



# Unidad Educativa Antioquía

## Proyecto EKKO

### Desarrollo de Proyecto

Juan Esteban Cabrera Galarza  
Fundador

Uriel José Cobo-Cuisana  
Fundador

Sistema de monitoreo ambiental.

PUNUNO – MISAHUALLÍ – TENA - ECUADOR

[secretaria@academiaantioquia.edu.ec](mailto:secretaria@academiaantioquia.edu.ec)

Teléfonos: **062890221 - 0992470058**

## Proyecto EKKO:

### Resumen:

En este documento, hemos presentado un proyecto para un dispositivo de monitoreo ambiental que utiliza un microcontrolador Arduino Nano para recopilar datos de sensores ambientales y enviarlos a un receptor para su procesamiento y visualización en una gráfica.

Hemos analizado el mercado actual de dispositivos de monitoreo ambiental y hemos concluido que hay una creciente demanda para dispositivos asequibles y personalizables, lo que hace que nuestro proyecto sea muy viable en términos económicos. Además, hemos identificado nuestras ventajas competitivas, que incluyen un diseño modular, el uso de un microcontrolador Arduino Nano y la capacidad de personalizar la PCB en el futuro.

También hemos proporcionado una descripción detallada de los componentes utilizados en nuestro proyecto, incluyendo sensores ambientales, microcontrolador Arduino Nano, receptor Python y la biblioteca de visualización de datos Matplotlib.

Además, hemos explicado la programación tanto del microcontrolador como del receptor Python, y hemos proporcionado fragmentos de código para ilustrar los procesos involucrados. También hemos discutido los costos involucrados en la fabricación de nuestro dispositivo de monitoreo ambiental y hemos analizado la viabilidad económica del proyecto.



En resumen, nuestro proyecto de dispositivo de monitoreo ambiental ofrece una solución asequible y personalizable para el monitoreo ambiental diferentes ámbitos científicos. Además, el diseño modular y la capacidad de personalizar la PCB en el futuro lo hacen muy atractivo para los usuarios que buscan una solución personalizada. Con un precio total de componentes de aproximadamente \$23.50, el proyecto es muy asequible y tiene el potencial de ser rentable si se fabrica en masa en una PCB personalizada.

### **Introducción:**

El Proyecto EKKO es un sistema de monitoreo y análisis estadístico ambiental diseñado para medir y registrar los Parámetros del medio ambiente en tiempo real. El objetivo principal del proyecto es proporcionar una herramienta útil, intuitiva, expandible y económica para la investigación científica y para el monitoreo y control ambiental en distintos entornos.

### **¿Por qué creamos el proyecto EKKO?**

La creación de este proyecto surge de la necesidad de contar con un sistema de monitoreo y análisis estadístico ambiental de bajo costo, fácil de usar y adaptable a distintas situaciones. La medición de la temperatura, presión, y humedad constituyen un factor preponderante en la investigación. Por ende, en el control de calidad de procesos y productos en la industria. El mismo ofrece una solución accesible y efectiva para medir y registrar estos parámetros de forma constante y en tiempo real.

### **Usos:**



Nuestro proyecto tiene diversos usos en distintos ámbitos científicos, incluyendo la biología, medicina, industria, arqueología, y geografía, entre otros. Algunos de los usos potenciales del proyecto incluyen:

- Estudios de impacto ambiental: Utilizado para monitorear y registrar las condiciones ambientales en áreas de interés para estudios de impacto ambiental, por ejemplo, en zonas de construcción o áreas de desarrollo urbano.
- Investigaciones en biotecnología: Para monitorear y controlar las condiciones ambientales en experimentos de biotecnología, como la producción de alimentos, la generación de biocombustibles y la biorremediación.
- Estudios de microbiología: Para monitorear y controlar las condiciones ambientales en estudios de microbiología, como la investigación de bacterias, hongos y virus.
- Investigación en ecología: Para monitorear y registrar las condiciones ambientales en estudios de ecología, como la dinámica de poblaciones animales, los cambios en la vegetación y la evolución de las comunidades de microorganismos.

### ¿Cuáles son las ventajas de nuestro producto?

**Flexibilidad:** Nuestro proyecto ha sido diseñado para ser utilizado en una variedad de entornos. Esta versatilidad se debe a que el producto es fácilmente expandible y modular, lo que permite a los usuarios añadir las funciones necesarias para satisfacer sus requerimientos específicos.



Para lograr esto, hemos desarrollado una programación que permite a los usuarios personalizar el producto según sus necesidades. De esta forma, el proyecto se convierte en una base expandible que abre las posibilidades del usuario, en lugar de limitarlas.

Nuestra meta es hacer que nuestro proyecto sea accesible para una amplia gama de clientes, para que pueda beneficiar a la mayor cantidad posible de personas.

**Simplicidad:** Otro aspecto destacado de nuestro proyecto es su simplicidad en la recolección de datos y su facilidad para transmitir la información. Hemos diseñado el producto para que la recopilación de datos sea un proceso menos tedioso y más eficiente, lo que aumenta su utilidad para una amplia gama de aplicaciones.

La simplicidad en la recolección de datos se logra mediante una interfaz de usuario intuitiva y fácil de usar. Los usuarios pueden acceder rápidamente a la información que necesitan y registrarla de manera sencilla y precisa. Además, nuestro proyecto está equipado con herramientas de registro automático que permiten recopilar datos de manera eficiente y reducir la intervención humana.

**Facilidad:** Por otro lado, la facilidad para transmitir la información es un aspecto importante para garantizar la utilidad de los datos recopilados. Nuestro proyecto cuenta con un sistema de comunicación claro y preciso, lo que permite a los usuarios transmitir la información de manera efectiva y sin retrasos. Esto es particularmente valioso para aplicaciones en tiempo real, como el monitoreo de en áreas industriales o la supervisión de parámetros en entornos médicos.

### **Teoría y Desarrollo:**



Después de conocer los usos y ventajas del proyecto EKKO, es importante mencionar cómo funciona y cómo se ha desarrollado tecnológicamente para poder cumplir con estos objetivos. Para ello, es necesario detenernos en los aspectos técnicos del proyecto, como el diseño de los sensores, la programación modular y la implementación del Arduino. De esta manera, podremos entender de manera más clara cómo funciona el código del programa y cómo se relaciona con el hardware utilizado.

### **Sensor DHT11**

El sensor DHT11 es un dispositivo electrónico que mide la temperatura y la humedad relativa del aire utilizando un termistor y un sensor capacitivo de humedad.

#### **Temperatura**

El termistor del sensor funciona con un sistema de resistencia eléctrica. Cuenta con un material que reacciona ante las temperaturas del ambiente. Cuando existen altas temperaturas, el material busca expandirse, separando sus moléculas y otorgando un mejor flujo de energía (viceversa con el frío). Este cambio de voltaje es convertido a una señal digital que será interpretado y relacionado posteriormente.

#### **Humedad**

El sensor capacitivo de humedad mide la humedad relativa del aire. Consiste en una malla que atrapa micropartículas de agua dispersas en el aire. La malla tiene un flujo constante de energía que se ve alterado al contacto de estas microgotas. El agua es un buen conductor eléctrico, por lo que acortará las rutas por las que la energía viaje, relacionando así mucha humedad con mayor voltaje. El cambio en la capacitancia se convierte en una señal analógica y luego se digitaliza para su posterior procesamiento.



Ambas señales digitales son procesadas por un microcontrolador o circuito integrado dentro del sensor, y se generan los valores. La precisión de las mediciones puede variar, pero en general el DHT11 tiene una precisión de  $\pm 2^{\circ}\text{C}$  en la medición de la temperatura y  $\pm 5\%$  en la medición de la humedad relativa.

### **Sensor YL69:**

El sensor YL69 es un sensor de humedad de suelo que utiliza dos electrodos que se colocan en el suelo para medir la conductividad eléctrica entre ellos, lo que a su vez se correlaciona con el contenido de humedad del suelo. Cuando se aplica una tensión al sensor, la corriente eléctrica fluye a través del suelo y la conductividad eléctrica entre los electrodos varía en función del contenido de humedad. Posteriormente, se relacionan los datos y el sensor arroja una señal que posteriormente se analizará.

### **Componente NRF24-L01**

El nRF 24L01 es el componente que nos permite transmitir y visualizar la información recolectada desde la estación climática a largas distancias. Utiliza una antena emisora que utiliza ondas electromagnéticas codificadas con la información. Las ondas las recibe una antena receptora como la de una laptop y visualiza la información en tiempo real. El componente emplea una técnica de modulación de espectro ensanchado por salto de frecuencia (FHSS) para evitar interferencias y mejorar la robustez de la señal.

### **Sensor BMP280**

PUNUNO – MISAHUALLÍ – TENA - ECUADOR

[secretaria@academiaantioquia.edu.ec](mailto:secretaria@academiaantioquia.edu.ec)

Teléfonos: **062890221 - 0992470058**



El BMP280 mide la presión barométrica usando un sensor de presión piezoeléctrico. El sensor de presión piezoeléctrico consta de una lámina delgada de material piezoeléctrico, como el cuarzo, que se coloca entre dos capas de metal. Cuando una fuerza externa, como la presión del aire, se aplica a la lámina, esta se deforma y se produce una carga eléctrica en las capas de metal. Esta carga eléctrica es proporcional a la fuerza aplicada y puede ser medida para determinar la presión. El sensor de presión mide la presión atmosférica y convierte la señal en un valor digital.

### **Microprocesador Arduino Nano:**

El microprocesador Arduino Nano es la base tecnológica de este proyecto, ya que es el encargado de procesar la información de los sensores y tomar decisiones en tiempo real. La información recopilada por los sensores se almacena en una tarjeta SD que se conecta al Arduino Nano, permitiendo así el registro y análisis de los datos recolectados en el futuro. Además, el Arduino Nano se comunica de forma inalámbrica con otros dispositivos mediante el uso de un transceptor NRF24L01, lo que permite la transmisión de datos a larga distancia.

El Arduino Nano es programado utilizando el lenguaje de programación C++, y se utiliza una metodología modular para facilitar el proceso de desarrollo y mantenimiento del código. La programación se divide en distintos módulos que se encargan de realizar diferentes tareas, como la adquisición de datos, el control de los sensores, la transmisión de datos y el almacenamiento en la tarjeta SD. El uso de esta metodología modular hace que el código sea más fácil de entender y mantener, lo que reduce los tiempos de desarrollo y mejora la eficiencia del proyecto.





En el futuro, tenemos planes de optimizar nuestro producto y mejorar su calidad mediante el uso de una placa de circuito impreso personalizada en lugar del Arduino Nano. Esto permitirá un mejor control sobre los componentes electrónicos y reducirá el tamaño del dispositivo. Además, al usar componentes específicos, podremos adaptar mejor nuestro proyecto a las necesidades específicas de nuestros clientes.

### **Programa de Recolección de datos en vivo:**

Para recibir la información enviada por el Arduino nano, se utiliza un programa escrito en Python. Este programa utiliza la librería PySerial para establecer una conexión serial con el Arduino y recibir los datos enviados por este.

Primero, se establece la conexión serial con el Arduino utilizando el puerto correspondiente. Luego, se utiliza un bucle "while" para recibir y procesar los datos continuamente. Dentro de este bucle, se utiliza la función "serial.readline()" para leer la línea enviada por el Arduino.

Una vez que se ha recibido la línea, se utiliza la función "split()" para separar los valores de temperatura, humedad y presión. Estos valores se almacenan en variables y se convierten a números flotantes para poder realizar operaciones matemáticas con ellos.

Luego, se puede utilizar la información recibida para realizar diversas tareas, como graficar la temperatura y humedad en tiempo real, almacenar los datos en una base de datos para su posterior análisis, o incluso enviar alertas en caso de que se detecte una condición peligrosa, como una humedad demasiado alta.



Para asegurar la integridad de los datos recibidos, se pueden utilizar diversas técnicas, como incluir un código de verificación en cada transmisión para detectar errores de transmisión, o incluso implementar un sistema de redundancia para asegurar que los datos se reciban correctamente.

### Programa de análisis gráfico de datos:

Para comenzar, el programa importa las librerías necesarias para leer el archivo CSV y graficar los datos. También se importan las librerías para crear checkboxes y una función para interactuar con ellos:

```
import csv
import matplotlib.pyplot as plt
import matplotlib.widgets as widgets
import os
```

Luego, se obtiene el directorio de trabajo actual y se imprimen los archivos que se encuentran en ese directorio:

```
current_directory = os.getcwd()
print("Current directory: ", current_directory)
print("Files in current directory: ", os.listdir())
```

A continuación, el programa lee los datos desde el archivo CSV y los almacena en un diccionario llamado data. Para hacer esto, se utiliza un bucle for para recorrer cada



```
with open('data.csv', 'r') as csvfile:

    reader = csv.DictReader(csvfile)

    data = {'time': [], 'humidity': [], 'temperature': [],
'pressure': []}

    for row in reader:

        data['time'].append(row['time'])

        data['humidity'].append(float(row['humidity']))

        data['temperature'].append(float(row['temperature']))

        data['pressure'].append(float(row['pressure']))
```

Después de leer los datos, se crea una figura y unos ejes con subplots(). Luego, se grafican los datos utilizando la función plot() y se asignan las etiquetas correspondientes a cada línea:

```
fig, ax = plt.subplots()

humidity_plot, = ax.plot(data['time'], data['humidity'],
label='Humidity')

temperature_plot, = ax.plot(data['time'], data['temperature'],
label='Temperature')

pressure_plot, = ax.plot(data['time'], data['pressure'],
label='Pressure')
```

A continuación, se agregan las etiquetas de los ejes y el título de la gráfica con set\_xlabel(), set\_ylabel() y set\_title(). También se agrega una leyenda para identificar cada línea en la gráfica:

```
ax.set_xlabel('Time')
ax.set_ylabel('Value')
ax.set_title('Arduino Data')
ax.legend()
```

Para permitir al usuario ocultar o mostrar cada línea en la gráfica, se crean checkboxes con CheckButtons() y se los ubica en una posición específica en la gráfica con axes(). Luego, se define una función update\_plot() que se ejecutará cada vez que se interactúe con un checkbox. Esta función modificará la visibilidad de cada línea en la gráfica según el estado de los checkboxes:

```
checkboxes_ax = plt.axes([0.05, 0.7, 0.1, 0.2])
checkboxes = widgets.CheckButtons(checkboxes_ax, labels=['Humidity',
'Temperature', 'Pressure'])

line_visibility = {'Humidity': True, 'Temperature': True,
'Pressure': True}

def update_plot(label):
    if label == 'Humidity':
        line_visibility['Humidity'] = not
line_visibility['Humidity']
        humidity_plot.set_visible(line_visibility['Humidity'])
    elif label == 'Temperature':
```

```

        line_visibility['Temperature'] = not
line_visibility['Temperature']
        temperature_plot.set_visible(line_visibility['Temperature'])
    elif label == 'Pressure':
        line_visibility['Pressure'] = not
line_visibility['Pressure']
        pressure_plot.set_visible(line_visibility['Pressure'])
    plt.draw()

checkboxes.on_clicked(update_plot)

```

Este programa leerá los datos de un archivo CSV generado por el Arduino, los procesa y los muestra en un gráfico utilizando la biblioteca Matplotlib de Python. Además, se proporcionan opciones para seleccionar qué datos se muestran en el gráfico mediante casillas de verificación interactivas.

### ¿Es este producto viable Económicamente?

En primer lugar, es importante tener en cuenta que el costo total de los componentes necesarios para este proyecto es de \$23.50. Si se tiene en cuenta que el precio de mercado de dispositivos similares es significativamente más alto, esto puede ser un factor importante para hacer que el proyecto sea atractivo para un público más amplio. Además, el hecho de que el proyecto sea fácil de replicar con componentes comunes y económicos hace que sea accesible para una audiencia más amplia,



incluyendo a personas que no tienen mucha experiencia en electrónica.

Además, si se considera la manufacturación a gran escala, la opción de usar una placa de circuito impreso (PCB) personalizada en lugar de un protoboard y cables puede reducir significativamente los costos y aumentar la eficiencia en la producción. Aunque el costo inicial de diseño y fabricación de la PCB puede ser alto, a largo plazo puede ser una inversión rentable.

El proyecto tiene una serie de fortalezas, como el bajo costo, la facilidad de acceso a los componentes y la capacidad de personalizar la funcionalidad según las necesidades del usuario. Sin embargo, hay algunas debilidades, como la necesidad de una fuente de alimentación constante y la necesidad de programación. En cuanto a las oportunidades, el proyecto tiene el potencial de llegar a una audiencia más amplia y tener aplicaciones en áreas como la agricultura, la industria alimentaria, la medicina y más. Por último, las amenazas incluyen la competencia de otros productos similares en el mercado y la posibilidad de que los costos de los componentes aumenten en el futuro.

En resumen, aunque hay algunos desafíos y factores a considerar, el proyecto tiene un gran potencial y puede ser una opción rentable y accesible para aquellos que buscan monitorear y analizar los datos ambientales en su entorno.

#### **Plan de mejoras para el proyecto en el futuro:**

- **Agregar más sensores:**

Agregar más sensores, como el sensor de pureza del aire y el nivel de CO<sub>2</sub>, puede aumentar la funcionalidad del proyecto de fábrica. Otorgar más posibilidades hace que el



producto sea más útil e interesante para más áreas.

- **Simplificar el código:**

Simplificar el código puede ayudar a que sea más fácil de entender y de mantener.

Puede lograr esto mediante la organización del código en funciones y módulos reutilizables, el uso de comentarios claros y el uso de nombres de variables y funciones descriptivas.

También puede considerar la posibilidad de utilizar librerías y módulos de código existentes en lugar de escribir su propio código desde cero. La documentación clara y concisa del código también puede ser útil para futuros desarrolladores que trabajen en su proyecto.

- **Desarrollar un software fácil de usar:**

Para que su software sea fácil de usar para cualquier persona, es importante tener en cuenta el nivel de experiencia y conocimiento del usuario final. Si el usuario no tiene experiencia en programación o en la manipulación de hardware, es posible que deba crear una interfaz gráfica de usuario (GUI) intuitiva que permita a los usuarios interactuar con su sistema sin necesidad de escribir código o realizar conexiones de hardware complicadas. También puede considerar la posibilidad de agregar características como la autoconfiguración del sistema y más funciones.

- **Agregar un sistema de respuesta artificial:**

Nuestro proyecto se presta mucho como un posible regulador autónomo, inteligente e independiente. Queremos agregarle herramientas de respuesta artificial para que el sensor se relacione y funcione con otras herramientas y pueda predecir y avisar de cambios de los parámetros y tomar decisiones automáticamente, por ejemplo, encender un techo



automático de un jardín que cubra las plantas cuando sienta muchas lluvias y demasiada agua.

### **Conclusión:**

En conclusión, hemos desarrollado un proyecto de monitoreo ambiental utilizando tecnología de Arduino y Python. Con nuestros sensores de temperatura, humedad, presión atmosférica y calidad del aire, podemos obtener datos precisos y útiles para entender mejor nuestro entorno. Además, con el uso de la radiofrecuencia y el almacenamiento de datos en una tarjeta SD, podemos recopilar información de forma remota y sin interrupciones.

El análisis FODA que hemos realizado muestra que nuestro proyecto tiene una gran oportunidad en el mercado debido a su bajo costo, fácil accesibilidad y calidad de datos precisos. También pudimos analizar los costos de producción y considerar la posibilidad de optimizar nuestra tecnología para su futura manufactura en PCB.

En general, hemos aprendido mucho a lo largo de este proyecto. Hemos mejorado nuestras habilidades en programación y electrónica, y hemos aprendido a trabajar en equipo en un proyecto de tecnología. Nos gustaría agradecer a nuestro tutor y a todos los que nos apoyaron en este proyecto, especialmente a nuestros compañeros de equipo que trabajaron arduamente para completar este proyecto.

Esperamos que este proyecto inspire a otros a desarrollar soluciones tecnológicas para problemas ambientales y que este proyecto pueda ser utilizado para promover la conciencia ambiental y la acción positiva en nuestras comunidades.