CS 411W Lab II

Product Specification

Shayan Ghahramani, Team Silver

04/01/2019

Table of Contents

List of Figures

List of Tables

1 Introduction

Crime HotSpot is a product produced my team Silver and is a web-based application.

1.1 Purpose

The crime mapping application, named Crime HotSpot, will assist users in making well informed decisions in regards to an areas safety based on facts. The use of a heatmap will be the basis of Crime HotSpots crime mapping platform. The heatmap itself will be overlaid on the Google Maps platform while using predetermined color schemes to represent crimes along with the calculated danger in a given area. The algorithms that has been create for Crime HotSpot will calculate an areas safety and store this information in an overlay which will be shown in the website as hotter and cooler areas of the heatmap. In addition to the web-based heatmap there will be a mobile app developing in order to notify users, with their permission, if they have entered an area that poses a significant risk to their well-being. The user will be able to view crime related data on the website and in turn make decisions based off of the provided data. This information can be used in order to plan for trips, visiting an acquaintances, looking for a new place to live, or simply exploring areas that the user has no knowledge of.

1.2 Scope

The goal of Crime HotSpot is to provide users with information regarding crime activiy and patterns in order to make more informed decisions in regards to the areas they are interested in. This can be applicable to businesses looking to open a new branch who desire a safe area, users who wish to travel to new cities or find the safest rout to their destination, or even finding a safe place to live. The user interface will be interactive, allowing users to determine what types of crimes will be displayed while viewing the heatmap.

1.3 Definitions, Acronyms, and Abbreviations

5.1 Application Programming Interface (API) - a set of functions and procedures allowing the creation of applications that access the features or data of an operating system, application, or other service.

5.2. Crime Map - A map that has crime statistical data overlaid on it to provided information on the criminal activity of an area.

5.3. Heatmap - a representation of data in the form of a map or diagram in which data values are represented as colors.

5.4. JavaScript MEAN Stack - MEAN is a free and open-source JavaScript software stack for building dynamic web sites and web applications. The MEAN stack is MongoDB, Express.js, AngularJS (or Angular), and Node.js.

5.5. JavaScript Object Notation (JSON) - a lightweight data-interchange format. It is easy for humans to read and write. It is easy for machines to parse and generate. It is based on a subset of the JavaScript Programming Language

5.6. SafetyScore - A number, proprietary to Crime HotSpot, that represents the relative safety of an area.

1.4   References

Bureau of Justice Statistics. (2018, December). *Criminal Victimization.* Retrieved October 5, 2018

> from Bureau of Justice Statistics: https://www.bjs.gov/content/pub/pdf/cv16_sum.pdf
> Business Insider. (n.d.). *Tourist.* Retrieved October 5, 2018 from Business Insider:
> amp.businessinsider.com/images/5abbaa40a54f322b2d8b4597-750-563.jpg

Crime HotSpot. (2018, December 15). *Presentations.* From Crime HotSpot:
> https://www.cs.odu.edu/~cpi/old/410/silverf18/presentation

CrimeMapping.com. (2018, December 5). *Helping You Build a Safer Community.* From TriTech
> Software Systems: CrimeMapping.com

FBI: UCR. (2017). *Offenses Known to Law Enforcement.* From FBI's Uniform Crime Reporting
(UCR):
> https://ucr.fbi.gov/crime-in-the-u.s/2017/crime-in-the-u.s.-2017/topic-pages/offenses-known-tolaw-enforcement

Lexis Nexis. (2018, December 18). *Lexis Nexis Community Crime Map.* From Lexis Nexis:
> https://communitycrimemap.com/

Microsoft Corporation. (2011, 2). *Heatmap.* Retrieved October 5, 2018 from
> alastaira.files.wordpress.com/2011/02/image24.png

Minnesota Brown. (2018, July 28). *Campaign Signs.* Retrieved October 5, 2018 from
> minnesotabrown.com/wp-content/uploads/2018/07/campaign-signs.png

Neighborhood Scout. (2018, October 8). *VA Crime Rates and Statistic.* From
> NeighborhoodScout: https://www.neighborhoodscout.com/va/norfolk/crime

NIJ. (n.d.). *Mapping Crime: Understanding Hotspots.* Retrieved September 5, 2018 from
> NCJRS: www.ncjrs.gov/pdffiles1/nij/209393.pdf

Old Dominion University. (2017, August 24). *Old Dominion University.* Retrieved September 1,
> 2018 from Old Dominion University: media.wric.com/nxs-wrictv-media-us-east-1/photo/2017/08/24/odu_37569108_ver1.0_1280_720.jpg

Search Business Analytics. (2011, July). *What is a Heatmap (Heatmap).* Retrieved September 5,
2018

> from SearchBusinessAnalytics:
> searchbusinessanalytics.techtarget.com/definition/heatmap Wikipedia. (n.d.). *Crime Mapping.* Retrieved October 5, 2018 from Wikipedia:
> https://en.wikipedia.org/wiki/Crime_mapping

CS 410, TEAM SILVER *Feasibility Presentation* (CS 410, 2018) from CS411:

https://www.cs.odu.edu/~411silver/presentation.html

CS 410, TEAM SILVER *Design Presentation* (CS 410, 2018) from CS411:
https://www.cs.odu.edu/~411silver/presentation.html

CS 411, TEAM SILVER *Lab 1*. (2019, March 28) from CS411:
https://www.cs.odu.edu/~411silver/labs.html

1.5   Overview

   This product description provides information in regards to the Crime HotSpot applications frontend, backend, website application, and database. There is also a description on the Crime HotSpot prototypes key features. The parameters for controlling, and managing, said features will be elaborated upon along with the prototype's performance in regards to outputs, inputs, user interface, and displays.

2   General Description

The prototype will have a user interface which will allow users to manipulate the displayed information on from the web-based applications heatmap.

2.1   Prototype Architecture Description

Crime Hotspot is broken down into the following main categories:

1. Front end: runs the website application while communication to the backend. The users interaction with the user interface will affect this category.
2. Back end: handles retrieval of data, generation of the heatmap, creation of Google Maps objects, and overlaying heatmap all fall under here. The calculation of the SafetyScore is also handled at this level.
3. Database: handles crime data occurs at this level. Crime data is added, stored, organized, and edited. The database will be managed by MongoDB Atlas.
4. Web-based application: allows the user to interact with the heatmap and the crime data on display. The users will be able to choose what crime types will be displayed by the heatmap as well as the crimes weighted value.
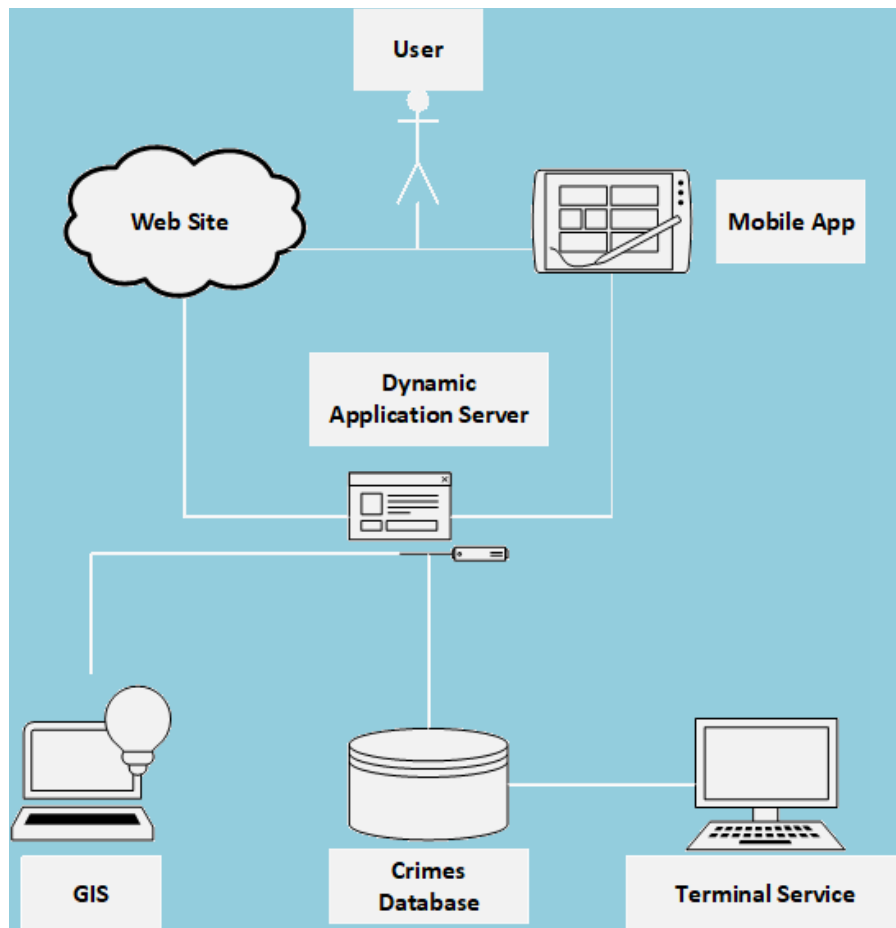

Figure 1. Major Functional Components Prototype Diagram
https://www.cs.odu.edu/~cpi/old/410/silverf18/presentation

6

2.2   Prototype Functional Description

The major functions of the Crime HotSpot prototype are as follows:

1. Hover function: This function allows the user to hover over elements through the use of the mouse, mousepad, or tablet pen to interact with the heatmap. Upon hovering over a given area the SafetyScore for said area will be displayed. The hover function also allows for viewing of the side-navigation bar as well as the main menu.
2. Filter interface: This function allows the users to alter the display of the heatmap by change a crimes weighted value as well as crime categories on display through the users interactions with the slider. The users will be able, through the use of buttons and scales, to alter the heatmaps displayed data.
3. Analytics: This function will allow the user to view more detailed information in regards to a specific area on the heatmap. Once selected a separate window will pop-up and display more detailed information about the selected area such as a list of the types of crimes that have occurred along with the number of crimes related to each crime category.

2.3 External Interfaces

The major external interfaces needed for the Crime HotSpot prototype are a web-browser, database, and a Linux based server. The prototype is currently designed in-order to be compatible with Google Chrome web-browser. In the future the prototype will be updated in-order to be more compatible with various mainstream web-browser such as Mozilla Firefox and Microsoft Edge. The Linux based server is needed for data security measures and housing the database which will in turn hold all crime related data. The manipulation of the crime related data, as well as all calculations requiring said data, will be handled by the server.

2.3.1 Hardware Interfaces

The Crime HotSpot prototype does not require any major hardware interface components.

2.3.2 Software Interfaces

The major software interfaces of the Crime HotSpot prototype are as follows:

- MongoDB Atlas: A database API that will manage crime related data stored inside the database.
- Google Maps API: The combination of Google Maps, Google Heatmaps, Google Geolocator, and Google object classes.

2.3.3 User Interfaces

The major user interface components of the Crime HotSpot prototype are as follows:

- A fully operational computer, computer screen and/or monitor is needed in order to view the prototypes web-basd application.
- A tablet pen, mouse, or mousepad is needed in order to interact with the heatmap and its weighted values. Said values include crime weight, type of crime, and SafetyScore.
- A network connection is required in order to load and update the display of the heatmap.

2.3.4 Communications Protocols and Interfaces

The major communication protocols and interface components of the Crime HotSpot prototype are as follows:

- A live network connection is required alongside a computer with a minimum of 10 mb or available ram.

3 Specific Requirements

- This checker.

3.1 Functional Requirements

- The SafetyScore shall be scaled by a factor based on the distance of the date of the crime's occurrence from the end date (e) of the period of observation.
  - ○ Crimes occurring on (e) shall have a multiple of 1.
  - ○ Crimes occurring prior the start date (s) or after e shall have a multiple of 0.
  - ○ Crimes occurring at some date (d) such that ($s \leq d \leq e$) will have a multiple of $1 - \frac{|e-d|}{|e-s|}$ where $|e-d|$ is the number of days elapsed from d to e.
- The two navigation bars will be displayed, or hidden, based on the user's actions.
  - ○ The main navigation bar will be located at the top center of the heatmap and web-browser.
  - ○ The side navigation bar will be located on the left side of the heatmap and web-browser.
  - ○ Both navigation bars will be visible if hovered over, or clicked on by the user.
- The user will be able to search and navigate the heatmap by entering a zip code or address. The user can also use their pointer to drag the map to their desired location.

3.1.1 Functional Requirement 1

The Crime HotSpot web-based application will allow the users to control the values of crimes and additional filters in addition to viewing the heatmap. The provided functional requirements are as flollows:

1. A side navigation bar containing crime categories in the form of boxes that the user can select, or deselect, in order to alter the information displayed by the heatmap.
   a. Crime weight
   b. Crime date
      i. Day
      ii. Month
      iii. Year
   c. Crimes committed in set timeframe
2. A side navigation bar for controlling the filter for the heatmap layer.
   a. Boxes that can be selected and/or deselected in order to affect the displayed crime data by the heatmap;
      i. Public Crimes
      ii. Property Crimes
      iii. Crimes Against Person
      iv. Severe Crime Against the Person

3.1.2 Functional Requirement 2

The hover function allows users to, upon positioning the mouse over a portion of the heatmap and remaining stationary, the ability to view a pop-up window on the web-based application.
1. View the SafetyScore of the area being hovered over.
2. Controls for the weight and filters.
3. List of each crime committed, based on a crime's relevant crime category.

Command/Condition	Response
Hover the mouse pointer over a spot on the heatmap and do not move.	Determine the coordinates that the mouse pointer is hovering over and give the relevant SafetyScore for that area.
Hover the mouse pointer over a spot on the heatmap and do not move. Once a pop-up window appears, toggle the controls for crime weight and category.	Determine the changes made to the controls and display the relevant data based on those changes.
Hover over side navigation bar on the far left-hand side of the heatmap. Select and/or deselect crime categories at one's own leisure.	The heatmap will show crimes relating only to the crime categories selected.
Change the date for crimes being displayed.	The heatmap will only show crimes from the dates selected.

Table 1. Steering Control Functional Requirements

3.2   Performance Requirements

The hover function must respond accordingly when the user hovers the mouse icon over any portion of the heatmap located on the web-based application.

3.2.1   Performance Requirement 1

The hover function should activate after the pointer has not moved for roughly 1 second or less before displaying the subsequent safety score alongside a pop-up page.

3.2.2   Performance Requirement 2

The weight & filters functions must be able to refresh and update within 3-5 seconds in the case of a user changing a setting or value. In the case of the webpage being refreshed the maximum time is extended to 10 seconds.

3.3   Assumptions and Constraints

The default values for the heatmap and filters will be the same for all users. The displayed crimes will not show crimes older than 1 year from the current time and date.

Condition	Type	Effect on Requirements
Heatmap, SafetyScore, and analytics page may only display data if the database has crime data in that area.	Constraint	The data within the database will affect the data displayed by the application to the user. If the crime is not stored within the database then the heatmap, SafetyScore, and analytics page will not display or include it in any way, shape, or form.
Database has a set size.	Constraint	The database cannot exceed over 512 MB of data at this time. This is due to ODU setting the maximum capacity of the provided server space.
Crime data is accurate.	Assumption	Data provided by government and law enforcement agencies is reliable and has not been tampered with.
Government and law enforcement agencies will provide crime data.	Assumption	If no data is given then the application will cease to provide new crime data.
Google API	Dependency	This application requires the services of Google API to function properly and obtain support for the heatmap and geolocation functions.

Table 2. Effects of Assumptions, Dependencies, and Constraints on Requirements.

3.4 Non-Functional Requirements

Crime HotSpot must be able to function without constant, or frequent, maintenance to its architecture while keeping its data and servers secure.

3.4.1 Security

The crime SafetyScore and other calculations requiring crime data from the database will not involve giving the user access to said crime data. The user will never gain authorized access to the products database or server in order to keep the crime data secure.

3.4.2 Maintainability

Crime HotSpot must be able to function without the need of scheduled maintenance for all parts other than the alteration of crime data located on the products database. The data must be updated and corrected in case of errors and new crimes. No user must be able to directly access the database or the crime data stored within. Other forms of maintenance would be the update to the application's general information, such as presentations, bios, and other such information available to the public through the we-based application.

3.4.3 Reliability

- The prototype must be available 24/7 to all users with a network connection.
- The heatmap must show the most recent crimes added to the crime database.
- The SafetyScore must be updated every time a setting is changed, or if the page is refreshed.
- The maximum number of users the prototype can handle simultaneously must be 100 at least.
- The end product must have adds in order to pay for Google API services in order to function.
- The end product must be able to repel unauthorized access to the database and server.
- The end product must not have any backdoors that unauthorized individuals may use to alter and/or steal from the product.

Appendix

List of required items for the prototype:
- Software
  - MongoDB
  - Google API
- Hardware
  - ODU server
    - Linux server