

Build a Lightweight Feedback System

We recommend you read through this thoroughly before you start working on it. We have a FAQ section at the end.

Context

You're building a tool for internal feedback sharing between managers and team members at a company. The goal is to enable structured, ongoing feedback in a simple, secure, and friendly interface.

This assignment is meant to judge your creativity and ability to convert open ended instructions into something real.

Core Features (MVP)

Authentication & Roles

Two user roles: Manager and Employee

Basic login system

A manager can only see their team members

Feedback Submission

A manager can submit structured feedback for each team member:

Strengths

Areas to improve

Overall sentiment (positive/neutral/negative)

There can be multiple feedbacks to the employee.

A history of feedback is visible to both manager and employee

Feedback Visibility

Employees can see feedback they've received

They cannot see other employees' data

Managers can edit/update their past feedback

Employees can acknowledge feedback they have read

Dashboard

For a manager: team overview (feedback count, sentiment trends)

For an employee: timeline of feedback received

Bonus (Optional)

Deployment link (e.g., Render, Vercel, Railway, etc.)

Employee can request feedback proactively

Anonymous peer feedback (with optional flag)

Notification (email or in-app)

Tags (e.g., "communication", "leadership", etc.)

Export feedback as PDF

Employees can submit their comments on feedback

Markdown support for comments



Tech Guidelines

Any frontend framework: Vue / React / Svelte

Backend: Python

Database: SQLite / Postgres / any

Additional Requirement: Provide a Dockerfile to run the backend app



Deliverables

Once you are done, submit your responses to <https://forms.gle/Epw8JTHVdZM4SnRx8>

Video Demo of the application - you can upload to a platform like youtube/Loom as an unlisted video and share the link - Limit the video to 5 minutes or less.

Video walkthrough for the codebase - you can upload to a platform like youtube/Loom as an unlisted video and share the link - Limit this to 10 minutes or less

A public GitHub repo with:

README.md containing:

Setup instructions

Stack and design decisions

A Dockerfile to build and run the backend

Note that we will look at the two videos before we look at the codebase. Make sure your presentation is top notch.

Video submission guidelines:

Make sure the video permissions are public if you're sharing a google drive link - if its not a public link, we can not look at it.

You need to record yourself talking during the demo - speak clearly and loudly

Please do some research on how to make a good product demo. You need to show the product in this demo

Do not use AI voiceover for videos - we want to hear you speak

🧠 What We're Evaluating
Communication skill and presentation skill

Clarity of UI/UX

Thoughtful DB design (role boundaries, access controls)

Clean, modular code (frontend & backend)

Bonus: creativity and empathy in solving a real-world problem

? FAQ

Q: Can I use NodeJS for backend?

A: No - we are specifically looking for Python expertise.

Q: Can I use AI tools to assist in writing code?

A: Its 2025 - everyone uses a bit of AI assistance. Yes, but be transparent in your README if you did. We're evaluating your judgment and design ability more than line-by-line code. However, we expect clarity on what you have built - you will review the AI generated code well before adding it to your solution.

Q: Isn't this is a lot of work for an assignment?

A: We understand this. Unfortunately, we have a very large number of applications and limited capacity to interview everyone. This is also why use of AI tools is absolutely fine - we're evaluating how well you deliver using the tools available to you. Make sure to understand AI output completely before adding it to your submission.

Q: Can I contact someone for clarifications or doubts?

A: Unfortunately, no. We receive a large number of applications and are unable to provide individual guidance. Use your creativity and best judgment—we're evaluating how you think and solve problems independently. However please submit your question at <https://forms.gle/mUX7yQu3aruK8aTA6> and we may update the FAQ with the answer.

Q: I am not familiar with Python. Can I still apply?

A: Sorry, this assignment is specifically for evaluating Python developers. We recommend not picking this up unless you're already comfortable with Python and love Python, as there will be other applicants with that experience.

Q: Can I use a frontend framework not listed above (e.g., Angular)?

A: Yes, any stack is allowed. Just make sure the UI is clean and functional.

Q: Can I use a UI framework not listed above?

A: Yes - you can choose to use a UI framework like bootstrap or Vuetify to make your life easier.

Q: Does the authentication have to be real?

A: No, mocked login is fine if you feel its a lot of work to implement auth. Just ensure role-based access works as expected.

Q: How much time should this take?

A: You have 1 week to complete the assignment. We want you to take enough time to do a thoughtful job, not rush through it. There are no extra points for fast submission.

Q: Can I work with a teammate or use past project code?

A: This should be an individual assignment. You can reuse generic boilerplate, but the core logic should be your own. During the first round of interview, we will try to understand your solution and ask you to add more features.

Q: Do I need to write tests?

A: Not mandatory, but definitely a plus!

Q: Can I do only the frontend or backend?

A: No. Only fully working end-to-end solutions will be considered. Make sure both parts are functional and integrated.

Q: How many open positions are there?

A: We are hiring for 3 positions.

Q: Is there a stipend for the internship?

A: Yes, the internship is paid.

Q: Is this a full-time internship?

A: Yes, it is a full-time position.

Q: Is it remote?

A: No.

Q: Can I apply if I am still in college?

A: No, this internship is only open to graduates.

Q: What if I can't finish the assignment in 1 week?

A: Only fully working submissions will be considered. If you are unable to complete it in 1 week, it's best not to submit.

Q: Will I get feedback on my submission?

A: Due to the volume of applications, we are unable to offer individual feedback.

Q: What happens after I submit?

A: We will review all submissions and reach out to shortlisted candidates within a week of the deadline.

Q: Can I use a UI library like Tailwind or Bootstrap?

A: Absolutely. Use any UI tooling you're comfortable with to build a clean and usable interface.

Q: Will the assignment be evaluated only on completion?

A: Candidates who have finished the solution are given more preference.

Q: Is dockerization mandatory

A: Yes.

Q: Do I need to use Docker for both frontend and backend?

A: No, only the backend needs to be dockerized. The frontend can be deployed normally.

Q: What if I don't have a public URL for deployment?

A: Please mention this clearly in your README. A video demo is mandatory, but a public deployment is strongly preferred for evaluation.

Q: How important is design and styling?

A: We're not expecting fancy UI, but the layout should be clean, intuitive, and show empathy for the user.

Q: Can I change or add to the problem statement?

A: Yes, as long as the core functionality is met, creative additions or enhancements are welcome and encouraged.

Q: What is the duration of the internship?

A: The internship duration will be 6 months.

Q: Can this internship convert into a full-time offer?

A: Yes, there is a possibility of conversion based on performance.

Q: Should I include a database schema diagram or ERD?

A: Not mandatory, but including one (or mentioning structure clearly in the README) is appreciated and shows clarity.