

# MERN Final Evaluation – Sales CRM (JUN-2025)

## Problem Statement

Build a Sales CRM Dashboard application where:

- Admins can monitor leads, employee performance, and daily activities.
- Sales employees can manage their assigned leads, schedule calls, and maintain attendance.
- Real-time insights are provided via analytics and visual data representations.

## 1. Dashboard Overview (Admin)

- Quick Stats Cards:
  - Unassigned Leads
  - Leads Assigned This Week
  - Active Salespeople
  - Conversion Rate (%)
- Sales Analytics Graph:
  - Daily Conversion Rates (bar chart)
  - Interactive Hover Stats

## 2. Attendance & Check-in System

- Check-in/Check-out Tracking:
  - Daily timestamps
  - Visual indicators (Green/Red)
- Break Log:
  - Start/end time for breaks (date-wise view)
  - Optional for Employee Dashboard only

## 3. Activity Overview

- Recent Activity Feed:
  - Timestamped logs (e.g., "Jay closed a deal — 2 hours ago")
  - Assignments and Closures

## 4. Lead Management (Admin)

- Lead Table View:
  - Columns: Name, Date, Assigned, Unassigned, Closed
  - Search & Sort functionality
- Add Leads:
  - Manual Entry
  - CSV Upload (with Drag & Drop + progress & verification)
  - Provide sample CSV
- Lead Distribution:
  - Equal distribution
  - Based on location/language preferences
- Bulk Actions:
  - Bulk delete employee → prompt for redistribution
  - Bulk lead assignment

## 5. Lead Details (Extendable)

- Lead Cards:
  - Contact name, email, lead date
  - Status: Ongoing, Closed
  - Lead Type Tags: Hot / Warm / Cold (Color-coded)
- Status Rules:
  - Cannot close a lead with scheduled calls

## 6. Scheduled Calls List

- Fields:
  - Lead Name, Contact
  - Call Type (Referral, Cold, etc.)
- Filters:
  - Today, Upcoming, All

## 7. Employee Management

- Table View:
  - Name, Email, Emp. ID, Status
  - Assigned & Closed Leads
- Add/Edit/Delete Employee:
  - Fields: Name, Email, Location, Language

- Auto lead assignment based on preferences
- Reassignment prompts on delete

## **8. User Profile Management**

- Editable Fields:
  - First/Last Name
  - Email
  - Location
  - Preferred Language
  - (Employee Dashboard: Password + Confirmation)

## **9. Settings**

- Lead Assignment Rules
- Admin Controls:
  - Delete employees
  - Confirm uploads

## **10. Navigation UI**

- Top Bar:
  - Global search (leads/employees)
- Bottom Bar (Mobile First):
  - Home → Dashboard
  - Leads → View/Add/Manage
  - Employees → Team Management
  - Schedule → Call Filters
  - Settings → Admin Config
  - Profile → Logout

## **Employee Dashboard (Limited Permissions)**

### **1. Attendance Management**

- Check-in / Check-out Tracking
- Visual Indicators
- Break Logs per day

### **2. Activity Overview**

- Recent Activity Feed:
  - Leads Assigned, Deals Closed
  - Timestamped (e.g., “1 hour ago”)

### **3. Lead Management**

- Lead Cards:
  - Status: Ongoing / Closed
  - Type: Hot / Warm / Cold
- Edit Options
- Status Restrictions (Scheduled Call Rule)
- Scheduled Calls List with filters

### **4. Profile Management**

- Editable Fields:
  - Name, Email, Password, Location

### **5. Bottom Navigation**

- Home → Dashboard + Attendance
- Leads → Management Interface
- Schedule → Filter & View Calls
- Profile → Manage Settings

## **Tech Stack**

- Frontend: React + Vanilla CSS (Modular CSS only)
- Backend: Node.js + Express
- Database: MongoDB
- Deployment: Vercel/Netlify (Frontend), Render/Heroku (Backend)

## **Submission Checklist**

- Pixel-perfect responsive UI (per Figma)
- JWT Auth Implementation

- Passwords encrypted + secure comparison
- Proper error handling (Frontend + Backend)
- Backend search, pagination, and filtering (No frontend search logic)
- CSV upload with verification
- Sample data included
- Hosted frontend/backend with working live links

## **Evaluation Criteria**

- Code quality & folder structure
- UI/UX consistency and mobile responsiveness
- Functional coverage of feature checklist
- Security (JWT, password encryption, role restrictions)
- Edge case handling, validations, error messaging
- Data accuracy & backend-driven logic