

A Geographic Information System Based on ArcGIS Using Angular Framework

Gaureesh Chauhan

Electronics and Communication Engineering
R.V. College of Engineering
Bengaluru, India
gaureeshchauhan.ec18@rvce.edu.in

Mr. S Praveen

Electronics and Communication Engineering
R.V. College of Engineering
Bengaluru, India
praveens@rvce.edu.in

Abstract—In this study, ArcGIS Server is used as the GIS server software, and ArcGIS Server Java ADF and JSF are examined as the primary technologies of the GIS application after studying the functional aspects of various ArcGIS Server versions. A technique for creating web-based GIS is suggested in order to develop a two-dimensional GIS application that describes the geographic data of all significant airports and airspaces throughout the world. In more detail, ArcGIS API for JavaScript is used to construct projects, Tomcat is used to load user requests, and SketchUp is used to create designs. The application system achieves a very high efficiency, according to experiments. The outcomes demonstrate that the suggested system architecture can fulfil the clients' application requirements.

Index Terms—ArcGIS, Angular, MVC, JavaScript, HTML, VectorTileLayer, Controller, Directive

I. INTRODUCTION

ArcGIS Server is a GIS software sharing object library that is primarily used to construct and develop GIS Server applications using C/S or Web two paradigms. After the second development by the developer, ArcGIS Server 9.0 is an output that may be used. However, ArcGIS Server 9.2 offers a new method that makes it easier for users to publish all services. The Manager tools can be used to establish and use the services, which include Map, Geodata, Geoprocessing, and more [1]. Furthermore, no programming knowledge is required for the entire process of designing web applications. You can now interact with the server more easily and adaptably thanks to these changes.

Model-view-controller (MVC) engineering, information authority, client-side formats, and dependency injection are all used by the open source JavaScript framework Angular to create a genuinely essential structure for creating web programmes. A JavaScript system is Angular. A JavaScript library was used to create it. With the help of a content tag, Angular may be added to a site page as a JavaScript document: HTML is enriched by AngularJS.

- This application is an AngularJS one according to the ng-application mandate.

Identify applicable funding agency here. If none, delete this.

- The estimation of HTML controls (input, select, and text-area) is tied to application information by the ng-show mandate.
- The HTML view is tied to application information by the ng-tie constraint.

A essential client-side web application is JavaScript. These days, the language is used to create a variety of frameworks, including Web applications with trendy User Interfaces [2]. As a result, we are keeping an eye on the introduction of new innovations and tools to shed light on common problems seen in applications like AJAX, AngularJS, Node.JS, and so forth. Today, the majority of systems rely on the Model-View-Controller (MVC) paradigm. The most well-known of these is AngularJS, which is created and maintained by Google.

II. BUILDING COMPONENTS AND DIRECTIVES

The building blocks that make up an application are called components. A TypeScript class with the "`@Component()`" decorator, an HTML template, and styles are all included in a component. The decorator indicated above specifies a CSS selector that describes how the component is used in a template. HTML elements in the template that match this selection turn into instances of the component, and an HTML template tells Angular how to render the component [3]. Example of a minimal Angular component:

Once the HTML record's content has been obtained, the program starts the inquiry and parsing process with the intention of creating the DOM tree. When the tree building is finished, the Angular compiler enters and starts to examine it, looking for rare types of attributes known as directives. The system's bootstrapping technique, which is carried out throughout data collection. There are several orders that define Angular, including *ng - application*, *ng - show*, *ng - tie*, *ng - rehash* among others.

III. SERVER DESCRIPTION FOR ARCGIS

In order to enable business GIS installations and applications, ArcGIS Server is built to distribute complex GIS

```

import { Component } from '@angular/core';
|
@Component({
  selector: 'hello-world',
  template: `
    <h2>Hello World</h2>
    <p>This is my first component!</p>
  `
})
export class HelloWorldComponent {
  // The code in this class drives the component's behavior.
}

```

capabilities from centralised servers. Developers may build targeted GIS Web applications and services using the framework provided by ArcGIS Server, which can be used by clients such as browser-based apps, ArcGIS Desktop, and ArcGIS Engine applications. Software elements known as ArcObjects are used to construct and expand the whole ArcGIS System. ArcGIS Server is an ArcObjects object server.

A. ArcGIS Server Composition

A GIS server and an Application Developer Framework are the two main components of ArcGIS Server (ADF). A Server Object Manager (SOM) and Server Object Container make up the GIS Server (SOC) [4]. ArcObjects running on the GIS server can be used in Web applications and Web services built with the help of the ADF framework. For developers using .NET and Java, the GIS Server and the ADF are accessible. The framework for the Java Platform is discussed in the article.

B. Configuring ArcGIS Server

The user must perform Post Installation Configuration following the installation of ArcGIS Server. The user must enter the server's name and password as well as those for the container account and licence configuration questions. To control who can access and manage the server, the ArcGIS Server security model relies on the security model of the operating system [5]. In order to add people to the ArcGIS Server Administrators group (agsadmin) on the server object management (SOM) computer, the user must first decide who needs administrative access to the server. The user must at the very least join themselves to the agsadmin group. The user should similarly determine who just need user-level access before adding them to the ArcGIS Server user group on SOM machine.

IV. SYSTEM ARCHITECTURE

The application's primary goal is to add layers to the map in order for it to function. In order to do this, data from the weather service is used, and ArcGIS from Esri is used to display the data on the map. Esri is a leader in using GIS to solve issues. As a top mapping and location analytics solution, they

provide ArcGIS. For integrating location-based analytics into users' business activities, ArcGIS offers distinctive capabilities and adaptable licences. By employing contextual tools to view and analyse the data, users can acquire deeper insights. Utilize dashboards, applications, maps, and reports to collaborate and communicate.

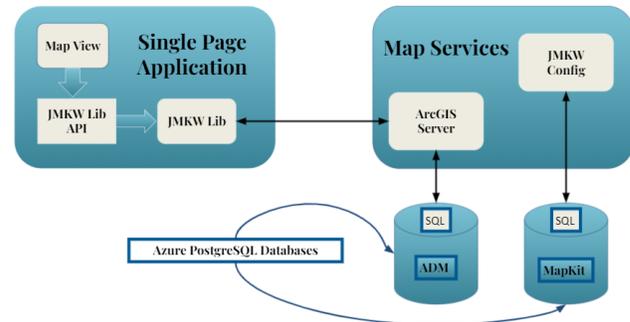


Fig. 1. System Design for the GIS based Application

The application's front-end user interface is created using the Angular framework, and the back-end is created using Java and hosted on Azure cloud services. The application processes the data it receives from the weather service and sends it to the ArcGIS server so that it may be rendered onto the map [6]b4. The ArcGIS data is hosted on a cloud server. Depending on the data set examined, the database will need to be updated and deployed. Some data sets (such as cultural data) will be directly transformed and placed into the cloud data repository, the Figure 1 shows the high level design of the application based on single page development blueprint.

To interchange data between Aviation Map's mobile, web, and organisation data services, some data sets will first be processed into the consolidated data format. Containerized services for the annotation generation software and tiling service will also be hosted in Azure as a temporal group to pre-process the vector tiles [7]. The created tiles and styles will be kept in Azure file storage. At least in one region, a central instance of the preferred Style editor will be accessible.

The ArcGIS based map web application, which integrates the mapkit web library and services as part of its offering to allow visualization of and interaction with geospatial features. The standard interface is a browser run on a desktop computer. The following main use cases towards the web component - independent of the rest of the web application - can be defined:

A. Logging-in to the Application

For all subsequent Mapkit web use cases, logging into the Mapkit map enabled web application is the first step. The GIS based web application user is authenticated after this use case to allow for future application interaction. The FedProxy federates the aviation map services.

B. Manipulate camera position and orientation

To adapt the camera position or orientation to the user's demands, the GIS based web application leverages input

devices such as a computer mouse, keyboard, touch interface, or other methods. In essence, that interaction involves panning the map, rotating or "tilting" the view, as well as zooming in and out. The mapkit web library will display a specific map view depending on the camera position and orientation.

C. Map and REST Services

When appropriate, map and REST services respond with JSON or *MapBox Protobuffers* in addition to the best practices described in the last session. Services must handle errors kindly and report standard error codes. They must also use nouns for endpoint pathways, plural nouns for data collections, nesting resources for hierarchical objects, and nouns for data collections. For faster map service performance, use a CDN like Akamai to cache data.

D. Select layer visibility

Based on the configuration of the web application the GIS based web user is using, aviation map will contain of a number of different data layers. The application user typically has the use case to change the visibility of individual data layers.

V. MAP DEVELOPMENT

The addition of layers onto the user interface is achieved from a custom flow that is based on the architecture of the application. The following points describe the steps taken to make adding layers possible on the GIS based map.

- Using the Esri-provided ArcGIS API for JavaScript, the aforementioned layers were displayed on the map. As Vector Tile Layers, these layers were displayed on the map. It uses cached data tiles and renders them in vector form. Vector tiles can be restyled for many purposes and can adjust to the resolution of the display device. Utilizing cached map tiles with vector map data, *VectorTileLayer* provides stylized maps.
- The Mapbox Style Specification lays forth the requirements for *VectorTileLayer* presentations. The style information for a *VectorTileLayers* tiles is kept separate. As a result, it seems possible to use numerous styles on a single set of vector tiles without having to make separate image caches for each style. This speeds up the creation of new map types and encourages space conservation.
- The custom layers designed first go through the Angular - Single Page Application's ArcGIS API for Javascript implementation before going through the CI/CD pipeline (Continuous Integration/Continuous Deployment). Prior to being merged onto the main branch of the Git version control system, which is used to track code changes, the new code for the layers passes through the CI/CD process.
- Optionally, once the layer result has returned (and assuming there are no errors), a button can be added to toggle the layer visibility, though the visibility can also be toggled.

VI. CONCLUSION

The GIS-based aviation map has fundamentally altered how people develop and use geographic information. The primary user experience is shaped by cloud-based maps, which also act as the creation, sharing, and delivery mechanisms. The airline sector would make extensive use of interactive GIS maps. As a result, they must understand the power of merging layers of all kinds in their maps to gain a deeper understanding of the world. The GIS-based system using this architecture has proven to be effective in practice and is capable of a wide range of complicated operations, including map view, map cartography, map geoprocessing, etc. Therefore, this strategy not only makes development easier and more flexible, but it can also increase the effectiveness of the GIS-based application.

ACKNOWLEDGMENT

The work described in the paper was supported by the department of Electronics and Communication (ECE), R.V. College of Engineering located in Bengaluru, Karnataka, India. The professors and head from the department supported in all the phases of the project, thus would like to express thanks for their guidance.

REFERENCES

- [1] T. Kaushalya and I. Perera, "Framework to Migrate AngularJS Based Legacy Web Application to React Component Architecture," 2021 Moratuwa Engineering Research Conference (MERCCon), 2021, pp. 693-698, doi: 10.1109/MERCCon52712.2021.9525659.
- [2] O. C. Novac, D. E. Madar, C. M. Novac, G. Bujdosó, M. Oproescu and T. Gal, "Comparative study of some applications made in the Angular and Vue.js frameworks," 2021 16th International Conference on Engineering of Modern Electric Systems (EMES), 2021, pp. 1-4, doi: 10.1109/EMES52337.2021.9484150.
- [3] M. R. Pratama and D. Sulistiyo Kusumo, "Implementation of Continuous Integration and Continuous Delivery (CI/CD) on Automatic Performance Testing," 2021 9th International Conference on Information and Communication Technology (ICoICT), 2021, pp. 230-235, doi: 10.1109/ICoICT52021.2021.9527496.
- [4] Harkushko, Lilia. "Angular: Best use cases and reason to opt for this tool." J. Available: <https://yalantis.com/blog/when-to-use-angular/>. [29 8 2019] (2020).
- [5] S. Guan, W. Hu and H. Zhou, "Front-end and Back-end Separation - React Based Framework for Networked Remote Control Laboratory," 2018 37th Chinese Control Conference (CCC), 2018, pp. 6314-6319, doi: 10.23919/ChiCC.2018.8483988.
- [6] M. R. da Silva, T. M. Ramos and M. T. de Holanda, "Geographic information system with public participation on IoT system," 2017 12th Iberian Conference on Information Systems and Technologies (CISTI), 2017, pp. 1-5, doi: 10.23919/CISTI.2017.7975792.
- [7] Camargos, Breno, Maristela Holanda, and Aletía Araújo. "A mobile public participation geographic information system architecture for collecting opinions about public services." In 2015 10th Iberian Conference on Information Systems and Technologies (CISTI), pp. 1-6. IEEE, 2015.