**Problem Statement:**

* Station is sharing live data in terms of geospatial
* Number of bicycles available in station
* Data analytics for which station to close.

**Background**

* IoT is one of the hot topics and research in same area is going on.
* Use case for this course is Smart Dock.
* In countries like England, Thailand and India, bicycles rental is good business.

**Data**

* The temporary data is generated as live data is not available.
* The data is Location data and live data as well as past data.

**Solve**

* To solve the problem we would use various algorithm.

**User Perspective:** Is it clear how to use this system?

* App
  + Registration
  + Book a Bicycle
  + Notifications

**Designer Perspective:** Is the behavior completely specified in order to allow a design to be performed?

* Dock design
  + Bicycle Pick Up Stations
  + After login users will be redirected on a” Bicycle pick up stations” list screen.
  + Here user can view all the available station list from where they can get a bicycle for a ride.

**Constraints:** Are all relevant constraints defined?

* Person has to go to dock
* Requires internet
* All systems are not available
* Data is not live

**Testing of Components** - Is the testing of individual components well described?

Step 1:- Test all the hardware individually

* Working of microcontroller (uploading basic programs like blinking LED for Arduino)
* Working of tag reader ( using tested code available on website)
* Working of tags ( using various mobile phone reader)
* Working of Ethernet shield without reader attached and where data is available on cloud services like IBM or not. Also send data from cloud and check whether data is received or not if received than is it the same that was sent.
* Working of motor for locking mechanism. Give DC to motor in both direction one after other and check the movement of lock.
* Working of motor with motor drive by giving direct DC signals

Step 2:- Software components testing

* Check GPS in app is giving correct location of stations or not
* Check for bugs on Login page by asking people to make data entry and check on database if correct entries are accepted
* Check station are generated with correct dock number and same signal is available on the cloud services as mentioned on above step.
* Check commands sent by app i.e. lock and unlock are exactly what is required.

**Integration Testing** - Is the testing of connected components well described?

Step 3:- Integrating all hardware together

* Connect microcontroller with Ethernet shield on top of that reader is attached.
* Connect motor drive with above circuit.
* Test all components connected by sending commands from cloud and reading testing tag

Step 4:- Integrate all app pages together

* Check Login is already than directly show stations available.
* Check if booking is working or not by sending commands on cloud service
* Check profile created is available on database. Also check history can be retrieved or not.

Step 5:- Once all the above components are working fine only part left is connecting cloud service which is easy as we have been testing it all the time manually.

**Methodology**

* Foursquare API
* SciKit-Learn
* matplotlib and Folium

**Results**

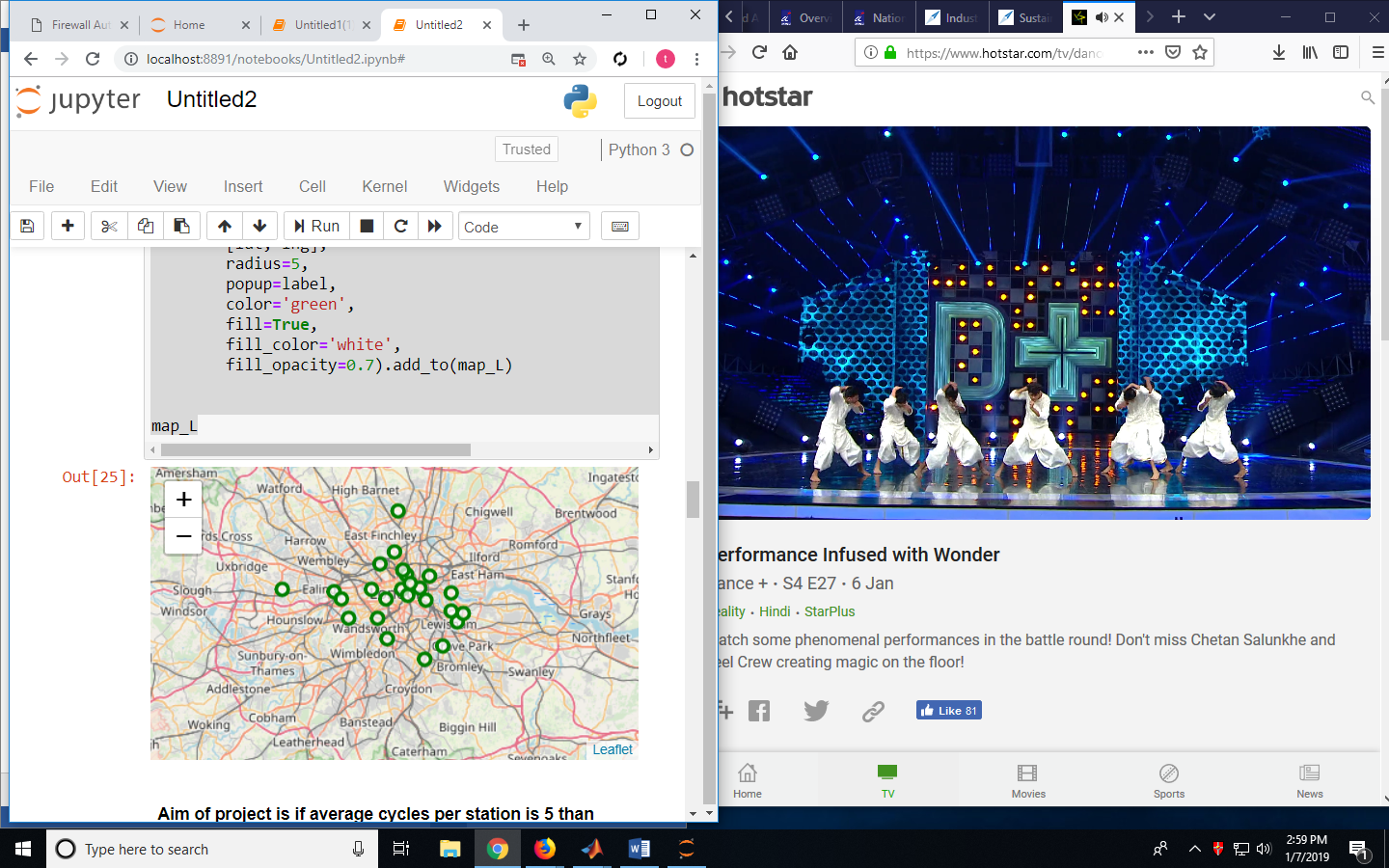


Figure 1 Co-ordinates from Map

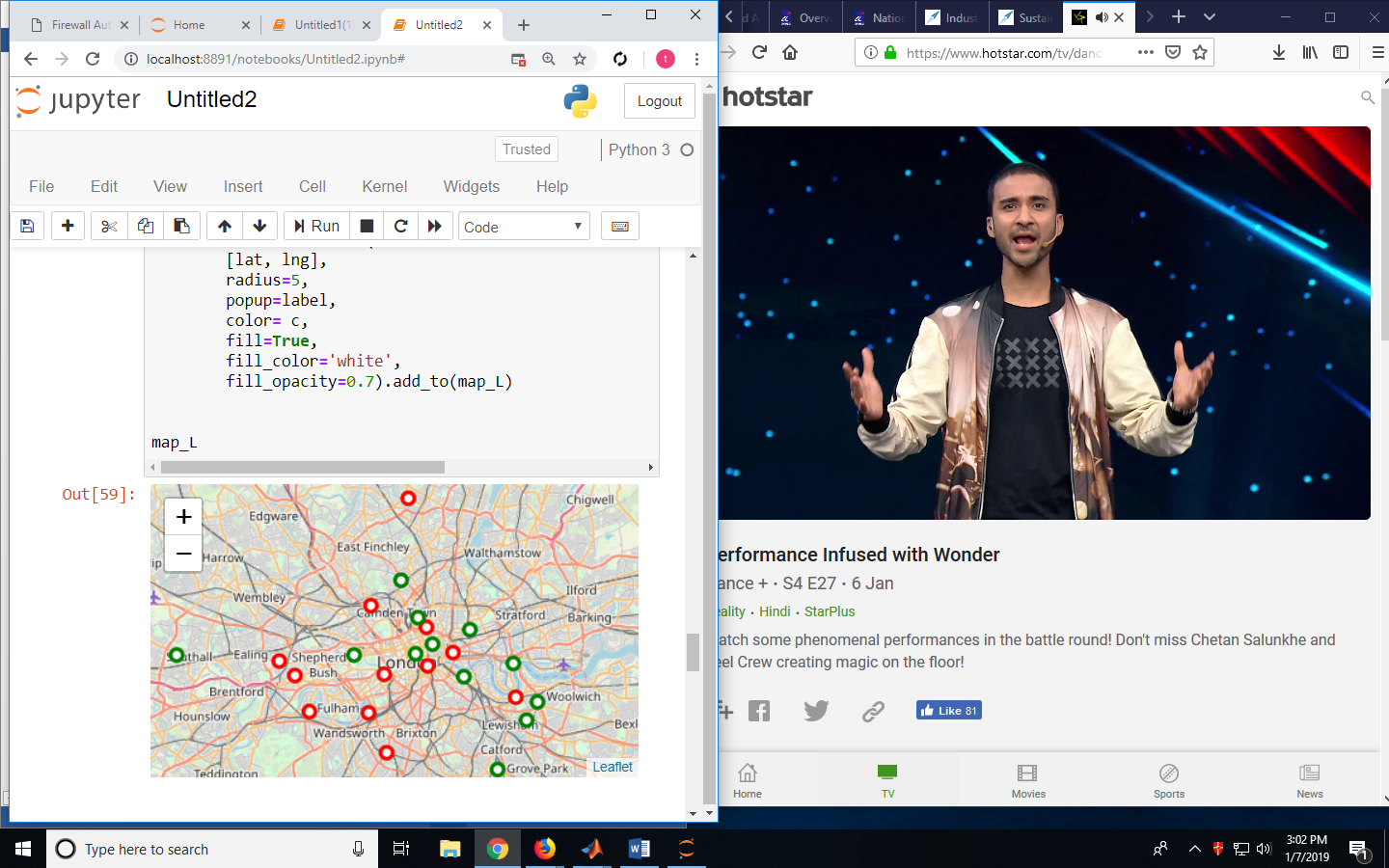


Figure 2 Which Station to be shut?

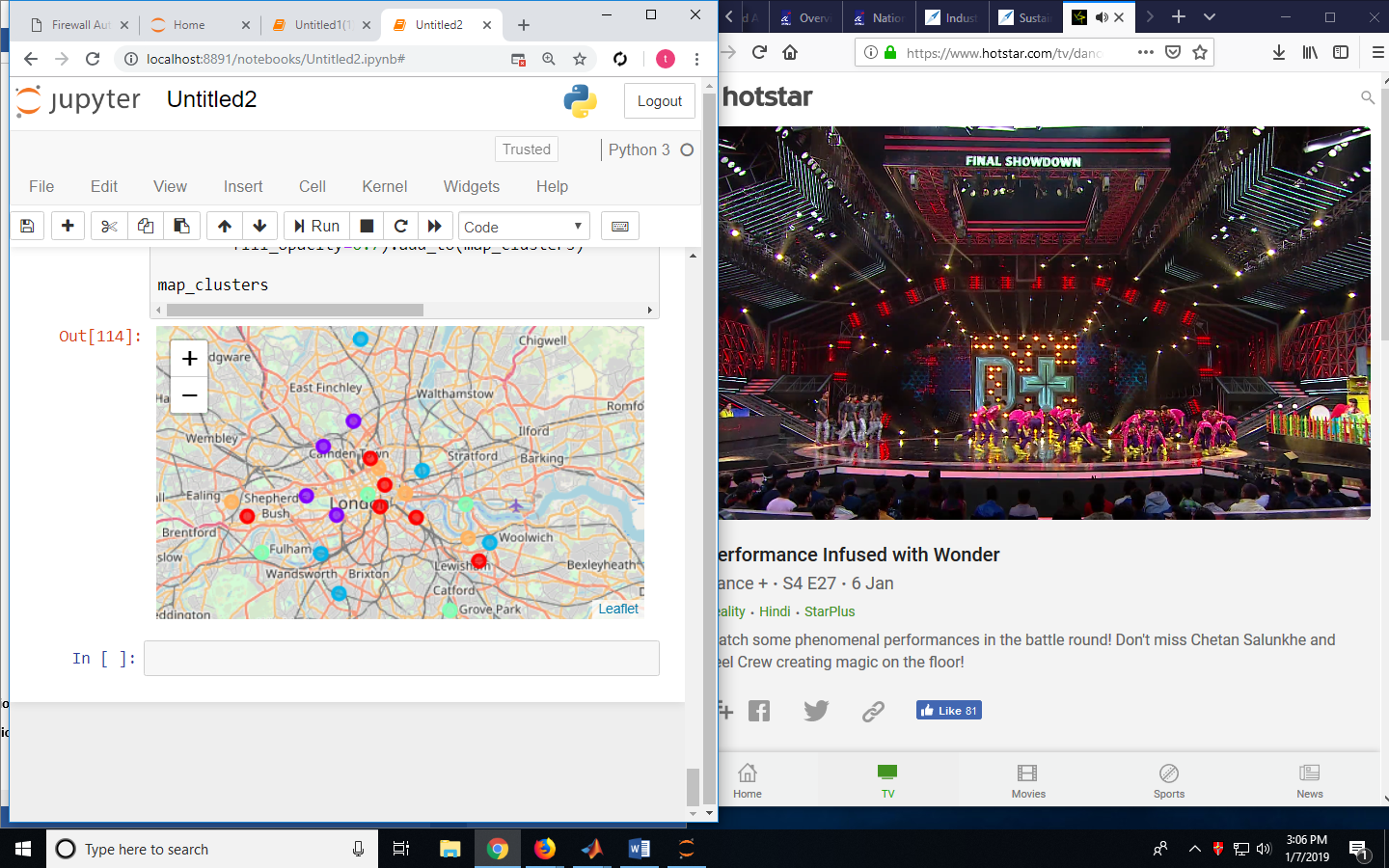


Figure 3 Clusters using Scikit Learn

**Discussion**

* Red is shutting of station and Green are station to be kept open
* The number of cycles that can be kept at station marked using different color for optimizing business

**Conclusion**

* Greenwich, Acton, Aldgate, Angel, Arnos Grove, Balham, Barnes, Battersea, Bedford Park, Belgravia, Belsize Park and Blackfriars are station which can be shut if required.
* Red clusters can be still kept open and reduced cycles from 10 to 5 for optimizing.