Backend Challenges

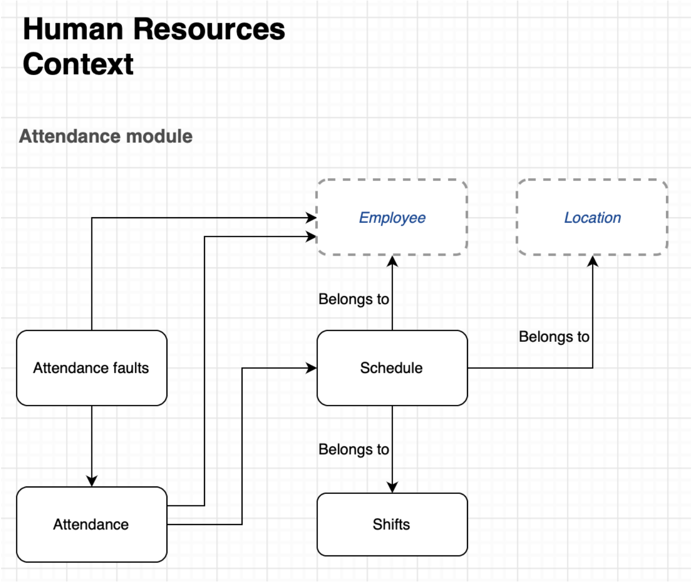
https://github.com/Connect-Group-Tech/Backend-coding-test

1. We appreciate a clean and simple creative solution.
2. The application must be in a Github repository with the commit history, we would like to see your repository management.
3. We prefer PHP (Laravel) for backend.
4. You need to use React JS with typescript for the front end.
5. You can choose your own design for the frontend
6. If there are some implementation details not specified, you can use any approach to complete the implementation.

Challenges

Challenge 1:

Create a migration to support the below structure



Create an API endpoint to upload excel attendance and store data in the database.

Create an API endpoint to return attendance information of an employee with total working hours.

Create a controller for attendance and call the methods from the below class

Create a service inside

```
src->AppHumanResources->Attendance->Application->AttendanceService.php
```

Create a model class :

```
src->AppHumanResources->Attendance->Domain->Attendance.php
```

Create a view in the front end to list the attendance information, and show N/A if the check in or checkout is not available.

Show the Attendance information in the following format.

Name | checkin | checkout | total working hours

Challenge 2:

Given an array a[] of size N which contains elements from 0 to N-1, you need to find all the elements occurring more than once in the given array.

Example :

Input:

N = 5

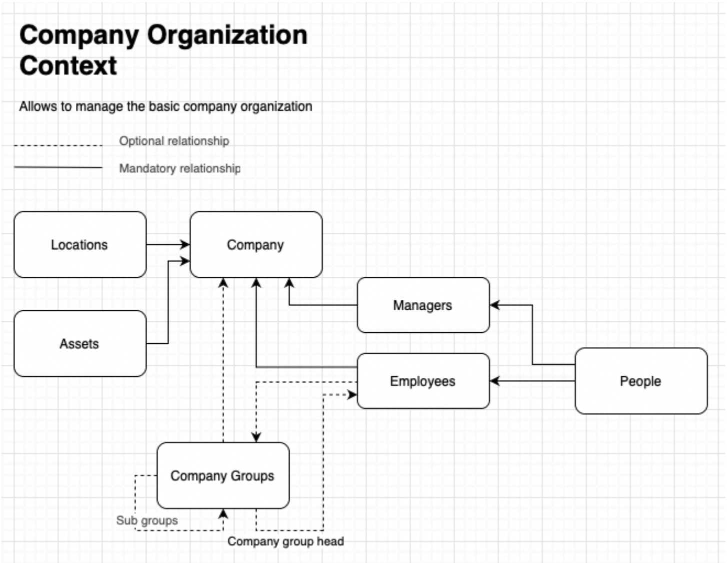
a[] = {2,3,1,2,3}

Output: 2 3

Explanation: 2 and 3 occur more than once in the given array.

Challenge 3:

Create a migration file to create the below structure, use the fields that you consider necessary for each table.



Challenge 4:

Create an application service “groupByOwnersService” to process the array given and return it in the desired format with a test to check the functionality.

For example, for associative array

["insurance.txt" => "Company A", "letter.docx" => "Company A", "Contract.docx" => "Company B"]

the groupByOwnersService function should return

["Company A" => ["insurance.txt", "letter.docx"], "Company B" => ["Contract.docx"]].