* Describe the goal of the application and value to a user

The goal of the application was to create an interactive interface/application/website using REACT JSX. The goal of my specific application was to create an interactive website for a bakery’s menu and ordering functionality. The value provided to a user is that the website allows a user to view the bakery’s menu items, view and filter the items that are of a certain type (bread, pastry, or cake), view and filter the items that are within a certain price range (under $5, between $5-15, and over $15 dollars), and the user can sort the menu items according to price from cheapest to smallest. The user can also add the item they wish to purchase to their cart, which adds both the item name to their cart, which they can view, but also adds the price of said item to the total cost for their cart (the aggregator) which they can also use. All of that interactive functionality is the goal of the website and the main value to a user.

* Link to your deployed web application running online

<https://tiredlion1234.github.io/development/>

* Explain the organization of your Components, and the props and state related to them

The main component is my CardComponent. The states related to them are totalPrice, cart, disData, sorted, calories, and type. Type is what menu item type (bread, cake, pastry) filter state is currently being applied to the CardComponents. totalPrice is what the state of the aggregator (sum total price for the current cart state) is. Cart is the list of items in the cart associated with which CardComponents were clicked. disData is the list of sort CardComponents, sorted is the list of sorted card components being sorted. Finally, calories is the list of price range (this one is misnamed because it’s purpose was changed halfway through building the website) filters currently being applied to the CardComponents. The props passed into the CardComponents are the item itself, which contains the item’s name, price, picture, and type, the addtoTotal and the removeFromTotal methods/props. The addtoTotal adds the clicked CardComponent’s name to the cart list, adds the component’s price to the aggregator and changes the button text and color. The removeFromTotal method does the opposite.

The only other components are the filter buttons, sort buttons, and cart. The filter buttons are related to the cart, disData, calories, and type states, all of which are explained above already. The sort buttons are related to the sorted and disData states. The cart is related to all of the state variables list above. The props related to sorting are type, calories, and a simple Boolean. The props related to the cart itself (not the filtering/state aspect of it) are just the item names and prices that get mapped. The props related to filtering are calories, type, name, and sorted.

* Note the usability principles considered for layout and hierarchy

Color and clarity for reading text were obviously important for the layout and hierarchy. Larger text implied more importance in the hierarchy, like a title for something or the name of an item. Using Nielsen’s list of 10 usability principles/heuristics, Visibility of system status was very important for the layout (the status of the cart and which items were added to it or not was important to be shown in both the cart and the add to/remove item buttons). Also, the status visibility for the filter/sort buttons pressed and the corresponding list of items displayed should have also been visibly updated to reflect the state change. Matching between system and real world was also important. I believe that my layout/hierarchy matches most people’s mental models of what intuitively they would expect given my project goals. User control and freedom can be seen in the filter/sort button design. Flexibility of use is shown by how both filters and the sort functionality can all be combined and still work. There’s a lot of Consistency and Standards in both the website text and also the item card design and layout themselves. Overall, it’s a Minimalist Design and it’s aesthetically pleasing and simple to use.