

PROJECT INFORMATION

NAME OF THE ORGANIZATION: KAJAK

PROJECT TITLE: Wheel of Food

PROJECT SUMMARY:

Objectives:

This app targets people who are trying to find a place to eat quickly and in the least use time as possible. It also caters towards the indecisive people that cannot decide between a category of food or certain restaurants. This app is not intended to be a stand alone, but as an extension to the very popular food app Yelp.

What:

This app will aim to create a user interface that displays a list of preferences that the user wants to input. These are inputted manually or by the meta-tags that we will implement into the software. The app can take in a single meta-tag or many tags that use the union the two or more options. Once the meta-tags have been decided, the app will move the user to another interface that holds the most popular restaurants that fit the meta-tag via the Yelp open API. The user can then add the restaurants to the roulette board up to a later determined maximum. After the user has determined their top restaurants, the app will take the user to another interface that will have the player play a Wheel of Fortune style game that determines the restaurant. The user can also filter the restaurants with the "Open Now?" drop down menu. The game will feature a wheel that has the tagged categories in each slot of the wheel. The first game will be to decide the meta tags of the types of restaurants. After the meta tag has been decided, the second UI will lead the user to another round of the same game except with the top restaurants within the area depending on the meta tag. Once decided, another interface will appear with the restaurant's information via Yelp with the directions accessible through a "Go There Now" button.

Goals:

The goal of this app is to efficiently and effectively determine a food location with an interactive game that allows customization in options and restaurant choices. The app is meant to enhance the quality of life by taking away hindrances such as indecisiveness within a large group of people, since this app allows for the addition of any number of meta-tags. In addition, the user can sign up for an account with the account via their email or Facebook account. This feature will allow users to bookmark restaurants and meta tag combinations, and rate restaurants with a thumbs up, thumbs down system.

PROJECT METHODOLOGY:

Approach:

The project will follow the Agile methodology. We want to avoid Waterfall because it may be time consuming if we're stuck on one part of the project. Agile allows flexibility within our schedules to meet when we can and work when we can as long as we meet the scheduled deadlines. We also want to avoid the Spiral methodology because we want to get the minimum viable product done as soon as possible. We wish to build on our previous tasks, but we do not want to spend too much time on a single portion of the project.

Work Breakdown (schedule/tasks):

The project will take in many factors. There is a need to create several user interfaces such as the mini-game itself, accessing the Yelp API in order to find the restaurants information, communicating with the phone's GPS system as well as finding efficient algorithms to create the searches within Yelp that do not take up time when finding a restaurant.

Work Schedule:

- October
 - Early: Begin planning the structure of the app. Look into class diagrams, flowcharts and UML documentation
 - Late: Begin creating the structure of the UML, removing and adding features that the app may or may not need
- November
 - Early: Begin creating the structure of the class diagrams, removing and adding classes that the app may or may not need
 - Late: Finish UML documentation and focus on class diagrams
- December
 - Early: Finish class diagrams and begin looking into flowcharts for logical purposes
 - Late: Finish flowcharts and begin designing the UI via drawings or taking inspiration from other sources
 - Winter Break: Begin learning Android studio and finalize flowcharts.
- January
 - Early: Finalize Android studio and begin the layout coding of the app.
 - Late: Begin designing and coding the stencil code of app
- February
 - Early: Finalize stencil code. Begin coding logic of app
 - Late: Solidify logic operations and core of app
- March
 - Early: Continue working on logic operations, core of app and begin research and implementing Yelp features into app
 - Late: Finalize core logic of app, logic operations and continue Yelp feature implementation
- April
 - Early: Finish Yelp feature implementation. Begin creation of UI
 - Late: Finalize UI creation

- May
 - Early: Finalize app via unit testing, and overall testing of app
 - Before graduation: Deployment

Individual Roles

Each team member within the group will be given a specialized area of concern. This person, in essence, will become the ambassador for that particular portion of the project. That is not to say that that individual will be solely working on that portion, but will become the *expert* to the materials and content pertaining to that section of the project.

- **Software Design Lead:** This role includes creating the *overall structure* of the app, such as flowcharts, UML, class diagrams and the overall skeleton of the code. In addition, this person will be in charge of maintaining the integrity of the design, meaning that the design that is given will not deviate too far from the original plan.
- **Database Engineer:** The database engineer will be working with the database service that is chosen and making sure that the data that is within the database retains its integrity. In addition, the database engineer will ensure the safety of all data within the database and work closely with the system engineer
- **Systems Engineer:** The systems engineer will be in charge of the backend content of the app. This includes the security of the app as well as ensuring that no parts of the code are subject to hacks or leaks. Most importantly, the systems engineer will be working with the database engineer to ensure security within the database as well as the integrity of the data.
- **Systems Tester:** The systems tester will conduct unit tests on most, if not all, parts of the app. This includes the functionality of the user interface, the database and the backend portions of the app. The systems tester is in charge of making sure that the debugging of the app goes smoothly and that if any parts of the app appear to be buggy that they are brought up immediately for redesigning or fixing.
- **UI Design Lead:** The UI design lead is in charge of mainly creating the user interface of the app as well as laying the foundation of the app. This person will be in charge of giving detailed illustrations of the desired user interface for the team to code and build. In addition, the UI lead will also be the one to create the UI and showing the software design lead the progress of the design. This person will also be working closely with the software design lead to ensure integrity of the app.

PROJECT TIME-FRAME:

Planning:

When planning for this project, we want to create UML diagrams and begin to think of pseudocode for certain parts of the project. The main focus of this project is to create a very friendly UI for the user as well as making it simple and elegant without adding too much “fluff” to try and entice the user to use the app.

Coding:

Because we are creating an app, our team has decided to use Android Studio for creating this app. We originally wished to use Swift for the iPhone, but quickly realized that Android Studio would be a far better choice for this particular project. We want to be able to quickly start coding the project and not have to wait for Apple to give us permission for their app store.

Testing:

When testing this app, we will need only one thing: an Android device. However, we wish to test the app on different brands of Android devices to ensure that our app works for a large variety of Android devices. In particular, there are a few areas of concern with this project that must be tested regularly.

1. Data integrity and security:
 - a. With the addition of accounts into the app, data integrity and security are a must. We must ensure that the data is secure and cannot be accessed through a single press of a button. We must implement security measures that must be tested regularly to ensure that they are up to date and coded to the best of our abilities. We also must ensure that the data is consistent to the user. Inconsistencies with the data will lead to unhappy users and frustrated programmers.
2. UI:
 - a. We must test that the UI, which includes the wheel game, buttons, tags, hyperlinks, etc. are working properly and correctly. Each of these can be tested individually once they have been designed **but** must be tested all together once all the pieces have been connected.
3. Database Querying
 - a. The database must have correct queries and insertion of data to ensure, again, secure and consistent data. This can be done with “dummy” accounts that have expected results and can be tested under certain conditions to ensure that they fall under the expected table. If not, the query must be redesigned.

Deployment:

Naturally, we will be deploying our app to the Google Play Store because we are using Android studio.

PROJECT RISK MANAGEMENT:

Risk Register:

During the course of this project, we will come across many risks that will affect the progress of our project.

1. Series of bugs that cannot be solved
 - a. In order to overcome these bugs, we will have to talk to the team in order to find ways around the bug, whether that be to redesign the section of code that is causing the bug, work through the bug and try to find a solution or get rid of the feature overall if the bug persists in other sections of the project.
 - b. We have to keep in mind our timeline when trying to redesign the solution if we run into problems that cannot be solved in an efficient way. If redesigned, we will have to design a feasible solution without it consuming the allocated time we have to the particular section of the project

2. Inability to finish project
 - a. In the event that the project cannot be completed with the current features, we will remove features we find that are the most time consuming. We wish to implement the minimal viable product, or MVP. After completing the MVP, we want to optimize the code we have but if the optimization causes more bugs as well as leads us to not complete the project. In the instance this occurs, we will not optimize the code and deploy it as the MVP.
3. Inability to collaborate
 - a. In the event that the group members are unable to collaborate, the team leader will bring these concerns to the project manager. However, the team leader will try to suffice and communicate with the team try to fix the miscommunication
4. Change of Yelp API
 - a. In the event that Yelp's API changes from being open source to private, we will need a solution to work around this. The team, in this case, will instead be using Google for searched queries of the desired meta tags that the user decides upon. Once Googled, the first few restaurants that are listed will be chosen instead of Yelp's. The query search will instead omit all Yelp options to avoid using their API and instead use the meta tags and the addition of "restaurant" in the Google query to shall all possible restaurants with that meta tag. The user will click on the restaurant name and be sent to their website, if one is provided.

PROJECT CONTACTS:

Kyle Ho - Software Design Lead and Project Team Leader

James Dinh - Systems Tester

Ashwin Bhosle - UI Design Lead

Alex Han - Database Engineer and Systems Engineer

Kevin Nguyen - Database Engineer and Systems Engineer