UDK 004.415.2

A. Stenin, K. Melkumyan, M. Soldatova, O.Polshakova

INFOLOGICAL MODEL DEVELOPMENT OF INNOVATIVE SOFTWARE PRODUCT

Annotation. An infological model of the process of managing the creation of a subject-oriented software product proposed, on which the development of an innovative software product interpreted as an information object that changes in content and structure in the process of its creation. When there are innovations in the subject area in the infological model, it is possible to change the control actions in acceptable values in accordance with changes in the current and final requirements for the subject area of the innovative software product. The procedure for making innovative decisions, which implemented, using an intelligent decision support system and is a cyclical process of human-computer interaction.

Keywords: subject area, infological model, innovative software product, procedure for making innovative decisions, intelligent decision support system

Introduction

It’s known that knowledge about the object under development determines the structure of its development management system. This is true for most objects of different physical nature and their purpose, including innovative software product. However, software products have an important feature: knowledge about them has a relatively large dynamic component, so large that it often taken into account even in operational management. A high level of validity of innovative solutions in the process of managing the development of an innovative software product cannot achieved without a strong cognitive ability of the developer in relation to the software product. This is due to the evolution of knowledge in the subject area. Today, decision-making involves a "field of knowledge" that is uncertain not only in terms of volume, but also in terms of dimension. In the process of developing a solution, dynamically changing economic, technological, organizational, natural, and other aspects taken into account. Sources of information can be the knowledge of experienced specialists, literature, normative materials of scientific and design organizations, and Internet resources [1-3].

In this paper, an infological model of the process of managing the development of innovation software product is proposed. The essence of the proposed approach is that the development of innovation software product interpreted as an information object that changes meaningfully and structurally in the process of its creation. Hence, updating information about this object, for which the innovative software product developed, requires constant adoption of decisions that are adequate to the changed situation.

Development of innovation software product is multi-alternative process, i.e. there is a problem of multi-criteria, which, as a rule, requires the involvement of intelligent decision support systems (DSS). This because, firstly, the presence of the human factor in the process

© A. Stenin, K. Melkumyan, M. Soldatova, O.Polshakova

of developing innovation software product introduces greater uncertainty. And, second, you need to look at the full range of permissible decisions in the use of innovative software product requires the development of automated methods for extracting knowledge of a given subject area from a variety of sources, including from Internet sources.

Analysis of software products development process models

The information structure of the innovative software product provide for the possibility of adding new knowledge to the management system. Each observation can also add new knowledge about the software product, which not provided for by the regulations, but, in the future, increases the effectiveness of management. Despite the high degree of uncertainty in the process of creating a software product due to the presence of the human factor, the process of creating a software product is quite deterministic and consists of a number of stages that can built according to different schemes. Let's consider the most well-known models of development innovation software product [4,5,9-11].

Cascade model. The first well-known model that really structures the development process is the cascade or waterfall model (Fig.1) [5,8]. This model divides the process of creating a software product into successive stages (it was already used by various developers earlier, but neither the number nor the content of the stages was unified).

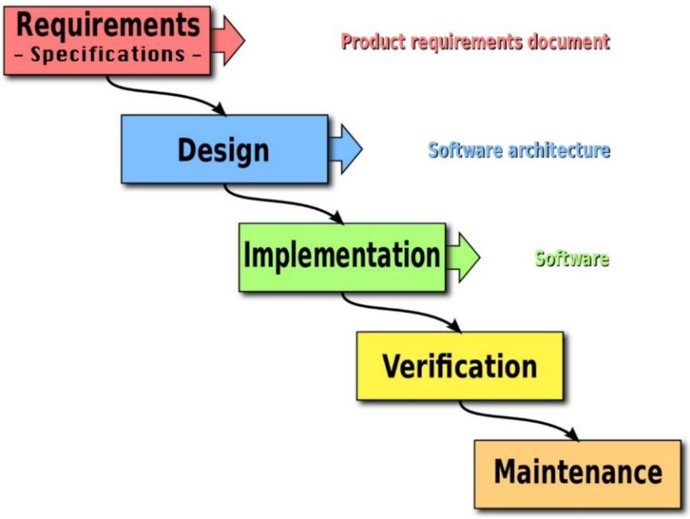

Fig. 1. Cascade model

However, the practical use of this model revealed many of its shortcomings, the main one being that it is more suitable for traditional engineering activities than for software development. In particular, one of the biggest problems was "predisposition" to possible inconsistencies between the resulting product and the requirements that were imposed on it. The main reason for this model is that a fully formed product appears only at the later stages of development. Since the work at different stages usually performed by different specialists, the project transferred from one group to another, the principle of a damaged phone turned out that the output was not exactly what intended at first.

V-model. It proposed precisely in order to eliminate the disadvantages of the cascade model, and the name-V-model, or hinge model, which name appeared because of its specific graphical representation (Fig.2) [5,10].

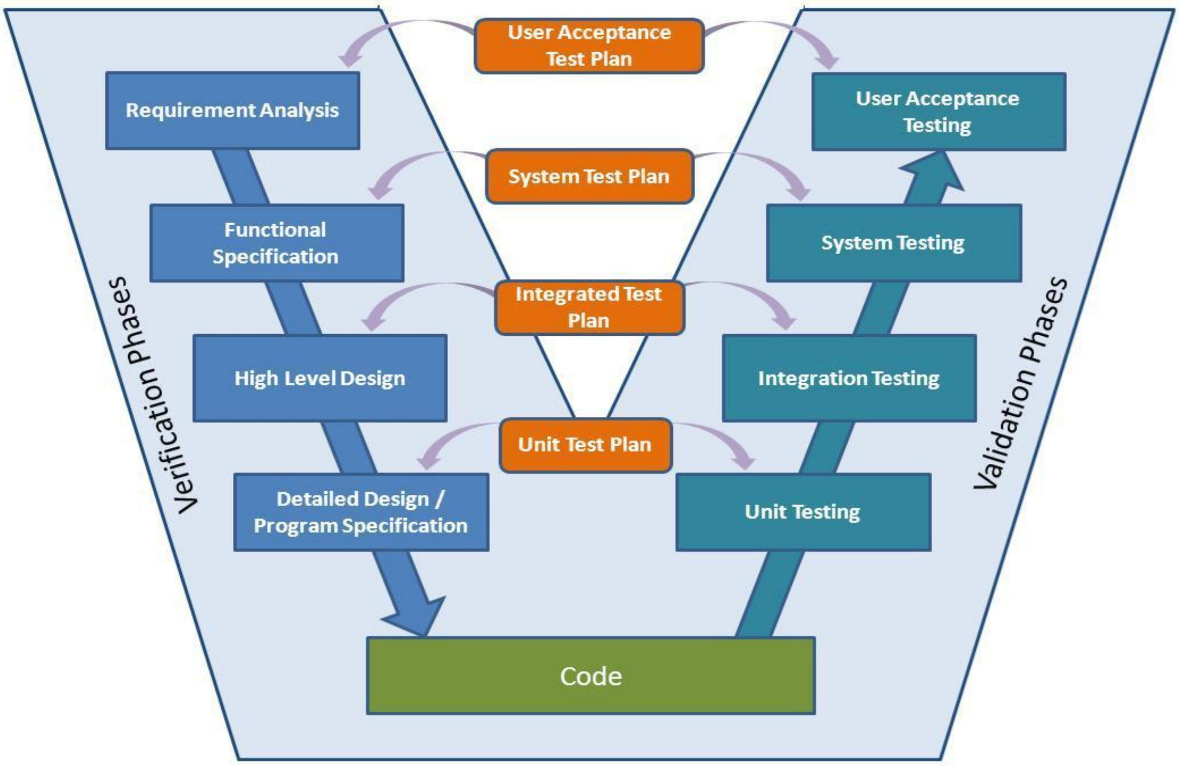

Fig. 2. V- model

The V-model made possible significantly improve the quality of software due to its focus on testing. Also largely solved the problem of compliance of the created product with the requirements due to verification and certification procedures at the early stages of development [the purple lines in the figure 2 indicate the dependence of the stages of planning (setting) the problem and testing (acceptance)] [7].

However, in General, the V-model is just a modification of the cascade model and has many of its disadvantages. In particular, both models are poorly adapted to possible changes in the customer's requirements. If the development process takes a long time (sometimes up to several years), the resulting product may actually be unnecessary for the customer, because their needs have changed significantly.

The issue of planning indicators of expected functionality is also important, since in these models it is nothing more than an assumption: in particular, it is almost impossible to determine what data processing speed the product will provide or how much memory it will take up at the stage of setting the task. If such requirements clearly stated in the terms of the contract between the customer and the contractor, it is likely that the resulting solution will not meet them, although this become known only at the final stages of development, when the main resources have already been spent.

 ISSN 1560-8956

Spiral model. Proposed by Barry Boehm in 1988, it was a significant breakthrough in understanding the nature of software development, although, by and large, it is a combination of two models: cascading and prototyping (Fig. 3) [6-8].

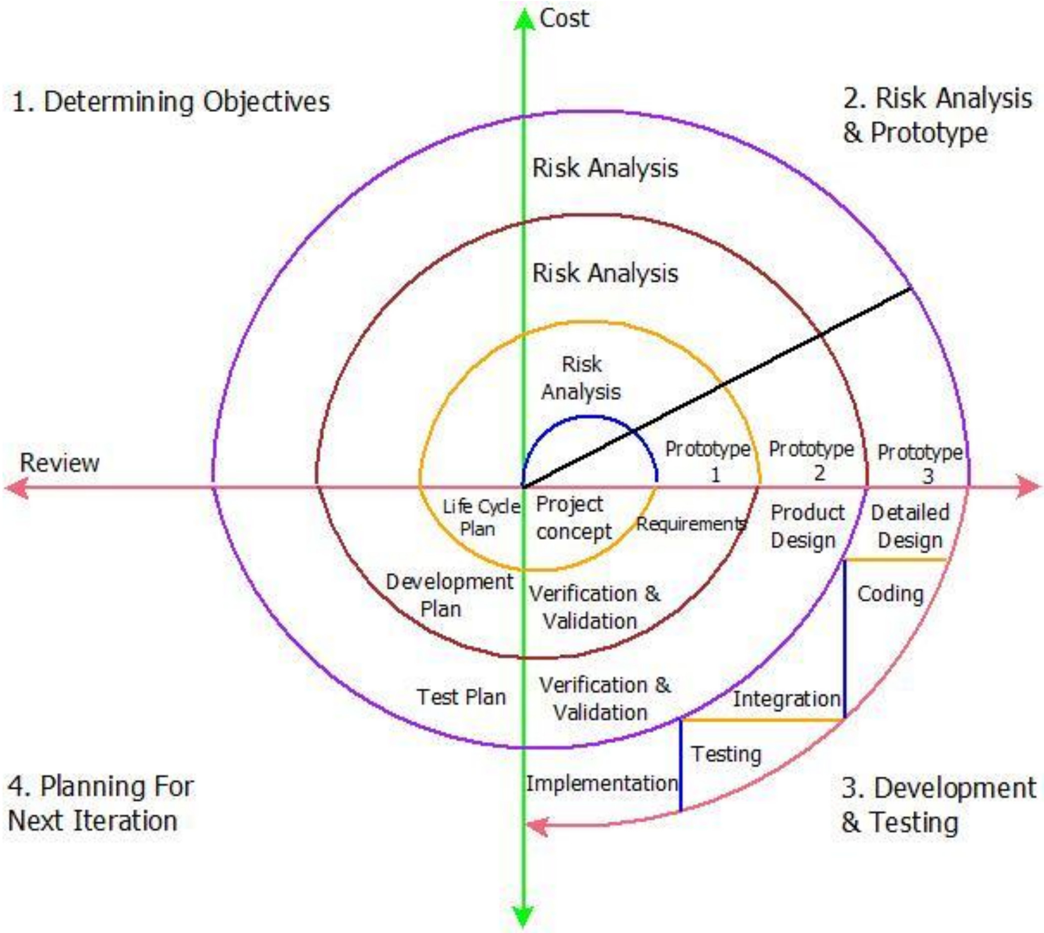

Fig. 3. Boehm spiral model

Boehm's spiral model focuses on design. In fact, software development takes place only at the last turn of the spiral according to the usual cascade model, but this preceded by several iterations of design based on prototyping – each iteration includes a stage of identifying and analyzing risks and the most complex tasks.

Since the spiral model mainly covers design, it is not widely used in its original form as a method for managing the entire software development life cycle. Main idea of spiral model, that the process of working on a project consist of cycles that go through the same stages, served as a starting point for the development of the infological model for managing the innovation software product development, which proposed below.

Problem statement

Project management methodology includes a set of models, methods, and software products used in the development and implementation of projects of various classes and types. The specificity of software development projects is that the result of development is immaterial – these are collective mental models written in a programming language. This leads to the fact that most of the software development projects completed with deadlines, budget overruns, and some projects not completed in principle.

One of the main tasks that directly affect the effectiveness of software development is the choice of the development process model. The problem is that there is no single optimal solution. The model should defined for each specific project and can change in a wide range, depending on the scale, novelty and criticality of the project, distribution of participants, and customer requirements. The choice of model also influenced by the possibility of further certification, which allows the developer to gain an advantage over competitors, attract new customers, and improve the image of the organization. Therefore, choosing a model for the software development process is a very important task when managing a project and a team of developers.

Infological model development of innovative software product

The essence of the proposed approach, based on this model, is that the development of innovative software product interpreted as an information object, which changes in content and structure during its creation. Therefore, the development process can described by an ordered sequence of states being developed, the last of which represents the finished software product, i.e.

$$o \rightarrow \dots \rightarrow i \rightarrow \dots \; n \qquad (1)$$

Here, each state *Si* characterized by a certain set of parameters $m_i^1\, m_i^2\, \dots m_i^m$, where i=$\overline{0,n}$. In this case, each state *Si* characterizes the degree of completion of the development of the innovation software product.

In the future, we will assume that each intermediate state *Si* corresponds to two integral estimates *Pi* and *Qi,* which exhaustively characterize the degree of completion of the innovation software product from the quantitative and qualitative sides. Obviously, the functions *P* and *Q* on an ordered set of states *Si* ($i=\overline{0,n}$)) must have an increasing character. In addition, we assume that the integral estimates of *P* and *Q* are independent of each other.

Further, we assume that development of innovation software product divided into separate subprocesses (development steps) that correspond to the development stages adopted in accordance with (1). We denote the increment of integral characteristics achieved at the *i*-th stage by $_{\Delta}P_i$ and $_{\Delta}Q_i$.

The process of managing the development of innovation software product consists in setting its control effect *u(i)* at each step, which determines the values ΔPi and ΔQi, and transfers the degree of innovation software product development from the state (P_{i-1},Q_{i-1}) to the state (P_i ,Q_i). Management of *u(i)* can considered as the choice of one of the alternative possible ways to ensure innovation of a software product. In this case, the transfer to a new state implemented by performing a certain set of procedures.

Naturally, at each step *i*, a number of natural and artificial restrictions are imposed on the control action *u(i).* Otherwise, *u(i)* can take values from a certain set of possible control actions, i.e.

$$u(i) \in V(i) \quad (2)$$

For $i=0$ we assume that $P_0 = Q_0 = 0$.

The values of the integral characteristics in the following steps determined by the formulas:

$$\begin{cases} P_i = \varphi(u(k), P_{i-1}); \\ Q_i = \gamma(u(k), Q_{i-1}); \\ (P_i, Q_i) = f(u(i), (P_{i-1}, Q_{i-1})); i = \overline{1,n} \end{cases} \quad (3)$$

By (Pi, Q i), we will understand the set of all states of the development process, to which it can be transferred from the initial state in i steps, using the next control $u(k) \in V(k), k = \overline{1,\iota}$.

Such a set is called the reachability set *(P_i, Q_i)*, which is defined using recurrent relations (4) of the form

$$\begin{cases} (P_k, Q_k) = F[u(k), (P_{k-1}, Q_{k-1})] \\ u(k) \in V(k), k = \overline{1,\iota}, i = \overline{1,n} \end{cases} \quad (4)$$

In the task of innovation software product development must identify the requirements, which must satisfied in ISP after the end of its development. Based on this, determine the indicators P_n and Q_n that characterize the final state of development, which should belong to a certain range of acceptable values

$$(P_n, Q_n) \in (P_n^{`}, Q_n^{`}) \quad (5)$$

Thus, the process of developing a ISP with control actions *V(u(i))* will be acceptable if *u(i)* transfer the ISP from the initial state to the final state, which will satisfy condition (5).

Based on this, in order to successfully achieve the goal of developing a ISP, the following condition must be met:

$$(P_i, Q_i) \cap (P_i^{`}, Q_i^{`}) \neq \emptyset, i = \overline{1,n} \quad (6)$$

Condition (6) means that the set of all ISP development states must be in the set of acceptable states of innovation software product in accordance with the requirements. Otherwise, if the innovation forecast has changed, it is necessary to either change the technical task for development and $(P_i^{`}, Q_i^{`}), i = \overline{1,n}$, or expand the scope of possible control actions *u(i)*, $i = \overline{1,n}$.

Let's assume that as a result of performing (i-1*)* steps, the innovation software product development process has moved to the state (P_{i-1}, Q_{i-1}). Then the set of acceptable control actions at i-th step defined as follows:

$$\begin{cases} V^{`}(i) = \{u(i): (P_i, Q_i) = f[u(i), (P_{i-1}, Q_{i-1})]\} \\ (P_i, Q_i) \in (P_i^{`}, Q_i^{`}), i = \overline{1,n} \end{cases} \quad (7)$$

As a result, the process of managing the development of innovation software product in its final form can written as:

$$u(i) \in V(i) \cap V^{`}(i), i = \overline{1,n} \quad (8)$$

Condition (8) means that from the point of view of emerging innovations in the development of innovation software product, it is possible to change the control actions in acceptable values in accordance with changes in current and final requirements. Condition (8) can satisfied by several control actions at each step.

In other words, on the above, in the process of developing an innovative software product, the problem of solving a multi-criteria problem arises in the conditions of uncertainty of the current situation, inconsistency and heterogeneity of data, and the evolution of the software market. Thus, the development of innovative software product is multi-alternative, i.e. there is a problem of multi-criteria, which, as a rule, requires the involvement of intelligent decision support systems (IDSS) [12,16]. Because, firstly, the presence of the human factor in the process of developing the innovation software product introduces greater uncertainty. And, second, you need to look at the full range of permissible decisions in the use of innovation software product requires the development of automated methods for extracting knowledge of a given subject area from a variety of sources, including from Internet sources. Infological model development of innovation software product, proposed in this paper, allows us to adapt the process to changes in this subject area.

The procedure for the adoption of innovative solutions in IDSS

Making an innovative decision in most cases consists in generating possible alternative solutions, evaluating them and choosing the best option. In difficult and critical moments, the decision-maker turns to experienced and knowledgeable people (experts).

In particular, when choosing a solution option in the process of developing an innovative software product, a large number of uncertain and contradictory factors must take into account. Contradictory factors occurs due to ambiguity in the assessment of situations, errors in the choice of priorities, which, in the end, greatly complicates decision-making. Uncertainty is an integral part of decision-making processes in the development of an innovative software product, and uncertainties in a particular subject area can divided into three classes [13]:

– an uncertainty, associated with incomplete knowledge of the problem to be solved;

– an uncertainty, associated with the inability to fully account for the reaction of the environment (demand market, current situation in the subject area, etc.);

– an uncertainty, associated with a wrong understanding of the tactical and strategic goals of creating a specific innovative software product by the decision-maker.

The organization of the innovation process of software development can shaped in the form of multi-connected graph of innovative technologies and strategies of subject area with a large number of links, i.e. of strategies and technologies in this subject area (Fig. 4). For the application of new technologies and strategies for the development of this area should base on the assessment of the potential scale of the diffusion base (initiative) innovation, measured by examining the possible markets and their formation use of positioning tools new software product. In the event that the expected scale of diffusion is large and this innovation can applied in various areas of production and, in the future, can use as a basis for meeting the needs in many industries and segments of the consumer market (a high degree of organizational distribution at the stage of creating the actual software product).

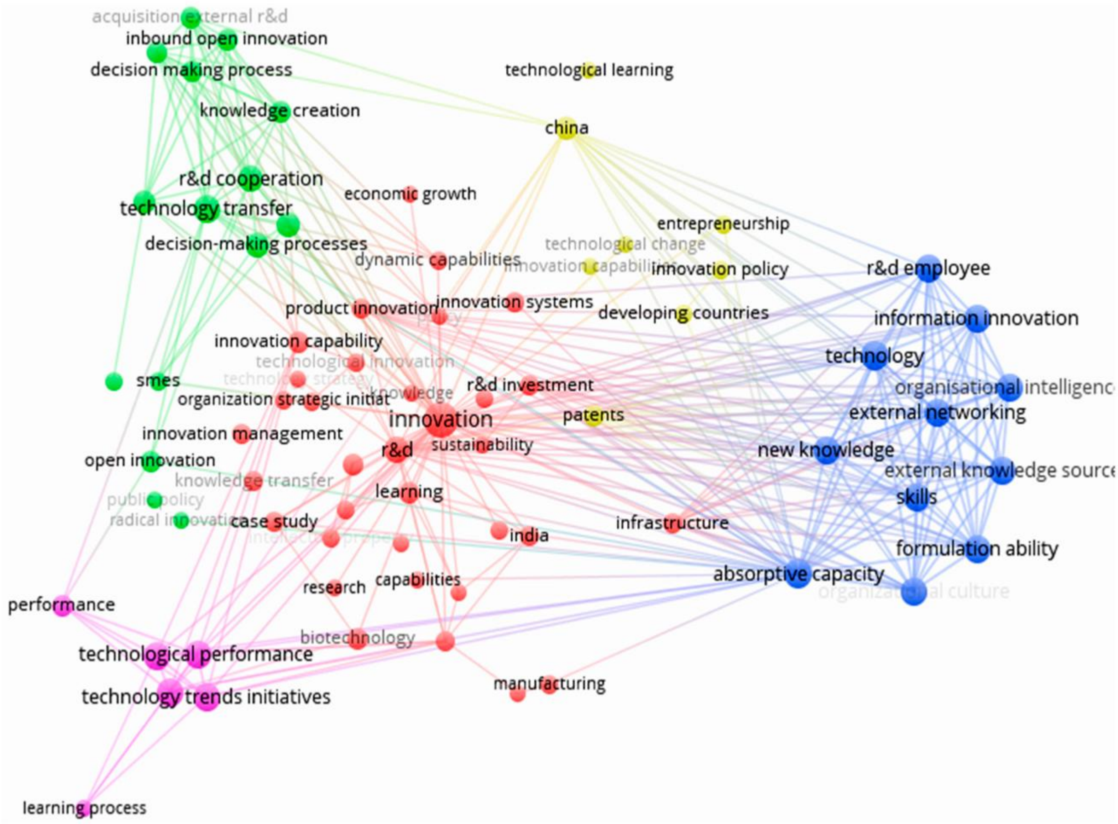

Fig.4. Multi-connected graph of innovative technologies and strategies of subject area

Research shows that decision makers, without additional analytical support, tend to use simplified and sometimes contradictory decision-making rules. In this case, the most effective tool for making a potentially better decision is IDSS. The main functions of such systems are [14]:

-assistance to the decision-maker in the analysis of initial information (assessment of the current situation and restrictions imposed by the external environment);

-identifying and ranking priorities, taking into account uncertainty in the assessments of the decision-maker, and forming his preferences;

-generating possible solutions (forming a list of alternatives);

-evaluating of possible innovative alternatives based on the preferences of the decision-maker and the constraints imposed by the external environment;

-analysis of possible consequences of decisions taken;

-choosing the potentially best solution at each stage (in each situation).

The procedure for making innovative decisions can proceed according to two main schemes:

-intuitive empirical (based on comparing the problem situation with previously encountered similar situations);

-formal heuristic (based on the construction and study of a model of a problem situation for a given specific subject area).

The decision-making procedure using IDSS is a cyclic process of human-computer interaction (fig. 5) and includes phases of analysis and problem formulation, phases of search and optimization of alternative solutions implemented using a computer.

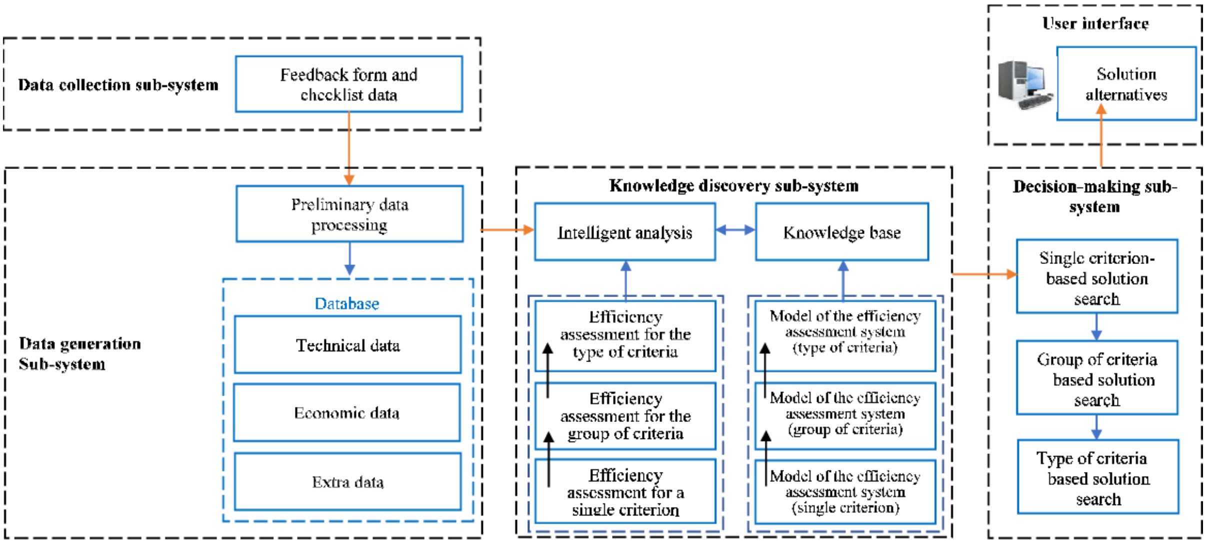

Fig. 5. General scheme of the decision-making procedure in the IDSS

Modern IDSS based on the use of specialized information storages (Data Warehouse) and OLAP (On-Line Analytical Processing) technologies - operational data analysis. The main purpose of OLAP technologies is dynamic multidimensional data analysis using an effective Data Mining tool, modeling and forecasting [15].

Conclusion

One of the main tasks that directly affect the effectiveness of software development is the choice of the development process model. The model should defined for each specific project and should vary widely depending on the scale, novelty and criticality of the project, the distribution of participants and the requirements of the customer. The choice of the model influenced by the possibility of further certification. This article offers an infological model for the development of an innovative software product. The essence of the proposed approach is that the development of an innovative software product interpreted as an information object, which changes in content and structure in the process of its creation. From the point of view of emerging innovations in the development of an innovative software product within the infological model, it is possible to change the control actions in acceptable values in accordance with changes in the current and final requirements for the subject area of an innovative software product. The procedure for making innovative decisions, which implemented using an intelligent decision support system and is a cyclical process of human-computer interaction.

 ISSN 1560-8956

REFERENCES

1. Kharin M. A. Review of automated knowledge extraction tools and their application in electronic document archives. Young scientist. 1, №5 (16), 2010. – P. 106-108.

2. Vityaev E. E. Extracting knowledge from data. Of computer knowledge. Models of cognitive processes. Monograph. - Novosibirsk: Novosibirsk state University. 2006. - 293 p.

3. Rogushina Yu. V. Intelligent search engines in the context of semantic web technologies / Yu. V. Rogushina, A. ya. Gladun // Intellectual analysis of information, no. 2, 2008. - P. 388-398.

4. Winston W. Royce led the development of large software systems. Process. IEEE WESKON, 1970, no. 6. - P. 1-9.

5. Orlov S. A. software development Technologies. - SPb.: publishing house Peter. - 2002. - 464 p.

6. Richard W. Selby. Software development: Barry W. Boehm's lifetime contributions to software development, management, and research. - John Wiley & Sons, 2007. - 834 p.

7. Dustin E., Rashka D., Paul D. Automated software testing. - M.: Lori. -2003 – - 590 p.

8. Solovyov N. A., Chernoprudova E. N. Software development automation Systems. - Orenburg. - LLC IPK "University". - 2012. - 230 p.

9. Bragina T. I., Tabunschik G. V. Comparative analysis of iterative software development models. –Zaporozhye: ZNGU, Radioelectronics, Informatics, management. No. 2, 2010. - P. 130-139.

10. Rakhimberdiev A. Modern software development processes [Text] / A. Rakhimberdiev / / RSDN Magazine. − 2006 − № 4. − P. 3–10.

11. Andreev N. D., Novikov F. A. Incremental subject-oriented process of application software development. Information and Control systems, no. 1, 2012. - P. 60-66.

12. Stenin A.A. , Pasko V.P., Shitikova I.G., Lemeshko V.A. Construction of subject-oriented IDSS of situational type. Scientific notes of the V. I. Vernadsky Tauride National University. Series: Technical Sciences, Volume 29 (68), No. 1, Part 2, 2018. - P.25-31.

13. Shcherbatov I.A. Classification of uncertainties for the analysis and control of complex pureformalizable systems. –Astrahan: SSTU Bulletin, №1(69),2013. –P.175-179/

14. Osipov V. P. et al. Intellectual core of the decision support system / Preprints of the Keldysh IPM. No.205, 2018. -23p. doi: 10.20948/prepr-2018-205 URL: http://library.keldysh.ru/preprint.asp?id=2018-205

15. Barseghyan A. A., Kupriyanov M. S., Stepanenko V. V., Kholod Technology I. I. of data analysis. Data Mining, Visual Mining, Text Mining, OLAP. Textbook-St. Petersburg: BHV Publishing House, 2007. - 384 p.

16. Stenin A., Lisovichenko O. , Drozdovich I., Stenin S. Analysis of passenger traffic and situational management of urban transport Bulgarian Journal for Engineering Design, issue. Mechanical Engineering Faculty, Technical University-Sofia №40, 2019. -P.7-12.