## Sistema Inteligente de Riego Automatizado (AIoT)

**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 propone el desarrollo de un sistema básico de riego automatizado orientado al uso doméstico o educativo, que monitorea la humedad del ambiente y la temperatura utilizando un sensor DHT11 o DHT22. La información recolectada se transmite a una computadora mediante Arduino y es procesada por un script en Python que muestra los datos en la consola y los almacena en un archivo CSV. En versiones futuras se podrá integrar una bomba o válvula controlada automáticamente, pero en esta versión inicial el sistema solo ofrece monitoreo en tiempo real y registro. Esta solución busca fomentar la conciencia del uso eficiente del agua, permitiendo observar cuándo se requieren condiciones de riego sin necesidad de intervención manual constante. Está dirigido a estudiantes, docentes, jardineros urbanos y cualquier persona interesada en automatización agrícola básica.

* **Objetivo General:**

Diseñar un sistema simple de monitoreo ambiental que registre la temperatura y humedad, permitiendo evaluar manualmente la necesidad de riego.

**Objetivos Específicos:**

* Conectar un sensor DHT11 o DHT22 a una placa Arduino para medir temperatura y humedad.
* Enviar los datos al computador a través del puerto serie.
* Guardar los datos en un archivo CSV para su análisis posterior.
* Mostrar los valores recolectados en consola en tiempo real.
* Generar gráficos simples de los datos usando Python.
* **Alcance:**

Este sistema únicamente medirá temperatura y humedad ambiental mediante un único sensor digital. No incluye automatización del riego (activación de bomba) en esta etapa. Se limita al monitoreo pasivo de variables ambientales, con almacenamiento local de datos y visualización por consola o gráficos simples en Python. No se incluye conectividad remota, aplicación móvil, ni bases de datos. Esta versión está pensada como un prototipo introductorio que puede ser escalado posteriormente.

### 2. Fundamentación

* **Justificación:**

El uso responsable del agua en actividades agrícolas o domésticas es fundamental, especialmente en un contexto de escasez hídrica creciente. Sin embargo, muchas veces el riego se realiza de manera intuitiva, sin tener en cuenta las condiciones reales del entorno. Este proyecto busca introducir una solución sencilla y económica que permita observar los niveles de humedad y temperatura para tomar decisiones más informadas. Su implementación es útil tanto en jardines domésticos como en contextos educativos, donde puede servir como base para aprender sobre sistemas embebidos, sensores ambientales y gestión de datos. Este sistema promueve la formación en tecnologías aplicadas al agro, en línea con los objetivos del curso al integrar electrónica básica, programación en Python y recolección de datos ambientales.

* **Estado del Arte:**

### Revisión de Literatura:

### Neiker (2023) expone estrategias para optimizar el uso de agua en riego, destacando la importancia de monitorear humedad ambiental.

### SciELO México documenta investigaciones en automatización del riego con sensores simples para pequeños cultivos.

### PlataformaExtension.cl ofrece manuales prácticos sobre riego y su relación con variables ambientales como temperatura y humedad.

### EOS.com describe cómo la humedad del suelo es un parámetro clave en la eficiencia del riego agrícola.

### ThingSpeak y Blynk son plataformas mencionadas en estudios similares, aunque no se incluyen en esta versión básica por simplicidad.

### Análisis Comparativo:

| Proyecto / Fuente | Sensores usados | Nivel de automatización | Almacenamiento | Valor agregado del presente proyecto |
| --- | --- | --- | --- | --- |
| Soluciones comerciales | Humedad + múltiples | Alta | Nube/servidor | Bajo costo, sin dependencias externas |
| Proyectos académicos IoT | Sensor DHT + relés | Media/Alta | Base de datos | Enfoque educativo y simple para principiantes |
| Presente proyecto | DHT11/DHT22 | Nula (monitoreo manual) | Archivo CSV | Accesible, didáctico y escalable |

### Marco Teórico:

El sensor DHT11 (o DHT22) permite medir temperatura y humedad relativa del aire. Estos sensores digitales envían datos mediante una única línea de comunicación al microcontrolador, en este caso un Arduino. La información es leída en el microcontrolador y transmitida por puerto serial a una computadora. Allí, un programa en Python con las bibliotecas pyserial y matplotlib registra y visualiza los datos, que también se almacenan en formato CSV. Este formato permite analizar tendencias históricas y tomar decisiones respecto al riego. Aunque en esta versión no se activa automáticamente una bomba, el sistema sienta las bases para automatizaciones futuras. Los conceptos clave incluyen sensores ambientales, comunicación serial, persistencia de datos y visualización.

### 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:*

*Sánchez-Duarte, C. J., & Sánchez-González, J. C. (2008). Automatización del riego en la agricultura. Revista Chapingo Serie Ingeniería Agrícola, 14(2), 61–70.* [*https://www.scielo.org.mx/scielo.php?pid=S0568-25172008000400009&script=sci\_arttext*](https://www.scielo.org.mx/scielo.php?pid=S0568-25172008000400009&script=sci_arttext)

*Plataforma de Extensión. (2020). Manual de riego.* [*https://www.plataformaextension.cl/archivos/2020/11/MANUAL-DE-RIEGO.pdf*](https://www.plataformaextension.cl/archivos/2020/11/MANUAL-DE-RIEGO.pdf)

*Neiker. (2023). Optimización del riego en la agricultura moderna.* [*https://neiker.eus/wp-content/uploads/2023/05/optimizacion-riego.pdf?utm\_source=chatgpt.com*](https://neiker.eus/wp-content/uploads/2023/05/optimizacion-riego.pdf?utm_source=chatgpt.com)

*EOS. (s.f.). Humedad del suelo: métodos de medición y su importancia.* [*https://eos.com/es/blog/humedad-del-suelo/*](https://eos.com/es/blog/humedad-del-suelo/)

* 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.