.2015.47535
17. Ievlanov M. Development of the model and method of selecting the description of rational architecture of information system // Eastern-European Journal of Enterprise Technologies. 2016. Vol. 1, Issue 2 (79). P. 4–12. doi: 10.15587/1729-4061.2016.60583
18. Evlanov M., Vasiltcova N., Panfyorova I. Modeli i metody syntezu opysu ratsionalnoi arkhitektury informatsiynoi systemy // Visnyk naukovoho universytetu «Lvivska politekhnika». Seriya: Informatsiyni systemy ta merezhi. 2015. Issue 829. P. 135–152.
19. Ievlanov M., Neumyvakina O. Metody analiza sformulirovannyh funkcional'nyh trebovaniy k informacionnoy sisteme // Informacionnye sistemy i tekhnologi: materialy 4-y Mezhdunarodnoy nauch.-tekhn. konf. Kharkiv: NTMT, 2015. P. 52–53.
20. Ievlanov M., Serdyuk N. Forming and analysis of requirements to information-analytical system of management by safety of labour in enterprise // Technology audit and production reserves. 2015. Vol. 4, Issue 3 (24). P. 41–45. doi: 10.15587/2312-8372.2015.47972
21. Core Concepts. URL: http://www.togaf.org/togaf9/chap02.html#tag_03_01
22. Temnenco V. TOGAF or not TOGAF: Extending Enterprise Architecture beyond RUP // IBM. URL: https://www.ibm.com/developerworks/rational/library/jan07/temnenco/index.html?S_TACT=105AGX99&S_CMP=CP
23. GOST ISO/MEK 10746-3-2001. Information technology. Open Systems Interconnection. Data control and Open Distributed Processing. Reference model. Part 3. Architecture. Moscow: IPK Izdatelstvo standartov, 2002. 57 p.
24. GOST ISO/MEK 10746-4-2004. Information technology. Open Distributed Processing. Reference Model. Part 4. Architectural semantics. Moscow: IPK Izdatelstvo standartov, 2004. 34 p.