

Phase	Team 1 (Frontend Work)	Team 2 (Backend Work)
1. Setup & Planning	- Design UI wireframes (Figma)- Setup project using React or Next.js - Folder structure (components, pages, services)- Setup Tailwind CSS	- Choose stack (e.g., FastAPI or Express.js) - Setup backend structure (routers, services, models)- Setup PostgreSQL/MongoDB- Setup .env , Dockerfile
2. Authentication	- Build login/signup forms- Role selection dropdown- UI validation- Store JWT in localStorage	- Implement JWT auth- User model with role- Login, Register, Role-based access API- Hashing (bcrypt)
3. Control Tower Dashboard	- Design Dashboard Layout- KPI Cards (current demand, inventory)- Alerts Panel UI (severity color-coded)- Interactive Network View (map/chart of all nodes)- Quick Access Cards to Rebalancer, Forecast, Settings	- Create /dashboard/ API returning metrics (KPI values)- Alerts DB model & grouping logic- /alerts/ API (grouped by type, urgency)- Node data API for Network View
4. Data Ingestion (File Upload)	- File upload UI (CSV/Excel)- Table preview before upload- Upload progress bar- Error preview UI (bad rows)	- File parsing (Pandas/CSV parser)- Validate field types & nulls- Create /upload/validate and /upload/process APIs- Store file logs- Alert if file missing or corrupted
5. Configuration Module	- Forms: Add/Edit Stores, Warehouses, SKUs- Dropdowns with async data (categories, paths)- CRUD UI for Reorder Level, Safety Stock- Strategy toggle UI (Push/Pull, Static/Dynamic)	- APIs to manage: - /nodes/ (warehouses, stores) - /products/ , /settings/ - Logic for supply chain paths- Role-based config edit restrictions- Store strategy modes in DB
6. Alerts System (Advanced)	- UI: Filter/sort/group alerts- Severity tags- Click to see alert source- Toast notifications for new alerts	- Tag alerts by similarity (rule-based or ML later)- API /alerts/grouped and /alerts/history - Store past alerts in DB
7. Ingestion (Advanced - APIs / Cloud Buckets)	- Toggle UI to select ingestion method (upload/API/cloud)- Show last success status, next expected upload time- Ingestion method config form	- Handle scheduled ingestion (cron job)- GCS/AWS S3 integration- Auto-validate incoming API data- Store ingestion status & generate alerts
8. User & Role Management	- Admin Panel UI- Add/edit/delete users- Assign roles (planner, RM, admin)	- /users/ API- RBAC logic (restrict config access)- Role table & permissions
9. Testing & Polishing	- Responsive checks- Form field validations- Final dashboard polish- Deploy on Vercel/Netlify	- Write unit/integration tests- Dockerize app- Deploy on Render/GCP- Add logging (Sentry/console)