cOMPETITIONs BETWEEN aIRBNB accommodations and hotels in Los Angeles

# Proposal of cit 550 fINAL pROJECT

Team: Charging Bulls

Team Members: Zhongyin Zhang ([johnianz@seas.upenn.edu](mailto:johnianz@seas.upenn.edu), Github: Johnianzhang),

Xiaofei Huang ([xh2342@seas.upenn.edu](mailto:xh2342@seas.upenn.edu), Github: xh2342),

Dian Gu ([diangu@seas.upenn.edu](mailto:diangu@seas.upenn.edu), Github: DianGu86),

Liang Tang (Email: [tangreb@seas.upenn.edu](mailto:tangreb@seas.upenn.edu), Github: LiangTang888)



The picture is adapted from https://www.vecteezy.com/vector-art/20004043-cartoon-angry-charging-bull-mascot

**Milestone 4**

**ROUTE 1**

**Functionality**: Page Route to Home Page

**Request Path:** “/”

**Request Parameter:**

* This route is a page route and does not require any parameters.

**Query Parameter:**

* This route is a page route and does not have any query parameters.

**Response Parameter:**

* This route is a page route and does not have any response parameters.

**ROUTE 2**

**Functionality**: Page Route to Property Features Page

**Request Path:** “/property\_features”

**Request Parameter:**

* This route is a page route and does not require any parameters.

**Query Parameter:**

* This route is a page route and does not have any query parameters.

**Response Parameter:**

* This route is a page route and does not have any response parameters.

**ROUTE 3**

**Functionality**: Page Route to Host Information Page

**Request Path:** “/host\_info”

**Request Parameter:**

* This route is a page route and does not require any parameters.

**Query Parameter:**

* This route is a page route and does not have any query parameters.

**Response Parameter:**

* This route is a page route and does not have any response parameters.

**ROUTE 4**

**Functionality**: Page Route to Hotel Page

**Request Path:** “/hotels”

**Request Parameter:**

* This route is a page route and does not require any parameters.

**Query Parameter:**

* This route is a page route and does not have any query parameters.

**Response Parameter:**

* This route is a page route and does not have any response parameters.

**ROUTE 5**

**Functionality**: Page Route to Top Listing Page

**Request Path:** “/top\_listing”

**Request Parameter:**

* This route is a page route and does not require any parameters.

**Query Parameter:**

* This route is a page route and does not have any query parameters.

**Response Parameter:**

* This route is a page route and does not have any response parameters.

**ROUTE 6**

**Functionality**: The route uses a GET() method to retrieve numbers of listings in different price ranges filtered by input property features.

**Request Path:** “/search\_features”

**Request Parameter:**

* roomType[String]: type of room
* accomodates[String]: a range for the number of accommodates
* bedrooms[String]: a range for the number of bedrooms
* beds[String]: a range for the number of beds

**Query Parameter:**

* roomType[int]: index of room type in the table
* accomodatesLow[int]: the minimum number of accommodates
* accomodatesHigh[int]: the maximum number of accommodates
* bedroomsLow[int]: the minimum number of bedrooms
* bedroomsLow[int]: the maximum number of bedrooms
* bedsLow[int]: the minimum number of beds
* bedsHigh[int]: the maximum number of beds

**Response Parameter:**

* data[[List[int]]: numbers of listings in different price ranges filtered by input property features.

**ROUTE 7**

**Functionality**: The route uses a GET() method and returns the total number of retrieved listings filtered by input property features.

**Request Path:** “/search\_features\_count”

**Request Parameter:**

* roomType[String]: type of room
* accomodates[String]: a range for the number of accommodates
* bedrooms[String]: a range for the number of bedrooms
* beds[String]: a range for the number of beds

**Query Parameter:**

* roomType[int]: index of room type in the table
* accomodatesLow[int]: the minimum number of accommodates
* accomodatesHigh[int]: the maximum number of accommodates
* bedroomsLow[int]: the minimum number of bedrooms
* bedroomsLow[int]: the maximum number of bedrooms
* bedsLow[int]: the minimum number of beds
* bedsHigh[int]: the maximum number of beds

**Response Parameter:**

* count[int]: the total number of retrieved listings filtered by input property features.

**ROUTE 8**

**Functionality**: The route uses a GET() method and returns the percentage of the total listing that is retrieved after filtering the data with input property features.

**Request Path:** “/search\_features\_percentage”

**Request Parameter:**

* roomType[String]: type of room
* accomodates[String]: a range for the number of accommodates
* bedrooms[String]: a range for the number of bedrooms
* beds[String]: a range for the number of beds

**Query Parameter:**

* roomType[int]: index of room type in the table
* accomodatesLow[int]: the minimum number of accommodates
* accomodatesHigh[int]: the maximum number of accommodates
* bedroomsLow[int]: the minimum number of bedrooms
* bedroomsLow[int]: the maximum number of bedrooms
* bedsLow[int]: the minimum number of beds
* bedsHigh[int]: the maximum number of beds

**Response Parameter:**

* percentage[float]: the percentage of the total listing that is retrieved after filtering the data with input property features.

**ROUTE 9**

**Functionality**: The route uses a GET() method to retrieve numbers of listings in different price ranges filtered by input host information.

**Request Path:** “/search\_host\_info”

**Request Parameter:**

* responseTime[String]: response time of a particular host
* responseRate[String]: response rate of a particular host
* acceptanceRate[String]: acceptance rate of a particular host
* superhost[String]: whether the host is a superhost
* totalListing[String]: a range of total listing a particular host has

**Query Parameter:**

* responseTimeLow[int]: the minimum response time of a particular host
* responseTimeHigh[int]: the maximum response time of a particular host
* responseRateLow[float]: the minimum response rate of a particular host
* responseRateHigh[float]: the maximum response rate of a particular host
* acceptanceRateLow[float]: the minimum acceptance rate of a particular host
* acceptanceRateHigh[float]: the maximum acceptance rate of a particular host
* superhost[int]: whether the host is a superhost [1/0]
* totalListingLow[int]: the minimum total listing a particular host has
* totalListingHigh[int]: the maximum total listing a particular host has

**Response Parameter:**

* data[[List[int]]: numbers of listings in different price ranges filtered by input host information

**ROUTE 10**

**Functionality**: The route uses a GET() method and returns the total number of retrieved listings filtered by input host information.

**Request Path:** “/search\_host\_info\_count”

**Request Parameter:**

* responseTime[String]: response time of a particular host
* responseRate[String]: response rate of a particular host
* acceptanceRate[String]: acceptance rate of a particular host
* superhost[String]: whether the host is a superhost
* totalListing[String]: a range of total listing a particular host has

**Query Parameter:**

* responseTimeLow[int]: the minimum response time of a particular host
* responseTimeHigh[int]: the maximum response time of a particular host
* responseRateLow[float]: the minimum response rate of a particular host
* responseRateHigh[float]: the maximum response rate of a particular host
* acceptanceRateLow[float]: the minimum acceptance rate of a particular host
* acceptanceRateHigh[float]: the maximum acceptance rate of a particular host
* superhost[int]: whether the host is a superhost [1/0]
* totalListingLow[int]: the minimum total listing a particular host has
* totalListingHigh[int]: the maximum total listing a particular host has

**Response Parameter:**

* count[int]: the total number of retrieved listings filtered by input host information

**ROUTE 11**

**Functionality**: The route uses a GET() method and returns the percentage of the total listing that is retrieved after filtering the data with input host information.

**Request Path:** “/search\_host\_info\_percentage”

**Request Parameter:**

* responseTime[String]: response time of a particular host
* responseRate[String]: response rate of a particular host
* acceptanceRate[String]: acceptance rate of a particular host
* superhost[String]: whether the host is a superhost
* totalListing[String]: a range of total listing a particular host has

**Query Parameter:**

* responseTimeLow[int]: the minimum response time of a particular host
* responseTimeHigh[int]: the maximum response time of a particular host
* responseRateLow[float]: the minimum response rate of a particular host
* responseRateHigh[float]: the maximum response rate of a particular host
* acceptanceRateLow[float]: the minimum acceptance rate of a particular host
* acceptanceRateHigh[float]: the maximum acceptance rate of a particular host
* superhost[int]: whether the host is a superhost [1/0]
* totalListingLow[int]: the minimum total listing a particular host has.
* totalListingHigh[int]: the maximum total listing a particular host has.

**Response Parameter:**

* percentage[float]: the percentage of the total listing that is retrieved after filtering the data with input host information.

**ROUTE 12**

**Functionality**: The route uses a GET() method to retrieve top rankings for the input review type

**Request Path:** “/top\_ranking”

**Request Parameter:**

* reviewType[String]: type of review that will be used to sort the available listings
* listingSize[int]: number of top listings to return from the query

**Query Parameter:**

* reviewType[int]: the index of the review type
* listingSize[int]: number of top listings to return from the query

**Response Parameter:**

* data[list[Object]]: a list of top listings sorted by the input review type with a size of input listing size

**ROUTE 13**

**Functionality**: The route uses a GET() method to retrieve information for a particular listing with listing\_id

**Request Path:** “/top\_listing/:listing\_id”

**Request Parameter:**

* Param/type/description
* Listing\_id[int]: a unique listing id that can be used to retrieve the associated listing information

**Query Parameter:**

* Listing\_id[int]: a unique listing id that can be used to retrieve the associated listing information

**Response Parameter:**

* Param/type/description
* Listing[Object]: a listing object that contains information about the listing location, neighborhood, price, and scores for various review metrics.

**ROUTE 14**

**Functionality**: The route uses a GET() method to retrieve both Airbnb and hotel data in the same neighborhood and perform cross-analysis on the price and review scores before providing recommendations on the accommodation.

**Request Path:** “/rec”

**Request Parameter:**

* Neighborhood[String]: the neighborhood that the user would like to accommodate

**Query Parameter:**

* Neighborhood[String]: the neighborhood that the user would like to accommodate

**Response Parameter:**

* better\_rating[String]: recommendation for a better rating between ‘Airbnb’ and ‘Hotel’
* better\_price[String]: recommendation for a better price between ‘Airbnb’ and ‘Hotel’