An Intelligent Task Management System Using Real-Time Prioritization and Skill-Based Work Allocation

Sujith Kumar[1], Divya S P[2] and Hemanth Kumar[3]
[1]Student, Dept. of MCA, Jawaharlal Nehru New College of Engineering, Shivamogga
[2]Assistant Professor, Dept. of MBA, Jawaharlal Nehru New College of Engineering, Shivamogga
[3]Associate Professor, Dept. of MCA, Jawaharlal Nehru New College of Engineering, Shivamogga

Abstract*—In present times, task assignment and prioritization becomes a toughest job in various organizations as it has multiple teams contains different roles and skills, the process becomes much more complex. Using manual task assignment techniques, the work does not get distributed evenly and poor prioritization of tasks leads to delays in execution. Therefore, the overall productivity of the company will be reduced. To fix this problem, a smart system is designed that makes the work allocation task automatic and prioritizing them in real time. For determining task priority levels, an AI based technique is used that updates level based on deadlines. A skill matching mechanism is implemented to find best fit employee so that tasks are distributed evenly based on employee skills. The system is accessible for different user roles such as Admins, Managers, Team Leads and Members. Basic things like System access, approvals, and accountability can be managed efficiently. This system supports Report generation using which users can monitor progress and evaluate performance which further helps in taking better decisions. This work showcases a practical and scalable approach to make work management better in modern organizations.

***Index Terms*—Automatic Task Prioritization, Organizational Performance Monitoring, Real-Time Systems, Skill-based Task Assignment, Workflow Automation.**

I. INTRODUCTION

Effective task management and prioritizations matter a lot when it comes to increase organization productivity, timely task execution, and better usage of resources. As organizations grow fast having multiple branches and departments, there is increase in complexity also. Employees may face difficulty in managing tasks, monitoring progress and taking responsibilities. Traditional task management methods completely depend on manual supervision and static task prioritizations can result in inefficient workload distribution, unclear accountability and delayed task execution. Another notable limitation seen in traditional systems is the absence of skill-based task allocation. Tasks may be distributed wrongly to employees which can reduce the quality of work and result in rework. Before assigning a task, Manager and Team Leads should review skills of employees which adds extra efforts and increases the possibility of human error. This work presents a solution to these problems as a smart system designed for making task management, real time prioritization and skill-based task assignment easy using AI based strategies. By considering upcoming deadlines tasks priority level updated automatically so that urgent activities will be done without delay. For allocating tasks to appropriate teams the skill-matching feature helps a lot. These will reduce manual effort in decision making. The proposed system provides support for different users such as admin, managers, team leads and members. It also supports real time task tracking and report generation that allows users to evaluate performance and decisions to be made based on data. By combining all these automations, the system provides a way to make organizational work more efficiently, maintain transparency, and complete tasks on time.

II. LITERATURE REVIEW

The study by Manazir Musthafa et al. in [1] uses machine learning models to rank tasks based on urgency and deadlines. It helps users manage time better, but does not handle task assignment to employees. Similarly, Sarveshsri et al. in [2] introduced a Task Flow platform that improves task tracking and workload balance using AI. It works well for monitoring, but misses skill-based task assignment. Samriddhi Singh et al. in [3] explains key features like task creation, prioritization, and tracking that help teams work better. Still, issues with integration and user adoption remain, showing the need for smarter systems. Another effort by

Bugayenko et al. in [4] proposed a backlog-based prioritization method that improves task ranking accuracy. But it mainly focuses on prioritization and does not handle task assignment or overall performance. Moreover, Melhem et al. in [5] explains that task prioritization improves time management and reduces stress. It also shows the importance of communication and work environment, but focuses more on management than technology. Abid et al. in [6] developed an AI-based scheduler that automatically organizes tasks using user behavior. It improves personal productivity, but does not focus on team or organizational task management. Taking a different approach Vishaliney et al. in [7] presents an AI-based to-do list that uses multiple models and techniques like BERT to understand and sort tasks. It works well for personal use but is not suited for team task management. Shetty et al. in [8] introduced an Employee Task Manager with dashboards and real-time updates for tracking progress. It improves communication, but gives less focus to smart prioritization and skill-based assignment. Another system by Sukstrienwong et al. in [9] uses a skill matching algorithm to assign tasks and supports real-time updates. It improves workload balance but pays less attention to task prioritization. The study by Bugayenko et al. in [10] uses machine learning to predict task importance. It shows that some areas are well explored, while others still need improvement for better systems. Gopalakrishnan et al. in [11] proposed a framework that allows switching tasks based on priority using optimization algorithms. It focuses more on theoretical models than real-world task management systems. Jain et al. in [12] presents a React-Redux application for task assignment and performance tracking. It helps in monitoring employees and making decisions, but gives less focus to automatic task prioritization. The study by Kandemir et al. in [13] combines skill matching and workload balancing to assign tasks efficiently. It considers both human and technical factors. Deviprasad et al. in [14] This work uses machine learning to automate HR tasks and reduce manual effort. But it also faces issues like privacy, bias, and high cost. Ugwumba et al. in [15] This study uses a Deep Q Network to improve task prioritization based on deadlines and complexity. It gives better accuracy than traditional methods but does not focus on task assignment or performance tracking. Garg et al. in [16] proposed a rule-based scheduler that assigns tasks using availability and deadlines. It improves efficiency, but lacks advanced learning or adaptive decision-making. Bafana et al. in [17] did a web-based task management system for creating, assigning, and tracking tasks. It supports structured monitoring, but depends on manual assignment and lacks smart decision-making. The research by Pechová et al. in [18] compares process-based and goal-based task assignment methods. It shows that employee preferences vary, and choosing the right approach can improve team performance. Druzhynin et al. in [19] proposed a model that combines task priority and employee skills to assign work effectively. It can adjust assignments when conditions change, improving transparency and project time. The work by Kumar et al. in [20] presents an automated task management system with scheduling and notifications. It uses machine learning to assign tasks based on skills and deadlines, improving task completion and engagement.

III. METHODOLOGY

A. Research Approach

This study uses a system development approach to design and build an automated system that can help organizations distribute work among employees more easily. The main goal of the system is to reduce the manual effort people have when they assign tasks, prevent one employee from having too much work, and make sure the workload is shared fairly among team members. The system also looks at the skills that employees have, so tasks can be given to the right person, and it uses a task priority method so important tasks can be handled first. During the development process, the system architecture was designed first. After that, backend algorithms were developed that could decide which tasks should get higher priority and which employee would be suitable for a task. Finally, a simple and easy to use web interface was added so users can interact with the system and manage their tasks without difficulty.

B. Automatic Task Prioritization Mechanism

The proposed system uses a real-time task prioritization method to automatically update the priority of tasks based on their deadlines. The system regularly checks all the active tasks and calculates how much time is left before each task's

deadline. Based on the remaining time, the system automatically sets the task priority so that tasks with closer deadlines get higher priority.

The following rules are used to decide the priority of tasks:

Remaining Time	Assigned Priority
Less than 2 days	High
2 – 5 days	Medium
More than 5 days	Low

Table 1. Task Priority Rules

As illustrated in Table 1, the system keeps updating task priorities whenever there are changes in deadlines or status, helping the team focus on urgent work first.

C. Skill-Based Task Allocation

To distribute work properly, the system uses a skill-matching method to find the most suitable team member for a task. Each task has a list of required skills, and every employee profile has the skills they possess. The system compares both lists and calculates a score to see how well the employee's skills match the task requirements.

The skill match score is computed using the following formula, as shown in Eq. (1):

SkillMatchScore = (Number of Matching Skills / Total Required Skills) × 100 .....Eq(1)

where, Number of Matching Skills refers to skills that the employee already possesses; Total Required Skills refers to total skills needed for the task; and SkillMatchScore is the percentage showing how well the employee matches the task.

The algorithm works in a few steps. First, the system gets the list of skills required for the task. Then it collects the skill details of all team members in that team. After that, the system compares the required skills with the skills of each employee and calculates a skill match score for everyone. The employee who has the highest score is selected, and the system automatically assigns the task to that person. This way, tasks are given to employees who have the most suitable skills to complete them.

IV. PROPOSED MODEL

A. Proposed System

The proposed system is an automated task manager and work distribution platform which uses AI mechanisms for assigning tasks to employees and automated task prioritizations. Different users like Admin, Managers, Team Leads and Members use the system easily by allowing them to do tasks like Creating, Assigning and reviewing them. It automatically updates task priority level based on coming deadlines and warns users about the task status. Skill matching algorithm is used to identify the right employees for the tasks. Also report generating feature plays important function for evaluating overall productivity of the organization. The system designed and developed in a simple manner which is easy to understand for users without any confusions.

B. System Architecture

Fig. (1) represents the architecture of the system. The first layer is the user interface where different users like the admin, manager, team lead, and team members access the system. Requests are sent by this layer using APIs whenever user perform actions like creating new task or updating task status. The server does most of the core processing that keeps everything running smoothly as it takes care of activities such as processing the requests, applying necessary logic, and make sure everything works as expected. All the information lives in one place, a database. whenever the system needs to show something, it simply fetches from there and keeps everything organized and up to date. This structure allows smooth communication between the components so that tasks managed efficiently with clear workflows and reliable data handling.

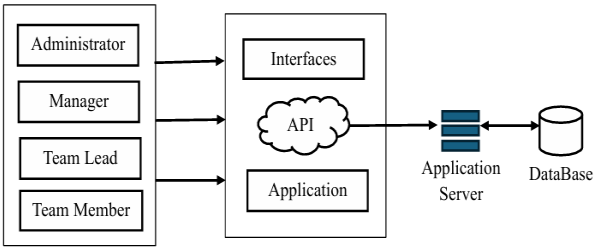


Fig. (1) System Architecture

V. MODEL EVALUATION

A. Evaluation Approach

Since the proposed tool is web based, functional and scenario-based testing were performed to evaluate the system whether it works correctly and gives

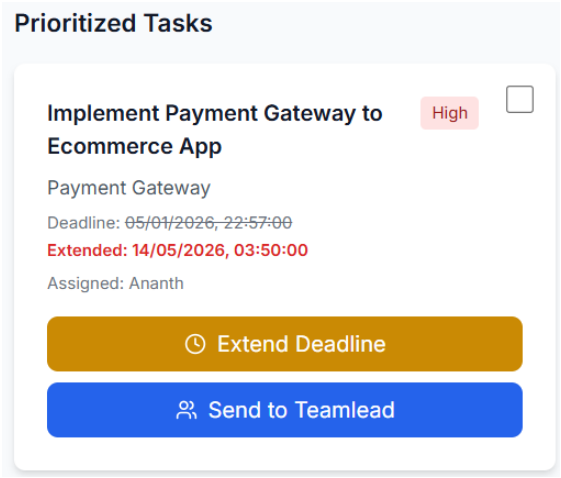
expected results under different practical conditions. Key features such as task prioritization, skill-based assignment, workload distribution, and role-based access are the primary focus of these scenarios. By testing the system under these scenarios, it becomes easy to verify whether it performs tasks correctly and simplifies the way work flows within the organization.

B. Test Scenarios

Various real-time scenarios were used to test the system's performance. Tasks with varying deadlines were set up to see how the system adjusts priorities making sure that urgent tasks are handled first. In another scenario, tasks that required specific skills were introduced to see if the system would assign them to the team members best suited for them based on their expertise. To further evaluate the system, multiple tasks were created at the same time to observe how the workload is sharing shared among team members, ensuring that no individual is burdened with too many tasks. The system was also tested by logging in with different user roles such as admin, manager, team lead and team member to make sure each role has the right permissions and can access only the features meant to be.

VI. RESULTS

The developed system produced effective results in task management and workload distribution within the organization. Different features of the system were observed during the implementation. The following figures illustrate some of the major functionalities and outputs generated by the system.



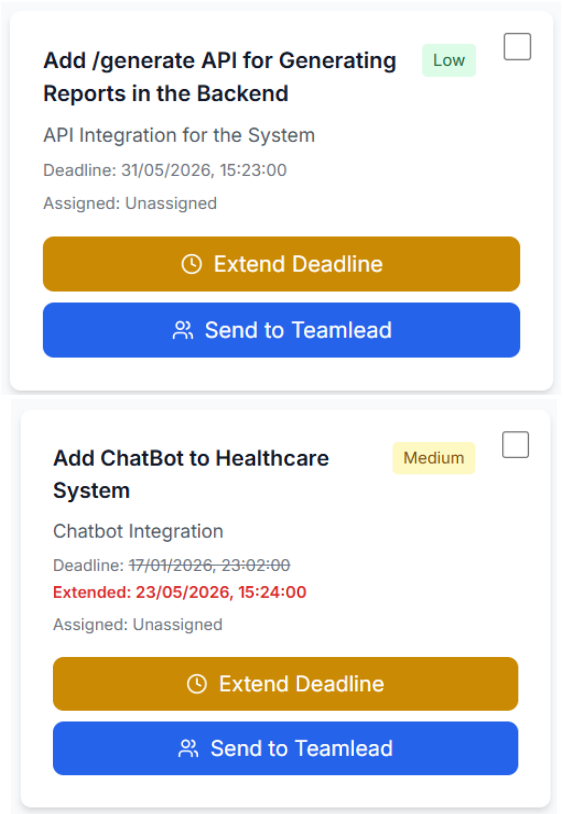


Fig. (2) Prioritized Tasks

Fig. (2) shows the prioritized task management interface of the system. Tasks are automatically categorized into different priority levels such as High, Medium, and Low based on their deadlines. This helps users easily identify urgent tasks and manage work more effectively.

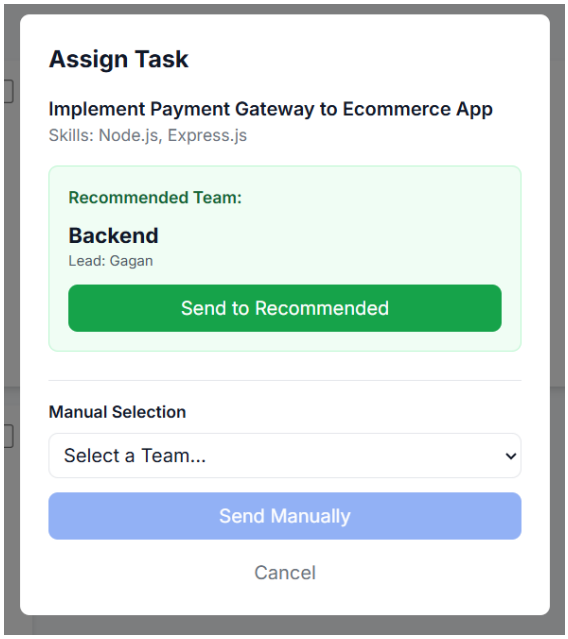


Fig. (3) Automated Task Assignment

Fig. (3) illustrates the automated task assignment feature of the system. Based on the required skills of a task, the system recommends the most suitable

team and allows tasks to be assigned automatically. This improves task distribution and reduces manual effort in selecting team members.

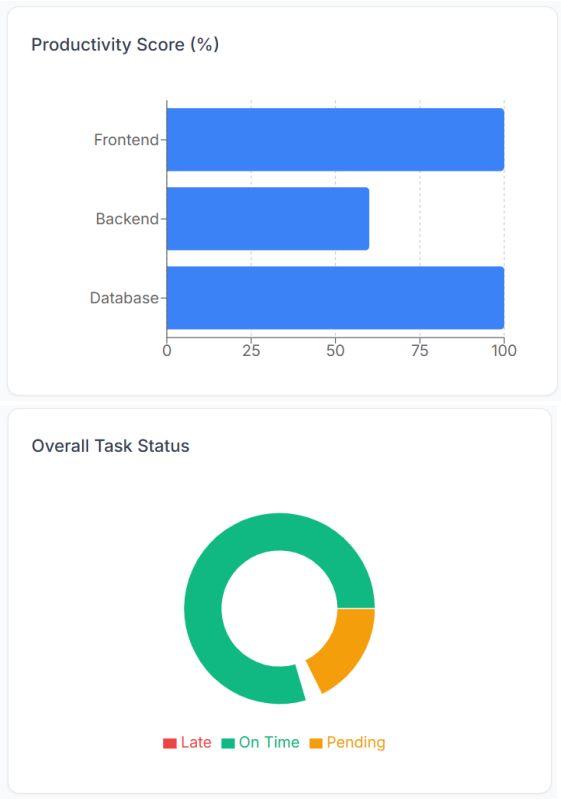


Fig. (4) Performance Overview

Fig. (4) presents the performance overview dashboard generated by the system. The dashboard displays task completion statistics, productivity scores, and overall task status using graphical representations. These reports help in monitoring employee performance and evaluating organizational productivity.

The system produced satisfactory results and successfully performed the intended functionalities. The outputs generated from the system show that it can support efficient and organized task management within an organization.

VII. CONCLUSION

This work resolves all the challenges seen in traditional systems. surely it minimizes human efforts and problems like poor prioritization, manual task assignment and uneven workload distribution are handled easily. This centralized platform improves the coordination and communication among users helps them to do work even better. Core features such as real time task prioritization, automatic task assignment allows distribution of work more efficiently and important tasks are completed on time. The features like report generation adds extra value to the system that help Managers to evaluate the performance of all employees in the organization. Finally, the system has produced the results as expected overall it improves task visibility and accountability and provides an efficient approach for modern organizational task management.

REFERENCES

[1] M. Musthafa and A. Muthukumaran, “Task Management System Using AI Prioritizations,” International Journal of Engineering Research & Technology (IJERT), vol. 13, no. 4, pp. 1–5, Apr. 2024.

[2] S. P. R. Sarveshsri and R. Reshma, “A Task Flow-Based Framework for Employee Task Management,” International Journal of Scientific Research in Engineering and Management (IJSREM), vol. 9, no. 10, pp. 1–6, Oct. 2025.

[3] S. Singh, “Optimizing Productivity: An In-Depth Analysis of Task Management Systems,” International Journal of Innovative Research in Technology (IJIRT), vol. 11, no. 1, pp. 1–5, June 2024.

[4] Y. Bugayenko, M. Farina, A. Kruglov, W. Pedrycz, Y. Plaksin, and G. Succi, “Automatically Prioritizing Tasks in Software Development,” IEEE Access, vol. 11, pp. 90322–90330, 2023.

[5] S. H. A. M. Melhem, “The Impact of Task Prioritization on Employees' Performance,” M.A. thesis, Dept. of Business Administration, Altınbaş University, Istanbul, Turkey, July 2022.

[6] T. Abid, M. Shareef, A. Jamal, and M. Ansar, “Smart Task Scheduler Using AI,” International Journal of Research Publication and Reviews, vol. 6, no. 5, pp. 13377–13380, May 2025.

[7] P. Vishaliney, C. S. Pemasiri, and M. Kanthakumar, “AI-Driven To-Do List: Optimizing Task Categorization and Prioritization Using Ensemble Models,” International Journal of Advanced Computer Science and Applications, vol. 12, no. 3, pp. 1–8, 2024.

[8] K. Shetty and S. K. Kini, “Employee Task Management System,” Journal of Emerging Technologies and Innovative Research (JETIR), vol. 12, no. 7, pp. 1–6, July 2025.

[9] A. Sukstrienwong and S. Pukdesree, "Task Assignment System for Remote Work by Skill Matching: Integration of Web and Mobile Technologies," International Journal of Information Systems and Technology, vol. 10, no. 2, pp. 1–7, 2024.

[10] Y. Bugayenko, A. Bakare, A. Cheverda, M. Farina, A. Kruglov, Y. Plaksin, W. Pedrycz, and G. Succi, "Prioritizing Tasks in Software Development: A Systematic Literature Review," IEEE Access, vol. 12, no. 1, pp. 1–15, 2024.

[11] S. Gopalakrishnan and D. Borrajo, "Assignment and Prioritization of Tasks with Uncertain Durations for Satisfying Makespans in Decentralized Execution," International Journal of Artificial Intelligence Research, vol. 8, no. 2, pp. 1–10, 2023.

[12] R. Jain, M. Merchant, N. Gowalker, and R. Gupta, "A New Paradigm in Employee Evaluation: Leveraging Project Management Insights for Promotions and Rewards," International Journal of Future Management Research (IJFMR), vol. 7, no. 1, pp. 1–8, Jan.–Feb. 2025.

[13] C. Kandemir and H. A. H. Handley, "Employee-Task Assignments for Organization Modeling: A Review of Models and Applications," in Advances in Organizational Studies, S. Long, Ed., vol. 5, no. 1, pp. 1–12, 2014.

[14] S. Deviprasad, N. Madhumithaa, I. W. Vikas, A. Yadav, and G. Manoharan, "The Machine Learning-Based Task Automation Framework for Human Resource Management in MNC Companies," Engineering Proceedings, vol. 59, no. 1, pp. 63–70, Dec. 2023. doi: 10.3390/engproc2023059063.

[15] N. K. Ugwumba and P. S. Jaja, "Enhanced Task Prioritization System Using Deep-Q-Network Model," International Journal of Computer Science Engineering Techniques, vol. 9, no. 6, pp. 1–7, Nov.–Dec. 2025.

[16] G. Garg and U. Dikshit, "Intelligent Employee Task Scheduler," International Journal of Computer Applications, vol. 183, no. 8, pp. 1–6, June 2021.

[17] S. S. Bafana, T. D. Patil, P. G. Shete, S. S. Patil, and V. R. Patil, "Employee Task Management System," International Journal of Advance Research and Innovative Ideas in Education (IJARIIE), vol. 11, no. 2, pp. 1–5, 2025.

[18] J. Pechová, H. Volfová, and A. Jírová, "Impact of Task Assignment on Effectiveness in Work Teams," Journal of Entrepreneurship and Sustainability Issues, vol. 10, no. 4, pp. 1–12, June 2023. doi: 10.9770/jesi.2023.10.4(10).

[19] V. Druzhynin, M. Gladka, I. Borysenko, Y. Hladkyi, and R. Lisnevskyi, "A Model for Allocating Labor Resources to Project Work Based on Task Prioritization," International Journal of Project Management Research, vol. 9, no. 3, pp. 1–10, 2024.

[20] S. Kumar, M. Sagar, M. Bisht, and S. Aggrawala, "Workforce Task Automation Using Spring Boot, JIRA," International Journal of Creative Research Thoughts (IJCRT), vol. 14, no. 2, pp. 1–6, Feb. 2026.