

Take-Home Assessment: Applicant Tracking Prototype

Background

Our current frontliner hiring process involves filling in a paper form, which is then sent back to HQ for HR to manually process. We want to digitize this and eventually integrate with an ATS.

Your task is to build a **prototype mobile application** (Flutter) with a simple **backend service** (Laravel + MySQL) that demonstrates how this process could work digitally.

Core Requirements

1. Flutter Mobile App

- A simple **job application form** where candidates can enter:
 - Full name, phone number, position applied, work experience (short text).
 - Email address (*optional* – note that some frontliners may not have one).
- On submission:
 - Data is sent to the backend.
 - Candidate receives a **confirmation screen**.
 - If an email is provided: send a confirmation email (real or mocked).
- A simple **“My Application Status” screen** where a candidate can check their progress (applied → screening → interview → offer).

2. Laravel Backend

- REST API to accept applications and store them in a database.
- API to update application status (to simulate HR actions).
- Trigger an email (real or mocked) when an application is submitted *if an email address exists*.

3. Database

- Use **MySQL** for schema design and queries.
- Each applicant record must track current status and timestamps.

- **Flexibility:** If you are more comfortable with another relational database (e.g., Postgres, SQLite), you may use it for this prototype — but please include a short note in your README on how this would translate to MySQL in production.

4. Tests:

Add at least a few unit or integration tests (e.g. for form validation, API endpoints, or database logic). Adding at least a few unit or integration tests will strengthen your submission and is something we actively look for.

Bonus (Optional)

Not required, but nice to see if you have extra time:

- **Mock job board posting API:** Define an endpoint that could “post a job” to external boards (no real integration needed).
 - **Basic recruiter dashboard:** A simple web page (can be Laravel Blade) where HR can view applications and change their status.
 - **Firebase build setup:** Show you can configure Firebase for app builds/releases (a short explanation in README is enough).
-

Time Expectation

We estimate this should take 2–4 hours if you use AI tools effectively, or 6–8 hours if coding manually.

We encourage the use of AI to handle repetitive tasks (boilerplate code, scaffolding, etc.), but we will be evaluating:

- **Your judgment** (what you built, how you scoped it).
 - **Your ability to integrate pieces into a working prototype.**
 - **Your clarity of explanation** in the README.
-

Deliverables

- Source code (Flutter + Laravel). Please create a public repository on your Github account and share it with us.
- A short README that includes:
 - Setup instructions (how to run backend and app).
 - Design decisions (how you structured the app + API).

- If you had more time, how you would extend this into a fuller ATS (job boards, automation, analytics, etc.).
-

Evaluation Criteria

- **Flutter:** UI clarity, state management, and user experience.
- **Backend:** Laravel code quality, API design, database schema.
- **Integration:** How smoothly the app and backend connect.
- **Code Quality:** Readability, maintainability, basic testing if possible.
- **Pragmatism:** Did you scope well? Did you avoid overbuilding?
- **Communication:** README clarity, explanation of trade-offs, and future thinking.