**TigerEats Product Eval**

Ishan Sinha, Paulo Frazao, Gabriel Birman, Jamison Mercurio

**Section 1:** “How well does the product work?”

Overall, we are very happy with our final product. We made our initial designs of the product with Princeton’s sports nutritionist and implemented them almost precisely as we designed them. With some functions, such as updating a user’s nutrition goals, we were able to add more clarity and functionality than originally designed. In this particular example, we gave the user two different ways of updating a user’s nutrition goals and filled in more gaps in this process than originally intended.

We find almost all aspects of the product to be robust. Many of our features rely on queries from our database. Though these queries are complex and required programming with care, drawing information from a database generally does not expose the system to harm. Furthermore, we successfully integrated CAS to protect any URL associated with the app, whether it is running on localhost:5000 or <https://tigereats.herokuapp.com/>. Not only did we integrate CAS, but in our MongoDB database we also created a smaller subset of Princeton community members that may use the product. So, after a user authenticates with CAS, we cross-check if they are one of the select Princeton community members who are allowed access, and only then allow access to the application; otherwise, they are sent to an error page.

The one aspect of the product we think may be fragile is the ability for the nutritionist to update user nutrition goals. There are other POST requests, but this is the only one where the user can enter in plain text as input to our database. We do several validations to make sure values are valid and cannot harm the database, but if there is any functionality of our product that may be insecure, it would be this functionality.

The known bugs are:

* When the user zooms in on the first day of the entire time series, the brush gets “stuck” while attempting to zoom out using scroll -- this is a bug with how the React Time Series Library works
* The Progress Page tracker line and brush range pan does not work for mobile or ipad users of the website as swipe functionality is not currently enabled
* If a non-authorized Princeton community member (i.e. someone who has a netid but who’s netid is not valid) logs into the application, the dashboard renders for a split second before redirecting to the error page
* Sometimes the user needs to click“Remove from Watchlist” twice on an individual Student Profile page for the change to register. This seems to happen only when clicked in this state after a refresh; the next click always requires 2 clicks instead.

The other areas as follows. We came up with these through our own analysis as well as from results of user testing (which we elaborate on in the next section):

* When filtering by Male or Female, the Dashboard sometimes slightly changes the left alignment of the students’ names, so the data shifts over slightly
* Dashboard filtering may be more intuitive if users had to select what they do want as opposed to de-selecting what they do not want -- this would require more user research though
* Dashboard could use a “Clear all filters” option
* The Student Profile pages for each student could use more line dividers to distinguish the different sections.
* In the Student Profile page, it may be helpful to allow the user to query a specific date for which to retrieve a summary of the user’s macronutrients and meal choices for that day; we attempted this, but could not come up with a design that was smooth both in implementation and flow
* In the Student Profile page, we could collapse the meal logs for different days into buttons with dates, and only display the meals and their information when the user clicks on the button for that day.
* We could allow users to select a start date and end date for the student progress graph, as opposed to having them slide a bar to expand the date range
* In general, we could clean up the responsiveness to screen size changing. We do a pretty good job of it, but it could be a bit more standard.
* The Progress Page does not have clear enough instructions, an issue that many users pointed out during user testing. Though we did add a modal to explain the time range change per user feedback, it would be helpful to clarify that the label axis statistics represent an average over the entire displayed time range in days, whereas the value axis represents the average for the given rollup.
* In addition to the above, a Date Picker for the time series would make time range adjustments simpler

Some other features we plan to add when we eventually integrate this with the mobile application’s database:

* The ability for the nutritionist to add a note on student’s meal, and then have that note show up in the user’s phone for that meal
* A messaging platform for the nutritionist to talk with the student user

**Section 2:** “How well does the product meet the needs of its users?” -- User Testing

We tested our product with 22 users. We gave them a very specific sequence of instructions to follow (essentially a slightly simplified version of the User’s Guide) and noted their experience with each task. While many of our user testers were students, we did get to test this website with a few nutritionists and sports coaches who will be the actual users of the product upon launch; we highlight their feedback when it diverges from student testers’ opinions. Now, we list each task that we instructed testers to perform, as well as the general responses from users about the difficulty of these tasks.

1. Visit https://tigereats.herokuapp.com/ on any computer (Did this load as intended? If not, how many tries did it take?)

In the beginning of our user tests, we still had Heroku memory issues, so a small minority of users had to load the page multiple times. After we resolved those issues (more than half of the user tests came after we resolved them), each user was able to load the page quickly on the first try.

1. Log into the application, using "isinha@princeton.edu" for the email and "password" for the password (Did the dashboard load? Are there entries in it?)

This has been replaced with “Login with CAS” but, regardless, all users successfully loaded the dashboard.

1. Dashboard: Only display students on the boy’s soccer team

All users were able to complete this task. Some users suggested that we select filters we want to apply as opposed to de-selecting filters we do not want to apply. This is something for which we would have to do more research to decide a final UX.

The nutritionists and sports coaches who are the final users of this application did not comment on the filtering scheme.

1. Dashboard: Display all of the students

All users were able to complete this task. Some users suggested that we could include a “Clear all filters” button so they would not need to de-select each option.

The nutritionists and sports coaches who are the final users of this application again did not comment on the filtering scheme.

1. Dashboard: Only display male students in the class of 2022 on the soccer team

All users were able to complete the task. The only qualms were the same qualms about filtering as before.

The nutritionists and sports coaches who are the final users of this application again did not comment on the filtering scheme.

1. Dashboard: Send Paulo Frazao an email! (Does this pull up your default email application?)

It triggered the correct functionality for all users, but not all users have the Mail app in their computer configured. In the future, we would create a messaging platform within the portal.

1. Dashboard: Check your watchlist, and then display all of the users again

All users accomplished this task with ease.

The nutritionists we tested were big fans of this feature. They tend to have some athletes with very specific goals (ex. Wrestlers before a meet).

1. Dashboard: Navigate to Gabriel Birman's student profile

All users accomplished this with ease.

1. Student Profile: What is Gabriel's goal weight? (Please state your answer here. Was finding this information intuitive?)

All users found this information with ease.

1. Student Profile: What did Gabriel have for dinner 2 nights ago, if anything? (Please state your answer here and also state which date it is today. Was finding this information intuitive?)

Most users were able to find this information, and most said they found it with ease. However, some users went to the Student Progress page as opposed to scrolling down in the Student Profile page for Gabriel. We could have a quick tutorial on this for the user; checking out the progress page is a one-time mistake that a tutorial remediates

1. Student Profile: Note any aspects of the page that were unintuitive or that you would consider tweaking. (Were there any elements on this page that did not work as intended?)

Users had three main points of feedback. First, they thought we could explicitly divide different sections of the page more clearly. As in, we could have more line dividers between different meals and bolder text to highlight different pieces of information about the user we are displaying. Second, they thought we should collapse the daily summaries of meals until the user clicked on that day; only then would we show all details. Third, one of our nutritionist testers specifically mentioned it could be helpful to have a calendar or other method to allow the nutritionist to select which date for which she would like to see a student’s meal history.

1. Student Profile + Change Goals: Navigate to the Change Goals page

Everyone got this with ease.

1. Change Goals: Change Gabriel Birman’s nutrition goals by inputting specific protein, fat, and carb goals. Navigate back to the dashboard. Did the values change?

All users were able to accomplish this task with ease.

1. Dashboard: Navigate back to the Change Goals page

All users were able to accomplish this task.

1. Change Goals: Change Gabriel Birman's nutrition goals by inputting a daily caloric goal and macronutrient breakdown percentages. Navigate back to the dashboard. Did the values change?

Unfortunately, we had a bug in this aspect of the application when about half of our user tests were conducted. After we fixed this bug, all users were successful.

The nutritionists and coaches who tested this feature really liked how there were two ways to input nutrient goals for users: sometimes they have specific macronutrient values in mind, whereas sometimes they have macronutrient percentage breakdowns in mind.

1. Dashboard: Navigate to Jamie Mercurio's student profile, and then view his progress on all his nutrients (Was this flow intuitive?)

All users accomplished the steps necessary for this task with ease. One of the nutritionists, however, said a lot was going on in the graphs, so it would be helpful to highlight or enlarge the text for the user’s goal values.

1. Progress: Check Jamie's progress for just carbs and protein

All users accomplished this task with ease.

1. Progress: Check Jamie's progress for fat on a monthly basis (Does the layout of this screen make sense? Do all of the values contained in it make sense?)

All users accomplished this task with ease.

1. Progress: How many calories did Jamie consume on 1/1/19?

All users were able to accomplish this task, although they commented that the font is a bit small.

1. Progress: Restrict the progress view to Jamie’s consumption only from 1/1/19 onwards

Most users were able to figure this out. However, they commented it was not intuitive and would prefer being able to set the date range by typing in dates as opposed to dragging a bar. Those who could not figure it out mentioned that they would know how to the second time around though (after we showed them how). One of the nutritionists mentioned that she was unclear how to change the date range on her own, so she would have needed a tutorial.

1. Progress: Log out of the application

All users accomplished this task.

1. Please feel free to provide any other feedback that was not listed above! Any and all notes help. Thanks again for helping to test TigerEats

Generally, people were very pleased with the application! There were many feedback points touched upon above, but the main points are: i) clarify the student progress graphs; ii) fix minor styling to bring out key pieces of information on the Student Profile and Student Progress pages; iii) have a brief tutorial of the application on first use.

**Section 2:** “How well does the product meet the needs of its users?” -- Nielsen heuristic evaluation

1. **Visibility of system status**:  
   The system should always keep users informed about what is going on, through appropriate feedback within reasonable time.

Generally, the system does an OK job of presenting appropriate feedback within a reasonable time frame.

Some examples of useful feedback include: updating the students matching the user-specified filters in real-time; displaying errors with data validation in the change goals page as an alert; indicating whether a user is in the watchlist or not within their profile by switching the button text between “Add to watchlist” and “Remove from watch list”; using a tracker line to detect mouse hovers over the time series, and using this to inform the date and the nutrient values in real-time; real-time adjustments to the time series display during zoom/panning of the brush; changing the bar sizes immediately to reflect the rollup level; the students dashboard refreshes asynchronously with every key press during student searches.

Some examples where improvement is necessary: changing the students goals in the change goals page redirects the user to the dashboard without showing the user how their change has been implemented; explaining that the dashed white lines in the bar charts represent the users goals; providing info boxes to explain to the users how to use the zoom features, as well as the function of the radio buttons/check buttons of the progress page; showing how the daily/weekly/monthly radio buttons affect the date format in the progress page.

1. **Match between system and the real world**:  
   The system should speak the user's language, with words, phrases and concepts familiar to the user, rather than system-oriented terms. Follow real-world conventions, making information appear in a natural and logical order.

The web app does a good job of using comprehensible terminology. The most esoteric terms used are dashboard, watchlist, goal, and macronutrients -- all except the last one being clear to the layman, and the last one being clear to nutritionists who comprise the user-base of the app. One example where real-world conventions are eschewed is in the “weekly” rollup of the progress page, where a week is currently defined from Thursday to Wednesday due to technical limitations, instead of using the widely recognized international standard ISO 8601 which defines a week starting from Monday.

1. **User control and freedom**:  
   Users often choose system functions by mistake and will need a clearly marked "emergency exit" to leave the unwanted state without having to go through an extended dialogue. Support undo and redo.

The navigation header on every page contains a link to the dashboard. Generally, actions such as radio button or check button clicks are easily redoable. Moving between pages does not support simple undos if the default state has been altered. In the case of the progress page for instance, if the user accidentally goes back to the student profile and then wants to return to the progress page, the component state reverts to the default as opposed to the previously existing state. If the user changes goals but then wants to undo/redo that choice, he or she would have to manually change the data back as the system has already moved the data into the database.

1. **Consistency and standards**:  
   Users should not have to wonder whether different words, situations, or actions mean the same thing. Follow platform conventions.

Intra-system consistency and standards are highly dependable. No deficiencies here to the best of the team’s knowledge.

1. **Error prevention**:  
   Even better than good error messages is a careful design which prevents a problem from occurring in the first place. Either eliminate error-prone conditions or check for them and present users with a confirmation option before they commit to the action.

Users are prevented from entering invalid data in the change goals section, and are alerted when they attempt to do so. In the progress page, if there is only one nutrient channel being displayed currently, its checkbox becomes disabled, preventing the user from accessing a “no charts displayed” state. Users are also prevented from zooming beyond a minimum resolution of one day, or moving the timeseries outside of the student’s first and last day of input data. The rest of the actions do not involve user input so there is no way to invoke a user-data error as a result.

1. **Recognition rather than recall**:  
   Minimize the user's memory load by making objects, actions, and options visible. The user should not have to remember information from one part of the dialogue to another. Instructions for use of the system should be visible or easily retrievable whenever appropriate.

Generally, data is made available to the user for any given component. For example, in the change goals page, we both display the current goals and present the user with text fields with which they can update goals. If the user wants to see which filters are on, they have the option of keep the filters menu open. An example where this is not the case is in the progress page – although the progress page for a given student is only accessible from that student’s profile page, there is no indication on the progress page of who that student is.

1. **Flexibility and efficiency of use**:  
   Accelerators—unseen by the novice user—may often speed up the interaction for the expert user such that the system can cater to both inexperienced and experienced users. Allow users to tailor frequent actions.

These kinds of enhancements do not exist in our app beyond having a user-friendly design. The TigerEats team is hoping to add pressing the return key to submit changes in the change goals pages as a future feature.

1. **Aesthetic and minimalist design**:  
   Dialogues should not contain information which is irrelevant or rarely needed. Every extra unit of information in a dialogue competes with the relevant units of information and diminishes their relative visibility.

The design is aesthetic and minimalist as it incorporates stylistic elements based on Google’s sleep Material design paradigm.

1. **Help users recognize, diagnose, and recover from errors**:  
   Error messages should be expressed in plain language (no codes), precisely indicate the problem, and constructively suggest a solution.

The only time the user encounters an error message is when they enter incorrect data in the change goals section, with intuitive and helpful error messages describing the issue.

1. **Help and documentation**:  
   Even though it is better if the system can be used without documentation, it may be necessary to provide help and documentation. Any such information should be easy to search, focused on the user's task, list concrete steps to be carried out, and not be too large.

Help and documentation is not available on the website itself, though it is available in external documents such as the Programmer’s Guide and the User’s Guide.

**Section 2:** “How well does the product meet the needs of its users?” -- Cognitive Walkthrough

We implicitly did this when developing the sequence of tasks we compiled for our user tests. The user tests were an unbiased cognitive walkthrough.