Wanna be a Front-end Engineer ?

Prove it.

Write a simple quiz app with the spec below. The exercise below isn't meant to take more than a day or two, so plan around that.

Ship it.

Once you're done, send us a link to the GitHub repo as well as the running app (how you deploy it is up to you). We'll review based on accuracy, best-practice, style, and... attitude.

Technical Requirements

1. Use
   a. GitHub
   b. AngularJS
   c. Jasmine
   d. Bonus points for use of HAML and Coffeescript
2. Avoid
   a. jQuery (rely on AngularJS' JQLite where possible)
3. Organize project in a way that makes sense to you.
4. Provide sufficient test coverage for all Javascript.

Product Requirements

1. Allow user to
   a. register (email & password)
   b. take quiz
   c. submit quiz
   d. view results
   e. view correct answers
   f. logout

Quiz Content

1. Which is not an advantage of using a closure?
   a. Prevent pollution of global scope
   b. Encapsulation

   c. Private properties and methods
   d. Allow conditional use of ‘strict mode’

2. To create a columned list of two-line email subjects and dates for a master-detail view, which are the most semantically correct?
   a. <div>+<span>
   b. <tr>+<td>
   c. <ul>+<li>
   d. <p>+<br>
   e. none of these
   f. all of these

3. To pass an array of strings to a function, you should not use...
   a. fn.apply(this, stringsArray)
   b. fn.call(this, stringsArray)
   c. fn.bind(this, stringsArray)

4. ____ and ____ would be the HTML tags you would use to display a menu item and its description.

5. Given <div id=”outer”><div class=”inner”></div></div>, which of these two is the most performant way to select the inner div?
   a. getElementById("outer").children[0]
   b. getElementsByClassName("inner")[0]

6. Given this:

```
angular.module(‘myModule’,[]).service(‘myService’,(function() {
        var message = “Message one!”
        var getMessage = function() {
                return this.message
        }
        this.message = “Message two!”
        this.getMessage = function() { return message }
        return function() {
                return {
                        getMessage: getMessage,
                        message: “Message three!”
                }
        }
})())
```

Which message will be returned by injecting this service and executing “myService.getMessage()”
   a. 1
   b. 2
   c. 3