

## Script for Final Project Demo

- Greeting
  - *Hello everyone, my name is Nicki “Boothman” and for my final project I designed an online user facing purchase application for ‘Sweet Cakes Bakery’ as an ecommerce marketing tool and to sell products. Users can create a user account and login, search for delicious items, and add those items to a cart for purchase. The idea for this project came from my family’s inspiration. I come from a long-line of bakers (none-of-which have their own website) so I decided for my project I would build on what I know-bakeries!*
- Homepage click through
  - *The homepage is styled with a main header with navigation links to different site pages. The logo serves as a ‘home’ button when navigating the website. Below the header is an image slideshow of different products the bakery offers and can highlight. The arrow tags on the right and left of the slideshow allow the user to click through the images to go forward or back to a picture they missed.*
  - *Displayed below the slideshow is a section that allows the user to jump immediately to explore the bakery items they want. By clicking on ‘cookies’, the button will render the ‘order’ page where the user can view and select the types of cookies they would like. The user can do the same for ‘pastries’, ‘cakes’, and ‘assorted gift boxes’.*
  - *Closer to the bottom of the page is a brief description of the bakery with an ‘image’ of the outside and a footer with the bakery’s information: the address, phone number, and social media widgets.*
- Login/Sign-up page click through
  - *On the main header, the user can click on the ‘person icon’ to login or create an account. This page features two forms: a create user form and a login user form. User can input their information in the given ‘become a member’ boxes and on submit- their information will be added to the users database in the backend of this project using an axios post request.*
  - *If a user does not complete each input in the form they will get an alert with an error specifying what they missed. Say a user is looking to create an account but they forget to add their first name: an alert will pop up at the top of the window saying the user should input their first name to complete sign-up. Same thing would happen if the user forgot their email instead of their first name: an error would appear that asks the user to add an email to complete sign-up.*
  - *Once a user has created an account, they are able to login. The login form uses an axios post request to post the user’s information (mainly their email and password) for comparison to the backend to then display their information in the frontend. If logged in successfully, the user will receive an alert saying they have been logged in. If an error occurs, an alert will show stating what the error was so the user can login successfully.*
  - *Also after successful login, the user will see that on the main header what was the person icon to login now has a ‘slash’ through it. When the user hovers over the slashed out user, they are told they can delete their account when they click the slashed out user button. This button will send a delete request to the backend to delete the user’s information from the database. Once the account is deleted, an alert will let the user know their account has been deleted.*

- Menu/Order pages click through

- *Let's say the user has created an account and now would like to order some items. The user can click either the 'menu' button or the 'order' button to view and search for whichever items they want.*
- *Once on the page, the user can see all the bakery items that are in this bakery's inventory. The items are displayed using an axios get request to the backend. Each bakery item tile is created using a function that loops through the inventory array in the database and creates a tile (made from a div, img tag, p tags, and a button) with the item's specific information, then appends each child to the bakery items container on the webpage.*
- *Say the user was in the mood for cake. They could either search 'cake' in the search bar which would render the different cakes the bakery has OR they could click on the category buttons just below the search bar, select 'cakes' and complete their search that way. Let's say the user wanted a danish- they can go through the same process. Search for 'danish' and see they have a cherry danish OR they could click 'pastries' and find the danishes that way. If the user ever changes their mind in their search and wants to see all items again, they can click 'all items' and all items will appear on the page again.*
- *Let's say the user has a hankering for something chocolate- they could either search 'chocolate' or click 'chocolate world' like shown before. The deluxe chocolate cake and the chocolate dipped oatmeal cookies are exactly what they want. The user can click the 'add to cart' button on each item and each item will be added to the cart above with the total price. The cart is also fixed to this corner, so if the user keeps the cart open and scrolls down to view other items to add to cart, the cart will follow them; so the user can keep track of how much they are spending and what's currently in their cart.*
- *Now let's say they don't want the oatmeal cookies anymore. They can simply click the trash can icon to the right of the item and that item will be removed from the cart with the resulting total price changed.*
- *Or maybe, since their new year's resolution is to watch what they eat and to cut back on the sweets they decide they don't want any items in the cart. Easy: click the 'clear cart' button at the bottom of the cart. All items will disappear, and the total will go back to \$0.*
- *In the cart, you can see there is an option to checkout. If the user wanted to purchase items, they would simply add the items to the cart and click the checkout button for an alert to pop up and thank them for their purchase and clear their cart. Since this is only a project, no credit card information is needed and everything is seemingly free (granted you won't get your items so it all works out).*

- Difficulties

- *This project really pushed me to apply all the knowledge I've learned during this 8-week foundations course and some parts of this project were very challenging. One part in particular was the cart functionality and the complex DOM manipulation I had to do to locate each tile 'add to cart' button, delete single items in the cart, and to change the total price value in the cart when items were added or removed. Running through my code, running tests, and de-bugging helped me solidify my understanding what each line is doing and improved my technical problem-solving to fix bugs and write code that would execute what I intended. This project also proved to me that I do not have the patience to be a front-end dev and am very excited to learn more about data analytics during specializations.*