

CoCo: Collaborative Collections

Jeff Garnier, Austin Chustz, Tom Kremer



CoCo is a way to collaboratively create digital memories of events.

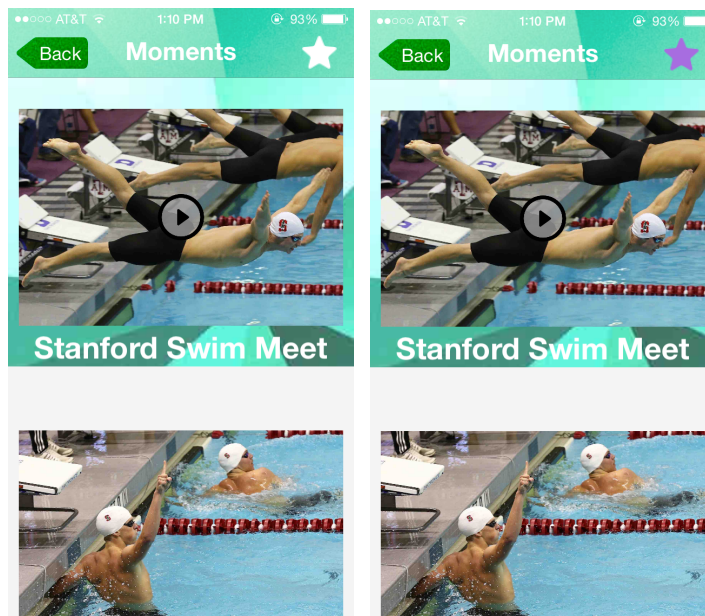
Problem and Solution Overview

When someone goes to an event, he or she can only experience it once, and only through his or her one point of view. Options for re-experiencing the event are limited and scattered, and usually don't actually simulate the feeling of being there again. With CoCo, anyone can see what the experience was like from other people's perspectives, making reminiscing much more engaging. With CoCo, an entire community of local event-goers can anonymously share photos and video to collections of content. The events will be posted by users, eliminating the harsh selection of which events get their own highlight reels, and the content can be submitted by anyone who attends the event, allowing for a wide spectrum of footage that helps the user view previously unseen parts of the event. The best content will be curated through a voting system to ensure that the final video summary is truly the best representation of what it was like to attend.

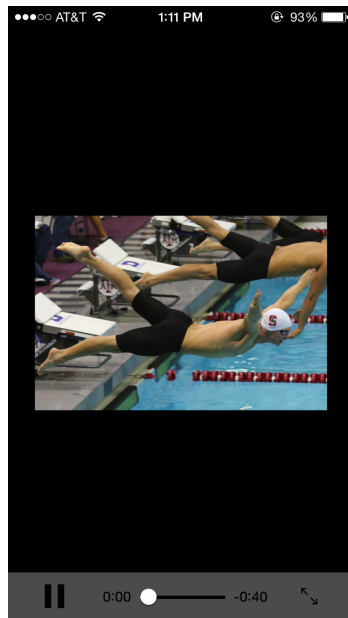
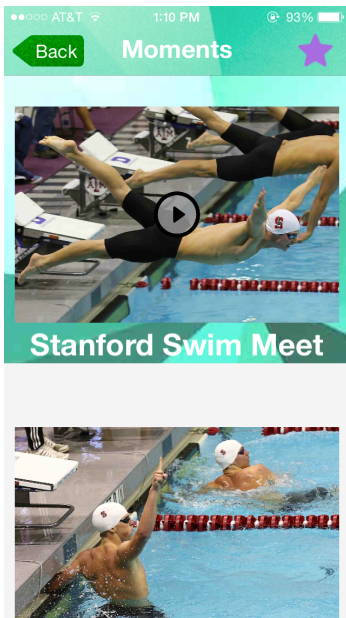
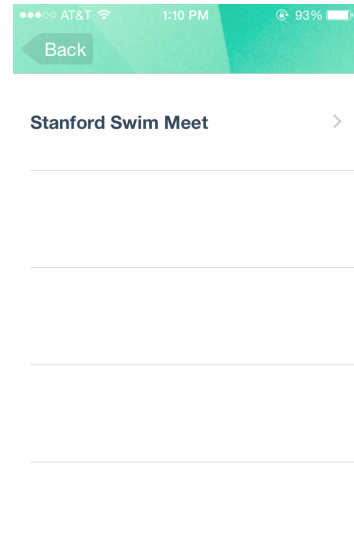
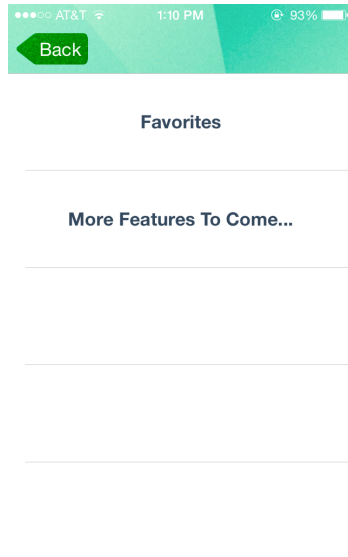
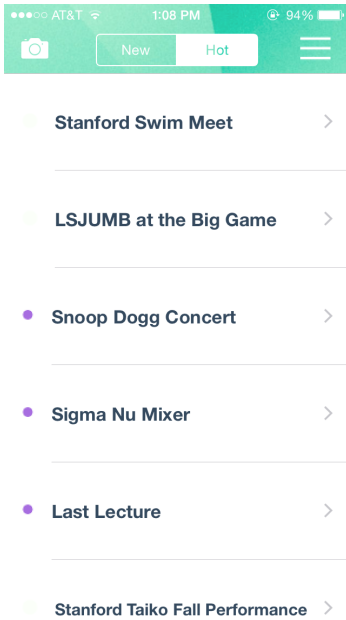
Task and Final Interface Scenarios

Reliving a Favorite Event (Simple)

Our simple task was to re-experience an event that you attended and enjoyed. We wanted to make this as simple as possible because the main focus of the app was to create a collaborative collection that users could watch to relive the event from the perspective of many different anonymous users. This task requires the user to first favorite events they enjoyed viewing and then view their favorite events and relive them by watching that events collection.



To favorite an event, just tap the favorite icon in the top right corner.

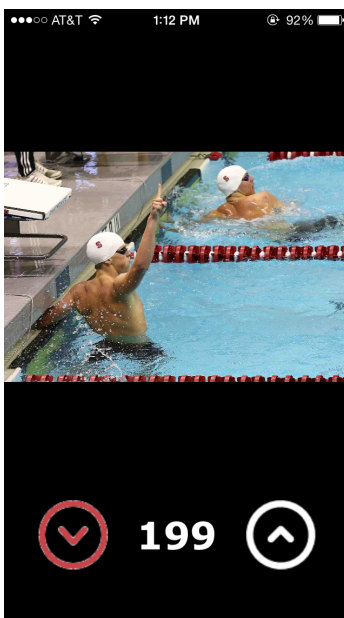
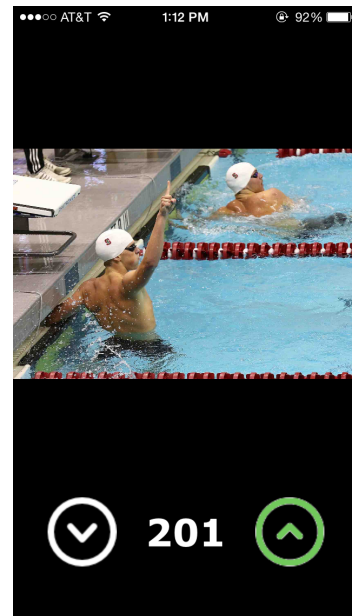
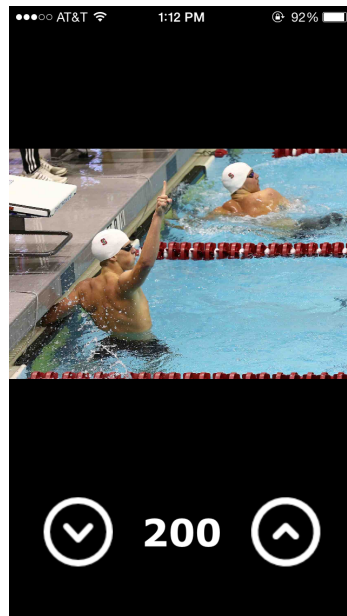


To watch a favorited event collection, users tap the menu icon in the top right corner and then tap favorites and select which event they want to relive. The first thing on the page is the collection; simply tap on it and the video screen will appear and start playing.

Viewing and Voting on Submitted Content (Moderate)

This task is a moderately difficult task to perform because it requires the cooperation of the whole CoCo community. Customers perform this task by upvoting or downvoting the posted content. This system aims to ensure that the coolest and most

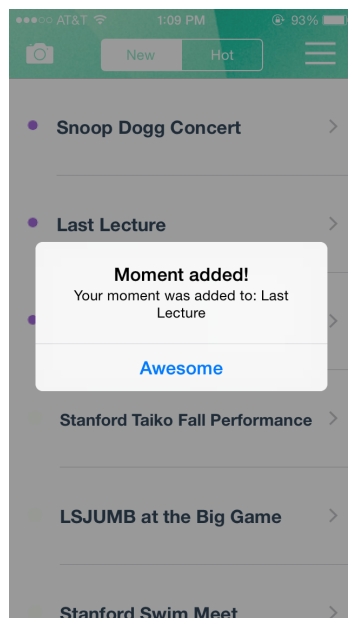
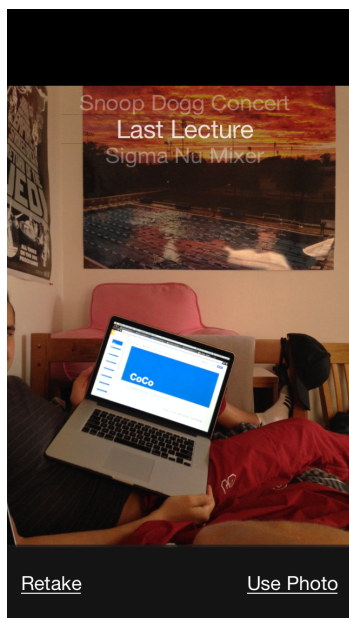
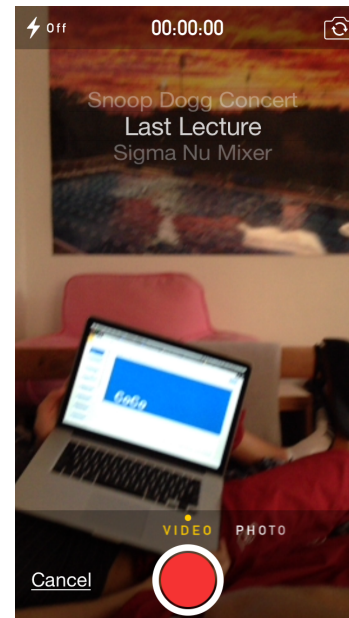
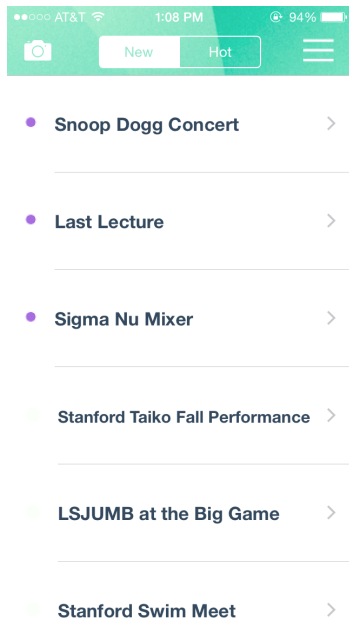
interesting content is found and the boring or inappropriate content is either reported or given negative feedback. If the whole community participates in curating the content then the final collection will be able to contain the best content from each event.



To vote on content, simply select an event and then select any submission on the following screen. The submission will appear on the page and tapping the up/down vote buttons will submit your vote. To get back to the event page you simply tap the image again.

Capturing and Sharing Content (Complex)

Capturing and sharing content is a complex task because it involves the user going to an event, taking a picture or video of something they think other people would enjoy seeing, and sharing it through CoCo. This task was chosen because customers want to share their experience of events so that other people can see what an event was. This is a very important task for the final collaborative collection.



To submit photo/video, tap the camera icon on the top left. This will give you the option to select photo or video. To switch between them, swipe left and right. It will also give the option to select which event you want to submit content by changing the scroller at the top of the camera. After taking the picture, the user can still change the event and it will ask if it should use that image or retake the picture. If the user decides to use that photo, the app returns to the home screen with a verification message for the user.

Major Usability Problems Addressed

Here is a list of the major heuristic violations from our medium-fidelity prototype, with some comments about the changes we applied to our high-fidelity prototype to address them.

The purpose/distinction of the three main event categories is still unclear

To avoid all the confusion of having three separate tabs for events, based on their times, we cut down the whole notion of three separate tabs and opted for only one list of events. This list contains the 25 most recent events, regardless of whether they are open for submission or not, though there are now purple circles that indicate if the event is still going on. We eliminated the “Upcoming” events tab because the focus of the application was to create crowdsourced video summaries, not to discover which local events are worth attending. This also allowed us to add the functionality of sorting the events in chronological order or by popularity. Lastly, this established a definite home screen, something we were arguing about during the earlier prototypes.

Upcoming	
Name	Relevance
Snoop Dogg Concert	97
Mausoleum	93
CoHo Open Mic Night	92
Cantor Art Show	86
Swim Meet vs USC	83
Amazon Info Lunch	80
Kappa Sig: Eurotrash	78

Upcoming Voting Archived

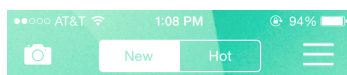
Voting	
Name	Relevance
Football vs Oregon S...	98
Beach Volleyball	93
Pumpkin Carving Co...	90
Theta Breakers	87
Men's Soccer vs Cal	85
CS 193p App Fair	84
Fireworks Show	82

Upcoming Voting Archived

Archived	
Name	Relevance
Water Polo vs UOP	94
Suites Sushi Dinner	91
CS Job Fair	90
MGMT at Frost	88
Stanford Taiko Autumn	84
Super Smash Bros To...	82
Jeff's Birthday Party	79

Upcoming Voting Archived

In the medium-fi screenshots above, the three partitions confused users. In the hi-fi screens below, one single tab streamlined navigation, while also allowing new features.



- Snoop Dogg Concert >

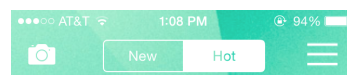
- Last Lecture >

- Sigma Nu Mixer >

- Stanford Taiko Fall Performance >

- LSJUMB at the Big Game >

- Stanford Swim Meet >



- Stanford Swim Meet >

- LSJUMB at the Big Game >

- Snoop Dogg Concert >

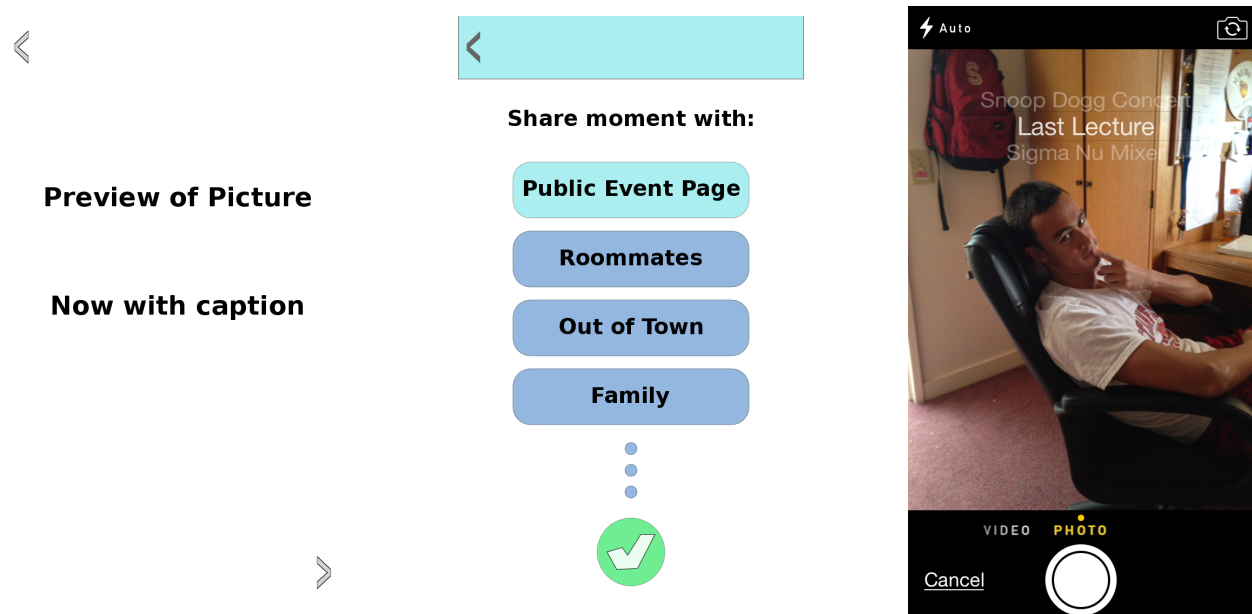
- Sigma Nu Mixer >

- Last Lecture >

- Stanford Taiko Fall Performance >

The app never allows the user to associate the content to an event

We already knew about this issue, but the tools we used in our medium-fidelity prototype made it difficult to address. However, in our high-fidelity prototype, we made the process much clearer. To submit a photo, you must select the event before confirming the submission in a scrolling menu that is overlaid over the camera interface. Apart from being really cool, this menu allows users to verify that they are sending their content to the right event page. If there were no events in the area at that time, then the user would not even be allowed to access the camera through the app, again ensuring that pictures are inherently paired with events.

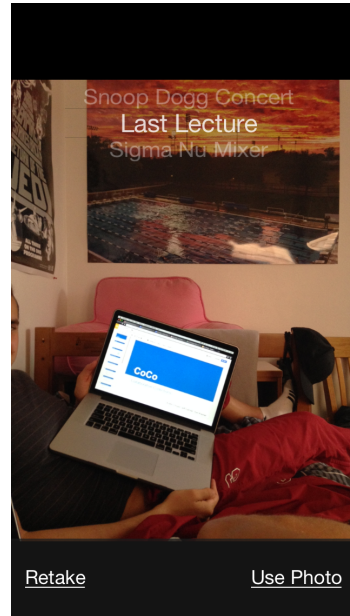
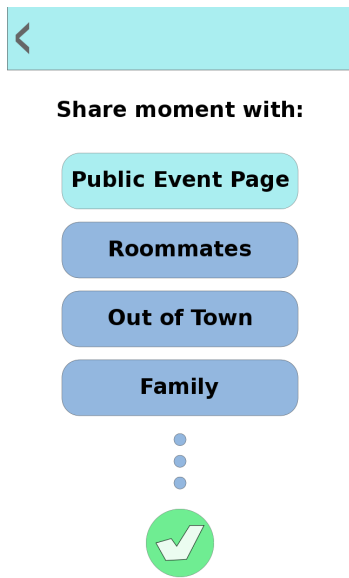


In the two medium-fi screens (the left and middle screenshots), the process was split and ambiguous. The hi-fi screen on the right gives the user a visual cue of where the photo will be submitted.

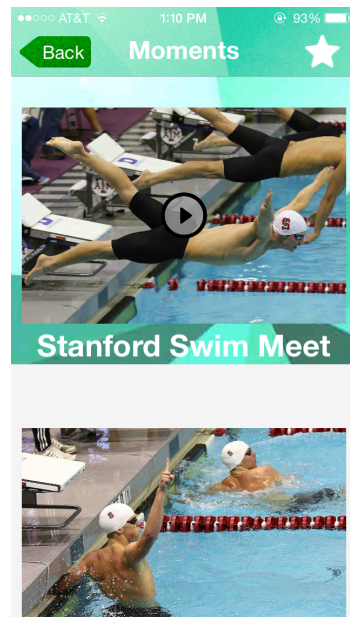
The share function is ambiguous. users were confused whether they were sharing with friends or setting public visibility options

For this problem we just removed the sharing with your friends functionality. This wasn't really the direction we wanted to go with the app and was just making things more complicated and unclear. When you take a picture/video you only have the option to share your moment to a specific event chosen from the event scroller on the camera screen. There is no sharing with groups anymore. We thought just submitting to an event made more sense and would confuse people less. Initially the app would use gps to locate where you were and what events were happening at your location but there could also be multiple events in the same area and this change also fixes this. We also completely removed the sharing final collections with groups of friends. We still have the ability to favorite and event so that you can relive it but we thought sharing event

collections was just more clutter to our app that was unnecessary.



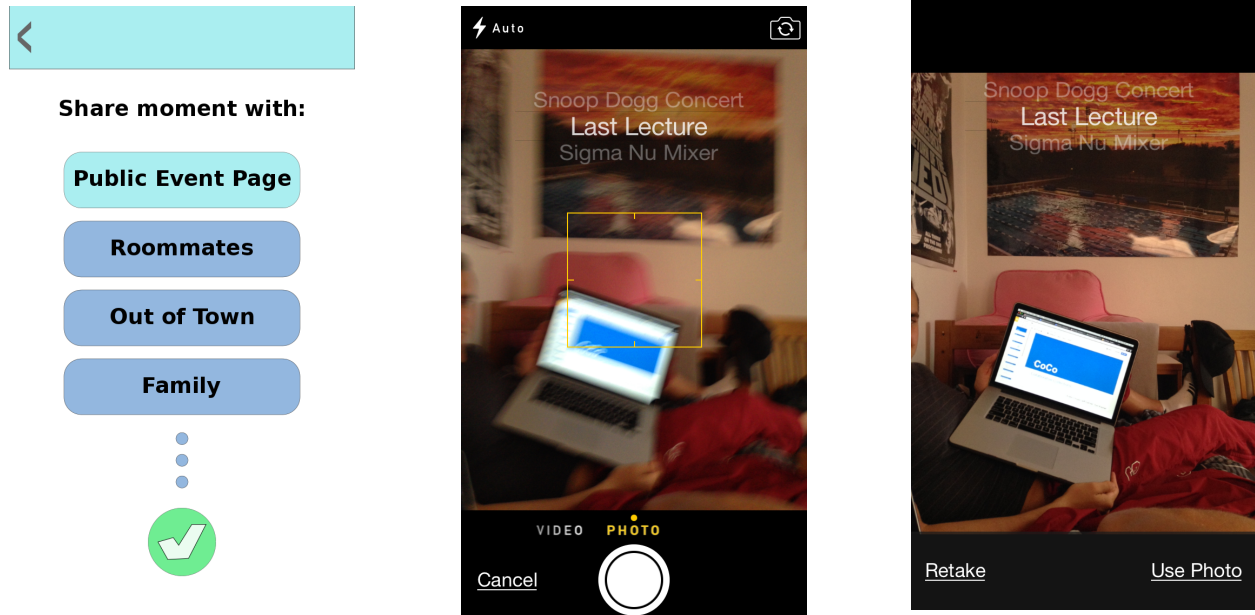
After the user took a picture in our medium-fi project, a share screen would pop up giving the user the option to share with different groups and the public. This was unclear and it was unclear which event it would be submitted to. The new version simply has an event selector, without any friend-related sharing.



We removed sharing the final collections with friends. You can still favorite the final collections though.

There is no way to back out of the share screen

Other than the fact that there is no more share screen, we made sure to have a way to back out of the camera/event selection screen, and also a way to retake the picture or video if the user does not like the shot.

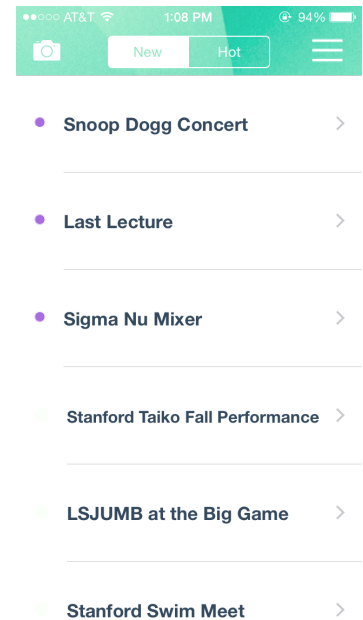
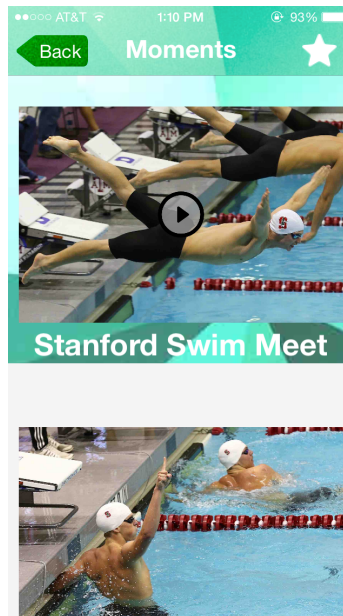
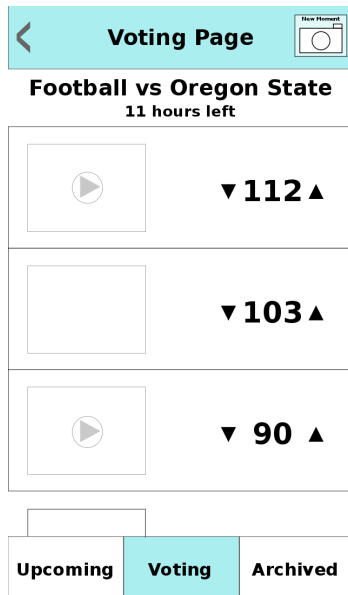


The medium-fi's share screen had a back button that we forgot to implement. The hi-fi definitely has a working "back out" function.

There is no way to remove content

The voting page is only sorted in descending votes (i.e. there is no way to view recently submitted content)

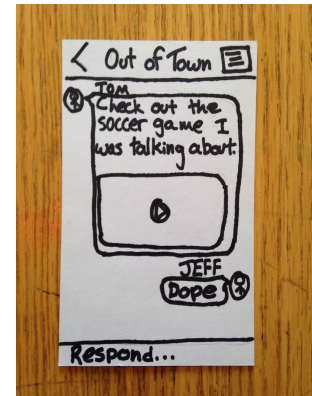
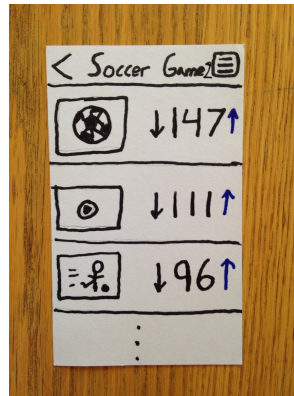
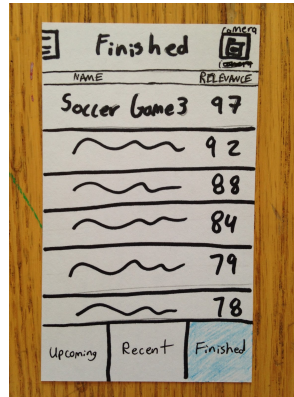
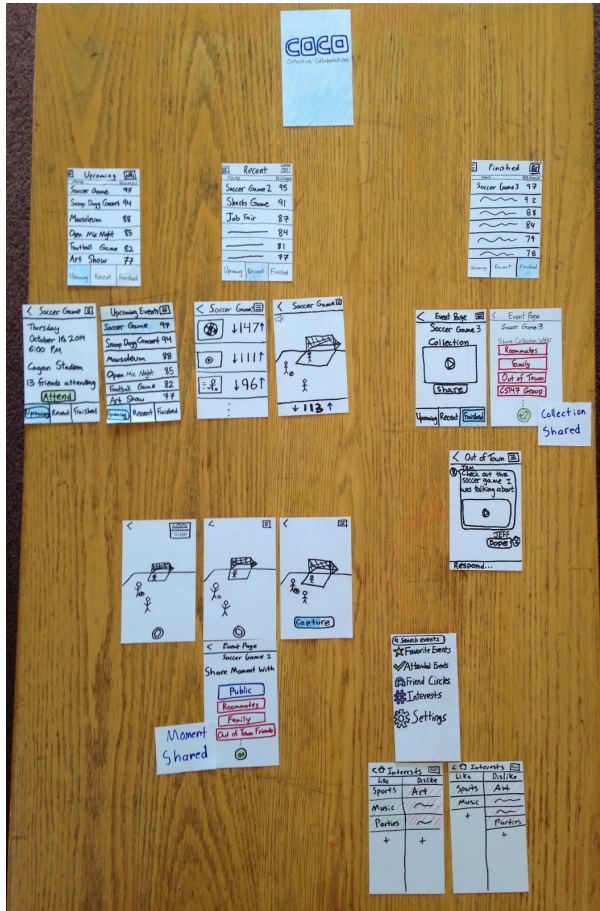
For these two heuristic violations, we actually never fixed them for our final hi-fidelity prototype. This was mostly due to time constraints; without any prior iOS programming experience, the team had a huge learning curve to overcome, which took longer than expected. Our main focus was on completing all of the tasks we outlined earlier, while also making the overall design and navigation smooth and logical. Features such as being able to remove submitted content and re-organize the submissions (like how the user can re-organize the events list) were among the top priority for secondary features. Given more time, we definitely would have implemented fixes to these issues, seeing as they are relatively simple.



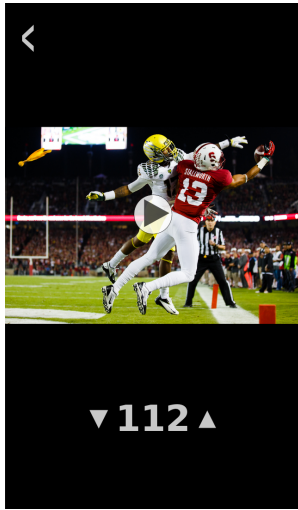
The medium-fi screen on the left did not have the ability to remove submitted content or to re-organize the submissions. The hi-fi screen in the middle did not have these features either due to time constraints, but we expect that it would look similar to the navigation bar from the events page, shown in the right screenshot.

Design Evolution

Let's begin with our low-fidelity prototype. It was designed with our contextual inquiries using the master-apprentice model in mind. For example, some interviewees told us about how they like sharing with friends more than they like sharing publicly. This led us to include friends and friend groups in the low-fidelity prototype of the app. The navigation map and some screen concepts are shown below.

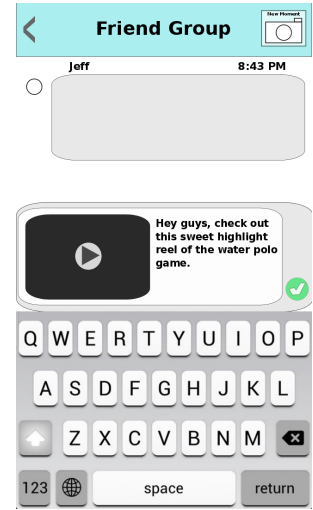
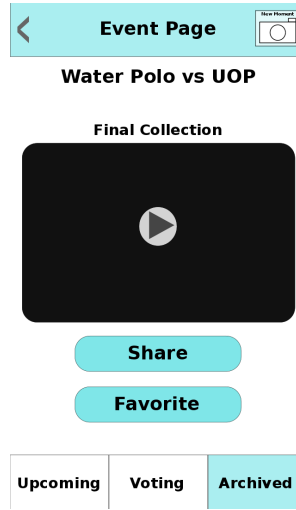


After having three people act as users on our low-fidelity prototype and attempt to complete the three main tasks, we did not see any alarming changes necessary. When people had suggestions, such as having an attend and unattend option for upcoming events, they were rather minor issues. The overall structure and navigation of the app did not change much for the medium-fidelity prototype, as shown in the pictures below.

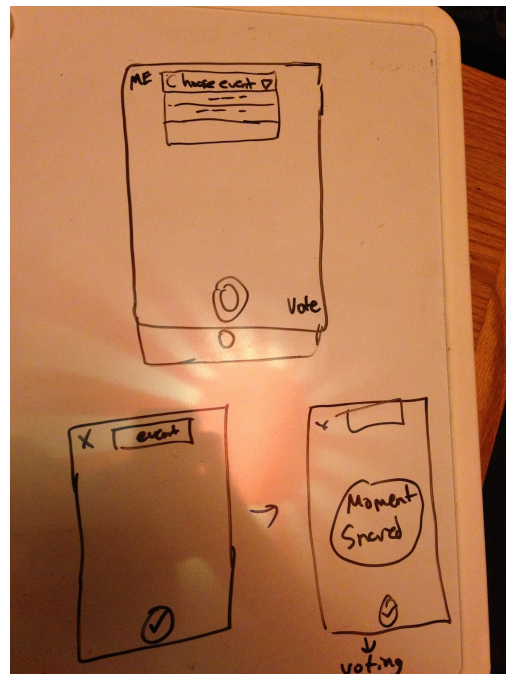
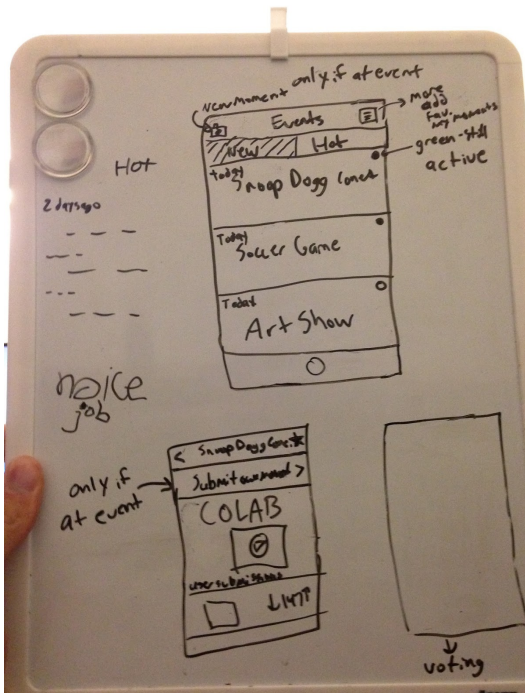


Archived	
Name	Relevance
Water Polo vs UOP	94
Suites Sushi Dinner	91
CS Job Fair	90
MGMT at Frost	88
Stanford Taiko Autumn 84	
Super Smash Bros To...82	
Jeff's Birthday Party	79

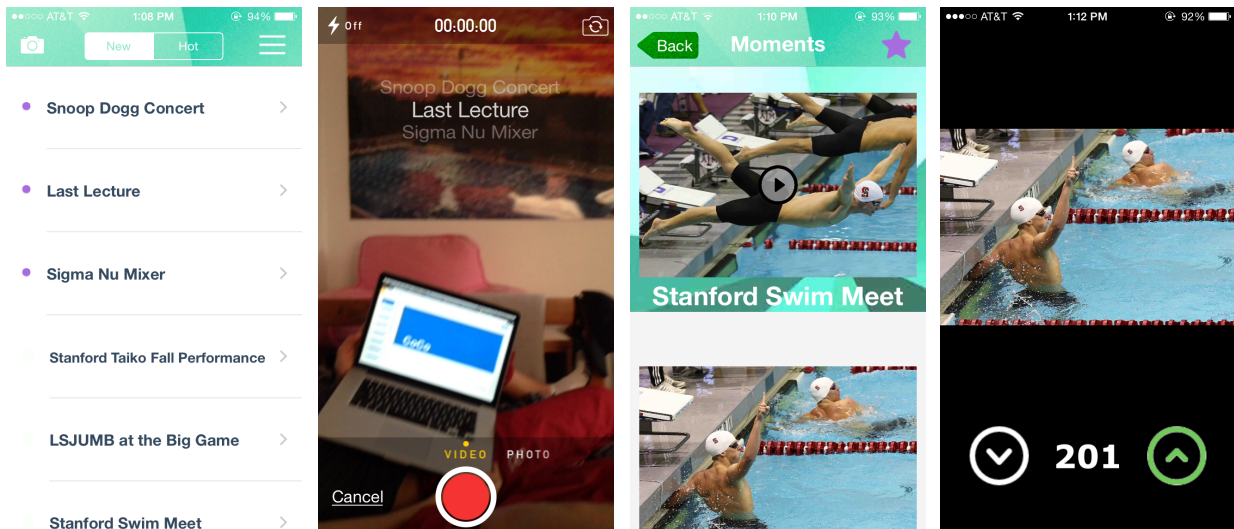
Upcoming Voting **Archived**



The heuristic evaluations on our medium-fidelity prototype were definitely the largest driving force for large design changes. We learned that there was much confusion over the different events tabs, too many different pages and screens, and clunky sharing options, whether with friends or publicly. We decided to completely overhaul the structure and navigation of the app in hopes of greatly simplifying it. The whiteboard pictures below show the new navigation map that we implemented for our high-fidelity prototype.



For the hi-fi prototype we used the heuristic evaluations results and new sketches to design the interface. Funny enough, our final screens were very similar to our whiteboard sketches above. One of the main things we removed was the three different categories of events. Instead we just have a constantly updating list of events that you can sort by popularity and time. We also made the background of the top navigation bar a cool image with the same color scheme of our whole app. We also made the method for submitting content much simpler and cleaner. If you are in the area of an event, then when you open the camera there will be a list of events you can submit to and you can just choose an event to submit your photo/video. If you are not at an event then when you tap the camera a screen would appear telling you that you are not within range of the event so you cannot submit content. Another large change we made was the removing of friend circles and sharing collections. This added a layer of unnecessary complexity to the app that took away from what we were trying to accomplish.



Prototype Implementation

To build our final hi-fidelity prototype we used xcode. We wrote our program in swift and used interface builder to help with the screens. Xcode was really helpful in making the screens in interface builder. It was fairly easy to make the screens we wanted and have them look the way we wanted visually. Xcode made it really easy to set colors and get the correct layouts. Most of the problems we had were from transitioning between screens. We had trouble linking screens together and sending information between screens. It was also difficult to test the camera because to run the

camera you actually had to get the app on your phone because the simulator did not have a camera. It was also tough getting used to the swift language since we had no prior experience with it. Over the course of developing the app, things started making more sense and becoming easier but the getting there took a significant amount of struggle and time. Also, since swift is a fairly new language the documentation for it was very sparse and hard to find.

We used several wizard of oz techniques in our hi-fi prototype. For the final collection of the swim meet we didn't actually make a video in the app by combining the top submissions. We made a video in iMovie from the submissions and when you clicked the collection for the swim meet it played the video we made. We also made screenshots of the video clips with a little play button on them to signify that those submissions were actually videos. We did not make a thumbnail for a video in the app.

Our app required a significant amount of hard-coded data points. We had to hard-code in all of the events and the initial submissions for those events. You could submit photos and videos in the app but we also wanted to have some submissions for each event already present so we had to hard-code those. We also had to hard-code some of the events to appear to be in your area so that we could perform the task of submitting content to an event.

One thing our app is missing is the location functionality. The app currently does not have a location associated with an event and does not check where you are to decide which events are in your area and you can submit to. In the future we would need to add this functionality. The events also don't change because we did not associate a time with an event, so the open events will always be open and the closed events will always be closed. The app also currently lacks the ability to remove content that specific user submitted. We wanted to add this feature but unfortunately we ran out of time. There is also currently no way to create an event. In the future we would like to add that functionality but it was not required to accomplish our three tasks.