

carpenters group

Interview Technical Test

For this test you will be using C#, HTML5 and JavaScript to build a web application for a restaurant to manage table bookings. The template solution provided to you is set up to use Angular, however you are free to use any other JavaScript framework or additional libraries you like.

Each task is designed to get progressively harder in order to challenge anyone from a graduate to 2 years' experience. Therefore, completing the entire test is not mandatory or expected for all candidates.

Please spend no longer than 48 hours on this test. If you run in to any problems or have any questions, please email: dtho@carpentersgroup.co.uk

Tips to getting started:

The template solution contains two Web API controllers: "Alive" and "Booking". When you run the project the browser will open to <http://localhost:52363>. This will make a call to the alive controller and you'll see "Hello World!" displayed – that's how you know the project is running correctly.

If you make a request to <http://localhost:52363/booking> then the booking Web API controller will hit the GET method – and return all bookings.

If you navigate to <http://localhost:52363/Index.html#> your index.html will be hit, and your UI will be displayed.

Create Booking				
Contact Name	Contact Number	No of Diners	Table Number	Booking Time
David Thompson	07826374821	4	1	25/03/2019 01:00
James Gartland	07326372821	1	3	25/03/2019 01:00
Stephen Hewitt	07826374521	3	2	25/03/2019 01:00
Richard Blears	07226374521	10	4	25/03/2019 01:00
Colin Hughes	07826124521	6	3	25/03/2019 01:00
James Edgson	07126124521	7	1	25/03/2019 01:00

1. Using JavaScript, make a request to the Web API controller to GET all bookings and display the results within a table;
2. Order the table by booking time;
3. Display bookings with more than 6 diner's as red;
4. Display bookings with only 1 diner as blue;
5. Add a button to the top of the table;
6. When the user clicks the "New Booking" button, navigate to a Create Booking page;

Create Booking

Contact Name

Contact Number

Number of People

Table Number

Booking Time

Save

7. Add a Create route on the API which will call the existing CreateBooking service method. Modify this service method so when the user clicks Save, the new booking is added to the bookings list;

Note: New bookings should have a unique BookingId, which is greater than the maximum existing BookingId. I.e. if you have 3 existing bookings with ids: 1, 3 and 7, the next booking Id will be 8.

Table Number

1

Contact Name

David Thompson

Contact No

07826374821

No of Diners

4

Date

25/03/2019 13:00

Submit

8. On a successful save, the page should automatically navigate to the bookings page;

Create Booking

Contact Name	Contact Number	No of Diners	Table Number	Booking Time
David Thompson	07826374821	4	1	25/03/2019 01:00
James Gartland	07326372821	1	3	25/03/2019 01:00
Stephen Hewitt	07826374521	3	2	25/03/2019 01:00
Richard Blears	07226374521	10	4	25/03/2019 01:00
Colin Hughes	07826124521	6	3	25/03/2019 01:00
James Edgson	07126124521	7	1	25/03/2019 01:00

9. Run the unit tests for the Web API project;

Failed Tests (8)

- ❌ CreateBooking_InvalidTableNumber_NegativeNumber_ThrowsArgumentOutOfRangeException
- ❌ CreateBooking_InvalidTableNumber_TooHigh_ThrowsArgumentOutOfRangeException
- ❌ CreateBooking_InvalidTableNumber_TooHigh_ThrowsArgumentOutOfRangeExceptionWithMessage
- ❌ CreateBooking_InvalidTableNumber_TooLow_ThrowsArgumentOutOfRangeExceptionWithMessage
- ❌ CreateBooking_InvalidTableNumber_Zero_ThrowsArgumentOutOfRangeException
- ❌ CreateBooking_NullBooking_ThrowsArgumentNullException
- ❌ CreateBooking_Valid_BookingAddedToBookings
- ❌ CreateBooking_Valid_BookingIdIncrementsMaxId

Modify your CreateBooking service method so that all unit tests pass.

▲ **Passed Tests** (8)

- ✓ CreateBooking_InvalidTableNumber_NegativeNumber_ThrowsArgumentOutOfRangeException
- ✓ CreateBooking_InvalidTableNumber_TooHigh_ThrowsArgumentOutOfRangeException
- ✓ CreateBooking_InvalidTableNumber_TooHigh_ThrowsArgumentOutOfRangeExceptionWithMe
- ✓ CreateBooking_InvalidTableNumber_TooLow_ThrowsArgumentOutOfRangeExceptionWithMe
- ✓ CreateBooking_InvalidTableNumber_Zero_ThrowsArgumentOutOfRangeException
- ✓ CreateBooking_NullBooking_ThrowsArgumentNullException
- ✓ CreateBooking_Valid_BookingAddedToBookings
- ✓ CreateBooking_Valid_BookingIdIncrementsMaxId

10. Finally, allow users to edit an existing booking. I.e. User clicks on James Gartland's row;

James Gartland	07326372821	1	3	25/03/2019 01:00
----------------	-------------	---	---	------------------

An "Edit Booking" page is displayed.

Table Number

3

Contact Name

James Gartland

Contact No

07326372821

No of Diners

1

Date

25/03/2019 13:00

Submit

The user modifies the booking and clicks save. On a successful save, the page automatically navigates to the table view where the booking has been updated.

Create Booking				
Contact Name	Contact Number	No of Diners	Table Number	Booking Time
James Gartland	07326372821	1	3	25/03/2019 01:00

From here you're free to expand and demonstrate any skills you like. Some suggestions:

- Replace the JSON file with a database
- Improve page styling
- UI Unit Tests (i.e. Jasmine)