



*International periodic scientific journal*

ONLINE

*www.sworldjournal.com*

*D.A.Tsenov Academy of Economics - Svishtov (Bulgaria)*

Indexed in  
INDEXCOPERNICUS  
(ICV: 73)  
GOOGLESCHOLAR

# SWorld Journal

**Issue №38**

**Part 1**

**July 2026**

*Published by:*

*SWorld & D.A. Tsenov Academy of Economics, Svishtov, Bulgaria*

**Editor:** Shibaev Alexander Grigoryevich, *Doctor of Technical Sciences, Professor, Academician*  
**Scientific Secretary:** Kuprienko Sergiy, *PhD in Technical Sciences*

**Editorial board:** More than 400 doctors of science. Full list on page:  
<https://www.sworldjournal.com/index.php/swj/about/editorialTeam>

**Expert-Peer Review Board of the journal:** Full list on page:  
<https://www.sworldjournal.com/index.php/swj/expertteam>

The International Scientific Periodical Journal "SWorldJournal" has gained considerable recognition among domestic and foreign researchers and scholars. Today, the journal publishes authors from different countries.

Journal Established in 2018. Periodicity of publication: 6 times a year

The journal activity is driven by the following objectives:

- Broadcasting young researchers and scholars outcomes to wide scientific audience
- Fostering knowledge exchange in scientific community
- Promotion of the unification in scientific approach
- Creation of basis for innovation and new scientific approaches as well as discoveries in unknown domains

The journal purposefully acquaints the reader with the original research of authors in various fields of science, the best examples of scientific journalism.

Publications of the journal are intended for a wide readership - all those who love science. The materials published in the journal reflect current problems and affect the interests of the entire public.

Each article in the journal includes general information in English.

The journal is registered in the INDEXCOPERNICUS, GoogleScholar.

DOI: 10.30888/2663-5712.2026-38-01

**Published by:**  
**SWorld &**  
**D.A. Tsenov Academy of Economics**  
*Svishtov, Bulgaria*  
e-mail: [editor@sworldjournal.com](mailto:editor@sworldjournal.com)

---

Copyright  
© Authors, scientific texts 2026



UDC 004.9:412

## MODULE FOR DYNAMIC ANALYSIS OF IT PROJECT TASKS PERFORMANCE

**Udovenko S.G.***d.t.s., prof.*ORCID:<https://orcid.org/0000-0001-5945-86-47>**Zatkhey V.A.***c.t.s., as.prof.*ORCID:<https://orcid.org/0000-0003-4426-7789>**Teslenko O.V.***c.t.s., as.prof.*ORCID:<https://orcid.org/0000-0003-3105-9323>*Simon Kuznets Kharkiv National University of Economics,  
Kharkiv, Nauky Ave. 9-A, 61165*

**Abstract.** *The purpose of this work is to develop a functional module of an agile IT design system that allows you to improve the task execution diagram by expanding the space of input parameters to increase the accuracy of the displayed results. In accordance with the purpose of this study, a set of indicators was determined that should be used by the developed module to build an IT project task execution diagram. Trends in IT project task execution using checkpoints were studied. The module was software implemented and tested. The use of this module in the field of agile project management will increase the efficiency of bottleneck analysis at the stages of planning, development and support of the design life cycle.*

**Keywords:** *IT design, Agile methodology, functional module, task diagram, database, Spring framework.*

### Introduction.

Classic indicators of IT project development efficiency are indicators of dynamic comparison of the actual amount of work performed with the planned amount. But today, Agile projects use simpler and more visual indicators of the development process status, for example, calculating the return on investment ratio, team speed, planned and spent resources, using the mastered volume method, etc. However, these indicators are not always informative enough to reflect productivity during the project development iteration period. Only the set of characteristics and metrics used in Agile allow for a qualitative forecast of the implementation of IT design tasks.

Currently, the IT design information technology market offers various parameter monitoring systems that allow you to operate with a certain set of characteristics to build performance charts, aggregate flow, development progress charts, etc. However, usually these systems provide only a part of the necessary tools for building a burn-



down chart, which would make it possible not only to track the success of task execution in an iteration, but also to predict the effectiveness of design in the future. Agile methodology actively uses the construction of task execution charts, which allow you to analyze existing problems at the time of development and eliminate them for the successful completion of the iteration [1, 2]. There are various options for such diagrams (in particular, automatic generation of diagrams built into defect tracking systems, or their creation in the Microsoft Excel environment). In the commercial error tracking system Jira and Trello, the main element of the design system efficiency analysis is the task (ticket). The task contains the project name, topic, type, priority, components and content. The set of tasks forms the basis for building a schedule for their execution RTC (Rational Team Concert), which consists of four curves:

- remaining work (the amount of work that needs to be done);
- ideal (the amount of work that the team should complete at each checkpoint);
- planned work (the amount of work planned at the beginning of the iteration);
- expected work (the amount of work that the team has completed so far).

The RTC platform is used to manage the IT design lifecycle to ensure contextual collaboration between design teams over time. It provides tools for configuring, managing, and applying processes that support the IT project development environment and helps teams integrate tasks across the entire design lifecycle. RTC simplifies planning and executing flexible or formal projects based on planning tools and templates.

However, it should be noted that the RTC platform does not take into account the impact of some important factors on the IT project task schedule. When building a diagram, indicators such as the number of planned tasks, execution time, and team resources are taken into account. However, this may not be enough to reflect the real situation on the project. During an iteration, the team goes through the main phases of developing an IT project (in particular, a software product): creating an architecture, analyzing requirements, implementing software, preparing test documentation, testing, etc. [3]. If the task execution schedule signals a lag, it is sometimes impossible to identify its cause based only on these characteristics. This happens, first of all, due to



the large amount of noise.

The purpose of this work is to develop a functional module of a flexible IT design system that allows you to improve the task execution diagram by expanding the space of input parameters to increase the accuracy of the displayed results. When developing the module, indicators characterizing human resources should be taken into account, namely: working hours, days, iteration duration, overtime hours, the degree of participation of a particular team member in the project, the maximum speed of task execution by the team.

### **Main part.**

According to the purpose of this study, it is necessary to:

- determine the set of indicators that should be used by the developed module to build a diagram of the execution of IT project tasks;
- determine the trends of the execution of IT project tasks using control points;
- carry out software implementation and testing of the module.

*Determination of indicators of the execution of IT project tasks.* The task execution diagram formed by the developed module contains a set of parameters that determine the trend for building a graph. The diagram consists of a set of characteristics used in Agile methodologies. Each of these characteristics contains certain information about the success of the team in the current iteration and determines the construction of curves on the graph. Such characteristics will be further referred to as: indicators of the reference landmark, indicators of actual task execution, indicators of forecasting task execution in subsequent iterations, indicators of current assessment of the effectiveness of task execution [4, 5].

The benchmark metrics are displayed on a benchmark curve, which defines the total and volume of work that the team must complete at each milestone to complete all tasks on time. Figure 1 shows an example of a benchmark trend that takes into account the estimated intensity of task completion by the design team during the calendar week of the sprint.



**Figure 1 – Example of a reference trend**

The benchmark indicators are defined in man-hours. This unit of measurement allows you to estimate the work during planning more accurately by comparing the number of workers and the timing of the task at control points. The benchmark is determined by the following formula:

$$E = \{E_{start}, E_{week1}, \dots, E_{weekN}\}, \quad (1)$$

where  $E_{start}$  is the estimated amount of work at the beginning of the iteration;  $E_{weekN}$  is the amount of work at each checkpoint.

The expected amount of work at the beginning of the iteration is determined by the sum of two components:

$$E_{start} = S + AVGA_{start}, \quad (2)$$

where  $S$  is the amount of work in the iteration;  $AVGA_{start}$  is the amount of unplanned tasks at the beginning of the iteration.

Each task, the execution of which is planned for a certain period of time (iteration), is estimated in ideal hours. This parameter is present in each task, which allows taking into account all processes for an accurate calculation of the resource intensity of the project at the iteration planning stage. The time with the task is summed up and forms the Team Capacity metric, which gives the team an indicator of how intensively the tasks should be performed.



The indicator of actual task execution (Scope) reflects the status of the team's execution of current tasks (Fig. 2). For each task, the time that has already been spent on its implementation (spent) and the time that remains to be spent (remaining) are determined.

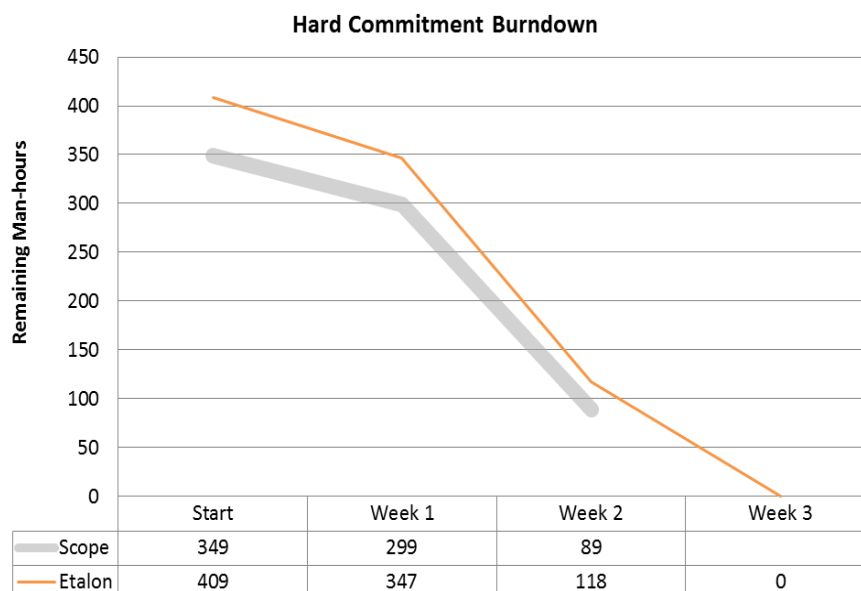


**Figure 2 – Example of actual task execution during an iteration**

It should be noted that usually monitoring the performance of tasks for the current iteration does not reflect a number of unplanned activities that must be performed (in particular, fixing defects and accompanying regression testing of the results). At the same time, the amount of planned work by the end of the sprint often exceeds the initial set of tasks. If the team does not take into account the forecast of the volume of unplanned tasks, this may lead to risks of loss of quality of the design product.

Thus, it is advisable to use in the developed module an indicator of normalized work volume, which includes a forecast for unplanned tasks and exceeding estimates:

$$S + AVG\Delta = S * UTC * EDC, \quad (3)$$

where  $UTC$  is the number of unplanned tasks;  $EDC$  is the measure of exceeding estimates.

An example of using such an indicator is shown in Fig. 3.

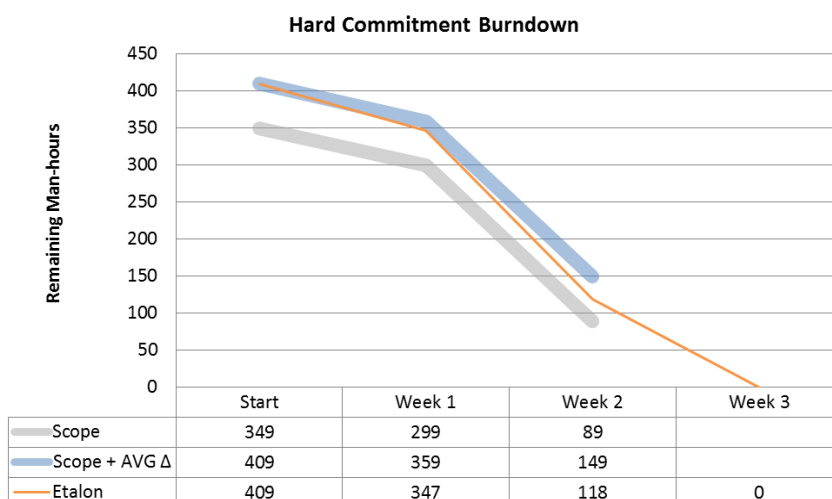


**Figure 3 – Example of using the normalized volume of unplanned tasks indicator**

Forecasting unplanned tasks and exceeding estimates is carried out to calculate the expected result at the end of the iteration. Error tracking systems do not take into account the indicators of such a forecast, which can contribute to the emergence of risks of reducing the quality of IT design. The forecasted characteristics are defined as a weighted average of the results of previous sprints. The weight of a specific iteration should be a consolidated estimate controlled by a set of adaptive rules. The integrated indicator includes the weight of all unplanned tasks at the beginning of the iteration. The forecast model loses its relevance and accuracy if there are more unplanned tasks in the current iteration than predicted (exceeding the weighted average). The indicator that affects the forecast is the difference between the initial estimate of the time to complete a given task and the time actually spent. In the event of a critical difference, it is necessary to influence the size of the increment of the unplanned volume of tasks. The procedure for adjusting the coefficient of such influence is determined for a specific iteration individually according to the following criteria: determining the most anomalous excesses of the estimate; identifying the reasons for the excess; determining preventive measures.

Indicators of the current assessment of the effectiveness of task performance are determined based on the values of the reference and actual indicators considered above according to the following formula:



$$LAG = S + AVG\Delta - E. \quad (4)$$

An example of using such an indicator is shown in Fig. 4.

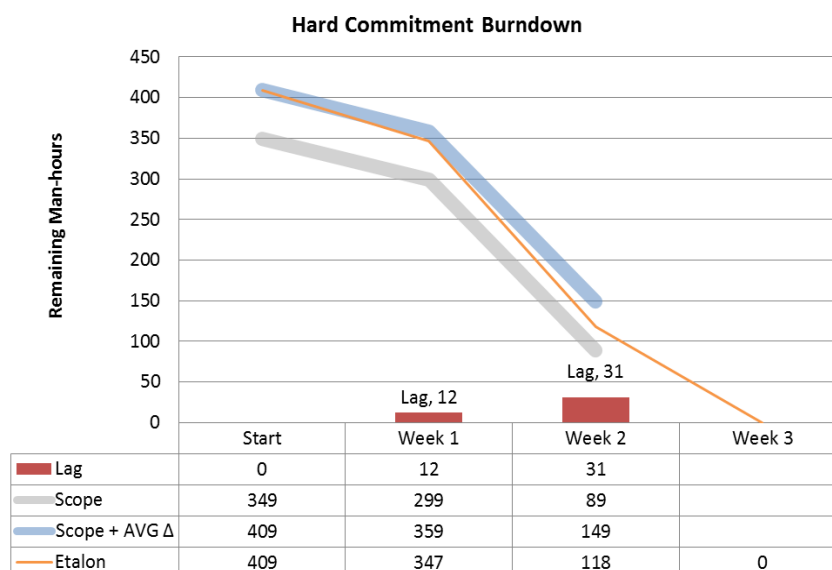


**Figure 4 – Example of current assessment of task performance effectiveness taking into account the forecast for deviations**

If the design team performs tasks evenly, then the deviation from the planned schedule will be zero. If the *Lag* parameter is negative, this signals that the team is ahead of schedule. A positive value of the *Lag* parameter indicates that the team does not have time to complete the work within the specified time. The following reasons may affect this: increasing task complexity; reducing resources; an error in the initial estimate at the iteration planning stage; finding a large number of defects after the testing phase; lack of experience and knowledge in the team to solve the tasks set.

The causes of the risks of lagging behind the schedule can be identified using other indicators, such as the focus factor and the number of man-hours in the current iteration. The focus factor (*FF*) is an indicator of the time that the team spends on the direct implementation of planned tasks. This indicator can be biased by various noises (in particular, the inability or unwillingness of the team to record the time spent at each stage of development). This indicator is calculated using the following formula:



$$FF = \frac{WD}{C} \times 100\%, \quad (5)$$

where  $WD$  is the number of man-hours spent on task implementation;  $C$  is the speed of task implementation in the current iteration.

An example of using the «focus factor» indicator is shown in Fig. 5.

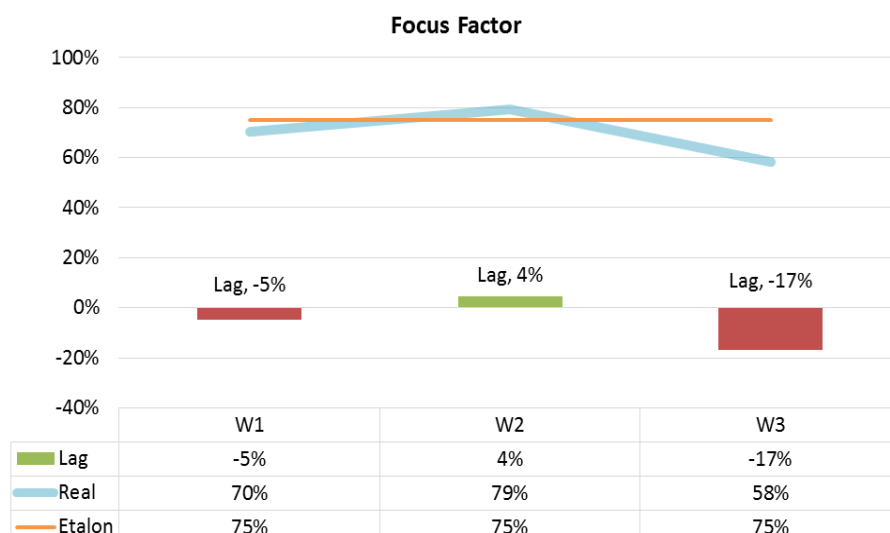


**Figure 5 – Example of using the “focus factor” indicator**

To most accurately identify the cause of a low team focus factor, this parameter should be considered for developers, testers, and other team members separately. For example, the overall indicator may be less than 75% due to insufficient efficiency of business analysts.

*Determining trends in the performance of IT project tasks using checkpoints.* We will determine the trends in the performance of IT project tasks in the module under development using the agile Scrum technology. This technology uses the sprint iteration, during which a functional build-up of the project under development is created. We will conditionally divide this iteration into 3 working weeks (the first and third weeks - 2 working days, the second week - 5 working days). The end of each week is a checkpoint where the current development progress schedule is analyzed. Note that the number of days in each working week is set by the module user at the beginning of the iteration, since the duration of the sprint can vary depending on the requirements and goals of the customer.



An example of setting checkpoints during one iteration is shown in Fig. 6.

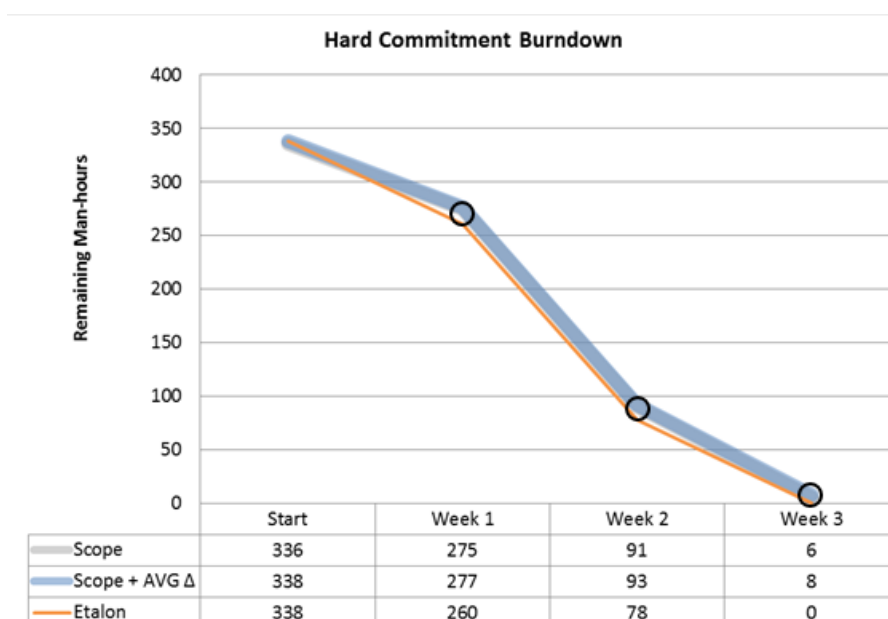


**Figure 6 – Example of setting checkpoints within one iteration**

The basis of the developed module is a database containing histories with a list of tasks, necessary developments, project resources, etc. The basic unit of the module is a task, which stores a set of parameters (criteria) necessary for building a project schedule. The basic set of such parameters is shown in Fig. 7.

| Hard | Plan | Task Name                                                                | Impl. | State |
|------|------|--------------------------------------------------------------------------|-------|-------|
| 1    | 1    | 506073: FT: Test Design TC03.1                                           | QA    | done  |
| 1    | 1    | 506081: CS: WildcardSearchService                                        | DEV   | done  |
| 1    | 1    | 506082: CS: ProjectListViewModel                                         | DEV   | done  |
| 1    | 1    | 506085: CS: ProjectSearchService                                         | DEV   | done  |
| 1    | 1    | 506141: CS: Dev testing                                                  | DEV   | done  |
| 1    | 1    | 506200: CS: List view                                                    | DEV   | done  |
| 1    | 0    | 506700: BUG: When create new project, parameter 'project type' is absent | DEV   | done  |
| 1    | 1    | 506871: FT: Test Design TC03.1                                           | QA    | done  |
| 1    | 1    | 506907: FT Execution TC03.1                                              | QA    | done  |
| 1    | 1    | 506908: FT Execution TC03.2                                              | QA    | done  |
| 1    | 0    | 506326: DEVBUG: fix search view model                                    | DEV   | done  |
| 1    | 0    | 506952: FT: Test data                                                    | QA    | done  |
| 1    | 0    | 506967: DEVBUG: tooltip incorrect                                        | DEV   | done  |
| 1    | 0    | 506973: CS: Remove properties from                                       | DEV   | done  |
| 1    | 0    | 507000: CS: SearchBoxViewModel refactoring                               | DEV   | done  |
| 1    | 0    | 507141: BUG: Search box should be cleared when open Home page            | DEV   | done  |
| 1    | 0    | 507786: Review Issues                                                    | DEV   | done  |
| 1    | 0    | 508732: FT Execution TCs Regression                                      | QA    | done  |

**Figure 7 – Basic set of task performance criteria**



The time assessment criteria used in the module are used to generate trends for building a project execution chart (including benchmarks, normalized volume of unplanned tasks, actual time spent, etc.).

An example of determining such trends is shown in Fig. 8.

|                      |     |     |     |     |     |
|----------------------|-----|-----|-----|-----|-----|
| Scope                | 209 | 336 | 275 | 91  | 6   |
| Scope + AVG $\Delta$ |     | 338 | 277 | 93  | 8   |
| Etalon               |     | 338 | 260 | 78  | 0   |
| Lag                  |     | 0   | 16  | 15  | 8   |
| Done                 |     |     | 18% | 73% | 98% |
| Weekly $\Delta$      |     |     | 18% | 55% | 25% |

**Figure 8 – Example of defining trends for building a task execution chart**

Software implementation and testing of the module. The implementation of the module for dynamic analysis of IT project task execution was carried out based on the MVC pattern scheme. This scheme makes it possible to separate data, presentation, and processing of the module user's actions into three separate components:

- model (model). This component provides data and methods for working with them, and also responds to requests by changing its state;
- presentation (view). This component is responsible for visualizing information using windows with graphic elements;
- controller (controller). This component provides the interface between the user and the module (in particular, it controls data input and uses the model and view to implement the required response).

Spring Framework, which has its own MVC platform for web applications, was used to create a flexible module architecture. Spring MVC is a query-oriented framework. It defines strategic interfaces for all functions of a modern interactive system. The model implements the principle of dependency inversion, which is used to reduce coupling. Graphical representation of the module functions usage options is implemented using the “Users”, “Teams”, “Cards” and “Iterations” forms.

The results of the current task performance analysis are presented on the page with the task board and the time spent graph (Fig. 9).

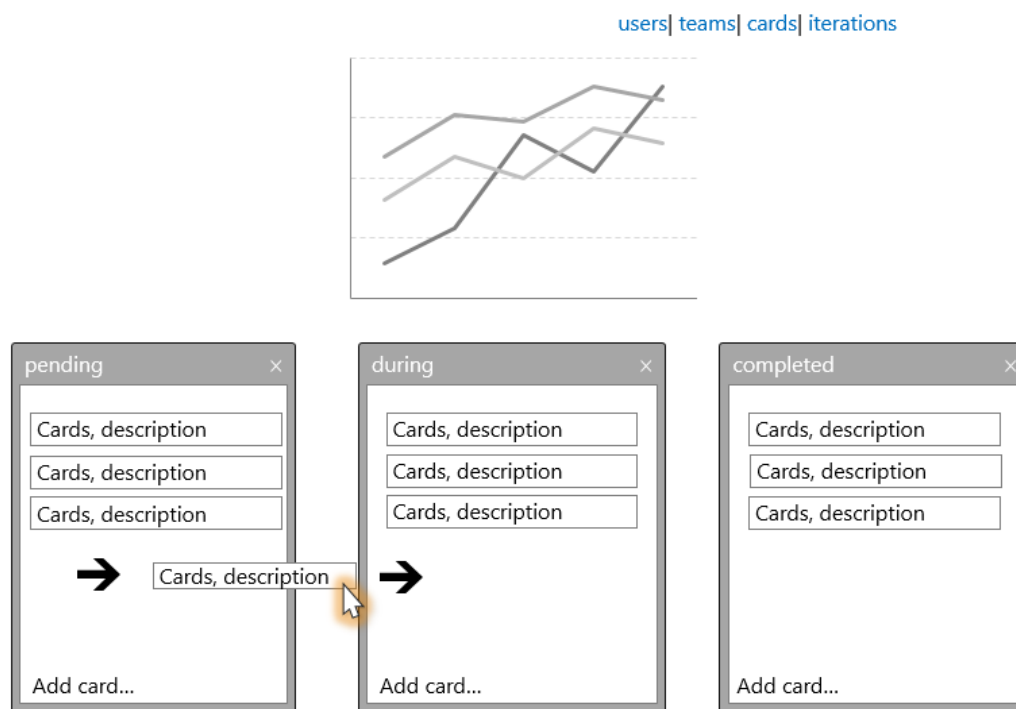


**Figure 9 – The side with the board is set and the schedule of hours spent**

The database was implemented using the Postgre SQL server DBMS, which meets the requirements for the module's functions. The structure of the module's database model is shown in Figure 10.

To work with the module, you need to create a list of developers and testers in order to later add participants to the created team. The next stage is to create iterations. The sprint creation form contains fields: name, data on the start and end of the iteration, as well as a section for filling in the number of working days in each working week (this number is determined according to the end date of the iteration).

The “Sprint List” page displays current iterations. The lines of the page contain information for quick identification of the desired iteration (name, data on the start and end of the iteration, as well as the status that the customer determines at the end of the sprint). The “Status” field is implemented as a drop-down list with the values failed (rejected), successful (successfully completed sprint).

The user is given the opportunity to create tasks within a specific user story. The task creation page contains the following fields: name, role, status, planned and actual execution time.

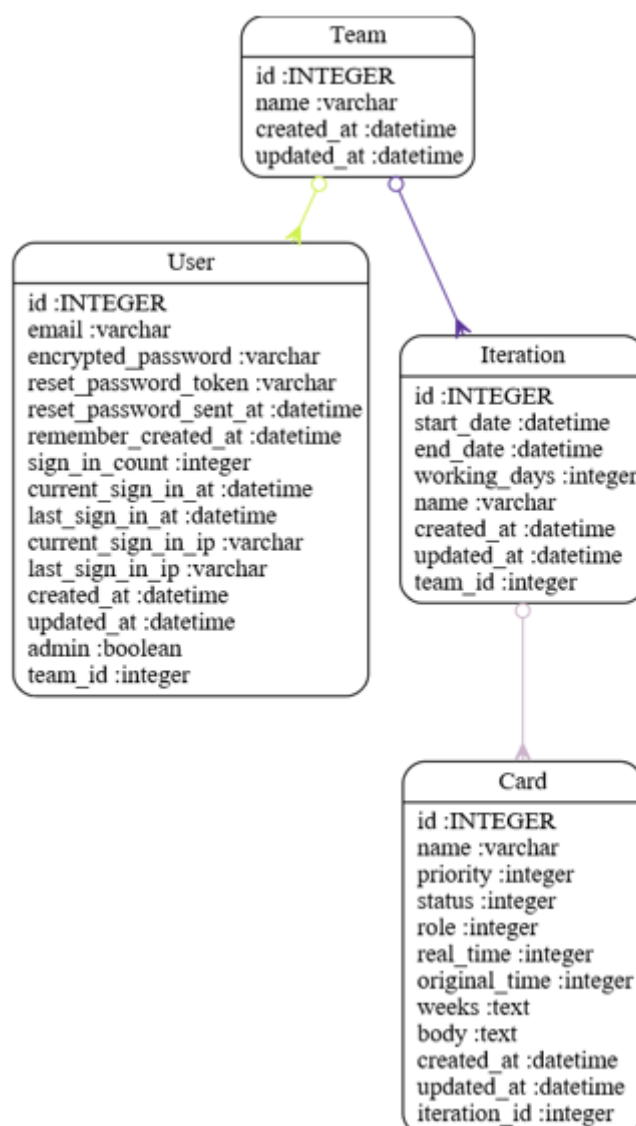


**Figure 10 – Module database model structure**

The Sprint Info page contains information about the current sprint: the number of participants and their roles in iterations, calculation of current resources, the amount of planned and completed work, focus factor, forecast for unplanned volume of tasks and error in estimation, task execution schedule, etc. The Team hard commitment column contains calculations for each working week for the indicators: benchmark, normalized volume of unplanned tasks, actual time spent at each checkpoint.

All indicators are calculated automatically after entering the number of working days for each team member. The maximum possible number of working days is determined at the time of sprint creation. The number of developers and testers corresponds to the size of the team. Based on the listed data, the user is given the opportunity to estimate resources in man-days.



The Hard Commitment table contains data for developers and testers to identify problem areas in the development process. The module also generates the value of the general indicator, which allows you to assess the progress of iterations as a whole.

The forecast of the volume of unplanned tasks and the excess of estimates is presented on the “Delta Hard Commitment” page (the corresponding calculations are displayed starting from the third iteration, when the module stores enough data for forecasting). These indicators can be analyzed in any phase of the sprint, which makes the model flexible and easily scalable.

The “Graphics” page presents the final task execution diagram, which uses the attributes and parameters described above. The user is given the opportunity to adjust the control dates to monitor the effectiveness of the design at any phase of the iteration.

An example of a task execution diagram obtained during the module testing process is shown in Fig. 11.

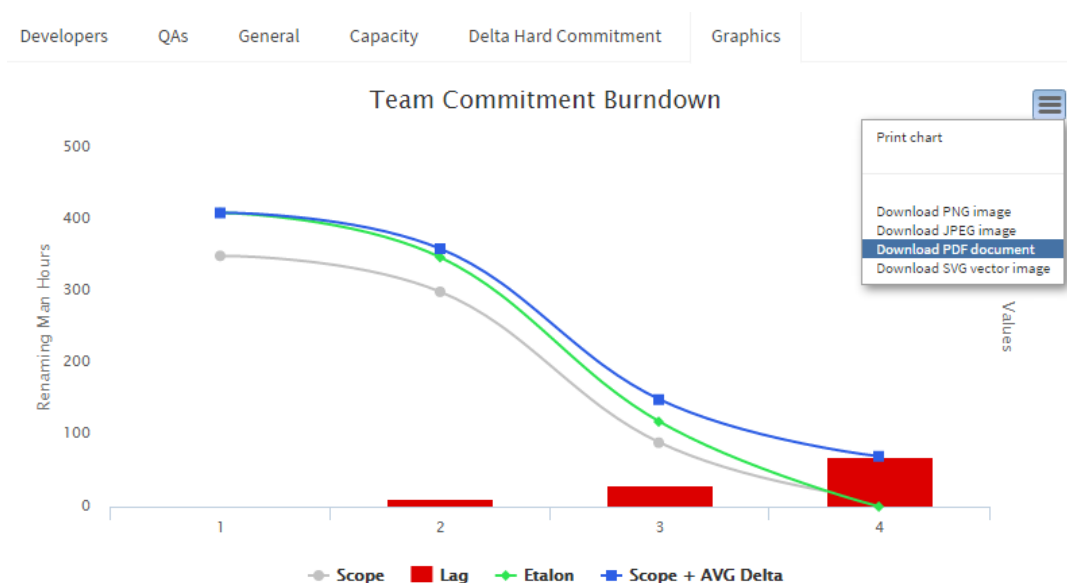


**Figure 11 – Example of a task flow chart**

In particular, the *Lag* indicator, presented in the diagram, clearly demonstrates the difference between the benchmark and how much the team is behind or ahead of schedule, taking into account the forecast for deviations.

The results obtained during the testing of the module indicate the prospects for its practical use.



## Conclusions.

The article presents the results of a study of IT project task performance indicators and prospects for increasing the efficiency of agile design based on the use of the proposed dynamic analysis module. Trends in IT project task performance using checkpoints were studied. The module was implemented software using the PostgreSQL server DBMS and the Spring framework. The results of testing the module are presented.

The use of this module in the field of agile project management will increase the efficiency of bottleneck analysis at the stages of planning, development and support of the design life cycle.

## References:

1. Brewer, J.L., & Dittman, K.C. *Methods of IT Project Management*. 4th ed. West Lafayette: Purdue University Press, 2022. 582 p.
2. Davidson, B. *Project Management Unlocked: An Easy Guide to Mastering Essential Principles*. 1st ed. It Was A Dark And Starry Night, 2024. 174 p.
3. Zhmayeva, Y., Udovenko, S., Chalaya, L. *Adaptivnoye prognozirovaniye optimal'nogo kolichestva resursov IT proyekta po metodologii Agile. Avtomatizirovannyye sistemy upravleniya i pribory avtomatiki*, (2015), 173, pp. 4 - 13.
4. Radiuk, P.M., Skrypnyk, T.K., & Karlechuk, D.T. *Applying Mental Models to Making Controlled Critically Safe Decisions in IT Project Management*. Herald of Khmelnytskyi National University. Technical Sciences, 2021, Vol. 301, no. 5, pp. 32–35. DOI: 10.31891/2307–5732–2021–301–5–32–35
5. Skrypnyk, T., Bahrii, R., Mazurets, O., Vozniuk, L., & Ilchyshyn, V. *Automated Process Distribution in IT Project Management*. Herald of Khmelnytskyi National University. Technical Sciences, 2025, No. 349(2), pp. 398–406. DOI: 10.31891/2307-5732-2025-349-5

Sent: 08.07.2026

© Udovenko S.G.



## CONTENTS

### Innovative engineering, technology and industry

- <https://www.sworldjournal.com/index.php/swj/article/view/swj38-01-010> 3  
**USE OF WILD RAW MATERIALS IN UKRAINIAN CUISINE**  
*Kyrpichenkova O.M., Tikhomirova E.S., Bilychenko A.A.*
- <https://www.sworldjournal.com/index.php/swj/article/view/swj38-01-015> 12  
**ENVIRONMENTAL OPTIMIZATION OF FURNITURE PRODUCTION:  
 REJECTION OF TOXIC ADHESIVES USING PRECISION  
 MECHANICAL JOINTS**  
*Khrypunov S.*
- <https://www.sworldjournal.com/index.php/swj/article/view/swj38-01-028> 24  
**ENERGY AUDIT EQUIPMENT TO DETERMINE ENERGY SAVING  
 POTENTIAL**  
*Prokopenko V.V., Zakladniy O.O.*
- <https://www.sworldjournal.com/index.php/swj/article/view/swj38-01-045> 39  
**IMPROVEMENT AND DEVELOPMENT OF METHODS FOR  
 PROCESSING THE RESULTS OF UNEQUALLY ACCURATE  
 MEASUREMENTS OF GEOMETRIC PARAMETERS**  
*Yefimenko N.A., , Podchashynskyi Yu.O., Chepiuk L.O.*
- <https://www.sworldjournal.com/index.php/swj/article/view/swj38-01-052> 49  
**PROJECTING PERFORMANCE INDICATORS FOR INNOVATIVE  
 GRILLING EQUIPMENT FOR THE RESTAURANT INDUSTRY**  
*Fedoriv V.M., Stechyshyn M. S. , Martynyuk A.V.  
 Liukhovets V.V. , Tkach B. O.*
- <https://www.sworldjournal.com/index.php/swj/article/view/swj38-01-071> 62  
**PLANT RAW BIOANTIOXIDANTS AS THE ANTI-AGING FACTORS**  
*Simakhina G.O., Naumenko N.V.*
- <https://www.sworldjournal.com/index.php/swj/article/view/swj38-01-077> 74  
**MANAGEMENT OF THE TOTAL POTENTIAL OF INNOVATIVE  
 DEVELOPMENT OF METALLURGICAL ENTERPRISES OF THE  
 EASTERN REGION OF UKRAINE**  
*Karpelianskyi A.*
- <https://www.sworldjournal.com/index.php/swj/article/view/swj38-01-083> 106  
**THE ROLE OF THE TOOL ROOM DEPARTMENT IN ENHANCING  
 THE EFFICIENCY OF ELECTRICAL ENGINEERING PRODUCTION**  
*Danyleichenko O.*



## Computer science, cybernetics and automatics

<https://www.sworldjournal.com/index.php/swj/article/view/swj38-01-009> 121

### CONTEXT-ORIENTED ARCHITECTURE FOR AI-ASSISTED DEVELOPMENT WITH MCP SERVERS AND AGENT SKILLS

*Kosovan V.M., Melnyk H.V., Melnyk V.S.*

<https://www.sworldjournal.com/index.php/swj/article/view/swj38-01-020> 129

### MODULE FOR DYNAMIC ANALYSIS OF IT PROJECT TASKS PERFORMANCE

*Udovenko S.G., Zatkhey V.A., Teslenko O.V.*

<https://www.sworldjournal.com/index.php/swj/article/view/swj38-01-035> 143

### RISK-ORIENTED MONITORING OF THE RESILIENCE OF WIRELESS SENSOR NETWORKS IN AGRICULTURAL PRODUCTION SYSTEMS UNDER A CHANGING OPERATIONAL CONTEXT

*Dyomina V. M. , Synyavina Y.V., Syry V.M.*

<https://www.sworldjournal.com/index.php/swj/article/view/swj38-01-040> 153

### DEVELOPMENT OF MATHEMATICAL SUPPORT FOR AN AUTOMATED SYSTEM FOR EARLY DETECTION OF STOCKING DURING DRILLING OF OIL AND GAS WELLS

*Shavranskyi M.V. , Tymofiiv M.R.*

<https://www.sworldjournal.com/index.php/swj/article/view/swj38-01-073> 166

### METHOD OF RECORD DEDUPLICATION IN HETEROGENEOUS DATA SOURCES BASED ON STRING SIMILARITY METRICS

*Bilousov D.*

<https://www.sworldjournal.com/index.php/swj/article/view/swj38-01-080> 181

### DEVELOPMENT OF A CLINICAL PROTOCOL MODELING AND VERIFICATION MODULE IN MEDICAL INFORMATION SYSTEMS BASED ON PETRI NETS

*Zhukova O. O.*

## Security systems in the modern world

<https://www.sworldjournal.com/index.php/swj/article/view/swj38-01-053> 193

### RECOGNITION OF DOS ATTACKS IN HETEROGENEOUS CRITICAL INFRASTRUCTURES VIA THE DEVELOPED SOFTWARE MODEL OF THE RECURRENT LSTM NEURAL NETWORK

*Pakhomova V. , Kolodiaznha M.*

<https://www.sworldjournal.com/index.php/swj/article/view/swj38-01-068> 202

### THE CONNECTION BETWEEN THE HYPOTHETICAL LAW OF INFORMATION CONSERVATION AND LABOR PROTECTION IN UKRAINE

*Abrakitov V.E.*



**Development of transport and transportation systems**

<https://www.sworldjournal.com/index.php/swj/article/view/swj38-01-067>

208

**THEORETICAL FOUNDATIONS OF AN INFORMATION-DIAGNOSTIC  
MODEL FOR PARAMETRIC RISK-BASED QUALITY AND RELIABILITY  
CONTROL OF AUTOMOTIVE PARTS**

*Beherskyi D.*



Scientific publication

*International periodic scientific journal*

# ScientificWorldJournal

Issue №38

Part 1

June 2026

Indexed in  
**INDEXCOPERNICUS**  
high impact factor (ICV: 73)

*Articles published in the author's edition*

*Academy of Economics named after D.A. Tsenov  
Bulgaria jointly with SWorld*

Signed: June 30, 2026

e-mail: [editor@sworldjournal.com](mailto:editor@sworldjournal.com)

site: [www.sworldjournal.com](http://www.sworldjournal.com)



[www.sworldjournal.com](http://www.sworldjournal.com)

