

I want sprint wise details

Here's a **detailed, sprint-wise development plan** for your "Engagely" employee engagement platform, showing deliverables and concrete steps for each phase. Each sprint is focused on manageable milestones to help you build and see progress rapidly.

Sprint-Wise Details

▮ Sprint 0: Project Setup & Planning (1 week)

- Finalize MVP features and user stories.
- Select tech stack (Node.js backend, Python AI service, React frontend).
- Create GitHub repo: `engagely-platform`.
- Set up folder structure: `/backend`, `/ai_service`, `/frontend`, `/docs`, `/docker`.
- Draft initial README.md with objectives and architecture.
- Configure version control, basic linting, and editor settings.

▮ Sprint 1: Backend Foundations (1 week)

- Initialize Node.js project in `/backend`.
- Set up Express.js (or NestJS), configure initial app structure.
- Design and implement database schema (User, Employee, Survey, Feedback).
- Integrate Sequelize or TypeORM with chosen DB (start with SQLite or PostgreSQL).
- Create authentication endpoints:
 - `POST /api/auth/register`
 - `POST /api/auth/login (JWT)`
- Implement basic user management endpoints:
 - `GET/POST/PUT/DELETE /api/employees`
- Protect API routes via JWT middleware.
- Test all endpoints with Postman.

▮ Sprint 2: Core Frontend & Survey APIs (2 weeks)

- Initialize React project in `/frontend`.
- Build sign-up and login pages; integrate with backend API.
- Create dashboard for employees and admins.

- Develop CRUD APIs for Surveys and Feedback:
 - POST/GET/PUT/DELETE /api/surveys
 - POST /api/feedback
- Build frontend forms for creating surveys and submitting feedback.
- Display employee lists, surveys, and feedback data.
- Connect frontend to backend with API calls (use Axios/fetch).

▮ Sprint 3: AI Integration (2 weeks)

- Initialize Python FastAPI or Flask app in /ai_service.
- Train basic sentiment analysis model (start with scikit-learn or Hugging Face).
- Expose POST /analyze-sentiment endpoint.
- Test the AI service with sample requests.
- Integrate backend with AI microservice:
 - When new feedback is submitted, backend sends text to AI endpoint and receives sentiment score.
 - Store sentiment results with feedback.
- Show sentiment score in employee/admin dashboard.

▮ Sprint 4: Analytics, Nudges & Rewards (2 weeks)

- Implement analytics endpoints:
 - Average sentiment per team/user over time.
 - Flags for low engagement/burnout risks.
- Add logic for manager nudges:
 - If negative sentiment detected, notify manager.
- Add email/Slack/Webhook notifications for alerts.
- Create basic reward system APIs:
 - Award points, redeem points.
- Build frontend leaderboard or recognition view.

▮ Sprint 5: MVP Polish & Integrations (2 weeks)

- Refine UI and enhance UX (improve forms, navigation, error handling).
- Improve dashboard visualizations (charts for analytics).
- Harden authentication, role-based access control.
- Integrate third-party services:
 - SendGrid for emails; Slack for manager alerts.
- Add comprehensive error handling and API/data validation.

- Finalize documentation (/docs).

● **Sprint 6: Deployment & QA (1 week)**

- Create Dockerfiles for backend, AI, and frontend.
- Set up `docker-compose.yml` to run all services.
- Deploy to Heroku, Render, or AWS Lightsail.
- Test deployment, run full QA (manual, automated unit/integration tests).
- Prepare onboarding materials and bug-fix as needed.

○ **Sprint 7+: User Feedback & Iteration (Ongoing)**

- Onboard test users/first customers.
- Gather feedback through surveys and direct use.
- Prioritize UX improvements, bug-fixing, and new features.
- Plan for regular 1-2 week sprints, adding:
 - Advanced AI models
 - Real-time notifications
 - Enhanced analytics and integrations

▮ **Pro Tips (Each Sprint)**

- Finish and test each API or UI screen before moving to next.
- Maintain a working state—deploy often, even if to staging.
- Document any decisions, issues, or lessons learned in /docs.
- Keep communication clear if collaborating with others in the future.

With these detailed sprint plans, you can remain focused and make visible progress at every step—just like a pro team lead would guide!