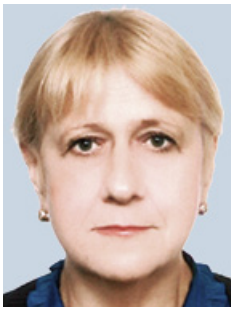
ECONOMICS

UDC 658.15
DOI: 10.31733/2786-491X-2021-2-48-55

Volodymyr GRIGORENKO©
D.Sc. in Technics,
Professor
(NTU "Dnipro Polytechnic"),
Ukraine

Bijay Kumar KANDEL©
Ph.D. (Management)
(Kathmandu University, University of the South Pacific),
Great Britain



Tetyana KADYLNYKOVA©
D.Sc. in Technics,
Professor
(Oles Honchar Dnipro National University),
Ukraine

MANAGEMENT OF THE SYSTEM OF OPTIMIZATION OF PARAMETERS OF BUSINESS PROJECTS

Abstract. In this work, an adaptive approach is used to optimize the functioning of projects, while a computer decision support system is being successfully implemented. The decision support system is based on the integration of information management systems and database management systems and includes data warehouses and tools for their processing.

The conducted studies have shown the possibility of improving the project parameters when using a decision support system that is adaptive to the incoming flow of requests, which dynamically changes the number of functioning structural elements, being within a given limit of the intensity of the information flow, using the available resources as efficiently as possible.

Keywords: *project, system, optimization, control, analysis, management*

Introduction. Constant changes in the situation on the world markets force many commercial enterprises and financial companies to optimize their business projects. The practice of optimizing such projects shows that for its implementation it is necessary to obtain detailed information for analyzing the parameters to determine their priority in the optimization process.

Optimizing a project is the process of improving its various characteristics and

© Grigorenko V., 2021
ORCID iD: https://orcid.org/0000-0002-1809-2842
gvu135gvu@i.ua

© Kandel B., 2021
ORCID iD: https://orcid.org/0000-0003-3793-2410
bijaykkandel@gmail.com

© Kadylnykova T., 2021
ORCID iD: https://orcid.org/0000-0003-0817-9466
kadilnikovatm@ukr.net

processes through transformation. The main characteristics that are improved in the process of project optimization are time, volume, degree of parallelism, or asynchrony of processes. These characteristics depend on the initial data, therefore, there is a distinction between absolute optimization, in which the values of the characteristics are improved for any initial data, and optimization of the priority of parameters.

Analysis of recent research and publications. There are many suggestions for prioritizing data processing optimization parameters. In (A. Shishkin, & Ye. Chernetsova, 2013), and information aggregation algorithm is proposed, based on combining data from various sources, which contain useful for analysis purposes information. To unify information (E. Akimkina et al., 2016), universes, which are a set of objects grouped into classes and reflecting the subject area of are used. In the works (P. Bocharov, & A. Pechinkin, 1995; F. Chan et al., 2016), storage modes and methods for collecting and merging different types of data are proposed, which make it possible to reduce the time for data processing and the amount of disk space for storing information. The works (V. Tomashevskiy, & Ye. Zhdanova, 2003, V. Sovetov, 2010) are devoted to the determination of the composition and structure of parameters in the corresponding base of models. However, for real-life optimization systems, the approach proposed in these works is not acceptable, since it does not consider the failure of individual optimization elements, their cyclical change depending on the current load on the system as a whole.

The purpose of the article is to study the management of the system of optimization of parameters of business projects.

Formulation of the main material. Currently, business project management involves the implementation of a number of stages:

– formation of target states of the project;
– process automation using IT solutions;
– control and analysis of the project (controlling);
– improvement of management processes.

Currently, the following main stages can be distinguished in the organization of controlling:

– definition of performance indicators and “control points”;
– monitoring of process indicators;
– analysis of project monitoring results;
– comparison of actual and planned indicators of the project and finding the reasons for deviations.

In this situation, it is advisable to use an adaptive approach for changing conditions of the external and internal environment in order to optimize the functioning, which can be successfully implemented using a computer decision support system (DSS).

DSS acts as a computing unit and a control object. The decision maker (DM) acts as a control link: he sets the input data, rules and algorithms, collects data from heterogeneous sources using queries and evaluates the solutions proposed by the computer system based on the described rules and algorithms, and, in the event of a risk decision making, forms additional requests and scenarios.

SSPR are developed on the basis of integration of information management systems and database management systems (Frenk, 2003) and include data warehouses and tools for their processing. This leads to the need to use specialized tools for analyzing processes, such as information systems of the Process Intelligence class, which is a Business Intelligence (BI) platform with advanced analysis tools.

The main advantages of the Process Intelligence toolkit are as follows:

– project monitoring based on metrics;
– automatic warning system in case of deviations from the planned values;
– the ability to make changes to business processes;

– automatic visualization of the stages of the project;
– benchmarking based on process indicators;
– identifying potential for optimizing business processes;
– control of measures to improve business processes.

A conceptual diagram has been developed to simulate the optimization processes of the project parameters (Fig. 1).

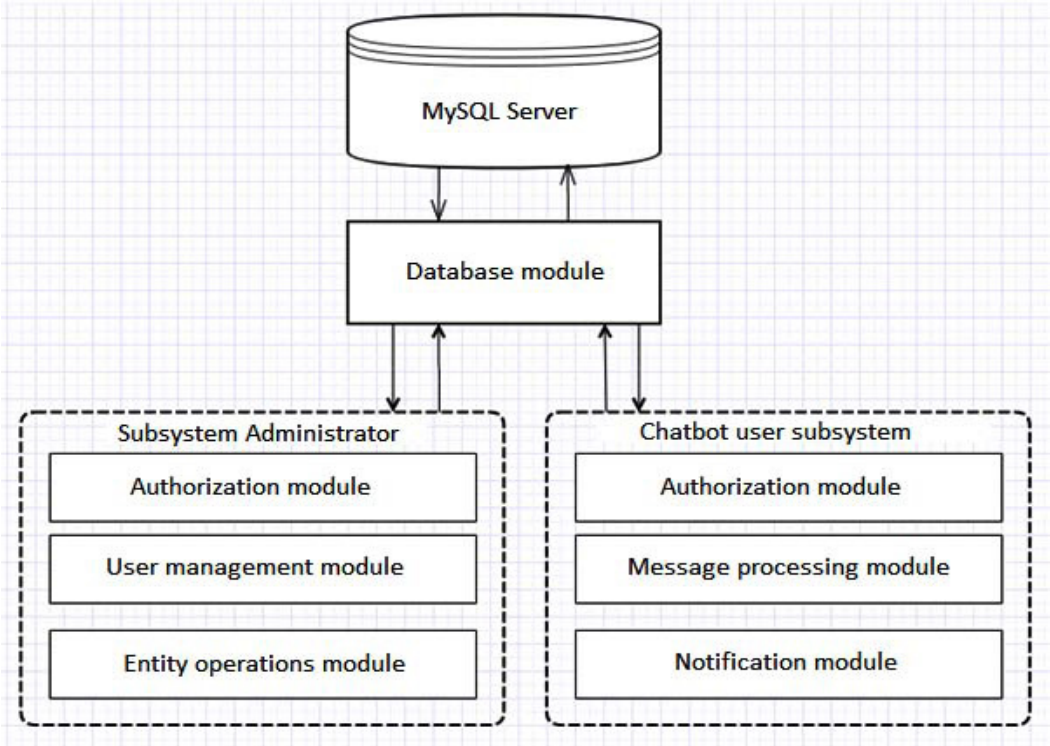


Figure 1 – Conceptual diagram of the project parameters optimization system

The diagram consists of the following parts:
– MySql Server – a server responsible for storing data necessary for the system to work;
– a module for working with a database, which is responsible for creating, reading, deleting and updating information in the database;
– Administrator of subsystem, which is used to manage users and operations with basic entities;
– a subsystem of the chatbot user responsible for the interaction of the chatbot users with the system.

Figure 1 shows that the administrator subsystem and the chatbot user subsystem interact with the same module for working with the database, which is the result of correct abstraction and providing the system with a generalized interaction interface.

At the same time, it is impossible to abstract the authorization modules, since the authorization and authentication methods in the administrator subsystem and the chatbot user subsystem are carried out on different servers: the administrator is authorized on the server side by validating the Json Web Token, and the chatbot user is authorized on the servers Telegram.

JSON Web Token (JWT) – it is an open standard (RFC 7519) for creating access tokens based on the JSON format. Typically used to pass authorization data in client-server applications. Tokens are created by the server, signed with a secret key, and transferred to the client, who later uses this token to confirm his identity.

The user management module, which is part of the administrator subsystem, is responsible for creating, confirming, editing, and deleting system users. Thus, the system administrator can create a new administrator or head of the structural department, confirm the user registration in the chatbot, change the data of the chatbot user, or delete the user. The entity operations module includes the ability to create, read, update or delete the main entities of the system. In turn, the chatbot user subsystem includes modules for processing messages and notifying users. The message processing module includes receiving a text message from the user, processing it, performing any actions in the system, and sending a response message to the user. The user notification module includes sending messages to users when events occur in the system that the user should be notified of. If any of the above events occur in the system, the user will receive a message from the chatbot with all the necessary information.

The general architecture of the project parameters optimization system is shown in Fig. 2.

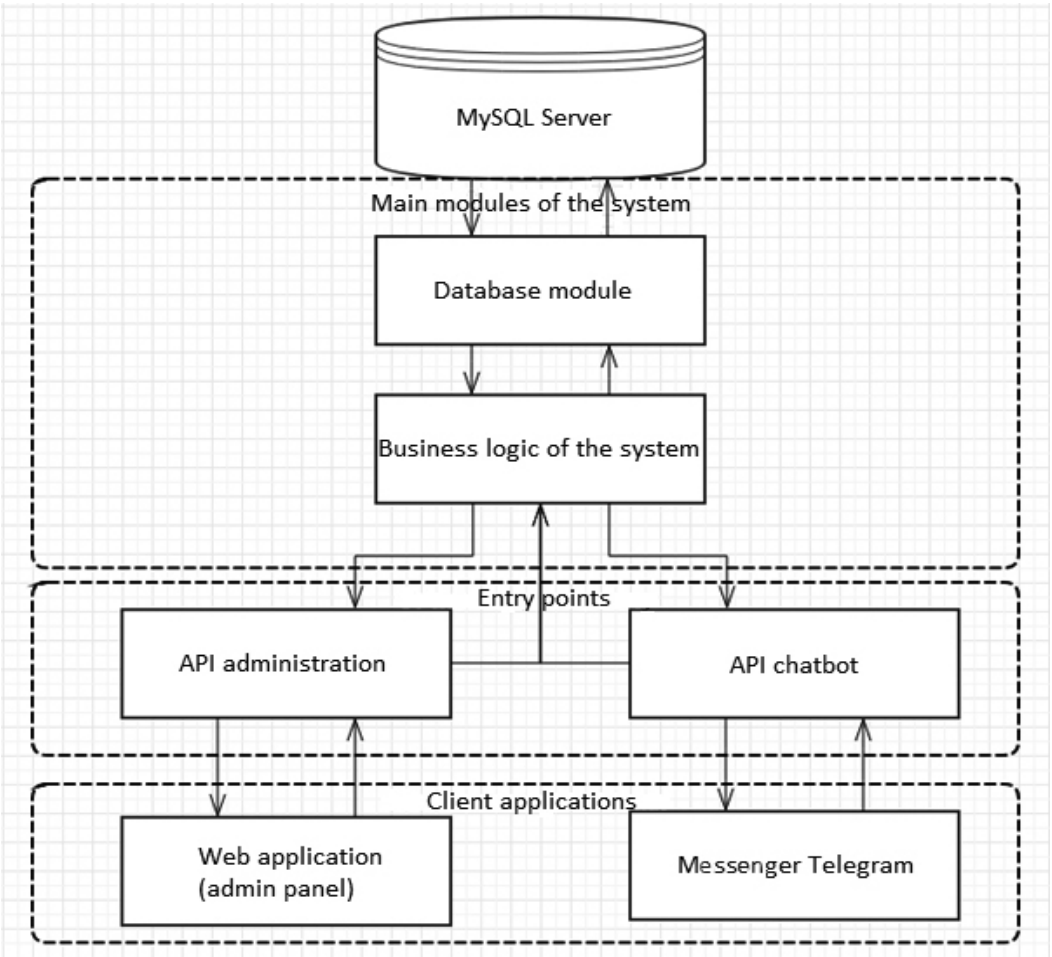


Figure 2 – General architecture of the project parameters optimization system

The server side of the system is a RESTful web service. REST (abbr. Representational State Transfer) is an architectural style of interaction between components of a distributed application in a network. REST is a consistent set of constraints to consider when designing a distributed hypermedia system. This, in certain cases, leads to an increase in system performance and a simplification of its architecture.

On the Internet, a remote procedure call can be a regular HTTP request (usually "GET" or "POST"; such a request is called a "REST request"), and the required data is passed as request parameters.

For web services built with REST in mind (that is, without violating the constraints imposed by it), the term "RESTful" is used.

Unlike SOAP-based web services (web services), there is no "official" standard for RESTful web APIs. This is because REST is an architectural style while SOAP is a protocol. Although REST is not a standard, most RESTful implementations use standards such as HTTP, URL, JSON, and XML.

Interaction with the server occurs by sending HTTP requests to specific URLs. Depending on the HTTP type, method, request path, and request body, the server performs certain actions and returns the result to the client.

It is advisable to adhere to the SOLID principles when designing the application architecture.

SOLID principles (short for single responsibility, open-closed, Liskov substitution, interface segregation, and dependency inversion) in programming are designed to increase the likelihood that a programmer will create a system that is easy to maintain and expand over time. The SOLID principles are guidelines that can be applied while working on software, instructing the programmer to refactor the source code until it is legible and extensible. This is part a of an overall agile and adaptive development strategy.

Here are the names and a brief description of these principles.

1. The principle of sole responsibility. There is only one reason for the class change.

2. The principle of openness (closedness). Software entities must be open for extension but closed for modification.

3. The substitution principle of Barbara Liskov. Objects in a program must be replaceable with instances of their subtypes without changing the correct execution of the program.

4. The principle of interface separation. Many customer-specific interfaces are better than one general-purpose interface.

5. The principle of inversion of dependencies. There is a dependency on abstractions, there is no dependence on specifics.

During the design of the architecture, it was decided to divide the project into 3 subprojects: API, Application Core, Infrastructure. Let's consider each of them in more detail.

The Application Core project is central to clean architecture design and should be referenced by all other projects. As a result, this project has a minimum number of external dependencies. This project should include:

– domain objects;
– interfaces;
– services;
– domain errors;
– specifications;
– DTO objects.

An interface is a programmatic structure that defines a relationship between objects that share a certain behavioral set and are not related in any other way. When designing classes, designing an interface is identical to designing a specification — a set of methods that every class that uses an interface must implement.

Interfaces, along with abstract classes and protocols, establish mutual obligations between elements of a software system, which is the foundation of the concept of programming by contract (DbC). An interface defines the boundary of interaction between classes or components by specifying a certain abstraction that the implementing side implements.

An OOP interface is a strictly formalized element of an object-oriented language and is widely used in the source code of programs.

Service is a class that inherits a specific interface and contains the business logic of the application.

A "specification" in programming is a design pattern by which the

ISSN 2786-491X (Print)

representation of business logic rules can be transformed into a chain of objects linked by Boolean logic operations. The business logic object inherits its functionality from the Composite Specification abstract aggregate class, which contains just one Is Satisfied By method that returns a boolean value. After instantiation, the object is chained with other objects. As a result, without losing flexibility in configuring business logic, we can easily add new rules.

Data Transfer Object (DTO) — a design pattern used to transfer data between subsystems of an application. A Data Transfer Object, unlike a business object or data access object, should not contain any behavior (pattern, design).

Most of the dependencies on external resources must be implemented in the classes defined in the Infrastructure project. These classes must implement the interfaces declared in the Application Core project. In this application, the Infrastructure project is responsible for working with the database. It implements the EfRepository class that implements the IRepositry <T> and IAsyncRepository <T> interfaces declared in the Application Core project.

One of the most commonly used patterns when working with data is the Repository pattern, which allows you to abstract from the specific connections to the data sources with which the program works, and is an intermediate link between the classes that directly interact with the data and the rest of the program.

With the standard approach, even in a small application that selects, adds, changes, and deletes data, leads to a large number of changes, and the Repository adds flexibility to the program when working with different types of connections.

The API project is the entry point to the application. This project must contain:

– controllers;
– filters;
– middleware;
– viewmodels.

The controller provides “connections” between the user and the system. Controls and directs data from the user to the system and vice versa. Uses model and view to implementing the required action.

Filter – allows you to execute a certain piece of code before or after certain stages of request processing. Middleware is software that pipelines an ASP.NET application to handle requests and responses.

ViewModel – is, on the one hand, an abstraction of the View, and on the other hand, it provides a wrapper for the data from the Model, which are subject to the binding. That is, it contains a Model that has been converted to a View, and also contains commands that the View can use to influence the Model.

Thus, following the principles of SOLID and dividing the project into 3 subprojects, we got an easily maintainable and easily extensible application that meets modern architectural software standards.

This software application is implemented using ASP.NET Core technology using the C # programming language. ASP.NET Core is a cross-platform, high-performance, open-source framework for building modern, Internet-connected cloud applications.

ASP.NET Core ships completely as NuGet packages. Using NuGet packages allows you to optimize your applications to include only the required dependencies (pattern, design).

Based on the fact that this application is a Web-API, the Postman application was chosen as the testing tool. The program has a built-in query editor, with the ability to encode, load from a file, and send binary data. The history of past requests is saved by category (collection). You can write Markdown comments for each request. Postman supports several types of authentication when sending requests: Basic Auth, Digest Auth, and OAuth 1. Microsoft Visual Studio Community 2017 was used as a development environment.

Microsoft Visual Studio – allows you to develop both console and GUI applications, including those supporting Windows Forms technology, as well as websites, web applications, web services in both native and managed code for

everyone platforms supported by Windows, Windows Mobile, Windows CE, .NET Framework, Xbox, Windows Phone.NET Compact Framework, and Silverlight.

Visual Studio includes a source editor with IntelliSense support and easy code refactoring. The built-in debugger can function as either a source-level or machine-level debugger. The rest of the plug-in tools include a form editor to simplify the creation of an application's GUI, a web editor, a class designer, and a database schema designer. Visual Studio allows you to create and connect third-party add-ons (plugins) to extend functionality at almost every level, including adding support for source code version control systems (such as Subversion and Visual SourceSafe), adding new toolkits (for example, for editing and visual design code in domain-specific programming languages) or tools for other aspects of the software development process (for example, the Team Explorer client for working with Team Foundation Server).

To access the documentation generated by the tool presented above, you need to follow the path/swagger:

– The URL to which you want to send the HTTP request;
– Required HTTP method;
– JSON object to be passed to the server;
– HTTP status codes that the server can return.

Thus, using the documentation generated using the Swagger tool and collections of requests from the Postman application, the developers of client applications have access to modern and reliable information about the capabilities of the Web-API with which they interact, which greatly increases the speed of development.

It was decided to use MySql as a DBMS.

MySQL is a free relational database management system. MySQL is a solution for small and medium-sized applications, included in WAMP, AppServ, LAMP servers, and in portable server assemblies Denver, XAMPP, VertrigoServ. Usually, MySQL is used as a server that local or remote clients access, however, the distribution includes an internal server library that allows MySQL to be included in standalone programs.

The flexibility of MySQL is provided by the support of a large number of table types: users can choose from both MyISAM tables, which support full-text search, and InnoDB tables, which support transactions at the level of individual records. Moreover, MySQL comes with a special EXAMPLE table type that demonstrates how to create new table types.

Thanks to its open architecture and GPL licensing, new types of tables are constantly appearing in MySQL. This application uses Entity Framework Core and the Code First approach to work with the database.

Entity Framework Core (EF Core) is an object-oriented, lightweight, and extensible technology from Microsoft for data access. EF Core is an ORM tool (object-relational mapping – mapping data to real objects) and allows you to work with databases, while presenting a higher level of abstraction, namely, it allows you to abstract from the database itself and its tables and works with data independently on the type of storage. If at the physical level we operate with tables, indexes, primary and foreign keys, but at the conceptual level that Entity Framework offers us, we are already working with objects.

With the Code-First approach, the model is first defined in code, then a so-called migration is created from it, which is applied to the database. Migration is a file containing commands that must be applied to an existing database for it to match the current domain model of the application. Thus, every time the domain model changes, a new migration must be created and applied.

By leveraging Entity Framework Core and a Code First approach, you can fully automate your database design process and start coding your domain model classes immediately.

Conclusions. The conducted studies have shown the possibility of improving the project parameters when using an adaptive DSS concerning the incoming flow of requests, which dynamically changes the number of functioning structural

ISSN 2786-491X (Print)

elements, being within a given limit, depending on the intensity of the information flow. Adaptive DSS is the most balanced version of the system, in which additional parameters are turned on only when the load on the system increases, and are turned off when the load on the system decreases, that is, it works in the optimal mode, using the available resources as efficiently as possible.

Conflict of Interest and other Ethics Statements
The authors declare no conflict of interest.

References

Akimkina, E., Samarov, K., Vasil'yev, N., & Belyuchenko, I. (2016). Evaluation of the effectiveness of customer service in the context of system analysis. *Issues of regional economy*, *2*(27), 123-134. (in Russian).

Bocharov, P., & Pechinkin, A. (1995). *Theory of queuing.* Moscow : RUDN. (in Russian).

Chan, V., Shcherbakov, M., Nguyen, T. & Skorobogatchenko, D. (2016). A method for collecting and merging heterogeneous data in proactive intelligent decision support systems. *Neurocomputers: development, application, 11*, 40-44. (in Russian).

Entity Framework Core (2021), Electronic Resources: finding resources by subject, https://metanit.com/sharp/entityframeworkcore/1.1.php.

Frenk, Kh. (2003). *Teoriya grafov.* Moscow: Yeditorial URSS. (in Russian).

Shishkin, A., & Chernetsova, Ye. (2013). Algorithm for optimizing the number of requests to information sources in the decision support system. *Information technology and system: management, economics, transport, law, 1*(10), 67-69. (in Russian).

Sovetov, V. (2010). Service system with a variable number of service elements. *Electrotechnical complexes and systems*, *6*(1), 10-14. (in Russian).

Tomashevskiy, V., & Zhdanova, Ye. (2003). *Simulation modeling in the GPSS environment.* Moscow : Bestseller. (in Russian).

Володимир Григоренко, Біджай Кумар Кандел, Тетяна Кадильникова

УПРАВЛІННЯ СИСТЕМОЮ ОПТИМІЗАЦІЇ ПАРАМЕТРІВ БІЗНЕС-ПРОЄКТІВ

Анотація. У даній роботі використовується адаптивний підхід з метою оптимізації функціонування проектів, при цьому успішно реалізується комп'ютерна система підтримки прийняття рішень. Адаптивна система підтримки прийняття рішень являє собою найбільш оптимальний варіант системи, в якому додаткові параметри включаються тільки тоді, коли навантаження на систему зростає, і вимикаються, коли навантаження на систему знижується, тобто працює в оптимальному режимі, максимально ефективно використовуючи наявні у неї ресурси.

Для здійснення оптимізації параметрів проекту необхідно отримання детальної інформації для аналізу з метою визначення першочерговості параметрів в процесі їх оптимізації. Системи підтримки прийняття рішень розробляються на основі інтеграції інформаційно-керуючих систем і систем управління базами даних і включають сховища даних і інструментальні засоби для їх обробки. Це призводить до необхідності застосовувати спеціалізовані інструменти аналізу процесів, як, наприклад, інформаційні системи класу Process Intelligence, що представляють собою платформу Business Intelligence (BI) з розширеними засобами аналізу.

У роботі обґрунтовано доцільність побудови концептуальної схеми системи оптимізації параметрів проекту, а також описані основні її модулі. При проектуванні архітектури системи оптимізації параметрів проекту було вирішено розділити проект на 3 підпроекти: API, ApplicationCore, Infrastructure, що дозволило отримати легко підтримуваний і легко розширюваний додаток, який відповідає сучасним архітектурним стандартам програмного забезпечення.

Проведені дослідження показали можливість поліпшення параметрів проекту при використанні адаптивної по відношенню до вхідного потоку запитів системи підтримки прийняття рішень, яка динамічно змінює кількість функціонуючих структурних елементів, перебуваючи в межах заданого ліміту інтенсивності потоку інформації.

Ключові слова: *проект, система, оптимізація, контроль, аналіз, управління*

Submitted: 16.05.2021
Revised: 29.08.2021
Accepted: 15.09.2021