#### Foundations Project Plan

**Idea:** Develop an online user facing purchase application for 'Generations Bakery' to ecommerce market and sell products.

# **Basic Requirements:**

# Highlighted values will be included in project

# Planning

- o project includes a wireframe for each view
  - [including below]
- project includes a list of MVP features \*
  - user will be able to search and view items
  - users will be able to add items to cart
  - users will be able to login to an existing account or create an account

# o project includes a data model

[including below]

#### MVP

# o app has at least 3 main features \*

- user will be able to search and view items
- users will be able to add items to cart
- users will be able to login to an existing account or create an account

# o front end makes a request to the server and handles the response \*

- user will request bakery item data from server (view items)
- user will request search of bakery items from server (search feature)
- user will request account verification from server (login feature)

### front end is interactive \*

- users will be able to click buttons
  - to add items to cart
  - to complete search for items
  - to click on items to view more information (name, quantity, price, ingredients, etc.)
- users will be able to submit forms
  - to type in items to search for
  - to add in user login information (both logging in and adding an account)

## app has custom styling \*

• [included below: documentation for styling]

#### Front End

# o app has at least 5 semantic tags \*

- <header>
  - Includes shop logo
- <footer>
  - Includes shop information (address, phone number, etc.)
- <details>
  - Includes additional information about the product (ingredients, 'special notes' from the baker)

- <section>
  - Includes images, animations, additional information, and forms for interactive web page
- <nav>
  - Includes nav links to other pages
- app includes 1 view \*
  - generic 'home' page of the web store app
- o app includes 1+ additional view(s)
  - 'bakery item' page for additional information on item and ability to add to cart
  - 'cart' page to view items for purchase
  - 'user account' homepage (login and create an existing account)
- styling includes flexbox
- o at least 1 view is responsive
- styling includes animations
  - slideshow/video of bakery items displayed in section on homepage
  - bubble floats up to cart and adds number of items in cart

#### Server

# o app includes a GET endpoint and handler function \*

- GET request for bakery items during search
- GET request for loading and viewing all bakery items in category
- GET request for logging into existing user profile

# app includes a POST endpoint and handler function \*

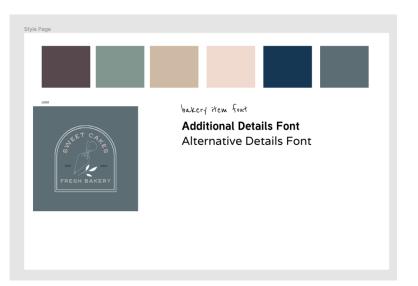
- POST request to create a user account
- POST request to document prior bakery items purchased for individual user
- o app includes a PUT endpoint and handler function
- o app includes a DELETE endpoint and handler function
- o app utilizes Sequelize
  - used to store data on specified user and bakery purchase history
- o project includes at least 1 controller file

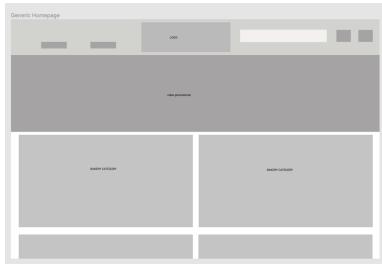
#### Database

- o project includes a seed file or function
- o app uses 1 table
  - user table
- o app uses 1+ additional tables
  - bakery items table
- o app uses a foreign key and join
  - foreign key used to connect user to previous purchase history

#### Presentation

- o discusses project purpose and demonstrates MVP \*
- o does not discuss broken/unimplemented features
- o recording is between 2-3 minutes











# users user\_id int user\_first\_name VARCHAR(50) user\_last\_name VARCHAR(100) user\_address VARCHAR(200) user\_phone\_number VARCHAR(10) user\_email VARCHAR(50)

# purchase\_id int user\_id INTEGER bakery\_item\_id INTEGER

bakery_items	
bakery_item_id	int
bakery_item_name	VARCHAR(50)
bakery_item_quantity	INTEGER
bakery_item_cost	INTEGER

