

Spotlight Front End Developer Interview - Technical Tests

Provide partial or full solutions to as many problems as you can, **in any order**.

Take up to 90 minutes to complete the test. You are not expected to complete all questions.

You may use Google, MSDN, or any other resources you would normally use whilst developing software. Be prepared to discuss and justify your solutions.

Instructions:

- You ***must*** attempt question 1: Recipe
- Please attempt ***one*** of questions 2, 3 or 4
- Then attempt as many of the remaining questions as time allows (***you are not expected to attempt all of them!***)
- You ***must*** create a new GitHub repository, commit all of your code to that repository and send us a link to the repository
- You ***must*** deploy your working solutions to a publicly-accessible web host (GitHub Pages, jsfiddle, codepen, or your choice of hosting) and send us the URL. We should be able to review your work by opening a web page in a web browser and following links to each of your submissions.
- You ***must*** include a text file that contains a JSON object describing yourself. You may add anything you wish to the object, but it should be valid JSON.

We will use the deployed web site to assess the design, behaviour and user experience of your solutions. We will use the GitHub repository - including commit messages, file timestamps and revision history - to assess the quality of your code and your development style, and to verify that you completed the test within the 90-minute time limit.

Some questions are harder than others; each question has a point score according to difficulty. Candidates are graded on total overall points.

By taking this test, you are agreeing not to share the contents or calculations with other agencies or candidates, nor will you publish it elsewhere. Failure to comply with this will immediately invalidate your application (and we do tend to find out about these things!).

Thanks and Good Luck!

Question 1: Recipe (20 points)

There is a PNG in the "RecipePage" folder of the ZIP file you have been sent. Please recreate this page using HTML and CSS/SCSS/Less. Try and match it as closely as possible. Extra points are awarded for clean, semantic markup, pragmatic use of CSS, and a responsive design that works on mobile.

Question 2: Javascript functions (12 points)

This task involves filling in the function bodies of some Javascript methods. There is a folder called "JavascriptTests" in the the ZIP file you have been sent. This contains the Jasmine Unit Test runner,

which verifies the behaviour of the functions. When you open the page **tests.html** it will show you that many of the tests fail. You need to fill in the function bodies so that the tests pass.

The **Functions.js** file is the one you should modify to complete the test.

You can look at the **Tests.js** to see the expected behaviour of each function.

Refresh the browser window that has **tests.html** open to re-run the tests against your latest javascript.

DO NOT MODIFY ANY EXISTING TESTS.

Question 3: ES6 and beyond feature demo (14 points)

Create a webpage that demonstrates the new features in ES2015/2016/2017. You must include at least 3 features, and feel free to choose the ones that you find most helpful or interesting. The web page should explain what the feature does and include a live interactive demo of the feature in action.

Question 4: Tic-Tac-Toe (20 points)

Implement a two-player "noughts and crosses" (aka "tic tac toe") game that runs in a web browser. Both players will sit at the same computer – we're not looking for web-sockets or anything. Use any framework that you like; React, Angular, Backbone, Bootstrap, jQuery, etc

The game starts with a blank 3x3 grid

- Players take it in turns to make a move.
 - When starting a game, either player may start the game by making the first move; once the game is in progress, the game should force players to take turns.
- Players make a move by clicking a blank grid square; this will mark a **O** or **X** in that square as appropriate. Once a move is made, it cannot be taken back.
- The first player to claim three squares in a row (horizontal, vertical or diagonal) is the winner. When this happens, your program should inform the players who has won, and allow them to start a fresh game.
- If the game reaches a point where neither player can win, your program should inform the players that the game is a draw, and allow them to start a fresh game.

You do not need to implement an "AI" computer player. Your web page should provide the game board and act as "referee" but will not be expected to make any moves.

Question 5: Webpage bug Hunt (15 points)

There is a folder called "BuggyPage" in the the ZIP file you have been sent. It contains a webpage called "page.html", which has a number of issues that make it ugly and unusable. Fix these and any other problems that you find.

You will need to change **page.html** and the **styles**. You may modify either the SCSS or the CSS. If you choose to use SCSS you will need to generate the CSS.

- Intro text should not all be bold

- Article grid should line up correctly
- Nav links should not be underlined
- Category links on articles should navigate to the category page
- Heading font should be better, choose anything that isn't bog standard sans-serif/serif
- Fix the missing footer paragraphs
- Make it look better on mobile-sized screens
- Fix any other issues that you notice

Question 6: Responsive images (8 points)

Create a responsive webpage that showcases an image, e.g in a gallery. You should use the correctly sized image for the device, whilst also showing a better image for devices that are capable of showing higher resolutions (e.g. retina displays).

The different renditions of the image that are available are shown below:

2048 pixels wide:

https://res.cloudinary.com/spotlightuk/image/fetch/w_2048,c_limit/https://spotlight-cmsmedia.s3-eu-west-1.amazonaws.com/media/1192/opened-space.png

1536 pixels wide:

https://res.cloudinary.com/spotlightuk/image/fetch/w_1536,c_limit/https://spotlight-cmsmedia.s3-eu-west-1.amazonaws.com/media/1192/opened-space.png

1024 pixels wide:

https://res.cloudinary.com/spotlightuk/image/fetch/w_1024,c_limit/https://spotlight-cmsmedia.s3-eu-west-1.amazonaws.com/media/1192/opened-space.png

768 pixels wide:

https://res.cloudinary.com/spotlightuk/image/fetch/w_768,c_limit/https://spotlight-cmsmedia.s3-eu-west-1.amazonaws.com/media/1192/opened-space.png

640 pixels wide:

https://res.cloudinary.com/spotlightuk/image/fetch/w_640,c_limit/https://spotlight-cmsmedia.s3-eu-west-1.amazonaws.com/media/1192/opened-space.png

320 pixels wide:

https://res.cloudinary.com/spotlightuk/image/fetch/w_320,c_limit/https://spotlight-cmsmedia.s3-eu-west-1.amazonaws.com/media/1192/opened-space.png