INSTRUCTIONS

In this challenge, you will be asked to create and work on some web APIS. The idea of this challenge is to measure how quickly you can learn new tools (if you do not know them) and how clean and simple is your code.

These challenges are not meant to stress you out, so keep your solutions as simple as possible. You will have what we consider enough time to submit your answers. However, if you need more time, please let us know in advance so we can work it out.

Communication from your end is very important so, any questions are allowed, feel free to ask anything you need.

Once you are ready to deliver your solution, upload it to a public GitHub repository where we can review your code. Send via email the GitHub URL to the recruiter who sent you this challenge.

Front-end (ReactJS)

- Create a ReactJS application that leverages the Bing search API.
- Must take a search input and display a list of results.
- Post the solution on GitHub.

Tips:

- Keep your code and solutions as simple as possible.

- You can use bootstrap with create-react-app or build it from scratch. You will be asked why you used the solution you chose later in the selection process.

- Try to use input validation. You will be asked whether you use it or not later during further interviews.

- Our team will do a Code review (code style, organization, naming conventions, declarations, commenting). Keep this in mind when writing code.

- Try to implement source control. Our team will review commit messages.

- Our team will review your UI Styling.

- Try to use typescript as much as possible.

- Try to use any dependency libraries. You will be asked whether you use it or not later during further interviews.
- Include, if possible, any extras you consider valuable to your solution: UI Tests, Deploy your solution to Azure, etc.

- The goal is to check whether you can code up or not an API-driven application.

- In a later technical interview, you might be asked how you can enhance/improve your solution via screen share. Areas of improvement include UI, build time, bundle size, adding a new feature, etc. Keep this in mind for the technical interview.

Backend (.Net)

- Create a Web API that will result in JSON response for locations with availability between 10 am and 1 pm.
- You will have to create your own list of at least 10 locations with different time availabilities. For the locations think about different businesses such as a Pharmacy, a Bakery, a barber shop, a supermarket, a candy store, a cinema complex, etc. You can also use any other location you want. Feel free to be creative.
- Post the solution on GitHub.

Tips:

- Keep your code and solutions as simple as possible

- The goal of this challenge is to check if you can code up a GET endpoint and properly formatted responses.

- Our team will do a Code review (code style, organization, naming conventions, declarations, commenting). Keep this in mind when writing code.

- Try to implement source control. Our team will review commit messages.

- Our team will review how you managed locations and times. Slurped up via c#, imported to database, copy-paste, etc.

- Our team will review if the API does return proper HTTP codes

- Our team will review how is the response object formatted.

- Our team will review what would be the JSON response if nothing is available.

- Try to perform unit testing and leave evidence.

- Some nice to haves that would give you extra points could be:
  - endpoint to add location.
  - Deploy the solution to azure.
  - Create your own CSV file with times and locations and pull it from filestream or database

- In a later technical interview, you might be asked how you can enhance/improve your solution via screen share.