IJRAR.ORG E-ISSN: 2348-1269, P-ISSN: 2349-5138

INTERNATIONAL JOURNAL OF RESEARCH AND ANALYTICAL REVIEWS (IJRAR) | IJRAR.ORG

An International Open Access, Peer-reviewed, Refereed Journal

DESIGN AND DEVELOPMENT OF AI AGENTS FOR ACADEMIC PROJECT PLANNING AND MANAGEMENT

Radhika Amashi, Pooja Shekharagouda Doddagoudar, Ambika Thabbasa Chavan,
Punarvasu Shetake, Smital Sanjay Kaginkar
Department of Computer Science and Engineering
S. D. M. College of Engineering & Technology, Dharwad, Karnataka, India

---

Abstract: This paper presents an AI-assisted academic project planning and management system that addresses challenges such as inefficient planning, poor task allocation, and lack of structured monitoring in academic projects. The system integrates Large Language Models using Ollama for local inference, MongoDB for data storage, and Streamlit for interactive dashboards. Intelligent AI agents automate literature review, objective formulation, Scrum-based task planning, progress tracking, and IEEE-compliant report generation, thereby improving efficiency, transparency, collaboration, and academic quality.

Index Terms - **Artificial Intelligence, AI Agents, Academic Project Management, Large Language Models, Scrum, Ollama, MongoDB**

I. INTRODUCTION

Academic projects are a fundamental component of higher education, enabling students to apply theoretical knowledge to practical problems while developing essential skills such as problem-solving, teamwork, innovation, and decision-making. However, many students face difficulties in managing academic projects due to challenges in research collection, project planning, task allocation, progress monitoring, and resource management. The reliance on fragmented tools such as spreadsheets, messaging applications, or manual coordination often leads to inefficiencies, communication gaps, uneven workload distribution, and missed deadlines. With the increasing complexity of engineering projects and the growing emphasis on research-oriented learning, there is a clear need for structured and intelligent solutions to support effective academic project management.

To address these challenges, this project proposes TeamCollab, an AI-assisted academic project planning and management platform designed to support students throughout the project lifecycle. The system is implemented as a role-based platform for Project Leads, Mentors, and Members, offering customized dashboards for each role. TeamCollab integrates intelligent AI agents to assist in literature analysis, objective formulation, methodology recommendation, Scrum-based task planning, timeline creation, progress tracking, and performance evaluation. By automating repetitive academic tasks and providing a centralized platform for collaboration, monitoring, and structured reporting, the system reduces management overhead, enhances productivity, and bridges the gap between academic project requirements and professional project management practices.

➢ OBJECTIVES

1. To collect and summarize relevant academic research information.
2. To plan projects with structured timelines, milestones, and suitable methodologies.
3. To break down goals into tasks, allocate them to team members, and track progress with alerts.
4. To facilitate collaboration and manage resources through communication hubs and repositories.
5. To generate structured documentation and provide AI-driven evaluation and feedback.

II. LITERATURE REVIEW

Several studies and tools have explored the use of AI in project management and research assistance. Wrike (2025) introduced AI-powered workflow automation and drafting tools to improve enterprise project management, demonstrating the potential of AI for task coordination and scheduling. SciSummary (2023) focused on AI-based summarization of scientific literature, significantly reducing the effort required for manual literature reviews. Iris.ai (2025) enhanced research discovery by enabling intelligent navigation and filtering of academic publications. Smith et al. (2025) proposed AI-based task modeling using large multimodal models to automate task planning, while Zhang et al. (2024) presented AI-optimized task allocation techniques based on skill matching and resource utilization. Although these approaches highlight the effectiveness of AI in automation and planning, most existing solutions are designed for enterprise or industrial environments and lack features specific to academic project workflows, such as mentor supervision, learning-oriented evaluation, and structured academic documentation. This research addresses these limitations by proposing a dedicated AI-assisted system tailored for academic projects.

III. SYSTEM ARCHITECTURE AND WORKFLOW

System Architecture

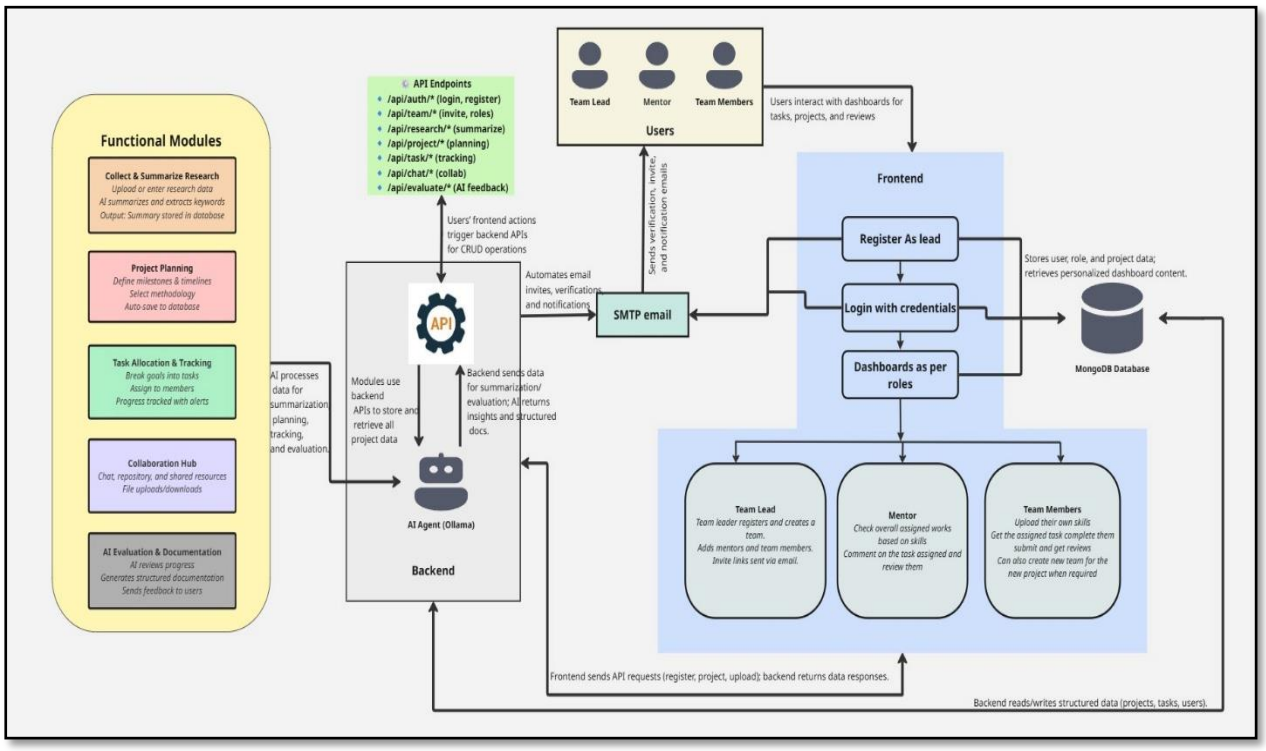

Fig. 1: System architecture

1. System Architecture and User Interface Layer

TeamCollab is designed as a modular, layered AI-assisted academic project management platform to ensure scalability, security, and maintainability. The system supports three primary user roles—Project Lead, Mentor, and Team Member—who interact through a web-based frontend. Role-based dashboards provide access to functionalities such as user authentication, project creation, task updates, and resource management, ensuring clear responsibility allocation and seamless user interaction.

2. Backend Services and Functional Modules

The backend serves as the central control unit, exposing secure API endpoints for user authentication, project planning, task allocation, collaboration, and evaluation. Core functional modules include research summarization, project planning, task allocation, collaboration, and documentation. The backend validates user requests, manages workflows, and coordinates interactions between system components.

3. AI Agent Layer with Local LLM Integration

TeamCollab integrates intelligent AI agents powered by large language models through Ollama to perform advanced academic tasks such as literature summarization, objective formulation, Scrum-based task planning, progress evaluation, and automated IEEE-style report generation. Local LLM inference ensures data privacy, offline operability, and reduced latency, making the system suitable for academic environments with limited connectivity.

4. Data Management, Communication, and System Workflow

MongoDB is employed as the primary data store for user profiles, project details, task progress, research summaries, and reports, enabling efficient data retrieval and updates. The system incorporates SMTP-based email services for automated notifications, including task assignments, deadline reminders, and evaluation alerts. User requests flow from the frontend to backend APIs, trigger AI processing when required, and are stored in the database before results are presented back to users, ensuring a seamless and efficient end-to-end workflow.

Flow Diagram

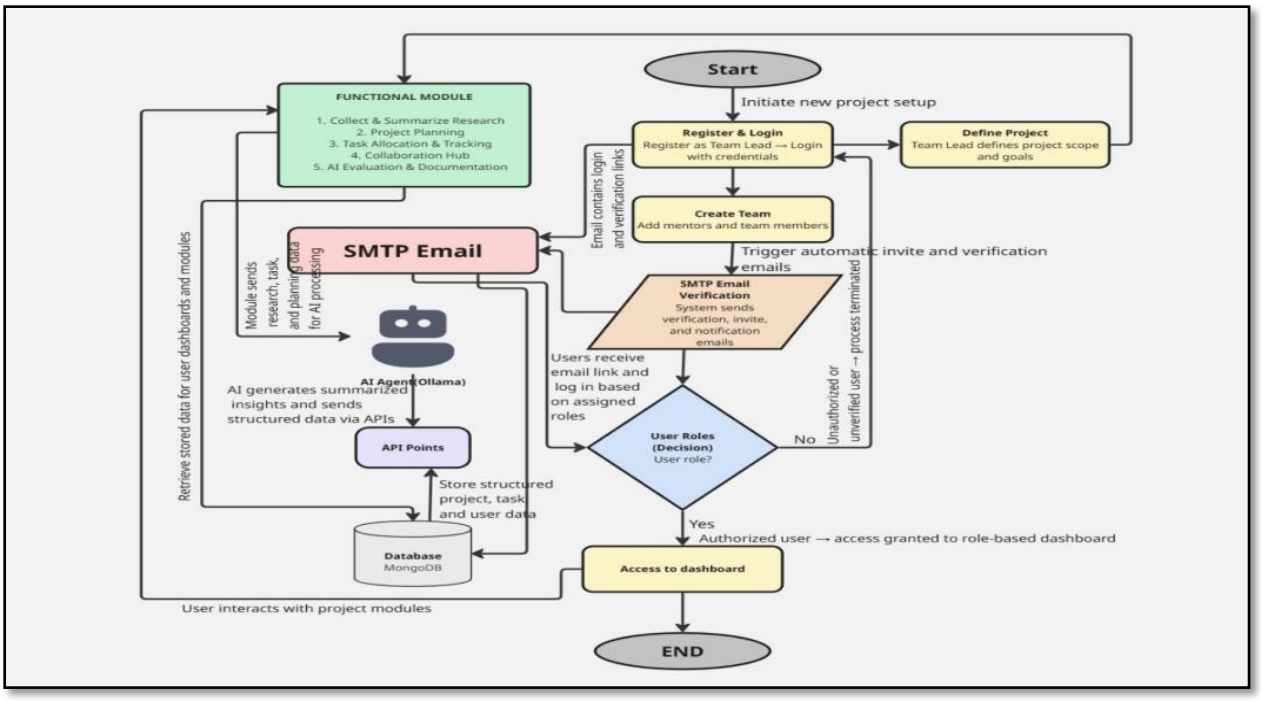

Fig. 2: Data Flow Diagram

It depicts the end-to-end workflow of the proposed system. The process begins with project initiation, user registration, and role verification through SMTP-based email authentication. Authorized users gain access to role-based dashboards, where they interact with functional modules for research collection, project planning, task allocation, collaboration, and evaluation. AI agents process user inputs through APIs and generate structured outputs, which are stored in MongoDB. This workflow ensures secure access, efficient coordination, and real-time monitoring throughout the project lifecycle.

- The process begins with the **Project Lead registering and logging in** to the system using valid credentials.
- After successful login, the **Project Lead defines the project scope, objectives, and goals**.
- The Project Lead then **creates a team** by adding mentors and team members.
- The system automatically **triggers SMTP email notifications** for user verification, invitations, and alerts.
- Invited users receive verification links and **log in based on their assigned roles**.
- The system performs **role verification** to determine authorized access.
- **Authorized users** are granted access to **role-based dashboards**, while unauthorized users are denied access.
- Users interact with various **functional modules**, including:
  - Research collection and summarization
  - Project planning
  - Task allocation and tracking
  - Collaboration and communication
  - Evaluation and documentation
- The backend invokes **AI agents (Ollama)** to generate summaries, insights, and structured data.

- All user, project, and task data are **stored and retrieved from the MongoDB database** through APIs.
- The workflow ends with users continuously interacting with dashboards for effective project execution.

Use – Case Diagram

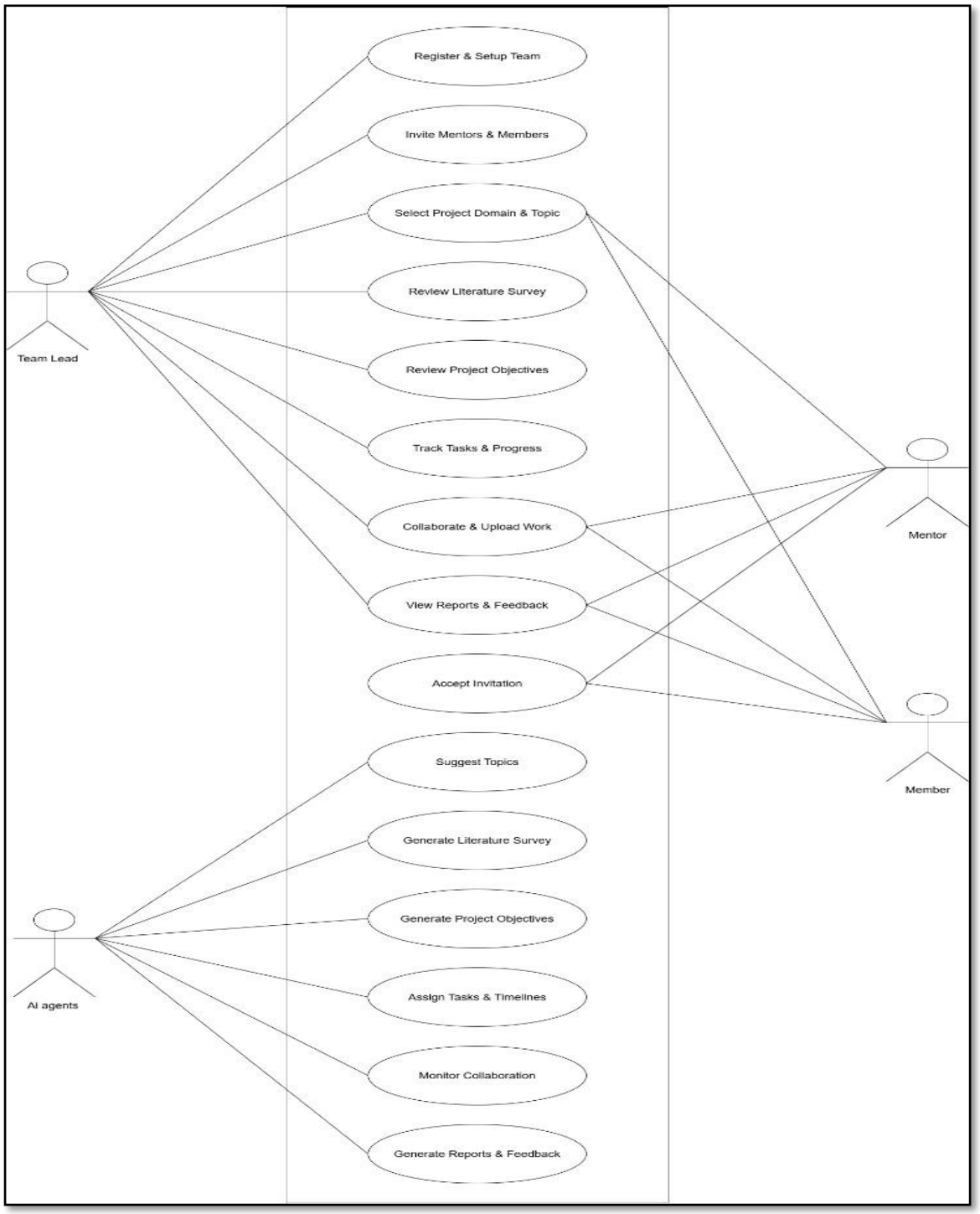

Fig. 3: Use Case Diagram for AI-Based Academic Project Management

- Project Lead uses AI agents for project topic selection, objective formulation, task allocation, and timeline planning.
- Team Members use AI assistance for literature review, task execution support, and progress updates.
- Mentors use AI-generated summaries, progress evaluations, and reports for supervision and feedback.
- AI agents automate literature surveys, task planning, progress monitoring, and report generation.
- Human users perform decision-making, review, supervision, and project execution tasks.

Project Schema

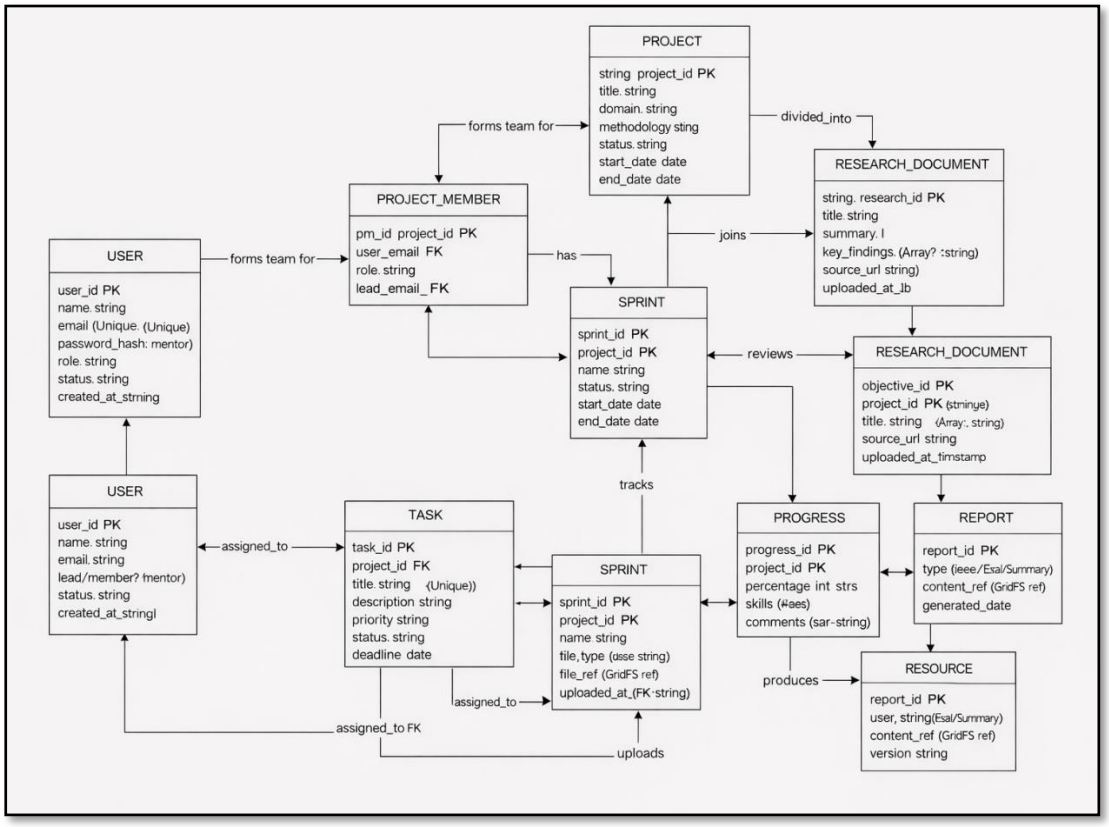

Fig. 4: Schema

IV. IMPLEMENTATION

The system is implemented using Python as the core programming language due to its flexibility and AI support. Streamlit is used for rapid development of an interactive and user-friendly web interface. MongoDB is utilized to store structured and semi-structured project data efficiently. Ollama is integrated to provide local Large Language Model (LLM) inference, ensuring data privacy and offline capability.

- **System Architecture**
  - The implementation is divided into three primary layers:
    - **Presentation Layer**: Streamlit - based dashboards for different user roles.
    - **Application Layer**: Python-based business logic and AI agent orchestration.
    - **Data Layer**: MongoDB database for persistent storage.
  - Each module operates independently, enabling easier debugging and future enhancements.

- **Technology Stack**
  - **Programming Language**: Python
  - **Frontend Framework**: Streamlit
  - **Database**: MongoDB
  - **AI Framework**: Ollama (Local LLM Execution)
  - **Project Management Model**: Scrum-based Agile methodology

- **Module-Wise Implementation**

1. **Lead Dashboard**
   - Allows project leads to create and manage projects.
   - Enables invitation of team members and mentors.
   - Supports objective definition and project configuration.
   - Provides task assignment and sprint overview.
   - Displays progress indicators and AI-assisted insights for decision-making.

2. **Mentor Dashboard**
    - Enables mentors to monitor project progress in real time.
    - Supports evaluation of submitted tasks and deliverables.
    - Integrates AI-assisted performance analysis.
    - Provides plagiarism detection and feedback suggestions.
    - Enhances academic supervision and guidance.

3. **Member Dashboard**
    - Displays assigned tasks and deadlines.
    - Allows members to update task progress.
    - Supports uploading of project files and resources.
    - Provides automated reminders and notifications.
    - Encourages collaboration and accountability.

4. **Literature Review Agent**
    - Collects relevant research papers from multiple academic sources.
    - Filters content based on project domain and keywords.
    - Generates concise summaries using LLM-based analysis.
    - Highlights research gaps and key contributions.
    - Stores summarized content in the database for future reference.

5. **Objectives Generator**
    - Uses literature summaries to generate project objectives.
    - Ensures objectives are clear, measurable, and achievable.
    - Aligns objectives with academic and project requirements.
    - Allows users to review, modify, and finalize objectives.

6. **Scrum Task Planner**
    - Divides the project into sprints and tasks.
    - Estimates timelines and task durations.
    - Matches tasks to team members based on skills.
    - Supports agile backlog creation and sprint planning.
    - Improves workload balance and deadline adherence.

7. **Report Generator**
    - Automatically generates IEEE-compliant academic reports.
    - Includes sections such as abstract, introduction, methodology, results, and conclusion.
    - Uses project data and progress updates for content generation.
    - Supports export in PDF and DOCX formats.
    - Reduces manual documentation effort.

- **Data Management**
    - MongoDB stores:
        - ❖ User profiles and roles
        - ❖ Project details and metadata
        - ❖ Tasks, sprints, and progress records
        - ❖ Literature summaries and objectives
        - ❖ Generated reports and submissions
    - Flexible schema supports evolving project requirements.

- **Security and Privacy**
    - Implements role-based access control.
    - Restricts data access based on user roles.
    - Ensures secure handling of academic data.
    - Local LLM inference via Ollama prevents data leakage to external servers.

V. RESULT AND DISCUSSION

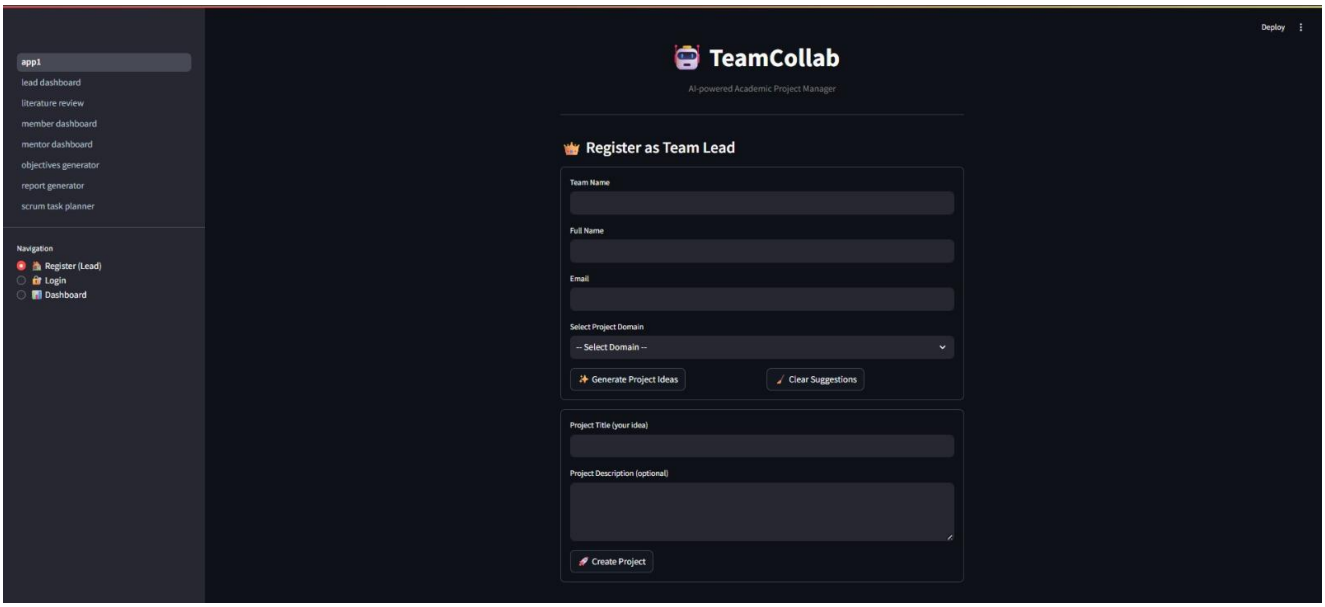

Fig. 5: Registration page

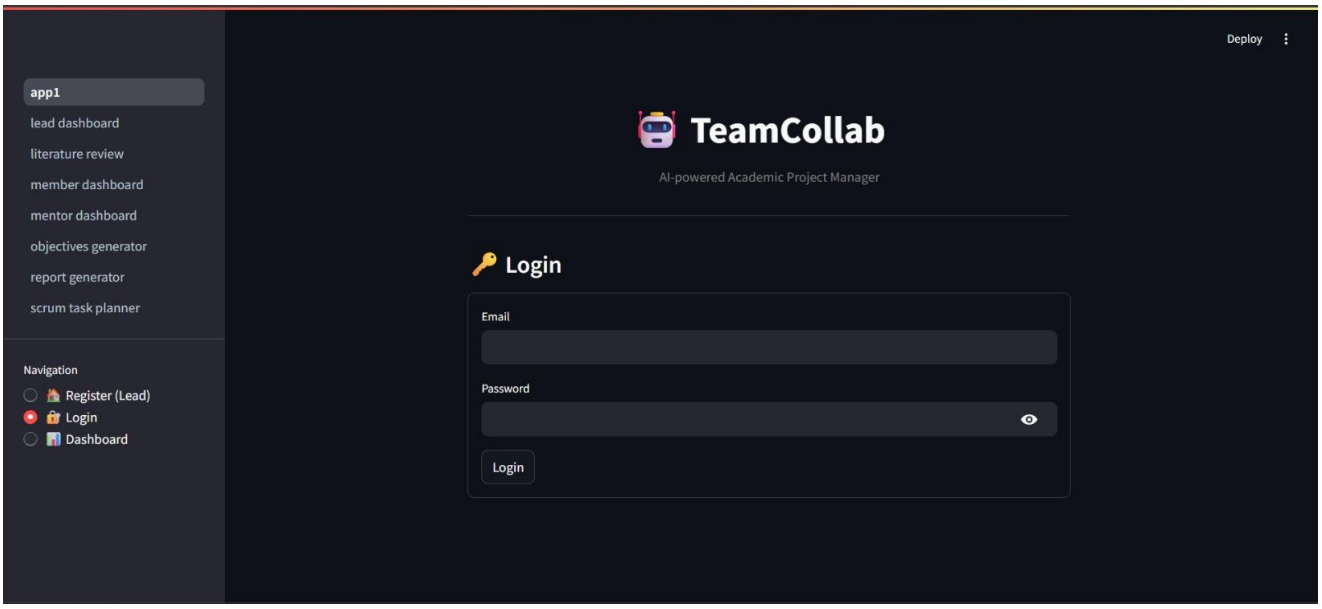

Fig. 6: Login page

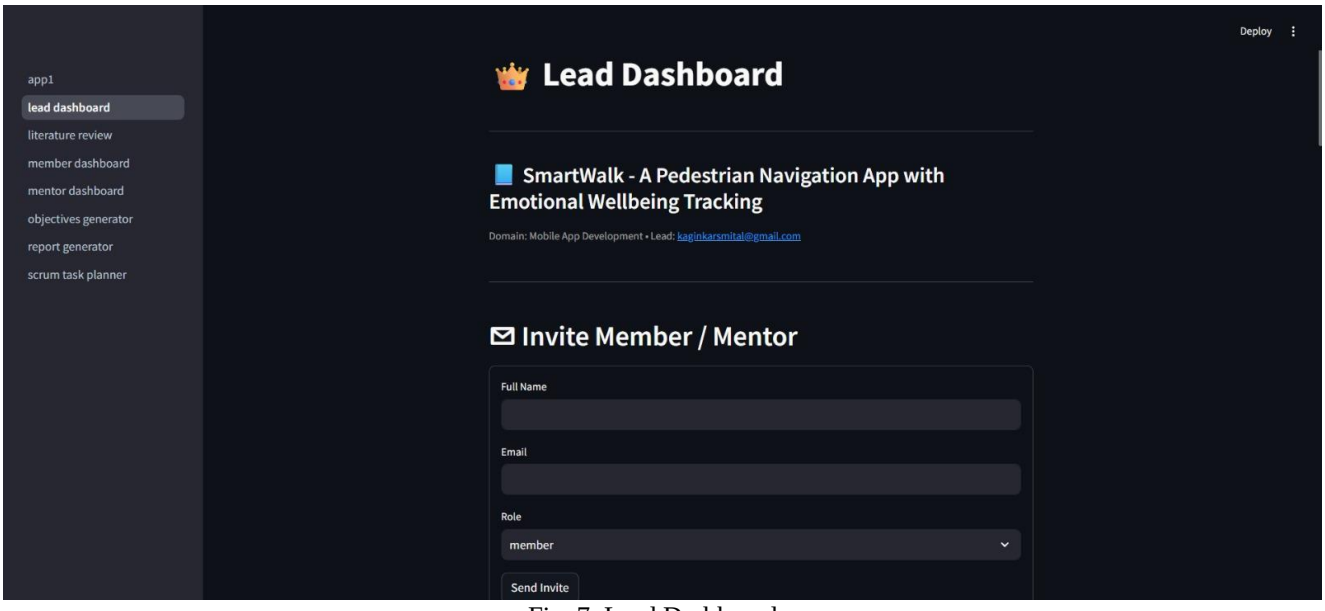

Fig. 7: Lead Dashboard page

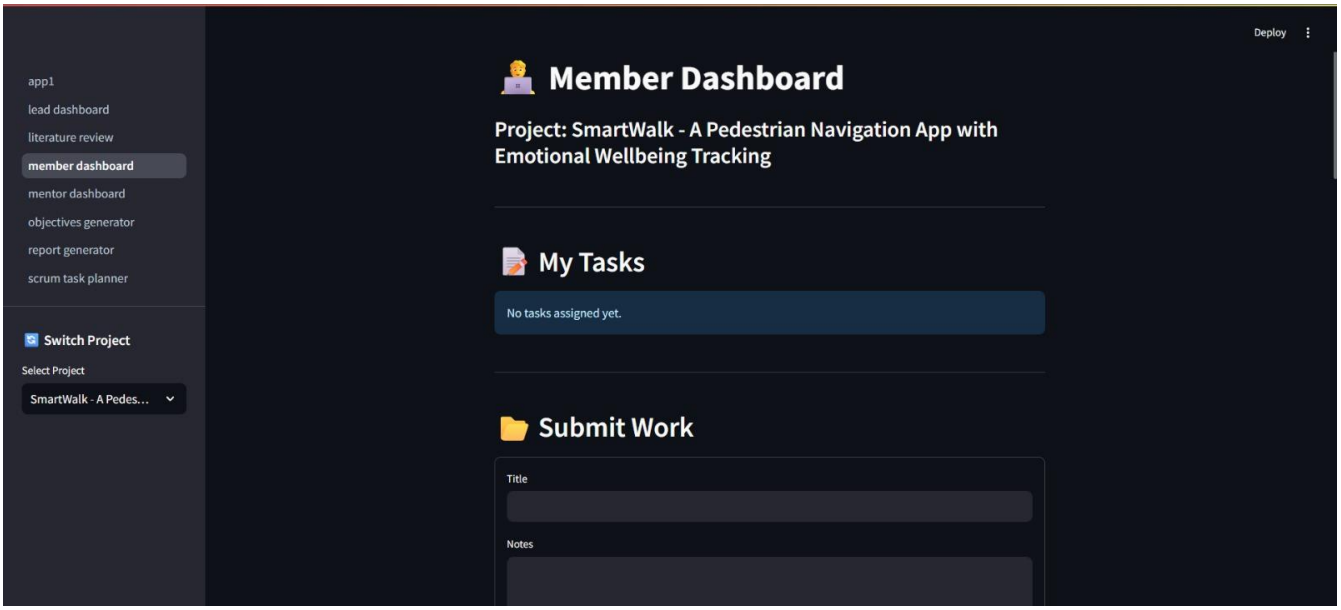

Fig. 8: Member Dashboard page

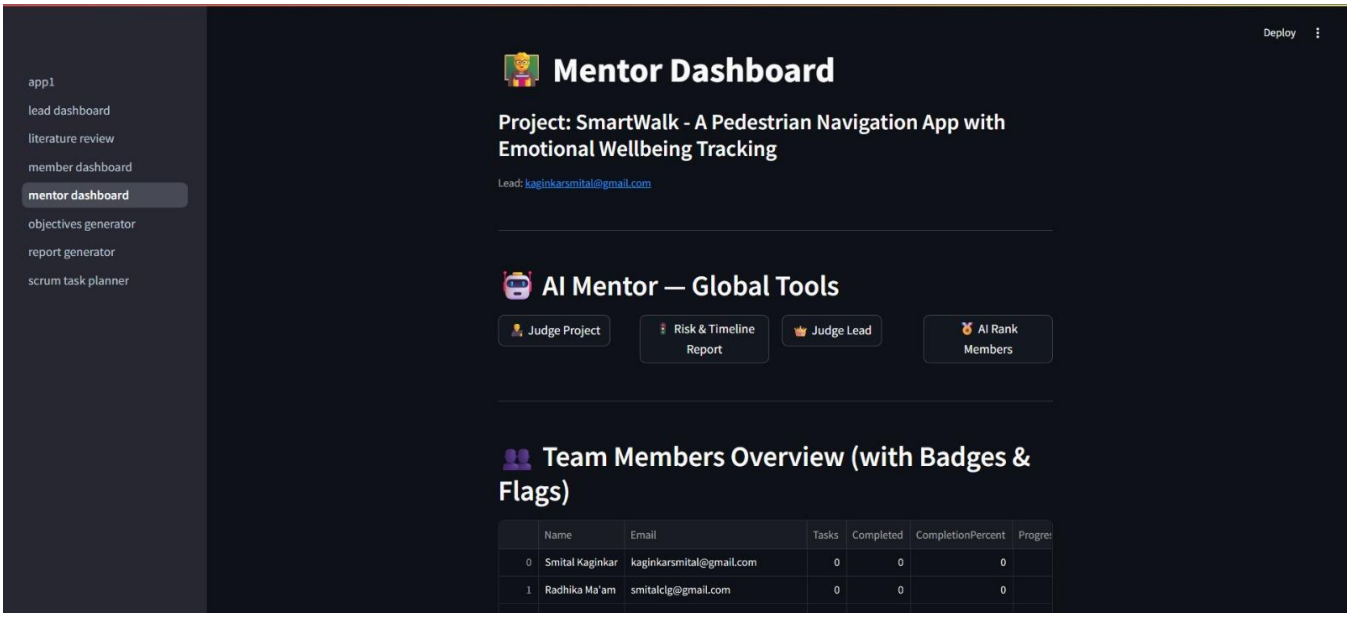

Fig. 9: Mentor Dashboard page

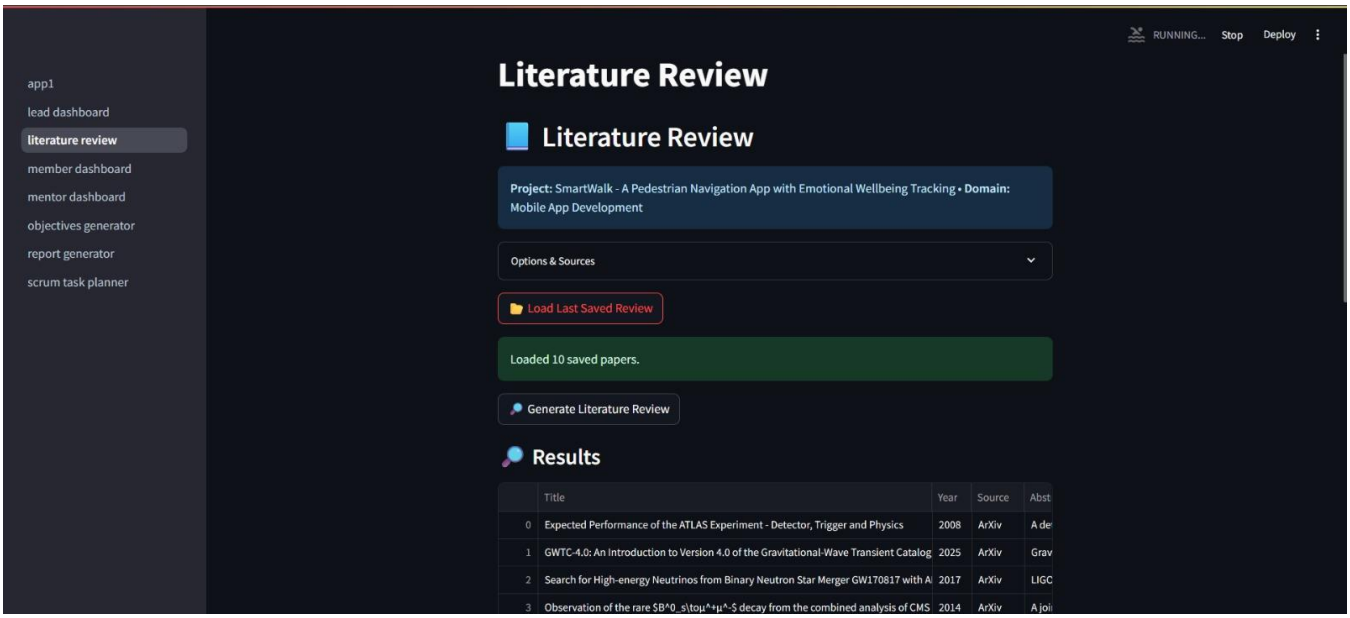

Fig. 10: Literature Review page

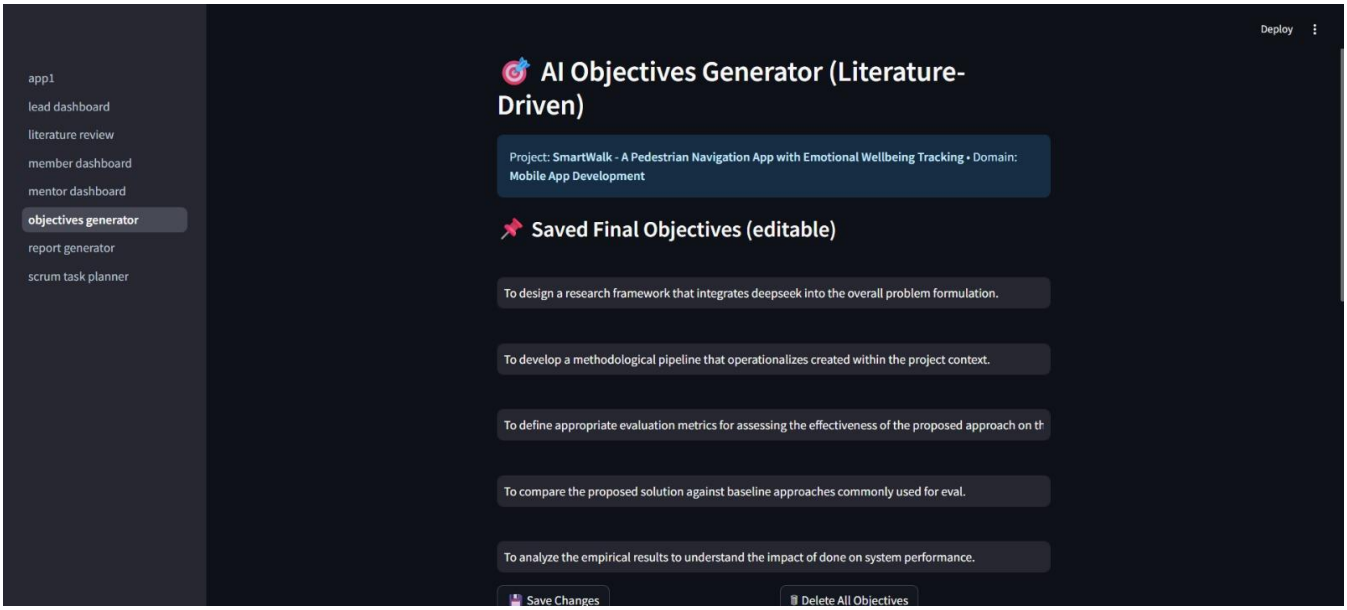

Fig. 11: Objectives generator page

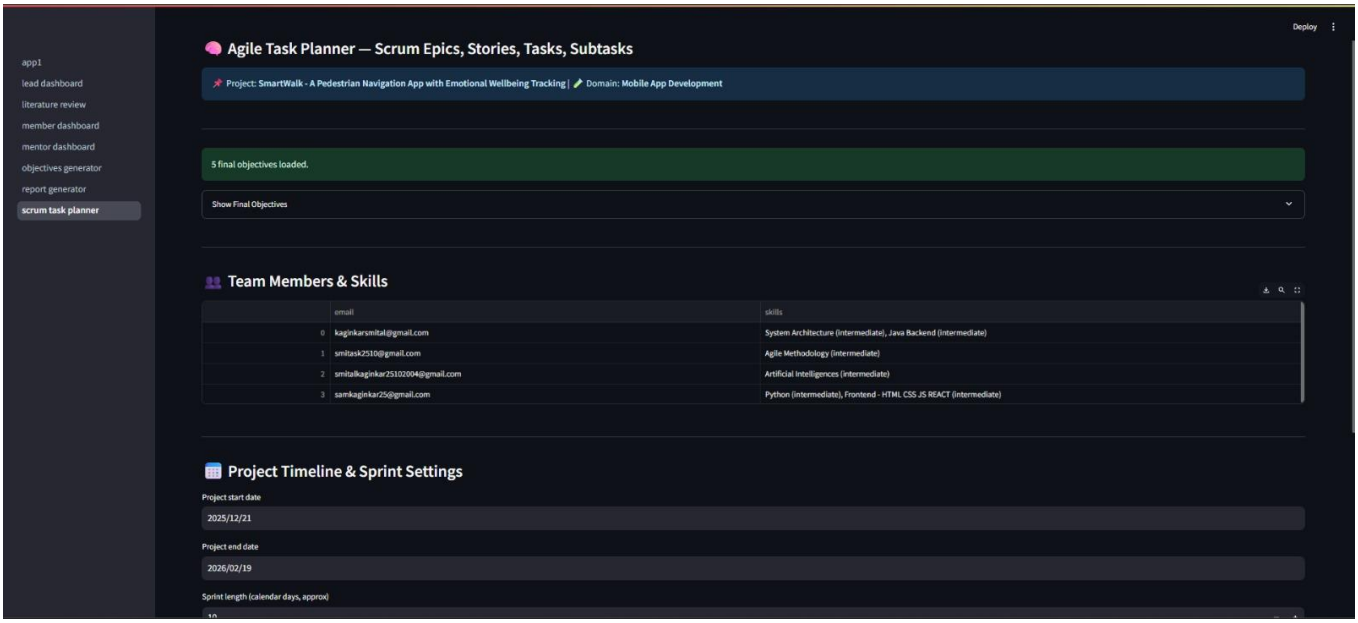

Fig. 12: Task Planner page

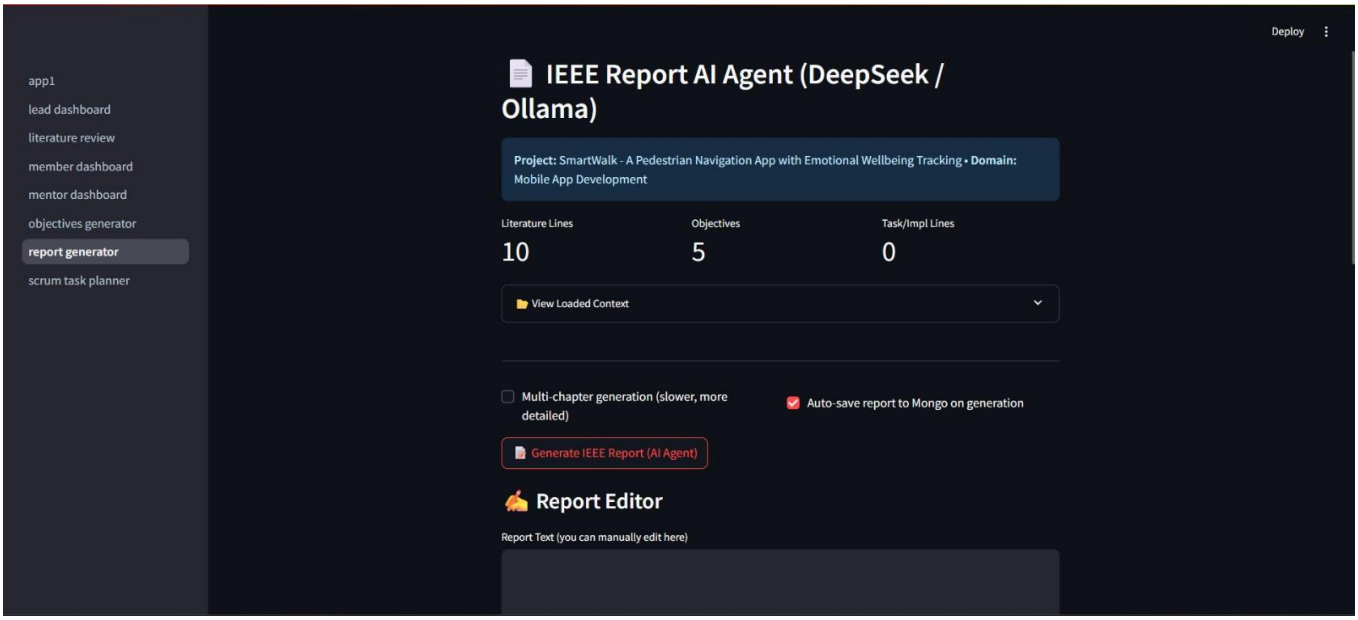

Fig. 13: IEEE Report Generator page

VI. Conclusion

Academic project management is a fundamental component of higher education, directly influencing learning outcomes, collaborative efficiency, and the successful completion of student projects. Conventional academic project management approaches often rely on manual coordination, disconnected tools, and limited mentor engagement, resulting in inefficiencies, uneven task distribution, and reduced academic quality. The proposed system, TeamCollab, addresses these limitations by integrating artificial intelligence–driven automation with structured academic workflows specifically designed for educational environments. By supporting the entire project lifecycle—from project ideation and literature review to task planning, progress tracking, and final documentation—TeamCollab enables systematic project execution. Role-based dashboards for Project Leads, Mentors, and Team Members provide clear responsibility allocation, improved coordination, and enhanced visibility into project progress, thereby fostering transparency and effective team management.

Moreover, TeamCollab enhances collaboration and accountability through real-time progress monitoring, automated notifications, and agile Scrum-based task planning, ensuring balanced workload distribution and timely milestone completion. The automation of repetitive academic tasks, including literature summarization, objective formulation, and report generation, significantly reduces manual effort while improving consistency and accuracy. AI-assisted mentor dashboards further strengthen academic supervision by enabling efficient evaluation and timely feedback. The deployment of local large language model inference using Ollama ensures data privacy, operational reliability, and offline usability, making the system suitable for institutions with constrained connectivity. Overall, TeamCollab bridges the gap between academic project requirements and industry-aligned project management practices, offering a secure, scalable, and effective platform that enhances project quality and prepares students for complex real-world project environments.

VII. Acknowledgement

The authors would like to express their sincere gratitude to the faculty of the Department of Computer Science and Engineering, Shri Dharmasthala Manjunatheshwar College of Engineering and Technology (SDMCET), Dharwad, for their valuable guidance and continuous support throughout this work. The authors also thank Prof. Radhika Amashi for her insightful guidance and encouragement. Further appreciation is extended to the Head of the Department and the Principal of SDM College of Engineering and Technology, Dharwad (Autonomous, affiliated to Visvesvaraya Technological University, Belagavi, Karnataka, India), for providing the necessary facilities and a supportive academic environment to carry out this research.

References

[1] Wrike, “AI-powered project management software,” *Wrike Inc.*, 2025. [Online]. Available: https://www.wrike.com

[2] SciSummary, “AI-based scientific article summarization tool,” 2023. [Online]. Available: https://scisummary.com

[3] A. Smith, J. Liu, and R. Kumar, “AI-based task modeling using large multimodal models and digital twins,” *IEEE Access*, vol. 13, pp. 44512–44526, 2025.

[4] Iris.ai, “AI-powered research discovery and literature review platform,” 2025. [Online]. Available: https://iris.ai

[5] M. Zhang and L. Wang, “AI-optimized task allocation for software project management,” *Journal of Systems and Software*, vol. 198, Art. no. 111235, 2024.