Kern Khanna
Cody Sugarman
Connor Hegarty
February 15, 2017

CS194W Project Proposal

The Product

SportsFeed is designed to be a mobile iOS app that allows one to follow their favorite NFL team. On first use of the app, the user will be prompted to enable geolocation services and create an account via either a Facebook OAuth or simple email/password combination through Firebase. Once an account is created, the user can tailor their main feed by selecting their favorite teams. Each of these teams will have a "team page" including that team's logo, recent scores, relevant statistics and upcoming matches. While the user will have the option of viewing and subscribing to any team (by adding them to their feed), SportsFeed will also make a team recommendation based on the user's location. For instance, a user in San Francisco would most likely follow the 49ers while a user in Boston would follow the Patriots.

SportsFeed will also include a feature that enables users to engage in game-day threads for supporters of both teams. This will be accessible through both team's pages by tapping on a "live" game in the provided list of upcoming matches. The threads will allow users to engage with friends by posting comments and pictures.

The Need

According to Facebook, 4 million users were part of the Super Bowl conversation on Sunday, up from 60 million last year — about a 7 percent gain. The company says there were 240 million Super Bowl-related interactions this year, which is up from 200 million last year (marking a 20 percent increase). The story is similar on Twitter, where the company claims there were 27.6 million global tweets about the Super Bowl this year (McGee). There is a clear need for a simplistic software that allows fans to engage in the sports viewing experience. Currently, there are many sites like ESPN and NFL.com that store statistics about sports team. However, the purpose of these sites is to educate fans about their favorite teams rather than to make sports watching more engaging. An issue with Internet game day threads on sites like Reddit is that they are saturated with advertising and not welcoming to the average fan. These websites are also incredibly broad and may not be as inviting to an audience primarily interested in football.

The Audience

The potential audience for this project will be fans who enjoy the NFL. Roughly 50% of all Americans follow the NFL, meaning the national market size would be close to 150 million people. We will begin marketing SportsFeed on college campuses because college educated kids are more inclined to try out new software and the insular network will allow us to test and refine the product before a larger launch. Assuming this launch goes well, we will then try to realistically capture up to 0.8 to 1% of the total market share after Year 1. Since the application is designed to be easy to use, users will not need to be technically sophisticated. The user will likely already be familiar with interfaces such as Snapchat and Facebook and we will design our icons and functionality similarly to ensure ease of use.

The Competition

While there is no shortage of sports app for iOS, the majority provides functionality that prioritizes score reporting, news, and live sports streaming. The most popular competitor is currently ESPN, topping the sports category for unique visitors (13.8 million) and tallying 785 million total minutes of usage in January alone. While this will be our biggest competitor, they do not offer chat feeds to engage the user, a feature we hope will enable us to take significant market share. Cable subscribers can also use the WatchESPN app to stream live games and watch on-demand videos. The most direct competitors to our application are the NFL app and GameOn. These both provide custom feeds for sports fans, with GameOn providing communication between fans of both the same and rival teams. Other competitors in this space would include WhatsApp and Facebook. While they are not designed around a centralized sports feed, they are currently the most common means of online communication for sports fans.

Where SportsFeed aims to stand out is in providing a centralized feed to be shared amongst friends of the user, something no other app currently does. Instead of connecting all fans of the sport, our goal is to create a small community amongst friends (via Facebook and geolocation) where fans and rivals can share photos, comments, and recent news.

Technical Design

We are designing an iOS application in Swift using an MVC framework to implement this product. To complete this project, we need to first fetch the relevant sports metrics, create game-day threads regarding this data, and then render the information in an appropriate manner.

We are planning to get our data from the ESPN NFL page. Specifically, we will initialize a URL session in the background to pull the relevant data (team logo, schedule, location) from the http://www.espn.com/nfl link. To help us pull the HTML and JSON data from ESPN, we plan to use the TFHpple Parser library (a common Objective C and Swift library) to gather the relevant metrics. In order to make sure that the app works offline, we will cache some of the data when the user pulls information on the team for the first time.

Another major goal of this project is to enable game-day threads. This will most likely be done using the backend server Firebase (JSON database). We will store user accounts and the chats on a firebase database. Once users are authenticated, we will be able to add privatization to their account and store their favorite teams for example. To implement the chats, we are going to use the provided JSQMessagesViewController (a strong messaging UI) with CocoaPods that allows instant messaging.

Using table-views and scroll-views will be essential to getting a good UI. We will most likely use a table view to store the number of teams in the NFL, and then a map view with a scroll view to show the basic schedule. To help visualize our intended goal, we have included potential mock designs.

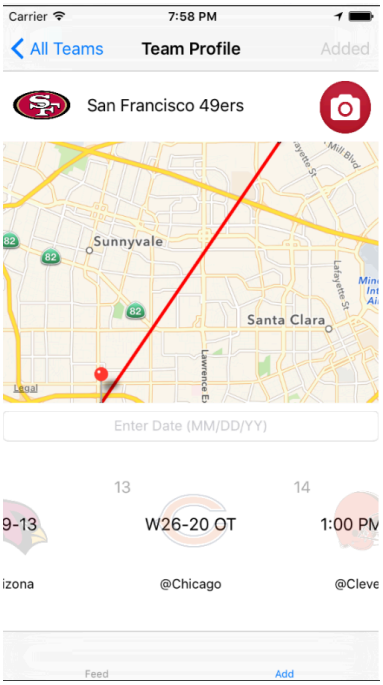

Figure 1: The "Team Page"

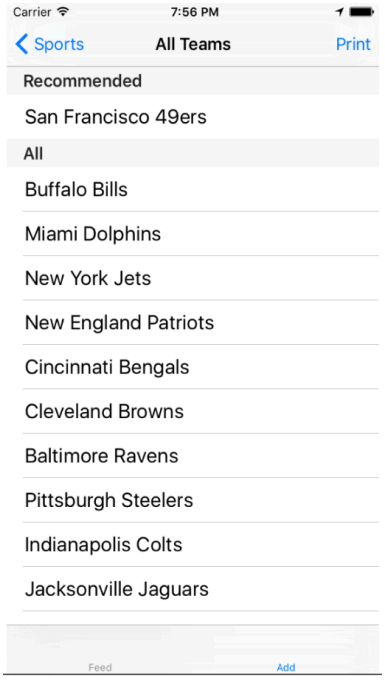

Figure 2: The feed of all NFL teams

In Figure 1, the user will see a map, which will ideally give directions to the stadium and a camera button to add pictures about the stadium. They should also be able

to see the result of the game closest to the date provided in the input field. Clicking on each individual game will bring up a chat thread, where users can talk to each other and converse. This will involve synchronizing the UIMapKit framework and the directions API. Since directions can be costly, clicking on the map will redirect the user to the third party Google Maps or Apple Maps. In Figure 2, note that it provides the user with a favorite team. The favorite team is the team geographically closest to the user, and it will be calculated also using the directions API.

Necessary Resources

We should be able to build our project with a Macbook, XCode, and developer tools such as Firebase. For testing purposes, we should only need an updated version of an iPhone. At this juncture we see no need for unusual requirement or particular challenges.

The Approach

While there are different ways to deal with the problem area our product addresses (as done by apps like ESPN and GameOn), none of these applications enable users to interact with their friends on one centralized platform. We believe our approach will encourage users to interact more frequently as it will ideally replace any group texts or GroupMe platforms they would previously use to discuss sporting events. We also project higher user retention due to the nature of the user base.

Associated Risks

As we will need to use developer tools and services our group members have no previous experience with there is a risk that the steep learning curve will slow the progress of our app development. We will mitigate this risk by assigning each group member specific development tasks so they can learn the technologies needed to achieve these goals and teach the rest of the group. As our app will not require any personal information from the user (outside of login credentials and geolocation) there should be mild security risks for this project.

Another potential risk is that a competing company might try to enter this sports market because it sees the same opportunity that our group members see. With only three people on the team, we would be hard-pressed to match the effort of a hardworking startup.

Next Steps

The first step we will take in building our project is determining how we want the UI to look and function for the user. We have started to use programs like Sketch to make mock designs in addition to pen and paper. Once we have determined how we want the application to work, we will start building out screens in XCode through Storyboard. After we have a draft completed, we will run user tests to get feedback and improve the UI.

Works Cited

McGee, Matt. "Super Bowl-related Social Activity Was up Sunday on Both Facebook & Twitter." Marketing Land. 07 Feb. 2017. Web. 18 Feb. 2017.