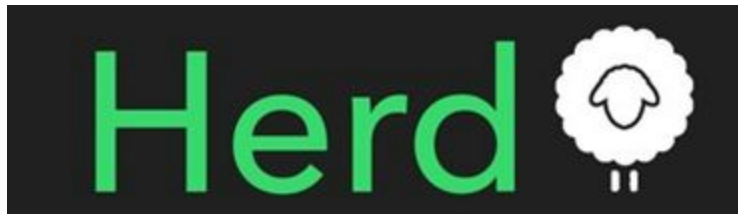


Hi-fi Prototype Report



Dartis Willis

Head Design &
User Testing

Cristian Lara

Design,
& Development

Neven Wang-Tomic

Documentation,
& Development

Value Proposition:

Never miss out

Problem and Solution Overview:

Finding reliable information about events is too hard. Deciding whether or not an event is worth going to is virtually guesswork, not to mention how difficult it is to find out about events in your area in the first place.

We are looking to fix these problems through Herd, a social platform dedicated to making event information easy and accessible. On Herd, users can find and share event information in a simple and streamlined manner. In addition to allowing users to keep up with events in their area, Herd empowers users to evaluate events by seeing ratings, comments, and pictures. Herd also gives event hosts an easy way to make sure their event reaches as many people as possible.

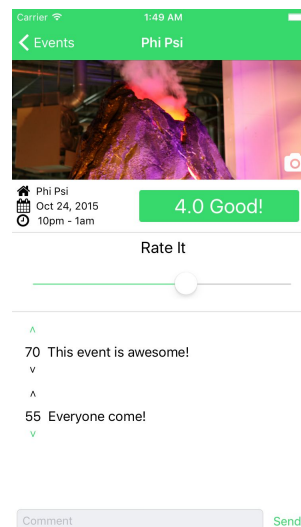
Tasks & Final Interface Scenarios:

1. Find and make informed decisions about events (Complex):

This is our core task, as it encompasses the main need our application is servicing. Broken down, this task is comprised of two subtasks: finding events and evaluating events. We decided to combine these two into a single task because they are linked. Restated, we found through testing that the latter subtask (evaluate) naturally follows from the former subtask (find). The task is achieved through two screens: a main events page (1) in which users can quickly see what events are going on in their area and how large/”hot” these events are, and a single event’s page (2) in which users can see more information about an event in addition to comments and images.



(1) Main events page

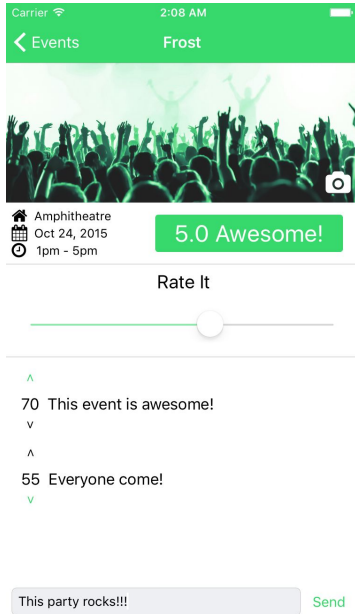


(2) “Phi Psi” event page

2. Share information about an event (Medium):

Because our primary event information is user-driven, it is important to us to provide a straightforward and easy way for users to contribute their opinions. There is a number of ways we allow users to achieve this. We have a comments section on every event in which users can

post and vote on comments (3). Users can also add images to events that they are currently at for others to see (4).



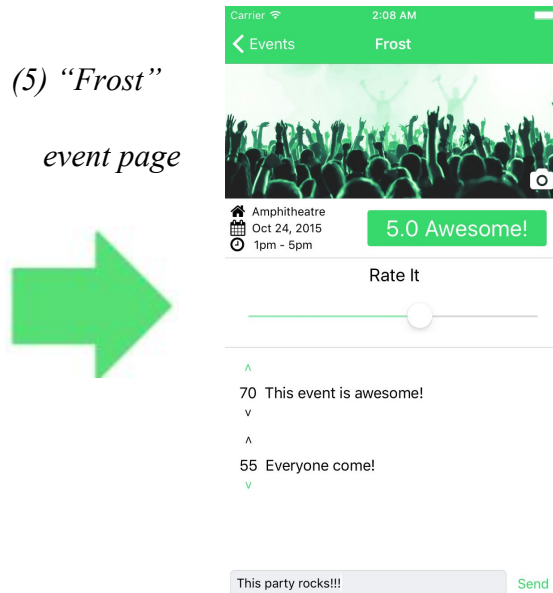
(3) “Frost” event page - adding a comment

3. Create an event (Simple):

Again, because our platform is user-driven, allowing event hosts and others to post events is a crucial piece Herd’s ecosystem. This task is very simple and simply involves a single screen (5) in which users post simple information about the event (location, date, etc).



(4) Main Events page

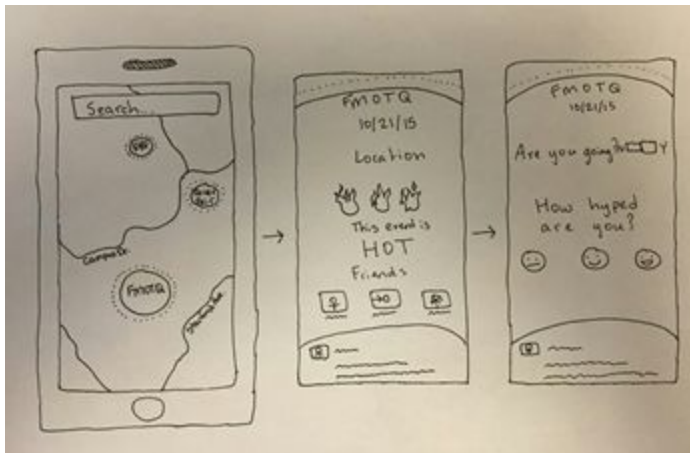


(5) “Frost” event page



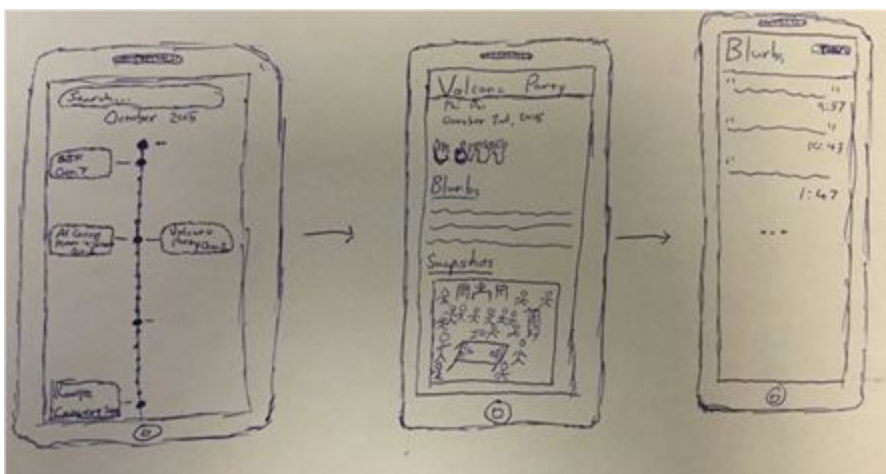
Design Evolution:

Once we finished our needfinding, establishing point-of-views, and generating “how might we” statements, we began our designing with three experience prototypes based on the aforementioned data.



(6) Bubble map prototype

The bubble map prototype represents events as bubbles on a map, with the bubbles placed at the event’s location. Pressing on an event bubble opens up that event’s page, where users can see basic event information, a rating, comments, photos, and a simple query as to whether the user is planning on going and how “hyped” they are for the event.



(7) Timeline prototype

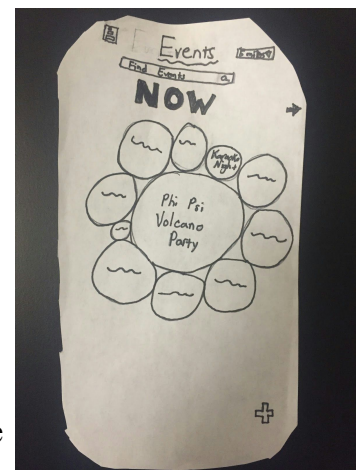
The timeline prototype represents events by their date and time, placing nodes on a timeline to represent each event, allowing users to view events that occurred in the past as well as events occurring in the future. The event page has the same in design as the bubble map prototype's event page.



(8) Real-time prototype

The real-time prototype displays a live feed of events, with each piece of the live feed leading to an event page that is, again, the same as the two previous experience prototypes.

For our low-fi prototype, we decided to go with a modified bubble map prototype, and put the other two design ideas on the backburner. The major feature we changed with the bubble map prototype is we removed map, so that the bubbles just float in space

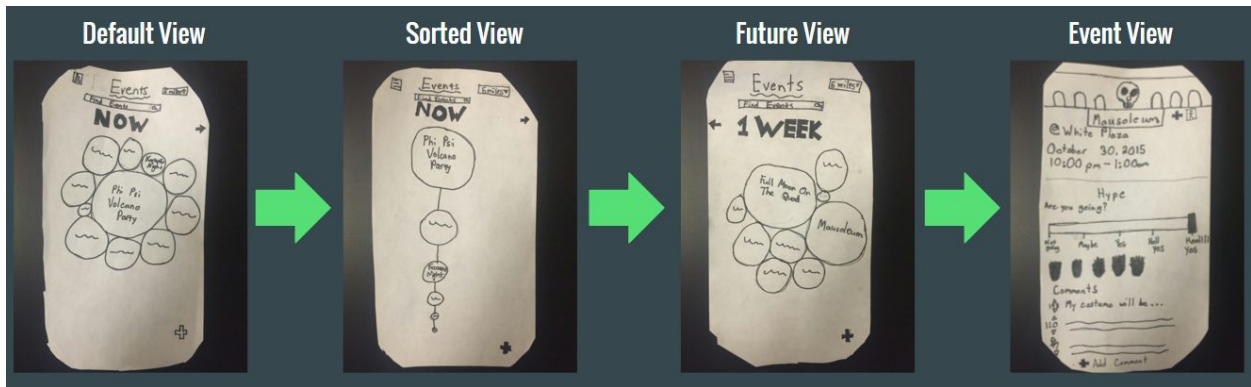


(9) Low-fi prototype main page

(similar to the way Apple Music presents genre preferences

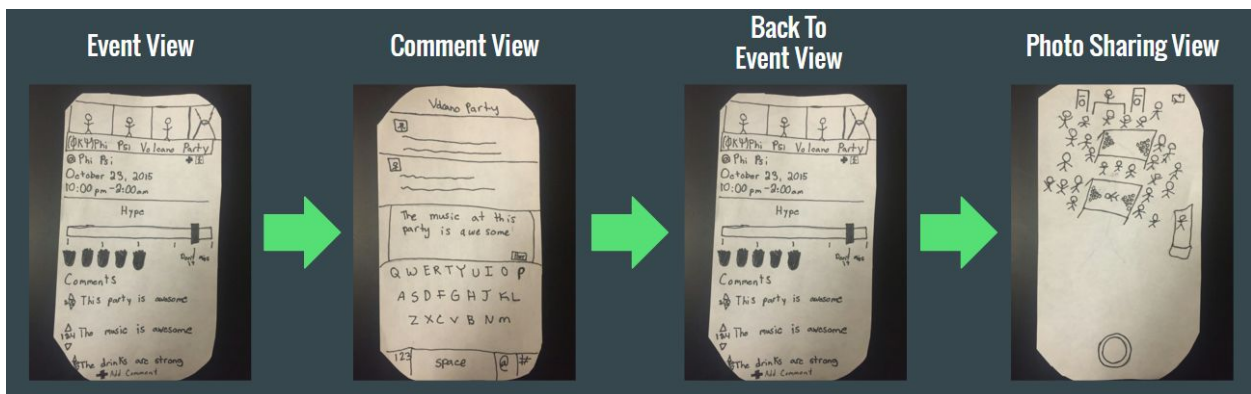
We decided to go with the first experience prototype because we felt that it was the simplest, cleanest, and most aesthetically pleasing prototype of the three we created. It was also our best

received experience prototype amongst our testers. We decided to remove the map because we felt that it added too much clutter to the event page. Additionally, we ran into design issues with the map; we felt like in many situations having a map created usability issues, such as how do we deal with places (such as San Francisco) which will have many events in an incredibly small space.



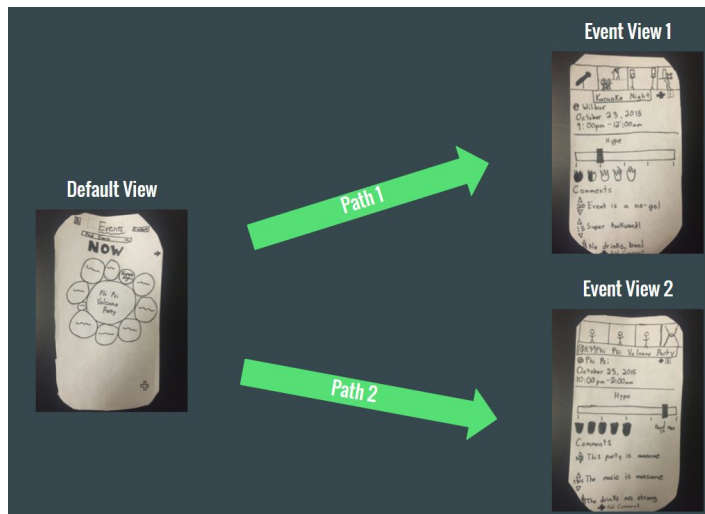
(10) Low-fi prototype task 1: Find events (simple)

Our simple task (10) was to find events. This can be achieved right from the main screen, which displays events around the user as bubbles. From here, users can see how events are faring against one another, and they can click on particular bubbles to open up the event's page to see more info. There is also a sorted view which organizes the bubbles by size. If users want to evaluate events that are happening in the future, they can sort the main page by date.



(11) Low-fi prototype task 2: Share event information (medium)

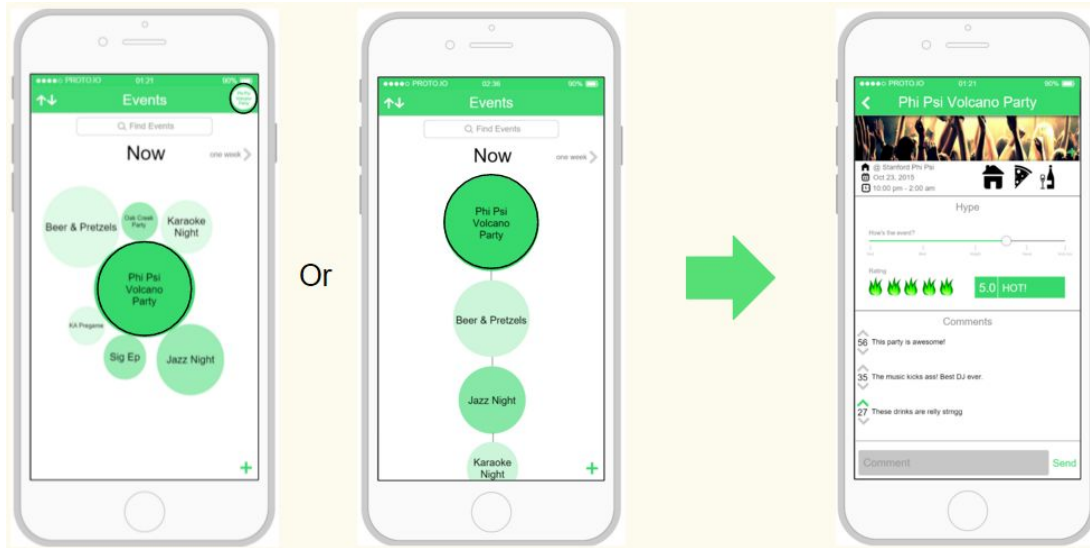
Our medium task (11) was to share event information. If a user goes to an event, the app detects that he or she is attending and allows him or her to rate the event, as well as add photos and comments. Adding photos is achieved through a native, in-app camera. Once the user posts content for an event, they can view it on that event page.



(12) Low-fi prototype task 3: Make informed decisions (complex)

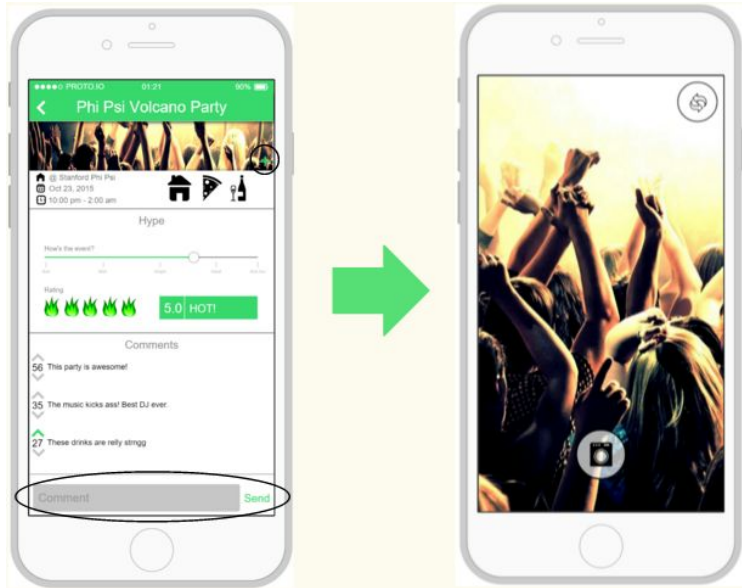
Our complex task (12) was for users to make informed decisions about events based on the data the app provides. Beyond finding events, we want users to be able to compare different events and their ratings to determine whether or not the event is worth going to for them. This task is simple in terms of executing it on our application, but complex when it comes to actually doing it because each person has different metrics for making what they decide is an informed decision. In this implementation, the size of the bubble indicates how popular an event is.

Our medium-fi prototype followed the same design principles as our low-fi prototype for the most part. The majority of our changes involved small quality of life improvements, cleaning the UI up, and fixing some obvious errors. Additionally, we changed our tasks a little bit.



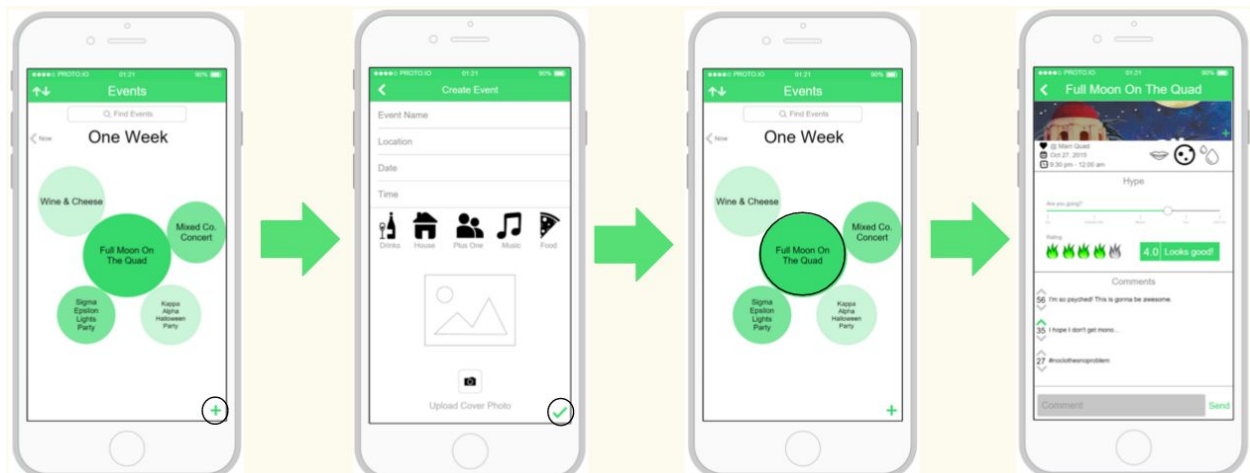
(13) Make informed decisions about events (complex)

We combined our complex and simple tasks from our low-fi prototype into one overarching complex task. Additionally, we added a “currently attending x event” button in the right-hand corner of the navigation bar on the events page. This allowed users to quickly get to the event they are currently at. If they are not at an event, this part is blank. Additionally, we changed our event page in certain ways. In the testing we conducted with the low-fi prototype, we found that users were primarily confused by the event page. Particularly, testers were confused about the slider bar for rating events and the event’s rating itself. The two were not well defined, and most testers could not tell what the two were for without having us tell them. In the medium-fi prototype we labeled the two parts, cleaned up their placement on the screen, and added a numerical rating with a message. We also played with adding icons to each event that the event’s creator could set. Finally we decided to add shading to the bubbles on the main page to differentiate between size of an event and its rating (size of bubble represents size of event, shade of bubble represents rating).



(14) Share information about an event (medium)

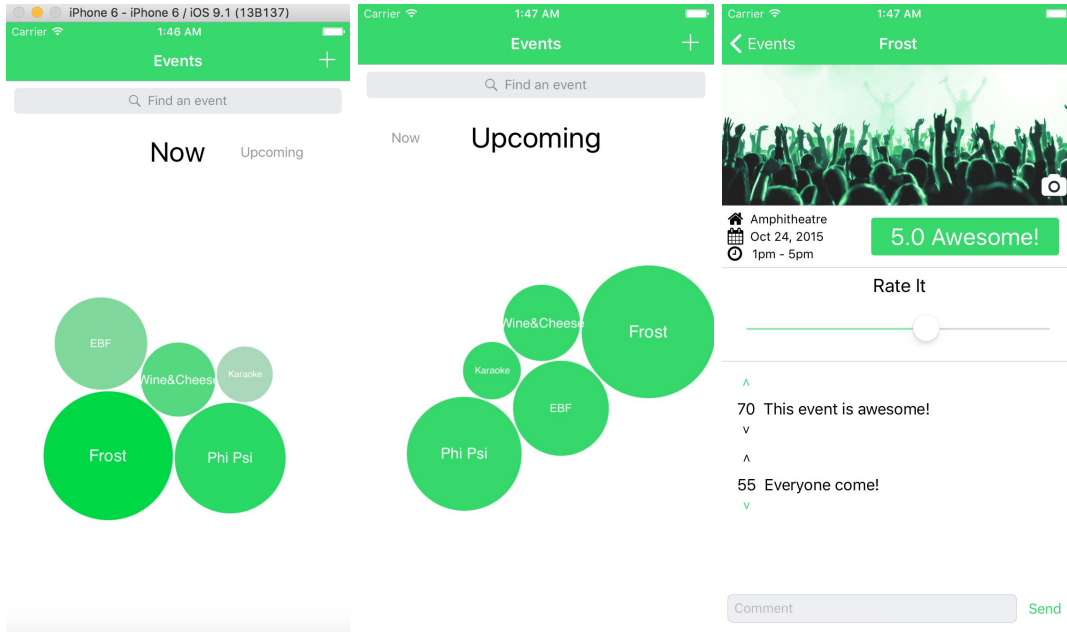
This design of the medium task stayed largely the same. We replaced the “add comment” button with an integrated comment text field because users expressed annoyance with having to go to an entirely different screen to add a comment. We also moved the “add photo” button to the bottom right-hand corner of the photo stream because previously the location of the icon was too far from the stream to indicate exactly what it is for.



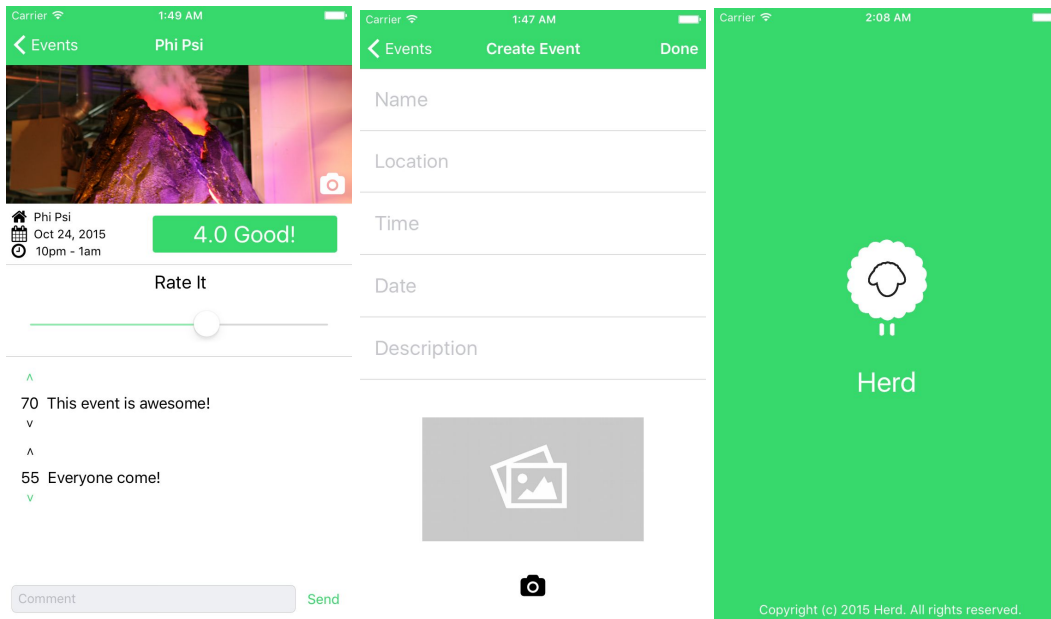
(15) Create event (simple)

We added creating an event as our new simple task because we honestly forgot that this needed to be a central part of the application. The task is very simple, users simply click the green plus at the bottom right-hand corner, fill in the fields, pick a few icons, then click the green check to confirm the event.

For our hi-fi prototype, we again stuck to largely the same design and structure as our medium-fi prototype. The changes we made were largely due to either fixing usability issues or making sure our design was consistent with apple's application standards. The large changes we made involved removing the "fire rating" from event pages, and simply having the numerical rating with a message. We made this change because we noticed many users simply didn't feel that the fire rating was any good, and that there was a disconnect between the shades of the bubbles and the fire ratings, causing a weird inconsistency in trying to represent the same data. We also decided to do away with the event icons because users were getting confused about their purpose, leading us to realize that they really just took up space without adding any meaning. Other small changes included moving the placement of the numerical rating and message, changing the future events page to display a generic "upcoming" message, and changing icons to match what apple standardizes for certain actions, such as changing the "add photo" icon to a camera (18).



(16) Main events page (now) (17) Main events page (future) (18) "Frost" event page



(19) "Phi Psi" event page (20) Create event page (21) Launch screen

Major Usability Problems Addressed:

After having our medium fidelity prototype reviewed by members of other groups we reviewed a compilation of their heuristic evaluations. Looking at the results we found a

multitude of minor violations with relatively quick fixes, however we also received a handful of serious violations. After reviewing each of these violations we took each reviewer's sentiments into serious consideration and changed our high fidelity prototype where necessary. Each of the major violations and changes are described below:

H2-1 Visibility of System Status [Severity 4]

“The bubbles display a lot of implicit information that would not be intuitive to understand for a first time user. On the home screen, does the size of the bubble mean the event has really great ratings? Or does it mean there are a lot of people going, or planning to go? Does the color scheme and shade also represent information? With the readme, I was able to understand this information. In though, practice you should have a first-time user walkthrough.”

and

H2-10 Help and documentation [Severity 3]

“The homepage the first time around can be confusing. For instance, if size of the circle is hype, then what is the significance of the color? Hype also? Perhaps you were planning on implementing this, but a first time walkthrough or descriptive tooltips.”

These confusions about the depiction of events in the home screen was a serious concern after it was unearthed by the evaluations. However, our desire to maintain a stripped down look for the interface lead us to not directly change the core look of the bubbles. In a completely finalized version beyond the scope of this course, we would include an initial walk-through with explanations of what each part means.. However, for the high-fi prototype, we did not implement this because it isn't a core function of our application; it would have just detracted from our demo.

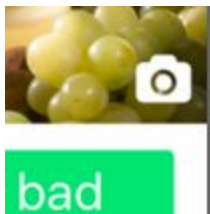
H2-5 Error Prevention [Severity 3]

“On the event pages, the add photo icon is small and hard-to-see. If I hadn’t read the readme, I would have not seen it at all. I had 3 other first time users look for the plus sign on their own and they could not find it until I pointed it out. It would be more intuitive if was a camera icon not a plus sign.”

This was a serious problem that ended up having an easy fix. Originally we used a small green plus icon for the add photo button that was hard to see (22). In the high fidelity prototype we used a bigger white camera icon for the add photo button (23). The new icon made the function of the button more clear for our users.



(22) Old Camera Icon



(23) New Camera Icon

H2-8 Aesthetic and minimalist design [Severity 4]

“On the homepage: minimalist and aesthetic now, but will get cluttered if enough events are added. Same goes for (and is probably even more noticeable for) ‘One Week’ view. Perhaps only show top 25 parties at given time.”

This projected problem is a situation that our program handles quite well. When 25+ events are added the main page actually looks fine. It could potentially be a problem if once the events page is flooded with 50+ events, but in general the system works for handling many events. Perhaps in the future we'll do some more testing to determine at what point to cap events if there are simply too many to reasonably display.



(24) Main page view of 25+ Events

Other interface changes

While going from the medium-fidelity to the high-fidelity prototypes we made the following other changes:

- Changed the sort icon to a more clear indicator of the type of sorting that would occur.
- We removed the “current event being attended” indicator on the top right of the home screen because it was deemed too confusing and unnecessary after further evaluation.

- We moved the “Add event” Icon to the top right of the upcoming events view because it was more intuitive to add events to “Upcoming events” versus the “Events Now” screen.
- We changed the shading scale for indicating the rating of events to better resonate with natural cultural cues.
- On the event page we removed the confusing fire icons, the confusing “party type” icons and simplified the event rating slider.

Prototype Implementation:

After receiving feedback on our medium fidelity prototype and planning out changes for the final prototype, we promptly decided to create the final version in Xcode using Swift. None of our group members had any experience using Swift or Xcode, however, Xcode’s storyboard view with drag and drop assets gave us enough confidence to proceed with a “learn as we go” attitude toward creating the high fidelity prototype. Collaborating on the code through a Github repository along with having no experience with Swift caused early problems. After a few instances of unintentionally overwriting other teammate’s progress and struggling with some basic aspects of making an application we gained confidence in our ability to produce an adequate high fidelity prototype by our deadline. Github & Xcode proved to provide a positive experience, however, we found that in some aspects Xcode left us at a loss. Certain nuanced features of our application (such as a segmented slider etc) were left out, the dynamic bubble system was a nightmare to figure out the first time, creating a scroll view for comments turned out to be buggy, and other setbacks arose. Overall Xcode provided a positive experience for us, and we were happy to create a final prototype of Herd that met our standards.

A large amount of Wizard of Oz techniques were necessary for a demonstration of our solution. In a real world application of our idea, all of the data including events and ratings would be solely supplied by our userbase. Because we didn't have a database, we manually **inputted** rating and size information for the sake of having a concise, clean demonstration. However, despite all of the preset data, we allowed judges to create their own event and see it appear as it's own bubble on the main screen. The nature of the presentation and the timeline we used for development of the demo not only lead us to utilize Wizard of Oz techniques, but it lead us to leave out some functionality and features that could have enhanced the experience of the judges and potential users. The main features/assets we left out are as follows:

- Indicators for current location for events being viewed
- Multifaceted filter on main page
- Sheep icons and a segmented sidebar on the event page
- Facebook friend integration
- Icons or information indicators within the bubbles
- Ability for users to post videos

Summary:

We had an awesome time creating this application. Having the opportunity to learn HCI design techniques by needfinding, creating designs, and testing our established ideas has been an eye-opening experience for all three of us. Being able to look back on all the steps we've taken and the progress we've made is really satisfying, and it's nice to have an end product to call our own. In regards to the future, we are planning on continuing to develop our app with a more

leisurely pace in mind. During winter quarter we are planning on meeting at least twice a week to continue designing and developing, and we're looking forward to seeing where we can move next.