Consultiv Utilities Web Developer Coding Test

Introduction

This coding test is intended to test the following skills:

* C#
* ASP.NET Core
* Entity Framework Core
* MVC
* HTML / CSS
* WebAPI
* Unit Testing

The supplied solution consists of a simple MVC front end which makes calls to a Web API to process customer information.

You will need to load the solution into Visual Studio 2019. The free community version is available at: <https://visualstudio.microsoft.com/downloads/>.

The solution uses an in memory database so no database server is required.

When you first run the solution the home page should display the message “Hello Username from the API”. This indicates that the UI and API are running and communicating correctly. If you don’t see this message ensure that both the UI and API projects are set as startup projects.

Test Format

You should spend around a hour on the test. It consists of a number of tasks to modify the existing code and a number of questions.

It is not necessary to complete all the coding tasks, but please answer all of the questions.

Coding Tasks

The existing code can be improved, please bear this in mind when working through the tasks.

Please add comments to any code you change or add.

* Improve the validation and formatting of the customer data in the UI.
* Change the sort order of the customer list to be by name and only show active customers.
* Add IncorporationDate (date) and Turnover (numeric) to the customer. Update the form accordingly. IncorporationDate should be mandatory.
* Add functionality to view the customer’s details.
* Add functionality to enable the user to amend the customer details.
* Add a search facility to the customer list. Name and company registration number, of active customers, should be both checked.
* Add any error handling and validation to the code.

Questions

* Please list in English the unit tests you would create for the UI Customer Controller methods.
* If the customer table contained a million records how would you improve performance of the search?

Paging the results.

* What suggestions would you make to improve the supplied solution?

Maybe add model mapping like automapper?

* Could unit of work be used to improve this solution?

With only one table probably not. If another table were to be added then the unit of work pattern could group up queries into one transaction or if multiple customers were added at the same time via an import for example.