**Bookatable Automation Engineer technical test**

Bookatable offer a range of electronic diary solutions for restaurants, allowing them to capture bookings and their associated covers (party size). The first two questions are based on one of our solutions.

1) We would like to understand how you would go about automating a given piece of functionality. In the language of your choice could you write an Acceptance test to automate the following steps?

NOTE:

Please ensure that you provide relevant documentation/a readme that will allow ANYONE to run the tests on a fresh machine, i.e. you cannot assume that the user running the test has all the tools you have used, using your own judgement, put in the amount of detail that you think is required to meet this criteria.

WARNING: Please do NOT create a new restaurant as if you do you may be contacted by our sales team

i) Open the following URL

https://erb.bookatable.com

ii) Log in as

Username: **See Email**

Password: T3ster

iii) Open Menu (Ham Burger Menu in top left)

iv) Expand Venue Settings

v) Open Services

vi) Add Service

2) The following is an example story which has been implemented in the past, can you describe how you would break this down within your team. What sort of information would you expect to be in the story

As a restaurateur

I want to set up services (dining period)

So that diners can make bookings for the times I am open

I would break it down according to different types of restaurants, different dining periods, bookings for different opening times, trying to book at a time that isn’t available/open.

3) Given the following Rest request, can you explain how you would ascertain the "short\_name" for country

http://maps.googleapis.com/maps/api/geocode/json?address=39+York+Road,+London,+SE1&sensor=false

Look Under

"types" : Country

The Short\_Name for that type is where I would look.

4) As a business we are trying to move towards continuous delivery/deployment. Can you give some example factors that should be considered for a CD environment

Have to think about the following:

* What will be your testing strategy, so process, tools, resources etc.
* QA and Dev VM Envionments controlled by QA/Devs so that there is no reliance on IT
* Which version control tool to use i.e. like Git
* Which automated build and software tool to use i.e. Jenkins, Teamcity
* What process will you use to merge code changes, manual or automated (using a continuous integration server)