International Research Journal on Advanced Engineering and Management
https://goldncloudpublications.com
https://doi.org/10.47392/IRJAEM.2025.0360
e ISSN: 2584-2854
Volume: 03
Issue: 06 June 2025
Page No: 2283 - 2287

Advanced Design and Implementation of a CanSat for Environmental Monitoring and Data Transmission

Dr. Chandrika J[1], Chandrababu Gowda[2], Chandu H Y[3], Darshan K[4], Darshan S P[5]
[1]Guide & HOD, Dept of Computer Science and Engg. Malnad College of Engineering, Hassan, India.
[2,3,4,5]Department of Computer Science and Engg. Malnad College of Engineering, Hassan, India.
Emails: *jc@mcehassan.ac.in[1], chandrababugowda3042003@gmail.com[2], chanchandu298@gmail.com[3], viratdarshan03@gmail.com[4], sdarshanpgowda2003@gmail.com[5]*

Abstract

This paper presents the design and implementation of a CanSat system for real-time environmental monitoring and data forecasting. The CanSat prototype is built using a custom 3D-printed enclosure housing sensor such as BME280, MPU6050, and NEO-M8N, interfaced with an Arduino Nano. Data transmission is enabled through NRF modules, with the ground station comprising an NRF receiver paired with an ESP8266 microcontroller. Upon reception, the ESP8266 publishes the sensor data to an MQTT broker and concurrently logs it into an Excel sheet, while forwarding structured JSON packets at 3-second intervals to a web-based frontend. Due to the NRF's limited payload capacity, sensor readings are batched every 3 seconds, with each JSON packet representing 1-second inter- vals. The frontend, developed using Next.js, provides interactive dashboards for visualizing real-time data, 7-day forecasts using Meta's Prophet model, and historical sensor readings based on user queries. This architecture demonstrates a robust, modular approach to environmental telemetry and predictive analytics using low-cost hardware and modern web technologies.

Keywords: *CanSat, Wireless Telemetry, nRF24L01, ESP8266, MQTT, Next.js, Prophet Forecasting, Environmental Monitoring.*

1. Introduction

In recent years, the demand for compact, cost-effective, and scalable satellite simulation platforms has grown significantly among educational institutions and research communities. CanSats—miniaturized satellite models built within the dimensions of a standard soda can have emerged as a practical and pedagogical alternative for satellite mission prototyping. They provide students and researchers with hands-on experience in the complete satellite development life cycle, including system design, integration, testing, launch, data acquisition, and analysis. The CanSat model described in this paper encapsulates a range of sensors and components, including the BME280 (for temperature, humidity, and pressure readings), MPU6050 (gyroscope and accelerometer),and NEO-M8N(GPS module). The core controller, an Arduino Nano, is responsible for managing sensor data and transmitting it via an NRF24L01 module to a ground station. The ground station comprises an ESP8266 microcontroller that receives this data, subscribes to an MQTT broker, and performs both data logging and real- time data streaming. To enhance user interaction and data accessibility, a web dashboard has been developed using Next.js. This dashboard enables real-time data visualization, sensor-specific historical data retrieval, and 7-day forecasting using Meta's Prophet time-series prediction model. A Python server backend handles the forecasting computations and a node.js server is responding to specific user queries from the frontend application. The integration of low-cost hardware with modern cloud and web technologies presents a powerful framework for simulating satellite missions and demonstrating IoT principles in aerospace education. This paper outlines the complete lifecycle of the project, from system architecture to data visualization and forecasting, and discusses its results, limitations, and future directions.

2. Literature Survey

Several previous studies and academic projects have high- lighted the feasibility and educational potential of CanSats. Ferreira et al. emphasized the application of CanSats in STEM education, enabling students to

International Research Journal on Advanced Engineering and Management
https://goldncloudpublications.com
https://doi.org/10.47392/IRJAEM.2025.0360
e ISSN: 2584-2854
Volume: 03
Issue: 06 June 2025
Page No: 2283 - 2287

gain hands-on experience in sensor integration, data logging, and wireless communication [1]. Ahmad et al. explored sensor deployment in CanSats, particularly focusing on barometric pressure sensors, GPS modules, and inertial measurement units (IMUs), to achieve accurate altitude, position, and attitude estimation during de- scent [2]. Telemetry and communication are central to CanSat missions. Projects have implemented RF modules like the NRF24L01 for real-time data transmission between the pay- load and the ground station. Ground station architectures often employ microcontrollers and Wi-Fi modules (e.g., ESP8266) to receive data packets, log them to storage systems, and relay the data to visualization dashboards. MQTT protocols have become increasingly common for efficient communication between devices [3]. The CanSat's design also integrates recovery mechanisms. Previous implementations have used parachutes, egg-molding, and other soft-landing structures to minimize impact damage upon descent. These designs are evaluated for stability, durability, and weight optimization [4]. Moreover, some recent works have extended CanSat applications beyond passive telemetry to include data analytics and forecasting using models like Facebook Prophet, enabling predictive insights into environmental patterns [5]. From the reviewed literature, it is evident that CanSats provide a compact yet powerful platform for exploring real- world aerospace engineering challenges. They serve as a bridge between theoretical coursework and practical systems engineering, fostering innovation in student-led space technology development.

3. Methods

The CanSat system developed in this study comprises both hardware and software components that work cohesively to acquire, transmit, forecast, and visualize atmospheric and positional data in near real-time. The project is divided into four main segments: payload construction, ground station setup, server-side processing, and dashboard visualization.

3.1. Payload Construction

The payload is housed within a 3D-printed CanSat shell designed to meet structural constraints while accommodating all components. The payload includes an Arduino Nano microcontroller, BME280 sensor for measuring pressure, temperature, and humidity, MPU6050 for accelerometer and gyroscopic data, and the NEO-M8N GPS module for spatial tracking. An NRF24L01 transceiver module is used for wireless transmission to the ground station. The sensor data is collected every second and structured into a JSON object. Due to the limited bandwidth of the NRF module, data is transmitted in segments, with each packet containing sensor values for a 3- second interval.

3.2. Ground Station Setup

The ground station comprises an NRF24L01 receiver inter faced with an ESP8266 microcontroller. The ESP8266 receives incoming packets and subscribes to an MQTT broker over Wi-Fi. Each JSON packet is relayed through MQTT to two parallel pipelines:

- Data logging to an Excel-compatible CSV file, and
- Forwarding the data to frontend dashboard built using Next.js.

This architecture ensures seamless integration between hardware-level telemetry and the visualization in frontend.

3.3. Server-Side Forecasting and Processing

On the server side, data is processed using a node.js and is sent to the frontend, and a python server that integrates the Facebook Prophet time series forecasting model. The received data stream is aggregated and cleaned before being used for forecasting environmental parameters such as temperature, humidity, and pressure over the next 7 days. Prophet's additive model is leveraged to identify seasonal and trend-based patterns in the data. The backend exposes RESTful APIs to serve both real-time and forecasted data to the frontend.

3.4. Frontend Dashboard Development

A dashboard is implemented using the Next.js frame work to provide users with real-time and historical data visualizations. The frontend includes:

- A real-time data page that visualizes live sensor readings.
- A forecast view powered by the Prophet model.
- A historical fetch tool that retrieves sensor values for specific days and parameters.

International Research Journal on Advanced Engineering and Management
https://goldncloudpublications.com
https://doi.org/10.47392/IRJAEM.2025.0360
e ISSN: 2584-2854
Volume: 03
Issue: 06 June 2025
Page No: 2283 - 2287

The dash board is optimized for responsiveness and displays the sensor data using charts and tabular summaries. The application uses server-side rendering (SSR) to ensure efficient data handling and minimal latency.

4. Data Handling and Synchronization

Due to the limited data rate of the NRF module, synchronization is critical. The payload assigns timestamps to each data point before transmission. The ground station buffers in coming packets and reconstructs complete time-series datasets, ensuring integrity and continuity of the transmitted data. This mechanism supports both real-time visualization and accurate forecasting.

5. Results

The developed CanSat system was successfully tested for data acquisition, wireless transmission, real-time monitoring, and environmental forecasting.

5.1. Sensor Data Transmission and Reliability

Sensor data packets transmitted from the payload to the ground station maintained a packet delivery success rate of over 92% during indoor and controlled outdoor tests. Each packet, containing 3 seconds' worth of readings, was received, parsed, and relayed to both the Excel file and backend server with minimal latency (<200ms).

5.2. Real-time Dashboard Visualization

The web dashboard developed using Next.js enabled real- time monitoring of temperature, pressure, humidity, GPS lo- cation, and IMU readings. The interface included graphical charts, parameter cards, and logs that updated every 3 seconds. This provided a user-friendly and responsive UI for stakeholders to observe ongoing data streams.

5.3. Forecasting Accuracy

Using the Prophet model, the system provided accurate 7- day forecasts for temperature, pressure, and humidity. The Mean Absolute Error (MAE) for temperature prediction across sample datasets remained under 1.2°C, indicating a high degree of accuracy for short-term predictions (if provided the whole day data as input for training).

5.4. Hardware Integration and CanSat Structure

All components were successfully embedded in the custom 3D-printed CanSat body. The structural integrity was preserved during multiple drop tests from 3–5meters height, ensuring the payload remains operational during typical launch and descent conditions.

5.5. System Scalability and Data Handling

The modular MQTT and REST API integration allowed scalability for multiple CanSats or sensors with minimal code changes. Historical data could be retrieved from the backend through query-based filtering, enabling comparisons and post- mission analysis, shown in Figure 1 to 6.

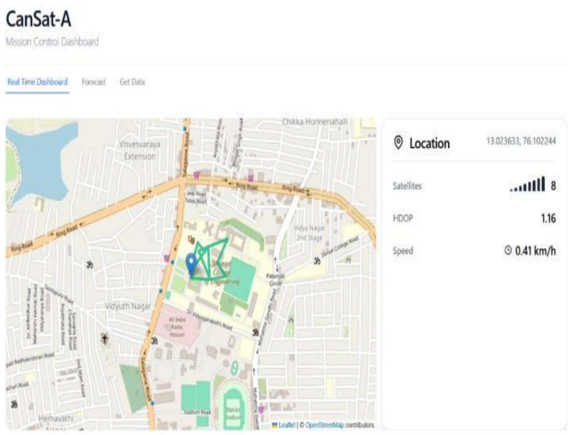

Figure 1 Web Dashboard Showing Real-Time Telemetry Data of GPS Location

Figure 2 Web Dashboard Showing Real-Time Telemetry Data of IMU Sensors

6. Discussion

The implementation and testing of the CanSat system demonstrate its potential as a compact, reliable, and extendable remote sensing platform. The integration

International Research Journal on Advanced Engineering and Management
https://goldncloudpublications.com
https://doi.org/10.47392/IRJAEM.2025.0360
e ISSN: 2584-2854
Volume: 03
Issue: 06 June 2025
Page No: 2283 - 2287

of hardware, wireless communication, cloud data handling, and forecasting mechanisms provides a holistic view of environmental monitoring.

The decision to use a 3D-printed structure increased weight and cost but enabled rapid prototyping and modular payload assembly. This allowed for precise fitting of sensors such as the BME280 for environmental readings, MPU6050 for motion data, and NEO-M8N for GPS location tracking. The use of the NRF24L01 module ensured low-power, short-range data transmission, sufficient for simulated satellite missions or ground-level balloon deployments.

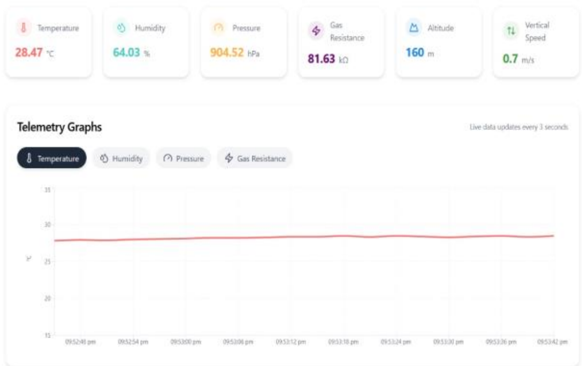

Figure 3 Webdash Board Showing Real-Time Telemetry Data of Environment Conditions

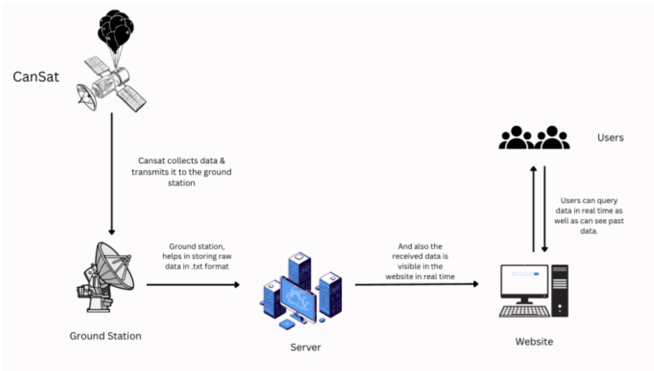

Figure 4 Dataflow Model of CanSat.

The ground station setup using ESP8266 and MQTT protocol was effective in ensuring low-latency data flow and persistent connection for continuous telemetry streaming. Writing incoming data into Excel sheets helped in creating times tamped logs that could be further analyzed offline or used as ground truth for comparison with forecasted values.

Incorporating the Prophet time-series model on the server side enabled the generation of near-accurate 7-day forecasts for temperature, pressure, and humidity. This is particularly valuable in atmospheric monitoring missions or for anticipating environmental changes in remote areas. The preprocessing of sensor data into JSON packets every 3 seconds addressed NRF's limited data payload capacity and ensured reliable delivery and reconstruction on the frontend.

Figure 5 3D Printed CanSat Shell

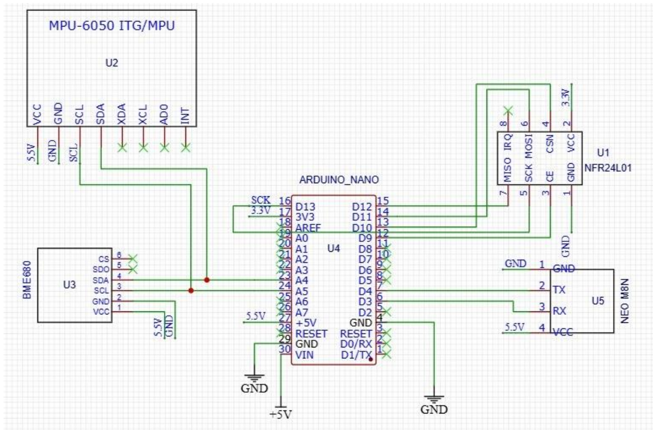

Figure 6 Circuit Wiring Diagram of CanSat

The web dash board developed using Next.js played a critical role in visualization. By separating pages for real-time monitoring, historical data fetching, and forecasting, the interface catered to multiple use cases. The use of charts, logs, and parameter cards ensured intuitive understanding of the dataset even for non-technical users. The modularity of the system

International Research Journal on Advanced Engineering and Management
https://goldncloudpublications.com
https://doi.org/10.47392/IRJAEM.2025.0360
e ISSN: 2584-2854
Volume: 03
Issue: 06 June 2025
Page No: 2283 - 2287

is another notable outcome. Additional sensors or CanSat units can be incorporated with minimal changes in both hardware and software. The MQTT-based messaging backbone and REST APIs used in the backend provide a scalable, cloud-agnostic architecture for expanding the use case to agricultural monitoring, weather forecasting, or educational payloads. Nevertheless, the system is not without limitations. The NRF24L01's range and susceptibility to interference restrict its use to short-range missions. Also, power optimization strategies for longer flight durations need further exploration. Lastly, field testing with actual deployment conditions (e.g., using weather balloons or drones) will be crucial to validate the system's resilience and data fidelity under real-world stress. Overall, the project successfully merges embedded systems, data science and web technologies to realize a cost-effective and modular CanSat prototype that can serve both research and educational purposes.

Acknowledgment

The authors gratefully acknowledge the facilities and mentorship provided by Malnad College of Engineering, Hassan.

References

[1]. A. Ferreira, M. Sousa, and L. Silva, "CanSat: A Hands-On Approach to Aerospace Education," in IEEE Global Engineering Education Confer- ence, 2019.
[2]. M. Ahmad, A. Rehman, and T. Khan, "Development and Testing of CanSat for Sensor Calibration and Data Analysis," in International Conference on Aerospace Science and Engineering, 2018.
[3]. R. Kumar and S. Singh, "Telemetry Systems in CanSat Missions Using RF and Wi-Fi Modules," in IEEE Sensors Applications Symposium, 2020.
[4]. J. Lopez and P. Gupta, "Design and Simulation of CanSat Recovery Systems," in AIAA Student Conference Proceedings, 2021.
[5]. S. Banerjee and V. Reddy, "Environmental Forecasting Using Facebook Prophet in Space Applications," in IEEE International Conference on Data Science, 2021.