MySwag iPhone Application Development Project

Marymount University

Draft

Honor Pledge: I acknowledge that the Capstone Project is an independent study project to be completed individually. On my honor, I have not received aid on my Capstone Project other than what was provided by my faculty mentor and any persons explicitly cited in my work. I further acknowledge that if I have given any aid to another student in this course, the instructor of this course was made aware of my contributions.

Contents

Objective

The goal of the project is to solve a problem I have seen my friends face repeatedly. I plan to create an application that will work as small tallying tool for people to quickly ask for their friend's opinions. The way the application works is that a user will either take a picture or choose an existing one from their library and then select the contacts that the user wants to send it too and press send. Once the user has selected their contacts and sent the message, their contacts get a text message that is not a group message, so each user gets a text message with a unique link for that specific user. The link takes the contacts to site where they can get to either like, dislike and if they wanted to add comments to the picture that the user sent, letting them know what they think of the picture.

By doing this I am trying to avoid group messages that go in a different direction from the intended goal in why the group message was created for in the first place. The second thing I am trying to avoid is that by avoiding the group message in general, I am trying to decrease the disturbance caused by multiple people commenting in the group message and everyone's phone going off each time a new comment comes in.

If time allows, I would also like to plan out how I am going to market the application once it is finished. This something that my project advisor and I have talked about seeing that the application is almost nearing completion, we would like to have a ready or thought out plan of continuing with the next steps.

Clients

When I set out to create the application MySwag, my idea of the clientele for the app were predominantly females. Starting from where I got the idea to the basic design, it has been a critical driving factor in the process of the developing the application. Now though, having developed and deployed beta tests, I see the applications flourishing in various situations - not only for females but people of all age groups. I could see someone using the app between family members, trying to inquire of their siblings or parents if they like the cake, car, or shoe they intended to buy. I could see it being used between a group of friends trying to decide which picture they should put up in social media, as some friends might like certain group pictures over others MySwag would be extremely helpful as a platform to gauge a small tally to decide upon an issue.

These are just some examples of what I believe would be circumstances in which the application would excel in solving problems. But like all applications, despite the developer's intentions, the end user will ultimately dictate how the app will be used. I believe I should be able to have an open mind towards the clients need and administer flexibility and cater to the evolutionary path my app can grow to become.

Faculty Advisor

My faculty advisor is Dr. Diane Murphy; she is a professor of Information Technology and also the Chair of the Department of Information Technology at Marymount University. She has been in the IT industry for more than three decades working as an analyst, programmer, project manager, vice president, president and CEO of different Information Technology companies. Moreover she is a Certified Software Tester which will

greatly help me in making sure that the application I create is put through robust and rigorous tests that can evaluate and discover all major breakpoints in the app.

Project Plan

Most of the initial work in building the application has been almost done. I had been working on the app for a while now and I am glad to report that there is a working version of the application that can be downloaded. The application however needs more work in my eyes and I have been working on it according to the plan I submitted in the topic submission paper. Below is where I am in my plan

Objective	Progress	Time Spent
Create new feature to take multiple pictures	I have found that this feature will require me to break or heavily modify the structure that has been put in place in the schema-less datatbase (Firebase) and will require a lot of time.	3 weeks
Improve Camera UI and the process of Capturing an Image	I have improved the aspect ratio of the camera proportion so as not to skew the images. I am still working on removing little bugs and hiccups when capturing images	3 weeks
Software Testing • Usability Testing		

• Functionality Testing		

Resources

During the initial phase of building the application there was need for only two primary resources: servers and online storage spaces. Through time, as I dived deeper into the development of the application, I slowly started adding more resources as the need arose- and the application grew more complex. For the application itself I had to decide early on whether or not I was going to build it as a generic application and use platforms like PhoneGap, allowing me to run the app on a number of mobile operating systems. Or if I was going to build the application as a native app. doing so added an additional resource that I was going to need, XCode. Moving on with the project I saw that I needed to use more resources like Firebase and Firebase Hosting, Cloudinary, Twilio and Bitly. Thankfully all of these resources have free tier in their pricing structures except Twilio, it does however have a free trial with certain restrictions which enough to run tests.

Project Details

After doing research I found that I was able to control and use the mobile device best and better user experience when I built a native app for the mobile operating system rather than use a generic one like a mobile web app or PhoneGap (Kohan & Montanez, 2015). So for my platform I decided to use XCode and ObjectiveC/Swift as my programming

language. When using XCode I had to be a member of Apple's Developer Community, which came at an $80 price tag, which the company I worked for, BLEN Corp, was willing to pay. Running through my options for this project I had the option of using website servers and storage space from BLEN Corp, but it wasn't until I started building the application out on XCode that I realized what using my own local servers meant.

Using the local resources I had meant more programming for me, as I had to set up and create an API endpoint for my application to access as well as create a way of communication between the API endpoint and the MySQL database on the servers for storage. This is the time where I had to start researching my options other than building everything manually, although it would have been a great learning experience I wanted to see if I could get the ball rolling on this project. After much research I found Firebase, a service provided online that provide an API endpoint and storage space, it does have different pricing structures but it does have a free tier, which I opted to use. I also chose this service as it provided a schema less database structure which I found would be helpful in the application I am trying to build as it gives me a simple way of adding data to the database. In addition to being simple it also process data faster than relational database (Leavitt, 2010).

Once data had been stored I had to find a way of showing the stored image on a website, this is where after seeing all the options available, I decided to go with Firebase Hosting and use AngularJS as the language to develop the online part of the application which was going to receive, count, and post the likes/dislikes that the users voted on the image that was uploaded.

At first since firebase did not store files but just text I was converting the images and encoding them into Base64 string. After playing around with this format for a while I found that as the images uploaded increased for a user downloading the data to show the user became increasingly slow because downloading the Base64 string then decoding it into a picture was taking a long time to process. After researching on ways to avoid this I found that encoding pictures into Base64 was actually increasing the process time by 33%, so I had to find a way to overcome this problem. I found an online resource called Cloudinary which was an image hosting site that was secure and had a free tier, in addition it had an API endpoint that I could access as well as an SDK for IOS development. After choosing the image hosting platform I changed the data stored in Firebase form a base64 encoded string to the link of the image hosted (Cogswell, 2014; Calhoun's 2011).

The next step involved using Twilio, which another online service that has easy API endpoint that I could use to send the link to the selected users the user wanted to share the image with over text messaging. It is a widely used service and has built a reputable security and dependability (Hardy 2014). But before sending the long link I chose to shorten the link so that it is aesthetically better for the viewers who are going to click on the link and also for generating statistics purposes. To do this I researched the available options and decided to go with Bitly, which provides URL shortening service with an API endpoint that I could use (Parr 2009).

Knowledge being applied to project

The courses that I applied to the project include Object Oriented Programming that I had learned in Java class. Another course that was applied in the building this course was

the software engineering class that I took in the spring semester of 2015. Currently the subject that is helping me in making this project or application gear towards a success is the Software Testing and Quality Assurance class that I am taking now.

Apart from courses that I am taking or have taken, knowledge gained with working with databases, web content manage systems and other programming languages helped me in building the application and getting it to a useable state.

Risk Factors

The risk factors that I have found while doing this project mostly pertain to the security of data. I am using online services such as Firebase, Cloudinary, Twilio and the problem with using these services there is always a risk of data security. Due to this I have taken precautions to avoid data leaks and possible threats of hack. I have used the tools that these services provide as two-way authentication with API Keys and access keys, adding security rules to the database as to who, what, and where the users can write to and many more additional security features that are provided by the services that the application is using.

Work to be Performed by Others

During the development of the application apart from using services offered online like that of Firebase, Cloudinary and the rest, I have also used the SDK's that were provided by these services as add-ons to my XCode project. They provide a platform and syntax style

that makes using the API endpoints they provide very easy, rather than building the session manager myself. The SDK's they provide are updateable and are easy to use and understand.

Other than the SDK's provided by the services, there are small application add-ons that I use in my project, all of which are done by programmers out there and are open source. The reason I chose to use these application, which are called pods, is because it makes my work easier since these pods are things that people use over and over again in most other applications. Simply put "Why reinvent the wheel?"

Resources

Calhoun, D. (2011, August 28). When to Base64 Encode Images (and When Not To). Retrieved October 22, 2015, from http://davidbcalhoun.com/2011/when-to-base64-encode-images-and-when-not-to/

Cogswell, J. (2014, February 10). Cloudinary vs. Blitline: Cloud-Image Services Compared - Dice Insights. Retrieved October 22, 2015, from http://insights.dice.com/2014/02/10/cloudinary-vs-blitline-cloud-image-services-compared/

Hardy, Q. (2014, February 1). When Your App Is in the Cloud. Retrieved October 18, 2015, from http://bits.blogs.nytimes.com/2014/02/01/when-your-app-is-in-the-cloud/?_r=0

Kohan, B., & Montanez, J. (2015, January 26). Native vs Hybrid / PhoneGap App Development Comparison. Retrieved October 22, 2015, from http://www.comentum.com/phonegap-vs-native-app-development.html

Leavitt, N. (n.d.). Will NoSQL Databases Live Up to Their Promise? Retrieved October 22, 2015.

Parr, B. (2009, April 5). 5 Reasons Why URL Shorteners Are Useful. Retrieved October 18, 2015, from http://mashable.com/2009/04/05/url-shorteners/#oYb30bMwuOqX