In [13]:
```
import pandas as pd
import os
import gmaps
import gmaps.datasets
import requests
import json
```

In [14]:
```
file_one = os.path.join('Santa_Clara_Home_Appreciation2.csv')
gmaps.configure(api_key="")
url='https://moto.data.socrata.com/resource/wrmr-tdyp.json'

sheriff_reports=requests.get(url).json()
crime_df=pd.DataFrame(sheriff_reports)
print(len(crime_df))
```

```
1000
```

In [15]:
```
file_one_df = pd.read_csv(file_one)
file_one_df = pd.read_csv(file_one, encoding="utf-8")
```

In [16]:
```
file_one_df.head()
```

Out[16]:

	Unnamed: 0	RegionID	City	State	Metro	CountyName	SizeRank	2007-01	2008-01	2009-
0	256	95123	San Jose	CA	San Jose	Santa Clara	257	668200	613200	5101
1	352	95035	Milpitas	CA	San Jose	Santa Clara	353	654200	630600	5318
2	379	95051	Santa Clara	CA	San Jose	Santa Clara	380	702400	677300	5950
3	404	95014	Cupertino	CA	San Jose	Santa Clara	405	1042400	1124300	10501
4	457	94087	Sunnyvale	CA	San Jose	Santa Clara	458	924600	977000	9190

5 rows × 22 columns

In [17]:
```
home_apprec_df=file_one_df[['LAT','LNG','Appreciation','Scale']]
home_apprec_df.head(5)
```

Out[17]:

	LAT	LNG	Appreciation	Scale
0	37.244238	-121.831604	82.544040	3
1	37.444349	-121.868859	81.731368	3
2	37.348640	-121.984373	95.354178	5
3	37.306491	-122.080640	78.438881	3
4	37.351529	-122.036959	89.630951	4

In [18]:
```
crime_locations=crime_df[['latitude','longitude']]
crime_locations.head(2)
```

Out[18]:

	latitude	longitude
0	37.3007963298659	-122.039332874362
1	37.2463001073712	-121.768668994571

In [19]:
```
locations = home_apprec_df[["LAT","LNG"]]
scale = home_apprec_df["Scale"].tolist()
print(type(crime_locations))
fig = gmaps.figure()
symbol_layer = gmaps.symbol_layer(locations, scale=scale, stroke_color="Red'
fig.add_layer(symbol_layer)
crime_locations=crime_df[['latitude','longitude']].apply(pd.to_numeric)
crime_layer=gmaps.symbol_layer(crime_locations,stroke_color='Green', scale=2
fig.add_layer(crime_layer)
```

```
<class 'pandas.core.frame.DataFrame'>
```

In [20]:
```
gmaps.heatmap_layer(crime_locations)

print(len(crime_locations))


#crime_locations.convert_objects(convert_numeric=True)
#pd.to_numeric(crime_locations)
```

```
1000
```

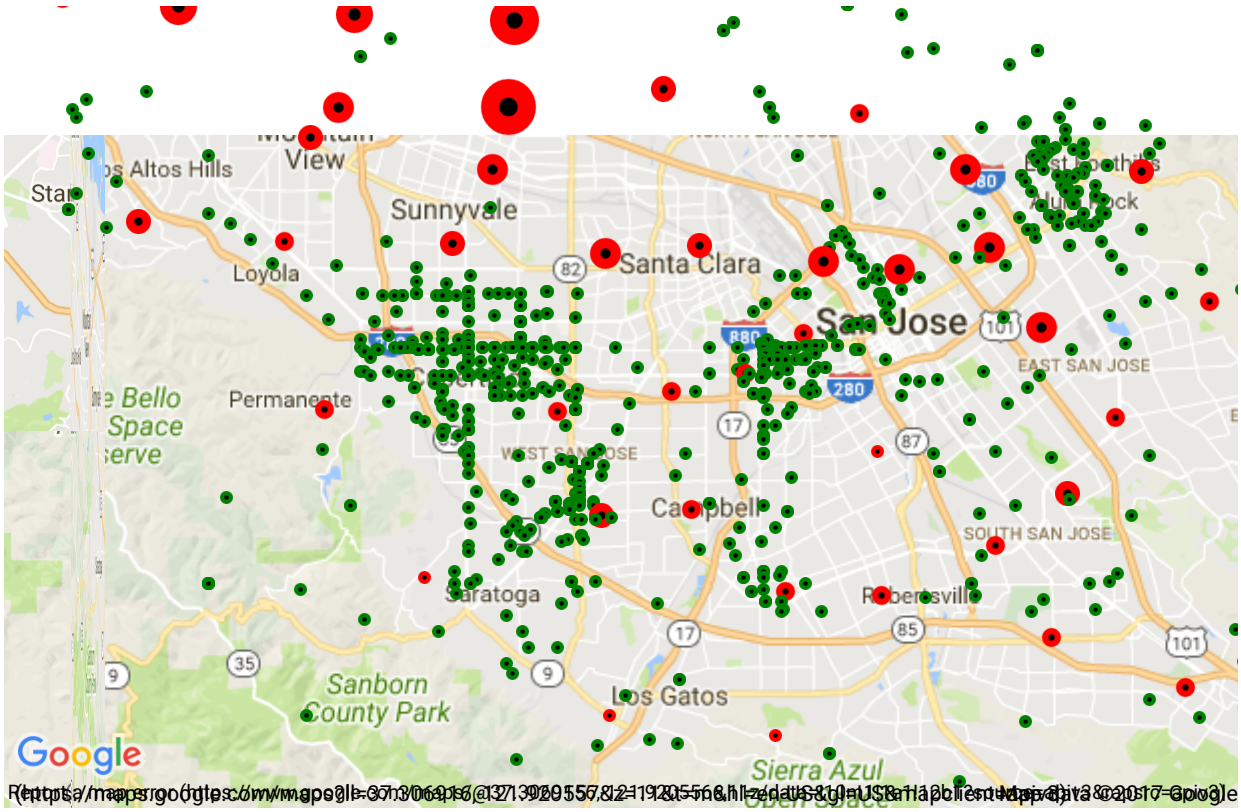
In [21]:
```
charter_df = pd.DataFrame(pd.read_csv("SJcharter_colors.csv"))
charter_df.dtypes
```

Out[21]:
```
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 [22]:
```
# public_df = pd.DataFrame(pd.read_csv("SJpublic_color.csv"))
# locations = public_df[["lat", "lon"]]
# weights = public_df["gsRating"]
# colors = []

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


# fig.add_layer(gmaps.symbol_layer(locations, fill_color= colors, stroke_co
fig
```



In [ ]: