



MANIPAL UNIVERSITY  
JAIPUR

# **Major Project**

## **Project - Synopsis**

MCA - IV Sem

Subject Code: CA7270

Submitted By

Student Name-Rupa Kumari

Student Registration-23FS20MCA00062

Faculty Coordinator

Dr. Amit Hirawat, Associate Professor

**DEPARTMENT OF COMPUTER APPLICATIONS**

## Title of the Project:

Employee Task Management System

## Team Details:

- **Team Members:**
  - Rupa Kumari - Frontend Development

## Introduction:

The **Employee Task Management System** is a web-based application designed to help organizations efficiently manage and assign tasks to employees. The system provides an intuitive **Admin Dashboard** where managers can assign, track, and update tasks. Employees can view their assigned tasks, update progress, and track time spent on each task. The system aims to streamline workflow, enhance productivity, and ensure transparency in task tracking.

## Objectives:

### System Features:

#### Admin Dashboard:

- Assign tasks to employees.
- View and update task progress.
- Track estimated and actual task completion time.
- Monitor overall employee performance.

#### Employee Dashboard:

- View assigned tasks with start and end dates.
- Update task status (In Progress, CompTo develop a **responsive frontend interface** for the **Admin and Employee Dashboards**.
- To enable **real-time task tracking and assignment**.
- To provide **employee-wise task segregation** for better monitoring.
- To implement **search, filtering, and task status updates**.

- To ensure **time tracking and reporting features** for employees.
- **leted, etc.).**
- Log time spent on each task.
- Add comments and feedback on tasks.

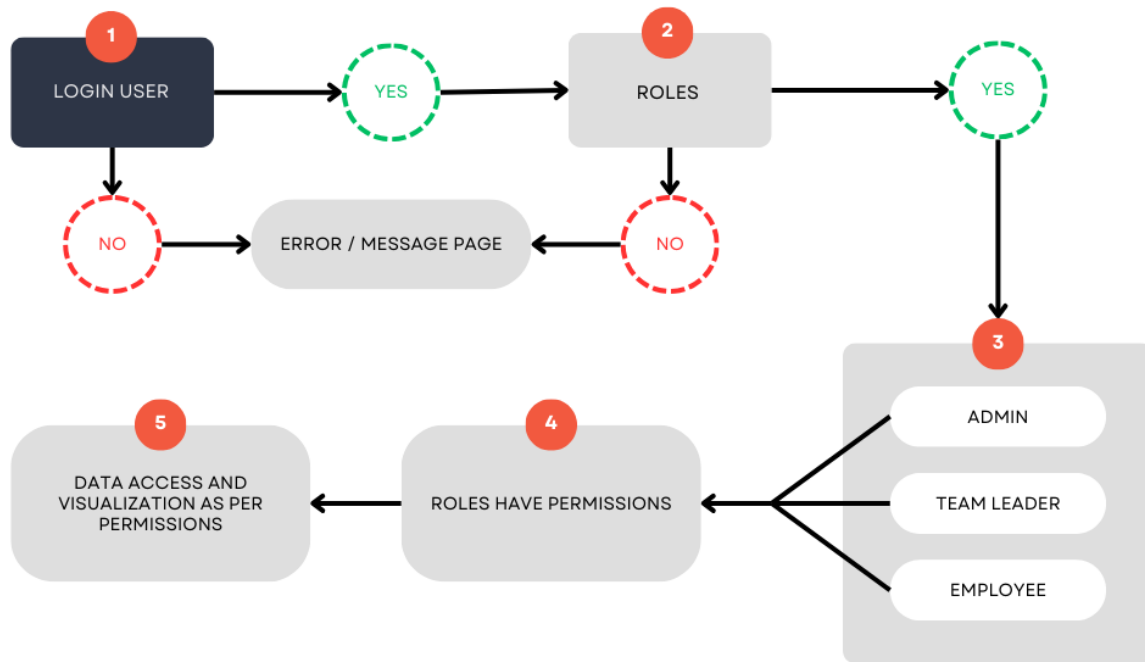
### **DFD (Data Flow Diagram):**

The Data Flow Diagram outlines the flow of information:

1. **User Interactions:** Employees and Managers login and perform actions.
2. **Task Assignment:** Managers create and assign tasks with deadlines.
3. **Task Updates:** Employees update task progress, tracking time spent.
4. **Notifications & Reports:** System generates alerts and task performance reports.

## Data Flow Diagram

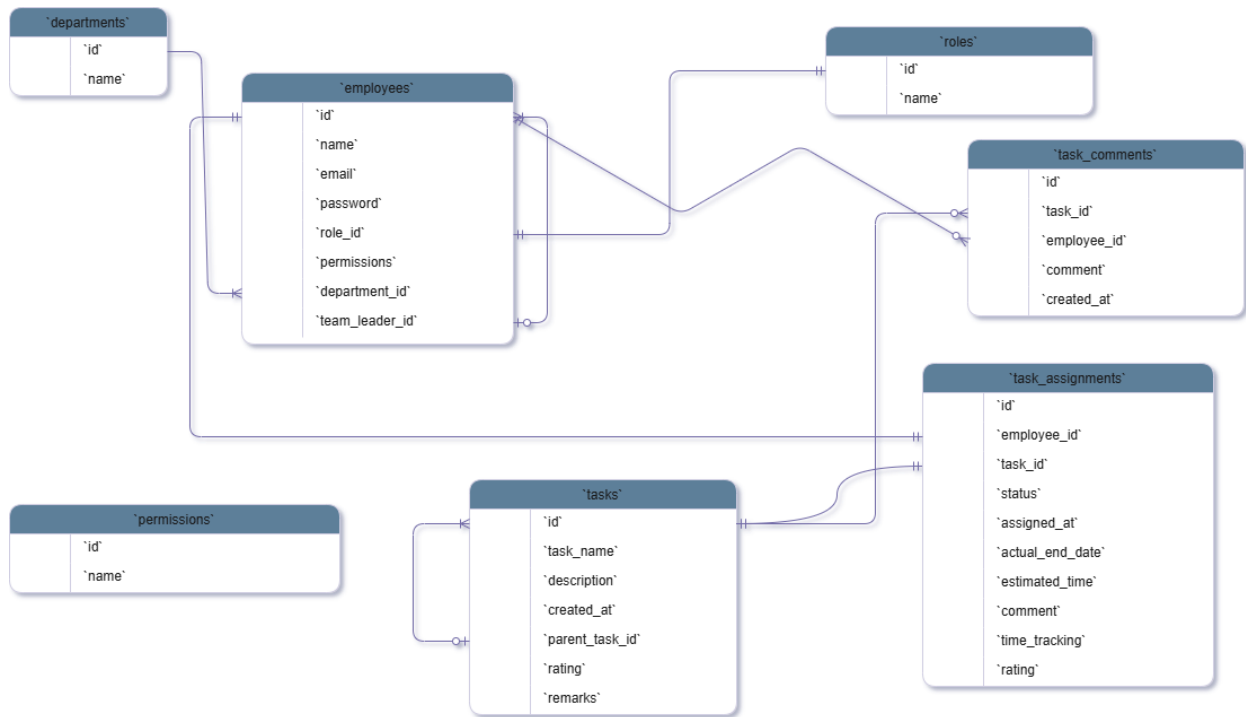
### Flowchart



### ER Diagram:

The Entity-Relationship diagram includes the following entities:

- **User** (\_id, name, email, password, role, createdAt, updatedAt)
- **Task** (\_id, title, description, assigned\_to, status, start\_date, end\_date, estimate\_time, actual\_end\_date, createdAt, updatedAt)
- **Comment** (\_id, task\_id, user\_id, message, createdAt, updatedAt)
- **TimeTracking** (\_id, task\_id, user\_id, time\_spent, time\_remaining, createdAt, updatedAt)
- **Notification** (\_id, user\_id, message, createdAt, updatedAt)



## Project Timeline:

- **Phase 1:** Planning & Design - January 1 to January 20
- **Phase 2:** Frontend Development - January 21 to March 10
- **Phase 3:** Backend Development - March 11 to April 15
- **Phase 4:** Testing & Deployment - April 16 to May 5

## Tools / Platform, Hardware and Software Requirement Specifications:

- **Languages:** Node.js, JavaScript, HTML, CSS
- **Frameworks:** Express (Backend), React (Frontend)
- **Database:** MySQL
- **Development Tools:** Visual Studio Code, Postman, Git
- **Hardware Requirements:** Minimum 8GB RAM, Dual-core processor
- **Software Requirements:** Windows/Linux/Mac OS, Node.js, npm

## References:

- <https://www.npmjs.com/package/react>
- <https://mui.com/material-ui/react-table/?srsltid=AfmBOoqvHDhrlZJ4J6hkPO4zEkfbigLLkM4G1LD4IVSK50XklmmQXQo->
- <https://nextjs.org/docs/pages/building-your-application/routing/dynamic-routes>
- <https://nextjs.org/docs/pages/building-your-application/data-fetching/client-side>