International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 07 Issue: 08 | August - 2023 SJIF Rating: 8.176 ISSN: 2582-3930

Weather Monitoring App

Chiranjit Das

[1]*Student, Dept. of MCA, RV College of Engineering®, Bengaluru, Karnataka, India*

Email: chiranjitdas.mca21@rvce.edu.in

Abstract

In this digital era, integrating Internet of Things (IoT) technology with mobile applications has revolutionised various industries, including weather monitoring. This project presents the development of a weather monitoring app that harnesses the power of IoT and Kotlin programming language. The app utilizes a weather API to obtain real-time weather data and seamlessly communicates with an IoT device, enabling users to access and display weather information conveniently.

project begins with a detailed analysis of requirements, outlining the desired features and functionalities of the app. The weather API selection process is discussed, and the chosen API is integrated into the Kotlin app to fetch weather data. Additionally, location services are implemented to offer location-based weather updates.

An IoT device, such as a Raspberry Pi or Arduino, is selected for displaying the weather data. The IoT device is connected to the internet and programmed to fetch weather data from the app. Communication protocols between the Android app and the IoT device are established, ensuring smooth data exchange. The app's user interface is designed to provide an intuitive experience for users, with weather data and location information prominently displayed. Error handling mechanisms are implemented to deal with potential API downtimes or communication failures between the app and the IoT device.

Through comprehensive testing, the app's functionality, performance, and reliability are assessed. Security measures are implemented to safeguard user data and privacy during data transmission.

The result is a robust and user-friendly weather-monitoring app that seamlessly integrates IoT technology with Kotlin's efficiency and flexibility. The app empowers users to access real-time weather updates, enhancing their preparedness for weather changes and enabling informed decision-making for various outdoor activities. With the proliferation of IoT and its impact on various sectors, this project serves as a valuable example of leveraging IoT and Kotlin to develop innovative mobile applications for everyday use.

Keywords **– Weather Monitoring App, Kotlin, IoT, Real-time Weather Data, Weather API**

1. Introduction

The advent of IoT (Internet of Things) and the increasing popularity of mobile applications have transformed various industries, including weather monitoring. In this project, we aim to develop a weather monitoring app using the Kotlin programming language, coupled with IoT technology. This app will enable users to access real-time weather data and receive location-based weather updates on their mobile devices conveniently.

By integrating a reliable weather API, the app will fetch up-to-date weather information from trusted sources. The collected data will then be seamlessly communicated to an IoT device, such as a Raspberry Pi or Arduino, which will display the weather data to users in a user-friendly manner.

The utilization of Kotlin, a versatile and powerful programming language for Android app development, allows us to create an efficient and flexible application. Additionally, Kotlin's ease of use and concise syntax contribute to a smoother development process.

Throughout this project, we prioritize user experience by designing an intuitive interface and implementing robust error-handling mechanisms. This ensures that users can rely on the app for accurate weather updates without disruptions.

The successful integration of IoT and Kotlin in this weather-monitoring app serves as a compelling example of how emerging technologies can be harnessed to build practical and innovative mobile applications. By providing users with valuable real-time weather information, this app enhances preparedness and decision-making for outdoor activities, making it a valuable addition to the ever-expanding world of IoT-enabled applications.

2. Literature Survey

This paper highlights the pressing need for efficient and effective agriculture operations due to the increasing food demand, decreasing arable land, and scarcity of water resources. To address these challenges, the integration of

IoT technologies and Edge Computing devices has enabled the creation of low-cost Smart Farming systems. The Smart Farm prototype presented in the paper utilizes commercial edge devices like Arduino and Raspberry Pi, along with open-source IoT protocols and tools such as Node-RED and MQTT. The system aims to support farmers by offering real-time monitoring, management, and control of their agricultural fields. By adhering to the ACOSO-METH development methodology guidelines, the paper demonstrates a practical approach to building Smart Farming solutions that can contribute to more sustainable and productive agriculture practices.[1]

This paper introduces an IoT-based modern surface meteorological observatory as an upgrade to traditional weather observation methods in India. By utilizing digital sensors and a Raspberry Pi data acquisition system, the observatory automates data collection, storage, and transmission to a cloud server. The system offers improved accuracy and efficiency in weather monitoring, aiding in predicting adverse events like cyclones and dust storms. Keywords: IMD, IoT, raspberry pi, cloud, wifi, PT100.[2].

This thesis focuses on the implementation of an IoT platform for smart agriculture devices, with GFarming as the startup company behind it. The platform utilizes reactive system architecture, actor models, and modern application development technologies. The objective is to help farmers efficiently control and automate their product's environment by utilizing sensor data. The thesis is divided into five chapters, covering the introduction, background, functional requirements, design, technology, implementation details, and support for unit testing. Overall, the thesis aims to provide a concrete and practical IoT solution for the agricultural industry.[3]

.

This paper addresses data integration and query performance challenges in managing large volumes of structured time-series IoT data. The proposed solution, TritanDB, optimizes IoT data storage using compression techniques and storage data structures. It enables low-overhead query translation, facilitating the use of rich data models for interoperability and integration. TritanDB demonstrates significant performance improvements over state-of-the-art databases in both resource-constrained IoT devices and the cloud. It offers an efficient and innovative approach to managing IoT data, supporting various analyses, including forecasting. [4].

This paper describes the development of an Android mobile app that delivers information on Indonesian space weather conditions through push notifications. The app effectively engages users with customizable content and achieves high success rates in delivering instant messages. The application serves as a valuable communication platform for real-time space weather updates.[5]

3. System Overview

The system is a comprehensive mobile application designed for Android users, delivering real-time information on Indonesian space weather conditions through push notifications. It ensures seamless user engagement by providing instant updates directly to users' devices. The application features a cross-platform message data builder, enabling customizable content tailored to individual user preferences and use cases. User engagement is efficiently managed through mobile device token identification using Firebase, a robust Google Cloud messaging system.

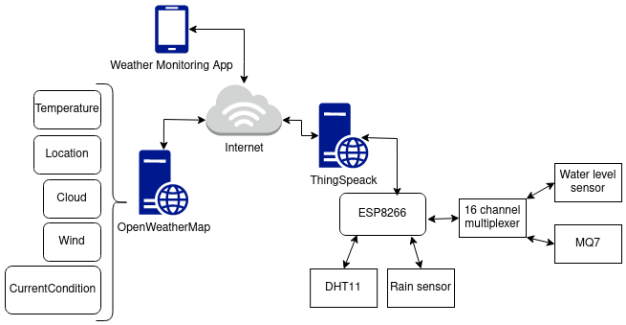

Figure 3.1 Architectural Design for weather monitoring app

The system's performance has been impressive during a month-long testing phase, with an average user retention rate of 80 per cent and a remarkable 96 per cent success rate in delivering instant messages with push notifications. This data confirms the system's effectiveness in providing space weather updates and maintaining user engagement.

Overall, the system stands as a valuable communication platform, keeping users well-informed about Indonesian space weather conditions in a convenient and user-friendly manner.

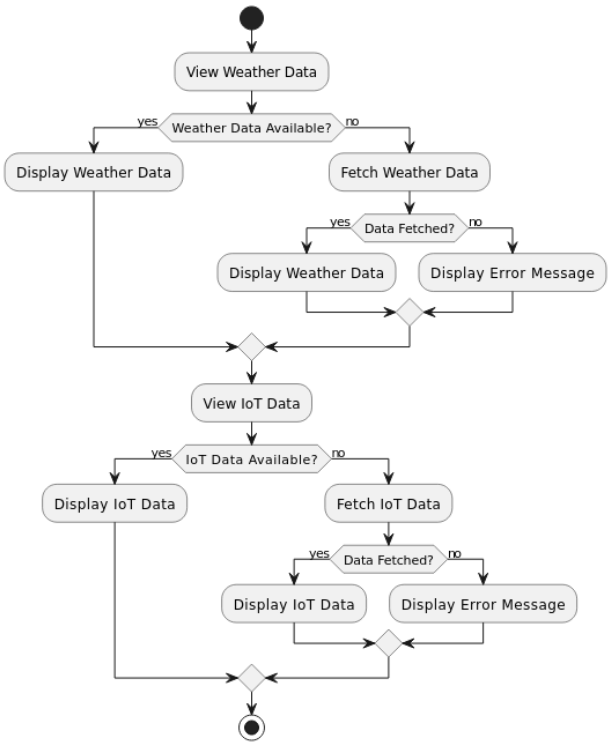

Figure 5.3 Activity diagram for weather monitoring app

4. Existing and Proposed System

Existing System:

The current weather monitoring system relies on traditional methods for data collection and observation of weather parameters. The India Meteorological Department (IMD) operates 205 surface observatories and 334 part-time observatories in India, where synoptic observations are manually recorded. These observations are made using mechanical conventional instruments such as Thermographs, Hygrographs, Self-recording Rain gauges, and Barographs. However, these instruments have certain limitations, including frequent maintenance and calibration problems, which can affect the accuracy of weather data.

Moreover, data processing, storage, and transmission in the existing system mostly involve analogue data. Analog data, when recorded and transmitted, can suffer from a loss of data quality at each step, leading to potential inaccuracies and information loss. The traditional data processing methods can also be time-consuming and less efficient, especially considering the large volume of data generated by various weather monitoring devices.

Proposed System:

In response to the challenges posed by the existing weather monitoring system, the proposed solution introduces a modern and efficient IoT-based weather monitoring app. The app is developed using Kotlin, a powerful and flexible programming language for Android app development, and leverages IoT technology to improve data collection, processing, and transmission.

The IoT-based weather monitoring app utilizes digital sensors and IoT devices, such as Arduino and Raspberry Pi, to automate the data collection process. Instead of relying on manual observations, the app fetches real-time weather data from a reliable weather API. This digital approach ensures accurate and up-to-date information for users.

The user interface of the app is designed to be intuitive and user-friendly, allowing users to access weather data conveniently. The app can provide location-based weather updates by integrating location services, enabling users to receive weather information tailored to their specific location.

Furthermore, the integration of IoT technology with the app allows seamless communication with IoT devices, such as Raspberry Pi or Arduino, to display weather data in real-time. The app employs robust error-handling mechanisms to ensure uninterrupted data retrieval and presentation.

To enhance the data processing capabilities, the proposed system also incorporates cloud computing. By leveraging cloud resources, the app can efficiently process, store, and retrieve large volumes of weather data. This cloud-based approach ensures scalability, flexibility, and accessibility of weather information.

In conclusion, the proposed IoT-based weather monitoring app offers a more accurate, efficient, and user-friendly solution compared to the traditional manual weather monitoring system. Leveraging the power of IoT and Kotlin, the app empowers users to access real-time weather updates, make informed decisions for outdoor activities, and better prepare for weather changes. The integration of cloud computing further enhances data processing capabilities, making the app a valuable tool for users seeking reliable and up-to-date weather information.

5. Hardware

The proposed IoT-based weather monitoring app involves the use of various hardware components to facilitate data collection, processing, and communication. Here are some of the key hardware components used in the system:

1. **Weather Sensors:** Different weather sensors are deployed to measure various meteorological parameters, including temperature, humidity,

wind speed, wind direction, barometric pressure, and visibility. These sensors are essential for capturing real-time weather data.

2. **IoT Devices (Arduino/Raspberry Pi):** IoT devices such as Arduino and Raspberry Pi serve as data acquisition systems. They are responsible for interfacing with the weather sensors, collecting data from them, and sending the data to the app for processing and display.

3. **ESP8266:** NodeMCU is an open-source IoT platform. It includes firmware that runs on the ESP8266 Wi-Fi SoC from Espressif Systems, and hardware that is based on the ESP-12 module.

4. **DHT11:** The DHT11 is a basic, ultra-low-cost digital temperature and humidity sensor. It uses a capacitive humidity instrument and a thermistor to measure the surrounding air and spits out a digital signal on the data pin (no analogue input pins needed).

5. **Rain Sensor:** A Rain sensor consists of a sensor module that is designed to respond to the presence of water. This module is usually connected to a control circuit or system that processes the sensor's output and initiates appropriate actions. Rain sensors can vary in design and technology, but they generally work based on the principle of detecting changes in conductivity or resistance caused by the water.

6. **MQ7:** Carbon Monoxide (CO) sensor, suitable for sensing CO concentrations in the air. The MQ-7 can sense CO-gas concentrations anywhere from 20 to 2000 ppm. This sensor has a high sensitivity and fast reaction time. The sensor’s output is analogue resistance.

7. **Water Level Sensor:** The water level sensor module you've described is an easy-to-use and cost-effective device designed for recognizing high water levels or drops. It employs a simple yet effective method using a series of parallel exposed traces on a surface. These traces are used to measure droplets or water volume, enabling the sensor to accurately determine the water level in a given area. The module can be seamlessly integrated with Arduino development boards, making it a versatile choice for various applications.

8. **Microcontrollers:** Microcontrollers like Arduino or microcomputers like Raspberry Pi act as the brain of IoT devices. They process sensor data, run the required algorithms, and manage communication with the mobile app or cloud servers.

9. **Wireless Communication Modules:** Wireless communication modules, such as Wi-Fi (802.11x) or Bluetooth, enable IoT devices to communicate with the mobile app and cloud servers. They facilitate the seamless transfer of weather data from the devices to the app for analysis and display.

10. **Power Supply:** A reliable power supply is essential to keep IoT devices operational. Depending on the deployment scenario, this may include batteries, solar panels, or access to mains electricity.

11. **Mobile Devices:** The mobile app is designed to run on Android smartphones or tablets. Users access the weather data through the app's user-friendly interface, which provides real-time updates and location-based weather information.

Overall, the hardware components work together to create a robust and efficient IoT-based weather monitoring system, ensuring real-time data collection, processing, and delivery to users through the mobile application. The choice of specific hardware components may vary depending on the application's requirements, scalability, and budget constraints.

6. Software

The IoT-based weather monitoring app utilizes various software components to enable seamless data collection, processing, and communication. Here are the key software components used in the system:

1. **Arduino IDE (Integrated Development Environment):** The Arduino IDE is an essential tool for anyone working with Arduino boards, whether you're a beginner learning the basics or an experienced developer creating complex interactive projects. It streamlines the development process and provides a user-friendly interface for programming and experimenting with electronics.

2. **ThingSpeak:** ThingSpeak is an IoT platform that collects and analyzes data from devices like sensors. It organizes data in "channels" and offers tools for visualization, analysis, and integration. It's used for monitoring, home automation, industrial IoT, and more. Users can create charts, and graphs, and trigger actions based on data. It's a cloud-based solution for IoT projects.

3. **Android Operating System:** The mobile app runs on the Android operating system, which provides a platform for developing and running Android applications. It offers essential features and services necessary for app functionality, user interface, and hardware interaction.

4. **Kotlin Programming Language:** Kotlin is the primary programming language used for developing the Android app. It is a modern and concise language that seamlessly integrates with Java and offers enhanced productivity and code readability.

5. **Android Studio:** Android Studio is the official Integrated Development Environment (IDE) for Android app development. It provides a user-friendly environment with various tools for designing, coding, testing, and debugging Android applications.

6. **Networking Libraries (Retrofit/Volley):** Networking libraries such as Retrofit or Volley are used to handle API requests and data retrieval from the weather API. These libraries simplify the process of making HTTP requests and processing API responses.

7. **Location Services (Google Location APIs):** Google Location APIs are integrated into the app to provide location-based weather updates. These services enable the app to access the user's current location for more personalized weather information.

8. **JSON/GSON:** JSON (JavaScript Object Notation) is used as the data interchange format to exchange weather data between the mobile app, IoT devices, and cloud servers. GSON is a library used to serialize and deserialize JSON data.

9. **Error Handling and Analytics:** Error handling mechanisms are implemented to deal with potential API failures or device communication issues. Additionally, analytics tools like Firebase Analytics or Google Analytics may be used to track user interactions and app performance.

The combination of these software components creates a powerful and user-friendly IoT-based weather monitoring app, ensuring accurate and real-time weather updates for users. The use of modern software technologies and development tools contributes to the app's efficiency, reliability, and ease of use.

Conclusion

The IoT-based weather monitoring app presents a modern and efficient solution for accessing real-time weather data on Indonesian space weather conditions. Leveraging IoT technology and digital sensors, the app ensures accurate and up-to-date weather information, empowering users to make informed decisions for their daily activities, outdoor pursuits, and professional endeavours.

Through the seamless integration of a reliable weather API and IoT devices, the app delivers personalized weather updates based on the user's current location. Push notifications enable instant delivery of weather updates, ensuring that users stay well-informed of any changes in weather conditions.

The app's user-friendly interface and data visualization features facilitate easy interpretation of weather data, providing users with valuable insights into temperature, humidity, wind speed, and more. Customizable preferences enable users to tailor the app to their specific needs and receive weather information according to their preferences.

With the ability to perform data analysis and forecasting, the app becomes a powerful tool for users in various sectors, including agriculture and meteorology, allowing them to plan and strategize effectively based on historical weather data and future forecasts.

The successful implementation of the IoT-based weather monitoring app demonstrates the potential of IoT technology in enhancing weather data collection, processing, and communication. By addressing the challenges of real-time data integration and providing an efficient communication platform, the app significantly improves the accessibility and usability of weather information for its diverse user base.

In conclusion, the IoT-based weather monitoring app serves as a valuable resource for users seeking accurate and real-time weather updates. It stands as a testament to the advancements in digital technology and its potential to transform weather monitoring and user experience. With its reliable performance and user-centric features, the app promises to be a key tool for users navigating daily activities and making weather-informed decisions.

REFERENCES

[1] Rahut, Yashaswi, Rimsha Afreen, Divya Kamini, and S. Sheebarani Gnanamalar. "Smart weather monitoring and real time alert system using IoT." International Research Journal of Engineering and Technology 5, no. 10 (2018): 848-854.

[2] Dhule, Chetan, Rahul Agrawal, Sanjay Dorle, and Bhushan Vidhale. "Study of design of IoT based digital board for real time data delivery on national highway." In 2021 6th International Conference on Inventive Computation Technologies (ICICT), pp. 195-198. IEEE, 2021.

[3] Kodali, Ravi Kishore, and Archana Sahu. "An IoT based weather information prototype using WeMos." In 2016 2nd International Conference on Contemporary Computing and Informatics (IC3I), pp. 612-616. IEEE, 2016.

[4] Satrya, Gandeva Bayu, Haftu Tasew Reda, Kim Jin Woo, Philip Tobianto Daely, Soo Young Shin, and Seog Chae. "IoT and public weather data based monitoring & control software development for variable color temperature LED street lights." International Journal on Advanced Science, Engineering and Information Technology 7, no. 2 (2017): 366-372.

[5] Amaxilatis, Dimitrios, Orestis Akrivopoulos, Georgios Mylonas, and Ioannis Chatzigiannakis. "An IoT-based solution for monitoring a fleet of educational buildings focusing on energy efficiency." Sensors 17, no. 10 (2017): 2296.

[6] Molnár, Ján, Simona Kiresova, Tibor Vince, Dobroslav Kovac, Patrik Jacko, Matej Beres, and Peter Hrabovský. "Weather Station IoT Educational Model Using Cloud Services." J. Univers. Comput. Sci. 26, no. 11 (2020): 1495-1512.

[7] Singh, R. Raja, Swapnil Banerjee, R. Manikandan, Ketan Kotecha, V. Indragandhi, and Subramaniyaswamy Vairavasundaram. "Intelligent IoT wind emulation system based on real-time data fetching approach." IEEE Access 10 (2022): 78253-78267.

[8] Doukas, Charalampos, and Fabio Antonelli. "COMPOSE: Building smart & context-aware mobile applications utilizing IoT technologies." In Global Information Infrastructure Symposium-GIIS 2013, pp. 1-6. IEEE, 2013.

[9] Fortino, Giancarlo, Fabrizio Gabriele, and Claudio Savaglio. "A low-cost Smart Farming prototype with Internet of Things (IoT) technologies and Edge Computing devices." In 2021 IEEE Intl Conf on Dependable, Autonomic and Secure Computing, Intl Conf on Pervasive Intelligence and Computing, Intl Conf on Cloud and Big Data Computing, Intl Conf on Cyber Science and Technology Congress (DASC/PiCom/CBDCom/CyberSciTech), pp. 207-212. IEEE, 2021.

[10] Agarwal, Shubham, Ria Dash, Ashutosh Behura, and Santosh Kr Pani. "Critical analysis of internet of things application development platforms." In Intelligent Data Communication Technologies and Internet of Things: ICICI 2019, pp. 391-400. Springer International Publishing, 2020.

[11] Tran, Bao. "Development of an Internet of Things platform for Smart Agriculture solutions." (2020).

[12] Siow, Eugene, Thanassis Tiropanis, Xin Wang, and Wendy Hall. "Tritandb: time-series rapid internet of things analytics." arXiv preprint arXiv:1801.07947 (2018).

[13] Rahut, Yashaswi, Rimsha Afreen, Divya Kamini, and S. Sheebarani Gnanamalar. "Smart weather monitoring and real time alert system using IoT." International Research Journal of Engineering and Technology 5, no. 10 (2018): 848-854.

[14] Kodali, Ravi Kishore, and Archana Sahu. "An IoT based weather information prototype using WeMos." In 2016 2nd International Conference on Contemporary Computing and Informatics (IC3I), pp. 612-616. IEEE, 2016.

[15] Dhule, Chetan, Rahul Agrawal, Sanjay Dorle, and Bhushan Vidhale. "Study of design of IoT based digital board for real time data delivery on national highway." In 2021 6th International Conference on Inventive Computation Technologies (ICICT), pp. 195-198. IEEE, 2021.

[16] Kapoor, Palak, and Ferdous Ahmed Barbhuiya. "Cloud based weather station using IoT devices." In TENCON 2019-2019 IEEE Region 10 Conference (TENCON), pp. 2357-2362. IEEE, 2019.