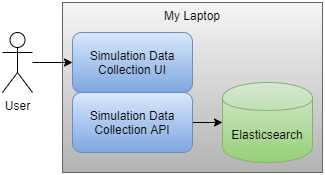
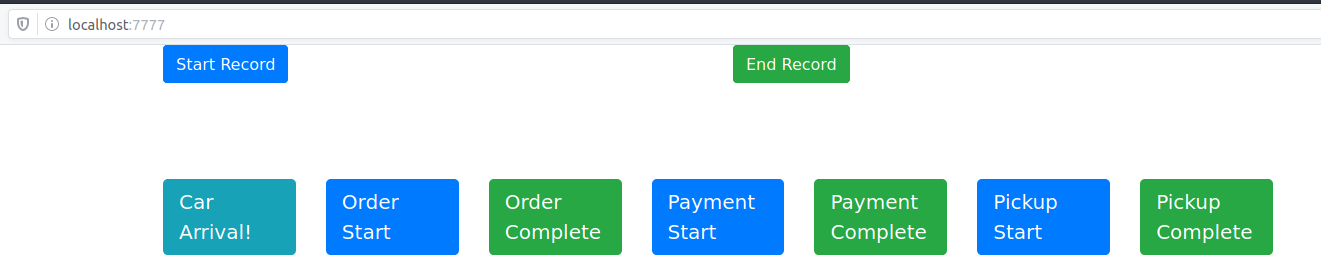
## Data Collection

As I was performing this alone and time was critical, I knew that recording information needed to be streamlined. I developed a system for recording the events that could be stored within a document store (Elasticsearch) for later extraction into data sets for submissions.



The allowed me to find a location where I could see traffic entering the parking lot, the order station, and both windows. The UI with my laptops touch screen meant I could focus on observing without my efforts to record interfering, simply tapping when each event was spotted.



The mechanisms support these can be found within my projects github repository under the “gathering” directory. Instructions for building/usage can be found within the notes.txt within.

# Arrivals

## Arrival Data Preparation

* Threw out data sets in XLSX files (just to save the time spent converting into CSV files)
* Arrival Data Curation
  + Cleaned data so that each source contained only the listing of timestamps (for easier programmatic iterations through each file’s content
  + Some data sets contained multiple columns of data the appeared to have a potential gap
    - Broken into separate files
    - Ex: Hill\_Somayire data set
  + Made some assumptions above the data represented
    - Ex: Jasinski data set contained values formatted to what looked like HH:MM:SS, but the content was more likely MM:SS:00 (unless they monitored the restaurant over its slowest 15 hour timeframe!). For these I just converted the MM:SS values to HH:MM:SS to match expected format
  + Excluded some data sets based on their time for recording arrivals
    - Ex: The Richards\_Teeters and Wical data sets only contained a handful of arrival times within their data sets compared to the normal 15-20.
    - Ex: Removed an entry from Ogletree\_Navuluri data set which contained the mythic 66th second within a minute
  + Within each data set, I use each adjacent timestamp to determine how many minutes between the arrivals. This list of “times between arrivals” are placed within an array and additional statistical properties are recorded for each value

## Arrival Data

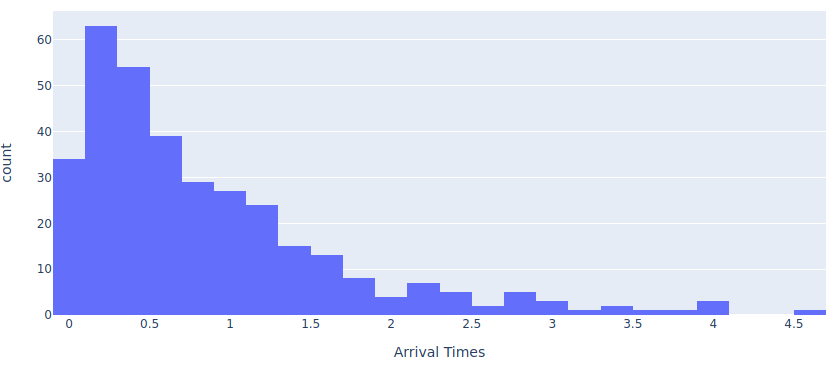
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Mean | Median | Std Dev | Min | Max | Source |
| 0.88 | 0.63 | 0.92 | 0.02 | 3.95 | Arrival\_Hill\_Somayire2.csv |
| 0.79 | 0.63 | 0.7 | 0.05 | 3.23 | Arrival\_Young.csv |
| 0.77 | 0.63 | 0.66 | 0.02 | 2.4 | Arrival\_Mitchell\_Gotteti.csv |
| 0.75 | 0.36 | 0.69 | 0.12 | 2.25 | Arrival\_Means.csv |
| 0.96 | 0.47 | 1.34 | 0.02 | 5.1 | Arrival\_Goetz.csv |
| 0.76 | 0.58 | 0.66 | 0.02 | 2.08 | Arrival\_Hill\_Somayire.csv |
| 0.85 | 0.6 | 0.8 | 0.08 | 3.37 | Arrival\_Woods\_Inman.csv |
| 1.04 | 0.6 | 0.99 | 0.07 | 3 | Arrival\_OgletreeNavuluri.csv |
| 1.67 | 0.97 | 2.03 | 0.08 | 10.55 | Arrival\_Read.csv |
| 0.83 | 0.57 | 0.73 | 0.03 | 2.73 | Arrival\_Marcum.csv |
| 1.1 | 1.02 | 0.81 | 0.33 | 3.73 | Arrival\_Hall.csv |
| 0.66 | 0.49 | 0.55 | 0 | 2.3 | Arrival\_Pettit.csv |
| 0.8 | 0.47 | 0.78 | 0.02 | 3.03 | Arrival\_Cheerala |
| 0.65 | 0.47 | 0.52 | 0.07 | 1.82 | Arrival\_Jasinski.csv |
| 0.94 | 0.33 | 1.15 | 0.02 | 3.58 | Arrival\_Mannam\_Susarapu.csv |

The remaining data has resulted in a mostly consistent collection of data sets. There are some outliers (i.e., Pettit’s instantaneous arrival time and Read’s nearly 11-minute gap in arrivals), but the overall collection appears safe to proceed with combining these into the overall arrival dataset.

When viewing the overall dataset, the 0 second arrival time isn’t so far-fetched (as there are many entries with similar values.) The singular 10.55-minute entry from the Read data set, however, is clearly an outlier as the nearest values are in the area of 4 to 5 minutes. To this, I’m removing the timestamps from the Read data set that causes this discrepancy, as the remaining data set appears sound. To support this, I’m splitting the Read data set into two sets using the two timestamps that generate the 10.55-minute delay as the pivot.

### Combined Arrivals Data Set

| Mean | Median | Std Dev | Min | Max |
| --- | --- | --- | --- | --- |
| 0.86 | 0.6 | 0.85 | 0.0 | 5.1 |



# Orders

## Order Data Preparation

* Threw out data sets in XLSX files (just to save the time spent converting into CSV files)
* Order Data Curation
  + Cleaned data so that each source contained only the listing of timestamp pairs (for easier programmatic iterations through each file’s content
  + Some data sets contained multiple columns of data the appeared to have a potential gap
    - Broken into separate files
    - Ex: Hill\_Somayire data set
  + Made some assumptions above the data represented
    - Ex: Jasinski data set contained value pairs indicating what appears to be timestamp 0:0:0 to MM:SS:00. I’m maintaining the assumption that these indicate that each customer arrives at relative time 0:0:0 and completes their order at the second value. Converting second column to match HH:MM:SS and keeping the values as they appear consistent with other data set’s times.
  + Excluded some data sets completely had unclear data
    - Ex: The Rupert data set contains a listing of timestamps that may be a row with an arrival timestamp, followed by a row with a completion timestamp, but the data granularity is by minutes and not seconds. This will have many orders taking 0 time within my analysis.

# Payments

## Payment Data Preparation

* Threw out data sets in XLSX files (just to save the time spent converting into CSV files)
* Payment Data Curation
  + Cleaned data so that each source contained only the listing of timestamp pairs (for easier programmatic iterations through each file’s content
  + Some data sets had trailing timestamp pairs (like within my data set!) where the experimental observation completed before the customer finished ordering. I’m curious why all other data sets had completed pairs. Did they wait for all stations to completely empty out? Did they stop recording new arrivals for each station after a certain time to ensure they had completed pairs? Either way, I’ve removed my unmatched pairs.
  + Some data sets contained multiple columns of data the appeared to have a potential gap
    - Broken into separate files
    - Ex: Hill\_Somayire data set
  + Made some assumptions above the data represented
    - Ex: Jasinski data set contained value pairs indicating what appears to be timestamp 0:0:0 to MM:SS:00. I’m maintaining the assumption that these indicate that each customer arrives at relative time 0:0:0 and completes their payment at the second value. Converting second column to match HH:MM:SS and keeping the values as they appear consistent with other data set’s times.
  + Excluded some data sets completely had unclear data
    - Ex: The Means data set contains a list of numbers that might represent the exact moment that payment occurred, but I’m omitting the data set for the sake of timeliness
    - Ex: The Rupert data set contains a listing of timestamps that may be a row with an arrival timestamp, followed by a row with a completion timestamp, but the data granularity is by minutes and not seconds. This will have many orders taking 0 time within my analysis.
    - Ex: Some timestamps in the Woods\_Inman data set followed an HH:MM:SS format, while others only appeared to be HH:MM (or perhaps MM:SS?) either way, the unclear determination rendered this data set overly complicated to curate.

# Pickup

## Pickup Data Preparation

* Threw out data sets in XLSX files (just to save the time spent converting into CSV files)
* Pickup Data Curation
  + Cleaned data so that each source contained only the listing of timestamp pairs (for easier programmatic iterations through each file’s content
  + Some data sets had trailing timestamp pairs (again, this was found in my data set) where the experimental observation completed before the customer finished ordering. I’ve removed my unmatched pairs from this data set.
  + Some data sets contained multiple columns of data the appeared to have a potential gap
    - Broken into separate files
    - Ex: Hill\_Somayire data set
  + Made some assumptions above the data represented
    - Ex: Jasinski data set contained value pairs indicating what appears to be timestamp 0:0:0 to MM:SS:00. I’m maintaining the assumption that these indicate that each customer arrives at relative time 0:0:0 and completes their pickup at the second value. Converting second column to match HH:MM:SS and keeping the values as they appear consistent with other data set’s times.
  + Excluded some data sets completely had unclear data
    - Ex: The Means data set contains a list of numbers that might represent the exact moment that pickup occurred, but I’m omitting the data set for the sake of timeliness
    - Ex: The Rupert data set contains a listing of timestamps that may be a row with an arrival timestamp, followed by a row with a completion timestamp, but the data granularity is by minutes and not seconds. This will have many orders taking 0 time within my analysis.
    - Ex: Some timestamps in the Woods\_Inman data set followed an HH:MM:SS format, while others only appeared to be HH:MM (or perhaps MM:SS?) either way, the unclear determination rendered this data set overly complicated to curate.