

Project 3 – Group 6 Documentation

Joe Ferguson, Joanne Osuchukwu, Eli Pappas, Chamaroke Umunna

Part	Pages
<u>Project Description & Research</u>	2&3
<u>Results of Research</u>	4-6
<u>Initial Design & Feedback</u> -> <u>10x10 Sketches</u> -> <u>Finalized Vanilla Sketch Based on 10 x 10.</u> -> <u>Figma Sketches</u> -> <u>Feedback</u>	7-13 7-10 10-11 11-12 12-13
<u>Implementation & Progress</u>	14-16
<u>Final Project Description</u> -> <u>Describe Interface In General</u>	17-23
<u>Demo Video & Links</u>	24

Project Description & Research

1) Choose a UI to fix: Identifying a UI that exists in the world that you think could benefit from a fundamental redesign. It is going to be better if you find a UI that has a significant flaw, or for which you think a significant re-envisioning would be valuable and interesting to explore. Don't pick a UI that is generally good, but that could be made better with a few small changes.

We are choosing the AltaFiber mobile UI.

- Submit this UI for a project checkpoint
Completed
- Present your initial idea for the redesign- considering whether you think the issue is a usability or design problem (A above) or you want to retarget the UI for a different goal (B above).

Big usability problem, it's hard to find out details concerning the service, contacting people when help is needed. Convoluted settings and overall design.

We will redesign to make the UI easy and fast to use, and cover all the need features for a internet provider.

2) Redesign research, data, planning: Follow a process for planning your redesign, and gather research + data.

A good redesign process will proceed from a solid motivation, draw from research or data, and be specific in diagnosing the problems. Since everyone's project is different, you need to consider what research you need to do or data you need to gather to justify and ground your redesign.

Do you think your UI has usability issues? What precisely are those issues and how serious are they? Do you think your UI could benefit from targeting different goals? How?

Select from these methods to advance your understanding and justification for the redesign:

The methods our groups intend to use for the project are methods:

- A) Capture UI/UX metrics, with a focus on self-reported metrics
- G) System Usability Scale

H) Survey

We decided on these by voting and then discussing them, shown below:

- Joe F – I would like to use method A, C, and G for our design research.
- Eli P – I would like to use A, B, and G
- Joanne O – A, C, D, G
- Chimaroke U – I would like to use G and H.

a. Capture UI/UX metrics:

g. System Usability Scale : [System Usability Scale](#)

h. Survey: [Survey](#)

Present a clear and detailed description of your findings from the methods you select. What precisely needs to be addressed. This means instead of writing "Users make too many errors", you write: "Users make the following kinds of errors in the following tasks - list them". Instead of writing "Design choices are poor", present the outcome from your heuristic evaluation with specific design problems. Instead of saying "UI hard to learn", observe a user trying to learn the interface and note their points of difficulty. Instead of saying "This UI hurts mental health", describe arguments and data from articles which explain how this UI or kind of UI may contribute to mental health problems.

Results of Research

A)

Self-reported metrics

- Users report their experiences, satisfaction, emotions, impressions...
 - Peter (User 1) found the Messages section confusing, since you must open “Messages” and then “Inbox.” He also said he never uses the Fioptics TV feature under “More.”
 - Joseph (User 2) said the blog on the home page isn’t needed, and that it makes the page look messy.
 - Frank (User 3), a new user, couldn’t find his profile right off looking at the interface as he had to go through the nav to finally see it hidden under “More,” which wasn’t intuitive.
 - Joanne (User 4), mostly uses the app from memory, doesn’t use the TV cable option, and said the home page should have quick links for common tasks and would love to see essential information on the homepage.

Performance metrics

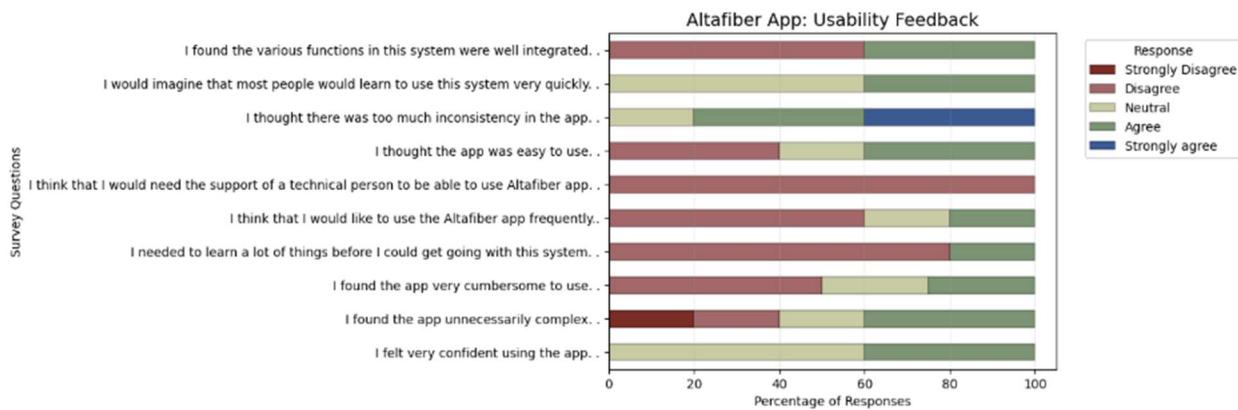
- Measuring user behaviors
 - During the think-aloud, users were asked to check their bill, access messages, enroll in autopay, find help for a billing issue, and contact support.
- Task success, time on task, errors, efficiency, ease of learning
 - Frank (User 3) spent time searching for his profile, clicking through several options before finding it.
 - Setting up autopay wasn’t difficult for users, and most were able to complete it without help.
 - Users said the same thing about reaching the Inbox page, which required extra steps going through multiple layers.
 - Users mentioned that contacting support wasn’t very efficient, as they had to go through several nested menus to find it. They said there should be a direct button in the Help section to access support faster.
 - User 3 clicked on the Total Bill amount, thinking it was a link that would open more details, mainly because of the way it was styled

G) Survey, we sent out these questions to people who were familiar with the app and compiled the responses into an excel spreadsheet. The results helped us gain insight into what is important and what needs to change. After looking at the results we found that transparency and simplicity is important and people don't use the app often as well as don't think the design looks appealing in general.

1. How often do you use the altafiber app? not often / sometimes / often
2. How frequently do you use the help button? not often / sometimes / often
3. On a scale of 1-5 how would you rate the overall user interface on looks?
4. On a scale of 1-5 how convenient is it to pay your bill?
5. On a scale of 1-5 how easy is it to find the bill summary and information?
6. On a scale of 1-5 how important is simplicity on the app to you?
7. On a scale of 1-5 how transparent is the billing cycle and due dates?

Questions	user1	user2	user3	user4	averages
How often do you use the altafiber app? not often / sometimes / often	not often	not often	not often	not often	not often
How frequently do you use the help button? not often / sometimes / often	not often	sometimes	not often	not often	not often
On a scale of 1-5 how would you rate the overall user interface on looks?	2	3	1	2	2
On a scale of 1-5 how convenient is it to pay your bill?	3	4	2	3	3
On a scale of 1-5 how easy is it to find the bill summary and information?	3	4	3	2	3
On a scale of 1-5 how important is simplicity on the app to you?	4	5	4	4	4.25
On a scale of 1-5 how transparent is the billing cycle and due dates?	4	3	3	5	3.75

H) We used a short Microsoft Forms survey with 5 respondents to gather user feedback on the Altafiber mobile app, based on the **System Usability Scale (SUS)**. The results revealed specific patterns:



- **Perceived complexity:** Responses on unnecessary complexity were evenly split between agreement and disagreement. Respondents found the app cumbersome.
- **Learnability and support:** Responses to needing to learn a lot were mixed, with many selecting Neutral. However, respondents disagreed that they would need technical support, with only a minority agreeing.

- **Confidence and frequency of use:** Confidence levels clustered around Neutral and Agree. However, 60% disagreed they would use the app frequently, and only 20% expressed interest in frequent use.
- **Consistency and integration:** Most respondents agreed there was too much inconsistency in the app. And most disagreed functions were well integrated.
- **Ease of use and learning speed:** Ease-of-use responses were balanced across disagreement and agreement. Views on whether most people could learn the app leaned towards neutral and “agree”.

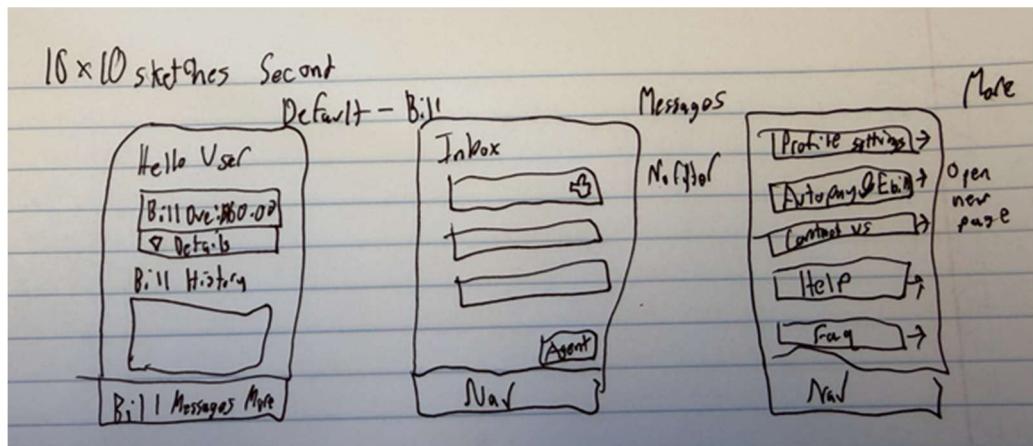
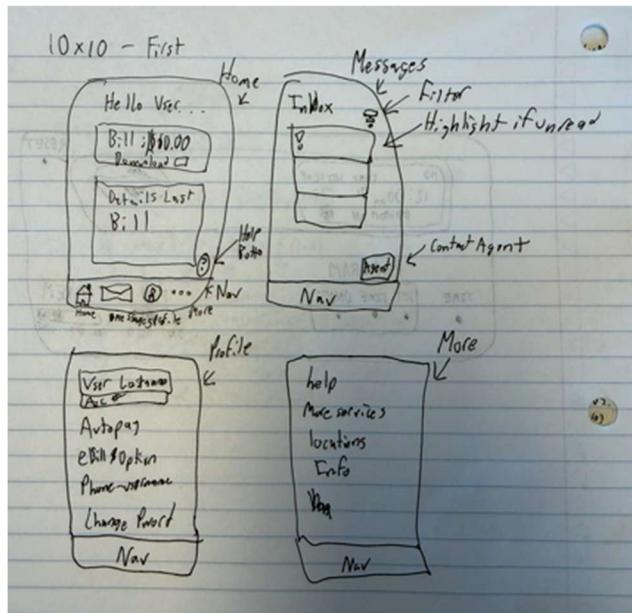
Conclusion: Key issues indicated by the chart are inconsistency, low desire for continued use, and uneven user experiences with complexity and ease of use. These findings justify a redesign focused on simplifying interactions and creating a more consistent, coherent interface.

Initial Design & Feedback

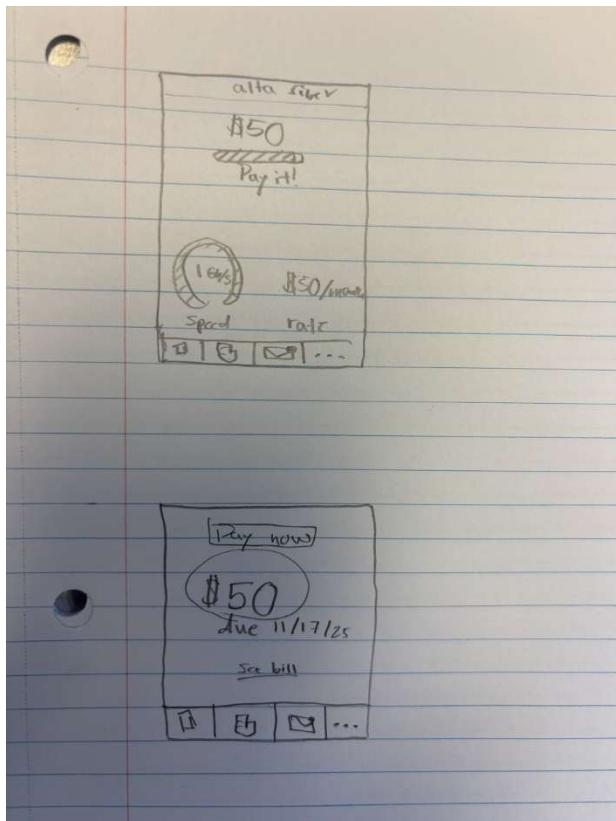
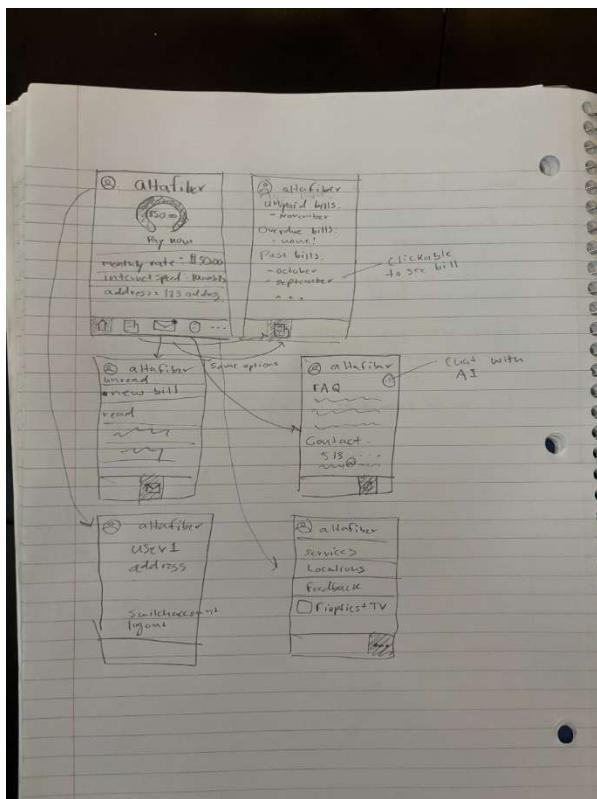
3) Sketch and get feedback: Use sketching and digital prototyping tools (e.g., Figma) to plan this redesign, and get feedback.

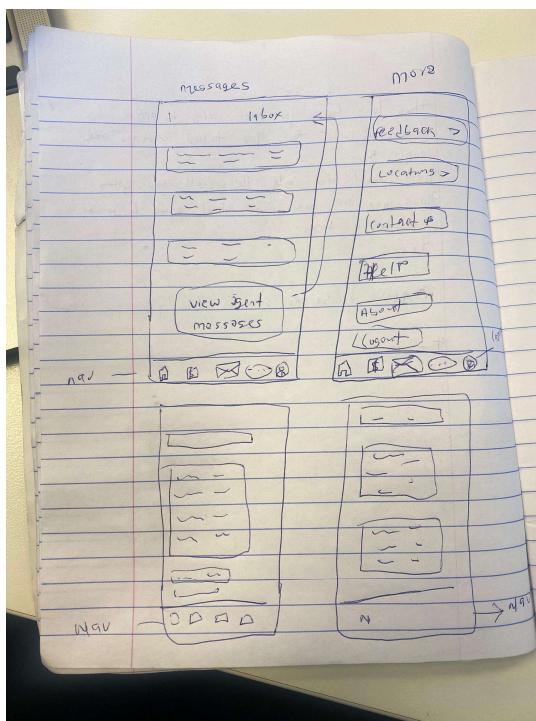
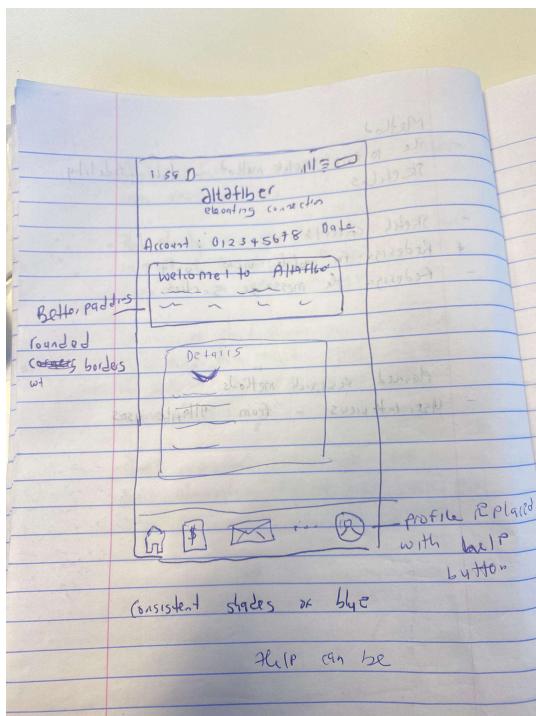
10 by 10 Sketches

Joe F-

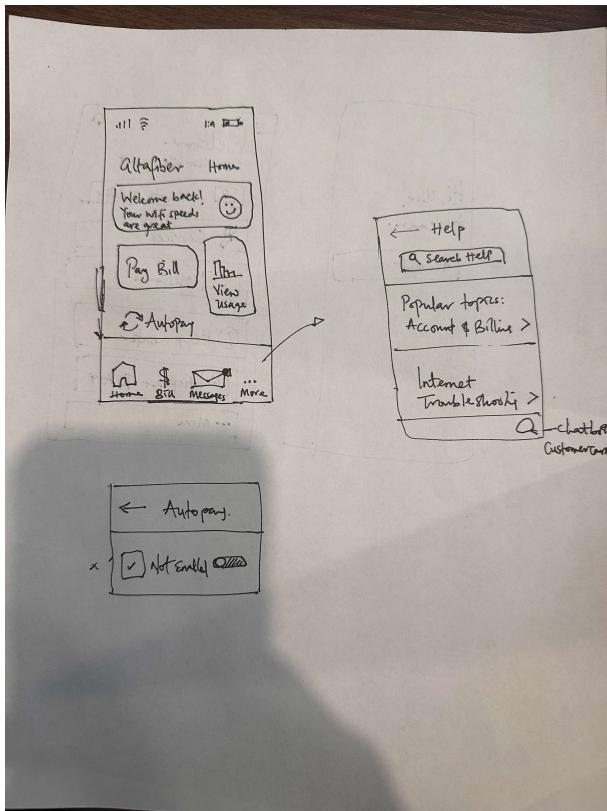


Eli P-

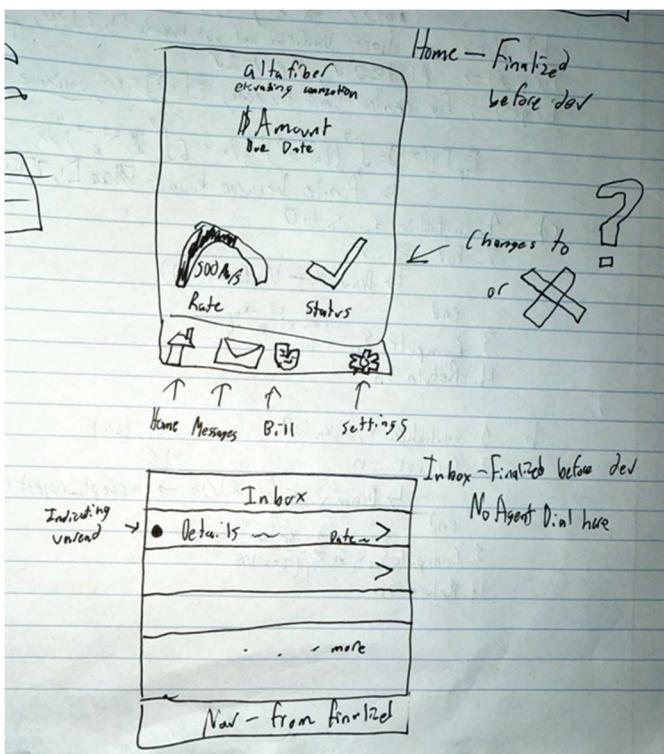


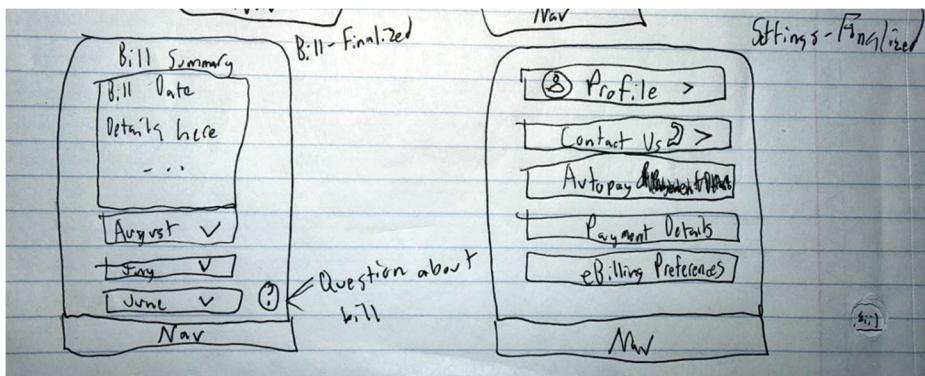


Joanne O-



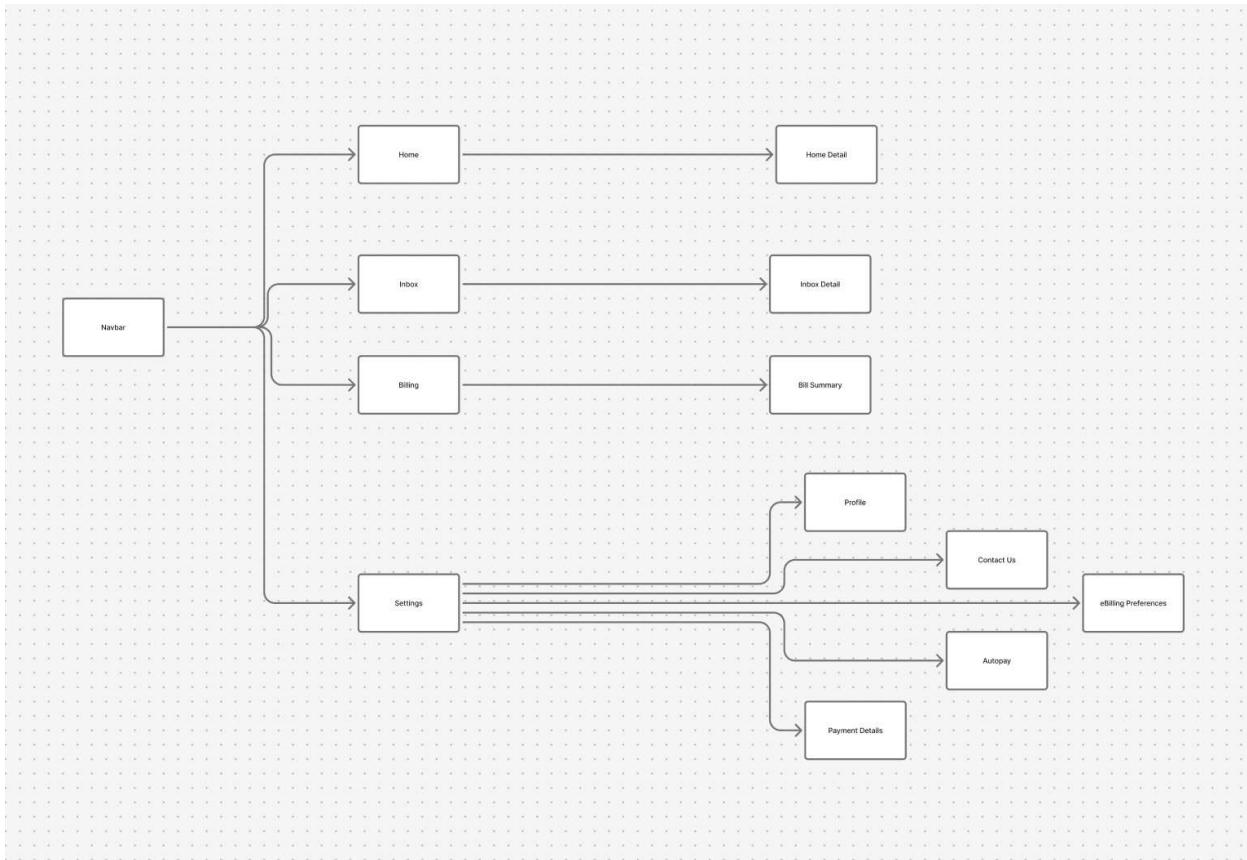
Finalized Vanilla Sketch Based on 10 x 10.





Figma sketches:

Low fidelity wireframes



Flowchart

Feedback:

Get and present feedback on these design plans. Your goal will be to gauge whether your redesign will address the identified issues before implementation. Choose a method to get this feedback.

We chose method H – writing a survey for getting feedback. We had the same people that participated in the research survey participate in this survey to have a good standpoint to work from. We copied each question that was possible to answer off the design wireframe alone, and excluded the ones that would need the fully developed UI to answer. We also included more questions based on much how the drafted design improves on the weaknesses of the old/original UI.

Questions (scale 1-5):

1. I would imagine that most people would learn to use this system very quickly.
2. I think that I would need the support of a technical person to be able to use Altfiber app.
3. I think that the new design is easier to understand than the old design.

4. I think that the new design is faster to use than the old design.

	user1	user2	user3	user4	averages
I would imagine that most people would learn to use this system very quickly.	5	5	5	4	4.75
I think that I would need the support of a technical person to be able to use Altafiber app.	1	1	1	1	1
I think that the new design is easier to understand than the old design.	4	3	5	4	4
I think that the new design is faster to use than the old design.	3	3	4	4	3.5

The responses we received seemed to indicate that we did a good job. They agree that most people would learn to use this very quickly, they were unanimous in agreement that no one should need the support of a technical person, they agree that the new design is easier to understand, and that the new design is faster than the old one. Some variances are obviously expected out of personal preference, but no one seemed to think that the new design was worse at all and better generally and some people really liked the new design much more than the old.

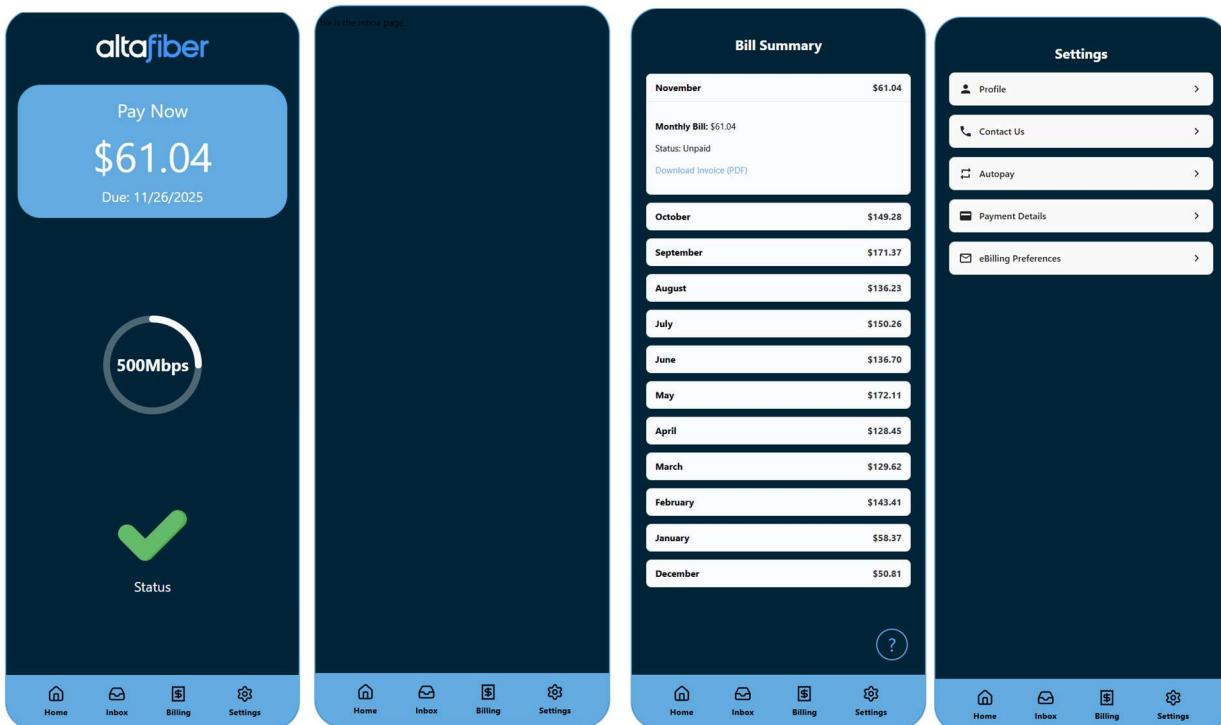
Implementation & Progress

Implement some meaningful portion of this envisioned redesign in a working prototype, written using Svelte or SvelteKit and Javascript. This prototype should be hosted publicly, so we can test it.

As with other projects- think of this as a prototype to illustrate the new design. Select the most interesting aspects of your redesign- no need to spend time on login pages or a routine settings page if this is not illustrating core goals in your redesign. By "a meaningful portion" I could envision a main page and a subpage or two. This depends on the UI you select and the major issues you want to tackle.

As with other projects- there is no need for a backend or a database. I tend to think this takes time away from front-end work and introduces un-needed complexity to projects in this class. This time, however, I won't penalize you if you want to do a backend with a database. Just make sure you can publicly host your code for us to test. Do this at your own risk, because it does take time away from the front end work.

Progress – 11/15/2025 – 6PM



Notes – Inbox was not prototyped yet at this time. Most clickable changes not functional. Some functions from the original app left out at this time. Not actual “pay bill” view available at this time.

Progress – 11/15/2025 – 9PM



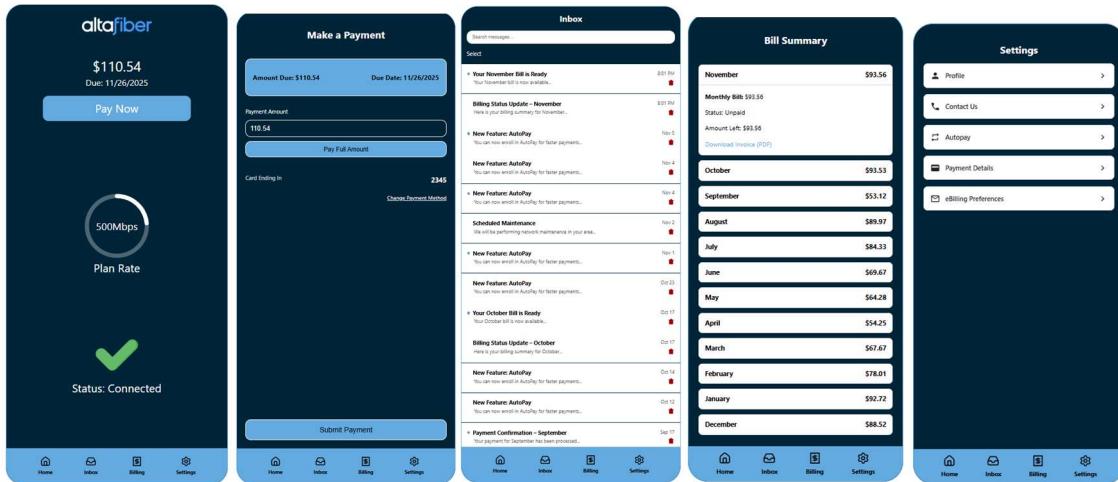
Inbox still not prototyped at this time. Some clickable changes added. Formatting standardized and payment page added. Review needed with group on stuff to add/change. Would like to have the routing store information so back buttons for settings forms actually go back to what form was at before.

Progress – 11/16/2025 – 9PM



Inbox component is currently functional and updated

Progress – 11/17/2025-8PM



Last progress update. Backend changes & standardization. Color / styling changes based on professor feedback and where saw fit basis. Alerts for unfinished buttons/functions, and example invoice fill download finished.

Final Project Description

Documentation:

As before, document your work so you can showcase this project online. This can then be used as a case-study for UI/UX positions.

For documentation: assume that someone is encountering your project for the first time. This documentation must be publicly available through one group members portfolio page. I strongly recommend making a personal copy on your portfolio page, and copying the repo to your git account, in case your team mates someday take their portfolio page down.

- Describe the project

We are redesigning the mobile Altafiber app because we believe its current design is very cluttered, confusing, and slow to use. This would fall into category A in the project description given on canvas. Essentially, the current UI is not efficient, and it has significant design flaws such as being unnecessarily complex, on top of having a poor layout and design.

- Present your design work

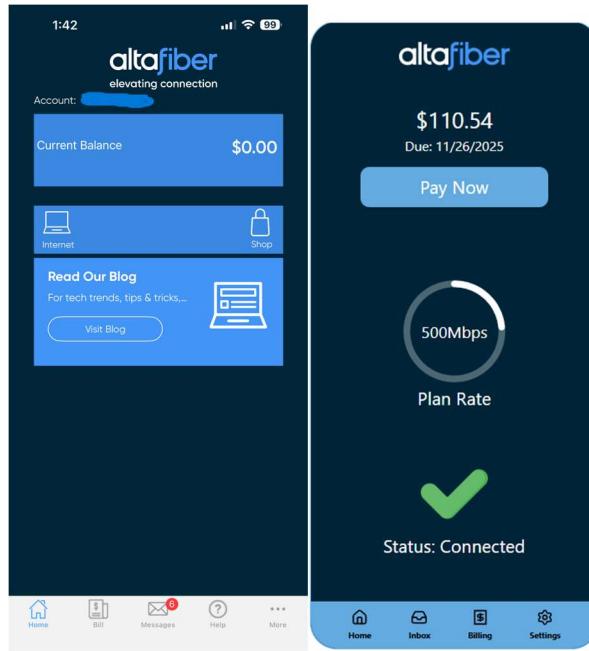
- *** For this project it is particularly important to explain and justify your redesign decisions. Be sure to present your research, methods, and findings clearly and comprehensively. ****
- Found in the earlier parts of this document [Project Description & Research](#)

Describe your interface in detail:

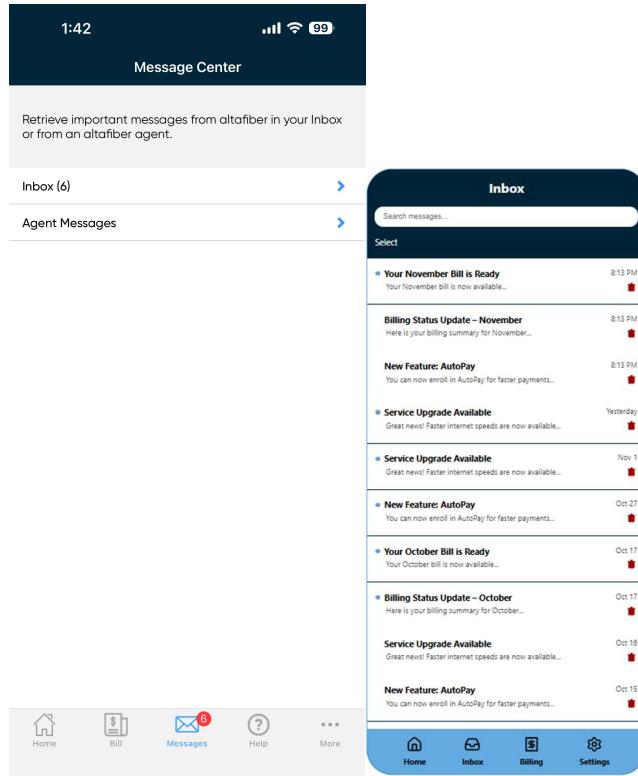
- Explain the features and controls
- Include plenty of screenshots to illustrate your interface and different actions users can perform within it

---All will be in before/after format---

Home/Landing Page

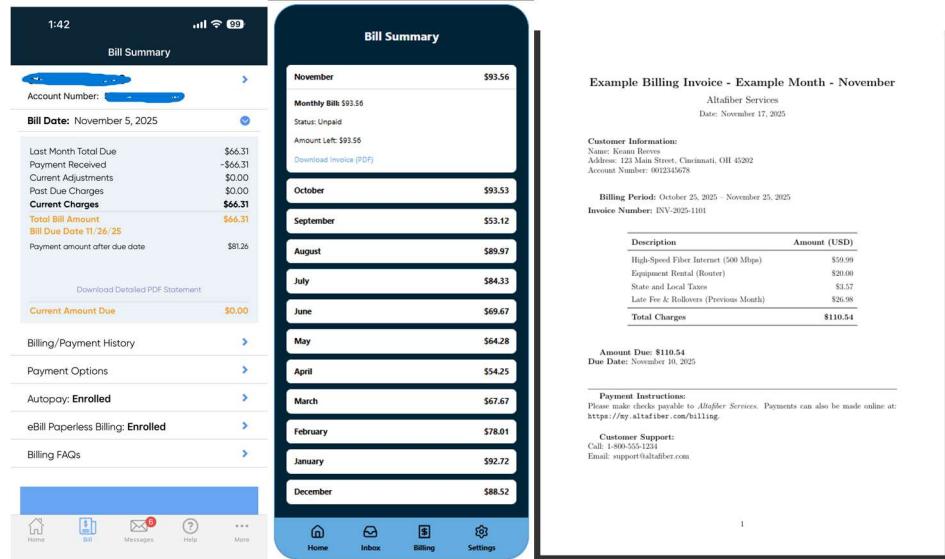


The Home Page is simplified and provides more details up front. The outstanding bills and payment date is front and center and easy to use if that is all you come to the app to use. The plan rate is another piece of information that we added to quickly glance at what rate you pay for as people may forget. The status is also easy to glance at to check if your internet is out. Finally, the styling has been modernized and utilizes altafiber's main colors to provide a clean looking aesthetically pleasing appearance. There is also a feature for testing that if you hit shift and tab at the same time you can toggle the status.

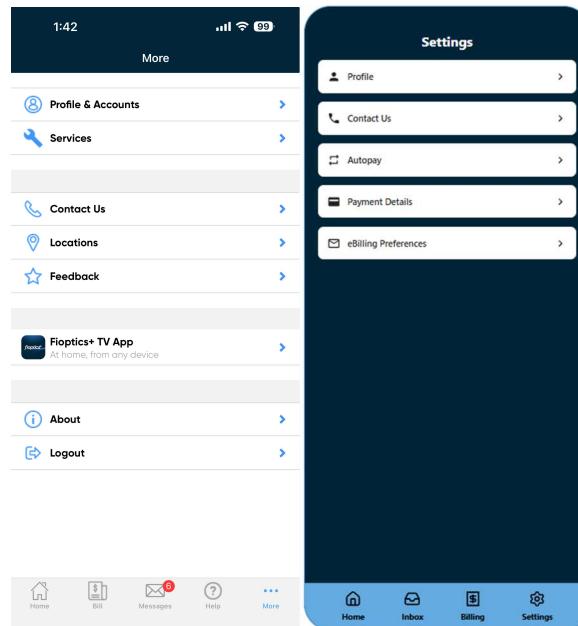


The inbox page lets you view all messages sent to your account. Instead of being nested inside another screen, it's now designed so that once you navigate to the inbox, your messages appear immediately for a cleaner, more direct experience. You can open any message to view its full details, and the layout keeps everything easy to scan at a glance.

We also added a search feature that lets you quickly filter messages by title or preview text, making it easier to find what you need. Along with that, there's now a mass-select and unselect option, so you can choose multiple messages at once for deletion instead of removing them one by one.



Bill page is a page to view the bill summary for the last year (12 months) of your account. You can see the total amount, the status (if it's paid or not), and if it is unpaid, the amount you have left to pay. You can also download the invoice, which is a pdf that contains the individual charges and taxes and whatnot. Currently the invoice pdf is just a template/example one.



Settings page is redesigned to improve clarity and usability. In the original app, Settings was buried under a cluttered “More” tab with unrelated items like “Locations” and the Fioptics TV App, making it hard for users to find essential controls

In our redesign, Settings is now a main item in the bottom navigation bar for quick access. It includes only the most essential user-focused options:

- **Profile:** lets users upload a profile photo and edit their name, email, and phone number. Fields include validation, and changes are saved through a single button.
- **Contact Us:** provides a short form with fields for name, email, and message, with required field checks before submission.
- **Autopay:** includes a toggle switch to enable or disable automatic billing, shows the last 4 digits of the current card, and allows users to update their payment method.
- **Payment Details:** includes fields for card number, name, and expiration date, all validated for format and required input before saving
- **eBilling Preferences:** toggle for switching paperless billing on or off, along with a preview of the email address linked to billing notices.

Each subpage uses a consistent layout with clearly labeled actions (like “Save Changes” and “Send Message”), form validation to prevent incomplete entries, and back buttons for smooth navigation. All user and billing data is managed centrally using a shared writable store, which keeps the UI state in sync across components. These updates make the Settings experience more organized, responsive, and user-friendly.

- Explain how you implemented this application (libraries, code structure....)

We used regular svelte (version 5), not svelteKit or any other extensions. We used the packages Router from svelte-spa-router for navigation between views and Iconify for transparent-background logos for the navigation pane and some of the other icons.

For coding, we split every separate view into an individual svelte file, main views in src folder, and sub views in respective folders. This helped not only with abstraction and organization, but with project development with 4 different people (we didn't have a single merge conflict!!).

We also had our main app.css styles contain color variable declarations for easy use in the individual svelte file CSS styles.

I believe most of us utilized copilot in VsCode for autocompletions and boiler plating, as well as supplementing AI for filler data generation or svelte-specific questions.

- Optional- Use of AI- If you used AI, describe how. Did you identify strengths/limitations of using AI for your application.

We used AI for filler data generation, such as the billing history and user details. We also used AI for svg generation, and for the template for the example invoice latex format.

The strengths of AI include the previously mentioned functions, as well as help with svelte syntax & available packages. It also is helpful for generating boilerplates for svelte projects (script section, div content, styles).

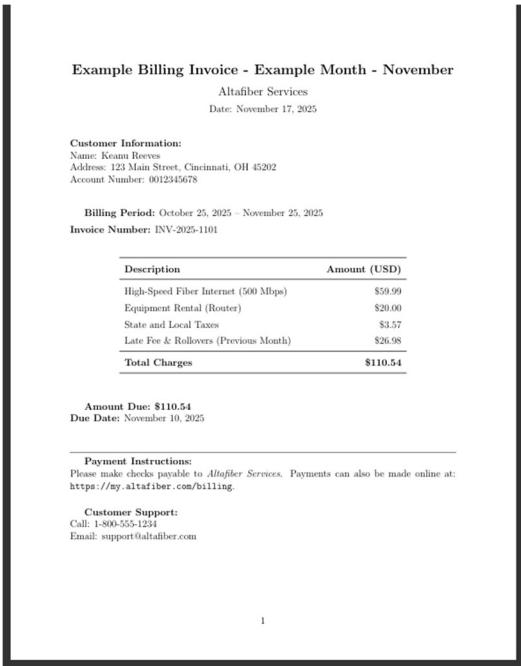
The weaknesses of AI are design, including any CSS styling at all. It also can't "think" project-oriented, so large queries/tasks fail to capture the overlying goal.

- Future work- No project is ever fully done. What would you do next? This is also a place to discuss the work you attempted but could not fully complete before the project deadline- include screenshots to illustrate and document your progress.

In the future, we would finish the reply and forward functionality of the message inbox, as well as the invoice pdfs from the billing summaries. We can build upon the security of the website (no encryption for messaging). We can also build a backend and transform the application into a mobile app (instead of a website that looks like a mobile app).

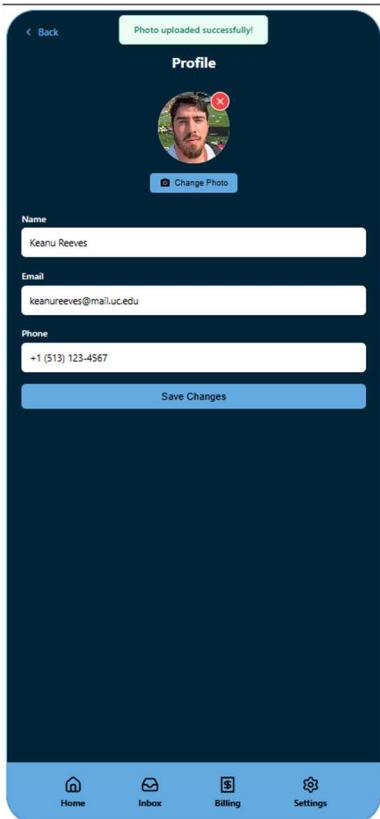
As far as work we attempted but did not complete, we didn't really have any tasks left uncompleted. If anything, we developed more than we originally planned. For instance, the settings items originally were not going to be functional (just have an alert or whatnot), and the example invoice was not planned as well.

Screenshot example of invoice file downloaded:



1

Screenshot of settings item functionality – profile picture upload:



Demo Video & Links

- Include a 2-3 minute demo video, showing your interface in action.
 - The easiest way to record this is with a screen capture tool, which also captures audio- such as Quicktime. Use a voiceover to explain your application. Include the name of the project, your name, the project components, and how your application works. You can present it on your webpage or on youtube, but it must be linked on your webpage.

Demo Video located in root folder of github repo.

- Include a link to your source code on github and a link to the publicly hosted application.

Source Repo: <https://jf-userinterfaceproj3.netlify.app/>

Svelte Proj in /UserInterfaceProj3

Public Hosting: <https://github.com/JosephAFerguson/UserInterfaceProj3>