

**Interactive Statistic Mapping Application**

**Software Users Manual (SUM)**

CMSC447\_Team3\_CodePods\_SUM

Version 1.2

May 14, 2017

|  |  |
| --- | --- |
| **Course** | CMSC 447 |
| **Team** | Team 3 - Code Pods |
| **Members** | Kevin Miller  David Pan  Benjamin Hazlett  Desiree Mercuree  Darrell Laffoon  Ian Moskunas |
| **Sponsor (Customer)** | Shawn Squire |

**Revision History**

|  |  |  |
| --- | --- | --- |
| **Date** | **Version** | **Description** |
| 03/26/2018 | 1.0 | First Draft Started |
| 05/13/2018 | 1.1 | Starting on the Document |
| 05/14/2018 | 1.2 | Finalized the Document |

**Table of Contents**

[**Introduction**](#_gjdgxs) **5**

[Purpose](#_5wkzo6sbs1qg) 5

[Scope](#_at6dn5z70qz8) 5

[**References and Definitions**](#_gnz8qc44f5cj) **5**

[Table 1 - Document Artifacts](#_6dj1j7eavujw) 5

[Table 2 - Glossary of Terms](#_fjapkn1xv87d) 6

[**System Overview**](#_aj1pdb38zxvo) **7**

[Diagram 1 - System Overview](#_rbv2sdz15po) 7

[Table 3 - Application Components](#_nwsbzxy9zdi2) 8

[Table 4 - Application Actors](#_7g80tbsrr37n) 8

[**Software summary**](#_26in1rg) **9**

[Software application](#_aqqqasjc8bm6) 9

[Software inventory](#_629x0kn4itpx) 9

[Software environment](#_wut2q1f6jt92) 9

[Software organization and overview of operation](#_iu8imc9coj3e) 9

[Assistance and problem reporting](#_qsh70q) 10

[**Access to the software**](#_1pxezwc) **10**

[First-time user of the software](#_2p2csry) 11

[Equipment familiarization](#_s5xdi395kkfl) 11

[Initiating a session](#_6x6z5gpcdfpk) 11

[Stopping and suspending work](#_ef2nezclz97i) 11

[**Processing reference guide**](#_wlj4fcl3jx59) **11**

[Capabilities](#_19c6y18) 11

[Conventions](#_boriwgpb03zg) 12

[Processing procedures](#_euf6zu8o73a4) 12

[Selecting a Statistic](#_k7obvr7sltr8) 12

[Related processing](#_jrjbq8ujp48o) 14

[Data backup](#_206ipza) 14

[**Notes**](#_b1nibhmqnq6w) **15**

# Introduction

## Purpose

The purpose of this Software User Manual (SUM) is to give an overview of the development and use of the Interactive Statistic Mapping Application.

## Scope

This SUM describes an overview of the system as a whole from the perspective of a user. The document will describe how to use the software and the necessary requirement for running the software.

The SUM contains the following information:

* Software Summary including:
  + The application
  + Usage of the application
  + Troubleshooting
* Software Access including:
  + Required equipment
  + Beginning to use the application
  + Stopping and Starting
* Processing Reference Guide:
  + Capabilities
  + Conventions
  + Processing Procedures

# References and Definitions

### *Table 1 - Document Artifacts*

|  |  |  |  |
| --- | --- | --- | --- |
| **Document** | **Description** | **Version** | **Date** |
| CMSC447\_Team3\_CodePods\_SRS | Software Requirements Specification | 12 | 03/26/2018 |
| CMSC447\_Team3\_CodePods\_SDP | Software Development Plan | 1.1 | 03/26/2018 |
| CMSC447\_Team3\_CodePods\_SDD | Software Design Description | 1.1 | 05/12/2018 |
| CMSC447\_Team3\_CodePods\_STD | Software Test Description | 1.1 | 05/14/2018 |
| CMSC447\_Team3\_CodePods\_STR | Software Test Report | 1.3 | 05/15/2018 |
| CMSC447\_Team3\_CodePods\_SUM | Software Users Manual | 1.2 | 05/14/2018 |

### 

### *Table 2 - Glossary of Terms*

|  |  |
| --- | --- |
| **Term** | **Meaning** |
| **The Application** | The targeted software solution - the Interactive Statistic Mapping Application |
| **The System** | The System that encompasses the application. The System and The Application could be used interchangeably in most cases |
| **The Customer** | The Customer that sponsored the project/software Application - Shawn Squire |
| **The Team** | The team of students who will build the Application |
| **SDLC** | Software Development Life Cycle - describes the steps and phases used to design, build and test the application |
| **Agile** | A set of principles that are used to define a iterative and incremental SDLC that is used to build the Application in iterations and allows for Customer feedback to guide the development of the Application |
| **Actor** | A person, external system, or other ‘actor’ who interacts with The Application or System |
| **Component** | A sub-system or part of the Application or System |
| **Map** | The selected, displayed Map, including boundaries. For example, it could be the State of Maryland, or Baltimore, or a neighborhood |
| **Map Data** | Detailed data about the selected Map |
| **Statistics Option** | One of the statistic choices such as crime, school ranking, etc. |
| **Proof of Concept** | A version of the Application that is used to test and prove aspects of the design |
| **Use Case** | Details behavioral requirements |
| **System Model** | Diagram that depicts system components and communication context |
| **Class Model** | Documents the Data entities of the System |
| **Sequence Model** | Documents the sequence of events between actors and components for particular use cases and application events |
| **State Model** | Documents state transitions for the system during particular uses casee and application events. |
| **Test Driven Development (TDD)** | A development methodology where unit tests are created before components are built. |
| **Software Repository** | A distributed data store that holds and tracks versions of the Application’s source code, documentation and other artifacts. |
| **Unit Test** | A test done on low level components as they are built |
| **Integration Test** | A test that tests the ability of system components to work together properly |
| **Regression Test** | A Test that checks that previously implemented features are still working after changes are deployed |

# System Overview

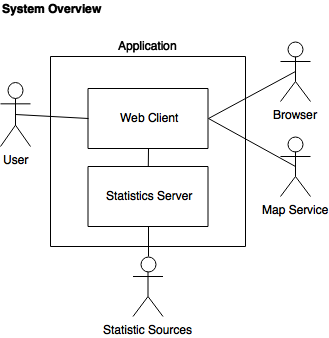
This section provides a high level System Overview for the Application. The Application’s purpose is to help a user answer a universal question:

"Where to I want to live, work, or retire?”

The Application will allow a user to select a Map and visually overlay important Statistics such as crime, income, school ranking, average commute, etc. The overlay will help the user identify areas on the Map that are more or less desirable. An expected, common use case would be a parent, who is looking to move to a safe and desirable place to raise her child, would use the Application to see and compare the crime rate and school rankings in the areas of interest.

*Diagram 1* depicts the key actors and components of the Application.

## *Diagram 1 - System Overview*

[](https://www.draw.io/#G1nqSkvUCma-uFmSX96F2KdXn3Pjgeq0vn)

The Application will consist of two components:

## *Table 3 - Application Components*

|  |  |
| --- | --- |
| **Component** | **Description** |
| **Web Client** | Frontend web application that allow a user to interactively identify a Map, and desired Statistics. It will dynamically update the Map with a visualization of the selected Map and selected Statistics Options |
| **Statistics Sources** | Backend server application that provides aggregated Statistical Data (to the Web Client), such as crime, income, commute, etc., from various trusted Statistic sources. This server encapsulates all data sources and converts them into a common, normalized format |

The Application will have of four primary actors :

## *Table 4 - Application Actors*

|  |  |
| --- | --- |
| **Actor** | **Description** |
| **User** | Will interact with the Web Client to select the Map and Statistics Option she wishes to see visualized |
| **Browser** | Will host the Application and provide input and output. Specifically, the browser will display the Map and a representation of the Statistical Data, etc. |
| **Map Service** | An external map service (Google Maps) that will provide the mapping capabilities and Map Data to the Web Client |
| **Statistic Services** | Publicly accessible Statistical Data sources that will return data for the user selected Map. *See Appendix for list of sources under consideration* |

The User will interact with the Web Client, which will present a Map and Statistics Options. As the User makes changes to the Map and/or chooses a Statistics Option, the Web Client will communicate with the Statistics Server to get updated Statistical Data for the selected Map. The Web Client will then redraw the Map to include Statistical Data based on the User’s selections. The Map Service will provide Map Data. The Statistics Sources will provide Statistics Data.

A detailed look at the user requirements of the Application and the corresponding functional and non-functional system requirements can be found in the Software Requirements Specification (SRS) document (CMSC447\_Team3\_CodePods\_SRS).

# Software summary

## Software application

Interactive Statistic Mapping Application is designed to be an interactive demographic statistic system which uses a map to graphically demonstrate the differences between areas a user wants to move to. The capabilities of this system are viewing the differences in demographic information between areas on a map. The different statistics that can be compared are income, crime, and commute time. These three parameters can be switched while viewing the map and a heatmap will be layered upon the map. This heatmap demonstrates the relative ranking of each area based on those features. The benefits expected from its use is the ability to be better prepared when deciding where to move based on income, crime, and commute time. This will allow users to better plan their future move based on what combination of statistics matter to them. Another benefit will be that the map is simple to read and does not require an extensive survey to find where someone wants to live. The system provides all of the information graphically on top of a map and allows the user to compare them dynamically.

## Software inventory

As this is website there is there is no need to download any file or databases all of the software is stored on a remote server and can be accessed by going to the following website [http://codepods.herokuapp.com/](http://dev-codepods.herokuapp.com/).

## Software environment

This website is designed to be viewed on a desktop or laptop computer. The only requirements for accessing the software are an internet browser of Chrome, Firefox, or Microsoft Edge and a connection to the internet. There is no installation need to use this web application.

## Software organization and overview of operation

* + 1. **Sections of the WebPage**

The website is broken up into three main sections, the dropdown selection menu, the map, and the information panel. The dropdown selection menu is located in the top right of the window and its purpose to to provide a way for the user to select what demographic statistic they want to view. The map component is in the center of the window and provides the user with the graphical representation of the statistic in the form of a heatmap for crime and commute and a kml polygon layer for income. The map also provides the ability to pan and zoom around a map of the Earth. Data is only guaranteed to be relevant to the state of Maryland. Finally the information panel is located at the bottom the screen and this display either information about a specific point clicked upon. The statistic selection, map, and information section are all related. The statistic selection provides information for the map and information section. Then on the map markers are displayed that when clicked on information will be displayed in the information section.

* + 1. **Performance**

Performance of the map is depended upon the zoom level, what statistic is selected, and where the map viewport is located. The larger the zoom goes out the slower the map will update as more information has to be requested from the API and a larger heatmap has to be displayed. The crime statistic generally the slowest statistic and takes the longest to update as there is more data processing that occurs to aggregate the information. The other two statistics should be of similar speeds. Finally if the map viewport is over area with a lot information to be requested from the API the site will slowdown. For example if the selected component selected is crime and the viewport is moved over a city it will be slower in response and processing time than if crime was selected and the viewport was over a small town. When selecting each of the different statistic an http request is performed to get data out of their respective API and from there the information will be displayed on the map.

* + 1. **Limitations**

Limitations of the system are that zooming out too far on the map causes some muddiness of the information displayed as the heatmap begins to blend together. Also zooming out causes an extreme slowdown of the system where wait time will likely exceed what is desired by the user. A purposeful limitation of the software is only allowing one statistic to be displayed at at time as having multiple heatmaps overlaid on top of each other would likely be confusion to differentiate.

* + 1. **Expected Error**

The displayed data is reliable as it is being pulled directly from trusted APIs, however, there is data obfuscation which occurs as the data is aggregated on a heatmap. The information from the API gives information about a specific area and that area is generalized slightly by the heatmap as the colors intersect and blend together.

## Assistance and problem reporting

To report bugs or other errors please contact David Pan through email at Dpan1@umbc.edu.

# Access to the software

This section shall contain step-by-step procedures oriented to the first time/occasional user. Enough detail shall be presented so that the user can reliably access the software before learning the details of its functional capabilities. Safety precautions, marked by WARNING or CAUTION, shall be included where applicable.

## First-time user of the software

### Equipment familiarization

The equipment required for the software is a laptop or desktop computer with a screen that has internet access. Along with the computer there is the requirement for a mouse and keyboard to be able to access and interact with the website. There are no requirements for the screen size as the site will adjust to the window, the size of the screen is preference of the individual user. The cursor on the screen will appear as a pointer when no action can be taken by clicking. When the cursor is over the map and turns into a fist dragging is possible by clicking and holding then moving the cursor with the mouse. When the cursor appears as a pointing hand clicking will take and action and this occurs when the mouse is over the dropdown menu or a marker. They keyboard is only necessary to navigate to the proper webpage. There are no requirements for turning the software on and off as it always is running on the website.

## Initiating a session

To start a session all that is required is to navigate to the [http://codepods.herokuapp.com/](http://dev-codepods.herokuapp.com/) website. Once you are there all that remains to do is start moving the map and changing statistics based upon what you want. The first load of the site may be slow as the server starts up. If the site does not load perform the following steps to see if it will load.

* Attempt to reload the page
* Check internet connection
* Contact David Pan through email to report a bug check section 4.5 for more information

## Stopping and suspending work

Information is not actively saved by this website so to stop working close the website none of your data will be saved and you may resume from the beginning by reopening the website.

# Processing reference guide

# This section shall provide the user with procedures for using the software. If procedures are complicated or extensive, additional Sections 6, 7, ... may be added in the same paragraph structure as this section and with titles meaningful to the sections selected. The organization of the document will depend on the characteristics of the software being documented. For example, one approach is to base the sections on the organizations in which users work, their assigned positions, their work sites, or the tasks they must perform. For other software, it may be more appropriate to have Section 5 be a guide to menus, Section 6 be a guide to the command language used, and Section 7 be a guide to functions. Detailed procedures are intended to be presented in subparagraphs of paragraph 5.3. Depending on the design of the software, the subparagraphs might be organized on a function-by-function, menu-b- y-menu, transaction-by-transaction, or other basis. Safety precautions, marked by WARNING or CAUTION, shall be included where applicable.

## Capabilities

The system’s capabilities are to graphically rank demographic statistics on a map using a heatmap and compare these statistics by swapping between heatmaps. Selecting a statistic is performed by clicking on a specific statistic from a dropdown menu. Selecting a statistic will cause an overlay to appear on the map in the form of a heatmap. This heatmap will update as the user drags around the map allowing for the ability compare multiple locations. Then if a new heatmap is selected the old heatmap will be taken away and the heatmap specific to that statistic will be overlayed. In this way the user is able to compare multiple statistics for a specific area by selection different statistics.

## Conventions

The heatmap uses the same coloring convention between each statistics there are three main colors, green, yellow, and red. Green is for the best areas, yellow is for medium areas, and red is for the worst areas. The crime metric stands for individual crimes this data is then aggregated into the violent and non-violent crimes. Crimes are group together on the heatmap and the color is specified by the following. If in that area there are more violent crimes than non-violent crimes then the heat map will display red, if there are more non-violent crimes then the heat map will display yellow, and if there are no crimes in the the heat map will display green. Commute and income information is organized by zip code and have their levels defined as the higher the income the better and the lower the commute time the better.

## Processing procedures

Selecting a Statistic: This section describes how to select a statistic, this should be done first.

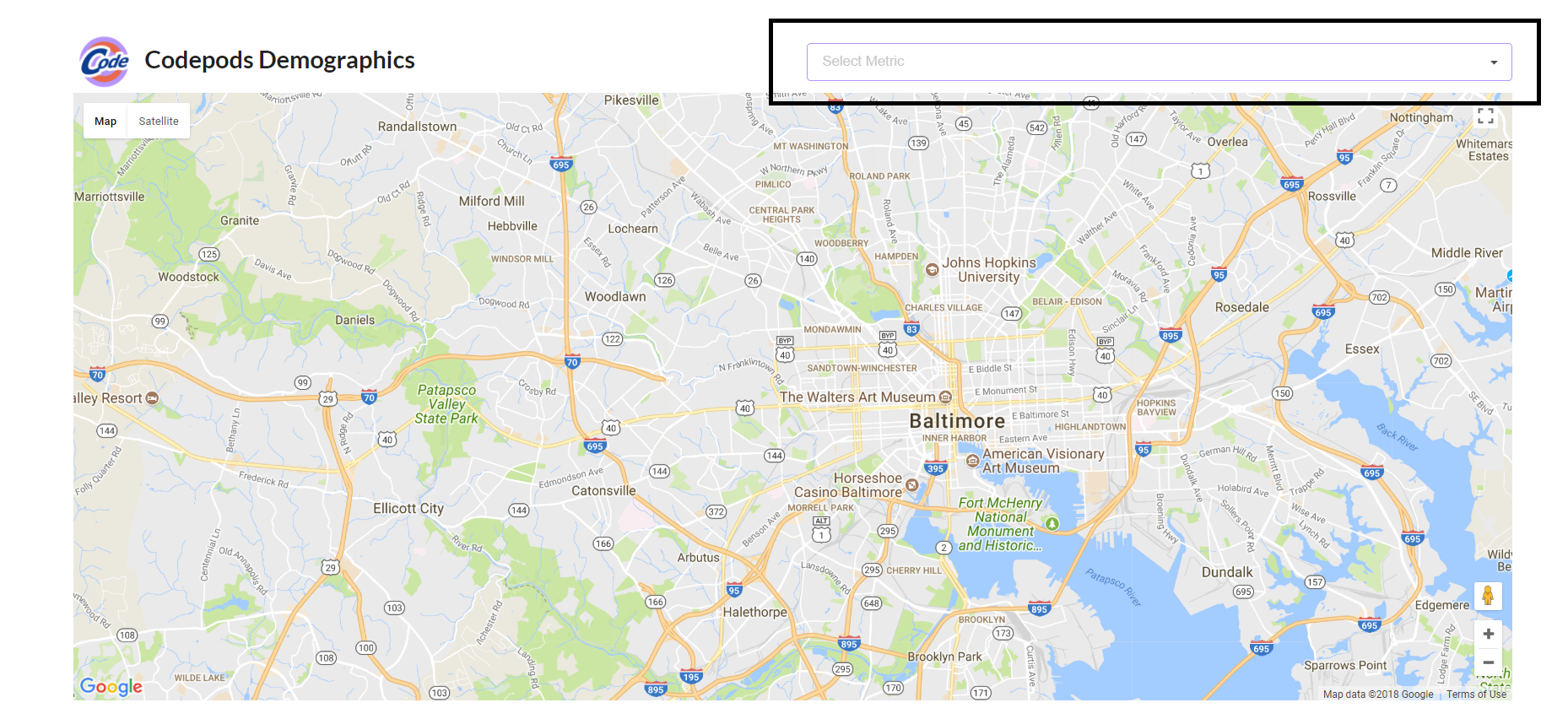
Interacting with a Map: This section describes how to interact with the map component

Using the Statistics Info Box: This section describes the info box component in at the bottom

The order should be these should be done are from top to bottom. Selecting a Statistic then Interacting with a Map then Using the Statistics Info Box. Interacting with a Map and Using the Statistics Info Box can happen at the same time.

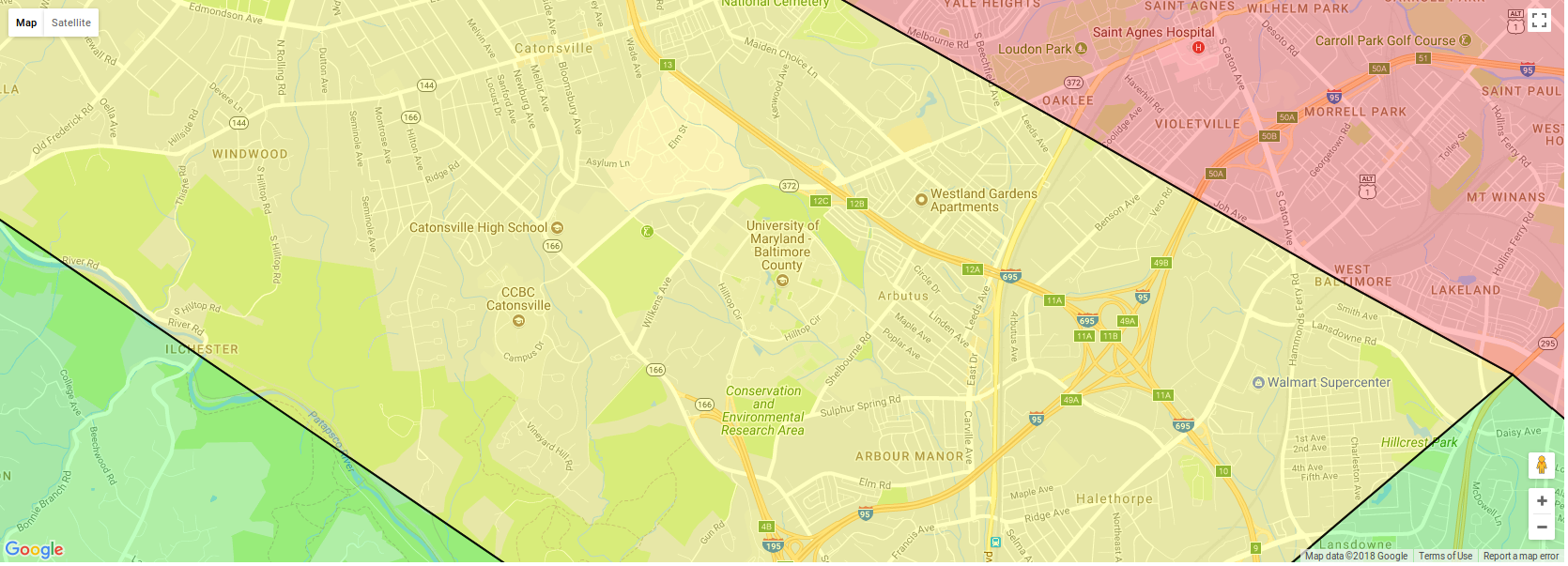
### Selecting a Statistic

In the top right of the webpage on the banner there is a drop down menu, this is indicated in the figure below by the black box



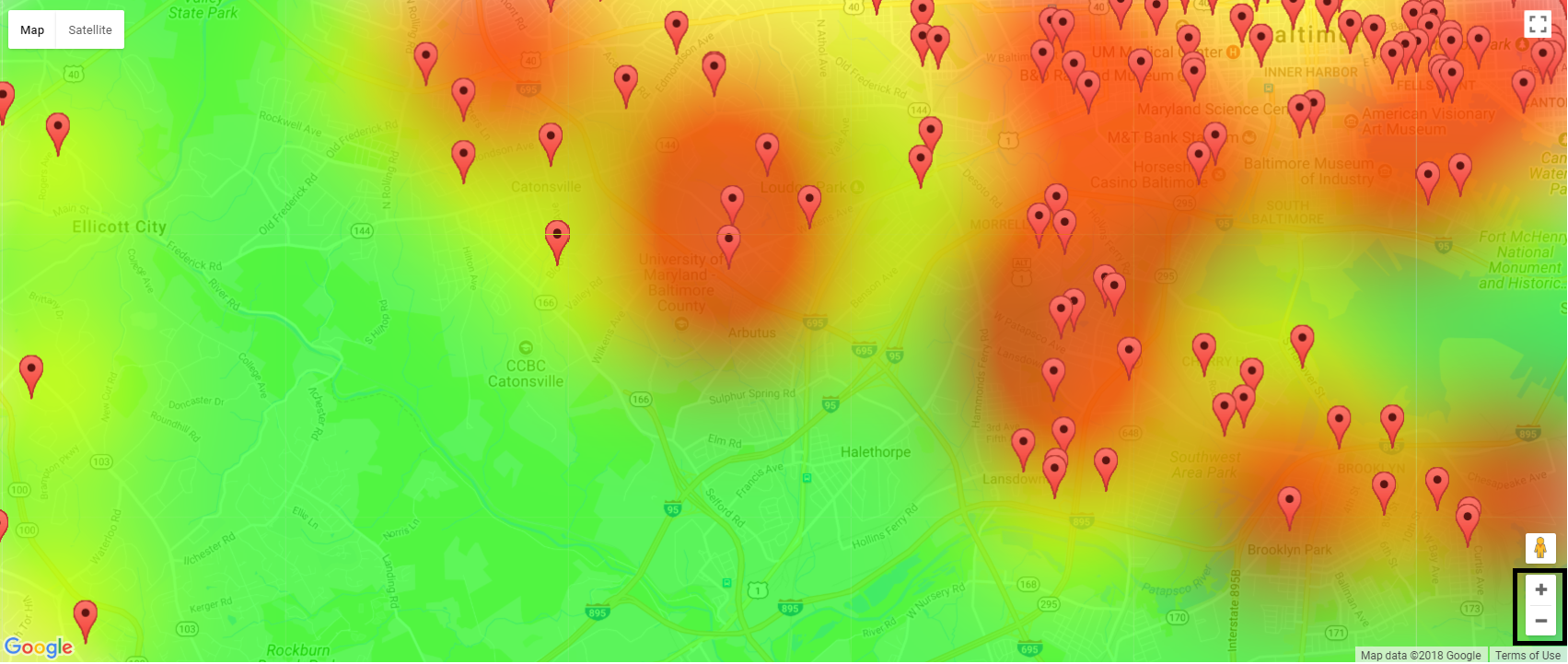
After clicking on the drop down menu a menu will appear and then you may click on a metric option in the menu to display information about that option on the map. If you click on a metric option it will display the heatmap. The map will look like the following. This is the example of a clicking on crime.



The map will look different if income is selected as a polygon kml layer will appear instead. This will look like the following.

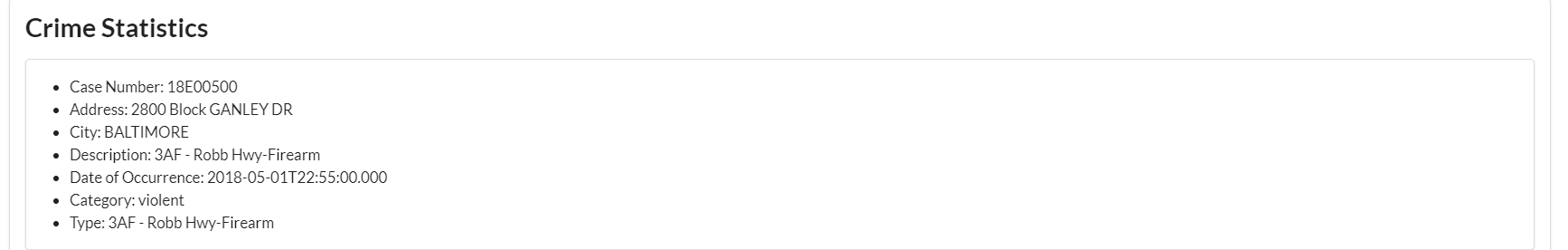
* + 1. **Interacting with the Map**

To interact with the map you can drag the map to move the viewport, the map will reload with a new heatmap after you finish dragging. Another way to interact with the map is to zoom in and out, that is controlled by the plus and minus at the bottom left of the map. This is shown by the following figure outlined in a black box. The other way to control the zoom is by moving the scroll wheel.



* + 1. **Using the Statistics Info Box**

At the bottom of the webpage there is a menu box which displays information about a specific location after it is selected.



This is an example of the Statistic Info Box after a marker is clicked.

## Related processing

In the background of the client side there is a process for making a grid out of the crime API. The data from the crime api is aggregated and put into two dimensional list. Each nod of the list is roughly one square mile and all crimes that occur in that are counted. Then a classification for the grid node is calculated based on the mode of non-violent and violent crimes. There is no need for the user to do anything to help process anything, other than select a metric.

## Data backup

Data is taken directly from an API, there is no set up for data backup.

# Notes

**Terms and Acronyms**

|  |  |
| --- | --- |
| API | Application Programming Interface |
| Dropdown Menu | A menu style where all options are hidden until interacted with and one option in the menu may be selected at a time |
| Google Maps | Interactive mapping software developed by Google |
| Heatmap | Graphical overlay on top of a map which indicates the relative rank of an area through color. |
| KML | Keyhole Markup Language |