

Duck-Libs

Final Documentation

Eliza H, Kevin O, Olivia O, Sarah S

Background and User Research, Part 1

The CSE community is innovative, dynamic, and accomplished, but at times can feel polarizing, intimidating, and isolating. Only 16% of students who take introductory level CSE classes are admitted to the department, leading to feelings of pressure, self-doubt, and competition (UW CSE Department). Our user research uncovered a central theme that finding one's place in the CSE community is often initially challenging and tension can exist among the competition for internships and jobs. One interviewee mentioned, "sometimes [CSE] is frustrating but it makes me feel better knowing I'm not the only one." Another interviewee discussed how, "[it] feels very competitive both before and after. When you're in the major everyone is name dropping internships and you feel a lot of pressure to get a good job. There is unspoken tension". Conversations about failure and vulnerability are often avoided, and many undergraduate students mentioned that it can be difficult to form a community and reach out and connect with others. This led to our central question, "how might we foster a sense of belonging in the CSE undergraduate community by connecting people based on shared experiences?"

With the competitive nature of the major, there is a greater gap created between the community with the addition of more building space. We surveyed 36 undergraduate CSE students about their sense of community in Allen compared to Gates and found that on average, participants ranked Allen's strength of community higher than that of Gates. Students described Allen as feeling "like home" and "more talkative and familiar". Allen was often noted as a place people would go to work for long periods of time, or grab food from Snack Overflow. Having the social Snack Overflow room "helped to build a sense of community with everyone hanging out" as noted by a CSE undergraduate student. People would leave game controllers or playful items in the room, and it helped to create an atmosphere where people could take a break and unwind. Although Gates has more open spaces, students said that areas still often felt exclusive. There are no set social spaces in Gates that connect the community around one centralized point. From fly-on-the-wall observations in the Gates Undergraduate Labs we found that students were using whiteboard spaces to write CSE jokes, draw pictures, and share words of encouragement. This inspired us to explore community displays that could create more of an inviting and personalized environment in the Gates Undergraduate Labs. We experimented with different empathy-building topics during our rounds of low-fidelity-prototyping and user testing and realized that students cared most about connecting over meaningful prompts. They enjoyed reading qualitative responses over quantitative "yes" or "no" answers. People appreciated reading responses that expressed shared struggles and made them feel like they weren't alone. Undergraduate CSE students mentioned being familiar with the concept of "rubber duck debugging" where they talk to a duck to help debug problems in their code. We tested the idea of using the rubber duck metaphor in our

design to focus on “debugging challenges in life”, and discovered that the playful idea of the duck made people more comfortable engaging with reflective prompts. Rooting the idea around a well-known CSE concept also helped to strengthen the feeling of community within the major, and provided a foundation for our overall design.

Final Design, Part 2

The purpose of Duck-Libs is to provide a platform for the CSE undergraduate community to reflect, share advice, and “de-bug” common challenges and experiences. It adds a personalized community activity to the Gates Undergraduate Labs, and creates a more connected and inviting environment. Students may talk to a duck to debug problems in their code, and now they can debug situations in their lives by reflecting on past and present experiences. The platform resembles “Mad-Libs”, a guided fill-in-the-blank children’s story activity, and utilizes ludic design to encourage playful reflection. When students approach the main podium they can begin interacting with the Duck-Libs interface which walks them through a series of prompts, asking them questions about their own life. Once they finish the prompts, they receive a personalized, printed story about their life. The story has an encouraging tone, and highlights that although everyone has different experiences, they are all still united as a CSE community and don’t have to “debug” life’s challenges alone. Students can take the story with them, and it serves as a tangible artifact to remind them of their journey and place in the CSE community.

The second piece of the installation is a projection that displays individual past failures experienced by members of the CSE community. After students fill-out their responses to the Duck-Libs prompts, they have the option to anonymously submit their past failures to the community wall. Our research emphasized that people wanted to know that they are not alone in their struggles, and by highlighting failures it banishes the shame behind these experiences and creates understanding in the community. There is a board located next to the projection displaying failures, and after students have received their own story, they also have the option of getting a blank receipt that they can use to write and pin-up advice or encouragement to their fellow CSE peers. These three main components create a full circle interaction where users can receive a personal story, view shared community struggles, and give and receive encouragement. The installation serves to remind students that they are not alone in experiencing pressure and stress. Based on proxemics, there are different levels of engagement students can have with the installation. From a further distance the large communal projection of failures provides groups of people the ability to interact with the installation, even when quickly passing by. The bright yellow podium acts as a playful pop of color in the Undergrad Labs, sparking delight and encouraging people to approach the podium for the personalized portion of the interaction. The yellow board with hand-written pieces of advice is noticeable from a distance, but students will need to be close to the piece to read and engage with the responses. This element of the interaction was designed to be a more intimate experience, and encourage students to read the pieces together at a closer physical distance. The hand-written element of the advice was inspired by the fly-on-the-wall observations of the Gates Undergrad Labs, and aims to make the new lab space seem more inviting and personalized. Visuals for the installation are based upon the rubber duck debugging metaphor and the main inter-

face includes illustrations of circuit ducks, providing a playful form for people to engage with the Duck-Libs activity that is related to a core concept in the CSE community.

In terms of the technical components, the installation is comprised of three main interconnected subsystems and details can be viewed in the UML Diagram. In the first subsystem students use a Microsoft Surface to input their responses to the Duck-Libs prompts on a web browser, and these responses are recorded and stored to later generate their printed story. The website is hosted on a GitHub server and was built using HTML and CSS. Javascript is utilized to receive the user input and store the sequence of responses in an array. The website contains the hard-coded Duck-Lib story, and Javascript is used to take the individual responses stored in the array to fill-in the story. The story prints via a USB connection to a thermal receipt printer, which is the second subsystem. The third subsystem is the second website which contains community responses to the question about past failures and is projected on the wall. Users can agree on the Duck-Libs site to have their response to the failure question be anonymously shared on the community projection. If they agree, TableTop API is used to fetch their response which is stored in Google Sheets and has been filtered for swear words, and Javascript is used to display it. The browser displaying community failures will auto refresh and repopulate with new responses each time responses appear 150 times.

Design Reflection, Part 3

The biggest learning our team experienced was how to scope a project in a challenging problem space, iterate, and pivot with feasibility limitations in mind. Our user research and affinity diagramming led us to focus on the initial design question, “how might we facilitate organic social interactions in the Gates and Allen buildings?” The original proposal and low fidelity prototype incorporated the concept of rubber duck debugging by having people exchange physical ducks with QR codes linked to surveys about CSE community topics to encourage movement, interaction, and conversation between the two buildings. While the aim of the design was to facilitate human interaction and build community in-person, our first user test revealed the concept actually drove users towards isolated interactions with their phones. Testers also expressed they would not be motivated enough to share the duck just to fill out a form. We received positive feedback on the playful appeal of rubber duck debugging and learned that users strongly preferred the reflective and vulnerable sentiment of the questions. These insights from the user test and critique from the teaching staff motivated us to reframe our design platform to more effectively engage users at a physical installation. We returned to our user research and reshaped our design question to “how might we foster a sense of belonging in the CSE undergraduate community by connecting people based on shared experiences?” After another round of ideation, we discarded the QR survey concept and transitioned to the more playful and interactive mad-lib podium concept. This pivot was initially daunting, especially because we were moving forward with a completely new concept without feedback, but the revised design addressed the previous issue of isolating interactions and had a more concrete rationale for fostering connections in the community. The previous prototype’s functionality had been mostly wizard-of-oz, so the majority of the code for transferring responses to a community page was preserved. At this stage in our timeline, we developed

the code and design specifications for the new design and created a strategy for rapidly getting materials to build the new platform. Our design and code specifications drastically differed from our initial concept, but revising so early allowed the final prototype to align very closely with our UML diagrams and architectural design with a few minor additions.

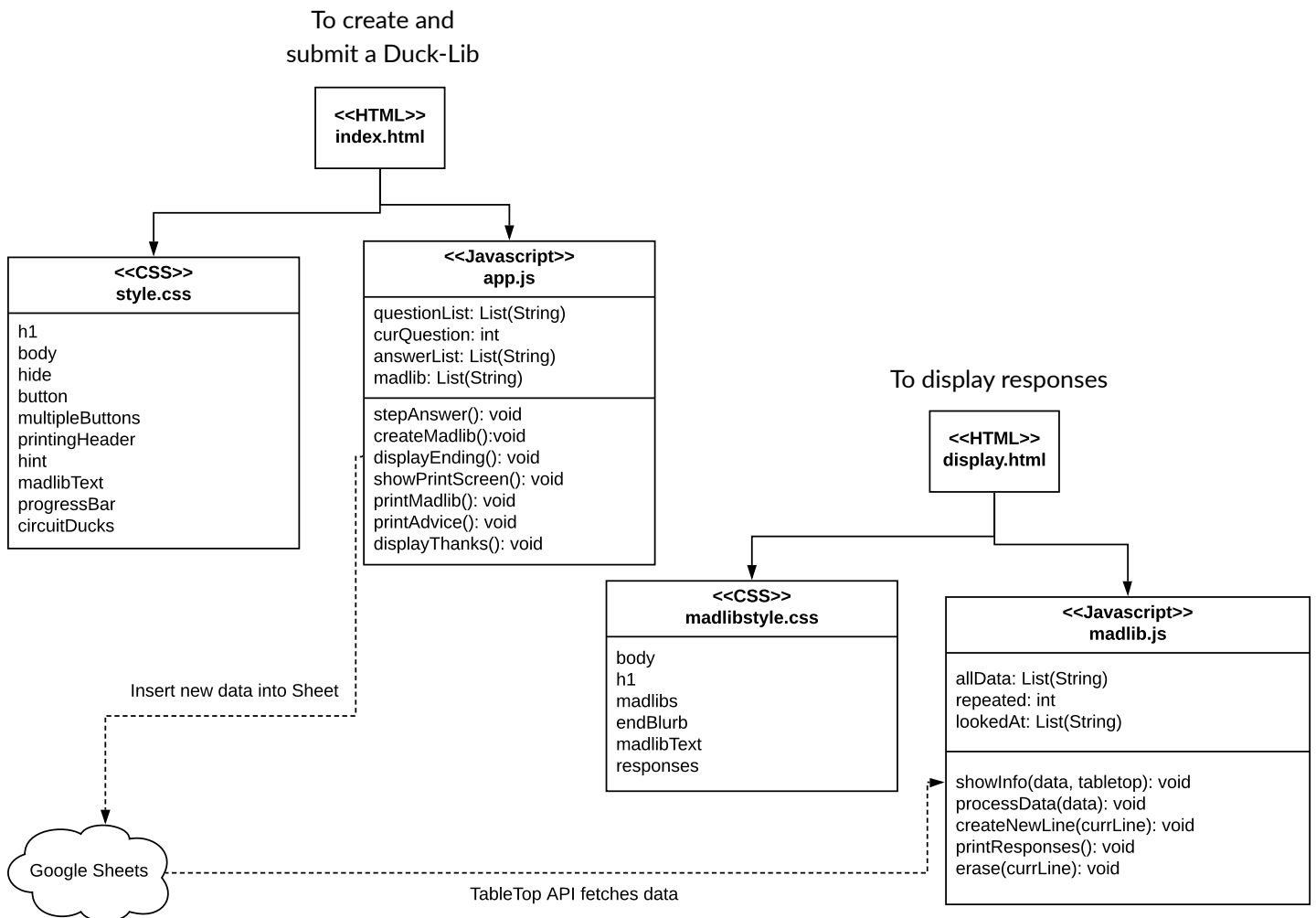
An unanticipated technical hurdle was getting the receipt printer to properly function. We initially planned to use an Arduino and thermal receipt printer, but budget and time constraints required us to find an alternative method for printing that required less code implementation. Our solution was a thermal receipt printer connected to our Duck-Libs website via USB, and were able to alter the settings of the printer to properly format the content of the story. We also faced the challenge that the print dialog flashes on the screen for a few seconds before printing, which can be confusing for the user. With constraints from using a web browser to print, we realized there was no way to disable the print dialog from flashing. After the pivot at milestone two, we closely adhered to our timeline to ensure we caught back up to speed and could get ample feedback on the new design direction. Eliza and Sarah mainly took on developing the UI and graphics and implementing the code, while Kevin and Olivia focused on rewriting copy and building the physical podium. We bumped into manageable hiccups including revising our visual language to better communicate to the target audience of CSE undergrad and troubleshooting the form factor of the podium. As the visuals were implemented, we updated our UML diagram to include more graphics placement functions. There were also considerations we had to make about being respectful in displaying past failures. We added a screen in our interface to get consent before displaying their response, built in extra precautions to filter out swear words before projecting them on the wall, and created a poster that asks people to respect the community space.

The main change we made to our original design specification was the addition of the advice board. After receiving feedback on our medium-fidelity prototype, we realized we needed a component to connect the podium with the communal response wall and complete the interaction with a forward-thinking message. The podium interaction is designed to inspire introspection and reflection on personal identity within CSE, and the response display serves as a unifying reminder that the CSE community is not alone in their failures. We were missing the opportunity to foster connections within the community by providing a call to action to share words of encouragement in response to the first two elements of the installation. This drove us to add an option to print out a blank slip of receipt paper to write and post advice to the board, creating a wholesome interaction that not only acknowledges the failures but also unifies the community on a positive note.

Overall, the rebound our team made from the critique of our initial concept encouraged reflection on the design rationale and code feasibility, improved our problem-solving abilities, and resulted in a stronger project.

Code Specifications

Duck-Libs UML Diagram



GitHub Repository

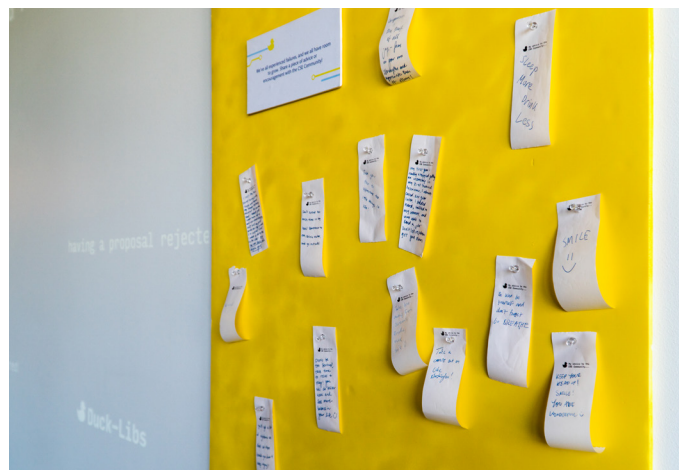
<https://github.com/ducklibs/ducklibs.github.io>



Duck-Libs installation with the interactive podium, community display, encouragement board, and plaque.



Interactive podium featuring the introduction to the Duck-Libs activity.



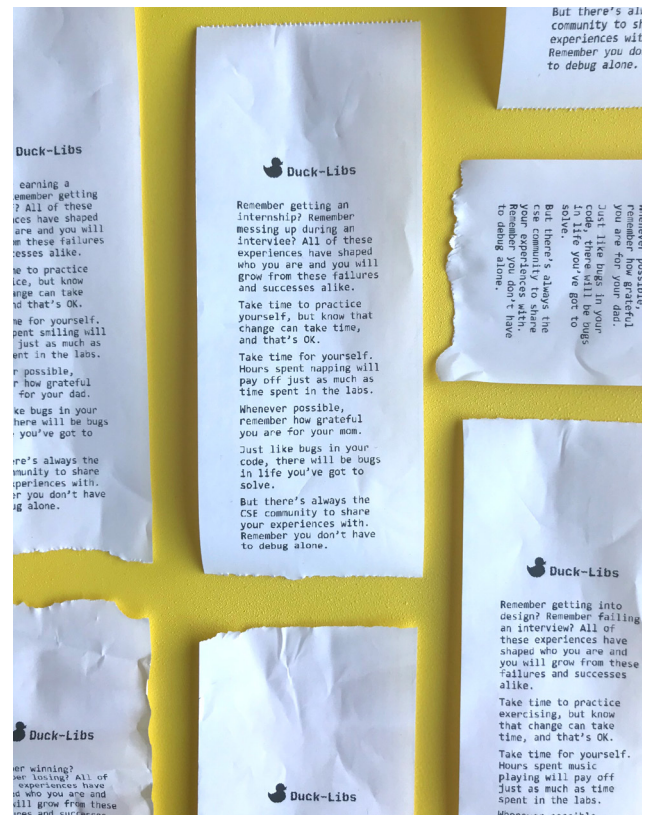
Board where CSE students can leave each other pieces of advice and encouragement.



People interacting with the Duck-Libs activity and writing advice on the board.



Person filling out the Duck-Libs activity.



Duck-Libs stories that print after the activity.