

## Proyecto ABP Módulo Fullstack

- 1. NOMBRE DEL PROYECTO:** Sistema de detección y extracción de gases inflamables.

**Integrantes:** Durigutti Vittorio, Guzmán Maria Lilen, Juncos Lisandro, Luján Luciano, Márquez José, Paez Tiziano, Velez Nahuel, Zalazar Joaquín.

- 2. TIPO DE PROYECTO:** Proyecto tecnológico de seguridad industrial.

El proyecto se enmarca en el desarrollo de una solución técnica orientada al monitoreo y control de condiciones ambientales dentro de un entorno industrial, utilizando tecnologías IoT, protocolos de comunicación inalámbricos y herramientas de visualización de datos.

- 3. ESPACIO CURRICULAR O ESPACIOS PARTICIPANTES EN EL MÓDULO:**

“Arquitectura y conectividad” a cargo del Ing. Morales Jorge Elías, “Sistemas de control” a cargo del profesor Gonzalez Mario Alejandro y “Plataformas” a cargo del profesor Mainero Alejandro.

- 4. EJES TEMÁTICOS/RED DE CONCEPTOS:**

### Arquitectura y conectividad

- ♦ Protocolos de comunicación IoT: Implementación de LoRa punto a punto para la transmisión inalámbrica de datos entre sensores y nodo central.
- ♦ Wi-Fi como gateway de salida local: Uso de un nodo central con capacidades de conexión Wi-Fi para visualizar o reenviar datos.
- ♦ MQTT como protocolo de mensajería: Uso de Mosquitto para simular el comportamiento de un broker en ambientes conectados.

### Sistemas de control

- ♦ Dashboards de monitoreo en tiempo real: Visualización de datos provenientes de sensores en Grafana, aplicando conocimientos sobre interpretación de señales y condiciones ambientales.

- ♦ Automatización basada en umbrales: Desarrollo de lógica de control que activa extractores en función de valores críticos sensados.

### Plataformas

- ♦ Contenerización con Docker: Despliegue de servicios y bases de datos en entornos virtualizados.

- ♦ Bases de datos relacionales: Almacenamiento de datos históricos para análisis posterior.

- ♦ Cloud Computing (referencial): Si bien no se usó AWS en producción, se discutieron conceptos relacionados que permitieron entender entornos escalables y posibles futuras migraciones a la nube.

**Competencias a fortalecer:** Capacidad para diseñar soluciones tecnológicas integrales, habilidad en la implementación de redes y protocolos de comunicación, dominio de herramientas para el control y automatización, manejo de entornos colaborativos y control de versiones (Git/GitHub), comunicación efectiva y documentación profesional.

5. **PROBLEMÁTICAS/NECESIDADES:** Tenemos distribuidos una serie de sensores de gas combustibles a lo largo del depósito, ubicados estratégicamente. El mismo dispone de energía eléctrica provista por generadores, pero no de internet. Mediante la aplicación de dispositivos sensores con comunicación LoRa incorporados buscamos estar informados en el estado a tiempo real dentro del sitio. Esto nos permite llevar un seguimiento y análisis preventivo, e identificar fallas que deban requerir intervención humana.

### 6. FUNDAMENTACIÓN:

#### ¿Por qué elegimos este problema/necesidad?

Este problema representa una situación real en espacios industriales que pueden poner en riesgo la seguridad de personas e infraestructura. Como futuros técnicos en telecomunicaciones, decidimos abordarlo aplicando nuestros conocimientos en conectividad, automatización y desarrollo fullstack.

### ¿Cuál es el potencial del proyecto?

El sistema puede escalarse y adaptarse a diferentes contextos industriales o rurales, donde el monitoreo ambiental y la actuación remota sean críticos. Tiene potencial para convertirse en un producto técnico replicable y de bajo costo.

### Relevancia respecto al perfil profesional:

El proyecto sintetiza competencias clave del técnico en telecomunicaciones: diseño de soluciones conectadas, automatización, visualización, documentación técnica y trabajo colaborativo.

### Impacto social/comunitario:

Contribuye a mejorar la seguridad en espacios de trabajo, optimiza el monitoreo de condiciones ambientales, y promueve el uso de tecnologías sustentables, autónomas y descentralizadas en zonas sin conectividad.

## 7. VISIÓN DEL PROYECTO:

### DISEÑO DE LOS OBJETIVOS

- **Objetivo general:** Desarrollar un sistema IoT para la detección y gestión de gases inflamables en un entorno industrial sin conectividad a internet, integrando sensores, lógica de control, visualización de datos y accionamiento remoto, con tecnologías abiertas
- **Objetivos específicos:**
  1. **Diseñar la arquitectura de sensado, comunicación y control**, adecuada para un entorno sin acceso a internet, utilizando tecnologías como LoRa, MQTT y bases de datos locales.
  2. **Implementar la comunicación inalámbrica entre sensores y gateway**, utilizando LoRa en ESP32 para transmitir datos de concentración de gases de forma robusta y eficiente.

3. **Desarrollar la lógica de control para extractores de aire**, automatizando la activación mediante condiciones predefinidas y control PWM desde el microcontrolador.
4. **Diseñar e implementar una aplicación de control remoto**, mediante interfaz móvil basada en MQTT para accionar manualmente los ventiladores.
5. **Registrar y visualizar los datos en tiempo real**, integrando una base de datos MariaDB con dashboards en Grafana para facilitar la toma de decisiones y análisis preventivo.
6. **Documentar y contenerizar la solución**, utilizando Docker para facilitar el despliegue, mantenimiento y escalabilidad del sistema completo.

## 8. SELECCIÓN DE ACCIONES

OBJETIVO ESPECÍFICO	ACCIONES
<b>Diseñar la arquitectura de sensado, comunicación y control</b>	<ul style="list-style-type: none"> <li>♦ Analizar los requerimientos técnicos del sistema y seleccionar las tecnologías adecuadas para un entorno industrial sin conexión a internet.</li> <li>♦ Diseñar e implementar la arquitectura general del sistema, incluyendo los módulos de sensado, control, comunicación y almacenamiento.</li> <li>♦ Elaboración de diagramas de arquitectura y flujo de datos.</li> </ul>

<b>Implementar la comunicación inalámbrica entre sensores y gateway</b>	<ul style="list-style-type: none"> <li>♦ Configuración de bibliotecas LoRa en dos placas ESP32.</li> <li>♦ Desarrollo de código no bloqueante para transmisión y recepción de datos desde sensor MQ2.</li> <li>♦ Validación de consistencia de datos y estructura de mensajes.</li> </ul>
<b>Desarrollar la lógica de control para extractores de aire</b>	<ul style="list-style-type: none"> <li>♦ Programación del control de velocidad con PWM en ESP32.</li> <li>♦ Diseño de clase instanciable y reutilizable.</li> <li>♦ Implementación de condiciones automáticas de activación por umbrales críticos.</li> </ul>
<b>Diseñar e implementar una aplicación de control remoto</b>	<ul style="list-style-type: none"> <li>♦ Configuración de interfaz en app IoT MQTT Panel.</li> <li>♦ Establecimiento de conexión con broker MQTT.</li> <li>♦ Desarrollo de botones de control manual para extractores.</li> </ul>

	<ul style="list-style-type: none"> <li>♦ Validación de funcionamiento en conjunto con el sistema.</li> </ul>
<b>Registrar y visualizar los datos en tiempo real</b>	<ul style="list-style-type: none"> <li>♦ Diseño de base de datos en MariaDB con estructura adecuada para logs de sensores.</li> <li>♦ Desarrollo de API REST para la inserción de datos desde el gateway vía HTTP POST.</li> <li>♦ Integración de Grafana con MariaDB para dashboards en tiempo real.</li> <li>♦ Configuración de dashboards con actualización automática.</li> </ul>
<b>Documentar y contenerizar la solución</b>	<ul style="list-style-type: none"> <li>♦ Redacción de README técnico con explicación de arquitectura, instalación y uso.</li> <li>♦ Dockerización de servicios API y base de datos mediante <code>Dockerfile</code> y <code>docker-compose.yml</code>.</li> <li>♦ Pruebas de despliegue local para asegurar portabilidad</li> </ul>

## 9. CRONOGRAMA:

CRONOGRAMA	SEMANA 1	SEMANA 2	SEMANA 3	SEMANA 4
<b>Diseñar la arquitectura de sensado, comunicación y control</b>	Analizar los requerimientos técnicos del sistema y seleccionar las tecnologías adecuadas para un entorno industrial sin conexión a internet.	Diseñar e implementar la arquitectura general del sistema, incluyendo los módulos de sensado, control, comunicación y almacenamiento.	Diagramar la arquitectura del sistema y los flujos de datos.	Validar la coherencia técnica del diseño con respecto a los recursos disponibles.
<b>Implementar la comunicación inalámbrica entre sensores y gateway</b>	Instalar y configurar librerías LoRa en ambas placas ESP32.	Desarrollar código de transmisión de datos desde el sensor al gateway usando LoRa.	Realizar pruebas de comunicación y validación de estructura de datos.	Integrar la comunicación LoRa al sistema completo con otros módulos.

<b>Desarrollar la lógica de control para extractores de aire</b>	Definir los criterios de activación automática basados en los umbrales del sensor MQ2.	Programar clase PWM para el control de velocidad del extractor.	Integrar la lógica de control al microcontrolador principal (gateway).	Realizar pruebas funcionales del control automático con datos reales.
<b>Registrar y visualizar los datos en tiempo real</b>	Diseñar la estructura de la base de datos MariaDB (tablas, campos, relaciones).	Desarrollar API REST en Flask para insertar datos del sensor en la base de datos.	Dockerizar la API y la base de datos con <code>docker-compose.yml</code> .	Configurar Grafana, crear dashboards dinámicos y validar datos en tiempo real.
<b>Diseñar e implementar una aplicación de control remoto</b>	Planificar los comandos necesarios para el control remoto de extractores.	Configurar la app IoT MQTT Panel con conexión al broker y topics definidos.	Programar los botones de encendido/apagado remoto y asociarlos a los actuadores.	Realizar pruebas funcionales de la app integrada al sistema completo.



<b>Documentar y contenerizar la solución</b>	Comenzar documentación técnica: arquitectura, decisiones tecnológicas, objetivos.	Redactar README con instrucciones de despliegue y uso del sistema.	Crear Dockerfile y docker-compose.yml para contenerización de los servicios.	Preparar la presentación, grabar la demo y subir todo al repositorio GitHub.
--	---	---	--	--

**10. PRODUCTO FINAL:** El producto final desarrollado por el **Equipo Ópalo** consiste en un **sistema IoT funcional** para la **detección y extracción de gases inflamables** en espacios industriales sin conexión a internet. El sistema se compone de los siguientes elementos integrados:

- ♦ Un **nodo sensor** con ESP32 y sensor de gas MQ2 que transmite datos mediante tecnología **LoRa punto a punto** hacia un nodo central (gateway).
- ♦ Un **gateway** con ESP32 que recibe los datos, activa **extractores de aire mediante control PWM** cuando se superan umbrales críticos y también permite el **control remoto manual** a través de una **app MQTT**.
- ♦ Una **API REST** desarrollada en Flask que recibe los datos del sensor y los almacena en una **base de datos MariaDB** contenida en **Docker**.
- ♦ Un **dashboard en Grafana**, conectado a MariaDB, que permite visualizar en tiempo real la concentración de gases mediante gráficos dinámicos
- ♦ Toda la solución se encuentra **documentada y versionada en GitHub**, incluyendo diagramas de arquitectura, código fuente y scripts de despliegue.

Este tipo de sistema puede ser replicado en **entornos industriales, rurales o alejados**, donde se requiera **monitoreo autónomo y local de gases peligrosos**, sin depender de conectividad a internet. Aporta a la **prevención de accidentes, mejora la seguridad laboral y fomenta el uso de tecnologías abiertas** para soluciones locales y sostenibles.

**11. BIBLIOGRAFÍA:** Normas APA, 7ma edición.

