



Chicago Pothole Repair Tracker

Jose Murillo
Jenny Xu



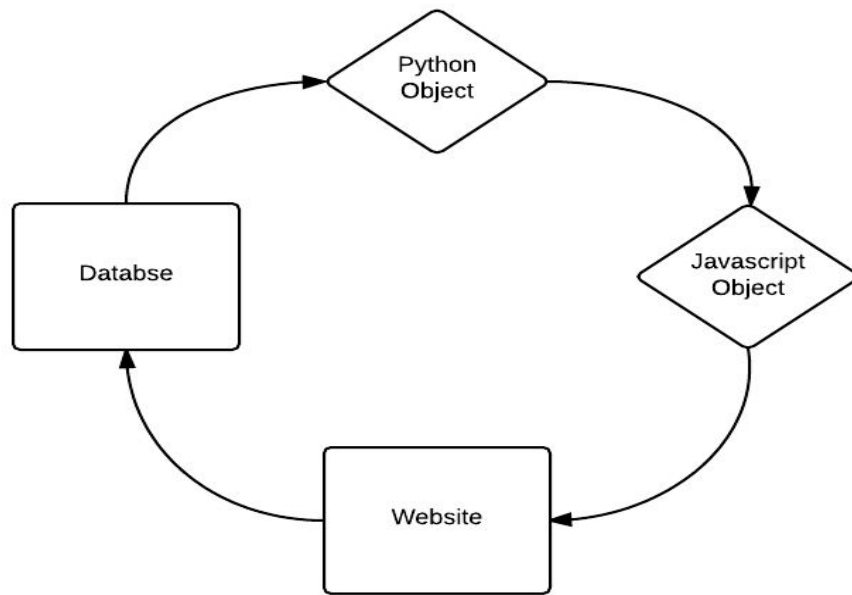
Project Goals

- Create an interactive online map that displays potholes in Chicago
- Features would include:
 - Animation of pothole repairs over a time frame
 - Ability to email the alderman
 - Side-by-side comparison of statistics on repair requests across different wards

Demo

Overview of project

- Data collection/cleaning
 - download csv files from City of Chicago website
 - move data into a SQL database
 - use SQL queries to convert data into Python objects
- Using the data
 - Form creates query which is send to Sqlite database
 - Data from database is used to create Python objects
 - Python objects are manipulated and transformed to Javascript objects
 - Javascript objects modify the content of the website



Interesting things we learned

- Emailing through the website

```
def email(sender, password, message, receivers):  
    import smtplib  
    smtpObj = smtplib.SMTP("smtp.mail.yahoo.com", 587)  
    smtpObj.ehlo()  
    smtpObj.starttls()  
    smtpObj.login(email, password)  
    x = smtpObj.sendmail(sender, receivers, message)  
    smtpObj.quit()
```

Building map markers

```
{% for pothole in ph %}  
var marker = new google.maps.Marker({  
  map:map,  
  position: {lat: {{pothole.lat}}, lng: {{pothole.lng}}},  
  icon: {  
    path: google.maps.SymbolPath.CIRCLE,  
    scale: 5,  
    strokeWeight: 1,  
    {% if pothole.status == "Completed" or pothole.status == "Completed - Dup" %}  
    fillOpacity: 1,  
    fillColor: 'blue'  
    {% endif %}  
  }  
});
```

Displaying pothole information

```
var infobox = new google.maps.InfoWindow();

google.maps.event.addListener(marker, 'click', function() {
  document.getElementById('content').innerHTML=
    '<h2>Pothole near {{pothole.street}}<br>
    Service request #:{{pothole.service_num}}</h2>
    <table><tr><td>Date reported:</td><td>{{pothole.creation_date}}</td></tr>
    <tr><td>Date repaired:</td><td>{{pothole.completion_date}}</td></tr>
    <tr><td>Repair status:</td><td>{{pothole.status}} (...)';
```


Challenges

- Integrating Python and Javascript
- Django
- Divide and Conquer