CODING TEST

Sr. Backend Developer Candidates

Dear Candidate,

Congratulations! You pass the 1st interview session.

And since we are looking for a great Sr. Backend Developer candidate to join our team, we want you to showcase your technical skill into the real code. So please read all the instruction and requirements below.

MAIN INSTRUCTION

Create a simple product and upload your project to GitHub, Gitlab, Google Drive then send us the link so we can download it and review it (send the link by replying the email sent by our HR). The timeline for this coding test is three days since you got the email from us.

After we got your link, our Technical Lead will review your code and if we want to proceed further with you, you will be invited to a final interview with our Technical Lead so we can know more about you and you can also present/explain your code or your technical skill. Write documentation about how to configure the apps needed to be run, test case about how to run or call the API.

BUSINESS CASE: TENDER MANAGEMENT SYSTEM

Develop an API which will be served as a back-end point for the any frontend/UI (c# winform, Angular SPA application, React js, etc) to manage the Tender details within the application. The API should provide the functionalities/end points which can provide below functionality:

* Create a tender
* Edit a tender
* Delete a tender
* Show the tender list
* View a tender detail

VALUE ADD

Please treat this exercise as a showcase of not only completing the project to the best of your ability but adding your own flair and spin on the development in whatever way you choose fit.

TECHNOLOGY STACK

Tools

* Visual Studio 2019/Visual studio code
* Microsoft SQL server

Architecture

* + Preferable using Clean Architecture, but any N-Tier architecture will do
    - Architecture should consider having a separation of concern.
  + Dependency injection Database
  + Microsoft SQL server, PostgreSQL, etc.
  + If you are using SQL
    - Write stored procedures for CRUD operations. (If possible)
    - Write table creation in SQL
    - You can user EF core 2/ Dapper/ ADO.net to call stored procedure.
  + Using Cosmos DB
    - Can connect with local storage emulator.
    - Create Table on the fly, if it’s not exists.

Backend Language

* + Asp.net Core Web API 3.1/5
  + User Attribute routing

ASSESSMENT CRITERIA

Here is some of the things we would like you to show case inside your coding: Web API dotnet core

* + Middleware
  + Filters (Validate Attribute on Request)
  + RESTful API
  + Authentication (JWT Token)
  + Logging
  + Swagger
  + Handling Error
    - SeriLog
    - Middleware to handle all the errors
  + Health Check End point
  + Configuration Loading (AppSettings)
    - Handling user secrets
    - Handling Different environment app settings
      * Staging/dev
  + Swagger end point

Testing

* Any testing tools such as NUnit, XUnit, moq

SCOPE OF THE PROJECT

1. Intro

The REST API should have the ability to Create/Edit/Delete/View Tender details. Also, the API should have an end point which can return list of tenders.

Additionally API can have an Account Api to return JWT token based on the login details. Must have End points

* + Create/Edit Tenders
  + Show the List of Tenders Created
  + Account API to get the JWT token.

\*All other functions are optional.

1. Log in Mechanism

Provide an API end point to provide Authentication token. User JWT token. And the token should carry the user details. (UserID/UserName)

Successful Authentication should be based on

* + Email Address
  + Password

You don’t need to have a registration end point, just create a table with user login details. And insert few records for the successful login. You can hard code the user login details on the repo class.

1. Tender List – Main Page

This API end point should return below details as a list. Tender ID

* + TenderId
  + Tender Contract Number
  + Tender Name
  + Tender Release Date
  + Tender Closing Date VALUE ADD

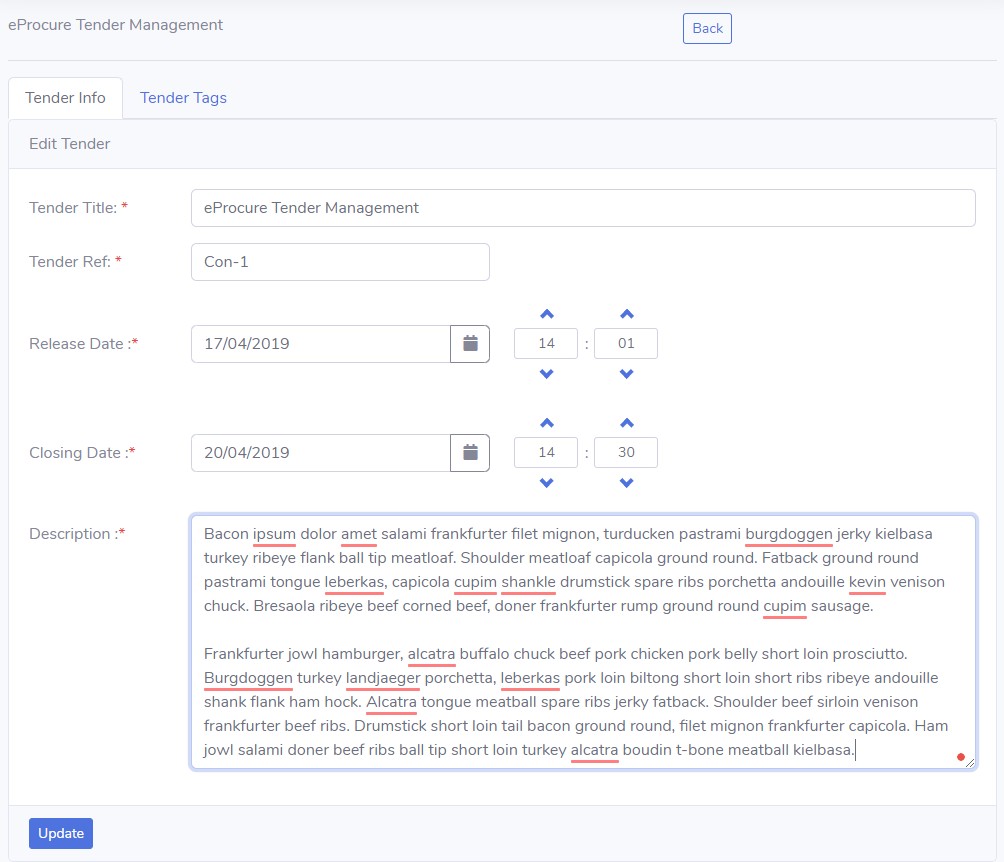
Having, Paging/Search query can be bonus as part of the list

1. Tender Details – View/Edit/Create Page

Please find the tender details below which will be used to create a tender. The API end point supposed to capture below details.

* + Tender ID
  + Tender Name
  + Tender Reference number.
  + Tender Release Date
  + Tender Closing Date
  + Tender Details
  + Tender Creator ID / (UserID) (This can come from JWT Token) Validation:
  + All the fields are required
  + Release Date and closing date should be on a future date
  + The closing date should be greater than the release date

*Example*



Good luck! :D