Cloud-Based Attendance & Leave Management System

This document describes a fresher-level, production-style MERN + AWS project designed for Indian IT companies. The system manages employee attendance, leave workflows, and reporting using React, Node.js, Supabase, and AWS.

1. Problem Statement

Many small and mid-sized Indian IT companies still use Excel or manual systems to track attendance and leave. This leads to errors, lack of transparency, and poor reporting. The goal is to build a secure, cloud-hosted web application to automate attendance and leave management.

2. High-Level Architecture Overview

Frontend (React) communicates with a Node.js + Express backend hosted on AWS EC2. Supabase is used as the managed PostgreSQL database and authentication provider. AWS S3 stores reports and employee images. IAM roles ensure secure access between EC2 and S3.

3. Technology Stack

Frontend: React, HTML, CSS, Axios

Backend: Node.js, Express.js, JWT

Database & Auth: Supabase (PostgreSQL + Auth)

Cloud: AWS EC2, AWS S3, IAM, NGINX

4. User Roles

1. Admin – Manage employees, approve leave, generate reports
2. Manager – Approve team leave, view attendance
3. Employee – Check-in/out, apply for leave, view history

5. Frontend Modules (React)

1. Authentication Module – Login, logout using Supabase Auth
2. Dashboard – Role-based dashboard (Admin / Manager / Employee)
3. Attendance Module – Check-in, check-out, daily status
4. Leave Management – Apply leave, view status, approvals
5. Reports Module – Monthly attendance and leave reports
6. Profile Module – Employee details and image upload
7. Admin Panel – Employee CRUD, role assignment

6. Backend Modules (Node.js + Express)

1. Auth Middleware – JWT validation and role-based access control
2. User Service – Sync Supabase users with application roles
3. Attendance Service – Check-in/out logic, validation
4. Leave Service – Leave request workflow and approvals

5. Report Service – Monthly report generation
6. File Upload Service – Upload images and reports to S3
7. Audit Logs – Track actions for security and debugging

7. Database Design (Supabase – PostgreSQL)

1. users (id, name, email, role, department)
2. attendance (id, user_id, check_in, check_out, date)
3. leave_requests (id, user_id, leave_type, start_date, end_date, status)
4. reports (id, month, s3_url, generated_at)

8. AWS Components Explanation

EC2: Hosts Node.js backend and NGINX reverse proxy.

NGINX: Routes traffic from frontend to backend securely.

S3: Stores employee images and generated PDF/CSV reports.

IAM Roles: EC2 is granted limited access to S3 buckets without hardcoding credentials.

9. Security Considerations

1. JWT-based authentication
2. Role-based authorization (RBAC)
3. Input validation at backend
4. Private S3 buckets with controlled access
5. HTTPS via NGINX (future enhancement)

10. Interview Discussion Points

1. Why Supabase instead of self-managed database
2. How role-based access is enforced
3. How attendance fraud is reduced
4. How the system scales with more employees
5. Future improvements like biometric or mobile app integration

11. Future Enhancements

Mobile app integration, biometric attendance, payroll integration, advanced analytics, and CloudFront CDN.