VERSION 1

AUGUST 17, 2016

POKEMON ACTIVITIES

Catch'em All

FINAL PROJECT

METCS 701 RICH INTERNET APPLICATIONS

PRESENTED BY: CHRISTIAN HUR

FACILITATED BY: HARSH PALEJA

PROFESSOR: SURESH KALATHUR

BOSTON UNIVERSITY

POKEMON ACTIVITIES

PROJECT OVERVIEW

Name:	Pokeman Activities
URL:	N/A
Web Technologies:	HTML5, CSS3, JavaScript, DOM, JSON, JSONP, Local Storage, XML, Google maps API, Geolocation, Drag and Drop

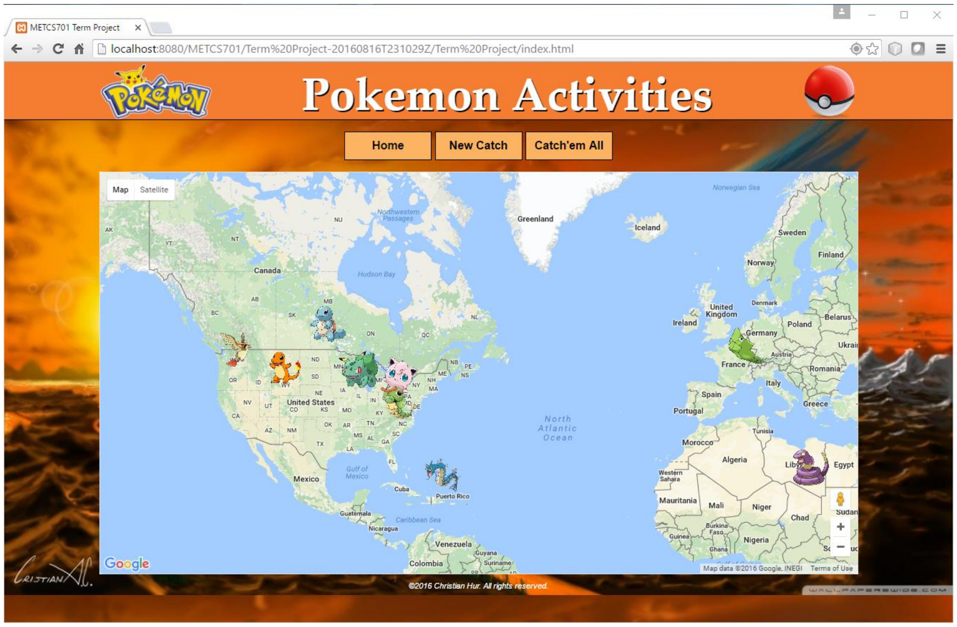
DESCRIPTION

The main objective of this project is purely for entertainment. The project is composed of three different Google maps: Home, New Catch, Catch'em All. The Home map displays custom markers of all the Pokemons captured. The New Catch allows a user to upload their newest catch. The Catch'em All map allows the users to catch as many Pokemon on the map as they can.

The Web technologies used in this project are mainly comprised of HTML5, CSS3, JavaScript, Local Storage, and the Google maps API. This project is built heavily in JavaScript, JSON/AJAX APIs, and Google Maps.

HOME MAP (INDEX.HTML)

The user is presented with the Home Map. There are very little user interactions on this map.



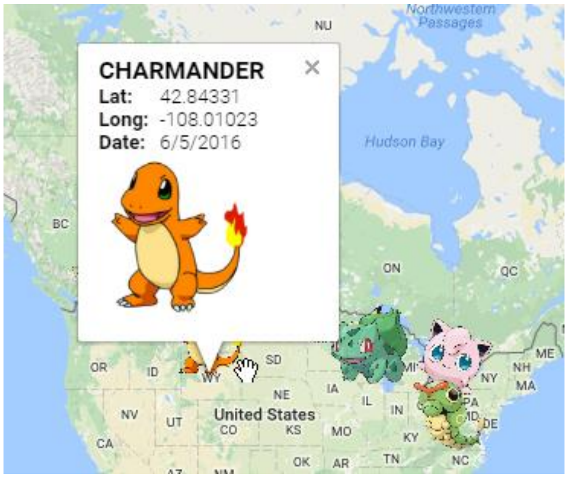
METCS701 Term Project

localhost:8080/METCS701/Term%20Project-20160816T231029Z/Term%20Project/index.html

Pokemon Activities

©2016 Christian Hur. All rights reserved.

The user can click on any of the Pokemon icons on the map to display the InfoWindow about the Pokemon. Information includes the name, coordinates it was caught, date it was caught, and the image of the Pokemon. Click anywhere on the map or click another Pokemon marker to close the current InfoWindow.



CHARMANDER

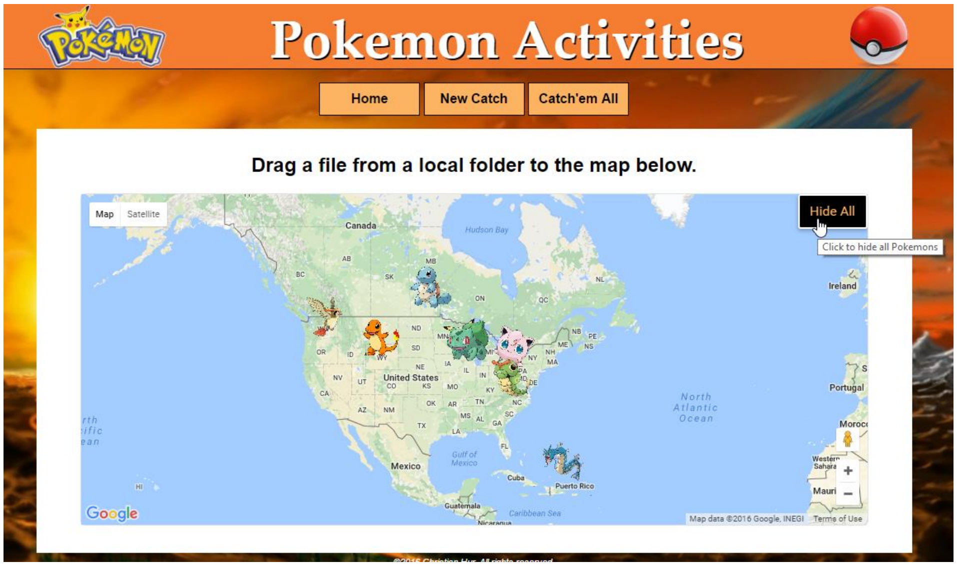
Lat: 42.84331

Long: -108.01023

Date: 6/5/2016

NEW CATCH (NEWCATCH.HTML)

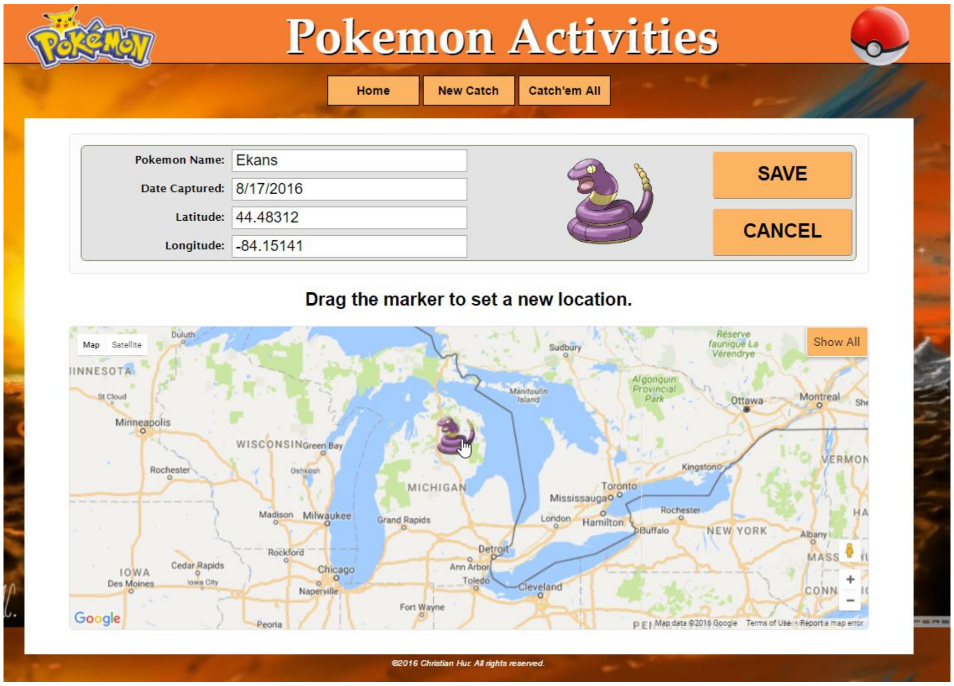
The user is presented with a new map. To the right of this map is an overlay button which toggles to show and hide all captured Pokemons on the map.



Pokemon Activities

Drag a file from a local folder to the map below.

The purpose of this map is to allow users to submit their latest catch. The entire Google map is a "**dropzone**" which allows the user to drag and drop an image file directly from their desktop to the map. After the image is dropped into the dropzone, the system captures the image and displays some default information as seen below.



Pokemon Activities

Pokemon Name:	Ekans
Date Captured:	8/17/2016
Latitude:	44.48312
Longitude:	-84.15141

Drag the marker to set a new location.

©2016 Christian Hur. All rights reserved.

The user can edit any of the four input fields. The Latitude and Longitude fields are spinners and they can be edited by typing directly into them or by clicking on the up/down arrow to the rigth of the field. In addition, the user can drag the marker to a new location on the map. As the marker is dropped to a new location, the Latitude and Longitude fields are also updated with the new coordinates. Any changes to these fields will be registered and reflected on the InfoWindow of the Pokemon.

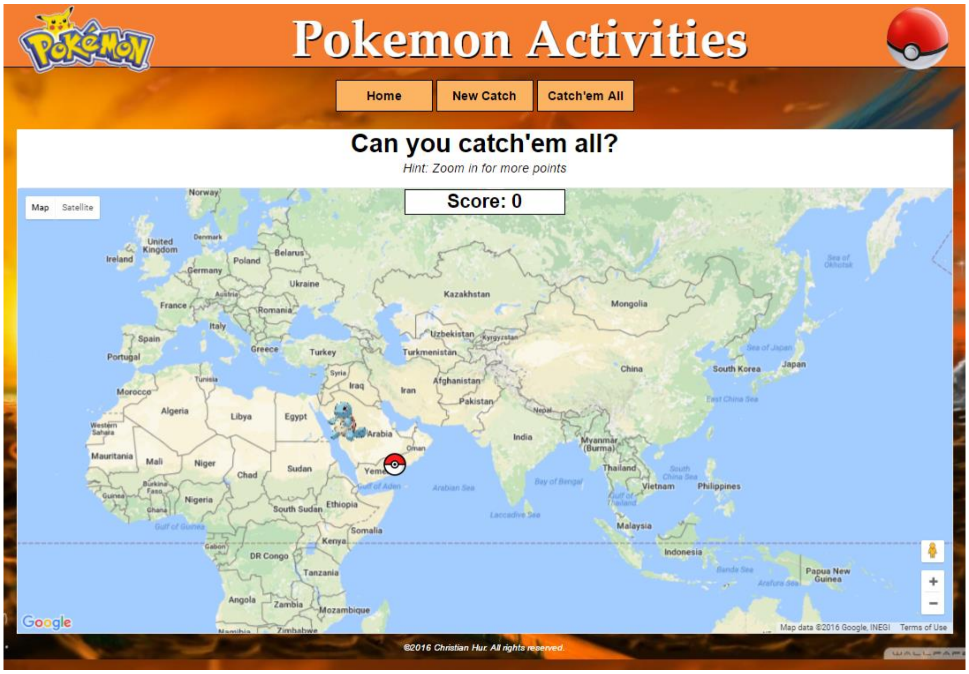
Click **SAVE** to save the current Pokemon to the list. The data is stored in a localStorage (temporary). Click **CANCEL** to cancel the upload and start over.

Note*: Validations are currently not implemented for the input fields at this point.*

CATCH'EM ALL (CATCHEM.HTML)

This is a tiny Pokemon game. A different Pokemon will be dropped onto the map at random coordinates. The purpose is to catch as many Pokemon as you can by click on the Pokemon marker on the map before it disappears. The user can zoom in and zoom out of the map. It is much easier to catch them when the map is fully zoomed out (level 0), however the points earned are minimal. In order to score more points, zoom into the map and catch it. The deeper the zoom, the higher the point. In future implementations, a High Score and a list of *top x scores* should be listed to keep users engaged.

Once a Pokemon has been caught, it is then added to the Local Storage. All the Pokemons caught will be displayed on the Home Map.



Pokemon Activities

Can you catch'em all?

Hint: Zoom in for more points

Score: 0

©2016 Christian Hur. All rights reserved.

TECHNICAL DOCUMENTATIONS

I was torn between the many technologies introduced in this course, and all of them seem really good. However, with the only short time we had, I chose to program my project entirely in pure JavaScript, HTML5, CSS3 using AJAX/JSONP to retrieve an XML file on the server. I was fiddling with jQuery sometimes but I wasn't sure if I could complete what I wanted within the short time. I was also thinking about using AngularJS to load all my pages using Routing but I was out of time.

JavaScript

There are three main JS scripts used for this project, one per map/page:

- pokemonTrack.js (home)
- newCatch.js (new catch)
- catchemAll.js (catch'em all)

However, many of the functionalities are overlapped and they could be consolidated into modules with more time permitted.

JSONP/XML

All the scripts start by loading some default data list from the **pokemons.xml** (via JSONP) or load directly from the Local Storage if there exists one already. This list is then populated as markers on the **home map** and **new catch map**. The list is used for random generation of Pokemon on the **catch'em map**. The XML file includes six fields as listed in the XML schema below:

```
<pokemons>
    <pokemon>
        <date>7/11/2016</date>
        <id>1</id>
        <lat>42.71234</lat>
        <lng>-90.12209</lng>
        <name>Pikachu</name>
        <photo>images/pikachu.png</photo>
    </pokemon>
    <pokemon>
        <date>8/15/2016</date>
        <id>2</id>
        <lat>42.44112</lat>
        <lng>-89.33431</lng>
        <name>Bulbasaur</name>
        <photo>images/bulbasaur.png</photo>
    </pokemon>
         .
         .
         .
</pokemons>
```

More fields could be added in future implementations to include the owner/trainer and other relevant information about the particular Pokemon (i.e. skills, level evolved, etc.).

Google Map API

There are several custom features implemented on the Google Map including the InfoWindow, markers, and overlay controls (buttons and cursor). The markers of Pokemons are stored in an Array object which allows the hiding and showing of these markers easily.

Wish List of Features

There are some features I had wanted to implement but ran out of time. These include filtering the map to show by specific types of Pokemons, date range, time zone, coordinates range, country, city, zip, trainer, etc. Another feature is to include a video clip in the InfoWindow.

REFERENCES

I couldn't have completed everything in this project without many hours of research, reading the text, going over the Modules and examples provided by Professor Kalathur, and other sample codes provided by many open-source contributors on the Web. I am thankful for all these resources. Anything not fully credited in this documentation is purely unintentional.

- MET CS701 - Rich Internet Application Development (Prof. Suresh Kalathur)
- http://www.geoimgr.com/en/tool
- http://www.htmlgoodies.com/html5/javascript/drag-files-into-the-browser-from-the-desktop-HTML5.html#fbid=9cuL6fUzzKO
- https://css-tricks.com/almanac/properties/c/cursor/
- https://developers.google.com/maps/documentation/javascript/reference
- http://www.stackoverflow.com
- http://pokemongo.com

Disclaimer: All Pokemon characters, names, and logos are properties of Nintendo.