Gryffindor Final Report

A Summary of *Compare* Development

Rafay Chaudhry, Whitney Dluhosh, Jared Foulds, Morgan Kennedy, and Carl Marais

** The Gryffindor crest is a trademarked artwork used here under fair use and not for profit. It is not affiliated with UVic.*
https://s-media-cache-ak0.pinimg.com/736x/77/c9/c1/77c9c1ab7fb1cbd41a551677f92bae82--hogwarts-houses-crests.jpg

Table of Contents

Introduction **4**

Part 1: Context **5**

Problem Description & Motivation 5
Existing Solutions 5
Store Websites 5
Flyers 6
A Proposed Solution 6
Data Gathering 7
Personas 7
Scenarios 8
Use Cases 8
Creating an Account 9
Guest Login 9
Existing User Login 9
Creating a New List 9
Accessing a Pre-Existing List 10
Editing a List 10
Filtering Allergies, Ingredients, and Dietary Restrictions 10
Comparing a List 11
Saving, Printing, Sharing, Exporting, and Reviewing a List 11
Explore 11

Part 2: Evolution of the Prototype **12**

Evaluation of the Prototype 12
Peer Review 12
User Study 13
Evolution of the Design 13
Low-Fidelity Prototype 14
Figure 1: Low-fidelity list management page 14
Figure 2: Low-fidelity filter page 14
Figure 3: Low-fidelity list editing page (grocery overview) 15
Figure 4: Low-fidelity list editing page (fruit overview) 15
Figure 5: Low-fidelity list editing page (apple overview) 16
Figure 6: Low-fidelity list editing page (adding cheapest option) 16
Figure 7: Low-fidelity list editing page (list overview) 17
Figure 8: Low-fidelity list compare page with cover flow 17
Medium-Fidelity Prototype 18
Figure 9: Medium-fidelity list management page 18
Figure 10: Medium-fidelity filter page 18

Figure 11: Medium-fidelity list editing page (grocery overview) 19
Figure 12: Medium-fidelity list editing page (fruit overview) 19
Figure 13: Medium-fidelity list editing page (apple overview) 20
Figure 14: Medium-fidelity list editing page (adding cheapest option) 20
Figure 15: Medium-fidelity list editing page (list overview) 21
Figure 16: Medium-fidelity list compare page with tabular view 21
High-Fidelity Prototype 22
Figure 17: High-fidelity list management page 22
Figure 18: High-fidelity list management page (naming a new list) 22
Figure 19: High-fidelity list management page (with newly created list) 23
Figure 20: High-fidelity list editing page (overview with empty list) 23
Figure 21: High-fidelity list editing page (with filter pop-out) 24
Figure 22: High-fidelity list editing page (grains) 24
Figure 23: High-fidelity list editing page (bread) 25
Figure 24: High-fidelity list editing page (white bread) 25
Figure 25: High-fidelity list editing page (adding Wonder Bread) 26
Figure 26: High-fidelity list editing page (overview with added bread) 26
Figure 27: High-fidelity comparison page 27
Complete High-Fidelity Prototype 27

Part 3: Present and Future **28**
Future Work 28
Lessons Learned 28

Conclusion **30**

References **31**

Appendix **32**

Introduction

Over a span of three months, we developed a desktop website intended to help shoppers find the cheapest groceries. Our product was designed to allow users to filter their grocery list results based on their dietary restrictions and allergies. We chose to develop this website as a way to simplify the grocery shopping experience. Currently, alternative solutions require users to spend time learning multiple platforms/websites in order to achieve the same desired results. This process depends on the user manually cross-referencing prices for grocery items across several stores - a process which is time-consuming and overly tedious. Overall, our goal is to provide a service that is efficient, useful, satisfying - and of course - effective.

We developed our website in stages; first, we determined a target audience and the possible scenarios in which this audience might use our website. This step involved creating two personas that best match our website's intended use cases.

Second, we conducted interviews and a survey (see Appendix) in order to verify the original assumptions made about our targeted user base. During this process, we made modifications to our personas and scenarios, adjusting them to better suit our envisioned website.

Third, we developed a series of prototypes with our personas in mind. Using Balsamiq Mockups [1], we made low-fidelity and medium-fidelity prototypes, making changes from one to the next based on a peer review of our submission. A usability study conducted on some classmates and members of our target audience informed necessary changes to our medium-fidelity prototype.

A high-fidelity prototype was created based on a piloted user study to assess our site's discoverability and learnability. Based on the feedback we received, we created our high fidelity prototype with Atomic [2] with the necessary changes in mind. We iteratively went through our own prototype, editing the appearance and functional approaches for accomplishing tasks. Our final, interactive submission is available here at the following link: https://app.atomic.io/d/3TCYhR9ZYZXC. Please note that our trial period for editing this project ends on August 11th, 2017.

Part 1: Context

Problem Description & Motivation

At its core, our website seeks to save users time and money. The problem is that there is currently no efficient way for shoppers to know if they could be saving money by buying their groceries at another store. Without Compare, if someone wants to find the best price for all their groceries, they would have to look up each item individually, and compare the cost of different brands at different stores. This obviously becomes prohibitively time consuming for a grocery list with more than a few items.

Current grocery store websites provide difficult interfaces for people to navigate. As a result, shoppers waste time trying to find items and/or their prices by switching to alternate media for the same information, or simply not completing the task because of the lack of support. Moreover, because of inconsistent pricing, people must cross-reference items across different stores to find the best deals. As full-time students, we have limited amounts of time and money to spend on grocery shopping. Because this problem is something we deal with in our own lives, we hope to be unbiased advocates when solving it.

Existing Solutions

There is some difficulty for users seeking grocery stores with the cheapest total bills. To tackle this problem, shoppers must apply several disjoint solutions together in order to accurately determine which store is cheapest.

Our website caters to multiple scenarios (as described below) that deal with different goals. Therefore, existing solutions may overlap with one another, but may not provide the same rich functionality that we intend to serve. For example, some shoppers may use multiple grocery store websites to research and compare the costs of items at their nearby stores. Other shoppers may also do this, but take into account and possibly add deals to their lists based on store flyers. Thus, these two existing solutions can be used in tandem to solve the same problem our website addresses. The following examples are just some of the existing solutions to this problem.

Store Websites

Many people visit multiple store websites, searching each product on their grocery list on each site in order to find the cheapest price for their groceries. These websites aren't user-friendly, their search bar only works when under certain sub categories. They have poor filtering processes, overwhelming users with results. They don't provide an ingredients list with the associated product, which is a concern for those with allergies, and dietary restrictions. These flaws in the website make them time consuming, and frustrating to use.

Flyers

An alternative method to browsing websites is using flyers from different stores. Flyers display a store's best weekly deals and are used to persuade consumers to shop there. They do not accurately represent store pricing or product, as the items in these flyers are accompanied by the best prices for that store, giving consumers the impression that the store's product pricing is lower than it actually is. This may be deceiving to consumers, tricking them into paying more when checking out at the grocery store.

A Proposed Solution

Our solution is to provide a website for people to enter a grocery list and be given the cheapest place to go shopping for their groceries. This website allows people to enter their dietary restrictions or allergies and will filter their grocery list accordingly. We hope to resolve three major obstacles observed when trying to do these tasks with existing services:

1. **Time Wasted**: Current solutions are tedious and require you to search multiple sites to find product prices at each store. In the case of dietary restrictions, you are required to read the list of ingredients of suspect products. Our site will allow you to make your list, filter your ingredients, and proceed to find the cheapest store for your groceries without needing to learn multiple platforms.

2. **Poor Usability**: Grocery store websites are intentionally difficult for users to navigate. This is because they provide their best deals evidently upon entering the site to encourage users to shop based on their best deals rather than their average product pricing (which may not be competitive with other stores). Our site is an unbiased third party made with the user in mind. Therefore, we want it to be easy to use and give you the best overall deal when shopping.

3. **Selective Presentation**: Grocery store flyers often only inform shoppers of the store's best deals of the week. Often, these items are not on a shopper's grocery list. Our site will provide a user with the information needed to get the best price for his or her entire grocery list - not just specific items.

Our website hopes to remedy these issues by use of a database of each store's inventory. A user will be able to input a list. The website will then compute a listing of the cheapest stores in a given range for the user to complete his or her shopping trip. While a user is creating a list, he or she will also be able to apply filters based on dietary restrictions and/or allergies. The site will take into account these preferences when choosing products the user has specified as cheapest at each store. Once the user is satisfied, the site will then show an itemized list and its total price for each store within a specified search radius. The user can then print, share, download, save the list to Compare for later use, or view the itemized list online. Additionally, the user has the option to edit or use any previously created list, saving time when shopping for similar items weekly.

Data Gathering

Before crafting our product, we suggested a target user base for the service. Initially, we assumed that it would attract people who are budget-sensitive (i.e. students and low-income families).

We conducted a survey (see Appendix for survey and results) of over 60 respondents. Most of the participants were students between the ages of 18 and 23. Data consolidated from this survey was used to inform and reaffirm our initial hypothesis. We created (and later modified) a set of personas and scenarios to best model this user base.

Personas

Over the duration of this term, we have iteratively designed two key personas. We feel our final submissions are representative of the typical users of our service.

1) Lucy is a single mom with two children, aged 3 and 5. Due to a car accident last year, she is unable to work and no longer owns a car. As a result, Lucy receives both disability and child benefits from the government each month, totalling $1100 and therefore is operating on a small budget. She is allergic to products containing nuts and is lactose intolerant. Her two children inherited both of her dietary restrictions so it often difficult for Lucy to try new food new products as she has to review every product's ingredients before purchasing them. This is even complicated further when she has to purchase food specifically for her children. Due to tediousness of reviewing every food product's label, she often finds herself buying the same items every week. This leads to dissatisfaction with her food purchases due to the limited variety. Lucy wishes for a service that would allow her quickly sift through food products while taking into account her and her children's dietary restrictions so she can quickly explore new food product options.

2) Kyle is a student at the University of Victoria and therefore has limited income as he is not currently working. Kyle has his student loan to help cover expenses, but after tuition and monthly rent payments, there is not much left of it. To make ends meet, Kyle checks weekly flyers for the best grocery store deals and uses public transit. However, when at the store, Kyle often finds himself putting items back as he cannot afford all of the items on his list. He does not own his own computer and usually goes to the library when needs to use one. To make his limited budget more effective, Kyle wishes for a better way to compile his monthly grocery lists.

We found during development that the personas we first created did not have sufficiently distinct goals. Previously both personas were, at the core, simply frugal shoppers. In order to motivate varied use cases, we changed the persona of Lucy. In the final version of our personas, Lucy is still on a tight budget, but is now also shopping with dietary restrictions.

Scenarios

For the two personas we created, we designed two scenarios. The scenarios were updated throughout the development process to reflect changes in our personas and goals. Our final scenarios provide context for how a typical user would interact with our website.

1) Lucy is planning a birthday party for her youngest daughter and needs to buy snacks and dessert items. Since these are food products she normally does not purchase, she creates a brand new grocery list. Utilizing the "filter" feature of the website, she adds milk and nuts as allergies so the website will highlight and notify her if she adds products violating her dietary restrictions. She begins by searching for dessert items such as cookies. The website then returns the results of her search and highlights any food products that violate her constraints so she knows not to add them. Since Lucy prefers to save money, she also selects the search option to display her returned cookie results in terms of the best unit price as she will be buying in bulk for the party. She continues searching for other food items needed for the party in the same fashion, adding each product to her list as needed. Once she is finished, Lucy submits her grocery list, and the system automatically returns the total price for all her required items from each store, from cheapest to most expensive. Lucy then chooses the grocery store with the third cheapest price for her groceries as it is really close to her and easy to commute via bus than the top two options. Travelling for her via public transit is difficult so she values the convenience of a short trip over the few dollars she could save.

2) Kyle is planning his weekly grocery shopping trip with a specific budget in mind. Since he normally purchases the same items every week, he refers back to his last saved grocery list and edits it slightly to create a new grocery list for the current week. Kyle realizes that he is missing 2% milk from his most current list so he begins searching for 2% milk using the "Food Groups" section. Once he finds all 2% milk offerings, Kyle selects the cheapest offering of 2% milk as he has no brand preference and is only interested in obtaining the cheapest grocery products. After compiling a new grocery list for the current week, the system then returns the total cost of his submitted grocery list for the surrounding grocery stores in his area. After ensuring the total costs matches his budget, Kyle then chooses the grocery store with the cheapest cost. Even though the third option would be quicker to commute to via the bus, Kyle values the extra money saved over the few minutes saved.

Earlier in development, the personas of Lucy and Kyle had essentially the same goals, which resulted in the scenarios feeling very similar. After we changed the goals of Lucy, the scenarios naturally changed as well. This change helped solidify our alternate use case of using the dietary filter.

Use Cases

Over the course of our project, we have made several modifications to our use cases, which list out the detailed steps a user needs to take to complete specific tasks on our website. After conducting our user research and creating our high fidelity prototype, we made several minor changes to our use cases while incorporating feedback to produce the following, final use cases:

A. Creating an Account

1. The user arrives at the Compare homepage.
2. The user clicks the *Get Started* button.
3. A registration window appears.
4. The user clicks the *Register* tab
5. The user types in his or her name, location, valid email address, and password before clicking the *Register* button.
6. The user's submitted account information is then checked if it is valid and/or has been previously submitted.
7. A verification email is then sent to the user, notifying them that the account information has been submitted and awaits validation.
8. The user clicks the link in the verification email which brings them to the *My Lists* page.

B. Guest Login

1. The user arrives at the Compare homepage.
2. The user clicks the *Get Started* button.
3. A registration window appears.
4. The user clicks the *Continue as guest* button
5. The user is then sent to the *My Lists* page with button options for creating a new list or viewing and editing his or her previously created grocery lists (the latter may be populated while the user is logged in as a guest).

C. Existing User Login

1. The user arrives at the Compare homepage.
2. The user clicks the *Login* button.
3. A login window appears.
4. The user types in an email address and password before clicking the *Login* button.
5. The user is then directed to the *My Lists* page with button options for creating a new list or viewing and editing his or her previously created grocery lists.

D. Creating a New List

1. The user executes Use Cases A, B, or C.
2. The user clicks the *New List* button on the the *My Lists* page.

3. A pop-up window entitled *Name Your New List* appears on top of the *My Lists* page.
4. The user types in the name of the new list before clicking the *Submit* button.
5. The pop-up window disappears and a new list icon labelled with the user's submitted name now appears in the *My Lists* page.

E. Accessing a Pre-Existing List

1. The user executes Use Case C or D.
2. The user clicks a created list icon on the the *My Lists* page.
3. The user is directed to the *Edit List* section.

F. Editing a List

1. The user executes Use Case E.
2. The user executes Use Case G to apply desired filters.
3. The user can search for grocery items by typing categories, products, or brands into the search bar. They also can click the food category buttons in the *Food Groups* section to navigate to specific items.
4. Each item entered is checked to see if it is valid by the website.
5. For validation, if the input contains an ingredient that violates a filter, the item is highlighted in red as a warning to the user. Hovering over the item displays the filter violation. The user can choose to remove that item from their list or ignore the warning and continue as normal.
6. If the user wants to remove an item from the list, he or she clicks the *x* icon on the item.
7. As the user types items into the search bar, specific brands related to the current item are suggested to the user.
8. The user may click on a suggestion to expand on it.
9. A pop-up appears prompting the user to declare an amount, volume, or weight of the desired item.
10. The user may submit the item to his or her list.
11. The user reviews the list to ensure all items have been entered. Else, he or she returns to step 5.

G. Filtering Allergies, Ingredients, and Dietary Restrictions

1. The user executes Use Case E or F.
2. The user clicks the *My Filters* button to bring up the filter drop-down menu.
3. The user then checks any boxes in the *Dietary Preferences* section that contain their dietary restrictions.
4. The user can press the plus symbol in the *Allergies* section and type in any ingredients they want to filter out.
5. The user can enter in a new location in this section if they want to search grocery stores that are not near their default location.

6. The user then clicks the *My Filters* button again to minimize the *My Filters* drop-down menu to view the whole *Edit List* section again.

H. Comparing a List

1. The user executes one of Use Cases E or F.
2. The user clicks the *Compare* button to submit the list to be compared.
3. The cheapest grocery lists for all nearby grocery stores based on the user's filters are then computed.
4. The grocery list for each store is then presented with a total cost and distance, ordered by ascending price by default.
5. The user may alter the radius of search for nearby grocery stores.
6. The user selects a grocery store and proceeds with Use Case I.

I. Saving, Printing, Sharing, Exporting, and Reviewing a List

1. The user may now choose what to do with the created list. The user has the option of saving the list on Compare for later use, printing the list, sharing the list, exporting it to local storage, or viewing the detailed list on Compare.
2. If the user clicks
    - *save,* the list is then stored.
    - *print,* the print preview will appear with print options.
    - *export,* the user may save the output list to his or her computer as a text file.
    - *share,* a popup appears prompting the user for an email address (if the user is logged in, the account email will appear in this space). The user then types recipient email addresses before clicking *submit*.
    - *View List* for a desired store, the user is directed to a detailed list review page as created by Compare.

J. Explore

1. The user clicks the *Explore* button.
2. The user then browses items by clicking through categories or types to search for an item via the search bar.
3. After clicking an item, the user compares the cost of the item amongst other stores in the specified area.
4. The user can specify sorting options and/or a range to view stores within.

Part 2: Evolution of the Prototype

Evaluation of the Prototype

Note: evaluation materials and data are included in the Appendix.

Individually, we proposed ideas for a low-fidelity mockup of Compare. As a unit we then collaborated, adjusting the final submission to reflect the best components from each team member's design. Some of these early designs have been added to the Appendix below.

Peer Review

From cognitive walkthroughs and peer review of our low-fidelity prototype, changes to our design included:

- adding an *Explore* page for quick price look-up,
- moving the search bar to the top of pages with search functions,
- replacing specific product and item text buttons with images for easier recognition.
- replacing the cover flow grocery store comparison with a tabulated view, and
- adding a *Refine* page (extending from the Compare page) to view a compiled grocery list for a specified store.

Including a function for browsing offers both novice and experienced users a quick way to explore and compare specific groceries. This addition came as a result of user desire to rapidly see price comparisons. For an example of this page design, see the medium- and low-fidelity prototypes in the appendix below.

Our peers also provided important feedback with regards to the search bar featured on the list editing pages. In searching for groceries to add, students explained that they felt that the search navigation was too hidden. In response, we moved the search bar to accompany the navigation buttons nearer to the top of the page. This change is reflected by Figures 3 and 11.

Product images were included for specific items in favour of the item text buttons on the list editing page. We opted to include this suggested modification after considering the way people shop. Visual shoppers tend to recognize colours and labels rather than brand names, so being able to see these differences would be a huge user experience improvement. This is highlighted in the differences between Figures 5 and 13.

Following, users described the low-fidelity grocery store comparison page as being difficult to scan. They were not able to easily compare prices between any two stores, which prevented users from solving the initial problem. Instead, we redesigned the page to feature a table of nearby stores, each with distance and total bill. We also included markers for

cheapest bill and closest grocery store. This modification is prevalent between Figures 8 and 16.

An extension to the redesigned comparison page was a page to refine and review a grocery list for a particular store. Spurred by user desire to directly view Compare's proposed grocery lists on-site, we made this change to make final changes before submitting the list to be shared, printed, or downloaded.

These changes informed the medium-fidelity prototype which was then assessed with our piloted user study.

User Study

We piloted our user study with five people to determine if we had met the usability and user experience goals of our medium-fidelity prototype. There was a lot of feedback regarding our page progression and forgiveness.

We found that the average user took 33 seconds to find the list editing page which was undesirably long. While watching the test users execute the list creation task, we found that when landing on the filters page, they assumed they were in the wrong place or had clicked a wrong button. This was a flaw in our logic of the page progression. We modified the design by turning the filters page into drop-down bar on the list editing page. Now, the filters can be easily viewed and altered when desired.

We also found that each user ended up going further back than necessary because we hadn't planned for users to go back from certain points. This was something we remedied in our high-fidelity prototype by showing many "back to *x*" buttons which lead to the previous page. We also added headings to the top of the page to signify the page you are currently on, so that when you decide to go back it will indicate the page you're going back to by name. To plan for forgiveness, we also added confirmation pop-ups to our prototype to confirm when a user wants to add and delete things from their list and for when users choose to save, export, send, or download.

Evolution of the Design

The following subsection shows some of the described changes to our prototype's design as we developed low-, medium-, and high-fidelity mockups. These screenshots depict the path for creating and editing a list as an existing user.

Low-Fidelity Prototype

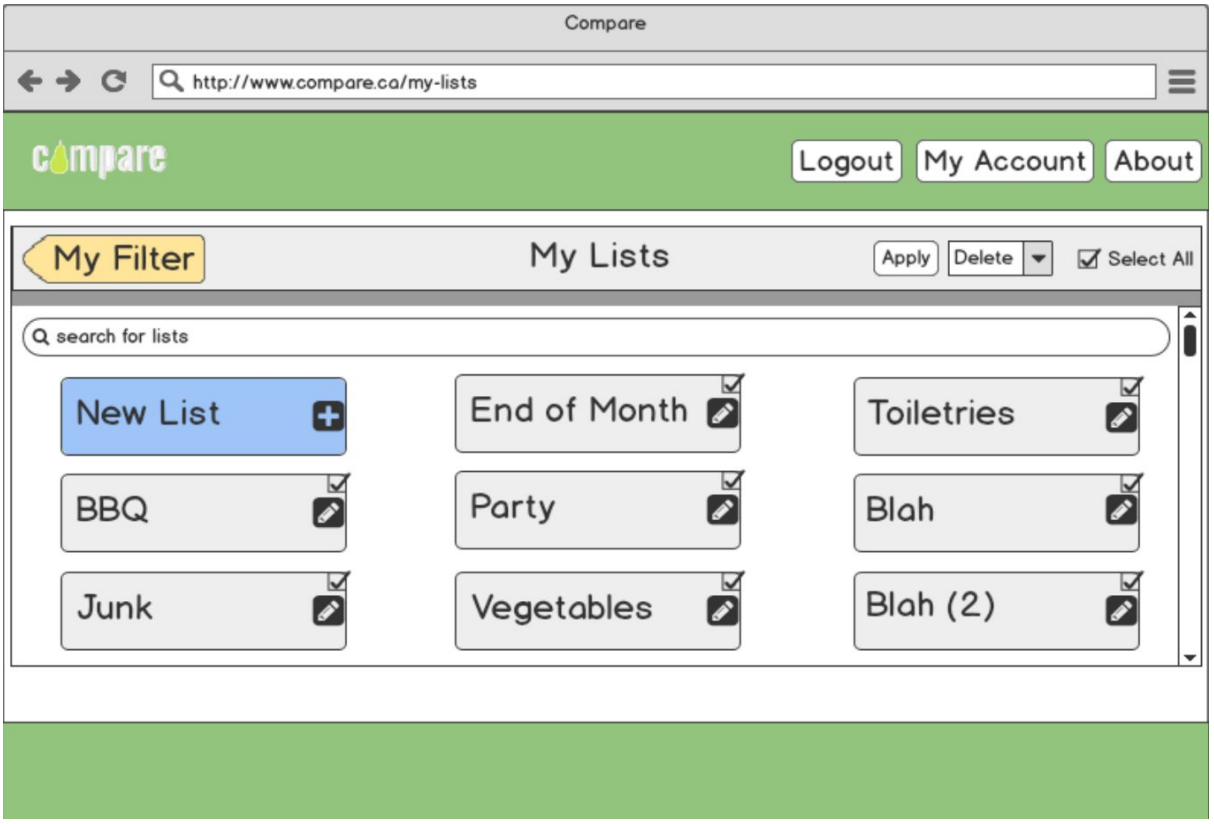


Figure 1: Low-fidelity list management page

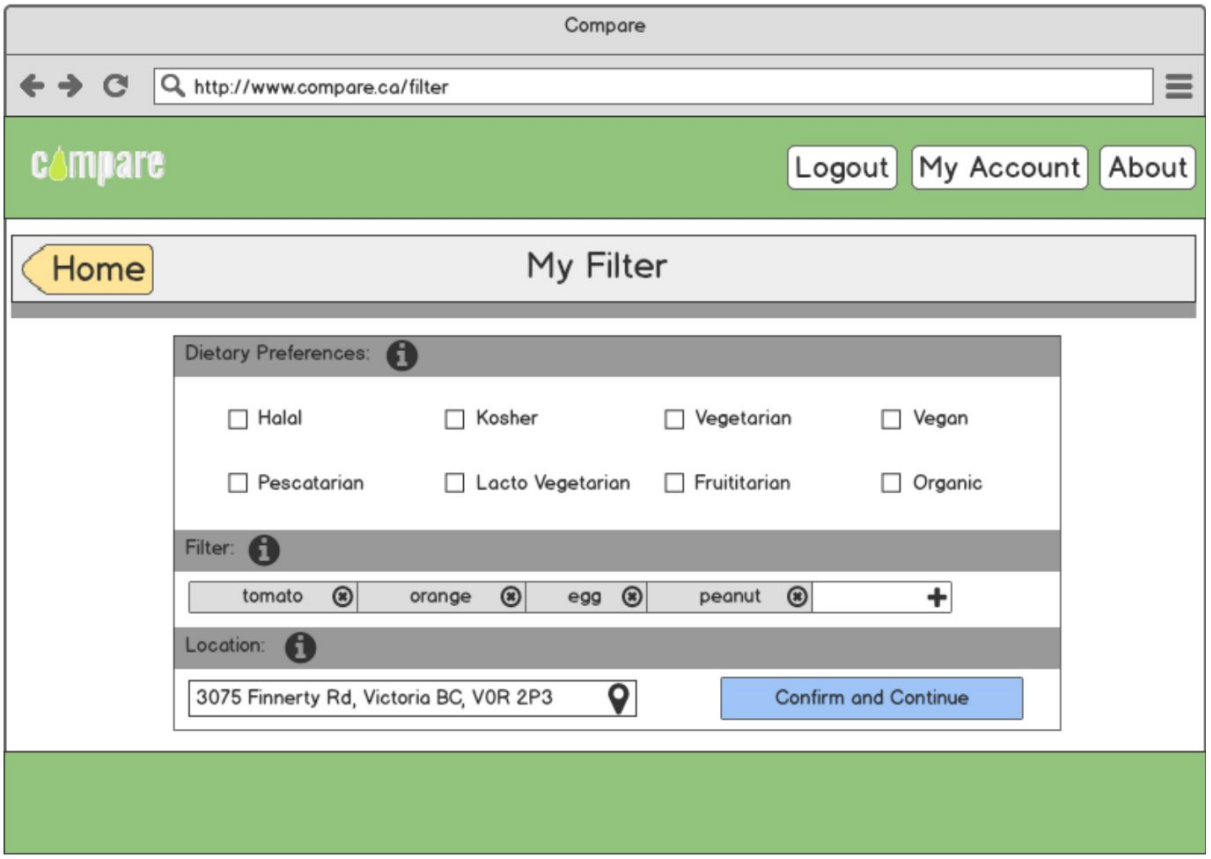


Figure 2: Low-fidelity filter page

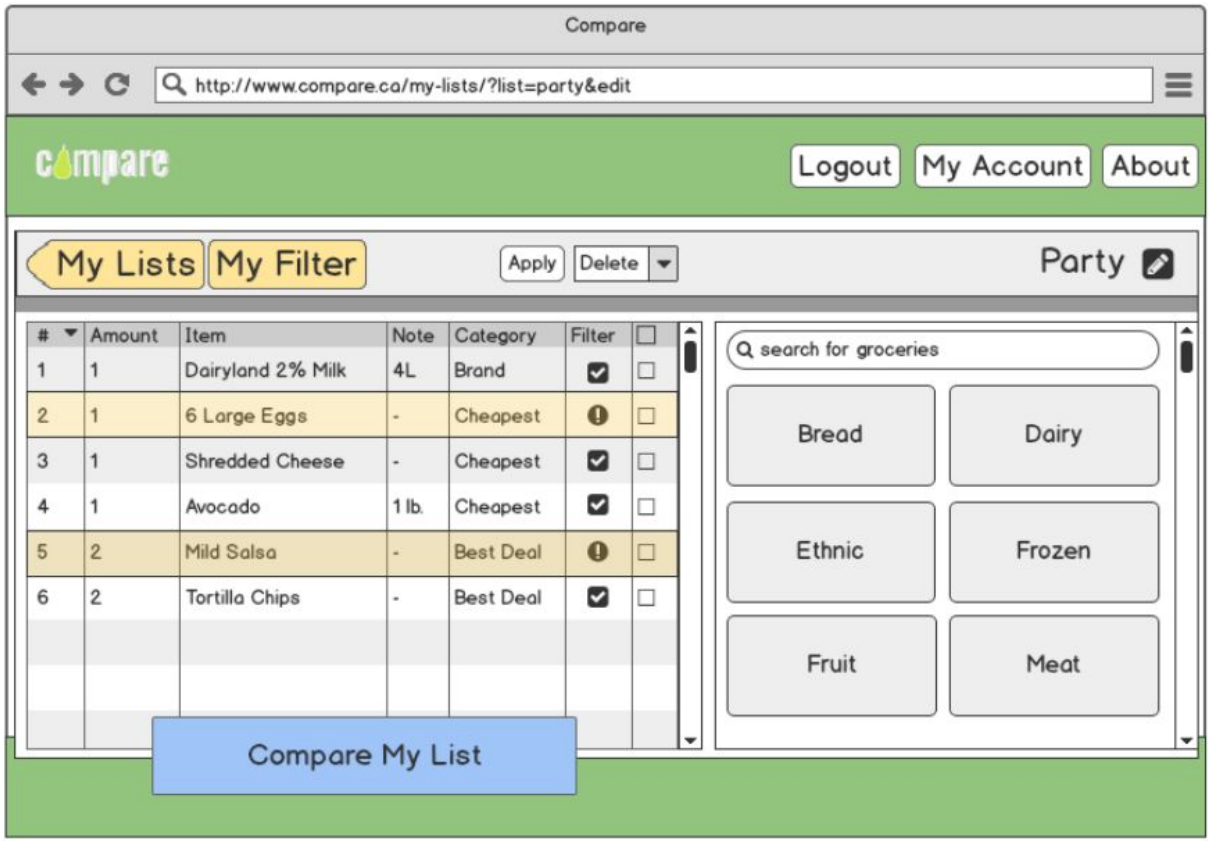


Figure 3: Low-fidelity list editing page (grocery overview)

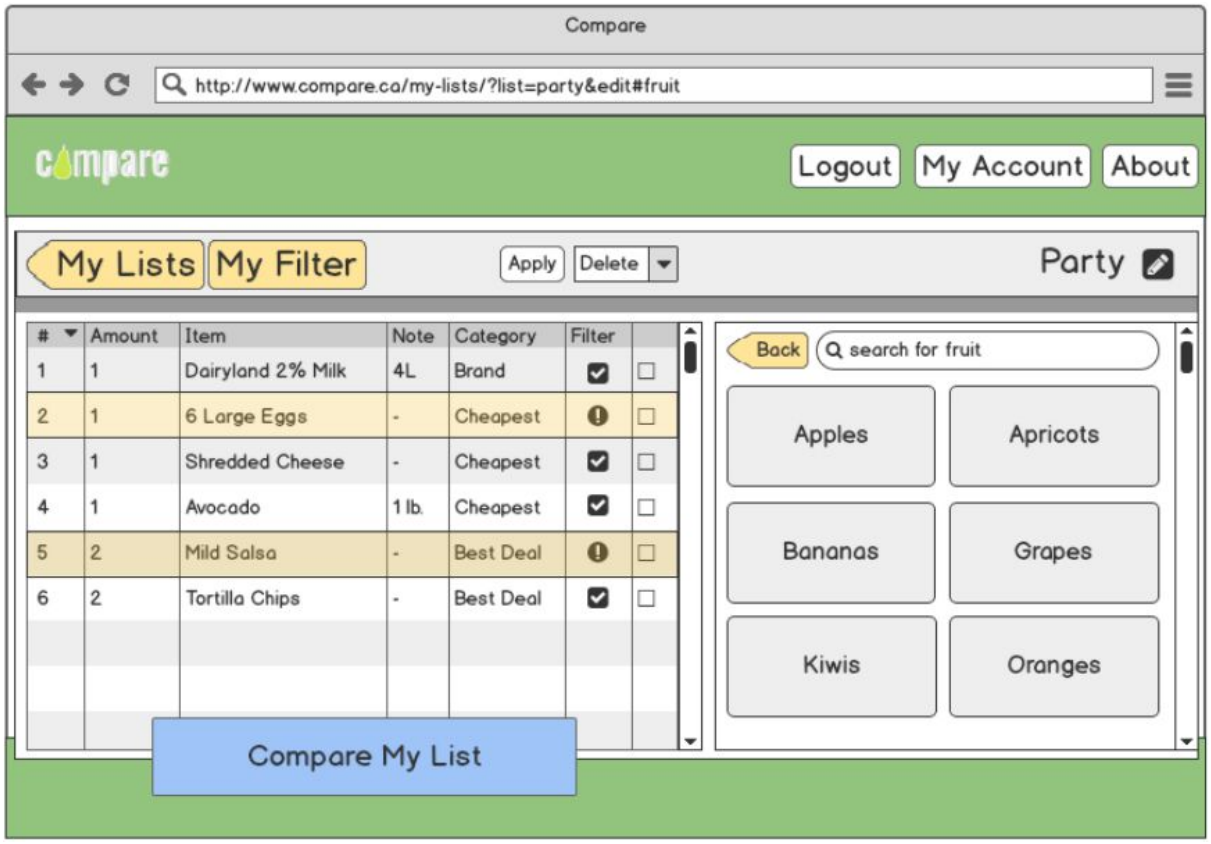


Figure 4: Low-fidelity list editing page (fruit overview)

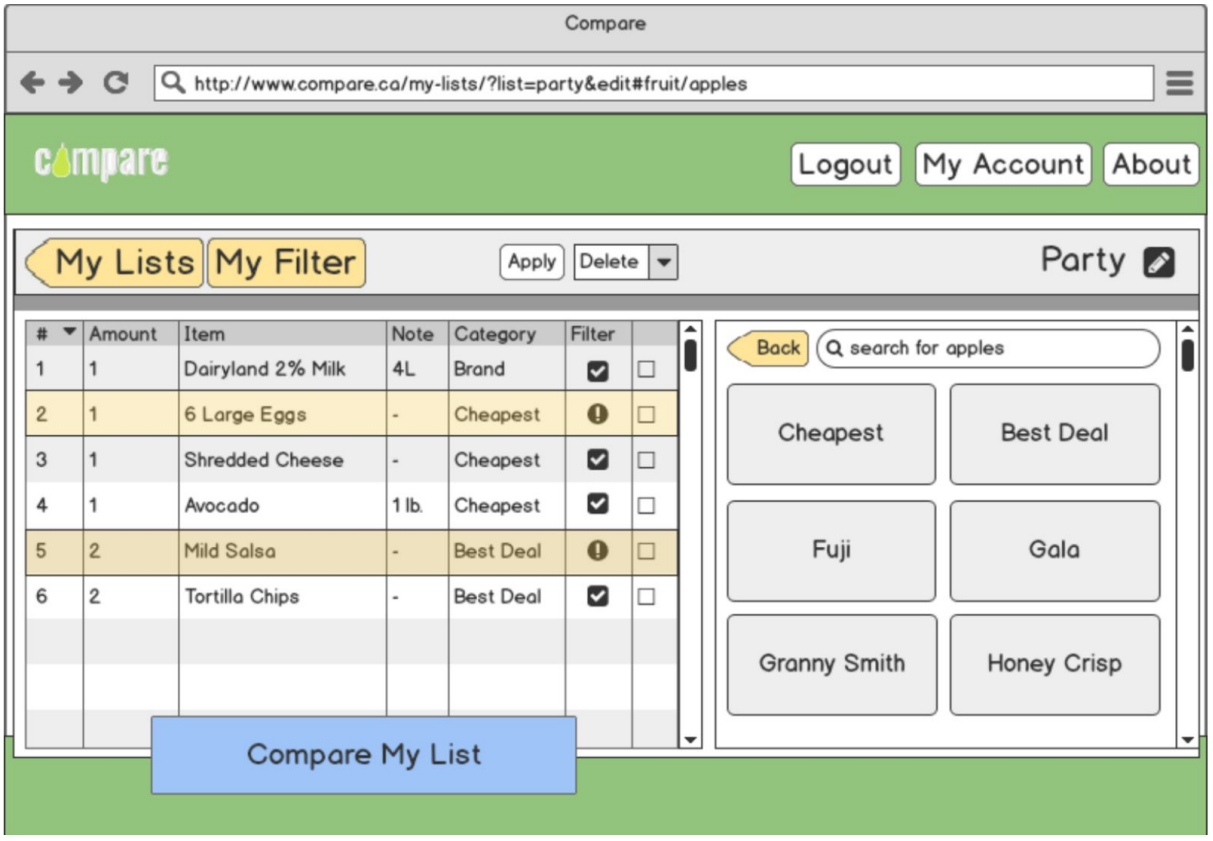


Figure 5: Low-fidelity list editing page (apple overview)

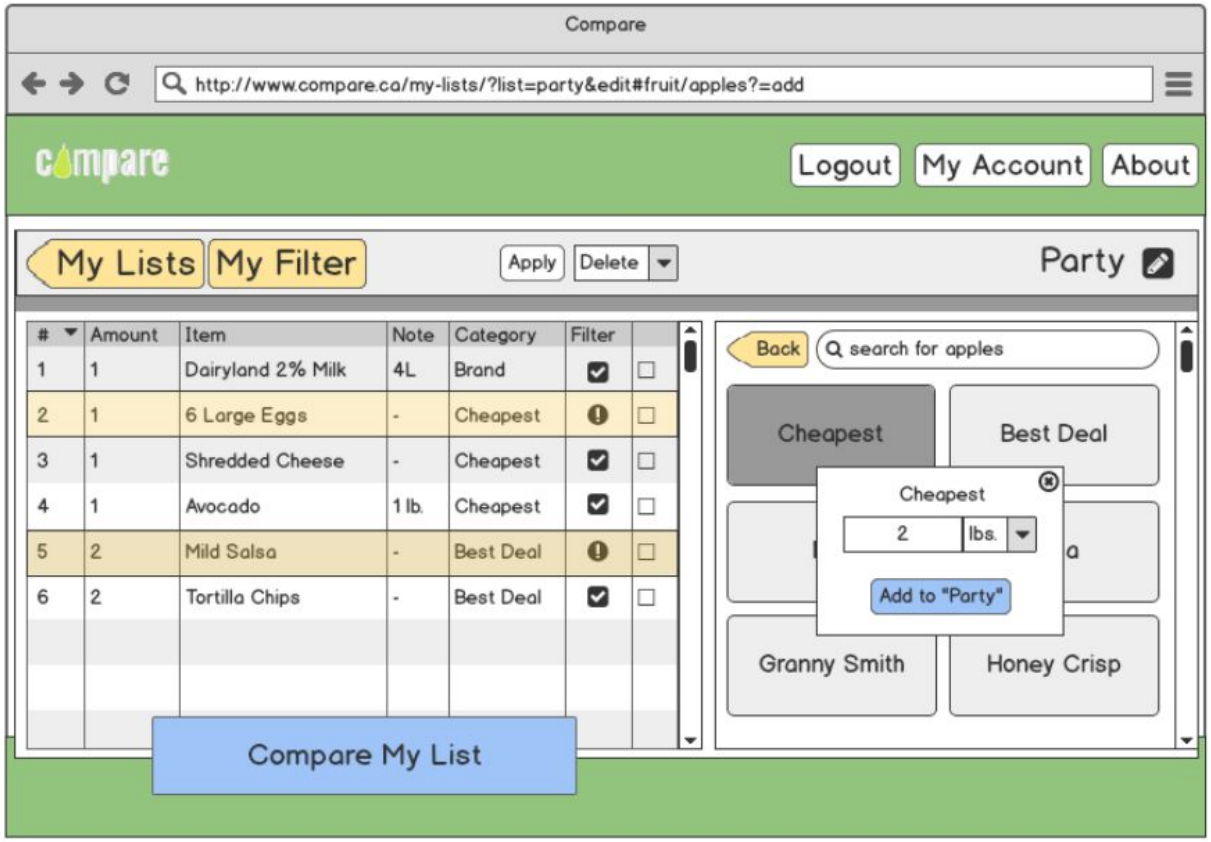


Figure 6: Low-fidelity list editing page (adding cheapest option)

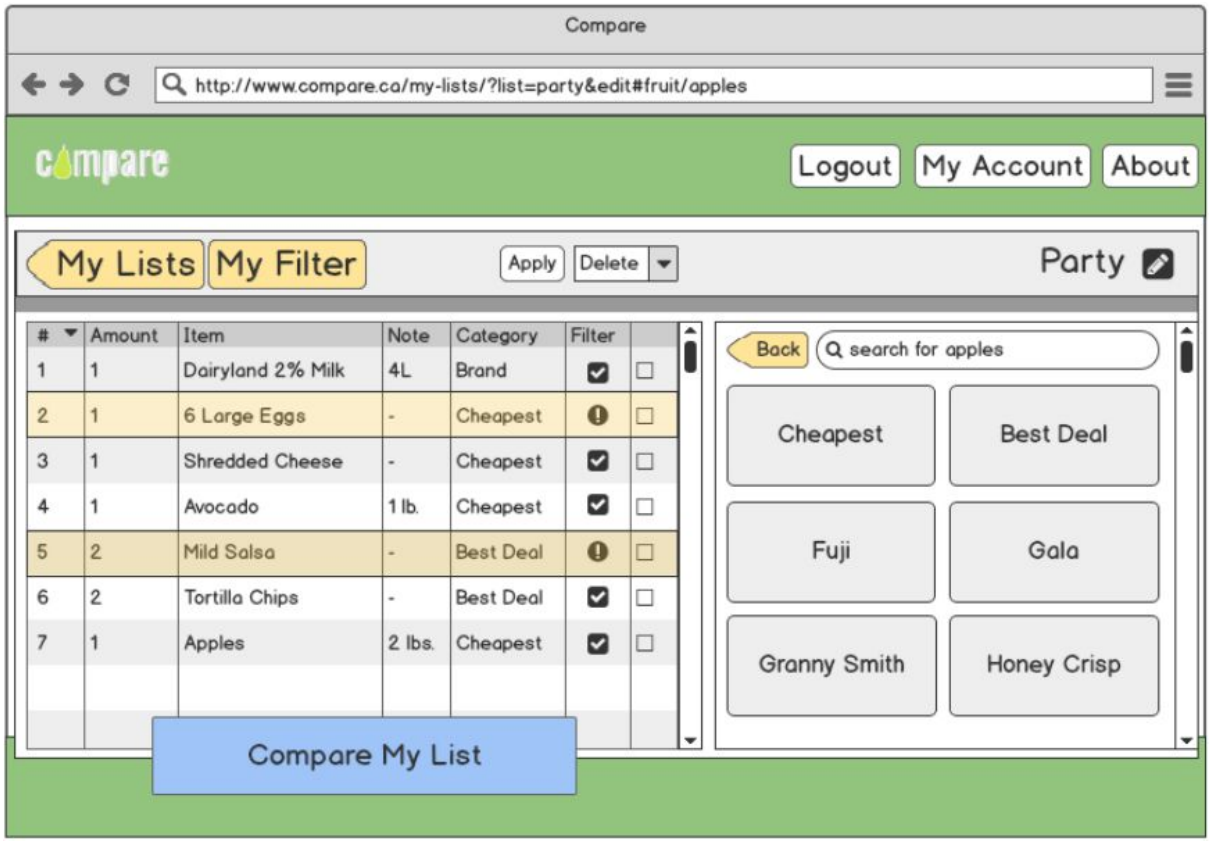


Figure 7: Low-fidelity list editing page (list overview)

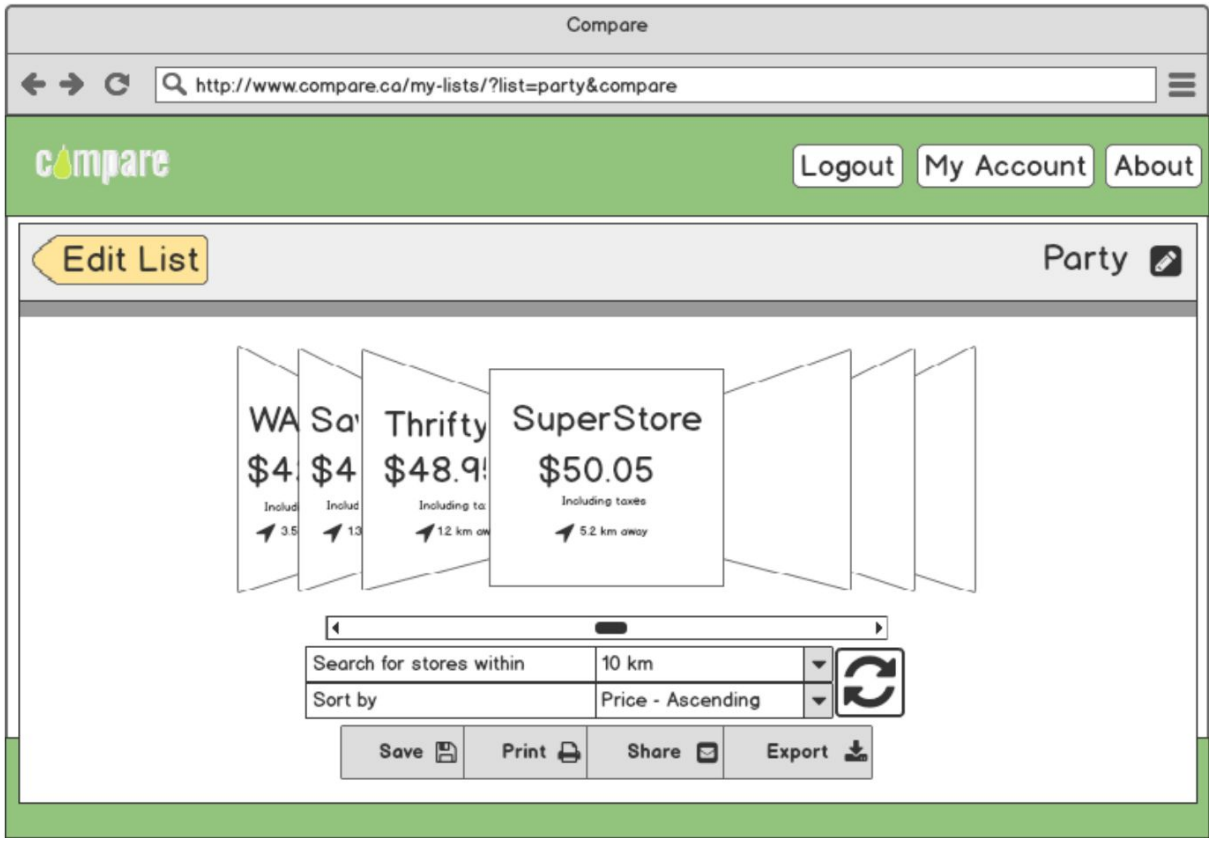


Figure 8: Low-fidelity list compare page with cover flow

Medium-Fidelity Prototype

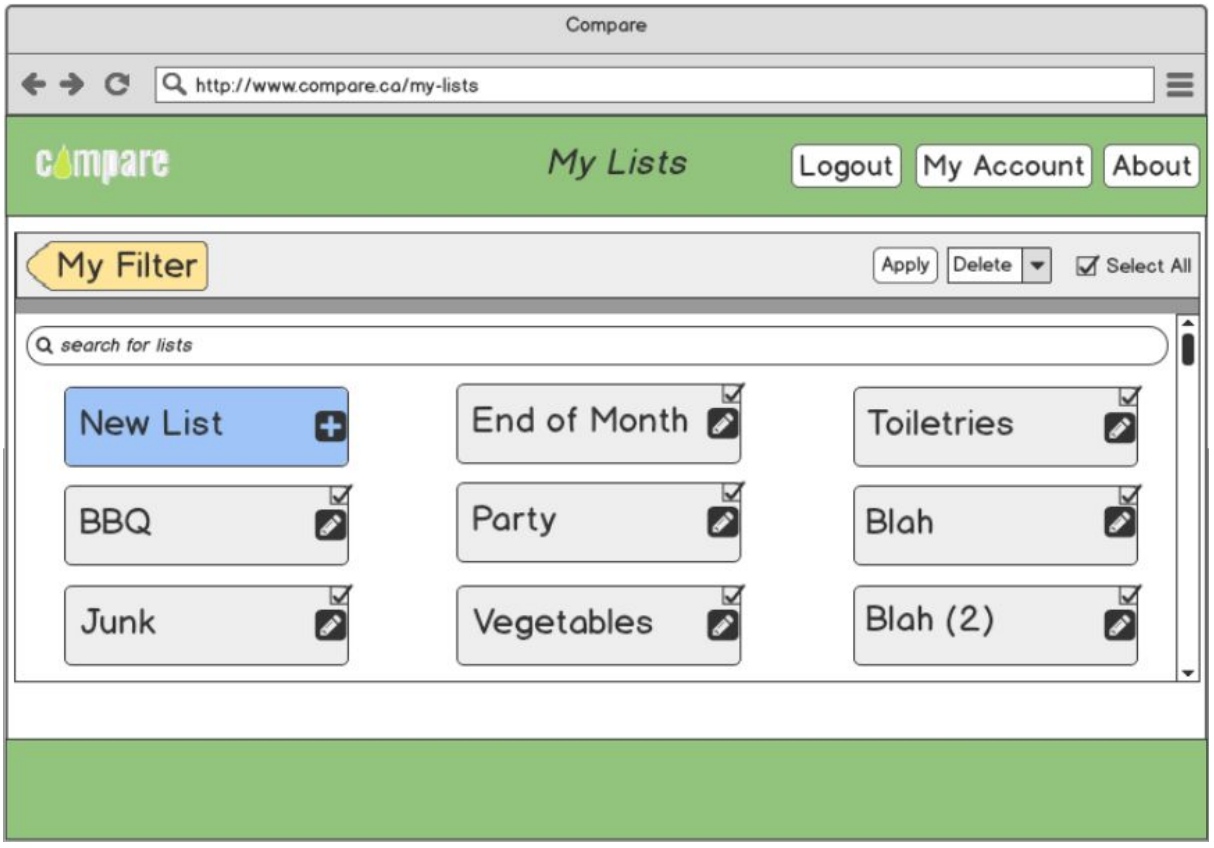


Figure 9: Medium-fidelity list management page

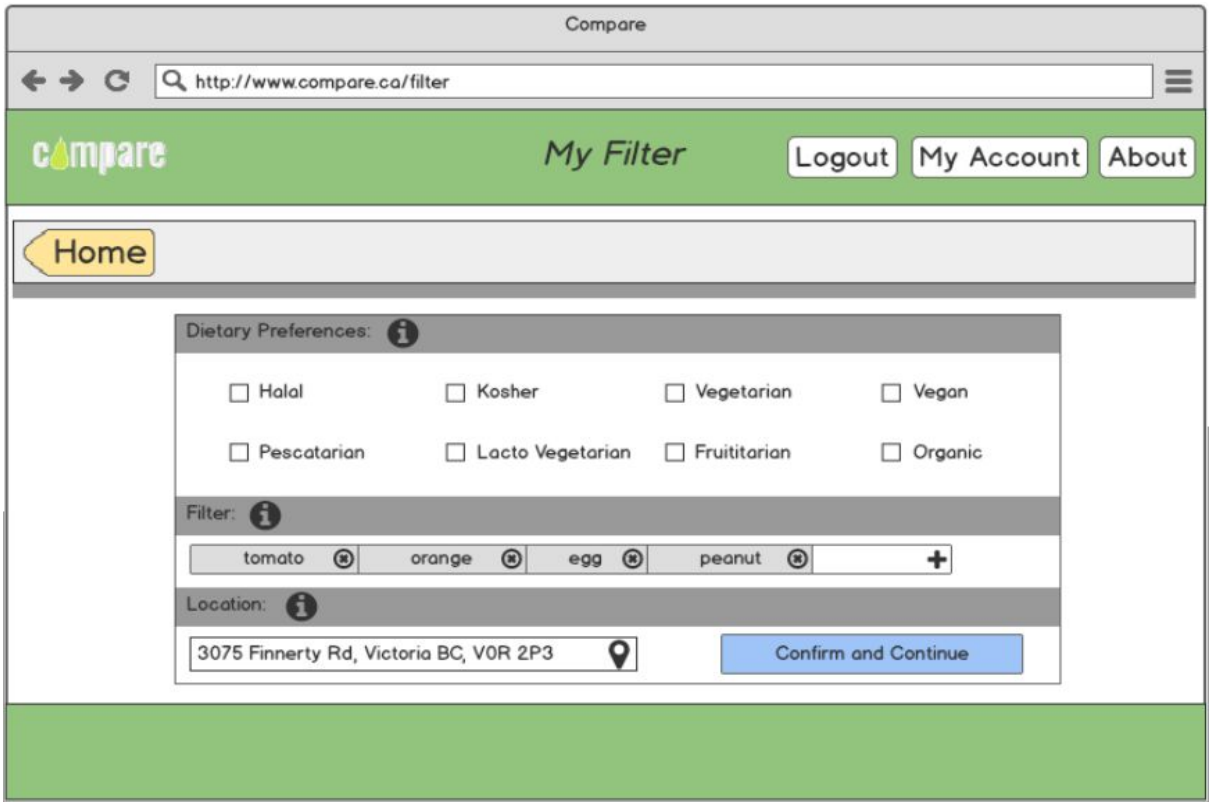


Figure 10: Medium-fidelity filter page

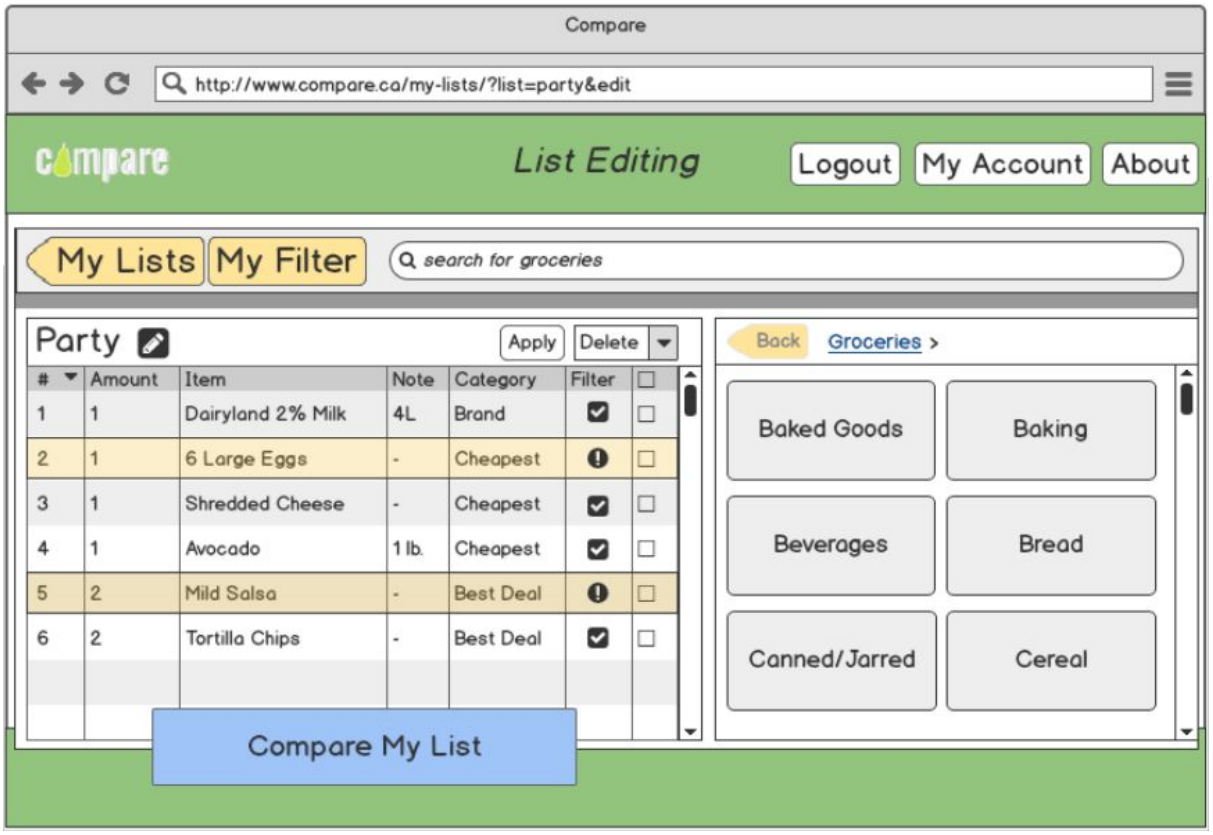


Figure 11: Medium-fidelity list editing page (grocery overview)

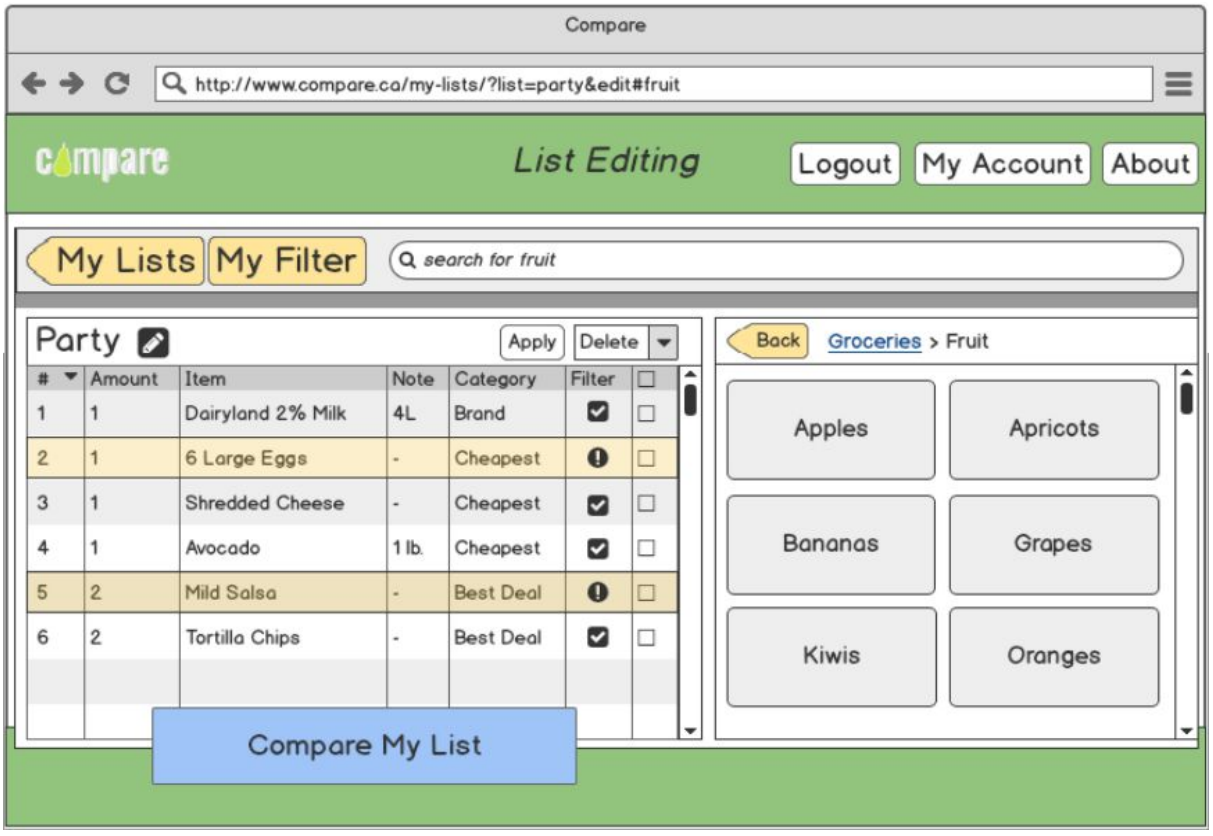


Figure 12: Medium-fidelity list editing page (fruit overview)

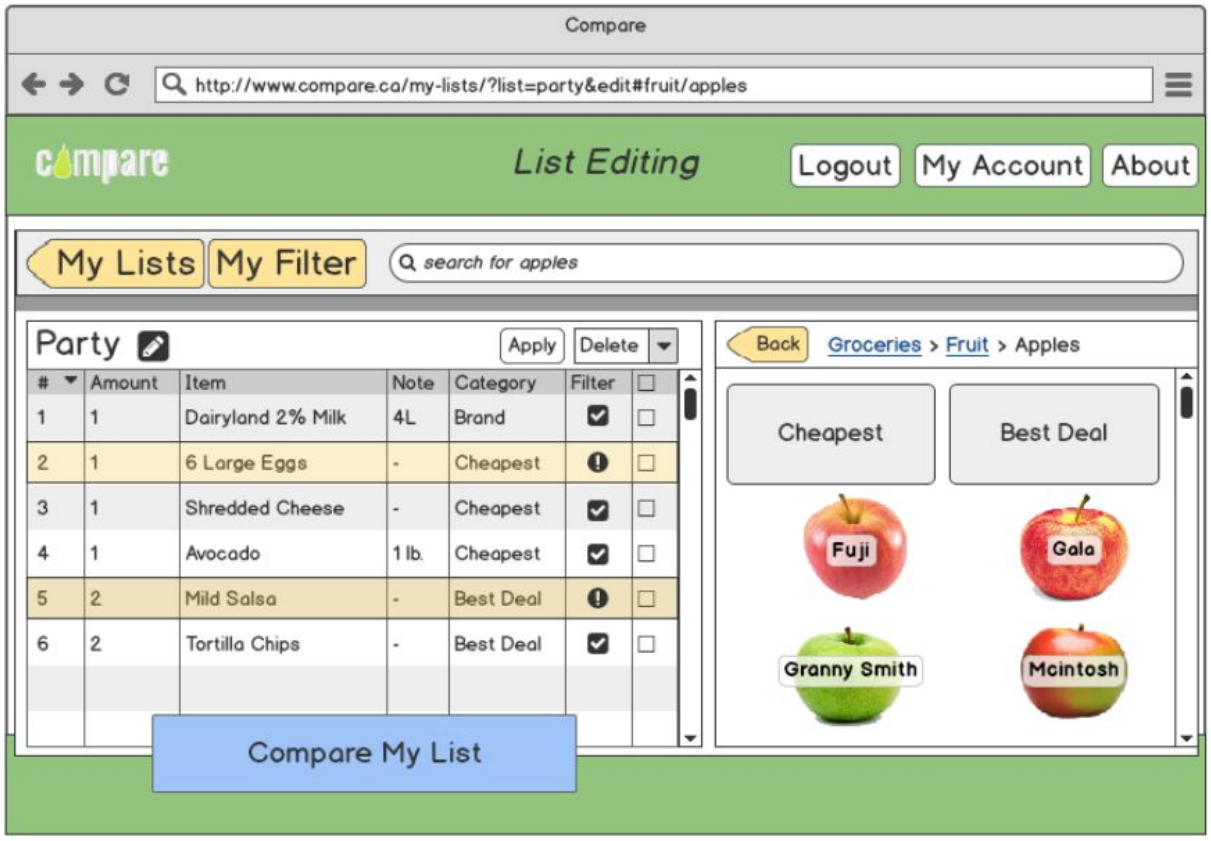


Figure 13: Medium-fidelity list editing page (apple overview)

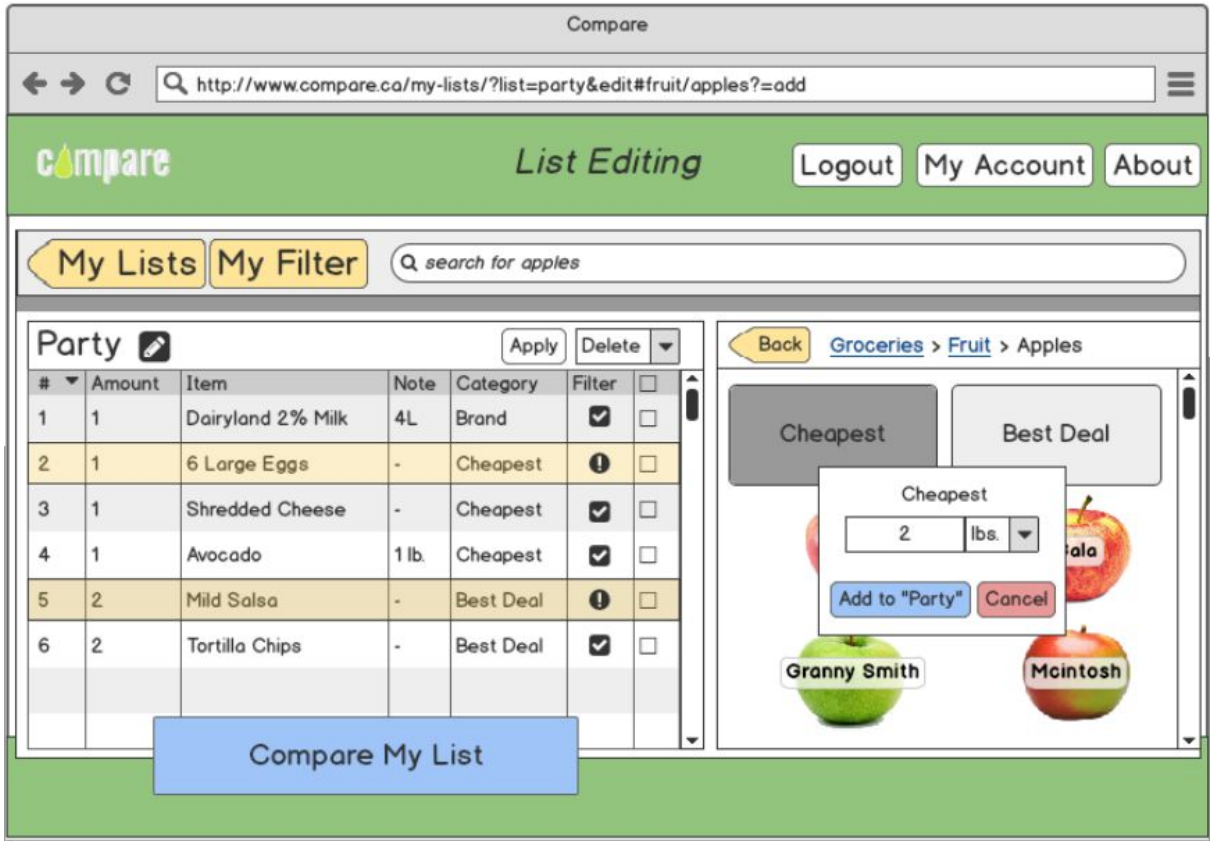


Figure 14: Medium-fidelity list editing page (adding cheapest option)

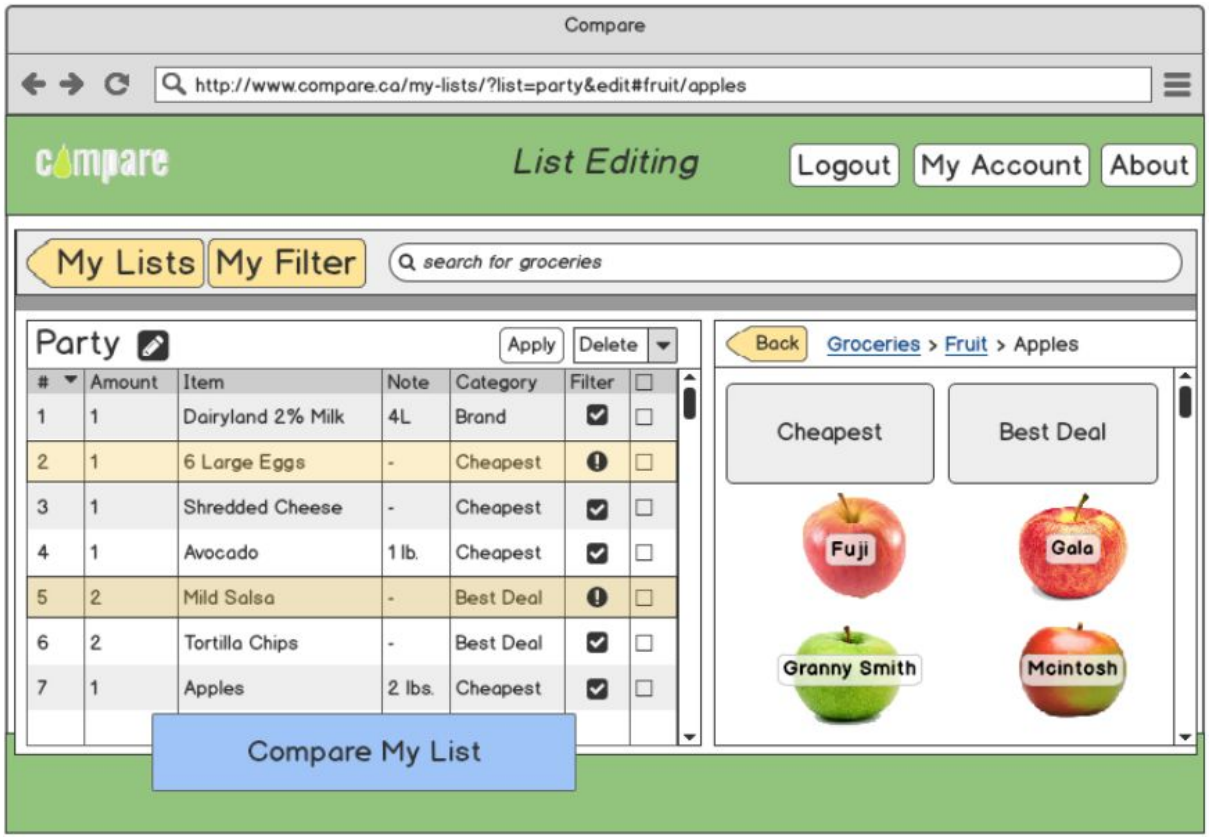


Figure 15: Medium-fidelity list editing page (list overview)

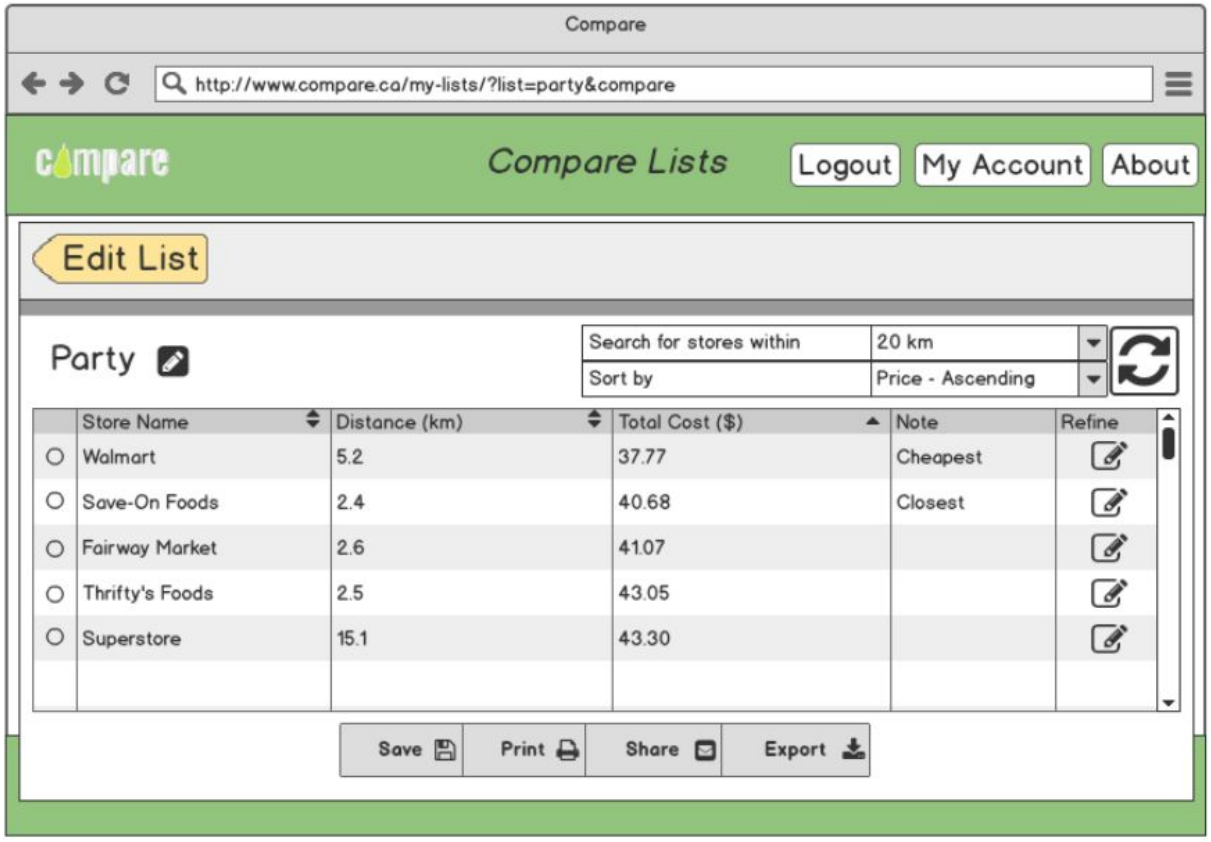


Figure 16: Medium-fidelity list compare page with tabular view

High-Fidelity Prototype

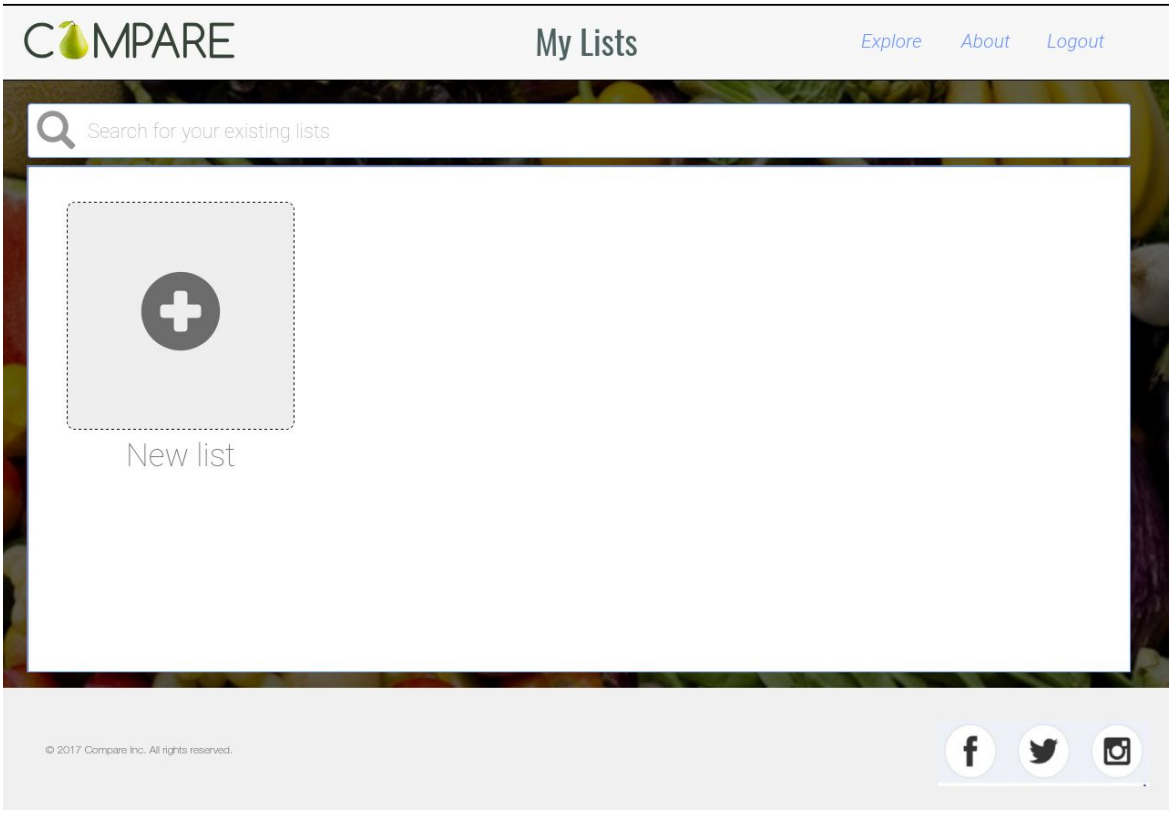


Figure 17: High-fidelity list management page

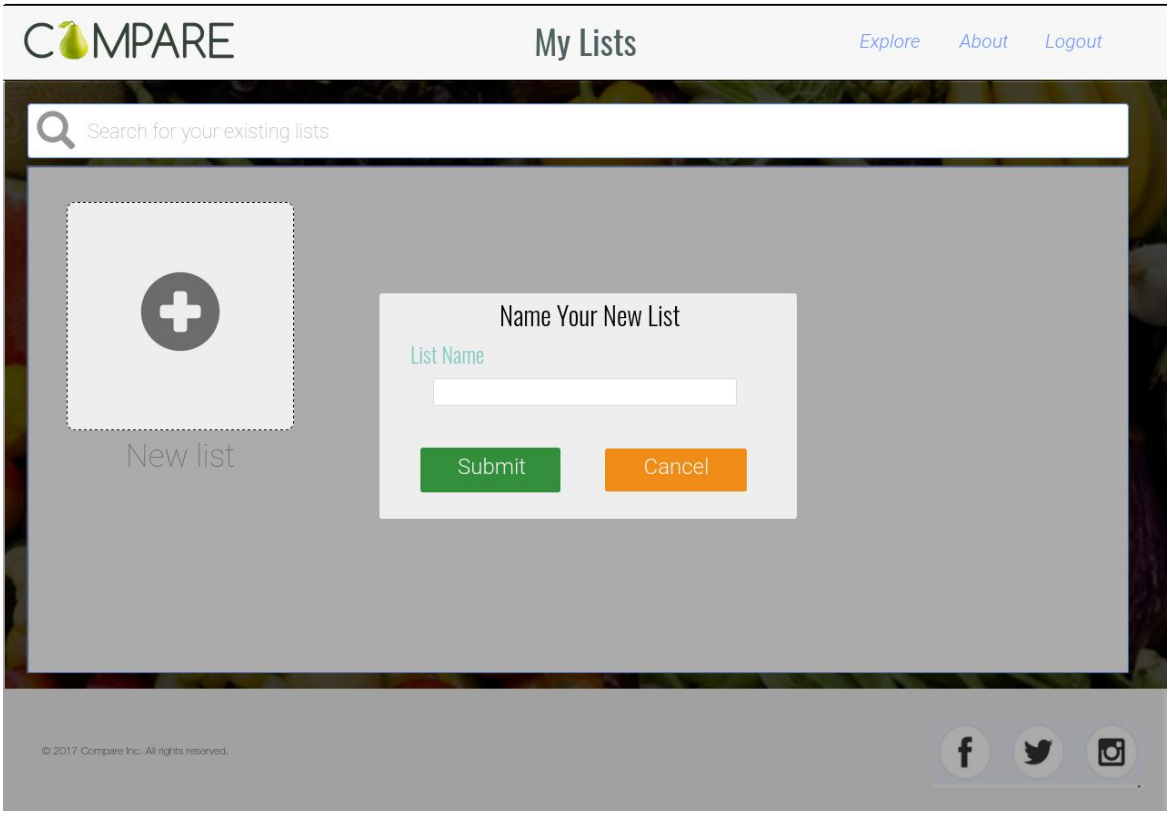


Figure 18: High-fidelity list management page (naming a new list)

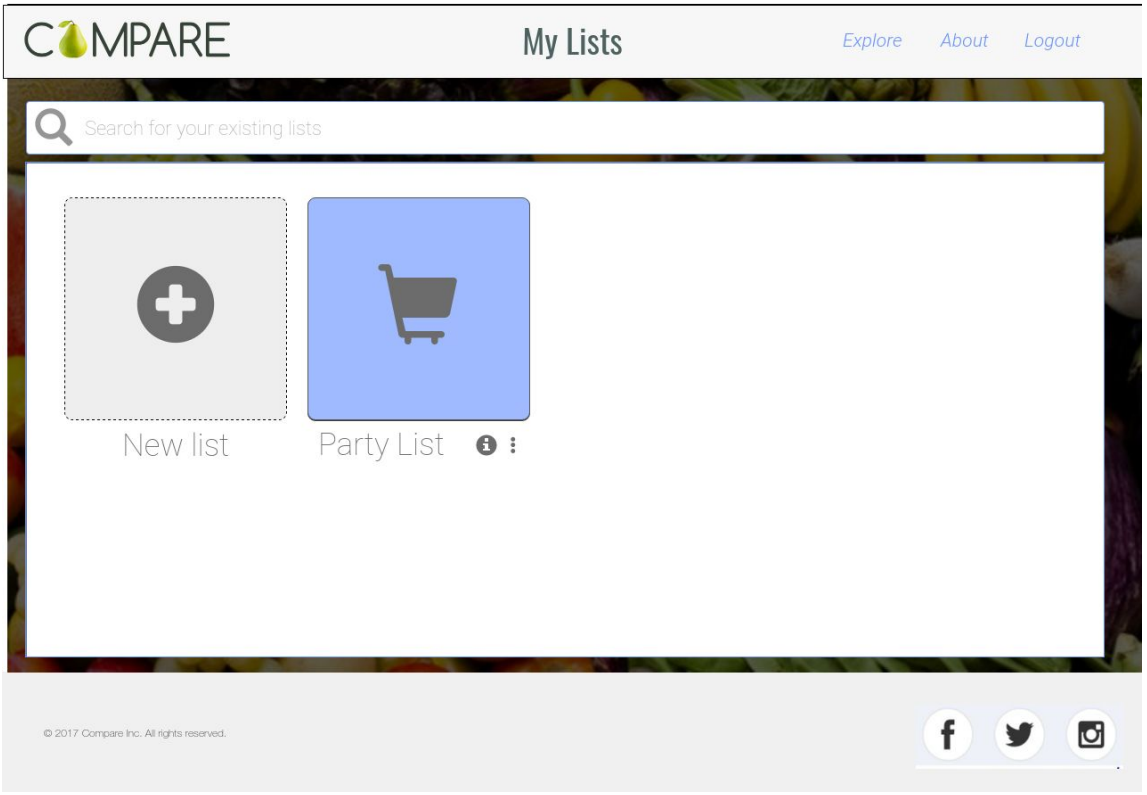


Figure 19: High-fidelity list management page (with newly created list)

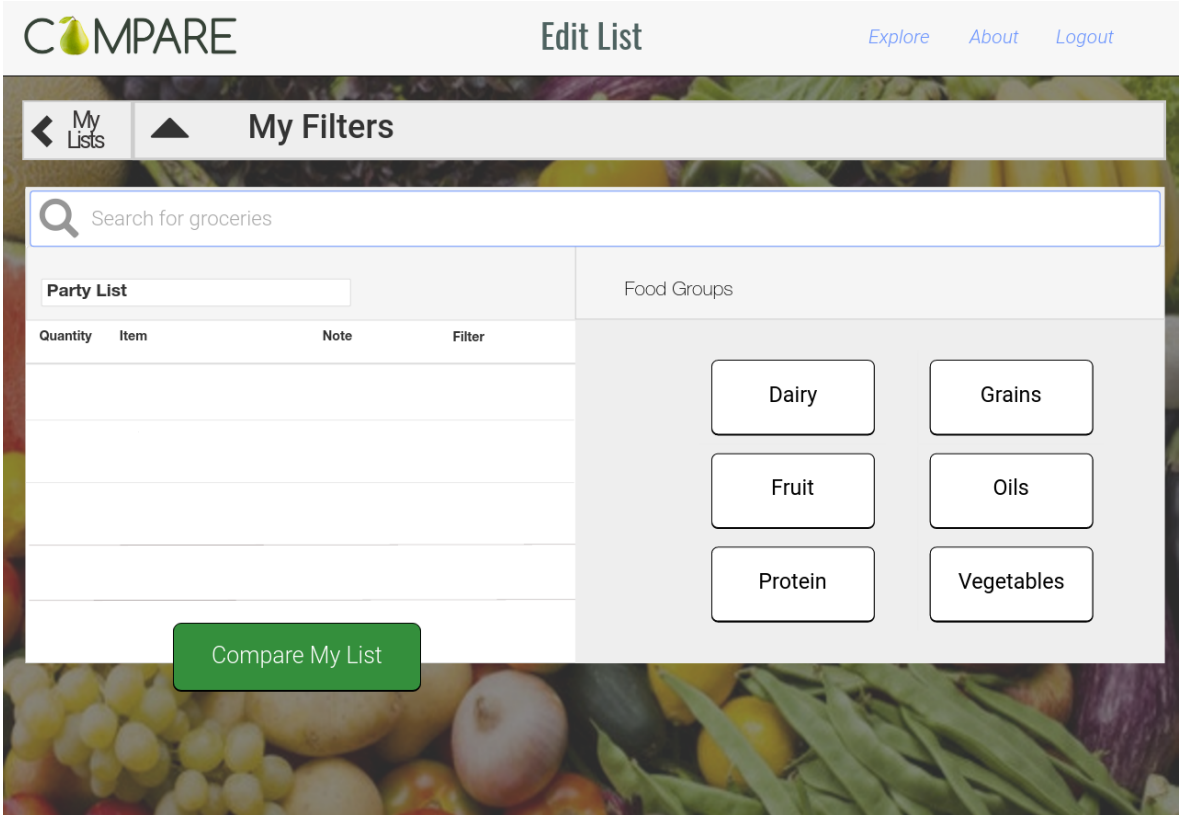


Figure 20: High-fidelity list editing page (overview with empty list)

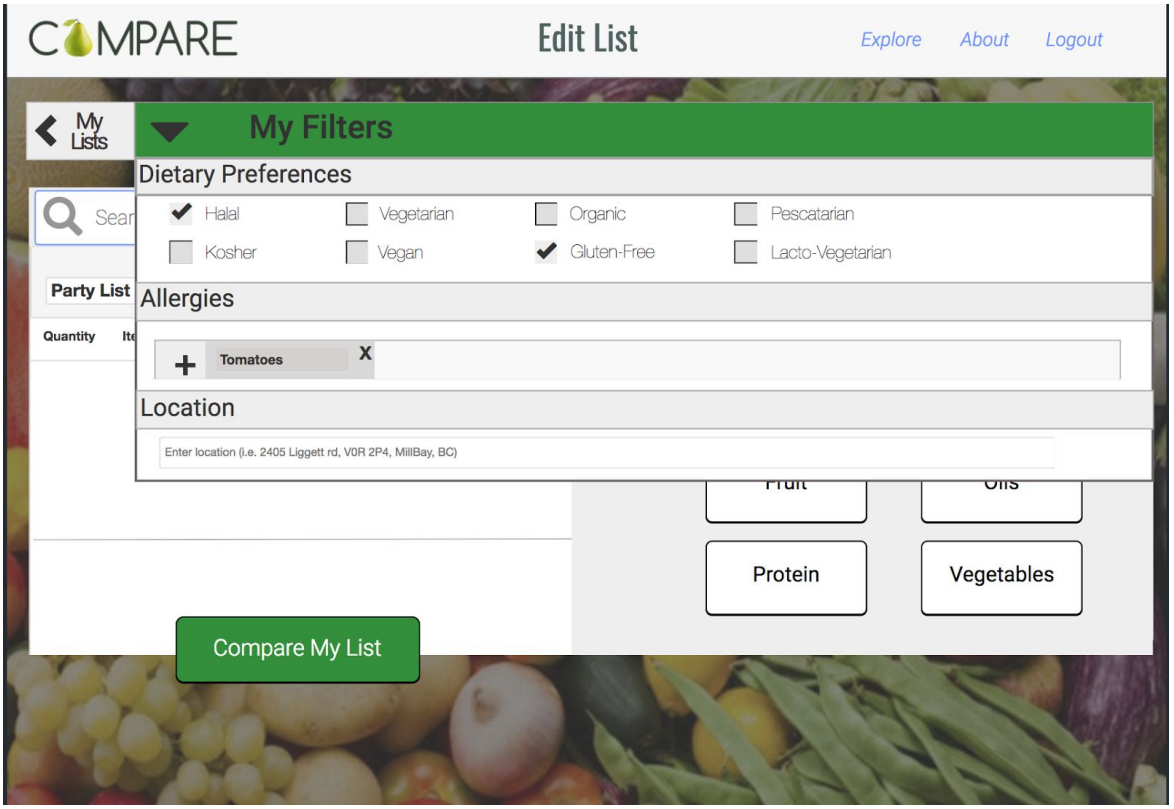


Figure 21: High-fidelity list editing page (with filter pop-out)

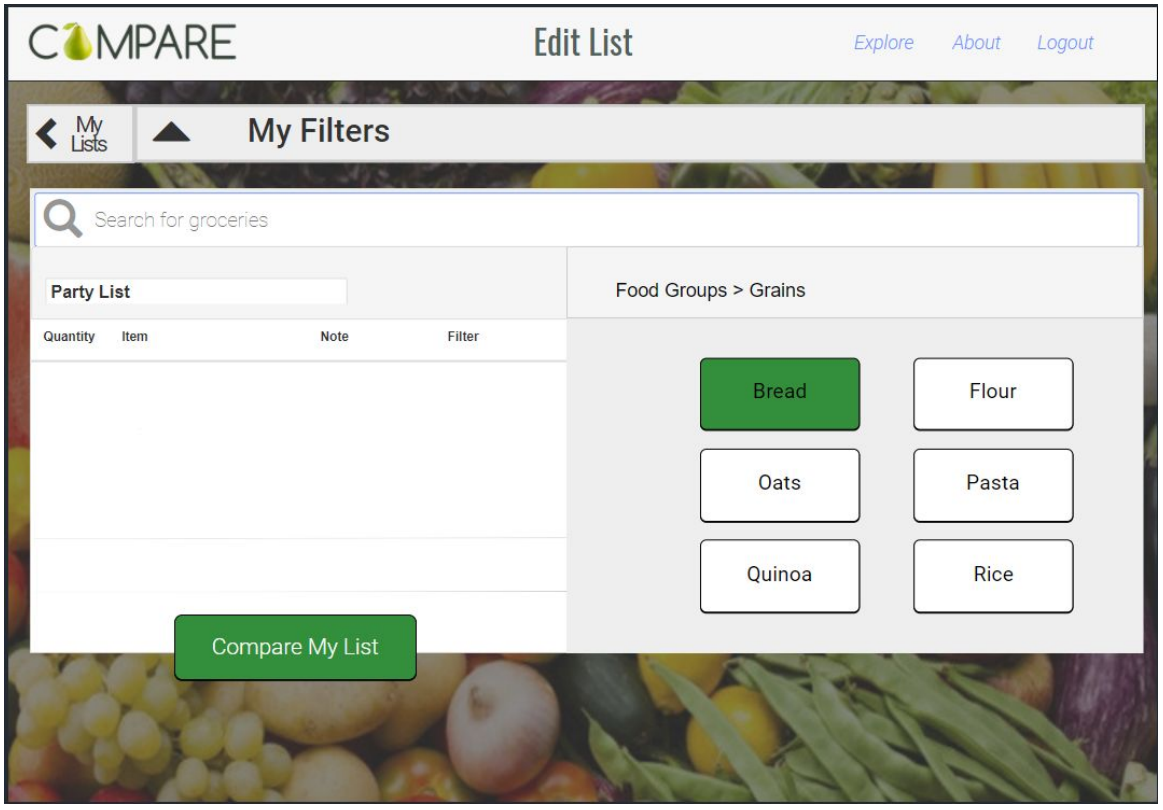


Figure 22: High-fidelity list editing page (grains)

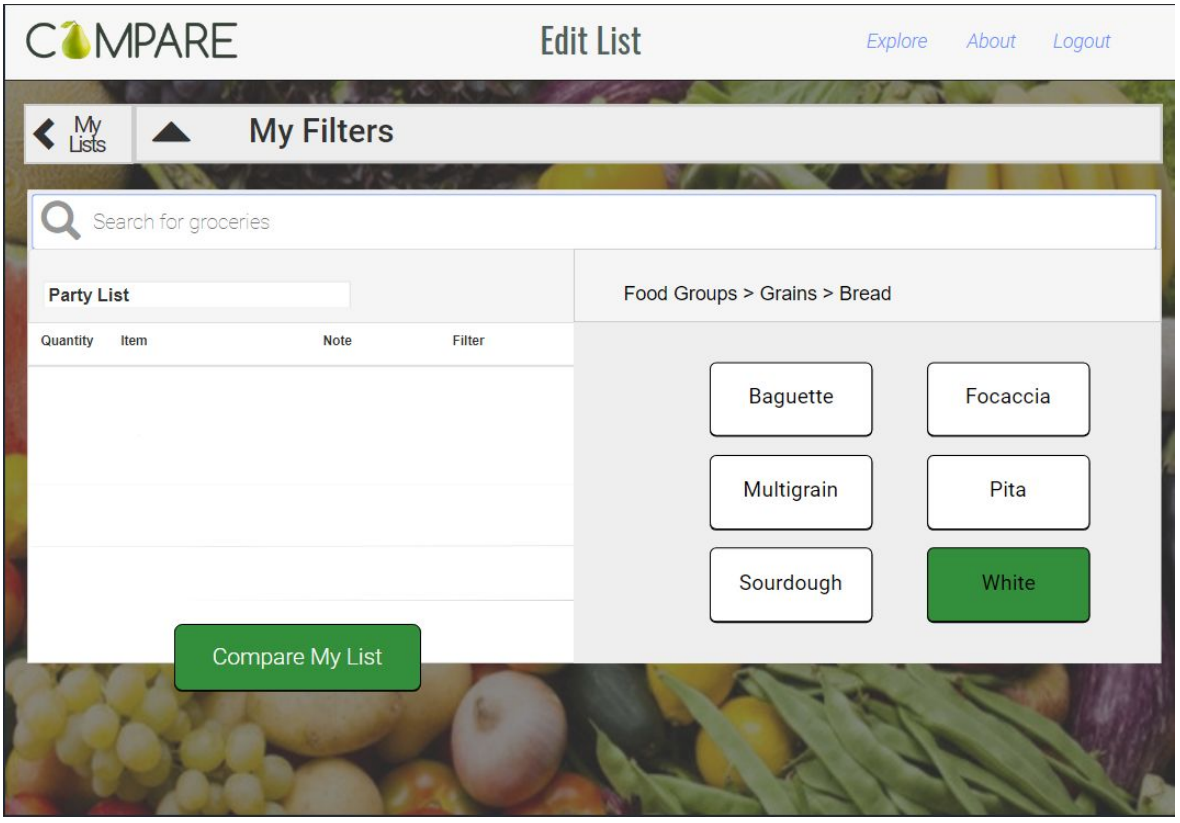


Figure 23: High-fidelity list editing page (bread)

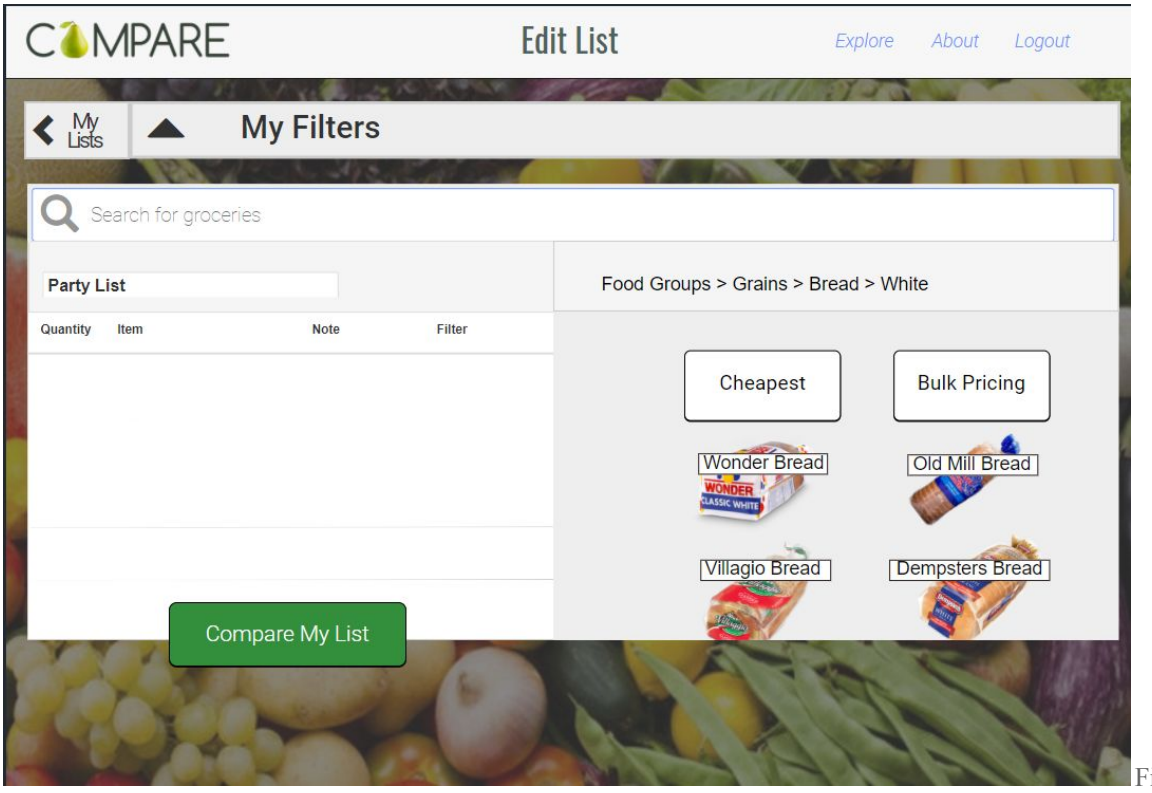


Figure 24: High-fidelity list editing page (white bread)

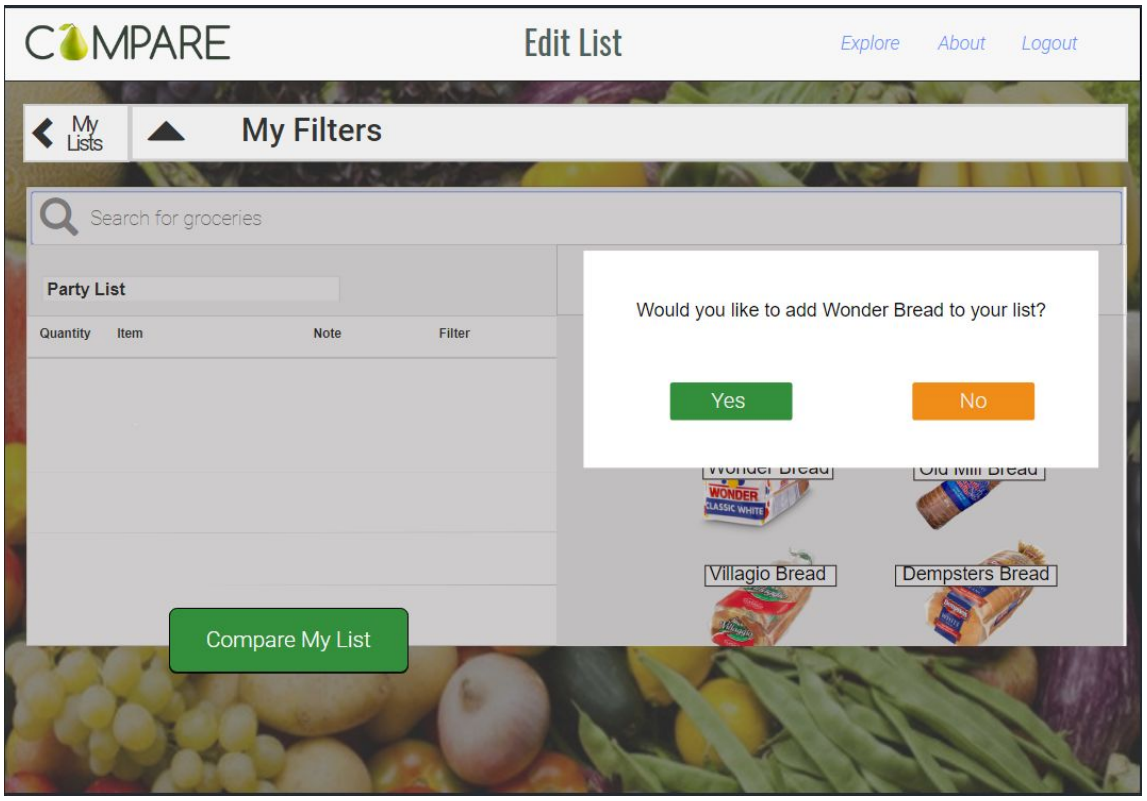


Figure 25: High-fidelity list editing page (adding Wonder Bread)

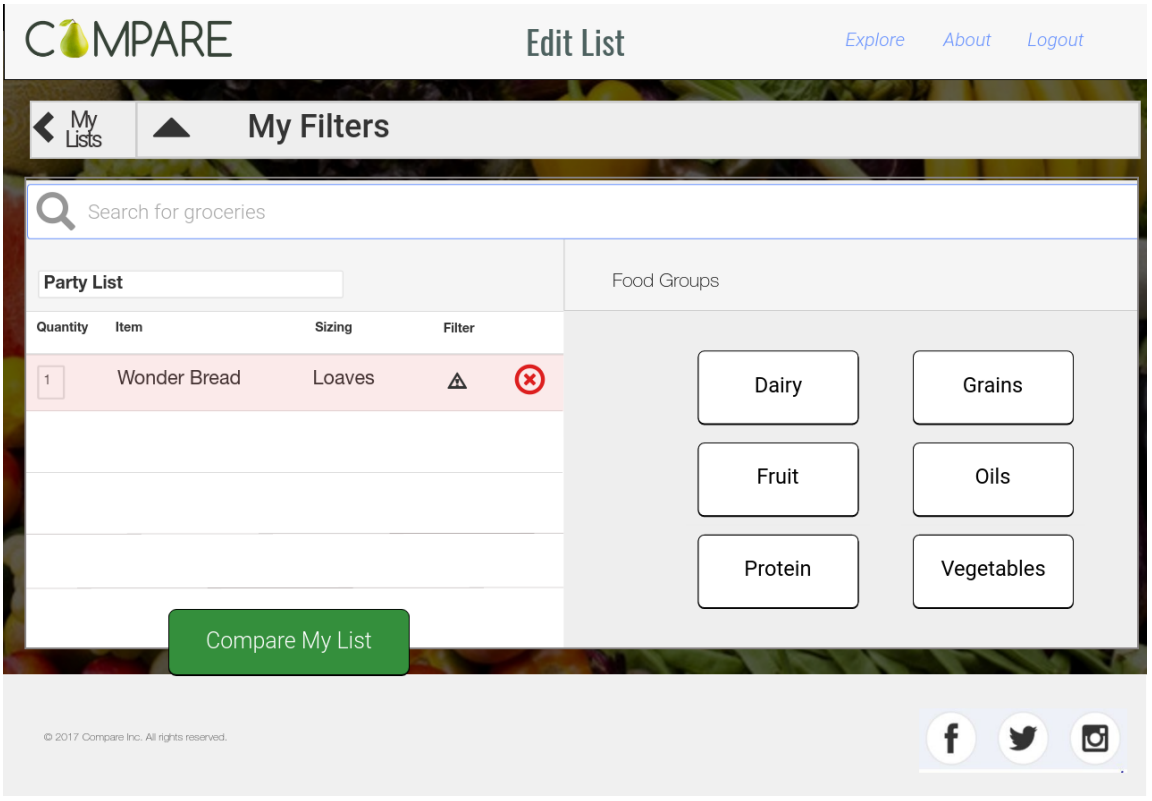


Figure 26: High-fidelity list editing page (overview with added bread)

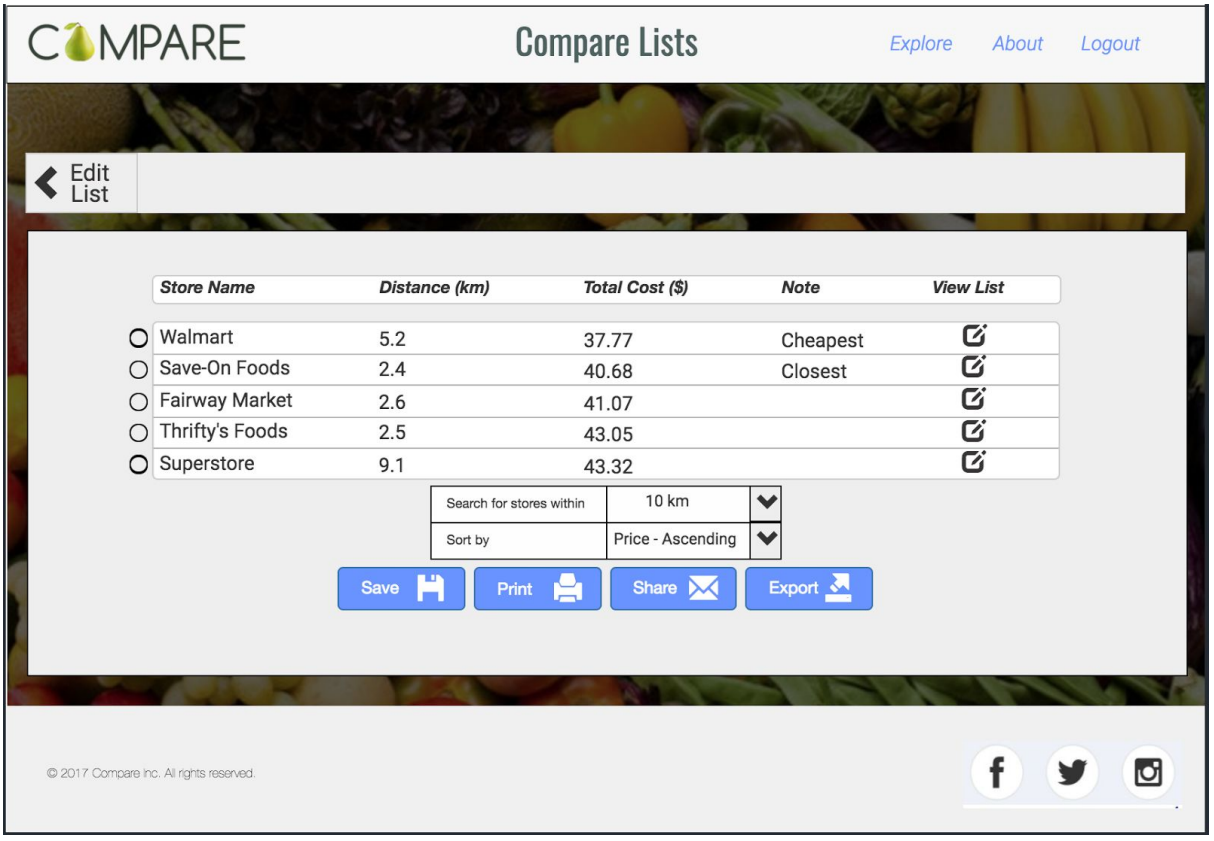
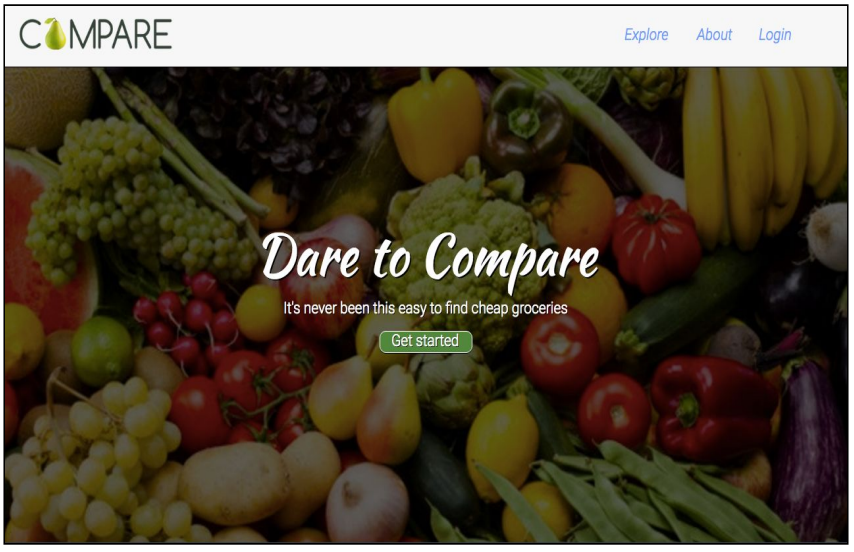


Figure 27: High-fidelity comparison page

Complete High-Fidelity Prototype

The complete high-fidelity prototype can be viewed at the following link: https://app.atomic.io/d/3TCYhR9ZYZXC.

Please note that there are some limitations to this demonstration as only pages with specified entries are accessible. For example, the *Explore* page is limited to only viewing Cocoa Puffs.



Part 3: Present and Future

Future Work

At the conclusion of this term, we are pleased with our final submission of the prototyped version of *Compare*. Of course, if we were to continue working on this project (to eventually release it as a fully-working product), we would need to put in some extra work.

Possible design modifications and improvements to our current high-fidelity prototype could include some of the following:

- Adding items to an existing list from the Explore page.
- Allow users to apply filters to Explore page/items.
- Show nutritional information and ingredients for listed items explicitly.

Each of these improvements would require additional usability studies to inform design changes.

If we were to take this project out of the design phase with the intention of creating a working implementation, there are a number of technical details we would need to consider.
The largest challenge would likely be maintaining a reliable database of products and prices for every store. It is crucial that when a user creates a list on the website, it reflects the actual price they will pay in the store. However, the data required to achieve this level of reliability may not be available to a third party website. It may be possible to overcome this challenge by collaborating with the grocery stores (to some extent).

Lessons Learned

Some of the major obstacles faced over the duration of this project were

- coordinating schedules as a team,
- gathering a user base for our studies and evaluations,
- learning new prototyping tools,
- finding a collaborative prototype building platform, and
- communication amongst team members.

The first major issue we faced was coordinating schedules amongst our five group members. After discussion and comparing schedules, we agreed to have weekly Sunday meetings as that was the only day where all five group members could meet.

Second, during our requirements gathering and usability study we had to recruit users to survey. We resolved this issue by reaching out to friends and family over social media to complete our requirements gathering survey. We selected a few users from this pool of

survey participants that were representative of our projected users to complete our usability study. The information we gathered was used to create our low-fidelity and high-fidelity prototypes.

A third challenge was learning how to use new prototyping tools. For our work, we used Balsamiq and Atomic to develop and design mockups. Learning to use these tools was heavily based on discoverability and was time-consuming to complete repetitive tasks.

After completing our low-fidelity prototype, we ran into a fourth obstacle when trying to find a high-fidelity prototyping platform that would allow collaboration. We tried several platforms (like Webflow, Axure, and Framer), but many had short trial periods or did not allow for team collaboration. Atomic, which offers a 30-day trial period and allows team collaboration, was suitable for our needs.

We ran into a fifth issue when designing our high-fidelity prototype. All group members participated in the process and were initially assigned to create certain sections of the website. A lack of communication amongst group members caused us to create a very divided website with an inconsistent design. As we did not make any decisions on the overall look of the website beforehand each member created sections of the website with his or her own ideas in mind. We resolved this issue by discussing design guidelines and agreeing on an implementation for our website. Communication allowed us to create a consistent and unified high-fidelity prototype.

All of these challenges tested us and forced us to learn how to

- coordinate and work as a team,
- reach out to our community for help,
- learn new prototyping software,
- develop and evaluate our designs from the ground up, and
- communicate proactively when facing challenges within the group.

Ultimately, we feel that this project has informed us of

- the necessity to design iteratively,
- the importance of evaluating *without* users,
- the importance of evaluating *with* users, and
- the precision and skills necessary to design an efficient, effective, and user-friendly website.

Conclusion

Designing a grocery store price comparison website was an iterative process. We sought to solve a problem for budget-sensitive consumers with special consideration to students and low-income families.

Research into alternative solutions showed that a product as robust as our envisioned service does not exist. Instead, we found that shoppers looking for the best deals on their groceries are bogged down by a fragmented series of inefficient methods. Our plan was to develop a resource that minimizes the amount of time cross-referencing items and their prices while maximizing consumers' savings. With our website, users can

- directly compare grocery items by their unit prices,
- compose grocery lists,
- compare grocery lists among nearby stores, and
- print, share, download and save grocery lists for future usage.

Next, data gathering was done to verify our initial assumptions. A survey (see Appendix for survey and results) was conducted on friends and family. With an idea of our user base, we proposed personas, scenarios and eventually a set of use cases to build our project around and to justify future decision-making.

Following, a low-fidelity prototype developed for our targeted audience and our use cases were dispatched to our peers and lab instructor for review. Feedback at this point offered ways to improve our website; constructive criticism was met with a plan for change as we began to develop a medium-fidelity prototype.

Once satisfied with our revisions, we performed usability testing with our evaluation plan and newly crafted prototype. Observation of our users handling website tasks suggested modifications to our design. We emphasized efficiency with redesigns, opting to replace confusing structures before designing a high-fidelity mockup.

Overall, this project was extremely successful in applying the concepts learned in class to better understand the importance of human-computer interaction design. Although there is some work that would need to be done before our product could be unleashed into the real world, we are pleased with our submission.

References

[1] Balsamiq. “Balsamiq Mockups.” *balsamiq.com*. [Online].
Available: https://balsamiq.com/products/mockups/ .
[Accessed: July 25, 2017].

[2] Atomic. “Cloud Prototyping for Product, UX and UI Design.” *atomic.io*. [Online].
Available: https://atomic.io/ .
[Accessed: July 25, 2017].

Appendix

In order, we have attached

- our survey and its results,
- our evaluation plan for the usability study,
- our low-fidelity prototype (followed by an early design mockup), and finally
- our medium-fidelity prototype.

Note that all iterative work is available for reading at the following link:
https://drive.google.com/open?id=1ErygPnkGa6qwpAoLFtwEsJJ62E4R3sNtWUxcr8fPwro

And again, our high-fidelity prototype is available for demonstration here:
https://app.atomic.io/d/3TCYhR9ZYZXC