

Our product manager wants to unify professional sign-ups that currently happen through different sources — the website (direct), partner referrals, or internal additions.

We need a small **backend + frontend prototype** to:

- Collect professional details from multiple sources
- Display them in one place
- Filter by signup source

## Task Overview

You'll build:

- A **Django REST Framework** backend API
- A **React** frontend interface

## Requirements

### Backend (Python / Django)

#### Endpoints

1. **POST /api/professionals/**
  - Create a professional
  - Fields: `full_name`, `email`, `phone`, `source`
  - Validate unique email, phone and valid source (`direct`, `partner`, `internal`)
2. **POST /api/professionals/bulk**
  - Accepts a list of user profiles to create or update.
  - Upsert each profile using its email as the unique key. If email isn't provided, use `phone_number` as the unique key.
  - Should be able to handle partial success.

### 3. GET `/api/professionals/`

- Returns all professionals
- Optional query param `source` (filter)

#### Model

##### Professional

- `id`
- `full_name (string)`
- `email (string, unique)`
- `company name ( string )`
- `job title (string )`
- `phone (string)`
- `source (choice: direct | partner | internal)`
- `created_at (auto timestamp)`

#### Tech

- Django REST Framework

### Frontend (React)

#### Pages

- **Add Professional Form**
  - Fields: Full Name, Email, job title, company name, Source dropdown
  - Submit → POST to backend
- **Professionals List**
  - Fetch all professionals → `/api/professionals/`
  - Table view
  - Dropdown to filter by source

# Extensions

Considerations for handling a PDF resume uploaded via **POST /api/professionals/**:

- How would the system process and interpret the content of the uploaded PDF?
- What is the proposed method for handling the actual file upload?
- What additional functionalities or elements would be incorporated into the frontend for this feature?

## Submission Format

### Option 1 – GitHub Repository (*preferred*)

- Push your solution to a **public GitHub repo** (or private with access granted to reviewers).
- Include a clear **README.md** with:
  - Setup instructions (backend & frontend)
  - Any assumptions or trade-offs you made
  - Estimated time spent and what you'd improve with more time

### Option 2 – Zip File

- Export the entire project folder (both **backend/** and **frontend/**) into a single **.zip** file.
- Include your **README.md** with the same details as above.

Note: You can use AI tools for coding and not spend more than 3 hours