

Product Engineer Assessment – Employee Pulse Application

Objective

- This assessment is designed to evaluate your ability to:
- Define and document product requirements with clarity and user empathy.
- Build and ship a functional product using modern full-stack tools.
- Make smart tradeoffs between product goals and engineering realities.
- Think holistically about quality, testing, and long-term maintainability.

You'll design and build an Employee Pulse Survey Application with both employee and admin functionality.

Assignment Tasks

1. Product Thinking – PRD & Feature Definition

Create a short Product Requirements Document (PRD) that includes:

- Problem Statement – What problem are we solving?
 - Target Users – Who will use this and why?
 - User Stories – 4–6 user scenarios from both employee and admin perspectives.
 - Key Features – Core functionality, split by Employee/Admin views.
 - Feature Prioritization – Must-have vs. Nice-to-have.
 - Success Metrics – How will you measure success?
 - (Optional) Include wireframes, mockups, or user flows.
-

2. Build the Product – End-to-End Development

Tech Stack:

- Frontend: React or any modern JS framework.
- Backend: NestJS (TypeScript).
- Database: MongoDB.

Employee Functionality:

- Register / Login.
- Take a pulse survey (text input is sufficient).
- View past submissions

Admin Functionality:

- Admin login.
- View all employee survey responses.
- Export data (CSV or JSON).

You're free to define how features are implemented, how data is modeled, and how APIs are structured.

3. Testing Strategy

Include a brief Test Strategy that outlines:

- What areas of the app you would test and why.
 - Types of tests you would write (unit, integration, end-to-end).
 - Sample test cases (write at least 10).
 - How you would maintain test quality and automation over time.
-

Deliverables

Create a GitHub repository containing:

- PRD (in Markdown or PDF format).
 - Complete frontend and backend code.
 - README with clear setup instructions.
 - Test strategy (in README or separate file).
 - At least 5 working test cases.
-

Notes

- You are welcome to use AI tools like ChatGPT, GitHub Copilot, etc., to increase development speed.
- Prioritize clarity, usability, and thoughtful design over perfection.
- We're evaluating how you think, build, and validate — not just your code.

