

Module 8 – LinqUp Design Challenge



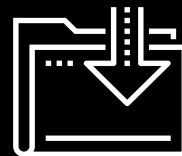
Prisha Rai



Mini Case Study

UX/UI Design Boot Camp

Module 8



Overview

What is this case study about?

This case study is about helping LinqUp develop their app further by building a new feature so that customers can find and connect with a career mentor.

Why did you do this work?

So that both coaches themselves and people looking for mentors have a convenient way to find each other.

What did you expect to learn?

I expected to learn more about the users and what they want, how the app itself works, and the good and bad of the app currently

Problem Statement

What was the problem?

The problem was that user's aren't able to find career coaches or people to coach easily

Why was this work necessary?

So that users are able to do everything they would like to on LinqUp with ease

Problem Statement

What was the problem?

All user's of Linqup, especially those looking for coaches/people to coach.

Why was this work necessary?

Solutions to the problems user's have been experiencing with the app.

Role(s) and Responsibilities

Who participated in this work?

Prisha Rai

What did they do, specifically?

- Conduct a heuristic evaluation
- Complete a user task flow in which the end goal was to find to make a “connection” with
- Analyze and annotate task flow
- Review user feedback

Scope and Constraints

Where does the work begin and end?

The work begins with me using the app myself, to see it through a user's eyes, and ends with solutions to all the heuristic issues, including a few new features for career coaching, decluttering, connection requests, and the difficulties for new users.

Were there limitations or challenges you had to overcome to complete this work?

I had to overcome the challenge of not having much familiarity with apps like these, so it took longer to understand and learn what needed to change because of less time spent on the app.

What was the timeline for completing this work?

It started with going through the task flow myself to understand it. Then analyzing it further for usability and interface heuristics. Then I took into account actual user's complaints to annotate the task flow further, so that I would taken into account all the changes that needed to be made and how.

[Link to Heuristic Evaluation Checklist](#)

Scope and Constraints

What methods were used and why?

Used methods such as analyzing, annotating, reviewing, and reflecting

What steps did you take?

After going through the user flow myself to take note of the heuristic issues, I also reviewed user complaints to put both opinions together to make the app even better

How did you document your work/findings?

I documented my findings by conducting a heuristic evaluation first and then annotating each page of the task flow after having a look at the feedback from users

What happened on the journey to the deliverable?

Multiple testings in which the new feature would be tested from both sides of the user flow (those looking for coaches and those looking to be a coach). The other heuristic issues noted would be tested separately as well.

Outcomes and Lessons

What were the key takeaways from the work done?

I learned much more about heuristic issues and what type of issues to look out for. I was also in general, able to learn what things need to be changed for LinqUp to be more successful. I decided on whom I would reach out to with a connection request based on whether they first fit the qualifications of the initial filters I put in, and then by reviewing his profile. I used info such as the college he went to, his skills, and past jobs to decide. I would have liked to ask if he was willing to speak with me so I could learn from him. I also would have wanted to know how he was introduced to field and what he likes about it. I feel like he would be quite a good mentor for me because on the many similarities I could see based on his profile.

What lessons did you learn?

Lessons I learned are that sometime thing you have no idea would be a problem or request from users could actually be something they really want fixed.

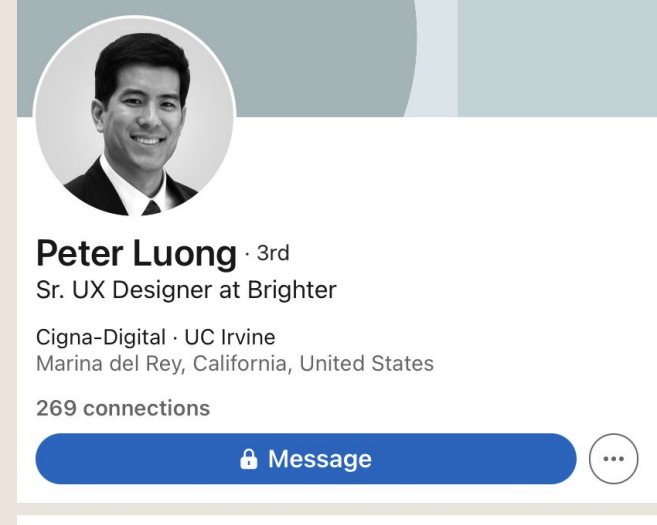
What will you do differently on your next project?

I would probably prefer look out for heuristic issues from the get go so analyzing and checking for them wouldn't take as long.

Heuristic Issue #1

The first heuristic issue is informed from the user feedback and is in regards to coaching. Many user's who are hoping to coach would like to make their services available to people in their industry, but there's no easy way to do so.

Profiles open to coaching, should have the option to have an additional button for people to request coaching from them. Owners of these profiles should also have the option for all potential students to be required to answer the questions written by them (if the coaches make it a requirement to fill out).



Heuristic Issue #2

This heuristic issue is informed from the user feedback and is in regards to coaching as well. Many users are hoping to find coaches/mentors to help them from this app, however there is no category or indication for it to be easy to find them.

In order for it to be easier for users to find coaches, there should be an additional filter option when searching. Users would then have the choice to choose whether they want all the people who show up in their search results to be a potential coach. It would be formatted similarly to the connections filter and not how the other filters are formatted.

Filters

Reset

Connections

1st

2nd

3rd+

Connections of

Any

Followers of

Any

Locations

California, United States X

Edit

Current company

Any

Past company

Any

School

Any

Industry

Any

Profile language

Any

Open to

Any

Service categories

Any

Show Results

Heuristic Issue #3

Another heuristic issue is just that many functions may be confusing for new users such as app specific terminology/actions as well as other things. For example, I did not have an understanding of what the 1st, 2nd, and 3rd+ types of connections really meant and therefore had no idea if that was something I wanted to use as a filter.

A solution for this would be to give brand new users the option to go through a brief guide to understand how to use the app. After creating their account, users would be asked if they would like this guide, and if marked yes, the app would guide them to press and view different buttons/pages while giving them explanations. If they would like they could easily skip this as well.

Filters[Reset](#)

Connections

1st

2nd

3rd+

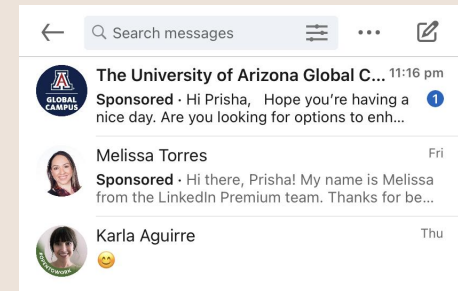
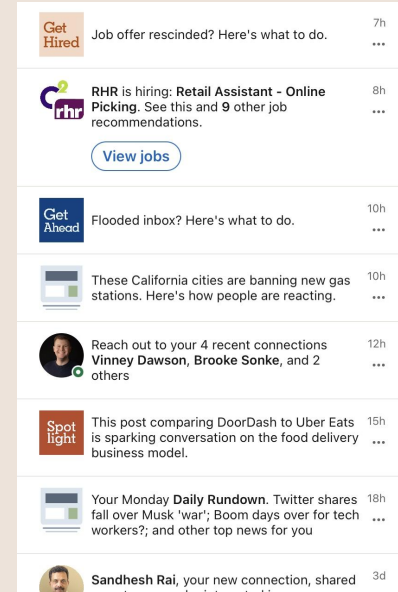
Connections of

Any

Heuristic Issue #4

This heuristic issue is about how much extra clutter there tends to be. Despite me only creating my own account 5 days ago and only having 10 connections, I have already received 2 sponsored messages and numerous notifications I would not care about as a user. Due to how many there are, I would probably miss the notifications that I would care about. Received numerous emails as well.

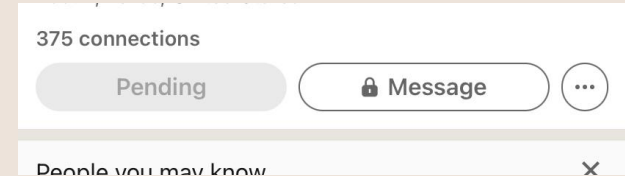
The way to solve this issue is to first eliminate the unnecessary emails. Users already have the option of choosing to get notifications from the app. It is understandable that many might not want to download the app, however they should be able to choose themselves just like requests for apps to send notifications are there. Many of the types of notifications are unnecessary as well, such as the “reach out” and “_____ shared a post” notifications. The second type of notification should only show up if the user wants to be notified for that specific connection.



Heuristic Issue #5

The final heuristic issue I noticed was the issues with requesting a connection. Whenever a user presses on the button to make a connection, it automatically goes to pending, waiting for the profile owner to accept or decline. However the problem with this is that until they accept/decline, the user who requested the connection cannot undo their request and at first are not even aware that they cannot undo the action. This can cause problems because user might accidentally press the button or simply change their mind as well.

To solve this, all users should be able to take back a connection request. It would function just like it would when cancelling a regular connection.



[Link to Figma frame with annotated heuristic issues](#)