

Cramr

Executive Summary

Cramr is an iOS app that enables cooperative learning by allowing users to seamlessly create and join study sessions on-the-fly through a light-weight interface. The need for the product is apparent to any student who has taken a class with a few number of friends, limited teaching resources, or has been in a situation where he/she needed immediate help. Cramr's advantage over alternatives to filling this need is that it enables all the functionalities of competing products in *one* cohesive and intuitive platform. Moreover, it's potential audience is immense; currently, there are 17.5 million student enrolled in colleges and universities across the United States.

High Level Description of the Project

Cramr is an application for iOS that seeks to provide an integrated, easy-to-use platform for students to form study groups. We envision the user-flow in our product as follows:

Upon first opening the application a user will sign in through Facebook and register for courses. After this first sign in, the user will be presented with our main view which allows the user to select a course. Another view will segue, presenting the user with the option of either joining an existing study session - ranked by proximity and number of friends in the session - or creating a new one.

When browsing a study session, the user is able to see what other users are currently in the session, the location of the session, and the time the session started. Should the user

choose to create a new session, Cramr will prompt for an input time, location, and type of assignment being worked on. That is it.

In designing Cramr, we want to create a user experience with the least intermittence; this concept is a central theme in our design. More specifically, we want the user to be able to open the application and sign up for a study session as quickly and as seamlessly as possible. Whether that be joining or creating sessions, the entire user-flow will take the least amount of interruptions. We believe this is crucial in the success of our product; Cramr is a light-weight platform that *enables* students as such, it will not be outwardly complicated and feature-heavy even while it's internals are complex.

Extensions

Possible extensions to the viable product must be examined thoroughly. As stated above, Cramr is a tool that facilitates and empowers, we do not want it to be a tool that distracts.

Social Media: The first extension that we will consider is linking other social media networks such as Google+, Twitter, and LinkedIn. Moreover, we will also consider designing Cramr such that it can post statuses on a users social media profile (i.e. allowing Facebook friends to know that he/she is starting a study session). This integration, we believe adds to the core values of our product without being cumbersome or distracting.

Messaging: Another possible extension that we will consider is messaging. We can further subcategorize this idea in two parts:

1) Sending Invites: After a user has created or joined a study session, Cramr will give the user the option of inviting a specific student to the session.

2) Chat: A user that has not joined a session will be able to chat with members of an existing study session. On the one hand, this feature may be helpful since it will allow a user to bypass study groups that are not appropriate for his/her needs (e.g. consider the situation where a session is working on a problem set and a user has already done the first three problems, it might be useful for the user to know how far the session has progressed in the problem set). On the other hand, this may conflict with the core value of our product, that Cramr is a tool for productivity. Chat could create an unwanted distraction that disaffirms our vision. For this reason, we plan on building prototypes to test with real users and gather feedback on the relative advantages and disadvantages of this feature.

Endorsements: Another feature that we would like to prototype and test is enabling users to give endorsements. More specifically, if a student finds that another user in a session was particularly helpful in teaching him/her the course material, Cramr would enable the user to endorse, or upvote, the helpful user for a particular set of skills. If this feature proves to enable the core vision of our product we may consider integrating LinkedIn endorsements with Cramr.

Integrating Piazza-API: For a richer set of content, we will consider integrating the unofficial Piazza-API, allowing users to search for relevant posts on Piazza via Cramr. This poses a challenge of filtering and sifting through Piazza posts, determining which are relevant for the current study session and presenting them in a way that is uncluttered and smooth.

Design Approach

We chose to develop for mobile devices because it enables the desired user-experience. The application is for students to find study groups on-the-fly and mobile platforms provide the best avenue for this experience. In addition, some of the features that we wish to include can

only be done with mobile devices. For example, location-based data is necessary for some of the features that we seek to implement.

Why Students Need Cramr

Every student is familiar with the pain of having to complete a difficult course or assignment alone. Students often need others with whom to collaborate and learn from. Current methods that address this problem are paid tutoring, office hours, or cumbersome group messaging. Cramr is free; it is not bound by understaffed classes or crowded office hours. Cramr is light-weight and seamless; it has none of the disadvantages of bulky and unfocused group messaging apps (see Advantages Over Competing Products below). Our vision is an application that facilitates and enables student success by fostering an environment of sustained cooperation and relieving students in situations where they need immediate help. Cramr was made by students for students, it's need is palpable for anyone that learns through active engagement and cooperation.

Advantages Over Competing Products

To explore possible competing products for our market, we systematically describe each of our product's functionalities and evaluate the market segment that these functionalities address:

A. Creating a study group.

- At its core, this functionality allows users to connect with other students (including their own friends). Therefore, group messaging and group sharing apps are all possible competing products. Examples of these include GroupMe, and Band. Cramer's advantages are twofold. One, our product allows users to

connect with others *who they do not know*. Thereby directly increasing a user's access to learning resources. Two, our product is focused on education and thus allows for a more centralized user-flow free of distractions.

B. Finding a study group

- Our product uses location-based data to allow users to find study groups.

Consequently, existing location-based messaging apps are possible competing products. Examples of these include MapChat, and Drop. These do not have a clearly defined vision and purpose Moreover, they do not have an integrated way of forming groups about a specific topic (such as a course).

C. Resolving Scheduling Conflicts

- As our product is designed to help users with differing availabilities schedule study sessions, apps and services that help resolve scheduling conflicts can be potential competitive market segments. Examples of these are Doodle, When Is Good, Time Bridge, and When2Meet. Our application is focused on immediacy, it is not about scheduling a meeting to work on a class a week from today. It is about enabling the user to learn and cooperate *now*.

Value Proposition

The value that our product brings is bundling all the functionalities stated above into *one* cohesive, easy to use platform. Applications exist for each of the functionalities that we are implementing, but they exist in *isolation*. Our product is designed for ease of use and elegance; it brings together the functionalities of competing products in different market segments under one user-friendly product.

Potential Audience

The potential audience for Cramr is immense. Currently, there are 17.5 million students enrolled in colleges and universities in the United States.¹ Our product could help all of these students. The application was designed with ease-of-use as our central theme, as a result all that is required to use the product is a social media account.

Major Technologies Used

The mobile application will run on iOS devices, principally iPhones. Therefore, it will be written in the Swift programming language. For design and implementation purposes, we will be utilizing a number of tools such as XCode's Interface Builder and Cocoapods for dependency control.

For the backend, we will use Parse (a mobile app hosting platform) integration, the relational schema will be designed in such a way that our application values speed and efficiency. To allow the user to sign in will use the Facebook iOS SDK and the Google+ SDK.

Resource Requirements

Parse will abstract most of the backend engineering needed to build the application. All our member have viable developer environments for Swift, thus the only resources that we will need to set up to enable a uniform developing environments are Cocoapods for dependency control and Github (a web-based git repository hosting service) for version control. In terms of testing, we will use three iphones of different models as well as two ipads.

¹ <http://www.statisticbrain.com/college-enrollment-statistics/>

By testing our application with different hardware specifications we seek to create a refined and unblemished product.

Assessment of Risks

One challenging aspect of this project is that our team has minimal iOS programming experience. None of us have completed a project of this scope for iOS. Therefore, we must quickly acclimate to the programming environment. A possible mitigation for this risk is the mentorship from our assigned TA. His advice and suggestions will prove to be invaluable in the success of our product. In terms of the risk that our product will not be as marketable as previously thought, we will iteratively create mock-ups of the UI and collect feedback from current students. Our final product will be influenced directly by user-feedback.

Next Steps

The next step in the development process is user-centered design. Now that we have developed the logistical model and need that we are addressing, we seek to build a product that addresses this need in the best possible way. To accomplish this, we will develop mockups and prototypes for each of the interfaces that our application will implement. Following the design thinking process, we will then test the ease-of-use, intuitiveness, and aesthetic appeal of the mockups with members of our intended audience. Re-iteration is key; our first attempts at interface design will be strengthened by getting feedback from actual users.

Following the test phase we then proceed to build the model for the backend-end technologies that we will use. Again, using the design thinking process will be crucial in building the best model. Once all the high-level abstractions have been tested and have been

revised (after user feedback), we will implement the app prototype; user-feedback and testing will be necessary to refine our product.