Applying PMBOK for Scope Definition and Schedule Estimation in Learning Management System Development: A Case Study at STIKES Pemkab Jombang

Vania Faustina Gunawan[1], Koko Wahyu Prasetyo[2*]

[1, 2] *Faculty of Science and Technology, Universitas Bhinneka Nusantara, Malang, Indonesia*

Keywords
scope management; schedule management; three-point estimation; software project management; pmbok;

Correspondence Email:
koko@ubhinus.ac.id

Abstract
The phenomenon of Fear of Missing Out (FoMO) represents one of the psychological impacts that occurs in the era of social media, especially among Generation Z. The TikTok platform, with its algorithm system that focuses on personalization and user interaction, plays an important role in shaping and strengthening this phenomenon. This study aims to analyze the role of the TikTok's algorithm in forming the FoMO phenomenon among Generation Z using media effects theory as a scientific basis. The research employs a descriptive qualitative through content analysis, focusing on viral TikTok content throughout the year 2025 that potentially triggers FoMO and the algorithmic influence on user psychology. The results of the analysis show that the TikTok algorithm works continuously to regulate the distribution of content based on user tastes and interactions, which in turn creates dependency. This process strengthens the psychological drive to stay updated with trends, heightens the fear of missing out, and leads to emotional consequences such as anxiety, stress, and reduced self-esteem. Furthermore, the For You Page (FYP) feature, which prioritizes popular content, accelerates the dissemination of viral trends and intensifies the FoMO effect among Generation Z. This process reinforces the psychological urge to keep up with trends, heightens the fear of missing out, and leads to emotional consequences such as anxiety, stress, and decreased self-esteem. Additionally, the FYP feature, which is based on content popularity, accelerates the dissemination of viral trends and intensifies the FoMO effect among Generation Z. Thus, this study explains that the TikTok algorithm plays an important role in creating and reinforcing the FoMO phenomenon through a content recommendation mechanism that is dependent and interaction-based, thereby impacting the social and psychological behavior of the younger generation in the digital era.

1. Introduction

Digital transformation has become a strategic priority in higher education, especially after the COVID-19 pandemic accelerated the adoption of online learning. Many institutions now rely on digital platforms to manage instructional content, assessments, and student engagement. However, several organizations still depend on general-purpose tools that lack integration and governance capabilities. STIKES Pemkab Jombang, a health sciences college in Indonesia, previously used Google Classroom as its main learning platform. Although functional for basic teaching activities, the system provided limited data tracking, weak integration with the academic information system, and minimal administrative control. These issues created operational inefficiencies and hindered the institution's goal of achieving a fully digital learning environment.

To address these limitations, STIKES Pemkab Jombang planned the development of an institutional Learning Management System (LMS) based on the open-source Moodle platform. The new LMS was expected to consolidate teaching materials, automate attendance and grading, and provide centralized reporting for academic administrators. The project followed a conventional Waterfall approach to ensure clear documentation and phased deliverables. Despite the technical feasibility, the institution faced a common challenge in educational IT projects: inadequate project planning. Many similar initiatives fail because of unrealistic schedules, underestimated workloads, or poorly defined project scope.

The Project Management Body of Knowledge (PMBOK) offers a globally recognized framework that supports structured project planning. Its process-based approach guides project teams through defining scope, estimating resources, and developing time-bound schedules. Applying PMBOK can help educational institutions avoid ad-hoc planning and establish measurable baselines for implementation. This study therefore adopts the PMBOK 6th Edition as a methodological reference for planning the LMS development project at STIKES Pemkab Jombang.

The objective of this research is to apply PMBOK to define the project scope, estimate the activity durations using the Three-Point Estimating method, and identify the critical path through Critical Path Method (CPM) analysis. The study aims to produce a realistic project schedule and demonstrate how a structured planning framework improves predictability in academic software projects. The paper contributes academically by extending the application of PMBOK to higher-education software development and contributes practically by providing a reference model for small institutions that intend to implement an open-source LMS using systematic project-management principles.

1.1 Literature Review

The Project Management Body of Knowledge (PMBOK Guide) Sixth Edition defines 49 processes across five process groups and ten knowledge areas. Two of these, Project Scope Management and Project Schedule Management, form the foundation of project planning and control. According to the PMBOK Guide (Project Management Institute, 2017), scope management ensures that the project contains all required work and excludes unnecessary work, while schedule management provides guidance for defining, sequencing, and estimating activities. Together, these areas help managers align deliverables with time and resource constraints in software and information system projects.

PMBOK offers a systematic planning structure that improves clarity and accountability in software development. Studies show that applying PMBOK processes enhances the success rate of technology projects by improving documentation, activity sequencing, and schedule predictability. (Nasir et al., 2015) demonstrated that PMBOK's process orientation supports success through clear scope definition and structured scheduling. (Rdiouat et al., 2021) later confirmed that combining PMBOK with Capability Maturity Model Integration (CMMI) strengthens continuous process improvement in software engineering by unifying best practices from both frameworks.

PMBOK also proves valuable in academic environments. (Bayona et al., 2018) applied it to research management projects at universities and found better coordination and measurable progress tracking. Similarly, (Barghoth et al., 2020) proposed a comprehensive software project management framework

integrating PMBOK and CMMI-DEV processes to handle people, process, and technology interactions more effectively. These findings highlight PMBOK's adaptability to different institutional contexts where transparency and structured planning are critical.

Quantitative and parameter-based approaches complement PMBOK's planning rigor. (Chomal et al., 2022) introduced the 4PCDT model for academic software projects and showed that measurable scheduling parameters improve estimation reliability. In parallel, hybrid frameworks have become increasingly important. (Silva & Nuñez, 2023) demonstrated that integrating PMBOK with Agile methods enhances flexibility without losing process discipline, while (Rosenberger & Tick, 2018) found PMBOK 6th Edition suitable for agile-developed IT projects when used at a strategic level.

In Indonesia, PMBOK-related practices have informed IT project management education and institutional systems development. (Purnomo et al., 2023) designed an information system to manage professional internship activities at a private university. The study emphasized systematic coordination of mentoring and project completion using structured process flows. Its method reflected PMBOK's planning philosophy, especially the importance of clear scope and activity sequencing to manage academic workflows efficiently.

Researchers also continue to explore PMBOK's role in risk and performance management. (Reshetnyak et al., 2023) showed that PMBOK provides stability and predictability in IT projects that demand precise control, while (Zambrano et al., 2024) observed that the PMBOK 7th Edition promotes value delivery and performance measurement. Even so, many educational institutions still rely on the 6th Edition because its prescriptive structure supports detailed scheduling and progress monitoring.

In summary, PMBOK remains a reliable reference for planning educational and software development projects. Its focus on defining scope, estimating duration, and identifying critical paths enables predictable and well-documented implementation. The findings reinforce the importance of structured management within academic contexts, while recent studies confirm PMBOK's continued relevance in hybrid and data-driven environments. These insights justify adopting PMBOK 6th Edition for the LMS planning project at STIKES Pemkab Jombang.

2. Research Methods

This study applies an applied case study design focused on the planning phase of a Learning Management System (LMS) project at STIKES Pemkab Jombang. The project followed the Project Management Body of Knowledge (PMBOK Guide) framework to ensure systematic planning and scheduling. The study examines how structured project management can improve predictability in academic IT implementations.

The case study design allows close observation of planning activities in their real institutional context. The research follows PMBOK's Planning Process Group by defining project scope, creating a Work Breakdown Structure (WBS), identifying and sequencing activities, estimating durations, and developing the final project schedule using the Critical Path Method (CPM). This structured process aligns with findings by Bănică et al. (2018), who demonstrated that PMBOK-based case studies improve clarity in task definition and time estimation.

Data were collected from interviews with the project manager of Digital Business Innovation (DBI) and the e-learning coordinator at STIKES, complemented by institutional documents and technical specifications. The data provided accurate information for constructing the activity list, defining dependencies, and validating duration estimates. The study used the Three-Point Estimation technique for each activity, calculating expected durations. Spreadsheet tools were used to compute Early Start (ES), Early Finish (EF), Late Start (LS), and Late Finish (LF), while network diagrams were created manually to visualize dependencies and identify the critical path.

Expert validation ensured the accuracy of estimates and logical sequencing. The external project manager reviewed the schedule and confirmed that the dependencies matched the actual implementation order. This

validation process aligns with recommendations from Gonçalves et al. (2017), who suggested that real-world case studies are essential for testing PMBOK's applicability in software development environments.

3. Result and Discussion

This section presents the main findings of the LMS development project planning at STIKES Pemkab Jombang. The analysis follows the PMBOK sequence for the Planning Process Group, starting with scope definition, followed by activity identification, duration estimation, and critical path analysis.

The planning phase produced a clear set of project deliverables organized into five major phases: Requirement Analysis (RA), System Design (SD), Implementation (IM), Testing (TS), and Deployment & Maintenance (DM). These phases formed the structure of the Work Breakdown Structure (WBS) and provided the foundation for the activity list shown in Table 1. Each activity in the table includes a unique WBS code, description, and person-in-charge (PIC), representing the resource assignment matrix. For example, activities RA1–RA5 describe early planning and documentation led by the project manager and Moodle consultant, while IM1–IM9 cover system configuration, SSO integration, and content setup performed by developers. This decomposition aligns with PMBOK's Define Activities process, ensuring that every deliverable is represented by measurable and accountable work packages.

Table 1 demonstrates that the planning process captured all major tasks required to complete the project. The sequence from requirement elicitation to deployment reflects a linear Waterfall workflow. Such structure supports the argument by Bănică et al. (2018) that PMBOK-based case studies improve clarity in task definition and coordination among stakeholders. In this project, the explicit assignment of PICs reduced role ambiguity and created an operational baseline for time estimation.

Table 1. Project Activity List Definition

WBS ID	Activity Name	Description of Work
RA1	Meeting with stakeholders	Initial discussion to understand system needs from relevant parties.
RA2	Requirement and scope identification	Gather and analyze user needs, define project scope, success indicators, and feasibility.
RA3	Academic data collection	Collect course data, lecturers, study programs, and student data needed for the LMS.
RA4	SRS (Software Requirements Specification) drafting	Compile user needs into an SRS document.
RA5	LMS template research (Moodle)	Review available Moodle templates and select the most suitable one.
SD1	Finalizing Moodle template & project planning	Select a suitable template; outline project plan (schedule, milestones, resources).
SD2	Selecting necessary plugins	Determine additional plugins required to meet user needs.
IM1	Purchase Moodle template	License purchase process for the chosen template.
IM2	Moodle installation	Install Moodle on the server.
IM3	Template customization	Adjust look-and-feel, layout, and campus identity.
IM4	Language & theme configuration	Configure language and visual themes.
IM5	Role & permission setup	Set user roles and access based on roles (admin, lecturer, student).
IM6	SSO (Single Sign-On) implementation	Integrate Moodle with the institution's SSO/UMS.
IM7	Plugin installation & setup	Install and configure required plugins.
IM8	Set up categories and courses	Create course categories and initial course structure per standard.
IM9	Learning backup & restore	Prepare backup files and restore courses.
TS1	User access & SSO testing	Ensure users can log in with valid roles and SSO works correctly.

TS2	Installed plugin testing	Verify each installed plugin functions as expected.
TS3	Course/learning management testing	Test course management, grading, and related functionalities.
TS4	Course backup/restore testing	Verify backup/restore integrity and data consistency.
DM1	System monitoring	Perform system monitoring after go-live, including performance checks.
DM2	System documentation	Prepare technical and configuration documentation.
DM3	Training material preparation	Prepare user training materials (admin/lecturer).
DM4	Conduct training session	Deliver training to lecturers/admins and collect feedback.

After defining the activity list, the research team estimated task durations using the Three-Point Estimation technique with optimistic (O), most likely (M), and pessimistic (P) values. The expected duration for each activity was calculated with the PERT formula. The Critical Path Method (CPM) then determined total duration, early and late start/finish times, and float. The outcomes appear in Table 2, which summarizes the results of the CPM analysis.

According to Table 2, all activities with zero float form the project's critical path, which follows the sequence [RA1-RA5] → SD1 → [IM1-IM9] → TS4 → DM1-DM2-DM4. The total project duration equals 49 working days. Non-critical tasks such as SD2 and TS1–TS3 contain float values between 2 and 7 days, allowing limited scheduling flexibility. The distribution of float confirms that the implementation and testing phases dominate the overall timeline, a pattern consistent with findings by Gonçalves et al. (2017), who emphasized the importance of detailed time estimation for controlling software project performance.

Table 2. Critical Path Time Estimates

WBS ID	Duration	ES	EF	LS	LF	Float	Crit?
RA1	2	0	2	0	2	0	Yes
RA2	2	2	4	2	4	0	Yes
RA3	5	3	8	3	8	0	Yes
RA4	3	8	11	8	11	0	Yes
RA5	1	11	12	11	12	0	Yes
SD1	3	12	15	12	15	0	Yes
SD2	2	15	17	22	24	7	No
IM1	1	15	16	15	16	0	Yes
IM2	1	16	17	16	17	0	Yes
IM3	2	17	19	17	19	0	Yes
IM4	1	19	20	19	20	0	Yes
IM5	2	20	22	20	22	0	Yes
IM6	2	22	24	22	24	0	Yes
IM7	2	24	26	24	26	0	Yes
IM8	1	26	27	26	27	0	Yes
IM9	3	27	30	27	30	0	Yes
TS1	2	24	26	29	31	5	No
TS2	3	26	29	28	31	2	No
TS3	1	27	28	30	31	3	No
TS4	1	30	31	30	31	0	Yes

DM1	5	31	36	31	36	0	Yes
DM2	2	36	38	36	38	0	Yes
DM3	1	36	38	36	38	0	Yes
DM4	1	38	39	38	39	0	Yes

The visualization in Figure 1 illustrates the Critical Path Network Diagram derived from the CPM computation. Red nodes represent critical activities with zero float, while white nodes represent non-critical ones. The diagram reveals a single continuous path from the initial requirement stage to project closure, validating the accuracy of the numeric calculations in Table 2. The network structure also indicates potential parallel execution of SD2 and TS1–TS3 without affecting total project duration. This insight helps project managers allocate human resources more efficiently while maintaining control over key milestones.

From an analytical perspective, the findings of this study demonstrate that the application of PMBOK's Scope and Schedule Management processes in an academic software project yields concrete and measurable outcomes. The hierarchical activity list and time estimation framework ensured consistency between scope definition and scheduling, supporting the project's ability to meet institutional objectives within a realistic time frame.

This result aligns closely with previous empirical studies that emphasized PMBOK's role in improving clarity and coordination during software project planning. Bănică et al. (2018) and Purnomo et al. (2023) observed that PMBOK-based planning in academic web development projects improved task traceability and stakeholder communication. Similarly, Chomal et al. (2022) confirmed that quantifiable and parameter-based scheduling increases accuracy in academic IT projects. The LMS case at STIKES reproduces both effects: clear traceability across activities and improved estimation reliability through the Three-Point Estimation method.

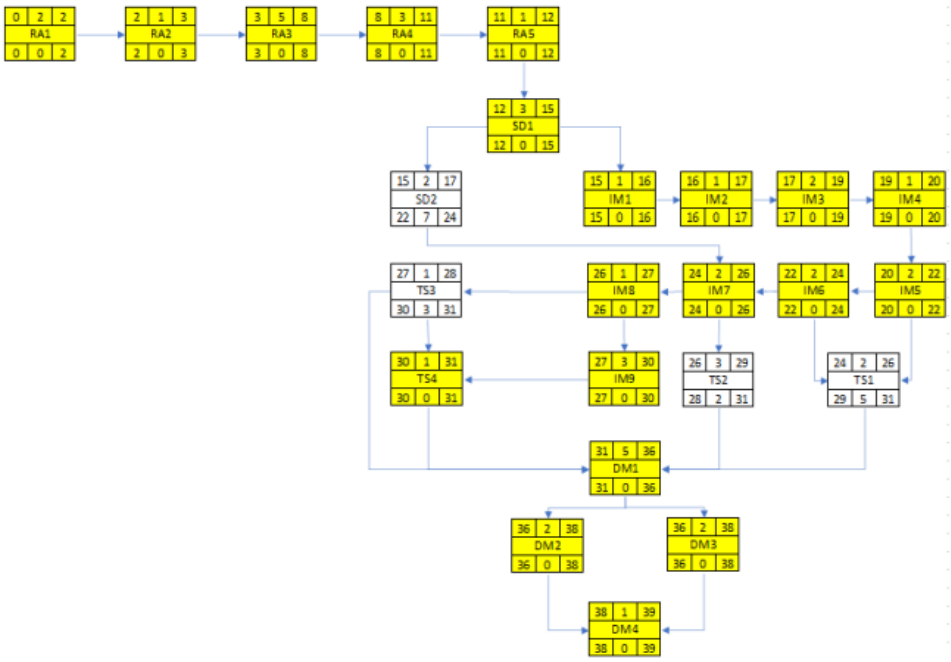

Fig. 1 Critical Path Network Diagram

Furthermore, Rdiouat et al. (2021) demonstrated that integrating PMBOK with CMMI principles enhances process maturity and continuous improvement in software engineering environments. The LMS planning

framework at STIKES shows similar process discipline through documentation and expert validation but remains simpler and more context-specific, suggesting PMBOK's adaptability to smaller organizational scales.

The results also resonate with Reshetnyak et al. (2023), who underlined PMBOK's value for projects requiring predictability and risk control. The STIKES project demonstrates how detailed scheduling and critical-path mapping improve control over time-sensitive activities without relying on probabilistic risk modeling. However, the present case study did not assess how risk management could interact with schedule variation, which remains an area for further exploration.

Overall, the comparative analysis confirms the robustness of PMBOK as a foundation for educational IT planning while reflecting the contextual constraints of small institutions. Yet, this study also reveals several research gaps. First, most PMBOK-based academic projects focus primarily on the planning phase, empirical evidence from execution and monitoring stages remains scarce. Second, although hybrid PMBOK-Agile models are increasingly discussed, few studies test them in real educational settings where organizational hierarchy and curriculum schedules influence project flexibility. Third, quantitative validation using Monte Carlo simulation or schedule-risk modeling could extend the reliability of time estimation beyond deterministic CPM results.

Future research should therefore expand into multi-phase analyses that examine how planned schedules perform during execution and how adaptive mechanisms can be integrated without compromising PMBOK's structured rigor. Such studies would bridge the gap between static planning and dynamic implementation, advancing PMBOK's applicability to the evolving landscape of educational software project management.

4. Conclusions

This study applied the Project Management Body of Knowledge (PMBOK) framework to plan a Moodle-based Learning Management System (LMS) at STIKES Pemkab Jombang. The framework enabled the team to define a complete project scope, sequence activities, and estimate durations using the Three-Point Estimation and Critical Path Method. The resulting schedule produced a 49-working-day plan supported by expert validation, ensuring realistic task dependencies and resource allocation.

The findings confirm that PMBOK's planning processes provide a reliable structure for academic software projects, ensuring alignment between deliverables and time control. The approach improved task traceability, estimation accuracy, and stakeholder coordination. These results reinforce previous findings by Bănică et al. (2018) and Chomal et al. (2022) on PMBOK's role in improving schedule clarity, while extending its application to small higher-education institutions in Indonesia.

Academically, this research expands the empirical basis for PMBOK adoption in educational IT planning. Practically, it offers a reference model for institutions that aim to implement open-source LMS platforms through structured project management. The study's limitation lies in its focus on the planning phase. Future research should test the framework during execution and monitoring, integrate hybrid PMBOK-Agile approaches, and apply quantitative simulation techniques to assess schedule risks. Addressing these directions will strengthen the framework's adaptability and validate its effectiveness in dynamic academic project environments.

5. References

Bănică, L., Hagiu, A., Bagescu, A., & Gherghinescu, A. (2018). Designing a website for a recruitment agency with PMBOK methodology. Scientific Bulletin – Economic Sciences, 17(2), 60–67.

Barghoth, M. E., Salah, A., & Ismail, M. (2020). A comprehensive software project management framework. Journal of Computer and Communications, 8(3), 83–96. https://doi.org/10.4236/jcc.2020.83009

Bayona, S., Saboya, N., & Bustamante, J. (2018). Application of PMBOK in the development of research. Proceedings of the 13th Iberian Conference on Information Systems and Technologies (CISTI), 1–5. https://doi.org/10.23919/CISTI.2018.8399340

Chomal, V. S., Saini, J. R., Gaikwad, H., & Kotecha, K. (2022). 4PCDT: A quantifiable parameter-based framework for academic software project management. International Journal of Advanced Computer Science and Applications, 13(1), 136–144. https://doi.org/10.14569/IJACSA.2022.0130136

Gonçalves, E. F., Drumond, G., & Méxas, M. (2017). Evaluation of PMBOK and Scrum practices for software development in the vision of specialists. Independent Journal of Management & Production, 8(5), 569–582. https://doi.org/10.14807/ijmp.v8i5.598

Nasir, M., Sahibuddin, S., Ahmad, R., & Fauzi, S. (2015). How the PMBOK addresses critical success factors for software projects: A multi-round Delphi study. Journal of Software, 10(11), 1283–1300. https://doi.org/10.17706/jsw.10.11.1283-1300

Purnomo, D. A., Tektanto, N. D., Prasetyo, K. W., & Hadi, H. N. (2023). Pengembangan Modul Pengelolaan Praktek Kerja Lapangan Pada Sistem Informasi Akademik Perguruan Tinggi (Studi Kasus: STIKI). J-INTECH, 11(2), 278-286. https://doi.org/10.32664/j-intech.v11i2.998

Project Management Institute. (2017). A Guide to the Project Management Body of Knowledge (PMBOK Guide) (6th ed.). Project Management Institute.

Rdiouat, Y., Nakabi, N., Kahtani, K., & Semma, A. (2021). Towards a new approach of continuous process improvement based on CMMI and PMBOK. Procedia Computer Science, 192, 2703–2712. https://doi.org/10.1016/j.procs.2021.09.036

Reshetnyak, O., Danko, N. I., & Yurchenko, O. K. (2023). Risk management of IT projects: The essence and features of the application of the PMBOK and Agile approaches. Business Inform, 10(1), 366–374. https://doi.org/10.32983/2222-4459-2023-10-366-374

Rosenberger, P., & Tick, J. (2018). Suitability of PMBOK 6th Edition for agile-developed IT projects. Proceedings of the 18th IEEE International Symposium on Computational Intelligence and Informatics (CINTI), 241–246. https://doi.org/10.1109/CINTI.2018.8928226

Silva, F. J. G., & Nuñez, A. V. (2023). Methodological proposal for the integration of agile methodologies and PMBOK. International Journal of Engineering Insights, 1(1), 12–23. https://doi.org/10.61961/injei.v1i1.4

Zambrano, L., Atencio, E., Mariani, C., & Mancini, M. (2024). Integration between PMBOK 7th concepts: A network analysis. Proceedings of the International Conference on Industrial Engineering and Operations Management, 447–456. https://doi.org/10.46254/an14.20240277