Using Cognitive Modeling During the Creation of IT Projects

Liubava Chernova, Anna Zhuravel, Lyudmila Chernova, Serhii Chernov, Irina Zhuravel

Admiral Makarov National University of Shipbuilding, Heroiv Ukrainy ave. 9, Mykolayiv, 54025, Ukraine

Abstract

The article is devoted to the applied aspects of modelling in the creation of IT projects. The article describes the main design stages, and the logical scheme of the IT project is presented. A hybrid, cascade-cyclic IT project formation algorithm is presented, designed to compensate for the shortcomings of both the cascade and iteration design paradigms and to ensure its quality and viability in short-term planning conditions. Cognitive modelling of IT projects refers to the process of using computational models to simulate human cognition and decision-making in the context of information technology (IT) projects. The goal of cognitive modelling in this context is to understand how people perceive, process, and respond to different aspects of IT projects, such as project management, software development, and user experience design. By simulating human cognition, cognitive modelling can help identify potential problems or inefficiencies in IT project processes, and suggest improvements or optimizations. It can also be used to evaluate the impact of different project management strategies, software development methodologies, and user interface designs on user behaviour and project outcomes.

Keywords 1

Cognitive modelling, logical scheme, cascade approach, spiral approach, iterative technology

1. Introduction

Cognitive modelling acquires a special specificity when designing modern IT projects, especially in those cases when a complex, multifaceted and large-scale activity is used for the automation of the project and the implementation of tasks for the IT project realization. As the experience of creating similar projects shows, in the vast majority of cases, it is necessary to involve heterogeneous, territorially distributed departments and institutions that work according to different methods, instructions and regulations. Such a situation arises, for example, in the process of automation on the scale of an entire industry, when the information process covers hundreds of disparate indicators, which are formed taking into account dozens of incompatible methods. In this and similar cases, there are increased requirements for the quality of design, since the further processing of not well-developed IT project is associated not only with significant costs of financial, material and human resources but also with the loss of time, which is critical in the system of public administration.

In this regard, ensuring the necessary design quality is impossible without a carefully developed model of the subject field, which, in turn, requires an appropriate modelling methodology, which determines the relevance of this article.

The purpose of this research was to study the applied aspects of modelling during the design of complex IT projects, when the cognitive factor becomes the decisive condition of the modelling process. To achieve the formulated goal, the following main tasks were set:

- analysis of directions of cognitive research in world practice;
- disclosure of general and private characteristics of the industry model and the IT project model;

IV International Workshop "IT Project Management" (ITPM 2023), On May 19, 2023, Warsawa
EMAIL: 19chls92@gmail.com (Liubava Chernova); nukannazhuravel@gmail.com (Anna Zhuravel); liudmyla.chernova@nuos.edu.ua (Lyudmila Chernova); 19chsk56@gmail.com (Serhii Chernov); iryna.zhuravel@nuos.edu.ua (Irina Zhuravel)
ORCID: 0000-0001-5191-0272 (Liubava Chernova); 0000-0001-8040-5590 (Anna Zhuravel); 0000-0002-0666-0742 (Lyudmila Chernova); 0000-0002-9069-0409(Serhii Chernov); 0000-0002-3747-4387(Irina Zhuravel)

- selection of the most significant areas of setting tasks during the design of complex IT projects;
- analysis of the specifics of the IT project development process, the advantages and disadvantages of existing design methods and the formation of a practically applicable approach to the design of complex IT projects.

2. Directions of cognitive research

The direction of cognitive research is broad and encompasses various disciplines, including psychology, neuroscience, linguistics, computer science, and philosophy. The primary goal of cognitive research is to understand how the human mind works, including how we perceive, think, learn, remember, and communicate. Cognitive research can be further broken down into several sub-disciplines, including cognitive psychology, cognitive neuroscience, cognitive linguistics, and computational cognitive science [1, 7, 9].

Cognitive psychology focuses on studying mental processes such as attention, perception, memory, and problem-solving. Cognitive neuroscience aims to understand the neural basis of cognition and how the brain supports mental processes. Cognitive linguistics examines the relationship between language and thought, while computational cognitive science uses computational models to simulate cognitive processes and test theories.

In recent years, there has been growing interest in the application of cognitive research to areas such as artificial intelligence, human-computer interaction, education, and healthcare. By better understanding how the mind works, cognitive research has the potential to improve the design of technologies and interventions that can enhance human cognition and improve quality of life.

The cognitive topic has been at the centre of scientists'[11, 12] attention for a long time, and in recent years even a new field of knowledge has appeared - "cognitive informatics", which most fully corresponds to scientific ideas about the relationship and interdependence of human mental activity and the information process. First, a significant part of research is devoted to the study of mental mechanisms of information perception and processing and making decisions on this basis. Secondly, the results of works in the field of artificial intelligence are used in terms of structural analysis of text, audio and visual information, as well as pattern recognition.

Thirdly, research is conducted in the field of (automated) intelligent processing of large arrays of unstructured data [3]. Efforts here are focused on the development of software tools that simulate human mental activity and allow the detection of implicit functional dependencies and non-obvious trends based on the analysis of large information arrays in favour of the development of objective and long-term forecasts, as well as making adequate decisions on this basis. At the same time, scientists and researchers still ignore the cognitive aspects of designing in general and designing and creating IT projects in particular. Although it is IT projects at the current stage of development that is the main tool for decision support, the quality and efficiency of making decisions largely depend on the effectiveness of their work. The development of modern complex and multi-functional IT projects, for example, for monitoring and evaluating the activities of an entire industry, is a multifaceted, multi-stage and long-term process that involves the interaction of various structural divisions of the work executor and, accordingly, many analysts, developers and administrators, often functionally separated and spatially distributed [4]. And the cognitive aspect plays not the last role in this work.

Cognitive modelling of IT projects is a field of research that uses computational models to simulate human cognitive processes in the context of information technology (IT) projects. The aim of cognitive modelling in this context is to better understand how people perceive, reason, and make decisions about IT project-related tasks and problems.

Cognitive modelling of IT projects involves developing mathematical and computational models that simulate human cognitive processes such as attention, perception, memory, learning, and decision-making. These models can be used to identify potential bottlenecks or inefficiencies in IT project processes, predict the outcomes of different project management strategies, and optimize the design of user interfaces and software features.

The development of cognitive models for IT projects involves collecting data through various methods, such as observation, interviews, surveys, and experiments. This data is then used to calibrate and validate the models. Once developed, cognitive models can be used to test and refine theories of human cognition, and to identify new areas for research and development in the field of IT project management.

Thus, the study of the role and significance of the cognitive component and its accounting in the process of designing an IT project, even in terms of formulating the problem and setting the task, is a new, unexplored and promising issue.

3. Modelling stages and connections between objects

Modelling during the creation of IT projects has some features compared to other areas. When designing an IT model, first of all, it is an image of an already existing entity (both material and virtual) and it reflects this entity in terms of automating the processes occurring in it through the introduction of IT. In this case, the model of the subject area serves as the basis for developing a model, prototype, and future project.

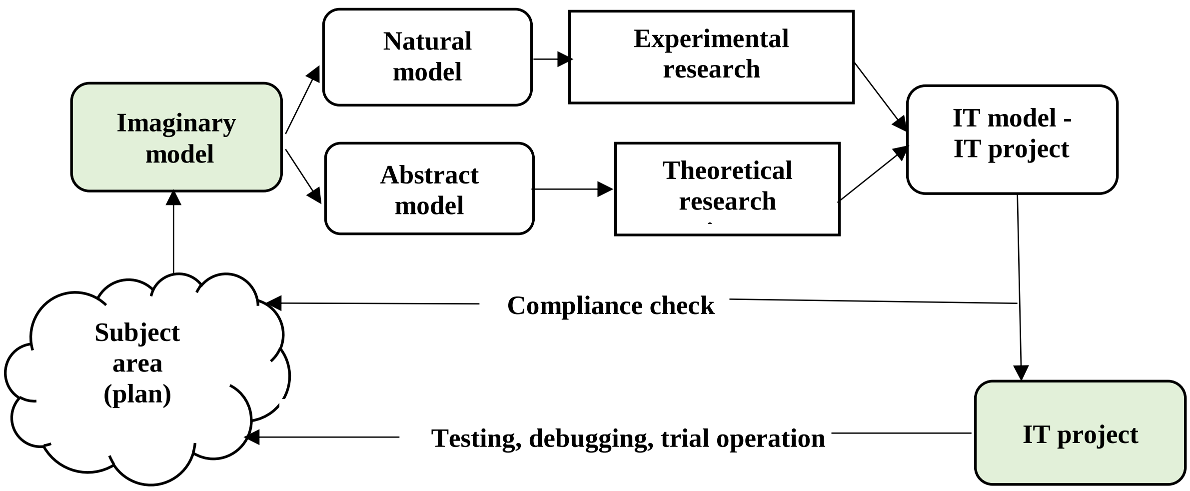

Figure 1: Logical scheme of building an IT project

Proceeding from this specificity and based on theoretical studies in the field of application of models and modelling, the logic of designing, developing and creating an IT project can be presented in the form of a sequence of actions depicted in Figure 1.

The logical scheme of designing an IT project, shown in Figure 1, follows that with the results of experimental or theoretical studies of the subject area reflected in its model, the IT model itself is developed. Then the compliance of the project with the goals and objectives of the design, and the specifics of the subject area are checked. Based on this check, the imaginary model is refined, and the resulting changes are implemented in a full-scale or abstract model with further adjustment of the project.

The described modelling logic dictates the expediency of the modelling stages [6]. At the stage of meaningful formulation of the problem, goals are formulated and modelling tasks are set. At the same time, the objects related to the tasks being solved, as well as the situation that needs to be implemented as a result of its solution, are defined.

To be able to describe the problem quantitatively and to use computer technology in solving it, it is necessary to make a qualitative and quantitative analysis of the objects and situations related to it. At the same time, the decomposition of complex objects into components (elements) is carried out, and the relationships of these elements, their properties and quantitative and qualitative values of these properties, and quantitative and logical relationships between them, expressed in the form of equations and inequalities, are determined. As a result of solving this problem of system analysis, the object is presented in the form of a system [10].

In the process of the mathematical formulation of the problem, a mathematical model of the object is built and methods (algorithms) are determined for obtaining a solution to the problem. At this stage, it may turn out that the previously conducted system analysis has led to such a set of elements, properties and relationships for which there is no acceptable method of solving the problem, and as a result, it is necessary to return to the stage of system analysis. As a rule, the problems solved in practice are standardized, and the system analysis is carried out based on a known mathematical model and the algorithm for its solution, the problem is only in choosing the appropriate method [11].

For complex objects consisting of a large number of elements and having a large number of properties, it may be necessary to use specialized packages for information modelling that provide database maintenance, tools for working with it, and methods of extracting data necessary for calculations.

For standard tasks, an appropriate package of application programs and database management systems can be used. At the final stage, the experimental operation of the model is carried out, and based on the analysis of the obtained results, the final version of the IT project is implemented.

If a deep decomposition of the subject area is carried out, as a result of which many objects are formed, the relationships between different classes and instances of objects are reflected in information communication models. There is a specific name in the model for each connection. In graphical form, a relationship is represented as a line with arrows between the objects that are connected and is indicated by a relationship identifier (for example, an entity-relationship diagram).

There are three types of connection: a one-to-one connection is implemented when one instance of one object is connected to a single instance of another; a one-to-many relationship occurs when one instance of the first object is associated with one or more instances of the second object, but each instance of the second object is associated with only one instance of the first; a many-to-many relationship exists when one instance of the first object is associated with one or more instances of the second and each instance of the second is associated with one or many instances of the first.

Connections are divided into unconditional and conditional. In an unconditional connection, each instance of the object is required to participate in it. Not all instances of the object participate in conditional connection. The connection can be conditional both on one side and on both sides.

All connections in the information model need a description, which at least includes:

- connection identifier;
- formulation of the essence of connection;
- type of connection (its multiplicity and convention);
- a way to describe a relationship using auxiliary attributes of objects.

The most famous structure united by a unidirectional connection is a queue. Possible generalizations of information models are cyclic structure tables.

A very important role is played by the tree structure information model [12], which is one of the most common types of classification structures. This model is built based on a relationship that reflects the relationship of a part to a whole, that is, a relationship of the "one-to-many" type. Thus, data types in programming are closely related to certain informational data models.

An even more general information model is the so-called graph structures, which are the basis for solving many information modelling tasks.

The experience of creating complex IT projects designed to solve interdisciplinary tasks and management tasks at the strategic level revealed several problems that complicate the effective design and development of such projects. First of all, it is about a high level of fragmentation and inconsistency of information related to this activity between experts, developers and administrators. In the scientific literature, the degree of such consistency is called the level of cognitive interoperability.

By definition, the problem of cognitive interoperability can be overcome by a carefully developed model of the subject area, which should provide for all project participants, regardless of their departmental affiliation and spatial location, a coherent and unambiguous idea of the subject area itself, of the goals and tasks of design, as well as of the functions assigned on an IT project.

As the most effective way to overcome this problem, it is proposed to introduce a section of model development regulations of the subject area and design, devoted to the issue of harmonizing individual parts of the subject area model and the project. In this section, the following important elements of interaction between all interested units and design participants should be provided:

- issues subject to the mandatory agreement;
- procedure for changing (shortening, expanding) the list of issues subject to the agreement;
- forms of documents, transferred in the order of agreement between the design participants, which provide for a possible full and unambiguous description of the essence of the issues subject to the agreement;
- strict and specific terms of the agreement, including the possibility of agreement "by default" in the case the response to the request was not received within the set deadline;
- the procedure for resolving disputed issues in the case it is impossible to reach an agreed solution between the design participants.

4. IT projects development for management in complex subject areas

IT projects development for management in complex subject areas is a specialized field that involves developing software systems and technologies to support decision-making and management in complex domains, such as healthcare, finance, and transportation. These subject areas are characterized by large volumes of data, high levels of uncertainty, and multiple stakeholders with conflicting interests and priorities.

IT projects development for management in complex subject areas requires a deep understanding of the domain, including its processes, regulations, and stakeholders. This understanding is used to develop software systems and technologies that can help users manage complex data and make informed decisions.

These systems and technologies may include data analytics and visualization tools, decision support systems, and simulation models. They may also incorporate machine learning algorithms and artificial intelligence to help users make sense of complex data and identify patterns and insights.

Developing IT projects for management in complex subject areas involves working closely with domain experts, stakeholders, and end-users to ensure that the technology meets their needs and is usable and effective. It also requires a strong focus on data security and privacy, as well as adherence to relevant regulations and industry standards.

The main problems of software engineering are the acceleration of changes and, accordingly, the need for quick adaptation to new conditions, ensuring, on the one hand, a high degree of flexibility and the ability to adjust the created systems to the rapidly changing needs of various categories of users and operating conditions, and on the other hand – functional reliability of the IT projects under development.

Since large IT projects cover all areas of enterprise activity [13], a necessary condition for creating effective management systems, increasing the efficiency of their adaptation, and reducing the complexity of support is the need to involve specialists in the development and modification of models in various subject areas.

Also, when creating complex IT projects, there is often a need to develop families of models that describe different aspects of the system's functioning and/or to implement multi-level models that describe the system with different degrees of abstraction. Among the many existing IT project design methodologies, three main approaches can be distinguished:

- cascade approach;
- the spiral approach;
- iterative approach.

Cascade development, or the so-called waterfall model, looks like a kind of flow of work that successively goes through the following stages:

- definition of requirements for the IT project being created;
- creation of a future project, during which project documentation is created, describing methods and a plan for implementing certain of the above requirements;
- implementation, that is, the actual creation of an IT project as a hardware and software complex;

- testing and debugging of the IT project — at this stage, to a certain extent, the shortcomings that appeared at the previous stages of development can be eliminated, but the elimination of errors in the requirements of the IT project is quite problematic;
- the implementation of the IT project and its further support, which is carried out during operation, is mainly aimed at eliminating errors and adding new functionality.

The cascade approach in its pure form assumes that the transition to the next stage occurs after the full and successful completion of the previous one, i.e. return to previous stages or their parallel execution is not envisaged. When managing large projects, the implementation of project stages in a strict sequence allows you to radically reduce many project risks and make its implementation more transparent and also provides an opportunity to assess the quality of the product at each stage. However, this leads to the lack of flexibility of such an approach and the declaration of formal project management as an end in itself, often to the detriment of terms, cost or quality. To eliminate these shortcomings, several different modifications of the cascade approach have been developed, which mainly consist in providing the possibility of returning to previous stages, but all of them, however, are associated with a significant increase in material and time costs for development.

The spiral life cycle model was proposed to overcome these problems. At the stages of analysis and design, the feasibility of technical solutions and the degree of satisfaction with the customer's needs are checked by creating prototypes [15]. Each turn of the spiral corresponds to the creation of a workable fragment or version of the system. This allows you to clarify the requirements, goals and characteristics of the project, determine the quality of the development, and plan the work of the next turn of the spiral. In this way, the details of the project are deepened and successively specified, and as a result, a reasonable option is selected that satisfies the actual requirements of the customer and is brought to implementation. The main problem of the spiral cycle is determining the moment of transition to the next stage. To solve it, time constraints are introduced for each of the stages of the life cycle, and the transition is carried out according to the plan, even if not all the planned work is finished. Planning is carried out based on statistical data obtained in previous projects and the personal experience of developers.

Iterative technology reflects the objectively existing spiral cycle of creating complex systems and involves the execution of each of the above stages in the mode of constant analysis of the obtained results and correction of previous stages of work. An IT project created in this way goes through a repeating PDCA cycle (plan-do-check-act) at each stage. In the course of development, existing or additional requirements and restrictions related to adopted technical solutions may be changed. Only during iterative development, they can be taken into account to the fullest extent, because only with this approach all project participants are fully ready for changes. Varieties of the cascade approach are V-model, spiral model, and prototyping.

The advantages of the iterative approach are:

- continuous iterative testing, early detection of conflicts between requirements, models and project implementation, reducing the impact of design in the early stages of the project, which leads to minimizing the costs of their elimination;
- effective feedback between project participants, in particular, the project team and the customer of the IT project, which ensures maximum satisfaction of his needs, effective use of accumulated experience;
- the ability to assess the degree of project implementation and, as a result, the great confidence of customers and executors in its successful completion, the ability to focus efforts on the most critical areas of the project;
- more uniform loading of project participants, and more uniform and predictable distribution of costs throughout the entire period of IT project creation.

The main disadvantage of this approach is the difficulty of formal and progressive project management, and the lack of fully completed stages almost till the end of the development period, which makes it difficult to distribute the project budget in detail between the various stages and does not allow to report about the completion of each of them before the end of the entire work process. This requirement is not so essential in business structures, since all work can often be done by one independent team of performers, and the customer can afford not to require the formation of interim reporting. In the field of government orders, in conditions of strict control over finances, this requirement can become significant, besides, such projects often involve many independent performing teams, which requires a clear distribution of tasks and financial resources among them. So, due to its advantages, the iterative approach is now the most common.

Turning to the issues of assessment and monitoring of the state of affairs in complex subject areas, it should be noted that these subject areas are often far from fully studied or studied in a fragmentary manner. The consequence of this is the lack of sufficiently complete, but at the same time generally accepted and practically useful ontologies (conceptual schemes) and information models of objects and processes, which provide an informational description of these subject areas in the created IT project. Moreover, there may also be a practical lack of obvious or sufficiently elaborated methods and models of the very process of monitoring a complex program-target activity (as well as the process of managing it), which take into account cause-and-effect dependencies and ensure the correct setting of goals and tasks of monitoring and management. The lack of such ontologies, models, and methods necessitates the use of cognitive modelling methods, which, are embedded in the IT project formation algorithm described below.

Developing a high-quality IT project for a complex subject area requires a comprehensive and well-planned approach. The following are some key steps that can be taken to ensure the success of such a project:

- requirements Gathering. Work closely with domain experts and stakeholders to gather requirements and define the scope of the project. This includes understanding the business processes, regulatory requirements, and user needs;
- design. Based on the requirements, design the architecture of the system and the software components. This should include the data model, algorithms, and user interface;
- development. Develop the software components using an appropriate programming language and frameworks. This should involve creating a codebase that is maintainable, scalable, and secure;
- testing. Test the software components using appropriate test cases and procedures. This includes unit testing, integration testing, and user acceptance testing;
- deployment. Deploy the software components in a test environment and then in a production environment. Ensure that the system is stable, reliable, and performs well;
- maintenance and Support. Provide ongoing maintenance and support for the system, including bug fixes, enhancements, and user support;
- documentation. Document the system architecture, data model, algorithms, and user interface. This documentation should be comprehensive, up-to-date, and easily accessible;
- training. Provide training to end-users, system administrators, and support personnel. This should include training on how to use the system, as well as how to troubleshoot and resolve common issues.

Requirements gathering is the process of collecting and documenting the needs, expectations, and constraints of stakeholders for a software system or IT project. It is a critical step in the development process as it helps ensure that the system meets the needs of its intended users and stakeholders.

Effective requirements gathering involves the following steps:

1. Identify stakeholders. Identify all stakeholders who will be affected by the system, including end-users, business owners, subject matter experts, and IT staff.
2. Determine requirements elicitation techniques. Choose appropriate techniques to gather requirements, such as interviews, surveys, focus groups, workshops, or prototyping.
3. Gather requirements. Use the chosen techniques to collect requirements from stakeholders, and document them in a clear, concise, and unambiguous way.
4. Analyze requirements. Analyze the requirements to identify any conflicts, inconsistencies, or missing information, and ensure that they are complete and feasible.
5. Prioritize requirements. Prioritize the requirements based on their importance, feasibility, and impact on the system's success.
6. Validate requirements. Validate the requirements with stakeholders to ensure that they accurately reflect their needs, and make any necessary adjustments based on feedback.
7. Manage requirements. Manage the requirements throughout the development process, including tracking changes, documenting decisions, and communicating them to stakeholders.

By following these steps, the requirements-gathering process can help ensure that the final IT project meets the needs of its intended users and stakeholders and is a success.

The key to developing a high-quality IT project for a complex subject area is to have a clear understanding of the requirements and user needs and to follow a well-defined process for design, development, testing, deployment, maintenance, and support.

During the development phase of an IT project, it is important to choose an appropriate development methodology or approach that suits the project's requirements. This can involve selecting from various methodologies, such as Agile, Waterfall, or DevOps. Each methodology has its own set of principles and practices, and the choice of methodology will depend on various factors, including the project's scope, timeline, and team structure.

Agile methodology is a flexible and iterative approach that emphasizes collaboration, feedback, and adaptability. It involves breaking down the project into smaller, more manageable parts or sprints, with each sprint delivering a working software component that is reviewed and tested by the team and stakeholders.

Waterfall methodology, on the other hand, is a more traditional, sequential approach that involves a linear progression of project stages, such as requirements gathering, design, development, testing, and deployment. Each stage is completed before moving on to the next, with little room for iteration or changes.

DevOps methodology is a combination of development and operations that emphasizes collaboration and communication between developers and IT operations staff. It involves continuous integration and delivery of software components, with a focus on automation and monitoring.

Ultimately, the choice of methodology will depend on the project's requirements, timeline, and team structure, and may involve a combination of different methodologies depending on the project's specific needs.

In connection with the above, the development of a high-quality IT project of a complex subject area on the fly using the cascade design method is difficult, if not impossible, and requires the use of iterative methods. On the other hand, the iterative approach in its pure form (described above) has a rather general character and is not easily formalized in terms of management and budgeting. As a result, it is necessary to specify it concerning both the main task of monitoring a complex subject area and the possible task of managing it (or its subjects that have appropriate control levers). In this regard, the application of a hybrid-cascade-cyclic algorithm for the formation of a monitoring (or monitoring-management) IT project for complex subject areas is envisaged.

Figure 2 presents a block diagram of this algorithm, designed for monitoring and evaluation of complex and multidisciplinary subject areas.

Presented in Figure 2 blocks have different natures:

1. The "Real subject area" block is objects and processes (physical, social, etc.), which are the subject of monitoring and management.
2. The blocks "Goals and tasks of the IT project", "Ontology and information model of the subject area" and "Monitoring and management algorithms" are, in fact, a set of project documentation, which should formally reflect:

- the first, is the goals and tasks of the monitoring (and management) process, which are simultaneously the goals and tasks of the corresponding IT project;
- in the second – an ontology (or conceptual scheme) and a specific information model of objects and processes of the subject field, implemented at least in the part that is necessary and sufficient for the goals and tasks at hand; in the third – algorithms of assessment and monitoring processes corresponding to the goals and objectives, as well as algorithms of controlling influences, if such functionality in the form of a management subsystem is envisaged for the IT project.

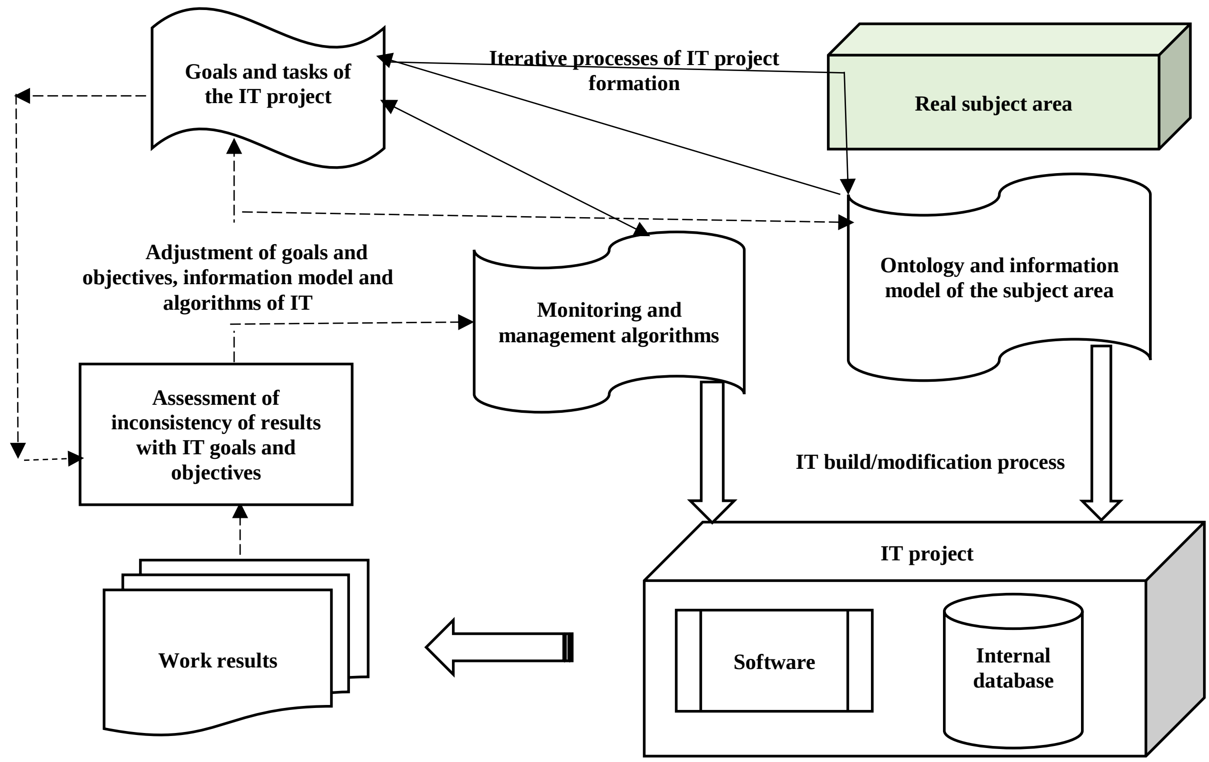

Figure 2: Algorithm of the cascade-cyclic process of forming an IT project

3. The "IT project" block is a hardware and software embodiment of the documents listed above, which includes a database built based on an information model, and software modules that implement monitoring (management) algorithms.

4. The "Work Results" block are formal documents generated as a result of the practical work of the IT project – statistical and analytical reports on the state and dynamics of the subject of monitoring and management.

5. The block "Assessment of inconsistency" is also a document (report) that reflects the degree of achievement of monitoring and management goals and objectives, formally – discrepancies between the necessary or planned values and the achieved evaluations of indicators – the results of monitoring program-target activities.

Let us consider the structure of these discrepancies.

1. The iterative process of forming an IT project (solid arrows). Since we are talking about complex subject areas, the creation of high-quality design documentation right away, using the waterfall approach, for the reasons stated above, is difficult, even if not impossible. In this regard, it is expected to use an iterative cyclical approach already at this design stage in the following general sequence:

- the goals and objectives of monitoring are preliminarily determined;
- following the goals and objectives, the subject area is studied to identify objects/processes in it, and establish cause-and-effect relationships between objects and processes, as a result of which an information model of this subject area is built/refined (the terminology is defined and a hierarchical structure of objects and processes is formed);
- an analysis of the obtained information model is carried out, based on which the goals and objectives of the IT project are specified, i.e., a transition to the previous paragraphs is carried out. A similar process is carried out for the selection of algorithms for the operation of information systems, however, the algorithms have usually a more abstract nature, and their methodology is well-developed, which is why the phase of accessing the corresponding subject area is generally not needed (therefore, this cycle is shown in figure 2 by one double-sided arrow).

2. After a sufficient study of the IT project, the process of its creation (or modification of an existing one) starts. This process, shown in Figure 2 with the solid arrows, includes the development of technical documentation, viz. programming and deployment of software on hardware platforms, filling with information about the IT project and its commissioning.

3. The main result of the functioning of the IT project is the periodic creation of information reports on the state of the monitoring subject. The process of creating reports is shown in Figure 2 with a dashed arrow. The basis of such reports are tables, which can be supplemented by timetables or textual explanations.

4. Based on the generated monitoring reports, secondary reports are formed. Discrepancy tables reflect the differences between the planned and actual values of the indicators of the monitoring subject. In general, the reasons that some of the results (indicator values) are not achieved can be:

- incorrect setting of goals and objectives of management;
- the imperfection of the management system;
- an incorrect approach to evaluate or monitor the results.

Thus, based on the value of the discrepancy, a conclusion about the quality of the monitoring system can be drawn, but at the same time, the influence of the imperfection of setting goals and objectives[16], as well as the system for managing the monitoring subject, must be excluded. The process of separating discrepancy values according to the presented factors above and the process of assessing the degree of influence of each of them today are difficult to be formalized, and therefore the main tool is the method of expert evaluation.

5. Based on the analysis of discrepancies, corrective actions are developed, which are exerted, firstly, on the formulation of IT goals and objectives, and secondly, on further refinement of the information model and algorithms for monitoring and evaluation (as well as management), with the subsequent implementation of all these changes at the hardware-software IT level.

The presented blocks and processes form a "big" cycle of a cascade-cyclic (iterative) algorithm for the formation of an IT project, designed for monitoring and evaluation of complex subject areas.

5. Conclusion

The process of information modelling should ensure the adequacy of the created model to the goals and tasks of the IT project, the economic feasibility of its implementation, the flexibility and scalability of the model and the IT project as a whole in the conditions of changing goals and tasks of functioning. For this, the traditional cycle of information modelling and the specifics of the models were considered, on the example of which parallels were drawn between the classes and features of the models and the real practical problems solved on their basis. On this basis, approaches to understanding the essence and features of modelling when creating an IT project are outlined. Further improvement of information modelling is associated with the development of ideas about relationships, structures and tasks that can be solved based on these relationships and structures.

As the most effective way to overcome the problem of an ambiguous understanding of various aspects of design, which significantly affect the development process and the functioning of the IT project itself, it is proposed to introduce into the regulations of the development of the model of the subject area and the creation of a section devoted to the issue of harmonizing individual parts of the model of the subject area and the project.

It was shown that it is difficult to create complex high-level IT projects using traditional cascade or cyclic development methods in their pure form. One of the reasons for this is the absence or insufficient development of relevant conceptual schemes and information models.

On the other hand, in the conditions of a program-targeted approach to the development and creation of similar IT projects, especially for short-term planning, the most suitable is a cascade approach, which not only reduces the quality of the considered algorithm of cascade-cyclic design, designed to eliminate the listed problems and ensure quality and viability IT project in terms of short-term planning. The presented scheme allows you to understand the role and place of each design method, as well as the expediency of their application and combination in different conditions and circumstances.

Thus, the formulated goal of the research was achieved, and the set tasks were solved. In addition, the work presents a schematic algorithm for the cascade-cyclic design of complex IT projects, which takes into account the cognitive features of this type of activity.

6. References

[1] R. Axelrod Schema theory: An information processing model of perception and cognition. The American Political Science Review 67(4) (1973) 1248–1266.

[2] S. Bushuyev, D. Bushuyev, N. Rusan, Emotional intelligence-the driver of the development of breakthrough competencies of the project, in: Proceedings of the 12th International Scientific and Technical Conference on Computer Sciences and Information Technologies, 2017, 2, pp. 1–6.

[3] S. Bushuyev, D. Bushuiev, V. Bushuieva, Interaction Multilayer model of Emotional Infection with the Earn Value Method in the Project Management Process, in: Proceedings of the International Scientific and Technical Conference on Computer Sciences and Information Technologies, 2020, 2, pp. 146–150.

[4] Liubava Chernova, Lyudmyla Chernova, Natalia Kunanets, Anna Zhuravel, Serhii Chernov, Olga Artemenko, Application of the cognitive approach in the field of IT project management, in: Proceedings of the 17th IEEE International Conference on Computer Science and Information Technologies, 2022, pp. 426–429.

[5] Structure of decision: The cognitive maps of political elites. Princeton, NJ, USA: Princeton University Press, 1976. p.404.

[6] S. Titov, L. Chernova, N. Kunanets, Lb. Chernova, E. Nedelko, S. Chernov, The algorithm of selecting candidates for IT projects based on the simplex method, in: Proceedings of the International Conference on Advanced Trends in Information Theory (ATIT 2020), 2020, pp. 221–232.

[7] R. Axelrod, W. D. Hamilton, The evolution of cooperation. Science 211(4489) (1981) 1390–1396.

[8] A. Voitushenko, S. Bushuyev, Development of project managers' creative potential: Determination of components and results of research. Advances in Intelligent Systems and Computing (2020) 1080 283–292.

[9] J. R. Anderson, C. F. Boyle, A. T. Corbett, M. W. Lewis, Cognitive modeling and intelligent tutoring. Artificial Intelligence 42(1) (1990) 7–49.

[10] S. Bushuyev, J. Babayev, D. Bushuiev, B. Kozyr, Emotional Infection of Management Innovation SMART Government Projects, in: IEEE European Technology and Engineering Management Summit, E-TEMS 2020. URL: https://ieeexplore.ieee.org/document/9111796.

[11] J. Funge, X. Tu, D. Terzopoulos, Cognitive modeling: Knowledge, reasoning and planning for intelligent characters, in: Proceedings of the 26th Annual Conference on Computer Graphics and Interactive Techniques, 1999. pp. 29–38.

[12] G. Jakobson, J.Buford, L. Lewis, A framework of cognitive situation modeling and recognition, in: IEEE Military Communications Conference (MILCOM-2006), 2006. pp. 1–7.

[13] F. Mizoguchi, H. Nishiyama, H. Iwasaki, A new approach to detecting distracted car drivers using eye-movement data, in: Proceedings of the 13th IEEE Conference (International) on Cognitive Informatics and Cognitive Computing, 2014. pp. 266–272.

[14] Session 3(B): Artificial Intelligence (AI), in: Proceedings of the 1st IEEE Conference (International) on Cognitive Informatics Proceedings. Calgary, Alberth, Canada, 2002, pp. 295–328.

[15] A. Bondar, S. Bushuyev, S. Onyshchenko, H. Tanaka, Entropy paradigm of project-oriented organizations management. CEUR Workshop Proceedings 2565 (2020) 233–243.

[16] S. Bushuyev, S. Onyshchenko, N. Bushuyeva, A. Bondar, Modelling projects portfolio structure dynamics of the organization development with a resistance of information entropy, in: International Scientific and Technical Conference on Computer Sciences and Information Technologies, 2021, 2, pp. 293–298.