CMSC 20370 GP3

Elizabeth Crowdus, Luke Giacalone, Lilly Hackworth, Jillian Ritchey

March 2, 2020

Project Description

Our project focuses on students with dietary restrictions and allergies at the University of Chicago. Currently, it can be difficult or unreliable to figure out how to comply with certain dietary restrictions, making the dining hall stressful to navigate for students with dietary restrictions. Menu information is currently available online, as well as on tablets in the dining halls, but not in easily searchable or accessible formats. Our project explores a better solution in this area, that will allow students at UChicago who are eating in the dining halls to better discover which foods they can and cannot eat at a given meal. Our system will support a wide array of dietary restrictions and any allergies, allowing students with these restrictions to find food they can eat. Non-exhaustive examples of students in this user group includes students following a vegetarian, vegan, or pescatarian diet, or students who cannot eat gluten or are allergic to other specific foods such as peanuts. A technological solution allowing students to view menus through the lens of specific allergies or dietary restrictions would allow them to see what their options for a specific meal are, allowing students to safely eat a wider variety of food without having to be concerned about eating something they should not because they were not sure about the ingredients.

Requirements Summary

- Support for dietary restrictions/allergies: Our system needs to support a broad range of dietary restrictions, as well as any allergens, not just a limited list of the most common allergies. Users should be able to enter their allergens or restrictions and have the menu options appropriately labeled or filtered by ingredient accordingly.
- Accuracy of food suggestions: In order to accommodate particularly severe allergies as well as strict dietary restrictions, and to ensure user trust, foods marked as appropriate for a user must fit in with their specified dietary restrictions, and any uncertainties or potential for cross-contamination be clearly expressed.
- Efficient and simple: Our system must be usable and practical for students on the go to quickly check which foods they can and cannot eat without having to put in significant time while checking or ahead of time to learn a system.
- Up-to-date info: Our system must reflect the most current info on the menu items being offered. Online menus are often out of date or otherwise do not reflect what is actually being served. Our system should take this into account, and accurately inform users as to the ingredients and allergen information of what is in the dining hall.
- Detail: Our system should go beyond just macro level labels of vegan, vegetarian, etc. to take into account specific ingredient allergies, for instance allergies to onion or dairy. It should be clear what restriction or allergy a food is being rejected for containing so that students can make an informed choice about whether they can eat the food anyway (severe vs mild restriction).

- All inclusive labeling: Our system should appropriately label all relevant items being served in the dining hall, not simply main entrees and their accompanying sides. This includes stations such as the salad bar, made to order pasta or stir-fry, and desserts.
- Detect mislabeling: Our system should appropriately handle the case of macro level labels not matching the actual ingredients, for instance a dish marked as vegan that actually contains cheese or eggs.
- Sides in dining hall: Our system should distinguish between the different dishes offered at a given station. Currently sides are grouped with their entree in the Bon Appetit filters, making it harder to find sides that conform with dietary restrictions even if the entree does not.
- Filtering: Our system should make it straightforward to filter by the dietary restriction or allergen, and readily surface the foods that can be eaten without having to manually read through all menu items.
- Best dining hall: Our system should help make it clear which of the dining halls has the best options for a given day so that users can choose accordingly.
- Enjoyable: Our system should provide a satisfying user experience that users wish to come back to and actually enjoy using.
- Accessible: Our system should meet a basic level of accessibility, and in particular should not require a high level of technical literacy.

Prototype Description

Overview

This prototype is essentially a hybrid of the original QR code and web-based filtration designs. Upon opening the mobile app, the user picks either filter or QR code mode. In filter mode, users can compare options across dining halls ahead of time, or view all information for a specific dining hall at once. Based on the user profile and menu ingredients, it displays which stations a user can or cannot eat at for the selected dining hall and meal, with more detail possible for individual stations and dishes. The QR code functionality allows users to scan a QR code, that would be at each station (such as Herbivore, Comfort, Kosher, etc.) in a dining hall, and see which dishes they can or cannot eat at that station based on their dietary profile, and why.

With both views, users can get to the level of an individual dish, seeing which or their allergens or dietary restrictions are met or not, potential cross-contaminants, and even exact ingredient lists. The user profile view allows users to specify the allergens and dietary restrictions they adhere to, which then serve as the lens through which everything else is displayed.

Prototype Details

We created our prototypes using Marvel, an online prototyping tool which simulates a mobile application by linking together screens using hotspots. We designed three versions of our prototypes for three different user personas, and assigned each tester a persona. The three profiles we prototyped were: a gluten-free user, a user with a dairy and peanut allergy, and a vegetarian user. We used a partial Wizard of Oz technique when users were interacting with the prototype. For the majority of the screens, the hotspots and linking worked as they would in a high-fidelity prototype, but in a few places (after "scanning" a QR code, when viewing or editing the user profile, and when initially opening the filtration view), the interviewer would manually make a selection to send the user to the appropriate page.

QR Scanner Prototype

The Welcome screen for the QR scanner prototype gives the user the option to scan a QR code or to view their profile (Figure 1a). Scanning a QR code presents a view of that station's offerings, with each of the dishes sorted into "compatible", "potentially compatible", and "incompatible" (Figure 1c). Icons provide a visual cue for the different sections. Clicking the back button returns the user to the camera to scan another code.

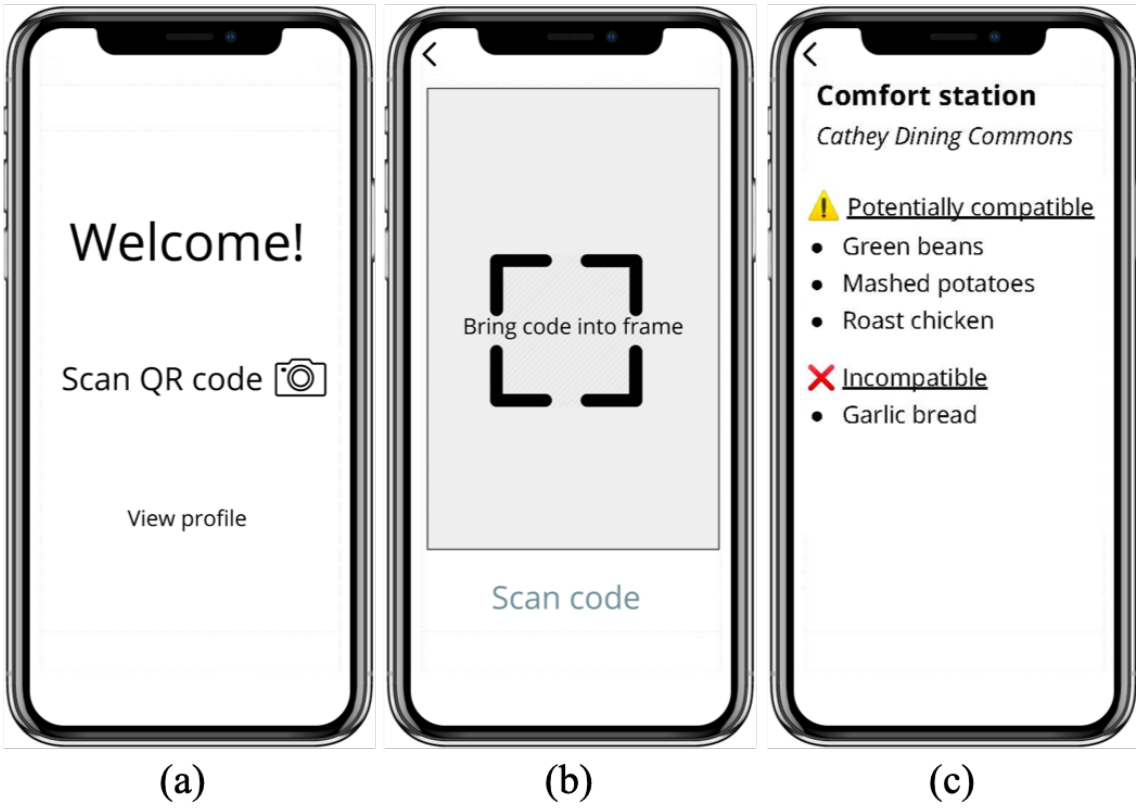

Figure 1: The initial page for the QR Scanner Prototype (a), the camera lets the user scan a QR code (b), the offerings at the "Comfort" station for a gluten-free user (c) sorted by compatibility.

Once the user has landed on the page for a station, they can select a dish to see more information about it's compatibility and ingredients. If a dish is incompatible, the ingredients that make the dish incompatible are listed, and the user can click "See more" to view the full ingredients list (Figure 2). If a dish does not contain incompatible ingredients, but was made at a station where incompatible ingredients are also being served, it is marked as potentially incompatible (Figure 3). The details page for a dish that is compatible with the user's restrictions calls out that it does not contain the user's allergens (Figure 4).

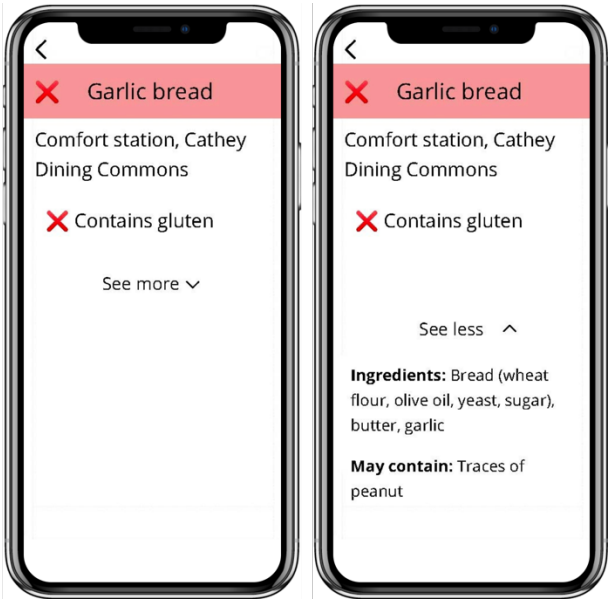

Figure 2: An incompatible dish for a gluten-free user

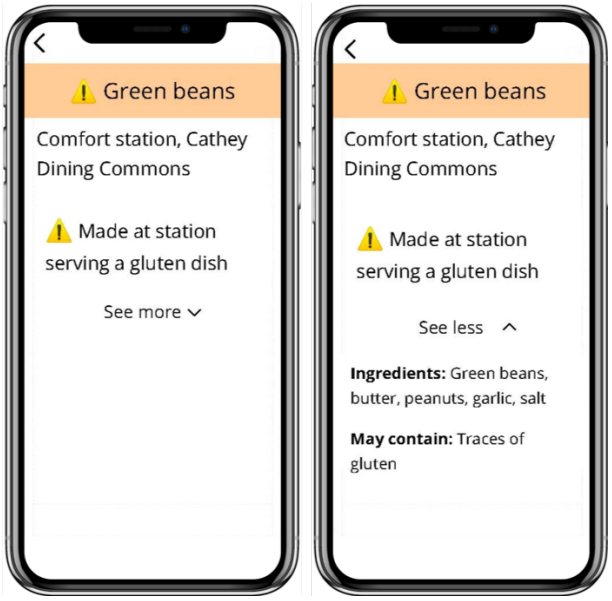

Figure 3: An dish that does not contain gluten, but is being served at a station with other gluten-containing dishes.

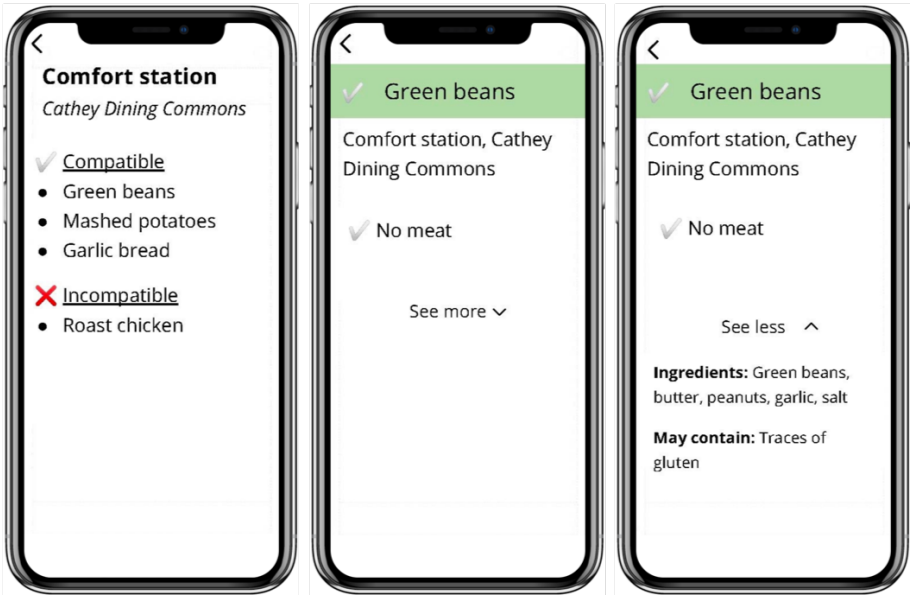

Figure 4: The view after a vegetarian user scans the QR code at the comfort station. The compatible dishes clearly label that they do not contain meat.

Filtration Prototype

The Welcome screen for the Filtration prototype first lets the user select the dining hall they wish to view the menus for, and on which day (Figure 5a). After optionally changing these selections (a default value is pre-filled), they can choose a meal to view. Selecting a meal will take the user to a screen with each of the stations available at that meal, with the station color-coded according to it's compatibility (Figure 5b, 5c). If all dishes at the station are compatible, it is green. If some of the dishes are incompatible or potentially incompatible, it is yellow. If no dishes are compatible, it is colored red. There are also corresponding icons. Selecting a station takes the user to the same view of the station as in the QR prototype (Figure 1c). They can then interact with the dishes the same way, viewing more details to see what makes them compatible/incompatible, and viewing ingredients lists (Figure 2, Figure 3). The back buttons let the user return to select a different station or a different meal, dining hall or date.

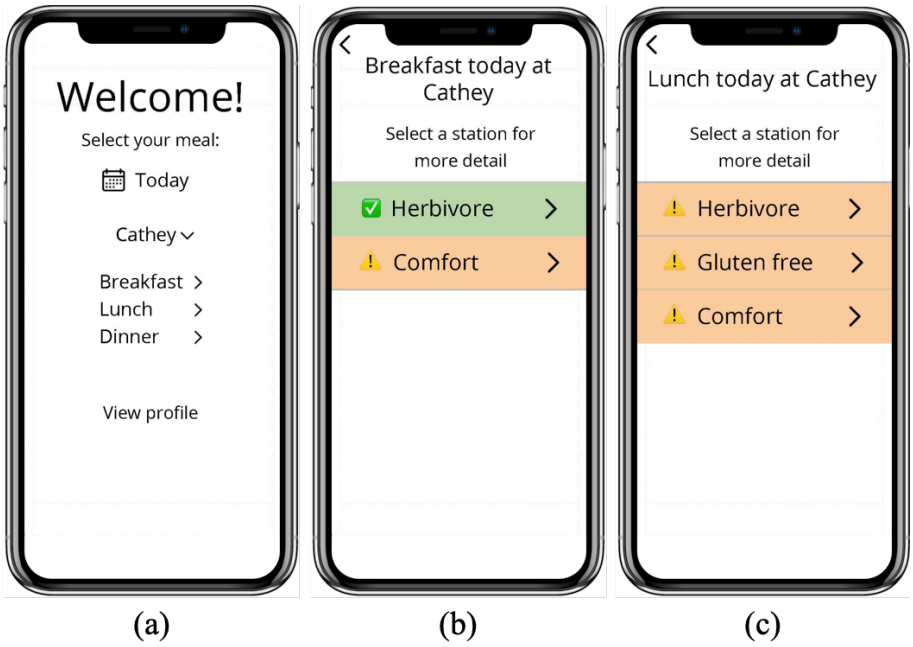

Figure 5: The Welcome screen on the Filtration prototype lets the user choose a dining hall, date, and meal (a). After choosing a meal, the stations serving at that meal are displayed with colors and icons indicating dish compatibility (b, c)

Profile Setup

Upon selecting "View Profile" from either prototypes initial screen, the user is presented with the profile view (Figure 6). If they have not yet created a profile, no dietary restrictions are shown. The "Edit Profile" button allows the user to change their profile and add new restrictions or allergens. The list of dietary restrictions is predefined, but for allergens, if a user has an allergy not included in the list, they can manually add it by name.

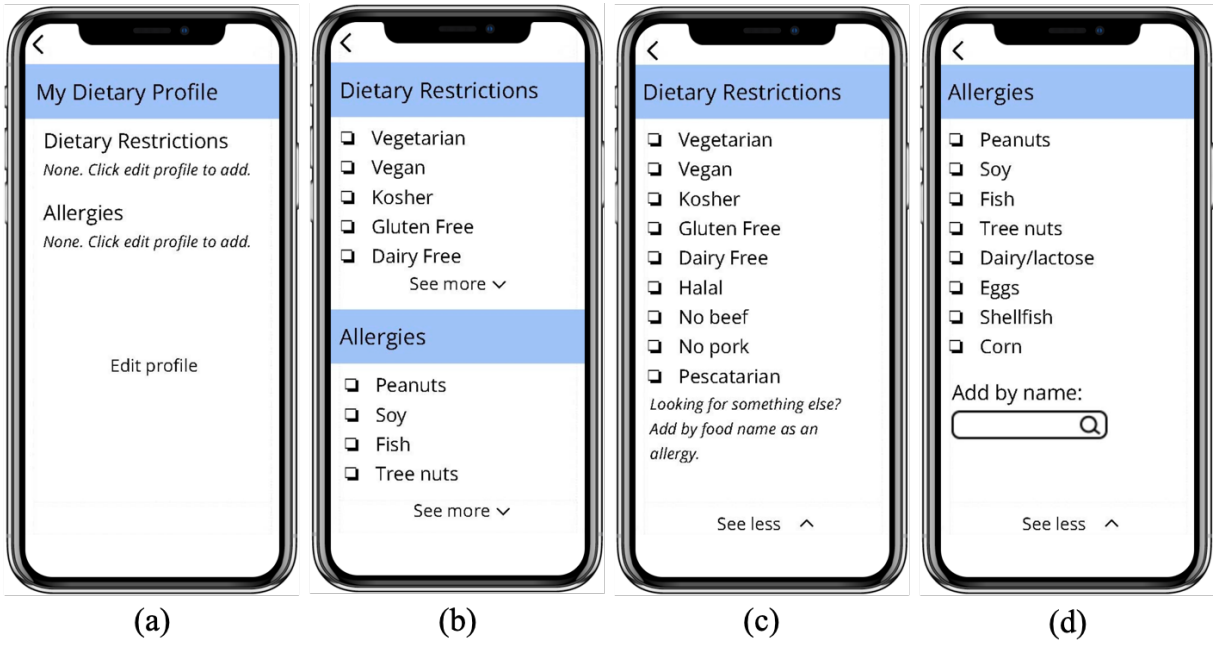

Figure 6: A user without a profile set up (a), the view after selecting Edit Profile (b), the view after selecting "See more" under Dietary Restrictions (c), and the view after selecting "See more" under allergies (d).

If a user has already set up a profile, they will be shown a summary of their restrictions and allergies. Clicking "Edit profile" lets the user make changes to their profile (Figure 7).

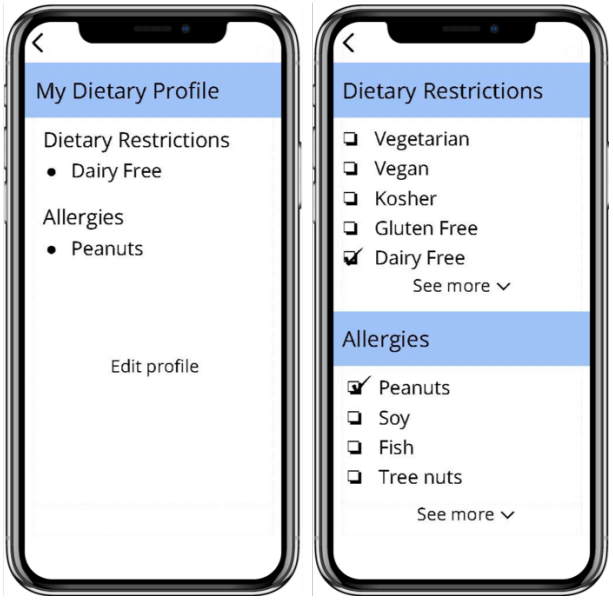

Figure 7: The profile of a user who has eats dairy-free and has a peanut allergy.

The back button on the "My Dietary Profile" page returns the user to the initial screen (either the QR code scanner or Filter page).

Scenario of Use

A user eats dairy-free and has a severe peanut allergy. With our prototype, the first time she uses the app, she can set up her profile by selecting the "Edit Profile" button from either the QR or Filtration page. She selects dairy-free under the "Dietary restrictions" section, and select peanuts under allergens. If she later returns to view her profile, she will see her preferences saved. She could also make changes to her profile, such as if she decides to become vegan.

One morning, before heading to classes for the day, she opens the app to see what options will be available in her usual dining hall, Cathey. She selects the Filtration view, and makes sure "Cathey" and "Today" are selected. Since she is about to go to breakfast, she selects "Breakfast". She sees that the Herbivore station is green, and so will be serving compatible dishes. The Comfort station is yellow, and so has some incompatible dishes. She clicks into the Herbivore station and sees that they are serving vegetables, sausage and rice. These sound like good options, so she heads to the dining hall knowing that she can eat at the Herbivore station without worry. If she didn't like the Herbivore options, she could have viewed the Comfort station and seen which dishes would be an option there. She can now go eat breakfast knowing ahead of time that there will be options available, and which stations they are at. This both gives her piece of mind and makes her visit to the dining hall more efficient.

Later the same day, she goes to a different dining hall, Baker, for lunch with a friend. Since this was last minute, she didn't have time to check the app ahead of time. Upon entering the dining hall, she thinks some of the options at the herbivore and Kosher stations look good, but is not sure if they meet her dietary restrictions. So she pulls out the app, and scans the QR code at the herbivore station. The page shows that the main dish and one of the sides being served are potentially compatible- they do not contain dairy or peanuts, but since the other side dish contains peanuts, there is a higher risk of contamination. The peanut containing side dish is listed as incompatible. She decides to avoid eating at this station. She instead scans the QR code at the Kosher station. All the dishes at this station are labeled as compatible, so she decides to eat here for lunch. Instead of having to read the ingredients list posted at the stations, she was able to get a summary of the dishes on one page and make an informed decision.

Design Rationale

We decided to create this hybrid prototype based on feedback from the design presentations that it seemed useful to both have the higher level information to compare dining halls, as well as more station specific, in-person interaction. We felt there were advantages both to being able to filter options ahead of time - and thus plan out which dining hall to go to, which stations could be skipped entirely, etc., but also to checking stations in real time, at the dining hall. Both of these options were also much more technically possible than the design of scanning food - while interesting and potentially valuable, without the technology to identify even trace ingredients, it did not seem to have much advantage over QR codes.

We conducted user interviews with three of our target users from the initial survey in order to get feedback on both parts of this hybrid design, as well as which they found more helpful. Overwhelmingly, while they did like both, their preference was for the filtration portion. P9 for instance said

> *I love the portability of the filtration system because I can plan out much more easily and it allows more flexibility...QR code would be a little redundant*

Similarly, from P13,

> *I don't see the advantage of the QR code. The filtration system gives you all the advantages of the QR system without the limitations of needing the QR code.*

P3 summed up the downsides of using the QR code design very simply:

> *The QR code is not accessible all the time.*

Based off of this feedback, we focused on the filtration functionality of our prototype for the in-class evaluation, and will be assessing that variant for further improvement as we iterate and build a higher fidelity prototype.

The major advantage to this design is the ability for users to check food options from anywhere, easily accessible on the smartphone they likely already carry. This allows a user to choose which dining hall is best for them for a given meal, avoiding wasting time and meal swipes only to find that they cannot eat anything being offered, as well as allowing them to easily plan out their day around where they will be eating. This also prevents users from having to walk around to each of the stations with their phone to scan the QR code and figure out what they can eat, a major inconvenience especially in a crowded dining hall with long lines.

This design also fulfills the major requirements of allowing users to personalize their dietary restrictions, and view the menus through the lens of those restrictions without having to tediously check through all foods being offered and reading all those ingredient lists. Information is presented first and foremost at the level of the dining hall, however users can then click to view more information about specific stations, and from there check individual dishes as well. This provides a level of detail up to the exact ingredient lists and potential cross-contaminants, while allowing those who simply want to know what does or does not meet their dietary restrictions quickly get that level of information without hunting through the details. The verdicts on dishes and stations are based off the user's personalized profile as well as the ingredients of a given dish and of other allergens at the same station, circumventing the untrusted and often inaccurate Bon Appétit icons.

However, there are still drawbacks, in particular the reliance on the ingredient lists. If these lists are inaccurate, or there are last minute changes, our prototype is unable to flag or handle this. Unfortunately, there is little technological way to get around this - if Bon Appétit makes false statements about the dishes at an ingredient level, no technology currently on the market can detect that.

Another weakness in this design is its reliance on users to input their preferences beforehand. This could be a dangerous design flaw if a user with a severe allergy incorrectly inputed their dietary restrictions and thought that they could eat an dietarily incompatible food. However, each dish view is explicit with stating all the allergens and whether they are or are not present, ideally helping to identify such discrepancies.

Results of In Class Evaluation and Early-Stage User Evaluations

Data Coding

	Target Users			Classmates			
	P3	P9	P13	P15	P16	P17	P18
info fatigue concerns					x	x	x
wanted ingredients list	x	x	x	x			
signifiers confusion	x	x	x	x	x	x	x
wants recommendation system				x		x	
liked the UI's simplicity					x		x
trust concerns	x	x	x				
would use system	x	x	x	n/a	n/a	n/a	n/a

Figure 8: Qualitative results from initial user research interviews and in-class feedback interview session.

Trust, feedback, and signifiers

We found that all of our target user interviewees wanted some sort of feedback from the system in order to help build trust and reinforce the notion that the system is personalized towards them. Users are used to looking for icons or other signifiers that reinforce trust that certain foods don't violate their restrictions, such as P3, who is allergic to eggs, milk, and nuts, and therefore looks for vegan and nut-free dishes even though they do consume meat but they want reassurance that a dish is egg- and milk-free.

These trust concerns necessitate a system with enhanced feedback and signifiers which communicate to the user that their inputted dietary restrictions are being considered. This design change would also help guard against the dangerous case in which a user forgets to indicate an allergen, notifying them to its absence. The signifiers that we had designed included stop, caution, and go icons and labels that deemed food compatible or incompatible. Based on this feedback, we are going to continue to use the icons for visual clarity, but provide a definition next to each. For example, if a user has a peanut allergy, a list of dishes which comply with the user's designated restrictions would include the title "dishes without peanuts," and a list of dishes that don't comply would be titled "dishes with peanuts." This change circumvents clarity confusion with both the icons and our ambiguous previous language choice of "compatible" and "incompatible" foods.

Target users liked the ability to personalize and filter by dietary concerns. Users with severe allergies also appreciated the ability to flag dishes that were made at stations containing foods that violate their restrictions. Our redesign will continue to provide this precise level of personalization, but enhance the features that signify to users that their restrictions are being considered.

Information fatigue

Most of our in-class users expressed concerns about information fatigue in the app, while all of our target users expressed a desire to see dishes' ingredient lists. We decided to defer to the expertise of our target users and, in the next iteration of our design, to allow users to click on a dish to see a full ingredients list with any ingredients that violate their restrictions highlighted. Also, in the list view of dishes, sorted by

whether or not they violate restrictions, we will say underneath or next to the name of an offending dish "contains [restricted item]". In doing so, we hope to strike a balance of providing not too much information, but still providing target users with information that is most important to the needs, trust, and habits of our target users.

Changes to Design and Evaluation Techniques

For our next iteration, we will continue to conduct user interviews with our new prototype and to data code the results of the interviews. Since we will have a higher fidelity prototype, we will have more possibilities for user interaction and personalization for individuals to try out and provide feedback on.

In terms of design, based on the above insights we will increase the degree of feedback and personalization in our next design, replacing the vague signifiers of "compatible" and "incompatible" with titles personalized to the restrictions of each user. Due to feedback that the icons were not always clear, particularly the caution one, we can augment those with more clear information as to whether some dishes are compatible and some not, vs all at risk of cross-contamination, etc. Each page of the app should make it clear that the user's dietary profile has been saved correctly and is being used to filter dishes in order to build trust in the system, beyond just the dish specific and ingredient pages. Furthermore, based on feedback from target users, we will list next to each dish the ingredient in it which does not comply with their restrictions. For example, a vegan user, instead of seeing "quiche" listed under "incompatible," they will see "quiche - contains eggs and cheese" under "foods which contain animal products."

Pros and Cons of Prototyping Tools

We used Marvel, which allowed us to quickly produce a series of panels on a web browser which participants could navigate through. The prototype was hard-coded, so participants were unable to personalize their dietary restrictions, just to visualize the way that they might do that. The prototype allowed us to display both a QR-based prototype and a web-based filtration system, as described in 2 of our GP2 designs. The fact that Marvel is web-based was also helpful because it mimicked most closely what our final prototype will be: a web-based mobile interface.