

SO YOU KNOW HTML, CSS & JS?

SHOW US YOUR SKILLZ

From time to time in London you'll need to catch a bus. Some lucky bus stops have live departure boards, but not all do. To help, we'd like you to build an app to show a live departure board for any bus stop in London.

The task is designed to assess your interest and experience in front-end web development. We want to see your code, your approach, and your talent.

WHAT YOU NEED TO BUILD

The minimum requirements for the bus departure board app are simply defined:

1. Make sure there's a map, which lets you find any stop in London
2. Make sure a user can select a specific stop, and see its imminent departures

That's it. We leave the design & user-experience to you.

WHAT WE'RE LOOKING FOR

In your app we'll be looking for:

- Completeness of solution - *does the app work as per the requirements?*
- Quality of code - *is your code clean & well written?*

If you can't complete your app, give us a description of what you were planning / how you would approach it.

GETTING STARTED

To get started, just go ahead and create your HTML, CSS, JavaScript files.

For your map, we recommend checking out the [Google Maps API](#) although you can use a different maps service if you wish.

If you're going to use an AJAX library, stick with [jQuery](#).

The bulk of your work will be finding bus stops and bus data. We've written a simple [Bus stop API](#) using the [TFL bus data](#) that will cover everything you'll need for your app.

WHEN YOU'RE DONE

We expect most people to spend between one to four hours on this task, although you're free to use more time if you want.

Once you're done, you'll need to zip your code and send it to us, so make sure it can work when unzipped.

BUS STOP API

The bus stop API consists of the following resources:

1. [/bus-stops](#)
2. [/bus-stops/{bus-stop-id}](#)

All requests support [JSONP](#) via the 'callback' parameter (eg.

`?callback=foo`).

GET /bus-stops

Return a list of bus stops within a given area (a bounding box defined by two opposing points of a rectangle).

Required parameters

- `northEast={lat,long}` - the top-right geographical point of a bounding box, where lat and long are decimal values.
- `southWest={lat,long}` - the bottom-left geographical point of a bounding box, where lat and long are decimal values.

Example URL

```
/bus-stops?  
northEast=51.52783450,-0.04076115&southWest=51.51560467,-0.10225884'
```

200 Response (ie. a successful request)

```
{  
  "markers": [  
    {  
      "id": "77994",  
      "name": "Aldersgate Street / Goswell Road",  
      "stopIndicator": "BX",  
      "lat": 51.52363797159915,  
      "lng": -0.09749245212910045,  
      ...  
    },  
    {  
      "routes": [{  
        "id": "55",  
        "name": "55"  
      }]  
    },  
    ...  
  ]  
}
```

400 Response (ie. an error)

```
{  
  "errorMessage": "Missing parameter. Please ensure you have both  
'northEast' and 'southWest' parameters defined."  
}
```

GET /bus-stops/{bus-stop-id}

Retrieve a list of bus arrivals for a single stop

Required parameters

There are none.

Example URL

```
/bus-stops/58382
```

200 Response

```
{
  "arrivals": [
    {
      "routeId": "K3",
      "destination": "Roehampton Vale",
      "estimatedWait": "3 min",
      "scheduledTime": "16:27",
      "isCancelled": false,
      ...
    },
    ...
  ]
  ...
}
```

400 Response (ie. an error)

```
{
  "errorMessage": "There was an error retrieving bus stop information.
Please ensure you use a valid bus stop id"
}
```

FINAL WORDS

We hope you enjoy building your bus departures app. We're interested in your feedback, so do let us know what you thought of the task.

And most finally, good luck!



DigitasLBi