⚙️ Project Scope Breakdown

- **Days Remaining:** 15 working days (Mon-Fri)
- **Team:** 6 core devs
  - 4 Backend (Spring)
  - 2 Frontend (React)
  - **Cross-work encouraged**

📅 Daily Plan (Starting May 6)

Date	Spring Dev Tasks	React Dev Tasks
Day-1 (May 6)	Setup Monorepo (EMS-backend), Initialize Auth & User Service, DB config	Setup Vite, Tailwind, Create React Layouts, Protected Routing (JWT Ready)
Day-2 (May 7)	Define Entity Models, Create Repos, Form JWT tokens, Design Login API	Build Login Page + Private Route Setup, Auth Context, Axios Config
Day-3 (May 8)	Complete Auth APIs (Login/Register), Secure endpoints	Dashboard skeleton (Sidebar, Navbar, Layout), Placeholder components
Day-4 (May 9)	Employee CRUD APIs, Pagination	Employee List Page, Table Component
Day-5 (May 10)	Attendance Service APIs, Log daily attendance	Attendance Page UI, Forms
Day-6 (May 13)	Leave Service APIs, Apply/Approve	Leave Request UI, Manager Approve Table
Day-7 (May 14)	Task Management API, Assign Tasks	Task Management UI, Status Filters
Day-8 (May 15)	Ticket Service API, Assign/View Tickets	Ticket UI, Status & Filters
Day-9 (May 16)	Feedback/Chat API	Chat UI (maybe simple poll), Feedback Form
Day-10 (May 17)	Integrate Services (Feign/Gateway) + Swagger/Doc	Hook up all pages to APIs, Loading & Error States
Day-11 (May 20)	Final Testing + Merge to Main	Final Testing + Merge to Main

---

🌟 Two Daily Meetings

- **9 AM Stand-up** → Task breakdown & blockers.
- **5 PM Evening Sync** → Push to `test` branch, demo progress.

🔑 **Important Tips:**

- First **3 days** are crucial (setup + Auth flow)
- **React Devs** can help backend once layouts are set.
- **Backend Devs** support API pairing with frontend.
- Prioritize **basic working features** before polish.
- Let's reserve **Friday, May 17** as buffer for RAEs (Random Ass Engineering).

🚀 15-Day Development Plan for Employee Management System (EMS)

🎯 **Goal:** Complete **React + Spring Boot** Microservices-based EMS by **May 20, 2025**
🧑‍💻 **Team:** 6 Members (4 Spring + 2 React) (Shared responsibilities)
📅 **Work Schedule: Monday–Friday** (Weekends Off)
📌 **Daily Meetings:**

- 🎬 **Standup (Morning)** → Assign tasks for the day
- 📌 **Evening Review** → Discuss progress & push to `test` branch

📌 General Development Flow

1. **Week 1 (May 6–10)** → Project Setup, Authentication, Database, API development
2. **Week 2 (May 13–17)** → Full Frontend, Integrations, Testing
3. **Final Days (May 20)** → Debugging, Deployment

🗓 Day-by-Day Plan

📆 May 6 (Monday) – Project Setup & Authentication

🔷 Spring Work (Backend)
- ✅ Setup `ems-backend` repo on GitHub
- ✅ Initialize Spring Boot Project
- ✅ Setup **Microservices Structure** (User, Leave, Attendance, Tasks, Tickets)
- ✅ Configure **MySQL Database & Entities**
- ✅ Implement **JWT Authentication API** (Login, Register)

🔷 React Work (Frontend)
- ✅ Setup `ems-frontend` repo on GitHub
- ✅ Install React with Vite, Tailwind CSS, React Router
- ✅ Design Folder Structure & Role-based Routing
- ✅ Build **Login Page UI**

📆 May 7 (Tuesday) – User & Role-Based Access

🔷 Spring Work (Backend)
- ✅ Implement **User Service API** (CRUD for Employees & Managers)
- ✅ Implement **Role-Based Access** (`MANAGER`, `EMPLOYEE`)
- ✅ Setup Spring Security with JWT

🔷 React Work (Frontend)
- ✅ Implement **Protected Routes** (Manager Dashboard / Employee Dashboard)
- ✅ Setup **Context API** for Authentication
- ✅ Design **Navigation & Sidebar**

📆 May 8 (Wednesday) – Leave & Attendance Services

🔷 Spring Work (Backend)
- ✅ Implement **Leave Management API** (Apply, Approve, Reject)
- ✅ Implement **Attendance API** (Check-in, Check-out, View Logs)

🔷 React Work (Frontend)
- ✅ Design **Leave Request Page**
- ✅ Design **Attendance Page**
- ✅ API Integration for **Login & User Details**

🗓️ May 9 (Thursday) – Tasks & Ticket Services

🔷 **Spring Work (Backend)**
✅ Implement **Task Management API** (Assign, Update, Complete Tasks)
✅ Implement **Ticket Management API** (Raise, Assign, Track Tickets)

🔷 **React Work (Frontend)**
✅ Design **Task Management Page**
✅ Design **Ticket Management Page**

🗓️ May 10 (Friday) – Feedback & Chat System

🔷 **Spring Work (Backend)**
✅ Implement **Feedback API** (Submit, Reply)
✅ Implement **Chat API** (WebSockets / Polling)

🔷 **React Work (Frontend)**
✅ Design **Feedback Page**
✅ Design **Chat UI**

🏁 End of Week 1 – Backend APIs Done, Frontend UI Ready

🚀 **Weekend Review:** Verify APIs using **Postman**
🔁 **Plan for Next Week: Start Integrating Frontend with Backend**

🗓️ May 13 (Monday) – Integrate Auth & User APIs

🔷 **Backend Work:**
✅ Debug & Refactor **Authentication & Role-Based APIs**

🔷 **Frontend Work:**
✅ Integrate **Login API with JWT**
✅ Fetch **User Details API**

🗓️ May 14 (Tuesday) – Integrate Leave & Attendance APIs

🔷 **Backend Work:**
✅ Final Testing of **Leave & Attendance APIs**

🔷 **Frontend Work:**
✅ Connect **Leave Management UI** with APIs
✅ Connect **Attendance Page** with APIs

🗓️ May 15 (Wednesday) – Integrate Task & Ticket APIs

🔷 **Backend Work:**
✅ Final Testing of **Task & Ticket APIs**

🔷 **Frontend Work:**
✅ Connect **Task Management UI** with APIs
✅ Connect **Ticket Management UI** with APIs

🗓️ May 16 (Thursday) – Integrate Feedback & Chat APIs

🔷 **Backend Work:**
✅ Final Testing of **Feedback & Chat APIs**

🔷 **Frontend Work:**
✅ Connect **Feedback Page** with APIs
✅ Connect **Chat UI** with APIs

🗓️ May 17 (Friday) – Final Frontend Improvements & UI Fixes

🔷 **Backend Work:**
✅ Fix API Issues & Optimize Code
✅ Add Error Handling

🔷 **Frontend Work:**
✅ Improve UI/UX (Loading, Validation)
✅ Fix Bugs & Mobile Responsiveness

🏁 May 20 (Final Day) – Testing & Deployment

✅ **Test Entire System**
✅ **Fix Bugs**
✅ **Deploy Backend & Frontend**
✅ **Final Demo & Submission**

🔄 Daily Workflow (Every Workday)

Time	Task
9:30 AM – 10:00 AM	Standup Meeting (Assign Work)
10:00 AM – 1:30 PM	Morning Work Session
1:30 PM – 2:30 PM	Lunch Break
2:30 PM – 5:30 PM	Afternoon Work Session
5:30 PM – 6:00 PM	Evening Review (Push Code to `test` Branch)

🔥 Final Checklist

✅ **API Testing in Postman**
✅ **Frontend-Backend Integration**
✅ **Bug Fixes & Debugging**
✅ **Deployment**