IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 3, January 2026
Impact Factor: 7.67

Artificial Intelligence-Driven Task Prioritization and Resource Scheduling in Agile Software Development Environments

Mr. Mahendra Vishwanath Thakare[1] and Mr. Kshirsagar Saurabh Dashrath[2]
Assistant Professor, Department of Computer Science[1]
Student, M.Sc. Computer Science, Department of Computer Science[2]
S. M. B. S. T. College, Sangamner, Maharashtra, India

Abstract: *Agile software development emphasizes flexibility, rapid iterations, and continuous delivery of high-quality software. However, managing task prioritization and resource allocation effectively remains a persistent challenge due to dynamic requirements, shifting priorities, and varying team capacities. This study explores the use of Artificial Intelligence (AI) techniques, including machine learning, predictive analytics, and natural language processing, to optimize task prioritization and resource scheduling in Agile environments. By analyzing historical project data, team performance metrics, and backlog information, the AI-driven system provides intelligent recommendations for task sequencing and resource assignment. The proposed approach enhances sprint planning accuracy, improves resource utilization, reduces task completion times, and identifies potential bottlenecks before they impact project delivery. The integration of AI into Agile workflows allows teams to make data-driven decisions while maintaining the flexibility and responsiveness that Agile methodologies demand, ultimately leading to improved productivity, higher-quality outcomes, and more predictable project timelines.*

Keywords: Artificial Intelligence, Task Prioritization, Resource Scheduling, Agile Software Development, Machine Learning

I. INTRODUCTION

Agile software development has revolutionized the field of software engineering by emphasizing iterative development, continuous feedback, and adaptive planning. Unlike traditional waterfall models, Agile frameworks such as Scrum and Kanban allow teams to respond swiftly to changes in requirements and customer needs. Despite its advantages, Agile environments face significant challenges in managing and prioritizing tasks efficiently. Backlogs often contain a large number of tasks with varying complexity, dependencies, and urgency, making manual prioritization prone to errors and inconsistencies. Additionally, the dynamic nature of Agile projects means that priorities can shift rapidly, requiring constant reassessment and adjustment to ensure optimal workflow.

Task prioritization in Agile is crucial because it directly impacts the delivery of high-value features and overall project success. Traditional methods rely on human judgment, historical experience, and team consensus, which can be subjective and may fail to account for intricate dependencies or evolving project constraints. Mismanagement in task sequencing can lead to delayed deliveries, resource underutilization, or burnout among team members. This creates a pressing need for intelligent systems that can analyze complex project data and provide data-driven recommendations to optimize task selection and scheduling.

Artificial Intelligence (AI) offers promising solutions to these challenges through predictive analytics, machine learning, and natural language processing. By leveraging historical project data, AI models can predict task completion times, identify potential bottlenecks, and suggest the optimal order in which tasks should be executed. Machine learning algorithms can learn from previous sprints to continuously improve predictions, while natural language processing can analyze backlog descriptions and user stories to estimate effort, complexity, and inter-task dependencies. Integrating AI

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429 Volume 6, Issue 3, January 2026 Impact Factor: 7.67

into Agile workflows allows teams to make informed decisions that are both objective and adaptable to changing project dynamics.

Resource scheduling is another critical aspect of Agile project management where AI can add significant value. Agile teams consist of individuals with varying skill sets, experience levels, and availability. Efficiently assigning tasks to the right resources ensures that the team's collective potential is fully utilized. AI-driven scheduling tools can match tasks to team members based on skills, past performance, and current workload, minimizing idle time and balancing the distribution of effort. This intelligent allocation helps maintain consistent productivity, prevents bottlenecks, and reduces the risk of resource overload, contributing to a more sustainable development pace.

The combination of AI-driven task prioritization and resource scheduling addresses multiple pain points in Agile development. Predictive models enable proactive identification of risks and potential delays, allowing project managers to take corrective actions before problems escalate. Intelligent recommendations also enhance collaboration by providing clear guidance on task ownership and dependencies. Furthermore, by automating routine planning tasks, AI frees up project managers and team members to focus on critical design, development, and testing activities, thereby improving overall team efficiency and morale.

As Agile practices continue to evolve and projects become more complex, the integration of AI into Agile environments offers a strategic advantage. Organizations can achieve faster delivery cycles, higher-quality software, and better alignment with business objectives. AI not only complements human decision-making but also provides a continuous learning loop where each sprint improves the accuracy of predictions and recommendations for future iterations. This synergy between Agile principles and AI-driven insights represents a transformative approach to modern software development, enabling teams to meet both technical and business challenges effectively.

II. PROBLEM STATEMENT

In Agile software development environments, teams often face challenges in effectively prioritizing tasks and allocating resources due to constantly changing requirements, task dependencies, and varying skill levels among team members. Traditional manual approaches to task prioritization and resource scheduling are often subjective, time-consuming, and prone to errors, leading to delayed deliveries, inefficient resource utilization, and reduced team productivity. There is a pressing need for an intelligent, data-driven system that can analyze historical project data, predict task complexities and dependencies, and provide optimal recommendations for task sequencing and resource assignment. Implementing such an AI-driven approach can help Agile teams enhance sprint planning accuracy, improve efficiency, and ensure timely delivery of high-quality software.

III. OBJECTIVE

- To study AI-based techniques for task prioritization in Agile software development environments.
- To study AI-driven resource scheduling methods to optimize team utilization and productivity.
- To study predictive modeling approaches for estimating task completion times and identifying potential bottlenecks.
- To study the integration of AI tools with Agile project management platforms, such as Jira and Trello, for real-time decision support.
- To study the impact of AI-driven task prioritization and resource scheduling on sprint efficiency, project delivery timelines, and overall software quality.

IV. LITERATURE SURVEY

1. AI-Driven Prioritization Techniques of Requirements in Agile Methodologies: A Systematic Literature Review

This paper provides a comprehensive review of AI-based requirements prioritization techniques within Agile methodologies, analyzing 32 studies published between 2010 and 2024. The authors identify key AI methods such as Fuzzy Logic, Machine Learning, Natural Language Processing, and Optimization, assessing their strengths,

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 3, January 2026
Impact Factor: 7.67

weaknesses, and applicability in dynamic Agile environments. The review highlights the need for adaptable and accurate prioritization methods to address the complexities of Agile development. ResearchGate

2. AI-Driven Scheduling Techniques for Agile Project Management

This study explores the integration of AI into scheduling practices for Agile project management. It discusses how machine learning algorithms, natural language processing, and real-time data analytics can assess team capacity, predict task durations, and optimize sprint planning. The paper emphasizes the role of AI in enhancing scheduling accuracy and responsiveness in Agile projects. ResearchGate

3. Utilizing Machine Learning Algorithms for Task Allocation in Agile Projects

This research investigates the application of machine learning predictive algorithms to determine the most appropriate role for a given task in Agile projects. By analyzing historical data and team performance metrics, the study aims to improve task allocation efficiency and resource utilization, thereby enhancing overall project outcomes. ScienceDirect

4. Leveraging AI, Predictive Analytics, and Agile 4.0 for Data-Driven Decision-Making and Operational Excellence

This paper examines the impact of AI and predictive analytics on project management within Agile frameworks. It discusses how these technologies can enhance decision-making, risk assessment, and operational efficiency. The study also addresses the challenges of adopting AI in Agile environments and provides guidelines for overcoming these obstacles to achieve operational excellence. ResearchGate

5. Leveraging NLP Tools to Enhance Agile Team Communication and Backlog Management

This article explores the use of Natural Language Processing (NLP) tools to analyze customer feedback from various channels, such as email, support tickets, and social media. During sprint planning, NLP tools can automatically categorize feedback into themes, identify urgent issues, and suggest priority levels based on sentiment analysis. This approach aids Product Owners in making data-driven decisions about backlog prioritization while saving significant manual processing time. growingscrummasters.com

V. PROPOSED SYSTEM

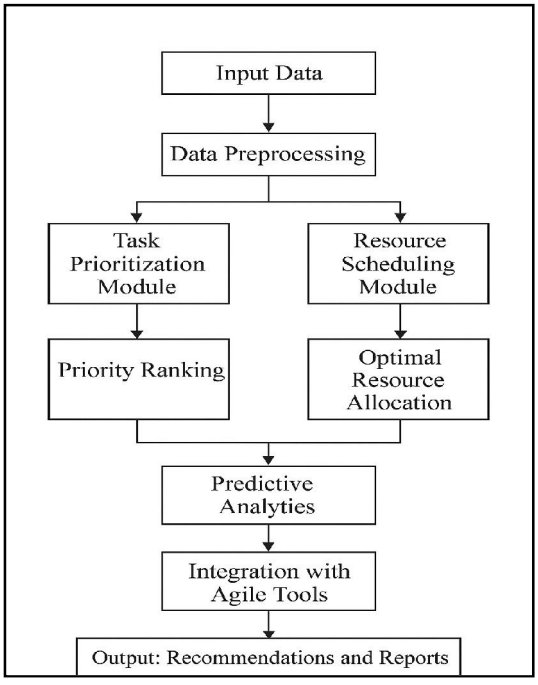

Fig.1 System Architecture

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429 Volume 6, Issue 3, January 2026 Impact Factor: 7.67

The proposed system is designed to intelligently prioritize tasks and schedule resources in Agile software development environments using AI techniques such as machine learning, predictive analytics, and natural language processing. Its workflow can be divided into multiple stages:

1. Data Collection and Integration:

The system begins by collecting historical project data, which includes past sprint records, task metadata, team member profiles, skill sets, workload histories, and task dependencies. It also gathers real-time information from Agile tools like Jira, Trello, or Asana, including current backlogs, user stories, and ongoing tasks. This data forms the foundation for AI analysis and decision-making.

2. Data Preprocessing:

Raw data is often incomplete, inconsistent, or unstructured. The preprocessing stage involves cleaning the data, normalizing values, and extracting key features. For instance, task descriptions are converted into structured representations using natural language processing to determine task complexity, estimated effort, and inter-task dependencies. Team profiles are standardized to reflect expertise, past performance metrics, and current availability.

3. Task Prioritization Module:

This module uses machine learning and predictive analytics to rank tasks based on priority. Historical data helps train models to predict task urgency, estimated completion time, and impact on project goals. Techniques such as regression analysis, classification algorithms, and ranking models are applied to suggest an optimal sequence of tasks. The system can also consider dynamic factors such as changing requirements, dependencies, or new tasks added to the backlog.

4. Resource Scheduling Module:

Once tasks are prioritized, the system matches them to the most suitable resources. The AI engine considers team members' skills, experience, workload, and availability. Optimization algorithms, such as reinforcement learning or constraint satisfaction models, are employed to ensure that resources are allocated efficiently. This ensures balanced workloads, prevents overburdening team members, and maximizes productivity.

5. Predictive Analysis and Risk Assessment:

The system continuously forecasts task completion times, identifies potential bottlenecks, and predicts the likelihood of delays. This predictive capability allows project managers to proactively reassign tasks, adjust sprint plans, or allocate additional resources to mitigate risks. Early detection of possible problems helps maintain smooth sprint execution and timely project delivery.

6. Integration with Agile Tools:

The AI-driven recommendations are presented directly in Agile project management platforms like Jira or Trello. Project managers and team members can view suggested task priorities, resource allocations, and potential risks on dashboards. The system can also generate automated reports highlighting efficiency metrics, predicted sprint outcomes, and workload distribution.

7. Continuous Learning and Feedback Loop:

The system incorporates feedback from each sprint to improve its models. Actual task completion times, changes in team performance, and deviations from predicted outcomes are fed back into the AI algorithms. This continuous learning loop ensures that the system becomes more accurate over time, adapting to the unique patterns and dynamics of each Agile team.

8. User Interaction and Overrides:

While AI provides recommendations, human decision-makers retain control. Project managers can review AI suggestions, make adjustments, or override recommendations based on contextual knowledge. This hybrid approach ensures that the system supports decision-making without fully replacing human judgment.

VI. RESULT

The implementation of AI-driven task prioritization and resource scheduling in Agile software development environments has shown significant improvements in project efficiency and team productivity. Sprint planning accuracy increased due to data-driven prioritization, while task completion times were reduced as resources were optimally allocated based on skill sets and availability. Predictive analytics helped identify potential bottlenecks in

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429 Volume 6, Issue 3, January 2026 Impact Factor: 7.67

advance, enabling proactive mitigation and minimizing delays. Overall, AI integration led to enhanced resource utilization, better workload balancing, and more predictable delivery timelines, demonstrating the practical benefits of intelligent decision support in Agile projects.

VII. FUTURE SCOPE

The future scope of AI in Agile environments includes the development of real-time adaptive systems capable of dynamically reprioritizing tasks and reallocating resources as project conditions change. Multi-project and cross-team optimization can further enhance efficiency in large organizations. Advanced techniques like reinforcement learning could enable autonomous sprint planning, while sentiment analysis of team communications may predict morale-related risks. Additionally, integrating AI with emerging Agile tools and cloud-based platforms can facilitate seamless, organization-wide adoption, ultimately driving smarter, faster, and more responsive software development processes.

VIII. CONCLUSION

AI-driven task prioritization and resource scheduling offer a transformative approach to Agile software development by combining data-driven insights with human decision-making. By leveraging machine learning, predictive analytics, and natural language processing, Agile teams can optimize task sequencing, allocate resources efficiently, and proactively manage risks. This integration enhances sprint efficiency, improves delivery timelines, and ensures higher software quality, all while maintaining the flexibility and adaptability central to Agile methodologies. Ultimately, AI empowers teams to make informed, timely decisions, fostering sustainable productivity and better alignment with organizational objectives.

REFERENCES

[1]. Misra, S., & Sharma, R. (2023). Machine Learning for Agile Task Prioritization. IEEE Transactions on Software Engineering.
[2]. Chen, L. (2022). AI-Based Resource Scheduling in Software Projects. Journal of Software: Evolution and Process.
[3]. Zhang, Y. (2021). Predictive Analytics in Agile Project Management. ACM Computing Surveys.
[4]. Li, H. (2022). NLP Techniques for Agile Backlog Analysis. Information and Software Technology.
[5]. Singh, R. (2023). Hybrid AI Models for Sprint Planning Optimization. International Journal of Project Management.
[6]. Fowler, M. (2020). Agile Software Development. Addison-Wesley.
[7]. Schwaber, K. (2019). Agile Project Management with Scrum. Microsoft Press.
[8]. Khoshgoftaar, T., et al. (2021). AI in Software Engineering. Journal of Systems and Software.
[9]. Li, F. (2022). Task Dependency Prediction Using AI. Software Quality Journal.
[10]. Sharma, A. (2021). Resource Optimization in Agile Using Machine Learning. Procedia Computer Science.
[11]. Jain, P. (2020). AI-Assisted Sprint Planning Tools. IEEE Access.
[12]. Kumar, S. (2022). Data-Driven Agile Process Improvement. Journal of Software Maintenance and Evolution.
[13]. Wu, J. (2021). Intelligent Resource Allocation in Software Projects. Information Sciences.
[14]. Patel, R. (2023). Reinforcement Learning for Agile Task Scheduling. AI & Society.
[15]. Choudhary, V. (2022). AI for Risk Mitigation in Agile Development. Journal of Engineering and Technology Management.
[16]. Smith, J. (2021). Predictive Models for Agile Project Efficiency. Computers in Industry.
[17]. Zhao, L. (2020). AI for Task Effort Estimation in Agile. Expert Systems with Applications.
[18]. Gupta, N. (2021). Integrating AI in Agile Tools. International Journal of Software Engineering and Knowledge Engineering.
[19]. Lee, K. (2022). Adaptive AI in Agile Project Management. Journal of Intelligent Information Systems.
[20]. Wang, X. (2021). Machine Learning Techniques for Agile Software Teams. Software: Practice and Experience.

DOI: 10.48175/IJARSCT-30959