UDC 004.8:159.955:005.32 DOI: 10.20998/2413-3000.2026.12.5

V. ZIUZIUN

FORMATION THE BASIC CONCEPT OF A METHOD FOR MANAGING THE RISK OF TIME LOSSES IN IT PROJECTS BASED ON MODELING THE TEAM’S COGNITIVE PROFILE

This scholarly work addresses the pressing problem of managing the risks of time losses in IT projects that arise from cognitive interruptions experienced by specialists while performing complex tasks. The author notes that contemporary Agile methodologies create an inherent conflict between the need for deep concentration (the «flow state») and the intensity of team communications. This conflict leads to the accumulation of cognitive debt – a latent risk reflecting a reduced ability of the team to return to productive work after interruptions. The purpose of the study is to develop a method that treats the team’s cognitive profile as a dynamic project resource for the quantitative forecasting of deadline-failure risks. The scientific novelty of the work lies in the introduction of formalized metrics for cognitive debt and interruption cost, which make it possible to assess the systemic consequences of cognitive losses along the critical path of the project dependency graph. The proposed method is based on modeling an individual specialist’s cognitive viscosity and calculating the reconcentration time required to restore the task’s mental models. The mathematical core of the method transforms a planned work schedule into a probabilistic model in which each interruption acts as a factor that extends lead time. A key element is the determination of interruption cost, which accounts not only for the personal losses of an individual developer but also for the cascading idle time of all dependent team members. The practical significance of the study lies in the possibility of integrating these models into IT project management systems for adaptive regulation of communications. Based on probabilistic risk assessment, the system can propose preventive measures such as a «cognitive quarantine» (blocking non-priority notifications) or dynamic sprint or Agile process rescheduling. In summary, the method enables a shift from reactive acknowledgment of delays to proactive management of cognitive resources. This provides a scientific foundation for protecting developers’ workspaces, minimizing cascading risks, and increasing the overall predictability of delivery timelines in cognitively intensive projects.

Keywords: intelligent filtering, cognitive viscosity, flow state, Agile team, mathematical modeling, team effectiveness, autonomy buffer, dependency graph, Agile process progress, communication in IT.

В. І. ЗЮЗЮН

ФОРМУВАННЯ БАЗОВОГО КОНЦЕПТУ МЕТОДУ УПРАВЛІННЯ РИЗИКАМИ ЧАСОВИХ ВТРАТ В ІТ-ПРОЄКТАХ НА ОСНОВІ МОДЕЛЮВАННЯ КОГНІТИВНОГО ПРОФІЛЮ КОМАНДИ

Наукова праця присвячена вирішенню актуальної проблеми управління ризиками часових втрат в ІТ-проєктах, що виникають через когнітивні переривання фахівців під час виконання складних завдань. Автор зазначає, що сучасні Agile-методології створюють конфлікт між потребою в глибокій концентрації («стані потоку») та інтенсивністю командних комунікацій. Це призводить до накопичення «когнітивного боргу» – прихованого ризику, що відображає зниження здатності команди повертатися до продуктивної роботи після відволікань. Метою дослідження є розробка методу, який розглядає когнітивний профіль команди як динамічний ресурс проєкту для кількісного прогнозування ризиків зриву дедлайнів. Наукова новизна полягає у впровадженні формалізованих метрик когнітивного боргу та вартості переривання, що дозволяють оцінити системні наслідки когнітивних втрат на критичному шляху графа проєктних залежностей. Запропонований метод базується на моделюванні індивідуальної «когнітивної в’язкості» фахівця та розрахунку часу реконцентрації, необхідного для відновлення ментальних моделей задачі. Математичне ядро методу трансформує плановий графік робіт у ймовірнісну модель, де кожне переривання виступає фактором подовження Lead Time. Ключовим елементом є визначення вартості переривання, яка враховує не лише особисті втрати розробника, а й каскадні простої всіх залежних учасників команди. Практичне значення роботи полягає у можливості інтеграції цих моделей у системи управління ІТ-проєктами для адаптивного регулювання комунікацій. На основі ймовірнісної оцінки ризику система може пропонувати превентивні заходи, такі як «когнітивний карантин» (блокування непріоритетних сповіщень) або динамічне перепланування спринту (Agile-процесу). Узагальнюючи, метод дозволяє перейти від реактивного констатування затримок до проактивного управління когнітивним ресурсом. Це забезпечує наукове підґрунтя для захисту робочого простору розробників, мінімізації каскадних ризиків та підвищення загальної передбачуваності термінів реалізації інтелектуально містких проєктів.

Ключові слова: інтелектуальна фільтрація, когнітивна в’язкість, стан потоку, Agile-команда, математичне моделювання, командна ефективність, запас автономності, граф залежностей, прогрес Agile-процесів, комунікація в ІТ.

Introduction and problem statement. In modern software engineering, the efficiency of developing complex systems largely depends on specialists’ ability to sustain prolonged, uninterrupted cognitive concentration, corresponding to a state of deep intellectual immersion, or the «flow state». At the same time, Agile methodologies presuppose a high intensity of communication, which creates a fundamental contradiction between individual focus and the demands of team synchronization.

The problem is exacerbated by the fact that software development is a process of constructing complex mental models; their disruption by unregulated notifications leads to the emergence of cognitive viscosity – a condition in which the time required to restore task context becomes critically large. Existing communication management mechanisms are predominantly based on static rules and do not account for the dynamics of an individual specialist’s cognitive state or the phase of the sprint (Agile process). As a result, decisions to interrupt work are made without considering their actual cost to the project as a whole.

Consequently, a cumulative effect of cognitive debt arises, reflecting a latent risk of time losses and leading to a degradation of the team’s ability to rapidly return to productive work. Since traditional monitoring tools typically register only the fact of missed deadlines, it becomes evident that there is a need to formalize a quantitative relationship between an individual developer's cognitive losses and team-level performance metrics. This, in turn, necessitates the development of adaptive methods capable of identifying cascading idle times and forecasting risks along the critical path of the project dependency graph before deadlines are actually violated.

Analysis of prior research and publications. An analysis of the scholarly literature in the field of managing cognitive concentration and communications in IT teams indicates a strong research interest in finding a balance between an individual developer's productivity and the overall dynamics of a project. A significant contribution to understanding the nature of interruptions was made by C. Parnin and S. Rugaber, whose works focus on programmers' need for contextual information after memory disruptions and on the design of interactive environments that support sustained concentration [1; 2]. Their studies demonstrate that frequent interruptions not only consume clock time but also deplete a developer's short-term memory, thereby extending the «warm-up» period required before productive work can resume.

Experimental results show that each unregulated interruption causes substantial time losses, averaging 10–15 minutes needed to reconstruct the task's mental models and re-enter the flow state.

In parallel with cognitive research, an active line of work has emerged on the development of intelligent notification-filtering systems. Contemporary approaches rely on machine learning methods to analyze user preferences, as exemplified by the PrefMiner system and other models of adaptive information flow management [3; 4; 5]. In this context, particular attention should be given to study [6], which introduces and investigates the concept of cognitive viscosity for optimizing the management of Agile teams.

The mathematical foundation for modeling structural interrelationships in such systems traditionally relies on graph theory [7; 8], while the analysis of the intensity of incoming information flows is carried out using the apparatus of queueing theory [9]. To address complex optimization problems, researchers often employ multicriteria methods and stochastic algorithms [10; 11; 12]. In particular, recent work in the field of Agile management has focused on the application of artificial intelligence to forecasting team productivity and managing large groups of developers [13; 14].

Despite the substantial body of publications, most existing approaches remain limited in scope, as they treat the specialist as an isolated unit. The cascading impact of individual cognitive losses on the critical path of the entire project and on the risks of deadline failure is often overlooked. The absence of a formalized link between the team's dynamic cognitive debt and IT project performance metrics underscores the need to develop new integrated methods for managing the risks of time losses.

The purpose of this work is to develop a method for managing the risks of time losses in IT projects based on modeling the team's cognitive profile as a dynamic resource. The method provides quantitative identification, assessment, and forecasting of deadline-failure risks by integrating specialists' cognitive characteristics, cognitive debt metrics, and interruption cost with a model of project dependencies and the critical path.

Main part of the study. The proposed method for managing the risks of time losses is based on a systems approach that views the development process as a dynamic interaction between individual cognitive resources and the project's structural dependencies. The core idea is to move from reactive management, which merely records delays after they occur, to a proactive intelligent monitoring system that relies on leading indicators of team state.

The method's concept encompasses an end-to-end process that begins with formalizing the human factor through cognitive profiles and culminates in automated, adaptive decision-making aimed at stabilizing the work schedule. Within this framework, each disruptive influence – such as unplanned communication or meetings – is interpreted as an event that generates an increase in cognitive debt and triggers a cascading interruption cost at the team level.

The mathematical core of the method enables the transformation of developers' subjective experiences into quantitative metrics, which are then integrated with the project task graph to compute a probabilistic deadline model. In this way, the method makes it possible to create a unified information space in which the project's technological constraints are synchronized with the cognitive capacities of specialists, allowing the manager to operate with precise data on the current sprint's resilience margin (Agile process).

Block A. Forming the team's cognitive profile as an input parameter of the risk management method. The formation of the team's cognitive profile is a foundational stage of the proposed method, as it enables the translation of the abstract notion of the "human factor" into the domain of measurable project indicators. Within this study, the cognitive profile is treated as a dynamic resource that reflects the collective's ability to perform cognitively intensive tasks without a degradation of overall productivity under the influence of external stimuli.

Cognitive debt is a formalized indicator of accumulated risk of time losses that increases as a result of a series of unregulated interruptions and fragmentation of the work process, leading to a degradation of a specialist's ability to quickly return to a state of deep concentration (the «flow state»).

The introduction of this term makes it possible to systematically assess the «cost» of each message or meeting through the lens of time losses along the critical path. When a specialist is exposed to an excessive number of external requests, their cognitive debt grows, giving rise to the phenomenon of cognitive viscosity [6] – a condition in which the time required to restore task context becomes critically large, thereby creating prerequisites for deadline violations.

To ensure that the cognitive profile becomes an effective risk management instrument, the method provides its enrichment along several key dimensions:

- *Aggregation of individual characteristics and roles.* The approach accounts not only for specialists' psychophysiological parameters, such as reconcentration speed, but also for the specifics of their roles within the project structure.
- *Monitoring of dynamic state.* The profile is continuously updated based on metrics of uninterrupted work and interruption frequency, enabling the real-time identification of zones where focus declines to critical levels.
- *Synchronization with structural dependencies.* Cognitive indicators are mapped onto the project task graph, making it possible to identify nodes at which the individual losses of a single developer may trigger cascading delays across the entire sprint (Agile process).

Thus, the cognitive profile functions not merely as a descriptive characteristic but as a controllable variable that provides a scientific basis for forecasting time losses. The availability of such a detailed profile enables a shift from general monitoring of team condition to the direct identification of specific risks, implemented through the calculation of quantitative metrics for the cost of each individual interruption.

Block B. Identifying cognitive risks through cognitive debt and interruption cost metrics. At the next stage of the method's implementation, the focus shifts from a descriptive cognitive profile to the direct identification of time-loss risks. A major challenge in Agile projects is that cognitive losses are typically «invisible» until a deadline is actually missed.

To address this issue, the method introduces a system of quantitative assessment based on two key metrics: *cognitive debt and interruption cost.* These indicators make it possible to transform subjective factors of intellectual fatigue and loss of focus into formalized data suitable for algorithmic analysis and optimization.

A central role in risk identification is played by the dynamics of cognitive debt, which within the method is interpreted as an accumulated risk of time losses. It reflects the systemic effect of workday fragmentation: each new interruption not only consumes time but also increases a specialist's level of cognitive viscosity, making every subsequent return to work more difficult. At the same time, the method accounts for the cognitive resource's capacity for self-recovery during periods of uninterrupted concentration, enabling the manager to observe the team's actual productivity «balance» at any point in the sprint (Agile process).

In parallel with the cumulative effect of cognitive debt, the method employs the interruption cost metric, which serves as a tool for assessing the immediate impact of managerial decisions on productivity:

- *Quantitative impact assessment.* The interruption cost allows determining how much total time the team would lose if a specific specialist at a critical project node were distracted by a non-priority message.
- *Forecasting systemic consequences.* By integrating with the project dependency graph model, this metric illustrates how even a brief interruption of a single developer can trigger cascading idle time for other team members through task structural dependencies.
- *Communication optimization.* Formalizing the interruption cost provides a basis for establishing adaptive notification-filtering rules, where the priority of incoming information is aligned with its potential harm to the specialist's current focus state.

Such a system of metrics enables a move beyond intuitive management toward the precise calculation of deadline-violation risks. Understanding the quantitative cost of each interruption and the accumulated level of cognitive debt sets the stage for the next step: mathematically modeling individual losses, which allows for a detailed reconstruction of the concentration recovery timeline for each specific developer – a process that will be examined further in the study.

Moreover, within the proposed method, risk identification is not limited to merely recording the moment of an interruption. It also encompasses analysis of the chain reaction triggered by time losses. When the risk of time loss materializes as a primary threat factor, it acts as a trigger for a series of secondary risks [15] that can destabilize the project at multiple levels (see Table 1).

The introduction *of cognitive debt and interruption* cost metrics allows these hidden threats to be detected early, at the stage of their emergence. Thus, the formalization of cognitive factors becomes a tool not only for maintaining the current development pace but also for ensuring overall stability and the quality of the intellectual product within an Agile environment.

Block C. Mathematical modeling of cognitive debt dynamics and specialist reconcentration. To translate the concept of cognitive losses into the domain of quantitative management, the method formalizes a specialist's cognitive state through a dynamic cognitive debt indicator. This approach accounts not only for the immediate time lost to task switching but also for the cumulative effect of reduced concentration capacity (*cognitive viscosity*).

1. *Formalizing the accumulation of cognitive debt.* The developer's cognitive debt (D_C) at time t is proposed to be considered as an integral quantity that grows due to a series of unregulated interruptions and work process fragmentation. The increase in debt for each interruption is defined as:

$$\Delta D_C = \sigma \cdot t_{flow} \cdot f(H), \tag{1}$$

where σ – the specialist's cognitive viscosity coefficient;

$f(H)$ – a function representing the complexity of the current intellectual task context;

t_{flow} – the duration of uninterrupted intellectual immersion (flow state) prior to the disruptive event.

Using σ as a multiplier allows modeling an individual specialist's sensitivity: the higher the cognitive viscosity, the faster the debt accumulates with each subsequent message or meeting.

This formula enables a quantitative assessment of the «strength» of destructive impact on the workflow. It shows that the longer a specialist has been in a state of deep focus (t_{flow}), the more painful and costly an interruption becomes, as it disrupts a more complex mental construct. Applying the σ coefficient allows managers to differentiate tasks: for creative architectural decisions, this value will be significantly higher than for routine test writing, enabling a more accurate prediction of cognitive debt growth depending on the type of work.

Table 1 – Identification of secondary risks for the project in the event that time-loss risk materializes as the primary threat factor

Risk category	**Risk name**	**Description (impact on the project)**
Structural-temporal	**Risk of cascading delays**	Due to the structural interdependence of tasks in the project graph, a delay at a single critical node caused by a cognitive interruption leads to idle time for related specialists, exponentially increasing the overall lead time.
	Risk of schedule slippage	The aggregated cognitive viscosity at critical nodes in the project graph results in an irreversible shift of the final deadlines for the sprint, Agile process, or release.
Technical and qualitative	**Risk of quality degradation**	In an effort to compensate for lost time, specialists may consciously or unconsciously simplify solutions, leading to increased technical debt and a higher number of code defects.
	Risk of loss of contextual knowledge	Frequent interruptions disrupt complex mental models, forcing additional time (10–15 minutes) to be spent on restoring context and re-examining sections of code.
Team-based and managerial	**Risk of disruption in team synchronization**	A high level of cognitive debt drives team members to avoid communication in order to protect their focus, creating information gaps and architectural errors.
	Risk of professional burnout	Working under constant «cognitive viscosity» and a deficit of «flow state» leads to emotional exhaustion and risks the loss of key experts.
	Risk of ineffective resource planning	Relying on static management rules without accounting for the dynamic cognitive state results in a misallocation of workload at critical stages of the project.

2. *Determining re-concentration (recovery) time.* According to experimental data, each interruption causes a loss of time needed to restore mental models. Within this method, the re-concentration time (τ_{rec}) is modeled as a function of the current level of cognitive debt:

$$\tau_{rec} = t_{base} \cdot (1 + ln(1 + \sigma \cdot D_C)), \qquad (2)$$

where t_{base} is the baseline time to return to a «flow state» (typically 10–15 minutes), and the logarithmic dependence reflects the nonlinear increase in difficulty when restoring complex contexts as the day becomes more fragmented.

The mathematical logic of this model is based on logarithmic productivity degradation: the first few interruptions are relatively easy to handle, but after reaching a certain threshold, the accumulated debt D_C begins to exponentially complicate the return to work. The practical application of this formula lies in identifying the «point of no return», after which it is more effective for a specialist to switch tasks or take an extended break to restore cognitive resources rather than continue the current task.

The mathematical interpretation of the «point of no return» within this model is based on analyzing the limiting behavior of τ_{rec} relative to the specialist's remaining available working time. Because the logarithmic dependence implies an accelerated increase in the complexity of mental model recovery at high values of D_C, a moment emerges at which the cumulative costs of re-concentration and the compensation of cognitive viscosity σ exceed the potential productive return from continuing work on the current intellectual task.

From a managerial perspective, the point of no return is identified under the condition that:

$$\tau_{rec} > \Delta T_{available}\,, \qquad (3)$$

where $\Delta T_{available}$ is denotes the time interval until the next scheduled interruption or the end of the working session. Reaching this threshold signals to management that the degree of day fragmentation has led to a critical degradation of concentration, at which any attempt to re-enter the «flow state» becomes economically unjustifiable due to the high cost of interruption (CoI) and the risk of cascading idle time across the entire team. Consequently, equation 3 serves not only as a means of estimating delays but also as a trigger for activating «cognitive quarantine» measures or for initiating urgent task rescheduling.

3. *Modeling the self-recovery effect of cognitive resources.* The method also accounts for the ability of cognitive resources to self-recover during periods of sustained, uninterrupted focus. If a specialist works without external distractions, the level of D_C gradually decreases, reflecting the stabilization of the work context and a reduced likelihood of quality degradation risks.

The calculated individual metrics allow determining the «cost» of an interruption for a specific developer. However, in Agile development, such losses have a cascading effect due to the structural interdependencies of tasks. This necessitates aggregating the cost of interruptions along the critical path of the project dependency graph, where individual losses are transformed into a team-wide risk metric for missed deadlines.

Overall, it can be stated that the proposed metrics D_C and CoI (the latter being responsible for estimating the cost of interruptions, with its specific properties discussed later in the study) enable early detection of the following derivative risks:

- *Burnout risk.* Monitored through a consistently high level of D_C that fails to recover over several days.

- *Quality degradation risk.* Correlates with increasing cognitive viscosity σ at critical nodes, when a specialist begins to «simplify» tasks to save time.

Block D. Aggregating the cost of interruptions along the critical path of the project dependency graph. After determining the individual re-concentration time τ_{rec} for each specialist, it is necessary to assess the systemic impact of these losses on the entire project. Within the proposed method, the structure of an IT project is modeled using graph theory, where tasks are represented as nodes and the logical dependencies between them as edges.

1. Integrating cognitive losses into graph parameters. The traditional task duration on the critical path (t_i) is augmented with a dynamic component representing cognitive losses. Consequently, the actual task execution time (t_{fact}) becomes a stochastic variable dependent on the number of interruptions and the accumulated cognitive debt:

$$t_{fact,i} = t_{plan,i} + \sum_{j=1}^{n} \tau_{rec,j}(D_C, \sigma), \qquad (4)$$

where $t_{plan,i}$ is the planned duration of the $\mathrm{i} - th$ task without considering destructive impacts;

$\sum \tau_{rec,j}$ is the total time spent restoring focus after each interruption during the task.

This model transforms a deterministic Gantt schedule into a probabilistic model, where each interruption acts as a factor extending task duration. It allows for the calculation of a realistic lead time, taking into account not only the code volume but also the communication «noise» within the development environment. In essence, formula (3) reflects the hidden time spent «warming up» cognitive resources after each meeting or message.

2. *Determining the cost of interruption (CoI).* In this method, the cost of an interruption is not treated as a fixed value but as an increase in the overall project lead time. If an interruption occurs to a specialist working on a critical node in the graph (Critical Path), the CoI equals the sum of the re-concentration time for that specialist plus the idle time of all dependent team members:

$$CoI_{total} = \tau_{rec,i} + \sum_{k \epsilon S} \Delta t_{idle,k}, \qquad (5)$$

where S is the set of successor tasks whose execution is delayed due to the cascading idle effect at the current node.

Formula (4) is crucial for identifying cascading risks: it demonstrates that the cost of interrupting a specialist on the critical path is not limited to their personal time but also includes the cumulative delay of all subsequent development stages. This provides a mathematical rationale for implementing strict «quiet» or «do not disturb» policies for key developers, as their distraction multiplies idle time across the entire team due to structural dependencies in the project graph.

3. *Cognitive viscosity as a factor in cascading delay risk.* Applying this coefficient allows the identification of the most «vulnerable» nodes in the project. High cognitive viscosity on the critical path means that even a minor interruption in terms of message volume can trigger a chain reaction of delays (Dependency risk). This creates information gaps and architectural errors, as team members begin avoiding communication to protect their own focus.

Aggregating the cost of interruptions across the project graph enables managers to obtain a real-time view of lead time. However, for strategic decision-making, it is not enough to merely report current losses; it is also necessary to forecast the likelihood of successfully completing a sprint (or Agile process). Accordingly, forecasting the risk of deadline violations should be considered based on the team's aggregated cognitive viscosity, which can serve as the foundation for a probabilistic model of deadline estimation in future research.

Block E. Forecasting deadline violation risk based on the team's aggregated cognitive viscosity. The final stage of the method involves moving from a static graph analysis to dynamic forecasting of the probability of completing the project within the established deadline (t_{dead}). Since the actual task duration $(t_{fact,i})$ is a stochastic variable dependent on communication intensity and accumulated cognitive debt, risk forecasting is performed through iterative modeling of the critical path state.

1. *Calculation of aggregated cognitive viscosity* $(\sum \sigma)$. The method involves computing the cumulative impact of «viscosity» on the project's critical sections. The higher this value at nodes with strong interdependencies, the greater the likelihood of a cascading delay (Dependency Risk).

2. *Probabilistic deadline assessment.* The risk of deadline violation $(R_{violation})$ is calculated as the probability that the total duration of tasks on the critical path, accounting for projected interruptions, will exceed the planned time:

$$(t_{total} > t_{dead}) = f(\sum \tau_{rec}, \sum \sigma, D_C), \qquad (6)$$

where t_{dead} is the expected lead time, taking into account the current level of the team's cognitive debt and the dynamics of its recovery.

The proposed probabilistic model enables the assessment of a sprint's (or Agile process's) «safety margin». Instead of a binary «on track / off track» evaluation, the manager receives a dynamic probability distribution curve that accounts for the team's current fatigue level (D_C) and the density of upcoming communications. This allows the automatic notification system to be fine-tuned: if the probability P exceeds a critical threshold (e.g., 30%), the system automatically recommends cognitive quarantine measures to stabilize the schedule.

3. *Adaptive decision-making.* Based on the forecast, the decision support system can recommend the following preventive measures:

- «Cognitive quarantine». Temporarily blocking all non-priority notifications for specialists on the critical path if D_C exceeds a critical threshold.
- Dynamic rescheduling. Adjusting task priorities to reduce cognitive viscosity at nodes with the highest concentration of logical dependencies.
- Velocity adjustment. Adapting planned metrics for upcoming sprints (Agile processes) based on the actual cost of interruptions (CoI), recorded during the current period.

Thus, aggregating cognitive metrics allows the transformation of the team's subjective sense of «overload» into a concrete percentage-based risk indicator, providing a scientific basis for preventive management in Agile development.

The proposed risk management method can be characterized as a proactive intelligent management system that shifts from reactive delay correction to preventive modeling. Unlike traditional Agile monitoring tools (e.g., burn-down charts), this method works with «leading indicators» – detecting drops in concentration before they manifest as actual idle time.

The method effectively creates a digital twin of the team's cognitive activity, where each message in a messenger or unplanned meeting is assigned a «cost» in units of lead time. This enables managers not merely to demand results, but to scientifically justify the protection of developers' workspaces, minimizing cascading risks along the project's critical path.

For practical implementation of the method in an Agile process, the following step-by-step algorithm is proposed:

Step 1. Profile formation. Define baseline cognitive viscosity coefficients (σ) or each specialist according to the type of tasks they handle (architectural, research, routine).

Step 2. Monitoring incoming flows. Continuously record the frequency and duration of interruptions (communications) to calculate the current cognitive debt (D_C).

Step 3. Quantitative loss assessment. Calculate the re-concentration time (τ_{rec}) for specialists based on their current state and the complexity of the context.

Step 4. Critical path analysis. Integrate the observed delays into the project dependency graph to compute the actual cost of interruptions (CoI_{total}).

Step 5. Probabilistic forecasting. Estimate the risk of deadline violation ($R_{violation}$) through iterative modeling of the remaining work execution.

Step 6. Adaptive response. Apply automated measures such as initiating a «quiet mode», redistributing workload, or adjusting the sprint scope (Agile process).

To illustrate a potential application of the method, consider a scenario in an Agile team working on a critical security module.

Situational task. A Senior developer is working on a task on the project's critical path. The planned execution time is 4 hours ($t_{plan} = 240\ min$). During this period, the developer receives two interruptions via Slack from adjacent teams.

Initial data of the specialist: cognitive viscosity coefficient $\sigma = 0{,}8$ (high architectural complexity); baseline recovery time $t_{base} = 15\ min$; current cognitive debt $D_{C_{start}} = 0$.

Loss calculation:

1.1. First interruption → Occurred after 60 minutes of deep focus ($t_{flow} = 60$).

1.2. Cognitive debt increase →
$\Delta D_{C_1} = 0{,}8 \cdot 60 = 48\ min$

1.3. Re-concentration time →
$\tau_{rec_1} = 15 \cdot (1 + ln(1 + 0.8 \cdot 48)) \approx 70\ min.$

2.1. Second interruption → Occurred 30 minutes after recovery.

2.2. Cognitive debt increase →
$\Delta D_{C_2} = 0{,}8 \cdot 30 = 24\ min\ (total\ \ D_C = 72\ min).$

2.3. Re-concentration time →
$\tau_{rec_2} = 15 \cdot (1 + ln(1 + 0{,}8 \cdot 72)) \approx 76\ min.$

3. Systemic impact on the project. Actual task duration and systemic impact (t_{fact}) increased from 240 minutes to →
$240 + 70 + 76 = 386\ min$ (approximately 6.5 hours).

Since the developer is on a critical node, these 146 minutes of delay automatically translate into idle time (t_{idle}) for three testers waiting for the module release. The total cost of interruption (CoI_{total}) for the team is:
$146\ min\ (developer) + 3 \cdot 146\ min\ (testers) =$
$= 584\ person - min\ (almost\ 10\ hours\ of\ work).$

Outcome. By identifying this risk through formula (5), the decision support system records that the probability of a sprint (Agile process) deadline violation has risen to 45%. The manager receives a recommendation to activate a «cognitive quarantine» – a complete blocking of non-critical communications for this node until the end of the workday.

Conclusions. The proposed study presents a comprehensive scientific concept that offers a qualitatively new perspective on efficiency in modern software engineering. A key advantage of the developed method is the shift from viewing a specialist as a static resource with fixed productivity to a dynamic «cognitive profile» model that accounts for mental fatigue, contextual complexity, and the cumulative effect of distractions.

The study convincingly demonstrates that traditional Agile metrics only capture actual delays, whereas the use of cognitive debt and cost-of-interruption metrics allows management to operate with leading risk indicators. This transforms the management process from reactive error correction to proactive modeling of the «safety margin» of each work cycle.

The scientific novelty lies in formalizing cascading delays through cognitive viscosity on the critical path of the project graph, providing a solid foundation for developing intelligent decision support systems. Implementing such tools allows managers to justify measures like «cognitive quarantine» and dynamic rescheduling, protecting the most valuable asset of an IT project – the developers' state of deep concentration.

The practical significance of the method is confirmed by its ability to quantitatively assess the systemic impact of communication noise, which is critical for cognitively intensive projects with complex logical dependencies. Thus, the study not only addresses the urgent issue of minimizing time losses but also contributes to the development of a mathematically precise approach to human resource management in the digital age, ensuring high predictability of intellectual product delivery without risking team burnout.

References

1. Parnin, C., Görg, C., & Rugaber, S. (2010). CodePad: interactive spaces for maintaining concentration in programming environments. *In Proceedings of the 5th international symposium on Software visualization (SOFTVIS '10). Association for Computing Machinery,* New York, NY, USA, pp. 15–24. Available at: https://doi.org/10.1145/1879211.1879217(Accessed 14 January 2026)
2. Parnin, C., & Rugaber, S. (2012). Programmer information needs after memory failure. *2012 20th IEEE International Conference on Program Comprehension (ICPC)*, Passau, Germany, pp. 123-132, Available at: https://doi.org/10.1109/ICPC.2012.6240479 (Accessed 14 January 2026)
3. Mehrotra, A., Hendley, R., & Musolesi, M. (2017). Interpretable Machine Learning for Mobile Notification Management: An Overview of PrefMiner. *GetMobile: Mobile Comp. and Comm*, 21 (2), pp. 35–38. Available at: https://doi.org/10.1145/3131214.3131225 (Accessed 15 January 2026)
4. Mehrotra, A., & Musolesi, M. (2018). Intelligent Notification Systems: A Survey of the State of the Art and Research Challenges. *Computer Science. Human-Computer Interaction.* https://doi.org/10.48550/arXiv.1711.10171 (Accessed 15 January 2026)
5. Bazinette, V., Cohen, N. H., & Ebling M. R., et al. (2001). An Intelligent Notification System. *IBM Research Report.* Available at: https://www.researchgate.net/publication/228970929_An_intelligent_notification_system (Accessed 14 January 2026)
6. Ziuziun, V. (2026). Substantiation of the Concept of Cognitive Viscosity for Optimizing AGILE Team Management. *Proceedings of the 5th International Scientific and Practical Conference «Scientific Innovation: Theoretical Insights and Practical Impacts»,* January 19–21, 2026, Naples, Italy, P. 69-73. Available at: https://doi.org/10.70286/EOSS-19.01.2026.001.69-73 (Accessed 14 January 2026)
7. Needham M., & Hodler, A. E. (2019). Graph Algorithms. Practical Examples in Apache Spark and Neo4j. Available at: https://web4.ensiie.fr/~stefania.dumbrava/GraphAlgorithms.pdf (Accessed 15 January 2026)
8. Wilson, R. J. (1996). Introduction to Graph Theory. Available at: https://webhomes.maths.ed.ac.uk/~v1ranick/papers/wilsongraph.pdf (Accessed 14 January 2026)
9. Carzaniga, A. (2025). Elements of Queuing Theory. Available at: https://www.inf.usi.ch/carzaniga/edu/adv-ntw/queuing-theory-notes.pdf (Accessed 15 January 2026)
10. Sharma, S., & Kumar, V. A. (2022). Comprehensive Review on Multi-objective Optimization Techniques: Past, Present and Future. *Arch Computat Methods.* Eng 29, 5605–5633. Available at: https://doi.org/10.1007/s11831-022-09778-9 (Accessed 15 January 2026)
11. Sharifi, M. R., Akbarifard, S., & Qaderi, K. et al. (2021). A new optimization algorithm to solve multi-objective problems. Sci Rep 11, 20326. Available at: https://doi.org/10.1038/s41598-021-99617-x (Accessed 14 January 2026)
12. Lapeyre, B. (2021). Monte Carlo Methods and Stochastic Algorithms. Available at: https://cermics.enpc.fr/~bl/monte-carlo/poly.pdf (Accessed 15 January 2026)
13. Ziuziun, V., Kulkovets, V., & Parasiuk, L. (2024). Development of a Decision Support Information System for Managing Large Agile Teams in IT Projects. *Collection of Scientific Papers of Admiral Makarov National University of Shipbuilding*, 4(497), 166-172, Available at: https://doi.org/10.15589/znp2024.4(497).23 (Accessed 15 January 2026)
14. Ziuziun, V. & Petrenko, N. (2025). AI-Enhanced System Design for Agile Sprint Management and Velocity Prediction. *2025 IEEE 5th International Conference on Smart Information Systems and Technologies (SIST)*, Astana, Kazakhstan, pp. 1-6, Available at: https://doi.org/10.1109/SIST61657.2025.11139278 (Accessed 15 January 2026)
15. Ziuziun, V. Exploring the concept of derivative risks arising from external influences in the context of business operations and their strategic stability. *Bulletin of the National Technical University «KhPI». Series: Strategic management, portfolio, program and project management,* 2024, 2(9), pp. 27-34. Available at: https://doi.org/10.20998/2413-3000.2024.9.4 (Accessed 15 January 2026)

Received (надійшла) 28.01.2026

Відомості про автора / About the Author

Зюзюн Вадим Ігорович (Ziuziun Vadym Ihorovych) – кандидат технічних наук (PhD), доцент, доцент кафедри технологій управління, Київський національний університет імені Тараса Шевченка, м. Київ, Україна; e-mail: vadym.ziuziun@knu.ua; ORCID: https://orcid.org/0000-0001-6566-8798.