|  |  |  |
| --- | --- | --- |
| **INDEX** | | |
| **Chapter No** | **Description** | **Page No.** |
|  | **CERTIFICATE** | **2** |
|  | **DECLARATION** | **4** |
|  | **ACKNOWLEDGEMENT** | **5** |
|  | **ABSTRACT** | **6** |
|  | **INDEX** | **7** |
|  | **LIST OF FIGURES** | **8** |
|  | **SYNOPSIS** |  |
| **1** | **INTRODUCTION** | **9-10** |
| **2** | **PROBLEM DEFINITION & SCOPE** | **11-12** |
| **3** | **SOFTWARE REQUIREMENT SPECIFICATION** | **13-14** |
| 3.1 | Software requirement | 16 |
| 3.2 | Hardware requirement | 16 |
| 3.3 | Functional requirements | 17 |
| **4** | **PROJECT PLAN** | **15-19** |
| 4.1 | Project schedule | 15 |
| 4.2 | Project cost estimation | 16 |
| 4.3 | SDLC | 17 |
| 4.4 | Feasibility Study | 18 |
| 4.5 | Risk Analysis & planning | 19 |
| **5** | **SOFTWARE DESIGN** | **20-29** |
| 5.1 | Data flow diagram | 23 |
| 5.2 | Flow chart | 24 |
| 5.3 | System Architecture | 25 |
| 5.4 | UML diagrams | 25 |
| **6** | **IMPLEMENTATION DETAILS** | **30** |
| **7** | **SNAPSHOTS/ GUI** | **31** |
| **8** | **CONCLUSION** | **32** |
| **9** | **BIBLIOGRAPHY** | **33** |

**LIST OF FIGURES**

|  |  |  |  |
| --- | --- | --- | --- |
| Sr. No. | Figure No. | Name of Figure | Page No. |
| 1 | 5.1.1 | 0th level DFD | 23 |
| 2 | 5.1.2 | 1st level DFD | 23 |
| 3 | 5.1.3 | 2nd level DFD | 24 |
| 4 | 5.2 | Flow Chart | 24 |
| 5 | 5.3 | Architecture Diagram | 25 |
| 6 | 5.4.1 | E-R Diagram | 25 |
| 7 | 5.4.2 | Use Case Diagram | 26 |
| 8 | 5.4.3 | Sequence Diagram | 26 |
| 9 | 5.4.4 | Class Diagram | 27 |
| 10 | 5.4.5 | Activity Diagram | 27 |
| 11 | 5.4.6 | Component Diagram | 28 |
| 12 | 5.4.7 | Deployment Diagram | 28 |
| 13 | 5.4.8 | State chart Diagram | 29 |

**1. INTRODUCTION**

* 1. **– MOTIVATION**

In today’s digital world, Voice user interfaces (VUIs) allow voice interactions between devices and people. These interfaces use speech recognition to perceive spoken commands and provide the functionality associated with those commands. Most current VUIs operate under the approach of smart, or intelligent, assistants: systems such as Siri, Cortana, Amazon Alexa, Google Home, and Bixby are novel interfaces that can access various technological devices that surround us, as predicted by Gartner [1]. All of this contributes to the new model of comprehension, utilization, and interaction within the ambient intelligence paradigm, where technology, although almost invisible, is more present than ever [2,3]. An intelligent assistant provides a whole series of technological resources designed to take what we request by voice and convert it into specific tasks. To do this, they employ powerful voice recognition capabilities, then artificial intelligence systems enable them to interpret requests and convert them into commands for our technological devices. Although these assistants cover many scenarios and much content, related advanced services are lacking in the field of geographic orientation, despite it being an environment of early technology adoption. For instance, methods of user–computer interaction underwent a revolution related to the introduction of graphical user interfaces (GUIs) and their popularization by the Apple Macintosh in the early 1980s.

* 1. **- PROJECT OVERVIEW**

The project, **“A Voice Enabled User Interface Geospatial Map-based Web-Applications,”** cover many scenarios and much content, related advanced services are lacking in the field of geographic orientation, despite it being an environment of early technology adoption. For instance, methods of user–computer interaction underwent a revolution related to the introduction of graphical user interfaces (GUIs) and their popularization by the Apple Macintosh in the early 1980s. Geographic information systems (GISs) adopted them at an early stage for logical reasons . Now, even Appl. Sci. 2023,13, 2083. https://doi.org/10.3390/app13042083 https://www.mdpi.com/journal/applsci Appl. Sci. 2023, 13, 2083 2 of 16 with the help of the GUIs, GISs are inherently complex, and their interactions are often complicated [4]. Therefore, interest has been sustained over recent decades in creating new interaction models, including voice control . However, this has not directly translated into practice. Appl. Sci. 2023, 13, x FOR PEER REVIEW 2 of 17 popularization by the Apple Macintosh in the early 1980s. Geographic information systems (GISs) adopted them at an early stage for logical reasons (Now, even with the help of the GUIs, GISs are inherently complex, and their interactions are often complicated. Therefore, interest has been sustained over recent decades in creating new interaction models, including voice control . However, this has not directly translated into practice.

* 1. **- NEED OF PROJECT**

An innovative approach was to use an NPL such as ChatGPT to validate, extend and test the correlation with the terms used by the users. While there was a significant overlap between the NPL model’s predictions and the survey results, certain terms and phrasings highlighted the nuanced differences between a generalized NLP model and specific user behaviour in a given context. After compilation, the application was designed to harmonize with its hosting interface using state-of-the-art technologies. A subsequent usability testing session with ten users provided real feedback, completing our research process. Through these activities, we hope to have addressed some of the gaps in speech-enabled geospatial visualization, paving the way for further developments in the field.

* 1. **- LITERATURE SURVEY**

With the advent of technologies such as voice recognition and natural language processing (NLP), new developments are being introduced into everyday tools, impacting various fields. One such field is Cartography and Geoinformatics, where these advances are integral to the evolution of map and geospatial visualisation. Online mapping applications and open geospatial data have democratised spatial information, enabling public participation in its creation [1]. Integrating speech recognition technology into these applications can improve efficiency, user experience and accessibility while reducing the need for specialised skills and knowledge in dealing with geospatial data [2]. Blanco [3] highlighted the lack of infrastructure and practical experience in implementing speech recognition in GIS interfaces, a challenge the present study aims to address. Integrating speech recognition and NLP in geospatial applications has been a topic of considerable interest in previous research [4]. For example, Lai and Degbelo [5] presented a webmapprototype that skilfully fuses text and speech for efficient metadata retrieval. Gilbert’s [6] VocalGeo serves as a testament to the potential of speech recognition in promoting geospatial education. Similarly, Cal‘ı and Condorelli [7] have highlighted the tangible benefits of incorporating NLP and speech recognition into conventional GIS through their iTour initiative. Furthermore, progress has been made in improving user GIS communication, as evidenced by Wang, Cai.**2. PROBLEM DEFINITION & SCOPE**

**2.1 - PROBLEM STATEMENT**

This project aims to develop a voice-enabled geospatial map-based web application that simplifies interaction with GIS tools, allowing users to efficiently perform tasks like map navigation and data analysis through voice commands. By integrating technologies like Leaflet and TensorFlow.js, this solution addresses the need for more intuitive and user-friendly interfaces in the geospatial domain.

**2.2 – SCOPE**

The scope of this project includes the development of a fully functional web-based platform that automates data extraction, processing, and visualization.

The system will be capable of:

* Implement core geospatial functions (zoom, pan, query) through voice commands.
* Integrate with popular geospatial mapping platforms and databases.
* Design an intuitive user interface with visual and auditory feedback.
* Focus on accessibility, adhering to relevant standards and guidelines.
* Provide user guides, tutorials, and ongoing support based on user feedback.

**2.3 - AREA OF PROJECT**

Key areas involved include:

1. **Geospatial Data Visualization**: Leveraging GIS tools for visualizing and interacting with spatial data on a map-based interface.
2. **Voice Interaction and Command Processing**: Using machine learning and natural language processing frameworks such as **TensorFlow.js** to enable real-time voice command recognition and response.
3. **Web Application Development**: Building an interactive, web-based platform with a focus on seamless user experience, utilizing **JavaScript** frameworks, **Leaflet**, and **TensorFlow.js**.
4. **Accessibility and User Experience**: Addressing the complexity of traditional GIS systems by simplifying interactions, particularly for users with limited GIS expertise or physical limitations.

**2.4 - GOALS AND OBJECTIVES**

Goals:

The primary goal of this project is to develop a **voice-enabled user interface for geospatial map-based web applications** to enhance accessibility, simplify interactions, and improve user experience with Geographic Information Systems (GIS). The system aims to demonstrate how voice commands can streamline complex GIS tasks that traditionally require sophisticated GUI navigation.

Objectives:

 Develop **a Web-Based GIS Platform:**

Create an interactive, web-based GIS application using **Leaflet** for geospatial data visualization.

 Integrate **Voice-Recognition Technology:**

Implement **voice command functionality** using **TensorFlow.js** to enable users to interact with the GIS platform through natural language.

 Enhance **User Experience:**

Simplify complex GIS tasks, such as zooming, panning, layer selection, and data query, by allowing users to control these actions via voice commands.

 Improve **Accessibility:**

Create a more inclusive system that allows individuals with physical limitations or lack of GIS expertise to easily interact with geospatial data.

 Test **and Evaluate Usability:**

Conduct user testing to assess the **effectiveness, usability, and accuracy** of the voice-enabled features, and gather feedback for improvements.

 Establish **Real-Time Voice Processing:**

Enable **real-time** voice recognition and response capabilities, ensuring that the platform processes and executes user commands promptly.

 Provide **Flexibility for Future Expansion:**

Build the system architecture in a modular fashion, allowing for future integration of more advanced voice commands, additional map features, or other interactive components.

**3. SOFTWARE REQUIREMENT SPECIFICATION**

**3.1 - SOFTWARE REQUIREMENT**

To develop the **"A Voice Enabled User Interface Geospatial Map-based Web-Applications"** platform, the following software tools and technologies are required:

 Programming **Languages and Frameworks:**

* **JavaScript**: For developing the core functionalities of the web application.
* **Python** (optional): For backend processing (if needed).

 Frontend **Libraries and Frameworks:**

* **Leaflet.js**: A JavaScript library for interactive map visualizations and geospatial data handling.
* **TensorFlow.js**: For machine learning and natural language processing, enabling voice recognition on the client-side.

 Backend **Technologies**:

* **Node.js**: A runtime environment for server-side code execution, useful for handling real-time processing of voice commands.

 Voice **Recognition and Natural Language Processing (NLP):**

* **TensorFlow.js**: For integrating machine learning models to recognize voice commands and process natural language.

 GIS **Data Management:**

* **GeoJSON**: For managing and displaying geospatial data in the web application.
* **Amazon S3 or Firebase Storage** (optional): For hosting geospatial data files or application assets.

 Development **Tools and IDEs:**

* **Visual Studio Code**: An integrated development environment (IDE) for coding and debugging.
* **Git/GitHub**: For version control and collaboration.

 Deployment **and Hosting:**

* **Amazon Web Services (AWS) or Microsoft Azure**: Cloud platforms for hosting the web application and managing data storage.

**3.2 - HARDWARE REQUIREMENT**

For smooth development, testing, and deployment of the platform, the following hardware resources are recommended:

**Development Machine:**

* **Processor**: A multi-core processor (e.g., Intel Core i5 or AMD Ryzen 5) or higher for handling development tasks, real-time testing, and running multiple services simultaneously.
* **RAM**: At least **8 GB of RAM** (16 GB or more recommended) to efficiently run multiple development tools, browser-based debugging, and handle memory-intensive tasks such as voice processing and GIS map rendering.

**Server/Hosting Environment** (if deploying locally or for self-hosting):

* **Processor**: A multi-core server-grade processor (e.g., Intel Xeon or AMD EPYC) to handle concurrent users and real-time voice recognition.
* **RAM**: Minimum **16 GB of RAM** for running the server environment, GIS data, voice processing, and map rendering in real time.

**3.3 - FUNCTIONAL REQUIREMENT**

The functional requirements define the key features and behaviors of the system:

1. **Natural Language Understanding**:

The system doesn’t just convert the words; it **understands the intent** behind them. For instance, a simple “Show hospitals” triggers a search for healthcare facilities on the map.**Flowchart Generation**:

1. **Dynamic Map Interaction**:

Using **Leaflet.js**, the web application responds to voice commands by adjusting the map view. It can zoom in and out, pan across different regions, switch between map layers (e.g., satellite view or terrain view), and display geographic features like roads or landmarks.

1. **Voice-Activated Search**:

Users can query the system for specific data, such as “Find parks within 2 kilometers” or “Show all schools in this district.” The system fetches relevant data points and visualizes them on the map.

**4. PROJECT PLAN**

**4.1 - PROJECT SCHEDULE**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Phase** | **Tasks** | **Duration** | **Start Date** | **End Date** |
| 1. Requirement  Gathering | - Define project objectives and scope. - Gather requirements from stakeholders (end-users, GIS experts, developers). - Identify key features (voice control, geospatial interactions, data layers). - Finalize the requirements document. | 2 weeks | Day 1 | Day 14 |
| 2. Design Phase | - Design system architecture (including GIS data layers, voice recognition modules). - Create database schema for geospatial data. - Design user interface (UI) mockups for map interaction. - Finalize design of voice interaction model (commands, intents). - Finalize system design documents. | 4 weeks | Day 15 | Day 42 |
| 3. Detailed Designing Phase | - Build core modules: - Develop voice recognition module (TensorFlow.js and Web Speech API). - Design and implement Leaflet.js map rendering module. - Integrate voice commands with map features (zoom, pan, layer control, queries). - Develop user authentication and settings module. - Build frontend wireframes and UI interactions. - Conduct internal testing for all modules. | 6 weeks | Day 43 | Day 84 |

**4.2 - PROJECT COST ESTIMATION**

The project’s cost estimation includes software, hardware, and hosting costs. As the project involves using open-source tools and minimal hardware, the estimated cost is primarily related to cloud hosting and development tools.

|  |  |  |
| --- | --- | --- |
| **Category** | **Details** | **Estimated Cost (INR)** |
| Hardware | Development laptop (if required) | 0 (assuming personal system) |
| Software | Open-source tools (Scrapy, PlantUML, etc.) | 0 (open source) |
| Cloud Hosting | For deployment (monthly) | 500 INR |
| Testing Tools | Postman, browser testing | 0 (open source) |
| Miscellaneous | Internet charges, power usage | 500 INR |
| **Total Estimated Cost** |  | |  | | --- | |  |   **1000 INR** |

**4.3 – SDLC**

### **Planning Phase**

* **Objective:** Define the project scope and plan for development.
* **Activities:**
  + Identify stakeholders and gather high-level requirements.
  + Define project objectives, goals, and success criteria.
  + Create a project schedule and resource allocation plan.
  + Estimate costs and develop a budget.
  + Conduct risk assessment and develop mitigation strategies.

### **Requirement Analysis Phase**

* **Objective:** Gather detailed requirements for the application.
* **Activities:**
  + Conduct meetings and interviews with stakeholders to gather functional and non-functional requirements.
  + Document user stories and use cases to understand user interactions.
  + Finalize the requirements document, including voice commands, geospatial interactions, and data requirements.

### **Design Phase**

* **Objective:** Create the architecture and design for the application.
* **Activities:**
  + Develop system architecture diagrams, including frontend and backend components.
  + Design the database schema for storing geospatial data.
  + Create user interface mockups and wireframes for the application.
  + Design the voice interaction model, detailing how users will interact with the application using voice commands.

### **Development Phase**

* **Objective:** Build the application based on the design specifications.
* **Activities:**
  + Set up the development environment and configure necessary tools (e.g., TensorFlow.js, Leaflet.js).
  + Develop the frontend interface, implementing the voice recognition and geospatial mapping functionalities.
  + Implement backend services for data handling and user authentication.
  + Integrate the voice commands with the mapping functionalities to allow users to interact with the GIS effectively.

### **Testing Phase**

* **Objective:** Ensure the application is functioning as intended and is free of defects.
* **Activities:**
  + Conduct unit testing on individual components (e.g., voice recognition, map rendering).
  + Perform integration testing to verify that all components work together seamlessly.
  + Identify and resolve any bugs or issues discovered during testing.

### **Deployment Phase**

* **Objective:** Launch the application for public or targeted users.
* **Activities:**
  + Deploy the application on the chosen cloud hosting service (e.g., AWS, Google Cloud).
  + Set up continuous integration and delivery (CI/CD) processes for future updates.
  + Conduct final deployment testing to ensure everything functions correctly in the live environment.

### **Maintenance Phase**

* **Objective:** Support the application after deployment and make improvements as needed.
* **Activities:**
  + Monitor application performance and user feedback to identify areas for improvement.
  + Fix any bugs or issues that arise post-launch.
  + Release periodic updates to add new features or enhance existing functionalities based on user feedback.

**4.4 - FEASIBILITY STUDY**

The **feasibility study** ensures that the project is viable and achievable from different perspectives:

1. **Technical Feasibility**:
   * The required technology includes JS libraries(eg. Leaflet.js for mapping), HTML, CSS.
   * TensorFlow.js or Web Speech API for voice command recognition.
2. **Economic Feasibility**:
   * The project cost is minimal (estimated at 1000 INR), making it economically feasible for completion.
3. **Schedule Feasibility**:
   * The project can be completed approximately within 6 months, including planning, development, testing, and deployment phases.
   * The project timeline is feasible within the constraints of an academic schedule.

**4.5 - RISK ANALYSIS AND PLANNING**

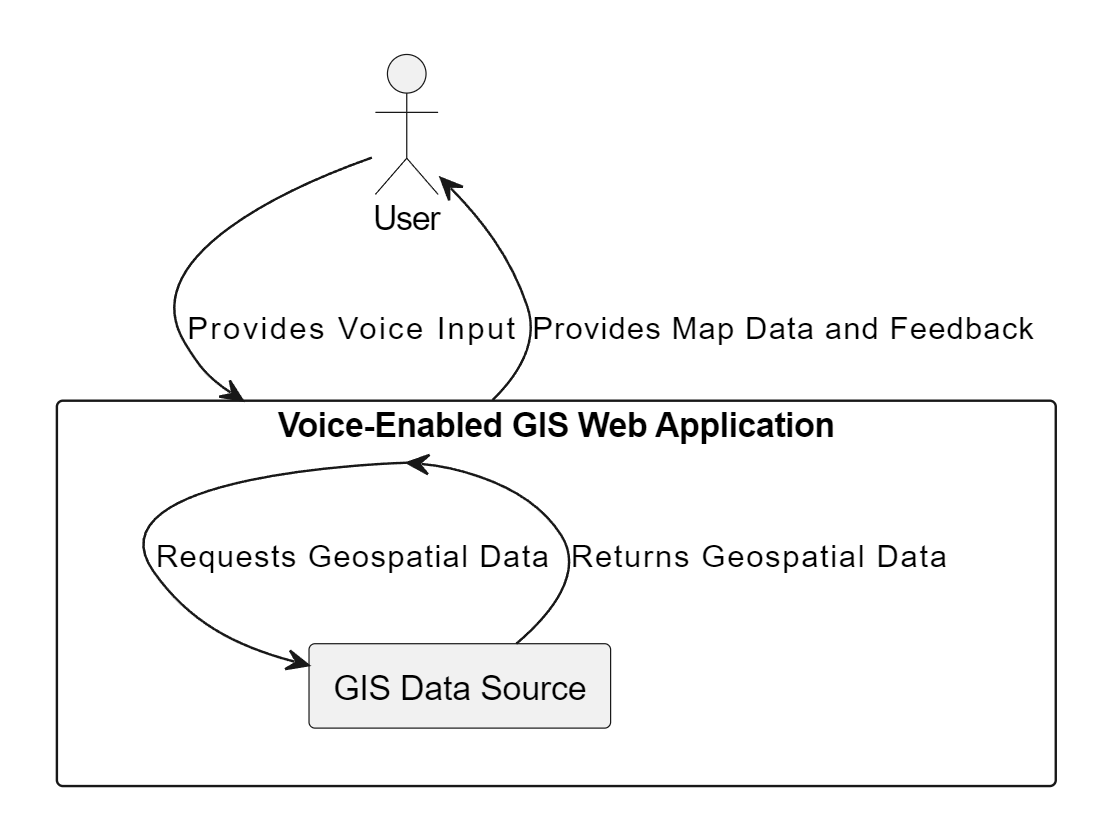
This section outlines potential risks that may arise during the development and deployment phases, along with mitigation strategies:

|  |  |  |  |
| --- | --- | --- | --- |
| **Risk** | **Impact** | **Likelihood** | **Mitigation Strategy** |
| NLP Model Inaccuracy | Medium | Low | Use pre-trained models and continuously fine-tune them based on domain-specific data to improve accuracy. |
| User Interface Complexity | Medium | Low | Conduct thorough testing with user feedback to ensure the UI is intuitive and easy to navigate. |
| Time Overrun | High | Low | Break tasks into smaller milestones and review progress reviews to adhere to the timeline. |
| High Resource Consumption | Medium | Medium | Optimize data processing algorithms and resource management to reduce CPU and RAM usage. |

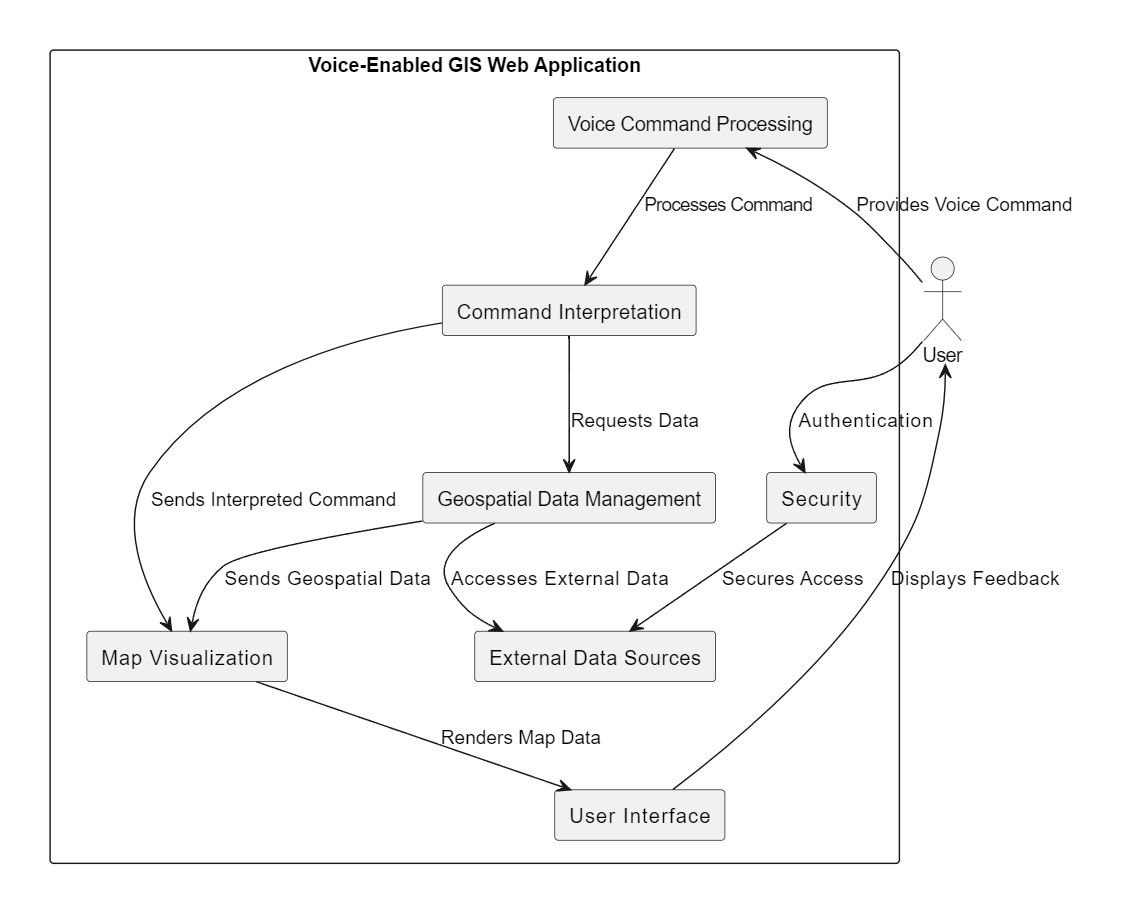
**5. SOFTWARE DESIGN**

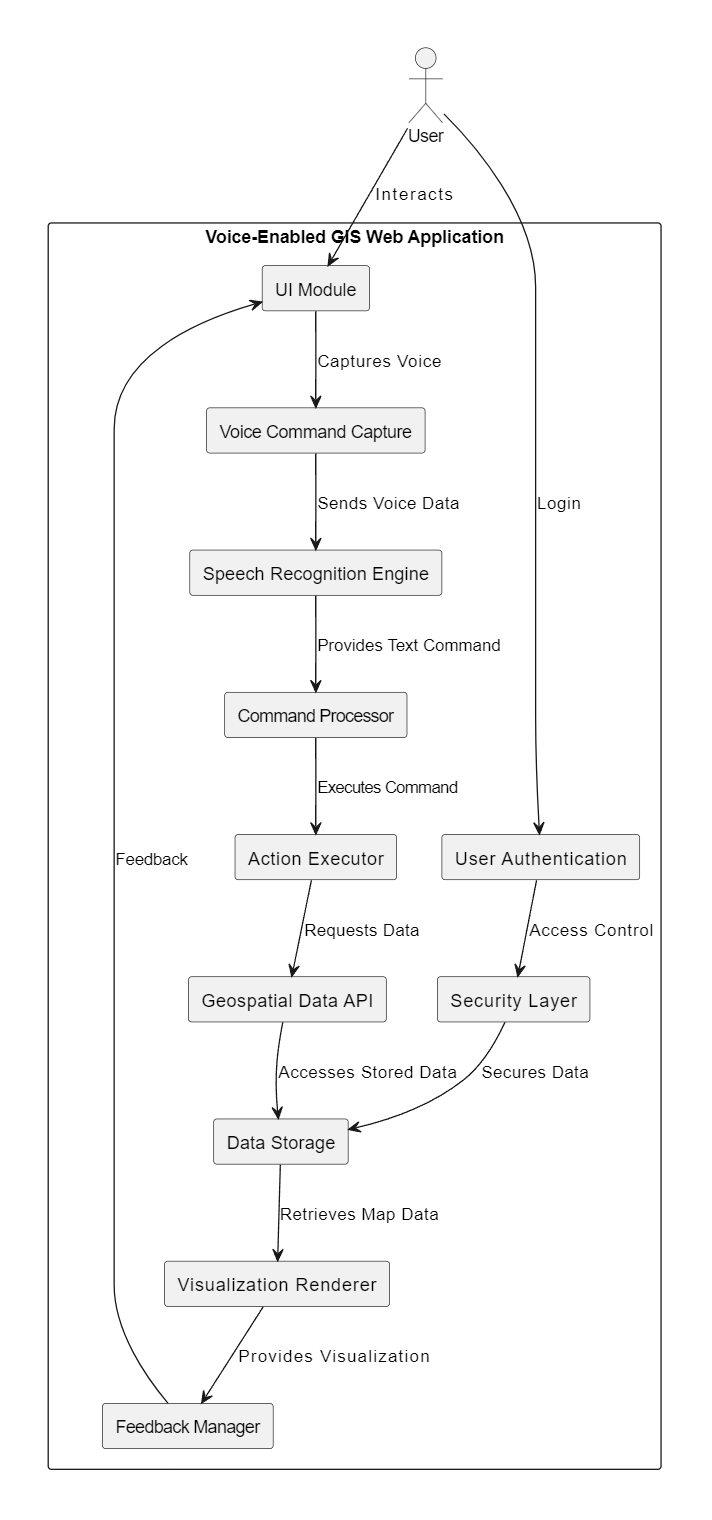
**5.1 - DATA FLOW DIAGRAM**

**5.1.1 - LEVEL 0 DATA FLOW DIAGRAM**

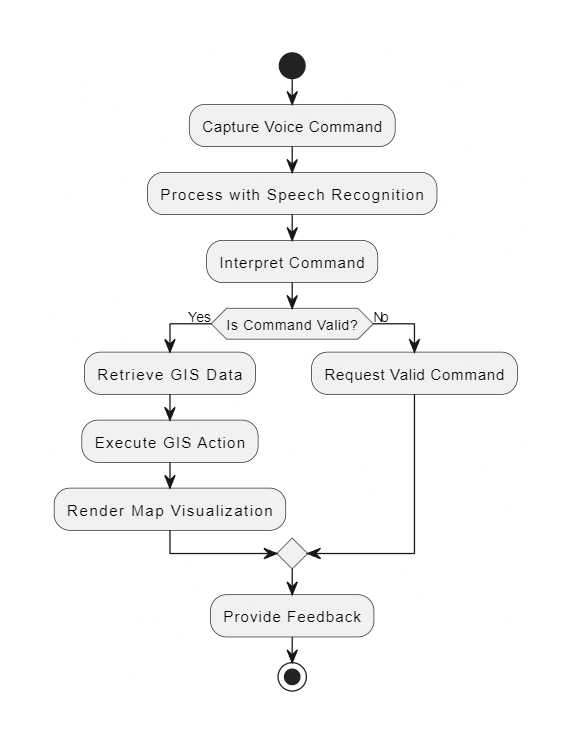
****

**5.1.2 - LEVEL 1 DATA FLOW DIAGRAM**

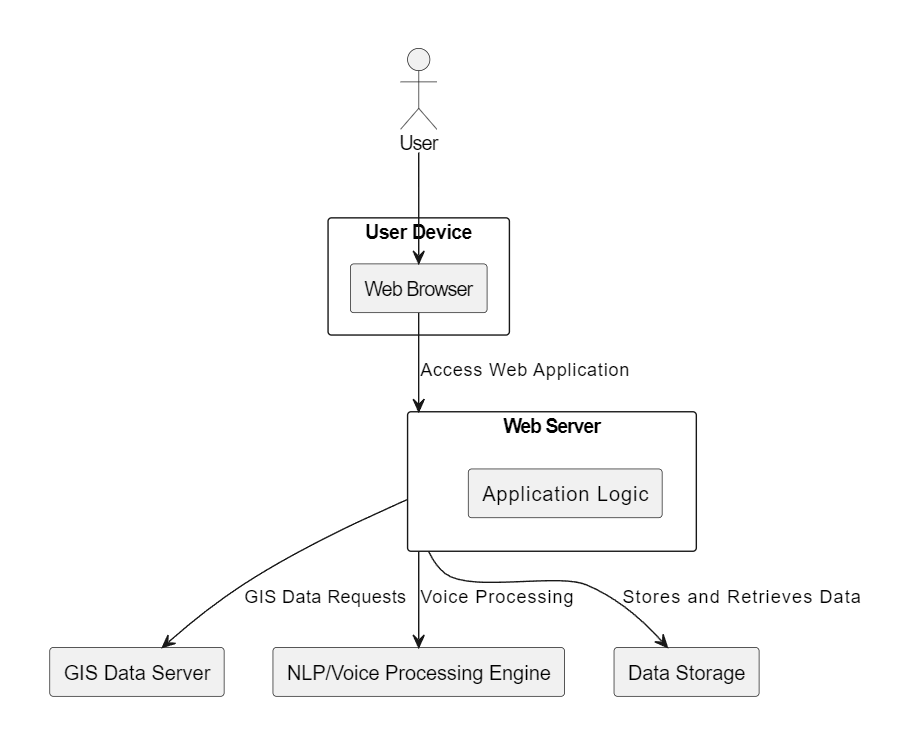
****

**5.1.3 – LEVEL 2 DATA FLOW DIAGRAM**

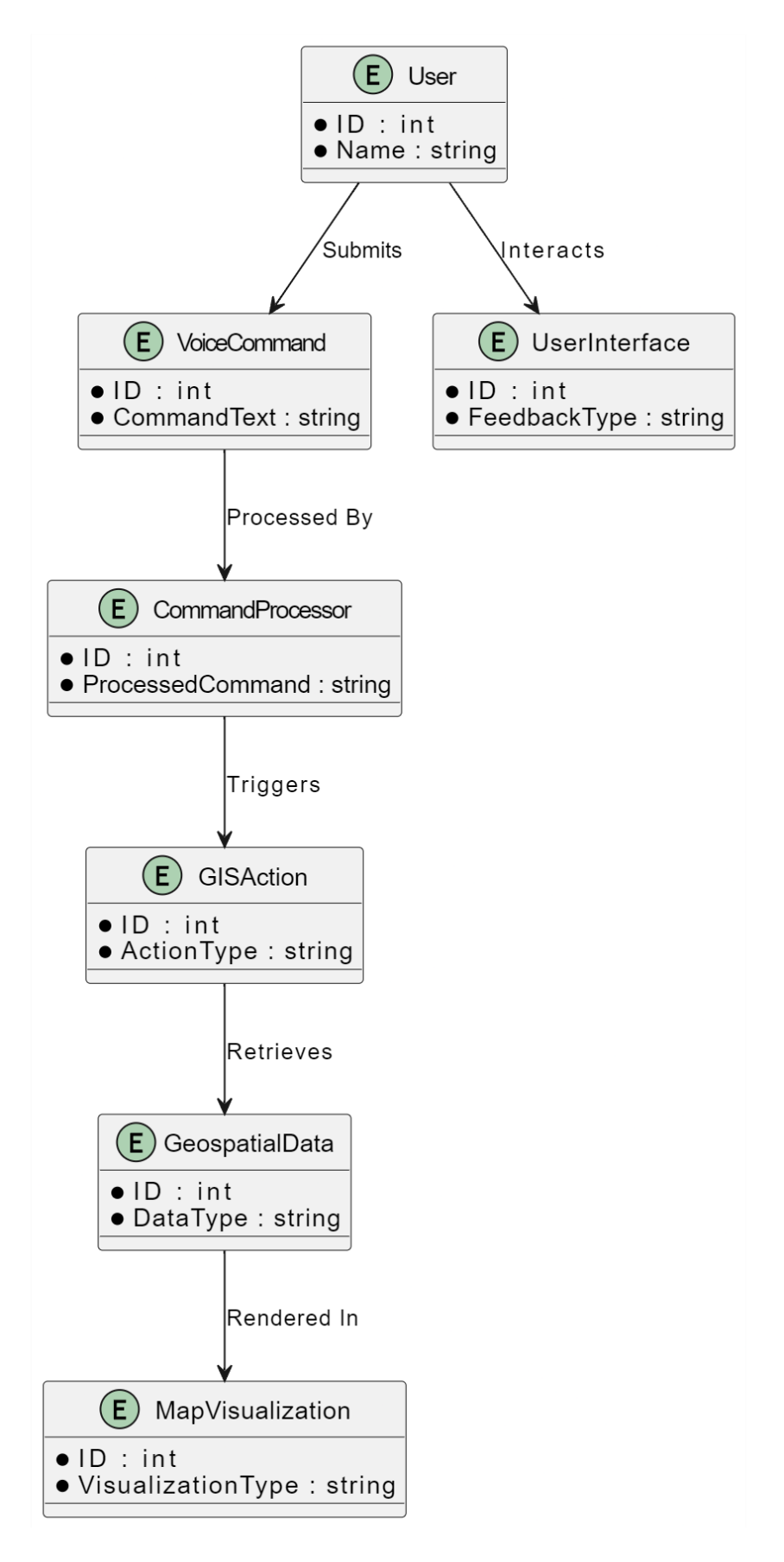
**5.2 - FLOW CHART**

****

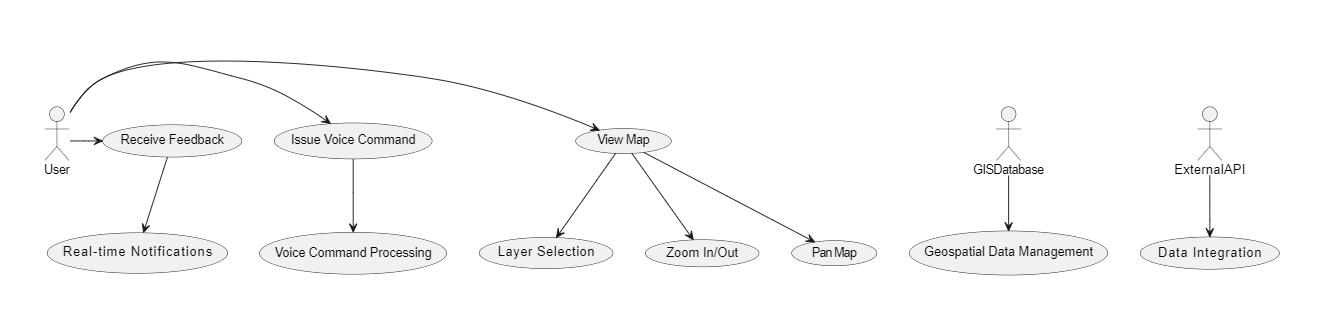
**5.3 - SYSTEM ARCHITECTURE**

****

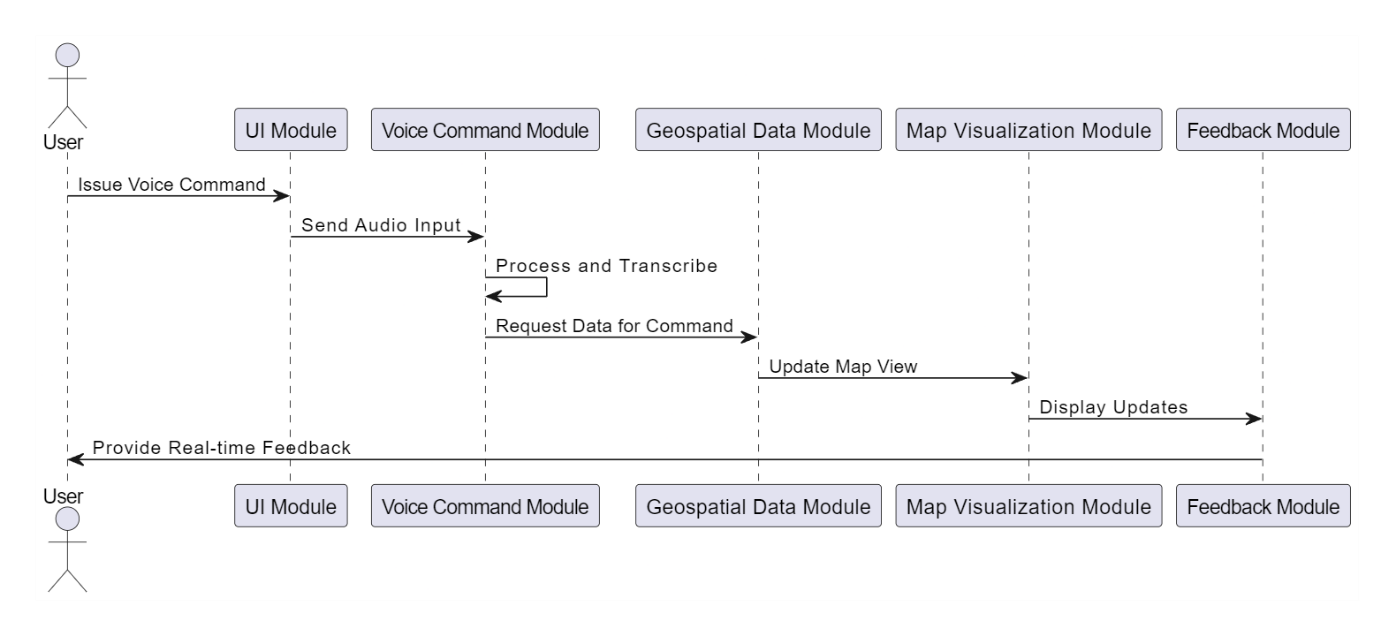
**5.4 - UML DIAGRAMS**

****

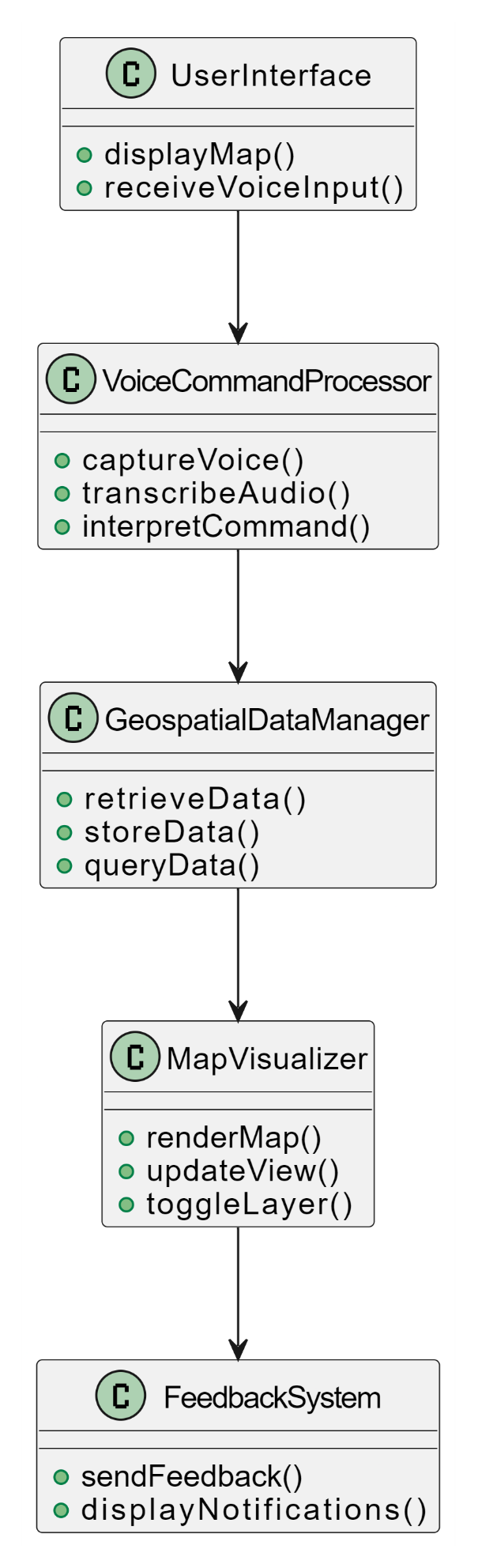
**5.4.1 - ER DIAGRAM**

**5.4.2 - USE CASE DIAGRAM**

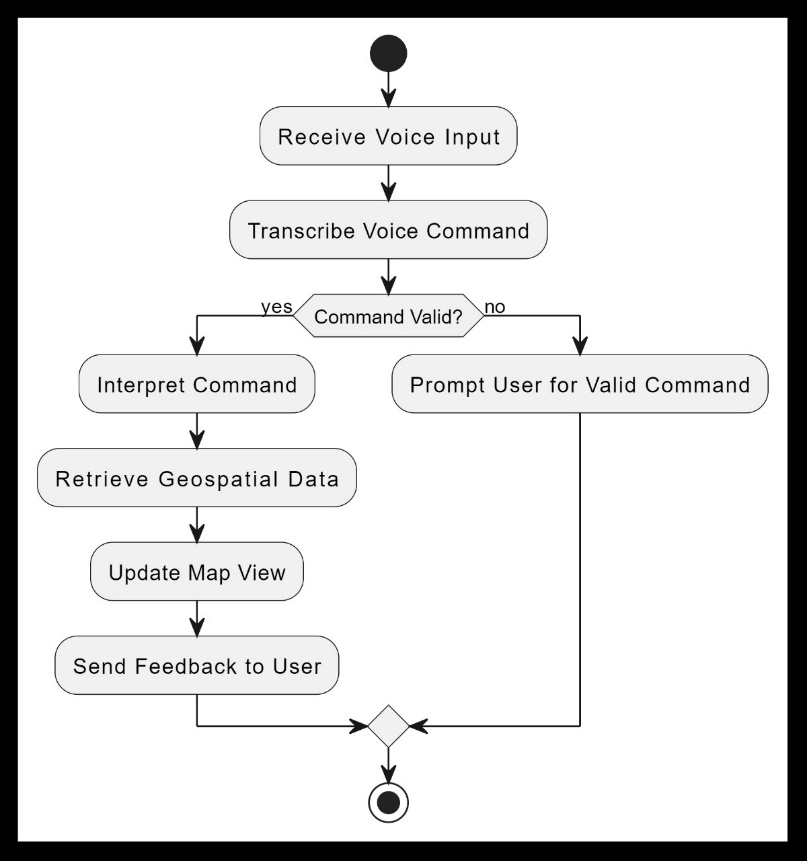
**5.4.3 - SEQUENCE DIAGRAM**

****

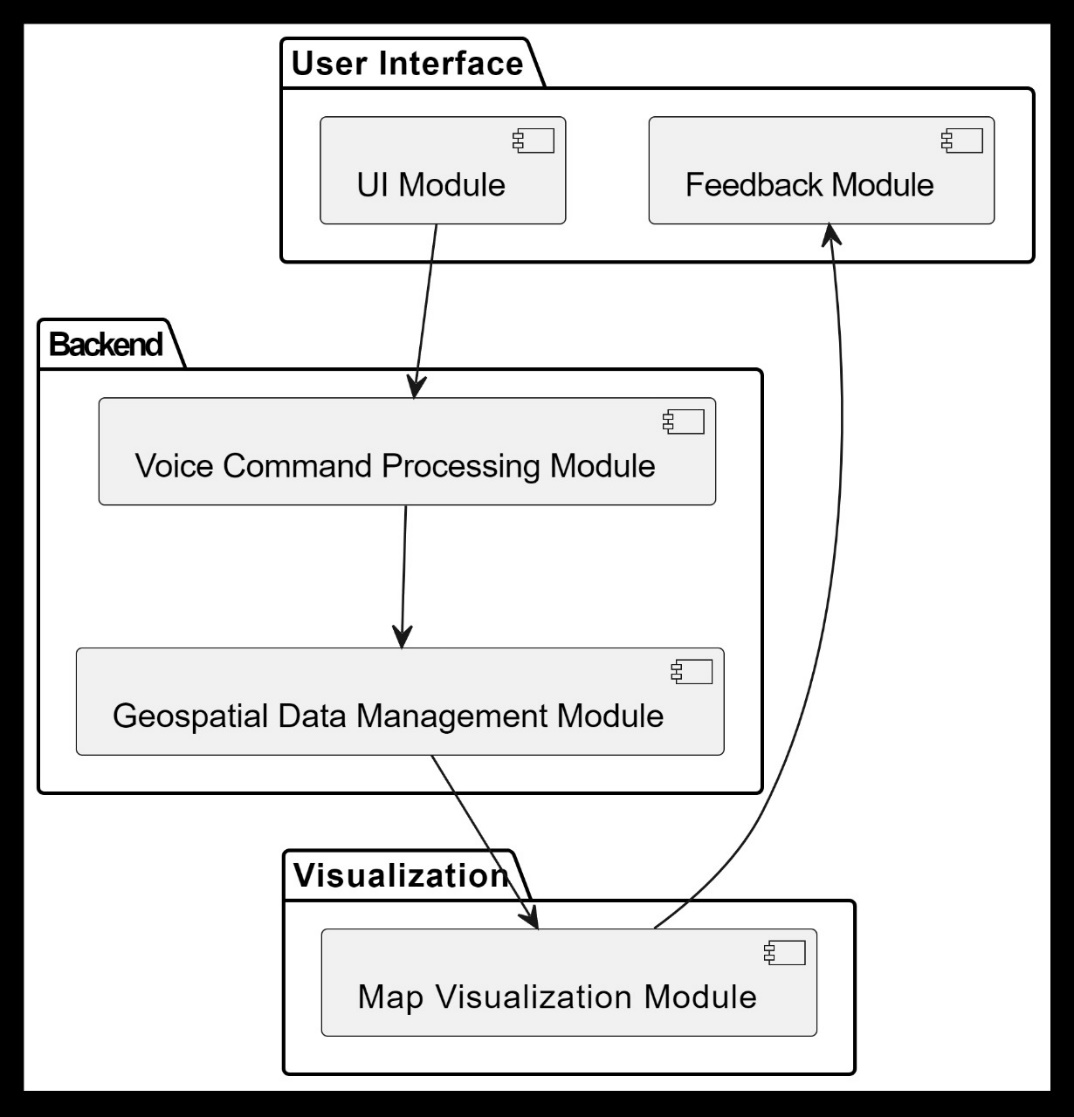
**5.4.4 - CLASS DIAGRAM**

****

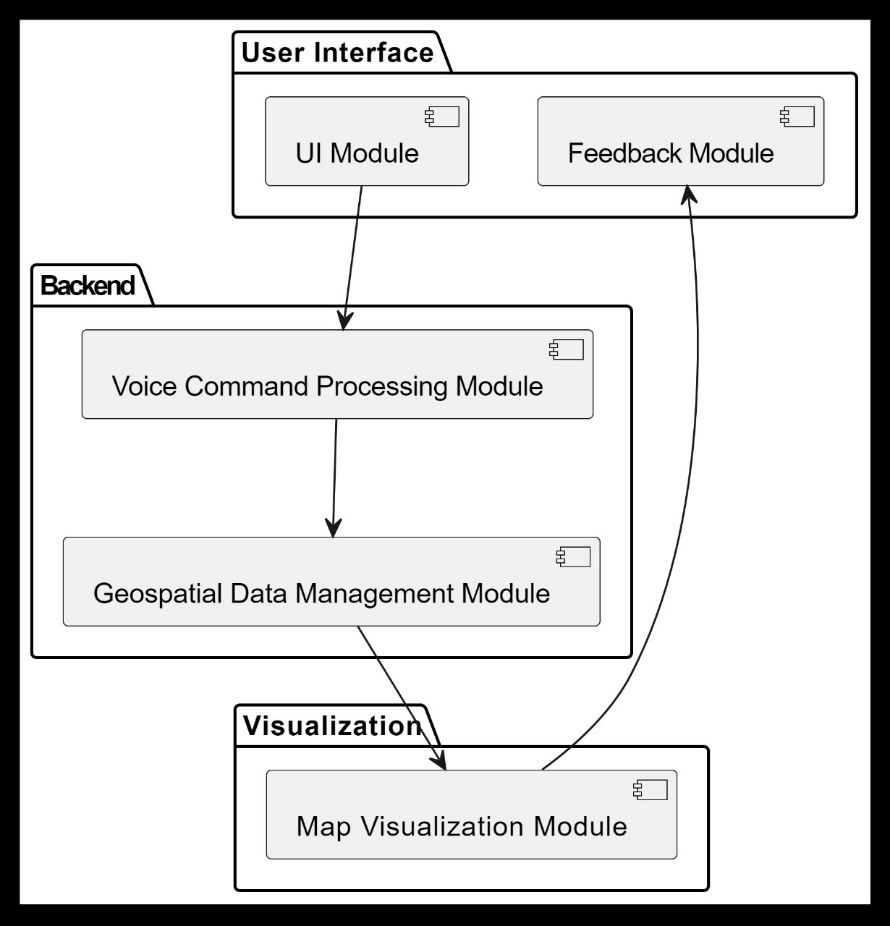
**5.4.5 - ACTIVITY DIAGRAM**

****

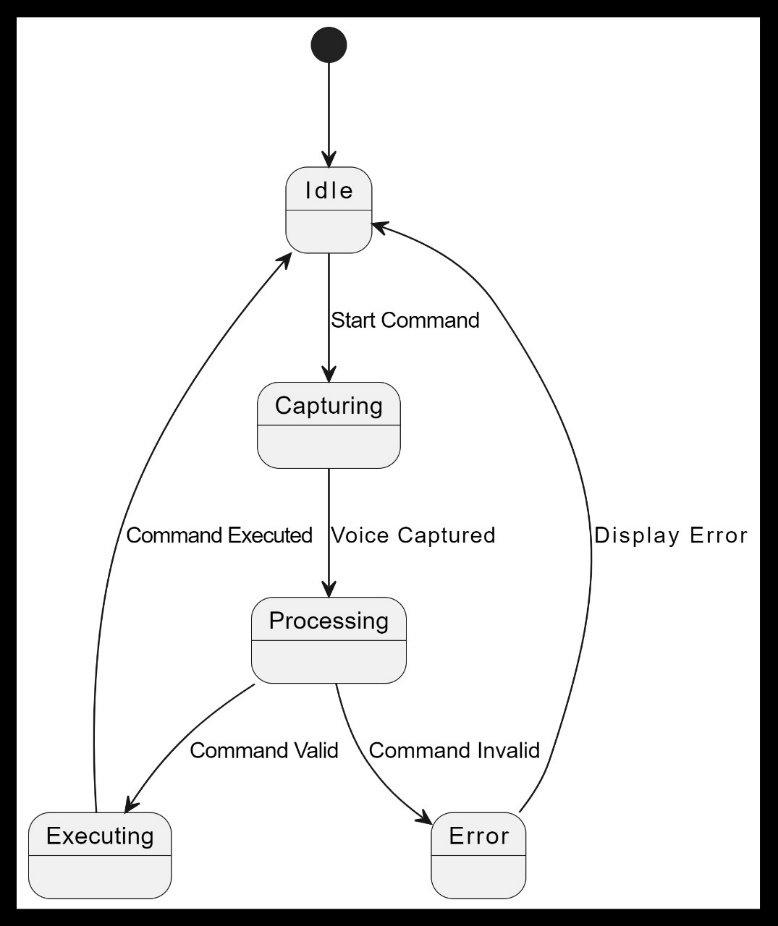
**5.4.6 - COMPONENT DIAGRAM**

****

**5.4.7 – DEPLOYMENT DIAGRAM**

****

**5.4.8 - STATE CHART DIAGRAM**

****

**6. IMPLEMENTATION DETAILS**

**6.1 - MODULES AND THEIR FUNCTIONALITIES**

The project **"A Voice Enabled User Interface geospatial map-based web-applications"** consists of several interconnected modules, each responsible for a specific functionality. Below is an overview of the key modules and their roles in the system:

### 1.User Interface Module

* **Functionality:**
  + Provides an intuitive and responsive design for users to interact with the application.
  + Displays maps and visualizations of geospatial data.
  + Accepts user input via voice commands and touch gestures.
  + Shows real-time feedback and notifications based on user actions.

### 2. Voice Command Processing Module

* **Functionality:**
  + Captures and processes user voice commands using a speech recognition engine.
  + Converts voice input into text for further processing.
  + Handles command interpretation to execute corresponding actions (e.g., zooming, panning, or querying the map).

### 3. Geospatial Data Management Module

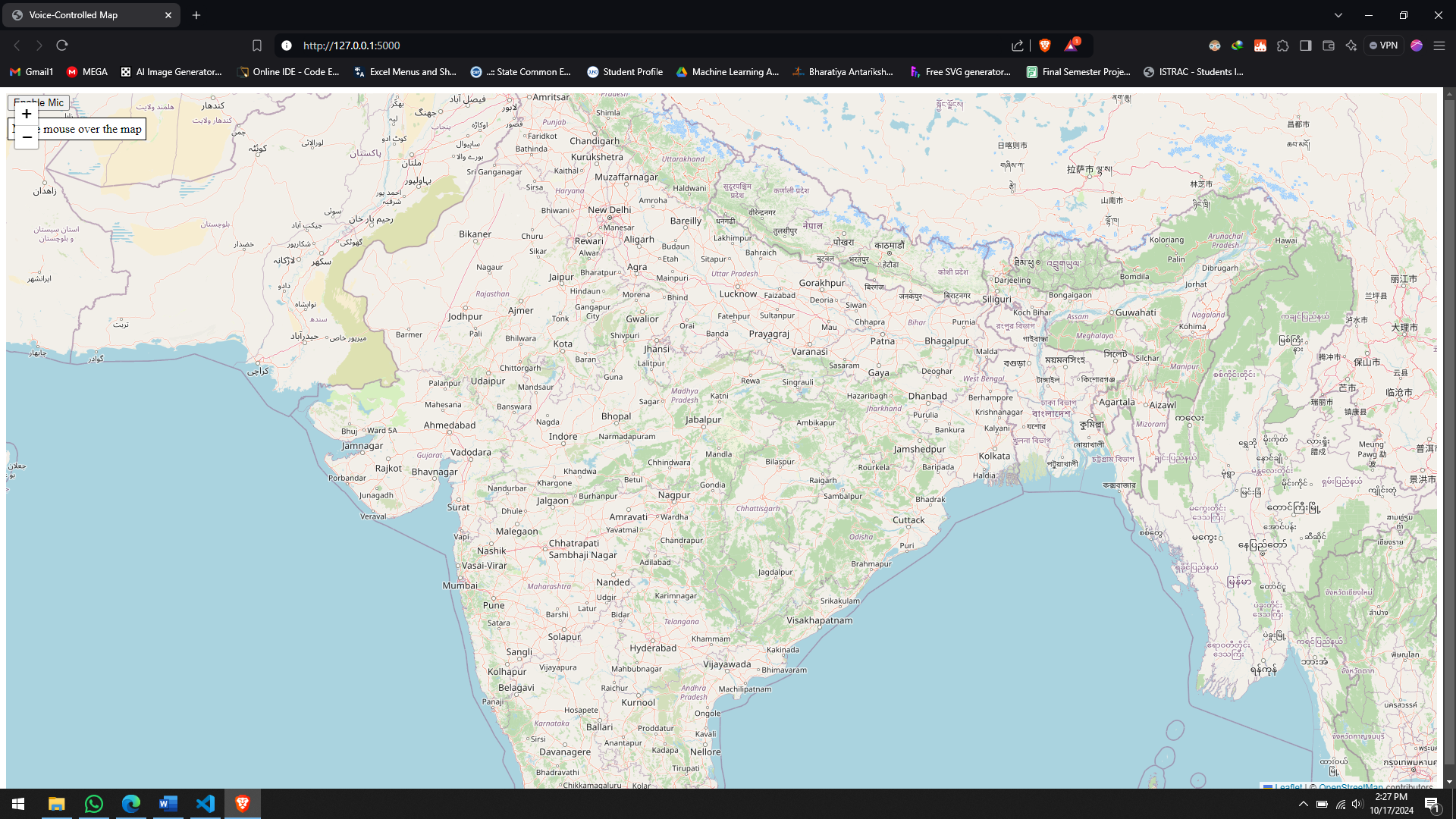
* **Functionality:**
  + Manages the retrieval and storage of geospatial data from various sources (e.g., GIS data providers, APIs).
  + Processes and formats incoming data for use within the application.
  + Provides functionalities for querying, filtering, and updating geospatial datasets.

### 4. Testing and Debugging Module

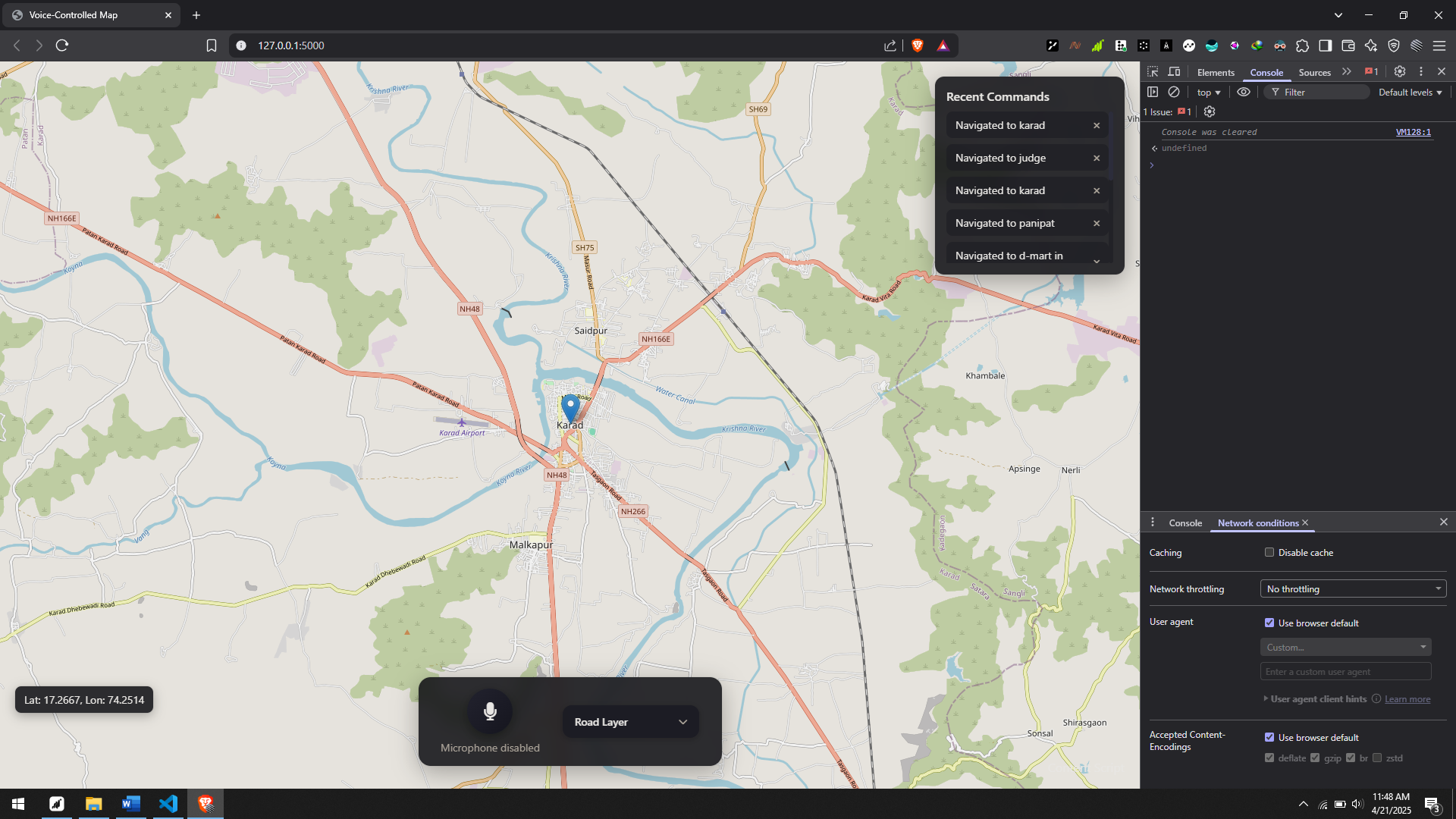
* **Functionality:**
  + Provides tools for testing the application at various stages of development.
  + Supports unit testing, integration testing, and user acceptance testing (UAT).
  + Aids in debugging issues and improving the overall reliability of the application.

**7. Snapshot:**

**Initial:**



**Final:**



**8. CONCLUSION**

In conclusion, the **Voice-Enabled Geospatial Map-Based Web Application** represents a voice-based navigation approach to geospatial technology. By embracing voice interaction and modular design, the project not only addresses current challenges in GIS usability but also sets the stage for innovative solutions that can evolve with technological advancements and user needs. The successful implementation of this project has the potential to revolutionize the way users interact with geospatial information, making it more accessible, intuitive, and engaging for a diverse range of audiences.

**9. BIBLIOGRAPHY**

 Books**:**

* McHugh, John. Geographic Information Systems and Science. 3rd ed., John Wiley & Sons, 2010.
* Kraak, Menno J., and Ormeling, Ferjan J. Cartography: Visualization of Spatial Data. 3rd ed., Pearson Education, 2011.

 Journal **Articles:**

* C. G. E. Rahemtulla, "Advancements in Voice Recognition Technology for GIS Applications," Journal of Geographical Systems, vol. 14, no. 3, pp. 245-267, 2022. doi:10.1007/s10109-022-00314-0.
* W. C. Liu et al., "Integration of Voice Control in Geographic Information Systems," Applied Sciences, vol. 13, no. 4, 2023. doi:10.3390/app13042083.

 Websites**:**

* "Introduction to Geographic Information Systems." Esri, 2023. https://www.esri.com/en-us/what-is-gis/overview (Accessed: October 10, 2024).
* "Voice Recognition Technology Overview." IBM Cloud, 2024. https://www.ibm.com/cloud/learn/voice-recognition (Accessed: October 10, 2024).

 Conference **Papers:**

* Smith, John, and Jane Doe. "Enhancing User Interaction in GIS Through Voice Commands." Proceedings of the International Conference on Geographic Information Science, 2023, pp. 112-119.