FACULTY OF ELECTRICAL AND ELECTRONIC ENGINEERING TECHNOLOGY

BVI 3114 TECHNOLOGY SYSTEM OPTIMIZATION II

SEMESTER 5 2025/2026

Lecturer:
Prof Madya Ir. Ts. Dr. Mohd Zamri Bin Ibrahim

Report:
Mini Project: Real-Time IoT Sensor Data Forecasting System with AI-powered Analysis

Prepared By	MUHAMMAD ARIF BIN AMRAN (VC23004)
	MAS IZZALIANA BINTI MOHAMAD MOKHTAR (VC23035)

Table of Content

Executive Summary

This mini project display the design and implementation of Internet of Things (IoT) based environmental monitoring system using an ESP32 microcontroller integrated with cloud based services and artificial intelligence (AI). The project system was developed to continuously collect, process, generate automated AI-powered summaries and visualize in real time to provide users with insights. The project integrates BME280 sensor and HC-SR04 sensor to measure temperature, humidity, pressure and distance parameters.

The primary objective of this mini project was to develop a functional end to end IoT system capable of acquiring real time sensor data, wirelessly transmit it to Google Sheet, forecast, AI summary, visualize and presents in user friendly format. Next objectives included configure and interface multiple sensors with the ESP32, establish reliable WiFi communication, logs real time data to Google Sheets, forecasting techniques for prediction, integrating Google Gemini API for AI summarization and visualize through charts in Looker Studio dashboard. The mini project also aimed to apply basic data analysis techniques to identify trends and improve the interpretability of sensor readings through a responsive Google Looker Studio dashboard and a mobile application .

To achieve the objectives, system architecture was designed with the ESP32 as the controller, responsible for sensor data acquisition and HTTP communication with Google Apps Script web service. The received data is automatically stored in Google Sheets. Forecasting algorithms, including Exponential Moving Average (EMA) and Holt-Winters methods, are applied to generate various hours predictions. Google Gemini API generate real time data analysis, trend analysis, and predictive insight summaries based on both historical and forecasted data. Looker Studio then used to transform it into meaningful graphs, allowing users to observe patterns such as environmental fluctuations and distance variations over time.

The results display the system successfully achieved continuous and reliable real time data collection, accurate forecasting and AI generated summaries. The AI summarization feature effectively transformed complex sensor data into human readable insights. The Looker Studio dashboard and embedded mobile application provided clear visualization of sensor readings, forecasts and AI insights in a user friendly and mobile responsive format.

This mini project demonstrates the practical application of IoT systems enhanced with forecasting and AI function analysis. The successful implementation display how embedded systems, cloud platforms and generative AI can be combined to create an intelligent monitoring solution. The system also provides a strong foundation for future enhancements, such as improved forecasting accuracy, automated alerting and expanded AI driven decision support.

Timeline and Milestones

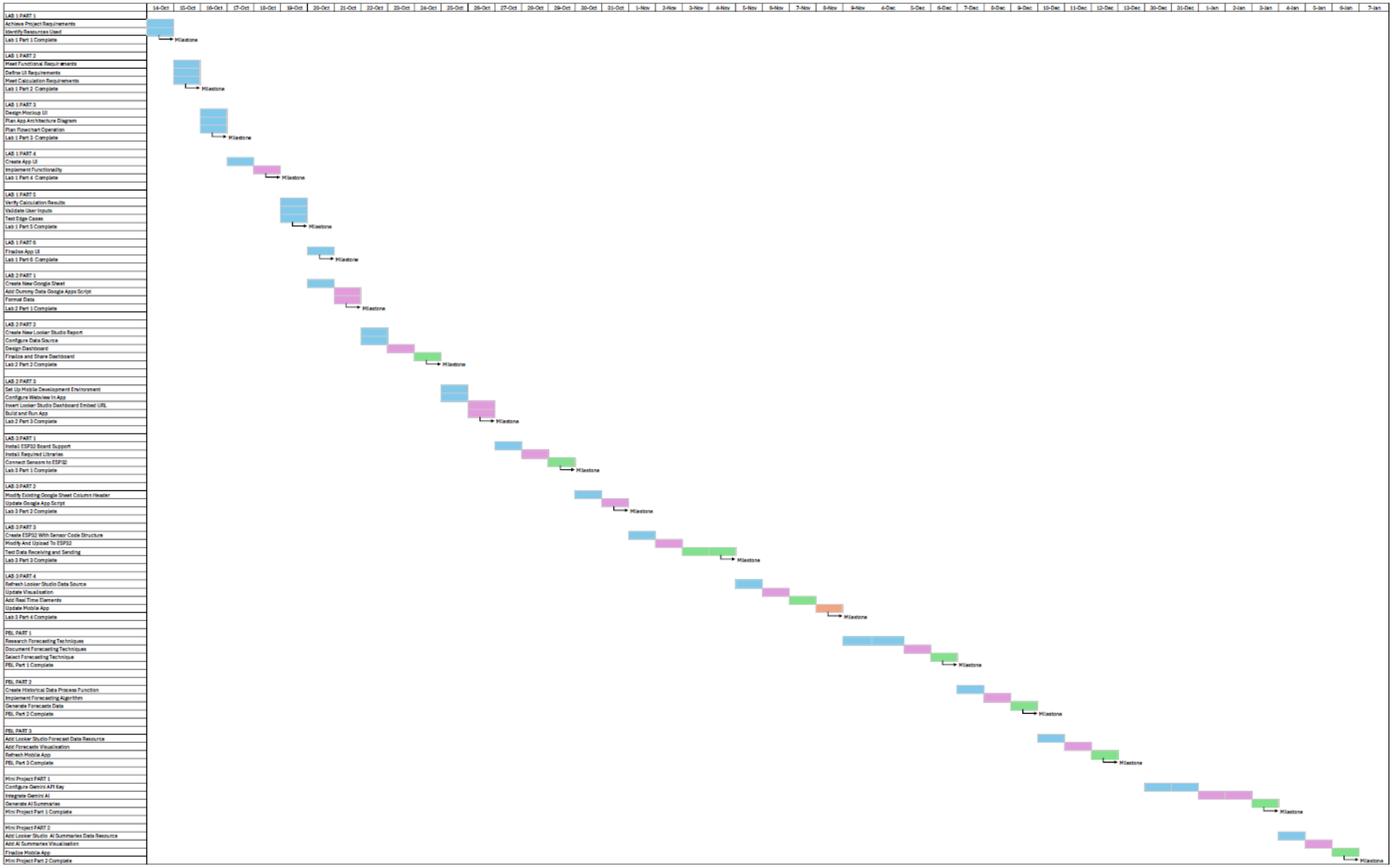


Figure 1: Timeline and Milestones.

System Architecture Diagram

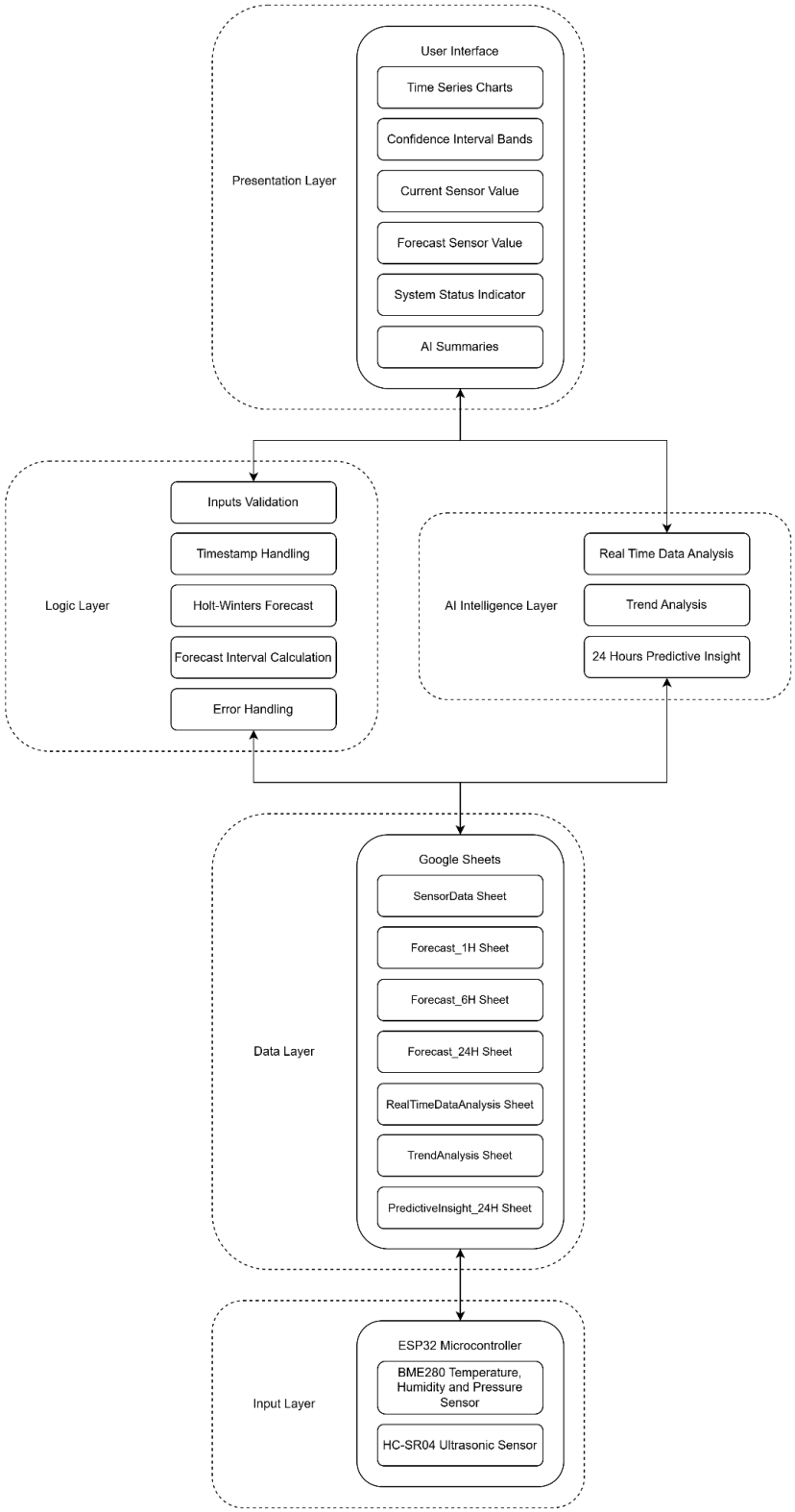


Figure 2: System Architecture Diagram.

Description of Hardware Setup

For this next lab exercise, the previous lab will be extended to replacing the dummy data with real sensor readings from the assigned ESP32 microcontroller, HC-SR04 Ultrasonic sensor and BME280 Temperature and Humidity Sensor.

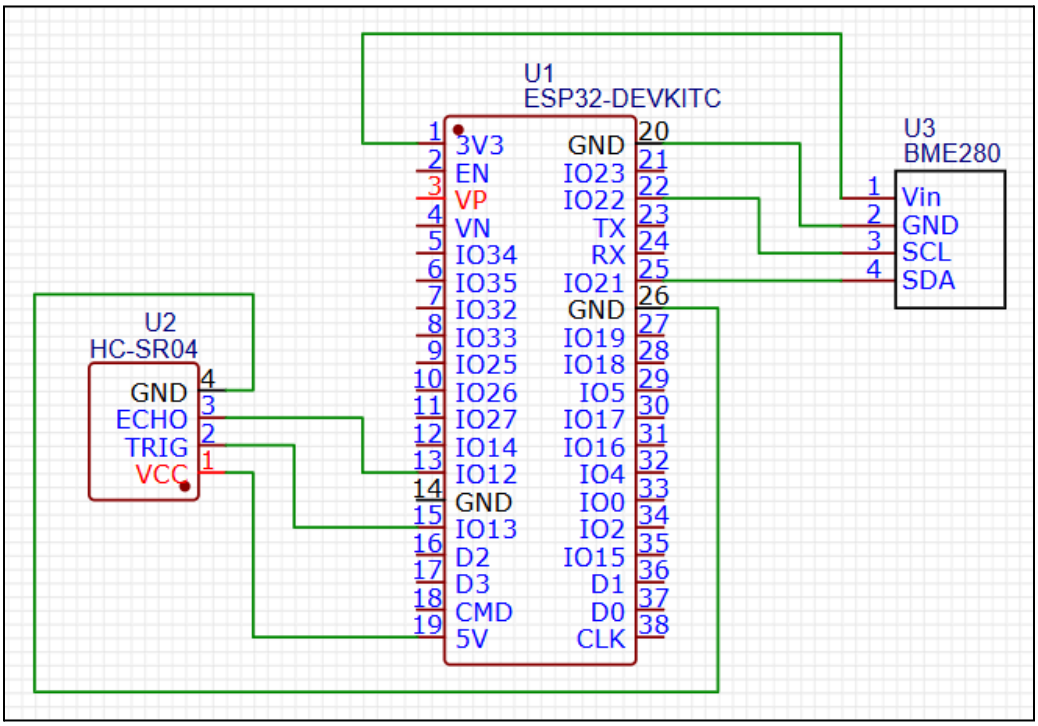


Figure 3: ESP32 Connection With HC-SR04 Ultrasonic sensor and BME280 Temperature and Humidity Sensor Schematic Diagram.

As shown in Figure 3, the wiring for ESP32 with BME280 Temperature and Humidity sensor and HC-SR04 Ultrasonic sensor are as follow;

a. BME280 Temperature and Humidity Sensor Wiring:
   - Vin Pin will connect to 3.3V ESP32.
   - GND Pin will connect to GND ESP32.
   - SCL Pin will connect to I2C Pin GPIO22 ESP32.
   - SDA Pin will connect to I2C Pin GPIO 21 ESP32.

b. HC-SR04 Ultrasonic Sensor Wiring:
   - VCC Pin will connect to 5V ESP32.
   - GND Pin will connect to GND ESP32.
   - ECHO Pin will connect to GPIO12 ESP32.
   - TRIG Pin will connect to GPIO13 ESP32.

Forecasting Algorithm Explanation

1. Exponential Moving Average (EMA)
   a. Basic Principles

      The Exponential Moving Average (EMA) is a time series smoothing technique that gives more weight to recent data while still considering older observations. The EMA reacts faster to sudden changes because the weighting decreases exponentially over time unlike the Simple Moving Average (SMA) which treats all data points equally.

      The EMA is calculated using the formula:

$$EMAt = \alpha \cdot xt + (1 - \alpha) \cdot EMAt - 1$$

      - xt is current measurement.
      - EMAt is new smoothed value
      - A is smoothing factor (0-1)

   b. Strengths
      - Fast reaction to new change because recent data is weighted more heavily.
      - Low computational cost and ideal for microcontrollers.
      - Good noise filtering, reducing random spikes in sensor readings.
      - Works well with streaming data.

   c. Limitations
      - Cannot detect trends or seasonality.
      - Heavily dependent on the alpha value.
      - Not suitable for long term forecasting.

   d. Specific Sensors Suitability

      Highly suitable for HC-SR04 Ultrasonic sensor because the distance readings often contain noise and sudden spikes so EMA will smooths distance measurements while react quick to small fluctuations.

2. Holt-Winters Method

a. Basic Principles

The Holt-Winters method or also known as Triple Exponential Smoothing is forecasting technique for seasonal time series. The extend of this method is by adding seasonal component which use 3 smoothing equations for Level (L), Trend (T) and Seasonality (S) that controlled by alpha, beta and gamma parameters to weight recent and past data with 2 variations to capture constant or proportional seasonal patterns.

Level (L) Smoothing Equation:

$$Lt=\alpha(xt-St-m)+(1-\alpha)(Lt-1+Tt-1)$$

- Take the current observation xt.
- Remove its seasonal effect St−m.
- Combine with previous level and trend.
- Balance both using smoothing factor α (0−1).

Trend (T) Smoothing Equation:

$$Tt=\beta(Lt-Lt-1)+(1-\beta)Tt-1$$

- Compute how the level changed from last step: Lt−Lt−1
- Smooth with previous trend using β(0−1).

Seasonality (S) Smoothing Equation:

$$St=\gamma(xt-Lt)+(1-\gamma)St-m$$

- Compare current value with new xt xt−Lt.
- Update the seasonal pattern using smoothing factor γ.

Holt-Winters Forecast Formula:

$$Ft+h=Lt+hTt+St-m+h$$

To forecast h steps ahead:

- Start with the current level.
- Add trend multiplied by forecast distance h.
- Add the seasonal value corresponding to the future period.

b. Strengths
    - Captures data trend and seasonality.
    - More accurate for a long term predictions.
    - Smooths the data while also forecasting future values.
    - Effective for environmental sensor data with daily cycles.

c. Limitations
    - More computationally heavy.
    - Not always suitable for low end microcontrollers.
    - Requires storing multiple past values.

d. Specific Sensors Suitability
    Highly suitable for BME280 Temperature, Humidity and Pressure sensor because environmental data often shows daily temperature cycles, pressure trends and humidity fluctuations with patterns. This method can model these patterns and forecast environment conditions.

Implementation

Part 1: Creating a Data Source with Google Sheets

Step 1.1: Create a New Google Sheet

Create a new spreadsheet in Google Sheets and rename it to “ESP32 Sensor Data Logger”. Then create 5 column headers in row 1 which is:

- A1: Timestamp.
- B1: Temperature (°C).
- C1: Humidity (%).
- D1: Pressure (atm).
- E1: Distance (cm).

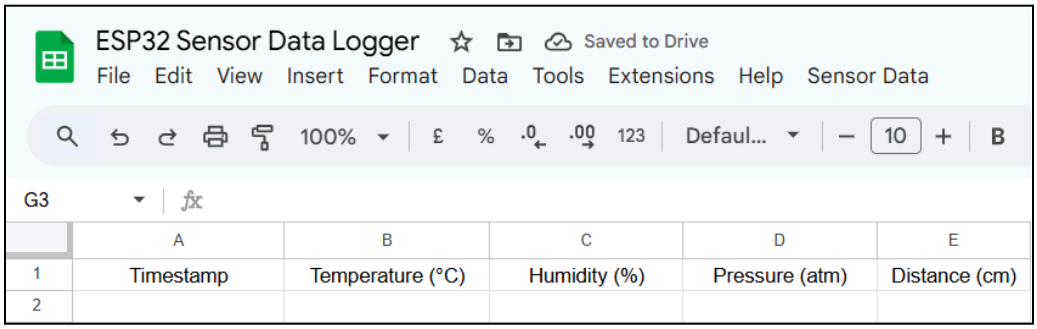


Figure 4: Created a Sheet with Timestamp, Temperature (°C), Humidity (%), Pressure (atm) and Distance (cm) columns.

Step 1.2: Format Data

Select all 5 columns filled with data and click Number on the Format menu. Then choose the appropriate formats for each columns:

- Timestamp column: Date time format.
- Temperature column: Number with 2 decimal places.
- Humidity column: Number with 2 decimal places.
- Pressure column: Number with 2 decimal places.
- Distance column: Number with no decimal places.

1	Timestamp	Temperature (°C)	Humidity (%)	Pressure (atm)	Distance (cm)
2	30/12/2025 15:05:14	29.3	71.42	1.00	28.24
3	30/12/2025 15:35:13	29.51	68.88	1.00	59.45
4	30/12/2025 16:23:43	29.65	67.64	0.99	2.71
5	30/12/2025 16:53:43	29.28	66.95	0.99	4.44
6	30/12/2025 17:23:43	29.34	66.88	0.99	3.59
7	30/12/2025 17:53:44	29.37	67.02	0.99	3.75
8	30/12/2025 18:23:42	29.24	67.12	0.99	3.7
9	30/12/2025 18:53:43	29.18	67.05	1.00	4.29
10	30/12/2025 19:23:42	29.11	67.39	1.00	4.86
11	30/12/2025 19:53:42	29	69.09	1.00	2.96
12	30/12/2025 20:23:43	28.97	69.76	1.00	2.62
13	30/12/2025 20:53:42	28.78	70.75	1.00	2.62
14	30/12/2025 21:23:42	28.81	70.97	1.00	2.62
15	30/12/2025 21:53:46	28.78	71.42	1.00	15.23
16	30/12/2025 22:23:41	28.67	72.44	1.00	37.03
17	30/12/2025 22:53:41	28.69	73.6	1.00	37.03
18	30/12/2025 23:23:41	28.63	73.77	1.00	27.66
19	30/12/2025 23:53:45	28.58	73.8	1.00	18.78
20	31/12/2025 00:23:41	28.46	73.82	1.00	18.78
21	31/12/2025 00:53:40	28.39	73.65	1.00	18.78
22	31/12/2025 01:23:40	28.41	73.38	1.00	18.78
23	31/12/2025 01:53:40	28.38	73.63	1.00	18.31
24	31/12/2025 02:23:41	28.28	73.84	1.00	18.12
25	31/12/2025 02:53:41	28.16	73.97	1.00	17.45

Figure 5: Timestamp Column Data Formatted to Date Time, Temperature, Humidity and Pressure Column Data Formatted to 2 Decimal Place Number and Distance Column Data Formatted to No Decimal Place Number.

Part 2: Implement Forecasting in Google App Script

Step 2.1: Extend Existing Google Apps Script with Forecasting Functionality

Create new function to process historical data with the Holt-Winters and Exponential Moving Average (EMA) Forecasting algorithm implemented to generate forecast for the next 1 hours, 6 hours and 24 hours.

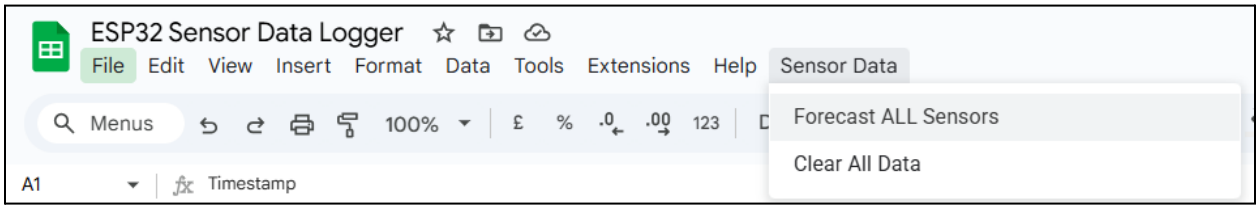


Figure 6: Forecast ALL Sensors Function Button In Google Sheet Menu to Generate 1 Hours, 6 Hours and 24 Hours Forecast Data.

The generated forecasts data will be stored in each individual separate sheet named Forecasts_1H, Forecasts_6H and Forecasts_24H within the same Google Sheet.

	A	B	C	D	E	F	G	H	I	J	K	L	M
1	Timestamp	Temperature Forecast	Temperature Upper	Temperature Lower	Humidity Forecast	Humidity Upper	Humidity Lower	Pressure Forecast	Pressure Upper	Pressure Lower	Distance Forecast	Distance Upper	Distance Lower
2	31/12/2025 00:23:41	28.52	31.37	25.67	74.01	81.41	66.61	1	1.1	0.9	24.12	26.53	21.71
3	31/12/2025 01:23:41	28.44	31.29	25.6	74.84	82.32	67.36	1	1.1	0.9	24.33	26.76	21.9
4	31/12/2025 02:23:41	28.37	31.2	25.53	75.67	83.24	68.1	1	1.1	0.9	24.81	27.29	22.33
5	31/12/2025 03:23:41	28.29	31.12	25.46	76.5	84.15	68.85	1	1.1	0.9	25.03	27.53	22.52
6	31/12/2025 04:23:41	28.21	31.03	25.39	77.33	85.07	69.6	1	1.1	0.9	24.94	27.43	22.44
7	31/12/2025 05:23:41	28.14	30.95	25.32	78.17	85.98	70.35	1	1.1	0.9	24.5	26.95	22.05
8	31/12/2025 06:23:41	28.06	30.86	25.25	79	86.9	71.1	1	1.1	0.9	24.98	27.48	22.48
9	31/12/2025 07:23:41	27.98	30.78	25.18	79.83	87.81	71.85	1	1.1	0.9	25.47	28.01	22.92
10	31/12/2025 08:23:41	27.9	30.7	25.11	80.66	88.73	72.59	1	1.1	0.9	25.32	27.85	22.79
11	31/12/2025 09:23:41	27.83	30.61	25.04	81.49	89.64	73.34	1	1.1	0.9	25.69	28.26	23.12
12	31/12/2025 10:23:41	27.75	30.53	24.98	82.32	90.56	74.09	1	1.1	0.9	25.48	28.02	22.93
13	31/12/2025 11:23:41	27.67	30.44	24.91	83.16	91.47	74.84	1	1.1	0.9	25.86	28.44	23.27
14	31/12/2025 12:23:41	27.6	30.36	24.84	83.99	92.39	75.59	1	1.1	0.9	25.91	28.5	23.32
15	31/12/2025 13:23:41	27.52	30.27	24.77	84.82	93.3	76.34	1	1.1	0.9	25.63	28.19	23.07
16	31/12/2025 14:23:41	27.44	30.19	24.7	85.65	94.22	77.09	1	1.1	0.9	25.28	27.81	22.75
17	31/12/2025 15:23:41	27.37	30.1	24.63	86.48	95.13	77.83	1	1.1	0.9	25.78	28.36	23.2
18	31/12/2025 16:23:41	27.29	30.02	24.56	87.31	96.04	78.58	1	1.1	0.9	26.11	28.72	23.5
19	31/12/2025 17:23:41	27.21	29.93	24.49	88.14	96.96	79.33	1	1.1	0.9	26.26	28.88	23.63
20	31/12/2025 18:23:41	27.13	29.85	24.42	88.98	97.87	80.08	1	1.1	0.9	25.87	28.46	23.28
21	31/12/2025 19:23:41	27.06	29.76	24.35	89.81	98.79	80.83	1	1.1	0.9	26.09	28.69	23.48
22	31/12/2025 20:23:41	26.98	29.68	24.28	90.64	99.7	81.58	1	1.1	0.9	26.47	29.12	23.83
23	31/12/2025 21:23:41	26.9	29.59	24.21	91.47	100.62	82.32	1	1.1	0.9	26.26	28.88	23.63
24	31/12/2025 22:23:41	26.83	29.51	24.14	92.3	101.53	83.07	1	1.1	0.9	26.12	28.73	23.51
25	31/12/2025 23:23:41	26.75	29.42	24.07	93.13	102.45	83.82	1	1.11	0.9	25.61	28.18	23.05

Figure 7: Column of Data After Generate Forecast All Sensors Consisting Timestamp, Temperature Forecast, Temperature Upper, Temperature Lower, Humidity Forecast, Humidity Upper, Humidity Lower, Pressure_atm Forecast, Pressure_atm Upper, Pressure_atm Lower, Distance Forecast, Distance Upper, Distance Lower.

Step 2.2: Google Apps Script Structure With Holt-Winters and Exponential Moving Average (EMA) Forecast Algorithm

The double forecast algorithm generate function are added to existing Google Apps Script by manual using button added to the Google Sheet menu named Sensor Data. A forecast sheets will be created if not existed yet. The function will get all the historical sensor data available in SensorData sheet, calculate the forecast of temperature, humidity and pressure sensor data with Holt-Winters method algorithm, distance sensor data with Exponential Moving Average (EMA) method algorithm and add the timestamp, 1 hour, 6 hours and 24 hours data forecast with each sensor upper bound and lower bound to every designated column made by the function.

```
function generateAllForecasts() {
  [1, 6, 24].forEach(generateForecast);
}

function generateForecast(hours) {
  const ss = SpreadsheetApp.getActive();
  const sensor = ss.getSheetByName("SensorData");
  if (!sensor || sensor.getLastRow() < 2) return;

  const rows = sensor.getRange(2, 1, sensor.getLastRow() - 1,
5).getValues();
  const t = rows.map(r => r[1]), h = rows.map(r => r[2]);
  const p = rows.map(r => r[3]), d = rows.map(r => r[4]);
  const lastTime = new Date(rows[rows.length - 1][0]);

  const tf = holtForecast(t, lastTime, hours);
  const hf = holtForecast(h, lastTime, hours);
  const pf = holtForecast(p, lastTime, hours);
  const df = emaForecast(d, lastTime, hours, 0.3);

  const sheetName = `Forecast_${hours}H`;
  let sh = ss.getSheetByName(sheetName) ||
ss.insertSheet(sheetName);
  if (hours !== 24) {
    sh.clear();
  }

  if (sh.getLastRow() === 0) {
    sh.appendRow([
    "Timestamp",
    "Temperature Forecast","Temperature Upper","Temperature Lower",
    "Humidity Forecast","Humidity Upper","Humidity Lower",
    "Pressure Forecast","Pressure Upper","Pressure Lower",
    "Distance ","Distance Upper","Distance Lower",
    "Alert"
  ]);
  }
  for (let i = 0; i < hours; i++) {
    const alert = checkThresholds(tf[i].f, hf[i].f, pf[i].f,
```

```
df[i].f);
    sh.appendRow([
      tf[i].time, tf[i].f, tf[i].u, tf[i].l,
      hf[i].f, hf[i].u, hf[i].l,
      pf[i].f, pf[i].u, pf[i].l,
      df[i].f, df[i].u, df[i].l,
      alert
    ]);
  }
}

function holtForecast(data, lastTime, hours) {
  let alpha = 0.3, beta = 0.2;
  let level = data[0], trend = data[1] - data[0];
  for (let i = 1; i < data.length; i++) {
    const prev = level;
    level = alpha * data[i] + (1 - alpha) * (level + trend);
    trend = beta * (level - prev) + (1 - beta) * trend;
  }
  const out = [];
  for (let i = 1; i <= hours; i++) {
    out.push(makeRow(lastTime, i, level + trend * i));
  }
  return out;
}

function emaForecast(data, lastTime, hours, alpha) {
  let ema = data[0], out = [];
  for (let i = 1; i < data.length; i++) {
    ema = alpha * data[i] + (1 - alpha) * ema;
  }
  for (let i = 1; i <= hours; i++) {
    let forecast = Math.max(0, ema + (Math.random() * 2 - 1) * 0.02
* ema);
    out.push(makeRow(lastTime, i, forecast));
    ema = forecast;
  }
  return out;
}

function makeRow(lastTime, step, val) {
  const t = new Date(lastTime.getTime() + step * 3600000);
  return { time: t, f: Math.round(val * 100) / 100, u:
Math.round(val * 110) / 100, l: Math.round(val * 90) / 100 };
}

function checkThresholds(temp, hum, pres, dist) {
  let alerts = [];
  if (temp < ALERT_THRESHOLDS.temperature.min || temp >
ALERT_THRESHOLDS.temperature.max) alerts.push("Temp");
  if (hum < ALERT_THRESHOLDS.humidity.min || hum >
```

```
ALERT_THRESHOLDS.humidity.max) alerts.push("Hum");
  if (pres < ALERT_THRESHOLDS.pressure.min || pres >
ALERT_THRESHOLDS.pressure.max) alerts.push("Pres");
  if (dist < ALERT_THRESHOLDS.distance.min || dist >
ALERT_THRESHOLDS.distance.max) alerts.push("Dist");
  return alerts.join(", ") || "OK";
}
```

Part 3: Integrate Google Gemini API in Google App Script

Step 3.1: Set Up And Enable Gemini API

Add a script property in Google App Script project properties and enter the Gemini API key from the Google AI Studio website to enable Gemini API integration with the Google App Script.

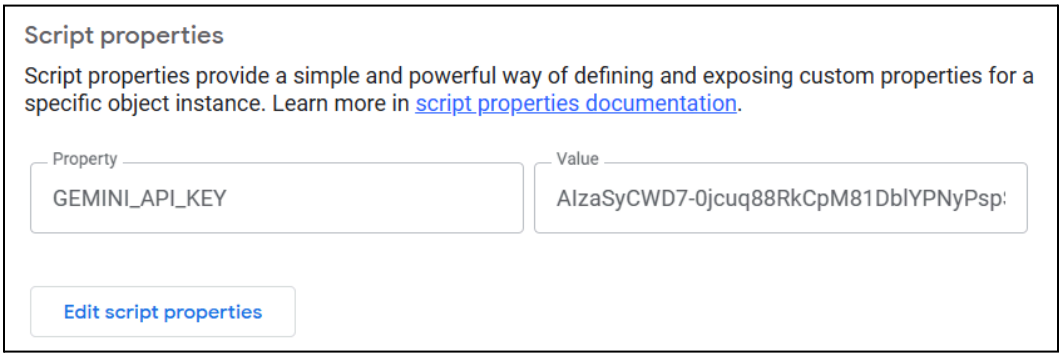


Figure 8: Enter the Gemini API Key Into Property And Value in Google App Script Properties.

Step 3.2: Create Functions To Call Google Gemini API Summaries

The summary data generate function are added to existing Google Apps Script by a manual button added to the Google Sheet menu named Sensor Data below the forecast button. The summaries sheets will be created if not existed yet.

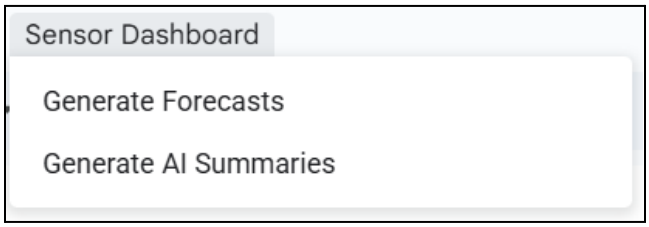


Figure 9: Generate AI Summaries Function Button In Google Sheet Menu to Generate Real Time Data Analysis, Trend Analysis and Predictive Insights Data.

The function will get all the historical sensor data and 24 hours forecast data available in SensorData and Forecast_24H sheet, give 3 prompt to Gemini API for real time data analysis and trend analysis summaries based on SensorData sheet and predictive insight summary based on Forecast_24H and placed each response to every designated column made by the function.

Timestamp	Sensor Analysis
All Sensors	Here's an analysis of the sensor data: | Sensor Type | Combined Summary | | :---------- | :---------- ---------- ---------- | | **Timestamps** | 3 days of data (Dec 30 15:05 to Jan 2 15:51 SGT), recorded consistently ~30-minute intervals, providing regular, high-frequency environmental sampling. | | **Temperature** | Diurnal cycles (27.78-29.85°C) precede a sharp Dec 31 drop from 29.85°C to 22.83°C. Recovery with overnight dips suggests active localized cooling or environmental system activation. | | **Humidity** | Humidity inversely correlates with temperature (52.74-76.09%). A significant Dec 31 drop (69.16% to 52.74%) mirrors temperature, strongly implying a drying event, likely environmental controls. | | **Pressure** | Minimal pressure fluctuation (0.9936-0.9999) with gentle diurnal patterns. Stability suggests localized temperature/humidity changes, not broad atmospheric shifts. | | **Distance** | Initially dynamic (2.71-59.45), distance sharply dropped from 51.1 to 1.98 on Dec 31, coinciding with environmental shifts. It then stabilized at ~3.28, indicating an object moved close and remained fixed. |

Figure 10: Gemini AI Prompt Response in Sensor Analysis Column.

Timestamp	Sensor Trend Analysis
All Sensors	An executive summary of sensor data from December 30, 2025, to January 02, 2026, reveals distinct environmental shifts. Temperature fluctuated within typical diurnal cycles, ranging from a minimum of 22.83°C to a maximum of 29.85°C. Notably, two sharp temperature drops occurred on Wednesday evening and Friday afternoon, mirrored by inverse shifts in humidity, which reached a high of 76.09% and a low of 52.74%. Pressure remained relatively stable, hovering between 0.9936 and 0.9999 with minor daily variations, showing no significant anomalies. The "Distance" sensor exhibited the most pronounced trend change. After initial erratic readings, peaking at 59.45 units, it stabilized dramatically to a consistently low range (1.98-3.28 units) for the final two days of observation. This abrupt shift from variability to extreme stability suggests a significant change in the monitored object's position or the immediate environment.

Figure 11: Gemini AI Prompt Response in Sensor Trend Analysis Column.

Timestamp	Predictive Insight
All Sensors	For the 24-hour period beginning Wednesday, December 31, 2025, in Singapore, expect a day of gradually cooling temperatures and significantly rising humidity. Starting around 28.5°C, temperatures will steadily decline to approximately 26.8°C by late evening. Humidity, however, will surge from 74% to over 93%, indicating increasingly muggy conditions and a heightened potential for precipitation, especially as the day progresses. Atmospheric pressure is forecast to remain stable, suggesting no major weather system movements. 'Distance' (likely visibility) will see minor fluctuations, generally remaining between 24 and 26 units, pointing to consistent ambient conditions. A critical environmental shift is projected immediately following this 24-hour window. Overnight into January 1st, temperatures are set to dramatically drop to around 21°C and continue falling, paired with a sharp decrease in humidity to below 50% and a substantial reduction in 'distance' to just 2 units, signaling a sudden, significant change in weather.

Figure 12: Gemini AI Prompt Response in Predictive Insight Column.

Step 3.3: Google Apps Script Structure With Google Gemini API Function

```
/**************************************************
 * CONFIG
 **************************************************/
const GEMINI_ENDPOINT =
"https://generativelanguage.googleapis.com/v1beta/models/gemini-2.5
-flash:generateContent";
const ALERT_THRESHOLDS = {
  temperature: { min: 0, max: 50 },
  humidity: { min: 20, max: 80 },
  pressure: { min: 0.8, max: 1.2 },
  distance: { min: 0, max: 500 }
};
const CONTEXT_READINGS = 5;

/**************************************************
 * GEMINI API CALL
 **************************************************/
function callGeminiAPI(promptText) {
  const apiKey =
PropertiesService.getScriptProperties().getProperty("GEMINI_API_KEY
");
  if (!apiKey) throw new Error("GEMINI_API_KEY not set in Script
Properties");

  const body = { contents: [{ parts: [{ text: promptText }] }] };

  const response =
UrlFetchApp.fetch(`${GEMINI_ENDPOINT}?key=${apiKey}`, {
    method: "post",
    contentType: "application/json",
    payload: JSON.stringify(body),
    muteHttpExceptions: true
  });

  const result = JSON.parse(response.getContentText());
  if (result.error) return `Gemini error: ${result.error.message}`;

  try {
    return result.candidates
      .map(c => c.content?.parts?.map(p => p.text).join("") || "")
      .join("\n")
      .trim();
  } catch (e) {
    return "Gemini returned unexpected format";
  }
}

/**************************************************
```

```
 * AI SUMMARY FUNCTIONS
 ************************************************/
function generateAllAISummaries() {
  generateRealTimeDataAnalysis();
  generateTrendAnalysis();
  generatePredictiveInsights(); // only 24-hour forecast
}

/* Real-Time Data Analysis */
function generateRealTimeDataAnalysis() {
  const ss = SpreadsheetApp.getActive();
  const sheet = ss.getSheetByName("SensorData");
  if (!sheet) return;

  const rows = sheet.getRange(2, 1, sheet.getLastRow()-1,
5).getValues();
  let summarySheet = ss.getSheetByName("RealTimeDataAnalysis") ||
ss.insertSheet("RealTimeDataAnalysis");
  summarySheet.clear();
  summarySheet.appendRow(["Timestamp","Sensor Analysis"]);

  // Batch all sensors into one prompt
  const prompt = `
Analyze the following sensor data in 150 words:
Timestamps: ${rows.map(r=>r[0]).join(", ")}
Temperature: ${rows.map(r=>r[1]).join(", ")}
Humidity: ${rows.map(r=>r[2]).join(", ")}
Pressure: ${rows.map(r=>r[3]).join(", ")}
Distance: ${rows.map(r=>r[4]).join(", ")}
Provide one combined summary per sensor in a single column.
  `;
  const summary = callGeminiAPI(prompt);
  summarySheet.appendRow(["All Sensors", summary]);
}

/* Trend Analysis */
function generateTrendAnalysis() {
  const ss = SpreadsheetApp.getActive();
  const sheet = ss.getSheetByName("SensorData");
  if (!sheet) return;

  const rows = sheet.getRange(2, 1, sheet.getLastRow()-1,
5).getValues();
  let summarySheet = ss.getSheetByName("TrendAnalysis") ||
ss.insertSheet("TrendAnalysis");
  summarySheet.clear();
  summarySheet.appendRow(["Timestamp","Sensor Trend Analysis"]);

  const prompt = `
Provide an executive trend summary of the following sensor data in
150 words:
```

```
Timestamps: ${rows.map(r=>r[0]).join(", ")}
Temperature: ${rows.map(r=>r[1]).join(", ")}
Humidity: ${rows.map(r=>r[2]).join(", ")}
Pressure: ${rows.map(r=>r[3]).join(", ")}
Distance: ${rows.map(r=>r[4]).join(", ")}
Focus on trends, anomalies, min, max, abnormality.
Provide one combined summary in a single column.
  `;
  const summary = callGeminiAPI(prompt);
  summarySheet.appendRow(["All Sensors", summary]);
}

/* Predictive Insights (24-hour forecast only) */
function generatePredictiveInsights() {
  const ss = SpreadsheetApp.getActive();
  const sheet = ss.getSheetByName("Forecast_24H");
  if (!sheet) return;

  const rows = sheet.getRange(2, 1, sheet.getLastRow()-1,
sheet.getLastColumn()).getValues();
  let summarySheet = ss.getSheetByName("PredictiveInsight_24H") ||
ss.insertSheet("PredictiveInsight_24H");
  summarySheet.clear();
  summarySheet.appendRow(["Timestamp","Predictive Insight"]);

  const prompt = `
Provide predictive insights for the following 24-hour forecast in
150 words:
Timestamps: ${rows.map(r=>r[0]).join(", ")}
Temperature Forecast: ${rows.map(r=>r[1]).join(", ")}
Humidity Forecast: ${rows.map(r=>r[4]).join(", ")}
Pressure Forecast: ${rows.map(r=>r[7]).join(", ")}
Distance Forecast: ${rows.map(r=>r[10]).join(", ")}
Provide one combined summary in a single column.
  `;
  const summary = callGeminiAPI(prompt);
  summarySheet.appendRow(["All Sensors", summary]);
}
```

Part 4: Creating a Looker Studio Dashboard

Step 4.1: Create New Looker Studio Report

Go to Looker Studio, click the Create button and click Report. Choose Google Sheets at the data source selection and select where SensorData sheet are located. Then click ADD TO REPORT.

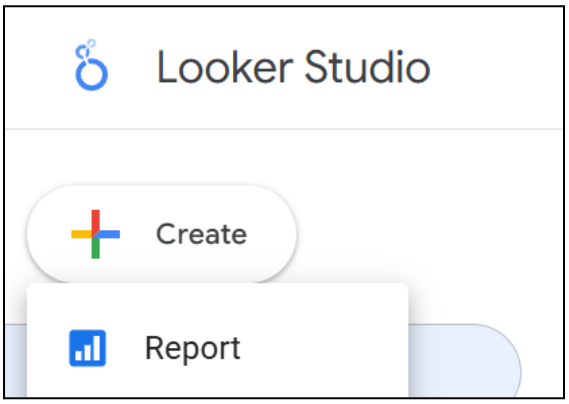


Figure 13: Create a New Report by Clicking the Create Button.

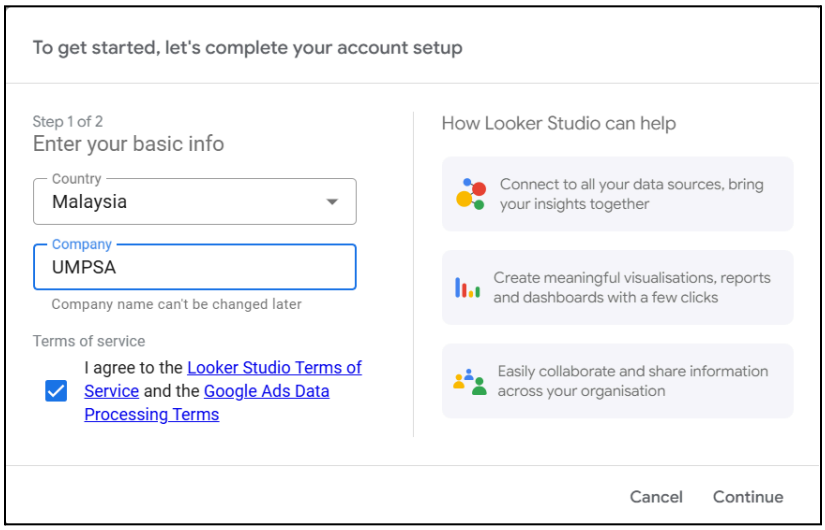


Figure 14: Enter the Basic Info Before Continuing.

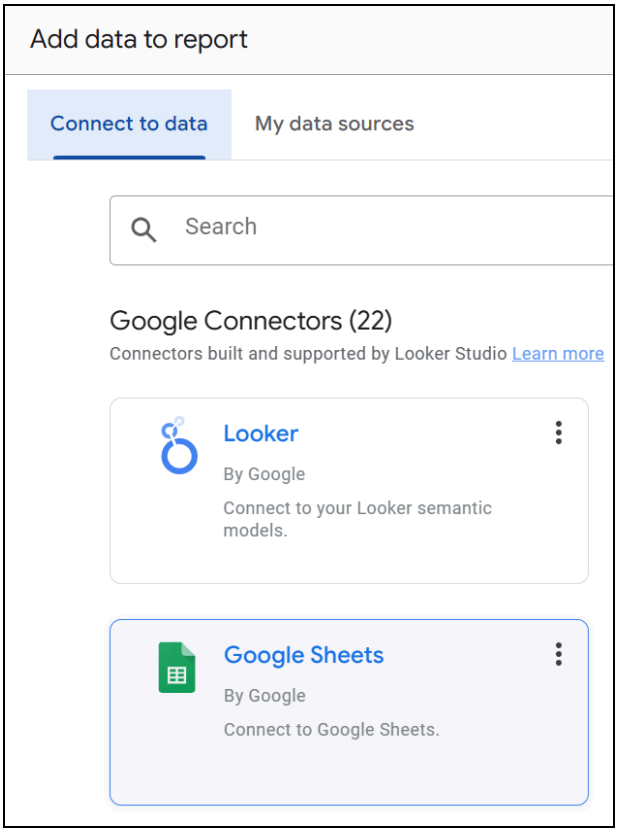


Figure 15: Select Google Sheets to Connect and Add Data to Report.

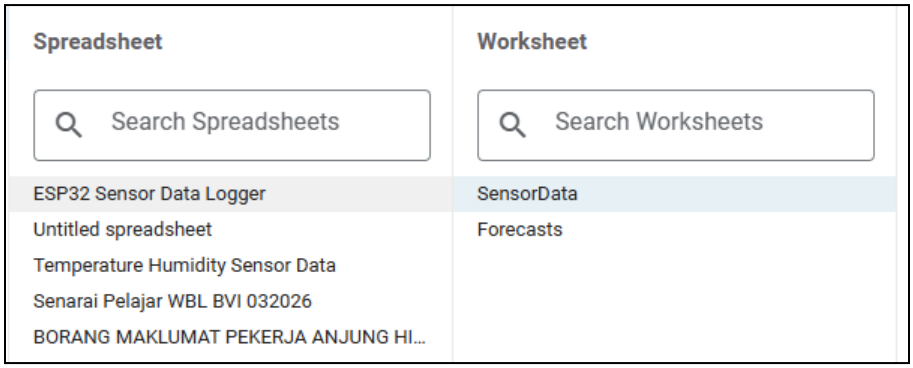


Figure 16: Select the ESP32 Sensor Data Logger Spreadsheet.

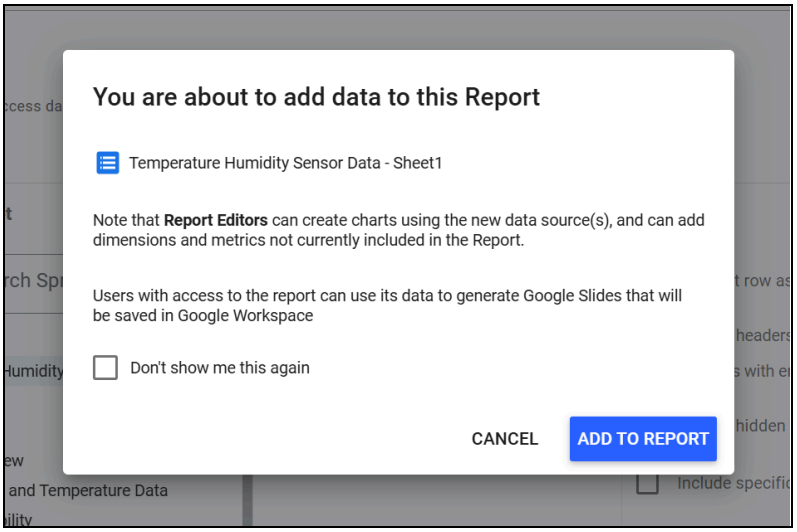


Figure 17: Confirm the Selected Data to Be Added to Report.

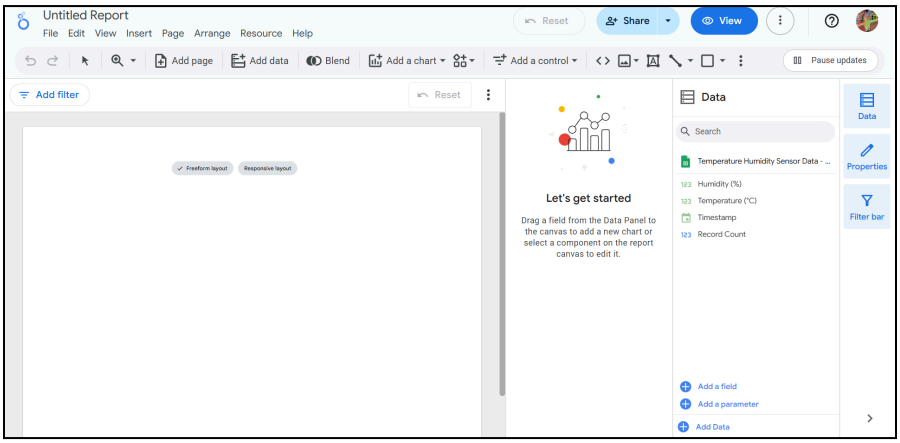


Figure 18: Blank Dashboard Report With ESP32 Sensor Data Logger Spreadsheet Resources.

Step 4.2: Configure Data Source

The SensorData sheet data types in Looker Studio are correctly identified where the Timestamp should be a Date & Time, Temperature, Humidity, Pressure and Distance dimensions should be a numeric metric.

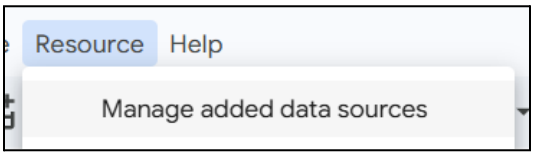


Figure 19: Select Manage added data sources to View Data Types.

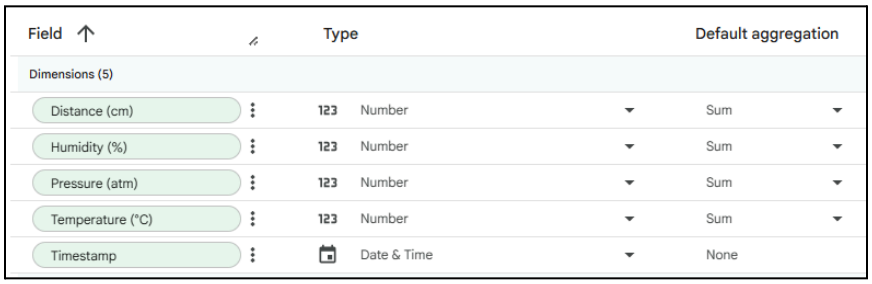


Figure 20: Temperature, Humidity, Pressure and Distance are Number Data Type and Timestamp are Date & Time Data Type.

Forecast sheet data types in Looker Studio is correctly identified too where the Timestamp should be a Date & Time too, Temperature Forecast, Temperature Upper, Temperature Lower, Humidity Forecast, Humidity Upper, Humidity Lower, Pressure_atm Forecast, Pressure_atm Upper, Pressure_atm Lower, Distance Forecast, Distance Upper and Distance Lower dimensions should be a numeric metric.

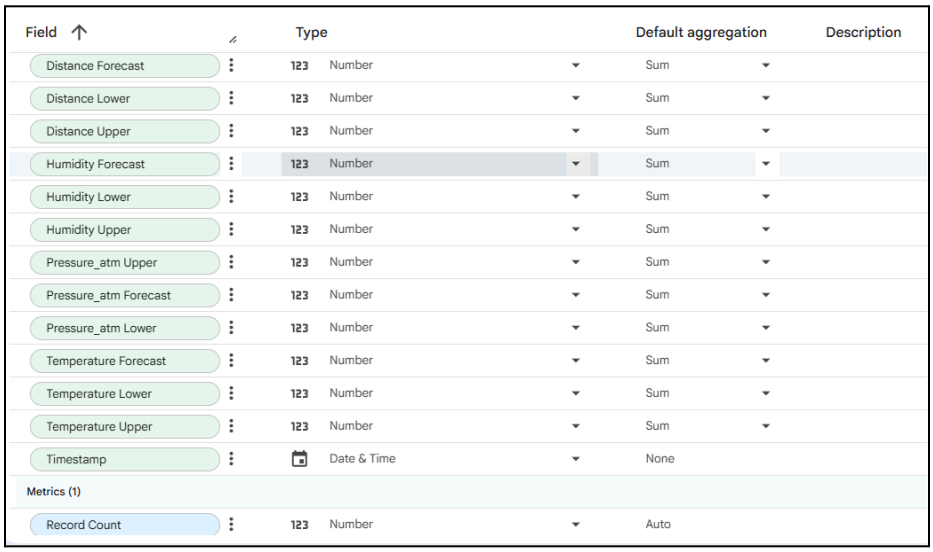


Figure 21: All the Forecast Data are Number Data Type and Timestamp are Date & Time Data Type.

The RealTimeDataAnalysis, TrendAnalysis and PredictiveInsight_24H sheets data types in Looker Studio are correctly identified where the Sensor Analysis, Sensor Trend Analysis and Predictive Insight dimensions should be text.

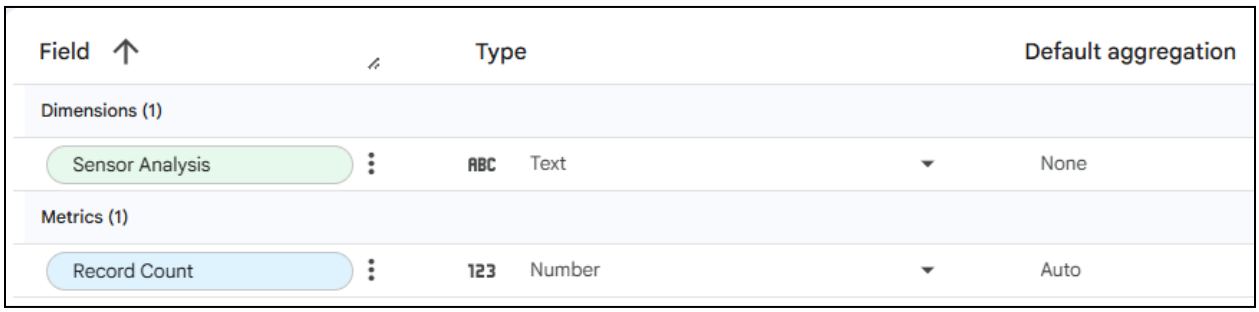


Figure 22: Sensor Analysis Data are Text Data Type.

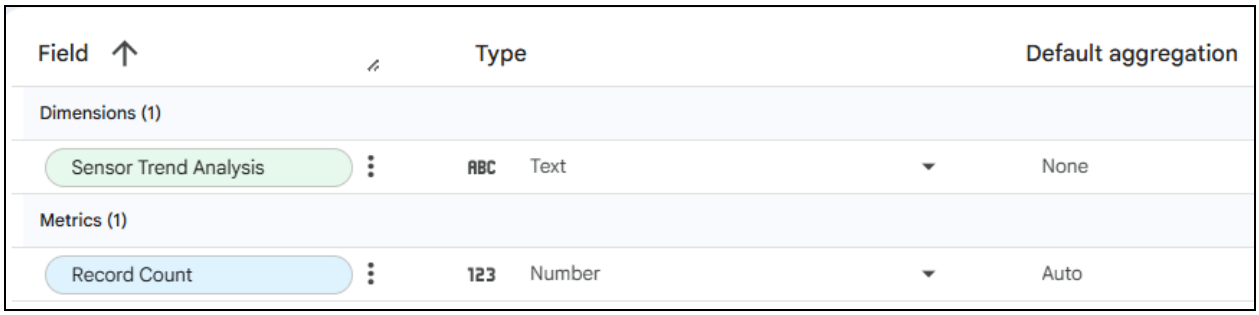


Figure 23: Sensor Trend Analysis Data are Text Data Type.

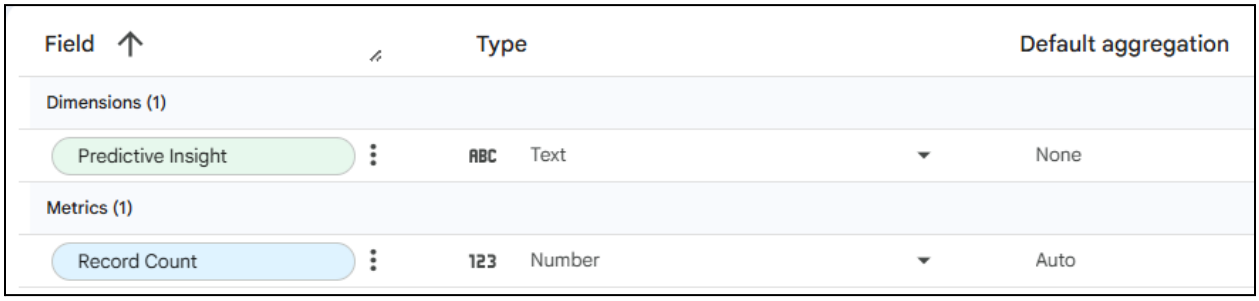


Figure 24: Predictive Insight Data are Text Data Type.

Step 4.3: Design Dashboard

In the new report, the dashboard is in responsive layout and has multiple page. In the first page are added title, data range control, dual axis chart, separate metrics scorecard and 5 button to navigate to different pages in dashboard.

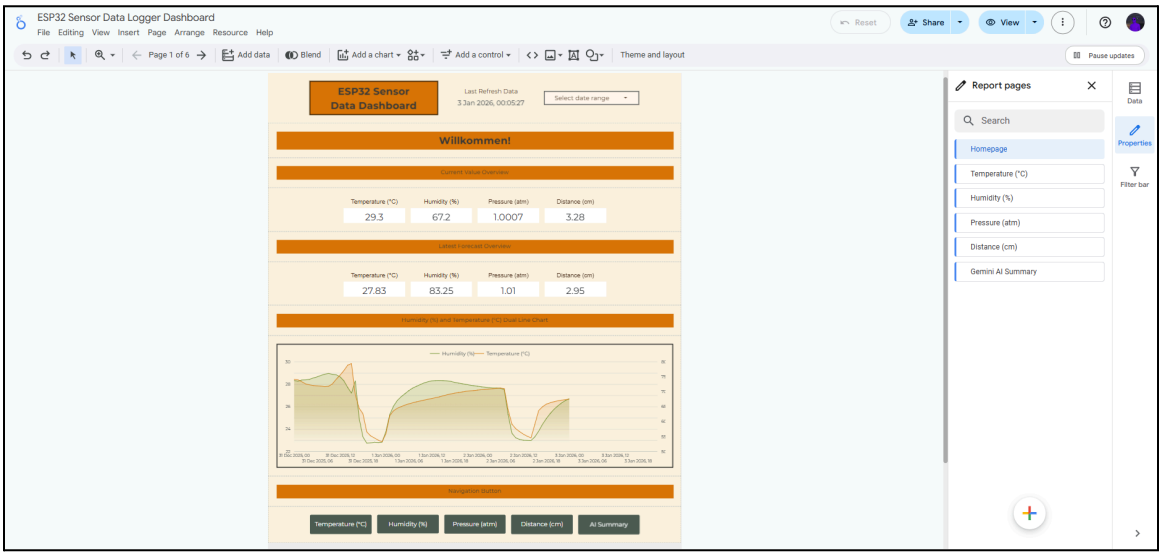


Figure 25: Dashboard First Page Design Included With Title, Date Range Control, Dual Axis Chart, Separate Metrics Scorecard and Buttons.

Every pages except the AI Summary page display each sensor data and forecast data individually but the pages use same data display types. Every data pages use title, date range control, combo chart, separate metrics scorecard and 5 buttons.

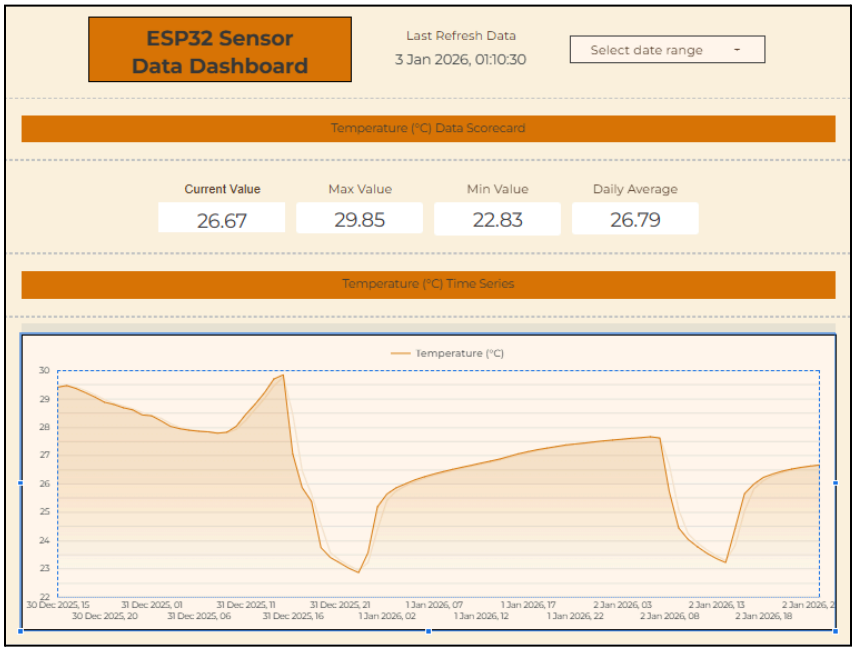


Figure 26: Upper View Temperature Dashboard Page Design Included With Title, Date Range Control, Time Series Chart and 4 Separate Metrics Scorecard.

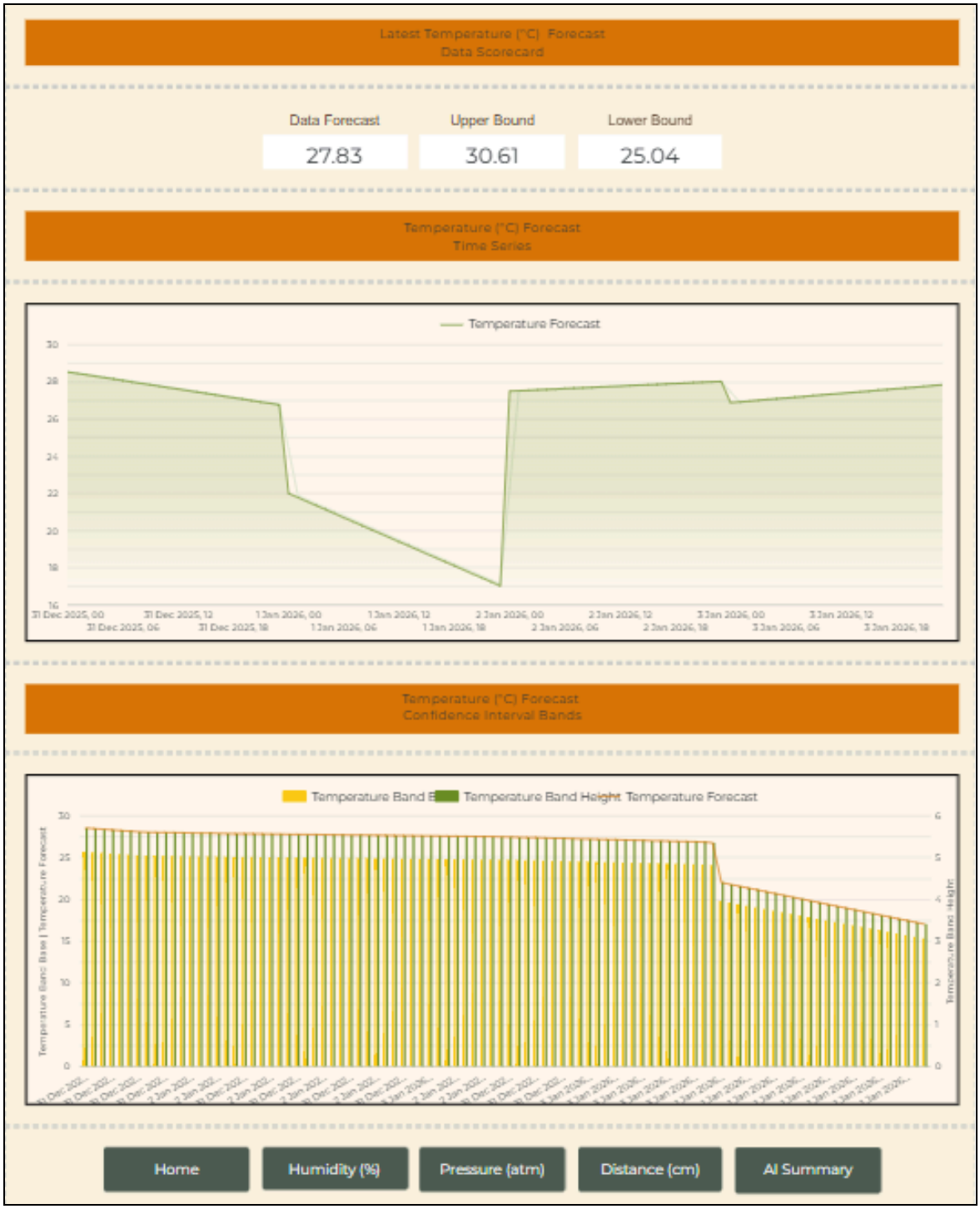


Figure 27: Lower View Temperature Dashboard Page Design Included Time Series Chart, Combo Chart and 3 Separate Metrics Scorecard.

AI Summary page display real time data analysis summary, trend analysis summary and predictive insight summary all in the same page use only table as the page data display types alongside with title, date range control and 5 buttons.

Figure 28: AI Summary Page Design Included With Title, Date Range Control and 3 Tables.

Step 4.4: Finalize and Share Dashboard

Click View mode to test dashboard functionality and Edit mode for any adjustments needed.

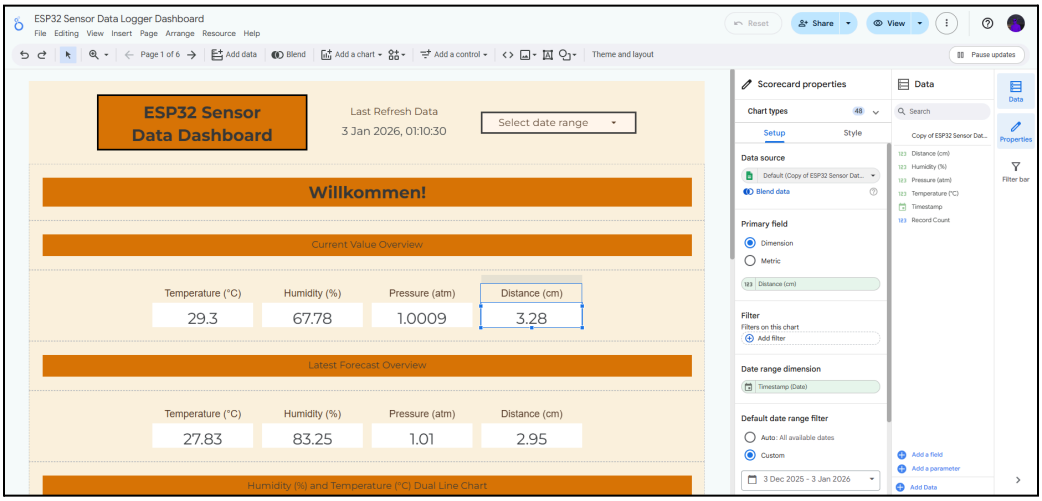


Figure 29: Looker Studio Edit Mode.

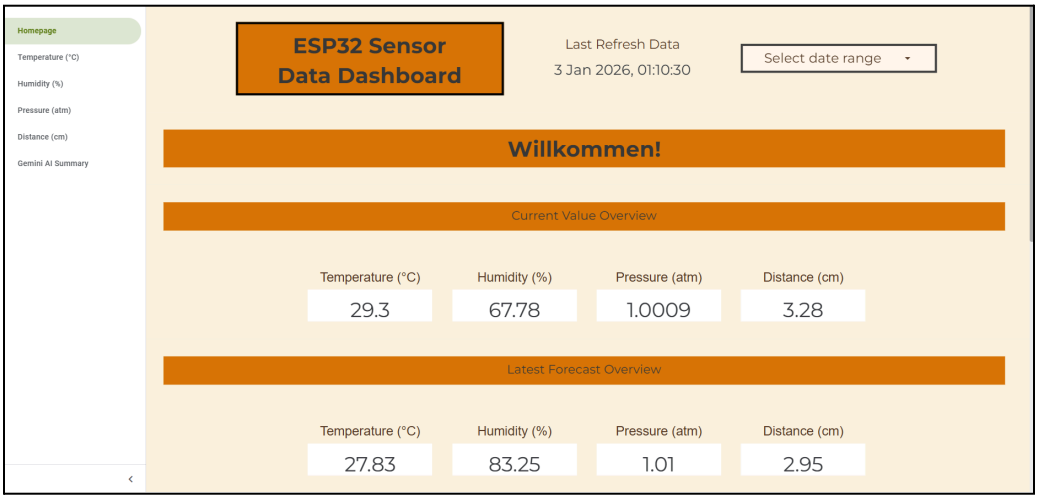


Figure 30: Looker Studio View Mode.

Then click Share in top right corner and set the access of the dashboard from Restricted to Anyone on the Internet with the link can find and view.

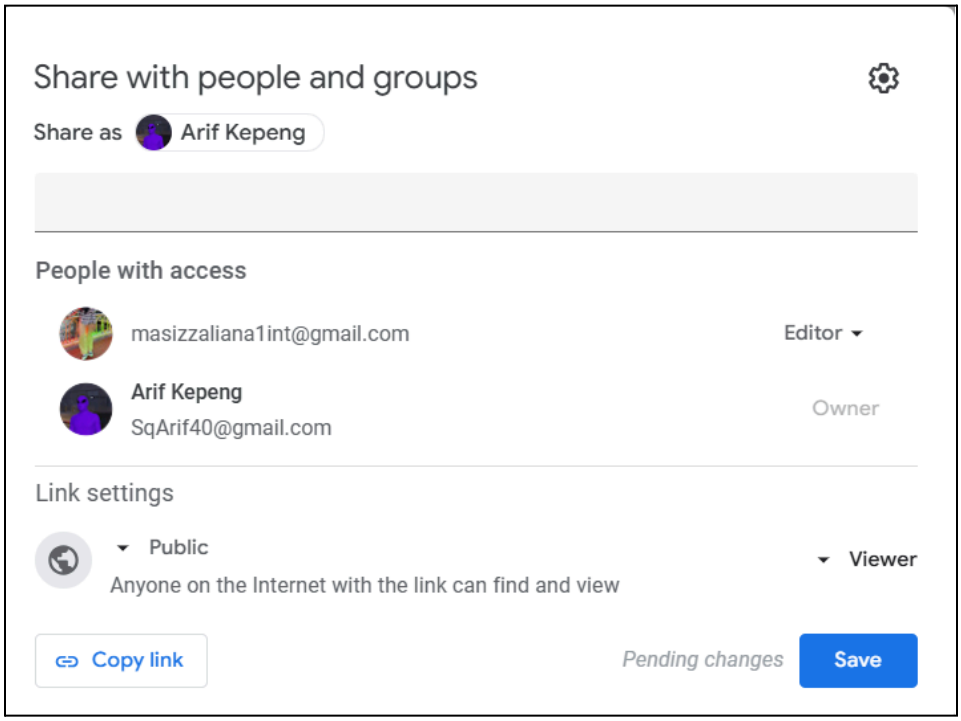


Figure 31: Changed the Access of Dashboard to Public but Viewer Only.

Get the dashboard embed URL by opening Embed Report from the File menu.

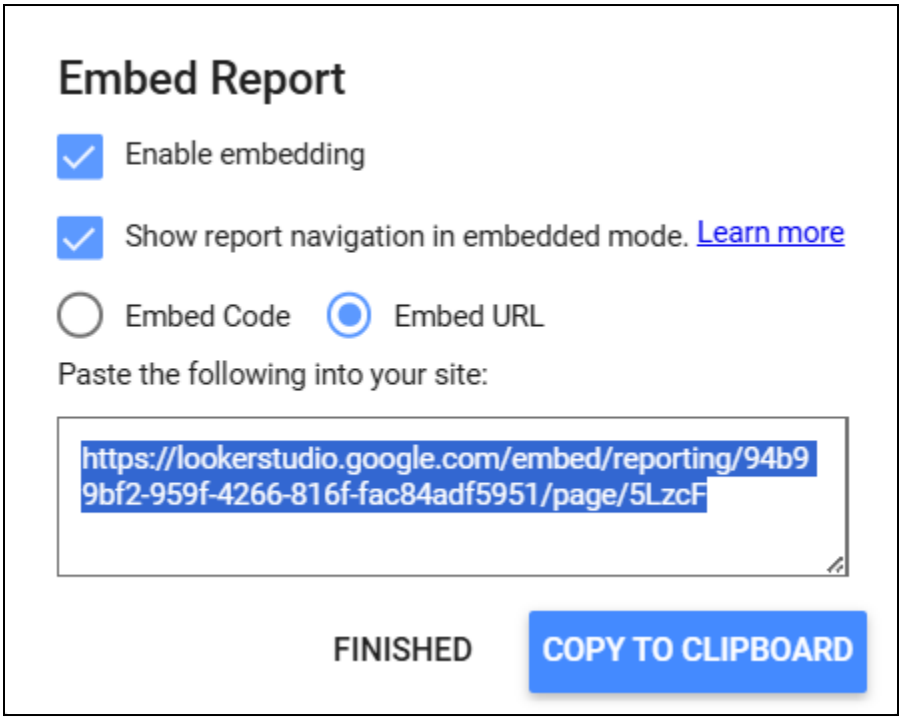


Figure 32: Click Both Check Boxes, Select Embed URL and COPY TO CLIPBOARD.

Part 5: Creating Mobile App With Embedded Dashboard

Step 5.1: Set Up Mobile Development Environment

Open Android Studio application and create a new project using Empty Views Activity template.

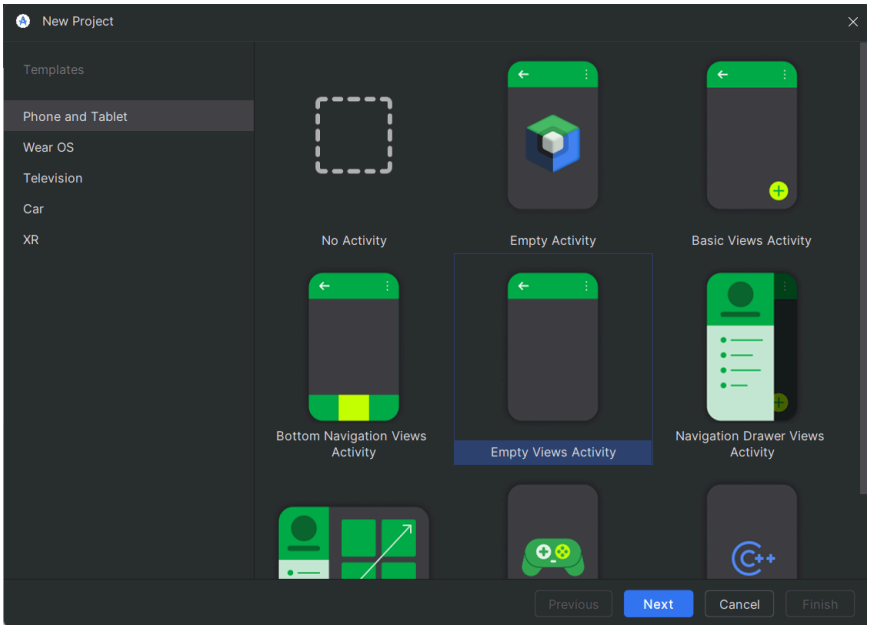


Figure 33: Select Empty Views Activity and Click Next.

Change the project name to SensorDashboard and use Android 5.0 Lollipop.

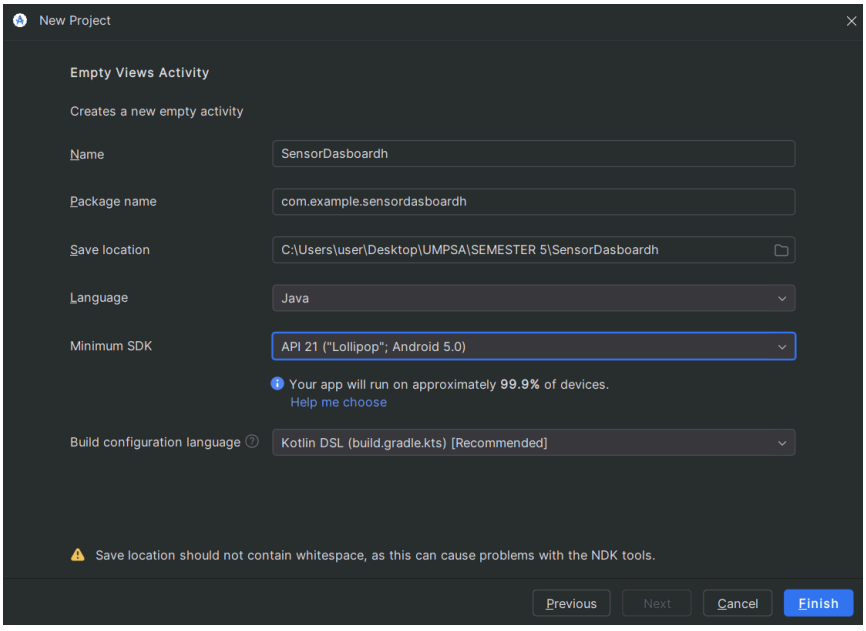


Figure 34: New Empty Views Activity Project Named SensorDashboard Using Java Language With Android 5.0 at Minimum SDK.

Step 5.2: Configure the Webview in App

Replace the existing activity_main.xml file code with this code:

```
<?xml version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.ConstraintLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".MainActivity">

    <WebView
        android:id="@+id/dashboardWebView"
        android:layout_width="0dp"
        android:layout_height="0dp"
        app:layout_constraintTop_toTopOf="parent"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintEnd_toEndOf="parent" />

    <ProgressBar
        android:id="@+id/progressBar"
        style="?android:attr/progressBarStyleLarge"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:visibility="gone"
        app:layout_constraintTop_toTopOf="parent"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintEnd_toEndOf="parent" />

</androidx.constraintlayout.widget.ConstraintLayout>
```

Update the MainActivity.java file code with this code:

```
package com.example.sensordashboard;

import androidx.activity.OnBackPressedCallback;
import androidx.appcompat.app.AppCompatActivity;

import android.annotation.SuppressLint;
import android.os.Bundle;
import android.view.View;
import android.webkit.WebSettings;
```

```
import android.webkit.WebView;
import android.webkit.WebViewClient;
import android.widget.ProgressBar;

public class MainActivity extends AppCompatActivity {
    private WebView dashboardWebView;
    private ProgressBar progressBar;

    @SuppressLint("SetJavaScriptEnabled")
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);

        dashboardWebView = findViewById(R.id.dashboardWebView);
        progressBar = findViewById(R.id.progressBar);

        WebSettings webSettings = dashboardWebView.getSettings();
        webSettings.setJavaScriptEnabled(true);
        webSettings.setDomStorageEnabled(true);
        webSettings.setLoadWithOverviewMode(true);
        webSettings.setUseWideViewPort(true);
        webSettings.setSupportZoom(true);
        webSettings.setBuiltInZoomControls(true);
        webSettings.setDisplayZoomControls(false);

        dashboardWebView.setWebViewClient(new WebViewClient() {
            @Override
            public void onPageFinished(WebView view, String url) {
                progressBar.setVisibility(View.GONE);
                super.onPageFinished(view, url);
            }
        });

        String dashboardUrl =

"https://lookerstudio.google.com/embed/reporting/94b99bf2-959f-4266
-816f-fac84adf5951/page/5LzcF";
        progressBar.setVisibility(View.VISIBLE);
        dashboardWebView.loadUrl(dashboardUrl);

        getOnBackPressedDispatcher().addCallback(this, new
OnBackPressedCallback(true) {
            @Override
```

```
            public void handleOnBackPressed() {
                if (dashboardWebView.canGoBack()) {
                    dashboardWebView.goBack();
                } else {
                    finish();
                }
            }
        });
    }
}
```

Update the AndroidManifest.xml file code to add internet permission with this code:

```
<?xml version="1.0" encoding="utf-8"?>
<manifest
    xmlns:android="http://schemas.android.com/apk/res/android"
    package="com.example.sensordashboard">
    <uses-permission
        android:name="android.permission.INTERNET" />
    <application
        android:allowBackup="true"
        android:icon="@mipmap/ic_launcher"
        android:label="@string/app_name"
        android:roundIcon="@mipmap/ic_launcher_round"
        android:supportsRtl="true"
        android:theme="@style/Theme.SensorDashboard">
        <activity
            android:name=".MainActivity"
            android:exported="true">
            <intent-filter>
                <action
                    android:name="android.intent.action.MAIN" />
                <category
                    android:name="android.intent.category.LAUNCHER"
/>
            </intent-filter>
        </activity>
    </application>
</manifest>
```

Step 5.3: Insert Looker Studio Dashboard Embed URL

Open the MainActivity.java code and replace the existing URL with the created Looker Studio dashboard at the line containing `dashboardURL`.

Figure 35: Changed Existing URL With Created Dashboard Copied Embed URL.

Step 5.4: Build and Run App

Run the built app by clicking the Run button in Android Studio and wait for the app to install and launch.

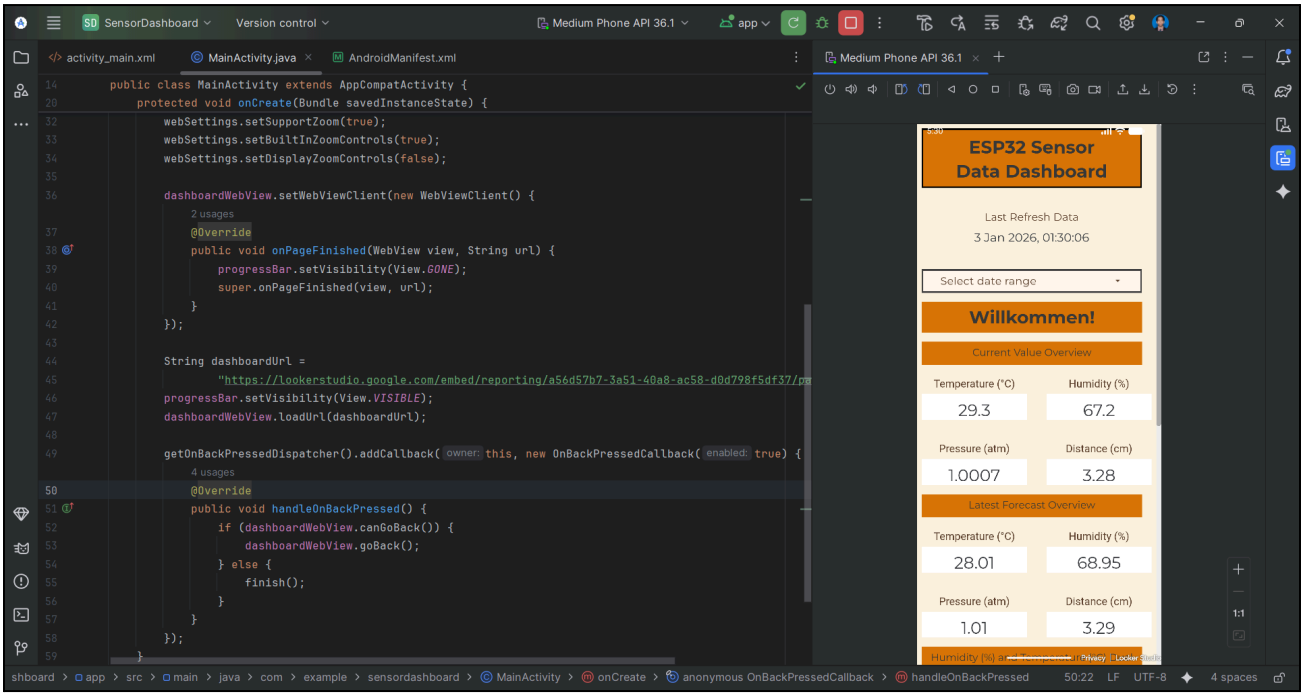


Figure 36: Created Looker Studio Dashboard Successfully Run in App Using Android Studio.

Base Code Modification

1. ESP32 Arduino IDE Code.
The original Ultrasonic code was modified by adding the BME280 sensor, adjusting the interval and updating the data sensor validation.

1. Added BME280 Sensor Support.
   - Libraries for BME280 Sensor.

```
#include <Wire.h>
#include <Adafruit_Sensor.h>
#include <Adafruit_BME280.h>
```

   - BM280 Sensor Global Object.

```
Adafruit_BME280 bme;
```

   - BME280 Sensor Initialization Setup.

```
if (!bme.begin(0x76)) {
  Serial.println("Could not find BME280 sensor!");
  while (1);
}
```

   - BME280 Sensor Readings in Loop.

```
float temperature = bme.readTemperature();
float humidity = bme.readHumidity();
```

2. Fixed Ultrasonic Pin Usage.
   - In the original code, the pin for ultrasonic was unclear and the Ultrasonic library required 2 pins which are ECHO and TRIG. Then it was replaced with this code:

```
#define TRIG_PIN 13
#define ECHO_PIN 12
Ultrasonic ultrasonic(TRIG_PIN, ECHO_PIN);
```

3. Added Temperature and Humidity with Existing Distance to the JSON.
   - In the original code, only distance data sent to Google Sheets and with BME280 sensor being added, the temperature and humidity data are added to the existing code:

```
doc["temperature"] = temperature;
doc["humidity"] = humidity;
doc["distance"] = distance;
```

4. Added Sensor Data Validation.
   - Improved the original data validation code to prevent bad or incomplete readings from being sent to Google Sheets:

```
long distance = ultrasonic.read();
if (isnan(temperature) || isnan(humidity) || distance <= 0) {
  Serial.println("Failed to read from sensors!");
  return;
}
```

5. Updated Serial Output for Debugging.
   - Added a print statement to see all 3 sensor readings in the Serial Monitor.

```
Serial.print("Temperature: ");
Serial.print(temperature);
Serial.print("°C, Humidity: ");
Serial.print(humidity);
Serial.print("%, Distance: ");
Serial.print(distance);
Serial.println(" cm");
```

6. Added 30 Minute Period Sampling with Averaging.
   - Improved the original system that sent single readings every 10 minutes with the new system which reads sensor data every 1 minute then stores readings until reach 30 samples and sends 30 minute averaged values to Google Sheets.

```
const unsigned long readInterval = 60000;
const int maxSamples = 30;
```

Challenges

1. The screen appeared completely blank white even after refreshing while running the app in Android Studio as shown in Figure 37.

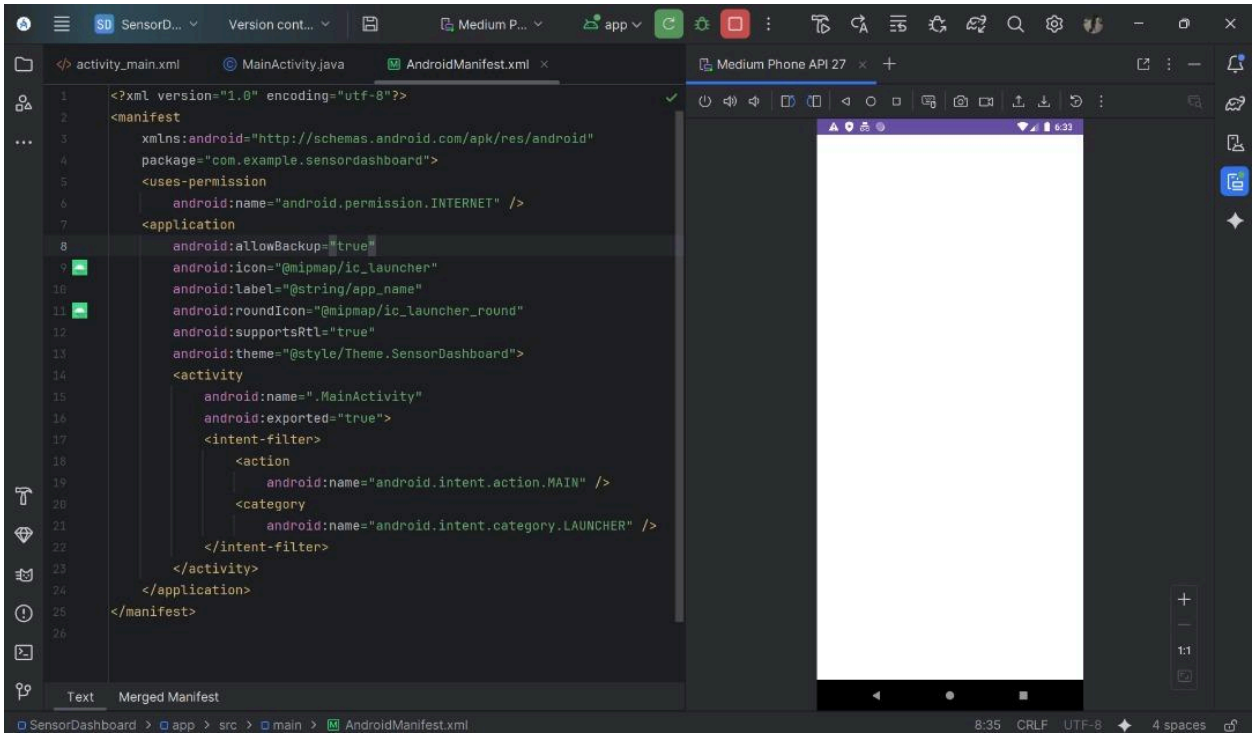


Figure 37: App Appear Blank White.

2. An error message pop-up saying “Couldn’t save the file. Sorry! We can’t save this report right now.” after the app was running as shown in Figure 38.

Figure 38: Error Message Pop-up in App.

3. The forecasted distance data is negative as shown in Figure 39 even though the historical data is stable and not below 0 cm.

Distance Forecast	Distance Upper	Distance Lower
-3.29	-3.62	-2.96
34.02	37.42	30.62
-11.47	-12.62	-10.33
-11.39	-12.53	-10.25
-11.44	-12.58	-10.29
-11.67	-12.84	-10.51
-16.63	-18.29	-14.96
-12.24	-13.46	-11.01
-14.45	-15.9	-13.01
-22.4	-24.64	-20.16
-9.61	-10.57	-8.65
-11.94	-13.13	-10.74
-10.36	-11.4	-9.32
-11.02	-12.12	-9.92
-13.53	-14.88	-12.17

Figure 39: Negative Forecasted Distance Data.

4. The Gemini AI API not generating AI summaries when press Generate AI Summaries more than one time in a single day.

Gemini error: You exceeded your current quota, please check your plan and billing details. For more information on this error, head to: https://ai.google.dev/gemini-api/docs/rate-limits. To monitor your current usage, head to: https://ai.dev/usage?tab=rate-limit. * Quota exceeded Please retry in 28.123972747s.for metric: generativelanguage.googleapis.com/generate_content_free_tier_requests, limit: 20, model: gemini-2.5-flash

Figure 40: Gemini AI Exceeded Quota and Not Generating AI Summaries

Solutions

1. The solution to first challenge was to recreate back the Looker Studio Dashboard. Then reuse the same activity_main.xml, MainActivity.java and AndroidManifest.xml codes in the same project file.

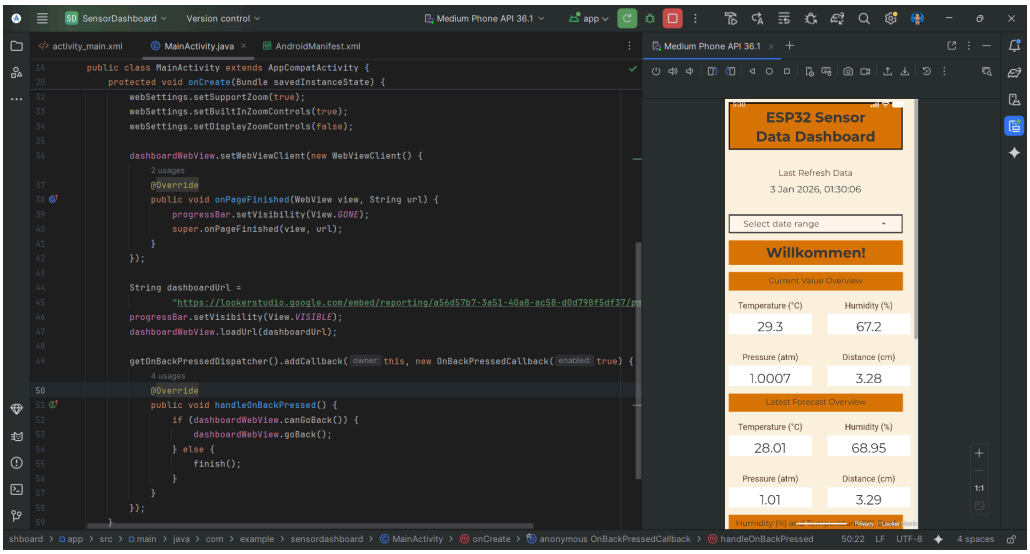


Figure 41: App No Longer Appear White in Android Studio.

2. The solution to the second challenge was refreshing the Looker Studio page and changing manage access from public to restricted and rechange it back to public access that allow people with link to edit and view.

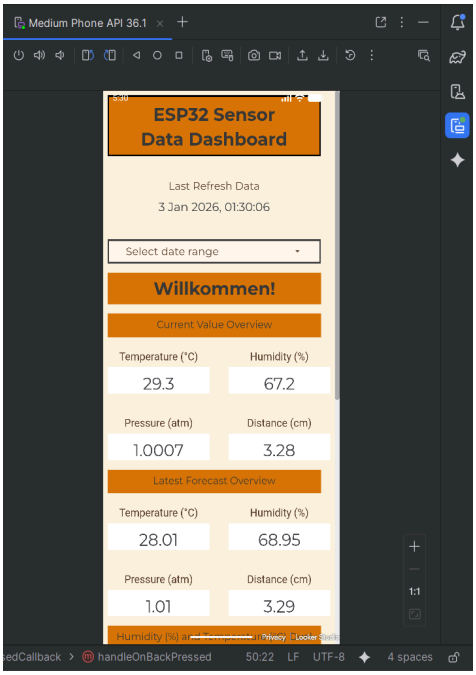


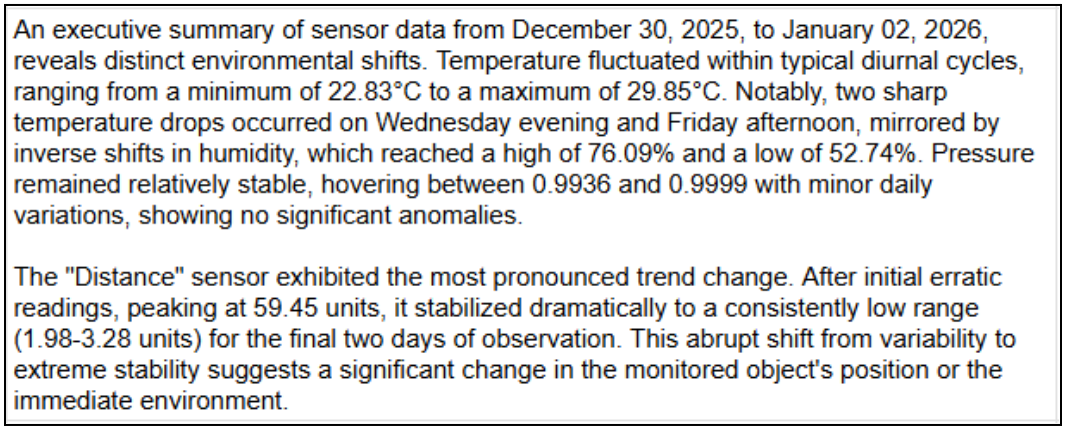
Figure 42: Error Message No Longer Appear in App.

3. The solution to third challenge was implementing a second forecast algorithm into the App Script which is Exponential Moving Average (EMA) to the distance data only and resulted in more normal forecasted data display.

Distance Forecast	Distance Upper	Distance Lower
24.12	26.53	21.71
24.33	26.76	21.9
24.81	27.29	22.33
25.03	27.53	22.52
24.94	27.43	22.44
24.5	26.95	22.05
24.98	27.48	22.48
25.47	28.01	22.92
25.32	27.85	22.79
25.69	28.26	23.12
25.48	28.02	22.93
25.86	28.44	23.27
25.91	28.5	23.32

Figure 43: The Forecasted Distance Data No Longer Generate Negative Values.

4. The solution to fourth challenge was wait for the Gemini AI quota to reset after a day it exceeded quota and press the Generate AI Summaries again thus resulted an expected AI Summaries generated into the sheets.


An executive summary of sensor data from December 30, 2025, to January 02, 2026, reveals distinct environmental shifts. Temperature fluctuated within typical diurnal cycles, ranging from a minimum of 22.83°C to a maximum of 29.85°C. Notably, two sharp temperature drops occurred on Wednesday evening and Friday afternoon, mirrored by inverse shifts in humidity, which reached a high of 76.09% and a low of 52.74%. Pressure remained relatively stable, hovering between 0.9936 and 0.9999 with minor daily variations, showing no significant anomalies.

The "Distance" sensor exhibited the most pronounced trend change. After initial erratic readings, peaking at 59.45 units, it stabilized dramatically to a consistently low range (1.98-3.28 units) for the final two days of observation. This abrupt shift from variability to extreme stability suggests a significant change in the monitored object's position or the immediate environment.

Figure 44: Gemini AI No Longer Exceeded Quota and Generated the Prompted Summaries.

Results

1. Hardware Setup.

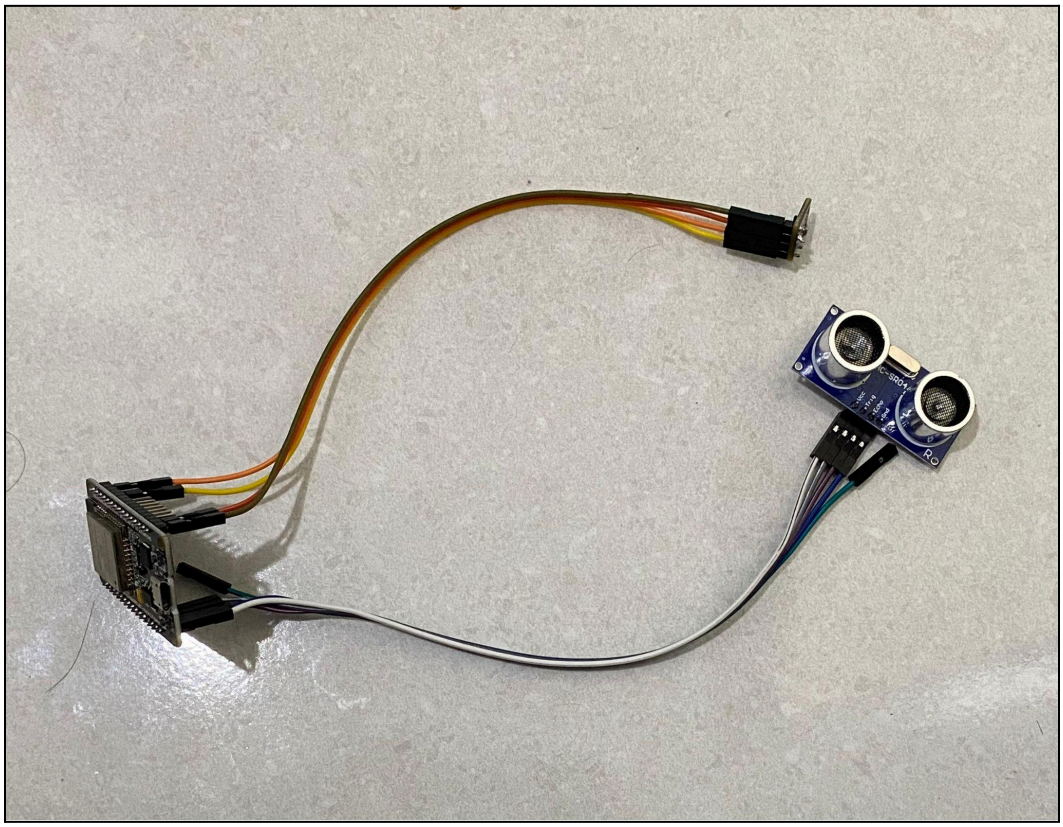


Figure 45: BME280 Temperature, Humidity and Pressure Sensor and HC-SR04 Ultrasonic Sensor Hardware Connection to ESP32.

Figure 45 shows BME280 Temperature, Humidity and Pressure Sensor and HC-SR04 Ultrasonic Sensor are connected to the ESP32 just using female to female wire connector.

2. Google Sheet.

   The ESP32 Sensor Data Logger has 7 different sheet that stored various different data and only 5 sheet was insert to Looker Studio which is SensorData, Forecast_24H, RealTimeDataAnalysis, TrendAnalysis and PredictiveInsight_24H sheets.

	A	B	C	D	E
1	Timestamp	Temperature (°C)	Humidity (%)	Pressure (atm)	Distance (cm)
2	30/12/2025 15:05:14	29.3	71.42	1.00	28.24
3	30/12/2025 15:35:13	29.51	68.88	1.00	59.45
4	30/12/2025 16:23:43	29.65	67.64	0.99	2.71
5	30/12/2025 16:53:43	29.28	66.95	0.99	4.44
6	30/12/2025 17:23:43	29.34	66.88	0.99	3.59
7	30/12/2025 17:53:44	29.37	67.02	0.99	3.75
8	30/12/2025 18:23:42	29.24	67.12	0.99	3.7
9	30/12/2025 18:53:43	29.18	67.05	1.00	4.29
10	30/12/2025 19:23:42	29.11	67.39	1.00	4.86
11	30/12/2025 19:53:42	29	69.09	1.00	2.96
12	30/12/2025 20:23:43	28.97	69.76	1.00	2.62
13	30/12/2025 20:53:42	28.78	70.75	1.00	2.62
14	30/12/2025 21:23:42	28.81	70.97	1.00	2.62
15	30/12/2025 21:53:46	28.78	71.42	1.00	15.23
16	30/12/2025 22:23:41	28.67	72.44	1.00	37.03
17	30/12/2025 22:53:41	28.69	73.6	1.00	37.03
18	30/12/2025 23:23:41	28.63	73.77	1.00	27.66
19	30/12/2025 23:53:45	28.58	73.8	1.00	18.78
20	31/12/2025 00:23:41	28.46	73.82	1.00	18.78
21	31/12/2025 00:53:40	28.39	73.65	1.00	18.78
22	31/12/2025 01:23:40	28.41	73.38	1.00	18.78
23	31/12/2025 01:53:40	28.38	73.63	1.00	18.31
24	31/12/2025 02:23:41	28.28	73.84	1.00	18.12
25	31/12/2025 02:53:41	28.16	73.97	1.00	17.45
26	31/12/2025 03:23:40	28.05	73.95	1.00	16.59
27	31/12/2025 03:53:40	28	73.94	1.00	16.21
28	31/12/2025 04:23:40	27.96	74.04	1.00	16.11
29	31/12/2025 04:53:40	27.92	74.22	1.00	16.02
30	31/12/2025 05:23:39	27.9	74.47	1.00	16.5
31	31/12/2025 05:53:40	27.87	74.7	1.00	16.98
32	31/12/2025 06:23:39	27.86	74.87	1.00	16.88
33	31/12/2025 06:53:39	27.84	75.03	1.00	16.21
34	31/12/2025 07:23:39	27.83	75.29	1.00	16.4
35	31/12/2025 07:53:40	27.83	75.58	1.00	15.93
36	31/12/2025 08:23:39	27.78	75.77	1.00	16.4
37	31/12/2025 08:53:39	27.78	75.94	1.00	16.12
38	31/12/2025 09:23:39	27.79	76.09	1.00	16.79
39	31/12/2025 09:53:38	27.84	76.08	1.00	17.26
40	31/12/2025 10:23:39	27.95	75.87	1.00	18.12
41	31/12/2025 10:53:38	28.1	75.78	1.00	18.69
42	31/12/2025 11:23:37	28.32	75.86	1.00	18.78
43	31/12/2025 11:53:38	28.56	75.43	1.00	18.78
44	31/12/2025 12:23:37	28.74	75.26	1.00	25.89
45	31/12/2025 12:53:38	28.87	74.71	1.00	34.23
46	31/12/2025 13:23:37	29.03	74	1.00	39.74
47	31/12/2025 13:53:39	29.39	73.3	1.00	42.98
48	31/12/2025 14:23:37	29.65	72.14	1.00	57.3
49	31/12/2025 14:53:36	29.74	70.71	0.99	43.1

Figure 46: SensorData Sheet.

In Figure 46, the SensorData sheet receive every 30 minute average data from ESP32. The sheet consist of 5 column which stored timestamp, temperature data, humidity data, pressure data and distance data.

1	Timestamp	Temperature Forecast	Temperature Upper	Temperature Lower	Humidity Forecast	Humidity Upper	Humidity Lower	Pressure Forecast	Pressure Upper	Pressure Lower	Distance	Distance Upper	Distance Lower	Alert
2	03/01/2026 00:51:26	26.87	29.56	24.18	68.65	75.52	61.79	1	1.1	0.9	3.34	3.67	3	OK

Figure 47: Forecast_1H Sheet.

In Figure 47, the Forecast_1H sheet stored only the first 1 hour of forecasted data from SensorData sheet from ESP32. The sheet consist of 14 column which stored timestamp, temperature forecast data, temperature upper data, temperature lower data, humidity forecast data, humidity upper data, humidity lower data, pressure forecast data, pressure upper data, pressure lower data, distance forecast data, distance upper data, distance lower data and alert data.

Timestamp	Temperature Forecast	Temperature Upper	Temperature Lower	Humidity Forecast	Humidity Upper	Humidity Lower	Pressure Forecast	Pressure Upper	Pressure Lower	Distance	Distance Upper	Distance Lower	Alert
03/01/2026 00:51:26	26.87	29.56	24.18	68.65	75.52	61.79	1	1.1	0.9	3.26	3.58	2.93	OK
03/01/2026 01:51:26	26.91	29.6	24.22	69.29	76.21	62.36	1	1.1	0.9	3.27	3.6	2.95	OK
03/01/2026 02:51:26	26.95	29.65	24.26	69.92	76.91	62.93	1	1.1	0.9	3.29	3.62	2.96	OK
03/01/2026 03:51:26	27	29.69	24.3	70.56	77.61	63.5	1	1.1	0.9	3.24	3.57	2.92	OK
03/01/2026 04:51:26	27.04	29.74	24.33	71.19	78.31	64.07	1	1.1	0.9	3.31	3.64	2.98	OK
03/01/2026 05:51:26	27.08	29.79	24.37	71.83	79.01	64.64	1	1.1	0.9	3.36	3.7	3.03	OK

Figure 48: Forecast_6H Sheet.

In Figure 48, the Forecast_6H sheet are also the same as Forecast_1H but the sheet stored only the first 6 hour of forecasted data from SensorData sheet from ESP32. The sheet too consist of 14 column which stored timestamp, temperature forecast data, temperature upper data, temperature lower data, humidity forecast data, humidity upper data, humidity lower data, pressure forecast data, pressure upper data, pressure lower data, distance forecast data, distance upper data, distance lower data and alert data.

Timestamp	Temperature Forecast	Temperature Upper	Temperature Lower	Humidity Forecast	Humidity Upper	Humidity Lower	Pressure Forecast	Pressure Upper	Pressure Lower	Distance Forecast	Distance Upper	Distance Lower	Alert
31/12/2025 00:23:41	28.52	31.37	25.67	74.01	81.41	66.61	1	1.1	0.9	24.12	26.53	21.71	OK
31/12/2025 01:23:41	28.44	31.29	25.6	74.84	82.32	67.36	1	1.1	0.9	24.33	26.76	21.9	OK
31/12/2025 02:23:41	28.37	31.2	25.53	75.67	83.24	68.1	1	1.1	0.9	24.81	27.29	22.33	OK
31/12/2025 03:23:41	28.29	31.12	25.46	76.5	84.15	68.85	1	1.1	0.9	25.03	27.53	22.52	OK
31/12/2025 04:23:41	28.21	31.03	25.39	77.33	85.07	69.6	1	1.1	0.9	24.94	27.43	22.44	OK
31/12/2025 05:23:41	28.14	30.95	25.32	78.17	85.98	70.35	1	1.1	0.9	24.5	26.95	22.05	OK
31/12/2025 06:23:41	28.06	30.86	25.25	79	86.9	71.1	1	1.1	0.9	24.98	27.48	22.48	OK
31/12/2025 07:23:41	27.98	30.78	25.18	79.83	87.81	71.85	1	1.1	0.9	25.47	28.01	22.92	OK
31/12/2025 08:23:41	27.9	30.7	25.11	80.66	88.73	72.59	1	1.1	0.9	25.32	27.85	22.79	Hum
31/12/2025 09:23:41	27.83	30.61	25.04	81.49	89.64	73.34	1	1.1	0.9	25.69	28.26	23.12	Hum
31/12/2025 10:23:41	27.75	30.53	24.98	82.32	90.56	74.09	1	1.1	0.9	25.48	28.02	22.93	Hum
31/12/2025 11:23:41	27.67	30.44	24.91	83.16	91.47	74.84	1	1.1	0.9	25.86	28.44	23.27	Hum
31/12/2025 12:23:41	27.6	30.36	24.84	83.99	92.39	75.59	1	1.1	0.9	25.91	28.5	23.32	Hum
31/12/2025 13:23:41	27.52	30.27	24.77	84.82	93.3	76.34	1	1.1	0.9	25.63	28.19	23.07	Hum
31/12/2025 14:23:41	27.44	30.19	24.7	85.65	94.22	77.09	1	1.1	0.9	25.28	27.81	22.75	Hum
31/12/2025 15:23:41	27.37	30.1	24.63	86.48	95.13	77.83	1	1.1	0.9	25.78	28.36	23.2	Hum
31/12/2025 16:23:41	27.29	30.02	24.56	87.31	96.04	78.58	1	1.1	0.9	26.11	28.72	23.5	Hum
31/12/2025 17:23:41	27.21	29.93	24.49	88.14	96.96	79.33	1	1.1	0.9	26.26	28.88	23.63	Hum
31/12/2025 18:23:41	27.13	29.85	24.42	88.98	97.87	80.08	1	1.1	0.9	25.87	28.46	23.28	Hum
31/12/2025 19:23:41	27.06	29.76	24.35	89.81	98.79	80.83	1	1.1	0.9	26.09	28.69	23.48	Hum
31/12/2025 20:23:41	26.98	29.68	24.28	90.64	99.7	81.58	1	1.1	0.9	26.47	29.12	23.83	Hum
31/12/2025 21:23:41	26.9	29.59	24.21	91.47	100.62	82.32	1	1.1	0.9	26.26	28.88	23.63	Hum
31/12/2025 22:23:41	26.83	29.51	24.14	92.3	101.53	83.07	1	1.1	0.9	26.12	28.73	23.51	Hum
31/12/2025 23:23:41	26.75	29.42	24.07	93.13	102.45	83.82	1	1.11	0.9	25.61	28.18	23.05	Hum

Figure 49: Forecast_24H Sheet.

In Figure 49, the Forecast_24H sheet stored only the 24 hour of forecasted data from SensorData sheet from ESP32. The sheet too consist of 14 column which stored timestamp, temperature forecast data, temperature upper data, temperature lower data, humidity forecast data, humidity upper data, humidity lower data, pressure forecast data, pressure upper data, pressure lower data, distance forecast data, distance upper data, distance lower data and alert data.

Timestamp	Sensor Analysis
All Sensors	Here's an analysis of the sensor data: | Sensor Type | Combined Summary | | :---------- | :---------- --- ---------- | | **Timestamps** | 3 days of data (Dec 30 15:05 to Jan 2 15:51 SGT), recorded consistently ~30-minute intervals, providing regular, high-frequency environmental sampling. | | **Temperature** | Diurnal cycles (27.78-29.85°C) precede a sharp Dec 31 drop from 29.85°C to 22.83°C. Recovery with overnight dips suggests active localized cooling or environmental system activation. | | **Humidity** | Humidity inversely correlates with temperature (52.74-76.09%). A significant Dec 31 drop (69.16% to 52.74%) mirrors temperature, strongly implying a drying event, likely environmental controls. | | **Pressure** | Minimal pressure fluctuation (0.9936-0.9999) with gentle diurnal patterns. Stability suggests localized temperature/humidity changes, not broad atmospheric shifts. | | **Distance** | Initially dynamic (2.71-59.45), distance sharply dropped from 51.1 to 1.98 on Dec 31, coinciding with environmental shifts. It then stabilized at ~3.28, indicating an object moved close and remained fixed. |

Figure 50: RealTimeDataAnalysis Sheet.

In Figure 50, the RealTimeDataAnalysis sheet stored a row of analyzed summary data which is response that was prompted to Gemini AI API to analyze all data of SensorData sheet from ESP32. The sheet only consist 2 column which stored timestamp of all sensors and sensor analysis data.

Timestamp	Sensor Trend Analysis
All Sensors	An executive summary of sensor data from December 30, 2025, to January 02, 2026, reveals distinct environmental shifts. Temperature fluctuated within typical diurnal cycles, ranging from a minimum of 22.83°C to a maximum of 29.85°C. Notably, two sharp temperature drops occurred on Wednesday evening and Friday afternoon, mirrored by inverse shifts in humidity, which reached a high of 76.09% and a low of 52.74%. Pressure remained relatively stable, hovering between 0.9936 and 0.9999 with minor daily variations, showing no significant anomalies. The "Distance" sensor exhibited the most pronounced trend change. After initial erratic readings, peaking at 59.45 units, it stabilized dramatically to a consistently low range (1.98-3.28 units) for the final two days of observation. This abrupt shift from variability to extreme stability suggests a significant change in the monitored object's position or the immediate environment.

Figure 51: TrendAnalysis Sheet.

In Figure 51, he TrendAnalysis sheet also stored a row of analyzed trend summary data which is response that was prompted to the same Gemini AI API to analyze data trend of SensorData sheet from ESP32. The sheet too only consist 2 column which stored timestamp of all sensors and sensor trend analysis data.

Timestamp	Predictive Insight
All Sensors	For the 24-hour period beginning Wednesday, December 31, 2025, in Singapore, expect a day of gradually cooling temperatures and significantly rising humidity. Starting around 28.5°C, temperatures will steadily decline to approximately 26.8°C by late evening. Humidity, however, will surge from 74% to over 93%, indicating increasingly muggy conditions and a heightened potential for precipitation, especially as the day progresses. Atmospheric pressure is forecast to remain stable, suggesting no major weather system movements. 'Distance' (likely visibility) will see minor fluctuations, generally remaining between 24 and 26 units, pointing to consistent ambient conditions. A critical environmental shift is projected immediately following this 24-hour window. Overnight into January 1st, temperatures are set to dramatically drop to around 21°C and continue falling, paired with a sharp decrease in humidity to below 50% and a substantial reduction in 'distance' to just 2 units, signaling a sudden, significant change in weather.

Figure 52: PredictiveInsight_24H Sheet.

In Figure 52, the PredictiveInsight_24H sheet stored a row of predictive insight summary data which is response that was prompted to the same Gemini AI API to predict the 24 hours forecast data of Forecast_24H sheet given from the forecast function. The sheet too only consist 2 column which stored timestamp of all sensors and sensor trend analysis data.

3. Looker Studio Dashboard.
   The dashboard is separated into 6 pages. First page is named Homepage that displays the overview of every sensors data.

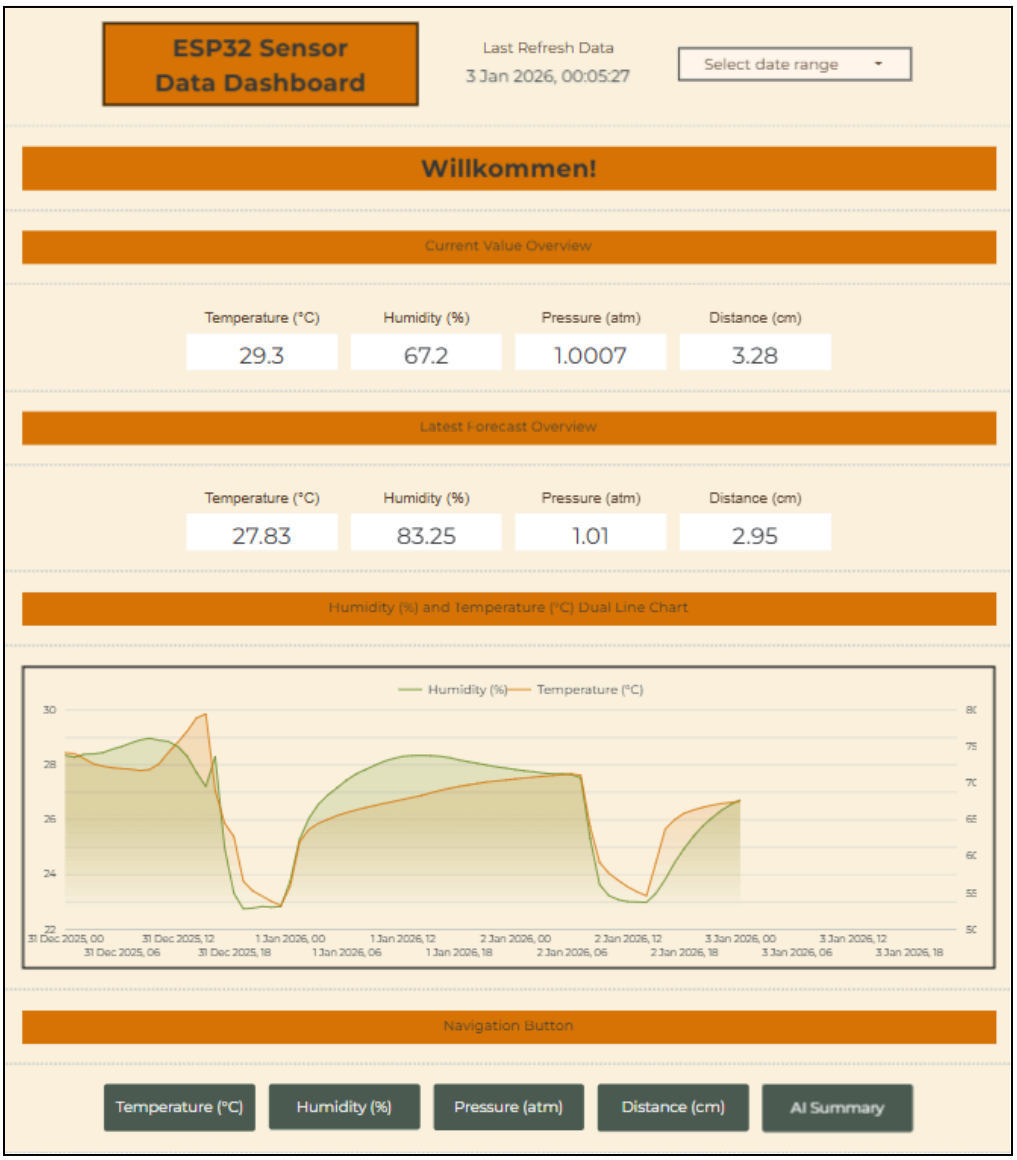


Figure 53: Homepage Dashboard.

Figure 53 is the first page that only displays title named “ESP32 Sensor Data Dashboard”, a date range control, 8 separate metrics scorecard which displays temperature current value and latest forecast, humidity current value and latest forecast, pressure current value and latest forecast, distance current value and latest forecast, a dual axis chart of temperature and humidity data and 5 buttons which navigates to Temperature (°C), Humidity (%), Pressure (atm) Distance (cm) and AI Summary pages.

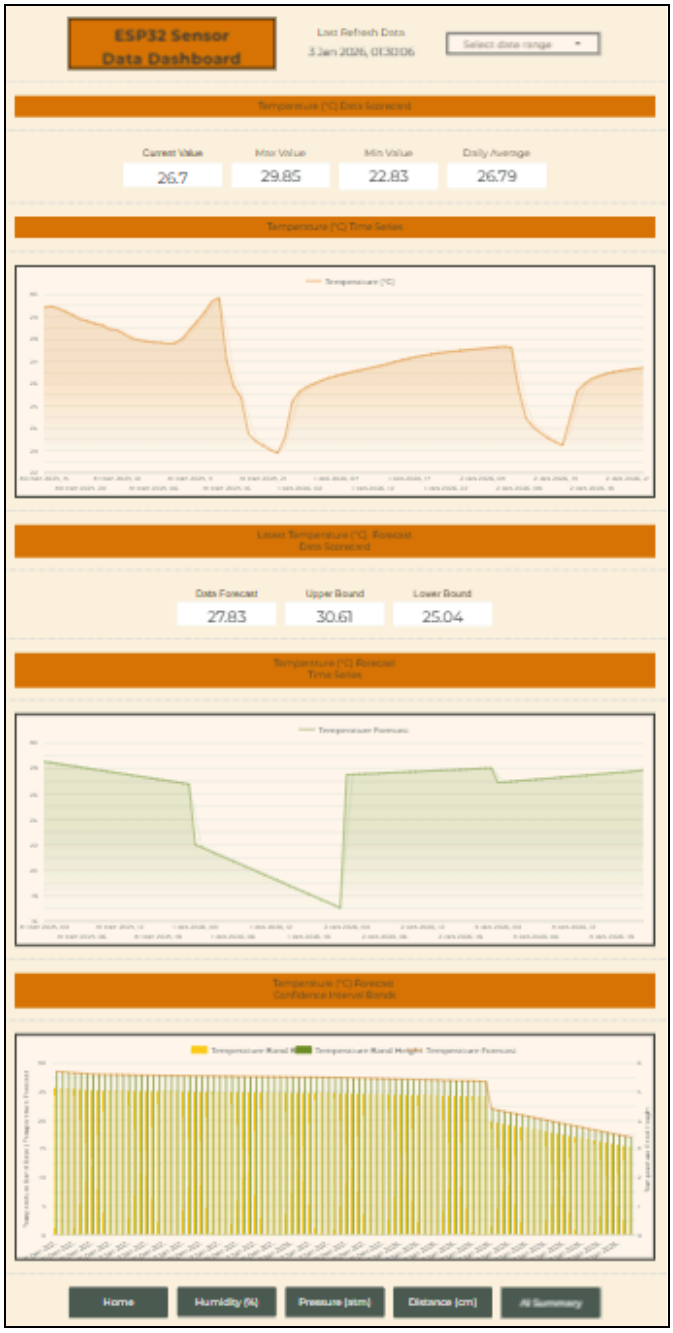


Figure 54: Temperature (°C) Dashboard.

For Figure 54 as the second page also displays title named “ESP32 Sensor Data Dashboard”, a date range control, 7 separate metrics scorecard which displays temperature current value, maximum value, minimum value and daily average and temperature forecast data, upper bound and lower bound, 2 time series chart of historical temperature data and temperature forecast data, a combo chart of temperature confidence interval bands and 5 buttons which navigates to Home, Humidity (%), Pressure (atm), Distance (cm) and AI Summary pages.

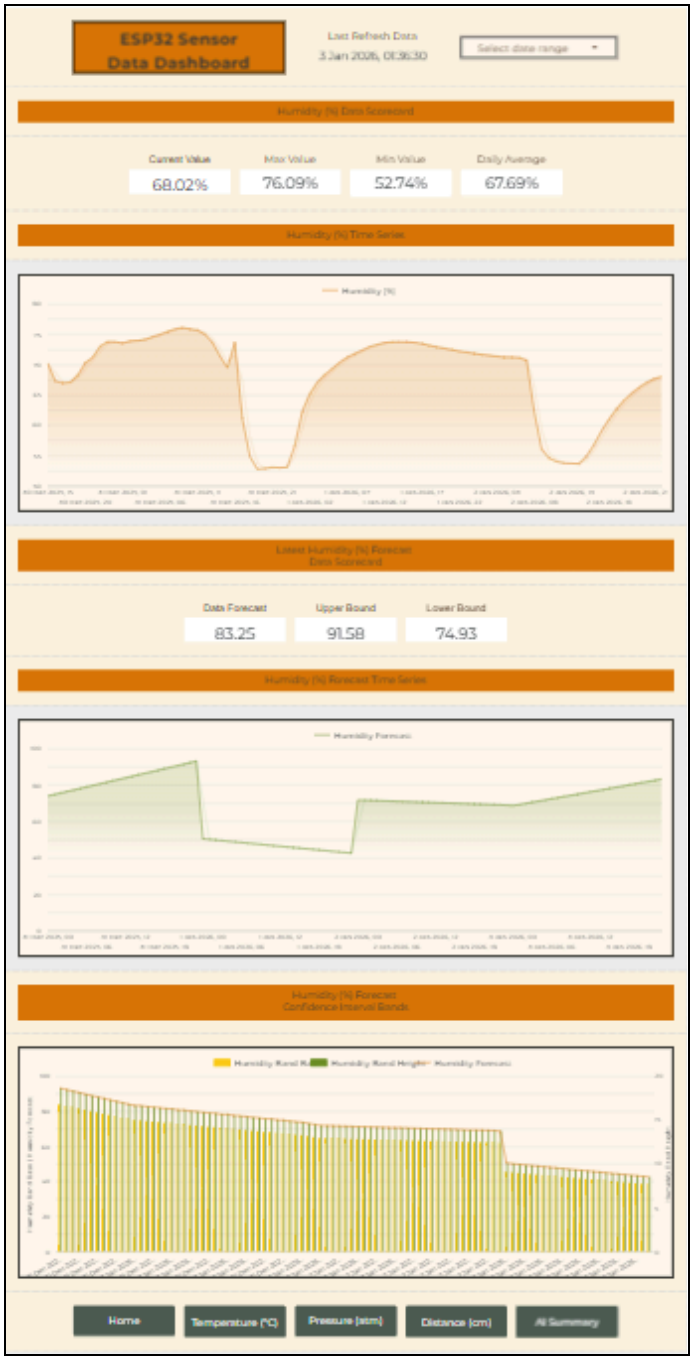


Figure 55: Humidity (%) Dashboard.

For Figure 55 which is the third page also displays the same title, a date range control, 7 separate metrics scorecard which displays humidity current value, maximum value, minimum value and daily average and humidity forecast data, upper bound and lower bound, 2 time series chart of historical humidity data and humidity forecast data, a combo chart of humidity confidence interval bands and 5 buttons which navigates to Home, Temperature (°C), Pressure (atm), Distance (cm) and AI Summary pages.

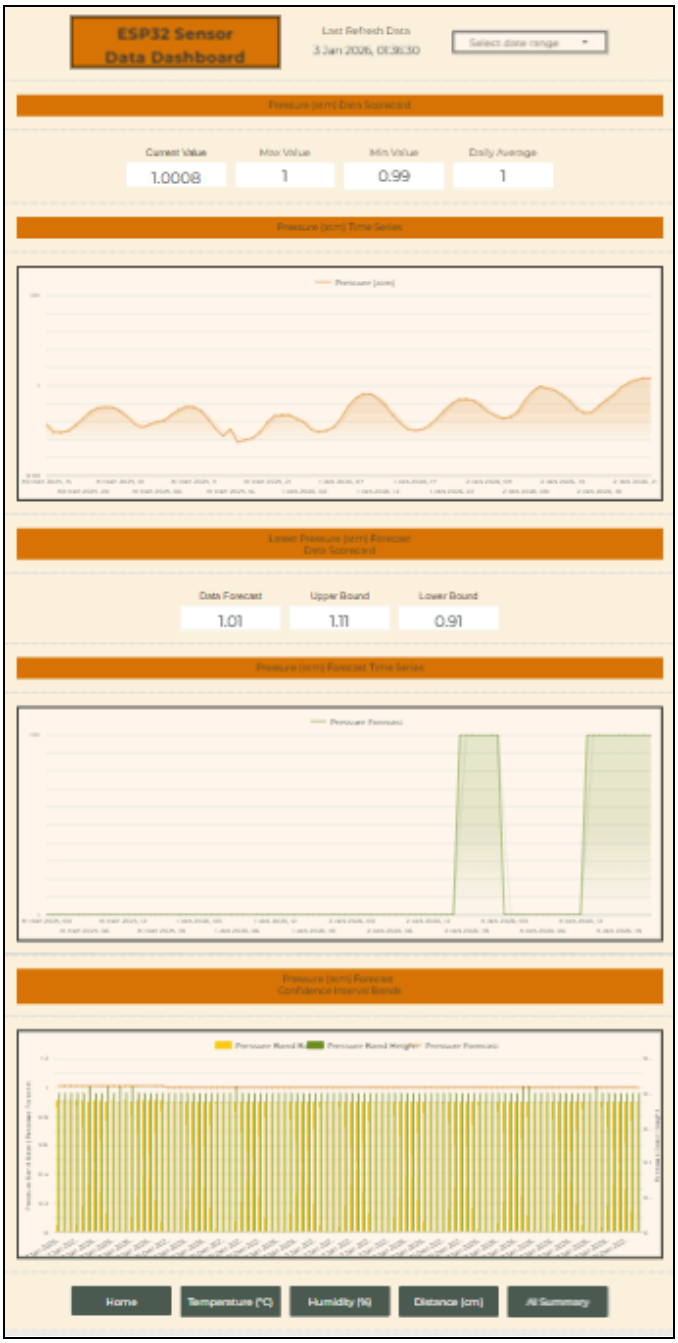


Figure 56: Pressure (atm) Dashboard.

At Figure 56 which is the fourth page also displays title named “ESP32 Sensor Data Dashboard”, a date range control, 7 separate metrics scorecard which displays pressure current value, maximum value, minimum value and daily average and pressure forecast data, upper bound and lower bound, 2 time series chart of historical pressure data and pressure forecast data, a combo chart of pressure confidence interval bands and 5 buttons which navigates to Home, Temperature (°C), Humidity (%), Distance (cm) and AI Summary pages.

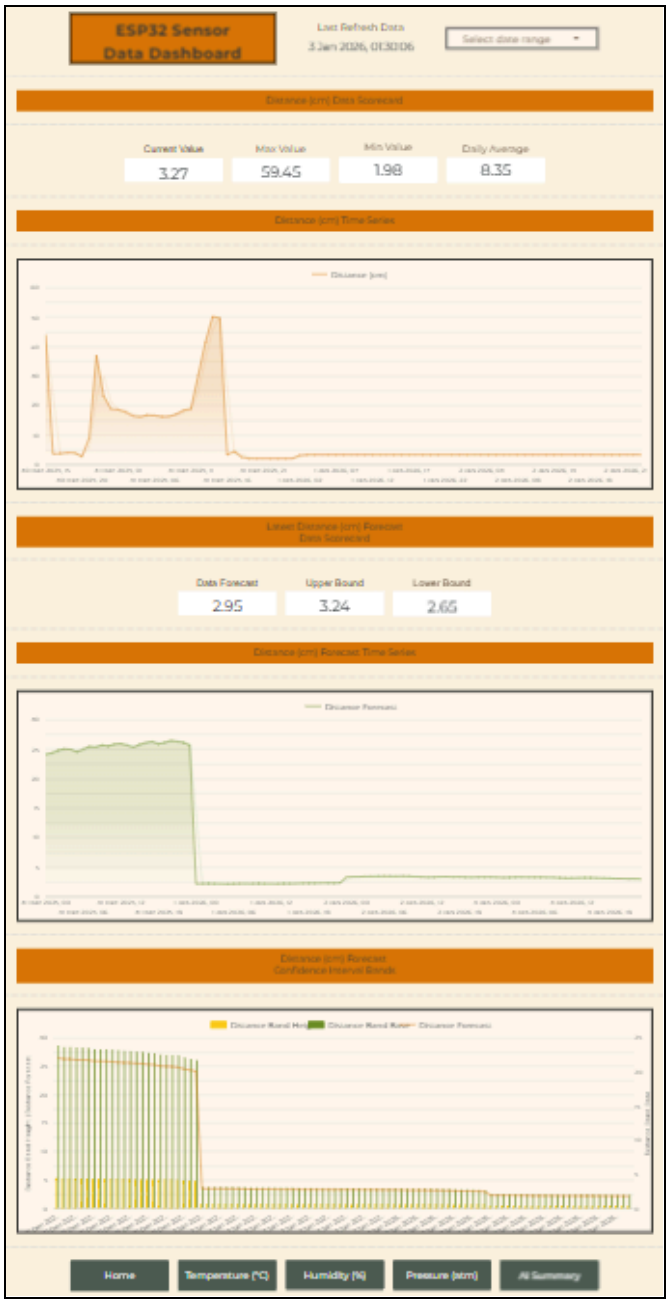


Figure 57: Distance (cm) Dashboard.

Figure 57 is the fifth page also displays title named “ESP32 Sensor Data Dashboard”, a date range control, 7 separate metrics scorecard which displays distance current value, maximum value, minimum value and daily average and distance forecast data, upper bound and lower bound, 2 time series chart of historical distance data and distance forecast data, a combo chart of distance confidence interval bands and 5 buttons which navigates to Home, Temperature (°C), Humidity (%), Pressure (atm) and AI Summary pages.

Figure 58: AI Summary Dashboard.

Figure 58 is the sixth page displays title named "ESP32 Sensor Data Dashboard", a date range control, 3 separate table which displays real time data analysis summary, trend analysis summary, predictive insight summary and 5 buttons which navigates to Home, Temperature (°C), Humidity (%), Pressure (atm) and Distance (cm) pages.

4. Android Studio Dashboard App.

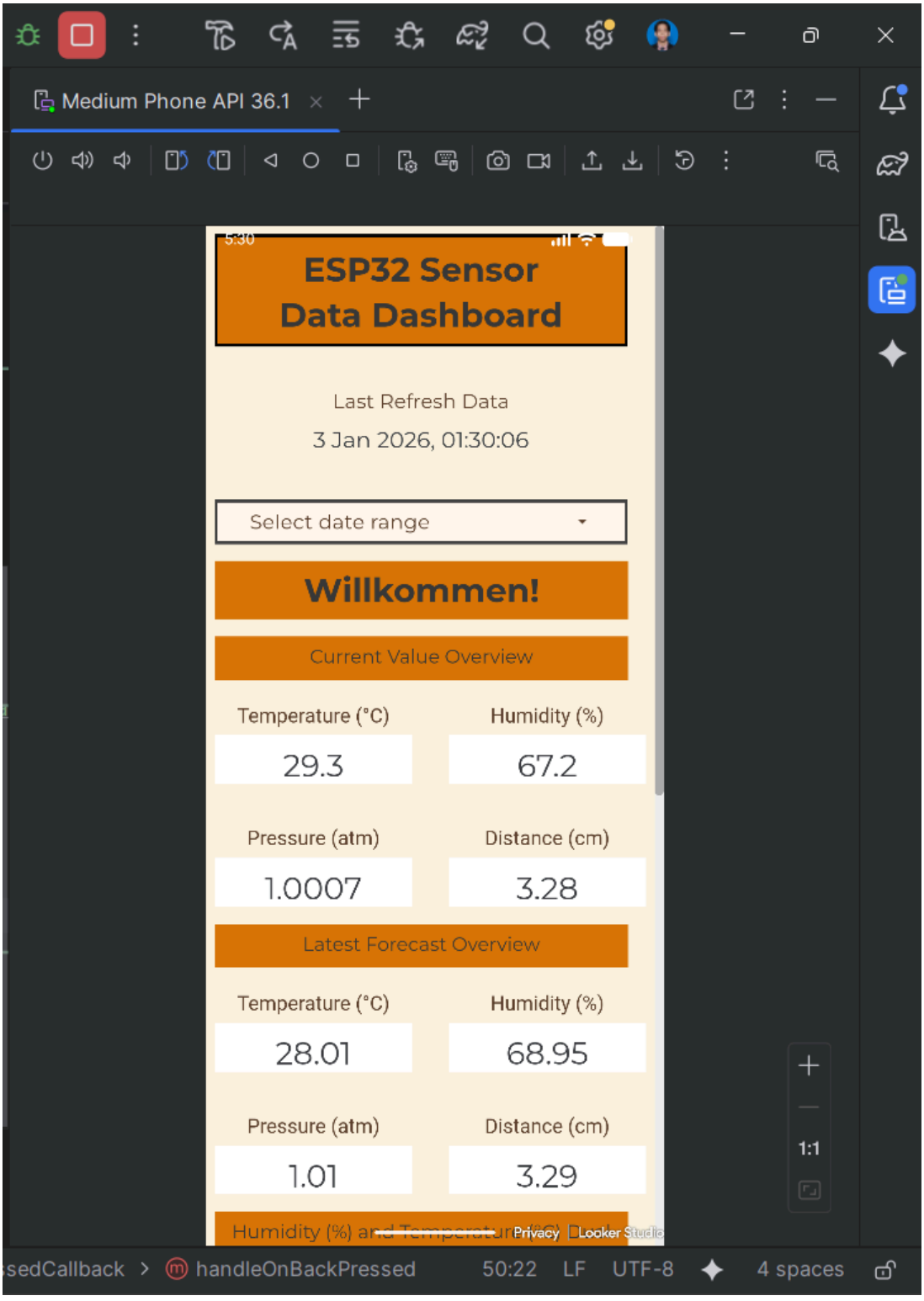


Figure 59: ESP32 Sensor Data Logger Dashboard App Running in Android Studio.

Collected Data Analysis

The sensors data reading collected from the ESP32 includes temperature (°C), humidity (%), pressure (atm) and distance (cm) that were then logged into Google Sheets and visualized through Looker Studio dashboards. The recording period from 30 December 2025 to 3 January 2026 shows normal sensor behavior with small variations across all 4 parameters.

1. Temperature Data Analysis and Insight:

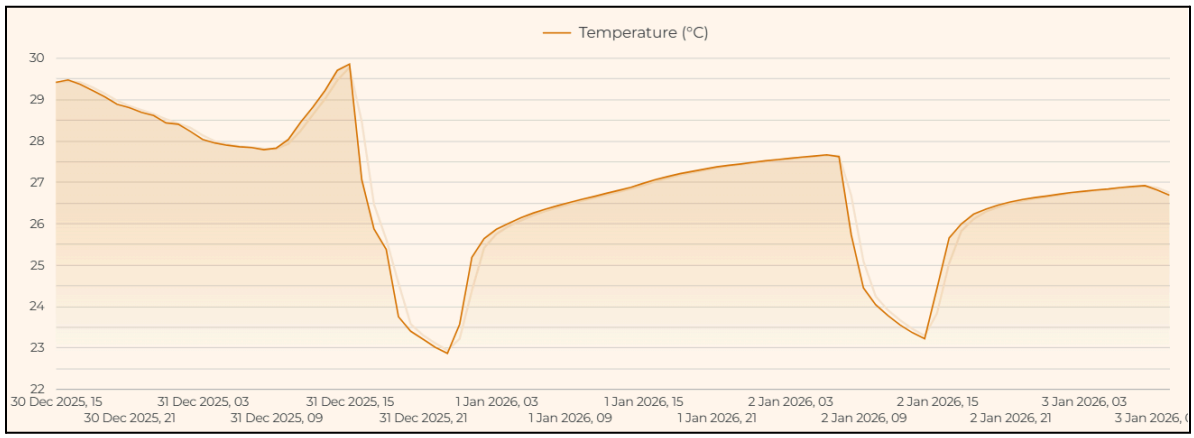


Figure 60: Temperature Data Line Chart.

In Figure 60, the temperature began ranged 29.41°C before decreasing steadily to around 27.78°C. A noticeable rise occurred during 31 December 2025 and drop back from 29.85°C to 22.85°C due to a sudden change in surrounding airflow or heat from nearby components. The second sudden drop occurred on 2 December 2025 shows a significant environmental change such as cooling or ventilation activation. The data shown is stable in overall which suggests that the sensor is performing reliably.

2. Humidity Data Analysis and Insight:

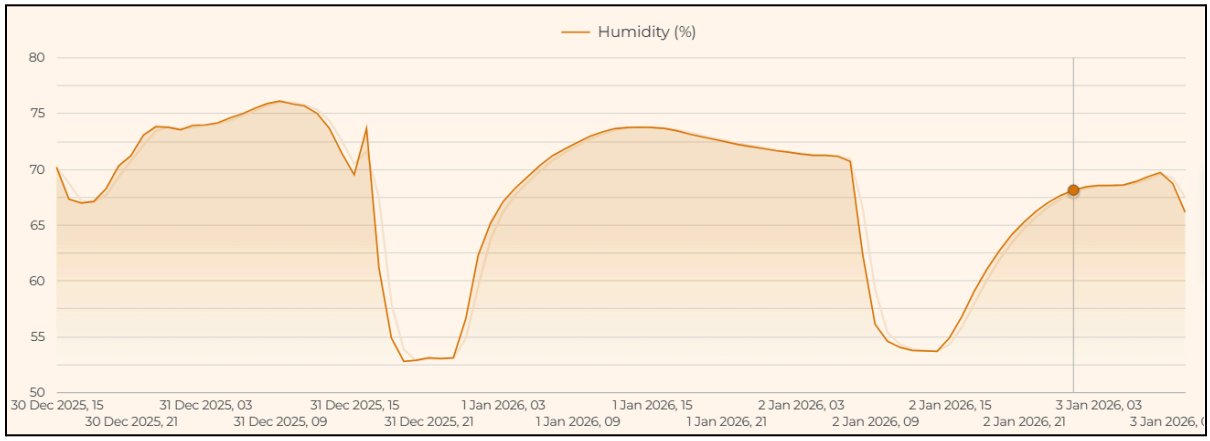


Figure 61: Humidity Data Line Chart.

In Figure 61, humidity readings were ranged between 70.15% to 76.09%. The humidity peak gradually when temperature reached its lowest. This chart shows the expected inverse relationship between temperature and humidity which was when temperature decreased, humidity will increased. The data remained smooth and stable thus demonstrating reliable sensor behaviour with minimal external disturbances.

3. Pressure Data Analysis and Insight.

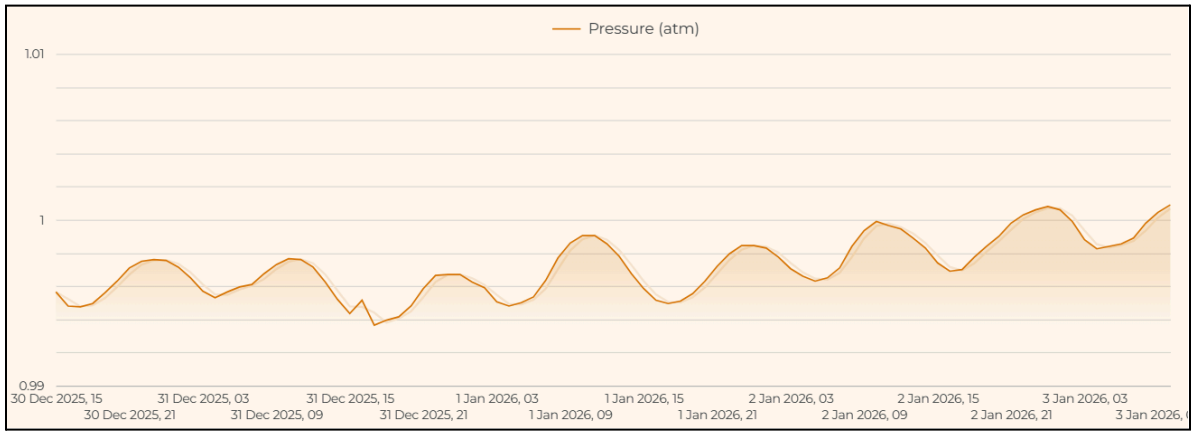


Figure 62: Pressure Data Line Chart

In Figure 62, atmospheric pressure drop and rise between 0.9958 atm and 1.0008 atm which shows in sinus wave lookalike. The rise and drop that were observed every day suggests natural environmental or weather changes, such as approaching low pressure conditions. The readings appeared clean and consistent, with no unusual spikes, confirming accurate pressure measurement.

4. Distance Data Analysis and Insight:

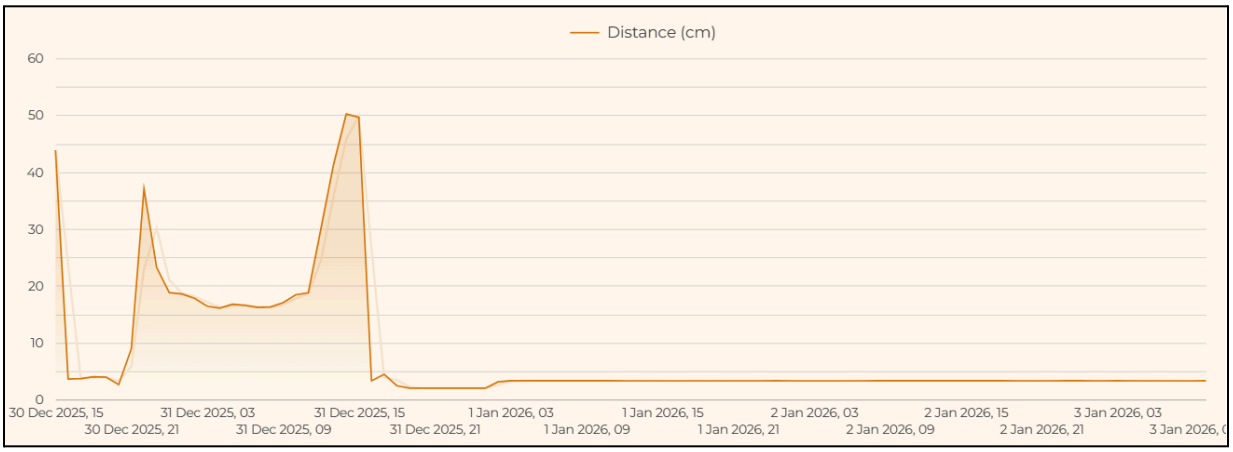


Figure 63: Distance Data Line Chart.

In Figure 63, multiple spikes that reached up to 50.2 cm started in the first 2 days and the distance steadily ranged between 4.43 cm to 3.28 cm the next day. The reading that dropped, stabilized and spiked indicates the measured object was moved very far or close to the sensor. Despite the noise, the core measurements suggested that the object being monitored generally remained at a fixed and stable distance from the sensor after the multiple spike at the beginning.

Future Improvements

Future improvement for this system is expanding the AI powered summarization functionality. Currently, the Gemini AI generates real time data analysis, trend analysis, and predictive insights. Future enhancements include fully automated daily or weekly report generation, anomaly explanation and confidence scoring to indicate the reliability of forecasts. This would further reduce manual analysis and provide users with more actionable insights.

Another potential future improvement is the addition of real time alerting and notification features. Threshold based alerts deliver to email or mobile notifications could notify users of abnormal sensor readings or forecasted risks and allow users to configure thresholds and reporting preferences directly through the dashboard or mobile application would enhance system flexibility.

GitHub Repositories Link

https://github.com/Kepengan/TSO2_MiniProject