Coding Exercise

Hello, \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_!

Below is a coding exercise that we believe will allow you to show off your amazing development skills!

We’re super excited to see what you come up with!

We expect it to take a few hours. We would ask that you make commits to a git repository every so often so that we can see how long it took. Here is what you need to do

1. Clone this repository locally and update the first line of this README with your name (so that it reads "Hello, YOUR NAME!"). Commit this change. This will serve as a starting timestamp
2. Complete the exercise below
3. Commit progress regularly
4. When you're done, upload your code to a personal GitHub account and email us the link.

# The Exercise

This is a simplified version of a piece of functionality we have in Breeze ChMS. Many of the churches we work with import their data from an existing system or a homegrown spreadsheet into Breeze. We provide tools for bulk importing of people, contribution, group and attendance records.

In this problem we're only going to consider **two data types: People and Groups.** A Person can be part of one Group.

For the People data type, each person can have a state of either 'active' or ‘archived’. **The id for each data type is globally unique**. As a result, if the id does not exist, create a new record, otherwise, update the existing record.

* **People columns:** id, first\_name, last\_name, email\_address, status
* **Group columns:** id, group\_name

Here’s an example:

**id, first\_name, last\_name, email\_address, status**

1, "Alex", "Ortiz-Rosado", "alex@breezechms.com", active

2, "Jon", "VerLee", "jon@breezechms.com", "archived"

3, "Fred", "Flintstone", "fredflintstone@example.com", "active"

4, "Marie", "Bourne", "mbourne@example.com", "active"

5, "Wilma", "Flintstone", "wilmaflinstone@example.com", "active"

**id, group\_name**

1, "Volunteers"

2, "Elders"

3, "Bible Study"

# Exercise Setup Help

*Help getting the code up and running*

**Prerequisites**

* Git, Composer, Laravel
* Node >= 8, Yarn
* Command Line PHP 7
* MySql 5.x installed locally, accessible via 127.0.0.1
* Clone the repository
  1. git clone git@github.com:BreezeChMS/coding-exercise-api-react.git
  2. cd coding-exercise-api-react
* Setup Laravel
  + cp .env.example .env
  + Edit .env with your mysql connection information: the following steps connect to local mysql database using root credentials
  + composer install
    - php artisan key:generate
    - php artisan migrate
* Start the Laravel API in one Terminal Window: **php artisan serve**
* Start the React/Node.js server in another Terminal Window: **yarn start**

# Expected Changes

* “add a new endpoint for /groups that supports crud ops”
  + Update the RESTful API (built using the Laravel framework) to add a *new* endpoint for /groups. This endpoint should support CRUD (Create, Read, Update, Delete) operations.
* “make a csv uploader in React using the rest api and render the results”
  + Update the ReactJS application to receive an uploaded People CSV file, import it using the RESTful API service and display the results on the screen. The same application will allow you to do the same thing for a Group CSV file.
  + Feel free to use a CSV parsing library. **The data will be displayed in a sortable table.**
* “render a list of groups & a list of people in each group”
  + You will need to determine the type of data in the CSV file based on the headers in the first row. Your program will output a list of Groups, and for each Group, a list of active People in that Group.

# Testing

We love TDD! So we’d love to see tests for the API and ReactJS application. Write automated tests to verify your results and account for gotchas (handling different column orders, invalid id's in the People CSV file, etc..). Classify your tests as either unit, integration, ui, or acceptance, but it is not required to use every type.

# Finally

We’re a fully remote team so communication is really important. Be sure to include any instructions needed for any of our team mates to run and test.

Good luck and we'll get back to you once we review it!