**Front-End Assignment**

**Version history**

|  |  |  |  |
| --- | --- | --- | --- |
| **Version** | **Date** | **Author** | **Remarks** |
| 0.1 | 08/01/2015 | Mo Morgan | Draft for comment |
| 1.0 | 20/01/2015 | Mo Morgan | Initial release |
|  |  |  |  |

**Contents**

Overview 2

Assignment 2

Brief

Timings

Technologies

Asking questions

Submitting your response 4

Assessment criteria 4

Glossary of terms 5

# Overview

This document outlines an assignment designed to allow front-end developers to demonstrate their experience and expertise. As well as challenging craft skills, the assignment also presents an opportunity to show capability in the management of timing and technical debt, as well as code maintainability.

# Assignment

## Brief

A new website is being developed on behalf of a client. The project has already undergone an extensive research phase to understand user behaviours. The user experience and design teams have completed a piece of work to devise a section of content navigation functionality that follows the client’s brand guidelines and is also straightforward for users.

Based on the wireframes the designers have prepared two mock-ups, which now need to be built so that further user testing can be conducted. Both mock-ups show the same content navigation device; one is the desired layout for handheld devices such as smartphones, the other for larger screens.

|  |  |
| --- | --- |
| Mobile | Tablet; Desktop; TV; etc. |
| [Macintosh HD:Users:momorgan:Dropbox:Hogarth:fecodetest:mobile.png](https://dl.dropboxusercontent.com/u/145601/hww/fecodetest/mobile.png) | [Macintosh HD:Users:momorgan:Dropbox:Hogarth:fecodetest:desktop.png](https://dl.dropboxusercontent.com/u/145601/hww/fecodetest/desktop.png) |

*Both of these mock-ups are available from us at full size. If you don’t have them, click the images above or contact us.*

There are five navigational elements, each associated with some content. In the narrower layout, the navigational elements function as an accordion: below the active element, the associated content is displayed. In the wider layout, the navigational elements function as tabs, and the content associated with the active element is displayed to the side.

Only the content associated with the active navigational element is shown at any one time, and the first element is active by default.

The active navigational element is indicated by a change in background colour. In the narrower layout, plus and minus icons are used to denote that elements may be expanded or collapsed respectively. In the wider layout, these symbols are replaced with chevrons to indicate their behaviour.

So, the brief is to realise the mock-ups into a single webpage so that users with a smartphone-based browser and a desktop browser can test them. It is likely that the user testing will bring about multiple rounds of amends, so your code should be as tidy, well-formed and explicit as possible for the benefit of the next developer to work on it.

## Timings

Once you have studied this document in detail, you must supply to us a firm deadline by which you will have submitted your response. You should also provide a short rationale for this deadline.

You may evaluate the assignment as if it were a client brief, and calculate your timings accordingly. Faster or slower is not necessarily better; we are interested in the balance between execution and delivery.

Should your proposed timings be significantly different from our expectations, we may ask you to revaluate or provide further justification.

## Technologies

Your response to the brief must include HTML5, CSS3 and JavaScript. You may use libraries and frameworks of your choosing, or you may elect to code from scratch. If you do incorporate a library or framework, you should be able to demonstrate your efforts to avoid unnecessary page weight.

You should provide a short rationale for the choices you make. You may also give examples of approaches you considered but decided not to pursue, again with a short rationale.

## Asking questions

If you need to, you may ask us questions before or during the assignment and we will help where we can. Note that the brief has been prepared in a way that will allow a certain amount of interpretation, so there may be some questions that we will not answer.

# Submitting your response

Your submission must address the brief in full.

You must supply your response as a compressed archive, containing the source files you have created. You should also include any deployment instructions, and any additional documentation you have created.

You must also specify the mobile and desktop browsers in which your submission should be reviewed.

You should also host your solution and include the URL (and any access credentials) in your submission. Your solution must not be publically accessible, so please do not share these details other than in your submission.

# Assessment criteria

An experienced member of our technical team will review the code that you submit in your response. They will read any documentation you provide, and will study the code both syntactically and as it renders in browser.

The reviewer will be looking for efficient and elegant code, with attention given to semantics and naming. Any comments you have added to the code should help the reviewer to grasp your solution quickly. If you have introduced frameworks or libraries, they will be looking to see how efficiently they have been applied. They will also note whether their inclusion, or not, has made it easier or harder for another developer of equal standing to execute amends to your solution, to build on top of it, or to integrate it into something else.

In the browser, the reviewer will compare how your solution renders against the mock-ups provided. It is assumed that the mock-ups prepared by the designers are an exact representation of how the solution should render.

We may also want to discuss your submission with you; no further preparation on your part will be necessary for this.

# Glossary of terms

For the avoidance of doubt, the words *must*, *must not*, *should* and *should not* and *may* are to be interpreted in accordance with the following definitions:

* *Must* means an absolute requirement.  
  Example: to obtain a driving licence you must pass a test.
* *Must not* means an absolute prohibition.  
  Example: if your licence is revoked, you must not drive.
* *Should* means a recommendation that may be ignored in particular circumstances but the full implications must be understood.  
  Example: you should eat five portions of fruit and/or vegetables per day.
* *Should not* means not recommended, although acceptable in particular circumstances but the full implications must be understood.  
  Example: you should not eat too much cake.
* *May* gives permission to perform a particular action, rather than to express a possibility where “can” or “might” are used instead.  
  Example: you may choose which route you use to get from A to B.