iJRASET
International Journal For Research in Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 14 **Issue:** III **Month of publication:** March 2026

DOI: https://doi.org/10.22214/ijraset.2026.78767

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 14 Issue III Mar 2026- Available at www.ijraset.com

Employee Task Management System Website Using REACT.TS

K. Asha Devi[1], K. Harika[2], K. Barathi[3], A. Eswar vivek[4], R. Manikanta[5]
[1]*Assistant professor, Department of Computer science and Engineering (Artificial intelligence & Machine Learning), Avanthi institute of engineering of technology, India*
[2, 3, 4, 5]*UG, Department of Computer science and Engineering (Artificial intelligence &Machine Learning), Avanthi institute of engineering of technology, India*

Abstract: ***The management of employee tasks in many organizations is often performed manually or through disconnected systems, which can lead to poor task tracking, communication gaps, and reduced productivity. This project proposes an Employee Task Management System, a web-based platform designed to help organizations efficiently assign, monitor, and manage employee tasks.***
The system provides a centralized environment where administrators can create tasks, assign them to employees, track progress, and manage task-related information. The platform is developed using modern web technologies including React with TypeScript for the frontend and Node.js with Express.js for the backend, ensuring a responsive and scalable system. Authentication is secured using JSON Web Tokens (JWT), and sensitive data such as passwords are protected using bcrypt encryption.
The system also supports file uploads through Multer, allowing employees to attach documents related to tasks. By organizing task workflows in a structured digital environment, the system improves productivity, transparency, and efficient task monitoring within organizations.
Keywords: Employee Task Management System, Task Tracking, Work Assignment Platform, Secure Authentication, Productivity Management

I. INTRODUCTION

In modern organizations, managing employee tasks efficiently is essential for maintaining productivity and ensuring timely completion of projects.
Traditional task management methods such as manual tracking, emails, and spreadsheets often create confusion and make it difficult for managers to monitor employee performance and task progress.
To address these challenges, the Employee Task Management System is developed as a centralized web-based application that allows organizations to manage employee tasks efficiently. The system enables administrators or managers to assign tasks to employees, monitor their progress, and manage task-related information through an interactive dashboard.
The platform is built using React with TypeScript, providing a dynamic and responsive user interface. The backend is implemented using Node.js and Express.js, ensuring efficient server-side operations and secure data handling. The system also includes secure authentication using JWT tokens and password encryption through bcrypt, ensuring safe access control for users. By digitizing the task management workflow, the system improves communication, enhances transparency, and increases overall organizational efficiency.

II. LITERATURE SURVEY

Early task management systems were primarily based on manual methods such as spreadsheets, emails, and paper-based task allocation. These methods often led to inefficiencies, lack of transparency, and difficulty in tracking employee performance.
With the advancement of digital technologies, web-based task management systems were introduced to automate task allocation and tracking. These systems allowed organizations to store task information in databases and provide dashboards for monitoring progress. Modern task management platforms now use advanced web technologies such as React.js, Node.js, and cloud databases to build scalable and interactive systems. These technologies enable real-time task updates, improved collaboration, and efficient workflow management.

©IJRASET: All Rights are Reserved

Security has also become an important aspect of modern web applications. Authentication mechanisms such as JSON Web Tokens (JWT) and password encryption using bcrypt are widely used to protect user accounts and sensitive information. Additionally, modern UI frameworks such as Tailwind CSS help developers build responsive and user-friendly interfaces that improve user experience. These advancements highlight the importance of integrating modern web technologies to develop efficient employee task management systems.

III. PROBLEM STATEMENT

Traditional employee task management methods rely heavily on manual coordination, which can lead to inefficient task tracking, miscommunication, and difficulty in monitoring employee performance. Managers often struggle to keep track of task assignments, deadlines, and progress using spreadsheets or email communication. These limitations create challenges in maintaining transparency, accountability, and productivity within organizations.

Therefore, there is a need for a centralized system that allows efficient task assignment, progress tracking, and secure management of employee activities.

The proposed Employee Task Management System addresses these challenges by providing a digital platform where managers can assign tasks, employees can update their progress, and both parties can track task status in real time.

IV. SYSTEM ARCHITECTURE

The proposed system architecture consists of a frontend, backend, and database layer that work together to manage employee tasks efficiently. The frontend is developed using React with TypeScript and Tailwind CSS, providing a responsive user interface where administrators and employees can interact with the system. Axios is used to communicate with the backend API, and React Router manages navigation between pages.

The backend is developed using Node.js and Express.js, which handle user authentication, task management, and data processing. Security is implemented using JWT for authentication and bcrypt for password hashing. File upload functionality is supported using Multer.

The database layer uses SQL to store user details, task information, and related records securely. This architecture ensures efficient data processing, secure authentication, and smooth communication between system components.

V. METHODOLOGY

1) User Registration and Authentication: The system begins with a secure user authentication process. Employees and administrators can register and log in to the platform. Passwords are securely encrypted using bcrypt, and authenticated sessions are maintained using JSON Web Tokens (JWT). This ensures that only authorized users can access the system.

2) Task Creation and Assignment: Administrators or managers can create tasks through a structured interface. Each task includes details such as task title, description, priority level, and deadline. Once created, tasks can be assigned to specific employees within the organization.

3) Task Tracking and Management: Employees can view tasks assigned to them through their dashboard. They can update task progress, upload related documents, and mark tasks as completed. The system records all updates and displays them in real time for managers.

4) File Upload and Documentation: The system supports file uploads using Multer, allowing employees to attach documents or reports related to assigned tasks. This feature helps maintain proper documentation of work progress.

VI. TECHNOLOGY STACK

The system is developed using React with TypeScript for the frontend to create a dynamic and responsive user interface. Tailwind CSS is used for styling the application and improving the visual design. Axios is used to handle API communication between the frontend and backend, while React Router manages navigation between application pages.

The backend is implemented using Node.js and Express.js, which handle server-side operations and API requests. Authentication and authorization are secured using JSON Web Tokens (JWT), while bcrypt.js is used for password encryption. The system also uses Multer to manage file uploads.

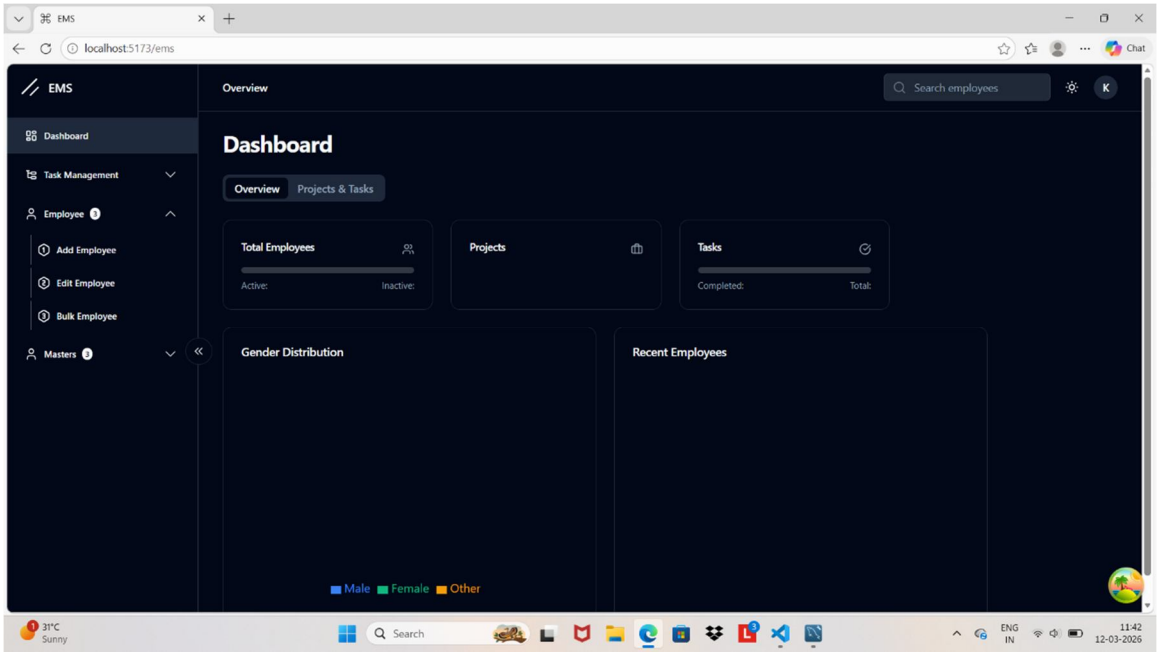
The database used in the system is SQL, which stores user data, task information, and related records in a structured format.

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 14 Issue III Mar 2026- Available at www.ijraset.com

VII. INPUT & RESULT

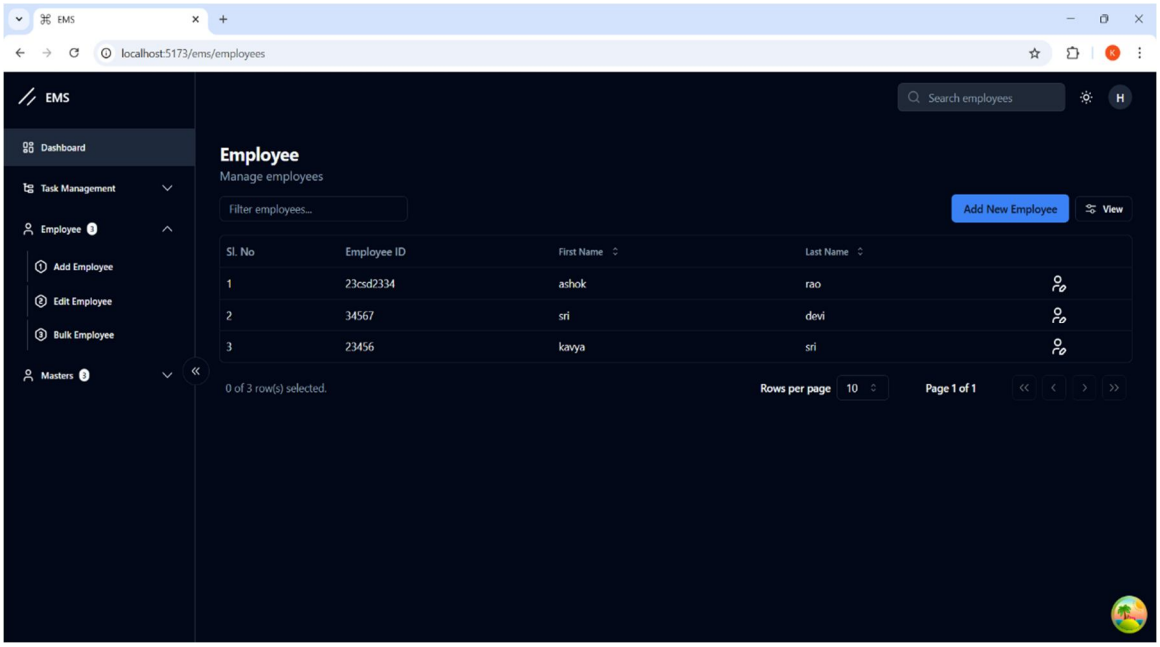
A. Input

1) Admin Dashboard



The system provides an administrator dashboard where managers can create, assign, and manage tasks for employees. Administrators can view all employees, assign tasks with deadlines, and monitor the progress of each task.

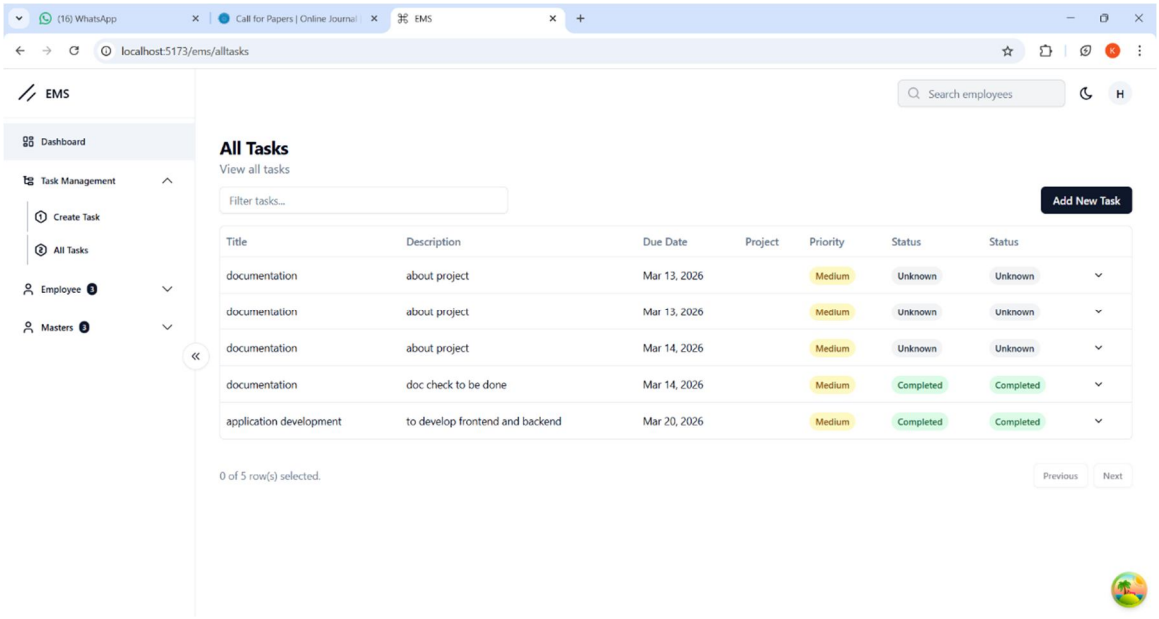
2) Employee Dashboard



Employees can log in to their dashboard to view tasks assigned to them. They can update task status, upload related files, and track deadlines to ensure timely completion.

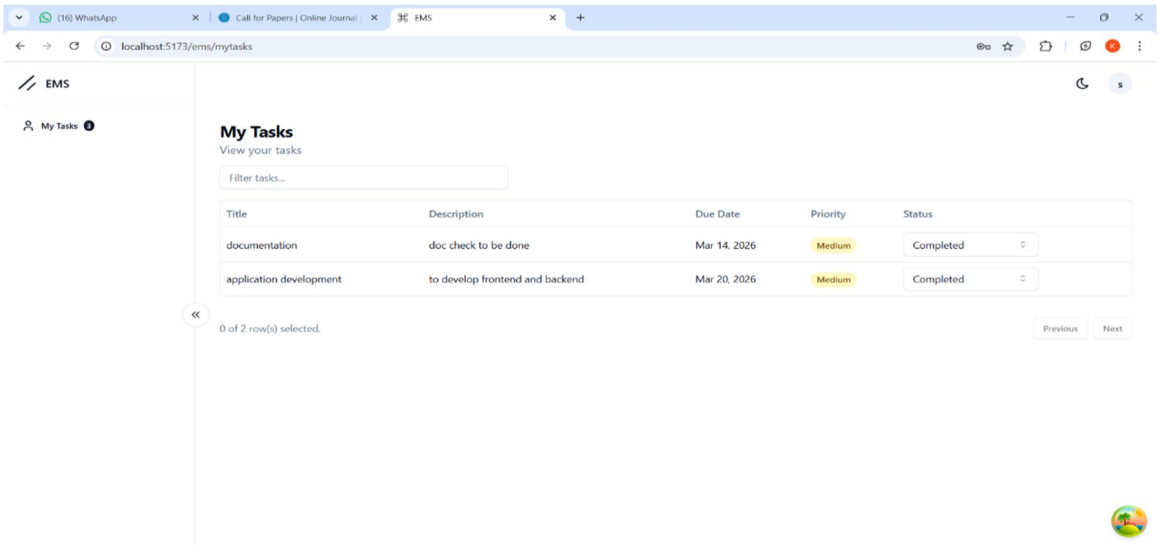
B. Result

1) Admin Result



Administrators can view task progress and monitor employee performance through a structured dashboard. They can check which tasks are completed, pending, or in progress.

2) Emplyoee Result



Employees can easily track their assigned tasks and update their work progress. The system helps employees stay organized and manage their responsibilities efficiently.

VIII. ADVANTAGES

The Employee Task Management System offers several advantages for organizations. It improves task organization by providing a centralized platform for task assignment and tracking. The system reduces manual effort by automating task management processes and improves communication between managers and employees. The secure authentication system ensures safe access to the platform, while the user-friendly interface makes it easy for employees to interact with the system. Overall, the system enhances productivity, transparency, and workflow efficiency.

©IJRASET: All Rights are Reserved | SJ Impact Factor 7.538 | ISRA Journal Impact Factor 7.894 |

IX. FUTURE SCOPE

The system can be further improved by integrating additional features such as real-time notifications, team collaboration tools, and performance analytics. Mobile application support can also be added to allow employees to manage tasks from their smartphones. Artificial Intelligence techniques can also be incorporated to analyze employee productivity and provide smart task recommendations. These enhancements can further improve efficiency and usability.

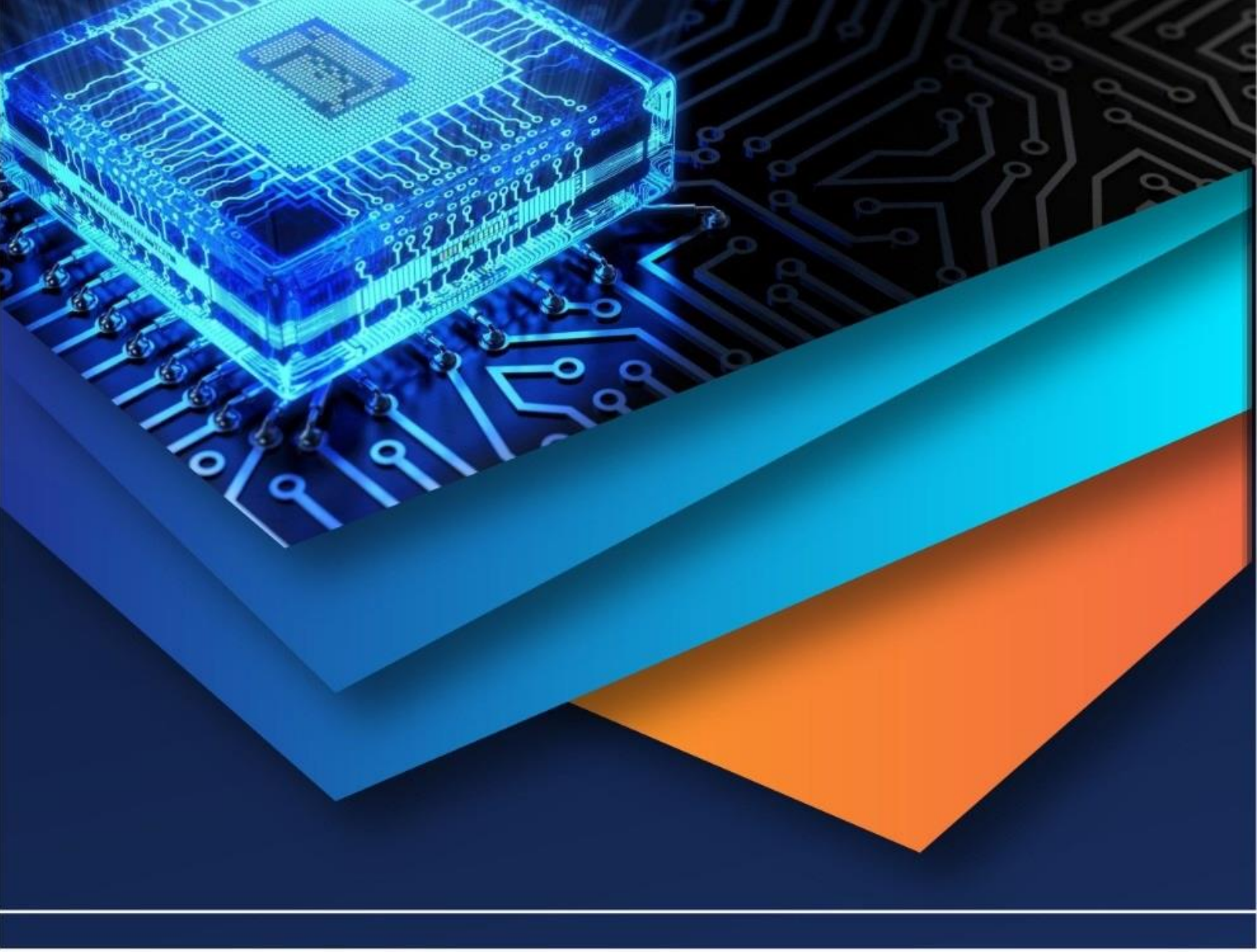
X. CONCLUSION

The Employee Task Management System was successfully developed to provide an efficient platform for managing employee tasks within an organization. The system allows managers to assign tasks, monitor progress, and maintain organized workflows.
By using modern technologies such as React, Node.js, Express.js, and SQL, the platform ensures scalability, security, and efficient performance. The system improves productivity, enhances transparency, and simplifies task management processes in organizations.

REFERENCES

[1] D. Flanagan, JavaScript: The Definitive Guide, 7th ed., O'Reilly Media, 2020.
[2] React Development Team, "React – A JavaScript Library for Building User Interfaces," Official Documentation, 2024.
[3] Node.js Foundation, "Node.js: JavaScript Runtime Environment," Technical Documentation, 2024.
[4] Express.js Community, "Express.js Web Application Framework for Node.js," Official Documentation, 2024.
[5] A. Banks and E. Porcello, Learning React: Modern Patterns for Developing React Apps, O'Reilly Media, 2022.
[6] Auth0, "JSON Web Token (JWT) Introduction and Authentication Guide," Technical Documentation, 2023.
[7] OpenJS Foundation, "bcrypt.js Library for Secure Password Hashing," Official Documentation, 2023.
[8] Tailwind Labs, "Tailwind CSS – Utility-First CSS Framework," Official Documentation, 2024.
[9] Axios Contributors, "Axios: Promise-based HTTP Client for the Browser and Node.js," GitHub Documentation, 2024.
[10] SQL Documentation, "Structured Query Language for Data Management," Database Systems Documentation, 2024.

©IJRASET: All Rights are Reserved | SJ Impact Factor 7.538 | ISRA Journal Impact Factor 7.894 |



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Call : 08813907089 (24*7 Support on Whatsapp)