Good afternoon, thank you for participating in our coding challenge.

As was noted, it is not expected that you will be able to complete this project. We want to know how far you can get in 1 to 1.5 hours, and during a face to face possibly work more on the project in person to get an idea of your work process.

You are welcome to return it however you like, we typically receive a compressed file back but it's very much up to you; a github posting is perfectly fine as well. Please use any language or tools you like.

Give me a call if you run into any serious issues.

Thanks, and good luck,

Mubin Khan

310-308-8999

For each of these steps, feel free to use libraries like AngularJS and jQuery if you would find them helpful.

If you don't know how to accomplish something, you're encouraged to research possible solutions. This is a time limited test so it’s recommended you seek to accomplish as much as possible vs. creating an actual viable product.

1) Pull 1000 names from some public API either in front-end code or back-end. A couple possibilities would be Facebook's Graph API (https://developers.facebook.com/docs/graph-api/reference) and LinkedIn's API (<https://developer.linkedin.com/docs/rest-api>). If you are unfamiliar with OAUTH it’s recommended you skip these options and seek an API that does not require it such as <https://randomuser.me/>

2) Store the results from step 1 locally (we suggest a SQLite database, but you can use any format you'd like). This should be run on the back-end (i.e. imagine this is running on a server where we need the data on a database on the server).

3) Develop a simple webpage that will display 100 names in a table (out of 1000).

4) Add email and phone columns to the webpage table. Then, provide a button for each table entry that allows you to manually enter a phone and/or email address. Add some basic validation for the phone and/or email address (e.g. '123' would be rejected as a phone number). Validation can be run on either front-end or back-end. The data should be stored in storage you set up in step 2.

5) Add buttons that allow you to sort the table by name or email address.

6) Add a search function to the table.

7) Add a button that allows you to export the table data to a .csv download file. The .csv file should reflect the current status of the table (including sort order and search results).

Please note that Gmail and many compressed files generate an issue. It is recommended you use another format or provide a google drive link to get the file to us.