HCI Final Report

Kevin Liu, Maximilian Huber, Sahil Kiran Bodke

This report is a reflection on the design process behind creating the digital interface for our final project: **resu.me**.

Problem:

Updating resumes is often difficult, stressful, and time-consuming. Most of our target users, young professionals and college students, rely on general-purpose text editors like Microsoft Word or Google Docs, which fail to meet the specific needs of this task. As Alan Cooper states in his book, “To create a product that must satisfy a broad audience of users, logic will tell you to make it as broad in functionality as possible to accommodate the most people. Logic is wrong. You will have far greater success designing for one single person.” **[7]**

Our app aims to simplify creating, updating, organizing, and polishing resumes, especially for young professionals who frequently gain new experiences, often lack knowledge of what makes a resume professional, and are applying for competitive jobs. To achieve this, the app must support tasks such as adding experiences, curating them onto a resume, and saving, exporting, and organizing resumes.

Design Process:

In order to best convey how we arrived at our final solution, we will start by providing justifications for our initial design decisions, and then for each iteration of user testing show how our feedback influenced the changes we made to each new prototype.

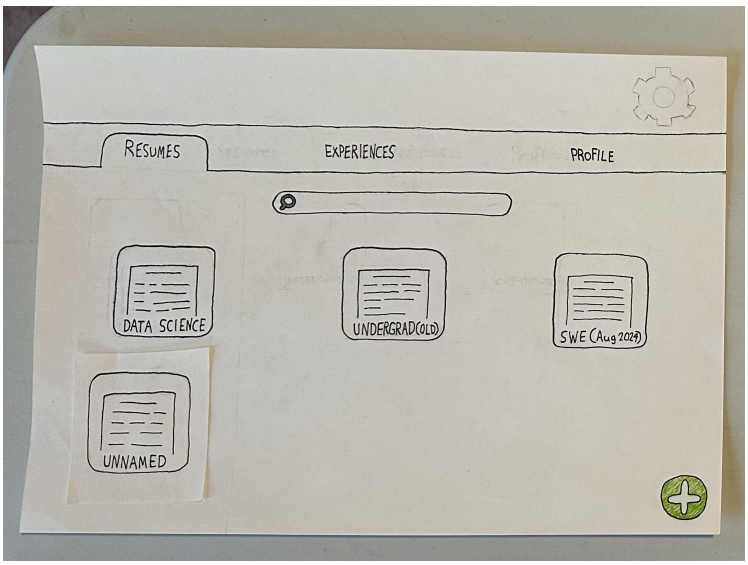

Figure 1. Initial home screen design

Initial Design Decisions:

1. **Modular Approach:** The main idea behind our project, and the main difference between our solution and standard text editors is the modular separation of content. We wanted to create a system that would allow the user to capture their experiences, projects, education, and more without those pieces of content being tethered to any one resume. When the time comes to create a resume, you can simply select which modules of content are most relevant to include, and you have the freedom to remove irrelevant content without permanently deleting anything. We felt that this workflow would make the process of gathering real-world experiences and updating resumes feel more fluid, constructive, and intuitive.
2. **Use of Generative AI:** One thing we noticed in our initial contextual inquiry is that many of our target users rely on generative AI tools like ChatGPT to help them rephrase experiences and provide a layer of professional “polish” to their content. We saw the opportunity that our modular approach provided in allowing us to embed additional tools directly in the workflow, and decided that a generative AI feature designed specifically for rephrasing experience descriptions would provide a great value add for our users.

What we Tested

1. **Create a new experience:** We wanted to test whether our home page and experience creation page was intuitive or not. By testing this, we would gain valuable feedback on the strengths and weaknesses of our homescreen and experience creator.
2. **Edit the experience, and generate a new description with the AI tool:** We wanted to learn whether our AI tool was intuitive or not, as we weren’t completely sure about it going into the user testing phase.
3. **Create a new resume, add 3 experiences to it, and save the resume:** We needed to test our actual resume creation screen and make sure that it was intuitive to users how to add experiences from that page so that our resume creation process would be smooth.

After Testing with Paper Prototype:

Improvements of Existing Features:

1. **AI Feature:** The button to activate the feature was not easy to find, the messaging throughout the feature’s pop-ups was unclear, there was no user control over the generation, and the output was tedious to integrate without deleting the existing experience description. To fix this, we added a text label to the button and separated it from buttons related to formatting, we changed the labeling within the feature’s pop-up, we added a text box for user prompting, and most importantly we switched from generating a text box description to generating modular contributions, so that the newly generated content could be added without overwriting previous content.
2. **Icon Selector:** The Icon Selector did not make intuitive sense. We moved it up to be the first field in the form, to indicate that it is related to identifying the experience as a whole. We also added an info tooltip which describes its purpose.
3. **Experience Cards:** Users had trouble figuring out what the cards are and what they are supposed to represent. This was partially due to the fact that we had given them pre-existing experiences, where an actual new user would not. We added the name of the company, job title, and timespan for each experience card in the gallery.

4. **Experience Selector:** Users were confused by what this pop-up was for, and how to add existing experiences or create a new one. We added labeling to make it clearer, and directly labeled the button for creating a new experience.

Suggestions for New Features:

1. **Sorting/Filtering:** While the resume and experience galleries contained a search bar, our users requested more options for filtering and sorting these galleries. We added a button to the cards to allow users to “favorite” them (pushing them to the top of the respective gallery), and made plans to implement a proper tagging and filtering system in future iterations.

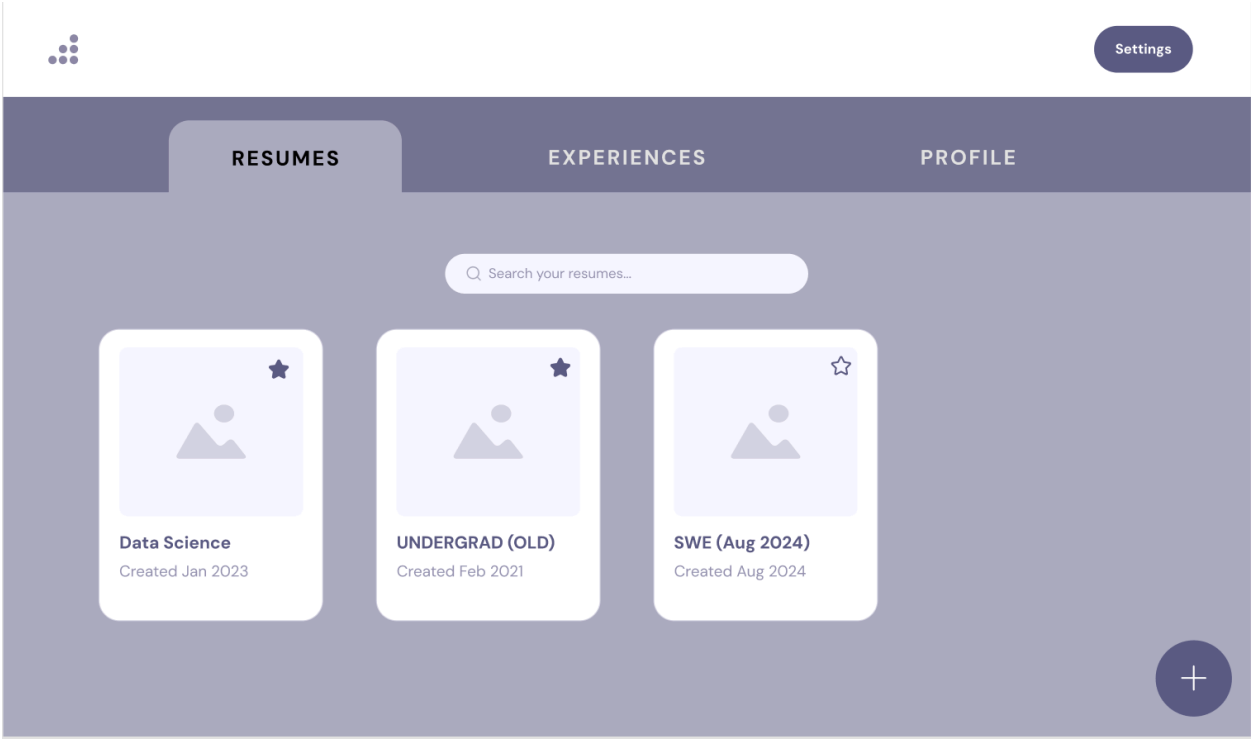

Figure 2. Initial Figma prototype home screen. You can see that the resumes have stars in the top right to indicate whether they are favorited or not, and a big plus in the bottom right that stands out so users have an easier time noticing it.

After Testing with Figma Prototype:

Improvements of Existing Features:

1. **AI Feature:** The button was still a bit hard to locate, users were unsure of what the prompt was for, and the labeling within the feature’s pop-ups was still confusing. We changed the button’s color to make it stand out, added a placeholder prompt to guide the user, and further refined the messaging within the pop-ups to explain at each step what the user is meant to do.
2. **Add Buttons:** Our users complained that the add/create buttons did not really stand out that much. We changed all of these buttons throughout the app to make them a consistent bright teal color. Whenever an add button appears in the interface, the color contrast provides a unique visual property that helps the button “pop-out” and naturally draw the user’s eye **[6]**.
3. **Dashboard Tabs:** Some of the users were confused that Resumes and Experiences were side-by-side in the Dashboard menu, given that experiences are one of many sections on any individual resume. To better convey the content hierarchy of the

interface, we switched from a horizontal menu to a vertical one, with a clear separation between the Resumes and other various content tabs. In this case, there was a feeling of incongruence between the user's *conceptual* model and the interface's *manifest* model. By making this change, the manifest model now more closely resembles the user's manifest model of the content hierarchy. **[9]**

Suggestions for New Features:

1. **Sorting/Filtering:** Following the peer evaluation, we mentioned to our users that we were planning to add a system for filtering and sorting existing content. One user mentioned that it would be useful to be able to sort experiences by setting a time span.
2. **Non-Empty New Resumes:** One user pain point that we identified consistently was that users felt that starting a new resume should not involve starting over from scratch. If they were updating a resume, there should be a way to add their new experiences without necessarily losing the old resume. To allow for this workflow, we added a duplicate button to the resume editor. We also received feedback that it would be nice to have a feature that allows users to create a new resume automatically just by pasting in a job description and having the system gather their relevant experiences and projects for them. The duplicate resume button was a quick solution to this problem, but in the future we would also like to explore the Automatic Resume Creation feature.

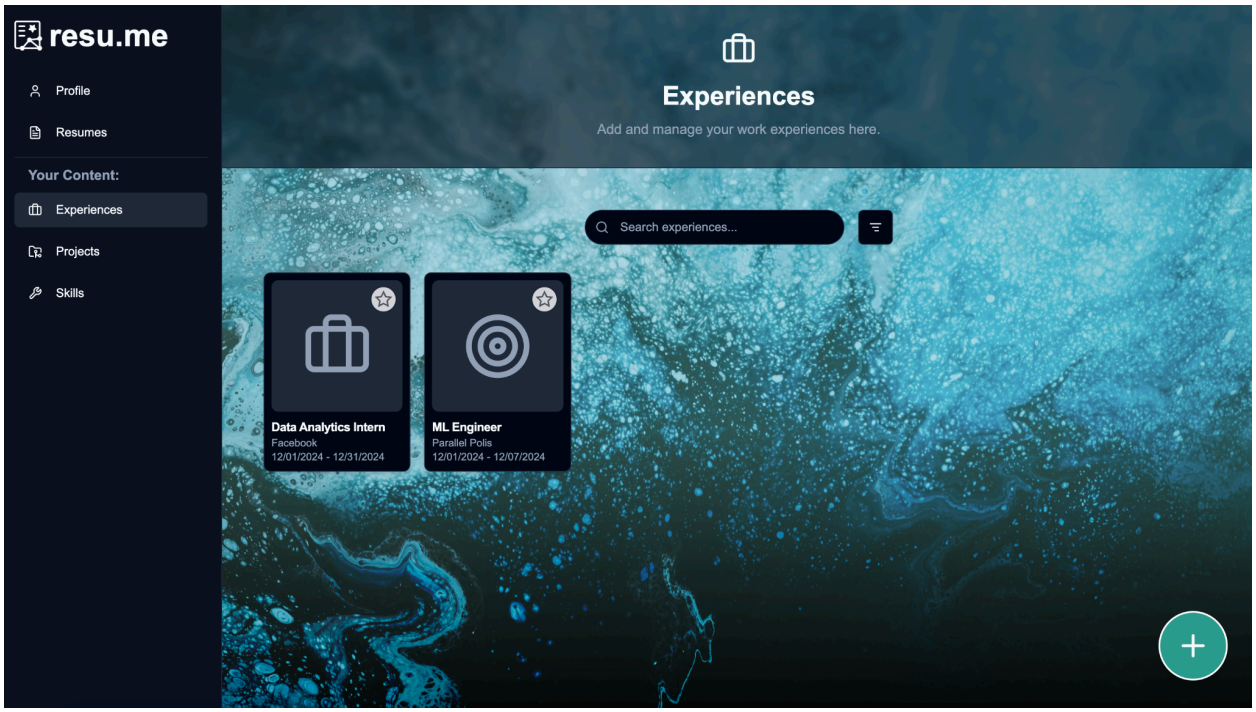

Figure 3. Our final home page. We moved the startup screen to the experiences, since they are our core focus. The add button has been given additional clarity, and we reorganized our navigation bar to be more in line with what the user would expect from a resume editor.

Final Design:

Our final interface prototype features a dashboard view and a resume editor view that is overall quite similar to the original design. We still have our initial design decisions present, though their execution has been slightly altered. The horizontal tabs, which house the modular content galleries, going across the top of the dashboard page are now a vertical menu on the side. The generative AI tool is still integrated into the app, and its function is still to help you polish your experiences, but the whole interaction flow and messaging of it has been altered, motivated directly by our user feedback. We have made many small improvements to various interactions

throughout the design, but the feedback we received throughout our testing about our modular system and generative AI integration were so positive that we can confidently say those were good initial design choices. Most of our unfinished feedback is in the form of suggestions for new features which can elevate the usability of our interface even further.

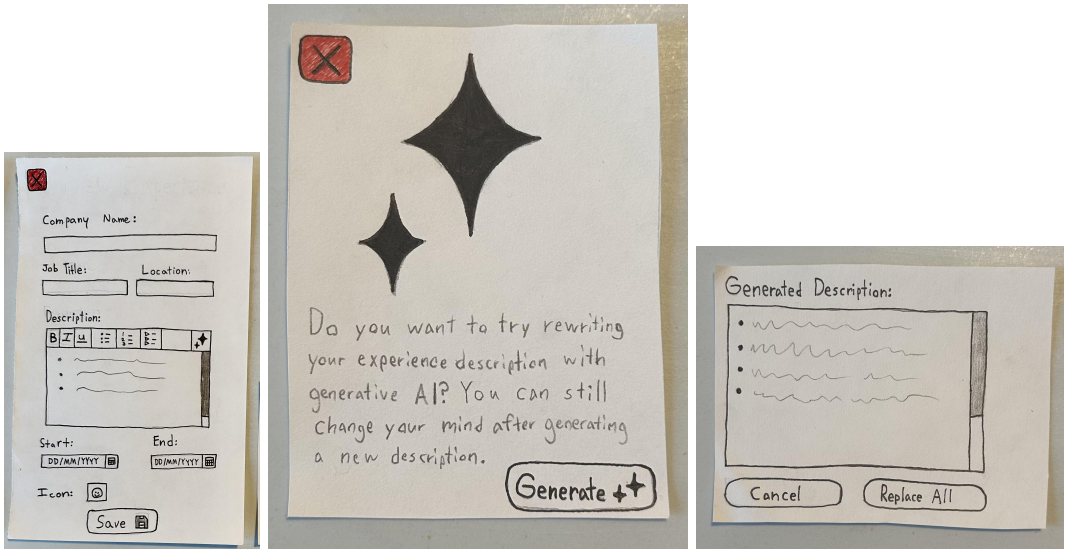

Figure 4. Our first generative AI prototype on paper.

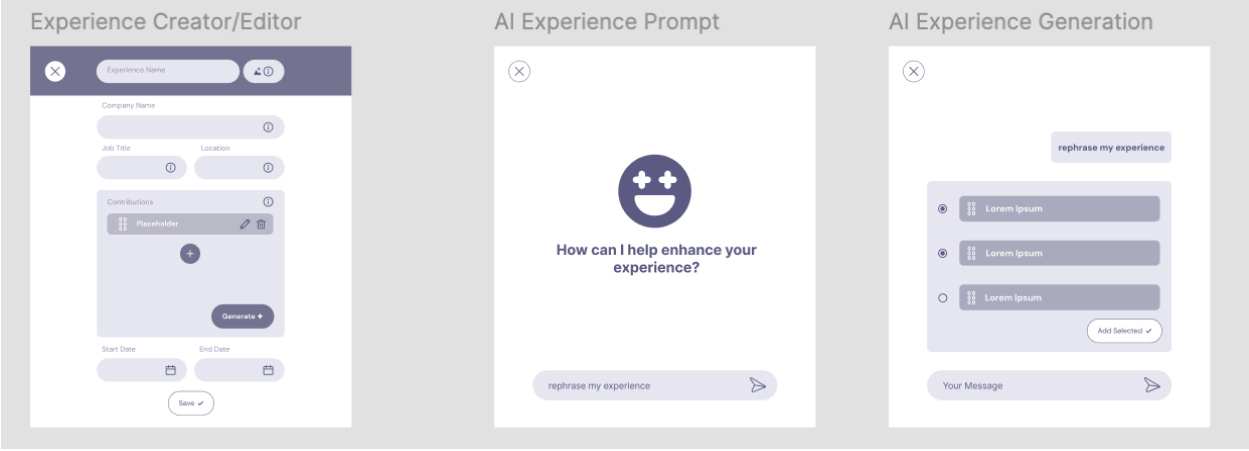

Figure 5. Our first iteration on the AI prototype. Notice the moved generate button and the added color, as well as the new generated part being modular instead of just a bullet point in a text box.

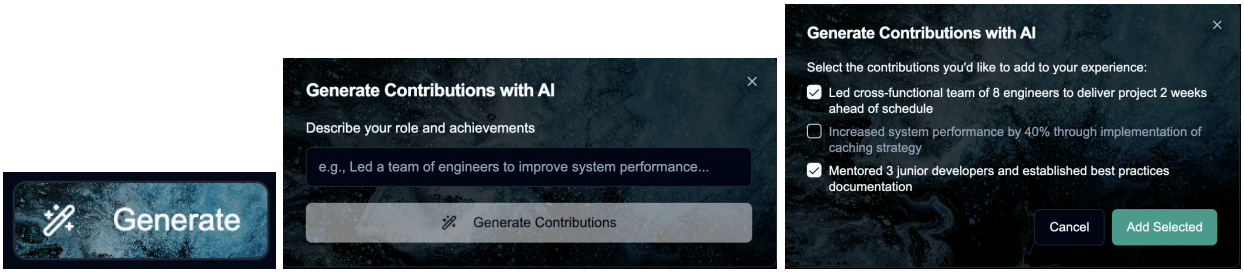

Figure 6. Our final prototype. The button now stands out even more, and the creation process has been simplified and shortened to look nicer and be easier to use.

Implementation:

Our third and final prototype is implemented in TypeScript, using NextJS **[1]** for routing and basic components, TailwindCSS **[2]** for styling, and schadcn/ui **[3]** for creating and implementing additional components. The backend uses Firebase **[4]** for user authentication and storing user data. The application is hosted using Vercel **[5]**.

While our implementation did not negatively impact interface usability, the process of testing a functional prototype highlighted previously unconsidered usability issues. For example, we had not designed any way to delete experiences, as our user tasks did not require it and no users mentioned its absence. To address this, we added a delete button to the Experience Form for editing existing resumes. The app also asks users to confirm before deleting an experience.

Some users had an issue with the navigation bar being side by side since it didn't feel natural to them. One design choice we made was to switch our navigation bar from a row at the top to a column on the left side. This helps illustrate the idea that resumes are sort of a hierarchy and having this layout helps users understand that experiences, projects, and more all fall under the resume. It also helps align the user's conceptual model with our manifest model. **[9]**

Another impactful design decision we made was switching experience descriptions from text fields to a list of contributions. While adjusting the web page components for this change was straightforward, modifying underlying database schema and functions was much more time-consuming. Since database schemas are typically defined early in development, confidence in data structure and type is crucial. This really showed us the value of our user testing and prototyping. This also aligns with our class material **[8]**, which shows that *prototyping* validates designs early, when iteration is cheap, while *user testing* confirms and provides conclusivity for design decisions.

Reflection:

If we were to do this project again, we would change a few things about how we prototyped and how we evaluated the results from the user testing. One thing that could have helped if we had more time was a contextual inquiry. From this, we could learn a little more about the frustrations that other people were having with creating resumes and have some more features in mind to target when creating our prototype. The generative AI idea came from a contextual inquiry that one of our group members, Max, did in his previous assignment. If we were able to conduct a contextual inquiry once we had a solid idea about what we wanted to do, we could have learned a lot about what others are going through during this process.

Another thing we could have done was add more user tasks. Some of the features that we chose were tested, but we weren't able to test all of the features that we wanted to. We were able to learn a lot about the experience editor and the AI tool, but some features such as projects, resume headers, and favoriting resumes went untested so we had no real feedback about those. Having more questions ready to ask post-testing would have been greatly helpful as well. We definitely learned the most from our interviews, as our actual tasks didn't offer as much. We found that people were more eager to talk about their thoughts afterwards and had a lot of them to share with us. Maybe we could have asked them some more questions during the actual user testing process, and that would alleviate some of the post-testing questions.

Future Steps:

If we develop any additional features, we would definitely prefer to test them using a lower-fidelity prototype first before integrating them into the functional prototype. Having the functional prototype gives us multiple levels of interactable prototypes, which would accelerate our design and iteration process moving forward. Given more time and resources to continue testing and developing our interface, here are some additional ideas we would like to consider:

1. **Missing Core Features:** While we have some new experimental features in mind, some core functionalities are still missing. Users need to export resumes, control formatting, and they need to add education, skills, and projects to their resumes. These content categories can be modularized like experiences, but this may introduce new design challenges requiring iteration and user testing. Though seemingly straightforward, even simple interactions often pose unexpected design or implementation challenges.
2. **Sorting/Filtering:** Test users frequently requested ways of sorting and filtering their collection of experiences and resumes. We would definitely want to test out different approaches to this with a lower fidelity prototype, to see what feels most intuitive for the user and identify any design challenges. We would like to support creating custom tags which are applied to resumes and experiences, as well as searching your list of experiences by setting a time span. We think this is an obvious feature to add, but we want to make sure we do it in a way that maximizes its usability.
3. **Automatic Resume Creation:** A standout value of our modular system lies in opportunities to build integrated tools like the AI Contributions feature. One idea we would love to explore is automatically curating existing experiences to generate a resume tailored to a specific job description. This feature could save users significant time and could even function as a browser extension.
4. **Contribution Groups:** The switch from text-based descriptions to modular contributions opens new possibilities. Based on TA feedback, allowing the same experience to appear on different resumes with unique contributions would be valuable for young professionals, as they often need to highlight diverse skills from the same experiences. A feature to group related contributions and enable or disable entire groups could make experiences more versatile and practical for users, allowing the same experience to highlight different aspects of a user's skill set depending on the resume it is used on.

References:

1. **NextJS - https://nextjs.org**
2. **TailwindCSS - https://tailwindcss.com**
3. **shadcn/ui - https://ui.shadcn.com**
4. **Firebase - https://firebase.google.com**
5. **Vercel - https://vercel.com/home**
6. **HCI Lecture Week 2.2 - Understanding People: Perception and Cognition**
7. **About Face: The Essentials of Interaction Design, Alan Cooper**
8. **HCI Lecture Week 9.1 - User Testing**
9. **HCI Lecture Week 3.1 - People and Technology**