Scrum-Driven Task Management for University IT Governance: A Multi-Role Usability Evaluation

Manuscript received April 16, 2026; revised June 3, 2026

Muhammad Faried Saputra
Department of Informatics Engineering Education
Universitas Sebelas Maret
Surakarta, Indonesia
faried_12@student.uns.ac.id

Aris Budianto
Department of Informatics Engineering Education
Universitas Sebelas Maret
Surakarta, Indonesia
arisbudianto@staff.uns.ac.id

***Abstract*— Task management is an essential aspect of organizational operations that involves coordinating various work activities and facilitating collaboration among team members. However, in many organizations, task management processes are still carried out through informal communication channels such as direct conversations, email, or instant messaging applications, which complicate task monitoring and documentation. This study aims to develop a web-based task management system that supports more structured task management within the Directorate of Information and Communication Technology at Universitas Sebelas Maret. The system was developed through four iterative Scrum sprints over 10 weeks. Functional testing employed the Black-Box Testing method, and usability was assessed using the System Usability Scale (SUS) with 20 purposively selected respondents across four user roles: administrators, sub-directorate heads, section heads, and staff members. The results indicate that all system functions operated successfully, and the system achieved an average SUS score of 75.25 — categorized as *Good* and exceeding the standard usability benchmark of 68, with scores ranging from 71.0 (section heads) to 81.0 (administrators). These findings suggest that Agile Scrum-based development produces task management systems that are adaptive to organizational needs and well-accepted across hierarchical user roles, with implications for IT governance practice in higher education institutions.**

Keywords*— *Agile Software Development, Scrum, System Usability Scale, Task Management System, Web-Based Information System

I. INTRODUCTION

Despite the widespread availability of digital collaboration tools, a significant proportion of organizational units in higher education institutions — particularly in developing countries — continue to rely on informal messaging applications such as WhatsApp and email for internal task coordination, a practice that systematically undermines work documentation, accountability, and operational transparency[1]. Task management is, therefore, a crucial component of modern organizational operations that involves team collaboration and the coordination of various work activities. In complex work environments, organizations require mechanisms that support systematic task planning, distribution, monitoring, and evaluation to ensure that work processes operate effectively and efficiently. Information technology–based task management systems have been widely developed to assist organizations in managing work activities in a structured manner while improving coordination among team members. Previous studies indicate that digital task management systems can enhance work efficiency, activity transparency, and the quality of operational coordination across various organizational contexts [1],[2]. Furthermore, the implementation of task management systems has been shown to support more structured monitoring of work processes, enabling organizations to manage operational activities more effectively [1].

However, in many organizations, task management processes are still conducted through informal communication mechanisms such as direct conversations, electronic mail, or instant messaging applications. This approach often creates several challenges in work coordination, particularly difficulties in monitoring task progress, limitations in documenting work activities, and the potential for miscommunication among team members. Previous studies indicate that poorly documented task coordination can lead to duplicated work, delays in task completion, and reduced efficiency in team collaboration within organizations [1], [2]. Therefore, organizations require a task management system capable of providing integrated mechanisms for recording, monitoring, and managing work activities so that coordination processes can be carried out more effectively and transparently.

With the increasing demand for information systems capable of supporting effective work coordination, software development methodologies have evolved toward more adaptive approaches. The Agile methodology is one of the most widely adopted software development approaches due to its ability to provide high flexibility in responding to changing user requirements. Agile emphasizes iterative system development, intensive collaboration among team members, and continuous feedback throughout the development process. This approach enables developers to continuously adjust the system being developed to the dynamic needs of organizations, allowing the system to deliver more responsive solutions to changes in the working environment [3], [4].

One of the most widely used frameworks within the Agile methodology is Scrum. Scrum provides a structured approach to software development by dividing the development process into short iterations known as sprints. Each sprint enables the development team to plan tasks, implement system features, and evaluate development outcomes incrementally. This approach allows the software development process to be carried out more adaptively and transparently, as each stage of development can be evaluated periodically.

In addition, Scrum provides several development artifacts, such as the product backlog and sprint backlog, which assist the team in managing work priorities and systematically monitoring project progress [5], [6]. Previous studies have also demonstrated that the Agile Scrum approach can enhance software development flexibility and support effective team collaboration throughout the system development process [7].

 EISSN: 2961-872X ●PISSN: 2961-8924●DOI: 10.56741/IISTR.esl.002246

The Directorate of Information and Communication Technology (DTIK) at Universitas Sebelas Maret is a work unit responsible for managing information technology infrastructure, maintaining information system services, and developing various applications that support the university's academic and administrative activities. These activities involve numerous technical and administrative tasks that must be managed by multiple team members with diverse responsibilities. Based on structured interviews with the Head of IT Planning at DTIK UNS, task distribution lacked systematic documentation, and progress monitoring relied entirely on informal communication channels such as WhatsApp groups and direct messaging, resulting in difficulties tracking task completion status, inconsistent work documentation, and the absence of a centralized mechanism for monitoring staff workload across units. These conditions highlight the urgent need for an integrated task management system capable of supporting structured task recording, transparent progress monitoring, and systematic work coordination within the organizational environment.

Based on these issues, this study aims to develop a web-based task management system to support task management processes within the Directorate of Information and Communication Technology at Universitas Sebelas Maret. The proposed system is designed to facilitate task recording, task assignment, work progress monitoring, and coordination among team members in a more structured and well-documented manner. The system development process adopts the Agile Software Development approach using the Scrum framework, enabling development to be carried out iteratively and adaptively in response to user requirements. Furthermore, this study evaluates the system's usability using the System Usability Scale (SUS) method to assess user acceptance of the implemented system.

Unlike previous studies that generally develop task management systems within business organization contexts, this study focuses on implementing the system within an information technology management unit in a higher education environment. In addition, this research conducts a usability evaluation involving several user roles within the organizational structure, namely administrators, sub-directorate heads, section heads, and staff members. This multi-role evaluation approach provides a more comprehensive understanding of system usage experiences across different levels of organizational hierarchy. Therefore, this study contributes empirically to the development of Agile Scrum–based task management systems and enriches the understanding of the implementation and usability evaluation of organizational information systems within the context of information technology governance in higher education institutions. This study makes three contributions: (1) empirical evidence for Agile Scrum adoption in higher education IT governance — an understudied context in the existing literature; (2) a multi-role SUS benchmark for organizational information systems, offering comparative usability insights across four distinct user roles within a hierarchical structure; and (3) a replicable development-evaluation framework that other university IT governance units can adopt to build and assess task management systems in similar institutional contexts.

II. Related Work

Research on task management systems has developed rapidly in response to the increasing need for organizations to manage work activities in a structured and well-documented manner. Various studies have developed information technology–based task management systems aimed at improving the efficiency of work coordination, enhancing organizational activity transparency, and increasing the effectiveness of project management. This section reviews previous studies relevant to the research topic, including the development of task management systems, the application of the Agile Scrum methodology in software development, and usability evaluation using the System Usability Scale (SUS) method. Through this review, the position of the research presented in this article can be more clearly understood within the context of existing knowledge development.

A. Task Management Systems

Task management systems represent a critical component in supporting work coordination within modern organizations. These systems enable organizations to plan, distribute, monitor, and evaluate work activities in a structured manner through a digital platform. Previous studies have shown that the implementation of web-based task management systems can enhance task management transparency and facilitate coordination among team members across various organizational contexts [1], [8].

Several studies have developed task management systems to support project management within organizations. In [9], A web-based task management system was developed to improve work coordination in a digital organizational environment. The developed system enables users to create, manage, and monitor tasks in an integrated manner within a single platform. The results of the study indicate that the implementation of a web-based task management system can enhance the transparency of work activities and reduce communication errors among team members. While Shah et al.[9]demonstrated improved group collaboration through web-based task management, their evaluation was limited to a single user group in a corporate context, leaving the multi-level hierarchical dynamics of academic IT governance units — such as the administrator, subdirectorate head, section head, and staff structure examined in the present study — unaddressed.

Another study conducted by [10]indicates that the use of digital task management systems can enhance the effectiveness of team collaboration within organizations, as each team member is able to monitor work progress in real time. The system enables organizations to manage work activities in a more systematic manner and improves accountability in task execution.

However, most studies on task management systems primarily focus on business organizations or industrial environments. Research on the implementation of task management systems within educational institutions, particularly within information technology management units in higher education, remains relatively limited. Therefore, research on the development of task management systems specifically designed to support work coordination within information technology units in higher education institutions still presents significant opportunities for further exploration.

B. Agile Scrum in Software Development

In modern software development, the Agile methodology has become one of the most widely adopted approaches due to its ability to provide high flexibility in responding to changing user requirements. Agile emphasizes an iterative

 EISSN: 2961-872X ●PISSN: 2961-8924●DOI: 10.56741/IISTR.esl.002246

and incremental software development process, in which systems are developed through multiple development cycles that enable continuous evaluation and improvement of the system [3].

One of the most popular frameworks within the Agile methodology is Scrum. Scrum divides the software development process into several iterations known as sprints, where each sprint produces incremental improvements in system functionality. This approach enables development teams to adapt system development to evolving user requirements throughout the development process [5].

Research conducted by [4] indicates that Agile functions not only as a software development methodology but also as a project management approach that can enhance team collaboration and improve responsiveness to changing system requirements. Furthermore, the study by [7] demonstrates that the implementation of Scrum in software development can increase the transparency of the development process and facilitate teams in managing work priorities through Scrum artifacts such as the product backlog and sprint backlog. Nevertheless, Moreno et al. [7] applied Agile Scrum within a construction project management context, and their findings were not validated through a formal usability evaluation instrument — limiting the extent to which their conclusions can be generalized to organizational information system development in higher education settings.

Although Agile Scrum has been widely applied in software development, most studies still focus on the development of commercial software products or enterprise information systems. The application of the Agile Scrum methodology in the development of organizational information systems within educational institutions has not been extensively examined, particularly in the context of task management within university information technology units.

C. Usability Evaluation using System Usability Scale

In addition to successful system development, usability is also a critical factor in determining the successful implementation of an information system. Usability refers to the degree to which a system can be used by users to complete assigned tasks effectively, efficiently, and satisfactorily. One of the most widely used usability evaluation methods in system development research is the System Usability Scale (SUS). SUS is a usability evaluation instrument consisting of ten statements rated using a Likert scale to measure users' perceptions of a system's ease of use [11]. This method is frequently applied in information systems research due to its high level of reliability and its ability to provide a quantitative assessment of user experience with the developed system.

Research conducted by [12] indicates that the SUS method can be used to evaluate system usability quickly and effectively, even with a relatively small number of respondents. Furthermore, the study by [1] demonstrates that usability evaluation using SUS can provide a clear overview of user acceptance levels toward the developed task management system. However, Kim et al. [1] conducted their SUS evaluation exclusively within a clinical healthcare setting and involving a single professional user group, which substantially limits the transferability of their usability findings to multi-role organizational IS contexts such as university IT governance units, where users span multiple levels of technical proficiency and administrative responsibility.

Although the SUS method has been widely used in information system evaluation studies, most research evaluates system usability within a single user group. Usability evaluations that involve multiple user groups with different roles within an organization are still relatively uncommon, resulting in limited comparative analysis of system usage experiences across different user groups.

D. Research Gap and Research Positioning

Based on the review of previous studies, it can be concluded that various studies have developed web-based task management systems and implemented the Agile Scrum methodology in software development. In addition, the System Usability Scale method has been widely used to evaluate the usability level of information systems.

However, several research gaps remain in previous studies. First, most research on task management systems has been conducted within the context of business organizations or industrial environments, while the implementation of such systems within information technology units in educational institutions has been relatively underexplored. Second, studies that develop task management systems using the Agile Scrum approach generally focus on system development without conducting comprehensive usability evaluations across different user groups within the organization. Third, most usability evaluation studies using the SUS method involve only a single user group, thereby providing limited insights into system usage experiences across various user roles within an organization.

Therefore, this study seeks to address these research gaps by developing a web-based task management system using the Agile Scrum methodology, specifically designed to support task management within information technology units in higher education institutions. In addition, this study conducts a usability evaluation using the System Usability Scale method involving multiple user groups within the organization to obtain a more comprehensive understanding of the usability level of the developed system.

TABLE I. COMPARISON OF RELATED TASK MANAGEMENT SYSTEM STUDIES

Author	System Type	Development Method	Evaluation Method	Limitation	Ref.
Binti M. Shah et al.	Web Task Management System	Agile	Usability testing	Single organizational context	[9]
Kim et al.	Digital task management platform	Agile development	SUS evaluation	Focus on the corporate environment	[1]
Moreno et al.	Project management system	Agile Scrum	User testing	Not evaluated in academic institutions	[7]

III. METHODOLOGY

This study adopts the Research and Development (R&D) approach to develop and evaluate a web-based task management system for the Directorate of Information and

Communication Technology at Universitas Sebelas Maret. Unlike the Design Science Research (DSR) approach, which targets generalizable design principles, R&D is oriented toward direct user testing within a real organizational context [7], [13] making it appropriate for this study's goal of producing a deployable information system artifact that can be immediately utilized within the organizational work environment. The R&D approach provides a systematic framework encompassing user requirements analysis, system design, software implementation, and iterative evaluation — enabling the system to be continuously refined in accordance with user needs throughout the development process [3]. To clarify the research process, this study presents the research methodology framework in Fig. 1, which illustrates the main stages of research conducted iteratively throughout the project.

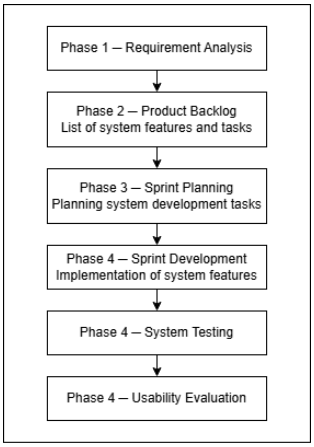


Fig. 1 Research Methodology Framework

A. System Development Method

The system development process in this study employs the Agile Software Development approach using the Scrum framework. The Agile methodology was selected because it provides high flexibility in addressing changes in user requirements and enables system development to be conducted iteratively and incrementally. The Agile approach emphasizes team collaboration, intensive communication with stakeholders, and system development through short development cycles, allowing the system to be continuously improved throughout the development process [3], [13].

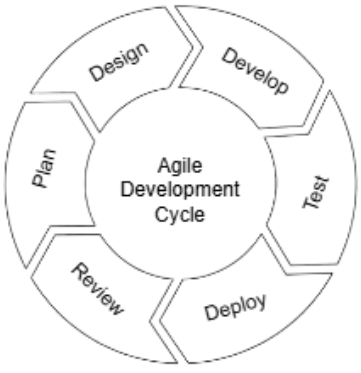


Fig. 2 Agile Development Cycle

The Agile approach has been widely adopted in modern software development because it can improve development efficiency, accelerate system delivery processes, and enhance the quality of the resulting software products. In addition, Agile enables developers to respond quickly to changes in user requirements through continuous iterative processes [7], [14].

B. Scrum Development Process

The system development process in this study follows the main stages of the Scrum framework, namely product backlog, sprint planning, sprint development, sprint review, and sprint retrospective. The complete development timeline of this study comprised one pre-development phase and four iterative sprints conducted between May and July 2025, with each phase targeting a specific set of system features as defined in the sprint backlog. Sprint performance was monitored using focus factor — calculated as the ratio of completed backlog items to available work capacity per sprint — yielding an overall average of 0.89, which indicates that 89% of all planned backlog items were completed within their scheduled time allocations throughout the entire development process. The product backlog stage involves the identification and collection of system requirements based on the results of user needs analysis. All system requirements are organized into a prioritized list of features to be developed in the task management system.

TABLE II. SCRUM SPRINT SUMMARY

Phase	Duration (Planned)	Duration (Actual)	Backlog Items	Status
Pre-Development	10 days	10 days	5 items	On schedule
Sprint 1	14 days	14 days	5 items	On schedule
Sprint 2	14 days	16 days	5 items	+2 days delay
Sprint 3	10 days	10 days	5 items	On schedule
Sprint 4	12 days	14 days	5 items	+2 days delay
Total	60 days	64 days	25 items	Focus Factor: 0.89

Two sprints experienced minor schedule overruns of two days each — Sprint 2 due to feature integration complexity between the task assignment and progress monitoring modules, and Sprint 4 due to final-stage system refinements required before field deployment — both of which were identified and addressed during their respective sprint retrospectives.

The next stage is sprint planning, which involves planning the tasks to be carried out within a single sprint cycle. At this stage, the development team determines the features to be developed based on the previously established backlog priorities. Subsequently, the sprint development process is conducted to implement system features in accordance with the agreed sprint plan.

After the development process is completed, a sprint review is conducted to evaluate the results of the system development and ensure that the implemented features meet user requirements. At this stage, stakeholders can also provide feedback on the developed system. The final stage is the sprint retrospective, which involves reflecting on the development process to identify improvements that can be implemented in the next sprint, thereby enabling the system development process to proceed more effectively [5], [7].

Table III summarizes the Scrum artifacts employed throughout the development process. The Product Backlog consisted of 25 prioritized system requirements obtained from stakeholder interviews and requirement analysis. During

 EISSN: 2961-872X ●PISSN: 2961-8924●DOI: 10.56741/IISTR.esl.002246

Sprint Planning, selected backlog items were transferred into the Sprint Backlogs, containing approximately five development items per sprint.

TABLE III. SCRUM ARTIFACT IMPLEMENTATION

Scrum Artifact	Description	Evidence in This Study
Product Backlog	Prioritized list of system requirements collected from stakeholders	25 backlog items identified during requirement analysis
Sprint Backlog	Selected backlog items assigned to each sprint	Each sprint contained 5 backlog items executed during development
Increment	Functional system features are produced at the end of each sprint	Dashboard, Task Management, Discussion, Attachment, Reporting features

Each sprint produced a potentially releasable Increment, including dashboard modules, task management features, task discussion functionality, attachment handling, and reporting components. These artifacts were continuously refined through sprint reviews and retrospectives to ensure alignment with stakeholder requirements.

TABLE IV. SAMPLE PRODUCT BACKLOG ITEMS

ID	Product Backlog Item	Priority
PB-01	User Authentication and Login	High
PB-02	Dashboard and Task Summary	High
PB-03	Task Creation and Assignment	High
PB-04	Task Discussion and Collaboration	Medium
PB-05	File Attachment Management	Medium

Table IV presents representative examples of Product Backlog items identified during the requirement analysis phase. The complete Product Backlog consisted of 25 prioritized requirements collected from stakeholder interviews and organizational needs assessment. These backlog items were prioritized according to their importance for supporting task management activities and subsequently distributed into Sprint Backlogs during Sprint Planning. The use of a prioritized Product Backlog is consistent with Scrum practices, which emphasize systematic requirement management and incremental value delivery throughout the development lifecycle [5] [15].

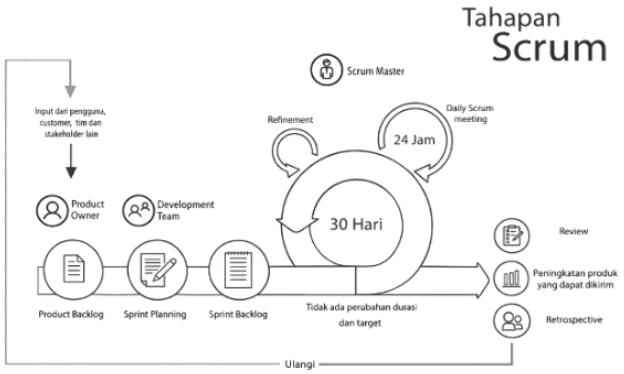


Fig. 3 Scrum Development Process

C. System Testing

System testing represents a critical phase in software development to verify that the resulting system operates as intended based on the defined requirements. In this research, system testing was carried out through the Blackbox Testing method, a testing approach that emphasizes evaluating system functionality from the user's perspective without examining the underlying program code structure.

The testing procedure was performed by supplying a variety of input scenarios to the system and examining the resulting outputs to verify that every system function operates as expected by users. The Blackbox Testing approach is commonly adopted in information system testing due to its capability to assess the degree of alignment between system functions and the previously established requirement specifications [3], [7].

Through this method, each system feature is tested based on predefined testing scenarios to ensure that all system functions operate properly without producing errors or system failures. This testing approach also enables the early identification of potential system errors before the system is deployed for user use [15], [16], .

TABLE V. BLACKBOX TESTING SCENARIO

Feature	Test Scenario	Expected Result
Login	User enters a valid email and password	The system successfully logs in and displays the dashboard
Logout	User presses the logout button	The system logs out and returns to the login page
User Data CRUD (Admin)	Admin adds, edits, and deletes user data	User data is successfully added, updated, and deleted
Task CRUD	User adds, edits, and deletes tasks	Tasks are successfully added, updated, and deleted
Task Discussion	User adds a comment to a task	The comment is successfully added and displayed
Task Attachment Upload	User uploads a file to the task	The file is successfully uploaded and attached to the task
Notification System	Admin assigns a task with a deadline to a staff member; the assigned user then logs in	The assigned user receives an in-system notification containing the task title, assigned deadline, and the assigning user's name
Time Tracking	User changes a task status from "In Progress" to "Completed"	The system records the task completion timestamp and displays the total elapsed time between the "In Progress" and "Completed" status transitions
Dashboard	User accesses the dashboard page	Dashboard information is displayed correctly
Reporting	User generates task report	Reports are generated and displayed successfully

D. Usability Evaluation

Beyond functional testing, this study also assessed how usable the system is by employing the System Usability Scale (SUS) as the evaluation instrument. SUS is a usability evaluation instrument widely used in information systems research to measure users' perceptions of system usability through a questionnaire consisting of ten statements rated on a five-point Likert scale [11]. This method is considered effective in providing a quick and reliable assessment of system usability across various software development contexts [17].

The usability evaluation involved 20 respondents distributed across four user roles, with five participants representing each role. This sample size was selected based on established usability evaluation practices, which suggest that approximately five users from a relatively homogeneous user group are often sufficient to identify a substantial proportion

of usability issues within an interface [19]. Furthermore, previous studies have shown that System Usability Scale (SUS) scores remain stable and reliable even when relatively small samples are used in usability assessments [20] [21]. Since each role group in this study consisted of users with similar responsibilities and system interaction patterns, five participants per role were considered adequate to capture role-specific usability perceptions while maintaining consistency with established usability evaluation practices.

The respondents were selected using purposive sampling, a participant selection method based on specific criteria relevant to the research objectives. This technique was employed to ensure that all respondents had direct experience using the developed system, enabling them to provide more accurate usability evaluations.

Before the evaluation process was conducted, all respondents were asked to use the task management system through several system usage scenarios, including task creation, task assignment, work progress monitoring, and task status updates. Once the respondents had gained sufficient familiarity with the core features of the system, they were required to fill out the SUS questionnaire, which comprises ten items measured on a five-point Likert scale spanning from strongly disagree to strongly agree.

The usability score was calculated using the standard SUS scoring method by adjusting the score of each item based on whether the statement was positive or negative. The final SUS score was obtained by summing all adjusted scores and multiplying the total by a conversion factor of 2.5, resulting in a usability score ranging from 0 to 100 [19].

$$SUS\ Score = (\Sigma S_i) \times 2.5 \quad (1)$$

where:

S_i = Adjusted score for each questionnaire item

2.5 = Multiplication factor used to convert the score to a 0–100 scale

The obtained SUS score is then used to determine the system's usability level based on the SUS rating categories commonly applied in system usability evaluations.

TABLE IIII. SYSTEM USABILITY SCALE QUESTIONNAIRE

No	Questions	Scale
1	I think that I would like to use this system frequently.	5 4 3 2 1
2	I found the system unnecessarily complex.	5 4 3 2 1
3	I thought the system was easy to use.	5 4 3 2 1
4	I think that I would need the support of a technical person to be able to use this system.	5 4 3 2 1
5	I found the various functions in this system were well integrated.	5 4 3 2 1
6	I thought there was too much inconsistency in this system.	5 4 3 2 1
7	I would imagine that most people would learn to use this system very quickly.	5 4 3 2 1
8	I found the system very cumbersome to use.	5 4 3 2 1
9	I felt very confident using the system.	5 4 3 2 1
10	I needed to learn a lot of things before I could get going with this system.	5 4 3 2 1

In addition to calculating the SUS score, the reliability of the questionnaire instrument was also analyzed to ensure the internal consistency of each question item. Reliability testing was conducted using Cronbach's Alpha coefficient, which is commonly applied in the evaluation of questionnaire-based research instruments. The test results indicate that the Cronbach's Alpha value is > 0.7, demonstrating that the System Usability Scale instrument has good reliability and can be used to consistently measure users' perceptions of the system's ease of use [11], [23].

E. Ethical Considerations

This research was carried out by adhering to ethical standards that apply to studies involving human subjects. All respondents involved in the usability evaluation were provided with an explanation of the research objectives and the system testing procedures before the data collection process. Participation in this study was voluntary, and each respondent had the right to withdraw from participation at any time without any consequences.

Before completing the System Usability Scale (SUS) questionnaire, all participants provided informed consent to participate in this study. The data obtained from the respondents were used solely for research purposes and were analyzed anonymously to protect the confidentiality of participant identities. All research data were stored securely and used exclusively for academic analysis purposes.

To minimize social desirability bias during data collection, questionnaire responses were collected anonymously and were not linked to participants' identities. Respondents were informed that the evaluation was intended to assess the usability of the system rather than individual performance and that there were no correct or incorrect answers. In addition, participants completed the questionnaire independently without direct supervision from the system developers, thereby reducing potential pressure to provide favorable responses. These procedures are consistent with recommendations for reducing response bias and enhancing the credibility of self-reported data in usability research.

IV. RESULTS

A. System Implementation

The results of this study include the development of a web-based task management system designed to support more structured task management processes within the Directorate of Information and Communication Technology at Universitas Sebelas Maret. The developed system aims to facilitate task recording, the distribution of work among team members, work progress monitoring, and coordination among team members within a single integrated platform. The implementation of a web-based task management system enables the organization to manage work activities more systematically and enhance transparency in the work management process [9], [20].

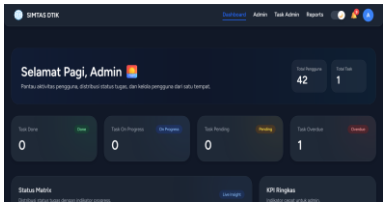
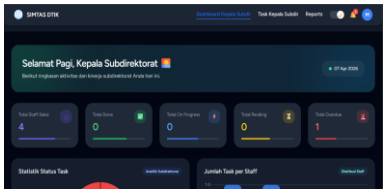
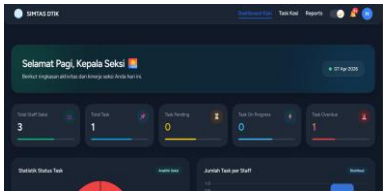
The web-based task management system provides a mechanism that enables users to create, manage, and monitor work activities centrally within an information system. This approach supports team collaboration, as each team member can track work progress in real time and participate more effectively in task coordination processes. The use of a task management system can also help organizations improve the efficiency of work coordination and reduce the potential for communication errors in task execution [8], [10].

Table VII presents the main dashboard interface of the system, displaying a summary of task activities, work status, and information about users involved in the system. The dashboard serves as an information center, providing an overview of work progress within the organization. The developed task management system includes several key

 EISSN: 2961-872X ●PISSN: 2961-8924●DOI: 10.56741/IISTR.esl.002246

features designed to support task management processes within the organization.

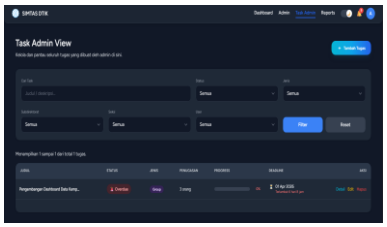
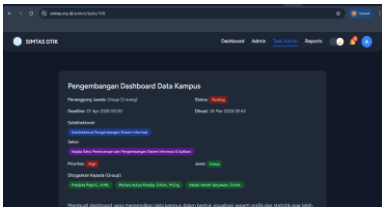
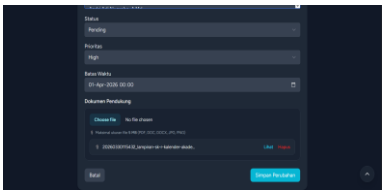
TABLE IVII. TASK MANAGEMENT SYSTEM DASHBOARD

Image	Description
	The Admin Dashboard displays a summary of all task activities and their status within the system.
	The Sub-Directorate Head Dashboard presents the distribution and progress of tasks across each section.
	The Section Head Dashboard provides an overview of tasks managed within a specific section.
	The Staff Dashboard displays the list of tasks assigned to the user along with their completion status.

These features include task creation, task assignment to team members, task tracking, and work status management. These functionalities enable users to manage work activities in a more structured and systematic manner and ensure that each task can be monitored in real time through an integrated system [9], [21].

An integrated task management system within a digital platform can also assist organizations in improving the effectiveness of project management and facilitating decision-making processes. With a centralized task monitoring mechanism, each team member can track work progress more transparently, thereby enhancing team productivity within the organization [7], [22].

TABLE VI. TASK MANAGEMENT SYSTEM FEATURES

Image	Description
	The Task Management page displays a list of all tasks along with search and filter features to facilitate task management.
	The Create Task page enables users to add new tasks by inputting the task title, description, and task type, as well as allocating the task to a particular user or group.
	The Task Detail page presents detailed information about a task, including the person responsible, priority, status, and deadline.
	The Edit Task page allows users to update task information such as status, priority, deadlines, and supporting documents.

This table presents the task management features that allow users to create new tasks, update work status, and monitor the progress of ongoing tasks within the system.

TABLE IX. TASK MANAGEMENT SYSTEM FEATURES

Sprint	Increment Produced
Pre-Development	Requirement specification, use case model, database design
Sprint 1	Authentication and dashboard modules
Sprint 2	Task creation and assignment modules
Sprint 3	Task discussion and attachment modules
Sprint 4	Reporting features, bug fixing, and system refinement

B. *Blackbox Testing Results*

System testing was performed to confirm that all system functions run as expected based on user requirements. In this research, the Blackbox Testing method was applied as the testing approach, which concentrates on assessing system functionality by observing the outputs produced from given inputs, without taking into account the internal program code structure. This method served to measure how well the system functions align with the requirement specifications that were established during the system analysis and design phases[24].

Testing was conducted on several main system features, including user authentication, task creation, task data management, and task status updates. Each feature was tested using predefined test scenarios by providing specific inputs to the system and evaluating the resulting outputs. The testing results indicate that all system features function properly and meet user requirements, demonstrating that the developed task management system satisfies the functional specifications defined during the system analysis stage [9], [21].

Initial Black Box Testing across 53 test cases identified five functional defects (9.4**%** initial failure rate) in the file upload and comment reply to features — specifically, a 404 Not Found error when attempting to retrieve uploaded files due to misconfigured storage path settings, and a 500 Internal Server Error when accessing the comment reply function. These defects were systematically resolved in Sprint 3 through targeted debugging of the file storage configuration and

comment processing logic. Post-correction re-testing confirmed a 100% pass rate across all 53 test cases, demonstrating that the developed task management system fully satisfies the functional specifications defined during the system analysis stage and is considered suitable for operational deployment within DTIK Sebelas Maret University. In addition to the core functional scenarios, the Notification System and Time Tracking features listed as operational components were explicitly included in the blackbox testing scenarios to verify their end-to-end behavior within the task lifecycle flow. Both features were confirmed to function as expected, with no defects identified during testing [3], [15].

C. Usability Testing Results

In addition to functional system testing, this study also evaluated the system's usability using the System Usability Scale (SUS) method. The SUS method is one of the most widely used usability evaluation instruments in system development research to measure the level of system usability based on user perceptions. This instrument consists of ten statements rated using a Likert scale, producing a usability score ranging from 0 to 100 [11].

The usability evaluation involved 20 respondents representing four groups of system users: administrators, sub-directorate heads, section heads, and staff members. Each user group consisted of five respondents who had used the task management system in operational activities within the Directorate of Information and Communication Technology at Universitas Sebelas Maret.

The usability evaluation results for each user group are presented in Table X through Table XIII, which display the questionnaire scores after the value conversion process according to the SUS calculation method. Based on these calculations, the usability scores obtained were 81 for administrators, 72.5 for sub-directorate heads, 71 for section heads, and 76.5 for staff members. A summary of the system usability evaluation results based on user roles is presented in Table XIV. The table indicates that all user groups provided SUS scores above the standard average usability score of 68, suggesting that the system has a good level of usability and is well accepted by users [11], [12].

The overall average SUS score obtained from all respondents was 75.25. According to the interpretation guidelines proposed by Bangor, Kortum, and Miller (2009), SUS scores above the standard benchmark value of 68 indicate above-average usability, while scores within the range of 70–80 are generally categorized as "Good." Therefore, the score achieved by the developed task management system suggests that users perceived the system as easy to use and satisfactory for supporting organizational task management activities. This finding indicates that the system provides a positive user experience across different organizational roles and demonstrates an acceptable level of usability for operational deployment within the university IT governance environment.

The usability result obtained in this study is also consistent with findings reported in previous information system usability research. [1] reported that a task management system developed for clinical environments achieved satisfactory usability and supported users in managing work-related activities effectively. Similarly, [12] found that digital systems achieving SUS scores above 70 generally provide satisfactory user experiences and acceptable usability performance. Compared with these findings, the average SUS score of 75.25 achieved in this study demonstrates that the developed system offers a competitive level of usability despite being implemented within a multi-role organizational environment involving administrators, sub-directorate heads, section heads, and staff members. This result suggests that the Agile Scrum development approach successfully supported the creation of a system capable of accommodating diverse user needs while maintaining a consistent level of usability across organizational roles.

The distribution of usability scores from all respondents is shown in Fig. 4. The graph indicates that most respondents rated the system's usability as good. These results indicate that the developed task management system has high user acceptance and effectively supports task management processes within the organizational environment.

TABLE X. SUS SCORES OF ADMINISTRATOR RESPONDENTS

Respondent	Question									
	Q1	Q2	Q3	Q4	Q5	Q6	Q7	Q8	Q9	Q10
R1	4	3	4	3	5	2	5	2	5	3
R2	5	3	3	2	4	2	4	2	5	2
R3	3	4	4	3	4	3	5	2	4	3
R4	3	3	4	3	4	3	4	3	4	2
R5	3	3	4	2	4	2	4	3	4	2

TABLE XI. SUS SCORES OF SUB-DIRECTORATE HEAD RESPONDENTS

Respondent	Question									
	Q1	Q2	Q3	Q4	Q5	Q6	Q7	Q8	Q9	Q10
R1	4	3	4	2	4	1	4	1	5	3
R2	5	2	4	1	4	1	4	2	4	3
R3	4	2	3	3	4	3	4	2	3	3
R4	4	3	3	2	3	2	3	2	5	3
R5	4	2	4	2	4	3	4	2	4	2

TABLE VIII. SUS SCORES OF SECTION HEAD RESPONDENTS

Respondent	Question									
	Q1	Q2	Q3	Q4	Q5	Q6	Q7	Q8	Q9	Q10
R1	3	2	4	2	4	2	4	2	4	3
R2	4	2	4	2	4	2	4	2	4	2
R3	4	2	4	3	4	3	3	2	4	3
R4	4	2	4	2	5	2	2	2	4	2
R5	4	2	4	2	3	2	3	2	5	2

TABLE VIII. SUS SCORES OF STAFF RESPONDENTS

Respondent	Question									
	Q1	Q2	Q3	Q4	Q5	Q6	Q7	Q8	Q9	Q10
R1	5	2	4	2	4	3	4	2	4	3
R2	4	3	2	1	4	2	5	2	4	2
R3	5	2	5	2	3	1	4	1	4	2
R4	4	3	5	1	4	2	4	2	4	2
R5	5	3	4	1	4	2	3	2	3	1

 EISSN: 2961-872X ●PISSN: 2961-8924●DOI: 10.56741/IISTR.esl.002246

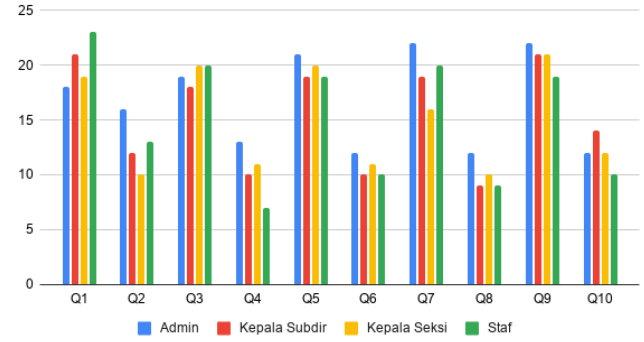


Fig. 1 Distribution of SUS Scores

Fig. 1 illustrates the distribution of usability scores given by respondents toward the developed task management system.

TABLE XIV. SUS USABILITY SUMMARY

Role	n	Mean SUS	Category
Administrator	5	81.0	Excellent
Sub-Directorate Head	5	72.5	Good
Section Head	5	71.0	Good
Staff	5	76.5	Good
Total	**20**	**75.25**	**Good**

Table XIV presents a summary of the system usability evaluation results, showing the average SUS scores across all user groups.

V. DISCUSSION

A. Impact of Agile Implementation in System Development

As stated in the research objectives, this study aims to develop a web-based task management system capable of improving task coordination within the organization and to evaluate the usability level of the resulting system. The research findings indicate that the implementation of the Agile Software Development methodology using the Scrum framework contributes meaningfully to the successful development of a task management system that is adaptive to user needs. The Agile approach enables the software development process to be conducted iteratively through development cycles known as sprints, allowing the development team to continuously evaluate and refine the system throughout the development process. This approach enables changes in user requirements to be accommodated more rapidly compared to traditional linear software development methods. Previous studies have shown that the Agile methodology provides high flexibility in system development and enhances the team's ability to respond to evolving user requirements during the development process [3], [14].

The results of this study are also consistent with empirical studies indicating that the implementation of Agile development can enhance team collaboration effectiveness and accelerate the software development process through continuous iterative mechanisms [7]. Through the processes of sprint planning, sprint development, and sprint review, the development team can conduct periodic evaluations of the system features being developed, allowing design errors or unmet user requirements to be identified and addressed promptly. This iterative approach has been shown to improve the quality of the developed system and strengthen communication between developers and stakeholders throughout the system development process [5].

In addition, this study demonstrates that the use of Scrum artifacts such as the product backlog and sprint backlog assists the development team in prioritizing system feature development based on user requirements. Previous studies indicate that Scrum artifacts play an important role in improving the transparency of the software development process and facilitating work coordination within development teams [25]. Sprint performance analysis via burndown charts yielded an average focus factor of 0.89 across all development phases, indicating that 89% of planned backlog items were completed within their scheduled time allocations — a level of execution consistency that reflects disciplined sprint management throughout the project. Two sprints (Sprint 2 and Sprint 4) experienced minor schedule overruns of two days each: Sprint 2 due to feature integration complexity between the task assignment and progress monitoring modules, and Sprint 4 due to final-stage system refinements required before field deployment. These delays were identified and addressed during sprint retrospectives, demonstrating that the Scrum feedback loop functioned as intended in detecting and correcting deviations from the development plan. The occurrence of minor delays in integration-heavy sprints is consistent with findings on Agile scalability challenges in organizational IS development, which suggest that inter-module dependencies and stakeholder feedback cycles are among the most common sources of sprint overrun in iterative projects [5],[7]. Thus, the implementation of Scrum in this study functions not only as a software development framework but also as a rigorous project management mechanism that provides empirical evidence of disciplined backlog execution — as demonstrated by the 0.89 focus factor — while remaining adaptive to the dynamic requirements of organizational task management development.

The implementation of Scrum artifacts was explicitly documented throughout the project. The Product Backlog contained 25 prioritized requirements, Sprint Backlogs were generated for each iteration, and every sprint produced a functional Increment that could be reviewed by stakeholders. This artifact-driven process improved development transparency and facilitated progress tracking across the four sprint cycles.

B. Analysis of Task Management System Implementation

The implementation of the task management system developed in this study indicates that the use of a web-based task management system can enhance the effectiveness of work coordination within the organization. The developed system provides key features such as task creation, task assignment to team members, task progress monitoring, and work status management integrated within a single information system platform. These features enable each team member to monitor work progress in real time, thereby allowing work coordination processes to be carried out more effectively.

The findings of this study are consistent with previous research indicating that web-based task management systems can enhance work transparency because all task activities can

be systematically recorded within an integrated digital platform [9]. Digital task management systems also enable organizations to monitor work activities more accurately, allowing decision-making processes to be conducted more quickly and based on data. Other studies have shown that the implementation of digital task management systems can improve team collaboration efficiency, as team members can more clearly understand the status of tasks and their respective responsibilities [8].

In addition to improving work transparency, task management systems also contribute to increased accountability within organizations. All work activities performed by users can be documented within the system, facilitating monitoring and team performance evaluation. Previous studies have shown that information technology–based task management systems can enhance the effectiveness of project management by enabling the integration of various work activities within a centralized system [2]. Therefore, the implementation of the task management system developed in this study has the potential to provide practical benefits for organizations with complex work coordination requirements.

To illustrate the relationship between system features and the operational benefits resulting from the implementation of the task management system, Table XV presents a mapping between the main system features and the operational benefits obtained in the organizational task management process.

TABLE XV. SYSTEM FEATURES AND OPERATIONAL BENEFITS

No	System Feature	Description	Operational Benefit
1	Task Creation and Assignment	Allow administrators to create tasks and assign them to specific staff members, along with deadlines and task details.	Improves task distribution and ensures responsibilities are clearly defined.
2	Task Monitoring	Enables supervisors to monitor task progress and completion status in real time.	Enhances transparency and facilitates better supervision of ongoing tasks.
3	Notification System	Provides automatic notifications and reminders for assigned tasks and approaching deadlines.	Helps users stay informed and reduces the risk of missed or delayed tasks.
4	Task Reporting	Generates reports on task completion, progress, and performance metrics.	Supports evaluation of employee performance and decision-making processes.
5	Time Tracking	Records the time spent on each task during execution.	Help measure productivity and improve task time management.
6	Task Discussion	Allow team members to communicate and discuss tasks within the system.	Improves collaboration and facilitates problem-solving during task execution.
7	Task Attachment	Enables users to attach supporting files or documents related to tasks.	Ensures all necessary resources and documentation are available within the task.
8	Task Priority and Status	Provides priority levels and status indicators such as pending, in progress, and completed.	Help users identify urgent tasks and manage workflow more effectively.

C. System Usability Evaluation

System usability evaluation was conducted using the System Usability Scale (SUS) method to measure the system's ease of use based on user perceptions. The evaluation results indicate that the developed task management system obtained an average SUS score of 75.25, demonstrating that the system has a good level of usability and is acceptable to users. This score is above the standard average usability benchmark of 68, indicating that the system can be categorized as having adequate usability for use within an organizational work environment. Previous studies suggest that SUS scores above the standard average indicate good user acceptance and support effective user interaction with the system [11].

The results of this evaluation are consistent with previous studies, indicating that information systems developed using an iterative development approach tend to achieve higher usability levels, as developers can integrate user feedback at each stage of the system development process[29]. The integration of user feedback in system development allows developers to adjust the system interface design according to user needs, thereby improving the overall user experience. Therefore, the Agile approach used in this study plays an important role in producing a system with a good level of usability.

The distribution of usability scores obtained from respondents is presented in Fig. 4, which shows that most respondents rated the system's usability within the good category. A Kruskal-Wallis test was conducted to determine whether SUS scores differed significantly across the four user roles. The results revealed no statistically significant difference ($H = 3.553$, $p = 0.314$), suggesting that usability perception was consistent across all levels of the organizational hierarchy — from administrators to staff members — regardless of positional authority or varying levels of technical proficiency. Item-level analysis of SUS responses revealed that the system demonstrated particular strengths in task assignment transparency, real-time progress monitoring, and role-based access control — features that received consistently high ratings across all user groups, particularly on Q7 (learnability: most people would learn to use this system very quickly) and Q9 (confidence: felt very confident using the system). However, the system exhibited limitations in technical accessibility and interface consistency, as reflected in the relatively lower mean scores on Q4 ("I think that I would need the support of a technical person to be able to use this system") and Q6 ("I thought there was too much inconsistency in this system"), suggesting that less technically experienced users — particularly section heads and staff — may face adoption barriers during initial system use. These findings indicate that future iterations should prioritize user onboarding mechanisms and interface standardization to further improve system accessibility across all organizational levels.

The absence of statistically significant differences in SUS scores among user roles indicates that the developed system provides a relatively consistent usability experience across the organizational hierarchy. Although administrators obtained the highest average SUS score (81.0) and section heads obtained the lowest (71.0), all role groups reported scores above the standard usability benchmark. This finding

 EISSN: 2961-872X ●PISSN: 2961-8924●DOI: 10.56741/IISTR.esl.002246

suggests that differences in organizational responsibilities and authority levels did not substantially influence users' perceptions of system usability. Consequently, the developed task management system can be considered suitable for supporting task coordination activities across multiple organizational roles within the university IT governance environment.

D. Theoretical and Practical Implications

This study provides a theoretical contribution to the development of literature on the application of the Agile Scrum methodology in the development of organizational information systems. From a theoretical standpoint, these findings are consistent with the Technology Acceptance Model's proposition that perceived ease of use is a primary predictor of system adoption [30] — as evidenced by the positive SUS scores across all organizational roles despite varying levels of technical proficiency, and further supported by the Kruskal-Wallis analysis ($H = 3.553$, $p = 0.314$), which confirmed that usability perception remained consistent regardless of users' positional authority or technical background. The findings theoretically indicate that an Agile-based system development approach, characterized by iterative processes and user feedback orientation, can produce systems with better usability levels. These findings reinforce perspectives in the software engineering literature that the integration of user feedback in each system development iteration is a key factor in improving the quality of developed information systems [3][7]. Therefore, this study expands the understanding of Agile Scrum implementation in the context of organizational information system development, particularly within the information technology management environment of higher education institutions.

In addition to its theoretical contributions, this study also provides practical implications for organizations that require complex work coordination. The implementation of the web-based task management system developed in this study demonstrates that the use of a digital task management system can enhance transparency in work management, improve coordination among team members, and facilitate the monitoring of task progress within the organization. Therefore, the Agile Scrum–based system development approach can serve as an effective strategy for organizations to develop information systems that are adaptive to user needs and capable of improving the efficiency of task management within the organizational work environment. Thus, the results of this study indicate that the research objectives of developing an Agile Scrum–based task management system and evaluating the system's usability in supporting organizational task coordination have been successfully achieved.

VI. CONCLUSION

A. Conclusion

This study successfully developed a web-based task management system designed to support structured task management within the Directorate of Information and Communication Technology at Universitas Sebelas Maret. The developed system provides key features, including task recording, task assignment, task progress monitoring, and work status management within a single integrated platform that supports more effective teamwork coordination. The results of functional testing indicate that all system features operate in accordance with user requirements. In addition, the usability evaluation using the System Usability Scale (SUS) method involving 20 respondents produced an average score of 75.25, which falls within the good usability category. These findings indicate that the developed system has a good level of usability and is well accepted by users in supporting organizational task management activities. By focusing on a university IT governance unit and involving four distinct user roles, this study directly addresses the gaps identified in the literature regarding task management system implementation in academic contexts and the absence of multi-role usability benchmarks. Collectively, these findings establish that Agile Scrum-based development with multi-role SUS evaluation is a viable and user-accepted approach for building organizational information systems in higher education — a contribution that extends beyond DTIK to other university IT governance contexts in developing countries.

B. Research Limitations

This study has several limitations that should be considered. The system implementation was conducted within a single organizational unit, so the evaluation results may not fully represent the application of the system in organizations with larger scales or different organizational structures. This condition limits the generalizability of the research findings across various organizational contexts. In addition, the system evaluation in this study only employed the System Usability Scale (SUS) method, which focuses on measuring user perceptions of the system's ease of use. This approach does not include a more in-depth analysis of user experience and therefore does not fully capture the quality of user interaction with the developed system. A third limitation concerns the cross-sectional evaluation design; usability was measured at a single point in time following system deployment, without longitudinal follow-up to assess whether the reported usability levels are sustained as users gain more experience with the system over time. Additionally, respondents were internal organizational members who were directly involved in the development process, which may introduce social desirability bias in SUS ratings — potentially inflating perceived usability scores relative to what independent external users might report.

C. Recommendations for Future Research

Future research should: (1) replicate this study in at least three additional university IT units with varying organizational scales to test external validity; (2) employ longitudinal evaluation spanning a minimum of six months post-adoption to assess sustained usability; and (3) integrate SUS with the Technology Acceptance Model (TAM) to distinguish perceived usefulness from ease of use as independent predictors of system adoption.

ACKNOWLEDGEMENTS

The authors are grateful to the Ministry of Education and Culture, as well as the Institute for Research and Community Service, Universitas Sebelas Maret, for the financial support provided to this study, under contract number 462/UN27.22/PT.01.03/2026

DATA AVAILABILITY STATEMENT

The data and source code supporting the findings of this study are the property of the Directorate of Information and Communication Technology (DTIK) Universitas Sebelas Maret and are subject to institutional confidentiality restrictions. Access requests can be directed to the corresponding author.

REFERENCES

[1] S. H. Kim, J. Jin, M. Sevinchan, and A. Davies, "How do automated reasoning features impact the usability of a clinical task management system? Development and usability testing of a prototype," *Int. J. Med. Inform.*, vol. 174, Jun. 2023, doi: 10.1016/j.ijmedinf.2023.105067.

[2] S. Mittal, R. K. Dudeja, R. S. Bali, and G. S. Aujla, "A distributed task orchestration scheme in collaborative vehicular cloud edge networks," *Computing*, vol. 106, no. 4, pp. 1151–1175, Oct. 2022, doi: 10.1007/s00607-022-01119-9.

[3] S. Al-Saqqa, S. Sawalha, and H. Abdelnabi, "Agile Software Development: Methodologies and trends," *International Journal of Interactive Mobile Technologies*, vol. 14, no. 11, pp. 246–270, Jul. 2020, doi: 10.3991/ijim.v14i11.13269.

[4] S. Magistretti and D. Trabucchi, "Agile-as-a-tool and agile-as-a-culture: a comprehensive review of agile approaches adopting contingency and configuration theories," *Review of Managerial Science*, vol. 19, no. 1, pp. 223–253, Jan. 2024, doi: 10.1007/s11846-024-00745-1.

[5] M. Žáček, A. Hamplová, J. Tyrychtr, and I. Vrana, "Improvements for the Planning Process in the Scrum Method," *Applied Sciences (Switzerland)*, vol. 15, no. 1, Dec. 2024, doi: 10.3390/app15010202.

[6] L. Bisikirskienė, L. Čeponienė, G. Vilutis, and A. Nečionytė, "Software Project Risk Identification Based on Scrum Artifacts," *Applied Sciences (Switzerland)*, vol. 15, no. 2, Jan. 2025, doi: 10.3390/app15020824.

[7] F. Moreno, E. Forcael, R. Romo, F. Orozco, G. Moroni, and F. Baesler, "Agile Project Management in the Pre-Construction Stage: Facing the Challenges of Projectification in the Construction Industry," *Buildings*, vol. 14, no. 11, Nov. 2024, doi: 10.3390/buildings14113551.

[8] L. Hang and D. H. Kim, "Reliable task management based on a smart contract for runtime verification of sensing and actuating tasks in IoT environments," *Sensors (Switzerland)*, vol. 20, no. 4, Feb. 2020, doi: 10.3390/s20041207.

[9] F. N. binti M. Shah, Z. binti Abdullah, A. binti Dahlan, and N. binti M. Dahlan, "Web-Based Task Management System for Improving Group Work Collaboration," *International Journal of Academic Research in Progressive Education and Development*, vol. 11, no. 3, Jul. 2022, doi: 10.6007/ijarped/v11-i3/15188.

[10] M. Chen, T. S. Martins, L. Zhang, and H. Dong, "Digital Transformation in Project Management: A Systematic Review and Research Agenda," 2025, doi: 10.3390/systems13080625.

[11] J. Brooke, "SUS-A quick and dirty usability scale," 1996.

[12] A. Macris, S. Drafta, Ştefania Martiniuc, and A. E. Petre, "Comparative Usability Evaluation of Three Digital Smile Design Software Tools Using the System Usability Scale," *Dent. J. (Basel).*, vol. 13, no. 9, Sep. 2025, doi: 10.3390/dj13090418.

[13] R. Wafa, M. Q. Khan, F. Malik, A. B. Abdusalomov, Y. I. Cho, and R. Odarchenko, "The Impact of Agile Methodology on Project Success, with a Moderating Role of Person's Job Fit in the IT Industry of Pakistan," *Applied Sciences (Switzerland)*, vol. 12, no. 21, Oct. 2022, doi: 10.3390/app122110698.

[14] H. A. Kheder, "An Innovative Approach to Enhancing Software Development Efficiency Through Agile Methodologies," *Kufa Journal of Engineering*, vol. 16, no. 2, pp. 328–343, Apr. 2025, doi: 10.30572/2018/KJE/160220.

[15] D. S. Ramadhani, N. Aulia, A. S. Putri, M. T. Nugroho, A. S. A. Sakti, and R. D. A. P. Satriyono, "Implementation of Agile Software Development in the Design of Management Information Systems Inventory and Finance at P2MKP Alang-Alang Tumbuh Subur," *Engineering Proceedings*, vol. 84, no. 1, Feb. 2025, doi: 10.3390/engproc2025084062.

[16] J. Vitorino, T. Dias, T. Fonseca, E. Maia, and I. Praça, "Constrained adversarial learning for automated software testing: a literature review," May 01, 2025, *Springer Nature*. doi: 10.1007/s42452-025-07073-3.

[17] M. Hyzy *et al.*, "System Usability Scale Benchmarking for Digital Health Apps: Meta-analysis," *JMIR Mhealth Uhealth*, vol. 10, no. 8, Aug. 2022, doi: 10.2196/37290.

[18] J. Nielsen and T. K. Landauer, "A Mathematical Model of the Finding of Usability Problems," in *Proc. INTERACT'93 and CHI'93 Conf. Human Factors Comput. Syst.*, ACM, Amsterdam, May 1993, pp. 206–213, doi: 10.1145/169059.169166.

[19] T. S. Tullis and J. N. Stetson, "A Comparison of Questionnaires for Assessing Website Usability," in *Proc. Usability Professionals Association (UPA) Annual Conf.*, Minneapolis, MN, USA, Jun. 2004, pp. 1–12.

[20] J. R. Lewis, "The System Usability Scale: Past, Present, and Future," *Int. J. Hum. Comput. Interact.*, vol. 34, no. 7, pp. 577–590, Mar. 2018, doi: 10.1080/10447318.2018.1455307.

[21] J. R. Lewis and J. Sauro, "Excel and R Companion to the Second Edition of 'Quantifying the User Experience: Practical Statistics for User Research' Rapid Answers to over 100 Examples and Exercises," 2016.

[22] K. S. Taber, "The Use of Cronbach's Alpha When Developing and Reporting Research Instruments in Science Education," *Res. Sci. Educ.*, vol. 48, no. 6, pp. 1273–1296, Dec. 2018, doi: 10.1007/s11165-016-9602-2.

[23] L. T. Hsieh, H. Liu, Y. Guo, and R. Gazda, "Task Management for Cooperative Mobile Edge Computing," in *Proceedings - 2020 IEEE/ACM Symposium on Edge Computing, SEC 2020*, Institute of Electrical and Electronics Engineers Inc., Nov. 2020, pp. 352–357. doi: 10.1109/SEC50012.2020.00051.

[24] N. Alesandro, F. Maulana, and L. Figo Lim, "Journal of Artificial Intelligence and Engineering Applications Development of a Campus Project Management System to Enhance Collaboration Using the Agile Method," 2025. [Online]. Available: https://ioinformatic.org/

[25] R. Wandri *et al.*, "Agile Scrum as a Development Approach: A Case Study of Web-based School Information System Design," 2025. [Online]. Available: http://sistemasi.ftik.unisi.ac.id

[26] S. G. Johnson, T. Potrebny, L. Larun, D. Ciliska, and N. R. Olsen, "Usability Methods and Attributes Reported in Usability Studies of Mobile Apps for Health Care Education: Scoping Review," Jun. 01, 2022, *JMIR Publications Inc.* doi: 10.2196/38259.

[27] J. Pasuksmit, P. Thongtanunam, and S. Karunasekera, "Story points changes in agile iterative development: An empirical study and a prediction approach," *Empir. Softw. Eng.*, vol. 27, no. 6, Nov. 2022, doi: 10.1007/s10664-022-10192-9.

[28] C. Lektip *et al.*, "The Development and Evaluation of the Application for Assessing the Fall Risk Factors and the Suggestion to Prevent Falls in Older Adults," *Informatics 2025, Vol. 12, Page 53*, vol. 12, no. 2, p. 53, Jun. 2025, doi: 10.3390/INFORMATICS12020053.

[29] F. D. Davis, "Perceived usefulness, perceived ease of use, and user acceptance of information technology," *MIS Q.*, vol. 13, no. 3, pp. 319–339, 1989, doi: 10.2307/249008.