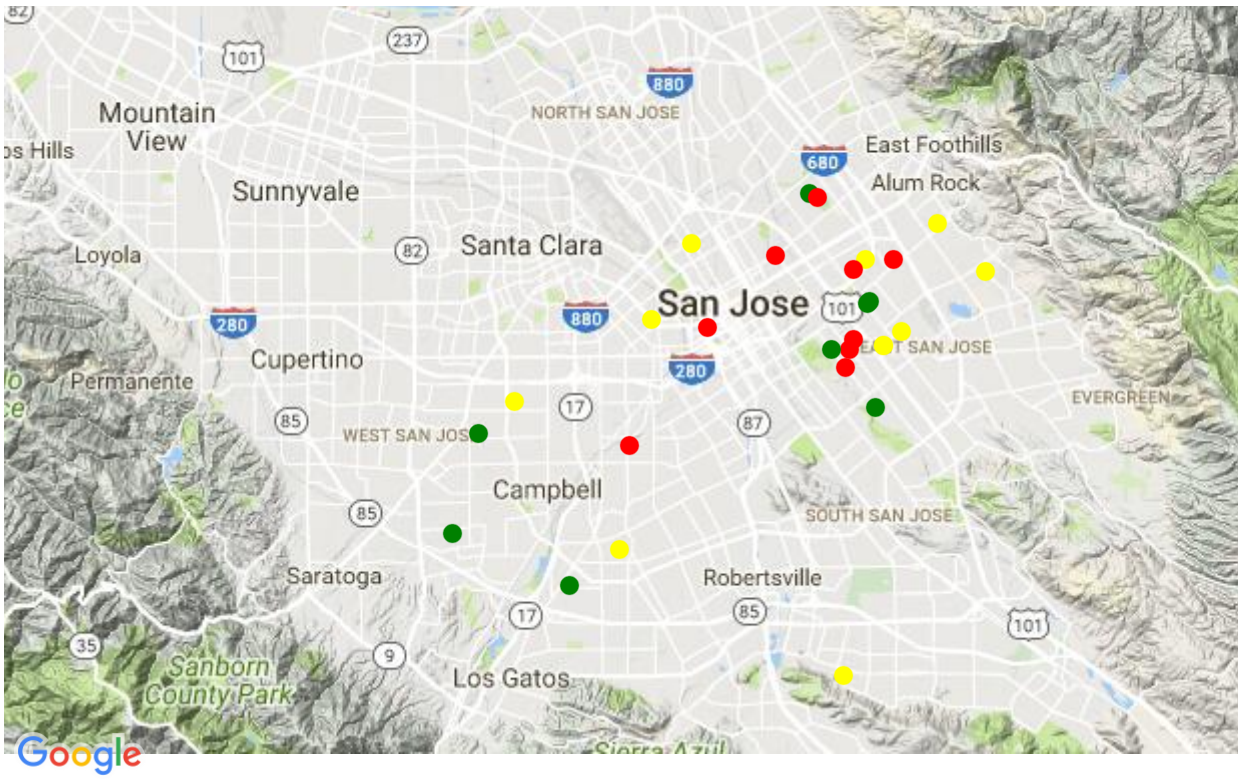
In [1]:
```
import pandas as pd
import numpy as np
import os
import gmaps
import gmaps.datasets
```

In [2]:
```
gmaps.configure(api_key="")
```

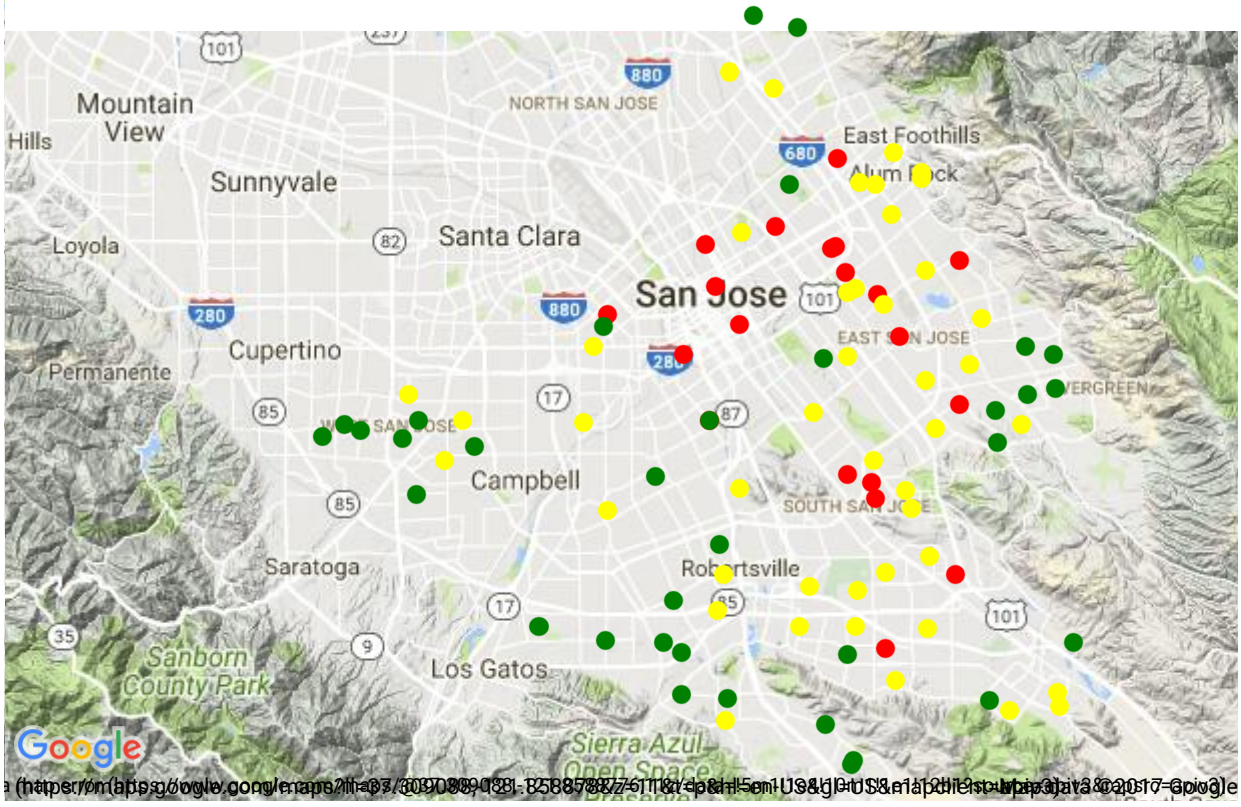
In [3]:
```
charter_df = pd.DataFrame(pd.read_csv("SJcharter_colors.csv"))
charter_df.dtypes
```

Out[3]:
```
Unnamed: 0          int64
address            object
city               object
district           object
districtId        float64
districtNCESId    float64
enrollment        float64
fax                object
gradeRange         object
gsId                int64
gsRating            int64
colorRating        object
lat               float64
lon               float64
name               object
ncesId            float64
overviewLink       object
parentRating      float64
phone              object
ratingsLink        object
reviewsLink        object
schoolStatsLink    object
state              object
type               object
website            object
dtype: object
```

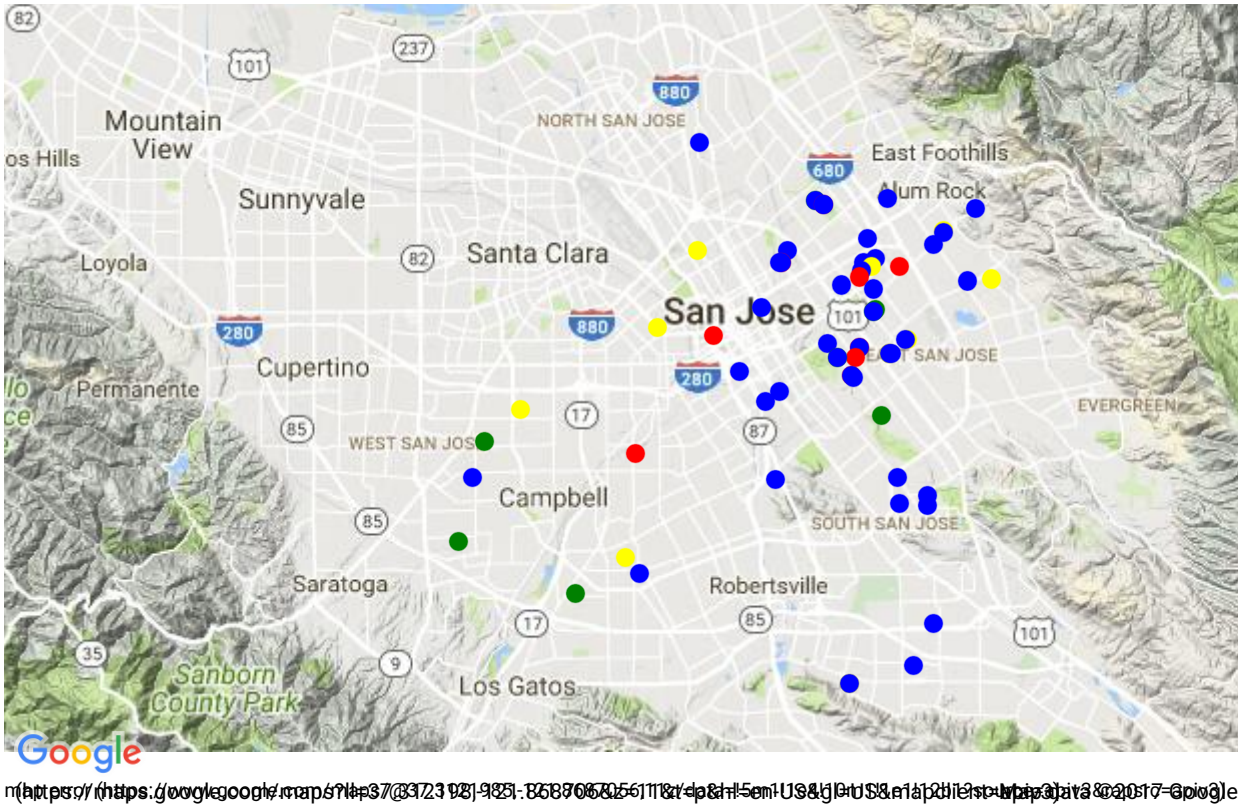
In [4]:
```
locations = charter_df[["lat", "lon"]]
weights = charter_df["gsRating"]
colors = []

for rating in charter_df["gsRating"]:
    if rating <= 3:
        colors.append('red')
    elif rating >3 and rating <7:
        colors.append('yellow')
    else:
        colors.append('green')

fig = gmaps.figure()
#fig.add_layer(gmaps.heatmap_layer(locations, weights=weights))
fig.add_layer(gmaps.symbol_layer(locations, fill_color=colors, stroke_color
fig
```



In [5]:
```
public_df = pd.DataFrame(pd.read_csv("SJpublic_color.csv"))
plocations = public_df[["lat", "lon"]]
pweights = public_df["gsRating"]
pcolors = []

for rating in public_df["gsRating"]:
    if rating <= 3:
        pcolors.append('red')
    elif rating >3 and rating <7:
        pcolors.append('yellow')
    else:
        pcolors.append('green')

pfig = gmaps.figure()
pfig.add_layer(gmaps.symbol_layer(plocations, fill_color= pcolors, stroke_c
pfig
```



In [7]:
```
opencharter_df = pd.DataFrame(pd.read_csv('top5CA_OpenSchools.csv'))


openchartersj_df = opencharter_df.loc[opencharter_df["City"]=="San Jose"]

clocations = openchartersj_df[["Latitude","Longitude"]]

# allcfig = gmaps.figure()
fig.add_layer(gmaps.symbol_layer(clocations, fill_color= "blue", stroke_colo
fig
```



In [ ]: