Date 01/11/2020 Contributors: Xiaodi Lin and Eliana Suarez

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Final Report

The main objective of this project is gathered information from Airbnb and Google API to find last year’s year historical prices and attractions based on Airbnb zip code. In order to accomplish this objective, we extracted the Airbnb Dataset from <http://insideairbnb.com/about.html#disclaimers> which utilizes public information from the Airbnb website. From this website, we selected two different csv datasets which are the Listing and Calendar.

While the listing Dataset presents snapshot of prices from last year and forecasted prices up to March of this year, the Calendar Dataset contains more detailed information about postings including the availability metrics and the type lodgment offered by the host. Both datasets share a common listing id which we employed to get more detailed information on how prices fluctuate depending on the date. The Calendar Dataset displays the latitude and longitude information for each of the posting not the address or the zip code.

In order to get our second dataset, we made a Google API call to retrieve a json response with specific information regarding attractions near Seattle such as attraction’s name, location, reviews, and open hours. We first retrieved Seattle’s Coordinates as it’s a requirement to search for places nearby. From the json response, we iterated through Json response and created a dictionary where all variables were stored. Once dictionary was created, we used Pandas to create a data frame and saved as csv.

To join the Airbnb and the Attractions csv we had to find a common key between the two Datasets. The foreign key between the Airbnb and Attraction csv was the latitude and longitude. The latitude and longitude allowed us to generate a more general key between the two databases which is the zip code. With Unizipcode, a programmable zip code-database in Python, we were able to transform those coordinates into zip codes. We used the Jupiter notebooks to read the csv, transformed the coordinates into zip code, and then saved it as csv. After, We joined the two databases using the zip code and restructured the order of the columns.

This Dataset can be very valuable for people or companies who have specific lodging arrangements needs. On the corporate side, this information may allow companies who have employees working in Seattle for a special project and might take longer than month to get an idea of prices in locations nearby the respective office and also information of which neighborhood offered the most long term options. On the other side, if a person is traveling solo and that person would be able to save money by spending more in attraction rather than going out to eat, this dataset permits them to get information of neighborhood with the most attraction and most affordable prices.