

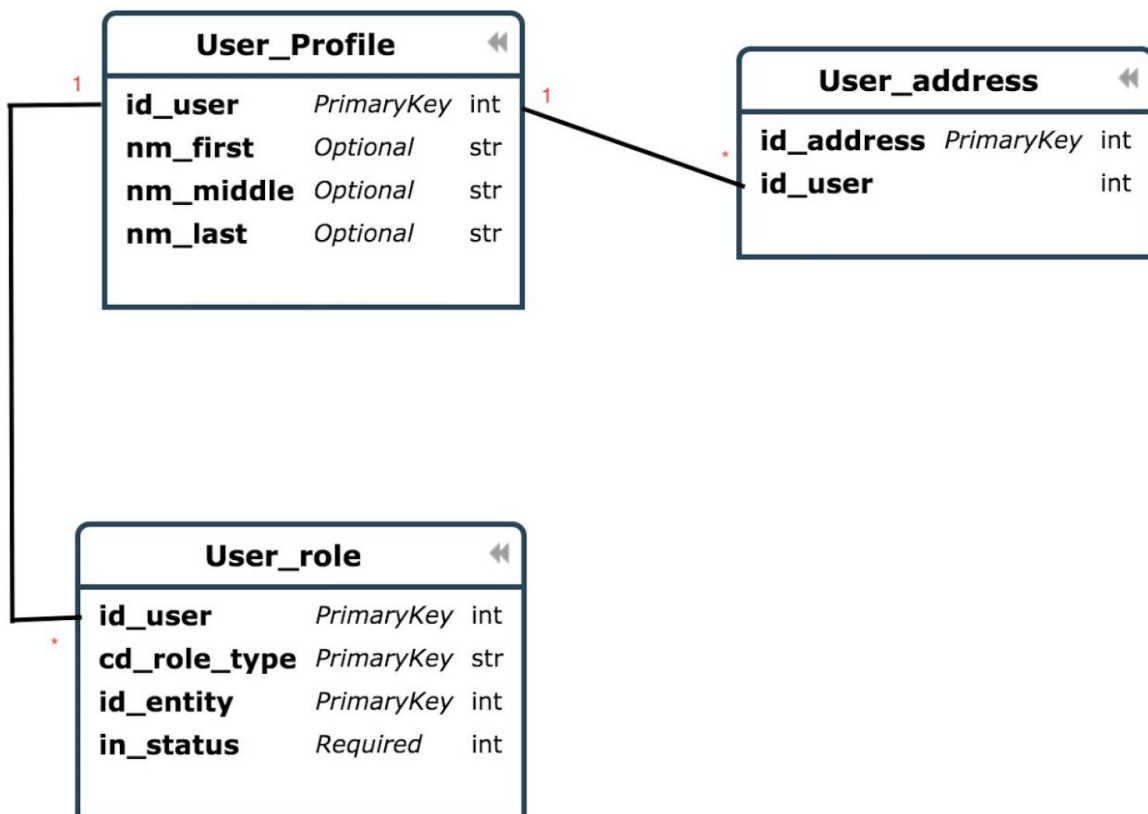
Technical Test

DB Test:

Our CEO wants to launch some new email campaigns to a very specific population of our users. In order to do that, it's needed a couple of reports and you are in charge of pulling them from the database just as they were requested.

Context data

- Below you will find described all the tables and columns you will need in order to complete your requirements:



- Active users are those ones whose primary key value (*user_profile.id_user*) is greater than "0" (zero), and that *user_role.in_status* equals to "1".
- The user type for a user is stored in the column *user_role.cd_role_type*.

Requested Reports

1. You are required to get how many active users we have per type, and for each type, the amount of users for whom we don't have a middle name stored. This report should contain only three columns:
 - a. "User Type", text column indicating the name of the column type.
 - b. "Total Active", numeric column indicating the total of active users that belongs to each type.
 - c. "No Middle Name", numeric column indicating the total of active users that belongs to each type, and don't have a middle name stored.

Notes: There should be only one record per type. Column titles should be the same as requested.

2. You are required to get the total active users of types "**LICENSEE**" and "**LIMITED**" that we have address information for. This report should only contain one column: "Active Licensees With Address Info", a numeric column indicating the requested total.

Notes: The column title should be the same as requested. There should be only one row and one column. Each user should count only once.

3. You are required to get the total of non-active users of type "**PROVIDER**". This report should only contain one column: "Non-active Providers", a numeric column indicating the requested total. Notes: Each user should only count once. Column title should be the same as required.

Expected Results

1. A folder with your name.
2. Within the folder, add an SQL file per report requested.

Frontend Test:

The company has created a new service to return the compliance status for licensees of some states in the US.

The service has been built with both REST and SOAP endpoints.

One of the main nonfunctional requirements this project was shipped with is transaction logging. All the data that comes through and goes out is stored in a separate log database.

As part of the project, the following endpoint was created to retrieve those log records based on a Date Range and/or a State Code:

GET

https://api.cebroke.com/v1/cerenewaltransactions/GetLogsRecordData?startdate=[MM/DD/YYYY]&enddate=[MM/DD/YYYY]&state=[State Code]

Returning an array of JSON like this:

```
[
  {
    "cd_cebroke_state": "FL",
    "pro_cde": 1701,
    "cd_profession": "RN",
    "id_license": "999999999",
    "dt_end": "2020-04-30T00:00:00",
    "cd_environment": "PRIMARY",
    "dt_Start_Log": "2017-08-03T19:26:13.89923",
    "ds_compl_status_returned": "IN_PROG",
    "dt_end_log": "2017-08-03T19:26:14.347996",
    "cd_machine": "JAX00",
    "id_client_nbr": "1"
  }
]
```

However, they have no frontend at all and sometimes is hard for developers to debug an error situation. So the company needs to build a dashboard page to display those transaction logs, with the following functional requirements:

- Me as a user, want to get a list of logs for the current day by default.
- Me as a user, want to see the following data per log (in the same order):
 - State Code
 - Pro Code
 - Profession

- License ID
 - Cycle End Date
 - Compliance Status
 - Client ID
 - Start Log Date
 - End Log Date
 - Environment
 - Machine
- Me as a user, want to get always the list of logs displayed in pages, with 20 records per page.
- Me as a user, want to get always the list of logs ordered by Start Log Date by default.
- Me as a user, want to be able to order the list of logs displayed by any other field.
- Me as a user, want to be able to filter the logs by Date Range.
- Me as a user, want to be able to filter the logs by State Code by selecting from a dropdown any of these values:
 - FL
 - OH
 - GA
 - LA
- Me as a user, want to see the following statistics calculated over the data being displayed:
 - Total Average Response Time (Number)
 - Average Response Time per Day (Line Graph)
 - Total Requests per Machine (Bar Graph)
 - Total Requests per Compliance Status (Bar Graph)

There are no specific restrictions or requirements regarding to the look and feel of the dashboard page, but it is really important that it is a fresh, clean and understandable design.

Backend Test:

We'll give you a mongoDB database with two collections, "providers" and "specialties". Please build an API REST and expose a CRUD for the providers collection. Following the info to access the database:

- Access via mongo client:

```
mongo ds125146.mlab.com:25146/foundation-test1 -u <dbuser> -p  
<dbpassword>
```

- Connection String:

```
mongodb://<dbuser>:<dbpassword>@ds125146.mlab.com:25146/found  
ation-test1
```

Please ask to the interviewer for the <dbuser> and <dbpassword>.

Play Rules!

1. You are free to use any backend/frontend technology of your choice. However, for the backend test, we are looking for javascript developers, so if you choose to use node, that might add up some extra points.
2. We think good design is good business so make the website looks as great as you can.
3. Both the page and the code must be written in English.
4. Please be sure to leave your comments in the code where needed.
5. You'll have the choice to estimate how long this test could take, so think well!
6. You can choose the logo and the top banner of your choice.
7. Give us the indications to run your projects.
8. Please, contact us if you have any doubt about this test.

Expected Results

- A folder with your name.
- Within the folder, put all the source code generated.
- Within the folder, add a Readme.txt file to explain how to run your project.