Developing a Web-based GIS Application Using Open Source Solutions

Gulben Kaplan
Akemi Murayama

DEPARTMENT OF GEOGRAPHY
CALIFORNIA STATE UNIVERSITY LONG BEACH

Masters of Science in Geographic Information Science (MSGISci)
Department of Geography, California State University, Long Beach

Introduction

Geographic Information Systems (GIS) enable users to not only view spatial data on maps, but also perform complex analysis on spatial data based on location and their associated databases. However, due to its cost and complexity, commercial GIS software can be inaccessible. To address this challenge, open source web-based GIS are becoming a cost effective and practical solution for accessing spatial data and tools. The purpose of this applied research project was to provide an opportunity to learn how to develop an open source web-based GIS solution using free and open source software (FOSS).

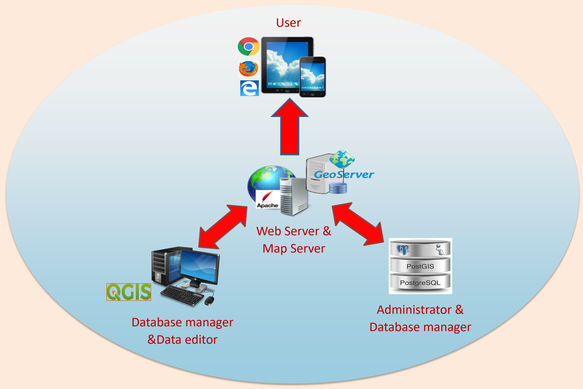


Figure 1. *Application Architecture*

Methodology

The objective of this methodology was to explore the feasibility of creating a web-based application only using open source software.

Four basic components:

Client Side: HTML, CSS, JavaScript Libraries (Leaflet and jQuery)

Server Side: Web Server (Apache), Geospatial Server (GeoServer)

Database: PostgreSQL and PostGIS extension

Desktop GIS solution: QGIS

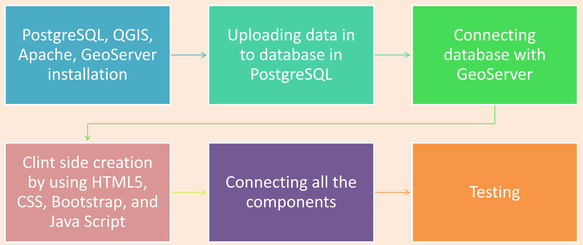
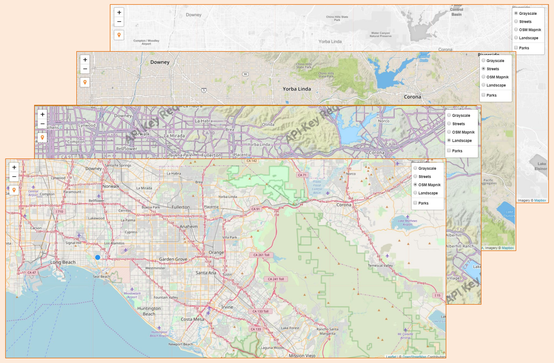


Figure 2. *Method Flowchart*

Data and Data Sources

Base maps used were downloaded from OpenStreetMap (www.openstreetmap.org). The Layers for parks and post offices in Long Beach, CA are included by default.



Database	Source
Greyscale	OpenStreetMap
Streets	OpenStreetMap
OSM Mapnik	OpenStreetMap
Landscape	OpenStreetMap
Parks	The City of Long Beach
Post Office	The City of Long Beach

Table 1. *List of data and data sources used in the project*

Results

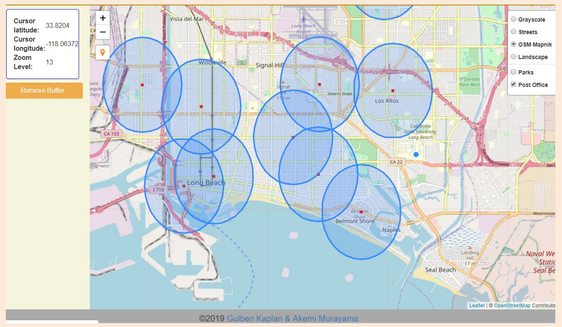


Figure 3. *User interface with hosted web map displaying the current user location and a layer from GeoServer with buffer*

The application was designed to perform the following functions:

- Geolocation to detect the user's current location.
- Ability to add base maps from third parties such as OpenStreetMap and GeoServer.
- Ability to identify the latitude, longitude and zoom level of the cursor on the screen.
- Integrate a simple spatial analysis tool to perform buffers.

Discussion

Focusing on using free and open source solutions enables this application to be free of charge. Selecting proper solutions among all the open source options available was a key to this project. The use of GeoServer as the geospatial server (rather than MapServer) was optimal since GeoServer has as more robust interface for managing data as well as a more streamlined implementation.

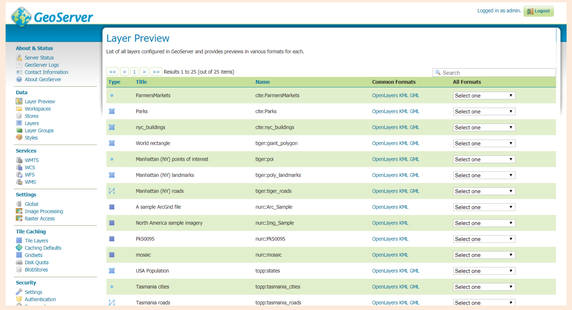


Figure 4. *Geoserver Interface*

Hosting applications was one of the limitations of this project. Free solutions for hosting are very limited, especially for applications with large files or heavy processing requirements. For the purpose of this project, this application was developed only for internal use.

Conclusion

Open source solutions take extra time to efficiently develop. However, they can offer similar functions and features found in commercial geospatial solutions at no or little cost. In a future project, Node.js, a web-server side technology, could be integrated as a potential solution to add smooth and efficient communication between web browsers and servers. Due to time limitations, this component was not incorporated. An open-source JavaScript server environment can run on various platforms such as Windows, Linux and Unix. Node.js can perform a number of functions including generating dynamic page content, performing file operation on a server, collecting form data, and modifying data in a database. With this feature, the web application developed for this applied research project can be further improved.

Submitted in partial fulfillment of the requirements of the Master of Science in Geographic Information Science (MSGISci), August 9, 2019.

For additional information please contact:
Gulben Kaplan (gulben.kaplan@student.csulb.edu) &
Akemi Murayama (akemi.murayama01@student.csulb.edu)