iHangry - Java Project Proposal

# PROJECT DESCRIPTION

**Purpose**

Most people struggle with deciding where they should go eat lunch. Our application, iHangry, aims to please those who are indecisive in choosing a restaurant when they are hungry. Our application will be a website where hungry users can quickly see a restaurant that is nearby that will fulfill their hunger needs. If the user doesn’t feel like eating at that location, they will be able to easily generate a new place with the click of a button and hopefully will discover local restaurants they haven’t tried before.

**Interface**

Our projects interface will consist mostly of a Google map that will have the location of the restaurant, and restaurant information displayed on it. There will also be a button for the user to click if they want to generate a new restaurant.

**Location**

* **Geolocation**The default method of getting the user's location will be by automatically detecting their geolocation through the browser that they are using. The user will need to confirm access before our application will be able to use their geolocation. Results will be ranked by distance from the user's current location.
* **User Input**If the user declines using the built-in browser geolocation, they will be able to type in the address that they are looking to find restaurants near.

**Stretch Goals**

* **Restaurant Filters**

If we have time, we may add filtering options based on a restaurant category (ie. Japanese, Pizza, Mexican, etc.) or price range ($-$$$$$).

* **Save Previous Restaurants**  
  A user could potentially login and indicate if they’ve eaten at a restaurant recently which would remove it from the list of suggested restaurants for at least a week’s time.

# 

# 

# MOCKUPS



# THIRD PARTY API’S

The APIs that we’ll be using are the [Google Places API](https://developers.google.com/places/) and the [Google Maps API](https://developers.google.com/maps/)

**Google Places API**

The Google Places API allows us to sort places by Place Type. The Place Type that we will be using is *restaurant*.

**Google Maps API**

The Google Maps API allow us to display a map to the user. Facilities for requesting the user's location, displaying that location, and routing the user to a destination are provided as well.

# DATABASE (optional)

An optional database may be implemented as part of a stretch goal. Data we have identified as being of interest include:

* User Identifiers
* User Favorites
* Reviews
* Ratings
* Location of Request

# MILESTONES

* **Week 10: Milestone 1**
  + UI Built
* **Week 11: Milestone 2**
  + Google Maps API integration
* **Week 12: Milestone 3**
  + Google Places API integration
* **Week 13: Project Demo, App Store Submission**
  + App Store Website