A Major Project Synopsis on

Employees Task Management Tool

Submitted to Manipal University, Jaipur

Towards the partial fulfillment for the Award of the Degree of

MASTER OF COMPUTER APPLICATIONS

2023-2025

by

Harsh Gupta

23FS20MCA00002

MANIPAL UNIVERSITY
JAIPUR

Under the guidance of

Dr. Pradeep Kumar

Department of Computer Applications

School of AIML, IoT&IS, CCE, DS and Computer Applications

Faculty of Science, Technology and Architecture

Manipal University Jaipur

Jaipur, Rajasthan

2025

Employee Task Management System

I. Introduction

The Employee Task Management System is a web-based application designed to simplify task assignment, tracking, and completion for employees within an organization. The system ensures efficient data management, user authentication, and seamless API communication, providing a secure and scalable solution.

The backend is developed using Node.js and SQL, ensuring smooth handling of data, authentication, and real-time updates. This system enhances task management, time tracking, and performance monitoring, improving overall productivity.

II. Motivation

The primary motivation for this system is to address the challenges of manual task management, such as miscommunication, tracking inefficiencies, and time-consuming processes. The system benefits both administrators and employees by:

For Administrators:
• Automating task assignments to employees.
• Tracking task progress in real time.
• Generating performance reports efficiently.
• Reducing manual workload and administrative overhead.

For Employees:
• Simplified task tracking and updates.
• Automated time logging for tasks.
• Easy communication through comments and feedback.
• Receiving notifications and reminders for deadlines.

III. Problem Statement

For Administrators:
• Streamlined task assignment without manual intervention.

• Automated tracking and reporting.
• Instant notifications and updates.
• Reduced reliance on external tracking tools.

For Employees:
• Clear visibility of assigned tasks and their deadlines.
• Automated time tracking to manage workloads efficiently.
• A centralized system for feedback and communication.
• Improved productivity and better work management.

IV. Methodology & Work Plan

1. API Development (Backend – Node.js & SQL):

• User Authentication & Authorization
  - JWT-based authentication for admins and employees.
  - Role-based access control for secure access.
• Task Management API:
  - CRUD operations: Create, Read, Update, Delete tasks.
  - Assign tasks dynamically to employees.
  - Track task progress and status.
• Time Tracking System:
  - Employees can log work hours per task.
  - The system calculates remaining time for each task.
• Comments & Feedback API:
  - Employees can add comments to tasks.
  - Managers can review and provide feedback.
• Database Management:
  - Efficient storage and retrieval of task and user data.
  - Optimized SQL queries for better system performance.

2. API Endpoints:

• User Routes: /login, /profile
• Task Routes: /tasks, /tasks/:id (CRUD operations)
• Comment Routes: /tasks/:id/comments
• Time Tracking Routes: /tasks/:id/time-tracking

V. Requirements for Proposed Work

1. Software Requirements:

- Operating System: Windows / Linux
- Backend Framework: Node.js (Express.js)
- Database: SQL (PostgreSQL / MySQL)
- Authentication: JWT (JSON Web Token)
- API Testing Tools: Postman, Swagger
- Deployment: Docker, AWS/Digital Ocean
- Version Control: Git & GitHub

2. Hardware Requirements:

- Processor: Minimum Pentium P4
- RAM: Minimum 256MB
- Storage: Minimum 10GB Hard Disk Space

VI. Bibliography & References

- Node.js Official Docs: https://nodejs.org/docs/latest/api/
- Express.js Guide: https://expressjs.com/
- JWT Authentication: https://jwt.io/
- PostgreSQL Documentation: https://www.postgresql.org/docs/
- MySQL Guide: https://www.w3schools.com/nodejs/nodejs_mysql.asp