1. Abstract

1.1. Project Purpose
What is the purpose of this project? What problems do you want to solve?

- Help people make decision on picking restaurants by picking restaurants randomly.

1.2. Background/Motivation
Why are you interested in doing this project? Have you worked on something similar before?

- I always find myself do not know where to eat after whole day of coding in Siebel although I have some restaurants in mind.

2. Technical Specifications

2.1. Platform: (i.e. Website, iOS App, Arduino hack)

- React Native App

2.2. Programming Languages: (i.e. PHP, Python, Lisp)

- React native

2.3. Stylistic Conventions: (i.e. commenting, naming conventions, camelCase)

- Comment each function
- Naming variable using camelCase

2.4. SDK: (i.e. Pebble SDK, LAMP Stack, Android SDK)

- WebStorm

2.5. IDE: (i.e. Eclipse, Visual Studio)

- WebStorm

2.6. Tools/Interfaces: (i.e. Android phone, Bluetooth tag, Google Chrome)

- Xcode simulator

2.7. Target Audience: (i.e. Botanists, iPhone users)

- Hungry people
- People who find hard to make a decision
- iPhone users

3. Functional Specifications

3.1. **Features**
Several bullet points of what kind of functionality your project will feature. What should the user be able to do?

- The user will be asked to pick several restaurant they want to go by searching and selecting options from api. They can save restaurant they love to his or her list. They can get a random result from what they have selected. They can randomize the result as many time as they want to.

3.2. **Scope of project**
What are some of the limitations of this product?

- Cannot give recommendation on restaurant picking but provide a randomized choice based on what people picked.

4. Timeline:

4.1. Week 1

- Create selecting page, and result page.
- Implement searching bar at top of the screen to select restaurant.
- Do research and find a restaurant api.
- Connect to the api founded.
- Unit Test.

4.2. Week 2

- Display data from api to the interface.
- Store data locally.
- Create favorite page and allow user to favorite restaurants.
- Build layout to be adapted.
- Unit Test.

4.3. Week 3

- Direct user to a webpage which shows decided restaurants information.
- Setting up a login page with Google login enabled.
- Send information about restaurant decision to logged in gmail.
- Allow user to turn off email notification.
- Unit Test.

4.4. Week 4

- Connecting favorite view to restaurant picking.
- Create restaurant info page for selected restaurant.
- Polish UI by making searching page to a gallery view with restaurants image which easy for user to pick.
- Debug.
- Unit Test.

5. **Future Enhancements**

What are some cool tweaks you'd want to make to your product after the core functionality is done? Are you planning to work on it in the future?

- Connect it to a mysql database to manage personal profile and try to develop the app with more social activities.