Project management system for self-organized student research volunteer task groups

Ignas Sabaliauskas[a] and Kazimieras Bagdonas[a]

[a] *Kaunas University of Technology, K. Donelaičio g. 73, Kaunas, LT-44249, Lithuania*

Abstract

This paper presents a project management framework tailored to volunteer-led research and development (R&D) groups in academic settings, specifically addressing the challenges of transient team membership, varying participant expertise, and high turnover. Drawing on Scrumban, a hybrid of Scrum and Kanban, the proposed approach integrates GitProject for task and workflow management, Obsidian for collaborative documentation, and Microsoft Teams for communication. Through practical examples in the KATH project, a student-driven bioinformatics R&D initiative, the framework demonstrates how a flexible task pull system, combined with a structured knowledge repository and real-time communication channels, can maintain project continuity and knowledge retention despite frequent membership changes. This solution aims to streamline volunteer participation, support iterative progress, and encourage self-organization while accommodating rapid expansions or contractions in team size.

Keywords[1]

Project Management, Digitalization, Open-Source Tools, Self-Organized Volunteer Task Groups

1. Introduction

The research in knowledge and skill acquisition has shown that students' involvement in problem-based learning and practical work on Research and Development (R&D) projects produce significantly better attainment (citation). Furthermore, students' involvement in group projects develops their soft skills and enables them to reach more complex goals than personal projects (citation). Such problem-based projects have been integrated into higher education curricula since the 1960s [3]. However, a class of voluntary extracurricular R&D activities self-managed by students lacks a functioning project management system to accommodate associated challenges (citation).

R&D projects have their inception, starting with a problem or an idea and developing into a project vision and plan. The nature of projects carried out by the Volunteer Task Groups (VTG) is facing specific challenges regarding the formation and purpose determination (citation), assignment of roles and responsibilities (citation), and motivation and management (citation). Furthermore, due to the high attrition and turnover rates, integrating and enabling new team members, knowledge, and skill transfer are crucial for medium to long-term R&D projects (citation). An active team is critical for any project's success, and one way it is structured is through decentralized coordination and self-management (from "The Standard for Project Management"). As Sharon Ryan and Rory V. O'Connor (2013) mentioned in their article about tacit knowledge, any agile team should be self-managed and committed to highly efficient and effective communication.

Faculty of Informatics students at Kaunas University of Technology (KTU) are participating in a VTG focused on developing a software platform (KATH) to assist genetics researchers in utilizing existing open-source DNA data processing and analysis tools. The project's complexity encompasses knowledge and skills in software development and a basic understanding of genetics. The students participating in KATH VTG receive mentorship from KTU, researchers from Massachusetts Eye and Ear, Harvard University, and SME Genomika. The project started at the end

[1]Proccedings Name, Month XX–XX, YYYY, City, Country
EMAIL: email1@mail.com (A. 1); email2@mail.com (A. 2); email3@mail.com (A. 3)
ORCID: XXXX-XXXX-XXXX-XXXX (A. 1); XXXX-XXXX-XXXX-XXXX (A. 2); XXXX-XXXX-XXXX-XXXX (A. 3)

of Q4 of 2023. Since its inception, over 15 students have participated, with an average of 8 active members. Since Q3 of 2024, the VTG has adopted basic Agile management principles, with volunteer students as scrum masters and team leads.

During the first year, the VTG presented and published a paper at a scientific student conference, received an award in the Innovation exhibition "Technorama 2024," and presented a poster at the international scientific conference DAMSS 2024. The team is scheduled to deliver the minimal viable prototype to HU researchers by the end of Q1 2025. However, due to member attrition, limited available time for project activities, and rising complexity of the tasks, a more comprehensive project management methodology is required to facilitate the productive activities of the VTG student members and ensure the continuity of the KATH project.

VTG R&D project requires a set of PM tools or systems that would facilitate the project's progress towards its goal and the development of the member's skills and competencies. Having a similar life cycle compared to any other VTG undertaking, it is critical to look through various viable solutions and highlight the ones that would suit KATH and similar long-term projects. A new model, consisting of six different segments involving four different open-source (further – OS) parts (Zotero, GitProjects, OpenProjects, and Logseq) has been created, which, in theory, could solve the aforementioned issues within KATH as well as other VTGs' projects while still leaving place for improvement and experimentation. The main goal of this paper is to review the existing PM and software solutions for VTG projects to create a PM framework that could assist in achieving project goals despite the challenges arising from the nature of the VTG.

2. Related work

It has been highlighted by [1] that when looking for a proper approach to a project, alternating between Mixed Agile and more traditional project management methodologies can be the best approach, if project and organizational characteristics enable the usage of both PM methodologies. In a statistical analysis of both Scrum and Kanban effects on software development projects concluded that there were no significant differences between tools, except when managing project schedules, in which case, Kanban outshone Scrum [2]. When looking at project methodology and its correlation to success, found out that applying relevant project management methodologies positively affects project success [3]. Applying Agile, as a PM methodology, was also relevant in increasing success factors, as noted [4].

Success factors project management, product and strategic success-wise - have been found to positively correlate with the use of PM software [5]. It has been determined that the appropriate tool for an appropriate methodologies enables project success within the Agile framework, with Jira being the example for the article [6]. A differential impact of PM tools on to the outcomes of the projects, may have significant differences in how the tools interact with Agile as a whole, and if there even is any interaction at all [7]. A framework that eased the selection for a PM software tool has been presented in [8]. It helps indicate required functionalities according to the team's needs. A research by suggests that having a geographically dispersed team has its own limits, and to mitigate communication and coordination costs, strong social connection is the way to go [9].

A bibliometric analysis has been conducted, and it was determined that the three main keywords in the last three decades are knowledge management, innovation and organizational performance, all of which are also mentioned to be critical to knowledge management [10]. The impact of organizations with proper knowledge acquisition appear to have a richer and more diverse knowledge base [11]. In a study [12] recommends a research model in which explicit knowledge (documentation) and tacit knowledge (personal interactions) are framed as a way of transfering system knowledge. In a master's thesis about knowledge management [13], signifies a cyclical flow of knowledge model, in which, movement of collected information is in two directions - first one being in the form of technical knowledge, moving to the users, and the second one, in the form of feedback, returning to the information's sender. The lack of up-to-date information in project documentation is often out-of-sync with any realistic scenario when coding has been noted as a significant challenge for project management [14].

It has been found that the more often collaboration tools were used, the less difficult data management was [15]. However it has been noted that a tool cannot meet every single requirement

and each company needs to analyze the status quo and figure out concrete needs beforehand [16]. A paper in a Journal of Organizational and End User Computing [17], found that knowledge about the online collaboration determinants would be improving university students' learning engagement and educational performance and that explaining thought processes and rationale behind learning should in succession lead to both educators and their students to be more thoughtful and reflective in their own tasks [18].

3. Requirements for the VTG Management System

To create a VTG management system solution from available software tools and platforms, we have identified the following functions that could benefit from integration: 1. project management methodologies, 2. project management software, 3. knowledge and documentation management software, and 4. communication and collaboration tools. After reviewing the available scientific literature on project management and learning lessons from the previous VTG management experience with the KATH project at KTU, we selected the following requirements for each function to evaluate available solutions.

For project management methodology, we have selected the required features that should be fulfilled:

1. Agile Compatibility—Cooperation in VTGs is crucial, as it allows for greater results in shorter time spans while enabling teams to collaborate more effectively [19]. Evaluating compatibility with Agile as a methodology is critical, as Agile allows for building a solid foundation on an otherwise fragile team structure.
2. Flexibility and Adaptability—Adapting to change is a must when it comes to project management, as the environment is rarely static [20], especially when working in VTGs. We decided to measure adaptability and flexibility through situation analysis if the methodology allows us to resolve an issue faster.
3. Decentralized Coordination Support - Having a centralized system for coordination and support increases delays and difficulties in completing complex tasks [21]. The methodology should enable cyclical or personal feedback and remove information relaying.
4. Suitability for Self-Managed Teams - Being self-sufficient enables a VTG to perform better, increases trust from within the organization, and, in critical cases, enhances the variation of outcomes to be skewed towards more positive ones [20]. In our case, suitability is measured if the methodology does not need a director or a higher manager to set up control systems, and processes and tasks can be managed locally.
5. Ease of Adoption for Student Volunteers—a volunteer group is highly likely to lack specific skills [23], and understanding the methodology should be self-explanatory, as it is the basis of any project. Understanding how to perform effectively within the bounds of PM methodology is a must and should be accessible to people from different backgrounds.
6. Scalability for VTGs—Any VTG can greatly increase in size in short periods of time, which is why a selected PM methodology should be able to withstand surges of change in scale [24].

For the project management platform, we have selected the required features that should be fulfilled:

1. Task Management—Project Management Software has to facilitate task management for individual team members and for the project as a whole. The required features for modern PM are defined in [book/paper]. We have selected the Agile framework [25] and are focusing on workflow centered around sprints, with Kanban boards, backlog, etc. The software shall enable these functionalities.
2. Collaboration and Communication—An agile team can communicate in real-time, share files, and, if necessary, hold conference calls [26]. These functionalities are considered crucial in VTGs.
3. Documentation Integration—A team has to have access to store information, either locally or remotely. This requirement is based on the current trends of increasing project duration and

scale, which come with turnover and loss of explicit and tacit knowledge [27]. The software should allow the team to assign files in .pdf, .mg, or .txt formats and attach web pages.

4. Resource & Time Management - Allocating proper time and resources for any task is vital to a project's longevity, as one of the main goals of any project is to deliver results on time [28]. Functions of scheduling tasks through calendars and/or Gantt graphs, project deliverable ETA's, reminders, milestones, etc., is a way to ensure that everything is submitted on time, or at least allows monitoring processes and resolving bottlenecks.
5. User Experience (Ease-of-use)—Integrating new people, especially into VTGs, can be very difficult [29], which is why evaluating software surface complexity is crucial. Because user experience (further UX) is vague in terms of measurement, we consider a positive UX when the tool has a simple user interface and/or a user manual.

The following criteria have been identified for knowledge management and documentation tools:

1. Documentation Capability—The tool must be able to secure information inside local or remote text files. Cloud solutions and databases can enhance this functionality. This criterion is critical, as text files are often the primary way of storing tacit and explicit knowledge [30].
2. Personal Knowledge Management - Leaving notes and comments for themselves or VTG teammates enhances work effectiveness [31], as turnovers and/or rotations between people heavily impede the tempo through increased time spent re-learning nuances and repeated processes. Thus, the option to attach personalized notes and comments is necessary.
3. Collaborative Editing & Sharing—The option to collaboratively edit documents and access and manage a singular database can make a project easier [32]. This functionality is critical nowadays and is a must in any project variation.
4. Bibliographic Management - Managing bibliographies in scientific projects and corresponding contexts enables teams to locate needed articles, papers, and books, making citing easier. It is crucial in academic task-solving scenarios [33]. We consider these criteria met if the knowledge management and documentation tool can gather bibliographic references in a library, search for specific sources, and insert citations straight into a document.
5. Ease of Use—A knowledge management and documentation tool must be simple enough for a team member to be engaged with it [34]. Due to high turnover rates in VTGs, it is critical to have a simple and easy-to-understand documentation system in place for new volunteers to have a low entry point.
6. Integration with PM Tools—Knowledge management tools complement PM tools, so it is critical to have a link between the two. Integrating PM into process documentation enables teams to retrace data and apply solutions for recurring issues. To meet this requirement, a tool has to have an option to connect to a project management system through plugins or direct integration [35].
7. Data Portability (Import/Export) - Data transferability is crucial due to rapid innovation and improvement of existing tools. The option to export or import information allows VTGs to search for more efficient ways to track changes and test solutions for issues [36]. An option to import or export information in the form of .txt, .pdf, or .json files is sufficient.
8. Cost-effectiveness—VTGs often have limited resources, so cost is one of the deciding factors when choosing one tool over another. We consider something cost-efficient if it is either free or can be accessed in limited form without premium functions.
9. Scalability & Flexibility—A knowledge management tool must be able to withstand large data inputs to uphold the universal trend of team growth in size. The documentation tool must also be able to maintain digital traffic for up to 25 people at a time to be viable for a growing team.
10. Security & Privacy—A documentation tool must be secure, both locally and remotely. While disabling systems in case of a breach is often unnecessary, especially for VTGs, authentication and backups are necessary for a team to save progress and protect data. These functionalities are enough for most VTGs' documentation and knowledge management needs [37].

The following criteria have been identified for communication collaboration tools:

1. Real-time Communication—Communication and collaboration must enable team members to call others at any given time, set up online conference meetings, and send messages. This is the standard for many collaboration tools. We have noticed that all of these functionalities are critical to maintaining efficient communication and keeping all members of our team up to speed.
2. Collaboration & Team Management—Seeing who is participating in activities is very useful, especially as it helps out mapping and evaluating team members [25]. Tracking activity is beneficial in Agile, as it enables sprint planning and workload distribution . In KATH, tracking activity and the option to notify others have been among the reasons for choosing current tools for collaboration.
3. Integration with PM Tools—The team's success heavily relies on strong communication integration into project management and its tools. Having a singular interface for collaboration is expected to increase team members' performance, which is why we analyze whether the tool has capabilities for plugins and/or integrated PM functionalities.
4. User Experience - UX is very subjective regarding tool evaluation. Interface simplicity and built-in tutorials can make UX much more enjoyable [25]. Considering these traits is a must in determining if a communication and/or collaboration tool is viable for a VTG, and in our case, it was critical when choosing one over many different options.
5. Scalability & Flexibility—The communication and collaboration tools selected for the VTG must effectively support dynamic team growth, as observed within the KATH project, where member numbers fluctuated significantly. Scalability refers to the tool's capacity to maintain performance and usability when accommodating rapid changes in the number of active contributors. Flexibility ensures the system can adapt to evolving volunteer participation, seamlessly handling transitions between active and inactive members, variable workloads, and diverse tasks without significant administrative overhead.
6. Cost-effectiveness—Volunteer teams often lack financial resources, which is why free and/or low-cost tools take the lead in terms of usage. Due to not having a user cost, free communication and collaboration tools have a more active user base, are optimized to perform well under significant loads, and have ways to integrate other tools into themselves . These criteria are met if the tool is free or accessed in a limited form without premium functions.
7. Security & Privacy—Security and privacy are of great concern nowadays [37], which is why we find it crucial for outsiders to not be able to disrupt communication and collaboration. We consider tools to be secure and/or private if access for inviting new people can be restricted and/or limited, video/audio calls and messages are not deleted after certain periods of time, and there is a way to enable authentication for VTG members.
8. Volunteer-centric—Given the volunteer-driven structure of VTGs, the selected tools must specifically address volunteer availability constraints, transient participation, and varied skill levels. Four key features were identified as critical:
    i. Lightweight Onboarding & Documentation—Tools must provide concise quickstart guides and self-guided tutorials to quickly onboard new volunteers, enabling rapid integration regardless of their background or experience. Clear documentation and intuitive interfaces help mitigate high turnover and reduce the time required for volunteers to reach productivity.
    ii. Automated Task Assignment & Skill Matching—Volunteers' interests, availability, and existing skill sets must be efficiently utilized. Tools should allow the creation and maintenance of volunteer skill profiles and include recommendation engines to automatically suggest relevant tasks, matching volunteer expertise with suitable project needs, thus enhancing engagement and productivity.
    iii. Simple Handover Mechanisms—Due to the frequent turnover within VTGs, mechanisms such as persistent task notes, task checkpointing, and accessible documentation of partial progress are essential. These mechanisms ensure that

volunteers departing the project leave behind clear progress trails, allowing new volunteers to seamlessly continue tasks without redundant efforts.

iv. Low-Friction Exit—Acknowledging volunteer participation's temporary and fluid nature, tools must feature simplified offboarding processes that automatically manage permissions and access revocation, protecting project security and data integrity. Additionally, incorporating optional exit surveys can provide valuable feedback, driving continuous improvement in volunteer management and system usability.

4. Analysis of available solutions for the management of the VTG

4.1. Project management methodologies

Scrum is a structured Agile methodology characterized by defined roles, events, and artifacts. It supports decentralized coordination and is highly suitable for self-managed teams. However, its moderate flexibility, adaptability, and higher initial onboarding complexity may challenge VTG teams with frequent volunteer turnover.

Kanban methodology offers exceptional flexibility and adaptability. It features a visual workflow that clearly communicates task status and progress. Its simplicity makes it highly suitable for volunteer groups, easing adoption for student volunteers. Additionally, Kanban excels in scalability, which is ideal for VTGs experiencing dynamic team growth or frequent changes.

Scrumban combines the structured elements of Scrum with the flexibility of Kanban, providing the highest decentralized coordination support among the evaluated methodologies. This hybrid methodology balances ease of use and suitability for self-managed teams, offering strong adaptability and scalability to VTGs.

4.2. Project management platform

Open Project provides comprehensive features such as task management, Agile workflows, strong documentation integration, resource and time management, and robust security. It is highly scalable and flexible but lacks strong capabilities in data portability.

GitLab PM supports Agile workflows, excellent task management, and strong documentation integration. While scalable, it faces limitations regarding built-in communication features, user experience simplicity, and data portability.

Taiga excels in Agile workflow management, collaboration, and documentation integration. It offers good scalability and flexibility and is cost-effective, making it suitable for volunteer-centric groups.

Redmine features strong task management, Agile workflow support, collaboration tools, and extensive integration capabilities. However, it has limitations regarding data portability and volunteer-centric features.

Git Projects provides excellent task management, Agile workflow support, and strong documentation integration capabilities. It supports scalability, flexibility, and integration but has limited communication features.

Kanboard offers simplicity in task management, good scalability, and flexibility. It integrates easily with other tools but lacks advanced collaboration and documentation integration features.

Jira offers comprehensive Agile workflow capabilities, robust task management, resource management, and integration capabilities. However, it is less cost-effective and lacks intuitive ease of use and volunteer-centric design.

Trello provides intuitive task management, supports Agile workflows, and offers high flexibility and scalability. Despite its simplicity, it lacks advanced documentation integration and robust security features.

ClickUp excels in task and resource management and offers good data portability and security. Its scalability, flexibility, and ease of use make it suitable for volunteer teams, although Agile workflow integration is limited.

Notion excels in user experience, task management, and collaboration capabilities. Its scalability, flexibility, and cost-effectiveness are notable, but it lacks strong Agile workflow integration and data portability.

4.3. Knowledge management and documentation

Obsidian is a local-first knowledge management system with extensive plugin support for integrations (e.g., Zotero, Git). Obsidian is a powerful local-first knowledge management tool that becomes even more versatile through its extensive integrations with project management, reference tracking, and version control systems. Connecting with Zotero enables seamless citation management, making research documentation more structured and accessible. Its Git integration ensures version-controlled note-taking, allowing teams to track changes and collaborate efficiently while maintaining documentation integrity. Synchronization with Nextcloud, Google Drive, or Dropbox provides accessibility across devices while preserving data privacy and security. Plugins like Kanban boards, task tracking, and calendar integrations support AGILE workflows, helping teams align knowledge documentation with project management. Custom API integrations and automation tools like RPA enhance efficiency by streamlining workflows and eliminating repetitive tasks. With its customizable plugin ecosystem, Obsidian can be adapted to fit various team needs, whether for software development, research organization, or structured documentation. Ultimately, it is a centralized, scalable, and flexible solution for teams needing a structured, efficient, and collaborative knowledge management system.

Notion (All-in-One Workspace): Collaborative knowledge and project management tool with a simple API for integrations. Notion is an all-in-one collaborative workspace that combines knowledge management, project tracking, and team collaboration in a highly customizable environment. Its simple API allows seamless integrations with external tools like GitHub, Slack, Google Drive, and Zapier, enhancing workflow automation and data synchronization. With its flexible database system, teams can structure their information using tables, kanban boards, calendars, and linked pages for better organization. Notion's real-time collaboration features enable teams to co-edit documents, track project progress, and centralize all resources in one accessible platform. Integrating with task management and development tools supports AGILE and SCRUM methodologies, making it ideal for project planning and execution. The custom template system streamlines repetitive documentation, ensuring project consistency and efficiency. Its ability to function as a central knowledge hub eliminates information silos and enhances accessibility for distributed teams. With these capabilities, Notion is a versatile, scalable, and intuitive solution for teams needing a unified space for documentation, planning, and collaboration.

Asana / Jira (Project Management): Task and issue-tracking systems with robust integrations. Asana and Jira are powerful project management tools for task tracking, issue management, and team collaboration in software development and business operations. Jira, explicitly built for AGILE and SCRUM workflows, provides advanced features like sprint planning, backlog management, and bug tracking, making it ideal for development teams. On the other hand, Asana offers a more flexible and intuitive task management system suitable for various industries beyond software development. Both platforms support robust integrations with tools like GitHub, Slack, Confluence, and CI/CD pipelines, ensuring seamless workflow automation. Their customizable dashboards and reporting features provide real-time insights into project progress, helping teams stay aligned on goals and deadlines. With automation features, teams can reduce manual work by setting up rules for task assignments, notifications, and status updates. By leveraging these platforms, organizations can improve collaboration, enhance productivity, and maintain clear visibility over complex projects and development cycles.

EndNote / Mendeley (Reference Management): Comprehensive bibliographic management solutions with citation plugins. EndNote and Mendeley are comprehensive reference management solutions designed to help researchers, students, and professionals efficiently organize, store, and cite academic resources. Both platforms offer citation plugins for seamless integration with word processors like Microsoft Word, Google Docs, and LaTeX, allowing users to insert and format references automatically. EndNote is mainly known for its advanced citation customization and

extensive journal formatting options, making it a preferred choice for high-level academic publishing. Conversely, Mendeley excels in collaborative research, providing cloud storage, shared libraries, and PDF annotation features that allow teams to manage references collectively. With integrations for Zotero, BibTeX, and other research tools, both platforms support cross-platform compatibility and data synchronization. Their AI-powered recommendation systems suggest relevant papers based on user libraries, improving literature discovery and research efficiency. By leveraging EndNote or Mendeley, researchers can streamline reference management, enhance collaboration, and ensure accuracy in citations and bibliographies.

4.4. Communication & collaboration tools

Discord excels in real-time communication, offering an excellent user experience, scalability, flexibility, and cost-effectiveness. It is highly volunteer-centric due to easy onboarding but lacks integration with project management tools and robust security features.
Slack offers strong real-time communication, integration with project management tools, excellent user experience, scalability, and robust security features. It is well-suited for volunteer groups despite being less cost-effective compared to alternatives.
MS Teams (Communication & Collaboration): Built-in integrations for project management and third-party tools like Zotero. Microsoft Teams is an assertive communication and collaboration platform integrating chat, video conferencing, file sharing, and project management tools into a unified workspace. Its built-in integrations with tools like Planner, Trello, Jira, and Azure DevOps enable seamless task tracking and project coordination. Researchers and development teams can centralize documentation and streamline knowledge sharing with third-party integrations, including Zotero, for reference management. Teams support real-time document collaboration through Microsoft 365 apps (Word, Excel, PowerPoint), allowing users to edit files simultaneously while keeping track of changes. The channel-based structure enhances organization by categorizing discussions, meetings, and resources within specific projects or departments. Its automation capabilities, powered by Power Automate and AI-driven bots, help reduce repetitive tasks and improve workflow efficiency. By leveraging Microsoft Teams, organizations can enhance remote collaboration, improve project transparency, and integrate essential research and management tools into a single platform.

5. Results

Working on KATH made it possible to evaluate selected PM methodologies—Scrum, Kanban, and a combination of both, ScrumBan. The evaluation is based on a category and its performance in each situation. Each methodology's compatibility with Agile is very high, as both have allowed front-end (further - FE) and back-end (further - BE) teams to repeatedly push forward and disperse complex tasks into smaller bits, which allowed FE and BE teams to achieve long-term and short-term visions. Having both methodologies at disposal did not impose decreased compatibility with the team's agility.

From a flexibility and adaptability standpoint, it has been very noticeable that the Scrum methodology was more rigid. Adjusting the sprint in the middle of one is challenging and puts much strain on the team when attempted. On the other hand, Kanban was very flexible and adaptable to FE and BE needs and urgent issues that needed resolution. ScrumBan, as a combination, stands in the middle between Scrum and Kanban methodologies.

ScrumBan enabled transparent and decentralized coordination support in KATH. While the components were not as potent, the combination has allowed for quick analysis and problem-solving, clearing up confusion and aiding the progress of different tasks and problem-solving situations. In the KATH case, adapting Scrum and Kanban has allowed the project to move much faster, so it was marked as "Very High" and "High," respectively.

From a management standpoint, Kanban has been straightforward, as the elements are essential to the design and are easy to set up. Scrum is a complex framework, and understanding it fully requires considerable effort. To lead individual stand-ups and initialize agility, communication systems, proper schedules, and additional planning are a must, as without the aforementioned aspects, Scrum becomes a hindrance instead of a proper tool [citata/šaltinis]. When merged with Kanban, this grading turns the

combination's evaluation into "Moderate." Over time, as the team grew in size, Kanban performed well despite the more significant number of people, while Scrum, on the other hand, did not. The meetings became much longer, vastly decreasing the team's effectiveness by draining members' time.

5.1. Selection of system components

Based on performed analysis and KATH VTGs experience tables 1 through 4 presents the findings.

Table 1
Project Management Methodologies

	Criterion	Scrum	Kanban	ScrumBan (Hybrid)
1	Agile Compatibility	Very high	Very high	Very high
2	Flexibility and Adaptability	Moderate	Very high	High
3	Decentralized Coordination Support	High	High	Very high
4	Suitability for Self-Managed Teams	Very high	High	High
5	Ease of Adoption for Student Volunteers	Moderate	Very high	Moderate
6	Scalability for VTGs	Moderate	Very high	Very high

Table 2
Project Management Software Tools

	Criterion	Open Project	Git Lab PM	Taiga	Redmine	Git Projects	Kanboard	Jira	Trello	ClickUp	Notion
1	Task Management	+	+	+	+	+	+	+	+	+	+
2	Agile Workflow	+	+	+	-	+	+	+	+	+	-
3	Collaboration and Communication	+	-	-	+	-	-	-	-	-	-
4	Documentation Integration	+	+	+	+	+	-	-	-	+	-
5	Resource & Time Management	+	+	+	+	+	+	+	+	+	+
6	User Experience	+	+	+	-	+	+	+	+	+	-
7	Scalability & Flexibility	+	+	+	+	+	+	+	+	+	+
8	Integration Capabilities	+	+	+	+	+	+	+	+	+	-
9	Data Portability	-	-	+	-	+	+	-	-	+	-
10	Cost	+	+	+	+	+	+	-	+	+	+

	effectiveness									
11	Security & Privacy	+	-	+	+	+	-	-	+	+	-
12	Volunteer Centric	+	+	+	+	+	+	-	+	-	-

Table 3
Knowledge Management & Documentation Tools

	Criteria	Zotero	Logseq	Obsidian
1	Documentation Capability	-	+	+
2	Personal Knowledge Management	+	+	+
3	Collaborative Editing & Sharing	-	-	+
4	Bibliographic Management	+	-	-
5	Ease of Use	+	+	+
6	Integration with PM Tools	+	+	+
7	Data Portability (Import/Export)	+	+	+
8	Cost-effectiveness	+	+	+
9	Scalability & Flexibility	+	+	+
10	Security & Privacy	-	-	+

Table 4
Communication & Collaboration Tools

	Criteria	Discord	Slack	Microsoft Teams
1	Real-time Communication	+	+	+
2	Collaboration & Team Management	+	+	+
3	Integration with PM Tools	-	+	+
4	User Experience	+	+	-
5	Scalability & Flexibility	+	-	+
6	Cost-effectiveness	+	-	+
7	Security & Privacy	-	+	+
8	Volunteer-centric (easy onboarding)	+	+	+

5.2. Proposed system architecture

Based on evaluating available components in relation to criteria, the following system architecture has been developed to assist in project management, organization, and knowledge transfer. Its structure is represented in Figure 1.

Scrumban is a hybrid project management approach that blends elements of Scrum (such as short, fixed-length sprints and periodic standups) with Kanban (emphasizing a continuous flow of work and

visual task boards). In a student-led research setting, this approach offers both the structure needed for collective milestones and the flexibility to handle volunteer availability. Under Scrumban, volunteers can hold short, regular syncs to set goals and maintain a "To Do – Doing – Done" flow board where they pull tasks as they have capacity, imposing work-in-progress (WIP) limits to avoid overload.

Within this framework, three tools comprise the backbone of a self-organized student group's day-to-day operations. GitProject is used to manage tasks and track progress. It provides a visual Kanban or Scrumban board where issues are created, prioritized, and associated with specific tasks and milestones. Each member can "pull" a task into "Doing" and then move it to "Done" once complete. If the tasks involve code or data pipelines, they may involve pull requests or merge reviews, ensuring some quality control without overwhelming formality.

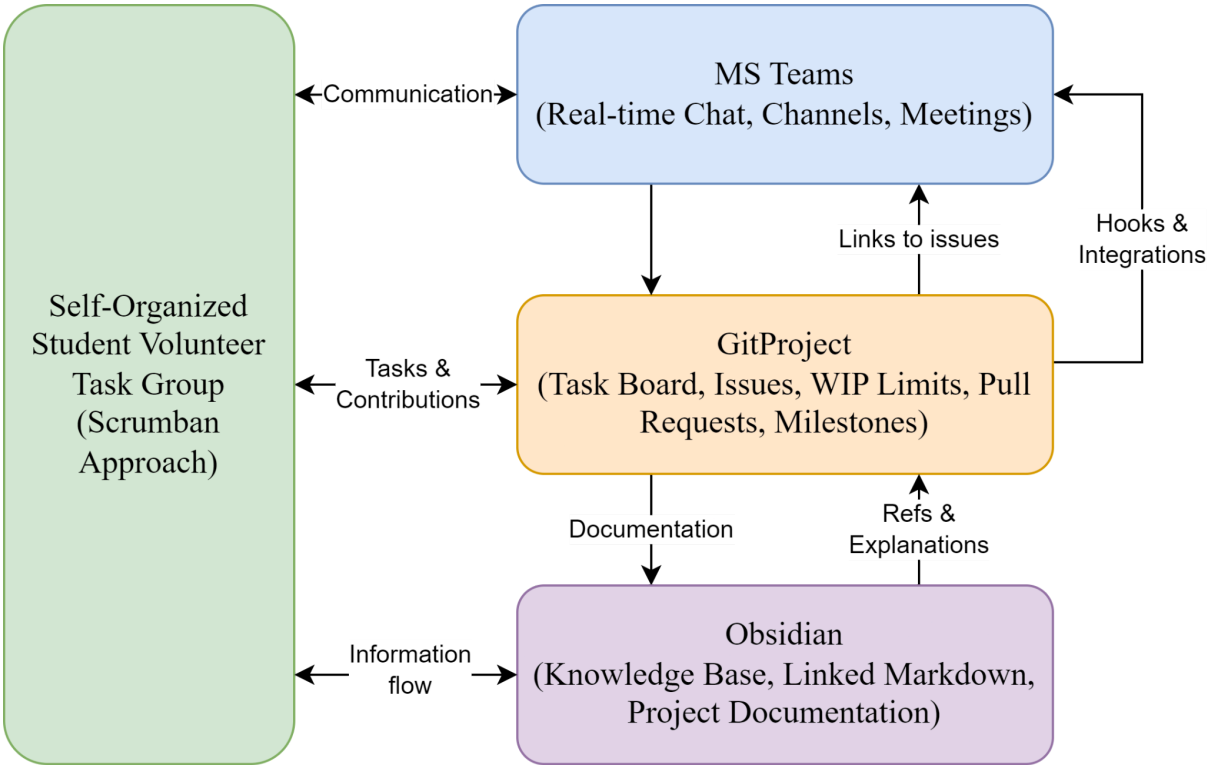

Figure 1: The structure of the proposed system

In parallel, Obsidian serves as the knowledge management and documentation hub. It stores meeting notes, how-to guides, and the rationale behind certain decisions. Unlike scattered documents, Obsidian's use of linked Markdown files allows volunteers to see connections among different pieces of information.

Microsoft Teams was selected as a primary communication channel. It offers real-time chat, channels dedicated to particular focus areas, and integration hooks that alert members whenever tasks are created or updated in GitProject. Once the issue is logged, references or instructions for tackling the task can be recorded in Obsidian. A volunteer then "pulls" that newly created task into their workload. As they work, they keep track of partial results, store or update relevant documentation in Obsidian, and post occasional status messages in Teams. Once finished, they move the task to "Done" or create a pull request for code changes.

At a broader level, the group runs regular Scrumban standups (either daily or less frequently) to discuss outstanding items, next steps, and roadblocks. When tasks accumulate in the "Doing" column, the team can consider adding more volunteers or splitting tasks further. If tasks in "Done" need validation, they may schedule an extra review. This cyclical process continues, punctuated by retrospective meetings, where students reflect on whether tasks were properly scoped if knowledge was documented transparently, and how the volunteer process can be refined for future cycles.

A simplified depiction of their interactions can be imagined as three core components—GitProject for task management, Obsidian for documentation, and MS Teams for communication—continuously

exchanging information. Teams discussions spark new issues in GitProject; GitProject notifies a Teams channel when issues or tasks change status; Obsidian content, in turn, links to or from GitProject tasks to maintain clarity about where knowledge resides. In this manner, volunteers can self-organize effectively: each participant sees precisely which tasks are pending, which are in progress, and which have been completed, while all relevant knowledge accumulates in Obsidian for future reference. This flexible yet structured approach is ideally suited to volunteer-based teams, allowing them to adapt quickly to changes and maintain continuity as new members join or others step away.

6. Conclusions and future work

The findings indicate that a Scrumban-based system, underpinned by lightweight but robust project management, knowledge management, and communication tools, can sustain progress in student-driven volunteer task groups despite fluctuating membership and skill diversity. Integrating GitProject, Obsidian, and Microsoft Teams ensures that essential documentation and task flow remain transparent, enabling contributors to onboard and exit with minimal disruption. Volunteer teams benefit from a hybrid approach that fosters autonomy, promotes collaborative learning, and preserves institutional knowledge, ultimately bolstering productivity and engagement. By aligning an iterative, pull-based workflow with transparent documentation practices and real-time communication, self-organized teams can quickly adapt to new challenges while advancing long-term objectives.

Future work will focus on developing additional integration frameworks that automate recurring data workflows and streamline inter-tool communication. This may include exploring robotic process automation (RPA) technologies such as Robot Framework or TagUI to orchestrate tasks between GitProject, Obsidian, and other potential platforms. Moreover, improved role-based access control (RBAC) and security standards will be investigated, mainly using unified authentication systems like Keycloak for larger-scale academic ecosystems. By adopting protocols such as OpenID Connect and SAML, volunteer research groups can reinforce data security, facilitate single sign-on (SSO), and maintain a shared identity framework that reduces administrative overhead while enhancing the volunteer experience.

7. References

[1] Špundak, M. (2014). Mixed agile/traditional project management methodology–reality or illusion?. Procedia-Social and behavioral sciences, 119, 939-948.

[2] Lei, H., Ganjeizadeh, F., Jayachandran, P. K., & Ozcan, P. (2017). A statistical analysis of the effects of Scrum and Kanban on software development projects. Robotics and Computer-Integrated Manufacturing, 43, 59-67.

[3] Joslin, R., & Müller, R. (2015). Relationships between a project management methodology and project success in different project governance contexts. International journal of project management, 33(6), 1377-1392.

[4] Serrador, P., & Pinto, J. K. (2015). Does Agile work?—A quantitative analysis of agile project success. International journal of project management, 33(5), 1040-1051.

[5] Stanimirovic, P., Borozan, T., Radojicic, M., & Tomic, A. D. (2023, July). Project Management Software Tools–One Step Closer to UN Sustainable Development Goals. In 2023 3rd International Conference on Electrical, Computer, Communications and Mechatronics Engineering (ICECCME) (pp. 1-5). IEEE.

[6] Stanimirović, P., Petrović, K., Borozan, T., & Radojičić, M. (2023). Project Management Software-Catalyst or Constraint?. https://spin. fon. bg. ac. rs/wp-content/uploads/2024/01/SPIN-2023-zbornik-radova. pdf.

[7] Brad, M. C., Birloi, F. C., Bratulescu, A., & Blaga, I. B. (2016). A comparative study of agile project management software tools. Academy of Economic Studies. Economy Informatics, 16(1), 27-38.

[8] Chaabouni, M., Jarrar, Q., Medini, K., & Bissardon, F. (2024). A framework for project management software tool selection. Procedia Computer Science, 239, 291-298.

[9] Tzabbar, D., & Baburaj, Y. (2019). Optimizing the effectiveness of geographically dispersed Research & Development teams. Organizational Dynamics, 48(4), 100684.

[10] Farooq, R. (2022). A review of knowledge management research in the past three decades: a bibliometric analysis. VINE Journal of Information and Knowledge Management Systems, 54(2), 339-378.
[11] Zawaideh, F., Al-Zoubi, M. I., Abualoush, S. H., Kanaan, R. K., & Masa'deh, R. E. (2018). The impact of knowledge documentation process as an intermediary variable among knowledge acquisition process, organizational culture and human capital. Modern Applied Science, 12(11), 151-168.
[12] Nakayama, M., Hustad, E., & Sutcliffe, N. (2021). Agility and system documentation in large-scale enterprise system projects: a knowledge management perspective. Procedia Computer Science, 181, 386-393.
[13] Elkelä, A. (2024). Knowledge flow effectiveness in technical documentation.
[14] Theunissen, T., van Heesch, U., & Avgeriou, P. (2022). A mapping study on documentation in Continuous Software Development. Information and software technology, 142, 106733.
[15] Bissaliyev, M. S. (2017). The effectiveness of collaboration tools on virtual project management. International Journal of Applied Engineering Research, 12(21), 10747-10755.
[16] Miljanic, M., & Zaric, N. (2020, February). Review of collaborative software applications and integration with standard collaboration tools. In 2020 24th International Conference on Information Technology (IT) (pp. 1-4). IEEE.
[17] Maican, C., Cazan, A. M., Lixandroiu, R., Dovleac, L., & Maican, M. A. (2021). Exploring the factors influencing the use of communication and collaboration applications. Journal of Organizational and End User Computing (JOEUC), 33(4), 94-124.
[18] Boling, E. C., Holan, E., Horbatt, B., Hough, M., Jean-Louis, J., Khurana, C., ... & Spiezio, C. (2014). Using online tools for communication and collaboration: Understanding educators' experiences in an online course. The Internet and Higher Education, 23, 48-55.
[19] Adamson, D., Dyke, G., Jang, H., & Rosé, C. P. (2014). Towards an agile approach to adapting dynamic collaboration support to student needs. International Journal of Artificial Intelligence in Education, 24, 92-124.
[20] Sánchez-Silva, M., & Calderón-Guevara, W. (2022). Flexibility and adaptability within the context of decision-making in infrastructure management. Structure and Infrastructure Engineering, 18(7), 950-966.
[21] Curlee, W. (2008). Modern virtual project management: The effects of a centralized and decentralized project management office. Project management journal, 39(1_suppl), S83-S96.
[22] Krause, S., Stojanová, H., Behrens, V. J., & Hangl, J. E. (2024). Self-managed teams-a literature review of a project organisation. International Journal of Project Organisation and Management, 16(3), 292-308.
[23] Halse, S. E., & Hansen, G. F. (2024). Innovation in Student Organisations-How Innovation Occurs in Non-Profit Organisations Managed and Run by Volunteer Students (Master's thesis, NTNU).
[24] Khalid, H., Ahmed, M., Sameer, A., & Arif, F. (2015). Systematic literature review of agile scalability for large scale projects. Int. J. Adv. Comput. Sci. Appl.(IJACSA), 6(9), 63-75.
[25] Mishra, A., & Mishra, D. (2013). Software project management tools: a brief comparative view. ACM SIGSOFT Software Engineering Notes, 38(3), 1-4.
[26] Binder, J. (2016). Global project management: communication, collaboration and management across borders. Routledge.
[27] Wasserman, A. I. (2005, June). Tool integration in software engineering environments. In Software Engineering Environments: International Workshop on Environments Chinon, France, September 18–20, 1989 Proceedings (pp. 137-149). Berlin, Heidelberg: Springer Berlin Heidelberg.
[28] Hazar, H. H. (2014). Time management tools and techniques for project management. Socio-economic research bulletin, (4), 57-62.
[29] Dimitrijević, S., Jovanović, J., & Devedžić, V. (2015). A comparative study of software tools for user story management. Information and Software Technology, 57, 352-368.
[30] Nakayama, M., Hustad, E., & Sutcliffe, N. (2021). Agility and system documentation in large-scale enterprise system projects: a knowledge management perspective. Procedia Computer Science, 181, 386-393.
[31] Gasik, S. (2011). A model of project knowledge management. Project Management Journal, 42(3), 23-44.
[32] Borgho, U. M., & Teege, G. (1993). Application of collaborative editing to software-engineering projects. ACM SIGSOFT Software engineering notes, 18(3).
[33] Moutinho, J. D. A., & Silva, L. F. D. (2022). Knowledge management in project management: mapping bibliographic convergence. Knowledge Management Research & Practice, 20(2), 266-278.
[34] Abualoush, S., Bataineh, K., & Alrowwad, A. A. (2018). The role of knowledge management process and intellectual capital as intermediary variables between knowledge management infrastructure and

organization performance. Interdisciplinary Journal of Information, Knowledge, and Management, 13, 279-309.
[35] Schniederjans, D. G., Curado, C., & Khalajhedayati, M. (2020). Supply chain digitisation trends: An integration of knowledge management. International Journal of Production Economics, 220, 107439.
[36] Graef, I., Husovec, M., & Purtova, N. (2018). Data portability and data control: lessons for an emerging concept in EU law. German Law Journal, 19(6), 1359-1398.
[37] Amo-Filva, D., Escudero, D. F., Sanchez-Sepulveda, M. V., García-Holgado, A., García-Holgado, L., García-Peñalvo, F. J., ... & Moreira, F. (2023, June). Security and privacy in academic data management at schools: SPADATAS project. In International Conference on Human-Computer Interaction (pp. 3-16). Cham: Springer Nature Switzerland.