KeyLime iOS Mobile Application

KeyLime

By Matthew Orgill
June 13, 2019
CPE Senior Project 2019
Advised by John Bellardo

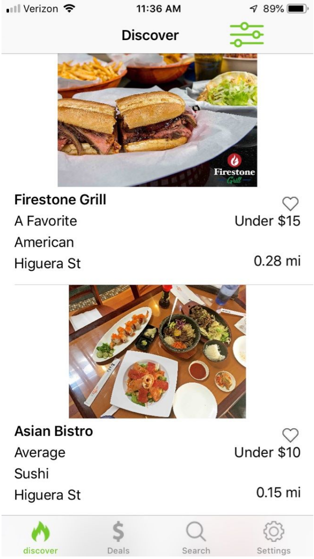
Abstract

This project creates an iOS mobile app geared specifically toward the students of California Polytechnic State University. The app aims to provide the ability for users to discover new restaurants to checkout in the central coast area. These restaurants can be filtered to the user's choosing based on the price of food, rating the restaurant has received, distance away from the user, and type of food. In addition, featured deals that local restaurants currently offer can be found on the app. Each restaurant can be favorited by the user to allow for better filtering of discovering new restaurants and to easily access deals of the user's favorite restaurants. Overall, the app serves to provide a fresh experience to finding delicious and affordable places to eat for college students who have become accustomed to social media communication and aesthetic. This report dives further into how the app was designed and developed as well as decisions that were made throughout the development process.

Introduction

KeyLime is designed to bring a social media vibe to finding trendy and affordable places to eat for college students living in the central coast. In addition, the design of this app includes an emphasis on relaying up to date information about restaurant deals to aid students to be able to afford grabbing a bite to eat. The app is organized by four distinct tabs to choose from with the two main views representing a place to find new restaurants to eat at that other students have enjoyed and a place to find the latest deals offered by restaurants nearby.

The first tab comprises the discover page where "trending" restaurants are displayed for the user to peruse. Here, an image followed up by a few informational pieces about the restaurant are laid out in a tableview for the user to quickly determine whether they would be interested in the restaurant or not. Each restaurant cell can be selected to take the user to a restaurant view. In this view, the user can see more information about the restaurant as well as rate the restaurant themselves and see the deals that restaurant is currently offering. In the top right portion of the screen, a filter button can be found. This button takes the user to another view that allows them to filter out certain restaurants that appear in the discover tab based on price, rating, distance, and type of food. In addition, the user can decide whether their favorite restaurants can appear in the discover view or not. This tab is shown in the screenshot on the right.



Firestone Grill
A Favorite Under $15
American
Higuera St 0.28 mi

Asian Bistro
Average Under $10
Sushi
Higuera St 0.15 mi

The second tab represents the deals tab. At the top of this view, the user can horizontally scroll through deals that their favorite restaurants are currently offering. Below this they can scroll vertically to see other deals that nearby restaurants are offering. All of these deals are organized by expiration date so that deals that will be gone the soonest are seen first. Each deal has a star in its cell to represent whether or not the user has used that deal. To use a deal, the user must present the deal to the restaurant and have the restaurant mark the star for them, which cannot be undone. Each cell for the deals can be clicked on to take the user to a deal view that gives more information about the deal. In this view, the star can actually be marked to signify the use of the deal.

The next tab that can be selected is the search tab. In this tab, the user is presented with a map of their current location. They can search for nearby restaurants that they might want to review or that they have heard of and want to check out its reviews. Restaurants appear as small lime slices on the map. When the user's desired restaurant appears, they can click on the lime icon to view quick facts about the restaurant such as its rating, price, and address. If they want more information about the restaurant or to rate the restaurant, they can hit the "go" button to go to this specific restaurant's view.

Finally, the last tab is the settings tab. For most users, the view displays their name and login email, and it includes a way to sign out or to change their password. In addition, it allows the user to view their favorite restaurants and the restaurants they have rated. The favorite restaurants and rated restaurants buttons here will just show a list of the restaurants the user has favorited or rated and allow the user to go to the individual restaurant views. For admin accounts, however, the user can see a button that allows them to display the admin view. In this view, the admin can edit their own restaurant and add deals for their restaurant.

All data for the app is stored using Google's Firestore. In our database within Firestore, we have collections for users, restaurants, and deals. Each user has fields to keep track of their username, email, whether or not they are an admin, deals they've used, restaurants they've rated, and their favorite restaurants. Each restaurant has information such as its name, location, address, image, price, rating, amount of ratings, deals, type of food, and url. Each deal keeps track of its title, description, expiration date, image, and which restaurant it belongs to. These allow us to keep our app up to date and manage all the data within it. The images are stored as urls to save space and make it easier to store/access them.

The name for the app was one of the more difficult creative tasks we had to tackle. As engineers, we mostly thought of names that related directly to what the app accomplishes but wasn't catchy and sounded boring. One of these boring names we almost chose if we couldn't come up with anything better was "UniEats." Since we were directing our app toward college students and our app is about finding restaurants for food, this made sense combining university and eats. However, this is a very forgettable name and almost too directed to what the app does. Thankfully, Eli came up with a bit of a random name by thinking of Apple and what our app does to come up with KeyLime. Although initially we weren't in love with this name, it grew on us and we stuck with KeyLime as the final name.

The design of the app described above evolved throughout the production of the app; however, the foundation was determined in the beginning. To begin to start designing the core of the app, we first needed to recognize who our audience was as well as checkout other similar apps already available in the app store. Focusing on what our audience would prefer while also differentiating our app from competitors allowed us to change initial ideas and refine the app until we ended with the finished product. Initially, we knew the main design decisions we would have to make included: color scheme, app flow, UI presentation of restaurants and deals, storing information for both restaurants and users, using deals, admin accounts, differentiation from other apps in rating and pricing scheme of restaurants, etc. Although we made most of the design decisions as a group and all helped out in every area of the app, the area of focus I was put in charge of was the UI and ensuring differentiation from competing apps. With this focus, the main design decisions I took charge of included: the discover tab layout, the rating and pricing scheme, and the filtering system.

Requirements

Originally a prototype of the app was going to be presented to restaurants around San Luis Obispo. Due to a slow start in the Winter Quarter and unforeseen difficulties in designing the app, getting restaurants on board and as sort of a client/customer became too much to include in this project. Therefore, we scrapped the plan. In order to make the project still feel complete even without the restaurant owners, we created our own data and restaurants to be displayed in the app. The functionality is still complete for having restaurant owners to provide their own data for their specific restaurants, but we became our own client/customer for the purposes of this project. With this in mind, we created our own requirements. To allow for restaurant owners to handle their own information in the future, we had to come up with a scheme to add admin accounts as well as views where admins can change the information of their restaurant and deals. In addition, having a specific audience of college students came with its own requirements. We needed to create an app that felt closer to a social media app in the way the user can favorite and rate restaurants yet also still conveyed the basic necessary information that users are looking for when determining whether or not to experience a new restaurant.

Background

While no current projects overlapped with this project, previous work and class helped in the understanding and development of the app. All of us took CSC 436 which is the mobile application development class. Josh was able to learn app development in iOS while Eli and I learned using Android. Though we had to learn almost all over again with swift and iOS, some of the Android understanding and techniques definitely made for an easier transition in understanding swift and iOS development in Xcode. The class also helped in our understanding of what was feasible within our own development of this application. In addition, I helped to develop a web app using Ruby on Rails over the previous summer. The app was a type of social media platform that attempted to allow people to stay connected through forming groups. Within the groups, the users could save other users' posts to store and allow for easier access. This helped in understanding how to coordinate the different information our app was attempting to relay as well as storing user information and projecting it throughout the app. In addition, the work over the summer opened my eyes to how to think about the user's/customer's experience with using the app and how we can design the app accordingly.

Key Design Decisions

Initially, we hoped to maintain a balance of separation with who made decisions on certain parts of the app; however, throughout the course of development the roles were a bit blurred as we worked on all parts of the app. I think this ultimately benefited us as we all were constantly in communication and able to create an app that all of us are on board with. Despite this blur of separation, we were still able to make certain design decisions individually. Ultimately, the design decisions I helped push forward with involved the layout of the discover view, the rating and pricing scheme, and the filtering system.

The first design decision I took charge of started before any coding of the app actually occurred. I knew we wanted to create a way to see the basic information of a restaurant in a confined space so that many restaurants could be viewed at once. This display needed to be large enough to fit all of the information of the restaurant but small enough so that there could more than one restaurant viewed on the screen at any one time. Before even coming up with my own design, I researched other apps that are currently in the app store that attempt to accomplish a similar task. Most of the apps that were observed were competing apps such as Yelp and others. Yelp has been around for so long and has done so well, so I looked to model our design after Yelp's design. In addition, I've loved some of the designs that Pocket Points uses in their app, and so I also took inspiration from Pocket Points. This resulted in the tableview cell that is in our current app. Priority was given to the image displayed for the restaurant as inspiration taken from Pocket Points' design, but also the image for the restaurant was given so much room because of our overall goal to make the app have a social media vibe where everything revolves around images.

The image is the primary object users will look at for each restaurant along with the restaurant's basic information. This information includes the restaurant's rating, pricing range, type of food, address, and its distance from the user. The name of the restaurant and then the rating and pricing come directly under the image because these are the most important pieces of information for each restaurant. Then below that are the type of food, address, and distance away information pieces. In addition, there is a heart button included just below the right side of the image to indicate whether the restaurant is one of the user's favorite restaurants or not. These restaurant cells fill the screen, again, to resemble social media apps. While this is how the discover view turned out and how its design drew on inspiration from two different apps, there were many other apps who had great designs but not something that fit our app. I researched through over ten different apps, but ultimately I decided on the designed described above drawing mostly from only two of the apps.

Going hand in hand with the design of the discover view was the design of the rating and pricing scheme. As these are two of the most important pieces of information being displayed about each restaurant, it fit that I took charge of deciding how we went about implementing a scheme for each of them after designing the display of the restaurants in the discover view. Ratings and pricings are nothing new when talking about restaurant apps or honestly with any app that deals with businesses. With all these apps, there seems to be a common theme of using stars to rate restaurants giving them anywhere between a one and a five star rating. These ratings are usually accompanied by a review and sometimes the stars are different images but still remain out of five. In addition, pricings for restaurants are usually categorized with dollar signs. Apps that display prices of restaurants tend to show between one and three dollar signs or sometimes up to four dollar signs. There are a few apps here and there that change up these schemes slightly, but not usually in a better way. However, I wanted to change up the way we displayed ratings and pricings for restaurants in order to stand out and make our app different. We knew going into the project that our app is very similar to Yelp, and so this was especially important to get right and have a different feel than Yelp, specifically a more social media styled feel. With this in mind, I decided that different phrases would be used to describe the rating and pricing of each restaurant. Not only would having phrases change things up but it would also work well with our college student audience to have descriptions of ratings be phrases of something they would say.

At first, I tried to create these phrases on my own, but I soon realized that it would be better to brainstorm ideas through other college aged people. I created a survey that asked students which option they would prefer to represent a certain righting or price and then also included a place for them to suggest their own phrase. The options they chose from were phrases that mostly I came up with but also some from Josh and Eli. After receiving responses back, I combined our own preferences with the majority preferences of the surveys to pick the top phrases. During this survey process, we received positive feedback on the rating phrases but not so much with the pricing phrases. Because of this and my own concern that the current pricing scheme that apps use doesn’t convey enough info, I decided to change up the phrase idea for pricing a bit.

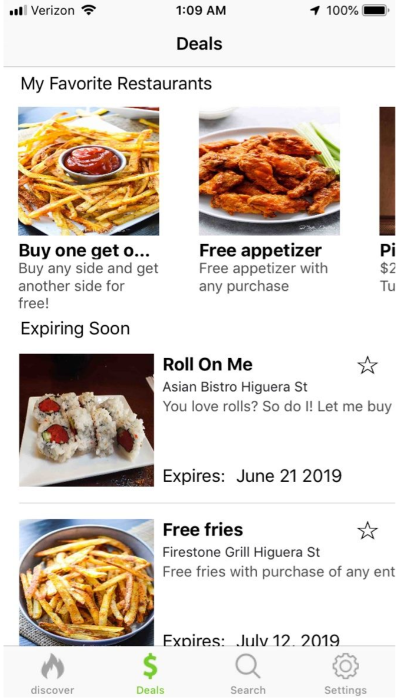
Instead of phrases that don’t convey any specific information about the pricing, I chose to do phrases that describe the average price for each individual item at a restaurant. For instance, at Firestone most entrees and sides cost $9.99 or less. Therefore, for Firestone a phrase of “under $10” would be used for its pricing. This way it isn’t completely different from the ratings phrase scheme, but also it conveys exactly the information a user needs to know what the restaurant’s prices are like. It mixes the two together in a way that best benefits the user.

In addition to using phrases for the ratings, actually rating the restaurant was changed up a bit as well. Instead of requiring a formal review or anything of the sort, users on KeyLime can simply tap a lime to give it the rating they want. When they tap the lime on the far left, it will display the phrase that corresponds to the lowest rating and vice versa for tapping on the lime on the far right. This changes the vibe of rating restaurants from something that can take time and feels formal to one that feels like “favoriting” a photo on social media. It is quick and simple. Also, it helps to make changing a user’s rating of the restaurant quicker. They don’t have to jump through hoops to change their rating, they simply just tap on a different lime and the rating automatically updates.

Working along the same lines of UI design and the user experience, another design decision I made was with creating the filtering system. We wanted the user to have the ability to preference the kinds of restaurants they want to check out with still leaving room for them to see restaurants they’ve never tried before. Also, we wanted to make sure they didn’t have to scroll through too many restaurants at any one time just to find a restaurant they would be interested in trying. With this in mind, I created a simple filtering view for the discover page that allows the user to change the price levels, ratings, and distances for the restaurants that appear in their discover view. In addition, they can choose whether or not to specify a certain type of food as well as choose to allow their favorite restaurants to show up or not. Initially, through setting the filters to the extremes, the filters allow every type of restaurant except for favorites, since the page is meant to discover new restaurants and not so much see restaurants the user already knows about. This allows the user to see all the different kinds of restaurants nearby until they decide to adjust the filter and shrink the range of restaurants they look through. I attempted to dress the view up a bit to make it pop and look cool, but eventually decided that keeping it simple was better. By only having a few words and subtle hints of our keylime green here and there, the view puts the focus on the functionality and makes it clear and simple to the user how they can utilize the view to filter the restaurants to their liking.

In addition to the view that gives the user the ability to filter the restaurants, the discover page has additional filtering that the user doesn’t have control over. To keep the “trending” restaurants at the top, the restaurants are filtered by the recency of their last rating. Essentially, whenever a user rates a restaurant, the restaurant has a field called “lastRating” that is then updated to the current date. This field is then used to determine what restaurants are “trending.” All in all, restaurants that can show up in the view are determined by the user’s choice of filters while the way the restaurants are sorted within the view is determined by the “trending” factor.

As for filtering the deals view, originally I had planned to have a similar filtering scheme as the discover page. However, as we developed the app, I realized that users probably would only want the ability to filter their favorite restaurants to the top so they can see those deals first. Otherwise, the only thing that really matters is the expiration date. Because of this, I decided to rework the deals view and filtering design. Taking inspiration again from the Pockets Point app, we made a collection view that the user can scroll through horizontally at the top of the view and a normal table view that the user can scroll through vertically below. In the collection view, all the deals from the user's favorite restaurants can be found, and the rest of the deals can be found in the table view. Both are sorted by expiration date so that deals that are expiring soon are seen first. It might have been nice to include a filter for changing specific types of foods of the restaurants whose deals are showing; however, one thing we wanted to be consistent in was simplifying over complicating. Adding the filter could just complicate it a bit more and push the app more towards being too similar to Yelp, which is the last thing we wanted. These views can be seen in the screenshot to the right.



Deals

My Favorite Restaurants

Buy one get o...
Buy any side and get another side for free!

Free appetizer
Free appetizer with any purchase

Pi
$2
Tu

Expiring Soon

Roll On Me ☆
Asian Bistro Higuera St
You love rolls? So do I! Let me buy
Expires: June 21 2019

Free fries ☆
Firestone Grill Higuera St
Free fries with purchase of any ent
Expires: July 12, 2019

Overall, I was able to make individual design decisions to help create and develop our app into the product it has become. It was a bit different than most the work I've done before as my decisions mostly related to the user's experience and the overall purpose of our app rather than decisions more directly related to the code. In the end, it allowed me to see the project from a different perspective and gain insight into the full development of a project.

Future Work

As this quarter comes to an end and the project is wrapped up, it is not the end all be all for the app. There were a few things we thought of before starting this project and many more ideas that we came up with along the way that we ended up not having time for. In the end, we decided to complete just the minimal viable product in order to meet the time frame and pass our other classes. Our other ideas that we'd like to complete in the future given enough time includes: teaming up and pitching our app to restaurants, improving the UI, create a more secure and simple method for using deals, and updating the way restaurants are determined to be "trending."

Even before we proposed this app idea for our senior project, there was the idea to eventually make this app available in the app store and obtain support from local restaurants to strengthen our data as well as user pull. Basically make this app a legit product and business. The plan was to create an alpha prototype that would serve as a proof of concept for restaurant owners. We could pitch our idea by showing this alpha prototype. As restaurants chose to partner with us, we would be able to fill our app with legit data of nearby restaurants and their deals as well as continue to improve the app. These restaurants could serve both as a client who could give us their thoughts on how to improve the app with other features and also as users who could tell us what was working well and what needs to be improved. Eventually the hope would be to create a system for us to make a profit from the restaurants. Once we have enough popularity and a strong user base, we could charge restaurants a certain percentage of their profits from every purchase that uses a deal on our app. Obviously, this was best case scenario if everything worked out, but it is definitely something we thought of and would strive for in the future.

One area that we never were able to give enough time to was the general UI for our app. The structure and flow of the app was definitely well thought through and planned out; however, how the app actually looks was never given too much attention. It doesn't help that none of us are very artistically inclined. Before starting the project, I hoped to channel more of my creative side to come up with a fun look for our app, but unfortunately, I wasn't creative enough with the time I was able to give to the app's UI. In the future, I would reach out to some of our graphic design friends to design a few options for our app. With these designs, we could move forward with the one we liked the most and maybe even tweak it a bit if needed to better fit our app.

In addition, the method we currently have for using deals definitely works, but it isn't the best way we could implement the deals system. In the future, we could take a couple different routes. One way that we could change the deal system is to create a button that says "use deal." When the user hits this button, it would generate a warning to confirm that the user really wants to use the deal. After confirming, the user would be shown a unique view that shows a QR code, or any code that the restaurants wants, for the specific deal. Once the user exits the view, they will no longer be able to return to that view for that deal, so basically they will no longer be able to use the deal anymore. This could allow for specific deals to only allow a certain amount of users to use it if a restaurant wanted to specify amount of uses instead of an expiration date.

Finally, the current formula for determining a "trending" restaurant is a bit too simple. It only takes into account the latest rating of a restaurant. Ideally, we could come up with a better formula that takes into account a couple of the latest ratings from a specific restaurant as well as the total amount of ratings it has received. One way I have thought of achieving this is to create another field for the restaurants in the database. This field could be a KeyLime score that takes into account the amount of ratings a restaurant has and produces a single score by multiplying the amount of ratings by a number that is based off the recency of the last rating. I wasn't able to come up with a proper multiplier to make this formula work during the course of this project, but I like to believe it would be a viable option in the future.

Reflections

After completing Capstone, I expected to be able to come up with a solid plan that would allow us to create an almost perfect foundation for our app. What I discovered is that all projects have their own issues, and no matter how much you research and plan in the beginning there will always be things you wish you could go back and do differently once the project comes to a close. I do think each project allows me to grow and learn new things that help me with the next; however, there are definitely many things I would do differently for this project upon reflection. Some of these things I would do differently include: research and plan a better setup for storyboards and podfiles to make collaboration through Github easier, finalize our database before starting to code, and create a schedule I can force myself to stick to.

When jumping into this project with two other people, none of us knew the pitfalls that awaited us with using Xcode and Github. We initially started working on the project with only one storyboard. This quickly made collaborating difficult because two of us would change something on the storyboard at the same time. By doing so, we had to open text files of the storyboard in order to merge the two different versions properly, and even then we weren't able to do it perfectly. Not to mention this took a full day of work to figure out. Eventually we thought better and split the project up into multiple storyboards for each main tab as well as a few extra for restaurants and deals. This allowed us to work on multiple storyboards at once without having merge conflicts. In addition, we initially didn't include our podfiles when pushing and pulling to Github. This made us have to delete our pod folder and reinstall them a few different times. Eventually we pushed these files once we knew we had all the pods we needed. In hindsight, I would make sure to establish solid organization with our app to ensure multiple storyboards to work on. In addition, I would make sure to obtain all the pods we needed at the beginning so that we only had to install them once, or at least only a few times because we might not have been able to know all of the pods we would need at the beginning. I think researching a bit more into helpful tips for collaborating with Github and Xcode would have been the best thing to do.

Similarly, I think we jumped the gun in starting the project and setting up our database rather than taking the necessary time to fully plan out the app and all the structures. Once we had a basic idea of the app from creating our proposal, we started to work on getting a database and setting up a few structures in it to store our data. As we continued to develop the app, we realized we needed to add an extra field here or there. Then we realized we needed to fully change a few fields. These different changes to our database weren't the end of the world; however, they caused us to spend a significant amount of time on re-adjusting our data and app to fit with our database. In addition, I think completely understanding how we wanted to store and manage all of our data would have helped us to plan the development of the app better.

Among all else, having to be disciplined and work on this project on our own terms was one of the most challenging aspects of this project. The project is meant to give you freedom to work on it and complete it as you can which made it convenient; however, I've learned I work much more efficiently with a set schedule. I wish I would have been more diligent in setting aside specific times of specific days to work on this project. Treating those time slots like any other class would have allowed me to consistently work on the project every week. Instead, I set aside only Friday as the day to work on the project. Unfortunately, Friday was also the day when many other reports and projects were due on a week to week basis. Because of this, Friday often became finishing up assignments that were due immediately and pushing off senior project as we had "so much time" to finish it. It didn't help that winter quarter was very difficult and time consuming with completing Capstone and Networks. Then after winter quarter finished, habits were already set with working on the project. This made it difficult to create a new schedule and form better habits for the spring quarter. Eventually I was able to work on the project much more consistently in the spring; however, starting the project with a more consistent schedule in the winter would have made the project go much smoother.

Conclusion

Through much planning, flexibility, and collaborating, KeyLime has become a finished project that we can be proud of. This app started as what seemed to be a simple idea, but soon become much more complex as it started to take form. Ideas that I thought were solid through and through ended up having to be reformed, but that is the lesson of this project. No matter the planning and preparation, there will always be unforeseen hiccups that call for change in the design. These changes can be great or small, but either way it is important to not become hung up on the original plan and be able to move on to what will work for the project at hand. Throughout the past two quarters, KeyLime has been through many changes but ultimately has become a full fledged application that can be pushed towards production. There were quite a few setbacks and pitfalls along the way, but in the end, we persevered and took care to make sure KeyLime worked as well as it could. I’m proud of what we have accomplished in creating this app. I am grateful for the lessons we have learned throughout the entire project. And I am excited to show friends and family what we have created as well as to making it even better in the future.