1. Describe what data is stored in the database. (Where is the data from, what attributes and information would be stored?)

   Our data includes a list of popular grocers and their relevant information such as service locations, number of warehouses, etc.

   Within each of the above grocers, we will require data about the produce that they retail and their relevant information such as name, price, quantity available, etc.

   Produce prices are either listed on the supermarket websites or may be provided by an API. A list of popular grocers in the US can be found on sites such as Wikipedia.

2. What are the basic functions of your web application? (What can users of this website do? Which simple and complex features are there?)

   Search for grocery items currently sold online by the various supermarkets and compare which supermarket offers the lowest price for each item

3. What would be a good creative component (function) that can improve the functionality of your application? (What is something cool that you want to include? How are you planning to achieve it?)

   Users who shop at smaller/speciality/non-chain supermarkets could upload prices of groceries they get at those places (with date/timestamp). These smaller supermarkets may not have a comprehensive website from which prices can be obtained by web scraping, and some users may prefer/have to buy from these places instead of the big chains (e.g. if they only need a small amount or if they cannot wait for the delivery).

4. Project Title

   Supermarket Produce Application

5. Project Summary:

   > Our project is a data centric web application that scrapes and aggregates data from popular US grocers in order to provide users with useful on-demand information on the current inventory, best prices, etc. The project will utilize a relational database schema such as MySQL to store relevant information on both grocers and their products.

The front end will include an interactive interface that allows users to query for their desired information. We will implement a search bar and filter function that will send HTTP requests to the web server which will in turn retrieve the relevant data from the back end relational database. There will also be a place for users to manually add/delete their own data.

6. **Description** of an application of your choice. State as clearly as possible what you want to do. What problem do you want to solve, etc.?

   Goal is to allow users to easily make price comparisons so they can always obtain groceries for the best price possible

7. **Usefulness**. Explain as clearly as possible why your chosen application is useful. Make sure to answer the following questions: Are there any similar websites/applications out there? If so, what are they, and how is yours different?

   Yes there are pre-existing supermarket grocery price comparison apps/websites available, e.g. listed on

   https://www.lifewire.com/best-grocery-store-price-comparison-apps-4169727

   Our application is different in that we provide additional functionality such as filtering/sorting by item name/price, manual addition/deletion of products, and favoriting stores.

8. **Realness**. Describe what your data is and where you will get it.

   Our data includes a list of popular grocers and their relevant information such as service locations, number of warehouses, etc.

   Within each of the above grocers, we will require data about the product that they retail and their relevant information such as name, price, quantity available, etc.

   Product prices are listed on the supermarket websites while a list of popular grocers can be found on sites such as Wikipedia.

9. Description of the **functionality** that your website offers. This is where you talk about what the website delivers. Talk about how a user would interact with the application (i.e. things that one could create, delete, update, or search for). Read the requirements for stages 4 and 5 to see what other functionalities you want to provide to the users. You should include:

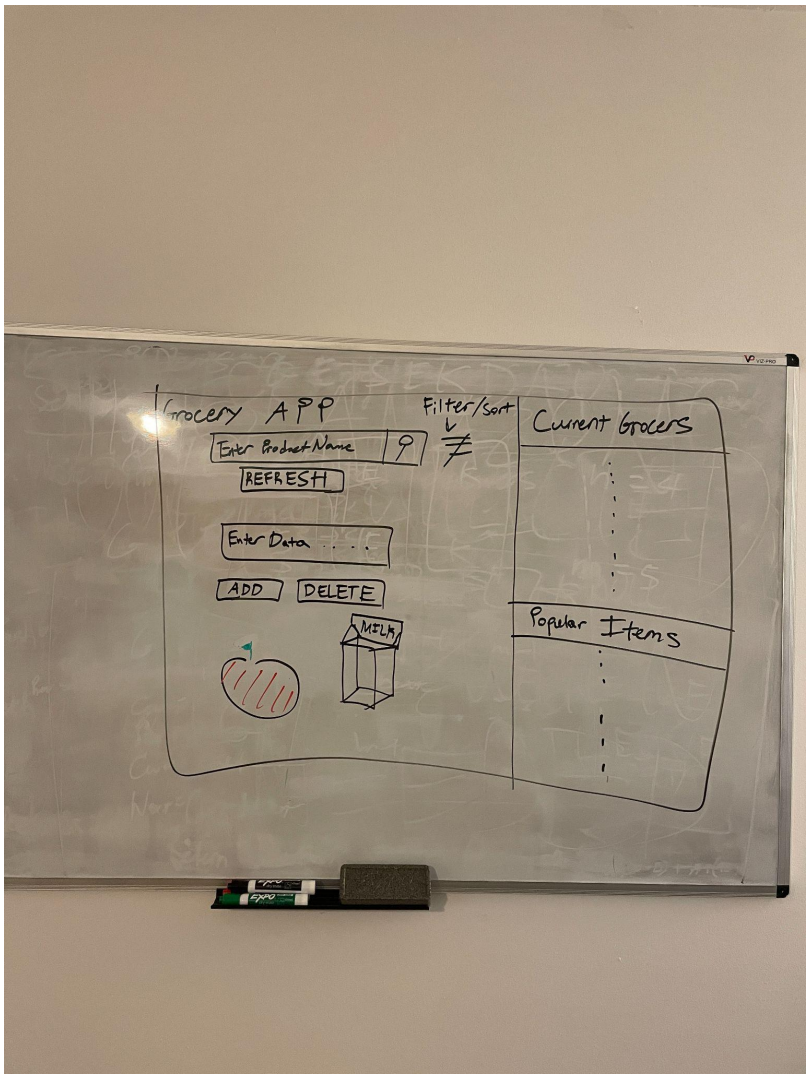
The web interface will allow users to input a variety of search commands such as searching for a specific product, searching for products in a specific price range, searching for products that are projected to arrive in a certain time frame, etc.

Users can create records in the database tables by manually adding certain products/grocer and their relevant information.

Users can delete records from the database tables by a similar manual deletion of a product/grocer and their relevant information.

In addition records may be added or deleted to/from the tables by requesting the web server re-scrape the relevant websites for new data.

10. **A low fidelity UI mockup**: What do you imagine your final application's interface might look like? A PowerPoint slide or a pencil sketch on a piece of paper works!



11. **Project work distribution**: Who would be responsible for each of the tasks or subtasks?
    List of the person responsible for which exact functionalities in section f. Explain how backend systems will be distributed across members. Be as specific as possible as this could be part of the final peer evaluation metrics.

    Everett: UI/Frontend
    Jie: Search and filtering functionality
    Khanh: Web scraping for data
    Ankushi: User self-upload feature