# Designing a Machine Learning-Calibrated IoT Sensor Network for Real-time Air Quality Assessment

**Adil Zhexenov,** B.Sc.

**Submitted in fulfilment of the requirements for the degree of Master of Science in Electrical and Computer Engineering**


**School of Engineering and Digital Sciences**
**Department of Electrical and Computer Engineering**
**Nazarbayev University**

53 Kabanbay Batyr Avenue,
Astana, Kazakhstan, 010000

**Supervisor:** Akhan Almagambetov

**Co-Supervisor:** Sultangali Arzykulov

**Date of submission:** March 2026

# Abstract

Existing air quality monitoring infrastructure in Kazakhstan provides limited spatial coverage, particularly in cities with extreme continental climates and coal-dominated PM2.5 emissions. This thesis presents the design, deployment, and evaluation of an IoT-based sensor network for real-time PM2.5 monitoring in Astana. Four ESP32-based sensor nodes with PMS5003 sensors were deployed across the city, collecting 14,444 measurements over 28 days (February–March 2026) with 89.95% data completeness at temperatures down to −26.8 °C. Co-location with the Kazhydromet-14 reference station enabled machine learning calibration, with Random Forest achieving the highest accuracy ($R^2$ = 0.84, RMSE = 3.80 μg/m³), satisfying the EPA performance criterion. Age-based calibration analysis revealed that linear model coefficients degrade by 76.8% within one week during seasonal transitions, while Random Forest maintains stable performance ($R^2$ = 0.93–0.99), leading to a weekly retraining recommendation. For 7-day PM2.5 forecasting, LSTM was identified as the best model ($R^2$ = 0.23, RMSE = 11.62 μg/m³).

**Keywords:** air quality, IoT sensor network, PM2.5 monitoring, machine learning calibration, Random Forest, low-cost sensors, PMS5003, LSTM forecasting.

# Acknowledgements

Firstly, I would like to express my uttermost gratitude towards my supervisors Akhan Almagambetov and Sultangali Arzykulov. It has been their supervision and direction throughout the duration of my studies which has allowed me to successfully complete this Master's program. I am appreciative for all the hours of discussion they have offered me, especially in the areas of IoT systems and machine learning.

Secondly, I would like to show my indebtedness to the administration staff at Nazarbayev University whose efforts a lot of the time go unnoticed, but are necessary to provide students with the resources and support they need to focus on their academic work.

Thirdly, I would like to thank my family and friends for their continued support during the writing of this thesis.

# Table of Contents

# List of Abbreviations

| | |
|---|---|
| ABS | Acrylonitrile Butadiene Styrene |
| AHT10 | Asair Humidity and Temperature Sensor |
| ANN | Artificial Neural Network |
| API | Application Programming Interface |
| AQI | Air Quality Index |
| AQICN | Air Quality Index China Network |
| BAM | Beta Attenuation Monitor |
| BiLSTM | Bidirectional Long Short-Term Memory |
| BMP280 | Bosch Environmental Pressure Sensor |
| CHP | Combined Heat and Power Plant |
| CI/CD | Continuous Integration / Continuous Deployment |
| CORS | Cross-Origin Resource Sharing |
| CSV | Comma-Separated Values |
| CV | Cross-Validation |
| DTR | Decision Tree Regression |
| EPA | Environmental Protection Agency (U.S.) |
| ESP32 | Espressif Systems 32-bit Microcontroller |
| GPIO | General Purpose Input/Output |
| GPS | Global Positioning System |
| HTTP | Hypertext Transfer Protocol |
| I²C | Inter-Integrated Circuit |
| IoT | Internet of Things |
| IP65 | Ingress Protection Rating 65 |
| JSON | JavaScript Object Notation |

| | |
|---|---|
| KFold | K-Fold Cross-Validation |
| KNN | K-Nearest Neighbors |
| LED | Light Emitting Diode |
| LiPo | Lithium Polymer |
| LSTM | Long Short-Term Memory |
| MAE | Mean Absolute Error |
| ML | Machine Learning |
| MLR | Multiple Linear Regression |
| NTP | Network Time Protocol |
| PCB | Printed Circuit Board |
| PM1 | Particulate Matter (≤1 μm) |
| PM2.5 | Fine Particulate Matter (≤2.5 μm) |
| PM10 | Particulate Matter (≤10 μm) |
| PMS5003 | Plantower Particulate Matter Sensor |
| $R^2$ | Coefficient of Determination |
| REST | Representational State Transfer |
| RF | Random Forest |
| RMSE | Root Mean Squared Error |
| RTC | Real-Time Clock |
| SVR | Support Vector Regression |
| UART | Universal Asynchronous Receiver-Transmitter |
| UI | User Interface |
| WHO | World Health Organization |
| Wi-Fi | Wireless Fidelity |
| XGBoost | Extreme Gradient Boosting |

# List of Figures

# List of Tables

# Chapter 1 – Introduction

## 1.1. Context and Background

Air pollution is a worldwide ecological problem, which negatively affects human health and the environment. Air pollution level can be determined by the key indicator, which is known as the concentration of fine particulate matter (PM2.5). The World Health Organization (WHO) recommends that the annual average PM2.5 value should be lower than 5 μg/m³ [1]. The annual PM2.5 concentrations in Kazakhstan is 22.2 μg/m³, which exceed the WHO's annual standards by 4.4 times [2]. The majority of residents remain unaware of risks due to limited public awareness, despite the current deplorable state of air pollution [3].

Traditional regulatory-class stations are used for the air pollution monitoring. Such traditional monitoring stations provide highly accurate measurements. However, they can face several limitations such as high cost, maintenance complexity and limiting covering area [4]. In contrast, the monitoring system based on the Internet of Things (IoT) can overcome these limitations. Such IoT-based monitoring systems can effectively collect air pollution measurements and transmit them for further analysis via wireless data transmission. These systems are easily scalable and cost-effective solutions, which can improve air pollution monitoring in urban areas.

## 1.2. Problem Statement

IoT-based monitoring systems face several limitations, such as sensor degradation over time, which reduces air pollution measurements accuracy [5]. Identical sensor nodes located close to each other collect different measurements [6], especially in regions with extreme climatic conditions like Kazakhstan. These factors often lead to premature failure of such IoT sensors and create data gaps. These challenges limit the reliability and long-term usability of IoT-based air quality monitoring systems.

The novelty of this study is a systematic age-based calibration analysis, determining the minimum retraining frequency in changing environmental conditions. This approach provides new empirical knowledge about the behavior of calibration parameters during the transition between seasons, and practical recommendations for retraining based on the real deployment. In contrast to the isolated solution of the above-mentioned parts, this study combines calibration, forecasting, automated health monitoring and real-time visualization into an end-to-end system.

## 1.3. Aim and objectives

**General Objective:** To design and evaluate a multi-node IoT-based monitoring network for air pollution assessment in Astana, which uses machine learning to calibrate raw data from designed sensor nodes and PM2.5 forecasting.

**Specific Objectives:**

1. To design and implement a prototype IoT-based monitoring device, which combines low-cost IoT sensors to collect PM2.5 and weather parameters.
2. To compare different machine learning techniques for PM2.5 calibration and determine the most efficient calibration model.
3. To investigate the impact of seasonal variations, especially the heating season, on the measurement accuracy from low-cost sensors.
4. To develop an automated cloud-based dashboard for real-time data visualization.

## 1.4. Research Questions

1. Which machine learning algorithms provide the highest calibration accuracy for low-cost IoT sensors for air pollution assessment?
2. What is the minimum frequency of updating ML models required to maintain high calibration accuracy?
3. How do seasonal variations, especially in the heating season, affect the accuracy and stability of low-cost IoT air quality sensors?

# Chapter 2 – Literature Review

This chapter provides an overview of the existing literature, which is related to the design and implementation of an ML-calibrated IoT-based sensor network for real-time air quality assessment. This chapter is divided into four sections. Section 2.1 examines the air pollution state in Kazakhstan and identifies critical gaps in the current air pollution monitoring infrastructure. Section 2.2 provides an assessment of IoT-based monitoring systems as a cost-effective alternative to traditional monitoring stations. Section 2.3 analyzes the sources of measurement errors and ways to eliminate these errors using different calibration methods in low-cost PM sensors. Section 2.4 considers different machine learning (ML) approaches for low-cost sensor calibration and PM2.5 forecasting, and identifies methodological issues, which motivate this study.

## 2.1. Challenges and Gaps in Air Quality Monitoring in Kazakhstan

Kazakhstan has one of the highest PM2.5 concentration values in Central Asia. This air pollution issue is becoming more urgent year after year [7]. In 2021, Kazakhstan was ranked 23rd among the most polluted countries in the world with a PM2.5 value of 31.1 μg/m³. This value exceeds the WHO limits of 5 μg/m³ by 6.2 times [8][9]. Also in 2021, the annual PM2.5 concentrations in cities of republican significance such as Astana and Almaty exceeded the WHO standards by 4.5 and 7.1 times, respectively [8]. Later measurements confirm that this situation persists. From August 2022 to July 2023, the annual PM2.5 concentrations in Almaty (35.6 μg/m³) and Astana (19.4 μg/m³) continued to exceed the WHO recommended levels [1][10]. In 2021, the daily PM2.5 concentrations exceeded the WHO 24-hour standards (15 μg/m³) [1] on 151 days in Astana and 217 days in Almaty [8]. Such exceedances of the WHO standards indicate that the population of both cities is exposed to dangerous air quality for a large part of the year.

The health consequences of such prolonged exposure are serious, and they remain largely unexplored in Kazakhstan [11]. Between 2015 and 2017, about 8 thousand deaths occurred annually in Kazakhstan due to diseases related to poor air quality [7]. In 2021, the number of deaths increased to more than 16 thousand [12]. Environmental exposure to PM2.5 has led to an

increase in mortality at the urban level. In 2022, the number of PM2.5-related deaths was approximately 2100 in Almaty and 660 in Astana [11]. These mortality rates exceeded deaths from other preventable causes such as traffic accidents and HIV/AIDS in both cities. Figure 2.1 illustrates the distribution of excess mortality by age group and major categories of diseases associated with PM2.5 exposure.

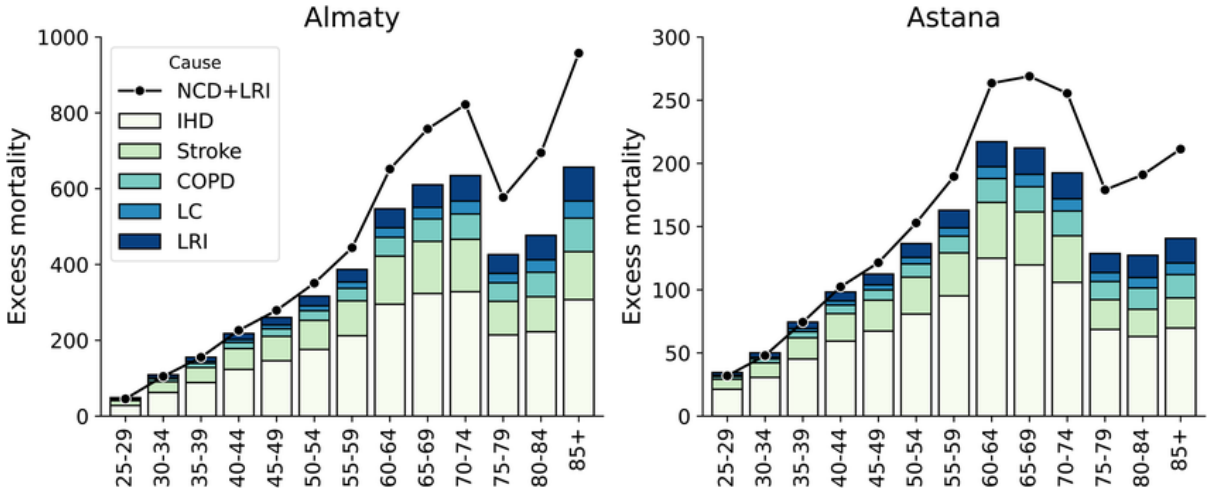


***Figure 2.1: The total premature deaths attributable to PM2.5 exposure during 2022–2024 (summed for 3 years) in Almaty (left) and Astana (right) [11].***

Kazakhstan is also facing large economic losses from PM2.5 exposure. At the national level, the impact of PM2.5 on human health in 2019 was estimated at $12 billion, which is equivalent to 6.7% of gross domestic product [8]. Economic losses at the city level from PM2.5-related mortality were estimated at $2.8–4.6 billion for Almaty and $0.9–1.5 billion for Astana per year [11]. This economic burden demonstrates that air pollution not only harms public health, but also slows down economic growth, reducing a high percentage of national economic production.

The high level of air pollution in Kazakhstan is mainly determined by the climatic and geographical features of the country, which create favorable conditions for PM2.5 accumulation. Long cold winters and frequent temperature inversions limit the vertical mixing of air and trap pollutants in the surface layer of the atmosphere [12]. Winter anticyclones, low boundary layer height and high relative humidity also increase the probability of fine particle accumulation [12][13].

These meteorological conditions become especially critical during the heating season (October-April), when coal burning at combined heat and power plants, autonomous boiler houses and residential heating systems leads to a substantial increase in PM2.5 emissions [14]. During this period, average PM2.5 concentrations at air pollution monitoring stations increase approximately twice [14], with winter averages reaching 35.3 μg/m³ in Astana and 76.0 μg/m³ in Almaty [8]. The chemical characterization of PM2.5 confirmed these seasonal patterns. In winter, both of these cities showed high peaks in the content of ammonium, nitrates, sulfates, and carbonaceous substances. These elevated concentrations indicate an increase in combustion emissions and limited atmospheric dispersion during the cold months, which are typical for the Kazakhstan climate [10]. Seasonal variations in PM2.5 levels in Astana and Almaty are illustrated in Figure 2.2, which shows higher PM2.5 concentrations in the heating season, especially in Almaty.

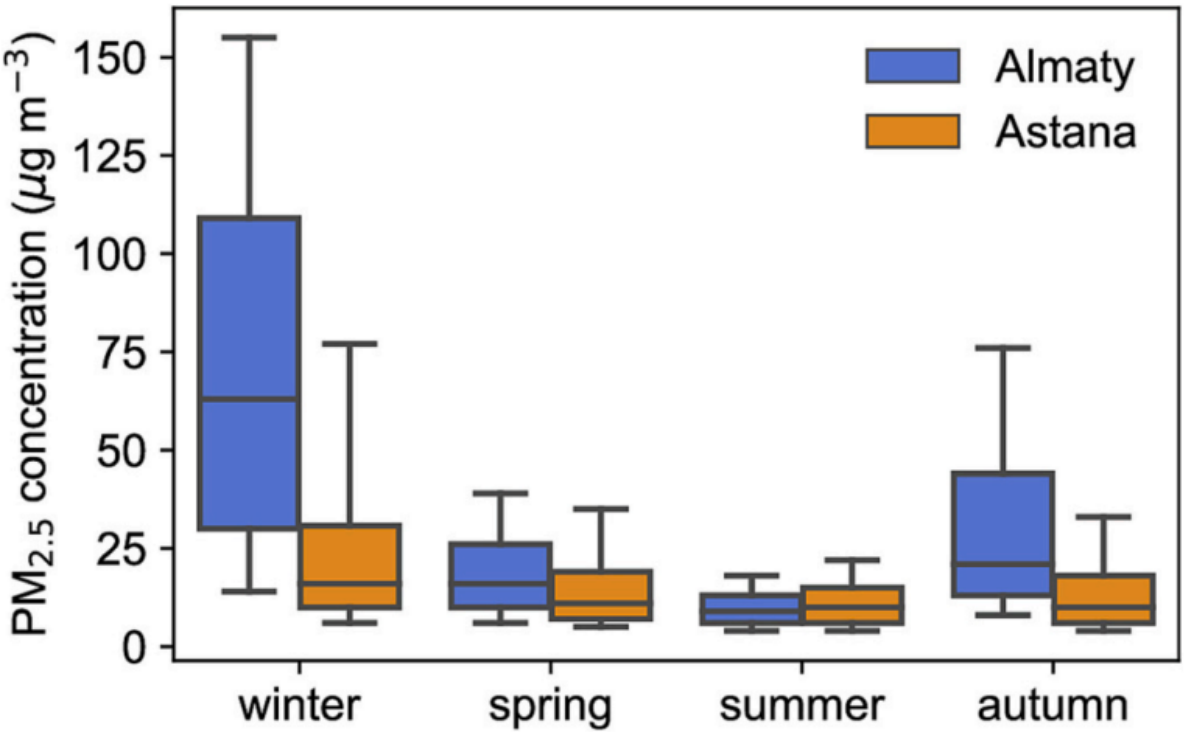


***Figure 2.2: Seasonal average PM2.5 concentrations for Almaty and Astana in 2021 [8].***

In Kazakhstan, air pollution sources are transportation, industrial emissions, and coal, which provides 69% of energy [7]. Coal-fired combined heat and power plants (CHPs) consume

3.7 and 2.2 million tons of coal annually in Almaty and Astana, respectively [10]. Such CHPs in Kazakhstan operate without $SO_2$ and $NO_2$ emission-control technologies [13]. Despite the high air pollution level caused by coal burning, the transition to cleaner fuels such as natural gas is still slow and is mainly limited to large cities [7]. In Astana, 100% of households used coal or wood for heating, and 26% of households in Almaty used solid fuels for heating [8]. A comprehensive survey of 11,944 households in Astana demonstrated that households annually consumed 39 thousand tons of coal and 13 thousand $m^3$ of biomass to heat residential premises [10]. These heating processes directly increase PM2.5 levels, causing air pollution peaks at monitoring stations in the heating season. Five different sources were identified in Almaty: resuspended urban road dust (32%), urban atmosphere (20%), power plants (18%), residential heating (16%) and exhaust emissions (14%). In Astana, residential heating, regional and local power plants, and traffic emissions accounted for 20% each. In addition, resuspended urban road dust and industrial emissions accounted for 22% and 18%, respectively [10]. Despite the fact that pollution sources were distributed relatively evenly across both cities, numerous pollution factors showed a strong correlation with coal and biomass combustion markers, with the total coal-related combustion sources reaching approximately 40% of the total amount of PM2.5 in both cities [10]. In addition, the growing number of outdated passenger cars and the lack of exhaust emission control systems contribute to additional air pollution [8].

Despite these negative statistics, the air pollution monitoring system in Kazakhstan remains substandard, fragmented, and often insufficient for collecting spatial and temporal variability of PM2.5 concentrations. The National Air Quality Monitoring Network, managed by Kazhydromet, provides only minimal coverage. Most cities in Kazakhstan are absent from global air quality rankings due to insufficient monitoring. Although Almaty and Astana have some level of monitoring infrastructure, these resources are unevenly distributed. Astana has only 4 monitoring stations, which is much less than in Almaty [12], and even those stations are poorly located near combined heat and power plants, as shown in Figure 2.3 [8].

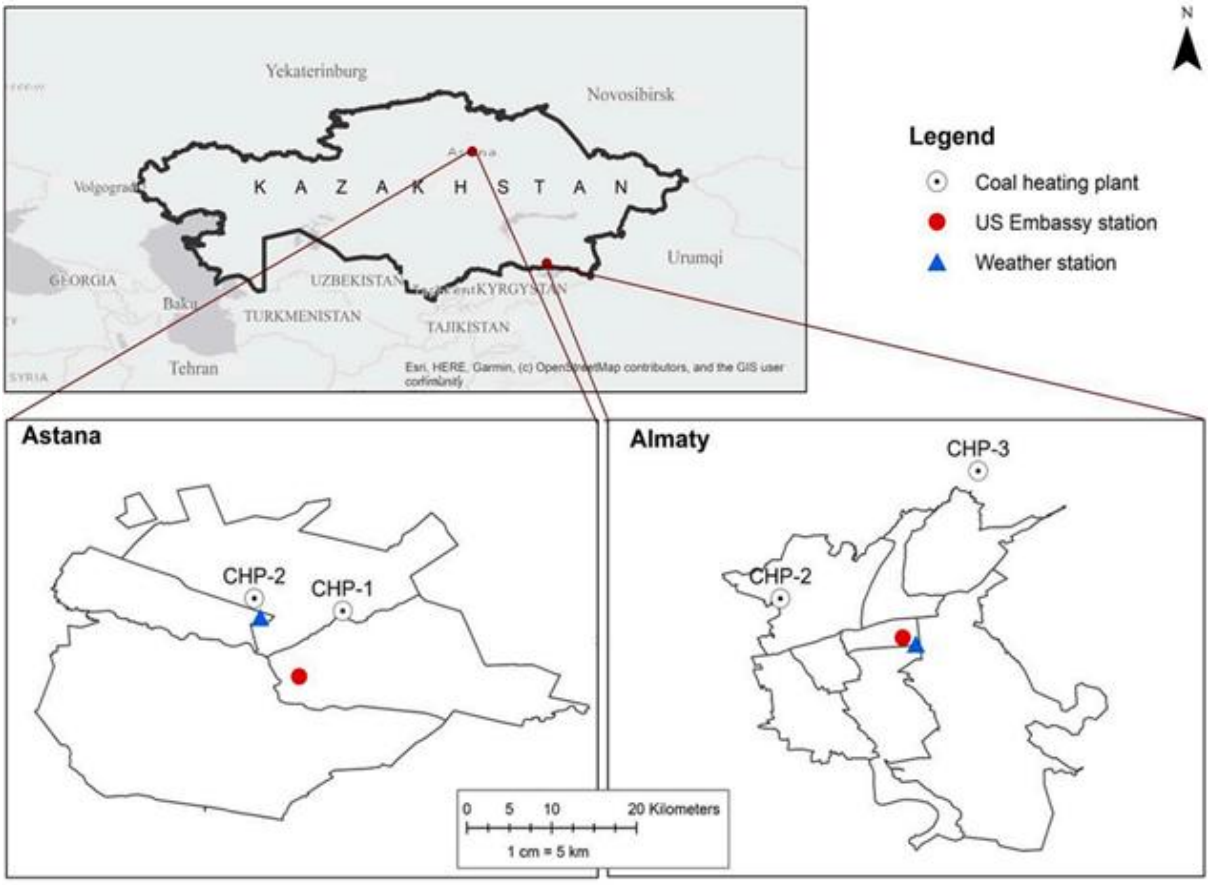


***Figure 2.3: Locations of air quality monitoring stations and coal-fired power plants in Almaty and Astana [8].***

Air quality data collection can have large gaps even at traditional regulatory stations such as the Beta Attenuation Monitoring station (BAM-1020). According to [7], even a globally reliable station such as the BAM-1020 can face problems with data completeness. This experiment showed an observed data gap of 8.8%. Actual industrial emissions cannot be independently checked because Kazakhstan lacks continuous emission monitoring systems (CEMS) at coal-fired power plants and metallurgical plants [12]. Such lack of CEMS leads to incomplete emission datasets, limiting their applicability in assessing industrial contributions and developing accurate forecasting systems. The lack of a comprehensive monitoring system further complicates the development of methods for analyzing the air pollution formation mechanisms, forecasting capabilities, early warning systems, and evidence-based assessment of measures [10]. In highly developed industrial cities, where air pollution levels are among the highest in the country, air

pollution monitoring networks remain extremely sparse. Despite the existence of CHPs, metallurgical complexes and winter smog, in the highly developed industrial cities such as Temirtau, Ust-Kamenogorsk, Balkhash and Ekibastuz, air pollution monitoring networks have only 1-3 stations [12].

## 2.2. IoT Solutions for Real-Time Air Quality Monitoring

Currently, air quality monitoring uses traditional monitoring stations based on reference devices such as stationary environmental stations like BAM-1020 [5][15]. Such traditional air quality monitoring stations provide high data accuracy and data completeness [16]. However, these stations are costly to install and maintain, limiting their spread over a large area [17]. These limitations make it economically impossible to create dense air quality monitoring networks [5]. Such sparse placement of monitoring stations leads to large geographical gaps, forcing authorities to extrapolate air pollution measurements over large areas, which reduces measurement accuracy and makes pollution trends unclear in areas far from stationary environmental stations [16]. In addition, these limitations demonstrate the low efficiency of traditional stations in harsh weather conditions and in responding to sensor failures, which is necessary for modern spatiotemporal air pollution analyses [18].

As a low-cost alternative, air quality monitoring can use systems based on the Internet of Things (IoT) sensors to measure gaseous and particulate concentrations in real-time [15][16][18]. Unlike traditional monitoring stations, which collect less detailed datasets due to sparse placement and high operating costs [17], IoT-based air pollution monitoring systems use low-cost sensor nodes that can be densely located in urban areas to achieve detailed spatiotemporal monitoring [19]. Dense monitoring networks based on low-cost sensors can assess the risk of PM2.5 exposure with higher spatial and temporal resolution than in regulatory monitoring systems. Such regulatory monitoring systems include the Federal Reference Method or the Federal Equivalence Method (FRM/FEM), which are usually located in sparse locations due to their high cost and complexity

of the system [20]. One example of such networks, the PurpleAir platform has increased from over 10,000 sensors in 2021 [21] to over 30,000 sensors by April 2022 [20] proving the easy scalability of such monitoring networks. In the Washington State metropolitan area alone, more than 700 PurpleAir sensors worked side by side with only about 15 regulatory stations. Such density demonstrates the advantage of low-cost sensor networks in spatial coverage [22].

IoT-based air quality monitoring nodes consist of a microcontroller like ESP32 and low-cost PM and environmental data sensors [23]. A typical IoT-based air quality monitoring system has a multi-level architecture consisting of several key components: outdoor-located sensor nodes, a communication network between nodes, the server-side application with cloud integration, and a user-oriented client application [24]. Such systems transmit collected data to storage via wireless sensor networks (WSNs) using Wi-Fi, LoRaWAN, or ZigBee communication protocols, providing an IoT sensor network and covering a large area [4][25][26][27]. Among these protocols, LoRaWAN has specific advantages for outdoor air quality monitoring due to its autonomous network architecture, low power consumption, wide range of connectivity, and low operating costs compared to Wi-Fi [24]. For multi-node IoT architectures, Message Queuing Telemetry Transport (MQTT) Protocol provides efficient communication between sensor nodes and edge or cloud devices [28][29]. There are alternative approaches to achieve full autonomy in remote locations. For example, hybrid power systems based on solar cells and batteries combined with 4G IoT SIM cards, which eliminate dependence on external electricity and fixed network infrastructure [30].

However, for the city-level deployment, where residential Wi-Fi infrastructure is easily available, a direct Wi-Fi connection remains the most practical choice. A direct Wi-Fi connection does not require additional gateway hardware and provides sufficient bandwidth for periodic data transmission with real-time Network Time Protocol (NTP) clock synchronization, which is essential for accurate definitions of time-series parameters. The above-mentioned low-cost PurpleAir monitoring system also uses Wi-Fi, which confirms this approach [21]. However, offline data recorded without Wi-Fi access may contain timestamp errors [21]. This data transfer approach

provides continuous data collection, data processing, and data transfer of air pollution measurements to the cloud storage without human intervention [31] as shown in Figure 2.4.

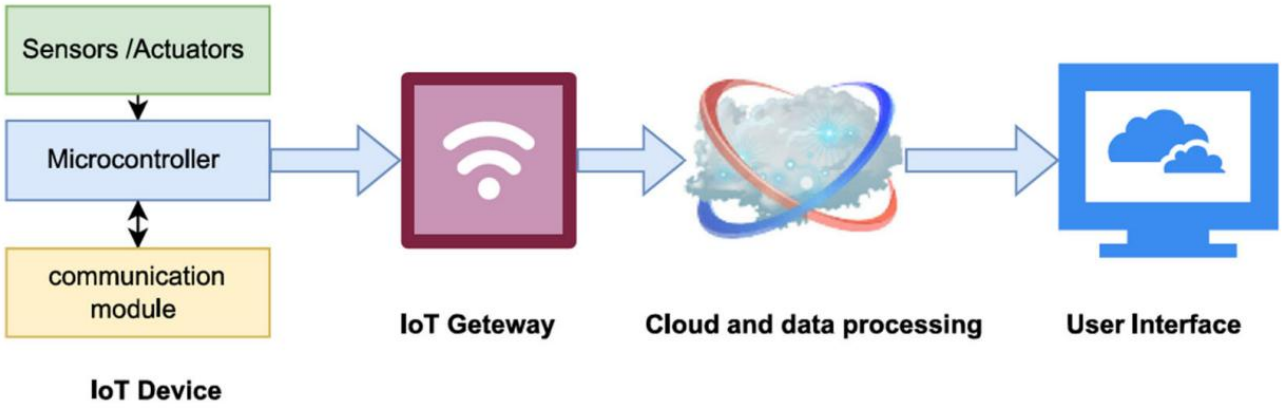


***Figure 2.4: Internet of Things system modules for real-time air quality monitoring [19].***

Low-cost particulate matter (PM) sensors, which are the major component in IoT-based air pollution monitoring nodes, are compact optical or nephelometric devices. These sensors estimate fine particle concentrations by measuring the intensity of light scattered by aerosols [32]. PM sensors use different approaches and architectures, depending on their cost and measurement accuracy. Plantower devices like the PMS5003 and PMS7003 or Honeywell HPMA115S0 are based on laser scattering cameras with a fan and fixed airflow [33]. Advanced sensors such as the Alphasense OPC-N2 are based on larger optical cameras and particle-counting capabilities. Such sensors provide higher measurement accuracy, but they are much more expensive than their Plantower analogues [32]. Therefore, the gold standard for IoT-based air pollution monitoring systems is the Plantower PMS5003, which is several times cheaper than the Alphasense OPC-N2 and provides higher measurement accuracy. Plantower devices are not efficient at measuring larger PM particles because these particles must make two or three 90-degree turns before passing the laser and photodetector assembly, reducing the sampling efficiency for coarse particles [34]. However, in the monitoring systems used in [7][27][32][33][35], the PMS5003 was a key component. These studies also demonstrate the successful long-term implementation of portable and stationary low-cost IoT-based devices integrated into large-scale network models such as MegaSense [36], which enable real-time air pollution monitoring using different sensor nodes.

Table 2.1 adapted from [32] shows the main characteristics of the above-mentioned PM sensors such as their physical size, cost, particle detection range, claimed accuracy, and so on.

**Table 2.1: Main characteristics of the low-cost PM sensors. Adapted from [32].**

| **Model** | **Size (mm) (H × W × D)** | **Price (USD)** | **Detection range (µm)** | **Concentration range (µg/m³)** | **Declared Accuracy (µg/m³)** |
|---|---|---|---|---|---|
| Alphasense OPC-N2 | 60 × 64 × 75 | 443 | 0.38 to 17 | 0.01 to 1,500 | Not known |
| Plantower PMS5003 | 38 × 21 × 50 | 28 | 0.3 to 10 | 0 to 500 | ±10 |
| Plantower PMS7003 | 37 × 12 × 48 | 28 | 0.3 to 10 | 0 to 500 | ±10 |
| Honeywell HPMA115S0 | 36 × 43 × 24 | 33 | Not known | 0 to 1,000 | ±15 |

Despite the advantages of an IoT-based air pollution monitoring system such as easy scalability and low-cost, their practical implementation faces technical limitations, which affect data reliability and long-term performance [5][16]. Low-cost IoT sensors have lower measurement accuracy compared to reference stations [16]. Measuring signals of low-cost IoT sensors are very sensitive to different environmental factors such as humidity, temperature, wind speed, and pressure, which affect the accuracy of the fine particle concentrations and gaseous pollutants [26]. In a study conducted in Almaty using a PMS5003 sensor to measure PM2.5 levels, data gaps were identified. These sensors were installed in two places, at the Kazakh National University and on Seifullin street, located close to each other. The results showed data gaps of 27% and 32%, respectively. In contrast, a traditional monitoring station, a BAM-1020, showed a data gap of 8.8% during the same period [7].

IoT-based monitoring systems also face infrastructural limitations. Many existing IoT architectures still depend on power or fixed Wi-Fi networks, which limits their deployment in low-maintenance or remote urban areas [23]. Data completeness also remains one of the challenges of

such systems, as IoT nodes often lose data due to communication failures, hardware failures, or the impact of environmental conditions [18]. Depending on the Wi-Fi connection and the power source stability, when IoT-based monitoring systems are really implemented, their data completeness can range from 54.2% to 99.5% [15]. High humidity and extreme temperatures, which are typical for Kazakhstan [12], degrade long-term data collection [18]. One of the key limitations of low-cost IoT air pollution sensors is long-term sensor drift, which reduces measurement accuracy over time. This makes uncalibrated data unreliable for scientific or regulatory purposes [5][35]. Such drifts can occur as early as several weeks or months after deployment, causing shifts in sensor response and reducing measurement accuracy [6].

## 2.3. Improving Accuracy of Low-Cost Air Quality Sensors

Using IoT sensors without any calibration can lead to inaccurate measurements, because their raw measurements often deviate from the reference stations [5]. These inaccuracies are caused by cross-sensitivities between different airborne pollutants, external factors, such as traffic, climatic conditions, and human activity, and sensor drift [14]. The regular overestimation of PM2.5 concentrations by about 40% has been reported in different independent studies, which were conducted in the United States, Europe and India [15][20][21][38]. The reason for this overestimation is the factory calibration of Plantower sensors, which is performed using atmospheric aerosol in some Chinese cities. Therefore, such factory calibration does not reflect the aerosol characteristics in the places where the sensor is actually deployed [21][39]. While PM2.5 concentrations are usually overestimated, PM10 concentrations are often underestimated due to the inability of optical sensors to detect large particles accurately [30].

In addition, low-cost IoT sensors demonstrate inconsistencies between the same IoT nodes, collecting different measurements even in the case of identical models, placement and weather conditions. This inconsistency further increases the importance of calibration for data analyses [6]. This variability between sensor nodes has been quantified in long-term field studies. Plantower

devices such as the PMS1003, overestimated the PM2.5 measurements by 1.89 times, and two PMS5003 sensors showed deviations of 1.47 and 1.08 times relative to the reference measurements, respectively [34]. These inconsistencies occur due to inherent manufacturing tolerances, as the electronic components have different performance parameters. During manual or automated assembly, changes are made that also affect the final device consistency [30][40]. Harsh weather conditions accelerate sensor degradation, thereby limiting long-term and high-quality air pollution monitoring [14]. This underlines the need for calibration for low-cost IoT sensors in Kazakhstan. Therefore, the accuracy of low-cost IoT sensors without any calibration remains insufficient for scientific or regulatory purposes, and their collected incorrect measurements can lead to incorrect findings during data analysis [26].

Relative humidity (RH) is the main source of measurement error in low-cost optical PM sensors due to a physical mechanism, known as hygroscopic particle growth [20][40]. In conditions of high humidity, aerosol particles absorb water vapor and increase in size, which enhances the light scattering signal and leads to an overestimation of the mass PM concentrations by the sensor [20][39]. Such visible distortions are detected already at RH of about 50%. The effect becomes critical at values above 80-85% [40]. Overall, RH can account for up to 30% of the variance of PM2.5 measurements, which are obtained by using low-cost optical sensors [40]. Internal heating of the electronic components inside the sensor can partially reduce the effect of RH on the accuracy of PM measurements, as proved by PurpleAir sensors, which show RH values 10-20% lower than reference RH. However, this approach does not eliminate completely the humidity-related bias [20][21][39]. In addition, the PMS5003 sensors demonstrate a strong seasonal dependence. During two subsequent winters, PM2.5 concentrations correlated well with reference devices, showing $R^2 > 0.87$, but in spring and during the forest fire season, the $R^2$ values decreased to 0.18–0.72 [34]. This pattern is also confirmed by experiments conducted in India, where PurpleAir sensors slightly overestimated PM2.5 measurements during the wet winter period, but underestimated PM2.5 values by 2-6 times in the dusty pre-monsoon season [38]. These

observations are especially relevant for Kazakhstan, where the heating season produces coal-dominated aerosol, and spring winds bring mineral dust from the steppe regions.

There are various calibration methods to improve low-cost IoT sensor accuracy. The most popular calibration method is co-location calibration, where low-cost sensors are installed near reference stations [35]. As illustrated in Figure 2.5, a network of low-cost sensors can be located next to the BAM 1020, where the calibration algorithm compares their data and builds an empirical calibration model [15]. Due to its adaptability to the installation location, this calibration method accounts for local climate conditions, and improves measurement accuracy [6]. For the correct co-location calibration, the low-cost sensors have to locate close to reference stations, usually within 50 m of the active FRM/FEM monitoring systems [21], but in some studies, the low-cost sensors were located within 0.5km of the reference station, which remains the standard in the calibration literature [20]. However, performance is improved by reducing the distance between reference stations and low-cost sensors.

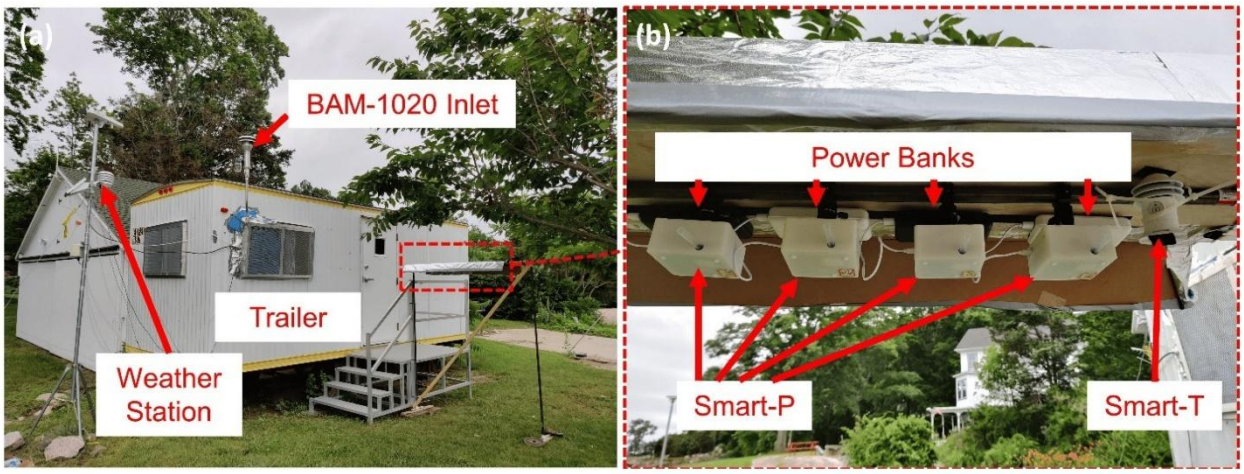


***Figure 2.5: Co-location calibration method based on BAM 1020 and IoT sensors [15].***

Different studies use different periods of co-location calibration, ranging from a few days to several months, highlighting that there is currently no standardized duration for such calibration method. Two independent studies indicate that approximately 6 weeks is the practical minimum duration for a co-location calibration period. When the calibration period exceeds approximately 6 weeks, RMSE values become small for several types of sensors [41]. When the training dataset

contains less than ~1000 hourly observations, which corresponds to about 6 weeks of continuous measurements, then the calibration accuracy is reduced [42]. Figure 2.6 illustrates that after 6 weeks (42 days) the calibration accuracy reaches the optimal value to be used in real air pollution monitoring systems.

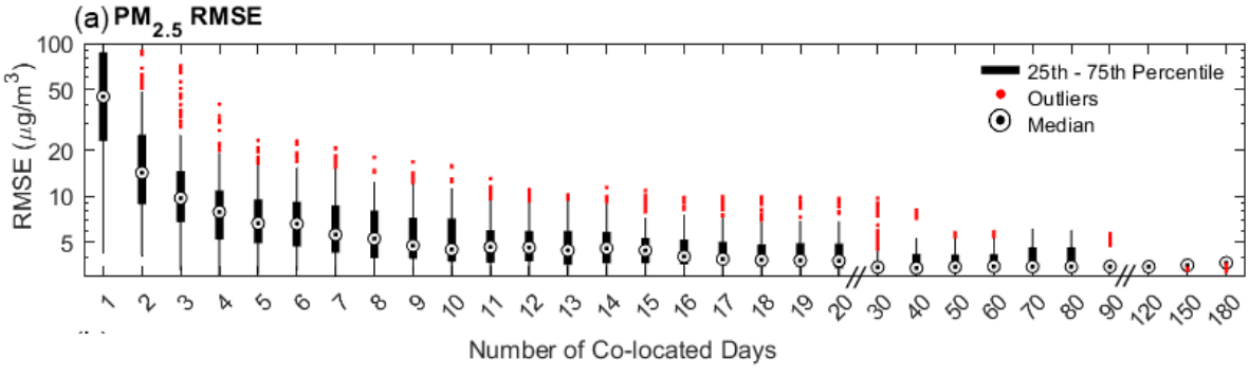


***Figure 2.6: Impact of co-location duration on PM2.5 calibration accuracy. Adapted from [41].***

The key factor is the representativeness of the calibration period relative to the full range of conditions in which the sensor nodes will operate. For PM2.5 optical sensors, RH has to cover at least 70% of the expected range, and temperature has to cover at least 50% [41]. The choice of the wrong calibration period can reduce calibration accuracy 6 times, while RMSE can range from 3.1 to 18.3 µg/m³, depending on the meteorological conditions, which are included in the calibration dataset [41]. These findings are especially relevant for Kazakhstan, where extreme seasonal climate variability, from -30°C in winter to +35°C in summer, requires longer co-location periods or season-adapted calibration strategies.

However, calibration methods do not guarantee high measurement accuracy. Calibration accuracy degrades because one calibration lasts for two weeks, after which sensor drift and other errors appear again [6]. A comprehensive analysis of 14,927 PurpleAir sensors in the United States showed that optical PM sensors like the PMS5003 will remain stable only during the first three years. About 2% of all these sensors have demonstrated permanent calibration accuracy reduction, which is not related to sensor age. The reasons for this behavior were external factors such as dust accumulation, insect intrusion, and physical damage [43]. The speed of such degradation of optical

sensors is also influenced by the climate zone where this optical sensor is located. In hot and dry climates, degradation occurs faster than in other zones due to the dust accumulation in the airway and on optical components. In hot and humid climates, sensors degrade due to moisture damage to electronic components. Based on these observations, it is recommended to delete the first 20 hours of sensor data collection due to the initial measurement instability and replace the Plantower sensors after about 4 years of usage, or after about 3 years in hot and dry climate zones [43].

Despite these limitations, calibration methods enhance the accuracy of low-cost sensors in different environmental conditions. For example, correction equations obtained via co-location calibration enhance the raw sensor data accuracy in multiple independent systems. Monitoring stations in the United States reduced RMSE values from 8 to 3 μg/m³ [21], in Dublin they reduced RMSE values from 52.5 to 2.3 μg/m³ [15], at 22 monitoring stations in Delhi calibration accuracy increased by 37-66% [44]. RH and temperature values collected from the sensor nodes themselves can be used as predictors in calibration models, since the meteorological parameters collected from the sensors correlate well with the reference environment values ($R^2 > 0.85$) [44]. However, the temporal correction coefficients stability remains a crucial limitation for different calibration methods. This variability of the correction coefficients is better explained by the measurement date (44%) than by differences between individual devices (3%). This indicates that the main source of calibration errors is changes in aerosol properties over time [45]. These observations highlight the necessity of automated, adaptive calibration approaches, which can take into account temporary changes in aerosol composition and environmental conditions by using different ML methods.

## 2.4. Machine Learning for Sensor Calibration and Forecasting

Periodic recalibration can show the most efficiency if this calibration method is based on machine learning (ML) or deep learning (DL) models [46]. It is preferable that these ML or DL models use cloud resources to reduce model execution time, enhancing the efficiency and stability of the chosen ML model [47]. Correctly chosen ML algorithms can improve the quality of low-

cost IoT sensors, providing high-accuracy measurements similar to traditional monitoring systems but at a lower cost [35]. There is no universal solution for calibrating low-cost IoT sensors, since each approach can face some limitations. However, any chosen ML models can improve their calibration accuracy by increasing their historical knowledge base [46]. In addition to calibration, ML methods have been applied to predict future PM2.5 concentrations based on historical PM and weather measurements, which is essential for early warning systems and active air pollution management [47][48].

The development of ML-based calibration methods requires the correct selection of suitable input features. The most commonly used bias correction parameters for Plantower sensors are temperature and RH [20][21]. However, including the linear RH term in the calibration model provided the lowest value of RMSE (2.52 μg/m$^3$), while using temperature term just slightly improved the model quality (RMSE = 2.84 μg/m$^3$). Therefore, particle number and RH have the greatest impact on calibration accuracy, while temperature plays a secondary role [42]. In addition, the calibration equations can include other meteorological variables, such as boundary layer height, wind speed, and surface atmospheric pressure [49]. In addition to using different variables for the calibration equation, using spatial autocorrelation based on near-located air pollution monitoring stations can enhance calibration accuracy ($R^2$ from 0.783 to 0.888) [30].

Determining the basic effectiveness of simple calibration methods is essential before using different ML algorithms. The most widely used correction equation was developed by Barkjohn for usage in the United States:

$$PM_{2.5} = 0.524 \times PA_{cf\_1} - 0.0862 \times RH + 5.75$$

where PA_cf1 is a channel with the Plantower PMS5003 correction factor, which provides lower calibration error than the alternative atmospheric correction channel (CF_atm). This linear correction allowed PurpleAir sensors to correctly determine the category of the air quality index (AQI) in 91% of cases [21]. Multiple linear regression (MLR) models can enhance results compared to this basic Barkjohn correction equation. Three-factor MLR model achieved $R^2$ = 73%

and RMSE = 2.96 μg/m$^3$ at hourly resolution and $R^2$ = 79% and RMSE = 2.24 μg/m$^3$ at daily resolution. This corresponds to a 16-23% improvement in error metrics compared to the Barkjohn equation [20]. Nonlinear formulations of the RH effect, including theoretical models of hygroscopic particle growth (κ-Köhler) do not improve calibration accuracy compared to a simple linear dependence on RH [21][38]. When the time averaging period increases, the accuracy of any calibration model increases. For PurpleAir sensors in Dublin, the $R^2$ value increased from 0.74 at hourly resolution to 0.91 at 24-hour resolution, while the RMSE value reduced from 6.6 to 2.3 μg/m$^3$ [15]. Simple equation-based calibration models are able to achieve high correlation with minimal computational effort. However, linear regression is not enough to calibrate PM concentrations under changing meteorological conditions. The correlation between the PM concentrations and environmental weather parameters is nonlinear [30][44]. Table 2.2 shows a comparison of empirical calibration equations developed for PMS5003-based sensors in various geographical and climatic conditions. Despite using the same sensor equipment, the calibration coefficients vary depending on the region and the aerosol characteristics.

**Table 2.2: Comparison of Empirical Calibration Equations for PMS5003-Based Sensors Across Different Regions.**

| Region | Calibration Equation | $R^2$ | RMSE (μg/m³) | Dataset Size | Avg. Period | Source |
|---|---|---|---|---|---|---|
| Dublin, Ireland | 0.536×CF1 – 0.008×RH + 3.21 | 0.91 | 2.3 | 9 264 hours | 24-h | [15] |
| SE United States | 4.33 + 0.418×CF1 – 0.045×RH + 0.075×T | 0.79 | 2.24 | 5 666 observations | 24-h | [20] |
| United States | 0.524×CF1 – 0.0862×RH + 5.75 | — | 3.0 | ~12 000 observations | 24-h | [21] |

Different ML algorithms can be used for low-cost sensor nodes to resolve these non-linear dependencies. These ML algorithms can include tree-based ensemble methods like Random Forest or Gradient Boosting with the highest calibration accuracy from different independent comparative

studies among classic ML algorithms, deep learning models such as ANN or LSTM with advantages in capturing temporal patterns with sequential air quality measurements, etc [42][49]. Table 2.3 provides a comparative analysis of ML-based calibration methods with an accuracy indicator and practical aspects.

**Table 2.3: Comparative Analysis of Machine Learning Models for Air Quality Calibration.**

| **ML Model** | **Advantages** | **Disadvantages** | **$R^2$ value** | **Sources** |
|---|---|---|---|---|
| Linear Regression (LR) | Simple, interpretable, fast | Limited for nonlinear relationships | 0.72–0.81 | [35][42] |
| K-Nearest Neighbors (KNN) | Simple, good for small datasets | Struggles with high-dimensional data | 0.74–0.86 | [35][42] |
| Support Vector Regression (SVR) | Good for nonlinear data | Computationally heavy, $O(n^3)$ training | 0.77–0.79 | [48][49] |
| Random Forest (RF) | Robust ensemble, feature importance, handles nonlinearity | Larger memory footprint | 0.86–0.94 | [42][49] |
| Gradient Boosting (GB) | Best for transferable calibration networks | Sensitive to hyperparameters | 0.87 | [44] |
| Decision Tree Regression (DTR) | Minimal computational requirements | Prone to overfitting | 0.92 | [49] |
| Artificial Neural Network (ANN) | Captures complex relationships | Computationally heavy, less transparent | 0.89 | [42][50] |
| LSTM | Handles sequential data well | Needs large dataset, slow training | 0.82–0.86 | [49][50] |
| BiLSTM | Bidirectional context capture | Highest parameter count | 0.98 | [49] |

The choice of algorithm also depends on the available training dataset size. Tree-based ensemble methods remain the most stable choice for small datasets, which are typical of short-term co-location experiments [49]. At the same time, with very small datasets, like two weeks of hourly measurements, SVM methods and neural networks show high calibration accuracy, demonstrating low sensitivity to the training sample size. Regression methods like Support Vector Regression (SVR) are recommended for projects with limited resources to adjust hyperparameters, because such methods provide an optimal balance between calibration accuracy and computational efficiency [42]. In addition to dataset size, ML-based calibration faces several practical limitations. Firstly, some complex ML algorithms cannot be broadly implemented into city monitoring systems due to limited interpretability and transferability. Secondly, most ML algorithms assume a Gaussian distribution and data independence, while air pollution data exhibit temporal dependencies and an unstable distribution [48]. Thirdly, the only two-year lifetime of low-cost IoT sensors limits the training dataset size, reducing initial model accuracy, especially in early deployment stages [50].

Temporal and spatial stability of calibration methods remains a critical limitation despite the improvements, which are achieved by using ML-based calibration. A calibration model, which is trained during one season is unreliably transferred to other seasons. For example, a calibration model trained on data collected before the monsoon in India was evaluated during the post-monsoon and winter seasons and showed normalized RMSE (NRMSE) greater than 100% and $R^2$ below 0.1. These results indicate complete breakdown of calibration characteristics in different seasonal conditions [38]. Figure 2.7 illustrates this temporal instability across three sites in India. These results highlight that seasonal differences in aerosol composition and meteorological conditions can lead to large calibration inaccuracies during different seasons. In Kazakhstan, the heating season produces an aerosol with a predominance of coal and PM2.5 concentrations several times greater than in summer. This seasonal variability requires seasonal recalibration and adaptive ML models.

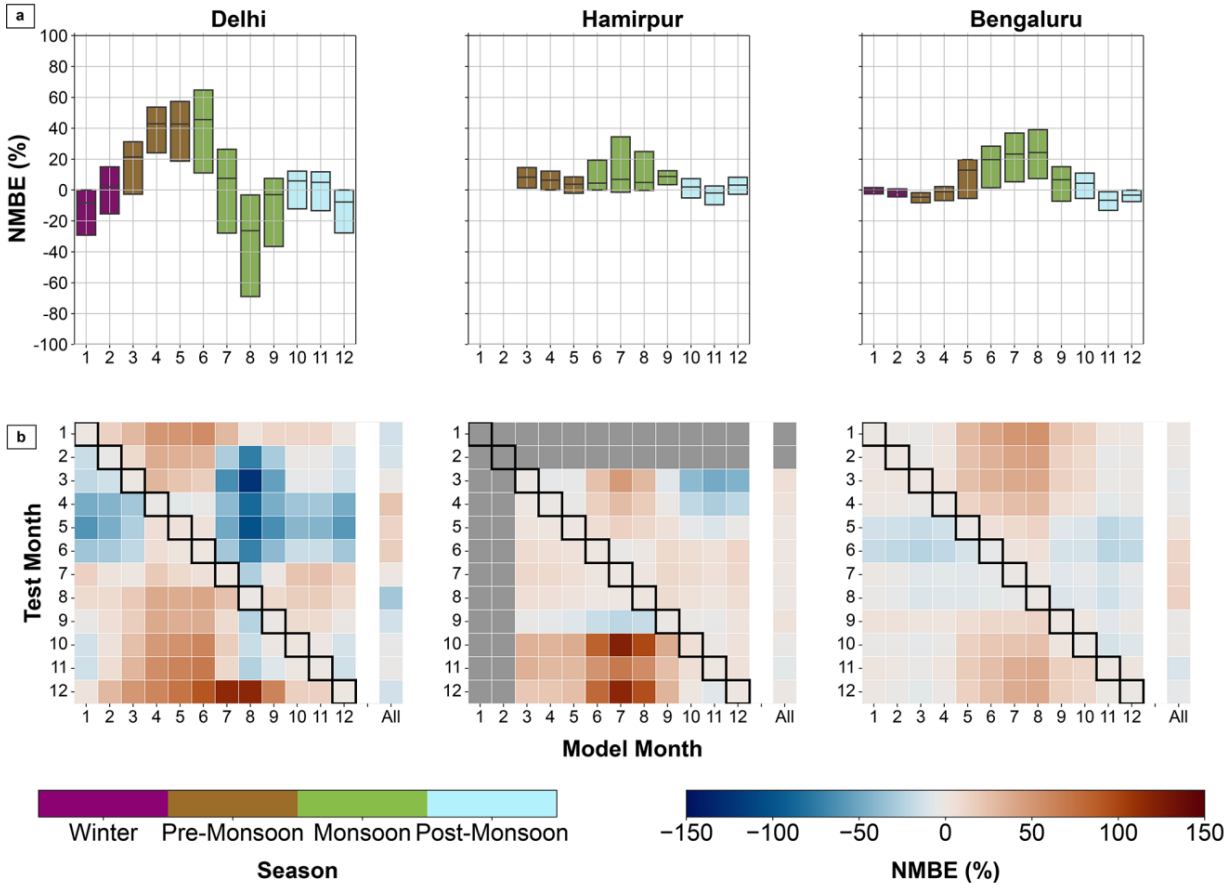


***Figure 2.7: Inter-seasonal transferability of 4-week calibration models at three sites in India (Delhi, Hamirpur, and Bengaluru) [38].***

An additional limitation is the spatial transferability of calibration models, since the spatial distance between training and deployment locations is a poor indicator of model transferability. In a comprehensive evaluation across 22 monitoring sites in Delhi, only 2 sites (9%) produced calibration models that met the U.S. Environmental Protection Agency (EPA) performance criterion ($R^2 \geq 0.70$) at all other sites in the network, as illustrated in Figure 2.8 [44]. This heatmap demonstrates that the highest $R^2$ values are concentrated along the diagonal, where the training and testing data were collected from the same location. Most of the non-diagonal cells are below the 0.7 threshold. The transferability is asymmetric, a model trained at location A can be successfully transferred to location B, but a model trained at location B cannot necessarily be transferred to location A [44].

| Sites | S1 | S2 | S3 | S4 | S5 | S6 | S7 | S8 | S9 | S10 | S11 | S12 | S13 | S14 | S15 | S16 | S17 | S18 | S19 | S20 | S21 | S22 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| S1 | 0.91 | 0.81 | 0.73 | 0.82 | 0.58 | 0.79 | 0.80 | 0.84 | 0.86 | 0.82 | 0.86 | 0.85 | 0.81 | 0.57 | 0.82 | 0.82 | 0.10 | 0.75 | 0.86 | 0.76 | 0.68 | 0.53 |
| S2 | 0.88 | 0.93 | 0.82 | 0.88 | 0.67 | 0.88 | 0.87 | 0.87 | 0.91 | 0.90 | 0.90 | 0.89 | 0.77 | 0.70 | 0.87 | 0.88 | 0.23 | 0.77 | 0.86 | 0.89 | 0.69 | 0.54 |
| S3 | 0.79 | 0.88 | 0.94 | 0.91 | 0.88 | 0.92 | 0.89 | 0.81 | 0.85 | 0.90 | 0.87 | 0.84 | 0.72 | 0.91 | 0.85 | 0.88 | 0.25 | 0.87 | 0.82 | 0.90 | 0.90 | 0.86 |
| S4 | 0.84 | 0.92 | 0.92 | 0.94 | 0.87 | 0.93 | 0.91 | 0.89 | 0.91 | 0.93 | 0.91 | 0.88 | 0.79 | 0.85 | 0.90 | 0.91 | 0.16 | 0.88 | 0.85 | 0.89 | 0.90 | 0.80 |
| S5 | 0.74 | 0.88 | 0.88 | 0.87 | 0.91 | 0.86 | 0.83 | 0.79 | 0.84 | 0.84 | 0.83 | 0.82 | 0.74 | 0.88 | 0.84 | 0.85 | 0.07 | 0.87 | 0.79 | 0.85 | 0.88 | 0.84 |
| S6 | 0.84 | 0.93 | 0.93 | 0.95 | 0.90 | 0.96 | 0.93 | 0.87 | 0.91 | 0.93 | 0.92 | 0.90 | 0.78 | 0.90 | 0.90 | 0.93 | 0.62 | 0.91 | 0.87 | 0.94 | 0.90 | 0.85 |
| S7 | 0.83 | 0.90 | 0.89 | 0.92 | 0.73 | 0.92 | 0.94 | 0.88 | 0.91 | 0.92 | 0.91 | 0.89 | 0.75 | 0.75 | 0.88 | 0.91 | 0.69 | 0.84 | 0.86 | 0.90 | 0.80 | 0.69 |
| S8 | 0.74 | 0.75 | 0.78 | 0.81 | 0.66 | 0.75 | 0.80 | 0.89 | 0.84 | 0.78 | 0.84 | 0.84 | 0.82 | 0.65 | 0.83 | 0.83 | 0.16 | 0.82 | 0.81 | 0.74 | 0.72 | 0.61 |
| S9 | 0.86 | 0.90 | 0.85 | 0.89 | 0.74 | 0.88 | 0.90 | 0.90 | 0.95 | 0.92 | 0.93 | 0.91 | 0.89 | 0.75 | 0.89 | 0.91 | 0.55 | 0.88 | 0.89 | 0.79 | 0.74 | 0.61 |
| S10 | 0.82 | 0.92 | 0.88 | 0.91 | 0.87 | 0.91 | 0.91 | 0.89 | 0.92 | 0.94 | 0.91 | 0.90 | 0.82 | 0.84 | 0.90 | 0.91 | 0.47 | 0.89 | 0.87 | 0.90 | 0.88 | 0.77 |
| S11 | 0.90 | 0.90 | 0.87 | 0.90 | 0.83 | 0.88 | 0.89 | 0.90 | 0.93 | 0.90 | 0.95 | 0.90 | 0.88 | 0.79 | 0.90 | 0.92 | 0.46 | 0.87 | 0.90 | 0.90 | 0.81 | 0.73 |
| S12 | 0.83 | 0.86 | 0.79 | 0.85 | 0.67 | 0.82 | 0.82 | 0.87 | 0.87 | 0.84 | 0.86 | 0.89 | 0.78 | 0.67 | 0.85 | 0.86 | 0.42 | 0.77 | 0.85 | 0.84 | 0.74 | 0.56 |
| S13 | 0.79 | 0.74 | 0.57 | 0.61 | 0.38 | 0.66 | 0.72 | 0.77 | 0.82 | 0.70 | 0.77 | 0.84 | 0.91 | 0.39 | 0.74 | 0.74 | 0.37 | 0.64 | 0.83 | 0.64 | 0.34 | 0.15 |
| S14 | 0.79 | 0.86 | 0.92 | 0.90 | 0.91 | 0.90 | 0.88 | 0.81 | 0.83 | 0.86 | 0.86 | 0.83 | 0.69 | 0.93 | 0.85 | 0.88 | 0.32 | 0.89 | 0.78 | 0.88 | 0.89 | 0.87 |
| S15 | 0.79 | 0.85 | 0.77 | 0.86 | 0.71 | 0.81 | 0.84 | 0.86 | 0.87 | 0.87 | 0.87 | 0.85 | 0.82 | 0.73 | 0.90 | 0.87 | 0.32 | 0.80 | 0.82 | 0.83 | 0.82 | 0.59 |
| S16 | 0.71 | 0.94 | 0.91 | 0.95 | 0.90 | 0.93 | 0.93 | 0.91 | 0.94 | 0.95 | 0.94 | 0.93 | 0.79 | 0.89 | 0.95 | 0.97 | 0.36 | 0.94 | 0.87 | 0.92 | 0.86 | 0.76 |
| S17 | 0.60 | 0.68 | 0.76 | 0.73 | 0.73 | 0.74 | 0.72 | 0.61 | 0.66 | 0.70 | 0.65 | 0.63 | 0.49 | 0.77 | 0.65 | 0.70 | 0.92 | 0.70 | 0.61 | 0.69 | 0.74 | 0.75 |
| S18 | 0.88 | 0.79 | 0.79 | 0.80 | 0.81 | 0.77 | 0.78 | 0.75 | 0.75 | 0.77 | 0.79 | 0.80 | 0.67 | 0.80 | 0.79 | 0.80 | 0.48 | 0.89 | 0.70 | 0.74 | 0.80 | 0.76 |
| S19 | 0.77 | 0.74 | 0.62 | 0.69 | 0.50 | 0.70 | 0.66 | 0.72 | 0.76 | 0.72 | 0.77 | 0.77 | 0.74 | 0.47 | 0.73 | 0.72 | 0.46 | 0.66 | 0.82 | 0.76 | 0.51 | 0.25 |
| S20 | 0.83 | 0.85 | 0.81 | 0.84 | 0.71 | 0.84 | 0.85 | 0.84 | 0.87 | 0.86 | 0.85 | 0.86 | 0.73 | 0.69 | 0.84 | 0.84 | 0.24 | 0.82 | 0.82 | 0.88 | 0.75 | 0.66 |
| S21 | 0.71 | 0.78 | 0.82 | 0.81 | 0.78 | 0.81 | 0.80 | 0.74 | 0.76 | 0.78 | 0.77 | 0.77 | 0.59 | 0.81 | 0.78 | 0.80 | 0.06 | 0.80 | 0.70 | 0.80 | 0.85 | 0.72 |
| S22 | 0.66 | 0.76 | 0.81 | 0.79 | 0.84 | 0.80 | 0.76 | 0.71 | 0.72 | 0.77 | 0.76 | 0.72 | 0.59 | 0.83 | 0.72 | 0.76 | 0.08 | 0.82 | 0.69 | 0.77 | 0.82 | 0.89 |

| Distance | 1-5 | 5-10 | 10-15 | 15-20 | 20-25 | 25-30 | 30-35 | 35-40 |
|---|---|---|---|---|---|---|---|---|
| Color | | | | | | | | |

***Figure 2.8: Spatial transferability of calibration models across monitoring sites [44].***

An important practical aspect in the development of calibration models based on machine learning methods is the validation methodology. Most studies use K-fold cross-validation with k = 5-10 as a standard approach. Specialized strategies such as leave-one-state-out (LOSO) and leave-out-by-date (LOBD) cross-validation are used to evaluate the spatial and temporal generalization. More complex models demonstrate greater instability of coefficients between different data folds [21]. Large air pollution monitoring networks can use a common calibration model for the entire batch of sensors with minimal loss of quality. However, small monitoring networks with a limited number of sensors can use personal models for each sensor [40].

ML methods are used not only for calibration, but also for forecasting PM2.5 concentrations. Such forecasting is essential for early warning systems and proactive air pollution management. The task of forecasting is essentially different from calibration. Calibration corrects the current sensor reading using simultaneously measured environmental weather parameters, while forecasting predicts future concentrations based on historical patterns. Traditional ML models such as Random Forest and XGBoost use lag features and rolling statistics as input and are able to predict PM concentrations several days in advance by using multi-objective regression [47]. Deep learning models such as LSTM are well suited for predicting air quality level due to

their ability to capture long-term time dependencies in consistent data. These architectures provide more resistance to sensor drift compared to static calibration coefficients [48][50]. However, deep learning (DL) models require larger training datasets and a longer training time compared to traditional ML methods. This data requirement limits their applicability in the early stages of sensor network deployment, when the dataset is still small [50].

The analyzed literature shows that Kazakhstan faces high PM2.5 concentrations. Such high air pollution level is caused by coal-dominated heating, extreme continental climate, and insufficient monitoring infrastructure. IoT-based sensor networks offer a cost-effective and easily scalable alternative solution to traditional monitoring stations. However, their raw measurements require calibration due to humidity, variability of parameters between sensor nodes, and long-term sensor drift. Co-location calibration using reference instruments remains the most widely adapted approach, with ML-based models, in particular tree-based ensemble methods, regularly outperforming simple empirical correction equations. However, three critical gaps remain in the existing literature. First, no study has implemented and evaluated an ML-calibrated IoT sensor network for PM2.5 monitoring under the extreme continental climate conditions of Central Asia, where winter temperatures below -30°C and coal-dominated aerosol composition pose unique calibration challenges. Second, while calibration drift has been recorded in temperate and tropical climates, the minimum recalibration frequency required to maintain acceptable measurement accuracy has not been systematically investigated. Third, most forecasting research is based on dense monitoring networks by regulatory authorities, while the integration of calibrated, low-cost sensor data with ML-based forecasting models remains poorly understood. This study addresses these gaps by designing, deploying, and evaluating a three-node IoT sensor network with one co-located calibration sensor node in Astana, calibrated against a Kazhydromet-14 reference station using nine ML models, as well as by evaluating both forecasting performance and age-based calibration coefficient stability.

# Chapter 3 – Methodology

This chapter describes the design and implementation of the IoT-based air pollution monitoring system developed in this study. The methodology consists of three main components of the proposed system: 1) the sensor node used to collect PM concentrations and environmental measurements; 2) the firmware, which is running on the ESP32 microcontroller to control data collection, wireless communication, and power management in C++; and 3) the server-side application, which is used to store, organize, and visualize collected measurements from the cloud database, as illustrated in Figure 3.1.

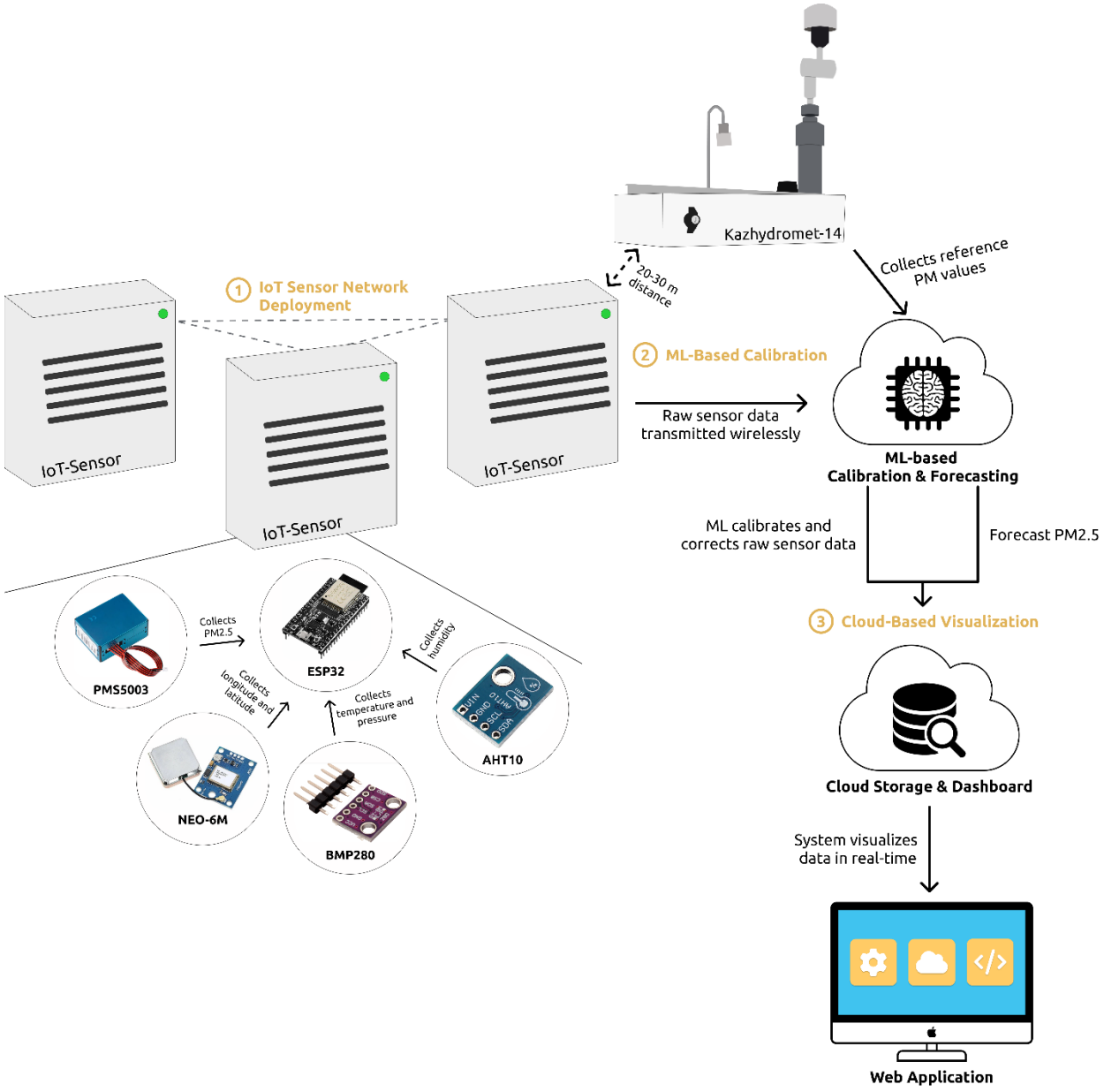


*Figure 3.1: Proposed IoT-based System for Real-time Air Pollution Monitoring*

## 3.1. Hardware Design and Implementation

### 3.1.1. Sensor Node Components

The prototype is a single IoT-based node on ESP32 microcontroller, which is designed to measure three types of parameters: particulate matter concentrations (PM1, PM2.5, PM10), environmental parameters such as temperature, humidity, and atmospheric pressure, and metadata such as coordinates, timestamp and device identifiers.

The proposed prototype uses the Plantower PMS5003 sensor to measure PM concentrations, which are the key parameters for any air pollution monitoring system. This is a low-cost optical PM sensor, which collects PM1, PM2.5 and PM10 concentrations in μg/m³ via a digital interface. The key part of this sensor is a small fan that pulls air into the optical chamber, where an infrared laser and a photodiode measure light scattering from particles. Its popularity in various low-cost air pollution monitoring systems is due to the presence of a well-documented UART protocol and support for three PM channels (PM1, PM2.5 and PM10) in one package [21][32]. In addition, the PMS5003 has a stable frame structure and a warm-up process, and a built-in deep sleep mode controlled by GPIO output for reducing battery power consumption of sensor node.

The proposed prototype uses a compact and low-power digital BMP280 sensor to measure atmospheric pressure and temperature. These measurements can substantially affect the accuracy of low-cost particulate sensors [16]. In addition, the temperature measurements help to assess the sensor stability in cold weather conditions, which is essential for the Kazakhstani case. The BMP280 is suitable for collecting environmental measurements for this prototype due to stable performance over a wide temperature range, sensor drift, fast measurement time, low power consumption, and easy interaction via an I²C interface.

The proposed prototype uses the AHT10 sensor to measure relative humidity, which is also connected via an I²C interface. This is a digital humidity sensor that provides stable and accurate relative humidity measurements with low power consumption. Unlike temperature and atmospheric pressure, relative humidity has a strong effect on the PM measurements. These

Plantower sensors show a stronger correlation with the reference PM optical sensors in the upper quartiles of humidity (76-98% RH) [32]. Due to this high sensitivity, the collected relative humidity measurements will be used as a key input feature in ML calibration models to correct deviations in raw PM measurements.

To enable geolocation into the proposed monitoring system, this prototype uses the NEO-6M module. This is a widely used GPS module, which provides latitude, longitude, and timestamp data via a UART interface. Geospatial metadata is not a physical measurement, unlike PM and weather measurements, but it is essential for air pollution monitoring networks, which require spatial analysis. Coordinates display air pollution levels in different locations, determine spatial gradients, and support multi-node deployment, where each node is uniquely located in geographical space. However, the current embedded code transmits coordinates only at the initial stage of deployment to reduce power consumption.

The core of the sensor node is an ESP32 microcontroller that manages all IoT node operations like data collection and data transfer to a cloud server. This microcontroller is chosen due to its high-performance dual-core Xtensa LX6 processor, embedded Wi-Fi module, multiple UART and I²C interfaces, and low-power modes, which is suitable for battery-powered standalone devices. This microcontroller is a practical solution for real-time air pollution monitoring systems that require frequent measurements, wireless data transmission, and long battery life due to above-mentioned capabilities. The PMS5003 communicates with the ESP32 via the UART2 interface, where the RX/TX pins are connected to GPIO 25 and GPIO 26. The BMP280 and AHT10 sensors communicate via a common I2C interface over the SDA and SCL lines. The PMS5003 deep sleep mode manage pin is connected to a special GPIO, which allows the ESP32 to switch the sensor to low-power mode between measurement cycles at a certain interval. The NEO-6M GPS module is connected to an additional UART interface that supports geolocation measurements. Table 3.1 shows the list of hardware components, which is used in the sensor node.

**Table 3.1: Summary of IoT sensor node components.**

| Component | Function | Interface | Key Specifications |
|---|---|---|---|
| ESP32 DevKit | Microcontroller | — | Dual-core 240 MHz, Wi-Fi, deep sleep |
| PMS5003 | PM1, PM2.5, PM10 | UART2 (GPIO 25/26) | 0–500 µg/m³, ±10 µg/m³ |
| BMP280 | Temperature and pressure | I²C (GPIO 21/22) | −40 to +85°C, 300–1100 hPa |
| AHT10 | Relative Humidity | I²C (GPIO 21/22) | 0–100% RH, ±2% accuracy |
| NEO-6M | GPS (lat, lon) | UART1 (GPIO 16/17) | 2.5m accuracy, 4+ satellites |
| LM2596S | Voltage regulation | — | Input up to 40V, output 5V stable |
| Tactile button | Wi-Fi reset | GPIO 27 | Long press (3 sec) to reset credentials |

### 3.1.2. Hardware Assembly

The sensor node was constructed on a solderless breadboard to provide flexibility for fast changing the circuit. All above-mentioned components were connected by pin assignments, which is described in Table 3.1. This implementation on the breadboard is shown in Figure 3.2. Since the plug-in connections on the breadboard provided sufficient mechanical reliability for a stationary outdoor installation, such a breadboard-based assembly was kept as the final configuration for deployment. During the data collection at temperatures ranging from −26.8°C to +25.6°C, there were no electrical connection failures or problems with pin desynchronization on any of the deployed sensor nodes. Each sensor node was assembled according to an identical connection scheme.

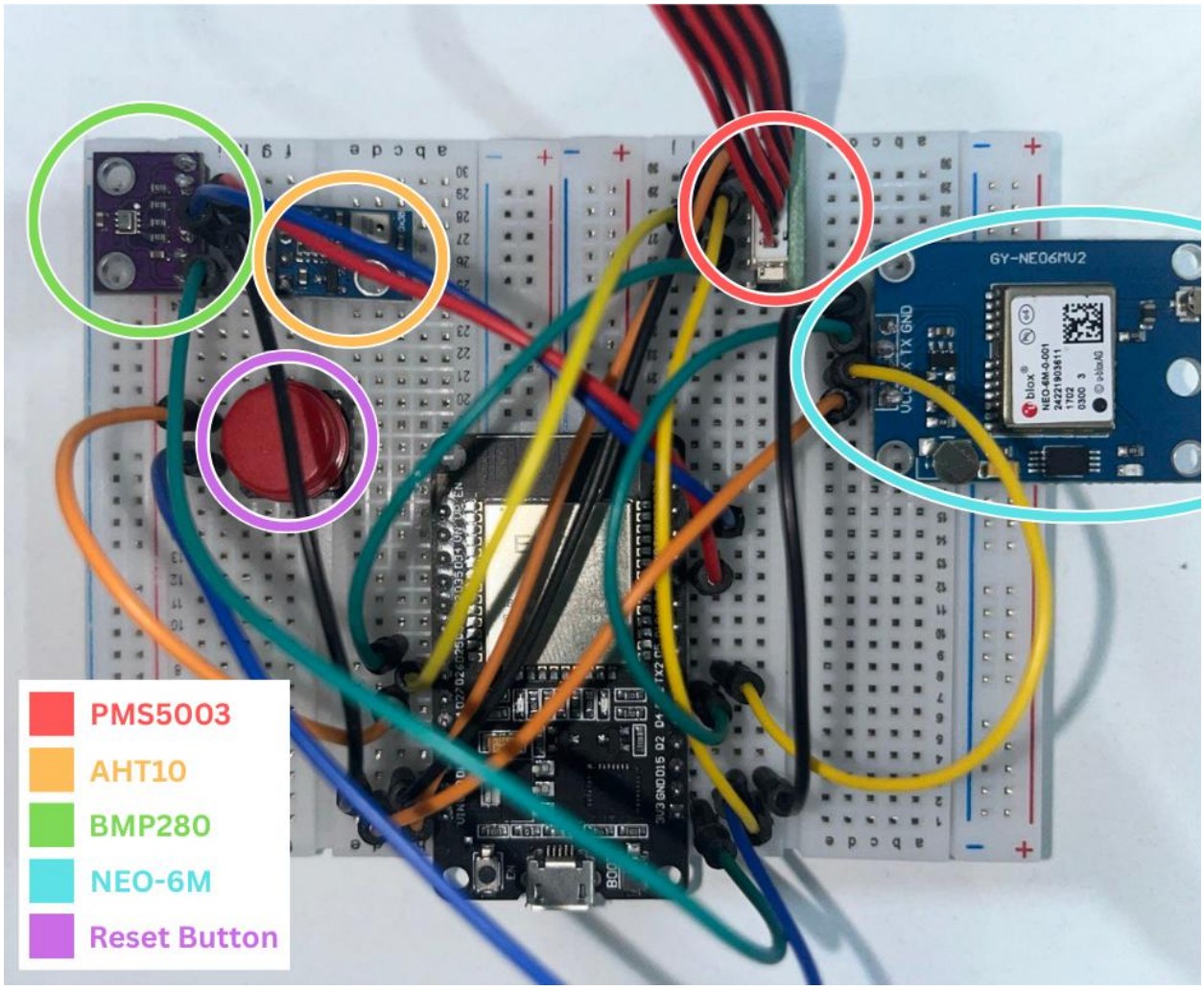


*Figure 3.2: Hardware implementation of the IoT air pollution monitoring node*

### 3.1.3. Custom PCB Design

To adapt the prototype based on breadboard to a more compact and reliable solution, which is suitable for field deployment, a special printed circuit board (PCB) was developed by using the special online design tool EasyEDA. This PCB layout integrates all the electrical connections between the ESP32 microcontroller, all above-mentioned sensor modules, reset button, and power supply on a single board to eliminate loose wiring and pin reliability issues. In this PCB layout, the ESP32 DevKit module is placed in the center and is mounted using two 15-pin connectors. The proposed PCB layout uses net labels rather than physical wires in the schematic, which is consistent with the recommendations for ensuring clarity of the PCB design. The PCB was designed as a two-layer board with a blue solder mask and a lead-free HASL surface coating, as shown in Figure 3.3.

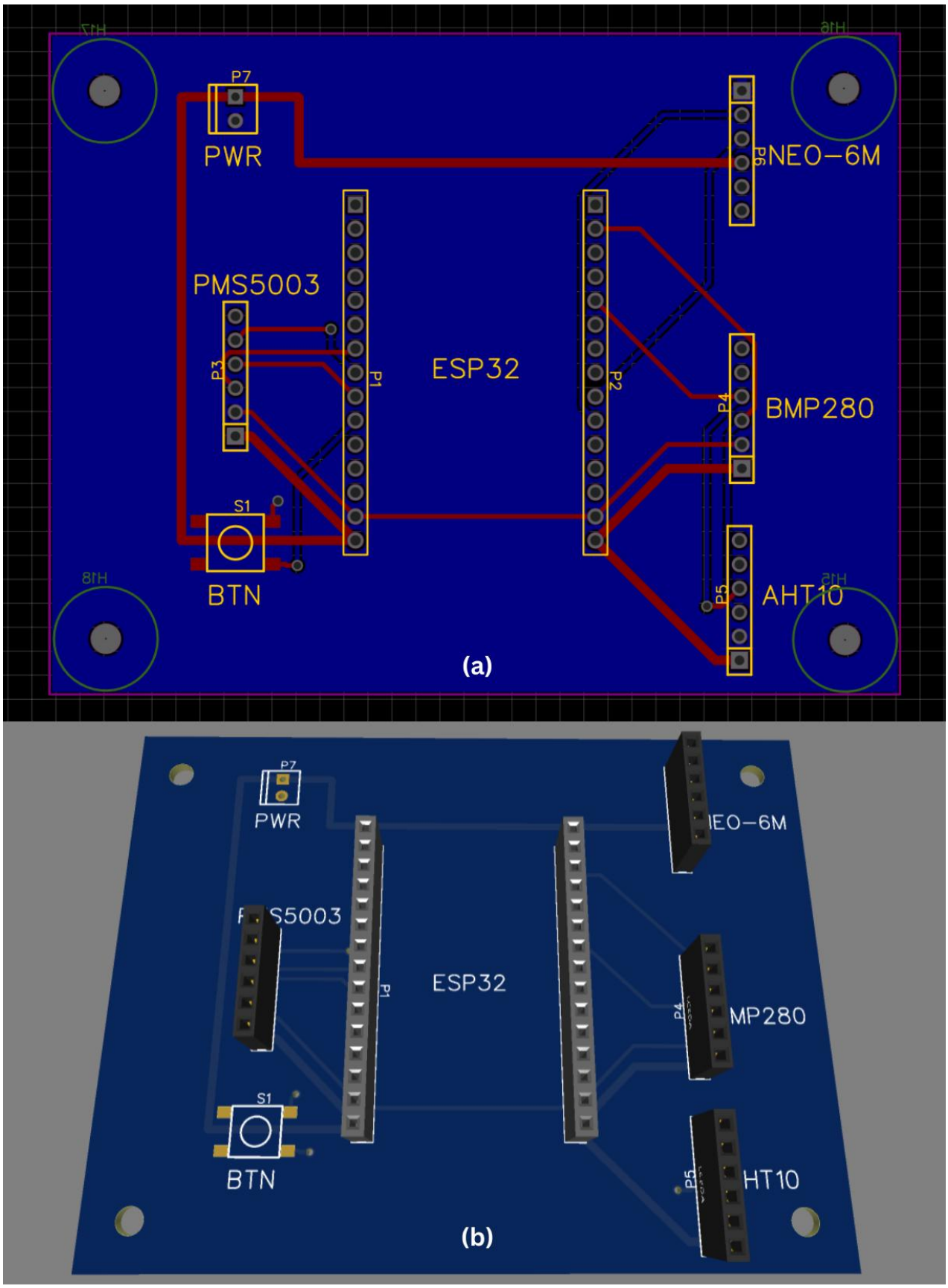


*Figure 3.3. Custom PCB design: (a) PCB layout; (b) 3D rendering.*

However, the PCB could not be delivered to Kazakhstan within the thesis timeline due to international transport restrictions. The manufacturing order with the proposed PCB was placed via JLCPCB, which is one the most popular PCB fabrication service due to its direct export of EasyEDA design files. Despite the successful manufacturing, the fabricated boards could not arrive in Astana due to unresolved bureaucratic issues, which are related to the customs clearance process of electronic components imported into Kazakhstan. Several attempts to remove these administrative barriers within the established time frame have not been successful. As a result, the breadboard-based version on sensor-node described in section 3.1.2 remained the final deployment configuration for all four sensor nodes. During the 28-day data collection period at temperatures from -26.8°C to +25.6°C, there were no electrical failures or problems with pin desynchronization, which confirms sufficient mechanical reliability during the study. The final PCB design remains available at EasyEDA for future sensor assembly improvements, where it will increase mechanical reliability, reduce the overall form factor, and eliminate the risk of loose contact connections during prolonged outdoor use.

### 3.1.4. Power Supply of Sensor Node

The sensor node is powered by a multi-section battery pack based on four 18650 Li-ion cells, each rated at 3.7V, which ensures autonomous operation during long-term field measurements. This battery pack can reach up to 16.8 V when fully charged, which exceeds the allowed range of the ESP32 input signal and the connected sensors. Therefore, the power supply system includes an LM2596S buck converter, which lowers the voltage to a stable level suitable for the microcontroller. The LM2596S forms a stable 5V line for the PMS5003 and provides stable 3.3V for the ESP32 and I²C sensors via the microcontroller's onboard regulators. Figure 3.4 illustrates how this power system is assembled directly on the breadboard, which simplifies testing of different voltage configurations.

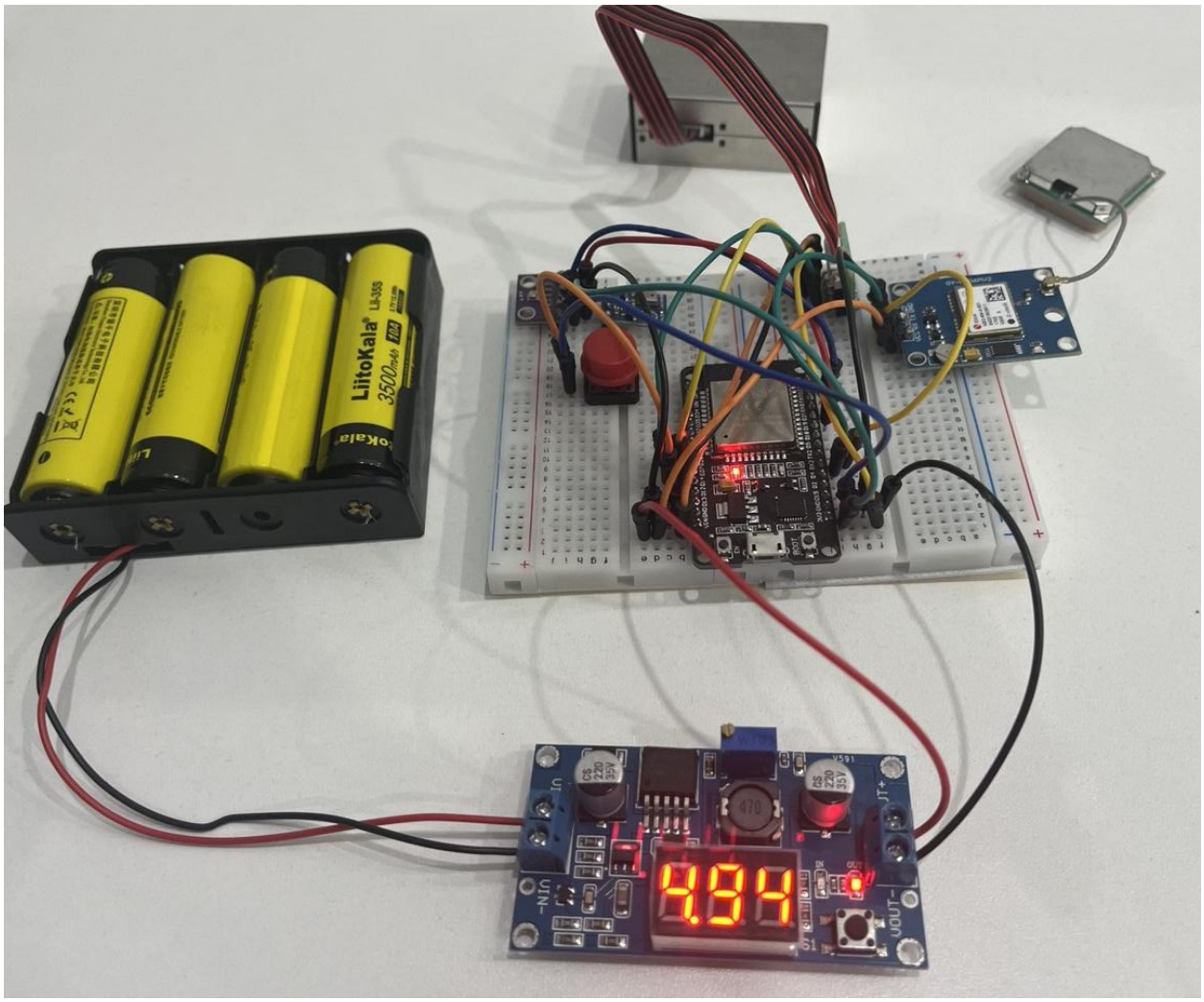


***Figure 3.4: Power supply system of the IoT prototype, including the 4×18650 battery pack and LM2596S buck converter.***

The total nominal battery capacity of the 4-cell series pack is 3500 mAh at 14.8V, which provides an energy reserve of 51.8 Wh. Considering the LM2596S conversion efficiency of up to 92%, the available energy at the output of 5 V is approximately 47.7 Wh. In the deep sleep mode, which is described in section 3.2.1, the measured 24-hour energy consumption is 7.6 Wh. This energy consumption provides a theoretical battery life of approximately 6 days under nominal conditions. However, at low temperatures, the lithium-ion cells capacity is reduced. At -15°C, the effective battery capacity decreases by 30-50% compared to the nominal value. The battery packs were replaced every 2-3 days during the entire observation period to ensure continuous data collection, since priority was given to data completeness over maximum battery utilization.

### 3.1.5. Field Enclosure and Sensor Deployment

The assembled sensor nodes were enclosed in an IP65-rated ABS enclosure (200 × 120 × 75 mm) to protect against precipitation, dust and direct wind exposure when placed outdoors as illustrated in Figure 3.5. All sensor modules and a rechargeable battery are compactly housed in the enclosure. The PMS5003 was installed near to the wall of the enclosure to provide an direct airflow. The BMP280 and AHT10 were installed in a partially open part of the enclosure to ensure that temperature and humidity measurements would be based on the environmental conditions, not inside the enclosure.

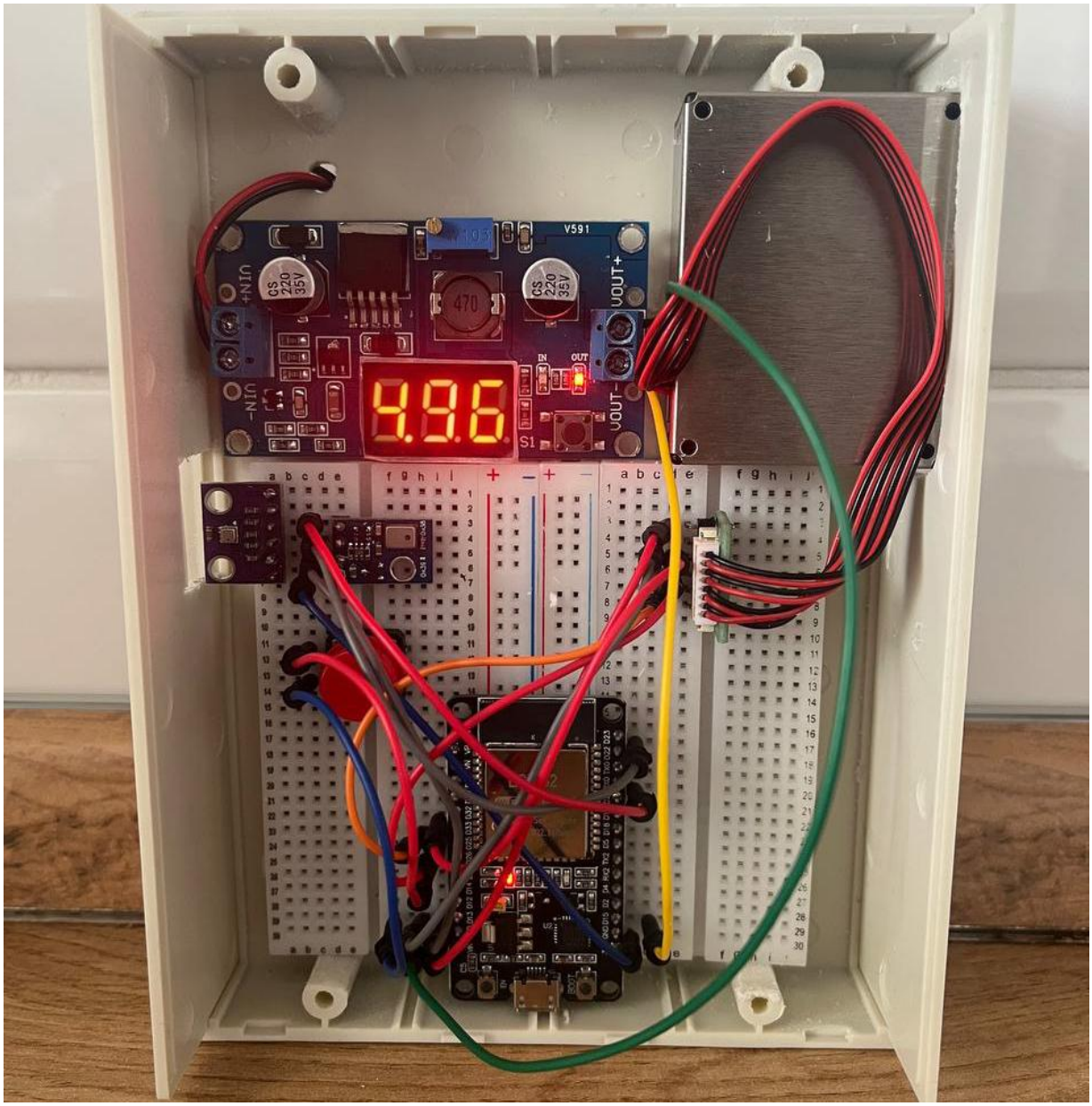


***Figure 3.5: Sensor node in IP65 ABS enclosure prepared for field deployment.***

Three IoT-based sensor nodes were assembled and deployed in different locations of Astana to provide spatial coverage of PM2.5 concentrations. Each node was placed on an open balcony at an elevation corresponding to the 3rd–4th floor level to ensure sufficient access to outdoor air and at the same time partially protected from direct precipitation and strong wind. However, balcony-level deployment can result in lower PM2.5 measurements compared to ground-level monitoring.

A fourth sensor node (Sensor-4) was installed approximately 20–30 meters from the Kazhydromet-14 air quality monitoring station, which is equipped with a BAM-1020. Such co-location distance of 20-30 meters is within the available range, which is described in the calibration literature (up to 500 meters) [20][21]. This Beta Attenuation monitor is installed at an altitude of 11-12 meters above ground level. This co-location installation provides paired measurements between the low-cost PMS5003 sensor and the BAM-1020 as reference station. These paired measurements are used for ML-based calibration, which is described in Section 3.4.

## 3.2. Software Implementation

The code running on the ESP32 microcontroller defines the full pipeline of data collection, data processing and data transfer inside the IoT node. This is written in C++ using the Arduino framework and initializes several communication interfaces, which are required to connect all above-mentioned sensors.

### 3.2.1. Deep Sleep Architecture

The embedded code works in a duty-cycled mode, which is based on the ESP32 deep sleep function. In deep sleep mode, the main processor, most of the RAM, and all digital peripherals are turned off. This reduces current consumption to about 10 μA. The real-time clock (RTC) controller and the RTC memory remain active. Since these components allow the ESP32 to turn off itself after the time interval has expired. Each time the microcontroller wake up, the function setup() is executed as a new boot. The function loop() is never executed, because the node moves into deep

sleep mode again at the end of setup(). This design turns each measurement cycle into an isolated, autonomous operation, which consists of the following sequential phases, which is illustrated in Figure 3.6:

1. The ESP32 wakes from deep sleep and initializes all communication interfaces: UART2 for the PMS5003 (RX on GPIO 25, TX on GPIO 26), I²C for the BMP280 and AHT10 (SDA on GPIO 21, SCL on GPIO 22).
2. The PMS5003 is powered on via the pin SET (GPIO 33) by setting it to HIGH mode, which starts a 30-second warm-up period. During the warm-up, the laser diode and the internal fan switch to continuous operation mode.
3. During the warm-up period, the ESP32 connects to Wi-Fi and synchronizes its internal clock with an NTP server using the GMT+5 offset for Astana, as described in Section 3.2.3.
4. After warm-up, the PMS5003 starts a 30-second data collection phase. Each valid frame contributes PM values to a running total sum, as described in Section 3.2.2.
5. Environmental measurements such as temperature, pressure, and humidity are read from the BMP280 and AHT10 sensors. Also, the heat index is calculated from temperature and humidity from sensors.
6. The code calculates average PM values from all valid samples, writes a JSON payload object, and transmits it to the server on cloud. If at the transmission stage Wi-Fi was unavailable, the record is buffered in RTC memory for the next cycle, as described in Section 3.2.3.
7. The ESP32 calculates the sleep duration needed to bring the next wake cycle in line with the nearest 10-minute boundary. (e.g., :00, :10, :20). This calculation takes into account the approximately 90-second duration of the active phase, which is subtracted from the time remaining until the next boundary. If the waiting time is less than 30 seconds, the code assigns the next 10-minute boundary. The ESP32 then turns off the PMS5003's power by setting the pin LOW mode, and start deep sleep mode until the estimated wake-up time.

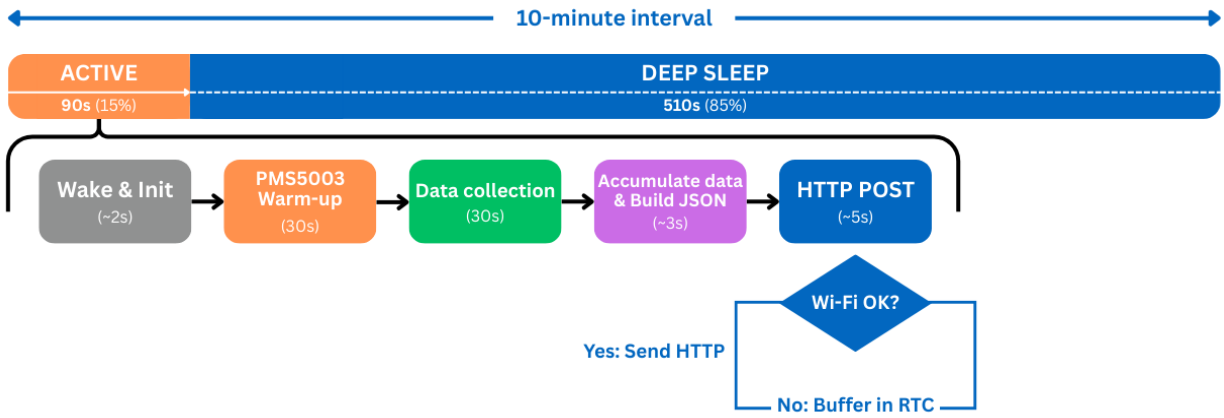


***Figure 3.6: Deep sleep duty cycle of the IoT sensor node.***

The power consumption of the IoT node was evaluated under two operating regimes: continuous operation (without deep sleep) and duty-cycled operation (with deep sleep), using a USB power analyzer connected in series with the power supply. In both cases, the device was monitored over a 24-hour period under identical conditions. Without deep sleep, the IoT-based sensor node kept a constant current of approximately 90 mA at a voltage of 5.10 V, resulting in a total energy consumption of 14,050 mAh for 24 hours. With enabled deep sleep mode, when the sensor node wakes up approximately every 10 minutes for a 90-second active phase, the total energy consumption during the same period decreased to 1,492 mAh. This means a reduction in power consumption of about 9.4 times. This finding directly leads to increased battery life when deployed offline.

### 3.2.2. Sensor Data Collection

The Plantower PMS5003 sensor is configured on the UART2 interface, where the RX and TX are connected to GPIO 25 and GPIO 26. This optical PM sensor transmits data in the form of binary frames of 32 bytes each at a rate of one frame per second. The code includes a custom parser that checks the header, frame length, checksum and extracts the values of PM1, PM2.5, and PM10. This parser checks each frame by checking the initial bytes (0x42, 0x4D), checking the frame length field, and calculating the checksum from the first 30 (0-29) bytes to compare with the checksum field in bytes 30-31. Only those frames, which have passed all three validation checks

are accepted for further processing. Using this approach ensures that only correct PM data is processed, which is important in case of a cold start or unstable airflow. During the 30-second data acquisition period described in Section 3.2.1, the firmware typically collects 25-30 acceptable frames. The concentrations of PM1, PM2,5, and PM10 are accumulated as sums, and then divided by the number of frames used, and this average value is sent to the server.

The BMP280 and AHT10 sensors run on a common I²C interface, which is connected via SDA and SCL in GPIO 21 and GPIO 22, respectively. The data from these sensors is read once per cycle using the corresponding Adafruit libraries. The BMP280 displays temperature in degrees Celsius and atmospheric pressure in pascals, which converted to hectopascals before its transmission to the server. The AHT10 displays relative humidity as a percentage. These measurements include ambient temperature, atmospheric pressure, and relative humidity, which are essential for calculating the heat index during each measurement loop. The heat index is calculated based on temperature and humidity values because it reflects the air temperature at different humidity levels using the *computeHeatIndex* method from the DHT11 library. This variable is essential as an additional environmental value for further calibration, because the heat index is more correlated with PM values [18].

The NEO-6M module uses UART1, which is connected to GPIO 16 and GPIO 17. This GPS module is enabled only until a valid coordinate is received. The code always reads the GPS UART stream and checks whether the location is valid and supported by at least four satellites, which is a typical condition for such air pollution real-time monitoring systems. After these conditions are met, the latitude, longitude, and number of satellites are stored in memory, and the GPS module is immediately shut down by using *gpsSerial.end()* to prevent unnecessary energy consumption. This is because GPS modules are among the highest-power sensors in the low-cost IoT segment. If the IoT node is moved, the Wi-Fi configuration reset button can be used to clear the stored coordinates and turn on the GPS module again to get new coordinates.

### 3.2.3. Wi-Fi Management and Data Transmission

Wi-Fi configuration is performed using the WifiManager library, which allows the node to automatically connect to available Wi-Fi networks. In addition, the system switches to access point mode if no saved Wi-Fi credentials are found. When the ESP32 starts up, it checks whether the reset button on the GPIO 27 is pressed down for more than three seconds. If yes, all saved Wi-Fi settings are erased, and WifiManager opens a captive portal named “AirQualitySensor-1_Setup”, which allows it to configure the Wi-Fi connection without flashing the device. This logic is important in the context of mobility for air pollution real-time monitoring. Figure 3.7 demonstrates the workflow of Wi-Fi configuration for the IoT air pollution monitoring node. After establishing a Wi-Fi connection, the firmware synchronizes the ESP32 internal clock with the NTP server (pool.ntp.org), using the GMT+5 offset for Astana. This synchronization is performed during the PMS5003 warm-up period in parallel to avoid loss of data collection phase time. Accurate clock synchronization is crucial for the calibration process. Since the paired sensor and reference measurements are synchronized in time by hour, and timestamp errors exceeding several minutes can lead to incorrect data pairing.

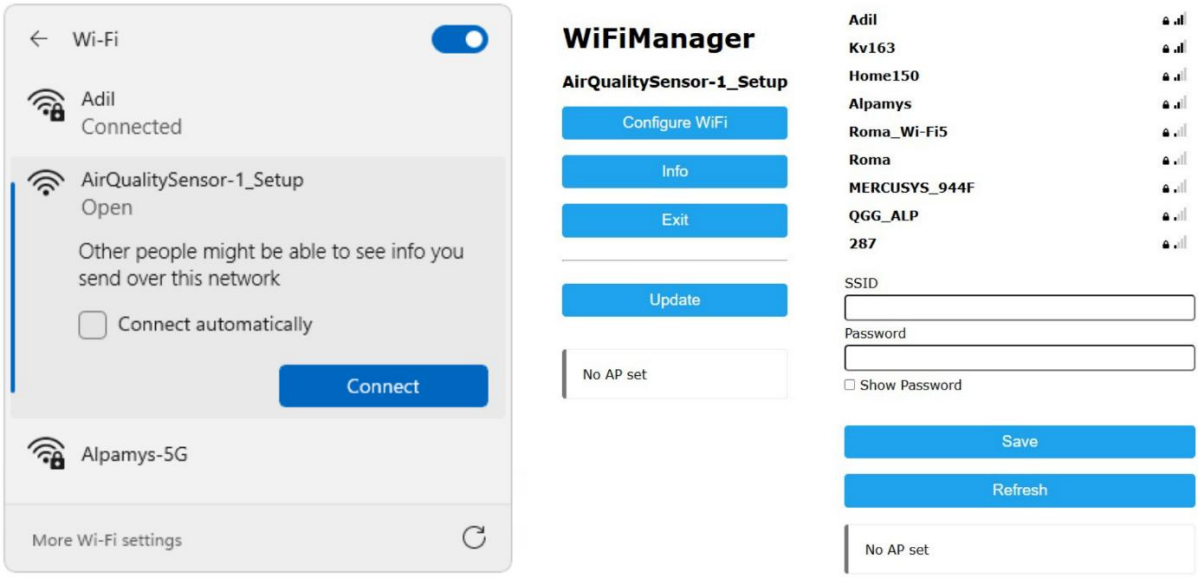


***Figure 3.7: Wi-Fi configuration process: (a) ESP32 access point; (b) WiFiManager setup interface.***

The final stage of the measurement cycle is the data transfer process, which is performed every ten minutes at the time of the next aligned timestamp. The firmware calculates the average values of PM concentrations and environmental measurements, using all correct measurements collected during the 30-second data collection phase from PMS5003, BMP280, and AHT10 sensors. These aggregated values and other values such as coordinates, the number of satellites, and the device metadata form the JSON object. This JSON object transmits to the server-side application via the opened HTTP connection by HTTPClient library of ESP32. The request to the POST HTTP request contains the special header Content-Type: application/json.

## 3.3. API Design and Cloud Integration

The server-side application provides storage and structured access to all measurements collected from the designed IoT nodes. This application is implemented as a RESTful API based on the Express.js framework and Mongoose library on top of the MongoDB Atlas cloud database. Using MongoDB Atlas as a managed cloud database provides high availability, auto backups, and easy scaling in case of memory shortage due to using additional IoT nodes in the future. This application is deployed on a cloud hosting platform with continuous integration and delivery (CI/CD) called Render. CI/CD provides automatic deployment of updates without manual server configuration. This architecture is designed such that the IoT node is responsible only for measuring and sending PM and environmental measurements, and the server-side application validates, stores, and then provides collected data for further visualization and calibration. In addition, the server-side application manages four automated background processes:

1) hourly air quality data collection from the Kazhydromet-14 reference station;
2) real-time calibration of incoming raw sensor measurements;
3) periodic retraining of the calibration model and PM2.5 forecasting;
4) automated sensor health monitoring with email alert notifications.

The starting point for collected measurements from the designed IoT nodes is the *POST /measurement/create* endpoint. On the server side, the Express.js parses the JSON body of the HTTP request and verifies the existence of all required fields (pm1_raw, pm25_raw, pm10_raw, temperature, humidity, pressure, heat_index, latitude, longitude, satellites, and deviceId) to validate them. These collected measurements are stored in the MongoDB collection. The Mongoose models create such collections and define the schema of each document. The schema reflects the structure of the JSON object, which is received from ESP32. In addition to the raw sensor measurements, the schema includes fields for PM calibrated values. These fields are filled in asynchronously after each measurement is saved, because the server starts a calibration model for a specific sensor, calculates the calibrated PM measurements, and updates the database record without blocking the HTTP response. After receiving the raw PM measurements, the server-side application determines the sensor deployment (starting from the first recorded measurement for this DeviceID) and calibrates the PM raw values using the corresponding weekly calibration coefficients from MongoDB. Using MongoDB's flexible document model makes it possible to extend the schema by adding calibration flags, anomaly indicators, or machine learning model results without disturbing existing records. Indexing by createdAt, deviceId, or latitude/longitude fields provides efficient queries by time ranges, nodes, or locations. Such an approach to querying is important for further generating time-series plots, daily summaries, and spatial views of air quality.

To provide the sensor calibration data reliability, the server-side application also automatically collects reference air quality and weather measurements using hourly scheduled tasks. Air quality measurements are taken from the AQICN service for the Kazhydromet-14 station located in Astana, which provides PM2.5 and PM10 as AQI index or $\mu g/m^3$ values. Weather measurements such as temperature, pressure, and humidity are collected from the OpenWeather API for the coordinates of the reference station. Both data sources are extracted in parallel and stored as measurement reference documents in a separate MongoDB collection.

The server-side application also manages two periodic ML tasks using the scheduling *node-cron* library. The calibration model retraining task is performed weekly (every Monday at 3 a.m.), running Python script, which combines sensor and reference data and trains a new ML model for calibration. The forecasting procedure is run daily at 4 a.m. by executing a Python script that processes the latest PM2.5 data and generates a 7-day forecast using a pre-trained machine learning model. This process is run as a child process to isolate Python dependencies from the Node.js runtime environment.

During the data collection, there were some hardware failures. To solve the problem of delayed detection of such hardware failures, the server-side application has an automated sensor health monitoring system, which runs hourly. This monitoring system evaluates three conditions for each registered sensor node:

1) whether the sensor has transmitted data in the last 30 minutes;

2) whether any measurements contain physically implausible values, such as pressure equal to zero or temperature greater than ±50°C;

3) whether the data completeness over the last hour exceeds 67% (at least 4 out of 6 expected records at 10-minute intervals).

After detecting an emergency, the server-side application sends an email notification with HTML template to the system administrator via SMTP by using the Nodemailer library. The deduplication mechanism, which is implemented based on the MongoDB warning collection, prevents repeated notifications of the same emergency by checking for an unresolved warning of the same type for the chosen sensor node.

The server-side application is a data source for the web-based dashboard, which is described in Section 3.5. A separate set of read-only endpoints allows the dashboards to query the latest measurements, filter them by device or time interval, and display them in plots. Separating write and read endpoints ensures the simplicity and reliability of the data reception channel from the designed IoT nodes, and the visualization and analysis layers can be developed independently.

## 3.4. Machine Learning Pipeline for Sensor Calibration and PM2.5 Forecasting

Machine learning in this project solves two different tasks, which are fundamentally different from each other: (1) calibrate raw PM2.5 measurements from the designed and deployed IoT sensor nodes compared data from selected reference station, and (2) forecast PM2.5 concentrations for a 7-day horizon. Figure 3.8 illustrates the complete ML pipeline, including the calibration and forecasting parts and their integration with the production system.

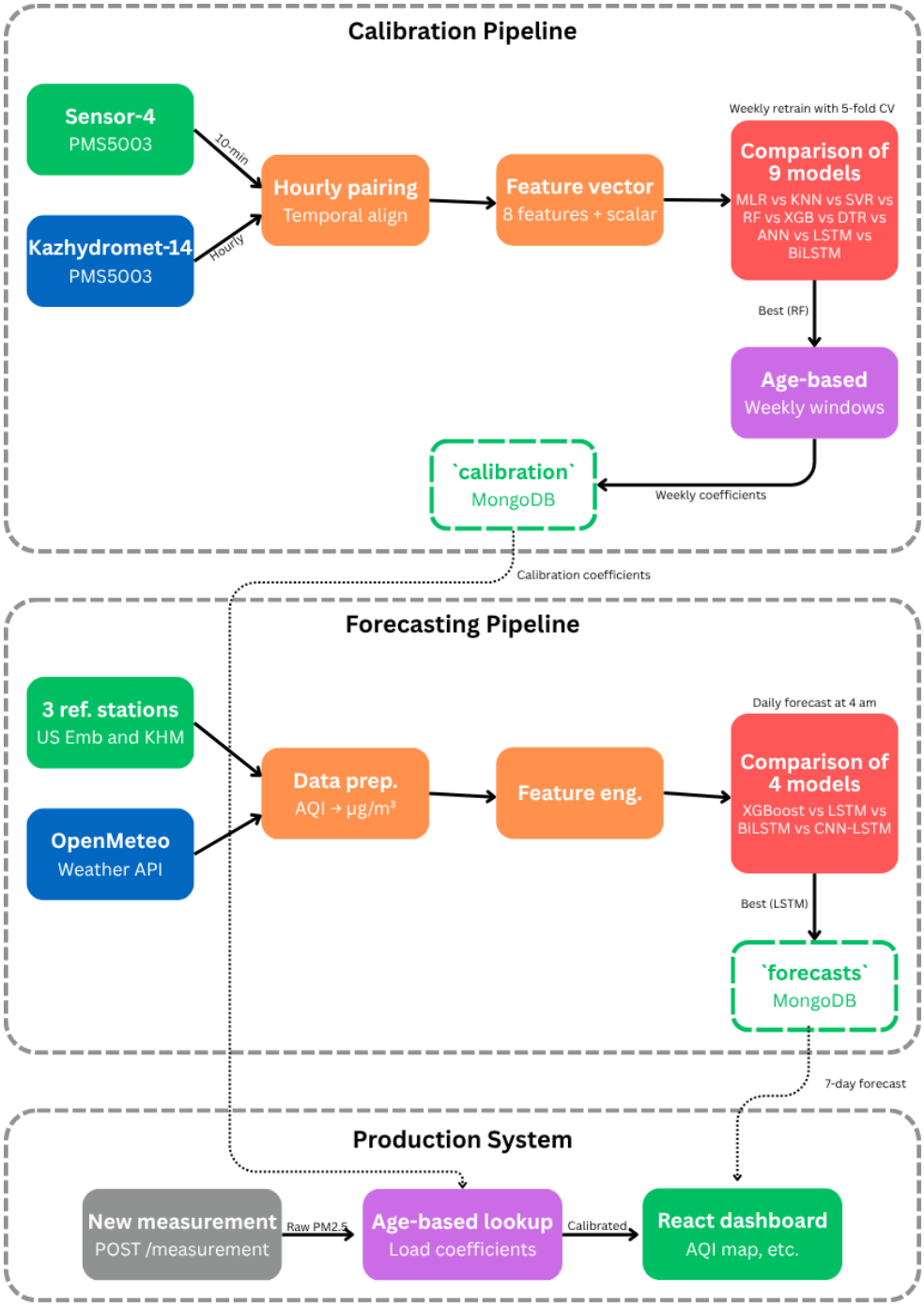


***Figure 3.8: Machine learning pipeline for PM2.5 calibration and forecasting.***

### 3.4.1. Data Preparation and Feature Engineering

The ML-based calibration process uses data from Sensor-4, which is co-located with Kazhydromet-14 reference station, as described in Section 3.1.4, and PM2.5 measurements in µg/m³ from the reference station. The 10-minute Sensor-4 readings are aggregated with hourly values to match the temporal resolution of the reference station, as calibration accuracy enhances with increasing averaging periods [15]. The hourly resolution provides a practical balance between temporal granularity and measurement accuracy [20][21]. The paired dataset is created using an inner join based on an hour timestamp with a minimum value. This approach collects only those hours when both sensor and reference values are available, ensuring each training record contains concurrent observations from both data sources.

The AQICN API returns the PM2.5 concentrations in both µg/m³ or Air Quality Index (AQI) numbers. Therefore, the Python program has conversion method from AQI to mass concentrations in µg/m³, which is running in the data fetching pipeline. The breakpoints of AQI for this conversion is described in Table 3.2. This conversion method runs only during the data fetching from AQICN API. Since the designed IoT sensor node based on the PMS5003 sensor collects PM values in µg/m³.

**Table 3.2: AQI breakpoints for PM2.5 (µg/m³).**

| AQI Category | AQI Range | PM2.5 Concentration (µg/m³) |
|---|---|---|
| Good | 0–50 | 0.0–12.0 |
| Moderate | 51–100 | 12.1–35.4 |
| Unhealthy for Sensitive Groups | 101–150 | 35.5–55.4 |
| Unhealthy | 151–200 | 55.5–150.4 |
| Very Unhealthy | 201–300 | 150.5–250.4 |
| Hazardous | 301–400 | 250.5–350.4 |
| Hazardous | 401–500 | 350.5–500.4 |

The feature vector for the calibration models consists of eight variables, as shown in Table 3.3. The raw PM2.5 concentration from the PMS5003 sensor is the primary predictor. Relative humidity is included as the second most important feature into vector, since the hygroscopic particle growth at high humidity is the main source of measurement error for optical PM sensors and can explain up to 30% of the variance in PM measurements [40]. Temperature and atmospheric pressure are used as meteorological predictors, which influence sensor response characteristics [42]. In addition, the model consists of the heat index, a composite indicator that combines temperature and humidity and demonstrates a stronger correlation with PM measurements than each of these variables separately [23]. The heat index is calculated using the NOAA formula, which is implemented in the DHT library on the ESP32 microcontroller. The feature vector also includes three temporal features:

1) the hour of day due to heating activity and traffic flows;

2) the month reflects seasonal variability due to the heating season [14];

3) the weekday, which reflects weekly patterns of human activity.

**Table 3.3: Feature vector for calibration models.**

| **Feature name** | **Source** | **Unit** | **Justification** |
|---|---|---|---|
| pm25_raw | PMS5003 | µg/m³ | Primary predictor |
| humidity | AHT10 | % | Hygroscopic growth correction |
| temperature | BMP280 | °C | Secondary meteorological factor |
| pressure | BMP280 | hPa | Atmospheric conditions |
| heat_index | Computed | °C | Compound temperature–humidity indicator |
| hour | Timestamp | 0–23 | Diurnal PM2.5 cycle |
| month | Timestamp | 1–12 | Seasonal variability |
| day_of_week | Timestamp | 0–6 | Weekly activity patterns |

All features are normalized using a StandardScaler, which subtracts the mean value and divides it by the standard deviation. The scaler is installed exclusively on the training set and applied to the test set to prevent data leakage. The forecasting process uses a separate historical dataset from several reference stations, which is described detailed in section 3.4.4.

### 3.4.2. Calibration Model Comparison

The calibration model comparison pipeline evaluates nine ML algorithms to determine which provides the highest accuracy for correcting raw PM2.5 values. This comparison directly addresses Research Question 1. The target variable is the reference station PM2.5 concentration in $\mu g/m^3$. The calibration models are trained based on the paired dataset. This co-location approach is consistent with the literature, according to which low-cost sensors are installed near the reference stations to receive measurements to train an empirical model [15][20][21].

The following nine models are compared: Multiple Linear Regression (MLR), K-Nearest Neighbors (KNN), Support Vector Regression (SVR) with RBF kernel, Random Forest (RF), XGBoost, Decision Tree Regression (DTR), Artificial Neural Network (ANN) with two hidden layers and ReLU activation, Long Short-Term Memory (LSTM), and Bidirectional LSTM (BiLSTM). These models were selected based on the comparative analysis presented in Table 2.3. For the classical ML models, hyperparameter tuning is performed using GridSearchCV or RandomizedSearchCV with cross-validation on the training dataset.

The verification strategy uses 5-fold cross-validation. For classic ML models (MLR, KNN, SVR, RF, XGBoost, DTR, ANN), a shuffled KFold is used. For the deep learning (LSTM, BiLSTM) models, TimeSeriesSplit is used to preserve the chronological order of observations and prevent temporal data leakage. Three evaluation metrics are computed for each model: the coefficient of determination ($R^2$), the root mean square error (RMSE, $\mu g/m^3$), and the mean absolute error (MAE, $\mu g/m^3$), which are consistent with the evaluation framework used throughout the calibration literature (Table 2.3). The best model is selected based on the highest mean $R^2$ across the 5 cross-validation folds, with the lowest RMSE used as a tiebreaker.

### 3.4.3. Age-Based Calibration Strategy

The paired dataset is divided into weekly intervals to explore the temporal stability of calibration parameters and determine relevant model update frequency. Two models are independently trained in each window, each of which serves its own purpose. The most efficient model is trained in each window to evaluate whether its calibration accuracy remains stable or degrades as environmental conditions change. In addition, Multiple Linear Regression (MLR) is trained on each window only for analytical purposes. Unlike tree-based or neural network models, MLR generates explicit numerical calibration coefficients for each input feature. Such calibration coefficients can be directly compared in different windows to evaluate changes in the sensor node-reference ratio over time, answering Research Question 2. This approach is based on the methodology of Campmier et al. [38], who demonstrated that regression performance and coefficient stability in different seasonal conditions depend on the calibration period. If MLR coefficients show a systematic drift between weekly intervals, this indicates that periodic model retraining is essential to keep calibration accuracy in the EPA performance criterion ($R^2 \geq 0.70$).

For each weekly period, the MLR model generates a set of calibration coefficients based on the eight input features defined in Table 3.3. These calibration coefficients are stored in a separate MongoDB collection with corresponding performance metrics ($R^2$, RMSE) and environmental context of the training weekly period such as mean temperature or mean reference PM2.5 value.

In production, after receiving a new measurement from the IoT sensor node, the server-side application calculates the sensor deployment time as the number of days since the first recorded measurement for this device. The appropriate coefficients are selected from the MongoDB collection, which are applied to incoming raw PM2.5 measurements based on this age. This automatic age search process provides that the calibration parameters reflect the environmental conditions closest to the current stage of deployment without manual recalibration or continuous co-location with a reference station.

### 3.4.4. Forecasting Model Comparison

The forecasting pipeline predicts daily PM2.5 concentrations for a 7-day horizon based on historical patterns. Unlike calibration, which corrects a current sensor measurement, forecasting generates predictions for future time steps without requiring simultaneous IoT sensor data. The forecasting models are trained on historical reference station data combined with weather parameters, which allows the use of substantially longer time series for training compared to the IoT sensor deployment period.

The training dataset for forecasting is constructed from three reference air quality stations in Astana: the US Embassy station (AQICN), Kazhydromet-9, and Kazhydromet-14. Historical PM2.5 data from these stations is combined with weather parameters (temperature, humidity, pressure, and wind speed) obtained from the OpenMeteo History API for the coordinates of each station. The data preparation pipeline includes AQI-to-μg/m³ conversion for the US Embassy station, filtering of physically implausible values (PM2.5 outside the 0–500 μg/m³ range), linear interpolation to fill gaps of up to 7 consecutive days per station, and selection of the longest continuous data segment per station. The multi-station approach increases the training dataset size and exposes the models to diverse PM2.5 distributions from different locations in Astana.

The feature engineering follows two separate approaches depending on the model type. For XGBoost, explicit lag and statistical features are constructed: 30 lag features (PM2.5 values from the previous 1 to 30 days), rolling means over 7-day, 14-day, and 30-day windows, rolling standard deviation over a 7-day window, weather features with a one-day shift to prevent data leakage, and three temporal features (month, day of week, and day of year), resulting in a total of 41 input features. For the deep learning models, raw feature sequences are used without explicit lag construction: the input consists of the last 30 days of measurements with 8 features per day (PM2.5, temperature, humidity, pressure, heat index, hour, month, and day of year), forming a tensor of shape (batch size, 30, 8). The deep learning models learn temporal patterns directly from the raw sequences, while XGBoost relies on the handcrafted lag and rolling features.

The following four models are compared: XGBoost, trained as a MultiOutputRegressor with 7 separate regressors (one per forecast horizon day), LSTM with two stacked layers (64 and 32 units) followed by a dense output layer with 7 neurons, BiLSTM with the same architecture but bidirectional processing, and CNN-LSTM, a hybrid architecture combining a one-dimensional convolutional layer (64 filters, kernel size 3) with max pooling, followed by an LSTM layer (64 units) and a dense output layer with 7 neurons. The deep learning models are trained with the Huber loss function ($\delta = 15.0$) for robustness to outliers, the Adam optimizer with a learning rate of 0.001, a batch size of 32, and early stopping with a patience of 15 epochs and a maximum of 200 epochs.

The validation strategy employs TimeSeriesSplit with 5 folds. The evaluation metrics are computed per horizon day (RMSE for day 1 through day 7) and as the overall average across all 7 days. The best forecasting model is selected based on the lowest mean RMSE across the cross-validation folds.

## 3.5. Dashboard and Visualization

The visualization layer of the air quality monitoring system is implemented as a single-page web application (SPA), which provides real-time reading to sensor measurements, calibration results, forecasts, and alert history. This client application was developed using React and TypeScript to ensure type security, styled using TailwindCSS, and Leaflet with OpenStreetMap tiles is used for geospatial visualization. The Recharts library is used to display interactive and dynamic charts to illustrate PM2.5 time-series. The client application interacts with the backend API described in Section 3.3 via HTTP requests, which returns the latest sensor readings with calibrated values, device metadata, historical measurements, 7-day PM2.5 forecasting results, and sensor health monitoring data. Figure 3.9 demonstrates the user interface (UI) architecture and the interaction between the dashboard components and the REST API endpoints from the server-side application.

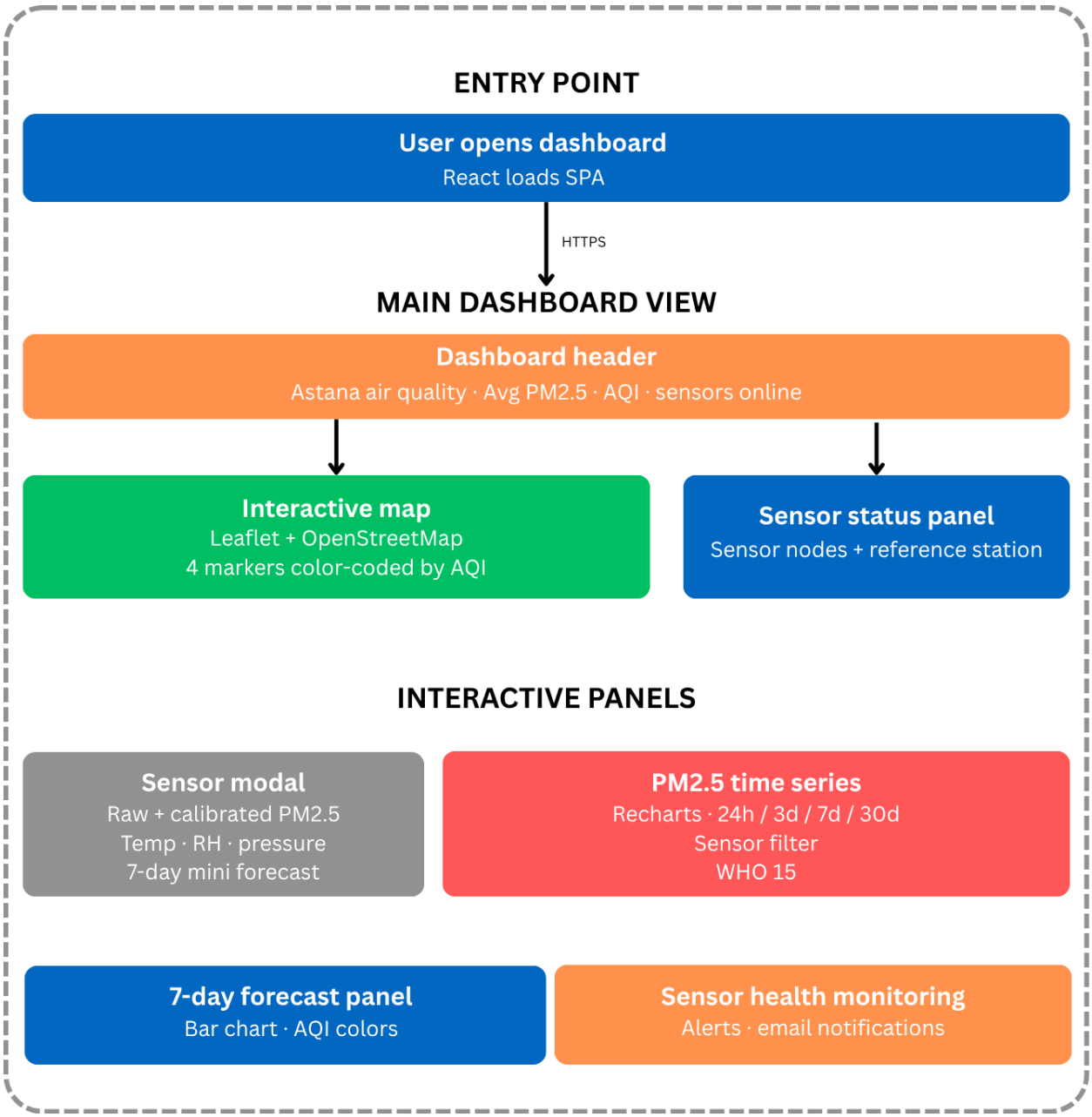


***Figure 3.9: UI architecture and interaction flow of the air quality monitoring dashboard.***

The dashboard displays color-coded AQI markers on the map for each sensor node location. During hover on the marker application, the modal windows display current raw and calibrated PM2.5 measurements and 7-day forecasts. The time-series panel supports interactive filtering by different time ranges (24 hours, 3 days, 7 days, 30 days). The Sensor Health Monitoring section displays a summary of detected warnings and a history of sensor failures, which allows the system administrator to view past hardware failures.

# Chapter 4 – Results and Discussion

## 4.1. Dataset Overview

The dataset used in this study was collected by four IoT sensor nodes deployed across different locations in Astana. The observation period is 28 days, from 25 February 2026 13:40 to 25 March 2026 15:50. The three main sensor nodes (Sensor-1, Sensor-2, Sensor-3) were placed on open balconies at elevated positions (3rd-4th floor) to ensure sufficient access to outdoor air while providing partial protection from direct precipitation and wind. A fourth sensor node (Sensor-4) was placed at a distance of 20-30 meters from the Kazhydromet-14 reference station as described in Section 3.1.4.

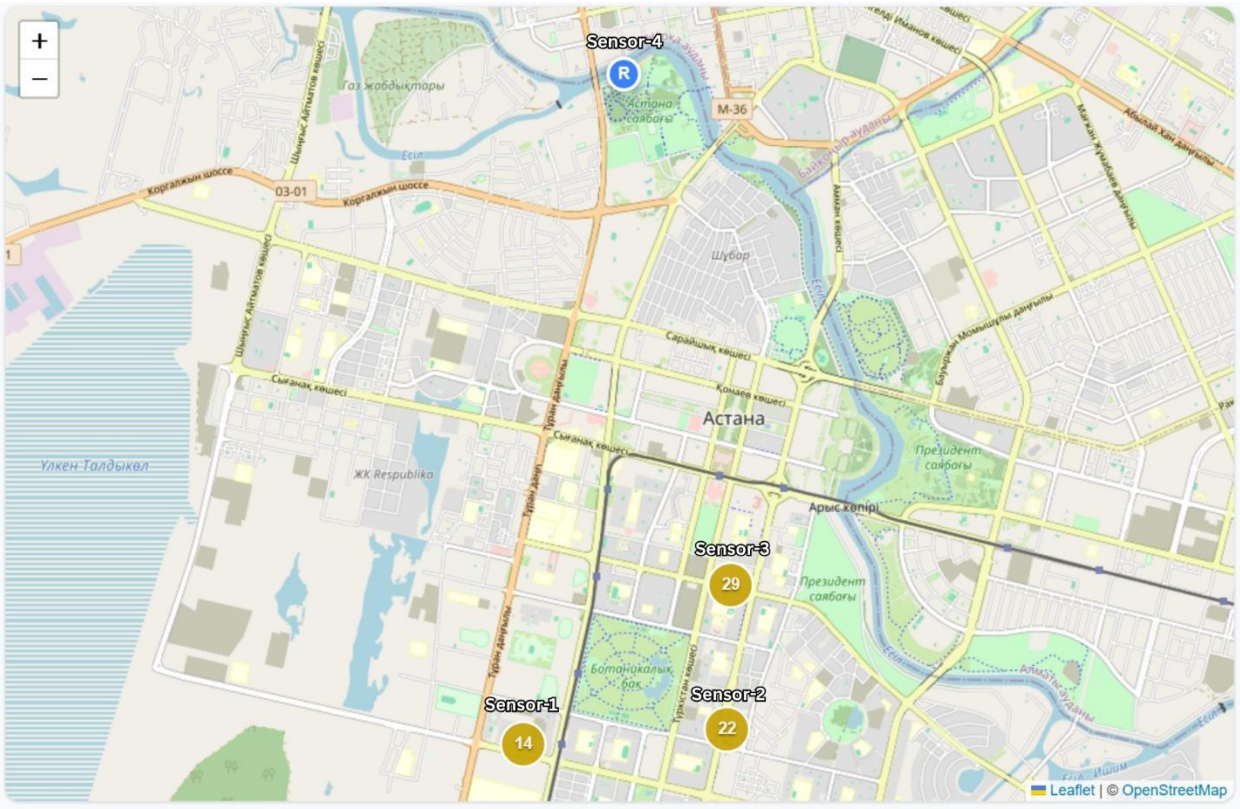


***Figure 4.1: Spatial distribution of the four IoT sensor nodes.***

Table 4.1 summarizes the deployment parameters and data volume for each sensor node. The dataset consists of 14,444 measurements collected from four designed IoT air pollution monitoring nodes. In particular, local features of air flows around the balconies of high-rise

buildings and the effects of vertical dispersion can reduce the PM concentrations, leading to artificially low or even zero PM values. These limitations should be taken into account when interpreting the dataset, especially during periods of high winds or snowfall when PM concentrations can decrease sharply. This dataset, which contains particulate matter concentrations (PM1, PM2,5, and PM10), environmental parameters (temperature, humidity, pressure, and heat index), and device metadata, including GPS coordinates and timestamps, and has 89.95% of data completeness.

**Table 4.1: Summary of IoT sensor node deployment.**

| Sensor ID | Latitude | Longitude | Total Records | Completeness (%) | Observation Period |
|---|---|---|---|---|---|
| Sensor-1 | 51.0999 | 71.4016 | 3,394 | 84.5 | Feb 25 – Mar 25 |
| Sensor-2 | 51.1013 | 71.4296 | 3,840 | 95.6 | Feb 25 – Mar 25 |
| Sensor-3 | 51.1138 | 71.4301 | 3,832 | 95.4 | Feb 25 – Mar 25 |
| Sensor-4 | 51.1580 | 71.4154 | 3,378 | 84.3 | Feb 25 – Mar 25 |
| Total | — | — | 14,444 | 89.95 | 28 days |

Sensor-2 and Sensor-3 showed data completeness above 95%, but Sensor-1 showed a lower completeness of 84.5% due to a hardware interruption described in Section 4.2. The lower data completeness of Sensor-4 was not related to sensor-side failures like in Sensor-1, but due to the absence of buffering mechanism, which is described in Section 3.2.1. The data gaps in the Sensor-4 reflect communication-level losses rather than hardware failures. The data completeness of the real IoT air pollution monitoring systems is in the range from 54.2% to 99.5% [15]. Therefore, the proposed monitoring system demonstrates data completeness within this range.

Table 4.2 illustrates the summary statistics of the collected measurements for all four deployed sensor nodes. PM2.5 concentrations ranged from 0.0 to 378.5 μg/m³ with an average value of 38.8 μg/m³, which is much higher than the WHO recommended daily level of 15 μg/m³ [1]. Environmental measurements reflect the late winter-early spring period in Astana, with temperatures ranging from -26.8 °C to +25.6 °C, and relative humidity ranging from 22.8% to 99.7%.

**Table 4.2: Summary statistics of all collected measurements.**

| Variable | Unit | Mean | Median | Min | Max | Std |
|---|---|---|---|---|---|---|
| PM1 | μg/m³ | 23.5 | 11.5 | 0.0 | 256.3 | 28.9 |
| PM2.5 | μg/m³ | 38.8 | 27.5 | 0.0 | 378.5 | 38.4 |
| PM10 | μg/m³ | 53.6 | 27.3 | 0.0 | 494.9 | 64.5 |
| Temperature | °C | −2.7 | −1.8 | −26.8 | 25.6 | 5.8 |
| Humidity | % | 68.2 | 67.8 | 22.8 | 99.7 | 14.1 |
| Pressure | hPa | 982.0 | 982.1 | 960.4 | 1007.3 | 7.8 |
| Heat Index | °C | −4.8 | −3.9 | −31.4 | 24.9 | 6.2 |

The atmospheric pressure measurements ranging from 960.4 to 1,007.3 hPa, which correspond to the expected values of atmospheric pressure in Astana (altitude approximately 347 m above sea level). Figure 4.2 demonstrates that the PM2.5 time series have two distinct modes during the 28-day data collection period, which captures different weather seasons and shows the behavior of the monitoring system during these periods.

During the first mode (25$^{th}$ February – 5$^{th}$ March), all four sensor nodes recorded high PM2.5 concentrations, with peaks often exceeding 100 μg/m³, and separate observations greater

than 250 µg/m³. This period corresponds to the peak of the heating season in Astana, when coal-fired thermal power plants and domestic coal-fired heating generate large PM2.5 emissions [10][14].

During the second period (from March 6 onwards), the PM2.5 concentrations reduced sharply to the range of 10-50 µg/m³, which is explained by two meteorological factors. Firstly, several snowfalls occurred during this period, which act as a natural mechanism for air cleaning from PM2.5 particles due to wet deposition, due to the effective removal of suspended particles from the atmosphere. Secondly, since the beginning of March, there was a warming (from about -20 °C at the end of February to +7 °C by mid-March), which reduced the domestic heating intensity and enhanced the vertical mixing of the atmosphere [12]. Despite the decrease in PM2.5 concentrations in the second half of data collection period, their levels remained above the WHO daily standard on most days. This confirms that even during the transition period between the heating season and spring, the air pollution level in Astana exceeds the WHO recommended limits.

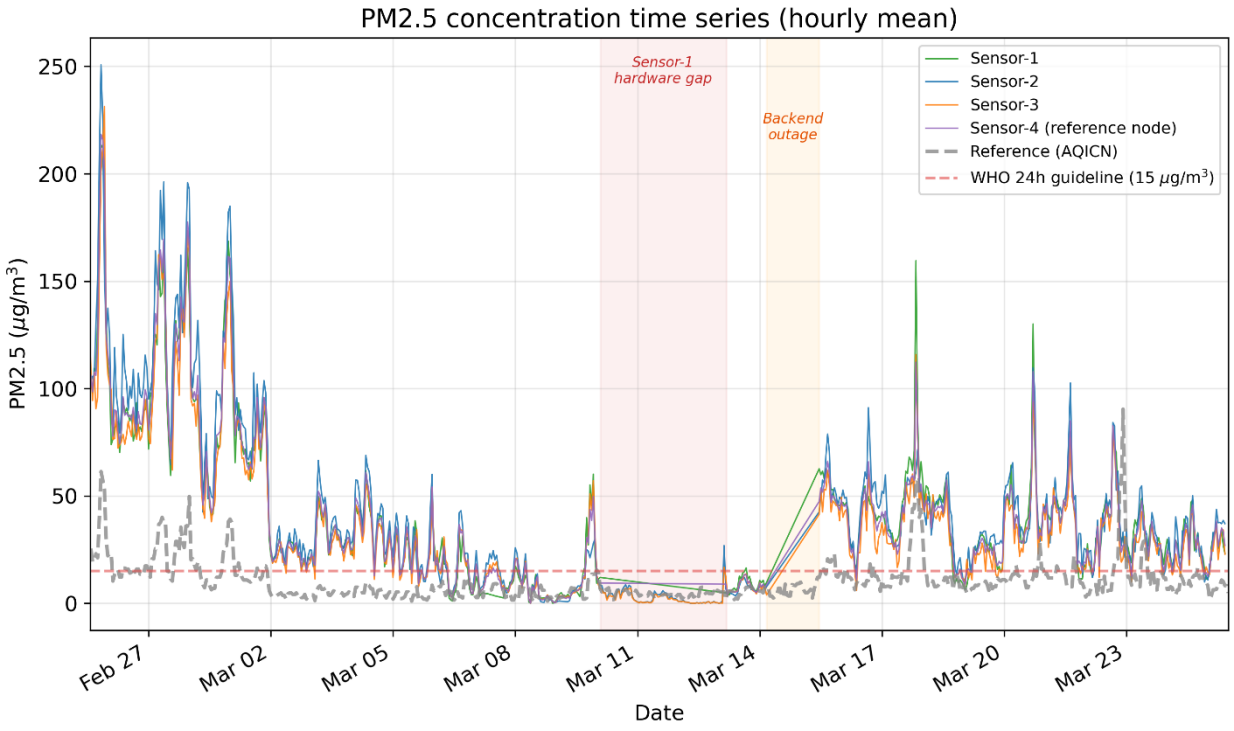


***Figure 4.2: Time series of raw PM2.5 measurements from all three IoT sensor nodes with Kazhydromet-14 reference station data overlay.***

An analysis of the average daily values of PM2.5 showed that the WHO recommended daily guideline (15 μg/m³) [1] was exceeded in 78% of the observation days for Sensor-1 (21 out of 27 days), in 72% of cases for Sensor-2 (21 out of 29 days), and 72% for Sensor-3 (21 out of 29 days), compared to 24% for the reference station (7 out of 29 days). The difference in WHO exceedance rates between the IoT sensors and the reference station reflects two factors: the systematic overestimation of PM2.5 by the PMS5003 sensors, which is consistent with the factory calibration error reported in the literature [15][20][21][38], and a genuine spatial deviation from the norm. variability of PM2.5 concentrations in different districts of Astana. This finding confirms the need for machine learning calibration, which is discussed in Section 4.3.

Correlation analysis between sensors has demonstrated high consistency between the three main IoT-based sensor nodes. The Pearson correlation coefficients between hourly PM2.5 readings were r = 0.96 (Sensor-1 vs Sensor-2, n = 566), r = 0.98 (Sensor-1 vs Sensor-3, n = 566) and r = 0.97 (Sensor-2 vs Sensor-3, n = 639). These strong correlations confirm that the three PMS5003 sensors exhibit consistent measurement behavior and that the observed PM2.5 characteristics are due to atmospheric conditions rather than noise from individual sensors [6][34]. The correlation between each IoT sensor and the control station was moderate: r = 0.69 (Sensor-1), r = 0.66 (Sensor-2), and r= 0.70 (Sensor-3), reflecting both the spatial distance between the sensors and the control station (~6 km) and the systematic overestimation error. PMS5003 [21][38].

Three evaluation indicators are used to evaluate the effectiveness of the model. The coefficient of determination ($R^2$) measures the fraction of deviation in the reference values of PM2.5, which are explained by the model, where $R^2 = 1$ indicates a perfect match. The root mean square error (RMSE) quantifies the average magnitude of forecasting errors in μg/m3. The average absolute error (MAE) is a linear measure of the average error. According to the efficiency criteria of the United States Environmental Protection Agency (EPA), low-cost calibration of sensors requires $R^2 \geq 0.70$ and RMSE $\leq 7$ μg/m³ [21][40].

## 4.2. Sensor Nodes Stability

During the 28-day data collection period, the designed IoT-based sensor nodes demonstrated stable operation in real outdoor conditions, including temperatures up to -26.8 °C. The software worked correctly in the deep sleep mode, which is described in Section 3.2.1, operating the duty cycle at 85%, and there was no UART desynchronization or sensor communication failures during normal operation. The PMS5003 sensor provided sequential measurements across all particulate channels. The AHT10 humidity sensor recorded smooth and physically realistic relative humidity fluctuations. However, during the data collection period, three data collection issues were found, which are described below.

The first data breach occurred between March 10 and March 13, 2026, when Sensor-1 was shut down due to a hardware problem and required a manual restart. During this three-day period, Sensor-2 and Sensor-3 continued to operate without interruption, demonstrating the benefits of a multi-node monitoring network in terms of sustainability. This event is the main reason for the lower completeness of Sensor-1 data (84.5%) compared to Sensor-2 (95.6%) and Sensor-3 (95.4%). The occurrence of such hardware failures is due to the fact that IoT-based sensor nodes often experience data loss due to hardware failures and environmental conditions [18].

The second data transmission failure occurred between March 14 and March 15, 2026, when the cloud server became unavailable due to the expiration of the cloud hosting subscription. During this period, all three sensor nodes continued to work and tried to transmit data, but the server did not accept incoming HTTP requests. The sensor nodes were unable to store all unsent record during this 2-day extended outage due to the limited size of the RTC memory buffer on the ESP32. This demonstrates the well-known infrastructure vulnerability of IoT-based monitoring systems as dependence on external cloud infrastructure creates a single point of failure in the data collection pipeline [23]. For future deployments, expanding the RTC memory buffer or implementing SD card-based local storage will ensure resilience to temporary server unavailability.

The third data completeness issue was related to BMP280 on Sensor-1 node, which stopped recording valid atmospheric pressure measurements after March 15. However, the temperature due to backup version AHT10, relative humidity and PM values continued to record stable. The atmospheric pressure values from the OpenMeteo API for the corresponding coordinates and timestamps of Sensor-1 are used to restore these missing measurements. The BMP280 sensors on the other sensor nodes operated correctly without failures during the entire data collection period. Figure 4.3 shows an email notification sent when BMP280 failure on Sensor-1 was detected.

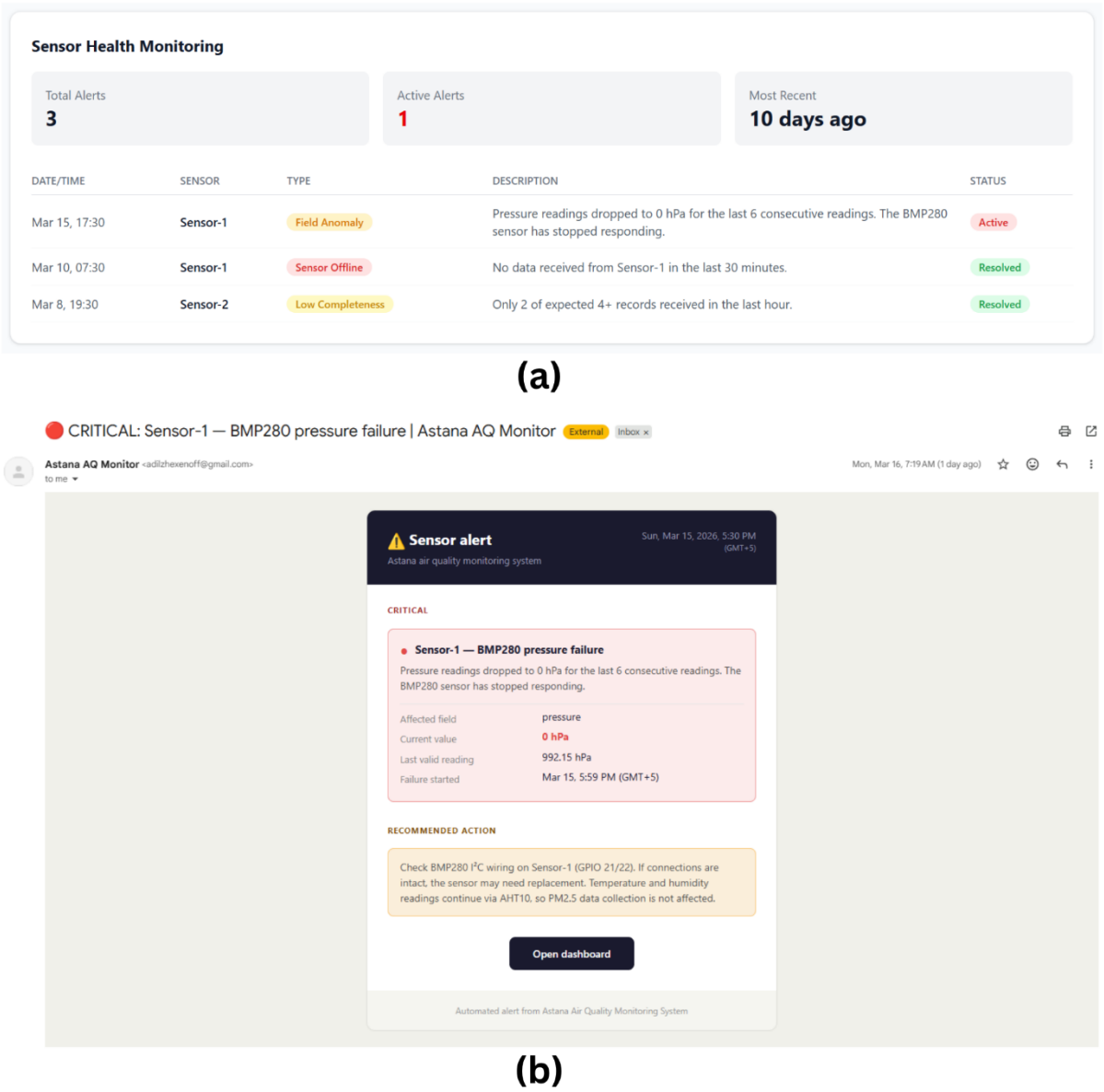


***Figure 4.3: Automated sensor health monitoring: (a) alert history on the web dashboard; (b) example of email notification.***

## 4.3. Calibration Model Comparison

The calibration model selection process was performed based on the paired dataset, which is described in Section 3.1.4. The raw PM2.5 measurements from the co-located Sensor-4 node showed a systematic overestimation compared to the PM2.5 measurements from the reference station with an average ratio of approximately 4:1, based on mean PM2.5 concentrations of 41.4 µg/m³ for Sensor-4 and 11.4 µg/m³ for the reference station. This overestimation is due to the factory calibration errors of the PMS5003 sensor as described in the literature, where overestimation coefficients were observed several times depending on the aerosol composition, relative humidity and temperature conditions [15][20][21][38]. Table 4.3 demonstrates results of cross-validation for all nine calibration models, which are described in Section 3.4.2.

**Table 4.3: Calibration model comparison results.**

| Model | $R^2$ (mean±std) | RMSE (µg/m³) | MAE (µg/m³) | Time (sec) |
|---|---|---|---|---|
| RF | 0.839±0.033 | 3.80 | 2.24 | 0.7 |
| XGBoost | 0.795±0.102 | 4.17 | 2.33 | 0.3 |
| KNN | 0.788±0.054 | 4.34 | 2.64 | 1.0 |
| ANN | 0.780±0.031 | 4.47 | 2.73 | 0.1 |
| MLR | 0.743±0.023 | 4.83 | 3.16 | 0.1 |
| DTR | 0.737±0.141 | 4.68 | 2.67 | 0.1 |
| SVR | 0.515±0.068 | 6.65 | 3.44 | 0.1 |
| BiLSTM | −0.798±0.993 | 6.28 | 4.46 | 6.5 |
| LSTM | −0.817±0.775 | 6.75 | 4.84 | 4.6 |

Figure 4.4 shows the $R^2$ values for cross-validation from Table 4.3, highlighting ML models that have passed the EPA efficiency threshold of $R^2 \geq 0.70$ [21][40]. This barchart demonstrates a clear division into three groups of models:

1) high-performance models (RF, XGBoost, KNN, ANN, MLR, DTR), which are exceeded the EPA efficiency threshold;
2) moderate-performance model (SVR), which is below the EPA efficiency threshold;
3) deep learning models (LSTM, BiLSTM), which are showed negative $R^2$ values due to small dataset size.

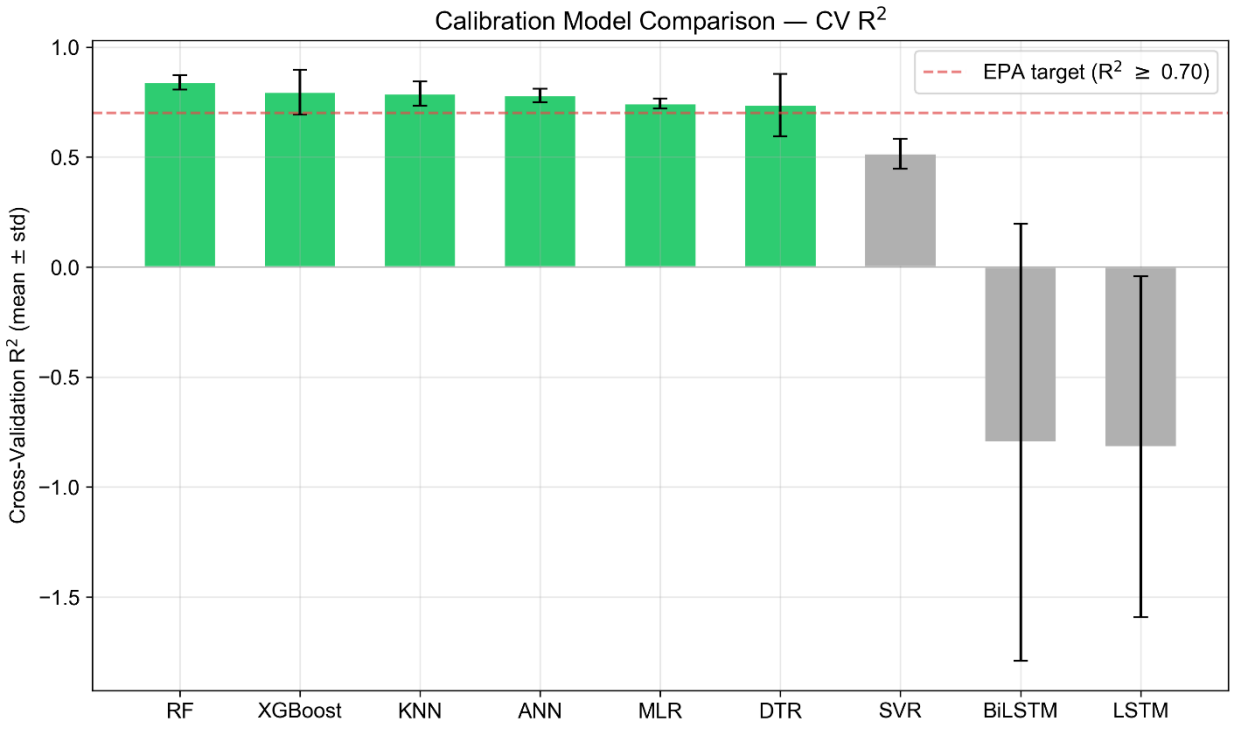


***Figure 4.4: Bar chart comparing cross-validation $R^2$ values across all nine calibration models.***

RF is the best-performing calibration model, which is achieved the highest $R^2$ of 0.839 ± 0.033 and the lowest RMSE of 3.80 µg/m³ and satisfied EPA efficiency threshold for low-cost sensor calibration [21][40]. RF showed higher efficiency than MLR on the paired co-located dataset, confirming that the PMS5003 sensor bias is nonlinear in real conditions in Astana [30][44][49]. MLR can be adjusted using simple regression equations like Barkjohn formula [21], while RF captures complex nonlinear characteristics [20][40][44].

Deep learning models LSTM and BiLSTM showed negative $R^2$ values (-0.82 and -0.80, respectively), indicating lower calibration efficiency than the predicted mean value due to the limited dataset size of 563 paired hourly measurements, which are aggregated from Sensor-4. This size is lower than the 1,000 paired hourly measurements, which is recommended as the practical minimum size for reliable low-cost sensor calibration [42]. The high cross-validation variance, which ranges from 0.77 and 0.99, confirms the instability of these DL models on small training datasets [50].

Figure 4.5 shows the visualization on the web-based dashboard of the raw and calibrated PM2.5 measurements for Sensor-1 over a 7-day period. The raw PM2.5 values (blue dashed line) are greater than both the calibrated PM2.5 values (blue solid line) and the reference station values (gray dashed line), while the calibrated values correspond more closely to the reference station. The red dashed horizontal line indicates the WHO 24-hour guideline (15 µg/m³) [1].

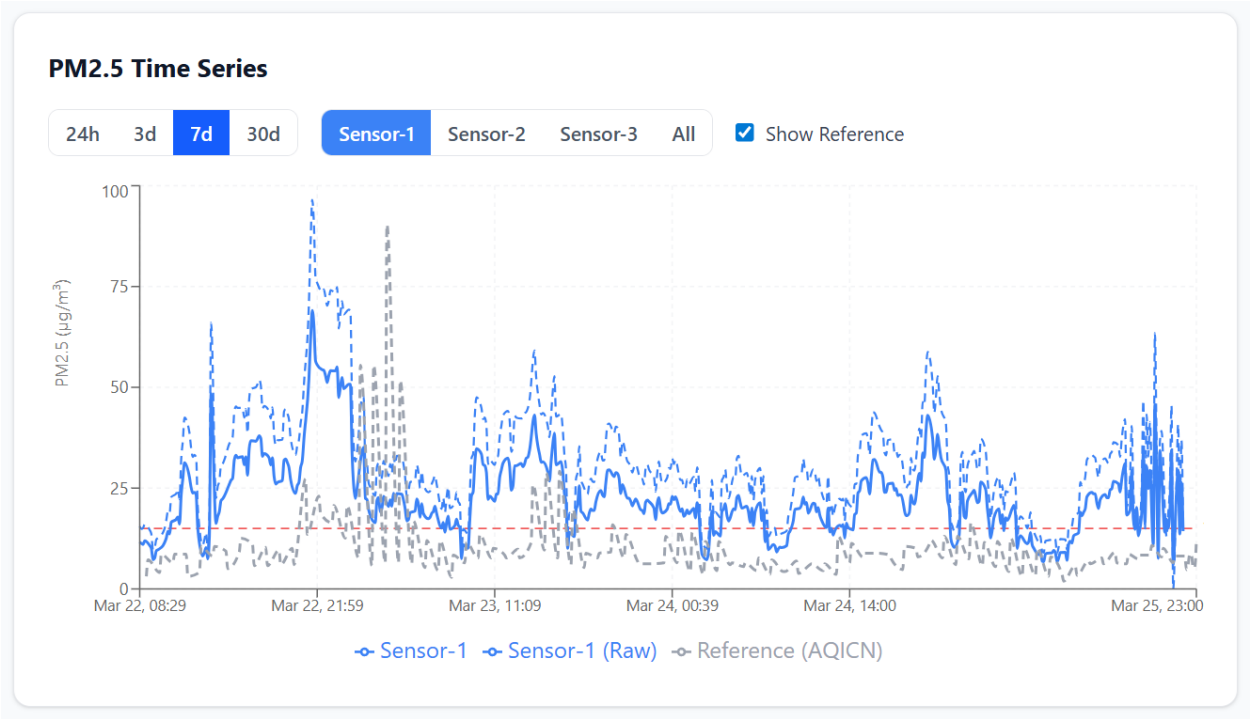


***Figure 4.5: The raw and calibrated PM2.5 measurements for Sensor-1 with the Kazhydromet-14 reference station overlay and WHO daily guideline on the web-based dashboard.***

### 4.3.1. Analysis of Feature Importance

The feature importance analysis based on the RF model provides a quantification of the contribution of each input variable to the calibration efficiency. Figure 4.6 demonstrates the corresponding feature importance estimates based on the trained RF model, with an emphasis on the variables with the greatest predictive relevance.

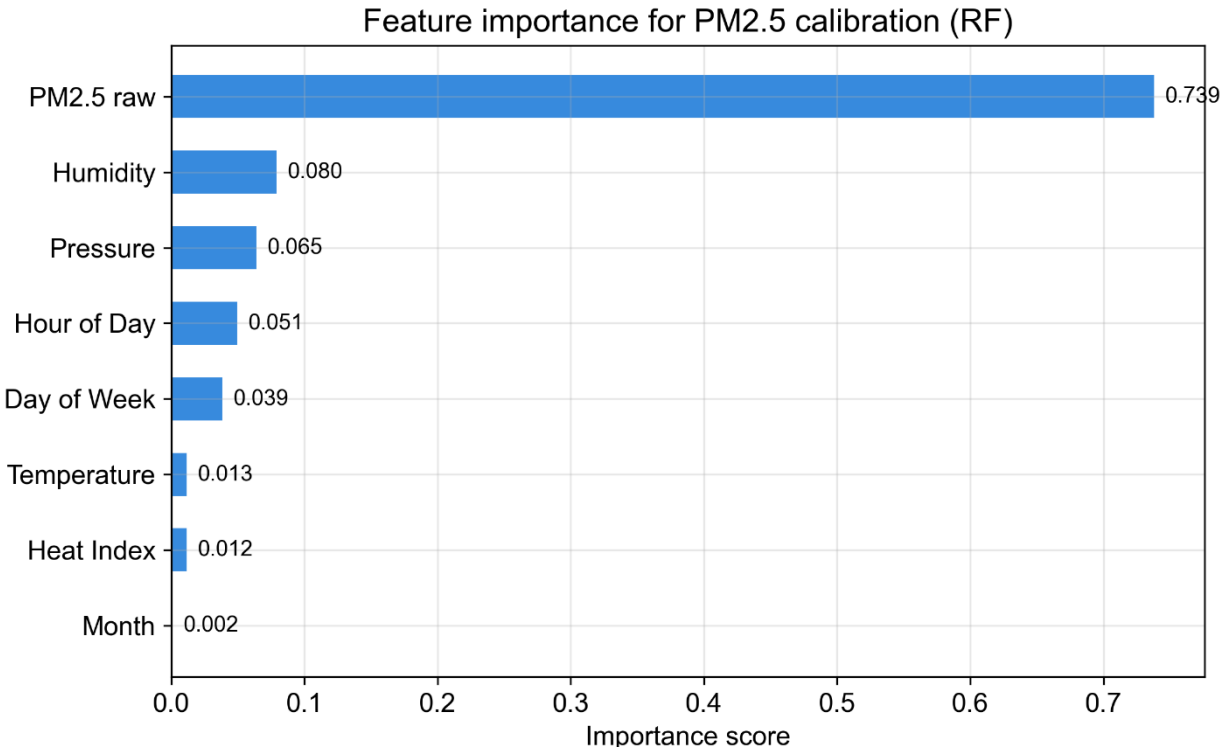


***Figure 4.6: Feature importance scores from the Random Forest calibration model.***

The most dominant predictor is the raw PM2.5 concentration, which is expected since the calibration task is to adjust the sensor output. Relative humidity (RH) is the second most important feature (0.08), which follows the literature, in which RH is defined as the main source of measurement error due to the hygroscopic particle growth, accounting for up to 30% of PM2.5 variance [40]. The temporal features like month (0.002) and day of week (0.039) demonstrated the minimum importance scores, since the observation period covers only 28 days, and therefore does not consider seasonal variability. Mathieu-Campbell et al. [20] also identified RH as the most important bias correction parameter for PMS5003 sensors, which is confirmed by this analysis.

### 4.3.2. Calibration Coefficient Stability Analysis

The 28-day paired dataset is divided into four weekly intervals to investigate the temporal stability of calibration parameters and determine the relevant model update frequency, answering Research Question 2. The most efficient Random Forest (RF) model and Multiple Linear Regression (MLR) were trained for each weekly window independently. Table 4.4 shows the MLR calibration coefficients and model performance indicators for each weekly period.

**Table 4.4: Age-based calibration coefficients (MLR) and Random Forest performance during four weekly windows.**

| **Metric** | **Week 1 (Feb 25 – Mar 3)** | **Week 2 (Mar 4–10)** | **Week 3 (Mar 11–17)** | **Week 4 (Mar 18–25)** |
|---|---|---|---|---|
| Number of records | 175 | 174 | 39 | 175 |
| Mean temp (°C) | -7.7 | -1.9 | 0.4 | -1.0 |
| Mean PM2.5 ref (µg/m³) | 15.8 | 4.9 | 7.2 | 13.0 |
| MLR Intercept | 15.83 | 4.9 | 7.24 | 13.04 |
| PM2.5 raw coefficient | 13.35 | 0.9 | 0.42 | 2.96 |
| Humidity coefficient | -0.86 | 1.08 | -2.76 | -1.94 |
| Temperature coefficient | -8.30 | -5.52 | -115.59 | -59.39 |
| MLR $R^2$ | 0.951 | 0.221 | 0.714 | 0.211 |
| RF $R^2$ | 0.991 | 0.927 | 0.946 | 0.952 |
| MLR RMSE (µg/m³) | 2.62 | 2.08 | 2.26 | 9.08 |
| RF RMSE (µg/m³) | 1.11 | 0.64 | 0.98 | 2.25 |

This age-based calibration analysis highlights two important findings. First, the RF model maintains consistently high performance within all four weeks, showing $R^2$ in the range from 0.93 to 0.99. RF is stable to changing environmental conditions. In contrast, the MLR model shows that $R^2$ decreased from 0.95 in Week 1 to 0.22 in Week 2. Such instability confirms that linear models like MLR cannot adequately reflect sensor bias when environmental conditions change between weekly intervals, while the nonlinear RF model adapts successfully.

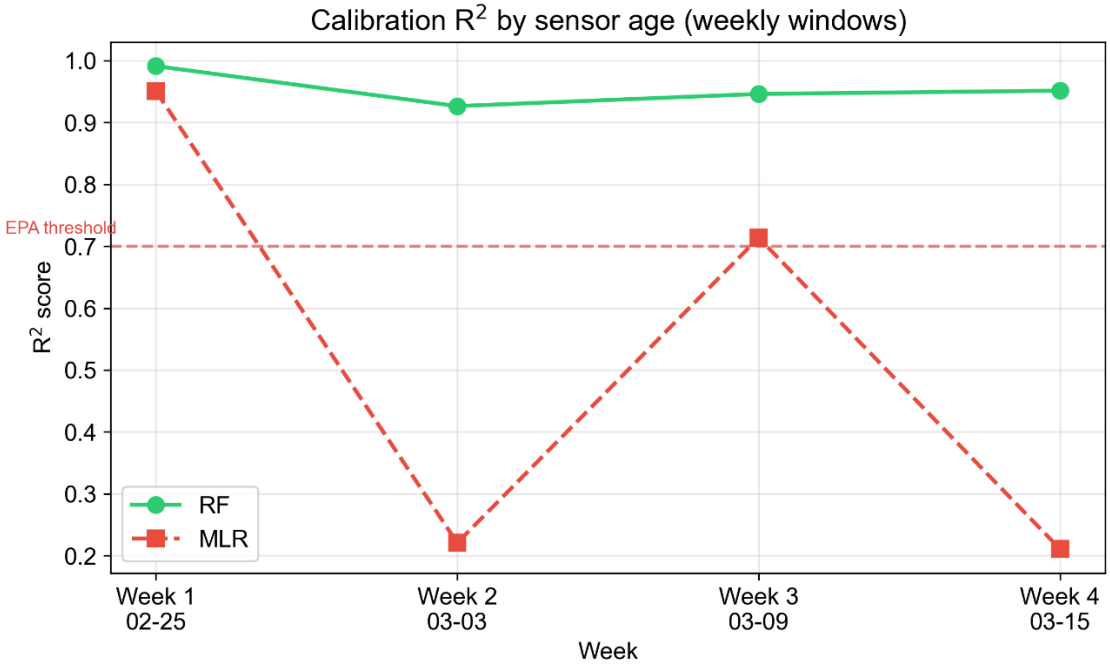


***Figure 4.7: Calibration model performance within four weekly windows for RF and MLR.***

Second, the MLR calibration coefficients show a systematic drift over weekly windows, decreasing the raw PM2.5 values from 13.35 in Week 1 to 0.42 in Week 3, as shown in Figure 4.8. This sharp change in values reflects a change in the ratio between sensor readings and the reference station as PM2.5 concentrations decrease from heating season levels (mean 15.8 µg/m³) to lower spring values (mean 4.9–7.2 µg/m³). The temperature coefficient demonstrates extreme values of -115.59 and -59.39 in the third and fourth weeks, respectively, indicating multicollinearity in the linear model with a small dataset and during periods of rapid environmental changes. Such changes in coefficients confirm that static calibration parameters become unreliable after 1-2 weeks in conditions of seasonal transitions.

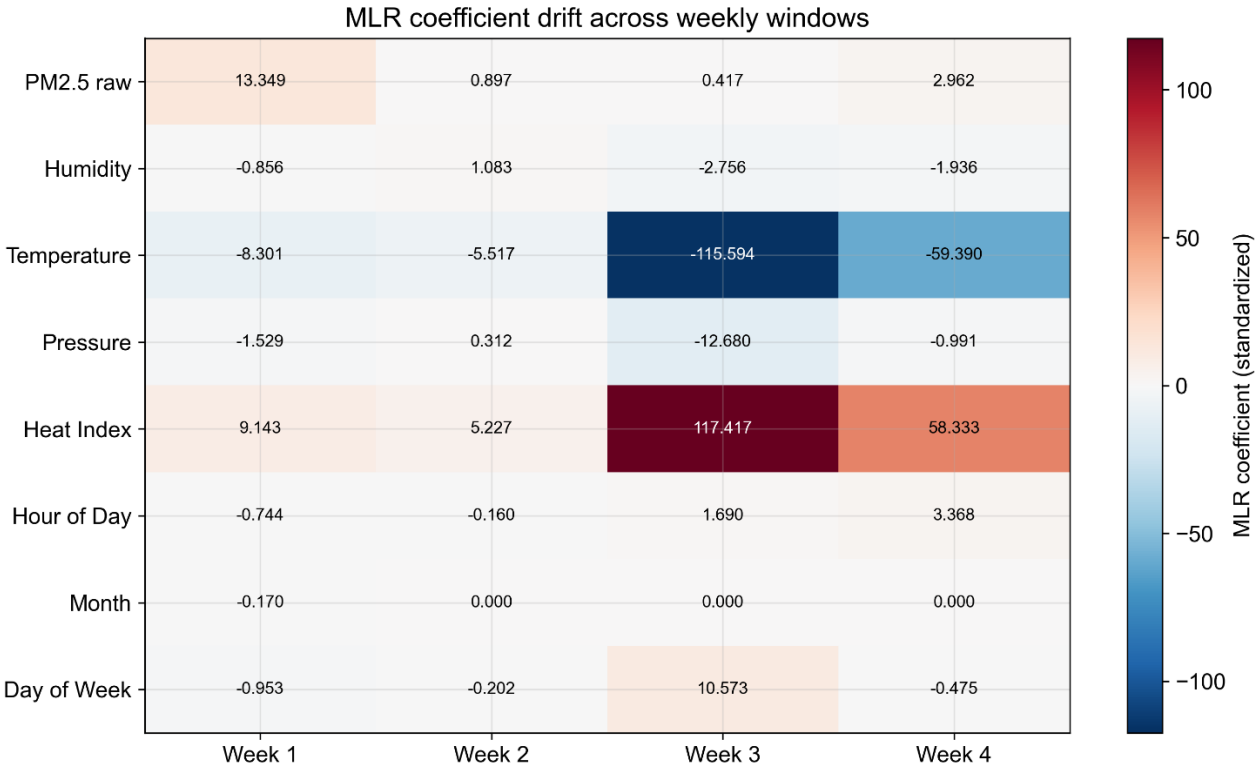


***Figure 4.8: MLR calibration coefficient drift across four weekly deployment windows.***

This temporal instability confirms the findings of Campmier et al. [38], who demonstrated that calibration models trained over 4-week periods showed that the normalized RMSE exceeds 100% when applied in different seasons in India (see Figure 2.7). In addition, this variability of correction coefficients confirms that 44% of the deviations in the calibration coefficient are due to the measurement date, rather than differences between sensor nodes [45]. As the minimum update frequency, weekly retraining is recommended based on these observations to ensure that the calibration model adapts to changing environmental conditions. This implemented monitoring system supports the proposed weekly calibration ML model retraining via cron job, which is described in Section 3.3. However, the 28-day observation period covers only the transition from the heating season to early spring, which limits the possibility of summarizing this observation within a single seasonal transition. A multi-season IoT-based sensor network deployment dataset is required to comprehensively determine the optimal update frequency, especially under different climatic and pollution conditions across different regions.

### 4.3.3. Comparison with State-of-the-Art

Table 4.5 shows the comparison between the calibration results of this study and the published results of PMS5003 calibration studies, which are conducted under different geographical and climatic conditions. RF $R^2$ value of 0.84 achieved in this study is within the reported range provided in comparable studies using a physical location and meets the EPA performance criterion.

**Table 4.5: Comparison of calibration results with published PMS5003 studies.**

| **Study** | **Location** | **Best Model** | **$R^2$** | **RMSE ($\mu g/m^3$)** |
|---|---|---|---|---|
| Chen (2025) [37] | Dublin | MLR | 0.81 | 2.3 (24-h) |
| Barkjohn (2021) [21] | US | MLR+RH | 0.65–0.94 | 3.0 (24-h) |
| Campmier (2023) [38] | India | RF | 0.86–0.94 | — |
| Nan (2025) [30] | China | XGBoost | 0.88 | — |
| Malyan (2024) [44] | Delhi | RF | 0.70–0.95 | — |
| This system | Astana | RF | 0.84 | 3.80 (1-h) |

This comparison creates several observations based on its results. First, the RF is the most efficient model for PMS5003 calibration and its $R^2$ value of 0.84 corresponds to the range as reported in the literature [38][44]. Second, only those studies, which were conducted in environments with more stable aerosol composition [21][37], where linear correction is sufficient, conclude that the MLR is the most efficient model for PMS5003 calibration. In Astana, a coal-dominated aerosol and high temperature variability create nonlinear bias patterns. Therefore, the nonlinear RF model outperforms MLR by about 10% in terms of $R^2$ (0.84 vs 0.74), which is similar to the performance gap observed in India [38] and Delhi [44].

## 4.4. Forecasting Model Comparison

The forecasting model comparison was performed based on a multi-station dataset, which includes 1,331 daily recordings from three reference stations in Astana (US Embassy, Kazhydromet-9, and Kazhydromet-14), with a total of 1,220 acceptable training sequences after feature construction as described in Section 3.4.4. Table 4.6 shows the results of cross-validation and final testing for all four models.

**Table 4.6: Forecasting model comparison results.**

| Model | $R^2$ (mean±std) | RMSE (μg/m³) | MAE (μg/m³) |
|---|---|---|---|
| LSTM | 0.232±0.128 | 11.62 | 3.08 |
| CNN-LSTM | 0.209±0.121 | 11.79 | 3.12 |
| BiLSTM | 0.204±0.124 | 11.83 | 3.05 |
| XGBoost | −0.346±0.242 | 15.95 | 4.93 |

Based on the selection criterion of the lowest mean RMSE across the cross-validation folds, LSTM was identified as the best-performing forecasting model with $R^2$=0.232 and RMSE = 11.62 μg/m³. CNN-LSTM and BiLSTM showed comparable performance, while XGBoost showed substantially worse results with a negative $R^2$ across cross-validation folds ($R^2$ = −0.346). The presence of built-in delay and shift functions that do not reflect temporal patterns creates a negative $R^2$ value for XGBoost, unlike learned representations in recurrent models. Traditional ML models such as XGBoost rely on explicitly created lag features and rolling statistics, while deep learning architectures learn temporal dependencies directly from raw data sequences [47][48]. Therefore, LSTM and similar recurrent architectures are well suited for PM2.5 forecasting due to their ability to capture long-term time dependencies [48][50].

The relatively low $R^2$ values for all four forecast models are expected over the 7-day forecast horizon, since PM2.5 concentrations show high variability and depend on weather events, heating patterns, and atmospheric dynamics, which become unpredictable over longer horizons [47]. Figure 4.9 illustrates the RMSE values for each horizon for all four models over a 7-day forecast period. LSTM has the best balance between short-term accuracy and long-term stability, with RMSE for each horizon ranging from 9.60 µg/m³ to 12.13 µg/m³. XGBoost shows stable high RMSE values (14.17-16.64 µg/m³) on all horizons, which confirms its unsuitability for the PM2.5 forecasting.

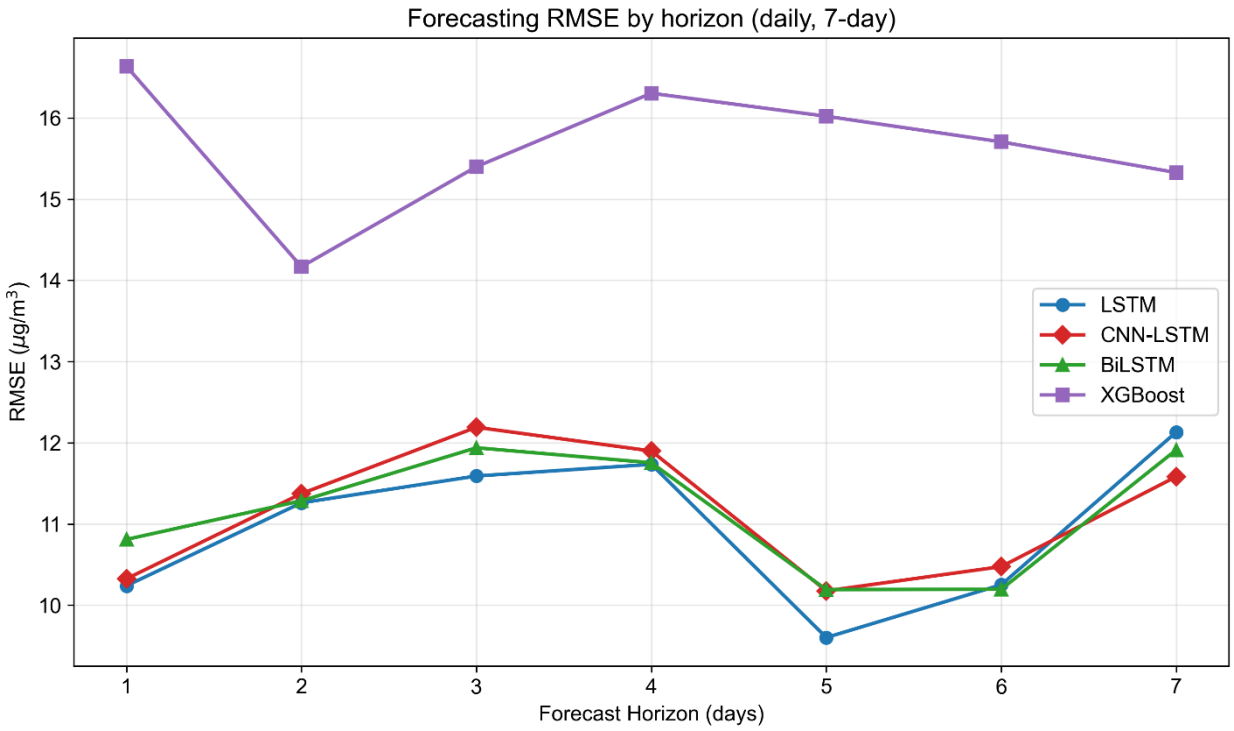


***Figure 4.9: Per-horizon RMSE for all forecasting models across the 7-day forecast period.***

Improving the forecasting accuracy is expected with additional training dataset. The current dataset of 1,331 daily observations is limited compared to multi-year datasets used in large-scale forecasting studies. The daily automated retraining of the forecasting ML model ensures that the LSTM model is updated with the latest dataset. This process is important during seasonal changes as described by the calibration analysis in Section 4.3.2.

## 4.5. System Demonstration

During the 28-day data collection period, the complete air pollution monitoring system was deployed and operated, demonstrating a full workflow from data collection by IoT-based sensor nodes with consideration of ML-based calibration and forecasting to real-time visualization in the web application. The map interface shows the spatial distribution of PM2.5 concentrations at four sensor locations in Astana in real time. Color-coded markers based on the Air Quality Index (AQI) provide a visual assessment of the air quality at each sensor installation point. The web dashboard during active operation of the system on March 25, 2026 illustrated in Figure 4.10.

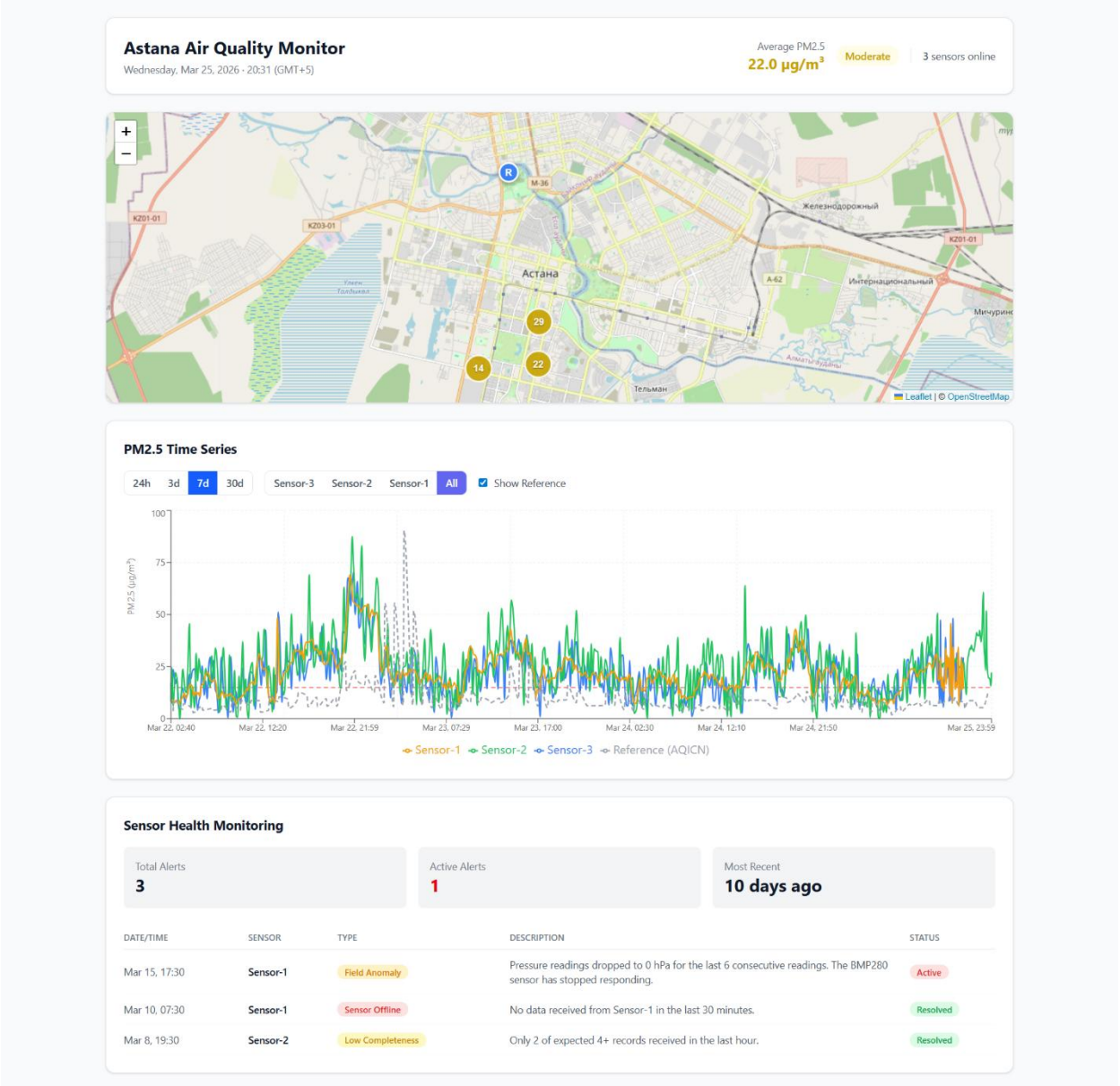


***Figure 4.10: Web-based dashboard of the air quality monitoring system.***

The time series panel provides a 28-day historical overview with a choice of time ranges (24 hours, 3 days, 7 days, 30 days), filtering by sensor nodes, and the option to show measurements from the Kazhydromet-14 reference station. The 30-day overview mode clearly highlights the transition from high air pollution period during the heating season (February – early March) to a lower air pollution level after snowfall and warming (mid-March onwards), which is consistent with the temporal patterns described in Section 4.1. The sensor health monitoring section at the bottom of the dashboard has a summary of detected warnings and a history of sensor failures, which allows the system administrator to view hardware failures, as shown in Section 4.2. The markers for each sensor node open the modal window, which displays the current raw and calibrated PM2.5 measurements, weather parameters (temperature, relative humidity, and atmospheric pressure), and 7-day PM2.5 forecasting as illustrated in Figure 4.11.

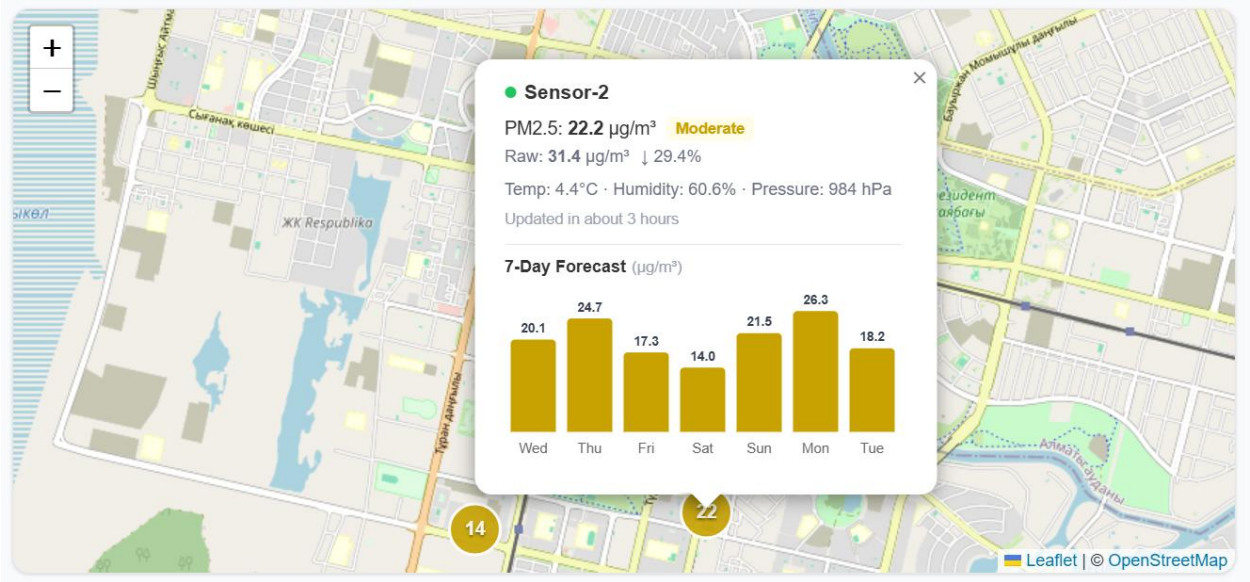


***Figure 4.11: Sensor-2 detail modal with calibrated PM2.5 measurements and 7-day forecast.***

The Sensor-2 modal window demonstrates the practical effectiveness of the ML pipeline. The raw PM2.5 value of 31.4 µg/m³ is reduced to 22.2 µg/m³ after Random Forest calibration, showing a 29.4% of reduction. The 7-day LSTM forecast provides predicted values ranging from 14.0 to 26.3 µg/m³ for the following week. This UI/UX approach allows the user to evaluate both the current air pollution level and the expected dynamics without moving to other pages.

# Chapter 5 – Conclusion

This thesis presented the design, implementation, and evaluation of an IoT-based air pollution monitoring system for real-time PM2.5 assessment in Astana. The proposed system integrates the designed sensor nodes based on the ESP32 microcontroller and the PMS5003 sensor, a cloud-based server-side application on Express.js with MongoDB Atlas, a machine learning pipeline for sensor co-location calibration and 7-day PM2.5 forecasting, sensor health monitoring system with email alerts, and a React-based web dashboard for real-time visualization.

Four sensor nodes were located in different locations in Astana, including co-located Sensor-4 with the Kazhydromet-14 reference station based on BAM-1020. During the 28-day data collection period at the end of the heating season (February–March 2026), 14,444 measurements were collected. The deployed sensor nodes demonstrated stable operation at temperatures down to -26.8 °C. The data completeness ranged from 84.3% to 95.6%, which corresponds to the range observed in real IoT-based monitoring systems [15]. The average PM2.5 concentration was 38.8 μg/m³, which exceeds the WHO daily recommendation level on 72-78% most days of observation. These findings confirm the air pollution issue in Astana during the heating season.

This chapter provides detailed answers to the three research questions, based on the experimental results from Sections 4.1–4.5. Research Question 1 explores which machine learning algorithms provide the highest calibration accuracy for low-cost IoT sensors. Random Forest (RF) showed the highest calibration accuracy among nine calibration models using paired dataset collected from co-located Sensor-4 and the Kazhydromet-14 reference station (see Table 4.3), showing a cross-validation $R^2$ of 0.839 ± 0.033 at RMSE 3.8 μg/m³. The calibration accuracy of RF satisfies the EPA performance criterion ($R^2 \geq 0.70$, RMSE ≤ 7 μg/m³) [21][40]. Using the RF model for calibration reduced the overestimation of PM values from the PMS5003 sensor, showing the average sensor node-to-reference ratio of approximately 4:1. This study demonstrates the practical effectiveness of the ML-based calibration pipeline under different extreme climate conditions, which are included in the 28-day observation period.

Research Question 2 calculates the minimum retrain frequency of ML models required to keep high calibration accuracy based on sensor-age analysis. The MLR calibration coefficients show a spread over weekly intervals as shown in Table 4.4. For example, the MLR performance degraded by 76.8% during one week (from $R^2 = 0.951$ to 0.221), while Random Forest keeps the calibration accuracy in a small interval ($R^2 = 0.927–0.991$) during the entire observation period. The difference in behavior during calibration between the tree-based ensemble models and linear models confirms that linear calibration coefficients become unreliable within 1–2 weeks during the transition between different seasons [45]. This proposed monitoring system recommends weekly retraining as the minimum retrain frequency, which is implemented by using the automated weekly cron job based on these observations. However, the current recommendation may change if the dataset is based on multi-season deployment at least one full year.

Research Question 3 considers the impact of seasonal variations on the accuracy and stability of low-cost IoT sensors. As the 28-day dataset includes the heating season and early spring in Astana, providing direct evidence of seasonal effects on both PM2.5 concentrations and sensor performance. The average PM2.5 value collected from the proposed IoT-based sensor network is 38.8 μg/m³, which exceeded the WHO 24-hour guideline (15 μg/m³) [1]. The time series shows a clear decrease in PM values from the end of February with peaks exceeding 200 μg/m³ during the heating season to mid-March with an average value below 50 μg/m³, which is consistent with a coal-burning decrease, as the environmental temperature increased from -26.8°C to +14°C during the heating season during the data collection period [10][14] as illustrated in Figure 4.2. The age-adjusted calibration analysis also demonstrates that the transition between seasons affects not only the PM2.5 concentration, but also the sensor bias characteristics. Since the MLR calibration coefficients change between the heating season and the spring transition weeks. These findings confirm the air pollution issue in Astana during the heating season and demonstrate the potential of low-cost IoT-based sensor networks to provide the spatial and temporal resolution, which is needed for evidence-based air quality management.

Some limitations inherent in the current study should be noted. The 28-day observation period covers only the transition from the heating season to early spring, which limits the possibility of generalizing calibration results and age-based results to a single seasonal transition. The shared location dataset, consisting of 563 paired hourly records, is below the recommended minimum of approximately 1,000 observations [42], which has caused the deep learning calibration models to fail. The accuracy of the prediction model ($R^2 = 0.23$) is limited by a limited set of training data consisting of 1,331 daily recordings from three stations. The location at the balcony level (3rd-4th floors) and the vertical height difference between sensor–4 (2.5-3 m) and BAM–1020 (11-12 m) can lead to systematic differences in paired measurements. The failure of the BMP280 pressure sensor on Sensor-1 after March 15 required data recovery using the external OpenMeteo API, and using the RTC memory buffer with a single entry resulted in data loss during a 31-hour server downtime.

The following improvements are planned for further work. First, an expanded co-location installation is needed, covering at least one full year to assess the effectiveness of calibration in all four seasons in Astana, including the summer period with a predominance of aerosol dust and the beginning of the autumn heating season. This dataset will provide a sufficient amount of training data for calibration models of deep learning over several seasons and will comprehensively determine the optimal frequency of retraining. Secondly, the forecasting system will be improved by including weather forecast data as external input and expanding the training dataset with multi-year historical records. Third, the sensor hardware will be improved through the introduction of local storage based on SD cards to prevent data loss during server failures and optimize the GPS data collection strategy for mobile deployment scenarios. Fourth, the monitoring network can be expanded with additional sensor nodes to increase spatial coverage around Astana, using the modular system architecture, which supports the addition of new nodes without making changes to the server side application or the ML pipeline.

# References


[1] WHO, "WHO global air quality guidelines: particulate matter (PM2.5 and PM10), ozone, nitrogen dioxide, sulfur dioxide and carbon monoxide," www.who.int, Sep. 22, 2021. https://www.who.int/publications/i/item/9789240034228

[2] "World's most polluted cities in 2023 - PM2.5 ranking | IQAIR." https://www.iqair.com/world-most-polluted-cities

[3] R. J. Isaifan, "Air pollution burden of disease over highly populated states in the Middle East," *Frontiers in Public Health*, vol. 10, Jan. 2023, doi: 10.3389/fpubh.2022.1002707.

[4] Y. Yang, "IoT-based air pollution monitoring system," *Highlights in Science Engineering and Technology*, vol. 17, pp. 299–307, Nov. 2022, doi: 10.54097/hset.v17i.2619.

[5] F. Concas *et al.*, "Low-Cost outdoor air quality monitoring and sensor calibration," *ACM Transactions on Sensor Networks*, vol. 17, no. 2, pp. 1–44, May 2021, doi: 10.1145/3446005.

[6] S. Diez *et al.*, "Long-term evaluation of commercial air quality sensors: an overview from the QUANT (Quantification of Utility of Atmospheric Network Technologies) study," *Atmospheric Measurement Techniques*, vol. 17, no. 12, pp. 3809–3827, Jun. 2024, doi: 10.5194/amt-17-3809-2024.

[7] A. Omarova *et al.*, "Emerging threats in Central Asia: Comparative characterization of organic and elemental carbon in ambient PM2.5 in urban cities of Kazakhstan," *Chemosphere*, vol. 370, p. 143968, Dec. 2024, doi: 10.1016/j.chemosphere.2024.143968.

[8] R. Mukhtarov *et al.*, "An episode-based assessment for the adverse effects of air mass trajectories on PM2.5 levels in Astana and Almaty, Kazakhstan," *Urban Climate*, vol. 49, p. 101541, Apr. 2023, doi: 10.1016/j.uclim.2023.101541.

[9] N. Temirbekov, S. Kasenov, G. Berkinbayev, A. Temirbekov, D. Tamabay, and M. Temirbekova, "Analysis of data on air pollutants in the city by Machine-Intelligent Methods considering climatic and geographical features," *Atmosphere*, vol. 14, no. 5, p. 892, May 2023, doi: 10.3390/atmos14050892.

[10] K. Tursun *et al.*, "Dominant sources of PM2.5 in Kazakhstan's urban cities: A PMF and HYSPLIT-based study for air quality management in Central Asia," *Urban Climate*, vol. 64, Art. no. 102706, Nov. 2025, doi: 10.1016/j.uclim.2025.102706.

[11] A. Muratuly, R. Mukhtarov, I. Radelyuk, F. Karaca, and N. Baimatova, "Urban PM 2.5 pollution in Kazakhstan: health burden and economic costs," *Environmental Science Advances*, vol. 5, no. 1, pp. 281–291, Dec. 2025, doi: 10.1039/d5va00194c.

[12] Z. Sarsenova, D. Yedilkhan, A. Yermekov, S. Saleshova, and B. Amirgaliyev, "Analysis and assessment of air quality in Astana: Comparison of pollutant levels and their impact on health," *Scientific Journal of Astana IT University*, vol. 19, pp. 98–117, Sep. 2024, doi: 10.37943/19szfa3931.

[13] D. Assanov, V. Zapasnyi, and A. Kerimray, "Air Quality and Industrial Emissions in the Cities of Kazakhstan," *Atmosphere*, vol. 12, no. 3, p. 314, Feb. 2021, doi: 10.3390/atmos12030314.

[14] A. Agibayeva, R. Khalikhan, M. Guney, F. Karaca, A. Torezhan, and E. Avcu, "An Air Quality Modeling and Disability-Adjusted Life Years (DALY) Risk Assessment Case Study: Comparing Statistical and Machine Learning Approaches for PM2.5 Forecasting," *Sustainability*, vol. 14, no. 24, p. 16641, Dec. 2022, doi: 10.3390/su142416641.

[15] M. Chen, W. Yuan, C. Cao, C. Buehler, D. R. Gentner, and X. Lee, "Development and performance evaluation of a Low-Cost Portable PM2.5 monitor for mobile deployment," *Sensors*, vol. 22, no. 7, p. 2767, Apr. 2022, doi: 10.3390/s22072767.

[16] M. J. Divan, M. L. Sanchez-Reynoso, J. E. Panebianco, and M. J. Mendez, "IoT-Based approaches for monitoring the particulate matter and its impact on health," *IEEE Internet of Things Journal*, vol. 8, no. 15, pp. 11983–12003, Aug. 2021, doi: 10.1109/jiot.2021.3068898.

[17] A. Ramachandran, K. Gayathri, R. H. Ramachandran, and Z. Beevi, “Modeling of internet of things enabled sustainable environment air pollution monitoring system,” *Global NEST Journal*, vol. 25, no. 4, pp. 172–179, Jan. 2023, doi: 10.30955/gnj.004707.

[18] G. Mekuria, “Air Pollution: A review of its impacts on health and ecosystems, and analytical techniques for their measurement and modeling,” *Journal of Environmental Informatics Letters*, vol. 10, no. 2, pp. 115–131, Nov. 2023, doi: 10.3808/jeil.202300116.

[19] O. Alsamrai, M. D. Redel-Macias, and M. P. Dorado, “Real-Time intelligent monitoring of outdoor air quality in an urban environment using IoT and machine learning algorithms,” *Applied Sciences*, vol. 15, no. 16, p. 9088, Aug. 2025, doi: 10.3390/app15169088.

[20] M. E. Mathieu-Campbell, C. Guo, A. P. Grieshop, and J. Richmond-Bryant, “Calibration of PurpleAir low-cost particulate matter sensors: model development for air quality under high relative humidity conditions,” *Atmospheric Measurement Techniques*, vol. 17, no. 22, pp. 6735–6749, Nov. 2024, doi: 10.5194/amt-17-6735-2024.

[21] K. K. Barkjohn, B. Gantt, and A. L. Clements, “Development and application of a United States-wide correction for PM 2.5 data collected with the PurpleAir sensor,” *Atmospheric Measurement Techniques*, vol. 14, no. 6, pp. 4617–4637, Jun. 2021, doi: 10.5194/amt-14-4617-2021.

[22] D. A. Jaffe *et al.*, “An evaluation of the U.S. EPA’s correction equation for PurpleAir sensor data in smoke, dust, and wintertime urban pollution events,” *Atmospheric Measurement Techniques*, vol. 16, no. 5, pp. 1311–1322, Mar. 2023, doi: 10.5194/amt-16-1311-2023.

[23] C. Soemphol, T. Thongsan, S. Ninkaew, and P. Panmuang, “Design and implementation of a solar-powered IoT-based real-time air quality monitoring system,” *Bulletin of Electrical Engineering and Informatics*, vol. 14, no. 5, pp. 4150–4160, Oct. 2025, doi: 10.11591/eei.v14i5.10473.

[24] W. A. Jabbar, T. Subramaniam, A. E. Ong, M. I. Shu’Ib, W. Wu, and M. A. De Oliveira, “LORAWAN-Based IoT System Implementation for Long-Range Outdoor Air Quality Monitoring,” *Internet of Things*, vol. 19, Art. no. 100540, May 2022, doi: 10.1016/j.iot.2022.100540.

[25] P. Das, S. Ghosh, S. Chatterjee, and S. De, “A low cost outdoor air pollution monitoring device with power controlled Built-In PM sensor,” *IEEE Sensors Journal*, vol. 22, no. 13, pp. 13682–13695, May 2022, doi: 10.1109/jsen.2022.3175821.

[26] O. Yildiz and H. S. Sucuoglu, “Development of Real-Time IOT-Based air quality Forecasting System using Machine learning approach,” *Sustainability*, vol. 17, no. 19, p. 8531, Sep. 2025, doi: 10.3390/su17198531.

[27] A. Osa-Sanchez and B. Garcia-Zapirain, “Real-Time air quality monitoring: a smart IoT system using Low-Cost sensors and 3-D printing,” *IEEE Journal of Radio Frequency Identification*, vol. 9, pp. 65–79, Jan. 2025, doi: 10.1109/jrfid.2025.3541816.

[28] A. S. Moursi, N. El-Fishawy, S. Djahel, and M. A. Shouman, “An IoT enabled system for enhanced air quality monitoring and prediction on the edge,” *Complex & Intelligent Systems*, vol. 7, pp. 2923–2947, Jul. 2021, doi: 10.1007/s40747-021-00476-w.

[29] A. Parmar *et al.*, “Development of end-to-end low-cost IoT system for densely deployed PM monitoring network: an Indian case study,” *Frontiers in the Internet of Things*, vol. 3, Feb. 2024, doi: 10.3389/friot.2024.1332322.

[30] F. Nan, H. Shen, and C. Zeng, “An integrated low-cost air quality sensor and a multi-task calibration framework for particulate matter,” *Environment International*, vol. 207, Art. no. 109981, Dec. 2025, doi: 10.1016/j.envint.2025.109981.

[31] J. Rosa-Bilbao, F. S. Butt, D. Merkl, M. F. Wagner, J. Schäfer, and J. Boubeta-Puig, “IOT-Based indoor air quality Management System for intelligent education environments,” *IEEE Internet of Things Journal*, vol. 12, no. 11, pp. 18031–18041, Jun. 2025, doi: 10.1109/jiot.2025.3539886.

[32] F. M. J. Bulot *et al.*, “Long-term field comparison of multiple low-cost particulate matter sensors in an outdoor urban environment,” *Scientific Reports*, vol. 9, no. 1, p. 7497, May 2019, doi: 10.1038/s41598-019-43716-3.

[33] S. Uğuz, P. Kumar, S. Tiwari, Y. Chang, and X. Yang, “Field testing of Low-Cost PM sensors in animal production facilities,” *Tekirdağ Ziraat Fakültesi Dergisi*, vol. 22, no. 3, pp. 732–747, Sep. 2025, doi: 10.33462/jotaf.1555650.

[34] T. Sayahi, A. Butterfield, and K. E. Kelly, “Long-term field evaluation of the Plantower PMS low-cost particulate matter sensors,” *Environmental Pollution*, vol. 245, pp. 932–940, Nov. 2018, doi: 10.1016/j.envpol.2018.11.065.

[35] D. Apostolopoulos, G. Fouskas, and S. N. Pandis, “Field calibration of a Low-Cost air quality monitoring device in an urban background site using machine learning models,” *Atmosphere*, vol. 14, no. 2, p. 368, Feb. 2023, doi: 10.3390/atmos14020368.

[36] P. Kortoçi *et al.*, “Air pollution exposure monitoring using portable low-cost air quality sensors,” *Smart Health*, vol. 23, p. 100241, Nov. 2021, doi: 10.1016/j.smhl.2021.100241.

[37] J. Chen, Ó. González, D. O’Connor, L. Tallon, and F. Pilla, “Assessment of IoT low-cost sensor networks for long-term outdoor and indoor air quality monitoring: a case study in Dublin,” *Atmospheric Pollution Research*, vol. 16, Art. no. 102651, Jul. 2025, doi: 10.1016/j.apr.2025.102651.

[38] M. J. Campmier *et al.*, “Seasonally optimized calibrations improve low-cost sensor performance: long-term field evaluation of PurpleAir sensors in urban and rural India,” *Atmospheric Measurement Techniques*, vol. 16, no. 19, pp. 4357–4374, Oct. 2023, doi: 10.5194/amt-16-4357-2023.

[39] C. Malings *et al.*, “Fine particle mass monitoring with low-cost sensors: Corrections and long-term performance evaluation,” *Aerosol Science and Technology*, vol. 54, no. 2, pp. 160–174, May 2019, doi: 10.1080/02786826.2019.1623863.

[40] M. R. Giordano *et al.*, “From low-cost sensors to high-quality data: A summary of challenges and best practices for effectively calibrating low-cost particulate matter mass sensors,” *Journal of Aerosol Science*, vol. 158, Art. no. 105833, Jul. 2021, doi: 10.1016/j.jaerosci.2021.105833.

[41] M. L. Zamora, C. Buehler, A. Datta, D. R. Gentner, and K. Koehler, “Identifying optimal co-location calibration periods for low-cost sensors,” *Atmospheric Measurement Techniques*, vol. 16, no. 1, pp. 169–179, Jan. 2023, doi: 10.5194/amt-16-169-2023.

[42] L. Liang and J. Daniels, “What influences low-cost sensor data calibration? - A systematic assessment of algorithms, duration, and predictor selection,” *Aerosol and Air Quality Research*, vol. 22, no. 9, Art. no. 220076, Jun. 2022, doi: 10.4209/aaqr.220076.

[43] P. deSouza *et al.*, “An analysis of degradation in low-cost particulate matter sensors,” *Environmental Science Atmospheres*, vol. 3, no. 3, pp. 521–536, Jan. 2023, doi: 10.1039/d2ea00142j.

[44] V. Malyan *et al.*, “Assessing the spatial transferability of calibration models across a low-cost sensors network,” *Journal of Aerosol Science*, vol. 181, Art. no. 106437, Jul. 2024, doi: 10.1016/j.jaerosci.2024.106437.

[45] J. Tryner *et al.*, “Laboratory evaluation of low-cost PurpleAir PM monitors and in-field correction using co-located portable filter samplers,” *Atmospheric Environment*, vol. 220, Art. no. 117067, Oct. 2019, doi: 10.1016/j.atmosenv.2019.117067.

[46] A. Mazinani, D. Antonucci, D. Pietro Pau, L. Davoli, and G. Ferrari, “Air quality prediction via embedded ML/DL and quantized models,” *IEEE Access*, vol. 13, pp. 154203–154218, Jan. 2025, doi: 10.1109/access.2025.3603920.

[47] S. Al-Eidi, F. Amsaad, O. Darwish, Y. Tashtoush, A. Alqahtani, and N. Niveshitha, “Comparative Analysis Study for air quality prediction in smart cities using regression techniques,” *IEEE Access*, vol. 11, pp. 115140–115149, Oct. 2023, doi: 10.1109/access.2023.3323447.

[48] C. Bachechi, F. Rollo, and L. Po, “HypeAIR: A novel framework for real-time low-cost sensor calibration for air quality monitoring in smart cities,” *Ecological Informatics*, vol. 81, p. 102568, Mar. 2024, doi: 10.1016/j.ecoinf.2024.102568.

[49] X. Ma *et al.*, “A Machine Learning-Based calibration framework for Low-Cost PM2.5 sensors integrating meteorological predictors,” *Chemosensors*, vol. 13, no. 12, p. 425, Dec. 2025, doi: 10.3390/chemosensors13120425.

[50] K. Aula, E. Lagerspetz, P. Nurmi, and S. Tarkoma, “Evaluation of Low-Cost Air Quality Sensor Calibration Models,” *ACM Transactions on Sensor Networks*, Apr. 2022, doi: 10.1145/3512889.