

Florida International University

Fall 2024 Senior Design Project

GeoMapper

Students: Nicholas Garcia
Mentor: *Dr. Antonio Hernandez, Florida International University*
Instructor/Faculty: *Masoud Sadjadi* Florida International University

PROBLEM

Most image files often lack proper tagging and organization, making it challenging for researchers, students, professionals and others to efficiently manage images with valuable metadata such as GPS coordinates, titles, and annotations. This application addresses this issue by providing an easy-to-use solution for organizing and enhancing image files with meaningful information.

CURRENT SYSTEM

GeoMapper was developed entirely from scratch, without utilizing any pre-existing codebases. While there are similar applications available, such as GeoSetter and HoudahGeo, our application was independently created and not based on these existing solutions. Additionally, these solutions do not allow for the conversion of images to KML/KMZ files.

REQUIREMENTS

User Upload of Images: Allow users to upload geotagged images into the application.
Media Management and Selection: Enable users to organize, search, and select specific images for processing.
Media Extraction: Extract metadata such as GPS coordinates, timestamps from uploaded images.
Media Manipulation and Editing: Provide tools to edit or annotate images, including updating metadata, adding titles, or applying annotations.
Image Export and Download: Allow users to export edited or annotated images and download them in formats such as KML/KMZ for integration with mapping tools.

SYSTEM DESIGN

The system design of GeoMapper is divided into two main components: frontend and backend.

Frontend: The frontend is built using a combination of HTML, CSS, and JavaScript, with Axios for handling API requests and Electron as a frontend framework.

Backend: The backend is made entirely in Java, providing handling for media management, metadata extraction, and manipulation processes. Postman was utilized for testing and debugging API endpoints, ensuring smooth communication between the frontend and backend.

OBJECT DESIGN

The object design of GeoMapper revolves around key points that facilitate media processing and metadata management:

Image Object: Represents an uploaded image, including properties like file path, GPS coordinates, title, annotations, and timestamps.

Metadata Object: Encapsulates the metadata associated with each image, including GPS data, resolution, date captured, and user-added annotations.

Export Object: Handles the preparation of images and metadata for export in formats like KML/KMZ.

User Interaction Object: Manages user actions, such as image uploads, edits, and export requests, ensuring a smooth and responsive interface.

IMPLEMENTATION

HTML, CSS, and JavaScript: Form the core of the user interface.

Axios: Facilitates communication with the backend by sending API requests and handling responses.

Electron: Provides a desktop application framework, enabling the deployment of GeoMapper as a cross-platform application.

Java: Used to build all backend functionalities for processing and managing image metadata.

Postman: Assisted in testing and validating API endpoints for frontend connection.

VERIFICATION

Frontend Verification: User Interface: Confirmed proper functionality of all UI elements, including image uploads, metadata edits, and export options.

API Communication: Tested API calls via Axios to ensure accurate and seamless data exchange with the backend.

Export Testing: Verified that exported KML/KMZ files retain all metadata and annotations correctly.

Backend Verification: Unit Testing: Validated metadata extraction, annotation saving, and image processing modules using mock data.

Error Handling: Ensured proper responses to invalid or malformed requests, maintaining backend stability.

API Verification: Endpoint Testing: Used Postman to test all endpoints for functionality, including image upload, metadata retrieval, and export processes.

SUMMARY

GeoMapper is an application designed to address the common challenge of managing and organizing geotagged images. It provides an efficient solution for researchers, students, and professionals who need to handle sets of images with valuable metadata, such as GPS coordinates, titles, and annotations.

With features like image upload, metadata extraction, annotation, and export to formats like KML/KMZ, GeoMapper enhances productivity and simplifies workflows. The tool is particularly useful for geospatial projects, academic research, and fieldwork documentation, ensuring important metadata is preserved and easily accessible.

By streamlining the process of image tagging and export, GeoMapper offers a reliable and user-friendly platform to organize media and leverage it for mapping, analysis, or presentations.

REFERENCES

Electron Documentation and resources for building desktop applications with Electron - Website: www.electronjs.org

Axios Documentation- a JavaScript library for making HTTP requests Website: axios-http.com

Postman Documentation Guide and tutorials for testing APIs and backend communication. Website: www.postman.com

Java Development Resources - Official Java documentation and resources for backend application development. Website: docs.oracle.com/en/java