



1. Header-project name

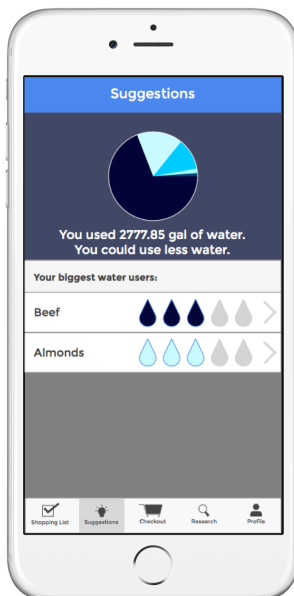
Mark Xue, Maia Miller, Jason Platt, Xin Jiang

Flood tracks the water impact of your groceries so that you can make informed decisions.

2. Problem and Solution Overview

We unknowingly consume over $\frac{2}{3}$ of our total water usage through the groceries we buy. “Flood” is a mobile-based shopping app that calls to attention the environmental impact of food, encouraging users to make greener choices in their grocery shopping.

Our basic approach is to allow users to easily access information about particular items as well as track how much water they’ve saved over time.



3. Tasks & Final Interface Scenarios

Our approach revolves around three user tasks:

1. Researching a particular item
2. Tracking progress over time
3. Looking at suggestions

The research task is the core feature of the app. Our app exists to help users make informed decisions about their shopping habits and there can be no informed decision making without easy access to the correct information. Therefore the research task is both our most important, and our simplest, task. The research of a particular item is accomplished if the user decides they want more information concerning a particular article. The user will click the research tab button and then type the name of the food in the search bar. A list of matching items will then appear, which the user can select to find out more about water consumption as well as a host of nutritional facts and other information. Some suggested alternative items will also appear to help the user create the most water efficient shopping list possible. Using the barcode scan feature, users can also access the research pages of items in the store without having to physically type in every item they are interested in researching. This allows for spur-of-the-moment decisions regarding the water usage in their shopping trips.

While our app mostly exists as a way for environmentally conscious people to make informed decisions, we also want to help them along by encouraging good habits. Part of this is showing them how much water they've used and pushing them to improve. At the end of every shopping trip the user will be prodded to scan a QR code on their receipt, which will then show them a pie chart highlighting their most water intensive purchases for that day. In the profile section there will also be a line graph showing water usage over time. Users can set personal goals for their water usage per shopping trip, which appears in the line graph, as a way of keeping themselves accountable.

Our last task is to search specifically for suggestions to reduce one's water footprint. If someone wants to reduce their water consumption but cannot figure out the best way to go about doing so they can go to the suggestions tab where they will be shown water friendly foods to add to a list based on their preferences and previous shopping trips. Users can also click on these suggestions to get more information from their "research" pages.

4.Design Evolution

Initial Sketches:

From our initial sketches we had two separate designs to choose from (FIG 1 and FIG 2). We designed three tasks for each sketch.

The simplest task involved conducting water usage research on a particular food item. The main difference between the versions was how users can search for water consumption information. In our first design users perform a traditional search by typing items or categories into the search bar. The other design features a bar code scanner.

Our second, more complicated task, was to make a shopping list. The main difference between the versions was the form of the shopping list. One featured a simple list with warning symbols that flag items with a high water footprint. The second design featured a dynamic pie

chart that tracks the water consumption of each item in addition to the basic list.

Our last (moderate) task was allowing the user to track their progress. One uses a bar chart showing the water impact of each trip while the other builds up a unique user profile and uses a line graph to track the total water consumed over time. The main distinction between the two designs was the location of this progress bar: the first design had a simple bar chart with very few extra information included, while the second included the progress bar as a subset of a general “profile” page for the user.

The final low-fi prototype was a combination of the two versions. Considerations in making the design decision include:

- 1) Exposure effect: people prefer to do something that they are familiar with. Use established UI techniques such as swiping.
- 2) Data visualization makes numbers more intuitive and forces user to see their environmental impact.
- 3) Easy switching between different tasks helps users navigate the app.
- 4) Keeping the app focused on the core idea, a shopping list, helps usability.

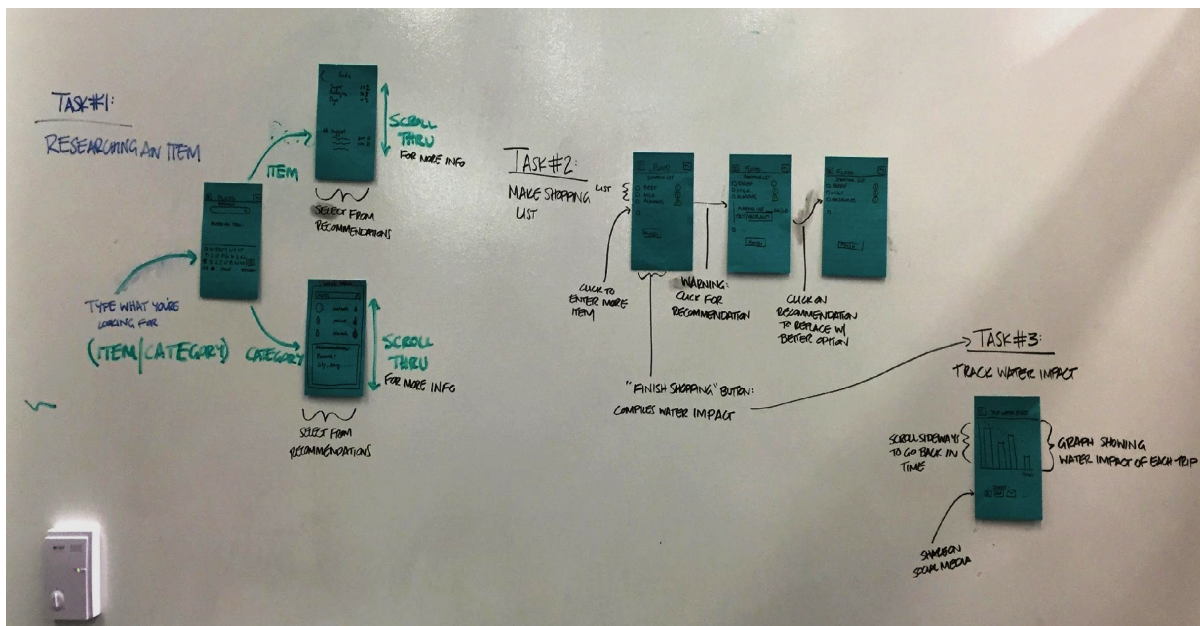


Figure 1: Our first initial sketch

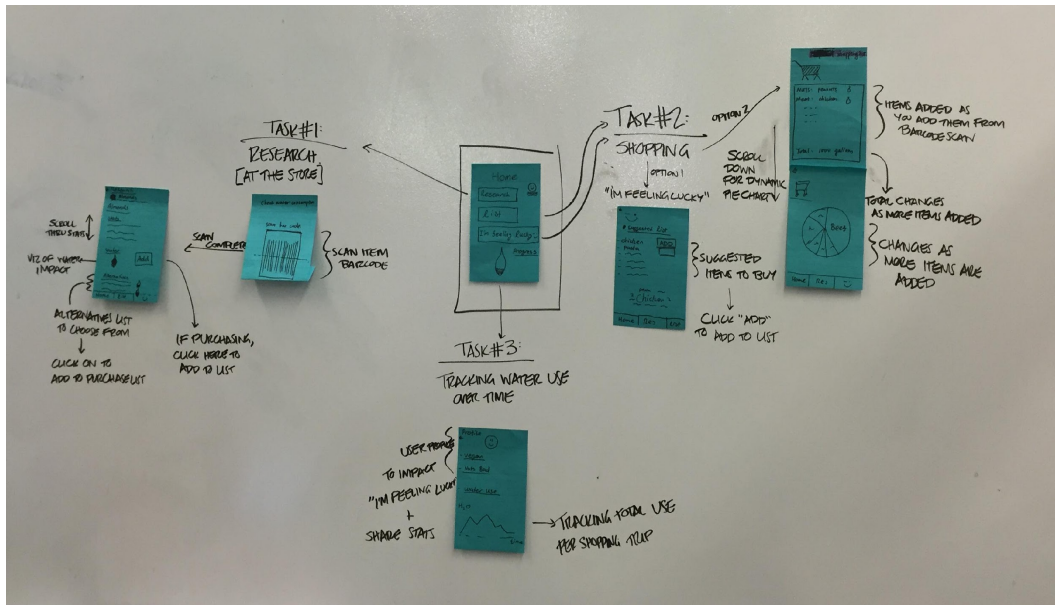


Figure 2: Our second initial sketch

Low-Fi Prototype:

Evaluation:

For our evaluation we made a low-fi prototype on index cards (FIG 3). We highlighted the interactive features of the app and had a human “computer” switch out cards (FIG 4). Our three tasks were the same as the initial sketches. The feedback we received from user testing was invaluable and enabled us to refine our ideas.

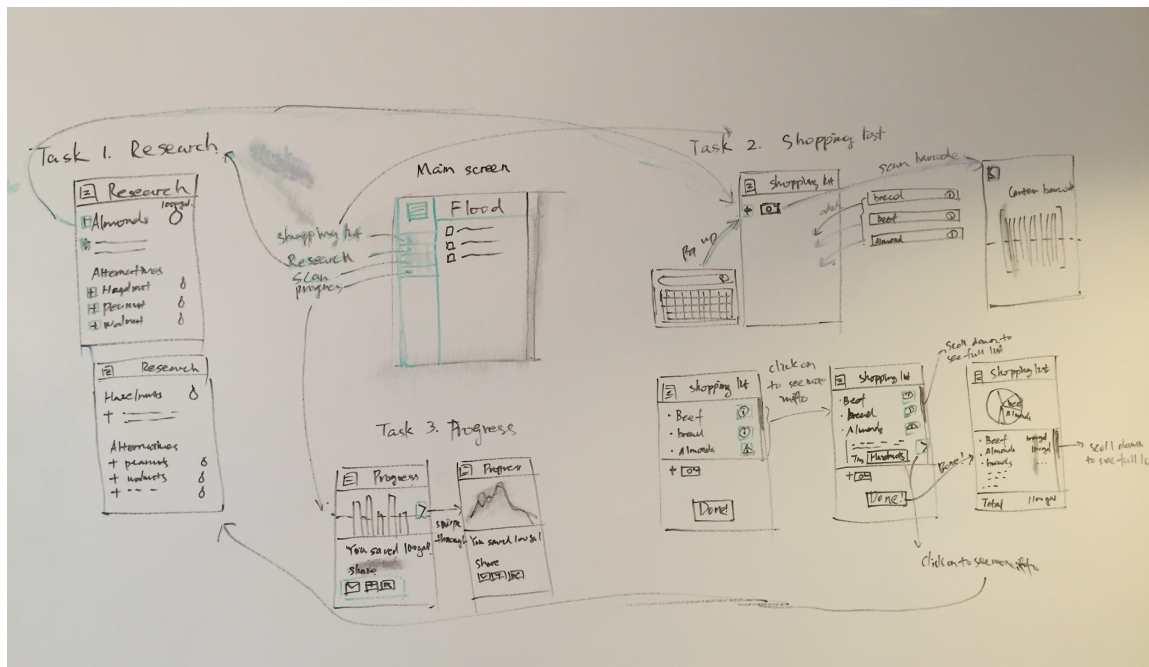


Figure 3: Our low-fi prototype design



Figure 4: Implementation of our low-fi prototype

Most people expressed interest in the app and overall had generally positive things to say. All of them enjoyed the barcode scanner option with one of them exclaiming “so cool!” when the screen appeared (FIG 5). The graphs were found to be “interesting and useful” by the majority of participants and the app straightforward and intuitive. The hamburger button especially was a big success since many people had seen a similar UI before and knew exactly how to navigate around with it.

Having said that, many of the users felt “stuck” on certain screens, especially in the research tab. One of them suggested an extra button to add the item they were viewing onto the shopping list. Other issues we noticed were that no one ever went to the progress bar. This page was accessible on a separate page in the hamburger menu, or by clicking the “finish shopping” button on the shopping list. Users were either uninterested in this feature, or did not notice it at all.

The last piece of feedback we received was that the icons for our buttons were confusing. Many thought the barcode scanner icon, which was originally in the shape of a camera, was a way for the user to take pictures of the items they were adding to their shopping list. Also the + button to bring up the keyboard was not obvious enough for many people. A better idea would be to have the keyboard pop up automatically and have people type their list right away.

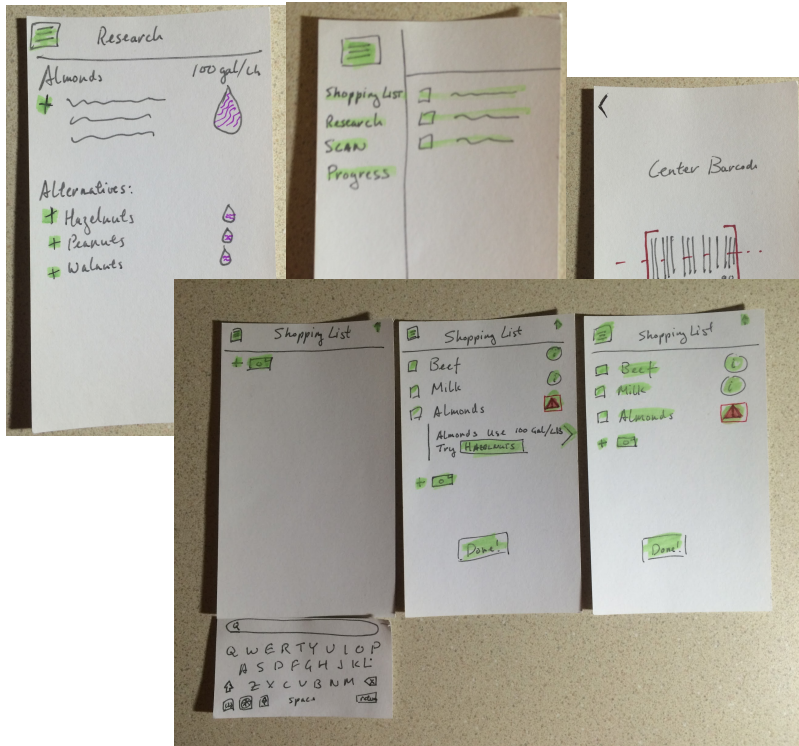


Figure 5: Sample screens from our low-fi prototype

Medium-Fi Prototype:

Our medium fidelity prototype had the same basic structure as the low fidelity version but incorporated the feedback we received from users during the previous round of testing. The changes we made were mostly to refine some of the ideas explored in the low fi prototype.

Change 1: We changed the progress tab to a profile tab that pops up when a user clicks the “finish shopping” button (FIG 6). This ensures that users see their progress, even if they don’t actively go to that page.

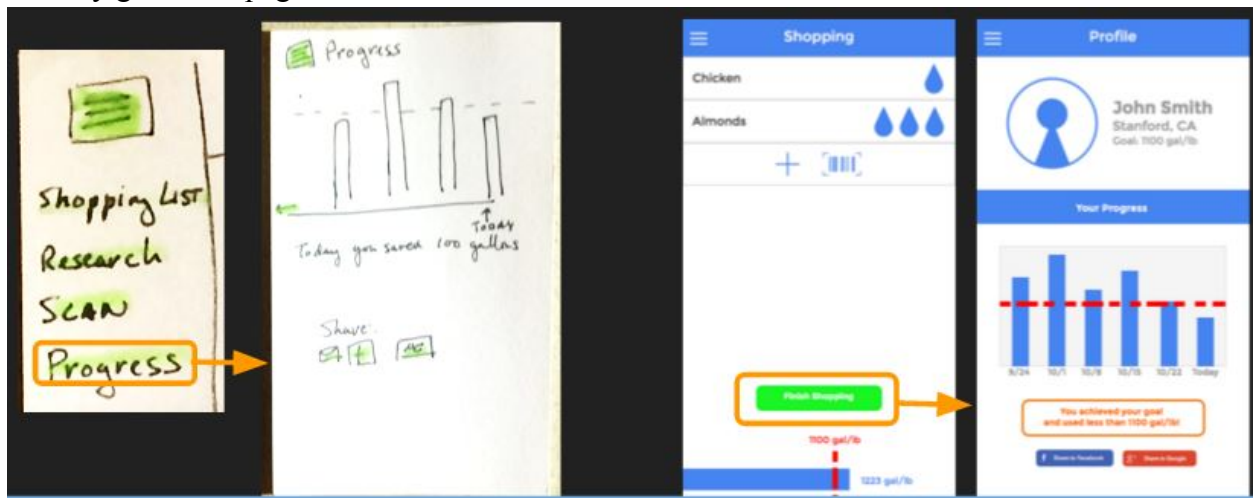


Figure 6: Change from low-fi “progress” tab to medium-fi “profile” page.

Change 2: We changed the pie chart (accessible via the “finish shopping” button) to a dynamic progress bar so that users can see more clearly how much water they are using with each item they add to their shopping list (FIG 7).

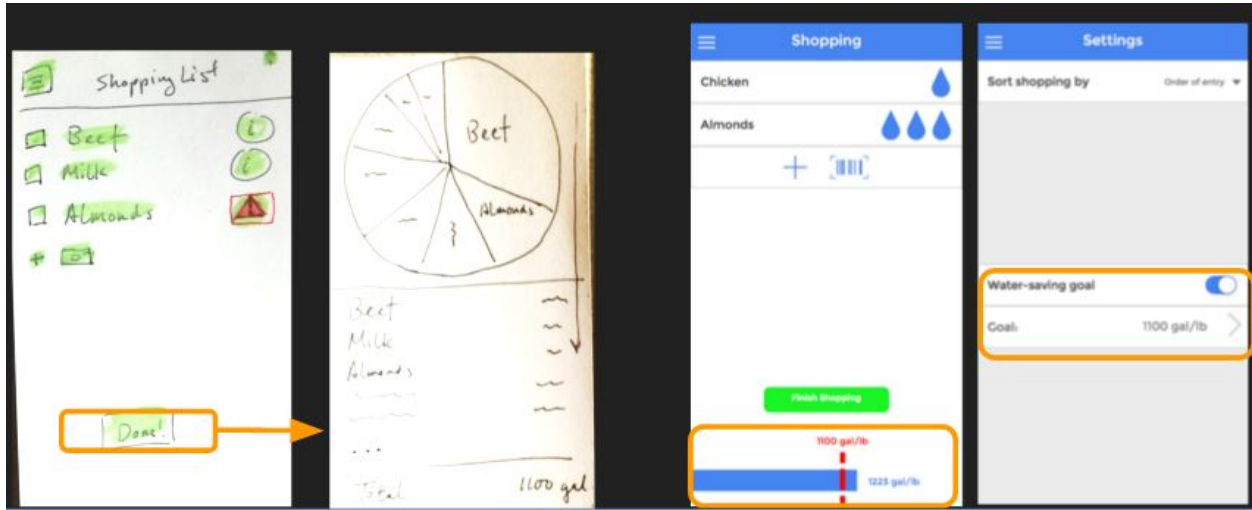


Figure 7: Change from a pie chart ending the shopping trip to a dynamic bar chart.

Change 3: Simplified the icons to make the tasks more intuitive for the user (FIG 8). This included changing our barcode scan icon from a camera to an icon resembling a barcode scanner. We also decided to encourage the user’s research on an item by changing the “information” button next to each item in the shopping list to a water drop ranking based on water usage per pound of that product produced.

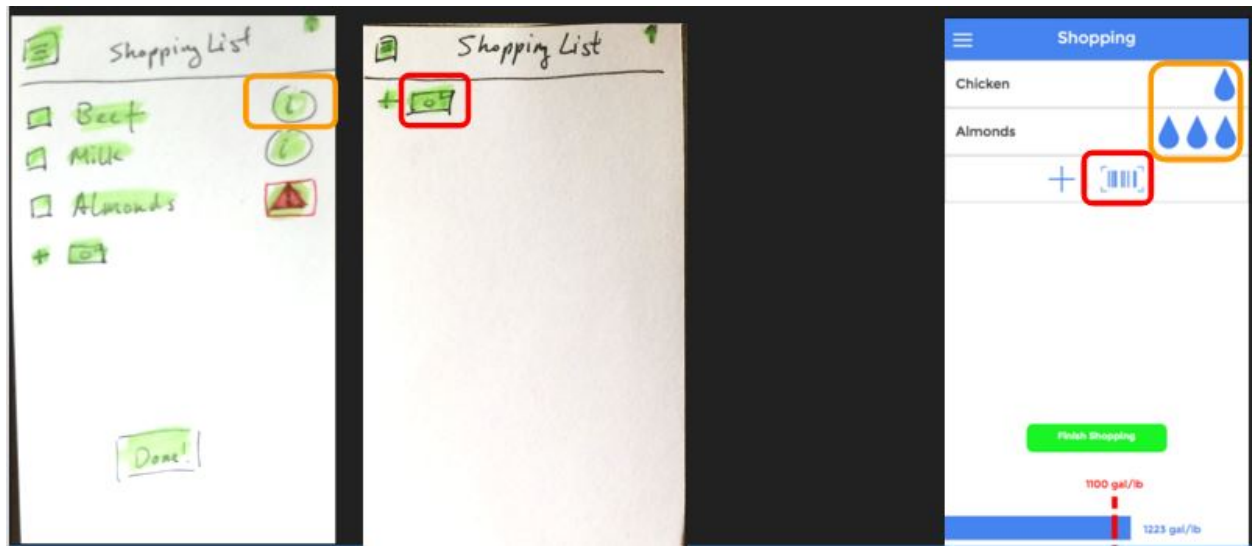


Figure 8: Simplifying the icons on our items and tasks.

Change 4: We decided that instead of including both a bar chart and a line graph in the “progress” page, we would use only a bar chart to track the user’s progress (FIG 9). We found that the bar chart easily and accurately communicated both the usage of each shopping trip, and the user’s overarching trend throughout time.



Figure 9: The shift from both a bar chart and line graph to a basic bar chart.

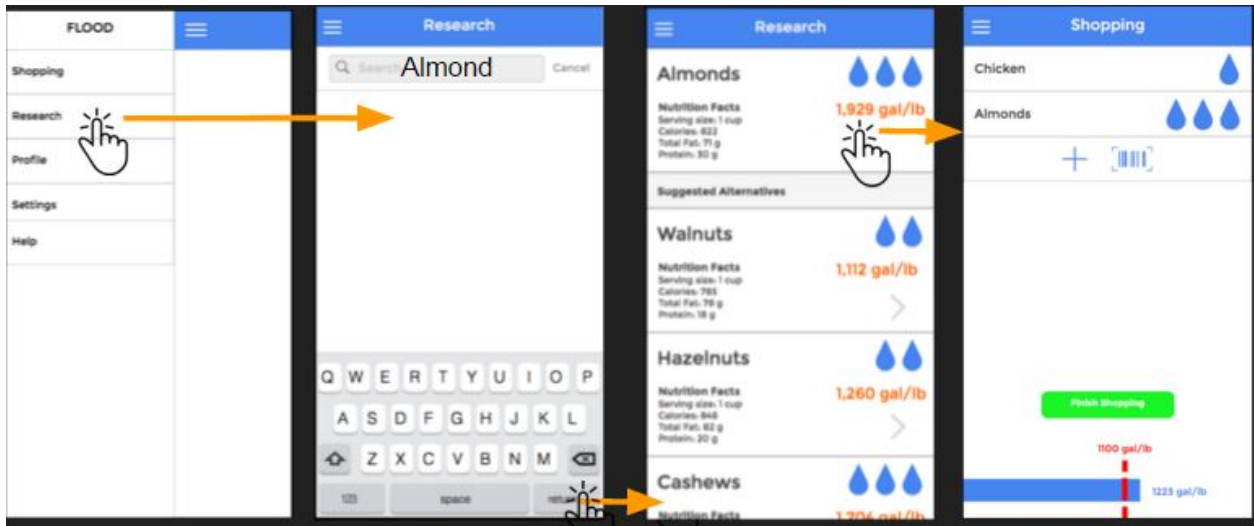


Figure 10: Task 1, researching items

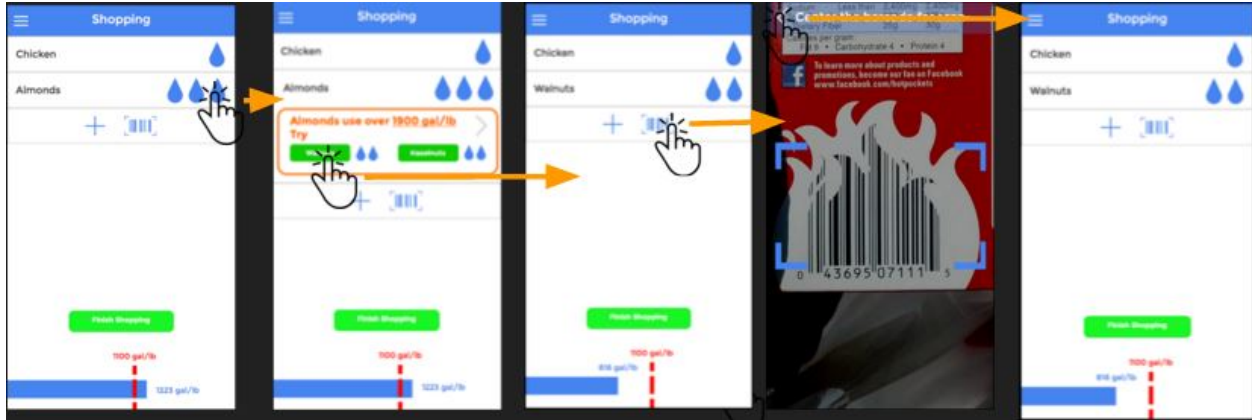


Figure 11: Task 2, creating a shopping list

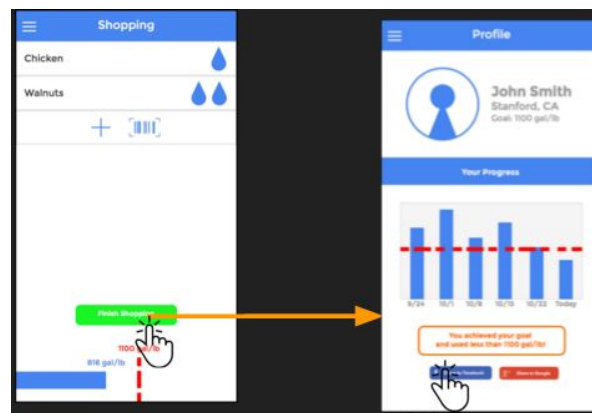


Figure 12: Task 3, tracking progress

5. Major Usability Problems Addressed

Overview of design changes:

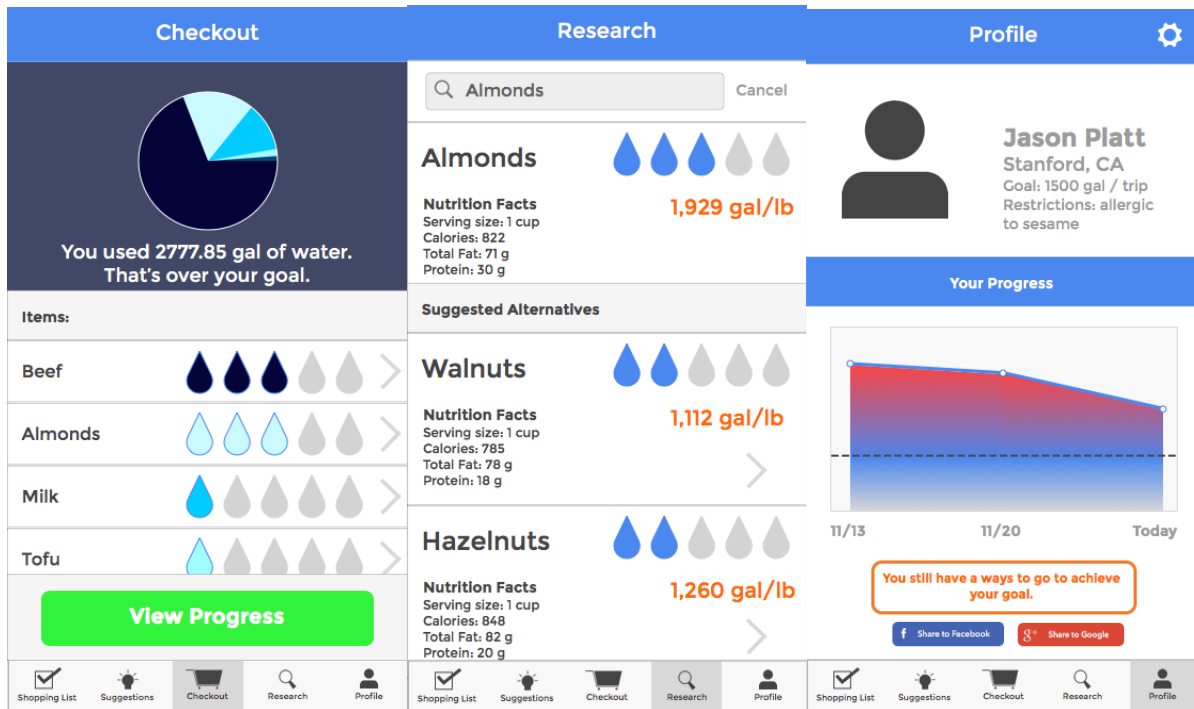
After the feedback from the heuristic evaluations we decided to radically alter our design. Instead of an app based around creating a shopping list, our app is now a resource to help shoppers research and keep track of water usage in their grocery shopping. Our three tasks have changed to:

1. Research water consumption of specific foods
2. Get feedback on water usage of each (and past) shopping trips
3. Get suggestions on altering food consumption for future shopping

Almost all of the severity three violations have to do with creating and manipulating the shopping list. Since we decided to jettison the shopping list many of these violations are no longer applicable. We will therefore provide a complete overview of the design changes, then specifically address the violations that were applicable to our new design.

Instead of having users enter in a shopping list, they now simply scan a receipt at the end of their trip (FIG 13). This will then enter their data into a pie chart that gives them a complete overview of their trip. Under the chart will be the shopping list and how much water each particular item used in case the user wants more detailed information. Each trip is saved in the profile tab with a line graph showing the overall trend. From our medium-fi evaluation we found that users would never make a shopping list with exact values, and are even less likely to add items while they are in the store. Additionally, we found that the method of tracking progress (by gallons of water per food purchased) did not represent this information accurately enough to the user to make active changes in their shopping. With this change, we would be able to provide more detailed feedback regarding which item actually used the most water in each trip.

Based on previous shopping trips the user can then look for suggestions to lessen their water impact. A user concerned about their water use can go to the suggestions tab and see what items they purchased are the biggest culprits and what they can change to be more eco friendly. This is in addition to the research tab which has stayed the same since the medium fidelity prototype.



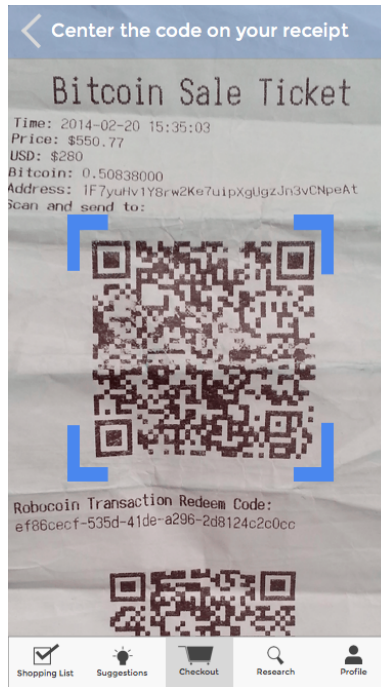
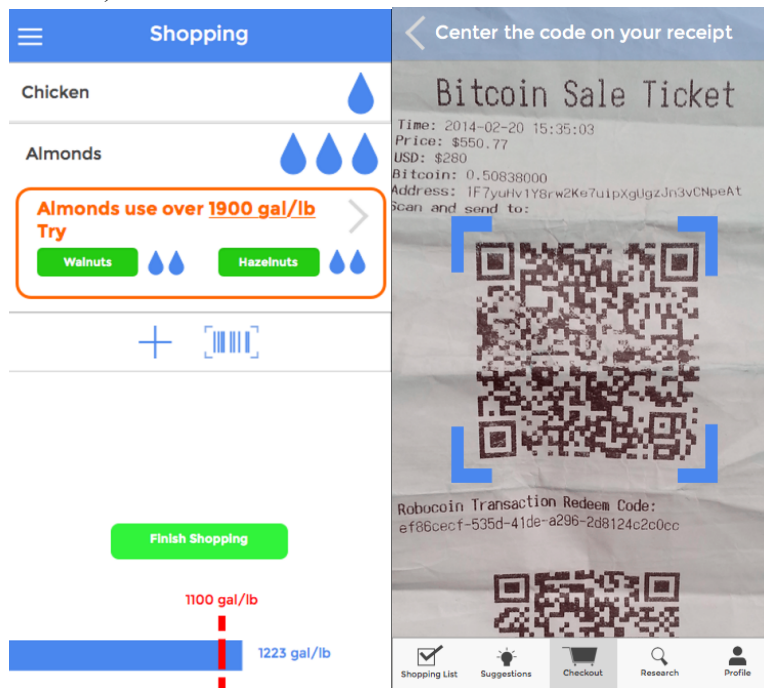


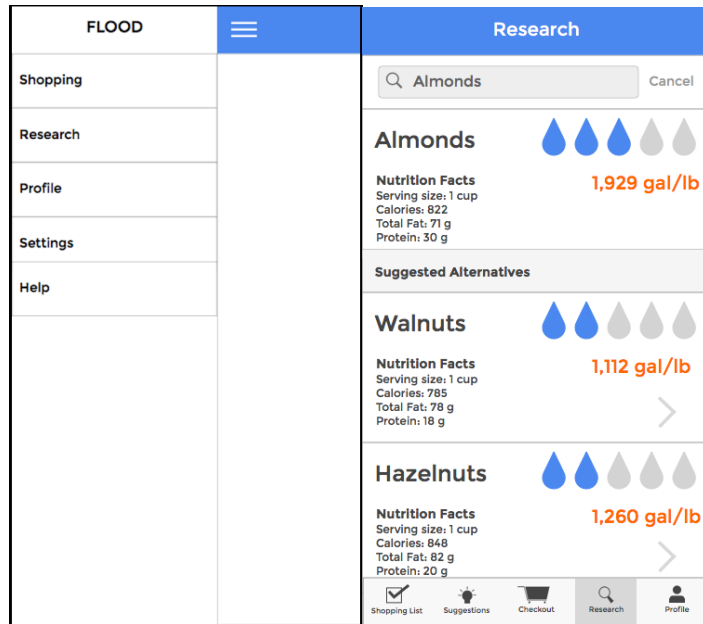
Figure 13: Our new Task 3 (tracking progress)

Major changes:

Change 1: track over shopping experience---> scan receipt after shopping (it's burdensome to enter the quantity of items)



Change 2: Hamburger button ---> tabs



Change 3: profile screen:



Relevant HE Violations:

[H2-4 Consistency and standards]: “When going to the research screen from the menu, there is an option to type the name of the product. However, the functionality to scan an item is not present. Thus, this is not consistent with the shopping list screen functionality of being able to scan an item. Fix this by including the scanning barcode functionality to the research screen.” [Severity: 3]

Fix: Added a scan button as well as a search bar in the research screen.

[H2-2 Match between system and the real world]: “In the shopping screen, different items have different number of water drops. It is clear that the drops are trying to convey how much water they use relative to the other items. However, it is not clear what a water drop means in terms of real world numbers. Is there a specific range of gal/lb that a single drop of water represents? This can be fixed by providing documentation, a tutorial, or legend that quantifies how much a water drop means. For example, the app can display information such as: one drop = 150 gal/lb.” [Severity: 3]

Fix: Added the number in gal/lb under water drop. Additionally, included faded gray (“unfilled”) water drops to better indicate the ranking scale (out of 5).

[H2-2: Match Sys & World]: “There should be some way to allow users to differentiate water intensity across brands. I acknowledge that this is an early prototype and not everything has been implemented yet, however there is no indication in the prototype that such a function will exist or not. For example, the prototype has “chicken” as one item. This is extremely generic, as chicken can come from many different places. I assume the statistics shown are an average. A user might find him or herself deciding between multiple brands of chicken. One brand might come from a farm that uses much less water than another brand’s farm. A function to either search the brand or scan the barcode would therefore be useful. This can be implemented by adding a button to do so for each item (for example, once at the store I could click on “chicken” in my list and then have an option to further specify brand).” [Severity: 3]

Fix: With more research we may be able to differentiate between brands but that would come in a later update.

[H2-1 Visibility of Status]: “It is unclear what happens to your data when you finish shopping. Is your shopping list saved somewhere? Does it just become that line on the bar graph. The app should say: “Your list has been archived”. Or if it’s not archived, that’s another violation.” [Severity: 3]

Fix: Though the shopping list is no longer a key task, the “checkout” tab clearly indicates a QR scan of the receipt, which results in a detailed pie chart and data summary of the water usage of that shopping trip. There is an automatic shift to the “profile” page, which includes this new shopping trip as a data point.

[H2-5 Error Prevention]: “After I click the water droplets in the ‘Shopping’ screen, I can see a couple of the top alternative items. When I clicked ‘Walnuts’, I expected to be taken to a screen where I could see more about walnuts (such as their water footprint and nutrition facts). Rather, I was surprised to find that they immediately replaced almonds in my list. Also, there is no way for me to undo this action. Add a small popup that displays deeper information about each of the top choices (whichever one I click on) and give me the option to undo the change.” [Severity: 3]

Fix: Again, shopping is no longer a key feature of this app. However, we incorporated this feedback in the “research” tab – when users click on a “suggested alternative” to the item they are researching, they are brought to the information page of that item.

Obsolete from design change:

[H2-4 Consistency]: “What’s the difference between researching an item and adding an item to your shopping list? right now that seems to be two names for the same thing. Change this by either adding more functionality to researching that is different from adding, or by integrating the two fully.”

[Severity: 3]

[H2-7 Efficiency of Use]: “How does an experienced user add an item to their shopping list directly, do they still go through the whole process of searching for an item, and then adding it to their list, or can they recall previous lists? Experienced users know which normal foods are good on water consumption, so I would add a quick add functionality. [Severity: 3]

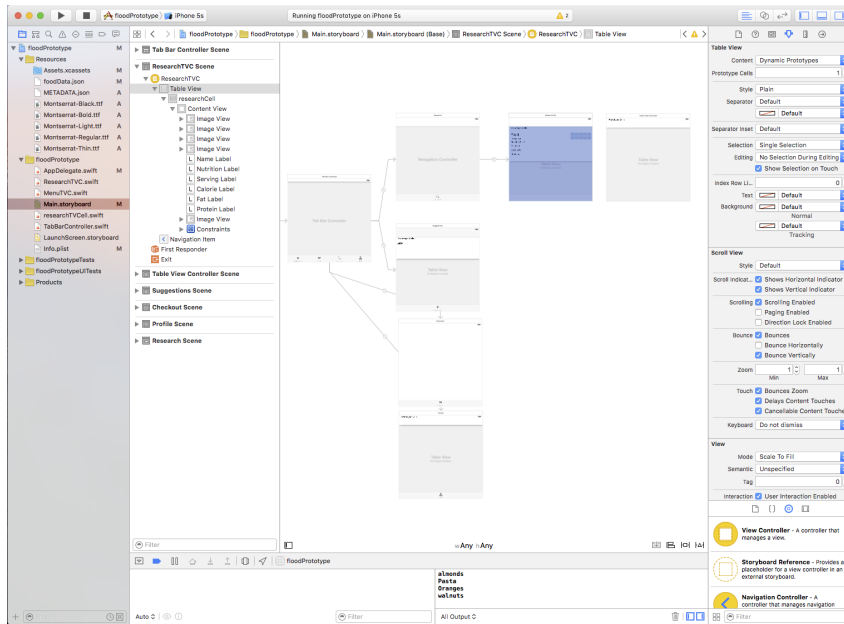
[H2-8 Aesthetic & Minimalist Design]: “As a user, I would expect the ‘Finish Shopping’ button to be at the very bottom of the screen, as that is how it is in many other common ecommerce platforms. However, I understand that because the bar graph displaying water consumption is dynamic, it would be awkward to put the ‘Finish Shopping’ button under the graph. I propose that we create the feeling that the bottom of the screen is at the ‘Finish Shopping’ button, and that the graph should not be clicked, by changing the color of the background behind the graph.” [Severity: 3]

[H2-4 Consistency & Standards]: “On the shopping screen, if the user clicks the water droplets, a dropdown menu appears, displaying a message showing how much water the almonds use, and then 2 suggestions below it. If the user clicks the message, he is taken to a research screen that shows the almonds and its suggestions in detail. However, if the user clicks one of the two suggestions in the dropdown menu, that item automatically replaces the almonds in the shopping list. This is a bit of inconsistency that should be fixed. I think they should all go to the research screen, where the user can then confirm their selection.” [Severity: 3]

[H2-5: Error Prevention]: “The barcode scanner function might be slightly confusing for users. Since this is a shopping list app, the list would presumably be made prior to going to the store. The barcode scanner would therefore be helpful for users who want to purchase more of what they already own (for example, if I really like a certain cheese I may keep the wrapper and then scan the barcode to add it to my list for my next trip). I think that many users may feel like they need to scan the barcodes of items in the list after arriving to the store, which is not the case. This can be fixed by adding the word “or” in between the button and the barcode scanner button, or by removing the scanner button completely from the shopping list and moving it within the screen that the plus button brings one to.” [Severity: 3]

6. Prototype Implementation

The prototype is built in Xcode as a iOS application for phones and tablets. As it is a full application development environment, it allowed us to build as much functionality as we desired - limited only by experience and development time. The default classes and prototypes allowed us to build the basic interface quickly, although there is also a bit of a learning curve and there are many UI quirks that are not easy to debug. The standard libraries also include the functionality to scan QR codes and read JSON-encoded data.



Wizard of Oz:

In reality, this application would have to be linked to an extensive and constantly updated database of UPC's and commercial products. The application would have a model layer that would sync to the database. For the prototype, we provide a limited set of examples. We created a hard-coded list of about 30 common food items that are stored as a dictionary in iOS's lightweight persistent storage (NSUserDefaults). We also created three sample receipts that the application will scan and produce pre-generated graphs and suggestions - the prototype does not currently have the ability to dynamically generate graphs and suggestions.

For future iterations, we would close the feedback loop by allowing users to add items to a shopping list from their suggestions or research tasks. We could also add a gamification element by devising a badge or reward scheme to encourage goal-setting. We would also more closely match the prototype to the color choices from the latest sketch to promote readability usability. We would also finish implementing dynamic reading of receipts and generating profiles and suggestions based on new input.

```

{"foods": { "almonds" : {
  "waterUsageGalPerLb": 1929,
  "servingSize": "1 cup",
  "calories": 822,
  "fatGrams": 71,
  "proteinGrams":30,
  "rating": 3,
  "text" : "Sample text",
  "url" : "sample url"},
  "walnuts": {
  "waterUsageGalPerLb": 1112,
  "servingSize": "1 cup",
  "calories": 785,
  "fatGrams": 78,
  "proteinGrams":18,
  "rating": 2,
  "text" : "Sample text",
  "url" : "sample url"},
  "Coffee": {
  "waterUsageGalPerLb": 2640,
  "servingSize": "1 cup",
  "calories": 2,
  "fatGrams": 0,
  "proteinGrams":0,
  "rating": 5,
  "text" : "Sample text",
  "url" : "sample url"},
  "Beef": {
  "waterUsageGalPerLb": 1847,
  "servingSize": "3 oz",
  "calories": 213,
  "fatGrams": 13,
  "proteinGrams": 22,
  "rating": 3,
  "text" : "Sample text",
  "url" : "sample url"},
  "Chicken": {
  "waterUsageGalPerLb": 518,
  "servingSize": "100g",
  "calories": 239,
  "fatGrams": 14,

```

Our current “database” of 20 items

7.Summary

We unknowingly consume over $\frac{2}{3}$ of our water usage through the groceries we buy. “Flood” is a mobile platform that sheds light on how our food consumption contributes to our water footprint. Our one-step process provides immediate feedback for every purchase, encouraging users to make greener choices in their grocery shopping. Over time, we give personalized suggestions and help people shape their shopping habits. Flood helps pave the way to a greener, more sustainable future.