

TRABAJO DE FIN DE GRADO

Memoria Final

Análisis y seguimiento de la calidad del
aire mediante estaciones de control en
la administración pública

Business Intelligence



Rubén Emanuel Rodrigues Morais
Universitat Oberta de Catalunya
Grado Ingeniería Informática



Esta obra está sujeta a una licencia CC BY-NC-ND 4.0
(Attribution-NonCommercial-NoDerivatives 4.0)
Más información sobre la licencia en: [Creative Commons](https://creativecommons.org/licenses/by-nc-nd/4.0/)

© Rubén Emanuel Rodrigues Morais

Reservados todos los derechos. Está prohibido la reproducción total o parcial de esta obra por cualquier medio o procedimiento, comprendidos la impresión, la reprografía, el microfilme, el tratamiento informático o cualquier otro sistema, así como la distribución de ejemplares mediante alquiler y préstamo, sin la autorización escrita del autor o de los límites que autorice la Ley de Propiedad Intelectual.

Ficha del trabajo final

Título del trabajo:	Análisis y seguimiento de la calidad del aire mediante estaciones de control en ayuntamientos
Nombre del autor:	Rubén Emanuel Rodrigues Morais
Nombre del consultor:	Xavier Martínez Fontes
Fecha de entrega:	22/04/2019
Idioma:	Castellano
Área de Trabajo:	Business Intelligence
Titulación:	Grado de Ingeniería Informática

Resumen del Trabajo (Máximo 250 palabras)

La problemática ambiental en la actualidad tiene una dimensión global por lo que involucra a todo el planeta. Uno de los problemas más grandes y peligrosos que existen en la actualidad es la contaminación, la cual destruye la tierra, la naturaleza y termina por destruir al ser humano.

La contaminación puede aparecer y mostrarse en diferentes medios y formas como pueden ser el aire, agua, suelo, etc. En el caso de la contaminación del aire, viene dada por alteraciones de la composición física y química que pueden producirse por polvo o gases excesivos como el dióxido de carbono.

El objetivo del presente trabajo es establecer una serie de indicadores que nos muestren el estado histórico de la contaminación del aire, que serán medidos mediante estaciones de control de gases, en las distintas comunidades autónomas donde se implante nuestro proyecto.

En base a los indicadores que estableceremos, desarrollaremos un cuadro de mando integral para que la alta dirección y los responsables medioambientales puedan tomar las decisiones óptimas en base a la toma de datos realizada.

Abstract (In English, 250 word or less)

Today's environmental problems have a global dimension that involves the whole planet. One of the biggest and most dangerous problems that exist today is pollution, which destroys the earth, nature and ends up destroying human beings.

In the case of air pollution, it is caused by alterations in the physical and chemical composition that can be produced by dust or excessive gases such as carbon dioxide.

The objective of this work is to establish a series of indicators that show the historical state of air pollution, which will be measured by gas control stations in the different autonomous communities where our project is implemented.

Based on the indicators we will establish, we will develop a dashboard so that senior management and environmental managers can make the best decisions based on the data taken.

Palabras clave (entre 4 y 8)

Business Intelligence, Dashboard, Data Warehouse, KPI, ETL, Ayuntamiento, Contaminación

Agradecimientos

Quisiera agradecer el apoyo y ayuda recibida por mi familia: mis padres, mi hermano, mis suegros y mis cuñadas y en especial a mi pareja, que fue la que me animó a empezar a estudiar esta carrera para quitarme esa espinita clavada que tenía.

Quisiera destacar que sin ella no habría podido sacar adelante esta ingeniería y aprovechar para decirle que todo lo malo pasa y que sin esfuerzo no existe recompensa (**Never give up!**). Se que es la que más ha sufrido mi ausencia, durante largas sesiones de estudio y trabajo para poder cumplir con el objetivo que me impuse hace ya tres años.

Por otro lado, me gustaría agradecer a tres grandes compañeros y amigos que he tenido en esta aventura Josh, Tony y Flo los cuales me han ayudado en todo momento con dudas, sesiones interactivas de estudio y me han animado en los momentos más difíciles de la carrera, en los cuales estuve a punto de abandonar.

Por último, agradecer el apoyo a mi consultor Xavier Martínez, por guiarme en este proyecto y darme la idea para poder llevarlo a cabo.

Índice

Ficha del trabajo final.....	3
Agradecimientos	5
1. Control de Versiones	10
2. Introducción	11
2.1 Contexto y Justificación del trabajo	11
2.2 Objetivos del trabajo.....	12
2.3 Alcance de los objetivos del trabajo	12
2.4 Enfoque y método seguido	13
2.5 Planificación temporal	13
2.6 Breve resumen de productos a obtener	15
3. Análisis del proyecto	16
3.1 Antecedentes	16
3.2 ¿Cómo cumplimos esto?	17
3.3 Principales Contaminantes.....	17
3.4 Efectos en la salud.....	19
3.5 ¿Qué podemos hacer frente a esto?.....	20
3.6 Definición de Indicadores (KPI)	20
3.6.1 Indicadores de calidad del aire según contaminante	21
3.6.2 Indicadores de calidad del aire según localización de estación autónoma	27
3.6.3 Indicadores de calidad del aire según representaciones graficas.....	28
4. Diseño del proyecto	28
4.1 Orígenes de datos	29
4.2 Investigación y selección de herramientas BI	29
4.2.1 Procesos ETL.....	30
4.2.2 DataWarehouse	32
4.2.3 Creación de CMI BI (Reporting).....	33
4.2.4 Características que debe tener nuestro software.....	34
4.2.5 Decisión y elección de software final.....	35
4.2 Arquitectura	36
4.3 DataSource	37
4.4 Extracción, transformación y carga de datos (ETL).....	37
4.5 Base de datos	37
4.5.1 Tabla Contaminantes	38
4.5.2 Tabla EstacionesControl.....	39

4.5.3 Tabla DatosDiarios	40
4.5.4 Tabla DatosTiempoReal	40
4.5.5 Tabla DatosHorarios.....	41
4.6 Cuadro de Mando Integral	42
4.6.1 Vista rápida de la calidad del aire en tiempo real.....	42
4.6.1 Datos en tiempo real.....	42
4.6.2 Datos Horarios.....	43
4.6.3 Datos Diarios	44
4.6.4 Gráficas.....	44
5. Implementación del proyecto	45
5.1 El entorno de trabajo	45
5.2 Software <i>PowerBI Desktop</i>	48
5.3 Obtención del <i>DataSource</i>	49
5.4 Creación de nuestro <i>DataWarehouse</i>	50
5.4.1 Conexión con el <i>Datasource</i>	51
5.4.2 Creación de tablas.....	52
5.5 Procesos ETL.....	55
5.6 Cuadro de Mando Integral (CMI)	56
6. Distribución en plataforma web	65
7. Conclusiones	68
8. Líneas de futuro	69
9. Glosario	70
10. Anexos.....	71
10.1 Conjunto de datos.....	71
10.2 Tablas	73
10.3 Procesos ETL.....	74
11. Bibliografía	88

Índice de Ilustraciones

Ilustración 1-Valores y límite contaminante S02 (Comunidad de Madrid).....	18
Ilustración 2-Valores y límite contaminante N02 (Comunidad de Madrid).....	18
Ilustración 3-Valores y límite contaminante N0x (Comunidad de Madrid)	18
Ilustración 4-Valores y límite contaminante PM10 (Comunidad de Madrid)	18
Ilustración 5 - Cuadrante mágico de Gartner 2019.....	30
Ilustración 6 - Arquitectura del proyecto.....	36
Ilustración 7 - Proceso ETL datos en tiempo real.....	37
Ilustración 8 - Modelo lógico.....	38
Ilustración 9 - Modelo lógico: Tabla contaminantes.....	38
Ilustración 10 - Modelo lógico: Tabla de estaciones de control	39
Ilustración 11 - Modelo lógico: Tabla de datos diarios	40
Ilustración 12 - Modelo lógico: Tabla de datos en tiempo real	40
Ilustración 13 - Modelo lógico: Tabla de datos horarios.....	41
Ilustración 14 - CMI: Vista rápida de la calidad del aire en tiempo real	42
Ilustración 15 - CMI: Datos en tiempo real	43
Ilustración 16 - CMI: Datos Horarios	43
Ilustración 17 - CMI: Datos Diarios.....	44
Ilustración 18 - CMI: Gráficos.....	44
Ilustración 19 - Información maquina anfitrión.....	45
Ilustración 20 - Windows Azure: Windows Server 2019.....	46
Ilustración 21 - Activar Hyper-V en Windows 10	46
Ilustración 22 - Creación Máquina Virtual WinServer16	47
Ilustración 23 - Instalación WinServer19	47
Ilustración 24 - Instalación de PowerBI Desktop	48
Ilustración 25 – Inicio PowerBI Desktop	49
Ilustración 26 - Repositorio abierto de datos CM.....	50
Ilustración 27 - Obtener datos PowerBI.....	51
Ilustración 28 - Carga de datos mediante URL.....	52
Ilustración 29 - Creación de tabla en PowerBI (Interfaz)	53
Ilustración 30 - Creación tabla PowerBI (Código)	53
Ilustración 31 - Relaciones entre tablas PowerBI.....	54
Ilustración 32- Relaciones entre tablas UML	54
Ilustración 33- Proceso ETL para DatosHorarios2019	56
Ilustración 34 - Proceso ETL para anexar a tabla principal DatosHorarios	56
Ilustración 35 - Vista rápida de la calidad del aire en tiempo real (Versión escritorio).....	57
Ilustración 36 - Vista rápida de la calidad del aire en tiempo real (Versión móvil)	58
Ilustración 37 -Datos de la calidad del aire en tiempo real (Versión escritorio).....	59
Ilustración 38 -Datos de la calidad del aire en tiempo real (Versión móvil)	59
Ilustración 39 -Datos horarios de la calidad del aire en tiempo real (Versión escritorio)	60
Ilustración 40 -Datos horarios de la calidad del aire en tiempo real (Versión móvil).....	61
Ilustración 41 -Datos diarios de la calidad del aire en tiempo real (Versión escritorio).....	62
Ilustración 42 -Datos diarios de la calidad del aire en tiempo real (Versión móvil)	62
Ilustración 43 -Gráficos de la calidad del aire en tiempo real (Versión escritorio).....	63
Ilustración 44 -Gráficos de la calidad del aire en tiempo real (Versión móvil)	64
Ilustración 45 - Pagina Inicio de la web.....	65
Ilustración 46 - Pagina de KPI de la web	66

Ilustración 47 - Pagina de CMI de la web 67

Índice de Tablas

Tabla 1 - Planificación del proyecto	15
Tabla 2 - Fechas de los entregables	15
Tabla 3 – Efectos de la contaminación en los humanos	20
Tabla 4 - Calidad del aire en base a la medición de PM ₁₀	21
Tabla 5 - Resumen Indicador: Medición de PM ₁₀	22
Tabla 6 - Calidad del aire en base a la medición de PM _{2.5}	22
Tabla 7 - Resumen Indicador: Medición de PM _{2.5}	22
Tabla 8 - Calidad del aire en base a la medición del Dióxido de Azufre	23
Tabla 9 - Resumen Indicador: Medición de Dióxido de Azufre	23
Tabla 10 - Calidad del aire en base a la medición del Dióxido de Azufre.....	24
Tabla 11 - Resumen Indicador: Medición de Dióxido de Nitrógeno	24
Tabla 12 - Calidad del aire en base a la medición del Monóxido de Carbono	25
Tabla 13 - Resumen Indicador: Medición de Monóxido de Carbono	25
Tabla 14 - Calidad del aire en base a la medición del Monóxido de Carbono	26
Tabla 15 - Resumen Indicador: Medición de Ozono	26
Tabla 16 - Calidad del aire en base a la medición del Benceno	26
Tabla 17 - Resumen Indicador: Medición de Benceno	27
Tabla 18 - Calidad del aire mediante colores	27
Tabla 19 - Resumen Indicador: Geolocalización de estaciones autónomas automáticas	28
Tabla 20 - Resumen Indicador: Realización de gráficos	28
Tabla 21 – Glosario.....	70

1. Control de Versiones

Versión	Descripción	Modifica	Fecha de Modificación	Revisa	Fecha Revisión	Aprueba	Fecha de Aprobación
1.0	Documento Inicial	Rubén Rodrigues	15/03/2019	Xavier Martínez	31/03/2019	Xavier Martínez	31/03/2019
1.1	Memoria	Rubén Rodrigues	22/04/2019	Xavier Martínez	10/05/2019	Xavier Martínez	10/05/2019
1.2	Memoria	Rubén Rodrigues	27/05/2019	Xavier Martínez	09/06/2019	Xavier Martínez	09/06/2019
1.3	Memoria Final	Rubén Rodrigues	13/06/2019				

2. Introducción

2.1 Contexto y Justificación del trabajo

El acto de respirar nos obliga a un contacto permanente entre el medio ambiente y el aparato respiratorio, es por ello por lo que, esta relación es imprescindible para la vida, aunque nos hacemos vulnerables a los posibles contaminantes que queden suspendidos en el aire que respiramos.

Según lo enunciado en la ley 34/2007 (BOE A. E.-B., Ley 34/2007, 2007), del 15 de noviembre, de calidad del aire y protección de la atmósfera, la atmósfera es un bien común indispensable para la vida, del cual todas las personas tienen el derecho de su uso y disfrute y la obligación de su conservación. Así pues, la contaminación atmosférica de las ciudades se ha convertido en un problema de salud pública universal y es considerada por la Organización Mundial de la Salud (OMS) como una de las prioridades mundiales más importantes en cuanto a salud se refiere.

Según diferentes estudios, la contaminación del aire es el responsable del 1.4% de las muertes en el mundo, siendo la mitad de este impacto debido a las emisiones producidas por vehículos a motor. Este problema afecta sobre todo a países desarrollados, concretamente a las grandes ciudades de estos, pero también afecta a países que están en vías de desarrollo. En las ciudades, la emisión de óxidos nitrosos y particulares en suspensión producen efectos nocivos en los enfermos crónicos con problemas respiratorios, aumentando tanto la necesidad de medicación, como los síntomas de sus enfermedades. Pero esto no solo afecta a personas con enfermedades cardiovasculares, sino también a las funciones pulmonares de los jóvenes sanos, en los que cada día existen más evidencias de la asociación de la aparición del asma con la contaminación atmosférica.

Es por esto por lo que se trata de un tema urgente a abordar y por ello, la administración pública tiene como una de sus principales responsabilidades el de velar por la salud y seguridad de sus ciudadanos. Para ello, se ponen en marcha muchas acciones encaminadas a la reducción y prevención de situaciones de riesgo con el fin de vigilar y mejorar la seguridad ciudadana.

Actualmente cada comunidad autónoma debe disponer de su propio plan de medioambiente y urbanismo, el cual consiste en la prevención del cambio climático, vigilando y mejorando la calidad del aire que respiramos. Esta vigilancia se realiza mediante estaciones remotas automáticas repartidas por las comunidades autónomas que recogen información básica para la correcta medición de niveles de gases y partículas.

Las emisiones y la contaminación se han convertido en un problema de gran alcance en nuestro país, teniendo que establecer medidas preventivas muy estrictas en las grandes ciudades como Madrid o Barcelona, tales como la restricción de vehículos al centro de la ciudad o la activación de protocolos de prevención como pueden ser el plan “Madrid Central” para reducir la contaminación del aire.

2.2 Objetivos del trabajo

El principal objetivo del presente trabajo consiste en que el alumno adquiera los conocimientos y capacidades necesarias para el análisis, diseño y construcción de un sistema de Business Intelligence, así como la familiarización del software necesario para este hecho.

Este sistema de BI deberá dotar a los altos cargos y a los responsables de medioambiente de la administración pública de información sobre la calidad del aire de la comunidad autónoma. Estos datos se podrán filtrar mediante parámetros básicos, con lo que se podrá visualizar cuan buena es la calidad del aire en sus comunidades, tanto históricamente como en tiempo real.

Este sistema de análisis de datos vendrá dado por la realización de un Cuadro de Mando Integral (CMI), el cual será desarrollado con información obtenida del Portal de datos abiertos del ayuntamiento de Madrid, donde tenemos datos históricos desde 2001 hasta la actualidad de todas sus estaciones remotas.

2.3 Alcance de los objetivos del trabajo

Para la consecución del objetivo descrito en el apartado anterior, será necesario alcanzar los siguientes objetivos:

- **Creación de un DataWarehouse**

Debemos crear un almacén de datos donde almacenaremos de manera eficiente toda la información que nos será relevante para nuestro Cuadro de Mando Integral. Esta información vendrá dada de diferentes archivos o bases de datos que se pueden descargar del almacén de datos abierto de la comunidad de Madrid.

Para la implantación de nuestro *DataWarehouse*, optaremos por la utilización de SQL server, ya que se trata de la herramienta que hemos utilizado para otras asignaturas en la carrera. En esta base de datos crearemos las tablas que creamos oportunas y las relacionaremos para su posterior visualización.

- **KPIs**

Se definirán una serie de indicadores que nos permitirán controlar e informar de la situación histórica de la calidad del aire en la comunidad de Madrid. Para la definición de los *KPI* se estudiará los datos que tenemos y una vez analizados procederemos a la definición de estos.

Estos indicadores estarán definidos en la memoria de trabajo.

- **Procesos ETL**

Se establecerán una serie de procesos ETL por los cuales extraeremos la información, la transformaremos y la cargaremos en nuestro *DataWarehouse*.

Para la realización de los procesos ETL, debemos realizar un trabajo de investigación sobre las herramientas que nos serán de utilidad para ello. Este trabajo de investigación vendrá reflejado en la memoria del trabajo.

- **Plataforma de creación CMI**

Seleccionaremos e implementaremos una plataforma de *Business Intelligence* en base a unos criterios de selección que marcaremos posteriormente, con la cual podremos realizar nuestro Cuadro de Mando Integral y que se adapte a las necesidades reales del proyecto. Debemos de realizar un Cuadro de Mando, en el que, según las necesidades del usuario, podremos ver de un solo vistazo una serie de *KPIs* que serán definidos en nuestro trabajo.

En este caso, debemos realizar una investigación de los distintos softwares disponibles en el mercado y elegiremos uno que se adapte a las necesidades. Este trabajo de investigación vendrá reflejado en la memoria.

- **Implementación en un entorno web**

Mediante el estudio de las plataformas de implementación de *Business Intelligence*, valoraremos positivamente la capacidad del software para exportar el *CMI* a un entorno web y poder así vitalizarlo desde cualquier dispositivo y lugar.

2.4 Enfoque y método seguido

Se utilizará una metodología ágil durante todo el proceso de elaboración del proyecto. Se opta por esta metodología debido a que nuestro proyecto tiene un tiempo limitado y es propenso a cambios propuestos por el coordinador del proyecto. Es por ello que, es necesario una metodología flexible para la elaboración de este.

Con esta metodología adoptamos una estrategia de desarrollo incremental, en vez de una planificación y ejecución completa del proyecto. Además, podemos basar la calidad del resultado más en el conocimiento que en la calidad de los procesos empleados. Y, por último, podemos realizar el solapamiento de las diferentes fases de las que dispone el proyecto, en lugar de realizar una tras otra.

Por otro lado, se tratará de llevar una comunicación activa con el consultor del proyecto para valorar las posibles entregas de éste y proponer posibles modificaciones al proyecto.

2.5 Planificación temporal

A continuación, vamos a elaborar una planificación que nos permitirá llevar la implantación del sistema de *Business Intelligence* de una manera ágil y pautada en el tiempo.

Esta planificación está basada en los siguientes puntos clave:

Adopción: Es la fase en la que se considera la necesidad o no de la implantación de un sistema de *Business Intelligence*. En nuestro caso esta fase está superada ya que, hemos decidido ya la implantación de un cuadro de mando.

Selección: Es la fase en la que se selecciona los softwares a utilizar para la implantación del sistema de *Business Intelligence*. Esta fase todavía no está quemada y se detallará la propuesta en la memoria del proyecto.

Implantación: Una vez elegida la herramienta a utilizar, se procederá a la implantación de esta, mediante la configuración de los distintos elementos que entran en el proyecto (*DataWarehouse*, ETL, CMI).

A continuación, establecemos la planificación en el tiempo:

Grupo	Tareas que realizar	Fecha Inicio	Fecha Fin
Definición del proyecto		25/02/2019	06/03/2019
	Videoconferencia	25/02/2019	25/02/2019
	Búsqueda de información	26/02/2019	05/03/2019
	Videoconferencia	05/03/2019	05/03/2019
	Análisis de posibles ideas	06/03/2019	06/03/2019
	Ronda de consultas previas	06/03/2019	06/03/2019
Plan de proyecto		06/03/2019	18/03/2019
	Análisis de la situación	06/03/2019	06/03/2019
	Estudio de fuentes de datos y herramientas	08/03/2019	15/03/2019
	Plan de trabajo	16/03/2019	18/03/2019
Evaluación Continua		18/03/2019	18/03/2019
	Entrega PEC 1	18/03/2019	18/03/2019
Análisis		19/03/2019	03/04/2019
	Investigación y selección de herramientas BI	19/03/2019	26/03/2019
	Investigación Origen de datos (BBDD)	27/03/2019	28/03/2019
	Definición de <i>KPIs</i>	29/03/2019	30/03/2019
	Modelado de datos	31/03/2019	02/04/2019
	Definición de <i>reports</i>	02/04/2019	03/04/2019
Diseño		04/04/2019	15/04/2019
	Estructuración de datos	04/04/2019	06/04/2019
	Obtención de datos actualizados del <i>Website</i>	07/04/2019	07/04/2019
	Extracción de información relevante	08/04/2019	10/04/2019
	Limpieza y Transformación de datos	11/04/2019	13/04/2019
	Realización de Informes y Maquetación	13/04/2019	15/04/2019
Evaluación Continua		15/04/2019	15/04/2019
	Entrega PEC 2	15/04/2019	15/04/2019
Implementación		16/04/2019	20/05/2019
	Carga de datos en sistema gestor de datos	16/04/2019	21/04/2019
	Configuración de herramientas BI	22/04/2019	25/04/2019
	Transformación de datos (Procesos ETL)	26/04/2019	01/05/2019
	Implementación de informes	02/05/2019	20/05/2019
Evaluación Continua		20/05/2019	20/05/2019
	Entrega PEC 3	20/05/2019	20/05/2019
Cierre del proyecto		21/05/2019	16/06/2019

	Realización de memoria de trabajo	21/05/2019	06/06/2019
	Autoinforme de competencias transversales	07/06/2019	09/06/2019
	Presentación Virtual	10/06/2019	16/06/2019
Evaluación continua		17/06/2019	17/06/2019
	Entrega Final	17/06/2019	17/06/2019
	Tribunal de evaluación	25/06/2019	26/06/2019
Gestión del cambio		25/02/2019	26/06/2019
Gestión del proyecto		25/02/2019	26/06/2019

Tabla 1 - Planificación del proyecto

Siguiendo la metodología de la UOC, el plan de estudio del proyecto se basará en la realización y entrega de diferentes partes del proyecto, para finalmente obtener una memoria completa junto a la presentación en vídeo del proyecto final. Además, se terminará con un tribunal de evaluación que revisará el trabajo y lo valorará acorde a conocimientos adquiridos.

Las fechas de presentación de los diferentes entregables son las siguientes:

Grupo	Entrega
PEC 1	18/03/2019
PEC 2	15/04/2019
PEC 3	20/05/2019
Entrega memoria / Producto Final	17/06/2019
Tribunal de evaluación	26/06/2019

Tabla 2 - Fechas de los entregables

2.6 Breve resumen de productos a obtener

El proyecto constará de diferentes entregables que se darán al finalizar el proyecto en formato digital. Estos entregables serán:

Memoria del proyecto: Este es el principal entregable del proyecto, en el que se describirán todo el trabajo realizado y se explicará de una forma clara las distintas fases por las que hemos pasado para la elaboración de nuestro Cuadro de Mando Integral.

CMI: Se entregará en el formato que corresponda el cuadro de mando realizado con los datos previamente cargados para su posible visualización. Además, también se entregará el enlace para su visionado en un entorno web.

Entregable visual: Se entregará un vídeo grabado por el estudiante en el que se explicará el TFG de una forma visual

PowerPoint: Se entregará el PowerPoint que se ha realizado y se ha utilizado para la presentación del vídeo anterior.

3. Análisis del proyecto

3.1 Antecedentes

Durante los años 70, empezó a aparecer en la población una inquietud ecológica, la cual derivó en la realización de diferentes actividades basadas en la medición de diferentes factores y el estudio del impacto de las acciones humanas en el medio ambiente.

En un primer momento estos esfuerzos iban dirigidos hacia la ecología y el cuidado del entorno natural, pero rápidamente se vio que la evolución tecnológica e industrial empezaba a generar un nuevo tipo de sustancias cuyo uso descontrolado podría afectar al entorno, tanto natural como humano.

Así pues, en la industria empezaron a tomarse las primeras medidas, ya que los efectos perjudiciales de las sustancias nuevas que se crearon y se utilizaron sin conocimiento, llevaron a reducir la calidad de vida de los trabajadores, e incluso llegaron a la muerte en alguno de ellos al exponerse a dosis letales.

En consecuencia, los gobiernos empezaron a plantear acciones para la incorrecta utilización de sustancias que tenían efectos dañinos. Se formaron consejos e instituciones que serían los encargados de comenzar con los sistemas de medición y control en el entorno de trabajo, y en consecuencia, se formaron leyes para el control del medio ambiente laboral donde, a partir de ese momento, se controlaría la atmósfera que es respirada por los trabajadores.

Esta serie de acciones y leyes introdujeron en el sistema conceptos nuevos como pueden ser agente contaminante, límite de aire respirado, límite de exposición, dosis letal, ergonomía, vigilancia del ambiente laboral, etc. las cuales llevaron a la evolución de una tecnología orientada a la medición de efectos privados por el contacto con sustancias desconocidas (efectos para el medio ambiente y efectos para la salud humana).

Todos estos análisis se comenzaron a traducir en medidas legales para que las empresas controlasen la exposición de sus trabajadores y sus efectos. Así dio comienzo a la medicina del trabajo y la prevención de riesgos laborales.

Por otra parte, se comenzó a tomar conciencia del beneficio de cuidar el medioambiente y se establecieron unos compromisos globales y legislaciones para el resto de la población. Con esto, se fue creando una legislación que estaría orientada a la mejora de la calidad de vida, no de los trabajadores sino de las personas, mediante la reducción del impacto en el medioambiente de las actividades generadas por los humanos.

La comunidad europea estableció una fecha límite para la reducción de las emisiones de los contaminantes y también unas emisiones mínimas que deben ser garantizadas. Estas emisiones se establecen en base a unas directivas y reales decretos:

- Directiva 96/62/CE, de 27 de septiembre (evaluación y gestión del aire) - (BOE A. E.-B., Directiva 96/62/CE, 1996)

- Directiva 1999/30/CE, de 22 de abril (relativa a los valores límites de SO₂ NO₂, NO_X, PM₁₀ y Pb) - (BOE A. E.-B., Directiva 1999/30/CE, 1999)
- Directiva 2000/69/CE, de 16 de noviembre (C₆H₆ y CO) - (BOE A. E.-B., Directiva 2000/69/CE, 2000)
- Directiva 2002/3/CE, de 12 de febrero (O₃) - (BOE A. E.-B., Directiva 2002/3/CE, 2002)
- Directiva 2004/107/CE del Parlamento Europeo y del Consejo de 15 de diciembre de 2004 relativa al arsénico, el cadmio, el mercurio, el níquel y los hidrocarburos aromáticos policíclicos en el aire ambiente - (BOE A. E.-B., Directiva 2004/107/CE, 2004)
- Real Decreto 1073/2002, de 18 de octubre, sobre evaluación y gestión de la calidad del aire ambiente en relación con el dióxido de azufre, dióxido de nitrógeno, óxidos de nitrógeno, partículas, plomo, benceno y monóxido de carbono (BOE A. E.-B., Real Decreto 1073/2002, 2002)

3.2 ¿Cómo cumplimos esto?

En los ayuntamientos de las grandes ciudades empezaron a aparecer los departamentos dedicados exclusivamente al medio ambiente. Este departamento se responsabilizaría en la gestión de todo lo que está relacionado con el medio ambiente y el impacto en la calidad de vida de sus ciudadanos. Dentro de este departamento existen muchas líneas medioambientales, pero una de ellas y la que vamos a abordar es el área que se dedica a la calidad del aire. La misión de este departamento recae en tomar medidas adecuadas para cumplir la legislación vigente que propuso la comunidad europea.

Así pues, el departamento de calidad del aire se dedica a la medición y el control de la contaminación de las ciudades, con la finalidad de comunicar a los organismos responsables y a los ciudadanos los niveles diarios de contaminación según las zonas, así como las posibles incidencias y riesgos derivados de una inadecuada calidad del aire respirado en las ciudades.

Existen cantidad de sectores que analizar que son los principales sectores contaminantes de las grandes ciudades, como pueden ser:

- El sector de la agricultura y medio natural
- El sector del transporte
- El sector residencial e institucional
- El sector Industrial

3.3 Principales Contaminantes

Generalmente, en ciudades industriales el principal contaminante es el dióxido de azufre (SO₂) debido al uso masivo de carbón en industria y hogares. Este combustible se ha ido sustituyendo paulatinamente y ha provocado una bajada sensible de los índices de SO₂, pero en su lugar, han aparecido otros contaminantes derivados de alto tráfico generado en grandes ciudades.

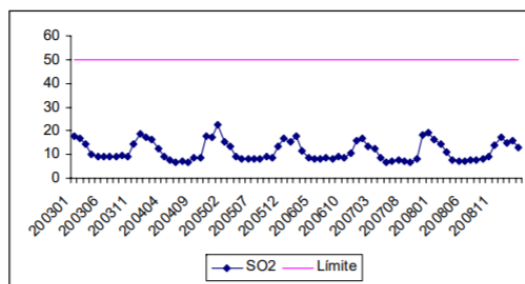


Ilustración 1-Valores y límite contaminante SO2 (Comunidad de Madrid)

En la actualidad los contaminantes más problemáticos son las partículas en suspensión (PM₁₀ o partículas menores a 10 micras), los dióxidos de nitrógeno (NO₂) y los óxidos de Nitrógeno (No_x) ya que en los últimos años vienen superando valores límite de protección a la salud humana.

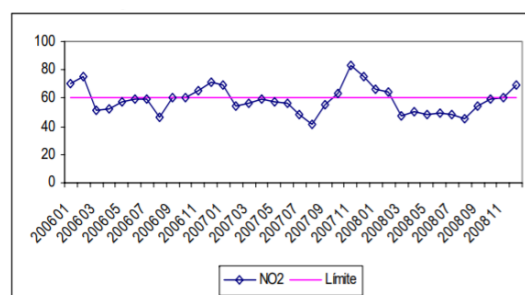


Ilustración 2-Valores y límite contaminante NO2 (Comunidad de Madrid)

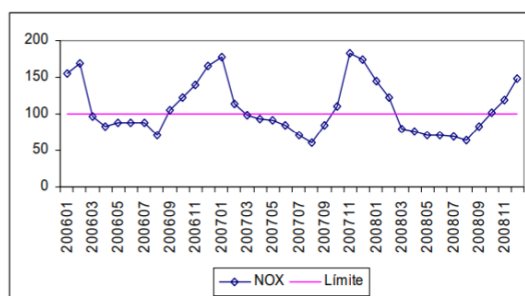


Ilustración 3-Valores y límite contaminante NOx (Comunidad de Madrid)

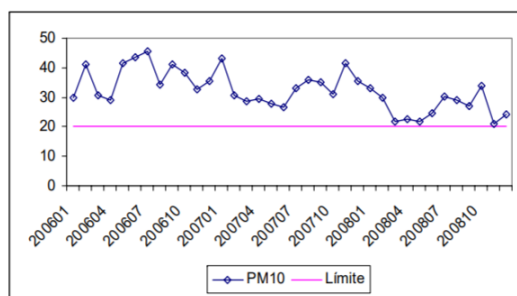


Ilustración 4-Valores y límite contaminante PM10 (Comunidad de Madrid)

3.4 Efectos en la salud

De los estudios realizados en diferentes comunidades autónomas se deduce que el sector emisor más problemático es el del transporte, el cual es responsable del 80% de la contaminación.

Los contaminantes más problemáticos que proceden de este sector son los óxidos de nitrógeno (NOx y NO2). Estos contaminantes afectan a los tramos más profundos de los pulmones, inhibiendo funciones de estos, como la respuesta inmunológica y produciendo una reducción de la resistencia a infecciones. Esto afecta gravemente a los niños y asmáticos. La contaminación por partículas está directamente relacionada con incrementos en la mortalidad total, mortalidad por enfermedades cardiovasculares y respiratorias, mortalidad por cáncer de pulmón e ingresos hospitalarios por afecciones cardiovasculares y respiratorias. Además, según estudios, la exposición de partículas en suspensión a largo plazo puede reducir su esperanza de vida entre varios meses y dos años.

Según organismos nacionales e internacionales de salud, así como estudios epidemiológicos realizado por la Organización Mundial de la Salud (OMS), las emisiones del tráfico tienen efectos a corto, medio y largo plazo sobre el bienestar del ciudadano. Estos estudios además señalan que se han encontrado evidencias de imposibilitar concentraciones umbrales por debajo de las cuales no se produce ninguna agresión a la salud.

Los efectos producidos a corto plazo son la irritación de los ojos, garganta y nariz, infecciones respiratorias, ataques al corazón y cerebro, ataques de asma y alteraciones en el bombeo del corazón. Los efectos producidos a largo plazo incluyen un desarrollo pulmonar más lento en niños, deficiencias respiratorias crónicas, cáncer de pulmón, enfermedades del corazón y daño potencial al cerebro y otros órganos.

A continuación, mostramos una tabla con los efectos de la contaminación del aire en la salud humana:

Sistema respiratorio	Sistema cardiovascular
Cambios fisiológicos y estructurales	Cambios fisiológicos y estructurales
Volumen y flujo de la espiración forzada Inflamación (local y sistémica) Factores de coagulación de la sangre Inflamación Estructura de los vasos sanguíneos	Pulso Presión arterial
Efectos Agudos	Efectos Agudos
Síntomas respiratorios Agravación bronquitis crónica, asma Uso de medicinas del asma Ausencia trabajo/escuela Muerte Uso de la red sanitaria por anteriores	Trombosis Infarto del miocardio Arritmia Ataque cerebral Muerte Uso de la red sanitaria por anteriores
Efectos crónicos	Efectos crónicos
Reducción de la función pulmonar Bronquitis crónica Cáncer de pulmón Incidencia de asma (en estudio)	Arteriosclerosis (en estudio) Reducción de la esperanza de vida (por muerte cardiovascular prematura)

Reducción de la esperanza de vida	
-----------------------------------	--

Tabla 3 – Efectos de la contaminación en los humanos

3.5 ¿Qué podemos hacer frente a esto?

Frente a la problemática que tenemos en nuestra actualidad, se propone realizar un sistema integral de vigilancia de Calidad del Aire, el cual estará directamente relacionado con la posterior toma de decisiones para la mejora de la calidad del aire en los ayuntamientos.

Este sistema de vigilancia vendrá formado por una serie de estaciones de control distribuidas por todo el territorio. Este sistema se subdivide en tres sistemas interconectados, sistemas de vigilancia, de predicción y de información. Así pues, se encargarán de medir la contaminación en la comunidad y remitirán los datos a una estación en la que se procesan y se almacenan. El objetivo de este sistema integral de vigilancia de la calidad del aire será la de conocer en cada momento los niveles de contaminación atmosférica, permitiendo así su estudio y evolución.

Para la puesta en marcha de este sistema integral se deberá desarrollar un proyecto de creación de un *datamart* específico para toda la información. Este proyecto tiene como objetivo:

- Crear un repositorio único de información el cual va a ser explotado mediante herramientas BI.
- Crear un cuadro de Mando de seguimiento del proyecto a partir de indicadores de rendimiento.
- Ser capaces de obtener conclusiones importantes a raíz de los datos conseguidos mediante el análisis de información.
- Crear una página web en donde podremos ver nuestro cuadro de mando y manipularlo.

3.6 Definición de Indicadores (KPI)

Los KPI (*Key Performance Indicator*) o indicadores de rendimiento son esenciales de cara a saber qué es lo que queremos analizar con nuestro proyecto. La información que nos proporciona nuestro origen de datos es inmensa y si no sabemos qué queremos analizar, no le vamos a sacar rendimiento. Es por ello por lo que entran en juego estos indicadores y nos ayudan a segmentar la información y aprovecharla según unos criterios establecidos. Además, estos indicadores nos ayudarán en la realización del CMI que proponemos.

En el proyecto que nos abarca, seleccionaremos unos KPI que nos permitan segmentar la cantidad de datos que nos ofrece el *open source* de la comunidad de Madrid, y poder mostrar datos de real importancia. Estos indicadores son seleccionados por ser de cumplimiento obligatorio según la normativa europea, la cual exige unos valores máximos que no se podrán sobrepasar. Los indicadores prioritarios que hemos elaborado son los siguientes:

3.6.1 Indicadores de calidad del aire según contaminante

Estos indicadores que vamos a describir a continuación van a poder ser explorados, según la fecha (día, mes o año) o la ubicación en la que está la estación autónoma de recolección de datos, recibiendo así valores únicos o medias aritméticas.

Todos los umbrales que vamos a establecer, tanto los valores límite como los rangos de calidad del aire de los indicadores se pueden consultar en base a:

- El Real Decreto 1073/2002 (BOE A. E., 2008)
- El portal de datos abiertos de la comunidad de Madrid (Madrid, 2018)

Indicador 1 – Medición de PM₁₀

Se trata de una medición de pequeñas partículas que están dispersas por la atmósfera. Estas partículas pueden ir desde polvo, ceniza, hollín, partículas metálicas, cemento o polen y pueden estar en cualquier estado, ya sea sólido o líquido. Además, para esta medición debemos tener en cuenta que su diámetro aerodinámico debe ser menor de 1 micrómetro (10µm). Son denominadas las partículas respirables.

Mostramos una tabla en la que se puede ver los rangos que obtienen estas partículas y su calidad de aire correspondiente:

Índice	Calidad del aire	PM ₁₀ (µg / m ³)
0-50	Buena	0-50
51-100	Admisible	51-90
101-150	Deficiente	91-150
>150	Mala	>150

Tabla 4 - Calidad del aire en base a la medición de PM₁₀

Tabla Resumen del indicador:

Nombre	Medición de PM₁₀
Finalidad	Medir las pequeñas partículas menores de 1 micrómetro dispersas en el aire
Categoría	Operacional
Responsabilidad	Departamento Medioambiental
Definición / Fórmula	No procede cálculo matemático, se mide en (µg / m ³)
F.M. / F. A.	Cada hora en punto
Inductores / Inducidos	Polvo, ceniza, hollín, partículas metálicas, cemento, polen
Quién lo mide	Estación autónoma de control automática
Fuente de los datos	Base de datos al alcance de cualquier ciudadano

Meta / Real / Umbral	Buena 0-50; Admisible 51-90; Deficiente 91-150; Mala > 150
Iniciativas	Activación de protocolo contra la seguridad medioambiental
Observaciones	Estas partículas pueden aparecer en estado sólido o líquido y son denominadas partículas respirables.

Tabla 5 - Resumen Indicador: Medición de PM₁₀

Indicador 2 – Medición de PM_{2.5}

Se trata de la misma medición que la anterior, pero la diferencia existente es que las partículas que viajan por la atmósfera son menores de 0.25 micrómetros (2.5µm) Son denominadas las partículas finas.

Mostramos una tabla en la que se puede ver los rangos que obtienen estas partículas y su calidad de aire correspondiente:

Índice	Calidad del aire	PM _{2.5} (µg / m ³)
0-50	Buena	0-25
51-100	Admisible	26-45
101-150	Deficiente	46-75
>150	Mala	>75

Tabla 6 - Calidad del aire en base a la medición de PM_{2.5}

Tabla Resumen del indicador:

Nombre	Medición de PM_{2.5}
Finalidad	Medir las pequeñas partículas menores de 0.25 micrómetro dispersas en el aire
Categoría	Operacional
Responsabilidad	Departamento Medioambiental
Definición / Fórmula	No procede cálculo matemático, se mide en (µg / m ³)
F.M. / F. A.	Cada hora en punto
Inductores / Inducidos	Polvo, ceniza, hollín, partículas metálicas, cemento, polen
Quién lo mide	Estación autónoma de control automática
Fuente de los datos	Base de datos al alcance de cualquier ciudadano
Meta / Real / Umbral	Buena 0-25; Admisible 26-45; Deficiente 46-75; Mala > 75
Iniciativas	Activación de protocolo contra la seguridad medioambiental
Observaciones	Estas partículas pueden aparecer en estado sólido o líquido y son denominadas partículas finas.

Tabla 7 - Resumen Indicador: Medición de PM_{2.5}

Indicador 3 – Medición de Dióxido de Azufre

El dióxido de azufre es un gas irritante e incoloro el cual tiene un olor penetrante. No es inflamable ni explosivo, pero es peligroso para la atmósfera y las personas. Durante su proceso de oxidación en la atmósfera, este gas forma sulfatos que luego forman parte del material particulado PM₁₀.

Como hemos comentado el azufre es un elemento venenoso para la salud de las personas y los animales. Para las personas un nivel de 300 µg/m³ (300 microgramos por metro cúbico en el aire) ya es un umbral de riesgo serio, mientras que para los árboles un valor de 200 µg/m³ ya es muy grave.

A continuación, se muestra una tabla en la cual se puede ver los rangos que obtienen estas partículas y su calidad de aire correspondiente:

Índice	Calidad del aire	SO ₂ (µg / m ³)
0-50	Buena	0-175
51-100	Admisible	176-350
101-150	Deficiente	351-525
>150	Mala	>525

Tabla 8 - Calidad del aire en base a la medición del Dióxido de Azufre

Tabla Resumen del indicador:

Nombre	Medición de Dióxido de Azufre
Finalidad	Medir la exposición al gas dióxido de azufre
Categoría	Operacional
Responsabilidad	Departamento Medioambiental
Definición / Fórmula	No procede cálculo matemático, se mide en (µg / m ³)
F.M. / F. A.	Cada hora en punto
Inductores / Inducidos	Mediante combustión del azufre y del sulfuro de hidrogeno. Mediante la oxidación producen partículas PM _{2.5}
Quién lo mide	Estación autónoma de control automática
Fuente de los datos	Base de datos al alcance de cualquier ciudadano
Meta / Real / Umbral	Buena 0-175; Admisible 176-350; Deficiente 350-525; Mala > 525
Iniciativas	Activación de protocolo contra la seguridad medioambiental
Observaciones	Gas irritante e incoloro con olor penetrante. No inflamable ni explosivo pero peligroso para las personas y vegetación.

Tabla 9 - Resumen Indicador: Medición de Dióxido de Azufre

Indicador 4 – Medición de Dióxido de Nitrógeno

El dióxido de nitrógeno es un gas irritante, toxico y a la vez precursor de partículas de nitrato. Estas partículas afectan directamente a la producción de ácido y elevados niveles de PM_{2.5}. Este compuesto químico se forma como subproducto en los procesos de combustión a temperaturas

altas. Así pues, este compuesto surge en las grandes ciudades debido a la combustión de vehículos motorizados (coches, motos, camiones...) y también por plantas eléctricas.

Seguidamente, se muestra una tabla en la que se puede ver los rangos que obtienen estas partículas y su calidad de aire correspondiente:

Índice	Calidad del aire	NO ₂ (µg / m ³)
0-50	Buena	0-100
51-100	Admisible	101-200
101-150	Deficiente	201-300
>150	Mala	>300

Tabla 10 - Calidad del aire en base a la medición del Dióxido de Azufre

Tabla Resumen del indicador:

Nombre	Medición de Dióxido de Nitrógeno
Finalidad	Medir la exposición al gas dióxido de nitrógeno
Categoría	Operacional
Responsabilidad	Departamento Medioambiental
Definición / Fórmula	No procede cálculo matemático, se mide en (µg / m ³)
F.M. / F. A.	Cada hora en punto
Inductores / Inducidos	Mediante combustión a temperaturas elevadas en vehículos a motor y también en plantas eléctricas. Estas partículas afectan a la producción de ácido y partículas PM2.5
Quién lo mide	Estación autónoma de control automática
Fuente de los datos	Base de datos al alcance de cualquier ciudadano
Meta / Real / Umbral	Buena 0-100; Admisible 101-200; Deficiente 201-300; Mala > 300
Iniciativas	Activación de protocolo contra la seguridad medioambiental
Observaciones	Gas irritante, tóxico y precursor de partículas de nitrato.

Tabla 11 - Resumen Indicador: Medición de Dióxido de Nitrógeno

Indicador 5 – Medición de Monóxido de Carbono

Se trata de un gas muy peligroso, incoloro y altamente tóxico. Si se respira este gas en niveles elevados puede causar la muerte, por ello se utilizaba en la segunda guerra mundial durante el Holocausto, como método de exterminio (cámara de gas).

Este contaminante se produce por la combustión deficiente de sustancias como el tabaco, madera, petróleo, gas, gasolina, queroseno o carbón. Por este motivo, aparecen en las ciudades, debido a los vehículos a motor y también a la industria.

A continuación, se muestra una tabla en la que se puede ver los rangos que obtienen estas partículas y su calidad de aire correspondiente:

Índice	Calidad del aire	CO (mg / m ³)
--------	------------------	---------------------------

		OCTOHORARIO MOVIL
0-50	Buena	0-5
51-100	Admisible	6-10
101-150	Deficiente	11-15
>150	Mala	>15

Tabla 12 - Calidad del aire en base a la medición del Monóxido de Carbono

Tabla Resumen del indicador:

Nombre	Medición de Monóxido de Carbono
Finalidad	Medir la exposición al gas monóxido de carbono
Categoría	Operacional
Responsabilidad	Departamento Medioambiental
Definición / Fórmula	No procede cálculo matemático, se mide en (mg / m ³)
F.M. / F. A.	Cada hora en punto
Inductores / Inducidos	Mediante la combustión deficiente de tabaco, madera, petróleo, gas, gasolina, queroseno, carbón.
Quién lo mide	Estación autónoma de control automática
Fuente de los datos	Base de datos al alcance de cualquier ciudadano
Meta / Real / Umbral	Buena 0-5; Admisible 6-10; Deficiente 11-15; Mala > 15
Iniciativas	Activación de protocolo contra la seguridad medioambiental
Observaciones	Gas muy peligroso, incoloro y altamente tóxico. Puede producir la muerte en niveles altos.

Tabla 13 - Resumen Indicador: Medición de Monóxido de Carbono

Indicador 6 – Medición de Ozono

El Ozono es una sustancia compuesta por tres átomos de oxígeno, que se crea al disociarse los dos átomos que componen el gas de oxígeno y asociarse a una molécula de oxígeno. Este gas solamente es peligroso si se respira en grandes cantidades y puede provocar irritación de ojos y garganta.

Tiene un componente beneficioso para la humanidad que evita que los rayos ultravioletas formados por el sol penetren en la atmósfera en grandes cantidades. Sin esta capa de ozono, situada en la estratosfera, la vida humana no sería posible, por eso la importancia de preservarla.

Mostramos una tabla en la cual se puede ver los rangos que obtienen estas partículas y su calidad de aire correspondiente:

Índice	Calidad del aire	O ₃ (µg / m ³) OCTOHORARIO MOVIL
0-50	Buena	0-90
51-100	Admisible	91-180
101-150	Deficiente	181-240

>150	Mala	>240
------	------	------

Tabla 14 - Calidad del aire en base a la medición del Monóxido de Carbono

Tabla Resumen del indicador:

Nombre	Medición de Ozono
Finalidad	Medir la exposición al gas Ozono
Categoría	Operacional
Responsabilidad	Departamento Medioambiental
Definición / Fórmula	No procede cálculo matemático, se mide en ($\mu\text{g} / \text{m}^3$)
F.M. / F. A.	Cada hora en punto
Inductores / Inducidos	Mediante la disociación de los dos átomos que componen el oxígeno y unirse a una molécula de oxígeno
Quién lo mide	Estación autónoma de control automática
Fuente de los datos	Base de datos al alcance de cualquier ciudadano
Meta / Real / Umbral	Buena 0-90; Admisible 91-180; Deficiente 181-240; Mala > 240
Iniciativas	Activación de protocolo contra la seguridad medioambiental
Observaciones	Gas que provoca irritación de ojos y garganta y solo es peligroso si se respira en grandes cantidades.

Tabla 15 - Resumen Indicador: Medición de Ozono

Indicador 7 – Medición de Benceno

El Benceno es un hidrocarburo aromático (C_6H_6), que tiene un especial uso en la fabricación de plásticos, resinas, nilón, fibras sintéticas, medicamentos, detergentes, pesticidas.... Este componente se encuentra de forma natural en los petróleos y la gasolina, además del cigarrillo, por lo que es frecuente que aparezca en ciudades.

Se trata de un elemento tóxico, el cual ingerido a cantidades bajas puede provocar somnolencia, mareos o taquicardias, y en cantidades muy altas la muerte. Además, está comprobado que aumenta el riesgo de leucemia en el hombre adulto.

A continuación, se muestra una tabla en la que se puede ver el umbral límite que tiene esta partícula en concreto:

Calidad del aire	BEN ($\mu\text{g} / \text{m}^3$)
Valor límite para la protección de la salud	<5
Excede valor límite	>=5

Tabla 16 - Calidad del aire en base a la medición del Benceno

Tabla Resumen del indicador:

Nombre	Medición de Benceno
Finalidad	Medir la exposición al gas benceno
Categoría	Operacional
Responsabilidad	Departamento Medioambiental
Definición / Fórmula	No procede cálculo matemático, se mide en ($\mu\text{g} / \text{m}^3$)
F.M. / F. A.	Cada hora en punto
Inductores / Inducidos	Presente de forma natural en petróleos, gasolina, cigarrillos y se usa en la fabricación de plásticos, resinas, nilón, fibras sintéticas, medicamentos, detergentes, pesticidas.
Quién lo mide	Estación autónoma de control automática
Fuente de los datos	Base de datos al alcance de cualquier ciudadano
Meta / Real / Umbral	Buena < 5; Mala > 5
Iniciativas	Activación de protocolo contra la seguridad medioambiental
Observaciones	Hidrocarburo que ingerido en cantidades bajas produce somnolencia, mareos y taquicardias, e ingerido en cantidades altas produce la muerte. Está comprobado que aumenta el riesgo de leucemia en el hombre adulto.

Tabla 17 - Resumen Indicador: Medición de Benceno

3.6.2 Indicadores de calidad del aire según localización de estación autónoma

Indicador 8 – Geolocalización de estaciones autónomas

Se establece un indicador para la rápida visualización mediante un mapa territorial, en el cual se puede ver de un único vistazo la calidad del aire de todas las zonas mediante los colores correspondientes. Los colores establecidos según la calidad serán los siguientes:

Calidad del aire
Buena
Admisible
Deficiente
Mala

Tabla 18 - Calidad del aire mediante colores

Tabla Resumen del indicador:

Nombre	Geolocalización de estaciones autónomas
Finalidad	Geolocalizar las estaciones autónomas en el mapa territorial
Categoría	Estratégico
Responsabilidad	Departamento Medioambiental
Proceso Relacionado	Mediciones de gases
Quién lo mide	Estación autónoma de control automática
Fuente de los datos	Base de datos al alcance de cualquier ciudadano

Meta / Real / Umbral	Buena; Admisible; Deficiente; Mala
Iniciativas	Verificar de forma rápida el estado de la estación
Observaciones	Se establecerán colores indicando el estado

Tabla 19 - Resumen Indicador: Geolocalización de estaciones autónomas automáticas

3.6.3 Indicadores de calidad del aire según representaciones graficas

Indicador 9 – Realización de gráficos

Mediante la inclusión o exclusión de datos, se realizan diferentes grafismos para mostrar. Se podrán realizar gráficos mensuales por estaciones y contaminantes, así como acotando por fecha seleccionada.

Tabla Resumen del indicador:

Nombre	Realización de gráficos
Finalidad	Establecer gráficos comparativos entre estaciones y contaminantes
Categoría	Estratégico
Responsabilidad	Departamento Medioambiental
Proceso Relacionado	Mediciones de gases
Quién lo mide	Estación autónoma de control automática
Fuente de los datos	Base de datos al alcance de cualquier ciudadano
Meta / Real / Umbral	Buena; Admisible; Deficiente; Mala
Iniciativas	Verificar de forma rápida la tendencia o progresión de las mediciones en el tiempo

Tabla 20 - Resumen Indicador: Realización de gráficos

4. Diseño del proyecto

Después del análisis realizado en el anterior apartado, se debe diseñar el proyecto en base al mismo. Para ello, se debe presentar las distintas fases por las que pasará para el posterior desarrollo e implementación.

Se debe partir desde una arquitectura de proyecto preestablecida, la cual facilitará la posterior implementación del proyecto. Se partirá de un origen de datos, el cual tiene muchos datos de interés, pero muchos otros que seguramente no serán válidos, es por ello que se debe identificar la información válida de la que no lo es.

Este origen de datos en concreto será de la comunidad de Madrid, ya que nos pone a disposición una gran cantidad de datos para su manipulación, por lo que establecemos este origen de datos como punto de partida. Además, este *datasource* será escalable, pudiendo ampliarse y agregar otras comunidades. Una vez conocido el origen de datos, se realiza una serie de procesos de ETL, los cuales permiten desgranar la información relevante del origen de datos y alimentar *datawarehouse*. Estos procesos ETL estarán basados en modificación de campos, unión de ellos, cambios de datos por otros, estructuración de tipos, formateo de fechas y vinculación de siglas con su correspondiente significado.

Posteriormente, se va a establecer una serie de tablas principales, que serán las que tienen los datos que se plasmarán en CMI. Además, también existirán unas tablas auxiliares que darán más información acerca de lo que se expone. Esta base de datos seguirá un modelo de entidad relación.

Por último, queda la realización de CMI, el cual va a mostrar datos mensuales, diarios y en tiempo real, por lo que se han elaborado tres tablas principales con los datos que se necesitan en ellas. Así pues, si se quiere consultar un contaminante en una zona en concreto y en tiempo real se dispondrá del valor de ese contaminante. Si además de eso, se quiere más información sobre el nombre del contaminante o donde se encuentra ubicada la estación de control automática, se procederá a sacar los datos de las tablas auxiliares.

4.1 Orígenes de datos

El *DataSource*, como su nombre bien indica, es el origen de datos, es decir, de donde se van a obtener los datos para poder explotarlos como más convenga.

En este caso en particular, se va a explotar los datos del ayuntamiento de Madrid. Estos datos se pueden encontrar en su repositorio de origen abierto para todo ciudadano. En este repositorio se encuentran todos los datos que recogen las estaciones de medición autónomas de la comunidad de Madrid, separados en archivos mensuales.

Estos datos de origen se encuentran en formato .csv, un formato conocido de representación de datos en forma de tabla y se encuentran separados por años y por meses. Como todos los datos están disgregados hay muchos archivos de origen y se debe atacar a todos mediante procesos ETL.

4.2 Investigación y selección de herramientas BI

En primer lugar, se ha de tener en cuenta lo que dice el portal Gartner, que cada año saca su cuadrante mágico con las mejores soluciones para BI. Como se observa a continuación, Microsoft está posicionada como la mejor solución actual para el análisis de datos.



Ilustración 5 - Cuadrante mágico de Gartner 2019

Como se ha podido ver en el análisis de herramientas, Microsoft está posicionado en cada uno de los procesos necesarios para el análisis de datos, está presente tanto en los procesos ETL, como en origen de datos y en la elaboración de informes. Es por ello que vamos a comparar sus herramientas con las de otros fabricantes para ver cual se adapta mejor a nuestras necesidades.

4.2.1 Procesos ETL

ETL es un acrónimo de origen inglés que significa *Extract, Transform and Load*. Se trata de un proceso que permite extraer datos de múltiples orígenes, transformarlos según las necesidades y después cargarlos en *Data Warehouse, Data Marts* o en otro sistema operacional.

Existen multitud de programas capaces de realizar procesos ETL de una forma sencilla e interactiva, aunque también los hay más complejos y completos. En esta situación, se va a analizar algunos y escoger el que mejor se adapte a la situación y requerimientos.

Keetle:

Se trata de una herramienta dentro del paquete Pentaho Data Integration (PDI). Se caracteriza por ser un software libre y portable, por lo que no requiere instalación ni licencia. Además, dispone de un entorno de diseño intuitivo, gráfico y fácil de usar que proporciona que la integración de datos sea cada vez más fácil para las organizaciones.

Destaca por las siguientes características:

- Supervisión mejorada del rendimiento del sistema
- Mejoras en el perfilado de datos
- Añade fácilmente nuevos *plugins*
- Entregar datos desde múltiples fuentes de datos
- Balanceo de carga de movimiento de datos
- Revertir cambios en transacciones de base de datos de trabajo

Esta tanto disponible para las plataformas de Windows, Linux y MacOs

Talend:

Se trata de una herramienta de *open-source*, con licencia GLP, que permite modelar transformaciones de datos de una forma visual y generando código java para ello.

Este software no se dedica solo a procesos ETL sino también para sincronización y replicación de bases de datos, migraciones y transformaciones de datos.

La constitución de los procesos ETL se realiza de una forma sencilla, ya que permite crearlos a base de componentes predefinidos (dispone de más de 400) o personalizarlos. El diseñador de procesos proporciona una vista gráfica con los componentes arrastrables, por lo que resulta cómodo y fácil realizar procesos sencillos. También dispone de una herramienta de *debug*, con la cual se puede seguir en tiempo real el proceso de transformación de datos.

Power Query:

Se trata de una tecnología de conexión de datos que permite tanto detectar, conectar, combinar y transformar orígenes de datos según necesidades. Estas características están incrustadas en aplicaciones de uso diario como puede ser Microsoft Excel o de uso empresarial como puede ser Power BI Desktop.

Se trata de una herramienta desarrollada específicamente para abordar la manipulación y la transformación de datos. Se caracteriza por tener una curva de aprendizaje corta en comparación con otras herramientas ETL. Además, se trata de una solución escalable, con lo que se puede emplear en un entorno local pero después puede ser desplegada a una solución más sólida y extensa como un entorno web.

4.2.2 DataWarehouse

El *DataWarehouse* es un almacén de datos electrónico, donde generalmente, una empresa u organización, guardan una gran cantidad de información y datos relevante. Estos datos se guardan de una forma segura, fiable y deben ser de fácil acceso, recuperación y administración.

Generalmente, el *DataWarehouse* se aloja en un servidor corporativo o, debido al auge actual, en la nube. Estos datos pueden proceder de diferentes aplicaciones *CRM*, *Excel*, *ERP*, *Portal Web*, *Archivos de texto*, etc. A estos datos en bruto, se les aplican procesos *ETL*, con lo que se transforman datos en información y esta información se almacena en *DataWarehouse* para posteriormente, y mediante software de procesamiento de *Reports*, técnicas *OLAP* o *DataMining*, transformarla en conocimiento.

En la actualidad existen multitud de programas que podrían actuar como repositorio de datos, por lo que se va a seleccionar algunos de los más conocidos y ver qué es lo que nos aportaría y si se adecúan al proyecto.

Postgresql:

Se trata de una opción interesante debido a que es una base de datos relacional *open-source* y de fácil acceso. Nació a mediados de los 80 bajo el nombre *Post Ingres* y de la mano de Michael Stonebraker (Stonebraker, 1943) con la idea de poder solucionar los distintos tipos de problemas que surgieron en torno a las bases de datos.

Como se ha comentado, se trata de una herramienta gratuita y libre, por lo que tiene una gran comunidad detrás y grandes características y opciones avanzadas que hacen que sea uno de los grandes protagonistas en cuanto a bases de datos se refiere. También aporta mucha flexibilidad a los proyectos ya que permite definir funciones personalizadas por medio de muchos lenguajes de programación (*pgsql*, *Perl*, *Python*, *PHP*, *Ruby*, *Java* o *Tcl*). Además, hay que destacar que se trata de un software multiplataforma y puede ser instalado tanto en *MacOS*, *Windows*, *Solaris*, *Red Hat*, *Debian* o *Ubuntu*.

MariaDB:

Derivado de *MySQL* y con licencia *GPL*, se trata de un sistema de gestión de bases de datos desarrollado por Michael Widenius (Widenius, 1962), la fundación *MariaDB* y la gran comunidad existente de desarrolladores.

Este sistema gestor se caracteriza por tener una alta compatibilidad con *MySQL* ya que tiene las mismas *APIs*, bibliotecas, ordenes e interfaces. Con esta compatibilidad está claro cuál es su principal objetivo, poder intercambiar un servidor por otro sin esfuerzo alguno.

MariaDB integra una serie de novedades y mejoras con respecto a *MySQL*, entre las que destacan las siguientes:

- Nuevos motores de búsqueda como *Aria* y *XtraDB*.
- Nuevas características directamente relacionadas con bases de datos *NoSQL*.

- Nueva gestión de conexiones con bases de datos
- Nuevos y mejores motores de funcionamiento en *cluster*, que mejora las posibilidades de cara a la migración al Cloud.

Al igual que Postgresql se trata de un software multiplataforma y es compatible con Windows, Linux, RedHat/CentOS/Fedora y Debian/Ubuntu.

SQL Server 2017 Express Edition:

Se trata del sistema gestor de bases de datos de Microsoft. Es ideal para desarrollar y producir aplicaciones de escritorio, *webapps* o pequeñas aplicaciones de servidor. En este caso es la versión gratuita y de uso libre, que ofrece unas características más que suficientes para el actual proyecto. Además, ofrece características de analítica avanzada mediante scripts propios desarrollados en Python y R, que darán unas visualizaciones interactivas de calidad y en cualquier dispositivo.

Una de sus principales características es que se puede ejecutar mediante una máquina virtual en Azure, por lo que se podrá sincronizarla con todos los programas de Microsoft. Por otro lado, tiene una alta compatibilidad de sistemas, y puede ser instalado tanto en Windows como en Linux y en contenedores Docker, el cual proporciona *sandbox* para realizar las pruebas sin riesgo.

4.2.3 Creación de CMI BI (Reporting)

La creación de los informes o reportes para *Business Intelligence* se realiza mediante software adaptado para ello. Estos softwares se pueden utilizar para tareas simples, como pueden ser creación de informes en base a un origen de datos, o la creación de *Dashboards* o cuadros de mando integrales interactivos.

En el actual proyecto, se va a tratar de elaborar unos CMI con los que se visualice de una forma rápida el historial de contaminación de todas las estaciones de control autónomas de cualquier comunidad. Para ello, y antes de seleccionar un software que permita la realización del CMI, se va a realizar un análisis de distintos softwares de cara a la elección del más propicio.

Microstrategy:

Es uno de los proveedores más antiguos en el área de *Business Intelligence*, más concretamente data de 1989. Fue creada originalmente para el análisis de datos relaciones, pero actualmente ya permite crear cuadros de mando interactivos con los que atender tendencias y previsiones. Tienen soporte para Hadoop y otras BBDD y motores específicos de *Big Data* e información no estructurada.

Entre sus características más destacadas se encuentran:

- *Scoreboards* y *Dashboards* personalizados
- Análisis OLAP y ROLAP
- Alertas y notificaciones proactivas

- Emplean tecnología *flash* para los *Dashboards* que presentan unos cuadros limpios y atractivos.

Pentaho:

Se trata de una herramienta de análisis para *back-end* que se caracteriza por ser una herramienta de código abierto y gran soporte por la comunidad. Proporciona gran flexibilidad y actualmente está a manos de Hitachi Vantara.

Entre sus ventajas destacan:

- Facilidad para la implementación de procesos ETL, mediante procesos gráficos.
- Dispone de una librería rica en componentes, para ayudar al usuario
- No es necesario teclear código para el análisis de datos.

PowerBI:

Se trata de una herramienta de Microsoft, la cual incluye un servicio en la nube de fácil implementación, que permite subir, compartir y tener un acceso a los informes desde cualquier dispositivo con conexión a internet. Al ser una herramienta de Microsoft, tiene la misma estética y funcionalidades que las ya conocidas de la marca, como Office365. Además, debido a la propia integración con Office365 permite el acceso a orígenes de datos tanto locales, como basados en la nube e incluso orígenes de bases de datos.

Dispone de varias ventajas a tener en cuenta:

- Se trata de una herramienta eficiente en cuanto a la manera autónoma de extraer informes cuando se requiera sin tener conocimiento específico de la herramienta.
- Es capaz de producir gráficos y gráficos personalizados debido a Quick Insights que determina las correlaciones y patrones de sus datos.
- Existe integración con el programa de análisis estadístico R, a través de scripts y objetos visuales. También tiene integración con Microsoft Azure Machine learning y Azure Stream Analytics.

4.2.4 Características que debe tener nuestro software

La elección que se realiza del software para la implementación del sistema de inteligencia en el negocio es esencial y debe ser la correcta. Equivocarse en esta elección puede suponer el fracaso del proyecto y la consecutiva pérdida de los fondos destinados al mismo. Para evitar esto, se va a establecer una serie de características que debe tener el software para así poder elegirlo con una base sólida.

Al tratarse de un proyecto de fin de grado, puesto que esta implementación no es real, algunas de las características del software estarán limitadas a las necesidades del estudiante, como pueden ser la no obligatoriedad de pagar por software.

El software, de cara a ser elegido, deberá de cumplir con los siguientes requisitos y/o valoraciones:

- Se valorará que el software sea de carácter gratuito
- Se valorará positivamente que el software sea open source
- Es necesario que aporte la capacidad de exportación a un entorno web
- Es necesario que el software sea capaz de gestionar el datawarehouse, los procesos ETL y la creación de informes por sí mismo
- Se buscará una interfaz de usuario amigable, limpia, reconocible y con una curva de aprendizaje suave
- Que el software sea escalable
- Capacidad para desarrollar elementos complejos mediante scripts de código
- Debido a la necesidad de geolocalizar las estaciones de control, el software debe ser capaz de ubicarlas en mapa

4.2.5 Decisión y elección de software final

Una vez que se ha realizado un análisis sobre las distintas herramientas para afrontar los diferentes procesos del *Business Intelligence*, se ha de elegir que herramientas se adecúen a las necesidades, las cuales aporten valor para la realización del proyecto. Además, también deben cumplir con las características y requisitos anteriormente citados.

Si se revisan todas las características que se han interpuesto para la elección del software, se observa cómo PowerBi de Microsoft se sitúa en una de las más destacadas, cumpliendo casi todas las características.

Se trata de una herramienta que está disponible gratuitamente para cualquier usuario. Es cierto que existe una versión de pago con más funciones, pero la versión gratuita es suficientemente potente como para el desarrollo del actual proyecto. Además, la UOC ofrece el paquete de Microsoft por ser estudiante, por lo que, con este paquete se puede iniciar sesión en el software y permitiendo así la exportación del trabajo a su nube personal. Con ello se gana acceso desde cualquier parte y cualquier dispositivo con conexión a internet.

También se destaca la interfaz que tiene los programas de Microsoft, la cual es utilizada en la mayoría de sus programas, como el paquete office. Es por esto por lo que la curva de aprendizaje del software se limita solo al uso de herramientas analíticas puramente, puesto que la interfaz es idéntica al resto del paquete Microsoft.

Se ha dado especial importancia a la geolocalización, ya que es esencial en el actual proyecto que se pueda visualizar dónde está cada estación de control de calidad del aire. PowerBi ofrece esta solución con dos sistemas diferentes, con geolocalización mediante sus mapas de Bing o con ArcGIS Maps for PowerBi.

Por último, con este programa de Microsoft, se asegura la total compatibilidad con los *DataSource* que actualmente se utiliza (se recuerda que eran archivos csv). También se asegura la realización de los procesos ETL, que permite el mismo PowerBI. Y, por último, es una de las herramientas más potentes para la realización de *Dashboards* interactivos, que es el objetivo final.

Es por ello por lo que la elección final será la de usar el software *PowerBI*.

4.2 Arquitectura

Como ya se ha adelantado en el apartado anterior, la arquitectura del proyecto será la base para empezar el diseño de este. Se debe conocer todos los estados por los que pasará el proyecto para poder diseñarlos y posteriormente implementarlos.

La arquitectura del proyecto se basa en un proyecto de inteligencia de negocio básico, por lo que contendrá los elementos básicos de cualquier proyecto. A continuación, se muestran en forma de diagrama cada estado del proyecto:

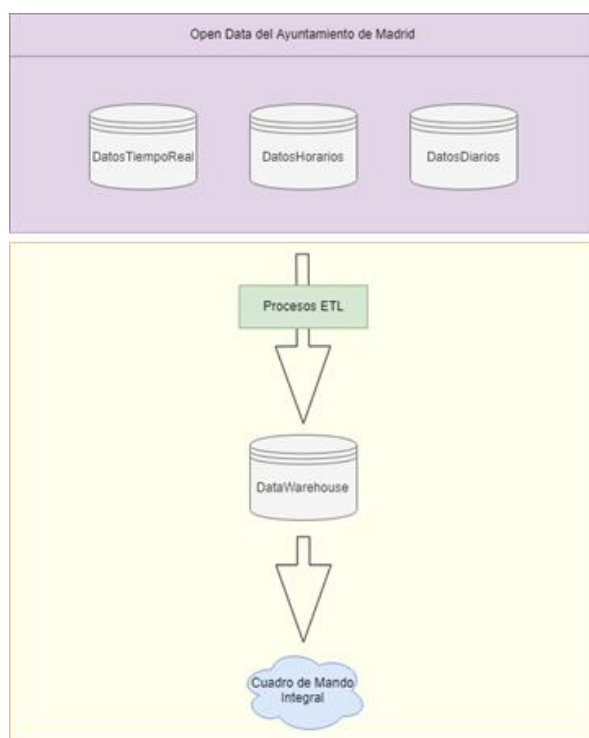


Ilustración 6 - Arquitectura del proyecto

Como se ha podido observar se pasa por varios estados. Primero se parte de la necesidad de un *datasource*, que será el que contendrá todos los datos a explotar. Este *datasource* será el de la Comunidad de Madrid, el cual tiene datos de sus estaciones de medición al alcance de cualquier ciudadano.

Se analizarán esos datos y se filtrarán mediante una serie de procesos ETL que se deben desarrollar. Una vez finalizado los procesos ETL, esta nueva información será trasladada a *datawarehouse* el cual está listo para el análisis y la explotación mediante un cuadro de mando integral.

4.3 DataSource

El origen de datos será el lugar del cual se obtendrán todos los datos. Estos datos se tendrán que procesar para obtener información. Así pues, aquí solo se encontrarán datos en bruto que vendrán del sistema integral de vigilancia de la calidad del aire.

Este origen de datos será el open data de la comunidad de Madrid, por lo que los datos de las estaciones autónomas serán los propios de la comunidad de Madrid.

4.4 Extracción, transformación y carga de datos (ETL)

Una vez que se tiene el origen de datos presente y se han analizado los datos que incluye, se debe elaborar una serie de procesos de extracción, transformación y carga de datos. Se pasará por tres estados en estos procesos.

Extracción: Mediante el software que se utilizará, se extraerán todos los datos procedentes del origen de datos.

Transformación: Una vez que se tienen todos los datos en el software, se debe proceder a aplicar unos filtros, cambios y modificaciones para obtener los datos que realmente se quieren y se van a utilizar para la explotación. En este proceso se transforma los datos en información.

Carga: Una vez que se obtienen los datos filtrados y en el formato que se desee, se deben cargarlos a la base de datos para después realizar el análisis y explotación de los mismos.

A continuación, se presenta un gráfico de cómo se hará este proceso y en posteriores apartados se documentará complementemente:

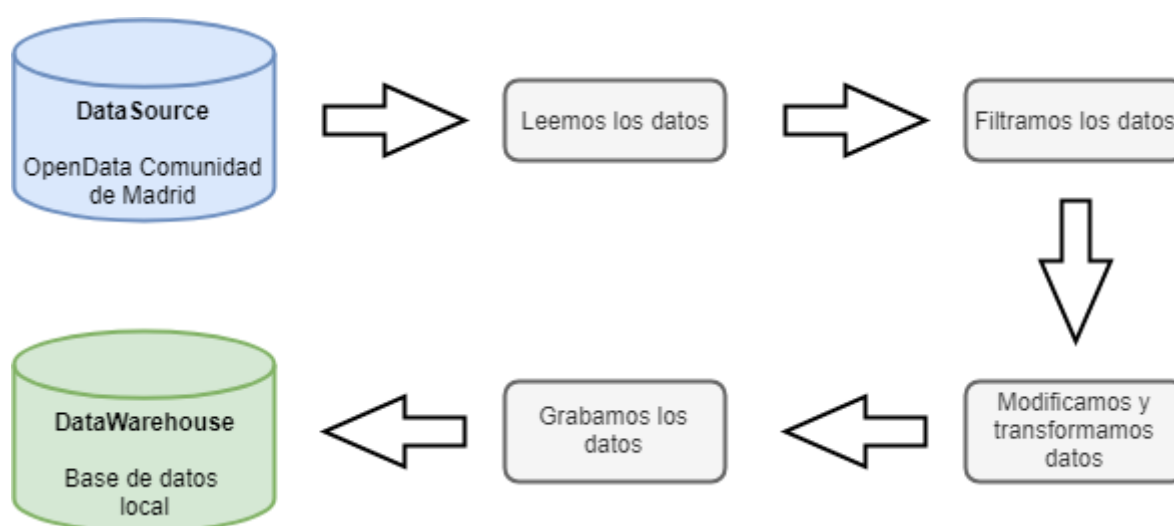


Ilustración 7 - Proceso ETL datos en tiempo real

4.5 Base de datos

En lo que a las tablas de la base de datos se refiere, se va a intentar mostrar los mínimos datos para su correcta visualización, por lo que en los procesos ETL pertinentes se deberá seleccionar sólo lo que interese.

A continuación, se muestra cómo sería el modelado de tablas, que se ha realizado con un diseñador de bases de datos online, *Vertabelo*:

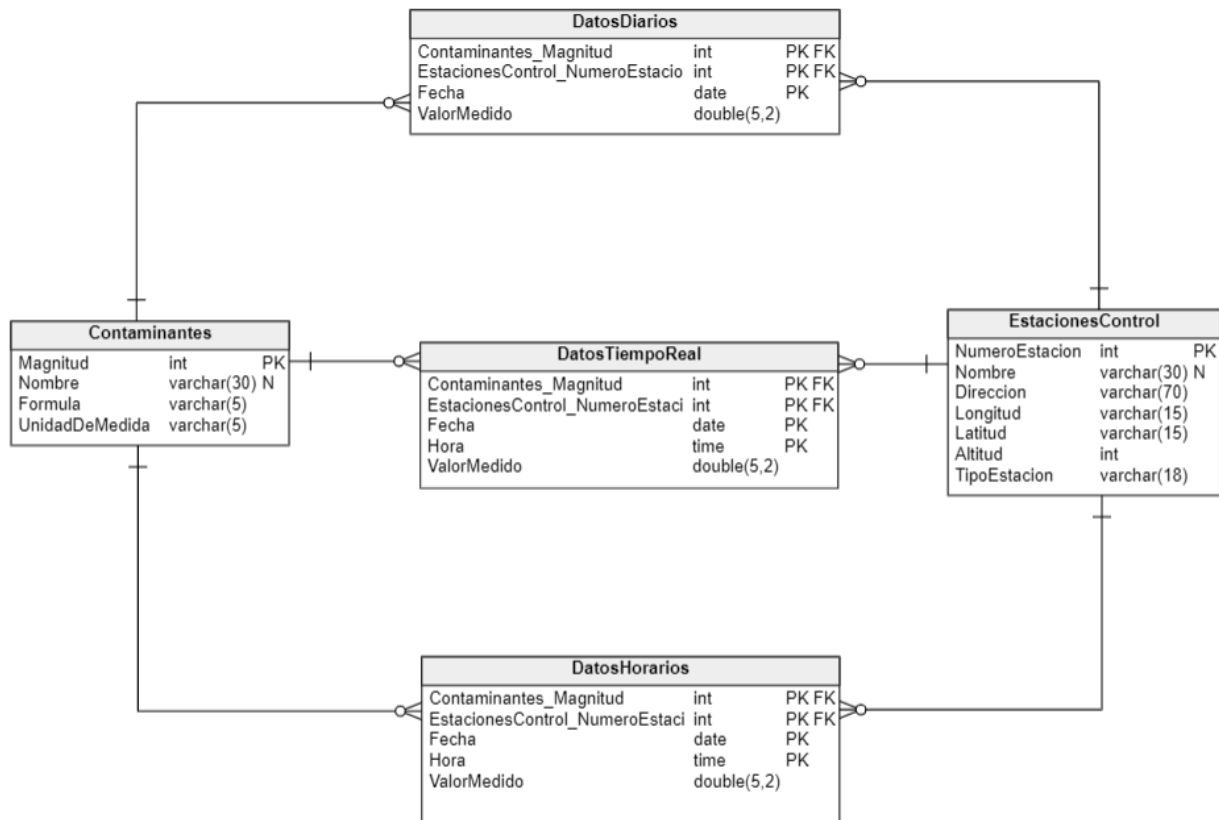


Ilustración 8 - Modelo lógico

A continuación, se pasa a explicar qué es lo que contiene cada atributo de cada tabla y para qué se usa.

4.5.1 Tabla Contaminantes

Contaminantes		
Magnitud	int	PK
Nombre	varchar(30) N	
Formula	varchar(5)	
UnidadDeMedida	varchar(5)	

Ilustración 9 - Modelo lógico: Tabla contaminantes

En esta tabla se guardan todos los datos relacionados con los contaminantes que se miden en la actualidad. La información de cada campo es la siguiente:

Magnitud: Será un número *Integer*, que representará el contaminante medido. Este campo será la clave primaria de la tabla.

Nombre: Será un valor *Char* de máximo treinta caracteres el cual representa el nombre asignado a la magnitud del contaminante. Este valor no puede ser nulo.

Fórmula: Será un valor *Char* de máximo cinco caracteres y es la abreviatura del contaminante. Por ejemplo: Monóxido de Carbono (CO).

UnidadDeMedida: Será un valor *Char* de máximo cinco caracteres y se trata de la unidad en la que se miden los contaminantes. Por ejemplo ($\mu\text{g} / \text{m}^3$).

4.5.2 Tabla EstacionesControl

EstacionesControl		
NumeroEstacion	int	PK
Nombre	varchar(30)	N
Direccion	varchar(70)	
Longitud	varchar(15)	
Latitud	varchar(15)	
Altitud	int	
TipoEstacion	varchar(18)	

Ilustración 10 - Modelo lógico: Tabla de estaciones de control

En esta tabla se guardarán los datos relacionados con las estaciones de control automáticas que miden los contaminantes. La información de cada campo es la siguiente:

NumeroEstacion: Se trata de un *Integer* que representa el número identificativo de la estación de control. Este campo será la clave primaria de la tabla.

Nombre: Será un valor *Char* de máximo treinta caracteres el cual representa el nombre asignado a la estación autónoma de control. Este valor no puede ser nulo.

Dirección: Será un valor *Char* de máximo setenta caracteres el cual representa la dirección donde se encuentra la estación autónoma de control.

Longitud: Será un valor *Char* de máximo quince caracteres el cual las coordenadas geográficas de la estación de control, en este caso representa la longitud.

Latitud: Será un valor *Char* de máximo quince caracteres el cual las coordenadas geográficas de la estación de control, en este caso representa la latitud.

Altitud: Será un valor *Integer* que representa la altitud a la que está situado la estación de control, en este caso representa la latitud.

TipoEstacion: Será un valor *Char* de máximo dieciocho caracteres que dice el tipo de estación que de la que se trata. Los tipos solo podrán ser: Suburbana, Urbana de tráfico, Urbana de fondo.

4.5.3 Tabla DatosDiarios

DatosDiarios		
Contaminantes_Magnitud	int	PK FK
EstacionesControl_NumeroEstacio	int	PK FK
Fecha	date	PK
ValorMedido	double(5,2)	

Ilustración 11 - Modelo lógico: Tabla de datos diarios

En esta tabla se guardarán las mediciones que realizan diariamente las estaciones de control. La información de cada campo es la siguiente:

Contaminantes_Magnitud: Se trata del mismo campo Magnitud de la tabla contaminantes. Es una clave foránea.

EstacionesControl_NumeroEstacion: Se trata del mismo campo NúmeroEstación de la tabla EstacionesControl. Es una clave foránea.

Fecha: Se trata de un campo *Date* que contiene la fecha, en formato dd/mm/aaaa, en la que se ha realizado la medición.

Los campos **Contaminantes_Magnitud**, **EstacionesControl_NumeroEstacion** y **Fecha** serán la clave primaria de esta tabla.

ValorMedido: Se trata de un campo *Double* de cinco números y dos decimales como máximo. Este campo será el valor de la medición realizada por la estación de control.

4.5.4 Tabla DatosTiempoReal

DatosTiempoReal		
Contaminantes_Magnitud	int	PK FK
EstacionesControl_NumeroEstaci	int	PK FK
Fecha	date	PK
Hora	time	PK
ValorMedido	double(5,2)	

Ilustración 12 - Modelo lógico: Tabla de datos en tiempo real

En esta tabla se guardarán las mediciones en tiempo real que realizan las estaciones de control. La información de cada campo es la siguiente:

Contaminantes_Magnitud: Se trata del mismo campo Magnitud de la tabla contaminantes. Es una clave foránea.

Contaminantes_Magnitud: Se trata del mismo campo NúmeroEstación de la tabla EstacionesControl. Es una clave foránea.

Fecha: Se trata de un campo *Date* que contiene la fecha, en formato dd/mm/aaaa, en la que se ha realizado la medición.

Hora: Se trata de un campo *Time* que contiene la hora exacta, en formato hh:mm, en la que se ha realizado la medición.

Los campos **Contaminantes_Magnitud**, **EstacionesControl_NumeroEstaci**, **Fecha** y **Hora** serán la clave primaria de esta tabla.

ValorMedido: Se trata de un campo *Double* de cinco números y dos decimales como máximo. Este campo será el valor de la medición realizada por la estación de control.

4.5.5 Tabla DatosHorarios

DatosHorarios		
Contaminantes_Magnitud	int	PK FK
EstacionesControl_NumeroEstaci	int	PK FK
Fecha	date	PK
Hora	time	PK
ValorMedido	double(5,2)	

Ilustración 13 - Modelo lógico: Tabla de datos horarios

En esta tabla se guardarán las mediciones que realizan cada hora las estaciones de control. La información de cada campo es la siguiente:

Contaminantes_Magnitud: Se trata del mismo campo Magnitud de la tabla contaminantes. Es una clave foránea.

EstacionesControl_NumeroEstaci: Se trata del mismo campo NúmeroEstación de la tabla EstacionesControl. Es una clave foránea.

Fecha: Se trata de un campo *Date* que contiene la fecha, en formato dd/mm/aaaa, en la que se ha realizado la medición.

Hora: Se trata de un campo *Time* que contiene la hora exacta, en formato hh:mm, en la que se ha realizado la medición.

Los campos **EstacionesControl_NumeroEstaci**, **Fecha** y **Hora** serán la clave primaria de esta tabla.

ValorMedido: Se trata de un campo *Double* de cinco números y dos decimales como máximo. Este campo será el valor de la medición realizada por la estación de control.

4.6 Cuadro de Mando Integral

El cuadro de mando será donde se visualizarán todos los datos que se quiera y que saldrán de las consultas a las tablas de la base de datos. Se va a realizar un pequeño diseño en el cual se va a poder ver cómo será CMI y qué es lo que se puede visualizar.

Para la representación del CMI se va a utilizar la herramienta Balsamiq Mockup, que permite realizar modelados de posibles CMI. Se ha realizado un *dashboard* interactivo con cuatro pestañas que se va a describir a continuación.

4.6.1 Vista rápida de la calidad del aire en tiempo real

En este apartado se puede ver de una manera rápida y sencilla los datos en tiempo real de las estaciones de control, seleccionando un contaminante se podrá ver los datos medidos en la jornada del día actual (dependerá de que el origen de datos esté actualizado).

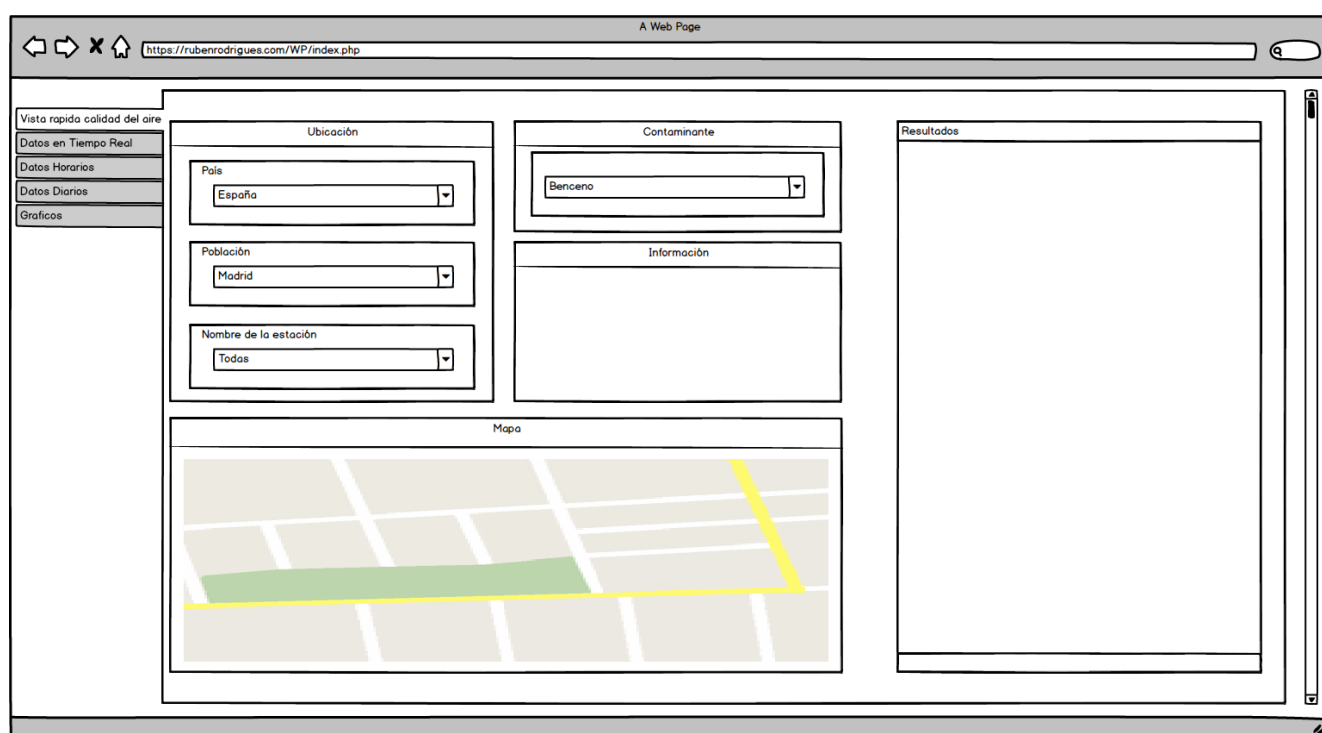


Ilustración 14 - CMI: Vista rápida de la calidad del aire en tiempo real

4.6.1 Datos en tiempo real

En este apartado se puede ver los datos en tiempo real de las estaciones de control, seleccionando uno o varios contaminantes se podrá ver qué datos están midiendo en el momento preciso (dependerá de que el origen de datos esté actualizado)

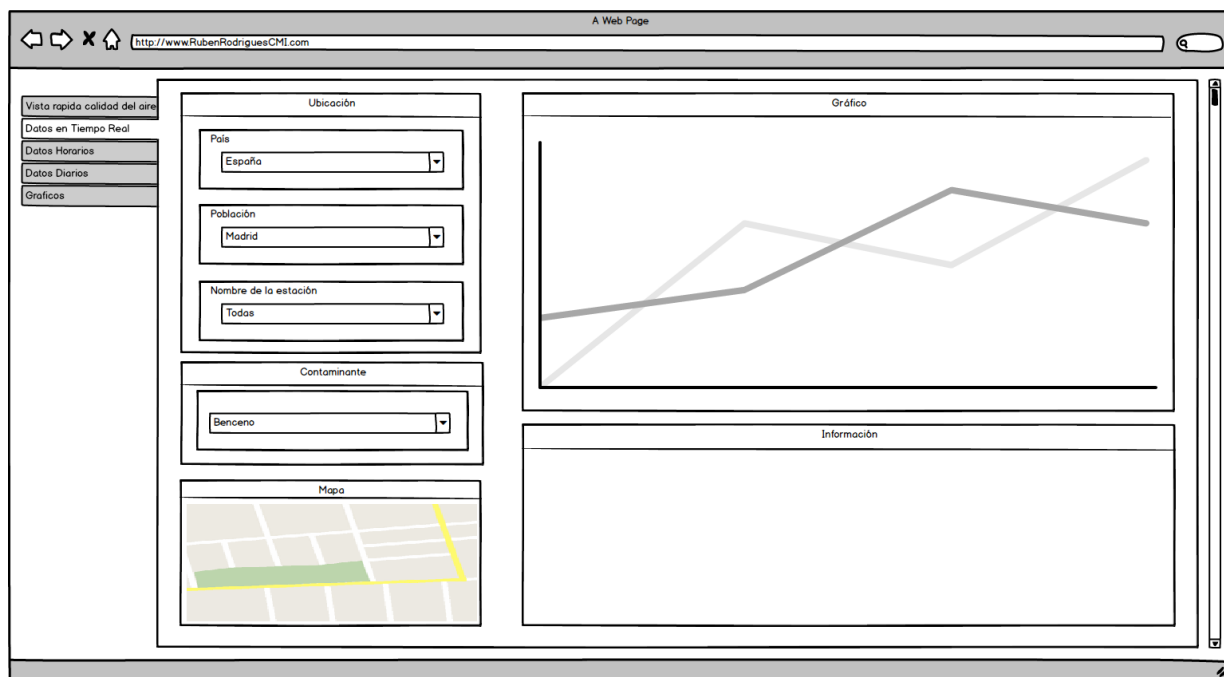


Ilustración 15 - CMI: Datos en tiempo real

4.6.2 Datos Horarios

En este apartado se podrá visualizar los datos según las horas del día que se seleccionen. Para ello, se debe seleccionar un o varios contaminante y una o varias estaciones de control y se verá la ubicación en el mapa y las mediciones realizadas a las horas en punto de ese día.

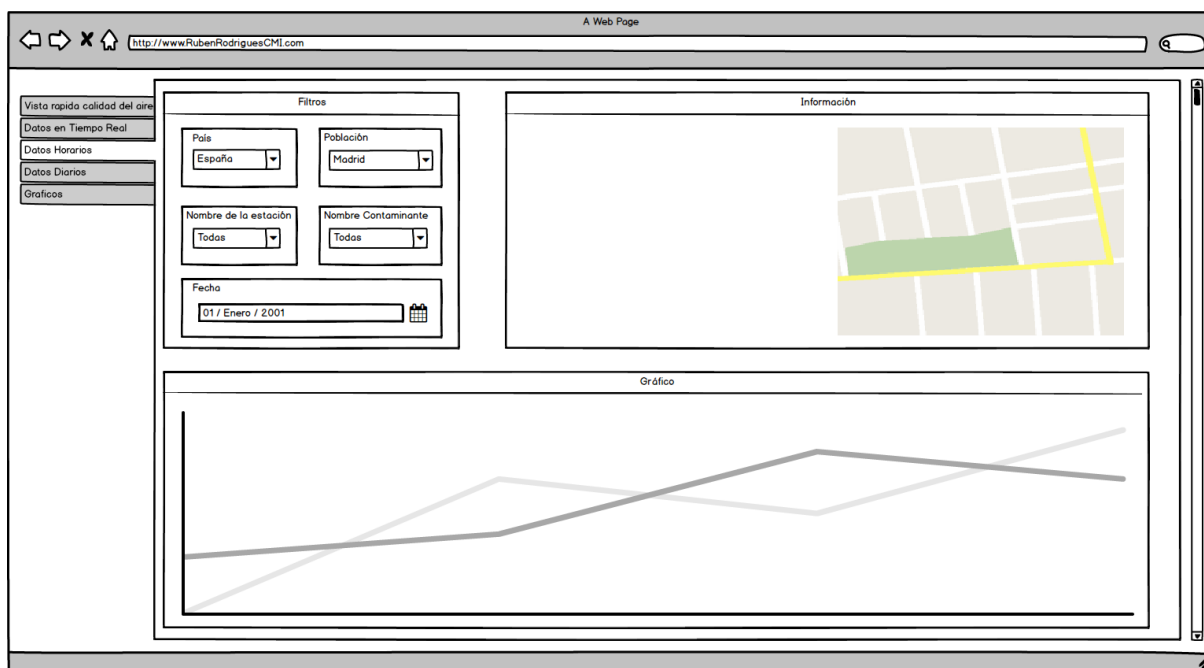


Ilustración 16 - CMI: Datos Horarios

4.6.3 Datos Diarios

En este apartado se podrá visualizar los datos diarios y ver cómo evolucionan en el tiempo. Seleccionando el contaminante y la estación de control y además una fecha de inicio y fin.

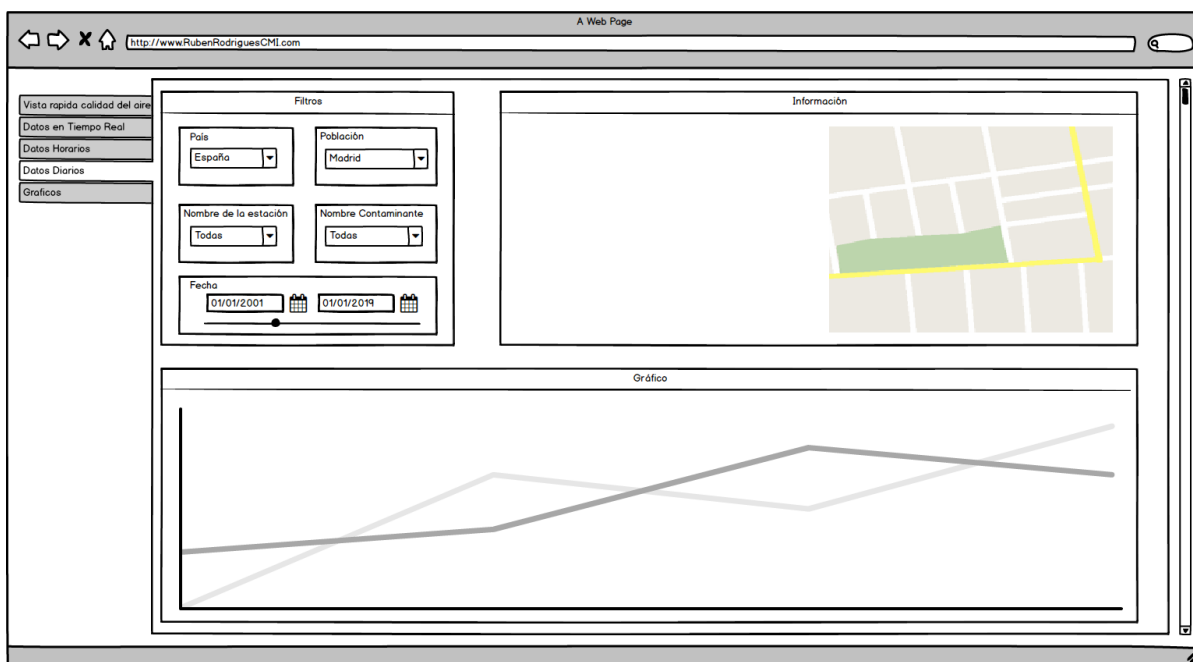


Ilustración 17 - CMI: Datos Diarios

4.6.4 Gráficas

En este apartado se puede visualizar mediante unas pestañas gráficos históricos, anuales o mensuales. Se trata de realizar una gráfica con todos los datos obtenidos o con algunos que se seleccionen a placer.

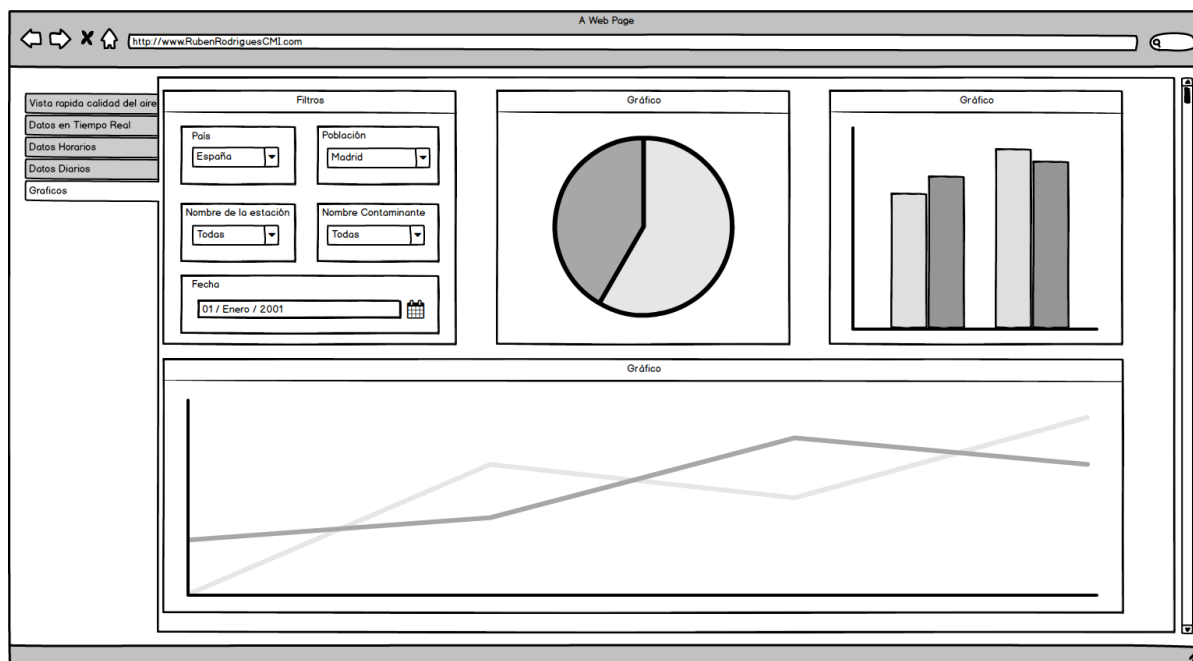


Ilustración 18 - CMI: Gráficos

5. Implementación del proyecto

A estas alturas del proyecto, solo queda realizar la implantación de este. Así pues, se detallan detenidamente todos los pasos que se van a realizar desde el inicio hasta la elaboración de CMI, esto incluye obtención de datos, instalaciones de software y manipulaciones de software necesarias para la consecución del objetivo.

5.1 El entorno de trabajo


El entorno de trabajo es esencial en un proyecto de *Business Intelligence*. Por lo general, y debido a la dimensión que puede acarrear, se debería realizar en un servidor o *cloud*. Como no se dispone de un servidor propio para realizar las tareas necesarias, se va a virtualizar un Windows Server 2019.

La virtualización se realizará en un ordenador personal, el cual, en este caso, tiene las siguientes características:

Edición de Windows

Windows 10 Pro

© 2018 Microsoft Corporation. Todos los derechos reservados.



Sistema

Procesador:	Intel(R) Core(TM) i5-3570K CPU @ 3.40GHz 3.40 GHz
Memoria instalada (RAM):	16,0 GB
Tipo de sistema:	Sistema operativo de 64 bits, procesador x64
Lápiz y entrada táctil:	La entrada táctil o manuscrita no está disponible para esta pantalla

Ilustración 19 - Información maquina anfitrión

Para ello se debe descargar la ISO desde la propia web de Microsoft Azure, ya que ofrece gratuitamente el Windows Server 2019. Se procederá a acceder a la web de descarga (<https://portal.azure.com/?Microsoft Azure Education correlationId=b0a3a712-9b12-4148-a348-57fab1323a47#blade/Microsoft Azure Education/EducationMenuBlade/software>) y se descargará:

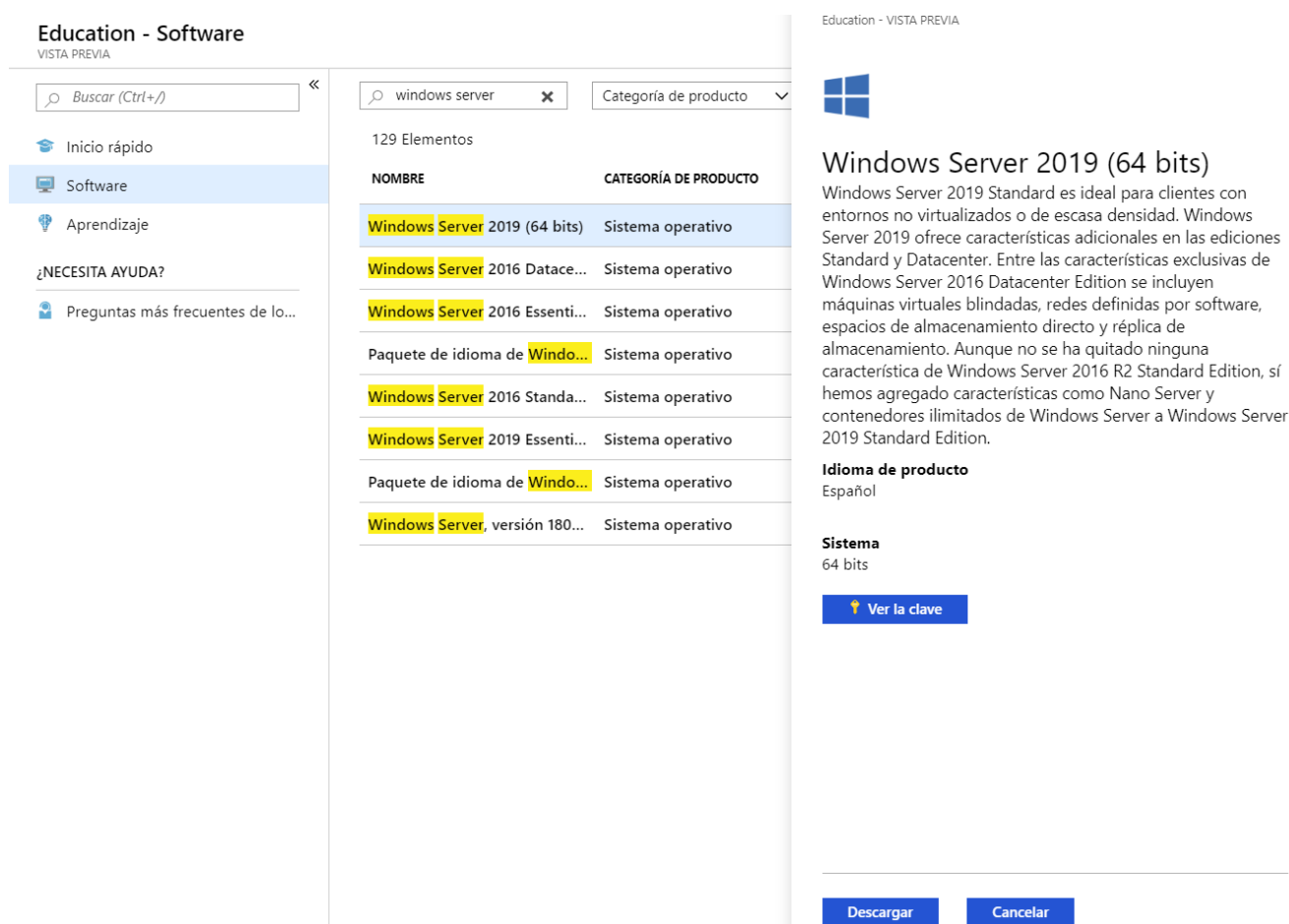


Ilustración 20 - Windows Azure: Windows Server 2019

Para poder instalar esta ISO, es necesario tener activado el sistema de virtualización de Windows (Hyper-V), ya que no viene preinstalado en el sistema operativo.

Para ello procedemos a activarlo, desde un pc con Windows 10, se accede al panel de control → Programas y características → Activar o desactivar características de Windows

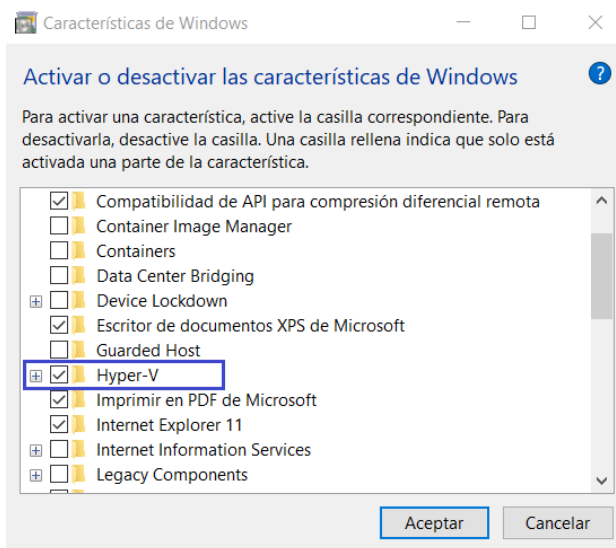


Ilustración 21 - Activar Hyper-V en Windows 10

Una vez instalado y reiniciado el ordenador, se procederá a lanzar Hyper-V. En el administrador de Hyper-V se puede crear una máquina virtual. Para la creación de la máquina virtual, en la barra de tareas se va a Acción → Creación rápida. Se deberá seleccionar el origen local de la imagen que anteriormente se ha descargado y se creará la máquina virtual con Windows Server 2019.

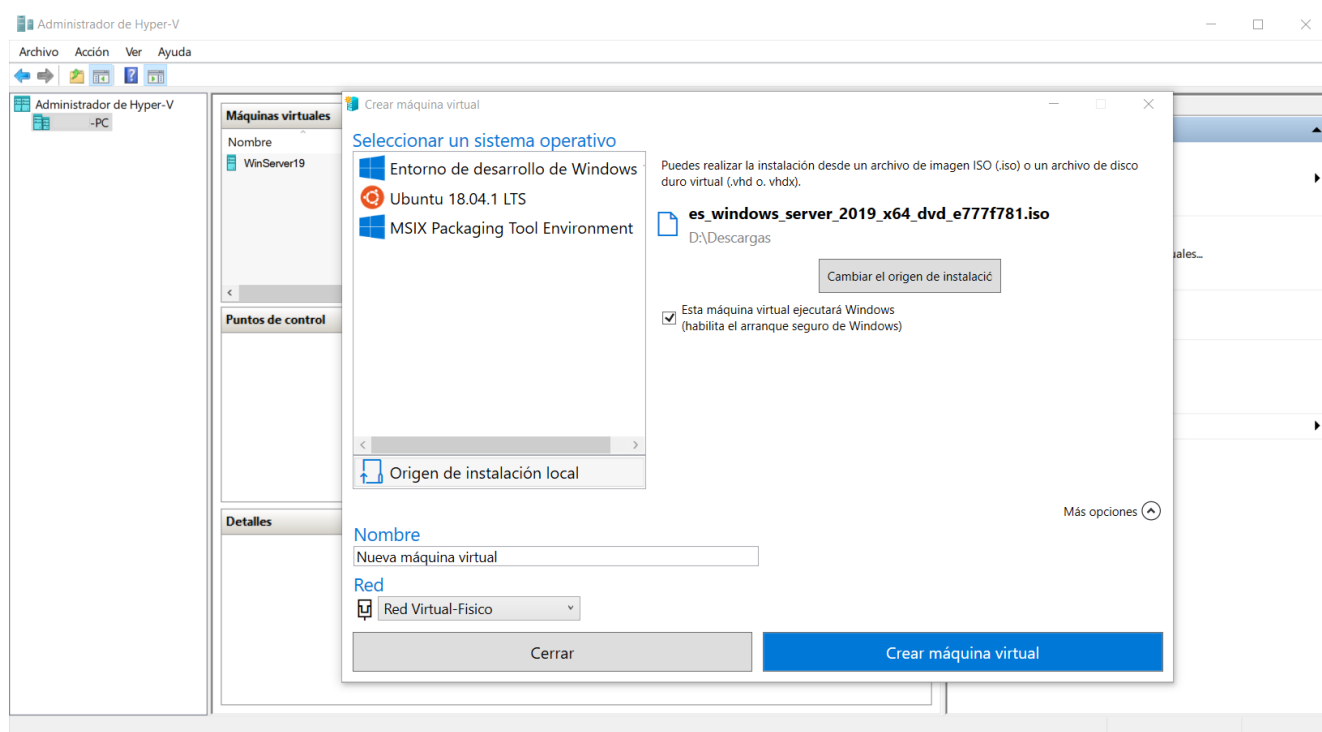


Ilustración 22 - Creación Máquina Virtual WinServer16

Se inicia la máquina virtual, y en el primer arranque del sistema se procederá a instalar el Sistema Operativo.

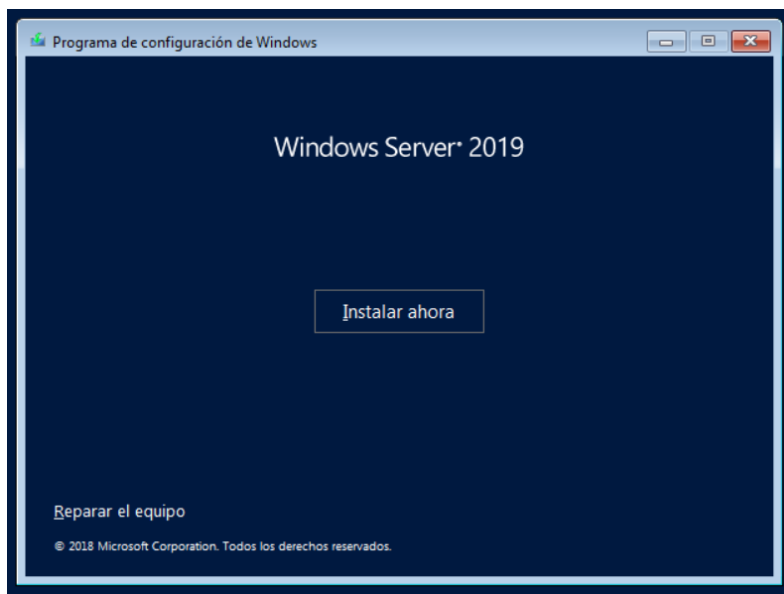


Ilustración 23 - Instalación WinServer19

Las características que tiene el servidor “WinServer19” son las siguientes:

Ver información básica acerca del equipo

Edición de Windows

Windows Server 2019 Standard

© 2018 Microsoft Corporation. Todos los derechos reservados.

Windows Server® 2019

Sistema

Procesador: Intel(R) Core(TM) i5-3570K CPU @ 3.40GHz 3.42 GHz
Memoria instalada (RAM): 8,00 GB
Tipo de sistema: Sistema operativo de 64 bits, procesador x64
Lápiz y entrada táctil: La entrada táctil o manuscrita no está disponible para esta pantalla

5.2 Software *PowerBI Desktop*

Como se ha explicado con anterioridad, con la herramienta Microsoft *PowerBI Desktop* se tiene todo lo que se necesita ya que incorpora el sistema gestor de base de datos Microsoft SQL server express, la creación de procesos ETL mediante *Power Query* y también se obtiene la creación de informes mediante *Power Pivot*.

Para la instalación del software, se debe descargarlos desde la web oficial que ofrece Microsoft: <https://www.microsoft.com/es-ES/download/details.aspx?id=45331>.

Cuando finalice la descarga, se procederá con su instalación en el servidor y posterior configuración.

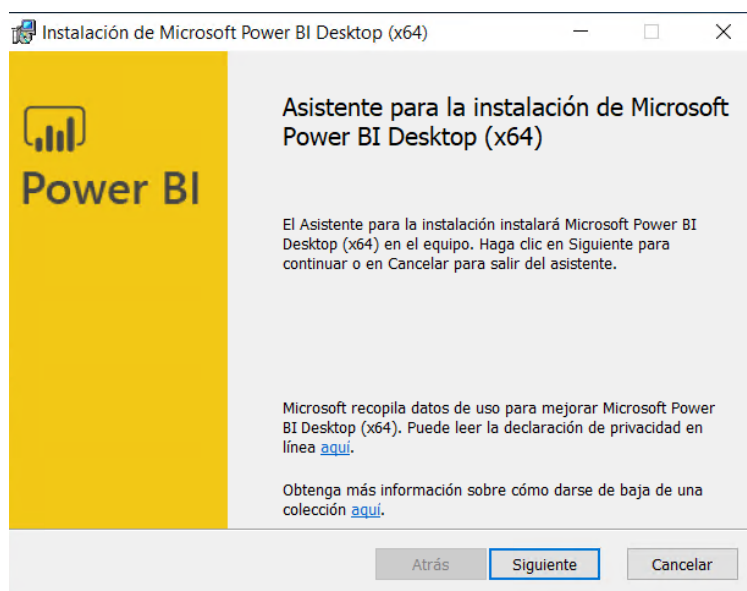


Ilustración 24 - Instalación de PowerBi Desktop

Una vez instalado, la primera vez que se abre es necesario dar unos datos previos. Los cuales se rellenarán y se iniciará sesión en el programa con la cuenta educativa de la universidad. Esto dará una ventaja, porque se podrá posteriormente publicar el CMI en la nube de Microsoft, con lo que será accesible por internet.

Una vez iniciada la sesión con la cuenta educativa, aparecerá la *Intro* del programa donde sugiere de donde se tiene que obtener los datos para empezar a trabajar. Se puede ver a continuación:

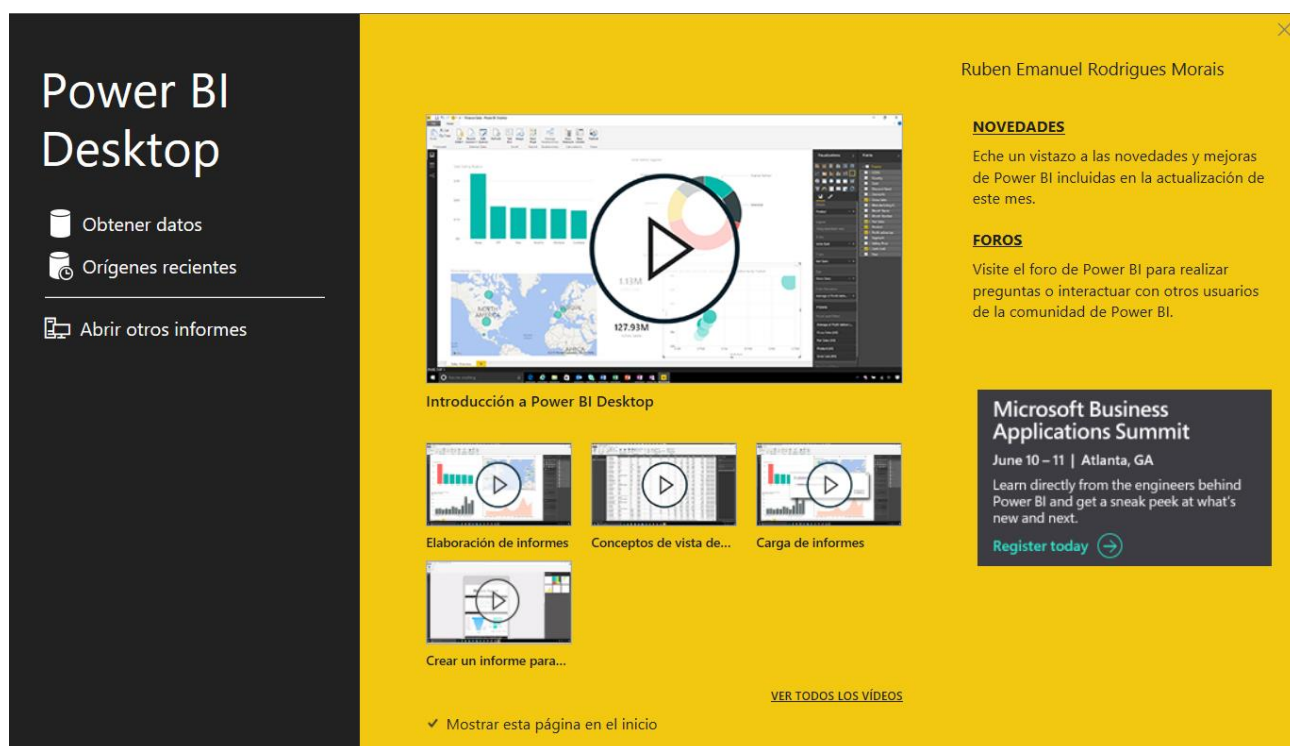


Ilustración 25 – Inicio PowerBI Desktop

5.3 Obtención del *DataSource*

Como en otros apartados se ha indicado, se trata de un proyecto para la medición de calidad del aire de cualquier ciudad. Para realizar un proyecto real, se debe usar datos reales, por lo que se va a utilizar una *DataSource* procedente de la comunidad de Madrid (CM).

Así pues, se debe ir al repositorio abierto que ofrecen a todos los ciudadanos para descargar todos los datos que se tiene sobre las estaciones de control autónomas y poder usarlos para el actual proyecto.

Para ello se irá a la siguiente dirección: <https://datos.madrid.es/portal/site/egob>

Desde este portal se podrá ir al catálogo de datos y filtrar por los datos que se quiera, que en este caso serán los datos medioambientales. Y así se obtendrá lo siguiente:

Conjuntos de datos




Listado con el catálogo completo de conjuntos de datos que el Ayuntamiento de Madrid pone a disposición de la ciudadanía. Este listado puede descargarlo en formato **RDF**, (**DCAT**) o **CSV**. De cada dataset se presenta su sector, la fecha en la que se incorporó al catálogo, su periodicidad, el número de descargas totales y los formatos en los que está disponible.

Se pueden realizar filtros utilizando los elementos que se despliegan en **'Filtrar por...'** para limitar los resultados a ciertos **sectores**, **formatos** y **periodicidades**, además de ordenar por nombre y fecha de incorporación.




Filtrar por... >

Total: **4** Mostrados: **1-4**



Calidad del aire. Datos diarios años 2001 a 2019

Sector: Medio ambiente	Actualización: Mensual	  
Descargas: 35407	Fecha de incorporación: 12/03/2014	

Calidad del aire. Datos en tiempo real

Sector: Medio ambiente	Actualización: Tiempo real	  
Descargas: 658159	Fecha de incorporación: 09/04/2015	

Calidad del aire. Datos horarios años 2001 a 2019

Sector: Medio ambiente	Actualización: Mensual	 
Descargas: 21844	Fecha de incorporación: 12/03/2014	

Calidad del aire. Estaciones de control


Sector: Medio ambiente	Actualización: Anual	
Descargas: 32073	Fecha de incorporación: 09/04/2015	

Ilustración 26 - Repositorio abierto de datos CM

Se procederá a descargar todos los datos, tanto los diarios, como horarios, y las estaciones de control. En cuanto a los datos en tiempo real, no se van descargar porque se actualizarán los datos del CMI directamente desde el repositorio de la CM.

5.4 Creación de nuestro *DataWarehouse*

Una de las características del software *PowerBI* es que permite la creación de proyectos directamente optando por un *datawarehouse* externo y conectándose a él, o, por otro lado, permite la creación del nuestro propio mediante la importación desde diferentes fuentes de información.

En proyectos de gran escala, se optaría evidentemente, por la inclusión de un *datawarehouse* externo, para no tener todo el groso en un mismo entorno. Como se trata de un proyecto de tamaño reducido y para evitar la complejidad y la posible aparición de incompatibilidades entre softwares y versiones, se ha optado por anularlo todo en el mismo programa de Microsoft.

5.4.1 Conexión con el *Datasource*

Como se ha decidido por la inclusión del *datawarehouse* en el propio software, se dispone a ver cómo se importa todos los archivos para la creación de este. En el anterior apartado se ha visto que se ha obtenido todos los archivos del que es el conjunto de datos. Ahora queda importarlos a *PowerBI* para la creación del *datawarehouse*.

Se va a reflejar cómo se importa un conjunto de datos desde el software. En este caso se va a ver cómo se importa desde un servicio web el cual va a servir para la recolección de datos en tiempo real.

Cuando se inicia *PowerBI* pide que se cree un origen de datos o que se cargue:

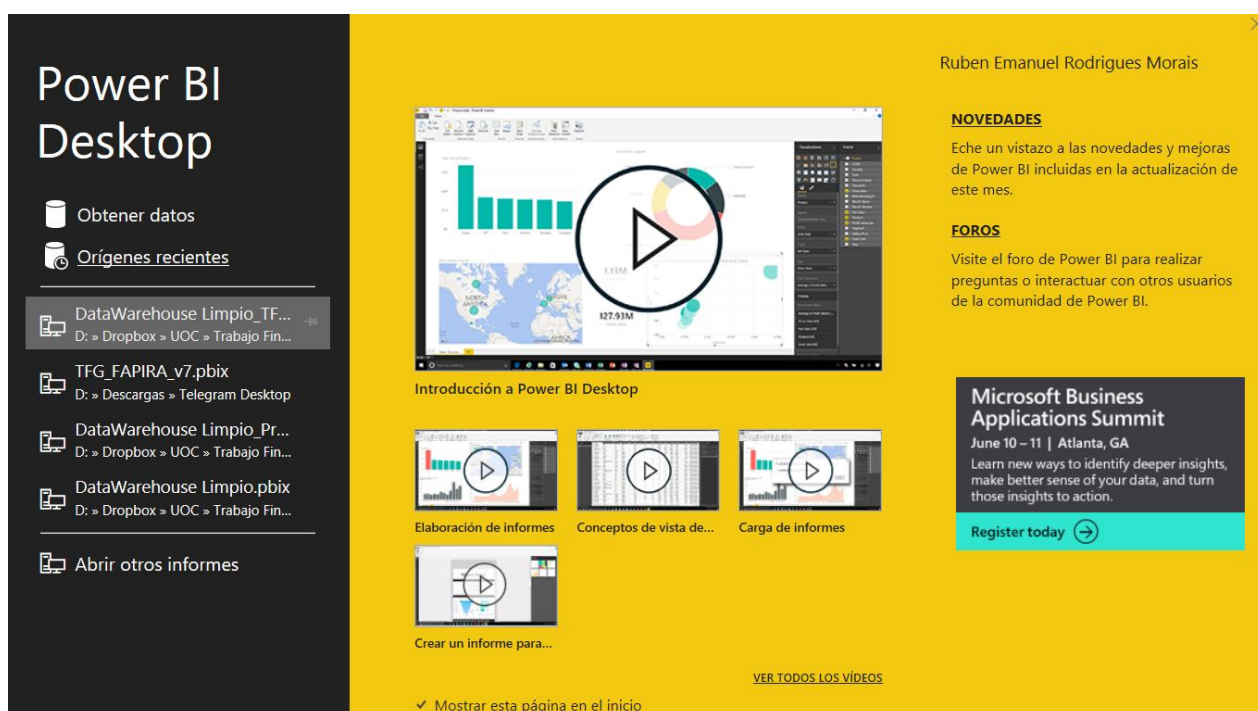


Ilustración 27 - Obtener datos PowerBI

Hay que seleccionar “Obtener Datos” para importar el propio *datawarehouse*. *PowerBI* ofrece muchas alternativas para la importación de éste, desde bases de datos conocidas como pueden ser SQL Server, Access, MySQL, PostgreSQL, SAP Hana, pasando por servicios web como URLs, Facebook, Github y hasta el propio sistema de Azure o archivos csv, txt o Excel.

En este caso se va a seleccionar “Web” y solo se tendrá que poner la dirección de la que se quiere recuperar datos, que, en este caso será donde se cuelgan a cada hora en punto los datos de las estaciones de control del aire de la comunidad de Madrid (<https://datos.madrid.es/egob/catalogo/212531-10515086-calidad-aire-tiempo-real.csv>)

Una vez que se introduce la web saldrá la información que contiene y se podrá editar y transformarla:

https://datos.madrid.es/egob/catalogo/212531-10515086-calidad-aire-tiempo-re...

Origen de archivo: 1252: Europeo occidental (Windows) | Delimitador: Punto y coma | Detección del tipo de datos: Basado en las primeras 200 filas

PROVINCIA	MUNICIPIO	ESTACION	MAGNITUD	PUNTO_MUESTREO	ANO	MES	DIA	H01	V01	H02	V02	H03
28	79	4	1	28079004_1_38	2019	5	26	2	V	2	V	2
28	79	4	6	28079004_6_48	2019	5	26	3	V	3	V	3
28	79	4	7	28079004_7_8	2019	5	26	5	V	6	V	6
28	79	4	8	28079004_8_8	2019	5	26	15	V	15	V	17
28	79	4	12	28079004_12_8	2019	5	26	23	V	23	V	27
28	79	8	1	28079008_1_38	2019	5	26	9	V	9	V	9
28	79	8	6	28079008_6_48	2019	5	26	1	V	1	V	1
28	79	8	7	28079008_7_8	2019	5	26	2	V	1	V	2
28	79	8	8	28079008_8_8	2019	5	26	29	V	23	V	21
28	79	8	9	28079008_9_47	2019	5	26	6	V	6	V	3
28	79	8	10	28079008_10_47	2019	5	26	10	V	10	V	7
28	79	8	12	28079008_12_8	2019	5	26	32	V	25	V	24
28	79	8	14	28079008_14_6	2019	5	26	5011	V	4989	V	5175
28	79	8	20	28079008_20_59	2019	5	26	7	V	5	V	5
28	79	8	30	28079008_30_59	2019	5	26	2	V	2	V	1
28	79	8	35	28079008_35_59	2019	5	26	1	V	1	V	1
28	79	8	42	28079008_42_2	2019	5	26	153	V	152	V	150
28	79	8	43	28079008_43_2	2019	5	26	149	V	149	V	148
28	79	8	44	28079008_44_2	2019	5	26	4	V	3	V	3
28	79	11	7	28079011_7_8	2019	5	26	1	V	1	V	1

Cargar Editar Cancelar

Ilustración 28 - Carga de datos mediante URL

Cuando se cargue ya se obtendrá *datawarehouse* con los primeros datos cargados para posteriormente editarlos y transformarlos.

La importación de todo el conjunto de datos estará en los anexos.

5.4.2 Creación de tablas

Como se ha visto con anterioridad, las estaciones de control del aire recaban una serie de datos. Estos datos pueden ser en tiempo real, diarios o horarios. Debido a la particularidad de estos y para responder a las preguntas iniciales marcadas como objetivos se ha tenido que realizar la creación de tres tablas para la inclusión de estos.

Además, y para otorgar más información de valor al proyecto, se han creado dos tablas más con información sobre las estaciones de control y los contaminantes medidos, para que sea más informativo y no tan estadístico como se pretendía.

Así pues, se ha procedido a realizar las tablas que con anterioridad se habían definido y se han relacionado entre ellas mediante sus claves primarias para la posterior explotación de datos.

Se puede hacer la creación de las tablas de dos formas, mediante código utilizado por *PowerBI*, o gráficamente. A continuación, se muestra la realización de una de las tablas.

Gráficamente:

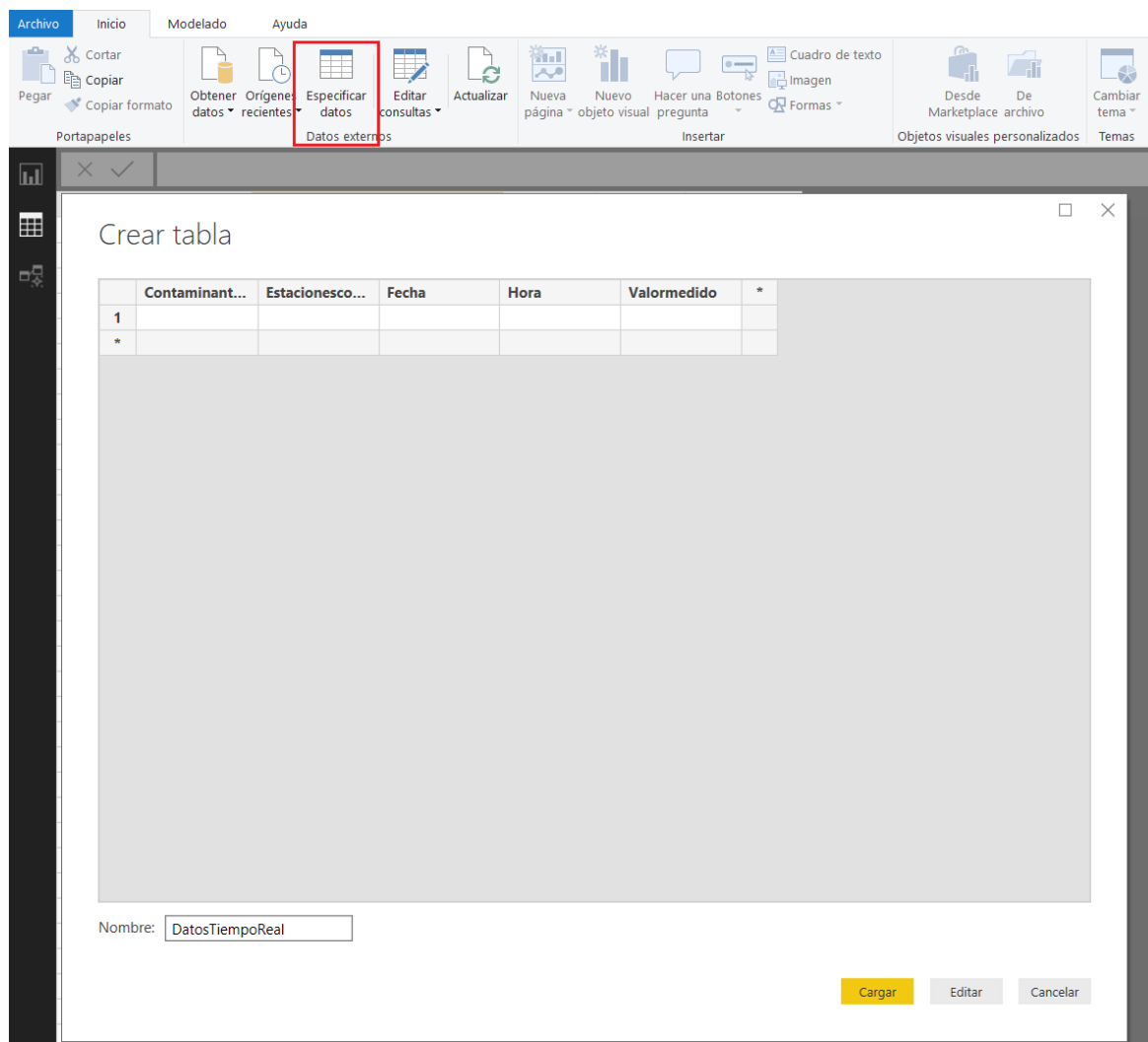


Ilustración 29 - Creación de tabla en PowerBI (Interfaz)

Mediante Código:

```
= Table.FromRows(Json.Document(Binary.Decompress(Binary.FromText("i44FAA==", BinaryEncoding.Base64), Compression.Deflate)), let _t = ((type text) meta [Serialized.Text = true]) in type table [Contaminantes_Magnitud = _t, EstacionesControl_NumeroEstacion = _t, Fecha = _t, Hora = _t, ValorMedido = _t])
```

Ilustración 30 - Creación tabla PowerBI (Código)

Como se observa la creación de tablas es muy sencillo, una vez que se crean todas las tablas que se necesitan, en este caso 5, se debe proceder a relacionarlas entre ellas.

Como todo el software, la relación de las tablas es sencilla e intuitiva y se realiza mediante la utilidad “Administrar Relaciones”. Tiene un sistema de autodetección de relaciones muy interesante, aunque en este caso se han realizado las relaciones manualmente quedando de la siguiente forma:

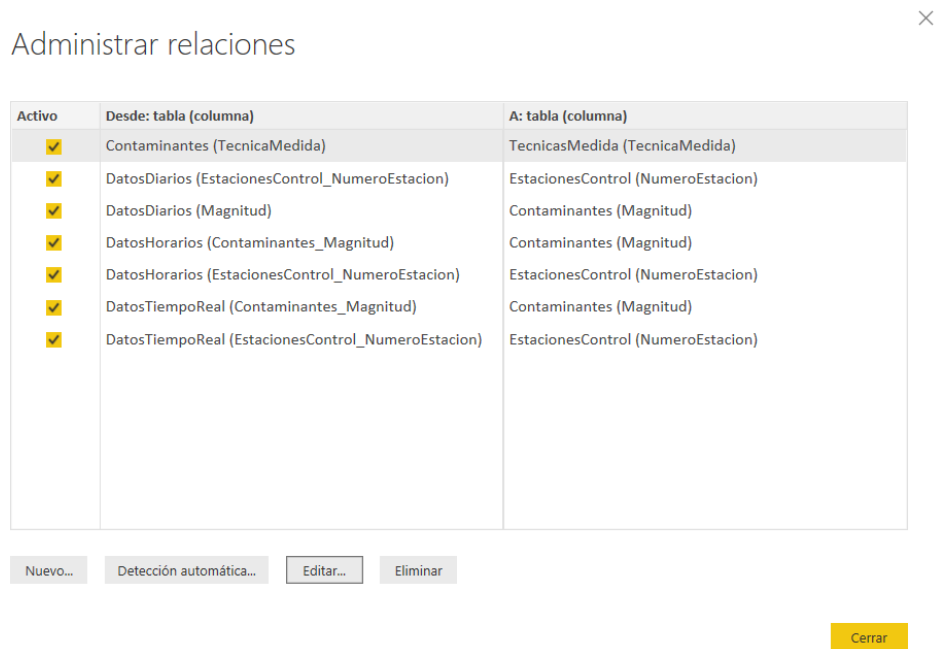


Ilustración 31 - Relaciones entre tablas PowerBI

Y de una manera más tradicional quedaría como podemos ver:

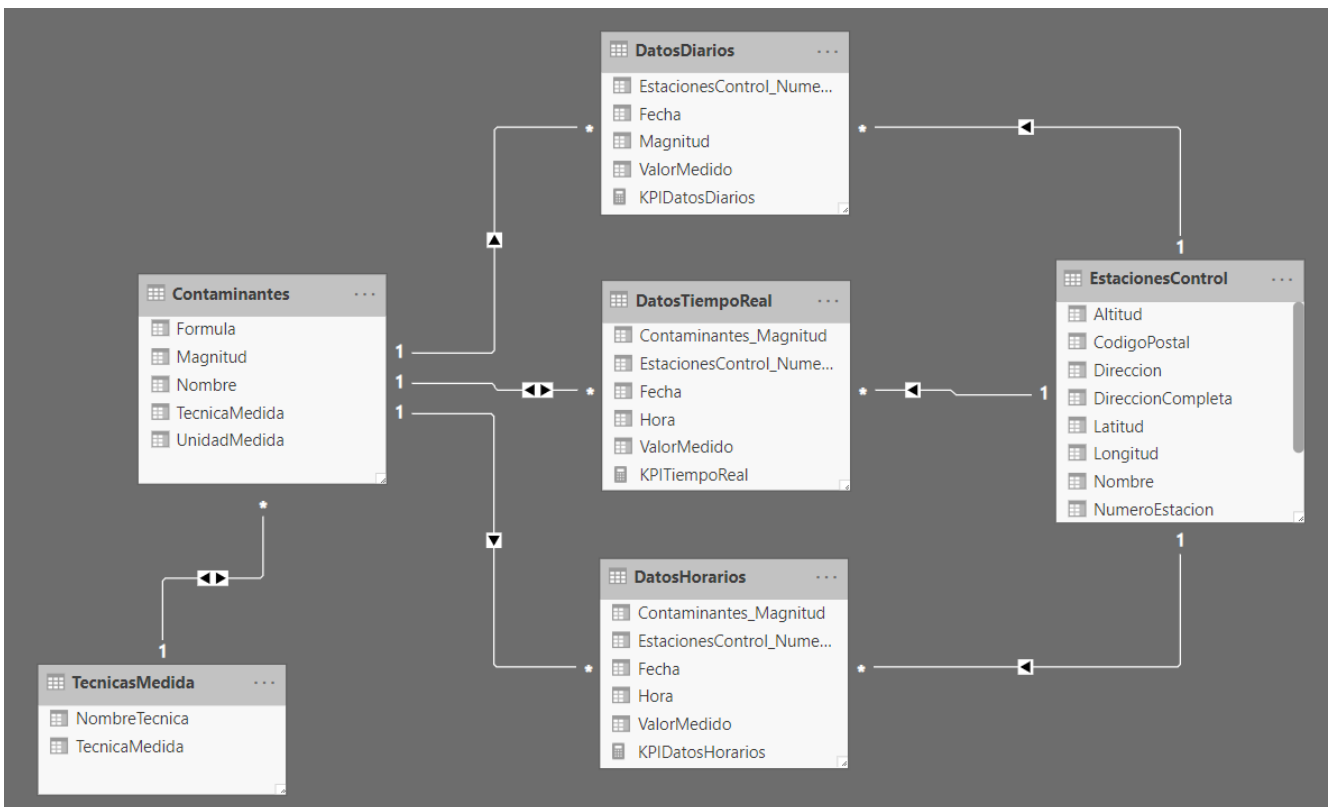


Ilustración 32- Relaciones entre tablas UML

La realización de todas las tablas estará en los anexos.

5.5 Procesos ETL

La manera de gestionar los procesos ETL en *PowerBI* es mediante la herramienta interna *Power Query*, con la que se puede realizar multitud de acciones, tales como:

- Buscar y conectar los orígenes de datos.
- Realizar búsquedas online de una gran cantidad de fuentes de datos públicas.
- Combinar datos de diferentes fuentes y ordenarlos para después explotarlos con *Power Pivot*, *Power View* y *Power Map*.
- Con *Power View* se pueden compartir y gestionar nuestras consultas creadas
- Se puede realizar una comunicación cruzada con Excel para compartir datos desde prácticamente cualquier fuente de datos.

Llegados a este paso, ya se tiene el *datawarehouse* construido, y todas tablas creadas, pero se deben de nutrir con información. Para ello, existen los procesos ETL, que se encargan de extraer, transformar y cargar la información.

Se debe de tener en cuenta que en el origen de datos existe mucha información irrelevante, además, se debe formatear los datos e incluso combinar algunos. En estas tablas existe una particularidad y es que todos los datos no son válidos. Existen datos incorrectos (formateados con una N) y existen datos correctos (Formateados con una V).

A continuación, se va a mostrar el código de un proceso ETL mediante el cual filtramos, combinamos, formateamos y estructuramos todos los datos que se tienen para así conseguir los datos necesarios para los objetivos que se han establecido. Estos procesos se deberán realizar para cada origen de datos anual y después anexarlas a la tabla principal.

Proceso ETL para la tabla DatosHorarios:

```

1 let
2   -- Origen = Folder:Files("D:\Dropbox\UOC\Trabajo Fin de Grado\DataSource\Datos Horarios\Anio2019"),
3   -- #Archivos ocultos filtrados1 = Table.SelectRows(Origen, each [Attributes][Hidden]? <= true),
4   -- #Invocar función personalizada1 = Table.AddColumn(#Archivos ocultos filtrados1, "Transformar archivo de Anio2019", each #"Transformar archivo de Anio2019"[Content]),
5   -- #Columnas con nombre cambiado1 = Table.RenameColumns(#"Invocar función personalizada1", {"Name", "Source.Name"}),
6   -- #Otras columnas quitadas1 = Table.SelectColumns(#"Columnas con nombre cambiado1", {"Source.Name", "Transformar archivo de Anio2019"}),
7   -- #Columna de tabla expandida1 = Table.ExpandTableColumn(#"Otras columnas quitadas1", "Transformar archivo de Anio2019", Table.ColumnNames(#"Transformar archivo de Anio2019"("Archivo de ejemplo (5)")),
8   -- #Columnas quitadas = Table.RemoveColumns(#"Columna de tabla expandida1", {"PROVINCIA", "MUNICIPIO", "PUNTO_MUESTREO"}),
9   -- #Errores quitados = Table.RemoveRowsWithErrors(#"Columnas quitadas", {"ESTACION"}),
10  -- #Columnas combinadas = Table.CombineColumns(Table.TransformColumnTypes(#"Errores quitados", {"#H01", type text}}, "es-ES",{"H01", "V01"},Combiner.CombineTextByDelimiter(", ", QuoteStyle.None), "Combinada"),
11  -- #Columnas combinadas1 = Table.CombineColumns(Table.TransformColumnTypes(#"Columnas combinadas", {"#H02", type text}}, "es-ES",{"H02", "V02"},Combiner.CombineTextByDelimiter(", ", QuoteStyle.None), "Combinada.1"),
12  -- #Columnas combinadas2 = Table.CombineColumns(Table.TransformColumnTypes(#"Columnas combinadas1", {"#H03", type text}}, "es-ES",{"H03", "V03"},Combiner.CombineTextByDelimiter(", ", QuoteStyle.None), "Combinada.2"),
13  -- #Columnas combinadas3 = Table.CombineColumns(Table.TransformColumnTypes(#"Columnas combinadas2", {"#H04", type text}}, "es-ES",{"H04", "V04"},Combiner.CombineTextByDelimiter(", ", QuoteStyle.None), "Combinada.3"),
14  -- #Columnas combinadas4 = Table.CombineColumns(Table.TransformColumnTypes(#"Columnas combinadas3", {"#H05", type text}}, "es-ES",{"H05", "V05"},Combiner.CombineTextByDelimiter(", ", QuoteStyle.None), "Combinada.4"),
15  -- #Columnas combinadas5 = Table.CombineColumns(Table.TransformColumnTypes(#"Columnas combinadas4", {"#H06", type text}}, "es-ES",{"H06", "V06"},Combiner.CombineTextByDelimiter(", ", QuoteStyle.None), "Combinada.5"),
16  -- #Columnas combinadas6 = Table.CombineColumns(Table.TransformColumnTypes(#"Columnas combinadas5", {"#H07", type text}}, "es-ES",{"H07", "V07"},Combiner.CombineTextByDelimiter(", ", QuoteStyle.None), "Combinada.6"),
17  -- #Columnas combinadas7 = Table.CombineColumns(Table.TransformColumnTypes(#"Columnas combinadas6", {"#H08", type text}}, "es-ES",{"H08", "V08"},Combiner.CombineTextByDelimiter(", ", QuoteStyle.None), "Combinada.7"),
18  -- #Columnas combinadas8 = Table.CombineColumns(Table.TransformColumnTypes(#"Columnas combinadas7", {"#H09", type text}}, "es-ES",{"H09", "V09"},Combiner.CombineTextByDelimiter(", ", QuoteStyle.None), "Combinada.8"),
19  -- #Columnas combinadas9 = Table.CombineColumns(Table.TransformColumnTypes(#"Columnas combinadas8", {"#H10", type text}}, "es-ES",{"H10", "V10"},Combiner.CombineTextByDelimiter(", ", QuoteStyle.None), "Combinada.9"),
20  -- #Columnas combinadas10 = Table.CombineColumns(Table.TransformColumnTypes(#"Columnas combinadas9", {"#H11", type text}}, "es-ES",{"H11", "V11"},Combiner.CombineTextByDelimiter(", ", QuoteStyle.None), "Combinada.10"),
21  -- #Columnas combinadas11 = Table.CombineColumns(Table.TransformColumnTypes(#"Columnas combinadas10", {"#H12", type text}}, "es-ES",{"H12", "V12"},Combiner.CombineTextByDelimiter(", ", QuoteStyle.None), "Combinada.11"),
22  -- #Columnas combinadas12 = Table.CombineColumns(Table.TransformColumnTypes(#"Columnas combinadas11", {"#H13", type text}}, "es-ES",{"H13", "V13"},Combiner.CombineTextByDelimiter(", ", QuoteStyle.None), "Combinada.12"),
23  -- #Columnas combinadas13 = Table.CombineColumns(Table.TransformColumnTypes(#"Columnas combinadas12", {"#H14", type text}}, "es-ES",{"H14", "V14"},Combiner.CombineTextByDelimiter(", ", QuoteStyle.None), "Combinada.13"),
24  -- #Columnas combinadas14 = Table.CombineColumns(Table.TransformColumnTypes(#"Columnas combinadas13", {"#H15", type text}}, "es-ES",{"H15", "V15"},Combiner.CombineTextByDelimiter(", ", QuoteStyle.None), "Combinada.14"),
25  -- #Columnas combinadas15 = Table.CombineColumns(Table.TransformColumnTypes(#"Columnas combinadas14", {"#H16", type text}}, "es-ES",{"H16", "V16"},Combiner.CombineTextByDelimiter(", ", QuoteStyle.None), "Combinada.15"),
26  -- #Columnas combinadas16 = Table.CombineColumns(Table.TransformColumnTypes(#"Columnas combinadas15", {"#H17", type text}}, "es-ES",{"H17", "V17"},Combiner.CombineTextByDelimiter(", ", QuoteStyle.None), "Combinada.16"),
27  -- #Columnas combinadas17 = Table.CombineColumns(Table.TransformColumnTypes(#"Columnas combinadas16", {"#H18", type text}}, "es-ES",{"H18", "V18"},Combiner.CombineTextByDelimiter(", ", QuoteStyle.None), "Combinada.17"),
28  -- #Columnas combinadas18 = Table.CombineColumns(Table.TransformColumnTypes(#"Columnas combinadas17", {"#H19", type text}}, "es-ES",{"H19", "V19"},Combiner.CombineTextByDelimiter(", ", QuoteStyle.None), "Combinada.18"),
29  -- #Columnas combinadas19 = Table.CombineColumns(Table.TransformColumnTypes(#"Columnas combinadas18", {"#H20", type text}}, "es-ES",{"H20", "V20"},Combiner.CombineTextByDelimiter(", ", QuoteStyle.None), "Combinada.19"),
30  -- #Columnas combinadas20 = Table.CombineColumns(Table.TransformColumnTypes(#"Columnas combinadas19", {"#H21", type text}}, "es-ES",{"H21", "V21"},Combiner.CombineTextByDelimiter(", ", QuoteStyle.None), "Combinada.20"),
31  -- #Columnas combinadas21 = Table.CombineColumns(Table.TransformColumnTypes(#"Columnas combinadas20", {"#H22", type text}}, "es-ES",{"H22", "V22"},Combiner.CombineTextByDelimiter(", ", QuoteStyle.None), "Combinada.21"),
32  -- #Columnas combinadas22 = Table.CombineColumns(Table.TransformColumnTypes(#"Columnas combinadas21", {"#H23", type text}}, "es-ES",{"H23", "V23"},Combiner.CombineTextByDelimiter(", ", QuoteStyle.None), "Combinada.22"),
33  -- #Columnas combinadas23 = Table.CombineColumns(Table.TransformColumnTypes(#"Columnas combinadas22", {"#H24", type text}}, "es-ES",{"H24", "V24"},Combiner.CombineTextByDelimiter(", ", QuoteStyle.None), "Combinada.23"),
34  -- #Columnas con nombre cambiado = Table.RenameColumns(#"Columnas combinadas23", {"Combinada", "01:00"}, {"Combinada.1", "02:00"}, {"Combinada.2", "03:00"}, {"Combinada.3", "04:00"}, {"Combinada.4", "05:00"}, {"Combinada.5", "06:00"}, {"Combinada.6", "07:00"}, {"Combinada.7", "08:00"}, {"Combinada.8", "09:00"}, {"Combinada.9", "10:00"}, {"Combinada.10", "11:00"}, {"Combinada.11", "12:00"}, {"Combinada.12", "13:00"}, {"Combinada.13", "14:00"}, {"Combinada.14", "15:00"}, {"Combinada.15", "16:00"}, {"Combinada.16", "17:00"}, {"Combinada.17", "18:00"}, {"Combinada.18", "19:00"}, {"Combinada.19", "20:00"}, {"Combinada.20", "21:00"}, {"Combinada.21", "22:00"}, {"Combinada.22", "23:00"}, {"Combinada.23", "00:00"}),
35  -- #Otras columnas con anulación de dinamización = Table.UpivotOtherColumns(#"Columnas con nombre cambiado", {"ESTACION", "MAGNITUD", "ANO", "MES", "DIA", "Atributo", "Valor"},
36  -- #Columnas con nombre cambiado = Table.RenameColumns(#"Otras columnas con anulación de dinamización", {"Atributo", "Hora"}),
37  -- #Filas filtradas = Table.SelectRows(#"Columnas con nombre cambiado", each Text.EndsWith(Valor, "V")),
38  -- #Texto extraído antes del delimitador = Table.TransformColumns(#"Filas filtradas", {"Valor", each Text.BeforeDelimiter(", ", type text)}),
39  -- #Columnas combinadas24 = Table.CombineColumns(#"Texto extraído antes del delimitador", {"ANO", "MES", "DIA"},Combiner.CombineTextByDelimiter("-", QuoteStyle.None), "Fecha"),
40  -- #Tipo cambiado = Table.TransformColumnTypes(#"Columnas combinadas24", {"Hora", type time}, {"Fecha", type date}), {"Valor", type number}}, "es-ES"},
41  -- #Tipo cambiado con configuración regional = Table.TransformColumnTypes(#"Tipo cambiado", {"Valor", type number}}, "es-CU"},
42  -- #Tipo cambiado = Table.TransformColumnTypes(#"Tipo cambiado con configuración regional", {"ESTACION", Int64.Type}, {"MAGNITUD", Int64.Type}),
43  -- #Columnas con nombre cambiado = Table.RenameColumns(#"Tipo cambiado", {"MAGNITUD", "Contaminantes_Magnitud"}, {"ESTACION", "EstacionesControl_NumeroEstacion"}, {"Valor", "ValorMedido"}),
44  -- #Columnas reordenadas = Table.ReorderColumns(#"Columnas con nombre cambiado", {"Contaminantes_Magnitud", "EstacionesControl_NumeroEstacion", "Fecha", "Hora", "ValorMedido"}),
45  in
46  -- #Columnas reordenadas
    
```

Ilustración 33- Proceso ETL para DatosHorarios2019

Y los anexamos a la tabla principal:

```

1 let
2   -- Origen = Table.FromRows(Json.Document(Binary.Decompress(Binary.FromText("i44FAA==", BinaryEncoding.Base64), Compression.Deflate)), let _t = (
3   -- (type text) meta [Serialized.Text = true]) in type table [Contaminantes_Magnitud = _t, EstacionesControl_NumeroEstacion = _t, Fecha = _t,
4   -- Hora = _t, ValorMedido = _t]),
5   -- #Tipo cambiado = Table.TransformColumnTypes(Origen,{{"Contaminantes_Magnitud", Int64.Type}, {"EstacionesControl_NumeroEstacion", Int64.Type},
6   -- {"Fecha", type date}, {"Hora", type time}, {"ValorMedido", type number}}),
7   -- #Consulta anexada = Table.Combine({#"Tipo cambiado", Horarios2019, Horarios2018, Horarios2017_hastaOct, Horarios2017_desdeOct, Horarios2016,
8   -- Horarios2015})
9   in
10  -- #Consulta anexada
    
```

Ilustración 34 - Proceso ETL para anexar a tabla principal DatosHorarios

La realización de todos los procesos ETL estarán en los anexos.

5.6 Cuadro de Mando Integral (CMI)

Una vez llegados a este punto solo queda la creación del CMI, en base a los mockups que se habían realizado con anterioridad. La herramienta a utilizar, como bien había visto en la selección del software CMI, es *PowerBI*. En base a esto y las especificaciones de diseño que tiene el software, los mockups se han visto ligeramente modificados, aunque los datos que se muestran son los mismos.

El CMI que se han implantado es interactivo, eso significa que se seleccionarán unos filtros básicos, como la ubicación, el contaminante a evaluar... pero además de esto, se puede interactuar con los gráficos, mediante jerarquías de fechas o selección de partes del mismo gráfico.

Este CMI está preparado para su inclusión en cualquier departamento medioambiental de cualquier país, puesto que se han introducido la selección del país y la población.

El actual *dashboard* cuenta con cinco páginas diferentes, en las cuales se podrán consultar distintos datos. Además, se ha desarrollado cada una de estas páginas para versión de escritorio y la aplicación móvil de *PowerBI*. A continuación, se muestran cada página del CMI y a realizar una breve descripción de lo que se puede consultar:

VistaRápidaCalidadAire: En este apartado se mostrará de un rápido vistazo y seleccionando los filtros de ubicación y contaminante, el estado actual de las estaciones de control autónomas. Además, también se mostrará un mapa con la ubicación de todas las estaciones de control que existen, el cual irá interactuándose y moviéndose según se filtre por el nombre de estación. También mostraremos una pequeña información de la estación de control: nombre, dirección, población, país, así como latitud y longitud de su ubicación.

El estado de la estación se mostrará mediante un semáforo de colores, el cual está filtrado según el contaminante y su rango de permisividad. Esto se había determinado en los indicadores de rendimiento (KPI).

Existen cuatro estados de colores: gris significa que no existen rangos para el filtrado de peligrosidad; verde significa que la calidad del aire es buena; amarillo significa que la calidad del aire es admisible; naranja significa que la calidad del aire es deficiente; rojo significa que la calidad del aire es mala.

Versión de escritorio:

Vista rápida de la calidad del aire en tiempo real

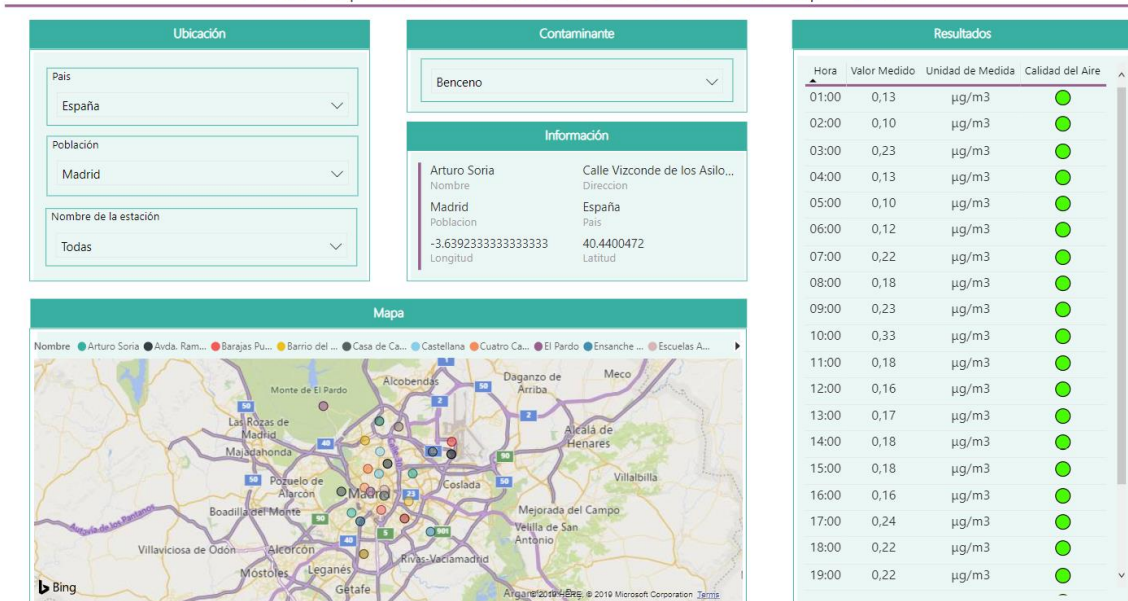


Ilustración 35 - Vista rápida de la calidad del aire en tiempo real (Versión escritorio)

Versión móvil:

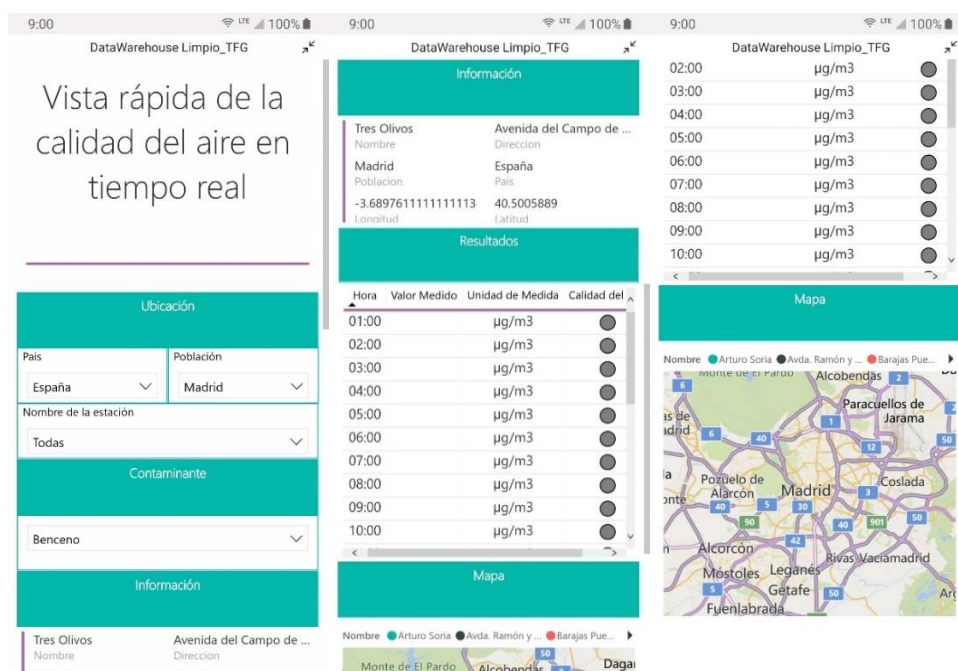


Ilustración 36 - Vista rápida de la calidad del aire en tiempo real (Versión móvil)

DatosTiempoReal: En este apartado se mostrará de una forma más detallada los datos de la calidad del aire. Estos datos serán mostrados según los filtros que apliquemos, que serán la ubicación de la estación, así como su nombre y el contaminante que se quiera medir. Además, también se mostrará un mapa con la ubicación de todas las estaciones de control que existen, el cual irá interactuándose y moviéndose según se filtre por el nombre de estación. También se mostrará una pequeña información de la estación de control: nombre, dirección, población, país, así como latitud y longitud de su ubicación. Por otro lado, se visualizará mediante una gráfica, la cual es interactiva y se podrá ir haciendo clic en las distintas horas, la evolución del contaminante durante las horas del día. Por último, se mostrará el valor de la medición correspondiente a la hora y el correspondiente estado en ese momento exacto de la estación de control.

El estado de la estación se mostrará mediante un semáforo de colores, el cual está filtrado según el contaminante y su rango de permisividad. Esto se había determinado en los indicadores de rendimiento (KPI).

Por otro lado, hay cuatro estados de colores: gris significa que no existen rangos para el filtrado de peligrosidad; verde significa que la calidad del aire es buena; amarillo significa que la calidad del aire es admisible; naranja significa que la calidad del aire es deficiente; rojo significa que la calidad del aire es mala.

Versión escritorio:

Datos de la calidad del aire en tiempo real

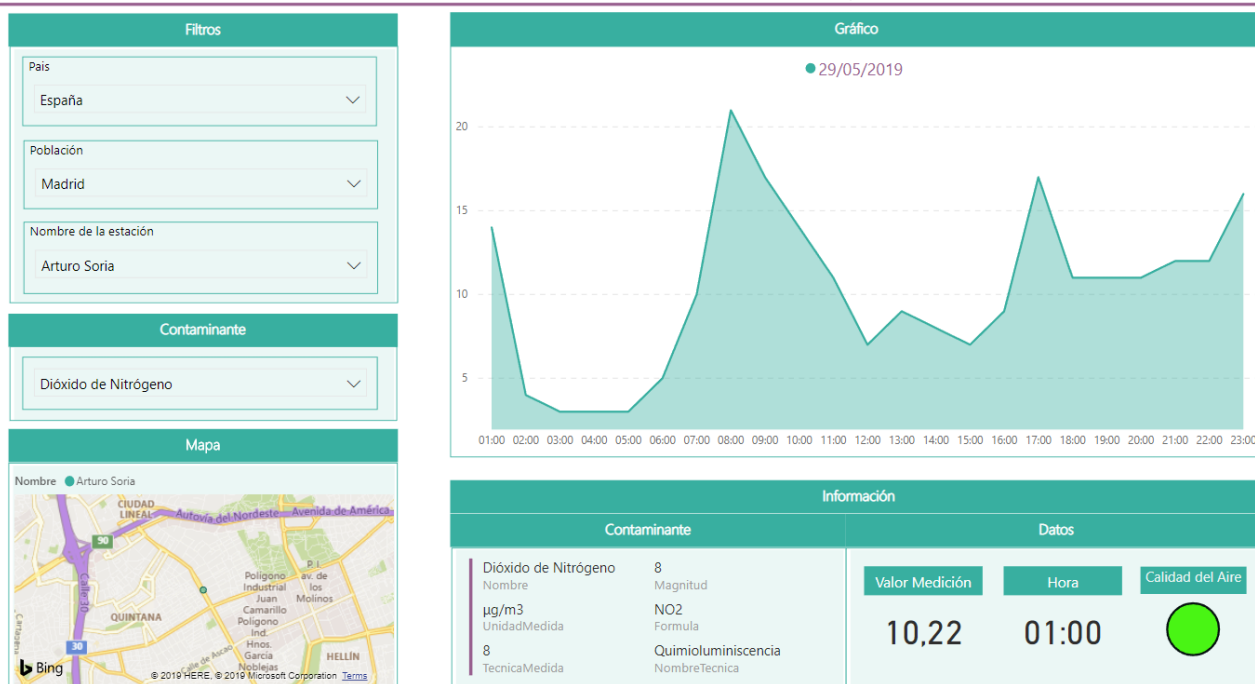


Ilustración 37 - Datos de la calidad del aire en tiempo real (Versión escritorio)

Versión móvil:

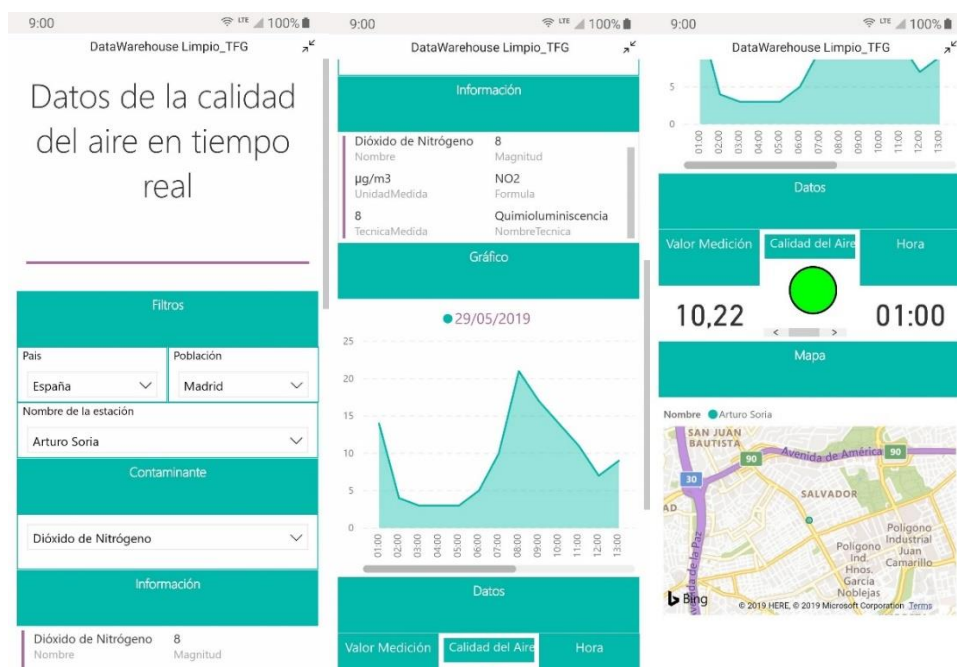


Ilustración 38 - Datos de la calidad del aire en tiempo real (Versión móvil)

DatosHorarios: En este apartado se mostrarán los datos horarios para cada estación y según los contaminantes. Estos datos serán mostrados según los filtros que se apliquen, que serán la ubicación de la estación, así como su nombre, los contaminantes que se quieran medir y la fecha. Además,

Versión móvil:

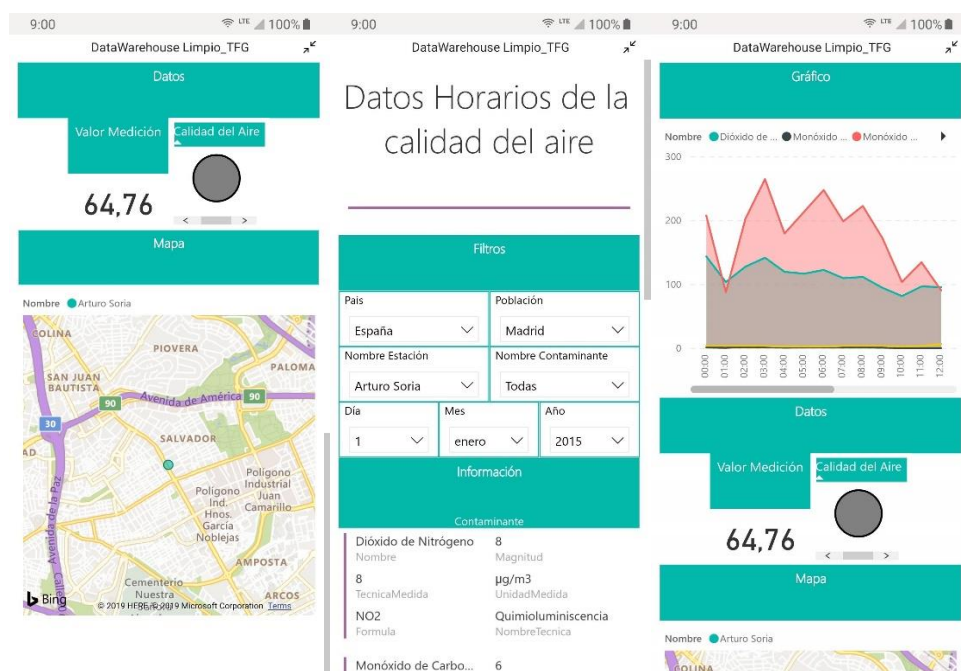


Ilustración 40 -Datos horarios de la calidad del aire en tiempo real (Versión móvil)

Datos Diarios: En este apartado se mostrarán los datos diarios para cada estación y según los contaminantes. Estos datos serán mostrados según los filtros que se apliquen, que serán la ubicación de la estación, así como su nombre, los contaminantes que se quieran medir y la fecha. Además, también se mostrará un mapa con la ubicación de todas las estaciones de control que existen, el cual irá interactuándose y moviéndose según se filtre por el nombre de estación.

También se mostrará una pequeña información de la estación de control: nombre, dirección, población, país, así como latitud y longitud de su ubicación. Por otro lado, se visualizará mediante una gráfica, la cual es interactiva y podremos ir haciendo clic en las distintas horas, la evolución del contaminante durante las horas del día. Se podrán seleccionar uno o más contaminantes simultáneamente por lo que se verá la gráfica dividida en colores diferentes.

Por otro lado, el filtro de la fecha será un rango mediante un *slider*, con lo que se podrá seleccionar desde una fecha concreta hasta otra. También, la gráfica será escalable mediante jerarquías de fechas, es decir, que se podrá ir bajando niveles, desde años, trimestres, meses y días mediante los botones de interacción de la gráfica. Por último, se mostrará el valor de la medición correspondiente al día y el correspondiente estado en ese momento exacto de la estación de control.

El estado de la estación se mostrará mediante un semáforo de colores, el cual está filtrado según el contaminante y su rango de permisividad. Esto se había determinado en los indicadores de rendimiento (KPI).

Los cuatro estados tienen asignado un color el cual tiene su significado: gris significa que no existen rangos para el filtrado de peligrosidad; verde significa que la calidad del aire es buena; amarillo significa

que la calidad del aire es admisible; naranja significa que la calidad del aire es deficiente; rojo significa que la calidad del aire es mala.

Versión escritorio:

Datos Diarios de la calidad del aire

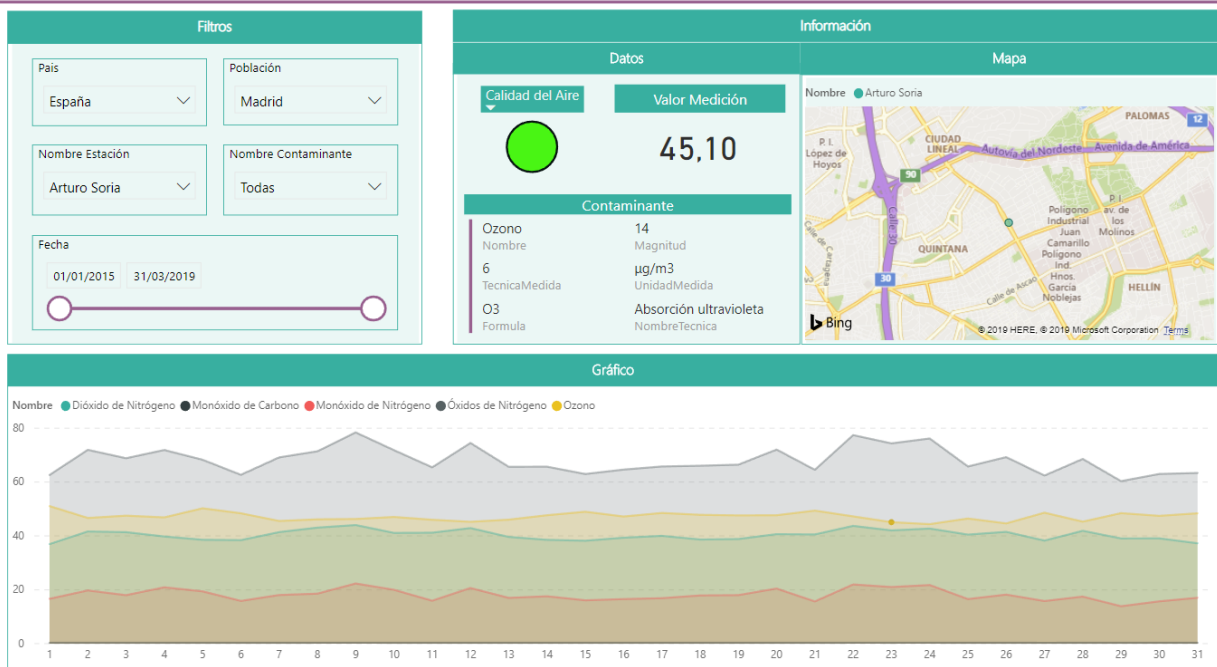


Ilustración 41 -Datos diarios de la calidad del aire en tiempo real (Versión escritorio)

Versión móvil:

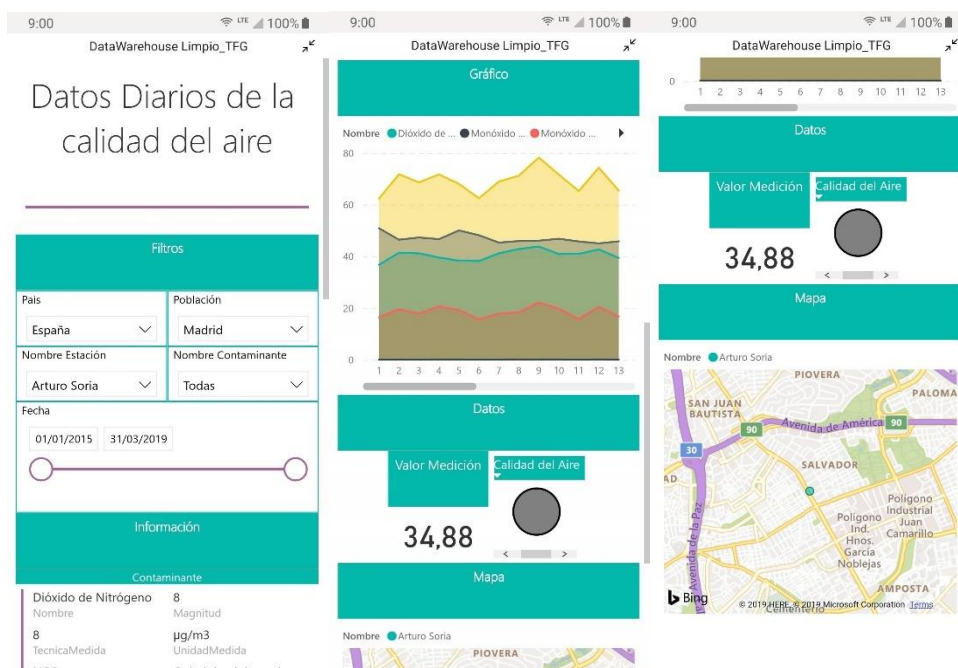


Ilustración 42 -Datos diarios de la calidad del aire en tiempo real (Versión móvil)

Gráficos: En esta última vista se podrá revisar una serie de gráficos indicativos de la calidad del aire. Estos datos vendrán dados mediante los filtros de ubicación, nombre de la estación, nombre de contaminantes y la fecha que se quiera consultar. La fecha, en este caso, puede ser multi fecha, seleccionando varios meses, varios años, o solamente un día en concreto.

Se podrá ver tres gráficos diferentes, los cuales uno de ellos se verá la distribución de cada contaminante mediante un gráfico circular. En otro de ellos se verá la progresión de los contaminantes en las diferentes fechas. Por último, se verá una comparativa del contaminante que se haya seleccionado con el valor en tiempo real en ese mismo momento.

Versión escritorio:

Gráficos de la calidad del aire

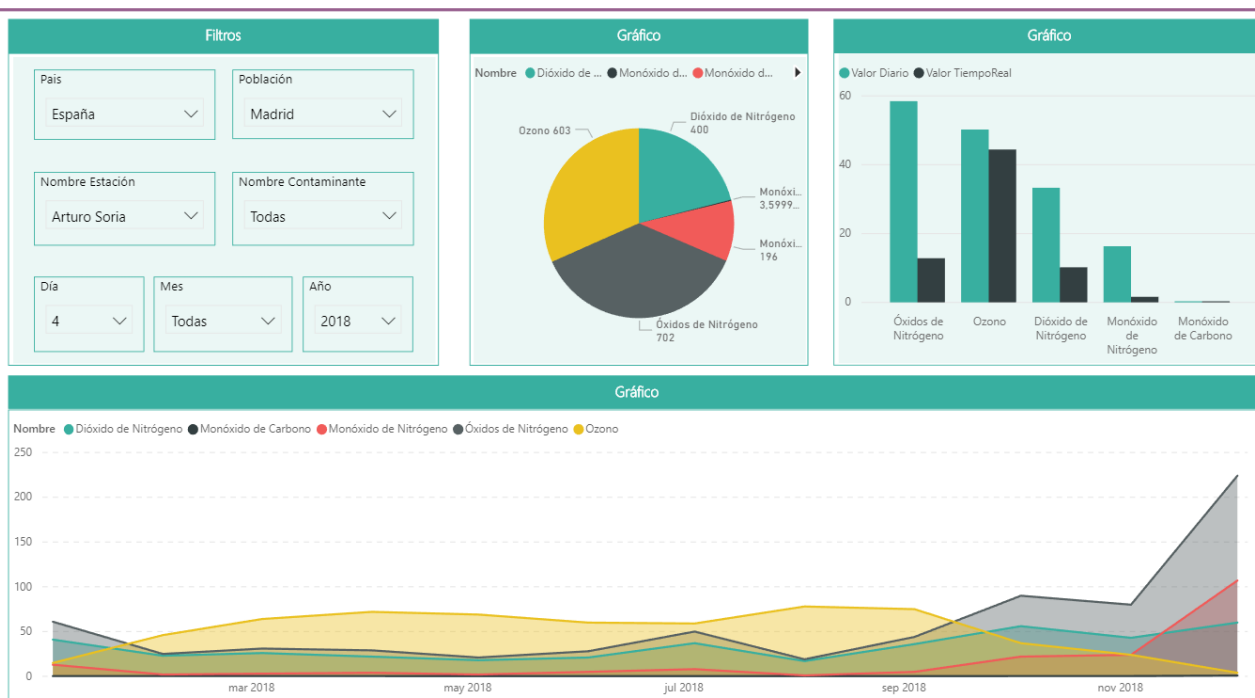


Ilustración 43 -Gráficos de la calidad del aire en tiempo real (Versión escritorio)

Versión móvil:

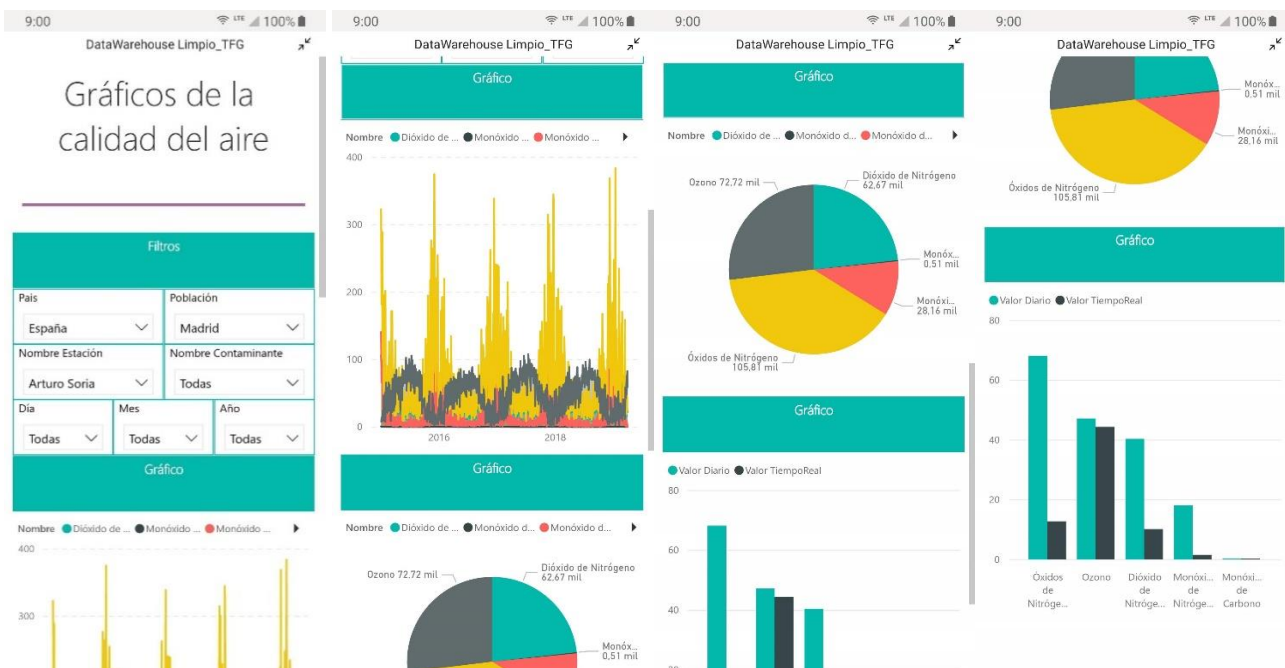


Ilustración 44 -Gráficos de la calidad del aire en tiempo real (Versión móvil)

6. Distribución en plataforma web

Una de las características que se había exigido al software en la elección del mismo era que debía ser capaz de poder exportarse para poder ser visualizado desde un recurso web. El software PowerBI de Microsoft tiene las características de que se puede publicar en su entorno web. Una vez que se ha publicado en su entorno, es capaz de generar un enlace de visualización o un código para embebed para nuestra web.

Se ha optado por la realización de una pequeña web en un hosting personal, donde mediante una plantilla de WordPress se presenta el trabajo.

El enlace para la visualización de la web y la navegación por la misma es el siguiente:

<https://rubenrodriguez.com/WP/index.php>

A continuación, se muestran unas pequeñas capturas de la versión web la cual tiene tres apartados diferenciados, la página principal (se encontrará el *abstract* del proyecto), la página de KPI (donde tenemos todos los indicadores de rendimiento) y la página del *dashboard* (donde se ha incrustado el CMI).

Página de inicio:

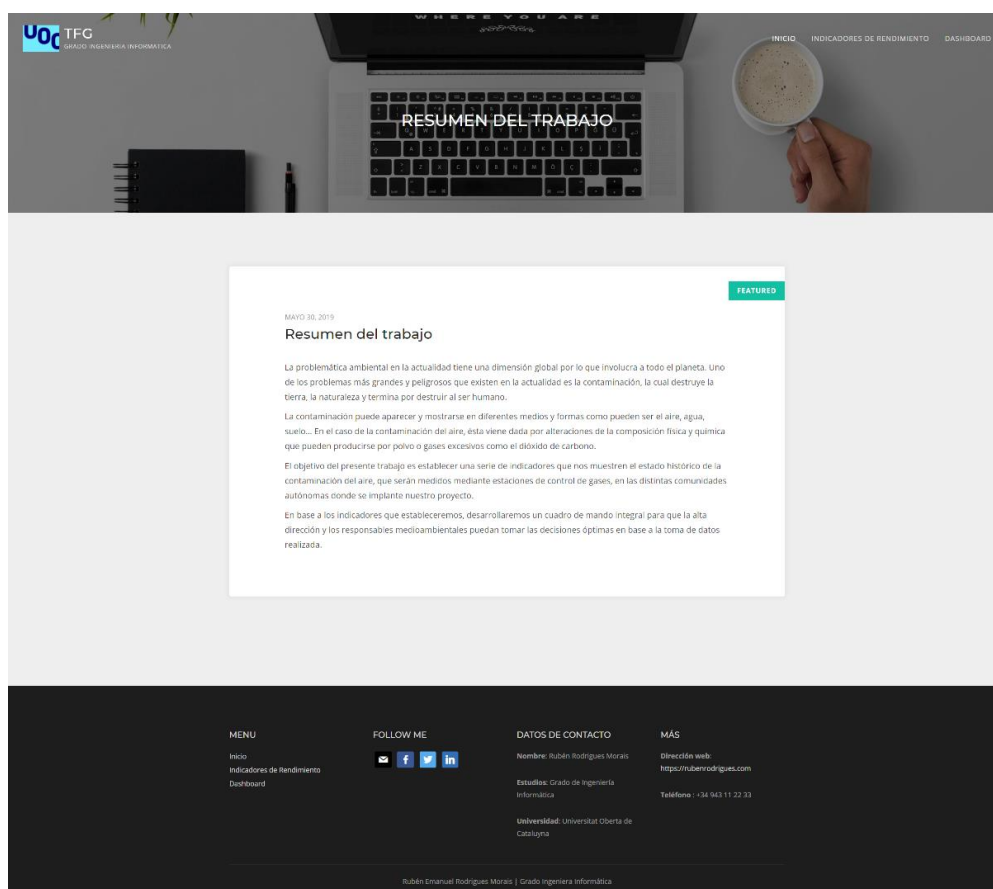


Ilustración 45 - Pagina Inicio de la web

Página de KPIs:



Indicadores de calidad del aire según contaminante

Indicador 1 – Medición de PM₁₀

Nombre	Medición de PM ₁₀
Finalidad	Medir las pequeñas partículas menores de 1 micrómetro dispersas en el aire
Categoría	Operacional
Responsabilidad	Departamento Medioambiental
Definición / Formula	No procede cálculo matemático, se mide en (µg / m ³)
F.M. / F. A.	Cada hora en punto
Inductores / Inducidos	Polvo, ceniza, hollín, partículas metálicas, cemento, polen
Quien lo mide	Estación autónoma de control automática
Fuente de los datos	Base de datos al alcance de cualquier ciudadano
Meta / Real / Umbral	Buena 0-50; Admisible 51-90; Deficiente 91-150; Mala > 150
Iniciativas	Activación de protocolo contra la seguridad medioambiental
Observaciones	Estas partículas pueden aparecer en estado sólido o líquido y son denominadas partículas respirables.

Rangos de calidad del aire según la Comunidad Europea

Indice	Calidad del aire	PM ₁₀ (µg / m ³)
0-50	Buena	0-50
51-100	Admisible	51-90
101-150	Deficiente	91-150
>150	Mala	>150

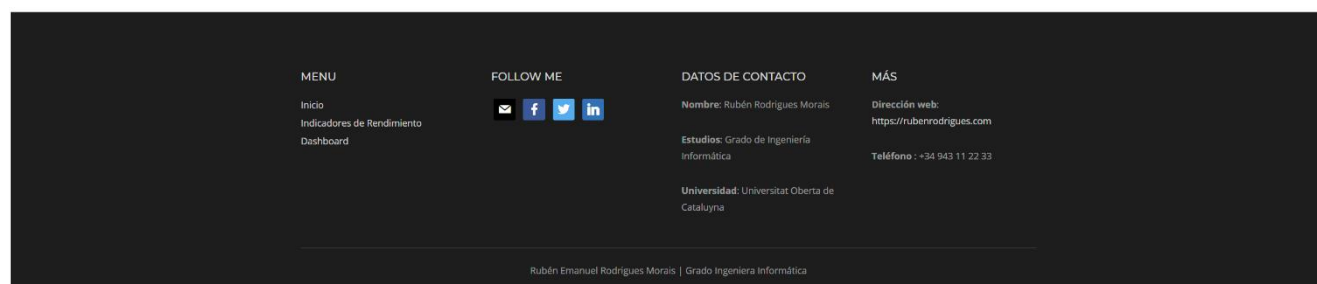


Ilustración 46 - Pagina de KPI de la web

Página de dashboard:



Dashboard interactivo para el control de la calidad del aire

Vista rápida de la calidad del aire en tiempo real

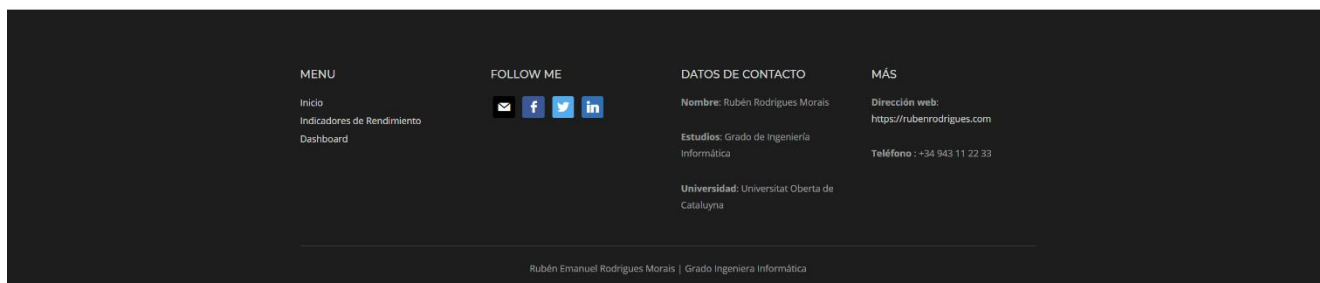
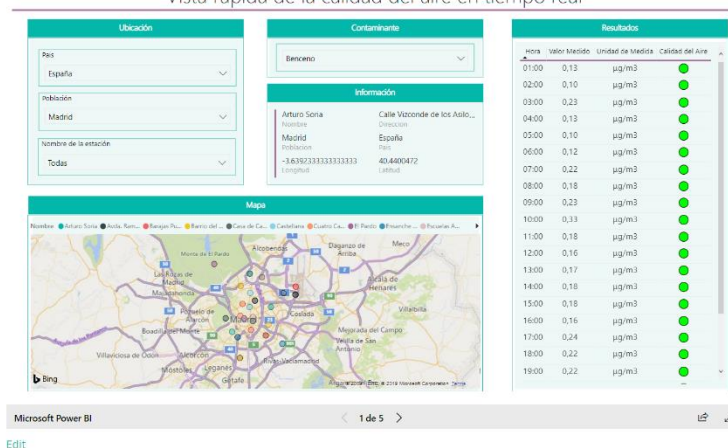


Ilustración 47 - Pagina de CMI de la web

7. Conclusiones

Después de los conocimientos adquiridos a lo largo de todo el Grado de Ingeniería Informática se puede decir que este culmina con el trabajo de fin de grado en el cual se plantea, desarrolla e implanta con éxito un proyecto de Business Intelligence real.

En el inicio del proyecto se planteó como objetivo un sistema de análisis de datos que permitiese la visualización de la calidad del aire en la administración pública, basado en el análisis de los contaminantes mediante mediciones horarias, diarias y en tiempo real. El resultado ha sido muy satisfactorio desde el punto de vista técnico ya que se ha conseguido la implantación del prototipo funcional. Además, se ha añadido soporte mediante web y móvil, por lo que el salto cualitativo es real. Sin embargo, no todo ha sido satisfactorio, la implantación de un sistema de colores en la geolocalización de las estaciones no ha sido posible, por falta de conocimientos y de tiempo.

Este trabajo ha permitido adquirir conocimientos para su desarrollo en el área del Business Intelligence. A su vez, se ha investigado sobre muchas de las herramientas más competitivas del ámbito, pudiendo así seleccionar la que más se adaptaba a las necesidades y aprender sobre ella. Así, la curva de aprendizaje del software utilizado ha sido uno de los escollos del proyecto, pero que se ha subsanado con el paso de las horas.

Creo que una de las claves de la consecución de los objetivos previstos ha sido la planificación del proyecto, la cual se ha tenido que realizar al principio del mismo y ha ayudado a trazar una línea de trabajo que posteriormente se ha agradecido. Aunque es cierto que cuando se ha acercado el tramo final, los plazos ya no eran los marcados y se han ido ajustando a medida. Debido a esto, se ha tenido momentos realmente tensos y de frustración, pensando incluso que no se llegaban a las metas marcadas pudiendo lastrar la progresión del trabajo.

Finalmente, se quiere destacar la importancia que acomete el ser consciente del estado medioambiental del planeta y ser condescendiente con el mismo. Durante el análisis del tema mencionado y la búsqueda de información, se puede ver que el estado actual de las grandes ciudades del mundo es realmente preocupante y no se va a poder reducir la contaminación del mundo mediante este tipo de medidas sin ayuda de la conciencia personal. Se trata de un tema donde todos los seres humanos deben aportar un granito de arena para poder disfrutar de un mundo más limpio y no forzar los posibles desastres naturales que puedan acometerse por la imprudencia.

8. Líneas de futuro

A partir de la experiencia que se ha obtenido tras la realización del trabajo, existen varias líneas de futuro claramente destacadas:

- Migración a un sistema de BBDD

Es necesario la separación de la base de datos del actual software para la realización de cuadro de mando. A la vez que se iba añadiendo repositorios de datos, el software se volvía muy pesado y costaba trabajar con él. Se cree que sería conveniente liberar al software de la realización de tareas propias de las bases de datos y migrarlo todo a un sistema propio para ello. Se podría aprovechar esta línea y las nuevas tecnologías y volcarlo todo en la nube, aprovechando Azure. O, por otro lado, se podría establecer un sistema gestor de base de datos algo más conservador, como puede ser SQL server u Oracle.

- Utilizar un sistema gestor de ETL

Como se ha comentado con anterioridad, sería bueno separar los procesos ETL del software principal, para aliviar la carga de datos. Existen multitud de software para realización de ETL y se cree que sería uno de los pasos importantes a dar.

- Implantación de un sistema de indicadores para la geolocalización

Uno de los objetivos era marcar mediante un semáforo de colores el estado real de cada estación en el mapa al consultado. Esto no se ha podido implantar en el prototipo, pero será una idea de cara a futuro interesante. De un solo vistazo se puede observar si se dispone de alguna estación de control con una calidad del aire no adecuada.

- Ampliación de la localización

Una de las claves de este proyecto es que permite escalabilidad. Se ha implementado en la BBDD campos de localización, por lo que se pueden agregar distintas comunidades o países para su seguimiento. Estaría bien que, de cara a la evolución, se fueran ampliando las comunidades autónomas al presente proyecto para dotarlo de más empaque y darle mayor valor.

- Informes

El CMI solo aporta datos mediante paneles, filtros y gráficas. De cara a un futuro, sería conveniente realizar informes sobre ellos, comparativas y que fuera posible exportarlos.

9. Glosario

Término	Descripción
TFG	Trabajo de fin de grado
BI	Business Intelligence
CMI	Cuadro de mando integral
ETL	Extract, Transform and Load (Extraer, transformar y cargar)
PBIX	Extension del programa Microsoft PowerBI
CSV	Comma Separated Values
KPI	Key performance indicator
CM	Comunidad de Madrid
OMS	Organización Mundial de la Salud
CE	Comunidad Europea
BOE	Boletín Oficial del Estado
PEC	Prueba de evaluación continua
OLTP	OnLine Transaction Processing
OLAP	On-Line Analytical Processing
RD	Real Decreto
ROLAP	Relational On-Line Analytical Processing
PDI	Pentaho Data Integration
GPL	General Public License
ERP	Enterprise Resource Planning
CRM	Customer Relationship Management
API	Application Programming Interface
ISO	International Organization for Standardization
PM ₁₀	Partículas pequeñas de menos de 10 micras
PM _{2.5}	Partículas pequeñas de menos de 2.5 micras
SO ₂	Dióxido de azufre
NO ₂	Dióxido de nitrógeno
CO	Monóxido de carbono
O ₃	Ozono

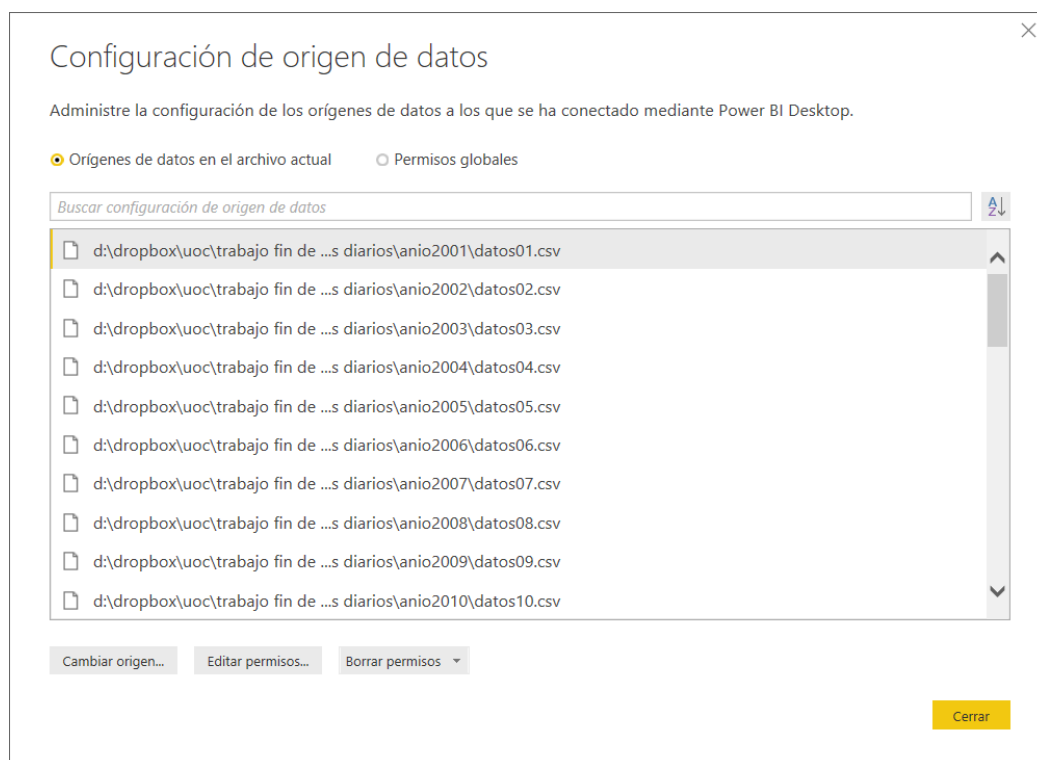
Tabla 21 – Glosario

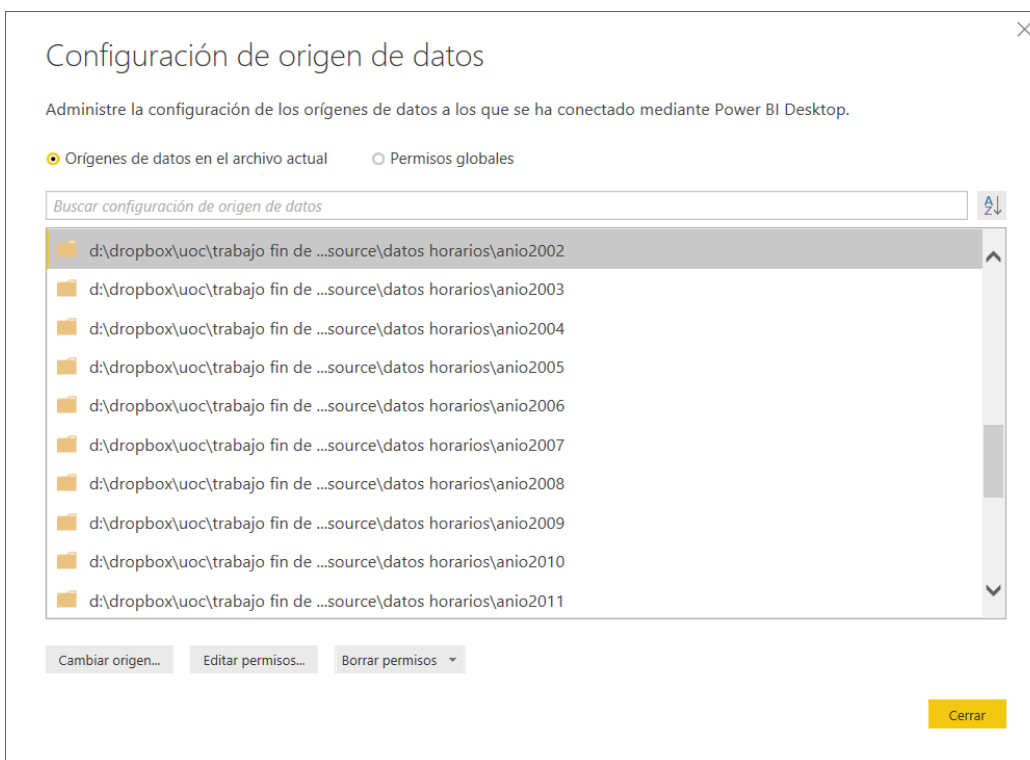
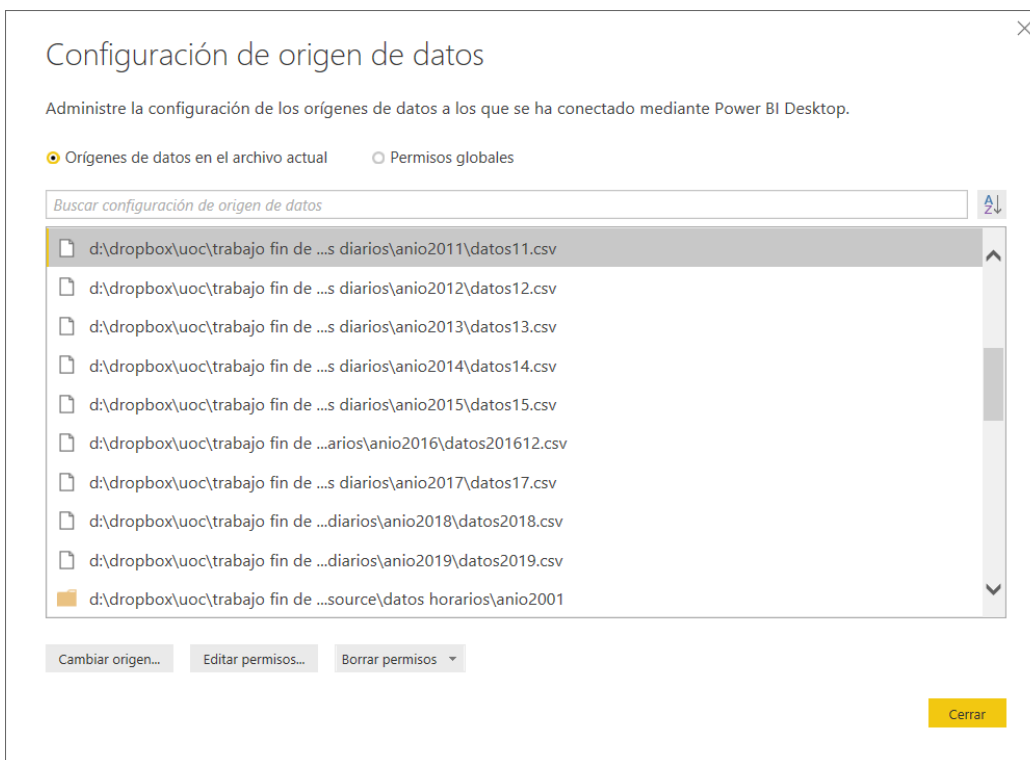
10. Anexos

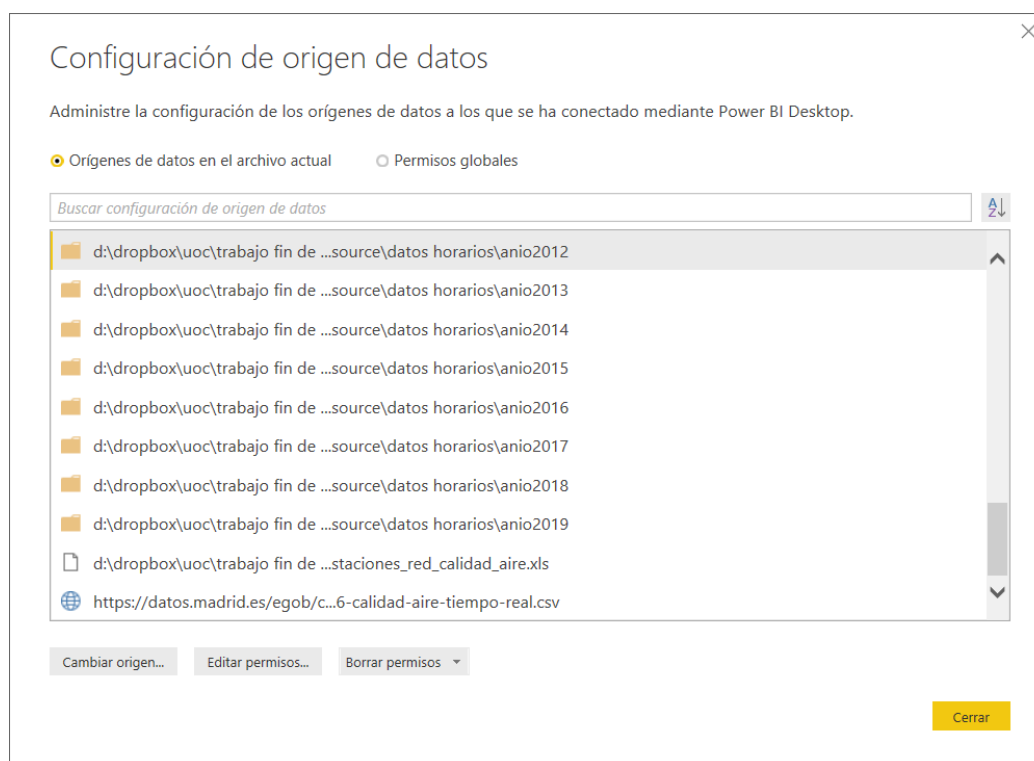
10.1 Conjunto de datos

Debido a la multitud de archivos en local descargados del repositorio, se ha tenido que importar a powerbi de varias formas: mediante supervisión de carpetas (para archivos distribuidos en meses), mediante txt o csv para archivos anuales y mediante link web (para la actualización en tiempo real).

Estos son todos los orígenes de datos importados:

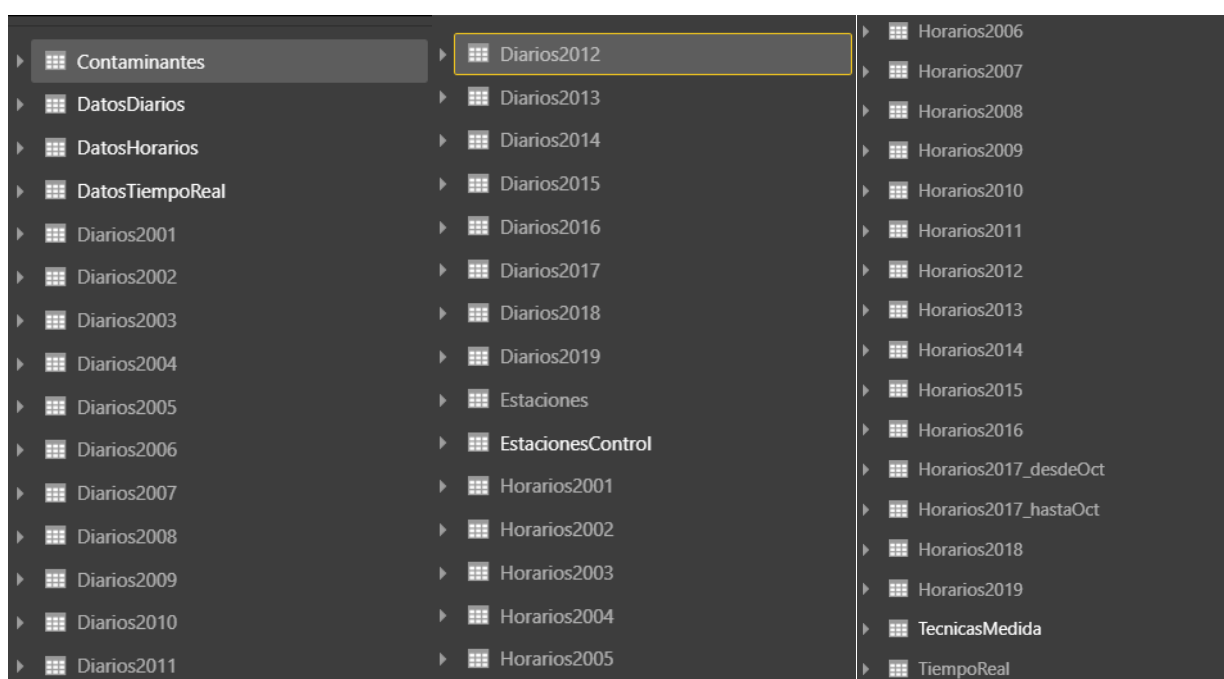






10.2 Tablas

Cada origen de datos es una tabla en nuestro modelo, pero para luego mostrar los datos se han tenido que unificarlas todas ellas en una tabla global, por temas de filtros. Se tienen 6 tablas principales y luego 41 auxiliares.



10.3 Procesos ETL

Los procesos ETL que se tienen en el actual proyecto son diversos y específicos. Se podrían agrupar por:

Procesos para orígenes de datos Diarios

Dentro de estos procesos se tienen dos procesos diferenciados, antes de octubre de 2017 y después de octubre 2017, porque hubo un cambio en la representación de los datos.

El proceso ETL para ambos sería:

Antes de octubre de 2017:

```
let
    Origen = Folder.Files("D:\Dropbox\UOC\Trabajo Fin de Grado\DataSource\Datos Horarios\Anio2017"),
    #"Filas filtradas" = Table.SelectRows(Origen, each ([Name] <> "dic17.csv" and [Name] <>
    "nov17.csv" and [Name] <> "oct17.csv")),
    #"Archivos ocultos filtrados1" = Table.SelectRows(#"Filas filtradas", each
    [Attributes]?[Hidden]? <> true),
    #"Invocar función personalizada1" = Table.AddColumn(#"Archivos ocultos filtrados1", "Transformar
    archivo de Anio2017", each #"Transformar archivo de Anio2017"([Content])),
    #"Columnas con nombre cambiado1" = Table.RenameColumns(#"Invocar función personalizada1",
    {"Name", "Source.Name"}),
    #"Otras columnas quitadas1" = Table.SelectColumns(#"Columnas con nombre cambiado1",
    {"Source.Name", "Transformar archivo de Anio2017"}),
    #"Columna de tabla expandida1" = Table.ExpandTableColumn(#"Otras columnas quitadas1",
    "Transformar archivo de Anio2017", Table.ColumnNames(#"Transformar archivo de Anio2017"("#"Archivo de
    ejemplo (2)")),
    #"Tipo cambiado" = Table.TransformColumnTypes(#"Columna de tabla expandida1",{"Source.Name",
    type text}, {"Column1", type text}),
    #"Dividir columna por posición" = Table.SplitColumn(#"Tipo cambiado", "Column1",
    Splitter.SplitTextByPositions({0, 2}, false), {"Column1.1", "Column1.2"}),
    #"Tipo cambiado1" = Table.TransformColumnTypes(#"Dividir columna por posición",{"Column1.1",
    Int64.Type}, {"Column1.2", type text}),
    #"Dividir columna por posición1" = Table.SplitColumn(#"Tipo cambiado1", "Column1.2",
    Splitter.SplitTextByPositions({0, 3}, false), {"Column1.2.1", "Column1.2.2"}),
    #"Tipo cambiado2" = Table.TransformColumnTypes(#"Dividir columna por posición1",{"Column1.2.1",
    Int64.Type}, {"Column1.2.2", type text}),
    #"Dividir columna por posición2" = Table.SplitColumn(#"Tipo cambiado2", "Column1.2.2",
    Splitter.SplitTextByPositions({0, 3}, false), {"Column1.2.2.1", "Column1.2.2.2"}),
    #"Tipo cambiado3" = Table.TransformColumnTypes(#"Dividir columna por
    posición2",{"Column1.2.2.1", Int64.Type}, {"Column1.2.2.2", type text}),
    #"Dividir columna por posición3" = Table.SplitColumn(#"Tipo cambiado3", "Column1.2.2.2",
    Splitter.SplitTextByPositions({0, 2}, false), {"Column1.2.2.2.1", "Column1.2.2.2.2"}),
    #"Tipo cambiado4" = Table.TransformColumnTypes(#"Dividir columna por
    posición3",{"Column1.2.2.2.1", Int64.Type}, {"Column1.2.2.2.2", type text}),
```

```

#"Dividir columna por posición4" = Table.SplitColumn("#Tipo cambiado4", "Column1.2.2.2.2",
Splitter.SplitTextByPositions({0, 2}, false), {"Column1.2.2.2.2.1", "Column1.2.2.2.2.2"}),

#"Tipo cambiado5" = Table.TransformColumnTypes("#Dividir columna por
posición4",{"Column1.2.2.2.2.1", Int64.Type}, {"Column1.2.2.2.2.2", type text}),

#"Dividir columna por posición5" = Table.SplitColumn("#Tipo cambiado5", "Column1.2.2.2.2.2",
Splitter.SplitTextByPositions({0, 2}, false), {"Column1.2.2.2.2.2.1", "Column1.2.2.2.2.2.2"}),

#"Tipo cambiado6" = Table.TransformColumnTypes("#Dividir columna por
posición5",{"Column1.2.2.2.2.2.1", Int64.Type}, {"Column1.2.2.2.2.2.2", type text}),

#"Dividir columna por posición6" = Table.SplitColumn("#Tipo cambiado6", "Column1.2.2.2.2.2.2",
Splitter.SplitTextByPositions({0, 2}, false), {"Column1.2.2.2.2.2.2.1", "Column1.2.2.2.2.2.2.2"}),

#"Tipo cambiado7" = Table.TransformColumnTypes("#Dividir columna por
posición6",{"Column1.2.2.2.2.2.2.1", Int64.Type}, {"Column1.2.2.2.2.2.2.2", type text}),

#"Dividir columna por posición7" = Table.SplitColumn("#Tipo cambiado7", "Column1.2.2.2.2.2.2.2",
Splitter.SplitTextByPositions({0, 2}, false), {"Column1.2.2.2.2.2.2.2.1",
"Column1.2.2.2.2.2.2.2.2"}),

#"Tipo cambiado8" = Table.TransformColumnTypes("#Dividir columna por
posición7",{"Column1.2.2.2.2.2.2.2.1", Int64.Type}, {"Column1.2.2.2.2.2.2.2.2", type text}),

#"Dividir columna por posición8" = Table.SplitColumn("#Tipo cambiado8",
"Column1.2.2.2.2.2.2.2.2", Splitter.SplitTextByPositions({0, 2}, false),
{"Column1.2.2.2.2.2.2.2.2.1", "Column1.2.2.2.2.2.2.2.2"}),

#"Tipo cambiado9" = Table.TransformColumnTypes("#Dividir columna por
posición8",{"Column1.2.2.2.2.2.2.2.2.1", Int64.Type}, {"Column1.2.2.2.2.2.2.2.2.2", type text}),

#"Dividir columna por posición9" = Table.SplitColumn("#Tipo cambiado9",
"Column1.2.2.2.2.2.2.2.2.2", Splitter.SplitTextByRepeatedLengths(6), {"Column1.2.2.2.2.2.2.2.2.2.1",
"Column1.2.2.2.2.2.2.2.2.2.2", "Column1.2.2.2.2.2.2.2.2.2.3", "Column1.2.2.2.2.2.2.2.2.2.4",
"Column1.2.2.2.2.2.2.2.2.2.5", "Column1.2.2.2.2.2.2.2.2.2.6", "Column1.2.2.2.2.2.2.2.2.2.7",
"Column1.2.2.2.2.2.2.2.2.2.8", "Column1.2.2.2.2.2.2.2.2.2.9", "Column1.2.2.2.2.2.2.2.2.2.10",
"Column1.2.2.2.2.2.2.2.2.2.11", "Column1.2.2.2.2.2.2.2.2.2.12", "Column1.2.2.2.2.2.2.2.2.2.13",
"Column1.2.2.2.2.2.2.2.2.2.14", "Column1.2.2.2.2.2.2.2.2.2.15", "Column1.2.2.2.2.2.2.2.2.2.16",
"Column1.2.2.2.2.2.2.2.2.2.17", "Column1.2.2.2.2.2.2.2.2.2.18", "Column1.2.2.2.2.2.2.2.2.2.19",
"Column1.2.2.2.2.2.2.2.2.2.20", "Column1.2.2.2.2.2.2.2.2.2.21", "Column1.2.2.2.2.2.2.2.2.2.22",
"Column1.2.2.2.2.2.2.2.2.2.23", "Column1.2.2.2.2.2.2.2.2.2.24"}),

#"Tipo cambiado10" = Table.TransformColumnTypes("#Dividir columna por
posición9",{"Column1.2.2.2.2.2.2.2.2.2.1", type text}, {"Column1.2.2.2.2.2.2.2.2.2.2", type text},
{"Column1.2.2.2.2.2.2.2.2.2.3", type text}, {"Column1.2.2.2.2.2.2.2.2.2.4", type text},
{"Column1.2.2.2.2.2.2.2.2.2.5", type text}, {"Column1.2.2.2.2.2.2.2.2.2.6", type text},
{"Column1.2.2.2.2.2.2.2.2.2.7", type text}, {"Column1.2.2.2.2.2.2.2.2.2.8", type text},
{"Column1.2.2.2.2.2.2.2.2.2.9", type text}, {"Column1.2.2.2.2.2.2.2.2.2.10", type text},
{"Column1.2.2.2.2.2.2.2.2.2.11", type text}, {"Column1.2.2.2.2.2.2.2.2.2.12", type text},
{"Column1.2.2.2.2.2.2.2.2.2.13", type text}, {"Column1.2.2.2.2.2.2.2.2.2.14", type text},
{"Column1.2.2.2.2.2.2.2.2.2.15", type text}, {"Column1.2.2.2.2.2.2.2.2.2.16", type text},
{"Column1.2.2.2.2.2.2.2.2.2.17", type text}, {"Column1.2.2.2.2.2.2.2.2.2.18", type text},
{"Column1.2.2.2.2.2.2.2.2.2.19", type text}, {"Column1.2.2.2.2.2.2.2.2.2.20", type text},
{"Column1.2.2.2.2.2.2.2.2.2.21", type text}, {"Column1.2.2.2.2.2.2.2.2.2.22", type text},
{"Column1.2.2.2.2.2.2.2.2.2.23", type text}, {"Column1.2.2.2.2.2.2.2.2.2.24", type text}),

#"Columnas quitadas" = Table.RemoveColumns("#Tipo cambiado10",{"Column1.1", "Source.Name",
"Column1.2.1"}),

#"Columnas con nombre cambiado" = Table.RenameColumns("#Columnas quitadas",{"Column1.2.2.2.1",
"Contaminantes_Magnitud"}, {"Column1.2.2.1", "EstacionesControl_NumeroEstacion"}),

#"Columnas quitadas1" = Table.RemoveColumns("#Columnas con nombre
cambiado",{"Column1.2.2.2.1", "Column1.2.2.2.2.1"}),

```

```

    #"Prefijo agregado" = Table.TransformColumns("#Columnas quitadas1", {"Column1.2.2.2.2.2.1",
each "20" & Text.From(_, "es-ES"), type text})),

    #"Columnas combinadas" = Table.CombineColumns(Table.TransformColumnTypes("#Prefijo agregado",
{"Column1.2.2.2.2.2.1", type text}, {"Column1.2.2.2.2.2.2.1", type text},
{"Column1.2.2.2.2.2.2.2.1", type text}}, "es-ES"), {"Column1.2.2.2.2.2.1",
"Column1.2.2.2.2.2.2.1", "Column1.2.2.2.2.2.2.2.1"}, Combiner.CombineTextByDelimiter("/",
QuoteStyle.None), "Fecha"),

    #"Tipo cambiado11" = Table.TransformColumnTypes("#Columnas combinadas", {"Fecha", type date})),

    #"Columnas con nombre cambiado2" = Table.RenameColumns("#Tipo
cambiado11", {"Column1.2.2.2.2.2.2.2.1", "01:00"}, {"Column1.2.2.2.2.2.2.2.2", "02:00"},
{"Column1.2.2.2.2.2.2.2.2.3", "03:00"}, {"Column1.2.2.2.2.2.2.2.2.4", "04:00"},
{"Column1.2.2.2.2.2.2.2.2.5", "05:00"}, {"Column1.2.2.2.2.2.2.2.2.6", "06:00"},
{"Column1.2.2.2.2.2.2.2.2.7", "07:00"}, {"Column1.2.2.2.2.2.2.2.2.8", "08:00"},
{"Column1.2.2.2.2.2.2.2.2.9", "09:00"}, {"Column1.2.2.2.2.2.2.2.2.10", "10:00"},
{"Column1.2.2.2.2.2.2.2.2.11", "11:00"}, {"Column1.2.2.2.2.2.2.2.2.12", "12:00"},
{"Column1.2.2.2.2.2.2.2.2.13", "13:00"}, {"Column1.2.2.2.2.2.2.2.2.14", "14:00"},
{"Column1.2.2.2.2.2.2.2.2.15", "15:00"}, {"Column1.2.2.2.2.2.2.2.2.16", "16:00"},
{"Column1.2.2.2.2.2.2.2.2.17", "17:00"}, {"Column1.2.2.2.2.2.2.2.2.18", "18:00"},
{"Column1.2.2.2.2.2.2.2.2.19", "19:00"}, {"Column1.2.2.2.2.2.2.2.2.20", "20:00"},
{"Column1.2.2.2.2.2.2.2.2.21", "21:00"}, {"Column1.2.2.2.2.2.2.2.2.22", "22:00"},
{"Column1.2.2.2.2.2.2.2.2.23", "23:00"}, {"Column1.2.2.2.2.2.2.2.2.24", "00:00"})),

    #"Otras columnas con anulación de dinamización" = Table.UnpivotOtherColumns("#Columnas con
nombre cambiado2", {"EstacionesControl_NumeroEstacion", "Contaminantes_Magnitud", "Fecha"},
"Atributo", "Valor"),

    #"Columnas con nombre cambiado3" = Table.RenameColumns("#Otras columnas con anulación de
dinamización", {"Atributo", "Hora"}, {"Valor", "ValorMedido"})),

    #"Tipo cambiado12" = Table.TransformColumnTypes("#Columnas con nombre cambiado3", {"Hora", type
time})),

    #"Filas filtradas1" = Table.SelectRows("#Tipo cambiado12", each Text.EndsWith([ValorMedido],
"V")),

    #"Dividir columna por posición10" = Table.SplitColumn("#Filas filtradas1", "ValorMedido",
Splitter.SplitTextByPositions({0, 1}, true), {"ValorMedido.1", "ValorMedido.2"}),

    #"Tipo cambiado con configuración regional" = Table.TransformColumnTypes("#Dividir columna por
posición10", {"ValorMedido.1", type number}}, "es-CU"),

    #"Columnas quitadas2" = Table.RemoveColumns("#Tipo cambiado con configuración
regional", {"ValorMedido.2"}),

    #"Columnas con nombre cambiado4" = Table.RenameColumns("#Columnas quitadas2", {"ValorMedido.1",
"ValorMedido"})

in

    #"Columnas con nombre cambiado4"

```

Después de octubre de 2017:

```

let

    Origen = Folder.Files("D:\Dropbox\UOC\Trabajo Fin de Grado\DataSource\Datos Horarios\Anio2017"),

    #"Filas filtradas" = Table.SelectRows(Origen, each ([Extension] = ".csv")),

    #"Archivos ocultos filtrados1" = Table.SelectRows("#Filas filtradas", each
[Attributes]?[Hidden]? <> true),

```

```

#"Invocar función personalizada1" = Table.AddColumn("#Archivos ocultos filtrados1", "Transformar
archivo de horarios2019_desdeOct", each #"Transformar archivo de horarios2019_desdeOct"([Content])),

#"Columnas con nombre cambiado1" = Table.RenameColumns("#Invocar función personalizada1",
{"Name", "Source.Name"}),

#"Otras columnas quitadas1" = Table.SelectColumns("#Columnas con nombre cambiado1",
{"Source.Name", "Transformar archivo de horarios2019_desdeOct"}),

#"Columna de tabla expandida1" = Table.ExpandTableColumn("#Otras columnas quitadas1",
"Transformar archivo de horarios2019_desdeOct", Table.ColumnNames("#Transformar archivo de
horarios2019_desdeOct"("#Archivo de ejemplo (3)")),

#"Columnas quitadas" = Table.RemoveColumns("#Columna de tabla expandida1",{"PROVINCIA", "MUNICIPIO",
"PUNTO_MUESTREO"}),

#"Errores quitados" = Table.RemoveRowsWithErrors("#Columnas quitadas", {"ESTACION"}),

#"Columnas combinadas" = Table.CombineColumns(Table.TransformColumnTypes("#Errores quitados",
{{"H01", type text}}, "es-ES"),{"H01", "V01"},Combiner.CombineTextByDelimiter(", ",
QuoteStyle.None),"Combinada"),

#"Columnas combinadas1" = Table.CombineColumns(Table.TransformColumnTypes("#Columnas
combinadas", {{"H02", type text}}, "es-ES"),{"H02", "V02"},Combiner.CombineTextByDelimiter(", ",
QuoteStyle.None),"Combinada.1"),

#"Columnas combinadas2" = Table.CombineColumns(Table.TransformColumnTypes("#Columnas
combinadas1", {{"H03", type text}}, "es-ES"),{"H03", "V03"},Combiner.CombineTextByDelimiter(", ",
QuoteStyle.None),"Combinada.2"),

#"Columnas combinadas3" = Table.CombineColumns(Table.TransformColumnTypes("#Columnas
combinadas2", {{"H04", type text}}, "es-ES"),{"H04", "V04"},Combiner.CombineTextByDelimiter(", ",
QuoteStyle.None),"Combinada.3"),

#"Columnas combinadas4" = Table.CombineColumns(Table.TransformColumnTypes("#Columnas
combinadas3", {{"H05", type text}}, "es-ES"),{"H05", "V05"},Combiner.CombineTextByDelimiter(", ",
QuoteStyle.None),"Combinada.4"),

#"Columnas combinadas5" = Table.CombineColumns(Table.TransformColumnTypes("#Columnas
combinadas4", {{"H06", type text}}, "es-ES"),{"H06", "V06"},Combiner.CombineTextByDelimiter(", ",
QuoteStyle.None),"Combinada.5"),

#"Columnas combinadas6" = Table.CombineColumns(Table.TransformColumnTypes("#Columnas
combinadas5", {{"H07", type text}}, "es-ES"),{"H07", "V07"},Combiner.CombineTextByDelimiter(", ",
QuoteStyle.None),"Combinada.6"),

#"Columnas combinadas7" = Table.CombineColumns(Table.TransformColumnTypes("#Columnas
combinadas6", {{"H08", type text}}, "es-ES"),{"H08", "V08"},Combiner.CombineTextByDelimiter(", ",
QuoteStyle.None),"Combinada.7"),

#"Columnas combinadas8" = Table.CombineColumns(Table.TransformColumnTypes("#Columnas
combinadas7", {{"H09", type text}}, "es-ES"),{"H09", "V09"},Combiner.CombineTextByDelimiter(", ",
QuoteStyle.None),"Combinada.8"),

#"Columnas combinadas9" = Table.CombineColumns(Table.TransformColumnTypes("#Columnas
combinadas8", {{"H10", type text}}, "es-ES"),{"H10", "V10"},Combiner.CombineTextByDelimiter(", ",
QuoteStyle.None),"Combinada.9"),

#"Columnas combinadas10" = Table.CombineColumns(Table.TransformColumnTypes("#Columnas
combinadas9", {{"H11", type text}}, "es-ES"),{"H11", "V11"},Combiner.CombineTextByDelimiter(", ",
QuoteStyle.None),"Combinada.10"),

#"Columnas combinadas11" = Table.CombineColumns(Table.TransformColumnTypes("#Columnas
combinadas10", {{"H12", type text}}, "es-ES"),{"H12", "V12"},Combiner.CombineTextByDelimiter(", ",
QuoteStyle.None),"Combinada.11"),

```

```

#"Columnas combinadas12" = Table.CombineColumns(Table.TransformColumnTypes("#Columnas
combinadas11", {"H13", type text}}, "es-ES"), {"H13", "V13"}, Combiner.CombineTextByDelimiter(",",
QuoteStyle.None), "Combinada.12"),

#"Columnas combinadas13" = Table.CombineColumns(Table.TransformColumnTypes("#Columnas
combinadas12", {"H14", type text}}, "es-ES"), {"H14", "V14"}, Combiner.CombineTextByDelimiter(",",
QuoteStyle.None), "Combinada.13"),

#"Columnas combinadas14" = Table.CombineColumns(Table.TransformColumnTypes("#Columnas
combinadas13", {"H15", type text}}, "es-ES"), {"H15", "V15"}, Combiner.CombineTextByDelimiter(",",
QuoteStyle.None), "Combinada.14"),

#"Columnas combinadas15" = Table.CombineColumns(Table.TransformColumnTypes("#Columnas
combinadas14", {"H16", type text}}, "es-ES"), {"H16", "V16"}, Combiner.CombineTextByDelimiter(",",
QuoteStyle.None), "Combinada.15"),

#"Columnas combinadas16" = Table.CombineColumns(Table.TransformColumnTypes("#Columnas
combinadas15", {"H17", type text}}, "es-ES"), {"H17", "V17"}, Combiner.CombineTextByDelimiter(",",
QuoteStyle.None), "Combinada.16"),

#"Columnas combinadas17" = Table.CombineColumns(Table.TransformColumnTypes("#Columnas
combinadas16", {"H18", type text}}, "es-ES"), {"H18", "V18"}, Combiner.CombineTextByDelimiter(",",
QuoteStyle.None), "Combinada.17"),

#"Columnas combinadas18" = Table.CombineColumns(Table.TransformColumnTypes("#Columnas
combinadas17", {"H19", type text}}, "es-ES"), {"H19", "V19"}, Combiner.CombineTextByDelimiter(",",
QuoteStyle.None), "Combinada.18"),

#"Columnas combinadas19" = Table.CombineColumns(Table.TransformColumnTypes("#Columnas
combinadas18", {"H20", type text}}, "es-ES"), {"H20", "V20"}, Combiner.CombineTextByDelimiter(",",
QuoteStyle.None), "Combinada.19"),

#"Columnas combinadas20" = Table.CombineColumns(Table.TransformColumnTypes("#Columnas
combinadas19", {"H21", type text}}, "es-ES"), {"H21", "V21"}, Combiner.CombineTextByDelimiter(",",
QuoteStyle.None), "Combinada.20"),

#"Columnas combinadas21" = Table.CombineColumns(Table.TransformColumnTypes("#Columnas
combinadas20", {"H22", type text}}, "es-ES"), {"H22", "V22"}, Combiner.CombineTextByDelimiter(",",
QuoteStyle.None), "Combinada.21"),

#"Columnas combinadas22" = Table.CombineColumns(Table.TransformColumnTypes("#Columnas
combinadas21", {"H23", type text}}, "es-ES"), {"H23", "V23"}, Combiner.CombineTextByDelimiter(",",
QuoteStyle.None), "Combinada.22"),

#"Columnas combinadas23" = Table.CombineColumns(Table.TransformColumnTypes("#Columnas
combinadas22", {"H24", type text}}, "es-ES"), {"H24", "V24"}, Combiner.CombineTextByDelimiter(",",
QuoteStyle.None), "Combinada.23"),

#"Columnas con nombre cambiado" = Table.RenameColumns("#Columnas combinadas23", {"Combinada",
"01:00"}, {"Combinada.1", "02:00"}, {"Combinada.2", "03:00"}, {"Combinada.3", "04:00"},
{"Combinada.4", "05:00"}, {"Combinada.5", "06:00"}, {"Combinada.6", "07:00"}, {"Combinada.7",
"08:00"}, {"Combinada.8", "09:00"}, {"Combinada.9", "10:00"}, {"Combinada.10", "11:00"},
{"Combinada.11", "12:00"}, {"Combinada.12", "13:00"}, {"Combinada.13", "14:00"}, {"Combinada.14",
"15:00"}, {"Combinada.15", "16:00"}, {"Combinada.16", "17:00"}, {"Combinada.17", "18:00"},
{"Combinada.18", "19:00"}, {"Combinada.19", "20:00"}, {"Combinada.20", "21:00"}, {"Combinada.21",
"22:00"}, {"Combinada.22", "23:00"}, {"Combinada.23", "00:00"})),

#"Otras columnas con anulación de dinamización" = Table.UnpivotOtherColumns("#Columnas con
nombre cambiado", {"ESTACION", "MAGNITUD", "ANO", "MES", "DIA"}, "Atributo", "Valor"),

#"Columnas con nombre cambiado2" = Table.RenameColumns("#Otras columnas con anulación de
dinamización", {"Atributo", "Hora"}),

#"Filas filtradas1" = Table.SelectRows("#Columnas con nombre cambiado2", each
Text.EndsWith([Valor], "V")),

```

```

    # "Texto extraído antes del delimitador" = Table.TransformColumns("#Filas filtradas1", {{"Valor",
each Text.BeforeDelimiter(_, ",", type text}}),

    # "Columnas combinadas24" = Table.CombineColumns("#Texto extraído antes del delimitador",{"ANO",
"MES", "DIA"},Combiner.CombineTextByDelimiter("/", QuoteStyle.None),"Fecha"),

    # "Tipo cambiado" = Table.TransformColumnTypes("#Columnas combinadas24",{"Hora", type time},
{"Fecha", type date}},

    # "Tipo cambiado con configuración regional" = Table.TransformColumnTypes("#Tipo cambiado",
{{"Valor", type number}}, "es-CU"),

    # "Tipo cambiado1" = Table.TransformColumnTypes("#Tipo cambiado con configuración
regional",{"ESTACION", Int64.Type}, {"MAGNITUD", Int64.Type}},

    # "Columnas con nombre cambiado3" = Table.RenameColumns("#Tipo cambiado1",{"MAGNITUD",
"Contaminantes_Magnitud"}, {"ESTACION", "EstacionesControl_NumeroEstacion"}, {"Valor",
"ValorMedido"}},

    # "Columnas reordenadas" = Table.ReorderColumns("#Columnas con nombre
cambiado3",{"Contaminantes_Magnitud", "EstacionesControl_NumeroEstacion", "Fecha", "Hora",
"ValorMedido"})

in

    # "Columnas reordenadas"

```

Procesos para orígenes de datos Horarios:

```

let

    Origen = Csv.Document(File.Contents("D:\Dropbox\UOC\Trabajo Fin de Grado\DataSource\Datos
Diarios\Anio2019\datos2019.csv"),[Delimiter=";", Columns=69, Encoding=1252,
QuoteStyle=QuoteStyle.None]),

    # "Encabezados promovidos" = Table.PromoteHeaders(Origen, [PromoteAllScalars=true]),

    # "Columnas quitadas" = Table.RemoveColumns("#Encabezados promovidos",{"PUNTO_MUESTREO",
"MUNICIPIO", "PROVINCIA"}),

    # "Errores quitados" = Table.RemoveRowsWithErrors("#Columnas quitadas", {"ESTACION"}),

    # "Columnas combinadas" = Table.CombineColumns(Table.TransformColumnTypes("#Errores quitados",
{{"D01", type text}}, "es-ES"),{"D01", "V01"},Combiner.CombineTextByDelimiter(",",",
QuoteStyle.None),"Combinada"),

    # "Columnas combinadas1" = Table.CombineColumns(Table.TransformColumnTypes("#Columnas
combinadas", {"D02", type text}}, "es-ES"),{"D02", "V02"},Combiner.CombineTextByDelimiter(",",",
QuoteStyle.None),"Combinada.1"),

    # "Columnas combinadas2" = Table.CombineColumns(Table.TransformColumnTypes("#Columnas
combinadas1", {"D03", type text}}, "es-ES"),{"D03", "V03"},Combiner.CombineTextByDelimiter(",",",
QuoteStyle.None),"Combinada.2"),

    # "Columnas combinadas3" = Table.CombineColumns(Table.TransformColumnTypes("#Columnas
combinadas2", {"D04", type text}}, "es-ES"),{"D04", "V04"},Combiner.CombineTextByDelimiter(",",",
QuoteStyle.None),"Combinada.3"),

    # "Columnas combinadas4" = Table.CombineColumns(Table.TransformColumnTypes("#Columnas
combinadas3", {"D05", type text}}, "es-ES"),{"D05", "V05"},Combiner.CombineTextByDelimiter(",",",
QuoteStyle.None),"Combinada.4"),

    # "Columnas combinadas5" = Table.CombineColumns(Table.TransformColumnTypes("#Columnas
combinadas4", {"D06", type text}}, "es-ES"),{"D06", "V06"},Combiner.CombineTextByDelimiter(",",",
QuoteStyle.None),"Combinada.5"),

```

```
#"Columnas combinadas6" = Table.CombineColumns(Table.TransformColumnTypes("#Columnas
combinadas5", {"D07", type text}}, "es-ES"), {"D07", "V07"}, Combiner.CombineTextByDelimiter(", ",
QuoteStyle.None), "Combinada.6"),

#"Columnas combinadas7" = Table.CombineColumns(Table.TransformColumnTypes("#Columnas
combinadas6", {"D08", type text}}, "es-ES"), {"D08", "V08"}, Combiner.CombineTextByDelimiter(", ",
QuoteStyle.None), "Combinada.7"),

#"Columnas combinadas8" = Table.CombineColumns(Table.TransformColumnTypes("#Columnas
combinadas7", {"D09", type text}}, "es-ES"), {"D09", "V09"}, Combiner.CombineTextByDelimiter(", ",
QuoteStyle.None), "Combinada.8"),

#"Columnas combinadas9" = Table.CombineColumns(Table.TransformColumnTypes("#Columnas
combinadas8", {"D10", type text}}, "es-ES"), {"D10", "V10"}, Combiner.CombineTextByDelimiter(", ",
QuoteStyle.None), "Combinada.9"),

#"Columnas combinadas10" = Table.CombineColumns(Table.TransformColumnTypes("#Columnas
combinadas9", {"D11", type text}}, "es-ES"), {"D11", "V11"}, Combiner.CombineTextByDelimiter(", ",
QuoteStyle.None), "Combinada.10"),

#"Columnas combinadas11" = Table.CombineColumns(Table.TransformColumnTypes("#Columnas
combinadas10", {"D12", type text}}, "es-ES"), {"D12", "V12"}, Combiner.CombineTextByDelimiter(", ",
QuoteStyle.None), "Combinada.11"),

#"Columnas combinadas12" = Table.CombineColumns(Table.TransformColumnTypes("#Columnas
combinadas11", {"D13", type text}}, "es-ES"), {"D13", "V13"}, Combiner.CombineTextByDelimiter(", ",
QuoteStyle.None), "Combinada.12"),

#"Columnas combinadas13" = Table.CombineColumns(Table.TransformColumnTypes("#Columnas
combinadas12", {"D14", type text}}, "es-ES"), {"D14", "V14"}, Combiner.CombineTextByDelimiter(", ",
QuoteStyle.None), "Combinada.13"),

#"Columnas combinadas14" = Table.CombineColumns(Table.TransformColumnTypes("#Columnas
combinadas13", {"D15", type text}}, "es-ES"), {"D15", "V15"}, Combiner.CombineTextByDelimiter(", ",
QuoteStyle.None), "Combinada.14"),

#"Columnas combinadas15" = Table.CombineColumns(Table.TransformColumnTypes("#Columnas
combinadas14", {"D16", type text}}, "es-ES"), {"D16", "V16"}, Combiner.CombineTextByDelimiter(", ",
QuoteStyle.None), "Combinada.15"),

#"Columnas combinadas16" = Table.CombineColumns(Table.TransformColumnTypes("#Columnas
combinadas15", {"D17", type text}}, "es-ES"), {"D17", "V17"}, Combiner.CombineTextByDelimiter(", ",
QuoteStyle.None), "Combinada.16"),

#"Columnas combinadas17" = Table.CombineColumns(Table.TransformColumnTypes("#Columnas
combinadas16", {"D18", type text}}, "es-ES"), {"D18", "V18"}, Combiner.CombineTextByDelimiter(", ",
QuoteStyle.None), "Combinada.17"),

#"Columnas combinadas18" = Table.CombineColumns(Table.TransformColumnTypes("#Columnas
combinadas17", {"D19", type text}}, "es-ES"), {"D19", "V19"}, Combiner.CombineTextByDelimiter(", ",
QuoteStyle.None), "Combinada.18"),

#"Columnas combinadas19" = Table.CombineColumns(Table.TransformColumnTypes("#Columnas
combinadas18", {"D20", type text}}, "es-ES"), {"D20", "V20"}, Combiner.CombineTextByDelimiter(", ",
QuoteStyle.None), "Combinada.19"),

#"Columnas combinadas20" = Table.CombineColumns(Table.TransformColumnTypes("#Columnas
combinadas19", {"D21", type text}}, "es-ES"), {"D21", "V21"}, Combiner.CombineTextByDelimiter(", ",
QuoteStyle.None), "Combinada.20"),

#"Columnas combinadas21" = Table.CombineColumns(Table.TransformColumnTypes("#Columnas
combinadas20", {"D22", type text}}, "es-ES"), {"D22", "V22"}, Combiner.CombineTextByDelimiter(", ",
QuoteStyle.None), "Combinada.21"),
```



```

#"Columnas combinadas22" = Table.CombineColumns(Table.TransformColumnTypes("#Columnas
combinadas21", {"D23", type text}}, "es-ES"), {"D23", "V23"}, Combiner.CombineTextByDelimiter(",",
QuoteStyle.None), "Combinada.22"),

#"Columnas combinadas23" = Table.CombineColumns(Table.TransformColumnTypes("#Columnas
combinadas22", {"D24", type text}}, "es-ES"), {"D24", "V24"}, Combiner.CombineTextByDelimiter(",",
QuoteStyle.None), "Combinada.23"),

#"Columnas combinadas24" = Table.CombineColumns(Table.TransformColumnTypes("#Columnas
combinadas23", {"D25", type text}}, "es-ES"), {"D25", "V25"}, Combiner.CombineTextByDelimiter(",",
QuoteStyle.None), "Combinada.24"),

#"Columnas combinadas25" = Table.CombineColumns(Table.TransformColumnTypes("#Columnas
combinadas24", {"D26", type text}}, "es-ES"), {"D26", "V26"}, Combiner.CombineTextByDelimiter(",",
QuoteStyle.None), "Combinada.25"),

#"Columnas combinadas26" = Table.CombineColumns(Table.TransformColumnTypes("#Columnas
combinadas25", {"D27", type text}}, "es-ES"), {"D27", "V27"}, Combiner.CombineTextByDelimiter(",",
QuoteStyle.None), "Combinada.26"),

#"Columnas combinadas27" = Table.CombineColumns(Table.TransformColumnTypes("#Columnas
combinadas26", {"D28", type text}}, "es-ES"), {"D28", "V28"}, Combiner.CombineTextByDelimiter(",",
QuoteStyle.None), "Combinada.27"),

#"Columnas combinadas28" = Table.CombineColumns(Table.TransformColumnTypes("#Columnas
combinadas27", {"D29", type text}}, "es-ES"), {"D29", "V29"}, Combiner.CombineTextByDelimiter(",",
QuoteStyle.None), "Combinada.28"),

#"Columnas combinadas29" = Table.CombineColumns(Table.TransformColumnTypes("#Columnas
combinadas28", {"D30", type text}}, "es-ES"), {"D30", "V30"}, Combiner.CombineTextByDelimiter(",",
QuoteStyle.None), "Combinada.29"),

#"Columnas combinadas30" = Table.CombineColumns(Table.TransformColumnTypes("#Columnas
combinadas29", {"D31", type text}}, "es-ES"), {"D31", "V31"}, Combiner.CombineTextByDelimiter(",",
QuoteStyle.None), "Combinada.30"),

#"Columnas con nombre cambiado" = Table.RenameColumns("#Columnas combinadas30", {"Combinada",
"1"}, {"Combinada.1", "2"}, {"Combinada.2", "3"}, {"Combinada.3", "4"}, {"Combinada.4", "5"},
{"Combinada.5", "6"}, {"Combinada.6", "7"}, {"Combinada.7", "8"}, {"Combinada.8", "9"},
{"Combinada.9", "10"}, {"Combinada.10", "11"}, {"Combinada.11", "12"}, {"Combinada.12", "13"},
{"Combinada.13", "14"}, {"Combinada.14", "15"}, {"Combinada.15", "16"}, {"Combinada.16", "17"},
{"Combinada.17", "18"}, {"Combinada.18", "19"}, {"Combinada.19", "20"}, {"Combinada.20", "21"},
{"Combinada.21", "22"}, {"Combinada.22", "23"}, {"Combinada.23", "24"}, {"Combinada.24", "25"},
{"Combinada.25", "26"}, {"Combinada.26", "27"}, {"Combinada.27", "28"}, {"Combinada.28", "29"},
{"Combinada.29", "30"}, {"Combinada.30", "31"})),

#"Otras columnas con anulación de dinamización" = Table.UnpivotOtherColumns("#Columnas con
nombre cambiado", {"ESTACION", "MAGNITUD", "ANO", "MES"}, "Atributo", "Valor"),

#"Columnas con nombre cambiado1" = Table.RenameColumns("#Otras columnas con anulación de
dinamización", {"Atributo", "Dia"}),

#"Filas filtradas" = Table.SelectRows("#Columnas con nombre cambiado1", each
Text.EndsWith([Valor], "V")),

#"Dividir columna por delimitador" = Table.SplitColumn("#Filas filtradas", "Valor",
Splitter.SplitTextByDelimiter(",", QuoteStyle.Csv), {"Valor.1", "Valor.2"}),

#"Tipo cambiado con configuración regional" = Table.TransformColumnTypes("#Dividir columna por
delimitador", {"Valor.1", type number}, "es-CU"),

#"Columnas con nombre cambiado2" = Table.RenameColumns("#Tipo cambiado con configuración
regional", {"Valor.1", "Valor"}),

#"Columnas quitadas1" = Table.RemoveColumns("#Columnas con nombre cambiado2", {"Valor.2"}),

```

```

    #"Columnas con nombre cambiado3" = Table.RenameColumns("#Columnas quitadas1",{{"ANO", "Año"}, {"MES", "Mes"}, {"MAGNITUD", "Magnitud"}, {"ESTACION", "Estacion"}}),

    #"Columnas combinadas31" = Table.CombineColumns("#Columnas con nombre cambiado3",{{"Año", "Mes", "Dia"}},Combiner.CombineTextByDelimiter("/", QuoteStyle.None),"Fecha"),

    #"Tipo cambiado" = Table.TransformColumnTypes("#Columnas combinadas31",{{"Fecha", type date}}),

    #"Columnas reordenadas" = Table.ReorderColumns("#Tipo cambiado",{{"Magnitud", "Estacion", "Fecha", "Valor"}},

    #"Tipo cambiado1" = Table.TransformColumnTypes("#Columnas reordenadas",{{"Magnitud", Int64.Type}, {"Estacion", Int64.Type}})
in

    #"Tipo cambiado1"

```

Procesos para origen de dato en TiempoReal:

```

let

    Origen = Csv.Document(Web.Contents("https://datos.madrid.es/egob/catalogo/212531-10515086-calidad-aire-tiempo-real.csv"),[Delimiter=";", Columns=56, Encoding=1252, QuoteStyle=QuoteStyle.None]),

    #"Encabezados promovidos" = Table.PromoteHeaders(Origen, [PromoteAllScalars=true]),

    #"Columnas quitadas" = Table.RemoveColumns("#Encabezados promovidos",{{"PROVINCIA", "MUNICIPIO", "PUNTO_MUESTREO"}},

    #"Errores quitados" = Table.RemoveRowsWithErrors("#Columnas quitadas", {"ESTACION"}),

    #"Columnas combinadas" = Table.CombineColumns(Table.TransformColumnTypes("#Errores quitados",{{"H01", type text}}, "es-ES"),{"H01", "V01"},Combiner.CombineTextByDelimiter(", ", QuoteStyle.None),"Combinada"),

    #"Columnas combinadas1" = Table.CombineColumns(Table.TransformColumnTypes("#Columnas combinadas", {{"H02", type text}}, "es-ES"),{"H02", "V02"},Combiner.CombineTextByDelimiter(", ", QuoteStyle.None),"Combinada.1"),

    #"Columnas combinadas2" = Table.CombineColumns(Table.TransformColumnTypes("#Columnas combinadas1", {{"H03", type text}}, "es-ES"),{"H03", "V03"},Combiner.CombineTextByDelimiter(", ", QuoteStyle.None),"Combinada.2"),

    #"Columnas combinadas3" = Table.CombineColumns(Table.TransformColumnTypes("#Columnas combinadas2", {{"H04", type text}}, "es-ES"),{"H04", "V04"},Combiner.CombineTextByDelimiter(", ", QuoteStyle.None),"Combinada.3"),

    #"Columnas combinadas4" = Table.CombineColumns(Table.TransformColumnTypes("#Columnas combinadas3", {{"H05", type text}}, "es-ES"),{"H05", "V05"},Combiner.CombineTextByDelimiter(", ", QuoteStyle.None),"Combinada.4"),

    #"Columnas combinadas5" = Table.CombineColumns(Table.TransformColumnTypes("#Columnas combinadas4", {{"H06", type text}}, "es-ES"),{"H06", "V06"},Combiner.CombineTextByDelimiter(", ", QuoteStyle.None),"Combinada.5"),

    #"Columnas combinadas6" = Table.CombineColumns(Table.TransformColumnTypes("#Columnas combinadas5", {{"H07", type text}}, "es-ES"),{"H07", "V07"},Combiner.CombineTextByDelimiter(", ", QuoteStyle.None),"Combinada.6"),

    #"Columnas combinadas7" = Table.CombineColumns(Table.TransformColumnTypes("#Columnas combinadas6", {{"H08", type text}}, "es-ES"),{"H08", "V08"},Combiner.CombineTextByDelimiter(", ", QuoteStyle.None),"Combinada.7"),

```

```
#"Columnas combinadas8" = Table.CombineColumns(Table.TransformColumnTypes("#Columnas
combinadas7", {"H09", type text}}, "es-ES"), {"H09", "V09"}, Combiner.CombineTextByDelimiter(", ",
QuoteStyle.None), "Combinada.8"),

#"Columnas combinadas9" = Table.CombineColumns(Table.TransformColumnTypes("#Columnas
combinadas8", {"H10", type text}}, "es-ES"), {"H10", "V10"}, Combiner.CombineTextByDelimiter(", ",
QuoteStyle.None), "Combinada.9"),

#"Columnas combinadas10" = Table.CombineColumns(Table.TransformColumnTypes("#Columnas
combinadas9", {"H11", type text}}, "es-ES"), {"H11", "V11"}, Combiner.CombineTextByDelimiter(", ",
QuoteStyle.None), "Combinada.10"),

#"Columnas combinadas11" = Table.CombineColumns(Table.TransformColumnTypes("#Columnas
combinadas10", {"H12", type text}}, "es-ES"), {"H12", "V12"}, Combiner.CombineTextByDelimiter(", ",
QuoteStyle.None), "Combinada.11"),

#"Columnas combinadas12" = Table.CombineColumns(Table.TransformColumnTypes("#Columnas
combinadas11", {"H13", type text}}, "es-ES"), {"H13", "V13"}, Combiner.CombineTextByDelimiter(", ",
QuoteStyle.None), "Combinada.12"),

#"Columnas combinadas13" = Table.CombineColumns(Table.TransformColumnTypes("#Columnas
combinadas12", {"H14", type text}}, "es-ES"), {"H14", "V14"}, Combiner.CombineTextByDelimiter(", ",
QuoteStyle.None), "Combinada.13"),

#"Columnas combinadas14" = Table.CombineColumns(Table.TransformColumnTypes("#Columnas
combinadas13", {"H15", type text}}, "es-ES"), {"H15", "V15"}, Combiner.CombineTextByDelimiter(", ",
QuoteStyle.None), "Combinada.14"),

#"Columnas combinadas15" = Table.CombineColumns(Table.TransformColumnTypes("#Columnas
combinadas14", {"H16", type text}}, "es-ES"), {"H16", "V16"}, Combiner.CombineTextByDelimiter(", ",
QuoteStyle.None), "Combinada.15"),

#"Columnas combinadas16" = Table.CombineColumns(Table.TransformColumnTypes("#Columnas
combinadas15", {"H17", type text}}, "es-ES"), {"H17", "V17"}, Combiner.CombineTextByDelimiter(", ",
QuoteStyle.None), "Combinada.16"),

#"Columnas combinadas17" = Table.CombineColumns(Table.TransformColumnTypes("#Columnas
combinadas16", {"H18", type text}}, "es-ES"), {"H18", "V18"}, Combiner.CombineTextByDelimiter(", ",
QuoteStyle.None), "Combinada.17"),

#"Columnas combinadas18" = Table.CombineColumns(Table.TransformColumnTypes("#Columnas
combinadas17", {"H19", type text}}, "es-ES"), {"H19", "V19"}, Combiner.CombineTextByDelimiter(", ",
QuoteStyle.None), "Combinada.18"),

#"Columnas combinadas19" = Table.CombineColumns(Table.TransformColumnTypes("#Columnas
combinadas18", {"H20", type text}}, "es-ES"), {"H20", "V20"}, Combiner.CombineTextByDelimiter(", ",
QuoteStyle.None), "Combinada.19"),

#"Columnas combinadas20" = Table.CombineColumns(Table.TransformColumnTypes("#Columnas
combinadas19", {"H21", type text}}, "es-ES"), {"H21", "V21"}, Combiner.CombineTextByDelimiter(", ",
QuoteStyle.None), "Combinada.20"),

#"Columnas combinadas21" = Table.CombineColumns(Table.TransformColumnTypes("#Columnas
combinadas20", {"H22", type text}}, "es-ES"), {"H22", "V22"}, Combiner.CombineTextByDelimiter(", ",
QuoteStyle.None), "Combinada.21"),

#"Columnas combinadas22" = Table.CombineColumns(Table.TransformColumnTypes("#Columnas
combinadas21", {"H23", type text}}, "es-ES"), {"H23", "V23"}, Combiner.CombineTextByDelimiter(", ",
QuoteStyle.None), "Combinada.22"),

#"Columnas combinadas23" = Table.CombineColumns(Table.TransformColumnTypes("#Columnas
combinadas22", {"H24", type text}}, "es-ES"), {"H24", "V24"}, Combiner.CombineTextByDelimiter(", ",
QuoteStyle.None), "Combinada.23"),

#"Columnas con nombre cambiado" = Table.RenameColumns("#Columnas combinadas23", {"Combinada",
"01:00"}, {"Combinada.1", "02:00"}, {"Combinada.2", "03:00"}, {"Combinada.3", "04:00"},
```

```

{"Combinada.4", "05:00"}, {"Combinada.5", "06:00"}, {"Combinada.6", "07:00"}, {"Combinada.7",
"08:00"}, {"Combinada.8", "09:00"}, {"Combinada.9", "10:00"}, {"Combinada.10", "11:00"},
{"Combinada.11", "12:00"}, {"Combinada.12", "13:00"}, {"Combinada.13", "14:00"}, {"Combinada.14",
"15:00"}, {"Combinada.15", "16:00"}, {"Combinada.16", "17:00"}, {"Combinada.17", "18:00"},
{"Combinada.18", "19:00"}, {"Combinada.19", "20:00"}, {"Combinada.20", "21:00"}, {"Combinada.21",
"22:00"}, {"Combinada.22", "23:00"}, {"Combinada.23", "00:00"}},

#"Otras columnas con anulación de dinamización" = Table.UnpivotOtherColumns("#Columnas con
nombre cambiado", {"ESTACION", "MAGNITUD", "ANO", "MES", "DIA"}, "Atributo", "Valor"),

#"Columnas con nombre cambiado1" = Table.RenameColumns("#Otras columnas con anulación de
dinamización",{"Atributo", "Hora"}),

#"Filas filtradas" = Table.SelectRows("#Columnas con nombre cambiado1", each
Text.EndsWith([Valor], "V")),

#"Texto extraído antes del delimitador" = Table.TransformColumns("#Filas filtradas", {"Valor",
each Text.BeforeDelimiter(_, ","), type text}),

#"Columnas combinadas24" = Table.CombineColumns("#Texto extraído antes del delimitador",{"ANO",
"MES", "DIA"},Combiner.CombineTextByDelimiter("/", QuoteStyle.None),"Fecha"),

#"Tipo cambiado" = Table.TransformColumnTypes("#Columnas combinadas24",{"Hora", type time},
{"Fecha", type date}),

#"Tipo cambiado con configuración regional" = Table.TransformColumnTypes("#Tipo cambiado",
{"Valor", type number}, "es-CU"),

#"Tipo cambiado1" = Table.TransformColumnTypes("#Tipo cambiado con configuración
regional",{"ESTACION", Int64.Type}, {"MAGNITUD", Int64.Type}),

#"Columnas con nombre cambiado2" = Table.RenameColumns("#Tipo cambiado1",{"MAGNITUD",
"Contaminantes_Magnitud"}, {"ESTACION", "EstacionesControl_NumeroEstacion"}, {"Valor",
"ValorMedido"}),

#"Columnas reordenadas" = Table.ReorderColumns("#Columnas con nombre
cambiado2",{"Contaminantes_Magnitud", "EstacionesControl_NumeroEstacion", "Fecha", "Hora",
"ValorMedido"})

in

#"Columnas reordenadas"

```

Procesos para origen de datos Estaciones

```

let

Origen = Excel.Workbook(File.Contents("D:\Dropbox\UOC\Trabajo Fin de Grado\DataSource\Estaciones
de control\informacion_estaciones_red_calidad_aire.xls"), null, true),

Hoja1 = Origen[Name="Hoja1"][Data],

#"Filas superiores quitadas" = Table.Skip(Hoja1,3),

#"Encabezados promovidos" = Table.PromoteHeaders("#Filas superiores quitadas",
[PromoteAllScalars=true]),

#"Filas inferiores quitadas" = Table.RemoveLastN("#Encabezados promovidos",4),

#"Columnas quitadas" = Table.RemoveColumns("#Filas inferiores quitadas",{"NO2", "SO2", "CO",
"PM10", "PM2,5", "O3", "BTX", "HC", "UV", "VV", "DV", "TMP", "HR", "PRB", "RS", "LL"}),

#"Columnas con nombre cambiado" = Table.RenameColumns("#Columnas quitadas",{"NÚMERO",
"NumeroEstacion"}, {"ESTACIÓN", "Nombre"}, {"DIRECCIÓN", "Direccion"}, {"LONGITUD", "Longitud"},
{"LATITUD", "Latitud"}, {"ALTITUD", "Altitud"}, {"TIPO ESTACION *", "TipoEstacion"}),

```

```

    #"Valor reemplazado" = Table.ReplaceValue("#Columnas con nombre cambiado","UT","Urbana de
Tráfico",Replacer.ReplaceText,{"TipoEstacion"}),

    #"Valor reemplazado1" = Table.ReplaceValue("#Valor reemplazado","UF","Urbana de
Fondo",Replacer.ReplaceText,{"TipoEstacion"}),

    #"Valor reemplazado2" = Table.ReplaceValue("#Valor
reemplazado1","S","Suburbana",Replacer.ReplaceText,{"TipoEstacion"}),

    #"Tipo cambiado" = Table.TransformColumnTypes("#Valor reemplazado2",{"NumeroEstacion",
Int64.Type}, {"CodigoPostal", Int64.Type}, {"Altitud", Int64.Type})

in

    #"Tipo cambiado"

```

Procesos para origen de datos Contaminantes

```

let

    Origen =
Table.FromRows(Json.Document(Binary.Decompress(Binary.FromText("bZK9boMwEIBfxWJqpag1P0nIWCgSQ8GpmqEV
zUDATS0BloyRUF6jT9AH6JQ1G3ueqTZuSUK8WCfzFXfH+ZLEMI2J8Ui6fUtyCnIMHnbNB8Pi8gVZ4jwetvclsEVku8Z6khgzEUa0
Ogl+yja0ouLaR+IoBS9xR+HzMR4Tzrr9FvdGjP5LSAUqxR11NDIsjbiQ4TJlvPvJmiKtwXsDoZ0B624KjodSfoxEfc468140od40
4SD2yJUnu+i+ZIU1psdw06PpiBjt1KiQfUHMesKS1Va0aFSaFXo6h6aLHrIl50EqU5AXxDpI/mvASbEZwMALdGD/PpinLSkUF72+
6ThXzenELfwcfAnEOB24Zy3nyPmFJGc0E+vTMDFFtnla4Brcf0I2reitnIAfnm0UtJRq/7WsdI50NIhzb2ioMS8+65IRi9qxFHo
jzKsfwE=", BinaryEncoding.Base64), Compression.Deflate)), let _t = ((type text) meta
[Serialized.Text = true]) in type table [Magnitud = _t, Nombre = _t, Formula = _t, UnidadMedida =
_t, TecnicaMedida = _t]),

    #"Tipo cambiado" = Table.TransformColumnTypes(Origen,{"Magnitud", Int64.Type}, {"Nombre", type
text}, {"Formula", type text}, {"UnidadMedida", type text}, {"TecnicaMedida", Int64.Type})

in

    #"Tipo cambiado"

```

Procesos para origen de datos EstacionesControl

```

let

    Origen = Table.FromRows(Json.Document(Binary.Decompress(Binary.FromText("i44FAA==",
BinaryEncoding.Base64), Compression.Deflate)), let _t = ((type text) meta [Serialized.Text = true])
in type table [NumeroEstacion = _t, Nombre = _t, Direccion = _t, Longitud = _t, Latitud = _t,
Altitud = _t, TipoEstacion = _t]),

    #"Tipo cambiado" = Table.TransformColumnTypes(Origen,{"NumeroEstacion", Int64.Type}, {"Nombre",
type text}, {"Direccion", type text}, {"Longitud", type text}, {"Latitud", type text}, {"Altitud",
Int64.Type}, {"TipoEstacion", type text}),

    #"Consulta anexada" = Table.Combine({"Tipo cambiado", Estaciones})

in

    #"Consulta anexada"

```

Procesos para origen de datos TecnicasMedida

```

let

    Origen =
Table.FromRows(Json.Document(Binary.Decompress(Binary.FromText("Zc49DsIwDAXgq1SZ0yB+y4iQkBgYmKsMbkgr
oySwnIshd2LiCL0YrpoFsb7v6dltq1ZrVasrBRzB4PQJ1cNWzoEHpWvRveipi8QLZpcYXkjOp1JopHDP6CXLHgNGY4PBBTczXlwm
tiX+H9gepHNDw9SBgzCwtPm9i6FnYKbnwruj8JnJQ6KBoZ/eMP89QLRRaf0F", BinaryEncoding.Base64),

```

```

Compression.Deflate)), let _t = ((type text) meta [Serialized.Text = true]) in type table
[TecnicaMedida = _t, NombreTecnica = _t]),

    #"Tipo cambiado" = Table.TransformColumnTypes(Origen,{"TecnicaMedida", Int64.Type},
{"NombreTecnica", type text})

in

    #"Tipo cambiado"

```

Procesos para anexionar todas las tablas en una:

DatosDiarios:

```

let

    Origen = Table.FromRows(Json.Document(Binary.Decompress(Binary.FromText("i44FAA==",
BinaryEncoding.Base64), Compression.Deflate)), let _t = ((type text) meta [Serialized.Text = true])
in type table [Contaminantes_Magnitud = _t, EstacionesControl_NumeroEstacion = _t, Fecha = _t,
ValorMedido = _t, Columna1 = _t]),

    #"Tipo cambiado" = Table.TransformColumnTypes(Origen,{"Contaminantes_Magnitud", Int64.Type},
{"EstacionesControl_NumeroEstacion", Int64.Type}, {"Fecha", type date}, {"ValorMedido", type text},
{"Columna1", type text})),

    #"Columnas quitadas" = Table.RemoveColumns(#"Tipo cambiado",{"Columna1"}),

    #"Tipo cambiado1" = Table.TransformColumnTypes(#"Columnas quitadas",{"ValorMedido", type
number}, {"Contaminantes_Magnitud", Int64.Type}, {"EstacionesControl_NumeroEstacion", Int64.Type})),

    #"Tipo cambiado con configuración regional" = Table.TransformColumnTypes(#"Tipo cambiado1",
{"ValorMedido", type number}}, "es-CU"),

    #"Consulta anexada" = Table.Combine({"Tipo cambiado con configuración regional", Diarios2019,
Diarios2018, Diarios2017, Diarios2016, Diarios2015, Diarios2014, Diarios2013, Diarios2012,
Diarios2011, Diarios2010, Diarios2009, Diarios2008, Diarios2007, Diarios2006, Diarios2005,
Diarios2004, Diarios2003, Diarios2002, Diarios2001}),

    #"Columnas quitadas1" = Table.RemoveColumns(#"Consulta anexada",{"Contaminantes_Magnitud",
"EstacionesControl_NumeroEstacion"}),

    #"Columnas con nombre cambiado" = Table.RenameColumns(#"Columnas quitadas1",{"Estacion",
"EstacionesControl_NumeroEstacion"})),

    #"Columnas quitadas2" = Table.RemoveColumns(#"Columnas con nombre cambiado",{"ValorMedido"}),

    #"Columnas con nombre cambiado1" = Table.RenameColumns(#"Columnas quitadas2",{"Valor",
"ValorMedido"})),

    #"Columnas reordenadas" = Table.ReorderColumns(#"Columnas con nombre
cambiado1",{"EstacionesControl_NumeroEstacion", "Magnitud", "Fecha", "ValorMedido"})

in

    #"Columnas reordenadas"

```

DatosHorarios:

```

let

    Origen = Table.FromRows(Json.Document(Binary.Decompress(Binary.FromText("i44FAA==",
BinaryEncoding.Base64), Compression.Deflate)), let _t = ((type text) meta [Serialized.Text = true])
in type table [Contaminantes_Magnitud = _t, EstacionesControl_NumeroEstacion = _t, Fecha = _t, Hora
= _t, ValorMedido = _t]),

```

```
#"Tipo cambiado" = Table.TransformColumnTypes(Origen,{"Contaminantes_Magnitud", Int64.Type},  
{ "EstacionesControl_NumeroEstacion", Int64.Type}, {"Fecha", type date}, {"Hora", type time},  
{"ValorMedido", type number}}),
```

```
#"Consulta anexada" = Table.Combine({#"Tipo cambiado", Horarios2019, Horarios2018,  
Horarios2017_hastaOct, Horarios2017_desdeOct, Horarios2016, Horarios2015, Horarios2014,  
Horarios2013, Horarios2012, Horarios2011, Horarios2010, Horarios2009, Horarios2008, Horarios2007,  
Horarios2006, Horarios2005, Horarios2004, Horarios2003, Horarios2002, Horarios2001})
```

```
in
```

```
#"Consulta anexada"
```

11. Bibliografía

BOE, A. E. (18 de Octubre de 2008). *RD. 1073/2002*. Obtenido de <https://www.boe.es/buscar/doc.php?id=BOE-A-2002-20933>

BOE, A. E.-B. (27 de Septiembre de 1996). *Directiva 96/62/CE*. Obtenido de BOE: <https://www.boe.es/buscar/doc.php?id=DOUE-L-1996-81917>

BOE, A. E.-B. (22 de Abril de 1999). *Directiva 1999/30/CE*. Obtenido de BOE: <https://www.boe.es/buscar/doc.php?id=DOUE-L-1999-81189>

BOE, A. E.-B. (16 de Noviembre de 2000). *Directiva 2000/69/CE*. Obtenido de BOE: <https://www.boe.es/buscar/doc.php?id=DOUE-L-2000-82419>

BOE, A. E.-B. (12 de Febrero de 2002). *Directiva 2002/3/CE*. Obtenido de BOE: <https://www.boe.es/buscar/doc.php?id=DOUE-L-2002-80450>

BOE, A. E.-B. (18 de Octubre de 2002). *Real Decreto 1073/2002*. Obtenido de BOE: <https://www.boe.es/buscar/doc.php?id=BOE-A-2002-20933>

BOE, A. E.-B. (15 de Diciembre de 2004). *Directiva 2004/107/CE*. Obtenido de BOE: <https://www.boe.es/buscar/doc.php?id=DOUE-L-2005-80075>

BOE, A. E.-B. (15 de Noviembre de 2007). *Ley 34/2007*. Obtenido de BOE: <https://www.boe.es/buscar/act.php?id=BOE-A-2007-19744&p=20171223&tn=1>

Madrid, A. (2018). *Ministerio Medioambiental de Madrid*. Obtenido de <http://www.mambiente.munimadrid.es/opencms/opencms/cal aire/SistemaIntegral/SistInformacion.html>

Stonebraker, M. (11 de Octubre de 1943). *Wikipedia*. Obtenido de https://en.wikipedia.org/wiki/Michael_Stonebraker

Widenius, M. (3 de Marzo de 1962). *Wikipedia*. Obtenido de https://es.wikipedia.org/wiki/Michael_Widenius