University of Nebraska - Lincoln

DigitalCommons@University of Nebraska - Lincoln

5-2024

Scoular Feed Marketplace

Prema Vasudevan
University of Nebraska - Lincoln

Mitra Vajjala
University of Nebraska-Lincoln

Natalia Trejo Andalon
University of Nebraska - Lincoln

Ealynn Hsu
University of Nebraska - Lincoln

Malcolm Saltzman
University of Nebraska - Lincoln

Follow this and additional works at: https://digitalcommons.unl.edu/honorsembargoed

Scoular Feed Marketplace

Honors Thesis | Design Studio 2023 – 2024

Prema Vasudevan, Mitra Vajjala, Ealynn Hsu, Malcolm Saltzman, Natalia Trejo Andalon

Executive Summary

Project Description

The Scoular Company provides agricultural supply chain solutions to customers around the world. Scoular buys, sells, stores, handles, and processes animal feed, while also managing transportation and logistics for farmers and ranchers worldwide.

Scoular Feed Marketplace aims to provide a unified platform on which customers and employees at Scoular (aka Scoular Contacts) can have a clear view of customer delivery schedules. Moreover, the goal is to provide a seamless way for customers to inform Scoular of their desired feed delivery schedules and request individual load and recurring delivery schedule changes.

Stakeholders

- David Tomlinson, **CIO**
- Chris Gervasini, **Director - Application Delivery**
- Ken Dworak, **Director – Business Enablement & CRM**
- Branden Folino-Haye, **Director – Data Integration, Analytics, Automation**
- Christi Klemme, **Business Relationship Manager**
- Andy Hohwieler, **Regional Manager & Product Group Manager**
- Josh Kasprzyk, **Regional Manager**
- Tyler Rogers, **Sales Representative**
- Jamie Dingley, **Lead Merchandiser**
- Matt Bradford, **Senior Merchant**

Problem

Customers work with Scoular to establish a recurring schedule to deliver feed to their farms. Customers often will want to make one-off changes to their scheduled deliveries or make recurring updates to when they want feed delivered to their farms. Currently, customers keep track of their feed delivery schedules themselves, and rely on phone calls, emails, and text messages to request changes to their feed deliveries from Scoular. Scoular employees receive these requests from dozens of customers and have no consistent way of tracking different customer schedules and changes they have requested.

Vision

The Design Studio team thus worked towards the following vision:

Empower Scoular customers by increasing delivery schedule visibility and simplifying the order editing process through an intuitive web application.

Opportunities

Via collaboration with the stakeholders listed above, the team identified the following project opportunities:

Increase feed schedule visibility for customers. Currently customers must manage their own feed schedules using a method of their choice. They are solely responsible for keeping this schedule up to date with any changes they request from Scoular. The goal of Feed Marketplace's schedule page is to make it so that customers no longer must manage their own order schedules. Their schedules on the application will automatically update when change requests are approved or denied by Scoular Contacts.

Simplify the order editing process for customers. Right now, when customers want to request a change to their feed order schedule, they contact their Scoular contact using a myriad of different methods such as phone calls, emails and text messages. By implementing the order edit request workflow, the goal is to simplify this process for customers and give them a single platform on which to reference their requested changes and the status of their changes.

Centralize communication between Scoular Contacts and customers. As previously mentioned, when customers want to request changes to their schedules, Scoular Contacts must field communication via a variety of channels. By creating the request approval/denial workflow, the Scoular Contact can respond to the customer's requested changes directly on Feed Marketplace, reducing the need for them to log and maintain these changes elsewhere, as well as preventing them from having to contact the customer separately.
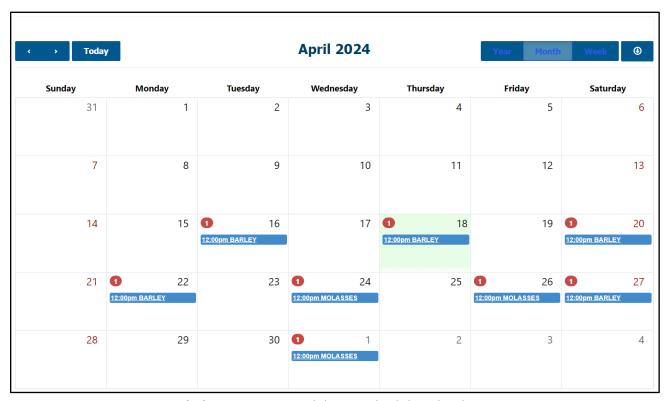
Solution

To leverage the above opportunities and execute our project vision, the Design Studio team created the Scoular Feed Marketplace, an application with a separate view for customers and Scoular Contacts
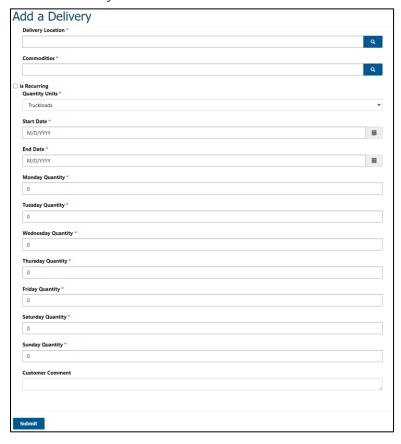
Customer Features

Delivery Schedule

The delivery schedule is implemented as a calendar view that gives customers a high-level look at the days, weeks, and months ahead. Customers can quickly see if they have upcoming deliveries and determine if additional deliveries must be added, or existing deliveries need to be changed.

Pictured above: Customer delivery schedule calendar view

Add Schedule of Deliveries

Add a Delivery

Delivery Location *

Commodities *

☐ is Recurring
Quantity Units *

Truckloads

Start Date *

M/D/YYYY

End Date *

M/D/YYYY

Monday Quantity *

0

Tuesday Quantity *

0

Wednesday Quantity *

0

Thursday Quantity *

0

Friday Quantity *

0

Saturday Quantity *

0

Sunday Quantity *

0

Customer Comment

Submit

This form allows customers to request additional or new deliveries of feed from Scoular. Deliveries requested can be for weeks, or recurring requests. This replaces the traditional workflow of calling or emailing a Scoular contact to request new delivery schedules.

Pictured above: *Submission form to request new deliveries*
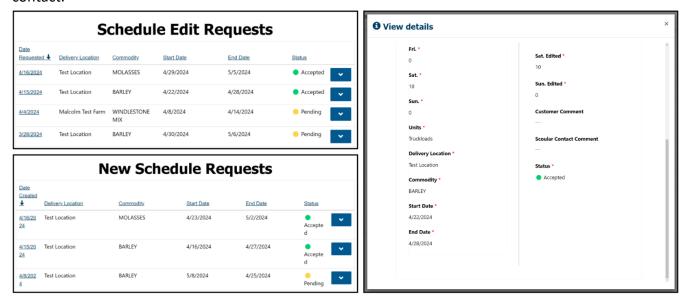
Edit Weekly or Recurring Schedule

These forms allow customers to request changes to weekly or recurring deliveries that are scheduled. Weekly and recurring deliveries are processed independently to ensure that customers are both able to make one-off changes or make modifications to their delivery schedules that persist.

Weekly Load to Edit *
Allows you to view the week that you want to make a request to change values to.

[🔍]

Monday Edited Quantity *

[0   ]

Tuesday Edited Quantity *

[0   ]

Wednesday Edited Quantity *

[0   ]

Thursday Edited Quantity *

[0   ]

Friday Edited Quantity *

[0   ]

Saturday Edited Quantity *

[0   ]

Sunday Edited Quantity *

[0   ]

Customer Comment

[   ]

[**Submit**]

Pictured above: *Submission form to request delivery changes*
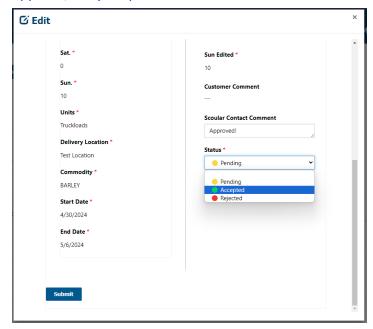
Request Status

Request ledgers allow customers to view the status of changes requested, or additional new deliveries requested. Clicking into the rows of tables gives customers more visibility into the request and allows them to see comments left on a request by themselves or their Scoular contact.

Pictured Above: *Customer Request Ledgers on the left, view of details when record is clicked on, on the right.*

Scoular Contact Features

Approve/Deny Requests

Like the customer, the Scoular Contact has request ledgers as seen in the figure above. The main difference is that the Scoular Contact can leave a comment of their own and either approve or deny a request when they click on it to view details. This information will then be reflected in the customer's request ledger(s).
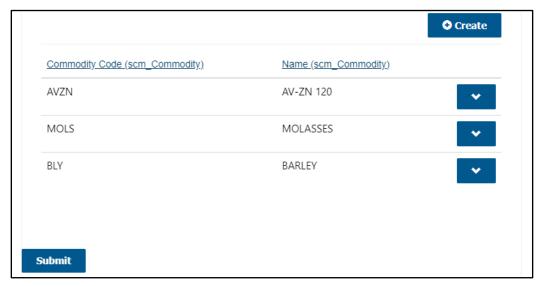
Pictured Above: *Scoular Contact view of request
details with the ability to approve or deny request.*

Limit Customer Commodities

Scoular Contacts can limit the commodities a customer can request schedules for using this page. When a customer is clicked on, the Scoular Contact sees a list of the commodities that the customer can currently access. If they click the create button on the left, they can add another commodity for the customer to interact with. The entries in this table can also be deleted by the Scoular Contact by clicking on the arrow on the right of each entry and pressing delete. This will restrict the customer from interacting with that specific commodity.

Pictured Above: Customer table that Scoular Contacts can use to limit customer commodities

Pictured Above: View of available commodities for a customer when they are clicked on.

Technical Implementation

Power Pages

The Design Studio team, in collaboration with the stakeholders at Scoular, decided to implement the application using Microsoft's low-code website creation platform Power Pages. Using Power Pages allowed the team to streamline aspects of the development process as Microsoft handles authentication and deployment of the platform itself. The team prioritized the main workflows of the application, leveraging Power Page's ability to generate basic table views and forms in minutes to create a minimum viable product.

Power Automate

The team used Power Automate to automate data workflows in the Power Pages application. Flows in Power Automate were used to update data tables based on user modifications on the front-end of the application. These flows essentially act as the "back-end" of our application, powering the background processes that display accurate data to the users.

Next Steps and Expansion

The goal of this project was to provide Scoular with a minimum viable product (MVP) to roll out to customers. This MVP is intended to gauge customer engagement with digitizing the feed order scheduling process. Below are some recommendations to implement before the MVP is provided to users, as well as some recommendations to enhance the overall user experience of the application.

Recommended Implementations for MVP:

1. **Implement User Invitations.** Feed Marketplace is currently not open to the public and thus allows open registration. Before the product is opened up to customers, a Power Automate flow for user invitations should be created. This flow will fill out information needed to onboard a user and email them a code they can use to register with the application.
2. **Live Data Integration**. Currently, Feed Marketplace relies solely on data input by the user, as well as mock data that was manually fed into the system. Before this product is rolled out, it will need to be connected to Scoular's live data systems (CTRM) so that Contract, Customer, and delivery location information is up to date across the platforms.

Other recommendations:

3. **Aesthetic Improvements**. When this platform was under development, functionality was prioritized over aesthetics. As a result, the UI design of the platform is barebones.

We recommend writing custom CSS and conducting some user interviews to improve the overall user experience of the application.

4. **Connect Schedule Editing to Calendar View.** Customers must navigate to a different page and choose the week of deliveries they want to edit from a table. We recommend allowing users to click on the daily entry for the weekly delivery they want to change and open a form that allows them to request edits to the corresponding weekly load. Similar functionality has already been implemented by the team on the Customer Commodities page for the Scoular Contact.

Project Retrospective

Overview

The following section serves as a retrospective for the 2023-2024 Scoular Design Studio project. Over the past academic year, Prema Vasudevan, Mitra Vajjala, Ealynn Hsu, Malcolm Saltzman and Natalia Trejo Andalon have worked closely with Scoular to deliver a MVP solution to increase delivery schedule visibility and simply the order editing process for customers of Scoular's animal feed business. This document outlines the team's successes and challenges, and it explores potential recommended changes we would make if the project was starting over again. At the end of the document, an evaluation of the project and Scoular as a sponsor has been provided as well.

Retrospective

What went well (Successes):

Project Rollout: The developers really appreciated the opportunity to get to know the project and sponsor contacts the Project Rollout that takes place at the Union in August. It was a great way to understand what each project entails, as well as understanding sponsor expectations at a high level. While it does not provide insight into changes that might occur later in the project, it gives developers a good baseline for what to expect from each project.

Improvement from release to release: From release one, to release six the team felt a market improvement in our own performance as we got more comfortable with release structure, communication, etc. Early on it was difficult to gauge if we were hitting the mark with our sponsor, and as time went on, we got better feedback that indicated that our release meetings were meeting their expectations. Outside of meeting sponsor expectations, we received helpful feedback from AL and PL on the structure of formal presentations, and how to improve presentation skills. The team started to set aside significant time before release meetings to practice the presentations which was helpful.

Consistent Meetings with Design Studio Staff: The team meets with our Academic Leads and Program Lead from Design Studio and our Industry Coach every week. These check-ins gave us a lot of objective feedback and helped ground us when the project felt difficult or challenging. All our Design Studio connections provided us with great advice and the consistent meetings made the staff and our coach feel more approachable, leading to candid conversations.

Relationship with our coach: Our coach, Brian Zimmer, was instrumentally helpful throughout the project. Particularly during R1-R3 we had issues with sponsor communication and didn't know how to effectively plan or roadmap our project. Brian was very helpful in helping us navigate communication challenges and his insights from working in the consulting industry clarified our relationship with the sponsors. Brian's advice on effective project and product management was very helpful to the team PM.

Team Culture: Our team culture, simply put, was very positive. Everyone on the team made a concerted effort to engage with one another and understand how to work effectively as a team. When we were faced with challenges with our technology stack, the team banded together and maintained a positive attitude. Everyone on the team was willing to step into roles that were new, difficult, and uncomfortable at times which made a huge positive impact on overall morale and effectiveness as a team. We enjoyed going on team walks, coffee runs, and working together outside of Kauffman to keep things upbeat!

Delivering a functional product: In the preliminary stages we doubted as a team if we would be able to provide more than proof of concept for our sponsors, and we had significant reservations about developing in Microsoft Power Pages. Though our reservations about Power Pages remain, we have been able to create a fully functional platform that supports two user groups and is ready for beta testing for the sponsors. This is a huge win for the team that did not always feel achievable!

What didn't go well (Challenges):

Data access from sponsor: The team spent most of our first semester waiting on data from our sponsors which we felt was pivotal in designing our database and determining our scope. However, as scope conversations proceeded, data access was deprioritized. It got to a point where the team had to sit down and design an ER diagram with very little context for what data Scoular had. As we found out later in the semester and in the Spring, a lot of the data we counted on them having did not exist in a centralized location and instead we needed to incorporate data input into our application. By the time we received the data from our sponsors, it was no longer relevant to us as we had already built out the structure of our application without it. Going forward, we feel it is extremely important for sponsors to understand what data they have and communicate that with the team before building projects around unknown/vague ideas.

Pushback to get user interviews and testing: The product we have created at its core is a external customer facing product, but we received consistent pushback in requesting customer interviews. The sponsors did not want to show customers a 'half-baked' idea and get their hopes up with something that would not resemble the final product. This hesitancy from the

sponsor started when we were in the ideation phase, continued once we created high fidelity prototypes, and persisted as we created the actual product that will eventually be put in front of customers. This means that at no point have we been able to hear the perspective from users themselves as to what their needs are and how we can meet them. We developed the entire application based on the perspective of Scoular employees' idea of what their customers need. It is also worth noting that despite a plethora of meeting attendees, we often only got advice or perspective from one or two individuals. Because of this, rather than executing the vision of customer needs, we are executing the vision of one or two people at the company. While their perspective is incredibly valuable and wide reaching, not hearing from any end users at any point goes against design principles we are taught to adhere to.

Sponsor-Student Communication early in the project: Early in the project, there was a misalignment between the level of communication that sponsors expected and what the student team provided. As students we learned how to communicate business requirements, priority conversations, scope conversations, etc. in a way that both worked for our team internally and gave the sponsor enough visibility without them being involved in every process. During these early stages of learning, the sponsors had very particular requests that we were unable to meet, which caused tension in communications between the sponsor contact and the PM and DM. A result of this miscommunication was that the student team was asked to partake in several additional meetings to discuss product requirements, scope, and other project details. As students, we also felt we lacked autonomy in how we went about planning out our project. At the height of our communication tensions, we had to pull in our PL to talk with our sponsor about expectations and communication boundaries. Following this conversation, and once we delivered concrete deliverables to the sponsor in R4, communication became much easier.

What we would do differently (Potential Changes):

More clarity in the Statement of Work: In our experience the statement of work given to us at the start of the project ultimately provided little to no value. The sponsors themselves were not in alignment with what they were looking for in a project. While we expect a certain amount of ambiguity and collaboration between sponsors and students to come to a solution together at the start of the project, the issue is that the sponsors did not know coming into the project how to brief the team about their business or what existing systems and data were at play that would impact our project. For instance, our project deals with scheduling deliveries. Our product does not integrate into existing delivery scheduling tech at Scoular because they were not prepared for that possibility, but, if our product integrated with their technologies, it would likely be a more effective product in the long run.

Rigidity in technology stack: Students came into this project expecting to utilize web development and we instead ended up developing the platform in Microsoft Power Pages, a low code solution. While it is understandable why our sponsor organization encouraged and pushed us towards Power Pages, it does not align with student teams expectations. The issue is that Design Studio projects are a big way for computer science and software engineering students to gain hands on development experience. While our team learned a lot this semester, they did not learn how to code which was a big reason they joined this project initially. Leads also came into the project hoping to utilize past web development experience in managing this project and were unable to since the project was no longer web development. Going forward projects with low code emphasis could be a great fit for business and actuarial science students. Having some form of restriction on sponsors to not make large changes in technology stack and increasing overall transparency before project selection helps protect students to be able to meet their educational goals, while still giving sponsors the benefits in having students create a project for their business.

Review

Project Success

As a team we feel this project was successful in delivering to our sponsors what they were looking for coming into Design Studio. As students we feel that the project was a partial success. We learned many valuable lessons and skills in project management and client communication but did not gain as much technical experience as was desired at the start of the project.

Working with a first-time sponsor, particularly during the first three releases of our project, we faced some challenges coming to a shared understanding with our sponsor about project requirements and communication expectations. Over the course of the year, we learned a lot about over-documenting vs. Under-documenting and how best to communicate expectations with our sponsor. We learned the best ways to mix our communication style with sponsor expectations and when and how to push back on some expectations that didn't align with Design Studio.

We also learned a lot about how to best manage a project with the specific dynamics of our team. We started the project with a rigid Agile format, and over time we came to shared practices that worked best for our team internally. We became more comfortable with backlog management and running weekly sprints throughout the project. We also worked together to foster a positive working environment, knowing when to take breaks as a team. Our positive work culture was a definite success for the team.

When it comes to the technical learnings gained over the semester, some aspects were a success while others were not. In executing this project, the team learned to utilize a low code solution, Microsoft Power Pages. None of us knew anything about this technology coming in. We learned a great deal about how to use Microsoft's low-code tools as well as how to manage a very steep learning curve while also actively developing a solution. We were able to develop a fully functional MVP for the sponsor within 3 release cycles due to a delay in technology decision, which is a huge success for our team.

One area where we had issues was developing the team's software engineering skills. We all came into the project with the expectation of learning full stack web development, including front-end engineering, back-end engineering, database and API structures, etc. based on our sponsor's statement of work. While we can understand why the sponsors shifted to a low-code solution, it ultimately did detract from our technical skill development as a team.

Sponsor fit with Design Studio

This sponsor is a decent fit with Design Studio; however, the team feels that there are opportunities for improvement so that the sponsor becomes a better fit going forward. First, particularly at the start of the project, the sponsors seemed to have high expectations in communication and project management that didn't consider the learning curve of us being students. The team engaged in several additional meetings outside the weekly sponsor meeting and outside of regular team times to try and accommodate their expectations. After a few key miscommunications we had to involve our Project Lead (Jake Koperski) in separate conversations with our Sponsor to adjust expectations. Following that conversation, communication improved greatly, but showed that the sponsor was not coming into the project with expectations that aligned with Design Studio.

Our sponsors also had several stakeholders involved in the project. Early in the project we sometimes had issues where there were so many people involved it was difficult to try to account for everyone's opinion and still make key decisions in a timely manner. At the same time, despite several people being involved on paper and in meetings, often it boiled down to only one- or two people's opinions or thoughts mattering at the end of the day from the sponsor's side. After a while, we had several people in meetings but only one or two people would even respond to questions we had. When suggesting a change for our project team to be a smaller and more agile team, we were met with resistance from our sponsors.

Project fit with Design Studio

Primarily for Computer Science and Software Engineering students, we would not recommend this project as the team worked in low-code environments that do not align with our educational pathways. For students with other majors, such as business majors in Design Studio

or students interested in Agribusiness, projects like this may be a good opportunity to learn real world technical skills outside of coding.

Another aspect to consider with a project fitting with Design Studio is the scope of the project itself. The sponsors and teams goal coming into Design Studio was always to have a functional and usable project at the end of Design Studio. While we have been able to deliver a functional MVP, the product we've created remains entirely unintegrated with Scoular's data and systems and has yet to be put in front of customers for testing. Due to these factors, this project is still a while away from being ready to roll out. Given the short timespan students have to work on projects, and the significant learning curve we experience as students we believe it is best to not necessarily expect projects to be rollout-ready at the end of Design Studio. While it is incredible and important to us that the sponsor is interested in using our work as more than a proof of concept, taking a project like this from the ground up and to production is likely more of a multi-year project to deliver quality in front of customers.

Advice for Future Design Studio Teams

Have a plan when communicating with sponsors. Adjust as needed.

One thing that really helped our team this year is going into weekly sponsor meetings with a clear agenda. We always started the meeting with roundtable updates from everyone on the team to involve every student. We always ended the meeting with a slide about logistics to remind everyone of upcoming meetings and any access or data that we have for the sponsor. In the middle is where our team always had time to ask any questions or engage in short decision-making sessions. Sometimes this time was needed, and other times we had no open questions and had a shorter meeting. This structure allowed our meetings to have direction and helped us recognize when certain topics of discussion warranted a separate discussion or meeting. What worked well for us was having a rigid structure in place, then loosening this structure as we became more familiar with our sponsors and were able to recognize what aspects were helpful and what aspects were unnecessary.

Keep meetings small and focused, within reason.

Decision making and discovery can become difficult when there are 'too many cooks in the kitchen'. Early in our project we dealt with meetings that had 10 people from the sponsor's side and only 2 people participating. Despite only 2 people participating we found ourselves often asking the other 8 for their input, which ultimately didn't lead to any meaningful discovery and made meetings needlessly lengthy and did not conclude with decisions that needed to be made. For instance, if there need to be discussions about product requirements, it can be helpful to exclude purely technical resources to focus the conversation on product requirements without sidetracking into technical implementation. While requirements and

technical implementation often go hand-in-hand, if your sponsor team is split clearly between product and technical, it can be helpful for the DS team to approach those conversations with a similar separation. For our team separating technical implementation into separate conversations helped us be able to fully understand ideal product implementation, and then come to the drawing table only when requirements needed to be changed to fit with how we can implement them.

Be confident in how you work.

While asking for input and feedback is critical, it is also possible to get too much feedback and lose yourself and your work style in the process. Design studio is an opportunity for us as students to learn how we like to work, organize, and operate effectively. Your sponsor organization may have certain norms and procedures that they are used to and even recommend. Take their suggestions into account and adopt them if they add to your organizational structure for your team. Sometimes it's ok to push back on suggestions of *how* to work from external sources. What matters the most is not *how* work gets done or is organized, what is most important is creating a shared understanding with sponsors on *what* the goal is. How *what* gets communicated and executed should adapt to what works best for your team.

Document commitments in writing but keep it simple.

One way to ease communication with your sponsor is to make it as easy as possible to communicate commitments for the project. Sponsors don't need to know every single story in the backlog, but often want to know in broad strokes what the roadmap of work is looking like. In initial scope conversations you'll likely have a lot of information about features, requirements, etc. floating around. We recommend condensing some of this information into a simple list of commitments in a timeline shared with the sponsor. This allows the DS team and sponsor to be on the same page when scope conversations occur. That said, don't ignore the finer details either. Have written documentation in whatever format suits you best that lays out important features and details for you and your sponsor. These documents help when the project scope needs to be revisited and if the implementation of a feature is a long time after the planning period.

Keep open communication with your team, take breaks, and have fun!

We spend a lot of time with our teams, and all that time is working hard towards an end goal. Make sure to keep open lines of communication between all team members so that people are comfortable speaking up about their needs, whether that's taking a brain break or asking for help with a task. Our team had an anonymous feedback form and anonymous Release retrospectives so that anyone could express changes they'd like to see in team time. Though the anonymous form was rarely ever used, its existence indicated a culture of open communication with our team. Also take time to bond with your team however makes sense

for your team. Whether it's short walks after meetings before you must go back to work, team dinner or game night, or just ending team time early every now and then, make sure to have some fun and relax as a team. These moments of pause help foster an open environment for feedback.

Don't be afraid to ask for help.

At the end of the day, Design Studio is a learning experience for everyone involved, don't be afraid to ask for help from DS Staff, Coaches, or even your sponsor. In our project we hit some technical roadblocks and needed help. When we communicated this to our sponsor, they gave us a technical contact who previously wasn't involved in the project. We set up weekly meetings with this contact who greatly helped our development process. Don't be shy about asking for help and accepting it when offered!

Push for customer interviews

When looking back on our project, it is sometimes difficult to discern the customer value, or if we went in the right direction with the product due to a lack of customer interviews. Customer interviews, while sometimes difficult to set up, are incredibly helpful in understanding the problem space and the customers the solution is meant to serve. This is particularly true for external customers facing projects. We recommend pushing for customer interviews on day 0 and pushing for interviews early and often. Potentially even as early as project rollout it is good to let sponsors know that the team would need to conduct customer interviews. This gives the sponsor the time and opportunity to help coordinate those interviews with students and gives students a chance to dispel any hesitation about customer interviews early on.