Take-Home Assignment - Intermediate/Senior Developer Shortlist

As Salam Alaikum and congrats on making it to our shortlist! 👋

This take-home assignment is designed to evaluate your ability to work with REST APIs and build a simple full-stack feature. This is a simulated environment with a mock API.

Scenario - Virtual Wellness Platform Integration

You're joining a team building an internal tool for a wellness clinic. Your task is to build a small system that allows clinic admins to manage clients and their appointments using an external API (mocked via Postman).

This exercise is inspired by the kind of work you'll do at Ruh: integrating with APIs, building robust and secure infrastructure, and creating great user experiences in a healthcare environment.

Time Expectation

This assignment is expected to take **6–9 hours** to complete. The deadline to submit the take-home assignment is Sunday, July 20th, 11:59 pm EST.

Please track the total time you spend on this assignment and include it in your final submission.

We understand your time is valuable. If you're short on time:

1. Prioritize the **core functionality** (client list, appointment creation)
2. Include **README notes** on how you'd complete any unfinished features

We value **clarity of thought, clean code, and strong communication** more than completeness. Feel free to make and document any reasonable assumptions.

Your Task

Backend

1. Create a backend using **Ruby** (or any language you're comfortable with from our stack)
2. Build an API wrapper to:
    a. Fetch a list of clients
    b. Fetch a list of appointments
    c. Create new appointments
3. Store data in **PostgreSQL** and simulate periodic syncing

Frontend

1. Build a UI using **React** (or React Native for mobile)
2. Display a list of clients showing: name, email, and phone number
3. Show upcoming appointments
4. Add a form to schedule a new appointment

Optional Bonus

1. Edit or cancel an appointment
2. Add search/filter for clients
3. Build a minimal mobile version using **Swift** or **Java**

Mock API Instructions

We have provided a **Postman Collection** you can import and use to simulate the API.

Download the Postman Collection via
https://drive.google.com/file/d/1ya4n3XlLuv9EZi2Wtp8Sxp6TEgbf77xy/view?usp=sharing

Setup Mock Server

1. Sign in to Postman or create a free account
2. Import the collection into your Postman workspace
3. Follow this guide to set up a mock server:
   https://learning.postman.com/docs/designing-and-developing-your-api/mocking-data/setting-up-mock/
4. Once created: Replace `https://mock.api` in the collection with your mock server URL
5. Make sure your code uses the new mock URL

Submission Guidelines

1. Upload your code to GitHub or share a zip file
2. Include a `README.md` with:
    a. Setup instructions
    b. Stack used
    c. Assumptions made
    d. (Optional) Deployment or demo link
    e. Any incomplete items and how you'd approach them
    f. Total time spent on the assignment
3. Send your submission to: omar@ruhcare.com

Evaluation Criteria

1. Clarity and structure of code
2. How well you handle API integration
3. Error handling and edge cases
4. Code readability and maintainability
5. UX and responsiveness
6. Communication through README and documentation

Thank you for taking the time to complete this challenge. We're really looking forward to seeing your work!