



Analysis of the Use of Cloud-Based Collaborative Tools (Notion and Google Docs) for Software Planning Optimization: A Literature Review

Bryant Tinambunan^{1*}, Zulfahmi Indra², Alya Namira³, Adinda Soleha⁴

^{1,2,3,4}Computer Science, Universitas Negeri Medan

*Jl. Willem Iskandar, Ps. V, Medan Estate, Kec. Percut Sei Tuan, Kab. Deli Serdang, Kota Medan – Sumatera Utara
bryanttinambunan12@gmail.com¹, Zulfahmi.indra@unimed.ac.id², alvanamira3010@gmail.com³, adindasoleha64@gmail.com⁴*

Abstract

Software planning is a crucial phase in the Software Development Life Cycle (SDLC) that significantly determines project success. However, in practice, development teams often face challenges such as miscommunication, unclear task distribution, and unstructured documentation. The advancement of cloud-based technology offers solutions through collaborative tools such as Notion and Google Docs. This study aims to analyze the utilization of cloud-based collaborative tools in optimizing software planning through a literature review approach. The research employs a qualitative method by examining relevant academic sources, including journals and scholarly publications. The results indicate that Notion plays a role in task management, progress tracking, and project information organization, while Google Docs effectively supports documentation and real-time collaborative writing. The combined use of these tools enhances team communication, clarifies task structure, and improves planning efficiency. Therefore, cloud-based collaborative tools provide a positive contribution to optimizing software planning processes.

***Keywords:** software planning, SDLC, collaborative tools, cloud computing, Notion, Google Docs*

1. Introduction

Software planning is one of the most fundamental stages in the Software Development Life Cycle (SDLC). Without proper planning, software projects face a high risk of failure, including schedule delays, cost overruns, and outcomes that do not meet user requirements. According to Sommerville, poor planning is one of the primary causes of software project failure worldwide. In practice, software development teams often encounter various challenges during the planning phase. Miscommunication among team members, unclear task allocation, and decentralized documentation are among the most frequently reported issues [1]. These challenges have been further intensified by the growing trend of remote working following the COVID-19 pandemic, where team coordination has become increasingly complex due to members being distributed across different geographical locations [2].

The advancement of cloud-based technologies has created new opportunities to address these challenges. Cloud computing services enable the storage, management, and sharing of information in real time via the internet, without dependence on specific devices or physical locations. In the context of software development, this technology has led to the emergence of various collaborative tools designed to support online teamwork. Among the cloud-based collaborative tools widely used by software development teams today are Notion and Google Docs. Notion is a knowledge management and productivity platform that allows teams to create databases, task boards, project roadmaps, and technical documentation within a single integrated interface. Meanwhile, Google Docs provides convenient real-time document collaboration through features such as commenting, revision history, and extensive document-sharing capabilities [3].

Although numerous studies have examined collaborative tools, most existing research focuses on general productivity or knowledge management aspects, with limited attention given to the planning phase of the software development lifecycle. There is a significant research gap regarding how tools such as Notion and Google Docs can specifically optimize software planning processes as a foundation of the SDLC [4]. Comprehensive and focused studies on this topic remain limited, particularly in Indonesian academic literature. Based on this background, this study aims to analyze how cloud-based collaborative tools, specifically Notion and Google Docs, can support the optimization of software planning through a review of relevant literature. By employing a literature review approach, this research is expected to provide a comprehensive and evidence-based understanding of the role of these tools in the planning stage of the SDLC. This research is important because it offers both theoretical and practical contributions. Theoretically, it enriches academic discussions on

collaborative tools in software engineering. Practically, the findings can serve as a reference for software development teams in both industry and academic environments when selecting and utilizing appropriate tools to support their project planning processes [5].

2. Literature Review

2.1. Software Planning

Software planning is a systematic process carried out before and during software system development, with the objective of defining the project's direction, scope, resources, and schedule in a structured manner. Software planning encompasses several essential activities, namely: (1) requirements analysis and documentation, (2) project scheduling, (3) human and technical resource planning, and (4) risk identification and mitigation. The role of planning in the Software Development Life Cycle (SDLC) is crucial and decisive. The quality of planning directly influences the success of subsequent stages in the development lifecycle, including design, implementation, testing, and maintenance. Without effective planning, development teams may struggle to manage project complexity, resulting in a higher likelihood of scope creep, schedule delays, and budget overruns.

In a collaborative team environment, effective planning requires clear communication, structured documentation, and transparent progress-tracking mechanisms. [3] found that teams with well-established planning systems supported by appropriate tools demonstrate significantly higher project success rates than teams that rely on informal planning practices. This finding highlights the importance of selecting suitable tools to support software planning activities. Software planning activities can generally be categorized into three main dimensions. First, requirements planning, which involves identifying, documenting, and validating both functional and non-functional system requirements. Second, schedule planning, which includes breaking down projects into smaller tasks, estimating task durations, and establishing project milestones. Third, resource planning, which covers the allocation of personnel, hardware, software, and budgets required throughout the development process.

2.2. Software Development Life Cycle (SDLC)

The Software Development Life Cycle (SDLC) is a process framework that defines the stages involved in developing a software system from its inception to the end of its operational life. According to Balaji and Murugaiyan 2012, SDLC generally consists of the following major phases: planning, requirements analysis, system design, implementation (coding), testing, deployment, and maintenance. Among all SDLC phases, the planning stage occupies the earliest and most strategic position. Critical decisions made during this stage, such as defining project scope, selecting a development methodology, and establishing the development team, form the foundation for all subsequent activities [5]. Errors made during the planning phase tend to have greater consequences and are more costly to correct than those identified in later stages of development.

Various SDLC models have been developed to accommodate different project requirements. The Waterfall model, introduced by Royce in 1970, was the first formally documented SDLC model and emphasizes the complete completion of one phase before proceeding to the next. In this model, comprehensive planning at the beginning of the project is considered essential. Subsequent developments gave rise to more adaptive approaches, such as Agile methodologies, which were popularized through the Agile Manifesto. Within Agile frameworks, planning is conducted iteratively and incrementally through short cycles known as sprints or iterations. Nevertheless, planning remains a core activity in each cycle, during which the team defines sprint goals, prioritizes the backlog, and allocates tasks among team members.

It is important to understand that the quality of the planning stage within the SDLC is strongly influenced by the effectiveness of communication and collaboration among team members. Studies have shown that communication barriers are among the most significant risk factors in software projects, particularly during the early planning phase. Therefore, adopting tools that facilitate effective communication and collaboration has become increasingly relevant and necessary, especially in an era characterized by decentralized and digitally driven work environments. According to [4], the digital transformation accelerated by the COVID-19 pandemic has fundamentally changed the way software development teams collaborate. The shift toward hybrid and remote working models has driven the widespread adoption of cloud-based tools, creating both new opportunities and new challenges in managing the planning phase of the SDLC. In this context, the role of collaborative tools such as Notion and Google Docs has become increasingly significant and worthy of further investigation.

2.3. Cloud-Based Collaborative Tools

Cloud-based collaborative tools are software applications that enable team members to work together online regardless of geographical location or time zone differences. According to state that collaborative tools allow teams to remain connected, work together, and achieve collective goals even though software development activities are rarely conducted in a single physical location. Meanwhile, Vetian et al. [7] explain that human resources in the Society 5.0 era are required to possess strong digital competencies, including the ability to manage cloud-based information and collaborate effectively.

The main advantages of cloud-based collaborative tools can be observed from several aspects. First, these tools support real-time collaboration, allowing team members to edit documents, discuss project features, and coordinate development activities simultaneously. Second, The use of cloud-based collaborative tools also contributes to improving team coordination efficiency. By providing a centralized digital workspace, all team members can access the same information quickly and accurately. This helps reduce miscommunication, duplicated work, and delays in information sharing, which are common challenges in software project management [6].

Cloud-based platforms facilitate cross-functional collaboration by integrating communication, artifact management, and task management within a single ecosystem. Furthermore, their high accessibility enables users to access project information from various devices and locations as long as an internet connection is available, making them suitable for both hybrid and remote working environments [7].

2.4. Notion in Software Planning

Notion is a multifunctional cloud-based productivity and project management platform that integrates task management, knowledge management, documentation, and project monitoring into a single workspace. According to Vetian et al. [7], Notion serves as a centralized digital workplace that supports project management while simultaneously functioning as an organizational knowledge repository. In the context of software planning, Notion provides various features that assist development teams in organizing their work systematically. Through Kanban Boards, team members can categorize tasks according to their progress status, enabling easier monitoring of project activities. In addition, the database feature allows project information to be stored in a structured and searchable manner. Notion also supports documentation through wiki pages, which can be used to store standard operating procedures (SOPs), system requirements, meeting notes, and other project-related information [7].

The implementation of Notion in software planning offers several benefits, including improved workflow transparency, better task distribution, and enhanced accountability among team members. Vetian et al. [7] reported that the implementation of Notion in digital organization management training increased participants' understanding of project management concepts from an average pre-test score of 35.0 to a post-test score of 85.0. These findings indicate that Notion has strong potential as a tool for supporting project planning and coordination activities. In addition to its comprehensive features, Notion provides a free personal plan that is sufficient for individual users and small teams. However, the platform requires a certain learning curve, particularly for users who are unfamiliar with relational databases and advanced workspace organization concepts. Therefore, initial training and guidance are recommended to ensure that team members can fully utilize the platform's capabilities [7].

2.5. Google Docs in Software Planning

Google Docs is a cloud-based document editor that forms part of the Google Workspace ecosystem. In software planning, Google Docs is commonly used for creating requirements documents, project proposals, progress reports, and various forms of technical documentation. Jackson et al. [6] define Google Docs as a shared document-editing platform designed to support both synchronous and asynchronous collaboration among development team members. One of the primary advantages of Google Docs is its ability to facilitate real-time collaboration. Multiple users can access and edit the same document simultaneously without the need for manual file merging. Features such as comments, suggestions, and revision history support efficient communication, discussion, and document review processes among team members [6].

During the software planning phase, Google Docs can be utilized to develop Software Requirements Specifications (SRS), record meeting outcomes, document project decisions, and collect feedback from stakeholders. Centralized document storage ensures that all team members work with the same information, thereby minimizing inconsistencies and misunderstandings [6]. Furthermore, Jackson et al. [6] emphasized that the integration of Google Docs with other collaborative tools contributes to a more efficient and organized workflow within software project planning activities. Research conducted by Vetian et al. [7] demonstrated that the integrated use of Google Workspace significantly improves document collaboration effectiveness compared to conventional approaches. This was evidenced by an increase in participants' understanding from an average score of 50.0 before training to 90.0 after training.

3. Research Methodology

This study employs a qualitative research approach using a literature review method. The approach was selected to obtain a comprehensive understanding of the utilization of cloud-based collaborative tools, particularly Notion and Google Docs, in supporting software planning activities. Research data were collected from relevant academic literature and subsequently analyzed and synthesized to provide a systematic overview of the role of these tools within software development environments.

3.1. Research Design

This research is qualitative in nature and adopts a literature review design. The literature review method was chosen to identify, examine, and analyze previous studies related to the use of cloud-based collaborative tools in software planning. This approach enables researchers to gain an in-depth understanding of the concepts, benefits, and implementation challenges associated with Notion and Google Docs without conducting primary data collection.

3.2. Data Sources

The data sources used in this study consist of scientific literature obtained from journals, conference proceedings, and academic publications relevant to the research topic. The selected literature was evaluated based on its relevance to cloud-based collaborative tools, its connection to software planning activities, and the recency of publication. Specifically, this study utilizes the work of Jackson et al. [6], which discusses collaboration tools for software developers, and the study by Vetian et al. [7], which examines the implementation of Google Workspace and Notion in supporting digital organizational management. These sources were selected because they provide both theoretical foundations and empirical evidence relevant to the objectives of this research.

3.3. Data Analysis Technique

The data analysis process was conducted in three stages. The first stage involved reading and reviewing all collected literature to understand the objectives, methodologies, and key findings of each study. During this stage, information directly related to the use of cloud-based collaborative tools in software planning was identified and documented. The second stage involved categorizing the collected information according to the main themes of the study. The themes included the concept of cloud-based collaborative tools, the use of Notion in software planning, and the role of Google Docs in documentation and team collaboration. This categorization facilitated the synthesis and organization of findings in a systematic manner.

The third stage involved comparing and integrating findings from different sources. Jackson et al. [6] provide insights into the role of collaborative tools within software development environments, while Vetian et al. [7] offer empirical evidence regarding the implementation of Google Workspace and Notion in organizational settings. The findings from both studies were synthesized to develop a comprehensive analysis of the effectiveness of cloud-based collaborative tools in optimizing software planning processes.

4. Results and Discussion

4.1. Literature Review Results

The literature reviewed in this study consistently indicates that cloud-based collaborative tools contribute positively to software planning activities. The selected studies emphasize several recurring themes, including communication enhancement, documentation management, task coordination, project visibility, and support for distributed software development teams.

To provide a clearer overview of the findings, Table I summarizes the main contributions of the reviewed studies.

Table 1: The main contributions of the reviewed studies

Reference	Research Focus	Key Findings	Relevance to Software Planning
Utomo & Kurniawan [1]	Collaborative learning and teamwork	Effective collaboration improves coordination and task completion	Supports communication and team coordination
Fitrian et al. [2]	Cloud storage utilization	Cloud services facilitate information sharing and accessibility	Supports centralized planning and documentation
Al Hazmi et al. [3]	Project management systems	Structured project management improves development efficiency	Supports planning and task management
Zulman et al. [4]	Cloud-based collaboration systems	Cloud platforms improve organizational collaboration	Supports distributed software planning
Hidayat et al. [5]	Software project management systems	Project visibility improves planning quality	Supports monitoring and scheduling
Jackson et al. [6]	Developer collaboration tools	Collaborative tools enhance communication and coordination	Supports planning in distributed teams
Vetian et al. [7]	Google Workspace and Notion implementation	Digital collaboration tools improve productivity and understanding	Supports documentation and project organization

Based on Table I, communication, documentation, and project coordination emerge as the three most frequently discussed factors influencing software planning effectiveness. These findings suggest that cloud-based collaborative tools address several fundamental challenges encountered during the planning phase of software development projects.

4.2. Analysis of the Role of Notion in Software Planning

The literature indicates that Notion functions as a comprehensive workspace that integrates documentation, task management, and project monitoring within a single platform. Unlike conventional documentation tools, Notion provides database structures, Kanban boards, calendars, and timeline views that allow teams to organize planning activities systematically. In software planning, Notion supports requirement management, sprint planning, roadmap development, and progress tracking. Through its database functionality, teams can store project artifacts in a structured manner and establish relationships between requirements, tasks, and project milestones. This capability enhances information traceability and reduces the risk of information fragmentation.

Vetian et al. [7] reported that the implementation of Notion significantly improved participants' understanding of project management practices. This finding suggests that Notion not only serves as a technical tool but also contributes to improving organizational planning capabilities. Furthermore, Notion provides transparency across project activities because all team members can access updated information through a centralized workspace. This transparency is particularly valuable in Agile software development environments where planning activities are iterative and continuously updated. However, despite its advantages, Notion requires users to understand workspace organization concepts and database structures. Consequently, new users may experience an initial learning curve before they can fully utilize the platform's capabilities.

4.3. Analysis of the Role of Google Docs in Software Planning

Google Docs plays a complementary role in software planning by supporting collaborative documentation processes. Unlike Notion, which focuses on project organization and task management, Google Docs primarily facilitates document creation, editing, and review.

The literature highlights that Google Docs is widely used for preparing Software Requirements Specifications (SRS), meeting minutes, project proposals, and planning reports. Its real-time editing capabilities allow multiple stakeholders to contribute simultaneously, thereby accelerating the review and approval process. Jackson et al. [6] emphasized that collaborative documentation tools reduce communication barriers and improve knowledge sharing among software development teams. Through features such as comments, suggestions, and revision history, Google Docs enables transparent discussions while maintaining a complete record of document changes.

Moreover, integration with Google Workspace enhances accessibility and allows project stakeholders to participate regardless of their geographical location. This characteristic is particularly important in remote and hybrid work environments where software teams operate across different regions. Nevertheless, Google Docs has limited support for project visualization and task management. Therefore, it is generally more effective when used alongside project management platforms such as Notion.

4.4. Comparative Analysis of Notion and Google Docs

Although both tools support collaboration, their contributions to software planning differ significantly.

Table 2: Comparison Of Notion and Google Docs in Software Planning

Aspect	Notion	Google Docs
Primary Function	Project and knowledge management	Collaborative documentation
Task Management	Excellent	Limited
Project Visualization	Kanban, Timeline, Calendar	Not Available
Real-Time Collaboration	Available	Excellent
Documentation Management	Excellent	Excellent
Progress Tracking	Available	Limited
Learning Curve	Moderate	Low
Integration Capability	High	High

Table II demonstrates that Notion is more suitable for planning and project management activities, whereas Google Docs is more effective for collaborative documentation. Therefore, the combined use of both tools provides a more comprehensive planning environment than relying on either tool individually.

4.5. Impact of Cloud-Based Collaborative Tools on Software Planning

The synthesis of the reviewed literature reveals that cloud-based collaborative tools influence software planning through three major dimensions: communication, documentation, and coordination.

1. Communication Enhancement

Collaborative tools reduce communication gaps by allowing information exchange in real time. This capability minimizes misunderstandings and ensures that planning decisions are shared consistently among team members.

2. Documentation Quality Improvement

Centralized document repositories help teams maintain accurate and up-to-date project documentation. This reduces information loss and improves knowledge retention throughout the project lifecycle.

3. Project Coordination Optimization

Task visualization features and collaborative workspaces improve team coordination by clarifying responsibilities, deadlines, and project priorities. Consequently, project planning becomes more structured and transparent.

These findings align with previous studies [6], [7], which conclude that cloud-based collaboration technologies significantly improve teamwork effectiveness and project planning performance.

4.6. Discussion and Research Implications

The findings of this study indicate that software planning is no longer solely dependent on managerial expertise but is increasingly influenced by the availability of digital collaboration technologies. As software development environments become more distributed, collaborative tools play an essential role in ensuring planning effectiveness.

The integration of Notion and Google Docs provides complementary benefits throughout the planning phase of the SDLC. Notion supports planning structure and project visibility, while Google Docs facilitates collaborative documentation and stakeholder communication. Together, these tools create a unified planning ecosystem capable of addressing many common challenges in software development projects.

From a practical perspective, software development teams are encouraged to adopt a combination of project management and collaborative documentation tools rather than relying on isolated solutions. From an academic perspective, future research may extend this study by conducting empirical evaluations involving software development teams to quantitatively measure the impact of these tools on project planning performance.

5. Conclusion

Based on the results of the literature review, it can be concluded that cloud-based collaborative tools, particularly Notion and Google Docs, play a significant role in supporting software planning optimization. These tools are capable of improving communication quality, clarifying task allocation, and supporting more structured documentation.

Notion serves as an integrated project management platform, while Google Docs functions as an effective collaborative documentation tool. The combination of these two tools provides a comprehensive solution during the SDLC planning phase. Therefore, the use of cloud-based collaborative tools has been proven to enhance the efficiency, transparency, and success of software project planning.

References

- [1] D. W. Utomo and D. Kurniawan, "Formasi kelompok dinamis untuk mendukung kolaborasi pembelajaran proyek perangkat lunak," *Jurnal Inovasi Teknologi Pendidikan*, vol. 7, no. 1, pp. 42–51, Aug. 2020, doi: 10.21831/jitp.v7i1.31378.
- [2] H. P. Fitriani, I. Kusdiawati, R. R. Nursoleh, R. Ramdani, and A. Rois, "Pemanfaatan Google Drive sebagai Layanan Cloud Computing untuk Penyimpanan Kolaborasi Data," *TAMIKA: Jurnal Tugas Akhir Manajemen Informatika & Komputerisasi Akuntansi*, vol. 5, no. 2, pp. 433–439, Dec. 2025, doi: 10.46880/tamika.Vol5No2.pp433-439.
- [3] M. Fariz Al Hazmi, U. Delfana Rosiani, and N. Santoso, "RANCANG BANGUN SISTEM MANAJEMEN PROYEK DAN KOLABORASI TIM DEVELOPER," 2015.
- [4] M. Reza Zulman, M. Arhami, J. Teknologi Informasi Dan Komputer Politeknik Negeri Lhokseumawe, and J. Teknik Kimia Politeknik Negeri Lhokseumawe anwarsy, "Journal of Artificial Intelligence and Engineering Applications Development of a Cloud Computing-Based Collaboration Information System," 2024. [Online]. Available: <https://ioinformatic.org/>
- [5] R. Hidayat, A. Arwan, and A. P. Kharisma, "Pengembangan Sistem Manajemen Proyek Perangkat Lunak (Studi Kasus: CV. Karya Studio Teknologi Digital)," 2021. [Online]. Available: <http://j-ptiik.ub.ac.id>
- [6] V. Jackson, A. van der Hoek, R. Prikladnicki, and C. Ebert, "Collaboration Tools for Developers," *IEEE Software*, vol. 39, no. 2, pp. 7–15, 2022. Doi: 10.1109/MS.2021.3132137
- [7] R. A. Vetian, A. A. Fajrin, A. Leony, and J. B. Lim, "Peningkatan Kompetensi Digital Melalui Pelatihan Collaborative Tools (Google Workspace & Notion) untuk Manajemen Organisasi," *JUAN: Jurnal Pengabdian Nusantara*, vol. 3, no. 1, pp. 53–59, 2026. doi: <https://doi.org/10.63545/juan.v3.i1.202>