TECHNICAL TEST - FULLSTACK DEVELOPER

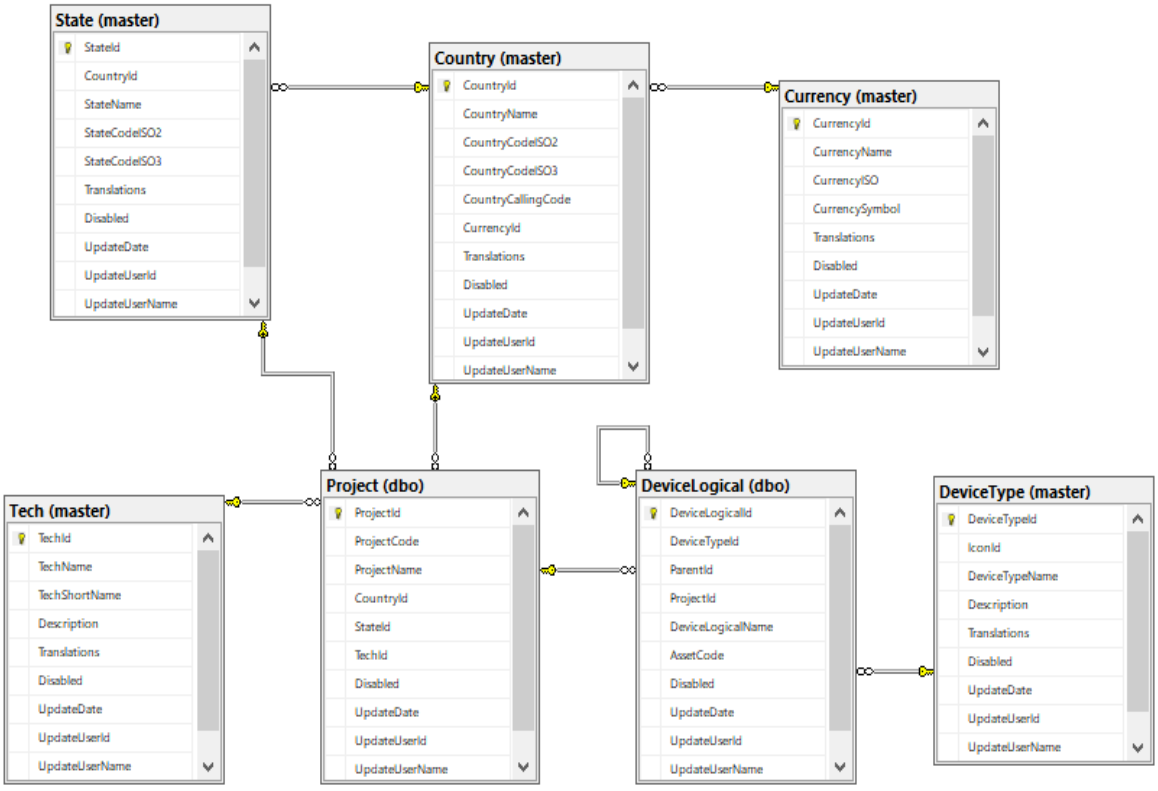
This technical test is designed to evaluate the candidate's skills in C# language and angular framework. The test will consist of implementing a REST API using an existing data model in database and use it as data provider for the client application

Instructions

- The time to execute the test is around 1,30/2h
- The test will be implemented using Visual Studio 2019, Visual Code and .NET Core 3.1
- Last LTS node version
- In the same solution folder, you can find a json to import new collection into postman to see the REST API requests to implement
- You can see the data model using Microsoft SQL Server Management Studio with the next connection parameters:
    - Server: qbi-dev-sql-server.database.windows.net
    - DataBase: TechnicalTestDB
    - User: TechnicalTestUser
    - Password: 7wACDD4%55?pM%7P
- The solution of the test has three projects with the next structure:
    - Core
        - Application layer
            - This layer contains business validation rules
            - It connects WebApi layer with data layer
        - Data layer
            - It contains repositories to access data
    - WebApi
        - REST API project
    - UI
        - Angular client application

Test

Based on the following database model:



1. Change the necessary code to return country, state and tech details in the projects request (GET/project). The response should be something like this:

```
{
    "projectId":1,
    "projectCode":"PROJ-A",
    "projectName":"PROJECT-A",
    "country":{
        "countryId":1,
        "contryName":"Spain"
    },
    "state":{
        "stateId":2,
        "stateName":"Madrid"
    },
    "tech":{
        "techId":1,
        "techName":"Solar PV"
    }
}
```

2. Create REST API methods for Device entity (including all the structures you need to implement the desired solution):
    2.1. Get Devices: This method should return all devices stored in database. Additionally, you can add a parameter to filter the query by “DeviceType” field.
    2.2. Get Device: This method should return a concrete device with all possible fields.

2.3. Create Device: This method should create a new Device in database.
2.4. Update Device: This method should update an existing Device in database.
2.5. Delete Device: This method should remove an existing Device in database.

3. The steps you must follow for the angular test are in the home page of the client application