



The theory of corporate information systems

Dmitry Yu. Stepanov

Department of Corporate Information Systems, MIREA – Russian Technological University
78, Vernadsky prosp., Moscow, 119454, Russia
E-mail: mail@stepanovd.com

Abstract The paper introduces a new discipline that combines the design of information systems, databases, programming, as well as project and change management, called the theory of corporate information systems. The content of each area correlates with the levels of software implementation: processes, data, application, technology as well as the project and changes. The theory assumes the formalization of strategies for analysis, design, roles and authorization, technical and business cutover, change management, build, data migration, training, testing and post go-live support, each of which is decomposed into subtasks, consistent with other activities, and then planned for execution. Planning plays a key role during the implementation of corporate information systems, since most of the project tasks involve receiving feedback from future users of the system. Thus, the proposed theory combines the existing technical disciplines and expands the scope of their application to larger-scale IT projects.

Keywords ERP-systems, Corporate Information Systems, Implementation.

1. Introduction

It is impossible to imagine our today's life without the use of modern information technologies. Smartphones and voice assistants, the Internet of Things and data mining, online cinemas and virtual reality [1]. The situation is similar in the workplace: we are surrounded by the industrial Internet of Things, robotics and predictive analytics, as well as many various automation tools [2]. Thus, digitalization affects almost all areas of our life. The unifying link in all these technologies is the need for analysis, development and implementation of software. The latter, depending on the application area and the scale of implementation, are characterized by the generally accepted terms information systems and corporate information systems [3]. Despite the fact that both definitions, in principle, say the same thing, the process of implementing these systems varies greatly. Ways, methods and approaches for deployment of information systems by default are trying to be applied to implement a more complex corporate information systems, which is not always applicable, rational and effective [4]. The practice of implementing software solutions related to ERP, ERP2 or ERP3 standards [5-7] confirms the existing knowledge gap in the theory of deployment of corporate information systems, which will be discussed in this paper.

Everyone who is interested in the field of information systems and technologies has the opportunity to easily find literature to their liking. Here we can highlight the works devoted to software development [8-9], information systems design [11-13, 3], as well as project management for the implementation of a similar class of software systems [14, 15]. The works [8, 9] are rightfully considered classics, thoroughly describing the ways, methods and algorithms of building software development. However, the issues of the implementation of big projects are practically not covered here. A greater emphasis on the design of software systems is made in [11-13, 3], where implementation methodologies, methods of modeling business processes and organizational structures, as well as typical realization problems are demonstrated. In these works, information and corporate systems are represented by synonyms. Finally, the monographs [14, 15] provide a foundation for managing any implementation projects, regardless of their subject orientation. Summarizing all these, we can say following: information and corporate systems are considered in most literature sources as software products of equal scale both in terms of their realization and implementation.

The purpose of this paper is to propose a theory of corporate information systems that demonstrates the distinctive features of the analysis, development and deployment of corporate systems as opposed to information

systems. Consideration of the differences will allow us to approach the implementation of corporate information systems more rationally and ensure a more likely success of the project.

2. Computer programs, information systems and corporate information systems

Depending on the coverage of the business processes of the enterprise by the programs, several types of software can be distinguished (Fig.1). The first of which is a minimal unit: a computer program. A computer program is a set of data and ordered commands designed to obtain a certain result by a computer [16]. Application development requires structural thinking, which makes it possible to present the logic of the program in the form of a sequence of operations, i.e. an algorithm, on information. Thus, this class of software almost always involves data processing and consists of three elements:

- ordered commands;
- structured data;
- algorithms.

It should be clarified, despite the fact that data is mentioned in the definition of a computer program, they are mainly having a secondary role: considered as input parameters. Examples of this class of programs are applications ABBYY Lingvo, FineReader, etc. Thus, computer programs allow you to solve single application tasks only.

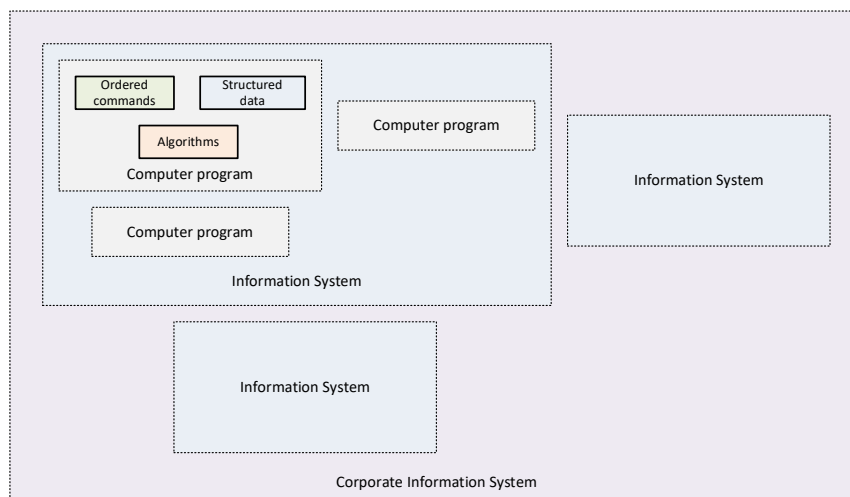


Fig. 1. Differences between computer programs, information systems and corporate information systems.

A set of software that defines the only problem domain and ensure its automation is called an information system (hereinafter – IS) [11]. Quite often in literary, IS is understood as a communication system for collecting, transmitting and processing information about an object that supplies employees with the data necessary to implement the management function [3]. Examples of such systems include 1C: Accounting and BOSS-HR, covering, respectively, the financial and human processes of the enterprise, which will be called automation standards later. The classical approach to the implementation of projects for such class of software implies execution of tasks in the following areas [3]:

- processes;
- application;
- data;
- technology,

which in [17] are called implementation levels. At the process level, the organizational structure and business processes of the enterprise are designed in AS-IS and TO-BE models, which allows a more balanced approach to identify requirements and bottlenecks in the company's work. The determined business processes are implemented in the IS at the application level, which requires software development for particular problem area.

The processes reflected in the IS are continuously connected with the creation, modification and deletion of data, for which corresponding level is allocated. The data level within the IS is about tasks of data control and normalization, which requires the deployment of database management systems (hereinafter – DBMS). And, finally, the technical level at which the developed IS and implemented DBMS «live» and successfully functioning. Using levels of software implementation, it is possible to formulate following statement characterizing the IS.

Statement 1. In information systems implementation projects, much attention is paid both to the development of software that automates a certain subject area of the company, and to processes, data and technology, without which successful deployment of the solution is impossible. Thus, increasing the number of programs to be developed leads increasing additional project tasks indirectly related to software.

The development of information systems provides automation of the only subject area of the company. And what happens if we try to process several problem areas at once? In this case, we are talking about corporate information systems (hereinafter – CIS). In [18], the corporate information system is defined as a set of methods and solutions for creating a unified information space for management and ensuring the company's activities. In short, the CIS is a set of IS combined together [17]. Examples of such systems are software solutions SAP ERP, Oracle EBS, 1C: ERP, etc. Then it is natural that corporate systems have the same implementation levels that are used in IS projects, is not it? This is true, however with a few comments: an increase in the number of IS creates the need for better management of project activities on the part of both the supplier and the customer, moreover data priorities are also changing – more attention is paid not to control, but to migration [19].

Statement 2. The implementation levels of corporate systems in comparison with information systems are complemented by the tasks of project management and changes, moreover, the data level is used mainly to the issues of migrating information from legacy to the target software system.

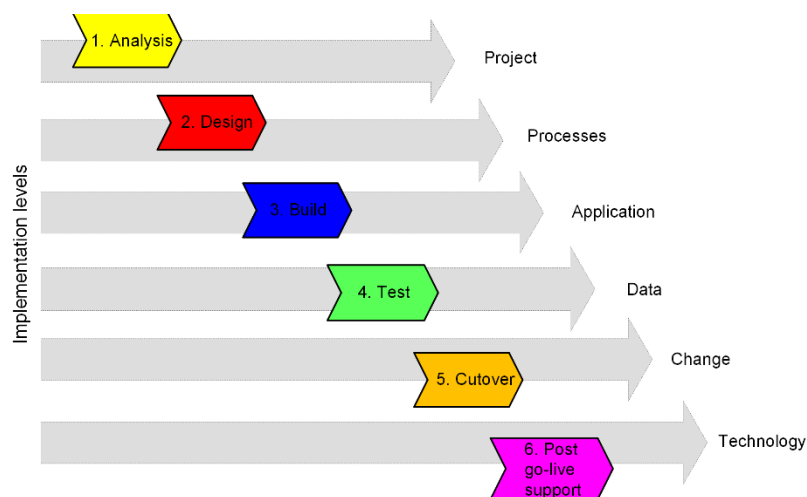


Fig. 2. Implementation levels of corporate information systems.

The change level makes it possible to ensure a smooth transition from the previous software solution to the one being developed, to prepare the company and its employees for the use of a new software, as well as to consolidate both technical and organizational changes in order to obtain significant advantages over competitors [20]. At the project level, planning, execution, control and adjustment of activities of all other levels are carried out, which is mainly regulated by the international project management body of knowledge (hereinafter – PMBoK) [15]. Figure 2 clearly demonstrates all six implementation levels of corporate information systems.

3. The theory of corporate information systems

Unlike user programs and IS, CIS implementation projects are characterized by a long duration and significant efforts due to the following distinctive features:

- mandatory adherence to the approved methodology for the implementation of software products, which sets the order and composition of tasks;

- planning of all works, as well as their further control and correction;
- the relationship of almost all activities performed and the need for their regular comparison and alignment;
- big volume of the tasks being implemented and the mass involvement of technical and business specialists;
- formalization of all performed tasks, as well as documentation of the results,

where the first four points correspond to the project level, and the last one is relevant to all other implementation levels.

Let's consider the development and implementation of CIS, starting from the project level. The fundamental standard regulating the list of tasks at this level is PMBoK [15], which identifies ten control parameters shown in Figure 3. Since PMBoK is used for all kinds of projects, not limited only by implementation of software, we will clarify it for the purposes of CIS, then:

- integration will mean the charter of the project, which sets, among other things, the implementation methodology for CIS;
- resources and cost will determine the resource plan and the project cost plan calculated on its basis;
- timeline will set the project schedule aligned with the resource plan.

The meaning of the parameters of stakeholders, communications, risks, quality and supplies stay unchanged. The remaining parameter characterizing the content requires more in-depth study to adapt it to the CIS deployment projects.

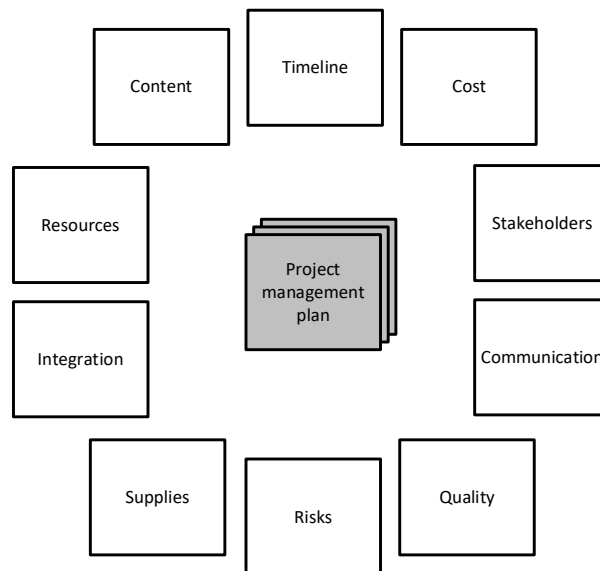


Fig. 3. Parameters of the project level.

Despite the fact that statement 2 introduces additional groups of tasks that distinguish CIS implementation projects from IS, their description remains unclear. Having analyzed the typical difficulties that technical specialists face in practice when implementing ERP-systems [17, 21], it is possible to identify extra activities inherent in CIS projects: requirements analysis, roles and authorization, business cutover and post go-live support. We have already noted earlier that CIS implementation projects are characterized by the formalization of all tasks performed. In practice, this means that any activity executed is a planned work related to a specific strategy for delivering CIS project. Now we can define the project parameter responsible for the content with the following statement.

Statement 3. The content of the CIS implementation project is set by the strategies of analysis, design, roles and authorization, changes, build, technical cutover, migration, training, testing, business cutover and post go-live support related to the implementation levels of processes, applications, data, technology and changes (Fig. 4).

Statement 3 allows us to clarify the content parameter related to the project level during the implementation of CIS (Fig. 3). Thus, we have all the information to summarize the distinctive features of the implementation of corporate information systems.

Definition 1. *The theory of corporate information systems is an interdisciplinary field of knowledge describing deployment of corporate information systems based on implementation levels presented by processes, applications, data, technology, as well as project and changes, which in turn are characterized by strategies for analyzing requirements, designing processes and organizational structures, maintaining roles and authorizations, building programs, technical and business cutover, data migration, user training, software testing and post go-live support, as well as change management.*

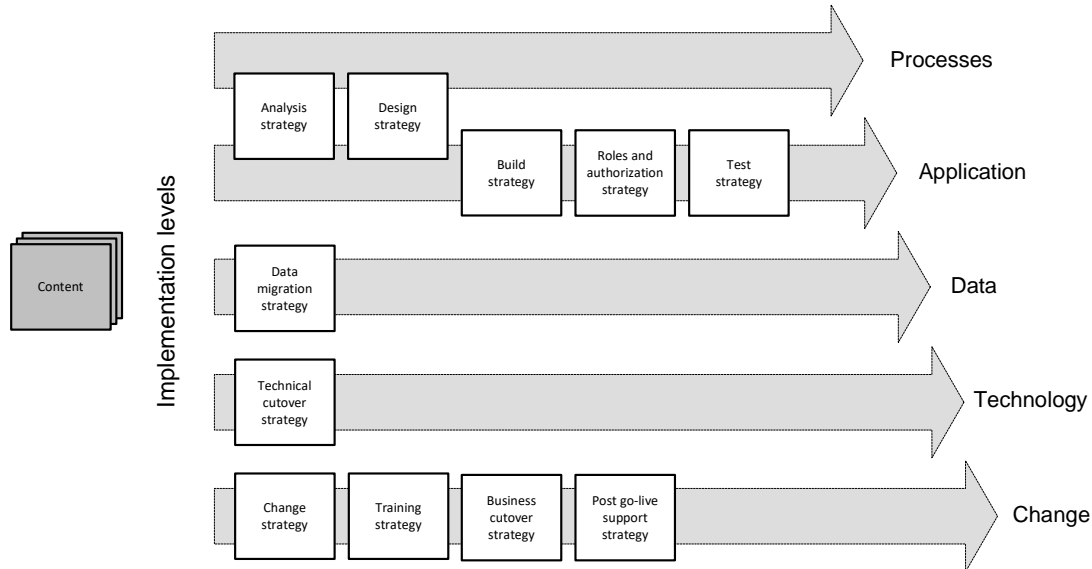


Fig. 4. Strategies per implementation levels.

4. Application of the theory in real ERP projects

Let's apply the theory of corporate information systems to real projects for the implementation of ERP-systems. The initial result of applying the theory is prepared strategies for managing and delivering the project content. The generated strategy documents are further detailed to specific steps, indicating the start and finish dates, as well as a set of documents to be prepared. Table 1 contains a list of key issues addressed in each of the strategies. The most typical solutions to the raised questions in practice are:

- a common analysis strategy is one in which the requirements are identified during the presenting of the demo version of the software system, and the estimator mechanism is used to calculate the efforts of development (allows for a pair of «type of development / configuration – complexity» to select the labor costs of preparing project documents and developing a program) [22];
- business process design involves creation of a business process map up to the 3rd level of detail, which makes it easier to identify requirements. ARIS VACD is often used as graphical notation for business processes at the top levels, SLD-based notation (Swim Lane Diagram) is used at the low levels, while detailing is carried out up to 4-5 levels [23];
- following the strategy of roles and authorization, usually a lot of business roles can be assigned to the end user, since in this case the creation and configuration of technical roles in the software system is greatly simplified [24];
- within technical cutover, the sandbox system is configured as an independent software subsystem, which provides the possibility of modifying critical functions and eliminates the impact on the CIS. Usually, separate copies of the quality control environment of CIS are created for all types of testing and migrations. The number of test technical cutovers mainly correlates with the number of tests and is equal to three [25];
- the change management strategy implies the choice of enterprise parameters to control their changes. When implementing CIS, usually only three parameters are considered: technologies, processes and people, they are enough to provide training for employees, updating job descriptions and work regulations [20];
- the naming convention provides a unified approach to the creation and naming of new program entities in CIS. Quality control of software products, in particular, the use of constant variables, mandatory indication of

organizational levels at the program selection screens and verification of user credentials, allows you to pre-exclude typical algorithmic errors. The best practice for the implementation of corporate information systems implies the formation of such a build strategy, which confirms the use of the naming convention, as well as quality control procedures [26];

- a common practice is the allocation of a separate team responsible for the migration of master data, thereby indicating a functional organizational structure of the migration team. The number of test migrations is aligned with the number of tests: unit, integration and acceptance, thus three waves of migration. Each test migration requires loading of a certain amount of real data, the value of which is mainly close to 100%. Unlike transactional data, the master ones do not change so regularly, so they are often migrated long before go-live [19];
- the training strategy should ensure maximum reduction of labor costs. Therefore, not user instructions are being prepared, but training materials containing a technical description of the operations performed. The key users of the customer are trained so that they are the ones who will train the end users in the future [27];
- in the context of the testing strategy, unit, integration and end-to-end acceptance testing are most often carried out. The criteria for successful completion of tests are the number of open critical defects tending to zero or 1-5 according to the results of the acceptance test [27];
- in the process of working out a business cutover, the shutdown of the enterprise during blackout period is often emulated;
- post go-live support is organized as follows. All the questions that arise from end users go through key users. If key user cannot give an answer, then a defect is registered. Usually support service is organized the way that representatives of the customer's Help desk work on the 1st line. The project team is working only at the 2nd line of support, when defects have already been registered and allocated. The main criteria for the completion of support is the absence of critical defects during last 1-3 days.

Table 1. Implementation strategies per level.

Level	Strategy	Parameter	Possible values
Processes, Application	Analysis	A way to identify requirements	Prototyping, demonstration of the system
		Requirements assessment method	Expert assessment, Estimator
Processes, Application	Design	The need to create a business process map	Yes, no
		Top-level design notation	ARIS VACD, IDEF0
		Low-level design notation	ARIS eEPC, BPMN SLD
		Depth of low-level description	3-5 levels
Application	Roles and authorization	The number of roles to be assigned to user	1 or more
Technology	Technical cutover	Type of sandbox system	Independent subsystem, dependent environment
		Number of copies of the quality control environment	1-3
		Number of test technical cutover	1-3
Change	Change	The need to assess of changes in technologies, processes and people	Yes, no
Application	Build	The need to use naming convention for technical objects	Yes, no
		Use of program quality control procedures	Yes, no
Data	Migration	Organizational structure of the migration team	Functional, matrix
		Number of test migrations	1-3
		%-data uploads for test migration waves	Meaning %-uploads
		The need for early migration of master	Yes,

Level	Strategy	Parameter	Possible values
		data	no
Change	Training	Type of training materials	Operational or process, technical or business
		Type of listeners	Key users, end users
		Training method	By the project team, by the key users
Application	Testing	Types of testing	Unit, integration, acceptance, stress, regression
		Criteria for successful completion of testing	%-passed test scenarios, the number of open critical defects
Change	Business cutover	The need for a company blackout rehearsal	Yes, no
Change	Post go-live support	The level of support at which the project team will work	1-3
		Criteria for the completion of productive support	Number of open critical defects

As you can see from the table above, each strategy defines a group of tasks that are interconnected with all the others. That is why project schedule and resource plan are based on formed strategies, from which it is possible to understand the efforts. Thus, any project plan is a set of work plans for corresponding delivery strategies. The effect of the proposed theory does not end there, further, according to the Deming PDCA cycle, each of the tasks must be performed, its result must be controlled, and also corrected in case of variances, which is done at the project level.

4. Conclusions

So, what prompted the author to write this article? After analyzing the literature, the following reasons can be distinguished:

- there are a large number of technical disciplines [3, 11-15] devoted to the design and development of information systems. Despite this, when you faced in real life with a project for the implementation of a corporate information system, it is not always clear and obvious how to use the knowledge gained;
- a detailed immersion in corporate systems implementation projects reveals a lack of knowledge, since the emphasis in them is on completely different tasks than described in textbooks and most articles. Moreover, the question of integration and alignment of all project activities remains open, because the implementation project has a single goal, regardless of its content;
- each project is unique due to the specificity of the customer's enterprise, which sets the focus and priorities of project tasks. The experience and knowledge gained on one project cannot simply be copied and applied to another. This means that there is no single universal formula for the implementation of corporate systems, some adaptation of the approach to implementation is always required.

Further, a more detailed analysis of the implementation levels of information systems was carried out [3], the results are as follows. If we look at the subject of databases in more detail, it is easy to find that the main focus is on creating balanced data tables, building the right relationships, as well as shaping SQL-queries [28]. But the situation is different in corporate systems implementation projects: the described tasks are typical and have already been implemented in most information systems. For the project, the activities of harmonization, extracting, transformation, loading and validation of data from the legacy to the target software system are much more important, which is practically not covered in the literature. A similar situation is observed with the design of business processes. Graphic notations are given in almost every textbook on information systems, which is

explained by the need to identify bottle necks in the company's work [23]. But this is not so important in CIS implementation projects, moreover, the deployment of a software system already involves standardization and transparency of processes, which is called best practice. Therefore, CASE tools in corporate systems implementation projects can be considered more as a mechanism for visualizing and explaining processes than as a means of analyzing and improving them.

Things are better with software development and technical architecture. Many scientific and popular works are devoted to program coding [8-9], but even here corporate systems contribute: more attention is paid to the unification and standardization of all developments than to the optimality of the program code. The technical architecture in the context of corporate information systems remains virtually unchanged, however, development, testing and productive environments are being introduced, since the requirements for software products may change during their use in real time. The problems of project management, as well as changes during the implementation of CIS, are practically not mentioned in the literature. The maximum that can be easily found is three classic software development models: cascading, iterative and spiral [4], as well as a description of the difficulties of implementing any changes in the company, which only indirectly highlights the issues mentioned. All this has created the need to offer a new discipline that combines the design of information systems, databases, programming, as well as project and change management, called the theory of corporate information systems.

Unlike subjects devoted only to the design of information systems and technologies, the theory of corporate systems pays attention to the formalization, planning and consistency of all project tasks performed. In addition, great emphasis is placed on the scale of operations that require user involvement. Project tasks are grouped by implementation levels, represented by processes, data, application and technology, as well as project and changes. Each implementation level is led by a separate dedicated subteams and aligned with all others. The theory assumes formalization of strategies for analysis, design, roles and authorization, technical and business cutover, change management, build, data migration, training, testing and post go-live support, each of which is decomposed into subtasks, consistent with other activities, and then planned for execution. Planning plays a key role during the implementation of corporate systems, since most of the tasks involve receiving feedback from end users. Thus, the theory of corporate information systems combines the existing technical disciplines and expands their use to larger-scale IT projects.

References

1. Lang V (2021) Digital fluency. APress.
2. Lorizio L (2021) Digitalization: the new normal of the post-pandemic world. Independently published.
3. Gvozdeva T, Ballod B (2009) Information systems design. Phoenix, Moscow. (in Russian).
4. Samara T (2018) ERP and information systems. Integration or disintegration. John Wiley & Sons Limited.
5. Gavrilov D (2008) Production management based on the standard MRPII. Piter, Spb. (in Russian)
6. Standards of corporate information systems (2022) Corporate information systems, Journal. <http://corpinfosys.ru/knowledgebase/standards>. Accessed 14 July 2022. (in Russian).
7. Stepanov D (2021) Shaping ERP3 standard to manage corporate information systems in the time of Industry 4.0. In: Abstracts of 3rd International Conference on Control Systems, Mathematical Modeling, Automation and Energy Efficiency, Russia, Lipetsk, 10-12 November 2021.
8. Knuth D (1997) The art of computer programming. Volume 1: fundamental algorithms. Addison-Wesley Professional.
9. Knuth D (1997) The art of computer programming. Volume 2: seminumerical algorithms. Addison-Wesley Professional.
10. Knuth D (1998) The art of computer programming. Volume 3: sorting and searching. Addison-Wesley Professional.
11. Ostroukh A, Surkova N (2019) Information system design. Lan, Moscow. (in Russian).
12. Laudon J, Laudon K (2002) Management information systems. Prentice hall.
13. Sudakov V (2016) Corporate information systems. MAI, Moscow. (in Russian).
14. Tsiteladze D (2022) Project management. Infra-M, Moscow. (in Russian).
15. Shirenbek H, Lister M, Kirmse Sh (2017) A guide to the project management body of knowledge: sixth edition. PMI.
16. Pfleeger S, Atlee J (2009) Software engineering: theory and practice. Pearson.
17. Stepanov D (2015) Analysis, design and development of corporate information systems: theory and practice. Russian technological journal: 227-238. (in Russian).
18. Oleinik P (2012) Corporate information systems. Piter, Spb. (in Russian).
19. Eremenko Ya (2019) Features of data migration to SAP ERP. Corporate information systems: 22-28. (in Russian).
20. Kozhevina O (2012) Change management. Infra-M, Moscow. (in Russian).
21. Habadi A (2017) An Introduction to ERP-systems: architecture, implementation and impacts. International journal of computer applications: 1-4.



22. Miglinets Yu (2008) Analysis of requirements for automated information systems. Binom, Moscow (in Russian).
23. Kovalev S, Kovalev V (2012) Secrets of successful enterprises: business processes and organizational structure. Bitek, Moscow (in Russian).
24. Petrov S (2018) Strategy of roles and authorizations in ERP projects. Corporate information systems: 53-58. (in Russian).
25. Pryadilnikov E (2018) Strategy of technical cutover in ERP projects. Corporate information systems: 44-49. (in Russian).
26. Pryadilnikov E (2018) Strategy of development in ERP-systems implementation projects. Corporate information systems: 10-16. (in Russian).
27. Stepanov D (2015) Problems of implementation of corporate information systems: application level. Management today: 180-191. (in Russian).
28. Scott L (2021) Relational database and transact SQL: second edition. Independently published.