Flowers For Dreams Frontend Challenge

There are two parts to the challenge. The first, is to write a page that interacts with a simple API that provides delivery availability for flowers. The second, is to write a responsive checkout page that allows users to edit product fields. Neither of the pages need to persist any data past a page reload, just the functionality is needed. Feel free to use any JS/CSS libraries available. Bonus points if you add some CSS that makes the pages more visually appealing or **double** extra points if you can get it to match our brand.

If you have any questions don't hesitate to email me at patrick@flowersfordreams.com I'll try to get back to you ASAP.

When you are done, send me a zip file or git repo with your work. If you use a CSS preprocessor please include the compiled CSS and SASS/LESS/etc files too.

Page One - Product Selection

The first page is a simplified version of the product page we use now. (https://www.flowersfordreams.com/flowers/sun)

Your objective will be to populate the delivery information for a specific zip code or zip code/ delivery date pair using two API endpoints. Each endpoint returns a JSON object containing delivery information. The 'delivery_available' key is a boolean value that denotes if delivery is available for that zip code or zip code/delivery date pair. If 'delivery_available' is false you can ignore all other fields returned.

The two endpoints are

1. http://dev.flowersfordreams.com/delivery/zip-check/{zip_code}
    a. Where zip_code is a five digit zip code
    b. http://dev.flowersfordreams.com/delivery/zip-check/60622/
2. http://dev.flowersfordreams.com/delivery/zip-check/{zip_code}/{delivery_date}
    a. This endpoint is used once users have filled out delivery info.
    b. Where delivery date is a date string in PHP format Y-m-d (2016-06-12 would be June 12th 2016)
    c. http://dev.flowersfordreams.com/delivery/zip-check/60622/2016-06-29

The following is how the user would interact with the page.

Page One User Interaction/Requirements:

- Submit a zip code to an API endpoint and retrieve data about delivery options for that zip
- If the zip code isn't available for delivery indicate that to the user

- If the zip code is available display the availability options and associated delivery fees on a recipient info form. They also choose a delivery date at this stage.
- Once the user fills out the form and submits it, use another API endpoint to determine if delivery is still available on that date.
- If the API data says delivery is permitted then load page 2
- Otherwise, display the error message returned by the API and let the user know that the date was invalid
- The recipient information form should be hidden until a valid zip code is entered

Page Two - Checkout

The second page is a checkout page that displays some items in the user's cart. You will need edit the markup/CSS to make the page responsive and more appealing visually. Additionally, you will need to write Javascript code that adds some basic functionality to the cart. Making the page look pretty is a big plus.

Page Two Requirements:

- User should be able to delete an item from the page
- When deleting an item a prompt should be displayed to confirm the deletion
- User should be able to edit each cart item
- When an item has been edited they should be able to click a save button and have the edits reflected in the same format as the original item
- The page needs to be responsive from mobile on upwards to desktop
- Deleting and updating cart items should work on different screen sizes
- The edits to the cart should only persist until page reload, no need to store it with cookies/HTML5 storage/other means
- Fields that are being edited (Name, Address, Zip, City) should be wrapped in markup that makes them easily identified, something like a div or span or some other ta