**INFORME DE RESULTADOS**

**Unidad 3 - Tarea 6**

**Peso: 10%**

**Equipo/Grupo :** 7

**Estudiantes :**

* **LIDER: MARIA CAMILA ORTIZ**
* **ALEXANDER VARGAS MEJÍA**
* **JAIME DAVID RESTREPO RODRÍGUEZ**
* **JUAN DAVID GARCÍA VARGAS**

**Objetivo General**

**Diseñar un Gobierno de Datos y la estructura de un Proyecto de Big Data de envergadura para enfrentar la problemática en relación a la implementación de una Arquitectura de Plataforma de control y monitoreo de concentración de gases tóxicos en las fábricas de la empresa “Sustancias Locas”.**

Se requiere la convergencia de talento humano, planificación del trabajo, contratación de diversos servicios, instalación y configuración de aplicaciones y servidores locales y en la Nube, desarrollo de software, soporte técnico, entre otros.

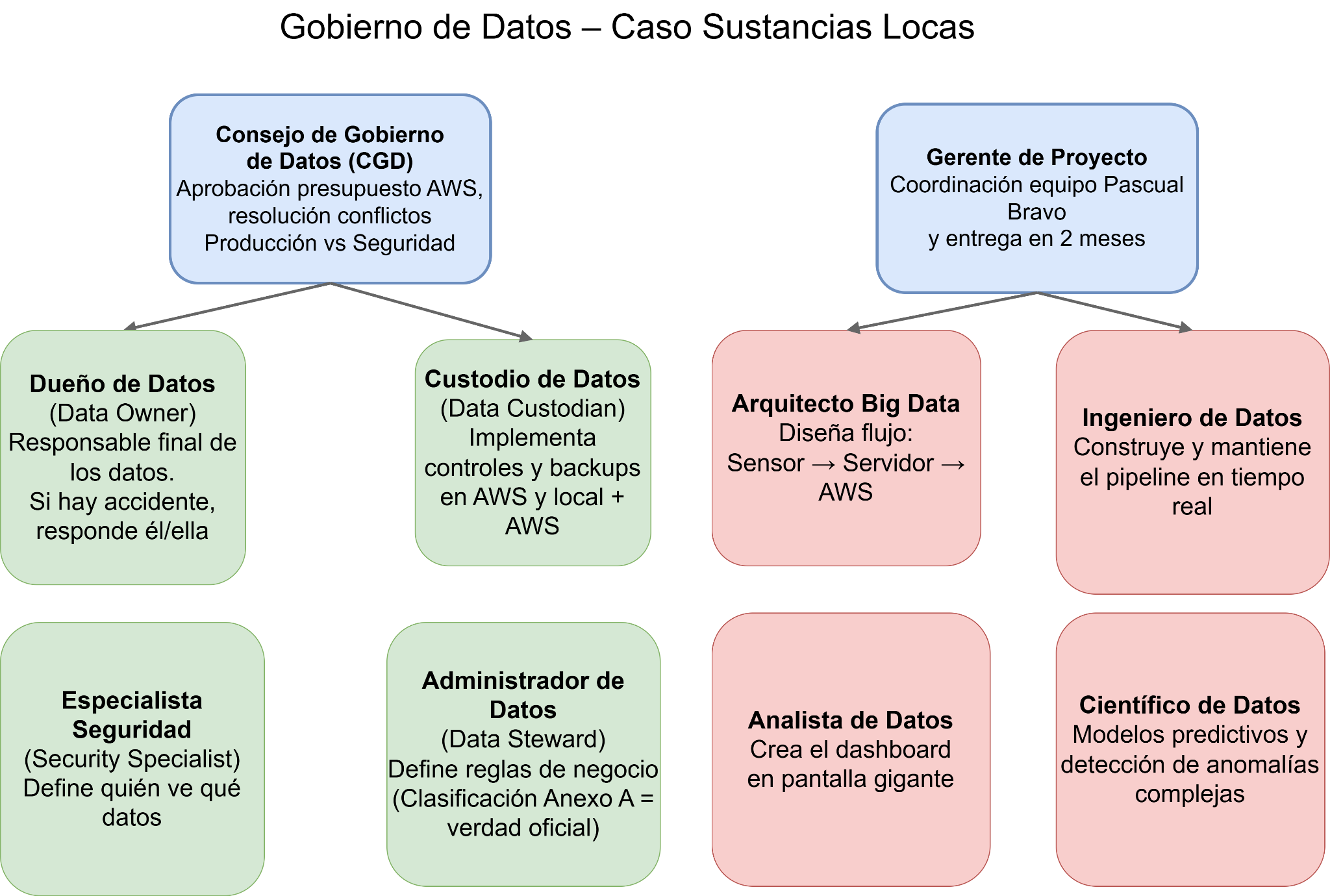
Por lo tanto, es necesario la conformación de un gobierno de datos y la estructuración de un Proyecto de Big Data; así como una lista de verificación del Proyecto Big Data que incluya todos los ítems necesarios y los tiempos estimados para lograr el objetivo.

**1.-Diseño de un Gobierno de Datos**

**1.1- Roles del gobierno de datos**

|  | **Departamento o Persona** | **Descripción** | **Rol, responsabilidades y/o funciones** |
| --- | --- | --- | --- |
| **1** | Científico de Datos (Data Scientist) | Encargado de analizar y extraer valor de los datos mediante algoritmos y técnicas estadísticas. | Analizar datos, desarrollar y validar modelos estadísticos y algorítmicos; extraer insights; colaborar en la interpretación de resultados. |
| **2** | Ingeniero de Datos (Data Engineer) | Responsable de construir, optimizar y mantener las infraestructuras de datos. | Diseñar e implementar pipelines de datos; transformar y preparar datos para consumo; mantener la infraestructura y procesos ETL/streaming. |
| **3** | Arquitecto de Big Data (Big Data Architect) | Diseña la arquitectura del sistema, asegurando la escalabilidad y eficiencia del procesamiento de datos. | Definir la arquitectura técnica y el modelo de datos; seleccionar tecnologías; asegurar escalabilidad y eficiencia del sistema. |
| **4** | Analista de Datos (Data Analyst) | Interpreta los resultados y genera insights para la toma de decisiones. | Generar reportes y visualizaciones; traducir resultados en recomendaciones; apoyar la toma de decisiones operativas y de negocio. |
| **5** | Especialista en Seguridad (Security Specialist) | Asegura la protección de los datos durante todo el proceso. | Definir e implementar políticas de seguridad; asegurar cifrado, control de accesos y cumplimiento de normativas; auditar registros. |
| **6** | Gerente de Proyecto (Project Manager) | Coordina el equipo y las fases del proyecto, asegurando que se cumplan los objetivos y tiempos. | Planificar y coordinar actividades; gestionar riesgos y comunicaciones; supervisar cumplimiento de objetivos y tiempos. |
| **7** | Dueño de Datos (Data Owner) | Ejecutivo o gerente de alto nivel responsable de la calidad, seguridad y uso de un dominio de datos específico. | Aprobar políticas de datos, definir requisitos de calidad, autorizar acceso y asegurar el cumplimiento normativo para su dominio de datos. |
| **8** | Administrador de Datos (Data Steward) | Experto en la materia responsable de gestionar la operativa diaria de un dominio de datos. | Definir y mantener metadatos, gestionar la calidad de los datos, resolver problemas de datos, implementar reglas de negocio y ser punto de contacto. |
| **9** | Custodio de Datos (Data Custodian) | Rol técnico (TI/Ingeniería) responsable de la implementación y mantenimiento de los datos. | Implementar controles de seguridad, gestionar almacenamiento, realizar copias de seguridad/recuperación y aplicar reglas de acceso definidas. |
| **10** | Consejo de Gobierno de Datos (Data Governance Council - CGD) | Comité directivo compuesto por Data Owners y otros líderes clave. | Definir la estrategia general de gobierno de datos, priorizar iniciativas, resolver conflictos entre dominios y supervisar el cumplimiento. |

***1.2.- Estructura gráfica del gobierno de datos***



**2.-Diseño del Proyecto Big Data**

**2.1- Elementos del Proyecto Big Data**

|  | **Elemento** | **Descripción** | **Observaciones** |
| --- | --- | --- | --- |
| **1** | Objetivo | Monitoreo y control en tiempo real de emisiones de benceno y otros gases tóxicos en las 4 líneas de producción. | Definir KPIs: detección, latencia, tasa de falsas alarmas. Gerencia General valida objetivos. |
| **2** | Equipo | Conformación del equipo: Científico de Datos, Ingeniero de Datos, Arquitecto de Big Data, Analista de Datos, Especialista en Seguridad y tecnólogos del Pascual Bravo. | Asignar responsables por rol y suplencias; coordinar con Supervisor de planta. |
| **3** | Sensores | Cambio de sensores antiguos por sensores de última generación instalados en microcontroladores (lecturas cada 10 s). | Proveedor entrega y tecnólogos calibran; un sensor por filtro/microcontrolador por punto crítico. |
| **4** | Filtros | Control y gestión de filtros que retienen emisiones; registro de serial y fecha de instalación. | Vida estimada hasta 15 días; programar recambio preventivo a 12 días para reducir riesgo. |
| **5** | Servidor local | Servidor on‑prem con PostgreSQL y Metabase para ingestión primaria y dashboard en planta. | Backups diarios, particionamiento por mes y visualización en pantalla gigante. |
| **6** | Pipeline nube | Envío a AWS: Kinesis Firehose → S3 (Parquet) → Glue → Redshift → Metabase para análisis histórico. | Particionar S3 por año,mes,día,fábrica; compresión rápida para reducir costos. |
| **7** | ETL y alarmas | Procesos que limpian datos, calculan agregados y aplican clasificación ppm para disparar protocolos. | Tabla `niveles\_peligrosidad` parametrizada: acciones automáticas (revisar microcontrolador, detener línea, llamar bomberos). |
| **8** | Dashboards | Paneles operativos en Metabase mostrando ppm, alarmas y estado de filtros. | Capacitación por turnos (3 turnos/día) y simulacros; acceso por roles. |

**2.2- Lista de verificación del Proyecto Big Data**

|  | **Elemento** | **Estado** | **Tiempo (días)** |
| --- | --- | --- | --- |
| **1** | Definir objetivos y KPIs | No iniciado | 2 |
| **2** | Asignar equipo y roles | No iniciado | 1 |
| **3** | Contratar/recibir sensores y filtros | No iniciado | 15 |
| **4** | Instalar y calibrar sensores | No iniciado | 9 |
| **5** | Provisionar servidor local y BD | No iniciado | 6 |
| **6** | Desarrollar API local e ingestión | No iniciado | 15 |
| **7** | Configurar pipeline en AWS | No iniciado | 6 |
| **8** | Desplegar dashboards y capacitar por turnos | No iniciado | 6 |

**3.- Conceptos y elementos de Big Data**

| **#** | **Componente** | **Descripción (breve cita bajo Norma APA)** | **Referencia (APA)** |
| --- | --- | --- | --- |
| **1** | Arquitectura de Sistema de Información | Es la forma en que se organizan los componentes tecnológicos de una empresa, como los programas, los servidores y las bases de datos, para que todo funcione correctamente y los datos fluyan de manera ordenada. | <https://cecropia.co/arquitectura-de-sistemas/> |
| **2** | Hadoop | Es una herramienta que sirve para almacenar y procesar grandes cantidades de datos usando varios computadores al mismo tiempo. Se usa mucho en proyectos de Big Data. | <https://hadoop.apache.org/> |
| **3** | Spark | Es una plataforma que permite analizar datos muy rápido porque trabaja en memoria. Se usa para hacer cálculos o reportes casi en tiempo real. | <https://spark.apache.org/> |
| **4** | PostgreSQL | Es un sistema de base de datos muy popular, gratis y de código abierto. Se usa para guardar información de forma segura y confiable. | <https://www.postgresql.org/> |
| **5** | Amazon Kinesis FireHose | Es un servicio de Amazon que ayuda a recibir y enviar datos en tiempo real a otros lugares, como S3 o Redshift, sin necesidad de hacer mucho código. | <https://aws.amazon.com/es/firehose/> |
| **6** | Amazon QuickSight | Es una herramienta que permite crear gráficos, informes y paneles para analizar los datos de forma visual e interactiva. | <https://aws.amazon.com/es/quicksight/> |
| **7** | Amazon Glue | Es un servicio que automatiza la preparación y limpieza de los datos antes de analizarlos. Se usa para procesos ETL (extraer, transformar y cargar). | <https://aws.amazon.com/es/glue/> |
| **8** | HTTP API | Es una forma en que dos programas pueden comunicarse entre sí por internet, enviando y recibiendo información a través del protocolo HTTP. | <https://developer.mozilla.org/es/docs/Glossary/API> |
| **9** | Boto3 | Es una librería de Python que permite conectar y usar fácilmente los servicios de Amazon Web Services desde un programa. | <https://boto3.amazonaws.com/v1/documentation/api/latest/index.html> |
| **10** | Amazon RedShift | Es una base de datos de Amazon diseñada para analizar grandes cantidades de información de manera rápida. Ideal para hacer reportes y estadísticas. | <https://aws.amazon.com/es/redshift/> |
| **11** | Amazon Athena | Permite consultar datos almacenados en Amazon S3 usando lenguaje SQL, sin tener que instalar o configurar servidores. | <https://aws.amazon.com/es/athena/> |
| **12** | Amazon CloudWatch | Es una herramienta de monitoreo que muestra en tiempo real cómo están funcionando los servicios y aplicaciones en AWS. | <https://aws.amazon.com/es/cloudwatch/> |
| **13** | Boto3 | N/A | N/A |
| **14** | Formato de datos JSON | Es un formato sencillo para guardar y enviar datos. Se entiende fácilmente por humanos y es muy usado en programación web. | <https://www.json.org/json-es.html> |
| **15** | Formato de datos CSV | Es un tipo de archivo donde los datos están separados por comas. Es muy común para guardar listas o tablas en hojas de cálculo. | <https://www.adobe.com/es/acrobat/resources/document-files/text-files/csv-file.html> |
| **16** | Formato de datos Parquet | Es un formato que guarda los datos por columnas, lo que hace que los archivos sean más livianos y rápidos de analizar en Big Data. | <https://parquet.apache.org/> |
| **17** | Microcontrolador Arduino | Es una pequeña tarjeta electrónica que se puede programar para leer sensores o controlar dispositivos como luces o motores. | <https://www.arduino.cc/> |
| **18** | Microcontrolador ESP8266 | Es un chip con conexión Wi-Fi que permite enviar datos por internet, muy usado en proyectos de monitoreo o IoT. | <https://www.espressif.com/en/products/socs/esp8266> |
| **19** | Sensor MQ-135 | Es un sensor que detecta gases tóxicos como amoníaco, humo o benceno, ideal para medir la calidad del aire en fábricas o laboratorios. | <https://www.sigmaelectronica.net/producto/mq-135/> |
| **20** | Benceno | Es una sustancia química inflamable y tóxica que se encuentra en el humo y en algunos procesos industriales, por eso debe controlarse en el aire. | <https://prtr-es.es/Benceno,15649,11,2007.html> |

**4.- Diseño de la estructura de la hoja de cálculo del microcontrolador “lecturas”**

**4.1. - Diseño de la estructura de la hoja de cálculo para almacenar lecturas del microcontrolador**

|  |  |  |  |
| --- | --- | --- | --- |
| **#** | **Nombre Columna** | **Descripción** | **Observaciones** |
| **1** | fecha\_hora | Fecha y hora exacta de la lectura | Formato: YYYY-MM-DD HH:MM:SS |
| **2** | concentracion\_ppm | Nivel de benceno detectado en ppm | Decimal con 3 cifras de precisión |
| **3** | temperatura | Temperatura ambiente medida por el sensor | °C |
| **4** | humedad | Humedad relativa del aire | Porcentaje (%) |
| **5** | presión atmosférica | Presión atmosférica | hPa |
| **6** | latitud | Coordenada latitud (GPS del microcontrolador) | WGS84(World Geodetic System 1984) |
| **7** | longitud | Coordenada longitud (GPS del microcontrolador) | WGS84(World Geodetic System 1984) |

**4.2. – Poblamiento de la hoja de cálculo del microcontrolador**

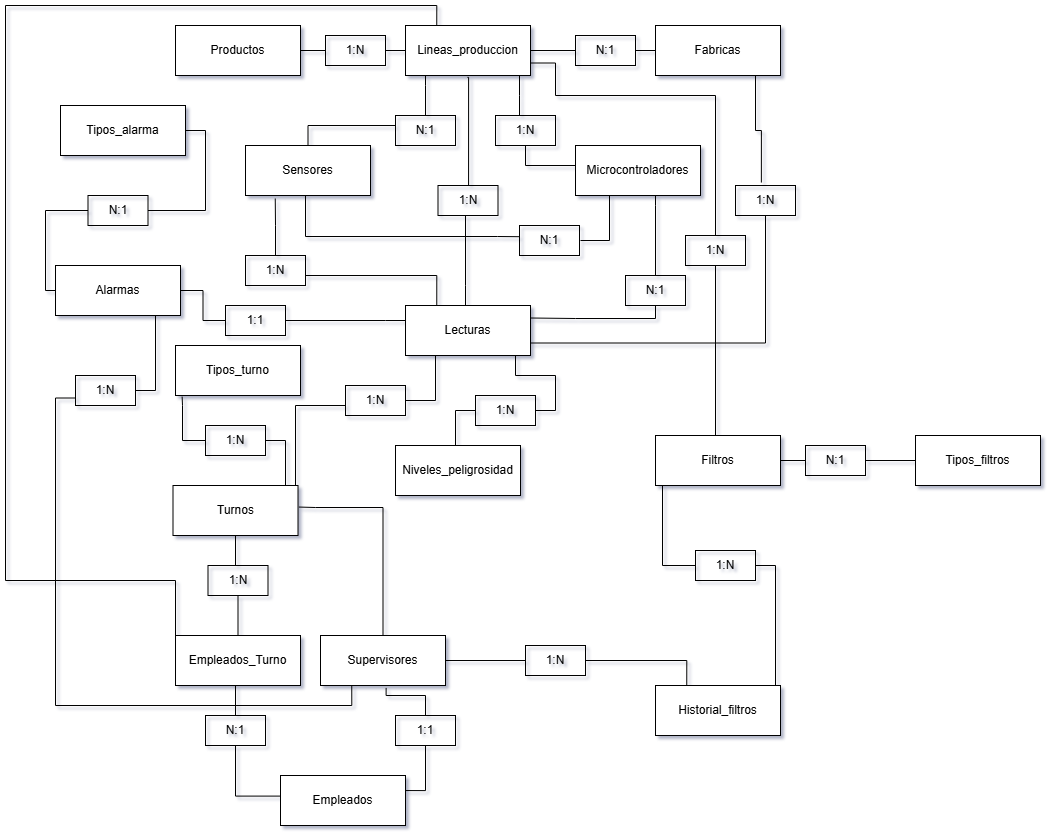
|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **#** | **fecha\_hora** | **concentracion\_ppm** | **temperatura** | **humedad** | **presion\_atmosferica** | **latitud** | **longitud** |
| **1** | 08/10/2024 08:30 | 0.25 | 24.5 | 55 | 1012 | 6.2442 | -75.5636 |
| **2** | 08/10/2024 08:31 | 0.3 | 24.6 | 55.2 | 1011.8 | 6.2442 | -75.5636 |
| **3** | 08/10/2024 08:32 | 0.28 | 24.5 | 55.1 | 1011.9 | 6.2442 | -75.5636 |
| **4** | 08/10/2024 08:33 | 0.32 | 24.7 | 55.3 | 1011.7 | 6.2442 | -75.5636 |
| **5** | 08/10/2024 08:34 | 0.4 | 24.8 | 55.5 | 1011.6 | 6.2442 | -75.5636 |
| **6** | 08/10/2024 09:00 | 5.75 | 28 | 50 | 1009.5 | 6.2 | -75.573 |
| **7** | 08/10/2024 09:01 | 5.9 | 28.1 | 50.2 | 1009.4 | 6.2 | -75.573 |
| **8** | 08/10/2024 09:02 | 6.1 | 28.2 | 50.3 | 1009.3 | 6.2 | -75.573 |
| **9** | 08/10/2024 09:03 | 6 | 28.2 | 50.4 | 1009.2 | 6.2 | -75.573 |
| **10** | 08/10/2024 09:04 | 5.85 | 28.1 | 50.6 | 1009.1 | 6.2 | -75.573 |
| **11** | 09/10/2024 15:20 | 25.12 | 30.2 | 48 | 1008.3 | 6.3 | -75.55 |
| **12** | 09/10/2024 15:21 | 26.05 | 30.3 | 48.1 | 1008.2 | 6.3 | -75.55 |
| **13** | 09/10/2024 15:22 | 24.95 | 30.4 | 48.2 | 1008.1 | 6.3 | -75.55 |
| **14** | 09/10/2024 15:23 | 25.5 | 30.4 | 48.3 | 1008 | 6.3 | -75.55 |
| **15** | 09/10/2024 15:24 | 27.1 | 30.5 | 48.5 | 1007.9 | 6.3 | -75.55 |
| **16** | 10/10/2024 02:10 | 420.9 | 36 | 60 | 1005 | 6.2442 | -75.5636 |
| **17** | 10/10/2024 02:11 | 415.8 | 36.1 | 60.3 | 1004.9 | 6.2442 | -75.5636 |
| **18** | 10/10/2024 02:12 | 430.1 | 36.3 | 60.5 | 1004.8 | 6.2442 | -75.5636 |
| **19** | 10/10/2024 02:13 | 440.55 | 36.5 | 60.8 | 1004.7 | 6.2442 | -75.5636 |
| **20** | 10/10/2024 02:14 | 450 | 36.6 | 61 | 1004.6 | 6.2442 | -75.5636 |

**5.- Diseño de la estructura de la tabla de “lecturas”**

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Tabla** | **lecturas** | |
| **#** | **Nombre Campo** | **Tipo de dato** | **Tamaño (Bytes)** |
| **1** | id\_lectura | bigserial | 12 |
| **2** | id\_sensor | integer | 4 |
| **3** | id\_linea | integer | 4 |
| **4** | id\_turno | integer | 4 |
| **5** | id\_nivel | integer | 4 |
| **6** | fecha\_hora | timestamp | 18 |
| **7** | concentracion\_ppm | numeric(10,3) | 11 |
| **8** | temperatura | numeric(5,2) | 6 |
| **9** | humedad | numeric(5,2) | 6 |
| **10** | presion\_atmosferica | numeric(7,2) | 8 |
| **11** | latitud | double precision | 10 |
| **12** | longitud | double precision | 12 |
| **13** | geom | geometry point | variable |
| **14** | payload | JSONB | variable |
| **Tamaño de un (1) registro en bytes** | | | 150 |

**6.- Diseño y creación de la Base de Datos “monitoreo-produccion”**

**6.1.- Diagrama Entidad-Relación (entidades, relaciones, atributos y cardinalidades)**

****

**6.2- Diccionario de Datos (tipos de dato, tamaño, claves primarias y foráneas)**

**TABLA: FÁBRICAS**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **CAMPO** | **TIPO** | **FORMATO** | **TAMAÑO(bytes)** | **RESTRICCIÓN** |
| id\_fabrica | SERIAL | 0000 | 4 | PK, NOT NULL |
| codigo\_fabrica | VARCHAR | TEXTO | 10 | NOT NULL, UNIQUE |
| nombre\_fabrica | VARCHAR | TEXTO | 50 | NOT NULL |
| direccion | TEXT | TEXTO | VARIABLE | NULL |
| ciudad | VARCHAR | TEXTO | 30 | NULL |
| pais | VARCHAR | TEXTO | 50 | NULL |
| geom | geometry(Point, 4326) | LONG, LAT | VARIABLE | NULL |
| contacto | TEXT | TEXTO | VARIABLE | NULL |

**TABLA: PRODUCTOS**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **CAMPO** | **TIPO** | **FORMATO** | **TAMAÑO(bytes)** | **RESTRICCIÓN** |
| id\_producto | SERIAL | 0000 | 4 | PK, NOT NULL |
| nombre\_producto | VARCHAR | TEXTO | 30 | NOT NULL, UNIQUE |
| descripcion | TEXT | TEXTO | VARIABLE | NULL |

**TABLA: MICROCONTROLADORES**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **CAMPO** | **TIPO** | **FORMATO** | **TAMAÑO(bytes)** | **RESTRICCIÓN** |
| id\_mc | SERIAL | 0000 | 4 | PK, NOT NULL |
| id\_linea | INTEGER | 0000 | 4 | FK, NOT NULL |
| codigo\_mc | VARCHAR | TEXTO | 30 | NOT NULL, UNIQUE |
| modelo | VARCHAR | TEXTO | 50 | DEFAULT ‘ESP8266’ |
| firmware\_version | VARCHAR | TEXTO | 50 | NULL |
| geom | geometry(Point, 4326) | LONG, LAT | VARIABLE | NULL |
| comentario | TEXT | TEXTO | VARIABLE | NULL |

**TABLA: LINEAS\_PRODUCCION**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **CAMPO** | **TIPO** | **FORMATO** | **TAMAÑO(bytes)** | **RESTRICCIÓN** |
| id\_linea | SERIAL | 0000 | 4 | PK, NOT NULL |
| id\_fabrica | INTEGER | 0000 | 4 | FK, NOT NULL |
| id\_producto | INTEGER | 0000 | 4 | FK, NOT NULL |
| nombre\_linea | VARCHAR | TEXTO | 50 | NOT NULL, UNIQUE |
| estado | VARCHAR | TEXTO | 20 | NOT NULL, DEFAULT ‘ACTIVA’ |
| fecha\_instalacion | DATE | YYYY-MM-DD | 10 | NOT NULL |
| ubicacion\_planta | VARCHAR | TEXTO | 100 | NOT NULL |

**TABLA: SENSORES**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **CAMPO** | **TIPO** | **FORMATO** | **TAMAÑO(bytes)** | **RESTRICCIÓN** |
| id\_sensor | SERIAL | 0000 | 4 | PK, NOT NULL |
| id\_mc | INTEGER | 0000 | 4 | FK, NOT NULL |
| id\_linea | INTEGER | 0000 | 4 | FK, NOT NULL |
| codigo\_sensor | VARCHAR | TEXTO | 50 | NOT NULL, UNIQUE |
| tipo\_sensor | VARCHAR | TEXTO | 50 | NOT NULL |
| fecha\_instalacion | DATE | YYYY-MM-DD | 10 | NOT NULL |
| fecha\_calibracion | TIMESTAMP | YYYY-MM-DD HH:MM:SS | 18 | NULL |
| estado | VARCHAR | TEXTO | 20 | NOT NULL,DEFAULT ‘OPERATIVO’ |
| precision | DECIMAL | 000,00 | 6 | NOT NULL |
| latitud | DOUBLE PRECISION | 00000000 | 8 | NULL |
| longitud | DOUBLE PRECISION | 00000000 | 8 | NULL |
| geom | geometry(Point,4326) | LONG,LAT | VARIABLE | NULL |
| observaciones | TEXT | TEXTO | VARIABLE | NULL |

**TABLA: FILTROS**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **CAMPO** | **TIPO** | **FORMATO** | **TAMAÑO(bytes)** | **RESTRICCIÓN** |
| id\_filtro | SERIAL | 0000 | 4 | PK, NOT NULL |
| id\_linea | INTEGER | 0000 | 4 | FK, NOT NULL |
| id\_tipo\_filtro | INTEGER | 0000 | 4 | FK, NOT NULL |
| codigo\_filtro | VARCHAR | TEXTO | 50 | NOT NULL, UNIQUE |
| fecha\_instalacion | DATE | YYYY-MM-DD | 10 | NOT NULL |
| fecha\_vencimiento | DATE | YYYY-MM-DD | 10 | NOT NULL |
| dias\_uso\_recomendado | INTEGER | 00 | 2 | NOT NULL, DEFAULT 15 |
| estado | VARCHAR | TEXTO | 20 | NOT NULL,DEFAULT ‘ACTIVO’ |
| proveedor | VARCHAR | TEXTO | 50 | NULL |
| observaciones | TEXT | TEXTO | VARIABLE | NULL |

**TABLA: HISTORIAL\_FILTROS**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **CAMPO** | **TIPO** | **FORMATO** | **TAMAÑO(bytes)** | **RESTRICCIÓN** |
| id\_historial | SERIAL | 0000 | 4 | PK, NOT NULL |
| id\_filtro | INTEGER | 0000 | 4 | FK, NOT NULL |
| fecha\_cambio | TIMESTAMP | YYYY-MM-DD HH:MM:SS | 18 | NOT NULL |
| dias\_uso\_real | INTEGER | 00 | 2 | NOT NULL, DEFAULT 15 |
| motivo\_cambio | VARCHAR | TEXTO | 200 | NOT NULL |
| id\_supervisor | INTEGER | 0000 | 4 | FK, NOT NULL |
| observaciones | TEXT | TEXTO | VARIABLE | NULL |

**TABLA: NIVELES\_PELIGROSIDAD**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **CAMPO** | **TIPO** | **FORMATO** | **TAMAÑO(bytes)** | **RESTRICCIÓN** |
| id\_nivel | SERIAL | 000 | 3 | PK, NOT NULL |
| nombre\_nivel | VARCHAR | TEXTO | 50 | NOT NULL, UNIQUE |
| ppm\_minimo | DECIMAL | 000000,00 | 9 | NOT NULL |
| ppm\_maximo | DECIMAL | 000000,00 | 9 | NOT NULL |
| color\_led | VARCHAR | TEXTO | 20 | NOT NULL |
| codigo\_color\_hex | VARCHAR | TEXTO | 7 | NOT NULL |
| nivel\_criticidad | INTEGER | 000 | 3 | NOT NULL |
| descripcion | TEXT | TEXTO | VARIABLE | NOT NULL |
| protocolo\_accion | TEXT | TEXTO | VARIABLE | NOT NULL |

**TABLA: LECTURAS**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **CAMPO** | **TIPO** | **FORMATO** | **TAMAÑO(bytes)** | **RESTRICCIÓN** |
| id\_lectura | BIGSERIAL | 000000000000 | 12 | PK, NOT NULL |
| id\_sensor | INTEGER | 0000 | 4 | FK, NOT NULL |
| id\_linea | INTEGER | 0000 | 4 | FK, NOT NULL |
| id\_turno | INTEGER | 0000 | 4 | FK, NOT NULL |
| id\_nivel | INTEGER | 000 | 3 | FK, NOT NULL |
| fecha\_hora | TIMESTAMP | YYYY-MM-DD HH:MM:SS | 18 | NOT NULL, DEFAULT NOW() |
| concentracion\_ppm | DECIMAL | 0000000,000 | 11 | NOT NULL |
| temperatura | DECIMAL | 000,00 (C°) | 6 | NULL |
| humedad | DECIMAL | 000,00 (%) | 6 | NULL |
| presion\_atmosferica | DECIMAL | 00000,00 (hPa) | 8 | NULL |
| latitud | DOUBLE PRECISION | 00000000 | 8 | NULL |
| longitud | DOUBLE PRECISION | 00000000 | 8 | NULL |
| geom | geometry(Point,4326) | LAT,LONG | VARIABLE | NULL |
| payload | JSONB | JSON | VARIABLE | NULL |

**TABLA: ALARMAS**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **CAMPO** | **TIPO** | **FORMATO** | **TAMAÑO(bytes)** | **RESTRICCIÓN** |
| id\_alarma | BIGSERIAL | 000000000000 | 12 | PK, NOT NULL |
| id\_lectura | BIGINT | 000000000000 | 12 | FK, NOT NULL |
| id\_tipo\_alarma | INTEGER | 0000 | 4 | FK, NOT NULL |
| id\_supervisor | INTEGER | 0000 | 4 | FK, NULL |
| fecha\_hora\_alarma | TIMESTAMP | YYYY-MM-DD HH:MM:SS | 18 | NOT NULL, DEFAULT NOW() |
| fecha\_hora\_atencion | TIMESTAMP | YYYY-MM-DD HH:MM:SS | 18 | NULL |
| estado\_alarma | VARCHAR | TEXTO | 20 | NOT NULL, DEFAULT ‘ACTIVA’ |
| acciones\_tomadas | TEXT | TEXTO | VARIABLE | NULL |
| tiempo\_respuesta | INTEGER | 00 | 2 | NULL |

**TABLA: TIPOS\_ALARMA**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **CAMPO** | **TIPO** | **FORMATO** | **TAMAÑO(bytes)** | **RESTRICCIÓN** |
| id\_tipo\_alarma | SERIAL | 000 | 3 | PK, NOT NULL |
| nombre\_tipo | VARCHAR | TEXTO | 50 | NOT NULL, UNIQUE |
| nivel\_prioridad | INTEGER | 00 | 2 | NOT NULL |
| requiere\_detencion\_linea | BOOLEAN | 1/0 | 1 | NOT NULL, DEFAULT FALSE |
| requiere\_detencion\_fabrica | BOOLEAN | 1/0 | 1 | NOT NULL, DEFAULT FALSE |
| requiere\_llamada\_bomberos | BOOLEAN | 1/0 | 1 | NOT NULL, DEFAULT FALSE |
| protocolo\_detallado | TEXT | TEXTO | VARIABLE | NOT NULL |
| tiempo\_max\_respuesta | INTEGER | 00 | 2 | NOT NULL |

**TABLA: TURNOS**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **CAMPO** | **TIPO** | **FORMATO** | **TAMAÑO(bytes)** | **RESTRICCIÓN** |
| id\_turno | SERIAL | 0000 | 4 | PK, NOT NULL |
| nombre\_turno | VARCHAR | TEXTO | 50 | NOT NULL, UNIQUE |
| hora\_inicio | TIME | HH:MM:SS | 8 | NOT NULL |
| hora\_fin | TIME | HH:MM:SS | 8 | NOT NULL |
| descripcion | VARCHAR | TEXTO | 100 | NOT NULL |

**TABLA: SUPERVISORES**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **CAMPO** | **TIPO** | **FORMATO** | **TAMAÑO(bytes)** | **RESTRICCIÓN** |
| id\_supervisor | SERIAL | 0000 | 4 | PK, NOT NULL |
| id\_turno | INTEGER | 0000 | 4 | FK, NOT NULL |
| nombre\_completo | VARCHAR | TEXTO | 20 | NOT NULL |
| cedula | VARCHAR | TEXTO | 20 | NOT NULL |
| celular | VARCHAR | TEXTO | 15 | NOT NULL |
| email | VARCHAR | TEXTO@DOMINIO | 30 | NOT NULL |
| fecha\_ingreso | DATE | YYYY-MM-DD | 10 | NOT NULL |
| estado | VARCHAR | TEXTO | 20 | NOT NULL, DEFAULT  ‘ACTIVO’ |
| certificaciones | TEXT | TEXTO |  | NULL |

**TABLA: EMPLEADOS**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **CAMPO** | **TIPO** | **FORMATO** | **TAMAÑO(bytes)** | **RESTRICCIÓN** |
| id\_empleado | SERIAL | 0000 | 4 | PK, NOT NULL |
| nombre\_completo | VARCHAR | TEXTO | 100 | NOT NULL |
| cedula | VARCHAR | TEXTO | 30 | NOT NULL |
| celular | VARCHAR | TEXTO | 20 | NOT NULL |
| email | VARCHAR | TEXTO@DOMINIO | 100 | NOT NULL |
| fecha\_ingreso | DATE | YYYY-MM-DD | 10 | NOT NULL |
| estado | VARCHAR | TEXTO | 20 | NOT NULL, DEFAULT  ‘ACTIVO’ |
| cargo | VARCHAR | TEXTO | 100 | NOT NULL |

**TABLA: EMPLEADO\_TURNO**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **CAMPO** | **TIPO** | **FORMATO** | **TAMAÑO(bytes)** | **RESTRICCIÓN** |
| id\_empleadoturno | SERIAL | 0000 | 4 | PK, NOT NULL |
| id\_empleado | INTEGER | 0000 | 4 | FK, NOT NULL |
| id\_turno | INTEGER | 0000 | 4 | FK, NOT NULL |
| fecha | DATE | YYYY-MM-DD | 10 | NOT NULL |
| id\_linea | INTEGER | 0000 | 4 | FK, NOT NULL |

**TABLA: TIPOS\_FILTRO**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **CAMPO** | **TIPO** | **FORMATO** | **TAMAÑO(bytes)** | **RESTRICCIÓN** |
| id\_tipo\_filtro | SERIAL | 0000 | 4 | PK, NOT NULL |
| nombre\_tipo | VARCHAR | TEXTO | 50 | NOT NULL, UNIQUE |
| vida\_estimada\_dias | INTEGER | 0000 | 4 | NOT NULL, DEFAULT 15 |
| costo\_usd | DECIMAL | 0000,00 | 7 | NULL |

**6.3- Scripts de creación de las tablas de la Base de Datos “monitoreo-produccion”**

|  |
| --- |
| **–TABLA: FABRICAS**  CREATE TABLE fabricas (  id\_fabrica SERIAL PRIMARY KEY,  codigo\_fabrica VARCHAR(10) NOT NULL UNIQUE,  nombre\_fabrica VARCHAR(100) NOT NULL,  direccion TEXT,  ciudad VARCHAR(100),  pais VARCHAR(100),  geom geometry(Point,4326), -- ubicación (long, lat) en EPSG:4326  contacto TEXT  );  **–TABLA: PRODUCTOS**  CREATE TABLE productos (  id\_producto SERIAL PRIMARY KEY,  nombre\_producto VARCHAR(100) NOT NULL UNIQUE,  descripcion TEXT  );  **–TABLA: LINEAS\_PRODUCCION**  CREATE TABLE lineas\_produccion (  id\_linea SERIAL PRIMARY KEY,  id\_fabrica INTEGER NOT NULL REFERENCES fabricas(id\_fabrica) ON DELETE CASCADE,  id\_producto INTEGER NOT NULL REFERENCES productos(id\_producto) ON UPDATE CASCADE,  nombre\_linea VARCHAR(50) NOT NULL,  estado VARCHAR(20) NOT NULL DEFAULT 'ACTIVA' CHECK (estado IN ('ACTIVA','INACTIVA','MANTENIMIENTO')),  fecha\_instalacion DATE NOT NULL CHECK (fecha\_instalacion <= CURRENT\_DATE),  ubicacion\_planta VARCHAR(100),  UNIQUE (id\_fabrica, nombre\_linea)  );  **–TABLA: MICROCONTROLADORES**  CREATE TABLE microcontroladores (  id\_mc SERIAL PRIMARY KEY,  id\_linea INTEGER NOT NULL REFERENCES lineas\_produccion(id\_linea) ON DELETE CASCADE,  codigo\_mc VARCHAR(30) NOT NULL UNIQUE, -- ex: A1M01  modelo VARCHAR(50) DEFAULT 'ESP8266',  firmware\_version VARCHAR(50),  geom geometry(Point,4326), -- ubicación del microcontrolador (opcional)  comentarios TEXT  );  **–TABLA: SENSORES**  CREATE TABLE sensores (  id\_sensor SERIAL PRIMARY KEY,  id\_mc INTEGER NOT NULL REFERENCES microcontroladores(id\_mc) ON DELETE CASCADE,  id\_linea INTEGER NOT NULL REFERENCES lineas\_produccion(id\_linea) ON DELETE CASCADE,  codigo\_sensor VARCHAR(50) NOT NULL UNIQUE, -- ex: A1S01  tipo\_sensor VARCHAR(100) NOT NULL, -- ex: MQ-135 calibrado para benceno  fecha\_instalacion DATE NOT NULL CHECK (fecha\_instalacion <= CURRENT\_DATE),  fecha\_calibracion TIMESTAMP,  estado VARCHAR(20) NOT NULL DEFAULT 'OPERATIVO' CHECK (estado IN ('OPERATIVO','FALLA','CALIBRANDO')),  precision DECIMAL(5,2) NOT NULL CHECK (precision > 0),  latitud DOUBLE PRECISION,  longitud DOUBLE PRECISION,  geom geometry(Point,4326),  observaciones TEXT  );  **–TABLA: TURNOS**  CREATE TABLE turnos (  id\_turno SERIAL PRIMARY KEY,  nombre\_turno VARCHAR(50) NOT NULL UNIQUE,  hora\_inicio TIME NOT NULL,  hora\_fin TIME NOT NULL,  descripcion VARCHAR(100),  CONSTRAINT chk\_turno\_horas CHECK (hora\_inicio <> hora\_fin)  );  **–TURNO: EMPLEADOS**  CREATE TABLE empleados (  id\_empleado SERIAL PRIMARY KEY,  nombre\_completo VARCHAR(100) NOT NULL,  cedula VARCHAR(30) UNIQUE,  celular VARCHAR(20),  email VARCHAR(100) UNIQUE,  fecha\_ingreso DATE,  estado VARCHAR(20) DEFAULT 'ACTIVO' CHECK (estado IN ('ACTIVO','INACTIVO','SUSPENDIDO')),  cargo VARCHAR(100)  );  **–TABLA: EMPLEADO\_TURNO**  CREATE TABLE empleado\_turno (  id\_empleadoturno SERIAL PRIMARY KEY,  id\_empleado INTEGER NOT NULL REFERENCES empleados(id\_empleado) ON DELETE CASCADE,  id\_turno INTEGER NOT NULL REFERENCES turnos(id\_turno) ON UPDATE CASCADE,  fecha DATE NOT NULL,  id\_linea INTEGER REFERENCES lineas\_produccion(id\_linea),  UNIQUE (id\_empleado, id\_turno, fecha)  );  **–TABLA: SUPERVISORES**  CREATE TABLE supervisores (  id\_supervisor SERIAL PRIMARY KEY,  id\_empleado INTEGER NOT NULL UNIQUE REFERENCES empleados(id\_empleado) ON DELETE CASCADE,  id\_turno INTEGER NOT NULL REFERENCES turnos(id\_turno) ON UPDATE CASCADE,  certificaciones TEXT  );  **–TABLA: TIPOS\_FILTRO**  CREATE TABLE tipos\_filtro (  id\_tipo\_filtro SERIAL PRIMARY KEY,  nombre\_tipo VARCHAR(100) NOT NULL UNIQUE,  vida\_estimada\_dias INTEGER NOT NULL DEFAULT 15,  costo\_usd NUMERIC(10,2)  );  **–TABLA: FILTROS**  CREATE TABLE filtros (  id\_filtro SERIAL PRIMARY KEY,  id\_linea INTEGER NOT NULL REFERENCES lineas\_produccion(id\_linea) ON DELETE CASCADE,  id\_tipo\_filtro INTEGER NOT NULL REFERENCES tipos\_filtro(id\_tipo\_filtro),  codigo\_filtro VARCHAR(50) NOT NULL UNIQUE,  fecha\_instalacion DATE NOT NULL CHECK (fecha\_instalacion <= CURRENT\_DATE),  fecha\_vencimiento DATE NOT NULL,  dias\_uso\_recomendado INTEGER NOT NULL DEFAULT 15,  estado VARCHAR(20) NOT NULL DEFAULT 'ACTIVO' CHECK (estado IN ('ACTIVO','INACTIVO','VENCIDO')),  proveedor VARCHAR(100),  observaciones TEXT  );  **–TABLA: HISTORIAL\_FILTROS**  CREATE TABLE historial\_filtros (  id\_historial SERIAL PRIMARY KEY,  id\_filtro INTEGER NOT NULL REFERENCES filtros(id\_filtro) ON DELETE CASCADE,  fecha\_cambio TIMESTAMP NOT NULL CHECK (fecha\_cambio <= NOW()),  dias\_uso\_real INTEGER NOT NULL DEFAULT 15,  motivo\_cambio VARCHAR(200),  id\_supervisor INTEGER NOT NULL REFERENCES supervisores(id\_supervisor) ON UPDATE CASCADE,  observaciones TEXT  );  **–TABLA: NIVELES\_PELIGROSIDAD**  CREATE TABLE niveles\_peligrosidad (  id\_nivel SERIAL PRIMARY KEY,  nombre\_nivel VARCHAR(50) NOT NULL UNIQUE,  ppm\_minimo DECIMAL(8,2) NOT NULL CHECK (ppm\_minimo >= 0),  ppm\_maximo DECIMAL(8,2) NOT NULL CHECK (ppm\_maximo > ppm\_minimo),  color\_led VARCHAR(20) NOT NULL,  codigo\_color\_hex VARCHAR(7) NOT NULL CHECK (codigo\_color\_hex ~ '^#[0-9A-Fa-f]{6}$'),  nivel\_criticidad INTEGER NOT NULL CHECK (nivel\_criticidad BETWEEN 1 AND 5),  descripcion TEXT,  protocolo\_accion TEXT  );  **–TABLA: TIPOS\_ALARMA**  CREATE TABLE tipos\_alarma (  id\_tipo\_alarma SERIAL PRIMARY KEY,  nombre\_tipo VARCHAR(100) NOT NULL UNIQUE,  nivel\_prioridad INTEGER NOT NULL CHECK (nivel\_prioridad BETWEEN 1 AND 5),  requiere\_detencion\_linea BOOLEAN NOT NULL DEFAULT FALSE,  requiere\_detencion\_fabrica BOOLEAN NOT NULL DEFAULT FALSE,  requiere\_llamada\_bomberos BOOLEAN NOT NULL DEFAULT FALSE,  protocolo\_detallado TEXT,  tiempo\_max\_respuesta INTEGER NOT NULL  );  **–TABLA: LECTURAS**  CREATE TABLE lecturas (  id\_lectura BIGSERIAL PRIMARY KEY,  id\_sensor INTEGER NOT NULL REFERENCES sensores(id\_sensor) ON DELETE CASCADE,  id\_mc INTEGER NOT NULL REFERENCES microcontroladores(id\_mc) ON DELETE CASCADE,  id\_linea INTEGER NOT NULL REFERENCES lineas\_produccion(id\_linea) ON DELETE CASCADE,  id\_fabrica INTEGER NOT NULL REFERENCES fabricas(id\_fabrica) ON DELETE CASCADE,  id\_turno INTEGER NOT NULL REFERENCES turnos(id\_turno) ON UPDATE CASCADE,  id\_nivel INTEGER REFERENCES niveles\_peligrosidad(id\_nivel),  fecha\_hora TIMESTAMP NOT NULL DEFAULT NOW(),  concentracion\_ppm DECIMAL(10,3) NOT NULL CHECK (concentracion\_ppm >= 0),  temperatura DECIMAL(5,2),  humedad DECIMAL(5,2),  presion\_atmosferica DECIMAL(7,2),  latitud DOUBLE PRECISION,  longitud DOUBLE PRECISION,  geom geometry(Point,4326),  payload jsonb -- para guardar metadatos originales (raw JSON del microcontrolador)  );  **–TABLA: ALARMAS**  CREATE TABLE alarmas (  id\_alarma BIGSERIAL PRIMARY KEY,  id\_lectura BIGINT NOT NULL REFERENCES lecturas(id\_lectura) ON DELETE CASCADE,  id\_tipo\_alarma INTEGER NOT NULL REFERENCES tipos\_alarma(id\_tipo\_alarma),  id\_supervisor INTEGER REFERENCES supervisores(id\_supervisor) ON UPDATE CASCADE,  fecha\_hora\_alarma TIMESTAMP NOT NULL DEFAULT NOW(),  fecha\_hora\_atencion TIMESTAMP,  estado\_alarma VARCHAR(20) NOT NULL DEFAULT 'ACTIVA' CHECK (estado\_alarma IN ('ACTIVA','ATENDIDA','RESUELTA')),  acciones\_tomadas TEXT,  tiempo\_respuesta INTEGER  );  **-- ÍNDICES OPTIMIZACIÓN**  CREATE INDEX idx\_lecturas\_fecha ON lecturas (fecha\_hora DESC);  CREATE INDEX idx\_lecturas\_sensor ON lecturas (id\_sensor);  CREATE INDEX idx\_lecturas\_fabrica\_fecha ON lecturas (id\_fabrica, fecha\_hora DESC);  CREATE INDEX idx\_alarmas\_estado ON alarmas (estado\_alarma);  CREATE INDEX idx\_sensores\_codigo ON sensores (codigo\_sensor); |

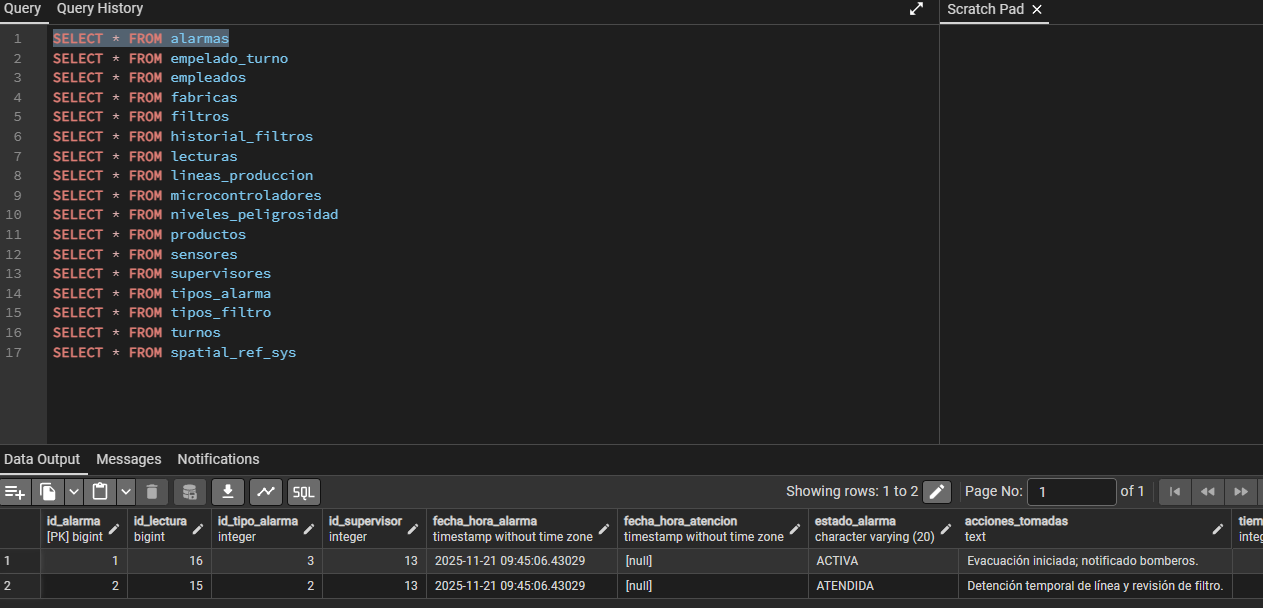
**7.- Poblamiento de la base de datos “monitoreo-produccion”**

**7.1.- Scripts de inserción de registros en todas las tablas de la base datos (INSERTS)**

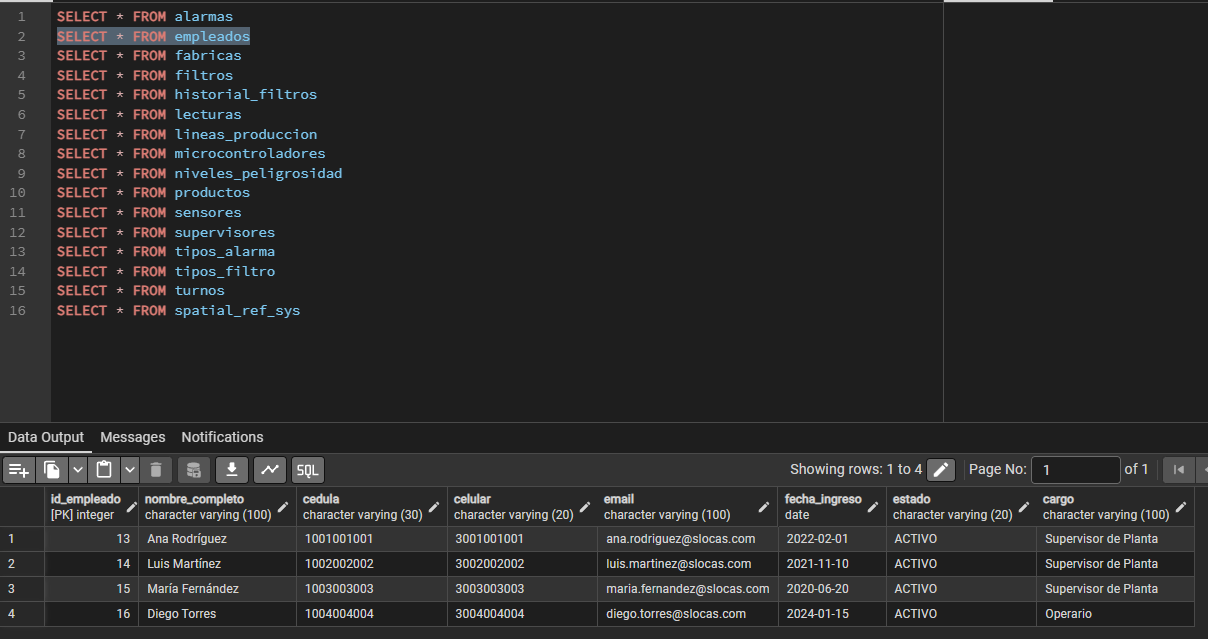
|  |
| --- |
| **-- =========================================**  **-- SCRIPT DE INSERCIONES**  **-- =========================================**  **-- 1 Productos**  **INSERT INTO productos (nombre\_producto, descripcion)**  **VALUES**  **('Benceno', 'Producto que emite benceno durante su proceso de producción (alto riesgo)'),**  **('Producto X', 'Producto secundario de ejemplo');**  **-- 2 Fábricas (A, B, C) con ubicación ejemplo (lon, lat)**  **INSERT INTO fabricas (codigo\_fabrica, nombre\_fabrica, direccion, ciudad, pais, geom, contacto)**  **VALUES**  **('A', 'Fábrica A', 'Carrera 1 #100', 'Medellín', 'Colombia', ST\_SetSRID(ST\_MakePoint(-75.5636, 6.2442),4326), 'gerenciaA@slocas.com'),**  **('B', 'Fábrica B', 'Calle 50 #20', 'Envigado', 'Colombia', ST\_SetSRID(ST\_MakePoint(-75.5730, 6.2000),4326), 'gerenciaB@slocas.com'),**  **('C', 'Fábrica C', 'Via 45 km 3', 'Bello', 'Colombia', ST\_SetSRID(ST\_MakePoint(-75.5500, 6.3000),4326), 'gerenciaC@slocas.com');**  **-- 3 Turnos**  **INSERT INTO turnos (nombre\_turno, hora\_inicio, hora\_fin, descripcion)**  **VALUES**  **('Turno 08-16', '08:00:00', '16:00:00', 'Turno diurno 8am-4pm'),**  **('Turno 16-00', '16:00:00', '00:00:00', 'Turno vespertino 4pm-12am'),**  **('Turno 00-08', '00:00:00', '08:00:00', 'Turno nocturno 12am-8am');**  **-- 4 Empleados y Supervisores**  **INSERT INTO empleados (nombre\_completo, cedula, celular, email, fecha\_ingreso, cargo)**  **VALUES**  **('Ana Rodríguez', '1001001001', '3001001001', 'ana.rodriguez@slocas.com', '2022-02-01', 'Supervisor de Planta'),**  **('Luis Martínez', '1002002002', '3002002002', 'luis.martinez@slocas.com', '2021-11-10', 'Supervisor de Planta'),**  **('María Fernández', '1003003003', '3003003003', 'maria.fernandez@slocas.com', '2020-06-20', 'Supervisor de Planta'),**  **('Diego Torres', '1004004004', '3004004004', 'diego.torres@slocas.com', '2024-01-15', 'Operario');**  **-- Mapear algunos empleados a supervisores (usamos IDs de empleados insertados)**  **INSERT INTO supervisores (id\_empleado, id\_turno, certificaciones)**  **VALUES**  **((SELECT id\_empleado FROM empleados WHERE nombre\_completo='Ana Rodríguez'), 10, 'Seguridad Industrial I'),**  **((SELECT id\_empleado FROM empleados WHERE nombre\_completo='Luis Martínez'), 11, 'Gestión Ambiental'),**  **((SELECT id\_empleado FROM empleados WHERE nombre\_completo='María Fernández'),12, 'Operación de Planta');**  **-- 5 Crear las 4 líneas por cada fábrica y asociarlas al producto "Benceno"**  **WITH p AS (SELECT id\_producto FROM productos WHERE nombre\_producto='Benceno' LIMIT 1)**  **INSERT INTO lineas\_produccion (id\_fabrica, id\_producto, nombre\_linea, fecha\_instalacion, ubicacion\_planta)**  **SELECT f.id\_fabrica, p.id\_producto,**  **CASE WHEN f.codigo\_fabrica = 'A' THEN concat('A', i, ' - Benceno')**  **WHEN f.codigo\_fabrica = 'B' THEN concat('B', i, ' - Benceno')**  **WHEN f.codigo\_fabrica = 'C' THEN concat('C', i, ' - Benceno') END,**  **-- fechas de instalacion ejemplo:**  **CASE WHEN i = 1 THEN DATE '2019/05/10' WHEN i = 2 THEN '2019/06/15' WHEN i = 3 THEN '2020/01/20' ELSE '2020/04/05' END,**  **CASE WHEN f.codigo\_fabrica = 'A' THEN 'Sector A' WHEN f.codigo\_fabrica = 'B' THEN 'Sector B' ELSE 'Sector C' END**  **FROM (SELECT id\_fabrica, codigo\_fabrica FROM fabricas) f**  **CROSS JOIN (SELECT 1 AS i UNION ALL SELECT 2 UNION ALL SELECT 3 UNION ALL SELECT 4) seq**  **CROSS JOIN p;**  **--6 Tipos de filtro**  **INSERT INTO tipos\_filtro (nombre\_tipo, vida\_estimada\_dias, costo\_usd)**  **VALUES**  **('Filtro Carbón Activo - Estándar', 15, 300.00),**  **('Filtro HEPA - Industrial', 20, 450.00);**  **-- 7 Instalar 1 filtro por línea**  **INSERT INTO filtros (id\_linea, id\_tipo\_filtro, codigo\_filtro, fecha\_instalacion, fecha\_vencimiento, dias\_uso\_recomendado, proveedor)**  **SELECT l.id\_linea, (SELECT id\_tipo\_filtro FROM tipos\_filtro WHERE nombre\_tipo LIKE 'Filtro Carbón Activo%'),**  **concat('FILT-', l.id\_linea, '-', to\_char(CURRENT\_DATE,'YYYYMMDD')),**  **CURRENT\_DATE - INTERVAL '10 day', (CURRENT\_DATE - INTERVAL '10 day') + INTERVAL '7 day', 15, 'ProveedorFiltros S.A.'**  **FROM lineas\_produccion l;**  **-- 8 Niveles de peligrosidad**  **INSERT INTO niveles\_peligrosidad (nombre\_nivel, ppm\_minimo, ppm\_maximo, color\_led, codigo\_color\_hex, nivel\_criticidad, descripcion, protocolo\_accion)**  **VALUES**  **('Bajo', 0.00, 1.00, 'Verde', '#00FF00', 1, 'Dentro de tolerancia OSHA/NIOSH', 'Continuar operación. Monitorizar.'),**  **('Moderado', 1.01, 10.00, 'Amarillo', '#FFFF00', 2, 'Precaución', 'Aumentar ventilación; revisar filtros y sensores.'),**  **('Alto', 10.01, 100.00, 'Naranja', '#FFA500', 3, 'Riesgo alto', 'Notificar supervisor; preparar detención de línea.'),**  **('Crítico', 100.01, 2000.00, 'Rojo', '#FF0000', 5, 'Riesgo crítico', 'Evacuar área; detener línea/fábrica; notificar bomberos.');**  **-- 9 Tipos de alarma**  **INSERT INTO tipos\_alarma (nombre\_tipo, nivel\_prioridad, requiere\_detencion\_linea, requiere\_detencion\_fabrica, requiere\_llamada\_bomberos, protocolo\_detallado, tiempo\_max\_respuesta)**  **VALUES**  **('Alarma Precaución', 2, FALSE, FALSE, FALSE, 'Aumentar ventilación; revisar filtros y sensores; notificar supervisor.', 30),**  **('Alarma Alto', 4, TRUE, FALSE, FALSE, 'Notificar supervisor; detener línea si no baja en 5 min; inspección técnica.', 15),**  **('Alarma Crítica', 5, TRUE, TRUE, TRUE, 'Evacuar sector; detener fábrica; llamar bomberos y defensa civil; activar plan de emergencia.', 5);**  **-- 10 CREAR microcontroladores y sensores MASIVAMENTE (240 MC y 240 sensores)**  **-- Estructura: fábrica A,B,C; líneas i=1..4 por fabrica; microcontroladores por línea m=1..20**  **-- Codigo MC ex: A1M01, sensor: A1S01**  **-- Insert microcontroladores**  **DO $$**  **DECLARE**  **fab RECORD;**  **line RECORD;**  **m INT;**  **cod\_mc TEXT;**  **BEGIN**  **FOR fab IN SELECT codigo\_fabrica, id\_fabrica FROM fabricas LOOP**  **FOR line IN SELECT id\_linea, nombre\_linea FROM lineas\_produccion WHERE id\_fabrica = fab.id\_fabrica LOOP**  **FOR m IN 1..20 LOOP**  **cod\_mc := fab.codigo\_fabrica || substring(line.nombre\_linea from 1 for 2) || 'M' || lpad(m::text,2,'0');**  **INSERT INTO microcontroladores (id\_linea, codigo\_mc, firmware\_version)**  **VALUES (line.id\_linea, cod\_mc, 'v1.0')**  **ON CONFLICT (codigo\_mc) DO NOTHING;**  **END LOOP;**  **END LOOP;**  **END LOOP;**  **END$$;**  **-- Insert sensores vinculados a microcontroladores**  **DO $$**  **DECLARE**  **mc RECORD;**  **cod\_sens TEXT;**  **BEGIN**  **FOR mc IN SELECT id\_mc, codigo\_mc, id\_linea FROM microcontroladores LOOP**  **cod\_sens := replace(mc.codigo\_mc,'M','S'); -- ejemplo: A1M01 -> A1S01**  **INSERT INTO sensores (id\_mc, id\_linea, codigo\_sensor, tipo\_sensor, fecha\_instalacion, precision, estado)**  **VALUES (mc.id\_mc, mc.id\_linea, cod\_sens, 'MQ-135 (calibrado para Benceno)', CURRENT\_DATE - INTERVAL '400 day', 0.05, 'OPERATIVO')**  **ON CONFLICT (codigo\_sensor) DO NOTHING;**  **END LOOP;**  **END$$;**  **-- 11 POBLAR lecturas de ejemplo**  **-- Insertaremos lecturas de 5 sensores distribuidos con distintos niveles.**  **INSERT INTO lecturas (**  **id\_sensor, id\_mc, id\_linea, id\_fabrica, id\_turno, id\_nivel,**  **fecha\_hora, concentracion\_ppm, temperatura, humedad, presion\_atmosferica,**  **latitud, longitud, geom, payload**  **)**  **VALUES**  **-- SENSOR BAJO – AA1M01 → AA1S01 → TURNO 10**  **(**  **(SELECT id\_sensor FROM sensores WHERE codigo\_sensor = 'AA1S01'),**  **(SELECT id\_mc FROM microcontroladores WHERE codigo\_mc = 'AA1M01'),**  **(SELECT id\_linea FROM microcontroladores WHERE codigo\_mc = 'AA1M01'),**  **(SELECT id\_fabrica FROM lineas\_produccion WHERE id\_linea = (SELECT id\_linea FROM microcontroladores WHERE codigo\_mc='AA1M01')),**  **10, 1,**  **'2024-10-08 08:30:00', 0.25, 24.5, 55.0, 1012.0,**  **6.2442, -75.5636, ST\_SetSRID(ST\_MakePoint(-75.5636,6.2442),4326),**  **'{"raw":"example","sensor":"AA1S01"}'::jsonb**  **),**  **-- SENSOR MODERADO – BB2M05 → BB2S05 → TURNO 11**  **(**  **(SELECT id\_sensor FROM sensores WHERE codigo\_sensor = 'BB2S05'),**  **(SELECT id\_mc FROM microcontroladores WHERE codigo\_mc = 'BB2M05'),**  **(SELECT id\_linea FROM microcontroladores WHERE codigo\_mc = 'BB2M05'),**  **(SELECT id\_fabrica FROM lineas\_produccion WHERE id\_linea = (SELECT id\_linea FROM microcontroladores WHERE codigo\_mc='BB2M05')),**  **11, 2,**  **'2024-10-08 17:45:00', 5.75, 28.0, 50.0, 1009.5,**  **6.2000, -75.5730, ST\_SetSRID(ST\_MakePoint(-75.5730,6.2000),4326),**  **'{"raw":"example","sensor":"BB2S05"}'::jsonb**  **),**  **-- SENSOR ALTO – CC3M10 → CC3S10 → TURNO 10**  **(**  **(SELECT id\_sensor FROM sensores WHERE codigo\_sensor = 'CC3S10'),**  **(SELECT id\_mc FROM microcontroladores WHERE codigo\_mc = 'CC3M10'),**  **(SELECT id\_linea FROM microcontroladores WHERE codigo\_mc = 'CC3M10'),**  **(SELECT id\_fabrica FROM lineas\_produccion WHERE id\_linea = (SELECT id\_linea FROM microcontroladores WHERE codigo\_mc='CC3M10')),**  **10, 3,**  **'2024-10-09 15:20:00', 25.125, 30.2, 48.0, 1008.3,**  **6.3000, -75.5500, ST\_SetSRID(ST\_MakePoint(-75.5500,6.3000),4326),**  **'{"raw":"example","sensor":"CC3S10"}'::jsonb**  **),**  **-- SENSOR CRÍTICO – AA4M20 → AA4S20 → TURNO 12**  **(**  **(SELECT id\_sensor FROM sensores WHERE codigo\_sensor = 'AA4S20'),**  **(SELECT id\_mc FROM microcontroladores WHERE codigo\_mc = 'AA4M20'),**  **(SELECT id\_linea FROM microcontroladores WHERE codigo\_mc = 'AA4M20'),**  **(SELECT id\_fabrica FROM lineas\_produccion WHERE id\_linea = (SELECT id\_linea FROM microcontroladores WHERE codigo\_mc='AA4M20')),**  **12, 4,**  **'2024-10-10 02:10:00', 420.900, 36.0, 60.0, 1005.0,**  **6.2442, -75.5636, ST\_SetSRID(ST\_MakePoint(-75.5636,6.2442),4326),**  **'{"raw":"example","sensor":"AA4S20"}'::jsonb**  **);**  **-- ADD**  **INSERT INTO lecturas (**  **id\_sensor, id\_mc, id\_linea, id\_fabrica, id\_turno, id\_nivel,**  **fecha\_hora, concentracion\_ppm, temperatura, humedad, presion\_atmosferica,**  **latitud, longitud, geom, payload**  **)**  **VALUES**  **-- =====================================================================**  **-- LECTURAS 1–5 → SENSOR BAJO (AA1S01 / AA1M01) → NIVEL 1 → TURNO 10**  **-- =====================================================================**  **(**  **(SELECT id\_sensor FROM sensores WHERE codigo\_sensor = 'AA1S01'),**  **(SELECT id\_mc FROM microcontroladores WHERE codigo\_mc = 'AA1M01'),**  **(SELECT id\_linea FROM microcontroladores WHERE codigo\_mc = 'AA1M01'),**  **(SELECT id\_fabrica FROM lineas\_produccion WHERE id\_linea = (SELECT id\_linea FROM microcontroladores WHERE codigo\_mc='AA1M01')),**  **10, 1,**  **'2024-10-08 08:30', 0.25, 24.5, 55, 1012,**  **6.2442, -75.5636, ST\_SetSRID(ST\_MakePoint(-75.5636,6.2442),4326),**  **'{"sensor":"AA1S01"}'**  **),**  **(**  **(SELECT id\_sensor FROM sensores WHERE codigo\_sensor = 'AA1S01'),**  **(SELECT id\_mc FROM microcontroladores WHERE codigo\_mc = 'AA1M01'),**  **(SELECT id\_linea FROM microcontroladores WHERE codigo\_mc = 'AA1M01'),**  **(SELECT id\_fabrica FROM lineas\_produccion WHERE id\_linea = (SELECT id\_linea FROM microcontroladores WHERE codigo\_mc='AA1M01')),**  **10, 1,**  **'2024-10-08 08:31', 0.30, 24.6, 55.2, 1011.8,**  **6.2442, -75.5636, ST\_SetSRID(ST\_MakePoint(-75.5636,6.2442),4326),**  **'{"sensor":"AA1S01"}'**  **),**  **(**  **(SELECT id\_sensor FROM sensores WHERE codigo\_sensor = 'AA1S01'),**  **(SELECT id\_mc FROM microcontroladores WHERE codigo\_mc = 'AA1M01'),**  **(SELECT id\_linea FROM microcontroladores WHERE codigo\_mc = 'AA1M01'),**  **(SELECT id\_fabrica FROM lineas\_produccion WHERE id\_linea = (SELECT id\_linea FROM microcontroladores WHERE codigo\_mc='AA1M01')),**  **10, 1,**  **'2024-10-08 08:32', 0.28, 24.5, 55.1, 1011.9,**  **6.2442, -75.5636, ST\_SetSRID(ST\_MakePoint(-75.5636,6.2442),4326),**  **'{"sensor":"AA1S01"}'**  **),**  **(**  **(SELECT id\_sensor FROM sensores WHERE codigo\_sensor = 'AA1S01'),**  **(SELECT id\_mc FROM microcontroladores WHERE codigo\_mc = 'AA1M01'),**  **(SELECT id\_linea FROM microcontroladores WHERE codigo\_mc = 'AA1M01'),**  **(SELECT id\_fabrica FROM lineas\_produccion WHERE id\_linea = (SELECT id\_linea FROM microcontroladores WHERE codigo\_mc='AA1M01')),**  **10, 1,**  **'2024-10-08 08:33', 0.32, 24.7, 55.3, 1011.7,**  **6.2442, -75.5636, ST\_SetSRID(ST\_MakePoint(-75.5636,6.2442),4326),**  **'{"sensor":"AA1S01"}'**  **),**  **(**  **(SELECT id\_sensor FROM sensores WHERE codigo\_sensor = 'AA1S01'),**  **(SELECT id\_mc FROM microcontroladores WHERE codigo\_mc = 'AA1M01'),**  **(SELECT id\_linea FROM microcontroladores WHERE codigo\_mc = 'AA1M01'),**  **(SELECT id\_fabrica FROM lineas\_produccion WHERE id\_linea = (SELECT id\_linea FROM microcontroladores WHERE codigo\_mc='AA1M01')),**  **10, 1,**  **'2024-10-08 08:34', 0.40, 24.8, 55.5, 1011.6,**  **6.2442, -75.5636, ST\_SetSRID(ST\_MakePoint(-75.5636,6.2442),4326),**  **'{"sensor":"AA1S01"}'**  **),**  **-- =====================================================================**  **-- LECTURAS 6–10 → SENSOR MODERADO (BB2S05 / BB2M05) → NIVEL 2 → TURNO 11**  **-- =====================================================================**  **(**  **(SELECT id\_sensor FROM sensores WHERE codigo\_sensor = 'BB2S05'),**  **(SELECT id\_mc FROM microcontroladores WHERE codigo\_mc = 'BB2M05'),**  **(SELECT id\_linea FROM microcontroladores WHERE codigo\_mc = 'BB2M05'),**  **(SELECT id\_fabrica FROM lineas\_produccion WHERE id\_linea = (SELECT id\_linea FROM microcontroladores WHERE codigo\_mc='BB2M05')),**  **11, 2,**  **'2024-10-08 09:00', 5.75, 28, 50, 1009.5,**  **6.20, -75.5730, ST\_SetSRID(ST\_MakePoint(-75.5730,6.2000),4326),**  **'{"sensor":"BB2S05"}'**  **),**  **(**  **(SELECT id\_sensor FROM sensores WHERE codigo\_sensor = 'BB2S05'),**  **(SELECT id\_mc FROM microcontroladores WHERE codigo\_mc = 'BB2M05'),**  **(SELECT id\_linea FROM microcontroladores WHERE codigo\_mc = 'BB2M05'),**  **(SELECT id\_fabrica FROM lineas\_produccion WHERE id\_linea = (SELECT id\_linea FROM microcontroladores WHERE codigo\_mc='BB2M05')),**  **11, 2,**  **'2024-10-08 09:01', 5.90, 28.1, 50.2, 1009.4,**  **6.20, -75.5730, ST\_SetSRID(ST\_MakePoint(-75.5730,6.2000),4326),**  **'{"sensor":"BB2S05"}'**  **),**  **(**  **(SELECT id\_sensor FROM sensores WHERE codigo\_sensor = 'BB2S05'),**  **(SELECT id\_mc FROM microcontroladores WHERE codigo\_mc = 'BB2M05'),**  **(SELECT id\_linea FROM microcontroladores WHERE codigo\_mc = 'BB2M05'),**  **(SELECT id\_fabrica FROM lineas\_produccion WHERE id\_linea = (SELECT id\_linea FROM microcontroladores WHERE codigo\_mc='BB2M05')),**  **11, 2,**  **'2024-10-08 09:02', 6.10, 28.2, 50.3, 1009.3,**  **6.20, -75.5730, ST\_SetSRID(ST\_MakePoint(-75.5730,6.2000),4326),**  **'{"sensor":"BB2S05"}'**  **),**  **(**  **(SELECT id\_sensor FROM sensores WHERE codigo\_sensor = 'BB2S05'),**  **(SELECT id\_mc FROM microcontroladores WHERE codigo\_mc = 'BB2M05'),**  **(SELECT id\_linea FROM microcontroladores WHERE codigo\_mc = 'BB2M05'),**  **(SELECT id\_fabrica FROM lineas\_produccion WHERE id\_linea = (SELECT id\_linea FROM microcontroladores WHERE codigo\_mc='BB2M05')),**  **11, 2,**  **'2024-10-08 09:03', 6.00, 28.2, 50.4, 1009.2,**  **6.20, -75.5730, ST\_SetSRID(ST\_MakePoint(-75.5730,6.2000),4326),**  **'{"sensor":"BB2S05"}'**  **),**  **(**  **(SELECT id\_sensor FROM sensores WHERE codigo\_sensor = 'BB2S05'),**  **(SELECT id\_mc FROM microcontroladores WHERE codigo\_mc = 'BB2M05'),**  **(SELECT id\_linea FROM microcontroladores WHERE codigo\_mc = 'BB2M05'),**  **(SELECT id\_fabrica FROM lineas\_produccion WHERE id\_linea = (SELECT id\_linea FROM microcontroladores WHERE codigo\_mc='BB2M05')),**  **11, 2,**  **'2024-10-08 09:04', 5.85, 28.1, 50.6, 1009.1,**  **6.20, -75.5730, ST\_SetSRID(ST\_MakePoint(-75.5730,6.2000),4326),**  **'{"sensor":"BB2S05"}'**  **),**  **-- =====================================================================**  **-- LECTURAS 11–15 → SENSOR ALTO (CC3S10 / CC3M10) → NIVEL 3 → TURNO 10**  **-- =====================================================================**  **(**  **(SELECT id\_sensor FROM sensores WHERE codigo\_sensor = 'CC3S10'),**  **(SELECT id\_mc FROM microcontroladores WHERE codigo\_mc = 'CC3M10'),**  **(SELECT id\_linea FROM microcontroladores WHERE codigo\_mc = 'CC3M10'),**  **(SELECT id\_fabrica FROM lineas\_produccion WHERE id\_linea = (SELECT id\_linea FROM microcontroladores WHERE codigo\_mc='CC3M10')),**  **10, 3,**  **'2024-10-09 15:20', 25.12, 30.2, 48, 1008.3,**  **6.30, -75.5500, ST\_SetSRID(ST\_MakePoint(-75.5500,6.3000),4326),**  **'{"sensor":"CC3S10"}'**  **),**  **(**  **(SELECT id\_sensor FROM sensores WHERE codigo\_sensor = 'CC3S10'),**  **(SELECT id\_mc FROM microcontroladores WHERE codigo\_mc = 'CC3M10'),**  **(SELECT id\_linea FROM microcontroladores WHERE codigo\_mc = 'CC3M10'),**  **(SELECT id\_fabrica FROM lineas\_produccion WHERE id\_linea = (SELECT id\_linea FROM microcontroladores WHERE codigo\_mc='CC3M10')),**  **10, 3,**  **'2024-10-09 15:21', 26.05, 30.3, 48.1, 1008.2,**  **6.30, -75.5500, ST\_SetSRID(ST\_MakePoint(-75.5500,6.3000),4326),**  **'{"sensor":"CC3S10"}'**  **),**  **(**  **(SELECT id\_sensor FROM sensores WHERE codigo\_sensor = 'CC3S10'),**  **(SELECT id\_mc FROM microcontroladores WHERE codigo\_mc = 'CC3M10'),**  **(SELECT id\_linea FROM microcontroladores WHERE codigo\_mc = 'CC3M10'),**  **(SELECT id\_fabrica FROM lineas\_produccion WHERE id\_linea = (SELECT id\_linea FROM microcontroladores WHERE codigo\_mc='CC3M10')),**  **10, 3,**  **'2024-10-09 15:22', 24.95, 30.4, 48.2, 1008.1,**  **6.30, -75.5500, ST\_SetSRID(ST\_MakePoint(-75.5500,6.3000),4326),**  **'{"sensor":"CC3S10"}'**  **),**  **(**  **(SELECT id\_sensor FROM sensores WHERE codigo\_sensor = 'CC3S10'),**  **(SELECT id\_mc FROM microcontroladores WHERE codigo\_mc = 'CC3M10'),**  **(SELECT id\_linea FROM microcontroladores WHERE codigo\_mc = 'CC3M10'),**  **(SELECT id\_fabrica FROM lineas\_produccion WHERE id\_linea = (SELECT id\_linea FROM microcontroladores WHERE codigo\_mc='CC3M10')),**  **10, 3,**  **'2024-10-09 15:23', 25.50, 30.4, 48.3, 1008.0,**  **6.30, -75.5500, ST\_SetSRID(ST\_MakePoint(-75.5500,6.3000),4326),**  **'{"sensor":"CC3S10"}'**  **),**  **(**  **(SELECT id\_sensor FROM sensores WHERE codigo\_sensor = 'CC3S10'),**  **(SELECT id\_mc FROM microcontroladores WHERE codigo\_mc = 'CC3M10'),**  **(SELECT id\_linea FROM microcontroladores WHERE codigo\_mc = 'CC3M10'),**  **(SELECT id\_fabrica FROM lineas\_produccion WHERE id\_linea = (SELECT id\_linea FROM microcontroladores WHERE codigo\_mc='CC3M10')),**  **10, 3,**  **'2024-10-09 15:24', 27.10, 30.5, 48.5, 1007.9,**  **6.30, -75.5500, ST\_SetSRID(ST\_MakePoint(-75.5500,6.3000),4326),**  **'{"sensor":"CC3S10"}'**  **),**  **-- =====================================================================**  **-- LECTURAS 16–20 → SENSOR CRÍTICO (AA4S20 / AA4M20) → NIVEL 4 → TURNO 12**  **-- =====================================================================**  **(**  **(SELECT id\_sensor FROM sensores WHERE codigo\_sensor = 'AA4S20'),**  **(SELECT id\_mc FROM microcontroladores WHERE codigo\_mc = 'AA4M20'),**  **(SELECT id\_linea FROM microcontroladores WHERE codigo\_mc = 'AA4M20'),**  **(SELECT id\_fabrica FROM lineas\_produccion WHERE id\_linea = (SELECT id\_linea FROM microcontroladores WHERE codigo\_mc='AA4M20')),**  **12, 4,**  **'2024-10-10 02:10', 420.9, 36, 60, 1005,**  **6.2442, -75.5636, ST\_SetSRID(ST\_MakePoint(-75.5636,6.2442),4326),**  **'{"sensor":"AA4S20"}'**  **),**  **(**  **(SELECT id\_sensor FROM sensores WHERE codigo\_sensor = 'AA4S20'),**  **(SELECT id\_mc FROM microcontroladores WHERE codigo\_mc = 'AA4M20'),**  **(SELECT id\_linea FROM microcontroladores WHERE codigo\_mc = 'AA4M20'),**  **(SELECT id\_fabrica FROM lineas\_produccion WHERE id\_linea = (SELECT id\_linea FROM microcontroladores WHERE codigo\_mc='AA4M20')),**  **12, 4,**  **'2024-10-10 02:11', 415.8, 36.1, 60.3, 1004.9,**  **6.2442, -75.5636, ST\_SetSRID(ST\_MakePoint(-75.5636,6.2442),4326),**  **'{"sensor":"AA4S20"}'**  **),**  **(**  **(SELECT id\_sensor FROM sensores WHERE codigo\_sensor = 'AA4S20'),**  **(SELECT id\_mc FROM microcontroladores WHERE codigo\_mc = 'AA4M20'),**  **(SELECT id\_linea FROM microcontroladores WHERE codigo\_mc = 'AA4M20'),**  **(SELECT id\_fabrica FROM lineas\_produccion WHERE id\_linea = (SELECT id\_linea FROM microcontroladores WHERE codigo\_mc='AA4M20')),**  **12, 4,**  **'2024-10-10 02:12', 430.1, 36.3, 60.5, 1004.8,**  **6.2442, -75.5636, ST\_SetSRID(ST\_MakePoint(-75.5636,6.2442),4326),**  **'{"sensor":"AA4S20"}'**  **),**  **(**  **(SELECT id\_sensor FROM sensores WHERE codigo\_sensor = 'AA4S20'),**  **(SELECT id\_mc FROM microcontroladores WHERE codigo\_mc = 'AA4M20'),**  **(SELECT id\_linea FROM microcontroladores WHERE codigo\_mc = 'AA4M20'),**  **(SELECT id\_fabrica FROM lineas\_produccion WHERE id\_linea = (SELECT id\_linea FROM microcontroladores WHERE codigo\_mc='AA4M20')),**  **12, 4,**  **'2024-10-10 02:13', 440.55, 36.5, 60.8, 1004.7,**  **6.2442, -75.5636, ST\_SetSRID(ST\_MakePoint(-75.5636,6.2442),4326),**  **'{"sensor":"AA4S20"}'**  **),**  **(**  **(SELECT id\_sensor FROM sensores WHERE codigo\_sensor = 'AA4S20'),**  **(SELECT id\_mc FROM microcontroladores WHERE codigo\_mc = 'AA4M20'),**  **(SELECT id\_linea FROM microcontroladores WHERE codigo\_mc = 'AA4M20'),**  **(SELECT id\_fabrica FROM lineas\_produccion WHERE id\_linea = (SELECT id\_linea FROM microcontroladores WHERE codigo\_mc='AA4M20')),**  **12, 4,**  **'2024-10-10 02:14', 450.0, 36.6, 61, 1004.6,**  **6.2442, -75.5636, ST\_SetSRID(ST\_MakePoint(-75.5636,6.2442),4326),**  **'{"sensor":"AA4S20"}'**  **);**  **-- 12 Insertar alarmas asociadas a lecturas críticas/altas (usando subqueries)**  **INSERT INTO alarmas (id\_lectura, id\_tipo\_alarma, id\_supervisor, fecha\_hora\_alarma, estado\_alarma, acciones\_tomadas, tiempo\_respuesta)**  **VALUES**  **(**  **(SELECT id\_lectura FROM lecturas WHERE concentracion\_ppm >= 100 ORDER BY fecha\_hora LIMIT 1),**  **(SELECT id\_tipo\_alarma FROM tipos\_alarma WHERE nombre\_tipo = 'Alarma Crítica' LIMIT 1),**  **(SELECT id\_supervisor FROM supervisores LIMIT 1),**  **NOW(),**  **'ACTIVA',**  **'Evacuación iniciada; notificado bomberos.',**  **NULL**  **),**  **(**  **(SELECT id\_lectura FROM lecturas WHERE concentracion\_ppm BETWEEN 10 AND 100 ORDER BY fecha\_hora LIMIT 1),**  **(SELECT id\_tipo\_alarma FROM tipos\_alarma WHERE nombre\_tipo = 'Alarma Alto' LIMIT 1),**  **(SELECT id\_supervisor FROM supervisores LIMIT 1),**  **NOW(),**  **'ATENDIDA',**  **'Detención temporal de línea y revisión de filtro.',**  **12**  **);**  **--13 inserción registros en la tabla empleado\_turno**  **INSERT INTO empleado\_turno (**  **id\_empleado, id\_turno, fecha, id\_linea**  **)**  **VALUES**  **(13, 10, '2024-10-01', 1),**  **(16, 11, '2024-10-01', 6),**  **(13, 12, '2024-10-01', 11),**  **(15, 10, '2024-10-02', 1),**  **(14, 10, '2024-10-03', 5),**  **(15, 11, '2024-10-03', 12),**  **(14, 11, '2024-10-04', 6),**  **(16, 12, '2024-10-04', 3),**  **(13, 10, '2024-10-05', 11),**  **(16, 10, '2024-10-06', 8);**  **--14 inserción registros en la tabla historial\_filtros**  **INSERT INTO historial\_filtros (**  **id\_filtro, fecha\_cambio, dias\_uso\_real, motivo\_cambio, id\_supervisor, observaciones**  **)**  **VALUES**  **(1, '2024-09-15 10:30:00', 15, 'Vencimiento del filtro', 13, 'El filtro llegó a su vida útil recomendada.'),**  **(2, '2024-09-20 14:10:00', 12, 'Lecturas fuera de rango', 14, 'El sensor reportó aumento de partículas, se realizó cambio preventivo.'),**  **(3, '2024-10-01 09:00:00', 15, 'Mantenimiento preventivo', 15, 'Cambio programado dentro del plan mensual.'),**  **(4, '2024-10-05 11:45:00', 10, 'Obstrucción parcial', 13, 'Se detectó obstrucción en el flujo; se reemplazó inmediatamente.'),**  **(5, '2024-10-08 08:20:00', 5, 'Daño estructural', 15, 'El filtro presentó deformaciones por impacto en la línea.'),**  **(1, '2024-10-12 07:50:00', 13, 'Lecturas inestables', 13, 'Se verificó el filtro y se encontró saturado.'),**  **(2, '2024-10-15 16:40:00', 15, 'Actualización de componente', 13, 'Se instaló un filtro de mayor capacidad.'),**  **(3, '2024-10-18 10:25:00', 14, 'Auditoría interna', 14, 'Se recomendó cambio tras revisión de calidad.'),**  **(4, '2024-10-20 13:00:00', 9, 'Exceso de humedad', 14, 'El material absorbente superó los niveles tolerables.'),**  **(5, '2024-10-22 06:55:00', 15, 'Cambio rutinario fin de turno', 13, 'Cambio ejecutado durante el cierre del turno nocturno.');**  **-- FIN DEL SCRIPT DE INSERCIÓN** |

**7.2- Pantallazos de consultas SELECT de las tablas pobladas**

**TABLA: alarmas**

****

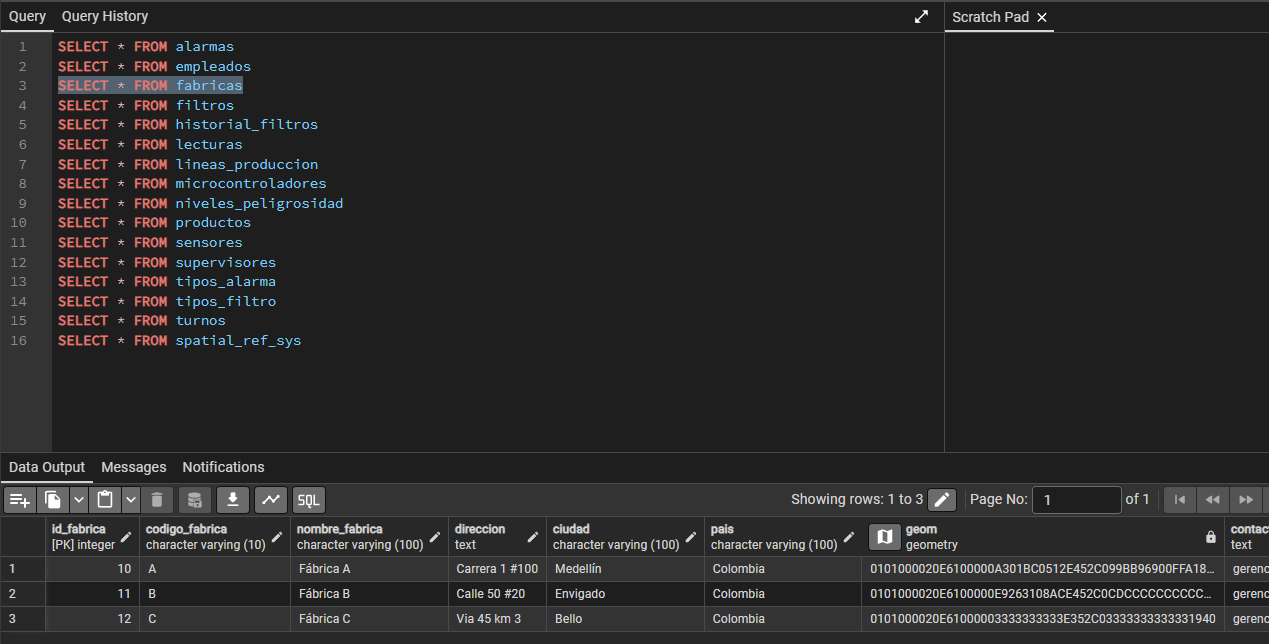
**TABLA: empleados**

****

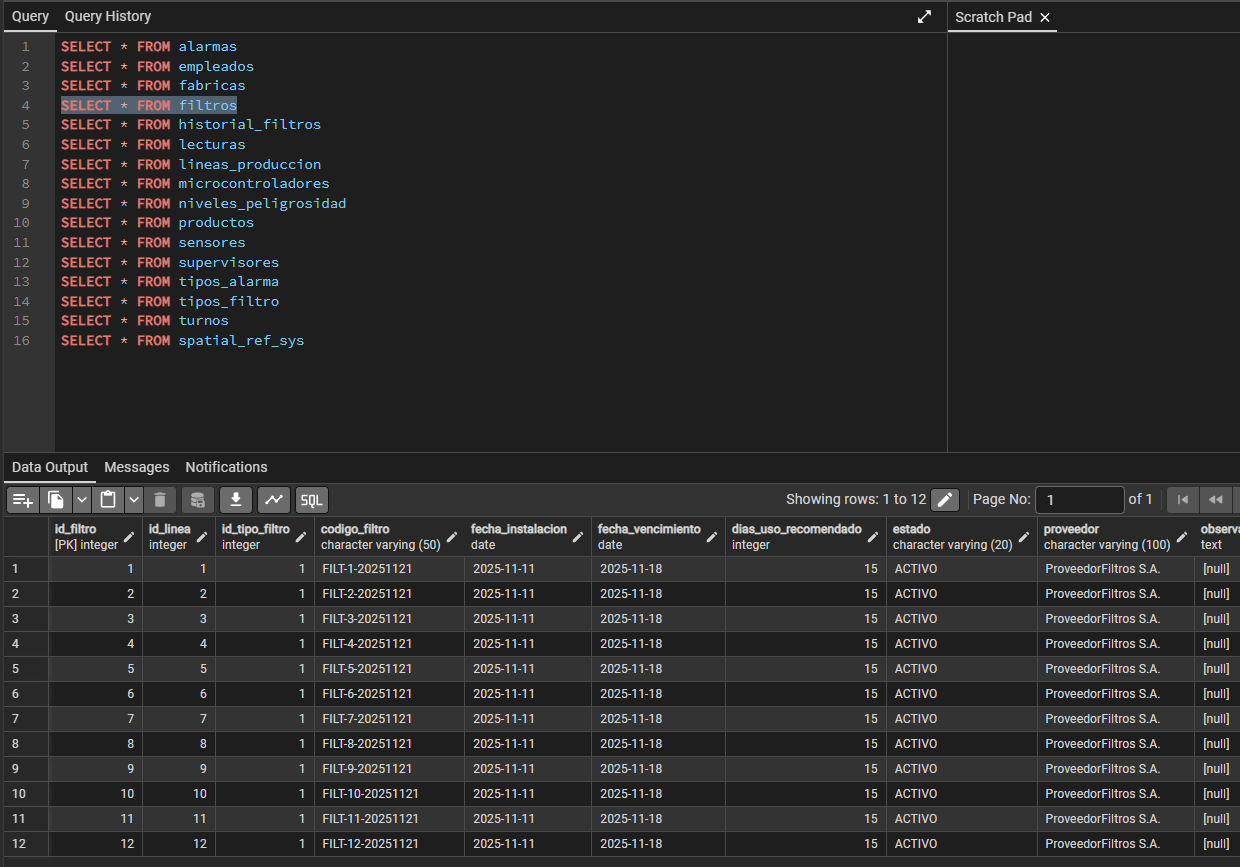
**TABLA: empleado\_turno**

****

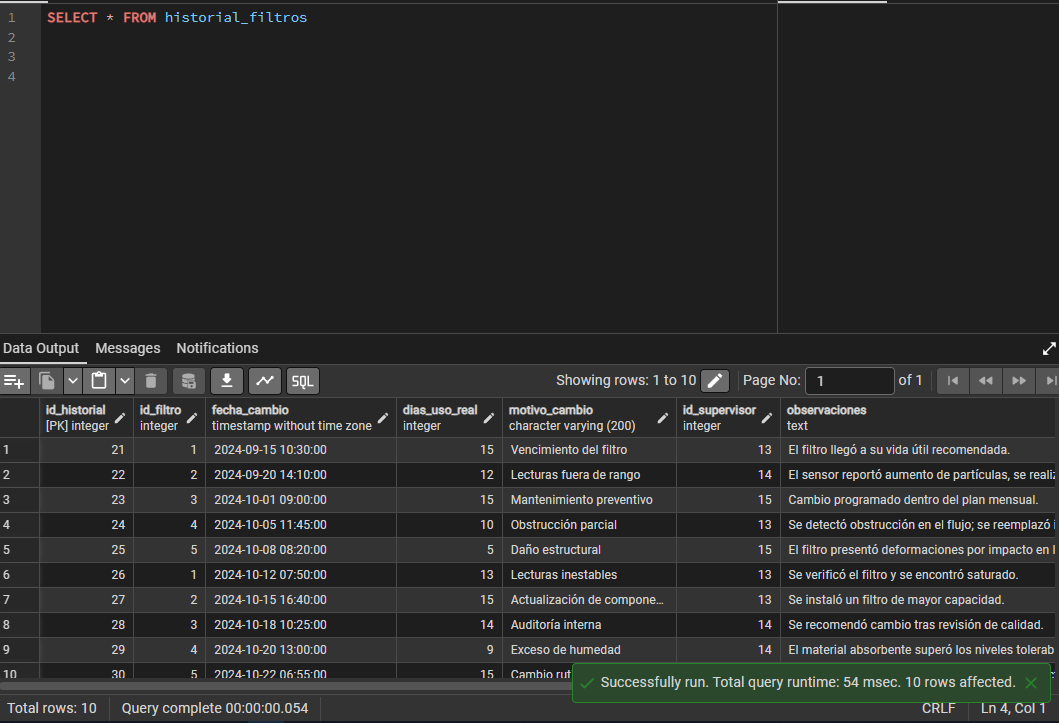
**TABLA: fabricas**

****

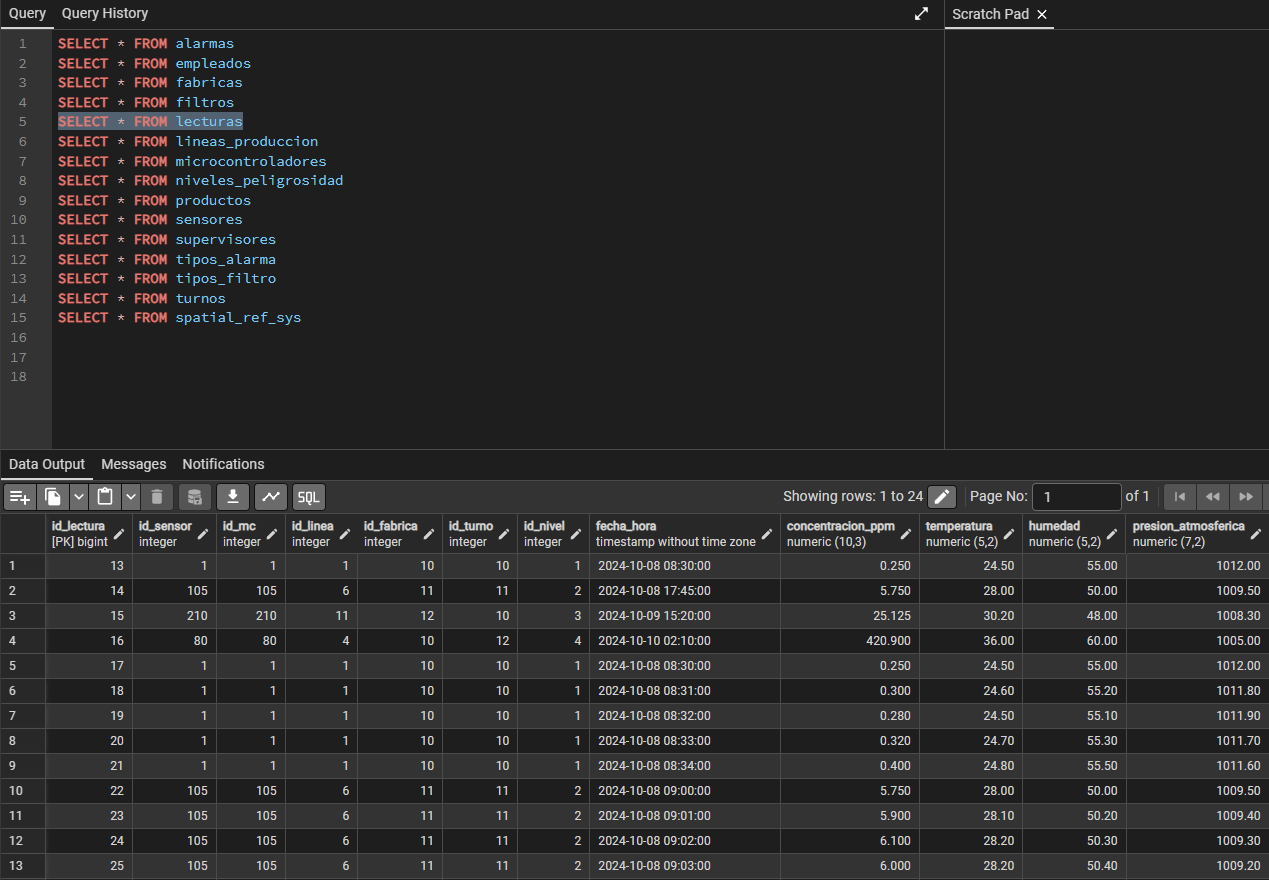
**TABLA: filtros**

****

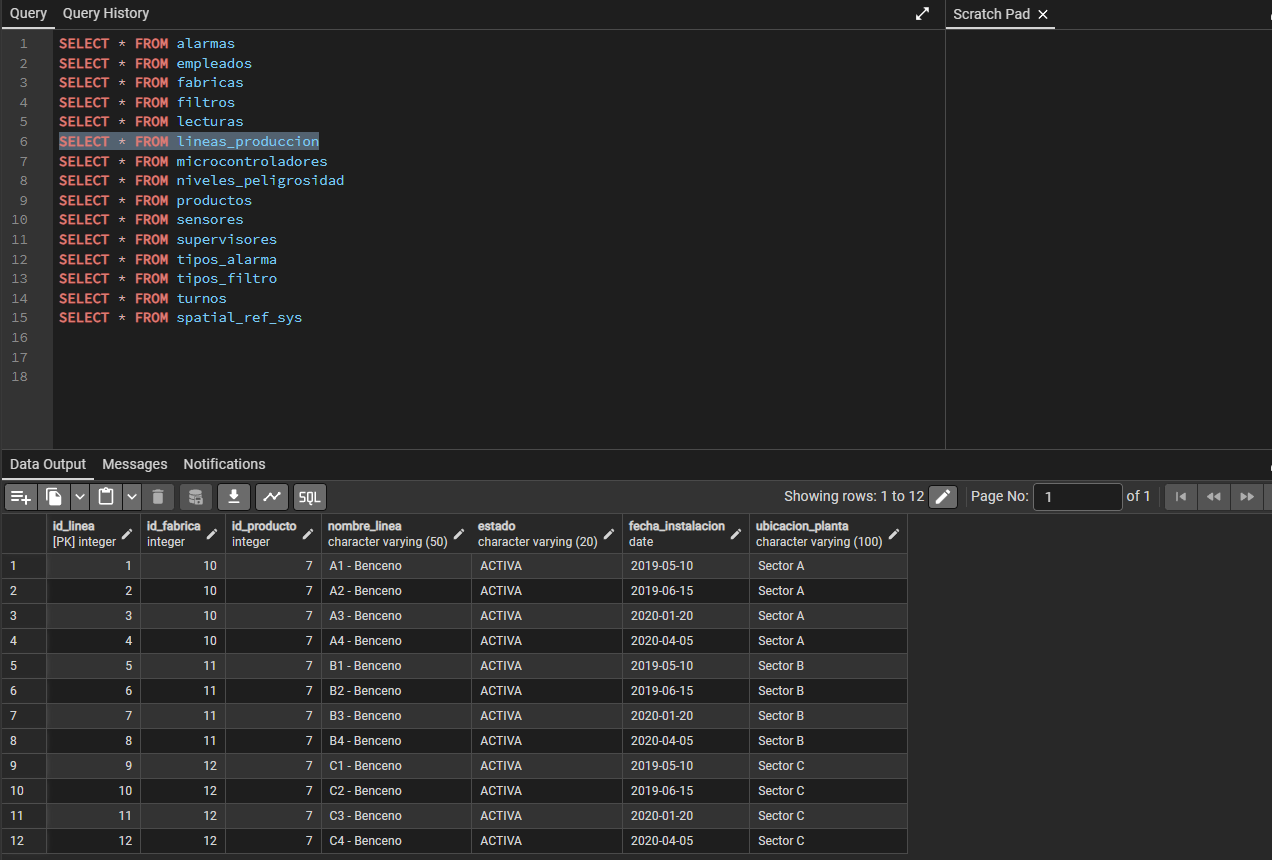
**TABLA: historial\_filtros**

****

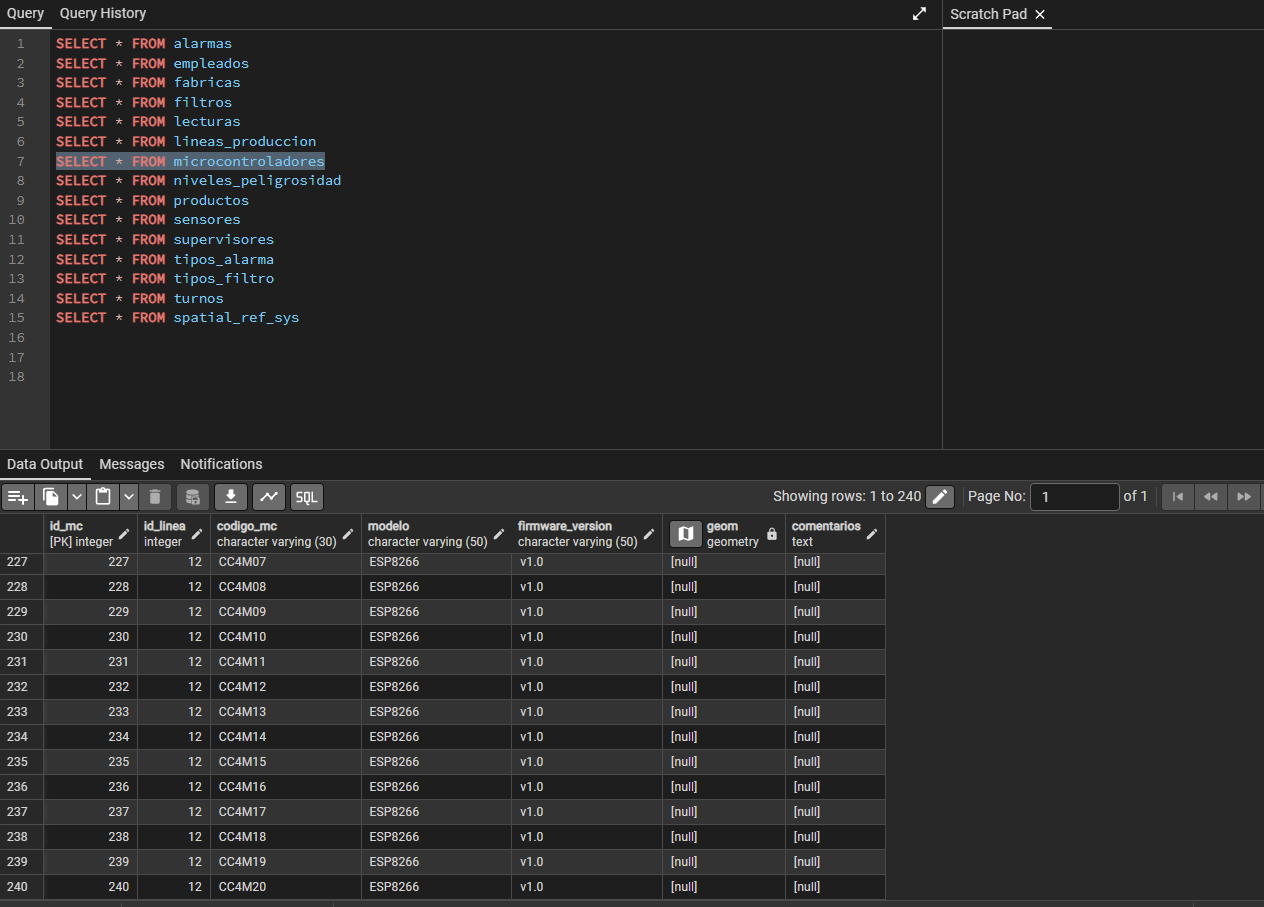
**TABLA: lecturas**

****

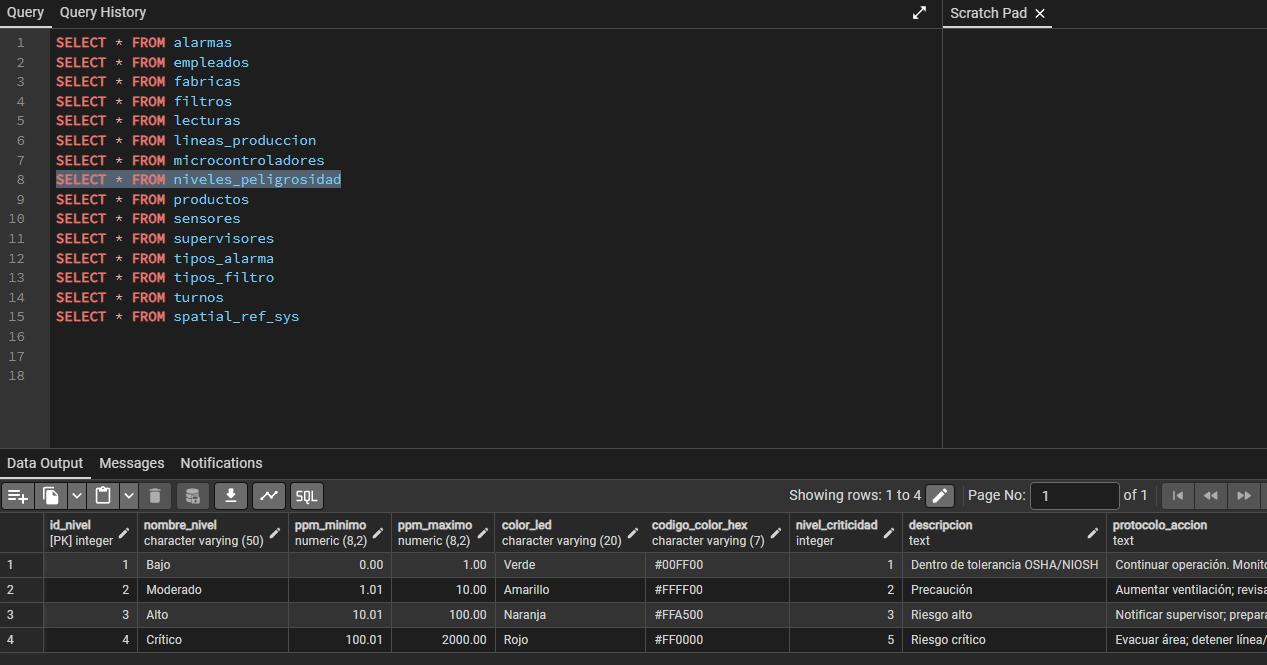
**TABLA: lineas\_produccion**

****

**TABLA: microcontroladores**



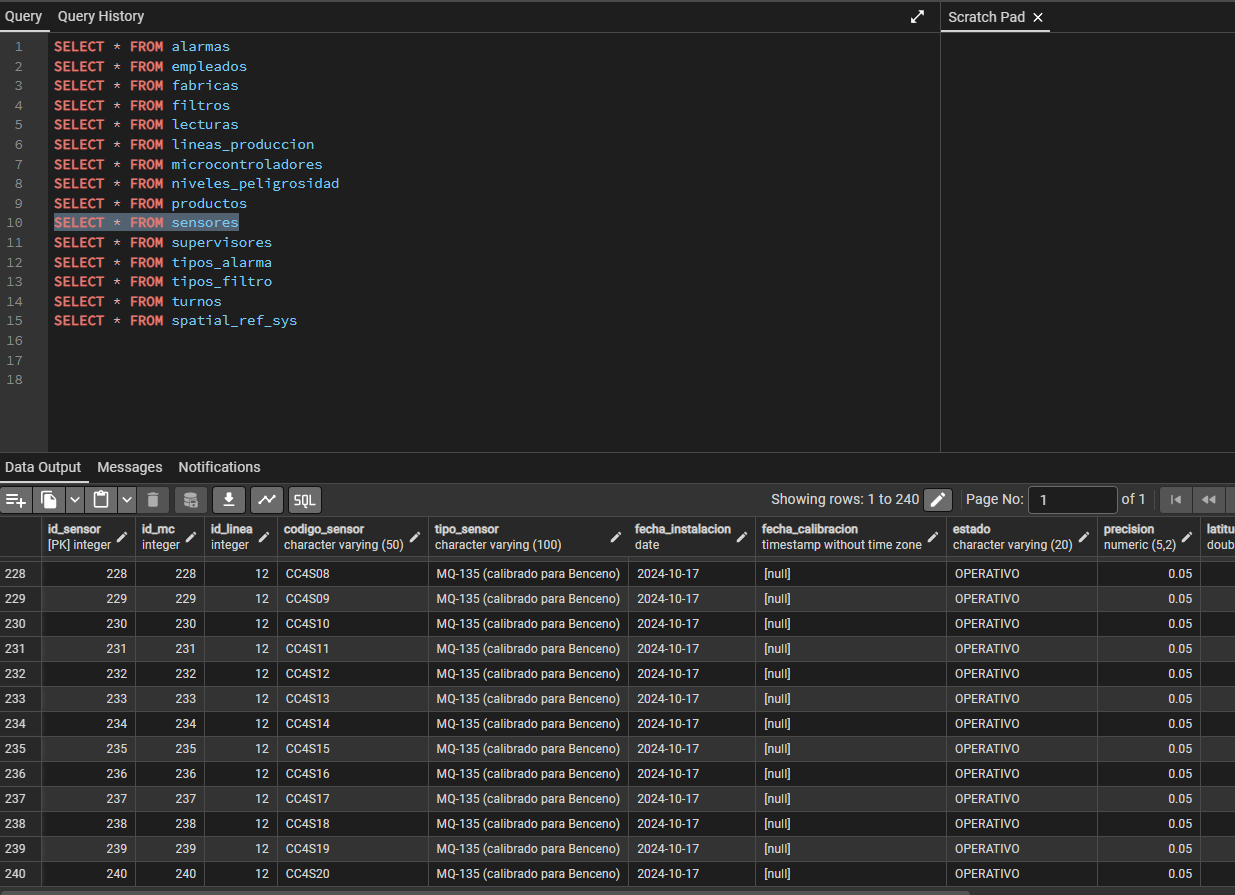
**TABLA: niveles\_peligrosidad**



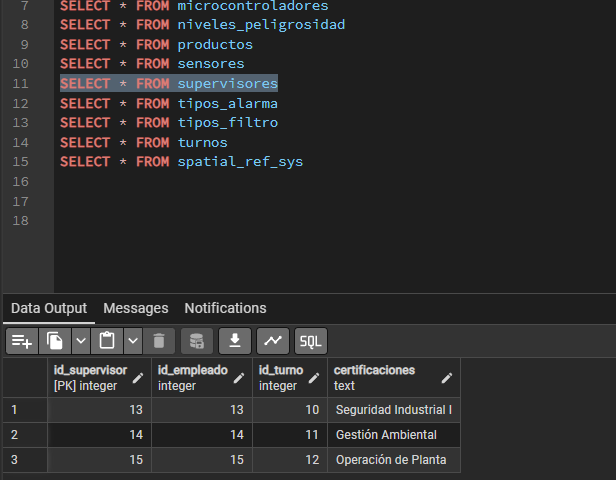
**TABLA: productos**



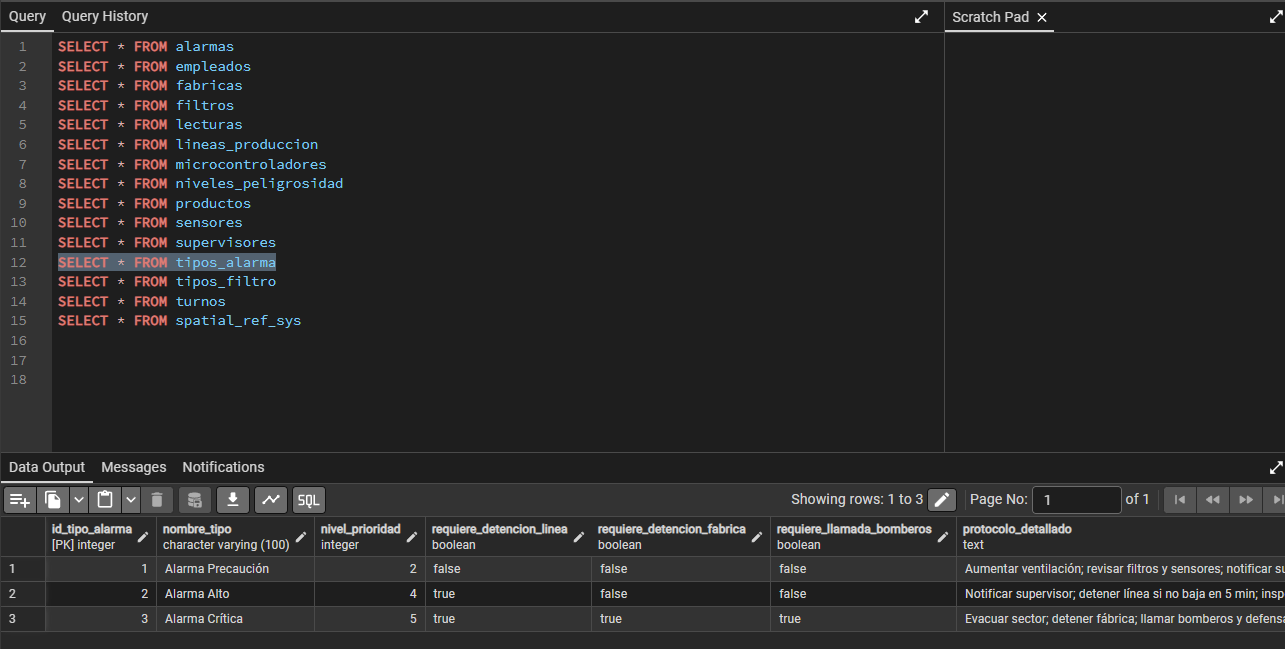
**TABLA: sensores**



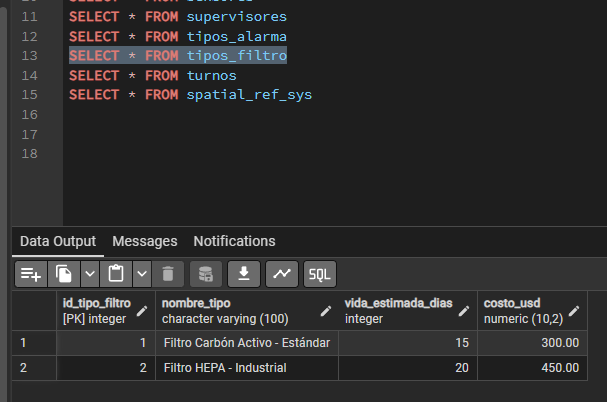
**TABLA: supervisores**

****

**TABLA: tipos\_alarma**

****

**TABLA: tipos\_filtro**

****

**TABLA: turnos**

****

**8.- Conclusiones.**

Consideramos que esta tarea fue fundamental para comprender la importancia de planear correctamente la arquitectura de un sistema de información y de analizar de forma detallada los requerimientos de almacenamiento, capacidad de cómputo y procesamiento de datos. Durante el desarrollo del trabajo, logramos entender cómo los datos generados por los sensores, cuando se proyectan a escenarios reales de producción, pueden crecer a niveles que sobrepasan fácilmente las capacidades de las herramientas tradicionales, lo cual evidencia la necesidad de soluciones más robustas y escalables.

Este ejercicio también nos permitió fortalecer conceptos claves vistos en la asignatura, como el diseño de estructuras de datos, cálculos de almacenamiento, arquitectura ETL y estimaciones de capacidad. Todo esto nos acercó a los retos reales que enfrentan las empresas que trabajan con grandes volúmenes de información.

Al inicio del curso esperábamos aprender únicamente sobre el manejo de datos, pero con esta tarea comprendimos que la planeación, la analítica y la estimación de recursos son igual de importantes que la recolección misma. Gracias a esta experiencia, ahora tenemos una visión más completa y profesional sobre cómo se construyen los sistemas de monitoreo industrial y cómo se toman decisiones basadas en datos.

En conjunto, este trabajo fortaleció nuestras habilidades colaborativas y nuestra capacidad para dividir, analizar y resolver un problema técnico realista. Sin duda, los conocimientos adquiridos aquí aportarán significativamente a nuestro desarrollo profesional y nos preparan mejor para los retos laborales relacionados con bases de datos, analítica y arquitecturas de sistemas.

[MARIA CAMILA ORTIZ](mailto:maria.ortiz346@pascualbravo.edu.co)

A lo largo de esta tarea pude entender de una forma mucho más clara cómo funciona un sistema que maneja grandes volúmenes de datos en un entorno real. Al principio, cuando empezamos la asignatura, yo tenía la idea de que Big Data era algo muy técnico y lejano, como solo usado por grandes empresas. Sin embargo, con este proyecto me di cuenta de que todos los pasos, desde la captura de datos hasta su análisis y visualización, siguen una lógica que se puede aprender y aplicar si se tiene un buen diseño y se entiende el propósito de cada componente.

[ALEXANDER VARGAS MEJIA](mailto:alexander.vargas581@pascualbravo.edu.co)

Durante el desarrollo de este proyecto pude entender mejor cómo funcionan los datos dentro de un sistema real, especialmente en algo tan delicado como el control de gases en una planta industrial. Al principio muchos conceptos me parecían complicados, pero a medida que avanzamos fui aprendiendo la importancia de organizar bien la información, seguir un orden y asegurar que los datos estén completos y sean confiables.

Este trabajo me ayudó a ver que el Gobierno de Datos no es solo teoría, sino algo que realmente se necesita para que un sistema funcione correctamente y pueda tomar decisiones a tiempo. También entendí cómo se relacionan las lecturas de los sensores con las bases de datos y los procesos que permiten analizarlas después. Siento que este proyecto me aportó conocimientos útiles que podré aplicar más adelante, tanto en estudios como en mi vida profesional, porque cada vez es más necesario saber manejar datos de forma adecuada y responsable.

[JUAN DAVID GARCIA VARGAS](mailto:juan.garcia567@pascualbravo.edu.co)

Hacer este trabajo fue clave, porque nos enseñó que el diseño de la arquitectura Big Data y su buena Gobernanza no es solo teoría, es la base para que el monitoreo de gases en "Sustancias Locas" funcione de verdad. Al principio, solo esperábamos manejar herramientas, pero al final, entendimos que el reto real es la planificación detallada y tener una arquitectura híbrida (PostgreSQL y AWS) que haga que los datos sean confiables. Sinceramente, este proyecto nos dio una base muy sólida. Sabemos que esta experiencia integral nos abre muchas puertas laborales, porque ya podemos meternos de lleno a diseñar soluciones complejas y responsables, que es lo que el mundo laboral de hoy necesita.

[JAIME DAVID RESTREPO RODRIGUEZ](mailto:jaime.restrepo413@pascualbravo.edu.co)

En esta tarea aprendí como es el manejo de la información en contextos de la vida real donde hay una gran cantidad de datos y requieren ser tratados con mucho cuidado, especialmente en algo tan delicado como el control de gases en una fábrica para que todo funcione, tiene que ser muy ordenado con los detalles y asegurarse de que cada dato que entra sea fiable al cien por cien. Aprendí que la clave está en la organización y en la calidad de lo que se maneja.

Todo esto nos aportó la experiencia para poder aplicar lo aprendido y saber manejar bien la información para poder tomar buenas decisiones y a tiempo. Entendí perfectamente cómo se conectan las lecturas que dan los aparatos con los archivos digitales y qué pasos hay que seguir para poder entenderlas después. Creo que esta experiencia me aportó una base muy sólida que me servirá mucho más adelante, en mis estudios o en mi empleo, porque lo que realmente vale hoy es saber ser responsable al gestionar los números.

**9.- Video de sustentación:**

LINK:[*https://drive.google.com/drive/folders/1\_DzBKUkH1roDKie\_fOHpdlIRqenKXn18*](https://drive.google.com/drive/folders/1_DzBKUkH1roDKie_fOHpdlIRqenKXn18)