Kevin Erickson

CIT 261:02 Online

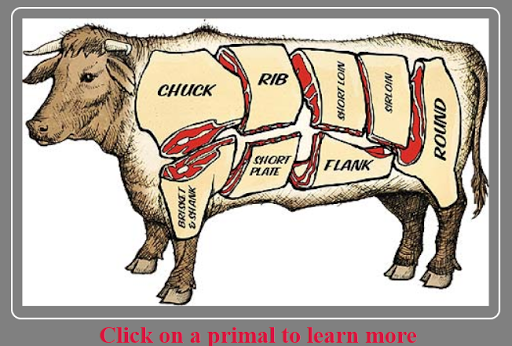
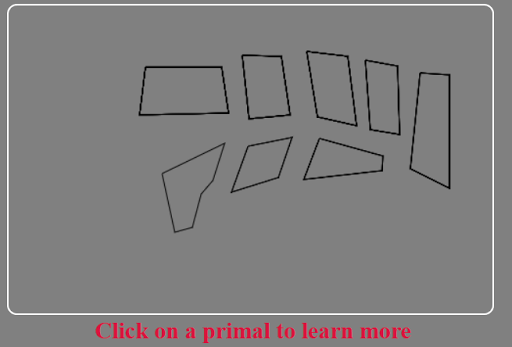
<http://360spinvinyl.com/app/personalApp.html>

“Learn About Cuts of Beef” app

My app is built to give the user information about specific cuts of meat in an interactive way. I have an image of a cow with each primal labeled. As the user chooses specific parts of the cow, they will be presented with preferred cooking methods and an opportunity to see similar cuts from that same primal. When the user clicks on the “click for more information” button, it activates the AJAX request for a JSON file on my Github account. This JSON file has all of the extra information about other cuts of each primal, so the user can cross reference how they should prepare it.

I wanted to keep this app simple yet informative. It was my goal all along to keep all of the content in one viewing area with no need to scroll down to see information. I was successful with that goal. I went with a black background to have contrast with all of the elements of the page. I tried other colors, but ultimately came back to black. It had the impact I was looking for.

As I first started building this app, my biggest obstacle was figuring out how to make the image clickable. Not only clickable, but picking up on where exactly the user clicked on the image. I also knew that using <Canvas> was a requirement. From here I placed the canvas over the cow image and plotted out polygons over each primal.



I was pretty excited. My issue here was how will I know that the user is clicking inside the polygon? This was a pretty big roadblock. So, I tried a different angle. Using these plotted coordinates, I made a JSON file that had all of the information I needed about that primal in one place, including the coordinates of the clickable area. {"id":"shortloin", "x":305, "y":40, "w":50, "h":70, "cook":"the grill and for braising"}, I made these boxes hidden after the fact.

I had some issues with figuring out how to add some CSS3 without it feeling like fluff. Ultimately, I added a little animation to the header, threw in a cleaver that does some chopping, and added a button to remove the cleaver. I thought that the cleaver animation might be something that the user would like to share with others which would give more exposure to my app. The animation of the cleaver was originally supposed to happen each time the user clicked on the canvas. I spent way too long trying to get that to work properly, so I dialed it back a bit and just added some initial animation. The main header mimics the movement of the cleaver.

I also added an area for the user to input their favorite cut. This goes to local storage and is also displayed on the right hand side. They can find out information about that cut through this app. Originally, I had the buttons in the app disappearing after they were clicked, but something went awry and I could not get them to disappear without messing with the alignment of elements after they were gone. Unfortunately, they stay visible now.

I am proud of so many aspects of this app. I am probably most proud of the JSON file I made for the clickable events on the canvas. There are easier ways to accomplish this through image map, but I saw that it could be done through the use of Canvas, so I went for it.

This has been a great opportunity to learn and grow as an IT major. I feel like I learned a vast amount about unfamiliar topics and my abilities to learn.

This app includes the following code topics:

Loops, Conditional Statements, Functions, Variables, Parameters, Arrays, Associative Arrays

Object Creation Functions, Inheritance, Properties

JSON Parse, Stringify

Using XMLHTTPRequest to Consume a JSON Web Service

Local Storage API, Storing and Retrieving Simple Data

Manipulating CSS Class Properties Using JavaScript

Creating CSS3 Transitions and Animations in CSS and triggering them with JavaScript

HTML5 Tags - Canvas

Designing, Defining, and Triggering CSS3 Transitions without Custom Libraries (Thought Library)

Designing, Defining, and Triggering CSS3 Transforms without Custom Libraries (Thought Library)

Designing, Defining, and Triggering CSS3 Animations without Custom Libraries (Thought Library)