Task Management Application

Om D. Tandulkar[1], Shubham S. Wattamwar[2], and Piyush Patil[3], Adarsh Jagdambe[4], Tanmayee Kute[5]
[1,2,3,4] *PES Modern College of engineering*
[5]*Prof.Tanmayee Kute/ PES Modern College of engineering*

Abstract— This research presents the development of a task management application for TNT Infra Limited, leveraging TypeScript and the Ionic framework to create a hybrid solution. The application addresses task allocation and monitoring, enabling HR, managers, and directors to assign tasks efficiently across various departments and sites. Key features include task prioritization, deadline management, real-time photo-based progress tracking, and automated notifications. This paper outlines the technical architecture, development methodology, and observed benefits of implementing the application.

Index Terms— Task management, TypeScript, Ionic framework, project tracking, hybrid application.

I. INTRODUCTION

Effective task management is vital for the success of large infrastructure organizations like TNT Infra Limited, which oversee complex projects across multiple departments and sites. Traditional task allocation methods often lead to communication gaps, delays, and inefficient task tracking, affecting productivity and project timelines.

Recognizing these challenges, TNT Infra Limited sought a solution to centralize task management. This paper discusses the development of a task management application built using TypeScript and the Ionic framework for cross-platform compatibility. The app streamlines task assignment, deadline setting, and real-time progress updates with photo submissions. Automated notifications ensure employees are informed and adhere to deadlines.

This research details the app's technical architecture, key features, and performance, highlighting its effectiveness in improving task workflows and operational efficiency across the organization.

II. MATH

 Task Scheduling Efficiency: To represent the goal of minimizing total task allocation time with priority consideration:

Minimize: ∑i=1nTi·Pi\text{Minimize: } \sum_{i=1}^{n} T_i \cdot P_iMinimize: i=1∑nTi·Pi

Where:

- TiT_iTi = time required for task iii
- PiP_iPi = priority weight of task iii

 Notification Impact on Timeliness: To show the probability of tasks being completed on time:

P(On Time)=Number of tasks completed on timeTotal number of tasksP(\text{On Time}) = \frac{\text{Number of tasks completed on time}}{\text{Total number of tasks}}P(On Time)=Total number of tasksNumber of tasks completed on time

 Efficiency Improvement: To express the percentage of task assignment efficiency improvement:

Efficiency Improvement (%)=Time before app−Time after appTime before app×100\text{Efficiency Improvement (\%)} = \frac{\text{Time before app} - \text{Time after app}}{\text{Time before app}} \times 100Efficiency Improvement (%)=Time before appTime before app−Time after app.

III. OBJECTIVES

- To design a TMA that supports task handling within various departments.
- To outline the development process using TypeScript and the Ionic Framework.
- To develop UML diagrams and graphs reflecting the system architecture and processes.
- To facilitate notifications and progress tracking, enhancing team performance.

IV. TECHNOLOGY STACK OVERVIEW

1 TypeScript

TypeScript, a superset of JavaScript, offers optional static typing and enhances code quality and maintainability. It enables developers to detect errors early in the development lifecycle, making it ideal for large-scale applications, such as task management systems.

2 Ionic Framework

The Ionic framework is an open-source UI toolkit for building high-quality mobile applications using web technologies like HTML, CSS, and JavaScript. The framework allows developers to create hybrid apps that run on multiple platforms (iOS, Android, and web) with a single codebase. Its compatibility with TypeScript ensures that the application leverages the benefits of both technologies.

3 MERN Stack

The MERN stack encompasses:

- MongoDB: A NoSQL database that stores data in flexible, JSON-like documents, suitable for dynamic task management data.
- Express.js: A web application framework for Node.js that facilitates building RESTful APIs..
- Node.js: A JavaScript runtime that allows server-side programming, essential for handling client requests and application logic.

V. APPLICATION ARCHITECTURE

This survey describes the application's architecture using a layered approach, consisting of a front-end client, back-end server, and database.

1 Front-End Client

Developed with React, the front end presents a user-friendly interface for HR managers to assign tasks, view progress, and manage priorities. The Ionic framework enhances the mobile experience, ensuring that managers can manage tasks on the go.

2 Back-End Server

The back end, constructed with Node.js and Express.js, manages the API endpoints for task creation, assignment, and progress tracking. It interacts with MongoDB to perform CRUD operations on task data.

3 Database

MongoDB serves as the primary data store, enabling the application to manage task details, user information, and project timelines effectively .

VI. KEY FEATURES

1 Task Assignment

HR and project managers can assign tasks to individuals or groups, specifying details such as priority, deadlines, and associated resources.

2 Progress Tracking

The application includes real-time tracking of task progress. Employees can update their status, upload images related to their tasks, and add comments, providing a transparent view of ongoing work.

3 Priority Management

Tasks can be organized based on priority levels, allowing managers to focus resources on high-priority tasks while balancing workloads across departments.

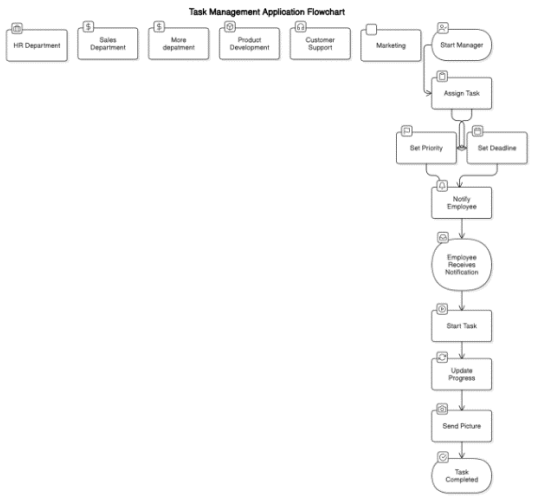
4 Deadline Monitoring

The system monitors deadlines and sends notifications to remind employees of impending due dates, ensuring projects adhere to schedules.

5 Reporting and Analytics

Comprehensive reporting features provide insights into task completion rates, project timelines, and resource allocation, enabling data-driven decision-making.

VIII. DATA FLOW DIAGRAMS



VII. CONCLUSION

Task management applications built using TypeScript, Ionic framework, and the MERN stack represent a powerful solution for modern organizational challenges. By facilitating clear task assignment, progress tracking, and effective priority management, these applications empower HR managers and project leads to optimize productivity and resource utilization.

VIII. REFERENCES

[1] Optimization of Resource Allocation and Task Allocation with Project Management Information Systems in Information Technology Companies Ilham Nur Pratama, Muhammad Dachyar University of Indonesia, 2023
[2] Project management using IT applications PROF. MUKESH BARAPATRE 2021
[3] Online Task Management System (OTMS) GRISHMA HEDAOO, 2018
[4] A study on task management system Parkavi A., N S Jyothii, 2016
[5] Adaptive case management: overview and research challengess HR Motahari-Nezhad 2013
[6] Soft-WSN: Software-defined WSN management system for IoT applications S Beraa, S Misraa 2016
[7] Efficient task offloading for IoT-based applications in fog computing using ant colony optimization MK Hussein, MH Mousa 2020
[8] A framework for native multi-tenancy application development and management CJ Guo, W Sun 2019
[9] An online reinforcement learning-based energy management strategy for microgrids with centralized control F Luo, Z Wang 2024