EmergencMe Hack Night

#sgfwebdevs

EmergencMe is a public safety app that ....

Jim wants to prototype a new feature that would allow the submission of safety alert information that is both submitted from third party sources as well as scraped from various public safety data repositories. So tonight the challenge is to create a web scaper that will crawl a fictitious web site, analyze the data, and then send the interpreted data off to the EmergencMe API, which will then store the alert data in their system.

There are no rules around which tools, frameworks, or languages you use, so use what you are most comfortable with. The goal for tonight is prove the concept and not necessarily to have the best and cleanest code. The winner will be the first person(or group) to accomplish all the goals or whoever gets the furthest the fastest.

All work can be done locally on your own machines. After each goal is reached, flag down the activity leader(s) to verify it is working on your system. Once confirmed your progress will be marked on the board.

Goals

1 - Create a loop to iterate over 10 requests to https://alerts.sgf.dev/ and print out the response

> For this we just want to see that you were able to generate 10 requests to the system and pull back the response.

HEREIN the remaining goals apply to each iteration of the loop

2 - Parse and print the the `details` as outlined below
3 - Parse and print the `type` and `reason` as outlined below
4 - Parse and print the `lat` and `lng` as outlined below
5 - Parse and print the `countyId` as outlined below
6 - Submit the request to `https://alerts.sgf.dev/SubmitAlert` as outlined below and get a successful response

The fictitious site that you will be scraping data from is https://alerts.sgf.dev/

When hitting this root URL you will be presented with a random safety alert. There are key pieces of data on this page that need to be used in order to satisfy the requirement needed to send the alert to the EmergencMe platform.

Here are the following fields that are required when sending data to the submission endpoint.

type
reason
description
countyId
lat
lng

All of these parameters are strings and need to be posted to https://alerts.sgf.dev/SubmitAlert and Form-encoded

Example http request
POST: https://alerts.sgf.dev/SubmitAlert

```
Accept: */*
Accept-Encoding: gzip, deflate
Content-Type: application/x-www-form-urlencoded
Accept-Language: en-us

type=2&reason=41&description=eFactory%20Test%205&countyId=2150&lat=37.2117165&lng=-93.292464
```

A successful request will result in a response in the following format

```
{“data”:{“latitude”:37.2117165,”longitude”:-93.292464,”alert_type”:”2”,”alert_sub_type”:””,”alert_id”:133,”circle_radius”:500,”time_activation”:”2019-05-01T12:31:41.000Z”,”time_deactivation”:”2019-05-01T14:31:41.000Z”,”system_name”:”Pia Society”,”admin_name”:”Talha “,”alert_description”:”eFactory Test”,”active”:
```

```
1,”reason_description”:”Informational Alert”,”attachments”:
[],”update_attachments”:[]},”message”:”Alert sent successfully”,”error”:false}
```

The most important part being

`“message”: “Alert sent successfully”`

An error might look something like this

```
{“message”:””alert_type” is required”,”error”:true}
```

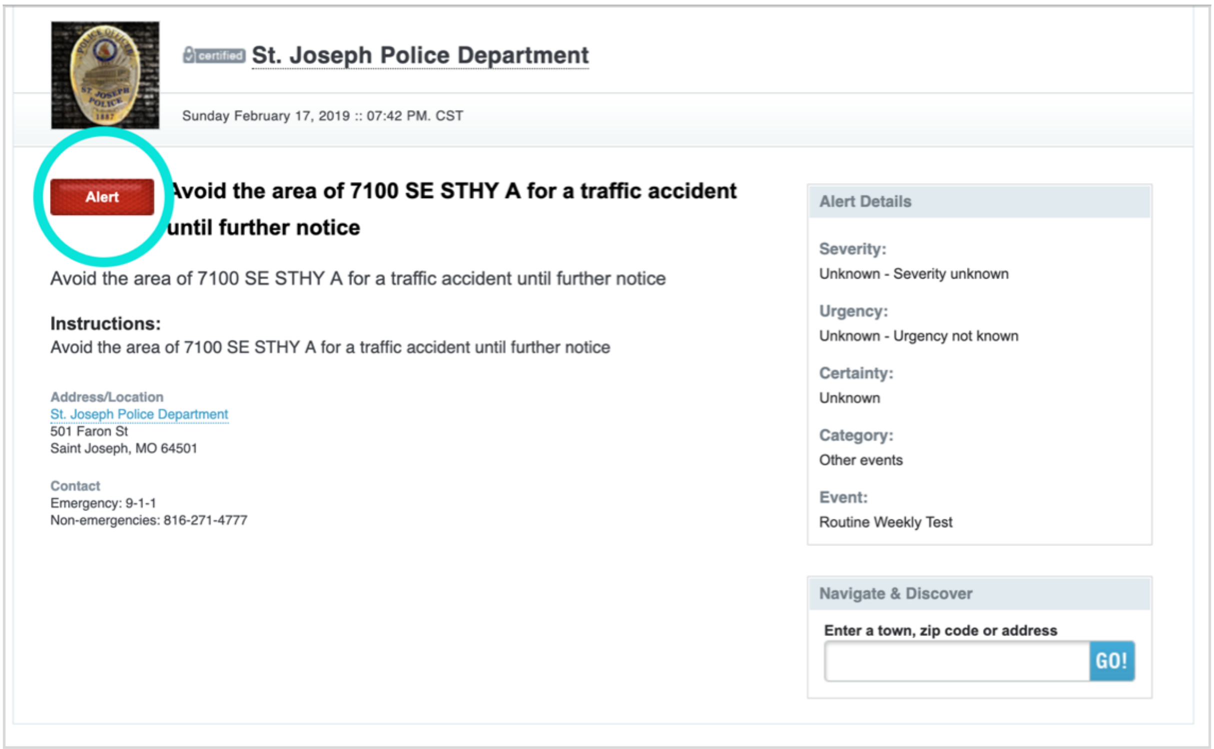
Now let’s break down what each of these parameters so you know how to convert the scarped data into what the EmergencMe system is expecting.

Type and Reason

`type` and `reason` are specific identifiers in the EmergencMe system. For tonight’s exercise we are only to concern ourselves with 2 variants of types and reasons.

If the given scraped example is an “Informational” then `type` will be “2” and `reason` will be “41”
If the given scraped example is an “Alert” then `type` will be “3” and `reason` will be “42”

When the element reads Alert:
The `type` will be set to “3” and the `reason` will be set to “42”



St. Joseph Police Department

Sunday February 17, 2019 :: 07:42 PM. CST

Avoid the area of 7100 SE STHY A for a traffic accident until further notice

Avoid the area of 7100 SE STHY A for a traffic accident until further notice

Instructions:
Avoid the area of 7100 SE STHY A for a traffic accident until further notice

Address/Location
St. Joseph Police Department
501 Faron St
Saint Joseph, MO 64501

Contact
Emergency: 9-1-1
Non-emergencies: 816-271-4777

Alert Details

Severity:
Unknown - Severity unknown

Urgency:
Unknown - Urgency not known

Certainty:
Unknown

Category:
Other events

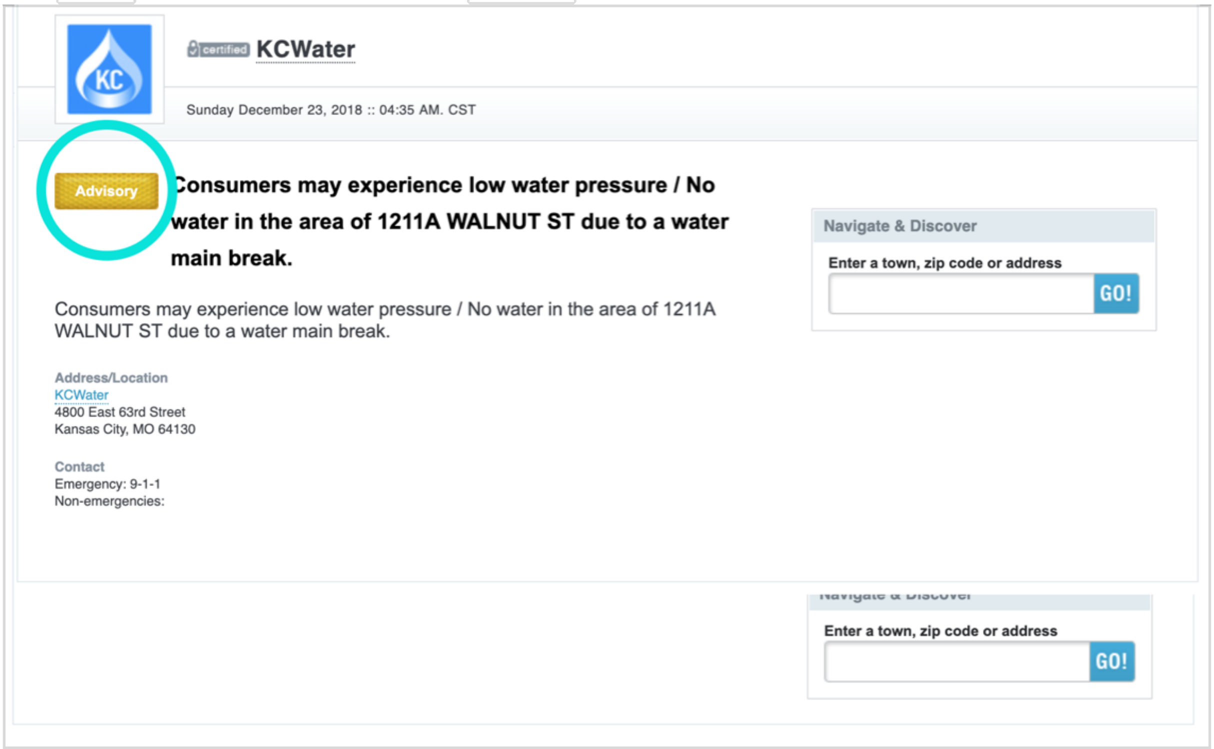
Event:
Routine Weekly Test

Navigate & Discover

Enter a town, zip code or address

When the element reads Advisory:

The `type` will be set to “2” and the `reason` will be set to “41”



KCWater

Sunday December 23, 2018 :: 04:35 AM. CST

Consumers may experience low water pressure / No water in the area of 1211A WALNUT ST due to a water main break.

Consumers may experience low water pressure / No water in the area of 1211A WALNUT ST due to a water main break.

Address/Location
KCWater
4800 East 63rd Street
Kansas City, MO 64130

Contact
Emergency: 9-1-1
Non-emergencies:

Navigate & Discover

Enter a town, zip code or address

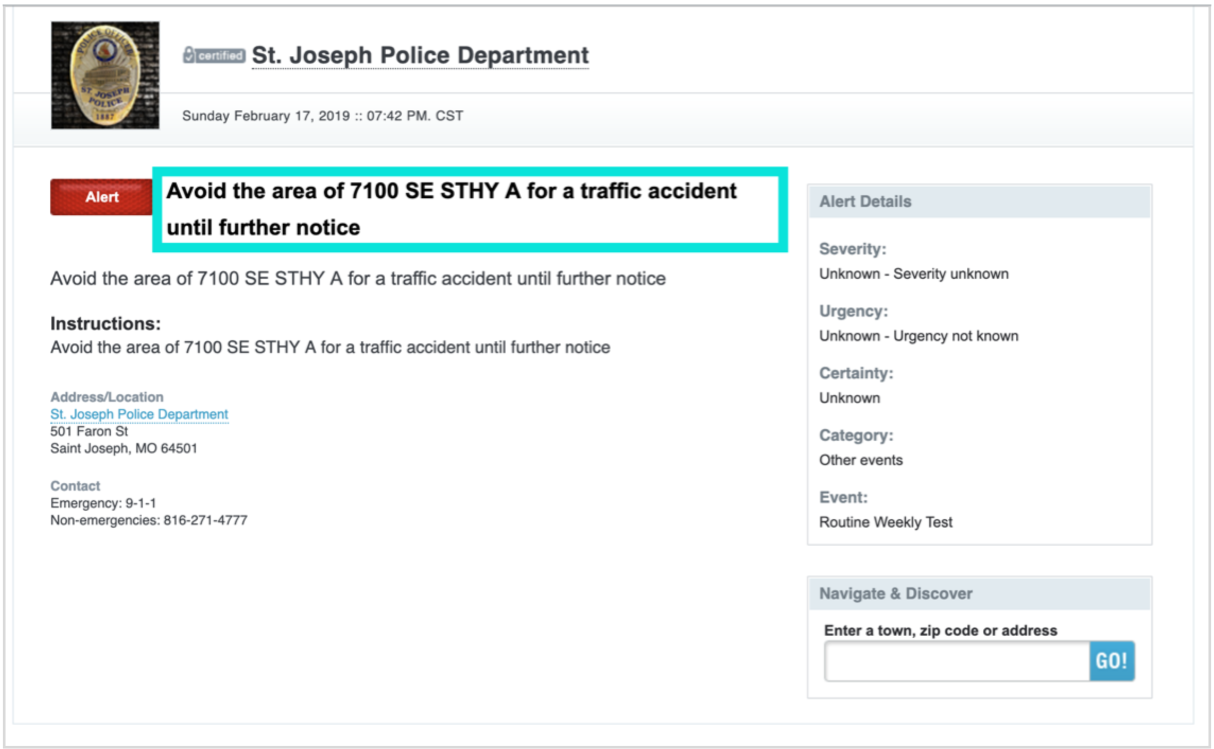
Navigate & Discover

Enter a town, zip code or address

Description

There are few descriptive elements on the page, but the one that will be used for `description` in the EmergencMe system is highlighted below. One addition to the description. For the sake of the competition, please append your name or your group name + " - " to the front of the description. This will make it easier to verify in the system. So in this example, the description would be:

"Bertram Gilfoyle - Avoid the area of 7100 SE STHY A for a traffic accident until further notice"



St. Joseph Police Department

Sunday February 17, 2019 :: 07:42 PM. CST

Avoid the area of 7100 SE STHY A for a traffic accident until further notice

Avoid the area of 7100 SE STHY A for a traffic accident until further notice

Instructions:
Avoid the area of 7100 SE STHY A for a traffic accident until further notice

Address/Location
St. Joseph Police Department
501 Faron St
Saint Joseph, MO 64501

Contact
Emergency: 9-1-1
Non-emergencies: 816-271-4777

Alert Details

Severity:
Unknown - Severity unknown

Urgency:
Unknown - Urgency not known

Certainty:
Unknown

Category:
Other events

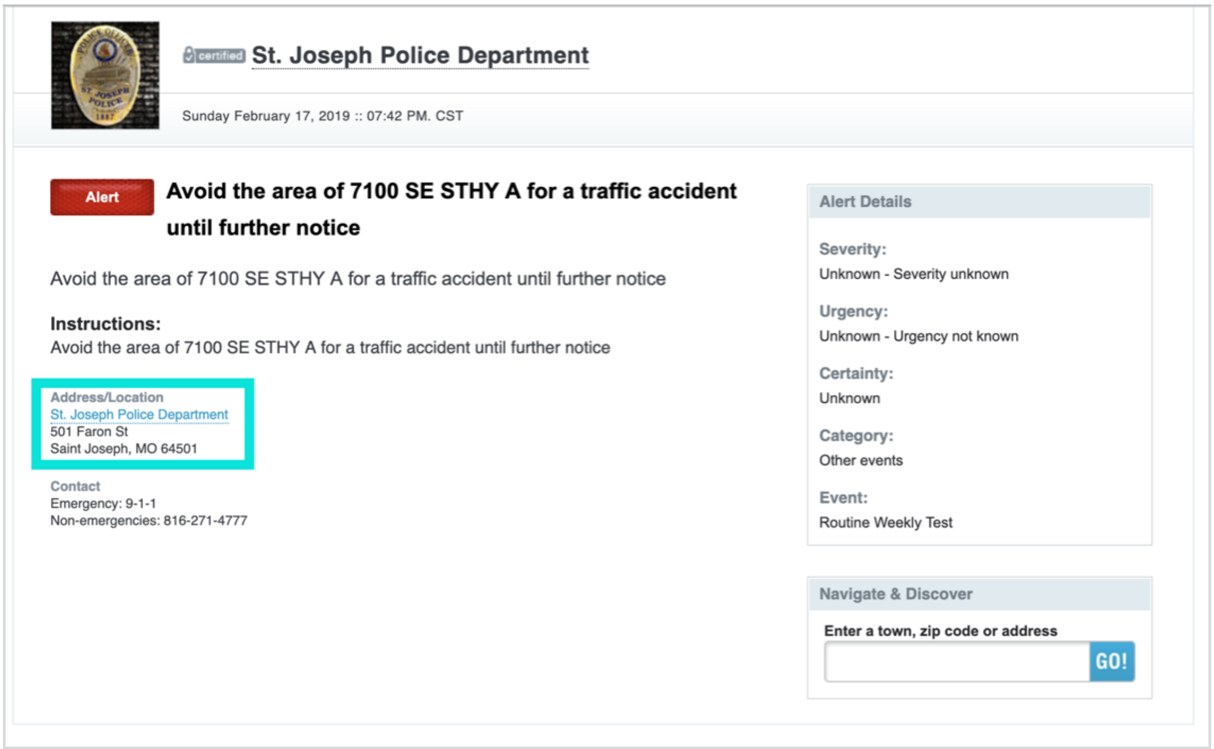
Event:
Routine Weekly Test

Navigate & Discover

Enter a town, zip code or address

Lat/Lng

If you have stared at these examples enough you will have probably already guessed that the geo coordinates will be derived from the Address/Location content outlined here.



certified

St. Joseph Police Department

Sunday February 17, 2019 :: 07:42 PM. CST

Alert

Avoid the area of 7100 SE STHY A for a traffic accident until further notice

Avoid the area of 7100 SE STHY A for a traffic accident until further notice

Instructions:
Avoid the area of 7100 SE STHY A for a traffic accident until further notice

Address/Location
St. Joseph Police Department
501 Faron St
Saint Joseph, MO 64501

Contact
Emergency: 9-1-1
Non-emergencies: 816-271-4777

Alert Details

Severity:
Unknown - Severity unknown

Urgency:
Unknown - Urgency not known

Certainty:
Unknown

Category:
Other events

Event:
Routine Weekly Test

Navigate & Discover

Enter a town, zip code or address

GO!

Use whatever tools/systems you'd like to geocode that address. But those damn Google's and their new API system! Worry not :) Here is an API key you can use. It is only allowed access to the Google Geocode API system and will be disabled after tonight, so don't get any ideas.

Google API Key for Geocode API: `AIzaSyCE7idJgRotl465ooRS0VCaCCP43S3di3s`

CountyId

And lastly the `countyId` param. This is a fun one. You will have needed to obtain the county name from the address provided. Google Geocoding will bring this back. So make sure if you are using another system for geocoding, that you get the county name from it as well.

The county name needs to be matched up with the countyId in the EmergencMe system. To accomplish this, there is an endpoint you will hit to pull a list of all counties in their system. Find the one with the county name that your address matches and use the Id associated with said county name for your request param.

Issue a GET request to https://alerts.sgf.dev/Counties to obtain a list of the MO counties in the EmergencMe system.

Example response(abbreviated):

```
{
    "data": [
    {
        "id": 2128,
        "state_code": "MO",
        "county": "Maries",
        "state": "Missouri"
    },
    {
        "id": 2129,
        "state_code": "MO",
        "county": "Miller",
        "state": "Missouri"
    },
    {
        "id": 2130,
        "state_code": "MO",
        "county": "Moniteau",
        "state": "Missouri"
    }, ]
}
```

So if the address translated to a Miller county location you would set you `countyId` to “2129”