DEVELOPMENT OF AN APPLICATION TOOL FOR THE SITE LOCATION OF SURVEY CONTROL BEACONS IN OYO TOWN, OYO STATE, NIGERIA.

[1]A.O OYEYODE, [2]D. N. OLAYINKA-DOSUMU AND [3]T. O. IDOWU

[1]Department of Surveying and Geoinformatics, Federal School of Surveying, Oyo, Nigeria.
[2]Department of Surveying and Geoinformatics, University of Lagos, Akoka-Yaba, Nigeria.
[3]Department of Surveying and Geoinformatics, Federal University of Technology, Akure, Nigeria,

ABSTRACT

Location of control points is one of the important factors to consider while planning for a surveying work of any order. Information needed for the site location of survey control points was previously domiciled in a hard copy file system before the advent of database management application. This has led to the limited and difficult location of survey control points on site by Surveyors within the country because information about the control points are not, in many cases, readily available to users. The recent relationship that exists between Geographic Information System (GIS) and other computer technologies has made it possible to host control points information on a platform that permit instantaneous visualization and access to digital information by users. Therefore, it is the objective of this paper to develop an interactive web-based survey control-point-locator (application tool) which allows storage, updating, security, availability and visualization for easy accessibility of survey control points on site. This would be achieved through the integration of GIS and other web technologies using QGIS Software as well as geo-location and mapping application-programming interfaces such as HTML5, Leaflet Javascript Mapping API, MONGODB, CSS etc. The eventual output of the study would be an interactive web application that can be accessed using any smart phone with the control points rendered on the user interface. The results obtained is expected to give insight into what the latest trend in technology has in stock for surveying and the need to arouse the interest of surveyors and other scientists in web application programming in order to develop more surveying-related applications.

Keywords: Site location, Survey beacon, Web application tool, GIS and Nigeria

INTRODUCTION

The present period can be phrased as "house of technology" as technology is a viral content which touch can be found in every aspect of human life, and surveying is also not out of the viral ring.

With the advancement of technology, more and more attention is drawn to the visualization and interoperation of geographic data. Integrating Web Technologies and Geographical Information System

(GIS) offers a wide array of solution to some surveying related problems involving the acquisition, storage, location, accessibility, visualization, analysis and usage of geospatial data.

Surveying is majorly concerned with the determination of position of points on, above or beneath the earth surface, however, new survey work is required to be referenced to a known coordinated point called 'control point' i.e points of relatively precise measurements established by triangulation, trilateration, traversing, intersection, resection, levelling, GNSS observation used to determine the position of features beneath, on or above the earth surface.

Furthermore, owing to technological advancement and the introduction of Geographic Information System (GIS), the field of surveying has gone beyond the measurement of points, and so many surveying operations are now computer-based.

These computer-based systems as defined by (Ghilani et-al.2012) consist of hardware, software, data, and organizational structure for collecting, storing, manipulating, and spatially analyzing "geo-referenced" data and displaying information resulting from those processes.

Web technologies are the general term referring to the many languages and multimedia packages that are used in conjunction with one another, to produce dynamic web sites. Each separate technology is fairly limited on its own, and depend on another (URL: http://www.alphadevx.com/a/7). Examples of web technologies include markup languages, programming language, web servers, databases and others.

According to Microsoft, Markup languages are used to inject text and web processing documents to describe how a document should look when displayed or printed. The internet uses markups to define how web pages should look when displayed in a browser or to define the data contained within a web document, examples of markup languages are HTML, DHTML, CSS, XML to mention but few. Another example of web technologies are the programming languages which enables you to create custom applications and add functionality that is not already part of an application on the internet, examples of web programming languages are Javascript, Vbscript, ASP.net etc.

This paper, however, applied the power of Geographic Information System and web technology to develop an application tool for the site location of survey control beacons in Oyo Town, Oyo State, Nigeria.

MATERIALS AND METHOD

Data

For this research, several orders of ground survey control points data were obtained from the Siwes and Practical's Unit of Federal School of Surveying, Oyo. These data contain coordinates of survey control points established by FSS, Oyo throughout oyo town for student's practical and learning purposes. The data was prepared in Geographic coordinate system (WGS84) and Universal Transverse Mercator projection (WGS84) and the conversion to Geojson format was done with QGIS software for use with the LEAFLET JAVASCRIPT API.

Table 1: An excerpt from control points database

Station ID	Eastings	Northings	Heights
XO1203	607861.719	870359.028	332.600
FSS1/1	607782.129	870106.603	336.111
FSS1/2	607694.051	869774.205	334.005
FSS1/3	607532.081	869115.449	324.627
FSS1/4	607376.756	868553.040	310.023

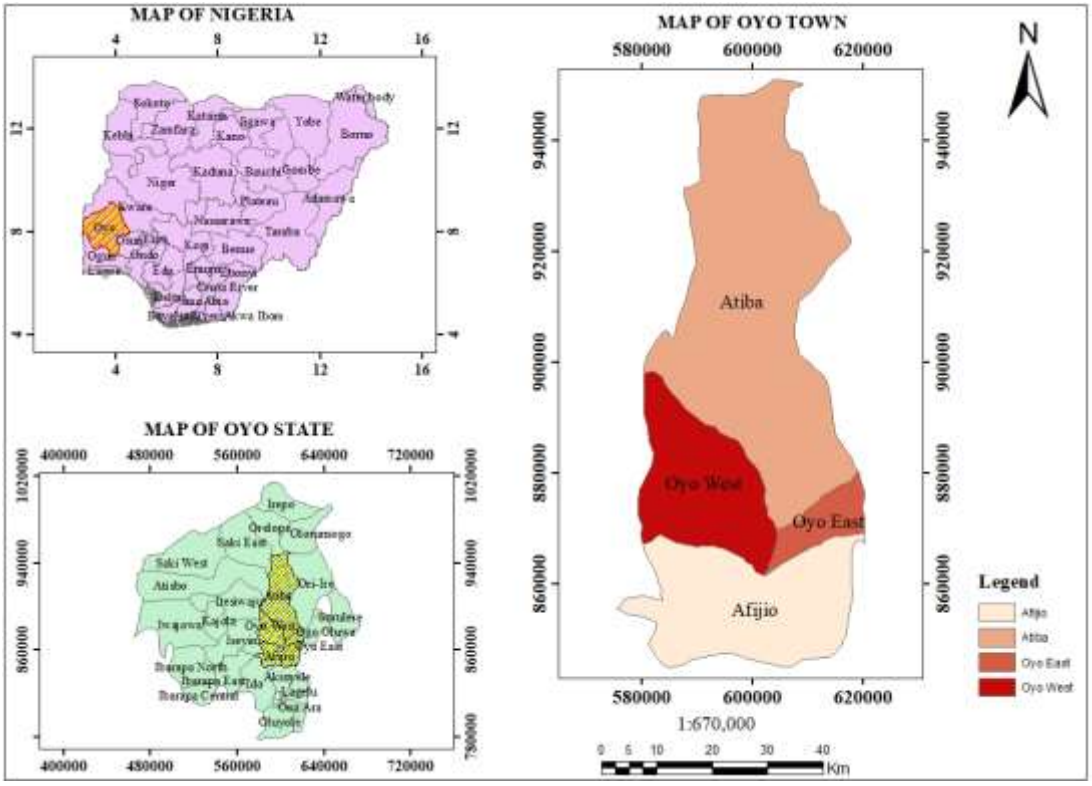

Figure 1: Study Area (Oyo Town)

Application Design and Architecture

To achieve the main goal of the study as earlier stated, system analysis and requirements specification were performed according to user's everyday needs. This was achieved with interviews and assessment, in order to specify the application in terms of its strategic objective and required functionalities, design and usability.

It is a three-tier architecture entirely based on Open-source technology and uses a set of powerful JavaScript framework - Model View Control (MVC) framework and libraries. On the MVC framework, there is a complete separation between the server-side and the client-side logic. The data model is

implemented in the server side, and the views and the controller are implemented in the client side to build more organized and maintainable code

Implementation

Development Environment Setup

To implement this application or the components of the application, a development environment was first setup by installing Atom development environment and Node.js Server with all the required dependencies especially Express and EJS template engine. Node.js is a software platform which helps to build asynchronous and event - driven network applications. It contains http server libraries which allow developers to create their own web server and build highly scalable web applications on top of it. Node.js introduces the power of JavaScript on the Server side allowing the developers to create both Server-Side and Client-Side logic in one single language "JavaScript". Node.js Server uses the Express framework which serves as a solid starting point, provides a robust set of features, middleware and methods. It is shipped with a directory generator which provides a basic directory structure for a web application.

The Main logic happens in package.json which defines the whole server, includes important dependences and connects routes. The route folder describes different endpoints. The endpoints in route folder use helpers from utility folder. The Model folder consists of mongoose models. Mongoose is an ORM (Object Relational Mapping) wrapper for talking to MongoDB. MongoDB is very flexible and does not enforce the developer to have a database scheme defined. MongoDB allows changing the database schemer on the fly, by adding new keys without any over head.

Database tier

One of the most important parts of any application is the database component. It is essential to understand how data will be handled, saved and organized before starting to build an application. Following the user requirements and in accordance with the non-functional requirements of the client, Control-finder database was implemented using MongoDB Server version 5.1 and Mongoose. MongoDB is an open-source, which stores its data in documents. If well designed, it provides swift way of retrieving data. The data in MongoDB is stored in Json format, therefore, once retrieved, it is ready to use with no need of conversion and mapping between formats and structures. The format in which the MongoDB stores the data is called BJSON, which stands for binary JSON. Since JSON is the JavaScript way of storing data, MongoDB works perfectly with the applications built with JavaScript framework. MongoDB, like MySQL uses unique identifier (primary key) for each document so that it is easy to query and find the data. MongoDB supports insert, query, update, and delete operations like any other databases.

One of the features that makes MongoDB stand out against the traditional databases is the inclusion of dynamic schema. Collections in MongoDB have different schema and the documents within the same collection can have as many different schema and shapes as required. This feature enables developers to

start storing data in the database without any consideration of the database structural design. The database schema of Control-finder is presented in Figure 4.

In order to populate and host application database online, GEOJSON (Geographic Javascript Object Notation), an open standard format designed for representing simple geographical features along with their non-spatial attributes, was used to convert control data to Geojson format in QGIS and subsequently migrated to the online Mongo database version; MLAB website (mLab, 2022)

Logic/Business Tier

On the logic tier, Node.js (Node.js, 2022), a server and server-side scripting web development component was used as a server-side scripting language for development of dynamic web pages. The server-side function of Node.js was augmented with Express, mongoose, Geojson, Ejs and MapBox JavaScript libraries dependencies. Node.js is a free and Open-Source JavaScript library and provides the necessary server functionality for complex query execution on the application database and the development of routing, search and update function of the application. Once a request or requests arrive, Node.js server connects to Node.js Server script files, which with the help of other dependencies/ libraries) then decides whether to start a database process, a search process or routing process using MapBox or any other service present on the server. The geospatial data or model of Control-finder application is stored in MongoDB and managed by mongoose and Geojson.

Client-Side Implementation

HTML handled the arrangement of the various contents in designing the user interface of the Control-finder application. CSS 3 (Cascading Style Sheet) handled the aspect of making the user interface attractive and other front-end frameworks were used to ensure that the best result is obtained. HTML5 Geolocation API was used for the geo-location features since it can work in any browser and can thereby work with any smart device.

Looking at the various web mapping API's available for use and also considering their functionalities, Leaflet JavaScript mapping API 1.8 (Leaflet, 2022) was used as the backbone for this application since the API is an open-source mapping API and it is a very light API compared to other web mapping APIs. Leaflet can be effectively incorporated with other JavaScript libraries and it can also be hosted locally on a desktop computer for testing. Leaflet handled the various map visualization, symbolization as well as interactivities with the map interface that were implemented in the design of Control-finder application. The remote database provider used for this research was MLAB. It is an online version of MONGODB and it allows one to connect to any mongo database from any server. Since Control-finder is a server-side application, we made use of Node-JS 8.9.4 as the server side scripting language.

Open Street Map ((https://{s}.tile.openstreetmap.org/{z}/{x}/{y}.png)) and satellite imagery (http://server.arcgisonline.com/ArcGIS/rest/services/World_Imagery/MapServer/tile/{z}/{y}/{

x}) were used as base maps to give the user a realistic view of the environment where the control points are located. HEROKU Cloud (HEROKU, 2022) was used for hosting Control-finder.

Application Administration: Updating of Database

While developing the application, the task of updating the coordinates of the control points within the database was put into consideration because as new control points are being established in the field, they have to be made available for users. This necessitated the development of another application called Control-Finder-Updater. This application can only be accessed by the administrator and it is linked directly to the Control-finder application page. The administrator prepares the data in a Microsoft Excel file and then updates the control points by logging in through the ***admin section*** of the Control-finder.

Creation of Views

Views in web development are the HTML pages. This aspect of the application coding involves the use of a templating engine to create the various views within the application. In order to achieve this, two additional directories, namely, the ***pages*** and ***partials*** were created within the views directory located in the Control-finder project directory. The page directory contains some HTML files with the .ejs extension. These files are the pages which were rendered to the users. The partials directory also contains some additional HTML, CSS and JAVASCRIPT files with the .ejs extension. These files are *Index.ejs, script.ejs, head.ejs, header.ejs*. We wrote the codes for the views.

1. **Index.ejs:** This file contains some HTML tags as well as the necessary .ejs tags that make the rendering of the page possible by combining all other files with the .ejs extension
2. **Script.ejs:** This file contains the Javascript codes responsible for the following: i. Displaying of base map to the user interface ii. Plotting of control points on the base map iii. Addition of some user interface widget for interactivity iv. Styling of the control points v. Generation of the shortest distance to a particular control point

vi. Displaying of additional information about each control points

3. Head.ejs: This file contains the major CSS code for styling the various HTML elements within the application.

4. **Header.ejs:** This file contains the various HTML tags that made up the sidebar within the application.

Application Deployment

This was the final stage involved in carrying out research. It entails sending the already developed Control-finder to the cloud for use. The application was deployed to the HEROKU cloud which is a free hosting site for testing web-based applications. In order to do this, an account was created with HEROKU before the application was then launched to the cloud.

RESULTS AND DISCUSSIONS

Results

The end product of this research is an interactive web-based survey control finder application called "Control-finder". A preview of the application is shown in Figure 5 and 6

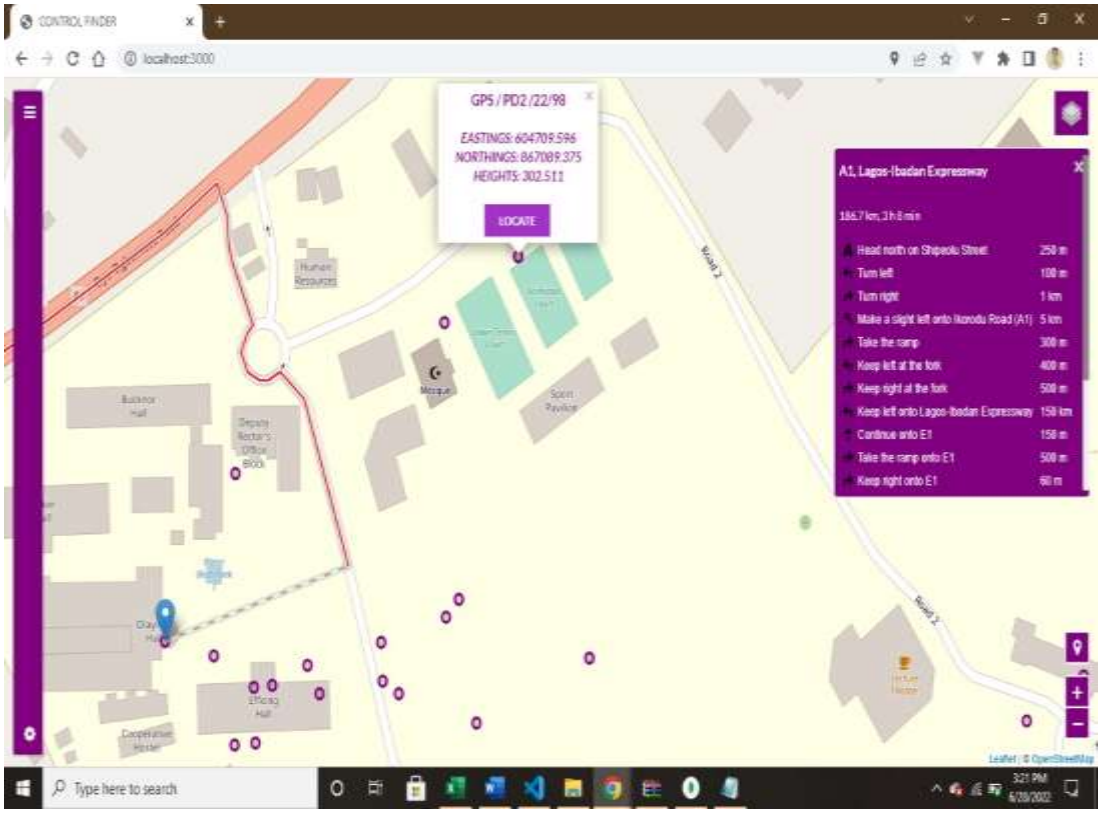

Figure 2: Control-finder application showing information about pillar GPS/PD2/22/98

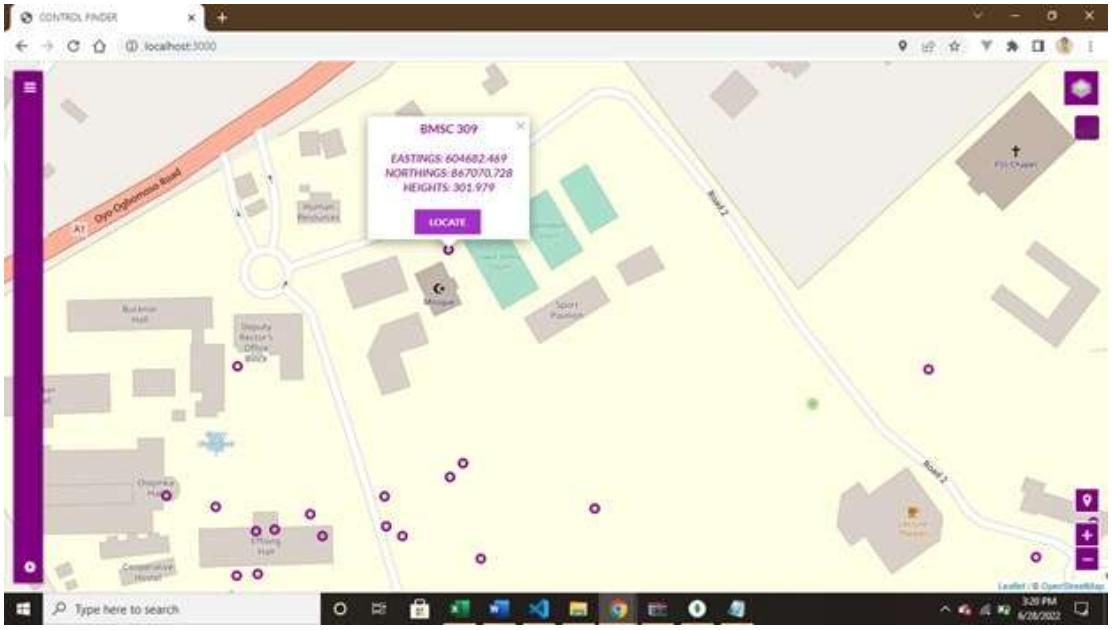

Figure 3: Control-finder application showing information about pillar BMSC 309

Discussions

Control-finder was designed to provide location-based services with respect to the nearest survey control point based on the location of any user within Oyo Town.

Furthermore, the application supports addition and updating of new survey control points into the on-line database since it is a server-side application. The application can be accessed online using the following link https://Control-finder.herokuapp.com. Control-finder can perform the following basic functions:

1. Tracking the current location of the user (this attribute makes it usable globally but the current version has no information on survey control points of other places except the test area which is Oyo Town). This application can be used by all so long as they have access rights to include the control points of their area or country into the system. It is not localized to Nigerian geodetic datum
2. Responding to users click on any of the user interface widgets
3. Generating a route from the user's position to any control point of choice
4. Displaying attribute information about any control point
5. Displaying the control points within a particular buffer radius

Control-finder also has the following features:

- A full screen responsive mobile application compatible with all smart devices
- A backend database for addition and easy updating of control points
- An elegant client side user interface for better user experience
- A routing optimization to help user locate control points nearest to them
- Possibility of updating the control points within the database which can only be done by the administrator. The administrator can login to the Control-finder UPDATER application by inputting his or her username and password in the ***admin section*** of the Control-finder app. After logging into the application, he or she navigates to the file containing the control points that is to be uploaded. He or she then clicks the ***upload*** button and a page will come up, notifying the administrator that the upload was successful.

User Friendly Interface

While developing the application, the end user was seriously put into consideration and as a result of this, the application was enriched with some important user interface widgets that can give satisfactory user experience. Below are some of the user interface widgets:

Layer control: The layer control handles the toggling between different base maps of choice ranging from the street map to the satellite imagery to give the user a realistic view of the environment where the control points are located.

Zoom control*:* The zoom control allows the user to zoom in/out on the desired level of details.

Locate control: The locate control allows the user to re-track its location after launching the application and helps the user to track its location when he or she moves to a different location.

The route control: The route control gives the user a description of what direction to follow along a particular route to locate any control point of choice.

The Popup window: The popup window displays the basic information about a control point whenever the user interacts with them by clicking.

The Sidebar: the sidebar was designed to give some pieces of information about the application. The sidebar control houses the contact section, buffer radius settings and the administrator's login section.

Responsivity Test

After the development, the application was tested for responsiveness by previewing the application on various devices of different screen sizes to see how the user interface is rendered. This was done to ensure that the application can work effectively on any smart device and that it would take over the screen size of the device without distortion to the user interface widgets.

Location Tracking Comparison

Control-finder was also tested to check its user tracking ability. This was done to ensure that the approximate location information provided by the application is accurate and suitable for use. This was carried out by comparing the user coordinates tracked by the application at a point and the actual coordinates of the same point as obtained from FSS database. The result showed one (1) metre difference in distance between the tracked coordinates and the actual coordinates. This result is considered suitable since any survey control point within 1-metre radius can easily be seen by the user.

Conclusion and Recommendation

The development of an interactive web-based survey control finder application (APP) which allows easy location of survey controls, while on the field and based on the location of the user, was successfully carried out and the Control-finder APP is readily available online (
https://Control-finder.herokuapp.com) for use. It is a cross-platform application, meaning that it can run on any smart device irrespective of the Operating System (OS). Interestingly, this Control-finder APP can be improved to meet users' needs and specifications.

This study gives insight into what the latest trend in technology has in store for surveying and how surveyors can maximize the use of some tools to make the surveying profession technologically driven. There is need to arouse the interest of learners in web application programming in order to develop more surveying related applications.

References

Alphadevx. (2022, March 13). *The basics of web technologies*. Retrieved from Alphadevx: https://www.alphadevx.com/a/7-The-Basics-of-Web-Technology

Amit K, & Vineet, K. (2011). Location Based Services using Android Mobile Operating System. *International Journal of Advances in Engineering & Technology*, 14-20.

D, D. P., Proja, J., & Abhishek, S. (2014). Location Based Services in Android. *International Journal of Engineering and Computer Science vol. 3, no. 3*, 5072-5077.

Finder, S. C. (2022, June 10). *Survey Control Finder*. Retrieved from Survey Control Finder: https://maps.sco.wisc.edu/surveyControl-finder/#7/44.735/90.148/NGS,county,USGS,CORS/terrain

HEROKU. (2022, May 18). *HEROKU.* Retrieved from Heroku Cloud Application Platform: http://www.heroku.com

Hu, H., & Lee, D. (2004). Semantic location modeling for location navigation in mobile environment. *Proceedings of the 2004 IEEE international conference on mobile data management.*

IAMAI. (2022, June 20). *Location Based Services on Mobile in India for IAMAI-version*. Retrieved from IAMAI: http://www.IAMAI.in/upload/policy/LBS_draft_indices

Ibrahim, A. O., & Mohsen, J. K. (2014). Designing and Implementation an Online Location Based Services Using Google Maps for Android. *International Journal of Computer Network and Communications Security*, 113-118.

Jegadeesnari, R., & Parameswaran, S. (2014). Location Based Using Autonomous GPS. *International Journal for Research in Applied Science and Engineering Technology, Special Issue 3*, 84-91.

Leaflet. (2022, June 20). *Leaflet Javascript Map API*. Retrieved from Leaflet: https://www.leafletjs.com

Leaflet. (2022, June 20). *Leaflet Locate Control*. Retrieved from Leaflet: https://github.com/domoritz/leaflet-locatecontrol

Leaflet. (2022, June 20). *Leaflet Routing Machine*. Retrieved from Leaflet: https://github.com/perliedman/leaflet-routing-machine

Leaflet. (2022, June 20). *Leaflet Sidebar*. Retrieved from Leaflet: https://github.com/turbo87/leaflet-sidebar

Manar, S., & Aruipam, S. (2012). Implementation of Location Based Services in Android Using GPS and Web Services. *International Journal of Computer Science, vol. 9, no. 2*, 237-242.

mLab. (2022, June 25). *mLab Cloud MongoDB Hosting*. Retrieved from mLab: http://mlab.com/signup

Network, C. o. (2022, June 26). *Survey Services Website.*

Node.js. (2022, June 20). *Nodejs Server side Scripting Language*. Retrieved from Node.js: https://www.nodejs.org

NPM. (2022, June 20). *Node Package Manager*. Retrieved from NPM: https://www.npmjs.com

UI, S. (2022, June 10). *Semantic UI Frontend Framework*. Retrieved from Semantic UI: https://www.semantic-ui.com