Skills Assessment

Design Studio

December 2, 2021

Front End Web Developer

Version 2.1

modern campus

Confidentiality Note: This document contains confidential information, which is proprietary to Modern Campus. In consideration of receipt of this document, the recipient agrees to maintain such information in confidence and not disclose this information to any party outside of the recipient's organization. This information may not be reproduced or displayed to any other party without the written consent of Modern Campus.

Modern Campus shall not be held liable for any technical or editorial errors and omission contained herein, nor any damages directly or indirectly arising from this document.

© 2021 Modern Campus. All rights reserved. Confidential. Destiny One® and Omni CMS® are registered trademarks of Modern Campus.

Version Control

Version	Description of Revisions	Date	Author
2.0	Modern Campus branding	1/5/2021	MEA
2.1	Update achecker link	5/18/2021	MEA

Welcome to the Skills Assessment Exercise for the Web Developer position at Modern Campus. We ask all our candidates to take this assessment.

This assessment is designed to introduce you to the tasks we perform on a regular basis. It is broken up into three separate assignments of varying difficulty. Please complete as much of the assessment as you can. Once you have completed all assignments be sure to submit the source files as instructed.

Task 1: Build an HTML page

Task 2: Parse an RSS feed

Task 3: Query and Display Information From an API

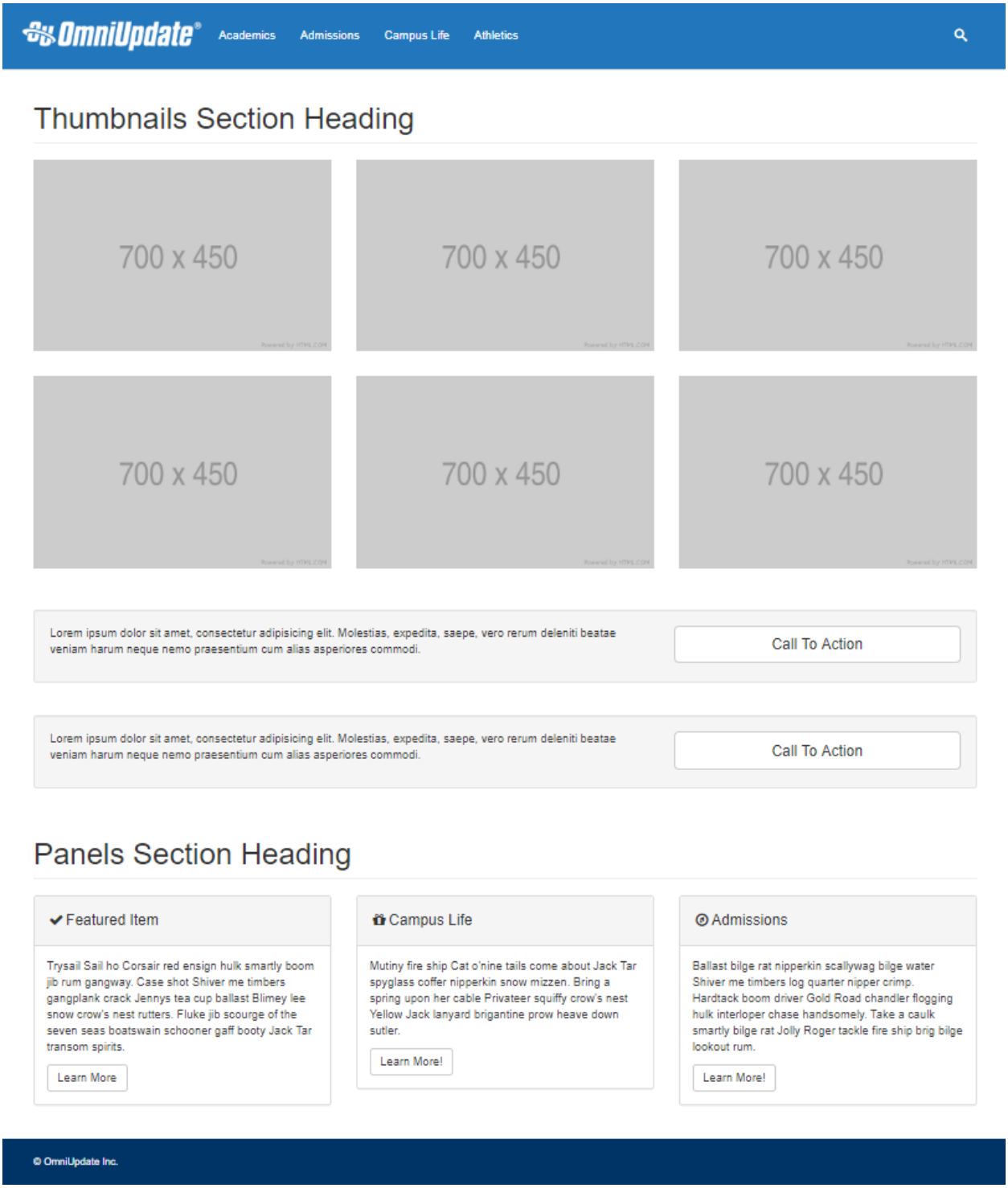
Task 4: Binary Tree Traversal & Sequence convergence

Do as much as you are able to complete. We are not expecting perfection, but the greater your attempts to perform the tasks, the better it shows us your dedication and ability to learn. When you have completed as much as you can, please contact Nick Catto at ncatto@moderncampus.com to inform him that you have finished the test. We look forward to seeing your talent!

.

Task 1

Create a full working webpage that looks like the following screenshot with the described functionality:



1. The page should be valid HTML5 and should also pass W3C and WCAG 2.0 (AA) accessibility standards.
2. Include all types of headings, paragraph, list, table tags on the page and style it using correct CSS.
3. Learn about jQuery plugin called light box- http://lokeshdhakar.com/projects/lightbox2/
4. Add 5 images to your web page with lightbox effect.
5. Make sure you have proper comments in your code.
6. Make sure you use CSS3 rounded corners as shown in the screenshot.

Useful Links:

http://checker.moderncampus.net/vnu/

http://checker.moderncampus.net/checker/

Task 2

Create a webpage that displays items from an RSS feed formatted similar to the output as shown in the screenshot:

The server supports PHP as the server-side scripting language. You can utilize PHP or JavaScript/jQuery for this task.

Source feed:

https://skills-assessment.oudemo.com/_resources/rss/news.xml

1. The links should open the page in a new tab.
2. Have conditions in your code that will display proper error messages when:
    a. Feed is empty
    b. Feed does not exist at the given URL
    c. Feed takes more than 5 seconds to load

Task 3

Create a webpage that requests information from an API with some parameters and displays the result.

The server supports PHP as the server-side scripting language. You can utilize PHP or JavaScript/jQuery.

The path to the API is https://skills-assessment.oudemo.com

Part One

1. You need to pass two parameters:
    a. "type" with the value "secret"
    b. "length" with the value "10"
2. On successful API request, it will return the status as "success" and will give you access to the "key".
3. Have your code reverse the "key" returned by the API and display on the page within a top-level heading.

Part Two

1. You need to pass a single parameter:
    a. “type” with the value “faculty”
2. On successful API request, it will return the status as “success” and will give you access to a set of faculty data.
3. Have your code display each set of faculty data in a listing-type view under the heading displayed in Part One above.
    a. The listing should be initially sorted by last name ascending, then first name ascending.
    b. The following fields should be searchable and sortable:
        i. First name, last name, title, department

Task 4

Part One - Binary Tree Traversal

https://www.codewars.com/kata/5268956c10342831a8000135

Part Two - Sequence convergence

https://www.codewars.com/kata/59971e64bfccc70748000068

Solve the above problems using Javascript.

You may use the CodeWars environment to test and verify your solution, but it is not required. Submit each solution as an individual file.

How to Submit

- All code for each task should be self-contained within its own folder named for the task, such as /task1.
- All files for the assessment should be compressed with a .zip extension that includes a reference to your name, Modern Campus, and the date completed. For example, candidate Fred Sand will submit fsand_ou_2021-01-16.zip.
- Email your submission to Nick Catto, ncatto@moderncampus.com or Natalie Burt, nburt@moderncampus.com