Technical Test

You should only need to spend 1-2 hours on this test

Things you need to do

- Create a simple SQL database, with one table named "Items" to hold the data types found in the Item.cs model
- Insert some dummy data into the newly created Items table
- Upon a request to the GetItems() method on ItemController, you should
    - Retrieve the items from the database, paginated in page sizes of 10
    - Return the data as a List
- Write a test to assert that only 10 items are retrieved at a time
- Write a test to assert that requesting the next page of data, returns a different set of 10 Items
- Refactor the API to be "more testable"
- Refactor the API in any way you see fit, to improve the readability, structure, security (or any other reason you may think of

Rules

- There are no rules, do what you think best
- You are free to use any testing framework you like
- You are free to use an ORM if you like
- You are free to use any mocking frameworks you like

What we are looking for

- First and foremost, follow your gut. Don't worry too much about what we might be expecting to see, the biggest part of this test, is how you explain your reasoning in a technical discussion.
- We are looking for someone who can make changes, and explain why in their opinion, these changes are valuable.
- We are looking to see how you like to structure and lay out your code. We're open to anything as long as you can justify it.

Bonus point

- Build an Azure DevOps pipeline to build and test the solution.

Once complete,

please send us the following to ratan.kalwa@segen.co.uk

- Zip file of your code - Copy of the database you've created, either as a backup file or scripted out (including data)