## Estación Meteorológica Inteligente con Análisis Predictivo

**Título del Proyecto:** (Título claro y conciso. Ejemplo: "Sistema de Monitoreo de Calidad del Aire")

**Integrantes:** (Nombres completos y números de identificación, indicando roles principales. Ejemplo: "Juan Pérez (Hardware), María Gómez (Software)")

**Fecha de Presentación:**

### 1. Introducción

* **Descripción General:**

Este proyecto busca desarrollar una estación meteorológica básica capaz de medir variables ambientales como temperatura, humedad y presión atmosférica, utilizando sensores accesibles y de bajo costo. En esta versión simplificada, los datos recogidos se almacenan localmente en un archivo CSV y se presentan mediante gráficos en Python, sin implementar algoritmos de predicción. El propósito es familiarizarse con la captura de datos ambientales, su procesamiento básico y visualización, como paso previo a sistemas predictivos más complejos. El sistema está orientado a estudiantes, docentes o usuarios interesados en la observación del clima local en tiempo real, con posibilidad de escalarlo a una estación meteorológica más completa en el futuro.

* **Objetivo General:**

Diseñar una estación meteorológica básica que registre parámetros ambientales y los visualice gráficamente para analizar tendencias climáticas simples.

**Objetivos Específicos:**

* Conectar sensores de temperatura, humedad y presión a una placa Arduino o ESP.
* Registrar los datos ambientales en archivos CSV para su almacenamiento local.
* Mostrar los datos en tiempo real mediante consola o gráficos simples usando Python.
* Analizar tendencias básicas de las variables registradas a lo largo del tiempo.
* Documentar el proceso de construcción y pruebas del sistema.
* **Alcance:**

Esta versión se enfocará únicamente en la medición y visualización de temperatura, humedad y presión. No se implementarán algoritmos de machine learning ni predicciones automatizadas. La interfaz será local (no web), y el almacenamiento se limitará a archivos CSV. No se incluirá conexión a internet ni servicios en la nube. El sistema funcionará en modo autónomo con actualizaciones manuales, ideal para fines educativos o prototipado básico.

### 2. Fundamentación

* **Justificación:**

La comprensión del clima local es clave para diversas actividades, desde la agricultura hasta la planificación urbana. Sin embargo, muchas estaciones meteorológicas comerciales presentan barreras económicas o técnicas para su implementación en contextos educativos. Este proyecto tiene como objetivo ofrecer una alternativa introductoria que permita a los usuarios observar variables ambientales esenciales mediante una herramienta sencilla y accesible. Su implementación fomenta la alfabetización científica y tecnológica, al combinar sensores físicos con programación y análisis de datos. Está alineado con los objetivos del curso al integrar electrónica básica, recolección de datos y visualización computacional, siendo una base sobre la cual se pueden construir sistemas más avanzados.

* **Estado del Arte:**

### Revisión de Literatura:

### MeteoEscuela (AEMET) ofrece recursos para el estudio del clima a nivel escolar, promoviendo estaciones meteorológicas caseras.

### Blog MeteoClim introduce a los principiantes en los conceptos básicos de meteorología, destacando la importancia de medir variables como temperatura y presión.

### 3Ciencias presenta artículos sobre estaciones meteorológicas construidas con sensores básicos y microcontroladores accesibles.

### ThingSpeak y otras plataformas permiten transmitir datos meteorológicos, aunque en esta versión no se integran por simplicidad.

### Proyectos previos con sensores DHT22 y BMP280 demuestran que es posible construir estaciones funcionales con bajo presupuesto y sin conexión remota.

### Análisis Comparativo:

| Proyecto / Fuente | Variables medidas | Visualización | Predicción | Valor agregado del presente proyecto |
| --- | --- | --- | --- | --- |
| Soluciones comerciales | Múltiples | Web o app móvil | Sí | Bajo costo, accesible, ideal para enseñar |
| Estaciones académicas completas | Temp, humedad, presión | Web + análisis | Algunas | Simplicidad, enfocado en fundamentos |
| Presente proyecto | Temp, humedad, presión | Python local | No | Bajo costo, visualización local, escalable |

### Marco Teórico:

### Una estación meteorológica tiene como función principal la recolección de datos del entorno para el estudio del clima. En este caso, se emplean sensores como el DHT22 para temperatura y humedad, y el BMP280 para presión atmosférica. Estos sensores digitales se conectan a una placa Arduino o ESP que adquiere los datos y los transmite vía serial a una computadora. Un script en Python, utilizando bibliotecas como pyserial, csv y matplotlib, permite almacenar y visualizar estos datos en tiempo real. Aunque esta versión no incluye predicción climática mediante aprendizaje automático, representa un primer paso para el entendimiento de tendencias ambientales y sienta las bases para una futura integración de algoritmos predictivos.

### 3. Marcos de Trabajo

Se detallarán metodologías, herramientas y cronograma.

* **Metodología de Desarrollo:** Descripción del enfoque (100-150 palabras) (ej., prototipado rápido, desarrollo iterativo). Explicación de etapas, justificación de la elección y descripción de la gestión del proyecto.
* **Herramientas y Tecnologías:** Enumeración de recursos:
  + **Hardware:** Listado de componentes electrónicos con modelos y especificaciones. Tabla detallada: Componente, Modelo, Especificaciones, Cantidad, Costo (opcional), Imagen (opcional).
  + **Software:** Listado de lenguajes, librerías, frameworks y entornos. Lista detallada: Nombre, Versión, Función.
* **Cronograma:** Cronograma de actividades (diagrama de Gantt sugerido) desde la investigación hasta la presentación, incluyendo todas las etapas, fechas y dependencias (si aplica).

### 4. Diseño del Proyecto

Se presentarán diagramas y modelos de la arquitectura y funcionamiento.

* **Diagramas de Conexión de Circuitos:** Esquemas detallados (usar Fritzing, Tinkercad). Diagramas para cada etapa/módulo, facilidad de comprensión, indicación de función de cada componente y conexión, y lista de materiales (BOM) correspondiente.
* **Diagramas de Flujo:** Representación gráfica de la lógica del programa (símbolos estándar). Descripción clara del inicio/fin, todas las rutas de ejecución y comentarios explicativos.
* **Diagramas UML (Opcional, recomendado):** Si aplica POO, incluir diagramas relevantes (clases, secuencia, estados). Explicación del propósito y relación con el proyecto. Usar herramienta de modelado (draw.io, Lucidchart). Diagramas precisos, completos y comprensibles.
* **Diseño de la Interfaz (si aplica):** Maquetas/wireframes de la interfaz. Descripción de la experiencia de usuario (UX) y usabilidad, incluyendo esquemas, descripción de elementos, flujo de navegación, consideraciones de usabilidad y herramientas de diseño (Balsamiq, Figma).

### 5. Implementación

Se detallará la construcción del proyecto.

* **Descripción Detallada:** Explicación de la construcción (200-300 palabras):
  + **Hardware:** Descripción paso a paso del ensamblaje del circuito, conexión de componentes, resolución de problemas y fotografías.
  + **Software:** Estructura del programa, función de módulos, algoritmos, manejo de datos del sensor e implementación de funcionalidades.
* **Código Fuente:** Fragmentos relevantes con comentarios. Si es extenso, incluir como anexo o enlace a repositorio (GitHub, GitLab) público y organizado.
* **Pruebas y Resultados:** Descripción de pruebas, presentación de resultados (datos, gráficos), evaluación del cumplimiento de objetivos. Incluir métodos de prueba, datos, análisis, comparación con objetivos e identificación de errores/mejoras.
* **Dificultades Encontradas y Soluciones:** Documentación de problemas y soluciones, con descripción específica de la naturaleza del problema y los pasos seguidos.

### 6. Conclusiones

Se presentarán las reflexiones finales.

* **Resumen de Logros:** Recapitulación de los logros y el cumplimiento de objetivos (150-200 palabras). Destacar resultados y resumir el cumplimiento de objetivos.
* **Análisis Crítico:** Evaluación del proyecto (150-200 palabras). Evaluación objetiva, identificación de fortalezas/debilidades, y consideración del impacto y potencial.
* **Trabajo Futuro:** Proposición de mejoras, extensiones o aplicaciones (100-150 palabras). Sugerir ideas, proponer extensiones y explorar aplicaciones en otros contextos.

### 7. Referencias

*Fuentes recomendadas para comenzar:*

*Agencia Estatal de Meteorología (AEMET). (s.f.). Recursos para la enseñanza de la meteorología.* [*https://meteoescuela.aemet.es/cantabria/recursos*](https://meteoescuela.aemet.es/cantabria/recursos)

*Meteoclim. (2017). Meteorología para principiantes (Parte I).* [*https://blog.meteoclim.com/meteorologia-para-principiantes-i*](https://blog.meteoclim.com/meteorologia-para-principiantes-i)

*Meteoclim. (2017). Meteorología para principiantes (Parte II).* [*https://blog.meteoclim.com/meteorologia-para-principiantes-ii*](https://blog.meteoclim.com/meteorologia-para-principiantes-ii)

*García, F. J. (2013). Manual básico de meteorología. 3Ciencias.* [*https://3ciencias.com/wp-content/uploads/2013/06/METEREOLOGIA.pdf*](https://3ciencias.com/wp-content/uploads/2013/06/METEREOLOGIA.pdf)

* Listar fuentes (libros, artículos, sitios web) con formato de citación consistente (APA, IEEE).
* Garantizar la inclusión de todas las fuentes citadas.
* Sugerir gestor de referencias (Zotero, Mendeley).

### 8. Anexos (Opcional)

* Incluir información adicional (código completo, datos de pruebas, manuales).
* Anexos claramente etiquetados y organizados.