

Chris Hardcastle

Technical test entry

About

This work is the result of my answer to the technical test as briefed in the instructions file.

I spent half a day planning my approach and how it could be run with minimum configuration. I spent a very enjoyable day writing the tests, then the implementation for the work along with the front end.

I was tempted to push jQuery out in favour of ReactJS. However the NPM based set up can be daunting if you're not familiar with it; so I settled on jQuery with an aim to keep things extremely simple and easy to follow.

Some complexities and performance enhancements have been overlooked in favour of time and simplicity. These are:

- Some repeated functionality between classes that could have otherwise shared via an abstract class titled "Person"
- Some AJAX heavy calls to populate the lists with data. The best thing about React is: the DOM for expired data only would have otherwise been updated instead of the DOM for both of the lists!
- I wanted to provide a remove from favourites / contacts feature but it wasn't in the instructions.

Set up

For convenience I've this test up to run on PHP's built in mini web server.

There's a front end and backend. In the absence of any tokens or authentication: the backend allows cross domain requests. This is because both sides should be kicked off in isolation as you'll see within the commands below.

If you're unable to run the project, please include the screenshots within the README folder as part of your review.

Build instructions

1. Navigate to the document root of the project on your computer

```
$cd [DOCUMENT_ROOT]
```

2. Open a PHP server session against the backend of this project using the router.php (api) file.

```
$php -S localhost:8000 router.php
```

3. Open another session for the front end

```
$php -S localhost:8800
```

Test instructions

Install and run composer in the same document root directory, this provides the means in which the PHPUnit tests for this project can be run.

Run PHPUnit over the tests directory. At this present time of writing: 4 tests, 13 assertions passed in: 504 ms, using 16.00MB of memory

Any feedback is gratefully received. Please visit www.chrishardcastle.co.uk in support of your consideration over me for your team.

Thanks for your interest in my work and taking a look at technical test result.