**Find recent (preferably 2018–2025) research papers related to web-based task management systems, team collaboration tools, productivity dashboards, and project tracking applications. The focus is on lightweight, user-friendly platforms built for small teams, students, or hobbyist groups — emphasizing collaboration, task tracking, and visualization of progress (dashboards, Kanban, etc.). Include studies or papers covering: Web-based task management and productivity tools Collaboration and coordination in small teams or student projects Design and usability of dashboards or task visualization interfaces AI-driven or smart task prioritization and workload distribution Open-source or low-resource management systems for teams I am researching for a product concept named "TaskWise – Smart Team Task & Collaboration Manager", which aims to simplify team coordination, task creation, assignment, and progress tracking in an academic or small-team context. Please provide relevant academic papers, conference proceedings, or technical reports that align with these objectives, along with summaries, key findings, and methods used in each paper.**

Ten studies detail lightweight task management systems for academic and small-team environments, with recent research (2018-2020) highlighting AI-enhanced features, open-source solutions, and user-friendly interfaces using Kanban boards and dashboards for task tracking and collaboration.


## Abstract

Ten studies describe lightweight, user-friendly systems for task management and team collaboration in academic and small-team settings. Pisoni and Hoogeboom (2019) detail a Trello-based interface that enables asynchronous coordination among geographically distributed students. Nagel and Schwade (2020) present UniConnect, which integrates task creation, assignment, tracking, and file sharing for academic and industry partners. Chasanidou et al. (2016) introduce UpWave with features such as role-based access and gamification, while Toxtli et al. (2018) report TaskBot's use of artificial intelligence for intent detection within a chat-integrated, hybrid system. Fraser et al. (2017) implement automated, dependency-based subtask assignment, and Luqman and Griss (2010) employ a mobile, open-source solution that uses context-aware, agent-driven allocation. Nakazawa and Tanaka (2015, 2016) propose Kanban-style tools that display workload visualization and support real-time synchronization.

Across these studies, eight systems rely on primarily user-driven task allocation, whereas two incorporate automated assignment methods enhanced by smart features. Four papers describe context-aware or AI-driven capabilities, and interfaces typically offer Kanban boards, dashboards, calendars, or mobile-first designs to support both synchronous and asynchronous collaboration. The papers collectively detail design strategies that simplify task creation, assignment, tracking, and progress visualization for students, small teams, and hobbyist groups.

## Paper search

We performed a semantic search using the query "Find recent (preferably 2018–2025) research papers related to web-based task management systems, team collaboration tools, productivity dashboards, and project tracking applications.

The focus is on lightweight, user-friendly platforms built for small teams, students, or hobbyist groups — emphasizing collaboration, task tracking, and visualization of progress (dashboards, Kanban, etc.).

Include studies or papers covering:

Web-based task management and productivity tools

Collaboration and coordination in small teams or student projects

Design and usability of dashboards or task visualization interfaces

AI-driven or smart task prioritization and workload distribution

Open-source or low-resource management systems for teams

I am researching for a product concept named "TaskWise – Smart Team Task & Collaboration Manager", which aims to simplify team coordination, task creation, assignment, and progress tracking in an academic or small-team context.

Please provide relevant academic papers, conference proceedings, or technical reports that align with these objectives, along with summaries, key findings, and methods used in each paper." across over 126 million academic papers from the Elicit search engine, which includes all of Semantic Scholar and OpenAlex.

We retrieved the 50 papers most relevant to the query.

## Screening

We screened in sources that met these criteria:

- **Web-based Platform**: Does the study focus on internet-accessible, web-based, or cloud-based task management or collaboration systems?
- **Task Management Functionality**: Does the study examine systems with core task management features such as task creation, assignment, tracking, or progress monitoring capabilities?
- **Small Team Context**: Does the research involve small teams, groups, or communities (2-20 members), such as student groups, academic project teams, or hobbyist communities?
- **Collaboration Features**: Does the research investigate tools that facilitate team communication, coordination, or collaborative work processes (rather than single-user productivity tools)?
- **Empirical Research Design**: Does the study use empirical research methodologies (experimental, quasi-experimental, observational, case study, survey) or is it a systematic review/meta-analysis that provides evidence-based findings (rather than being purely theoretical or conceptual)?
- **Small Team Applicability**: Is the study applicable to small teams rather than focusing exclusively on large-scale enterprise systems (>50 users) without small team relevance?
- **Digital System Focus**: Does the study focus on digital, networked systems rather than paper-based, desktop-only, or non-networked task management approaches?

We considered all screening questions together and made a holistic judgement about whether to screen in each paper.

## Data extraction

We asked a large language model to extract each data column below from each paper. We gave the model the extraction instructions shown below for each column.

- **System Type**:

  Extract details about the task management system including:

- Platform type (web-based, mobile, desktop, hybrid)
- System architecture (standalone, integrated with other tools, bot-based, multi-agent)
- Technical approach (prototype, production system, conceptual framework)
- Open-source status and availability
- Integration capabilities (GitHub, communication tools, etc.)

- **Target Context**:

  Describe the intended users and use context including:

- User type (students, small teams, professionals, specific domains)
- Team size and structure (distributed, collocated, ad-hoc, established)
- Setting (academic, business, disaster response, software development)
- Specific collaboration challenges addressed
- Resource constraints mentioned (low-resource, lightweight requirements)

- **Core Features**:

  Extract all task management and collaboration features including:

- Task creation, assignment, and tracking capabilities
- Progress monitoring and status updates
- Communication and coordination tools
- File sharing and documentation features
- Role-based access and permissions
- Real-time synchronization and updates
- Notification and alert systems

- **Interface Design**:

  Document the user interface and visualization approaches including:

- Task visualization methods (Kanban boards, dashboards, lists, timelines)
- Progress tracking displays and metrics shown
- Mobile vs desktop interface considerations
- Information architecture and layout decisions
- Visual design principles applied
- Accessibility and usability design features

- **Smart Features**:

  Extract any AI-driven or intelligent features including:

- Automated task assignment or distribution algorithms
- Smart prioritization or workload balancing

- Context-aware recommendations or adaptations
- Predictive analytics or progress forecasting
- Machine learning applications
- Decision support capabilities
- Criteria and factors used in automated decisions

- **Study Method**:

  Describe the research methodology including:

- Study design (experiment, case study, observational study, prototype evaluation)
- Duration of study and data collection period
- Number and characteristics of participants/teams studied
- Data collection methods (surveys, interviews, usage analytics, observation)
- Analysis approach used
- Comparison conditions or baseline systems

- **Key Findings**:

  Extract main results and insights including:

- Primary outcomes and effectiveness measures
- Quantitative results (performance improvements, usage statistics, efficiency gains)
- Qualitative insights about user experience and collaboration
- Identified barriers and challenges
- Success factors and design recommendations
- Comparison results vs existing tools or baseline conditions

- **Team Coordination**:

  Document specific findings about team collaboration and coordination including:

- How the system affected team communication patterns
- Impact on task distribution and workload management
- Changes in coordination overhead or startup time
- Effects on team awareness and visibility
- Collaboration patterns observed (synchronous vs asynchronous)
- Conflict resolution and decision-making support

# Results

## Characteristics of Included Studies

| Study | Study Focus | System Type | Target Users | Key Features | Full text retrieved |
|---|---|---|---|---|---|
| Pisoni and Hoogeboom, 2019 | Analysis of virtual student team dynamics via Trello | Web-based, not integrated with other platforms (Trello) | Students, small geographically distributed academic teams | Task creation, assignment, tracking, progress monitoring, communication, file sharing, real-time synchronization, notifications | No |
| Lin et al., 2014 | Task allocation in novice Agile teams using Agile Project Management tool | Web-based | Undergraduate software engineering students, small physically co-located teams | We didn't find mention in the abstract | No |
| Nagel and Schwade, 2020 | Task management feature usage in academic collaboration platform (UniConnect) | Web-based, integrated with other platforms | Students, academics, industry partners; geographically distributed or physically co-located | Task/project creation, assignment, tracking, progress monitoring, communication, file sharing, role-based access, real-time synchronization, notifications | Yes |
| Chasanidou et al., 2016 | Enabling team collaboration with UpWave | Web-based, integrated with other platforms | Professionals in various industries, geographically distributed teams | Task creation, assignment, tracking, progress monitoring, communication, file sharing, role-based access, real-time synchronization, notifications, gamification | Yes |

| Study | Study Focus | System Type | Target Users | Key Features | Full text retrieved |
|---|---|---|---|---|---|
| Toxtli et al., 2018 | Chatbot-mediated task management (TaskBot) | Hybrid (chatbot in communication tools), integrated with other platforms | Information workers, small physically co-located business teams | Task creation, assignment, tracking, progress monitoring, communication, notifications, artificial intelligence-driven intent detection | Yes |
| Jones et al., 2016 | Journaling interfaces for collaborative tasks/goals | We didn't find mention in the abstract, integrated with other platforms | Students, intelligence analysts | Task/goal capture, tracking, progress monitoring, file/document association, aggregate visualizations | No |
| Fraser et al., 2017 | Automated task distribution for physically co-located assembly workers (WeBuild) | Mobile | Professionals in construc-tion/manufacturing physically co-located teams | Automated task assignment, personalized instructions, dashboard visualization | No |
| Nakazawa and Tanaka, 2015 | Prototype Kanban tool for geographically distributed agile teams | We didn't find mention in the abstract, GitHub integration | Software development professionals, geographically distributed teams | Task display, assignment, workload visualization, GitHub integration, real-time synchronization | No |

| Study | Study Focus | System Type | Target Users | Key Features | Full text retrieved |
|---|---|---|---|---|---|
| Luqman and Griss, 2010 | Context-aware mobile task management for disaster response (Overseer) | Mobile, open-source, multiple autonomous agents | First responders, volunteers, ad-hoc geographically distributed teams | Automated/context-aware task assignment, progress monitoring, communication, role-based access, real-time synchronization, notifications | Yes |
| Nakazawa and Tanaka, 2016 | Web-based Kanban tool for agile teams | Web-based | Software development professionals, established teams | Work In Progress visual-ization/limiting, improved communication, conferencing, consultation | No |

System Type:

- Web-based systems: 6 studies
- Mobile systems: 2 studies
- Hybrid (chatbot-integrated): 1 study
- System type not mentioned in the abstract: 1 study
- Integrated with other platforms: 4 studies
- Not integrated with other platforms: 1 study
- Open-source and multiple autonomous agent system: 1 study

Target Users:

- Students: 4 studies
- Professionals: 4 studies
- Academics, industry partners, information workers, intelligence analysts, first responders, volunteers: 1 study each
- Geographically distributed teams: 5 studies
- Physically co-located teams: 3 studies
- Both distributed and co-located: 1 study
- Established and ad-hoc teams: 1 study each

Key Features:

- Task creation/capture/display: 6 studies
- Task assignment (including automated/context-aware): 7 studies
- Task/project tracking: 5 studies
- Progress monitoring: 6 studies

- Communication features: 6 studies
- File sharing or document association: 4 studies
- Real-time synchronization: 5 studies
- Notifications: 5 studies
- Role-based access: 3 studies
- Visualization features (workload, dashboard, Work In Progress): 4 studies
- Artificial intelligence/automation features (including intent detection and automated assignment): 3 studies
- Gamification, conferencing, consultation, and personalization: 1 study each
- Integration with external tools (e.g., GitHub, communication tools): 2 studies
- We didn't find mention of key feature information for 1 study

---

## Thematic Analysis

**Task Allocation and Distribution Mechanisms**

| Study | Task Allocation/Distribution Approach | Smart/Automated Features |
|---|---|---|
| Pisoni and Hoogeboom, 2019 | User-driven task assignment via Trello | We didn't find mention in the abstract |
| Lin et al., 2014 | User-driven, with analysis of allocation decisions; focus on competence/task difficulty | Context-aware decision support (proposed) |
| Nagel and Schwade, 2020 | User-driven assignment in UniConnect; inconsistent use | We didn't find mention |
| Chasanidou et al., 2016 | User-driven assignment in UpWave; color-coded priorities | We didn't find mention |
| Toxtli et al., 2018 | TaskBot enables assignment via chat; reminders | Artificial intelligence-driven intent detection, message parsing |
| Jones et al., 2016 | User-driven capture of tasks/goals; journaling | Automated data collection/visualization |
| Fraser et al., 2017 | Automated subtask assignment via custom algorithm | Dynamic assignment based on dependencies/skills |
| Nakazawa and Tanaka, 2015 | User-driven assignment; workload visualization | We didn't find mention |
| Luqman and Griss, 2010 | Automated, context-aware assignment via multiple autonomous agents | Context-aware, predictive analytics, decision support |
| Nakazawa and Tanaka, 2016 | User-driven assignment; Work In Progress limiting | We didn't find mention |

Key findings:

- User-driven (manual) task allocation was reported in 8 studies.
- Automated allocation was reported in 2 studies (Fraser et al., 2017; Luqman and Griss, 2010).

- Among user-driven approaches, some included additional support such as visualization, color-coded priorities, or chat-based assignment, but the allocation itself remained user-driven.
- Four studies included automated or smart features, such as artificial intelligence-driven intent detection, automated data collection, dynamic assignment algorithms, or context-aware decision support.
- In five studies, we didn't find mention of smart or automated features.
- In one study, smart features were proposed but not implemented.
- All studies with automated allocation also included smart or automated features.
- Among studies with user-driven allocation, two (Toxtli et al., 2018; Jones et al., 2016) included some form of automation or smart support, but the allocation process itself was still user-driven.
- We didn't find mention of any studies using a fully mixed or hybrid allocation approach.

---

**Collaboration Coordination Strategies**

| Study | Coordination Features | Observed Collaboration Patterns | Barriers/Challenges |
|---|---|---|---|
| Pisoni and Hoogeboom, 2019 | Trello boards for asynchronous coordination | Asynchronous, geographically distributed | We didn't find mention in the abstract |
| Lin et al., 2014 | Agile Project Management tool for Agile teams | We didn't find mention in the abstract | We didn't find mention in the abstract |
| Nagel and Schwade, 2020 | Task management features in UniConnect; inconsistent use | Irregular, inconsistent | Inconsistent use, lack of updates |
| Chasanidou et al., 2016 | UpWave supports time tracking, dependencies, gamification | Asynchronous, geographically distributed | Adoption, motivation, diverse teams |
| Toxtli et al., 2018 | TaskBot in chat for in-situ coordination | Synchronous/asynchronous | Handling multiple tasks, intent ambiguity |
| Jones et al., 2016 | Journaling for situational awareness | We didn't find mention in the abstract | We didn't find mention in the abstract |
| Fraser et al., 2017 | Personalized instructions, dashboard | We didn't find mention in the abstract | We didn't find mention in the abstract |
| Nakazawa and Tanaka, 2015 | Kanban as communication hub | Synchronous (real-time synchronization) | We didn't find mention in the abstract |
| Luqman and Griss, 2010 | Overseer automates coordination via context | Asynchronous, context-aware | Usability, adoption, security |
| Nakazawa and Tanaka, 2016 | Kanban for communication, Work In Progress limiting | We didn't find mention in the abstract | We didn't find mention in the abstract |

Key findings:

- Kanban-style coordination was reported in three studies (Nakazawa and Tanaka, 2015; Nakazawa and Tanaka, 2016; Pisoni and Hoogeboom, 2019 via Trello).

- Other coordination features included Trello boards, Agile Project Management tools, task management features, UpWave, TaskBot/chatbot, journaling, personalized instructions, dashboards, and context-aware automation.
- Asynchronous collaboration was observed in four studies.
- Synchronous collaboration was observed in two studies.
- Geographically distributed collaboration was observed in two studies.
- Context-aware collaboration was observed in one study.
- Irregular or inconsistent collaboration was observed in one study.
- Real-time synchronization was observed in one study.
- We didn't find mention of collaboration patterns in four studies.
- Barriers and challenges included inconsistent use, lack of updates, adoption, motivation, diverse teams, handling multiple tasks, intent ambiguity, usability, and security.
- We didn't find mention of barriers or challenges in six studies.

---

**User Interface Design and Visualization Approaches**

| Study | Visualization Methods | User Interface/User Experience Design Features | Accessibility/Usability |
|---|---|---|---|
| Pisoni and Hoogeboom, 2019 | Kanban boards (Trello) | We didn't find mention in the abstract | We didn't find mention in the abstract |
| Lin et al., 2014 | We didn't find mention in the abstract | We didn't find mention in the abstract | We didn't find mention in the abstract |
| Nagel and Schwade, 2020 | We didn't find mention in the abstract | We didn't find mention in the abstract | We didn't find mention in the abstract |
| Chasanidou et al., 2016 | Kanban boards, calendar, color coding | Simple, intuitive, minimal clicks | Emphasis on usability, security |
| Toxtli et al., 2018 | Conversational (chatbot) | Text-based, integrated in chat | We didn't find mention in the abstract |
| Jones et al., 2016 | Aggregate visualizations | We didn't find mention in the abstract | We didn't find mention in the abstract |
| Fraser et al., 2017 | Dashboard, mobile instructions | Mobile-first | We didn't find mention in the abstract |
| Nakazawa and Tanaka, 2015 | Kanban-like, horizontal rows | Visual workload overview | We didn't find mention in the abstract |
| Luqman and Griss, 2010 | Dependency graph, tactile feedback | Mobile, glove-usable, visual/tactile | Designed for challenging environments |
| Nakazawa and Tanaka, 2016 | We didn't find mention in the abstract | We didn't find mention in the abstract | We didn't find mention in the abstract |

Key findings:

- Kanban-style visualizations (including Kanban boards, Kanban-like, and Trello boards): 4 studies
- Calendar and color coding: 1 study each

- Conversational/chatbot interface: 1 study
- Aggregate visualizations: 1 study
- Dashboard: 1 study
- Mobile instructions: 1 study
- Horizontal rows: 1 study
- Dependency graph: 1 study
- Tactile feedback: 1 study
- We didn't find mention of the visualization method in three studies
- User interface/user experience design features were not mentioned in five studies
- Simple, intuitive, and minimal-click design: 1 study
- Text-based and integrated in chat: 1 study
- Mobile-first design: 1 study
- Visual workload overview: 1 study
- Mobile, glove-usable, and visual/tactile features: 1 study
- Accessibility or usability was not mentioned in eight studies
- Usability and security were emphasized in one study
- Design for challenging environments was specified in one study

---

**Context-Awareness and Adaptive Features**

| Study | Context-Aware/Adaptive Features | Artificial Intelligence/Smart Capabilities |
|---|---|---|
| Pisoni and Hoogeboom, 2019 | We didn't find mention in the abstract | We didn't find mention in the abstract |
| Lin et al., 2014 | Proposed situation-aware decision support | Context-aware recommendations (proposed) |
| Nagel and Schwade, 2020 | We didn't find mention | We didn't find mention |
| Chasanidou et al., 2016 | We didn't find mention | We didn't find mention |
| Toxtli et al., 2018 | We didn't find mention | Artificial intelligence-driven intent detection, machine learning for message parsing |
| Jones et al., 2016 | Automated data collection/visualization | Passive instrumentation |
| Fraser et al., 2017 | Dynamic assignment based on skills/dependencies | Custom algorithm for task distribution |
| Nakazawa and Tanaka, 2015 | We didn't find mention | We didn't find mention |
| Luqman and Griss, 2010 | Context-aware task assignment, predictive analytics | Multiple autonomous agents, context-aware, machine learning planned |
| Nakazawa and Tanaka, 2016 | We didn't find mention | We didn't find mention |

Key findings:

- Context-aware or adaptive features were described in four studies (Lin et al., 2014; Jones et al., 2016; Fraser et al., 2017; Luqman and Griss, 2010).
- These features included situation-aware decision support, automated data collection/visualization, dynamic assignment based on skills or dependencies, and context-aware task assignment or predictive analytics.
- Artificial intelligence, machine learning, or other smart capabilities were described in five studies (Lin et al., 2014; Toxtli et al., 2018; Jones et al., 2016; Fraser et al., 2017; Luqman and Griss, 2010).
- These included context-aware recommendations, artificial intelligence-driven intent detection, machine learning for message parsing, passive instrumentation, custom algorithms for task distribution, and multiple autonomous agents or context-aware machine learning (planned).
- Three studies (Lin et al., 2014; Jones et al., 2016; Fraser et al., 2017) described both context-aware/adaptive features and artificial intelligence/smart capabilities. One additional study (Luqman and Griss, 2010) described both, but noted that artificial intelligence/machine learning was planned rather than implemented.
- We didn't find mention of either context-aware/adaptive features or artificial intelligence/smart capabilities in five studies.

---

## References


C. Ailie Fraser, Tovi Grossman, and G. Fitzmaurice. "WeBuild: Automatically Distributing Assembly Tasks Among Collocated Workers to Improve Coordination." *International Conference on Human Factors in Computing Systems*, 2017.

Carlos Toxtli, A. Monroy-Hernández, and Justin Cranshaw. "Understanding Chatbot-Mediated Task Management." *International Conference on Human Factors in Computing Systems*, 2018.

Dimitra Chasanidou, B. Elvesæter, and Arne-Jørgen Berre. "Enabling Team Collaboration with Task Management Tools." *International Symposium on Open Collaboration*, 2016.

Faisal B. Luqman, and Martin L. Griss. "Overseer: A Mobile Context-Aware Collaboration and Task Management System for Disaster Response." *2010 Eighth International Conference on Creating, Connecting and Collaborating Through Computing*, 2010.

G. Pisoni, and M. Hoogeboom. "Investigating Effective Dynamics of Virtual Student Teams Through Analysis of Trello Boards." *International Conference on Emerging eLearning Technologies and Applications*, 2019.

Jun Lin, Han Yu, Zhiqi Shen, and C. Miao. "Studying Task Allocation Decisions of Novice Agile Teams with Data from Agile Project Management Tools." *International Conference on Automated Software Engineering*, 2014.

Paul Jones, Sidharth Thakur, Sean Streck, and Christopher Kampe. "Journaling Interfaces to Support Knowledge Workers in Their Collaborative Tasks and Goals." *International Conference on Collaboration Technologies and Systems*, 2016.

S. Nakazawa, and Tetsuo Tanaka. "Development and Application of Kanban Tool Visualizing the Work in Progress." *IIAI International Conference on Advanced Applied Informatics*, 2016.

———. "Prototype of Kanban Tool and Preliminary Evaluation of Visualizing Method for Task Assignment." *2015 International Conference on Computer Application Technologies*, 2015.

Sabine Nagel, and Florian Schwade. "Analysis of Task Management in Virtual Academic Teams." *Hawaii International Conference on System Sciences*, 2020.