



Fall 2024
ENGI 9837 - Software Engineering Capstone
Project Requirements/Specification

Radar Data Viewer

Project Supervisor

Reza Shahidi

Group Members

1. Eialid Ahmed Joy - 202384815
2. Monirul Islam - 202381823
3. Jakaria Hossain - 202293102
4. B M Al-amin Badhon - 202384559
5. Sadia Noshin Shuchi - 202380413

Submission Date

October 01, 2024

Overview

The Radar Data Viewer Project will be a web-based software designed to visualize high-frequency (HF) radar and X-Band radar data. Initially, it will focus on displaying radar data files in a B-Scan format, with plans to expand to multiple files and animations in the future. The web application will also show key details such as when and where the radar data was collected, the transmission frequency, bandwidth, and the maximum range. Although the first version of the software will have limited functionality, it will be accessible from different operating systems like Windows, Ubuntu, etc. In addition to that, we plan to integrate advanced data analysis features, including spectrum and parameter extraction algorithms, in future versions of the software.

Objective

Our goal is to develop a user-friendly and easy-to-use web application for visualizing HF and X-Band radar data, which will allow users to easily upload, view, and analyze radar data files, making it easier for users to understand and work with radar data.

Functional Requirements

1. Data Loading and Visualization

- a. HF Radar Data:
 - i. The software should be able to load HF radar data stored in SORT files.
 - ii. Visualize the loaded HF radar data in B-Scan format as part of the MVP.
- b. X-Band Radar Data:
 - i. It should also support loading X-Band radar data files.
 - ii. Visualize these X-Band radar files in B-Scan format for the MVP.
- c. Future Enhancements:
 - i. In the future, support for scan-converted formats will be added.
 - ii. Enable visualization of multiple consecutive radar data files, either displayed one by one or as an animation.

2. Display Key Data Parameters

When the radar data will be loaded and visualized, the following key information should be displayed:

- a. Date and time when the data was captured.
- b. Transmission frequency and bandwidth (particularly important for HF radar data).
- c. Location where the radar was situated when the data was acquired.
- d. The maximum range the radar data covers.

3. Cross-platform Compatibility

- a. Design Consideration: The software should be built in a way that makes it easy to adapt to different operating systems, such as Windows 11, Ubuntu, and others.

Note: Cross-platform compatibility is not required for the MVP, but the design should ensure that expanding to multiple platforms in the future will be easy.

4. Augmentation with Wave Spectrum and Parameter Extraction

- a. Design Consideration: The software should be structured to allow future integration of wave spectrum analysis and parameter extraction algorithms.

Note: These features need not be implemented in the MVP, but the system design should accommodate their easy addition in later stages.

Non-functional Requirements

1. Usability

The software should be easy to use, with a clear and straightforward interface. Users should be able to quickly load radar files, see visualizations, and find key data without any confusion.

2. Performance

The software needs to work efficiently, even when dealing with large radar files. Users should be able to load and view radar data without experiencing long wait times or slowdowns.

3. Extensibility

The system should be flexible and designed in a way that makes it easy to add new features in the future, like advanced analysis tools or different ways of displaying the radar data, without needing to overhaul the whole system.

Minimum Viable Product (MVP)

1. Core Features:

- a. A single HF radar file needs to be displayed in B-Scan Format. It would be good if the system could display multiple consecutive files either separately or as an animation.
- b. Display key parameters of a data file such as the information of data acquisition, transmission frequency, bandwidth, particularly for HF radar, and location of the data for each one.
- c. The system should primarily run on the web. However, If the system architecture is designed properly, it is possible to run this in major OS like Windows, Linux, etc.

- d. The system architecture should be able to handle algorithms with various wave spectrums and parameter extraction.
2. Out of Scope for Minimum Viable Product (MVP):
- a. Working with Both HF radar data files (in SORT files) and X-Band Radar Data, visualizing them in different formats simultaneously, such as B-Scan and scan-converted formats, is not required.
 - b. For now, the software is not required to run on cross-platforms such as Windows, Linux, etc
 - c. Implementation of various wave spectrums and parameter extraction algorithms is not required now.

Technology

1. Backend:
- a. Python, due to its strong support for data processing and scientific computing.
 - b. Django/Flask: A lightweight and flexible web framework for handling and processing radar data requests.
2. Frontend:
- a. React.js: A powerful JavaScript library for building an interactive and responsive user interface.
3. Data Processing Libraries:
- a. NumPy and Pandas: For efficient data manipulation and analysis.
 - b. Matplotlib: For visualizing radar data in both B-Scan and scan-converted formats.
4. Communication between Frontend and Backend:
- a. RESTful APIs: To ensure smooth and efficient data exchange between the frontend and backend.
5. Cross-Platform Accessibility:
- a. Web-based Application: Accessible on Windows, Linux, and macOS via web browsers, ensuring platform independence.
6. Scalability:
- a. The technology stack is designed for future enhancements, such as adding animation features and integrating wave spectrum analysis algorithms.

Time Milestone

The project timeline for the Radar Data Viewer is outlined as follows:

Total Duration: 8 weeks (2 months)

1. UI/UX Design:
 - a. Duration: 1 week
 - b. Focus: Designing the visual layout and overall user experience.
2. Front-End Development:
 - a. Duration: 3 weeks
 - b. Focus: Implementing the user interface using React.js to ensure an interactive and responsive experience.
3. Back-End Development:
 - a. Duration: 2 weeks
 - b. Focus: Building the core functionality, radar data processing, and API development using Django/Flask.
4. Testing:
 - a. Duration: 1 week
 - b. Focus: Conducting thorough testing of the system to ensure stability and performance.
5. Buffer Time:
 - a. Duration: 1 week
 - b. Purpose: Allowing for unexpected delays or challenges.

Conclusion

The Radar Data Viewer will offer an intuitive web platform for visualizing and analyzing HF and X-Band radar data. Initially, we will focus on the Minimum Viable Product, enabling users to view single radar files in B-Scan format and key metadata. Over time, we plan to expand the capabilities of our project by implementing more advanced visualization techniques and modular integration for wave spectrum analysis and parameter extraction.