

Team Management App (Web-Based)

Core Roles

Need strict role-based access control:

- CEO
- HR
- Team Lead
- Intern

Each role sees different dashboards and actions.

System Flow (How It Should Work)

HR

Creates Intern accounts
Assigns Intern → Team Lead
Can assign/change Team Lead
Views all reports and analytics

CEO

Assigns Team Leads
Can reassign Team Leads
Views overall company analytics
Access to all project progress

Team Lead

Receives interns under them
Creates projects/tasks
Sets deadlines
Reviews intern weekly submissions (PDF)
Approves / rejects / adds feedback

Intern

Views assigned project
Uploads weekly work:

Title
Description
Required documents (PDF)
Sees feedback
Sees progress tracker

Main Modules

1. Authentication Module

Role-based login
Secure JWT authentication
Password reset
Profile management

2. Project Assignment Module

Fields for Project:
Project Title
Description
Start Date
Deadline
Assigned Intern
Assigned Team Lead
Status (Ongoing / Completed)

3. Daily and Weekly Work Submission Module

Intern uploads:
Serial no
Title
Description
PDF upload
Optional: additional documents
Team Lead actions:
Approve
Request changes
Add comments

4. Analytics Dashboard

CEO Dashboard

- Total Interns
- Total Active Projects
- Completion Rate %
- Daily and Weekly submission consistency
- Team-wise performance chart

HR Dashboard

- Intern count per Team Lead
- Attendance or submission tracking
- Delayed submissions
- Active vs Completed projects

Team Lead Dashboard

- Pending reviews
- Intern progress %
- Missed deadlines
- Performance comparison

Progress Tracking

- Submission consistency graph
- Completion percentage bar
- Timeline tracker
- Productivity score (based on deadlines + approvals)

Notifications System

- Project assigned → Notify intern
- Submission uploaded → Notify Team Lead
- Feedback added → Notify Intern
- Deadline reminder (auto reminder 1 day before)

Can implement:
In-app notification system
Email notifications

Optional later: WhatsApp/Slack integration

Suggested Tech Stack

Frontend:

React (Next.js recommended)

Tailwind CSS

Backend:

Node.js + Express

OR

Django (if Python)

Database:

PostgreSQL

Firebase (if you want simpler setup)

File Storage:

Cloudinary / AWS S3 / Firebase Storage

Authentication:

JWT + Role-based middleware

Deployment:

Vercel (frontend)

Render / Railway (backend)