

# GrabAGrad

## Product Evaluation

Last Updated May 2, 2022

# Testing

## Internal Testing

To test our code internally, we utilized print statements throughout the code at various checkpoints to ensure that the data being stored in variables and sent throughout the application had values that we expected them to have.

Additionally, we used print statements to ensure that the proper statements in the code were being executed based on specific conditions present. For instance, in the script.js file, console.log() statements were added within certain conditional statements in the windowResize() function to verify that the correct conditionals were being executed depending on the size of the window.

Character limits and type requirements were placed on specific inputs to ensure proper data was inserted into our database and to also help defend against SQL injection attacks. For example, for the “Number of Years Worked” input, only numbers are enabled to be input. We added Python functions in effort to modify a users’ searches if we thought they would create errors for our program, specifically regarding interactions with our database. This is described more thoroughly in **Boundary Testing**.

To test database connection errors, we purposefully modified the name of the database to cause issues in connectivity to properly test whether the correct errors were being thrown on the frontend of the application.

## White-Box External Testing

**Boundary Testing** - In order to perform boundary testing on our application we tested first on the “form page” inputting responses with the largest number of characters possible in all the inputs. This enabled us to determine whether the card mockup could handle extremely long inputs while retaining proper styling and whether the cards on the search page and details popups could properly display long inputs. This ultimately led us to decide to truncate long responses using “...” to maintain organization on the site.

Additionally, we tested the app's response to the lack of any inputs on the "form page" (which caused an error, as certain inputs were required) and on the "search page" (where all graduates in the database were displayed. By the same token, we checked to make sure that the application functioned properly if all the "filter by" inputs on the search page were filled in and all the inputs on the "form" page were filled in.

An additional boundary test we utilized involved attempting to resubmit the graduate student form multiple times to make sure that the same profile wasn't being rewritten into the database more than once.

We also extensively applied boundary tests to our Search and Create a Graduate / Update a Graduate pages for various inputs. One example of this was that initially, our program was unable to determine if a user input a '%' in one of the Filter By options on the Search page, and without boundary testing here, we wouldn't have caught and handled the resulting undesirable performance. On the Create a Graduate / Update a Graduate pages, we tried inputting in html code to see if our application would allow users to put malicious links or html output within our site. While this information is added to our database (which is consistent with our expected functionality), it would not edit the html pages, which is our desired result.

**Statement Testing** - To make sure that all statements were executed at least once, we spent time intentionally testing every piece of functionality offered by the application, to make sure that all the endpoints were being tested and all the functionality within each of those endpoints were properly being executed.

**Path Testing** - Using a procedure similar to that utilized for statement testing, we tested every piece of functionality offered by the application, which in the process tested all paths available for users and administrators.

## Black-Box External Testing

**Use Case Testing** - To test specific use-cases, we took a look back at our project overview document and identified all the possible use-cases that we had planned for at the start, and throughout, the development process. Then, we acted as if we were real users and tested to ensure that each use case was

satisfied. For instance, one important use case involved allowing a graduate student to modify their profile after they had already created it. Therefore, we created a profile and then tested that we were able to update the profile immediately after creating it as well as after browsing the other functionalities of the application, then coming back to the profile page.

**Stress Testing** - In order to get the maximum possible amount of data running through our application, we enlisted the help of graduates we knew to recruit other graduate students. Those who were interested in the GrabAGrad idea submitted their information to us, which enabled us to test our system with random and truthful data.

We also all used the app concurrently, and opened many tabs of the app to ensure that the application could run with more than one user. This was an attempt to strain the database from heavier loads, but we found no noticeable results when we did this.

Another stress test that we did was type as fast as we could. Because our system uses AJAX, typing quickly means that many calls would be made to our database. The app showed no noticeable decrease in performance.

## Known Bugs

No known bugs as of the most recent update to this document.

## Robust Aspects of the Product

Our application makes use of prepared statements (parameterized queries) in SQLAlchemy to protect against SQL injection attacks and sets character limits on inputs to provide security against other potential attacks. Additionally our use of CAS Authentication ensures that only specific users are granted access to the application and only distinguished users have admin privileges. There is also no way to add administrator privileges through our app, it has to be done manually with a direct connection to the database, via the terminal or on pgAdmin, for instance. This protection along with CAS authentication ensures that no user will mistakenly gain access to administrator privileges.

Our application properly responds to changes in the screen size through the use of JavaScript, enabling our application to change its styling in multiple ways to make the user experience easier and more appealing.

Our application provides error handling in regards to database and server errors.

## Fragile Aspects of the Product

Our application struggles to load substantial amounts of data to the user interface quickly. For example, as the number of graduate students in our database increases, the amount of time it takes for the search page to load grows.

Our application is responsive to changes to mobile screen sizes but faces difficulty in optimizing the user experience for mobile users, as certain functionality, such as the explore card for example are difficult to properly navigate on small screen sizes.

## Evaluation By Users

### Does our Application Meet the Needs of Users:

This is discussed more in the **Room For Improvement** subheader below, but to some extent, our testers didn't understand the point of our application. We think that we can address this by making a prominent **About** button on the application's Home page, to encourage users to read more about the purpose of the application. We also think that to some degree, testers not understanding what problem GrabAGrad addresses is a difficulty that would resolve itself as the application gains traction, especially because real users would come to the app with a specific goal in mind. Testers, on the other hand, are introduced to the app but have no real idea what it's for, which doesn't reflect the baseline knowledge about GrabAGrad that we would expect from anyone who made

their way to the page in the first place. Once testers had read the **About** page, specifically, the first sentence at the top of the page, which describes how GrabAGrad is a platform for connecting graduate and undergraduate students, we found that testers understood why someone would want to use the application.

With this out of the way, we found through our user evaluations that our app was very successful in enabling graduate users to create profiles with relevant, yet brief amounts of background information on their lives, and we found that undergraduate users had no trouble navigating our app to learn more about graduate students. The **Explore** page of our application was especially helpful for undergraduate users who were unsure what to search for beyond their own department. They thought it was very interesting to read about graduates' research focuses, and found this was a good place to draw connections to their own interests and potentially strike up conversation if they were to reach out to the graduate students to meet.

## Summary of Positive Feedback

- ❖ Users appreciated the clean and minimalist design of the application
- ❖ Users felt that the application was easy to navigate using the dynamic navigation bar
- ❖ Users felt that the application was very intuitive to use
- ❖ Users liked the simplicity of the general idea of GrabAGrad

## Room For Improvement

- ❖ Users felt they couldn't get a grasp of the motivation behind the application
  - To mediate this, we added a button linking to the "About" page on the landing page to help first time users easily navigate to a page where they could view more information about the app and how to use it.
- ❖ Users commented that the mobile application experience could be improved through increasing font size

- Following the gathering of user input, we modified the font sizes especially on mobile to enable users to better view the contents of the application
- ❖ Users frequently tried to search academic departments by typing in the acronym for that department (for example, “COS” instead of “Computer Science”)
  - To fix this issue, we implemented slight modifications to our search functionality to map acronym inputs to their respective fully spelled out input names.
- ❖ Undergraduate users didn’t want to have to exit the app in order to send a message to a graduate student they were interested in connecting with.
  - This was one of our stretch goals, and we would have loved to incorporate a feature like this into our application. However, we were unable to achieve this within the timeline of the class and the other deliverables.

**\*\*\*See Appendix for specific feedback notes**

## Evaluation By Experts

### Nielsen's 10 Factors

### Visibility of System Status

Our system always keeps users informed of their status on the app by quickly altering visual outputs. For instance, as soon as a user clicks the “Favorite” button on a graduate’s popup or the Explore page, that button will toggle to “Unfavorite” as soon as our app has registered the favoriting action. Furthermore, if a user clicks “Unfavorite” on a graduate’s card while on the Favorites page, that card will be removed from the Favorites grid as soon as the user exits from their popup. The same holds true for the Admin Page if an administrator deletes a graduate profile. Furthermore, any search bar input will automatically update the outputted graduate list without the user having to also click a “Search” button. This maximizes efficiency and simplicity for the user. Some other examples of our app keeping users constantly informed of the app’s status are below:

Message in response to the user inputting search criteria for a graduate student that doesn't exist.

The screenshot shows a search interface with the following elements:

- Header: Home, Search, Explore, Favorites, Admin Page, GrabAGrad
- Section: Filter By (Doesn't Exist, Department, Industry, Years Worked, Undergraduate Institution, Master's Institution)
- Section: Sort By (First Name: A-Z dropdown)
- Text: No Grad Students Match The Search Criteria

Message in response to the database not being available.

The screenshot shows a search interface with the following elements:

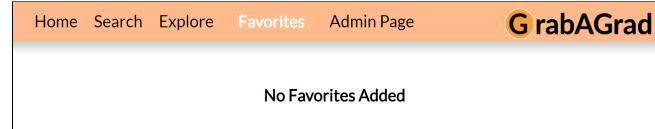
- Header: Home, Search, Explore, Favorites, Admin Page, GrabAGrad
- Section: Filter By (Error, Department, Industry, Years Worked, Undergraduate Institution, Master's Institution)
- Section: Sort By (First Name: A-Z dropdown)
- Text: Server side error occurred. If this problem persists, please contact hknoll@princeton.edu.

Message in response to the user creating or updating their profile.  
Note: The same message appears for either updating or creating a profile, but the data received from the users is displayed either way, so the user can verify that the correct information has either been updated or inputted.

The screenshot shows a confirmation message with the following elements:

- Header: Search, Update My Profile, GrabAGrad
- Text: Thanks for your submission
- Section: What we got from you:
  - First Name Test
  - Last Name Test
  - Academic Department
  - Years of Work Experience

This message serves to inform the reader why no graduate students are present when they click the “Favorites” page without previously favoriting any graduates.



## Match Between System and the Real World

Our system uses common phrases to ensure that the user consistently is able to have a full understanding of the full functionality of each page. For instance, on the search page, the terms “Filter By” and “Sort By” are used to split up the inputs available to the user. Using common terms to describe each set of inputs makes it clear to the user that the first set of six inputs are used to filter the list of graduates by common attributes, such as “Name,” “Department,” etc. Further, the names of placeholders were specifically chosen to provide succinct descriptions of the target input. Additionally, on the search page, the phrase “Learn More” on each of the graduate student cards serves to convey in common language that clicking on the button will enable the user to gain more insight into the specific graduate student.

The names of the pages at the top of the application were also chosen specifically to give the user a clear understanding of the role each page plays. For instance, “Search,” “Explore,” and “Favorites” all convey very specific functionality in common language.

The first screen that a user sees serves to use common language to direct the user to the proper side of the application.

The layout of the search page, with a grid of graduate student cards containing a photo and first name, serves to mimic the real world practice of meeting people. The first impression a user has of a graduate student is the face on the front of their card. They can look below the face to see that graduate’s first name and academic department. Then, if that limited impression appeals to the

user, they can invest more time learning more about them (just as in conversation) by interacting with the card and clicking “Learn More”. Thus, the card layout serves to make it easier for users to browse the catalog of graduate students to find specific graduates that interest them.

## User Control and Freedom

The consistent presence of the navigation bar at the top of the application always provides the user with a way to “escape” the page they are currently on. Additionally, the application logo directly brings users to the About page, where they can seek guidance on how to properly navigate GrabAGrad.

In the case that a user makes a mistake in creating their profile, the “Update Profile” option is always available after creating the initial profile, where the user can modify information that they previously may have made an error in inputting.

## Consistency and Standards

Throughout the application, each term utilized, such as “Search,” “Explore,” “Favorite,” etc. were only utilized to describe one piece of functionality / page on the site. This was intentionally done to ensure that the user would never have any situation where one word, situation, or action meant something different in two different scenarios.

## Error Prevention

The most pervasive source of errors in our application originates from users inputting incorrectly formatted or unsuitable information into input fields. For example, one source of error is a user inputting text that doesn’t describe an academic department into the “Academic Department” field. To prevent such an error in the first place we implemented the following:

- ❖ **Provided Drop Down Menus with Suggested Input** - Providing drop down menus for autofilling specific input fields containing a variety of options for the users to select from discourages users from inputting data

that our application is unable to properly handle, while simultaneously improving the user experience by reducing the number of decisions a user is required to make

**Filter By:**

Name

Com

Applied and Computational Math  
 Comparative Literature  
 Computer Science  
 Electrical and Computer Engineering

Master's Institution

## Update Your I

First Name \*

Last Name \*

Academic Department

Years of Work Experience

Undergraduate Institution  Prince

Undergraduate Major

Masters Institution

Masters Degree

Prince Mohammad Bin Fahd University  
 Prince Sultan College for Tourism and Hotel Sciences  
 Prince Sultan University  
 Prince William Sound College

- ❖ **Required Inputs** - Specific inputs that the user enters when they create a profile are required in order to ensure that any created profile contains several minimal qualities so that the profile is useful to undergraduates

Preferred Contact Email  \* test@princeton.edu

Preferred Phone Number

Profile Photo



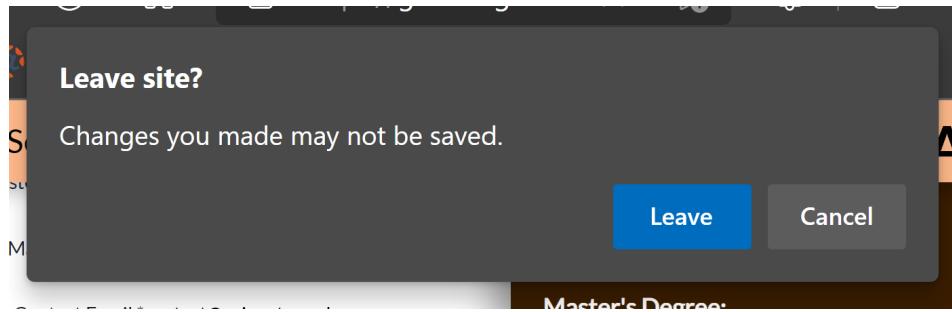
Current Research Focus (120 chars)

Industry Experience (120 chars)

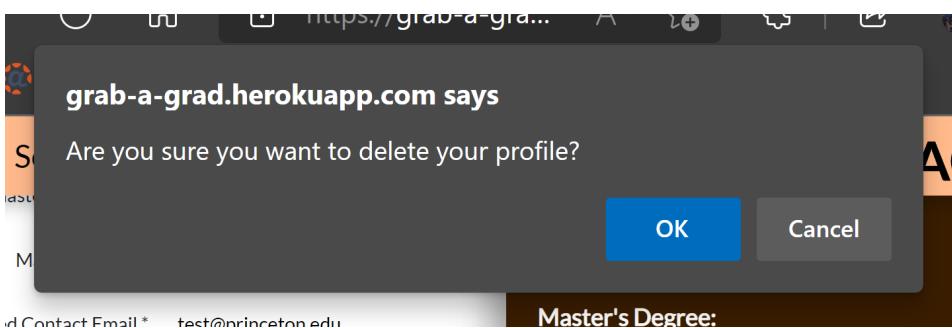
\* Indicates Required Field

- ❖ **Dialogues Box Before Leaving Page** - In the case that a user tries to leave the “Form Page” without submitting their information, a dialogue

box appears to notify the user that leaving the page will result in their information possibly being lost



- ❖ **Dialogue Box Before Deleting Profile** - In the case that a user attempts to delete their profile, a dialog box appears in order to make sure that this decision isn't accidental by asking them to confirm their action before proceeding



- ❖ **Restricting Input Type** - Restricting the type of input a user can enter prevents unsupported data from being entered in input fields. For instance, in the "Number of Years Worked" field, only numbers are able to be input

Years of Work Experience 2

## Recognition Rather than Recall

Our system prevents the need for a user to remember the information they input in one part of the application by storing their profile information and accessing

that information using their netid. Therefore, whenever a specific user enters GrabAGrad, we automatically fetch their profile information, if they have an existing profile, and fill it in the form for them so that they don't have to worry about memorizing it themselves. This specific functionality applies exclusively to the graduate students that create a profile.

For undergraduate students, the profiles they've favorited in the past are automatically saved and retrieved when a specific user enters the GrabAGrad platform. Saving each user's favorite graduated students prevents them from having to memorize the specific graduate students they liked in the past.

Since the "Search" feature dynamically updates the set of graduates shown to the user, the user's last search query is available until they modify at least one of the input fields, at which point the new set of inputs queries the database.

## Flexibility and Efficiency of Use

Our application implements several strategies in order to cater to specific users. For starters, our landing page immediately divides users into the category "Undergraduate" or "Graduate." This dramatically reduces the amount of time each type of user must spend navigating the site, as if the user clicks "Graduate" they are immediately brought to the page where they input their information to create a profile, and if the user clicks "Undergraduate" they are immediately brought to the page where they can search for graduates.

I am a...

Undergraduate

Graduate

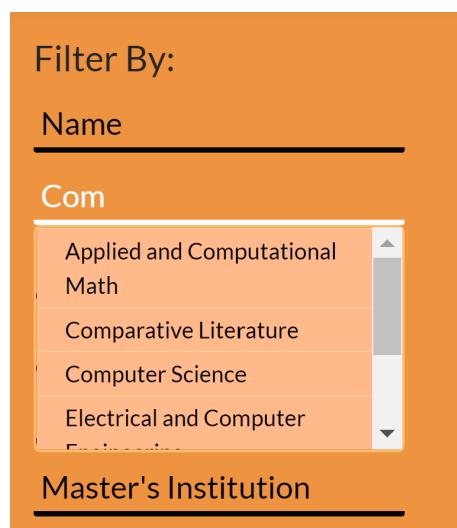
Additionally, depending on whether "Undergraduate" or "Graduate" is selected on the landing page, the links available in the navigation bar change. For instance, if the user selects "Graduate", the navigation bar for the "Create a Profile" page looks like this:

By only showing the “Home” and “Search” options to graduate students, their experience is made more efficient by simplifying the amount of options they have. They only need to see the Create A Profile / Update My Profile page and possibly peek at the Search page in the case that they want to confirm that their profile is updated.

However, if the user selects “Undergraduate” the navigation bar looks like this:

Providing a wealth of navigation options for undergraduate students serves to add efficiency to their experience by providing a clear overview of the functionality available to them, while excluding a link to the Create A Profile / Update My Profile page prevents them from wasting time navigating to a page that they aren’t intended to use.

In addition to the dynamic navigation bar, the drop down menus provided to users as they input specific pieces of information adds efficiency to their experience by reducing the amount of time they need to spend fully typing out their entire response, or even the time needed to think of a response, by suggesting autofills.



The dynamic side navigation bar on the search page enforces the flexibility of the platform by enabling users to expand and collapse the search window so they can see more or less graduates at a time.

### Expanded Search Bars:

This screenshot shows the GrabAGrad search interface with its sidebar navigation expanded. On the left, there's a sidebar titled "Filter By:" containing dropdown menus for Name, Department, Industry, Years Worked, Undergraduate Institution, and Master's Institution. Below these is a checkbox for "Only My Favorites". Under "Sort By:", there's a dropdown set to "First Name: A-Z". The main content area displays three graduate profiles in cards: Adelaide (Neuroscience), Alexandra (Mechanical and Aerospace Engineering), and Anna (Quantitative and Computational Biology). Each card features a large orange letter (A, X, A) and a "Learn More" button. Below the cards are small thumbnail images of the graduation photos.

### Collapsed Search Bars:

This screenshot shows the GrabAGrad search interface with its sidebar navigation collapsed. The sidebar is now a thin vertical strip on the far left. The main content area displays the same three graduate profiles (Adelaide, Alexandra, Anna) in cards, each with a large orange letter and a "Learn More" button. Below the cards are small thumbnail images of the graduation photos.

## Aesthetic and Minimalist Design

The color scheme of the site is consistent and only consists of four colors and a singular font style. This minimizes distraction of the user from the main functionality of the site due to ill-chosen colors.

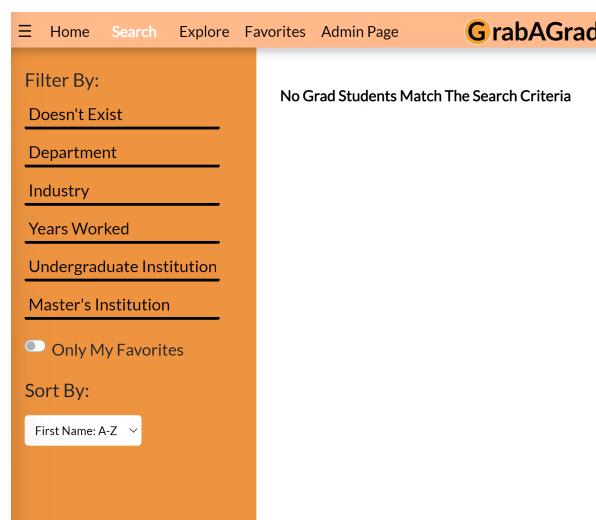
The application was designed with minimalism in mind, especially on the Search Page where a multitude of information is present. Much of the information is hidden and is only visible when the user requests to see it, by clicking “Learn More.” Utilizing pop-ups on the search page serves to give the user the minimal amount of information necessary at a time but easily provides additional information if the user seeks it.

## Help Users Recognize, Diagnose, and Recover from Errors

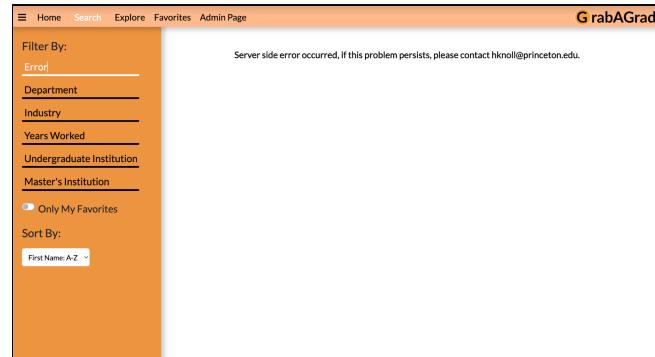
In order to make sure that users are able to properly understand the error messages they are presented with and accordingly determine how to resolve the errors, we implemented descriptive error messages in plain language.

Examples:

Message in response to the user inputting search criteria for a graduate student that doesn't exist.



Message in response to the database not being available.



This message serves to inform the reader why no graduate students are present when they click the “Favorites” page without previously favoriting any graduates.



## Help and Documentation

We purposefully included a detailed “About” page on our application that gives users a brief overview of the functionality of the application and how to best utilize the platform.

Additionally, the suite of documentation including the User’s Guide and Programmer’s Guide that correspond to this Product Evaluation Document provide a detailed set of instructions on how to best utilize GrabAGrad and resolve any issues.

## Appendix

### User Testing Tasks

1. Click **Undergraduate**.
2. Login to CAS.
3. Search for graduates that interest you

4. Visit and play around with the Explore page
5. Visit and play around with the Favorites feature
6. Click "Home"
7. Click **Graduate**
8. Fill out the form
9. Update your profile if necessary

## User Responses / Notes

### User 1:

- ❖ Big font so easy to read
- ❖ Easy to understand
- ❖ Pretty intuitive
- ❖ Layout is easy to understand
- ❖ Filtering is useful
- ❖ Critique
  - Random matching
    - Student enters their details and then they are randomly matched
      - May make more sense as the database grows and you have more people
  - Home and GrabAGrad logo should lead to the same page
    - A user wouldn't know they should click on the logo
  - What's the incentive for both undergraduates and graduates to join
    - Need context

### User 2:

- ❖ GrabAGrad is an interesting name but it implies that undergrads are the user, which is fine but consider how this makes a grad student feel. Mind you if you know a grad student it would be good to get their feedback
- ❖ The landing page is nice in terms of simplicity and directness BUT I feel like there should be a bit of info about what this platform is about. Perhaps a quote/header/slogan "Connecting Undergrads to Grads." If a complete stranger were to come on to your site, they would have no idea what they were doing.

- ❖ When I log in I am greeted by a much busier page which can be a bit jarring to users. It's a bit of an information load and I'm not sure where to start. It would be good to have instruction pop-ups.
- ❖ I don't think your "home" page should be the "I am a..." page. They should be separate.
- ❖ The Search function and page looks good BUT let's say I find someone cool. I have their email sure but why not make a button that says "Email me!" Making the user copy and paste an email can be annoying. Otherwise this seems to some degree like a contact/phonebook list but with graduates too.
- ❖ Explore page is alright but I feel like you could make it more fun. What if you had a Tinder-like function where you swipe left if meh, swipe right if cool, swipe up if neutral. Especially if you have a lot of contacts this is going to be a very very long explore page that will become boring if there is not a more active engagement aspect of it.
- ❖ I feel like you should add an "availability" section. Assuming the goal of this app is not just provide contact info but build connections, highlighting availability or desire (like what does a grad student want from an undergrad) that would be very helpful.
- ❖ Overall, I like the straightforwardness of the app but I do think there could be some clarifications and more intractability to really push this further beyond a contact sorter for grad and undergrad students. I could see myself potentially using this but the value proposition needs to be clearer and it definitely needs to have an availability section.

### User 3:

- ❖ Style
  - The names of the grad students on the main page are slightly cut off by the white area containing their major and it draws my attention more than other things that I should be focusing on
  - Having the logo on the top right of the screen makes me not pay attention to it, I didn't even notice it was there until about five minutes into my testing and didn't think about clicking it for another couple of minutes
    - I feel I just expect it to be in the top right and the current placement throws that off
  - I didn't like the brown on the “about” page:
    - It felt a little out of place, but maybe I'm just not a fan of brown

- The home page feels a bit unbalanced with the placement of the undergrad and grad buttons
  - Like I can see that it was meant to center the div that surrounds them, but I feel like it might be better visually if the gap between them was in the center of the page instead
- ❖ Content
  - I didn't know what the website is meant to do at first
    - Maybe add a little blurb on the home screen with a sentence or two explaining what the user is supposed to get out of this
      - like a shortened version of the about page, or at least a link to the about page that isn't just the logo
- ❖ UX
  - As long as I have logged in through CAS, I can view both the graduate and undergraduate pages without checking permissions
    - I can also go straight to the grad search page without making a profile by just manually setting '?user=grad'
  - In a similar vein to the above point, I can view your ajax requests by just manually entering the urls for them i.e.  
`'/load_header_tabs?user=undergrad&page=explode'`
    - Idk how much you can do about this atm though
  - This isn't a super big deal, but when I click to view the profile of a graduate student, I can still scroll the background page and it feels like I shouldn't be able to do that
  - The 'Explore' page doesn't really feel like it fits what I expect when I click on something named explore
    - It basically feels like a 'View Random Graduate Student' type page because there is no explanation of what is going on
    - This might be a purposeful feature, but I can infinitely scroll through the grad students and there is no indication of any repetition except for my memory and I think it might be nice to have something

## User 4:

- ❖ Overall nice intuitive layout! It was easy for me to figure where to find things
- ❖ I think a “clear search” button would be nice for when you’ve typed out a lot but want to get rid of it

- ❖ Since the contact information is the most important part on the card, maybe it could be highlighted more? A mail:to send email button might be helpful
- ❖ A bit more context on what the explore page is would be helpful — wasn't sure if it was just a random grad student each time or something else
- ❖ Smooth scrolling on back to top might be a cleaner UX

## User 5:

- ❖ Positives:
  - sleek UI
  - straightforward to use
  - I like how it automatically applies filters as you type them
- ❖ Constructive criticisms:
  - everything is a string search, but doesn't match keywords (i.e. transistors won't match to semiconductors)
  - why isn't everybody required to fill in all identifying fields?
  - maybe add a field that is PhD vs Master's, and what year grad student they are
  - the main menu is legit way too big
  - allow for 3-letter abbreviation when searching for a department
  - add filter by department alphabetically?
  - maybe grey out the filter names, to emphasize on the screen what you typed and what is just a label

## User 6:

- ❖ Typed in acronym for academic department as opposed to the full name
- ❖ Wants suggestions for industry
- ❖ What do the grads presented want to talk about?
  - Is there a specific thing people want to be contacted about?
    - Is it more casual or is it for research, etc?
- ❖ What exactly do the undergraduates get out of the networking?
- ❖ It would be helpful to see what professors each graduate student is working with for research purposes.
- ❖ It could be helpful to see what courses the graduate students are affiliated with and their position
- ❖ Years of work - does that include just industry experience or also research experience

- ❖ If the grad is affiliated with a lab, that lab should be linked
  - If the grad has a personal website, that could be linked
- ❖ It could be intimidating for an undergrad to reach out if they don't know exactly what the graduate student is looking to be contacted about
- ❖ Likes the layout

## User 7:

- ❖ I think the website is super easy to use and to understand and I really like the explore and favorites sections since it is easier to sift through and save people you like with.
- ❖ I feel like it's a little unclear how to find the "How it Works" page, which I think is maybe a useful page, but I also think that's a minor problem since the concept and execution are very intuitive.
- ❖ I also really like the mockup card when you create a profile as a Graduate student.
- ❖ The filter works great, but maybe an easier thing to use would be if the department section was a dropdown since then it'd be easier to see all the possible department options.
- ❖ The page looks really nice and I do like the cards, especially since they have the most important features on the front.