

# Component Specification & Project Plan

## Components

### Visualization:

**Description:** The visualization is the interface the user uses when deciding if they are using the Guest Visualization or the Host Visualization. Uses a bash command to run and open

**Interactions:** The visualization uses a dataframe that is constructed by the data gathering and cleaning modules, and the results of the price prediction model.

#### bokeh\_plot.py

**Description:** User interface for the Guest to view the price listings and Hosts to estimate the listing price for their listing.

**Inputs:** cleaned listings file

**Outputs:** data visualization and user interface

### Data Gathering and Cleaning:

**Description:** A group of modules used to read in data from [Inside Airbnb](#) and clean and aggregate the data into forms that the machine learning model can use.

**Interactions:** These modules are used to create a cleaned dataframe that can be used by the machine learning model and the visualization/user interface

#### convert\_to\_matrix.py

**Description:** Converts dataframe into numerical array to be used in our model

**Inputs:** Cleaned dataframe of airbnb listings

**Outputs:** Numerical array of features and array of prices

#### get\_calendar\_summary.py

**Description:** creates aggregate prices for each listing by weekday price, weekend price, and seasonal price (fall, spring, summer, winter)

**Inputs:** a dataframe of calendar information

**Outputs:** a dataframe and .csv with average price by weekday, weekend, summer, winter, spring and fall

#### get\_cleaned\_listings.py

**Description:** cleans listing data pulled from [Inside Airbnb](#), and splits the combined amenities column into individual amenities columns

**Inputs:** listings data

**Outputs:** a cleaned dataframe and a .csv file

#### get\_data.py

**Description:** used to read in datasets off of [Inside Airbnb](#)  
**Inputs:** city, state abbreviation, country, date of data compilation (yyyy\_mm\_dd), and filename.  
**Outputs:** a dataframe

#### sentiment.py

**Description:** performs a sentiment analysis on the reviews of the Airbnb listings  
**Inputs:** a dataframe of the listings reviews  
**Outputs:** a dataframe and .csv of the mean, variance and count of the reviews by listing

#### transform\_input.py

**Description:** Uses boxcox transformation on the input to be fed into the model  
**Inputs:** Data of the listings in array form  
**Outputs:** Data of the listings in array form boxcox transformed

#### zillowbnb.py

**Description:** main module to run to read in data, identify specific columns to use in the predicting price model, run cleaning procedures, and sentiment analysis.  
**Inputs:** city, state abbreviation, country, date of data compilation (yyyy\_mm\_dd), and filename  
**Outputs:** a merged dataset of the sentiment analysis, cleaned listings data, and price data

## Price Prediction Model:

**Description:** Machine Learning model used to predict listing prices of Airbnb listings  
**Interactions:** requires a cleaned dataset that is created from running the data cleaning modules

#### price\_prediction.py

**Description:** Creates price predictions off of the listing data provided  
**Inputs:** Data of the listings in array form and name of the city  
**Outputs:** Array of predicted prices

# Project Plan

## Week 1:

- Visualization tool Tech Review -
  - Create preliminary visualizations for each technology
  - Dash, Tableau, Bokeh
- Clean Airbnb datasources
  - listings
  - calendar
  - reviews
- Import data sources into format for model

## Week 2:

- PCA (Feature Selection) for Listings Data
- Simplify Calendar Data
- Perform Sentiment Analysis
- More Bokeh - commit initial work
- Begin creating unit tests
  - Input file matches the correct criteria
  - Valid address input from host
  - Typing filter on visualization returns error when invalid value
- Create model and define features users can filter by
  - Validation
  - Start visualizations and user filters

## Week 3:

- Update Bokeh visualizations using features selected by PCA
- Convert user input into array for model
- Finalize model and store model coefficients
- Run evaluation metric on listing dataset: Good, bad, meh metric (Too Low (-50%), Low (-25%), Average( +-10%), High (+25%) , Too High (+ 50%)) Probably not these values, but something like this. Don't really know until we get model running
- Begin drafting Final Presentation
  - Future work: increase scope (other cities)

## Week 4:

- Final Presentation dry run
- Complete user test
- Clean up git repo and finalize Functional Specifications and Component Specifications

## Week 5:

- Final presentation