GeoClover Backend Methodology

# Overview

The GeoClover backend is responsible for handling real-time data transmission, managing geospatial updates, and integrating with the ArcGIS Feature Layer API. This backend was developed using Node.js with Express.js and Socket.IO.

# Setup Steps

## 1. Project Initialization

A new Node.js project was initialized using npm. Essential dependencies were installed, including Express for HTTP server creation, Socket.IO for real-time communication, and Axios for handling HTTP requests.

## 2. Real-Time Socket.IO Configuration

The server establishes WebSocket communication using Socket.IO, enabling bidirectional real-time updates between the client and the backend.

## 3. REST API Integration with ArcGIS

Using Axios, the server sends POST requests to the ArcGIS Feature Layer API whenever a user places a new pin. These updates are formatted in JSON and include spatial reference, geometry, and attributes like username, timestamp, and image URL.

## 4. Environment Variables

Sensitive information like the ArcGIS Client ID, Client Secret, and Token are stored in a `.env` file, which is loaded using dotenv to keep credentials secure and out of version control.

## 5. Running the Server

The server listens on port 3000 and is tested locally using tools like Postman or frontend integration.

## 6. Testing and Validation

Pins are tested by emitting events to the server and validating that they appear in the ArcGIS Feature Layer correctly.

# Conclusion

This backend enables a seamless geospatial collaboration tool by combining Node.js, WebSockets, and ArcGIS APIs to support real-time map-based interactions.